TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

Matti Raitoharju

Linear Models and Approximations in Personal Positioning

Julkaisu 1262 • Publication 1262

Tampere 2014

Tampereen teknillinen yliopisto. Julkaisu 1262
Tampere University of Technology. Publication 1262

Matti Raitoharju

Linear Models and Approximations in Personal Positioning

Thesis for the degree of Doctor of Science in Technology to be presented with due permission for public examination and criticism in Konetalo Building, Auditorium K1702, at Tampere University of Technology, on the 21st of November 2014, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2014

ISBN 978-952-15-3404-1 (printed)
ISBN 978-952-15-3421-8 (PDF)
ISSN 1459-2045

Abstract

If an estimation problem can be modeled with linear equations and normal distributed noise, it has closed form solutions that can be computed efficiently. However, applicability of linear models is limited and when linear models cannot be used nonlinear models are needed. For solving nonlinear models closed form algorithms do not always exist and approximate methods have to be used.

This thesis considers linearity and nonlinearity and it is divided into two main parts along with a background section. In the first part of the thesis I investigate how the measurement nonlinearity can be measured and how the nonlinearity can be reduced. I concentrate on nonlinearity within each prior component of a Gaussian Mixture Filter. The Gaussian Mixture Filter uses a sum of normally distributed components to represent a probability density function. When a nonlinear measurement is used to update the estimate, a local linearization of the measurement function is made within every component. The update of a component results in a poor estimate when nonlinearity within the component is high.

To reduce nonlinearity, the components can be split into smaller components. The main contribution of the first part of the thesis is a novel method for computing the directions of nonlinearity and using this information in splitting a component in such a way that the number of components is minimized while reducing the nonlinearity to a set threshold and approximating the original component well. The applicability of the novel methods introduced in the first part is not restricted to the field of positioning, but they may be applied generally to state estimation with Gaussian Mixture Filters.

In the second part, a different approach for reducing nonlinearity is discussed. Instead of splitting a prior into smaller components to mitigate the effect of nonlinearity, the whole problem is modeled using a linear model. The performance of the linear models com-

pared to nonlinear models is evaluated on three different real-world examples. The linear models are coarser approximations of reality than the nonlinear models, but the results show that in these real world situations they can outperform their nonlinear counterparts.

The first considered problem is the generation of Wireless Local Area Network (WLAN) maps with unlocated fingerprints. The nonlinear models presented use distances between fingerprints and access points, whereas the linear model uses only the information whether a fingerprint and an access point can be received simultaneously. It is shown that when noise level increases, the estimate computed using an iterative method based on the nonlinear model becomes less accurate than the linear model. The noise level resulting in equal accuracies of the linear and nonlinear models is similar to the noise level occurring usually when doing WLAN signal strength measurements.

The second problem discussed is positioning using WLAN. Linear-Gaussian coverage area models for WLAN access points are compared with nonlinear parametric and nonparametric WLAN positioning methods. Results show that by using two linear-Gaussian coverage area models for different received signal strengths values the positioning performance is similar or slightly less accurate than with nonlinear methods, but the database size is smaller and the algorithm is computationally less demanding.

Third, I consider the use of linear-Gaussian coverage area models in pedestrian dead reckoning with measurements from an inertial measurement unit. The pedestrian movement is modeled with a linear model and two nonlinear models. The linear model uses only heading change information from an inertial measurement unit, whereas the nonlinear model can use step length measurements in addition. Pedestrian location estimates are solved for the linear model with a Kalman Filter and for the nonlinear model with an Extended Kalman Filter that linearizes the nonlinear state model at the estimate mean. Results show that the linear model has a better accuracy when the uncertainty of heading is large, which is usually the case when the positioning is started.

Preface

This research was carried out at the Positioning Algorithms group at the Department of Mathematics and at the Department of Automation Science and Engineering, Tampere University of Technology during 2009–2014. The research was funded by Nokia and Tampere Doctoral Programme in Information Science and Engineering. I gratefully acknowledge the additional financial support from Jenny and Antti Wihuri Foundation and Nokia Foundation and the computational resources provided by CSC — IT center for science.

I am grateful to my supervisor Prof. Robert Piché for his guidance and trust in me. I thank my instructor Dr. Simo Ali-Löytty for his counsel and his skills in proofreading equations. I thank all my colleagues at the Positioning Algorithms group, especially Dr. Henri Pesonen for discussions and peer support and Juha Ala-Luhtala for listening patiently all my ideas and reading and suggesting improvements to this thesis. I acknowledge gratefully also help from Philipp Müller and Henri Nurminen in the preparation of this thesis. I thank Dr. Jari Syrjärinne and Dr. Lauri Wirola at Nokia for giving me insight on the research in industry. I would also want to express my gratitude to the pre-examiners of this thesis, Prof. Uwe Hanebeck and Dr. Patrick Robertson, for their comments and suggestions.

A special mention goes to my friends at KeparDI. The clubroom acted as my second office where I could do research on a comfortable sofa with a cup of coffee in my hand and playing cards in the other hand.

I thank my parents, Kirsi and Pentti, for all their support and my sisters, Reetta and Emma, for motivating me in my PhD studies by becoming doctors. Special thanks go to Emma for reading and

commenting my thesis.

Finally, I thank my wife, Jenni, for her love and her help in improving the language and comprehensibility of this thesis and my daughter, Annikki, for scheduling my days between interesting work and important play.

Tampere, November 2014,

Matti Raitoharju

Contents

List of publications

This thesis consists of an introduction and the following publications, in chronological order:

P1. "Estimation of basestation position using timing advance measurements" (with Simo Ali-Löytty and Lauri Wirola). In *Proceedings of the International Conference on Signal and Information Processing (ICSIP 2010)*, pages 182–186, Changsha, China, December 2010.

P2. "An adaptive derivative free method for Bayesian posterior approximation" (with Simo Ali-Löytty). In *IEEE Signal Processing Letters*, February 2012.

P3. "Using unlocated fingerprints in generation of WLAN maps for indoor positioning" (with Toni Fadjukoff, Simo Ali-Löytty, and Robert Piché). In *Proceedings of the IEEE/ION Position Location and Navigation Symposium (PLANS 2012)*, pages 576–583, Myrtle Beach, USA, April 2012.

P4. "Positioning with multilevel coverage area models' (with Marzieh Dashti, Simo Ali-Löytty, and Robert Piché). In *Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN 2012)*, Sydney, Australia, November 2012.

P5. "A linear state model for PDR+WLAN positioning" (with Henri Nurminen and Robert Piché). In *Proceedings of the Conference on Design and Architectures for Signal and Image Processing (DASIP 2013)*, Cagliari, Italy, October 2013.

P6. "A field test of parametric WLAN-fingerprint-positioning methods" (with Philipp Müller and Robert Piché). In *Proceedings of the 17th International Conference on Information Fusion (FUSION 2014)*, Salamanca, Spain, July 2014.

P7. "Binomial Gaussian Mixture Filter" (with Simo Ali-Löytty and Robert Piché). *EURASIP Journal on Advances in Signal Processing* (Under review).

The main contributions of the publications are:

- GSM base station is located using timing advance measurements [P1].

- Evaluation of measurement nonlinearity and finding the direction of maximum nonlinearity [P2].
- Algorithms for generating radio map for radio map for positioning using WLAN having location measurements only outdoors was developed in [P3].
- In [P4] WLAN positioning based on Bayesian coverage area estimates is enhanced by using multiple coverage areas.
- In [P5] a pedestrian dead reckoning is performed by using a linear state model instead of linearized nonlinear model. According to evaluation the linear model has advantages over the linearized model.
- In [P6] different parametric models for WLAN positioning are compared in real data test scenarios.
- In [P7] Binomial Gaussian Mixture Filter is presented. It is shown that the Binomial Gaussian Mixture converges to the prior in the sense of probability density function and cumulative distribution function. Furthermore, an algorithm for generating a Binomial Gaussian Mixture that reduces nonlinearity to a given value, while minimizing number of components required for a good approximation of the prior, is presented.

The author's role in the shared publications:

- Publication [P1]: the author implemented the presented algorithms, collected data, and wrote the manuscript.
- Publication [P2]: the author derived the algorithms and wrote the manuscript.
- Publication [P3]: the author developed the models, implemented the required algorithms, and wrote the manuscript.
- Publication [P4]: the author wrote the code, carried out the tests, and wrote the results section of the manuscript.
- Publication [P5]: the author developed the model, implemented the code, and wrote the manuscript.
- Publication [P6]: the author implemented the presented algorithms, carried out the tests, and prepared the figures 2 to 6.
- Publication [P7]: the author developed the algorithm and the code and wrote the manuscript.

Abbreviations

AP	Access Point
APLS	Access Point Least Squares
AS	Adaptive Splitting
BinoGM	Binomial Gaussian Mixture
BinoGMF	Binomial Gaussian Mixture Filter
BS	Base Station
CA	Coverage Area
cdf	cumulative distribution function
EKF	Extended Kalman Filter
FP	Fingerprint
GGM	Generalized Gaussian Mixture
GGMF	Generalized Gaussian Mixture Filter
GM	Gaussian Mixture
GMEM	Gaussian Mixture Expectation Maximization
GMF	Gaussian Mixture Filter
GN	Gauss-Newton
GNMax	Gauss-Newton Max Range
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSM	Global System for Mobile Communications
IMU	Inertial Measurement Unit
KF	Kalman Filter
LTE	Long Term Evolution
pdf	probability density function
PDR	Pedestrian Dead Reckoning
PF	Particle Filter
PL	Pathloss
RSS	Received Signal Strength
SLAM	Simultaneous Localization and Mapping
TA	Timing Advance
TDoA	Time Difference of Arrival
UKF	Unscented Kalman Filter
VARAPLS	Variance Access Point Least Squares
WKNN	Weighted k-nearest Neighbor
WLAN	Wireless Local Area Network
WLLS	Weighted Linear Least Squares

Symbols

α	pathloss exponent
β	component split parameter
Γ	weighting factor
δ	displacement vector
$\delta(x)$	Dirac delta function
$\Delta\theta$	change of heading
ϵ	sample from a probability distribution
ε_{GPS}	GPS location error
ε_x	state transition error
ε_y	measurement error
ε_z	fingerprint location error
η	amount of nonlinearity
θ	heading
λ	eigenvalue
Λ	diagonal matrix containing eigenvalues
μ	mean of a state or a part of state
μ^-	mean of a prior state
μ_0	mean before splitting
$\hat{\mu}$	resampled particle mean
σ^2	variance of a scalar
Σ	covariance matrix of a coverage area
χ	sigma point
ω	weight in Particle Filter resampling
a	vector for component splitting
c	mean of a coverage area
d	dimension of a measurement
e_i	ith column of an identity matrix
$f(x)$	state transition function
$f(x, \varepsilon_x)$	general state transition function
F	state transition matrix
$h(x)$	measurement function
H	Hessian matrix
$\hat{H}$	decorrelated Hessian
I	identity matrix
J	Jacobian matrix
K	Kalman gain

$L(P)$	matrix square root: $P = L(P)\,L(P)^T$
m	number of components or particles
M	number of samples
$\mathrm{N}(\mu, P)$	normal distribution
n	dimension of a state
$p(x)$	pdf of random variable x
$p_{\mathrm{GMF}}(x)$	pdf of a GMF
$p_{\mathrm{N}}(x)$	pdf of a normal distribution
P,P_{xx}	state covariance
P^-	prior covariance
P_0	covariance before splitting
P_{xy}	cross covariance matrix of x and $h(x)$
P_{yy}	covariance matrix of $h(x)$
P_ε	covariance matrix of linearization error
r_{AP}	access point location
r_{BS}	base station location
R	measurement error covariance
RSS	Received Signal Strength
RSS_0	RSS 1 m from source
s	footstep length
S	innovation covariance
t	time index
u	eigenvector
U	matrix containing eigenvectors
$\mathrm{U}([a,b])$	uniform distribution on interval $[a,b]$
v	footstep vector
w	weight
w^+	posterior weight
$\hat{w}$	resampled particle weight
W	covariance of a state transition error
x	state
x^-	prior state
y	measurement value
y^-	predicted measurement
$\hat{y}$	decorrelated measurement
z	fingerprint or user location
z_{GPS}	location measured with GPS

$\alpha_{\mathrm{UKF}}, \beta_{\mathrm{UKF}}, \kappa_{\mathrm{UKF}}, \xi_{\mathrm{UKF}}$	UKF parameters
$\nu, \tilde{\Sigma}, \tau, \phi, l, T,$	variables used in CA fitting
γ, Δ, Q	parameters for nonlinearity estimation
A, b	parameters for linearization $f(x) \approx Ax + b$
$A_{[i,j]}$	jth element of ith row of matrix A
$A_{[:,j]}$	jth column of matrix A
$A_{(t)}$	matrix A at time t
A_i	ith matrix A
$x\|y$	conditional probability of x given y

CHAPTER 1

Introduction

This thesis consists of an introduction, six articles published in scientific conferences and journals, and one article that is under review for a scientific journal. The purpose of this introductory chapter is not to repeat the derivations or results given in the publications [P1]–[P7], but rather to give a short unified background, and summarize the contribution in context.

This introduction is divided into three parts. The first part presents the background of personal positioning and algorithms for Bayesian estimation. In the second part, situations where a nonlinear measurement is used to update a prior are investigated. Measures for determining the amount and direction of nonlinearity are presented. These measures are used in Gaussian Mixture Filters (GMFs) to split prior components into smaller components in such a way that nonlinearity is not high within them. The methods found in literature are compared with the contribution of the author. In the third part, positioning problems are modeled with linear and nonlinear models and results are compared. It is shown that there are situations when a nonlinear model can be replaced with a simpler linear model without loss in accuracy.

The contribution of my publications is presented in the different sections as follows: The model for presenting a range measurement as a Gaussian Mixture (GM) from Publication [P1] is used as an example of different GMFs in the background section (Section 1). The main contributions of publications [P2] and [P7] are compared with the existing methods in Section 2. The publication [P3] is presented in Section 3.1. The linear Coverage Area (CA) models in Section 3.2 are from [P4] and the evaluation of them with other Wireless Local Area Network (WLAN) positioning methods is from [P6]. Publication [P5] is presented in Section 3.3.

1 Background

1.1 Personal positioning

Outdoors personal positioning is usually based on Global Navigation Satellite Systems (GNSSs), such as Global Positioning System (GPS). In GNSSs satellites emit signals that contain the information of when and where the signal was emitted. The radio signals travel at the speed of light and the distances to satellites depend on the time difference between emitting and receiving times. The position can be solved from a set of time difference equations.

Positioning indoors is more complex, because the GNSS signals are too weak to penetrate into buildings or, if they are received, they can be reflected. The time of flight of reflected signals is longer than the shortest path to the satellites. Indoors movement is usually done by foot, which is can be harder to predict compared to, for example, car navigation where movement is confined on roads. There is no globally working accurate indoor positioning system in use, although cellular network protocols have support for positioning that works also indoors [78]. The cellular positioning can achieve accuracy of tens of meters [47, 51, 56], which is not accurate enough for a room-level positioning. Some promising results for using Time Difference of Arrival (TDoA) measurements in Long Term Evolution (LTE) cellular system are introduced in [24]. In their tests median errors less than 5 m were achieved. In the test setup Base Stations (BSs) were close to the test building, which may reduce noise in signals and improve the accuracy.

In practice, there are three choices for providing location estimates indoors:

1. Construct a dedicated positioning system
2. Use inertial navigation
3. Use available signals that are not designed for positioning

These options are not exclusive and can be used with each other.

The construction of a dedicated positioning system indoors requires installation of positioning infrastructure. Dedicated positioning system may also require specific hardware for users. Dedicated indoor

positioning systems can be expensive to install and they cause maintenance costs. At least for now they do not have extensive coverage.

Positioning using inertial navigation is based on computing the user's route using information on orientation and relative movement. Inertial positioning is traditionally done using Inertial Measurement Unit (IMU) that consists of accelerometers and gyroscopes [80]. Other measurements sources, such as magnetometer (compass) for measuring heading and barometer for altitude estimation, can be fused with an IMU [59]. Also a camera can be used to measure the relative motion [36].

In indoor scenarios the user is usually pedestrian and specific information of pedestrian movement is exploited in Pedestrian Dead Reckoning (PDR) [12, 38, P5]. In PDR steps and possibly step lengths are detected and these are combined with heading change.

Positioning based on inertial navigation requires absolute position estimates from other sources. If a GNSS location is available before entering a building, the positioning can be done indoors for a while based on IMU data only, but eventually the location estimate will drift from the true location. The use of floor plans can help reduce the drift, but the use of a floor plan requires knowledge on which map (e.g. which building, which floor) the positioning is done. It is also possible to generate a map from IMU measurements only [63]. In such case the obtained floor plan should still be matched to a database of floor plans to obtain absolute position information.

Using available signals has the advantage that no additional equipment is needed for positioning. Some available signals that can be used as positioning measurements in smart phones are:

- Availability of telecommunications signals e.g. is a BS or an Access Point (AP) receivable
- Time of flight e.g. Timing Advance (TA) of Global System for Mobile Communications (GSM)
- Signal strength e.g. Received Signal Strength (RSS) from Wireless Local Area Network (WLAN)
- Strength of the magnetic field e.g. steel structures of buildings affect to the strength of the magnetic field

- Images from camera e.g. shape of a room can be extracted from an image and then matched to the floorplan

Because these signals are not intended for positioning they do not contain position information and the use of these measurements requires mapping of the signal environment. Generating and updating of radio maps require different amounts of work with different signals. Smart phones contain also low-cost inertial sensors and it is possible to combine positioning using radio signals with information from inertial sensors [77].

The use of TA measurement requires localization of GSM BSs [P1]. The localization can be done with a device that is equipped with GSM for receiving TA values and GPS for getting absolute positions. In this case there is no need for user interaction in the mapping phase and the solved BS locations can be later used in solving locations of devices that do not have GPS available.

Making an indoor radio map of RSS values of WLAN APs for indoor positioning requires significantly more work than locating GSM BSs. For generating an indoor map for WLAN positioning Fingerprints (FPs) are collected. A FP is a measurement report that contains RSS values at a certain location. Because GPS is not available indoors, the measurement locations have to be entered into the system manually or an indoor positioning system has to be available when the measurements are done. The density of FPs and measurement duration at these locations affects the positioning accuracy. The deployment of WLAN APs is not regulated and anyone can install an AP and they can be moved. This generates a constant need for updating the radio map.

In the location fingerprinting, positioning is done by comparing the current RSS values to the RSS values in FPs stored in the database [33]. Building radio maps for large areas for location fingerprinting requires a huge amount of data, because every RSS value and corresponding IDs of APs in every FP has to be stored. Parametric methods, for example multilevel Coverage Areas (CAs) presented in [P4], require only a fixed number of parameters for every AP, and thus can be used to reduce the database size [P6]. Also the required

bandwidth for transmitting positioning data [79] may affect what kind of radio maps can be used and, thus, the positioning accuracy.

A method for reducing the work needed in radio map generation was presented in [P3]. The method generates a radio map for indoor WLAN positioning using FPs collected outdoors with GPS locations and FPs collected indoors without location information. This allows automated radio map generation, but the positioning accuracy drops from room-level accuracy, obtained with radio maps generated with accurate indoor location data, to wing-level accuracy.

One approach for constructing radio maps (and other maps) of an unknown environment is Simultaneous Localization and Mapping (SLAM). In SLAM, the map of the environment is constructed while exploring it. The relative motion is provided by sensors like IMU. In some SLAM-based systems other measurement sources are used to refine the whole track. In Graph SLAM [75], occasional GPS fixes are used to improve the mapping results. In [34], the information of WLAN is used. The use of WLAN is based on the assumption that, when the RSS values are similar, the corresponding locations on a track are close to each other. This helps SLAM to avoid drift that would be present in an IMU only based system. Publication [64] presents a method for constructing a map modeling pedestrian movement possibilities within a building. The method requires a foot mounted IMU as the only measurement source.

WLAN round trip times from a receiver to an AP can be also used for positioning. The use of round trip time requires that the WLAN APs are localized. WLAN based on IEEE 802.11b standard has a theoretical granularity of 6.8 m for measuring the range i.e. the distance between WLAN AP and the user. Positioning based on WLAN round trip times can achieve accuracy of a couple of meters [57]. A problem with using WLAN round trip times in consumer devices is that the common WLAN cards use timestamps with granularity of 1 μs [27]. During that time the signal has traveled already 300 m, which is more than the range of a normal WLAN AP and the round trip time does not provide any information.

In addition to using camera as a gyroscope [36], the visual data can

be used for absolute positioning. In this approach the image is segmented and then matched to a map of the environment. The map can be a floorplan [31] or it can contain other landmarks that can be detected from images [46].

The ambient magnetic field changes within buildings, because construction materials, e.g. steel structures, affect it. This can also be used as a navigation aid. Positioning using magnetic fields require also a mapping of the environment. [26, 29]

In addition to the already mentioned aspects also the computational resources found in mobile devices may restrict the applicable algorithms, e.g. in [53] a tablet computer could process in real time a Particle Filter (PF) algorithm with 400 particles, which is still too few to have same estimated route on every test run.

1.2 Estimation algorithms

Estimation in the context of this thesis is the process of finding an estimate of a state given measurements and possibly a prior state. In positioning the state x contains 2 or 3 position variables and possibly other variables e.g. velocity. In this section a few commonly used estimation algorithms are presented. The presented algorithms will be used later in this introduction and Gaussian Mixture Filters (GMFs) will be discussed in detail in Section 2.

Sequential Bayesian Filter

The estimation of the probability distribution of the state x at time t can be done using a direct application of the Bayes' rule

$$p(x_{(0:t)}|y_{(1:t)}) = \frac{p(y_{(1:t)}|x_{(0:t)})p(x_{(0:t)})}{p(y_{(1:t)})}, \tag{1}$$

where $p(x_{(0:t)})$ is the prior probability density function (pdf) of the state defined by a dynamic model, $p(y_{(1:t)}|x_{(0:t)})$ is the likelihood model for the measurements $y_{(1:t)}$ and $p(y_{(1:t)})$ is the normalization constant defined as

$$p(y_{(1:t)}) = \int p(y_{(1:t)}|x_{(0:t)})p(x_{(0:t)})\mathrm{d}x_{(0:t)} \text{ [67].} \tag{2}$$

This formulation requires computation of the probability distributions of every time step at once. In the sequential estimation some assumptions are made to make estimation more feasible. According to these assumptions, the predicted state depends only on the previous state

$$p(x_{(t)}|x_{(0:t-1)}) = p(x_{(t)}|x_{(t-1)}) \tag{3}$$

and the measurement likelihood depends only on the current state

$$p(y_{(t)}|x_{(0:t)}) = p(y_{(t)}|x_{(t)}). \tag{4}$$

Using these assumptions, the marginal distribution of the state at time step t can be obtained using the following recursion steps [18]: Prediction:

$$p(x_{(t)}|y_{(1:t-1)}) = \int p(x_{(t)}|x_{(t-1)})p(x_{(t-1)}|y_{(1:t-1)})\mathrm{d}x_{(t-1)} \tag{5}$$

Update:

$$p(x_{(t)}|y_{(1:t)}) = \frac{p(y_{(t)}|x_{(t)})p(x_{(t)}|y_{(1:t-1)})}{\int p(y_{(t)}|x_{(t)})p(x_{(t)}|y_{(1:t-1)})\mathrm{d}x_{(t)}} \tag{6}$$

These two steps form the Sequential Bayesian Filter.

Even though the recursion formulas are simpler than (1) they do not have general analytic solutions. In the following some algorithms that are either exact solutions to a special case or approximate algorithms for estimation in more general situations are presented.

In the algorithm presentations the following shorthand notations are used $x^{-}_{(t)} = x_{(t)}|y_{(1:t-1)}$ and $x_{(t)} = x_{(t)}|y_{(1:t)}$.

Weighted linear least squares

Weighted Linear Least Squares (WLLS) can be used in situations where x is related to measurement y through a noisy set of linear equations

$$y = Jx + \varepsilon_y, \tag{7}$$

where J is a Jacobian matrix and ε_y is a normal distributed zero mean noise term with a nonsingular covariance matrix R. WLLS can

be interpreted in the sequential Bayesian framework to be a single update of an uninformative prior. The WLLS estimate of x is normal distributed and has mean

$$\mu = (J^T R^{-1} J)^{-1} J^T R^{-1} y \tag{8}$$

and covariance

$$P = (J^T R^{-1} J)^{-1}. \tag{9}$$

If the rank of J is less than the dimension of the state, the system has an infinite number of solutions and matrix $(J^T R^{-1} J)$ is singular and does not have an inverse.

Kalman Filter

The Kalman Filter (KF) is an algorithm for doing estimation in time series. It was first proposed in [42] and its Bayesian filter interpretation in [32]. The estimate produced by the KF is the optimal Bayesian estimate if the system is linear and noise terms are white and normal distributed. In this kind of system the state evolution can be written as

$$x^-_{(t)} = F_{(t)} x_{(t-1)} + \varepsilon_x, \tag{10}$$

where $x^-_{(t)}$ is the prior state at time t, $F_{(t)}$ is a state transition matrix and ε_x is a zero mean normal distributed noise term that is independent of the state or measurements at other time steps. The prior is updated with linear measurements of the form

$$y_{(t)} = J_{(t)} x_{(t)} + \varepsilon_y, \tag{11}$$

where $y_{(t)}$ is the measurement value, $J_{(t)}$ is a measurement matrix and ε_y is a normal distributed noise term that is independent of state.

The KF has two stages:

1. Prediction:

$$\mu^-_{(t)} = F_{(t)} \mu_{(t-1)} \tag{12}$$

$$P^-_{(t)} = F_{(t)} P_{(t-1)} F^T_{(t)} + W_{(t)}, \tag{13}$$

where P is the covariance matrix of the state, W is the covariance matrix of the state transition noise and the variables with superscript $^-$ are parameters of the predicted state.

2. Update:

$$y^-_{(t)} = J_{(t)}\mu^-_{(t)} \tag{14}$$
$$S_{(t)} = J_{(t)}P^-_{(t)}J^T_{(t)} + R_{(t)} \tag{15}$$
$$K_{(t)} = P^-_{(t)}J^T_{(t)}S^{-1}_{(t)} \tag{16}$$
$$\mu_{(t)} = \mu^-_{(t)} + K_{(t)}(y_{(t)} - y^-_{(t)}) \tag{17}$$
$$P_{(t)} = (I - K_{(t)}J_{(t)})P^-_{(t)}. \tag{18}$$

Equation (14) gives the predicted measurement value y^- at prior mean μ^-. The matrix S is called the innovation covariance and K is the Kalman gain. The updated state mean and covariance are computed in (17) and (18).

The prediction and update steps are not required to be done in order and either step can be applied several times without the other. For example, in situations where no new measurements are available, the update steps can be repeated until a new measurement becomes available. Compared to the static solution computed using the WLLS the KF has the benefit that the state estimate contains information from all measurements until time t instead of just from the current time and the measurement matrix does not need to be full rank. For example, it is possible to estimate a state containing location and velocity variables using only location measurements.

Kalman Filter extensions

If the state or measurement model is nonlinear, the KF cannot be applied directly, but there are several extensions of the KF that allow the use of nonlinear state models or measurements [5, 37, 39]. These extensions apply some kind of linearization to the nonlinear models and then update the state similarly to the KF.

In the Extended Kalman Filter (EKF) the nonlinear state transition and measurement functions are approximated using the first order Taylor approximations [37]. The state transition model is

$$x^-_{(t)} = f_{(t)}(x_{(t-1)}) + \varepsilon_x, \tag{19}$$

where $f_{(t)}(x)$ is a nonlinear state transition function. The measurement model is

$$y_{(t)} = h_{(t)}(x_{(t)}) + \varepsilon_y, \tag{20}$$

where $h_{(t)}(x)$ is a nonlinear measurement function.

The EKF update is done by applying the following substitutions in the KF equations:

$$\mu^-_{(t)} \approx f(\mu_{(t-1)}) \qquad \text{in (12)} \tag{21}$$

$$F_{(t)} \approx \left.\frac{\partial f(x)}{\partial x}\right|_{x=\mu_{(t-1)}} \qquad \text{in (13)} \tag{22}$$

$$y^-_{(t)} \approx h(\mu^-_{(t)}) \qquad \text{in (14)} \tag{23}$$

$$J_{(t)} \approx \left.\frac{\partial h(x)}{\partial x}\right|_{x=\mu^-_{(t)}} \qquad \text{in (15 - 18).} \tag{24}$$

In some cases the analytical differentiation in (22) or (24) may be difficult.

Figure 1 shows an update of a Gaussian prior with the EKF. The measurement function applied to the prior is a second order polynomial

$$y = 2 = \frac{x^3}{2} + x^2 - x + \varepsilon_y. \tag{25}$$

The true measurement likelihood and the posterior are multimodal. The estimate computed with the EKF is unimodal and it is located at the same position as the left peak of the true posterior, but the minor peak on the right is not in the EKF estimate at all.

Another KF extension, the Unscented Kalman Filter (UKF) [39] does not require analytical differentiation and can be used instead of the EKF. In the UKF a set of sigma points is chosen so that they have the same mean μ and variance P as the original distribution. The sigma points are propagated through nonlinear functions and the estimate is updated using the transformed sigma points. The UKF can be used also in situations where the state transition and measurement noises are non-additive, but here I present an UKF version for additive noise.

A popular choice of sigma points is [39]

$$\chi_0 = \mu$$

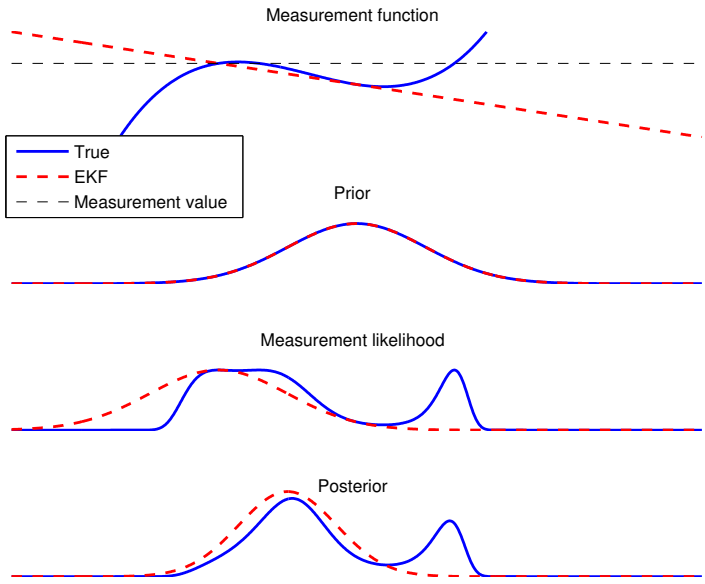

Figure 1: Update of a prior with a nonlinear measurement using the EKF. At the top the measurement function, its linearized approximation, and the measurement value are shown. In the second plot the normal distributed prior pdf is shown. The third plot shows the true likelihood and likelihood of the linearized model used in the EKF. The likelihood has maximum values at locations where the measurement function or its approximation has the same value as the realized measurement value is. In the bottom plot the true posterior pdf is compared with the one obtained with the EKF.

$$\chi_i = \mu + \sqrt{n+\xi_{\text{UKF}}} L(P)_{[:,i]}, \qquad 1 \le i \le n \tag{26}$$
$$\chi_i = \mu - \sqrt{n+\xi_{\text{UKF}}} L(P)_{[:,i-n]}, \qquad n+1 \le i \le 2n,$$

where n is the dimension of state, ξ_{UKF} is an algorithm parameter and $L(P)$ is a matrix square root for which

$$L(P)\, L(P)^T = P. \tag{27}$$

This matrix square root can be computed, for example, with Cholesky decomposition. Parameter ξ_{UKF} is defined as

$$\xi_{\text{UKF}} = \alpha_{\text{UKF}}^2 (n + \kappa_{\text{UKF}}) - n, \tag{28}$$

where parameters α_{UKF} and κ_{UKF} define how much the sigma points are spread.

Assuming that the state transition noise is zero mean additive normal with covariance $W_{(t)}$, the prior mean and covariance computed using sigma points are

$$\mu_{(t)}^- = \sum_{i=0}^{2n} w_i^s f_{(t)}(\chi_i) \tag{29}$$

$$P_{(t)}^- = W_{(t)} + \sum_{i=0}^{2n} w_i^c \left[f_{(t)}(\chi_i) - \mu_{(t)}^- \right] \left[f_{(t)}(\chi_i) - \mu_{(t)}^- \right]^T, \tag{30}$$

where the sigma points are computed from the posterior of the previous time step using (26) and the weights are

$$w_0^s = \frac{\xi_{\text{UKF}}}{n + \xi_{\text{UKF}}} \tag{31}$$

$$w_0^c = \frac{\xi_{\text{UKF}}}{n + \xi_{\text{UKF}}} + (1 - \alpha_{\text{UKF}}^2 + \beta_{\text{UKF}}) \tag{32}$$

$$w_i^s = w_i^c = \frac{1}{2n + 2\xi_{\text{UKF}}} \qquad , i > 0. \tag{33}$$

The parameter β_{UKF} is related to the distribution of the state. In the case of Gaussian distribution $\beta_{\text{UKF}} = 2$ is optimal. [76]

In the update step of the UKF the predicted measurement value and innovation covariance are

$$y_{(t)}^- = \sum_{i=0}^{2n} w_i^s h_{(t)}(\chi_i) \tag{34}$$

$$S_{(t)} = R_{(t)} + P_{yy} = R_{(t)} + \sum_{i=0}^{2n} w_i^c \left[h_{(t)}(\chi_i) - y_{(t)}^- \right] \left[h_{(t)}(\chi_i) - y_{(t)}^- \right]^T. \quad (35)$$

The sigma points for an update can be computed from a prior or the propagated sigma points from the prior computation can be used.

The cross covariance matrix for the state and measurement is

$$P_{xy} = \sum_{i=0}^{2n} w_i^c \left[\chi_i - \mu_{(t)}^- \right] \left[h_{(t)}(\chi_i) - y_{(t)}^- \right]^T \quad (36)$$

and the UKF Kalman gain is

$$K_{(t)} = P_{xy} S_{(t)}^{-1}. \quad (37)$$

The updated estimate mean and covariance are

$$\mu_{(t)} = \mu_{(t)}^- + K_{(t)}(y_{(t)} - y_{(t)}^-) \quad (38)$$

$$P_{(t)} = P_{(t)}^- - K_{(t)} S_{(t)} K_{(t)}^T. \quad (39)$$

Compared to the EKF, the UKF uses information of the state transition and measurement functions also in the neighborhood of the mean, whereas EKF linearization is based only on the linearization at the mean.

Among other KF extensions, the cubature Kalman Filters are similar to the UKF in the sense that they also use a set of sigma points, but differ in the use of weights and the background theory [5]. The second order EKF uses the second order Taylor approximation to take also the second order terms into account. There is also a numerical method for computing the second order EKF update [65]. Nonlinear functions can be also statistically linearized [23]. While the statistically linearized KF has a good estimation accuracy, its major disadvantage is the need of closed form formulas for certain expected values. These expected values can be computed only in some simple special cases [73].

Gaussian Mixture Filter

All filters that are based on only one normal (or other unimodal) distribution are unable to estimate multimodal posteriors well e.g. the

situation in Figure 1. To overcome this problem the Gaussian Mixture Filters (GMFs) use sums of normal distributions to approximate a probability distribution [71].

The pdf of a Gaussian Mixture (GM) is

$$p_{\mathrm{GMF}}(x) = \sum_{i=1}^{m} w_i p_{\mathrm{N}}(x|\mu_i, P_i), \tag{40}$$

where m is the number of components and w_i, μ_i and P_i are the weight, mean and covariance of the ith component. The component weights are non-negative and normalized so that

$$\sum_{i=1}^{m} w_i = 1. \tag{41}$$

The mean of the whole mixture is

$$\mu = \sum_{i=1}^{m} w_i \mu_i \tag{42}$$

and the covariance is

$$P = \sum_{i=1}^{m} w_i \left[P_i + (\mu_i - \mu)(\mu_i - \mu)^T \right]. \tag{43}$$

This thesis considers probably the most used variant of the GMF, where the prediction and update steps for each component mean and covariance are done with a KF extension and the weights are updated in the update step as

$$w_i^+ = w_i p_{\mathrm{N}}(y|y_i^-, S_i) \tag{44}$$

and then normalized to satisfy (41). The components can be split into smaller components before the update and merged again into smaller number of components after the update step.

Figure 2 shows the same update as in Figure 1 done with a 3-component GMF. In the figure, the red dashed GMF line is the weighted sum of the components. The GMF prior is almost, but

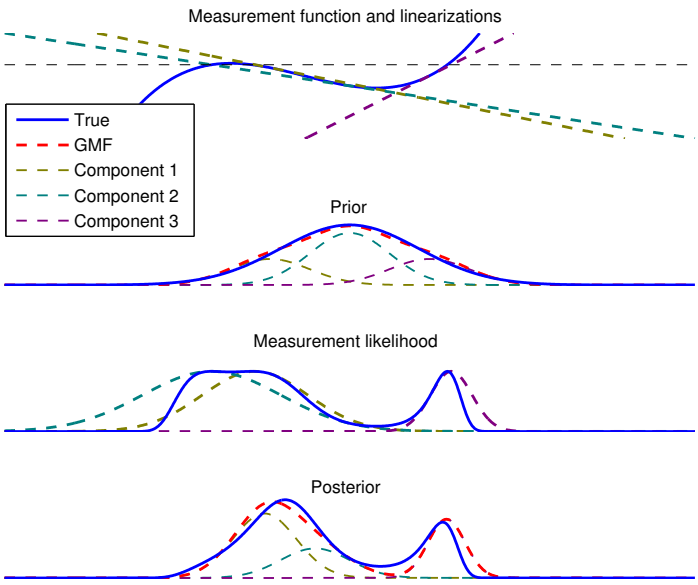

Figure 2: Update of a prior using the GMF. The top plot shows the measurement function and its linearization for each GM component. The second plot shows the prior pdf and the GM approximation of it. In the third plot the true measurement likelihood and linearized likelihoods of the GM components are shown and in the bottom plot the resulting posteriors are shown.

not exactly, Gaussian. In this example, the components are updated using the EKF. Linearized likelihoods are computed separately for each GMF component. Two of the likelihoods have the most likely region on the larger likelihood peak on the left, but the third is sharper and located on the smaller peak on the right. The posterior estimate produced with the 3-component GMF is much closer to the true posterior than the single component EKF estimate in Figure 1.

The main problem with the GMFs is how to select the number of components so that the estimate is good enough while keeping the computation feasible. If the system is close to linear, fewer components are needed, but, if there are several nonlinearities, more

components are needed to achieve a good approximation.

Section 2 and publications [P2] and [P7] discuss how to compute the amount of nonlinearity and split the prior so that the nonlinearity is reduced to an acceptable level while keeping the number of components low. Publication [P2] presents an algorithm for finding the direction of maximum nonlinearity and doing the split in that direction. In [P7] this is extended so that the splitting can be done in multiple directions at once. The splitting is usually done in such a way that the mean (42) and covariance (43) are preserved. In [1], a mixture splitting algorithm that produces a mixture whose cumulative distribution function (cdf) converges to the prior cdf is presented. In [P7] the Binomial Gaussian Mixture (BinoGM) is presented, which has convergence of pdf as well as of cdf.

The number of components can be reduced by removing components with low weights or by merging similar components so that the mixture mean and covariance are preserved. In [P7] the merging is done recursively using the algorithm presented in [66]. The merging methods are not discussed more deeply in this thesis. A review of merging algorithms can be found in [14].

In [P1] a GMF is applied in localization of a GSM BS using TA measurements. A TA measurement defines roughly the range from a BS to the mobile phone. The initial estimate of a BS location is constructed as a GM whose components form a circular pdf around the GPS location of the mobile phone. The components are updated using the EKF. Because the BS is assumed static, the components are not split. In [P1] the number components is reduced when the weight of a component becomes low or two components have similar mean.

There are also other variants of GMFs. Instead of using a linearized measurement, the measurement can be approximated as a mixture of measurements. A component this kind of mixture is

$$y_j = h_j(x) + \varepsilon_{y,j}, \tag{45}$$

where $\varepsilon_{y,,j}$ is a Gaussian noise term. For every measurement component there is associated weight $\sum_j w_{y,j}$. Sum of weights is $\sum_j w_{y,j} = 1$

and $w_{y,j} \geq 0$. Updates are done with a KF extension for each pair of prior and measurement components. The number of components after the update is the product of components in prior and components in measurement. An updated weight for a component is

$$w_{i,j}^{+} \propto w_i w_{y,j} p_{\mathrm{N}}(y | y_{i,j}^{-}, S_{i,j}) \tag{46}$$

One use case for this kind of measurement models is modeling a non-Gaussian noise. In [11], a two-component GM is used to model measurement noise under multipath conditions. The first component has a zero bias and the second component that approximates the possible multipath has a positive bias. Another use case for this kind of GMs is approximating the likelihood of a non-linear measurement function with multiple linear-Gaussian components.

This kind of measurement model could be applied in the GSM BS positioning in [P1]. Every measurement could be approximated as a similar GM as was used in forming of the prior and computing the posterior using a set of linear measurements instead of updating the components with the EKF. The number of components would grow very fast in this application. If a measurement is approximated using 8 components then the number of components after only 6 measurements is 262144 if no component reduction is applied.

In the Generalized Gaussian Mixture Filter (GGMF) the measurement likelihood is also modeled using a GM [50, 49]. The GGMF allows the component weights to be negative. This allows the modeling of ring shaped distributions using only 2 components. These models cannot approximate very accurate measurements. This method could also be applied in the situation of [P1].

Figure 3 shows how the GMFs mentioned above update a range measurement similar to TA measurements used in [P1]. In the first row the exact prior, measurement likelihood, and posterior are shown. In the second row the prior is initialized using a 6-component GM and it is updated using the EKF. The third row uses a similar GM for the update as was used as the prior. This generates posterior with 36 components. Compared to the first GM posterior, this posterior is symmetric as is the true posterior. In the last row the computation

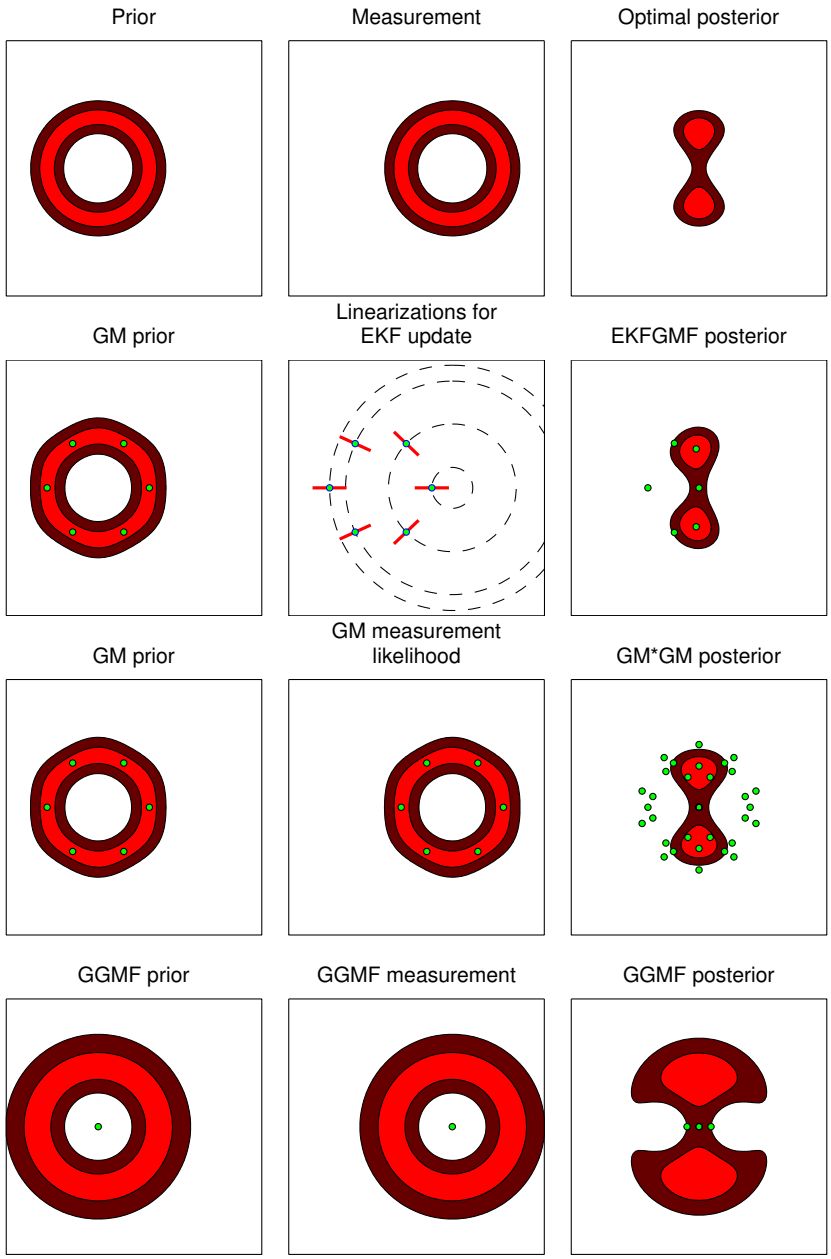

Figure 3: An update of a range measurement prior using different types of GMFs. The green dots show the component means.

of posterior is done using the GGMF. The GGMF estimate has only 4 components, but the posterior has larger variance than the true posterior or other GM estimates.

In [28], the GMF update is done using a progressive scheme, where the progression starts from an update that can be analytically solved and ends to the true update as a progression parameter approaches 1. The update is formulated as a differential equation and components are split while solving the differential equation, if necessary.

Usually and in the situations that are more thoroughly investigated in this thesis, the component weights are updated only in the update phase. It is also possible to update them in the prediction phase as is done in [74].

Particle Filter

Particle Filters (PFs) are Monte Carlo methods that use point masses to approximate the probability distribution of the state. In the literature there exist several different variants of the PF [62]. The PFs are particularly useful with nonlinear and non-Gaussian problems [15]. Increasing the number of particles makes the particle approximation closer to optimal. In [13], a survey of convergence results of the PFs can be found.

The pdf of a PF can be written as

$$p(x) = \sum_{i=1}^{m} w_i \delta(\mu_i - x), \tag{47}$$

where δ is the Dirac delta function, m is the number of particles, μ_i is the location of ith particle and weights w_i ($w_i \geq 0$) sum to one. Here I present a variant called Bootstrap Particle Filter [25]. In the prediction step the point masses are moved according to the state transition model

$$\mu_{i,(t)} = f(\mu_{i,(t-1)}, \epsilon_i), \tag{48}$$

where $f(\mu_{i,(t-1)}, \epsilon_i)$ is a general state transition function that does not require state transition error to be additive and ϵ_i is a sample from the distribution of state transition model noise. In the update step

particles are reweighed according to the measurement likelihood

$$w_{i,(t)} \propto w_{i,(t-1)} p(y_{(t)}|\mu_{i,(t)}) \tag{49}$$

and then normalized.

After some measurements, the weights of particles tend to concentrate on a few particles only. A few particles cannot usually approximate the true distribution well. To address this problem, resampling is done. Resampling can be done at every time instance or when the weight is concentrated on a small portion of particles. In resampling a new set of particles is drawn from the distribution of the old particles. All the new particles have equal weights.

A pseudocode of so-called systematic resampling (aka stratified resampling) is shown in Algorithm 1. In systematic resampling a new set of particles is selected from the old ones in such a way that there is at least one new particle at the location of every old particle having $w \geq \frac{1}{m}$[10]. In [16], a review of different resampling algorithms and their properties is presented.

Draw $\omega \sim \mathrm{U}((0, \frac{1}{m}])$ `// Initial reference weight`
$i \leftarrow 1$ `// Index for new particle locations`
$j \leftarrow 1$ `// Index for old particle locations`
while $i \leq m$ **do**
  `// If the cumulative particle weight from 1st to` k`th particles is more than reference weight, add` k`th particle to the resampled set of particles`
  if $\sum_{k=1}^{i} w_k \geq \omega$ **then**
    $\hat{\mu}_j \leftarrow \mu_i$ `// New particle location`
    $\hat{w}_j \leftarrow \frac{1}{m}$ `// New particle weight`
    $\omega \leftarrow \omega + \frac{1}{m}$ `// Increase reference weight`
    $j \leftarrow j + 1$
  else
    $i \leftarrow i + 1$
  end
end

Algorithm 1: Systematic resampling

Figure 4 shows the same update example as in Figures 1 and 2 done with a PF that uses 20 particles. Without resampling the posterior would be the same as the particles updated with the likelihood function. In the particle cloud with resampled particles, more particles are concentrated on the positions where the particle weights were high. In this phase the non-resampled particles represent the true posterior better, but after the prediction step the PF without resampling has many low-weight particles at the area of the local minimum in the center, whereas the resampled PF has more particles at the peaks of the true posterior.

The problems with the PFs are similar to problems with the GMFs: how to choose the number of particles so that the estimate is good but without wasting computational resources. In [22], the number of particles is adapted by dividing space into bins and adapting the number of particles depending on how particles occupy bins. The division of space into bins works in practice only in low dimensional situations. In method proposed in [72] the amount of particles is based on making a normal approximation of the particle cloud and computing expected mean error. This approximation may be poor when the estimated distribution is not normal distributed.

In [58], a rule for the number of particles in the initialization of a PF by sampling particles from a normal distribution was given. If the required number of particles m_1 is known for covariance P_1 e.g. from previous experiments, then the required number of particles for P_2 is

$$m_2 = m_1 \sqrt{\frac{\det P_2}{\det P_1}}. \tag{50}$$

For the bootstrap PF the required number of particles is very high in situations where the state has many dimensions, the state transition model (48) has a lot of noise and the measurements are accurate. For this kind of situations there exist PFs that take the measurements into account already in the prediction phase. These filters are not as simple as bootstrap PF and they require more computational resources for a single particle, but require fewer particles for a good approximation. This kind of PFs are, for example, Unscented Particle Filter [48] and Auxiliary Particle Filter [55].

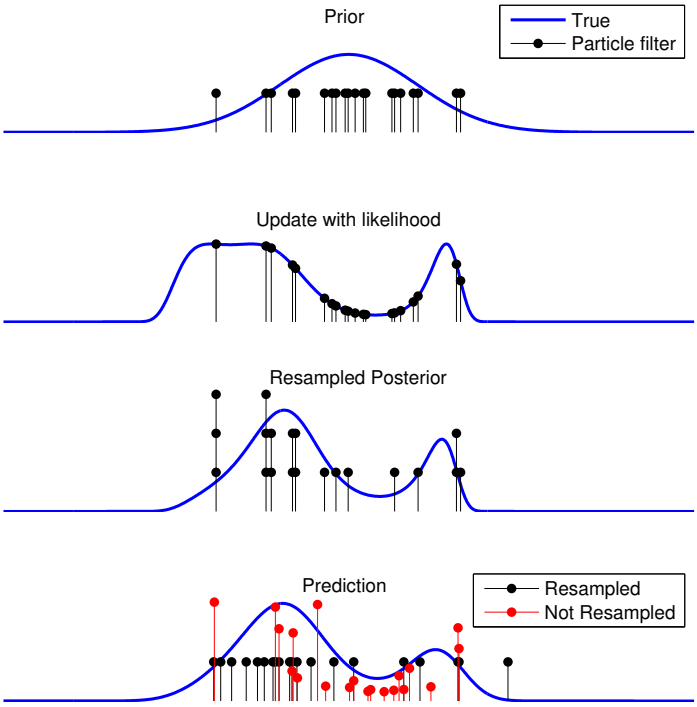

Figure 4: Update of a prior using a PF. In the first plot the prior is approximated with a cloud of equally weighted particles, whose density is proportional to the probability density of prior. In second plot the particle weights are updated with the measurement likelihood. In the third plot the particles are resampled so that all particles have equal weights, but there are more particles located where the most probable particles were. The last plot shows the state evolution of particles after one prediction step with and without resampling. Particles that were not resampled are more concentrated on less likely regions, while the resampled particles are better distributed on the likely areas of the true distribution.

Rao-Blackwellized Particle Filter [17] can be used when a part of the state is linear-Gaussian and a part is nonlinear. By using a KF for the linear-Gaussian part the dimension of the nonlinear part can be reduced and, thus, the required number of particles.

2 Measuring nonlinearity and using nonlinearity information in Gaussian Mixture Filters

When a measurement model is almost linear and the measurement error is additive and normally distributed, the Extended Kalman Filter (EKF), the Unscented Kalman Filter (UKF), or other Kalman Filter (KF) extensions can be used to update a Gaussian prior without causing much error. In situations where nonlinearity is high the linearization errors involved in the update are large. A question is how to determine when the nonlinearity is high. If the nonlinearity is high, the estimation can be done with a Particle Filter (PF) or, in the case of the Gaussian Mixture Filter (GMF), the original Gaussian can be split into smaller Gaussian components until the nonlinearity within every component is at an acceptable level. In the GMFs a second question is how to use the nonlinearity information to perform the splitting in an efficient way.

In the following subsections, ways of measuring the amount of nonlinearity, finding the direction of nonlinearity, and using the nonlinearity measures for splitting Gaussian components are presented. The discussion will consider only the nonlinearity of measurement functions, but the methods can be also adapted for nonlinear state transition models. In the following, the measurement error is assumed to be additive and normally distributed.

2.1 Amount of nonlinearity

The purpose of measuring the amount of nonlinearity is to know whether the posterior estimate provided by a KF extension is good. The most straightforward way to do this would be a direct comparison of the true posterior and the approximated posterior. One measure for comparing the difference of probability density functions (pdfs) $p(x)$ and $q(x)$ is the Kullback-Leibler divergence [44]

$$\eta^{\mathrm{KL}} = \int \ln\left(\frac{p(x)}{q(x)}\right) p(x)\mathrm{d}x. \tag{51}$$

When $p(x)$ is the pdf of the true posterior and $q(x)$ is the pdf of the approximated posterior estimate, the Kullback-Leibler divergence can be interpreted as "information lost when q is used to approximate p" [9].

The problem with the Kullback-Leibler divergence is that an analytic solution exists only in special cases. Numerical computation of the integral requires numerical computation of the true posterior and, after the numerical estimate of the true posterior is computed, there is no need of a Gaussian estimate in online estimation. The computation of the true posterior numerically may require a lot of computational resources. Of course, the Kullback-Leibler divergence can be used offline to test the quality of estimates.

Next I will present approximate methods for evaluating the amount of nonlinearity that should be easier to compute than the Kullback-Leibler divergence. At the end of this subsection, the presented methods will be compared by testing how often they agree with the Kullback-Leibler divergence when comparing which one of two updates is more difficult for the EKF.

In [61], a modified version of the Kullback-Leibler divergence (51) is proposed

$$\eta^{\mathrm{R\&H}} = \int \left[\ln\left(\frac{q(x)}{p(x)} \right) \right]^{2i} q(x) \mathrm{d}x, \tag{52}$$

where i is an integer (only value $i = 1$ is used in [61]), $p(x)$ is the pdf of the true posterior and $q(x)$ is the pdf of the normal approximation computed with the EKF. This modification is done to make analytical solutions available for a wider range of measurement functions. For measurement functions that are polynomial or trigonometric involving sine and cosine the measure (52) can be expressed in terms of moments of the updated Gaussian $q(x)$ [61]. Still this concerns only a part of measurement functions and finding an analytical expression for the nonlinearity measure may require a lot of work and numerical integration of (52) is numerically similar problem to computing a numerical estimate of the true posterior similarly as with computing the Kullback-Leibler divergence.

In [30], Havlak and Campbell use the same sigma points as the UKF

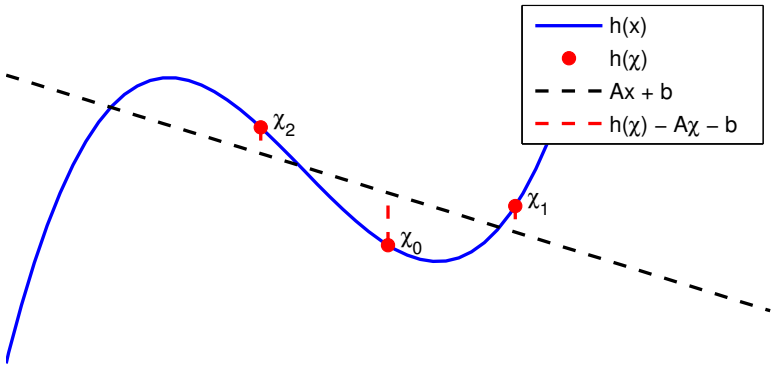

Figure 5: Variables used in the computation of the amount of nonlinearity according to $\eta^{\text{H\&C}}$ measure

in (26) in determining the degree of nonlinearity of a nonlinear state transition model. Here the method is applied to nonlinear measurement functions. The degree of nonlinearity $\eta_i^{\text{H\&C}}$ corresponding to the ith sigma point χ_i is

$$\eta_i^{\text{H\&C}} = \left\| h(\chi_i) - A\chi_i - b \right\|, \tag{53}$$

where A and b are parameters of the linear model that is the least squares fit to the measurement function at the sigma points, that is,

$$\arg\min_{A,b} \sum_{i=0}^{2n} \left\| h(\chi_i) - A\chi_i - b \right\|^2. \tag{54}$$

To determine whether a situation is nonlinear, the maximal $\eta_i^{\text{H\&C}}$ is compared to a threshold value. A least squares fit and used variables are shown in Figure 5. The used nonlinear function is $h(x) = \frac{x^3}{2} + x^2 - x$ (25).

In [20], Faubel, McDonough, and Klakow propose a nonlinearity measure that is computed also at UKF sigma points:

$$\eta^{\text{SM}} = \sum_{i=1}^{n} \frac{\left\| h(\chi_i) + h(\chi_{i+n}) - 2h(\chi_0) \right\|^2}{\left\| h(\chi_i) - h(\chi_{i+n}) \right\|^2}. \tag{55}$$

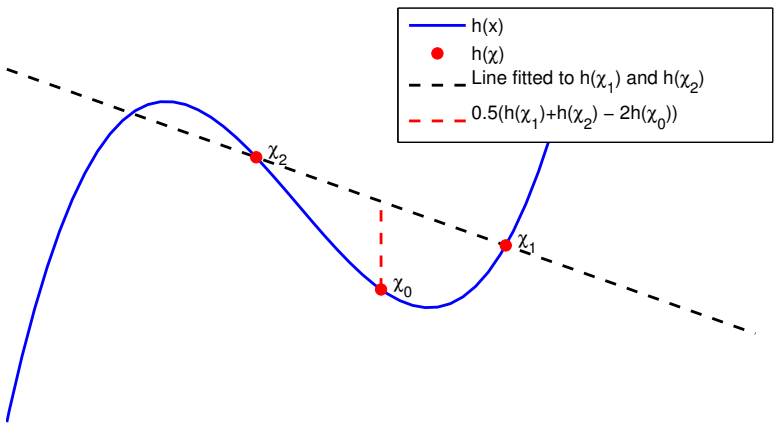

Figure 6: Using a line fitted to $h(\chi_1)$ and $h(\chi_2)$ to compute nonlinearity at $h(\chi_0)$ for (56).

In this nonlinearity measure the division by $\left\|h(\chi_i)-h(\chi_{i+n})\right\|^2$ may cause a division by zero.

Later in [19], Faubel and Klakow use a nonlinearity measure, which is almost the same, but does no longer have the risk of a division by zero. In this work, the amount of nonlinearity corresponding to each sigma point pair is

$$\eta_i^{\mathrm{F\&K}} = \frac{\left\|h(\chi_i)+h(\chi_{i+n})-2h(\chi_0)\right\|^2}{2}. \tag{56}$$

The total amount of nonlinearity, which is compared to a threshold value to determine whether the situation is nonlinear, is

$$\eta_{\mathrm{TOT}}^{\mathrm{F\&K}} = \frac{\sum_{i=1}^{n}\eta_i^{\mathrm{F\&K}}}{n}. \tag{57}$$

The nominator in η^{SM} (55) and $\eta^{\mathrm{F\&K}}$ (56) can be interpreted as fitting a line to go through sigma points χ_i and χ_{i+n} and then comparing how much its residual is at mean χ_0. This interpretation is illustrated in Figure 6

In [35], the nonlinearity criterion is based on the statistical linear regression of the measurement. The covariance matrix of the statistical

linearization error is

$$P_\varepsilon = P_{yy} - P_{xy}^T P_{xx}^{-1} P_{xy}, \tag{58}$$

where the matrices come from the covariances of the augmented state that contains the prior state x^- and the predicted measurement $h(x^-)$

$$\operatorname{cov}\begin{bmatrix} x^- \\ h(x^-) \end{bmatrix} = \begin{bmatrix} P_{xx} & P_{xy} \\ P_{xy}^T & P_{yy} \end{bmatrix}. \tag{59}$$

The amount of nonlinearity corresponding to a Gaussian prior is computed as

$$\eta^{\mathrm{H}} = \operatorname{tr} P_\varepsilon \tag{60}$$

In the general case, these covariances cannot be analytically computed, but they may be approximated numerically. One possibility presented in the paper [35] is to compute these matrices using same unscented transform that is used in the UKF. The prior covariance P_{xx} is known and other matrices can be approximated using the equations (35) and (36).

For a second order polynomial the statistical linearization can be made analytically and the covariance of the statistical linearization error is

$$P_\varepsilon = \frac{1}{2}\begin{bmatrix} \operatorname{tr} PH_1PH_1 & \operatorname{tr} PH_1PH_2 & \dots \\ \operatorname{tr} PH_2PH_1 & \operatorname{tr} PH_2PH_2 & \dots \\ \vdots & \vdots & \ddots \end{bmatrix}, \tag{61}$$

where H_i is the Hessian of the ith component of the measurement function evaluated at the prior mean. The nonlinearity measure η^{H} for second order polynomials is

$$\frac{1}{2}\sum_{i=1}^{n} \operatorname{tr} PH_iPH_i. \tag{62}$$

In [37], was proposed that for scalar measurements the measurement function is considered nonlinear when

$$\operatorname{tr} PHPH > R, \tag{63}$$

which is similar to (62), but the nonlinearity depends also on the measurement variance R. In [3], this criterion was used for multidimensional measurements in the form

$$\frac{\sqrt{\operatorname{tr} PH_iPH_i}}{\sqrt{R_{[i,i]}}} > 1, \text{ for some } i. \tag{64}$$

Because the measure (64) uses only diagonal components of the measurement covariance R, the nonlinearity values may be inconsistent if the components of the measurement vector are correlated. In [P7], this problem is solved by decorrelating the measurements by applying a linear discrete Karhunen-Loève transformation

$$\hat{y} = L(R)^{-1} y = L(R)^{-1} h(x) + L(R)^{-1} \varepsilon_y. \tag{65}$$

The covariance of $\hat{y}$ is an identity matrix and thus the transformed measurements are independent and can be treated separately as scalar measurements.

An example where (64) without decorrelation gives a poor estimate of the nonlinearity has identity prior covariance matrix and measurement

$$y = h(x) = \begin{bmatrix} 100x_{[1]} \\ 100x_{[1]} + x_{[2]}^2 \end{bmatrix} + \varepsilon_y, \varepsilon_y \sim \mathrm{N}\left(0, \begin{bmatrix} 10000 & 10000 \\ 10000 & 10001 \end{bmatrix}\right) \tag{66}$$

These measurements are almost linear according to (64)

$$\frac{\sqrt{\operatorname{tr} PH_1PH_1}}{\sqrt{R_{[1,1]}}} = 0 \tag{67}$$

$$\frac{\sqrt{\operatorname{tr} PH_2PH_2}}{\sqrt{R_{[2,2]}}} \approx 0.02. \tag{68}$$

After decorrelating with $\begin{bmatrix} 100 & 0 \\ 100 & 1 \end{bmatrix}^{-1}$ the measurement is

$$\hat{y} = \begin{bmatrix} x_{[1]} \\ x_{[2]}^2 \end{bmatrix} + \varepsilon_y, \varepsilon_y \sim \mathrm{N}\left(0, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) \tag{69}$$

and the amounts of nonlinearities are

$$\frac{\sqrt{\operatorname{tr} PH_1PH_1}}{\sqrt{R_{[1,1]}}} = 0 \tag{70}$$

$$\frac{\sqrt{\operatorname{tr} PH_2PH_2}}{\sqrt{R_{[2,2]}}} = 2 > 1 \tag{71}$$

and the second element of the decorrelated measurement function is considered highly nonlinear. Before decorrelation a large amount of correlated Gaussian noise affects both measurements. In (64) the cross terms are not taken into account and the diagonal elements of the covariance matrix have so large values that the nonlinearity is deemed low, but when the measurements are decorrelated it can be seen that there is a significant amount of nonlinearity.

In the decorrelation the choice of $L(R)$ has an effect on the individual nonlinearity values. However, the sum of squared decorrelated nonlinearities is independent of the choice of $L(R)$, because the trace term of the ith element of the decorrelated d-dimensional measurement is

$$\operatorname{tr} P\left(\sum_{i=1}^{d} L(R)^{-1}_{[j,i]} H_i\right) P\left(\sum_{i=1}^{d} L(R)^{-1}_{[j,i]} H_i\right) \tag{72}$$

and the sum can be written as a new nonlinearity measure η^{R}:

$$\begin{aligned}
\eta^{\mathrm{R}} &= \sum_{j=1}^{d} \operatorname{tr} P\left(\sum_{i=1}^{d} L(R)^{-1}_{[j,i]} H_i\right) P\left(\sum_{i=1}^{d} L(R)^{-1}_{[j,i]} H_i\right) \\
&= \sum_{j=1}^{d} \operatorname{tr} \sum_{i=1}^{d}\sum_{k=1}^{d} L(R)^{-1}_{[j,i]} L(R)^{-1}_{[j,k]} PH_iPH_k \\
&= \operatorname{tr} \sum_{i=1}^{d}\sum_{k=1}^{d} R^{-1}_{[k,i]} PH_iPH_k.
\end{aligned} \tag{73}$$

Due to this, the larger nonlinearity value for decorrelated measurement components of (66) is in the range from 1 to 2 depending on choice of $L(R)$.

In [P2], a squared version of (64) is used and a numerical approximation for computing $\operatorname{tr} PHPH$ is given. Matrix Q whose elements

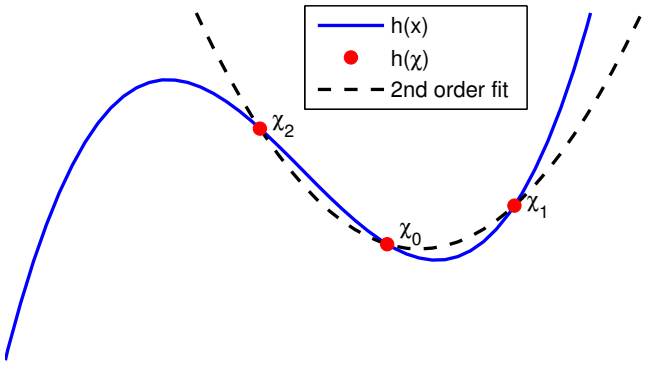

Figure 7: Second order polynomial fit to sigma-points

are

$$Q_{[i,j]} = \begin{cases} \left[h(\mu+\Delta_i)+h(\mu-\Delta_i)-2h(\mu)\right] & ,i=j \\ \Big[h(\mu+\Delta_i+\Delta_j)+h(\mu-\Delta_i-\Delta_j)- \\ \quad 2h(\mu)-Q_{[i,i]}-Q_{[j,j]}\Big] & ,i\neq j \end{cases}, \tag{74}$$

where $\Delta_i = \gamma L(P)_{[:,i]}$ and γ determines the spread of the evaluation points, can be used as an approximation

$$Q \approx \gamma^2 L(P)^T H L(P). \tag{75}$$

This approximation is exact when the measurement function is a second order polynomial [P2]. The trace can now be approximated as

$$\operatorname{tr} PHPH \approx \operatorname{tr} \frac{QQ}{\gamma^4}. \tag{76}$$

This can naturally be used also with multidimensional nonlinearity measure (73)

$$\operatorname{tr} PH_iPH_j \approx \operatorname{tr} \frac{Q_iQ_j}{\gamma^4}, \tag{77}$$

To reuse measurement function evaluations in the UKF, γ can be chosen to be $\sqrt{n+\xi_{\mathrm{UKF}}}$.

Figure 7 shows a second order polynomial fit to sigma-points. The numerical computation of nonlinearity η^{R} using (74) uses this polynomial approximation.

The measures (76) and (56) are related: $Q_{[i,i]}^2 = 2\eta_i^{\mathrm{F\&K}}$ and when Q is diagonal

$$\operatorname{tr} QQ = 2n\eta_{\mathrm{TOT}}^{\mathrm{F\&K}}. \tag{78}$$

This means that $\eta_{\mathrm{TOT}}^{\mathrm{F\&K}}$ does not take the non-diagonal terms of Q into account and is not normalized in the sense of the sigma point spread.

The nonlinearity measure η^{R} has the following benefits over η^{H}, $\eta^{\mathrm{F\&K}}$ and $\eta^{H\&C}$:

- The result is independent of chosen sigma points. In both $\eta^{\mathrm{F\&K}}$ and $\eta^{\mathrm{H\&C}}$ the used matrix square root $L(P)$ and the sigma point spread affect the results. When using (76) to approximate $\operatorname{tr} PHPH$ there may be some variation based on the chosen sigma points.
- The result is independent of used units and the measure is dimensionless
- The effect of measurement noise level is taken into account

Because of these properties the nonlinearity measure η^{R} can be used with an absolute nonlinearity threshold e.g. 1 in (64) while thresholds for η^{H},$\eta^{\mathrm{F\&K}}$ and $\eta^{\mathrm{H\&C}}$ are case dependent. The drawback of measures based on $\frac{\operatorname{tr} PH_iPH_i}{R_{[i,i]}}$ is that the H_i has to either be analytically computed or, if it is approximated numerically with (74), the number of required measurement function evaluations is increased from $2n+1$[1] to $\frac{n^2+n}{2}+1$.

The reason for taking the measurement covariance, R, into account is that the nonlinearity affects more when measurements are more accurate and then also the measure becomes independent of used units. An example of the effect of measurement variance is shown in Figure 8. The figure shows contours of exact posterior distributions compared to normal distributions having the same mean and covariance. The measurement function used in the figure is

$$h(x) = x_{[1]}^2 - x_{[2]} + \varepsilon_y \tag{79}$$

with realized value 0 and normal measurement noise with variance R. The prior is zero mean normal with covariance P. Because the

[1] η^{H} can be computed also with other number of measurement function evaluations

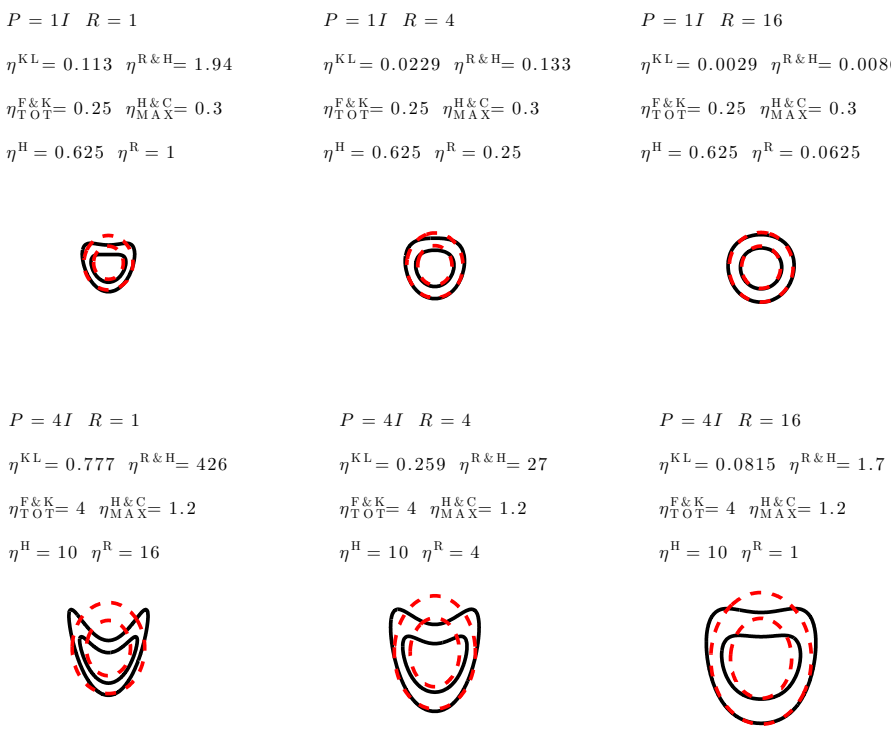

Figure 8: Posteriors with different prior and measurement covariances and the amount of nonlinearity computed with different methods presented in this section.

measurement function is a second order polynomial, numerical computation of η^{R} (76) would produce exact results. The black lines show contours that contain 1/3 and 2/3 of the exact posterior probability. The dashed red lines show corresponding lines for a normal distribution having the same mean and covariance as the true posterior. The figure shows how the true posterior is closest to the normal distribution when the prior covariance is small and measurement covariance is large and also then the nonlinearity η^{R} is smallest. The Kullback-Leibler divergence η^{KL} and $\eta^{\mathrm{R\&H}}$ give similar orders of the amounts of nonlinearity as η^{R}, except that they consider the last plot slightly less nonlinear than the first plot.

The values of nonlinearity measures η^{H},$\eta^{\mathrm{F\&K}}_{\mathrm{TOT}}$, $\eta^{\mathrm{H\&C}}_{\mathrm{MAX}}$ are affected by the prior covariance but are not affected by the measurement covariance and the values will change if the sigma points are selected in some other way. Both $\eta^{\mathrm{F\&K}}$ and $\eta^{\mathrm{H\&C}}$ could be improved by using Mahalanobis distance with covariance R instead of the Euclidean norm. Similarly η^{H} could be enhanced by using R to normalize it as in (73).

The numerical methods that use UKF sigma points or the second order polynomial approximation may fail in some special cases. For example, consider a one-dimensional prior state with zero mean and a measurement function

$$h(x) = x^3. \tag{80}$$

When this measurement function is evaluated at $\mu^- = 0$, the Hessian is 0 and so $\mathrm{tr}\, PHPH = 0$. The symmetric sigma points are: $-\chi$, 0 and χ. Because $-\chi^3+\chi^3-2\cdot 0^3 = 0$ both $\eta^{\mathrm{F\&K}}$ (56) and η^{R}_Q computed using (74) indicate zero nonlinearity. The linear model (54) fits perfectly to data points ($A = \chi^2, b = 0$) and so also (53) indicates zero nonlinearity. Nonlinearity measure η^{H} (60) could detect nonlinearity depending on the points used for the statistical linearization.

Next the different nonlinearity measures are tested by computing the nonlinearity estimates when varying measurements are applied to varying priors. The measured amounts of nonlinearity in different test situations are compared pairwise and then compared if their order agrees with the Kullback-Leibler divergence η^{KL} (51) of an EKF update compared to the true posterior. To evaluate this I compute the following measure

$$\frac{\sum_{i,j} \left| \mathrm{sign}\left(\eta_i - \eta_j\right) - \mathrm{sign}\left(\eta_i^{\mathrm{KL}} - \eta_j^{\mathrm{KL}}\right) \right|}{\sum_{i,j} 2}, \tag{81}$$

where η_i and η_j are the amounts of nonlinearity in the ith and jth case using the evaluated nonlinearity measure and η_i^{KL} and η_j^{KL} are the amounts of nonlinearity according to the Kullback-Leibler divergence. The two measures agree, if the signs of the differences are the same. If a nonlinearity measure gives same nonlinearity value for

two situations, i.e. it is indecisive, it is considered to be half right by the measure (81).

Because η^{KL} is not symmetric it is also tested with the distributions exchanged. This is denoted with $\overline{\eta^{\mathrm{KL}}}$. Nonlinearity values for η^{KL}, $\overline{\eta^{\mathrm{KL}}}$, and $\eta^{\mathrm{R\&H}}$ are computed using numerical integration. η^{R} is tested with analytical Hessians and with numerical computation of (76), which is denoted with η^{R}_{Q}. η^{H} is computed using UKF sigma points.

In the test setup the prior was updated using two range measurements from two Base Stations (BSs)

$$y = \|x - r_{\mathrm{BS}}\| + \varepsilon_y, \tag{82}$$

where r_{BS} is the location of a BS and x contains two position variables. Five parameters are varied between three possible values and all the combinations are gone through. This results in $3^5 = 243$ different combinations and 59049 nonlinearity value comparisons. The parameters and their values were:

- Prior means: $\begin{bmatrix}-2\\0\end{bmatrix}, \begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}2\\0\end{bmatrix}$
- Prior covariances: $\begin{bmatrix}1&0\\0&1\end{bmatrix}, \begin{bmatrix}4&0\\0&4\end{bmatrix}, \begin{bmatrix}4&-1\\-1&1\end{bmatrix}$
- BS locations: $\left(\begin{bmatrix}-4\\0\end{bmatrix}, \begin{bmatrix}0\\4\end{bmatrix}\right), \left(\begin{bmatrix}1\\1\end{bmatrix}, \begin{bmatrix}-1\\-1\end{bmatrix}\right), \left(\begin{bmatrix}3\\0.5\end{bmatrix}, \begin{bmatrix}2\\0.5\end{bmatrix}\right)$
- Measurement values: $\begin{bmatrix}5\\5\end{bmatrix}, \begin{bmatrix}1\\5\end{bmatrix}, \begin{bmatrix}2\\2\end{bmatrix}$
- Measurement covariances: $\begin{bmatrix}1&0\\0&1\end{bmatrix}, \begin{bmatrix}4&0\\0&4\end{bmatrix}, \begin{bmatrix}4&-1\\-1&1\end{bmatrix}$

In Table 1 the agreement of different nonlinearity measures with the Kullback-Leibler divergence is shown.

Among the tested nonlinearity measures (excluding $\overline{\eta^{\mathrm{KL}}}$), the measure $\eta^{\mathrm{R\&H}}$ agrees most often from the tested methods with the Kullback-Leibler divergence, but it cannot be analytically evaluated in every situation e.g. it is unclear if it has an analytical solution in the tested situations. It can also be noted that even though definition of $\eta^{\mathrm{R\&H}}$ (52) is almost same as $\overline{\eta^{\mathrm{KL}}}$ (51) it agrees more rarely with η^{KL}.

Measure	$\overline{\eta^{\mathrm{KL}}}$	$\eta^{\mathrm{R\&H}}$	η^{R}	η^{R}_{Q}	$\eta^{\mathrm{F\&K}}$	$\eta^{\mathrm{H\&C}}$	η^{H}
Agreement % with η^{KL}	89	84	77	78	70	70	71

Table 1: The agreement percentage of the evaluated nonlinearity measures with the Kullback-Leibler divergence on 59049 comparisons.

From the fast numerical measures η^{R} and η^{R}_{Q} give similar results that are better than results of $\eta^{\mathrm{F\&K}}$, $\eta^{\mathrm{H\&C}}$, and η^{H}, which are again quite similar.

2.2 Direction of the maximum nonlinearity

In addition to determining the amount of nonlinearity, the direction of nonlinearity can be estimated. This information is used later in Section 2.3 for constructing a Gaussian Mixture (GM) so that the number of components is higher and components are smaller in the direction of nonlinearity.

In both [30] and [19], a similar approach is used for finding the direction of the maximum nonlinearity. In [30], the direction of maximum nonlinearity is the eigenvector corresponding to the maximum eigenvalue of matrix C, which can be computed as

$$w_i = \frac{\eta_i^{\mathrm{H\&C}}}{\sum_{i=0}^{2n} \eta_i^{\mathrm{H\&C}}} \tag{83}$$

$$\mu = \sum_{i=0}^{2n} w_i \chi_i \tag{84}$$

$$C = \sum_{i=0}^{2n} w_i (\chi_i - \mu)(\chi_i - \mu)^T. \tag{85}$$

In [19], the amounts of nonlinearity are computed for sigma point pairs and the C matrix is

$$C = \sum_{i=1}^{n} \eta_i^{\mathrm{F\&K}} \frac{(\chi_i - \chi_0)(\chi_i - \chi_0)^T}{\left\|\chi_i - \chi_0\right\|^2}. \tag{86}$$

To avoid the computation of the eigenvector corresponding to the largest eigenvalue, [19] presents also an approximation for the direction of the maximum nonlinearity, which is

$$\frac{\sum_{i=1}^{n} \eta_i^{\mathrm{F\&K}} (\chi_i - \chi_0)}{\left\| \sum_{i=1}^{n} \eta_i^{\mathrm{F\&K}} (\chi_i - \chi_0) \right\|}. \tag{87}$$

In [35], the direction of the maximum nonlinearity is determined by evaluating for which eigenvector u_i of the prior covariance the associated nonlinearity $\tilde{\eta}_i^{\mathrm{H}}$

$$\tilde{\eta}_i^{\mathrm{H}} = \int \left\| h(x_v) - A(\mu + v\, u_i) - b \right\|^2 p_{\mathrm{N}}(\mu + v\, u_i, \mu, P) \mathrm{d}v \tag{88}$$

is largest. This can detect the nonlinearity only in the directions of the eigenvectors of the prior. Because this integral does not have analytic solution in general, numerical integration has to be used.

In [P2], the direction of the maximum nonlinearity is computed as the eigenvector of PH corresponding to the largest eigenvalue in magnitude. In a set of equally probable state vectors

$$(x-\mu)^T P^{-1} (x-\mu) = \text{constant} \tag{89}$$

the eigenvector of PH corresponding to the largest absolute eigenvalue maximizes $(x-\mu)^T H (x-\mu)$. This can be seen as finding a vector that maximizes the second order term of the Taylor series of the measurement function

$$h(x) \approx h(\mu) + J(x-\mu) + \frac{1}{2}(x-\mu)^T H (x-\mu) + \ldots. \tag{90}$$

The eigendecomposition

$$L(P)^T H L(P) \approx Q = U \Lambda U^T \tag{91}$$

can be used to numerically compute the direction vector of the maximum nonlinearity a as

$$a = L(P) U e_i, \tag{92}$$

	H&C [30]	F&K [19]	H [35]	AS [P2]
$\alpha_{\mathrm{UKF}} = 0.5$	22.3	22.3	22.3	$3.53 \cdot 10^{-6}$
$\alpha_{\mathrm{UKF}} = 10^{-3}$	22.3	22.3	22.3	0.762

Table 2: Average error on estimation of the direction of the maximum nonlinearity in degrees.

where i is the index of the largest eigenvalue in magnitude and e_i is the ith column of an identity matrix.

This direction of nonlinearity was introduced only for one-dimensional measurements and for multidimensional measurement each measurement element has to be treated separately. If the measurement errors are linearly dependent, decorrelation (65) should be applied before this.

In [P7], the different nonlinearity direction estimation methods were tested in a two-dimensional case where a scalar measurement function

$$h(x) = e^{b^T x}, \tag{93}$$

where b was randomly chosen, was evaluated with an isotropic prior. This measurement function has all of the nonlinearity in the direction of b, because the component of x that is perpendicular to b does not affect the measurement function at all. Here I do the same test and include also the direction computation proposed in [35]. The test is done 10000 times with random b and with 2 different values for $\alpha_{\mathrm{UKF}} = [0.5, 10^{-3}]$, which also affects the γ in (76). Table 2 shows the mean errors of the computed direction of nonlinearity. Methods H&C [30], F&K [19] and H [35] result in same error of to 22.3 degrees. This is close to choosing the main axis, which is closer to the direction of nonlinearity, as the direction of the maximum nonlinearity. The method proposed in [P2] resulted in an error of less than a degree, depending on the value of α_{UKF}.

Figure 9 shows an example of the determining of the direction of nonlinearity. The background shows the value of function (93). The circles are the used sigma points and the diamonds the additional points required for computation of Q. The nonlinearity direction

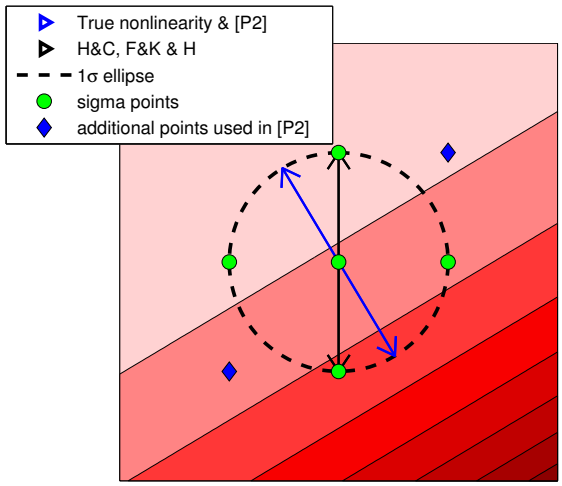

Figure 9: Different estimates of the direction of the maximum nonlinearity and the sigma points used to compute them

produced by the algorithm from [P2] and the true nonlinearity are so close to each other that they are drawn with a single arrow. Similarly estimates produced by algorithms from H&C, F&K and H produce identical estimates, which are aligned with a sigma point pair.

2.3 Use of nonlinearity information in Gaussian Mixture Filters

The amount of nonlinearity is used in GMFs to determine whether a component update with a linearized KF produces a good enough posterior estimate. If the nonlinearity is low, the update can be done with a KF extension, but if the nonlinearity is considered high, the prior component is split into smaller components.

Usually the amount of computational resources give some upper limit for the number of components and in many GMF variants also the component weight is used for choosing the next component to be split. In some GMF algorithms found in the literature, the split Gaussian mixture is generated without using the information of the direction of nonlinearity. If the measurements are linear in

some dimensions, this causes generation of an excessive number of components in splitting, because the splitting is done also in linear dimensions, which is unnecessary. Examples of this kind of GMFs are Prior Density Splitting Mixture Estimator [61], Box GMF [1, 3], Sigma Point GMF [4], Adaptive Gaussian sum squared-root cubature Kalman filter [81], and Split and Merge Unscented GMF [20].

In [61], the nonlinearity measure $\eta^{\mathrm{R\&H}}$ (52) is computed for every resulting component and the prior is split into smaller components until the maximal nonlinearity and the sum of nonlinearities of the resulting components are below set thresholds. A component is split along all the dimensions of the prior covariance (unless splitting directions are manually determined). The splitting is based on precomputed parameters for 1-dimensional GM that approximates a normal distribution with a zero mean and a unit variance. If the parameters for the ith component of an m-component one-dimensional mixture are denoted as weight $w_{m,i}$, mean $\mu_{m,i}$ and variance $\sigma^2_{m,i}$. Parameters for a resulting component are

$$w = w_0 \prod_{i=1}^{n} w_{m,c_i} \tag{94}$$

$$\mu = L(P_0) \begin{bmatrix} \mu_{m,c_1} \\ \vdots \\ \mu_{m,c_n} \end{bmatrix} + \mu_0 \tag{95}$$

$$P = L(P_0)\,\mathrm{diag}\left(\sigma^2_{m,c_1}, \ldots, \sigma^2_{m,c_n}\right) L(P_0)^T, \tag{96}$$

where w_0 is the weight of split component, μ_0 is its mean, $L(P_0)$ is the Cholesky decomposition of the covariance of the split component, c_i is an index in range $1, \ldots, m$, and n is the dimension of the state. For the complete mixture all combinations of c_i are used. The number of resulting components of a split of one component grows exponentially as a function of the state dimension $m_{\mathrm{TOT}} = m^n$.

In Box GMF [1, 3], the nonlinearity of a component is determined using the nonlinearity measure (64). A nonlinear component is divided by a grid and each grid cell is replaced with a normal distribution having the mean and covariance of the pdf inside the cell and a weight equal to the amount of probability mass inside the cell. The resulting

mixture converges weakly to the prior component i.e. the cumulative distribution function (cdf) converges, but the pdf becomes more peaky when number of components increases. To reduce the peakiness of the prior approximation the component covariances can be made larger, but then the covariance of the original component is no longer preserved [3]. The number of components also in Box GMF grows also exponentially.

In Sigma Point GMF, [4] only one-dimensional measurements were considered and the amount of nonlinearity was computed with the one-dimensional version of (64). Sigma Point GMF spreads the resulting components so that they have means similar to the UKF sigma points in (26). This produces $2n+1$ components from every split. Even though it is no longer an exponential number of components it can still be very ineffective when the state dimension is high.

In [20], [81], [19], and [P2], a prior component is split into two equally weighted components so that the mean and covariance of the mixture do not change[2]. The parameters of new components are

$$\mu_+ = \mu_0 + \sqrt{\beta}\, a \tag{97}$$

$$\mu_- = \mu_0 - \sqrt{\beta}\, a \tag{98}$$

$$P = P_0 - \beta\, a\, a^T \tag{99}$$

$$w = \frac{w_0}{2}, \tag{100}$$

where μ_+ and μ_- are new component means, P_0 is the component covariance before splitting, P is the new component covariance, a is the displacement vector of the mean that determines the direction where new component means are located compared to the original mean, and β determines the shape of the resulting components. The parameter β is chosen from range $[0,1]$, where 0 produces components having same covariances as the original component and 1 singular components. Figure 10 shows the effect of the choice of β on the shape of the resulting components.

In [20], the component to be split is chosen to be the one with highest posterior probability (44) that has the amount of nonlinearity

[2] [19] presents also similar equations for splitting into three components at once

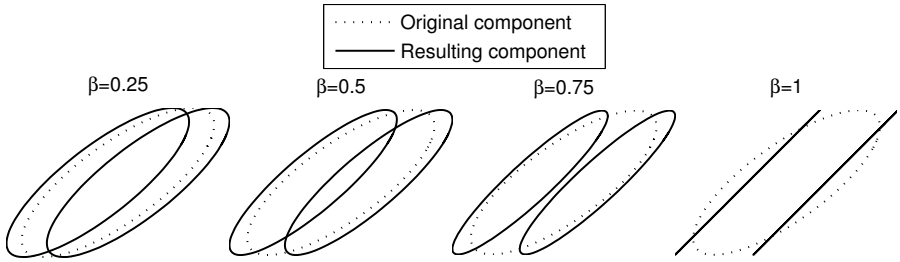

Figure 10: The effect of the value β on the shape of resulting components [P2]

according to the measure (55) above a set threshold. The direction of splitting a is chosen to be the direction of the eigenvector, u_{MAX}, corresponding to the largest eigenvalue, λ_{MAX}:

$$a = \sqrt{\lambda_{\mathrm{MAX}}} u_{\mathrm{MAX}}. \tag{101}$$

In [81], the splitting direction the same, but the component to be split is chosen using the nonlinearity measure $\eta^{\mathrm{F\&K}}_{\mathrm{TOT}}$ (57).

In [35], the vector a is computed as in (101), but instead using eigenvector corresponding to maximum eigenvalue, the eigenvector is chosen to be the one that produces most nonlinearity according to (88). The component to be split next is the component with highest value of

$$w_i^{\Gamma} \left(1 - e^{-\eta^{\mathrm{H}}}\right)^{1-\Gamma}, \tag{102}$$

where Γ is a weighting factor. When $\Gamma = 0$ the value of the nonlinearity measure η^{H} (60) determines which component is split and when $\Gamma = 1$ the component with the largest weight is split.

In [19], a is chosen to be

$$a = \sqrt{\lambda} u, \tag{103}$$

where u the is eigenvector of P_0 closest to the direction of the maximum nonlinearity computed using (86) and λ is the corresponding eigenvalue.

In [P2], a method called the Adaptive Splitting (AS) was presented. In the AS the splitting is done in the direction of the maximum nonlinearity computed using (92). The change of the trace term tr $PHPH$ in this splitting can be computed. The trace term before splitting is

$$\operatorname{tr} P_0 H P_0 H = \operatorname{tr} L(P_0)^T H L(P_0)\, L(P_0)^T H L(P_0) = \sum \lambda_i^2, \tag{104}$$

where λ_i is ith eigenvalue computed in (91), and after the split the trace term becomes

$$\operatorname{tr} PHPH = (1-\beta)^2 \lambda_i^2 + \sum_{j \neq i} \lambda_j^2. \tag{105}$$

The components are split into two until the amount of nonlinearity according to measure (64) is below 1 or the maximum number of recursions is reached.

The division of the prior into two components recursively one component at a time in the direction of the maximum nonlinearity preserves the mean and covariance of the original distribution, but does not converge to the original distribution in terms of the pdf or cdf.

In [30], the splitting is done by replacing the first dimension of an n-dimensional standard normal distribution with a precomputed mixture[3]. Then an affine transformation is applied to align the first dimension of the new components with the direction of the maximum nonlinearity and to scale and transform the mixture to approximate the original prior well. Precomputation of the mixture parameters allows to use more computational resources in optimization than computing the parameters during the estimation process. The precomputed mixtures require storing or transmitting mixtures with different numbers of components to the positioning device. For one-dimensional mixtures the storage requirements are rather small. This method does the splitting only in one direction and so it does not work well in situations where multiple dimensions are nonlinear.

In [P7], splitting of a prior component using weights of the binomial distribution is presented. This kind of mixture is called the Binomial

[3] [35] proposes similar procedure as an alternative to splitting into 2 components recursively.

Gaussian Mixture (BinoGM). When the number of components is increased in the BinoGM the pdf and cdf converge to the pdf and cdf of the prior component while preserving the mean and covariance.

In addition to the convergence results, [P7] presents algorithms and rules for the practical use of the BinoGM in a filter called the Binomial Gaussian Mixture Filter (BinoGMF). In the BinoGMF, measurements are decorrelated (65) and each decorrelated measurement is treated separately. The amount of nonlinearity for a component is $\operatorname{tr} P\hat{H}_i P\hat{H}_i$, where $\hat{H}_i$ is the Hessian of the ith element of the decorrelated measurement function. In the BinoGMF, the component splitting can happen in multiple directions. Because the nonlinearity measure can be written as a squared sum of eigenvalues of PH (104), each squared eigenvalue can be seen as the amount of nonlinearity to the direction of the corresponding eigenvector. The splitting in multiple directions at once saves computational resources as the nonlinearity measure is evaluated only once.

In the BinoGMF each new component has the same covariance

$$P = L(P_0)\, U \operatorname{diag}\left(\frac{1}{m_1}, \ldots, \frac{1}{m_n}\right) U^T L(P_0)^T, \tag{106}$$

where U is computed using eigendecomposition (91) and m_i is number of mixture components in the ith direction. The nonlinearity of a resulting component is

$$\operatorname{tr} P\hat{H}P\hat{H} = \sum_{i=1}^{n} \frac{\lambda_i^2}{m_i^2}, \tag{107}$$

where λ_i is the ith eigenvalue computed with eigendecomposition (91). It is shown in [P7] that if the required number of components for achieving a desired nonlinearity level is minimized and the integer nature of m_i is neglected the optimal values for m_i satisfy

$$\frac{\lambda_i^2}{m_i^2} = \frac{\lambda_j^2}{m_j^2} \vee m_i = 1 \tag{108}$$

on conditions $m_i \geq 1$ and $\sum \frac{\lambda_i^2}{m_i^2} = \eta_{\text{limit}}$, where η_{limit} is the threshold value for nonlinearity. The parameter value $m_i = 1$ is used for directions which would otherwise have number of components $m_i < 1$.

The component mean is

$$\mu = L(P_0)U\text{diag}\left(\frac{1}{\sqrt{m_1}},\ldots,\frac{1}{\sqrt{m_n}}\right)\begin{bmatrix}2c_1-m_1-1\\2c_2-m_2-1\\\vdots\\2c_n-m_n-1\end{bmatrix}+\mu_0, \quad (109)$$

where c_i is an integer $1 \leq c_i \leq m_i$ and the whole mixture contains all combinations of c_i. The weight of a component is

$$w = w_0 \prod_{i=1}^{n} \binom{m_i-1}{c_i}\left(\frac{1}{2}\right)^{m_i-1}, \quad (110)$$

where w_0 is the component weight before splitting. Splitting in only nonlinear directions helps to keep the number of components low. In the BinoGMF the maximum number of resulting components of splitting a component is defined to be proportional to the weight of the component.

The above equations for splitting are similar to splitting done in [61] (94)-(96). The main difference is that the BinoGM split assigns the number of components for each dimension according to the nonlinearity in that dimension and reduces the amount of required components. When the number of dimensions, which are considered almost linear, increases, the number of resulting components does not change and in this kind of situations the BinoGMF can avoid the exponential growth of the number of components.

Figure 11 shows the forming of the GM in the BinoGMF. The background contours of the top row show the second order term of a measurement function, the arrows show the directions of eigenvectors in (91), and the ellipse is the 2σ ellipse of the covariance matrix. The top left plot shows the original component and the squared eigenvalues are $\lambda_1^2 = 8$ and $\lambda_2^2 = 2$. According to the rule (108), the resulting mixture will have 4 components in the direction of the first eigenvector and 2 components in the direction of the second eigenvector. The resulting components will all have the same covariance. The 2σ ellipse of this covariance is shown on the top right. The total nonlinearity is reduced to 1. The bottom left plot shows the pdf of

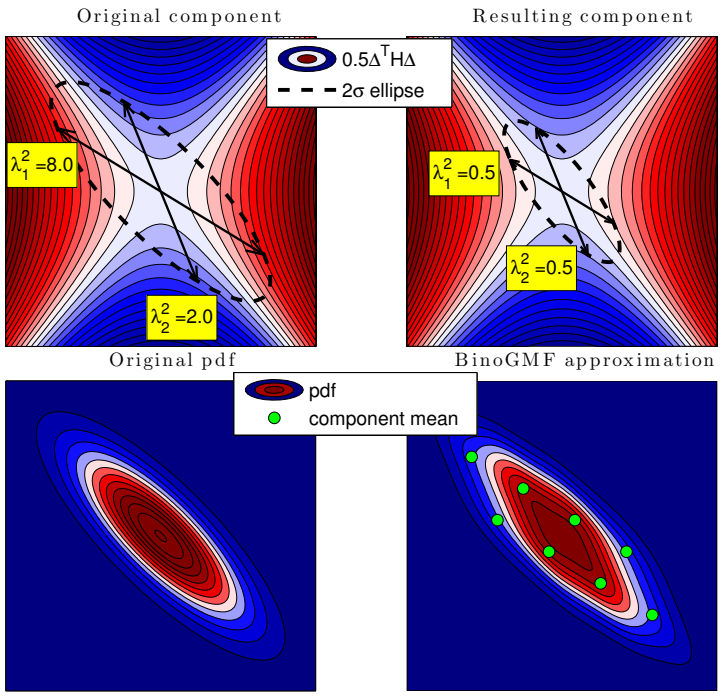

Figure 11: An example of nonlinearity reduction and a generation of a BinoGM. The background of the top row shows the second order term of a measurement function. The total nonlinearity is reduced from 10 to 1 for the resulting component. The bottom left plot shows the pdf of the original component and the bottom right plot shows the pdf of the BinoGM.

the original component and the bottom right plot shows the pdf of the resulting BinoGM.

Some of the component splitting methods presented in this section are evaluated in Figure 12. The figure shows the mean residuals of the estimated means to the mean of the true posterior in four different scenarios. Markers show the 5%, 25%, 50%, 75%, and 95% quantiles of the residuals. Each test scenario was run 400 times. The different nonlinearity based limits for number of components were not tested and the maximum number of components was limited to 16. All component updates were done using the UKF with parameters $\alpha_{\mathrm{UKF}} = 1$, $\kappa_{\mathrm{UKF}} = 0$ and $\beta_{\mathrm{UKF}} = 2$. The methods that use the UKF sigma

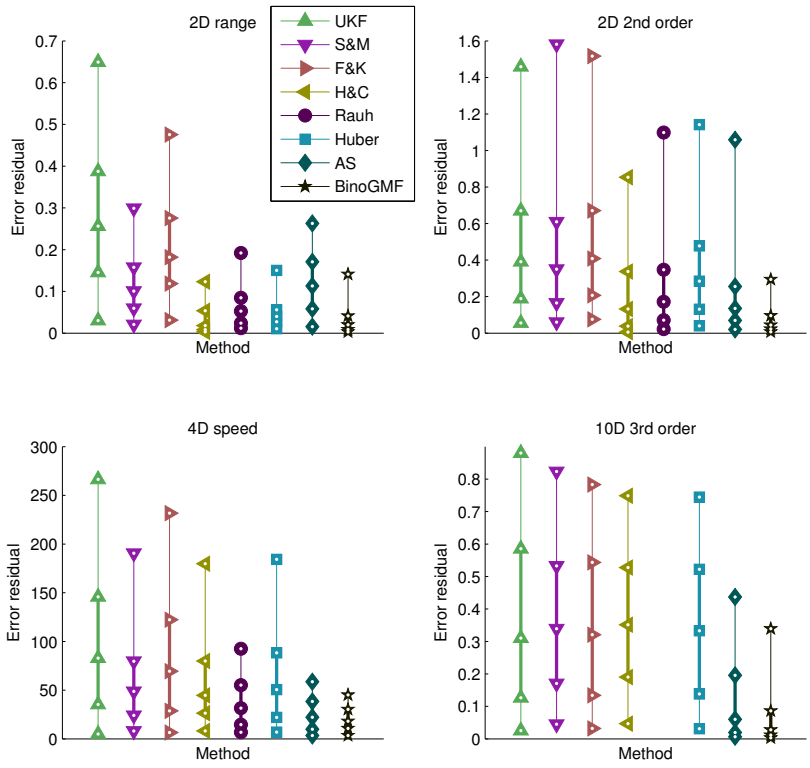

Figure 12: Quantiles of mean estimation errors of GMs generated with different splitting strategies in four test scenarios. All GMs are updated with the UKF.

points for nonlinearity evaluations used these same parameters. The scenarios are:

- 2D range – a range measurement in two-dimensional positioning
- 2D 2nd order – measurement consists of a second order polynomial term aligned with a random direction and a linear measurement aligned with another random direction
- 4D speed – a speedometer measurement with a highly uncertain prior
- 10D third order – a 3rd order polynomial measurement along a random direction in a ten-dimensional state

Detailed explanations of the test scenarios can be found in [P7].

In the figure, the method S&M is from [20]. The method F&K is from [19] and splits each component every time into three. H&C was implemented with different offline optimization than originally presented in [30], but this should not affect the results. Rauh from [61] used the same offline optimized mixtures as H&C but splits the prior into multiple directions at once. Huber was implemented according to [35]. AS is from [P2] with the alteration that it splits the most nonlinear component into two instead of splitting components recursively until the nonlinearity threshold is reached. The BinoGMF is implemented according to [P7].

In these results, the use of the BinoGMF for splitting a prior components results in the most accurate posterior. The method Rauh that splits the prior along every direction is not shown in the 10D scenario, because splitting the prior into two along 10 dimensions would generate 1024 components. In this scenario the AS and the BinoGMF were the only methods that improved the median error compared to the UKF. This is due to the accurate estimation of the direction of the maximum nonlinearity.

The AS has better accuracy than the BinoGMF sometimes when the Hessian H changes much within the prior i.e. when the second order approximation of the measurement function within the prior is inaccurate. In [P7] it was found that in such cases the sigma point spread in the BinoGMF should be larger in the nonlinearity computation than commonly used in the UKF implementations. Figure 13 shows a situation, where the AS has better accuracy than the BinoGMF.

On the left the background contours show the pdfs of different priors used and on right the posterior pdfs are shown. The white dashed line shows the maximum likelihood circle of the applied range measurement and the green dots are the component means. α_{UKF} defines the sigma point spread in the numerical computation of nonlinearity.

In the figure titles, m is the number of components when the splitting stopped. In the BinoGMF the nonlinearity value is not evaluated for resulting components whereas AS does splits until the amount of nonlinearity in every component is under a set threshold $\frac{\mathrm{tr}\,PHPH}{R} \leq 1$.

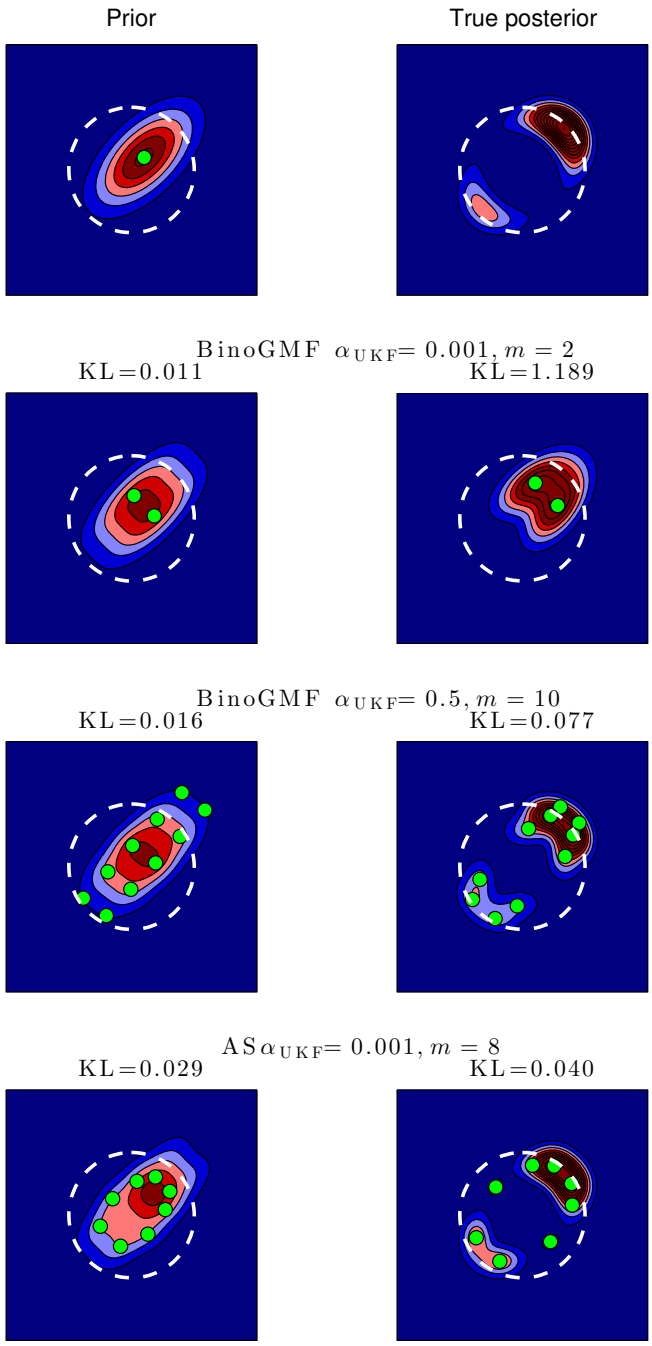

Figure 13: Comparison of priors and posteriors of the AS and the BinoGMF with different parameters

KL is the Kullback-Leibler divergence (51), which describes how good is the GM approximation of the true distribution.

The BinoGMF that uses small α_{UKF} values uses numerical approximation of Hessian H that is close to the analytical Hessian at the mean. The analytical solution has all the nonlinearity tangential to the range measurement circle. This results in splitting only in that direction. When the α_{UKF} value is increased, the numerical approximation takes into account also the error in the direction of the range measurement. This results in more components and more accurate posterior estimate. The AS splits every time in the direction of the maximum nonlinearity and because the amount of nonlinearity is re-evaluated for new components the choice of the sigma point spread is not as important as with the BinoGMF.

3 Linear models for positioning problems

The previous section considered methods for evaluating the nonlinearity within a Gaussian component used in a Gaussian Mixture Filter (GMF). When the amount of nonlinearity was small, the component was updated with a Kalman Filter (KF) extension and when nonlinearity was high the component was split into smaller components. In this section a different approach is considered. Instead of locally linearizing a nonlinear model, a globally linear model is used.

While the previous section considered general situation and can be applied to various different estimation problems, the construction of a globally linear model is case specific and in this thesis positioning problems are considered. The problems considered are real-world problems and the used models are only approximate models of true phenomena. The solutions to the linear models are fast to compute and thus they can be applied in real-world situations, where computational capacity is limited.

The benefits of using linear models and assuming noise normally distributed are as follows:

- Closed form solutions exist e.g. the KF and the Rauch-Tung-Striebel smoother [60]
- Algorithms are fast
- The result is a normal distribution, which is unimodal and has the mean and mode in the same point

The drawbacks are as follows:

- Models are more restricted, it is possible that some information cannot be used
- Assumption of normally distributed noise is not always good
- There is no general algorithm for finding a linear model to be used instead of a nonlinear model
- No good linear model exists for every problem

In some publications “tricks” are applied to make a linear model from nonlinear, but the solutions computed are not solutions to original problem [70, III-C].

Figure 14 presents three situations of estimation with a linear and nonlinear model. In every situation the nonlinear model is $y = x + 2\sin(x)$ and the linear model is $y = x$. The nonlinear model is assumed to be the "true" model. Three different ways to compute an estimate of state x given measurement y are discussed:

1. Solving x from the linear model
2. Solving the estimate from the nonlinear model using a linearization
3. Solving the estimate from the nonlinear model using an iterative solver that converges to a local optimum

In the first subplot, the linearization point is close enough to the true state to produce a linearized estimate close to the true state. If the linearization point is used as an initial state of the iterative solver, it will most probably converge to the correct local optimum. In this case the linear estimate has much larger error than the linearized estimate.

In the subplot in the middle the situation changes. The true state is further away from the linearization point. The estimate solved from the linear model is now better than the estimate solved from the linearized model. An iterative solver would have a risk of not finding the global optimum, but converging to a local optimum, because there are two local optima between the linearization point and the true location.

In the bottom plot, noise is added to the measurement and the measurement does not intersect with the true model in the location of the true state. In this case all the local optima have more error than the linear estimate and the linear estimate is the most accurate.

These examples show how an accurate nonlinear model can produce better results than the linear model, but if there is no good prior information of the state or if the measurements are noisy a linear model may provide better results. Of course, not all nonlinear models have this good linear counterpart, but in the following subsections three real-world situations, where a linear model has benefits over a nonlinear model, are presented. The situations are based on publications [P3] - [P6].

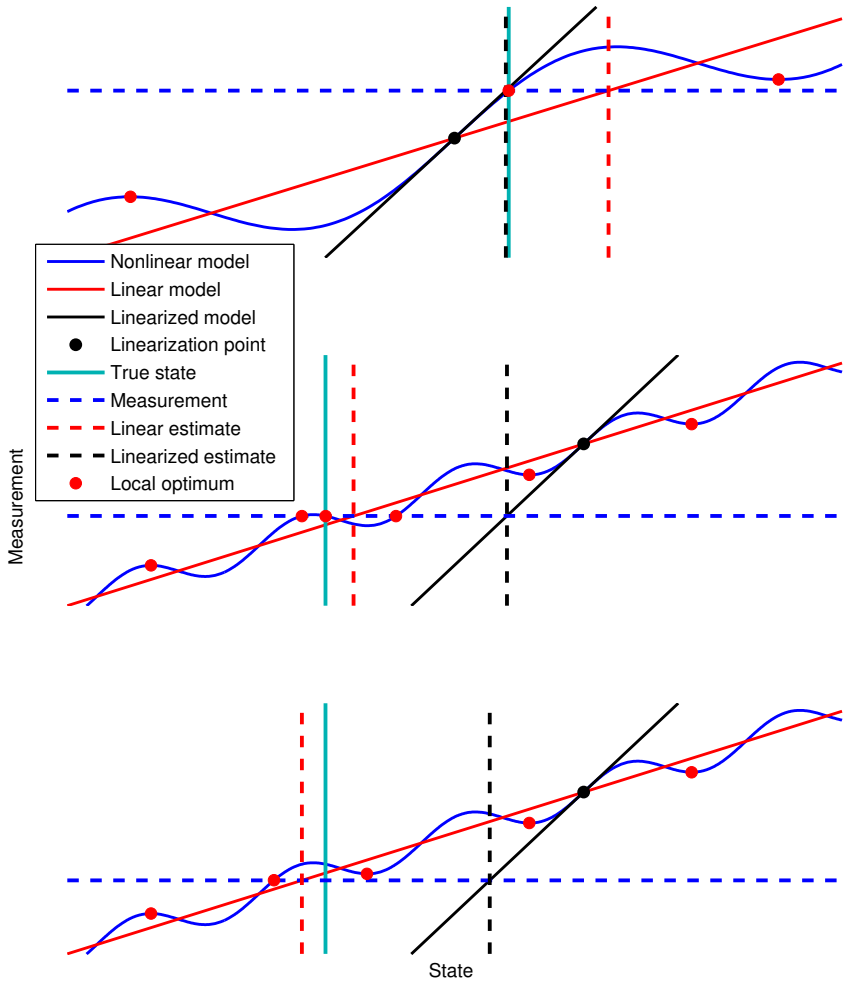

Figure 14: An example of estimates obtained from linear, and linearized, and nonlinear models

3.1 Generation of a radio map using unlocated fingerprints

Generation of a radio map using unlocated Fingerprints (FPs) arises in the situation when one wants to generate a radio map for Wireless Local Area Network (WLAN) positioning indoors, where both Global Positioning System (GPS) and manual position estimates are unavailable. This is the case when WLAN mapping is done by crowdsourcing and has GPS position estimates only outdoors. This section is based on publication [P3].

The problem at hand is shown in Figure 15. The FPs outside the box have position information and the FPs inside are unlocated. Connections show which WLAN Access Points (APs) were received at a FP. The goal is to build a radio map of APs. Here I present an overview of the algorithms in the context of the thesis. The full algorithms and derivations can be found in [P3].

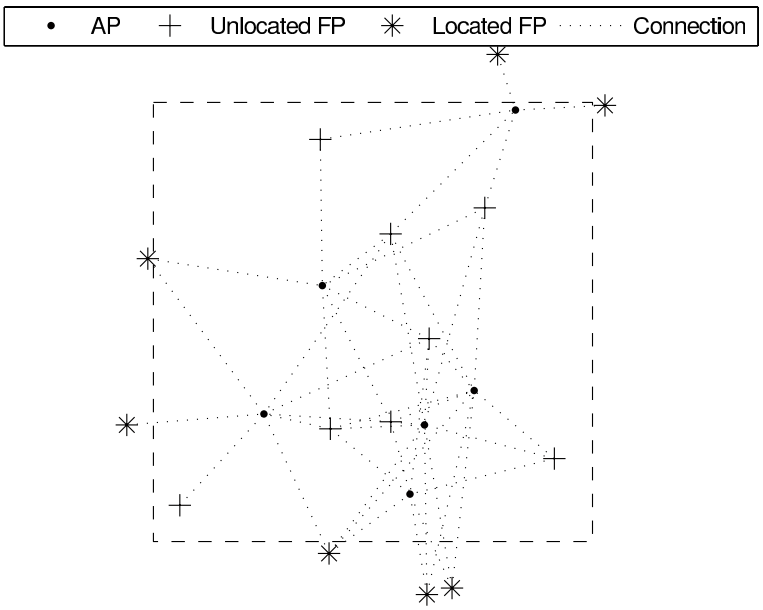

Figure 15: Unlocated and located fingerprints forming connections with APs [P3]

The first algorithm used is a very simple mean algorithm that is used as a baseline in tests. The location of tth FP z_t in this model is

$$z_t = \begin{cases} z_{\text{GPS}} & \text{if GPS is available} \\ \sum_{i \in \text{FP}} \frac{r_{\text{AP},i,t-1}}{M} & \text{if GPS is not available} \end{cases}, \tag{111}$$

where z_{GPS} is a location estimate provided by GPS, $r_{\text{AP},i,t-1}$ is an old estimate of the ith AP and M is the number APs in the tth FP. The updated AP location is

$$r_{\text{AP},i,t} = \frac{(t-1)r_{\text{AP},i,t-1} + z_t}{t}, \tag{112}$$

when ith AP is received t times.

In the second model a Pathloss (PL) model is used. In this model, the distance to an AP is related to Received Signal Strength (RSS) at the tth FP using model

$$\text{RSS}_{i,t} = \text{RSS}_0 - 10\alpha \log_{10} \left\| z_t - r_{\text{AP},i} \right\| + \varepsilon_{y,t}, \tag{113}$$

where RSS_0 is the signal strength 1 meter from the AP, α is an attenuation factor, z_t is the location of tth FP , $r_{\text{AP},i}$ is the location estimate of ith AP and $\varepsilon_{y,t}$ is the measurement noise. Some of the z_t are unknown (unlocated FPs) and some are known. The map is generated and positioning is done in two dimensions.

In [P3] the distance is solved from (113) without taking the measurement error into account

$$\left\| z_t - r_{\text{AP},i} \right\| = 10^{\frac{\text{RSS}_{i,t} - \text{RSS}_0}{10\alpha}}.^4 \tag{114}$$

Estimates for locations of APs can be solved from a set of nonlinear equations containing one equation (114) for every received AP in every FP. This causes the system to become large when the number of measurements grows.

To limit the number of equations it is possible to consider only the positions of APs and not FPs. Because the distance between two APs

[4] In [P3], this is equation (1) and has a sign error

is less or equal than the sum of their distances to the location of a FP, the following inequality holds

$$\left\| z_t - r_{\mathrm{AP},i} \right\| + \left\| z_t - r_{\mathrm{AP},j} \right\| \geq \left\| r_{\mathrm{AP},i} - r_{\mathrm{AP},j} \right\|. \tag{115}$$

For each pair of APs received simultaneously in an unlocated FP only one inequality is required

$$\min_t \left[10^{\frac{\mathrm{RSS}_{i,t} - \mathrm{RSS}_0}{10\alpha}} + 10^{\frac{\mathrm{RSS}_{j,t} - \mathrm{RSS}_0}{10\alpha}} \right] \geq \left\| r_{\mathrm{AP},i} - r_{\mathrm{AP},j} \right\|, \tag{116}$$

where the left hand side of the inequality is the smallest observed upper limits for distance between the APs. Using only the upper limits for distances between APs limits the number of equations to be solved, but loses some information. But even though the number of inequalities does not increase when new FPs, where APs i and j are received, are added the upper limit may become lower and the estimate can be thus improved.

The AP locations are solved from nonlinear a set of nonlinear equations (114) or inequalities (116) using the Gauss-Newton algorithm [7]. The Gauss-Newton algorithm is an iterative algorithm that does not necessarily converge to a global minimum.

In addition to the nonlinear models, a linear model was introduced in [P3]. The linear model employs the idea that the set of locations where each pair of APs can be received simultaneously has its own mean. Thus a FP, where APs i and j are received simultaneously, is located at

$$z_t = r_{\mathrm{AP},i} + \delta_{i,j} + \varepsilon_{z,i,j} = r_{\mathrm{AP},j} + \delta_{j,i} + \varepsilon_{z,j,i}, \tag{117}$$

where $\delta_{i,j}$ and $\delta_{j,i}$ are translation vectors from the AP locations to a virtual mean of FP locations and $\varepsilon_{z,i,j} = \varepsilon_{z,j,i}$ is a translation vector from this mean to the true FP location z_t.

Because in the mapping phase, the user locations are not of interest, they can be eliminated by taking a difference between locations of two APs

$$r_{\mathrm{AP},i} - r_{\mathrm{AP},j} = \delta_{j,i} - \delta_{i,j}. \tag{118}$$

If the translation vectors are assumed to be independent zero mean random vectors, the variance of an AP location difference has zero mean and covariance

$$\operatorname{cov}\left(r_{\mathrm{AP},i} - r_{\mathrm{AP},j}\right) = \operatorname{cov}\delta_{j,i} + \operatorname{cov}\delta_{i,j}. \tag{119}$$

The translation vectors are assumed to be time invariant, which means that having multiple measurements where both APs were received does not affect the estimate. The FPs with GPS locations are used also as linear measurements

$$r_{\mathrm{AP},i} = \frac{\sum_{i=1}^{M} z_{\mathrm{GPS},i}}{M} + \varepsilon_{\mathrm{GPS}}, \tag{120}$$

where $z_{\mathrm{GPS},i}$ is a location of a FP where GPS was available and ith AP was received and $\varepsilon_{\mathrm{GPS}}$ is a normal distributed noise term.

In simulated tests in [P3], the measurements were generated using (113) with normal distributed noise. For APs a maximum range was used to determine whether the signal was received at a FP. The locations were solved using the mean model (111), the nonlinear models (114), and (116) and linear model (117). The transformed measurement noise is not anymore normal in the equations (114) and (116), but this error was considered small compared to real-world modeling errors. The linear model was tested also with variances of (119) based on the signal strengths so that weak measurements were transformed into large variances of δ and strong measurements to small variances.

Figure 16 shows the simulated results for positioning APs as a function of measurement noise. Simulation was done by evaluating the accuracy of each method for 11 levels of noise 864 times. The method "Mean" is the algorithm that computes the weighted mean recursively (111). The Access Point Least Squares (APLS) uses the linear model (117) and Variance Access Point Least Squares (VARAPLS) is the linear model with variable weighting. Gauss-Newton (GN) uses (114) and Gauss-Newton Max Range (GNMax) (116). GN and GNMax have also versions with initial guess taken from the APLS, denoted with "w prior" in the legend. Otherwise the initial guess for all the AP locations in GN and GNMax was the mean of the located FPs i.e.

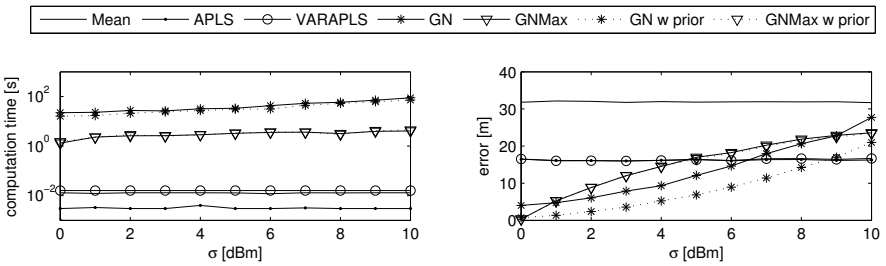

Figure 16: Simulated results with different noise levels for locating APs with unlocated FPs

in the first Gauss-Newton iteration all FPs had same location.. The linear methods outperformed all the nonlinear methods when the measurement noise level was over 9 dBm. The figure shows also that the nonlinear methods were over 100 times slower than the linear. The use of APLS as the initial guess for GN improved the accuracy, but did not have a significant effect on the computation time.

In [40], the standard deviation for the RSS of a static receiver was measured to be between 1 dB and 5 dB, but the mean was changing up to 4 dB depending on the orientation of the user. In [82], the standard deviation of WLAN RSS values for the model (113) was found to be usually more than 10 dB. In their tests the orientation of the user changed RSS values up to 14 dB and state of a door (open or closed) 5 dB. Due to these findings, even if the RSS signal strength is attenuated according to (113), the linear models should outperform the nonlinear models in real-world tests.

In [P3], the positioning using unlocated FPs was tested also in one real-world scenario. The results showed that positioning using the linear model produced better user location estimates than the nonlinear models. The absolute positioning accuracy was enough to determine in which wing of the building the user is. This accuracy is enough to show the user a floor plan from the correct part of building, if the correct floor is not needed, but is not enough for plotting the exact position on the floor plan.

The algorithms used in the article [P3] could be also used in the Simultaneous Localization and Mapping (SLAM) framework. The

algorithms can build automatically a WLAN map for positioning and update it while moving in the mapping area.

The nonlinear model that uses equalities solves estimates for FP locations and could be used directly for SLAM. The model could be enhanced further by using a motion model for the FP locations. If the other models are used for SLAM, the user locations has to be solved separately.

3.2 Linear coverage area models

In this section I consider positioning using WLAN signals. The environment is first mapped and models are created by collecting FPs with manually determined positions and the positioning phase happens after the map is created. This section is based on publications [P4] and [P6].

Positioning with Coverage Area (CA) models is based on knowledge of the geographical area where signals can be received. CAs can be modeled as different geometrical or statistical objects. One could for example construct a convex hull around all the FP locations where an AP was received. In the positioning phase, the intersection of the polygon models of all received APs can be used as the location estimate. This approach has some drawbacks, for example, the storage and memory requirement for a model is not bounded, because number of vertices is not limited and the intersection of the convex hulls may be empty. A geometrical intersection of convex hulls can be formulated as

$$p(x) \propto \prod_i h_i(x), \tag{121}$$

where

$$h_i(x) = \begin{cases} 1, \ x \text{ inside } i\text{th polygon} \\ 0, \ \text{otherwise} \end{cases} \tag{122}$$

In geometrical intersection state x is within the intersection when $p(x) > 0$.

In [43], the CA of an AP was modeled as a two dimensional normal distribution. With this kind of model the CAs expand to whole space, which means that their "statistical intersection" is not empty.

input : M FPs locations z_i where an AP was received
output : mean c and covariance Σ of a CA model
parameters : degrees of freedom ν
strength of the prior τ
prior radius l
number of iterations T

$\phi_{1:M} \leftarrow 1$ // Initialize weights
for $t = 1 : T$ **do**
  $c \leftarrow \frac{\sum_{i=1}^{M} \phi_i z_i}{\sum_{i=1}^{M} \phi_i}$ // CA mean estimate
  $S \leftarrow \sum_{i=1}^{M} \phi_i (z_i - c)(z_i - c)^T$
  $\tilde{\Sigma}^{-1} \leftarrow (M + \tau - n - 1)\left(S + \tau l^2\right)^{-1}$ // Shape matrix
  for $i = 1 : M$ **do**
    $\phi_i \leftarrow \frac{n+\nu}{\nu + (z_i - c)^T \tilde{\Sigma}^{-1} (z_i - c)}$ // Update weights
  end
end
$\Sigma = \frac{\nu}{\nu - 2} \tilde{\Sigma}$ // Covariance estimate

Algorithm 2: CA model fitting to FP locations where an AP was received [54]

In "statistical intersection" $h_i(x)$ in (121) is the probability density function (pdf) of the distribution of the ith CA model and (121) is proportional to the pdf of the "statistical intersection". The normally distributed model also allows using a prior to represent the shape of a typical CA. The prior affects the shape most when there is only one measurement and the effect decreases as the number of FPs increases. The use of a normal distribution as the estimate of a CA also allows using fast linear algorithms in both generating the models and in positioning.

When there are outliers in the FP data their impact on normal models is large. In [54], Student-t models are fitted to the FP data instead of normal models using iterative expectation-maximization algorithm (Algorithm 2). When modeling the CA of an AP, the model is made for 2-dimensional positioning i.e. $n = 2$. The FP locations are denoted with z and the number of FPs where the AP was received is M. The resulting CA is parameterized with a mean c and a covariance matrix Σ. When the degrees of freedom ν approaches infinity the

distribution approaches normal distribution and the parameters for a CA can be computed without iterations:

$$c = \frac{\sum_{i=1}^{M} z_i}{M} \tag{123}$$

$$\Sigma = \frac{\sum_{i=1}^{M} (z_i - c)(z_i - c)^T + \tau l^2}{M + \tau - n - 1}. \tag{124}$$

Even though Algorithm 2 is not very complex, the computation of linear-Gaussian model using formulas (123) and (124) is even simpler, faster, and does not require iterations.

The positioning using statistical CA models can be done using the Weighted Linear Least Squares (WLLS) (8)&(9). The WLLS produces the parameters of the optimal "statistical intersection" of normal distributions. The CA models of received APs are assumed independent and the WLLS algorithm can be simplified further: The mean of a statistical intersection of CA models of the received APs in a FP is

$$\mu = \left(\sum_{i \in \mathrm{FP}} \Sigma_i^{-1} \right)^{-1} \sum_{i \in \mathrm{FP}} \Sigma_i^{-1} c_i \tag{125}$$

and the covariance matrix is

$$P = \left(\sum_{i \in \mathrm{FP}} \Sigma_i^{-1} \right)^{-1}. \tag{126}$$

In [52], the positioning with normal CA models had a mean error of 8.8 m in indoor WLAN positioning whereas PL-based method reached a 6.9 m mean error and Weighted k-nearest Neighbor (WKNN) [33] had a 5.7 m error. In this comparison, the CA model was the only method that did not take the value of the RSS into account.

In [P4], the statistical CA models are enhanced to take the RSS values into account. This is achieved by making multiple models for a CA of an AP based on the RSS values. In the article it was reported, that two CA models for an AP improved positioning results indoors by 2 m and the accuracy was similar to WKNN. Increasing the number of CA models to 3 for an AP did not further improve the estimate

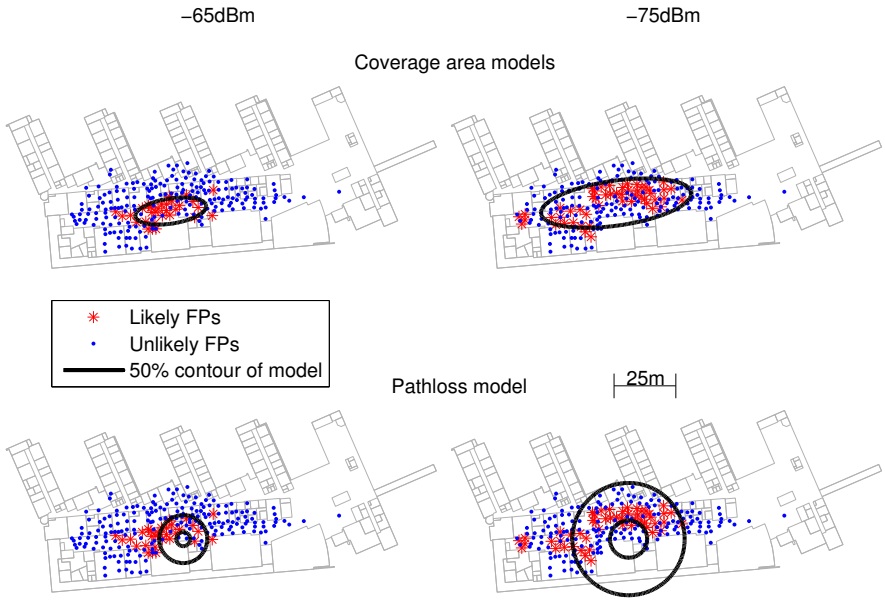

Figure 17: Comparison of a two-level CA model with a PL model when modeling the FPs with different RSS values

significantly. The tests were done using real data measured indoors and outdoors. In the indoor scenario Student-t ($\nu = 5$) models were better than the normal models ($\nu = \infty$), but outdoors the situation was opposite.

Figure 17 shows a comparison of the 2-level CA model and a PL model with two different measured RSS values for an AP. The PL model is similar to (113), but takes also the model uncertainty into account [52]. The black lines enclose 50% of the probability mass of the models and the likely FPs are chosen according to RSS in such a way that they have 50% of the probability mass. The used probability for computing likely FPs is

$$p(\mathrm{FP}_i) \propto p_{\mathrm{N}}(\mathrm{RSS}_y | \mathrm{RSS}_i, 6^2\,(\mathrm{dBm})^2), \tag{127}$$

where RSS_y is the current measured RSS value and RSS_i is the RSS value at the ith FP.

The figure shows how the models are different, but there is no clear difference which method represents the distribution of likely FPs better. The PL model extends to areas where the FPs were not collected, whereas the CA model does not. The property of extrapolating data may be desired if the FP data does not cover the whole interesting positioning area, but it may also be undesired property. For example, if a building is well mapped and outside the building GPS positions are used, the position estimate, when only WLAN measurements are available, should be indoors.

In [P6] the 1- and 2-level CA positioning is compared with other WLAN positioning methods. Figure 18 shows the quantiles of positioning errors on true-data tests. WKNN is only non-parametric method in comparison. It compares the received RSS values of APs with a database of previously collected FPs. The location estimate is computed as a weighted mean of the locations of k FPs that have most similar RSS values [33]. The weights are determined by the similarity of FPs. In the tests $k = 5$. CA 1-level and 2-level are normal CA models with one and two CA models for an AP. PL [52] uses a PL model similar to (114) to determine range to the APs. Generalized Gaussian Mixture (GGM) uses a generalized version of Gaussian Mixtures (GMs) to estimate the ranges from the AP [49]. Gaussian Mixture Expectation Maximization (GMEM) is a method that is based on estimating the RSS map using a GM [41].

The tests were made in two buildings using 4 test tracks that contained 308 test points in total. In these tests, 2-level CA positioning has the best 95%-quantile error and the median is on the same level as with other methods. Time is the computing time of methods in milliseconds and is only suggestive as the different algorithms have different optimization levels. The time does not take database access into account. For static results the position is estimated without using time series and for filtering results time series are used. Figure 19 shows the sizes of compressed radio map databases required for each method. Clearly the parametric models require less storage space than the WKNN. A more detailed explanation on test setup and more results can be found in [P6].

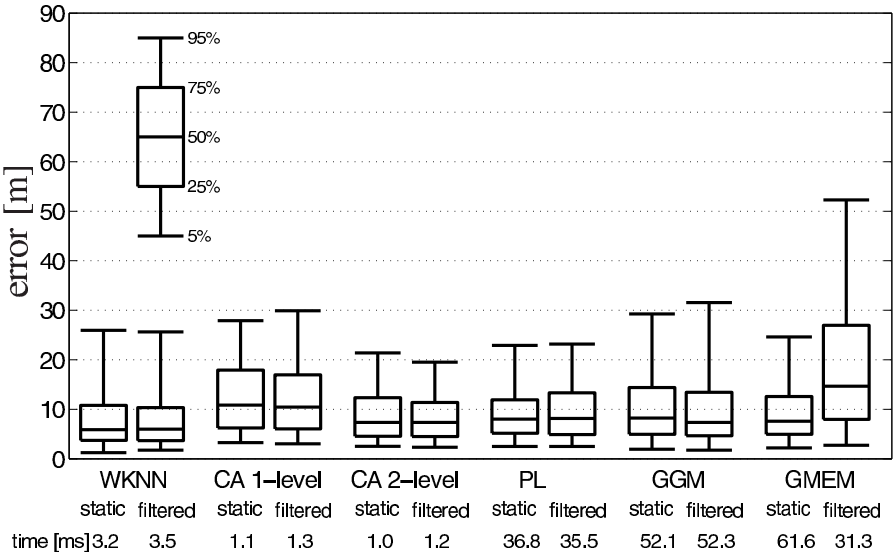

Figure 18: Accuracy and time comparison of different WLAN positioning methods on true-data test tracks [P6]

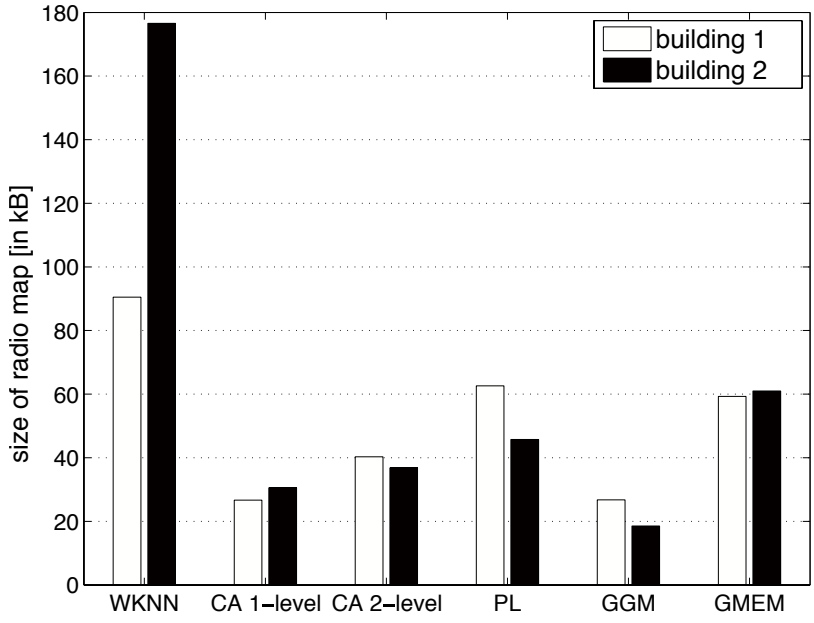

Figure 19: Database size comparison of different WLAN positioning models [P6]

3.3 Linear state model for Pedestrian Dead Reckoning

In Pedestrian Dead Reckoning (PDR) the location of a pedestrian is estimated using information on footsteps and heading change fused with other measurements providing occasional absolute position estimates. This section is based on publication [P5] and the absolute position estimates are computed using the CA models presented in previous section and in [P4].

Footsteps and heading changes can be obtained from an Inertial Measurement Unit (IMU), which consists of gyroscopes and accelerometers. The device's heading change $\Delta\theta$ can be computed by projecting the gyroscope measurements to a horizontal plane, which is estimated using accelerometers [12]. Footsteps can be detected from the accelerometer data. Often in literature the footstep length s is integrated from the accelerometer data assuming that when the foot hits the ground it is static [6]. This kind of footstep length estimation requires the accelerometers to be mounted on a foot, which is impractical in many use cases. There are also other methods to estimate the step length that are based on the time between steps [45] or the time and variance of acceleration [68]. Methods based on time do not require the sensors to be mounted on a foot, but they may have some bias depending on the person's physiological properties.

PDR systems in literature, for example in [45], are usually modeled with two-dimensional position z, heading θ, and, if the footstep length is estimated, the footstep length s. When a step is detected the state transition is

$$x_{(t+1)} = \begin{bmatrix} z_{[1],(t+1)} \\ z_{[2],(t+1)} \\ \theta_{(t+1)} \\ s_{(t+1)} \end{bmatrix} = \begin{bmatrix} z_{[1],(t)} + s_{(t)} \cos\theta_{(t)} \\ z_{[2],(t)} + s_{(t)} \sin\theta_{(t)} \\ \theta_t + \Delta\theta_{(t)} \\ s_{(t)} \end{bmatrix} + \varepsilon_x, \qquad (128)$$

where ε_x is a random vector containing the state noise for the state variables. This state model is nonlinear. If the footstep length is obtained directly from the IMU, then s can be dropped from the state and instead the measured value of the s is used. In this and the

following model, the step is assumed to be done at the heading in time t and the heading change happens at the end of the step.

In [P5], a linear state model for PDR is proposed. In this model, instead of estimating the heading and footstep length, a two-dimensional step vector v is estimated. The model is

$$x_{(t+1)} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & \cos\Delta\theta_{(t)} & \sin\Delta\theta_{(t)} \\ 0 & 0 & -\sin\Delta\theta_{(t)} & \cos\Delta\theta_{(t)} \end{bmatrix} \begin{bmatrix} z_{[1],(t)} \\ z_{[2],(t)} \\ v_{[1],(t)} \\ v_{[2],(t)} \end{bmatrix} + \varepsilon_x. \tag{129}$$

At first glance it may seem that this model is also nonlinear as there are trigonometric functions in the state transition. This is not the case because the measured $\Delta\theta$ is used as a constant and errors are assumed to be independent of the state and $\Delta\theta$ and they are modeled with a normal distributed noise term ε_x.

If the estimated footstep length is needed, it can be computed from the estimate as

$$s = \|v\|. \tag{130}$$

The linear model cannot use a prior footstep length (other than 0) without having a prior knowledge of the heading. On the other hand the linear model does not need to have a prior mean for the heading, unlike the nonlinear model (128) (if used with a KF extension). The linear model cannot use the footstep length measurement and remain linear because (130) is a nonlinear relationship between the variables.

Because PDR measurements themselves cannot give an absolute position, other measurements giving absolute positions are required to complement the method. To get all the benefits of using linear model, the location measurements should also be linear. Outdoors Global Navigation Satellite System (GNSS) location estimates can be approximated as normal distributed location estimates and indoors WLAN measurements with normally distributed CAs can be used as in Section 3.2. The use of PDR improves the positioning accuracy and the location can be updated more often, because PDR is used at every footstep while WLAN scans take longer.

In [P5], the accuracy of the linear model (129) is compared with the nonlinear model (128) with and without footstep length measurements. The KF is used for the linear model and the Extended Kalman Filter (EKF) for the nonlinear models. The tests show that the linear model's accuracy is similar to the nonlinear model's accuracy when the initial state is known, but better when the initial state is unknown. Thus, the use of an estimated footstep length or a prior footstep length in nonlinear models is not a significant reason to use the nonlinear models.

Due to the good performance of the linear model when the initial location and heading are not known, it can be used also as an initialization filter, for example, for Particle Filters (PFs) as is done in [53]. In [53], the KF using the linear model was compared with a PF that used wall information and a nonlinear state model. The PF with wall information reduced the mean error of KF from 4.8 m to 2.0 m, while the PF without wall information had slightly worse accuracy than the KF.

Figure 20 shows the routes estimated with the linear model (129) and with the traditional nonlinear models (128) in a real-world test. The traditional model with footstep length uses footstep lengths estimated from the time interval of footsteps. The nonlinear models require a value for the mean of initial heading to be used. In this case, an initial heading with a 100-degree error and a large variance was used. For the linear model the initial step vector is a zero vector. The initial location used is the location of the first WLAN measurement. In this test case WLAN measurements do not provide accurate position estimates. The figure shows how the route computed using the linear model has the best accuracy in the first wing of the route. The nonlinear models find the correct walking direction slower, but eventually achieve similar accuracy as the linear model.

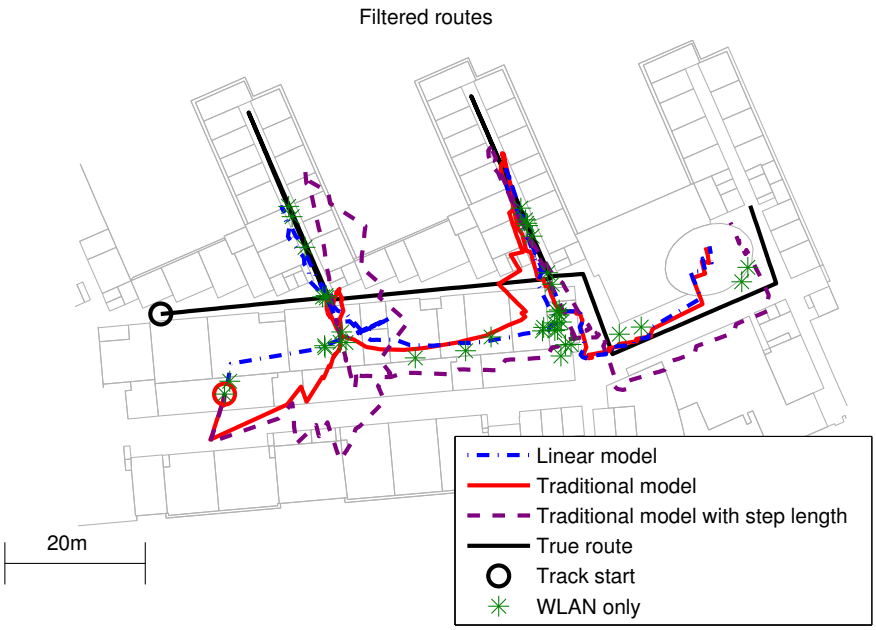

Figure 20: Filtered routes using different models in a real-world test

4 Conclusions and future work

> There is a lot of future study left in the current GMF. For example, how to build a more efficient partitioning of the state space in BGMF, how many components are enough for some given accuracy, how to reduce the number of components more efficiently, and how to select what kind of filter to use and to make this selection adaptively.
>
> Simo Ali-Löytty [2]

The quote is from Simo Ali-Löytty's PhD thesis that was done in our research group. This thesis offers answers to some of the questions above. Before the work presented in this thesis the state-of-the-art partitioning methods split the state space along the direction estimated to be most nonlinear. The accuracy of the estimation of the most nonlinear direction was similar to choosing from the main axes the one that causes the most nonlinearity. The algorithms did not estimate how much the nonlinearity is reduced in the split.

An improved algorithm for finding the direction of the maximum nonlinearity was presented in [P2]. The algorithm finds the direction of the maximum nonlinearity of the second order approximation of the measurement function within the prior using an eigendecomposition. The method was found to be more accurate in estimating the direction of the maximum nonlinearity than the algorithms published before it. The splitting direction is used in an algorithm called the Adaptive Splitting (AS) that also quantified the reduction of nonlinearity in the component splitting when the Hessian of the measurement function is assumed constant.

The Binomial Gaussian Mixture Filter (BinoGMF) [P7] extends the use of the nonlinearity information so that the nonlinearity into multiple directions is handled at once in the splitting. Also the quantification of the nonlinearity reduction is used so that the number of required components is minimized to achieve a desired level of nonlinearity. Due to its convergence properties, the BinoGMF is well suited for situations where the number of resulting components in a single direction is high, and the AS is suited better in situations where the Hessian of the measurement function varies a lot within

the prior. The tests show that the splitting schemes presented in this thesis achieve better posterior estimation than other methods.

An idea to consider in the future is the construction of a Gaussian Mixture Filter (GMF) that merges the good sides of the BinoGMF and the AS without increasing the computational burden too much. A straightforward way would be to test for each resulting component of a BinoGMF split if the nonlinearity is still high. This could be possibly done in a computationally lighter way by either computing the nonlinearity only for selected components or by employing the regular structure of Binomial Gaussian Mixtures (BinoGMs). The regular structure could also be exploited in the sigma point based update to reduce computational load. The idea of taking the posterior probability of the components into account [20] could also be merged into the BinoGMF to avoid splitting components that have only a small effect on the posterior.

The BinoGMF makes multidimensional measurement updates by decorrelating the measurement vector first and then updating prior one element of a decorrelated measurement at a time. The order of processing the measurements is not specified and the decorrelation matrix is not unique. The decorrelation matrix could be chosen so that the nonlinearity of the first measurement element is minimized and thus a minimal number of components is required for the first update. The update reduces the component covariance and thus may reduce the nonlinearity of the remaining measurements. If the amount of nonlinearity is reduced also the need of splitting of remaining elements of the measurement function is reduced.

The methods discussed in this thesis can be used to determine whether a measurement function is highly nonlinear and to exploit that information in situations where noise terms are additive and Gaussian. Extending these measures for non-additive and non-Gaussian noise and generating a Gaussian Mixture (GM) model for the noise, if necessary, would be an interesting topic for further research and make the GMFs a competitor for the Particle Filters (PFs) also in that kind of problems. This could employ a statistical approach, similar to that presented in [35], to compute the amount of nonlinearity.

> When building mathematical models of real-world problems, it is crucial to aim for the simplest and most general form that still enables one to draw conclusions about the model's real-world counterpart. Another important concept is to consider a question and its answer separately, for the question might be very simple even when the answer is prohibitively complex or intractable to compute.
>
> Niilo Sirola [69]

The quote above is from the conclusions of Niilo Sirola's PhD thesis, which was also made in our research group. My thesis continues the work on the same problems. The use of linear models instead of nonlinear models can be seen as using models that are further away from the real-world counterpart, but it is easier to find out whether the model contains an answer to a question and what is the answer.

Nonlinear models can be easier to construct than their linear counterparts and they may be superior in accuracy in some cases. However, the existence of noise in real-world situations can make finding the desired answers from the nonlinear models much more difficult than finding the solutions in a noiseless environment. These difficulties can make linear models with easily solved answers superior compared to nonlinear models.

The proposed linear models consider indoor positioning problems. They have unique solutions that are fast to compute. The linear models sacrifice some information for linearity, but combined with the optimal algorithms they perform well in real-world situations. Because the solutions are fast to compute from the linear models, they may be also used as initialization points for solvers using nonlinear models.

The first proposed linear model was used for construction of a radio map autonomously with a Global Positioning System (GPS) and Wireless Local Area Network (WLAN) equipped mobile phone or other device. The linear model produced as good maps as the nonlinear models under realistic noise conditions. In the second proposed model, WLAN Coverage Areas (CAs) were modeled using multiple linear-Gaussian models. The positioning accuracy was in the tests

on a similar level as for the tested nonlinear models and the position estimates were faster to compute from the linear model than from the nonlinear models. In the third model a nonlinear state transition model used in Pedestrian Dead Reckoning (PDR) was replaced with a linear model. The linear model performed especially well when the initial heading was unknown.

In WLAN based Simultaneous Localization and Mapping (SLAM) algorithms sets of nonlinear equations are solved to build the map of the environment [8, 21, 34]. In future it would be an interesting task to develop a linear SLAM based on the work presented in the sections 3.2 and 3.3 and investigate under which conditions it can be used instead of the existing SLAM algorithms.

References

[1] S. Ali-Löytty. On the convergence of the Gaussian mixture filter. Department of Mathematics. Research report 89, Tampere University of Technology, 2008. URL http://urn.fi/URN:NBN:fi:tty-2011041510704.

[2] S. Ali-Löytty. *Gaussian Mixture Filters in Hybrid Positioning*. PhD thesis, Tampere University of Technology, August 2009. URL http://URN.fi/URN:NBN:fi:tty-200905191055.

[3] S. Ali-Löytty. Box Gaussian mixture filter. *IEEE Transactions on Automatic Control*, 55(9):2165–2169, September 2010. doi:10.1109/TAC.2010.2051486.

[4] S. Ali-Löytty and N. Sirola. Gaussian mixture filter and hybrid positioning. In *Proceedings of ION GNSS 2007, Fort Worth, Texas*, pages 562–570, Fort Worth, September 2007.

[5] I. Arasaratnam and S. Haykin. Cubature Kalman filters. *IEEE Transactions on Automatic Control*, 54(6):1254–1269, June 2009. doi:10.1109/TAC.2009.2019800.

[6] S. Beauregard. Omnidirectional Pedestrian Navigation for First Responders. In *4th Workshop on Positioning, Navigation and Communication (WPNC'07)*, pages 33–36, 2007. doi:10.1109/WPNC.2007.353609.

[7] Å. Björck. *Numerical Methods for Least Squares Problems*. Society for Industrial and Applied Mathematics, Philadephia, PA, 1996. doi:DOI:10.1137/1.9781611971484.

[8] L. Bruno and P. Robertson. WiSLAM: improving FootSLAM with WiFi. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, pages 1–10, September 2011. doi:10.1109/IPIN.2011.6071916.

[9] K. Burnham and D. Anderson. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. Springer, 2nd edition, 2002. ISBN 9780387953649.

[10] J. Carpenter, P. Clifford, and P. Fearnhead. Improved particle filter for nonlinear problems. *IEE Proceedings - Radar, Sonar and Navigation*, 146:2–7, February 1999. doi:10.1049/ip-rsn:19990255.

[11] J. Chen, L. Cheng, and M.-G. Gan. Extension of SGMF using gaussian sum approximation for nonlinear/non-gaussian model and its application in multipath estimation. *Acta Automatica Sinica*, 39(1):1 – 10, 2013. ISSN 1874-1029. doi:10.1016/S1874-1029(13)60001-4.

[12] J. Collin, O. Mezentsev, and G. Lachapelle. Indoor positioning system using accelerometry and high accuracy heading sensors. In *Proceedings of the ION GPS/GNSS 2003 Conference*, pages 9–12, 2003.

[13] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3):736–746, March 2002. doi:10.1109/78.984773.

[14] D. Crouse, P. Willett, K. Pattipati, and L. Svensson. A look at Gaussian mixture reduction algorithms. In *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, 2011.

[15] P. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. Bugallo, and J. Miguez. Particle filtering. *Signal Processing Magazine, IEEE*, 20(5):19–38, September 2003. ISSN 1053-5888. doi:10.1109/MSP.2003.1236770.

[16] R. Douc and O. Cappe. Comparison of resampling schemes for particle filtering. In *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis. ISPA 2005*, pages 64–69, September 2005. doi:10.1109/ISPA.2005.195385.

[17] A. Doucet, N. d. Freitas, K. P. Murphy, and S. J. Russell. Rao-blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, UAI '00, pages 176–183, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc. ISBN 1-55860-709-9.

[18] A. Doucet, N. De Freitas, and N. Gordon. *Sequential Monte Carlo methods in practice*. Springer, 2001.

[19] F. Faubel and D. Klakow. Further improvement of the adaptive level of detail transform: Splitting in direction of the nonlinearity. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2010.

[20] F. Faubel, J. McDonough, and D. Klakow. The split and merge unscented Gaussian mixture filter. *IEEE Signal Processing Letters*, 16(9):786 –789, September 2009. doi:10.1109/LSP.2009.2024859.

[21] B. Ferris, D. Fox, and N. Lawrence. WiFi-SLAM using Gaussian process latent variable models. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2480–2485, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[22] D. Fox. Adapting the sample size in particle filters through kld-sampling. *International Journal of Robotics Research*, 22:2003, 2003.

[23] A. Gelb. *Applied optimal estimation*. MIT Press, 1974.

[24] C. Gentner, E. Munoz, M. Khider, E. Staudinger, S. Sand, and A. Dammann. Particle filter based positioning with 3gpp-lte in indoor environments. In *IEEE/ION Position Location and Navigation Symposium (PLANS 2012)*, pages 301–308, April 2012. doi:10.1109/PLANS.2012.6236895.

[25] N. Gordon, D. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. *IEE Proceedings F – Radar and Signal Processing*, 140(2):107–113, 1993.

[26] B. Gozick, K. Subbu, R. Dantu, and T. Maeshiro. Magnetic maps for indoor navigation. *Instrumentation and Measurement, IEEE Transactions on*, 60(12):3883–3891, December 2011. ISSN 0018-9456. doi:10.1109/TIM.2011.2147690.

[27] A. Günther and C. Hoene. Measuring round trip times to determine the distance between WLAN nodes. In *NETWORKING 2005. Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communications Systems*, pages 768–779. Springer, 2005.

[28] U. D. Hanebeck, K. Briechle, and A. Rauh. Progressive bayes: a new framework for nonlinear state estimation. In *Proc. SPIE*, volume 5099, pages 256–267, 2003. doi:10.1117/12.487806.

[29] J. Haverinen and A. Kemppainen. Global indoor self-localization based on the ambient magnetic field. *Robotics and Autonomous Systems*, 57(10):1028 – 1035, 2009. ISSN 0921-8890. doi:10.1016/j.robot.2009.07.018. 5th International Conference on Computational Intelligence, Robotics and Autonomous Systems (5th CIRAS).

[30] F. Havlak and M. Campbell. Discrete and continuous, probabilistic anticipation for autonomous robots in urban environments. *IEEE Transactions on Robotics*, PP(99):1–14, 2013. doi:10.1109/TRO.2013.2291620.

[31] H. Hile and G. Borriello. Positioning and orientation in indoor environments using camera phones. *IEEE Computer Graphics and Applications*, 28(4):32–39, July 2008. ISSN 0272-1716. doi:10.1109/MCG.2008.80.

[32] Y. Ho and R. Lee. A Bayesian approach to problems in stochastic estimation and control. *IEEE Transactions on Automatic Control*, 9(4):333 – 339, October 1964. doi:10.1109/TAC.1964.1105763.

[33] V. Honkavirta, T. Perälä, S. Ali-Löytty, and R. Piché. A comparative survey of WLAN location fingerprinting methods. In *Proceedings of the 6th Workshop on Positioning, Navigation and Communication (WPNC'09)*, pages 243–251, March 2009. doi:10.1109/WPNC.2009.4907834.

[34] J. Huang, D. Millman, M. Quigley, D. Stavens, S. Thrun, and A. Aggarwal. Efficient, generalized indoor wifi GraphSLAM. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1038–1043, May 2011. doi:10.1109/ICRA.2011.5979643.

[35] M. Huber. Adaptive Gaussian mixture filter based on statistical linearization. In *Proceedings of the 14th International Conference on Information Fusion (FUSION)*, pages 1–8, July 2011.

[36] V. Huttunen and R. Piché. A monocular camera gyroscope. *Gyroscopy and Navigation*, 3(2):124–131, 2012. doi:10.1134/S2075108712020046.

[37] A. H. Jazwinski. *Stochastic Processes and Filtering Theory*, volume 64 of *Mathematics in Science and Engineering*. Academic Press, 1970.

[38] R. Jirawimut, P. Ptasinski, V. Garaj, F. Cecelja, and W. Balachandran. A method for dead reckoning parameter correction in pedestrian navigation system. *IEEE Transactions on Instrumentation and Measurement*, 52(1):209–215, 2003. doi:10.1109/TIM.2002.807986.

[39] S. J. Julier and J. K. Uhlmann. A New Extension of the Kalman Filter to Nonlinear Systems. In *International Symposium Aerospace/Defense Sensing, Simulation and Controls*, volume 3068, pages 182–193, 1997. doi:10.1117/12.280797.

[40] K. Kaemarungsi and P. Krishnamurthy. Properties of indoor received signal strength for WLAN location fingerprinting. In *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services. MOBIQUITOUS 2004.*, pages 14 – 23, August 2004. doi:10.1109/MOBIQ.2004.1331706.

[41] K. Kaji and N. Kawaguchi. Design and implementation of WiFi indoor localization based on Gaussian mixture model and particle filter. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN 2012)*, pages 1–9, November 2012. doi:10.1109/IPIN.2012.6418943.

[42] R. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[43] L. Koski, R. Piché, V. Kaseva, S. Ali-Löytty, and M. Hännikäinen. Positioning with coverage area estimates generated from location fingerprints. In *Proceedings of the 7th Workshop on Positioning, Navigation and Communication (WPNC'10)*, pages 99–106, Dresden Germany, March 2010. doi:10.1109/WPNC.2010.5653409.

[44] S. Kullback and R. A. Leibler. On information and sufficiency. *Ann. Math. Statist.*, 22(1):79–86, 1951.

[45] H. Leppäkoski, J. Collin, and J. Takala. Pedestrian navigation based on inertial sensors, indoor map, and WLAN signals. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1569–1572, 2012. doi:10.1109/ICASSP.2012.6288192.

[46] M. Mata, J. Armingol, A. De la Escalera, and M. Salichs. A visual landmark recognition system for topological navigation of mobile robots. In *IEEE International Conference on Robotics and Automation, 2001*, volume 2, pages 1124–1129 vol.2, 2001. doi:10.1109/ROBOT.2001.932762.

[47] J. Medbo, I. Siomina, A. Kangas, and J. Furuskog. Propagation channel impact on lte positioning accuracy: A study based on real measurements of observed time difference of arrival. In *20th International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 2213–2217, September 2009. doi:10.1109/PIMRC.2009.5450144.

[48] R. V. D. Merwe, A. Doucet, N. D. Freitas, and E. Wan. The unscented particle filter, 2000.

[49] P. Müller, S. Ali-Löytty, M. Dashti, H. Nurminen, and R. Piché. Gaussian mixture filter allowing negative weights and its application to positioning using signal strength measurements. In *9th Workshop on Positioning Navigation and Communication (WPNC'12)*, pages 71–76, March 2012. doi:10.1109/WPNC.2012.6268741.

[50] P. Müller, H. Wymeersch, and R. Piché. UWB positioning with generalized Gaussian mixture filters. *IEEE Transactions on Mobile Computing*, 13(10):2406–2414, October 2014. ISSN 1536-1233. doi:10.1109/TMC.2014.2307301.

[51] H. Nurminen, J. Talvitie, S. Ali-Löytty, P. Müller, E.-S. Lohan, R. Piché, and M. Renfors. Statistical path loss parameter estimation and positioning using RSS measurements. In *Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS2012)*, pages 1–8, October 2012. doi:10.1109/UPINLBS.2012.6409754.

[52] H. Nurminen, J. Talvitie, S. Ali-Löytty, P. Müller, E.-S. Lohan, R. Piché, and M. Renfors. Statistical path loss parameter estimation and positioning using RSS measurements in indoor wireless networks. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN 2012)*, pages 1–9, November 2012. doi:10.1109/IPIN.2012.6418856.

[53] H. Nurminen, A. Ristimaki, S. Ali-Löytty, and R. Piché. Particle filter and smoother for indoor localization. In *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, pages 1–10, October 2013. doi:10.1109/IPIN.2013.6817903.

[54] R. Piché. Robust estimation of a reception region from location fingerprints. In *International Conference on Localization and GNSS (ICL-GNSS)*, Tampere, June 2011. doi:10.1109/ICL-GNSS.2011.5955261.

[55] M. K. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filters. *Journal of the American Statistical Association*, 94(446):590–599, 1999. doi:10.1080/01621459.1999.10474153.

[56] D. Porcino. Performance of a OTDOA-IPDL positioning receiver for 3GPP-FDD mode. In *Second International Conference on 3G Mobile Communication Technologies*, pages 221–225, 2001. doi:10.1049/cp:20010045.

[57] J. Prieto, A. Bahillo, S. Mazuelas, R. Lorenzo, J. Blas, and P. Fernandez. Adding indoor location capabilities to an IEEE 802.11 WLAN using real-time RTT measurements. In *Wireless Telecommunications Symposium, 2009. WTS 2009*, pages 1–7, April 2009. doi:10.1109/WTS.2009.5068948.

[58] M. Raitoharju, H. Nurminen, and P. Robert. Kalman filter with a linear state model for PDR+WLAN positioning and its application to assisting a particle filter. *EURASIP Journal on Advances in Signal Processing*, 2014. Under review.

[59] J. Rantakokko, J. Rydell, P. Stromback, P. Handel, J. Callmer, D. Tornqvist, F. Gustafsson, M. Jobs, and M. Gruden. Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization. *IEEE Wireless Communications*, 18(2):10–18, April 2011. ISSN 1536-1284. doi:10.1109/MWC.2011.5751291.

[60] H. E. Rauch, C. T. Striebel, and F. Tung. Maximum Likelihood Estimates of Linear Dynamic Systems. *Journal of the American Institute of Aeronautics and Astronautics*, 3(8):1445–1450, August 1965.

[61] A. Rauh, K. Briechle, and U. Hanebeck. Nonlinear measurement update and prediction: Prior density splitting mixture estimator. In *Control Applications, (CCA) Intelligent Control, (ISIC), 2009 IEEE*, pages 1421–1426, July 2009. doi:10.1109/CCA.2009.5281167.

[62] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, Boston, MA USA, January 2004.

[63] P. Robertson, M. Angermann, and B. Krach. Simultaneous localization and mapping for pedestrians using only foot-mounted

inertial sensors. In *Proceedings of the 11th International Conference on Ubiquitous Computing*, Ubicomp '09, pages 93–96, New York, NY, USA, 2009. ACM. doi:10.1145/1620545.1620560.

[64] P. Robertson, M. Angermann, and B. Krach. Simultaneous localization and mapping for pedestrians using only foot-mounted inertial sensors. In *Proceedings of the 11th International Conference on Ubiquitous Computing*, Ubicomp '09, pages 93–96, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-431-7. doi:10.1145/1620545.1620560.

[65] M. Roth and F. Gustafsson. An efficient implementation of the second order extended Kalman filter. In *Proceedings of the 14th International Conference on Information Fusion (FUSION 2011)*, pages 1–6, July 2011.

[66] A. Runnalls. Kullback-Leibler approach to Gaussian mixture reduction. *IEEE Transactions on Aerospace and Electronic Systems*, 43(3):989–999, 2007. doi:10.1109/TAES.2007.4383588.

[67] S. Särkkä. *Bayesian filtering and smoothing*, volume 3. Cambridge University Press, 2013.

[68] S. H. Shin, C. Park, J. W. Kim, H. Hong, and J. M. Lee. Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors. In *IEEE Sensors Applications Symposium, 2007. SAS '07*, pages 1–5, 2007. doi:10.1109/SAS.2007.374406.

[69] N. Sirola. *Mathematical Methods for Personal Positioning and Navigation*. PhD thesis, Tampere University of Technology, August 2007. URL `http://URN.fi/URN:NBN:fi:tty-200810021113`.

[70] N. Sirola. Closed-form algorithms in mobile positioning: Myths and misconceptions. In *Proceedings of the 7th Workshop on Positioning, Navigation and Communication (WPNC'10)*, pages 38–44, Dresden Germany, March 2010. doi:10.1109/WPNC.2010.5653789.

[71] H. W. Sorenson and D. L. Alspach. Recursive Bayesian estimation using Gaussian sums. *Automatica*, 7(4):465–479, 1971. doi:10.1016/0005-1098(71)90097-5.

[72] A. Soto. Self adaptive particle filter. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, IJCAI'05, pages 1398–1403, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

[73] S. Särkkä. On Unscented Kalman filtering for state estimation of continuous-time nonlinear systems. *IEEE Transactions on Automatic Control*, 52(9):1631–1641, September 2007. doi:10.1109/TAC.2007.904453.

[74] G. Terejanu, P. Singla, T. Singh, and P. Scott. Adaptive gaussian sum filter for nonlinear bayesian estimation. *IEEE Transactions on Automatic Control*, 56(9):2151–2156, September 2011. ISSN 0018-9286. doi:10.1109/TAC.2011.2141550.

[75] S. Thrun and M. Montemerlo. The Graph SLAM algorithm with applications to large-scale mapping of urban structures. *Int. J. Rob. Res.*, 25(5-6):403–429, May 2006. ISSN 0278-3649. doi:10.1177/0278364906065387.

[76] E. Wan and R. Van der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the Adaptive Systems for Signal Processing, Communications, and Control Symposium. AS-SPCC*, pages 153–158, 2000. doi:10.1109/ASSPCC.2000.882463.

[77] H. Wang, H. Lenz, A. Szabo, J. Bamberger, and U. Hanebeck. Wlan-based pedestrian tracking using particle filters and low-cost mems sensors. In *Positioning, Navigation and Communication, 2007. WPNC '07. 4th Workshop on*, pages 1–7, March 2007. doi:10.1109/WPNC.2007.353604.

[78] L. Wirola. *Studies on location technology standards evolution in wireless networks*. PhD thesis, Tampere University of Technology, 2010. URL `http://urn.fi/URN:NBN:fi:tty-201002121065`.

[79] L. Wirola, L. Wirola, and R. Piché. Bandwidth and storage reduction of radio maps for offline WLAN positioning. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013)*, 2013. URL `http://urn.fi/URN:NBN:fi:tty-201403051120`.

[80] O. J. Woodman. An introduction to inertial navigation. Technical Report UCAM-CL-TR-696, University of Cambridge, Computer Laboratory, August 2007.

[81] L. Yu, D. Kai, W. Haipeng, L. Jun, H. You, and P. Lina. Adaptive Gaussian sum squared-root cubature kalman filter with split-merge scheme for state estimation. *Chinese Journal of Aeronautics*, 2014. ISSN 1000-9361. doi:10.1016/j.cja.2014.09.007. In press.

[82] S. Zvanovec, P. Pechac, and M. Klepal. Wireless LAN network design: site survey or propagation modeling? *Journal of Radioengineering, Czech Technical University*, 12(4):44–49, 2003.

PUBLICATION 1

Matti Raitoharju, Simo Ali-Löytty, and Lauri Wirola: Estimation of Base Station Position Using Timing Advance Measurements. In *Proceedings of the International Conference on Signal and Information Processing (ICSIP)*, pages 182–186, Changsha, China, December 2010.

Estimation of Base Station Position Using Timing Advance Measurements

Matti RAITOHARJU, Simo ALI-LÖYTTY
Tampere University of Technology
Finland
matti.raitoharju@tut.fi, simo.ali-loytty@tut.fi

Lauri WIROLA
Nokia Inc.
lauri.wirola@nokia.com

***Abstract*—Timing Advance is used in TDMA (Time Division Multiple Access) systems, such as GSM and LTE, to synchronize the mobile phone to the cellular BS (Base Station). Mobile phone positioning can use TA measurements if BS positions are known, but in many cases BS positions are not in the public domain. In this work we study how to use a set of TA measurements taken by mobile phones at known positions to estimate the position of a BS. This paper describes two methods -- GMF (Gaussian Mixture Filter) and PMF (Point Mass Filter) for estimation of the BS position. Positioning performance is evaluated using simulated and real measurements. In suburban field tests, TA measurements suffice to determine BS position with an error comparable to the TA granularity (550m). GMF computes BS position much faster than PMF and is only slightly less accurate.**

Keywords - Timing advance; Nonlinear filters; Mobile communication; Position measurement; Positioning; GSM;

I. INTRODUCTION

Mobile phone positioning may be done, for example, using GNSS (Global Navigation Satellite System) or using information of cellular network. Many mobile phones have an integrated GNSS receiver and in most cases GNSS position estimate is more accurate than positioning based on the network information. However, many low-end mobile phones do not have integrated GNSS receiver. Even for GNSS-equipped phones, cell based positioning has some advantages to GNSS. First position fix in GNSS may take a while or may be impossible due to signal blocking. Also the cellular positioning accuracy is good enough for some applications, for example, local weather information. Benefits of cellular methods in these cases compared to GNSS methods are the much lower power consumption, faster initial position fix and better availability.

When the serving cell's geographical coverage area is known, the position of a mobile phone may be estimated e.g. as the coverage area centroid. If the handset is connected to several networks or can hear several BSs (Base Stations) the position can be estimated by consideration of the overlapping coverage areas [1]. In addition to the cell identity other network parameters may be used in positioning e.g. received signal strength, TA (Timing Advance) in GSM (Global System for Mobile Communications), LTE (Long Term Evolution) and TD-SCDMA (Time Division Synchronous Code Division Multiple Access) networks, AoA (Angle of Arrival), TDOA (Time Difference of Arrival), among others. For a comprehensive study of these methods see [2].

If the handset has WLAN (Wireless Local Area Network) capabilities, the cell positioning may be done exploiting heard WLAN stations. WLAN BS has a significantly smaller range than GSM BS, and this leads to better positioning accuracy. Problems with WLAN cell positioning are the scarcity of WLANs in rural areas and the possibility that WLAN BSs may be mobile. Mapping of personal WLAN access points also raises privacy issues.

TA can be used to estimate the distance of mobile phone to BS and these range measurements can be used for mobile phone positioning, either standalone or combined with other measurements such as cell id [3]. TA-based mobile phone positioning can be appropriate in rural settings where cells are large and for some reason GNSS is not available. However, for the method to work, it is necessary to know the BS locations, and this information is not always in the public domain. For this reason, there is interest in methods to determine locations of BSs.

In this paper we show how BS position can be estimated from a set of TA measurements taken by GNSS-equipped mobile phones. In the next section the TA is presented. In Section III methods for BS position estimation are presented. Results are presented in Section IV and future work in Section V. Section VI concludes the article.

II. TIMING ADVANCE

TA is the RTT (Round Trip Time) from mobile phone to BS. It is specified in [4] and is used to minimize interference in TDMA (Time-Division Multiple Access) systems. When multiple mobile phones are sending on the same physical channel they need to know the right time to send so that data arrives to BS antenna on the right timeslot. The mobile phone and BS do initial synchronization on RACH (Random Access Channel) using zero timing advance. After this the BS tells the mobile phone how much transmission has to be advanced. Granularity of TA is a GSM bit (48/13μs). TA is not always available in mobile phone e.g. when the radio link is in idle-state TA is not available.

TA measurement may be transformed to a discrete distance measurement with granularity of $\Delta_{TA}=c\cdot 24/13\mu s\approx$ 550m. In LTE networks TA will have much finer granularity of $16/(15000\cdot 2048)$s that corresponds to ≈78m in range [5, 6], but because the LTE networks and devices are not currently widespread we here will concentrate on GSM TA.

A. Ideal Model

The discrete TA measurement presented above may be also written as

$$\mathrm{TA} = \mathrm{floor}\left(\frac{\left\|\mathbf{x}^{\mathrm{M}} - \mathbf{x}^{\mathrm{BS}}\right\|}{\Delta_{\mathrm{TA}}}\right), \quad (1)$$

where $\mathbf{x}^{\mathrm{M}}$ is the position of the mobile phone and $\mathbf{x}^{\mathrm{BS}}$ is the BS position. In this paper we assume that $\mathbf{x}^{\mathrm{M}}$ is known (it is computed using mobile phone GNSS). Based on this model we can write the likelihood of the ideal TA measurement

$$p(\mathrm{TA} \mid \mathbf{x}^{\mathrm{BS}}) = \begin{cases} 1, & 0 \le \left\|\mathbf{x}^{\mathrm{M}} - \mathbf{x}^{\mathrm{BS}}\right\| - \mathrm{TA}\Delta_{\mathrm{TA}} < \Delta_{\mathrm{TA}} \\ 0, & \text{otherwise} \end{cases}. \quad (2)$$

B. Measurement Model from Real Data

In practice the ideal model presented above does not hold because of errors. TA measurement may have different values at same distance from the BS or even on the same spot at different times. Some TA values measured in different locations and the position of the BS are shown in Fig. 1.

In this paper we assume that the real measurement model (likelihood) is rotationally symmetric. Using this assumption it is enough to approximate from the real data one-dimensional likelihood $p(\mathrm{TA}|r)$, where $r=\|\mathbf{x}^{\mathrm{M}}-\mathbf{x}^{\mathrm{BS}}\|$. Note that the ideal likelihood $p(\mathrm{TA}|r)$ (see (2)) is proportional to pdf (probability density function) of uniform distribution

$$\mathrm{Uniform}\left(\mathrm{TA}\Delta_{\mathrm{TA}}, (\mathrm{TA}+1)\Delta_{\mathrm{TA}}\right), \quad (3)$$

whose mean is $(2\mathrm{TA}\Delta_{\mathrm{TA}}+\Delta_{\mathrm{TA}})/2$ and variance is $(\Delta_{\mathrm{TA}})^2/12$.

One option to approximate likelihood is to use following algorithm. First the n measurements are sorted in ascending order by the distance to the BS. The weight of each measurement is then set to

$$w_i = \begin{cases} k_{\mathrm{TA}_i} d_2 & , i = 1 \\ k_{\mathrm{TA}_i} (d_{i+1} - d_{i-1}) & , 1 < i < n \\ 2k_{\mathrm{TA}_i} (d_i - d_{i-1}) & , i = n \end{cases}, \quad (4)$$

where k_{TA_i} are such that $\sum_{\mathrm{TA}_i = ta} w_i = 1$ for all received TA values (ta) and d_i is the distance from a measurement to the BS. Now we can approximate, that

$$p(\mathrm{TA} = ta \mid r) \propto \mathrm{N}_{\sigma_{ta}^2}^{\mu_{ta}}(r), \quad (5)$$

where $\mathrm{N}_{\sigma_{\mathrm{ta}}^2}^{\mu_{\mathrm{ta}}}(r)$ is pdf of the one-dimensional Gaussian distribution $\mathrm{N}(\mu, \sigma^2)$

$$\mu_{ta} = \sum_{\mathrm{TA}_i = ta} w_i d_i \text{ and} \quad (6)$$

$$\sigma_{ta}^2 = \sum_{\mathrm{TA}_i = ta} w_i (d_i - \mu_{ta})^2. \quad (7)$$

We consider two different Gaussian models for TA measurement. In the first model (5) the mean and variance are fitted for each separate TA measurement value independently. In the second model we subtract $ta\Delta_{\mathrm{TA}}$ from each distance and use just a single measurement model for all different TA values. In positioning the variance σ_{ta}^2 is the same for all TA measurements and the means are

$$\mu_{\mathrm{ta}} = \mu_0 + ta\Delta_{\mathrm{TA}}. \quad (8)$$

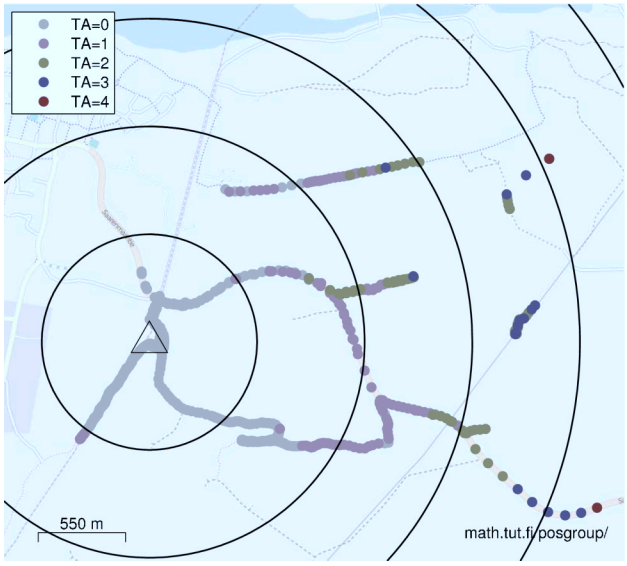

Figure 1. TA-measurements from one base station in different positions. The circles are based on the ideal model (2).

III. Base Station Position Estimation

In following sections two different methods for estimating BS position are presented. The first method is PMF (Point Mass Filter) [7, 8] that is a quite simple and converges to the exact posterior distribution as the number of computation points increases. The second method is GMF (Gaussian Mixture Filter) [9-12] that estimates the probability distribution using multiple Gaussians and is faster than PMF [13]. Both of these filters, PMF and GMF are Bayesian filters, whose aim is to compute posterior distribution of the BS position $p(\mathbf{x}^{\mathrm{BS}}|\mathrm{TA}_{1:n})$, given all available TA measurements $\mathrm{TA}_{1:n}=\{\mathrm{TA}_1,\mathrm{TA}_2,\ldots,\mathrm{TA}_n\}$. We assume:

- TA measurements are conditionally independent given the $\mathbf{x}^{\mathrm{BS}}$
- BS position is static
- The initial prior distribution $p\left(\mathbf{x}_{t_0}^{\mathrm{BS}} \mid \mathrm{TA}_{1:0}\right) = p\left(\mathbf{x}_{t_0}^{\mathrm{BS}}\right)$ is uniform
- Local 2D coordinate system is used
- First measurement is at the origin of the coordinate system
- Each measurement position has zero error (i.e. position error has been incorporated into the TA measurement model)

The posterior distribution can be computed recursively:

$$\text{Prediction: } p\left(\mathbf{x}_{t_k}^{\text{BS}} \mid \text{TA}_{1:k-1}\right) = p\left(\mathbf{x}_{t_{k-1}}^{\text{BS}} \mid \text{TA}_{1:k-1}\right) \quad (9)$$

$$\text{Update: } p\left(\mathbf{x}_{t_k}^{\text{BS}} \mid \text{TA}_{1:k}\right) \propto p\left(\text{TA}_k \mid \mathbf{x}_{t_k}^{\text{BS}}\right) p\left(\mathbf{x}_{t_k}^{\text{BS}} \mid \text{TA}_{1:k-1}\right) \quad (10)$$

A. Point Mass Filter

PMF approximates posterior distributions using convex combinations of delta functions (point masses)

$$p(\mathbf{x}^{\text{BS}} \mid \text{TA}_{1:k}) = \sum_{j=1}^{N_{\text{PMF}}} \omega_j \delta(\mathbf{x}^{\text{BS}} - \mathbf{x}_j^{\text{PMF}}), \quad (11)$$

where points $\mathbf{x}_j^{\text{PMF}}$ are deterministically set. Here $\omega_j \geq 0$ and $\sum_{j=1}^{N_{\text{PMF}}} \omega_i = 1$. In this paper points are chosen on an equispaced grid.

B. Gaussian Mixture Filter

GMF approximates posterior distributions using convex combinations of Gaussian distributions:

$$p(\mathbf{x}^{\text{BS}} \mid \text{TA}_{1:k}) = \sum_{j=1}^{N_{\text{GMF}}} \alpha_j \text{N}_{\text{P}_j}^{\mu_j}(\mathbf{x}^{\text{BS}}), \quad (12)$$

where $\text{N}_{\text{P}_j}^{\mu_j}(\mathbf{x}^{\text{BS}})$ is pdf of 2D Gaussian distribution $\text{N}(\mu,\text{P})$. Here $\alpha_j \geq 0$ and $\sum_{j=1}^{N_{\text{GMF}}} \alpha_i = 1$. We use abbreviation

$$\text{M}(\alpha_j, \mu_j, \text{P}_j)_{(j, N_{\text{GMF}})} \quad (13)$$

to present the full GMF. Algorithm of general GMF is given below.

Algorithm 1 Gaussian Mixture Filter

Initialization of GMF at time t_1 (Sec. III-B1):

$$\text{M}(\alpha_{j,1}, \mu_{j,1}, \text{P}_{j,1})_{(j, N_{\text{GMF},1})}$$

for $k = 2$ to n **do**

1. *Prediction:*

$$\text{M}(\alpha_{j,k}^-, \mu_{j,k}^-, \text{P}_{\text{j,k}}^-)_{(j, N_{\text{GMF}^-,\text{k}})} :=$$

$$\text{M}(\alpha_{j,k-1}, \mu_{j,k-1}, \text{P}_{\text{j,k-1}})_{(j, N_{\text{GMF}^-,\text{k-1}})}$$

2. *Update:* Compute posterior approximation using a bank of EKFs (Sec. III-B2)

$$\text{M}(\alpha_{j,k}^+, \mu_{j,k}^+, \text{P}_{\text{j,k}}^+)_{(j, N_{\text{GMF}^+,\text{k}})}$$

3. *Reduce the number of components: (Sec. III-B3)*

$$\text{M}(\alpha_{j,k}, \mu_{j,k}, \text{P}_{\text{j,k}})_{(j, N_{\text{GMF}\ ,\text{k}})}$$

endfor

A specific GMF for BS position estimation is explained next:

1) Initialization of GMF:

The initialization is done using the first TA measurement. The idea is to compute Gaussian mixture approximation of the posterior distribution at time t_1. Because the initial prior distribution is uniform, we can compute Gaussian mixture approximation of the first likelihood function $p(\text{TA}_1|\mathbf{x}^{\text{BS}})$. In this case we use

$$\text{M}(\alpha_{j,1}, \mu_{j,1}, \text{P}_{j,1})_{(j, N_{\text{GMF},1})}, \quad (14)$$

where

$$\alpha_{j,1} = \frac{1}{N_{\text{GMF},1}}$$

$$\mu_{j,1} = \text{Q}_j \frac{\int_A \mathbf{x}^{\text{BS}} p(\text{TA}_1 \mid \mathbf{x}^{\text{BS}}) d\mathbf{x}^{\text{BS}}}{\int_A p(\text{TA}_1 \mid \mathbf{x}^{\text{BS}}) d\mathbf{x}^{\text{BS}}} \approx \text{Q}_j \begin{bmatrix} \mu_{ta} \\ 0 \end{bmatrix}$$

$$\text{P}_{j,1} = \text{Q}_j \frac{\int_A (\mathbf{x}^{\text{BS}} - \mu_{j,1})(\mathbf{x}^{\text{BS}} - \mu_{j,1})^T p(\text{TA}_1 \mid \mathbf{x}^{\text{BS}}) d\mathbf{x}^{\text{BS}}}{\int_A p(\text{TA}_1 \mid \mathbf{x}^{\text{BS}}) d\mathbf{x}^{\text{BS}}} \text{Q}_j^T$$

$$\text{Q}_j = \begin{bmatrix} \cos\frac{2\pi j}{N_{\text{GMF},1}} & \sin\frac{2\pi j}{N_{\text{GMF},1}} \\ -\sin\frac{2\pi j}{N_{\text{GMF},1}} & \cos\frac{2\pi j}{N_{\text{GMF},1}} \end{bmatrix}$$

$$A = \left\{ \mathbf{x} \middle| \frac{\mathbf{x}_1}{\|\mathbf{x}\|} \geq \cos\left(\frac{\pi}{N_{\text{GMF},1}}\right) \right\}.$$

In our implementation the number of components used is $N_{\text{GMF},1}$ and $\text{P}_{j,1}$ is computed numerically.

2) Update:

Here we convert likelihood to the range measurement model (for simplicity we suppress the subscript k)

$$\mu_{ta_j} = \text{h}_j(\mathbf{x}^{\text{BS}}) + \mathbf{v}_j = \left\| \mathbf{x}^{\text{BS}} + \mathbf{x}_j^{\text{M}} \right\| + \mathbf{v}_j, \quad (15)$$

where $\mathbf{v} \sim \text{N}(0, \sigma_{ta_j}^2)$. Now the posterior approximation is

$$\text{M}(\alpha_j^+, \mu_j^+, \text{P}_{\text{j}}^+)_{(j, N_{\text{GMF}^+})}, \quad (16)$$

where

$$\text{H}_j = \begin{cases} \dfrac{(\mu_j^- - \mathbf{x}^{\text{M}})^T}{\left\| \mu_j^- - \mathbf{x}^{\text{M}} \right\|}, & \text{if } \left\| \mu_j^- - \mathbf{x}^{\text{M}} \right\| \neq 0 \\ [0\,0], & \text{otherwise} \end{cases}$$

$$\text{K}_j = \frac{1}{\text{H}_j \text{P}_j^- \text{H}_j^T} \text{P}_j^- \text{H}_j^T$$

$$\alpha_j^+ \propto \alpha_j^- e^{-\frac{(\mu_{ta_j} - \text{h}_j(\mu_j^-))^2}{2(\text{H}_j \text{P}_j^- \text{H}_j^T + \sigma_{ta_j}^2)}}$$

$$\mu_j^+ = \mu_j^- + \mathrm{K}_j(\mu_{ta_j} - \mathrm{h}_j(\mu_j^-))$$
$$\mathrm{P}_j^+ = (\mathrm{I} - \mathrm{K}_j\mathrm{H}_j)\mathrm{P}_j^-.$$

After the above computations are done for each component, the weights are normalized so that $\sum_{j=1}^{N_{\mathrm{GMF}+}} \alpha_j^+ = 1$.

3) Reduce the Number of Components:

Components are reduced using two different methods [11, 14, 15]. In the first method, components having a weight α_j^+ less than some threshold are simply dropped and the weights of the other components are renormalized. In the second method components are merged, if they are close to each other i.e. $\left\|\mu_i^+ - \mu_j^+\right\| < \theta$, where θ is the component merge distance. The parameters for the new merged component may be written as follows

$$\mu_l = \frac{\alpha_i^+\mu_i^+ + \alpha_j^+\mu_j^+}{\alpha_i^+ + \alpha_j^+} \tag{17}$$

$$\mathrm{P}_l = \frac{\alpha_i^+}{\alpha_i^+ + \alpha_j^+}\left[\mathrm{P}_i^+ + (\mu_i^+ - \mu_l)(\mu_i^+ - \mu_l)^T\right] + \frac{\alpha_j^+}{\alpha_i^+ + \alpha_j^+}\left[\mathrm{P}_j^+ + (\mu_j^+ - \mu_l)(\mu_j^+ - \mu_l)^T\right] \tag{18}$$

$$\alpha_l = \alpha_i^+ + \alpha_j^+ \tag{19}$$

IV. Results

A. Model Fitting

In Fig. 2 the cumulative probability distributions of the measurements shown in Fig. 1 are plotted. Compared to the ideal model the TA measurements are clearly expanded to larger areas and have means close to upper limit of the ideal model.

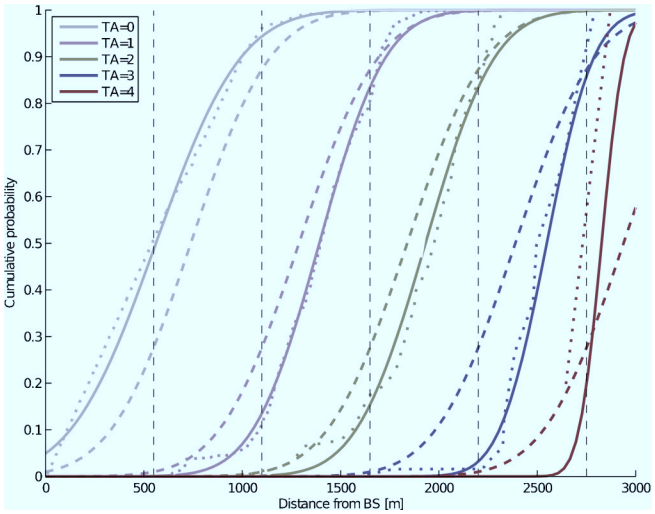

Figure 2. Fitted CDF functions. Solid line is the Gaussian fit with separate mean and variance for each TA, dashed line is a single model fit, dotted line is the original cumulative density function and vertical dashed lines present the limits of the ideal model

B. BS Positioning

BS positioning is done using the PMF and GMF. The component merge distance is θ=100m and the component discard value is 10^{-6}.

1) Simulated Results:

In simulations 20-100 TA measurements are simulated using likelihood

$$p(\mathrm{TA} \mid \mathbf{x}^{\mathrm{BS}}) = \begin{cases} 1, & 0 \le \left\|\mathbf{x}^{\mathrm{M}} - \mathbf{x}^{\mathrm{BS}}\right\| - \mathrm{TA}\Delta_{\mathrm{TA}} < \Delta_{\mathrm{TA}} \\ e^{-\frac{\left|\left\|\mathbf{x}^{\mathrm{M}} - \mathbf{x}^{\mathrm{BS}}\right\| - \frac{\mathrm{TA}\Delta_{\mathrm{TA}}}{2}\right|}{2\Delta_{\mathrm{TA}}^{ta}}}, & \text{otherwise} \end{cases}, \tag{20}$$

where $\Delta_{\mathrm{TA}}^{ta} \ge \Delta_{\mathrm{TA}}$ is a width parameter that expands some TA measurements more than others. Exponential tails are also used to model measurement errors.

Measurement models for GMF are fitted using 1000 samples generated using random measurement positions and the likelihood above. The ideal model uses the mean and variance of (3). Single model is done using (8) and multiple models are computed with (6) and (7).

Table I presents the simulation results. The reference solution is computed using PMF with the optimal model and 90000 points (300×300 grid).

TABLE I. Simulated BS Positioning Results

Filter	Mean error [m]	Mean reference error [m]
Ideal model GMF	442	414
Single model GMF	349	322
Multiple model GMF	160	84
PMF	141	0

2) Real BS Positioning

The real measurements were collected in a suburban area using a Nokia mobile phone that was calling all the time to get the TA measurements.

In Fig. 3 the grid filter and GMF are compared. The probability distribution of estimated BS position is shown in red. The blue triangle is the true BS position. In this case the likelihood given in (20) is used with $\Delta_{\mathrm{TA}}^{ta}$ = 550m for all TA measurement values and the GMF measurement model means were the ideal means but variance 4 times bigger. Fitted models are not used in this case to prevent filters from having any extra information of the BS position.

V. Future Work

In future the effect of base station position in user positioning should be evaluated. The effect of using TA measurement should be compared to cell model positioning without TA measurement.

More measurements from different BSs would be needed to make a good model of real TA measurements and use that

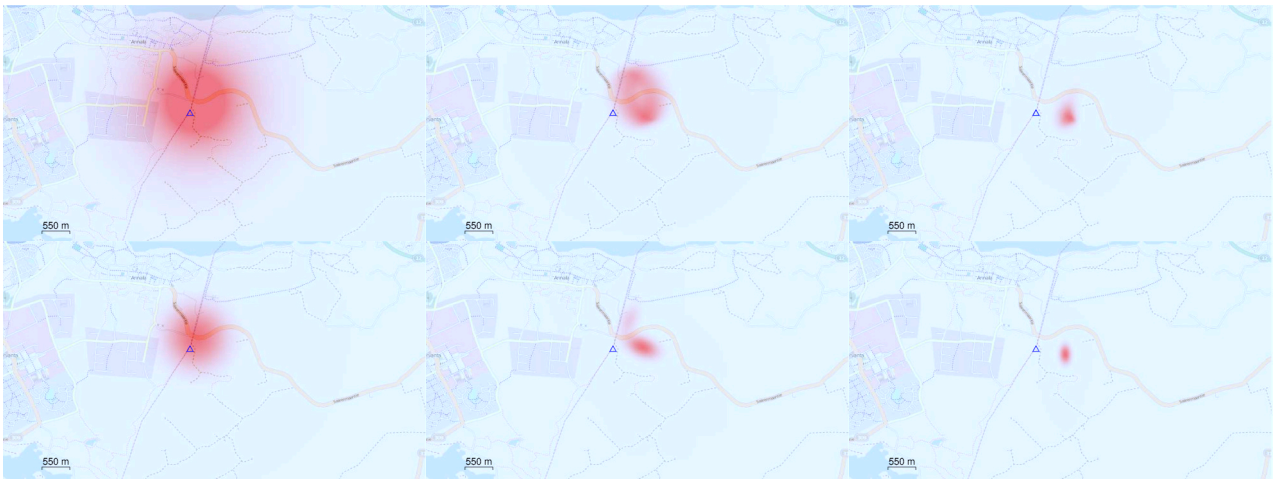

Figure 3. PMF (top) and GMF (bottom) distributions after 1, 10 and 30 real measurements

model in BS positioning case that is independent of the modeling case. Also a measurement model should cover all 63 TA values –we measured only 5 values. A measurement campaign should be done in rural environment to get measurements with higher TA values.

Other things to do in future might be to account for the fact that most of cells are not round but more like sectors and to investigate the error models further.

VI. CONCLUSIONS

This paper showed that the BS position may be solved using TA measurements from known locations. Simulated results show that the accuracy of GMF is close to ideal when the measurement errors are well modeled. On other hand the real measurement case showed that even with a crude model the BS position may be estimated with accuracy close to the granularity of TA measurements.

One problem with TA measurements is that they are not available when the radio link is in idle state. To avoid this phone may open a data connection before the positioning. Another issue is that the UMTS (Universal Mobile Telecommunications System) networks using WCDMA (Wideband Code Division Multiple Access) measure RTT, but the network does not provide RTT to mobile phone. So this feature can be used only on the network side.

ACKNOWLEDGMENT

The authors wish to thank Professor Robert Piché for guidance and comments. Nokia Inc. funded this paper.

REFERENCES

[1] L. Koski, R. Piché, V. Kaseva, S. Ali-Löytty, and M. Hännikäinen, "Positioning with coverage area estimates generated from location fingerprints," in *Proceedings of the 7th Workshop on Positioning, Navigation and Communication 2010 (WPNC'10)*, 2010, (accepted).

[2] L. Wirola, "Studies on location technology standards evolu- tion in wireless networks," Ph.D. dissertation, Tampere Uni- versity of Technology, March, http://URN.fi/URN:NBN:fi:tty-201002121065.

[3] M. Spirito, S. Poykko, and O. Knuuttila, "Experimental performance of methods to estimate the location of legacy handsets in gsm," *Vehicular Technology Conference, 2001. VTC 2001 Fall. IEEE VTS 54th*. no. 54ND, 2001, pp. 2716 –2720 vol.4.

[4] http://www.3gpp.org, "3GPP TS 45.010 Radio Subsystem Synchronization."

[5] ——, "3GPP TS 25.211 E-UTRA Physical channels and modulation ."

[6] ——, "3GPP TS 36.213 E-UTRA Physical layer procedures."

[7] N. Bergman, "Bayesian Inference in Terrain Navigation," Licenciate Thersis, Linköping University, 1997, thesis No. 649.

[8] R. S. Bucy and K. D. Senne, "Digital synthesis of non-linear filters," *Automatica*, vol. 7, no. 3, pp. 287–298, May 1971.

[9] D. L. Alspach and H. W. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 439–448, Aug 1972.

[10] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*, ser. Electrical Engineering, T. Kailath, Ed. Prentice-Hall, Inc., 1979.

[11] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, July 1971.

[12] S. Ali-Löytty, "Gaussian mixture filters in hybrid positioning," Ph.D. dissertation, Tampere University of Technology, August 2009, http://URN.fi/URN:NBN:fi:tty-200905191055.

[13] N. Sirola, S. Ali-Löytty, and R. Piché, "Benchmarking nonlinear filters," in *Nonlinear Statistical Signal Processing Workshop NSSPW06*, Cambridge, September 2006.

[14] S. Ali-Löytty and N. Sirola, "Gaussian mixture filter in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2007*, 2007.

[15] D. J. Salmond, "Mixture reduction algorithms for target tracking," *State Estimation in Aerospace and Tracking Applications, IEE Colloquium on*, pp. 7/1–7/4, 1989.

PUBLICATION 2

Matti Raitoharju and Simo Ali-Löytty: An Adaptive Derivative Free Method for Bayesian Posterior Approximation. *IEEE Signal Processing Letters*, Vol 19, No 2, February 2012.

An Adaptive Derivative Free Method for Bayesian Posterior Approximation

Matti Raitoharju*, Simo Ali-Löytty

***Abstract*—In the Gaussian mixture approach a Bayesian posterior probability distribution function is approximated using a weighted sum of Gaussians. This work presents a novel method for generating a Gaussian mixture by splitting the prior taking the direction of maximum nonlinearity into account. The proposed method is computationally feasible and does not require analytical differentiation. Tests show that the method approximates the posterior better with fewer Gaussian components than existing methods.**

I. Introduction

In Bayes' theorem an n-dimensional state vector x is estimated by updating its prior distribution using given measurements. The posterior distribution given measurement y is

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}, \tag{1}$$

where $p(x)$ is the prior probability density function (pdf) of the state, $p(y)$ is a normalizing constant, $p(y|x)$ is the measurement likelihood and $p(x|y)$ is the posterior pdf. In general the update cannot be done analytically.

In this paper the prior is assumed to be a Gaussian and the measurement y to be a scalar that may be written in the form

$$y = h(x) + \varepsilon, \tag{2}$$

where $h(x)$ is the measurement function and ε is the measurement error, assumed to be zero mean Gaussian independent of the prior.

If the measurement function is linear, i.e. $h(x)$ may be written as Jx, the posterior can be computed with the Kalman update [1]

$$\begin{array}{ll} z = h(x) & S = JPJ^T + R \\ C = PJ^T & K = CS^{-1} \\ x^+ = x + K(y - z) & P^+ = P - KSK^T \end{array}, \tag{3}$$

where x and x^+ are the prior and posterior means, P and P^+ are the prior and posterior covariances and R is the variance of the measurement error. In this paper we assume that P and R are nonsingular.

If the measurement model is not linear the above update cannot be used directly. For nonlinear cases one of the simplest update methods is to compute the Jacobian of the measurement function in the prior mean and use it as J in the Kalman update (3). This is used in the Extended Kalman Filter (EKF) [2, p. 278]. This requires analytical differentiation of $h(x)$, which can be difficult or impossible to perform, also the approximation may be poor if the Jacobian J varies a lot in a small area around the prior mean.

The Unscented Kalman Filter (UKF) is an alternative to EKF that does not require analytical differentiation. The UKF update is based on the evaluation of the measurement function at the so called sigma points. The computation of the sigma points require the computation of a matrix square root of the covariance matrix

$$P = LL^T \tag{4}$$

using, for example, Cholesky decomposition. The extended symmetric sigma point set is

$$\begin{aligned} \chi_0 &= x \\ \chi_i &= x + \Delta_i,\ 1 \leq i \leq n \\ \chi_i &= x - \Delta_{i-n},\ n < i \leq 2n, \end{aligned} \tag{5}$$

where $\Delta_i = \sqrt{n+\xi}L_{:,i}$ ($L_{:,i}$ is the i^{th} column of L) and ξ is an algorithm parameter. The prior is updated by using the following approximations in the Kalman update (3)

$$\begin{aligned} z \approx& \sum \Omega_{i,m} h(\chi_i) \\ S \approx& R + \sum \Omega_{i,c} \left(h(\chi_i) - z\right)\left(h(\chi_i) - z\right)^T \\ C \approx& \sum \Omega_{i,c} \left(\chi_i - x\right)\left(h(\chi_i) - z\right)^T, \end{aligned} \tag{6}$$

where $\Omega_{0,m} = \frac{\xi}{n+\xi}$, $\Omega_{0,c} = \frac{\xi}{n+\xi} + (1 - \alpha_{\text{UKF}}^2 + \beta_{\text{UKF}})$, $\Omega_{i,c} = \Omega_{i,m} = \frac{1}{2n+2\xi}, (i > 0)$ and $\xi = \alpha_{\text{UKF}}^2(n + \kappa_{\text{UKF}}) - n$. The variables with subscript UKF are algorithm parameters [3], [4]. Although the UKF update evaluates the measurement in several points, the posterior distribution is approximated with a single Gaussian. In many cases a single Gaussian is not enough to give a good approximation of the posterior.

Gaussian mixture filters use a weighted sum of Gaussian components to approximate the pdfs [5],

$$p(\bullet) = \sum w_k p_{\text{N}}(\bullet|x_k, P_k), \tag{7}$$

where w_k is the component weight and $p_{\text{N}}(\bullet|x_k, P_k)$ is a pdf of normal distribution with mean x_k and covariance P_k. This allows better approximation of the posterior especially when the true posterior is multimodal. The update of any single component may be done using the EKF or UKF formula and the weight of a component is multiplied by the innovation likelihood

$$w_k^+ \propto w_k p_N(y|z_k, S_k), \tag{8}$$

Contact information: Tampere University of Technology. Department of Mathematics, PO Box 553, 33101 Tampere, Finland. e-mal: {matti.raitoharju, simo.ali-loytty}@tut.fi tel. +358456300763. EDICS: SAS-STAT, SAS-SYST. This work was supported by Nokia Inc.

and normalized so that the sum of weights is one.

Although this paper concentrates only on a single update the proposed method is intended to be part of a GMF that does the estimation of a time series. Usually GMFs do not have a constant number of components and a critical issue is to keep the number of components low while still estimating the distributions well.

In algorithms found in the literature the splitting of the prior component in case of nonlinearity into components depends only on the prior distribution. Examples of this kind of GMFs are Sigma Point GMF (SPGMF) [6], Box GMF (BGMF) [7] and Split and Merge Unscented GMF [8]. The first two algorithms require analytical differentiation of the measurement equation and the third uses a simple numerical method for testing if the measurement equation is nonlinear. In this work we propose a new method for prior component splitting that evaluates nonlinearity without the need for analytical differentiation and does the component splitting by taking into account both the prior distribution and the measurement function. In contrast to the methods found in the literature the proposed method does not add components in linear directions.

In the next section a measure of nonlinearity and a formula for its estimation is discussed. Then the splitting of prior according to nonlinearity is presented. In Section IV we show test results of performance of the proposed method compared to existing methods. The paper is concluded in Section V.

II. Measuring Nonlinearity

A second order Taylor series expansion of a scalar function with a single vector parameter may be written as

$$h(x+\Delta) = h(x) + J\Delta + \frac{1}{2}\Delta^T H \Delta + \varepsilon(\Delta), \qquad (9)$$

where H is the Hessian and $\varepsilon(\Delta)$ is the error caused by the higher order components of the measurement equation. If the quadratic term $\frac{1}{2}\Delta^T H\Delta$ and the higher order term $\varepsilon(\Delta)$ are zero then the Kalman update (3) may be used directly. If the quadratic and higher order terms are small the EKF and UKF approximations should work well.

In GMF it is necessary to evaluate whether the nonlinearity is large or small. In [6], it is proposed that a measurement should be considered highly nonlinear if

$$\operatorname{tr} PHPH > R. \qquad (10)$$

This criterion comes from the comparison of the EKF and the Second Order Gaussian filter [2, pp. 345-349], [9, p. 385]. The term $\operatorname{tr} PHPH$ is called nonlinearity in this paper.

Next we propose a numerical method for computing the term PH. In this derivation the higher order term $\varepsilon(\Delta)$ is assumed negligible and the matrix L in (4) is computed using Cholesky decomposition of P. We define matrix Q as

$$Q_{i,j} = \begin{cases} h(x+\Delta_i) + h(x-\Delta_i) - 2h(x) & , i = j \\ \frac{1}{2}[h(x+\Delta_i+\Delta_j) + h(x-\Delta_i-\Delta_j) - \\ \quad 2h(x) - Q_{i,i} - Q_{j,j}] & , i \neq j \end{cases} \qquad (11)$$

where $\Delta_i = \gamma L_{:,i}$. If γ is chosen as $\gamma = \sqrt{n+\xi}$ then the computed values of the measurement equation in (11) may also be used in the UKF component update (6).

Using (9) with (11) we get

$$\begin{aligned} Q_{i,i} =& h(x) + J\Delta_i + \frac{\Delta_i^T H \Delta_i}{2} + h(x) - J\Delta_i \\ &+ \frac{(-\Delta_i)^T H(-\Delta_i)}{2} - 2h(x) = \Delta_i^T H \Delta_i \end{aligned} \qquad (12)$$

and

$$\begin{aligned} Q_{i,j} =& \frac{1}{2}[(\Delta_i+\Delta_j)^T H(\Delta_i+\Delta_j) - \Delta_i^T H\Delta_i - \Delta_j^T H\Delta_j] \\ =& \Delta_j^T H \Delta_i = \Delta_i^T H \Delta_j, H \text{ is symmetric.} \end{aligned} \qquad (13)$$

Thus matrix Q may be written in matrix form

$$Q = \gamma L^T H \gamma L, \qquad (14)$$

which implies that matrix PH may be computed by

$$PH = \frac{1}{\gamma^2}\gamma L L^T H \gamma L L^{-1} = \frac{1}{\gamma^2} L Q L^{-1}. \qquad (15)$$

The computation of the nonlinearity value (10) does not need the inverse of L, because

$$\operatorname{tr} PHPH = \operatorname{tr} \frac{1}{\gamma^2} L Q L^{-1} \frac{1}{\gamma^2} L Q L^{-1} = \frac{\sum_{i,j} Q_{i,j}^2}{\gamma^4}. \qquad (16)$$

III. Splitting the Prior

In this section we propose a novel method for choosing the components of the Gaussian mixture formed from a Gaussian prior by finding the direction of the maximum nonlinearity. If the measurement is nonlinear according to criterion (10) within a Gaussian component, the component is split into a mixture of two Gaussians that preserves the mean and the covariance of the original component. If the nonlinearity is high in resulting components the split is done recursively for the nonlinear components. The recursive splitting helps to take higher order nonlinearities of the measurement equation into account.

The split vector a is chosen from a set of vectors that have the same probability density

$$p(a) = \frac{1}{\sqrt{2\pi \det P}} e^{-\frac{1}{2}a^T P^{-1} a} = \text{ constant} \qquad (17)$$

that maximizes the absolute value of quadratic term in (9). This may be written as

$$\arg\max_a |a^T H a|, \text{ subject to } a^T P^{-1} a = \beta, \qquad (18)$$

where β is a positive algorithm parameter. Using Lagrange multipliers we see that critical points of the optimization problem are vectors that satisfy the constraint and

$$2Ha = 2\lambda P^{-1} a \Leftrightarrow PHa = \lambda a. \qquad (19)$$

Thus, the critical points are eigenvectors of PH that are scaled to satisfy $a^T P^{-1} a = \beta$. Using (19) with (18) we have

$$\arg\max_a |a^T H a| = \arg\max_a |a^T \lambda P^{-1} a| = \arg\max_a |\lambda \beta|, \qquad (20)$$

from which it is seen that an eigenvector a corresponding to the eigenvalue having the largest absolute value is in the direction of maximum nonlinearity.

Equation (14) may be rewritten

$$\frac{1}{\gamma^2}Q = L^T H L. \tag{21}$$

Because matrix $L^T H L$ is real and symmetric its eigenvalues are real and eigenvectors may be chosen orthonormal. Now the eigenvalue decomposition of $L^T H L$ is

$$L^T H L V = V\Lambda, \tag{22}$$

where matrix V is orthonormal having the eigenvectors as its columns; the diagonal elements of the diagonal matrix Λ are the corresponding eigenvalues. Multiplying the above equation from the left by L we have

$$LL^T H L V = LV\Lambda \Leftrightarrow (PH)LV = LV\Lambda, \tag{23}$$

from which we see that matrices PH and $L^T H L = \frac{1}{\gamma^2}Q$ have the same eigenvalues and that an eigenvector of $\frac{1}{\gamma^2}Q$ multiplied from the left by L is an eigenvector of PH. Now the split vector may be written as

$$a = \sqrt{\beta} L V e_i, \tag{24}$$

where e_i is the i^{th} column of the identity matrix and i is the index of to the largest eigenvalue in magnitude. The parameters of a two component mixture that preserves the mean and covariance of the prior may be written

$$\begin{array}{ll} \tilde{x}_+ = x + a & \tilde{x}_- = x - a \\ \tilde{P} = P - aa^T & \tilde{w} = \frac{1}{2}w \end{array}, \tag{25}$$

where w is the weight of the original component and the parameters marked with $\tilde{}$ are parameters of the new split components [8]. To ensure that the covariance matrix stays positive definite we have to ensure that $q^T \tilde{P} q > 0$ for any $q \neq 0$, that is,

$$\begin{aligned} q^T \tilde{P} q &= q^T (P - aa^T) q = q^T L L^T q - \beta q^T L V e_i e_i^T V^T L^T q \\ &= ||L^T q||^2 - \beta \cos^2\theta ||L^T q||^2 ||V e_i||^2 \geq ||L^T q||^2 (1 - \beta), \end{aligned} \tag{26}$$

where θ is the angle between $L^T q$ and $V e_i$. Thus β must be chosen from the range $[0, 1[$.

Because of the trace properties it holds that $\operatorname{tr} PHPH = \sum \lambda^2$. The reduction of nonlinearity may be evaluated by looking at the change of the eigenvalues in the resulting component, assuming that the Hessian H does not change. Because

$$\begin{aligned} \tilde{P} H L V &= \left(P - \sqrt{\beta} L V e_i (\sqrt{\beta} L V e_i)^T\right) H L V \\ &= \left(L L^T - \beta L V e_i e_i^T V^T L^T\right) H L V \\ &= \left(L - \beta L V e_i e_i^T V^T\right) L^T H L V \\ &= \left(L - \beta L V e_i e_i^T V^T\right) V \Lambda \\ &= L V \left(\Lambda - \lambda_i \beta e_i e_i^T\right). \end{aligned} \tag{27}$$

it follows that the new matrix $\tilde{P}H$ has the same eigenvectors as the original matrix PH and only the i^{th} eigenvalue is changed, from λ_i to $(1 - \beta)\lambda_i$. Thus the nonlinearity is reduced by $(2\beta - \beta^2)\lambda_i^2$.

In Figure 1 the effect of parameter β on the resulting components is presented. The original Gaussian represented by the dashed contour line is horizontally split into two new components. Using β close to 1, the nonlinearity decreases fast in splits, but the resulting approximation may be bad. On other hand using a small value of β reduces the nonlinearity more slowly. In our tests in Section IV we used $\beta = 0.5$ as a compromise that gave good results in our test scenario.

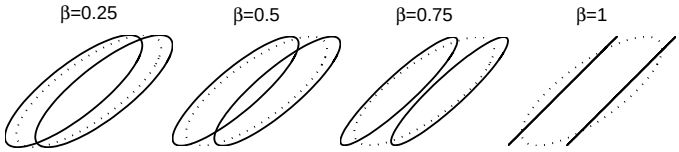

Fig. 1. Effect of β on resulting components of splitting

If the measurement is not scalar the splitting could be done for each measurement component separately. Further if the measurements are independent then the update may be done separately for each measurement component.

IV. Results

Evaluation of the performance is done by comparing the posterior distributions computed by several methods. The proposed method is called adaptive splitting (AS) where the prior is split until none of the mixture components is considered highly nonlinear according to criterion (10). The method is also tested in a variant where at most one split is allowed (AS2). Other methods in comparison are single component UKF, SPGMF [6] with parameters $\tau_{\text{SPGMF}} = 0.5$ and $\kappa_{\text{SPGMF}} = 4$ and BGMF [7], with $N = 1$ and $c_\Sigma = 1$. SPGMF and BGMF use analytic computation of nonlinearity to decide whether the prior shall be split. All tested methods use UKF update (6) with $\alpha_{\text{UKF}} = 10^{-3}$, $\kappa_{\text{UKF}} = 0$ and $\beta_{\text{UKF}} = 2$ [4]. The reference solution is computed using a dense grid where the probability density function is evaluated in each point using Bayes' update formula (1).

Our simulation scenario was a two dimensional positioning case. The measurement function used in simulations was a range measurement from the origin,

$$h(x) = ||x|| + \varepsilon, \tag{28}$$

where ε is a zero mean Gaussian error term with variance R.

In the simulations the range measurement had mean chosen randomly from a uniform distribution in $[0, 10]$ and a unit variance. The prior mean was uniformly distributed with both dimensions in range $[0, 10]$ and covariance matrix had all 10 on diagonal and non diagonal elements were uniformly randomly chosen from the range $[-10, 10]$. The split distance parameter β used in simulations was set to 0.5 i.e. the eigenvalue of PH in the split direction was halved in each split.

An example of a test is presented in Figure 2, where a single 2D range measurement is applied to a Gaussian prior. When comparing visually the true posterior and the different posterior approximations it is seen that the single Gaussian of UKF approximation is not enough in this case and that the proposed method (AS) produces estimates at least as good as the other methods.

In Table I are mean results from 10000 simulation runs. "Time" is the relative time of the method compared to UKF.

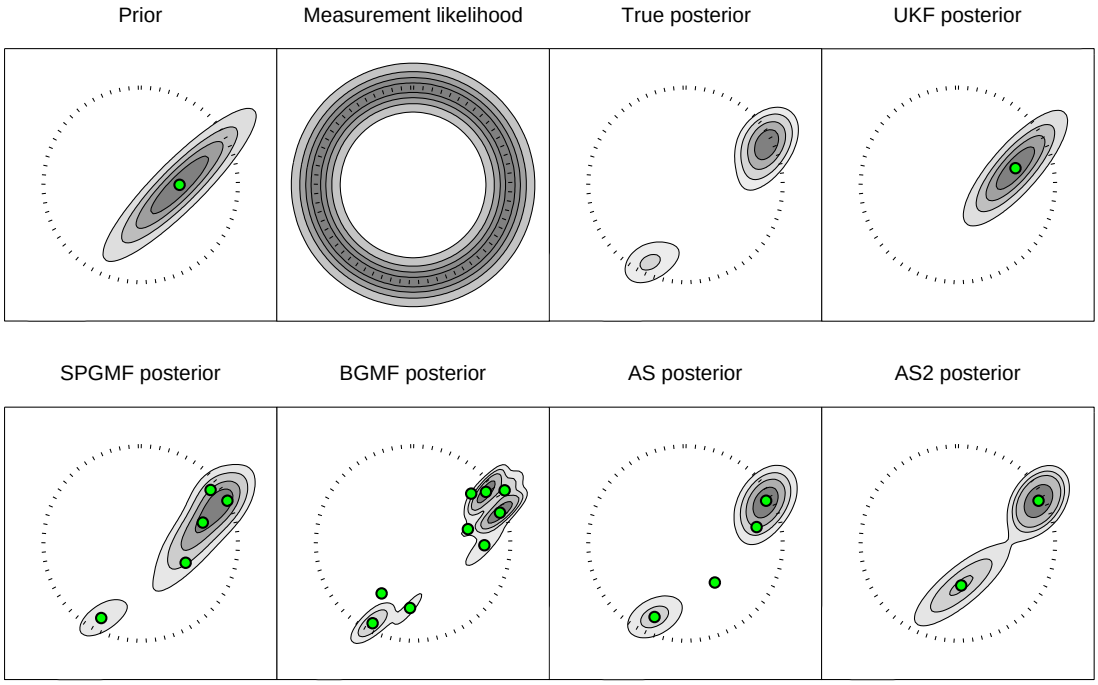

Fig. 2. Exemplary prior and posterior approximations in case of one range measurement. Pdfs are presented with contour maps and the component means are shown as dots.

TABLE I
COMPARISON OF DIFFERENT POSTERIOR APPROXIMATION METHODS

Method	Time	K-L divergence	Components
UKF	1.0	0.74	1
SPGMF	2.9	0.50	3.8
BGMF	12.5	0.47	6.6
AS	4.9	0.39	2.6
AS2	2.4	0.47	1.7

"K-L divergence" (Kullback-Leibler divergence [10]) is defined as

$$D_{\mathrm{KL}}(p||q) = \int p(x) \log \frac{p(x)}{q(x)}, \tag{29}$$

where $p(x)$ is the reference pdf and $q(x)$ is the pdf of the Gaussian mixture approximation. "Components" is the number of components in the posterior approximations.

Results show that the proposed method produces a posterior that is clearly closer to the true posterior than UKF, SPGMF or BGMF, and that performs at least as well as the other methods even when the maximum number of components is limited to two. This is a clear indication that the measurement nonlinearity should be taken into account in splitting. The test for nonlinearity (10) gave same result in all 10000 cases for the analytical and numerical methods.

If the state would include more dimensions, for example, the 2D velocity, the number of components of SPGMF and BGMF would have increased from 5 to 9 and from 9 to 81 respectively, whereas AS would not have any more components. Although the SPGMF and BGMF could be programmed in a such way that they do not do splitting in linear dimensions, this would require manual work to customize the algorithms.

V. CONCLUSION

In this paper it was shown that the nonlinearity of a measurement may be estimated numerically and that if the prior is split in the direction of the maximum nonlinearity the posterior approximation may be done accurately with a relatively small number of components. The proposed method produces better results with a smaller number of components than existing methods and may be used when the measurement equation is hard or even impossible to differentiate.

REFERENCES

[1] Y. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *Automatic Control, IEEE Transactions on*, vol. 9, no. 4, pp. 333 – 339, oct 1964.

[2] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*, ser. Mathematics in Science and Engineering. Academic Press, 1970, vol. 64.

[3] S. J. Julier, J. K. Uhlmann, and H. F. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *American Control Conference*, vol. 3, 1995, pp. 1628–1632.

[4] E. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, 2000, pp. 153 –158.

[5] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, 1971. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/0005109871900975

[6] S. Ali-Löytty and N. Sirola, "Gaussian mixture filter and hybrid positioning," in *Proceedings of ION GNSS 2007, Fort Worth, Texas*, Fort Worth, September 2007, pp. 562–570. [Online]. Available: http://math.tut.fi/posgroup/ali-loytty_sirola_ion2007a.pdf

[7] S. Ali-Löytty, "Box Gaussian mixture filter," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2165–2169, September 2010.

[8] F. Faubel, J. McDonough, and D. Klakow, "The split and merge unscented Gaussian mixture filter," *Signal Processing Letters, IEEE*, vol. 16, no. 9, pp. 786 –789, sept. 2009.

[9] Y. Bar-Shalom, T. Kirubarajan, and X.-R. Li, *Estimation with Applications to Tracking and Navigation*. New York, NY, USA: John Wiley & Sons, Inc., 2002.

[10] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

PUBLICATION 3

Matti Raitoharju, Toni Fadjukoff, Simo Ali-Löytty, and Robert Piché: Using Unlocated Fingerprints in Generation of WLAN Maps for Indoor Positioning. In *Proceedings of the IEEE/ION Position Location and Navigation Symposium (PLANS)*, pages 576–583, Myrtle Beach, USA, April 2012.

Using Unlocated Fingerprints in Generation of WLAN Maps for Indoor Positioning

Matti Raitoharju, Toni Fadjukoff, Simo Ali-Löytty, Robert Piché
Tampere University of Technology
Tampere, Finland
matti.raitoharju@tut.fi

***Abstract*—This paper presents five methods for generation of WLAN maps for indoor positioning using crowdsourced fingerprints. A fingerprint is assumed to contain identifiers of WLAN access points, received signal strength values and, if the fingerprint is collected outdoors, a GPS position. The proposed methods use the fingerprints' information to generate a WLAN map that contains estimated access point locations. Two of the proposed methods use RSS values in access point location estimation. In our evaluation with simulations and with real data, the Access Point Least Squares method, which does not use RSS information, is the fastest and its accuracy is as good as more complex methods that use RSS information.**

I. Introduction

WLAN (Wireless Local Area Network) is the most commonly used method for enabling wireless network connections in mobile devices. The coverage area of a single WLAN AP (access point) is on the order of tens of meters. Although WLAN was not designed for positioning purposes, the abundance of APs and the prevalence of WLAN receivers in mobile devices makes positioning using WLAN an alternative to GPS (Global Positioning System), especially for positioning indoors (airports, malls etc.) where GPS is often unusable.

Most of proposed WLAN positioning systems belong to either of following two categories

1) Fingerprinting methods: In fingerprinting the area of interest is mapped by identifying which WLAN stations can be received and their signal strengths in known positions; each such measurement is called a fingerprint (FP). A mobile device's position can then be computed by comparing its received WLAN signals with those in database. A survey of different fingerprinting methods is done in [1].
2) Network topology modeling methods: In these methods the measured FPs are processed to estimate parameters of the network, for example coverage areas, AP positions and signal attenuation models. An example of a coverage area method is presented in [2].

The large scale collection of fingerprints may be done in massive and expensive data collection campaigns. An alternative is for measurements to be collected by normal users with their own equipment. This method is called crowdsourcing. Crowdsourcing should require as little as possible user interaction and the data sent in by users may have a lot of errors. The WLAN scans may be done automatically and GPS may be used for receiving absolute position outdoors, but indoors there usually is not position information available.

In this paper we study methods to generate WLAN maps that contain estimated positions for WLAN APs for positioning using fingerprints collected with GPS enabled devices. A FP is assumed to contain unique identifiers for WLAN APs that are received, possibly received signal strength (RSS) values and GPS positions when the measurements are done outdoors. Figure 1 shows an example of a situation where FPs contain a list of received APs located inside the square building, and only the FPs outside the building have location information. The goal of our methods is to estimate the AP locations. The methods are developed considering that they should be applicable to a building meaning that they can handle some hundreds of APs simultaneously. This does not restrict the applicability of the presented methods on global scale as GPS is available between buildings and the mapping may be then divided into building scale subproblems.

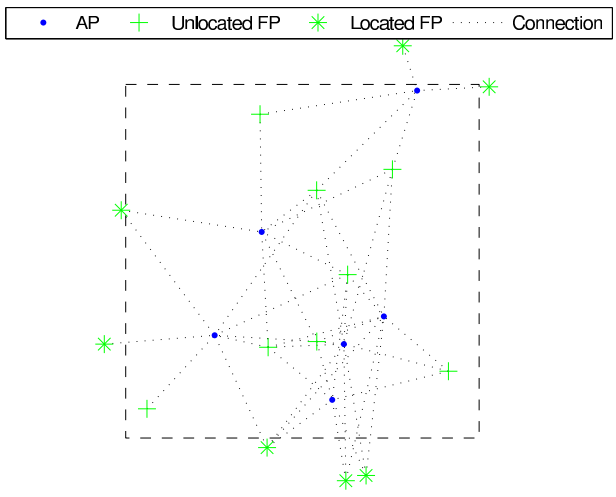

Figure 1. APs, located FPs and unlocated FPs in and around a square building

The problem is similar to localization of sensors in sensor networks. These methods are covered for example in [3]. The main difference to sensor localization methods is that in addition to AP locations we have significant numbers of FPs that do not have location information and are not interesting for us. Most of evaluated methods are designed to be such that the distribution of FPs should not affect results e.g. in buildings where there are more and less used areas.

The rest of the article is organized as follows. In Section II the different sources of measurements (input data of algorithms) are presented. In Section III the different algorithms for doing AP positioning are presented. A simple positioning method that uses AP locations is presented in Section IV. In Section V the performance of algorithms in AP and user positioning is evaluated and Section VI concludes the article.

II. Measurements

In the following sub sections the used measurement sources are presented. The measurements are related to WLAN and GPS systems, which both are available in modern smartphones.

A. Absolute Position

Absolute position is assumed to be received using GPS. GPS receivers provide position estimates with error of a couple of meters in good signal conditions. In this paper we assume that when the GPS position is available it is exact. In real situations the GPS may have hundreds of meters of error, especially close to buildings where only a few GPS signals are available. For real use of the proposed methods we assume that the quality of GPS is monitored and that bad GPS positions have been discarded.

B. Connectivity

Connectivity is determined by checking which APs are received at FPs. Any APs that are received simultaneously have overlapping coverage areas.

C. Received Signal Strength

RSS (received signal strength) measurements are commonly used to do localization. RSS measurements can be converted to range measurements using a simple path loss model [4]

$$\mathrm{RSS} = \mathrm{RSS}_0 + 10\alpha \log_{10} ||r - x|| + \varepsilon, \tag{1}$$

where RSS_0 is the signal strength at range of $1\ \mathrm{m}$ from the AP, α is the attenuation factor, r is the receiver position, x the AP location and ε is the measurement error. In vacuum the value of α is 2, but in realistic situations it should be more. The distance to AP can be explicitly solved from (1) and is

$$||r - x|| = 10^{\frac{\mathrm{RSS} - \mathrm{RSS}_0}{10\alpha}}. \tag{2}$$

III. AP Positioning Algorithms

The goal of AP positioning algorithms is to produce a map of APs using FPs some of which do not have position information. In our case we assume that the measurement come in as a batch and may be treated simultaneously.

A. Mean

In the mean algorithm the position estimate of AP is computed by taking the mean of the FP positions where AP was received. If a FP does not have GPS information the position is the mean of the estimates of the received APs. Because the update is done one FP at a time the outcome of algorithm is dependent on the order of FPs.

The update of a position of a AP location may be expressed as follows

$$x_{i,t} = \frac{(t-1)x_{i,t-1} + r_t}{t}, \tag{3}$$

where $x_{i,t}$ is the position estimate of an AP when t measurements have been processed. The position is

$$r_t = \begin{cases} r_{GPS} & \text{, if GPS available} \\ \sum_{j \in FP} \frac{x_j^{t-1}}{m} & \text{, if GPS not available} \end{cases}, \tag{4}$$

where m is the number of APs that were in FP and already had an estimated position. The position is not computed and t is not updated if there is no GPS measurement or position estimate for any AP in the FP. The Mean algorithm is simple to implement and has fast runtime and as such it may be considered to be a baseline method that other algorithms should outperform in accuracy.

B. Gauss-Newton

The Gauss-Newton (GN) method is named after the Gauss-Newton optimization algorithm [5]. In GN method we use all the data to solve the AP positions and in addition to AP positions the actual FP positions are solved as a side product. The model uses distances between FP and AP locations as given by (2). The optimization goal is to find such locations to FPs and APs that their distances are as close as possible to distances computed by RSS values. This is done by minimizing the sum

$$\sum_{i,j} \left(10^{\frac{\mathrm{RSS}_{i,j} - \mathrm{RSS}_0}{10\alpha}} - ||x_i - r_j|| \right)^2, \tag{5}$$

where $\mathrm{RSS}_{i,j}$ is the received signal strength of ith AP in jth FP. The minimization is done using the Gauss-Newton algorithm.

In Gauss-Newton optimization the sum of squares is minimized iteratively by taking following steps

$$\hat{z}_i = \hat{z}_{i-1} + \beta J_{\hat{z}_{i-1}} \backslash (y - h(\hat{z}_{i-1})), \tag{6}$$

where $J_{\hat{z}_{i-1}}$ is the Jacobian of the measurement function $h(\hat{z}_{i-1})$ and β is a positive scalar. The $\backslash$ operator is used as it is in Matlab i.e. if $J_{\hat{z}_{i-1}}$ is fully determined it solves the

system of linear equations, if it is overdetermined the result is the least squares solution and if it is underdetermined it returns one of the solutions. The vector $\hat{z}$ contains all AP and FP location estimates concatenated and the vector y contains all distances computed with (2) and the measurement function has all the corresponding distances as functions of FP and AP locations

$$||x_i - r_j||\,. \tag{7}$$

The GN requires an initial value $\hat{z}_0$ for AP and FP positions and may converge to different local minima depending on the initial value.

The part of the Jacobian corresponding to FP and AP parts are

$$J_j^{FP} = \frac{(r_j - x_i)^T}{||r_j - x_i||} \tag{8}$$

$$J_i^{AP} = -\frac{(r_j - x_i)^T}{||r_j - x_i||}. \tag{9}$$

When the FP has a GPS position, the r_j is fixed and the Jacobian does not contain elements corresponding to this variable and if $||r_j - x_i|| = 0$ the corresponding Jacobian element is assigned the value 0.

The parameter β is chosen at each iteration to ensure that the sum of squares of $y - h(\hat{z}_{i-1})$ decreases. In our implementation it is set to one at the beginning of each iteration and halved until a value is found such that the objective function decreases. We stop the iteration when all corrections $\|x_{i,t} - x_{i,t-1}\|$ are smaller than a predetermined threshold.

C. Gauss-Newton Max Range

The Gauss-Newton max range (GNMax) method is based on the same iterative optimization method as the GN method, but the model is different. Whereas in Gauss-Newton the positions of unlocated FPs are estimated and the number of equations grows, as the number of FPs grows. In GNMax the number of equations is limited to be the number of AP connections. If two APs are received at once, then the distance between APs is less than or equal to the sum of distances from the FP location to both APs. This may be expressed with the triangle inequality

$$\begin{aligned} ||x_i - x_k|| &= ||(x_i - r_j) + (r_j - x_k)|| \\ &\leq ||x_i - r_j|| + ||r_j - x_k||\,. \end{aligned} \tag{10}$$

While the GN tries to find such APs and FPs that the distances between those match as well as possible to measurement values, the GNMax finds such locations of APs that they are not too far away from each other.

If the FP and the APs are collinear then (10) is an equality. If there are more than one FPs where we receive same APs then the measurement used is the one where the sum of distances computed with RSS values is smallest. In the optimization phase we discard distances where estimates of APs are closer to each other than the measurement indicates. This way the estimation uses only the RSS measurements only to limit the maximum distance between APs. The optimization of measurements is done as in GN, but if two APs is current estimate are closer than the sum of distances, the measurement is neglected. The Jacobian between two APs are

$$J_k^{AP} = \frac{(x_k - x_i)^T}{||x_k - x_i||} \tag{11}$$

$$J_i^{AP} = -\frac{(x_k - x_i)^T}{||x_k - x_i||}, \tag{12}$$

if

$$||x_k - x_i|| > 10^{\frac{\mathrm{RSS}_{i,j} - \mathrm{RSS}_0}{10\alpha}} + 10^{\frac{\mathrm{RSS}_{k,j} - \mathrm{RSS}_0}{10\alpha}} \tag{13}$$

otherwise the distance is not used in update (6).

The GPS measurements are taken into account by using up to three FPs with location information. If there are more than three FPs for an AP with GPS positions we choose three points from the convex hull of the points that produce the largest triangle. This should produce a good geometry for measurements.

D. Access Point Least Squares

Access Point Least Squares (APLS) method may be thought as a spring model, where two APs that are received simultaneously or a AP and a GPS location have a spring pulling them together. The model behind the method is explained more mathematically next.

If a device at r receives signals from two WLAN APs it is located inside of the intersection of the coverage areas. This may be written as

$$r = \hat{x}_{i,j} + \varepsilon_{i,j}, \tag{14}$$

where $\hat{x}_{i,j} = \hat{x}_{j,i}$ is the "center" of the intersection area and $\varepsilon_{i,j} = \varepsilon_{j,i}$ is the position displacement from the center. The center of the intersection may be written using translation vectors $v_{i,j}$ and $v_{j,i}$ that represent the relative positions of the intersections compared to APs positions. The variables used in APLS are presented in Figure 2. By doing the following subtraction, the FP location is eliminated

$$r = \hat{x}_{i,j} + \varepsilon_{i,j} = x_i + v_{i,j} + \varepsilon_{i,j} = x_j + v_{j,i} + \varepsilon_{j,i} \tag{15}$$

$$\Rightarrow 0 = x_i - x_j + v_{i,j} - v_{j,i}. \tag{16}$$

If the translation vectors are independent random variables then the variance of difference is

$$\mathrm{var}(v_{i,j} - v_{j,i}) = \mathrm{var}(v_{i,j}) + \mathrm{var}(v_{j,i}). \tag{17}$$

If we consider only pairwise relationships between the positions of the APs, the AP locations may be estimated by finding a least squares solution for the following equations:

$$\begin{aligned} x_i - x_j &= 0, \text{ for all } i \text{ and } j \text{ connected} \\ x_i &= r_i, \text{ for all } i \text{ with GPS positions,} \end{aligned} \tag{18}$$

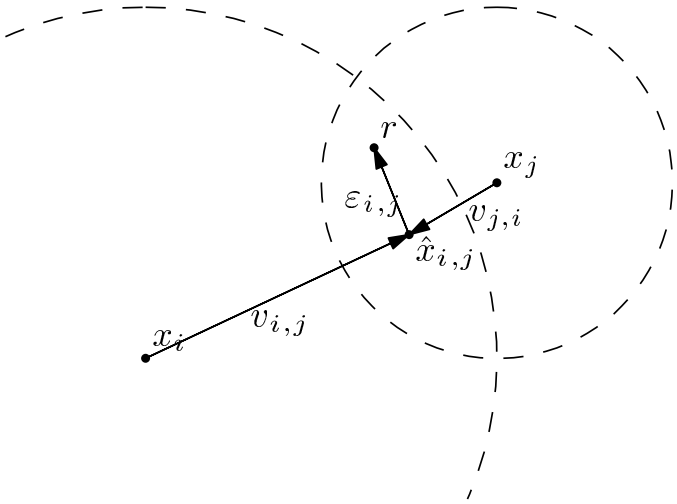

Figure 2. Variables used in APLS

where r_i is the mean of GPS positions of FPs that has ith AP.

These equations may be written in matrix form

$$\begin{bmatrix} \mathrm{D} \\ \mathrm{G} \end{bmatrix} \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ r_1^T \\ \vdots \\ r_k^T \end{bmatrix}, \tag{19}$$

where D contains all the differences and G all the GPS measurements. To find the least squares solution to this set of equations assume that the equations are linearly independent and let $\mathrm{W_D}$ denote the diagonal matrix containing the reciprocals of variance of AP location differences $\frac{1}{\sigma^2_{i,j}}$ on the diagonal and similarly $\mathrm{W_G}$ containing the reciprocals of GPS variances. The weighted least squares solution is now

$$\begin{aligned} \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} = & \left(\begin{bmatrix} \mathrm{D}^T & \mathrm{G}^T \end{bmatrix} \begin{bmatrix} \mathrm{W_D} & 0 \\ 0 & \mathrm{W_G} \end{bmatrix} \begin{bmatrix} \mathrm{D} \\ \mathrm{G} \end{bmatrix} \right)^{-1} \cdot \ldots \\ & \ldots \begin{bmatrix} \mathrm{D}^T & \mathrm{G}^T \end{bmatrix} \begin{bmatrix} \mathrm{W_D} & 0 \\ 0 & \mathrm{W_G} \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ r_1^T \\ \vdots \\ r_k^T \end{bmatrix} \\ = & \left(\mathrm{D}^T \mathrm{W_D} \mathrm{D} + \mathrm{G}^T \mathrm{W_G} \mathrm{G} \right)^{-1} \mathrm{G}^T \begin{bmatrix} r_1^T \\ \vdots \\ r_k^T \end{bmatrix} \end{aligned} \tag{20}$$

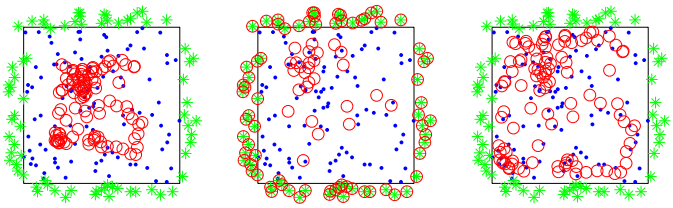

Figure 3. Effect of θ to AP positioning. Left θ too small, center θ too big and right optimal θ

Because of the structure of the matrices, the terms of this equation may be written as

$$\begin{aligned} & \mathrm{D}^T \mathrm{W_D} \mathrm{D} = \mathrm{A} \\ & = \begin{bmatrix} \sum_{j=2}^{n} \tilde{\xi}(1,j) & -\tilde{\xi}(1,2) & \ldots & -\tilde{\xi}(1,n) \\ -\tilde{\xi}(2,1) & \sum_{j=1,j\neq 2}^{n} \tilde{\xi}(2,j) & \ldots & \\ \vdots & \ddots & \ddots & \\ -\tilde{\xi}(n,1) & & & \sum_{j=1}^{n-1} \tilde{\xi}(n,j) \end{bmatrix} \end{aligned} \tag{21}$$

, where

$$\tilde{\xi}(i,j) = \begin{cases} 0, \text{ if } i \text{ and } j \text{ have never been received at once} \\ \frac{1}{\sigma^2_{i,j}}, \text{ if } i \text{ and } j \text{ have been received at once} \end{cases} \tag{22}$$

and

$$\mathrm{G}^T \mathrm{W_G} \mathrm{G} = \mathrm{B} = \mathrm{diag}(\phi_1, \ldots, \phi_n), \text{where} \tag{23}$$

$$\phi_i = \begin{cases} \frac{1}{\sigma^2_{i,\mathrm{gps}}}, \text{ if any FP has GPS position} \\ 0, \text{ otherwise} \end{cases} \tag{24}$$

$$\mathrm{G}^T \begin{bmatrix} r_1^T \\ \vdots \\ r_k^T \end{bmatrix} = \tilde{\mathbf{y}} = \begin{bmatrix} \phi_1 r_1^T \\ \vdots \\ \phi_n r_k^T \end{bmatrix} \tag{25}$$

The solution is now simply

$$\begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} = \left(\mathrm{A} + \mathrm{B} \right)^{-1} \tilde{y}. \tag{26}$$

In APLS all the variances between APs are set to be equal. We found that the AP positioning gives best results when the GPS weight $1/\sigma^2_{i,\mathrm{gps}}$ is set to be $\theta \mathrm{A}_{i,i}$, where θ is a scalar parameter. In Figure 3 the different weighting schemes are evaluated. Dots represent the AP positions, stars are GPS positions and circles are estimated AP positions. In leftmost the weight between GPS measurement and APs is low. In center the GPS weight is infinite and on right an optimal weight is used. The optimal value of θ is discussed in Section V.

E. Variance Access Point Least Squares

Variance APLS (VARAPLS) takes the RSS values between APs into account when computing the variances, as follows

$$\sigma_{i,j}^2 = \left(\min ||r - x_j|| + ||r - x_i||\right)^2 . \quad (27)$$

IV. User Positioning

As the ultimate goal of the research is to position user and not the AP we need to evaluate AP positioning algorithms in user positioning. For user positioning we compute the mean of the five strongest APs received

$$r = \frac{\sum_{i \in \text{five strongest}} x_i}{5}. \quad (28)$$

This method is used as it uses only AP locations and in results section we show that if the AP positions are correctly located it provides rather good positioning performance.

V. Results

First we look at finding the optimal θ for APLS and VARAPLS. We varied the parameter value and did positioning tests similar to one in Figure 3. In Figure 4 the effect of θ on AP localization is presented. Optimal values were found to be 1.63 for APLS and 1.58 for VARAPLS.

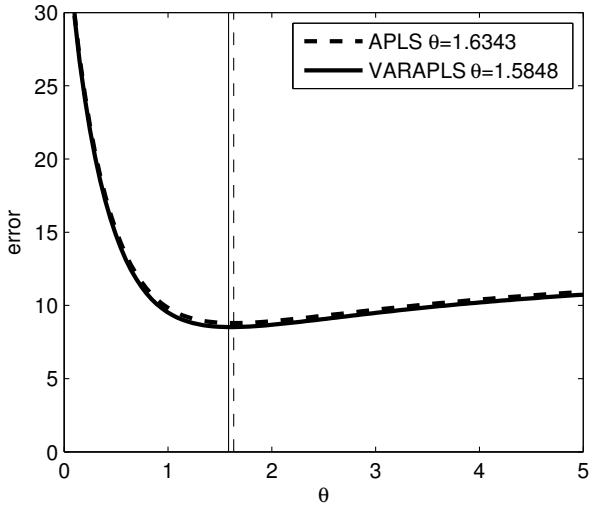

Figure 4. Effect of θ on AP positioning accuracy

In Figure 5 is shown the effect of different parameters to runtime and accuracy of the methods in a simulated environment. In simulations AP:s were randomized inside a box of area $100\,\mathrm{m} \times 100\,\mathrm{m}$. The FPs were randomized on a slightly larger box that exceeded the AP box by 10% in all directions. The FPs that were located outside the AP box were considered to have exact position information. We investigated the influence of five parameters on the performance of the presented methods. The tests were done using combinations of different parameters. The GN methods require an initial estimate of AP positions. As initial estimate we used the mean position of all GPS FPs for all of APs and FPs. The methods marked with prior in Figure 5 use APLS estimate as initial estimate for APs. As the stopping condition for GN methods we used that all APs move less than $10^{-4}\,\mathrm{m}$ in (6).

In first row of Figure 5 parameter σ is the standard deviation of RSS measurements. The errors were rounded samples from zero mean normal distribution. In GN methods the RSS value is transformed into distance. The error in distance does not have zero mean. This bias is left uncompensated because to simulate the modeling errors in path loss parameters. The computation time of Gauss-Newton method increases as measurement error increases, because the solution does not have a clear optimum when error increases. In the error dimension we see that GNMax gives worse results than APLS when standard deviation is more than $5\,\mathrm{dBm}$ and for GN with prior this limit is $9\,\mathrm{dBm}$. This implies that if the signals are noisy or the RSS model is not accurate the APLS should be used. On other hand, if accurate distance measurements are available these methods have small errors. In [6] the standard deviation of WLAN measurements is studied and results show that the value varies between 1 and 5 when the device does not move, but the received mean may change $4\,\mathrm{dBm}$ depending on the orientation of user and in [7] the standard deviation of WLAN measurements was found to be usually more than $10\,\mathrm{dBm}$.

In the second row of Figure 5 the effect of the number of APs is evaluated. In the third row the number of FPs is evaluated. There one should note that the runtime of GNMax gets smaller as the number of FPs increase. In fourth row the range of an AP is evaluated. There we see that the APLS based methods work better the smaller the range is, that is, they work best if the building is big. On the other hand the GN benefits from long ranges of measurements.

In the last row of Figure 5 the effect of nonuniform distribution of FPs is analyzed. The abscissa is probability that the FP, instead of being randomly located on whole area, is located inside $20\,\mathrm{m}$ wide corridor in the center of the building. All proposed methods suffer from the nonuniform distribution of FPs.

In general we see that GN benefits a lot from using APLS as the prior, whereas GNMax is not so sensitive to the initial condition. The accuracy of the Mean method is worse than the accuracy of any other method. Also the time of execution of GN methods is much greater than APLS and Mean method and our APLS implementation is even faster than the implementation of Mean algorithm.

In Figure 6 routes solved from AP locations estimated by different methods are presented. The building was walked once around outside and then some measurements were done inside without GPS available. The grey dots in the figure represent FPs with GPS positions. All the GPS measurements were made outdoors and the points that are inside of building show

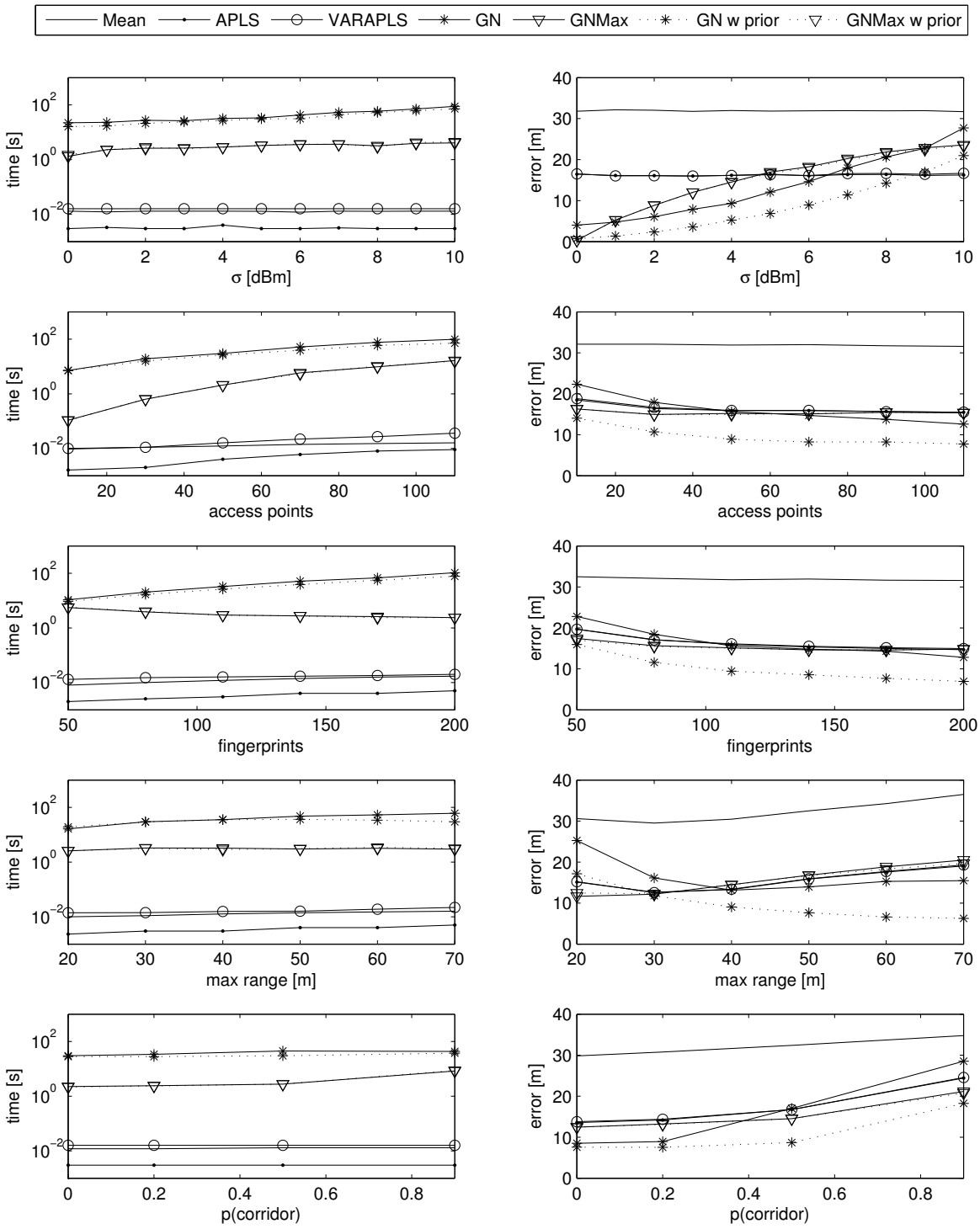

Figure 5. Performance of different methods in simulated environment

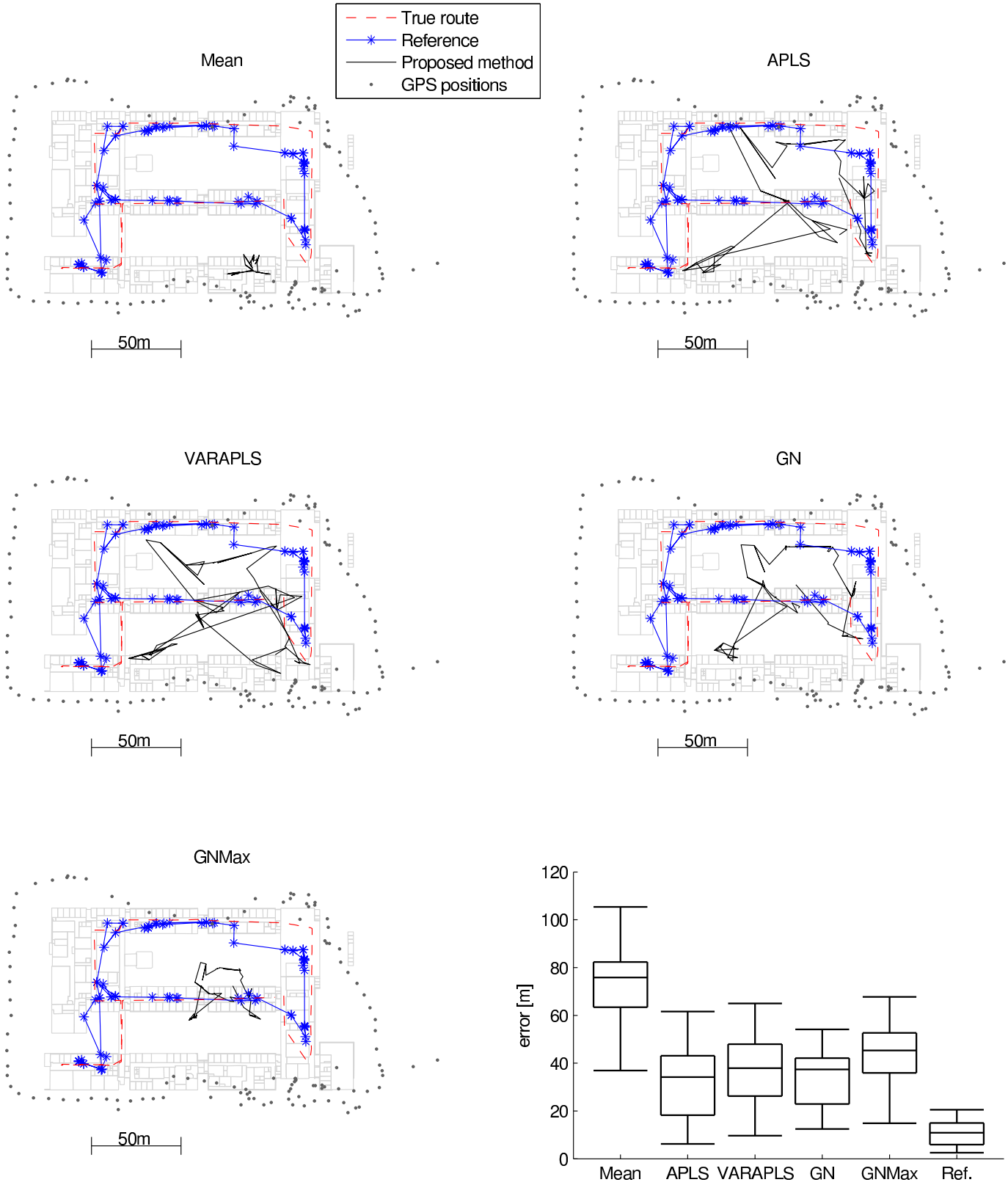

Figure 6. Real data positioning done using radiomaps generated with different methods and error distributions

the effect of errors in GPS measurements. The data had 440 APs and 400 FPs in total of which 120 had GPS position and 280 were taken indoors without GPS. In GN methods we used path loss parameters $\mathrm{RSS}_0 = -40\,\mathrm{dBm}$ and $\alpha = 3.5$. The boxplots in the lower right corner show 5%, 25%, 75%, 95% quantiles and the mean of error. For the reference track, the AP locations were defined by making measurements in different points inside the building and choosing the AP location to be the mean of the five locations where the AP was received with strongest RSS. APs for the reference map were located using measurements in the same floor where the test track was walked. The times that different methods used were less than $0.05\,\mathrm{s}$ for Mean and both APLS variants, for GNMax it took 260 seconds to find an estimate and for GN the time was around 10000 seconds. For GN and GNMax the iterations were stopped when all APs moved less than 2 meters on one iteration. From these results we see that the Mean method does not work at all. The GNMax gives estimates close to the center of the building. APLS, VARAPLS and GN give quite similar performance, where the right side of building was estimated better and the left side worse. APLS had the best mean error of 33 meters, while the reference mean was 11 meters.

VI. Conclusions

In this paper we presented five methods for localization APs in cases where position information is not always available. Our results show that proposed methods APLS and VARAPLS perform quite similarly and perform well in realistic simulation cases where noise is present. The positioning test with real measurements showed that APLS may be used to generate a map with 440APs in less than $0.05\,\mathrm{s}$ that may be used in positioning at least on a rough scale or as a prior for more complex methods.

Acknowledgement

This work was supported by Nokia, Inc.

References

[1] V. Honkavirta, T. Perälä, S. Ali-Löytty, and R. Piché, "A comparative survey of WLAN location fingerprinting methods," in *Proceedings of the 6th Workshop on Positioning, Navigation and Communication 2009 (WPNC'09)*, March 2009, pp. 243–251. [Online]. Available: http://math.tut.fi/posgroup/honkavirta_et_al_wpnc09a.pdf

[2] L. Koski, T. Perälä, and R. Piché, "Indoor positioning using WLAN coverage area estimates," in *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Zurich, Switzerland, September 2010. [Online]. Available: http://ieeexplore.ieee.org

[3] G. Mao and B. Fidan, *Localization algorithms and strategies for wireless sensor networks*, ser. Premier Reference Source. Information Science Reference, 2009. [Online]. Available: http://books.google.fi/books?id=tUB1Ooj75OcC

[4] X. Li, "Collaborative localization with received-signal strength in wireless sensor networks," *Vehicular Technology, IEEE Transactions on*, vol. 56, no. 6, pp. 3807 –3817, nov. 2007.

[5] Åke Björck, *Numerical Methods for Least Squares Problems*. Philadephia, PA: Society for Industrial and Applied Mathematics, 1996. [Online]. Available: http://dx.doi.org/10.1137/1.9781611971484

[6] K. Kaemarungsi and P. Krishnamurthy, "Properties of indoor received signal strength for wlan location fingerprinting," in *Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on*, aug. 2004, pp. 14 – 23.

[7] S. Zvanovec, P. Pechac, and M. Klepal, "Wireless lan network design: site survey or propagation modeling?" *Journal of Radio-engineering, Czech Technical University*, vol. 12, no. 4, pp. 44–49, 2003.

PUBLICATION 4

Matti Raitoharju, Marzieh Dashti, Simo Ali-Löytty, and Robert Piché: Positioning with Multilevel Coverage Area Models. In *Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sydney, Australia, November 2012.

2012 International Conference on Indoor Positioning and Indoor Navigation, 13-15th November 2012

Positioning with Multilevel Coverage Area Models

Matti Raitoharju, Simo Ali-Löytty, Robert Piché
Tampere University of Technology
Tampere, Finland
matti.raitoharju@tut.fi

Marzieh Dashti
The University of Sheffield
Sheffield, UK
m.dashti@sheffield.ac.uk

***Abstract*—Fingerprinting techniques provide good indoor and urban user location estimates, but using them in large scale requires an enormous radio map (RM) database. To reduce the database size, we build a statistical model of the coverage area (CA) of each wireless communication node (CN) using "fingerprints" (FP), i.e. reception samples. In previous work we modeled each CA as a single ellipse, so only 5 parameters need to be stored in the RM for each CN. In this paper, we investigate the use of multiple CAs for every CN. FPs are grouped based on received signal strength (RSS) criteria and CA models are fitted to different FP groups. Different choices of RSS boundaries are examined with real data. We present a method for positioning using the proposed "multilevel coverage area radio map". The proposed method is applied on real data sets. The positioning results are compared with those of conventional single level CA positioning and a basic location fingerprint methods. The results show improvement of positioning accuracy compared with positioning with a single level CA. The improvement is due to better use of RSS level information in both the offline phase (constructing the CA radio map) and in the online phase (user positioning). The proposed multilevel CA positioning works with a much smaller RM database than the basic location fingerprint method, without degrading the positioning accuracy in indoor positioning.**

Radio map; Wireless LAN; RSS; Coverage area; Student-t distribution; Normal distribution; Fingerprint; Positioning

I. Introduction

Location fingerprinting is a well-known positioning technique that determines user's location using a database of radio signal measurements. A "fingerprint" (FP) contains the location of the user equipment (UE) and may include a set of radio characteristics records from a variety of radio networks, like received signal strength (RSS) and identifier of the transmitter e.g. identity of a communication node (CN). CN may be a radio station, a TV station, a cellular network base station, a WLAN access point or some other sensor node in a wireless network. In this work the CNs are WLAN access points. A UE may be a laptop, a mobile phone, or any other device connected to a wireless network. Location fingerprinting consists of two phases, an offline data-collecting phase and an online positioning phase. In the data-collecting phase, FPs are measured at various locations using positioning-capable UE [1–4]. The fingerprint database is processed and used to generate a radio map, which provides information about radio signal properties as a function of position. In the positioning phase, the UE samples measurements from CNs and computes user's location using the radio map [5].

Location fingerprinting takes into account the effects that buildings and environment have on radio signals. Hence, in contrast to many other positioning methods, it does not require line of sight conditions to ensure acceptable accuracy. This makes the location fingerprinting method often precise and reliable in complex environments such as indoor and urban environments. A drawback of the fingerprinting method is that, while accuracy may be good when the radio map is up-to-date, it degrades with time because the radio environment changes constantly [4, 6]. Moreover, performing extensive data collecting is needed and storing a huge database (e.g. covering an entire city or country) is costly.

To reduce the database size, we have used FP data to construct a statistical model of the coverage area of a wireless CN [5, 7]. Instead of raw FP data, the radio map consists of the parameters of the coverage areas (CA). Here, a CA means the region in the plane where signals from the CN can be received by the network user. The CAs are modeled as probability distributions whose parameters may be described using the mean and the covariance; this CA model may be visualized as an ellipse. Only five parameters have to be stored in the radio map for each CA. In the online positioning phase the CAs of the heard CNs are used to infer the position of the user.

In [5], the CA is modeled by computing a posterior distribution using Bayes' rule. The Bayesian prior models information about "typical" CAs. This information is especially important when the FP database contains only a few observations from a CN. In [5], CAs are fitted by modeling fingerprints as having a normal (Gaussian) distribution. However, the normal regression model is well known to lack robustness, i.e. outliers produce coverage areas that are too large. The Student-t distribution is an alternative to the Normal distribution that, due to its heavy tails, is better suited as a model of data that may contain outliers. In this paper, CAs are modeled using Student-t and normal distributions as explained in [7, 8].

In our previous works, only one coverage area for every CN was stored in the radio map [5, 7]. In this paper we consider the use of multilevel CAs for every CN in radio map. The FPs are classified into different sets based on their RSS level, and multilevel ellipse-shaped CAs are fitted to each set of FP data. In this paper, different criteria for classifying FPs into different sets are investigated.

A method to use the proposed so-called "multilevel coverage area radio map" for positioning is presented. The method

is tested using real indoor and outdoor positioning data. Positioning results are compared with conventional single-level CA positioning and a basic location fingerprint method. The results indicate the enhancement of positioning accuracy compared to the positioning with one level CA. The proposed multilevel CA positioning requires much smaller radio maps than the basic location fingerprint method, without degrading the positioning accuracy in our indoor tests.

The remainder of this paper is organized as follow: The coverage area estimation model is presented in Section II. Section III describes the coverage area positioning. The test setup for evaluating positioning performance is described in Section IV. The positioning results using multilevel coverage areas are compared in Section V. Concluding remarks are given in Section VI.

II. Multilevel Coverage area models

A. One-level coverage area models

In this section, a method for fitting an ellipse-shaped CA to location FPs is presented. FPs are assumed to follow multivariate Student-t distribution. The method is less sensitive to outliers than existing smallest-enclosing ellipse and Normal-distribution based methods [8].

Here, each observation is modeled as a bivariate ($d = 2$) Student-t random vector z_n with mean μ, shape Σ, and ν degrees of freedom. When the degrees of freedom ν is fixed the mean and shape parameters for the multivariate Student-t distribution may be computed using Expectation-Maximization algorithm [8]:

Algorithm for modeling the CA from FPs

initialize $u_{1:N} \leftarrow$ ones
for $t = 1$ to T do
  $\mu \leftarrow \sum_n u_n z_n / \sum_n u_n$
  $\mathrm{S} \leftarrow \sum_n u_n (z_n - \mu)(z_n - \mu)^T$
  $\Sigma^{-1} \leftarrow (N + \tau - d - 1)\,(\mathrm{S} + \sigma \mathrm{I})^{-1}$
  for $n = 1$ to N do
    $u_n \leftarrow \frac{d+\nu}{\nu + (z_n - \mu)^T \Sigma^{-1} (z_n - \mu)}$
  end
end

In the algorithm τ is a weight parameter describing the 'strength" of the prior and $\sigma = \tau r^2$ where r represents a typical range of a CA.

The relationship between covariance matrix P and the shape matrix Σ is

$$\mathrm{P} = \frac{\nu}{\nu - 2}\Sigma. \tag{1}$$

When degrees of freedom $\nu \to \infty$ the distribution approaches to normal and $\mathrm{P} = \Sigma$.

B. Multi-level coverage area models

To determine multi-level CAs of a specific CN, FPs are classified into groups and a CA ellipse is fitted to each set of FPs. Three different criteria for classifying the FPs are proposed as follows:

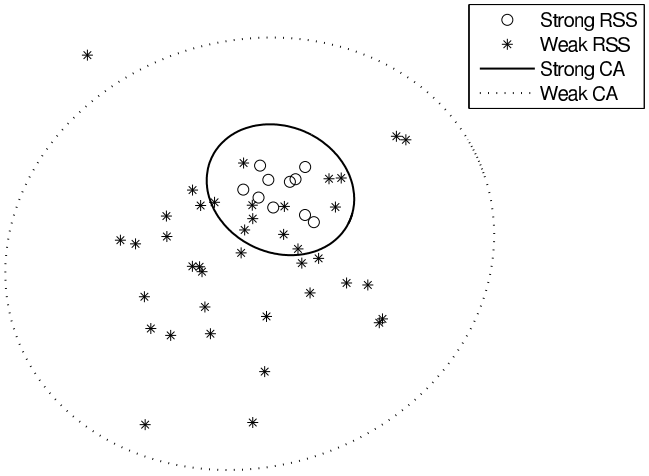

Figure 1: Two-level coverage areas

1) RSS-level: a presumed RSS threshold value is used to determine if a FP has strong signal strength.

2) n-strongest: the n fingerprints with the highest RSS values are classified to the strong set.

3) $x\%$-strongest: the $x\%$ strongest RSS-values of each FP to belong to the strong set.

We also investigate two different ways to deal with the "weak" area. In one case the RSS-values not considered to be strong are used for the weak area and in the other method all of the FPs are used to construct the 'weak" area. Fig. 1 illustrates the basic idea of modeling two-level CAs, where the strong CA is constructed using strong FPs (circles) and weak CA using the weak FPs (asterisks).

III. Positioning

A. Positioning using CAs

Assuming a radio map containing multi-level CAs of CNs is constructed, the goal is to estimate the user's position using the radio map and information that the user receives; the identification codes of heard CNs and their RSS levels.

Let $\mu_1, \ldots, \mu_n$ be the means of the CAs of the CNs that are observed and $\mathrm{P}_1, \ldots, \mathrm{P}_n$ be the corresponding covariances. If the CAs are assumed to be independent measurements, then the best linear unbiased estimator (BLUE) of the user position is

$$x = \left(\mathrm{L}^T \mathrm{W} \mathrm{L}\right)^{-1} \mathrm{L}^T \mathrm{W} \left[\mu_1^T, \ldots, \mu_n^T\right]^T, \tag{2}$$

where

$$\mathrm{L} = [\mathrm{I}, \ldots, \mathrm{I}]^T \tag{3}$$

$$\mathrm{W} = \mathrm{diag}(\mathrm{P}_1, \ldots, \mathrm{P}_n)^{-1}. \tag{4}$$

This may be simplified into form

$$x = \left(\sum_i \mathrm{P}_i^{-1}\right)^{-1} \sum_i \mathrm{P}_i^{-1} \mu_i. \tag{5}$$

This is the same equation as derived in [5], where it was shown to be the Bayesian estimate when the coverage areas are modeled as Gaussians and the prior is uninformative.

As explained in Section II-B, the CAs were constructed using different sets of FPs. The positioning phase uses the same rules to choose which CA is used for each CN.

B. Positioning using fingerprints

As a reference method for positioning we used the weighted k-nearest neighbor method (WKNN) [4]. In WKNN a FP database is searched for k FPs that have the most similar RSS values of CNs and then the user position is computed as a weighted mean of the positions of FPs in the database. In our implementation we used $k = 5$, the similarity of all RSS values is computed using the 2-norm and the weight of a FP is proportional to the inverse of the 2-norm. If a FP in the database did not contain a CN that was in the positioning measurement it is assumed to have a weak RSS value (-105 dBm).

IV. Test setup

In our tests CA models were fitted using different parameter values. We tested all combinations of the following values of parameters: $\tau \in \{5, 10, 20\}$ and $r \in \{5, 10, 20, 40, 80\}$. Degree of freedom for Student-t was set to $\nu \in \{5, \infty\}$, a typical value for general-purpose robust regression and the normal model.

A. Indoor

These tests were done inside a university building. The CAs were fitted using 243 FPs and 331 CNs. Each CN is contained on average in 28 FPs.

Examples of two-level CA ellipses of three specific transmitters are shown in Fig. 2. The ground truth for our test route was manually marked using a laptop during the measurement session. The process of marking positions manually causes some error to the true route but it should be on the order of a meter or two.

Figure 2: Three examples of two-level coverage area models fitted using normal and Student-t regression. In the first example the two strong FPs in left hand side affect strongly on normal model while the Student-t is not affected that much. The second and third examples show the difference of CAs if the strong FPs are included or excluded from the weak CA.

B. Outdoor

In this test the data contained 26921 FPs collected mostly on streets in a suburban area. Fingerprints in the test route had 857 unique CN IDs that were found in the CA database. On average a CN had 57 FPs containing it. The CA models were constructed using the same rules and parameters as with the indoor data. The ground truth was determined using GPS, meaning that there is a couple of meters of error in the true route.

V. Results

Table I shows the positioning results on a route inside a university building for positioning using different rules for coverage areas. The CAs are constructed according to rules given in the table. In the positioning phase the CA corresponding to the first rule, which is true for the FP is used. The parameters shown in the table are those that had the smallest mean error for the rule. r_1 is the prior "range" for the CA generated by Rule 1 and $r_{2,3}$ is for Rules 2 and 3. Mean, Median and 95% columns are the mean, median and 95% quantile of the positioning errors given in meters. The bold numbers are the smallest of each column.

Results show following

- Two CA models give better positioning performance than one
- Student-t ($\nu = 5$ always in Table I) outperform normal models with all rules
- It is better to use all FPs for the "weak" CA instead just "not strongs"
- In most of cases it was best to use small and weak prior for CAs ($\tau = 5, r = 5$)

The Student-t models gave on average 0.7 meters better accuracy than the normal models. We can also see that the use of the third CA for a CN does not improve the performance

Table I: Results for indoor positioning

CAs	Rule 1	Rule 2	Rule 3	τ	r_1	$r_{2,3}$	ν	Mean	Median	95%
1	all	-	-	20	5	5	5	11.6	9.8	23.0
2	1-strongest	all	-	5	5	5	5	10.9	10.2	18.8
2	1-strongest	not 1-strongest	-	5	5	5	5	11.0	10.5	19.1
2	3-strongest	all	-	5	5	5	5	9.7	9.6	**16.3**
2	3-strongest	not 3-strongest	-	5	5	5	5	9.9	9.9	16.5
2	5-strongest	all	-	5	5	5	5	9.5	8.9	17.4
2	5-strongest	not 5-strongest	-	5	5	5	5	9.8	9.3	18.4
2	7-strongest	all	-	5	5	5	5	10.2	9.5	23.0
2	7-strongest	not 7-strongest	-	5	5	5	5	11.4	9.9	28.4
2	10%-strongest	all	-	5	5	5	5	10.1	9.3	17.7
2	10%-strongest	not 10%-strongest	-	5	5	5	5	10.5	10.4	18.1
2	15%-strongest	all	-	5	5	5	5	9.6	9.1	18.4
2	15%-strongest	not 15%-strongest	-	5	5	5	5	10.3	10.6	18.2
2	20%-strongest	all	-	20	5	5	5	9.5	9.6	17.8
2	20%-strongest	not 20%-strongest	-	5	5	5	5	10.5	10.2	18.9
2	30%-strongest	all	-	20	5	5	5	9.9	10.0	18.8
2	30%-strongest	not 30%-strongest	-	20	5	40	5	11.0	10.0	24.3
2	40%-strongest	all	-	5	5	5	5	9.8	9.2	19.2
2	40%-strongest	not 40%-strongest	-	20	5	80	5	10.1	9.5	22.0
2	-75dBm	all	-	10	5	5	5	9.8	9.7	18.0
2	-75dBm	not -75dBm	-	5	5	5	5	10.7	10.5	19.3
2	-85dBm	all	-	5	5	10	5	9.6	9.2	19.4
2	-85dBm	not -85dBm	-	5	5	40	5	10.6	9.9	20.4
3	1-strongest	5-strongest	all	5	5	5	5	9.5	9.0	17.8
3	3-strongest	7-strongest	all	5	5	5	5	9.6	**8.7**	20.9
3	15%-strongest	30%-strongest	all	5	5	5	5	**9.3**	9.1	17.0
Fingerprinting								9.9	8.1	25.1

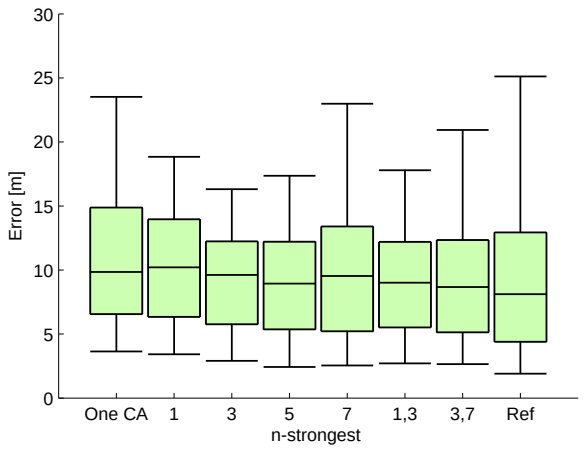

Figure 3: Indoor results with different rules for selecting the strong CAs

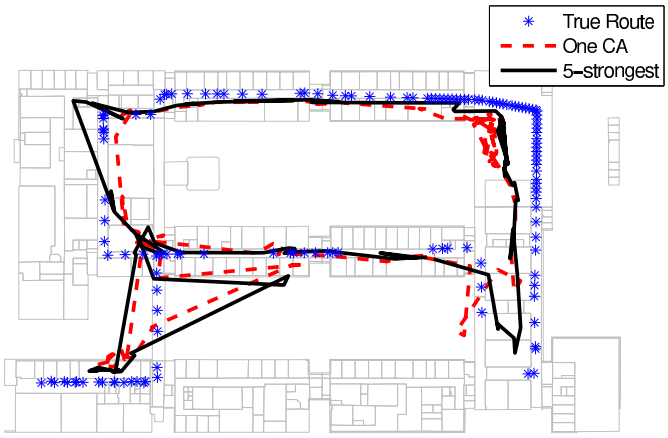

Figure 4: Comparison of routes

much and the 2- and 3-level positioning is comparable to fingerprinting in accuracy.

In Fig. 3 is a boxplot showing 5%, 25%, 50%, 75% and 95% error quantiles for different n-strongest rules in indoor positioning. In cases where two values are given in the x-axis we used three CA models. Used parameter values are $\nu = 5, r_1 = 5, r_2 = 5$ and $\tau = 5$, which seemed to the best for the indoor positioning. From this figure we can see that the median error of all the methods is almost the same, but the 75% and 95% error quantiles are smallest when using two level coverage areas with $n = 3$ or $n = 5$ and three level areas with limits 1 and 3. The reference fingerprinting method has the best 5% and 25% errors, but worse 75% and 95% error quantiles than the best multilevel models.

In Fig. 4 the routes given by our positioning algorithms are illustrated. Stars show the reference locations used in positioning. Dashed line presents the results computed using a single Student-t CA model and the solid line is the route computed using two CAs using 5-strongest rule. The 5-strongest rule is close to accuracy of best methods by all numbers in Table I.

The results for an outdoor scenario (Table II) show smaller improvement of use of the multiple coverage areas compared to the indoor tests. In Fig. 3 is a boxplot showing the error quantiles for different n-strongest rules in outdoor positioning.

Table II: Results for outdoor positioning

CAs	Rule 1	Rule 2	Rule 3	τ	r_1	$r_{2,3}$	ν	Mean	Median	95%
1	all	-	-	20	20	5	∞	51.9	45.7	103.4
2	1-strongest	all	-	20	20	20	∞	51.7	45.9	107.7
2	1-strongest	not 1-strongest	-	20	20	20	∞	52.0	46.4	107.9
2	3-strongest	all	-	20	5	10	∞	50.7	**39.5**	121.4
2	3-strongest	not 3-strongest	-	20	5	10	∞	52.0	42.0	121.9
2	5-strongest	all	-	20	10	20	∞	51.8	44.4	117.0
2	5-strongest	not 5-strongest	-	20	5	10	∞	51.5	43.7	116.9
2	7-strongest	all	-	20	5	20	∞	49.8	42.5	**96.5**
2	7-strongest	not 7-strongest	-	20	5	10	∞	**48.5**	41.7	103.1
2	10%-strongest	all	-	20	20	20	∞	51.6	45.1	111.2
2	10%-strongest	not 10%-strongest	-	20	20	20	∞	52.8	45.1	115.3
2	15%-strongest	all	-	20	10	20	∞	51.8	44.4	114.1
2	15%-strongest	not 15%-strongest	-	20	5	10	∞	51.1	42.1	113.1
2	20%-strongest	all	-	20	5	10	∞	50.7	43.7	113.2
2	20%-strongest	not 20%-strongest	-	20	5	10	∞	50.5	43.6	104.0
2	30%-strongest	all	-	20	10	20	∞	52.5	44.4	111.5
2	30%-strongest	not 30%-strongest	-	10	10	20	∞	52.7	44.8	107.5
2	40%-strongest	all	-	20	10	20	∞	51.5	43.7	102.7
2	40%-strongest	not 40%-strongest	-	20	10	40	∞	52.5	45.1	103.4
2	-75dBm	all	-	20	5	20	∞	51.0	45.7	102.7
2	-75dBm	not -75dBm	-	20	5	20	∞	52.8	46.8	111.8
2	-85dBm	all	-	20	20	20	∞	50.9	45.5	106.7
2	-85dBm	not -85dBm	-	20	20	40	∞	52.6	48.8	112.1
3	1-strongest	5-strongest	all	20	20	20	∞	52.0	45.8	110.4
3	3-strongest	7-strongest	all	20	5	10	∞	50.7	40.5	119.7
3	15%-strongest	30%-strongest	all	20	5	10	∞	52.2	46.3	120.2
Fingerprinting								43.4	39.0	85.4

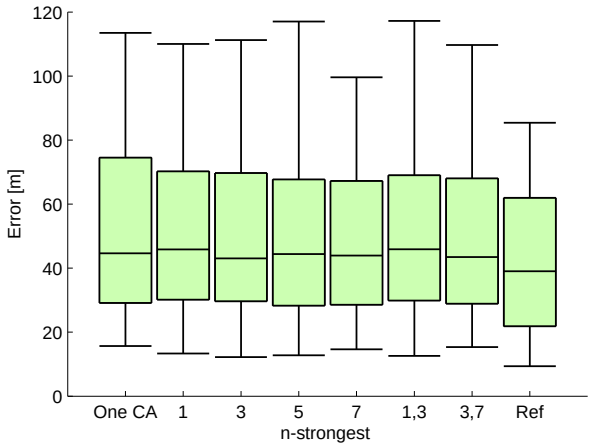

Figure 5: Outdoor results with different rules for selecting the strong CAs

Used parameter values are $\nu = \infty, r_1 = 10, r_2 = 20$ and $\tau = 20$, which seemed to be good values for the outdoor positioning. From this figure we can see that the use of multilevel coverage areas does not improve the positioning as much as in the indoor positioning case.

The normal distribution models outperform the Student-t and optimal values for τ and r are larger than in our indoor test. The mean difference between positioning error between Student-t and normal models is 2.6 meters. The positioning accuracy is somewhat worse than with the reference fingerprinting method. The reason why normal models are better than the Student-t models in our outdoor test is illustrated in

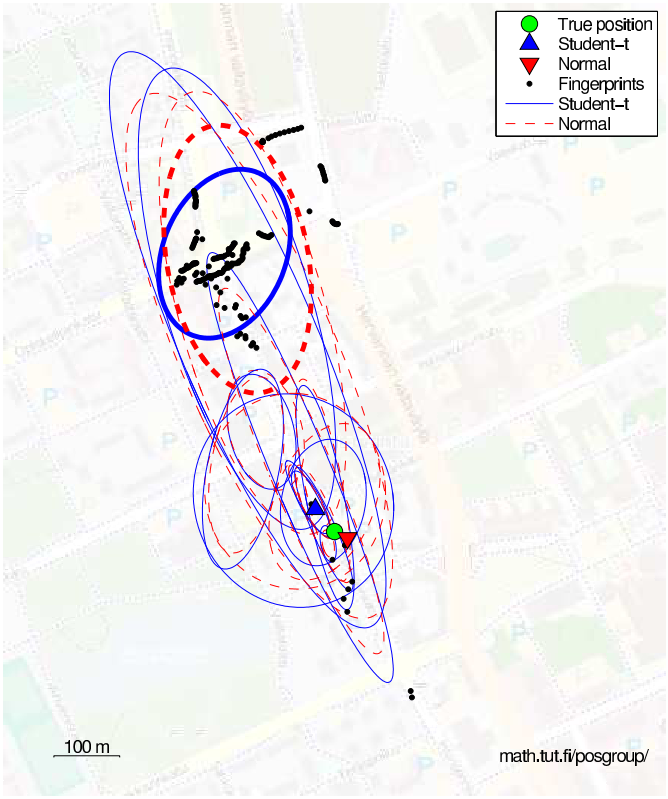

Figure 6: An outdoor positioning example

Fig. 6. The dots show the locations of FPs that were used for generating the CAs with thick lines. There are a few FPs in lower right corner that are considered as outliers in the

Student-t model. In the positioning phase this ellipse affects more to the estimate than the normal ellipse.

VI. Concluding Remarks

This paper examines the use of multiple CA models for a CN instead of one CA for positioning purposes. The proposed positioning method, using multilevel CA models, is compared with conventional CA positioning, using one-level CA models, and reference fingerprinting method.

In our tests with real data we got results showing that the use of multiple CA models for each CN improved the positioning results. The proposed method was tested using real indoor and outdoor positioning data. In indoor tests where the FPs covered the building well the proposed positioning method produced results that were even slightly better than the reference fingerprinting method.

Furthermore, the results show that the CAs constructed using Student-t regression provide better positioning results compared to the CAs constructed with normal regression indoors. In our test the use of three CA models did not give significant improvement compared to 2-level models.

In outdoors test the improvement was smaller and the positioning results were better for normal model. The normal model outperformed the Student-t method because the FP distribution was not uniform and some FPs that were inliers were considered as outliers in Student-t regression. To enhance the positioning accuracy of these cases a robust positioning algorithm should be used.

Our research shows that when building a multilevel radio map the following should be considered:

- Weak CAs should contain all FPs
- If collected data is nonuniform a larger τ and r values should be used compared to uniform data
- RSS-level rule should be avoided because it does not provide better accuracy than rules based on relative strengths of RSS values and the absolute RSS value is dependent of the UE used [9]

For the space requirements of multilevel CA models compared to fingerprinting methods we can consider our outdoor test case. There we had on average 57 FPs for each CN. For fingerprinting at least two numbers have to be stored in the FP for each CA (ID of the CA and the RSS). In our test case we have to store at least 114 numbers on average for a CN. The two level coverage areas require only 10 numbers for a CN, 4 numbers for two means and 6 for two covariance matrices. This simple calculation shows that a positioning database containing two-level CA models require lot less space than a fingerprinting database.

VII. Acknowledgment

This work was supported by Nokia, Inc.

References

[1] H. Laitinen, J. Lähteenmäki, and T. Nordström, "Database correlation method for GSM location," in *IEEE 51st VTC*, Rhodes Greece, May 2001.

[2] S. Ahonen and H. Laitinen, "Database correlation method for UMTS location," in *IEEE 57st VTC*, Rhodes Greece, April 2003.

[3] P. Bahl and V. N. Padmanabhan, "Radar: An in-building RF-based user location and tracking system," in *Proceedings of the 19th International conference on Computer Communications (InfoCom). IEEE, vol. 2*, Dresden Germany, March 2000, pp. 775–784.

[4] V. Honkavirta, T. Perälä, S. Ali-Löytty, and R. Piché, "A comparative survey of WLAN location fingerprinting methods," in *Proceedings of the 6th Workshop on Positioning, Navigation and Communication 2009 (WPNC'09)*, March 2009, pp. 243–251.

[5] L. Koski, R. Piché, V. Kaseva, S. Ali-Löytty, and M. Hännikäinen, "Positioning with coverage area estimates generated from location fingerprints," in *Proceedings of the 7th Workshop on Positioning, Navigation and Communication 2010 (WPNC'10)*, Dresden Germany, March 2010, pp. 99–106.

[6] L. Wirola, I. Halivaara, and J. Syrjärinne, "Requirements for the next generation standardized location technology protocol for location-based services," *Journal of Global Positioning Systems*, vol. 7, no. 2, pp. 91–103, 2008.

[7] M. Dashti, S. Ali-Löytty, L. Wirola, P. Müller, H. Nurminen, and R. Piché, "Robust Kalman filter for positioning with wireless BS coverage areas," in *Proceedings of the 7th Workshop on Positioning, Navigation and Communication 2012 (WPNC'12)*, Dresden Germany, March 2012.

[8] R. Piché, "Robust estimation of a reception region from location fingerprints," in *International Conference on Localization and GNSS*, Tampere, June 2011, pp. 31–35.

[9] K. Kaemarungsi, "Distribution of WLAN received signal strength indication for indoor location determination," in *1st International Symposium on Wireless Pervasive Computing*, Jan. 2006.

PUBLICATION 5

Matti Raitoharju, Henri Nurminen, and Robert Piché: A Linear State Model for PDR+WLAN Positioning. In *Proceedings of the Conference on Design and Architectures for Signal and Image Processing (DASIP)*, Cagliari, Italy, October 2013.

A Linear State Model for PDR+WLAN Positioning

Matti Raitoharju, Henri Nurminen and Robert Piché
Department of Automation Science and Engineering
Tampere University of Technology
P.O. Box 692, FI-33101 Tampere, Finland
E-mail addresses: firstname.lastname@tut.fi

***Abstract*—Indoor positioning based on WLAN signals is often enhanced using pedestrian dead reckoning (PDR) based on an inertial measurement unit. The state evolution model in PDR is usually nonlinear. We present a new linear state evolution model for PDR. In simulated-data and real-data tests of tightly coupled WLAN-PDR positioning, we find that the positioning accuracy with this linear model is almost as good as with traditional models when the initial state is known, and better when the initial state is not known. The proposed method is computationally light and is also suitable for smoothing.**

***Keywords*—Sensor based localization, Hybridization approaches, Signals-of-Opportunity, Pedestrian dead reckoning**

I. Introduction

Wireless Local Area Network (WLAN) access points (APs) are numerous and ubiquitous in most indoor environments. Although WLAN is meant for data transfer, the WLAN signals may be used for user localization. Because the WLAN APs are not meant for positioning, they do not usually send information on their own location for clients. WLAN positioning therefore makes use of a "radio map", which describes certain features of the WLAN signal characteristics at given locations. A radio map is created and updated using data collected on site. These data are called fingerprints (FP). A FP is a report that contains at least the receiver location and the IDs and the received signal strength (RSS) values of APs within reception range. A radio map is constructed off-line based on the collected FPs. The accuracy of positioning based on WLAN signals depends on the model and environment. In small scale (a few buildings) it is possible to achieve positioning results of the order of a couple of meters [1]. For large scale positioning (a city or larger) when the size of the database is a limiting factor the positioning accuracy can be of the order of tens of meters [2].

Pedestrian dead reckoning (PDR) uses an inertial measurement unit (IMU) to detect when a user takes footsteps and how the direction changes between footsteps. The IMU has three axis accelerometers and gyroscopes. The user heading change is computed by projecting the gyroscope measurements to the horizontal plane which is estimated from the accelerometer [3]. The footstep length may also be estimated from the IMU data. If the sensor is mounted on the foot, it is possible to detect when the foot is still and then integrate the footstep length from the IMU data [4]. If the IMU is handheld the footstep length can be inferred from the IMU data also by other methods, see for example [5, 6]. A PDR system can greatly improve the positioning locally as the position estimate may be updated every footstep, but because the errors accumulate over time PDR is often combined with other sensors that can, at least occasionally, provide information of the absolute position.

In this paper we investigate models for fusing PDR measurements with WLAN measurements. We propose a linear state model for the state evolution, whereas in the literature the state model used with PDR system is usually nonlinear [6–8]. For nonlinear estimation in general there is no closed form optimal algorithm. In this paper we use Kalman and Extended Kalman filters, which are computationally light algorithms.

Section II contains the filtering and smoothing algorithms that are used to estimate the user's kinematic state. In Section III we present the WLAN model that is used for positioning. The evaluated PDR models are presented in Section IV. In Section V we evaluate the performance of different models with real and simulated data and Section VI concludes the paper.

II. Filtering Algorithms

The Kalman filter is an algorithm for estimating the state of the system given a time-series of measurements in the case of linear state-evolution and measurement models. If the measurements and state transitions are also normally distributed, the Kalman filter is optimal. The algorithm uses the following update equations at each time index t [9]

$$x_{t|t-1} = F_t x_{t-1|t-1} \tag{1}$$
$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^T + Q_t \tag{2}$$
$$S_t = H_t P_{t|t-1} H_t^T + R_t \tag{3}$$
$$K_t = P_{t|t-1} H_t^T S_t^{-1} \tag{4}$$
$$x_{t|t} = x_{t|t-1} + K_k(y_t - H_t x_{t|t-1}) \tag{5}$$
$$P_{t|t} = (I - K_t H_t) P_{t|t-1}, \tag{6}$$

where x is the state vector, P is the state covariance matrix, F is the state transition matrix, Q is the state transition error covariance matrix, R is the measurement error covariance matrix and

$$y_t = H_t x_t \tag{7}$$

is the linear measurement equation. If the state model is nonlinear then (1) has to be replaced with

$$x_{t|t-1} = f(x_{t-1|t-1}) \tag{8}$$

and F_t in (2) with

$$F_t = \frac{\partial f(x_{t-1|t-1})}{\partial x_{t-1|t-1}}, \tag{9}$$

to get the approximative nonlinear estimation algorithm known as the Extended Kalman filter (EKF) [10].

The Rauch-Tung-Striebel smoother [11] may be used to enhance the state estimates when measurements of future time instants can also be used, for example when plotting the track over a given time interval . The recursive smoothing equations are

$$C_t = P_{t|t} F_t^T P_{t+1|t}^{-1} \tag{10}$$
$$x_{t|m} = x_{t|t} + C_t(x_{t+1|m} - x_{t+1|t}) \tag{11}$$
$$P_{t|m} = P_{t|t} + C_t(P_{t+1|m} - P_{t+1|t})C_t^T, \tag{12}$$

where m is the last time index.

III. Coverage Area Positioning

In its simplest form, probabilistic coverage area (CA) positioning is a method for radio map construction in which the reception area of each AP is modeled as a two-dimensional normal distribution. The radio map does not contain any raw RSS data, and thus computational, memory and communication complexity is much lower compared to conventional FP positioning methods. The algorithm and derivations are explained in [12] and here we only briefly present the algorithm.

The coverage area estimate is

$$\mu_n = \frac{\sum z_i}{n} \tag{13}$$
$$\Sigma_n = \frac{\sum z_i z_i^T + B - n\mu_n\mu_n^T}{n+1}, \tag{14}$$

where μ_n and Σ_n are the mean and covariance of the CA estimate based on n FPs, z_i is the location of the i^{th} FP and B is the prior covariance.

When a new measurement is received, these parameters may be updated by

$$\mu_{n+1} = \frac{n\mu_n + z_{n+1}}{n+1} \tag{15}$$
$$\Sigma_{n+1} = \frac{(n+1)(\Sigma_n - \mu_{n+1}\mu_{n+1}^T) + z_{n+1}z_{n+1}^T + n\mu_n\mu_n^T}{n+2} \tag{16}$$

Because the AP models are linear-Gaussian they may be used in KF directly with

$$y = \mu \tag{17}$$
$$H = [I \quad \mathbf{0}] \tag{18}$$
$$R = \Sigma. \tag{19}$$

Here for H it is assumed that the first variables in the state are the position variables.

IV. Pedestrian Dead Reckoning

In most of the literature, the state model used in PDR is nonlinear [6–8]. For comparison purposes, we consider two traditional models. In the first one, the state contains the user location and the direction of movement

$$x_{t|t-1} = \begin{bmatrix} r_{1,t} \\ r_{2,t} \\ \theta_t \end{bmatrix} = \begin{bmatrix} r_{1,t-1} + s_t \cos\theta_{t-1} \\ r_{2,t-1} + s_t \sin\theta_{t-1} \\ \theta_{t-1} + \Delta\theta_t \end{bmatrix}, \tag{20}$$

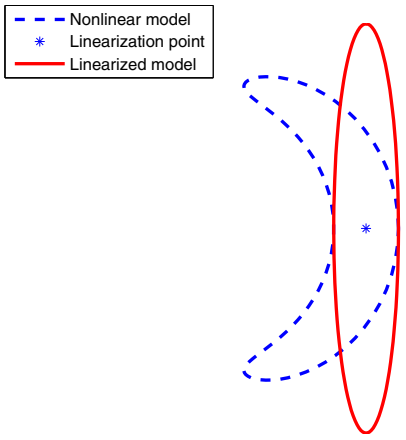

Figure 1. Footstep propagation with nonlinear and linearized state models

where $\Delta\theta_t$ is the change of heading obtained from gyroscopes and s_t is the footstep length estimated from accelerometer data. The linearized state transition matrix is

$$F_t = \begin{bmatrix} 1 & 0 & -s_t \sin\theta_{t-1} \\ 0 & 1 & s_t \cos\theta_{t-1} \\ 0 & 0 & 1 \end{bmatrix}. \tag{21}$$

and the linearized state transition noise covariance is

$$Q_t = \begin{bmatrix} \sigma_s^2 \cos^2\theta_{t-1} & 0 & 0 \\ 0 & \sigma_s^2 \sin^2\theta_{t-1} & 0 \\ 0 & 0 & \sigma_{\Delta\theta}^2 \end{bmatrix}. \tag{22}$$

In our second traditional model the footstep length is also estimated:

$$x_{t|t-1} = \begin{bmatrix} r_{1,t} \\ r_{2,t} \\ \theta_t \\ s_t \end{bmatrix} = \begin{bmatrix} r_{1,t-1} + s_{t-1}\cos\theta_{t-1} \\ r_{2,t-1} + s_{t-1}\sin\theta_{t-1} \\ \theta_{t-1} + \Delta\theta_t \\ s_{t-1} \end{bmatrix}, \tag{23}$$

$$F_t = \begin{bmatrix} 1 & 0 & -s_{t-1}\sin\theta_{t-1} & \cos\theta_{t-1} \\ 0 & 1 & s_{t-1}\cos\theta_{t-1} & \sin\theta_{t-1} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

and

$$Q_t = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\Delta\theta}^2 & 0 \\ 0 & 0 & 0 & \sigma_{\Delta s}^2 \end{bmatrix}. \tag{25}$$

Figure 1 shows position estimates after taking one footstep from known position using this model. The dashed line shows how the state is propagated if the original nonlinear model is used and the solid line shows the linearized state.

We propose the linear state model

$$x_{t|t-1} = \begin{bmatrix} r_{1,t} \\ r_{2,t} \\ v_{1,t} \\ v_{2,t} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & \cos\Delta\theta_t & -\sin\Delta\theta_t \\ 0 & 0 & \sin\Delta\theta_t & \cos\Delta\theta_t \end{bmatrix} \begin{bmatrix} r_{1,t-1} \\ r_{2,t-1} \\ v_{1,t-1} \\ v_{2,t-1} \end{bmatrix} = F_t x_{t-1|t-1} \tag{26}$$

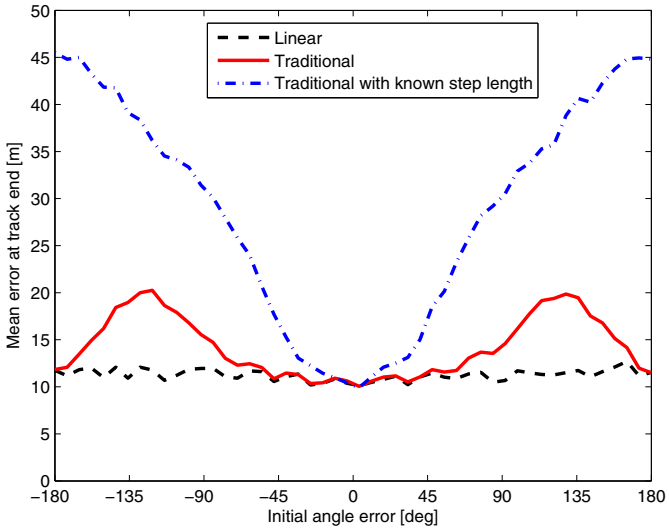

Figure 2. Mean of errors at track end as the function of initial direction estimate

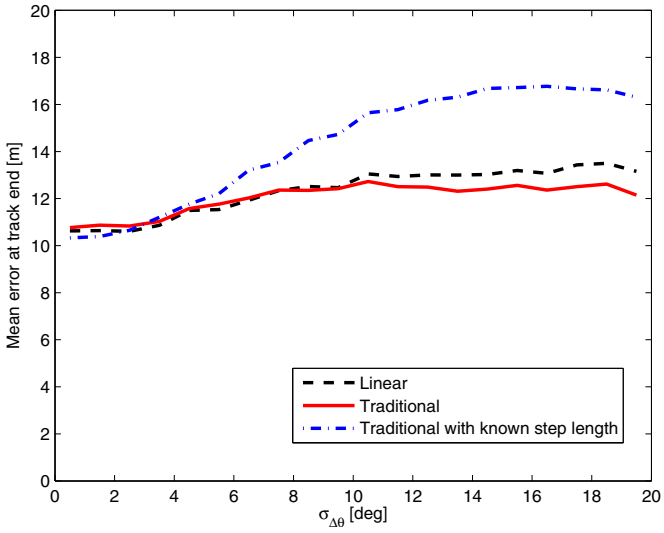

Figure 3. Mean of errors at track end as the function of heading change error

where v is the footstep vector estimate. The process noise covariance matrix is

$$Q_t = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_v^2 & 0 \\ 0 & 0 & 0 & \sigma_v^2 \end{bmatrix}. \quad (27)$$

Compared to the traditional models the proposed model has the benefit that the state transition matrix and the state transition covariance matrix are independent of the state. To keep those independent of the state estimate the state transition error cannot have different variances for heading and footstep length.

V. Performance Evaluation

A. Simulated Tests

In simulations we tested how the linear model performs against the traditional models when the data is generated using the traditional model. For the first simulation, the state model is such that the standard deviations of footstep length is $\sigma_{\Delta s} = 0.01\,\mathrm{m}$ and for the heading change is $\sigma_{\Delta\theta} = \frac{0.01}{0.7}\,\mathrm{rad}$. The initial footstep length is set to $0.7\,\mathrm{m}$. If the footstep length does not change much during the track, the linear model, where $\sigma_v = 0.01\,\mathrm{m}$, has same amount of propagated error in position. The simulated test track is 50 footsteps and at every time step there is a 10% probability of receiving a location measurement with variance $10^2\,\mathrm{m}^2 I$. Initial covariance variance for location dimensions is $10^2\,\mathrm{m}^2$ and for rest of variables $\sigma_{s_0} = 1\mathrm{m}$, $\sigma_\theta = \frac{1}{0.7}\,\mathrm{rad}$ and $\sigma_{v_0} = 1\,\mathrm{m}$.

Figure 2 shows the position errors of the last time step as a function of initial state heading error. Methods tested are the linear and traditional models that estimate both the heading and the footstep length and also the traditional model that gets accurate footstep lengths. From the figure we can see that the linear and traditional models are almost equally accurate when the angle error is small. On the other hand the method with footstep length information works badly when the initial heading error is large. The reason why the traditional method with footstep length estimation is again good when the angle approaches 180 degrees is that the footstep length estimate is equivalent to the negative value of the estimate for 0 degrees.

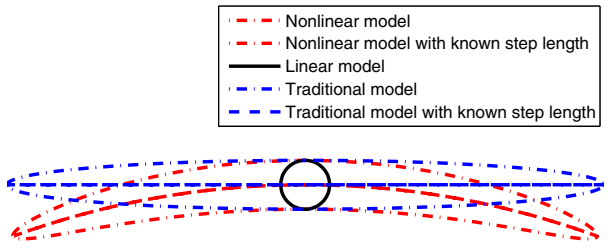

Figure 4. Model errors when $\sigma_{\Delta\theta} = 10$ deg

In Figure 3 we investigate how the different methods perform when the heading error (i.e. gyroscope accuracy) is changed. The traditional models have the correct σ_θ and σ_s, but the linear model is the same as in the first tests because we want to keep the variance independent of the user state. For this test the initial heading is accurate.

From the figure we see that if the heading error is small, the traditional method with accurate footstep length estimates has similar accuracy as the proposed method, but when the angular error grows the model performs badly. The proposed method performs somewhat worse than the traditional method when the variance in angular direction is large, but slightly better when it is small.

Figure 4 shows what kind of error estimates the different methods produce when the $\sigma_{\Delta\theta} = 10$ deg. At this point there is not significant accuracy difference between the proposed and traditional method. This shows how the error can be quite asymmetric before the linear model has worse accuracy than the traditional model. The figure also shows the difference

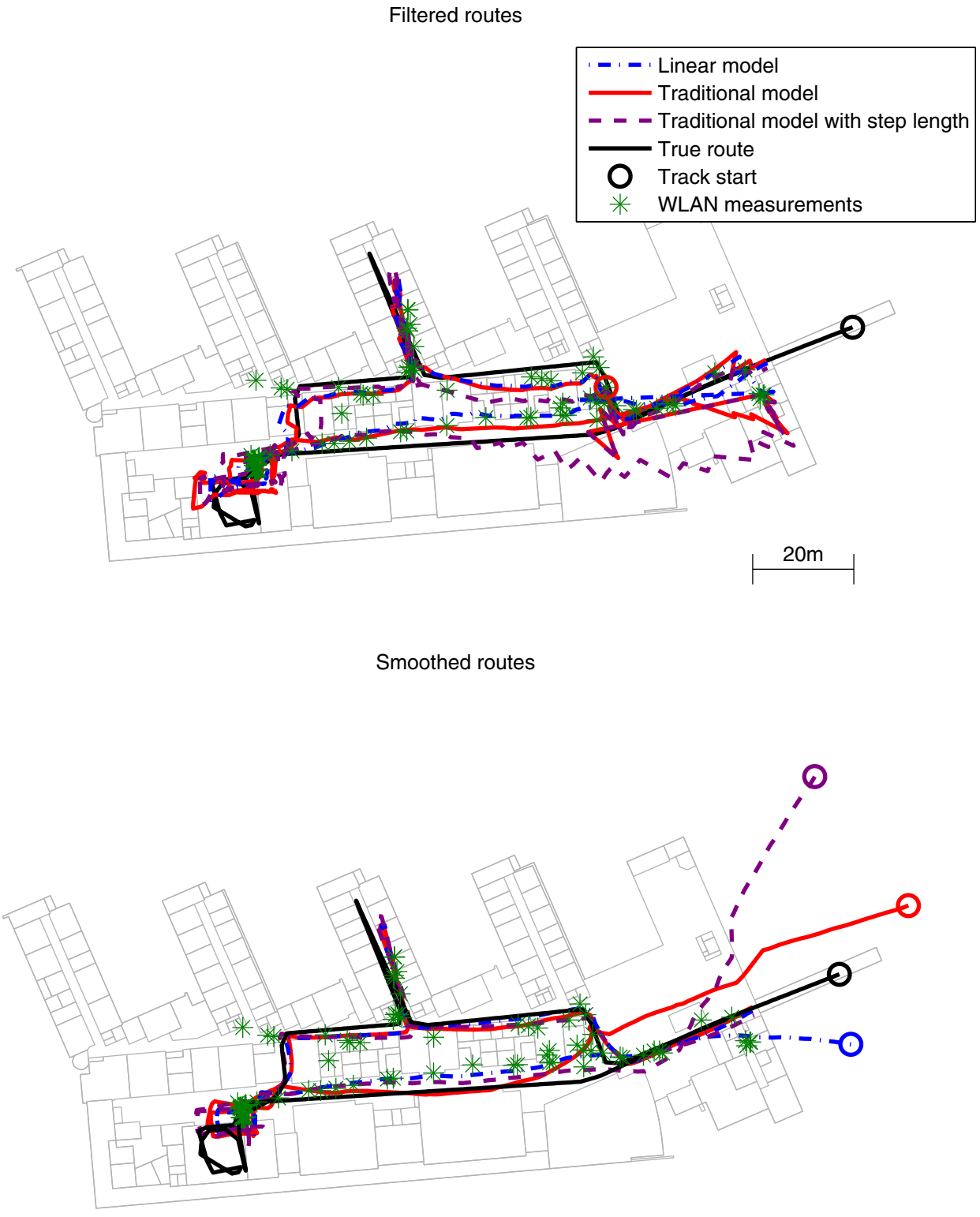

Figure 5. Estimated routes with different models

between the traditional model with known footstep length and how the error would be without linearization. The poor accuracy of the traditional model with known footstep length is caused by the linearization error, which is not taken into account in the model error.

B. True Data

In the true data test the algorithm is tested in one floor of a building in the Tampere University of Technology. The radio map of the test floor contains approximately 200 WLAN APs. The measurements were collected using a XSENS MTi IMU and Acer Iconia tablet. Both devices were carried in hand while doing measurements.

For the WLAN positioning we use the two-level coverage area method proposed in [13]. We generate two coverage areas: a weak CA that is constructed using all FPs, and a strong CA that is constructed with only FPs that have signal strength ≥ -70dBm. The prior for weak CA is $B = 1000^2\,\mathrm{m}^2 I$ and for strong CA $B = 5^2\,\mathrm{m}^2 I$. According to [13] the two-level normal CA models have around 10% larger errors than traditional location fingerprinting, but require only storage of 10 parameters for each AP. Also, the measurements can be used as linear measurements in the Kalman filter.

Figure 5 shows filtered and smoothed tracks. The initial heading for traditional models is 90 degrees off and the initial footstep length estimate of the traditional model is $0.8\,\mathrm{m}$. The traditional method is tested also with step lengths estimated from the sensor data with $\sigma_s = 0.3\,\mathrm{m}$. For the linear model the initial footstep vector is $\mathbf{0}$ with variance $1\,\mathrm{m}^2 I$. The initial position variance is large i.e. the prior is almost uninformative. The smoothed routes have quite big difference in the right hand side of the picture, where the route began. The traditional models are more off than the linear model. This is caused from the wrong linearization of the F matrix in the beginning of the track. The smoothing was done using these F matrices in (10); better results might be obtained with more complex methods such as the Unscented Rauch-Tung-Striebel smoother [14].

Some error statistics are given in Table I. In the table θ_0 is the initial error on the heading. "Static" denotes the WLAN-only position estimates without filtering. When there is only footstep measurements without WLAN measurements, the "Static" uses the last position computed from WLAN measurements as the position estimate. When filtering the proposed method has the best accuracy except when the traditional methods are initialized with the correct initial heading. In smoothing the traditional method with footsteps estimated from sensor data has a slightly better accuracy also when the initial heading is 45 degrees off. The more the initial heading error is the worse the accuracy of traditional methods get. When the initial heading is 180 degrees off neither of traditional methods improve "Static" estimates and the traditional model without footstep lengths from sensor data has even larger errors than "Static".

Footstep length approximations given by different methods are shown in Figure 6. The footstep lengths are computed in the case shown in Figure 5 ($\theta_0 = 90°$). Here we see that the footstep length estimated with the traditional model (s) and the proposed linear model ($||v||$) are rather different in the beginning of the track, but become similar towards the end

Table I. MEAN ERRORS [m] OF DIFFERENT METHODS AND DIFFERENT INITIAL HEADING ERRORS

Method	θ_0	Filtered	Smoothed
Traditional w footstep length	$0°$	6.7	3.8
Traditional w/o footstep length	$0°$	7.4	4.3
Traditional w footstep length	$45°$	7.3	4.3
Traditional w/o footstep length	$45°$	9.2	5.1
Traditional w footstep length	$90°$	7.4	5.2
Traditional w/o footstep length	$90°$	10.7	6.9
Traditional w footstep length	$135°$	7.5	6.1
Traditional w/o footstep length	$135°$	13.4	9.7
Traditional w footstep length	$180°$	8.2	7.3
Traditional w/o footstep length	$180°$	12.1	8.0
Linear		6.8	4.6
Static		8.2	

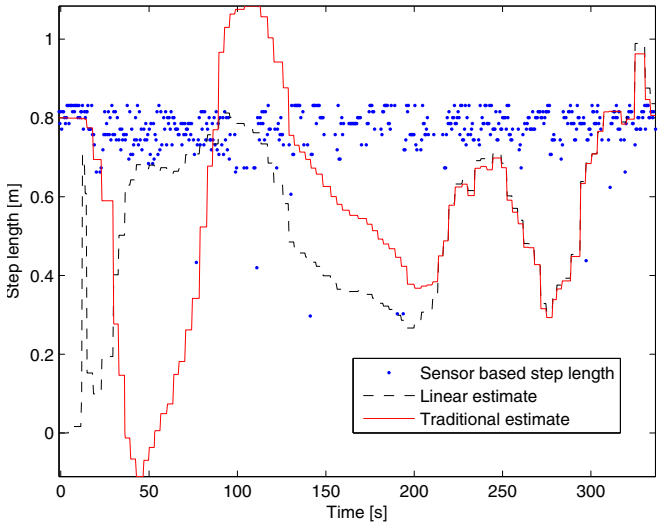

Figure 6. Footstep length estimates

of the track. From 40s to 120s the linear step length is rather constant whereas the estimates computed using the traditional model varies from $-0.1\,\mathrm{m}$ to $1.1\,\mathrm{m}$. In the middle of the route the footstep length estimates are shorter than the footsteps were in reality. This is caused by the WLAN estimates in lower left part of the Figure 5 that are all in same location whereas the true route goes around the lecture hall.

VI. CONCLUSIONS

We proposed in this paper a novel linear model for PDR in indoor personal positioning and compared it to models that are common in the literature. The evaluation shows that although the model is simpler than the traditional methods it performs well and is especially suited for situations where the initial heading and position are not known. As the proposed model is linear it can also be smoothed with the Rauch-Tung-Striebel smoother.

ACKNOWLEDGEMENTS

This work was supported by Tampere Doctoral Programme in Information Science and Engineering, Nokia inc., Nokia Foundation and Jenny and Antti Wihuri Foundation. The authors want to thank Jussi Collin and Jussi Parviainen for codes that estimate the footsteps and headings from the sensor data.

REFERENCES

[1] C. Rizos, A. G. Dempster, B. Li, and J. Salter, "Indoor positioning techniques based on wireless LAN," in *AusWireless '06, Sydney, Australia*, March 2006.

[2] Y.-C. Cheng, Y. Chawathe, A. LaMarca, and J. Krumm, "Accuracy characterization for metropolitan-scale Wi-Fi localization," in *Proceedings of the 3rd International Conference on Mobile systems, Applications, and Services*, ser. MobiSys '05. New York, NY, USA: ACM, 2005, pp. 233–245.

[3] J. Collin, O. Mezentsev, and G. Lachapelle, "Indoor positioning system using accelerometry and high accuracy heading sensors," in *Proceedings of the ION GPS/GNSS 2003 Conference*, 2003, pp. 9–12.

[4] S. Beauregard, "Omnidirectional Pedestrian Navigation for First Responders," in *4th Workshop on Positioning, Navigation and Communication, 2007. (WPNC '07)*, 2007, pp. 33–36.

[5] S. H. Shin, C. Park, J. W. Kim, H. Hong, and J. M. Lee, "Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors," in *Sensors Applications Symposium, 2007. SAS '07. IEEE*, 2007, pp. 1–5.

[6] H. Leppäkoski, J. Collin, and J. Takala, "Pedestrian navigation based on inertial sensors, indoor map, and WLAN signals," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2012*, 2012, pp. 1569–1572.

[7] R. Jirawimut, P. Ptasinski, V. Garaj, F. Cecelja, and W. Balachandran, "A method for dead reckoning parameter correction in pedestrian navigation system," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, no. 1, pp. 209–215, 2003.

[8] F. Evennou and F. Marx, "Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning," *Eurasip Journal on Applied Signal Processing*, vol. 2006, pp. 164–164, 2006.

[9] Y. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Transactions on Automatic Control*, vol. 9, no. 4, pp. 333 – 339, oct 1964.

[10] A. Gelb, *Applied optimal estimation*. MIT Press, 1974.

[11] H. E. Rauch, C. Striebel, and F. Tung, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, 1965.

[12] L. Koski, R. Piché, V. Kaseva, S. Ali-Löytty, and M. Hännikäinen, "Positioning with coverage area estimates generated from location fingerprints," in *7th Workshop on Positioning Navigation and Communication (WPNC)*, 2010, pp. 99–106.

[13] M. Raitoharju, M. Dashti, S. Ali-Löytty, and R. Piché, "Positioning with multilevel coverage area models," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN2012)*, November 2012.

[14] S. Särkkä, "Unscented rauch–tung–striebel smoother," *Automatic Control, IEEE Transactions on*, vol. 53, no. 3, pp. 845–849, 2008.

PUBLICATION 6

Philipp Müller, Matti Raitoharju, and Robert Piché: A Field Test of Parametric WLAN-Fingerprint-Positioning Methods. In *Proceedings of the 17th International Conference on Information Fusion (FUSION)*, Salamanca, Spain, July 2014.

A Field Test of Parametric WLAN-Fingerprint-Positioning Methods

Philipp Müller, Matti Raitoharju, and Robert Piché
Department of Automation Science and Engineering, Tampere University of Technology,
P.O. Box 692, 33101 Tampere, Finland
Emails: {philipp.muller, matti.raitoharju, robert.piche}@tut.fi

***Abstract*—Fingerprint-based (FP) positioning methods determine a receiver's position using a database of radio signal strength measurements that were collected earlier at known locations. For positioning with WLAN signals, nonparametric methods such as the weighted k-nearest neighbour (WKNN) method are widely used. Due to their large data storage and transmission requirements those methods are infeasible for large-scale mobile device services. In this paper we consider parametric FP methods, which use model-based representations of the survey data. We analyse the positioning performance of those methods using real-world WLAN indoor data and compare the results to those of the WKNN method.**

I. Introduction

In the last decade indoor positioning techniques have received extensive attention, and are nowadays an essential feature of many commercial and public service networks. Since many of these applications have to run on small mobile devices, the positioning algorithms have strict limits on allowed energy, memory, bandwidth, and computational resources.

Positioning in indoor environments can rely on measurements from e.g. an inertial measurement unit (IMU) or from radio signals such as Bluetooth, wireless local area networks (WLAN), or ultra-wideband (UWB). In contrast to Bluetooth and UWB, the infrastructure required for WLAN-based positioning, namely WLAN access points (APs) and receivers in user devices (UEs), is already in place. Instead of using signal propagation time, most WLAN-based positioning algorithms exploit the correlation between the received signal strength (RSS) and the UE's location (see e.g. [1, p. 47]). This choice is supported by the fact that media access control (MAC) addresses of APs and the corresponding RSS values are already contained in transmission data [2, pp. 57 ff.], [3, 4], so no changes to the AP software are required.

Because modelling signal propagation, especially in indoor environments, is rather complex, (nonparametric) location fingerprinting methods are widely applied for positioning [5]. Those methods estimate the UE's position by comparing the list of current AP RSS measurements to a database (called a radio map) of information (called fingerprints) on APs and their corresponding RSS values for known positions.

Fingerprints (also known as location reports, reception reports, or observations) are collected in an offline phase by site survey, war-driving or crowd-sourcing. In addition to the UE's current position, each fingerprint (FP) contains radio characteristic records. When FPs are collected from a WLAN they include, in general, at least identifiers (IDs) for the APs from which the UE received a signal and their corresponding RSS values. For localisation in the online phase, a common approach is to use some variant of the weighted k-nearest neighbour (WKNN) method, a nonparametric estimation method. The idea of the WKNN method is to compute a location estimate as a weighted mean of the k FP locations from the radio map whose vectors of AP signal strength values are in some sense closest to the vector of currently measured AP signal strength values [5]. For overviews on nonparametric location fingerprinting methods we refer the reader to [5, 6] and references therein.

Nonparametric fingerprinting methods have the advantage that modelling the signal propagation is not needed. These algorithms have been shown to be reasonably precise and reliable in indoor environments (see e.g. [5, 7]). Their disadvantage is that they work directly with the FP data, and the size of this database can be a critical issue when FP-based positioning is offered as a large-scale service for mobile devices, especially in cellular telephone networks. For example, the Third Generation (3G) system provides transmission rates of 5 Mbit/s at most; for Second Generation (2G) systems those rates are significantly lower [1, pp. 7 ff.]. For WLANs the size of the database is less critical since these systems provide sufficient data transmission rates. Thus, positioning in real time on a mobile UE may be unfeasible since data transmission from the server might be too time consuming or expensive [8]. In addition, large amounts of data have to be stored on the server [9]. One approach for mitigating this problem is to apply data-compression to the radio map [10, 11]. A more fundamental way to address the issue is to use parametric (model-based) FP methods. Those methods use models with small numbers of parameters to describe the FP data, which besides data transmission is also beneficial for the positioning.

In this paper we present an experimental comparison of some of the parametric FP methods that have been proposed in the literature. We start with a brief summary of the methods. In Section II we look at a parametric FP approach that uses elliptical probability distributions for modelling the area in which an AP's signal can be received. Section III is dedicated to a signal propagation path loss model that is calibrated from FP data. In Section IV we consider an approach that uses mixtures of Gaussian distributions to approximate multimodal

distributions. This technique is useful for nonlinear and/or non-Gaussian systems for which traditional approaches such as Kalman filter (KF) and extended KF (EKF) perform poorly. In Section V the performance of these different parametric FP techniques, with and without filtering, is compared in benchmark tests using real-world WLAN measurements in two university buildings. Section VI summarises and concludes

II. Coverage area models

In [6, 8] a computationally light method for parametric fingerprinting is proposed. In order to reduce the size of the FP radio map the authors represent the coverage area (CA; aka reception region) of any AP by an elliptical probability distribution, which can be represented by five parameters [12]. This probability distribution represents only the region in which a signal from the AP can be received; other than an implied reception strength threshold, it gives no information about the RSS. The approach enables fast transmission of the radio map to a UE [6, 8] and fast computation of the UE's position.

A. Coverage area estimation

The coverage area is modelled in [6, 8] by a posterior distribution for the ellipse parameters θ given the FP locations $\mathbf{z} = \{\mathbf{z}_1, \mathbf{z}_2, .., \mathbf{z}_n\}$ where the AP was heard. The distribution is given by Bayes' rule

$$p(\theta|\mathbf{z}) \propto p(\mathbf{z}|\theta)p(\theta), \tag{1}$$

where the likelihood and prior are Gaussian. In other words, the CA is modelled by fitting the mean and covariance of a multivariate Gaussian to the data. Alternatively, to obtain a fit that is robust to outliers, the likelihood and prior could be Student-t distributed [12, 13]. A Bayesian formulation of the regression problem has two advantages. Firstly, the Bayesian prior $p(\theta)$ allows one to exploit information about "typical" coverage areas, which is crucial when only a few FPs are available [6, 8]. Such information is available through experimental studies. For example, the typical reception range for WLAN in indoor environments is 20–50 m [2, p. 9]. Secondly, using Bayes' rule allows recursive estimation and updating of estimates [8].

The CA method considered above ignores the specific RSS values corresponding to IDs of observed APs. Hence it is less sensitive to changes in the radio environment than fingerprinting methods that use these values. This gain in robustness, however, comes generally at the cost of lower accuracy compared with nonparametric fingerprinting methods (e.g. WKNN), which besides FP locations and IDs of APs observed in each FP often[1] also store corresponding RSS values, and use them in the positioning phase.

A coverage area method that uses RSS information is proposed in [6, 13]. Instead of storing only one CA per AP in the database, several CAs per AP are stored, which are modeled from FP data that is grouped according to RSS. In [13] the authors examine the use of one, two and three CAs per AP assuming both Gaussian and Student-t distribution for location reports. FPs are grouped based on their RSS values and different CAs are generated using only location reports of their corresponding group. An important feature is that any FP can be part of more than one group. Three different grouping rules are considered: RSS-level, n strongest APs of each FP and $x\%$ strongest APs of each FP; see [13] for details.

[1]Some authors store RSS-based rankings of AP-IDs or RSS ratios or RSS differences rather than the measured RSS values.

B. Positioning using coverage areas

A position estimate for a UE using coverage areas [6, 8, 13] can be obtained by applying Bayes' rule. The position estimate and an uncertainty measure of the estimate can be extracted from a Gaussian posterior probability density function $p(\mathbf{x}|\mathbf{c})$ of the UE position $\mathbf{x}$ given a list $\mathbf{c} = (c_1, c_2, ..., c_N)^T$ of APs observed by the UE in the current position. For the conjugate (i.e. Gaussian) prior pdf of this position, a suitable mean and covariance, which represent prior knowledge on UE's position, could be chosen. In case such information is unavailable, setting the covariance very large is justified [6]. For computing the likelihood $p(\mathbf{c}|\mathbf{x})$ [6, 8, 13] it is assumed that prior probabilities of observing c_n are equal for all $n = 1, ..., N$ and that observations are conditionally independent given $\mathbf{x}$.

III. Path loss models

Path loss (PL) models refer to models of the signal power loss L_P or the received signal strength P_{RSS} along a radio link, averaged over large-scale and small-scale fading [1, p. 127]. In the simplest models the PL depends only on the transmit power and the distance a radio wave travels; more complex models take further factors into account. For an overview of propagation mechanisms and PL models we refer the reader to [1, 2, 14] and references therein.

The relation between the RSS and the radio wave's traveled distance can be used for positioning. From RSS measurements and PL models the distances between a set of reference nodes and the target node are estimated, which then enables estimation of the target node's position. However, the position estimate is sensitive to signal noise and PL model parameter uncertainties because the distance-power gradient is relatively small [15]. Consequently, these estimates are generally less accurate than radio-signal based estimates that are derived using angle of arrival or time delay measurements.

A. Parameter estimation for PL models

It had been shown that it is ill-suited for several real-world applications to assume the parameters of the PL models to be known a-priori [3]. Therefore, the parameters should be estimated, simultaneously with the AP positions (in case they are unknown), from FP data consisting of AP-IDs and corresponding RSS values.

A widely applied PL model (see e.g. Dil and Havinga [4]) that is used for describing the RSS dependency of distance

d between AP and UE in any indoor environment is the log-distance model

$$P_{\text{RSS}}(d) = A - 10n\log_{10}(d) + w, \tag{2}$$

where $A = P_{\text{RSS}}(1)$ (apparent transmission power) and n (PL exponent) are the unknown parameters, and w is a zero-mean Gaussian random variable with variance σ_w^2 used for modelling the shadow fading (aka slow fading). Nurminen et al. [16] estimate Gaussian distributions for AP position as well as A and n of the AP's PL model simultaneously using the Iterative Reweighed Least Squares (IRLS) method. Similarly to the method introduced in Section II, the Bayesian approach used in [16] permits updating AP position estimate and PL model parameters as new fingerprint data becomes available.

The algorithm uses uninformative Gaussian priors. Nurminen at al. [16] argue that one can choose the valid prior mean values for A and n arbitrarily, since for large numbers of FPs the posterior distribution is typically unimodal, which is supported by Li's findings [3]. Nevertheless, for cases with limited data, a well-chosen informative prior is beneficial. Various studies yielded values for the PL parameters (e.g. [17]). For the prior AP position more care should be taken in order to prevent IRLS placing the AP in an area of weak RSS values [16]. However, even with such measures it cannot be guaranteed that the algorithm finds the correct AP position, but covariance matrices yielded by the IRLS give the user a tool to distinguish between reliable and unreliable AP position and PL parameter estimates. To account for correlation in measurement errors the authors add a small constant diagonal matrix for the AP position's covariance matrix. The cross-correlation between AP position and PL model parameters is, however, neglected, mainly to limit the number of parameters.

B. Positioning using PL models

Once the parameters of the PL model and the positions for all APs are estimated, trilateration or some other nonlinear estimation technique can be used to estimate the position of the UE. In [16] Nurminen et al. test three different methods that use the PL model (2) with real WLAN data in an indoor office environment: a grid method that uses standard Monte Carlo integration, the Metropolis-Hastings algorithm, and the IRLS. All three methods are analysed using both point estimates and Gaussian distributions for A and n. The tests show that assuming Gaussian distributions for the parameters rather than point estimates is, in general, beneficial. These results are not surprising, since the PL model contains approximation errors [18]. If less FP data is available for estimating the PL model those errors, in general, are larger. Therefore, in such situations it should be beneficial, from a theoretical point of view, to assume more uncertainty in the parameter estimates. Furthermore, the PL exponent n can be assumed constant only for a limited time in an environment [3]. Since those changes are minor as long as the environment stays the same, they can be captured to some extent by assuming some uncertainty in the PL exponent estimate.

In terms of computational demand the grid method and the Metropolis-Hastings algorithm have no edge compared with the WKNN that is used as reference, whereas the IRLS is significantly faster and achieves running times close to those of the CA method presented in Section II.

IV. Gaussian mixtures models

A known disadvantage of the CA approach discussed in Section II is that most of the probability mass is located near the centre of the ellipse that is used for describing an AP's coverage area. However, for weak signals the UE is more likely to be close to the edge of the CA. Therefore, CAs yield in such cases rather poor estimates in the positioning phase [6]. In the previous section we looked at approaches that address such issues by taking into account the RSS in addition to the AP-ID by using PL models. Alternatively, we could apply Gaussian mixture (GM) models (aka Gaussian sum models).

A Gaussian mixture is a convex combination of Gaussian density functions $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, namely

$$p(\mathbf{x}) = \sum_{n=1}^{N} \omega_n \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_n, \boldsymbol{\Sigma}_n), \tag{3}$$

where weights ω_n are nonnegative and sum to one. The main theoretical motivation behind GM and filters based on it is that any density function can be approximated, except at discontinuities, by a convex combination of Gaussian densities arbitrarily closely [19].

A. Representing FP data using GMs

In [20] Kaji and Kawaguchi introduce an approach that represent an AP's RSS distribution as a GM model. Although this approach generally will require more data to be stored in the radio map than the CA approach of Section II, it should still require considerably less storage compared with traditional FP databases.

In their algorithm the collected FP data is first transformed into a point distribution, where the point density depends on the signal strength received in a FP (the higher the RSS the higher the point density). Then the parameters of the GM model, namely mean values $\{\boldsymbol{\mu}_n\}_{n=1}^N$, covariance matrices $\{\boldsymbol{\Sigma}_n\}_{n=1}^N$ and component weights $\{\boldsymbol{\omega}_n\}_{n=1}^N$, are optimised by expectation maximisation [21]. Kaji and Kawaguchi point out that their approach allows updating the GM models as new FP data becomes available. They do not provide an equation or rule for determining the number of components N. In our tests in Section V we use $N = \max([K/40], 8)$, where K is the number of FPs in which the specific AP is observed.

B. Positioning using GM approximation of the PL model

In positioning tasks the statistical model is often non-Gaussian and/or significantly nonlinear (see e.g. [22] for a criterion for significant nonlinearity). Therefore the Bayesian recursion is generally unsolvable in closed form [19]. Applying a GM to solve such generally multimodal systems has the advantage that it can follow the multiple peaks of the probability distribution function, unlike the EKF. However,

to ensure fast computing times, which would allow real-time positioning on mobile devices, the number of components should be kept small.

Therefore, Müller et al. [23] introduced a generalised version of GM that relaxes the non-negativity restriction on component weights (GGM). For more details see [24]. Assuming isotropic ranging models, i.e. omni-directional AP antennas, the GGM yields a satisfying approximation of the normalised measurement likelihood with only two Gaussian components, namely

$$p(d|\mathbf{x}) \approx \mathcal{N}(\mathbf{x}_u;\ \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1) \cdot (1 - \bar{c} \cdot \mathcal{N}(\mathbf{x}_u;\ \boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2)), \quad (4)$$

with component 1 having positive weight and component 2 having negative weight, and $\mathbf{x}_u$ being the position vector contained in state $\mathbf{x}$. Furthermore, the formula $\bar{c} = c \cdot (2\pi)^{\frac{n_u}{2}} \sqrt{\det(\boldsymbol{\Sigma}_2)}$, where $c \leq 1$ and n_u is the dimension of $\mathbf{x}_u$, ensures nonnegativity of (4). To achieve similar approximation quality the traditional GM would require a significantly larger number of components due to the infinite number of peaks of the likelihood, which would prohibit its application for real-time positioning.

The main principle of the GGM is to use the AP location as mean values $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$, rather than some of the normalised likelihood's peaks. It then uses the range measurement d, which in case of WLAN RSS measurements is derived from the PL model (2) using the RSS, to determine the components' covariance matrices. Due to assuming isotropic ranging models those covariance matrices are multiples of the identity matrix, i.e. $\boldsymbol{\Sigma}_1 = \sigma_1^2\mathbf{I}$ and $\boldsymbol{\Sigma}_2 = \sigma_2^2\mathbf{I}$ with $\sigma_2 < \sigma_1$. The values σ_1 and σ_2 are determined using a heuristic model, whose parameter(s) are optimised in the offline phase by either minimising the Lissack-Fu distance [23] or the Kullback-Leibler divergence [24] between the exact normalised likelihood and its GGM approximation.

For positioning a GGM for each observed AP is determined, based on the AP's RSS value. Those likelihood approximations are then multiplied with the prior position estimate to get a new Gaussian mixture. Finally, this mixture is collapsed to a single Gaussian to obtain the posterior position estimate and its covariance.

Kaji and Kawaguchi [20] use a particle filter for positioning for their approach, which represents FPs by GM models.

V. Comparative testing

In this section we compare the performance of the parametric fingerprinting and positioning methods described in the previous sections. We evaluated these methods by analysing their WLAN based positioning accuracy for six test tracks located within two buildings of Tampere University of Technology. For two of the tracks measurements were collected several months later than for the other four tracks, which were collected at the same time as the data used for generating the radio maps. Some of the test tracks had floor changes, which were assumed to be known. The radio maps were built separately for each floor. Table I shows for each floor of the two buildings the number of detected APs, the number of FPs, and the number of test points (TP) for the four tracks collected at the same time as the data used for the radio maps. TPs are points on the test tracks that we positioned in our evaluation.

Building	Floor	APs	FPs	TPs
1	1	200	889	19
1	2	289	243	47
1	3	212	160	22
2	1	154	1 530	168
2	2	186	1 582	33
2	3	148	333	19

Table I
Data set sizes. Some APs could be heard on several floors.

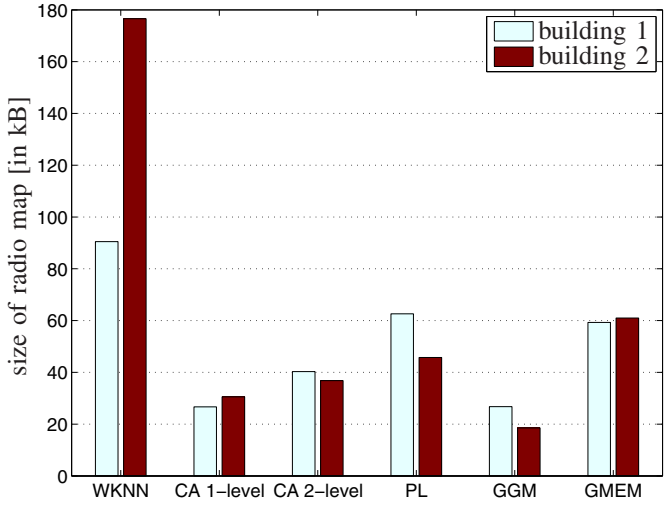

Figure 1. Data storage requirements for radio maps for tested methods in our two test buildings. There were 1 292 FPs in building 1 and 3 445 FPs in building 2

For comparison we implemented CA-based positioning with single CA [8] and 2-level CAs with limit -70 dBm [13], PL model [16], GGM approximation of the PL model [23] and the signal strength estimation model from [20] (denoted GMEM). The standard deviation for RSS based methods was set to 6 dB. In addition to these parametric methods we used a weighted k-nearest neighbours method (WKNN) with $k = 5$ as a reference.

Figure 1 shows the sizes of radio maps for both buildings for each method. The WKNN does not summarise the FPs in any way, and therefore has the highest requirements. All parametric methods reduce the size of the radio map considerably. In our tests the radio map size is reduced between 30% and 90%. However, because the size of the radio map used by WKNN depends on the number of FPs and the size of the radio map used by the other methods depend on the number of APs those numbers cannot be generalised.

Detailed analysis of the radio maps revealed that the PL exponent estimate $\hat{n}$ of an AP tends to take values smaller than 2 if the AP has been observed only in a small number of FPs. If the AP has been observed in a larger number of FPs, then a PL exponent estimate smaller 2 is less likely (68% of all APs that were observed in fewer than 100 FPs have $\hat{n} < 2$, but only 27% of all APs that were heard in more than 100

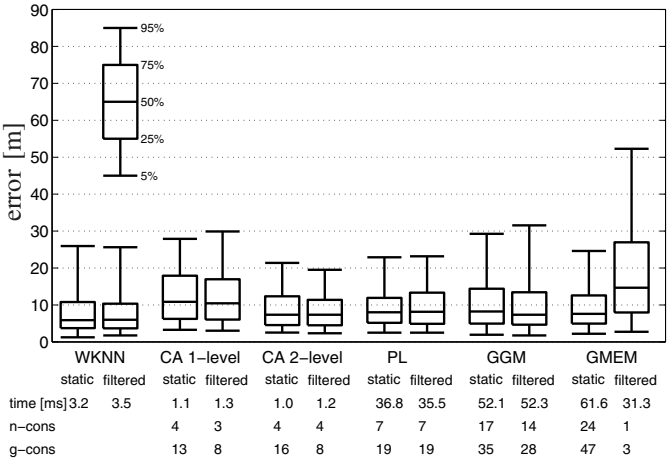

Figure 2. Method performances with all data

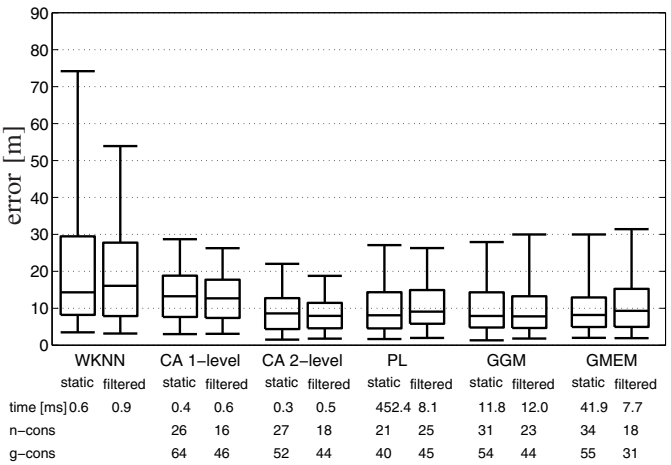

Figure 3. Method performances with five strongest measurements

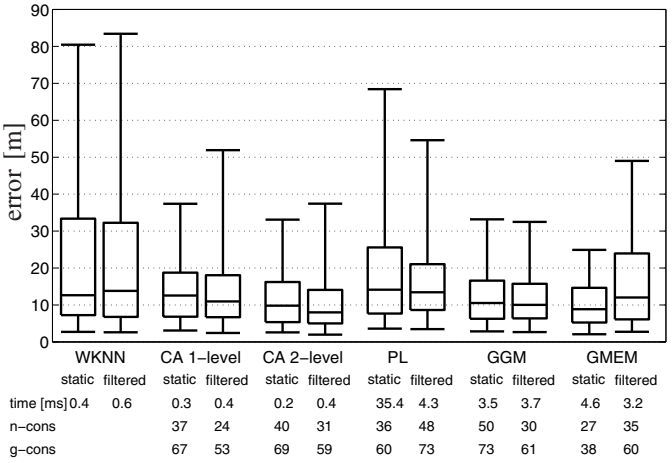

Figure 4. Method performances with 90% of APs dropped

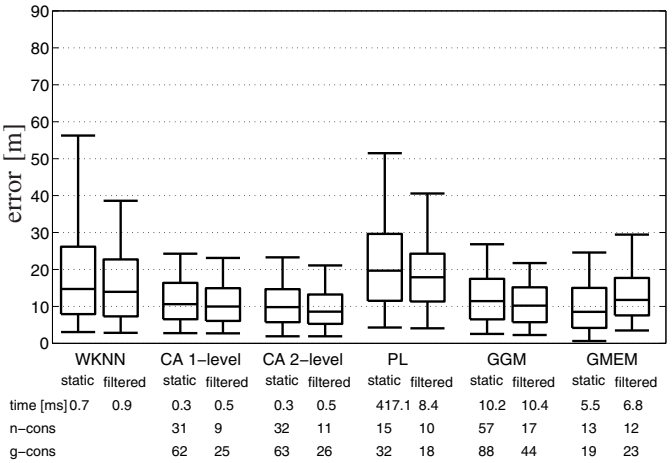

Figure 5. Method performances when positioning done several months after data for radio map generation was collected

FPs have $\hat{n} < 2$). PL exponents smaller 2 can be explained by the fact that the corridors in which we collected our FPs acted as waveguides [1, p. 66].

The true routes for all six test tracks were measured by clicking a map plot on a touch screen while walking and interpolating between the clicks, and were estimated for both static case and filtered case (i.e. time series). For the filtering we considered the state vector $\mathbf{x}_k$ containing location and velocity of the UE. Both CA-models and GGM were updated using a plain Kalman filter. In addition, we collapsed the GGM to a single component after 5 measurements and after each time step. The GMEM used a grid for static position estimation and a particle filter with 300 particles for the time series estimation; the PL model method used Gauss-Newton for static positioning and a GM filter [25] for time series. In time series the effect of parameter uncertainties varied depending on the location, and therefore was computed in the prior mean of the estimate. The WKNN was given a standard deviation of 10 m for filtering with a Kalman filter.

The methods were tested in four different scenarios:

Fig. 2: full data

Fig. 3: only the APs with five strongest signals were used for positioning

Fig. 4: 90% of APs were dropped pseudorandomly to check how the methods perform in situations with low AP density

Fig. 5: data for generating the radio maps and data for positioning were collected with a time gap of several months to evaluate the methods' performance degradation over time

In Figs. 2 – 5 we present quantiles with box plots for the positioning errors, absolute time for one position estimate and consistency values that can be used to evaluate the accuracy of the estimated position's covariance matrix that is reported by a method. For the n-cons (normal consistency [26, p. 235 ff.]) values we assumed Gaussian distributed positioning errors, and computed how often the errors were within the 50% ellipse, i.e.

$$(\hat{\mathbf{x}}_u - \mathbf{x}_u)^T \mathbf{P}_k^{-1} (\hat{\mathbf{x}}_u - \mathbf{x}_u) \leq \chi_2^2(0.5) = 1.3863, \quad (5)$$

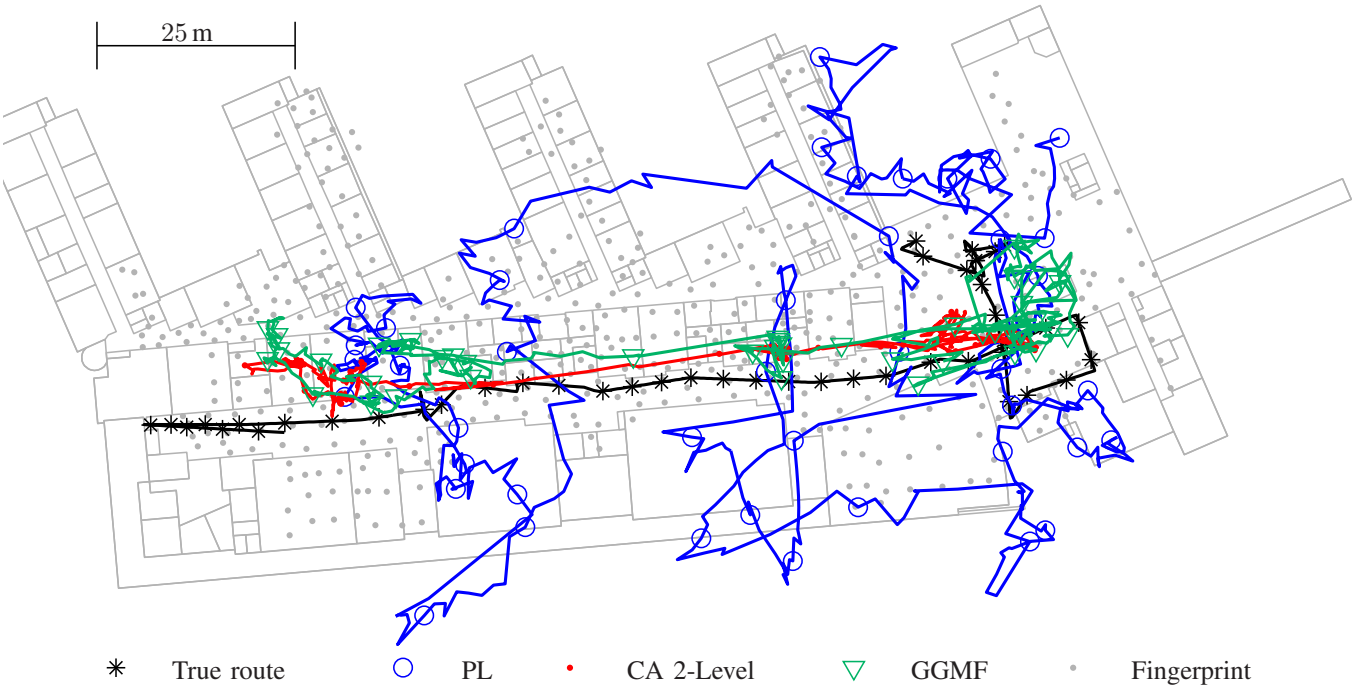

Figure 6. Filtererd routes of selected methods when data for generating the radio map and data for positioning where collected with a time gap of several months.

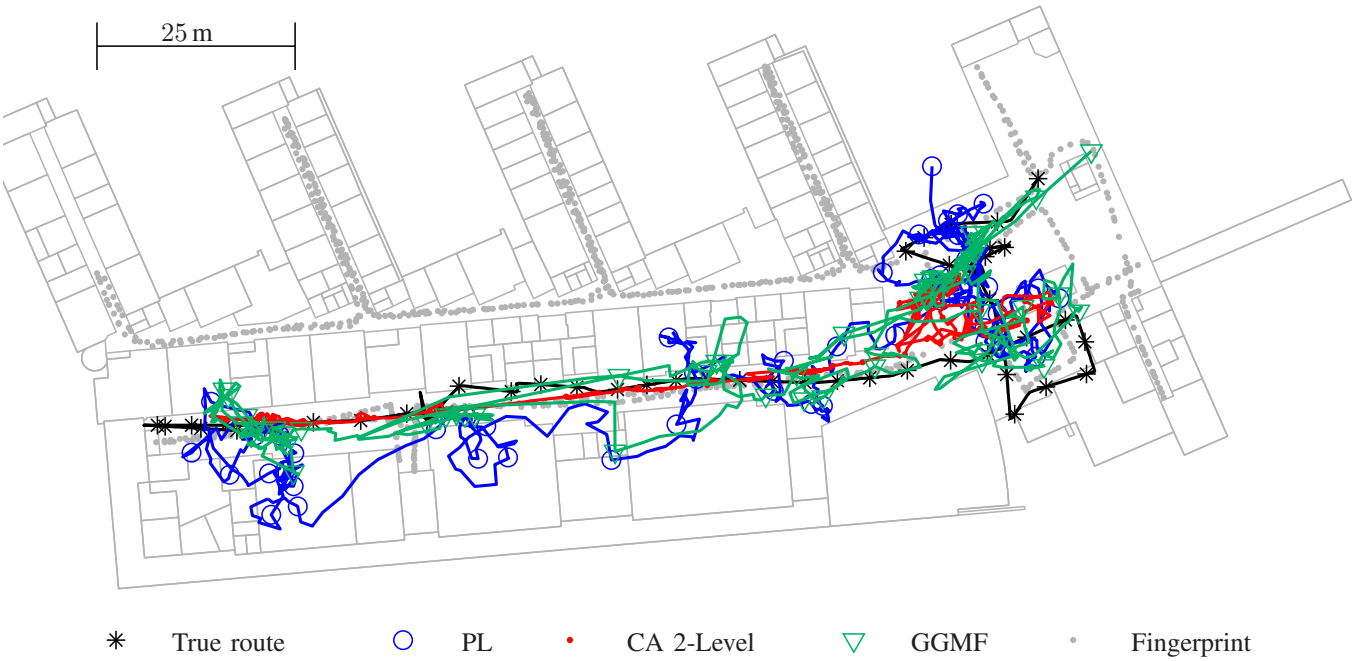

Figure 7. Filtererd routes of selected methods when data for generating the radio map and data for positioning where collected within one month.

where $\hat{\mathbf{x}}_u$ is the estimated UE position, $\mathbf{P}_k$ its covariance matrix and $\mathbf{x}_u$ the true UE position. This measure may be used for checking the error estimate in both ways (if it is too small or large) as long as the distribution is close to the normal distribution. In g-cons (general consistency [27]) we computed how often the errors were within 50% for any distribution using the modified Chebyshev inequality, namely

$$(\hat{\mathbf{x}}_u - \mathbf{x}_u)^T \mathbf{P}_k^{-1} (\hat{\mathbf{x}}_u - \mathbf{x}_u) \leq \frac{2}{0.5} = 4. \quad (6)$$

When using all of the data all parametric methods were inconsistent, with n-cons values far from the desirable 50% and g-cons values far from the 50% that can be interpreted as minimum requirement (a g-cons of 60% is not necessarily better/worse than 55%), and there are no significant differences between the accuracies of the different methods, except for filtered GMEM. The computation time for static GMEM is higher than for filtered GMEM because it is computed on a grid, whereas the filtered GMEM uses a particle filter. The results suggest that the 300 particles proposed in [20] was too few. Using only the five strongest measurements improved the consistency and reduced the relative computing time for all methods. The large time value for static PL can be explained by the facts that in two (of 308) TPs the convergence was extremely slow and that our implementation did not restrict the number of iterations. Since the GGM's computational demand depends in exponential manner on number of measurements [24] the reduction in computation time for static and filtered GGM could be expected, although in our tests the dependence is not exponential due to collapsing a GGM after five measurements to a single Gaussian. At the same time the positing accuracy degraded significantly only for WKNN and the CA 1-level approach. This is evidence for dependency of the measurements. In the test building there were some Multiple Input Multiple Output (MIMO) APs that produced dependent measurements.

Fig. 4 reveals that the more sophisticated approaches (PL, GGM and WKNN) perform worse or similar than the relatively simple CA methods for scenarios with low AP density. The same holds for the scenario in which the radio map was outdated (compare Fig. 5).

One possible reason for the static and the filtered GGM's poor performance in all four scenarios (compared with their performance in [23] and in [24]) might be that we used a different approach for determining the covariance matrices of the GGM's two Gaussian components, since our tests where carried out in a WLAN rather than in cellular telephone or UWB network. We believe that there exist better approaches than the heuristic we used, but more research on this topic will be necessary. A deeper analysis of the GGM can be found in [24].

Fig. 6 shows an example for positioning with outdated radio maps. None of the filtering methods provide satisfactory positioning accuracy in the lower vertical corridor. One reason for the poor performance in that area is that the radio maps are missing FPs from the southern, central part of the building. Another reason may be that the newer data used for positioning contains weaker RSS values than the older data used for radio map generation, which causes the distance from UE to APs to be overestimated. This causes the PL approach, which relies heavily on the RSS, to position the UE outside the building at several occasions.

Using an up-to-date radio map (Fig. 7) the filtering methods provide satisfying positioning accuracy with the PL approach still struggling. This shows how critical it is to have accurate PL model parameter estimates.

VI. Conclusion

In this paper we presented an overview of parametric fingerprinting and positioning methods, and tested them with real WLAN data for different test tracks and scenarios. Besides their positioning accuracies and consistencies, we also compared the storage requirements for their radio maps with that of the WKNN (as an example of a nonparametric FP method).

All parametric methods enable a significant reduction in the size of the radio map used in the positioning phase. In addition, our tests show that all parametric methods, except the CA 1-level and the filtered GMEM method, provide similar positioning accuracy as the nonparametric WKNN in case of a high CN density and when using all available measurements. When using only the five strongest measurements their computation time drops significantly. Furthermore, all parametric methods still show similar positioning performances, while the WKNN's performance degrades considerably. This means that the parametric methods achieve satisfying positioning accuracy even with few observable APs. When the AP density is low or the mapping data is outdated then the simple CA techniques achieve at least similar positioning accuracy than the more sophisticated parametric techniques and the WKNN. Thus, the CA technique gives the best trade-off between accuracy and computational demand. The other parametric methods are, like the WKNN, more vulnerable to harsh environments.

Acknowledgment

Philipp Müller acknowledges the financial support of the TUT Doctoral Programme. Matti Raitoharju is supported by Tampere Doctoral Programme in Information Science and Engineering, Nokia Inc., Nokia Foundation and Jenny and Antti Wihuri Foundation.

We thank Henri Nurminen for his advice and for provision of Matlab code for his approach [16].

References

[1] A. F. Molisch, *Wireless Communications*, 2nd ed. Wiley - IEEE, January 2011.

[2] R. Mautz, "Indoor positioning technologies," Habilitation Thesis, http://e-collection.library.ethz.ch/eserv/eth:5659/eth-5659-01.pdf, February 2012.

[3] X. Li, "RSS-based location estimation with unknown pathloss model," *IEEE Transactions on Wireless Communications*, vol. 5, no. 12, pp. 3626–3633, December 2006.

[4] B. Dil and P. Havinga, "RSS-based localization with different antenna orientations," in *2010 Australasian Telecommunication Networks and Applications Conference (ATNAC)*, 2010, pp. 13–18.

[5] V. Honkavirta, T. Perälä, S. Ali-Löytty, and R. Piché, "A comparative survey of WLAN location fingerprinting methods," in *Proceedings of the 6th Workshop on Positioning, Navigation and Communication 2009 (WPNC'09)*, March 2009, pp. 243–251.

[6] L. Koski, R. Piché, V. Kaseva, S. Ali-Löytty, and M. Hännikäinen, "Positioning with coverage area estimates generated from location fingerprints," in *Proceedings of the 7th Workshop on Positioning, Navigation and Communication 2010 (WPNC'10)*, March 2010, pp. 99–106.

[7] J. Machaj, R. Piché, and P. Brida, "Rank based fingerprinting algorithm for indoor positioning," in *2011 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, September 2011, pp. 6–11.

[8] L. Koski, T. Perälä, and R. Piché, "Indoor positioning using WLAN coverage area estimates," in *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, September 2010, pp. 1–7.

[9] L. Wirola, T. A. Laine, and J. Syrjärinne, "Mass-market requirements for indoor positioning and indoor navigation," in *2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, September 2010, pp. 1–7.

[10] L. Wirola, L. Wirola, and R. Piché, "Bandwidth and storage reduction of radio maps for offline WLAN positioning," in *2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, October 2013.

[11] S. Eisa, J. Peixoto, F. Meneses, and A. Moreira, "Removing useless APs and fingerprints from WiFi indoor positioning radio maps," in *2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2013, pp. 739–745.

[12] R. Piché, "Robust estimation of a reception region from location fingerprints," in *2011 International Conference on Localization and GNSS (ICL-GNSS)*, June 2011, pp. 31–35.

[13] M. Raitoharju, S. Ali-Löytty, R. Piché, and M. Dashti, "Positioning with multilevel coverage area models," in *2012 International Conference on Indoor Positioning and Indoor Navigation*, http://URN.fi/URN:NBN:fi:tty-201310311408, November 2012.

[14] H. Hashemi, "The indoor radio propagation channel," *Proceedings of the IEEE*, vol. 81, no. 7, pp. 943–968, July 1993.

[15] S. Gezici, Z. Tian, G. B. Biannakis, H. Kobayashi, A. F. Molisch, H. V. Poor, and Z. Sahinoglu, "Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 70–84, July 2005.

[16] H. Nurminen, J. Talvitie, S. Ali-Löytty, P. Müller, E. S. Lohan, R. Piché, and M. Renfors, "Statistical path loss parameter estimation and positioning using RSS measurements in indoor wireless networks," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, http://urn.fi/URN:NBN:fi:tty-201310311407, November 2012.

[17] S. Shrestha, E. Laitinen, J. Talvitie, and E. S. Lohan, "RSSI channel effects in cellular and WLAN positioning," in *Proceedings of the 9th Workshop on Positioning, Navigation and Communication 2012 (WPNC'12)*, March 2012, pp. 187–192.

[18] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Transactions on Vehicular Technology*, vol. 29, no. 3, pp. 317–325, August 1980.

[19] H. W. Sorenson and D. L. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, July 1971.

[20] K. Kaji and N. Kawaguchi, "Design and implementation of WiFi indoor localization based on Gaussian mixture model and particle filter," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, November 2012.

[21] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[22] S. Ali-Löytty and N. Sirola, "Gaussian mixture filter in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2007*, May 2007, pp. 831–837.

[23] P. Müller, S. Ali-Löytty, M. Dashti, H. Nurminen, and R. Piché, "Gaussian mixture filter allowing negative weights and its application to positioning using signal strength measurements," in *Proceedings of the 9th Workshop on Positioning, Navigation and Communication 2012 (WPNC'12)*, http://urn.fi/URN:NBN:fi:tty-201311061423, March 2012, pp. 71–76.

[24] P. Müller, H. Wymeersch, and R. Piché, "UWB positioning with generalized Gaussian mixture filters," *IEEE Transactions on Mobile Computing*, 2014 (in press).

[25] M. Raitoharju and S. Ali-Löytty, "An adaptive derivative free method for Bayesian posterior approximation," *IEEE Signal Processing Letters*, vol. 19, no. 2, pp. 87–90, February 2012.

[26] Y. Bar-Shalom, R. X. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation, Theory Algorithms and Software*. John Wiley & Sons, 2001.

[27] S. Ali-Löytty, N. Sirola, and R. Piché, "Consistency of three Kalman filter extensions in hybrid navigation," in *Proceedings of The European Navigation Conference GNSS 2005*, Munich, Germany, Jul. 2005.

PUBLICATION 7

Matti Raitoharju, Simo Ali-Löytty, and Robert Piché: Binomial Gaussian mixture filter. *EURASIP Journal on Advances in Signal Processing* (Under review)

Manuscript will be added here after the review process is completed.

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O.B. 527
FI-33101 Tampere, Finland

ISBN 978-952-15-3404-1
ISSN 1459-2045