A Lagrangian relaxation view of linear and semidefinite hierarchies

Jean B. Lasserre

Abstract—Consider the polynomial optimization problem

\[
P : f^* = \min_{x} \{ f(x) : x \in K \}
\]

where \( K \) is a compact basic semi-algebraic set. We first show that the standard Lagrangian relaxation yields a lower bound as close as desired to the global optimum \( f^* \), provided that it is applied to a problem \( P \) equivalent to \( P \), in which sufficiently many redundant constraints (products of the initial ones) are added to the initial description of \( P \). Next we show that the standard hierarchy of LP-relaxations of \( P \) (in the spirit of Sherali-Adams' RLT) can be interpreted as a brute force simplification of the above Lagrangian relaxation. So we provide a systematic improvement of the LP-hierarchy by doing a much less brutal simplification which results into a parametrized hierarchy of semidefinite programs (and not linear programs any more). For each semidefinite program in the hierarchy parametrized by \( k \), the semidefinite constraint has a fixed size \( O(n^k) \), independently of the rank in the hierarchy, in contrast with the standard hierarchy of semidefinite relaxations.

I. INTRODUCTION

Recent years have seen the development of (global) semi-algebraic optimization and in particular LP- or semidefinite relaxations for the polynomial optimization problem:

\[
P : f^* = \min_{x} \{ f(x) : x \in K \}
\]

where \( f \in \mathbb{R}[x] \) is a polynomial and \( K \subset \mathbb{R}^n \) is the basic semi-algebraic set

\[
K = \{ x \in \mathbb{R}^n : g_j(x) \geq 0, \quad j = 1, \ldots, m \}
\]

for some polynomials \( g_j \in \mathbb{R}[x], j = 1, \ldots, m \). Associated with \( P \) are two hierarchies of convex relaxations:

- **Semidefinite** relaxations based on Putinar’s certificate of positivity on \( K \) \cite{Putinar93}, where the \( d \)-th convex relaxation of the hierarchy is a semidefinite program which solves the optimization problem

\[
\gamma_d = \max_{t,\sigma_j} \left\{ t : f - t = \sigma_0 + \sum_{j=1}^{m} \sigma_j g_j \right\}
\]

The unknowns \( \sigma_j \) are sums of squares polynomials with the degree bound constraint \( \deg \sigma_j g_j \leq 2d, j = 0, \ldots, m \), and the expression in (3) is a certificate of positivity on \( K \) for the polynomial \( x \mapsto f(x) - t \).

- **LP-relaxations** based on Krivine-Stengle’s certificate of positivity on \( K \) \cite{KrivineStengle86, Lasserre01}, where the \( d \)-th convex relaxation of the hierarchy is a linear program which solves the optimization problem

\[
\theta_d = \max_{\lambda \geq 0, t} \left\{ t : f - t = \sum_{(\alpha,\beta) \in \mathbb{N}_d^2} \lambda_{\alpha\beta} \prod_{j=1}^{m} (g_j^{\alpha_j} (1 - g_j)^{\beta_j}) \right\},
\]

where \( \mathbb{N}_d^2 = \{(\alpha,\beta) \in \mathbb{N}^2 : \sum_j \alpha_j + \beta_j \leq d\} \). The unknowns are \( t \) and the nonnegative scalars \( \lambda = (\lambda_{\alpha\beta}) \), and it is assumed that \( 0 \leq g_j \leq 1 \) on \( K \) (possibly after scaling) and the family \( \left\{ g_j, 1 - g_j \right\} \) generates the algebra \( \mathbb{R}[x] \) of polynomials.

For instance, the LP-hierarchy from Sherali-Adams’ RLT \cite{SheraliAdams90} and their variants are of this form. (See more details in §III-C.)

In both cases, \( (\gamma_d) \) and \( (\theta_d) \), \( d \in \mathbb{N} \), provide two monotone nondecreasing sequences of lower bounds on \( f^* \) if \( K \) is compact then both converge to \( f^* \) as one let \( d \) increase. For more details as well as a comparison of such relaxations the interested reader is referred to e.g. Lasserre \cite{Lasserre01}, \cite{Lasserre03} and Laurent \cite{Laurent09}, as well as Chlamtac and Tulsiani \cite{ChlamtacTulsiani08} for the impact of LP- and SDP-hierarchies on approximation algorithms in combinatorial optimization.

Of course, in principle, one would much prefer to solve LP-relaxations rather than semidefinite relaxations (i.e. compute \( \theta_d \) rather than \( \gamma_d \)) because present LP-software packages can solve problems with millions of variables and constraints, which is far from being the case for semidefinite solvers. However, on the other hand, the LP-relaxations (4) suffer from several serious theoretical and practical drawbacks. For instance, it has been shown in \cite{Lasserre03}, \cite{Lasserre01} that the LP-relaxations cannot be exact for most convex problems, i.e., the sequence of the associated optimal values converges to the global optimum only asymptotically and not in finitely many steps. Moreover, the LPs of the hierarchy are numerically ill-conditioned. This is in contrast with the semidefinite relaxations (3) for which finite convergence takes place for convex problems where \( \nabla^2 f(x^*) \) is positive definite at every minimizer \( x^* \in K \) (see de Klerk and Laurent \cite[Corollary 3.3]{deKlerkLaurent06}) and occurs at the first relaxation for SOS-convex problems \cite[Theorem 3.3]{Nie09}. In fact, as demonstrated in a recent work of Nie \cite{Nie11}, finite convergence is generic (even for non convex problems).

So would it be possible to define a hierarchy of convex relaxations in between (3) and (4), i.e., with some of the

\[\footnote{An SOS-convex polynomial is a convex polynomial whose Hessian factors as \( L(x)L(x)^T \) for some rectangular matrix polynomial \( L \). For instance, separable convex polynomials are SOS-convex.}\]
nice features of the semidefinite relaxations but with a much less demanding computational effort (hence closer to the LP-relaxations)? This paper is a contribution in this direction.

**Contribution.** We provide one theoretical and one algorithmic contribution. First we describe a new hierarchy of convex relaxations for $P$ with the following feature. Each relaxation in the hierarchy is a finite-dimensional convex optimization problem parametrized by $d \in \mathbb{N}$, and of the form:

$$\rho_d = \max_{\lambda} \{ G_d(\lambda) : \lambda \geq 0 \},$$

(5)

where $G_d(\cdot)$ is the concave function defined by:

$$G_d(\lambda) := \min_x \{ f(x) - L_d(x, \lambda) \}$$

with

$$L_d(x, \lambda) := \sum_{(\alpha,\beta) \in \mathbb{N}^m} \lambda_{\alpha \beta} \prod_{j=1}^m (g_j^\alpha x_j + g_j^\beta x_j)$$

(7)

for nonnegative $\lambda \in \mathbb{R}^s(d)$ with $s(d) = (2m+d)$. Therefore $\rho_d \leq f^*$ for all $d$ and we prove that:

(a) $\rho_d \geq \theta_d$ for all $d$, and so $\rho_d \to f^*$ as one let $d$ increase.

(b) For convex problems $P$, i.e., when $f_i - g_j$ are convex, form the most natural class of discrete optimization that their Lagrangian relaxations are convex and occur at the first relaxation, i.e., $\rho_1 = f^*$.

(c) For 0/1 optimization, i.e., when $K \subseteq \{0,1\}^n$, finite convergence takes place and the optimal value $\rho_d$ provides a better lower bound than the one obtained with Sherali-Adams’ RLT hierarchy.

(d) Finally (5) has a nice interpretation.

Indeed solving (5) is just applying the dual method of multipliers in Non Linear Programming (i.e., Lagrangian relaxation) to a problem $P_d$ obtained from $P$ by adding new (but redundant) constraints formed with some products of the original constraints (hence $P_d$ is equivalent to $P$ and has same optimal value). And so the Lagrangian relaxation applied to $P_d$ provides a lower bound close to $f^*$ as desired, $P_d$ is sufficiently large, i.e., provided that sufficiently many redundant constraints are added to the description of $P$. This provides a rigorous rationale for the well-known fact that adding redundant constraints helps for solving $P$. Indeed, even though the new problems $P_d$, $d \in \mathbb{N}$, are all equivalent to $P$, it is part of the folklore of discrete optimization that their Lagrangian relaxations are not equivalent to that of $P$.

Even though (5) is a convex optimization problem, evaluating $G_d(\lambda)$ at a point $\lambda \geq 0$ requires computing the unconstrained global minimum of the Lagrangian $L_d(\cdot, \lambda)$ in (7), an NP-hard problem in general. After all, in principle the goal of Lagrangian relaxation is to end up with a problem which is easier to solve than $P$, and so, in this respect, the hierarchy (5) is not practical.

So in a second algorithmic contribution we show that the LP-relaxations (4) can be interpreted as a way to “restrict” and simplify the hierarchy (5) by a simple and brute force trick, so as to make it tractable (but of course less efficient). Inspired by this interpretation, we next propose a systematic way to define improved versions of the LP-hierarchy (4) by simplifying (5) in a much less brutal manner. The increase of complexity is completely controlled by a parameter $k \in \mathbb{N}$ chosen by the user. That is, in the new resulting hierarchy parametrized by $k$, each LP of the hierarchy (4) now becomes a semidefinite program but whose size of the semidefiniteness constraint is fixed and equal to $(\frac{n+k}{d})$, independently of the rank $d$ in the hierarchy. When $k = 1$ one obtains the so-called “Sherali-Adams + SDP” hierarchy in 0/1 optimization.

**But what do we gain by this increase of complexity?**

Of course, from a computational complexity point of view, one way to evaluate the efficiency of those relaxations is to analyze whether they help reduce integrality gaps, e.g. for some 0/1 optimization problems. For the level $k = 1$ (i.e. the “Adams-Sherali + SDP hierarchy”) some negative results in this direction have been already provided in Benabbas and Magen [2], and in Benabbas et al. [3].

But in a different point of view, we claim that a highly desirable property for a general purpose method (e.g., the hierarchies (3) or (4)) aiming at solving NP-hard optimization problems, is to behave “efficiently” when applied to a class of problems considered relatively “easy” to solve. Otherwise one might raise reasonable doubts on its efficiency for more difficult problems, not only in a worst-case sense but also in “average”. Convex problems $P$ as in (1)-(2), i.e., when $f_i - g_j$ are convex, form the most natural class of problems which are considered easy to solve by some standard methods of Non Linear Programming; see e.g. Ben-tal and Nemirovski [1]. Recall that the semidefinite hierarchy (3) somehow recognizes convexity (see e.g. [5], [8]) whereas the LP-hierarchy (4) behaves poorly on such problems (see e.g. [7], [9]). So we prove that the gain by this (controlled) increase of complexity is precisely to permit finite convergence for a non trivial class of convex problems.

II. MAIN RESULT

A. Notation and definitions

Let $\mathbb{R}[x]$ be the ring of polynomials in the variables $x = (x_1, \ldots, x_n)$. Denote by $\mathbb{R}[x]_d \subset \mathbb{R}[x]$ the vector space of polynomials of degree at most $d$, which forms a vector space of dimension $s(d) = \binom{n+d}{d}$, with e.g., the usual canonical basis $(x^n)$ of monomials. Also, denote by $\Sigma[x] \subset \mathbb{R}[x]$ (resp. $\Sigma[x]_d \subset \mathbb{R}[x]_d$) the space of sums of squares (s.o.s.) polynomials (resp. s.o.s. polynomials of degree at most $2d$). If $f \in \mathbb{R}[x]_d$, write $f(x) = \sum_{\alpha \in \mathbb{N}^m} f_\alpha x^\alpha$ in the canonical basis and denote by $f = (f_\alpha) \in \mathbb{R}^s(d)$ its vector of coefficients. Finally, let $S^n$ denote the space of $n \times n$ real symmetric matrices, with inner product $\langle A, B \rangle = \text{trace} \ AB$, and where the notation $A \succeq 0$ (resp. $A \succ 0$) stands for $A$ is positive semidefinite. With $g_0 := 1$, the quadratic module $Q(g_1, \ldots, g_m) \subset \mathbb{R}[x]$ generated by polynomials $g_1, \ldots, g_m$, is defined by

$$Q(g_1, \ldots, g_m) := \left\{ \sum_{j=0}^m \sigma_j g_j : \sigma_j \in \Sigma[x] \right\}.$$
We briefly recall two important theorems by Putinar [12] and Krivine-Stengle [6], [14] respectively, on the representation of polynomials positive on $K$.

**Theorem 1:** Let $g_0 = 1$ and $K$ in (2) be compact.

(a) If the quadratic polynomial $x \mapsto M - ||x||^2$ belongs to $Q(g_1, \ldots, g_m)$ and if $f \in \mathbb{R}[x]$ is strictly positive on $K$ then $f \in Q(g_1, \ldots, g_m)$.

(b) Assume that $0 \leq g_j \leq 1$ on $K$ for every $j$, and the family $\{g_j, 1 - g_j\}$ generates $\mathbb{R}[x]$. If $f$ is strictly positive on $K$ then
\[
 f = \sum_{\alpha, \beta \in \mathbb{N}_m^2} c_{\alpha, \beta} \prod_j (g_j^{\alpha_j} (1 - g_j)^{\beta_j}),
\]
for some finitely many nonnegative scalars $(c_{\alpha, \beta})$.

**B. Main result**

With $K$ as in (2) we make the following assumption:

**Assumption 1** $K$ is compact and $0 \leq g_j \leq 1$ on $K$ for all $j = 1, \ldots, m$. Moreover, the family of polynomials $\{g_j, 1 - g_j\}$ generates the algebra $\mathbb{R}[x]$.

Notice that if $K$ is compact and Assumption 1 does not hold, one may always rescale the variables $x_i$ so as to have $K \subset [0, 1]^n$, and then add redundant constraints $0 \leq x_i \leq 1$ for all $i = 1, \ldots, m$. Then the family $\{g_j, 1 - g_j\}$ (which includes $x_j$ and $1 - x_j$ for all $j$) generates the algebra $\mathbb{R}[x]$ and Assumption 1 holds.

With $d \in \mathbb{N}$ and $0 \leq \lambda = (\lambda_{\alpha, \beta}), (\alpha, \beta) \in \mathbb{N}_m^2$, let $\lambda \mapsto G_d(\lambda)$ be the function defined in (6), with associated problem:

\[
 \rho_d = \max_{\lambda} \{ G_d(\lambda) : \lambda \geq 0 \}.
\]

Observe that $G_d(\lambda) \leq f^*$ for all $\lambda \geq 0$, and computing $\rho_d$ is just solving the Lagrangian relaxation of the problem:

\[
 \hat{P}_d : \min_{x} \{ f(x) : \prod_{j=1}^m g_j(x)^{\alpha_j} (1 - g_j(x))^{\beta_j} \geq 0 \}.
\]

**Theorem 2:** Let $K$ be as in (2), $f \in \mathbb{R}[x], d \in \mathbb{N}$, and let Assumption 1 hold. Consider problem (8) associated with $\hat{P}_d$ and with optimal value $\rho_d$. Then the sequence $(\rho_d), d \in \mathbb{N}$, is monotone nondecreasing and $\rho_d \rightarrow f^*$ as $d \rightarrow \infty$.

**Proof:** We first prove that $\rho_{d+1} \geq \rho_d$ for all $d$, so that the sequence $(\rho_d), d \in \mathbb{N}$, is monotone nondecreasing. Let $0 \leq \lambda = (\lambda_{\alpha, \beta})$ with $(\alpha, \beta) \in \mathbb{N}_m^2$. Then $0 \leq \lambda$ with $\lambda_{\alpha, \beta} = \lambda_{\alpha, \beta}$ whenever $(\alpha, \beta) \in \mathbb{N}_m^2$ and $\lambda_{\alpha, \beta} = 0$ whenever $|\alpha + \beta| > d$, is such that $G_{d+1}(\lambda) = G_d(\lambda)$ and so $\rho_{d+1} \geq \rho_d$. Next, let $\epsilon > 0$ be fixed, arbitrary. The polynomial $f - (f^* + \epsilon)$ is positive on $K$ and therefore, by [14], [9, Theorem 2.23],

\[
 f - (f^* + \epsilon) = \sum_{(\alpha, \beta) \in \mathbb{N}_m^2} c_{\alpha, \beta} \prod_{j=1}^m (g_j^{\alpha_j} (1 - g_j)^{\beta_j}),
\]

for some nonnegative vector of coefficients $c^* = (c_{\alpha, \beta}^*)$. Equivalently,

\[
 f - \sum_{(\alpha, \beta) \in \mathbb{N}_m^2} c_{\alpha, \beta} \prod_{j=1}^m (g_j^{\alpha_j} (1 - g_j)^{\beta_j}) = (f^* - \epsilon).
\]

Letting $d_\epsilon := \max_{\alpha, \beta} \{|\alpha + \beta| : c_{\alpha, \beta} > 0\}$, we obtain $f^* \geq G_{d_\epsilon}(c^*) = f^* - \epsilon$. And so

\[
 f^* \geq \max_{\lambda} \{ G_d(\lambda) : \lambda \geq 0 \} \geq f^* - \epsilon.
\]

As $\epsilon > 0$ was arbitrary, the desired result follows.

**Corollary 1:** Let $K$ be as in (2), Assumption (1) hold and let $P_d, d \in \mathbb{N}$, be as in (9). Then for every $\epsilon > 0$ there exists $d_\epsilon \in \mathbb{N}$ such that for every $d \geq d_\epsilon$, the Lagrangian relaxation of $\hat{P}_d$ yields a lower bound $f^* - \epsilon \leq \rho_d \leq f^*$. This follows from Theorem 2 and the fact that computing $\rho_d$ is just solving the Lagrangian relaxation associated with $\hat{P}_d$. So the interpretation of Corollary 1 is that the Lagrangian relaxation technique in non convex optimization can provide a lower bound as close as desired to the global optimum $f^*$ provided that it is applied to an equivalent formulation of $P$ that contains sufficiently many redundant constraints which are products of the original ones. It also provides a rigorous rationale for the well-known fact that adding redundant constraints helps solve $P$. Indeed, even though the new problems $P_d, d \in \mathbb{N}$, are all equivalent to $P$, their Lagrangian relaxations are not equivalent to that of $P$.

**C. Convex programs**

In this section, the set $K$ is not assumed to be compact.

**Theorem 3:** Let $K$ be as in (2) and assume that $f$ and $-g_j$ are convex, $j = 1, \ldots, m$. Moreover, assume that Slater’s condition holds and $f^* > -\infty$. Then the hierarchy of convex relaxations (5) has finite convergence at step $d = 1$, i.e., $\rho_1 = f^*$, and $\rho_1 = G_1(\lambda^*)$ for some nonnegative $\lambda^* \in \mathbb{R}_m^m$.

**Proof:** This is because the dual method applied to $P$ (i.e. $P_1$) converges, i.e.,

\[
 f^* = \max_{\lambda \geq 0} \left\{ \min_{x} \{ f(x) - \sum_{j=1}^m \lambda_j g_j(x) \} \right\} = \max_{\lambda} \{ G_1(\lambda) : \lambda \geq 0 \} = \rho_1.
\]

Next, let $\lambda^{(n)}$ be a maximizing sequence, i.e., $G_1(\lambda^{(n)}) \rightarrow f^*$ as $n \rightarrow \infty$. Since Slater’s condition holds (say at some $x_0 \in K$), one has

\[
 G_1(\lambda^{(0)}) \leq G_1(\lambda^{(n)}) \leq f(x_0) - \sum_{j=1}^m \lambda_j^{(n)} g_j(x_0),
\]

for all $n$, and so $\lambda_j^{(n)} \leq (f(x_0) - G_1(\lambda^{(0)}))/g_j(x_0)$ for every $j = 1, \ldots, m$, and all $n \geq 1$. So there is a subsequence $(\lambda_k)$, $k \in \mathbb{N}$, and $\lambda^* \in \mathbb{R}_m^m$, such that $\lambda^{(n_k)} \rightarrow \lambda^* \geq 0$ as $k \rightarrow \infty$. Finally, let $x \in \mathbb{R}_n$ be fixed, arbitrary. From

\[
 G_1(\lambda^{(n_k)}) \leq f(x) - \sum_{j=1}^m \lambda_j^{(n_k)} g_j(x), \quad \forall k,
\]

letting $k \rightarrow \infty$ yields $f^* \leq f(x) - \sum_{j=1}^m \lambda_j^{(n_k)} g_j(x)$. As $x \in \mathbb{R}_n$ was arbitrary, this proves $G_1(\lambda^*) \geq f^*$, which combined

---

1668
with \( G_1(\lambda^*) \leq f^* \) yields the desired result \( G_1(\lambda^*) = f^* \).

Observe that this does not hold for the LP-relaxations (4) where generically \( \theta_d < f^* \) for every \( d \in \mathbb{N} \); see e.g. [7], [9].

III. A parametrized hierarchy of semidefinite relaxations

Problem (8) is convex but in general the objective function \( G_d \) is non-differentiable.

Moreover, another difficulty is the computation of \( G_d(\lambda) \) for each \( \lambda \geq 0 \) since \( G_d(\lambda) \) is the global optimum of the possibly non-convex function \( (x, \lambda) \mapsto L_d(x, \lambda) \) defined in (7). So one strategy is to replace (8) by a simpler convex problem (while preserving the convergence property) as follows.

A. Interpreting the LP-relaxations

Observe that the LP-relaxations (4) can be written

\[
\theta_d = \max_{\lambda \geq 0} \{ \ t : L_d(x, \lambda) - t = 0, \ \forall x \in \mathbb{R}^n \},
\]

where \( L_d \) has been defined in (7).

And so the LP-relaxations (4) can be interpreted as simplifying (8) by restricting the nonnegative orthant \( \{ \lambda : \lambda \geq 0 \} \) to its subset of \( \lambda \)'s that make the polynomial \( x \mapsto L(x, \lambda) - t \) constant and equal to zero, instead of being only nonnegative. This subset being a polyhedron, solving (10) reduces to solving a linear program. At first glance, such an a priori simple and naive brute force simplification might seem unreasonable (to say the least). But of course the LP-relaxations (4) where not defined this way. Initially, the Sherali-Adams' RLT hierarchy [13] was introduced for 0/1 programs and finite convergence was proved by using ad hoc arguments. But in fact, the rationale behind convergence of the more general LP-relaxations (4) is the Krivine-Stengle positivity certificate [9, Theorem 2.23].

However, even though this brute force simplification still preserves the convergence \( \theta_d \to f^* \) thanks to [9, Theorem 2.23], we have already mentioned that it also implies serious theoretical (and practical) drawbacks for the resulting LP-relaxations (like slow asymptotic convergence for convex problems and numerical ill-conditioning).

B. A parametrized hierarchy of semidefinite relaxations

However, inspired by this interpretation we propose a systematic way to improve the LP-relaxations (4) along the same lines but by doing a much less brutal simplification of (8). Indeed, one may now impose on the same nonnegative polynomial \( x \mapsto L(x, \lambda) - t \) to be a sum of squares (SOS) polynomial \( \sigma \) of degree at most \( 2k \) (instead of being constant and equal to zero as in (10)), and solve the resulting hierarchy of optimization problems:

\[
g_k^d = \max_{\lambda, t, \sigma} \{ \ t : \ L_d(x, \lambda) - t = \sigma, \ \forall x \in \mathbb{R}^n, \ \lambda \geq 0, \ \sigma \in \Sigma[x]_k \}
\]

with \( d = 1, 2, \ldots, \) and parametrized by \( k \), fixed. (Recall that \( \Sigma[x]_k \) denotes the set of SOS polynomials of degree at most \( 2k \).) To see that (11) is a semidefinite program, write

\[
x \mapsto L_d(x, \lambda) - t := \sum_{\beta \in \mathbb{N}^n_{2k}} L_\beta(\lambda, t) x^\beta,
\]

where \( s = d \max_{j} [\deg g_j] \) and \( L_\beta(\lambda, t) \) is linear in \( (\lambda, t) \) for each \( \beta \in \mathbb{N}^n_{2k} \).

Next, for \( k \in \mathbb{N} \) such that \( 2k \leq s \), let \( v_k(x) \) be the vector of the monomial basis \( (x^\beta, \beta \in \mathbb{N}^n_{2k}) \), of \( \mathbb{R}[x]_s \), and write

\[
v_k(x) v_k(x)^T = \sum_{\beta \in \mathbb{N}^n_{2k}} x^\beta B_\beta,
\]

for some appropriate real symmetric matrices \( (B_\beta), \beta \in \mathbb{N}^n_{2k} \). Then problem (11) is the semidefinite program:

\[
g_k^d = \max_{\lambda, t, Q} \{ \ t : \ L_\beta(\lambda, t) = (B_\beta, Q), \ \forall \beta \in \mathbb{N}^n_{2k}, \ L_\beta(\lambda, t) = 0, \ \forall \beta \in \mathbb{N}^n, |\beta| > 2k, \ \lambda \geq 0; \ Q = Q^T \geq 0, \}
\]

where \( Q \) is a \( (n+k) \times (n+k) \) real symmetric matrix.

Of course \( g_k^d \geq g_d \) for all \( d \) because with \( \sigma = 0 \) one retrieves (4). Moreover in the semidefinite program (12), the semidefinite constraint \( Q \succeq 0 \) is concerned with a real symmetric \( (n+k) \times (n+k) \) matrix, independently of the rank \( d \) in the hierarchy. For instance if \( k = 1 \) then \( \sigma \) is a quadratic SOS and \( Q \) has size \( (n+1) \times (n+1) \). In other words, even if the number of variables \( \lambda = (\lambda_{i,0}) \) increases fast with \( d \), the LMI constraint \( Q \succeq 0 \) has fixed size, in contrast with the semidefinite relaxations (3) where the size of the LMIs increases with \( d \). And it is a well-known fact that crucial for solving semidefinite program is the size of the LMIs involved rather than the number of variables.

C. Sherali-Adams’ RLT for 0/1 programs

Consider 0/1 programs with \( f \in \mathbb{R}[x] \), and feasible set \( K = \{ x : Ax \leq b \} \cap \{ 0, 1 \}^n \), for some real matrix \( A \in \mathbb{R}^{m \times n} \) and some vector \( b \in \mathbb{R}^m \). The Sherali-Adams’s RLT hierarchy [13] belongs to the family of LP-relaxations (4) but with a more specific form since \( K \subset \{ 0, 1 \}^n \).

Notice that the family \( \{ 1, x_1, (1 - x_1), \ldots, x_n, (1 - x_n) \} \) generates the algebra \( \mathbb{R}[x] \). Let \( g_\ell(x) = (b - Ax)_\ell \), \( \ell = 1, \ldots, m \), and \( g_0(x) = 1 \). Following the definition of the Sherali-Adams’ RLT in [13], the resulting linear program at step \( d \) in the hierarchy reads:

\[
\theta_d = \max_{\lambda \geq 0, t, h} \{ \ t : f(x) - t = \sum_{i=1}^n h_i(x) (x_i (1 - x_i)) \}
\]

\[
+ \sum_{\ell=0}^m \sum_{I,J = \emptyset \cup [1, \ldots, n]} \lambda_{I,J}^d g_\ell(x) \prod_{i \in I} (1 - x_j); \ h_i \in \mathbb{R}[x]_{d-1} \}
\]

where \( \lambda \) is the nonnegative vector \( (\lambda_{I,J}^d) \). (If there are linear equality constraints \( g_\ell(x) = 0 \) the corresponding variables
\( \lambda_{ij} \) are not required to be nonnegative. So all products between the \( g \)'s are ignored (see the paragraph before Lemma 1 in [13, p. 414]) even though they might help tighten the relaxations. In the literature the dual LP of (13) is described rather than (13) itself.

In this context, the problem \( \bar{\mathbf{P}}_{d} \) equivalent to \( \mathbf{P} \) and defined in (9) by adding redundant constraints formed with products of original ones, reads:

\[
\min \{ f(x) : \ x^\alpha x_j (1-x_j) = 0, \ j \leq n; \ \alpha \in \mathbb{N}_{0}^n; \\
g(\mathbf{x}) \prod_{i \in I} x_i \prod_{j \in J} (1-x_j) \geq 0, \ 0 \leq \ell \leq m, \\
I, J \subset \{1, \ldots, n\}; \ I \cap J = \emptyset; |I \cup J| \leq d \}.
\]

Hence the 0/1 analogue of (11) reads

\[
q_{k}^{d} = \max_{\lambda_{0,1,h}} \left\{ t : f(x) - t = \sigma(x) + \sum_{i=1}^{n} h_i(x) x_i (1-x_i) \right\}
\]

\[
+ \sum_{\ell=0}^{m} \lambda_{ij} g(\mathbf{x}) \prod_{i \in I} x_i \prod_{j \in J} (1-x_j); \\
\sigma \in \Sigma[x]; \ h_i \in \mathbb{R}[x]_{d-1} \quad i = 1, \ldots, n .
\]

For 0/1 programs with linear or quadratic objective function, and for every \( k \geq 1 \), the first semidefinite relaxation (14), i.e., with \( d = 1 \), is at least as powerful as that of the standard hierarchy of semidefinite relaxations (3). Indeed (14) contains products \( g(\mathbf{x})x_j \) or \( g(\mathbf{x})(1-x_k) \), for all \( (\ell, j, k) \), which do not to appear in (3) with \( d = 1 \). And so in particular, the first such relaxation for MAXCUT has the celebrated Goemans-Williamson’s performance guarantee while the standard LP-relaxations (4) do not. On the other hand, for 0/1 problems and for the parameter value \( k = 1 \), the hierarchy (14) is what is called the Sherali-Adams + SDP hierarchy (basic SDP-relaxation + RLT hierarchy) in e.g. Benabas and Magen [3] and Benabas et al. [2]; and in [3], [2] the authors show that any (constant) level \( d \) of this hierarchy, viewed as a strengthening of the basic SDP-relaxation, does not make the integrality gap decrease.

IV. COMPARING WITH STANDARD LP-RELAXATIONS

What do we gain by going from the LP hierarchy (4) to the semidefinite hierarchy (12) parametrized by \( k \)?

Recall that a highly desirable property for a general purpose method aiming at solving NP-hard optimization problems, is to behave efficiently when applied to a class of problems considered relatively easy to solve. Otherwise one might raise reasonable doubts on its efficiency for more difficult problems not only in a worst-case sense but also in average. And convex problems \( \mathbf{P} \) as in (1)-(2), i.e., when \( f, g \) are convex, form the most natural class of problems which are considered easy to solve by some standard methods of Non Linear Programming.

**Theorem 4:** With \( \mathbf{P} \) as in (1)-(2) let \( f, g \) be convex, \( j = 1, \ldots, m \), let Slater’s condition hold and let \( f^* = -\infty \). Then:

(a) If \( \max[\deg f, \deg g] \leq 2 \) then \( q_{1}^{d} = f^* \), i.e., the first relaxation of the hierarchy (12) parametrized by \( k = 1 \), is exact.

(b) If \( \max[\deg f, \deg g] \leq 2k \), then \( f^* \) is an SOS-convex, then \( q_{k}^{d} = f^* \), i.e., the first relaxation of the hierarchy (12) parametrized by \( k \), is exact.

**Proof:** Under the assumptions of Theorem 4, \( \mathbf{P} \) has a minimizer \( x^* \in K \) and the Karush-Kuhn-Tucker optimality conditions hold at \( (x^*, \lambda^*) \in K \times \mathbb{R}^{n} \) for some \( \lambda^* \in \mathbb{R}^{m} \). And so if \( k = 1 \), the Lagrangian polynomial

\[
L_{1}(\cdot, \lambda^*) - f^*
\]

is a nonnegative quadratic polynomial and so an SOS \( \sigma^* \in \Sigma[x]_{1} \). Therefore as \( q_{2}^{d} \leq f^* \) for all \( d \), the triplet \( (\lambda^*, f^*, \sigma^*) \) is an optimal solution of (11) with \( k = d = 1 \), which proves (a). Next, if \( k > 1 \) and \( f, g \) are all SOS-convex then so is the Lagrangian polynomial

\[
L_{k}(\cdot, \lambda^*) - f^*
\]

in addition, as

\[
\nabla_{\lambda}L_{k}(\cdot, \lambda^*) = 0 \quad \text{and} \quad L_{k}(\cdot, \lambda^*) - f^* = 0,
\]

the polynomial

\[
L_{k}(\cdot, \lambda^*) - f^* \text{ is SOS}; \text{ see e.g. [8]. Hence } L_{k}(\cdot, \lambda^*) - f^* = \sigma^*
\]

for some \( \sigma^* \in \Sigma[x]_{k} \), and again, the triplet \( (\lambda^*, f^*, \sigma^*) \) is an optimal solution of (11) with \( d = 1 \), which proves (b).

Hence one recovers a nice and highly desirable property for the resulting hierarchy, but at the price of solving now a hierarchy of semidefinite programs; however the increase in complexity is controlled by the parameter \( k \) since the size of the LMI in the semidefinite program (12) is \( O(n^k) \), independently of the rank \( d \) in the hierarchy.

**REFERENCES**


