# Differential Multidimensional Scaling for Self-Localization of TDOA Sensor Networks


Johannes Schmitz, Saeed Shojaee,
Vimal Radhakrishnan, Rudolf Mathar
RWTH Aachen University
52074 Aachen, Germany
Email: {schmitz, mathar}@ti.rwth-aachen.de

Sivan Toledo
Tel-Aviv University
Tel-Aviv 69978, Israel
Email: stoledo@tau.ac.il

Roberto Carlos Hincapié Reyes
Universidad Pontificia Bolivariana
Medellín, Colombia
Email: roberto.hincapie@upb.edu.co



***Abstract*—We present a novel algorithm for self-localization in sensor networks without any prior knowledge on the locations of the sensors. We assume that all sensors in the network can receive and transmit, thus we obtain time difference of arrival measurements for all combinations of sensors. Using the full set of these differences in arrival times in the network we are able to obtain the relative location of the sensors nodes, the shape of the network. This leaves us with the problem of anchoring the network to its absolute location, which we solve using additional transmitting beacons at known locations. Experimental results from numerical simulation demonstrate the performance of our approach under various conditions.**



***Index Terms*—Time Difference of Arrival, Sensor Networks, Self-localization, Multidimensional Scaling**


## I. Introduction

In distributed localization systems, knowledge about the sensor locations is crucial. Often, due to practical issues, this information is incomplete or erroneous. In ad-hoc sensor networks the locations might be even completely unknown. To overcome this problem various methods for calibration and self-localization have been developed, which can be classified into active and passive. Active methods require the sensors to receive as well as transmit, while passive methods must rely on signals transmitted by targets because the sensors can only receive. Passive self-localization approaches for time difference of arrival (TDOA) systems have been developed only very recently [1], [2]. In general, the optimization problems associated with active as well as passive self-localization are not convex. Research results on passive methods indicate that the underlying minimization problem severely suffers from many local minima, so great effort is necessary to find the global optimum and thereby the sensor locations. In this paper we therefore consider active self-localization based on time difference of arrival (TDOA) measurements of the received signals. Due to the non-convex nature of the problem it is difficult to provide mathematical rigorous proofs of convergence. However, our numerical simulations show a more manageable behavior of the passive problem with respect to local minima as the passive problem tends to have no local minima close to the global optimum solution.

Classical active self-localization relies on distance measurements between sensors [3], [4], [5]. The relative sensor locations can be determined from this distance measurements using the theory of multidimensional scaling [6]. This assumes that the sensors are able to perform two way ranging in order to determine the pairwise distances. However, in TDOA systems, the ability to perform ranging between sensors might not be available due to technical reasons. On the other hand, differential measurements between two receiving sensors and a third transmitting sensor are a natural operation and require insignificant changes in the system software. However, there is no known approach to solve the multidimensional scaling problem from differential measurements. Therefore, in the present paper we introduce and study an iterative algorithm that is able to derive the sensor locations from the full set of TDOA measurements between all combinations of sensors. Shepard and later Kruskal developed an algorithm now known under the names of the authors [7], [8]. The algorithm iteratively decreases a cost function called *Stress*, much like a gradient descent type algorithm. We essentially define a new Stress function for the time difference measurements, to develop a similar iterative approach of finding the sensor locations by minimizing this Stress. Further, we compare the resulting algorithm with a numerical gradient descent on the Stress function.

A practical real-world application that motivates our work, is to apply it to the ATLAS system [9]. This is a wildlife tracking system in Hula Valley in northern Israel, an area that lies on an important bird-migration route. For the ATLAS project, birds are tagged with a transmitter and their movements can be observed in real-time. The system uses a number of basestations that perform time of arrival measurement of the bird tags. These arrival times are sent to a central fusion center that calculates the locations based on the resulting TDOAs. Locations of the sensor antennas, most of which are mounted on cellular network towers, can be erroneous, e.g., due to the GPS antenna not placed next to the receive antenna or errors in the GPS position. The ATLAS system has some transmitting beacons in well known absolute locations. We explain how those can be used to anchor the solution of the self-localization.

The paper is structured as follows. Section II defines a system model and notations. Sections III introduces the algorithms for differential multidimensional scaling based on TDOA measurements. Following that, Section IV discusses the problem of anchoring the solution to absolute locations. Finally, Section V provides numerical simulation results to give a better insight into the algorithm and to study its performance under noisy conditions. A short conclusion of the research is given in Section VI.

## II. System Model

We consider a system of $M$ sensor nodes located in a two dimensional euclidean space. The sensor locations $\mathbf{x}_j = [x_{j1} \ \ x_{j2}]^T, j \in [1 \dots M]$ are initially unknown. Further a number of $N$ beacons with known locations $\mathbf{x_b}$ are available as anchors. The beacons are constantly transmitting and are not able to receive whereas the sensors are able to receive as well as transmit signals. After deployment the system enters into a self-localization phase. In this phase, each sensor transmits a calibration signal while all the other sensors are listening. We assume the sensors to be perfectly time synchronized. Further, we assume the signals emitted by the sensors to be orthogonal to the beacon signals such that they do not interfere with the measurements.

Assuming free-space propagation, the TDOA measurements for a transmitting sensor at location $\mathbf{x}_j$ and receiving sensors at locations $\mathbf{x}_k$ and $\mathbf{x}_l$ can be expressed as

$$\tau_{j,k,l} = \frac{1}{c}\|\mathbf{x}_j - \mathbf{x}_k\|_2 - \frac{1}{c}\|\mathbf{x}_j - \mathbf{x}_l\|_2 + \eta, \quad j \neq k \neq l$$

where $c$ denotes the speed of the wave and $\eta$ is a Gaussian distributed noise term. Equivalently we define distance differences

$$\Delta_{j,k,l} = c * \tau_{j,k,l} + \eta'.$$

The full set of $\Delta_{j,k,l}$ contains $M(M-1)(M-2)$ measurements, the problem is symmetric in the sense that $\Delta_{j,k,l} = -\Delta_{j,l,k}$, this can be exploited to further speed up the algorithm described in the next section. We also define the matrix $D_{j,k} = \|\mathbf{x}_j - \mathbf{x}_k\|$ of all pairwise sensor distances.

## III. Differential Multidimensional Scaling

After obtaining the full set of differential measurements we would like to estimate the locations of the sensors. Clearly, if the locations of the sensors are rigidly translated, rotated, and reflected, the pairwise distances and their differences will not change. Therefore, we can only determine sensor locations up to a transformation of the plane (a composition of translation, rotation, and reflection). We refer to these relative locations as a *configuration*.

Multidimensional scaling, as first introduced in [6], provides a well known solution to this problem if the distance matrix $D$ is known. We introduce an algorithm that can solve for a configuration even if only the differential distances $\Delta_{j,k,l}$ are known. We call this algorithm differential multidimensional scaling (DMDS). Clearly, this is only possible for a set of more than three sensors. Showing the ambiguity of a three sensor solution is easily possible by finding two sets of sensor locations for the same difference measurements. Therefore, we require a minimum of four sensors. The algorithm is inspired by the Shepard-Kruskal algorithm [7], [8], [10] that finds a configuration by defining an iterative rule for moving the points $\hat{\mathbf{x}}^i$ to new points $\hat{\mathbf{x}}^{i+1}$.

### A. Stress criteria for differential MDS

In order to evaluate the fitness of the configuration, we define the function

$$\hat{\Delta}_{j,k,l} = \|\hat{\mathbf{x}}_j^i - \hat{\mathbf{x}}_k^i\|_2 - \|\hat{\mathbf{x}}_j^i - \hat{\mathbf{x}}_l^i\|_2, \quad j \neq k \neq l.$$

Now we introduce a cost function (Stress) $S$, equivalent to the Stress criteria defined by Kruskal [8],

$$S(\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_M) = \sqrt{\frac{\sum_{j,k,l}(\Delta_{j,k,l} - \hat{\Delta}_{j,k,l})^2}{\sum_{j,k,l}\Delta_{j,k,l}^2}},$$

This is a functional that maps $\mathbb{R}^{2\times M}$ ($M$ points in the plane) to $\mathbb{R}$.

### B. Iterative displacement of points

We start with a uniformly-distributed random configuration $\hat{\mathbf{x}}_j^0$, $j \in \{1 \dots M\}$. Ensuring $\hat{\mathbf{x}}_j^0 \neq \hat{\mathbf{x}}_k^0$ for $j \neq k$, we then seek to minimize $S$. The classical algorithm introduced by Sheppard intuitively achieves this by comparing all distances of tuples in the configuration to the measurements and then displacing the points accordingly, i.e., moving them slightly closer or further away from all other points of the configuration in each iteration. However, in our problem we have differential measurements, consisting of triples of points, i.e. all combinations of one transmitting sensor and two receiving sensors. Therefore, it is not directly obvious how to displace points in order to achieve convergence to the minimum-Stress configuration. Based on empirical studies, we propose to displace only the receiving sensors in each triple and to choose the direction of displacement as the direction from the receiver to the transmitter,

$$\mathbf{b}_{jk} = \frac{\hat{\mathbf{x}}_k^i - \hat{\mathbf{x}}_j^i}{\|\hat{\mathbf{x}}_k^i - \hat{\mathbf{x}}_j^i\|}.$$

The velocity of displacement is chosen based on the residual error between the current configuration and the measurement,

$$v_{jkl} = \Delta_{j,k,l} - \hat{\Delta}_{j,k,l}.$$

From this, we determine the displacement vector of each point in each iteration as

$$\mathbf{d}_j = \sum_{kl} v_{jkl}\mathbf{b}_{jk}.$$

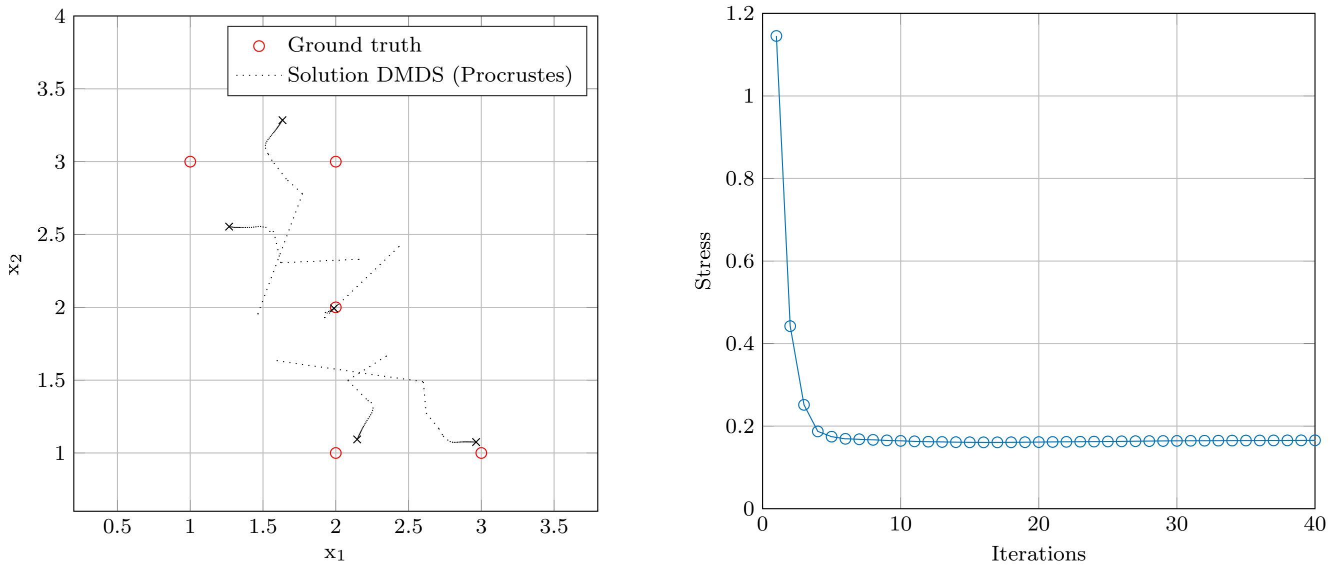
Finally, the new location of each point is calculated

$$\hat{\mathbf{x}}_j^{i+1} = \hat{\mathbf{x}}_j^i - \alpha\frac{1}{(M-1)(M-2)}\mathbf{d}_j,$$

where $(M-1)(M-2)$ is a normalization term and $\alpha$ is a step-size parameter that we choose experimentally. After each iteration, we shift the configuration so that its center of mass coincides with the origin,

$$\mathbf{o}^{i+1} = \frac{1}{M}\sum_j \hat{\mathbf{x}}_j^{i+1},$$

$$\hat{\mathbf{x}}_j^{i+1} = \hat{\mathbf{x}}_j^{i+1} - \mathbf{o}^{i+1}.$$




(a) (b)

Fig. 1: Example of the algorithm running into a local minimum (a) and the evolution of the Stress function (b), showing that it has converged but not to a the near-zero value we expect to see in the ground-truth configuration.

The procedure is repeated until convergence. Criteria for convergence are discussed next.

### *C. Convergence and stopping criteria*

A possible measure for convergence is the Stress function $S$. We define a threshold $\epsilon$ and assume convergence if $S < \epsilon$. Due to the general non-convex nature of the problem, the proposed multidimensional scaling procedure is not always guaranteed to converge to the global optimum and might get stuck in a local minimum. Therefore, we introduce a second criteria to measure the change of $S$

$$R^i = S^{i+1} - S^i$$

If $R < \delta$ and $S > \epsilon$ we assume to have reached a local minima. A scenario where a local minimum has been reached is depicted in Fig. 1. These thresholds need to be determined empirically, especially for the case of noisy measurements.

### *D. Overcoming local minima*

The non-convexity of the problem makes it difficult to find a solution within a limited runtime. However, we offer two ideas to help cope with local minima. A simple practical solution to overcome the minima problem is to restart the algorithm with a new random configuration. In simulations with up to 100 nodes we found the probability of running into local minima is so low that we could always find the global minima. Another idea that works for some scenarios is to evaluate a Stress function for individual nodes and interchange the location of nodes that exhibit the strongest stress values.

### *E. Numerical gradient descent*

At this point, we have fully defined an iterative algorithm that minimizes the Stress function and thereby finds the locations of the sensors. For comparison we will now formulate the problem in a way that enables us to perform a gradient descent. The Stress $S$ is essentially a scalar functional

$$f(\hat{\mathbf{X}}) = \sum_{j,k,l} \left( \Delta_{j,k,l} - \hat{\Delta}_{j,k,l}(\hat{\mathbf{X}}) \right)^2$$

of the vector $\hat{\mathbf{X}} = [\hat{x}_{11}, \hat{x}_{12}, \ldots, \hat{x}_{M1}, \hat{x}_{M2}]^T$, $\hat{\mathbf{X}} \in \mathbb{R}^{2M}$ that specifies locations $\hat{\mathbf{x}}_j$ for the sensors. The minimizer of

$$\min_{\hat{x}_{11}, \hat{x}_{12}, \ldots, \hat{x}_{M1}, \hat{x}_{M2}} f(\hat{\mathbf{X}}),$$

also solves the original self-localization problem. A common approach to minimize $f$ is to iteratively perform a gradient descent step, moving in the opposite direction to the gradient

$$\nabla f(\hat{\mathbf{X}}) = \left[ \frac{\partial f}{\partial \hat{x}_{11}}, \frac{\partial f}{\partial \hat{x}_{12}}, \ldots, \frac{\partial f}{\partial \hat{x}_{1M}}, \frac{\partial f}{\partial \hat{x}_{2M}} \right]^T . \quad (1)$$

The gradient descent method starts with a random $\hat{\mathbf{X}}^0$ value as an initial position guess and iteratively calculates new values by the recursive formula

$$\hat{\mathbf{X}}^{i+1} = \hat{\mathbf{X}}^i - \alpha \frac{1}{(M-1)(M-2)} \nabla f(\hat{\mathbf{X}})|_{\hat{\mathbf{X}} = \hat{\mathbf{X}}^i} . \quad (2)$$

As the analytical evaluation of the gradient (1) used in (2) is impractical for large a number of sensors, another way is to numerically evaluate it in a local region $2\Delta$ around the variables

$$\frac{\partial f}{\partial \hat{x}_{jk}} = \frac{f([\hat{x}_{11}, \ldots, \hat{x}_{jk} + \Delta, \ldots, \hat{x}_{M2}])}{2\Delta} - \frac{f([\hat{x}_{11}, \ldots, \hat{x}_{jk} - \Delta, \ldots, \hat{x}_{M2}])}{2\Delta}, \quad k \in (1, 2).$$

Similar to the previous algorithm, $\alpha$ has to be determined experimentally and measures for overcoming local minima have to be taken.

## IV. Anchoring

After convergence of the DMDS we have a configuration that exhibits the correct shape of the sensor network, but in general is shifted, rotated and reflected. Therefore, we introduce anchoring points at known locations. In practice these can be very inexpensive, constantly transmitting beacons, with no receive capability. They should be placed at a point that is accessible and can be localized with high accuracy using methods from land surveying. For example, in the ATLAS system [9], they are placed on the roof of a bird-watching huts. In two dimensions we require at least 3 beacons in order to find the true locations using the following approach. Based on the final configuration obtained from DMDS, we localize the beacons using a TDOA localization algorithm as described in [11]. Other TDOA localization algorithms from the literature might be used as well, which leads to similar results. Next we perform Procrustes analysis [12] to determine the transformation

$$\mathbf{P} = \begin{bmatrix} \cos\theta & \sin\theta & a \\ -\sin\theta & \cos\theta & b \end{bmatrix}$$

that maps the localized beacon locations to the true ones with least squared error using a shift specified by $[a \;\; b]^T$ and a rotation by the angle $\theta$. Finally, we apply the same transformation to the sensor locations in order to find the true locations of the sensors on the global map.

$$\hat{\mathbf{y}}_j = \mathbf{P}\,[\hat{\mathbf{x}}_j^T \; 1]^T.$$

Note that due to the dilution of precision problem [13], it is important to have beacons located in the center of the system, otherwise the accuracy of the solution might severely be affected. Each solution step of the process is shown in Fig. 2. This concludes the self-localization phase; the system may now enter into its operational state.

## V. Results

We have proposed an iterative algorithm for self-localization in TDOA sensor networks. This section presents some numerical simulation results in order to investigate and visualize its performance. First a value for the thresholds that are needed to detect convergence of the algorithm has been experimentally determined. Values of $\epsilon = 0.001$ and $\delta = 0.001$ have been found to be working well. Further, good values for the step size, controlled by $\alpha$ have to be determined. Note that for values of $\alpha$ that are too high, oscillation and divergence of the algorithm can occur. Whereas for very small values of $\alpha$, the number of necessary iterations becomes large. For all presented results the value of $\alpha$ has been set to 1 in the DMDS and 0.1 in the gradient descend algorithm, which yields a good compromise between stability and speed of convergence. Two types of experiments have been performed. First the proposed DMDS algorithm has been compared with the gradient descent. The same initial configuration has been used for both algorithms. Fig. 3 shows an example for 6 sensor nodes and measurements not affected by noise. It can be observed that the trajectories of both algorithms follow

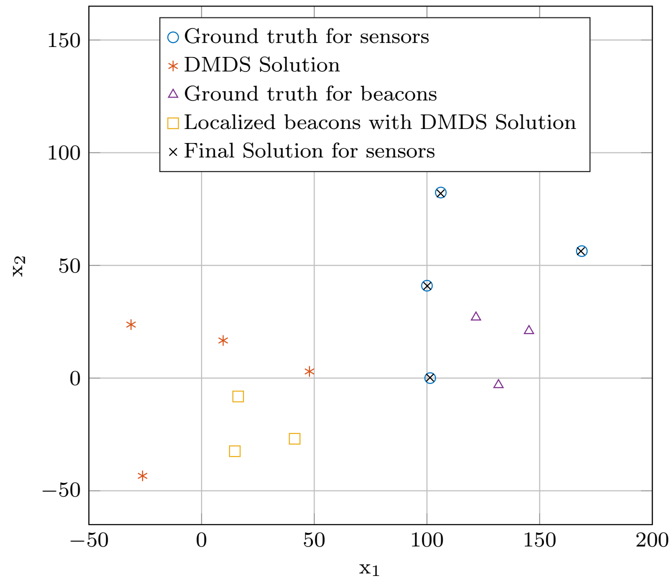



Fig. 2: The steps of the self-localization and anchoring process visualized. The final solution is obtain using Procrustes analysis on the localized beacons.

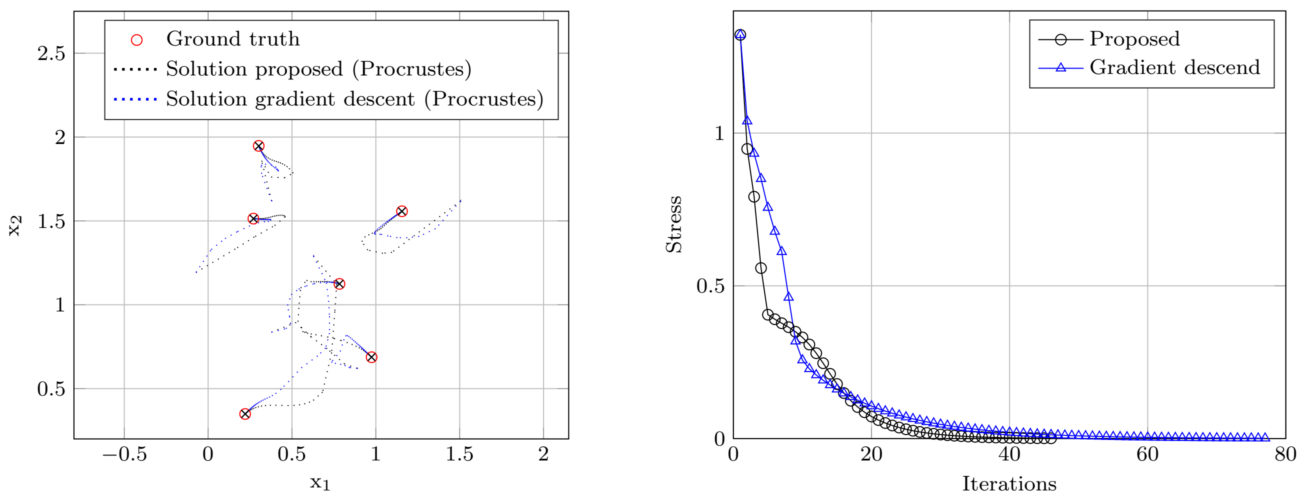
a similar path. None of the algorithms has been observed to be always quicker. Repeating the experiment for many different random sensor locations, the convergence times of the two algorithms have been found to be comparable. Similarly, both of the algorithms might run into local minima. However, this doesn't necessarily occur jointly in both algorithms for a certain random instance of locations. In a second experiment the performance under noise conditions has been studied. For that, a regular grid of sensors as shown in Fig. 4 has been used. The noise $\eta \sim \mathcal{N}(0, \sigma^2)$ on the measurements is increased step wise and the performance in terms of root means square error (RMSE) is observed for different numbers of sensors. We observe the algorithm to be stable as long as the noise term is reasonably low. Further, for the considered sensor placement an increased number of sensors increases the robustness of the system against noise. This can be intuitively explained by the increased number of available measurements.

## VI. Conclusion

An iterative multidimensional scaling algorithm for the case of differential measurements obtained from triples of sensor nodes has been introduced Further, it has been shown how to anchor the obtained configuration using additional beacons at known absolute locations. Different simulation results have been presented to evaluate the behavior and effectiveness of the new scheme. We suggest it to be deployed in TDOA sensor networks if two way ranging and thus classical multidimensional scaling is not easily possible as it may require heavy changes to the hardware and software of the sensors.

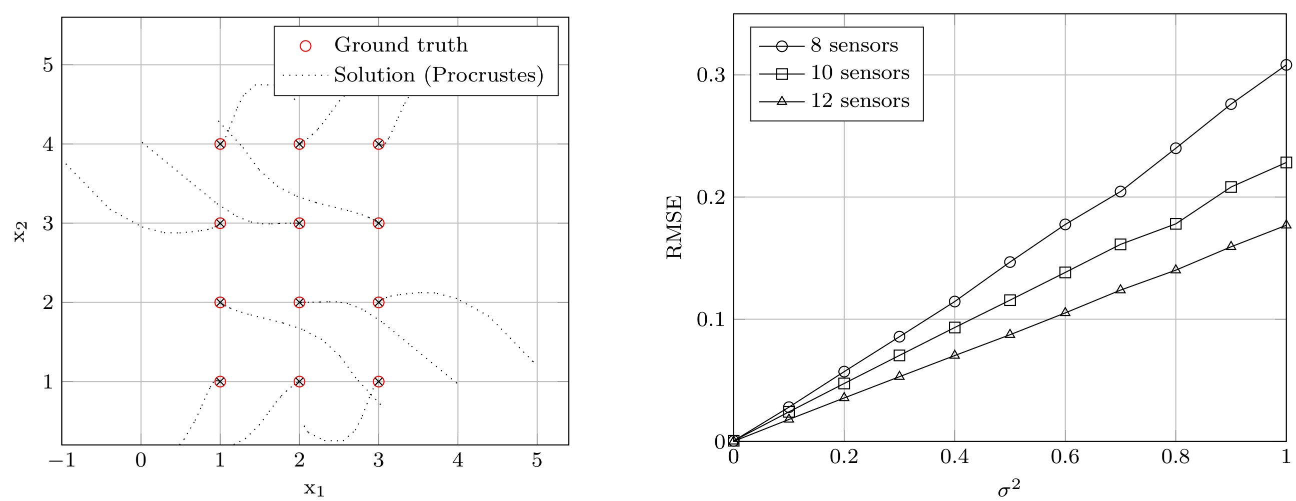
## VII. Acknowledgment


This work was partially supported by the Minerva short-term research grant of Max-Planck-Gesellschaft and the DFG grant MA 1184/23-1.




(a) (b)

Fig. 3: Trajectories of the sensor nodes during the iterations of the algorithm (a), together with the Stress functions (b).




(a) (b)

Fig. 4: A simple scenario to study the behavior with noisy measurements (a). Adding more sensors and thereby more measurements, results in improved robustness of the self-localization against the measurement noise (b).

## References


[1] J. Wendeberg, F. Höflinger, C. Schindelhauer, and L. Reindl, "Calibration-free TDOA self-localisation," *Journal of Location Based Services*, vol. 7, no. 2, pp. 121–144, 2013.

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

[3] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proc. 2nd Int. Conf. on Embedded Networked Sensor Systems (SenSys)*. ACM, 2004, pp. 50–61.

[4] X. Ji and H. Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *IEEE INFOCOM 2004*, vol. 4. IEEE, 2004, pp. 2652–2661.

[5] S. T. Birchfield and A. Subramanya, "Microphone array position calibration by basis-point classical multidimensional scaling," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 5, pp. 1025–1034, 2005.

[6] W. S. Torgerson, *Theory and methods of scaling.* Wiley, 1958.

[7] R. N. Shepard, "The analysis of proximities: Multidimensional scaling with an unknown distance function. I." *Psychometrika*, vol. 27, no. 2, pp. 125–140, 1962.

[8] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, 1964.

[9] A. W. Weiser, Y. Orchan, R. Nathan, M. Charter, A. J. Weiss, and S. Toledo, "Characterizing the accuracy of a self-synchronized reverse-GPS wildlife localization system," in *ACM/IEEE Int. Conf. on Information Processing in Sensor Networks (IPSN)*. IEEE, 2016, pp. 1–12.

[10] R. N. Shepard, "The analysis of proximities: Multidimensional scaling with an unknown distance function. II." *Psychometrika*, vol. 27, no. 3, pp. 219–246, 1962.

[11] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Trans. Signal Process.*, vol. 42, no. 8, pp. 1905–1915, Aug 1994.

[12] J. C. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.

[13] J. D. Bard and F. M. Ham, "Time difference of arrival dilution of precision and applications," *IEEE Trans. Signal Process.*, vol. 47, no. 2, pp. 521–523, 1999.