Proc. of IEEE Intl. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2017, Daegu, Korea. DOI: 10.1109/MFI.2017.8170347

Dynamic Covariance Estimation – A Parameter Free Approach to Robust Sensor Fusion

Tim Pfeifer, Sven Lange and Peter Protzel
Dept. of Electrical Engineering and Information Technology
TU Chemnitz, Germany
Email: {firstname.lastname}@etit.tu-chemnitz.de

***Abstract*— In robotics, non-linear least squares estimation is a common technique for simultaneous localization and mapping. One of the remaining challenges are measurement outliers leading to inconsistency or even divergence within the optimization process. Recently, several approaches for robust state estimation dealing with outliers inside the optimization back-end were presented, but all of them include at least one arbitrary tuning parameter that has to be set manually for each new application. Under changing environmental conditions, this can lead to poor convergence properties and erroneous estimates. To overcome this insufficiency, we propose a novel robust algorithm based on a parameter free probabilistic foundation called Dynamic Covariance Estimation. We derive our algorithm directly from the probabilistic formulation of a Gaussian maximum likelihood estimator. Through including its covariance in the optimization problem, we empower the optimizer to approximate these to the sensor's real properties. Finally, we prove the robustness of our approach on a real world wireless localization application where two similar state-of-the-art algorithms fail without extensive parameter tuning.**

I. Introduction

The probabilistic fusion of different kinds of sensor data is an extensively explored field of research. In robotics, the special case of simultaneous localization and mapping (SLAM) was solved by different techniques like Kalman or particle filter, but the de facto standard are optimization based algorithms. These algorithms reformulate the sensor fusion problem into a non-linear least squares optimization. While the least squares problem contains the probabilistic relations between sensor data and the states of a system, an optimization back-end is used to find the most likely system state. GTSAM [1] and Ceres [2] are just two examples for the most common optimization back-ends in recent robotic applications.

Standard least squares optimization is based on the idea of Gaussian distributed measurement noise with a known variance. However, many real world applications violate this essential assumption. Our motivation in this context are GNSS-based navigation systems like GPS as well as local wireless localization systems. They suffer form multipath and non-line-of-sight (NLOS) effects and cause heavy tailed or even multi-modal error distributions. Although they are bounded in range, their shape is non-Gaussian and changes with the properties of the environment like the height of buildings or the position of reflecting objects. Other examples are optical sensors which can be influenced by bad weather or challenging lighting conditions and wheel odometry which can be affected by slipping on difficult grounds. These errors can not be estimated online without additional sensor data and possibly distort the estimated system state significantly.

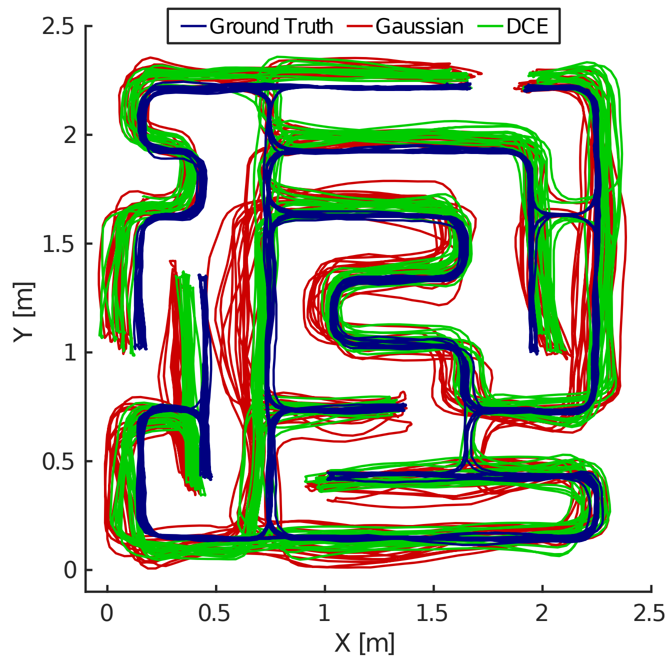

Fig. 1. The figure shows the estimated trajectory of our robot, equipped with a wireless localization system. Non-Gaussian outliers cause a distortion of the non-robust estimate. The proposed Dynamic Covariance Estimation (DCE) algorithm offers robustness against erroneous measurements without manual parameter tuning.

In the past, several algorithms were proposed to improve the back-end robustness for SLAM applications [3]–[8]. The majority of them can be divided in two groups, one tries to model a non-Gaussian error distribution while the other tries to weight down measurements that don't fit to a Gauss distribution.

In [9] we compared representatives from both groups. While the down weighting is advantageous for small outlier ratios, the probabilistic modeling approach seems to be a more general solution. The downside of all algorithms is the requirement of parametrization, since all of them introduce

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

at least one additional parameter compared to standard least squares. Corresponding to the concept of the algorithm, these parameters describe a probabilistic distribution or can be arbitrary. While a distribution's parameter can be estimated by statistical methods, the arbitrary parameters have to be tuned manually. Likewise, both can be extremely difficult if the error distribution vary over time.

In this paper, we want to introduce an approach that combines the robust down-weighting of Switchable Constraints (SC) with a probabilistic consistent design that is able to parametrize itself. Hence, we propose the novel Dynamic Covariance Estimation (DCE) algorithm that does not require any new parameter compared to standard least squares. We achieve this by including the variance of the sensor in the optimization process, as an additional state variable. Similar to Dynamic Covariance Scaling (DCS), we also provide an analytically optimized closed form alternative which can be expressed as an m-estimator. Finally, we validate both variants with a real world wireless localization system (see Figure 1) in comparison to SC and DCS.

II. Related Work

With the rising popularity of SLAM in the past decade, a broad variety of methods were proposed to achieve robustness against wrong data association and outliers. While SLAM can be viewed as special case of sensor data fusion, the majority of these techniques can be applied to general sensor fusion problems, as we showed in [9] or [10]. The common basis of these algorithms is (1) which maximizes the probability of the state variables $\boldsymbol{X}$ for a given set of measurements $\boldsymbol{Z}$.

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}}{\operatorname{argmax}} \prod_i \boldsymbol{P}(x_i|z_i) \tag{1}$$

The transformation of this maximum likelihood estimator for $\boldsymbol{X}^*$ to a non-linear least squares problem (2)[1] is possible under assuming a Gaussian distribution for $\boldsymbol{P}(x_i|z_i)$. For simplicity, we consider the one dimensional case, but the algorithm applies also to multidimensional problems.

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}}{\operatorname{argmin}} \sum_i \| \underbrace{e(x_i, z_i)}_{e_i} \|^2_\Sigma \tag{2}$$

Due to the vulnerability of least squares against outliers, the existing robust approaches modify this equation to limit or prevent the outliers influence. Max Mixture [6], [11] and generalized iSAM [7] wrap the error function e_i with a probabilistic model superior to a uni-modal Gaussian. Hence, it is possible to approximate an arbitrary error distribution for a wide variability of non-Gaussian measurement noise to obtain more realistic estimation results. While these class of algorithms can achieve a high level of robustness, for sensors with unknown or time-dependent distributions, they are difficult to parametrize.

The other class of algorithms is introducing an additional weight w_i to scale each error independently[2].

[1] $\| \cdot \|^2_\Sigma$ is the squared mahalanobis distance with covariance Σ.
[2] Another valid notation would be $\psi\left(\|e_i\|^2_\Sigma\right)$.

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}}{\operatorname{argmin}} \sum_i \|w_i \cdot e_i\|^2_\Sigma \tag{3}$$

These weights can be determined by the optimizer itself in case of Switchable Constraints (SC) [8], [12], [13] or calculated as function of e_i for Dynamic Covariance Scaling (DCS) [4], [14], [15] and other m-estimators.

A. Switchable Constraints

SC introduced a novelty for robust lest squares, a weight (or switch) s that is directly included in the optimization process. To prevent the trivial solution $s_i = 0 \; \forall \; i$ a prior constraint e_{SP} is added for each switch, which leads to (4). Additionally, a limitation of s_i between 0 and 1 is required.

$$\boldsymbol{X}^*, \boldsymbol{S}^* = \underset{\boldsymbol{X},\boldsymbol{S}}{\operatorname{argmin}} \left[\sum_i \|s_i \cdot e_i\|^2_\Sigma + \sum_i \| \underbrace{1 - s_i}_{e_i^{SP}} \|^2_{\Xi_i} \right] \tag{4}$$

A critical point of SC is the introduction of the new parameter Ξ as covariance of the switch prior e_i^{SP}. This tuning constant adjusts the trade-off between robustness against outliers and the tendency to weight valid measurements down. Due to the missing probabilistic relation between Ξ and the distribution of the sensor, this essential value is difficult to determine and has to be fine-tuned manually. Nevertheless, there seems to be a certain range of valid values for typical SLAM datasets [8], but no guarantee can be given that this also applies to general sensor fusion problems. In fact, in section VII, we show counterexamples.

B. Dynamic Covariance Scaling

Proposed as improvement of SC, the DCS algorithm applies an analytical optimization to transform (4) to an m-estimator similar to (3) while calculating each weight with (5).

$$s_i = \min\left(1, \frac{2\Phi}{\Phi + \|e_i\|^2_{\Sigma_i}}\right) \text{ with } \Phi = \Xi^{-1} \tag{5}$$

Through the closed form, an instantaneous calculation of s_i provides a faster convergence while achieving the same level of robustness compared to original SC. However, it still keeps the arbitrary tuning variable and all the disadvantages that go along with manual parameter tuning. In addition, our previous work [9] showed some potential convergence problems in sensor fusion applications.

C. Self-tuning M-Estimators

A parameter free solution for some cases is presented in the work of Agamennoni et al. [3], where a class of self tuning m-estimators is proposed. With the use of elliptical distributions, they are able to include the parameter ϕ of an m-estimator $\psi(\cdot, \phi)$ in the optimization problem. To keep the log-likelihood and therefore the least squares problem positive, they add a regularization term $\ln c\,(\psi)$.

$$\boldsymbol{X}^*, \phi^* = \underset{\boldsymbol{X},\phi}{\operatorname{argmin}} \left[n \cdot \ln c\,(\psi) + \frac{1}{2} \sum_{i=1}^{n} \psi\left(\|e_i\|^2_\Sigma\right) \right] \tag{6}$$

For many applications this approach proves to be valid. In some cases however, using the single parameter ϕ – instead of the set of weights $s_i \in \boldsymbol{S}$ of Switchable Constraints – may be insufficient. Furthermore, the adaptation of ϕ to a time-variable distribution is ad hoc not possible. Another limitation is the selection of m-estimators. DCS and other fast descending m-estimators have no corresponding elliptical distributions, so the self tuning can not be applied.

A common problem of many m-estimators in least squares problems is the bad convergence behaviour in case of inaccurate or incorrect initialization. Papers like [16], [17] address this issue but whether these techniques can be applied to DCS or the self-tuning m-estimators is unclear.

In the following section, we introduce a novel formulation of a robust least squares algorithm that is closely related to SC but parameter free and based on the Gauss distribution itself.

III. Dynamic Covariance Estimation

While Switchable Constraints adds a scaling factor to the error function, our approach scales the variance (or its root, the standard deviation) of the assumed error distribution directly. Since the standard deviation also scales the error, this appears identical at first glance, but it comes with different side effects on the estimation problem. To get a better understanding, we have to look at the missing steps between (1) and (2).

The transformation from a maximum likelihood estimator to a non-linear least squares problem is done by applying the natural logarithm which results in (7).

$$\boldsymbol{X}^* = \underset{\boldsymbol{X}}{\operatorname{argmin}} - \sum_i \ln\left(\boldsymbol{P}(x_i|z_i)\right) \tag{7}$$

$$\boldsymbol{P}(x_i|z_i) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left(-\frac{e_i^2}{2\sigma^2}\right) \tag{8}$$

With the probability density function of a zero-mean Gaussian (8), the log-likelihood for one measurement can be described as:

$$-\ln(\boldsymbol{P}(x_i|z_i)) = \underbrace{\ln\left(\sqrt{2\pi\sigma^2}\right)}_{\text{const.}} + \frac{1}{2}\|e_i\|^2_{\Sigma} \tag{9}$$

In standard least squares, the constant first part gets neglected. If we add the standard deviation to the estimation process on the other hand, this part becomes variable and has to be included. With this natural regularization term, a trivial solution like $\sigma = \infty$ can be prohibited.

To apply this to measurements with a non-constant error distribution, we replace the constant σ by a set of time dependent variances $\sigma_i \in \boldsymbol{\sigma}$. This leads to the new problem formulation in (10) and (11).

$$\boldsymbol{X}^*, \boldsymbol{\sigma}^* = \underset{\boldsymbol{X},\boldsymbol{\sigma}}{\operatorname{argmax}} \prod_i \boldsymbol{P}(x_i, \sigma_i|z_i) \tag{10}$$

$$-\ln(\boldsymbol{P}(x_i, \sigma_i|z_i)) = \underbrace{\ln\left(\sqrt{2\pi\sigma_i^2}\right)}_{\text{variable}} + \frac{1}{2}\|e_i\|^2_{\Sigma_i} \tag{11}$$

To keep this proposed equation valid for least squares optimization, we have to guarantee $-\ln(\boldsymbol{P}(x_i, \sigma_i|z_i)) \geq 0$, which is not possible for arbitrary σ_i. Therefore, we set a lower bound σ_{min} and shift (11) by a corresponding constant regularization term $-\ln\left(\sqrt{2\pi\sigma_{min}^2}\right)$, similar to Rosen et al. in [7] or the self tuning m-estimators mentioned in II-C. Through reformulation of (12) and step (13), we get our final Dynamic Covariance Estimation formulation (14). Equivalent to SC, DCE can be considered as two separated error functions where $\ln\|\sigma_i\|^2_{\Sigma_{min}}$ is a non-linear prior.

$$-\ln(\boldsymbol{P}) \propto \ln\left(\sqrt{2\pi\sigma_i^2}\right) - \ln\left(\sqrt{2\pi\sigma_{min}^2}\right) + \frac{1}{2}\|e_i\|^2_{\Sigma_i} \tag{12}$$

$$-\ln(\boldsymbol{P}) \propto \ln\left(\frac{\sigma_i}{\sigma_{min}}\right) + \frac{1}{2}\|e_i\|^2_{\Sigma_i} \tag{13}$$

$$-\ln(\boldsymbol{P}) \propto \frac{1}{2}\ln\|\sigma_i\|^2_{\Sigma_{min}} + \frac{1}{2}\|e_i\|^2_{\Sigma_i} \tag{14}$$

$\Sigma_{min} = \sigma_{min}^2$ is the pre-defined covariance of our physical sensor under normal conditions and can be determined experimentally or read from the datasheet. It is not a new free parameter since all other algorithms including the non-robust least squares also require this fundamental parameter. In consequence, we allow the optimizer to reduce the weight of erroneous measurements while preventing an overfitting of the exact ones. An advantage over m-estimator based approaches is the convex surface of the error term $\frac{1}{2}\|e_i\|^2_{\Sigma_i}$ which allows a well-behaved convergence. However, in common with SC, we have the computational burden of additional state variables.

IV. Closed Form DCE

Similar to the transformation between SC and DCS, through analytical optimization of DCE a closed form m-estimator can be provided. The optimization can be described with (15), where the error value e_i is treated as constant.

$$\sigma_i^* = \underset{\sigma_i}{\operatorname{argmax}} \frac{1}{2}\ln\|\sigma_i\|^2_{\Sigma_{min}} + \frac{1}{2}\|e_i\|^2_{\Sigma_i} \tag{15}$$

The maximum of this log-likelihood exists for $\sigma_i = \pm e_i$. Under the condition $\sigma_i \geq \sigma_{min}$, two cases have to be differentiated:

$$\sigma_i^* = \begin{cases} \sigma_{min} & \text{if } |e_i| \leq \sigma_{min} \\ |e_i| & \text{if } |e_i| > \sigma_{min} \end{cases} \tag{16}$$

Through substituting σ_i in (14) with (16) the resulting equation describes the closed form m-estimator of DCE (cDCE).

$$-\ln(\boldsymbol{P}(x_i|z_i)) \propto \begin{cases} \frac{1}{2}\|e_i\|^2_{\Sigma_i} & \text{if } |e_i| \leq \sigma_{min} \\ \frac{1}{2}\ln\|e_i\|^2_{\Sigma_{min}} + \frac{1}{2} & \text{if } |e_i| > \sigma_{min} \end{cases} \tag{17}$$

With σ_{min}, (17) requires the same probabilistic parameter as DCE. However, this parameter is quite simple to determine, as mentioned before.

While the resulting log-likelihood of DCE and cDCE is identical, there is no guarantee for an identical convergence behaviour. Since DCE (as well as SC) treats the variance and the resulting error as separate dimensions during the optimization, it can converge differently compared to cDCE. Therefore, we expect a better convergence of DCE similar to SC (compared to DCS) in our former work [9].

When applying cDCE it is important to remember the trade-off between robustness and convergence that all m-estimators share. Compared to DCS, we pushed this compromise closer to a better convergence but keep a decent level of robustness as Figure 2 shows.

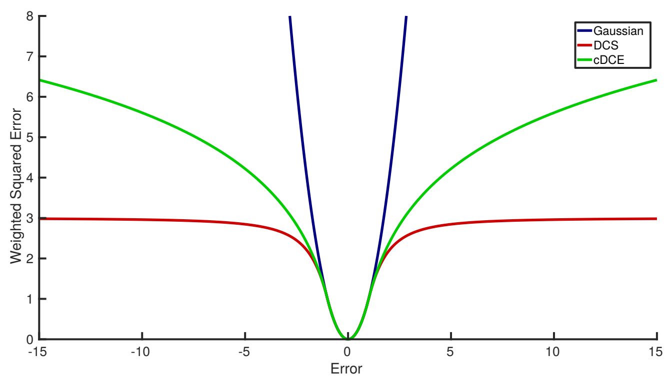

Fig. 2. Each error term adds a weighted and squared error to the optimization process. Robust weight-functions limit the influence of these terms. The cDCE m-estimator provide less robustness against extreme outliers, but better convergence properties than DCS.

V. Localization as Factor Graph

A basic class of sensor fusion problems is the position estimation based on multiple data sources, generally called localization. Our Evaluation is focused on a specific case, where relative wheel odometry is combined with an set of absolute range measurements to fixed points. To examine the structure of this estimation problem, we can describe it as a factor graph as shown in Figure 3. This graphical model shows the probabilistic connections (small dots) between the state variables (big circles). In the following section we explain these connections, also called error functions, that define our estimation problem.

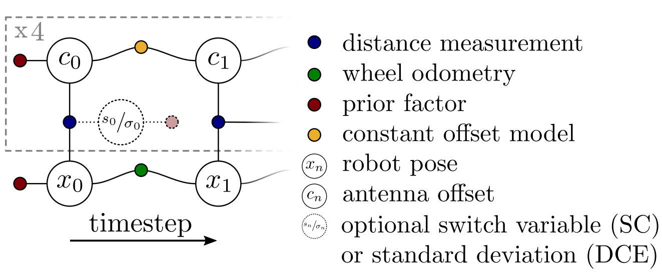

Fig. 3. Factor graph of the least squares problem. Big circles represent state variables while small circles are the probabilistic connections between these variables. The dotted switch or standard deviation variable is only added for SC or DCE. Note that the framed range and offset factors are present 4 times.

A. Range Error Model

Through Time-of-Flight measurement, the UWB modules provide an absolute distance between itself $x^{x,y}_{Mod}$ and the position of our robot $x^{x,y}_i$. In addition to a Gaussian noise and the NLOS errors, each range measurement z^{rng}_i contains a certain offset c^{Mod}_i caused by the physical properties of the antenna. This antenna offset depends on the orientation and distance between transmitter and receiver and is estimated for each module independently. With (18) the corresponding error function is provided.

$$e^{rng}_i = \left\| \sqrt{\|x^{x,y}_{Mod} - x^{x,y}_i\|^2} + c^{Mod}_i - z^{rng}_i \right\|^2_{\Sigma_{rng}} \tag{18}$$

B. Odometry Model

The state transition between poses as well as the initialization of new ones is based on a motion model of a differential drive robot. By using both wheels' velocity measurement as input, the error function is given as e^{odo}_i. Since a two-wheeled differential drive robot is non-holonomic, the velocity perpendicular to the driving direction also has to be considered to formulate a well-posed estimation problem. We assume these to have a zero mean, which results in a measurement vector $z^{odo}_i = [\, v_l,\, v_r,\, v_y{=}0\,]^\intercal$ that contains one additional entry along with the wheel velocities.

$$e^{odo}_i = \left\| \mathbf{T} \cdot (x^{x,y,\phi}_i - x^{x,y,\phi}_{i-1}) \Delta t^{-1} - z^{odo}_i \right\|^2_{\Sigma_{odo}} \tag{19}$$

The formulation of (19) in the measurement space, requires a transformation from two consecutive global poses to a set of differential drive velocities with the matrix T. Therefore, (20) contains a rotation from a global to a local coordinate frame combined with the inverse kinematic of a differential drive robot. x^{ϕ}_i denotes the rotational component of the 2D pose and $2 \cdot d_w$ the distance between both wheels.

$$\mathbf{T} = \underbrace{\begin{bmatrix} 1 & 0 & -d_w \\ 1 & 0 & d_w \\ 0 & 1 & 0 \end{bmatrix}}_{\text{differential kinematic}} \cdot \underbrace{\begin{bmatrix} \cos x^{\phi}_i & \sin x^{\phi}_i & 0 \\ -\sin x^{\phi}_i & \cos x^{\phi}_i & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotation}} \tag{20}$$

C. Constant Offset Model

The module specific offset e^{offset}_i is caused by the influence of the antenna on the electromagnetic wave propagation. Not only the physical characteristics but also the alignment of the antennas affect this value. Therefore, a continuous distance and orientation variation caused by the robot's movement, requires also a continuous offset estimation. To ensure the consistency of c^{Mod}_i over time we use a simple constant value model (21).

$$e^{offset}_i = \left\| \left(c^{Mod}_i - c^{Mod}_{i-1}\right) \cdot \Delta t^{-1} \right\|^2_{\Sigma_{offset}} \tag{21}$$

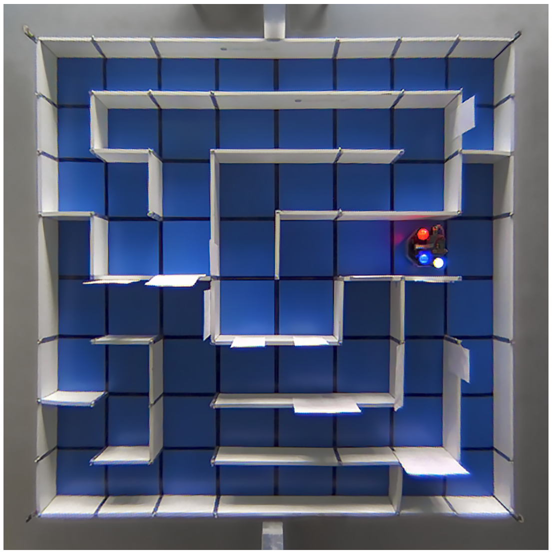

Fig. 4. Our robotic system inside the labyrinth, seen from the tracking system. One UWB module is placed in each corner and one on top of the robot. The white paper sheets contain two layer of aluminium foil to enforce NLOS measurements.

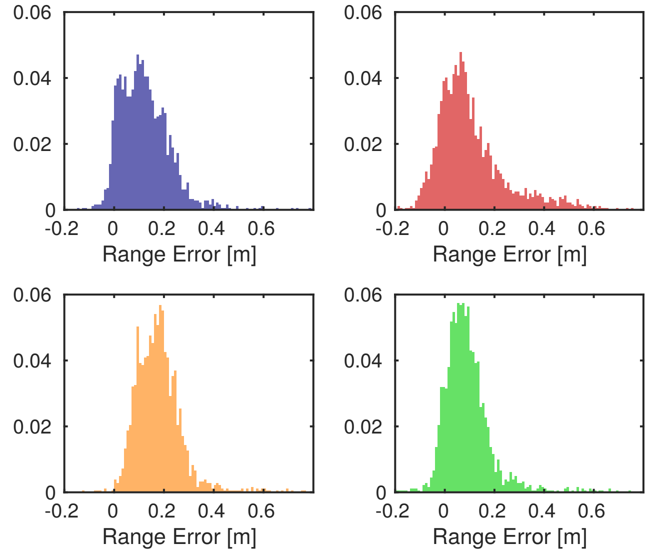

Fig. 5. These histograms show the error distributions of the wireless distance measurements to the 4 fixed modules. Due to the geometric differences in the labyrinth they can differ. All distributions are asymmetric and contain a different amount of outliers.

VI. Experiment

To explore the different robustness and convergence properties, we decided to evaluate our proposed DCE/cDCE algorithm along with SC and DCS on a real world dataset. Based on our experience with a synthetic benchmark in [9], we designed an equivalent setup with one of our small educational robots [18] that navigates through a labyrinth. Due to the flat surface, we reduced this to a two-dimensional localization problem. As ground truth for evaluation, an optical mono-camera tracking system as introduced in [19] is used. We adapted this system to the 2D case and achieve a centimetre-level accuracy, which is sufficient for our comparison. Our final dataset contains about 15 minutes of continuous driving.

A. Wireless Localization Dataset

For long term consistent localization, the wheel odometry is complemented by a set of five wireless ranging sensors. One on top of the robot and one in each corner of the labyrinth as seen in Figure 4. Once each time step, the robot measures the distance to one of the static modules. These ultra-wideband sensors [20] are ideal to compare robust algorithms, since they suffer from non-Gaussian error distributions. While they are inherent robust against multipath effects, they cannot measure a correct distance if the direct line-of-sight is blocked by an obstacle. In these NLOS cases, the measured distance is significantly longer than the real one. The resulting heavy-tailed distribution is challenging for non-robust optimization. So the corresponding error function is robustified by our set of algorithms. Obstacles could be walls, furniture or humans for indoor applications or entire buildings in satellite based localization applications. We enforce a decent amount of NLOS measurements with artificial walls, made of aluminium foil.

B. Parametrization

All factors included in Figure 3 require a covariance for each dimension of the particular error function. These are summarized in Table I. A correct parametrization of these covariances is mandatory for a meaningful comparison. Hence, we estimated these values very carefully in separate experiments. By using the visual ground truth, we estimated the odometry's standard deviation over several minutes of driving. The standard deviation of our wireless distance measurements was determined outdoors in a flat environment. Under this free space conditions, we were able to measure the sensors noise. A separate test-drive under LOS conditions the completed the characterization of the UWB sensors with the constant offset model's covariance. The covariance of the used priors is the only one that we set manually. It represents the knowledge about the robot's start state which is limited since we only know that the robot is somewhere in the labyrinth.

TABLE I

Covariances of the Estimation Problem

model	error function	covariance
distance measurement	e_i^{rng}	$\Sigma_{rng} = [0.025\,\mathrm{m}]^2$
wheel odometry	e_i^{odo}	$\Sigma_{odo} = \mathrm{diag}\begin{bmatrix}0.03\,\mathrm{m\,s^{-1}}\\0.03\,\mathrm{m\,s^{-1}}\\0.03\,\mathrm{m\,s^{-1}}\end{bmatrix}^2$
constant offset model	e_i^{offset}	$\Sigma_{offset} = \left[1\,\mathrm{mm\,s^{-1}}\right]^2$
antenna offset prior	e_{offset}^{prior}	$\Sigma_{offset}^{prior} = [0.1\,\mathrm{m}]^2$
pose prior	e_{prior}^{pose}	$\Sigma_{prior}^{pose} = \mathrm{diag}\begin{bmatrix}10\,\mathrm{m}\\10\,\mathrm{m}\\2\pi\,\mathrm{rad}\end{bmatrix}^2$

We implemented our least squares problem with the Ceres solver [2] as optimization back-end. While the growing estimation problem would violate any real time condition, we apply a sliding window approach to exclude old state variables from optimization. The length of the time window balances the trade-off between computation time and estimation quality. We choose a length of 10 s to keep the required optimization time bounded without losing noteworthy precision.

VII. Results

A. Parameter Tuning of SC and DCS

The performance of SC and DCS depends strongly on the chosen tuning parameter. To show this dependency, we performed several runs of both algorithms with a different parametrization. Figure 6 plots the resulting absolute trajectory error (ATE) [21] for parameter values between $\xi = 0.001$ and 1.6.

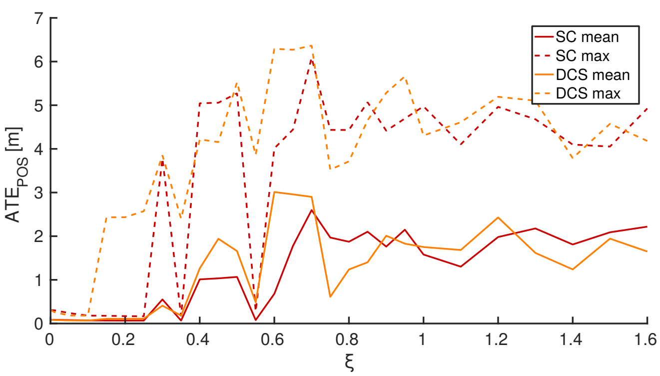

Fig. 6. Our parameter evaluation of SC and DCS shows the absolute trajectory error for different parameter values ξ. SC applies $\Xi = \xi$ while DCS uses the inverse $\Phi = \xi^{-1}$. Only a small range between 0.1 and 0.25 offers an optimal performance.

According to our results, DCS and SC converge only for values of ξ below 0.25. Values smaller than 0.1 don't break the optimization process but lead to a performance identical to the non-robust optimization. In the final comparison, we used the best parametrization for both algorithms. Hence, SC uses a Ξ of 0.2 and DCS a Φ of 10. Compared to the results of DCS in [4] and [14], we got a significantly smaller range of valid values. In case of SC there is an even more distinct difference to [22], where a range of 0.3 to 1.5 is recommended. These tests were performed on SLAM datasets with synthetic error distributions, unclear covariances and different optimization back-ends, which could explain some of the discrepancies. Furthermore, our sliding window approach can be more challenging than the batch optimization in previous publications.

B. Final Evaluation

Since the performance of SC and DCS depends on the tuning parameters, we included two parameter sets to our comparison. We run both with their default parameter of 1.0 and with the prior tuned parameters, which lead to different results. Our proposed DCE/cDCE algorithms achieved the same results as the manually tuned SC and DCS versions. All of these algorithms are able to suppress the influence of outliers to the resulting estimate. Especially the maximum trajectory error is reduced by almost 50% as shown in Figure 7. However, as summarized in Table II, SC as well as DCS fail to converge on this problem with their tuning parameter set to 1. Therefore, both algorithms cannot improve the estimation result over the bare odometry initialization with the recommended parametrization.

TABLE II

Results of the final run

Algorithm	ATE[m] mean	ATE[m] max	Time [s]
Odometry Initial	1.7015	4.3317	-
Gaussian	0.0858	0.3060	34.2
DCE	0.0659	0.1671	96.5
cDCE	0.0647	0.1801	52.8
SC ($\Xi = 1.0$)	1.5795	4.9743	48.6
SC ($\Xi = 0.2$)	0.0667	0.1693	54.4
DCS ($\Phi = 1.0$)	1.7515	4.3051	34.5
DCS ($\Phi = 10$)	0.0687	0.1806	41.3

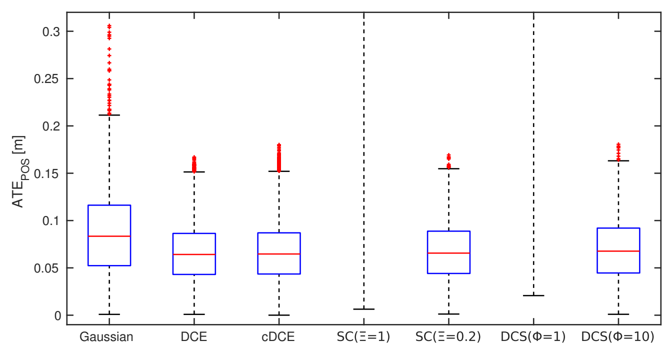

Fig. 7. Absolute Trajectory error of the estimated positions. DCE/cDCE as well as tuned SC/DCS performed well, but standard SC/DCS failed to converge.

The accumulated runtime of the sliding window filter given in Table II was measured on an Intel i7-4770 system. To mention is the advantage of the closed form algorithms (DCS and cDCE), both require less computational time then their respectively alternatives with additional state variables. Also, DCE and cDCE require more time to converge to a final solution but compared to the dataset length of 930 seconds, all algorithms are sufficiently fast.

VIII. Conclusion

We introduced a novel robust estimation algorithm which was designed with a probabilistic foundation and without the introduction of an arbitrary tuning constant. The basic idea is to include the covariance of the sensor itself to the optimization process. Furthermore, we derived a closed form alternative to our Dynamic Covariance Estimation algorithm, providing comparable robustness with reduced computational cost. We compared both variants to the similar Switchable Constraints and Dynamic Covariance Scaling algorithm on a real world wireless localization dataset. So we were able to show the advantage of our parameter free algorithms since SC and DCS perform only equivalent with extensive parameter tuning in advance.

In our future work, we will extend this comparison to different datasets to generalize our observations from this specific setting. Furthermore, an evaluation of the theoretical properties of the DCE algorithm will be useful to expose the relationship between the estimated and the real covariance. An analysis of possible probabilistic connections between the individual covariance variables could also be very interesting in this context.

REFERENCES

[1] F. Dellaert and Others, "GTSAM," http://research.cc.gatech.edu/borg/gtsam.

[2] S. Agarwal, K. Mierle, and Others, "Ceres Solver," http://ceres-solver.org.

[3] G. Agamennoni, P. Furgale, and R. Siegwart, "Self-tuning m-estimators," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[4] 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.

[5] Y. Latif, C. Cadena, and J. Neira, "Robust loop closing over time for pose graph SLAM," *Intl. Journal of Robotics Research*, 2013.

[6] E. Olson and P. Agarwal, "Inference on networks of mixtures for robust robot mapping," *Intl. Journal of Robotics Research*, 2013.

[7] 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.

[8] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose graph slam," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2012.

[9] T. Pfeifer, P. Weissig, S. Lange, and P. Protzel, "Robust factor graph optimization – a comparison for sensor fusion applications," in *Proc. of Intl. Conf. on Emerging Technologies and Factory Automation (ETFA)*, 2016.

[10] S. Lange, N. Sünderhauf, and P. Protzel, "Incremental smoothing vs. filtering for sensor fusion on an indoor UAV," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2013.

[11] R. Morton and E. Olson, "Robust sensor characterization via max-mixture models: GPS sensors," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[12] N. Sünderhauf, M. Obst, S. Lange, G. Wanielik, and P. Protzel, "Switchable constraints and incremental smoothing for online mitigation of non-line-of-sight and multipath effects," in *Proc. of Intelligent Vehicles Symposium (IV)*, 2013.

[13] N. Sünderhauf and P. Protzel, "Switchable constraints vs. max-mixture models vs. rrr – a comparison of three approaches to robust pose graph SLAM," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2013.

[14] P. Agarwal, G. Grisetti, G. Diego Tipaldi, L. Spinello, W. Burgard, and C. Stachniss, "Experimental analysis of dynamic covariance scaling for robust map optimization under bad initial estimates," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2014.

[15] P. Agarwal, "Robust graph-based localization and mapping," Ph.D. dissertation, University of Freiburg, 2015.

[16] D. M. Rosen, C. DuHadway, and J. J. Leonard, "A convex relaxation for approximate global optimization in simultaneous localization and mapping," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2015.

[17] J. Wang and E. Olson, "Robust pose graph optimization using stochastic gradient descent," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2014.

[18] S. Lange, P. Weissig, A. Uhlig, and P. Protzel, "Tuc-bot: A microcontroller based robot for education," in *Proc. of Intl. Conf. on Robotics in Education (RiE)*, 2017.

[19] S. Lange and P. Protzel, "Cost-Efficient Mono-Camera Tracking System for a Multirotor UAV Aimed for Hardware-in-the-Loop Experiments," in *Proc. of Intl. Multi-Conference on Systems, Signals and Devices (SSD)*, 2012.

[20] DecaWave, *ScenSor DWM1000 Module*. [Online]. Available: https://www.decawave.com/sites/default/files/resources/DWM1000-Datasheet-V1.6.pdf

[21] A. Bonarini, W. Burgard, G. Fontana, M. Matteucci, D. G. Sorrenti, and J. D. Tardos, "RAWSEEDS: Robotics Advancement through Web-publishing of Sensorial and Elaborated Extensive Data Sets," in *Proc. of Intl. Conf. on Intelligent Robots and Systems (IROS) Workshop on Benchmarks in Robotics Research*, 2006.

[22] N. Sünderhauf, "Robust optimization for simultaneous localization and mapping," Ph.D. dissertation, TU Chemnitz, 2012.