**Example 5.3** $g(x_1,x_2) = 2x_1^4 + 2x_1^3x_2 - x_1^2x_2^2 + 5x_2^4$ has a s.o.s. decomposition since

$$\begin{aligned} g(x_1,x_2) &= \begin{pmatrix} x_1^2 \\ x_2^2 \\ x_1x_2 \end{pmatrix}^T \begin{pmatrix} 2 & -3 & 1 \\ -3 & 5 & 0 \\ 1 & 0 & 5 \end{pmatrix} \begin{pmatrix} x_1^2 \\ x_2^2 \\ x_1x_2 \end{pmatrix} \\ &= \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}} \begin{pmatrix} x_1^2 \\ x_2^2 \\ x_1x_2 \end{pmatrix}^T \begin{pmatrix} 2 & 0 \\ -3 & 1 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 2 & -3 & 1 \\ 0 & 1 & 3 \end{pmatrix} \begin{pmatrix} x_1^2 \\ x_2^2 \\ x_1x_2 \end{pmatrix} \\ &= \frac{1}{2}(2x_1^2 - 3x_2^2 + x_1x_2)^2 + \frac{1}{2}(x_2^2 + 3x_1x_2)^2. \end{aligned}$$

It is known that a non-negative multivariate polynomial over reals can be represented by a sum of rational functions (Emil Artin 1927. It is known as Hilbert' 17th Problem).

Of course, every s.o.s polynomial is a non-negative polynomial. On the other hand, the Motzkin polynomial (1965) is an example of a non-negative polynomial which can not be written as an s.o.s.

$$p(x_1,x_2) = 1 + x_1^2x_2^2(x_1^2 + x_2^2 - 3)$$

Although this is true, it is also known that small perturbations of non-negative polynomials can be approximated asymptotically by s.o.s. [Lasserre].

Now, suppose we want to find a minimum of a multivariate polynomial $p \in \mathbb{R}[\boldsymbol{x}]_{2d}$ of degree $2d$.

$$(\text{POP}) \begin{cases} \text{minimize} & p(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{x} \in \mathbb{R}^n \end{cases}$$

This problem is equivalent to

$$(\text{POP}) \begin{cases} \text{maximize} & \lambda \\ \text{subject to} & p(\boldsymbol{x}) - \lambda \geq 0 \\ & \lambda \in \mathbb{R},\ \forall \boldsymbol{x} \in \mathbb{R}^n. \end{cases}$$

Since every s.o.s. polynomial is a non-negative polynomial, we can consider the following semidefinite program relaxation of this Polynomial Optimization Problem (POP):

$$(\text{SOS}_d) \begin{cases} \text{maximize} & \lambda \\ \text{subject to} & p(\boldsymbol{x}) - \lambda \in \Sigma[\boldsymbol{x}]_{2d}, \end{cases}$$

where $\Sigma[\boldsymbol{y}]_d \subseteq \mathbb{R}[\boldsymbol{y}]_d$ denotes the set of all s.o.s polynomials of degree $d$ or less.

We will show now that $(\text{SOS}_d)$ is an semidefinite program.

By Proposition 5.2, $\exists \boldsymbol{Q} \in \mathcal{S}_+^{s(d)}$ such that $p(\boldsymbol{x}) = \sum_{|\boldsymbol{\alpha}| \leq 2d} p_{\boldsymbol{\alpha}} \boldsymbol{x}^{\boldsymbol{\alpha}} = \boldsymbol{v}_d(\boldsymbol{x})^T \boldsymbol{Q} \boldsymbol{v}_d(\boldsymbol{x}) = \langle \boldsymbol{Q}, \boldsymbol{v}_d(\boldsymbol{x})\boldsymbol{v}_d(\boldsymbol{x})^T \rangle$.

This means that for every $\boldsymbol{\alpha}$ such that $|\boldsymbol{\alpha}| \leq 2d$, $q_{\boldsymbol{\alpha}} = \sum_{ij} Q_{ij}$, for appropriate indices $i$ and $j$.

In conclusion, the condition $p(\boldsymbol{x}) - \lambda \in \Sigma[\boldsymbol{x}]_{2d}$ can be replaced by $\boldsymbol{Q} \in \mathcal{S}_+^{s(d)}$ and $\sum_{ij} Q_{ij} = p_{\boldsymbol{\alpha}} \quad (|\boldsymbol{\alpha}| \leq d)$, which turns $(\text{SOS}_d)$ a semidefinite program.

$$(\text{SOS}_d) \begin{cases} \text{maximize} & p(\boldsymbol{0}) - Q_{11} \\ \text{subject to} & \sum_{ij} Q_{ij} = p_{\boldsymbol{\alpha}}, \quad |\boldsymbol{\alpha}| \leq 2d \\ & \boldsymbol{Q} \in \mathcal{S}_+^{s(d)} \end{cases}$$

As a consequence:

$$(\text{SOS}_d) \leq (\text{SOS}_{d+1}) \leq \ldots \leq p^*,$$

where $p^*$ is the optimal value of (POP).

Of course, as $d$ increases, the size of the semidefinite program increases, turning the problem hard (or even impossible) to solve numerically. Also solving $(\text{SOS}_d)$, we can only obtain a lower bound for the optimal value $p^*$ of (POP). To obtain also an approximate solution, it is necessary to solve its dual problem which has an interpretation as an optimization problem over the moment (matrix).

It is also possible to extend POP for polynomially constrained problems:

$$\begin{cases} \text{minimize} & p(\boldsymbol{x}) \\ \text{subject to} & \boldsymbol{x} \in \mathbb{K}, \end{cases}$$

where $\mathbb{K} := \{\boldsymbol{x} \in \mathbb{R}^n \mid g_i(\boldsymbol{x}) \geq 0,\ i = 1, 2, \ldots, m\}$ is a closed basic semi-algebraic set which additionally needs to be assumed to be a bounded set. See [Lasserre] for details.

# 6 Second-Order Cone Program Relaxation

Any quadratically constrained quadratic program can be written w.l.o.g in the form:

$$\begin{cases} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \boldsymbol{x}^T \boldsymbol{Q}_i \boldsymbol{x} + 2\boldsymbol{q}_i^T \boldsymbol{x} + \gamma_i \leq 0, \quad i = 1, 2, \ldots, m. \end{cases} \tag{10}$$

Observe that

$$\boldsymbol{x}_i^T \boldsymbol{Q}_i \boldsymbol{x} + 2\boldsymbol{q}_i^T \boldsymbol{x} + \gamma_i = \left\langle \begin{pmatrix} \gamma_i & \boldsymbol{q}_i^T \\ \boldsymbol{q}_i & \boldsymbol{Q}_i \end{pmatrix}, \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \right\rangle \quad (i = 1, 2, \ldots, m)$$

Similar to the max-cut problem, introducing the new variable $\boldsymbol{X} \in \mathbb{R}^{n \times n}$, the condition $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T = \boldsymbol{O}$ can be replaced by $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \in \mathcal{S}_+^n$, which in turn is equivalent to the condition

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}_+^{n+1}.$$

Therefore, we can obtain the following semidefinite program relaxation of (10).

$$\begin{cases} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \left\langle \begin{pmatrix} \gamma_i & \boldsymbol{q}_i^T \\ \boldsymbol{q}_i & \boldsymbol{Q}_i \end{pmatrix}, \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \right\rangle \leq 0, \quad i = 1, 2, \ldots, m \\ & \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}_+^{n+1}. \end{cases} \tag{11}$$

In general, the above semidefinite problem relaxation of (10) is better than the RLT (Reformulation Linearization Technique) on the same problem in the sense that it gives a better lower bound for the objective function.

Now, since $\mathcal{S}_+^n$ is a self-dual cone, we can say that

$$\boldsymbol{Y} \in \mathcal{S}_+^n \Leftrightarrow \langle \boldsymbol{C}, \boldsymbol{Y} \rangle \geq 0,\ \forall \boldsymbol{C} \in \mathcal{S}_+^n.$$

Therefore, if we choose $\ell$ matrices $\boldsymbol{C}_j \in \mathcal{S}_+^n \quad (j = 1, 2, \ldots, \ell)$ and replace the condition $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \in \mathcal{S}_+^n$ by $\langle \boldsymbol{C}_j, \boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \rangle \geq 0$, we will have a further relaxation of (11).

$$\begin{cases} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} \\ \text{subject to} & \left\langle \begin{pmatrix} \gamma_i & \boldsymbol{q}_i^T \\ \boldsymbol{q}_i & \boldsymbol{Q}_i \end{pmatrix}, \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \right\rangle \leq 0, \quad i = 1, 2, \ldots, m \\ & \boldsymbol{x}^T \boldsymbol{C}_j \boldsymbol{x} - \langle \boldsymbol{C}_j, \boldsymbol{X} \rangle \leq 0, \qquad j = 1, 2, \ldots, \ell \end{cases} \tag{12}$$

The advantage of (12) over (11) is that the former is a second-order cone program (SOCP) instead of semidefinite program (SDP). The SOCP is extremely cheap in terms of computational cost (but not cheaper than linear program).

For instance, if $\boldsymbol{Q}_i \notin \mathcal{S}_+^n$, we will have $\boldsymbol{Q}_i = \sum_{k=1}^{n} \lambda_{ik} \boldsymbol{u}_{ik} \boldsymbol{u}_{ik}^T$ with $\lambda_{i1} \geq \lambda_{i2} \geq \lambda_{is_i} \geq 0 > \lambda_{i,s_i+1} \geq \ldots \geq \lambda_{in}$. Therefore, $\boldsymbol{C}_i = \sum_{k=1}^{s_i} \lambda_{ik} \boldsymbol{u}_{ik} \boldsymbol{u}_{ik}^T - \sum_{k=s_i+1}^{n} \lambda_{ik} \boldsymbol{u}_{ik} \boldsymbol{u}_{ik}^T \in \mathcal{S}_+^n$ will be a good candidate for (12). Further, since $\boldsymbol{C}_i \in \mathcal{S}_+^n$, it also can be written as $\boldsymbol{C}_i = \boldsymbol{L}_i \boldsymbol{L}_i^T$.

The next lemma will be useful for further discussion.

**Lemma 6.1** Given $\boldsymbol{w} \in \mathbb{R}^n$, $\eta, \zeta \in \mathbb{R}$,

$$\boldsymbol{w}^T \boldsymbol{w} \leq \eta\zeta, \text{ and } \eta \geq 0, \ \zeta \geq 0 \Leftrightarrow \left\| \begin{pmatrix} \eta - \zeta \\ 2\boldsymbol{w} \end{pmatrix} \right\|_2 \leq \eta + \zeta$$

*Proof:*

$$4\boldsymbol{w}\boldsymbol{w}^T \leq 4\eta\zeta \Leftrightarrow (\eta - \zeta)^2 + 4\boldsymbol{w}\boldsymbol{w}^T \leq (\eta + \zeta)^2$$

∎

Finally, using Lemma 6.1, we obtain the following second-order cone program relaxation of (10).

$$\left\{ \begin{array}{lll} \text{minimize} & \boldsymbol{c}^T \boldsymbol{x} & \\ \text{subject to} & \left\langle \begin{pmatrix} \gamma_i & \boldsymbol{q}_i^T \\ \boldsymbol{q}_i & \boldsymbol{Q}_i \end{pmatrix}, \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \right\rangle \leq 0, & i = 1, 2, \ldots, m \\ & \begin{pmatrix} \langle \boldsymbol{C}_j, \boldsymbol{X} \rangle + 1 \\ \langle \boldsymbol{C}_j, \boldsymbol{X} \rangle - 1 \\ 2\boldsymbol{L}_i^T \boldsymbol{x} \end{pmatrix} \in \mathbb{Q}_+^{n+2}, & i = 1, 2, \ldots, m \end{array} \right.$$

where $\mathbb{Q}_+^n = \{\boldsymbol{x} \in \mathbb{R}^n \mid x_1^2 \geq \sum_{i=2}^{n-1} x_i^2\}$.