Efficient Time-of-Arrival Self-Calibration using Source Implicitization

Malte Larsson
Combain Mobile AB
Centre for Mathematical Sciences
Lund University, Sweden
malte.larsson@math.lth.se

Viktor Larsson
Centre for Mathematical Sciences
Lund University, Sweden
viktor.larsson@math.lth.se

Magnus Oskarsson
Centre for Mathematical Sciences
Lund University, Sweden
magnus.oskarsson@math.lth.se

***Abstract*—In this paper we revisit the Time-of-Arrival self-calibration problem. In particular we focus on imbalanced problem instances where there are significantly more sources compared to the number of receivers, which is a common configuration in real applications. Using an implicit representation, we are able to re-parameterize the sensor node self-calibration problem using only the parameters of the receiver positions. Making the source positions implicit, we show that it is possible to linearize the maximum-likelihood error around the measured distances, resulting in a Sampson-like approximation. Given four unknown receiver positions and a large number of unknown sender positions, we show that our formulation leads to algorithms for robust calibration, with significant speed-up compared to running the full optimization over all unknowns. The proposed method is tested on both synthetic and real data.**

***Index Terms*—Time-of-Arrival, Sensor node calibration, robust optimization**

I. Introduction

For many signal-based mapping and positioning applications we need to perform sensor array calibration as having accurate receiver-sender node positions are typically a key prerequisite [1]. When senders and receivers are synchronized, it is possible to obtain absolute distance measurements between senders and receivers by measuring the signal propagation time. These measurements can be used for self-calibration and such problems (Time-of-Arrival problems, TOA) have been studied in a large body of work [2]–[10]. In cases where either the receivers or sources are unsynchronzed, the problem is instead called Time-Difference-of-Arrival (TDOA), [11]–[14].

In this paper we focus on the TOA-setting, and consider the special case where we only have four receivers, which is the smallest number that allows for self-calibration, i.e. jointly estimating receiver and source positions.

In many applications these problems are heavily imbalanced—in the sense that we often have a much larger number of sources compared to receivers. For example, a moving continuous sound source might generate hundreds or thousands of source locations in a very short time frame.

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP) funded by Knut and Alice Wallenberg Foundation, the ADACORSA project with funding from ECSEL JU in the H2020 Framework Programme (H2020/2014-2020) and National Authorities, under GA 876019, and the strategic research project ELLIIT.

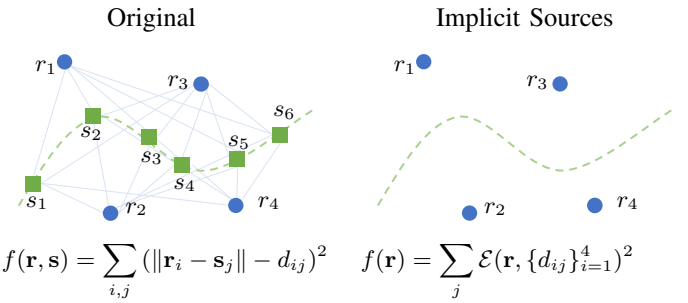

Fig. 1. Time-of-Arrival self-calibration aims to estimate both receiver $\mathbf{r}_i$ and source positions $\mathbf{s}_j$ from pairwise distance measurements d_{ij}. In many cases there are significantly more sources compared to receivers. We propose a source implicitization technique, that allows for receiver self-calibration without explicitly parameterizing the source. This leads to a simplified error function, that only depends on the four receiver positions $\mathbf{r}_i$.

The main idea in this paper is to eliminate the source positions from the estimation problem using a novel implicitization method. This allows us to achieve the same result, while only optimizing over the receiver positions without explicitly parameterizing the sources. An overview of the approach is shown in Figure 1.

II. Problem formulation

The problem we address involves $m = 4$ receiver positions $\mathbf{r}_i \in \mathbb{R}^3$, $i = 1, \ldots, 4$ and n source positions $\mathbf{s}_j \in \mathbb{R}^3$, $j = 1, \ldots, n$. These could for example represent the microphone positions and locations of sound emissions, respectively. We denote the arrival time of a signal sent from sender j to receiver i by t_{ij}, and the time of emission with τ_j. If we multiply the travel time $t_{ij} - \tau_j$ with the speed v of the signal, we obtain the distance between the sender and receiver

$$d_{ij} = v(t_{ij} - \tau_j) = \|\mathbf{r}_i - \mathbf{s}_j\|, \tag{1}$$

where $\|.\|$ denotes the ℓ^2-norm. The transmission speed v and time offsets τ_j are assumed to be known and constant. Let d_{ij} be the measured distances, which we assume to be perturbed by Gaussian noise and might additionally contain outliers with substantially larger errors. Recovering the receiver positions $\mathbf{r}_i$ and source positions $\mathbf{s}_j$ from measured pairwise distances d_{ij} is then known as the Time-of-Arrival sensor node self-calibration problem.

The self-calibration problem can be formulated as a robust optimization problem over the receiver and source positions,

$$\min_{\mathbf{r},\mathbf{s}} \sum_{ij} \min((\|\mathbf{r}_i - \mathbf{s}_j\| - d_{ij})^2, \sigma^2), \tag{2}$$

where we minimize the squared residuals to the measured distances, truncated at some fixed threshold $\sigma \in \mathbb{R}_+$ to prevent outlier measurements to impact and corrupt the solutions. This optimization problem concerns $3 \cdot 4 + 3n - 6$ variables.[1] In applications the number of sources $n >> 6$ often dominates the number of receivers, making the optimization of (2) computationally expensive.

In the next sections we will introduce an implicitization scheme, where we derive a surrogate problem that approximates (2) while allowing us to only optimize over the receiver positions (six degrees of freedom).

III. The Quadrisonal Tensor

Assume that we have four receivers, in any given configuration, and one single unknown source. There will then exist a condition, that the receiver positions and the four distances to the source must fulfill. This internal constraint is called the *quadrisonal tensor*, and it was first introduced by Stewenius [6]. Mathematically, we can formulate the setup as

$$\|\mathbf{r}_i - \mathbf{s}\|^2 = d_i^2, \quad i = 1, 2, 3, 4, \tag{3}$$

where $\mathbf{r}_i$ is the receiver with distance d_i to the source $\mathbf{s}$. To fix the gauge freedom in the coordinate system we can without loss of generality choose the receiver positions as

$$[\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3 \ \mathbf{r}_4] = \begin{bmatrix} 0 & x_1 & x_2 & x_4 \\ 0 & 0 & x_3 & x_5 \\ 0 & 0 & 0 & x_6 \end{bmatrix}. \tag{4}$$

The quadrisonal constraint is then obtained by eliminating the three coordinates of the source $\mathbf{s}$ from the four equations in (3). Expanding and introducing an extra unknown α we get

$$\|\mathbf{r}_i\|^2 - 2\mathbf{r}_i^T\mathbf{s} + \alpha = d_i^2, \quad i = 1, 2, 3, 4, \tag{5}$$

$$\alpha - \mathbf{s}^T\mathbf{s} = 0. \tag{6}$$

Equation (5) is linear in $\mathbf{s}$ and α and the solution is given by

$$\begin{pmatrix} \mathbf{s} \\ \alpha \end{pmatrix} = \begin{bmatrix} -2\mathbf{r}_1^T & 1 \\ \vdots & \vdots \\ -2\mathbf{r}_4^T & 1 \end{bmatrix}^{-1} \begin{pmatrix} d_1^2 - \|\mathbf{r}_1\|^2 \\ \vdots \\ d_4^2 - \|\mathbf{r}_4\|^2 \end{pmatrix}. \tag{7}$$

Taking the solution and inserting into (6) yields a rational constraint (due to the inverse in (7), and the square of $\mathbf{s}$ in (6)) in terms of $\{\mathbf{r}_i\}_{i=1}^4$ and $\{d_i\}_{i=1}^4$. Multiplying with the denominator yields a polynomial of degree eight in $\mathbf{r}_i$ and four in d_i. This is the called the *quadrisonal constraint*. In [6] it is shown that this can be written as a scalar product

$$\langle T(\mathbf{r}),\ S(\mathbf{d})\rangle = 0 \tag{8}$$

[1]There are six degrees of gauge freedom in the choice of coordinate system in 3D.

where $T(\mathbf{r}) \in \mathbb{R}^{11}$ is a vector of polynomials in $\mathbf{r}$, and $S(\mathbf{d}) \in \mathbb{R}^{11}$ is a vector with polynomials in the distances $\mathbf{d}$. The vector $T(\mathbf{r})$ is the *quadrisonal tensor* and only depends on the configuration of the four receivers. Given any 4-tuple of distances $\mathbf{d} = (d_1, d_2, d_3, d_4)$, the quadrisonal tensor allows us to easily check whether they are consistent with the four receiver positions, i.e. if it is possible to exactly trilaterate a source $\mathbf{s}$ that satisfies (3) with the given distances.

A. Source Implicitization

Our goal now is to find a surrogate optimization problem for (2) that only depends on the measurements d_{ij} and the receiver positions. One naïve approach is to directly use the quadrisonal constraints from the previous section and minimize the squared equation residuals from (8). This residual is however purely algebraic and not geometrically meaningful, which might lead to biased results. Furthermore, it is unclear how to set the inlier threshold σ in (2) for this type of residual.

First let us consider the maximum likelihood residual, i.e.

$$\mathcal{E}_{ML}(\mathbf{r}, \mathbf{d}) = \min_{\mathbf{s}\in\mathbb{R}^3} \sum_{i=1}^{4} (\|\mathbf{r}_i - \mathbf{s}\| - d_i)^2. \tag{9}$$

This residual is expensive to evaluate (as it requires solving the optimal trilateration problem). Using the quadrisonal tensor, we can however reformulate the residual in (9) as

$$\mathcal{E}_{ML}(\mathbf{r}, \mathbf{d}) = \min_{\hat{\mathbf{d}}\in\mathbb{R}^4} \|\hat{\mathbf{d}} - \mathbf{d}\|^2 \quad \text{s.t.} \quad \langle T(\mathbf{r}), S(\hat{\mathbf{d}})\rangle = 0, \tag{10}$$

i.e. finding the closest set of distances $\hat{\mathbf{d}}$ that satisfies the quadrisonal constraint. This is equivalent since in this case we know that there exists an $\mathbf{s} \in \mathbb{R}^3$ such that $\|\mathbf{r}_i - \mathbf{s}\| = \hat{d}_i$, which will then be the optimal $\mathbf{s}$ in (9). The function in (10) is however still expensive to evaluate, now requiring to solve a complicated constrained optimization problem. To arrive at a tractable alternative, we now draw inspiration from the Sampson approximation [15] which is used to approximate the distance from points to a given conic. The main idea is to linearize the complicated constraint in (10) at the noisy measured distances $\mathbf{d}$ and solve the linearized problem, i.e.

$$\mathcal{E}_{QS}(\mathbf{r}, \mathbf{d}) = \min_{\hat{\mathbf{d}}\in\mathbb{R}^4} \|\hat{\mathbf{d}} - \mathbf{d}\|^2, \tag{11}$$

$$\text{s.t.} \quad \langle T(\mathbf{r}), S(\mathbf{d}) + J_S\left(\hat{\mathbf{d}} - \mathbf{d}\right)\rangle = 0, \tag{12}$$

where the Jacobian $J_S \in \mathbb{R}^{11\times 4}$ is evaluated at $\mathbf{d}$. The problem is now a quadratic optimization problem with a single linear constraint, having the closed form solution (see e.g. [15])

$$\mathcal{E}_{QS}(\mathbf{r}, \mathbf{d}) = \frac{\langle T(\mathbf{r}), S(\mathbf{d})\rangle^2}{\|J_S(\mathbf{d})\|^2} \tag{13}$$

This provides a first-order approximation of (9) which is cheap to compute (only requiring a few polynomial evaluations), and that only depends on the receiver positions and distances.

IV. Receiver Calibration with Implicit Sources

Using the approximation developed in the previous section we can now restate the self-calibration in (2) as a minimization over only the receiver positions as

$$\min_{\mathbf{r}} \sum_j \min(\mathcal{E}_{QS}(\mathbf{r}, \mathbf{d}_j), \sigma^2) \tag{14}$$

where $\mathbf{r} = (\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{r}_4)$ and $\mathbf{d}_j = (d_{1j}, d_{2j}, d_{3j}, d_{4j})$. While it is now a minimization problem with only six degrees of freedom, it is still highly non-convex, in particular due the robust truncation of the residual at σ.

To minimize this cost we propose to use a random sampling based intialization as in RANSAC [16], and in particlar we propose to use the LO-RANSAC [17], [18] variant which performs additional local optimization in the sampling loop. The idea is that since we don't know which of the measurements are outliers, we randomly select small subsets (in our case the minimum number is six) of residuals and fit a model to these. If these were all inliers, the model will be roughly correct and can serve as a starting point for local optimization. To ensure robustness, multiple minimal samples are drawn from which we generate multiple candidates, before performing local optimization starting from the best one. To estimate the receivers from six measurements we use the method initially introduced in [7] and later optimized in [9].

Since any configuration of four receivers can be rigidly transformed into the form (4) we only need to optimize over the six parameters $(x_1, x_2, x_3, x_4, x_5, x_6) \in \mathbb{R}^6$. Once the receiver positions are estimated, the senders can then be trilaterated, e.g. using the method from [19].

Implementation Details: The optimization problems are solved using Ceres [20] in C++. We use the default Levenberg-Marquardt optimizer, with automatic differentiation to compute derivatives of the objective function. All other code is written in MATLAB and is run on a Zenbook UM3402YA laptop with a AMD Ryzen 7 5825U.[2]

V. Experimental Evaluation

In the experiments we will compare the classical ML-estimate $\mathcal{E}_{ML}$ (9), optimizing jointly over both $\mathbf{r}$ and $\mathbf{s}$, with our proposed approximation $\mathcal{E}_{QS}$ (13), which only depends on the receiver positions. We compare them both in the context of model-scoring in RANSAC, as well as for performing local optimization of both receivers and senders. In the following sections we first evaluate the approximation error, followed by a comparison in local refinement and finally in RANSAC for solving the full TOA problem. Experiments have been done on both real and synthetic data. Descriptions of the data sets are included in the following two sections.

A. Evaluation of Quadrisonal Sampson Approximation

We first experimentally evaluate the residual $\mathcal{E}_{QS}$ given by the Quadrisonal Sampson approximation (13) compared to the maximium likelihood residual $\mathcal{E}_{ML}$ (9). In Figure 3 we can see how the noise level in the data affects the approximation quality. This is done for simulated data where sender and receiver coordinates are chosen uniformly from [-2m, 2m], and to each distance a zero mean Gaussian noise with standard deviation σ_A is added. We plot the mean of the error for 100 random such data sets with four receivers and 100 senders each. For noise levels in the distance data up to $\sigma_A \approx 0.2m$ we get that the quadrisonal sampson approximation lies very close to that which you would get by trilateration. For larger errors in the distance data, the proposed approximation underestimates the true error residual.

The approximation takes around 5 μs to compute per sender as compared to around 50 μs for the trilateration, i.e., it is around 10 times faster to evaluate.

[2]Code is available at https://github.com/maltelarsson2/source-implicitization-self-calibration.

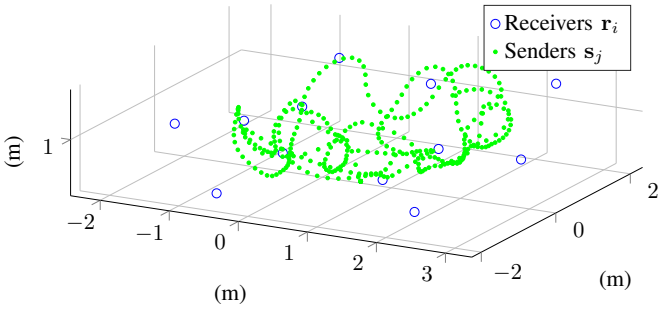

Fig. 2. The ground-truth receiver and sender positions for the real dataset used in the experiments.

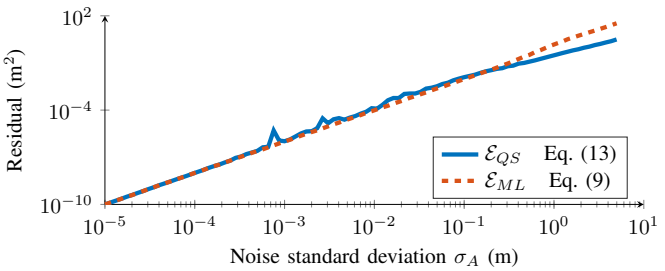

Fig. 3. The figure shows the approximation error as a function of the noise in measurements. Since we linearize around the measurements, the approximation works better for smaller noise levels.

B. Evaluation of Local Refinement

Next we will compare how well local refinement works when computing residuals with $\mathcal{E}_{QS}$ compared with $\mathcal{E}_{ML}$. The rest of the experiments in this section and the next use a data set that has been collected in a motion capture lab and consists of 12 microphones (receivers) and a sound source that moved around in the room [21]. Figure 2 shows their ground truth positions. While originally a TDOA-data set, offsets have been removed using ground truth to simulate TOA measurements. The distance measurements within 0.05m of the ground truth are considered to be inliers and the rest as outliers. For the experiment we select subsets of this data, randomly choosing four receivers together with a random selection inliers and outliers. Since the problem degenerates for configurations with planar receivers, we filter out these from our random selection.

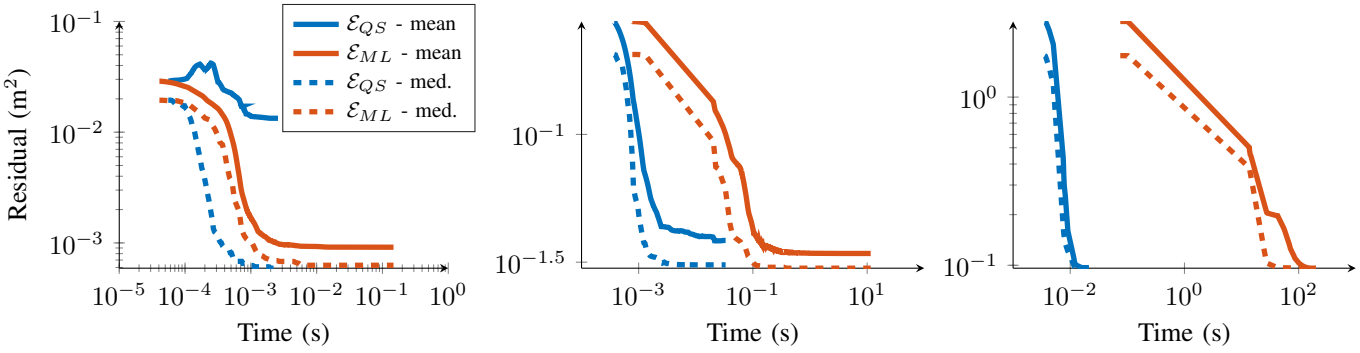

Fig. 4. The residuals of (2) plotted against time for 10, 100, and 1000 sources. For the case of 10 senders we see that the average values are much worse using the $\mathcal{E}_{QS}$ as compared to $\mathcal{E}_{ML}$. This behaviour is not seen in the median. The success rate thus seems to be lower when using fewer senders. This is not a large problem as the gains of using this approximation is larger for more senders. As we can see in the leftmost plot, the proposed optimizer is only slightly faster, but with more senders this time gain is much larger. The rightmost graph shows that with 1000 senders we can get similarly accurate results several orders of magnitude faster.

In Figure 4 we can see the convergence over time, when having 10, 100 and 1000 senders, respectively. It is run on the real data set with random choices of senders and receivers. The receiver initial point is chosen as ground truth positions with zero mean Gaussian noise with $\sigma_B = 0.05m$ added to each coordinate. From this, sender coordinates are trilaterated (using [19]) as the additional initialization points for the full optimizer. For 10 and 100 points we show the mean error over time for 100 random choices of datasets. Due to the longer run time for the full optimizer we only ran 10 different data sets with 1000 senders.

We can see that the proposed optimizer is faster overall with comparable results. Optimizing over the full problem ($\mathbf{r}$ and $\mathbf{s}$) is significantly slower with an increased number of senders—with 1000 senders it can take from half a minute to several minutes to get close to convergence. In comparison, the QS-optimizer runs in less than a second even for 1000 senders, which is several orders of magnitude faster.

In Figure 5 we can see the success rate for the two optimizers. They are run on real data with 100 senders and initial point is chosen by adding zero mean Gaussian noise to each of the receivers coordinates for some value of σ_B. They are run for 100 different samples of data sets, and considered successful if the final value is less than 0.09. For comparison the ground truth values gave around 0.063.

For small enough values of σ_B the initial point will be close enough to the real value to already give a solution considered successful. Still if the QS-optimizer worked poorly it could technically fail since it only optimizes approximate values. This is not the case however and we see that both almost always succeeds for noise levels smaller than $\sigma_B = 0.1$. When we start further away from the initial position we see that while both succeeds less often, the full optimizer succeeds much more often than the QS-optimizer.

C. Evaluation of Self-Calibration

In Figure 6 the different RANSAC variants are run on the real data with 100 inliers and 50 outliers. We consider the currently best solution over time in the RANSAC loop and

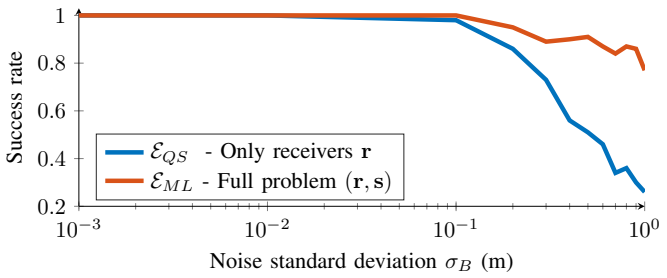

Fig. 5. Comparison with full refinement of (r, s). The plot shows the success rate after initializing with noisy initial estimates.

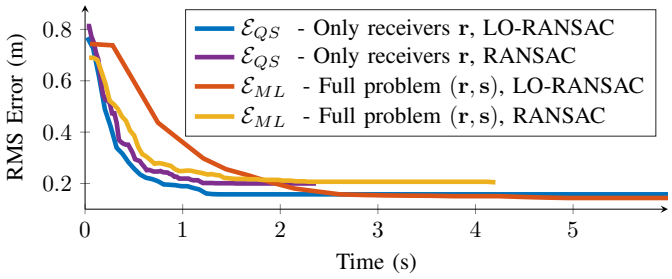

Fig. 6. Comparison of median RMS error of receivers to ground truth over time in RANSAC loop for the different solver options.

compute the RMS error of the four distances between this solution and the ground truth positions. The plot shows the median values over 200 runs. The baseline RANSAC variant with local optimization gets the closest result, but also takes the longest time. The QS-LO variant performs better than the baseline without LO as it achieves a slightly better result faster. We used $\sigma^2 = 0.01$ as the threshold in (14) for all experiments in this section. This worked well but could potentially be optimized. Using the same setup, if we instead look at the classification of inliers vs. outliers, we can in Figure 7 see the F1 score of this classification as a function of time. We can see that all variants achieve approximately the same value, but the QS-variants are faster, by a slight margin compared to the baseline without LO and a few seconds faster than the baseline with local optimization.

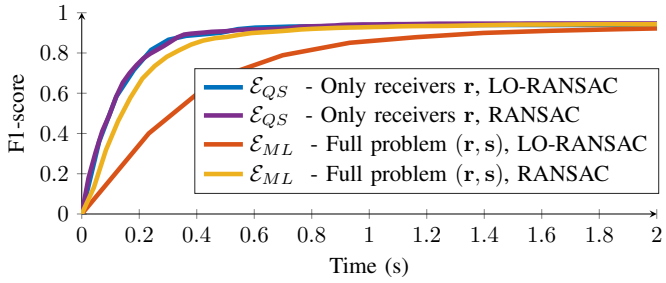

Fig. 7. Comparison of F1 values of inliers vs. outliers over time in RANSAC loop for the different solver options.

TABLE I
RUN TIMES IN SECONDS FOR DIFFERENT SOLVERS FOR DIFFERENT NUMBER OF SENDERS. BEST RESULTS IN BOLD.

Method	$n = 10$		$n = 100$		$n = 1000$	
	init	total	init	total	init	total
$\mathcal{E}_{ML}$ RS	0.41	0.42	0.77	1.1	4.2	150
$\mathcal{E}_{ML}$ LO-RS	0.39	0.40	2.0	2.3	340	540
$\mathcal{E}_{QS}$ RS	0.37	0.38	0.42	0.44	**0.66**	**0.91**
$\mathcal{E}_{QS}$ LO-RS	**0.33**	**0.34**	**0.38**	**0.41**	0.75	0.99

In table I we can see the run time for different amount of senders for a couple of RANSAC variants: RANSAC using $\mathcal{E}_{ML}$ ($\mathcal{E}_{ML}$ RS), LO-RANSAC using $\mathcal{E}_{ML}$ ($\mathcal{E}_{ML}$ LO-RS), RANSAC using $\mathcal{E}_{QS}$ ($\mathcal{E}_{QS}$ RS) and LO-RANSAC using $\mathcal{E}_{QS}$ ($\mathcal{E}_{QS}$ LO-RS). There are times both with and without final refinement. They are run on real data with 4, 40 and 400 outliers for 10, 100 and 1000 inliers, respectively. For 10 and 100 senders they are average times for 100 runs. Due to the longer times for 1000 inlier senders, these are only the average of 3 runs for the $\mathcal{E}_{ML}$ LO-RS and 10 runs for the others. The times are fairly similar for 100 senders and less, but with 1000 senders there are some differences. $\mathcal{E}_{ML}$ RS without refinement takes a little bit longer than the $\mathcal{E}_{QS}$ RS variants. With this many senders, the speedup from using the QS-approximation of the errors starts being noticeable. The $\mathcal{E}_{ML}$ RS with refinement and the $\mathcal{E}_{ML}$ RS-LO variants are significantly more time consuming than the QS variants. This is explained by how much longer time the full optimizers take compared to the QS-approximation optimizers, as could be seen in the previous subsection.

VI. CONCLUSION

In this paper we have shown how sensor node calibration for imbalanced TOA-problems can be significantly sped up, using a novel implicitization scheme. Using the so-called quadrisonal tensor, we parameterize such problems using only six parameters, regardless of the number of sources. In order to arrive at a geometrically proper optimization problem, we linearize the problem around the measured data points, using a Sampson-like approximation. This leads to highly efficient and robust algorithms for both initialization and refinement, with negligible accuracy losses, compared to running the full optimization over all parameters. Here we target the restricted problem of four receivers, but we believe that our framework can be extended to larger problem settings in a natural way.

REFERENCES

[1] A. Plinge, F. Jacob, R. Haeb-Umbach, and G. A. Fink, "Acoustic microphone geometry calibration: An overview and experimental evaluation of state-of-the-art algorithms," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 14–29, 2016.

[2] S. T. Birchfield and A. Subramanya, "Microphone array position calibration by basis-point classical multidimensional scaling," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, 2005.

[3] M. Crocco, A. Del Bue, M. Bustreo, and V. Murino, "A closed form solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012.

[4] J. C. Chen, R. E. Hudson, and K. Yao, "Maximum likelihood source localization and unknown sensor location estimation for wideband signals in the near-field," *IEEE transactions on Signal Processing*, vol. 50, 2002.

[5] P. Pertila, M. Hamalainen, and M. Mieskolainen, "Passive temporal offset estimation of multichannel recordings of an ad-hoc microphone array," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 11, pp. 2393–2402, Nov. 2013.

[6] H. Stewénius, "Gröbner basis methods for minimal problems in computer vision," Ph.D. dissertation, Lund University, 2005.

[7] Y. Kuang, S. Burgess, A. Torstensson, and K. Åström, "A complete characterization and solution to the microphone position self-calibration problem," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2013.

[8] K. Batstone, M. Oskarsson, and K. Åström, "Robust time-of-arrival self calibration with missing data and outliers," in *European Signal Processing Conference (EUSIPCO)*, 2016.

[9] V. Larsson, K. Åström, and M. Oskarsson, "Polynomial solvers for saturated ideals," in *International Conference on Computer Vision (ICCV)*. IEEE, 2017.

[10] M. Kreković, I. Dokmanić, and M. Vetterli, "Echoslam: Simultaneous localization and mapping with acoustic echoes," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 11–15.

[11] Y. Kuang and K. Åström, "Stratified sensor network self-calibration from tdoa measurements," in *European Signal Processing Conference (EUSIPCO)*, 2013.

[12] L. Wang, T. Hon, J. D. Reiss, and A. Cavallaro, "Self-localization of ad-hoc arrays using time difference of arrivals," *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 1018–1033, Feb 2016.

[13] M. Larsson, G. Flood, M. Oskarsson, and K. Åström, "Fast and robust stratified self-calibration using time-difference-of-arrival measurements," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 4640–4644.

[14] D. El Badawy, V. Larsson, M. Pollefeys, and I. Dokmanić, "Localizing unsynchronized sensors with unknown sources," *IEEE Transactions on Signal Processing*, 2023.

[15] P. D. Sampson, "Fitting conic sections to "very scattered" data: An iterative refinement of the bookstein algorithm," *Computer graphics and image processing*, vol. 18, no. 1, pp. 97–108, 1982.

[16] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–95, 1981.

[17] O. Chum, J. Matas, and J. Kittler, "Locally optimized ransac," in *Joint Pattern Recognition Symposium*. Springer, 2003, pp. 236–243.

[18] K. Lebeda, J. Matas, and O. Chum, "Fixing the locally optimized ransac–full experimental evaluation," in *British machine vision conference*, vol. 2. Citeseer, 2012.

[19] M. Larsson, V. Larsson, K. Astrom, and M. Oskarsson, "Optimal trilateration is an eigenvalue problem," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5586–5590.

[20] S. Agarwal, K. Mierle, and T. C. S. Team, "Ceres Solver," 3 2022. [Online]. Available: https://github.com/ceres-solver/ceres-solver

[21] E. Tegler, M. Larsson, M. Oskarsson, and K. Åström, "Sensor node calibration in presence of a dominant reflective plane," in *2022 30th European Signal Processing Conference (EUSIPCO)*. IEEE, 2022, pp. 1941–1945.