2020 IEEE International Conference on Robotics and Automation (ICRA)
31 May - 31 August, 2020. Paris, France

Optimal Topology Selection for Stable Coordination of Asymmetrically Interacting Multi-Robot Systems

Pratik Mukherjee[1], Matteo Santilli[2], Andrea Gasparri[2] and Ryan K. Williams[1]

***Abstract*— In this paper, we address the problem of *optimal topology selection* for stable coordination of multi-robot systems with asymmetric interactions. This problem arises naturally for multi-robot systems that interact based on sensing, e.g., with limited field of view (FOV) cameras. From our previous efforts on motion control in such settings, we have shown that *not all interaction topologies yield stable coordinated motion when asymmetry exists*. At the same time, not all robot-to-robot interactions are of equal quality, and thus we seek to optimize asymmetric interaction topologies subject to the constraint that the topology yields stable multi-robot motion. In this context, we formulate an optimal topology selection problem (OTSP) as a mixed integer semidefinite programming (MISDP) problem to compute optimal topologies that yield stable coordinated motion. Simulation results are provided to corroborate the effectiveness of the proposed OTSP formulation.**

I. Introduction

Multi-robot systems are experiencing a recent swell in interest from the robotics community. Indeed, multi-robot systems are being actively deployed in various important applications such as search and rescue [1] and autonomous inventory management [2]. However, when operating in a realistic environment, a multi-robot system must rely on perceptual sensors like cameras, laser rangefinders, ultrasonic detectors, etc., all of which exhibit limited fields of view (FOVs). From a theoretical perspective, limited FOV perception induces *asymmetry* in robot-to-robot interaction which introduces the possibility of degeneracies in typical coordinated motion control schemes such as [3]–[11]. Specifically, our recent work [12] demonstrated that unlike symmetrically interacting systems, asymmetric interactions must be carefully chosen to yield stable and safe coordinated motion. At the same time, many robotic platforms have an array of sensors that facilitate robot-to-robot interaction, and thus it is natural to ask: *what is the optimal selection of asymmetric interactions that yield stable coordinated motion in a multi-robot system?*

Related work in directed consensus has been well-studied such as [13], [14]. However, topology control for directed graphs is quite sparse, with recent examples [12], [15], [16] focusing on overcoming the theoretical shortcuts that are lost when the symmetry assumption is broken. Interaction optimization is instead a more mature area, with examples including connectivity maximization [17], [18] and optimal rigid graph construction for multi-agent localization [19]. More works like [20]–[22], have formulated network topology design using convex optimization techniques like mixed integer semidefinite programming (MISDP) for applications such as efficient communication for average consensus, and improvement of network connectivity and reconfiguration of a multi-robot system in the event of resource failure, respectively.

While recent works have made progress in various areas, we propose in this work a problem that spans asymmetric control and interaction optimization. In this regard, our contribution is as follows: using our existing framework [12], we formulate a novel optimal topology selection problem (OTSP) as an MISDP which allows us to compute optimal topologies that yield stable coordinated motion. We corroborate our theoretical formulation with simulations that demonstrate the effectiveness of our approach. It is important to point out that we focus in this work on the fundamental characteristics of our OTSP for stable coordination. Thus, the developed formulation is centralized in order to clarify the complex nature of our problem. Our ultimate goal is to extend our methods to a distributed context.

The rest of the paper is organized as follows. Background material and notations are summarized in the preliminaries in Section II. We then motivate our OTSP problem in Section III. In Section IV, the formulation of OTSP as an MISDP is presented. Finally, we provide simulation results in Section V and conclude the paper in Section VI.

II. Preliminaries

We begin with elements from graph theory used for modeling robot interaction. A dynamic directed graph is denoted by $\mathcal{G}(t) \triangleq (\mathcal{V}, \mathcal{E}(t))$[1] at time $t \in \mathbb{R}_{\geq 0}$, with node set $\mathcal{V} \triangleq [v_1, ..., v_n]$ representing n coordinating robots, and directed edge set $\mathcal{E}(t) \subseteq \mathcal{V} \times \mathcal{V}$, where edge $(i,j) \in \mathcal{E}^2$ indicates *asymmetric* interaction between a robot i sensing another robot j, i.e., $(i,j) \in \mathcal{E} \implies (j,i) \in \mathcal{E}$. It is also assumed that $(i,i) \notin \mathcal{E}$. The *incidence matrix* $\mathcal{B}(\mathcal{G}(t)) \in \mathbb{R}^{n \times |\mathcal{E}|}$ of a graph $\mathcal{G}(t)$, is a matrix with rows indexed by robots and columns indexed by edges, such that $\mathcal{B}_{ij} = 1$ if the j^{th} edge, denoted e_j, leaves vertex v_i (robot i senses robot j), -1 if it enters vertex v_i (robot j senses robot i), and 0 otherwise. The *outgoing incidence matrix* $\mathcal{B}_+$ contains only the outgoing (positive) parts of the incidence matrix $\mathcal{B}$, with incoming (negative) parts set to zero. The *undirected Laplacian matrix* $\mathcal{L} \in \mathbb{R}^{n \times n}$ is obtained as $\mathcal{L} = \mathcal{B}\mathcal{B}^T$, whereas the *directed*

[1]P. Mukherjee and R.K. Williams are with Electrical and Computer Engineering Department, Virginia Polytechnic Institute and State University, Blacksburg, VA USA, {mukhe027, rywilli1}@vt.edu

[2]M. Santilli and A. Gasparri are with the Engineering Department, Roma Tre University Roma, 00146, Italy, matteo.santilli@uniroma3.it, gasparri@dia.uniroma3.it

[1]Note that dependence on time, state, and/or a graph will only be shown when introducing new concepts or symbols. Subsequent usage will drop these dependencies for clarity of presentation.

[2]We use shorthand (i,j) for edge $(v_i, v_j) \in \mathcal{E}$ for convenience.

978-1-7281-7395-5/20/$31.00 ©2020 IEEE

Laplacian matrix $\mathcal{L}_d \in \mathbb{R}^{n\times n}$ is computed as $\mathcal{L}_d = \mathcal{B}\mathcal{B}_+^T$. The undirected Laplacian matrix $\mathcal{L}$ is symmetric positive-semidefinite, whereas the directed Laplacian matrix $\mathcal{L}_d$ is generally asymmetric and indefinite. We will also make use of the *undirected edge Laplacian* $\mathcal{L}_\mathcal{E} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$ defined as $\mathcal{L}_\mathcal{E} = \mathcal{B}^T\mathcal{B}$ and the *directed edge Laplacian* $\mathcal{L}_\mathcal{E}^d \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$ given by $\mathcal{L}_\mathcal{E}^d = \mathcal{B}^T\mathcal{B}_+$ as described in [23] and [24]. For any vector $T \in \mathbb{R}^m$, $\mathbf{Diag}(T) \in \mathbb{R}^{m\times m}$ denotes a matrix with elements of T along its diagonal. $\|M\|_F$ denotes the Frobenius norm of a matrix $M \in \mathbb{R}^{n\times n}$. We use $\mathbf{1}^m$ and $\mathbf{0}^m$ to represent a vector of ones and zeros of dimension m, respectively. The trace and rank of a matrix are given by $\mathbf{trace}(M)$ and $\mathbf{rank}(M)$, respectively. A symmetric positive semidefinite matrix M is denoted as $M \succeq 0$. Finally, I_d is an identity matrix of size d.

A. Stable Directed Graphs

Here we briefly review important results from our stable coordination framework originally proposed in [12], which specifies the conditions under which asymmetrically interacting multi-robot systems are stable. Assuming robots have a single integrator dynamics in $\mathbb{R}^d$, the Lyapunov function for studying stability $V : \mathbb{R}^{nd} \to \mathbb{R}_{\geq 0}$ can be defined in the standard manner $V(\mathbf{x}(t)) = \sum_{i=1}^n \sum_{j\neq i} V_{ij}(\|x_{ij}(t)\|)$ where $\mathbf{x}(t) = [x_1^T(t), \ldots, x_n^T(t)]^T \in \mathbb{R}^{nd}$ is the stacked vector of robot states, $\|x_{ij}(t)\| = \|x_i(t) - x_j(t)\|$, and V_{ij} is an energy function from which robot controls are derived as $u_i(t) = -\nabla_{x_i} V_{ij}$. From [12] we derive an *edge-based* form of $\nabla_\mathbf{x} V(\mathbf{x}(t))$ and $\dot{\mathbf{x}}(t)$ which yields:

$$\dot{V}(\mathbf{x}(t)) = (\nabla_\mathbf{x} V)^T \dot{\mathbf{x}} = -\mathbf{x}^T \left[\left(\mathcal{B}W\mathcal{L}_\mathcal{E}^d W\mathcal{B}^T\right) \otimes I_d\right] \mathbf{x} \tag{1}$$

where $W(\mathbf{x}(t)) = \mathbf{Diag}\left(\left[a_{e_1}, \ldots, a_{e_{|\mathcal{E}|}}\right]\right)$ with $a_{e_k}(\|x_{e_k}(t)\|) \in \mathbb{R}$ a smooth, time-varying scalar weight function that can take arbitrary values for edges $e_k \in \mathcal{E}$. With the time-varying weight matrix W, which comes from each V_{ij}, proving stability means that it is necessary to prove that for *every* matrix W encoding relative controls at any time t, the asymmetric matrix $\mathcal{B}W\mathcal{L}_\mathcal{E}^d W\mathcal{B}^T \in \mathbb{R}^{n\times n}$ is positive semidefinite. Thus, we want to know the conditions under which *any* weight matrix W yields a stable system (in a Lyapunov sense), which begins with the following result.

Lemma 1. *(Lemma 3.1 in [12]) For any* $W = \mathbf{Diag}\left(\left[a_{e_1}, \ldots, a_{e_{|\mathcal{E}|}}\right]\right) \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$ *associated with* $\mathcal{G}$*, there exists* $\widehat{W}$ *such that* $\left(W\mathcal{B}^T \otimes I_d\right) = \left(\mathcal{B}^T\widehat{W} \otimes I_d\right)$ *whenever* $\mathcal{B}^T\mathcal{B} \triangleq \mathcal{L}_\mathcal{E}$ *is invertible.*

Next, by applying Lemma 1 to (1), we obtain:

$$\dot{V} = -\mathbf{z}^T \left[\underbrace{\left(\mathcal{L}\mathcal{L}_d^T\right)}_{S} \otimes I_d\right] \mathbf{z} \tag{2}$$

with $\mathbf{z} = (\widehat{W} \otimes I_d)\mathbf{x} \in \mathbb{R}^{nd}$ and $S = \mathcal{L}\mathcal{L}_d^T \in \mathbb{R}^{n\times n}$ the *structural Lypaunov matrix*. With the above result, [12] demonstrates that generally stable asymmetric topologies are a subset of directed spanning trees, where it is important to note that *not all spanning trees are stable*. As $\dot{V}$ is in a typical quadratic form, it follows that the stability of our system ($\dot{V} \leq 0$) depends on the properties of S. However, as the S matrix is asymmetric in nature, the typical algebraic definition of positive semidefiniteness does not apply (i.e., non-negativity of eigenvalues). Hence, the Lyapunov stability analysis is carried out on the symmetrized S, $S_+ = \frac{1}{2}(S+S^T)$, as positive semidefiniteness of S_+ implies positive semidefiniteness of S [25]. Therefore, assuming Lemma 1 holds and $S_+ \succeq 0$, a multi-robot system is stable in the sense that if V is initially finite it remains finite for all time $t > 0$ as detailed in Theorem 3.1 [12]. This stability can then be exploited to guarantee various coordination objectives, such as swarming, formation control, topology control, etc.

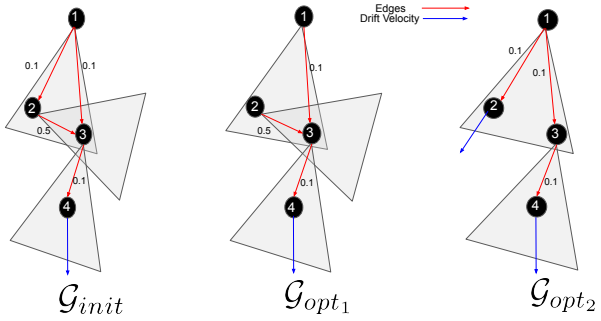


Fig. 1: Four robots in limited FOV interactions (represented by gray triangles) over 3 different topologies. Here $\mathcal{G}_{init}$ is the initial interaction graph, $\mathcal{G}_{opt_1}$ is a potential optimal and stable interaction graph, and $\mathcal{G}_{opt_2}$ with minimal total edge cost, is an unstable interaction graph.

B. Mixed Integer Semidefinite Programs

A general form of MISDP [26] which will facilitate our methods in Section IV is given here. For a set of the first n positive integers $[n] := \{1, ..., n\}$, the general MISDP form is:

$$\begin{aligned} \min \quad & C \bullet X & & \\ \text{s.t.} \quad & A_i \bullet X = b_i & \forall\, i \in [m] & \\ & X \succeq 0, \quad X_{ij} \in \mathbb{Z} & \forall\, (i,j) \in J & \end{aligned} \tag{3}$$

where $C \bullet X = \sum_{i=1}^n \sum_{j=1}^n C_{ij}X_{ij}$ with $C \in \mathbb{S}^n$, $X \in \mathbb{S}^n$, $A_i \in \mathbb{S}^n$, $b \in \mathbb{R}^m$, $[m] = \{1, ..., m\}$ is a set of the first m positive integers, $J \subseteq [n] \times [n]$, $\mathbb{S}^n$ is the set of symmetric matrices of size n, and $\mathbb{Z}$ is the set of integers.

III. Motivating Example

We motivate this work with an example of stable coordination of multi-robot systems with asymmetric interactions [27], [28], where a team of unmanned aerial vehicle (UAVs) achieves leader following (also demonstrated in the attached video[3]). Figure 1 shows four UAVs with front stereo cameras, forming a connected, directed interaction graph denoted by $\mathcal{G}_{init}$ where the interaction quality is denoted by a cost on the

[3]Note that the video focuses on the coordinated motion that can be achieved *after* an optimal topology is selected, which we use as motivation. Insight into our optimization method is instead given in simulations shown in Section V.

edges represented by red arrows. Such a weighting of robot-to-robot interactions can be captured, for instance during outdoor multi-robot experiments [27], as the varying quality of camera images that each robot uses for relative control. Here, $\mathcal{G}_{init}$ demonstrates a leader-follower approach with robot 4 as the leader, and $\mathcal{G}_{opt_1}$ and $\mathcal{G}_{opt_2}$ are two potential spanning trees in $\mathcal{G}_{init}$. In terms of cost alone, the best interaction graph is $\mathcal{G}_{opt_2}$(minimum spanning tree) with a total interaction cost (edge cost) lower than that of $\mathcal{G}_{opt_1}$. However, our work in [12] classifies $\mathcal{G}_{opt_2}$ to be an unstable graph and $\mathcal{G}_{opt_1}$ to be a stable graph. Intuitively this can be explained using the example in Figure 1. In $\mathcal{G}_{opt_2}$, robot 2 and robot 4 have no outgoing interactions, and thus they only move according to a drift velocity (blue arrow). Now if robots 2 and 4 move in different directions, it will be difficult for robot 1 to maintain interactions with robots 2 and 3 to keep the graph connected. Indeed, this system is degenerate and an instability may occur as robots 2 and 4 drift away. On the other hand, with topology $\mathcal{G}_{opt_1}$, the system maintains stability as robot 4 drifts with a certain velocity and all other robots follow suit. Note again that the stable topology $\mathcal{G}_{opt_1}$ is a spanning tree, but not all spanning trees (e.g., $\mathcal{G}_{opt_2}$), or even all single-rooted spanning trees, are stable with respect to general coordinated objectives [12], which renders our OTSP non-trivial.

IV. Optimal Stable Topology Selection

A. General Formulation

The high-level problem of selecting an optimal directed motion control graph $\mathcal{G}^c \subseteq \mathcal{G}$ can be expressed as follows:

$$\begin{aligned} \min_{\mathcal{G}^c \triangleq (\mathcal{V}, \mathcal{E}^c) \subseteq \mathcal{G}} \quad & f(\mathcal{G}^c) \\ \text{s.t.} \quad & \text{(i)} \quad \mathbf{rank}(\mathcal{L}_{\mathcal{E}^c}) = |\mathcal{E}^c|, \quad \text{(ii)} \quad S_+(\mathcal{E}^c) \succeq 0 \end{aligned} \tag{4}$$

where $\mathcal{L}_{\mathcal{E}^c} \in \mathbb{R}^{|\mathcal{E}^c| \times |\mathcal{E}^c|}$ is the edge Laplacian with respect to the edge set, $\mathcal{E}^c$, of the control graph, $f(\cdot) : \mathbb{G}_d^n \to \mathbb{R}$ is an objective function that measures the cost (quality) of a set of control interactions $\mathcal{G}^c$ with $\mathbb{G}_d^n$ the space of all directed graphs of size n, and $S_+(\mathcal{E}^c)$ is the symmetrized structural Lyapunov matrix of the directed control graph. The high-level formulation in (4) seeks an optimal control graph $\mathcal{G}^c$ that ensures: (1) Lemma 1, a stability requirement achieved by constraint (i), is satisfied; and (2) $\dot{V}$ must be negative semidefinite to guarantee stable motion (according to (2)), which is achieved by constraint (ii).

B. Optimal Topology Selection as an MISDP

In the following subsections we will derive an MISDP formulation [20]–[22], a type of semidefinite programming (SDP) problem [29]–[31], that solves our problem (4). The high-level road map for our formulation is: (1) we recognize that as S_+ is described by a *product* of matrices we aim to optimize, the semidefiniteness constraint on S_+ does not fit into an MISDP directly; (2) we thus *relax* the constraint on S_+ by applying *semidefinite relaxation* (SDR) techniques [32]; (3) SDR techniques have been shown to work best when coupled with *Rank Minimization Problem* (RMP) techniques [33], [34] and thus as a first step we apply the so-called trace heuristic method [35] (an RMP) which gives an approximate solution with tight SDR approximation as rank is minimized; and (4) to achieve a final feasible and locally optimal solution we use the solution from a trace heuristic as an initialization to an Iterative Rank Minimization (IRM) problem [33], [34]. Importantly, we point out that in the sequel we combine several theoretical results from the above mentioned works to yield a novel solution to our problem. Indeed, the contexts of [33]–[35] are far different than our problem setting and thus all results shown below are contextualized to our OTSP, with proofs eliminated due to space limitations.

1) Graph Related Constraints: In order to obtain solutions that correctly represent directed graphs, first we define constraints (5)–(8) to represent an adjacency matrix in the form of a binary matrix $\Gamma \in \{0,1\}^{n \times n}$, $\Gamma_{ij} = \gamma_{ij}$, where $\gamma_{ij} = 1 \iff (i,j) \in \mathcal{E}^c$. Constraints (7) and (8) ensure the selected graph has no cycles and consists of edges necessary for a spanning tree, respectively. These two constraints have replaced constraint $\mathbf{rank}(\mathcal{L}_{\mathcal{E}^c}) = |\mathcal{E}^c|$ in (4) by exploiting two ideas : i) a spanning tree is weakly connected which is a necessary condition for coordination ; and ii) $\mathcal{L}_{\mathcal{E}^c}$ needs to be full rank (Lemma 1) but loses rank when loops are present [36].

$$\gamma_{ij} \in \{0,1\} \qquad i,j = 1,...,n, \tag{5}$$

$$\gamma_{ii} = 0 \qquad i = 1,...,n, \tag{6}$$

$$\sum_{(i,j) \in \mathcal{E}^c(\mathcal{V}')} \gamma_{ij} \leq |\mathcal{V}'| - 1, \quad \forall\, \mathcal{V}' \subset \mathcal{V}, \mathcal{V}' \neq \emptyset \tag{7}$$

$$\sum_{(i,j) \in \mathcal{E}^c} \gamma_{ij} = n - 1 \tag{8}$$

Next, constraints (9)–(15) are applied to allow us to represent correct undirected and directed Laplacian matrices.

$$l_{ij}^d \leq 0, \qquad i,j = 1,...,n, i \neq j \tag{9}$$

$$l_{ij}^d < (1 - \gamma_{ij}), \qquad i,j = 1,...,n, i \neq j \tag{10}$$

$$l_{ij}^d \geq M_l(\gamma_{ij}), \qquad i,j = 1,...,n, i \neq j \tag{11}$$

$$v > 0 \tag{12}$$

$$\alpha \mathbf{1}^n \mathbf{1}^{n^T} - v I_n + \mathcal{L} \succeq 0 \tag{13}$$

$$\mathcal{L}_d \mathbf{1}^n = \mathbf{0}^n \tag{14}$$

$$\mathcal{L} = \frac{\mathcal{L}_d + \mathcal{L}_d^T}{2} \tag{15}$$

Constraint (9) implies that all non-diagonal entries of $\mathcal{L}_d$ are non-positive for all $i,j \in \mathcal{V}$ which is a general property of the Laplacian matrix. The variable transformation constraints (10) and (11) in the form of inequalities, where l_{ij}^d is the entry of $\mathcal{L}_d$ located at the i^{th} row and j^{th} column of $\mathcal{L}_d$ and $M_l < 0$ is an arbitrarily small value, are obtained using Theorem 1 below and define the relationship between adjacency matrix elements and directed Laplacian matrix elements.

Theorem 1. *(Theorem 4.1 in [20]) For each $l_{ij}^d \leq 0, i,j = 1,...,n, i \neq j$ introduce a binary variable $\gamma_{ij} \in \{0,1\}$ and inequalities* (9)–(11)*. Assuming an arbitrarily small value of $M_l < 0$, $l_{ij}^d = 0 \Leftrightarrow \gamma_{ij} = 0$, $l_{ij}^d < 0 \Leftrightarrow \gamma_{ij} = 1$ and all inequalities remain valid.*

Constraints (12) and (13), derived in [20], are a direct application of the Courant-Fischer theorem. These two constraints are typically used to replace non-linear eigenvalue constraints on the second smallest eigenvalue (algebraic connectivity) of the undirected graph Laplacian matrix in a linear matrix inequality (LMI) form. Further, constraint (14) ensures that the directed Laplacian matrix has one zero eigenvalue and using $\mathcal{L}_d$ the undirected graph Laplacian $\mathcal{L}$ is expressed as shown in (15). In the next subsection we will discuss constraints related to the *structural Lyapunov matrix* S and consequently S_+.

2) Constraints for S matrix using SDRs with RMP: In order to achieve the constraint $S_+ \succeq 0$, by definition we must first ensure $S = \mathcal{L}\mathcal{L}_d^T$. However, such a non-linear matrix equality constraint cannot be used in an MISDP formulation directly. Thus, we will relax the equality constraint $S = \mathcal{L}\mathcal{L}_d^T$ using SDR techniques. Specifically, we apply a specific set of Schur complements to relax equality constraints of the above form to LMIs. Notably, tight approximations by relaxed constraints occurs when low rank solutions of the SDR are obtained via RMP techniques as shown in [32] (with Proposition 1 given in the sequel for our specific problem). Therefore, to solve our problem with SDR constraints such that they are low rank, we will couple our MISDP problem with an RMP method known as the trace heuristic method [35].

Equation (16) with constraints (17)–(20) shows how we formulate the trace heuristic RMP. Note that the formulation below only highlights the details of the SDRs and trace heuristic formulation for clarity. In reality, and as shown later, $\mathcal{L}, \mathcal{L}_d$ below are variables of optimization and their associated constraints discussed previously would also appear in the partial formulation below.

$$\min_{S\in\mathbb{R}^{n\times n}, Y,Z\in\mathbb{S}^n, t\in\mathbb{R}} \quad t \tag{16}$$

$$\text{s.t.} \quad \textbf{trace}(Y) + \textbf{trace}(Z) \leq 2t \tag{17}$$

$$\mathcal{Y} = \begin{bmatrix} Y & \mathcal{L} \\ \mathcal{L}^T & I_n \end{bmatrix} \succeq 0 \tag{18}$$

$$\mathcal{Z} = \begin{bmatrix} Z & \mathcal{L}_d \\ \mathcal{L}_d^T & I_n \end{bmatrix} \succeq 0 \tag{19}$$

$$M = \begin{bmatrix} Y & S \\ S^T & Z \end{bmatrix} \succeq 0 \tag{20}$$

First, to ensure $S = \mathcal{L}\mathcal{L}_d^T$, constraint (20) should satisfy the form in (21).

$$\begin{bmatrix} Y & S \\ S^T & Z \end{bmatrix} = \begin{bmatrix} \mathcal{L}\mathcal{L}^T & \mathcal{L}\mathcal{L}_d^T \\ \mathcal{L}_d\mathcal{L}^T & \mathcal{L}_d\mathcal{L}_d^T \end{bmatrix} = \begin{bmatrix} \mathcal{L} \\ \mathcal{L}_d \end{bmatrix} \begin{bmatrix} \mathcal{L}^T & \mathcal{L}_d^T \end{bmatrix} \succeq 0 \tag{21}$$

To realize the relation in (21) we will first obtain constraints that will solve for $Y = \mathcal{L}\mathcal{L}^T$ and $Z = \mathcal{L}_d\mathcal{L}_d^T$. To solve for $Y = \mathcal{L}\mathcal{L}^T$ and $Z = \mathcal{L}_d\mathcal{L}_d^T$, we can apply rank constraints of the form $\textbf{rank}(\mathcal{Y}) \leq n$ and $\textbf{rank}(\mathcal{Z}) \leq n$ as given by the following proposition.

Proposition 1. *(Proposition 2 in [34]) $Y = \mathcal{L}\mathcal{L}^T$ or $Z = \mathcal{L}_d\mathcal{L}_d^T$ is equivalent to $\textbf{rank}(\mathcal{Y}) \leq n$ or $\textbf{rank}(\mathcal{Z}) \leq n$ where $Y, Z, \mathcal{L} \in \mathbb{S}^n$,$\mathcal{L}_d \in \mathbb{R}^{n\times n}$.*

However, rank constraints $\textbf{rank}(\mathcal{Y}) \leq n$ and $\textbf{rank}(\mathcal{Z}) \leq n$ are highly non-linear and discontinuous. So we will relax these rank constraints and use SDR constraints (18) and (19) instead. The result of Proposition 1 (proof in [34]) reiterates the point mentioned earlier that for SDR of type (18) with dimensions $\mathcal{Y} \in \mathbb{R}^{2n\times 2n}$, tight approximations are only obtained at low ranks when $\textbf{rank}(\mathcal{Y}) = n$, which applies to constraint (19) as well. Therefore, we will have to solve the trace heuristic RMP to obtain low rank solutions of these SDRs. The trace heuristic formulation in (16) with constraint (17) is clearly minimizing functions $\textbf{trace}(Y)$ and $\textbf{trace}(Z)$ with the implication that minimizing the trace of positive semidefinite matrices like Y and Z is equivalent to nuclear norm minimization, and hence minimizing the ranks of Y and Z and consequently the rank of their positive semidefinite block matrices $\mathcal{Y}$ in (18) and $\mathcal{Z}$ in (19), respectively, as also derived in Theorem 1 in [35] .

Now that we have obtained constraints that will help us obtain Y and Z, we need to apply constraints such that $S = \mathcal{L}\mathcal{L}_d^T$ and M looks like (21). Since S is an asymmetric matrix by nature as it represents properties of a directed graph, to apply constraints on S in an MISDP problem we need to embed it in a symmetric matrix first. Therefore, Proposition 1 cannot be directly applied to S. However, we will apply results from [35] to show that the trace heuristic RMP can be utilized to minimize the rank of S by first embedding S in a positive semidefinite SDR type constraint (20). This is possible by invoking semidefinite embedding Lemma 2 to embed S, an asymmetric matrix, in a positive semidefinite matrix M as in constraint (20) or the form in (21).

Lemma 2. *(Lemma 1 in [35]) $S = \mathcal{L}\mathcal{L}_d^T$ and $\textbf{rank}(S) \leq n$ if and only if there exists $Y = \mathcal{L}\mathcal{L}^T \in \mathbb{S}^n$ and $Z = \mathcal{L}_d\mathcal{L}_d^T \in \mathbb{S}^n$ such that $\textbf{rank}(Y) + \textbf{rank}(Z) \leq 2n$ and $M \succeq 0$.*

The semidefinite embedding of S in a larger block matrix M in Lemma 2 implies that there is a direct relation between the rank of the S matrix and the ranks of Y and Z. Note, we already know that minimizing the trace of positive semidefinite matrices like Y and Z over the variable t is equivalent to nuclear norm minimization, and hence minimizing the ranks of Y and Z. Invoking Lemma 2, we understand that minimizing over a variable t, as done in the trace heuristic formulation in (16) implies minimizing the *nuclear norm* $\|S\|_*$, which further implies minimizing the convex envelope of the rank function $\textbf{rank}(S)$ as explained by Theorem 1 from [35]. Therefore, the trace heuristic problem in (16) with the constraints, (17) and (20), solve for $\textbf{rank}(S) \leq n$ and obtain $S = \mathcal{L}\mathcal{L}_d^T$.

At this point we remind the reader that the MISDP formulation so far uses a trace heuristic which gives an approximate solution with tight SDR approximation as the ranks of the SDR constraints are minimized. However, to solve for the exact solution we will implement an IRM method as described in Section IV-C. In the formulation of the IRM problem there exists a specific constraint (27). The IRM method solves this constraint at every iteration using information from the previous iteration. Therefore, the IRM problem in (26) needs the constraint (27) to be initialized. For the first iteration of the IRM, we will initialize the constraint by including it in

the trace heuristic RMP. Thus, with this goal we apply the following constraint:

$$\tilde{\mathcal{M}} = \begin{bmatrix} I_n & \mathcal{Q}^T \\ \mathcal{Q} & M \end{bmatrix} \succeq 0 \quad \mathcal{Q} = \begin{bmatrix} \mathcal{L} \\ \mathcal{L}_d \end{bmatrix} \tag{22}$$

Exploiting Proposition 1, $M = \mathcal{Q}\mathcal{Q}^T$ and thus (21) holds when $\textbf{rank}(\tilde{\mathcal{M}}) \leq n$. Note that, again for $\tilde{\mathcal{M}} \in \mathbb{S}^{3n}$ a low rank solution, $\textbf{rank}(\tilde{\mathcal{M}}) \leq n$, is desired for the SDR. Like prior constraints, the intention of using SDR constraint (22) is to obtain the form in (21) by relaxing the rank constraint. However, unlike previous cases, we will apply an iterative RMP to obtain a reduced rank solution for (22) as a result of Corollary 1 in Section IV-C. The iterative RMP will not solve for an approximate solution of the constraint $\tilde{\mathcal{M}}$ but converge to the exact solution by minimizing the rank of the SDR constraint $\tilde{\mathcal{M}}$ such that $\textbf{rank}(\tilde{\mathcal{M}}) = n$ and $M = \mathcal{Q}\mathcal{Q}^T$. Hence, from Corollary 1, it will be clear that constraint (22) has to be initialized first in the trace heuristic problem (16) for it to be used in consecutive iterations in iterative RMP described in Section IV-C.

Finally, we apply constraints to satisfy the graph stability condition, $S_+ \succeq 0$. Constraint (23) expresses S_+ in terms of S and (24) is a constraint on the positive semidefiniteness of S_+.

$$S_+ = \frac{S + S^T}{2} \tag{23} \qquad S_+ \succeq 0 \tag{24}$$

3) OTSP Formulated As MISDP Coupled With Trace Heuristic RMP: With all essential constraints defined, we now present the complete MISDP problem formulation coupled with the trace heuristic RMP method as

$$\min_{\substack{\Gamma \in \{0,1\}^{n\times n}, \\ \mathcal{L}_d, S \in \mathbb{R}^{n\times n}, t, v, \alpha \in \mathbb{R}, \\ \mathcal{L}, Y, Z, S_+ \in \mathbb{S}^n}} \quad (1-\beta) \sum_{i,j=1, i\neq j}^{n} w_{ij}\gamma_{ij} + \beta t \tag{25}$$

$$\text{s.t.} \quad eqs : (5) - (15), (17) - (20), (22) - (24)$$

The first part of the objective function in (25), $\sum_{i,j=1,i\neq j}^{n} w_{ij}\gamma_{ij}$, minimizes the summation of the product of the elements of the given weight matrix $W \in \mathbb{R}^{n\times n}$, w_{ij}, and the elements of the binary adjacency matrix Γ, γ_{ij}. The elements of weight matrix W encode the quality of interaction as a cost. We then want to minimize this cost of interaction over stable graphs as motivated in Section III. The second part with the optimization variable t is for the trace heuristic in (16). These terms are then combined convexly through a weight factor $\beta \in (0, 1)$. Thus, the solution space of the objective function in (25) lies on a convex Pareto curve [37]. The objective here is not to obtain a Pareto optimal solution as the graph cost is a priority, hence we always solve the bi-objective MISDP in (25) with a small value of β initially. However, β values will be incremented if an infeasible solution is detected in the iterative RMP method as described in Algorithm 1 in Section IV-C.

The MISDP formulation above is ultimately a *heuristic* which implies that the solution may not be optimal or even feasible. Thus, the solution from problem (25) is used as an initialization to an *Iterative Rank Minimization (IRM) Problem* [34] for convergence to the solution.

C. Iterative Rank Minimization for the OTSP

The idea of IRM is to first obtain an initial solution from the trace heuristic method in (25), and then use that initial solution as initial conditions for the IRM method to converge to the locally optimal and feasible solution to our OTSP. This is required because the IRM method solves a trace heuristic RMP like the one in (16) at every iteration with the added constraint (27). Thus, for the first iteration of the IRM, part of constraint (27) must be initialized in the trace heuristic RMP (25).

The IRM problem is directly derived from the trace heuristic in (16) with the added constraint (27).

$$\min_{\substack{t_k, e_k \in \mathbb{R}, \tilde{\mathcal{M}}^k \in \mathbb{S}^{3n}, \\ S_k, Y_k, Z_k, S_{+_k} \in \mathbb{R}^{n\times n}}} \quad t_k + \omega\, e_k \tag{26}$$

$$\text{s.t.} \quad S_k \in \mathcal{C}$$

$$eqs : (17) - (20), (22) - (24)$$

$$e_k I_{2n} - V_{k-1}^T \tilde{\mathcal{M}}^k V_{k-1} \succeq 0, \tag{27}$$

where k is the iteration step, $\omega > 1$ is weight factor, $\mathcal{C}$ is some convex set, and $V_{k-1} \in \mathbb{R}^{3n\times 2n}$ are the eigenvectors corresponding to the $2n$ smallest eigenvalues of $\tilde{\mathcal{M}}$ solved at previous iteration $k-1$. Therefore, for the IRM problem, we use the obtained V from $\tilde{\mathcal{M}}$, $\mathcal{L}$ and $\mathcal{L}_d$ from (25) to solve the IRM optimization problem in (26) primarily with the goal that $\tilde{\mathcal{M}}$ satisfies minimal rank conditions such that $M = \mathcal{Q}\mathcal{Q}^T$. Constraint (27), as per *Corollary 3.2* in [33] and Corollary 1, is a replacement of rank constraint $\textbf{rank}(\tilde{\mathcal{M}}) \leq n$.

Corollary 1. *(Corollary 3.2 in [33]) When $e = 0$ and $\tilde{\mathcal{M}} \in \mathbb{S}^{3n}$ is a positive semidefinite matrix, then $\textbf{rank}(\tilde{\mathcal{M}}) \leq n$ if and only if $eI_{2n} - V^T\tilde{\mathcal{M}}V \succeq 0$ where $V \in \mathbb{R}^{3n\times 2n}$ are the eigenvectors corresponding to the $2n$ smallest eigenvalues of $\tilde{\mathcal{M}}$.*

From Corollary 1 we clearly see that the application of constraint (22) in the initial MISDP problem is required to obtain V at iteration $k = 0$ for iteration $k = 1$ of IRM as also shown in Algorithm 1. We now invoke Lemma 3 below and notify the reader that unlike in [33], [34] where $\mathcal{Q}$ in the context of the IRM problem is unknown, in this paper $\mathcal{Q}$ is derived from solving optimization problem (initial trace heuristic) (25) for $\mathcal{L}$ and $\mathcal{L}_d$ which represents the selected topology.

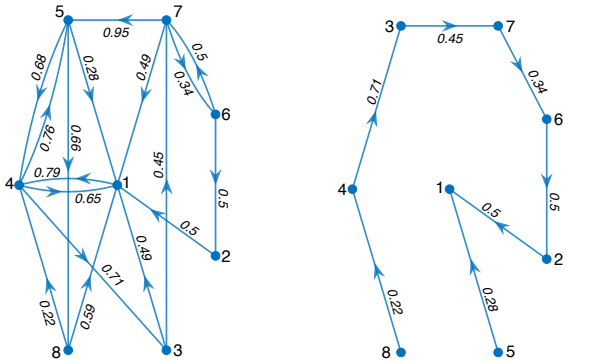
Lemma 3. *(Lemma 3 in [34]) Let $\mathcal{Q} \in \mathbb{R}^{2n\times n}$ be a given matrix. Then $\textbf{rank}(\mathcal{Q}) \leq n(= n)$ if and only if there exists $M \in \mathbb{S}^{2n}$ such that $\textbf{rank}(M) \leq n(= n)$ and $\textbf{rank}(\tilde{\mathcal{M}}) \leq n$.*

Algorithm 1 provides steps to the complete optimization problem.

Remark 1. For the sake of brevity we have omitted the proof of convergence of the IRM method but it follows directly from [33] because the form of the primal IRM problem in (26) has affine equality constrains with all other constraints also affine which satisfies Slater's refined condition [37] for strong duality which is a required condition used in [33] to prove convergence.

Algorithm 1 : Iterative OTSP

Input: $\beta, W, n, M_l, \epsilon, fail = 1$
Output: Γ^*
while $fail = 1$ **do**
  if $iter > 1$ **then**
    $\beta = \beta + 0.1$ and initialize all variables
  end if
  Initialize Set $k = 0$, solve **trace heuristic** problem
  in (25) to obtain V_0 from $\tilde{\mathcal{M}}^0$, $\mathcal{L}$ and $\mathcal{L}_d$
  Set $k = k + 1$
  while $k \leq k_{max}$ || $e_k \geq \epsilon$ **do**
    Solve the **IRM** subproblem (26) and obtain $\tilde{\mathcal{M}}^k$
    Update V_k from $\tilde{\mathcal{M}}^k$
    $k = k + 1$
  end while
  if $S_+ \succeq 0$ **then**
    $fail = 0$
  end if
  $iter = iter + 1$
end while
Find Γ^*



(a) Initial graph of robot interactions. (b) Optimally selected graph.

Fig. 2: Optimal selection of graph (b) based on initial graph (a) of $n = 8$ robot interactions.

V. Simulation Results

The optimization problem shown in Algorithm 1 is solved using YALMIP [38] with the SeDuMi [39] SDP solver in MATLAB. Given a directed graph describing asymmetric robot interactions, as shown in Figure 2a where the weighted edges represent quality of interaction, Figure 2b shows the optimally selected stable graph. The eigenvalues for the computed S_+ matrix for the optimally selected graph are $\lambda(S_+) = \{0, 0.02, 0.24, 0.97, 1.92, 3.25, 5.36, 7.25\}$ for $n = 8$.

Remark 2. Note that the selected optimal control graph $\mathcal{G}^c$ will always be a spanning tree as in Figure 2b. While this may seem limiting from a control perspective, there are two points to be made. First, although a subset of sensed interactions are used for control, the entire set of interactions $\mathcal{E}$ can still be exploited for other purposes, such as collaborative localization. Second, the theory from [12] which defines stable topologies is very general as it holds for controllers derived from *any* energy functions V_{ij}. For specific coordinated motion controllers, and thus particular choices for the V_{ij}, there will be a much richer class of stable graphs to which this paper's methods will apply with minimal changes.

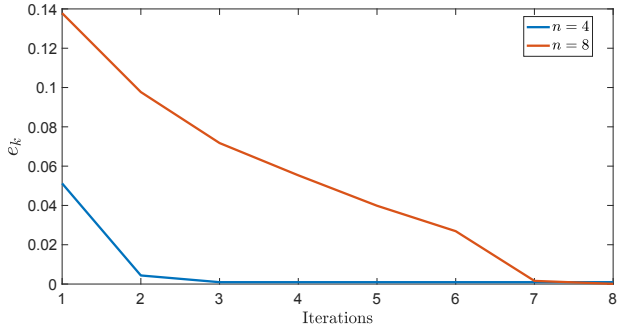


Fig. 3: Convergence of e_k for graphs with nodes $n = 4$ (blue) and $n = 8$ (red).

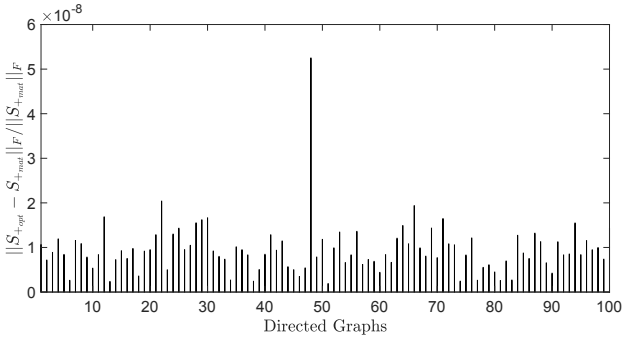


Fig. 4: Relative error $\|S_{+_{opt}} - S_{+_{mat}}\|_F / \|S_{+_{mat}}\|_F$ for 100 random directed graphs with $n = 8$ nodes.

Next, Figure 3 shows the convergence of e_k of the IRM algorithm. This plot shows convergence within eight iterations of the IRM algorithm for graphs with $n = 8$ and a different simulation conducted for comparison, with $n = 4$. Finally, we conducted a Monte Carlo simulation over one hundred random directed graphs with $n = 8$ and the results for the relative error at convergence for each graph is presented in Figure 4, indicating the correctness of our method. Relative error is defined by relative Frobenius norm and denoted as $\|S_{+_{opt}} - S_{+_{mat}}\|_F / \|S_{+_{mat}}\|_F$ for optimally computed $S_{+_{opt}}$ using Algorithm 1 and the MATLAB computed $S_{+_{mat}}$.

VI. Conclusions

In this paper, we solved the problem of selecting optimal topologies for stable coordination in multi-robot systems using a MISDP formulation coupled with RMP techniques. Finally, we provided numerical results that demonstrate the performance of the OTSP. In the future, we will further develop a distributed method of OTSP and apply it to our portable experimental setup [27].

References

[1] G. Bevacqua, J. Cacace, A. Finzi, and V. Lippiello, "Mixed-initiative planning and execution for multiple drones in search and rescue missions." in *ICAPS*, 2015, pp. 315–323.

[2] E. Guizzo, "Three engineers, hundreds of robots, one warehouse," *IEEE spectrum*, vol. 45, no. 7, pp. 26–34, 2008.

[3] D. E. Koditschek and E. Rimon, "Robot navigation functions on manifolds with boundary," *Advances in Applied Mathematics*, vol. 11, no. 4, pp. 412 – 442, 1990.

[4] M. M. Zavlanos and G. J. Pappas, "Potential fields for maintaining connectivity of mobile networks," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 812–816, Aug 2007.

[5] M. Ji and M. Egerstedt, "Distributed coordination control of multiagent systems while preserving connectedness," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693–703, Aug 2007.

[6] D. V. Dimarogonas and K. J. Kyriakopoulos, "Connectedness preserving distributed swarm aggregation for multiple kinematic robots," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1213–1223, Oct 2008.

[7] M. M. Zavlanos, H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Hybrid control for connectivity preserving flocking," *IEEE Transactions on Automatic Control*, vol. 54, no. 12, pp. 2869–2875, Dec 2009.

[8] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 863–868, May 2007.

[9] R. K. Williams and G. S. Sukhatme, "Constrained Interaction and Coordination in Proximity-Limited Multi-Agent Systems," *IEEE Transactions on Robotics*, vol. 29, pp. 930–944, 2013.

[10] R. K. Williams, A. Gasparri, G. S. Sukhatme, and G. Ulivi, "Global connectivity control for spatially interacting multi-robot systems with unicycle kinematics," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1255–1261.

[11] A. Gasparri, L. Sabattini, and G. Ulivi, "Bounded control law for global connectivity maintenance in cooperative multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 700–717, June 2017.

[12] P. Mukherjee, A. Gasparri, and R. K. Williams, "Stable motion and distributed topology control for multi-agent systems with directed interactions," in *Decision and Control (CDC), 2017 IEEE 56th Annual Conference on*, 2017.

[13] A. D. Domínguez-García and C. N. Hadjicostis, "Distributed strategies for average consensus in directed graphs," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 2124–2129.

[14] T. Qi, L. Qiu, and J. Chen, "Consensus over directed graph: Output feedback and topological constraints," in *2013 9th Asian Control Conference (ASCC)*. IEEE, 2013, pp. 1–6.

[15] M. M. Asadi, A. Ajorlou, and A. G. Aghdam, "Distributed control of a network of single integrators with limited angular fields of view," *Automatica*, vol. 63, pp. 187–197, 2016.

[16] L. Sabattini, C. Secchi, and N. Chopra, "Decentralized estimation and control for preserving the strong connectivity of directed graphs," *IEEE Trans Cybern*, vol. 45, no. 10, pp. 2273–2286, Oct. 2015.

[17] P. Yang, R. A. Freeman, G. J. Gordon, K. M. Lynch, S. S. Srinivasa, and R. Sukthankar, "Decentralized estimation and control of graph connectivity for mobile sensor networks," *Automatica*, vol. 46, no. 2, pp. 390–396, 2010.

[18] R. K. Williams and G. S. Sukhatme, "Locally constrained connectivity control in mobile robot networks," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 901–906.

[19] A. Gasparri, R. K. Williams, A. Priolo, and G. S. Sukhatme, "Decentralized and Parallel Constructions for Optimally Rigid Graphs in R^2," *IEEE Transactions on Mobile Computing*, vol. 14, no. 11, pp. 2216–2228, Nov. 2015. [Online]. Available: http://ieeexplore.ieee.org/document/7018921/

[20] M. Rafiee and A. M. Bayen, "Optimal network topology design in multi-agent systems for efficient average consensus," in *49th IEEE Conference on Decision and Control (CDC)*, Dec 2010, pp. 3877–3883.

[21] D. Xue, A. Gusrialdi, and S. Hirche, "A distributed strategy for near-optimal network topology design," in *21st International Symposium on Mathematical Theory of Networked and Systems (MTNS 2014)*, 2014.

[22] R. K. Ramachandran, J. A. Preiss, and G. S. Sukhatme, "Resilience by reconfiguration: Exploiting heterogeneity in robot teams," *arXiv preprint arXiv:1903.04856*, 2019.

[23] Z. Zeng, X. Wang, and Z. Zheng, "Edge agreement of multi-agent system with quantised measurements via the directed edge laplacian," *IET Control Theory Applications*, vol. 10, no. 13, pp. 1583–1589, 2016.

[24] D. Zelazo, A. Rahmani, and M. Mesbahi, "Agreement via the edge laplacian," in *2007 46th IEEE Conference on Decision and Control*, Dec 2007, pp. 2309–2314.

[25] C. R. Johnson and R. Reams, "Semidefiniteness without real symmetry," *Linear Algebra and its Applications*, vol. 306, no. 1-3, pp. 203–209, 2000.

[26] T. Gally, M. E. Pfetsch, and S. Ulbrich, "A framework for solving mixed-integer semidefinite programs," *Optimization Methods and Software*, vol. 33, no. 3, pp. 594–632, 2018.

[27] P. Mukherjee, M. Santilli, A. Gasparri, and R. K. Williams, "Experimental validation of stable coordination for multi-robot systems with limited fields of view using a portable multi-robot testbed - extended abstract," in *2019 International Symposium on Multi-Robot and Multi-Agent Systems (MRS)*, Aug 2019, pp. 4–6.

[28] M. Santilli, P. Mukherjee, A. Gasparri, and R. K. Williams, "Distributed connectivity maintenance in multi-agent systems with field of view interactions," in *2019 American Control Conference (ACC)*, July 2019, pp. 766–771.

[29] F. Alizadeh, "Interior point methods in semidefinite programming with applications to combinatorial optimization," *SIAM Journal on Optimization*, vol. 5, pp. 13–51, 1993.

[30] L. Vandenberghe and S. Boyd, "Semidefinite programming," *SIAM Rev.*, vol. 38, no. 1, pp. 49–95, Mar. 1996. [Online]. Available: http://dx.doi.org/10.1137/1038003

[31] M. Todd, K. Toh, and R. Ttnc, "On the nesterov–todd direction in semidefinite programming," *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 769–796, 1998. [Online]. Available: https://doi.org/10.1137/S105262349630060X

[32] Z.-Q. Luo, W.-k. Ma, A. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *Signal Processing Magazine, IEEE*, vol. 27, pp. 20 – 34, 06 2010.

[33] C. Sun and R. Dai, "An iterative approach to rank minimization problems," in *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, 12 2015.

[34] ——, "Rank-constrained optimization and its applications," *Automatica*, vol. 82, pp. 128–136, 2017.

[35] M. Fazel, "Matrix rank minimization with applications," 2002.

[36] D. Zelazo and M. Mesbahi, "Edge agreement: Graph-theoretic performance bounds and passivity analysis," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 544–555, 2010.

[37] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[38] J. Lofberg, "Yalmip : a toolbox for modeling and optimization in matlab," in *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, Sep. 2004, pp. 284–289.

[39] J. F. Sturm, "Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones," *Optimization Methods and Software*, vol. 11, 11 1998.