A Generic Approach for Robust Probabilistic Estimation with Graphical Models

Niko Sünderhauf and Peter Protzel
Department of Electrical Engineering and Information Technology
Chemnitz University of Technology, Germany
Email: niko.suenderhauf@etit.tu-chemnitz.de

***Abstract*—Probabilistic estimation using graphical models plays an important role in today's intelligent and autonomous systems. This paper summarizes our work on *robust* probabilistic estimation using such models. This robustness, i.e. the algorithmic fault-tolerance in the presence of outliers is crucial for any autonomous system aiming at long-term operation. We show how probabilistic estimation using factor graphs can be made tolerant against outliers in the underlying data and demonstrate the feasibility of the proposed generic scheme in the domains of SLAM and satellite-based localization.**

I. Introduction

Autonomous systems aiming at long-term operation have to be fault-tolerant. This does not only apply to the hardware and the mechanical design of an autonomous system, but of course to its software and algorithms as well. Fault-tolerant or *robust* algorithms in this sense are methods that can cope with unmodelled effects, inadequacies and errors in the underlying data without breaking or diverging. Probabilistic estimation based on optimization (i.e. least squares methods) has become a widespread tool in many different domains in autonomous robotics (e.g. [6, 14]). At the same time, however, it is well-known that least squares optimization suffers from an inherent vulnerability against outliers. It is therefore worthwhile to explore how the robustness and outlier tolerance of such algorithms can be improved in order to increase the overall system's performance and capabilities.

In the following, we shortly review the concept of factor graphs and how they can be used to represent different kinds of probabilistic estimation problems. We then point out the problem of the lacking robustness against outliers before the proposed generic approach for robust estimation is explained and results the domains of SLAM and GNSS-based localization are presented.

II. Factor Graphs

Factor graphs are bipartite undirected graphs and have been proposed by [5] as a general tool to model factorizations of large functions with many variables into smaller local subsets. The idea can be applied to general probabilistic estimation problems where the joint conditional probability distribution one wants to estimate can be expressed as a product over several single *factors*. These factors are formed according to the dependency structure between the hidden variables $\mathcal{X}$ and the given evidence $\mathcal{Z}$ (e.g. measurements or a-priori knowledge):

$$P(\mathcal{X}|\mathcal{Z}) = \prod_i P_i(\bar{\mathcal{X}}_i|\bar{\mathcal{Z}}_i) \tag{1}$$

where $\bar{\mathcal{X}}_i \subseteq \mathcal{X}$ and $\bar{\mathcal{Z}}_i \subseteq \mathcal{Z}$ are arbitrary subsets of $\mathcal{X}$ and $\mathcal{Z}$ respectively.

Since factor graphs are bipartite, they contain two sets of nodes: one for the hidden variables and the other for the probabilistic relations (the factors) between them.

A. Finding the Maximum a Posteriori Solution

The maximum a posteriori (MAP) estimate of the distribution $P(\mathcal{X}|\mathcal{Z})$, i.e. the most likely variable configuration $\mathcal{X}^*$ given the data $\mathcal{Z}$, is formalized as an optimization problem of the form

$$\mathcal{X}^* = \operatorname*{argmax}_{\mathcal{X}} P(\mathcal{X}|\mathcal{Z}) = \operatorname*{argmax}_{\mathcal{X}} \prod_i P_i(\bar{\mathcal{X}}_i|\bar{\mathcal{Z}}_i) \tag{2}$$

If the single factors P_i are Gaussian, they are of the general form

$$P_i(\bar{\mathcal{X}}_i|\bar{\mathcal{Z}}_i) = \eta \exp -\frac{1}{2}\|\mathbf{e}_i(\bar{\mathcal{X}}_i, \bar{\mathcal{Z}}_i)\|^2_{\mathbf{\Sigma}_i} \tag{3}$$

where $\mathbf{e}_i(\bar{\mathcal{X}}_i, \bar{\mathcal{Z}}_i)$ is a problem-specific error function. Using this relation and taking the negative logarithm, we can transform (2) into

$$\mathcal{X}^* = \operatorname*{argmin}_{\mathcal{X}} \sum_i \|\mathbf{e}_i(\bar{\mathcal{X}}_i, \bar{\mathcal{Z}}_i)\|^2_{\mathbf{\Sigma}_i} \tag{4}$$

which is a least squares optimization problem, since we seek the minimum over a sum of squared terms. Such problems can be solved efficiently using a variety of methods like Levenberg-Marquardt, Gauss-Newton or Powell's Dog-Leg.

B. Factor Graphs for SLAM and Localization

1) Pose Graph SLAM: As an example, in pose graph SLAM the set of robot poses X is to be estimated given the control inputs U and loop closure detections. This can be factored into the following expression, where loop closures are represented by a factor between non-successive robot state nodes:

$$P(X|U) = \underbrace{\prod_i P(\mathbf{x}_{i+1}|\mathbf{u}_i, \mathbf{x}_i)}_{\text{Odometry Constraints}} \cdot \underbrace{\prod_{ij} P(\mathbf{x}_j|\mathbf{x}_i, \mathbf{u}_{ij})}_{\text{Loop Closure Constraints}} \tag{5}$$

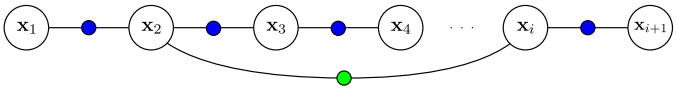


Fig. 1: Factor graph representation of the pose graph SLAM problem. The large vertices represent the unknown robot poses, while probabilistic relationships between them are expressed by the small vertices. Odometry factors are represented by blue nodes while the loop closure factor is shown in green.

The associated factor graph representation is illustrated in Fig. 1. For illustrative purposes, the variable nodes are always depicted larger than the factor nodes between them.

The maximum a posteriori solution to the distribution represented by the graph is found according to

$$X^* = \underset{X}{\operatorname{argmin}} \underbrace{\sum_i \|f(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_{i+1}\|^2_{\mathbf{\Sigma}_i}}_{\text{Odometry Constraints}} + \underbrace{\sum_{ij} \|f(\mathbf{x}_i, \mathbf{u}_{ij}) - \mathbf{x}_j\|^2_{\mathbf{\Lambda}_{ij}}}_{\text{Loop Closure Constraints}} \tag{6}$$

2) GNSS-based Localization: In GNSS[1]-based localization, the position of a vehicle is to be estimated from the pseudorange observations to satellites orbiting the earth. The state space in this problem contains at least the 3D position of the vehicle and the receiver clock error, leading to a state space that is at least 4-dimensional: $\mathbf{x} \in \mathbb{R}^4 = (x, y, z, \delta^{\text{clock}})^T$.

Several satellites are observed from every vehicle state $\mathbf{x}_t$, each providing a pseudorange measurement ρ_{tj}. Given the receiver position $\mathbf{x}_t^{x,y,z}$ and the position of the observed satellite $\mathbf{x}_{tj}^{\text{SAT}}$, the expected pseudorange measurement is given by the measurement function

$$h(\mathbf{x}_t, j) = \|\mathbf{x}_{tj}^{\text{SAT}} - \mathbf{x}_t^{x,y,z}\| + \delta^{\text{EarthRotation}} + \delta^{\text{Atmosphere}} + \mathbf{x}_t^{\delta^{\text{clock}}} \tag{7}$$

The terms $\delta^{\text{EarthRotation}}$ and $\delta^{\text{Atmosphere}}$ correct ranging effects caused by the earth's rotation and atmosphere (ionospheric and tropospheric propagation errors).

If we assume the measured pseudorange ρ_{tj} is given by the measurement function $h(\mathbf{x}_t, j)$ plus a zero-mean Gaussian error term, then the error function of a single pseudorange factor is given as

$$\|\mathbf{e}_{tj}^{\text{pr}}\|^2_{\mathbf{\Sigma}_{tj}} = \|h(\mathbf{x}_t, j) - \rho_{tj}\|^2_{\mathbf{\Sigma}_{tj}} \tag{8}$$

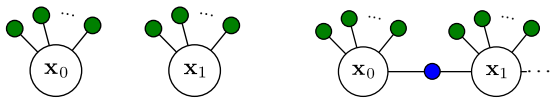
with $\mathbf{\Sigma}_{tj}$ the covariance associated to the pseudorange measurement ρ_{tj}. We can now solve for the maximum a posteriori estimates of the vehicle states $X = \{\mathbf{x}_t\}$:

$$X^* = \underset{X}{\operatorname{argmin}} \sum_{tj} \|\mathbf{e}_{tj}^{\text{pr}}\|^2_{\mathbf{\Sigma}_{tj}} \tag{9}$$

Any additional factors that account for further measurements and sensor data can be easily incorporated by extending the error function. For instance to incorporate state transition or motion model factors, we solve

$$X^* = \underset{X}{\operatorname{argmin}} \sum_{tj} \|\mathbf{e}_{tj}^{\text{pr}}\|^2_{\mathbf{\Sigma}_{tj}} + \|\mathbf{e}_t^{\text{st}}\|^2_{\mathbf{\Sigma}_t^{\text{st}}} \tag{10}$$

and so forth. Fig. 2 illustrates the resulting factor graph representations with and without state transition factors.



(a) (b)

Fig. 2: Two vehicle state nodes with their associated pseudorange factors (green) in the GNSS-based localization problem. (a) in the most general graphical model, there are no connections between the vehicle state nodes. In (b), a state transition or motion model factor joins two successive vehicle nodes.

III. Outliers in Factor Graphs

The last section provided a condensed introduction to factor graphs and how they are a convenient representation for probability distributions. The maximum a posteriori (MAP) estimate of that distribution can be found by solving the least squares optimization problem stated in (4). The single terms $\mathbf{e}_i$ that constitute the error function are built and maintained by an application-dependent front-end.

Let us now consider that some of these $\mathbf{e}_i$ that were created by the front-end, are *outlier* constraints. That is, the optimization problem consists of *inlier* constraints and *outlier* constraints which cannot be distinguished beforehand.

Where could these outliers originate from? In the context of pose graph SLAM, outlier constraints typically are false positive loop closure detections. That means, due to perceptional aliasing, i.e. self similarity of the environment, the place recognition module in the front-end fails and erroneously declares a loop closure between two places that do not correspond. This typically results in heavily distorted and unusable maps. In the context of GNSS-based localization, so called multipath satellite observations constitute the outliers. These multipath measurements arise when the direct line of sight to a satellite is blocked by a building, but the signal is received via a reflection on buildings or the ground. The signal path therefore is longer than the non-reflected direct path. This results in severely biased position estimates. Multipath errors can also occur when the signal is received multiple times, e.g. via the direct line of sight but also via a reflection. In this case, correlation errors lead to wrong pseudorange measurements that again negatively influence the resulting position estimate. In general, the reasons for outliers are dependent on the problem domain and the techniques applied in the front-end. They usually occur due to data association errors or unmodelled effects or wrong or over-simplified assumptions on the data.

[1]Global Navigation Satellite System

A. Dealing with Outliers – Related Work

The well known vulnerability of least squares methods against outliers has of course led to a number of approaches that are commonly found in the literature. In the domain of model-fitting, regression and the like, sample consensus methods like RANSAC [1], M-SAC or MLESAC [13] are state of the art, along with more recent developments. Such methods however cannot always be applied, since there might not be a well-defined minimal set to solve the problem at hand.

A better alternative seem to be so called *robust cost functions* like the Huber [4], pseudo-Huber or Blake-Zisserman [3, ch. A6.8] functions. Although such approaches reduce the influence of outliers on the optimization, we found they are not sufficient to cope with severe outliers like false positive loop closure constraints in SLAM or multipath errors in GNSS-based localization [10, 11].

IV. Robust Optimization for Probabilistic Estimation

In order to make the optimization problem (4) robust against outliers in the problem formulation, we propose the following approach. Our general idea is to make the topology of the factor graph partially variable and subject to the optimization. This way, constraints can be removed from the problem formulation as part of the optimization process.

This is achieved by *augmenting* the original optimization problem (4) by a new set of hidden variables. In addition to $\mathcal{X}^*$, which is the most likely configuration of variables given the evidence $\mathcal{Z}$ according to the MAP estimate, the augmented problem also estimates a set of so called *switch variables* S.

Each switch variable $s_i \in \mathbb{R}$ is associated with a constraint $\mathbf{e}_i$ that could potentially be an outlier. As we are going to see, each switch variable s_i determines whether the associated $\mathbf{e}_i$ is removed from the optimization problem, or if it is maintained. The switch variable exercises this influence on the constraint via a multiplicative *switch function* $\Psi : \mathbb{R} \to [0, 1]$.

The original optimization problem in (4) in its augmented form is given as:

$$\mathcal{X}^*, S^* = \underset{\mathcal{X}}{\operatorname{argmin}} \underbrace{\sum_i \|\Psi(s_i) \cdot \mathbf{e}_i\|^2_{\mathbf{\Sigma}_i}}_{\text{switched constraints}} + \underbrace{\sum_i \|\mathbf{e}_i^{\text{sp}}\|^2_{\Xi_i}}_{\text{switch prior constraints}} \tag{11}$$

Different switch functions Ψ can be defined, e.g. a step function, or a sigmoid. However, our experiments showed that a simple piecewise linear function of the form

$$\omega_i = \Psi_a^{\text{lin}}(s_i) : \mathbb{R} \to [0, 1] = \begin{cases} 0 & : s_i < 0 \\ \frac{1}{a} s_i & : 0 \leq s_i \leq a \\ 1 & : s_i > a \end{cases} \tag{12}$$

with parameter $a = 1$ is a suitable choice and superior to the previously proposed sigmoid function [11, 10].

The idea behind the switch variables is that the influence of a constraint $\mathbf{e}_i$ can be removed by driving the associated switch variable s_i to a value so that $\Psi(s_i) \approx 0$.

The influence of the switch variables can be described and understood in two equivalent ways [10, 11]: In the topological interpretation, a switch can enable or disable the constraint edge it is associated with, thus literally remove it from the graph topology. In the probabilistic interpretation, the switch variable influences the information matrix of the factor it is associated with and can drive it from its original value to zero, thus increasing the covariance associated with this factor until infinity. The resulting information matrix $\hat{\mathbf{\Sigma}}_i^{-1}$ is given as

$$\hat{\mathbf{\Sigma}}_i^{-1} = \Psi(s_{ij})^2 \cdot \mathbf{\Sigma}_i^{-1} \tag{13}$$

The switch prior constraints $\|\mathbf{e}_i^{\text{sp}}\|^2_{\Xi_i}$ in (11) are necessary to anchor the switch variables at their initial values. Since it is reasonable to initially accept all constraints proposed by the front-end, a proper and convenient initial value for all switch variables would be $s_i = 1$ when using the linear switch function Ψ_a^{lin}. For the following, we call these initial values γ_i. Like any other variable or observation in our probabilistic framework, the switch variables s_i are modelled as normally distributed Gaussian variables. The initial value is used as mean of the distribution, so that $s_i \sim \mathcal{N}(\gamma_i, \mathbf{\Xi}_i)$. Notice that a proper value for the switch prior covariance $\mathbf{\Xi}_i$ has to be determined. This value is a free parameter of the proposed robust back-end formulation. It has been shown in [10, ch. 6.4] that $\mathbf{\Xi}_i$ can be safely chosen from a range of values, independent of the problem domain or the underlying data. For all conducted experiments, we set $\mathbf{\Xi}_i = 1$.

V. Discussion

Before we apply the proposed scheme of robust probabilistic estimation to two different domains and present results, we want to shortly discuss the implications that arise from it with respect to the size and hardness of the optimization problem.

First of all notice that it is not necessary to provide a switch variable for *every* constraint in the optimization problem. In fact, this should be avoided. Instead, a switch variable should only be associated to those constraints that can potentially be outliers. E.g. in the context of SLAM, each loop closure constraint should be associated with a switch variable, while odometry constraints might go safely without them.

a) Influence on the Problem Size: Obviously, the proposed robustified problem formulation significantly enlarges the original optimization problem. For each potential outlier constraint present in the original problem, another variable and an associated prior factor are added to the problem. However, given today's efficient solvers that exploit the sparseness of the optimization problem, the *size* of the problem (i.e. the number of variables and constraints) is not the most crucial factor that determines the runtime behaviour. By far more important is the sparse structure of the system's Jacobian.

b) Influence on the Sparseness of the Problem: The initial sparse structure of the optimization problem is not changed by the additional switch variables and their prior factors. This is clear because each of the switch variables governs only *one* constraint edge. The overall Jacobian of the proposed robust problem formulation remains sparse. The resulting Hessian

TABLE I: The datasets used during the evaluation.

Dataset	synthetic / real	2D/3D	Poses	Loop Closures
Manhattan (original)	synthetic	2D	3500	2099
Manhattan (g^2o version)	synthetic	2D	3500	2099
City10000	synthetic	2D	10000	10688
Sphere2500	synthetic	3D	2500	2450
Intel	real	2D	943	894

remains sparse as well which is expressed in the sparse interconnectivity of the problem's graph representation [10].

VI. Robust Estimation for Pose Graph SLAM

Following the scheme laid out above, we augment the original pose graph SLAM problem in order to make it robust against false positive loop closure constraints. Our proposed problem formulation for robust pose graph SLAM is:

$$
\begin{aligned}
X^*, S^* = \underset{X,S}{\operatorname{argmin}} & \underbrace{\sum_i \| f(\mathbf{x}_i, \mathbf{u}_i) - \mathbf{x}_{i+1} \|^2_{\mathbf{\Sigma}_i}}_{\text{Odometry Constraints}} \\
& + \underbrace{\sum_{ij} \| \Psi(s_{ij}) \cdot (f(\mathbf{x}_i, \mathbf{u}_{ij}) - \mathbf{x}_j) \|^2_{\mathbf{\Lambda}_{ij}}}_{\text{Switched Loop Closure Constraints}} \\
& + \underbrace{\sum_{i,j} \| \gamma_{ij} - s_{ij} \|^2_{\Xi_{ij}}}_{\text{Switch Prior Constraints}}
\end{aligned}
\tag{14}
$$

Fig. 3 illustrates the factor graph corresponding to this problem formulation. Notice that only the loop closure constraints have been augmented to *switched* constraints, while the odometry constraints remain unchanged.

In order to show the versatility and general feasibility of the proposed approach, a number of different datasets were used for the evaluation. Table I lists and summarizes their important properties. These datasets are pose graphs consisting of odometry measurements and loop closure constraints. They are free of outliers, i.e. all loop closure constraints are correct. To evaluate and benchmark the robust back-end, the datasets are spoiled by additional, wrong loop closure constraints. That means, loop closure constraints which do not connect corresponding poses and thus are outliers are added to the dataset, following different policies to simulate a realistic distribution of false positives. These four outlier policies added the outlier constraints either totally at random, in groups of 20, only locally, or in groups of local constraints. Given the spoiled datasets, the performance of the robust back-end was evaluated by comparing the resulting trajectory against the ground truth solution. The relative pose error RPE [7] was chosen as an error metric.

Fig. 4 compares the RPE of the proposed robust back-end to that of the non-robust state of the art formulation (solved by g2o [6]) for all datasets and different numbers of outliers. The robust back-end performs orders of magnitudes better, since the state of the art approach is not able to cope with outliers, despite being supported by the Huber cost function [4]. This

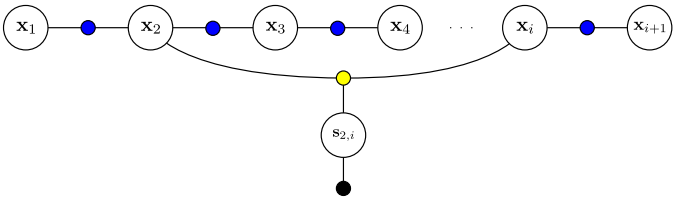


Fig. 3: Factor graph representation of the robustified pose graph SLAM formulation evaluated in this paper. Individual robot pose nodes are connected by odometry factors (blue). The switch variable $s_{2,i}$ governs the loop closure factor (yellow). Depending on the value assigned to the switch variable s_{ij}, the loop closure factor is gradually activated or deactivated as part of the optimization process. The switch variable is governed by a prior factor (black) that penalizes the deactivation of loop closures.

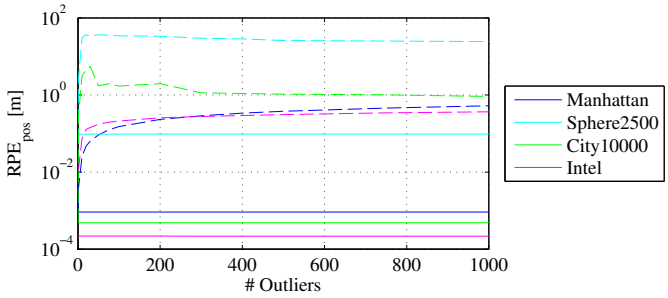


Fig. 4: Comparison of RPE measures between the proposed robust (solid line) and the state of the art non-robust back-ends (dashed). Notice how the robust solution is up to two orders of magnitude more accurate for large numbers of outliers and stays constant, independently of the amount of outliers. The non-robust solution was supported by the Huber cost function, as proposed in [2].

is also illustrated in Fig. 5 that compares the robust and non-robust solutions for two trials on the Manhattan world dataset.

Table II summarizes the results for the different datasets. The minimum, maximum, and median $\mathrm{RPE}_{\mathrm{pos}}$ measures are listed, as well as a success rate which measures the percentage of correct solutions.

We see that the overall success rates are very high. In total, from all 2500 trials, only two failed, leading to success rates equal or close to 100%. We want to remark that the two failure cases for the original Manhattan dataset and for the sphere world dataset could be successfully resolved by using the Huber cost function in combination with the proposed back-end. If we allow this further extension (which comes at the

TABLE II: Overall $\mathrm{RPE}_{\mathrm{pos}}$ metric for the different datasets, with 0 to 1000 outliers using all policies and 500 trials per dataset.

Dataset	max outl. ratio	min $\mathrm{RPE}_{\mathrm{pos}}$	max $\mathrm{RPE}_{\mathrm{pos}}$	median $\mathrm{RPE}_{\mathrm{pos}}$	success rate
Manhattan (g^2o)	47.6%	0.0009	0.0009	0.0009	100%
Manhattan (orig.)	47.6%	0.0009	5.9659	0.0009	99.8%
City10000	9.4%	0.0005	0.0005	0.0005	100%
Sphere2500	40.8%	0.0953	18.1674	0.0964	99.8%
Intel	111.9%	0.2122	0.2147	0.2132	100%

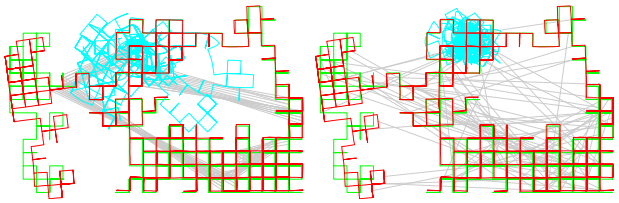
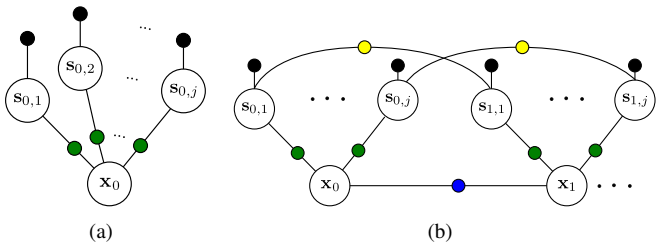


Fig. 5: Exemplary results on the Manhattan world dataset. 100 false positive loop closure constraints have been added in groups of 20 (left) or totally at random (right). In both cases, the proposed robust back-end converges to a correct solution (red), while current the state of the art back-ends like g^{2}o produce an unusable map (blue), despite being supported by the Huber robust cost function. Ground truth is shown in green for comparison, the deactivated false positive loop closures are visible in grey.



(a) (b)

Fig. 6: (a) A vehicle state vertex with three switched pseudorange factors $\mathbf{e}^{\text{spr}}$ (green), the associated switch variables and their prior factors $\mathbf{e}^{\text{sp}}$ (black). (b) Illustration of the most complex factor graph used in this paper: The switch variables are connected by switch transition factors $\mathbf{e}^{\text{swt}}$ (yellow) and the state transition factors $\mathbf{e}^{\text{st}}$ (blue) connect the state vertices.

cost of slower convergence, due to the partially linear cost function), we can conclude that a success rate of 100% was reached.

VII. Robust Estimation for GNSS-based Localization

Applying the same approach of robust estimation to the problem of GNSS-based localization, we gain the augmented *switched* pseudorange factor as

$$\|\mathbf{e}^{\text{spr}}_{tj}\|^2_{\mathbf{\Sigma}_{tj}} = \|\Psi(s_{tj}) \cdot (h(\mathbf{x}, j) - \rho_{tj})\|^2_{\mathbf{\Sigma}_{tj}} \tag{15}$$

and together with the switch prior constraint defined as in (14), this leads to the extended robust problem formulation:

$$X^*, S^* = \underset{X}{\operatorname{argmin}} \sum_{tj} \|\mathbf{e}^{\text{spr}}_{tj}\|^2_{\mathbf{\Sigma}_{tj}} + \|\mathbf{e}^{\text{sp}}_{tj}\|^2_{\mathbf{\Xi}_{tj}} \tag{16}$$

Fig. 6(a) illustrates this extended formulation for a single vehicle state variable. Notice how each pseudorange measurement is associated with its own switch variable.

In contrast to the switch variables in the pose graph SLAM problem, the switch variables in the GNSS-based localization problem are not independent: If a satellite j is observed from two successive vehicle locations $\mathbf{x}_{t-1}$ and $\mathbf{x}_t$, then s_{tj} is likely to be equal to $s_{t-1,j}$. We can capture this conditional dependence and model $P(s_{ij}|s_{t-1,j})$ as a Gaussian with

$$P(s_{tj}|s_{t-1,j}) \sim \mathcal{N}\left(s_{t-1,j}, \mathbf{\Sigma}^{\text{swt}}_{tj}\right) \tag{17}$$

which leads us to the *switch transition* factor

$$\|\mathbf{e}^{\text{swt}}_{tj}\|^2_{\mathbf{\Sigma}^{\text{swt}}_{tj}} = \|s_{tj} - s_{t-1,j}\|^2_{\mathbf{\Sigma}^{\text{swt}}_{tj}} \tag{18}$$

that can be easily incorporated as an additional factor into the overall optimization problem. Using the switch transition factors, we would solve

$$X^* = \underset{X}{\operatorname{argmin}} \sum_{tj} \|\mathbf{e}^{\text{spr}}_{tj}\|^2_{\mathbf{\Sigma}_{tj}} + \|\mathbf{e}^{\text{sp}}_{tj}\|^2_{\mathbf{\Xi}_{tj}} + \|\mathbf{e}^{\text{swt}}_{tj}\|^2_{\mathbf{\Sigma}^{\text{swt}}_{tj}} \tag{19}$$

for the maximum a posteriori estimate of X. More factors (e.g. a motion model) can be incorporated in the same convenient way, as illustrated in Fig. 6(b).

A. Results and Evaluation

The necessary data for the evaluation was collected in the city center of Chemnitz, Germany, using the Carai concept vehicle [9]. The vehicle was driven over a road junction several times, where severe multipath effects regularly occur due to nearby high buildings. Among other sensor systems, the Carai vehicle is equipped with a high-precision differential GPS and inertial measurement unit that allows to determine the vehicle's position with a precision of 2 cm. The position estimates of this high-precision unit were used as ground truth for the following analysis. In addition to the high-precision GPS unit, a consumer-class device (ublox LEA4) provided the pseudorange measurements that served as inputs for the optimization framework. To compare the estimation results with the ground truth provided by the high-precision GPS and IMU-devices from the Carai vehicle, the RMSE metric was used. Furthermore, we compare the solution of the robust optimization against a raytracing approach for multipath detection [8].

The best results are achieved when combining the three proposed factors for the switched pseudorange $\mathbf{e}^{\text{spr}}$, the switch transition $\mathbf{e}^{\text{swt}}$, and the vehicle state transition $\mathbf{e}^{\text{st}}$. Fig. 7 shows the RMSE for each vehicle pose for the proposed robust estimation, the conventional least squares method and the raytracing approach of [8]. The resulting trajectories for the robust and the conventional non-robust estimate can be compared in Fig. 8.

As we can see, our proposed robust optimization approach converges towards much better position estimates since it is able to detect and reject multipath measurements during the optimization process. It does not require an additional pre-processing step or additional knowledge or models of the environmental structure or the surrounding buildings. It outperforms the conventional non-robust least squares solution but also a sophisticated and computationally involved raytracing approach for multipath detection. Further details on the application of the proposed robust estimation scheme can be found in [10, 12].

VIII. Conclusions

We presented a generic scheme for robust, optimization-based probabilistic estimation and demonstrated its feasibility in two very different domains where the proposed method

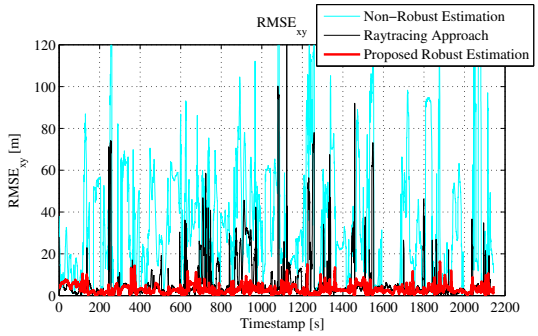


Fig. 7: RMSE metric compared for the proposed robust optimization (red), the conventional least squares (blue) and the raytracing method of [8] (black). Notice that the errors of the proposed robust optimization are constantly low while the other two approaches show significant spikes where the position estimation failed due to unhandled multipath effects.

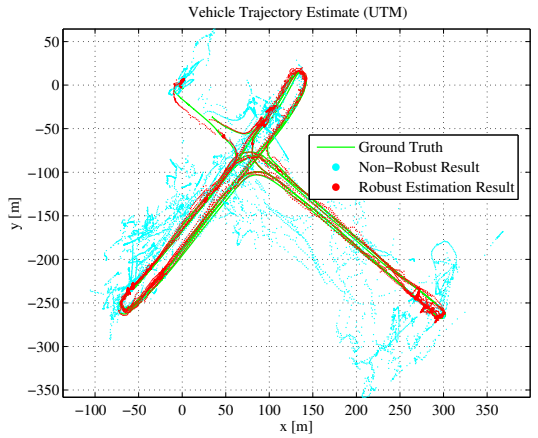


Fig. 8: Estimated vehicle trajectory in an urban scenario. While the conventional estimate (blue) based on the pseudorange readings from a consumer-class GPS receiver is extremely biased due to multipath errors, the robust estimation method proposed in this paper is able to detect and reject these outlier observations. The resulting trajectory estimate (red) is much closer to the ground truth (green).

outperformed non-robust state of the art approaches. Applying fault-tolerant estimation algorithms that can cope with the inevitable outliers and data association errors of the front-end is a key technique to reliable and safe long-term autonomy in real-world applications.

An implementation of the approach for g^{2}o along with the SLAM datasets used in the evaluation will be made available to the community at our website (http://www.tu-chemnitz.de/etit/proaut/forschung/robustSLAM.html.en).

References

[1] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981. ISSN 0001-0782.

[2] Giorgio Grisetti, Rainer Kümmerle, Hauke Strasdat, and Kurt Konolige. g^2o: A general Framework for (Hyper) Graph Optimization. Technical report, 2011.

[3] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[4] Peter J. Huber. Robust regression: Asymptotics, conjectures and monte carlo. *The Annals of Statistics*, 1(5): 799–821, 1973.

[5] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 47(2):498–519, February 2001. ISSN 0018-9448. doi: 10.1109/18.910572.

[6] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. g2o: A general framework for graph optimization. In *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2011.

[7] Rainer Kümmerle, Bastian Steder, Christian Dornhege, Michael Ruhnke, Giorgio Grisetti, Cyrill Stachniss, and Alexander Kleiner. On Measuring the Accuracy of SLAM Algorithms. *Auton. Robots*, 27:387–407, 2009. doi: 10.1007/s10514-009-9155-6.

[8] Marcus Obst, Sven Bauer, and Gerd Wanielik. Urban Multipath Detection and Mitigation with Dynamic 3DMaps for Reliable Land Vehicle Localization. In *IEEE/ION PLANS*, 2012.

[9] Robin Schubert, Eric Richter, Norman Mattern, Philipp Lindner, and Gerd Wanielik. A concept vehicle for rapid prototyping of advanced driver assistance systems. In *Advanced Microsystems for Automotive Applications 2010*, pages 211–219. Springer, 2010.

[10] Niko Sünderhauf. *Robust Optimization for Simultaneous Localization and Mapping*. PhD thesis, Chemnitz University of Technology, 2012. URL http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-86443.

[11] Niko Sünderhauf and Peter Protzel. Towards a Robust Back-End for Pose Graph SLAM. In *Proc. of IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2012.

[12] Niko Sünderhauf, Marcus Obst, Gerd Wanielik, and Peter Protzel. Multipath Mitigation in GNSS-Based Localization using Robust Optimization. In *Proc. of IEEE Intelligent Vehicles Symposium (IV)*, 2012.

[13] P. Torr and A. Zisserman. Mlesac: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156, 2000.

[14] René Wagner, Oliver Birbach, and Udo Frese. Rapid Development of Manifold-Based Graph Optimization Systems for Multi-Sensor Calibration and SLAM. In *Proc. of IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2011.