

LP formulations for mixed-integer polynomial optimization problems

Daniel Bienstock and Gonzalo Muñoz, Columbia University, December 2014

Abstract

We present a class of linear programming approximations for constrained optimization problems. In the case of mixed-integer polynomial optimization problems, if the intersection graph of the constraints has bounded tree-width our construction yields a class of linear size formulations that attain any desired tolerance. As a result, we obtain an approximation scheme for the “AC-OPF” problem on graphs with bounded tree-width. We also describe a more general construction for pure binary optimization problems where individual constraints are available through a membership oracle; if the intersection graph for the constraints has bounded tree-width our construction is of linear size and exact. This improves on a number of results in the literature, both from the perspective of formulation size and generality.

1 Introduction

A fundamental paradigm in the solution of integer programming and combinatorial optimization problems is the use of extended, or lifted, formulations, which rely on the binary nature of the variables and on the structure of the constraints to generate higher-dimensional convex relaxations with provably strong attributes.

In this paper we consider both pure binary and mixed-integer polynomial optimization problems. We develop a reformulation operator which relies on the combinatorial structure of the constraints to produce linear programming approximations which attain provable bounds. A graph-theoretic parameter, the tree-width, is used to parameterize the computational effort involved in the approximation. Although our results focus on integer and mixed-integer problems, we extend previous work on exploiting structured sparsity, by Laurent [35], Lasserre [32], Waki, Kim, Kojima and Muramatsu [52] in the continuous polynomial optimization setting, as well as similar work concerning pure integer programs by Wainwright and Jordan [51] and Bienstock and Özbay [11].

Our first result concerns a broad class of linear objective optimization problems, with binary variables, that we will term *general binary* optimization problems, or GB for short.

Problem GB

- (i) Variables are indexed by a set \mathcal{V} . Write $n \doteq |\mathcal{V}|$.
- (ii) There are m constraints. For $1 \leq i \leq m$, constraint i is characterized by a subset $K[i] \subseteq \mathcal{V}$ and a set $S^i \subseteq \{0, 1\}^{K[i]}$. Set S^i is implicitly given by a *membership oracle*, that is to say a mechanism that upon input $y \in \{0, 1\}^{K[i]}$, truthfully reports whether $y \in S^i$.
- (iii) The problem is to minimize a linear function $c^T x$, over $x \in \{0, 1\}^{\mathcal{V}}$, and subject to the constraint that for $1 \leq i \leq m$ the projection of x to $\mathbb{R}^{K[i]}$ is contained in S^i .

Any linear-objective, binary optimization problem whose constraints are explicitly stated can be recast in the form GB; e.g., each set S^i could be described by a system of algebraic equations in the variables x_j for $j \in K[i]$. However the membership oracle framework extends beyond such special cases.

Theorem 1 *Let \mathcal{P} be an instance of problem GB. Given a tree-decomposition of width ω of the intersection graph for the constraints, there is an exact linear programming formulation for problem \mathcal{P} with $O(2^\omega n)$ variables and constraints, with $\{0, 1, -1\}$ -valued constraint coefficients.*

Note that the size of the formulation in this theorem is independent of the number constraints. To explain this statement we need to define two concepts: *intersection graph* and *tree-decomposition*. The intersection graph of a system of constraints is a fundamental concept introduced by Fulkerson and Gross [21] and extended here:

Definition 2 *The intersection graph of a system of constraints is the undirected graph which has a vertex for each variable and an edge for each pair of variables that appear explicitly in any common constraint. In the case of a problem of type GB, a variable x_j “appears explicitly” in a constraint i if $j \in K[i]$. For brevity, we will sometimes view the x_j as the vertices of the graph, rather than the indices j .*

Example 3 *Given the system of five constraints*

$$\begin{aligned} (1) \quad & 3x_1^2 - \log_2(1 + x_2) \geq 0, & (2) \quad & -2x_2^2 + (1 + x_3)^3 \geq 0, \\ (3) \quad & x_3^3 - \sqrt{2 + x_4^2} < 0, & (4) \quad & x_4 - x_1 = 0, & (5) \quad & 3x_3^2 + 4x_5 \text{ is a prime integer} \end{aligned}$$

The intersection graph has vertices $\{1, 2, 3, 4, 5\}$ and edges $\{1, 2\}$, $\{2, 3\}$, $\{3, 4\}$, $\{4, 1\}$ and $\{3, 5\}$. Note that e.g. $K[5] = \{3, 5\}$ and S^5 consists of the tuples $(y_3 = 1, y_5 = 0)$ and $(y_3 = 1, y_5 = 1)$.

Next we review the concept of *tree-decomposition*.

Definition 4 *Let G be an undirected graph. A tree-decomposition [46], [47] of G is a pair (T, Q) where T is a tree and $Q = \{Q_t : t \in V(T)\}$ is a family of subsets of $V(G)$ such that*

- (i) *For all $v \in V(G)$, the set $\{t \in V(T) : v \in Q_t\}$ forms a subtree T_v of T , and*
- (ii) *For each $\{u, v\} \in E(G)$ there is a $t \in V(T)$ such that $\{u, v\} \subseteq Q_t$, i.e. $t \in T_u \cap T_v$.*

The width of the decomposition is $\max\{|Q_t| : t \in V(T)\} - 1$. The tree-width of G is the minimum width of a tree-decomposition of G . See Example 5.

Example 5 (Tree-decomposition) *Consider the intersection graph G arising in Example 3. See Figure 1(a). A tree-decomposition with tree T is shown in Figure 1(b)-(c).*

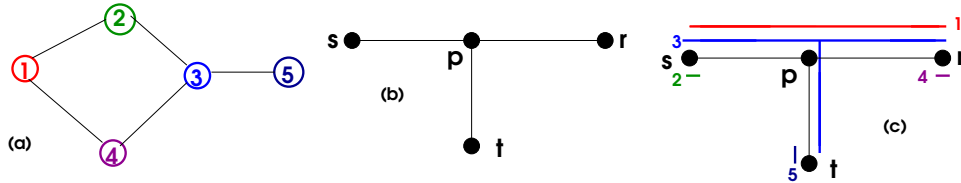


Figure 1: A tree-decomposition. (a) Graph G . (b) Tree T . (c) Tree T with subtrees T_v .

We next turn to mixed-integer polynomial optimization problems. We will prove:

Theorem 6 *Let \mathcal{P} be a linear objective mixed-integer optimization problem over n (bounded) variables and where every constraint is a polynomial inequality of maximum degree $\leq \pi$. Given a tree-decomposition of width ω of the intersection graph for the constraints, and $0 < \epsilon < 1$, there is a linear programming formulation of size $O(\epsilon^{-\omega-1} \pi^\omega n \log \epsilon^{-1})$ that solves \mathcal{P} within feasibility and optimality tolerance ϵ .*

Below we will provide an extended statement for this result, as well as a precise definition of ‘tolerance’. However, the statement in Theorem 6 is indicative of the fact that as $\epsilon \rightarrow 0$ we converge to the optimal solution, and the computational workload grows proportional to $O(\epsilon^{-\omega-1} \log \epsilon^{-1})$. Moreover, we will prove

Theorem 7 *Unless $P=NP$, no polynomial time algorithm for mixed-integer polynomial optimization exists that improves on the dependence on ϵ given by Theorem 6.*

Our next set of results concern *graphical* mixed-integer polynomial optimization problems. A problem in this family has a linear objective, bounded variables, and is polynomially constrained, and there is an underlying graph, G , such that each variable is associated with a vertex and for each constraint there is a vertex v so that all variables explicitly appearing the constraint are associated with v or a neighbor of v .

Example 8 . Consider the polynomial optimization problem with constraints

$$x_1^2 + x_2^2 + 2x_3^2 \leq 1, \quad x_1^2 - x_3^2 \geq 0, \quad x_3x_4 + x_5^2 - x_6 \geq 1/2, \quad x_1(1 - x_2) - x_4 \leq 0$$

$$0 \leq x_1 \leq 1, \quad 0 \leq x_2 \leq 1, \quad 0 \leq x_3 \leq 1, \quad 0 \leq x_5 \leq 1, \quad 0 \leq x_6 \leq 1, \quad x_4 \in \{0, 1\}.$$

Here, variables x_1 and x_2 are associated with vertex a of the graph in Figure 2, and variables x_3 , x_4 , x_5 and x_6 are associated with vertices b, c, d and e , respectively. The first and second constraints are associated with the neighborhood of vertex a , the third is associated with vertex b and the fourth with vertex c .

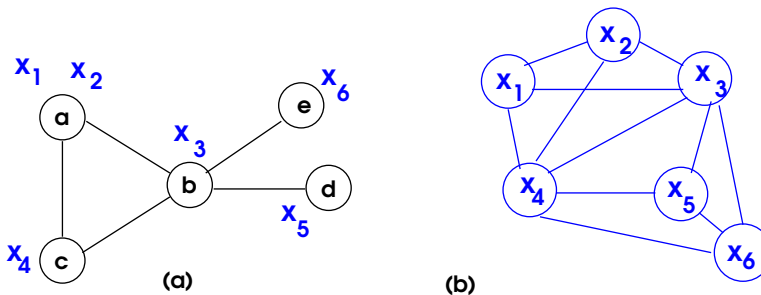


Figure 2: (Example 8) (a) Graphical polynomial optimization problem. (b) The intersection graph.

This problem family includes, as a special case, the well-known “AC-OPF” problem and mixed-integer extensions. We will prove:

Theorem 9 Let \mathcal{P} be a graphical mixed-integer polynomial optimization problem over a graph G , with n variables and where every constraint is a polynomial inequality of maximum degree $\leq \pi$. Suppose that there are at most K variables associated with any vertex of G . Given a tree-decomposition of width ω of the graph G , and $0 < \epsilon < 1$, there is a linear programming formulation of size $O(\epsilon^{-\omega-1} \pi^\omega K n \log \epsilon^{-1})$ that solves \mathcal{P} within feasibility and optimality tolerance ϵ .

We stress that in Theorem 9 the tree-decomposition is of the underlying graph G , and *not* of the intersection graph of the problem, which in general will have larger tree-width. See Figure 2 where G has tree-width 2 whereas the intersection graph has tree-width 3 (due to the clique arising from the third constraint). Thus a direct application of Theorem 6 does not yield Theorem 9. As a consequence of Theorem 9, we obtain a polynomial-size, ϵ -tolerant formulation for the AC-OPF problem (where $\pi = 2$ and $K = O(1)$) and mixed-integer extensions, when the tree-width of the underlying network is bounded by a constant.

1.0.1 Organization of the paper

In Section 2 we will present a detailed analysis of the pure binary problems addressed by Theorem 1 and a proof of this result. Mixed-integer polynomial optimization problems and a proof of Theorem 6 are covered in section 3. Finally, graphical mixed-integer polynomial optimization problems and Theorem 9 are addressed in Section 4.

2 Pure binary problems

Here we consider Theorem 1 of the Introduction. We will provide additional background, survey previous results, and state and prove an expanded version of Theorem 1. First we begin with some examples for problem GB.

Example 10 (*Linear binary integer programming*). Let A be an $m \times n$ matrix, and consider a problem $\min\{c^T x : Ax \geq b, x \in \{0, 1\}^n\}$. To view this problem as a special case of GB, we set for $1 \leq i \leq m$, $K[i] = \{1 \leq j \leq n : a_{ij} \neq 0\}$ and $S[i] = \{x \in \{0, 1\}^{K[i]} : \sum_{j \in K[i]} a_{ij} x_j \geq b_i\}$. In this special case, problem GB can be addressed by a variety of methods. Of particular interest in this paper are the reformulation or lifting methods of Lovász and Schrijver, and Sherali and Adams.

Next we consider a more complex example, chosen to highlight the general nature of the problem.

Example 11 Let d, n, p be positive integers. Consider a constrained semidefinite program over binary variables of the form

$$\min \sum_{k=1}^n \sum_{i=1}^d \sum_{j=1}^d c_{kij} X_{i,j}^k \quad (2a)$$

$$\text{subject to: } M^k \bullet X^k = b_k, \quad 1 \leq k \leq n, \quad (2b)$$

$$X^k \in S_d^+, \quad 1 \leq k \leq n, \quad (2c)$$

$$\sum_{i,j} X_{i,j}^k \equiv 0 \pmod{p}, \quad 1 \leq k \leq n, \quad (2d)$$

$$X_{i,1}^k = X_{i,d}^{k-1}, \quad 1 \leq i \leq d, \quad 2 \leq k \leq n, \quad (2e)$$

$$X_{i,j}^k \in \{0, 1\}, \quad \forall i, j, k. \quad (2f)$$

Here S_d^+ is the set of $d \times d$ positive-semidefinite matrices, M_1, \dots, M_n are symmetric $d \times d$ matrices, and b and c are vectors. Constraint (2e) states that the first column of matrix X^k is identical to the last column of matrix X^{k-1} .

We obtain an instance of problem GB with $m = 2n - 1$, as follows. First, for each $1 \leq k \leq n$ we let $K[k]$ be the set of triples (i, j, k) with $1 \leq i, j \leq n$, and S^k to be the set of binary values $X_{i,j}^k$ that satisfy (2b)-(2d). Next, for each $2 \leq k \leq n$ we let $K[n+k-1]$ be the set of all triples $(i, 1, k-1)$ and all triples (i, d, k) and S^{n+k-1} to be the set of binary values (indexed by $K[n+k-1]$) such that (2e) holds.

In the case of this example, a direct application of standard integer programming methods appears difficult. Moreover, we stress that the sets S^i in problem GB are completely generic and that the membership oracle perspective can prove useful (see the discussion in Section 2.0.2, below).

We can now state the main result we will prove in this section, which implies Theorem 1. Recall that as per Definition 2, given a problem instance \mathcal{I} of GB, the intersection graph for \mathcal{I} has a vertex for each $j \in \mathcal{V}$, and an edge $\{j, k\}$ whenever there exists $1 \leq i \leq m$ such that $\{j, k\} \subseteq K[i]$, that is to say, j and k appear in a common constraint (ii) in problem GB.

Theorem 12 Given an instance \mathcal{I} of GB, let (T, Q) be a tree-decomposition of the intersection graph of \mathcal{I} . Then there is an exact (continuous) linear programming reformulation $LP(\mathcal{I})$ for instance \mathcal{I} with $O(\sum_t 2^{|\mathcal{Q}_t|})$ variables and constraints, the same objective vector c and constraints with $\{0, 1, -1\}$ -valued coefficients.

As a corollary, if the width of (T, Q) is ω , the formulation has $O(2^\omega n)$ variables and constraints. Hence for each fixed ω the formulation has linear size.

The ‘‘corollary’’ statement follows because if an n -vertex graph has a tree-decomposition of width d , say, then it has one with the same width and where in addition the tree has at most n vertices¹. We will prove Theorem 12 below, however first we discuss implications of this result.

¹This is implied, for example, by the fact that in a chordal graph there exists a vertex whose neighbors induce a clique.

Example 13 (Example 11, continued). Here we will set

$$\mathcal{V} = \{(i, j, k) : 1 \leq k \leq n \text{ and } 1 \leq i, j \leq d\}.$$

The intersection graph of the problem will have

- (a) the edge $\{(i, j, k), (i', j', k)\}$ for all $1 \leq k \leq n$ and $1 \leq i, j, i', j' \leq d$, arising from constraints (2b)-(2d)
- (b) the edge $\{(i, 1, k), (i, d, k-1)\}$ for each $1 \leq k < n$ and $1 \leq i \leq d$, arising from constraints (2e).

A tree-decomposition (T, Q) of the intersection graph, of width $O(d^2)$, is obtained as follows. Here, T is path with vertices $v_1, u_2, v_2, u_3, \dots, v_{n-1}, u_n, v_n$. For $1 \leq k \leq n$ we set $Q_{v_k} = \{(i, j, k) : 1 \leq i, j \leq d\}$ and for $2 \leq k \leq n$ we set $Q_{u_k} = \{(i, 1, k), (i, d, k-1) : 1 \leq i \leq d\}$. Sets Q_{v_k} account for all edges of type [(a)], whereas the sets Q_{u_k} cover all edges of type [(b)]. Thus Theorem 12 states that there is an LP formulation for problem (2) with $O(2^{d^2} d^2 n)$ variables and constraints.

The methodology used to obtain Theorem 12 is best seen as an example of the use of extended, or “lifted” formulations for 0/1 integer programs to obtain provable guarantees. The classical examples in this vein are the reformulation-linearization technique of Sherali and Adams [48], the cones of matrices method of Lovász and Schrijver [39], the lift-and-project method of Balas, Ceria and Cornuéjols [4], and the moment relaxation methodology of Lasserre [31]. Laurent [34] presents a unifying analysis; another comparison is provided by Au and Tunçel [2].

In addition to these ‘generic’ operators, extended formulations are often found in combinatorial optimization settings, with the reformulation exploiting the specific nature of a problem. See Cornuéjols, Conforti, Zambelli [18]. An analysis of the impact of semidefiniteness is presented by Goemans and L. Tunçel [24]. More broadly, the general theory of disjunctive programming underlies this family of reformulation methods, see Balas [3]. Our method is most closely related to the subset algebra method developed in [12] and the PhD thesis [54], and to the use of the Sherali-Adams reformulation in the context of packing integer programs in [11].

2.0.2 Reduction to the linear case

Consider an instance \mathcal{I} of GB. An apparently simpler alternative to the general approach we follow would be to construct, for $1 \leq i \leq m$, the polyhedron

$$P_i \doteq \text{conv} \left\{ x \in \{0, 1\}^{K[i]} : x \in S^i \right\} \subseteq \mathbb{R}^{K[i]}.$$

Thus we can write P_i as the projection onto $\mathbb{R}^{K[i]}$ of a polyhedron $\{x \in [0, 1]^V : A^i x \geq b^i\}$ where each row of A^i has zero entries on any column not in $K[i]$. Then instance \mathcal{I} can be restated as the equivalent linear integer program

$$\min c^T x \tag{3a}$$

$$\text{subject to: } A^i x \geq b^i, \quad 1 \leq i \leq m \tag{3b}$$

$$x \in \{0, 1\}^V. \tag{3c}$$

Since GB includes linear integer programs as a special case, we can apply Theorem 12 directly to formulation (3). Using this approach, the intersection graph will be the standard intersection graph arising from the constraint matrix in (3b), i.e. the graph with vertex-set \mathcal{V} and an edge $\{j, k\}$ whenever j and k appear with nonzero coefficients in a common row of the matrix. By construction this graph is a subgraph of the intersection graph of instance \mathcal{I} and thus its tree-width is at most ω . Thus, applying Theorem 12 directly to (3) rather than to GB would apparently yield a smaller formulation, and one may question the utility of relying on the general form GB.

However, this analysis ignores the size of formulation (3). For $d \geq 1$ large enough there exist examples of 0/1-polytopes in \mathbb{R}^d with at least

$$\left(\frac{d}{\log d} \right)^{d/4}$$

facets (up to constants). See [5], [22], [30]. Using this observation, one can construct examples of problem GB where each of the matrices A^i has more than $\omega^{\omega/4}$ inequalities. This dependence on ω is strictly larger than that in Theorem 12.

Example 14 Choose $d \geq 2$ large enough so that there is a 0/1-polyhedron $P \subseteq \mathbb{R}^d$ with more than $(cd/\log d)^{d/4}$ facets for some c . Let P be given by the system $Ax \geq b$, where A is $M \times d$ ($M \geq (cd/\log d)^{d/4}$). Choose $N \geq 1$, and consider the system of inequalities over binary variables x_j^i , for $1 \leq i \leq N$ and $1 \leq j \leq d$:

$$Ax^i \geq b, \quad 1 \leq i \leq N, \quad (4a)$$

$$x_j^1 = x_j^i \quad 2 \leq i \leq N, \quad 1 \leq j \leq \lfloor d/2 \rfloor. \quad (4b)$$

$$x_j^i \text{ binary for all } i \text{ and } j. \quad (4c)$$

Constraint (4a) indicates that this system includes N copies of polyhedron P , with each copy described using a different coordinate system. Constraint (4b) states that the first $\lfloor d/2 \rfloor$ coordinates take equal value across all such systems.

Any linear program over (4) can be viewed as an example of problem PB with $m = 2N - 1$; for $1 \leq i \leq N$, $K[i]$ is used to represent the d variables x_j^i ($1 \leq j \leq d$) and S^i is a copy of the set of binary points contained in P (i.e. the extreme points of P).

The intersection graph of this instance of PB will be the union of N cliques (one for each set of variables x^i) plus the set of edges $\{x_1^1, x_1^i\}$ for $2 \leq i \leq N$. A tree-decomposition (T, Q) of this graph, of width $d - 1$, is as follows: T has vertices $u(0)$, as well as $u(i)$ and $v(i)$, for $1 \leq i \leq N$. Further, $Q_{u(0)} = \{x_j^1 : 1 \leq j \leq \lfloor d/2 \rfloor\}$; and for $1 \leq i \leq N$ $Q_{u(i)} = Q_{u(0)} \cup \{x_j^i : 1 \leq j \leq \lfloor d/2 \rfloor\}$ and $Q_{v(i)} = \{x_j^i, 1 \leq j \leq d\}$. Thus, $\omega = d$ and Theorem 12 states that any linear objective problem over constraints (4) can be solved as a continuous LP with $O(2^d d N)$ variables and constraints. In contrast, system (4) has more than $(cd/\log d)^{d/4} N$ constraints.

In particular, formulation (3) may be exponentially larger than the linear program $LP(\mathcal{I})$ stated in Theorem 12.

2.0.3 Comparison with the Sherali-Adams approach

In the case that problem GB is a linear integer program (or was converted into one by means of the linearization above) it could be addressed using the Sherali-Adams (or Lasserre, or variants of the Lovász-Schrijver) reformulation operator, at level $\omega + 1$. It can be shown that this will also yield a polynomial-size LP reformulation for fixed ω (e.g., see [11]). **However**, the number of variables and constraints in the resulting formulations will grow as

$$n^{\omega+1} \text{ and } m n^{\omega+1}, \text{ respectively,}$$

rather than the $2^\omega n$ dependence in Theorem 12, which relies on a different, linear programming reformulation operator described below in Section 2.1. In fact the $2^\omega n$ estimate can be conservative compared with the bound $\sum_t 2^{|Q_t|}$ given in the statement of Theorem 12. For example, if a single set Q_t has size $\omega + 1$ while all other sets have size $O(1)$, then the bound provided by the Theorem is $O(2^\omega + n)$.

2.0.4 Relationship of Theorem 12 to prior work

Previous work has produced results that are related to Theorem 12. Bienstock and Özbay (2004) consider *packing binary integer programs*, i.e. problems of the form

$$\max\{c^T x : Ax \leq b, x \in \{0, 1\}^n\} \quad (5)$$

where $A \geq 0$ and integral and b is integral. Given a valid inequality $\alpha x \geq \beta$ for the feasible region, its associated graph has a vertex j whenever $\alpha_j \neq 0$ and an edge $\{j, k\}$ whenever $a_{ij} \neq 0$ and $a_{ik} \neq 0$

for some row i . The following is proved in [11]:

Theorem A. [11] Given a problem (5), and $\omega \geq 1$, the Sherali-Adams reformulation at level- ω implies every valid inequality whose associated graph has tree-width $\leq \omega - 1$. If A is 0/1-valued, the same property holds when the associated graph has tree-width $\leq \omega$.

Corollary B. Given a graph G with tree-width $\leq \omega$, the Sherali-Adams reformulation of the vertex packing linear program $\{x \in [0, 1]^{V(G)} : x_u + x_v \leq 1 \forall \{u, v\} \in E(G)\}$, which has $O(\omega n^{\omega+2})$ variables and constraints, is exact.

Wainwright and Jordan (2004) consider binary polynomial optimization problems of the form

$$\min\{c^T x : x \in \{0, 1\}^n, g_i(x) \geq 0, 1 \leq i \leq M\}. \quad (6)$$

Here, for $1 \leq i \leq M$,

$$g_i(x) = \sum_{h \in s(i)} a_{ih} m_{ih}(x) \quad (7)$$

with $a_{ih} \neq 0$ and $m_{ih}(x)$ a monomial for each $h \in s(i)$. Given a problem (6) one can define the intersection graph as we did above, i.e. the graph with vertices $1 \leq j \leq n$ and an edge $\{j, j'\}$ if there is an index i ($1 \leq i \leq M$) such that x_j appears in at least one monomial $m_{ih}(x)$, and $x_{j'}$ also appears in at least one monomial $m_{ih}(x)$ (possibly different monomials).

Theorem C. [51] Consider an instance of problem (6) where the tree-width of the intersection graph is $\leq \omega$. Then the level- ω Sherali-Adams or Lasserre reformulation of (6) is exact, and as a consequence there is an LP formulation for (6) with $O(n^{\omega+2})$ variables and $O(n^{\omega+2}M)$ constraints.

Laurent (2010) provides a comprehensive survey of results on polynomial optimization and related topics. Section 8 of [35] builds on the work in Laurent (2001) [34], which provides a common framework for the Sherali-Adams, Lovász-Schrijver and Lasserre reformulation operators. This framework is used to show the following results. First, the vertex packing problem on a graph with n vertices and tree-width $\leq \omega$ has a formulation of size $O(2^\omega n)$; which is stronger than Corollary B. Second, a stronger result than Theorem C is obtained:

Theorem D.[35] Consider an instance of problem (6) where the tree-width of the intersection graph is $\leq \omega$. Then there is an LP formulation for problem (6) with $O(2^\omega n)$ variables and $O(2^\omega M)$ constraints.

A comparison between Theorems C and D, and Theorem 12 can be made as follows. Any polynomial optimization problem (6) can be reduced to an equivalent instance of problem GB by setting $m = M$, and using, for $1 \leq i \leq m$ each constraint $g_i(x) \geq 0$ to constitute a constraint (ii) in problem GB, with the set $K[i]$ defined as the set of variables that appear in $g_i(x)$ and S^i defined as the projection onto $K[i]$ of $\{x \in \mathbb{R}^n : g_i(x) \geq 0\}$.

If we apply the construction in Theorem 12 to this instance of problem GB, we will obtain an LP formulation for (6) with $O(2^\omega n)$ variables and constraints. This bound improves on the size of the formulations obtained by applying Theorems C and D, especially when M is large². In Section 2.0.2 we have already described a class of examples where that is the case. In the polynomial optimization setting we can make a stronger observation:

Remark. Let $d \geq 1$ be an integer. Then for each subset $S \subseteq \{0, 1\}^d$ there is a polynomial $p_S : \mathbb{R}^d \rightarrow \mathbb{R}$ with the property that $p_S(x) = 0$ iff $x \in S$.

²To be sure, the construction of the formulation in Theorem 12 will require $O(2^\omega M)$ queries to the membership oracle; however the focus in this discussion is on the size of the formulation.

This observation implies that for $d \geq 1$ there are at least 2^{2^d} polynomials that are distinct already on $\{0, 1\}^d$; and as a result for $\omega \geq 0$ one can construct instances of problem (6) where the intersection graph has tree-width ω , and yet

$$M > \frac{1}{2} n 2^{2^\omega},$$

with all constraints nontrivially distinct.

In summary, thus, Theorem 12 provides both a more general construction than that in previous work, and it also produces a smaller formulation. It is worth noting that the proofs of Theorems A, C, D and 12 all include a common ingredient. We will return to this point later.

It is worth pointing out that there is a vast literature on polynomial-time algorithms for combinatorial problems on graphs with small tree-width, through appropriately developed dynamic programming algorithms. See [16], [1], [6], [13]; also see [10]. From a broader perspective, stressing the connection with constraint programming, see [28]. This broader perspective is important given the generic setting of our Theorem 12. One of the earliest works we are aware of is [7] where the terminology “nonserial dynamic programming” was introduced to denote algorithms that take advantage of tree-decompositions of small width. The above algorithms for combinatorial problems on small tree-width graphs make use of this idea. For an earlier graph-theoretic perspective, see [27]. Finally, note that constant tree-width can be recognized in linear time [14].

A different result which nevertheless appears related is obtained by Cunningham and Geelen in [19]. They consider an optimization problem of the form $\min\{c^T x : Ax = b, x \in \mathcal{Z}_+^n\}$. They show that if either $A \geq 0$ or if bounds are imposed on the variables, the optimization problem can be solved in polynomial time if the matroid $M(A)$ has bounded *branchwidth* (a parameter related to tree-width in the case of graphs). It appears difficult to provide a direct comparison between this result and those cited above.

We will provide an additional review, focused on continuous optimization problems, in Section 3.0.2.

2.1 Proof of Theorem 12

Consider an instance \mathcal{I} of problem GB. Let $\Gamma = \Gamma[\mathcal{I}]$ be the corresponding intersection graph, and (T, Q) be a tree-decomposition of Γ of width ω . We begin with some general remarks.

Remark 15 *Suppose that (\bar{T}, \bar{Q}) is a tree-decomposition of a graph \bar{G} . Then for any clique K of \bar{G} there exists $t \in \mathcal{V}(\bar{T})$ with $K \subseteq \bar{Q}_t$.*

As a result, for $1 \leq i \leq m$ there exists $t \in V(T)$ with $K[i] \subseteq Q_t$.

Remark 16 *Without loss of generality, $|V(T)| \leq n$. To see this, note that the tree-decomposition (T, Q) gives rise to a chordal supergraph H of Γ . Since H is chordal, there exists a vertex u whose neighbors (in H) induce a clique. The claim follows by induction applied to the graph $H - u$, using Remark 15 and noting that a tree-decomposition of H is also a tree-decomposition of Γ .*

Definition 17 *Let $t \in V(T)$.*

- (a) *We say that $v \in \{0, 1\}^{Q_t}$ is **Q_t -feasible** if $\text{proj}(v, \mathbb{R}^{K[i]}) \in S^i$ for every $1 \leq i \leq m$ such that $K[i] \subseteq Q_t$ ($\text{proj}(x, A)$ is projection of x onto the subspace A).*
- (b) *We denote by \mathbf{F}_t the set $\{v \in \{0, 1\}^{Q_t} : v \text{ is } Q_t\text{-feasible}\}$.*
- (c) *We let $\mathbf{\Omega}_t$ denote the set of pairs (Y, N) with $Y \cap N = \emptyset$, $Y \cup N \subseteq Q_t$, and such that*

1. $|Y| = 1$ and $|N| = 0$, or
2. (Y, N) partition $Q_t \cap Q_{t'}$, for some $t' \in V(T)$ with $\{t, t'\} \in E(T)$.

Example 18 (Examples 3 and 5, continued). Refer to Figure 1. Consider vertex t of T . We have that $Q_t = \{3, 5\}$, and (5) is the only constraint i with $K[i] \subseteq Q_t$. As in Example 3 we then have that $v \in \{0, 1\}^{Q_t}$ is Q_t -feasible iff $v_3 = 1$. The set Ω_t consists of the pairs $(\{3\}, \emptyset)$, $(\{5\}, \emptyset)$, $(\{3\}, \{5\})$, $(\{5\}, \{3\})$, $(\{3, 5\}, \emptyset)$, $(\emptyset, \{3, 5\})$ and $(\emptyset, \{3\})$. The last pair arises because $Q_t \cap Q_p = \{3\}$.

We next construct the formulation $LP(\mathcal{I})$. The variables are as follows:

- A variable $X[Y, N]$, for each pair $(Y, N) \in 2^{\mathcal{V}} \times 2^{\mathcal{V}}$ with $(Y, N) \in \Omega_t$ for some $t \in V(T)$.
- A variable λ_v^t , for each $t \in V(T)$ and each vector $v \in F_t$.

The formulation is given by:

$$(LP-GB): \quad \min \sum_{j \in \mathcal{V}} c_j X[\{j\}, \emptyset] \quad (8a)$$

s.t. $\forall t \in V(T) :$

for all $(Y, N) \in \Omega_t$,

$$X[Y, N] = \sum \{ \lambda_v^t : v \in F_t, v_j = 1 \forall j \in Y, v_j = 0 \forall j \in N \} \quad (8b)$$

$$\sum_{v \in F_t} \lambda_v^t = 1, \quad \lambda^t \geq 0. \quad (8c)$$

We will show below that (a) LP-GB is a relaxation of GB and (b) the relaxation is exact and that the polyhedron defined by (8b)-(8c) is integral. We stress that the formulation (8) depends on the tree-decomposition (T, Q) and is thus not directly obtained from the formulation for problem GB.

Example 19 (Example 18, continued). Consider vertex t of the tree T in Figure 1. As discussed in Example 19, $v = (v_3, v_5) \in \{0, 1\}^{Q_t}$ is Q_t -feasible iff $v_3 = 1$; in other words $F_t = \{(v_3, v_5) \text{ binary} : v_3 = 1\}$. Thus, in the case that $(Y, N) = (\{3\}, \emptyset)$ constraint (8b) reads:

$$X[\{3\}, \emptyset] = \lambda_{(1,0)}^t + \lambda_{(1,1)}^t.$$

Likewise, $X[\{5\}, \emptyset] = \lambda_{(1,1)}^t$, $X[\{3\}, \{5\}] = \lambda_{(1,0)}^t$, $X[\{3, 5\}, \emptyset] = \lambda_{(1,1)}^t$. On the other hand

$$X[\emptyset, \{3\}] = X[\{5\}, \{3\}] = X[\emptyset, \{3, 5\}] = 0$$

because in each of these cases the sum in (8b) is empty. See Remark 20 (b) below. Finally, constraint (8c) reads

$$\lambda_{(1,0)}^t + \lambda_{(1,1)}^t = 1.$$

Remark 20

(a) A given pair (Y, N) may belong to Ω_t for more than one vertex $t \in V(T)$. Nevertheless, there is a single variable $X[Y, N]$. On the other hand, the variables λ^t are indexed by t .

(b) The sum on the right-hand side of constraint (8b) could be empty. This will be the case if for any $v \in \{0, 1\}^{Q_t}$ with $v_j = 1$ for all $j \in Y$ and $v_j = 0$ for all $j \in N$ there exists $1 \leq i \leq m$ with $K[i] \subseteq Q_t$ and yet $\text{proj}(v, \mathbb{R}^{K[i]}) \notin S^i$. Then (8b) states $X[Y, N] = 0$.

(c) For $v \in F_t$ define $Y = \{j \in Q_t : v_j = 1\}$ and $N = Q_t - Y$. Then (8b) states $X[Y, N] = \lambda_v^t$.

First we show that LP-GB is a relaxation for GB, in a strong sense.

Lemma 21 Let \tilde{x} be a feasible solution to \mathcal{I} .

(i) There is a feasible, 0/1-valued solution $(\tilde{X}, \tilde{\lambda})$ to (8) such that for each variable $X[Y, N]$ in (8) we have $\tilde{X}[Y, N] = \prod_{j \in Y} \tilde{x}_j \prod_{j \in N} (1 - \tilde{x}_j)$.

(ii) As a corollary $\sum_{j \in \mathcal{V}} c_j \tilde{X}[\{j\}, \emptyset] = c^T \tilde{x}$.

Proof. (i) For each variable $X[Y, N]$ in problem (8) we set $\tilde{X}[Y, N] = \prod_{j \in Y} \tilde{x}_j \prod_{j \in N} (1 - \tilde{x}_j)$. Further, for each $t \in V(T)$ let $\tilde{v}(t) \in \{0, 1\}^{Q_t}$ be the restriction of \tilde{x} to Q_t , i.e. $\tilde{v}(t)_j = \tilde{x}_j$ for each $j \in Q_t$. Since \tilde{x} is feasible for \mathcal{I} , $\tilde{v}(t) \in F_t$. Then we set $\tilde{\lambda}_{\tilde{v}(t)}^t = 1$ and $\tilde{\lambda}_v^t = 0$ for every vector $v \in F_t$ with $v \neq \tilde{v}(t)$. By construction for every $t \in V(T)$ and $(Y, N) \in \Omega_t$ we have $\tilde{X}[Y, N] = 1$ iff $\tilde{v}(t)_j = 1$ for all $j \in Y$ and $\tilde{v}(t)_j = 0$ for all $j \in N$; in other words (8b) is satisfied.

(ii) This follows from (i). ■

As a consequence of Lemma 21, Theorem 12 will follow if we can prove that the constraint matrix in (8) defines an integral polyhedron. This will be done in Lemma 25 given below. In what follows, we will view T as rooted, i.e. all edges are directed so that T contains a directed path from an arbitrarily chosen **leaf** vertex r (the root of T) to each other vertex. If (v, u) is an edge thus directed, then we say that v is the parent of u and u is a child of v .

Definition 22 A rooted subtree \tilde{T} is a subtree of T , such that there exists a vertex u of \tilde{T} so that \tilde{T} contains a directed path from u to every other vertex of \tilde{T} . We then say that \tilde{T} is rooted at u .

Definition 23 Let \tilde{T} be a rooted subtree of T . (a) We denote by $\Omega(\tilde{T})$ the set $\bigcup_{t \in \tilde{T}} \Omega_t$. (b) We denote by $\mathcal{V}(\tilde{T})$ the set $\{j \in \mathcal{V} : j \in Q_t \text{ for some } t \in \tilde{T}\}$.

Below we will prove the following result:

Theorem 24 Let $(\hat{X}, \hat{\lambda})$ be a feasible solution to the LP-GB problem (8). Then for every rooted subtree \tilde{T} there is a family of vectors

$$p^{k, \tilde{T}} \in \{0, 1\}^{\Omega(\tilde{T})},$$

vectors

$$x^{k, \tilde{T}} \in \{0, 1\}^{\mathcal{V}(\tilde{T})}$$

and reals

$$0 < \mu^{k, \tilde{T}} \leq 1,$$

($k = 1, 2, \dots, n(\tilde{T})$) satisfying the following properties:

- (a) For each $1 \leq k \leq n(\tilde{T})$ and each constraint $1 \leq i \leq m$ of problem \mathcal{I} , if $K[i] \subseteq Q_t$ for some $t \in \tilde{T}$, then $x^{k, \tilde{T}} \in S^i$.
- (b) For $1 \leq k \leq n(\tilde{T})$ and each pair $(Y, N) \in \Omega(\tilde{T})$,

$$p^{k, \tilde{T}}[Y, N] = \prod_{j \in Y} x_j^{k, \tilde{T}} \prod_{j \in N} (1 - x_j^{k, \tilde{T}}).$$

As a result, for each $1 \leq k \leq n(\tilde{T})$ and $j \in \mathcal{V}(\tilde{T})$, $x_j^{k, \tilde{T}} = p^{k, \tilde{T}}[\{j\}, \emptyset]$.

(c) $\sum_{k=1}^{n(\tilde{T})} \mu^{k, \tilde{T}} = 1.$

(d) For each $(Y, N) \in \Omega(\tilde{T})$,

$$\hat{X}[Y, N] = \sum_{k=1}^{n(\tilde{T})} \mu^{k, \tilde{T}} p^{k, \tilde{T}}[Y, N].$$

The family of vectors $p^{k, \tilde{T}}$ and reals $\mu^{k, \tilde{T}}$ will be called a **decomposition of $(\hat{X}, \hat{\lambda})$ over \tilde{T}** .

Pending a proof of Theorem 24, we can show that the polyhedron defined by the constraints in LP-GB is integral.

Lemma 25 The polyhedron defined by (8b)-(8c) is integral and problems GB and LP-GB have the same value.

Proof. Let $(\hat{X}, \hat{\lambda})$ be a feasible solution to LP-GB. We apply Theorem 24 with $\tilde{T} = T$ obtaining a family of vectors $p^k \in \{0, 1\}^{\Omega(r)}$, vectors $x^k \in \{0, 1\}^{\mathcal{V}}$ and reals μ^k , for $1 \leq k \leq n(r)$, satisfying conditions (a)-(d) of the theorem. By (a), each vector x^k is feasible for \mathcal{I} . By (d), the vector \hat{X} is a convex combination of the vectors p^k . This completes the proof, using Remark 20 (c) to handle the λ^t variables. ■

This result completes the proof of Theorem 12, pending Theorem 24 (tackled in the next Section) and the following basic observation:

Remark 26 *Suppose there is a tree-decomposition of a graph H with width w . Then there is a tree-decomposition (T, Q_t) of H , of width $\leq w$ and $|V(T)| \leq O(|V(H)|)$.*

2.1.1 Proof of Theorem 24

Assume we have a feasible solution $(\hat{X}, \hat{\lambda})$ to (8). The proof of Theorem 24 will be done by induction on the size of \tilde{T} . First we handle the base case.

Lemma 27 *If \tilde{T} consists of a single vertex u there is a decomposition of $(\hat{X}, \hat{\lambda})$ over \tilde{T} .*

Proof. We have that $\Omega(\tilde{T}) = \Omega_u$ (see Definition 23). By (8c) we have $\sum_{v \in F_u} \hat{\lambda}_v^u = 1$. Let $n(\tilde{T}) > 0$ be the number of elements $v \in F_u$ with $\hat{\lambda}_v^u > 0$ and denote these vectors by $\{w(1), \dots, w(n(\tilde{T}))\}$. Then, for $1 \leq k \leq n(\tilde{T})$ let $x^{k, \tilde{T}} = w(k)$ and $\mu^{k, \tilde{T}} = \hat{\lambda}_{w(k)}^u$. Finally, for $1 \leq k \leq n(\tilde{T})$ we define the vector $p^{k, \tilde{T}} \in \Omega_u$ by setting

$$p^{k, \tilde{T}}[Y, N] = \prod_{j \in Y} x_j^{k, \tilde{T}} \prod_{j \in N} (1 - x_j^{k, \tilde{T}})$$

for each pair $(Y, N) \in \Omega_u$. Now we will verify that conditions (a)-(d) of Theorem 24 hold. Clearly (a)-(c) hold by construction. To see that (d) holds, note that $(\hat{X}, \hat{\lambda})$ satisfies (8b), i.e.,

$$\begin{aligned} \hat{X}[Y, N] &= \sum \{ \mu^{k, \tilde{T}} : x_j^{k, \tilde{T}} = 1 \forall j \in Y, x_j^{k, \tilde{T}} = 0 \forall j \in N \} \\ &= \sum_{k=1}^{n(\tilde{T})} \mu^{k, \tilde{T}} p^{k, \tilde{T}}[Y, N] \end{aligned}$$

which is condition (e), as desired. ■

Next we prove the general inductive step needed to establish Theorem 24. This will be done by extending a technique from [11]. A similar technique was also used in [51] (where it is described as related to the ‘‘junction tree theorem’’ [37]) and also in [36] (where a related result due to Lasserre [33] is mentioned). From our perspective, the common techniques in [11], [51] and [36] are related to the concept of *nonserial dynamic programming* introduced in 1972 in [7].

Thus, consider a vertex u of T and a subtree \tilde{T} rooted at u with more than one vertex. Let v be a child of u . We will apply induction by partitioning \tilde{T} into two subtrees: the subtree L consisting of v and all its descendants in \tilde{T} , and the subtree $H = \tilde{T} - L$. Consider a decomposition of $(\hat{X}, \hat{\lambda})$ over L given by the vectors $p^{k, L} \in \{0, 1\}^{\Omega(L)}$ and the positive reals $\mu^{k, L}$ for $k = 1, 2, \dots, n(L)$, and a decomposition of $(\hat{X}, \hat{\lambda})$ over H given by the vectors $p^{k, H} \in \{0, 1\}^{\Omega(H)}$ and the positive reals $\mu^{k, H}$ for $k = 1, 2, \dots, n(H)$.

Denote by \mathcal{P} the set of partitions of $Q_u \cap Q_v$ into two sets. Then (see Remark ??) either $Q_v \subseteq Q_u$ or $Q_u \subseteq Q_v$. We assume the former (the two cases are symmetric). Thus, by Definition 17, for each partition $(\alpha, \beta) \in \mathcal{P}$ we have a variable $X[\alpha, \beta]$. Note that $\Omega(\tilde{T}) = \Omega(H) \cup \Omega(L)$. We construct a family of vectors and reals satisfying (a)-(d) Theorem 24 for \tilde{T} , as follows.

For each partition $(\alpha, \beta) \in \mathcal{P}$ such that $\hat{X}[\alpha, \beta] > 0$, and each pair i, h such that $1 \leq i \leq n(L)$, $1 \leq h \leq n(H)$, and $p^{h, H}[\alpha, \beta] = p^{i, L}[\alpha, \beta] = 1$ we create a vector $q_{ih}^{\alpha, \beta}$ and a real $\gamma_{ih}^{\alpha, \beta}$ using the rule:

For any vertex t in \tilde{T} and $(Y, N) \in \Omega_t$.

(r.1) if $t \in V(L)$ we set $q_{ih}^{\alpha,\beta}[Y, N] = p^{i,L}[Y, N]$.

(r.2) if $t \in V(H)$ we set $q_{ih}^{\alpha,\beta}[Y, N] = p^{h,H}[Y, N]$.

Further, we set

$$\gamma_{ih}^{\alpha,\beta} = \frac{\mu^{i,L} \mu^{h,H}}{\hat{X}[\alpha, \beta]}.$$

To argue that this construction is valid we note that since $\hat{X}[\alpha, \beta] > 0$, pairs of indices i, h as listed above must exist, by (d) of the inductive assumption applied to H and L . Furthermore, we have $\gamma_{ih}^{\alpha,\beta} > 0$.

Now we will prove that the q_{ih} and the γ_{ih} provide a decomposition of $(\hat{X}, \hat{\lambda})$ over \tilde{T} . Let i and h be given. Since the restriction of $p^{i,L}$ (and $p^{h,H}$) to L (resp., H) satisfy (a) and (b) of the inductive assumption, so will q_{ih} . Next, consider (d). Let $(Y, N) \in \Omega(\tilde{T})$, say $(Y, N) \in \Omega(H)$. We claim that

$$\sum_{\alpha,\beta,i,h} \gamma_{ih}^{\alpha,\beta} q_{ih}^{\alpha,\beta}[Y, N] = \sum_{(\alpha,\beta) \in \mathcal{P}: \hat{X}[\alpha,\beta] > 0} \sum_{i=1}^{n(L)} \sum_{h=1}^{n(H)} \frac{\mu^{i,L} \mu^{h,H}}{\hat{X}[\alpha, \beta]} p^{i,L}[\alpha, \beta] p^{h,H}[\alpha, \beta] p^{h,H}[Y, N]. \quad (9a)$$

This equation holds because in any nonzero term in either expression we must have $p^{i,L}[\alpha, \beta] = p^{h,H}[\alpha, \beta] = 1$ and since $(Y, N) \in \Omega(H)$ we also have that $q_{ih}^{\alpha,\beta}[Y, N] = p^{h,H}[Y, N]$.

Now the right-hand side of (9a) equals

$$\sum_{(\alpha,\beta) \in \mathcal{P}: \hat{X}[\alpha,\beta] > 0} \left[\left(\sum_{i=1}^{n(L)} \frac{\mu^{i,L} p^{i,L}[\alpha, \beta]}{\hat{X}[\alpha, \beta]} \right) \left(\sum_{h=1}^{n(H)} \mu^{h,H} p^{h,H}[\alpha, \beta] p^{h,H}[Y, N] \right) \right] = \quad (10a)$$

$$\sum_{(\alpha,\beta) \in \mathcal{P}: \hat{X}[\alpha,\beta] > 0} \left(\sum_{h=1}^{n(H)} \mu^{h,H} p^{h,H}[\alpha, \beta] p^{h,H}[Y, N] \right), \quad (10b)$$

by the inductive assumption (d) applied to subtree L . The expression in (10b) equals

$$\sum_{h=1}^{n(H)} \left(\left[\sum_{(\alpha,\beta) \in \mathcal{P}: \hat{X}[\alpha,\beta] > 0} p^{h,H}[\alpha, \beta] \right] \mu^{h,H} p^{h,H}[Y, N] \right). \quad (11a)$$

But by inductive property (b) applied to subtree H , given $1 \leq h \leq n(H)$ we have that $p^{h,H}[\alpha, \beta] = 1$ for exactly one partition $(\alpha, \beta) \in \mathcal{P}$, and so expression (11a) equals

$$\sum_{h=1}^{n(H)} \mu^{h,H} p^{h,H}[Y, N]. \quad (12)$$

In summary,

$$\sum_{\alpha,\beta,i,h} \gamma_{ih}^{\alpha,\beta} q_{ih}^{\alpha,\beta}[Y, N] = \sum_{h=1}^{n(H)} \mu^{h,H} p^{h,H}[Y, N].$$

But by induction applied to the subtree H this quantity equals $\hat{X}[Y, N]$. Thus property (d) does indeed hold. ■

3 Mixed-integer polynomial optimization problems

In this section we consider mixed-integer polynomial optimization problems and prove a result (Theorem 31, below) that implies Theorem 6 given in the introduction. Later, in Section 4, we will

introduce a special class of polynomial optimization problems, which we term *graphical*. We will then show how to modify any instance \mathcal{I} of a graphical polynomial optimization problem into an equivalent instance \mathcal{I}' so that an application of Theorem 31 to \mathcal{I}' yields Theorem 9 stated above. This implication will then be used in the context of the AC-OPF problem (Theorems 38, 40 and 43).

To formally describe the problems of interest, let \mathcal{V} be a finite set partitioned as $\mathcal{V} = \mathcal{V}_R \cup \mathcal{V}_Z$, and consider a *mixed-integer polynomial optimization* problem of the form

$$\text{(PO): } f^* \doteq \min f_0(x) \quad (13a)$$

$$\text{subject to: } f_i(x) \geq 0 \quad 1 \leq i \leq m \quad (13b)$$

$$x_j \in [0, 1] \quad \forall j \in \mathcal{V}_R, \quad x_j \in \{0, 1\} \quad \forall j \in \mathcal{V}_Z \quad (13c)$$

where for $0 \leq i \leq m$, $f_i(x)$ is a polynomial, i.e. it has the form

$$f_i(x) = \sum_{k \in I(i)} a_{i,k} \pi_{i,k}(x) \quad (14)$$

where each $I(i)$ is a finite set, the $a_{i,k}$ are rationals and $\pi_{i,k}(x)$ is a monomial in x :

$$\pi_{i,k}(x) = \prod_{j \in T(i,k)} x_j^{p(i,j,k)},$$

where $T(i,k) \subseteq \mathcal{V}$ and each $p(i,j,k) \in \mathbb{Z}_+$. We will also use the following notation:

$$|\pi(i,k)| \doteq \sum_{j \in T(i,k)} p(i,j,k).$$

Any mixed-integer polynomial optimization problem where the feasible region is compact can be reduced to the form (13) by appropriately translating and scaling variables.

Assumption 28 *In what follows we will assume, without loss of generality, that $f_0(x)$ is a linear function. This assumption is justified because if any of the monomials $\pi_{0,k}(x)$ is nonlinear, we can replace it with a new variable $m_{0,k}$ while adding the constraint $m_{0,k} = \pi_{0,k}(x)$ (which implies $0 \leq m_{0,k} \leq 1$). With these modifications in place, the objective function for PO now takes the form, after appropriately redefining \mathcal{V}*

$$f_0(x) = \sum_{j \in \mathcal{V}} c_j x_j. \quad (15)$$

In what follows we will assume that such modifications have been made and that the objective for PO is of the form (15) rather than (13a).

Definition 29 *Given a problem instance of problem PO, let its intersection graph be the undirected graph with vertex-set \mathcal{V} and where for $1 \leq i \leq m$ the set $\bigcup_{k \in I(i)} T(i,k)$ induces a clique.*

Definition 30 *Consider a problem instance of problem PO.*

(a) *Given $\epsilon > 0$, a vector $x \in \mathbb{R}^{\mathcal{V}}$ is **scaled- ϵ feasible** for PO if*

$$f_i(x) \geq -\epsilon \|a_i\|_1, \quad 1 \leq i \leq m. \quad (16)$$

(b) *We set $P_* \doteq \max_{1 \leq i \leq m} \max_{k \in I(i)} |\pi(i,k)|$.*

We will prove, as a consequence of Theorem 12, the following result:

Theorem 31 *Let ω be the width of a tree-decomposition of the intersection graph of an instance \mathcal{I} of problem PO. Let $0 < \epsilon < 1$. Then there is a linear program*

$$LP_2(\mathcal{I}) : \min \{ \hat{c}^T y : \hat{A}y \geq \hat{b} \}$$

with the following properties:

(a) The number of variables and constraints is $O(\epsilon^{-\omega-1} P_*^\omega |\mathcal{V}| \epsilon^{-1} \log \epsilon^{-1})$, and all coefficients are of polynomial size.

(b) Given any feasible solution x to \mathcal{I} , there is a feasible solution y to $LP_2(\mathcal{I})$ with

$$\hat{c}^T y \leq f_0(x) + \epsilon \|c_0\|_1.$$

(c) Given an optimal solution y^* to $LP_2(\mathcal{I})$, we can construct a vector $x^* \in [0, 1]^{\mathcal{V}}$ such that:

$$1. \quad x^* \text{ is scaled-}\epsilon \text{ feasible for PO, and} \tag{17a}$$

$$2. \quad f_0(x^*) = c^T y^*. \tag{17b}$$

Remark 32 Conditions (b) and (17b) indicate that the vector x^* in (c) is approximately optimal for PO, while (17a) states that it is approximately feasible. Condition (a) states that the formulation $LP_2(\mathcal{I})$ is of pseudopolynomial size.

We will prove Theorem 31 in Section 3.0.3.

3.0.2 Relationship of Theorem 31 to prior work

Theorems 31 and 12 leverage the structure of the intersection graph of a problem so as to obtain compact formulations. It is worth noting that when the width ω is small such structure implies sparsity (because a graph with N vertices and tree-width $\leq \omega$ has $O(\omega N)$ edges) but the converse is not true – sparse graphs can have arbitrarily large tree-width.

In the pure continuous case of Theorem 31 prior work has focused on exploiting “sparsity” interpreted in a different manner. Lasserre (2006) [32] considers polynomial optimization problems

$$\min\{f(x) : g_i(x) \geq 0 \text{ for } 1 \leq i \leq m\}. \tag{18}$$

under certain assumptions.

- (1) There is a finite family of subsets I_k , $k = 1, \dots, p$ with $\bigcup_{k=1}^p I_k = \{1, \dots, n\}$. This family has the property that for each polynomial $g_i(x)$ there is an index $1 \leq k \leq p$ such that $g_i(x)$ only involves variables x_j for $j \in I_k$.
- (2) Second, $f(x)$ has the structure $f(x) = \sum_{k=1}^p f_k(x)$ where each polynomial f_k only involves variables from the set I_k .
- (3) Third, for $1 \leq k \leq p-1$ there exists $s \leq k$ such that

$$I_{k+1} \cap \bigcap_{j=1}^k I_j \subseteq I_s.$$

- (4) Finally, the feasible region is assumed to be contained in a compact set in \mathbb{R}^n .

Define $\kappa \doteq \max\{\kappa_1, \kappa_2\}$ where κ_1 is the largest number of variables in a monomial appearing in $f(x)$, and κ_2 is the maximum number of variables appearing in any constraint $g_i(x) \geq 0$. Assuming conditions (1)-(3) hold, [32] presents a hierarchy of semidefinite relaxations for problem (18), where the r^{th} relaxation $r = 1, 2, \dots$ has $O(p\kappa^{2r})$ variables. If we set $n = |\bigcap_{j=1}^k I_j|$, the formulation has

$$O(n\kappa^{2r-1}) \text{ variables, and } m+p \text{ LMI constraints.}$$

These results are related to those in Waki et al [52], also see [25] and Section 8 of [35]. However [32] additionally shows that there is convergence, i.e. as $r \rightarrow +\infty$ the value of the semidefinite relaxation in [32] converges to that of problem (18). In comparison with our results, we can make some remarks.

- (i) First, a problem of the form (18) can be recast into one with linear objective, without increasing the complexity estimate, by adding constraints used to represent each monomial in $f(x)$.
- (ii) Second, condition (3) may not always be attained for a problem of the form (18). Here [32] suggests a procedure for enlarging the sets I_k so that (3) is attained. Together with (i), it can be shown that this procedure renders problem (18) into one in which the intersection graph (in our terminology) has tree-width $\leq \kappa$.
- (iii) It may be possible to argue that under appropriate conditions the semidefinite relaxation in [32] proves exact for finite r ; in such cases it would be of interest to understand the rate of growth of r as a function of problem parameters. Further we note that under the bit model of computing exact solutions to semidefinite programs are not computable.
- (iv) In summary, thus, we expect that the (ϵ -approximate) linear programming formulation in Theorem 31 will in general be smaller than the semidefinite programming formulation obtained from [32].

3.0.3 Proof of Theorem 31

To prove the theorem we will rely on a technique originally due to Glover [23] and used in [8] and [20]. Also see [26] and citations therein. Suppose that $0 \leq r \leq 1$. Then we can approximate r as a sum of inverse powers of 2. Let $0 < \gamma < 1$ and $L = L(\gamma) \doteq \lceil \log_2 \gamma^{-1} \rceil$. Then there exist 0/1-values z_h , $1 \leq h \leq L$, such that

$$\sum_{h=1}^L 2^{-h} z_h \leq r \leq \sum_{h=1}^L 2^{-h} z_h + 2^{-L} \leq \sum_{h=1}^L 2^{-h} z_h + \gamma \leq 1. \quad (19)$$

Next we reformulate problem PO as a problem of type GB. As per Assumption 28 the objective is of the form (15). Recall that $P_* = \max_{1 \leq i \leq m} \max_{k \in I(i)} \sum_{j \in T(i,k)} p(i, j, k)$. For each $1 \leq i \leq m$ and $k \in I(i)$ write

$$\mathbf{R}(i, \mathbf{k}) \doteq T(i, \mathbf{k}) \cap \mathcal{V}_R, \quad \mathbf{Z}(i, \mathbf{k}) \doteq T(i, \mathbf{k}) \cap \mathcal{V}_Z.$$

Let

$$\delta(\gamma) \doteq [1 - (1 - \gamma)^{P_*}],$$

and consider the following formulation:

$$\begin{aligned} \text{(GB}(\gamma)\text{)} : \quad & \min \sum_{j \in \mathcal{V}} c_j \left(\sum_{h=1}^L 2^{-h} z_{j,h} \right) \\ \text{s.t.} \quad & \sum_{k \in I(i)} a_{i,k} \left[\prod_{j \in \mathbf{Z}(i,k)} x_j \prod_{j \in \mathbf{R}(i,k)} \left(\sum_{h=1}^L 2^{-h} z_{j,h} \right)^{p(i,j,k)} \right] \geq -\delta(\gamma) P_* \|a_i\|_1, \quad 1 \leq i \leq m \\ & z_{j,h} \in \{0, 1\}, \quad \forall j \in \mathcal{V}, \quad 1 \leq h \leq L. \end{aligned}$$

Remark. This formulation replaces, in PO, each continuous variable x_j with a sum of powers of two, using the binary variables $z_{j,h}$ in order to effect the approximation (19).

Lemma 33

- (a) Suppose \tilde{x} is feasible for GB. Then there is a vector \tilde{z} feasible for GB(γ) and with objective value at most $c^T \tilde{x} + \delta(\gamma) \|c\|_1$.
- (b) Conversely, if \hat{z} is feasible for GB(γ) then, writing $\hat{x}_j = \sum_{h=1}^L 2^{-h} \hat{z}_{j,h}$ for each $j \in \mathcal{V}$, we have that \hat{x} is scaled- $\delta(\gamma)$ -feasible for PO and has objective value $\sum_{j \in \mathcal{V}} c_j \left(\sum_{h=1}^L 2^{-h} \hat{z}_{j,h} \right)$.

Proof. (a) For $j \in \mathcal{V}$ choose binary values $\tilde{z}_{j,h}$ so as to attain the approximation in (19). Then for each $1 \leq i \leq m$ and $k \in I(i)$ we have

$$\prod_{j \in R(i,k)} \left(\sum_{h=1}^L 2^{-h} \tilde{z}_{j,h} \right)^{p(i,j,k)} \leq \prod_{j \in R(i,k)} \tilde{x}_j^{p(i,j,k)} \leq \prod_{j \in R(i,k)} \left(\sum_{h=1}^L 2^{-h} \tilde{z}_{j,h} \right)^{p(i,j,k)} + \delta(\gamma)P_*,$$

where we use the fact that

$$0 \leq x, 0 \leq y, x + y \leq 1, \text{ and } p \in \mathbb{Z}_+ \Rightarrow (x + y)^p - x^p \leq 1 - (1 - y)^p.$$

Thus \tilde{z} is feasible for $GB(\gamma)$ and the second assertion is similarly proved.

(b) Follows by construction. ■

We can now complete the proof of Theorem 31. Given an instance of problem PO together with a tree-decomposition of its intersection graph, of width ω , we consider formulation $GB(\gamma)$ for $\gamma = \epsilon P_*^{-1}$. As an instance of GB, the formulation has at most $|\mathcal{V}|L(\gamma)$ variables and its intersection graph has width at most $\omega L(\gamma)$. Suppose we apply, to this instance of GB, Theorem 12. We obtain a continuous linear programming reformulation for $GB(\gamma)$, the following properties:

- The reformulation is exact.
- The number of variables and constraints in the reformulation is, for $\epsilon < 1/2$,

$$O(2^{\omega L(\gamma)} |\mathcal{V}| L(\gamma)) = O(\epsilon^{-\omega-1} P_*^\omega |\mathcal{V}| \log \epsilon^{-1}).$$

In view of Lemma 33 the proof of Theorem 31 is complete.

3.0.4 Can the dependence on ϵ be improved upon?

A reader may wonder why or if “exact” feasibility (or optimality) for PO cannot be guaranteed. From a trivial perspective, we point out that there exist simple instances of PO (in fact convex, quadratically constrained problems) where all feasible solutions have irrational coordinates. Should that be the case, if any algorithm outputs an explicit numerical solution in finite time, such a solution will be infeasible. One can, instead, attempt to output solutions that are approximately feasible. This is the case in our result above. Moreover, one can of course select the value of ϵ so as to reduce the feasibility and optimality errors. Theorem 31 indicates the resulting tradeoff in terms of running time. From a more fundamental perspective we have that unless $P = NP$ no polynomial-time algorithm that improves on the formulation in Theorem 31 exists.

4 Graphical mixed-integer polynomial optimization problems

The next problem class we consider are *graphical* polynomial optimization problems. Here we will present a proof of Theorem 9 in the introduction. We will rely on the following formal definition.

Definition 34 (*Graphical mixed-integer polynomial optimization problem*)

(G.1) We are given an undirected, simple graph H .

(G.2) For each vertex v we have a finite set $J(v)$, and for $j \in J(v)$ there is a variable $x_j \in \mathbb{R}$. Denoting $\mathcal{V} = \cup_{v \in V(H)} J(v)$, we are given a partition $\mathcal{V} = \mathcal{V}_R \cup \mathcal{V}_Z$.

(G.3) For each $j \in \mathcal{V}$ the set $\{v \in V(H) : j \in J(v)\}$ induces a connected subgraph of H .

(G.4) We denote by x the vector of all variables x_j for $j \in \cup_{v \in V(H)} J(v)$. For each $v \in V(H)$ and each $u \in \delta(v)$ we denote by $x^{v,u}$ the subvector of x restricted to x_j for $j \in J(v) \cup J(u)$.

(G.5) For each vertex v , $k = 1, \dots, N(v)$ and each $u \in \delta(v)$ we have a family of polynomials $p_{v,u,k}(x^{v,u})$.

(G.6) The family $J = \{J(v) : v \in V(H)\}$ is termed the **index set** of the problem, and we set $J^* = \max\{|J(v)| : v \in V(H)\}$

For $c \in \mathbb{R}^{\mathcal{V}}$ we obtain the problem

$$(GPO): \quad \min c^T x \quad (21a)$$

$$\text{subject to:} \quad \sum_{u \in \delta(v)} p_{v,u,k}(x^{v,u}) \geq 0, \quad v \in V(H), \quad k = 1, \dots, N(v) \quad (21b)$$

$$x_j \in [0, 1] \quad \forall j \in \mathcal{V}_R, \quad x_j \in \{0, 1\} \quad \forall j \in \mathcal{V}_Z. \quad (21c)$$

Clearly each instance of GPO is a special case of problem PO. As was the case for PO, that the more general problem class where the objective is nonlinear, of the form

$$\min \sum_{v \in V(H)} \sum_{u \in \delta(v)} f_{v,u}(x^{v,u}) \quad (22)$$

where each $f_{v,u}$ is a polynomial, can be reduced to the linear objective form (21) by adding new variables and extending the sets $J(v)$. This is done by replacing (22) with $\min \sum_{v \in V(H)} k(v)$, and adding the constraints: $k(v) = \sum_{u \in \delta(v)} f_{v,u}(x^{v,u})$, $\forall v \in V(H)$, where $k(v)$ is a new variable which is added to the set $J(v)$.

A direct application of Theorem 31 to GPO problems will not yield the strongest result one can obtain. The reason for this shortfall is that even if the underlying graph G for an instance of problem GPO has small tree-width, the *intersection graph* arising from that instance may have much larger tree-width if some constraints (21b) have many terms (which requires that G has vertices of high degree).

Example 35 Consider the following optimization problem:

$$\min - \sum_{i=1}^{10} y_i \quad (23a)$$

$$\text{s.t.} \quad y_i = (2x_i - 1)^2 \quad 1 \leq i \leq 10 \quad (23b)$$

$$x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 + 6x_6 + 7x_7 + 8x_8 + 18x_9 + 18x_{10} = 36 \quad (23c)$$

$$x \in [0, 1]^{10}, \quad y \in [0, 1]^{10}. \quad (23d)$$

There are several ways to represent this problem as an instance of GPO where the underlying graph is a tree. For example this can be attained by using the tree with vertex-set $\{v_0, v_1, \dots, v_{10}, u_1, \dots, u_{10}\}$ and edges $\{v_0, v_i\}$ and $\{v_i, u_i\}$ for $1 \leq i \leq 10$ and such that $J(v_0) = \emptyset$, and $J(v_i) = \{x_i\}$ and $J(u_i) = \{y_i\}$ for $1 \leq i \leq 10$. In this case constraint (23c) is of the form (21b) with $v = v_0$.

Thus the tree-width of the underlying graph is 1, yet the intersection graph of problem (23) has a clique of size 10 and so its tree-width is ≥ 9 .

To circumvent this difficulty we will now show how to convert any instance of GPO into an equivalent instance over a graph of bounded degree, while increasing tree-width by at most a constant factor. This will be done in Lemma 37 and Theorem 38. Finally in Theorem 40 we will prove a stronger version of Theorem 9.

Definition 36 Let G be an undirected graph. A **simplification** of G is a graph H with the following properties:

- (1) The maximum degree of a vertex of H is at most 3.
- (2) $V(H)$ is partitioned into a family sets $\{S(v) : v \in V(G)\}$ such that each set $S(v)$ induces a tree in H , and

- (3) For each edge $e = \{v, u\} \in E(G)$ there is an edge $\{v(e), u(e)\} \in E(H)$, termed a **pendant edge**, such that $v(e)$ and $u(e)$ are leaves of the trees induced by $S(v)$ and $S(u)$, respectively. Thus, contracting every set $S(v)$ into a single vertex yields a graph isomorphic to G .

Lemma 37 Let G be an undirected graph and (T, Q) a tree-decomposition of G of width ω . Then there is a simplification \bar{G} of G and a tree-decomposition (\bar{T}, \bar{Q}) of \bar{G} of width at most $2\omega + 1$.

Proof. We first modify (T, Q) in a sequence of steps.

Step 1. For any edge $e = \{u, v\} \in E(G)$, choose an arbitrary $t \in V(T)$ with $e \in Q_t$. Then we modify T by adding to T a new vertex, t^e and the edge $\{t^e, t\}$. Further, we set $Q_{t^e} = \{u, v\}$.

Step 2. Without loss of generality, every vertex of T has degree at most 3. To attain this condition, consider any $t \in V(T)$ with $\delta_T(t) = \{s_1, \dots, s_d\}$ (say) where $d > 3$. Then we alter T by replacing t with two vertices adjacent vertices t^1 and t^2 , such that t^1 is also adjacent to s_1 and s_2 and t^2 is adjacent to s_3, \dots, s_d . Finally, we set $Q_{t^1} = Q_{t^2} = Q_t$. Continuing inductively we will attain the desired condition.

Step 3. We modify T by subdividing each edge $\{t, t'\} \in E(T)$ by introducing a new vertex $r = r(t, t')$. We set $Q_r = Q_t \cap Q_{t'}$. We will refer to each original vertex of T as a *blue* vertex and to each new vertex $r(t, t')$ as a *red* vertex. Denote the new tree by \hat{T} . Then (\hat{T}, Q) is a tree-decomposition of G satisfying the properties set in Steps 1 and 2.

Step 4. We modify the sets Q_t , obtaining sets \hat{Q}_t , as follows:

- (i) For each blue vertex, t and each $v \in Q_t$, we add a new vertex, $v(t)$ to Q_t , and to each set Q_r where r is a red neighbor of t .
- (ii) For each $v \in v(G)$, we remove v from all sets Q_t with $v \in Q_t$.

Moreover, we construct the graph \hat{G} where

$$V(\hat{G}) = \{v(t) : v \in Q_t, t \text{ blue}\},$$

and $E(\hat{G})$ is constructed as follows.

- (a) First, for any $v \in V(G)$ we will have the edge $\{v(t), v(t')\}$ whenever $r(t, t')$ is a vertex of \hat{T} .
- (b) Second, for any vertex of T of the form t^e constructed in Step 1, if $e = \{u, v\}$ (say) then we add to \hat{G} the edge $\{u(t^e), v(t^e)\}$.

Claim 1. \hat{G} is a simplification of G . To see this, consider any vertex $v \in G$. By construction, the set of vertices

$$N(v) \doteq \{s \in V(\hat{T}) : v(t) \in \hat{Q}_s \text{ for some blue } t\}$$

is a subtree of \hat{T} and so is connected. Moreover by rule (a) above this subtree is isomorphic to the subgraph of \hat{Q} induced by the set of vertices $\{v(t) : t \text{ blue}\}$. Together with rule (b), this fact guarantees that contracting every set $N(v)$ into a single vertex yields a graph isomorphic to G . Finally since we assumed that \hat{T} has maximum degree at most three, the same also holds for \hat{G} . This concludes the proof of Claim 1.

Claim 2. The width of (\hat{T}, \hat{Q}) is at most $2\omega + 1$. This fact follows when we observe that t is a blue vertex of \hat{T} , then $|\hat{Q}_t| = |Q_t|$, while for a red vertex $r = r(t, t')$ we have $|\hat{Q}_r| \leq |\hat{Q}_t| + |\hat{Q}_{t'}|$.

Claims 1 and 2 yield complete the proof. ■

Theorem 38 *Suppose we have an instance \mathcal{I} of GPO on a graph G , using index set J and together with a tree-decomposition (T, Q) of G width ω . Let \bar{G} be a simplification of G and (\bar{T}, \bar{Q}) a tree-decomposition of \bar{G} of width $\leq 2\omega + 1$. Then in polynomial time we can construct an equivalent instance \mathcal{I}' of GPO, such that*

- (a) \mathcal{I}' uses an index set \bar{J} with $\bar{J}^* \leq J^* + 4$,
- (b) Each polynomial $p_{v,u,k}$ appearing in a constraint

$$\sum_{u \in \delta(v)} p_{v,u,k}(x^{v,u}) \geq 0 \quad (24)$$

of \mathcal{I} is a sum of polynomials $q_{v,u,k,h}$ appearing in constraints of \mathcal{I}' (here $h = 1, \dots, H = H(v, u, k)$) such that

$$\sum_{h=1}^H \|q_{v,u,k,h}\| \leq 2 \deg_G(v) \|p_{v,u,k}\|.$$

Proof. First, we apply Lemma 37 to obtain the simplification \bar{G} of G with tree-decomposition (\bar{T}, \bar{Q}) . Fix a given vertex $v \in V(G)$ and $1 \leq k \leq N(v)$, and consider the constraint (21b) of GPO. We will next show to modify (24) so as to obtain an equivalent system on graph \bar{G} . Let $T(v)$ be the subgraph of \bar{G} induced by $S(v)$ together with all pendant edges $\{v_e, u_e\}$. Then by (2) and (3) of Definition 36 $T(v)$ is a tree, with degrees at most 3, and each pendant edge $\{v(e), u(e)\}$ in $T(v)$ is such that $u(e)$ is a leaf of $T(v)$.

Let us view $T(v)$ as rooted, with root $r = r(v)$. Write

$$\sigma_i \doteq \sum \{ \|p_{v,u,k}\| : u \in \delta(v), u_e \text{ descendant of } i \}, \quad (25)$$

in other words, the sum coefficients of all polynomials $p_{v,u,k}$ such that u_e is below i (and we omit the dependence on v and k for convenience).

To obtain a system equivalent to (24) we begin by adding, for each edge $\{i, j\} \in E(T(v))$ two new variables $w_{\{i,j\}}^+$ and $w_{\{i,j\}}^-$, and imposing

$$0 \leq w_{\{i,j\}}^+ \leq 1, \quad 0 \leq w_{\{i,j\}}^- \leq 1, \quad \forall \{i, j\}, \quad (26)$$

Next we constrain the new variables by writing an equation for each vertex i of $T(v)$. Suppose first that $i \neq r$ and is also not a leaf, with children j, k and parent h . Then we impose:

$$\sigma_j \left(w_{\{i,j\}}^+ - w_{\{i,j\}}^- \right) + \sigma_k \left(w_{\{i,k\}}^+ - w_{\{i,k\}}^- \right) = \sigma_i \left(w_{\{h,i\}}^+ - w_{\{h,i\}}^- \right). \quad (27)$$

If any of j, k or h do not exist then the corresponding term in (27) is omitted. On the other hand, if i is a leaf, then there is a pendant edge $\{v(e), u(e)\}$ with $i = v(e)$. We write the constraint

$$p_{v(e),u(e),k}(x^{v(e),u(e)}) = \sigma_{v(e)} \left(w_{\{v(e),u(e)\}}^+ - w_{\{v(e),u(e)\}}^- \right). \quad (28)$$

Finally, let the root r have children j and k . Then we write

$$\sigma_j \left(w_{\{r,j\}}^+ - w_{\{r,j\}}^- \right) + \sigma_k \left(w_{\{r,k\}}^+ - w_{\{r,k\}}^- \right) \geq 0, \quad (29)$$

where as before if r only has one child we omit the second term.

We claim that (26)-(29) is equivalent to (24). First we will show that if x satisfies (24) with $0 \leq x_j \leq 1$ for all j then we can construct the vectors w^+, w^- so that (26)-(29) hold. To do so define for any edge $\{i, j\} \in E(T(v))$ (with i the parent of j , say)

$$\sigma_j \left(w_{\{i,j\}}^+ - w_{\{i,j\}}^- \right) = \sum \{ p_{v,u,k}(x) : u \in \delta(v), u_e \text{ descendant of } j \}. \quad (30)$$

By definition of σ_j we can always choose $w_{\{i,j\}}^+$ and $w_{\{i,j\}}^-$ so that (26) holds. Moreover, if $\{i,j\}$ is a pendant edge $\{v(e), u(e)\}$ then (30) is identical to (28). Consider a non-leaf vertex $i \neq r$, with children j and k . Then by construction,

$$\sigma_j + \sigma_k = \sigma_i, \quad (31)$$

and so (27) holds. Finally, by (30), the left-hand side of (29) equals the left-hand side of (24); hence (29) holds as well.

Conversely, suppose we are given a solution x, w^+, w^- to (26)-(29) with $0 \leq x_j \leq 1$ for all j . Then adding (27)-(29) yields (24), as desired.

In summary, by transforming every inequality (24) into (27)-(29) we obtain from an instance \mathcal{I} of GPO an equivalent instance \mathcal{I}' of PO. But we can argue that this is in fact an instance of GPO, as follows. Note that by Definition 36 (1), $V(\bar{G}) = \cup_{v \in V(G)} S(v)$. Let $v \in V(G)$ and $i \in S(v)$, with parent k , say (ignored when i is the root $r(v)$). Then when i is not incident with a pendant edge we set

$$\bar{J}(i) \doteq J(v) \cup \{k^+, k^-\}$$

where k^+ and k^- are associated with $w_{\{i,k\}}^+$ and $w_{\{i,k\}}^-$, resp. And if i is incident with the pendant leaf $\{v(e), u(e)\}$, then we set

$$\bar{J}(i) \doteq J(v) \cup \{k^+, k^-, d^+, d^-\}$$

where k^+ and k^- are as above and d^+ and d^- are associated with $w_{\{v(e), u(e)\}}^+$ and $w_{\{v(e), u(e)\}}^-$, respectively. This definition of the index set captures the structure of constraints (27)-(29). Moreover, the sets $\bar{J}(i)$ have at most four more members than the corresponding sets $J(i)$. Hence (a) holds.

To prove (b), note that at any non-leaf vertex i of $T(v)$, the coefficient norm of the corresponding constraint (28), (27) or (29) is at most $2\sigma_i$ (using (31)). Using that the fact that $T(v)$ has by construction $\deg_G(v)$ leaves and the definition (25) we obtain (b), as desired. ■

4.0.5 Main result

We can now state and prove our main result on graphical mixed-integer problems, where we use the following convention:

Notation 39 Given a polynomial P , its **coefficient norm**, denoted by $\|P\|$, is the sum of absolute values of coefficients in P .

Theorem 40 Suppose we have an instance \mathcal{I} of GPO on a graph G with a tree-decomposition of width ω . Let P^* be the largest sum of (polynomial) degrees in any one of the terms $p_{v,u,k}$ appearing in one of the constraints of \mathcal{I} . Let $0 < \epsilon < 1/2$. Then there is a linear program $LP_3(\mathcal{I}) : \min\{\hat{c}^T y : \hat{A}y \geq \hat{b}\}$ such that:

- (a) The number of variables and constraints is $O(2^{O(\omega J^*)} |\mathcal{V}| P^* \epsilon^{-1} \log \epsilon^{-1})$.
- (b) Given a feasible solution x to \mathcal{I} , there is a feasible solution y to $LP_3(\mathcal{I})$ with $\hat{c}^T y \leq c^T x + \epsilon \|c\|_1$.
- (c) Given an optimal solution \hat{y} to $LP_3(\mathcal{I})$, we can construct a vector $\hat{x} \in [0, 1]^{\mathcal{V}}$ such that:

$$c^T \hat{x} = \hat{c}^T \hat{y} \quad (32a)$$

$$\sum_{u \in \delta(v)} p_{v,u,k}(\hat{x}^{v,u}) \geq -\epsilon \deg_G(v) \|p_{v,k}\|, \quad v \in V(G), \quad k = 1, \dots, N(v). \quad (32b)$$

Here, for $v \in V(G)$ and $1 \leq k \leq N(v)$ we write $\|p_{v,k}\| \doteq \sum_{u \in \delta(v)} \|p_{v,u,k}\|$.

Thus the theorem states that \hat{x} in (b) is scaled- ϵ -feasible for each constraint, as well as approximately optimal as per (a), (32a). The proof will be broken up into several steps.

First, we construct an equivalent instance \mathcal{I}' of GPO on a graph \bar{G} using Theorem 38. We will next argue that there is a tree-decomposition (\bar{T}, \hat{Q}) of the intersection graph of \mathcal{I}' such that the application of Theorem 31 to \mathcal{I}' and (\bar{T}, \hat{Q}) yields Theorem 40.

Let (\bar{T}, \bar{Q}) be a decomposition of \bar{G} . Consider the family of sets $\{\hat{Q}_t : t \in V(\bar{T})\}$ defined by the following rule, where for $v \in V(\bar{G})$ we write $N(v) = J(v) \cup_{u \in \delta_{\bar{G}}(v)} J(u)$:

$$\text{for each } t \in V(\bar{T}), \text{ set } \hat{Q}_t = \bigcup_{v \in \bar{Q}_t} N(v). \quad (33)$$

Then we obtain :

Proposition 41 (\bar{T}, \hat{Q}) is a tree-decomposition of the intersection graph of \mathcal{I}' .

Proof. First we need to prove that for $j \in \mathcal{V}$, the set $\{t \in V(\bar{T}) : j \in \hat{Q}_t\}$ induces a subtree of \bar{T} . For $v \in V(\bar{G})$ let

$$\bar{T}_v = \{t \in V(\bar{T}) : v \in \bar{Q}_t\},$$

and define

$$\nu(j) \doteq \{v \in V(\bar{G}) : j \in J(v)\},$$

and

$$X(j) \doteq \{t \in V(\bar{T}) : \bar{Q}_t \cap \nu(j) \neq \emptyset\} = \bigcup_{v \in \nu(j)} V(\bar{T}_v).$$

By (G.3) of Definition 34, $\nu(j)$ induces a connected subgraph of \bar{G} . Since (\bar{T}, \bar{Q}) is a tree-decomposition of \bar{G} , it follows that $X(j)$ induces a connected subtree of \bar{T} .

Moreover, if for some $t \in V(\bar{T})$ we have $j \in \hat{Q}_t$, then either $t \in X(j)$ or there exists $u \in \bar{Q}_t$ and $v \in \delta_{\bar{G}}(u)$ such that $j \in J(v)$. In the second case, since (\bar{T}, \bar{Q}) is a tree-decomposition of \bar{G} , there is a vertex $t' \in V(\bar{T}_u) \cap V(\bar{T}_v)$. Then $t' \in X(j)$, and, moreover, each vertex t'' in the path between t and t' is contained in \bar{T}_u and thus is such that $j \in \hat{Q}_{t''}$. Thus, indeed the set $\{t \in V(\bar{T}) : j \in \hat{Q}_t\}$ induces a subtree of \bar{T} .

To complete the proof that (\bar{T}, \hat{Q}) is a tree-decomposition of the intersection graph of \mathcal{I}' we also need to show that, the set of variables that appear in any given constraint of \mathcal{I}' appear in some set \hat{Q}_t . But since \mathcal{I}' is an instance of GPO, the set of variables that appear in any given constraint of \mathcal{I}' are a subset of $N(v)$ for some $v \in \bar{G}$. ■

Proposition 42 The width of (\bar{T}, \hat{Q}) is at most $O(\omega J^*)$.

Proof. By construction, for any $t \in \bar{T}$ we have

$$|\hat{Q}_t| \leq \sum_{v \in \bar{Q}_t} |N(v)| \leq \sum_{v \in \bar{Q}_t} |1 + \delta_{\bar{G}}(v)| J^* \leq 4|\bar{Q}_t| J^*. \quad \blacksquare$$

We can now complete the proof of Theorem 40. Suppose we apply Theorem 31 to the pair \mathcal{I}' , (\bar{T}, \hat{Q}) . We claim that this yields (a)-(c) of Theorem 40. Conditions (b) and (c) are clear, and so we just need to show that (a) holds.

To do so, note that the quantity P_* as in Definition 30 (b) is the largest sum of (polynomial) degrees in any of the constraints of the PO formulation. Meanwhile, P^* in the statement of Theorem 40, is the largest sum of degrees in any one of the terms $p_{v,u,k}$ appearing in one of the constraints of \mathcal{I} . But since \bar{G} has degree at most three, it follows that $P_* = O(P^*)$. In view of Propositions 41, 42, we conclude that (a) indeed holds.

This concludes the proof of Theorem 40. ■

4.1 The quadratic case, and the AC-OPF problem

A *graphical quadratic optimization problem* is a graphical (mixed-integer) polynomial optimization problem where each polynomial $p_{v,e}(x^e)$ is quadratic. These problems include, as a special case, the *AC-OPF problem in rectangular coordinates*, which has the following general structure.

- We are given an undirected graph G with vertex-set $V(G)$ and edge-set $E(G)$, possibly including parallel edges. For each $v \in V(G)$ we have two variables, denoted e_v and f_v ³.
- For each vertex v and every edge $\{v, u\}$ we are given two quadratics on the four variables e_v, e_u, f_v, f_u , denoted by $p_{(v,u)}(e_v, e_u, f_v, f_u)$ and $q_{(v,u)}(e_v, e_u, f_v, f_u)$ ⁴.

³The real and imaginary components of the voltage at v , respectively.

⁴Representing active and reactive flow injections into (v, u) .

- For each vertex v we have a one-variable quadratic $C_v(x)$, and six (finite) rationals, M_v^{\min} , M_v^{\max} , P_v^{\min} , P_v^{\max} , Q_v^{\min} and Q_v^{\max} .
- For each edge $\{u, v\}$ we have a finite rational $U_{\{u,v\}}^{\max}$.

The problem can then be stated as:

$$\text{(AC-OPF):} \quad \min \sum_{v \in V(G)} C_v(\gamma_v) \quad (34a)$$

$$\text{s.t.} \quad \gamma_v = \sum_{\{v,u\} \in \delta(v)} p_{(v,u)}(e_v, e_u, f_v, f_u) \quad \forall v \in V(G) \quad (34b)$$

$$M_v^{\min} \leq e_v^2 + f_v^2 \leq M_v^{\max} \quad \forall v \in V(G) \quad (34c)$$

$$P_v^{\min} \leq \sum_{\{v,u\} \in \delta(v)} p_{(v,u)}(e_v, e_u, f_v, f_u) \leq P_v^{\max} \quad \forall v \in V(G) \quad (34d)$$

$$Q_v^{\min} \leq \sum_{\{v,u\} \in \delta(v)} q_{(v,u)}(e_v, e_u, f_v, f_u) \leq Q_v^{\max} \quad \forall v \in V(G) \quad (34e)$$

$$p_{(v,u)}^2(e_v, e_u, f_v, f_u) + q_{(v,u)}^2(e_v, e_u, f_v, f_u) \leq U_{\{u,v\}}^{\max} \quad \forall \{v, u\} \in E(G) \quad (34f)$$

$$p_{(u,v)}^2(e_v, e_u, f_v, f_u) + q_{(u,v)}^2(e_v, e_u, f_v, f_u) \leq U_{\{u,v\}}^{\max} \quad \forall \{v, u\} \in E(G). \quad (34g)$$

In particular settings one may define variations of AC-OPF as stated here, however always having the general structure of quadratic constraints restricted to neighborhoods of vertices. The structure of the quadratics is specific but several variants exist; for the purposes of this paper the above generic structure will suffice. We will use the following notation: given a vector (e, f, g) of variables for AC-OPF, its objective value is denoted by $\mathbf{C}_0(e, f, g)$.

In fact, AC-OPF can be equivalently represented as a graphical quadratic optimization problem. To attain this representation for $v \in V(G)$ we define the index set $J(v)$ to be used in (21) to include the variables γ_v, e_v and f_v as well as an additional variable k_v . We also add to the constraints in (34) the constraint

$$k_v = C_v(\gamma_v)$$

(for each v) and replace the objective function with

$$\min \sum_{k \in V(G)} k_v.$$

In the resulting formulation each constraint is of the form (21b), with quadratic polynomials. After appropriate scaling and translation of variables we also obtain (21c).

AC-OPF problems have recently gathered increased attention, because viewed as QCQPs (quadratically constrained quadratic programs) their SDP relaxation can prove quite tight. See [38] and for additional background, [44]. A broader topic concerns the existence of low-rank solutions to SDP relaxations of QCQPs under special cases, for example in the case that the sparsity pattern corresponds to a tree. See [15], [49]. An alternative approach to AC-OPF is described in [45]. For a recent, brief survey see [9].

On large-scale realistic examples the semidefinite relaxation of AC-OPF problems can prove quite challenging and general-purpose solvers may fail to converge. However, many practical instances of AC-OPF are on graphs G with small tree-width; this provides a vehicle for efficient implementation of the semidefinite programming algorithm under which the SDP relaxation of (34) can in fact be solved in large examples in reasonable CPU time. This observation which draws from fundamental ideas in convex optimization (see e.g. [53], [36]) has been used to develop effective implementations of SDP solvers for AC-OPF problems. See [29], [43], [40]. This approach can even be extended to (partially developed) higher-moment relaxations of AC-OPF [41].

On the negative side, we are not aware of any results that prove a guaranteed approximation bound for the SDP relaxation of (34). One can produce small examples where the lower bound proved by the SDP relaxation is weak [17], [42].

As a result of these developments, one wonders whether AC-OPF itself (and not just a relaxation) can be efficiently solved or approximated when the graph G has small tree-width. As a corollary of Theorem 40, we obtain:

Theorem 43 *Suppose we have an instance \mathcal{I} of AC-OPF on a graph G with n nodes and maximum degree Δ , equipped with a tree-decomposition of width ω . Let $0 < \epsilon < 1/2$. Then there is a linear program*

$$LP_4(\mathcal{I}) : \min\{\hat{c}^T y : Ay \geq b\}$$

with the following properties

- (a) *The number of variables and constraints is $O(\epsilon^{O(-\omega)} n \Delta \log \epsilon^{-1})$.*
- (b) *Given any feasible solution (e, f, γ) to \mathcal{I} , there is a feasible solution y to $LP_4(\mathcal{I})$ with $\hat{c}^T y \leq C_0(\gamma) + n \frac{\epsilon}{\Delta}$.*
- (c) *Given an optimal solution y^* to $LP_4(\mathcal{I})$, we can construct a vector (e^*, f^*, γ^*) such that:*

$$C_0(e^*, f^*, \gamma^*) = \hat{c}^T y^*, \quad \text{and} \tag{35a}$$

$$(e^*, f^*, \gamma^*)^* \text{ is scaled-}\epsilon\text{-feasible for each constraint of AC-OPF.} \tag{35b}$$

4.1.1 Discussion

The complexity of the approximation algorithm implicit in Theorem 43 is, as discussed above, *not* polynomial because of the dependence on ϵ^{-1} in the number of variables and constraints. In fact, AC-OPF is known to be NP-hard on *trees*.

However, this hardness result does not imply that our result is best possible, because AC-OPF is (only) weakly NP-hard on trees. In a sense, our linear program formulation provides a pseudopolynomial formulation that outputs an approximately feasible, approximately optimal solution. But in the spirit of combinatorial optimization problems one would wonder if there is a linear programming formulation with attributes (a) but with stronger guarantees than (c), in particular, one that guarantees optimality and feasibility and errors that do not depend on the norm of the objective and constraints. We comment on this point below.

On the other hand, Theorem 43 is in a sense tight because AC-OPF is strongly NP-hard on general graphs [50]. This makes it unlikely that a pseudopolynomial-complexity algorithm exists for AC-OPF on general graphs attaining approximation guarantees of the form (35).

As was the case for PO problems, similar remarks concerning “exactness” of solutions apply. In general, in finite time we can in general only guarantee approximate feasibility and optimality. Concerning the *quality* of the our feasibility error in (35b) we note that in practical applications the quantities M_v^{\min}, M_v^{\max} are typically chosen (very) close to 1.0, and that the coefficients in the quadratics $p_{(v,u)}$ and $q_{(v,u)}$ are, typically, also relatively small values. Thus our guarantee of scaled- ϵ -feasibility is tantamount to $O(\epsilon)$ additive feasibility error for each constraint.

5 Acknowledgments

Many thanks to Gábor Pataki for useful comments. This research was partially funded by ONR award N00014-13-1-0042, LANL award “Grid Science” and DTRA award HDTRA1-13-1-0021.

References

- [1] S. ARNBORG, D. CORNEIL, AND A. PROSKUROWSKI, *Complexity of finding embeddings in a k -tree*, SIAM Journal on Algebraic Discrete Methods, 8 (1987), pp. 277–284.
- [2] Y. H. AU AND L. TUNÇEL, *A comprehensive analysis of polyhedral lift-and-project methods*, December 2013. arXiv:1312.5972.

- [3] E. BALAS, *Disjunctive programs: Cutting planes from logical conditions*, in Nonlinear Programming, O.L. Mangasarian, R.R. Meyer, and S.M. Robinson, eds., vol. 2, Academic Press, 1975, pp. 279–312.
- [4] E. BALAS, S. CERIA, AND G. CORNUÉJOLS, *A lift-and-project cutting plane algorithm for mixed 0-1 programs*, Math. Program., 58 (1993), pp. 295–324.
- [5] I. BÁRÁNY AND A. PÓR, *0-1 polytopes with many facets*, Advances Math., 161 (2001), pp. 209–228.
- [6] M. BERN, E. LAWLER, AND A. WONG, *Linear-time computation of optimal subgraphs of decomposable graphs*, Journal of Algorithms, 8 (1987), pp. 216 – 235.
- [7] U. BERTELE AND F. BRIOSCHI, *Nonserial Dynamic Programming*, Academic Press, 1972.
- [8] D. BIENSTOCK, *Histogram models for robust portfolio optimization*, J. Comput. Finance, 11 (2007), pp. 1 – 64.
- [9] D. BIENSTOCK, *Progress on solving power flow problems*, Optima, 93 (2013), pp. 1–7.
- [10] D. BIENSTOCK AND M. A. LANGSTON, *Chapter 8 algorithmic implications of the graph minor theorem*, in Network Models, C. M. M.O. Ball, T.L. Magnanti and G. Nemhauser, eds., vol. 7 of Handbooks in Operations Research and Management Science, Elsevier, 1995, pp. 481 – 502.
- [11] D. BIENSTOCK AND N. ÖZBAY, *Tree-width and the Sherali-Adams operator*, Discrete Optimization, 1 (2004), pp. 13–21.
- [12] D. BIENSTOCK AND M. ZUCKERBERG, *Subset algebra lift operators for 0-1 integer programming*, SIAM Journal on Optimization, 15 (2005), pp. 63–95.
- [13] H. BODLAENDER, *Dynamic programming on graphs with bounded treewidth*, in Automata, Languages and Programming, T. Lepist and A. Salomaa, eds., vol. 317 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 1988, pp. 105–118.
- [14] H. BODLAENDER, *A linear-time algorithm for finding tree-decompositions of small treewidth*, SIAM Journal on Computing, 25 (1996), pp. 1305–1317.
- [15] S. BOSE, D. F. GAYME, S. H. LOW, AND K. M. CHANDY, *Quadratically constrained quadratic programs on acyclic graphs with application to power flow*, arXiv:1203.5599v1, (2012).
- [16] D. J. BROWN, M. R. FELLOWS, AND M. A. LANGSTON, *Polynomial-time self-reducibility: Theoretical motivations and practical results*, International Journal of Computer Mathematics, 31 (1989), p. 19.
- [17] C. COFFRIN, D. GORDON, AND P. SCOTT, *Nesta, the NICTA energy system test case archive*, CoRR, abs/1411.0359 (2014).
- [18] M. CONFORTI, G. CORNUJOLS, AND G. ZAMBELLI, *Extended formulations in combinatorial optimization*, 4OR, 8 (2010), pp. 1–48.
- [19] W. CUNNINGHAM AND J. GEELEN, *On integer programming and the branch-width of the constraint matrix*, in Integer Programming and Combinatorial Optimization, M. Fischetti and D. Williamson, eds., vol. 4513 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2007, pp. 158–166.
- [20] S. DASH, O. GÜNLÜK, AND A. LODI, *On the mir closure of polyhedra*, in Integer Programming and Combinatorial Optimization, Springer Berlin Heidelberg, 2007, pp. 337–351.
- [21] D. R. FULKERSON AND O. GROSS, *Incidence matrices and interval graphs*, Pacific Journal of Mathematics, 15 (1965), pp. 835–855.

- [22] D. GATZOURAS, A. GIANNOPOULOS, AND N. MARKOULAKIS, *Lower bound for the maximal number of facets of a 0/1 polytope*, Discrete Comput. Geometry, 34 (2005), pp. 331–349.
- [23] F. GLOVER, *Improved linear integer programming formulations of nonlinear integer problems*, Management Science, 22 (1975), pp. 455–460.
- [24] M. GOEMANS AND L. TUNÇEL, *When does the positive semidefiniteness constraint help in lifting procedures*, Mathematics of Operations Research, 26 (2001), pp. 796–815.
- [25] D. GRIMM, T. NETZER, AND M. SCHWEIGHOFER, *A note on the representation of positive polynomials with structured sparsity*, Arch. Math., 89 (2007), p. 399403.
- [26] A. GUPTE, S. AHMED, M. CHEON, AND S. DEY, *Solving mixed integer bilinear problems using MIP formulations*, SIAM Journal on Optimization, 23 (2013), pp. 721–744.
- [27] R. HALIN, *S-functions for graphs*, Journal of Geometry, 8 (1976), pp. 171–186.
- [28] J. HOOKER, *Logic-based methods for optimization: combining optimization and constraint satisfaction*, John Wiley and Sons, 2000.
- [29] R. JABR, *Zero duality gap in optimal power flow problem*, IEEE Trans. Power Systems, 27 (2013), pp. 1138–1130.
- [30] U. H. KORTENKAMP, J. RICHTER-GEBERT, A. SARANGARAJAN, AND G. M. ZIEGLER, *Extremal properties of 0/1-polytopes*, Discrete & Computational Geometry, 17 (1997), pp. 439–448.
- [31] J. LASSERRE, *An explicit exact sdp relaxation for nonlinear 0-1 programs*, in Integer Programming and Combinatorial Optimization, K. Aardal and B. Gerards, eds., vol. 2081 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2001, pp. 293–303.
- [32] ———, *Convergent SDP-relaxations for polynomial optimization with sparsity*, in Mathematical Software - ICMS 2006, A. Iglesias and N. Takayama, eds., vol. 4151 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 2006, pp. 263–272.
- [33] J. LASSERRE, *Convergent sdp relaxations in polynomial optimization with sparsity*, SIAM Journal on Optimization, 17 (2006), pp. 822–843.
- [34] M. LAURENT, *A comparison of the Sherali-Adams, Lovász-Schrijver and Lasserre relaxations for 0-1 programming*, Mathematics of Operations Research, 28 (2001), pp. 470–496.
- [35] ———, *Sum of squares, moment matrices and optimization over polynomials*, IMA, (2010), pp. 1–147.
- [36] M. LAURENT AND A. VARVITSIOTIS, *A New Graph Parameter Related To Bounded Rank Positive Semidefinite Matrix Completions.*, Mathematical Programming Series A, 145 (2014), pp. 291 – 325.
- [37] S. L. LAURITZEN, *Graphical Models*, Oxford University Press, 1996.
- [38] J. LAVAEI AND S. H. LOW, *Zero duality gap in optimal power flow problem*, IEEE Trans. Power Systems, 27 (2012), pp. 92–107.
- [39] L. LOVÁSZ AND A. SCHRIJVER, *Cones of matrices and set-functions and 0-1 optimization*, SIAM JOURNAL ON OPTIMIZATION, 1 (1991), pp. 166–190.
- [40] R. MADANI, M. ASHRAPHIJO, , AND J. LAVAEI, *OPF Solver*. [http://www.ee.columbia.edu/~sim\\$lavaei/Software.html](http://www.ee.columbia.edu/~sim$lavaei/Software.html), 2014.
- [41] D. MOLZAHN AND I. HISKENS, *Sparsity-exploiting moment-based relaxations of the optimal power flow problem*, To appear in IEEE Transactions on Power Systems, (2014).

- [42] ———, *A tighter second-order cone relaxation of the optimal power flow problem*, submitted, (2014).
- [43] D. MOLZAHN, J. HOLZER, B. LESIEUTRE, AND C. DEMARCO, *Implementation of a large-scale optimal power flow solver based on semidefinite programming*, IEEE Transactions on Power Systems, 28 (2013), pp. 3987–3998.
- [44] D. K. MOLZAHN, *Application of Semidefinite Optimization Techniques to Problems in Electric Power Systems*, dissertation, University of Wisconsin-Madison Department of Electrical and Computer Engineering, August 2013.
- [45] D. T. PHAN, *Lagrangian duality and branch-and-bound algorithms for optimal power flow*, Operations Research, 60 (2012), pp. 275–285.
- [46] N. ROBERTSON AND P. SEYMOUR, *Graph minors. iii. planar tree-width*, Journal of Combinatorial Theory, Series B, 36 (1984), pp. 49 – 64.
- [47] N. ROBERTSON AND P. D. SEYMOUR, *Graph minors II: Algorithmic aspects of tree-width*, Journal of Algorithms, 7 (1986), pp. 309 – 322.
- [48] H. SHERALI AND W. ADAMS, *A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems*, SIAM Journal on Discrete Mathematics, 3 (1990), pp. 411–430.
- [49] S. SOJODI AND J. LAVAEI, *Exactness of semidefinite relaxations for nonlinear optimization problems with underlying graph structure*, SIAM Journal on Optimization, 4 (2014), pp. 1746 – 1778.
- [50] A. VERMA, *Power Grid Security Analysis : An Optimization Approach*, dissertation, Columbia University, 2009. <http://www.columbia.edu/~dano/theses/theses.html/>.
- [51] M. J. WAINWRIGHT AND M. I. JORDAN, *Treewidth-Based conditions for exactness of the Sherali-Adams and Lasserre relaxations*, Tech. Rep. 671, University of California, September 2004.
- [52] H. WAKI, S. KIM, M. KOJIMA, AND M. MURAMATSU, *Sums of squares and semidefinite programming relaxations for polynomial optimization problems with structured sparsity*, SIAM Journal on Optimization, 17 (2006), pp. 218–242.
- [53] H. WOLKOWICZ AND M. F. ANJOS, *Semidefinite programming for discrete optimization and matrix completion problems*, Discrete Appl. Math., 123 (2002), pp. 513–577.
- [54] M. ZUCKERBERG, *A Set Theoretic Approach to Lifting Procedures for 0,1 Integer Programming*, dissertation, Columbia University, 2009. <http://www.columbia.edu/~dano/theses/theses.html/>.