

Incrementally learned Mixture Models for GNSS Localization

Tim Pfeifer and Peter Protzel

Dept. of Electrical Engineering and Information Technology
TU Chemnitz, Germany

Email: {firstname.lastname}@etit.tu-chemnitz.de

Abstract—GNSS localization is an important part of today’s autonomous systems, although it suffers from non-Gaussian errors caused by non-line-of-sight effects. Recent methods are able to mitigate these effects by including the corresponding distributions in the sensor fusion algorithm. However, these approaches require prior knowledge about the sensor’s distribution, which is often not available.

We introduce a novel sensor fusion algorithm based on variational Bayesian inference, that is able to approximate the true distribution with a Gaussian mixture model and to learn its parametrization online. The proposed Incremental Variational Mixture algorithm automatically adapts the number of mixture components to the complexity of the measurement’s error distribution. We compare the proposed algorithm against current state-of-the-art approaches using a collection of open access real world datasets and demonstrate its superior localization accuracy.

I. INTRODUCTION

A reliable localization is crucial for autonomous systems like self-driving cars or mobile robots. While global navigation satellite systems (GNSS) can provide this information for open-field scenarios, their accuracy and consistency is decreased in urban areas. The reason are signal reflections on tall buildings, which cause non-line-of-sight (NLOS) measurements. Recent publications like [1] demonstrated the non-Gaussian characteristic of the resulting error distributions. Therefore, a lot of effort has already been invested to make GNSS more robust against NLOS effects [2, 3].

One approach to reduce the influence of these errors, is to incorporate knowledge about their true distribution into the state estimation process. With robust factor graph optimization, there already is a class of algorithms that allows to include such distributions either in a predefined [4, 5] or self-tuning [6, 7, 8] way. To describe the non-Gaussian errors, Gaussian mixture models (GMM) are the established state-of-the-art. However, existing approaches are not able to fully adapt the GMM during the online estimation process. Therefore, they do not address applications where the error characteristic changes significantly over time. For GNSS this can easily happen, since the NLOS error depends on the receiver’s surrounding.

Based on our recently published Adaptive Mixture algorithm [7], we want to introduce a novel approach that combines variational Bayesian inference [9] with non-linear

The project is funded by the "Bundesministerium für Wirtschaft und Energie" (German Federal Ministry for Economic Affairs and Energy).

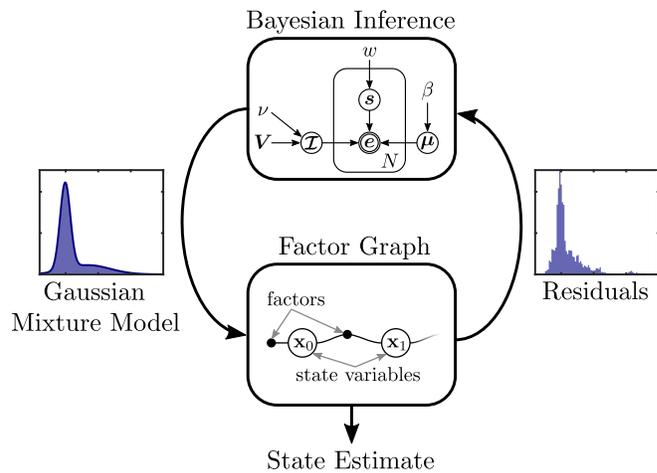


Fig. 1. The combination of Bayesian inference and factor graph optimization allows robust state estimation even if the true error distribution is unknown. Our approach incrementally learns the non-Gaussian error distribution simultaneously to the state estimation process.

graph optimization to achieve an even better robustness and accuracy. Due to an incremental construction of the GMM, we preserve the real-time capability of the original approach and eliminate existing drawbacks.

Since the proposed algorithm is the result of an incremental research process, we want to give an overview over prior work in Sec. II. How GNSS localization and Gaussian mixtures are connected to graph based optimization is explained in Sec. III, while Sec. IV gives an overview about the algorithms that are used to estimate Gaussian mixtures from data. Then we combine these techniques to the proposed Incremental Variational Mixture approach in Sec. V before we evaluate it in Sec. VI using several real world GNSS datasets and finally conclude.

II. PRIOR WORK

Several recently published algorithms try to solve the problem of non-Gaussian sensor fusion in different ways. From our point of view, they can be separated by two properties of their error models: If they are using static or dynamic ones and whether the models are Gaussian or not. We summarize existing work in Tab. I, categorized by the error model’s properties. For example, the standard least squares approach [10] is a static Gaussian method since it uses a Gaussian error model that is predefined.

TABLE I
ROBUST FACTOR GRAPH ALGORITHMS CATEGORIZED REGARDING THE
APPLIED ERROR MODEL

| Error Model | Static | Dynamic |
|--------------|---|---|
| Gaussian | Standard Factor Graph | SC [11] DCE [12] |
| Non-Gaussian | DCS [13] cDCE [12] Max-Mix. [4] Sum-Mix. [5] | Self-tuning M-Est. [14] Self-tuning Mix. [6] Adaptive Mix. [7] Batch Cov. Est. [8] |

With Switchable Constraints (SC) [11] and its modification Dynamic Covariance Estimation (DCE) [12] exist two algorithms that implement Gaussian but dynamic error models. While SC introduces tunable weights for each Gaussian, DCE estimates the covariance of the distribution. Both are able to handle Gaussian distributions with outliers but tend to fail for heavily multimodal distributions that can occur when GNSS is used in urban canyons.

A common robust method, not only in optimization, is the application of predefined non-Gaussian weight functions, so called M-estimators. These static non-Gaussian methods include not only the well-known Huber or Tukey estimators but also techniques like Dynamic Covariance Scaling (DCS) [13] or the closed form of DCE (cDCE) [12]. Usually they come with a tuning parameter to adjust the trade-off between robustness and well-behaved convergence. Beside M-estimators, there are two approaches to include different distributions in the optimization problem. Max-Mixtures [4] allows approximated inference over a Gaussian mixture by replacing the sum-operator by a max-operator. The work of Rosen [5] includes all kinds of continuous probability distributions including Gaussian mixtures. However, all methods with static error models require exact knowledge about the expected error distribution. For GNSS in changing environments, this information is often not available.

Algorithms that apply non-Gaussian error models and estimate their parameter dynamically are a relatively new class of robust solutions to non-linear optimization. The authors of [14] were the first who described an algorithm that is capable to estimate the optimal parameter of a certain group of M-estimators, using adaptive importance sampling. Later in [6], the parameters of a Gaussian mixture model (GMM) were optimized simultaneously during the estimation process. In recent work [7], Expectation-Maximization (EM) was applied to estimate the GMM in a more reliable way. Also, Wang et al. introduced a similar algorithm in the field of laser scan matching [15]. They estimate a mixture of exponential power to describe the residuals of the point-to-point registration process. The prerequisite for this type of algorithms is the selection of a suitable model respectively its complexity. For GNSS applications this can be a limitation, since the complexity (e.g. the number of modes) can change with the environment over time.

Watson et al. currently proposed a novel algorithm, based on the infinite Gaussian mixture model, which is adaptive regarding its number of components [8]. Since the applied

Gibbs sampling is not real-time capable, the approach can only be applied to batch problems. For the GNSS localization of an autonomous system this is a harsh limitation, since it usually has to be solved in real time. Also, the algorithm applies one common model for the whole dataset, so it cannot address time dependent changes.

While existing algorithms tune a predefined distribution to approximate the estimation error, we want to go one step further and learn its structure and parametrization online. Therefore, we want to propose an algorithm that is able to exploit the changing complexity of the error distribution in urban scenarios. Compared to the most recent self-tuning mixtures algorithm [7], our approach is novel in the following ways:

- 1) The proposed algorithm incrementally constructs a Gaussian mixture with a (theoretically) unlimited number of components.
- 2) We are able to remove unused Gaussian components to preserve real-time capability.
- 3) To ensure a numerically stable solution, we apply variational Bayesian inference instead of Expectation-Maximization to estimate the GMM.

With this combination of techniques, we propose the first real-time capable sensor fusion algorithm, that is able to learn the complexity and parametrization of the measurement's error distribution over time.

III. GNSS LOCALIZATION AS FACTOR GRAPH

In the following section, we want to give a brief overview how the GNSS problem can be described as factor graph and how non-Gaussian error distributions can be represented. Factor graphs, a graphical representation of least squares problems, are widely used in robotics. Therefore, we assume basic knowledge about their theoretical properties and refer readers to [10] for a more detailed introduction.

A. Factor graphs and Least Squares

The GNSS localization of a moving system can be described as optimization problem (1), where \mathbf{Z} is a set of pseudorange and odometry measurements and \mathbf{X} is the set of estimated states including the vehicle's position. \mathbf{X}^* is the most likely set of states according to the set of measurements. By applying the Bayes theorem, it can be written as maximum a posteriori (MAP) problem (2) with $\hat{\mathbf{X}}$ as MAP estimator of the true state variables.

$$\mathbf{X}^* = \underset{\mathbf{X}}{\operatorname{argmax}} \mathbf{P}(\mathbf{X}|\mathbf{Z}) \quad (1)$$

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmax}} \mathbf{P}(\mathbf{Z}|\mathbf{X})\mathbf{P}(\mathbf{X}) \quad (2)$$

At each time step t , one odometry measurement \mathbf{z}_t^{odo} and multiple pseudorange measurements $\mathbf{z}_{t,i}^{pr}$, each corresponding to the i th satellite, are available. The set of state variables can be separated in two subsets that consist of the 3D pose $\mathbf{x}_t^{x,y,z,\phi}$ and the GNSS specific clock error state $\mathbf{x}_t^{\delta,\dot{\delta}}$. The Cartesian coordinates x, y, z are in the earth-centered-earth-fixed (ECEF) frame, while ϕ denotes the rotation around the

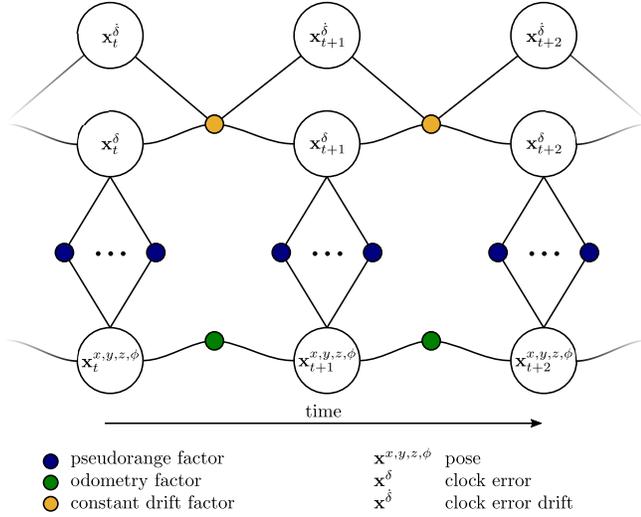


Fig. 2. Factor graph model of the GNSS localization problem. Big circles represent the estimated state variables and small colored dots the error functions which are factors of (3). The pseudorange factors suffer from non-Gaussian NLOS errors, therefore robust error models have to be applied.

vehicles upright axis. The clock error δ and its derivation $\dot{\delta}$ are required for the constant clock error drift model (CCED). By assuming uninformative priors, $\mathbf{P}(\mathbf{X})$ can be omitted and the posterior likelihood is expanded to (3). A detailed explanation of these error terms can be found in [6]. Fig. 2 shows a small example of the corresponding factor graph.

$$\begin{aligned} \mathbf{P}(\mathbf{Z}|\mathbf{X}) \propto & \prod_t \left[\prod_i \mathbf{P}(\mathbf{z}_{t,i}^{pr} | \mathbf{x}_t^{x,y,z,\delta}) \right] \\ & \cdot \prod_t \mathbf{P}(\mathbf{z}_t^{odo} | \mathbf{x}_{t+1}^{x,y,z,\phi}, \mathbf{x}_t^{x,y,z,\phi}) \\ & \cdot \prod_t \mathbf{P}(\mathbf{x}_{t+1}^{\delta,\dot{\delta}} | \mathbf{x}_t^{\delta,\dot{\delta}}) \end{aligned} \quad (3)$$

Instead of maximizing the likelihood (2) directly, $\hat{\mathbf{X}}$ is usually estimated by minimizing the negative log likelihood:

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \sum_n -\ln(\mathbf{P}(\mathbf{z}_n | \mathbf{x}_n)) \quad (4)$$

One common index n is used for all factors here. By defining $\mathbf{P}(\mathbf{Z}|\mathbf{X})$ as normal distribution $\mathcal{N}(\boldsymbol{\mu}_n, \mathcal{I}_n)$ over the non-linear error function $\mathbf{e}_n = f(\mathbf{x}_n, \mathbf{z}_n)$ with mean $\boldsymbol{\mu}_n$ and information matrix \mathcal{I}_n , the problem can be formulated as non-linear least squares estimation:

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \sum_n \frac{1}{2} \left\| \mathcal{I}_n^{\frac{1}{2}} (\mathbf{e}_n - \boldsymbol{\mu}_n) \right\|^2 \quad (5)$$

As already mentioned, the Gaussian assumption does not apply to all GNSS measurements. Therefore, a reliable state estimation requires more advanced distributions.

B. Factor Graphs and Gaussian Mixtures

Since empirical sensor distribution can be asymmetric or even multimodal, a class of distributions is required that is able to represent these properties. For this work we choose GMMs, because they fulfill this condition and have several

positive properties regarding their application in autonomous systems [16]. To apply them to a least squares problem, we use a solution proposed in [5] that allows almost arbitrary non-Gaussian distributions $\mathbf{P}(\mathbf{z}_n | \mathbf{x}_n)$, using the following:

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmin}} \sum_n \left\| \sqrt{-\ln \left(\frac{\mathbf{P}(\mathbf{z}_n | \mathbf{x}_n)}{\gamma} \right)} \right\|^2 \quad (6)$$

The normalization constant γ ensures a numerically stable solution. We reference the approach as Sum-Mixture (SM) in difference to Max-Mixture of [4], which only approximates a GMM. In this work, a K -component Gaussian mixture defined by (7) is used to describe the pseudorange error $\mathbf{e}_{t,i}^{pr} = f(\mathbf{x}_t^{x,y,z,\delta}, \mathbf{z}_{t,i}^{pr})$. Each Gaussian component k is scaled with a normalized weight w_k .

$$\mathbf{e}_{t,i}^{pr} \sim \sum_{k=1}^K w_k \cdot \mathcal{N}(\boldsymbol{\mu}_k, \mathcal{I}_k) \quad \text{with} \quad \sum_{k=1}^K w_k = 1 \quad (7)$$

Since our approach is not limited to the pseudorange error, we use the generic error \mathbf{e} instead of $\mathbf{e}_{t,i}^{pr}$ for the following equations.

The likelihood of the GMM is defined by:

$$\begin{aligned} \mathbf{P}(\mathbf{z}_n | \mathbf{x}_n) = & \sum_k c_k \cdot \exp \left(-\frac{1}{2} \left\| \mathcal{I}_k^{\frac{1}{2}} (\mathbf{e} - \boldsymbol{\mu}_k) \right\|^2 \right) \\ \text{with } c_k = & w_k \cdot \det \left(\mathcal{I}_k^{\frac{1}{2}} \right) \end{aligned} \quad (8)$$

Therefore, the normalization constant can be set to $\gamma = \sum_k c_k$. For a more detailed derivation of these equations we want to refer the reader to the original work [5] and to [7] for its application to GMMs.

IV. MIXTURE MODEL ESTIMATION

As shown in prior work [7, 8, 14], the residual $\mathbf{e}(\mathbf{x}, \mathbf{z})$ of the estimation problem can be used to get an approximation of the measurement's true distribution. To describe the measurement error with a K -component Gaussian mixture model, the GMM's parameters $\boldsymbol{\theta} = \{w, \boldsymbol{\mu}, \mathcal{I}\}$ have to be estimated with (9) from N samples of \mathbf{e} .

$$\boldsymbol{\theta}^* = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \mathbf{P}(\boldsymbol{\theta} | \mathbf{e}) \quad (9)$$

The set of parameters $\boldsymbol{\theta}$ includes a weight w_k , mean $\boldsymbol{\mu}_k$ and information matrix \mathcal{I}_k for each component k of K . Since the probability that measurement n belongs to component k is not known, the corresponding hidden parameter $s = \{s_{kn}\}$ have to be included in the estimation problem. The directed graphs of two possible interpretations are shown in Fig. 3 and explained in the following subsections.

A. Expectation Maximization

One interpretation, shown in Fig. 3a, is to treat $\boldsymbol{\theta}$ as parameters and describe the estimation as maximum-likelihood problem:

$$\begin{aligned} \boldsymbol{\theta}^* = & \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \mathbf{P}(\mathbf{e} | \boldsymbol{\theta}) \\ \text{with } \mathbf{P}(\mathbf{e} | \boldsymbol{\theta}) = & \int \mathbf{P}(\mathbf{e}, \mathbf{s} | \boldsymbol{\theta}) d\mathbf{s} \end{aligned} \quad (10)$$

TABLE II
LEGEND OF FIG. 3

| Symbol | Name |
|---------------|--|
| \mathcal{I} | Information Matrix of the GMM Components |
| μ | Mean of the GMM Components |
| w | Weight of the GMM Components |
| e | Measurement Error |
| s | Hidden Variable |
| ν | Wishart Degree of Freedom |
| \mathbf{V} | Wishart Scale Matrix |
| β | Information Matrix Scaling |
| N | Number of Measurements |

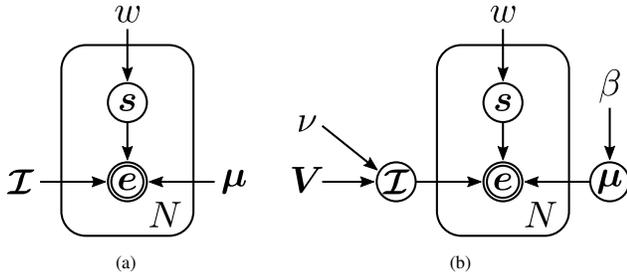


Fig. 3. Directed graphs in plate notation of the maximum-likelihood (a) and the Bayesian (b) interpretation of a Gaussian mixture. The circles mark hidden random variables and the double circles the observed ones. Unmarked letters are parameters of the corresponding model. A legend is provided by Tab. II.

Since the integral is intractable, the iterative Expectation-Maximization (EM) algorithm [17] splits the problem into two subproblems: The E-step estimates the hidden variable s based on an initial guess of θ . The M-step estimates the maximum likely set of parameters θ based on the hidden variables that were estimated before. As demonstrated in [7], the EM can be applied in an alternating sequence with the least squares optimization of the original state estimation problem (1). On the downside, the choice of the parameter K is not trivial because it depends on the complexity of the true distribution of e . While a too small K results in less robustness, a too large number of components leads to numerical instabilities. These instabilities can occur, if only a few samples are responsible for a specific component. There exists a variety of EM derivations that aim to overcome this limitation like the greedy EM [18] or the split and merge EM [19], but their parametrization can be even more difficult. Even a simple merging criterion is not easy to chose, as the author of [20] demonstrates.

B. Bayesian Methods

A fully Bayesian interpretation of the GMM optimization (9), treats the distribution parameters θ as random variables with corresponding distributions. After adding a set of priors $\mathbf{P}(\theta)$, the estimation problem can be written as:

$$\theta^* = \operatorname{argmax}_{\theta} \frac{\mathbf{P}(e|\theta)\mathbf{P}(\theta)}{\int \mathbf{P}(e|\theta)\mathbf{P}(\theta)d\theta} \quad (11)$$

There exist different suggestions for the choice of this priors. In difference to the more common model used in [8, 16, 21] or [22, p.475], we follow the suggestion of the authors of [9] and omit the Dirichlet prior of the mixture's weight w .

Fig. 3b shows the corresponding graph with the remaining Wishart prior (13) for the information matrix and the normal prior (12) for the mean of the GMM. The matrix \mathbf{I} is the identity matrix here.

$$\mu \sim \mathcal{N}(0, \beta_0 \mathbf{I}) \quad (12)$$

$$\mathcal{I} \sim \mathcal{W}(\mathbf{V}_0, \nu_0) \quad (13)$$

To omit the prior of w comes with two advantages, one obvious is the reduced number of hyperparameters which makes the algorithm easier to apply. The other one is the possibility to reduce weights to zero, which allows us to remove unused components from the model. Since the integral in (11) can not be calculated directly, approximate solutions like Markov Chain Monte Carlo (MCMC) methods or variational Bayesian inference (VBI) have to be applied. MCMC algorithms, such as Gibbs sampling [23] used in [16] and [8], approximate the integral over the set of parameters θ by generating many samples of them from their estimated distribution. This process is computationally expensive and therefore rather unsuitable for real-time applications, which leads us to variational inference.

The key idea of VBI is to approximate the posterior $\mathbf{P}(\theta|e)$ with a distribution $\mathbf{Q}(\theta)$.

$$\mathbf{P}(\theta|e) \approx \mathbf{Q}(\theta) \quad (14)$$

To find the distributions over all hidden variables θ , it is assumed that they can be partitioned in j independent groups $\{\theta_j\}$. According to these groups, $\mathbf{Q}(\theta)$ can be factorized, as shown in (15), and each factor $\mathbf{Q}_j(\theta_j)$ can be estimated separately. For the model proposed in [9], the groups are mean μ , information matrix \mathcal{I} and the correspondence variable s .

$$\mathbf{Q}(\theta) = \prod_j \mathbf{Q}_j(\theta_j) = \mathbf{Q}_{\mathcal{I}}(\mathcal{I}) \mathbf{Q}_{\mu}(\mu) \mathbf{Q}_s(s) \quad (15)$$

Since we use exactly the same model that is proposed in [9], we refer interested readers to the original work. For a more gentle introduction we can also recommend the tutorial paper [24].

Nevertheless, there are still some drawbacks of Bayesian inference in general and of the concrete VBI solution. Bayesian estimators require a set of predefined priors to represent the knowledge that is available in advance. If there is little to no prior knowledge about the estimated parameters, it can be hard to specify meaningful priors. This work can not provide a general solution for this problem, but we try to find a good trade-off between uninformative priors and prior knowledge that exists about NLOS errors. Variational inference require a predefined number of Gaussian components similar to the maximum-likelihood approach. However, our approach is robust against an exaggerated number of components, since the weight of unused ones drops to zero without causing singularities.

As a last statement of this section, we want to emphasize that to our knowledge no "best" algorithm exists to estimate a Gaussian mixture from empirical data. Nevertheless, VBI seems to be well suited for the problem we want to address.

V. LEARNING COMPLEXITY ONLINE

In this section, we describe the complete algorithm which is composed of robust factor graph optimization and variational Bayesian inference. The factor graph applies the GMM to robustly estimate the system's state and the VBI estimates the mixtures based on the empirical distribution of the sensor. An overview of the general approach is given by Fig. 1 on the first page and an algorithmic description is provided with Alg. 1 and Alg. 2.

As explained in [7] the estimation problem (16) can be described as EM algorithm with the set of measurements \mathbf{Z} as observed variable and the parameters of the mixture model θ as hidden ones.

$$\hat{\mathbf{X}} = \underset{\mathbf{X}}{\operatorname{argmax}} \mathbf{P}(\mathbf{X}|\mathbf{Z})$$

$$\text{with } \mathbf{P}(\mathbf{X}|\mathbf{Z}) = \int \mathbf{P}(\mathbf{X}, \theta|\mathbf{Z})d\theta \quad (16)$$

Again, the intractable integral is solved by alternately estimating the expected value of the hidden parameters θ (E-step) and the state variables (M-step). Since the problem is formulated as sliding window estimation, one iteration per time step is enough to achieve good convergence.

In the E-step, the posterior probability of θ is calculated with VBI based on a previously determined MAP estimate of the error $\hat{\mathbf{e}}$:

$$\hat{\theta} = \underset{\theta}{\operatorname{argmax}} \mathbf{P}(\theta|\hat{\mathbf{e}}_{t-1}) \quad (17)$$

The desired expected value $\mathbb{E}(\hat{\theta}_t)$ is computed during the variational inference. Again, further details are available in [9].

The M-step uses the estimated distribution parameter $\mathbb{E}(\hat{\theta}_t)$ and applies it to the least squares optimization according to the Sum-Mixture formulation (6). After the optimization, the non-linear error function \mathbf{e} can be used to get a MAP estimate of the error \mathbf{e} as shown in:

$$\hat{\mathbf{X}}_t = \underset{\mathbf{X}}{\operatorname{argmax}} \mathbf{P}(\mathbf{X}|\mathbf{Z}_t, \mathbb{E}(\hat{\theta}_t)) \quad (18)$$

$$\hat{\mathbf{e}}_t = \mathbf{e}(\hat{\mathbf{X}}_t, \mathbf{Z}_t) \quad (19)$$

The complete procedure is described with Alg. 2.

To adapt the mixture's number of components K to the complexity of the empirical distribution, we add one new component in each time step, starting with $K = 2$ at $t = 0$. This leads to a fast rising number of components. However, through the variational inference, the weight of unused components drops to zero. Therefore, components with a weight below a predefined threshold w_{min} can be removed. We set this pruning threshold to $1/N$ with N as the number of measurements. Each added component is initialized according to its prior distribution with zero mean $\mu_0 = 0$ and the information matrix $\mathcal{I}_0 = \nu_0 \mathbf{V}_0^{-1}$. With this simple approach, we are able to choose the right number of components just by "saturating" the mixture with components. Since the saturation point depends on the parametrization

of the Gauss and Wishart prior, a careful parametrization is essential for a good performance.

To ensure real time capability even for critical applications, we specify an optional upper limit K_{max} for the number of components. If this limit is exceeded, the component with the lowest weight is removed before a new one is added. The online complexity learning is described by Alg. 1.

Algorithm 1: Complexity Learning (CL)

Data: $\hat{\mathbf{e}}_t, \hat{\theta}_{t-1}$
Result: $\hat{\theta}_t$
begin Initialization of $\hat{\theta}_t$
 $\hat{\theta}_t = \hat{\theta}_{t-1}$;
 if $K \geq K_{max}$ **then**
 Remove the smallest component from $\hat{\theta}_t$;
 end
 Add new component to $\hat{\theta}_t$;
end
begin Estimation of $\hat{\theta}_t$
 repeat
 Perform one iteration of VBI [9];
 if any $w_k < w_{min}$ **then**
 Remove component k from $\hat{\theta}_t$;
 end
 Increment iteration count i ;
 Compute relative likelihood ΔL ;
 until $i > i_{max}$ or $\Delta L < \Delta L_{min}$;
end

Algorithm 2: Incremental Variational Mixture (IVM)

Data: \mathbf{Z}
Result: $\hat{\mathbf{X}}$
Initially estimate $\hat{\mathbf{X}}_{t=0}$ with (5);
Perform Alg. 1 twice to get $\hat{\theta}_{t=0}$;
foreach time step t **do**
 Add $\mathbf{z}_t, \mathbf{x}_t$ to $\mathbf{P}(\mathbf{Z}|\mathbf{X})$;
 Remove \mathbf{z}, \mathbf{x} older than 60s from $\mathbf{P}(\mathbf{Z}|\mathbf{X})$;
 Calculate $\hat{\mathbf{e}}_t$;
 Perform Alg. 1 to get $\hat{\theta}_t$;
 Update $\mathbf{P}(\mathbf{Z}|\mathbf{X})$ with $\hat{\theta}_t$;
 Estimate $\hat{\mathbf{X}}_t$ by optimizing $\mathbf{P}(\mathbf{Z}|\mathbf{X})$;
end

VI. EVALUATION

In this section, we demonstrate the performance of the proposed Incremental Variational Mixture (IVM) algorithm on several real world GNSS datasets. The resulting trajectories are shown in Fig. 4. We do not only compare against the robust state-of-the-art approaches DCS [13] and cDEC [12] but also show the impact of VBI and the proposed Complexity Learning (CL) mechanism. Beside a comparison based on position error metrics, we want to demonstrate the sensitivity

TABLE III
ACCURACY AND RUNTIME OF THE FINAL EVALUATION.

| Algorithm | Chemnitz | | Berlin PP | | Berlin GM | | Frankfurt MT | | Frankfurt WT | |
|-----------------|--------------|----------|--------------|----------|--------------|----------|--------------|----------|--------------|----------|
| | ATE [m] | Time [s] |
| Gaussian | 30.0 | 58.6 | 29.2 | 9.5 | 13.38 | 49.7 | 30.97 | 58.1 | 23.54 | 31.9 |
| DCS [13] | 4.403 | 54.9 | 25.04 | 14.7 | 19.11 | 64.5 | 13.25 | 69.5 | 11.39 | 35.2 |
| cDCE [12] | 4.326 | 54.5 | 17.91 | 14.3 | 14.59 | 65.3 | 14.93 | 73.6 | 11.12 | 34.9 |
| SM+EM [7] | 2.378 | 102.0 | 12.45 | 39.8 | 13.88 | 163.0 | 10.72 | 136.0 | 6.42 | 76.5 |
| SM+VBI | 3.106 | 119.0 | 12.4 | 33.3 | 13.55 | 186.0 | 11.23 | 131.0 | 4.117 | 84.4 |
| SM+EM+CL | 6.723 | 172.0 | 12.23 | 45.0 | 13.37 | 312.0 | 10.41 | 214.0 | 9.706 | 152.0 |
| IVM (SM+VBI+CL) | 2.48 | 122.0 | 11.56 | 59.4 | 10.99 | 352.0 | 8.586 | 248.0 | 4.626 | 146.0 |

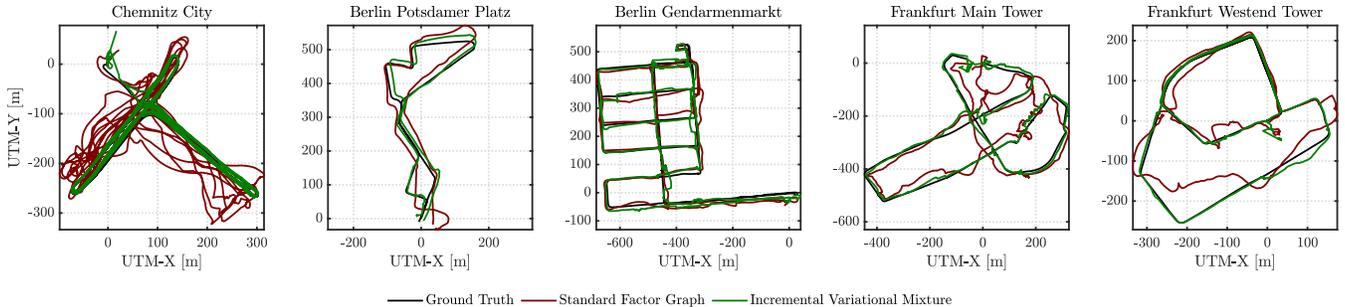


Fig. 4. Resulting UTM trajectories of the non-robust factor graph (red) and the proposed Incremental Variational Mixture algorithm (green). The distortions caused by NLOS are significantly reduced.

of our approach to its parametrization, which is important for practical applications. To proof the performance under real time conditions, we perform the estimation under online conditions. This means, results are calculated and stored without information from future measurements.

A. The Datasets

We use five different datasets, that consist of raw pseudo-range measurements from a mass market receiver and wheel odometry from measurement vehicle. These datasets are collected in different urban areas in Germany and have an accumulated length of 18.6 km respectively 92 min. A precise ground truth is provided by a combination of differential GNSS and a tactical grade inertial measurement unit. Technical details are published in [25] and the datasets themselves are available online ¹.

B. Parametrization

Since the choice of parameters has a significant impact on the performance of the proposed algorithms, we want to explain them clearly. The noise parameters of the factor graph are identical to our previous work, therefore we want to refer to [7] for detailed information.

To apply the variational inference, several parameters have to be specified that are summarized in Tab. IV. Not all of them are equally important, especially the convergence criteria and the pruning threshold w_{min} have just a small impact on the overall performance. The optional maximum of Gaussian components K_{max} is also easy to parametrize, it has to be high enough to reflect the worst case complexity of the

TABLE IV
PARAMETERS OF THE INCREMENTAL VARIATIONAL MIXTURE ALGORITHM

| Parameter Name | Symbol | Value |
|---------------------------------|------------------|------------------------------------|
| Wishart Prior Degree of Freedom | ν_0 | 2 |
| Wishart Prior Scale Matrix | \mathbf{V}_0 | $\text{var}(\hat{\mathbf{e}})/\nu$ |
| Normal Prior Information Matrix | β_0 | 10^{-6} m^{-2} |
| Pruning Threshold | w_{min} | $1/N$ |
| Max. Iterations | i_{max} | 1000 |
| Min. Δ Likelihood | ΔL_{min} | 10^{-6} |
| Max. Number of Components | K_{max} | 8 |

empirical distribution. A value of $K_{max} = 8 \dots 12$ should be enough for all practical applications and if the algorithm is applied to a fully unknown problem, the limit can be omitted. The information matrix of the normal prior β_0 defines the initial uncertainty about each component's mean. Since NLOS errors over 1 km are very unlikely, a value of $\beta_0 = 1/(1 \text{ km})^2$ seems to be sufficient.

More difficult is the parametrization of the Wishart prior $\mathcal{W}(\mathbf{V}_0, \nu_0)$. The scale matrix \mathbf{V}_0 in combination with the degree of freedom ν_0 determines the expected information matrix of the estimated mixture components:

$$\mathbb{E}(\mathcal{I}) = \nu_0 \cdot \mathbf{V}_0^{-1} \quad (20)$$

A coarse guess of the information matrix can be estimated directly from the error $\hat{\mathbf{e}}$ by calculating its variance. So, a meaningful prior can be defined with:

$$\mathbf{V}_0 = \frac{\text{var}(\hat{\mathbf{e}})}{\nu_0} \quad (21)$$

¹<http://mytuc.org/GNSS>

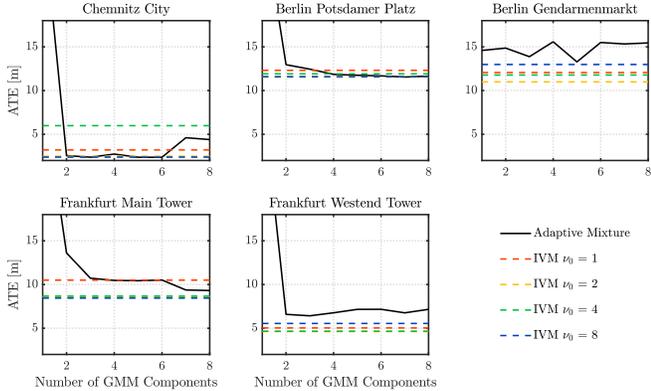


Fig. 5. Comparison between the Adaptive Mixture approach [7] and the proposed Incremental Variational Mixture (IVM) algorithm on all datasets. The solid black line shows the mean ATE for different choices of the number of mixture components. Each dotted line corresponds to a run of the IVM with a specific ν_0 . Since the number of components is variable here, we draw it as horizontal line. The comparison shows that IVM outperforms the other algorithm for a ν_0 of 2.

For the positive integer ν_0 , a straightforward approximation is not possible. The authors of the variational model [9] recommend an informative Wishart prior, which means value of $\nu_0 = 1$ or at least close to 1. We evaluated the proposed algorithm with different values and visualized the localization error in Fig. 5. Therefore, we chose a prior with $\nu_0 = 2$ to achieve the best performance over all datasets. The reason for the strong impact of ν_0 to the overall result is the convergence of the VBI algorithm. An increased ν_0 improves the convergence speed and increases the number of removed mixture components. With $\nu_0 = 1$, the number of components over-fits the empirical distribution which leads to more local minima in the state estimation problem and therefore reduced accuracy. Although, the performance of different values of ν_0 is not fully consistent over all datasets, the choice of $\nu_0 = 2$ generalizes at least for the tested ones. The comparison to the EM based algorithm in Fig. 5 shows that its fixed number of components K also have to be chosen carefully. We set $K = 3$ for a fair comparison since it is the best choice over all datasets.

C. Implementation Details

We implemented the algorithm as part of our robust sensor fusion library libRSF². The VBI algorithm will be published as open source in a future release. The non-linear least squares optimization is based on the Ceres solver [26] and the GMM estimation is implemented in C++ using Eigen [27] without multi-threading. All tests were performed on a Intel i7-7700 system.

D. Results

Metric of our comparison is the absolute trajectory error (ATE). We define the ATE as euclidean distance between estimated trajectory and ground truth in the local XY-plane. Tab. III summarizes the mean ATE as well as the runtime of the evaluated algorithms. Along with the proposed IVM,

we compare different combinations of self-tuning algorithms. Common is the Sum-Mixture (SM) method to represent a GMM, but we differentiate between EM and VBI estimation and whether the proposed Complexity Learning (CL) is applied or not. Please notice that SM+EM equals the Adaptive Mixture approach from [7] and SM+VBI+CL is the proposed IVM algorithm.

Both VBI and CL seem to have a positive influence regarding the localization accuracy. Although, only in combination, they are able to consistently reduce the ATE over all datasets. Therefore, we would prefer the proposed IVM for applications where a maximum precision is required. Drawback of the CL is increased runtime of the algorithm, nevertheless it is 5 times faster than the recording time of the datasets.

If runtime is more important, we would tend to prefer the Adaptive Mixture (SM+EM) approach, since it offers a solid performance without the trouble of parameterizing the variational priors.

The results of the M-estimators DCS and cDCE show improvements compared to the non-robust factor graph, but both fall behind the GMM based methods.

VII. CONCLUSION

The proposed algorithmic approach, allows to apply least squares optimization to sensor fusion problems with non-Gaussian and time dependent error distributions like GNSS localization in urban environments. We not only adapt the parameters of the Gaussian mixture representation, we also learn the right number of components to represent its complexity. The applied variational Bayesian inference allows a numerically stable and real time capable solution.

The comparison against several state-of-the-art algorithms were performed on a set of open access GNSS datasets that reflect the localization of an autonomous vehicle in an urban environment. Our approach demonstrated a superior estimation quality over the majority of datasets in combination with an increased but acceptable computation time.

The described connection between state estimation and adaptive error models opens a wide field of possible future improvements. Since the parametrization of the variational priors is the most critical drawback currently, we will address it in future work. Variational Bayesian inference also offers further methods to adaptively split and merge Gaussian components which could be explored in this context. Finally, a broader evaluation on different domains like SLAM would be useful to validate the approach and the choice of parameters.

REFERENCES

- [1] L.-T. Hsu, "Analysis and modeling gps nlos effect in highly urbanized area," *GPS Solutions*, 2017.
- [2] J. Breßler, P. Reisdorf, M. Obst, and G. Wanielik, "GNSS positioning in non-line-of-sight context—a survey," in *Proc. of Intl. Conf. on Intelligent Transportation Systems (ITSC)*, 2016.
- [3] N. Zhu, J. Marais, D. Bétaille, and M. Berbineau, "GNSS position integrity in urban environments: A review of literature," *IEEE Transactions on Intelligent Transportation Systems*, 2018.
- [4] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," in *Proc. of Robotics: Science and Systems (RSS)*. Sydney, Australia: Robotics: Science and Systems Foundation, 2012.

²<http://mytuc.org/libRSF>

- [5] D. M. Rosen, M. Kaess, and J. J. Leonard, "Robust incremental online inference over sparse factor graphs: Beyond the Gaussian case," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2013.
- [6] T. Pfeifer and P. Protzel, "Robust sensor fusion with self-tuning mixture models," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.
- [7] —, "Expectation-maximization for adaptive mixture models in graph optimization," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2019.
- [8] R. Watson, J. Gross, C. Taylor, and R. C. Leishman, "Batch measurement error covariance estimation for robust localization," in *Proc. of Intl. Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+)*, 2018.
- [9] A. Corduneanu and C. Bishop, "Variational bayesian model selection for mixture distributions," in *Proc. of Intl. Conf. on Artificial Intelligence and Statistics (AISTATS)*. Morgan Kaufmann, 2001.
- [10] F. Dellaert, M. Kaess *et al.*, "Factor graphs for robot perception," *Foundations and Trends in Robotics*, 2017.
- [11] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose graph slam," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2012.
- [12] T. Pfeifer, S. Lange, and P. Protzel, "Dynamic covariance estimation – a parameter free approach to robust sensor fusion," in *Proc. of Intl. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2017.
- [13] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2013.
- [14] G. Agamennoni, P. Furgale, and R. Siegwart, "Self-tuning m-estimators," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2015.
- [15] D. Wang, J. Xue, T. Zhongxing, Y. Zhong, D. Cui, S. Du, and N. Zheng, "Accurate mix-norm-based scan matching," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2018.
- [16] D. M. Rosen and J. J. Leonard, "Nonparametric density estimation for learning noise distributions in mobile robotics," in *1st Workshop on Robust and Multimodal Inference in Factor Graphs, ICRA*, 2013.
- [17] 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)*, 1977.
- [18] N. Vlassis and A. Likas, "A greedy em algorithm for gaussian mixture learning," *Neural Processing Letters*, 2002.
- [19] N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton, "Split and merge em algorithm for improving gaussian mixture density estimates," *Journal of VLSI signal processing systems for signal, image and video technology*, 2000.
- [20] C. Hennig, "Methods for merging gaussian mixture components," *Advances in data analysis and classification*, 2010.
- [21] D. M. Blei, M. I. Jordan *et al.*, "Variational inference for dirichlet process mixtures," *Bayesian analysis*, 2006.
- [22] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer: New York, 2006.
- [23] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1984.
- [24] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for bayesian inference," *IEEE Signal Processing Magazine*, 2008.
- [25] P. Reisdorf, T. Pfeifer, J. Breßler, S. Bauer, P. Weissig, S. Lange, G. Wanielik, and P. Protzel, "The problem of comparable gnss results – an approach for a uniform dataset with low-cost and reference data," in *Proc. of Intl. Conf. on Advances in Vehicular Systems, Technologies and Applications (VEHICULAR)*, 2016.
- [26] S. Agarwal, K. Mierle, and Others, "Ceres Solver," <http://ceres-solver.org>.
- [27] G. Guennebaud, B. Jacob *et al.*, "Eigen v3," <http://eigen.tuxfamily.org>, 2010.