ML-BASED SENSOR NETWORK LOCALIZATION AND TRACKING: BATCH AND TIME-RECURSIVE APPROACHES

Pinar Oguz-Ekim, João Gomes, João Xavier, and Paulo Oliveira

Institute for Systems and Robotics – Instituto Superior Tecnico
Av. Rovisco Pais, 1049–001 Lisboa, Portugal
phone: + (351) 218418296, fax: + (351) 218418291, email: {poguz, jpg, jxavier, pjcro}@isr.ist.utl.pt
web: www.isr.ist.utl.pt

ABSTRACT

Simultaneous localization and tracking (SLAT) in sensor networks aims to determine the positions of sensor nodes and a moving target in a network, given incomplete and inaccurate range measurements. One of the established methods for achieving this goal is to maximize a likelihood function (ML), which requires initialization with an approximate solution to avoid convergence towards local extrema. In this paper a Euclidean Distance Matrix (EDM) completion problem is solved to obtain initial sensor/target positions. The likelihood function is then iteratively optimized through either a Majorization-Minimization (MM) or Newton method. To reduce the computational load, an incremental scheme is proposed whereby each new target position is estimated from range measurements, providing additional initialization for ML without the need for solving an expanded EDM completion problem. The performance of these methods is assessed through simulation.

1. INTRODUCTION

This work addresses the problem of tracking a single target from distance-like measurements taken by nodes in a sensor network whose positions are not precisely known. The goal is to estimate the position of all the sensors and the target, given only partial or no *a priori* information on the spatial configuration of the network. The ability to track a target is a key component in several scenarios of wireless sensor networks, and avoiding the need for careful calibration of sensor positions is practically relevant.[1]

In [1], [2] SLAT is formulated in a Bayesian framework that resembles the related and well-studied problem of Simultaneous Localization and Mapping (SLAM) in robotics. The *a posteriori* probability density function of sensor/target positions and calibration parameters is recursively propagated in time as more target sightings become available. In [1], these observations are true range measurements obtained through a combination of transmitted acoustic and radio pulses, whereas in [2] range and bearing information is estimated from camera images. Range can also be estimated from the Received Signal Strength (RSS) of radio transmissions [3], although these are less reliable than the direct measurements used in [1].

Algorithm initialization issues are only very briefly considered in [1], [2], but the underlying assumption is that the initial position estimates should be sufficiently close to the true spatial configuration to avoid convergence to undesirable local extrema. In this paper, an EDM completion problem is proposed to initialize the iterative ML algorithm with little *a priori* knowledge of sensor/target positions. A similar idea for localization and tracking based on EDM has independently been discussed in [3], although the authors pursue a distinct method for approximately solving the completion problem. Related EDM-like approaches have also been adopted previously for localization of static sensor network nodes [4], [5].

This paper focuses on plain ML estimation, rather than MAP/Bayesian estimation used in [1], [2]. A basic iterative optimization approach using the MM or Newton methods is first developed for batch estimation, i.e., when all measurements are processed simultaneously. A time recursive method is then obtained by estimating each target position as the corresponding range measurements become available, and then re-optimizing the expanded ML cost function with a few iterations of the batch method. This recursive approach only requires EDM initialization at the first time step, which is computationally less complex than processing all target measurements. We use a technique proposed in [6] to obtain a cost function for incremental target position estimation which, despite being non-convex, can be globally optimized using efficient numerical tools.

The main technical contribution of this paper is the proposed time recursive ML estimation method. Our derivations of the MM and Newton methods for maximizing the likelihood function with EDM initialization have also not appeared in the literature, although a similar MM method is given in [7] using a slightly different cost function and majorization approximations.

The paper is organized as follows. In section 2, the SLAT and EDM completion problems are introduced. Sections 3 and 4 develop the MM and Newton methods for iterative likelihood maximization, respectively. Section 5 develops the time recursive method using incremental estimation of target positions. Numerical results for two distinct simulation scenarios

[1] This work was supported by Fundacão para a Ciência e a Technologia (ISR/IST plurianual funding). The work of P. Oguz-Ekim was supported by a Phd Student Scholarship, SFRH/BD/44771/2008, from the Portuguese FCT POCTI programme.

of batch and time recursive approaches are presented in Section 6. Finally, Section 7 summarizes the main conclusions and discusses directions for future research.

2. PROBLEM FORMULATION AND EUCLIDEAN DISTANCE MATRIX COMPLETION

2.1 Problem Formulation

The network comprises sensors at unknown positions $\{x_1, x_2, \dots, x_n\} \in R^2$, a set of reference sensors (anchors) at known positions $\{a_1, a_2, \dots, a_l\} \in R^2$, and target positions $\{e_1, e_2, \dots, e_m\} \in R^2$. A central processing node has access to range measurements between each target position and all sensors and anchors, namely, $d_{ij} = \|x_i - e_j\| + w_{ij}$ and $d_{kj} = \|a_k - e_j\| + w_{kj}$, where w_{ij}, and w_{kj} denote noise terms and. A practical system that provides such range measurements is used, e.g., in [1]. If errors are Gaussian, independent and their variances are identical, maximizing the likelihood for the full batch of observations is equivalent to minimizing the cost function

$$\theta(x) = \sum_{i,j}\left(\|x_i - e_j\| - d_{ij}\right)^2 + \sum_{k,j}\left(\|a_k - e_j\| - d_{kj}\right)^2. \tag{1}$$

The full set of unknown sensor and target positions is concatenated into column vector x, the argument of θ. Due to the nature of this problem the function θ is invariant to global rotation, translation and reflection in the absence of anchors. In our simulations $l = 3$ anchors are used, which is enough to remove those ambiguities and to obtain a well-posed optimization problem. Although the localization problem is formulated here in R^2 the proposed algorithms could handle any embedding dimension by appropriate number of anchors. As in many other ML problems, the function θ is in general nonconvex and multimodal, hence iterative optimization algorithms have to be initialized sufficiently close to the global minimum to avoid convergence towards local minimizers. In this work a suitable initial point is obtained through EDM completion.

2.2 Euclidean Distance Matrix Completion

A partial pre-distance matrix C is a matrix with zero diagonal entries and with certain elements fixed to given nonnegative values; the remaining elements are considered free. In this particular setup the fixed elements are the squared observed distances, $C_{ij} = d_{ij}^2$. The nearest EDM problem is to find an EDM that is nearest in the Frobenius norm to matrix C, when the free variables are not considered. The geometry and properties of EDM (a convex cone) have been extensively studied in the literature [8], [9]. The nearest Euclidean distance matrix problem is formulated as

$$\begin{aligned} &minimize \quad \|W \odot (C - D)\|_F^2\,, \\ &subject\ to \quad D \in EDM\ Cone \end{aligned} \tag{2}$$

where W is a mask matrix with zeros in the entries corresponding to free elements of the pre-distance matrix C, and ones elsewhere, and $\odot$ denotes the Hadamard product. Problem (2) is equivalent to a semidefinite program (SDP), which can be solved by standard convex optimization software.

2.3 Estimation of Sensors and Target Positions

Define a matrix Y whose columns hold all sensor, anchor and target coordinates, globally translated so that their average is located at the origin. Then the Gram matrix Y^TY can be obtained from the EDM matrix D by a linear transformation [8, Sec. 8.3], from which spatial coordinates Y are extracted by singular value decomposition (SVD) up to a unitary matrix. In most cases the SVD will return a coordinate matrix whose rank is greater than the embedding dimension (2, in this work), so valid coordinates are obtained by truncating the SVD to the appropriate rank.

Anchors are used to estimate the residual unitary matrix Q after SVD by solving the Procrustes problem, [10]

$$\begin{aligned} &minimize \quad \|A - QY_A\|_F^2\,, \\ &subject\ to \quad Q^TQ = I \end{aligned}$$

where the columns of A hold the anchor positions, and Y_A denotes the relevant subset of the columns of the truncated SVD output Y. This problem has a closed-form solution.

Observation noise can significantly disrupt the estimated sensor/target coordinates through EDM completion and rank truncation, and it was found that much more accurate results are obtained by using those as a starting point for likelihood maximization. We propose to iteratively minimize the cost function (1) using the MM and Newton methods.

3. METHOD I: MAJORIZATION-MINIMIZATION

The key idea of MM is to find, at a certain point x^t, a simpler function that has the same function value at x^t and anywhere else is larger than or equal to the objective function to be minimized. Such a function is called a majorization function. By minimizing the majorization function we obtain the next point of the algorithm, which also decreases the cost function [11]. Define

$$f_{ij}(x) = \|x_i - e_j\| \text{ and } g_{kj}(x) = \|a_k - e_j\|, \tag{3}$$

and assume that sensors and targets are not at the same positions, i.e., $x_i \neq e_j$, $a_k \neq e_j$. Expand (1) as

$$\theta(x) = \sum_{i,j}(f_{ij}^2(x) - 2d_{ij}f_{ij}(x) + d_{ij}^2) + \sum_{k,j}(g_{kj}^2(x) - 2d_{kj}g_{kj}(x) + d_{kj}^2)\,.$$

Since f and g are convex functions there holds

$$f_{ij}(x) \geq f_{ij}(x^t) + \nabla^T f_{ij}(x^t)(x - x^t),$$

hence,

$$\theta(x) \leq \sum_{i,j}(f_{ij}^2(x) - 2d_{ij}f_{ij}(x^t) - 2d_{ij}\nabla^T f_{ij}(x^t)(x - x^t) + d_{ij}^2) + \sum_{k,j}(g_{kj}^2(x) - 2d_{kj}g_{kj}(x^t) - 2d_{kj}\nabla^T g_{kj}(x^t)(x - x^t) + d_{kj}^2). \tag{4}$$

Thus, the proposed majorization function on the right side of (4) is quadratic and easily minimized. The MM iteration is

$$x^{t+1} = argmin_x \sum_{i,j}\left(f_{ij}^2(x) - 2d_{ij}\nabla^T f_{ij}(x^t)x\right) + \sum_{k,j}\left(g_{kj}^2(x) - 2d_{kj}\nabla^T g_{kj}(x^t)x\right).$$

To find the solution, rewrite (3) as $f_{ij}(x) = \|M_{ij}x\|$, where the linear operator M_{ij} extracts from x the difference $x_i - e_j$.

The same process is applied to the function $g_{kj}(x) = \|a_k + N_j x\|$ with suitably defined N_j, yielding the gradients

$$\nabla f_{ij}(x) = \frac{{M_{ij}}^T M_{ij} x}{\|M_{ij} x\|} \text{ and } \nabla g_{kj}(x) = \frac{{N_j}^T (a_k + N_j x)}{\|a_k + N_j x\|}.$$

Therefore, the new point is obtained by solving the linear system:

$$\left[\sum_{i,j} M_{ij}^T M_{ij} + \sum_{k,j} N_j^T N_j\right] x^{t+1} = \sum_{i,j} d_{ij} \nabla f_{ij}(x^t) + \sum_{k,j} d_{kj} \nabla g_{kj}(x^t) - \sum_{k,j} a_k^T N_j.$$

4. METHOD II: NEWTON ALGORITHM

Newton's algorithm with an appropriate line search applied to a convex function f ensures $f(x^{t+1}) < f(x^t)$ except when x^t is an optimal point. A search direction at the point x is found from the gradient and Hessian of f as

$$\Delta x = -\nabla^2 f(x)^{-1} \nabla f(x).$$

The step length in that direction can be calculated using backtracking line search as follows. Start with $\delta = 1$ as a step length and check if Armijo's rule holds:

$$f(x + \delta\Delta x) \leq f(x) + \alpha\delta\nabla f(x)^T \Delta x,$$

where $\alpha \in (0, 0.5)$. If not, reduce δ by half and recheck. If it holds, then the next iterate is $x = x + \delta\Delta x$. Continue until the stopping criterion holds [8].

Using the output of EDM completion as the initial estimate to Newton's method is expected to start sufficiently close to the optimal point to benefit from the algorithm's well known asymptotic quadratic convergence. Again, the objective function is (1), written here with the notation defined in the previous section

$$\theta(x) = \textstyle\sum_{i,j}\left(\|M_{ij}x\| - d_{ij}\right)^2 + \sum_{k,j}\left(\|a_k + N_j x\| \ - d_{kj}\right)^2. \quad (5)$$

Since the function is not convex Newton's method may fail to work properly. Therefore, the search direction should always be tested to confirm that it is in fact a descent direction, i.e., $\nabla f(x)^T \Delta x < 0$. If not, the negative of the gradient is used instead as a search direction, $\Delta x = -\nabla f(x)$. This does not sacrifice the asymptotic speed of Newton's method and provides improved robustness in convergence.

The gradient of (5) is

$$\nabla\theta(x) = \textstyle\sum_{i,j} 2\left(\|M_{ij}x\| - d_{ij}\right) \frac{M_{ij}^T M_{ij} x}{\|M_{ij}x\|} + \sum_{k,j} 2\left(\|a_k + N_j x\| - d_{kj}\right) \frac{{N_j}^T (a_k + N_j x)}{\|a_k + N_j x\|}.$$

The Hessian is given at the bottom of this page.

5. TIME RECURSIVE POSITION ESTIMATION

Suppose a batch of observations has been processed, and a new target position y is to be estimated. We could repeat the previous approach by redefining the new batch as the old one concatenated with the new set of observations. However, this is computationally expensive due to the EDM step. Also, previous estimated positions would be ignored. Thus, to alleviate this, we propose a simple methodology to obtain a good initial point, which avoids the EDM step. It consists of fixing the previous positions at their estimated values and only estimating the new target position. More precisely, we minimize

$$\psi(y) = \sum_i \left(\|x_i - y\|^2 - d_i^2\right)^2 + \sum_k \left(\|a_k - y\|^2 - d_k^2\right)^2, \quad (6)$$

where, y is the new target position and d_i, d_k denote the Euclidean distances between the new target and the sensors/anchors. Note that in (6) we are trying to match *squared* distances. Squaring the range measurements results in non-zero mean noise (with non-Gaussian distribution) unless the noise variance is very small. However, minimizing (6) is easy, as it can be reformulated as a Trust Region Problem, [6]. After an optimal target position is obtained, we return to the cost function (1) and apply MM or Newton to refine all the estimates. This incremental or time recursive procedure can be applied to either new targets or sensors.

6. RESULTS AND COMPARISONS

6.1. Results of EDM Initialization for MM or NEWTON Based Refinement

The results will be demonstrated in two scenarios. The first scenario is a randomly generated one, which contains 22 unknown positioned sensors and 3 anchors. The batch is processed at a central node after 8 target positions are gathered. In other words, 30 positions are to be estimated in this batch. We tested the algorithms for the simulations with the error which is unbiased and disturbed by white Gaussian noise with a standard deviation of 15 cm for ranges between 0-4 m considering practical scenarios. In general, the localization performance degrades gracefully as noise increases.

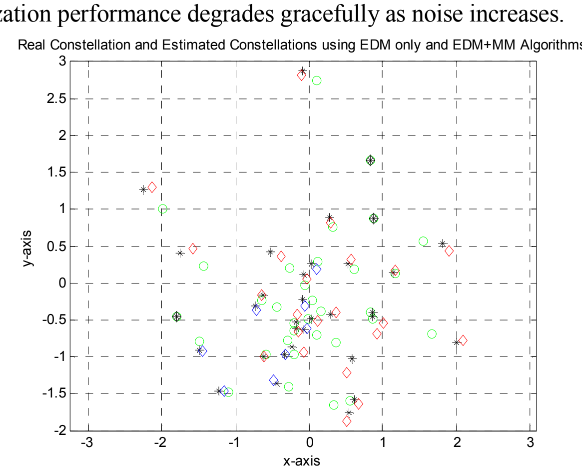

Figure 1 – The red diamonds are the real sensor positions, the blue diamonds are the real target positions and the blue stars are 3 anchor positions. The green circles are the estimated positions by EDM only, the black * are the estimated positions by EDM-MM.

$$\nabla^2\theta(x) = \sum_{i,j}\left[2\frac{\left(M_{ij}^T M_{ij} x x^T M_{ij}^T M_{ij}\right)}{\|M_{ij}x\|^2} + \frac{2\left(\|M_{ij}x\| - d_{ij}\right)}{\|M_{ij}x\|}\left\{M_{ij}^T M_{ij} - \frac{\left(M_{ij}^T M_{ij} x x^T M_{ij}^T M_{ij}\right)}{\|M_{ij}x\|^2}\right\}\right]$$
$$+ \sum_{k,j}\left[2\frac{\left({N_j}^T (a_k + N_j x)(a_k + N_j x)^T N_j\right)}{\|a_k + N_j x\|^2} + \frac{2\left(\|a_k + N_j x\| \ - d_{kj}\right)}{\|a_k + N_j x\|}\left\{{N_j}^T N_j - \frac{\left({N_j}^T (a_k + N_j x)(a_k + N_j x)^T N_j\right)}{\|a_k + N_j x\|^2}\right\}\right]$$

Fig. 1 shows the true constellation and estimated constellation using only EDM or EDM concatenated with MM Algorithms of the first scenario. The advantages of giving the EDM completion output as an initial starting point to MM or Newton Algorithms are obvious.

To verify the improvement of MM or NEWTON over plain EDM solutions, we compute the objective function in (1) at each iteration. In Fig. 2, which plots the objective function value versus the number of MM and Newton iterations, it is observed that after eight MM iterations or five Newton iterations, the value of $\theta(x)$ drops rapidly and significantly. This demonstrates that MM or Newton's methods improve the overall localization results.

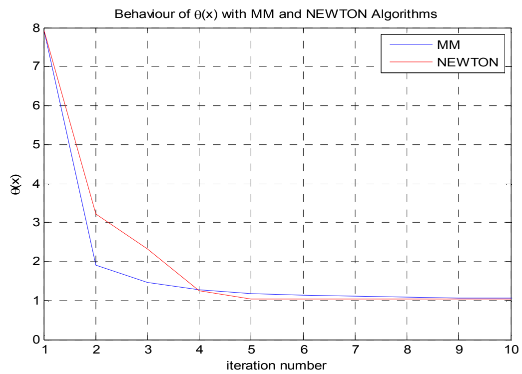

Figure 2 – $\theta(x)$ versus iteration number of MM and Newton.

We randomly generated a network of 20 sensors and 7 target positions to examine the accuracy of the methods. Monte Carlo simulation was used to find the mean and the variance of the positions estimated by EDM-MM for this sensor network. The accuracy of the EDM-MM method is clearly seen in Fig. 3.There is no uncertainty ellipsoid for the three anchor nodes, as their position are known.

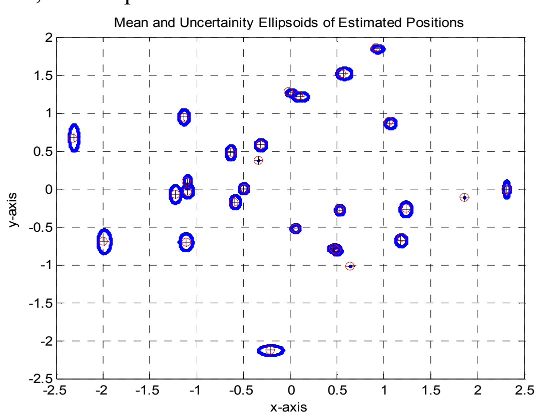

Figure 3 – Calculated mean and variance of estimated positions. The red circles are the real positions of sensors and target. The black + is the mean of the positions and the blue ellipsoids show the uncertainty around the mean values.

The second simulation scenario is created in such a way that most sensors are placed over two parallel lines, and the target moves along the middle line. Since measured distances are approximately the same along the trajectory for the real sensor and its mirror image with respect to the trajectory axis, ambiguities have more impact. Distinguishing and estimating the sensor and target positions becomes harder. For instance, in Fig. 4, sensors 1, 2 and 3 and their mirror images mostly obtain the same range measurements. Hence, at the end of the optimization algorithms estimated positions may come at reflected positions, which is the case in Fig. 4. Nevertheless, the advantage of giving the output of EDM completion to MM or Newton is more obvious in this scenario.

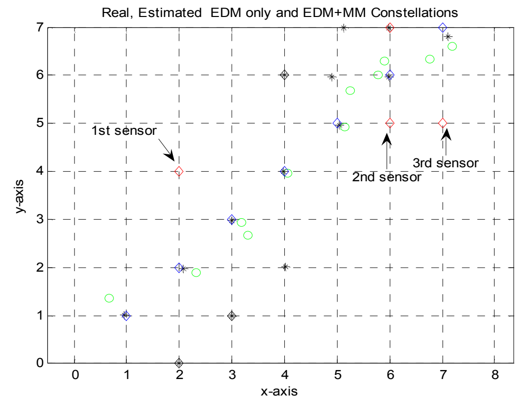

Figure 4 –The red diamonds are the real sensor positions, the blue diamonds are the real target positions and the blue stars are 3 anchor positions. The green circles are the estimated positions by EDM only, the black * are the estimated positions by EDM-MM.

Fig. 5 depicts the behavior of $\theta(x)$ for this scenario. Although the convergence rate of Newton is impressive, MM was found to be more robust.

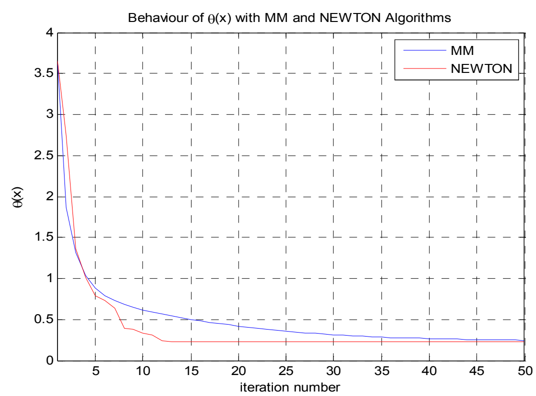

Figure 5– $\theta(x)$ versus iteration number of MM and Newton.

Both MM and Newton's method could not reach the optimal point for the function in (1) unless a good starting point is available. Thus, using the output of EDM completion as an initial estimate to the methods is crucial.

6.2. Results and Comparisons of Time Recursive Initialization for MM or Newton Based Refinement

A sensor network of 17 unknown positioned sensors, 3 anchor and 7 target positions are randomly generated to test the time recursive or incremental initialization algorithm.

24 positions are estimated in the first batch with EDM-MM method. Next, a new target range measurement is obtained by the sensors and a new position is estimated by fixing the previously estimated positions while minimizing (6). The newly estimated target position and all positions estimated in

the first batch are given as an initial point to start MM or Newton algorithms to further refine those positions. As a benchmark, EDM-MM is applied to the expanded batch with the same 24 positions plus the new target position. The behavior of $\theta(x)$ in these two approaches is shown in Fig. 6. The time recursive-MM approach takes advantage of previously estimated positions to start with a lower cost than EDM-MM but reaches the same final error value.

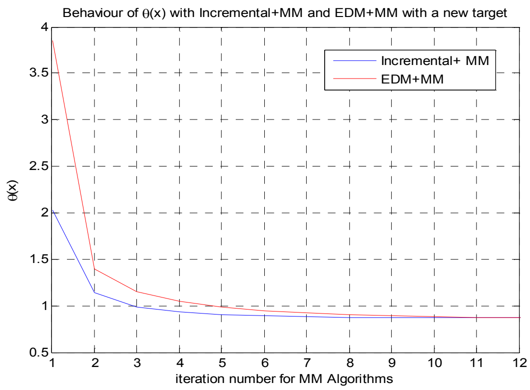

Figure 6– $\theta(x)$ versus iteration number of MM for EDM-MM and Incremental-MM approaches (1^{st} target position).

The two approaches are compared for 10 new target positions as well. Fig. 7 shows the behavior of $\theta(x)$ at the last incremental step, 10^{th} target position, of Incremental-MM method and of EDM-MM applied to the whole 10 new target positions. Incremental-MM processes each target position incrementally, which means it repeatedly estimates one target position by fixing the already estimated positions. However, EDM-MM makes a fresh start to the process without using the previous knowledge at every new position to be estimated, solving different and increasingly large EDM completion problems for ML initialization.

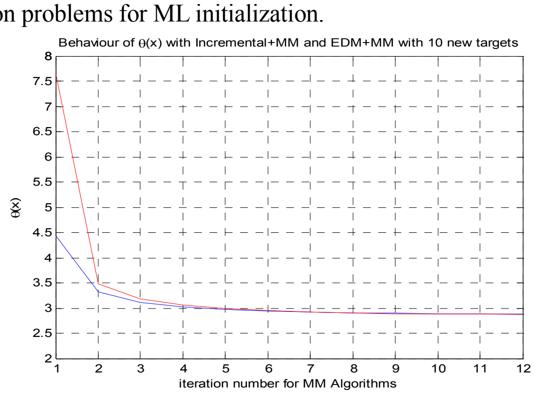

Figure 7– $\theta(x)$ versus iteration number of MM for EDM-MM and Incremental-MM approaches (10^{th} target position).

7. CONCLUSION AND FUTURE WORK

In this paper, we have presented a ML based technique to solve a *SLAT* problem. *MM* and *Newton* optimization methods are proposed to maximize the non-convex likelihood function, for which a good initialization is required. Therefore, we have investigated two initialization schemes, *batch approach* and *time-recursive approach*. After the first batch of measurements is obtained, *EDM completion* is used for the first initialization of the sensor network topology. However, EDM completion is not scalable, so we select a second initialization scheme for new positions. The *time recursive method* uses the already estimated positions at each time a new position is to be estimated; afterwards the newly estimated position and the already estimated ones are given as an initialization to the optimization methods. With this methodology, we guarantee a good initialization and a scalable solution for the SLAT problem. With these initialization schemes, simulation results show that both MM and Newton methods give accurate position estimates. In spite of Newton's faster convergence rate, the MM method appears to be more robust.

We used anchors to avoid some inherent ambiguities of the sensor/target localization problem using range measurements. As a future work, however, we will examine unambiguous formulations of SLAT on quotient spaces. Applying the optimization algorithms on those spaces might lead to more accurate solutions or algorithms with improved robustness. Another interesting topic would be to develop methods for incrementally combining estimates based on different blocks of range measurements taken along target trajectory.

REFERENCES

[1] C. Taylor, H. Shrobe, A. Rahimi, J. Bachrach, A. Grue, "Simultaneous Localization, Calibration, and Tracking in an ad Hoc Sensor Network," *Proc. Int. Conf. on Information Processing on Sensor Networks (IPSN'06)*, Nashville, USA, April 2006, pp. 27–33.

[2] S. Funiak, C. Guestrin, M. Paskin, R. Sukthankar, "Distributed Localization of Networked Cameras," *Proc. IPSN'06*, Nashville, USA, April 2006, pp. 34–42.

[3] R. Rangarajan, R. Raich, A. Hero III, "Euclidean Matrix Completion Problems in Tracking and Geo-localization," *Proceedings of ICASSP'08,* pp. 5324-5327, March 2008.

[4] Y. Ding, N. Krislock, J. Qian, H. Wolkowicz, "Sensor Network Localization, Euclidean Distance Matrix Completions, and Graph Realization," Research report CORR 2006-23, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ONN2L 3G1, Canada. [arXiv:math/0612388v1]

[5] P. Biswas, Tzu-Chen Liang, Kim-Chuan Toh, Yinyu Ye, "Semidefinite Programming Approaches for Sensor Network Localization with Noisy Distance Measureents," *IEEE Transactions on Automation Science and Engineering, IEEE,* vol. 3, pp. 360–371, Oct. 2006.

[6] A. Beck, P. Stoica, J. Li, "Exact and Approximate Solutions of Source Localization Problem," *IEEE Trans. on Signal Processing*, vol. 56, no. 5, pp. 1770-1778, May 2008.

[7] J. Costa, N. Patwari, A. Hero, "Distributed Weighted-Multidimensional Scaling for Node Localization in Sensor Networks,",*ACM Transactions on Sensor Networks*, vol. 2, no. 1, pp. 39-64, Feb. 2006.

[8] S.Boyd, L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[9] J. Dattorro, *Convex Optimization and Euclidean Distance Geometry*. Mεβoo publishers, 2005.

[10] G. Golub, C. Van Loan, *Matrix Computations*. The John Hopkins University Press, 1989.

[11] D, Hunter, K. Lange, "A Tutorial on MM Algorithms,"' *The American Statistician*, vol. 58, no. 1, pp. 30–37, Feb. 2004.