ISSN 1342-2804

Research Reports on Mathematical and Computing Sciences

Exploiting Sparsity in SDP Relaxation of Polynomial Optimization Problems

Sunyoung Kim and Masakazu Kojima

May 2010, B–462

Department of
Mathematical and
Computing Sciences
Tokyo Institute of Technology

SERIES B: Operations Research

B-462 Exploiting Sparsity in SDP relaxation of Polynomial Optimization Problems

Sunyoung Kim† and Masakazu Kojima‡

May 2010

Abstract.

We present a survey on the sparse SDP relaxation proposed as a sparse variant of Lasserre's SDP relaxation of polynomial optimization problems. We discuss the primal and dual approaches to derive the sparse SDP and SOS relaxations, and their relationship. In particular, exploiting structured sparsity in the both approaches is described in view of the quality and the size of the SDP relaxations. In addition, numerical techniques used in the Matlab package SparsePOP for solving POPs are included. We report numerical results on SparsePOP and the application of the sparse SDP relaxation to sensor network localization problems.

Key words.

Polynomial optimization, Semidefinite programming relaxation, Sparsity exploitation, Matlab software package.

† Department of Mathematics, Ewha W. University, 11-1 Dahyun-dong, Sudaemoon-gu, Seoul 120-750 Korea. S. Kim's research was supported by NRF 2009-007-1314 and NRF 2010-000-8784. *skim@ewha.ac.kr*

‡ Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152-8552 Japan. M. Kojima's research was supported by Grant-in-Aid for Scientific Research (B) 22310089. *kojima@is.titech.ac.jp*

1 Introduction

Polynomial optimization problems (POP) are nonlinear optimization problems whose objective and constraint functions involve only polynomials. They, however, provide a general framework to represent various application problems in science and engineering. In particular, POPs include quadratic optimization problems with or without 0-1 constraints on their variables. POPs are nonconvex in general, and they have served as a unified mathematical model for the study of global optimization of nonconvex continuous and discrete optimization problems. See the book [Las10] and the references therein. A POP is described as

$$\text{minimize} \quad f_0(\boldsymbol{x}) \quad \text{subject to} \quad f_k(\boldsymbol{x}) \geq 0 \ (k = 1, 2, \ldots, m), \tag{1}$$

where $f_k(\boldsymbol{x})$ denotes a polynomial in $\boldsymbol{x} \in \mathbb{R}^n$ $(k = 0, 1, \ldots, m)$. Let f_0^* be the optimal value of the POP (1); f_0^* may be $-\infty$ if the POP (1) is unbounded and $+\infty$ if it is infeasible.

A hierarchy of semidefinite programming (SDP) relaxations proposed by Lasserre in [Las01] is known as a powerful method for computing a global optimal solution of the POP (1). The hierarchy is arranged according to a positive integer, called the relaxation order in this article. It determines qualities and sizes of SDP relaxation problems in the hierarchy. Each SDP problem with a relaxation order ω provides a lower bound ζ_ω^d for the optimal objective value f_0^* of the POP (1). It was established in [Las01] that ζ_ω^d converges f_0^* as $\omega \to \infty$ under a moderate assumption that requires the compactness of the feasible region of (1). The size of the SDP relaxation, however, increases very rapidly as the number of variables of (1), the degree of polynomials involved in (1), and/or the relaxation order ω increase. In practice, it is often difficult to obtain an approximation to the global optimal solution of (1) because the resulting SDP relaxation is too large to solve. Without employing techniques to reduce the size of SDP relaxations, an approximation to the global optimal solution of a medium- to large-scale POP is difficult to obtain. One important technique to cope with this difficulty is exploiting structured sparsity of POPs, which is the subject of this article.

The purpose of this article is to present a survey of the sparse SDP relaxation, which was originally proposed in [WKK06] as a sparse variant of Lasserre's SDP relaxation [Las01]. We call Lasserre's SDP relaxation the dense SDP relaxation. The focus is on the algorithmic aspects of the sparse SDP relaxation. For its theoretical convergence, we refer to [Las06]. Figure 1 shows an overview of the dense and sparse SDP relaxations. We can describe these SDP relaxations in two methods: A primal approach, which is our primary concern in this article, and a dual approach.

In the primal approach, we first choose a relaxation order ω, and then, convert the POP (1) into an equivalent polynomial SDP (PSDP) by adding valid polynomial matrix inequalities. The relaxation order ω is used at this stage to restrict the degree of the valid polynomial matrix inequalities added to at most 2ω. Thus, it controls the size and quality of the SDP relaxation that will be derived. At this stage, we can also incorporate the sparsity characterized by a chordal graph structure to construct a sparse PSDP. After expanding real and matrix-valued polynomial functions in the dense (or sparse) PSDP, the linearization by replacing each monomial by a single real variable is followed to obtain the sparse (or dense) SDP relaxation.

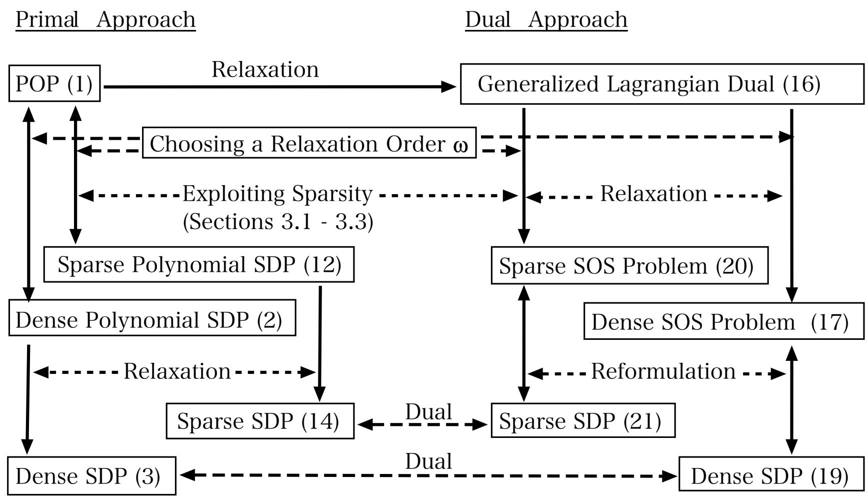

Figure 1: Overview of Lasserre's hierarchy of (dense) SDP relaxations [Las01] and its sparse variant [WKK06].

In the dual approach, a generalized Lagrangian relaxation to the POP (1) is applied to obtain a generalized Lagrangian dual [KKW05, Las01, Put93] of (1) that includes sums of squares (SOS) polynomials for Lagrangian multipliers. After choosing a relaxation order ω, we perform SOS relaxation for the Lagrangian dual. The relaxation order ω is chosen to restrict the degrees of SOS polynomials used there by at most 2ω. As a result, it controls the quality and the size of the SOS relaxation. As in the primal approach, the same structured sparsity can be exploited at this stage to obtain a sparse SOS problem. Finally, the sparse (or dense) SOS relaxation problem is reformulated as a sparse (or dense) SDP. We note that the sparse (or dense) SDP obtained in the primal and dual approaches have a primal-dual relationship.

In Section 2, we describe Lasserre's dense SDP relaxation of the POP (1) after introducing notation and symbols used throughout the article. Section 3 is devoted to the primal approach to (1). In Section 3.1, we present a class of SDP problems having multiple but small-sized matrix variables that can be solved efficiently by primal-dual interior-point methods [Bor99, SDPA, Str99, TTT03, YFN10]. This class of SDPs serves as target SDPs into which POPs are aimed to be relaxed. In Section 3.2, a sparse Cholesky factorization and a chordal graph are introduced. These are used in Section 3.3 for formulating structured sparsity from POPs. The sparse SDP relaxation of the POP (1) is described in Section 3.4. In Section 4, we present the dual approach for the POP (1). Section 5 contains additional techniques used in the software package SparsePOP [WKK08, SPOP], which is an implementation of the sparse SDP relaxation. Numerical results on SparsePOP are reported in Section 6, and the applications of the sparse SDP relaxation [KKW09a, KKW09b, SFSDP] to the sensor network localization are presented in Section 7.

2 Preliminaries

2.1 Notation and Symbols

Throughout the paper, we let $\mathbb{R}$ be the set of real numbers, and $\mathbb{Z}_+$ the set of nonnegative integers. We use $\mathbb{R}^n$ for the n-dimensional Euclidean space, and $\mathbb{Z}^n_+$ for the set of nonnegative integer vectors in $\mathbb{R}^n$. Each element $\boldsymbol{x}$ of $\mathbb{R}^n$ is an n-dimensional column vector of $x_i \in \mathbb{R}$ $(i = 1, 2, \ldots, n)$, written as $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$, and $\boldsymbol{x}^T$ denotes the n-dimensional row vector. Let $\mathbb{S}^n$ denote the space of $n \times n$ real symmetric matrices and $\mathbb{S}^n_+ \subset \mathbb{S}^n$ the cone of positive semidefinite matrices. We write $\boldsymbol{Y} \succeq \boldsymbol{O}$ if $\boldsymbol{Y} \in \mathbb{S}^n_+$ for some n.

$\mathbb{R}[\boldsymbol{x}]$ is the set of real-valued multivariate polynomials in x_i $(i = 1, 2, \ldots, n)$. Each $f \in \mathbb{R}[\boldsymbol{x}]$ is expressed as $f(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{F}} c(\boldsymbol{\alpha})\boldsymbol{x}^{\boldsymbol{\alpha}}$ for some nonempty finite subset $\mathcal{F}$ of $\mathbb{Z}^n_+$ and $c(\boldsymbol{\alpha}) \in \mathbb{R}$ $(\boldsymbol{\alpha} \in \mathcal{F})$, where $\boldsymbol{x}^{\boldsymbol{\alpha}} = x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ for every $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ and every $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in \mathbb{Z}^n_+$. We define the support of f by $\mathrm{supp}(f) = \{\boldsymbol{\alpha} \in \mathcal{F} : c(\boldsymbol{\alpha}) \neq 0\}$ and the degree of f by $\deg(f) = \max\{\sum_{i=1}^n \alpha_i : \boldsymbol{\alpha} \in \mathrm{supp}(f)\}$.

For every nonempty finite subset $\mathcal{G}$ of $\mathbb{Z}^n_+$, let $\mathbb{R}^{\mathcal{G}}$ denote the $|\mathcal{G}|$-dimensional Euclidean space whose coordinates are indexed by $\boldsymbol{\alpha} \in \mathcal{G}$. Each vector of $\mathbb{R}^{\mathcal{G}}$ is denoted as a column vector $\boldsymbol{w} = (w_{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathcal{G})$. The indices $\boldsymbol{\alpha} \in \mathcal{G}$ can be ordered arbitrary except for the element $\mathbf{0} \in \mathbb{Z}^n_+$, which is assumed to be the first index whenever $\mathbf{0} \in \mathcal{G}$. We use the symbol $\mathbb{S}^{\mathcal{G}}$ for the set of $|\mathcal{G}| \times |\mathcal{G}|$ symmetric matrices with coordinates $\boldsymbol{\alpha} \in \mathcal{G}$. Let $\mathbb{S}^{\mathcal{G}}_+$ be the set of positive semidefinite matrices in $\mathbb{S}^{\mathcal{G}}$; $\boldsymbol{V} \in \mathbb{S}^{\mathcal{G}}_+$ iff

$$\boldsymbol{w}^T \boldsymbol{V} \boldsymbol{w} = \sum_{\boldsymbol{\alpha} \in \mathcal{G}} \sum_{\boldsymbol{\beta} \in \mathcal{G}} V_{\boldsymbol{\alpha}\boldsymbol{\beta}} w_{\boldsymbol{\alpha}} w_{\boldsymbol{\beta}} \geq 0 \quad \text{for every } \boldsymbol{w} = (w_{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathcal{G}) \in \mathbb{R}^{\mathcal{G}}.$$

For every nonempty finite subset $\mathcal{G}$ of $\mathbb{Z}^n_+$, let $\boldsymbol{u}(\boldsymbol{x}, \mathcal{G})$ denote the $|\mathcal{G}|$-dimensional column vector of elements $\boldsymbol{x}^{\boldsymbol{\alpha}}$ $(\boldsymbol{\alpha} \in \mathcal{G})$; $\boldsymbol{u}(\boldsymbol{x}, \mathcal{G}) = (\boldsymbol{x}^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathcal{G}) \in \mathbb{R}^{\mathcal{G}}$ for every $\boldsymbol{x} \in \mathbb{R}^n$. Obviously, $\boldsymbol{u}(\boldsymbol{x}, \mathcal{G})\boldsymbol{u}(\boldsymbol{x}, \mathcal{G})^T \in \mathbb{S}^{\mathcal{G}}_+$ for every $\boldsymbol{x} \in \mathbb{R}^n$. For every $\psi \in \mathbb{Z}_+$, let

$$\mathcal{A}_\psi = \left\{ \boldsymbol{\alpha} \in \mathbb{Z}^n_+ : \sum_{i=1}^n \alpha_i \leq \psi \right\}.$$

Then, we see that $\mathbf{0} \in \mathcal{A}_\psi$. For any $\boldsymbol{x} \in \mathbb{R}^n$, the upper left element of the matrix $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\psi)\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\psi)^T \in \mathcal{S}^{\mathcal{A}_\psi}_+$ is the constant $\boldsymbol{x}^{\mathbf{0}} = 1$. We also note that $\mathcal{A}_\psi + \mathcal{A}_\psi = \mathcal{A}_{2\psi}$ for every $\psi \in \mathbb{Z}_+$, where $\mathcal{G} + \mathcal{H}$ denotes the Minkowski sum of two $\mathcal{G}$, $\mathcal{H} \subset \mathbb{Z}^n_+$, *i.e.*, $\mathcal{G} + \mathcal{H} = \{\boldsymbol{\alpha} + \boldsymbol{\beta} : \boldsymbol{\alpha} \in \mathcal{G},\ \boldsymbol{\beta} \in \mathcal{H}\}$. These facts are used in the next subsection.

We represent each polynomial f_k in the POP (1) as

$$f_k(\boldsymbol{x}) = \sum_{\boldsymbol{\alpha} \in \mathcal{F}_k} c_k(\boldsymbol{\alpha})\boldsymbol{x}^{\boldsymbol{\alpha}} \ \text{ for every } \boldsymbol{x} \in \mathbb{R}^n \ (k = 0, 1, \ldots, m).$$

We may assume without loss of generality that $\mathbf{0} \notin \mathcal{F}_0$. We also let $\omega_k = \lceil \deg(f_k)/2 \rceil$ $(k = 0, 1, , \ldots, m)$ and $\omega_{\max} = \max\{\omega_k : k = 0, 1, \ldots, m\}$. We use the following examples to illustrate the dense and sparse SDP relaxations in the subsequent discussions.

Example 2.1.

$$\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{x}) = x_2 - 2x_1x_2 + x_2x_3 \\ \text{subject to} & f_1(\boldsymbol{x}) = 1 - x_1^2 - x_2^2 \geq 0, \ f_2(\boldsymbol{x}) = 1 - x_2^2 - x_3^2 \geq 0. \end{array}$$

Notice that $\omega_0 = \lceil \deg(f_0)/2 \rceil = 1$, $\omega_1 = \lceil \deg(f_1)/2 \rceil = 1, \omega_2 = \lceil \deg(f_2)/2 \rceil = 1$ and $\omega_{\max} = 1$.

Example 2.2.

$$\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{x}) = \sum_{i=1}^n (a_i x_i^\gamma + b_i x_i^{\gamma-1}) + c x_1 x_n \\ \text{subject to} & f_k(\boldsymbol{x}) = 1 - x_k^2 - x_{k+1}^2 \geq 0 \ (k = 1, 2, \ldots, n-1), \end{array}$$

where a_i $(i = 1, 2, \ldots, n)$, b_i $(i = 1, 2, \ldots, n)$ and c are real constants chosen randomly from $[-1, 1]$ and γ is a positive integer not less than 2. We see that $\omega_0 = \lceil \deg(f_0)/2 \rceil = \lceil \gamma/2 \rceil$, $\omega_k = \lceil \deg(f_k)/2 \rceil = 1$ $(k = 1, 2, \ldots, n-1)$ and $\omega_{\max} = \lceil \gamma/2 \rceil$.

2.2 Lasserre's Dense SDP Relaxation of a POP

Lasserre's SDP relaxation method [Las01] for a POP generates a hierarchy of SDP problems, parametrized by an integer parameter $\omega \geq \omega_{\max}$. Solving the hierarchy of SDP problems provides a sequence of monotonically nondecreasing lower bounds $\{\zeta_\omega^d : \omega \geq \omega_{\max}\}$ for the optimal value of the POP. We call each problem in the hierarchy the dense SDP relaxation problem with *the relaxation order* ω in this article. After choosing a relaxation order $\omega \geq \omega_{\max}$, we first transform the POP (1) into an equivalent polynomial SDP (PSDP)

$$\left.\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})^T f_k(\boldsymbol{x}) \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}} \\ & \qquad\qquad (k = 1, 2, \ldots, m), \\ & \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega)\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega)^T \in \mathbb{S}_+^{\mathcal{A}_\omega}. \end{array}\right\} \quad (2)$$

To verify the equivalence between the POP (1) and the PSDP (2), we first observe that they have the same objective function. If $\boldsymbol{x} \in \mathbb{R}^n$ is a feasible solution of the POP (1), then it is a feasible solution of the PSDP (2) because the symmetric matrices $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})^T$ and $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega)\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega)^T$ are positive semidefinite for any $\boldsymbol{x} \in \mathbb{R}^n$. The converse is also true because the symmetric matrices $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})^T$ $(k = 1, 2, \ldots, m)$ have the element $\boldsymbol{x}^{\boldsymbol{0}} = 1$ in their upper-left corner. This confirms the equivalence.

Let $\mathcal{F}^d = (\mathcal{A}_\omega + \mathcal{A}_\omega) = \mathcal{A}_{2\omega}$ denote the set of all monomials involved in the PSDP (2). Since the objective function is a real-valued polynomial and the left-hand side of the matrix inequality constraints are real symmetric matrix-valued polynomials, we can rewrite the PSDP (2) as

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^d} c_0^d(\boldsymbol{\alpha}) \boldsymbol{x}^{\boldsymbol{\alpha}} \\ \text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^d} \boldsymbol{L}_k^d(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}} \ (k = 1, \ldots, m), \\ & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^d} \boldsymbol{M}^d(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_\omega}, \end{array}$$

for some $c_0^d(\boldsymbol{\alpha}) \in \mathbb{R}$ ($\boldsymbol{\alpha} \in \mathcal{F}^d$), real symmetric matrices $\boldsymbol{L}_k^d(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^d,\ k = 1, \ldots, m$) and $\boldsymbol{M}^d(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^d$). Replacing each monomial $\boldsymbol{x}^{\boldsymbol{\alpha}}$ by a single variable $y_{\boldsymbol{\alpha}} \in \mathbb{R}$ provides the dense SDP relaxation problem of the POP (1):

$$\left.\begin{array}{ll}\text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^d} c_0^d(\boldsymbol{\alpha}) y_{\boldsymbol{\alpha}} \\ \text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^d} \boldsymbol{L}_k^d(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}}\ (k = 1, \ldots, m), \\ & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^d} \boldsymbol{M}^d(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_\omega},\ y_{\boldsymbol{0}} = 1. \end{array}\right\} \tag{3}$$

(Note that $\boldsymbol{0} \in \mathcal{F}^d = \mathcal{A}_{2\omega}$ and $\boldsymbol{x}^{\boldsymbol{0}} = 1$). If $\boldsymbol{x} \in \mathbb{R}^n$ is a feasible solution of the PSDP (3), $(y_{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathcal{F}^d) = (\boldsymbol{x}^{\boldsymbol{\alpha}} : \boldsymbol{\alpha} \in \mathcal{F}^d)$ is a feasible solution of (3) with the same objective value $\sum_{\boldsymbol{\alpha}\in\mathcal{F}^d} c_0^d(\boldsymbol{\alpha}) y_{\boldsymbol{\alpha}}$ as the objective value $\sum_{\boldsymbol{\alpha}\in\mathcal{F}^d} c_0^d(\boldsymbol{\alpha}) \boldsymbol{x}^{\boldsymbol{\alpha}}$ of the PSDP. This implies that (3) is a relaxation of the PSDP, hence, a relaxation of (1).

Using Example 2.1, we illustrate the dense SDP relaxation. If we take the relaxation order $\omega = \omega_{\max} = 1$, then

$$\left.\begin{array}{rcl} \mathcal{A}_{\omega-\omega_1} &=& \mathcal{A}_0 = \{(0,0,0)\}, \\ \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_1}) &=& \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_0) = \boldsymbol{u}(\boldsymbol{x}, \{(0,0,0)\}) = 1, \\ \mathcal{A}_{\omega-\omega_2} &=& \mathcal{A}_0 = \{(0,0,0)\}, \\ \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_2}) &=& \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_0) = \boldsymbol{u}(\boldsymbol{x}, \{(0,0,0)\}) = 1, \\ \mathcal{A}_\omega &=& \{(0,0,0),\ (1,0,0),\ (0,1,0),\ (0,0,1)\}, \\ \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega) &=& (1, x_1, x_2, x_3). \end{array}\right\} \tag{4}$$

The PSDP (2) is

$$\begin{array}{ll} \text{minimize} & x_2 - 2x_1x_2 + x_2x_3 \\ \text{subject to} & 1^2 \cdot (1 - x_1^2 - x_2^2) \geq 0,\ 1^2 \cdot (1 - x_2^2 - x_3^2) \geq 0, \\ & \begin{pmatrix} \boldsymbol{x}^{000} & \boldsymbol{x}^{100} & \boldsymbol{x}^{010} & \boldsymbol{x}^{001} \\ \boldsymbol{x}^{100} & \boldsymbol{x}^{200} & \boldsymbol{x}^{110} & \boldsymbol{x}^{101} \\ \boldsymbol{x}^{010} & \boldsymbol{x}^{110} & \boldsymbol{x}^{020} & \boldsymbol{x}^{011} \\ \boldsymbol{x}^{001} & \boldsymbol{x}^{101} & \boldsymbol{x}^{011} & \boldsymbol{x}^{002} \end{pmatrix} = \begin{pmatrix} 1 & x_1 & x_2 & x_3 \\ x_1 & x_1^2 & x_1x_2 & x_1x_3 \\ x_2 & x_1x_2 & x_2^2 & x_2x_3 \\ x_3 & x_1x_3 & x_2x_3 & x_3^2 \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}_\omega}. \end{array}$$

Here we simply write $\boldsymbol{x}^{\alpha_1\alpha_2\alpha_3}$ instead of $\boldsymbol{x}^{(\alpha_1,\alpha_2,\alpha_3)}$. Replacing each $\boldsymbol{x}^{\boldsymbol{\alpha}}$ by $y_{\boldsymbol{\alpha}}$, we obtain an SDP relaxation problem:

$$\begin{array}{ll} \text{minimize} & y_{010} - 2y_{110} + y_{011} \\ \text{subject to} & y_{000} - y_{200} - y_{020} \geq 0,\ y_{000} - y_{020} - y_{002} \geq 0, \\ & \begin{pmatrix} y_{000} & y_{100} & y_{010} & y_{001} \\ y_{100} & y_{200} & y_{110} & y_{101} \\ y_{010} & y_{110} & y_{020} & y_{011} \\ y_{001} & y_{101} & y_{011} & y_{002} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}_\omega},\ y_{000} = 1. \end{array}$$

Let

$$\begin{array}{rcl} \mathcal{F}^d &=& \mathcal{A}_{2\omega} \\ &=& \{(0,0,0),\ (1,0,0),\ (0,1,0),\ (0,0,1),\ (2,0,0), \\ && (1,1,0),\ (1,0,1),\ (0,2,0),\ (0,1,1),\ (0,0,2)\}. \end{array}$$

Then, we can rewrite the previous SDP relaxation problem as

$$\left.\begin{array}{ll} \text{minimize} & y_{010} - 2y_{110} + y_{011} \\ \text{subject to} & y_{000} - y_{200} - y_{020} \geq 0,\ y_{000} - y_{020} - y_{002} \geq 0, \\ & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^d} \boldsymbol{M}^d(\boldsymbol{\alpha},\omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_\omega},\ y_{000} = 1, \end{array}\right\} \tag{5}$$

for some $\boldsymbol{M}^d(\boldsymbol{\alpha},\omega) \in \mathbb{S}^{\mathcal{A}_\omega}$ $(\boldsymbol{\alpha} \in \mathcal{F}^d)$; for example,

$$\boldsymbol{M}^d((1,0,0),\omega) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},\ \boldsymbol{M}^d((0,1,1),\omega) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Let ζ_ω^d denote the optimal value of (3). Then, $\zeta_\omega^d \leq \zeta_{\omega+1}^d \leq f_0^*$ for every $\omega \geq \omega_{\max}$, where f_0^* denotes the optimal value of the POP (1). Under a moderate assumption that requires the compactness of the feasible region of (1), ζ_ω^d converges f_0^* as $\omega \to \infty$ [Las01].

We note that

$$\begin{aligned} \text{the size of } \boldsymbol{L}_k^d(\boldsymbol{\alpha},\omega) &= |\mathcal{A}_{\omega-\omega_k}| = \binom{n+\omega-\omega_k}{\omega-\omega_k} \\ &\qquad (k = 1, 2, \ldots, n-1), \\ \text{the size of } \boldsymbol{M}^d(\boldsymbol{\alpha},\omega) &= |\mathcal{A}_\omega| = \binom{n+\omega}{\omega}, \\ \text{the number of variables} &= \left|\mathcal{F}^d\right| = \binom{n+2\omega}{2\omega}, \end{aligned}$$

which increase rapidly with n and/or ω. Therefore, solving the dense SDP relaxation problem (3) becomes increasingly time-consuming and difficult as n and/or ω grow.

Numerical results are presented in Table 1 to show the growth of the three numbers of the dense SDP relaxation applied to Example 2.2. We needed to take $\omega \geq \omega_{max} = \lceil \gamma/2 \rceil$. In Table 1, notice that the size of $\boldsymbol{L}_k^d(\boldsymbol{\alpha},\omega)$, the size of $\boldsymbol{M}^d(\boldsymbol{\alpha},\omega)$ and the number of variables $|\mathcal{F}^d|$ increase rapidly as n and/or ω becomes large. Large values of n and ω often resulted in out-of-memory error, as a result, SDPA could not solve the SDP relaxation problem. We were able to obtain an accurate approximation to the optimal solution when SDPA could solve its SDP relaxation problem.

3 Sparse SDP Relaxations of a POP

3.1 Semidefinite Programming Problems That Can Be Efficiently Solved by the Primal-Dual Interior-Point Method

As shown in Section 2, POPs are transformed and relaxed into SDPs. The class of SDPs described in this section are the target SDPs into which POPs with sparsity are aimed to be relaxed for computational efficiency.

γ	n	$\omega = \omega_{\max}$	Size $\boldsymbol{L}_k^d$	Size $\boldsymbol{M}^d$	$\lvert\mathcal{F}^d\rvert$	eTime
2	10	1	1	11	66	0.05
2	20	1	1	21	231	0.08
2	40	1	1	41	861	1.08
4	10	2	11	66	1,001	2.23
4	20	2	21	231	10,626	1063.16
4	40	2	41	861	135,751	Out-of-memory
6	10	3	66	286	8,008	721.21
6	20	3	231	1771	230,230	Out-of-memory
6	40	3	861	12,341	9,366,819	Out-of-memory

Table 1: Numerical results on the dense SDP relaxation applied to Example 2.2. eTime denotes the elapsed time to solve the SDP problem by the Matlab version of SDPA 7.3.1 [SDPA] on 3.06 GHz Intel Core 2 Duo with 8HB memory.

Consider the equality standard form of SDP and its dual

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{A}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{A}_p \bullet \boldsymbol{X} = b_p \ (p = 1, 2, \ldots, m), \ \boldsymbol{X} \succeq \boldsymbol{O}, \end{array}\right\} \tag{6}$$

$$\left.\begin{array}{ll} \text{maximize} & \sum_{p=1}^m b_p y_p \\ \text{subject to} & \boldsymbol{A}_0 - \sum_{p=1}^m \boldsymbol{A}_p y_p \succeq \boldsymbol{O}, \end{array}\right\} \tag{7}$$

where $\boldsymbol{A}_p \in \mathbb{S}^n$ $(p = 0, 1, \ldots, m)$. We call the SDP (7) the linear matrix inequality (LMI) form of SDP. If no assumption is made on the sparsity of the coefficient matrices $\boldsymbol{A}_p \in \mathbb{S}^n$ $(p = 0, 1, \ldots, m)$, the efficiency of solving the SDP by the primal-dual interior-point method [Bor99, Str99, TTT03, YFN10] mainly depends on two factors. The first is the size n of the matrix variable $\boldsymbol{X}$ in (6) (or the size of the LMI constraint in (7)), and the second is the number m of equalities in (6) (or the number of the real variables y_p $(p = 1, 2, \ldots .m)$). We note that the number m determines the size of the Schur complement matrix, the $m \times m$ positive definite coefficient matrix of the Schur complement equation, which is solved at each iteration of the primal-dual interior-point method. If either n or m increases, more elapsed time and memory are required to solve the SDP. See [FFK00], for example. Increasing the efficiency of the primal-dual interior-point method can be achieved by exploiting sparsity. We discuss this issue in detail with the LMI form of SDP (7).

The simplest structured sparsity that makes the primal-dual interior-point method work efficiently is a block-diagonal structure of the coefficient matrices. Suppose that each $\boldsymbol{A}_p$ is of the form

$$\boldsymbol{A}_p = \operatorname{diag}(\boldsymbol{A}_{p1}, \boldsymbol{A}_{p2}, \ldots, \boldsymbol{A}_{p\ell}) = \begin{pmatrix} \boldsymbol{A}_{p1} & \boldsymbol{O} & \cdots & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{A}_{p2} & \cdots & \boldsymbol{O} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{A}_{p\ell} \end{pmatrix} \tag{8}$$

$(p = 0, 1, \ldots, m)$, where

$$\boldsymbol{A}_{pj} \in \mathbb{S}^{n_j} \ (j = 1, 2, \ldots, \ell, \ p = 0, 1, \ldots, m).$$

Note that for each j, the matrices $\boldsymbol{A}_{0j}, \boldsymbol{A}_{1j}, \ldots, \boldsymbol{A}_{mj}$ are of a same size $n_j \times n_j$, and that $\sum_{j=1}^{\ell} n_j = n$. In this case, we can rewrite the LMI form SDP (7) as

$$\left.\begin{array}{ll} \text{maximize} & \sum_{p=1}^{m} b_p y_p \\ \text{subject to} & \boldsymbol{A}_{0j} - \sum_{p=1}^{m} \boldsymbol{A}_{pj} y_p \succeq \boldsymbol{O} \ (j = 1, 2, \ldots, \ell). \end{array}\right\} \tag{9}$$

If we compare the original LMI form of SDP (7) with the transformed SDP (9), we notice that the single $n \times n$ matrix inequality is decomposed into multiple matrix inequalities of reduced size. Thus, if their sizes are small, the SDP (9) can be solved more efficiently than the SDP (7).

Notice that the SDP (9) involves the same number of real variables y_p $(p = 1, 2, \ldots, m)$ as in the original SDP (7), which indicates that the size of the Schur complement equation remains the same. Nevertheless, the SDP (9) can have a considerable advantage over the SDP (7). To see this clearly, let us assume without loss of generality that all coefficient matrix $\boldsymbol{A}_p$ $(p = 1, 2, \ldots, m)$ are nonzero in (7). In fact, if some $\boldsymbol{A}_p = \boldsymbol{O}$, then we must have $b_p = 0$ since otherwise the SDP (7) is unbounded. On the other hand, some of the block matrices $\boldsymbol{A}_{p1}, \boldsymbol{A}_{p2}, \ldots, \boldsymbol{A}_{p\ell}$ can be zero in (8), although the entire matrix $\boldsymbol{A}_p$ is not, for each $p = 1, 2, \ldots, m$. Then, the Schur complement matrix often becomes sparse so that the sparse Cholesky factorization can be applied to the Schur complement matrix.

In the primal-dual interior-point method [Bor99, Str99, TTT03, YFN10], each element B_{pq} of the $m \times m$ Schur complement matrix $\boldsymbol{B}$ is given by

$$B_{pq} = \sum_{j=1}^{\ell} \boldsymbol{T}_j \boldsymbol{A}_{pj} \boldsymbol{U}_j \bullet \boldsymbol{A}_{qj} \ (p, q = 1, 2, \ldots .m).$$

for fully dense $\boldsymbol{T}_j \in \mathbb{S}^{n_j}$, $\boldsymbol{U}_j \in \mathbb{S}^{n_j}$ $(j = 1, 2, \ldots, \ell)$. For details, we refer to [FFK00]. Notice that B_{pq} is nonzero if and only if there is a j such that both the coefficient matrix $\boldsymbol{A}_{pj}$ of the variable y_p and the coefficient matrix $\boldsymbol{A}_{qj}$ of the variable y_q are nonzero matrices. Thus, if we define the $m \times m$ symbolic matrix $\boldsymbol{R}$ by

$$R_{pq} \quad = \quad \begin{cases} \star & \text{if } \boldsymbol{A}_{pj} \neq \boldsymbol{O} \text{ and } \boldsymbol{A}_{qj} \neq \boldsymbol{O} \text{ for some } j = 1, 2, \ldots, \ell, \\ 0 & \text{otherwise.} \end{cases}$$

$\boldsymbol{R}$ represents the sparsity pattern of the Schur complement matrix $\boldsymbol{B}$. The matrix $\boldsymbol{R}$ is called *the correlative sparsity pattern matrix (abbreviated as the csp matrix)* of the SDP (9). A class of SDPs of the form (9) that can be solved efficiently by the primal-dual interior-point method is characterized by the conditions:

(A) The sizes of the coefficient matrices $\boldsymbol{A}_{pj}$ $(j = 1, 2, \ldots, \ell,\ p = 1, 2, \ldots, m)$ are small.

(B) The csp matrix $\boldsymbol{R}$ allows a (symbolic) sparse Cholesky factorization.

Let us consider the dense SDP relaxation problem for these conditions. The dense SDP relaxation problem (3) of the POP (1) is of the form (9). As discussed at the end of Section 2.2, the size of each coefficient matrix increases rapidly as n or ω becomes large. See Table 1. Furthermore, its csp matrix is fully dense, resulting in the dense Schur complement matrix. This was a main reason for "out-of-memory" error in Table 1. Thus, the dense SDP relaxation problem (3) satisfies neither of the conditions (A) and (B).

3.2 A Sparse Cholesky Factorization and a Chordal Graph

Let $\boldsymbol{R}$ be a symbolic symmetric matrix that represents the sparsity pattern of a class of $n \times n$ symmetric matrices introduced in the previous subsection. We assume that the nonzero symbol $\star$ is assigned to all diagonal elements and some off-diagonal elements. It is well-known that a sparse Cholesky factorization is characterized in terms of a chordal graph. We call an undirect graph *chordal* if every cycle of length ≥ 4 has a chord, i.e., an edge joining two nonconsecutive vertices of the cycle. For $\boldsymbol{R}$, an undirected graph $G(N,E)$ is defined with the node set $N = \{1, 2, \ldots, n\}$ and the edge set E such that $(i,j) \in E$ if and only if $R_{ij} = \star$ and $i > j$. Note that edge (j,i) is identified with (i,j). The graph $G(N,E)$ is called *the sparsity pattern graph* of $\boldsymbol{R}$. If the graph $G(N,E)$ is chordal, then there exists a permutation matrix $\boldsymbol{P}$ such that the matrix $\boldsymbol{P}\boldsymbol{R}\boldsymbol{P}^T$ can be factorized (symbolically) as $\boldsymbol{P}\boldsymbol{R}\boldsymbol{P}^T = \boldsymbol{L}\boldsymbol{L}^T$ with no fill-in, where $\boldsymbol{L}$ denotes a lower triangular matrix. The matrix $\boldsymbol{P}$ is obtained from a perfect elimination ordering. The maximal cliques of a chordal graph are computed with reference to the perfect elimination ordering. In fact, let $C_k = \{i : [\boldsymbol{P}^T\boldsymbol{L}]_{ik} = \star\} \subset N$ $(k = 1, 2, \ldots, n)$. Then, the maximal sets of the family of sets C_k $(k = 1, 2, \ldots, n)$ form the maximal cliques of the chordal graph $G(N,E)$.

If the graph $G(N,E)$ is not chordal, a chordal extension of $G(N,E)$ can be constructed using the sparse Cholesky factorization. Define the $n \times n$ symbolic sparsity pattern matrix $\boldsymbol{R}$ by

$$R_{pq} = \begin{cases} \star & \text{if } p = q,\ (p,q) \in E \text{ or } (q,p) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

A simultaneous row and column permutation to $\boldsymbol{R}$ such as the symmetric minimum degree ordering can be applied before the factorization of $\boldsymbol{R}$. Let $\boldsymbol{P}$ be the permutation matrix corresponding to such a permutation. Applying the Cholesky factorization to $\boldsymbol{P}\boldsymbol{R}\boldsymbol{P}^T$, we obtain a lower triangular matrix $\boldsymbol{L}$ such that $\boldsymbol{P}\boldsymbol{R}\boldsymbol{P}^T = \boldsymbol{L}\boldsymbol{L}^T$. Then the edge set $\overline{E}$ of a chordal extension $G(N,\overline{E})$ of $G(N,E)$ is obtained by $\overline{E} = \{(i,j) : i \neq j,\ [\boldsymbol{P}^T\boldsymbol{L}]_{ij} = \star\}$, and its maximal cliques can be chosen from the family of cliques $C_k = \{i : [\boldsymbol{P}^T\boldsymbol{L}]_{ik} = \star\} \subset N$ $(k = 1, 2, \ldots, n)$ as described previously. For the basic definition and properties of chordal graphs, we refer to [Gol80].

Example 3.1. Consider the sparsity pattern matrix

$$\boldsymbol{R} = \begin{pmatrix} \star & \star & 0 & 0 & 0 & 0 \\ \star & \star & \star & \star & 0 & 0 \\ 0 & \star & \star & \star & 0 & \star \\ 0 & \star & \star & \star & \star & 0 \\ 0 & 0 & 0 & \star & \star & \star \\ 0 & 0 & \star & 0 & \star & \star \end{pmatrix},$$

which yields the sparsity pattern graph $G(N,E)$ on the left of Fig. 2. This graph is not chordal because the cycle consisting of 4 edges $(3,4)$, $(4,5)$, $(5,6)$, $(6,3)$ does not have a chord. A chordal extension $G(N,\overline{E})$ of the graph $G(N,E)$ is shown on the right of Fig. 2. In this case, if the identity matrix is chosen for $\boldsymbol{P}$ and the Cholesky factorization is applied

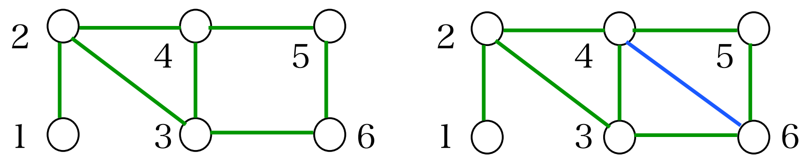

Figure 2: $G(N, E)$ on the left vs. $G(N, \overline{E})$ on the right

to $\boldsymbol{R}$, the lower triangular matrix satisfying $\boldsymbol{R} = \boldsymbol{L}\boldsymbol{L}^T$ is

$$\boldsymbol{L} \quad = \quad \begin{pmatrix} \star & 0 & 0 & 0 & 0 & 0 \\ \star & \star & 0 & 0 & 0 & 0 \\ 0 & \star & \star & 0 & 0 & 0 \\ 0 & \star & \star & \star & 0 & 0 \\ 0 & 0 & 0 & \star & \star & 0 \\ 0 & 0 & \star & \star & \star & \star \end{pmatrix}.$$

Note that a fill-in occurs in the $(6, 4)$th element, which corresponds to the edge $(6, 4)$ of the chordal extension $G(N, \overline{E})$. Each column of $\boldsymbol{L}$ leads a clique, thus, the resulting 6 cliques are

$$\{1, 2\}, \ \{2, 3, 4\}, \ \{3, 4, 6\}, \ \{4, 5, 6\}, \ \{5, 6\} \text{ and } \{6\}.$$

Choosing the maximal ones from them, we have the 4 maximal cliques

$$\{1, 2\}, \ \{2, 3, 4\}, \ \{3, 4, 6\} \ \text{ and } \{4, 5, 6\} \tag{10}$$

of the chordal extension $G(N, \overline{E})$.

3.3 Formulating Structured Sparsity

The sparsity that can be extracted from the POP (1) for a sparse SDP relaxation is discussed in this subsection. We call this sparsity *the correlative sparsity*. Let $N = \{1, 2, \ldots, n\}$. For every $k = 1, 2, \ldots, m$, the set of indices i of variables x_i in the kth inequality $f_k(\boldsymbol{x}) \geq 0$ is defined:

$$F_k = \{i \in N : c_k(\boldsymbol{\alpha}) \neq 0 \ \text{ and } \alpha_i \geq 1 \text{ for some } \boldsymbol{\alpha} \in \mathcal{F}_k\}.$$

We construct an undirected graph $G(N, E)$ to represent the sparsity structure of (1) by connecting a pair $(i, \ j)$ with $i \neq \ j$ selected from the node set N as an edge, i.e., $(i, \ j) \in E$, if and only if either there is an $\boldsymbol{\alpha} \in \mathcal{F}_0$ such that $c_0(\boldsymbol{\alpha}) \neq 0$, $\alpha_i > 0$ and $\alpha_j > 0$ or $i, \ j \in F_k$ for some $k = 1, 2, \ldots, m$. We identify each edge $(i, \ j)$ with $i > j$ with the edge (j, i). The graph $G(N, E)$ constructed this way is called *the correlative sparsity pattern (csp) graph*. Let $G(N, \overline{E})$ be a chordal extension of $G(N, E)$, and $C_1, C_2, \ldots, C_\ell$ be its maximal cliques. Since each F_k is a clique of $G(N, E)$, we can take a maximal clique $\widetilde{C}_k \in \{C_1, C_2, \ldots, C_\ell\}$ such that

$$F_k \subset \widetilde{C}_k \ (k = 1, 2, \ldots, m). \tag{11}$$

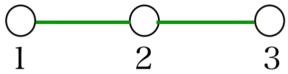

Figure 3: The csp graph $G(N, E)$ of the POP in Example 2.1.

For Example 2.1,

$$N = \{1,2,3\},\ F_1 = \{1,2\},\ F_2 = \{2,3\} \text{ and } E = \{(1,2),\ (2,3)\}.$$

Fig. 3 shows the csp graph $G(N, E)$, which is apparently chordal because there is no cycle, and the maximal cliques are $C_1 = \{1, 2\}$ and $C_2 = \{2, 3\}$. Hence, we can take $\widetilde{C}_1 = \{1, 2\} = F_1$ and $\widetilde{C}_2 = \{2, 3\} = F_2$.

Example 3.2.

$$\begin{array}{ll} \text{minimize} & -\sum_{i=1}^{6} x_i^2 \\ \text{subject to} & x_1^2 + x_2^2 \leq 1,\ x_2 + x_3^2 + x_4^2 \leq 1, \\ & x_4 + x_5^2 \leq 1, x_3 + x_6^2 \leq 1, x_5^2 + x_6^2 \leq 1. \end{array}$$

In this case, we have

$$F_1 \ = \ \{1,2\},\ F_2 = \{2,3,4\},\ F_3 = \{4,5\},\ F_4 = \{3,6\} \text{ and } \ F_5 = \{5,6\}.$$

We also see that the csp graph $G(N, E)$ coincides with the one on the left of Fig. 2. Thus, the graph on the right of Fig. 2 is a chordal extension of the csp graph $G(N, E)$, and its maximal cliques are given in (10). Consequently, we can take $\widetilde{C}_k$ $(k = 1, 2, \ldots, 5)$ as

$$\begin{array}{rcl} \widetilde{C}_1 & = & \{1,2\},\ \widetilde{C}_2 = \{2,3,4\},\ \widetilde{C}_3 = \{4,5,6\}, \\ \widetilde{C}_4 & = & \{3,4,6\} \ \text{ and } \ \widetilde{C}_5 = \{4,5,6\}. \end{array}$$

We introduce a property characterizing a class of POPs for which the sparse SDP relaxation works effectively:

(C) The csp graph $G(N, E)$ has a sparse chordal extension such that the sizes of its maximal cliques are small.

We show that Example 2.2 satisfies this property. Suppose that $n \geq 4$. Then,

$$F_k \ = \ \{k,\ k+1\}\ (k = 1,2,\ldots,n-1),$$

and the objective polynomial function $f_0(\boldsymbol{x})$ contains a monomial cx_1x_n. The csp graph $G(N, E)$ consists of n edges $(1, 2),\ (2, 3),\ \ldots,\ (n - 1, n),\ (n, 1)$, which form a cycle. The graph $G(N, E)$ is not chordal, but it has a sparse chordal extension $G(N, \overline{E})$ shown in Fig. 4. And, the extended chordal graph $G(N, \overline{E})$ consists of $n - 2$ cycles $C_k = \{k,\ k + 1,\ n\}$ $(k = 1, 2, \ldots, n - 2)$. Obviously, $F_k \subset C_k$ $(k = 1, 2, \ldots, n - 2)$ and $F_{n-1} \subset C_{n-2}$, thus, we can take $\widetilde{C}_k = C_k$ $(k = 1, 2, \ldots, n - 2)$ and $\widetilde{C}_{n-1} = C_{n-2}$.

In the next subsection, we discuss how the property (C) of a POP is transferred to its sparse SDP relaxation satisfying the properties (A) and (B).

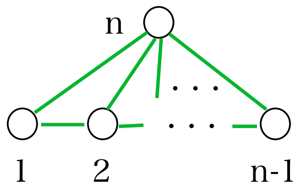

Figure 4: A chordal extension $G(N, \overline{E})$ of the csp graph $G(N, E)$ of the POP in Example 2.2

3.4 Sparse SDP Relaxation of a POP

We use the following set instead of $\mathcal{A}_\psi$ in the dense SDP relaxation. For every $\psi \in \mathbb{Z}_+$ and every nonempty $C \subset N = \{1, 2, \ldots, n\}$, define

$$\mathcal{A}_\psi^C \;=\; \{\boldsymbol{\alpha} \in \mathbb{Z}_+^n : \alpha_i = 0 \ (i \notin C) \ \text{ and } \sum_{i \in C} \alpha_i \leq \psi\},$$

where ψ stands for either a relaxation order ω or $\omega_k - \omega$ $(k = 1, 2, \ldots, m)$ and $C \subset N$ a maximal clique of a chordal extension of the csp graph.

We choose a relaxation order $\omega \geq \omega_{\max}$. Using ω_k, $\widetilde{C}_k$ $(k = 1, 2, \ldots, m)$ and C_j $(j = 1, 2, \ldots, \ell)$, we transform the POP (1) into an equivalent PSDP

$$\left.\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k}^{\widetilde{C}_k}) \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k}^{\widetilde{C}_k})^T f_k(\boldsymbol{x}) \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}^{\widetilde{C}_k}} \\ & \qquad\qquad (k = 1, 2, \ldots, m), \\ & \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega^{C_j}) \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega^{C_j})^T \in \mathbb{S}_+^{\mathcal{A}_\omega^{C_j}} \ \ (j = 1, 2, \ldots, \ell). \end{array}\right\} \tag{12}$$

Let

$$\mathcal{F}^s = \bigcup_{j=1}^{\ell} \left(\mathcal{A}_\omega^{C_j} + \mathcal{A}_\omega^{C_j} \right) = \bigcup_{j=1}^{\ell} \mathcal{A}_{2\omega}^{C_j}. \tag{13}$$

Then, we can rewrite the PSDP above as

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} c_0^s(\boldsymbol{\alpha}) \boldsymbol{x}^{\boldsymbol{\alpha}} \\ \text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}^{\widetilde{C}_k}} \ \ (k = 1, 2, \ldots, m), \\ & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_\omega^{C_j}} \ \ (j = 1, 2, \ldots, \ell), \end{array}$$

for some $c_0^s(\boldsymbol{\alpha}) \in \mathbb{R}$ $(\boldsymbol{\alpha} \in \mathcal{F}^s)$, real symmetric matrices $\boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega)$ $(\boldsymbol{\alpha} \in \mathcal{F}^s, \ k = 1, 2, \ldots, m)$ and $\boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega)$ $(\boldsymbol{\alpha} \in \mathcal{F}^s, j = 1, 2, \ldots, \ell)$. A sparse SDP relaxation problem of the POP (1) is obtained by replacing each monomial $\boldsymbol{x}^{\boldsymbol{\alpha}}$ by a single real variable $y_{\boldsymbol{\alpha}}$.

$$\left.\begin{array}{ll} \text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^s} c_0^s(\boldsymbol{\alpha})y_{\boldsymbol{\alpha}} \\ \text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha},\omega)y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k}} \ (k=1,\ldots,m), \\ & \displaystyle\sum_{\boldsymbol{\alpha}\in\mathcal{F}^s} \boldsymbol{M}_j^s(\boldsymbol{\alpha},\omega)y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}^{C_j}_{\omega}} \ (j=1,\ldots,\ell), \ y_{\boldsymbol{0}}=1. \end{array}\right\} \tag{14}$$

Let ζ^s_ω denote the optimal value of (14). Then, $\zeta^s_\omega \leq \zeta^s_{\omega+1} \leq f_0^*$ and $\zeta^s_\omega \leq \zeta^d_\omega$ for every $\omega \geq \omega_{\max}$, where f_0^* denotes the optimal value of (1) and ζ^d_ω the optimal value of the dense SDP relaxation problem (3). The second inequality $\zeta^s_\omega \leq \zeta^d_\omega$ indicates that the sparse SDP relaxation is not always as strong as the dense SDP relaxation. The convergence of ζ^s_ω to f_0^* as $\omega \to \infty$ was shown in [Las06] under a moderate condition similar to the one for the convergence of the dense SDP relaxation [Las01].

Let us apply the sparse SDP relaxation to Example 2.1 with $\omega = \omega_{\max} = 1$. Then,

$$\left.\begin{array}{rcl} \mathcal{A}^{\widetilde{C}_1}_{\omega-\omega_1} & = & \mathcal{A}^{\widetilde{C}_1}_0 = \{(0,0,0)\}, \\ \boldsymbol{u}(\boldsymbol{x},\mathcal{A}^{\widetilde{C}_1}_{\omega-\omega_1}) & = & \boldsymbol{u}(\boldsymbol{x},\mathcal{A}^{\widetilde{C}_1}_0) = \boldsymbol{u}(\boldsymbol{x},\{(0,0,0)\}) = 1, \\ \mathcal{A}^{\widetilde{C}_2}_{\omega-\omega_2} & = & \mathcal{A}^{\widetilde{C}_2}_0 = \{(0,0,0)\}, \\ \boldsymbol{u}(\boldsymbol{x},\mathcal{A}^{\widetilde{C}_2}_{\omega-\omega_2}) & = & \boldsymbol{u}(\boldsymbol{x},\mathcal{A}^{\widetilde{C}_2}_0) = \boldsymbol{u}(\boldsymbol{x},\{(0,0,0)\}) = 1, \\ \mathcal{A}^{C_1}_\omega & = & \{(0,0,0),\ (1,0,0),\ (0,1,0)\}, \\ \boldsymbol{u}(\boldsymbol{x},\mathcal{A}^{C_1}_\omega) & = & (1,x_1,x_2), \\ \mathcal{A}^{C_2}_\omega & = & \{(0,0,0),\ (0,1,0),\ (0,0,1)\}, \\ \boldsymbol{u}(\boldsymbol{x},\mathcal{A}^{C_2}_\omega) & = & (1,x_2,x_3). \end{array}\right\} \tag{15}$$

The PSDP (2) is

$$\begin{array}{ll} \text{minimize} & x_2 - 2x_1x_2 + x_2x_3 \\ \text{subject to} & 1^2\cdot(1-x_1^2-x_2^2) \geq 0,\ 1^2\cdot(1-x_2^2-x_3^2)\geq 0, \\ & \begin{pmatrix} \boldsymbol{x}^{000} & \boldsymbol{x}^{100} & \boldsymbol{x}^{010} \\ \boldsymbol{x}^{100} & \boldsymbol{x}^{200} & \boldsymbol{x}^{110} \\ \boldsymbol{x}^{010} & \boldsymbol{x}^{110} & \boldsymbol{x}^{020} \end{pmatrix} = \begin{pmatrix} 1 & x_1 & x_2 \\ x_1 & x_1^2 & x_1x_2 \\ x_2 & x_1x_2 & x_2^2 \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}^{C_1}_\omega}, \\ & \begin{pmatrix} \boldsymbol{x}^{000} & \boldsymbol{x}^{010} & \boldsymbol{x}^{001} \\ \boldsymbol{x}^{010} & \boldsymbol{x}^{020} & \boldsymbol{x}^{011} \\ \boldsymbol{x}^{001} & \boldsymbol{x}^{011} & \boldsymbol{x}^{002} \end{pmatrix} = \begin{pmatrix} 1 & x_2 & x_3 \\ x_2 & x_2^2 & x_2x_3 \\ x_3 & x_2x_3 & x_3^2 \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}^{C_2}_\omega}. \end{array}$$

Let

$$\begin{array}{rcl} \mathcal{F}^s & = & \mathcal{A}^{C_1}_{2\omega} \bigcup \mathcal{A}^{C_2}_{2\omega} \\ & = & \{(0,0,0),\ (1,0,0),\ (0,1,0),\ (2,0,0),\ (1,1,0), \\ & & \ (0,2,0),\ (0,0,1),\ (0,1,1),\ (0,0,2)\}. \end{array}$$

Replacing each $\boldsymbol{x}^{\boldsymbol{\alpha}}$ by $y_{\boldsymbol{\alpha}}$ ($\boldsymbol{\alpha} \in \mathcal{F}^s$), we obtain an SDP relaxation problem

$$\begin{array}{ll} \text{minimize} & y_{010} - 2y_{110} + y_{011} \\ \text{subject to} & y_{000} - y_{200} - y_{020} \geq 0, \ y_{000} - y_{020} - y_{002} \geq 0, \\ & \begin{pmatrix} y_{000} & y_{100} & y_{010} \\ y_{100} & y_{200} & y_{110} \\ y_{010} & y_{110} & y_{020} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}_\omega^{C_1}}, \\ & \begin{pmatrix} y_{000} & y_{010} & y_{001} \\ y_{010} & y_{020} & y_{011} \\ y_{001} & y_{011} & y_{002} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}_\omega^{C_2}}, \ y_{000} = 1. \end{array}$$

We can rewrite this problems as

$$\begin{array}{ll} \text{minimize} & y_{010} - 2y_{110} + y_{011} \\ \text{subject to} & y_{000} - y_{200} - y_{020} \geq 0, \ y_{000} - y_{020} - y_{002} \geq 0, \\ & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{M}_1^s(\boldsymbol{\alpha}, \omega)\boldsymbol{\alpha}) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_\omega^{C_1}}, \\ & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{M}_2^s(\boldsymbol{\alpha}, \omega)\boldsymbol{\alpha}) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_\omega^{C_2}}, \ y_{000} = 1, \end{array}$$

for some $M_j^s(\boldsymbol{\alpha}, \omega) \in \mathbb{S}^{\mathcal{A}_\omega^{C_1}}$ ($\boldsymbol{\alpha} \in \mathcal{F}^s, \ j = 1, 2$); for example,

$$M_1^s((1,0,0),\omega) \ = \ \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \ M_2^s((0,1,1),\omega) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}.$$

Assume that the POP (1) satisfies the property (C). We show how its sparse SDP relaxation problem inherits the property. Let $G(N, E)$ be the csp graph and $G(N, \overline{E})$ a chordal extension of $G(N, E)$ with the maximal cliques $C_1, C_2, \ldots, C_\ell$. Let $\boldsymbol{R} \in \mathbb{S}^{\mathcal{F}^s}$ be the csp matrix of the sparse SDP relaxation problem (14), and $G(\mathcal{F}^s, \mathcal{E})$ the sparsity pattern graph of $\boldsymbol{R}$, where $\mathcal{F}^s$ is given by (13). By construction, we know that

$$\begin{array}{lll} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) = \boldsymbol{O} & \text{if } \boldsymbol{\alpha} \notin \mathcal{A}_{2\omega}^{\widetilde{C}_k} & (k = 1, 2, \ldots, m), \\ \boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega) = \boldsymbol{O} & \text{if } \boldsymbol{\alpha} \notin \mathcal{A}_{2\omega}^{C_j} & (j = 1, 2, \ldots, \ell) \end{array}$$

and that each $\mathcal{A}_{2\omega}^{\widetilde{C}_k}$ is contained in some $\mathcal{A}_{2\omega}^{C_j}$. Suppose that we construct a graph $G(\mathcal{F}^s, \overline{\mathcal{E}})$ such that $(\boldsymbol{\alpha}, \boldsymbol{\beta}) \in \overline{\mathcal{E}}$ if and only if $(\boldsymbol{\alpha}, \boldsymbol{\beta}) \in \mathcal{A}_{2\omega}^{C_j}$ for some $j = 1, 2, \ldots, \ell$. Then $\mathcal{E} \subset \overline{\mathcal{E}}$, *i.e.*, $G(\mathcal{F}^s, \overline{\mathcal{E}})$ is an extension of $G(\mathcal{F}^s, \mathcal{E})$. Furthermore, we can prove that $G(\mathcal{F}^s, \overline{\mathcal{E}})$ is a chordal graph with the maximal cliques $\mathcal{A}_{2\omega}^{C_j}$ ($j = 1, 2, \ldots, \ell$). See Lemma 6.1 of [KKK08]. As a result, the chordal extension $G(\mathcal{F}^s, \overline{\mathcal{E}})$, with the maximal clique $\mathcal{A}_{2\omega}^{C_j}$ ($j = 1, 2, \ldots, \ell$), of the sparsity pattern graph of the sparse SDP relaxation problem (14) satisfies the same sparse structure as the chordal extension $G(N, \overline{E})$, with the maximal cliques C_j, ($j = 1, 2, \ldots, \ell$), of the csp graph $G(N, E)$ of the POP (1). We note that the size of the former is larger than that of the latter. Consequently, if the maximal cliques C_j ($j = 1, 2, \ldots, \ell$) are small, the SDP relaxation problem (14) satisfies the properties (A) and (B). We illustrate this by applying the sparse SDP relaxation to Example 2.2.

For Example 2.2, the extended chordal graph $G(N, \overline{E})$ of the csp graph consists of $n-2$ cycles $C_k = \{k,\ k+1,\ n\}$ $(k = 1, 2, \ldots, n-2)$. Hence, the size of the maximal cliques is 3. It follows that

$$\begin{array}{rcl} \text{the size of } \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) & = & \left|\mathcal{A}_{\omega-\omega_k}^{\widetilde{C}_k}\right| = \begin{pmatrix} 3+\omega-\omega_k \\ \omega-\omega_k \end{pmatrix} \\ & & \quad (k = 1, 2, \ldots, n-1), \\ \text{the size of } \boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega) & = & \left|\mathcal{A}_{\omega}^{C_j}\right| = \begin{pmatrix} 3+\omega \\ \omega \end{pmatrix}, \\ & & (j = 1, 2, \ldots, n-2), \\ \text{the number of variables} & = & |\mathcal{F}^s| \leq (n-2) \begin{pmatrix} 3+2\omega \\ 2\omega \end{pmatrix}, \end{array}$$

It should be noted that the sizes of $\boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega)$ and $\boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega)$ are independent of n and that the number of variables is linear in n. Table 2 shows numerical results on the sparse SDP relaxation applied to Example 2.2. Critical differences can be observed comparing Table 1 with Table 2. We mention that an accurate approximation to the optimal solution of the POP was computed for all cases of Table 2. More numerical results will be reported in Section 7.

γ	n	$\omega = \omega_{\max}$	Size $\boldsymbol{L}_k^d$	Size $\boldsymbol{M}^d$	$\lvert\mathcal{F}^d\rvert$	eTime
2	10	1	1	4	38	0.03
2	20	1	1	4	78	0.04
2	40	1	1	4	158	0.08
4	10	2	4	10	175	0.06
4	20	2	4	10	375	0.13
4	40	2	4	10	775	0.28
6	10	3	10	20	476	0.22
6	20	3	10	20	1036	0.53
6	40	3	10	20	2156	1.23

Table 2: Numerical results on the sparse SDP relaxation applied to Example 2.2. eTime denotes the elapsed time to solve the SDP problem by the Matlab version of SDPA 7.3.1 [SDPA] on 3.06 GHz Intel Core 2 Duo with 8HB memory.

4 Sums of Squares Relaxation

In this section, we derive dense and sparse relaxation problems of the POP (1) using sums of squares (SOS) polynomials combined with the generalized Lagrangian dual [KKW05, Las01, Put93]. This approach is regarded as a dual of the SDP relaxation of the POP (1). We transform the dense and sparse SOS relaxation problems to SDPs, which are dual to the dense SDP relaxation problem (3) and the sparse SDP relaxation problem (14), respectively.

4.1 Sums of Square Polynomials

We say that $f \in \mathbb{R}[\boldsymbol{x}]$ is an SOS polynomial if it can be represented as $f(\boldsymbol{x}) = \sum_{j=1}^{q} g_j(\boldsymbol{x})^2$ for a finite number of polynomials $g_j \in \mathbb{R}[\boldsymbol{x}]$ $(j = 1, 2, \ldots, q)$. Let $\mathbb{R}[\boldsymbol{x}]^2$ be the set of SOS polynomials. For every nonempty finite subset $\mathcal{G}$ of $\mathbb{Z}^n_+$, $\mathbb{R}[\boldsymbol{x}, \mathcal{G}] \subset \mathbb{R}[\boldsymbol{x}]$ denotes the set of polynomials in x_i $(i = 1, 2, \ldots, n)$ whose support is in $\mathcal{G}$; i.e.,

$$\mathbb{R}[\boldsymbol{x}, \mathcal{G}] = \{f \in \mathbb{R}[\boldsymbol{x}] : \mathrm{supp}(f) \subset \mathcal{G}\}.$$

We denote the set of SOS polynomials in $\mathbb{R}[\boldsymbol{x}, \mathcal{G}]$ by $\mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2$. Obviously, if $g \in \mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2$, then $g(\boldsymbol{x}) \geq 0$ for every $\boldsymbol{x} \in \mathbb{R}^n$. This simple fact allows us to use each $g \in \mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2$ for a nonnegative Lagrange multiplier in a generalized Lagrangian relaxation of the POP (1), and to replace the inequality ≥ 0 by the inclusion relation $\in \mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2$ for an SOS relaxation of the POP (1). By construction, $\mathrm{supp}(g) \subset \mathcal{G} + \mathcal{G}$ if $g \in \mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2$. It is also known [CLR95, KKW03, Par03, PW98] that the set $\mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2$ can be rewritten as

$$\mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2 = \left\{ \boldsymbol{u}(\boldsymbol{x}, \mathcal{G})^T \boldsymbol{V} \boldsymbol{u}(\boldsymbol{x}, \mathcal{G}) : \boldsymbol{V} \in \mathbb{S}^{\mathcal{G}}_+ \right\}.$$

4.2 Lasserre's Dense SOS Relaxation of a POP

Let

$$\Phi = \left\{ \boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_m) : \ \varphi_k \in \mathbb{R}[\boldsymbol{x}]^2 \ (k = 1, 2, \ldots, m) \ \right\}.$$

Then the generalized Lagrangian function for (1) is defined as

$$\begin{aligned} L(\boldsymbol{x}, \boldsymbol{\varphi}) \quad &= \quad f_0(\boldsymbol{x}) - \sum_{k=1}^{m} \varphi_k(\boldsymbol{x}) f_k(\boldsymbol{x}) \\ &\quad\quad \text{for every } \boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_m) \in \Phi \ \text{ and } \boldsymbol{x} \in \mathbb{R}^n. \end{aligned}$$

For each fixed $\boldsymbol{\varphi} \in \Phi$, the unconstrained minimization problem

$$\text{minimize } L(\boldsymbol{x}, \varphi) \text{ subject to } \boldsymbol{x} \in \mathbb{R}^n$$

serves as a generalized Lagrangian relaxation of the POP (1), which provides a lower bound for the optimal objective value of the POP (1). We can rewrite this problem as a semi-infinite maximization problem

$$\text{maximize } \eta \text{ subject to } L(\boldsymbol{x}, \varphi) - \eta \geq 0 \ (\boldsymbol{x} \in \mathbb{R}^n),$$

where $\boldsymbol{x} \in \mathbb{R}^n$ is a parameter vector for the continuum number of inequality constraints, not a variable vector. Thus, the best lower bound for the optimal objective value of the POP (1) among Lagrangian relaxations is obtained from the generalized Lagrangian dual problem

$$\text{maximize } \eta \text{ subject to } L(\boldsymbol{x}, \boldsymbol{\varphi}) - \eta \geq 0 \ (\boldsymbol{x} \in \mathbb{R}^n), \ \boldsymbol{\varphi} \in \Phi. \tag{16}$$

Lasserre's (dense) SOS relaxation [Las01] described in the following and its sparse variant [WKK06] described in Section 4.3 are numerically tractable subproblems of this Lagrangian dual problem.

We present the dense SOS relaxation of the POP (1) using the same quantities as the dense SDP relaxation (3) of POP (1), *i.e.*, $\omega_k = \lceil \deg(f_k)/2 \rceil$ $(k = 0, 1, , \ldots, m)$, $\omega_{\max} = \max\{\omega_k : k = 0, 1, \ldots, m\}$, $\mathcal{A}_{\omega-\omega_k}$ $(k = 1, 2, \ldots, m)$ and $\mathcal{A}_\omega$. For every $\omega \geq \omega_{\max}$, define

$$\Phi^d_\omega \;=\; \left\{ \boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_m) : \begin{array}{l} \varphi_k \in \mathbb{R}[\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k}]^2 \\ (k = 1, 2, \ldots, m) \end{array} \right\} \subset \Phi.$$

Then, the dense SOS relaxation with a relaxation order $\omega \geq \omega_{\max}$ is described as

$$\left.\begin{array}{ll} \text{maximize} & \eta \\ \text{subject to} & L(\boldsymbol{x}, \boldsymbol{\varphi}) - \eta \in \mathbb{R}[\boldsymbol{x}, \mathcal{A}_\omega]^2, \; \boldsymbol{\varphi} \in \Phi^d_\omega. \end{array}\right\} \tag{17}$$

Since $\mathbb{R}[\boldsymbol{x}, \mathcal{G}]^2 = \left\{ \boldsymbol{u}(\boldsymbol{x}, \mathcal{G})^T \boldsymbol{V} \boldsymbol{u}(\boldsymbol{x}, \mathcal{G}) : \boldsymbol{V} \in \mathbb{S}^{\mathcal{G}}_+ \right\}$ holds for every nonempty subset $\mathcal{G}$ of $\mathbb{Z}^n_+$, we can rewrite the SOS problem as

$$\left.\begin{array}{ll} \text{maximize} & \eta \\ \text{subject to} & f_0(\boldsymbol{x}) - \displaystyle\sum_{k=1}^m \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k})^T \boldsymbol{V}^k \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_k}) f_k(\boldsymbol{x}) - \eta \\ & = \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega)^T \boldsymbol{W} \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_\omega) \; (\boldsymbol{x} \in \mathbb{R}^n) \\ & \boldsymbol{V}^k \in \mathbb{S}^{\mathcal{A}_{\omega-\omega_k}}_+ \; (k = 1, 2, \ldots, m), \; \boldsymbol{W} \in \mathbb{S}^{\mathcal{A}_\omega}_+, \end{array}\right\} \tag{18}$$

where $\eta \in \mathbb{R}$ is a real variable, $\boldsymbol{V}^k$ $(k = 1, \ldots, m)$ and $\boldsymbol{W}$ matrix variables, and $\boldsymbol{x} \in \mathbb{R}^n$ a parameter for the continuum number of equality constraints. Obviously, the equality constraint of (18) requires two polynomials in the left and right sides to be equal. Comparing the coefficients of all monomials $\boldsymbol{x}^{\boldsymbol{\alpha}}$ $(\boldsymbol{\alpha} \in \mathcal{F}^d = \mathcal{A}_{2\omega})$ in the both sides, the equality constraint can be replaced by a finite number of linear equalities in $\eta \in \mathbb{R}$, $\boldsymbol{V}^k$ $(k = 1, 2, \ldots, m)$ and $\boldsymbol{W}$. As a result, we obtain an SDP of the form

$$\left.\begin{array}{ll} \text{maximize} & \eta \\ \text{subject to} & g^d_{\boldsymbol{\alpha}}(\eta, \boldsymbol{V}^1, \boldsymbol{V}^2, \ldots, \boldsymbol{V}^m, \boldsymbol{W}) = b^d_{\boldsymbol{\alpha}} \; (\boldsymbol{\alpha} \in \mathcal{F}^d), \\ & \boldsymbol{V}^k \in \mathbb{S}^{\mathcal{A}_{\omega-\omega_k}}_+ \; (k = 1, 2, \ldots, m), \; \boldsymbol{W} \in \mathbb{S}^{\mathcal{A}_\omega}_+, \end{array}\right\} \tag{19}$$

where $g^d_{\boldsymbol{\alpha}}$ denotes a real valued linear function in $\eta \in \mathbb{R}$, $\boldsymbol{V}^k \in \mathbb{S}^{\mathcal{A}_{\omega-\omega_k}}_+$ $(k = 1, 2, \ldots, m)$ and $\boldsymbol{W} \in \mathbb{S}^{\mathcal{A}_\omega}_+$, and $b^d_{\boldsymbol{\alpha}} \in \mathbb{R}$.

We can prove that the SDP problem (19) is the dual of the dense SDP problem (3). Specifically, η, $\boldsymbol{V}^k \in \mathbb{S}^{\mathcal{A}_{\omega-\omega_k}}_+$ $(k = 1, 2, \ldots, m)$ and $\boldsymbol{W} \in \mathbb{S}^{\mathcal{A}_\omega}_+$ are dual variables (Lagrange multipliers) corresponding to the equality $y_{\boldsymbol{0}} = 1$, the LMI constraints $\sum_{\boldsymbol{\alpha} \in \mathcal{F}^d} \boldsymbol{L}^d_k(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}^{\mathcal{A}_{\omega-\omega_k}}_+$ $(k = 1, \ldots, m)$ and $\sum_{\boldsymbol{\alpha} \in \mathcal{F}^d} \boldsymbol{M}^d(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}^{\mathcal{A}_\omega}_+$ of (3), respectively. Also, the variables $y_{\boldsymbol{\alpha}} \in \mathbb{R}$ $(\boldsymbol{\alpha} \in \mathcal{F}_d)$ of (3) are dual variables (Lagrange multipliers) corresponding to the equality constraints of (19). We note that the standard weak and strong duality relations hold for the SDP problems (3) and (19).

Using Example 2.1, we describe the dense SOS relaxation. As seen in (4), $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_1}) = 1$, $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega-\omega_2}) = 1$ and $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}_{\omega}) = (1, x_1, x_2, x_3)$. Thus, the SOS problem (18) is

$$
\begin{array}{ll}
\text{maximize} & \eta \\
\text{subject to} & x_2 - 2x_1x_2 + x_2x_3 - V^1_{000}(1 - x_1^2 - x_2^2) \\
& -V^2_{000}(1 - x_2^2 - x_3^2) - \eta \\
& = (1, x_1, x_2, x_3)\boldsymbol{W}(1, x_1, x_2, x_3)^T \ (\boldsymbol{x} \in \mathbb{R}^n) \\
& \boldsymbol{V}^1 = (V^1_{000}) \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_1}} = \mathbb{R}_+, \\
& \boldsymbol{V}^2 = (V^2_{000}) \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_1}} = \mathbb{R}_+, \\
& \boldsymbol{W} = \begin{pmatrix} W_{000} & W_{100} & W_{010} & W_{001} \\ W_{100} & W_{200} & W_{110} & W_{101} \\ W_{010} & W_{110} & W_{020} & W_{011} \\ W_{001} & W_{101} & W_{011} & W_{002} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}_\omega},
\end{array}
$$

and the SDP (19)

$$
\begin{array}{ll}
\text{maximize} & \eta \\
\text{subject to} & \eta + V^1_{000} + V^2_{000} + W_{000} = 0, \ W_{100} = 0, \ 2W_{010} = 1, \\
& W_{001} = 0, \ V^1_{000} - W_{200} = 0, \ 2W_{110} = -2, \ W_{101} = 0, \\
& V^1_{000} + V^2_{000} - W_{020} = 0, \ 2W_{011} = 1, \ V^2_{000} - W_{002} = 0, \\
& \boldsymbol{V}^1 = (V^1_{000}) \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_1}} = \mathbb{R}_+, \\
& \boldsymbol{V}^2 = (V^2_{000}) \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_1}} = \mathbb{R}_+, \\
& \boldsymbol{W} = \begin{pmatrix} W_{000} & W_{100} & W_{010} & W_{001} \\ W_{100} & W_{200} & W_{110} & W_{101} \\ W_{010} & W_{110} & W_{020} & W_{011} \\ W_{001} & W_{101} & W_{011} & W_{002} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}_\omega}.
\end{array}
$$

4.3 Sparse SOS Relaxations of a POP

We derive the sparse SOS relaxation of the POP (1) from the Lagrangian dual problem (16) using the same method as in Section 4.2. First, we choose a relaxation order $\omega \geq \omega_{\max}$. Define

$$
\Phi^s_\omega = \left\{ \boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_m) : \begin{array}{l} \varphi_k \in \mathbb{R}[\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k}]^2 \\ (k = 1, 2, \ldots, m) \end{array} \right\} \subset \Phi^d_\omega \subset \Phi,
$$

where each $\widetilde{C}_k$ denotes a maximal clique of the chordal extension $G(N, \overline{E})$ of the csp graph $G(N, E)$ of the POP (1) satisfying (11). Then, the sparse SOS relaxation with the relaxation order ω is described as

$$
\left.
\begin{array}{ll}
\text{maximize} & \eta \\
\text{subject to} & L(\boldsymbol{x}, \boldsymbol{\varphi}) - \eta \in \sum_{j=1}^{\ell} \mathbb{R}[\boldsymbol{x}, \mathcal{A}^{C_j}_\omega]^2, \ \boldsymbol{\varphi} \in \Phi^s_\omega.
\end{array}
\right\} \tag{20}
$$

If we compare the sparse SOS relaxation to the dense SOS relaxation (17), we notice that $\boldsymbol{\varphi} \in \Phi^d_\omega$ is replaced by $\boldsymbol{\varphi} \in \Phi^s_\omega$ and $\mathbb{R}[\boldsymbol{x}, \mathcal{A}_\omega]^2$ by $\sum_{j=1}^{\ell} \mathbb{R}[\boldsymbol{x}, \mathcal{A}^{C_j}_\omega]^2$. Since

$$
\Phi^s_\omega \subset \Phi^d_\omega \ \text{ and } \sum_{j=1}^{\ell} \mathbb{R}[\boldsymbol{x}, \mathcal{A}^{C_j}_\omega]^2 \subset \mathbb{R}[\boldsymbol{x}, \mathcal{A}_\omega]^2,
$$

the optimal value η_ω^s of (20) is smaller than or equal to the optimal value η_ω^d of (19). Thus, the sparse SOS relaxation is weaker than the dense SOS relaxation in general.

As in the dense SOS relaxation, we can convert the sparse SOS relaxation (20) to an SDP problem:

$$\left.\begin{array}{ll} \text{maximize} & \eta \\ \text{subject to} & g^s_{\boldsymbol{\alpha}}(\eta, \boldsymbol{V}^1, \ldots, \boldsymbol{V}^m, \boldsymbol{W}^1, \ldots, \boldsymbol{W}^\ell) = b^s_{\boldsymbol{\alpha}} \ (\boldsymbol{\alpha} \in \mathcal{F}^s), \\ & \boldsymbol{V}^k \in \mathbb{S}_+^{\mathcal{A}^{\overline{C}_k}_{\omega-\omega_k}} \ (k = 1, 2, \ldots, m), \\ & \boldsymbol{W}^j \in \mathbb{S}_+^{\mathcal{A}^{C_j}_{\omega}} \ (j = 1, 2, \ldots, \ell), \end{array}\right\} \tag{21}$$

where $g^s_{\boldsymbol{\alpha}}$ denotes a real valued linear function in $\eta \in \mathbb{R}$, $\boldsymbol{V}^k \in \mathbb{S}_+^{\mathcal{A}^{\overline{C}_k}_{\omega-\omega_k}}$ $(k = 1, 2, \ldots, m)$, and $\boldsymbol{W}^j \in \mathbb{S}_+^{\mathcal{A}^{C_j}_{\omega}}$ $(j = 1, 2, \ldots, \ell)$. This SDP is the dual of the sparse SDP relaxation problem (14). Let η_ω^s be the optimal value of (21). Then $\eta_\omega^s \leq \eta_{\omega+1}^s \leq f_0^*$ and $\eta_\omega^s \leq \zeta_\omega^s \leq f_0^*$ for every $\omega \geq \omega_{\max}$, where f_0^* denote the optimal value of the POP (1) and ζ_ω^s the optimal value of the sparse SDP relaxation problem (14).

Remark 4.1. The dense and sparse SOS relaxations were presented in [Las01] and [Las06]. It was shown that the optimal value ζ_ω^d of the dense SDP relaxation problem (3) (or the optimal value ζ_ω^s of the sparse SDP relaxation problem (14)) of the POP (1) with the relaxation order $\omega \geq \omega_{\max}$ converges to the optimal value f_0^* of (1) as $\omega \to \infty$ under the assumptions that require the compactness of the feasible region of (1). A key fact used in the proof of the convergence was Putinar's Lemma (Lemma 4.1 of [Put93]), which was applied to the dense SOS relaxation problem (17) (or the sparse SOS relaxation problem (20)). We note that the convergence of ζ_ω^d (or ζ_ω^s) to f_0^* as $\omega \to \infty$ follows from the convergence of η_ω^d (or η_ω^s) to f_0^* as $\omega \to \infty$ since $\eta_\omega^d \leq \zeta_\omega^d \leq f_0^*$ (or $\eta_\omega^s \leq \zeta_\omega^s \leq f_0^*$) for every $\omega \geq \omega_{\max}$.

We now apply the sparse SOS relaxation presented previously to Example 2.1. Let $\omega = \omega_{\max} = 1$. Recall that the csp graph $G(N, E)$ shown in Fig. 3 is a chordal graph having the maximal cliques $C_1 = \{1, 2\}$ and $C_2 = \{2, 3\}$, $F_1 = \widetilde{C}_1 = C_1$, $F_2 = \widetilde{C}_2 = C_2$. In addition, we have observed in (15) that $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_1}_{\omega-\omega_1}) = 1$, $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_2}_{\omega-\omega_2}) = 1$, $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{C_1}_{\omega}) = (1, x_1, x_2)$ and $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{C_2}_{\omega}) = (1, x_2, x_3)$. Thus, we obtain an SOS relaxation problem

$$\begin{array}{ll} \text{maximize} & \eta \\ \text{subject to} & x_2 - 2x_1x_2 + x_2x_3 - V^1_{000}(1 - x_1^2 - x_2^2) \\ & -V^2_{000}(1 - x_2^2 - x_3^2) - \eta \\ & = (1, x_1, x_2)\boldsymbol{W}^1(1, x_1, x_2)^T \\ & +(1, x_2, x_3)\boldsymbol{W}^2(1, x_2, x_3)^T \ (\boldsymbol{x} \in \mathbb{R}^n) \\ & \boldsymbol{V}^1 = (V^1_{000}) \in \mathbb{S}_+^{\mathcal{A}^{\widetilde{C}_1}_{\omega-\omega_1}} = \mathbb{R}_+, \\ & \boldsymbol{V}^2 = (V^2_{000}) \in \mathbb{S}_+^{\mathcal{A}^{\widetilde{C}_2}_{\omega-\omega_2}} = \mathbb{R}_+, \\ & \boldsymbol{W}^1 = \begin{pmatrix} W^1_{000} & W^1_{100} & W^1_{010} \\ W^1_{100} & W^1_{200} & W^1_{110} \\ W^1_{010} & W^1_{110} & W^1_{020} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}^{C_1}_{\omega}}, \\ & \boldsymbol{W}^2 = \begin{pmatrix} W^2_{000} & W^2_{010} & W^2_{001} \\ W^2_{010} & W^2_{020} & W^2_{011} \\ W^2_{001} & W^2_{011} & W^2_{002} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}^{C_2}_{\omega}}, \end{array}$$

and an equivalent SDP problem

$$
\begin{array}{ll}
\text{maximize} & \eta \\
\text{subject to} & \eta + V^1_{000} + V^2_{000} + W^1_{000} + W^2_{000} = 0,\ W^1_{100} = 0, \\
& 2W^1_{010} + 2W^2_{010} = 1,\ W^2_{001} = 0, \\
& V^1_{000} - W^1_{200} - W^2_{200} = 0,\ 2W^1_{110} = -2, \\
& W^1_{101} = 0,\ V^1_{000} + V^2_{000} - W^1_{020} = 0, \\
& 2W^2_{011} = 1,\ V^2_{000} - W^2_{002} = 0, \\
& \boldsymbol{V}^1 = (V^1_{000}) \in \mathbb{S}_+^{\mathcal{A}^{\widetilde{C}_1}_{\omega-\omega_1}} = \mathbb{R}_+, \\
& \boldsymbol{V}^2 = (V^2_{000}) \in \mathbb{S}_+^{\mathcal{A}^{\widetilde{C}_2}_{\omega-\omega_2}} = \mathbb{R}_+, \\
& \boldsymbol{W}^1 = \begin{pmatrix} W^1_{000} & W^1_{100} & W^1_{010} \\ W^1_{100} & W^1_{200} & W^1_{110} \\ W^1_{010} & W^1_{110} & W^1_{020} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}^{C_1}_{\omega}}, \\
& \boldsymbol{W}^2 = \begin{pmatrix} W^2_{000} & W^2_{010} & W^2_{001} \\ W^2_{010} & W^2_{020} & W^2_{011} \\ W^2_{001} & W^2_{011} & W^2_{002} \end{pmatrix} \in \mathbb{S}_+^{\mathcal{A}^{C_2}_{\omega}}.
\end{array}
$$

5 Additional Techniques

We present important techniques used in SparsePOP [WKK06] for its efficiency, accuracy and numerical stability, and for its application to practical problems.

5.1 Handling Equality, Lower Bound and Upper Bound Constraints

In practice, the sparse SDP relaxation described in Section 3.4 is applied to POPs with inequalities, equalities, lower bounds and/or upper bounds. More precisely, consider

$$
\left.
\begin{array}{lll}
\text{minimize} & f_0(\boldsymbol{x}) & \\
\text{subject to} & f_k(\boldsymbol{x}) \geq 0 & (k = 1, 2, \ldots, q), \\
& f_k(\boldsymbol{x}) = 0 & (k = q+1, \ldots, m), \\
& \lambda_i \leq x_i \leq \nu_i & (i \in N),
\end{array}
\right\} \tag{22}
$$

where $-\infty \leq \lambda_i < \nu_i \leq \infty$ $(i \in N)$. We use the same notation and symbols as in Section 3.4. We first observe that the POP (22) is equivalent to the PSDP

$$
\begin{array}{ll}
\text{minimize} & f_0(\boldsymbol{x}) \\
\text{subject to} & \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k}) \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k})^T f_k(\boldsymbol{x}) \in \mathbb{S}_+^{\mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k}} \\
& \qquad\qquad\qquad (k = 1, 2, \ldots, q), \\
& \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k}) \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{\widetilde{C}_k}_{\omega-\omega_k})^T f_k(\boldsymbol{x}) = \boldsymbol{O} \\
& \qquad\qquad (k = q+1, q+2, \ldots, m), \\
& \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{C_j}_{\omega}) \boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{C_j}_{\omega})^T \in \mathbb{S}_+^{\mathcal{A}^{C_j}_{\omega}} \ (j = 1, 2, \ldots, \ell), \\
& \lambda_i \leq x_i \leq \nu_i \ (i \in N),
\end{array}
$$

where the relaxation ω is chosen such that $\omega \geq \omega_{\max} = \max\{\omega_k : k = 0, 1, \ldots, m\}$. We then rewrite this problem as

$$
\begin{array}{ll}
\text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} c_0^s(\boldsymbol{\alpha}) \boldsymbol{x}^{\boldsymbol{\alpha}} \\
\text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}^{\tilde{C}_k}} \ (k = 1, 2, \ldots, q), \\
& \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} = \boldsymbol{O} \ (k = q+1, q+2, \ldots, m), \\
& \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega) \boldsymbol{x}^{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega}^{C_j}} \ (j = 1, 2, \ldots, \ell), \\
& \lambda_i \leq x_i \leq \nu_i \ (i \in N),
\end{array}
$$

for some $c_0^s(\boldsymbol{\alpha}) \in \mathbb{R}$ ($\boldsymbol{\alpha} \in \mathcal{F}^s$), real symmetric matrices $\boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^s$, $k = 1, 2, \ldots, m$) and $\boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^s, j = 1, 2, \ldots, \ell$). Thus, replacing each monomial $\boldsymbol{x}^{\boldsymbol{\alpha}}$ by a single real variable $y_{\boldsymbol{\alpha}}$, we obtain the SDP relaxation problem of the POP (22)

$$
\left.
\begin{array}{ll}
\text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} c_0^s(\boldsymbol{\alpha}) y_{\boldsymbol{\alpha}} \\
\text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega-\omega_k}^{\tilde{C}_k}} \ (k = 1, 2, \ldots, q), \\
& \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} = \boldsymbol{O} \ (k = q+1, q+2, \ldots, m), \\
& \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}_+^{\mathcal{A}_{\omega}^{C_j}} \ (j = 1, 2, \ldots, \ell), \ y_{\boldsymbol{0}} = 1, \\
& \lambda_i \leq y_{\boldsymbol{e}^i} \leq \nu_i \ (i \in N),
\end{array}
\right\} \tag{23}
$$

where $\boldsymbol{e}^i$ denotes the ith unit vector in $\mathbb{R}^n$ $(i \in N)$. For each $k = q+1, q+2, \ldots, m$, all coefficient matrices $\boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^s$) of the equality constraint

$$
\sum_{\boldsymbol{\alpha} \in \mathcal{F}^s} \boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} = \boldsymbol{O} \in \mathbb{S}^{\mathcal{A}_{\omega-\omega_k}^{\tilde{C}_k}}
$$

are symmetric. Hence, we can rewrite this equality constraint with respect to each component in the lower triangular part of $\boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^s$).

5.2 Computing Optimal Solutions

After solving the sparse SDP relaxation by an SDP solver, an approximation to an optimal solution of the POP (22) needs to be extracted. We describe a simple method used in SparsePOP. By default, SparsePOP assumes that the POP (22) to be solved has a unique optimal solution. Let ζ_ω denote the optimal objective value and $(y_{\boldsymbol{\alpha}}^\omega : \boldsymbol{\alpha} \in \mathcal{F}^s)$ the optimal solution of the SDP (23). Note that the values of $y_{\boldsymbol{\alpha}}^\omega$ ($\boldsymbol{\alpha} \in \{\boldsymbol{e}^i : i \in N\}$) correspond to the variable x_i $(i \in N)$ in the POP (22). Thus, these values can be used as an approximation to an optimal solution $\boldsymbol{x}^\omega$ of the POP (22). Let $x_i^\omega = y_{\boldsymbol{e}^i}$ $(i \in N)$ and $\boldsymbol{x}^\omega = (x_1^\omega, x_2^\omega, \ldots, x_n^\omega)$. We know that $f_0(\boldsymbol{x}) \geq f_0^* \geq \zeta_\omega$ for every feasible solution $\boldsymbol{x}$ of the POP (22), where f_0^* denotes the unknown optimal value of (22). Therefore, we may regard $\boldsymbol{x}^\omega$ an approximation to the optimal solution of (22) (a) if $\boldsymbol{x}^\omega$ (approximately) satisfies the constraints of (22) and (b) if $f_0(\boldsymbol{x}^\omega) - \zeta_\omega$ is sufficiently small. SparsePOP provides output information to decide whether (a) and (b) are satisfied.

If the POP (22) has multiple global optimal solutions, the method described previously will not work. In this case, SparsePOP replaces the objective polynomial $f_0(\boldsymbol{x})$ by $f_0(\boldsymbol{x}) + \epsilon \boldsymbol{d}^T \boldsymbol{x}$ with an n-dimensional column vector $\boldsymbol{d}$ whose components are chosen randomly from $[0, 1]$ and $\epsilon > 0$ a small positive parameter controlling the magnitude of the perturbation term $\epsilon \boldsymbol{d}^T \boldsymbol{x}$. The POP with this perturbed objective function is expected to have a unique optimal solution. Then, we can apply the method described previously, and we may regard $\boldsymbol{x}^\omega$ an approximation to an optimal solution of the POP (22) if (a) and (b) are satisfied. See the paper [WKK06] for more details.

A linear algebra method proposed by Henrion and Lasserre [HL03] computes multiple optimal solutions of the POP (1) from an optimal solution of the dense SDP relaxation problem (3). This method was extended to the sparse SDP relaxation problem (14) [Las06]. It was not implemented in SparsePOP since the cost of solving large-scale POPs is expected to be much more expensive than the simple method discussed in this subsection.

5.3 Choosing a Higher Relaxation Order

If no parameter is specified, SparsePOP applies the spares SDP relaxation with the relaxation order $\omega = \omega_{\max}$ to a POP to be solved. If the obtained solution is not within the range of desired accuracy, one can run SparsePOP again with different values of the parameters to attain an optimal solution of higher accuracy. In particular, the relaxation order ω determines both the quality of an approximate solution and the size of the SDP relaxation problem (23) of the POP (22). An approximation to an optimal solution of the POP (22) with higher accuracy or not lower accuracy is expected by solving the SDP relaxation problem (23) as a larger value is chosen for ω. This, however, increases the cost of solving the SDP relaxation (23). Thus, $\omega = \omega_{\max}$ is used initially, and it is successively increased by 1 if the approximate solution $\boldsymbol{x}^\omega$ with higher accuracy needs to be found.

5.4 Scaling

In many POPs from applications, it is necessary to perform a scaling to obtain meaningful numerical solutions. The scaling technique described here is intended to improve the numerical stability. Suppose that both of the lower and upper bounds on x_i are finite, *i.e.*, $-\infty < \lambda_i < \nu_i < \infty$ $(i \in N)$ in the POP (22). We perform a linear transformation to the variables x_i such that $z_i = (x_i - \lambda_i)/(\nu_i - \lambda_i)$. The objective and constrained polynomials $g_k \in \mathbb{R}[\boldsymbol{z}]$ $(k = 0, 1, \ldots, m)$ become

$$\begin{aligned} & g_k(z_1, z_2, \ldots, z_n) \\ = \;& f_k((\nu_1 - \lambda_1)z_1 + \lambda_1, (\nu_2 - \lambda_2)z_2 + \lambda_2, \ldots, (\nu_n - \lambda_n)z_n + \lambda_n). \end{aligned}$$

Then, normalize each $g_k \in \mathbb{R}[\boldsymbol{z}]$ such that $h_k(\boldsymbol{z}) = g_k(\boldsymbol{z})/\chi_k$, where χ_k denotes the maximum of the magnitude of the coefficients of the polynomial $g_k \in \mathbb{R}[\boldsymbol{z}]$ $(k = 0, 1, 2, \ldots, m)$. Consequently, we obtain a scaled POP

$$\begin{array}{lll} \text{minimize} & h_0(\boldsymbol{z}) & \\ \text{subject to} & h_k(\boldsymbol{z}) \geq 0 & (k = 1, 2, \ldots, q), \\ & h_k(\boldsymbol{z}) = 0 & (k = q+1, \ldots, m), \\ & 0 \leq z_i \leq 1 & (i \in N), \end{array}$$

which is equivalent to the POP (22).

5.5 Reducing the Sizes of SDP Relaxation Problems

A method to exploit the sparsity of SOS polynomials was proposed in [CLR95] to reduce the size of the SOS relaxation. See also [KKW03]. This method is implemented in SparsePOP to reduce the number and the sizes of the coefficient matrices $\boldsymbol{M}^s_j(\boldsymbol{\alpha}, \omega)$ $(\boldsymbol{\alpha} \in \mathcal{F}^s,\ j = 1, 2, \ldots, \ell)$ in the sparse SDP relaxation (23) of the POP (22). We provide a brief description of the method for the sparse SOS and SDP relaxations of the POP (1).

For every $\mathcal{G}_j \subset \mathcal{A}^{C_j}_\omega$ $(j = 1, 2, \ldots, \ell)$, let

$$\begin{array}{l} \Gamma(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell) \\ \quad = \left\{ (\boldsymbol{\varphi}, \eta) \in \Phi^s_\omega \times \mathbb{R} : L(\boldsymbol{x}, \boldsymbol{\varphi}) - \eta \in \displaystyle\sum_{j=1}^{\ell} \mathbb{R}[\boldsymbol{x}, \mathcal{G}_j]^2 \right\}. \end{array}$$

Then, the constraint of the SOS problem (20) can be rewritten as $(\boldsymbol{\varphi}, \eta) \in \Gamma(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell)$ with $(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell) = (\mathcal{A}^{C_1}_\omega, \mathcal{A}^{C_2}_\omega, \ldots, \mathcal{A}^{C_\ell}_\omega)$. As a result, if

$$\Gamma(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell) = \Gamma(\mathcal{A}^{C_1}_\omega, \mathcal{A}^{C_2}_\omega, \ldots, \mathcal{A}^{C_\ell}_\omega) \tag{24}$$

holds for some subset $(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell)$ of $(\mathcal{A}^{C_1}_\omega, \mathcal{A}^{C_2}_\omega, \ldots, \mathcal{A}^{C_\ell}_\omega)$, then the constraint of the SOS problem (20) can be replaced by $L(\boldsymbol{x}, \boldsymbol{\varphi}) - \eta \in \sum_{j=1}^{\ell} \mathbb{R}[\boldsymbol{x}, \mathcal{G}_j]^2$.

Recall that the sparse SOS relaxation is a dual of the sparse SDP relaxation described in Section 3.4. The primal SDP relaxation corresponding to the sparse SDP relaxation is obtained by replacing the constraint $\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{C_j}_\omega)\boldsymbol{u}(\boldsymbol{x}, \mathcal{A}^{C_j}_\omega)^T \in \mathbb{S}^{\mathcal{A}^{C_j}_\omega}_+$ $(j = 1, 2, \ldots, \ell)$ in the PSDP (12) by $\boldsymbol{u}(\boldsymbol{x}, \mathcal{G}_j)\ \boldsymbol{u}(\boldsymbol{x}, \mathcal{G}_j)^T \in \mathbb{S}^{\mathcal{G}_j}_+$ $(j = 1, 2, \ldots, \ell)$, $\mathcal{F}^s$ by $\mathcal{G}^s = \bigcup_{j=1}^{\ell} (\mathcal{G}_j + \mathcal{G}_j)$, and the sparse SDP problem (14) by

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{G}^s} c^s_0(\boldsymbol{\alpha}) y_{\boldsymbol{\alpha}} \\ \text{subject to} & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{G}^s, \omega} \boldsymbol{L}^s_k(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}^{\mathcal{A}^{\tilde{C}_k}_{\omega - \omega_k}}_+ \ (k = 1, \ldots, m), \\ & \displaystyle\sum_{\boldsymbol{\alpha} \in \mathcal{G}^s} \boldsymbol{N}^s_j(\boldsymbol{\alpha}, \omega) y_{\boldsymbol{\alpha}} \in \mathbb{S}^{\mathcal{G}_j}_+ \ (j = 1, \ldots, \ell),\ y_{\boldsymbol{0}} = 1, \end{array}$$

for some $c_0^s(\boldsymbol{\alpha}) \in \mathbb{R}$ ($\boldsymbol{\alpha} \in \mathcal{G}^s$), real symmetric matrices $\boldsymbol{L}_k^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{G}^s,\ k = 1, 2, \ldots, m$) and $\boldsymbol{N}_j^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{G}^s, j = 1, 2, \ldots, \ell$). Note that if $\mathcal{G}_j$ is a proper subset of $\mathcal{A}_\omega^{C_j}$), then the number of variables $y_{\boldsymbol{\alpha}}$ ($\boldsymbol{\alpha} \in \mathcal{G}^s$) is smaller than that in the SDP relaxation problem (14) and the size of the coefficient matrices $\boldsymbol{N}_j^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{G}^s$) is smaller than the size of the corresponding coefficient matrices $\boldsymbol{M}_j^s(\boldsymbol{\alpha}, \omega)$ ($\boldsymbol{\alpha} \in \mathcal{F}^s$).

We now present the procedure for finding a proper subset $(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell)$ of $(\mathcal{A}_\omega^{C_1}, \mathcal{A}_\omega^{C_2}, \ldots, \mathcal{A}_\omega^{C_\ell})$ satisfying (24).

Step 0: Let $\mathcal{F}$ denote the set of exponents of monomials $\boldsymbol{x}^{\boldsymbol{\alpha}}$ in $L(\boldsymbol{x}, \boldsymbol{\varphi}) - \zeta \boldsymbol{x}^{\boldsymbol{0}}$ for some $\boldsymbol{\varphi} \in \Phi_\omega^s$, and $\mathcal{F}^e = \{\boldsymbol{\alpha} \in \mathcal{F} : \alpha_i \text{ is even } (i \in N)\}$. Let $(\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_\ell) = (\mathcal{A}_\omega^{C_1}, \mathcal{A}_\omega^{C_2}, \ldots, \mathcal{A}_\omega^{C_\ell})$.

Step 1: Find $\boldsymbol{\alpha} \in \bigcup_{j=1}^{\ell} \mathcal{G}_j$ such that

$$2\boldsymbol{\alpha} \notin \mathcal{F}^e \ \text{ and } 2\boldsymbol{\alpha} \notin \bigcup_{j=1}^{\ell} \{\boldsymbol{\beta} + \boldsymbol{\gamma} : \boldsymbol{\beta} \in \mathcal{G}_j,\ \boldsymbol{\gamma} \in \mathcal{G}_j,\ \boldsymbol{\beta} \neq \boldsymbol{\alpha}\}.$$

Step 2: If there is no such $\boldsymbol{\alpha} \in \bigcup_{j=1}^{\ell} \mathcal{G}_j$, then stop. Otherwise, let $\mathcal{G}_j = \mathcal{G}_j \backslash \{\boldsymbol{\alpha}\}$ ($j = 1, 2, \ldots, \ell$), and go to Step 1.

See [KKW03] for more details.

6 Numerical Results on SparsePOP with SDPA

SparsePOP [SPOP] is a Matlab package for solving unconstrained and constrained POPs of the form (22) by the sparse (or dense) SDP relaxation method. When SparsePOP is called for a POP specified in either the SparsePOP format [SPOP] or the GAMS scalar format [GAMS], a sparse (or dense) SDP relaxation problem is first constructed with given parameters, and then solved by the SDP solver SeDuMi [Str99] or SDPA [SDPA, YFN10]. At the end of computation, various information including approximations to the global optimal value and solution, a lower bound for the global optimal value of the POP and the elapsed time for solving the SDP problem is provided. As an option, SparsePOP refines the computed approximation to the global optimal solution by applying the Matlab functions fminunc, fmincon or lsqnonlin in Matlab Optimization Toolbox.

We compared the dense and sparse SDP relaxations, which were implemented in SparsePOP 2.20, with selected test problems from [CGT88, MFH81, Glo] to confirm the effectiveness of exploiting the sparsity of POPs for improving the computational efficiency. We used a Matlab version of SDPA 7.3.1 as an SDP solver, and performed all numerical experiments on a 2.8GHz Intel Quad-Core i7 with 16GB memory.

The numerical results are shown in Table 3. The chained singular function, the Broyden tridiagonal function, the Rosenbrock function, and the chained wood function in Table 3 are unconstrained problems of the form: minimize $f_0(\boldsymbol{x})$ over $\boldsymbol{x} \in \mathbb{R}^n$. The other test problems are constrained POPs of the form (22) from [Glo]. The Broyden tridiagonal function has two global minimizers, the one with $x_1 > 0$ and the other with $x_1 < 0$. We added the constraint $x_1 \geq 0$ to exclude the second solution so that the function has a unique global optimal solution subject to the constraint. For the numerical stability, we added lower and upper bounds for the variables of some constrained POPs. See Section 5.4. The following is the notation for Table 3.

$$\begin{aligned}
\text{eTime} &= \text{the elapsed time in SDPA (the elapsed time} \\
&\quad\ \text{in the Matlab function fmincon) in seconds,} \\
\text{rObjErr} &= \frac{|\text{opt. val. of SDP} - f_0(\hat{\boldsymbol{x}})|}{\max\{1, f_0(\hat{\boldsymbol{x}})\}}, \\
\text{absErr} &= \min\{f_j(\hat{\boldsymbol{x}})\ (j = 1, 2, \ldots, q),\ -|f_k(\hat{\boldsymbol{x}})|\ (k = q+1, \ldots, m)\},
\end{aligned}$$

where $\hat{\boldsymbol{x}}$ denotes an approximation to the optimal solution.

In Table 3, we observe that the size of unconstrained POPs that could be handled by the sparse SDP relaxation is much larger than that of the dense SDP relaxation. More precisely, the sparse SDP relaxation provided an accurate solution for the unconstrained problems with $n = 10,000$ while the dense SDP relaxation solved the problems of size up to $n = 24$. For the constrained problems, the elapsed time for solving the sparse SDP relaxation problem is much shorter than that for solving the dense SDP relaxation problem. We can see that exploiting the sparsity was crucial to reduce the elapsed time. The errors in the sparse SDP relaxation are compatible to those in the dense SDP relaxation.

Important factors that affect the computational efficiency for solving SDPs are:

$$\begin{aligned}
\text{no} &= \text{the number of positive semidefinite matrix variables} \\
&\quad\ \text{of the SDP relaxation problem in the SDPA sparse format,} \\
\text{max} &= \text{the maximum size of positive semidefinite matrix variables} \\
&\quad\ \text{of the SDP relaxation problem in the SDPA sparse format,} \\
\text{size}\boldsymbol{L} &= \text{the size of the Schur complement matrix,} \\
\text{nnz}\boldsymbol{L} &= \text{the number of nonzeros in a sparse Cholesky factor} \\
&\quad\ \text{of the Schur complement matrix.}
\end{aligned}$$

Table 4 shows these numbers for the dense and sparse SDP relaxations. We notice that the number of positive semidefinite variable matrices of the sparse SDP relaxation is larger than that of the dense SDP relaxation. In fact, the difference is obtained by subtracting one from the number of the maximum cliques C_ℓ $(j = 1, 2, \ldots, \ell)$ of the chordal extension of the csp graph of the POP to be solved. The maximum size of positive semidefinite matrix variables coincides with the size of $\mathcal{A}_\omega$ in the dense SDP relaxation problem (3), while it coincides with the maximum of sizes of $\mathcal{A}_\omega^{C_j}$ $(j = 1, 2, \ldots, \ell)$ in the sparse SDP relaxation problem (14). We observe that the difference becomes larger as n or ω increases. Notice that the size and the number of nonzeros of the Cholesky factor $\boldsymbol{L}$ of the Schur complement matrix $\boldsymbol{B}$ are both much smaller in the sparse SDP relaxation. This confirms that these factors considerably contributed to improving the efficiency of the sparse SDP relaxation.

				Dense SDP (+fmincon)			Sparse SDP (+fmincon)		
Problem	n	$\omega_{\max}$	ω	eTime	rObjErr	absErr	eTime	rObjErr	absErr
Chained Singular	12	2	2	8.7	4.6e-05		0.1	6.9e-04	
	24	2	2	6281.7	2.2e-05		0.1	3.3e-04	
	1000	2	2				6.2	8.8e-04	
	10000	2	2				79.1	5.8e-04	
Broyden Tridiag.	12	2	2	8.4	1.6e-06		0.1	5.7e-07	
	24	2	2	6364.9	7.8e-07		0.1	1.2e-06	
	1000	2	2				5.6	4.3e-06	
	10000	2	2				61.8	9.2e-04	
Chained Wood	12	2	2	0.2	7.9e-06		0.0	5.1e-05	
	24	2	2	30.9	7.3e-07		0.0	1.0e-05	
	1000	2	2				1.5	4.4e-04	
	10000	2	2				20.7	4.4e-03	
Rosen-brock	12	2	2	4.6	2.2e-06		0.0	8.2e-05	
	24	2	2	3974.7	8.6e-07		0.1	9.4e-05	
	1000	2	2				3.3	6.0e-05	
	10000	2	2				37.2	7.2e-05	
ex2_1_8	24	1	1	0.2(0.4)	6.5e00	-2.7e-16	0.1(0.0)	6.5e00	-2.7e-16
			2	34.2(0.0)	1.2e-06	-1.7e-16	4.6(0.0)	2.7e-06	-1.8e-16
ex3_1_1	8	1	2	0.3(0.2)	3.3e-02	-9.4e-13	0.1(0.1)	3.5e-02	-1.8e-14
			3	72.8(0.0)	4.8e-07	-7.0e-14	0.4(0.0)	4.3e-07	-6.2e-14
ex5_2_2 _case1	9	1	1	0.0(0.1)	5.7e02	-1.4e-12	0.0(0.1)	5.7e02	-1.4e-12
			2	0.3(0.0)	2.0e-04	-2.5e-09	0.1(0.1)	1.3e-01	-9.5e-17
			3	86.2(0.0)	1.8e-04	-2.5e-12	0.8(0.0)	1.8e-03	-2.8e-16
			4				14.5(0.9)	4.8e-04	-1.4e-16
ex5_3_2	22	1	1	0.0(4.3)	4.6e-01	-6.7e-17	0.0(4.2)	4.6e-01	-6.7e-17
			2	43.2(0.1)	5.0e-06	-1.5e-09	0.9(5.7)	1.5e-01	-9.1e-17
			3				48.1(1.1)	1.3e-04	-3.3e-14
ex5_4_2	8	1	2	0.4(0.9)	5.1e-01	-2.0e-16	0.1(0.1)	5.2e-01	-4.8e-13
			3	65.7(0.0)	1.1e-05	-1.9e-16	0.5(0.0)	5.3e-08	-1.3e-14
alkyl	14	2	2	2.8(0.3)	1.0e-01	-1.3e-08	0.1(0.1)	1.5e-01	-1.4e-08
			3	8975.3(0.0)	1.4e-05	-3.6e-13	0.9(0.0)	8.2e-06	-1.1e-07

Table 3: Numerical results on the dense and sparse relaxations applied to unconstrained and constrained POPs.

			Dense SDP				Sparse SDP			
			Mat. var.		$\boldsymbol{B} = \boldsymbol{L}\boldsymbol{L}^T$		Mat. var.		$\boldsymbol{B} = \boldsymbol{L}\boldsymbol{L}^T$	
Problem	n	ω	no	max	size$\boldsymbol{L}$	nnz$\boldsymbol{L}$	no	max	size$\boldsymbol{L}$	nnz$\boldsymbol{L}$
Broyden Tridiag.	12	2	2	91	1819	1655290	11	10	214	5005
	24	2	2	325	20474	209602575	23	10	454	10885
	1000	2					999	10	19974	489125
	10000	2					9999	10	199974	4899125
Chained Wood	12	2	1	34	398	79401	11	4	53	261
	24	2	1	103	2989	4468555	23	4	107	531
	1000	2					999	4	4499	22491
	10000	2					9999	4	44999	224991
ex5_4_2	8	2	23	14	320	36315	25	7	97	1929
		3	23	83	2459	3023718	25	22	310	20496
alkyl	14	2	29	32	1181	308751	38	10	203	3886
		3	29	264	22071	125984806	38	28	834	64729

Table 4: Comparison of the dense and sparse SDP relaxations

7 An Application to the Sensor Network Localization Problem

We consider a sensor network localization (SNL) problem of n sensors in $\mathbb{R}^s$: Compute locations of sensors when distances between pairs of sensors located closely are available and locations of some of the sensors are provided. We present the full SDP (FSDP) relaxation [BY04] of the SNL problem and its sparse variant SFSDP relaxation [KKW09a] using the frameworks of the dense SDP relaxation in Section 2 and the sparse SDP relaxation in Section 3, respectively.

7.1 Quadratic Optimization Formulation of the SNL Problem

We assume that the location $\boldsymbol{a}_r \in \mathbb{R}^s$ of sensor r is known for $r = m+1, \ldots, n$. These sensors are called *anchors* in the subsequent discussion. We denote the number of anchors by $m_a (= n - m)$. Let $\rho > 0$ be a radio range, which determines the set $\mathcal{N}_x^\rho$ for pairs of sensors p and q such that their (Euclidean) distance d_{pq} does not exceed ρ, and the set $\mathcal{N}_a^\rho$ for pairs of a sensor p and an anchor r such that their distance d_{pr} does not exceed ρ;

$$\begin{aligned}
\mathcal{N}_x^\rho &= \{(p,q) : 1 \leq p < q \leq m, \|\boldsymbol{x}_p - \boldsymbol{x}_q\| \leq \rho\}, \\
\mathcal{N}_a^\rho &= \{(p,r) : 1 \leq p \leq m,\ m+1 \leq r \leq n, \|\boldsymbol{x}_p - \boldsymbol{a}_r\| \leq \rho\},
\end{aligned}$$

where $\boldsymbol{x}_p \in \mathbb{R}^s$ denotes the unknown location of sensor p ($p = 1, 2, \ldots, m$). Let $\mathcal{N}_x$ be a subset of $\mathcal{N}_x^\rho$ and $\mathcal{N}_a$ a subset of $\mathcal{N}_a^\rho$. By introducing zero objective function and the distance equations as constraints, we have the following form of SNL problem with exact distance:

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & \|\boldsymbol{x}_p - \boldsymbol{x}_q\|^2 = d_{pq}^2 \ (p,q) \in \mathcal{N}_x, \\ & \|\boldsymbol{x}_p - \boldsymbol{a}_r\|^2 = d_{pr}^2 \ (p,r) \in \mathcal{N}_a, \end{array}$$

or equivalently,

$$\left.\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & \boldsymbol{x}_p^T\boldsymbol{x}_p - 2\boldsymbol{x}_p^T\boldsymbol{x}_q + \boldsymbol{x}_q^T\boldsymbol{x}_q = d_{pq}^2 \ (p,q) \in \mathcal{N}_x, \\ & \boldsymbol{x}_p^T\boldsymbol{x}_p - 2\boldsymbol{a}_r^T\boldsymbol{x}_p + \boldsymbol{a}_r^T\boldsymbol{a}_r = d_{pr}^2 \ (p,r) \in \mathcal{N}_a. \end{array}\right\} \tag{25}$$

For problems with noise, we consider

$$\left.\begin{array}{ll} \text{minimize} & \displaystyle\sum_{(p,q)\in\mathcal{N}_x} \left(\xi_{pq}^+ + \xi_{pq}^-\right) + \sum_{(p,r)\in\mathcal{N}_a} \left(\xi_{pr}^+ + \xi_{pr}^-\right) \\ \text{subject to} & \boldsymbol{x}_p^T\boldsymbol{x}_p - 2\boldsymbol{x}_p^T\boldsymbol{x}_q + \boldsymbol{x}_q^T\boldsymbol{x}_q + \xi_{pq}^+ - \xi_{pq}^- = \hat{d}_{pq}^2 \ (p,q) \in \mathcal{N}_x, \\ & \boldsymbol{x}_p^T\boldsymbol{x}_p - 2\boldsymbol{a}_r^T\boldsymbol{x}_p + \boldsymbol{a}_r^T\boldsymbol{a}_r + \xi_{pr}^+ - \xi_{pr}^- = \hat{d}_{pr}^2 \ (p,r) \in \mathcal{N}_a, \\ & \xi_{pq}^+ \geq 0, \ \xi_{pq}^- \geq 0 \ (p,q) \in \mathcal{N}_x, \\ & \xi_{pr}^+ \geq 0, \ \xi_{pr}^- \geq 0 \ (p,r) \in \mathcal{N}_a, \end{array}\right\} \tag{26}$$

where $\xi_{pq}^+ + \xi_{pq}^-$ (or $\xi_{pr}^+ + \xi_{pr}^-$) indicates 1-norm error in the square of estimated distance $\hat{d}_{pq}$ between sensors p and q (or estimated distance $\hat{d}_{pr}$ between sensor p and anchor r).

7.2 Dense SDP Relaxation

We apply the dense SDP relaxation described in Section 2 to (25). The resulting SDP relaxation (27) coincides with FSDP relaxation proposed in the paper [BY04]. Obviously, $\omega_{\max} = 1$. Let $\omega = \omega_{\max}$. Then, the QOP (25) is transformed to an equivalent quadratic SDP (QSDP) as the POP (1) transformed to the PSDP (2). Notice in (25) that each vector variable $\boldsymbol{x}_p$ appears in the inner products but its elements x_{pi} $(i = 1, 2, \ldots, s)$ do not appear explicitly. Using this observation, we can modify the construction of an QSDP equivalent to (25). We consider the QSDP

$$\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & \boldsymbol{x}_p^T\boldsymbol{x}_p - 2\boldsymbol{x}_p^T\boldsymbol{x}_q + \boldsymbol{x}_q^T\boldsymbol{x}_q - d_{pq}^2 = 0 \ (p,q) \in \mathcal{N}_x, \\ & \boldsymbol{x}_p^T\boldsymbol{x}_p - 2\boldsymbol{a}_r^T\boldsymbol{x}_p + \boldsymbol{a}_r^T\boldsymbol{a}_r - d_{pr}^2 = 0 \ (p,r) \in \mathcal{N}_a, \\ & (\boldsymbol{I}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)^T(\boldsymbol{I}, \boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m) \succeq \boldsymbol{O}, \end{array}$$

which is equivalent to (25). Here $\boldsymbol{I}$ denotes the $s \times s$ identity matrix. Define an $s \times m$ matrix variable $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m)$. Then the last positive semidefinite constraint can be rewritten as

$$\begin{pmatrix} \boldsymbol{I} & \boldsymbol{X} \\ \boldsymbol{X}^T & \boldsymbol{X}^T\boldsymbol{X} \end{pmatrix} \succeq \boldsymbol{O}.$$

Replacing the quadratic term $\boldsymbol{X}^T\boldsymbol{X}$ by a matrix variable $\boldsymbol{Y} \in \mathbb{S}^m$ leads to the FSDP relaxation [BY04] of (25).

$$\left.\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Y_{pp} - 2Y_{pq} + Y_{qq} = d_{pq}^2 \ (p,q) \in \mathcal{N}_x, \\ & Y_{pp} - 2\boldsymbol{a}_r^T \boldsymbol{X}_{.p} + \boldsymbol{a}_r^T \boldsymbol{a}_r = d_{pr}^2 \ (p,r) \in \mathcal{N}_a, \\ & \begin{pmatrix} \boldsymbol{I} & \boldsymbol{X} \\ \boldsymbol{X}^T & \boldsymbol{Y} \end{pmatrix} \succeq \boldsymbol{O}, \end{array}\right\} \tag{27}$$

where $\boldsymbol{X}_{.p}$ denotes the pth column of $\boldsymbol{X}$, *i.e.*, $\boldsymbol{X}_{.p} = \boldsymbol{x}_p$ $(p = 1, 2, \ldots, m)$.

Similarly, we obtain the dense SDP relaxation of the SNL problem (26) with noise.

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{(p,q)\in\mathcal{N}_x} \left(\xi_{pq}^+ + \xi_{pq}^-\right) + \sum_{(p,r)\in\mathcal{N}_a} \left(\xi_{pr}^+ + \xi_{pr}^-\right) \\ \text{subject to} & Y_{pp} - 2Y_{pq} + Y_{qq} + \xi_{pq}^+ - \xi_{pq}^- = \hat{d}_{pq}^2 \ (p,q) \in \mathcal{N}_x, \\ & Y_{pp} - 2\boldsymbol{a}_r^T \boldsymbol{X}_{.p} + \boldsymbol{a}_r^T \boldsymbol{a}_r + \xi_{pr}^+ - \xi_{pr}^- = \hat{d}_{pr}^2 \ (p,r) \in \mathcal{N}_a, \\ & \xi_{pq}^+ \geq 0, \ \xi_{pq}^- \geq 0 \ (p,q) \in \mathcal{N}_x, \\ & \xi_{pr}^+ \geq 0, \ \xi_{pr}^- \geq 0 \ (p,r) \in \mathcal{N}_a, \\ & \begin{pmatrix} \boldsymbol{I} & \boldsymbol{X} \\ \boldsymbol{X}^T & \boldsymbol{Y} \end{pmatrix} \succeq \boldsymbol{O}. \end{array}$$

7.3 Sparse SDP Relaxation

We apply the sparse SDP relaxation described in Section 3 to (25). A chordal structured sparsity is extracted from (25) using the fact that the expression of (25) includes each vector variable $\boldsymbol{x}_p$ in the inner products, not individual element x_{pi} $(i = 1, 2, \ldots, s)$. Specifically, each vector variable $\boldsymbol{x}_p$ is regarded as a single variable when the csp graph $G(V, E)$ of (25) is constructed. Let $V = \{1, 2, \ldots, m\}$ and

$$E = \left\{ (p,q) \in V \times V : \begin{array}{l} p < q, \ \boldsymbol{x}_p \text{ and } \boldsymbol{x}_q \text{ are involved} \\ \text{in an equality constraint of (25).} \end{array} \right\}$$

By construction, we know that $E = \mathcal{N}_x$. Let $G(V, \bar{E})$ be a chordal extension of $G(V, E)$ and C_j $(j = 1, 2, \ldots, \ell)$ its maximal cliques. Then, consider the QSDP

$$\left.\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & \boldsymbol{x}_p^T \boldsymbol{x}_p - 2\boldsymbol{x}_p^T \boldsymbol{x}_q + \boldsymbol{x}_q^T \boldsymbol{x}_q - d_{pq}^2 = 0 \ (p,q) \in \mathcal{N}_x, \\ & \boldsymbol{x}_p^T \boldsymbol{x}_p - 2\boldsymbol{a}_r^T \boldsymbol{x}_p + \boldsymbol{a}_r^T \boldsymbol{a}_r - d_{pr}^2 = 0 \ (p,r) \in \mathcal{N}_a, \\ & \begin{pmatrix} \boldsymbol{I} & \boldsymbol{X}_{.C_j} \\ \boldsymbol{X}_{.C_j}^T & \boldsymbol{X}_{.C_j}^T \boldsymbol{X}_{.C_j} \end{pmatrix} \succeq \boldsymbol{O} \ (j = 1, 2, \ldots, \ell), \end{array}\right\} \tag{28}$$

which is equivalent to (25). Here $\boldsymbol{X}_{.C_j}$ denotes the submatrix of $\boldsymbol{X}$ consisting of column vectors $\boldsymbol{x}_p$ $(p \in C_j)$. We replace every $\boldsymbol{x}_p^T \boldsymbol{x}_q$ in (28) by a real variable Y_{pq} and define a matrix variable $\boldsymbol{Y}_{C_j C_j}$ of Y_{pq} $((p,q) \in C_j \times C_j)$ $(j = 1, 2, \ldots, \ell)$. Then, the resulting SDP is

$$\left.\begin{array}{ll} \text{minimize} & 0 \\ \text{subject to} & Y_{pp} - 2Y_{pq} + Y_{qq} - d_{pq}^2 = 0 \ \ (p,q) \in \mathcal{N}_x, \\ & Y_{pp} - 2\boldsymbol{a}_r^T \boldsymbol{X}_{.p} + \boldsymbol{a}_r^T \boldsymbol{a}_r - d_{pr}^2 = 0 \ (p,r) \in \mathcal{N}_a, \\ & \begin{pmatrix} \boldsymbol{I} & \boldsymbol{X}_{.C_j} \\ \boldsymbol{X}_{.C_j}^T & \boldsymbol{Y}_{C_jC_j} \end{pmatrix} \succeq \boldsymbol{O} \ (j = 1, 2, \ldots, \ell). \end{array}\right\} \tag{29}$$

The sparse SDP relaxation problem (29) is exactly the same as SFSDP relaxation problem proposed in Section 3.3 of [KKW09a] for a sparse variant of FSDP [BY04], although the derivation of SFSDP from FSDP there is different. It was also shown in [KKW09a] that SFSDP (*i.e*, (29)) is equivalent to FSDP (*i.e.*, (27)) in the sense that their feasible solution sets coincide with each other.

In the sparse SDP relaxation problem (29), we usually have $C_j \cap C_k \neq \emptyset$ for some distinct j and k. Thus, two positive semidefinite constraints

$$\begin{pmatrix} \boldsymbol{I} & \boldsymbol{X}_{.C_j} \\ \boldsymbol{X}_{.C_j}^T & \boldsymbol{Y}_{C_jC_j} \end{pmatrix} \succeq \boldsymbol{O} \ \text{ and } \begin{pmatrix} \boldsymbol{I} & \boldsymbol{X}_{.C_k} \\ \boldsymbol{X}_{.C_k}^T & \boldsymbol{Y}_{C_kC_k} \end{pmatrix} \succeq \boldsymbol{O}$$

share some variables X_{ip} $(i = 1, 2, \ldots, s, \ p \in C_j \cap C_k)$ and Y_{pq} $(p \in C_j \cap C_k, \ q \in C_j \cap C_k)$. Thus, the sparse SDP problem (29) is not a standard SDP. It should be converted to an equality standard form or an LMI standard form of SDP to apply the primal-dual interior-point method [Bor99, SDPA, Str99, TTT03, YFN10]. A simple method is to represent each matrix $\begin{pmatrix} \boldsymbol{O} & \boldsymbol{X}_{.C_j} \\ \boldsymbol{X}_{.C_j}^T & \boldsymbol{Y}_{C_jC_j} \end{pmatrix}$ as a linear combination of some constant matrices with the variables X_{ip} $(i = 1, 2, \ldots, s, \ p \in C_j)$ and Y_{pq} $((p,q) \in C_j \times C_j, \ p \leq q)$ in the matrix. See [KKM09] for more details on such conversions.

Although how $\mathcal{N}_x$ is selected from $\mathcal{N}_x^\rho$ and $\mathcal{N}_a$ from $\mathcal{N}_a^\rho$ is not mentioned, it is a very important issue since their choice determines both the chordal structured sparsity and the quality of the sparse SDP relaxation. As more edges from $\mathcal{N}_x^\rho$ are chosen for $\mathcal{N}_x$, the csp graph $G(V,E)$ becomes denser. (Recall that $E = \mathcal{N}_x$.) Conversely, if not enough edges from $\mathcal{N}_x^\rho$ are chosen for $\mathcal{N}_x$, then the quality of the resulting sparse SDP relaxation would be deteriorated. For details, we refer to [KKW09a, KKW09b].

7.4 Numerical Results on SFSDP

We report numerical results on the software package SFSDP [KKW09b, SFSDP], which is a Matlab implementation of the sparse SDP relaxation in the previous subsection. The package also includes the dense SDP relaxation in Section 7.2. We used a Matlab version of SDPA 7.3.1 as an SDP solver, and performed all numerical experiments on a 2.8GHz Intel Quad-Core i7 with 16GB memory. SNL problems with 1000 to 5000 sensors in $\mathbb{R}^3$ were used for numerical experiments. Sensors and anchors were distributed randomly in the unit cube $[0,1]^3$. The number of anchors was 10% or 5% of the number of sensors. The noisy factor σ was changed from 0.0 to 0.2, the radio range ρ was fixed to 0.25, and the distances were perturbed to create a noisy problem such that

$$\hat{d}_{pq} = \max\{(1+\sigma\epsilon_{pq}),\ 0.1\}\|\boldsymbol{a}_p - \boldsymbol{a}_q\| \ ((p,q) \in \mathcal{N}_x^{\rho}),$$
$$\hat{d}_{pr} = \max\{(1+\sigma\epsilon_{pr}),\ 0.1\}\|\boldsymbol{a}_p - \boldsymbol{a}_r\| \ ((p,r) \in \mathcal{N}_a^{\rho}),$$

where ϵ_{pq} and ϵ_{pr} were chosen from the standard normal distribution $N(0,1)$, and $\boldsymbol{a}_p$ denotes the true location of sensor p. To measure the accuracy of locations of m sensors computed by SDPA, and the accuracy of refined solutions by the gradient method [BLT06, LWY04], we used the root mean square distance (RMSD)

$$\left(\frac{1}{m}\sum_{p=1}^{m}\|\boldsymbol{x}_p - \boldsymbol{a}_p\|^2\right)^{1/2},$$

where $\boldsymbol{x}_p$ denotes the computed locations of sensor p.

Table 5 shows the performance of the dense and sparse SDP relaxations for solving SNL problems with 1000 sensors and 100 randomly distributed anchors. The noisy factor σ was changed from 0.0 to 0.2. SDPA eTime denotes the elapsed time by SDPA. We see that the sparse SDP relaxation solves the problems much faster than the dense SDP relaxation while achieving compatible accuracy as indicated in RMSD. The left figure of Fig. 5 shows the locations of anchors $\diamondsuit$, the true locations of sensors $\circ$ and the computed locations of sensors $\star$ for the problem with 1000 sensors, 100 anchors, $\sigma = 0.2$ and $\rho = 0.25$. We also observe that the maximum size of matrix variables and the number of nonzeros of the Cholesky factor $\boldsymbol{L}$ of the Schur complement matrix $\boldsymbol{B}$ are much smaller in the sparse SDP relaxation than those in the dense SDP relaxation. This results in much faster elapsed time by the sparse SDP relaxation. See also the right figure of Fig. 5.

Test problems		SDP	Mat. var.		$\boldsymbol{B} = \boldsymbol{L}\boldsymbol{L}^T$		RMSD		SDPA
$m,\ m_a,\ \rho$	σ	relaxation	no	max	size$\boldsymbol{L}$	nnz$\boldsymbol{L}$	SDP	w.Grad.	eTime
$m = 1000$,	0.0	Dense	1	1003	5554	30846916	5.2e-4	3.1e-5	189.7
$m_a = 100$		Sparse	962	27	7234	337084	2.3e-5	6.3e-6	5.7
distributed	0.1	Dense	1	1003	20034	401361156	5.5e-2	9.3e-3	744.1
randomly.		Sparse	962	27	20586	405785	5.5e-2	9.3e-3	7.11
$\rho = 0.25$	0.2	Dense	1	1003	20034	401361156	8.0e-2	2.2e-2	860.1
		Sparse	962	27	20586	405785	8.0e-2	2.2e-2	7.3

Table 5: Comparison of the dense and sparse SDP relaxations to solve 3-dimensional problems with 1000 sensors, 100 anchors, and $\rho = 0.25$.

We further tested the sparse SDP relaxation for SNL problems of 3000 and 5000 sensors, and showed the results in Table 6. The elapsed time by SDPA for solving the resulting SDPs remains short, obtaining accurate values of RMSD. We confirm that exploiting sparsity greatly reduces elapsed time.

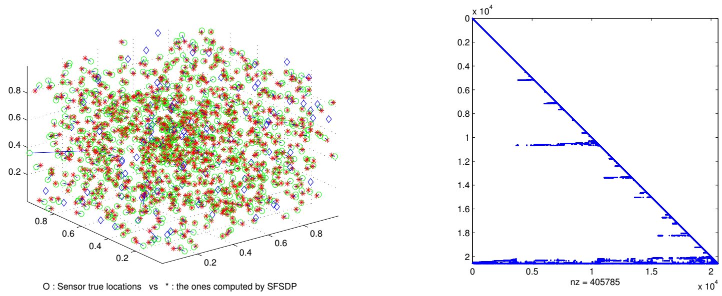

Figure 5: The depicted solutions and Cholesky factor of SNL problem of 1000 sensors, 100 anchors, $\rho = 0.25$, and $\sigma = 0.2$.

Test problems		SDP	Mat. var.		$\boldsymbol{B} = \boldsymbol{L}\boldsymbol{L}^T$		RMSD		SDPA
$m,\ m_a,\ \rho$	σ	relaxation	no	max	size$\boldsymbol{L}$	nnz$\boldsymbol{L}$	SDP	w.Grad.	eTime
$m = 3000$	0.0	Sparse	2966	28	18825	525255	4.0e-7	4.0e-7	12.6
$m_a = 300$	0.1	Sparse	2966	28	77369	833934	4.0e-2	1.3e-2	13.1
$\rho = 0.25$	0.2	Sparse	2966	28	77369	833934	6.8e-2	2.6e-2	13.5
$m = 3000$	0.0	Sparse	2955	28	19364	613190	2.6e-6	2.6e-6	18.9
$m_a = 150$	0.1	Sparse	2955	28	69336	873486	6.0e-2	1.5e-2	12.8
$\rho = 0.25$	0.2	Sparse	2955	28	69336	873486	8.4e-2	3.1e-2	13.1
$m = 5000$	0.0	Sparse	4969	25	30683	691716	4.7e-7	4.7e-7	27.8
$m_a = 500$	0.1	Sparse	4969	25	130405	1217155	3.0e-2	1.2e-2	46.8
$\rho = 0.25$	0.2	Sparse	4969	25	130405	1217155	5.4e-2	2.5e-2	40.0
$m = 5000$	0.0	Sparse	4970	26	30810	719808	1.1e-6	1.1e-6	29.3
$m_a = 250$	0.1	Sparse	4970	26	125282	1215716	3.3e-2	1.3e-2	42.4
$\rho = 0.25$	0.2	Sparse	4970	26	125282	1215716	5.6e-2	2.7e-2	46.4

Table 6: The sparse SDP relaxation to solve 3-dimensional SNL problems with 3000 sensors, 300 or 150 anchors, and $\rho = 0.25$ and 3-dimensional SNL problems with 5000 sensors, 500 or 250 anchors, and $\rho = 0.25$.

8 Concluding Discussions

We have presented a survey of the sparse SDP relaxation of POPs. The methods described in Section 3 for exploiting the sparsity characterized by the chordal graph structure were originally proposed for SDP problems [FFK00]. See also [NFF03]. Recently, they have been extended to nonlinear SDP problems in the paper [KKM09], which proposed conversion methods for nonlinear SDP problems, including linear and polynomial SDP problems, into smaller-sized problems, problems with smaller-sized matrix variables, and/or smaller-sized matrix inequalities.

Lasserre's dense SDP relaxation was extended to polynomial SDP problems [HL06, HS04, Koj04], and to more general POPs over symmetric cones [KM07]. The sparse SDP relaxation in Section 3 was also extended to polynomial optimization problems over symmetric cones [KM09]. When we deal with a polynomial SDP problem, we can first apply the conversion methods proposed in [KKM09] to the problem to reduce its size, and then apply the extended sparse SDP relaxation [KM09] to the reduced polynomial SDP problem. Some numerical results on quadratic SDP problems were shown in [KKM09].

References

[BLT06] Biswas, P., Liang, T.-C., Toh, K.-C., Wang, T.-C., Ye, Y.: Semidefinite programming approaches for sensor network localization with noisy distance measurements. IEEE Transactions on Automation Science and Engineering, **3**, 360–371 (2006)

[BY04] Biswas, P., Ye, Y.: Semidefinite programming for ad hoc wireless sensor network localization. In: Proceedings of the third international symposium on information processing in sensor networks. ACM press, 46–54 (2004)

[Bor99] Borchers, B.: SDPLIB 1.2, A library of semidefinite programming test problems. Optim. Methods Softw., **11-12**, 683–689 (1999)

[CLR95] Choi, M. D., Lam, T. Y., Reznick B.: Sums of squares of real polynomials. Proc. Symposia Pure Math., **58**, 103–126 (1995).

[CGT88] Conn, A. R., Gould, N. I. M., Toint, P. L.: Testing a class of methods for solving minimization problems with simple bounds on the variables. Math.Comp., **50**, 399–430 (1988)

[FFK00] Fujisawa, K., Fukuda, M., Kojima, M., Nakata, K.: Numerical evaluation of SDPA (SemiDefinite Programming Algorithm). In: Frenk, H., Roos, K., Terlaky, T., Zhang, S. (ed) High Performance Optimization. Kluwer Academic Press, 267–301 (2000)

[GAMS] GAMS HomePage http://www.gams.com/

[Glo] GLOBAL Library, `http://www.gamsworld.org/global/globallib.htm`

[Gol80] Golumbic, M. C.: Algorithmic graph theory and perfect graphs. Academic Press, New York (1980)

[HL03] Henrion, D., Lasserre, J.-B.: Detecting global optimality and extracting solutions in GloptiPoly. In: D. Henrion and A. Garulli, (ed) Positive Polynomials in Control. Lecture Notes in Control and Inform. Sci. 312, Springer-Verlag, Berlin, 293–310 (2005)

[HL06] Henrion, D., Lasserre, J.-B.: Convergent relaxations of polynomial matrix inequalities and static output feedback. IEEE Trans. Automatic. Cont., **51**, No.2, 192–202 (2006)

[HS04] Hol, C. W. J., Scherer, C. W.: Sum of squares relaxations for polynomial semidefinite programming. In: De Moor, B., Motmans, B. (ed) Proceedings of the 16th International Symposium on Mahematical Theory of Networks and Systems. Leuven, Belgium, July, 1–10 (2004)

[KKW05] Kim, S., Kojima, M., Waki, H.: Generalized Lagrangian duals and sums of squares relaxations of sparse polynomial optimization problems. SIAM J. Optim., **15**, 697–719 (2005)

[KKW09a] Kim, S., Kojima, M., Waki, H.: Exploiting sparsity in SDP relaxation for sensor network localization. SIAM J. Optim., **20**, 1, 192-215 (2009)

[KKW09b] Kim, S., Kojima, M., Waki, H., Yamashita, M.: SFSDP: a Sparse version of full semidefinite programming relaxation for sensor network localization problems. Research Report B-457, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo 152-8552 (2009)

[KKM09] Kim, S., Kojima, M., Mevissen, M., Yamashita, M.: Exploiting sparsity in linear and nonlinear matrix inequalities via positive semidefinite matrix completion. Research Report B-457, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo 152-8552 (2009)

[Koj04] Kojima, M.: Sums of squares relaxations of polynomial semidefinite programs. Research Report B-397, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo 152-8552, (2003)

[KKW03] Kojima, M., Kim, S., Waki, H.: Sparsity in sums of squares of polynomials. Math. Program., **103**, 45–62 (2005)

[KKK08] Kobayashi, K., Kim, S., Kojima, M.: Correlative sparsity in primal-dual interior-point methods for LP, SDP and SOCP. Appl. Math. Optim., **58**, 69–88 (2008)

[KM07] Kojima, M., Muramatsu, M.: An Extension of sums of squares relaxations to polynomial optimization problems over symmetric cones. Math. Program., **110**, 2, 315–336 (2007)

[KM09] Kojima, M., Muramatsu, M.: A note on sparse SOS and SDP relaxations for polynomial optimization problems over symmetric cones. Comp. Optim. Appl., **42**, 1, 31–41 (2009)

[Las01] Lasserre, J. B.: Global optimization with polynomials and the problems of moments. SIAM J. Optim., **11** 796–817 (2001)

[Las06] Lasserre, J. B.: Convergent SDP-relaxations in polynomial optimization with sparsity. SIAM J. Optim., **17**, 3, 822–843 (2006)

[Las10] Lasserre, J. B.: Moments, positive polynomials and their applications. Imperial College Press, London (2010)

[LWY04] Lian, T.-C., Wang, T.-C., Ye, Y.: A gradient search method to round the semidefinite programming relaxation solution for ad hoc wireless sensor network localization. Technical report, Dept. of Management Science and Engineering, Stanford University (2004)

[MFH81] More, J. J.,Garbow, B. S., Hillstrom, K. E.: Testing unconstrained optimization software. ACM Trans. Math. Soft., **7**, 17–41 (1981)

[NFF03] Nakata, K., Fujisawa, K., Fukuda, M., Kojima, M., Murota, M.: Exploiting sparsity in semidefinite programming via matrix completion II: Implementation and numerical results. Math. Program., **95**, 303–327 (2003)

[Par03] Parrilo, P. A.: Semidefinite programming relaxations for semialgebraic problems. Math. Programming, **96**, 293–320 (2003)

[PW98] Powers, V., Wörmann, T.: An algorithm for sums of squares of real polynomials. J. Pure Appl. Algebra, **127**, 99–104 (1998)

[Put93] Putinar, M.: Positive polynomials on compact semi-algebraic sets. Indiana U. Math. J., **42**, 969–984 (1993)

[SDPA] SDPA Online http://sdpa.indsys.chuo-u.ac.jp/sdpa/

[SFSDP] SFSDP Homepage http://www.is.titech.ac.jp/∼kojima/SFSDP/SFSDP.html

[SPOP] SparsePOP Homepage http://www.is.titech.ac.jp/∼kojima/SparsePOP/

[Str99] Strum, J. F.: SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw., **11 & 12**, 625–653 (1999)

[TTT03] Tütüncü, R. H., Toh, K. C., Todd, M. J.: Solving semidefinite-quadratic-linear programs using SDPT3. Math. Program., **95**, 189–217 (2003)

[WKK06] Waki, H., Kim, S. Kojima, M., Muramtasu, M.: Sums of squares and semidefinite programming relaxations for polynomial optimization problems with structured sparsity. SIAM J. Optim., **17**, 218–242 (2006)

[WKK08] Waki, H., Kim, S. Kojima, M., Muramtasu, M., Sugimoto, H.: Algorithm 883: SparsePOP: A sparse semidefinite programming relaxation of polynomial pptimization problems. ACM Trans. Math. Softw, **35**, 15 (2008)

[YFN10] Yamashita, M., Fujisawa, K., Nakata, K., Nakata, M., Fukuda, M., Kobayashi, K., Goto, K. : A high-performance software package for semidefinite programs: SDPA 7. Research Report B-460, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo 152-8552 (2009)