CONSTRAINED LOCALIZATION IN STATIC AND DYNAMIC SENSOR NETWORKS

By

Mona Mahmoudi

and

Guillermo Sapiro

IMA Preprint Series # 2091

(January 2006)

INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

UNIVERSITY OF MINNESOTA

400 Lind Hall
207 Church Street S.E.
Minneapolis, Minnesota 55455–0436

Phone: 612/624-6066 Fax: 612/626-7370

URL: http://www.ima.umn.edu

Constrained Localization in Static and Dynamic Sensor Networks

Mona Mahmoudi and Guillermo Sapiro

***Abstract*— In this note we propose to introduce physical constraints in the localization problem in sensor networks. This is based on extending the classical STRESS function from distance geometry and multidimensional scaling. We present the underlying framework and demonstrate its importance with three examples: penalizing the sensors for being in high elevation areas, removing sensors from forbidden areas, and forcing the sensors to be on pre-described curves. We also extend the work to dynamic environments.**

Index Terms - **Sensor localization, sensor networks, STRESS function, multidimensional scaling, distance geometry, physical constraints.**

I. Introduction

Automatic sensor localization is one of the most fundamental problems in the area of sensor networks. Sensor data needs to be registered to its physical location to be of use in the major applications of sensor networks. For large scale and inexpensive networks, it is not possible to include GPS capability on every device. Therefore, automatic sensor localization based on pairwise (local) information has received a lot of attention in recent years. The basic idea is to use information such as signal strength, time-of-arrival, or angle-of-arrival, between a sensor or group of sensors and some of their local neighbors (often denoted as pairwise or set-wise *dissimilarities*), to compute the physical coordinates of the sensors. See for example [4] for some literature on the subject and details on the basic requirements of sensor localization algorithms.

Computing point (sensor) coordinates from pairwise dissimilarities is a classical problem in distance geometry [2]. Such problems arise for example in molecular biology, where protein structures are to be determined from a few noisy measurements of pairwise distances obtained from X-ray crystallography or NMR. The same task is the fundamental problem in multidimensional scaling [1], [5], where the primary goal is to represent and visualize in low dimensional Euclidean space a set of pairwise dissimilarities obtained, for example, from psychophysical experiments. The sensor localization problem is nothing else than another application/extension of these theories.

Often, in addition to the pairwise dissimilarities, some prior or learned information about the physical environment is also available. For example, if we are localizing active cellular phones in the heart of the winter in Minnesota, it is very unlikely that they will be located in the middle of a lake with thin ice. Similarly, it is unlikely that sensors for ocean waves activity are located far inland. The same physical prior knowledge is valid for chemical and control sensors for example. This physical information is very common and imposes an additional constraint in the sensor localization problem. Although multidimensional scaling has studied the constrained scenario [1], [5], this important problem has not been part of the sensor localization techniques developed in the literature. It is the goal of this paper to present a simple framework for sensor localization with physical constraints and to show the importance of this type of constraint. Both static and dynamic environments are studied.

Next, Section II, describes the framework for sensor localization with constraints. Experimental results for particular physical scenarios are presented in Section III, while concluding remarks are given in Section IV.

II. Constrained sensor localization

We now introduce the proposed framework for constrained node localization in sensor networks. We first start with the static scenario, which is easily extended to the dynamic case. Consider a network with N nodes in a D dimensional space (usually $D = 2, 3$). Let $x_i \in \Re^D$, $i = 1..N$, be the coordinates for each one of the sensors. We assume that we measure pairwise dissimilarities δ_{ij} between sensors i and j at positions x_i and x_j as Euclidean distances:[1]

$$\delta_{ij} = \| x_i - x_j \| = \sqrt{(x_i - x_j)^T (x_i - x_j)}.$$

These pairwise distances can be obtained for example via received signal strength or time of arrival, and are often noisy. Also, not all the pairwise distances need to be available, often only close-by sensors are considered available.

We propose to find the constrained sensor positions from the available set δ_{ij} via the minimization of the following global cost function:

$$S = \sum_{ij} w_{ij} (\delta_{ij} - d_{ij}(X))^2 + \lambda \sum_i f(x_i). \qquad (1)$$

Here, w_{ij} represents the accuracy of the measurements δ_{ij} (e.g., $w_{ij} = 0$ is a measurement between sensors i and j is

We thank Alejandro Ribeiro for comments on the work here described. The work here presented is partially supported by the Office of Naval Research, the National Science Foundation, DARPA, and the National Geospatial-Intelligence Agency.

Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455, {mahmo022,guille}@umn.edu

[1] The framework introduced here, as well as the general theory of multidimensional scaling, is applicable to other measures of dissimilarities, including geodesic distances for example [11].

not available), X stands for the $N \times D$ matrix of unknown coordinates, $d_{ij}(X)$ is the Euclidean distance between the searched coordinates for the sensors i and j, $f(x_i)$ provides the penalty (constraint) for positioning sensor i at coordinate x_i, and λ is a scalar parameter that controls the tradeoff between the accuracy to the provided dissimilarities δ_{ij} and the constraints given by $f(\cdot)$. The first sum is over all the pairs, while the second one is over each sensor.

The first term in the energy (1) is the classical STRESS in multidimensional scaling [1], [5], and has been previously used for sensor localization, see [4] and references therein. The novelty in our approach is in the introduction of the second term, $\sum_i f(x_i)$. The penalty function f can represent probability of finding sensor i at a given location, or can penalize for locating the sensors in forbidden or unreasonable areas. In other words, it provides constraints on the sensor localization that come from prior or learned knowledge about the sensors network physical environment.[2]

Note that all practical algorithms for sensor localization that follow energy functions of the form of the STRESS cost are prone to local minima. Distance functions, the dissimilarities δ_{ij}, are also affected by the environment and are usually noisy. As a consequence, it is likely they will locate the sensors in low probability or even forbidden areas. This can be even more severe when the localization solution is not unique due to the noise or the lack of sufficient pairwise dissimilarities δ_{ij}. Some of these multiple solutions can be eliminated with the physical constraint. Thereby, using the cost function (1), even as a second refinement step, is crucial. This will be illustrated below.

A. Constrained sensor localization in dynamic environments

The above mentioned framework can be readily extended to dynamic sensor network environments, that is, when the dissimilarity measurements are available dynamically, we have time sampled dissimilarities δ_{ij}^t. The dynamic STRESS then becomes

$$S = \sum_t \sum_{ij} w_{ij}^t (\delta_{ij}^t - d_{ij}(X^t))^2, \tag{2}$$

and the goal is to locate the sensors X at all the time frames. Different types of constraints can be added. For example, we could consider a smoothness constraint, where we penalize for sensors moving from frame to frame:

$$S = \sum_t \sum_{ij} w_{ij}^t (\delta_{ij}^t - d_{ij}(X^t))^2 + \lambda \sum_t \sum_i \parallel x_i^{t+1} - x_i^t \parallel^2 . \tag{3}$$

λ can depend on the expected velocity. Position constraints as in Equation (1) could be added as well of course.

If the sensor x_i is attached for example to a moving vehicle, for which a rough estimate of the motion $\vec{V}_i$ is available, then we could think of an energy of the form

$$S = \sum_t \sum_{ij} w_{ij}^t (\delta_{ij}^t - d_{ij}(X^t))^2 + \lambda \sum_t \sum_i \parallel (x_i^{t+1} - x_i^t) - \vec{V}_i^t \parallel^2 . \tag{4}$$

The above formulations assume that the localization is simultaneously solved for all the frames, or at least for a block of them. We could also of course think of a "real time" scenario, where the sensors are located for frame t based on the location in previous frames and the current constraint. The formulas are equivalent, simply eliminating the summation over t.

III. Experimental results

For the examples in this paper, we find it sufficient to use standard optimization techniques from the Matlab Optimization and Statistics Toolboxes, e.g., *fminsearch* and *mdscale* [9], as well as standard majorization as commonly applied to the STRESS function when possible (in particular, when the constraint is quadratic), [4], [5]. While more sophisticated optimization techniques could be used as well, this is beyond the scope of this paper, and efficient techniques, extending for example the recent works reported in [3], [4], [7], [10], as well as those with constraints in [1], [5], [6], should be developed depending on the particular constraint used. When the constrained formulation here proposed is used as a second step following an advanced non-constrained optimization algorithm, see below for an example, the particular optimization is less relevant if a good initial condition has been achieved (using the mentioned advanced optimization approaches for un-constrained penalties). This is also true in the dynamic case, when the localization in time t is used as initial condition to the optimization for $t + 1$. Finally, note that the local cost decomposition for the STRESS developed in [4] is also valid for the constrained STRESS here introduce, e.g., equations (1)-(4).

We first present examples of our proposed constrained sensor localization framework in a static environment. In particular, three different selections for the penalty function f in Equation (1) are presented. First, in Figure 1 we localize the sensors following a topographic map. We define f to act as a probability function, with probability proportional to the local elevation, the higher the point the lower the probability to find a sensor there. In Figure 2, we penalize for sensors located inside the two large lake areas. This is done by designing f to be strictly positive inside the forbidden areas (lake) and zero otherwise. Moreover, f increases with the distance to the border of the lake. Finally, in Figure 3, we constrain the sensors to be on a given curve (the yellow line). This is done by defining f to be the unsigned distance to the curve. In all these examples we start from a given configuration, blue squares for Figure 1 and green squares for figures 2 and 3, that can be obtained from any of the current state-of-the-art and unconstrained sensor localization techniques, e.g., [4], [7], [10]. Then, we minimize the energy given by (1), obtaining the red dots as the new and now constrained sensors position.

In Figure 4 we explicitly simulate the use of our framework as a "correction" step to un-constrained approaches, also

[2]Costa *et al.*, [4], proposed an extra quadratic term in their modified STRESS function that penalizes for individual sensors i to be located far from pre-established positions $\bar{x}_i$: $\sum_i r_i \parallel x_i - \bar{x}_i \parallel$, for confidence values r_i. This is a particular case of the physically motivated constrained framework here proposed. Energy based penalization in sensor networks are also studied in [8], the particular penalization coming from acoustic models, and is not a result of the general physical framework here proposed.

illustrating the importance of having constraints when the pairwise distances are noisy and incomplete, as in all realistic sensor network scenarios. Using the same penalty as in Figure 3, we first located sensors on the yellow curve, and then added noise to the pairwise distances and considered only the availability of distances between close-by sensors. This simulates a real scenario, with very conservative noise and missing data, see figure captions. The green dots show the result without constraint, using *mdscale*,[3] while the red ones are the ones after our constrained correction. Not only the sensors are correctly placed on the yellow line but also the overall error to the original positions is reduced (to 50%, 30%, and 50% respectively).

We now conclude with examples from dynamic environments, Figure 5, following in this case Equation (3). This is a quadratic constraint, and as in [4], is efficiently minimized via distributed classical majorization [5]. The use of the constraint not only improves accuracy, but also accelerates the minimization convergence. See mountains.ece.umn.edu/∼guille/Sensors for the video for these and other examples.

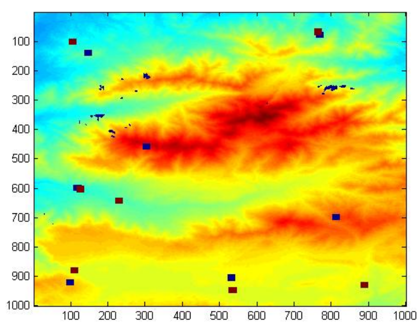

Fig. 1. *The elevation data is used to define the penalty function f. The higher the area, the lower the probability of finding a sensor there. Note how the blue squares, marking the original sensor locations, move to the red squares located at lower elevation regions. The elevation map goes from dark blue, representing low elevations, to dark red, representing high elevations. The blue spots are zones of holes in the elevation information.*

IV. Conclusions

In this note we have addressed for the first time the problem of static and dynamic sensor localization with physical constraints. We have extended the classical STRESS function from distance geometry and multidimensional scaling theories, which is frequently used in the sensor networks arena, to include an extra term that represents the available prior information about the physical environment. We exemplified the ideas by constraining the sensors to be in low elevation areas, outside of forbidden zones, and to be located on predefined curves. Dynamic environments were exemplified as well.

The proposed algorithm can be used as a second step, after constraint-free and efficient sensor localization techniques have been applied to the available pairwise dissimilarity measures. To efficiently use our proposed framework directly, distributed minimization algorithms have to be developed. For some constraints, these are already available in the multidimensional and distance geometry literature, while for others, such as those coming from distance functions, they need to be developed (based on these available algorithms). Extending the current work to angle-of-arrival information is of great significance as well. These directions, together with the use of our technique for large real sensor networks, are the subject of future research.

[3] To address the invariance to rotation/translation/symmetry, we use three anchor points.

References

[1] I. Borg and P. Gorenen, *Modern Multidimensional Scaling: Theory and Applications*, Springer, 1996.
[2] L. M. Blumenthal, *Theory and Applications of Distance Geometry*, Chelsea House Pub, 1970.
[3] M. M. Bronstein, A. M. Bronstein, R. Kimmel, and I. Yavneh, "Multigrid multidimensional scaling," *Numerical Linear Algebra with Applications*, 2005, to appear.
[4] J. Costa, N. Patwari, and A. O. Hero III, "Distributed multidimensional scaling with adaptive weighting for node localization in sensor networks," *ACM Trans. on Sensor Networks*, to appear, 2005.
[5] T. Cox and M. Cox, *Multidimensional Scaling*, Chapman & and Hall, London, 1994.
[6] W. J. Heiser and J. Meulman, "Constrained multidimensional scaling, including confirmation," *Applied Psychological Measurement* **7**, 381-404, 1983.
[7] A. T. Ihler, J. W. Fisher III, R. L. Moses, and A. S. Willsky, "Nonparametric belief propagation for self-calibration in sensor networks," *IEEE Journal of Selected Areas in Communication*, 2005.
[8] D. Li and Y. H. Hu, "Energy based collaborative souce localization using acoustic microsensor array," *EURASIP JASP* **2003:4**, pp. 321-337, 2003.
[9] *Matlab Optimization Toolbox*, http://www.mathworks.com/
[10] A. M-C. So and Y. Ye, "Theory of semidefinite programming for sensor network socalization," *Proc. ACM-SIAM Symposium on Discrete Algorithms*, 2005.
[11] J. B. Tenenbaum, V. DeSilva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science* **290**, pp. 2319-2323, 2000.

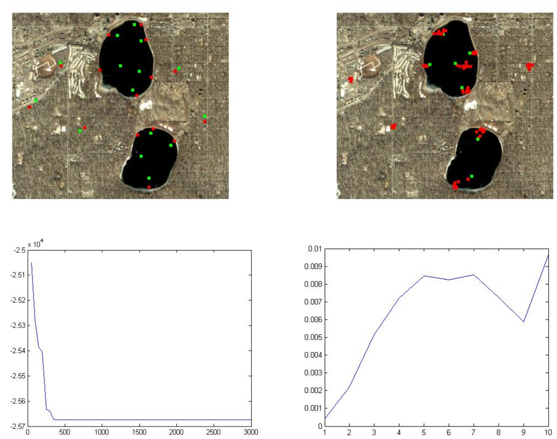

Fig. 2. *Examples of placing the sensors outside of the lake areas. On the top-left we see the results, where all the sensors have been moved out of the lake. Next we show the sensor positions for different values of the parameter λ in Equation (1). Next, second row, we plot the energy (1) as a function of the minimization iteration. Lastly, we plot the average percentage deviation of the pairwise distance function, with respect to the initial condition of the optimization, as a function of λ (this corresponds to the figure on the top right).*

Fig. 3. *Examples of placing the sensors in a curve, the yellow line. Results for two different values of λ are shown in the first figures, with a higher value on the right (note how all the sensors are located on the yellow line). A different distances configuration is shown in the third figure. In the last figure an additional example is shown, where some of the pairwise distances between the sensors are not available ($w_{ij} = 0$ in Equation (1)).*

Fig. 4. *Simulation of the use of our constrained algorithm to correct initial results obtained from other sensor localization techniques. Three examples are presented, with different sensor configurations and noise levels. Although the sensors are originally located on the yellow curve, a small amount of noise (less than 0.2% for the first two and 5% for the third), and a few of the largest pairwise distances missing (less than 30% for the first two and 42% for the third), representing a very optimistic sensor network scenarios, leads to failures in the localization, green dots. Using this as initial condition to our constrained optimization, brings the sensors to their legal position on the yellow curve, red dots. The overall position error is significantly reduced as well (about half of the error without constraints).*

Fig. 5. *Example of dynamic sensor localization. Three consecutive frames are shown on top. On the bottom we see how working without constraints (3 left frames), the localization (given by Xs) is far from the original positions (given by the squares), while very accurate localization is obtained when adding the constraint (3 right frames). Same number of iterations where used for both cases. See mountains.ece.umn.edu/$\sim$guille/Sensors for the video for these examples.*