DISSERTATIONES
INFORMATICAE
UNIVERSITATIS
TARTUENSIS
61

TOIVO VAJAKAS

Towards integration of mobile network data into analyzing human mobility

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

61

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS
61

TOIVO VAJAKAS

Towards integration of mobile network data into analyzing human mobility

UNIVERSITY OF TARTU
Press
1632

Institute of Computer Science, Faculty of Science and Technology, University of Tartu, Estonia.

Dissertation has been accepted for the commencement of the degree of Doctor of Philosophy (PhD) in informatics on October 1, 2024 by the Council of the Institute of Computer Science, University of Tartu.

Supervisors

Prof. Dr.	Eero Vainikko University of Tartu, Estonia
Assoc. Prof. Dr.	Amnir Hadachi University of Tartu, Estonia

Opponents

Prof. Dr.	Sidharta Gautama Ghent University, Belgium
Assoc. Prof. Dr.	Claudio Roncoli KU Leuven, Belgium Aalto University, Finland (visiting prof.)

The public defense will take place on November 28, 2024 at 12:15 in Narva Rd. 18-2046

The publication of this dissertation was financed by the Institute of Computer Science, University of Tartu.

ISSN 2613-5906 (print) ISSN 2806-2345 (pdf)
ISBN 978-9916-27-725-6 (print) ISBN 978-9916-27-726-3 (pdf)

University of Tartu Press
`http://www.tyk.ee/`

To my family and friends

ABSTRACT

This dissertation investigates the ways to improve the accuracy of human mobility assessments based on passive mobile positioning.

Four research questions are investigated: What cell-hopping suppression techniques are most effective for reducing the uncertainty in human mobility studies based on passive mobile positioning data? What is an effective model for handling overlapping cells in positioning data? What is optimal method to detect the stop episodes and locations in trajectory data? Do real-life cellplans exhibit "feeder-swap" distortion, i.e. the cellplan has sometimes the azimuth attributes assigned to wrong antenna?

This work compares several existing cell-hopping suppression techniques with the methodology devised by the autor where the cellhopping statistics from whole dataset is used as background knowledge while interpreting the individual trajectory. The developed methodology is applied to two separate datasets from different mobile operators. The improvement from the proposed method is up to 1.8 times.

The author proposes a probabilistic Bayesian model to describe the distribution of spatial probabilities in case of cell overlapping and tests it on limited test data. The application of this model reduces the uncertainty of location estimate up to 20%.

These theses describe a Continuous Time Markov model for estimation of the positioning location. The model is an improvement over prior baseline Switching Kalman model. Three improvements to the baseline algorithm are described. The parameters of proposed model are more closely semantically related to human mobility as compared to baseline. Uncertainty improvement up to 20% is demonstrated on test data.

An algorithm for detecting cellplan distortion is proposed to identify occurrences of "feeder-swap" distortion caused by incoorrect azimuth data in cellplan. The algorithm is tested on data of one mobile operator in Estonia, both in rural and urban area. The test results show that the algoritm is sufficiently sensitive on given dataset but no significant distortions of this kind are present in these data.

Several improvement attempts are presented in this theses and certain success is reported but the original ultimate goal to find a generic solution for optimal interpretation of the passive mobile positioning data is not fully achieved. Further research direction might be using more other data sources (geographical data, census data, transport schedules, etc) for more rich Bayesian models.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

CDR – Call Detail Records
CGI – Cell Global Identifier
CPB – cellplan-based ping-pong suppression
CPR – cell pair routing
CPR-PPS – Cell Pair Routing with Ping-Pong Suppression
CTSKF – Continuous Time Switching Kalman Filtering
DDR – Data Detail Records
GIS – Geographical Information System
GNSS – Global Navigation Satellite System
GPS – Global Positioning System
MC – Markov Chain
MLE – Maximum Likelihood Estimate
MS – Mobile Station
NOP – No Processing
NRMSE – Normalized Root Mean Squared Error
NSS – Network Switching Subsystem
OD – Origin-Destination (matrix)
PDF – Probability Density Function
PET – Privacy Enhancing Technology
PPS – Ping-Pong Suppression
RAM – Random Access Memory
RAN – Radio Access Network
ROC – Receiver Operating Characteristic (curve)
RTS – Algorithm named after Rauch, Tung, and Striebel
SINR – Signal-to-Interference-and-Noise Ratio
SKF – Switching Kalman Filtering
SPDF – Spatial Probability Density Function
TTQ – Transition Time Quantile

LIST OF ORIGINAL PUBLICATIONS

List of original publications directly related to the contents of this dissertation

I **Vajakas, Toivo**, Vajakas, J., Lillemets, R. (2015) "Trajectory reconstruction from mobile positioning data using cell-to-cell travel time information." International Journal of Geographical Information Science 29.11 (2015): 1941-1954.

Scientific contribution: We provide an overview of mobile positioning data and the distorting artifacts in data; we compare various techniques to suppress "cell-hopping" distortion in data; we devise a novel algorithm that utilizes the distribution of cell-to-cell travel times; we test the data on the algorithm on data from two MNOs.

We describe a universal pipeline for passive mobile positioning data processing that produces mobility-based reports. The pipeline is based on map-matching.

Author's contributions: The author raised the main algorithm ideas and hypotheses, devised the algorithms, defined the results to be calculated and evaluated, implemented some parts of code, designed data visualization, wrote the paper.

II **Vajakas, T.**, Rõõmusaare, J. (2016). On optimal spatial probability density estimation of passive mobile positioning events. In 2016 15th Biennial Baltic Electronics Conference (BEC) (pp. 127-130). IEEE.

Scientific contribution: We describe a novel Bayesian probabilistic model for location probability density estimate and demonstrate its applicability to experimental data.

Author's contributions: The author raised the main algorithm ideas and hypotheses, devised the algorithms, defined the results to be calculated and evaluated, implemented some parts of code, designed data visualization, wrote the paper except some mathematical notations devised by Jaan Vajakas.

III **Vajakas, T.**, Kiis, T., Hadachi, A., Vainikko, E. (2018). Mobility episode discovery in the mobile networks based on enhanced switching Kalman filter. In 2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT) (pp. 1-7). IEEE.

Scientific contribution: We identify some weaknesses of the prior art algorithm for episode discovery in passive mobile positioning data and propose an improved algorithm; we test the proposed algorithm on real mobile positioning data.

Author's contributions: The author raised the main algorithm ideas and hypotheses, defined the results to be calculated and evaluated, implemented

some parts of code, designed data visualization, wrote the paper. The mathematical methods and notations were devised by Tanel Kiis.

IV **Vajakas, T.** "A methodology to detect azimuth errors in cellplan."
Status: to be submitted.
Scientific contribution: We devised a technique to test if a cellplan provided by MNO contains gross azimuth errors. The technique was tested on real data of one mobile operator. The test results indicated that the given cellplan did not contain gross azimuth errors.
Author's contributions: The author raised the main algorithm ideas and hypotheses, devised the algorithms, defined the results to be calculated and evaluated, implemented some parts of code, designed data visualization, wrote the paper.

Other published works of the author

V Haav, H. M., Kaljuvee, A., Luts, M., Vajakas, T. (2009). Ontology-based retrieval of spatially related objects for location based services. In OTM Confederated International Conferences" On the Move to Meaningful Internet Systems" (pp. 1010-1024). Springer, Berlin, Heidelberg.

VI Haav, H., Kaljuvee, A., Luts, M., Vajakas, T. (2011). Ontology-driven Development of Personalized Location Based Services. In Databases and Information Systems VI: Selected Papers from the Ninth International Baltic Conference, DB & IS 2010 (Vol. 224, p. 3). IOS Press.

VII Mändar, H., Felsche, J., Mikli, V., Vajakas, T. (1999). AXES1. 9: New tools for estimation of crystallite size and shape by Williamson–Hall analysis. Journal of applied crystallography, 32(2), 345-350.

VIII Julge, K., Vajakas, T., Ellmann, A. (2017). Performance analysis of a compact and low-cost mapping-grade mobile laser scanning system. Journal of Applied remote sensing, 11(4), 044003.

IX Lang, M., Sims, A., Pärna K., Kangro, R., Möls, M., Mõistus, M., Kiviste, A., Tee, M., Vajakas, T., and Rennel, M. (2020) "Remote-sensing support for the Estonian National Forest Inventory, facilitating the construction of maps for forest height, standing-wood volume, and tree species composition." Forestry Studies 73, no. 1 (2020): 77-97.

1. INTRODUCTION

1.1. Motivation

Passive mobile positioning data, gathered by mobile operators as a byproduct of mobile radio access network (RAN) operations, describes the location of a mobile station (MS). MS is a device such as a phone or a modem for environment sensors or security devices. Such data has gained much popularity in human geography studies and transportation due to the availability of large samples and its potential in human mobility related applications [53].

Practically all applications of mobile positioning data include an intermediate processing step where individual positioning events are interpreted as the movement history of a single moving object. An important part of such analysis is annotating the time into stop and movement episodes. Typical examples of such applications are regional planning with origin-destination matrix estimation [15], traffic analysis [20], etc. The accuracy of this analysis depends on the reliable detection of movement and stop episodes.

Mobile positioning determines the position of an MS with significant uncertainty; also, the velocity of MSs is not directly observable. In addition, this type of data tends to be very sparse in time. From such data, it is easy to detect stop and movement episodes in situations where travel distance is significantly larger than uncertainties. For example it is trivial to detect travel between cities. However it is difficult to detect reliably shorter movements – for example, intra-city travel. Low spatial resolution is limiting usefulness of passive mobile positioning data [15].

For these reasons, it is very important to develop algorithms that detect as detailed movements and stops as feasible.

One most prominent characteristic of positioning data is the presence of very strong spatiotemporal autocorrelations between measurements, due to the continuous nature of the trajectory, repeated travel patterns of the same person, and similar route selection behavior for different persons. At the same time, it is common to passive mobile positioning data that the uncertainty distribution has very heavy tails – as a rough generalization from working with data from different mobile operators the author has impression that typically there are circa 70% of measurements that match the accuracy predictions based on assumed radio cell geometry provided by mobile operators, and the rest of data are outliers with spatial uncertainty that significantly exceeds these predictions.

The author formulated a hypothesis that due to the strong autocorrelation assumption, it is possible to effectively detect and filter out most of the outliers and thus one can improve the accuracy of the movement detected from passive mobile positioning data. This work presents some efforts to pursue this goal.

1.2. Research questions and contribution

Main research question: Do relatively universal techniques exist to improve the accuracy of human mobility assessments based on passive mobile positioning?

Under this main question, following sub-questions were investigated

1. *Research question:* What cell-hopping suppression techniques are most effective for reducing the uncertainty in human mobility? Can we improve the cell-hopping suppression in one given trajectory by using the distribution of features in the whole available dataset?
   Contribution: The author devised a method for comparing the effect from different cell-hopping algorithms based on the effect on known traffic mobility estimates. The author also devised a new technique utilizing the cell-hopping statistics in a mobile positioning dataset and investigated how this technique improves the accuracy of vehicle mobility estimates. The developed methodology was applied to two separate datasets from different mobile operators. The improvement was up to 1.8 times.
2. *Research question:* What is an effective model for handling overlapping cells in positioning data? How much effect has cell overlapping on location uncertainty?
   Contribution: The author devised a probabilistic Bayesian model to describe the distribution of spatial probabilities in case of cell overlapping and applied it to limited test data. The application of this model reduced the uncertainty of location estimate up to 20%. The model describes one aspect of probabilistic position estimation that was previously not covered.
3. *Research question:* What is optimal method to detect the stop episodes and locations in trajectory data?
   Contribution: We analyzed and improved the baseline algorithm described in [8]. We found three aspects of the algorithm where we could make semantically justified improvements to the algorithm. The improved algorithm was implemented and tested on real data. On some data the improvement was significant. The changes to model also brought it closer to human mobility semantics (e.g., state transition frequency is related directly to how often human phone-owner switches to different behavior mode, per hour) as opposed to technical semantics of baseline method (e.g., state transition frequency is related to how many positioning events RAN creates before state change).
4. *Research question:* Do real-life cellplans exhibit "feeder-swap" distortion, i.e. the cellplan has sometimes the azimuth attributes of cells swapped?
   Contribution: We developed a methodology to find feeder-swapped cells by erratic trajectory behavior of MS. The sensitivity of the test was evaluated on the data with artificially swapped cells. The methodology was used on

real data from one operator in Estonia. No swapped cells were found in real data for an Estonian MNO. The method could be used to check cellplan quality based on passive mobile data. It is easily applicable as it does not require additional data except road network data.

2. BACKGROUND

2.1. General characteristics of mobile positioning data

Mobile positioning is a byproduct of the operations of Radio Access Network (RAN). Radio signals between RAN and MS are served via transceivers that are located in radio towers. The geographical area served by one transceiver is called a cell. Each cell has a unique identifier called Cell Global Identifier (CGI). The extent of a cell are defined by the probability of connecting to a given transceiver of a base station. Usually, cells are modeled as polygons or ellipses representing an approximate area of significant connection probability. Passive mobile positioning is based on observing the operations of RAN. RAN has information on which transceiver is used by MS to connect to a network (Fig.1). This information can be saved as passive mobile positioning events (Table 1). Passive positioning denotes data collection where no specific activities are initiated in RAN for the sole purpose of acquiring the position of MS. Active positioning could produce more accurate results, but large-scale active positioning is not feasible due to RAN load overhead [16].

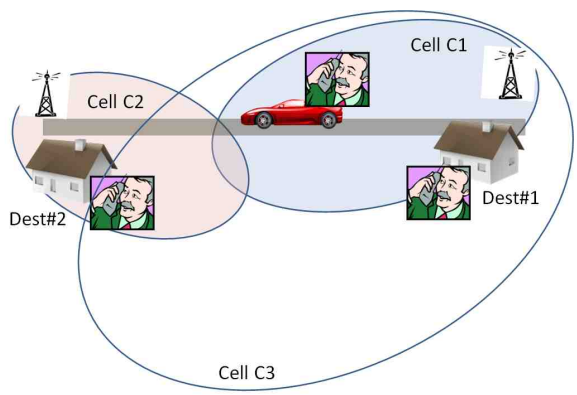


Figure 1. Example of radio network and positioning. The tower on the left has an antenna with cell area C2. The tower on the right side has antennas with cell areas C1 and C3. If, within given time interval, the phone is seen both in C1 and C3, then the phone might still be in same stop location. When the phone was first connected to C1 and after that to C2, then probably the phone moved.

For mobile positioning, it is essential to know the geographical shape of each cell. A detailed model of the cell shape would be a spatial probability density function (SPDF), which also varies over time. The actual cell shape depends on

Table 1. Example of passive mobile positioning data

timestamp	pseudonym	CGI
2018.03.22 14.33.22	12377	11-22-33-44444
2018.03.22 14.35.54	978210	11-22-44-55555

many factors, such as antenna radiation pattern and height, network load [52], signal attenuation on the landscape and indoors [47], signal reflections, radio interference and noise [61], and network configuration parameters such as handover threshold and neighbor cell lists [5]. In practice the exact SPDF is unknown and spatial uncertainty is modeled as the increased size of cells.

Passive mobile positioning is recorded as Call Detail Records (CDR) or Data Detail Records (DDR). The primary purpose of such data is billing. DDRs are produced more frequently than CDR, but only during the open data session. A network switching subsystem (NSS) stream of events also includes location updates, and network connection and disconnection events. Periodic location update events periodically report (typically every hour or two) the current location of a mobile phone and improve coverage of its location. The network also generates location update events when a phone moves from one location area (a group of closely situated base stations) to another. Further on, we refer to data from any of these sources as CGI data. In all cases, the relevant fields in data are timestamp, pseudonym of MS, CGI of cell that was connected that time.

The time between two consecutive events in the mobile network can differ greatly between different devices and throughout the day for any particular device. While actively using mobile Internet, the network can register several events per minute, but when the device stays idly, there can be several hours between events.

2.1.1. Cell-hopping

CGI data has specific 'ping-pong' noise caused by several cells being simultaneously connectable from the same location with seemingly random handovers occurring between these cells. Geographically static device exposed to spurious handovers between several cells can be perceived by the mobile network as highly mobile [20].

2.2. Basic characteristics of human mobility

Human mobility follows certain patterns. These patterns are relatively stable over different communities and locations, albeit with local variations [55], [22], [50].

Typically humans are interested in moving from a destination location to another destination as economically as possible. The mobility has different timescale patterns – daily, weekly, seasonal, lifespan of human actor going through phases of a small child, student, working, retired.

In mobility studies, the general patterns can be used as prior distribution in Bayesian statistics sense. One distinctive feature of human mobility is that some aggregate quantitative characteristics over a large number of people are rather stable and easy to describe [22]. The trajectories of individual persons require intimate knowledge of spatial, social, and personal circumstances to be fully un-

derstood – there are irregularities that are not easily explained in detail without such knowledge [55].

The majority of applications of human mobility are not interested in travel *per se* but in the intentions and actions taken by individuals. Human geography is focused on relations between spatial and other aspects of human life. Human geography has tried to apply the methods from physics to spatiotemporal human behavior, and on very large scale has some promises [27], but it remains superficial and cannot fully model the richness of motivations why people behave as they do, including movement. Explicit modeling of individuals is addressed by agent-based modelling [18] where main difficulty is assigning realistic motivations and relational patterns to the agents, i.e., modelling the personality of the individuals. Attempts to describe the functioning of individuals in society are addressed in more psychological and philosophical approaches, e.g. [32], [17], but these are rather open-ended and philosophical and challenging to apply in concrete modeling.

2.3. Mobile positioning and map-matching

There are many publications about trajectory reconstruction, also known as map-matching.

Over time the scope and goal of map-matching have changed, reflecting new use cases, the increased available processing power, and emerging more complex algorithms. The definition of map-matching has evolved also. In earlier days, the goal could be defining the road segment the vehicle is on [9]. Nowadays, the goal is usually to estimate the actual position and speed of the vehicle at any time moment [41]. We use following the latter definition. An important aspect of map-matching is detecting stop events, i.e., time intervals where the positioned MS is stationary.

Many different techniques for map-matching have been develeped based on conditional random fields [25], Extended Kalman Filter [40], Dempster–Shafer's theory of evidence [19], particle filters [23], Bayesian inference [35]. Trajectory reconstruction usually includes establishing the relationship between the trajectory and the road network; such a technique is called map matching [43]. An overview of the field of map matching is given by [45].

The wide availability of positioning techniques have become available over the last decades, and this has raised the need for the interpretation of raw data and giving it meaning. Map-matching algorithm can be described as a way to reconcile inaccurate locational data with an inaccurate map/network [9]. Over the years, the areas where positioning techniques are applied have become more and more widespread.

In new emerging application areas, specific techniques have been developed in an attempt to find the algorithm fitting best for particular data available and optimized for specific goals, e.g., real-time personal navigation [64], batch processing

for traffic, and human mobility analysis [7].

The majority of publications are oriented towards use-cases where only GNSS data is available [45], but numerous other data sources are also used either individually or combined – WIFI signal, mobile positioning, inertial navigation system, 3D model of city landscape [59]. In the context of this work, the main focus is on map matching using passive mobile positioning data and batch processing.

2.3.1. Interpreting cell-hopping data

Several techniques have been developed to reduce 'ping-pong' distortions in interpreting CGI data [20] [49] [14]. Fiadino et al. [20] presented a study on trajectory reconstruction, where they used a 'ping-pong' suppression method that ignores events where the device connects back to the previous cell within a predefined time window (transformation of subsequence in the event history $ABA \rightarrow AB$). Schlaich et al. [49] describe a method for computing edit distances between event sequences where short-term handovers to another cell and repeated events can be ignored (transformation pattern $ABA \rightarrow A$). For CGI data Calabrese et al. [14] used a method that was inspired by earlier work for GPS data by Ye et al. [62] and Krumm [31], which performs a clustering of measurement points and replaces original events with the barycenter of the cluster. The techniques have been introduced *ad hoc* for obvious data cleansing purposes. No explicit formalized goal was defined for selecting the best possible technique and parameters.

2.3.2. Evaluation and improvement of cellplan quality

Based on available information, the shape of each cell has to be defined to give location estimates for mobile positioning. Cell data provided by mobile operators can be translated to cell shapes as Voronoi polygons by using the assumption that a phone connects to the nearest tower (e.g, Ahas et al. [2]); as best server data polygons by using the assumption that a mobile phone connects to the cell with the strongest signal [12]; or as a raster model based on the assumption that the probability of connecting to a cell is a function of distance from the antenna tower [11]. In related literature, the identification of possible distortions in cell shapes has not been considered as part of the trajectory reconstruction problem, except for the cleaning procedure that excluded obviously erroneous cells [20].

2.4. Traffic and Origin-Destination reports

Caceres et al. [10] provide an overview of techniques used to obtain traffic parameters (e.g., speed, travel times, flow) based on information from a mobile phone service provider. They concluded that for traffic flow evaluations, location update events are more suitable than CDR data. Schlaich et al. [49] investigated trajectory reconstruction using only location update events, spatially location was defined on location area level, ignoring individual cells. One location area is a set

of cells, used by MNOs to partition large network into smaller partitions. The traffic flow estimates for German highways were reported to correspond to automatic number plate recognition (ANPR) measurements. The measurements used only trajectories that included at least three location areas (approximate lower limit of trajectory length equaled 20km). There are several studies concerning the detection of the origin-destination (OD) matrix from mobile data [37]. Calabrese et al. [13] estimated OD matrix from mobile positioning data and validated it against census data. OD matrix and traffic flow estimation problems are closely related. It has been common practice to use traffic flow to derive an approximate OD matrix, see Abrahamsson [1] for an overview. Traffic flow can be derived from an OD matrix using trajectory estimation. Iqbal et al. [26] developed a method to measure an origin-destination matrix from CDR data, estimated traffic flow from the OD matrix, and compared it with traffic counter data.

3. MAP-MATCHING WITH CELL-HOPPING SUPPRESSION

3.1. Motivation and problem statement

Cell-hopping is one of the most prominent artifacts in passive mobile positioning data. Cell-hopping is expected in situations where cells have significant overlap. In practice, one can observe cell-hopping also between the cells that have significant distance and consequently unrealistic velocity of apparent movement. This kind of contradiction between expected data patterns and observed data patterns leads to distorted and unreliable results in analysis – e.g., the phantom movement of cell-hopping can increase traffic estimates in certain areas. Therefore it is important to minimize the effect of cell-hopping distortion on reconstructed trajectories.

In literature two major approaches are used – detecting clusters and replacing cluster with single location, or replacing several events by single event using some pattern replacement rule. There are many techniques proposed for cell-hopping suppression based on these two approaches, each technique with their own differences. At the same time there is no comparison available of the optimality of various techniques and parameter values.

Problem statement: Passive mobile positioning data contains specific noise that makes human mobility estimates significantly less reliable and less accurate compared to the theoretical spatial resolution of cell-based location estimates. Removing that noise would substantially improve the value of mobility reports based on passive mobile positioning data.

Research question: What cell-hopping suppression techniques are most effective for reducing the uncertainty in human mobility? Can we improve the cell-hopping suppression in one given trajectory by using the distribution of features in the whole available dataset?

3.2. Our trajectory reconstruction method

This section first introduces our trajectory reconstruction algorithm, which we call Cell Pair Routing with Ping-Pong Suppression (CPR-PPS). It presents notations for its input and output data (Subsection 3.2.1) and then the algorithm itself with its variations that we tried, namely different ping-pong suppression techniques (Subsections 3.2.2 and 3.2.3).

3.2.1. The input and output of the algorithm

The CPR-PPS algorithm needs the following inputs.

The first input is a sequence of events $E_1, \ldots, E_m$ for each MS where each event E has two attributes: a timestamp t_E and the ID of the cell C_E where the

event occurred. We call the sequence events for one MS the cell-trajectory of the MS. The events in a cell-trajectory are assumed to be sorted by their timestamps.

The second input is a cellplan: a mapping that to each cell C of the mobile network assigns a polygon S_C, the cell shape – an approximation of the area where MSs can connect to this cell. The third input is a road network graph. One PPS variant (TTQ) also needs a fourth input, the cell transition time quantiles (see the description of TTQ in Subection 3.2.3).

The output of the CPR-PPS algorithm is for each MS a continuous possible trajectory in the road network. A possible trajectory determines the coordinates and road segment where the MS is located at any moment. The probabilistic algorithm generates possible trajectories. Our objective when creating the algorithm was not to find the possible trajectory with maximum likelihood, but to generate possible trajectories from a probability distribution that models uncertainty of our knowledge as well as possible. In the context of computing traffic flow, the true objective is clearly the second one. Indeed, if for some cell-trajectory MSs with such cell-trajectory actually choose several different routes, then the correct traffic flows would result if the trajectory reconstruction algorithm generated the possible trajectories with the right probabilities, not if it always chose the same trajectory, even the most likely one.

3.2.2. The CPR-PPS algorithm

The CPR-PPS algorithm consists of two phases: the ping-pong suppression (PPS) algorithm cleans up the cell-trajectory by removing some events; the cell pair routing (CPR) algorithm generates possible trajectories from the resulting cell-trajectory.

The PPS algorithm decides whether two observed consecutive events from the same MS, but connected to different cells, are the result of a handover without physical movement of the MS or the cells are too far apart to be accessible at the same time, and removes events from the trajectory in the former case. We considered several variants of PPS, which are described in Section 3.2.3.

The CPR algorithm is as follows: for every event E in the cell-trajectory, we choose a random point in its cell C_E; then, for every two events, we computed the fastest path in the road network between the corresponding points and regarded the path as a possible trajectory for the MS. More precisely, a random point x is chosen from a cell C as follows: firstly, a random point x_0 is chosen from the uniform distribution on the cell shape S_C; then the nearest road segment r to x_0 is found; x is chosen as the endpoint of r that is closer to x_0. The MS is assumed to have taken the fastest path between the two points; the path is stretched in time to cover the time interval between the two events, with each road segment's traversal time proportional to its traversal time at its nominal speed in the road graph. For this algorithm to perform well, it is crucial to compute the fastest path efficiently. To this end, we created a Contraction Hierarchy index and precomputed the back-

ward and forward search spaces for each node in the road graph (Geisberger et al., [21]).

3.2.3. Variants of ping-pong suppression

We considered five PPS techniques, described below.

1. First, the simplest variant of PPS, no processing (NOP), removes only events whose cell is the same as that of the previous event.
2. The second variant, cellplan-based ping-pong suppression (CPB), checks for cell shape overlap to determine whether an appearance of events with different cell IDs in a cell-trajectory was likely the result of an actual movement. The CPB algorithm works by dividing the cell-trajectory into episodes of consecutive events and removing all except one event from each episode, as follows. Let $E_1, \ldots, E_m$ be the cell-trajectory before applying PPS. To begin with, we denote the index of the first event of the first episode by $k_0 = 1$. We find the smallest integer $k_1 > k_0$ such that cells $C_{E_{k_0}}$ and $C_{E_{k_1}}$ do not overlap, where we say that two cells C and C' overlap if their shapes have non-empty intersection: $S_C \cap S_{C'} \neq \varnothing$. If such an integer does not exist then we set $k_1 = m$. The events $E_{k_0}, \ldots, E_{k_1-1}$ will form the first episode. If $k_1 < m$, the algorithm will continue to determine the other episodes. We find the smallest integer $k_2 > k_1$ such that the shapes of cells $C_{E_{k_1}}$ and $C_{E_{k_2}}$ do not overlap (with $k_2 = m$ if there is no such integer) and form the second episode from events $E_{k_1}, \ldots, E_{k_2-1}$, and so on until the end of the cell-trajectory. For each episode, we choose the representative event to be the event among the events of the episode whose cell's shape's centroid is closest to the centroid of the centroids of the shapes of the cells of all the events in the episode. The CPB algorithm is similar to the technique applied by Calabrese et al. [14].
3. The third option, TTQ (transition time quantile based ping-pong suppression), is an enhancement of the CPB techniques: as the shapes in the cellplan are often not reliable (usually too small), we tried to compensate for that by using real event data, as follows. If on a trajectory of an MS there are two consecutive events E and E' such that $C_E = C$ and $C_{E'} = C'$, then we call the pair (E, E') a transition from C to C' and $t_{E'} - t_E$ the transition time. From the trajectories of all MSs, a histogram of transition times from cell C to cell C' is computed. Let $t(C, C', \alpha)$ be the α-quantile of the histogram (e.g. if $\alpha = 0.05$ then 5% of the transitions between C and C' were shorter than $t(C, C', \alpha)$). We let α be a tuning parameter of the technique. Now we say that the cells C and C' overlap if and only if $S_C \cap S_{C'} \neq \emptyset$ or $t(C, C', \alpha) < t_O$, where t_O (*overlap threshold time*) is another tuning parameter. The TTQ technique is CPB with this modified definition of overlap. Therefore, if we frequently observe almost instant handovers from one cell to another, then we can assume that these fast handovers are not a result of significant actual

movement but that the cells have significant overlap, and the TTQ algorithm removes such transitions from the cell-trajectory.

4. The fourth and fifth technique that we considered are simple algorithms found in the literature. The fourth, the $ABA \rightarrow AB$ technique, traverses the cell-trajectory starting from the oldest event and applies two rules. Firstly, it removes an event if it has the same cell as the preceding event (as the NOP technique). Secondly, it removes the third event from a sequence of three events where the third event's cell is the same as that of the first event (an 'ABA' pattern) if the time between the first and the third event is less than the time window length t_W, a tuning parameter. For example, given a cell-trajectory consisting of eight events with cell IDs ABBABACA such that the events fit within the time window, the $ABA \rightarrow AB$ technique reduces it to a cell-trajectory of four events: $ABBABACA \rightarrow ABABACA \rightarrow ABBACA \rightarrow ABACA \rightarrow ABCA$. The $ABA \rightarrow AB$ technique was used by Fiadino et al. [20] under the name 'ping-pong filter'.
5. The fifth PPS variant, the $ABA \rightarrow A$ technique, also applies recursively two rules: it replaces two consecutive events in the same cell with the first event (as the NOP technique) and three consecutive events in an ABA pattern, which are within a time window of length t_W, with the first event. For example: $ABBABACA \rightarrow ABABACA \rightarrow ABACA \rightarrow ACA \rightarrow A$. Schlaich et al. [49] used the $ABA \rightarrow A$ rule for computing edit distances between cell-trajectories.

To illustrate the motivation for the TTQ technique, Figure 2 shows a typical distribution of observed transition time. The darker histogram corresponds to a pair of cells separated by a transition time of ca. 15 minutes. The lighter histogram corresponds to a pair of cells that exhibit instant handover, i.e., the separated shapes described in the cellplan are incorrect. It is more difficult to find out which of the two cells has an underestimated shape in the cellplan.

Figure 3 shows an example of trajectory reconstruction using different PPS variants: CPB excludes C5 as it has explicit overlap with C4 in the cellplan. TTQ excludes C5 for the same reason as CPB and additionally excludes C3 as it actually statistically overlaps with C2 (detected from transition time statistics). Figure 3 also shows short sideways movement artifacts caused by the naivety of the CPR-PPS algorithm, as it does not distinguish actual stops in cells and transit through cells. The trajectory would be more realistic if it did not leave the main road because of transit cells, as the main road provides a more economical route.

3.3. Experiment design and validation criterion

We evaluated the CPR-PPS algorithm, and, in particular, the different choices for PPS, by using the generated trajectories to compute estimated values of traffic flow and comparing the estimated traffic flow with measurements of traffic flow

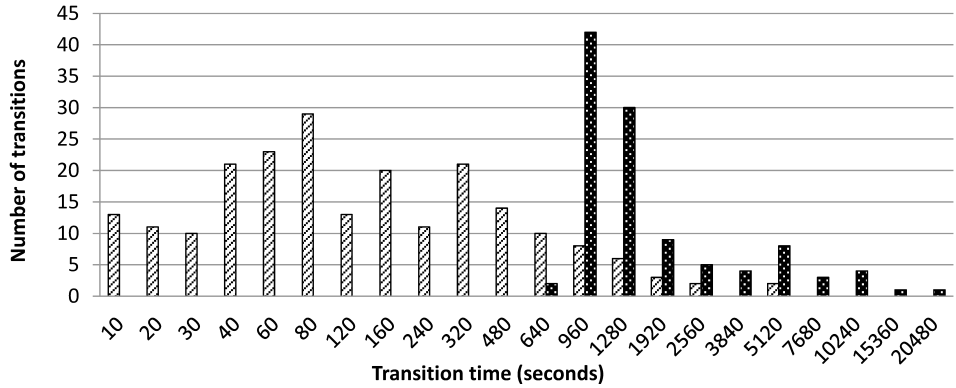


Figure 2. Sample histograms of transition time between two pairs of cells. Both pairs have an intercell distance of 24 km, as calculated from the cellplan. Vertical axis: count of transitions between the cells in a cell pair. Horizontal axis: time in seconds. NB! The bin width is larger for longer times.

sensors. This section describes the data and methods used.

3.3.1. Description of the data used in the experiments

a) Mobile positioning data. We used mobile positioning data from two Estonian mobile operators. The mobile operators preferred to be not named. The processing was performed in pre-GDPR era, i.e., the regulations and practices were a bit less strictly defined. Later the data was deleted as required by data retention agreement. Therefore it is not possible to repeat the calculations on exactly same data.

The experiments were run for two 24-hour periods: the main test period was a Monday; we additionally tested on a Sunday six days later. In order to calculate traffic flow for the 24h test period, we reconstructed the trajectories over five days – including 48h before the test period and 48h after it – to avoid any possible edge effects (but we skipped routings outside the 24h test period). Table 1 shows the main parameters of the data. The data from Operator 1 included location update events, and therefore was expected to be more uniformly distributed over subscribers, and indeed the median number of events in trajectory and median time coverage (hours that have at least one event) was higher for Operator 1. Surprisingly the number of events per person was lower than expected in presence of location update events. The data of Operator 2 consisted of CDR (25%) and DDR (75%), there were no location update events included. The trajectory reconstruction handled all events regardless of technology (2G, 3G) or origin (CDR, DDR, location update) as a single homogenous stream of events.

b) Traffic counter data. The traffic counter data originates from stationary counters operated by Estonian Road Administration (Maanteeamet). Considerable volume of data was missing values or values with questionable timestamp. Fig.5 shows overall pattern of missing values.

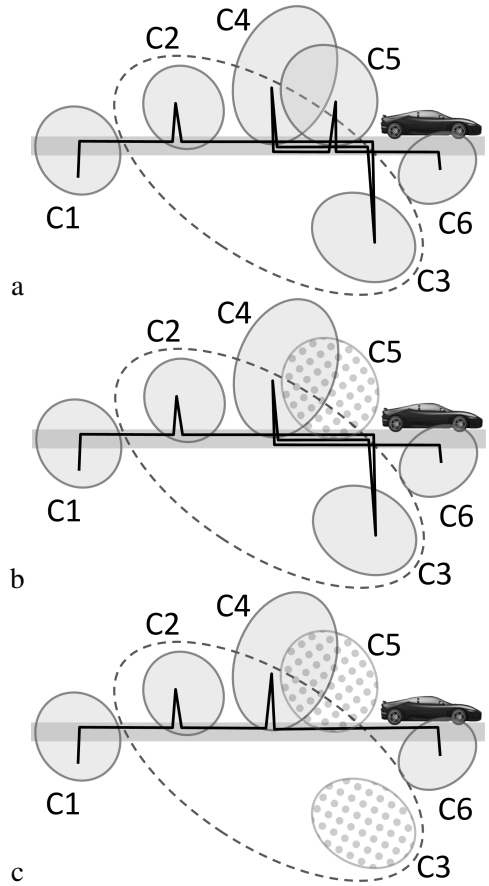


Figure 3. Illustration of trajectory reconstruction with different trajectory reconstruction algorithm variants, based on a vehicle moving along a main road (shown as the horizontal stripe), and generating a cell-trajectory of C1, C2, ..., C6. The filled ellipses mark the cell shapes as determined by the cellplan. The dashed ellipse is the real shape of C3, which differs significantly from the shape in the cellplan. Dotted filling marks cells that are excluded by the trajectory reconstruction algorithm. The reconstructed trajectory (black line) is shown for three different algorithm variants: (a) NOP (overlap only with itself); (b) CPB (cellplan-based overlap); (c) TTQ (transition time quantile based overlap).

After removing the suspicious records the cleaned data seemed consistent – examples of time series at various locations are given in Fig.6, 7. Geographical distribution of traffic counters is depicted on Fig.4.

c) Road network data. We used a road network graph from the GIS company Regio AS for routing. We compared the calculated traffic flow estimates to traffic loop sensor data from the Estonian Road Administration. In the road network graph, Regio AS had divided the road segments into 12 road classes, ranging from internationally important roads to very narrow streets and accessways. The road segments where the sensors had been installed belonged to the four highest road classes: 35 sensors on class 1 (main road that belongs to the international

Table 2. Description of input data

Parameter	Operator 1	Operator 2
Number of cell-trajectories in test dataset	370 000	340 000
Presence of user activity events (call, SMS)	Yes	Yes
Presence of location update events	Yes	No
Presence of data detail events	Not	Yes
Median event count per cell-trajectory	13	8
Average event count per cell-trajectory	25	32
Median time coverage hours	7	5
Average time coverage hours	7.6	7.4
Total number of events per day, millions	10	10

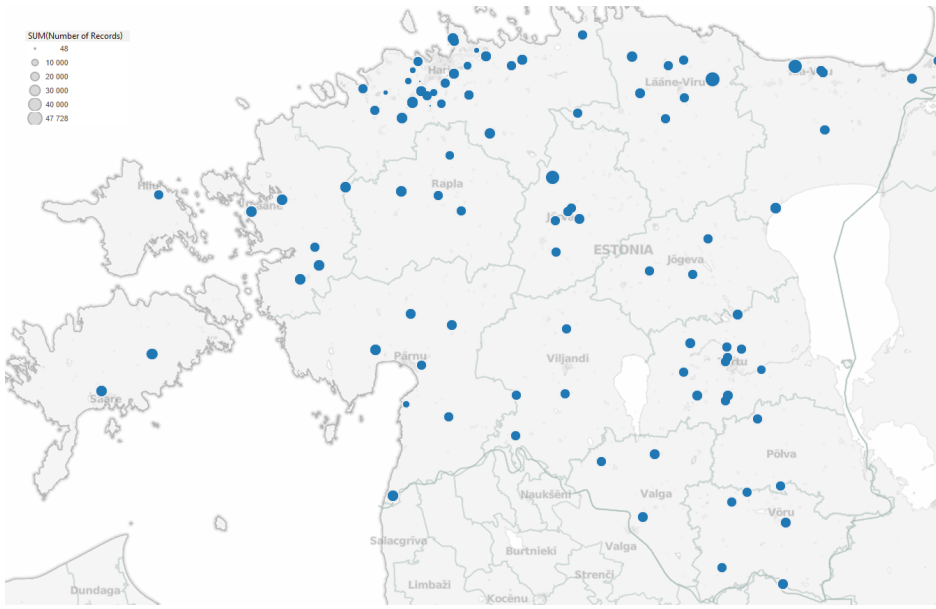


Figure 4. Geographical locations of traffic counter. Each blue ring marks measurement location. Each location had separate counters in different directions. The diameter of ring expresses total count of vehicles counted, including both directions. The effects from missing values are not corrected.

E-road network), 20 sensors on class 2 (main road), 32 sensors on class 3 (basic road) and 5 sensors on class 4 (local road) roads.

The sensors lay in near-urban and rural areas throughout Estonia. Schlaich et al. [49] noted that their traffic volume estimates were more reliable on highways than non-highway roads; therefore, in our experiment, we fitted traffic flow both for each class separately as well as for the whole dataset. We removed from the dataset the five sensors on class 4 roads, as there were not enough of them to be representative of their road class.

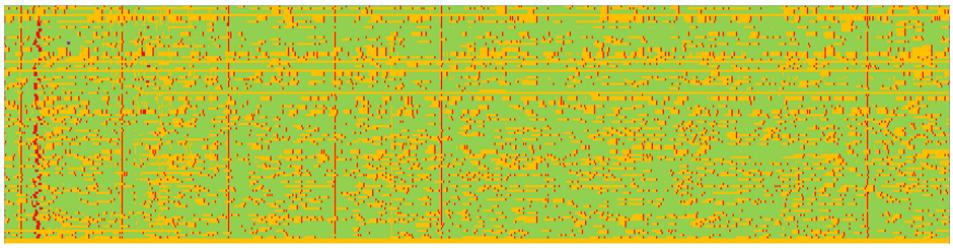


Figure 5. Data coverage example for 18 days period. Each row of pixels corresponds to one sensor (one driving direction in one location). Each column corresponds to 15 minute time period. Colors: green – in traffic data exactly one traffic density value for given period; yellow – no data for given period; red – more than 1 value for given period.

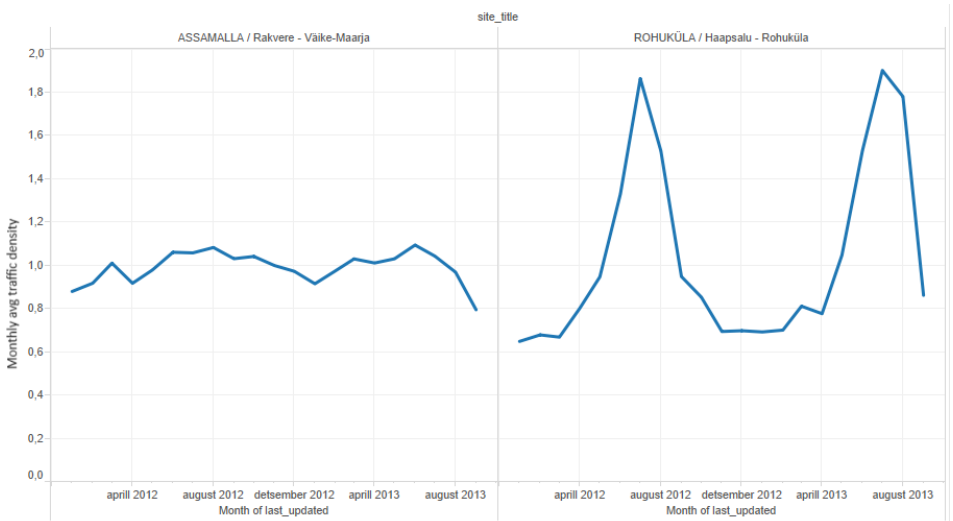


Figure 6. Examples of traffic density time series for traffic counter locations with weak and strong seasonal effect. Monthly average of traffic density (arbitrary units)

3.3.2. Methodology of the experiments

We compared the five ping-pong suppression techniques defined in Section 3.2.3:

- NOP – takes all events as provided by history (has no parameters)
- $ABA \rightarrow A$ – suppresses forward ping-pong (has tuning parameter t_W)
- $ABA \rightarrow AB$ – suppresses backwards ping-pong (has parameter t_W)
- CPB – clusters events by overlap of cells (has no parameters)
- TTQ – clusters events by overlap and time distance (has parameters t_O and α)

We evaluated traffic flow estimation accuracy with each technique by computing the NRMSE indicator defined below. For each technique with tuning parameters, we tried several parameter values to find the optimal ones. For $ABA \rightarrow A$ and $ABA \rightarrow AB$, we used $t_W \in \{1s, 120s, 1h, 4h, \infty\}$. For TTQ we used $\alpha \in \{0.01, 0.03, 0.1\}$ and $t_O \in \{10s, 30s, 60s, 120s, 240s, 480s, 900s\}$.

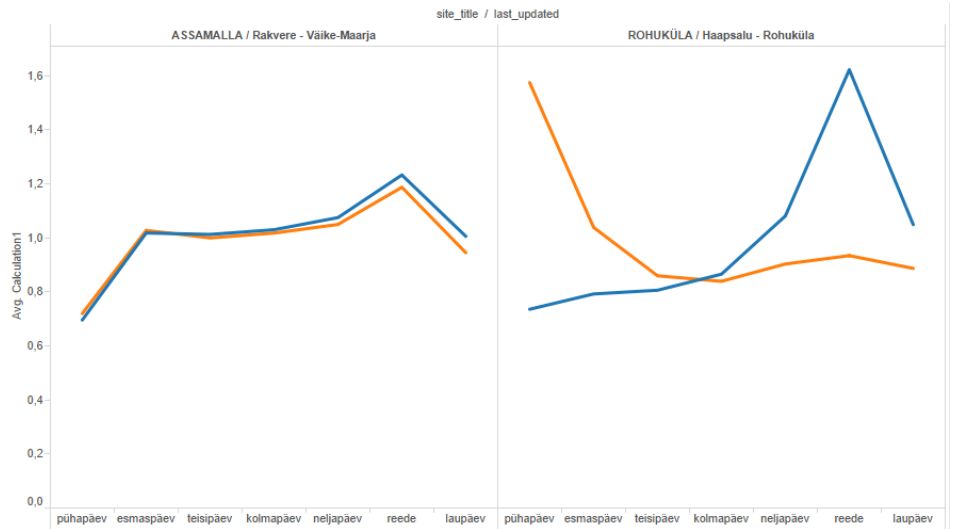


Figure 7. Example of traffic flow counter time series for locations with similar traffic in both directions and asymmetric traffic location.

3.3.3. Traffic flow estimation procedure and accuracy indicator NRMSE

For each traffic sensor, we found its corresponding road segment in the routing graph. For each road segment with a traffic sensor, we estimated from the mobile data its traffic flow, i.e. the number of traversals of a road segment by vehicles during the 24-hour test period. To this end, we used CPR-PPS to generate a possible trajectory for each MS. For each road segment with a traffic sensor, we counted how many times the possible trajectories of all MSs traversed the road segment. Thus we obtained for each road segment a number $x_{estimate}$ estimating how many times the road segment was traversed by MSs. We assumed that traffic flow is proportional to the number of traversals by MSs; we determined the scaling factor λ empirically, as described below.

Let y_{actual} be the actual traffic flow as measured by a sensor and $y_{estimate}$ our estimate of the traffic flow at that sensor.

$y_{estimate}$ is defined by linear approximation with one parameter λ over actual data points, where the fitting line is having no offset:

$$y_{estimate} = \lambda \cdot x_{estimate} \tag{3.1}$$

where scaling factor λ is defined as

$$\lambda = E(x_{actual} \cdot y_{actual}) / E(x_{actual} \cdot x_{actual}) \tag{3.2}$$

As a criterion of the quality of the estimation we used the normalized root mean error squared (NRMSE) over the sensors:

$$\begin{aligned} NRMSE = ((E(y_{estimate} - y_{actual})^2)/(E(y_{actual})^2))^{1/2} = \\ ((E(\lambda \cdot x_{estimate} - y_{actual})^2)/(E(y_{actual})^2))^{1/2}. \end{aligned} \tag{3.3}$$

Hollander and Liu [24] recommend using mean square error as a measure of fit over mean absolute error, as squared error places higher penalty on larger errors. We chose the scaling constant λ such as to minimize NRMSE. We computed the scaling constant and the corresponding NMRSE for the whole dataset (87 sensors) and for road classes 1, 2 and 3 separately. The software was written in Java and the tests were run on a 2 $\times$ Intel® Xeon® CPU E5-2650@ 2.00GHz using 12 GB RAM. Processing 5 days of event data (50 million events) and calculating traffic flow estimates with one of the investigated techniques took below 90 seconds.

3.4. Results

We describe one main experiment in greater detail. For some variations of experiment, we report only major differences in results.

3.4.1. Main experiment

In the main experiment, the test period was a Monday. Figure 8 shows the NRMSE values for each technique, road class and operator. The corresponding scaling constants and optimal parameter values are given in Table 3. The TTQ technique had consistently the lowest NRMSE, except for Operator 2 and road class 2 where it was still close to the best result (NRMSE was higher than the best alternative by 0.02). Figure 9 presents scatterplots of the actual traffic flow and TTQ's estimated number of traversals by MSs at each sensor. Most traffic flow sensors had relatively low values and only a few large values were observed. From Table 3 one can see that the scaling constant of TTQ was the highest among all the techniques for all road classes for both operators, indicating that this technique removed more 'ping-pong' movements than the others. Figure 10 shows maps of the error of the TTQ technique for both operators. Near large cities (Tallinn and Tartu) and in the countryside the TTQ technique had a tendency to respectively over- and underestimate traffic flow. The correlation between NRMSE and population density is not very clear – there were roads with underestimated traffic flow on the outskirts of cities and overestimated roads in the countryside.

3.4.2. Effects of filtering trajectories by event count

We repeated the test by removing input trajectories with event count in the 24-hour test period below some limit, and in another test run by removing trajectories with event count above some limit. We used several different limit values: lower limit 1, 2, 3, 10, 30, 100 and upper limits 30, 300, 3000 events per 24h. We expected to see improvement of NRMSE when extreme trajectories are excluded. TTQ remained the best technique during this experiment. Heavy filtering of input worsened the NRMSE value for the TTQ technique, as expected. Slight filtering caused modest improvement to the NRMSE for TTQ. For most of the eight operator and road class combinations (two operators, three road classes plus

Table 3. Traffic flow estimation scaling constants and best-fitting parameter values

Test	**Scaling constant λ**					**Best parameter values**			
	NOP	CPB	TTQ	$ABA \rightarrow AB$	$ABA \rightarrow A$	α for TTQ	t_O (sec) for TTQ	t_W for $ABA \rightarrow AB$	t_W for $ABA \rightarrow A$
Operator 1									
All sensors	0.58	0.91	1.54	0.81	1.05	0.01	240	∞	∞
Class 1	0.75	1.07	1.55	0.94	1.18	0.01	60	4 h	∞
Class 2	0.46	0.64	1.25	0.63	0.81	0.01	480	∞	∞
Class 3	0.40	0.76	1.58	0.62	0.87	0.01	240	∞	∞
Operator 2									
All sensors	1.36	1.56	2.00	1.69	1.95	0.01	30	∞	∞
Class 1	1.28	1.47	1.98	1.62	1.86	0.01	60	∞	∞
Class 2	1.57	1.76	2.02	1.57	1.75	0.1	60	0	1 h
Class 3	1.71	2.07	2.56	1.96	1.97	0.1	120	4 h	1 h

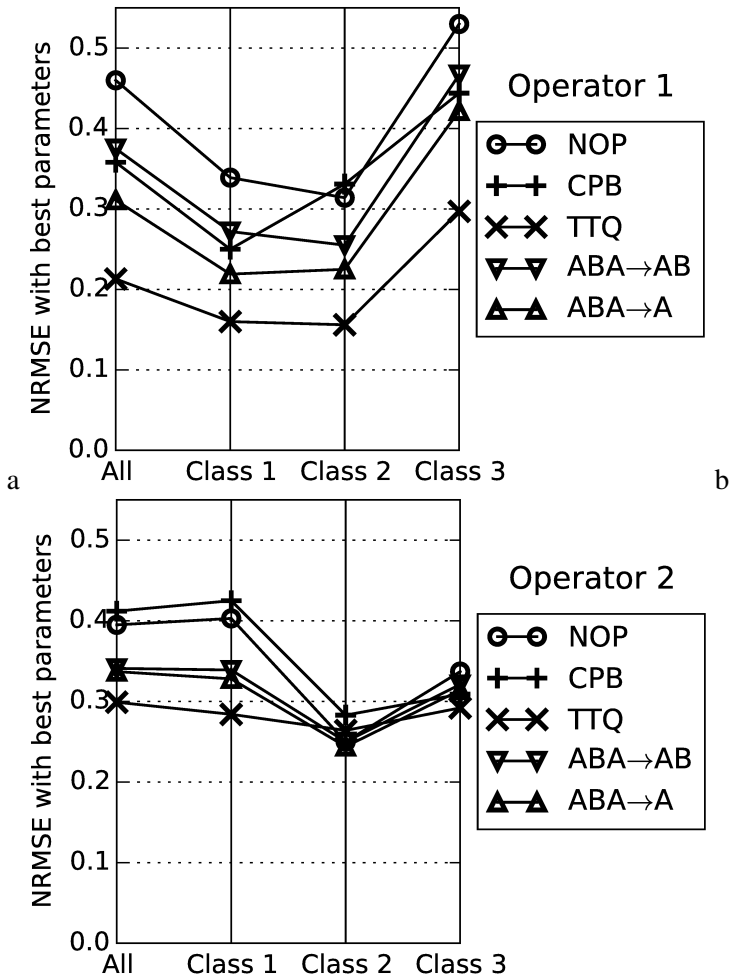


Figure 8. Prediction quality of compared techniques: NRMSE values of all techniques (with optimal parameters) for all sensors together and for sensors in each road class separately, for (a) Operator 1; (b) Operator 2.

all road classes), the TTQ NRMSE improvement did not exceed 0.01 with any limit applied. There were three exceptions: for Operator 1 and road class 3, the improvement was 0.05; for Operator 2 and road classes 1 and 3 the improvement was 0.04 and 0.03, respectively. The greatest positive effect to the TTQ NRMSE for Operator 2 in class 1 was achieved by removing trajectories with less than 30 events, but in class 3 for both operators by removing trajectories with more than 30 events.

3.4.3. TTQ performance on a weekday versus weekend

We repeated the experiment for another test period, a Sunday. The test setup was the same as in the main experiment, except the date. On Sundays people mostly

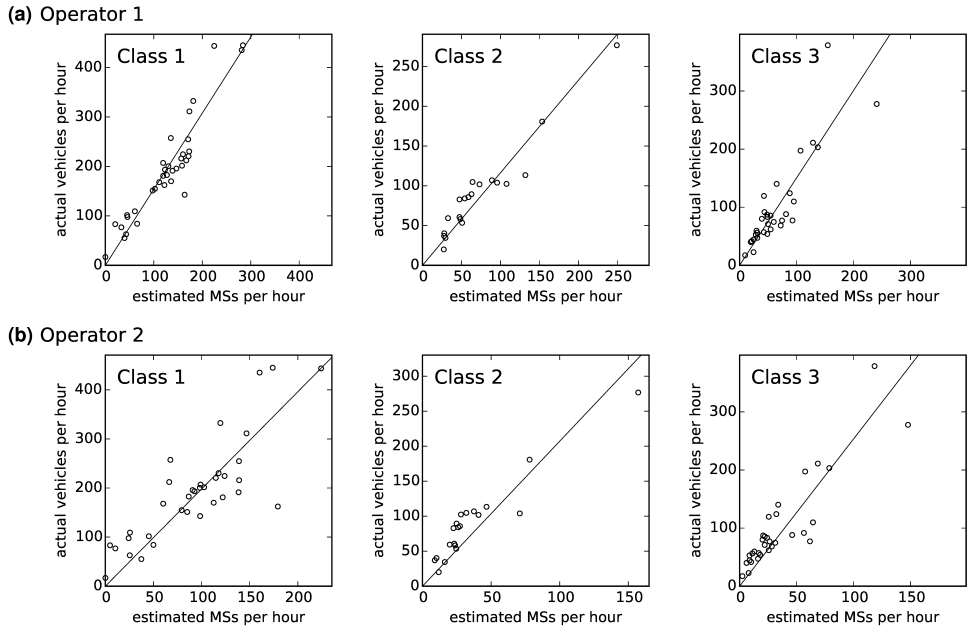
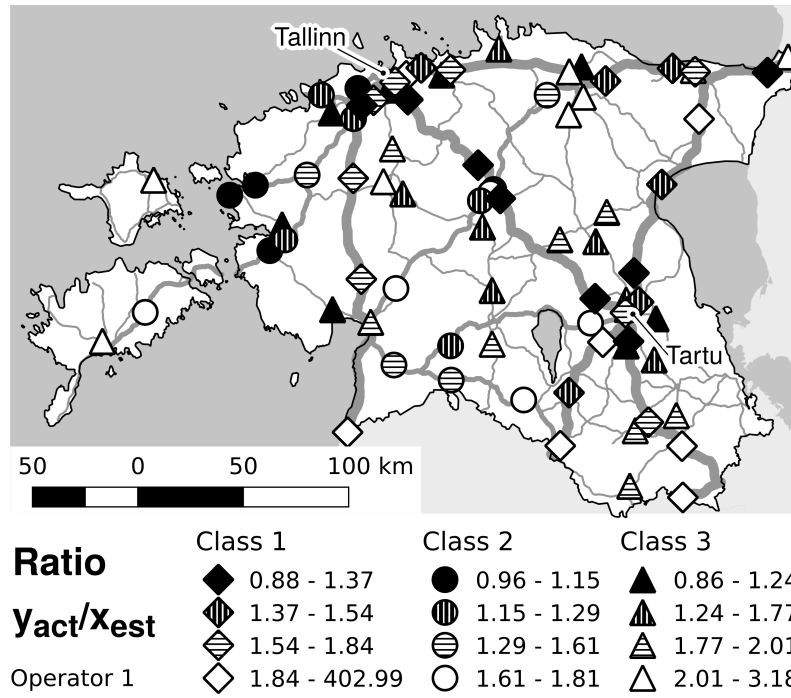
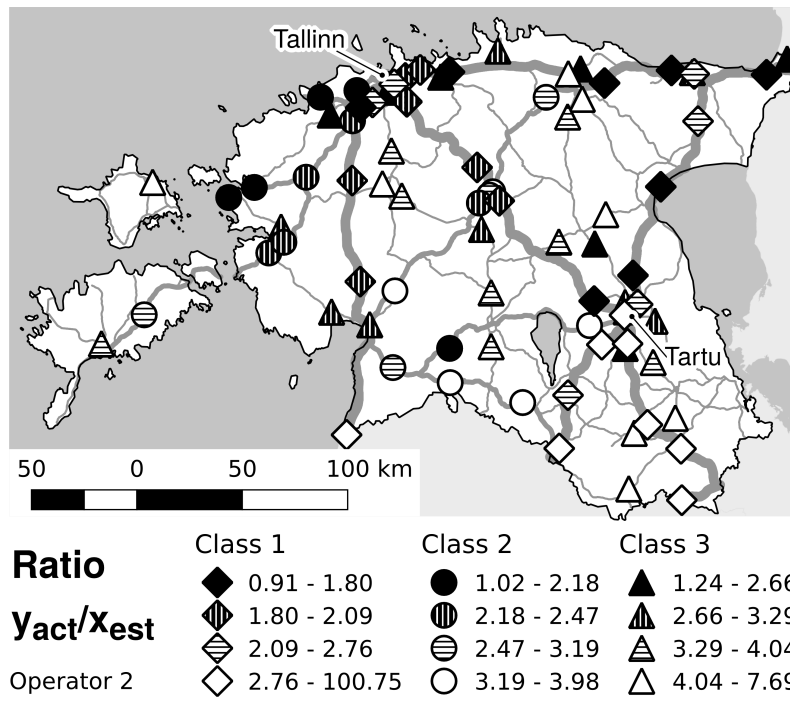


Figure 9. Scatterplot of $x_{estimate}$, the number of traversals by MSs estimated from events (horizontal axis) and y_{actual}, the actual traffic flow measured by sensors (vertical axis), by road class. Ping-pong suppression technique TTQ, $\alpha = 0.03$, $t_O = 60$ s, Operator 1 (a) and Operator 2 (b).



a



b

Figure 10. Geographical distribution of TTQ's prediction error: the ratio $y_{actual}/x_{estimate}$ of y_{actual}, the traffic flow, to $x_{estimate}$, the estimated number of traversals by MSs. (a) Operator 1; (b) Operator 2. The TTQ parameters α and t_O were optimized for each operator and road class separately.

do not go to work or school and the human mobility patterns are different than on weekdays [30]. The relative performance of TTQ technique remained the same – TTQ was the best among all tested techniques in terms of NRMSE, except for Operator 2 and road class 2 where it was still close to the best result.

3.4.4. Sensitivity to parameters

We tested the sensitivity of the TTQ technique to the parameters used. We changed each of the parameters α and t_O from the optimal value (given in Table 2) to a 2 times larger or smaller value and observed the changes in NRMSE. With all dataset and road class combinations the maximum change of NRMSE by such single parameter change was at most 0.025.

3.4.5. Discussion

We investigated the reasons why using cell-to-cell transition statistics improves NRMSE. Many cell pairs had a large distance between the cellplan-provided shapes, but in reality exhibited an almost instant handover, as illustrated by the lighter histogram on Figure 2. The most significant inconsistencies of cells were observed for cells located near the sea or a large lake. In these particular cellplans, radio wave propagation over water was underestimated.

In our experiment the TTQ algorithm was consistently the best performer in terms of NRMSE, showing low sensitivity to parameter variations and consistency of the results among datasets and road classes. This proves that cell-to-cell temporal relationships information can be beneficial in trajectory reconstruction. The TTQ technique does not require any additional input data and requires modest computational resources, making it easily applicable in practice.

Some factors were not taken into account in the current NRMSE experiments.

- The results are affected by the geographically non-uniform market penetration of mobile operators. This affect the average number of observed phones per vehicle, i.e., the scaling factor λ.
- The average number of passengers per vehicle might vary by road and region also, e.g., vehicle use might differ on rural vs urban roads. This affects the scaling factor λ.
- The traffic mode is not distinguished in our analysius. We assumed all traffic to be on roads, but in reality people also use train and ferry. Agricultural workers might "travel" on fields and in forest.
- Moreover, traffic estimates for sensors that are relatively close to country border suffer from edge effects.

In this chapter we employed the ping-pong suppression technique TTQ together with a simple map-matching method CPR. We plan to test the TTQ technique in combination with more advanced map-matching techniques and test it also in urban environment.

3.5. Conclusions

In this chapter, we described a new ping-pong filtering technique, called TTQ, that improves the accuracy of trajectory reconstruction by mitigating some cell shape inaccuracies and replacing some cellplan-based spatial relations with temporal relations derived statistically from mobile positioning data on cell identity level. The results provided a noticeable improvement over existing techniques on real data. The increased accuracy can make mobile positioning methods more useful in spatial planning.

The analysis has several limitations. It was based on one map-matching technique which might be more sensitive to particular distortion. So other mapmatching techique might benefit less from proposed TTQ technique. The only Key Performance Indicator (KPI) used was the accuracy of estimates for traffic intensity. Parallel use of multiple KPIs would give more complete picture. If one had true GPS accuracy trajectory then one could also analyse the trajectory itself, not only the resulting traffic estimate. The results are promising but further work is needed to assess the applicability of the presented approach for other situations.

4. BAYESIAN PROBABILISTIC MODEL OF CELLPLAN

4.1. Motivation and problem statement

Cellplan is used as a spatial PDF for assigning locations to events in positioning data. The quality of the spatial PDF used in estimation has a major effect on accuracy of the mobile positioning. The majority of networks (except A-GPS) have problems with the relatively low accuracy of mobile positioning [3], and any opportunity of improvement is worth a try.

Several approaches have been used to convert CGI into spatial PDF of the location of the object positioned. A CGI is usually spatially represented as a point, a Voronoi polygon, an area of sufficient signal strength based on a radio propagation model, or the best service area (the area where the cell has strongest radio signal among all cells active in the given location, based on radio propagation model). The probability is often assumed to be equal over the whole service area; or some kind of rasterization method is used where spatial PDF value is calculated for each cell and raster pixel, e. g. [12].

In most works, the PDF calculation does not consider overlap between neighbor cells. Simplistic approach would be just ignoring the effect of overlap. If the events happen in given cell, then probability is assumed uniform over all cell geometry. When cells overlap then overlapping areas have higher probability density than the areas covered by one cell only. Fig. 11 illustrates a heatmap of such situation on some synthetic data with sector cells and circular cells. Overlapping areas are clearly of higher intensity. There are no good reasons to believe that humans are with higher probability in cell overlap area. Overlap is just a technical property of RAN and it has no direct connection to human mobility. We have prior beliefs about spatial probability distributions – one might assume a uniform prior or assume that people are mostly near buildings and along the roads. In this chapter we will use Bayesian statistics to address these considerations.

There are two separate topics discussed in this chapter – what is realistic cell SPDF and how does cell overlap affect the positioning. Realistic SPDF is addressed by modifying the cell area by buffer operation and blur operation. Realistic cell SPDF is data preparation issue and cell overlap is relatively deep probabilistic model issue. If data quality is low, i.e., data is unrealistic, then the effect of probabilistic model is overshadowed with data quality problems and cannot be investigates. So the optimal adjusting of cell SPDF is necessary to investigate the effects of the Bayesian probabilistic model.

Research question: What is a suitable model for handling overlapping cells in positioning data? How much effect has cell overlapping on location uncertainty?

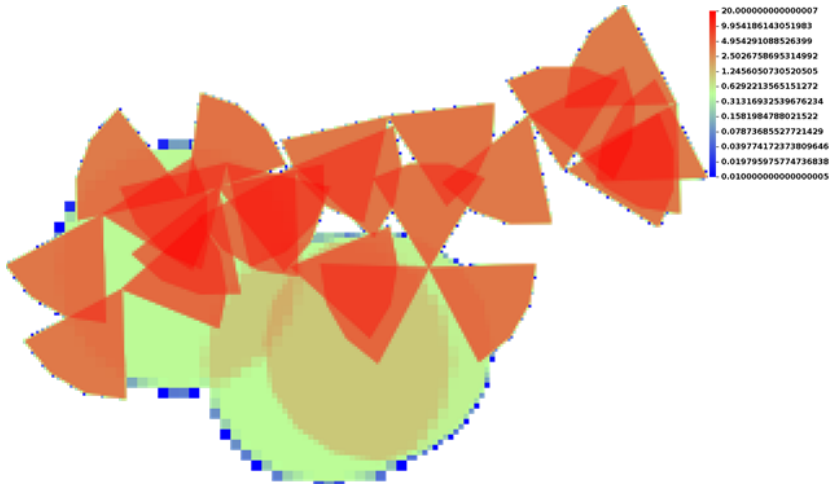


Figure 11. Sample heatmap for additive probability.

4.2. Bayesian approach to location estimation

The Bayesian approach is a natural choice when analyzing positioning data in case of multi-modal beliefs and evidence. For example, in paper [29] it is investigated how cell database correlation method is enhanced with Bayesian approach. The method is significantly different from the one discussed in this chapter but the application domain is same. The reported location accuracy improvements were 9..12%.

A thorough methodology applying Bayesian methods for defining the spatial PDF for mobile positioning was given in [63]. That paper provided a solution for the situation where neighbor antennas are present. The estimate was based on signal-to-interference-and-noise ratio (SINR) calculations. The paper provided a Bayesian probability estimate for situations where CGI is the only information known, and additionally, it is known that no other cells had good enough SINR. It was also shown how to combine PDF estimate with prior information about the distance to a cell tower, based on round trip time, and with prior knowledge of the distribution of population. As the estimate of the location of the MS during a mobile call, the maximum likelihood estimate (MLE) of the location, i.e. the location of the maximum of the PDF. The method was tested against the subset of emergency call data limited to situations where only one cell was within the reach of the MS. They found that the difference between the location measured by GPS and the MLE provided by that method was improved by 20%, compared to the baseline method. No direct PDF tests were performed in [63].

The Bayesian probability formulas in [63] do not consider the effect that the probability of connecting to a concrete cell will be reduced in locations where other cells are also reachable. If one event is generated in a certain situation, then one of the available cells is connected and the probability of selecting a certain cell is less than the total probability that an event is generated at all.

For example, a phone call start record will contain only one CGI, selected from the list of cells visible to the MS. This omission does not affect the specific case where it is prior knowledge that other cells are not reachable, handled in [63].

It is also assumed in [63] that the only mechanism how cells affect each other is SINR value change. This assumes that all cells work in the same frequency band. In reality, a RAN comprises several technologies and frequency bands.

For these reasons, the methodology of [63], while correct for the specific case investigated in that paper, was not adequate for our purposes, and the methodology described in the current paper was developed.

Six years after our publication [56] a paper was published by Tennekes and Gootzen [54] which took the same Bayesian approach as presented in the paper [56]. We had finished our work on given topic and there were no opportunities to combine our data with that work. The paper [54] discussed thoroughly the signal dominance model, i.e., how is the antenna selected in RAN to which the MS connects to. The paper includes illustrations of results on fictional example but no experimental quantitative improvement results were reported.

4.3. Methods

4.3.1. Mathematical model of spatial probability density function

Given a mobile operator's event logs for some time period, we want to map each event probabilistically to the geographical space where the mobile event occurred. We assign spatial PDF to each cell such that PDF defines the probability that the event occurred in any given location.

For more straigthtforward (and closer-to-computation) presentation, we consider here only discrete probability densities obtained by dividing the area of interest into pixels of appropriate size. The extension to the continuous case (the "ideal probability density" being approximated by discretizations) is quite obvious. We denote random variables by underlining them (following the van Dantzig convention).

The radio area network (RAN) consists of a finite set of cells $\mathscr{C}$. For each cell C and each pixel x, we want to compute $\mathrm{P}(x|C) = \mathrm{P}(\underline{x} = x \,|\, \underline{C} = C)$, the probability that the MS is at location x if it generates an event in cell C. In the following, we will describe a method how to do it. (The raster $\mathrm{P}(\underline{x} = \cdot \,|\, \underline{C} = C)$ is then what we call the spatial PDF of cell C.)

The probability $\mathrm{P}(x|C)$ is determined by the Bayes' formula:

$$\mathrm{P}(x|C) = \frac{\mathrm{P}(C|x)\,\mathrm{P}(x)}{\mathrm{P}(C)}, \tag{4.1}$$

where

- $\mathrm{P}(C|x) = \mathrm{P}(\underline{C} = C \,|\, \underline{x} = x)$ is the probability that if an MS is at location x, then it is connected to cell C (rather than any other cell or no cell);

- $\mathrm{P}(x) = \mathrm{P}(\underline{x} = x)$ is the Bayesian prior density, i.e., the probability for a person (or more precisely, a mobile station) to be in pixel x;
- $\mathrm{P}(C) = \mathrm{P}(\underline{C} = C)$ is the probability a mobile station (at a random location) to be connected to cell C, i.e., $\mathrm{P}(C) = \sum_x \mathrm{P}(C|x)\,\mathrm{P}(x)$ where x ranges over the plane.

The Bayesian prior $\mathrm{P}(x)$, representing our prior belief, can be constructed e. g. from population density data, road, and building layers (people are more likely to be on road or in a building).

The probability $\mathrm{P}(C|x)$ is computed as follows:

$$\mathrm{P}(C|x) = \mathrm{P}(\text{connected}|x) \cdot \mathrm{P}(C|x, \text{connected}) \tag{4.2}$$

where $\mathrm{P}(\text{connected}|x)$ is the probability that the MS is connected to the RAN at all (i.e., can generate an event) if it is at location x, and $\mathrm{P}(C|x, \text{connected})$ is the probability that if an MS is at location x and is connected, then it is connected to cell C.

Let $\underline{S}$ denote the active set of the MS, i.e. the set of cells detectable by the MS at a given time moment. A rough estimate of the conditional probability $\mathrm{P}(C \in \underline{S}|x)$ (that a certain cell C is detectable by the MS if the MS is at point x) is provided by the mobile operator in the form of cell polygon: $\mathrm{P}(C \in \underline{S}|x)$ is 1 if location x lies inside the cell's polygon and 0 if outside, so the active set depends deterministically on x. In a more refined model, $\mathrm{P}(C \in \underline{S}|x)$ could have non-zero values to reflect our ignorance of the precise coverage area and the stochastic nature of cell coverage caused by effects like Rayleigh fading and weather changes.

We can only receive CDR events when the mobile phone is connected to the network. Also, Estonia is very well covered and connection probability is almost 100%. We do not know actual connection rate. Therefore, we simplify and calculate for each raster pixel $P'(C|x) \approx P(C, connected|x)$.

For the probability $\mathrm{P}(C|x, \text{connected})$ we propose an adhoc formula (in contrast to the other formulas in this paper, which have some theoretical justification): namely, we take the probability to be proportional to the square of the probability $\mathrm{P}(C \in \underline{S}|x)$, i.e.,

$$\mathrm{P}(C|x, \text{connected}) = \frac{\mathrm{P}(C \in \underline{S}|x)^2}{\sum_{C \in \mathscr{C}} \mathrm{P}(C \in \underline{S}|x)^2}. \tag{4.3}$$

(This model does not take into account some properties of actual RANs, like the fact that cell handover may trigger the generation of some events. Handover events are naturally correlated with locations where cells overlap and have PDF different from all other events. In this paper the PDF of non-handover events is considered. Examples of such events are call start, call end, forced location update.)

Real RANs have sophisticated rules for selecting the active cell from all available cells. For example, higher frequency and higher technology are usually

selected whenever possible. However, such behavior has many unknown configurable parameters and rules. We considered it impossible to model the RAN behavior with more detail than the equation 4.3. This inability to fully describe cell selection behavior in RAN has been recognized by other publications also – for example, [54] devised "signal dominance model" based on signal quality and mentions "[T]he second phenomenon is switching between cells that is influenced by some decision making system in the network that tries to optimize the load balancing within the network [51]. The specifics of this system are considered unknown". The cell PDF P in 4.3 expresses not only our knowledge about the actual signal quality at given location but also our unability to know the actual signal quality for given MS, due to many factors affecting it – even most detailed cellplan can never fully reflect the real SINR values as experienced by the MS – the spatial resolution of cellplan cannot model the building, the position of cell relative to human body, building or vehicle, etc.

As an application, one can use the probabilities $\mathrm{P}(x|C)$ to compute a heatmap of a sample of mobile events, as follows. For each cell $C \in \mathscr{C}$, let its weight w_C be defined as the number of events measured at that cell. The value of each pixel x in the PDF, that we want to compute, is $\mathrm{E}(\underline{N}(x) \,|\, (w_C)_{C\in\mathscr{C}})$, the average number of events taking place in pixel x, the average being taken over all possible geographical locations of the events that would yield the measured event count w_C for each cell C. We make the simplifying assumption that all events are independent (most importantly, we don't take into account the correlation of events belonging to the same MS), so we can write

$$\mathrm{E}\,(\underline{N}(x) \,|\, (w_C)_{C\in\mathscr{C}}) = \sum_{C\in\mathscr{C}} w_C \mathrm{P}(x|C). \tag{4.4}$$

4.3.2. The implementation algorithm for computing the Bayesian PDF of cells and the heat map aggregate statistics using the cells

For practical use, the computations are separated into two steps: at first, the PDF is calculated for each cell, and the calculated PDFs can be used for various statistical calculations.

This particular implementation was optimized towards heat map calculations, where each cell is given a weight (e. g. the number of people connected to given cell) and raster image is generated, showing spatial density distribution.

For computations, we represented the probability field of each cell as a raster image in the RAM (random access memory). We used the Web Mercator projection. We had a fixed resolution called the full resolution, which is the resolution we were referring to in the formulas above (when talking about the heatmap pixel x).

In order to save space, some areas of the rasters were stored in computer memory in lower resolution. By *resolution*, we mean pixels per coordinate unit. More precisely, our variable-resolution raster was a grid of tiles, each tile a square matrix of floating-point numbers (pixels) with side length a power of two pixels.

The admissible resolutions for storing the tiles were the full resolution and the resolutions a power of two times lower than the full resolution.

The rendering was done at a resolution inversely proportional to the square root of the area of the polygon (rounded to the nearest higher admissible resolution or full resolution if no higher resolution was available). This resolution reduction is justified by the fact that the cell shapes have relatively large spatial uncertainty proportional to cell size — larger cells have larger uncertainty of the boundary of the cell's actual service area.

For each cell C, we obtained the raster $\mathrm{P}(C \in \underline{S}|x)$, the probability that cell C was detectable at a given location by rendering the cell's polygon from the cellplan (1 if the pixel is inside the polygon, 0 if outside, antialiasing on the edges). Optionally we applied a buffer and smoothing effect (blur) to the raster obtained from cellplan. These rasters were used in the next step for Bayesian PDF calculations.

Using the formulas above, we combined the rasters of the probability fields $\mathrm{P}(C \in \underline{S}|x)$ (uniform-resolution, but different cells possibly at different resolutions) to produce the variable-resolution rasters $\mathrm{P}(C|x)$ and finally $\mathrm{P}(x|C)$. During the computations, we did not reduce the resolution anywhere. The resulting rasters were stored in memory for use in statistical calculations.

We implemented heatmap calculation functionality based on Bayesian PDF. For heatmap calculation, one uses the PDF in appropriate resolution. If PDF is stored in lower resolution and higher resolution does not exist, then the available raster is upscaled accordingly.

4.3.3. The methodology to test PDF against real data

We compared how well different PDF estimates (obtained with and without the Bayesian method) conform to real measurements by calculating the likelihood of different PDFs, i.e., we compared the probability that the model produces the observed set of data. More appropriate model has a higher likelihood value. Suppose we have m mobile positioning events connected to cells C_i, $i = 1...m$, and we also know for these events the actual locations x_i, $i = 1...m$. For convenience, we calculated the logarithm of the likelihood $\mathscr{L}$,

$$\log \mathscr{L} = \log \mathrm{P}(\text{measurements}|\text{model}) = \log \prod_i \mathrm{P}(x_i|C_i) = \sum_i \log \mathrm{P}(x_i|C_i). \quad (4.5)$$

A higher likelihood value indicates that the PDF is a more precise estimate of the real (and not directly observable) probability distribution that generated the measured data.

In practice, the measured data may contain a few outliers that deviate significantly from typical behavior. I.e., the data can be a mixture of two distributions: typical behavior corresponding to our model and some unknown distribution. We assume that the percentage of the typical distribution in the mixture is at least $1 - q$, where q is a relatively small value. Therefore we apply an outlier elimina-

tion procedure to the data: when calculating the likelihood, the $\lfloor m \cdot q \rfloor$ data items with the lowest $P(x_i|C_i)$ will be excluded from the dataset.

4.3.4. Test dataset #1

A dataset from two mobile phones in the Czech Republic was used. The phones were used while traveling, so the events were spread nationwide. The phones enabled only data traffic, so only data events were used. Events were obtained from the RAN of a mobile operator. The mobile operator preferred to stay unnamed. The actual geographical location was recorded using a mobile phone's internal GPS. More than 25% of GPS data points were outside of corresponding cell area.

4.3.5. Test dataset #2

Data consists of GPS measurements of volunteers (coworkers and family members) who have installed GPS track recording software into their mobile phones, and CDR data for same persons from one mobile operator in Estonia. The mobile operator preferred to stay unnamed. For each CDR record, we found from GPS track the location of a person for given time moment. The records not covered by GPS track were ignored. We used 4 personal tracks from same 8 month time period. An example of data is given in Fig 12. Due to a small number of persons, the spatial coverage was very uneven – majority of events near home and work locations of the persons involved.

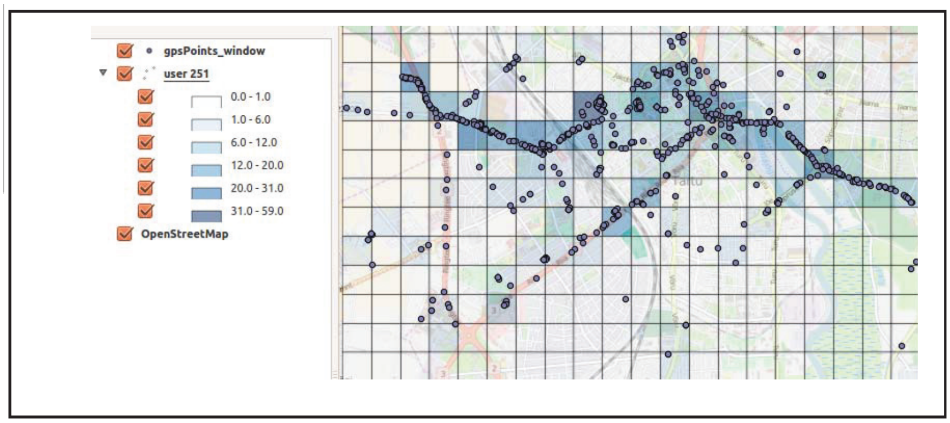


Figure 12. Measured data used for model accuracy estimation. On the left is color scale for a count of measurements in a grid cell. Each grid cell is 630m square.

4.4. Results

4.4.1. PDF likelihood test results on Dataset #1

We tested if modifying cell shapes using the Bayesian technique improves the likelihood. In the original cellplan, each cell was assigned a polygon and probability within the polygon was not specified, i.e., it was assumed to be uniform.

The original cellplan was clearly inaccurate in describing the PDF of events. More than 25% of the positioning events occurred when the MS was outside of the cell polygon. Therefore, before applying the Bayesian technique, we first transformed the original cellplan in two ways: by adding some buffer around the original polygon and by adding blur with Gaussian convolution. As we had the test data, we used calculated log likelihood as an objective measure which transformation gives a more accurate cellplan in terms of describing the spatial distribution of events.

We calculated the log likelihood of sample data for different cellplan transforms, varying the added buffer and added blur, and testing with Bayesian technique (i.e., computing $\mathrm{P}(x|C)$ according to subsection 4.3.1) and without (i.e., taking $\mathrm{P}(x|C)$ to be proportional to $\mathrm{P}(C \in \underline{S}|x)$ obtained directly from the cellplan; recall that we set $\mathrm{P}(C \in \underline{S}|x)$ to be 1 if the location x is within the cell's polygon and 0 if not).

A larger buffer and a larger radius of blur were applied to larger cells: the added buffer and applied blur radius were proportional to the square root of the cell area. We denote by bufRR the ratio of the added buffer ratio to the square root of the area of the cell polygon in the original cellplan and by blurRR the ratio of the radius of the Gaussian blur (i.e., the standard deviation of the one-dimensional Gaussian marginal distribution) to the square root of the area of the polygon after the buffer had already been applied. Blur was truncated at 3 blur radii.

The original cellplan had log likelihood $-\infty$ even at $q = 25\%$, i.e., more than 25% of the events were not inside the original cellplan's cell polygons. So the original unmodified cellplan was not usable for likelihood calculations, and we used only cellplans where some kind of transform had been applied.

For each outlier rejection level $q = 0\%, 1\%, 5\%$, we found the optimum transform such that the likelihood was maximum for a given q. Higher level q means that the distribution's tails were ignored, i.e., PDF without tails was favored. Lower q meant that the PDF was required to describe the tails of distribution too.

- The transform parameters for q=0% are bufRR=0.5, blurRR=2.
- The transform parameters for q=1% are bufRR=1, blurRR=0.5.
- The transform parameters for q=5% are bufRR=0, blurRR=0.5.

After finding the optimal cellplan for a given level q, we calculated the likelihood of our test dataset for different levels q.

The results are presented on Figure 13. If the results for lowest q values are missing on chart then some positioning entries for given q were in location where tested SPDF of cellplan was 0, hence infinitely low likelihood $\log \mathscr{L}$

For larger q values 25%, 10%, and 5%, the likelihood from Bayesian PDFs were practically equal to baseline for the given dataset. When q was 1% or 0%, then the Bayesian cellplan explained the data with significantly higher likelihood than the baseline cellplan.

If one is not interested in the tails of the distribution then the Bayesian cellplan

did not exhibit any advantages. This situation could occur if data is expected to be of low quality, and it is considered satisfactory that cellplan does not describe adequately the locations for a significant part of the events.

The dataset used in the test did not contain very distant outliers. It was easy to modify the cellplan by buffer and gaussian blur operations so that practically all events were described by it. In such case, the Bayesian cellplan likelihood was considerably better than the baseline method on the data we used.

4.4.2. PDF likelihood calculation performance

PDF calculation routine was executed in one thread in an openJDK 7 Java virtual machine on a 64-bit Linux laptop with 2.53GHz CPU i3-380M. Calculations of Bayesian PDF for one operator on the territory of Estonia took 7.3 seconds. Calculations of a 1920x1080 pixel heatmap over the whole Estonia took 84 ms. The cellplan of the given operator contained ca. 8000 cells.

4.4.3. Examples of heatmap visualization – synthetic data

Visual effects were explored with synthetic cellplan data. The data were generated with very simple script producing random geometries, with some parameter tweaking to get desired level of overlap. The actual geographic scale does not matter in this synthetic data, the algorithms work the same way on differently scaled data.

This synthetic heatmap exemplifies the visual clutter: some small areas, where cell polygons overlap, have artificially high intensity when using the original cellplan. Bayesian PDF levels out this effect considerably compared to the naive additive formula (i.e., using the probabilities $\mathrm{P}(C \in \underline{S}|x)$ instead of $\mathrm{P}(x|C)$ in formula 4.4). See Figure 14.

With blurred areas, the effect is still present but less prominent: see Figure 15.

4.4.4. Examples of heatmap visualization – real data

The real cellplan data was visualized with Bayesian PDF and the original cellplan PDF. The results are shown in Figure 16.

4.4.5. PDF likelihood test results on Dataset #2

a) Estimating SPDF from cellplans. We divided test area into quadratic pixels of size 630 meters. The pixel size was selected to be large enough so that many pixels contain multiple experimental GPS data location points. For all cellplan variants, the values of raster $\mathrm{P}'(C|x)$ were calculated. As illustration on Fig 17 is a cross-section of area, showing relative probability of each cell in given point.

b) Model likelihood calculations. Using the calculated SPDF rasters we calculated with formula 4.5 the likelihood of data given each SPDF variant tested. There were two cellplan datasets for the same network, one based on RSSI (Re-

ceived Signal Strength Indication) data and another one based on tower+azimuth (used to construct Voronoi polygons). Derived variations include:

- rssi_default : RSSI data, unchanged
- rssi_A1AP2 : twice expanded beam width
- rssi_A2AP1 : twice expanded beam along azimuth
- rssi_A3AP1 : three times expanded along azimuth
- rssi_convexhull : convex hull over original geometry
- rssi_ellipse : original approximated with ellipse
- voronoi_default : tower coordinate data, Voronoi constructed
- voronoi_A1AP2 : expanded twice width of beam
- voronoi_A2AP1 : expanded twice along the beam
- voronoi_A3AP1 : stretched three times along the beam
- voronoi_convexhull : original
- voronoi_ellipse : original shape replaced with ellipse

The results are shown in Fig 18.

4.5. Discussion

4.5.1. PDF estimation

The main findings are

- Accounting for cell overlap effect with Bayes rule had in the majority of cases positive effect.
- Some processing variants performed significantly better than the others for certain situations, but there was no single best method for all situations. When backed with large test dataset one could develop a mixed processing procedure using a different method for different areas but it is limited by overfitting concerns.
- SPDF calculated from RSSI data was not superior to simplistic Voronoi-based SPDF. We had expected that RSSI input enables better estimation than tower location and azimuth data.
- The dataset consisting of four personal tracks characterizes some location sufficiently but is not sufficient to represent an overall picture. Further investigation with larger dataset is needed.

In future work, we plan to analyze specific situations where the performance of one or other SPDF estimation variant degrades and optimize the methods accordingly. Also, we plan to investigate effects of the previous state, e.g. MS approaching cell, or stationary in the neighborhood of cell.

Improved PDF estimates should improve the results based on this, e. g., more accurate trajectory reconstruction. Still this is just a hypothesis and needs to be tested in future work.

In the current work, we also did not test the effects of applying prior knowledge during Bayesian PDF estimations. In the particular case where prior belief about population density is exactly equal to the number of cells visible in a given location (as defined by cellplan), the Bayesian PDF is the same as the PDF defined by the cellplan. In the examples in this chapter, a uniform distribution of population was assumed. An improved prior distribution is expected to give PDFs with higher likelihood.

Due to the small size and nonuniform coverage of the datasets, it was not possible to have separate independent data for determining the optimal cellplan transform parameters and for testing the likelihood level on the transformed cellplan. Separation of the datasets for model fitting and model testing is a prerequisite for convincing conclusions.

4.5.2. Visualization improvements

The results did show that in the heatmaps produced using Bayesian PDF, the contours of cells are less prominent, being closer to expectations. There is less visual clutter on drawing and explicit cell structure is deemphasized, without any loss in population density info. For example, when comparing the heatmaps (a) and (b) on Figure 16 then unlike additive heatmap (a) the Bayesian heatmap (b) does not over-emphasize the areas of cell overlap and cell contours are less pronounced.

More thoroughly modeled spatial uncertainty might be used explicitly in visualization, applying various visualization techniques, for example, glyphs and animation presented in [42] or the uncertainty glyphs and uncertainty ribbon technique presented in [48].

4.6. Conclusion

The Bayesian PDF model for cellplan was presented, and some test results were generated using this methodology. The limited test data indicated improvements in PDF compared to the PDF described directly in the cellplan.

The heat map visualization improved by Bayesian PDF as the resulting image was smoother and cell overlap areas were not over-amplified as with baseline additive model.

The results are promising and worth further study. Still, it requires more testing with real data to be sure that real RANs are described well by the Bayesian model described in this paper.

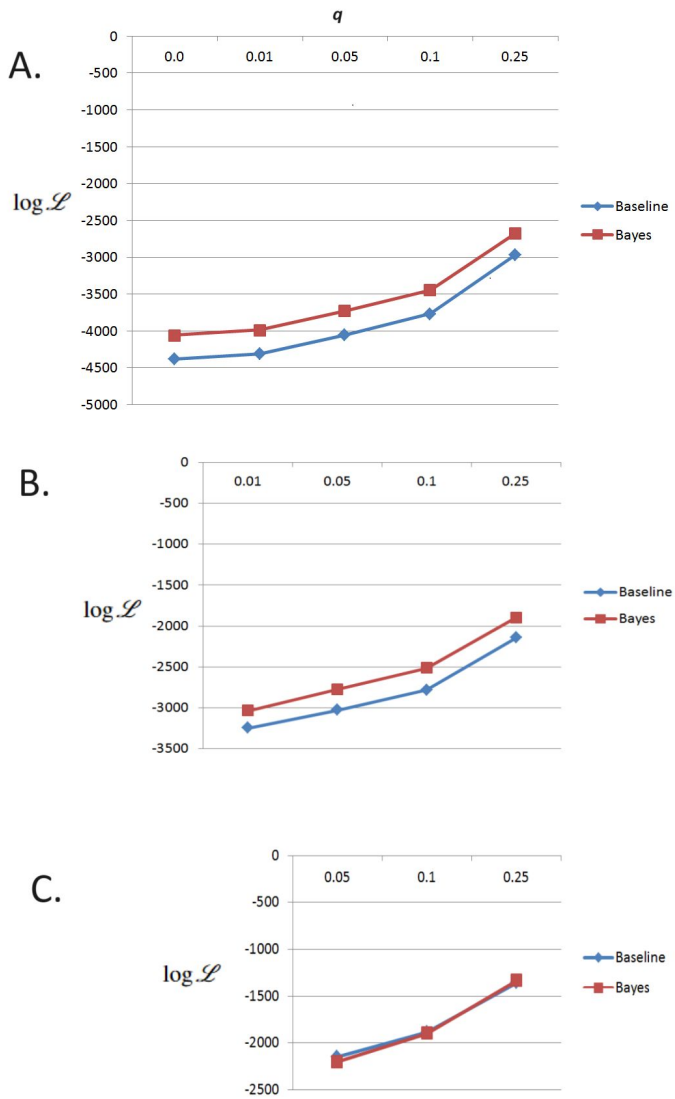


Figure 13. Log likelihood for cellplan optimized for (A) $q = 0\%$, (B) $q = 1\%$, (C) $q = 5\%$. Horizontal axis:the value of the outlier rejection level q. Vertical axis: log likelihood over the whole test dataset after eliminating outliers according to the outlier rejection level q.

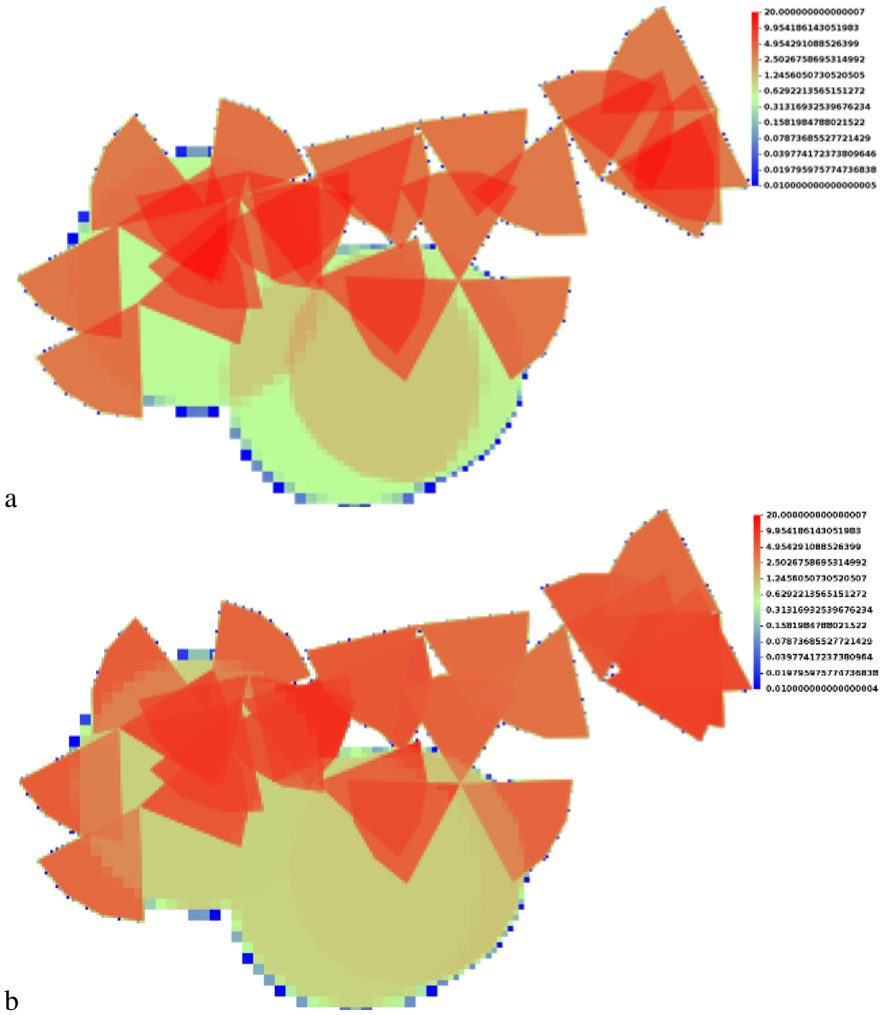
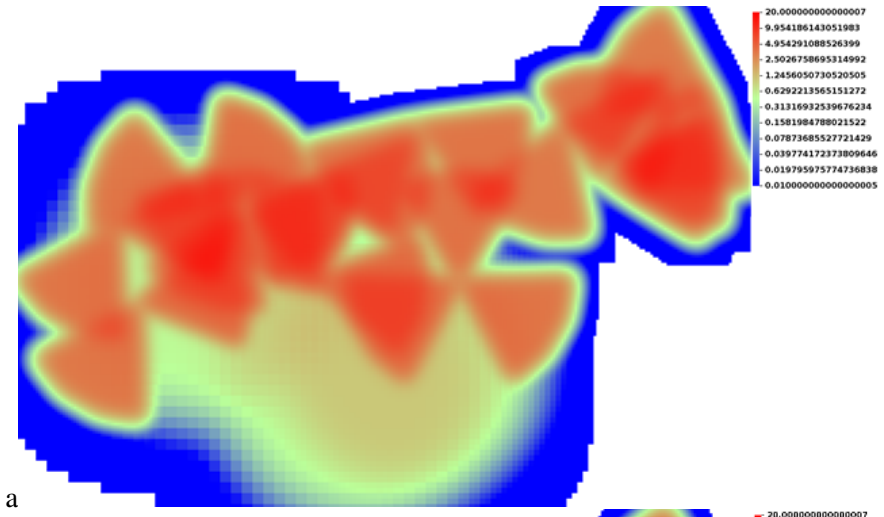
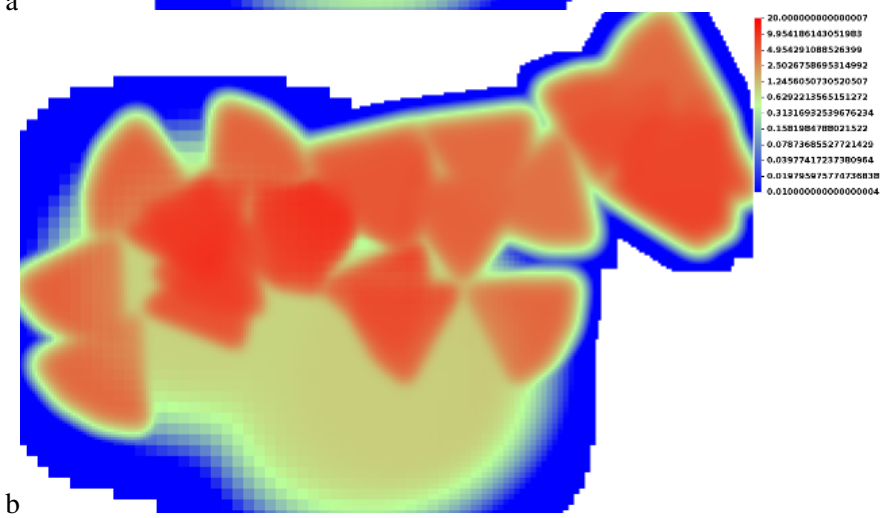


Figure 14. Sample heatmap for additive (a) and Bayesian (b) formulas, for polygonal cells, intensity is uniform within the polygon. (a) is same as Figure 11



a



b

Figure 15. Sample heatmap for additive (a) and Bayesian (b) formulas, for blurred cells. Blur is applied to each cell separately; Bayesian PDF is calculated after the blur.

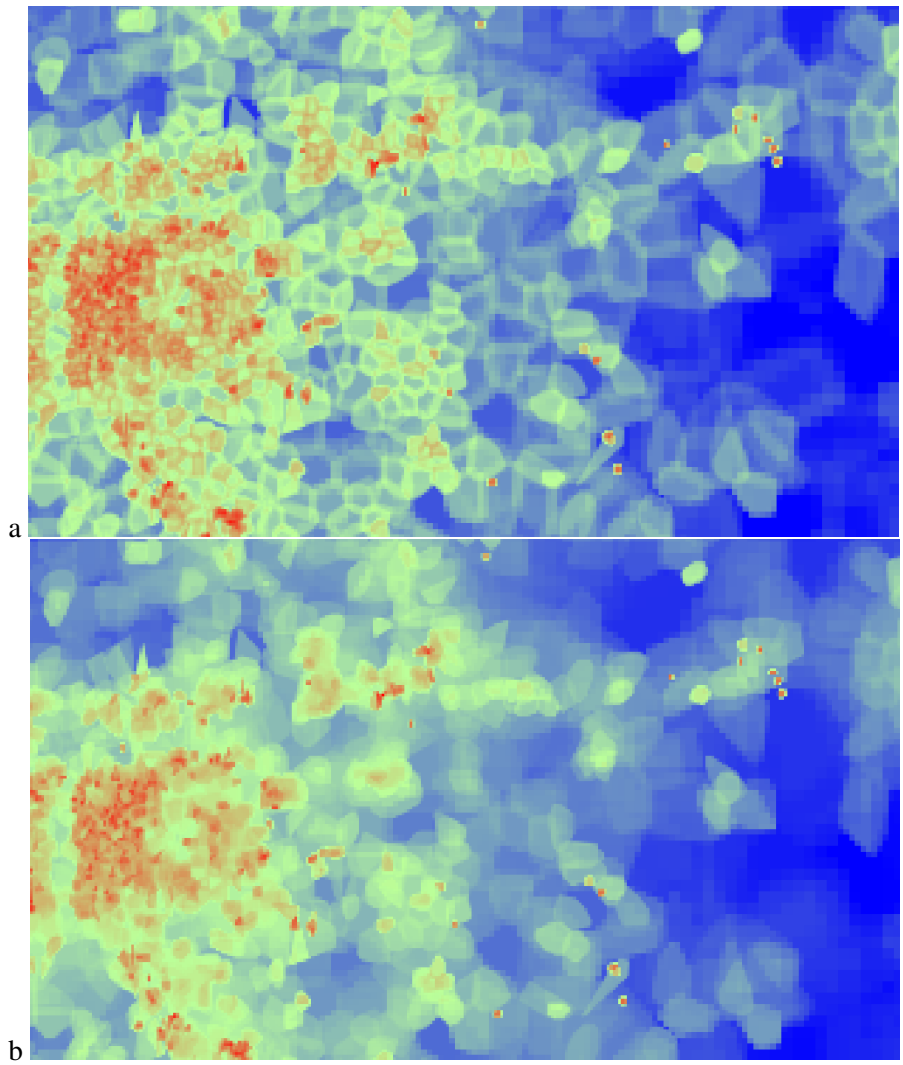


Figure 16. Sample heatmap for additive (a) and Bayesian (b) PDF formulas, for polygonal cells, intensity is uniform within the polygon. The size and location of the visualized area are not available for publication.

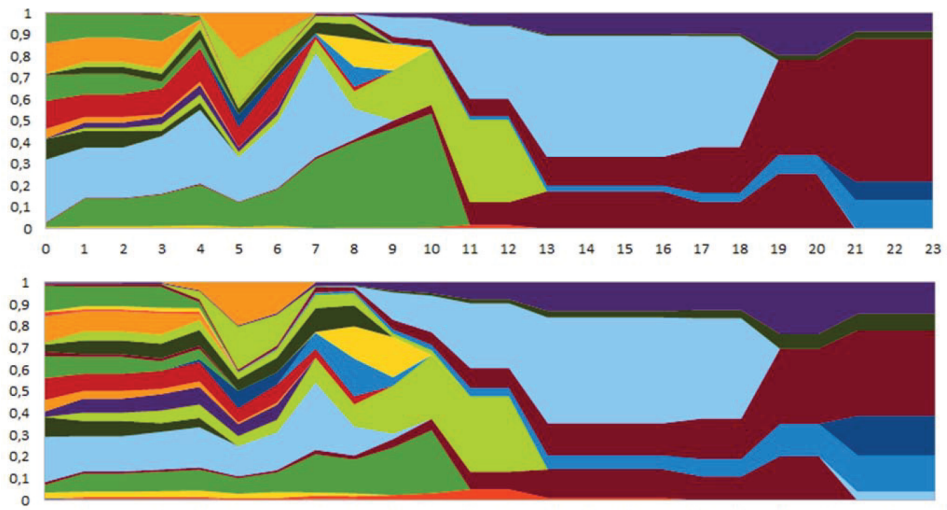


Figure 17. SPDF of all cells along a straight line along the cross-section of the test area, showing the probability of each cell in given location. Each colored polygon corresponds to one cell. Disconnected polygons of the same color are separate cells. The horizontal axis is pixel number (each pixel is 630m) and the vertical axis is stacked probabilities of cells. Upper chart is generated with applying Bayesian overlapping cell model, lower chart without considering overlapping effects.

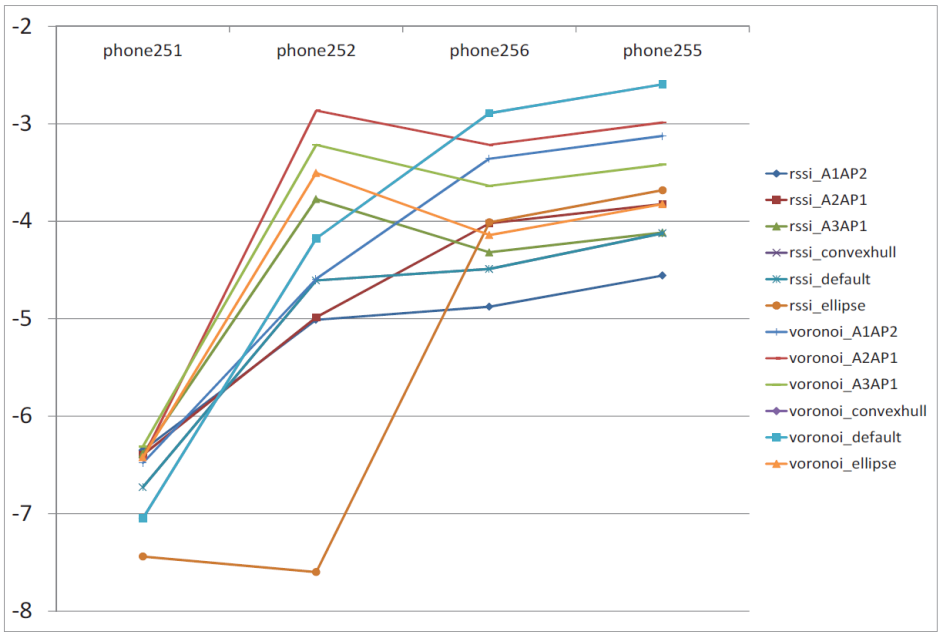


Figure 18. Relative performance of various cellplan variants. Horizontal axis – different test phone tracks (subsets of positioning data, with different spatial distribution). Vertical axis – average log P(C|x) for CDR records of given track. Results produced with same processing parameters are connected with line, for easier comparison of performance of the methods on different text phones.

5. MOBILITY EPISODE DISCOVERY IN THE MOBILE NETWORKS BASED ON ENHANCED SWITCHING KALMAN FILTER

5.1. Motivation and research question

One important problem in mobility research is splitting the trajectory into move and stop episodes. Stop locations and durations are extremely valuable indicators used to generate semantic annotations to a person's behavior, especially repeatedly visited anchor points [4] [53].

Research question: What is optimal method to detect the stop episodes and locations in trajectory data?

5.2. Related work

There are many publications in the literature about motion episode detection in the mobility trajectories extracted from different sensors. For example, in [31], the author dealt with a similar problem regarding detecting mobility episodes based on Global Positioning System (GPS) trajectory data. High uncertainties and specific noise patterns like "ping-pong handover" phenomena require developing methods specific to CGI data. Several techniques have been developed to reduce "ping-pong handover" distortions in the interpretation of CGI data [14, 20, 49].
When it comes to depicting mobility episode, there is interesting research focusing on handling noisy trajectories in detecting stops. For example, in [60], the authors focused on extracting the stops based on two key factors – the use of the sequence density based on time and space proximity and the Eps-reachability sequence. Eps-reachability is a spatio-temporal clustering criterion for stop detection proposed in this paper. The proposed method was tested against two baseline approaches using real data, and demonstrated optimistic outcomes. Moreover, Fiadino et al [20] presented a study on trajectory reconstruction, where they used a "ping-pong" suppression method that ignores events where the device connects back to the previous cell within a predefined time window. In [49], the authors describe a method for computing edit distances between event sequences where short-term handovers to another cell and repeated events can be ignored.

Concerning the CGI data Calabrese et al [14] used a method that was inspired by earlier work for GPS data in [62] and [31], which performs a clustering of measurement points and replaces original events with the barycenter of the cluster. Clustering has also been used, for example, in the form of sequence analysis providing outstanding computational throughput [57] or requiring more computations with density-based clustering [60].
Based on available information, the shape of each cell has to be defined to give location estimates for mobile positioning. Cell data provided by mobile operators

can be translated to cell shapes as Voronoi polygons by using the assumption that a phone connects to the nearest tower (e.g. [4]); as best server data polygons by using the assumption that a mobile phone connects to the cell with the strongest signal [15]; or as a raster model based on the assumption that the probability of connecting to a cell is a function of distance from the antenna tower [11].
Another approach that addressed detecting the mobility episodes was proposed in [8], where the authors presented a technique based on a discrete Switching Kalman filter for identifying the movement episode *Stay, Move, and Jump*. There *Move* episodes describe continuous movements and *Jump* episodes describe relocation without observations between endpoints.

However, the approach presented by the authors of [8] assumed that the state transition probabilities were represented as a function of the number of observed positioning events. RAN can emit positioning events with different intervals, and this caused the model to depend on RAN operation specifics. In reality the human behavior (*stay*, *move*, *jump*) is directly connected to time and only related to the internal operations of RAN as much as the latter correlates to time.
In this work, we propose a continuous-time switching Kalman filter for discovering the mobility episodes that is an extension of [8].

5.3. Methodology

5.3.1. The basic idea behind the improved model

When we have the trajectories extracted from the mobile positioning data, they are just a set of cell locations. For human mobility analysis, we need to know if the user of the mobile device was just passing through that coverage area or stopping to do something. From this perspective, we got inspired by previous work proposed in [8], and looked for ways to increase the accuracy in discovering the mobility episodes (*stay*, *move*, *jump*).

We used the assumption that human behavior is more closely related to time rather than count of mobile positioning records. E.g., statement "50% of stops are shorter than 1 hour" is directly relatable to human mobility, but statement "50% of stops have less than 7 positioning events" is describing RAN behavior, and human behavior is only indirectly related to that.

Therefore we had the hypothesis that replacing the discrete event model proposed in [8] with the CTSKF model with the corresponding handling of state switching probabilities, signal noise, and the correlated measurements will improve the resulting probability estimates of stop and move.

5.3.2. Kalman and Switching Kalman filtering

Kalman filter is a recursive algorithm for finding maximum likelihood estimations (MLE) for the hidden state of a linear dynamic system [28]. The estimation uses

the observations that are related to the hidden state. All the observations are assumed to have a Gaussian white noise with known covariance. There are two main equations, which define the dynamic linear system upon which the Kalman filter will be applied:

$$\mathbf{x}_t = \mathbf{F}_t \mathbf{x}_{t-1} + \mathbf{q}_t \tag{5.1}$$

$$\mathbf{y}_t = \mathbf{H}_t \mathbf{x}_t + \mathbf{r}_t \tag{5.2}$$

Equation (5.1) describes the change of the hidden state $\mathbf{x}$ from time step $t-1$ to time step t. The matrix $\mathbf{F}_t$ is the state transition matrix, which can differ for each time t. Vector $\mathbf{q}_t$ is a Gaussian noise with covariance matrix $\mathbf{Q}_t$. Equation (5.2) relates the hidden state with the observations. $\mathbf{H}_t$ is the observation matrix. The Observation noise $\mathbf{r}_t$ has covariance $\mathbf{R}_t$ and $\mathbf{y}_t$ are the observations made.

The actual value of $\mathbf{x}_t$ will remain unknown. The best prediction, given observations up to time t', is used: $\hat{\mathbf{x}}_{t|t'}$. This prediction also has an error estimate, that can be described by a covariance matrix: $\mathbf{P}_{t|t'}$. Following equations are applied for every time step t:

$$\hat{\mathbf{x}}_{t|t-1} = \mathbf{F}_t \hat{\mathbf{x}}_{t-1|t-1} \qquad \text{the noise is centered at zero} \tag{5.3}$$

$$\mathbf{P}_{t|t-1} = \mathbf{F}_t \mathbf{P}_{t-1|t-1} \mathbf{F}_t^T + \mathbf{Q}_t \tag{5.4}$$

$$\hat{\mathbf{e}}_t = \mathbf{y}_t - \mathbf{H}_t \hat{\mathbf{x}}_{t|t-1} \qquad \text{measurement residual} \tag{5.5}$$

$$\mathbf{S}_t = \mathbf{H}_t \mathbf{P}_{t|t-1} \mathbf{H}_t^T + \mathbf{R}_t \qquad \text{residual covariance} \tag{5.6}$$

$$\mathbf{K}_t = \mathbf{P}_{t|t-1} \mathbf{H}_t^T \mathbf{S}_t^{-1} \qquad \text{optimal Kalman gain} \tag{5.7}$$

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{\mathbf{t|t-1}} + \mathbf{K}_t \hat{\mathbf{e}}_t \qquad \text{updated estimate} \tag{5.8}$$

$$\mathbf{P}_{t|t} = (\mathbf{I} - \mathbf{K}_t \mathbf{H}_t) \mathbf{P}_{t|t-1} \qquad \text{updated estimated covariance} \tag{5.9}$$

Kalman filtering provides optimal estimation given only the past information. For offline uses, fixed interval smoothing can be used to obtain the MLE for the hidden state and the covariance of errors. This algorithm uses all the observations up to time $T > t$. The algorithm is named after its creators Rauch, Tung and Striebel (RTS):

$$\mathbf{C}_t = \mathbf{P}_{t|t} \mathbf{F}_{t+1}^T \mathbf{P}_{t+1|t}^{-1} \qquad \text{RTS gain} \tag{5.10}$$

$$\hat{\mathbf{x}}_{t|T} = \hat{\mathbf{x}}_{t|t} + \mathbf{C}_t (\hat{\mathbf{x}}_{t+1|T} - \hat{\mathbf{x}}_{t+1|t}) \qquad \text{smoothed state} \tag{5.11}$$

$$\mathbf{P}_{t|T} = \mathbf{P}_{t|t} + \mathbf{C}_t (\mathbf{P}_{t+1|T} - \mathbf{P}_{t+1|t}) \mathbf{C}_t^T \qquad \text{smoothed state covariance} \tag{5.12}$$

These formulas are applied iteratively, starting from time T, as a result of the MLE for all the hidden states, given that all the observations are found [46].

If the nature of the underlying process changes in time, then this can be modeled by changing the matrices $\mathbf{F}_t$, $\mathbf{Q}_t$, $\mathbf{H}_t$, and $\mathbf{R}_t$ correspondingly. In the context of this work, the most important change in the process is whether the observed mobile device is moving or staying still. Classical Kalman Filter requires that this change is known in advance. To overcome this limitation, an algorithm called Switching Kalman filter has been designed [38].

The switching Kalman filter requires a fixed set of Kalman filters to be chosen, such that a linear combination of these could describe each state transition. All of these Kalman filters are applied to every time step t, and the measurement residuals $\hat{\mathbf{e}}_t$ are used for finding the probabilities for each model that this transition occurred according to that model. In some cases, these probabilities can be even more important results of the algorithm than the estimation of $\hat{\mathbf{x}}_{t|t}$, that is the weighted average of the predictions of all the models.

In this work, three mobility types are distinguished and corresponding models associated with them: Stay, Jump, and Move. The Stay model describes a stationary device. Jump and Move models are both describing a non-stationary device: under the Jump model, the device changes location without leaving a trail of events during the actual movement, the Move model, on the other hand, describes the change in location that is accompanied by a sequence of events during the movement. In the Jump model, the change in location is modeled by having a very large location uncertainty for a short period. The Move model uses the speed of the device as the main cause for the location change.

In the switching Kalman model, a matrix of probabilities $\mathbf{Z}_t$ can be specified. An element of this matrix $\mathbf{Z}_{t\,i\,j}$ is the probability that if the transition to time step $t-1$ was made according to model i, then the transition to time step t will be done according to model j. We expect the consecutive transitions to be more likely made by the same model, except the Jump model: several consecutive jumps should be covered by the Move model.

When modeling the movements of mobile devices, the hidden state vector has four components: $\mathbf{x}_t = \begin{pmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{pmatrix}$. Observation vector has only two components: $\mathbf{y}_t = \begin{pmatrix} x \\ y \end{pmatrix}$ [8].

The observation matrix is time-independent and the same for all the models:

$$\mathbf{H}_t = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

For the Stay and Jump models, the state transition matrix $\mathbf{F}_t$ is a unit matrix. The state transition matrix of the Move model corresponds to simple linear movements:

$$\mathbf{F}_t = \begin{pmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{5.13}$$

In this equation, Δt is the time between time steps $t-1$ and t. For each model, the covariance matrix of the observation noise corresponds to the size of the coverage area of the antenna which the device was connected to.

The most significant difference between the models comes from their noise covariance matrices of state transition $\mathbf{Q}_t$:

- The Stay model should have the smallest speed and location variance. According to this model, the actual hidden state does not change at all.
- In the Move model, the uncertainty of location is also small, but speed is allowed to vary greatly to model more complex movements: turning, acceleration, etc.
- While the Jump model includes a change in location, it is not caused by the speed, but by very high uncertainty in the location. The variance in speed is set to be as low as in the Stay model.

Similarly to the Kalman filter, the Switching Kalman filter only considers the past observations when computing the maximum likelihood estimations. Fortunately, the SKF also has the possibility for a smoothing step. The set of equations for this is significantly more complex and is very well covered in the original article [38].

5.3.3. Proposed improvements to the baseline method

a) Model switching that is sampling rate independent . In the overview of the Kalman and Switching Kalman filters, at two places an implicit assumption was made that the time difference between two consecutive events is constant. In reality, it can change at least two orders of magnitude. The first of these was made when the model switching matrix $\mathbf{Z}_t$ was chosen to be fixed. In the baseline article, it was proposed to use the following values:

$$\mathbf{Z}_t = \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.45 & 0.45 & 0.1 \end{pmatrix}.$$

The ordering of the models is Stay, Move, Jump. This matrix defines a discrete Markov chain (MC) between these three states [39]. A toy example can be considered, to illustrate how the change in the frequency of the events changes the model. Suppose a person is standing still at time $t = 0$; its change in mobility type is described by a Markov chain with $\mathbf{Z}_t$ as its transition matrix. The baseline method assumes that the person's behavior should change depending on the number of events it does in the mobile network. The use of fixed probabilities

Z_t causes devices that make more events have a higher probability to switch its model without having actual extra evidence for it.

When using the given values for $\mathbf{Z}_t$, the asymptotic value of the probability for staying still is 45 %. This probability is independent of the sampling rate and should express the probability of staying still, when there have been no observations from this device in a long time. Mobile devices are most commonly carried by people with them, so it is reasonable to assume that the mobility patterns of mobile devices are rather similar to those of people. Analysis of human mobility in urban areas has shown, that on average every person takes 2.8 trips every day, with average length of 26 minutes [33, 36]. This is about 5 % of the day on the move - 11 times less than the 55 % proposed by the values in $\mathbf{Z}_t$.

Continuous-time Markov chain can be used for calculating the values of $\mathbf{Z}_t$ in such a way that the sampling rate does not affect the outcome. Unlike the discrete Markov chain, which is parametrized with transition probabilities, the continuous time MC is parametrized with transition rates [39]. The non-negative transition rates λ_{ij} express the number of transitions that take place from state i to state j in a unit of time on average. Transitions from state i to i are not modeled. These values are arranged in a generator matrix $\mathbf{A}$, such that for off-diagonal elements $\mathbf{A}_{ij} = \lambda_{ij}$ and the diagonal elements are chosen such that all rows add up to 0. By solving either the Kolmogorov forward or backward equations, an expression for $\mathbf{Z}_t$ is obtained [39]

$$\mathbf{Z}_t(\Delta t) = e^{\mathbf{A}\Delta t},$$

Δt is the time between time step t and $t-1$. Matrix exponentiations are performed using the eigenvalue decomposition $\mathbf{A} = \mathbf{B}\Lambda\mathbf{B}^{-1}$ and Taylor expansion. Resulting in the following equation:

$$\mathbf{Z}_t(\Delta t) = \mathbf{B}e^{\Lambda\Delta t}\mathbf{B}^{-1}.$$

Λ is a diagonal matrix, and exponentiation of a diagonal matrix is another diagonal matrix whose elements are elementwise exponents of the initial matrix. The computational cost of this is negligible compared to the rest of the algorithm.

The need for separate move and jump models does not come from human mobility patterns, but from the behavior of the mobile network. Because of this, human mobility statistics can not be used as prior for these models. The mobile network data shows that about 10 % of trips are happening according to the jump model, and others are taking place according to the move model. The estimate for generating matrix, while considering also the average length of a trip and the average number of trips per day, is following:

$$\mathbf{A} = \begin{pmatrix} -0.125 & 0.113 & 0.012 \\ 2.31 & -2.31 & 0 \\ 2.31 & 0 & -2.31 \end{pmatrix} \frac{\text{transitions}}{\text{h}}. \tag{5.14}$$

The ordering of the models is Stay, Move, Jump.

b) Sampling rate independent process noise. In the original article, the values for process noise covariance $\mathbf{Q}_t$ were fixed [8]. This simplification assumes that the time difference Δt between time steps is constant. As seen in the previous subsection, this assumption does not hold. To get a better understanding of the effect this assumption has, a simple example can be considered: a person with a known initial location and speed ($x(0) = 0\,\mathrm{m}$, $\dot{x}(0) = 1\,\mathrm{m\,s^{-1}}$) is observed over a period of 1 h. Its movement is modeled as linear ($\mathbf{F} = \begin{pmatrix} 1 & \Delta t \\ 0 & 1 \end{pmatrix}$) with fixed process noise $\mathbf{Q} = \begin{pmatrix} 0.01 & 0 \\ 0 & 0.01 \end{pmatrix}$. The predicted location is precisely the same for any sampling rate. The confidence in this prediction decreases as the sampling rate increases. This observation implies that while the state transition matrix $\mathbf{F}_t$ is correct, the process noise covariance $\mathbf{Q}_t$ should depend on Δt somehow.

The following derivation for $\mathbf{Q}_t(\Delta t)$ is based on the article of P. Axelsson and F. Gustafsson [6]. The state transition equation (5.1) is a discretized version of the stochastic differential equation

$$\mathbf{dx}(t) = \mathbf{A}\mathbf{x}(t)dt + \mathbf{d}\beta(t), \tag{5.15}$$

where $\beta(t)$ is a Brownian motion with

$$E[\mathbf{d}\beta(t)\mathbf{d}\beta(t)^T] = \mathbf{S}dt. \tag{5.16}$$

Discretization is done by integrating the equation (5.15) over the time interval $[t_{t-1}, t_t]$. The integration results in

$$\underbrace{\mathbf{x}(t_t)}_{\mathbf{x}_t} = \underbrace{e^{\mathbf{A}\Delta t}}_{F_t}\underbrace{\mathbf{x}(t_{t-1})}_{\mathbf{x}_{t-1}} + \underbrace{\int_{t_{t-1}}^{t_t} e^{\mathbf{A}(\tau - t_t)}\mathbf{d}\beta(\tau)}_{q_t}. \tag{5.17}$$

Using the definition of covariance of white noise $E[q_{t_1}q_{t_2}] = \mathbf{Q}_{t_1}\delta_{t_1 t_2}$, the covariance matrix $\mathbf{Q}_t$ can be expressed as

$$\mathbf{Q}_t = \int_0^{\Delta t} e^{\mathbf{A}\tau}\mathbf{S}e^{\mathbf{A}^T\tau}d\tau. \tag{5.18}$$

Solving this sort of integral for a general case is a rather complicated problem, but there are some approaches to it, for example, using Lyapunov equations. When all of the eigenvalues for matrix $\mathbf{A}$ are zeros, then this integral will have an analytical solution. This condition holds for all models: Stay, Move, and Jump. For the Move model the matrix $\mathbf{A}$ is

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

for Stay and Jump models, it is zero matrix. The solution to integral (5.18) under the assumption that all the eigenvalues of $\mathbf{A}$ are zeros is

$$\mathbf{Q}_t = \sum_{i=0}^{p-1}\sum_{j=0}^{p-1} \frac{\Delta t^{1+i+j}}{i!j!(1+i+j)}\mathbf{A}^i\mathbf{S}\mathbf{A}^{jT},$$

where p is the dimensionality of the square matrix $\mathbf{A}$. Because of the sparsity of $\mathbf{A}$ in these models, most of the terms are 0, resulting in a simplified expression for $\mathbf{Q}_t$, for Stay and Jump models it will be

$$\mathbf{Q}_t = \mathbf{S}\Delta t \tag{5.19}$$

and for the Move model it will be

$$\mathbf{Q}_t = \mathbf{S}\Delta t + (\mathbf{AS} + \mathbf{SA}^T)\frac{\Delta t^2}{2} + \mathbf{ASA}^T\frac{\Delta t^3}{3}. \tag{5.20}$$

There is no reason to believe that the noise for different components of $\mathbf{x}$ would be directly correlated, so matrix $\mathbf{S}$ can be a diagonal matrix. The values on diagonal can satisfy the same conditions that were proposed for a fixed $\mathbf{Q}$. Selection of these values by parameter optimization is discussed in the results section.

c) Correlation and overconfidence. One of the preconditions for applying the Kalman Filter is that the process noise $\mathbf{q}_t$ and observation noise $\mathbf{r}_t$ have to be uncorrelated in time [28]. There is no reason to believe that this assumption does not hold for the process noise, but there is one case where it does not hold for the observation noise. When the device is stationary, it is rather common that it will keep connecting to the same cell over a long period. If we assume these positionings as uncorrelated measurements, then the predicted hidden state will converge to the observation, as depicted in Figure 19. This results in unrealistically overconfident predictions. Completely ignoring these consecutive events for the Stay model would also be unwary because even when fully correlated, they might still be carrying information with them.

Some work has been done to design a Kalman filter generalization for correlated noise. Here we will consider the results of Petovello et al [44]. In their work, they generalized the observation equation (5.2) to include correlated error:

$$\mathbf{y}_t = \mathbf{H}_t\mathbf{x}_t + \mathbf{u}_t. \tag{5.21}$$

In this new relation $\mathbf{u}_t$ is a time-correlated error, with correlation expressed through equation

$$\mathbf{u}_t = \Psi_t\mathbf{u_{t-1}} + \mathbf{r}_t. \tag{5.22}$$

As in the original Kalman filter, the error $\mathbf{r}_t$ is white Gaussian noise with covariance $\mathbf{R}_t$. A concise overview of filtering a dynamic system, which follows this relation, has been given by Wang et al, along with a comparison to alternative approaches [58].

As stated before, the correlation does not affect the Jump or Move models. With the Stay model there are two possible cases:

- If the observation differs from the previous, then there is no correlation, and the matrix Ψ_t is zero. In this case, the regular Kalman filter can be applied.

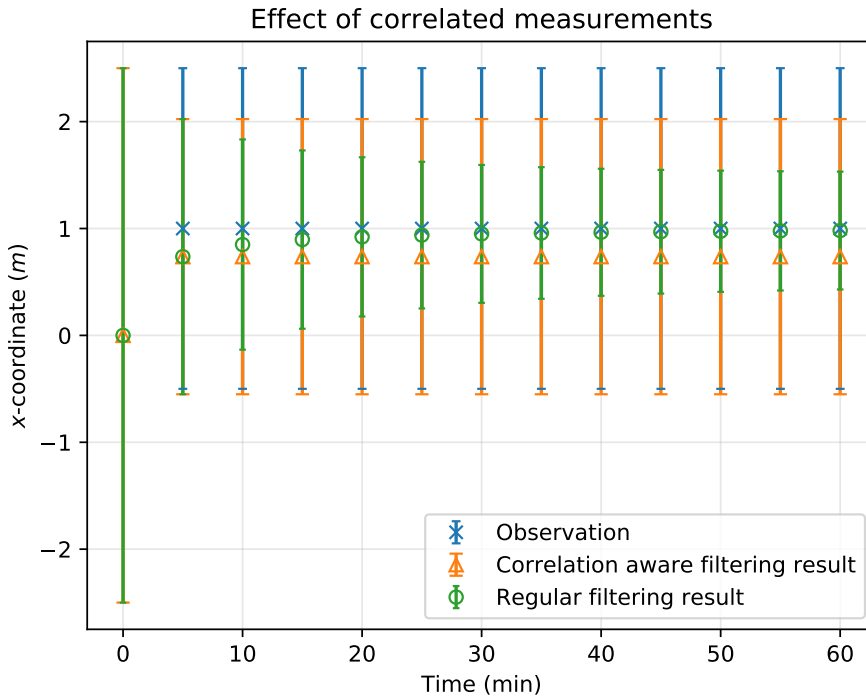


Figure 19. The behavior of Kalman filters predicted location and covariance with correlated measurements.

- If observation $\mathbf{y}_t$ is the same as the previous one $\mathbf{y}_{t-1}$, then under this model, the correlation transition matrix is a unit matrix, and there is no added white noise, meaning that $\mathbf{R}_t$ is zero.

When substituting these assumptions along with the state transition matrix for stay model $\mathbf{F}_t = \mathbf{I}$ into the new update formulas presented in [58], then they will simplify significantly:

$$\hat{\mathbf{x}}_{t|t} = \hat{\mathbf{x}}_{t-1|t-1}, \tag{5.23}$$

$$\mathbf{P}_{t|t} = \mathbf{P}_{t-1|t-1}. \tag{5.24}$$

According to this result, correlated measurements have no effect on the estimated hidden state $\hat{\mathbf{x}}$, but they will compensate for the effects of process noise on the covariance of the prediction $\mathbf{P}$. This can be considered as if the device, that stays connected to the same cell, is frozen in time according to the observer, until it will connect to a new cell. An example of how this sort of model behaves under correlated measurements is depicted in Figure 19. As expected, the correlation-aware model does not converge to an overconfident prediction but stays at a realistic value.

When using a Switching Kalman filter with the regular Move and Jump models along with correlation aware Stay model, then there will still be some convergence

because the final result is a weighted average of these three models. This observation is not a problem to be solved but a sign of the predictive power of SKF. When there is strong evidence that the device is stationary, the weight of the Stay model will be very close to 1 and the convergence is negligible.

The RTS smoothing algorithm also requires changes to work with correlated measurements. Like in the filtering case, this is only applied to the stay model. If observation $\mathbf{y}_t$ is not the same as the next one $\mathbf{y}_{t+1}$, then regular RTS smoothing equations can be used. Otherwise, correlation aware update equations must be used:

$$\hat{\mathbf{x}}_{t|T} = \hat{\mathbf{x}}_{t+1|T}, \tag{5.25}$$

$$\mathbf{P}_{t|T} = \mathbf{P}_{t+1|T}. \tag{5.26}$$

5.4. Results

5.4.1. Sample data

Sample data were used to assess the effectiveness of the developed tools. This data contained both the mobile network events and also accurate location data from GPS measurements. For this work, a single person was tracked over a period of one month. During that time, the subject traveled in an area with a radius of 35 km. There are, in total 40 000 mobile events made by its device, with an average gap of 67 s between them.

The algorithm was implemented in Java programming language with JAMA linear algebra package [34]. Average single-threaded processing throughput was 6000 positionings/second on 2.4GHz i5-6300U CPU.

On figure 20 the subject's event difference distribution is depicted. This subject is more avid mobile user, than the average - for the entire population 95 % of events happen with less than 5 min after previous, but for our subject it is 3.5 min.

This person moves mainly along the shore of a body of water. This pattern makes its trajectory rather anisotropic: 90 % of its location variance is explained by its first principal component. Because of this, in the following visualizations, only its movement in the direction of the first principal component can be shown without any significant loss in information. Absolute coordinates are avoided and all the coordinates are deviations from the trajectories' centroid to preserve the anonymity of the subject and the mobile operator.

5.4.2. Quality measure for model results

The main purpose of using the Switching Kalman filter is detecting whether the subject is moving. Improved location accuracy is a secondary result of this algorithm. For this reason, the quality of the model is assessed by its ability to correctly locate the times when the subject was stationary. The probabilities ($\hat{p}_{\text{stop}_t}$)

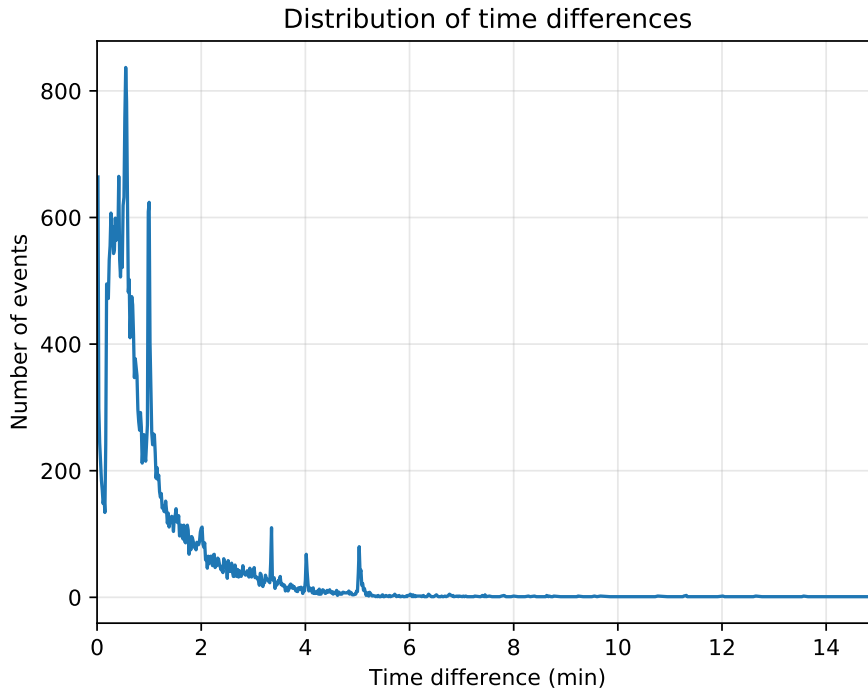


Figure 20. Distribution of time differences between consecutive network events in test data.

calculated from the SKF do not describe the observation times t, but the transitions from previous observation $t-1$ to this. For each of these transitions from the GPS track, it is found whether the device was moving during that time. The probability from GPS (p_{stop_t}) is either 0 or 1. These probabilities are combined into a time-averaged log loss:

$$-\frac{\sum_t (p_{\text{stop}_t} \log(\hat{p}_{\text{stop}_t}) + (1 - p_{\text{stop}_t}) \log(1 - \hat{p}_{\text{stop}_t}))\Delta t}{\sum_t \Delta t}. \tag{5.27}$$

5.4.3. Parameter optimization

The Switching Kalman filter has two sets of parameters: the values of the generator matrix, which determine the values of the model switching matrix $\mathbf{Z}(\Delta t)$, and the value of the process noise covariance matrices $\mathbf{S}$. The values of generator matrix $\mathbf{A}$ are computed using the results of some statistical research papers and are shown in equation (5.14). Each movement type model (Stay, Jump, Move) has its own unique process noise covariance matrix. To avoid having to fit too many parameters, the same assumptions are made that were proposed by Batrashev et al [8]:

- The model is isotropic - the uncertainty in the easting and northing will grow in time at the same rate.

- The covariance matrix will have only values at its main diagonal set.
- The Stay and Jump models will have the same noise for speed.
- The Stay and Move models will have the same noise for location.

These assumptions reduce the number of parameters to four: two location noises and two speed noises.

Half of the trajectory was used to find optimal values for these variables. After performing parameter optimization the following values resulted in the minimal log-loss:

$$S_{\text{stay}} = \begin{pmatrix} 96\,\text{m}^2\,\text{s}^{-1} & 0 & 0 & 0 \\ 0 & 96\,\text{m}^2\,\text{s}^{-1} & 0 & 0 \\ 0 & 0 & 6.8\,\text{m}^2\,\text{s}^{-3} & 0 \\ 0 & 0 & 0 & 6.8\,\text{m}^2\,\text{s}^{-3} \end{pmatrix} \tag{5.28}$$

$$S_{\text{jump}} = \begin{pmatrix} 1500\,\text{m}^2\,\text{s}^{-1} & 0 & 0 & 0 \\ 0 & 1500\,\text{m}^2\,\text{s}^{-1} & 0 & 0 \\ 0 & 0 & 6.8\,\text{m}^2\,\text{s}^{-3} & 0 \\ 0 & 0 & 0 & 6.8\,\text{m}^2\,\text{s}^{-3} \end{pmatrix} \tag{5.29}$$

$$S_{\text{move}} = \begin{pmatrix} 96\,\text{m}^2\,\text{s}^{-1} & 0 & 0 & 0 \\ 0 & 96\,\text{m}^2\,\text{s}^{-1} & 0 & 0 \\ 0 & 0 & 22\,\text{m}^2\,\text{s}^{-3} & 0 \\ 0 & 0 & 0 & 22\,\text{m}^2\,\text{s}^{-3} \end{pmatrix}. \tag{5.30}$$

5.4.4. Effect of the improvements to the algorithm

Three independent improvements were proposed to the original algorithm. To get a quantitative understanding on the effects of these, the loss function was evaluated on the second half of the trajectory with all different combinations of these improvements applied to it. These results are summarized in Table 4.

Table 4. Log-loss values for different variations of the Switching Kalman filter.

Switching matrix	Process noise	Correlation	Log-loss
Original	Original	Original	0.2089
Original	Original	Improved	0.1889
Original	Improved	Original	0.2011
Original	Improved	Improved	0.1787
Improved	Original	Original	0.2056
Improved	Original	Improved	0.1855
Improved	Improved	Original	0.2006
Improved	Improved	Improved	0.1749

It is reassuring to see that the worst-performing model is the one with none of the improvements applied and the one with all of them is the best-performing model. It is also important to note that adding an additional improvement to any of the models improves the performance.

Three improvements were considered in this paper. The highest performance gain comes from correlation aware stay model. A common example of a situation, where this improvement helps is depicted in Figure 21. Before and after the depicted time there was a long period of correlated measurements making

the original algorithm's prediction have unreasonably low uncertainty in the device's location. This reduced the impact of new observations significantly and the predicted location just drifted from one location to another without opting for the move model. The improved variant does not produce unrealistically accurate location predictions.

Making the process noise Q depend on the time difference Δt was the second-best improvement. For the stay and jump models the process noise scales linearly in time (equation (19) and for move model the scaling is with a cubic equation (20). When only this improvement is applied, then the major differences between models should occur at time points that have unusually large gaps between them. One of such cases is demonstrated on Figure 22.

The classical algorithm [12] models this movement as two distinct jumps; however, our improved model puts more emphasis on the measurements and correctly identifies them as one continuous movement. These are the results when the improvements are used one at a time. By using all of them together the score improves even further, but the observations made on the partial result also hold for the complete model. It is important to note, that these changes to the model can in some parts of the data make the prediction slightly worse, but unlike the improvements, these do not seem to be systematic effects, but rather random fluctuations.

The least significant improvements came from making the model switching matrix Zt depend on the time difference Δt. Similarly to the process noise, it has the most significant effect in the regions with unusually large Δt - an example of this is depicted in Figure 23. Unlike two previous examples, there is almost no change in the predicted location nor its covariance - only a slight change in the model probabilities.

5.5. Conclusion

In this chapter, we presented an enhanced method of the switching Kalman filter by introducing a continuous-time aspect into the model in order to discover mobility episodes (*Move, Stay, Jump*) from the CDR data. Our evaluation and testing has been conducted on two different datasets with different characteristics and in both cases the results were very encouraging. In general, our model using continuous-time switching Kalman filter performed better than the discrete switching Kalman filter.

The results of this approach can be useful for future application and research in refining the trajectories extracted from the CDR data and their usage in solving issues related to mobility, transportation, and urban planning.

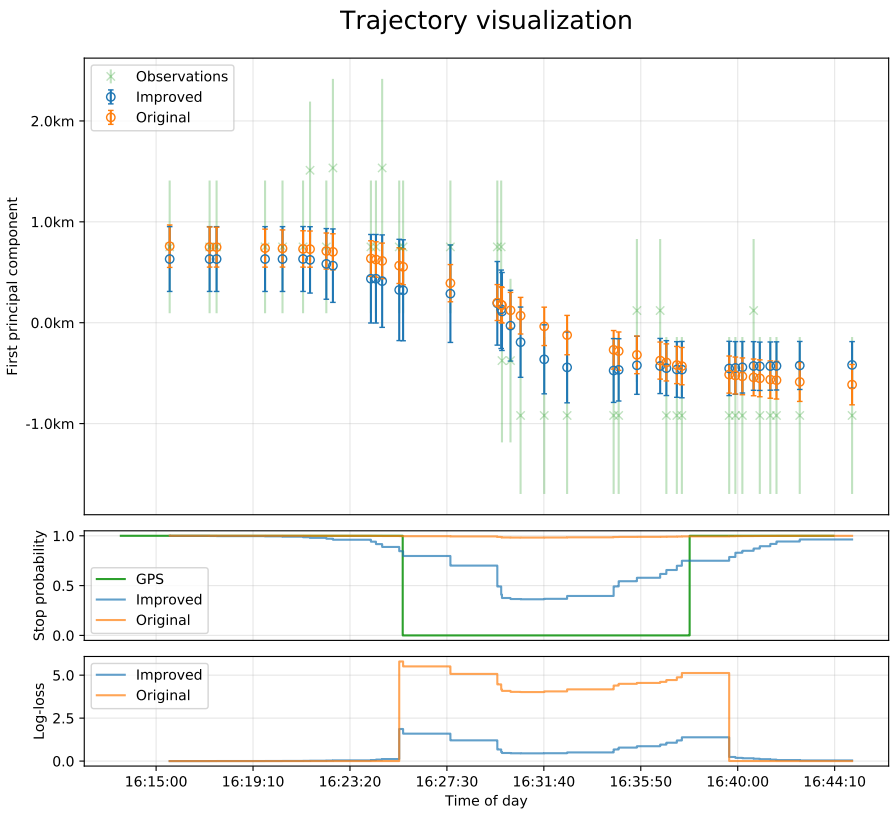


Figure 21. Comparison of the outputs of the original switching Kalman filter and a improvement with correlation aware Stay model.

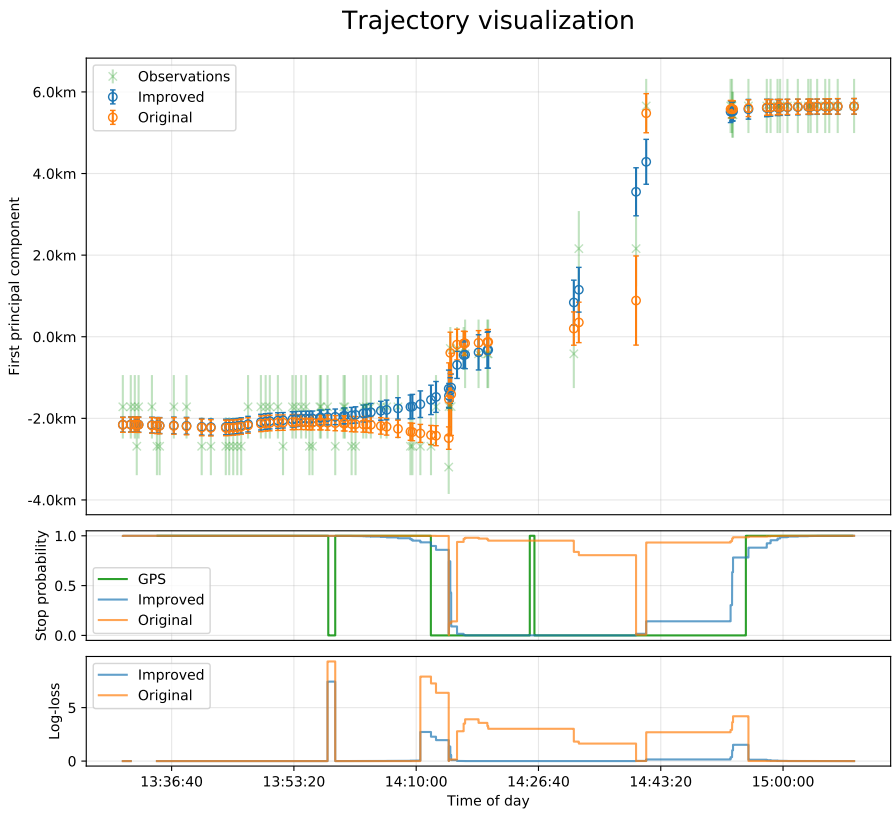


Figure 22. Comparison of the outputs of the original switching Kalman filter and improvement with sampling-rate-dependent process noise.

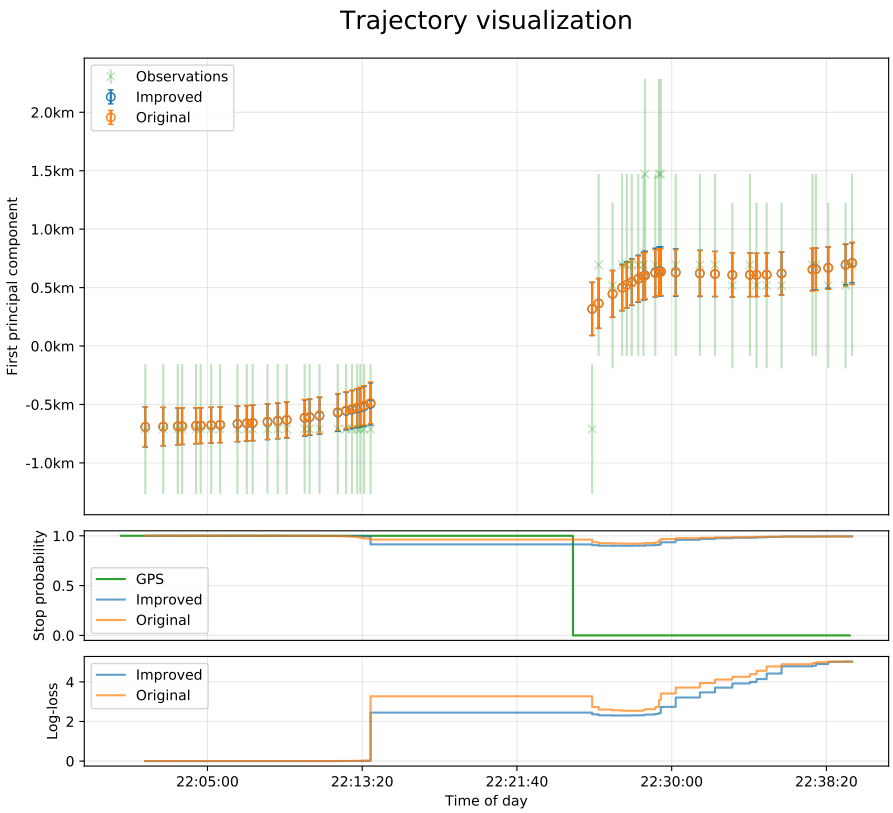


Figure 23. Comparison of the outputs of the original switching Kalman filter and a improvement with sampling rate dependent model switching probabilities.

6. METHODOLOGY TO DETECT AZIMUTH ERRORS IN CELLPLAN

6.1. Motivation, problem statement, and research design

We observed in mobile positioning data artifacts that some passive mobile positioning events clearly did not match the expected behavior, and the observed data could not be explained by cellplan as spatial PDF of position. One possible hypothesis that arose was "feeder swap", i.e., the antennas of the same site are swapped, either wrong azimuths in cellpan data or hardware cables swapped. The latter is unlikely due to the design of newer generations of RAN hardware. The consistency of positioning accuracy is a low priority concern for the engineers in RAN department of mobile operators, and distortions in provided cellplans seemed a plausible hypothesis.

Research question: Do real-life cellplans exhibit "feeder-swap" distortion, i.e., the cellplan has sometimes the azimuth attributes of cells swapped?

6.1.1. Experiment plan

The overall plan was following

1. Feeder-swap was expected be detectable by distortions in map-matched trajectories near antenna towers. Several heuristic indicators were devised and implemented to assess the distortion (ISBEST, azimuth mismatch).
2. The statistics over map-matched trajectories was collected, the classification quality of antenna pairs as "normal" or "swapped" was tested with synthetic swapping methodology, results were presented as ROC curves.
3. The detailed data around most questionable antennas was investigated manually to get expert opinion if there are cells with probable feeder-swap.
4. High-fidelity map-matching technique was expected to be needed to detect the distortions in spatial uncertainty caused by inaccurate mapmatching. The map-matching technique presented in chapter 3 was considered too simplistic and introducing considerable distortions. Therefore a considerably more stable map-matching technique was developed. This map-matching technique is described below in next section.

6.2. Probabilistic map-matching using route hierarchy

This section describes the map-matching technique used during azimuth error experiments.

6.2.1. The goals of probabilistic map-matching technique

We had earlier used the map-matching as described in presented in chapter 3. That technique had some known problems:

- the algorithm just selected a point in a cell area and routed the trajectory through it. This generated a trajectory which mainly was on main road but included short deviations from main road because the selected points were not exactly on main road
- in case of natural barriers like river or railway, the selected point could be on the wrong side of the barrier and thus the generated trajectory would be signifcantly distorted as it had to route around this barrier
- the algorithm did not allow for non-uniform probability distribution inside cell

The goal of new algorithm was to overcome these limitations:

- generate a trajectory without short deviation artefacts
- in trajectory generation consider multiple paths and discard unrealistically ineffective ones
- take into account non-uniform probability distribution in cell

6.2.2. Assumptions used about the likelihood of trajectories

The observed positioning events provide evidence, that the MS was moving through a given cell area. Cell area does not have well-defined shape. Cells are also relatively large. This causes a large uncertainty in detecting the true trajectory.

The following assumptions were used to reconstruct more likely trajectories:

- It is more likely that a person chooses a faster trajectory, i.e., likelihood is related to travel duration.
- It is reasonable to expect that the probability of being served by a given cell decreases with increasing distance. In implemented code, we model the spatial PDF of the cell service area as three concentric polygons around the cell location, where the larger area has lower probability. An example of such a 3-level spatial PDF is illustrated in Figure 24 as three overlapping ellipses.

6.2.3. Selecting representative junctions in the cell area

The cell area is a two-dimensional continuous area. The map-matched trajectory can be estimated from cell visits by utilizing routing algorithms. Effective routing algorithms implement point-to-point routing. We applied a Contraction Hierarchies shortest path algorithm [21]. For point-to-point routing, we approximated the cell with a set of representative junction points. We selected a finite number of likely nodes (junctions) in the road graph to represent the cell area. It was assumed that at least one junction from this set of junctions is traversed during routing.

The selection algorithm of representative junctions utilizes the index built by the Contraction Hierarchy routing algorithm [21] – all nodes in the road network

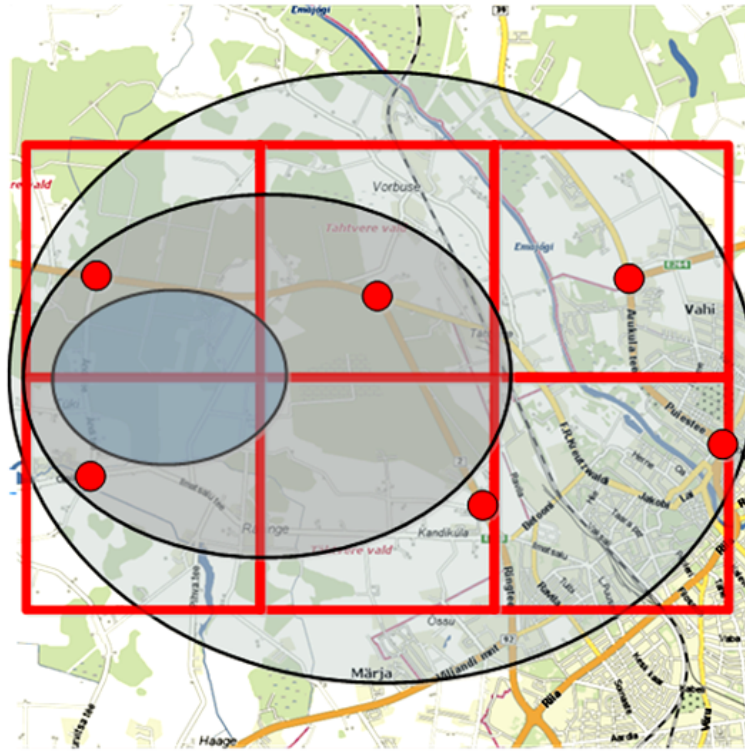


Figure 24. Heuristic discovery of relevant transit junction candidates. Cell area bounding box rectangle is split into $n \times m$ cells. In each cell the road junction of highest hierarchy value is found (red dot).

are strictly ordered by rank. Higher rank is a good predictor that given junction is likely traversed in longer travels. The algorithm worked as follows:

1. From cellplan, determine the bounding box rectangle of the cell area and split it into $n \times m$ smaller rectangles. In Figure 24, the grid is marked with red lines, the grid size is 2×3. n and m are empirical parameters. We usually used $n = m = 4$. in our experiments.
2. Find in each smaller rectangle a node with the highest rank. Only one candidate from each small rectangle was selected. If the rectangle did not contain any road network nodes, then no candidates were generated from this rectangle.

If the cell area did not contain any road network nodes, then the given cell was ignored in map-matching.

6.2.4. Likelihood model

The probability of the path is

$$\mathrm{P}_{path} = e^{-\Sigma_{i=0}^{n} f_i * t_i} * \prod_{i=0}^{n} \mathrm{P}(C_i|L_i) \tag{6.1}$$

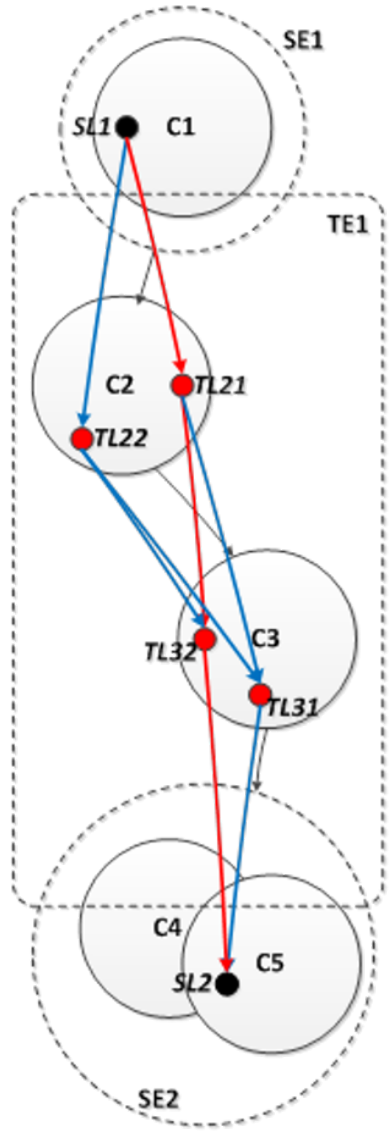


Figure 25. Routing graph through relevant transit junction candidates. ’ The dashed line denotes boundary for Stop Episodes (SE) and Transit Episode (TE) cell visits

where

- $C_0, ..., C_n$ are the cells of the representative events of the clusters (C_0 is the cell of the previous stop cluster, C_n = cell of the current stop cluster, $C_1, .., C_{n-1}$ are cells of transit clusters),
- $\mathrm{P}(C_i|L_i)$ is the probability of being in cell C_i at a geographic location (transit vertex) L_i,
- t_i is the time of the path provided by the routing graph from the vertex chosen from $vertexSet_i$ to the vertex chosen from $vertexSet_{i+1}$
- f_i is a tuning parameter.

We seek the path that maximizes P_{path}. Actually, we solve the equivalent problem of minimizing $log(P_{path}) = -\sum_{n=0}^{n} f_i * t_i - \sum_{i=0}^{n} \log \mathrm{P}(C_i|L_i)$.

In the process of likelihood optimisation, the likelihood of outlier was considered also – if removing single transit event improved the path likelihood more than $\mathrm{P}_{outlier}$ then such event was removed. $\mathrm{P}_{outlier}$ was a tuning parameter. Its value was selected so that a minor fraction of events were classified as outliers. Multiple subsequent events were not removed together.

6.2.5. Generating the map-matching trajectories

A set of map-match trajectories was generated with the following procedure:

- Events were filtered for cell-hopping to produce clusters of events as described in Chapter 3, and these clusters were marked as stop or move.
- Several paths were generated, selecting one random geographical point in each stop cluster.
- For each set of stop locations, the path between stop clusters was generated by finding the minimal solution to Eq. 6.1.

The examples of map-matching are depicted in Figures 26, 28.

6.3. Cellsets and comparing the permutations

In radio access network (RAN), not all cells are equal. For example, cells with 4G technology cannot serve 2G and 3G phones. Therefore it is more useful to compare the behavior of cells that work in the same frequency band and are based on the same technology. Typical antennas used in mobile network have half-power angle 60°, and full 360° horizon is covered typically by 3 to 4 antennas in same band. Such cells on the same site (i.e., on the same antenna tower) we call cellsets.

We will calculate the azimuth consistency measures for all possible cell pair combinations for cells in the cellset – whether swapped or not swapped pair would be better conformance with observed trajectories. We will not compare the behavior of cells that belong to different cellsets. We calculate for each such pair of cells a "badness" function B: a value indicating if original assignment of azimuths is better than swapped azimuth values.

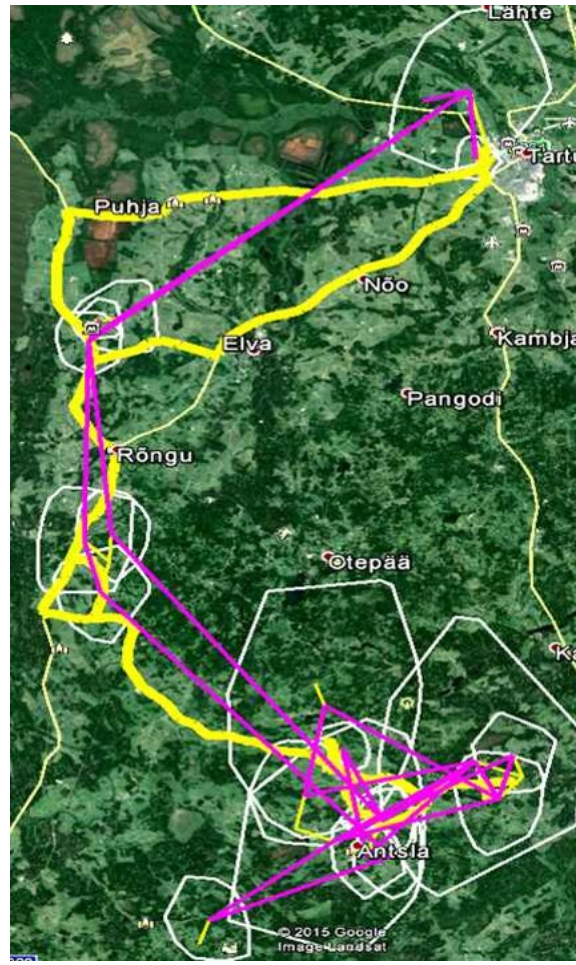


Figure 26. Example of mapmatching from sparse passive positioning data. White polygons are cell areas in cellplan. Violet lines connects cell centroids of positioning events. Yellow line is the reconstructed trajectory. Real trajectory mostly matched the reconstructed trajectory. The lowest cell on left is ignored by the algorithm, due to unrealistic fast cell-hoppings. Map area is circa 46×79 km.

6.4. Calculation process for the statistical indicators of azimuth mismatch

Statistical indicators were calculated to charaterize azimuth vs trajectories mismatch.

6.4.1. Processing the trajectories to find the likelihood of cell permutations and nearest juncions

The calculations were organized in following steps

1. Trajectories from the positioning input data were constructed by map-matching technique described above and full trajectory timeline was split into stop

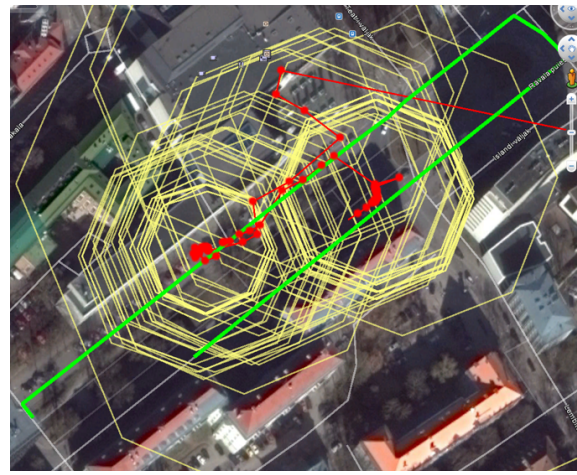


Figure 27. Example of problematic mapmatching from GPS data. Red dots are GPS measurements, red lines connect GPS measurements, yellow polygons are uncertainty areas around GPS measurement, green line is reconstructed trajectory.

and move episodes. Stop was defined as a cluster of consecutive events that is compact in space and exceeds given duration threshold parameter.

2. In each move episode e a stable subset over mapmatched sample trajectories was found. The concept of stable subset is illustrated in Fig. 30.
3. For each cell C visited in stable subset of episode e, the junction $j_{e,C}$ in mapmatched trajectory closest to the cell C site (antenna tower) is found. The junction coordinates together with cell ID are used as an input to calculation of azimuth badness value as described in next subsection.

6.4.2. Azimuth mismatch indicator calculations

Weighted centroid $Z_C = (x_C, y_C)$ of nearest junction visits is calculated for each cell:

$$x_C = \sum_e w(d(C, j_{e,C})) * x_{j_{e,C}} / \sum_e w(d(C, j_{e,C})) \tag{6.2}$$

$$y_C = \sum_e w(d(C, j_{e,C})) * y_{j_{e,C}} / \sum_e w(d(C, j_{e,C})) \tag{6.3}$$

where

- x_C, y_C are weighted centroid coordinates for cell C
- e are all move episodes detected in all trajectories
- $j_{e,C}$ is the nearest road junction on mapmatched trajectory in stable subset of episode e to the site of cell C
- x_j, y_j are the coordinates of junction j
- $d(C, j)$ is the distance between site of cell C and the road junction j
- $w(d)$ is a weight function whose argument d is the distance between the road junction j and cell site.

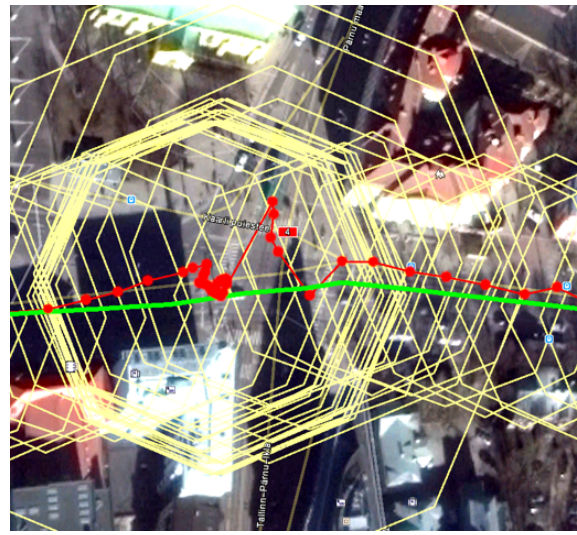


Figure 28. Example of successful mapmatching from GPS data. Red dots are GPS measurements, red lines connect GPS measurements, yellow polygons are uncertainty areas around GPS measurement, green line is reconstructed trajectory.

6.4.3. Calculations per cell pairs

The badness factor B is calculated for all possible combinations of two cells c', c'' which belong to same cellset:

$$\begin{aligned} B =& norm(a_{C'} - az(C', Z_{C'}))^2 + norm(a_{C''} - az(C'', Z_{C''}))^2 \\ & - norm(a_{C'} - az(C', Z_{C''}))^2 - norm(a_{C''} - az(C'', Z_{C'}))^2 \end{aligned}$$

where

- a_C is the azimuth attribute of cell in cellplan
- $az(C,Z)$ is the azimuth from the site of cell C to the junction-visiting centroid Z. The site coordinates are the same for C' and C'' as we specified that they are from the same cellset and and all cells in cellset are located on same site (same antenna tower).
- $norm(\alpha)$ is a normalizing function such that the value of $norm()$ is in range $-180 \leq \alpha \leq 180$ degrees by subtracting or adding full periods to the argument α.

Badness for swapped azimuths B_{swap} is calculated with same formula as B but with swapped $a_{C'}$ and $a_{C''}$:

$$\begin{aligned} B_{swap} =& norm(a_{C''} - az(C', Z_{C'}))^2 + norm(a_{C'} - az(C'', Z_{C''}))^2 \\ & - norm(a_{C''} - az(C', Z_{C'}))^2 - norm(a_{C'} - az(C'', Z_{C'}))^2 \end{aligned}$$

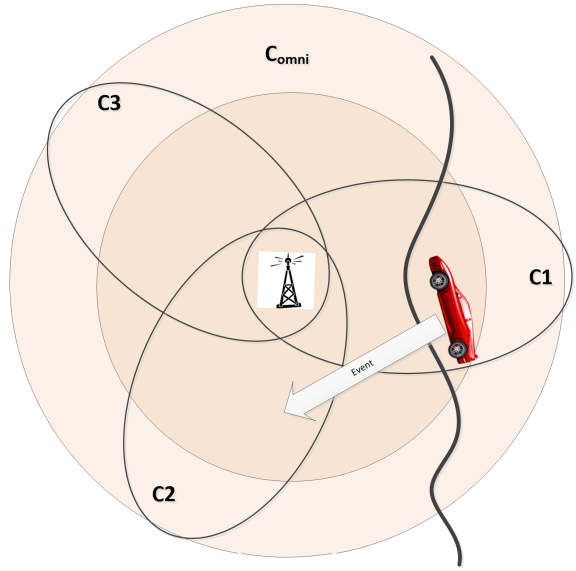


Figure 29. Example of replacing transit cell PDF with omnidirection PDF covering all cells in cellset C1,C2,C3. Suppose the car travelled along road (black line) and cells C1,C2 were swapped so the event was generated by C2 (which in reality had the PDF attributed to C1 in cellplan). In such case the generated road trajectory does not go into areas where C2 PDF is high (as given in cellplan). Therefore we can say that likelihood of permutations that (rightfully) assign to C2 the position of C1 is higher than the likelihood of permutation corresponding to original layout.

The result of these calculations is a dataset with two records per each possible cell pair where cells are from same cellsets. One record for correct badness value and another one for simulated "swapped" state of the pair (cellplan azimuth values swapped, resulting with B_{swap} as badness value). The columns in dataset: cell IDs, badness (B or B_{swap}), swap flag (true/false).

We tested if badness-based classifier could predict the value in swap flag column.

6.5. ROC results

Before the experiment we expected that the azimuth value is unreliable for trajectories that get very close to antenna tower and these should be de-emphasized with lower weights.

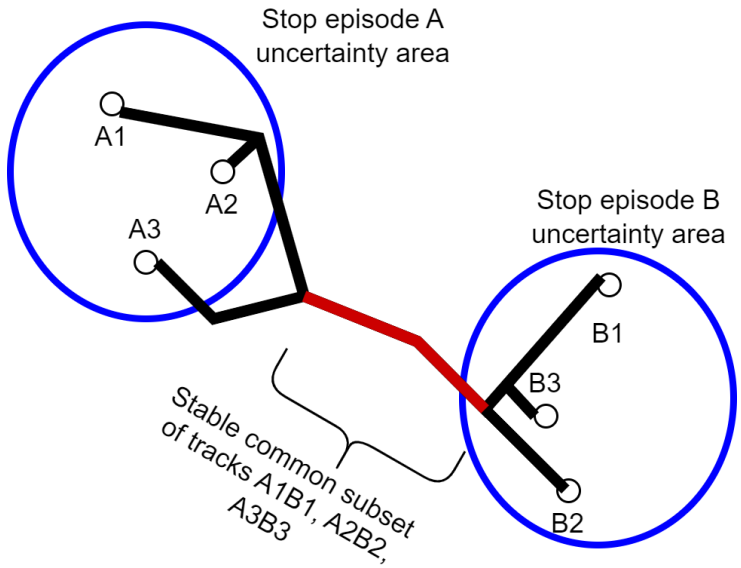


Figure 30. Illustration to stable subset of trajectories. We know that the track is from stop area A to stop area B. We select randomly some predefined number of points in area each area and route map-matched trajectory from point in A to point in B. On this drawing three points are selected and the points define tracks A_1B_1, A_2B_2, A_3B_3. Due to the hierarchical nature of the road networks there exists usually some non-empty common subset of all these trajectories. Common subset is marked with red color.

Therefore the calculations were conducted with three variations of weight function w :

$$w_{UNIFORM}(d) = 1$$

$$w_{LIN500M\_10KM}(d) = \begin{cases} 10^{-30}, & \text{if } d < 500m \\ d[m] - 500, & \text{if } 500m \leq d < 10000m \\ 9500, & \text{otherwise} \end{cases}$$

$$w_{LINEAR5KM}(d) = \begin{cases} d[m], & \text{if } d < 5000m \\ 5000, & \text{otherwise} \end{cases}$$

We applied the methodology in two areas:

- In central Estonia around Paide: flat landscape, no large water bodies, no large towns/cities
- In and near Tallinn: proximity of sea , includes part of Tallinn urban area.

The results did not confirm the assumption that nearby visits carry less azimuth information. The results for Paide and Tallinn areas are represented in Figures 31, 32 as ROC curves.

We manually inspected the cells with the most questionable "badness" value. We did not find any large anomalies; found one kind of perfectly swapped cell that had anomalously low traffic – only a couple of events. These findings had no practical significance.

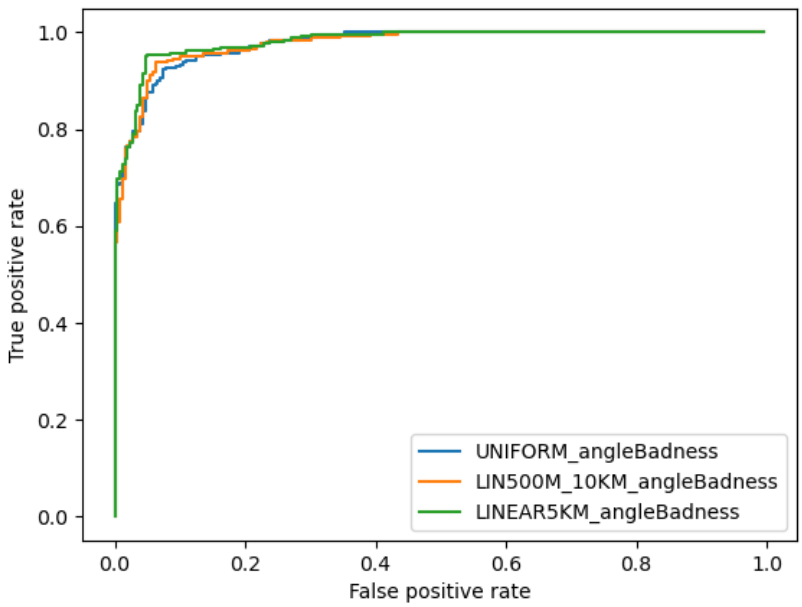


Figure 31. ROC curves for Paide area

6.6. Discussion

The resulting ROC curves show that the method is sensitive to cell swapping artifacts, and did detect the swapped cells introduced artificially into data by our computing experiment.

Inland rural data were handled better than seaside locations. This could be attributed to improved radio wave propagation conditions at large water bodies that cause more overlap and uncertainties in cell boundaries and therefore reduces the accuracy of map-matching. Less pronounced hierarchy in city road network, as compared to rural road network, might also contribute to lower accuracy of map-matching.

The behavior of most suspicious individual cells was investigated manually (examples not included here due to data usage restrictions). In each such case, there was either extremely small data volume or some other explanation available. So no significant azimuth disorders were found in the test dataset. We concluded that for practical work, given MNO had good cellplan azimuth data and our problems in map-matching accuracy are caused by other factors.

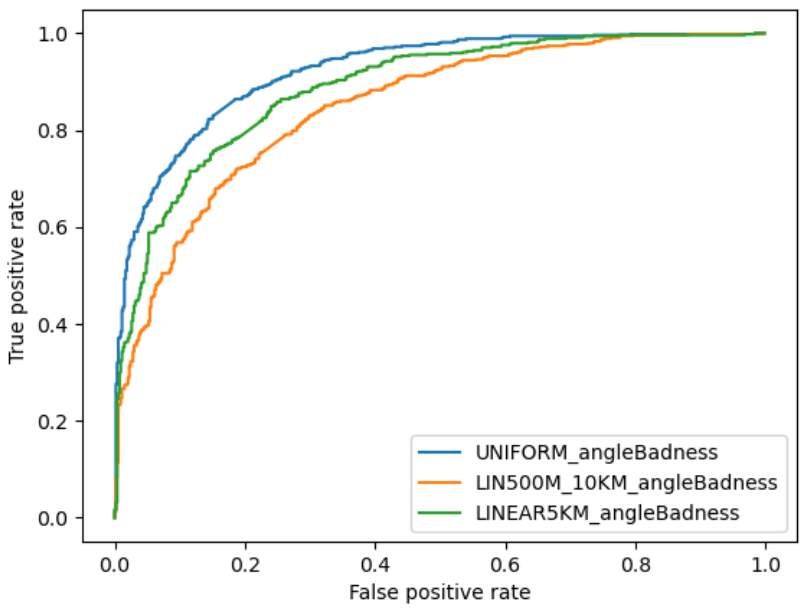


Figure 32. ROC curves for Tallinn area

7. CONCLUSION

When I started with Ph.D. studies, I aimed to develop a generic probabilistic model for the most significant distortion classes in mobile positioning data, and based on that model, I expected to find well-substantiated methods for optimal suppression of these distortions. Such a generic model turned out to be an elusive idea. The source of distortions mainly remained unexplained. We found useful results in some situations and tested some hypotheses where the result was negative, i.e., the hypothesized causes were absent, e.g., the chapter about feeder-swap azimuth errors. During these studies I worked in a company in the area of analyzing passive mobile positioning data. The quantitative improvement was in most cases around 20% and often the effect was not universal. This result was not enough to justify increased complexity of processing and the risk that more complex processing could introduce new distortions.

One possible explanation to modest success could be that there are too many unknowns that vary from case to case – some physical factors (radio wave propagation, reflection, inference), some human geography related (the model of how people actually move and stay), some radio network peculiarities (the complex logic how network selects next serving cell, actual network topology might be more complex than indicated by cellplan – one cell in cellplan could be several in-building nano cells, etc.).

If search for general low parameter count model fails then it is possible that one could achieve better results with neural network models. The main obstacle in this approach is obtaining training data. The model with large number of parameters requires large volume of detailed ground truth training data and it is not trivial to obtain, especially in the era where privacy concerns are recognized more and more.

The positioning data used was obtained by my employer under commercial agreements from mobile operators. GPS data was obtained from volunteers for time-limited usage. The retention time has already passed for all data items and all data are diligently deleted from servers. Therefore the results in this dissertation are not reproducible as it is impossible to obtain the same data.

During our work, we had very limited data that could be used as ground truth. This limitation excluded very complex models. One possible research topic could be using a neural network based model but this assumes large sets of training data.

The privacy of data owners has become much more of a concern in recent years and data processing has become more regulated by the laws. Privacy Enhancing Technologies (PET) are emerging as a (partial) solution to this. One research direction could be finding best practices involving PET such that data owners' privacy is protected and at the same time high accuracy reports can be produced from mobile positioning data.

All-in-all, this effort was an interesting journey.

BIBLIOGRAPHY

[1] Torgil Abrahamsson. Estimation of origin-destination matrices using traffic counts-a literature survey. 1998.

[2] Rein Ahas, Anto Aasa, Antti Roose, Ülar Mark, and Siiri Silm. Evaluating passive mobile positioning data for tourism surveys: An estonian case study. *Tourism Management*, 29(3):469–486, 2008.

[3] Rein Ahas, Anto Aasa, Siiri Silm, Raivo Aunap, H Kalle, and Ülar Mark. Mobile positioning in space–time behaviour studies: social positioning method experiments in estonia. *Cartography and Geographic Information Science*, 34(4):259–273, 2007.

[4] Rein Ahas, Siiri Silm, Erki Saluveer, and Olle Järv. Modelling home and work locations of populations using passive mobile positioning data. *Location based services and TeleCartography II*, pages 301–315, 2009.

[5] Mehdi Amirijoo, Pål Frenger, Fredrik Gunnarsson, Harald Kallin, Johan Moe, and Kristina Zetterberg. Neighbor cell relation list and physical cell identity self-organization in lte. In *Communications Workshops, 2008. ICC Workshops' 08. IEEE International Conference on*, pages 37–41. IEEE, 2008.

[6] Patrix Axelsson and Fredrik Gustafsson. Discretizing stochastic dynamical systems using lyapunov equations. 2014.

[7] Hugo Barbosa, Marc Barthelemy, Gourab Ghoshal, Charlotte R James, Maxime Lenormand, Thomas Louail, Ronaldo Menezes, José J Ramasco, Filippo Simini, and Marcello Tomasini. Human mobility: Models and applications. *Physics Reports*, 734:1–74, 2018.

[8] Oleg Batrashev, Amnir Hadachi, Artjom Lind, and Eero Vainikko. Mobility episode detection from cdr's data using switching kalman filter. In *Proceedings of the Fourth ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems*, pages 63–69. ACM, 2015.

[9] David Bernstein, Alain Kornhauser, et al. An introduction to map matching for personal navigation assistants. 1996.

[10] N Caceres, JP Wideberg, and FG Benitez. Review of traffic data estimations extracted from cellular networks. *IET Intelligent Transport Systems*, 2(3):179–192, 2008.

[11] F Calabrese. Using cell-phone data to understand urban dynamics in the city of amsterdam. *MIT SENSEable City Laboratory.*, 2008.

[12] Francesco Calabrese, Massimo Colonna, Piero Lovisolo, Dario Parata, and Carlo Ratti. Real-time urban monitoring using cell phones: A case study in rome. *Intelligent Transportation Systems, IEEE Transactions on*, 12(1):141–151, 2011.

[13] Francesco Calabrese, Giusy Di Lorenzo, Liang Liu, and Carlo Ratti. Estimating origin-destination flows using opportunistically collected mobile phone location data from one million users in boston metropolitan area. 2011.

[14] Francesco Calabrese, Mi Diao, Giusy Di Lorenzo, Joseph Ferreira Jr, and Carlo Ratti. Understanding individual mobility patterns from urban sensing data: A mobile phone trace example. *Transportation research part C: emerging technologies*, 26:301–313, 2013.

[15] Francesco Calabrese, Laura Ferrari, and Vincent D Blondel. Urban sensing using mobile phone network data: a survey of research. *Acm computing surveys (csur)*, 47(2):25, 2015.

[16] Suma S Cherian and Ashok N Rudrapatna. Lte location technologies and delivery solutions. *Bell Labs Technical Journal*, 18(2):175–194, 2013.

[17] Paul Cloke, Ian Cook, Philip Crang, Mark Goodwin, Joe Painter, and Chris Philo. *Practising human geography*. Sage, 2004.

[18] Andrew T Crooks and Alison J Heppenstall. Introduction to agent-based modelling. In *Agent-based models of geographical systems*, pages 85–105. Springer, 2011.

[19] Maan E El Najjar and Philippe Bonnifait. A road-matching method for precise vehicle localization using belief theory and kalman filtering. *Autonomous Robots*, 19(2):173–191, 2005.

[20] Pierdomenico Fiadino, Danilo Valerio, Fabio Ricciato, and Karin Hummel. Steps towards the extraction of vehicular mobility patterns from 3g signaling data. *Traffic Monitoring and Analysis*, pages 66–80, 2012.

[21] Robert Geisberger, Peter Sanders, Dominik Schultes, and Daniel Delling. Contraction hierarchies: Faster and simpler hierarchical routing in road networks. In *International Workshop on Experimental and Efficient Algorithms*, pages 319–333. Springer, 2008.

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

[23] Fredrik Gustafsson, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, and P-J Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, 2002.

[24] Yaron Hollander and Ronghui Liu. The principles of calibrating traffic microsimulation models. *Transportation*, 35(3):347–362, 2008.

[25] Timothy Hunter, Pieter Abbeel, and Alexandre Bayen. The path inference filter: model-based low-latency map matching of probe vehicle data supplementary materials.

[26] Md Shahadat Iqbal, Charisma F Choudhury, Pu Wang, and Marta C González. Development of origin–destination matrices using mobile phone call data. *Transportation Research Part C: Emerging Technologies*, 40:63–74, 2014.

[27] Marko Jusup, Petter Holme, Kiyoshi Kanazawa, Misako Takayasu, Ivan Romić, Zhen Wang, Sunčana Geček, Tomislav Lipić, Boris Podobnik, Lin Wang, et al. Social physics. *Physics Reports*, 948:1–148, 2022.

[28] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[29] Mohamed Khalaf-Allah and Kyandoghere Kyamakya. Bayesian mobile location in cellular networks. In *2006 14th European Signal Processing Conference*, pages 1–5. IEEE, 2006.

[30] Thomas R Kirchner, Hong Gao, Andrew Anesetti-Rothermel, Heather Carlos, and Brian House. Longitudinal human mobility and real-time access to a national density surface of retail outlets. In *New York City, NY: International Workshop on Urban Computing*, 2014.

[31] John Krumm. Real time destination prediction based on efficient routes. Technical report, SAE Technical Paper, 2006.

[32] Kurt Lewin. Field theory and experiment in social psychology: Concepts and methods. *American journal of sociology*, 44(6):868–896, 1939.

[33] Kevin Manaugh, Luis F Miranda-Moreno, and Ahmed M El-Geneidy. The effect of neighbourhood characteristics, accessibility, home–work location, and demographics on commuting distances. *Transportation*, 37(4):627–646, 2010.

[34] The MathWorks and NIST. Jama : A java matrix package, 2012.

[35] Oleksiy Mazhelis. Using recursive bayesian estimation for matching gps measurements to imperfect road network data. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 1492–1497. IEEE, 2010.

[36] Ronald W McQuaid and Tao Chen. Commuting times–the role of gender, children and part-time work. *Research in transportation economics*, 34(1):66–73, 2012.

[37] Erik Mellegard, Simon Moritz, and Mohamed Zahoor. Origin/destination-estimation using cellular network data. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 891–896. IEEE, 2011.

[38] Kevin P Murphy. Switching kalman filters. 1998.

[39] James R Norris. *Markov chains*. Number 2. Cambridge university press, 1998.

[40] Washington Y Ochieng, Mohammed Quddus, and Robert B Noland. Map-matching in complex urban road networks. 2003.

[41] Erdem Ozdemir, Ahmet E Topcu, and Mehmet Kemal Ozdemir. A hybrid hmm model for travel path inference with sparse gps samples. *Transportation*, 45(1):233–246, 2018.

[42] Alex T Pang, Craig M Wittenbrink, and Suresh K Lodha. Approaches to uncertainty visualization. *The Visual Computer*, 13(8):370–390, 1997.

[43] Christine Parent, Stefano Spaccapietra, Chiara Renso, Gennady Andrienko, Natalia Andrienko, Vania Bogorny, Maria Luisa Damiani, Aris Gkoulalas-Divanis, Jose Macedo, Nikos Pelekis, et al. Semantic trajectories modeling and analysis. *ACM Computing Surveys (CSUR)*, 45(4):1–32, 2013.

[44] Mark G Petovello, Kyle O'Keefe, Gérard Lachapelle, and M Elizabeth Cannon. Consideration of time-correlated errors in a kalman filter applicable to gnss. *Journal of Geodesy*, 83(1):51–56, 2009.

[45] Mohammed A Quddus, Washington Y Ochieng, and Robert B Noland. Current map-matching algorithms for transport applications: State-of-the art and future research directions. *Transportation Research Part C: Emerging Technologies*, 15(5):312–328, 2007.

[46] Herbert E Rauch, CT Striebel, and F Tung. Maximum likelihood estimates of linear dynamic systems. *AIAA journal*, 3(8):1445–1450, 1965.

[47] Dennis M Rose and Thomas Kürner. Outdoor-to-indoor propagation—accurate measuring and modelling of indoor environments at 900 and 1800 mhz. In *Antennas and Propagation (EUCAP), 2012 6th European Conference on*, pages 1440–1444. IEEE, 2012.

[48] Jibonananda Sanyal, Song Zhang, Jamie Dyer, Andrew Mercer, Philip Amburn, and Robert J Moorhead. Noodles: A tool for visualization of numerical weather model ensemble uncertainty. *Visualization and Computer Graphics, IEEE Transactions on*, 16(6):1421–1430, 2010.

[49] Johannes Schlaich, Thomas Otterstätter, and Markus Friedrich. Generating trajectories from mobile phone data. In *Proceedings of the 89th Annual Meeting Compendium of Papers, Transportation Research Board of the National Academies*, 2010.

[50] Markus Schläpfer, Lei Dong, Kevin O'Keeffe, Paolo Santi, Michael Szell, Hadrien Salat, Samuel Anklesaria, Mohammad Vazifeh, Carlo Ratti, and Geoffrey B West. The universal visitation law of human mobility. *Nature*, 593(7860):522–527, 2021.

[51] Abhijit Sharma, Avijit Roy, Suman Ghosal, Rituparna Chaki, and Uma Bhattacharya. Load balancing in cellular network: A review. In *2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12)*, pages 1–5. IEEE, 2012.

[52] Iana Siomina and Di Yuan. Load balancing in heterogeneous lte: Range optimization via cell offset and load-coupling characterization. In *Commu-*

nications (ICC), 2012 IEEE International Conference on, pages 1357–1361. IEEE, 2012.

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

[54] Martijn Tennekes and Yvonne Gootzen. Bayesian location estimation of mobile devices using a signal strength model. *Journal of Spatial Information Science*, (25):29–66, 2022.

[55] Eran Toch, Boaz Lerner, Eyal Ben-Zion, and Irad Ben-Gal. Analyzing large-scale human mobility data: a survey of machine learning methods and applications. *Knowledge and Information Systems*, 58:501–523, 2019.

[56] Toivo Vajakas and Joosep Rõõmusaare. On optimal spatial probability density estimation of passive mobile positioning events. In *2016 15th Biennial Baltic Electronics Conference (BEC)*, pages 127–130. IEEE, 2016.

[57] Toivo Vajakas, Jaan Vajakas, and Rauni Lillemets. Trajectory reconstruction from mobile positioning data using cell-to-cell travel time information. *International Journal of Geographical Information Science*, 29(11):1941–1954, 2015.

[58] Kedong Wang, Yong Li, and Chris Rizos. A new practical approach to kalman filtering with time-correlated measurement errors. 2009.

[59] Lei Wang, Paul D Groves, and Marek K Ziebart. Urban positioning on a smartphone: Real-time shadow matching using gnss and 3d city models. The Institute of Navigation, 2013.

[60] Longgang Xiang, Meng Gao, and Tao Wu. Extracting stops from noisy trajectories: A sequence oriented clustering approach. *ISPRS International Journal of Geo-Information*, 5(3):29, 2016.

[61] Kemeng Yang, Iqbal Gondal, Bin Qiu, and Laurence S Dooley. Combined sinr based vertical handoff algorithm for next generation heterogeneous wireless networks. In *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*, pages 4483–4487. IEEE, 2007.

[62] Yang Ye, Yu Zheng, Yukun Chen, Jianhua Feng, and Xing Xie. Mining individual life pattern based on location history. In *Mobile Data Management: Systems, Services and Middleware, 2009. MDM'09. Tenth International Conference on*, pages 1–10. IEEE, 2009.

[63] Hui Zang, Francois Baccelli, and Jean Bolot. Bayesian inference for localization in cellular networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1–9. IEEE, 2010.

[64] Farzaneh Zangenehnejad and Yang Gao. Gnss smartphones positioning: Advances, challenges, opportunities, and future perspectives. *Satellite navigation*, 2:1–23, 2021.

ACKNOWLEDGEMENT

I got lots of support from many people, and I would like to thank you all. Special thanks to supervisors Eero Vainikko, who guided beginning of my journey, and Amnir Hadachi, who gave necessary support and pressure to get me back into the business when I was ready to give up; to Teet Jagomägi, who provided the opportunity to work on these interesting topics in his company; to Jaan Vajakas and Tanel Kiis who often provided deeper mathematical insight when I was not ready to tackle the problem on my own; to coauthors and colleagues Joosep Rõõmusaare and Rauni Lillemets for valuable contributions and support.

SISUKOKKUVÕTE

Mobiilsidevõrgu andmete kasutamisest inimeste liikuvuse analüüsimisel

See doktoritöö sisaldab passiivse mobiilpositsioneerimise andmete positsioneerimistäpsust käsitlevate uuringute tulemusi

Mobiilioperaatorite serverites tekib tavalise äritegevuse kõrvalsaadusena pidevalt andmeid inimeste käitumise kohta, andmeid on vaja salvestada arveldamiseks ja mobiilivõrgu käigus hoidmiseks. Muuhulgas on nendes andmetes kirjas, mis antenn inimest teatud ajahetkel teenindas. Teades antennide teeninduspiirkondi, saab sedalaadi infost tuletada inimese ligikaudse asukoha ja trajektoori, seda nimetatakse passiivseks mobiilpositsioneerimiseks. Termin "passiivne"tähistab seda, et nende andmete kogumisel ei tehta ühtegi tegevust spetsiaalselt positsioneerimiseks,

Passiivsed mobiilpositsioneerimise andmed on väga hea katvusega üle populatsiooni, enamusel inimestest on sisselülitatud mobiiltelefon kaasas. See muudab passiivse mobiilpositsioneerimise väga ahvatlevaks võimaluseks teha statistilisi uurimusi inimeste liikumise kohta, see pakub huvi ruumiplaneerimises omavalitsustele, transporditeenuse pakkujatele, ettevõtetele oma ettevõtete rajamise planeerimisel. Lisaks kasutatakse passiivset mobiilpositsioneerimist ka asukohapõhiseks reklaamiks; ja mobiilioperaator kasutab positsioneerimist mobiiltelefoni kasutajakogemuse kvaliteedimõõdikute sidumiseks asukohaga, et parendada teenuse kvaliteeti.

Antud töö käsitleb passivset mobiilpositsioneerimist ennekõike inimeste liikuvuse statistilise hindamise kontekstis.

Praktikas on passiivse mobiilpositsioneerimise kasutamisel suureks takistuseks andmete ebatäpsus ja madal kvaliteet. Algpõhjuseks on see, et mobiilvõrk ei ole algselt ehitatud passiivse positsioneerimise kvaliteeti silmas pidades. Andmete täpsus on madal, kuna antenn võib teenindada küllaltki suurt ala, ja ka see tegelik ala pole kuigi täpselt teada. Mingi protsent mobiilvõrgu produtseeritud andmeid on lihtsalt vigased, selleks on erinevaid põhjusi, ja seda ei ole võimalik täielikult vältida. Lisaks on mobiilpositsioneerimisandmed ajas väga ebaühtlaselt jaotunud – teatud ajalõigud võivad olla väga tihedalt andmetega kaetud, samas võib esineda päris pikki ajavahemikke, kus andmed puuduvad, sest midagi ei toimunud mobiilivõrgus. Autori kogemuse kohaselt umbes $70 \pm 20\%$ kirjetest vastas oodatavale jaotusele ja ülejäänud kirjed olid suurte moonutustega (nn raskete sabadega jaotus).

Autoril tekkis hüpotees, et mobiilpositsioneerimise üksikkirjete vahel on tugev korrelatsioon, sest inimese liikumine on pideva iseloomuga, samuti inimene kordab eri päevadel samu mustreid, ja ka eri inimeste telefonid käituvad samades oludes sarnaselt. Selle korrelatsiooni pinnalt oli lootus luuda statistilised mudelid, mis kirjeldaks andmemoonutuste olemust ja millele toetudes saaks efektiivselt

eristada “mõistlikke” andmekirjeid tugeva moonutusega võõrväärtustest.

Töö põhineb neljal publikatsioonil. Kõigepealt antakse ülevaade passiivse mobiilpositsioneerimise valdkonnast koos positsioneerimisandmete tõlgendamisel tekkivate probleemide kirjelduse ja võimalike lahendusteedega.

Mobiilsidevõrgus tekib passiivpositsioneerimisel tihti ping-pong effekt – ka paigalseisev telefon vahetab vahel tihti masti, millega on ühenduses. Seoses sellega tekib näiv liikumine, mis sõltub tugevalt andmete tõlgendusviisist.. Esimeses publikatsioonis võrreldakse erinevaid viise antennide vahelise hüppamise (ping-pong) moonutuste mahasurumiseks ning pakutakse välja uus algoritm, mida testitakse kahe mobiilioperaatori andmete põhjal. Algoritmi testiti kahe Eesti mobiilioperaatori andmetel ja täheldati olulist liiklustiheduse hinnangu täpsuse tõusu, kuni 1,8 korda.

Võrguplaan ehk mobiilsidevõrgu antennide teeninduspiirkondade kirjeldus määratleb positsioneerimisel ruumilise tõenäosustiheduse. Teises publikatsionis esitatakse selle tõenäosustiheduse hindamiseks meetod, mis põhineb Bayesi statistikal, meetodit testitakse väikese andmekogumiga. Meetodi rakendamisel täheldati asukoha määramatuse mõõdukat vähenemist ca 20%. Meetod vähendas ka ruumilise tiheduse visualiseerimisel nähtavaid häirivaid moonutusi.

Kolmandas publikatsioonis esitatakse peatuste ja liikumise episoodide tuvastamise täiustatud meetod. Meetodis kombineeritakse kahte mudelit – põhiolekute (paigalseis, ühtlane liikumine, hüpe) vaheldumist kirjeldav Markovi ahel ja liikumise dünaamikat kirjeldav Kalmani filter. Publikatsioon põhineb varasemal publikatseeritud algoritmil [8]. Kui algne algoritm vaatles sisendit kui diskreetset sündmuste jada, kus protsessi alusskaalaks on sündmuste järjekorranumber, siis täiustatud meetodis arvestatakse pideva ajaskaalaga, milles sündmused toimuvad. Näiteks olekuvahetus on seotud möödunud ajavahemikuga, mitte positsioneerimissündmuste arvuga. Seetõttu mudeli parameetrid muutuvad palju sisulisemalt tõlgendatavaks. Täiustatud meetodit testitakse ka tegelike passiivse mobiilpositsioneerimise andmete põhjal. Test näitas episoodide tuvastamise täpsuse tõusu ca 20%.

Neljandas osas on kirjeldatud võrguplaanis antenni asimuudi atribuutide moonutuse tuvastamise meetod. Meetodi väljatöötamise ajendiks oli oletus, et osa positsioneerimisandmete mittekonsistentsusest tuleneb võrgupaani moonutusest, kus plaanis on antennid omavahel vahetatud (ingl.k. feederswap) ja seetõttu asimuutide väärtused valed. Meetodit testiti tegelike andmete põhjal kahes Eesti piirkonnas. Test näitas, et meetod on piisavalt tundlik, et leida jämedate asimuudivigadega antennipaarid. Konkreetsetes testandmetes jämedaid asimuudivigu ei tuvastatud.

Kokkuvõtteks võib öelda, et töö käigus sai katsetatud mitmeid ideid andmekvaliteedi parandamiseks, eri situatsioonides mõni meetod andis tuntava efekti. Samas universaalset andmete moonutuste mudelit ja universaalset võõrväärtuste andmetest filtreerimise meetodit ei õnnestunud luua.

CURRICULUM VITAE

Personal data

Name:	Toivo Vajakas
Date of birth:	June 20th, 1960
Contact:	tvajakas@gmail.com

Education

2013–	University of Tartu, Institute of Computer Science, PhD studies
1994–1999	University of Tartu, Department of Physics, Institute of Experimental Physics, PhD studies (did not graduate)
1978–1983	University of Tartu, Department of Mathematics, Applied Mathematics (5 years diploma)

Employment

2020–	Software architect and software process lead in Cybernetica AS, Estonia
2006–2020	System architect in Regio/Reach-U AS, Estonia
2000–2006	Software architect in Cognitive Dynamics AS, Estonia
1999–2000	Software engineer in Institute of Physics, Estonia
1997–1999	Guest researcher in Institute of Solid State Physics, Chalmers University of Technology, Sweden
1982–1999	Engineer/senior engineer in Bureau for Complex Scientific Student Research, University of Tartu, Estonia

Teaching

2016–2019	Supervising three MSc theses in Institute of Computer Science, University of Tartu
2005–2006	Realtime Systems course, University of Tartu
2001–2002	Embedded Microcontrollers course, University of Tartu
1996–1997	C language course, University of Tartu

Scientific work

Main fields of interest:

- applying privacy enhancing technologies in IT solutions
- how to build and measure trust in society and in organisations
- mobile positioning, human geography

ELULOOKIRJELDUS

Isikuandmed

Nimi:	Toivo Vajakas
Sünnikuupäev:	20. juuni 1960
Kontakt:	tvajakas@gmail.com

Haridus

2013–	Tartu Ülikool, Arvutiteaduste Instituut, doktoriõpingud
1994–1999	Füüsika osakond, Füüsika-Keemia teaduskond, Tartu ülikool, doktoriõpingud (lõpetamata)
1978–1983	Matemaatika teaduskond, Tartu Riiklik Ülikool, (rakendusmatemaatika, 5a diplom)

Teenistuskäik

2020–	Tarkvara arhitekt ja tarkvaraarenduse protessijuht, Cybernetica AS, Eesti
2006–2020	Süsteemiarhitekt, Regio/Reach-U AS, Eesti
2000–2006	Tarkvara arhitekt, Cognitive Dynamics AS, Eesti
1999–2000	Tarkvara insener, Füüsika Instituut, Eesti
1997–1999	Külalisteadur, Chalmersi Tehnikaülikool, Rootsi
1982-1999	Insener/vaneminsener, Üliõpilaste Kompleksse Teadusliku Uurimistöö Büroo, Tartu Ülikool, Eesti

Õppetöö

- 2016–2019 Kolme magistritöö juhendamine, Arvutiteaduse instituut, Tartu Ülikool
- 2005–2006 Reaalajasüsteemide kursus, Füüsika osakond, Tartu Ülikool
- 2001–2002 Mikroprotsessorite kursus, Füüsika osakond, Tartu Ülikool
- 1996–1997 C keele kursus, Füüsika osakond, Tartu Ülikool

Teadustegevus

Peamised uurimisvaldkonnad:

- privaatsuskaitse tehnoloogiate rakendamine IT lahendustes
- usalduse loomine ühiskonnas ja organisatsioonides
- mobiilpositsioneerimise andmete kasutamine inimgeograafias

DISSERTATIONES INFORMATICAE PREVIOUSLY PUBLISHED IN DISSERTATIONES MATHEMATICAE UNIVERSITATIS TARTUENSIS

19. **Helger Lipmaa.** Secure and efficient time-stamping systems. Tartu, 1999, 56 p.
22. **Kaili Müürisep.** Eesti keele arvutigrammatika: süntaks. Tartu, 2000, 107 lk.
23. **Varmo Vene.** Categorical programming with inductive and coinductive types. Tartu, 2000, 116 p.
24. **Olga Sokratova.** Ω-rings, their flat and projective acts with some applications. Tartu, 2000, 120 p.
27. **Tiina Puolakainen.** Eesti keele arvutigrammatika: morfoloogiline ühestamine. Tartu, 2001, 138 lk.
29. **Jan Villemson.** Size-efficient interval time stamps. Tartu, 2002, 82 p.
45. **Kristo Heero.** Path planning and learning strategies for mobile robots in dynamic partially unknown environments. Tartu 2006, 123 p.
49. **Härmel Nestra.** Iteratively defined transfinite trace semantics and program slicing with respect to them. Tartu 2006, 116 p.
53. **Marina Issakova.** Solving of linear equations, linear inequalities and systems of linear equations in interactive learning environment. Tartu 2007, 170 p.
55. **Kaarel Kaljurand.** Attempto controlled English as a Semantic Web language. Tartu 2007, 162 p.
56. **Mart Anton.** Mechanical modeling of IPMC actuators at large deformations. Tartu 2008, 123 p.
59. **Reimo Palm.** Numerical Comparison of Regularization Algorithms for Solving Ill-Posed Problems. Tartu 2010, 105 p.
61. **Jüri Reimand.** Functional analysis of gene lists, networks and regulatory systems. Tartu 2010, 153 p.
62. **Ahti Peder.** Superpositional Graphs and Finding the Description of Structure by Counting Method. Tartu 2010, 87 p.
64. **Vesal Vojdani.** Static Data Race Analysis of Heap-Manipulating C Programs. Tartu 2010, 137 p.
66. **Mark Fišel.** Optimizing Statistical Machine Translation via Input Modification. Tartu 2011, 104 p.
67. **Margus Niitsoo**. Black-box Oracle Separation Techniques with Applications in Time-stamping. Tartu 2011, 174 p.
71. **Siim Karus.** Maintainability of XML Transformations. Tartu 2011, 142 p.
72. **Margus Treumuth.** A Framework for Asynchronous Dialogue Systems: Concepts, Issues and Design Aspects. Tartu 2011, 95 p.
73. **Dmitri Lepp.** Solving simplification problems in the domain of exponents, monomials and polynomials in interactive learning environment T-algebra. Tartu 2011, 202 p.

74. **Meelis Kull.** Statistical enrichment analysis in algorithms for studying gene regulation. Tartu 2011, 151 p.
77. **Bingsheng Zhang.** Efficient cryptographic protocols for secure and private remote databases. Tartu 2011, 206 p.
78. **Reina Uba.** Merging business process models. Tartu 2011, 166 p.
79. **Uuno Puus.** Structural performance as a success factor in software development projects – Estonian experience. Tartu 2012, 106 p.
81. **Georg Singer.** Web search engines and complex information needs. Tartu 2012, 218 p.
83. **Dan Bogdanov.** Sharemind: programmable secure computations with practical applications. Tartu 2013, 191 p.
84. **Jevgeni Kabanov.** Towards a more productive Java EE ecosystem. Tartu 2013, 151 p.
87. **Margus Freudenthal.** Simpl: A toolkit for Domain-Specific Language development in enterprise information systems. Tartu, 2013, 151 p.
90. **Raivo Kolde.** Methods for re-using public gene expression data. Tartu, 2014, 121 p.
91. **Vladimir Šor.** Statistical Approach for Memory Leak Detection in Java Applications. Tartu, 2014, 155 p.
92. **Naved Ahmed.** Deriving Security Requirements from Business Process Models. Tartu, 2014, 171 p.
94. **Liina Kamm.** Privacy-preserving statistical analysis using secure multi-party computation. Tartu, 2015, 201 p.
100. **Abel Armas Cervantes.** Diagnosing Behavioral Differences between Business Process Models. Tartu, 2015, 193 p.
101. **Fredrik Milani.** On Sub-Processes, Process Variation and their Interplay: An Integrated Divide-and-Conquer Method for Modeling Business Processes with Variation. Tartu, 2015, 164 p.
102. **Huber Raul Flores Macario.** Service-Oriented and Evidence-aware Mobile Cloud Computing. Tartu, 2015, 163 p.
103. **Tauno Metsalu.** Statistical analysis of multivariate data in bioinformatics. Tartu, 2016, 197 p.
104. **Riivo Talviste.** Applying Secure Multi-party Computation in Practice. Tartu, 2016, 144 p.
108. **Siim Orasmaa.** Explorations of the Problem of Broad-coverage and General Domain Event Analysis: The Estonian Experience. Tartu, 2016, 186 p.
109. **Prastudy Mungkas Fauzi.** Efficient Non-interactive Zero-knowledge Protocols in the CRS Model. Tartu, 2017, 193 p.
110. **Pelle Jakovits.** Adapting Scientific Computing Algorithms to Distributed Computing Frameworks. Tartu, 2017, 168 p.
111. **Anna Leontjeva.** Using Generative Models to Combine Static and Sequential Features for Classification. Tartu, 2017, 167 p.
112. **Mozhgan Pourmoradnasseri.** Some Problems Related to Extensions of Polytopes. Tartu, 2017, 168 p.

113. **Jaak Randmets.** Programming Languages for Secure Multi-party Computation Application Development. Tartu, 2017, 172 p.
114. **Alisa Pankova.** Efficient Multiparty Computation Secure against Covert and Active Adversaries. Tartu, 2017, 316 p.
116. **Toomas Saarsen.** On the Structure and Use of Process Models and Their Interplay. Tartu, 2017, 123 p.
121. **Kristjan Korjus.** Analyzing EEG Data and Improving Data Partitioning for Machine Learning Algorithms. Tartu, 2017, 106 p.
122. **Eno Tõnisson.** Differences between Expected Answers and the Answers Offered by Computer Algebra Systems to School Mathematics Equations. Tartu, 2017, 195 p.

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

1. **Abdullah Makkeh**. Applications of Optimization in Some Complex Systems. Tartu 2018, 179 p.
2. **Riivo Kikas**. Analysis of Issue and Dependency Management in Open-Source Software Projects. Tartu 2018, 115 p.
3. **Ehsan Ebrahimi**. Post-Quantum Security in the Presence of Superposition Queries. Tartu 2018, 200 p.
4. **Ilya Verenich**. Explainable Predictive Monitoring of Temporal Measures of Business Processes. Tartu 2019, 151 p.
5. **Yauhen Yakimenka**. Failure Structures of Message-Passing Algorithms in Erasure Decoding and Compressed Sensing. Tartu 2019, 134 p.
6. **Irene Teinemaa**. Predictive and Prescriptive Monitoring of Business Process Outcomes. Tartu 2019, 196 p.
7. **Mohan Liyanage.** A Framework for Mobile Web of Things. Tartu 2019, 131 p.
8. **Toomas Krips.** Improving performance of secure real-number operations. Tartu 2019, 146 p.
9. **Vijayachitra Modhukur.** Profiling of DNA methylation patterns as biomarkers of human disease. Tartu 2019, 134 p.
10. **Elena Sügis.** Integration Methods for Heterogeneous Biological Data. Tartu 2019, 250 p.
11. **Tõnis Tasa.** Bioinformatics Approaches in Personalised Pharmacotherapy. Tartu 2019, 150 p.
12. **Sulev Reisberg.** Developing Computational Solutions for Personalized Medicine. Tartu 2019, 126 p.
13. **Huishi Yin.** Using a Kano-like Model to Facilitate Open Innovation in Requirements Engineering. Tartu 2019, 129 p.
14. **Faiz Ali Shah.** Extracting Information from App Reviews to Facilitate Software Development Activities. Tartu 2020, 149 p.
15. **Adriano Augusto**. Accurate and Efficient Discovery of Process Models from Event Logs. Tartu 2020, 194 p.
16. **Karim Baghery.** Reducing Trust and Improving Security in zk-SNARKs and Commitments. Tartu 2020, 245 p.
17. **Behzad Abdolmaleki.** On Succinct Non-Interactive Zero-Knowledge Protocols Under Weaker Trust Assumptions. Tartu 2020, 209 p.
18. **Janno Siim.** Non-Interactive Shuffle Arguments. Tartu 2020, 154 p.
19. **Ilya Kuzovkin.** Understanding Information Processing in Human Brain by Interpreting Machine Learning Models. Tartu 2020, 149 p.
20. **Orlenys López Pintado.** Collaborative Business Process Execution on the Blockchain: The Caterpillar System. Tartu 2020, 170 p.
21. **Ardi Tampuu.** Neural Networks for Analyzing Biological Data. Tartu 2020, 152 p.

22. **Madis Vasser.** Testing a Computational Theory of Brain Functioning with Virtual Reality. Tartu 2020, 106 p.
23. **Ljubov Jaanuska.** Haar Wavelet Method for Vibration Analysis of Beams and Parameter Quantification. Tartu 2021, 192 p.
24. **Arnis Parsovs.** Estonian Electronic Identity Card and its Security Challenges. Tartu 2021, 214 p.
25. **Kaido Lepik.** Inferring causality between transcriptome and complex traits. Tartu 2021, 224 p.
26. **Tauno Palts.** A Model for Assessing Computational Thinking Skills. Tartu 2021, 134 p.
27. **Liis Kolberg.** Developing and applying bioinformatics tools for gene expression data interpretation. Tartu 2021, 195 p.
28. **Dmytro Fishman.** Developing a data analysis pipeline for automated protein profiling in immunology. Tartu 2021, 155 p.
29. **Ivo Kubjas.** Algebraic Approaches to Problems Arising in Decentralized Systems. Tartu 2021, 120 p.
30. **Hina Anwar.** Towards Greener Software Engineering Using Software Analytics. Tartu 2021, 186 p.
31. **Veronika Plotnikova.** FIN-DM: A Data Mining Process for the Financial Services. Tartu 2021, 197 p.
32. **Manuel Camargo.** Automated Discovery of Business Process Simulation Models From Event Logs: A Hybrid Process Mining and Deep Learning Approach. Tartu 2021, 130 p.
33. **Volodymyr Leno.** Robotic Process Mining: Accelerating the Adoption of Robotic Process Automation. Tartu 2021, 119 p.
34. **Kristjan Krips.** Privacy and Coercion-Resistance in Voting. Tartu 2022, 173 p.
35. **Elizaveta Yankovskaya.** Quality Estimation through Attention. Tartu 2022, 115 p.
36. **Mubashar Iqbal.** Reference Framework for Managing Security Risks Using Blockchain. Tartu 2022, 203 p.
37. **Jakob Mass.** Process Management for Internet of Mobile Things. Tartu 2022, 151 p.
38. **Gamal Elkoumy.** Privacy-Enhancing Technologies for Business Process Mining. Tartu 2022, 135 p.
39. **Lidia Feklistova.** Learners of an Introductory Programming MOOC: Background Variables, Engagement Patterns and Performance. Tartu 2022, 151 p.
40. **Mohamed Ragab.** Bench-Ranking: A Prescriptive Analysis Approach for Large Knowledge Graphs Query Workloads. Tartu 2022, 158 p.
41. **Mohammad Anagreh.** Privacy-Preserving Parallel Computations for Graph Problems. Tartu 2023, 181 p.
42. **Rahul Goel.** Mining Social Well-being Using Mobile Data. Tartu 2023, 104 p.

43. **Anti Ingel.** Algorithms using information theory: classification in brain-computer interfaces and characterising reinforcement-learning agents. Tartu 2023, 142 p.
44. **Shakshi Sharma.** Fighting Misinformation in the Digital Age: A Comprehensive Strategy for Characterizing, Identifying, and Mitigating Misinformation on Online Social Media Platforms. Tartu 2023, 158 p.
45. **Kristiina Rahkema.** Quality Analysis of iOS Applications with Focus on Maintainability and Security Aspects. Tartu 2023, 182 p.
46. **Ivan Slobozhan.** Studying Online Social Media Engagement in CIS Countries during Protests, Mass Demonstrations and War. Tartu 2023, 81 p.
47. **Nurlan Kerimov.** Building a catalogue of molecular quantitative trait loci to interpret complex trait associations. Tartu 2023, 248 p.
48. **Pavlo Tertychnyi.** Machine Learning Methods for Anti-Money Laundering Monitoring. Tartu 2023, 117 p.
49. **Abasi-amefon Obot Affia.** A Framework and Teaching Approach for IoT Security Risk Management. Tartu 2023, 180 p.
50. **Raimond-Hendrik Tunnel.** Video Game Design and Development Bachelor's Curriculum for Estonia. Tartu 2024, 137 p.
51. **Ahto Salumets.** Bioinformatics analysis of various aspects in immunology. Tartu 2024, 198 p.
52. **Mohammed Abdulhameed Shaif Ali.** Deep Learning Methods for Cell Microscopy Image Analysis. Tartu 2024, 143 p.
53. **Pille Pullonen-Raudvere**. Foundations of Efficient and Secure Algorithm Development for Secure Multiparty Computation. Tartu 2024, 265 p.
54. **Marili Rõõm.** Multiple approaches to learners' success and factors affecting it in computer programming MOOCs. Tartu 2024, 170 p.
55. **Shivananda Rangappa Poojara.** Design and Orchestration of Scalable, Event-Driven Serverless Data Pipelines for Internet of Things (IoT) Applications. Tartu 2024, 172 p.
56. **Hassan Abdulgaleel Hassan Salim Eldeeb.** Empowering Machine Learning Pipelines with Automated Feature Engineering. Tartu 2024, 121 p.
57. **Muhammad Uzair.** Soft decision making for agri-food 4.0. Tartu 2024, 158 p.
58. **Kirill Milintsevich.** Estimation of Depression Level from Text: Symptom-Based Approach, External Knowledge, Dataset Validity. Tartu 2024, 130 p.
59. **Maksym Del.** Multilingual and Multi-Domain Representational Patterns Across Transformer-Based Models. Tartu 2024, 131 p.
60. **Kristo Raun.** Adaptive Out-of-order Handling in Streaming Conformance Checking. Tartu 2024, 118 p.