Proceedings of the 1st IFAC Workshop on Estimation and Control of Networked Systems, Venice, Italy, September 24-26, 2009

On the Use of Convex Optimization in Sensor Network Localization and Synchronization

Iman Shames, Brian D. O. Anderson, Barış Fidan

* *The Australian National University and National ICT Australia, Canberra, Australia*
Email: {Iman.Shames,Brian.Anderson,Baris.Fidan}@anu.edu.au.

Abstract: In this paper we report some new results obtained in the field of multi-agent systems that are based on convex optimization. First, we provide review of a set of polynomial function optimization tools including sum of squares (SOS) and semidefinite programming (SDP). Then we present several applications of these tools in various multiagent system localization and synchronization tasks. As the first application, we propose a method based on SOS relaxation for agent localization using noisy measurements and describe the solution through SDP. Later, we apply this method to address the problems of cooperative target localization in the presence of noise and robot pose determination based on range measurements. Then we introduce the problem of anchor selection for minimizing the effect of noise in sensor networks via SDP. We use the same machinery to propose a method based on SDP to enhance synchronizability in networks. We do so by proposing a distributed algorithm for adding new edges to the network to enhance synchronizability. Finally, we present a method to identify the node in a network loss of which inflicts the most damage on the synchronizability of the network. Conclusions are presented in the last section.

Keywords: Coordinated control and estimation over networks; Decentralized and cooperative optimization

1. INTRODUCTION

The application of methods based on convex optimization and in particular semidefinite programming (SDP) in localization of sensor networks in the presence of noisy measurements is well known, see e.g. Carter et al. (2006), Biswas et al. (2006), Biswas and Ye (2004), Ding et al. (2008). For a complete survey the reader may refer to Ye (2006). Although recently the use of SDP for other applications such as tracking is being recognized as well, e.g. Wang and Ding (2008), the main focus has been on the localization of nodes in the sensor network using SDP relaxations, and other possible applications of SDP have not been fully explored yet.

In this paper we report different results related to sensor networks where optimization tools based on sum of squares (SOS), SDP formulation and relaxation have been applied. First, we revisit the problem of localization of agents in sensor networks and propose a solution using the tools from algebraic geometry and specifically sum of squares relaxation: this relaxation requires solving a SDP problem. Sum of squares has been already implementted to solve the problem of localization in sensor networks where range measurements are available, see Nie (2009), Nie and Demmel (2006). However, here we propose a method for localization of the nodes using not only range measurements but other types of measurements as well; in particular we formulate the problem of localization using noisy range difference measurements. Furthermore, we propose a solution to the pose determination problem using the same methodology. Later in the paper, apart from applying polynomial optimization methods to address problems in network localization, we introduce the fact that the selection of anchors (nodes with known position) affects the precision of a localization solution, and we propose a method based on SDP to select anchors in a network such that this effect is minimized.

Finally, we look at the problem of enhancing synchronizability in a network. Noting that adding edges to the network in order to enhance its synchronizability is standard, we propose a method to add edges to the network using SDP. Then we propose a method to address the problem of edge addition in a distributed fashion. Finally we identify the node in a network loss of which inflicts the most damage on the synchronizability of the network using SDP.

2. POLYNOMIAL FUNCTION OPTIMIZATION

Before proceeding we present mathematical notions and definitions. We use $\mathbb{R}[\boldsymbol{x}]$, where $\boldsymbol{x} = [x_1, \cdots, x_n]$, to denote the ring of all polynomials in n indeterminants (variables), $x_1, x_2, \cdots, x_n$, with real coefficients. The set $I \subset \mathbb{R}[\boldsymbol{x}]$ is an *ideal* if $qh \in I$ for any $q \in I$ and $h \in \mathbb{R}[\boldsymbol{x}]$. Given polynomials $g_1, \cdots, g_r$ we write $\langle g_1, \cdots, g_r \rangle$ to represent the set of all polynomials that are polynomial linear combinations of $g_1, \cdots, g_r$. By Hilbert's Basis Theorem every ideal $I \subset \mathbb{R}[\boldsymbol{x}]$ is finitely generated. In other words, there always exists a finite set $\{f_1, \cdots, f_m\} \subset \mathbb{R}[\boldsymbol{x}]$ such that for every $f \in I$, we can find $g_i \in \mathbb{R}[\boldsymbol{x}]$ that satisfy $f = \sum_{i=1}^{m} g_i f_i$. Now we present the polynomial function optimization problem formally.

Problem 2.1. Consider a polynomial function $F(\boldsymbol{x})$ of degree $2d$, over the polynomial ring $\mathbb{R}[\boldsymbol{x}]$. Find F^* and $x^* = [x_1^*, \cdots, x_n^*]$ such that $F^* = F(\boldsymbol{x}^*) = \min F$.

An obvious method to find the solution to Problem 2.1 is to find all the critical points of F. The usual methods to solve such a system of nonlinear equations use Newton method, and there is not any guarantee that the solutions of the system are going to be found completely. However, there is another strong tool to find the global minimum and minimizer of polynomial functions which is described below.

2.1 Sum of Squares and Semidefinite Programming Approaches

A polynomial function $F(\boldsymbol{x})$ of degree $2d$ over the polynomial ring $\mathbb{R}[\boldsymbol{x}]$ is sum of squares (SOS) if one can write

$$F(\boldsymbol{x}) = \sum_{i=1}^{q} Q_i^2(\boldsymbol{x}) \tag{1}$$

where $q \in \mathbb{Z}^+$ and $Q_i(x)$ are the polynomials over $\mathbb{R}[\boldsymbol{x}]$. Denote the global minimizer and global minimum of a polynomial function $F(\boldsymbol{x})$, respectively, by x^* and $\gamma = F(\boldsymbol{x}^*)$. x^* can be calculated solving the following optimization problem:

$$\begin{aligned} &\text{maximize } \gamma \\ &\text{subject to } F(\boldsymbol{x}) - \gamma \geq 0. \end{aligned} \tag{2}$$

One can relax (2) and write it as

$$\begin{aligned} &\text{maximize } \gamma \\ &\text{subject to } F(\boldsymbol{x}) - \gamma \quad \text{is SOS}. \end{aligned} \tag{3}$$

Remark 2.1. The relaxed problem is often computationally much easier to solve, and may yield the same solution. However, in general (2) and (3) are not identical, since there are positive polynomials that are not in the SOS form. For more information see Parrilo (2003).

We know that any SOS polynomial $F(\boldsymbol{x})$ of degree $2d$, with x an n-tuple of variables, can be written as $F(\boldsymbol{x}) = Z^\top QZ$, where Z is a vector of all monomials of degree up to d obtained from the variables in $\boldsymbol{x}$ with the first entry equal to one, and Q is a positive semi-definite matrix obtained by solving a set of LMIs Parrilo (2003). So one can reformulate (3) as,

$$\begin{aligned} &\text{maximize } \gamma \\ &\text{subject to } Q - \hat{E}\gamma \geq 0, \end{aligned} \tag{4}$$

where $\hat{E}$ is a matrix with $E_{11} = 1$ and the rest of the entries are zero. The problem stated in (4) is an SDP problem and can be solved by SDP techniques Parrilo (2003). By solving the dual problem of the semidefinite programming problem stated in (4), one can obtain the minimizer of F as well, using the procedure in Henrion and Lasserre (2005).

Remark 2.2. For polynomials with 2 variables and degree of 4 (2) and (3) are equivalent (Parrilo (2003)).

In addition, we consider the following constraint optimization problem with polynomial F, g_i and h_j:

$$\begin{aligned} &\text{minimize } F(\boldsymbol{x}) \\ &\text{subject to } g_i(\boldsymbol{x}) \geq 0 \quad i = 1, \cdots, M \\ &\qquad\qquad\;\; h_j(\boldsymbol{x}) = 0 \quad j = 1, \cdots, N. \end{aligned} \tag{5}$$

Assume there exists a set of SOS $\sigma_i(x)$, and a set of polynomials $\lambda_j(x)$ such that

$$\begin{aligned} F(\boldsymbol{x}) - \gamma = \sigma_0(\boldsymbol{x}) + \sum_j \lambda_j(\boldsymbol{x}) h_j(\boldsymbol{x}) + \sum_i \sigma_i(\boldsymbol{x}) g_i(\boldsymbol{x}) \\ + \sum_i \sigma_{i_1,i_2}(x) g_{i_1}(\boldsymbol{x}) g_{i_2}(x) + \sigma_M \prod_{i=1}^{M} g_i(\boldsymbol{x}) \end{aligned} \tag{6}$$

Then γ is a lower bound for (5) Parrilo (2003). So by maximizing γ as before one can get a lower bound that gets tighter as the degree of (6) increases. There are wellknown SDP based solutions to the aforementioned problem. For more information one may refer to Nie et al. (2005) and references therein.

2.2 Noisy Cooperative Target Localization

The first problem that we consider in this section is as follows:

Problem 2.2. Consider n anchor agents in $\mathbb{R}^N$ ($N \in \{2,3\}$) at the known positions p_i, $i \in \{1, \cdots, n\}$, and an agent 0 at the unknown position p^*. Let the noisy measurement d_i of the distance of agent 0 to each agent i for $i \in \{1, \cdots, n\}$ be available (to agent 0). The task (of agent 0) is to produce the estimate p of p^* using the noisy distance measurements $d_1, \cdots, d_n$.

A solution to this problem can be obtained by finding the point $p = [x, y]^\top$ which solves the following minimization problem.

$$\text{minimize } J_r(p) = \sum_{i=1}^{n} \left(\|p - p_i\|^2 - d_i^2 \right)^2 \tag{7}$$

In general $J(p)$ is not convex (or concave), so ordinary convex optimization methods will not yield to the desired result. Now, we introduce another problem whose formulation was presented in Beck et al. (2008) in what follows.

Problem 2.3. Consider n anchor agents in $\mathbb{R}^N$ ($N \in \{2,3\}$) at the known positions p_i, $i \in \{1, \cdots, n\}$, another agent $n+1$ is at the origin, and agent 0 at the unknown position p^*. Let δ_i, the noisy measurement of $\delta_i^* = d_i^* - \|p^*\|$ for $i \in \{1, \cdots, n\}$, be available (to agent 0). The task (of agent 0) is to produce the estimate p of p^* using the noisy range difference measurements $\delta_1, \cdots, \delta_n$.

To solve the problem one is interested in solving the following minimization problem

$$\text{minimize} J_d(p) = \sum_{i=1}^{n} \left(\delta_i^2 - \|p_i\|^2 + 2\delta_i \|p\| + 2p_i^\top p \right)^2. \tag{8}$$

Denoting $\|p\|$ by D, and considering that $D^2 - x^2 - y^2 = 0$, (8) can be rewritten as

$$\begin{aligned} &\text{minimize } \sum_{i=1}^{n} \left(\delta_i^2 - \|p_i\|^2 + 2\delta_i D + 2p_i^\top \begin{bmatrix} x \\ y \end{bmatrix} \right)^2 \\ &\text{subject to } D^2 - x^2 - y^2 = 0. \end{aligned} \tag{9}$$

By setting $F(x) = J_d(p)$ and using (4), based on Remark 2.2 we can find the exact solution to (7). For (9), note that (9) is a special case of (5), where $F(x) = J_d(p)$, inequality constraints do not exist, and the only equality constraint is $h_1(p) = \|p\|^2 - D^2$, and we can find the solution to it by using the methods introduced in the previous section. For more information and a comparison with existing results see Shames et al. (2009c)

2.3 Range-Based Pose Determination

In Zhou and Roumeliotis (2008), Shames et al. (2009d) the problem of determining the relative reference frames of a pair of robots that move on a plane while measuring distance to each other is studied. In Trawny et al. (2007) the authors introduced the same problem in three dimensional space. Next, we state a more general form of these problems and state a solution.

Problem 2.4. Consider two agents (robots) A_1 and A_2 in $\mathbb{R}^N$, $N = 2, 3$ whose initial reference frames are indicated by Σ_1 and Σ_2 respectively. The two agents move through a sequence of n unknown different positions with a reference frame associated with each position, $\Sigma_1, \Sigma_3, \cdots, \Sigma_{2n-1}$ for A_1 and $\Sigma_2, \Sigma_4, \cdots, \Sigma_{2n}$ for A_2, where $n \in \mathbb{Z}^+$. Their inter-agent distance, $d_{i,i+1}$ is measured at each of these positions, where $i \in \{1, 3, \cdots, 2n-1\}$. In addition each agent is capable of estimating its current reference frame orientation and displacement with respect to its initial reference frame using dead-reckoning (odometry). In other words A_1 and A_2 estimate the position vectors $p_3^1, \cdots, p_{2n-1}^1$ and $p_4^2, \cdots, p_{2n}^2$ respectively. Additionally they know the rotation matrices that relate the orientations of their initial reference frame and all the others in their own sequence, e.g. R_3^1 is the rotation matrix relating Σ_1 to Σ_3. The task is to find p_2^1 and R_2^1 using this information.

As stated in Shames et al. (2009d) for the case $N = 2$ we need $n = 4$ measurements to be able to answer the problem uniquely. In what comes next we focus on the three dimensional case. First, we assume that the origins of the reference frames of the agents at each time are the vertices of a graph, and if the distance between any pair of the origins is known there is an edge connecting them together. We call the resulting graph G_p. Without loss of generality we select Σ_1 as our reference frame for solving the problem. As a result of this selection, the origins of the reference frames $\Sigma_1, \Sigma_3, \cdots, \Sigma_{2n-1}$ can be calculated and they can be considered as anchor points for the formation with G_p as its underlying graph (In addition the rotation matrix relating each of them to Σ_1 can be calculated as well.). The goal here is to find the positions of the origins of $\Sigma_2, \Sigma_4, \cdots, \Sigma_{2n}$ in Σ_1.

The origins of the reference frames $\Sigma_1, \cdots, \Sigma_{2n-1}$ form a complete graph, as do the origins of $\Sigma_2, \cdots, \Sigma_{2n}$. For $n \geq 4$, these two complete graphs are connected with n edges (Note that these edges do not share any vertex with each other.) From Yu et al. (2006) we know that the resulting graph is a globally rigid graph if (and only if) $n \geq 7$. So Problem 2.4 generically has a unique solution in the presence of 7 or more distance measurements. In order to answer Problem 2.4 we assume that $n \geq 7$. We can construct the following cost function.

$$J_p(P) = \sum_{i=1}^{n} \left(\|p_{2i-1}^1 - p_{2i}^1\|^2 - d_{2i-1,2i}^2 \right)^2 + \sum_{i,j \in \{1,\cdots,n\}} \left(\|p_{2j}^1 - p_{2i}^1\|^2 - \|p_{2j}^2 - p_{2i}^2\|^2 \right)^2 \quad (10)$$

where $P = [p_2^{1\top}, \cdots, p_{2n}^{1\top}]$. The global minimizer of $J_p(P)$, $P^* = [p_2^{1*\top}, \cdots, p_{2n}^{1*\top}]$ gives us the solution to the first part of Problem 2.4. To calculate the rotation matrix R_2^1, first we note that this rotation matrix can be written as Horn (1987)

$$\begin{bmatrix} s^2+t^2-u^2-v^2 & 2(tu-sv) & 2(tv+su) \\ 2(tu+sv) & s^2-t^2+u^2-v^2 & 2(uv-st) \\ 2(tv-su) & 2(uv+st) & s^2-t^2-u^2+v^2 \end{bmatrix} \quad (11)$$

where

$$s^2+t^2+u^2+v^2 = 1, \quad (12)$$

and $s, t, u, v \in \mathbb{R}$. Then each of the equations of the form

$$p_{2i}^1 = p_2^1 + R_2^1 p_{2i}^2, \quad i \in \{1, \cdots, n\} \quad (13)$$

results in three scalar equations $\Phi_{ij} = 0$, $j = 1, 2, 3$, in indeterminants s, t, u, and v. We construct the objective function, $J_\Phi(s,t,u,v) = \sum_{i=1}^{n\geq 7} \sum_{j=1}^{3} \Phi_{ij}^2$.

Then the global minimizer of J_Φ subject to the constraint $s^2 + t^2 + u^2 + v^2 = 1$ is the estimate for R_2^1. One can use the method based on sum of squares that was introduced earlier to find the global minimizers of J_p and J_Φ subject to $s^2+t^2+u^2+v^2 = 1$.

2.4 Anchor Selection in a Sensor Network

In this section we review the problem of minimization of the effect of noise on localization of a network by anchor selection. While we again use convex optimization, the techniques that are used here and in the rest of the paper are quite different from the previous ones.

Consider a network $(\mathcal{G}, \Pi)$ in $\mathbb{R}^2$ with vertex set $\mathcal{V}\{i\}_{i=1}^{|\mathcal{V}|}$, each vertex representing a node with the same label, and edge set $\mathcal{E} = \{e_i\}_{i=1}^{|\mathcal{E}|}$. An edge connects two vertices if the nodes associated with those vertices have an information link, e.g. have communication, can sense each other, etc. If e_i connects vertices j and k, in the rest of this paper for simplicity we denote e_i by $\{j, k\}$. The graph $\mathcal{G}$ is called the *underlying graph* of $\mathcal{N}$, and with the coordinates set $\Pi = \{p_i\}_{i=1}^{|\mathcal{V}|}$. Let the coordinates of vertex j be $\pi_j = (x_j, y_j)^\top$. The rigidity matrix is defined with an arbitrary ordering of the vertices and edges, and has $2|\mathcal{V}|$ columns and $|\mathcal{E}|$ rows. Each edge gives rise to a row, and if the edge links vertices j and k, the nonzero entries of the row of the matrix are in columns $2j-1$, $2j$, $2k-1$ and $2k$, and are respectively $x_j - x_k$, $y_j - y_k$, $x_k - x_j$, $y_k - y_j$. We make the following assumption on the network.

Assumption 2.1. Consider a network, $\mathcal{N}(\mathcal{G}, \Pi)$, with the underlying sensing graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Suppose a subset of the agents are anchor agents, i.e. we know their global position. Denote the corresponding subset of $\mathcal{V}$ by $\mathcal{V}'$. Adopt the standard convention that two anchor agents know their inter-agent distance uncorrupted by noise, and let $\mathcal{E}' \subset \mathcal{E}$ denote the set of edges in $\mathcal{G}$ joining vertices in $\mathcal{V}'$.

The problem of interest is presented below.

Problem 2.5. Consider a formation of the type described in Assumption 2.1, such that the reduced rigidity matrix, $R_{r,|\mathcal{E}\setminus\mathcal{E}'|\times 2(|\mathcal{V}|-|\mathcal{V}'|)}$, is obtained by discarding the rows and columns associated to the position of the anchor nodes and their interconnecting edges. What selection of $|\mathcal{V}'|$ nodes as anchors results in the smallest localization error?

Note that the statement of Problem 2.5 fails to specify what is meant by the smallest. We shall adopt the view that we need to mitigate the effect of random errors, one such scalar measure is $\lambda_{max}(R_r^\dagger (R_r^\dagger)^\top)$, where $R_r^\dagger$ is the pseudo-inverse of R_r, noting that use of other measures is possible, as well. For more details and discussions see Shames et al. (2009b).

One can recast Problem 2.5 as the problem of maximizing $\lambda_{min}(R_r(R_r^\top)$. Indeed we have the following straightforward proposition:

Proposition 2.1. Let r_i denote the i-th column of the rigidity matrix, $R_{|E|\times 2|V|}$ The smallest eigenvalue of $R_r R_r^\top$ is equal to the smallest nonzero eigenvalue of $\bar{R}_r \bar{R}_r^\top = \sum_{i=1}^{2n} s_i r_i r_i^\top$ where $\mathbf{1}^\top s = 2n - 2m$, $s_i \in \{0, 1\}$, $\quad i = 1, \cdots, 2n$, $s_{2i-1} = s_{2i}$, and $\mathbf{1}$ is a vector with 1 entries.

So the problem of anchor selections can be formulated as follows (Assume eigenvalues of a symmetric X obey $\lambda_i(X) >$

$\lambda_{i+1}(X)$):

$$\begin{aligned} \text{maximize} \quad & \sum_{j=2n-2m}^{|E|} \lambda_j \left(\sum_{i=1}^{2n} s_i r_i r_i^\top \right) \\ \text{subject to} \quad & \mathbf{1}^\top s = 2n - 2m \\ & s_i \in \{0,1\}, \quad i = 1, \cdots, 2n \\ & s_{2i-1} = s_{2i} \end{aligned} \tag{14}$$

Now, because the constraint $s_i \in \{0,1\}$ is Boolean, (14) is not solvable using convex optimization techniques. However one can relax this constraint and replace it with a convex constraint. We have the following relaxed version of (14):

$$\begin{aligned} \text{maximize} \quad & \sum_{j=2n-2m}^{|E|} \lambda_j \left(\sum_{i=1}^{n} s_i r_i r_i^\top \right) \\ \text{subject to} \quad & \mathbf{1}^\top s = 2n - 2m \\ & 0 \le s_i \le 1, \quad i = 1, \cdots, 2n \\ & s_{2i-1} = s_{2i}. \end{aligned} \tag{15}$$

One can solve (14) using standard convex optimization techniques. Assume s^* is the solution to this problem; it is not necessarily a solution to the original problem, since s_i^* can take a non-integer value. For the method to compute s from s^* see Shames et al. (2009b).

Standard convex optimization may appear not immediately applicable to solving the convex optimization problem (15); however, if we cast the problem in the SDP framework it can be solved easily, see Shames et al. (2009b) and the references therein.

3. CONVEX OPTIMIZATION AND ENHANCING SYNCHRONIZABILITY

In this section we consider the problem of adding edges (information links) to a network in order to enhance the synchronizability of the network. Consider a network, $\mathcal{N}$, of n interconnected nodes where the node interconnection is represented by an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, with vertex set $\mathcal{V}\{i\}_{i=1}^{|\mathcal{V}|=n}$, each vertex representing a node with the same label, and edge set $\mathcal{E} = \{e_i\}_{i=1}^{|\mathcal{E}|=p}$. An edge connects two vertices if the nodes associated with those vertices have an information link, e.g. have communication, can sense each other, etc. If e_i connects vertices j and k, in the sequel for simplicity we denote e_i by $\{j, k\}$. The graph $\mathcal{G}$ is called the *underlying graph* of $\mathcal{N}$. Additionally, for each node labelled i one has the state dynamics

$$\dot{x}_i = f(x_i) + c \sum_{j=1}^{n} a_{ij} x_j, \quad i = 1, \cdots, n \tag{16}$$

where $x_i = [x_{i1}, \cdots, x_{id}]^\top \in \mathbb{R}^d$, $d \in \mathbb{N}$, is the state variable of node i, $c > 0$ is the coupling strength Barahona and Pecora (2002), Wang and Chen (2002), and the differentiable function f represents the dynamics of an isolated node. For the sake of simplicity and without loss of generality, we assume that the weight for each edge is equal to one. Furthermore, we define the coupling matrix $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{n \times n}$ to be $\mathcal{A} = -\mathcal{L}$, where $\mathcal{L}$ is the Laplacian matrix associated with $\mathcal{G}$. We know that $\mathcal{L} = \mathcal{H}^\top \mathcal{H}$, where $\mathcal{H}$ is the incidence matrix associated with $\mathcal{G}$. One can construct $\mathcal{H} = [h_{ij}] \in \mathbb{R}^{p \times n}$ in the following way. If edge e_i is incident on nodes (vertices) j and k, where $j < k$, then $h_{ij} = 1$, and $h_{ik} = -1$, and $h_{il} = 0$ for any $l \notin \{j, k\}$. In other words, the i-th row of the matrix $\mathcal{H}$ corresponds to edge e_i. For connected $\mathcal{G}$, the following relation between the eigenvalues of $\mathcal{A}$ holds, see Wu and Chua (1995):

$$0 = \lambda_1(\mathcal{A}) > \lambda_2(\mathcal{A}) \ge \cdots \ge \lambda_n(\mathcal{A}) \tag{17}$$

The system (16) obtains synchronization when $x_1(t) = \cdots = x_n(t) = z(t)$ as $t \to \infty$, where trajectory z can be a stable equilibrium point, a limit cycle or a chaotic attractor of an isolated node, with:

$$\dot{z}(t) = f(z(t)). \tag{18}$$

It is shown in Wang and Chen (2002) network synchronizability depends on the second largest eigenvalue of $\mathcal{A}$, and consequently one can enhance this synchronizability by increasing the value of $\lambda_2(\mathcal{A})$. So we identify $\lambda_2(\mathcal{A})$ as a performance index and try to minimize it for the rest of this section. This idea is described in the next section in more detail.

3.1 Enhancing Synchronizability Via Edge Addition

A known procedure to enhance synchronizability of a network is to add new edges between agents in the network, e.g. see Chavez et al. (2005); Zhao et al. (2005). In this section we discuss a result describing a methodology for adding edges to the network using convex optimization. The results reviewed here regarding centralized edge addition is mainly based on Shames et al. (2009a). Before proceeding we make the following definition.

Definition 3.1. The complement of a graph $\mathcal{G}$ is the graph $\mathcal{G}^c$ with the same vertex set but whose edge set consists of the edges not present in $\mathcal{G}$ (i.e., the complement of the edge set of $\mathcal{G}$ with respect to all possible edges on the vertex set of $\mathcal{G}$).

Let $\mathcal{H} = \left[h_1^\top, \cdots, h_p^\top\right]^\top$, where $h_i = [h_{i1}, \cdots, h_{in}]$. We have $\mathcal{A} = -\mathcal{H}^\top \mathcal{H} = -\sum_{i=1}^{p} h_i^\top h_i$. In addition, we define $\mathcal{G}^c(\mathcal{V}, \mathcal{E}^c)$ as the complement graph of $\mathcal{G}$, where $\mathcal{E}^c = \{e_i^c\}_{i=1}^{|\mathcal{E}^c|=p^c}$, and denote $\mathcal{H}^c = \left[{h_1^c}^\top, \cdots, {h_{p^c}^c}^\top\right]^\top$ to be its incidence matrix, where $p^c = \frac{n(n-1)}{2} - p$. Now we formally define the following problem.

Problem 3.1. Consider a network of n interconnected nodes with underlying undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. The goal is to add m edges to $\mathcal{G}$, call the new graph $\mathcal{G}'$ with associated coupling matrix $\mathcal{A}'$, to obtain the smallest possible $\lambda_2(\mathcal{A}')$. Where should one add these edges to obtain this goal?

First we describe $\mathcal{A}'$ in more detail. The new graph $\mathcal{G}'$ can be described as a graph with the vertex set $\mathcal{V}$ as before and edge set $\mathcal{E}'$, where $\mathcal{E}' = \mathcal{E} \cup \mathcal{E}^+$, $\mathcal{E}^+ \subset \mathcal{E}^c$, and $|\mathcal{E}^+| = m$. Additionally, term the graph with $\mathcal{V}$ as vertex set and $\mathcal{E}^+$ as edge set, $\mathcal{G}^+$. Call the incidence matrix and Laplacian matrix associated with this graph, $\mathcal{H}^+$ and $\mathcal{L}^+$ respectively. Then, the Laplacian matrix of $\mathcal{G}'$, viz. $\mathcal{L}'$ can be calculated as $\mathcal{L}' = \mathcal{L} + \mathcal{L}^+$, and consequently $\mathcal{A}' = -\mathcal{L}'$.

We can cast Problem 3.1 as the following minimization problem.

$$\begin{aligned} \text{minimize} \quad & \lambda_2 \left(\mathcal{A} - \sum_{i=1}^{p^c} s_i {h_i^c}^\top h_i^c \right) \\ \text{subject to} \quad & \mathbf{1}^\top s = m \\ & s_i \in \{0,1\}, \quad i = 1, \cdots, p^c. \end{aligned} \tag{19}$$

where $s = [s_1, \cdots, s_{p^c}]$, and $\mathbf{1}$ is a vector with 1 entries.

Remark 3.1. Note that in (19), the objective function is actually $\lambda_2(\mathcal{A}')$, and $-\sum_{i=1}^{p^c} s_i h_i^{c\top} h_i^c = \mathcal{A}^+$. Because the s_i are all 0 or 1, the matrix $\sum_{i=1}^{p^c} s_i h_i^{c\top} h_i^c$ is obtained by selecting certain edges (corresponding to $s_i = 1$) to constitute $\mathcal{E}^+$ and the matrix $\mathcal{A} - \sum_{i=1}^{p^c} s_i h_i^{c\top} h_i^c$ is $\mathcal{A}'$. The constraint $\mathbf{1}^\top s = m$ ensures that precisely m additional edges are selected.

Before continuing further we introduce the following proposition:

Proposition 3.1. Minimization of the objective function of (19) is the same as minimization of $\sum_{j=1}^{2} \lambda_j \left(\mathcal{A} - \sum_{i=1}^{p^c} s_i h_i^{c\top} h_i^c \right)$.

Using Proposition 3.1 and the relaxation used in the previous section we can provide a solution to (19). For more details on how the edge addition is executed the reader may refer to Shames et al. (2009a).

3.2 A Distributed Approach to Edge Addition

Here we propose a *distributed* method for enhancing synchronizability of a network by adding new edges to the network. By distributed we mean that the edge addition is achieved by having access to local information at each individual node Again consider network $\mathcal{N}$ with underlying graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. For any $i \in \mathcal{V}$ define $\mathcal{G}_i(\mathcal{V}_i, \mathcal{E}_i)$ such that $j \in \mathcal{V}_i$ if there is path of maximum length 2 between i and j, and $\{j,k\} \in \mathcal{E}_i$ if $\{j,k\} \in \mathcal{E}$ and $j,k \in \mathcal{V}_i$. Furthermore, call the coupling matrix associated with each $\mathcal{G}_i$, $\mathcal{A}_i$. Moreover, we use η_i to denote the set of the immediate neighbours of i, and $L_\mathcal{G}$ as the diameter of $\mathcal{G}$. We propose the following algorithm for distributed addition of one edge.

Algorithm 1.

1. For each $\mathcal{G}_i$, $i = 1, \cdots, n$, solve (19) with $m := 1$ and call the edge that is proposed to be added ε_i.
2. Set $\rho_i := \dfrac{\lambda_2(\mathcal{A}_i') - \lambda_2(\mathcal{A}_i)}{\lambda_2(\mathcal{A}_i)}$ for each $\mathcal{G}_i$
3. Set $t := 0$
4. while $t \leq L_\mathcal{G}$
   1. Each node i broadcasts ε_i and ρ_i to all $j \in \eta_i$.
   2. Each node j compares all ρ_i, $i \in \eta_j$ with ρ_j.
   3. Node j replaces ρ_j by $\max_{k \in \eta_j \cup j} \{\rho_k\}$ and ε_j by the edge associated with the new ρ_j.
   4. Set $t := t + 1$
5. Add edge $\varepsilon = \varepsilon_1 = \cdots = \varepsilon_n$ to the network.

Proposition 3.2. After $L_\mathcal{G}$ steps ρ_i and ε_i, $\forall i \in \mathcal{V}$ are equal under Algorithm 1.

Proof. Assume $\rho_k = \max_{i \in \mathcal{V}}\{\rho_i\}$. Each agent j receives the value ρ_k and replaces the value of ρ_j by this after $t = L_{jk}$ steps, where L_{jk} is the shortest path between j and k in $\mathcal{G}$. Furthermore, since $L_{jk} \leq L_\mathcal{G}$, after at most $L_\mathcal{G}$ steps, ρ_i and ε_i, $\forall i \in \mathcal{V}$ are equal.

■

Remark 3.2. For distributed addition of m edges, each node i calculates m proposed edges, $\varepsilon_{i1}, \cdots, \varepsilon_{im}$ and their corresponding $\rho_{i1}, \cdots, \rho_{im}$. The edges with the highest associated ρ throughout the network will be added as new edges.

Node	ρ_i	ε_i	Node	ρ_i	ε_i
1	1.5858	$\{1,5\}$	2	1.5858	$\{1,5\}$
3	0.8993	$\{2,6\}$	4	1.1602	$\{1,8\}$
5	1.1602	$\{1,8\}$	6	2	$\{3,8\}$
7	2	$\{3,8\}$	8	0.9472	$\{5,8\}$

Table 1. ε_i and ρ_i for each node in graph depicted in Fig. 1(a)

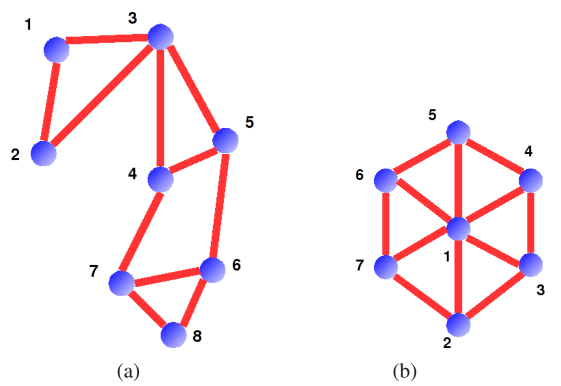

Fig. 1. (a) The graph presented in Example 3.1. (b) A Sample wheel graph.

Example 3.1. Consider the network depicted in Fig. 1(a). We run Algorithm 1 to add one edge to the graph locally. The algorithm results in addition of edge $\{3,8\}$. The values for ρ_i and ε_i, $\forall i \in \mathcal{V}$ are presented in Table 1.

3.3 The Effect of Node Loss on Network Synchronization

Imagine a graph similar to the one depicted in Fig 1(b). It is rather obvious to see that losing node 1 has the greatest effect on the synchronizability of the network. However, the question here is: how one can identify in a large network the loss of which node (or nodes) inflicts the most damage to the synchronizability of the network. We formally define the problem of interest next.

Problem 3.2. Consider the network as before with underlying graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, coupling matrix $\mathcal{A}$, The loss of which m nodes inflicts the most damage to the synchronizability of the network?

We need to solve the following optimization problem to answer the problem.

$$\begin{aligned} \text{minimize} \quad & \sum_{j=1}^{2} \lambda_j \left(- \sum_{i=1}^{p} s_i {h_i}^\top h_i \right) \\ \text{subject to} \quad & \mathbf{1}^\top s \leq p \\ & s_i \in [0, p], \quad i = 1, \cdots, p \end{aligned} \tag{20}$$

Then for each node $i \in \mathcal{V}$ we define an index $\mu_i = \sum_{j \in \eta_i} s_j$, where $j \in \alpha_i$, where α_i is the set of edges incident on vertex i. The m nodes with the largest μ_i are identified as the nodes whose loss inflicts the most damage to the synchronizability of the network.

Example 3.2. Consider the network introduced in Example 3.1, Here the aim is to identify the loss of which node will result in the worst damage to synchronizability of the network. Solving (20) we find μ_i associated with each node i, and since μ_3 is the largest we find node 3 to be the one whose loss inflicts the most damage to synchronizability of the network. By looking at the original graph it was obvious that by losing 3, the network becomes disconnected and hence the synchronizability suffers the most damage. To further test the method we add an edge between nodes 1 and 8; in this case the node i with the largest μ_i is found to be 8.

4. CONCLUSIONS

In this paper we studied convex optimization based methods with applications in sensor networks. We first proposed methods based on SOS relaxation to solve cooperative target localization and range-based pose determination in the presence of noise. Second, we discussed the application of convex optimization in selecting anchors in a network to reduce the effect of noise on localization performance, and finally we have proposed a distributed algorithm for edge addition to a network and introduced a method to identify the node in a network whose loss of which inflicts the most damage on the synchronizability of the network.

Possible future research directions include but are not limited to application of the same optimization methods to solve formation acquisition problems, optimal anchors placement under geometric constraints, etc.

ACKNOWLEDGEMENTS

This work is supported by NICTA, which is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

REFERENCES

Barahona, M. and Pecora, L. (2002). Synchronization in small-world systems. *Physical Review Letters*, 89, 054101–(1–4).

Beck, A., Stoica, P., and Li, J. (2008). Exact and approximate solutions of source localization problems. *IEEE Transactions on Signal Processing*, 56, 1770–1778.

Biswas, P., Lian, T.C., Wang, T.C., and Ye, Y. (2006). Semidefinite programming based algorithms for sensor network localization. *ACM Transaction on Sensor Networks*, 2(2), 188–220.

Biswas, P. and Ye, Y. (2004). Rsemidefinite programming for ad hoc wireless sensor network localization. In *ACM IPSN*, 46–54.

Carter, M.W., Jin, H.H., Saunders, M.A., and Ye, Y. (2006). An adaptive subproblem algorithm for scallable wireless sensor network localization. *SIAM Journal on Optimization*, 1102–1128.

Chavez, M., Hwang, D.U., Amann, A., Hentschel, H.G.E., and Boccaletti, S. (2005). Synchronization is enhanced in weighted complex networks. *Physical Review Letters*, 94, 218701.

Ding, Y., Krislock, N., Qian, J., and Wolkowicz, H. (2008). Sensor network localization, euclidean distance matrix completions, and graph realization. *Optimization and Engineering*.

Henrion, D. and Lasserre, J. (2005). Detecting global optimality and extracting solutions in gloptipoly. In D. Henrion and A. Garulli (eds.), *In Positive Polynomials in Control*, Lecture Notes on Control and Information Sciences. Springer-Verlag.

Horn, B.K.P. (1987). Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America*, 4, 629–642.

Nie, J. (2009). Sum of squares methods for sensor network localization. To appear in Computational Optimization and Applications.

Nie, J., Demmel, J., and Sturmfels, B. (2005). Minimizing polynomials via sum of squares over the gradient ideal. *Mathematical Programming, Ser. A*, 106, 587–606.

Nie, J. and Demmel, J. (2006). Sparse sos relaxations for minimizing functions that are summations of small polynomials. *SIAM Journal on Optimization*, 19(4), 1534–1558.

Parrilo, P. (2003). Semidefinite programming relaxations for semi-algebraic problems. *Mathematical Programming*, 96(2), 293–320.

Shames, I., Anderson, B.D.O., Wang, X., and Fidan, B. (2009a). Network synchronizability enhancement using convex optimization. In *Proc. of European Control Conference*. Budapest, Hungary.

Shames, I., Fidan, B., and Anderson, B.D.O. (2009b). Minimization of the effect of noisy measurements on localization of multi-agent autonomous formations. *Automatica*, 45(4), 1058–1065.

Shames, I., Taghipour-Bibalan, P., Fidan, B., and Anderson, B.D.O. (2009c). Polynomial methods in noisy network localization. In Proceedings of Mediterranean Control Conference.

Shames, I., Fidan, B., Anderson, B.D.O., and Hmam, H. (2009d). Cooperative self-localization of mobile agents moving in planar formations. Submitted to IEEE Transactions on Robotics.

Trawny, N., Zhou, X., Zhou, K., , and Roumeliotis, S.I. (2007). D relative pose estimation from distance-only measurements. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, 1071–1078.

Wang, X.F. and Chen, G. (2002). Synchronization in scale-free dynamical networks: Robustness and fragility. *IEEE Transactions on Circuits and Systems-I*, 49, 54–62.

Wang, Z. and Ding, Y. (2008). Real-time tracking for sensor networks via sdp and gradient method. In *MELT '08: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, 109–112.

Wu, C. and Chua, L. (1995). Application of graph theory to the synchronization in an array of coupled nonlinear oscillators. *IEEE Transactions on Circuits and Systems-I*, 42, 494–497.

Ye, Y. (2006). Semidefinite programming for sensor network and graph localization. In A.J. Kurdila, P.M. Pardalos, and M. Zabarankin (eds.), *IRobust Optimization-Directed Design*, onconvex Optimization and Its Applications. Springer-Verlag.

Yu, C., Fidan, B., and Anderson, B. (2006). Principles to control autonomous formation merging. In *American Control Conference*, 762–768. Minneapolis, Minnesota.

Zhao, M., Zhou, T., Wang, B., and Wang, W. (2005). Enhanced synchronizability by structural perturbations. *Physical Review E*, 72, 057102.

Zhou, X.S. and Roumeliotis, S.I. (2008). Robot-to-robot relative pose estimation from range measurements. *IEEE Transactions on Robotics.*, 24(6), 1379–1393.