SEMIDEFINITE PROGRAMMING AND THE SENSOR NETWORK LOCALIZATION PROBLEM, SNLP

CLAYTON W. COMMANDER, MICHELLE A. RAGLE, AND YINYU YE

1. SYNONYMS

The SENSOR NETWORK LOCALIZATION PROBLEM is a variant of the EUCLIDEAN DISTANCE MATRIX COMPLETION PROBLEM [2, 3] and the GRAPH REALIZATION PROBLEM.

2. INTRODUCTION

Research on ad hoc wireless sensor networks has increased greatly in recent years [24]. Sensor networks usually consist of a large number of sensors which are deployed to collect data of interest. Such networks are versatile tools which provide a low-cost method of target tracking, as well as monitoring seismic activity, temperature, sound levels, and light [7]. Information gathered by the sensors is only useful if the positions of the sensors are known. However, it is often the case that the use of a GPS system is too costly or, consumes too much power, or the network is being deployed in a location in which GPS is denied [20].

Recently, techniques have been developed which estimate the node locations based on a mixture of distance measurements and angle measurements between pairs of nodes in the network. This problem is referred to as the SENSOR NETWORK LOCALIZATION PROBLEM, (SNLP) and can be formally stated as follows: Given the true positions of some of the nodes and the pair-wise distances between some nodes, how can the positions of all of the nodes be estimated? [6, 7, 28].

2.1. **Organization.** Throughout the article, we will investigate the SNLP. In the following section, we formally define the problem statement and in Section 4, we review several solution techniques which appear throughout the literature. In Subsection 4.2, we describe a semidefinite programming (SDP) model for the problem. We then provide a general assessment of this approach and describe some implementation details. We highlight this method specifically because of its advantages over heuristic methods. Particularly, the SDP method is known to localize any network whenever a unique solution exists, and to do so in polynomial time. We provide some concluding remarks in Section 5 and indicate directions of future research. Finally, a list of cross references is provided in Section 6. We conclude this section with an introduction to some of the symbology that will appear most prevalently throughout the article.

2.2. **Idiosyncrasies.** Here we briefly introduce some of the symbols and notations we will employ throughout this paper. Define the trace of a symmetric matrix A, denoted $\mathrm{Trace}(A)$ as the sum of the diagonal entries. The standard trace inner product of two matrices A and B is given as $\langle A, B \rangle = \mathrm{Trace}(A^{\mathrm{T}}B)$. The 2-norm of a vector x is denoted as $||x||$ and is defined to be $\sqrt{\langle x, x \rangle}$. A positive semidefinite matrix A will be denoted as $A \succeq 0$. Agree to let I_d and $\mathbf{0}_d$ respectively represent the identity matrix and a vector containing all zeros, both with dimension $d \in \mathbb{Z}$. Finally, we will use *italics* for emphasis, $\mathcal{CALLIGRAPHY}$ to refer to formulations, and SMALL CAPS for problem names. Any other locally used terms and symbols will be defined in the sections in which they appear.

3. Formulation

A wireless sensor network is typically made up of a number of densely distributed sensors that collect data. An instance of the SNLP consists of a set of m so-called *anchor* points whose positions are known a priori [7, 11, 21]. The object is to determine the location of n *sensor* points in the system based upon information obtained from the anchor sensors. Let the anchor points and the sensor points be respectively denoted as $a_1, a_2, \ldots, a_m \in \mathbb{R}^d$ and $x_1, x_2, \ldots, x_n \in \mathbb{R}^d$. The Euclidean distances $\bar{d}_{kj}$ between points a_k and x_j for some k, j, and d_{ij} between x_i and x_j for some $i < j$ are also given. Let $N_a = \{(k, j) : \bar{d}_{k,j} \text{ is specified}\}$ denote the sensor/sensor pairs and $N_x = \{(i, j) : i < j, d_{ij} \text{ is specified}\}$ represent the sensor/anchor. Then the SENSOR NETWORK LOCALIZATION PROBLEM as defined in [7] is to find the localization (estimated position) of $x_i, x_2, \ldots, x_n \in \mathbb{R}^d$ such that:

$$\mathcal{SNLP}: \quad \|a_k - x_j\|^2 \quad = \quad \bar{d}_{kj}^2, \ \forall \, (k, j) \in N_a, \text{ and}$$

$$\|x_i - x_j\|^2 \quad = \quad d_{ij}^2, \ \forall \, (i, j) \in N_x. \tag{1}$$

From this seemingly simple formulation, many difficult questions arise. For a given instance of the SNLP, does this instance have a realization in the required dimension? If so, is the realization unique? We should note that these two seemingly related questions are quite different from a computational perspective. It has been shown that determining if an instance of the SNLP has a unique realization in $\mathbb{R}^2$ can be determined efficiently under certain assumptions [19]. On the other hand, it remains $\mathcal{NP}$-complete [16] to compute a realization on the plane, even if the instance is *known* to have a unique realization [5]. This is the main problem of interest in this article.

4. Methods

In this section, we review several solution methods which have been applied to the SNLP. Particularly in Subsection 4.2, we highlight the techniques of Ye et al. [1, 7, 8, 28] and the application of semidefinite programming methods for efficiently computing solutions to large-scale instances of the SNLP under a variety of circumstances.

4.1. **Review of Solution Approaches.** Several techniques have been applied the the SNLP, all having some redeeming qualities [7, 15, 17]. Several techniques involve the use of distance or angle measurements between the anchor points in order to compute a localization [12, 14, 23, 25, 26, 27]. Another common technique used by Bulusu et al. [10] and Howard et al. [18] is to employ a grid or a set of surveyed points whose locations are known. Then, the sensor localization is attempted using the relative distances between sensors and the set of *beacon* points [7].

The so-called "DV-Hop" technique of Niculescu and Nash [23] is an efficient method in dense topologies whereby the anchor nodes flood the network with their location information. This allows other points to *triangulate* their positions based on the information of the anchor nodes. This information is then passed along to other sensor nodes who use the combined locations to triangulate their positions. However, for widely dispersed and irregular topologies, the relative errors in the node estimation tends to be fairly substantial.

A similar technique proposed by Savarese et al. [25] uses a method similar to the "DV-Hop" algorithm described above to provide a rough estimate of the location information. These estimates are then improved by applying a least-squares triangulation using these estimates as well as a new collection of estimated positions [7].

The "iterative multilateration" technique of Savvides et al. [26] is another effective method especially when the number of anchor nodes is relatively high. This method calculates via triangulation the positions of those nodes that are adjacent to at least three anchor

points. Then these new localized nodes become anchor points and the process continues. In the end every node in the network has become an anchor.

In [14], the so-called "mutlidimensional scaling" algorithm is proposed. The heuristic begins by making an initial estimate of the node positions based solely on the connectivity and basic distance and angle information of the *non-anchor* nodes. Then using a variant of singular value decomposition [29], a map is generated of the relative locations of the nodes. Finally, these estimates are greatly improved and an absolute global map is produced by taking into account the locations of the anchor nodes and updating the estimates accordingly.

The work of Doherty et al. [12] involves a technique in which linear bounding hyperplanes are used to model the proximity constraints on the nodes which can communication with each other as convex constraints. [1, 7]. However, these constraints are often too loose and provide solutions which are not helpful in terms of calculating the unique realization of the sensors.

As we see, the drawback with most sensor network localization techniques involving heuristics is that they do not always find a unique solution even when it exists, or require excessive computation time to do so [28]. A recently developed method introduced by So and Ye in [28] uses a semidefinite programming (SDP) model that guarantees the discovery of a unique solution when it exists. Furthermore, the solution can be computed in polynomial time. In the following subsection, we present the SDP model, discuss the motivation behind using this approach and analyze some properties of the model.

4.2. **Semidefinite Programming Model.** A semidefinite program is a convex optimization problem where the objective function is linear and the constraint is defined by a linear matrix inequality. Given a vector $c \in \mathbb{R}^m$, and $m+1$ symmetric matrices $F_0, F_1, \ldots, F_m \in \mathbb{R}^{nxn}$, a semidefinite program can be written in the form:

$$\min_{x \in \mathbb{R}^m} \{c^T x | F(x) \succeq 0\},$$

where $F(x) = F_0 + \sum_{i=1}^m x_i F_i$, and $F(x) \succeq 0$ implies that $F(x)$ is positive semidefinite [13, 30]. Hence, both the objective function and the constraint are convex, and therefore semidefinite programs are closely related to linear programs, and many algorithms for solving linear programming problems have generalizations that apply to semidefinite programs as well [30].

In [28], the authors note that (1) is a non-convex optimization problem, which is difficult to solve in general. They propose a SDP relaxation by converting the nonconvex quadratic distance constraints into linear constraints as follows. Specifically, let $X = [x_1, x_2, \ldots, x_n]$ be the $d \times n$ matrix which we are are trying to determine. Let $e_{ij} \in \mathbb{R}$ be the vector where the i-th position is 1, the j-th position is -1, and all other entries are zeros. Then for all $(i, j) \in N_x$, we have that:

$$||x_i - x_j||^2 = e_{ij}^T X^T X e_{ij}. \tag{2}$$

Furthermore, for all $(k, j) \in N_a$ it follows that

$$||a_k - x_j||^2 = (a_k; e_j)^T [I_d; X]^T [I_d; X](a_k; e_j), \tag{3}$$

where e_j is a vector of all zeros except for -1 at entry j, and $(a_k; e_j) \in \mathbb{R}^{d+n}$ is a vector consisting of a_k "on top of" e_j [7]. Using these definitions, we can reformulate the problem

as follows.

$$\text{Find } X \in \mathbb{R}^{d \times n} \text{ and } Y \in \mathbb{R}^{n \times n} \tag{4}$$

such that

$$e_{ij}^{\mathrm{T}} Y e_{ij} = d_{ij}^2, \ \forall \, (i,j) \in N_x, \tag{5}$$

$$(a_k; e_j)^{\mathrm{T}} \begin{pmatrix} I_d & X \\ X^{\mathrm{T}} & Y \end{pmatrix} (a_k; e_j) = \bar{d}_{kj}^2, \ \forall \, (k,j) \in N_a, \tag{6}$$

$$Y = X^{\mathrm{T}} X. \tag{7}$$

The intuition behind the SDP formulation is to relax constraint (7) to $Y \succeq X^{\mathrm{T}} X$; thus implying that $Y - X^{\mathrm{T}} X$ is positive semidefinite. Boyd et al. [9] among others have shown that a positive semidefinite matrix $Y - X^{\mathrm{T}} X$ can be expressed as

$$Z = \begin{pmatrix} I_d & X \\ X^{\mathrm{T}} & Y \end{pmatrix} \succeq 0. \tag{8}$$

Define $Z_{1:d,1:d}$ to be the $d \times d$ principle submatrix of Z. Then the SDP relaxation of the SNLP as given in [28], is to find $Z \in \mathbb{R}^{(d+n) \times (d+n)}$ to:

$$\mathcal{SDP} : \qquad\qquad \text{maximize} \quad 0 \tag{9}$$

subject to

$$Z_{1:d,1:d} = I_d, \tag{10}$$

$$\langle (\mathbf{0}; e_{ij})(\mathbf{0}; e_{ij})^{\mathrm{T}}, Z \rangle = d_{ij}^2, \ \forall \, (i,j) \in N_x, \tag{11}$$

$$\langle (a_k; e_j)(a_k; e_j)^{\mathrm{T}}, Z \rangle = \bar{d}_{kj}^2, \ \forall \, (k,j) \in N_a, \tag{12}$$

$$Z \succeq 0. \tag{13}$$

Notice that by definition, any feasible solution matrix Z must have at least rank d [7]. We can formulate the dual of the SDP relaxation as

$$\mathcal{SDP}\text{-}\mathcal{D} : \text{minimize} \quad \langle I_d, V \rangle + \sum_{(i,j) \in N_x} y_{ij} d_{ij}^2 + \sum_{(k,j) \in N_a} w_{kj} \bar{d}_{kj}^2 \tag{14}$$

subject to

$$\tag{15}$$

$$\begin{pmatrix} V & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} + \sum_{(i,j) \in N_x} y_{ij} (\mathbf{0}; e_{ij})(\mathbf{0}; e_{ij})^{\mathrm{T}} + \sum_{(k,j) \in N_a} w_{kj} (a_k; e_j)(a_k; e_j)^{\mathrm{T}} \succeq 0.$$

Notice that the dual formulation is always feasible. In particular having $V = 0$, $y_{ij} = 0$ for all $(i,j) \in N_x$, and $w_{kj} = 0$ for all $(k,j) \in N_a$ forms a feasible solution.

In [28], So and Ye postulate and prove several results regarding the above formulations. We will highlight the key theorems and provide a basic analysis. For detailed proofs and a more in-depth study, see [28].

The first result provides a class of instances for which the SDP relaxation is exact, i.e. for instances when the matrix Z has rank d. Suppose that formulation $\mathcal{SDP}$ is feasible. This implies that the distance measurements d_{ij} and $\bar{d}_{kj}$ are exact for the positions $\bar{X} = [\bar{x}_1, \ldots \bar{x}_n]$. Then, we have the following result.

Theorem 1. *Let $\bar{Z}$ be a feasible solution for $\mathcal{SDP}$ and $\bar{U}$ be an optimal slack matrix of $\mathcal{SDP}\text{-}\mathcal{D}$. Then by the duality theorem for semidefinite programming* [4]*, it follows that:*

(1) $\langle \bar{Z}, \bar{U} \rangle = 0$;
(2) $\mathrm{rank}(\bar{Z}) + \mathrm{rank}(\bar{U}) \leq d + n$;
(3) $\mathrm{rank}(\bar{Z}) \geq d$ *and* $\mathrm{rank}(\bar{U}) \leq n$.

A immediate consequence of this theorem is that for optimal dual slack matrices $\bar{U}$ such that $\mathrm{rank}(\bar{U}) = n$, it follows that $\mathrm{rank}(\bar{Z}) = d$. Therefore, formulation $\mathcal{SNLP}$

is equivalent to formulation $\mathcal{SDP}$ implying that the $\mathcal{SNLP}$ formulation can be solved optimally in polynomial time [28].

The next theorem establishes the existence of a large group of efficiently localizable graphs.

Theorem 2. *Suppose that the network in question is connected. Then the following are equivalent:*

(1) *Problem $\mathcal{SNLP}$ is uniquely localizable.*
(2) *The max-rank solution matrix of $\mathcal{SDP}$ has rank d.*
(3) *The solution matrix of $\mathcal{SDP}$, represented by (8), satisfies $Y = X^T X$.*

This theorem has several significant implications. First, we have that as long as $\mathcal{SNLP}$ has a unique localization, then it can be computed in polynomial time by solving the corresponding semidefinite relaxation. The converse also holds. That is, if the solution matrix to the semidefinite relaxation X has rank d, then X is the unique localization for formulation $\mathcal{SNLP}$ [28]. Lastly, as we mentioned above we have the existence of a family of graphs for which the localization can be efficiently computed despite the underlying $\mathcal{NP}$-completeness of the SNLP in general.

The seminal work of So and Ye [28] which we highlighted above provides a baseline to which many extensions can be made. To be with, the results presented above are based on the assumption that the distance measurements are exact. The work of Biswas et al. in [7] provides extensions to handle inaccurate and incomplete measurements. This greatly improves the robustness of the SDP formulation, making the model more applicable to real-world scenarios in which inaccuracies are inevitable. Furthermore, in [1] the authors provide SDP formulations of the SNLP which incorporate angle information which can be used alone or in concert with distance information to calculate sensor realizations. This method is particularly useful when the sensors can detect multiple angles [1]. We see that many extensions are possible and that the by using semidefinite programming methods, a large class of SENSOR NETWORK LOCALIZATION PROBLEMS are able to be solved more efficiently and effectively than by previous heuristic techniques.

5. CONCLUSION

The focus of this article was the SENSOR NETWORK LOCALIZATION PROBLEM (SNLP), with particular attention given to a set of robust solution technique based on a semidefinite programming model. After an introduction to the problem, we highlighted several solution approaches which have been applied. Next, we presented an analyzed the SDP formulation of So and Ye [28]. The results proved for the SDP relaxation of the SNLP have provided a framework which can be extended to other problems in distance geometry in which angle and distance information are mutual between pairs of points. Such problems include Euclidean ball packing and most recently 3-dimensional molecule conformation problems [22].

6. CROSS REFERENCES

See also: **Graph realization via semidefinite programming; Semidefinite programming and determinant maximization; Semidefinite programming and structural optimization; Semidefinite programming; Optimality conditions and stability; Solving large scale and sparse semidefinite programs.**

REFERENCES

[1] H. Aghajan, P. Biswas, and Y. Ye. Semidefinite programming algorithms for sensor network localization using angle of arrival information. In *39th Annual Asilomar Conference on Signals, Systems and Computers*, 2005.

[2] A.Y. Alfakih and H. Wolkowicz. On the embeddability of weighted graphs in Euclidean spaces. Technical Report CORR 98-12, University of Waterloo, Dept. of Combinatorics and Optimization, 1998.

[3] A.Y. Alfakih and H. Wolkowicz. Euclidean distance matrices and the molecular conformation problem. Technical Report CORR 2002-17, University of Waterloo, Dept. of Combinatorics and Optimization, 2002.

[4] F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5:13–51, 1995.

[5] J. Aspnes, D. Goldenberg, and R. Yang. On the computational complexity of sensor network localization. In *ALGOSENSORS 2004, LCNS 3121*, pages 32–44. 2004.

[6] J. Beutel. Geolocation in a picoradio environment. M.S. thesis, ETH Zurich, Switzerland/UC Berkeley, 1999.

[7] P. Biswas, T. Lian, T. Wang, and Y. Ye. Semidefinite programming based algorithms sensor network localization. *ACM Transactions on Sensor Networks*, 2(2):188–220, 2006.

[8] P. Biswas, T. Liang, K. Toh, Y. Ye, and T. Wang. Semidefinite programming approaches for sensor network localization with noisy distance measurements. *IEEE Transactions on Automation Science and Engineering*, 3(4):360–371, 2006.

[9] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, 1994.

[10] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. Technical report, Computer Science Department, University of Southern California, April 2000.

[11] C. Chong and S.P. Kumar. Sensor networks: Evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.

[12] L. Doherty, L.E. Ghaoui, and S.J. Pister. Convex position estimation in wireless sensor networks. *IEEE Infocom*, 3:1655–1663, 2001.

[13] M.K.H. Fan and Y. Gong. Eingenvalue multiplicity estimate in semidefinite programming. *Journal of Optimization Theory and Applications*, 94(1):55–72, 1997.

[14] M. Fromherz, W. Ruml, Y. Shang, and Y. Zhang. Localization from connectivity in sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 15(11):961–974, 2004.

[15] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker. An empirical study of epidemic algorithms in large scale multihop wireless networks. Technical report, University of California, Los Angeles, 2002.

[16] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979.

[17] J. Hightower and G. Borriello. Location systems for upbiquitous computing. *Computer*, 34:57–66, 2001.

[18] A. Howard, M. Matarić, and G. Sukhatme. Relaxation on a mesh: a formalism for generalized localization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1055–1060, Wailea, Hawaii, 2001.

[19] B. Jackson and T. Jordán. Connected rigidity matroids and unique realizations of graphs. *Journal of Combinatorial Theory, Series B*, 94:1–29, 2005.

[20] E. Kaplan. *Understanding GPS: Principles and Applications*. Artech House, Norwood, MA, 1996.

[21] K. Langendoen and N. Reijers. Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks*, 43:499–518, 2003.

[22] P. Biswas T. Liang, K. Toh, and Y. Ye. An SDP based approach for anchor-free 3d graph realization. *SIAM Journal on Scientific Computing*, submitted, 2005.

[23] D. Niculescu and B. Nath. Ad hoc positioning system (aps). *IEEE GLOBECOM*, 1:2926–2931, 2001.

[24] M.G.C. Resende and P.M. Pardalos. *Handbook of Optimization in Telecommunications*. Springer, 2006.

[25] C. Savarese, J. Rabay, and K. Langendoen. Robust positioning algorithms for distributed ad-hoc wireless sensor networks. In *USENIX Technical Annual Conference*, 2002.

[26] A. Savvides, C.-C. Han, and M.B. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Mobile Computing and Networking*, 166-179 2001.

[27] A. Savvides, H. Park, and M.B. Srivastava. The bits and flops of the n-hop multilateration primitive for node localization problems. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, pages 112–121, 2002.

[28] A. So and Y. Ye. Theory of semidefinite programming for sensor network localization. *Mathematical Programming*, published online, DOI: 10.1007/s10107-006-0040-1, 2006.

[29] G. Strang. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, third edition, 1998.

[30] L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38:49–95, 1996.

(C.W. COMMANDER) AIR FORCE RESEARCH LABORATORY, MUNITIONS DIRECTORATE, AND, DEPT. OF INDUSTRIAL AND SYSTEMS ENGINEERING, UNIVERSITY OF FLORIDA, GAINESVILLE, FL USA.
  E-mail address: clayton.commander@eglin.af.mil

(M.A. RAGLE) CENTER FOR APPLIED OPTIMIZATION, DEPT. OF INDUSTRIAL AND SYSTEMS ENGINEERING, UNIVERSITY OF FLORIDA, GAINESVILLE, FL USA.
  E-mail address: raglem@ufl.edu

(Y. YE) DEPT. OF MANAGEMENT SCIENCE AND ENGINEERING AND, BY COURTESY, ELECTRICAL ENGINEERING, STANFORD UNIVERSITY, STANFORD, CA USA.

E-mail address: yinyu-ye@stanford.edu