Mathematical Programming
https://doi.org/10.1007/s10107-022-01845-0

FULL LENGTH PAPER

Series B

# A graph-based decomposition method for convex quadratic optimization with indicators

**Peijing Liu[1] · Salar Fattahi[2] · Andrés Gómez[1] · Simge Küçükyavuz[3]**

Received: 4 December 2021 / Accepted: 6 June 2022


## Abstract

In this paper, we consider convex quadratic optimization problems with indicator variables when the matrix $Q$ defining the quadratic term in the objective is sparse. We use a graphical representation of the support of $Q$, and show that if this graph is a path, then we can solve the associated problem in polynomial time. This enables us to construct a compact extended formulation for the closure of the convex hull of the epigraph of the mixed-integer convex problem. Furthermore, motivated by inference problems with graphical models, we propose a novel decomposition method for a class of general (sparse) strictly diagonally dominant $Q$, which leverages the efficient algorithm for the path case. Our computational experiments demonstrate the effectiveness of the proposed method compared to state-of-the-art mixed-integer optimization solvers.

**Keywords** Quadratic optimization · Indicator variables · Sparsity · Decomposition · Graphical models · Fenchel dual · Convex hull

This research is supported, in part, by NSF grants 2006762, 2007814, 2152776, and ONR grant N00014-22-1-2127.

✉ Simge Küçükyavuz
simge@northwestern.edu

Peijing Liu
peijingl@usc.edu

Salar Fattahi
fattahi@umich.edu

Andrés Gómez
gomezand@usc.edu

[1] Daniel J. Epstein Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA, USA

[2] Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI, USA

[3] Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, USA

Published online: 21 June 2022

**Mathematics Subject Classification** 90C11 (Mixed-integer optimization) · 49M27 (decomposition methods) · 90C25 (convex optimization)

## 1 Introduction

Given a positive semi-definite matrix $Q \in \mathbb{R}^{n\times n}$ and vectors $a, c \in \mathbb{R}^n$, we study the mixed-integer quadratic optimization problem

$$\min_{x\in\mathbb{R}^n, z\in\{0,1\}^n} a^\top z + c^\top x + \frac{1}{2}x^\top Qx \tag{1a}$$

$$\text{s.t. } x_i(1 - z_i) = 0 \qquad i = 1, \ldots, n. \tag{1b}$$

Binary vector of indicator variables, $z$, is used to model the support of the vector of continuous variables, $x$. Indeed, if $a_i > 0$, then $z_i = 1 \Leftrightarrow x_i \neq 0$. Problem (1) arises in portfolio optimization [13], sparse regression problems [9, 18], and probabilistic graphical models [40, 43], among others.

### 1.1 Motivation: Inference with graphical models

A particularly relevant application of Problem (1) is in sparse inference problems with Gaussian Markov random fields (GMRFs). Specifically, we consider a special class of GMRF models known as Besag models [10], which are widely used in the literature [11, 12, 31, 37, 48, 56] to model spatio-temporal processes including image restoration and computer vision, disease mapping, and evolution of financial instruments. Given an undirected graph $\mathcal{G}_{MRF} = (N, E)$ with vertex set $N$ and edge set $E$, where edges encode adjacency relationships, and given distances $d_{ij}$ associated with each edge, consider a multivariate random variable $V \in \mathbb{R}^N$ indexed by the vertices of $\mathcal{G}_{MRF}$ with probability distribution

$$p(V) \propto \exp\left(-\sum_{(i,j)\in E} \frac{1}{d_{ij}}(V_i - V_j)^2\right).$$

This probability distribution encodes the prior belief that adjacent variables have similar values. The values of $V$ cannot be observed directly, but rather some noisy observations $y$ of $V$ are available, where $y_i = V_i + \varepsilon_i$, with $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$. Figure 1 depicts a sample GMRF commonly used to model spatial processes, where edges correspond to horizontal and vertical adjacency.

In this case, the maximum a posteriori estimate of the true values of $V$ can be found by solving the optimization problem

$$\min_x \sum_{i\in N} \frac{1}{\sigma_i^2}(y_i - x_i)^2 + \sum_{(i,j)\in E} \frac{1}{d_{ij}}(x_i - x_j)^2. \tag{2}$$

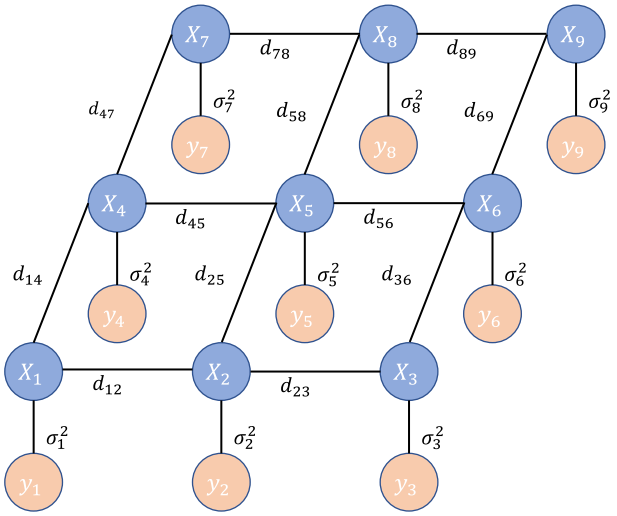


**Fig. 1** Two-dimensional GMRF

Problem (2) can be solved in closed form when there are no additional restrictions on the random variable. However, we consider the situation where the random variable is also assumed to be sparse [7]. For example, few pixels in an image may be salient from the background, few geographic locations may be affected by an epidemic, or the underlying value of a financial instrument may change sparingly over time. Moreover, models such as (2) with sparsity have also been proposed to estimate precision matrices of time-varying Gaussian processes [25]. In all cases, the sparsity prior can be included in model (2) with the inclusion of the $\ell_0$ term $\sum_{i\in N} a_i z_i$, where $a$ is a penalty vector and binary variable $z_i$ indicates whether the corresponding continuous variable $x_i$ is nonzero, for $i \in N$ . This results in an optimization problem of the form (1):

$$\min_{x,z} \sum_{i\in N} \frac{1}{\sigma_i^2}(y_i - x_i)^2 + \sum_{(i,j)\in E} \frac{1}{d_{ij}}(x_i - x_j)^2 + \sum_{i\in N} a_i z_i \tag{3a}$$

$$\text{s.t. } -Mz_i \le x_i \le Mz_i \qquad \forall i \in N \tag{3b}$$

$$x \in \mathbb{R}^N,\ z \in \{0,1\}^N. \tag{3c}$$

Note that constraint (3b) corresponds to the popular big-M linearization of the complementarity constraints (1b). In this case, it can be shown that setting $M = \max_{i\in N} y_i - \min_{i\in N} y_i$ results in a valid mixed-integer optimization formulation. Therefore, it is safe to assume that $(x, z)$ belongs to a compact set $\mathcal{X} = \{(x, z) \in \mathbb{R}^N \times [0, 1]^N : -Mz \le x \le Mz\}$.

### 1.2 Background

Despite problem (1) being NP-hard [17], there has been tremendous progress towards solving it to optimality. Due to its worst case complexity, a common theme for successfully solving (1) is the development of theory and methods for special cases of the

problem, where matrix $Q$ is assumed to have a special structure, providing insights for the general case. For example, if matrix $Q$ is diagonal (resulting in a fully separable problem), then problem (1) can be cast as a convex optimization problem via the perspective reformulation [16]. This convex hull characterization has led to the development of several techniques for problem (1) with general $Q$, including cutting plane methods [26, 27], strong MISOCP formuations [1, 33], approximation algorithms [57], specialized branching methods [35], and presolving methods [5]. Recently, problem (1) has been studied under other structural assumptions, including: quadratic terms involving two variables only [2, 3, 28, 34, 38], rank-one quadratic terms [4, 6, 51, 52], and quadratic terms with Stieltjes matrices [7]. If the matrix can be factorized as $Q = Q_0^\top Q_0$ where $Q_0$ is sparse (but $Q$ is dense), then problem (1) can be solved (under appropriate conditions) in polynomial time [22]. Finally, in [19], the authors show that if the sparsity pattern of $Q$ corresponds to a tree with maximum degree $d$, and all coefficients $a_i$ are identical, then a cardinality-constrained version of problem (1) can be solved in $\mathcal{O}(n^3 d)$ time—immediately leading to an $\mathcal{O}(n^4 d)$ algorithm for the regularized version considered in this paper.

We focus on the case where matrix $Q$ is sparse, and explore efficient methods to solve problem (1). Our analysis is closely related to the support graph of $Q$, defined below.

**Definition 1** Given matrix $Q \in \mathbb{R}^{n\times n}$, the support graph of $Q$ is an undirected graph $\mathcal{G} = (N, E)$, where $N = \{1, \ldots, n\}$ and, for $i < j$, $(i, j) \in E \Leftrightarrow Q_{ij} \neq 0$.

Note that we may assume without loss of generality that graph $\mathcal{G}$ is connected, because otherwise problem (1) decomposes into independent subproblems, one for each connected component of $\mathcal{G}$.

### 1.3 Contributions and outline

In this paper, we propose new algorithms and convexifications for problem (1) when $Q$ is sparse. First, in Sect. 2, we focus on the case when $\mathcal{G}$ is a path. We propose an $\mathcal{O}(n^2)$ algorithm for this case, which improves upon the complexity resulting from the algorithm in [19] without requiring any assumption on vector $a$. Moreover, we provide a compact extended formulation for the closure of the convex hull of

$$X = \left\{(x, z, t) \in \mathbb{R}^n \times \{0, 1\}^n \times \mathbb{R} : t \geq x^\top Q x,\ x_i(1 - z_i) = 0,\ \forall i \in N\right\}$$

for cases where $\mathcal{G}$ is a path, requiring $\mathcal{O}(n^2)$ additional variables. In Sect. 3, we propose a new method for general (sparse) strictly diagonally dominant $Q$, which leverages the efficient algorithm for the path case. In particular, using Fenchel duality, we relax selected quadratic terms in the objective (1a), ensuring that the resulting quadratic matrix has a favorable structure. In Sect. 4, we elaborate on how to select the quadratic terms to relax. Finally, in Sect. 5, we present computational results illustrating that the proposed method can significantly outperform off-the-shelf mixed-integer optimization solvers.

### 1.4 Notation

Given a matrix $Q \in \mathbb{R}^{n\times n}$ and indices $0 \le i < j \le n+1$, we denote by $Q[i, j] \in \mathbb{R}^{(j-i-1)\times(j-i-1)}$ the submatrix of $Q$ from indices $i+1$ to $j-1$. Similarly, given any vector $c \in \mathbb{R}^n$, we denote by $c[i, j] \in \mathbb{R}^{j-i-1}$ the subvector of a vector $c$ from indices $i+1$ to $j-1$. Given a set $S \subseteq \mathbb{R}^n$, we denote by $\text{conv}(S)$ its convex hull and by $\text{cl conv}(S)$ the closure of its convex hull.

## 2 Path Graphs

In this section, we focus on the case where graph $\mathcal{G}$ is a path, that is, there exists a permutation function $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that $(i, j) \in E$ if and only if $i = \pi(k)$ and $j = \pi(k+1)$ for some $k = 1, 2, \ldots, n-1$. Without loss of generality, we assume variables are indexed such that $\pi(k) = k$, in which case matrix $Q$ is tridiagonal and problem (1) reduces to

$$\zeta = \min_{x\in\mathbb{R}^n, z\in\{0,1\}^n} a^\top z + c^\top x + \frac{1}{2}\sum_{i=1}^{n} Q_{ii}{x_i}^2 + \sum_{i=1}^{n-1} Q_{i,i+1}x_i x_{i+1} \tag{4a}$$

$$\text{s.t. } x_i(1-z_i) = 0 \qquad i = 1, \ldots, n. \tag{4b}$$

Problem (4) is interesting in its own right: it has immediate applications in the estimation of one-dimensional graphical models [25] (such as time-varying signals), as well as sparse and smooth signal recovery [7, 44, 58]. In particular, suppose that our goal is to estimate a sparse and smoothly-changing signal $\{x_t\}_{t=1}^n$ from observational data $\{y_t\}_{t=1}^n$. This problem can be written as the following optimization:

$$\min_{x\in\mathbb{R}^n, z\in\{0,1\}^n} a^\top z + \sum_{t=1}^{n}(x_t - y_t)^2 + \sum_{t=1}^{n-1}(x_{t+1} - x_t)^2 \tag{5a}$$

$$\text{s.t. } x_t(1-z_t) = 0 \qquad t = 1, \ldots, n. \tag{5b}$$

The first term in the objective promotes sparsity in the estimated signal, while the second and third terms promote the closeness of the estimated signal to the observational data and its temporal smoothness, respectively. It is easy to see that (5) can be written as a special case of (4).

First, we discuss how to solve (4) efficiently as a shortest path problem. For simplicity, we assume that $Q \succ 0$ (unless stated otherwise).

### 2.1 A shortest path formulation

In this section, we explain how to solve (4) by solving a shortest path problem on an auxiliary directed acyclic graph (DAG). Define for $0 \le i < j \le n+1$

$$w_{ij} \stackrel{\text{def}}{=} \sum_{k=i+1}^{j-1} a_k + \min_{x[i,j]\in\mathbb{R}^{j-i-1}} \left\{ \sum_{k=i+1}^{j-1} c_k x_k + \frac{1}{2} \sum_{k=i+1}^{j-1} Q_{kk} {x_k}^2 + \sum_{k=i+1}^{j-2} Q_{k,k+1} x_k x_{k+1} \right\}$$
$$= \sum_{k=i+1}^{j-1} a_k - \frac{1}{2} c[i,j]^\top Q[i,j]^{-1} c[i,j], \tag{6}$$

where the equality follows from the fact that

$$x^*(i,j) = -Q[i,j]^{-1} c[i,j] \tag{7}$$

is the corresponding optimal solution. By convention, we let $w_{i,i+1} = 0$ for all $i = 0, \ldots, n$.

We start by discussing how to solve a restriction of problem (4) involving only continuous variables. Given any fixed $\bar{z} \in \{0,1\}$, let $x(\bar{z})$ be the unique minimizer of the optimization problem

$$\zeta(\bar{z}) = a^\top \bar{z} + \min_{x\in\mathbb{R}^n} \left\{ c^\top x + \frac{1}{2} \sum_{i=1}^{n} Q_{ii} {x_i}^2 + \sum_{i=1}^{n-1} Q_{i,i+1} x_i x_{i+1} \right\} \tag{8a}$$
$$\text{s.t. } x_i(1-\bar{z}_i) = 0 \qquad i = 1, \ldots, n. \tag{8b}$$

Lemma 1 discusses the structure of the optimal solution $x(\bar{z})$ in (8), which can be expressed using the optimal solutions $x^*(i,j)$ of subproblems given in (7).

**Lemma 1** *Let $0 = v_0 < v_1 < v_2 < \cdots < v_\ell < v_{\ell+1} = n+1$ be the indices such that $\bar{z}_j = 0$ if and only if $j = v_k$ for some $1 \le k \le \ell$ and $\ell \in \{0, \ldots, n\}$. Then $x(\bar{z})_{v_k} = 0$ for $k = 1, \ldots, \ell$, and $x(\bar{z})[v_k, v_{k+1}] = x^*(v_k, v_{k+1})$. Finally, the optimal objective value is $\zeta(\bar{z}) = \sum_{k=0}^{\ell} w_{v_k, v_{k+1}}$.*

***Proof*** Constraints $x_{v_k}(1 - \bar{z}_{v_k}) = 0$ and $\bar{z}_{v_k} = 0$ imply that $x_{v_k} = 0$ in any feasible solution. Moreover, note that since $x_{v_k} = 0$ for all $k = 1 \ldots, \ell$, problem (8) decomposes into $\ell + 1$ independent subproblems, each involving variables $x[v_k, v_{k+1}]$ for $k = 0, \ldots, \ell$. Note that some problems may contain no variables and are thus trivial. Finally, by definition, the optimal solution of those subproblems is precisely $x^*(v_k, v_{k+1})$. The optimal objective value can be verified simply by substituting $x$ with its optimal value. □

Lemma 1 shows that, given the optimal values for the indicator variables, problem (4) is decomposable into smaller subproblems, each with a closed-form solution. This key property suggests that (4) can be cast as a shortest path (SP) problem.

**Definition 2** (SP graph) Define the weighted directed acyclic graph $\mathcal{G}_{SP}$ with vertex set $N \cup \{0, n+1\}$, arc set $A = \{(i,j) \in \mathbb{Z}_+^2 : 0 \le i < j \le n+1\}$ and weights $w$ given in (6). Figure 2 depicts a graphical representation of $\mathcal{G}_{SP}$.

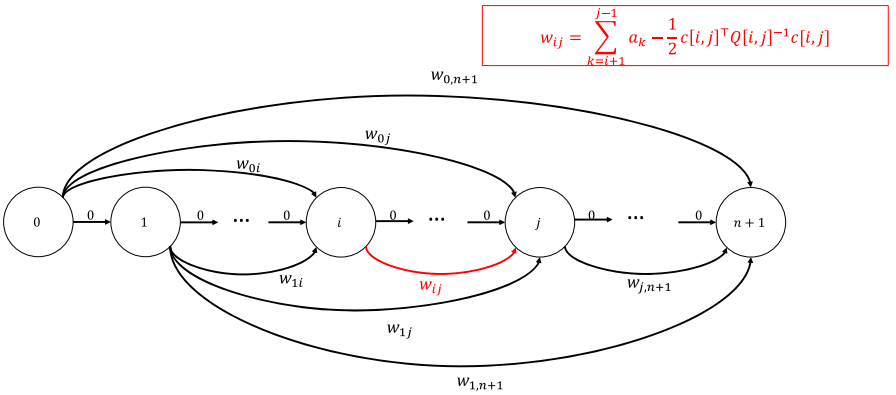


**Fig. 2** Graphical depiction of $\mathcal{G}_{SP}$

**Proposition 1** *The length of any* $(0, n+1)$*-path* $p = \{0, v_1, \ldots, v_\ell, n+1\}$ *on* $\mathcal{G}_{SP}$ *is the objective value of the solution of problem (4) corresponding to setting* $\bar{z}_v = 0$ *if and only if* $v \in p$*, and setting* $x = x(\bar{z})$*.*

***Proof*** There is a one-to-one correspondence between any path $p$ on $\mathcal{G}_{SP}$, and the solution $(x(\bar{z}), \bar{z})$ where $\bar{z}$ is given as in Lemma 1, that is, $\bar{z}_j = 0 \Leftrightarrow j = v_k$ for some $k = 1, \ldots, \ell$. By construction, the length of the path is precisely the objective value associated with $(x(\bar{z}), \bar{z})$. □

Proposition 1 immediately implies that the solution with smallest cost corresponds to a shortest path, which we state next as a corollary.

**Corollary 1** *An optimal solution* $(x(z^*), z^*)$ *of (4) can be found by computing a* $(0, n+1)$*-shortest path on* $\mathcal{G}_{SP}$*. Moreover, the solution found satisfies* $z_i^* = 0$ *if and only if vertex* $i$ *is visited by the shortest path.*

### 2.2 Algorithm

Observe that since graph $\mathcal{G}_{SP}$ is acyclic, a shortest path can be directly computed by a labeling algorithm in complexity linear in the number of arcs $|A|$, which in this case is $\mathcal{O}(n^2)$. Moreover, computing the cost $w_{ij}$ of each arc requires solving the system of equalities $Q[i, j]x[i, j] = -c[i, j]$, which can be done in $\mathcal{O}(n)$ time using Thomas algorithm [20, Chapter 9.4]. Thus, the overall complexity of this direct method is $\mathcal{O}(n^3)$ time, and it requires $\mathcal{O}(n^2)$ memory to store graph $\mathcal{G}_{SP}$. We now show that this complexity can in fact be improved.

**Proposition 2** *Algorithm 1 solves problem (4) in* $\mathcal{O}(n^2)$ *time and using* $\mathcal{O}(n)$ *memory.*

Observe that since Algorithm 1 has two nested loops (lines 3 and 7) and each operation inside the loop can be done in $\mathcal{O}(1)$, the stated time complexity of $\mathcal{O}(n^2)$ follows. Moreover, Algorithm 1 only uses variables $\bar{c}, \bar{q}, \bar{k}, \ell_0, \ldots, \ell_{n+1}$, thus the stated memory complexity of $\mathcal{O}(n)$ follows. Therefore, to prove Proposition 2, it suffices to show that Algorithm 1 indeed solves problem (4).

**Algorithm 1** Algorithm for problem (4)

**Input:** $a, c \in \mathbb{R}^n$, $Q \in R^{n\times n}$ tridiagonal positive definite.
**Output:** Optimal objective value $\zeta$ of (4).
$\ell_0 \leftarrow 0$
$\ell_k \leftarrow \infty$ for $k = 1, \ldots, n+1$ ▷ Shortest path labels, initially $\infty$
**for** $i = 0, \ldots, n$ **do**
  $\ell_{i+1} \leftarrow \min\{\ell_{i+1}, \ell_i\}$ ▷ $w_{i,i+1} = 0$
  $\bar{c} \leftarrow 0, \bar{q} \leftarrow \infty$ ▷ Stores linear and quadratic coefficients
  $\bar{w} \leftarrow 0$
  **for** $j = i+2, \ldots, n+1$ **do**
    $\bar{c} \leftarrow c_{j-1} - \frac{Q_{j-2,j-1}}{\bar{q}}\bar{c}$ ▷ Assume $Q_{0,1} = 0$
    $\bar{q} \leftarrow Q_{j-1,j-1} - \frac{(Q_{j-2,j-1})^2}{\bar{q}}$ ▷ Assume $Q_{0,1} = 0$
    $\bar{w} \leftarrow \bar{w} - \frac{1}{2}\frac{\bar{c}^2}{\bar{q}} + a_{j-1}$ ▷ $\bar{w} = w_{ij}$
    $\ell_j = \min\{\ell_j, \ell_i + \bar{w}\}$
  **end for**
**end for**
**return** $\ell_{n+1}$

Algorithm 1 is based on the forward elimination of variables. Consider the optimization problem (6), which we repeat for convenience

$$w_{ij} = \sum_{k=i+1}^{j-1} a_k + \min_{x[i,j]\in\mathbb{R}^{j-i-1}} \left\{ \sum_{k=i+1}^{j-1} c_k x_k + \frac{1}{2}\sum_{k=i+1}^{j-1} Q_{kk}{x_k}^2 + \sum_{k=i+1}^{j-2} Q_{k,k+1}x_k x_{k+1} \right\}. \tag{9}$$

Lemma 2 shows how we can eliminate the first variable, that is, variable $x_{i+1}$, in (9).

**Lemma 2** *If $j = i+2$, then $w_{ij} = a_{i+1} - \frac{c_{i+1}^2}{2Q_{i+1,i+1}}$. Otherwise,*

$$w_{ij} = a_{i+1} - \frac{c_{i+1}^2}{2Q_{i+1,i+1}} + \sum_{k=i+2}^{j-1} a_k + \min_{x[i+1,j]\in\mathbb{R}^{j-i-2}} \left\{ \sum_{k=i+2}^{j-1} \tilde{c}_k x_k + \frac{1}{2}\sum_{k=i+2}^{j-1} \tilde{Q}_{kk}{x_k}^2 + \sum_{k=i+2}^{j-2} Q_{k,k+1}x_k x_{k+1} \right\},$$

*where $\tilde{c}_{i+2} = c_{i+2} - c_{i+1}\frac{Q_{i+1,i+2}}{Q_{i+1,i+1}}$, $\tilde{c}_k = c_k$ for $k > i+2$, $\tilde{Q}_{i+2,i+2} = Q_{i+2,i+2} - \frac{Q_{i+1,i+2}^2}{Q_{i+1,i+1}}$, and $\tilde{Q}_{kk} = Q_{kk}$ for $k > i+2$.*

***Proof*** If $j = i+2$ then the optimal solution of $\min_{x_{i+1}\in\mathbb{R}}\{a_{i+1} + c_{i+1}x_{i+1} + \frac{1}{2}Q_{i+1,i+1}x_{i+1}^2\}$ is given by $x_{i+1}^* = -c_{i+1}/Q_{i+1,i+1}$, with objective value $a_{i+1} -$

$\frac{c_{i+1}^2}{2Q_{i+1,i+1}}$. Otherwise, from the KKT conditions corresponding to $x_{i+1}$, we find that

$$c_{i+1} + Q_{i+1,i+1}x_{i+1} + Q_{i+1,i+2}x_{i+2} = 0 \implies x_{i+1} = \frac{-c_{i+1} - Q_{i+1,i+2}x_{i+2}}{Q_{i+1,i+1}}.$$

Substituting out $x_{i+1}$ in the objective value, we obtain the equivalent form

$$\begin{aligned}
&\sum_{k=i+1}^{j-1} a_k - \frac{c_{i+1}^2}{Q_{i+1,i+1}} - \frac{c_{i+1}Q_{i+1,i+2}x_{i+2}}{Q_{i+1,i+1}} + \sum_{k=i+2}^{j-1} c_k x_k + \frac{1}{2}\frac{\left(-c_{i+1} - Q_{i+1,i+2}x_{i+2}\right)^2}{Q_{i+1,i+1}} \\
&\quad + \frac{1}{2}\sum_{k=i+2}^{j-1} Q_{kk}x_k^2 - \frac{Q_{i+1,i+2}}{Q_{i+1,i+1}}c_{i+1}x_{i+2} - \frac{\left(Q_{i+1,i+2}x_{i+2}\right)^2}{Q_{i+1,i+1}} + \sum_{k=i+2}^{j-2} Q_{k,k+1}x_k x_{k+1} \\
&= \sum_{k=i+1}^{j-1} a_k - \frac{c_{i+1}^2}{2Q_{i+1,i+1}} + \left(c_{i+2} - c_{i+1}\frac{Q_{i+1,i+2}}{Q_{i+1,i+1}}\right)x_{i+2} + \sum_{k=i+3}^{j-1} c_k x_k \\
&\quad + \frac{1}{2}\left(Q_{i+2,i+2} - \frac{Q_{i+1,i+2}^2}{Q_{i+1,i+1}}\right)x_{i+2}^2 + \frac{1}{2}\sum_{k=i+3}^{j-1} Q_{kk}x_k^2 + \sum_{k=i+2}^{j-2} Q_{k,k+1}x_k x_{k+1}.
\end{aligned}$$

□

The critical observation from Lemma 2 is that, after elimination of the first variable, only the linear coefficient $c_{i+2}$ and diagonal term $Q_{i+2,i+2}$ need to be updated. From Lemma 2, we can deduce the correctness of Algorithm 1, as stated in Proposition 3 and Corollary 2 below.

**Proposition 3** *Given any pair of indices $i$ and $j$ corresponding to the outer (line 3) and inner (line 7) loops of Algorithm 1, respectively, $\bar{w} = w_{ij}$ in line 10.*

***Proof*** If $j = i + 2$, then $\bar{c} = c_{i+1}$, $\bar{q} = Q_{i+1,i+1}$, $\bar{w} = a_{i+1} - \frac{c_{i+1}^2}{2Q_{i+1,i+1}}$ and the conclusion follows from Lemma 2. If $j = i + 3$, then $\bar{c} = \tilde{c}_{i+2}$, $\bar{q} = \tilde{Q}_{i+2,i+2}$, and the conclusion follows from a recursive application of Lemma 2 to the reduced problem, after the elimination of variable $x_{i+1}$. Similarly, cases $j > i+3$ follow from recursive applications of Lemma 2. □

**Corollary 2** *At the end of Algorithm 1, label $\ell_k$ corresponds to the length of the shortest $(0, k)$-path. In particular, Algorithm 1 returns the length of the shortest $(0, n+1)$ path.*

***Proof*** The proof follows due to the fact that line 11 corresponds to the update of the shortest path labels using the natural topological order of $\mathcal{G}_{SP}$. □

***Remark 1*** Algorithm 1 can be easily modified to recover, not only the optimal objective value, but also the optimal solution. This can be done by maintaining the list of predecessors $p$ of each node (initially, $p_k \leftarrow \emptyset$) throughout the algorithm; if label $\ell_j$ is updated at line 11, then set $p_j \leftarrow i$. The solution can then be recovered by backtracking, starting from $p_{n+1}$. ■

### 2.3 Convexification

Recall the definition of

$$X = \left\{(x, z, t) \in \mathbb{R}^n \times \{0, 1\}^n \times \mathbb{R} : t \geq x^\top Q x,\ x_i(1 - z_i) = 0,\ \forall i \in N\right\},$$

which we repeat for convenience. The polynomial time solvability of problem (4) suggests that it may be possible to find a tractable representation of the convex hull of $X$ when $Q$ is tridiagonal. Moreover, given a shortest path (or, equivalently, dynamic programming) formulation of a pure integer linear optimization problem, it is often possible to construct an extended formulation of the convex hull of the feasible region, e.g., see [23, 29, 41, 53]. There have been recent efforts to generalize such methods to nonlinear integer problems [21], but few authors have considered using such convexification techniques in nonlinear mixed-integer problems as the ones considered here. Next, using lifting [47] and the equivalence of optimization over $X$ to a shortest path problem proved in Sect. 2.1, we derive a compact extended formulation for cl conv$(X)$ in the tridiagonal case.

The lifting approach used here is similar to the approach used recently in [6, 32]: the continuous variables are projected out first, then a convex hull description is obtained for the resulting projection in the space of discrete variables, and finally the description is lifted back to the space of continuous variables. Unlike [6, 32], the convexification in the discrete space is obtained using an extended formulation (instead of finding the description in the original space of variables).

In particular, to construct valid inequalities for $X$ we observe that for any $(x, z, t) \in X$ and any $\theta \in \mathbb{R}^n$,

$$\begin{aligned}
&\frac{1}{2}t \geq \frac{1}{2}x^\top Q x \\
&\quad \Leftrightarrow \frac{1}{2}t - \theta^\top x \geq -\theta^\top x + \frac{1}{2}x^\top Q x \\
&\quad \Longrightarrow \frac{1}{2}t - \theta^\top x \geq g_\theta(z) \stackrel{\text{def}}{=} \min_x \left\{-\theta^\top x + \frac{1}{2}x^\top Q x : x_i(1 - z_i) = 0,\ i \in N\right\}.
\end{aligned} \tag{10}$$

We now discuss the convexification in the space of the discrete variables, that is, describing the convex envelope of function $g_\theta$.

#### 2.3.1 Convexification of the projection in the *z* space

We study the epigraph of function $g_\theta$, given by

$$G_\theta = \left\{(z, s) \in \{0, 1\}^n \times \mathbb{R} : s \geq g_\theta(z)\right\}.$$

Note that conv$(G_\theta)$ is polyhedral. Using the results from Sect. 2.1, we now give an extended formulation for $G_\theta$. Given two indices $0 \leq i < j \leq n + 1$ and vector

$\theta \in \mathbb{R}^n$, define the function

$$g_{ij}(\theta) \stackrel{\text{def}}{=} -\frac{1}{2}\theta[i,j]^\top Q[i,j]^{-1}\theta[i,j].$$

Observe that $g_{ij}(\theta) = g_{ij}(-\theta)$ for any $\theta \in \mathbb{R}^n$, and that weights $w_{ij}$ defined in (6) are given by $w_{ij} = g_{ij}(-c[i,j]) + \sum_{k=i+1}^{j-1} a_k$. Moreover, for $0 \le i < j \le n+1$, consider variables $u_{ij}$ intuitively defined as "$u_{ij} = 1$ if and only if arc $(i,j)$ is used in a shortest $(0, n+1)$ path in $\mathcal{G}_{SP}$." Consider the constraints

$$\sum_{i=0}^{n}\sum_{j=i+1}^{n+1} g_{ij}(\theta)u_{ij} \le s \tag{11a}$$

$$\sum_{i=0}^{k-1} u_{ik} - \sum_{j=k+1}^{n+1} u_{kj} = \begin{cases} -1 & \text{if } k = 0 \\ 1 & \text{if } k = n+1 \\ 0 & \text{otherwise.} \end{cases} \qquad k = 0, \ldots, n+1 \tag{11b}$$

$$\sum_{i=0}^{k-1} u_{ik} = 1 - z_k \qquad k = 1, \ldots, n. \tag{11c}$$

$$u \ge 0, 0 \le z \le 1. \tag{11d}$$

**Proposition 4** *If $Q$ is tridiagonal, then the system (11) is an extended formulation of conv$(G_\theta)$ for any $\theta \in \mathbb{R}^n$.*

***Proof*** It suffices to show that optimization over $G_\theta$ is equivalent to optimization over constraints (11). Optimization over $G_\theta$ corresponds to

$$\begin{aligned} &\min_{(z,s)\in G_\theta} \{\beta^\top z + s\} \\ &\quad \Leftrightarrow \min_{x,z} \left\{-\theta^\top x + \beta^\top z + \frac{1}{2}x^\top Q x\right\} \text{ s.t. } x_i(1-z_i) = 0,\ z \in \{0,1\}^n \end{aligned}$$

for an arbitrary vector $\beta \in \mathbb{R}^n$. On the other hand, optimization over (11) is equivalent, after projecting out variables $z$ and $s$, to

$$\begin{aligned} &\min_{u\ge 0} \sum_{i=1}^{n} \beta_i \left(1 - \sum_{\ell=0}^{i-1} u_{\ell i}\right) + \sum_{i=1}^{n}\sum_{j=i+1}^{n+1} g_{ij}(\theta)u_{ij} \text{ s.t. } (11b), (11d) \\ &\quad \Leftrightarrow \min_{u\ge 0} \sum_{i=1}^{n} \beta_i \left(\sum_{\ell=0}^{i-1}\sum_{j=i+1}^{n+1} u_{\ell j}\right) + \sum_{i=1}^{n}\sum_{j=i+1}^{n+1} g_{ij}(\theta)u_{ij} \text{ s.t. } (11b), (11d) \qquad (12) \\ &\quad \Leftrightarrow \min_{u\ge 0} \sum_{i=1}^{n}\sum_{j=i+1}^{n+1} \left(g_{ij}(\theta) + \sum_{\ell=i+1}^{j-1} \beta_\ell\right) u_{ij} \text{ s.t. } (11b), (11d), \end{aligned}$$

where the first equivalence follows from the observation that if node $i$ is not visited by the path ($\sum_{\ell=0}^{i-1} u_{\ell i} = 0$), then one arc bypassing node $i$ is used. The equivalence between the two problems follows from Corollary 1 and the fact that (11b), (11d) are precisely the constraints corresponding to a shortest path problem. □

### 2.3.2 Lifting into the space of continuous variables

From inequality (10) and Proposition 4, we find that for any $\theta \in \mathbb{R}^n$ the linear inequality

$$\frac{1}{2}t - \theta^\top x \geq \sum_{i=0}^{n} \sum_{j=i+1}^{n+1} g_{ij}(\theta) u_{ij} \tag{13}$$

is valid for $X$, where $(u, z)$ satisfy the constraints in (11). Of particular interest is choosing $\theta$ that maximizes the strength of (13):

$$\frac{1}{2}t \geq \max_{\theta \in \mathbb{R}^n} \left\{ \sum_{i=0}^{n} \sum_{j=i+1}^{n+1} g_{ij}(\theta) u_{ij} + \theta^\top x \right\}. \tag{14}$$

**Proposition 5** *If $Q$ is tridiagonal, then inequality (14) and constraints (11b)–(11d) are sufficient to describe cl conv$(X)$ (in an extended formulation).*

Proposition 5 is a direct consequence of Theorem 1 in [47]. Nonetheless, for the sake of completeness, we include a short proof.

***Proof of Proposition 5*** Consider the optimization problem (1), and its relaxation given by

$$\min_{x,z,u} a^\top z + c^\top x + \max_{\theta \in \mathbb{R}^n} \left\{ \sum_{i=0}^{n} \sum_{j=i+1}^{n+1} g_{ij}(\theta) u_{ij} + \theta^\top x \right\} \tag{15a}$$

$$\text{s.t. (11b)–(11d).} \tag{15b}$$

It suffices to show that there exists an optimal solution of (15) which is feasible for (1) with the same objective value, and thus it is optimal for (1) as well. We consider a further relaxation of (15), obtained by fixing $\theta = -c$ in the inner maximization problem:

$$\min_{x,z,u} a^\top z + \sum_{i=0}^{n} \sum_{j=i+1}^{n+1} g_{ij}(-c) u_{ij} \tag{16a}$$

$$\text{s.t. (11b)–(11d).} \tag{16b}$$

In particular, the objective value of (16) is the same for all values of $x \in \mathbb{R}^n$. Moreover, using identical arguments to Proposition 4, we find that the two problems have in

fact the same objective value. Finally, given any optimal solution $z^*$ to (16), the point $(x(z^*), z^*)$ is feasible for (1) and optimal for its relaxation (16) (with the same objective value), and thus this point is optimal for (1) as well. □

We close this section by presenting an explicit form of inequalities (14). Define $\bar{Q}(i,j) \in \mathbb{R}^{n\times n}$ as the matrix obtained by completing $Q[i,j]$ with zeros, that is, $\bar{Q}(i,j)[i,j] = Q[i,j]$ and $\bar{Q}(i,j)_{k\ell} = 0$ otherwise. Moreover, by abusing notation, we define $\bar{Q}(i,j)^{-1} \in \mathbb{R}^{n\times n}$ similarly, that is, $\bar{Q}(i,j)^{-1}[i,j] = Q[i,j]^{-1}$ and $\bar{Q}(i,j)^{-1}_{k\ell} = 0$ otherwise.

**Proposition 6** *For every $(x,u) \in$ cl conv$(X)$, inequality (14) is equivalent to*

$$\begin{pmatrix} t & x^\top \\ x & \left(\sum_{i=0}^{n}\sum_{j=i+1}^{n+1} \bar{Q}(i,j)^{-1}u_{ij}\right) \end{pmatrix} \succeq 0. \tag{17}$$

***Proof*** Note that

$$\begin{aligned} \frac{1}{2}t \geq \max_{\theta\in\mathbb{R}^n} \left\{ \sum_{i=0}^{n}\sum_{j=i+1}^{n+1} g_{ij}(\theta)u_{ij} + \theta^\top x \right\} \\ \Leftrightarrow \quad \frac{1}{2}t \geq \max_{\theta\in\mathbb{R}^n} \left\{ -\sum_{i=0}^{n}\sum_{j=i+1}^{n+1} \left(\frac{1}{2}\theta[i,j]^\top Q[i,j]^{-1}\theta[i,j]\right)u_{ij} + \theta^\top x \right\} \\ \Leftrightarrow \quad \frac{1}{2}t \geq \max_{\theta\in\mathbb{R}^n} \left\{ -\frac{1}{2}\theta^\top \underbrace{\left(\sum_{i=0}^{n}\sum_{j=i+1}^{n+1} \bar{Q}(i,j)^{-1}u_{ij}\right)}_{M(u)} \theta + \theta^\top x \right\}. \end{aligned} \tag{18}$$

Observe that matrix $M(u)$ is positive semidefinite (since it is a nonnegative sum of psd matrices), thus the maximization (18) is a convex optimization problem, and its optimal value takes the form

$$\begin{cases} \frac{1}{2}x^\top M(u)^\dagger x & \text{if } x \in \text{Range}(M(u)), \\ +\infty & \text{if otherwise}, \end{cases}$$

where $M(u)^\dagger$ and Range$(M(u))$ denote the pseudo-inverse and range of $M(u)$, respectively. If $x \notin$ Range$(M(u))$, then inequality (14) is violated, and hence, $(x,u) \notin$ cl conv$(X)$. Therefore, we must have $x \in$ Range$(M(u))$, or equivalently, $M(u)M(u)^\dagger x = x$. In other words,

$$t \geq x^\top M(u)^\dagger x \qquad \text{and} \qquad M(u)M(u)^\dagger x = x.$$

Invoking the Schur complement [14, Appendix A.5] completes the proof. □

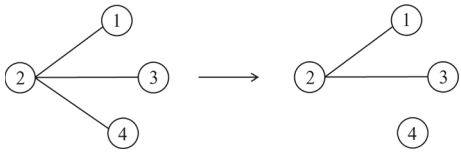


**Fig. 3** Support graphs for Example 1. Left: original; right: after dropping $0.4(x_2 - x_4)^2$

## 3 General (Sparse) Graphs

In this section, we return our attention to problem (1) where graph $\mathcal{G}$ is not a path (but is nonetheless assumed to be sparse), and matrix $Q$ is strictly diagonally dominant, i.e., $D_{ii} \stackrel{\text{def}}{=} Q_{ii} - \sum_{j \neq i} |Q_{ij}| > 0$. In this case, we rewrite the problem as

$$\zeta^* = \min_{x \in \mathbb{R}^n, z \in \{0,1\}^n} a^\top z + c^\top x + \frac{1}{2} \sum_{i=1}^n D_{ii} x_i^2 + \frac{1}{2} \sum_{i=1}^n \sum_{j=i+1}^n |Q_{ij}|(x_i \pm x_j)^2 \quad (19a)$$

$$\text{s.t. } -Mz \leq x \leq Mz, \quad (19b)$$

where $D_{ii} > 0$ for all $i = 1, \ldots, n$. Note that (3) is a special case of (19) with $n = |N|$, $c_i = -2y_i/\sigma_i^2$, $Q_{ij} = -2/d_{ij}$ if $(i, j) \in E$ and $Q_{ij} = 0$ otherwise, $D_{ii} = 2/\sigma_i^2$, and every "$\pm$" sign corresponds to a minus sign.

A natural approach to leverage the efficient $\mathcal{O}(n^2)$ algorithm for the tridiagonal case given in Sect. 2.2 is to simply drop terms $|Q_{ij}|(x_i \pm x_j)^2$ whenever $j > i + 1$, and solve the relaxation with objective

$$a^\top z + c^\top x + \frac{1}{2} \sum_{i=1}^n D_{ii} x_i^2 + \frac{1}{2} \sum_{i=1}^n |Q_{i,i+1}|(x_i \pm x_{i+1})^2. \quad (20)$$

Note that, since $Q$ is strictly diagonally dominant, $D_{ii} > 0$ and objective (20) is convex. Intuitively, if matrix $Q$ is "close" to tridiagonal, the resulting relaxation could be a close approximation to (19). Nonetheless, as Example 1 below shows, the relaxation can in fact be quite loose.

***Example 1*** Consider the optimization problem with support graph given in Fig. 3:

$$\begin{aligned} \zeta^* = \min & -1.3x_1 - 2.5x_2 + 4.6x_3 - 7.8x_4 + 3x_1^2 + 6x_2^2 + 3x_3^2 + 2x_4^2 \\ & - 1.5x_1x_2 - x_2x_3 - 0.8x_2x_4 + 2(z_1 + z_2 + z_3 + z_4) \quad (21a) \\ \text{s.t. } & x_i(1 - z_i) = 0, \ z_i \in \{0, 1\} \qquad i = 1, \ldots, 4. \quad (21b) \end{aligned}$$

The optimal solution of (21) is $x^* = (0, 0, -1.53, 3.9)$ with objective value $\zeta^* \approx -14.74$. After deletion of the term $0.4(x_2 - x_4)^2$ from (21a), we obtain the tridiagonal problem

$$\begin{aligned} \zeta_0 = \min & -1.3x_1 - 2.5x_2 + 4.6x_3 - 7.8x_4 + 3x_1^2 + 5.6x_2^2 + 3x_3^2 + 1.6x_4^2 \\ & - 1.5x_1x_2 - x_2x_3 + 2(z_1 + z_2 + z_3 + z_4) \end{aligned}$$

$$\text{s.t. } x_i(1-z_i)=0,\ z_i\in\{0,1\} \qquad i=1,\dots,4,$$

with optimal solution $x_0^*=(0,0,-1.53,6.5)$ and $\zeta_0=-24.88$. The optimality gap is $\frac{\zeta^*-\zeta_0}{|\zeta^*|}=68.8\%$. □

We now discuss how to obtain improved relaxations of (19), which can result in much smaller optimality gaps.

### 3.1 Convex relaxation via path and rank-one convexifications

The large optimality gap in Example 1 can be attributed to the large effect of completely ignoring some terms $|Q_{ij}|(x_i\pm x_j)^2$. To obtain a better relaxation, we use the following convexification of rank-one terms for set

$$X_2=\left\{(x,z,t)\in\mathbb{R}^2\times\{0,1\}^2\times\mathbb{R}:(x_1\pm x_2)^2\le t,\ x_i(1-z_i)=0,\ i=1,2\right\}.$$

**Proposition 7** *(Atamtürk and Gómez 2019 [4])*

$$cl\ conv(X_2)=\left\{(x,z,t)\in\mathbb{R}^2\times[0,1]^2\times\mathbb{R}:\frac{(x_1\pm x_2)^2}{\min\{1,z_1+z_2\}}\le t\right\}.$$

We propose to solve the convexification of (19) where binary variables are relaxed to $0\le z\le 1$, complementarity constraints are removed, each term $|Q_{ij}|(x_i\pm x_j)^2$ with $j>i+1$ is replaced with the convexification in Proposition 7, and the rest of the terms are convexified using the results of Sect. 2.3. Formally, define the "zero"-indices $1=\tau_1<\cdots<\tau_\ell<\tau_{\ell+1}=n+1$ such that

$$\{\tau_2,\dots,\tau_\ell\}=\left\{i\in N:Q_{i-1,i}=0\right\}.$$

Moreover, given indices $0\le i<j\le n+1$, define

$$X(i,j)=\Big\{(x,z,t)\in\mathbb{R}^{j-i-1}\times\{0,1\}^{j-i-1}\times\mathbb{R}:t\ge x^\top\hat{Q}(i,j)x,\\ x_t(1-z_t)=0,\ t=1,\dots,j-i-1\Big\},$$

where $\hat{Q}(i,j)$ is the tridiagonal $(j-i-1)\times(j-i-1)$ matrix corresponding to indices $i+1$ to $j-1$ in problem (19), that is,

$$\hat{Q}(i,j)_{tk}=\begin{cases}D_{t+i,t+i}+|Q_{t+i,t+i+1}| & \text{if } t=k\\ Q_{t+i,k+i} & \text{if } t-k=\pm1\\ 0 & \text{otherwise.}\end{cases}$$

Because $Q$ is strictly diagonally dominant, the matrix $\hat{Q}(i,j)$ is positive definite. Hence, we propose to use the description of cl conv$(X(i,j))$ given in Proposition 5

to obtain a relaxation of (19) given by

$$\zeta_p = \min_{x,z,t} a^\top z + c^\top x + \frac{1}{2}\sum_{k=1}^{\ell} t_k + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{i,j}| \frac{(x_i \pm x_j)^2}{\min\{1, z_i + z_j\}} \tag{23a}$$

$$\text{s.t. } (x[\tau_k - 1, \tau_{k+1}], z[\tau_k - 1, \tau_{k+1}], t_k) \in \text{cl conv}(X(\tau_k - 1, \tau_{k+1})) \quad k = 1, \ldots, \ell \tag{23b}$$

$$x \in \mathbb{R}^n, z \in [0, 1]^n, t \in \mathbb{R}^\ell. \tag{23c}$$

Note that the strength of relaxation (23) depends on the order in which variables $x_1, \ldots, x_n$ are indexed. We discuss how to choose an ordering in Sect. 4. In the rest of this section, we assume that the order is fixed.

We first establish that relaxation (23) is stronger than the pure rank-one relaxation

$$\zeta_{R1} = \min_{x\in\mathbb{R}^n, z\in[0,1]^n} a^\top z + c^\top x + \frac{1}{2}\sum_{i=1}^{n} D_{ii}\frac{x_i^2}{z_i} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+1}^{n} |Q_{ij}| \frac{(x_i \pm x_j)^2}{\min\{1, z_i + z_j\}} \tag{24}$$

used in [4] obtained via the perspective strengthening of the terms $D_{ii}x_i^2$ (recall that $D_{ii} > 0$ by assumption) and the rank one strengthening of the last term in (20).

**Proposition 8** *Given any vectors $a, c$ and matrix $Q$, $\zeta_{R_1} \le \zeta_p \le \zeta^*$.*

***Proof*** Since both (23) and (24) are relaxations of (19), it follows that $\zeta_{R1} \le \zeta^*$ and $\zeta_p \le \zeta^*$. Moreover, note that

$$\min_{x,z,t} a^\top z + c^\top x + \frac{1}{2}\sum_{k=1}^{\ell} t_k$$

$$\text{s.t. } (x[\tau_k - 1, \tau_{k+1}], z[\tau_k - 1, \tau_{k+1}], t_k) \in \text{cl conv}(X(\tau_k - 1, \tau_{k+1})) \quad k = 1, \ldots, \ell$$

$$x \in \mathbb{R}^n, z \in [0, 1]^n, t \in \mathbb{R}^\ell$$

is an *ideal* relaxation of the discrete problem

$$\min_{x\in\mathbb{R}^n, z\in\{0,1\}^n} a^\top z + c^\top x + \frac{1}{2}\sum_{i=1}^{n} D_{ii}x_i^2 + \frac{1}{2}\sum_{i=1}^{n} |Q_{i,i+1}|(x_i \pm x_{i+1})^2$$

$$\text{s.t. } -Mz \le x \le Mz,$$

whereas (24) is not necessarily an ideal relaxation of the same problem. Since the two relaxations coincide in how they handle terms $|Q_{ij}|(x_i \pm x_j)^2$ for $j > i+1$, it follows that $\zeta_{R1} \le \zeta_p$. □

While relaxation (23) is indeed strong and is conic-representable, it may be difficult to solve using off-the-shelf solvers. Indeed, the direct implementation of (23b) requires the addition of $\mathcal{O}(n^2)$ additional variables *and* the introduction of the positive-semidefinite constraints (17), resulting in a large-scale SDP. We now develop a tailored decomposition algorithm to solve (23) based on Fenchel duality.

Decomposition methods for mixed-integer *linear* programs, such as Lagrangian decomposition, have been successful in solving large-scale instances. In these frameworks, typically a set of "complicating" constraints that tie a large number of variables are relaxed using a Lagrangian relaxation scheme. In this vein, in [24], the authors propose a Lagrangian relaxation-based decomposition method for spatial graphical model estimation problems under cardinality constraints. The Lagrangian relaxation of the cardinality constraint results in a Lagrangian dual problem that decomposes into smaller mixed-integer subproblems, which are solved using optimization solvers. In contrast, in this paper, we use Fenchel duality to relax the "complicating" terms in the objective. This results in subproblems that can be solved in polynomial time, in parallel. Furthermore, the strength of the Fenchel dual results in a highly scalable algorithm that converges to an optimal solution of the relaxation fast. Before we describe the algorithm, we first introduce the Fenchel dual problem.

### 3.2 Fenchel duality

The decomposition algorithm we propose relies on the Fenchel dual [8] of terms resulting from the rank-one convexification.

**Proposition 9** (Fenchel dual) *For all* $(x, z)$ *satisfying* $0 \leq z \leq 1$ *and each* $\alpha, \beta_1, \beta_2 \in \mathbb{R}$

$$\frac{(x_1 \pm x_2)^2}{\min\{1, z_1 + z_2\}} \geq \alpha(x_1 \pm x_2) - \beta_1 z_1 - \beta_2 z_2 - f^*(\alpha, \beta_1, \beta_2), \tag{25}$$

*where*

$$\begin{aligned} f^*(\alpha, \beta_1, \beta_2) &\stackrel{def}{=} \max_{x,z} \ \alpha(x_1 \pm x_2) - \beta_1 z_1 - \beta_2 z_2 - \frac{(x_1 \pm x_2)^2}{\min\{1, z_1 + z_2\}} \\ &= \max\{0, \frac{\alpha^2}{4} - \min\{\beta_1, \beta_2\}\} - \min\{\max\{\beta_1, \beta_2\}, 0\}. \end{aligned}$$

*Moreover, for any fixed* $(x, z)$*, there exists* $\alpha, \beta_1, \beta_2$ *such that the inequality is tight.*

***Proof*** For simplicity, we first assume that the "$\pm$" sign is a minus. Define function $f$ as

$$f(\alpha_1, \alpha_2, \beta_1, \beta_2, x_1, x_2, z_1, z_2) = \alpha_1 x_1 + \alpha_2 x_2 - \beta_1 z_1 - \beta_2 z_2 - \frac{(x_1 - x_2)^2}{\min\{1, z_1 + z_2\}},$$

and consider the maximization problem given by

$$f^*(\alpha_1, \alpha_2, \beta_1, \beta_2) = \max_{x,z} \ f(\alpha_1, \alpha_2, \beta_1, \beta_2, x_1, x_2, z_1, z_2) \tag{26a}$$

$$\text{s.t. } x_1, x_2 \in \mathbb{R}, z_1, z_2 \in [0, 1]. \tag{26b}$$

We now compute an explicit form of $f^*(\alpha_1, \alpha_2, \beta_1, \beta_2)$. First, observe that if $\alpha_1 + \alpha_2 \neq 0$, then $f^*(\alpha_1, \alpha_2, \beta_1, \beta_2) = +\infty$, (with $z_1 = z_2 = 1, x_1 = x_2 = \pm\infty$). Thus we assume without loss of generality that $\alpha_1 + \alpha_2 = 0$, let $\alpha = \alpha_1$ and use the short notation $f^*(\alpha, \beta_1, \beta_2)$ and $f(\alpha, \beta_1, \beta_2, x_1, x_2, z_1, z_2)$ instead of $f^*(\alpha, -\alpha, \beta_1, \beta_2)$ and $f(\alpha, -\alpha, \beta_1, \beta_2, x_1, x_2, z_1, z_2)$, respectively.

To compute a maximum of (26), we consider three classes of candidate solutions.

1. Solutions with $z_1 = z_2 = 0$. If $x_1 = x_2$, then $f(\alpha, \beta_1, \beta_2, x_1, x_2, z_1, z_2) = 0$. Otherwise, if $x_1 \neq x_2$, then $f(\alpha, \beta_1, \beta_2, x_1, x_2, z_1, z_2) = -\infty$.
2. Solutions with $0 < z_1 + z_2 \leq 1$. We claim that we can assume without loss of generality that $z_1 + z_2 = 1$. First, if $z_1 = 0$, then the objective is homogeneous in $z_2$, thus there exists an optimal solution where either $z_2 = 0$ (but this case has already been considered) or $z_2 = 1$. The case $z_2 = 0$ is identical. Finally, if both $0 < z_1, z_2$ and $z_1 + z_2 < 1$, then from the optimality conditions we find that

$$0 = -\beta_1 + \left(\frac{x_1 - x_2}{z_1 + z_2}\right)^2 = -\beta_2 + \left(\frac{x_1 - x_2}{z_1 + z_2}\right)^2,$$

   and in particular this case only happens if $\beta_1 = \beta_2$. In this case the objective is homogeneous in $z_1 + z_2$, thus there exists another optimal solution where either $z_1 + z_2 \to 0$ (already considered) or $z_1 + z_2 = 1$.
   Moreover, in the $z_1 + z_2 = 1$ case,

$$\begin{aligned} f(\alpha, \beta_1, \beta_2, x_1, x_2, z_1, z_2) &= \alpha(x_1 - x_2) - (x_1 - x_2)^2 - \beta_1 z_1 - \beta_2 z_2 \\ &\leq \frac{\alpha^2}{4} - \beta_1 z_1 - \beta_2 z_2 \\ &\leq \frac{\alpha^2}{4} - \min\{\beta_1, \beta_2\}. \qquad \text{(equal if } x_1 - x_2 = \alpha/2) \end{aligned}$$

3. Solutions with $z_1 + z_2 > 1$. In this case, the objective is linear in $z$, thus in an optimal solution, $z$ is at its bound. The only case not considered already is $z_1 = z_2 = 1$, where

$$\begin{aligned} f(\alpha, \beta_1, \beta_2, x_1, x_2, z_1, z_2) &= \alpha(x_1 - x_2) - (x_1 - x_2)^2 - \beta_1 z_1 - \beta_2 z_2 \\ &\leq \frac{\alpha^2}{4} - \beta_1 - \beta_2. \qquad \text{(equal if } x_1 - x_2 = \alpha/2) \end{aligned}$$

Thus, to compute an upper bound on $f^*(\alpha, \beta_1, \beta_2)$, it suffices to compare the three values 0, $\alpha^2/4 - \min\{\beta_1, \beta_2\}$, and $\alpha^2/4 - \beta_1 - \beta_2$ and choose the largest one. The result can be summarized as

$$f^*(\alpha, \beta_1, \beta_2) \leq \max\{0, \frac{\alpha^2}{4} - \min\{\beta_1, \beta_2\}\} - \min\{\max\{\beta_1, \beta_2\}, 0\}.$$

Finally, for a given $(x, z)$, we discuss how to choose $(\alpha, \beta_1, \beta_2)$ so that inequality (25) is tight.

- If $z = 0$ and $x_1 - x_2 = 0$, then set $\alpha = \beta_1 = \beta_2 = 0$.
- If $z = 0$ and $x_1 - x_2 \neq 0$, then set $\alpha = \rho(x_1 - x_2)$ with $\rho \to \infty$, and set $\beta_1 = \beta_2 = \alpha^2/4$.
- If $0 < z_1 + z_2 < 1$, then set $\alpha = 2(x_1 - x_2)$ and $\beta_1 = \beta_2 = (x_1 - x_2)^2/(z_1 + z_2)$.
- If $1 \leq z_1 + z_2$, then set $\alpha = 2(x_1 - x_2)$ and $\beta_1 = \beta_2 = 0$.

□

***Remark 2*** Function $f^*$ is defined by pieces, corresponding to a maximum of convex functions. By analyzing under which cases the maximum is attained, we find that

$$f^*(\alpha, \beta_1, \beta_2) = \begin{cases} 0 & \text{if } \min\{\beta_1, \beta_2\} > \alpha^2/4 \\ \alpha^2/4 - \beta_1 - \beta_2 & \text{if } \max\{\beta_1, \beta_2\} < 0 \\ \alpha^2/4 - \min\{\beta_1, \beta_2\} & \text{otherwise.} \end{cases}$$

Indeed:

1. Case $\min\{\beta_1, \beta_2\} > \alpha^2/4$. Since $\alpha^2/4 - \beta_1 - \beta_2 < \alpha^2/4 - \min\{\beta_1, \beta_2\} < 0$, we conclude that $f^*(\alpha, \beta_1, \beta_2) = 0$ with $x_1^*, x_2^*, z_1^*, z_2^* = 0$.
2. Case $\max\{\beta_1, \beta_2\} < 0$. Since $0 < \alpha^2/4 - \min\{\beta_1, \beta_2\} < \alpha^2/4 - \beta_1 - \beta_2$, we conclude that $f^*(\alpha, \beta_1, \beta_2) = \alpha^2/4 - \beta_1 - \beta_2$, with $x_1^* - x_2^* = \alpha/2$, $z_1^* = z_2^* = 1$.
3. Case $\min\{\beta_1, \beta_2\} \leq \alpha^2/4$ and $\max\{\beta_1, \beta_2\} \geq 0$. Since $0, \alpha^2/4 - \beta_1 - \beta_2 \leq \alpha^2/4 - \min\{\beta_1, \beta_2\}$, we conclude that $f^*(\alpha, \beta_1, \beta_2) = \alpha^2/4 - \min\{\beta_1, \beta_2\}$ with $x_1^* - x_2^* = \alpha/2$, $z_1^* + z_2^* = 1$. ■

From Proposition 9, it follows that problem (23) can be written as

$$\zeta_p = \min_{x,z,t} \max_{\alpha,\beta} a^\top z + c^\top x + \frac{1}{2}\sum_{k=1}^{\ell} t_k + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}|\Big(\alpha_{ij}(x_i \pm x_j) - \beta_{ij,i} z_i - \beta_{ij,j} z_j - f^*(\alpha_{ij}, \beta_{ij,i}, \beta_{ij,j})\Big) \tag{27a}$$

$$\text{s.t. } (x[\tau_k - 1, \tau_{k+1}], z[\tau_k - 1, \tau_{k+1}], t_k) \in \text{cl conv}(X(\tau_k - 1, \tau_{k+1})) \quad k = 1, \ldots, \ell \tag{27b}$$

$$(x, z) \in \mathcal{X}, t \in \mathbb{R}^\ell, \tag{27c}$$

where $\mathcal{X} = \{(x, z) \in \mathbb{R}^n \times [0, 1]^n : -Mz \leq x \leq Mz\}$. Define

$$\psi(x, z, t; \alpha, \beta) \stackrel{\text{def}}{=} \sum_{i=1}^{n}\left(a_i - \frac{1}{2}\sum_{j=i+2}^{n}|Q_{ij}|\beta_{ij,i} - \frac{1}{2}\sum_{j=1}^{i-2}|Q_{ji}|\beta_{ji,i}\right) z_i$$

$$+\sum_{i=1}^{n}\left(c_i+\frac{1}{2}\sum_{j=i+2}^{n}|Q_{ij}|\alpha_{ij}+\frac{1}{2}\sum_{j=1}^{i-2}(\pm|Q_{ji}|\alpha_{ji})\right)x_i+\frac{1}{2}\sum_{k=1}^{\ell}t_k.$$

**Proposition 10** *(Strong duality) Strong duality holds for problem (27), that is,*

$$\zeta_p=\max_{\alpha,\beta}\ h(\alpha,\beta),$$

*where*

$$h(\alpha,\beta)\overset{def}{=}-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n}|Q_{ij}|f^*(\alpha_{ij},\beta_{ij,i},\beta_{ij,j})+\min_{x,z,t}\left\{\psi(x,z,t,\alpha,\beta)\right\} \tag{28a}$$

$$s.t.\ (x[\tau_k-1,\tau_{k+1}],z[\tau_k-1,\tau_{k+1}],t_k)\in cl\ conv(X(\tau_k-1,\tau_{k+1}))$$
$$k=1,\ldots,\ell \tag{28b}$$

$$(x,z)\in\mathcal{X},t\in\mathbb{R}^{\ell}. \tag{28c}$$

***Proof*** Problem (28) is obtained from (27) by interchanging min and max. Equality between the two problems follows from Corollary 3.3 in [49], because $f^*$ is convex (since by definition it is a supremum of affine functions) and $\mathcal{X}$ is compact. □

### 3.3 Subgradient algorithm

Note that for any fixed $(\alpha,\beta)$, the inner minimization problem in (28) decomposes into independent subproblems, each involving variables $(x[\tau_k-1,\tau_{k+1}]$, $z[\tau_k-1,\tau_{k+1}],t_k)$. Moreover, each subproblem corresponds to an optimization problem over cl conv$(X(\tau_k-1,\tau_{k+1}))$, equivalently, optimization over $X(\tau_k-1,\tau_{k+1})$. Therefore, it can be solved in $\mathcal{O}(n^2)$ time using Algorithm 1. Furthermore, the outer maximization problem is concave in $(\alpha,\beta)$ since $h(\alpha,\beta)$ is the sum of the concave functions $-f^*(\alpha_{ij},\beta_{ij,i},\beta_{ij,j})$ and an infimum of affine functions, thus it can in principle be optimized efficiently. We now discuss how to solve the latter problem via a subgradient method.

Similar to Lagrangian decomposition methods for mixed-integer linear optimization [55], subgradients of function $h$ can be obtained directly from optimal solutions of the inner minimization problem. Given any point $(\bar\alpha,\bar\beta_1,\bar\beta_2)\in\mathbb{R}^3$, denote by $\partial f^*(\bar\alpha,\bar\beta_1,\bar\beta_2))$ the subdifferential of $f^*$ at that point. In other words, $(\xi(\bar\alpha),\xi(\bar\beta_1),\xi(\bar\beta_2)))\in\partial f^*(\bar\alpha,\bar\beta_1,\bar\beta_2))$ implies

$$f^*(\alpha,\beta_1,\beta_2)\geq f^*(\bar\alpha,\bar\beta_1,\bar\beta_2)+\xi(\bar\alpha)(\alpha-\bar\alpha)+\xi(\bar\beta_1)(\beta_1-\bar\beta_1)+\xi(\bar\beta_2)(\beta_2-\bar\beta_2) \tag{29}$$

for all $(\alpha,\beta_1,\beta_2)\in\mathbb{R}^3$. The next proposition shows that subgradients $\rho(\alpha,\beta)\in\mathbb{R}^{(3/2)(n-1)(n-2)}$ of function $h(\alpha,\beta)$ (for maximization) can be obtained from subgradients of $f^*$ and optimal solutions $(\bar x,\bar z)$ of the inner minimization problem in (28), as

$$\begin{pmatrix} \rho(\alpha_{ij}) \\ \rho(\beta_{ij,i}) \\ \rho(\beta_{ij,j}) \end{pmatrix} = -\begin{pmatrix} \xi(\alpha_{ij}) \\ \xi(\beta_{ij,i}) \\ \xi(\beta_{ij,j}) \end{pmatrix} + \begin{pmatrix} \bar{x}_i \pm \bar{x}_j \\ -\bar{z}_i \\ -\bar{z}_j \end{pmatrix}.$$

Later, in Proposition 12, we explicitly describe the subgradients $\xi$ of $f^*$.

**Proposition 11** *Given any $(\bar{\alpha}, \bar{\beta}) \in \mathbb{R}^{(3/2)(n-1)(n-2)}$, let $(\bar{x}, \bar{z}, \bar{t})$ denote an optimal solution of the associated minimization problem (28), and let*

$$(\xi(\bar{\alpha}_{ij}), \xi(\bar{\beta}_{ij,i}), \xi(\bar{\beta}_{ij,j})) \in \partial f^*(\bar{\alpha}_{ij}, \bar{\beta}_{ij,i}, \bar{\beta}_{ij,j})$$

*for all $j > i+1$. Then for any $(\alpha, \beta) \in \mathbb{R}^{(3/2)(n-1)(n-2)}$,*

$$\begin{aligned} h(\alpha,\beta) \le h(\bar{\alpha},\bar{\beta}) &+ \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| \left(-\xi(\bar{\alpha}_{ij}) + \bar{x}_i \pm \bar{x}_j\right)(\alpha_{ij} - \bar{\alpha}_{ij}) \\ &+ \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| \left(-\xi(\bar{\beta}_{ij,i}) - \bar{z}_i\right)(\beta_{ij,i} - \bar{\beta}_{ij,i}) \\ &+ \frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| \left(-\xi(\bar{\beta}_{ij,j}) - \bar{z}_j\right)(\beta_{ij,j} - \bar{\beta}_{ij,j}). \end{aligned}$$

***Proof*** Given $(\alpha, \beta) \in \mathbb{R}^{(3/2)(n-1)(n-2)}$, let $(x^*, y^*, t^*)$ be the associated solution of the inner minimization problem (28). Then we deduce that

$$\begin{aligned} h(\alpha,\beta) &= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| f^*(\alpha_{ij},\beta_{ij,i},\beta_{ij,j}) + \psi(x^*,z^*,t^*;\alpha,\beta) \\ &\le -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| \Big( f^*(\bar{\alpha}_{ij},\bar{\beta}_{ij,i},\bar{\beta}_{ij,j}) + \xi(\bar{\alpha}_{ij})(\alpha_{ij}-\bar{\alpha}_{ij}) + \xi(\bar{\beta}_{ij,i})(\beta_{ij,i}-\bar{\beta}_{ij,i}) \\ &\quad + \xi(\bar{\beta}_{ij,j})(\beta_{ij,j}-\bar{\beta}_{ij,j}) \Big) + \psi(x^*,z^*,t^*;\alpha,\beta) \\ &\le -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| \Big( f^*(\bar{\alpha}_{ij},\bar{\beta}_{ij,i},\bar{\beta}_{ij,j}) + \xi(\bar{\alpha}_{ij})(\alpha_{ij}-\bar{\alpha}_{ij}) + \xi(\bar{\beta}_{ij,i})(\beta_{ij,i}-\bar{\beta}_{ij,i}) \\ &\quad + \xi(\bar{\beta}_{ij,j})(\beta_{ij,j}-\bar{\beta}_{ij,j}) \Big) + \psi(\bar{x},\bar{z},\bar{t};\alpha,\beta) \\ &= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=i+2}^{n} |Q_{ij}| \Big( f^*(\bar{\alpha}_{ij},\bar{\beta}_{ij,i},\bar{\beta}_{ij,j}) + \left(\xi(\bar{\alpha}_{ij}) + \bar{x}_i \pm \bar{x}_j\right)(\alpha_{ij}-\bar{\alpha}_{ij}) \\ &\quad + \left(\xi(\bar{\beta}_{ij,i}) - \bar{z}_i\right)(\beta_{ij,i}-\bar{\beta}_{ij,i}) + \left(\xi(\bar{\beta}_{ij,j}) - \bar{z}_j\right)(\beta_{ij,j}-\bar{\beta}_{ij,j}) \Big) + \psi(\bar{x},\bar{z},\bar{t};\bar{\alpha},\bar{\beta}), \end{aligned}$$

where the first inequality follows from (29), the second inequality follows since $(x^*, z^*, t^*)$ is a minimizer of $\psi(\cdot, \alpha, \beta)$ whereas $(\bar{x}, \bar{z}, \bar{t})$ may not be, and the last equality follows since $\psi$ is linear in $(\alpha, \beta)$ for fixed $(\bar{x}, \bar{z}, \bar{t})$. The conclusion follows. □

**Proposition 12** *A subgradient of function $f^*$ admits a closed form solution as*

$$(\xi(\alpha_{ij}), \xi(\beta_{ij,i}), \xi(\beta_{ij,j})) = \begin{cases} (0,0,0) & \text{if } \beta_{ij,i}, \beta_{ij,j} > \alpha_{ij}^2/4, \\ (\alpha_{ij}/2, -1, 0) & \text{if } \beta_{ij,i} \le \alpha_{ij}^2/4 \ \& \ \beta_{ij,j} \ge 0 \ \& \ \beta_{ij,j} \ge \beta_{ij,i}, \\ (\alpha_{ij}/2, 0, -1) & \text{if } \beta_{ij,j} \le \alpha_{ij}^2/4 \ \& \ \beta_{ij,i} \ge 0 \ \& \ \beta_{ij,i} > \beta_{ij,j}, \\ (\alpha_{ij}/2, -1, -1) & \text{if } \beta_{ij,i}, \beta_{ij,j} < 0. \end{cases}$$

***Proof*** The result follows since $f^*$ is the supremum of affine functions described in Remark 2. □

Algorithm 2 states the proposed method to solve problem (28). Initially, $(\alpha, \beta)$ is set to zero (line 1). Then, at each iteration: a primal solution $(\bar{x}, \bar{z}, \bar{t})$ of (28) is obtained by solving $\ell$ independent problems of the form (4), using Algorithm 1 (line 4); a subgradient of function $h$ is obtained directly from $(\bar{x}, \bar{z})$ using Propositions 11 and 12 (line 5); finally, the dual solution is updated using first order information (line 6).

**Algorithm 2** Subgradient ascent

**Input:** $a, c \in \mathbb{R}^n$, $Q \succ 0$ diagonally dominant.
**Output:** $\zeta_p$.
1: $(\alpha, \beta) \leftarrow (0, 0)$ ▷ Initialize
2: $k \leftarrow 1$ ▷ Iteration counter
3: **repeat**
4: $(\bar{x}, \bar{z}, \bar{t}) \in \arg\min_{x,z,t} \psi(x, z, t; \alpha, \beta)$ ▷ Algorithm 1
5: $\rho(\bar{x}, \bar{z}) \in \partial h(\alpha, \beta)$ ▷ Propositions 11 and 12
6: $(\alpha, \beta) \leftarrow (\alpha, \beta) + s_k \rho(\bar{x}, \bar{z}) / \|\rho(\bar{x}, \bar{z})\|_2$ ▷ $s_k$ =Step size at iteration $k$
7: $k \leftarrow k + 1$ ▷ Iteration counter
8: **until** Termination criterion is met
9: **return** $h(\alpha, \beta)$

We now discuss some implementation details. First, note that in line 1, $(\alpha, \beta)$ can in fact be initialized to an arbitrary point without affecting correctness of the algorithm. Nonetheless, by initializing at zero, we ensure that the first iteration of Algorithm 2 corresponds to solving the relaxation obtained by completely dropping the complicating quadratic terms; see Example 1. Second, each time a primal solution is obtained (line 4), a lower bound $h(\alpha, \beta) \le \zeta_p \le \zeta^*$ can be computed. Moreover, since the solution $(\bar{x}, \bar{z})$ is feasible for (19), an upper bound $\bar{\zeta}(\bar{x}, \bar{z}) \ge \zeta^*$ can be obtained by simply evaluating the objective function (19a). Thus, at each iteration of Algorithm 2, an estimate of the optimality gap of $(\bar{x}, \bar{z})$ can be computed as $\texttt{gap} = (\bar{\zeta}(\bar{x}, \bar{z}) - h(\alpha, \beta))/\bar{\zeta}(\bar{x}, \bar{z})$. Third, a natural termination criterion (line 8) we use in our experiments is to terminate if $\texttt{gap} \le \epsilon$ for some predefined optimality tolerance $\epsilon$, if $k \ge \bar{k}$ for some iteration limit $\bar{k}$, or if a given time limit is exceeded. Note that under mild conditions [15, 45, 46] (e.g., if $s_k = 1/k$), the objective value returned by Algorithm 2 converges to $\zeta_p$, that is, the objective value of the convex relaxation (23). However, since in general the convex relaxation is not exact (unless matrix $Q$ is tridiagonal) and $\zeta_p < \zeta^*$, we find that the estimated gap produced by Algorithm 2 may

**Table 1** Algorithm 2 applied to problem (30) with $s_k = \frac{1}{1.01^k}$

| Iteration | $\bar{x}$ | $\alpha$ | $\beta_1$ | $\beta_2$ | $\zeta_p$ | Gap |
|---|---|---|---|---|---|---|
| 1 | (0,0,–1.53,6.50) | 0 | 0 | 0 | –24.87 | 67.93% |
| 2 | (0,0,–1.53,6.16) | –1.0 | 0.25 | 0 | –22.44 | 57.23% |
| 3 | (0,0,–1.53,5.83) | –1.99 | 0.25 | 0 | –20.36 | 46.04% |
| 4 | (0,0,–1.53,5.50) | –2.97 | 0.25 | 0 | –18.62 | 34.78% |
| 5 | (0,0,–1.53,5.18) | –3.94 | 0.25 | 0 | –17.21 | 24.02% |
| 6 | (0,0,–1.53,4.86) | –4.90 | 0.25 | 0 | –16.13 | 14.45% |
| 7 | (0,0,–1.53,4.54) | –5.85 | 0.25 | 0 | –15.36 | 6.84% |
| 8 | (0,0,–1.53,4.23) | –6.79 | 0.25 | 0 | –14.90 | 1.88% |
| 9 | (0,0,–1.53,3.92) | –7.72 | 0.25 | 0 | –14.73 | $< 0.01\%$ |

never reach 0, that is, $\texttt{gap} \geq (\zeta^* - \zeta_p)/\zeta_p$. Thus, it is necessary to have a termination criterion other than the optimality gap, as otherwise Algorithm 2 may not terminate for small values of $\epsilon$.

We close this section by revisiting Example 1, demonstrating that Algorithm 2 can indeed achieve substantially improved optimality gaps.

***Example 1*** [Continued] The Fenchel dual of (21) is

$$\zeta_p = \max_{\alpha,\beta} -0.4 f^*(\alpha, \beta_1, \beta_2) + \min_{x,z} \Big\{ -1.3x_1 + (-2.5 + 0.4\alpha)x_2 + 4.6x_3 \tag{30a}$$

$$- (7.8 + 0.4\alpha)x_4 + 3x_1^2 + 5.6x_2^2 + 3x_3^2 + 1.6x_4^2 - 1.5x_1x_2 \tag{30b}$$

$$- x_2x_3 + 2z_1 + (2 - 0.4\beta_1)z_2 + 2z_3 + (2 - 0.4\beta_2)z_4 \Big\} \tag{30c}$$

$$\text{s.t. } x_i(1 - z_i) = 0, z_i \in \{0, 1\}, \qquad i = 1, \ldots, 4. \tag{30d}$$

Table 1 shows the first nine iterations of Algorithm 2.

## 4 Path Decomposition

In the previous section, we showed that if $Q$ does not possess a tridiagonal structure, then it is possible to relax its "problematic" elements via their Fenchel duals, and leverage Algorithm 1 to solve the resulting relaxation. In this section, our goal is to explain how to select the nonzero elements of $Q$ to be relaxed via our proposed method. In particular, our goal is to obtain the best permutation matrix $P$ such that $PQP^\top$ is close to tridiagonal.

To achieve this goal, we propose a path decomposition method over $\mathcal{G}$, where the problem of finding the best permutation matrix for $Q$ is reformulated as finding a maximum weight subgraph of $\mathcal{G}$, denoted as $\widetilde{\mathcal{G}}$, that is a union of paths. In particular, define $y_{ij}$ as an indicator variable that takes the value 1 if and only if edge $(i, j)$ is

included in the subgraph. Therefore, the problem of finding $\widetilde{\mathcal{G}}$ reduces to:

$$p^* = \max \sum_{(i,j)\in E} |Q_{ij}| y_{ij} \tag{31a}$$

$$s.t. \sum_{j\in\delta(i)} y_{ij} \leq 2 \qquad \forall i = 1, 2, \ldots, n \tag{31b}$$

$$\sum_{i,j\in S} y_{ij} \leq |S| - 1 \qquad \forall S \subseteq \{1, 2, \ldots, n\} \tag{31c}$$

$$y_{ij} \in \{0, 1\} \qquad \forall (i, j) \in E, \tag{31d}$$

where $\delta(i)$ denotes the neighbors of node $i$ in $\mathcal{G}$. Let the objective function evaluated at a given $y$ be denoted as $p(y)$. Moreover, let $y^*$ and $p^*$ be an optimal solution and its corresponding objective value respectively. Constraints (31b) ensure that the constructed graph is the union of cycles and paths, whereas constraints (31c) are cycle-breaking constraints [39, 42, 54]. Despite the exponential number of constraints (31c), it is known that cycle elimination constraints can often be efficiently separated [54]. Nonetheless, our next result shows that problem (31) is indeed NP-hard.

**Theorem 1** *Problem (31) is NP-hard.*

***Proof*** We use a reduction from Hamiltonian path problem: given an arbitrary (unweighted) graph $\mathcal{G}$, the Hamiltonian path problem asks whether there exists a simple path that traverses every node in $\mathcal{G}$. It is known that Hamiltonian path problem is NP-complete [30].

Given an arbitrary graph $\mathcal{G}(N, E)$, construct an instance of (31) with $|Q_{ij}| = 1$ if $(i, j) \in E$, and $|Q_{ij}| = 0$ otherwise. Let us denote the optimal solution to the constructed problem as $y^*$, and the graph induced by this solution as $\mathcal{G}^*(N, E^*)$. In other words, $(i, j) \in E^*$ if and only if $y^*_{ij} = 1$. It is easy to see that $\sum_{(i,j)\in E} y^*_{ij} \leq n - 1$; otherwise, the graph $\mathcal{G}^*$ contains a cycle, which is a contradiction. Therefore, we have $p(y^*) \leq n - 1$. We show that, we have $p(y^*) = n - 1$ if and only if $\mathcal{G}$ contains a Hamiltonian path. This immediately completes the proof.

First, suppose that $\mathcal{G}$ has a Hamiltonian path. Therefore, there exists a path in $\mathcal{G}$ with exactly $n - 1$ edges. A solution $\tilde{y}$ defined as $\tilde{y}_{ij} = 1$ for every edge $(i, j)$ in the path, and $\tilde{y}_{ij} = 0$ otherwise is feasible for the constructed instance of (31), and it has the objective value $p(\tilde{y}) = p(y^*) = n - 1$. Conversely, suppose that $p(y^*) = n - 1$. Then, the graph $\mathcal{G}^*$ has exactly $n - 1$ edges, and it is a union of paths. It is easy to see that if $\mathcal{G}^*$ has at least two components, then $|E^*| < n - 1$, which is a contradiction. Therefore, $\mathcal{G}^*$ is a Hamiltonian path. □

Due to hardness of (31), we propose in this section an approximation algorithm based on the following idea:

1. Find a vertex disjoint path/cycle cover of $\mathcal{G}$, that is, a subset $\widetilde{E}$ of the edges of $E$ such that, in the induced subgraph of $\mathcal{G}$, each connected component is either a cycle or a path.
2. From each cycle, remove the edge $(i, j) \in \widetilde{E}$ with least value $|Q_{ij}|$.

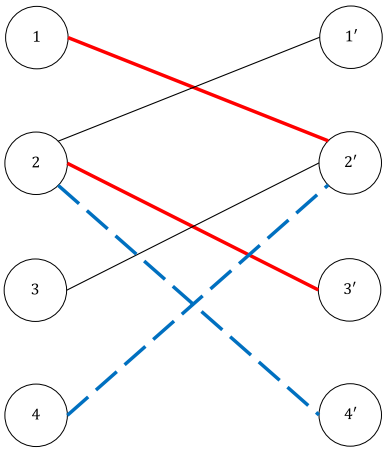


**Fig. 4** Graph $\mathcal{G}_M$ corresponding to the support graph in Fig. 3 (left). Bold red: Max. cardinality matching corresponding to the decomposition shown in Fig. 3 (right). Dashed blue: Alternative max. cardinality matching corresponding to using edge $(2, 4)$ twice, resulting in a length two cycle

Note that a vertex disjoint cycle cover can be found by solving a bipartite matching problem [50] on an auxiliary graph, after using a node splitting technique. Specifically, create graph $\mathcal{G}_M = (V_M, E_M)$ with $V_M = N \cup \{1', 2', \ldots, n'\}$ and $E_M$ that is determined as follows: if $(i, j) \in E$, then $(i, j') \in E_M$ and $(i', j) \in E_M$. Then any matching on $\mathcal{G}_M$ corresponds to a cycle cover in $\mathcal{G}$, with edge $(i, j')$ in the matching encoding that "$j$ follows $i$ in a cycle." Figure 4 illustrates how to obtain cycle covers via bipartite matchings.

In the resulting decomposition from this method, each connected component is a cycle, possibly of length two (that is, using the same edge twice). However, in inference problems with graphical models, graph $\mathcal{G}$ is often bipartite (see Figure 1), in which case we propose an improved method which consists of solving the integer optimization problem

$$\max \sum_{(i,j)\in E} |Q_{ij}| y_{ij} \tag{32a}$$

$$s.t. \sum_{j\in\delta(i)} y_{ij} \leq 2, \qquad \forall i = 1, 2, \ldots, n \tag{32b}$$

$$y_{ij} \in \{0, 1\}, \qquad \forall (i, j) \in E. \tag{32c}$$

Problem (32) is obtained from (31) after dropping the cycle elimination constraints (31c). Note that in any feasible solution of (32) each edge can be used only once, thus preventing cycles of length two. Moreover, some of the connected components may already be paths. Finally, we note that (32) is much simpler than (31), since it can be solved in polynomial time for certain graph structures.

**Proposition 13** *For bipartite $\mathcal{G}$, the linear programming relaxation of the problem (32) is exact.*

***Proof*** It is easy to verify that the constraint matrix for (32) is totally unimodular if $\mathcal{G}$ is bipartite, and therefore, the linear programming relaxation of (32) has integer solutions. □

Let the optimal solution to (32) be denoted as $\hat{y}$. Define the weighted graph $\widehat{\mathcal{G}}(N, \widehat{E}) \subseteq \mathcal{G}(N, E)$ induced by $\hat{y}$ such that $(i, j) \in \widehat{E}$ with weight $|Q_{ij}|$ if and only if $\hat{y}_{ij} = 1$. Suppose that the graph $\widetilde{\mathcal{G}}(N, \widetilde{E})$ is obtained after eliminating a single edge with smallest weight from every cycle of $\widehat{\mathcal{G}}$. Finally, define $\tilde{y}$ such that $\tilde{y}_{ij} = 1$ if $(i, j) \in \widetilde{E}$, and $\tilde{y}_{ij} = 0$ if $(i, j) \in E \backslash \widetilde{E}$. Next, we show that the above procedure leads to a 2/3-approximation of (31) in general, and a 3/4-approximation for bipartite graphs.

**Theorem 2** *We have* $\frac{p(\tilde{y})}{p^*} \leq \frac{2}{3}$. *Moreover, if* $\mathcal{G}$ *is bipartite, then* $\frac{p(\tilde{y})}{p^*} \leq \frac{3}{4}$.

***Proof*** Recall that $\widehat{\mathcal{G}}$ is a union of paths and cycles. Let $\gamma$ and $\eta$ denote the number of path and cycle components in $\widehat{\mathcal{G}}$, respectively. Moreover, let $\mathcal{P} = \{P_1, P_2, \ldots, P_\gamma\}$ and $\mathcal{C} = \{C_1, C_2, \ldots, C_\eta\}$ denote the set of paths and cycle components in $\widehat{\mathcal{G}}$. Clearly, we have $p(\hat{y}) \leq p^* \leq p(\tilde{y})$, since problem (32) is a relaxation of (31), and $\hat{y}$ is a feasible solution to (31). On the other hand, we have

$$
\begin{aligned}
p(\hat{y}) &= \sum_{P_k \in \mathcal{P}} \sum_{(i,j) \in P_k} |Q_{ij}| + \sum_{C_k \in \mathcal{C}} \sum_{(i,j) \in C_k} |Q_{ij}| \\
&\geq p(\tilde{y}) \\
&= \sum_{P_k \in \mathcal{P}} \sum_{(i,j) \in P_k} |Q_{ij}| + \sum_{C_k \in \mathcal{C}} \left( \left( \sum_{(i,j) \in C_k} |Q_{ij}| \right) - \min_{(i,j) \in C_k} |Q_{ij}| \right) \\
&\geq \sum_{P_k \in \mathcal{P}} \sum_{(i,j) \in P_k} |Q_{ij}| + \frac{2}{3} \sum_{C_k \in \mathcal{C}} \sum_{(i,j) \in C_k} |Q_{ij}| \\
&\geq \frac{2}{3} \left( \sum_{P_k \in \mathcal{P}} \sum_{(i,j) \in P_k} |Q_{ij}| + \sum_{C_k \in \mathcal{C}} \sum_{(i,j) \in C_k} |Q_{ij}| \right) \\
&= \frac{2}{3} p(\hat{y}),
\end{aligned}
$$

where in the second inequality, we used the fact that removing an edge with the smallest weight from a cycle can reduce the weight of that cycle by at most a factor of $\frac{2}{3}$. This implies that $\frac{2}{3} p(\hat{y}) \leq p(\tilde{y}) \leq p^* \leq p(\hat{y})$, and hence, $\frac{p(\tilde{y})}{p^*} \leq \frac{2}{3}$. The last part of the theorem follows since bipartite graphs do not contain cycles of length 3, thus each cycle of $\widehat{\mathcal{G}}$ has length four or more. Therefore, removing an edge with the smallest weight from a cycle of a bipartite graph reduces the weight by at most a factor of $\frac{3}{4}$. □

***Remark 3*** It can be easily shown that the procedure applied to a pure cycle cover of $\mathcal{G}$, including cycles of length 2, would lead to a 1/2-approximation. Thus the proposed method indeed delivers in theory higher quality solutions for bipartite graphs, reducing the optimality gap of the worst-case performance by half. ■

**Table 2** Perfomance solving tridiagonal instances

| Metric | Method | $n = 10$ | $n = 50$ | $n = 100$ | $n = 200$ |
|---|---|---|---|---|---|
| | Algorithm 1 | 3e-4(1e-4) | 2e-3(4e-4) | 7e-3(9e-4) | 4e-2(2e-2) |
| Time(s) | Direct $\mathcal{O}(n^3)$ | 3e-3(8e-4) | 1e-1(5e-3) | 1e+0(6e-2) | 2e+1(5e+0) |
| | Big-M | 4e-2(1e-2) | 4e-1(3e-1) | 7e+2(1e+3) | TL |
| B&B node | Big-M | 1e+1(5e+0) | 3e+3(4e+3) | 7e+6(1e+7) | 2e+7(5e+5) |
| Gap | Big-M | 0.0%(0.0%) | 0.0%(0.0%) | 0.1%(0.1%) | 2.2%(1.0%) |

TL: Time Limit (1 hour)

## 5 Computational Results

We now report illustrative computational experiments showcasing the performance of the proposed methods. First, in Sect. 5.1, we demonstrate the performance of Algorithm 1 on instances with tridiagonal matrices. Then, in Sect. 5.2, we discuss the performance of Algorithm 2 on instances inspired by inference with graphical models.

### 5.1 Experiments with tridiagonal instances

In this section, we consider instances with tridiagonal matrices. We compare the performance of Algorithm 1, the direct $\mathcal{O}(n^3)$ method mentioned in the beginning of Sect. 2.2, and the big-M mixed-integer nonlinear optimization formulation (3), solved using Gurobi v9.0.2. All experiments are run on a Lenovo laptop with a 1.9GHz Intel®Core™ i7-8650U CPU and 16 GB main memory; for Gurobi, we use a single thread and a time limit of one hour, and stop whenever the optimality gap is 1% or less.

In the first set of experiments, we construct tridiagonal matrices $Q \in \mathbb{R}^{N\times N}$ and vectors $a, c \in \mathbb{R}^N$ randomly as $c = \text{Uniform}[-10, 3]$, $a = \text{Uniform}[0, 1]$, $Q_{i,i+1} = \text{Uniform}[-2, 2]$, $Q_{ii} = |Q_{i,i-1}|+|Q_{i,i+1}|+\text{Uniform}[0, 4]$. Table 2 reports the time in seconds required to solve the instances by each method considered, as well as the gap and the number of branch-and-bound nodes reported by Gurobi, for different dimensions $n \leq 200$. Each row represents the average (in parenthesis, the standard deviation) over 10 instances generated with the same parameters.

As expected, mixed-integer optimization approaches struggle in instances with $n = 200$, whereas the polynomial time methods are much faster. Moreover, as expected, Algorithm 1, with worst-case complexity of $\mathcal{O}(n^2)$, is substantially faster than the direct $\mathcal{O}(n^3)$ method. To better illustrate the scalability of the proposed methods, we report in Figure 5 the time used by the polynomial time methods to solve instances with $10 \leq n \leq 10{,}000$. We see that the direct $\mathcal{O}(n^3)$ method requires over 10 minutes to solve instances with $n = 500$, and over one hour for instances with $n \geq 1000$. In contrast, the faster Algorithm 1 can solve instances with $n \leq 1000$ in under one second, and instances with $n = 10{,}000$ in less than one minute. We also see that the practical performance of both methods matches the theoretical complexity.

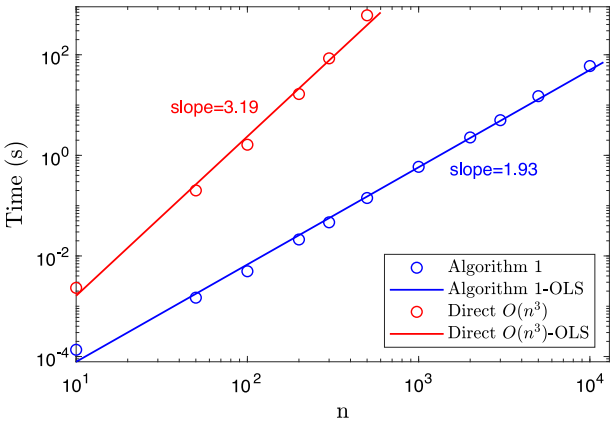


**Fig. 5** Time (in seconds) required to solve tridiagonal problems with $10 \leq n \leq 10{,}000$, in logarithmic scale. Lines in the graph are obtained by fitting a least square regression

**Table 3** Results with one-dimensional graphical models (5), $n = 1000$. We set $a_i = \mu$ for all $i \in N$, and choose $\mu$ so that approximately $\|x\|_0 = 0.1n$ in an optimal solution. For comparison, in [7], the authors report solution times between one and two seconds (in the same instances) when solving a tailored conic quadratic *relaxation* of the problem, which is able to prove optimality gaps in the order of 0.5%

| $\sigma$ | $\mu$ | Algorithm 1 | Big-M | | |
|---|---|---|---|---|---|
| | | Time(s) | Time(s) | Gap | Nodes |
| 0.10 | 0.01 | 0.86 (0.04) | 2440.16 (1318.63) | 1.7% (1.0%) | 3.3e6 (1.9e6) |
| 0.50 | 0.02 | 0.84 (0.03) | TL | 40.9% (4.6%) | 4.5e6 (2.3e5) |
| 1.00 | 0.12 | 0.86 (0.04) | TL | 42.1% (4.0%) | 4.0e6 (4.3e5) |

We also tested Algorithm 1 in inference problems with one-dimensional graphical models of the form (5), which are naturally tridiagonal. For this setting, we use the data $\{y_t\}_{t=1}^n \in \mathbb{R}^n$ with $n = 1000$ used in [7], available online at https://sites.google.com/usc.edu/gomez/data, corresponding to the distribution of noisy observations of a GMRF, as discussed in Sect. 1.1. Instances are classified according to a noise parameter $\sigma$, corresponding to the standard deviation of the noise $\epsilon_i$, see Section 1.1 (all noise terms have the same variance). The results are reported in Table 3. Each row shows the average (in parenthesis, the standard deviation) over 10 instances generated with identical parameters.

Once again, Algorithm 1 is substantially faster than the big-M formulation solved using Gurobi. More interestingly perhaps are how the results reported here compare with those of [7]. In that paper, the authors propose a conic quadratic relaxation of problem[1] (5), and solve this relaxation using the off-the-shelf solver Mosek. The authors report that solving this relaxation requires two seconds in these instances. Note that solution times are not directly comparable due to using different computing environments. Nonetheless, we see that, using Algorithm 1, the mixed-integer optimization

[1] They consider a slightly different term, where the sparsity is imposed via a cardinality constraint $a^\top z \leq k$ instead of a penalization in the objective.

**Table 4** $10 \times 10$ Graphical model, i.e., $n = 100$. We set $a_i = \mu$ for all $i \in N$, and choose $\mu$ so that in an optimal solution, $\|x\|_0$ approximately matches the number of nonzeros of the underlying signal. In all cases, the optimality gap of Algorithm 2 is at most 1%

| $\sigma$ | $\mu$ | Alg 2, $s_k = 1.01^{-k}$ | | Alg 2, $s_k = 1/k$ | | Big-M | | |
|---|---|---|---|---|---|---|---|---|
| | | Iter. | Time(s) | Iter. | Time(s) | Nodes | Time(s) | Gap |
| 0.02 | 0.5 | 8 (7) | 0.14 (0.11) | 9 (2) | 0.17 (0.02) | 7.5e1 (3.9e1) | 0.13 (0.02) | <1.0% |
| 0.1 | 0.5 | 18 (21) | 0.31 (0.34) | 7 (1) | 0.14 (0.02) | 6.7e2 (5.2e2) | 0.35 (0.18) | <1.0% |
| 0.3 | 0.1 | 93 (56) | 1.51 (0.90) | 11 (3) | 0.19 (0.04) | 5.2e6 (6.1e6) | 1012.13 (1242.50) | <1.0% |
| 0.5 | 0.1 | 192 (34) | 3.13 (0.72) | 21 (13) | 0.35 (0.18) | 1.3e7 (6.5e6) | 2759.50 (1384.31) | 5.8% (4.0%) |

problem (5) can be solved *to optimality* in approximately the same time required to solve the convex relaxation proposed in [7]. Moreover, Algorithm 1 can be used with arbitrary tridiagonal matrices $Q \succeq 0$, whereas the method of [7] requires the additional assumption that $Q$ is a Stieltjes matrix.

### 5.2 Inference with two-dimensional graphical models

In the previous section, we reported experiments with tridiagonal matrices, where Algorithm 1 delivers the optimal solution of the mixed-integer problem. In this section, we report our computational experiments with solving inference problems (1) using Algorithm 2 (which is not guaranteed to find an optimal solution) and the big-M formulation. In the considered instances, graph $\mathcal{G}$ is given by the two-dimensional lattice depicted in Figure 1, that is, elements of $N$ are arranged in a grid and there are edges between horizontally/vertically adjacent vertices. We consider instances with grid sizes $10 \times 10$ and $40 \times 40$, thus resulting in instances with $n = 100$ and $n = 1600$, respectively. The data for $y$ are generated similarly to [36], where $\sigma$ is the standard deviation of the noise terms $\epsilon_i$. The data is available online at https://sites.google.com/usc.edu/gomez/data. In these experiments, we execute Algorithm 2 by first permuting the variables according to Section 4.

We test two different step sizes[2] $s_k = 1/k$ and $s_k = (1.01)^{-k}$ for Algorithm 2. For both the big-M formulation and Algorithm 2, we stop whenever the proven optimality gap is less than 1%. Moreover, we also set a time limit of one hour. Tables 4 and 5 report results with $n = 100$ and $n = 1600$, respectively. Here, in all the tested instances, Algorithm 2 yields gaps of less than 1% within the time limit, hence we omit the gaps from the tables. However, for the big-M formulation, the time limit is reached in some of the instances, therefore, tables not only show the time and the number of branch-and-bound nodes explored by Gurobi, but also provide the gaps. Each row shows the average (in parenthesis, the standard deviation) over ten instances generated with identical parameters.

We see that the big-M formulation can be solved fast for low noise values, but struggles in high-noise regimes. For example, if $n = 100$, problems with $\sigma \leq 0.1$ are

[2] For step size $s_k = 1/k$, we modify line 6 of Algorithm 2 to $(\alpha, \beta) \leftarrow (\alpha, \beta) + s_k \rho(\bar{x}, \bar{z})$ (without normalization), since this version performed better in our computations.

**Table 5** $40 \times 40$ Graphical model, i.e., $n = 1600$.We set $a_i = \mu$ for all $i \in N$, and choose $\mu$ so that in an optimal solution, $\|x\|_0$ approximately matches the number of nonzeros of the underlying signal. In all cases, the optimality gap of Algorithm 2 is at most 1%

| $\sigma$ | $\mu$ | Alg 2, $s_k = 1.01^{-k}$ | | Alg 2, $s_k = 1/k$ | | Big-M | | |
|---|---|---|---|---|---|---|---|---|
| | | Iter. | Time(s) | Iter. | Time(s) | Nodes | Time(s) | Gap |
| 0.02 | 0.05 | 9 (2) | 28.8 (6.9) | 14 (1) | 25.5 (17.3) | 5.9e3 (3.7e3) | 25.5 (17.3) | <1.0% |
| 0.1 | 0.05 | 7 (1) | 23.5 (3.3) | 14 (2) | 42.8 (7.1) | 7.1e5 (2.1e4) | TL | 4.2% (0.7%) |
| 0.3 | 0.05 | 53 (33) | 163.0 (99.4) | 18 (6) | 54.9 (16.7) | 7.9e5 (4.1e4) | TL | 24.2% (0.9%) |
| 0.5 | 0.05 | 200 (53) | 609.4 (158.9) | 85 (51) | 260.0 (154.6) | 7.2e5 (5.9e4) | TL | 30.3% (1.7%) |

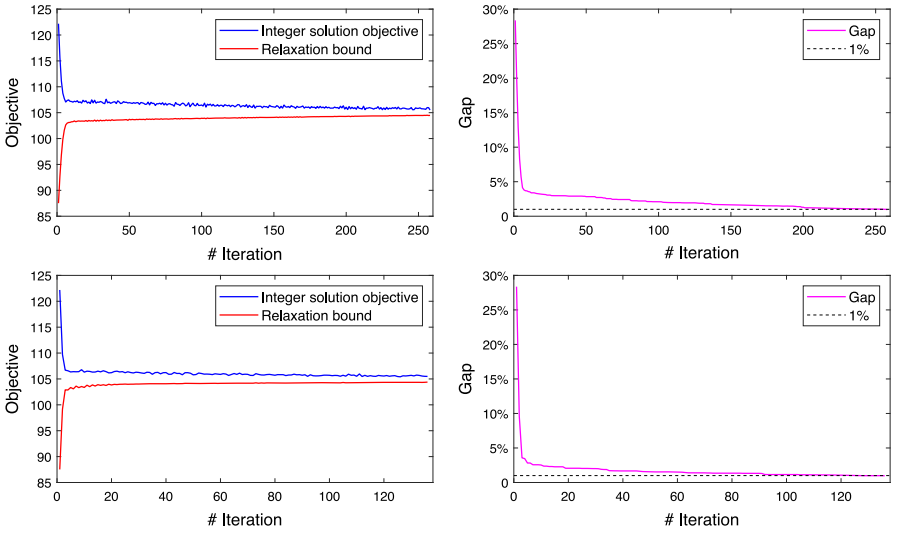


**Fig. 6** The top panel shows, for the instance that requires the most number of iterations, the evolution of Algorithm 2 for step size $s_k = (1.01)^{-k}$, while the bottom panel shows that for step size $s_k = \frac{1}{k}$. Left: Upper and lower bounds found at each iteration of the algorithm. Right: Optimality gap, obtained from best upper/lower bounds found so far

solved in under one second, while problems with $\sigma = 0.5$ sometimes cannot be solved within the time limit. For instances with $n = 1600$, gaps can be as large as 30% in high noise regimes. In contrast, Algorithm 2 consistently delivers solutions with low optimality gaps with running time in seconds on instances with $n = 100$, and in under ten minutes on average on instances with $n = 1600$.

To better understand the evolution of Algorithm 2, we plot the optimality gap as a function of the iteration number in Figs. 6 and 7. In Fig. 6, we present the case which yields 1% gap the slowest among the 10 instances with $n = 1600, \sigma = 0.5$. The evolution of integer solution objective values and relaxation bounds are also shown here. We can see that the integer solution objective values and relaxation bounds are not guaranteed to improve in each iteration, hence the optimality gap, which is computed based on the best lower and upper bounds observed, is not guaranteed to

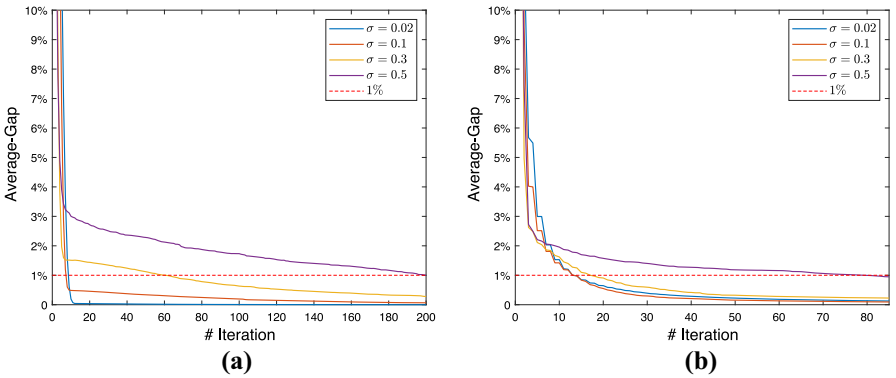


**Fig. 7** Evolution of the average gaps over 10 instances with the iteration number for different noise level $\sigma$. Left: Step size $s_k = (1.01)^{-k}$. Right: Step size $s_k = \frac{1}{k}$

strictly improve in each iteration. However, in general, Algorithm 2 makes substantial progress towards a good lower bound. Moreover, Fig. 7 shows the evolution of the average gap over 10 instances as a function of the iteration number for different noise levels, $\sigma = 0.05, 0.1, 0.3, 0.5$, and both step sizes. We observe that in general, the optimality gap decreases faster in lower noise cases.

In summary, for the instances that are not solved to optimality using the big-M formulation, Algorithm 2 is able to reduce the optimality gaps by at least an order of magnitude while requiring only a small fraction of the computational time.

**Acknowledgements** We thank the AE and the referees whose comments improved this paper.

## References

1. Aktürk, M.S., Atamtürk, A., Gürel, S.: A strong conic quadratic reformulation for machine-job assignment with controllable processing times. Oper. Res. Lett. **37**, 187–191 (2009)
2. Anstreicher, K.M., Burer, S.: Quadratic optimization with switching variables: The convex hull for $n = 2$. Math. Program. **188**, 421–441 (2021)
3. Atamtürk, A., Gómez, A.: Strong formulations for quadratic optimization with M-matrices and indicator variables. Math. Program. **170**, 141–176 (2018)
4. Atamtürk, A., Gómez, A.: Rank-one convexification for sparse regression. arXiv preprint arXiv:1901.10334 (2019)
5. Atamtürk, A., Gómez, A.: Safe screening rules for L0-regression from perspective relaxations. In International Conference on Machine Learning, pages 421–430. PMLR, (2020)
6. Atamtürk, A., Gómez, A.: Supermodularity and valid inequalities for quadratic optimization with indicators. arXiv preprint arXiv:2012.14633, (2020)
7. Atamtürk, A., Gómez, A., Han, S.: Sparse and smooth signal estimation: Convexification of L0-formulations. J. Mach. Learn. Res. **22**(52), 1–43 (2021)
8. Bertsekas, D.P.: Local convex conjugacy and Fenchel duality. IFAC Proceedings Volumes **11**(1), 1079–1084 (1978)
9. Bertsimas, D., King, A., Mazumder, R.: Best subset selection via a modern optimization lens. Ann. Stat. **44**, 813–852 (2016)
10. Besag, J.: Spatial interaction and the statistical analysis of lattice systems. J. Roy. Stat. Soc.: Ser. B (Methodol.) **36**(2), 192–225 (1974)

11. Besag, J., Kooperberg, C.: On conditional and intrinsic autoregressions. Biometrika **82**(4), 733–746 (1995)
12. Besag, J., York, J., Mollié, A.: Bayesian image restoration, with two applications in spatial statistics. Ann. Inst. Stat. Math. **43**(1), 1–20 (1991)
13. Bienstock, D.: Computational study of a family of mixed-integer quadratic programming problems. Math. Program. **74**(2), 121–140 (1996)
14. Boyd, S., Boyd, S.P., Vandenberghe, L.: Convex optimization. Cambridge University Press, Cambridge (2004)
15. Boyd, S., Xiao, L., Mutapcic, A.: Subgradient methods. Lecture notes of EE392o, Stanford University, Autumn Quarter, 2004:2004–2005, (2003)
16. Ceria, S., Soares, J.: Convex programming for disjunctive convex optimization. Math. Program. **86**, 595–614 (1999)
17. Chen, Y., Ge, D., Wang, M., Wang, Z., Ye, Y., Yin, H.: Strong np-hardness for sparse optimization with concave penalty functions. In International Conference on Machine Learning, pages 740–747. PMLR (2017)
18. Cozad, A., Sahinidis, N.V., Miller, D.C.: Learning surrogate models for simulation-based optimization. AIChE J. **60**(6), 2211–2227 (2014)
19. Das, A., Kempe, D.: Algorithms for subset selection in linear regression. In Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing, pages 45–54, (2008)
20. Datta, B.N.: Numerical linear algebra and applications, vol. 116. SIAM, Philadelphia (2010)
21. Davarnia, D., Van Hoeve, W.-J.: Outer approximation for integer nonlinear programs via decision diagrams. Math. Program. **187**(1), 111–150 (2021)
22. Del Pia, A., Dey, S.S., Weismantel, R.: Subset selection in sparse matrices. SIAM J. Optim. **30**(2), 1173–1190 (2020)
23. Eppen, G., Martin, R.: Solving multi-item capacitated lot-sizing problems with variable definition. Oper. Res. **35**(6), 832–848 (1987)
24. Fang, E.X., Liu, H., Wang, M.: Blessing of massive scale: spatial graphical model estimation with a total cardinality constraint approach. Math. Program. **176**(1), 175–205 (2019)
25. Fattahi, S., Gómez, A.: Scalable inference of sparsely-changing Markov random fields with strong statistical guarantees. Forthcoming in NeurIPS, (2021)
26. Frangioni, A., Furini, F., Gentile, C.: Improving the approximated projected perspective reformulation by dual information. Oper. Res. Lett. **45**, 519–524 (2017)
27. Frangioni, A., Gentile, C.: Perspective cuts for a class of convex 0–1 mixed integer programs. Math. Program. **106**, 225–236 (2006)
28. Frangioni, A., Gentile, C., Hungerford, J.: Decompositions of semidefinite matrices and the perspective reformulation of nonseparable quadratic programs. Math. Oper. Res. **45**(1), 15–33 (2020)
29. Gade, D., Küçükyavuz, S.: Formulations for dynamic lot sizing with service levels. Nav. Res. Logist. **60**(2), 87–101 (2013)
30. Garey, M.R., Johnson, D.S.: Computers and intractability, vol. 174. freeman, San Francisco (1979)
31. Geman, S., Graffigne, C.: Markov random field image models and their applications to computer vision. In: Proceedings of the International Congress of Mathematicians, vol. 1, page 2. Berkeley, CA, (1986)
32. Gómez, A.: Outlier detection in time series via mixed-integer conic quadratic optimization. SIAM J. Optim. **31**(3), 1897–1925 (2021)
33. Günlük, O., Linderoth, J.: Perspective reformulations of mixed integer nonlinear programs with indicator variables. Math. Program. **124**, 183–205 (2010)
34. Han, S., Gómez, A., Atamtürk, A.: 2x2 convexifications for convex quadratic optimization with indicator variables. arXiv preprint arXiv:2004.07448, (2020)
35. Hazimeh, H., Mazumder, R., Saab, A.: Sparse regression at scale: Branch-and-bound rooted in first-order optimization. Mathematical Programming, 2021. Article in Advance, https://doi.org/10.1007/s10107-021-01712-4
36. He, Z., Han, S., Gómez, A., Cui, Y., Pang, J.-S.: Comparing solution paths of sparse quadratic minimization with a Stieltjes matrix. Optimization Online: http://www.optimization-online.org/DB_HTML/2021/09/8608.html, (2021)
37. Hochbaum, D.S.: An efficient algorithm for image segmentation, Markov random fields and related problems. Journal of the ACM (JACM) **48**(4), 686–701 (2001)
38. Jeon, H., Linderoth, J., Miller, A.: Quadratic cone cutting surfaces for quadratic programs with on-off constraints. Discret. Optim. **24**, 32–50 (2017)

39. Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem. Proceedings of the American Mathematical Society **7**(1), 48–50 (1956)
40. Küçükyavuz, S., Shojaie, A., Manzour, H., Wei, L.: Consistent second-order conic integer programming for learning Bayesian networks. arXiv preprint arXiv:2005.14346, (2020)
41. Lozano, L., Bergman, D., Smith, J.C.: On the consistent path problem. Operations Resesarch **68**(6), 1913–1931 (2020)
42. Magnanti, T.L., Wolsey, L.A.: Optimal trees. Handbooks Oper. Res. Management Sci. **7**, 503–615 (1995)
43. Manzour, H., Küçükyavuz, S., Wu, H.-H., Shojaie, A.: Integer programming for learning directed acyclic graphs from continuous data. INFORMS Journal on Optimization **3**(1), 46–73 (2021)
44. Mao, X., Qiu, K., Li, T., Gu, Y.: Spatio-temporal signal recovery based on low rank and differential smoothness. IEEE Trans. Signal Process. **66**(23), 6281–6296 (2018)
45. Nesterov, Y.: Primal-dual subgradient methods for convex problems. Math. Program. **120**(1), 221–259 (2009)
46. Nesterov, Y.E.: A method for solving the convex programming problem with convergence rate $O(1/k^2)$. In Doklady Akademii Nauk SSSR **269**, 543–547 (1983)
47. Richard, J.-P.P., Tawarmalani, M.: Lifting inequalities: a framework for generating strong cuts for nonlinear programs. Math. Program. **121**, 61–104 (2010)
48. Saquib, S.S., Bouman, C.A., Sauer, K.: ML parameter estimation for Markov random fields with applications to Bayesian tomography. IEEE Trans. Image Process. **7**(7), 1029–1044 (1998)
49. Sion, M.: On general minimax theorems. Pac. J. Math. **8**(1), 171–176 (1958)
50. Tutte, W.T.: A short proof of the factor theorem for finite graphs. Can. J. Math. **6**, 347–352 (1954)
51. Wei, L., Gómez, A., Küçükyavuz, S.: Ideal formulations for constrained convex optimization problems with indicator variables. Mathematical Programmming **192**(1–2), 57–88 (2022)
52. Wei, L., Gómez, A., Küçükyavuz, S.: On the convexification of constrained quadratic optimization problems with indicator variables. In International Conference on Integer Programming and Combinatorial Optimization, pages 433–447. Springer, (2020)
53. Wolsey, L.A.: Solving multi-item lot-sizing problems with an MIP solver using classification and reformulation. **48**(12), 1587–1602, (2002)
54. Wolsey, L.A.: Integer programming. John Wiley & Sons, Newyork (2020)
55. Wolsey, L.A., Nemhauser, G.L.: Integer and combinatorial optimization. John Wiley & Sons, Newyork (1999)
56. Wu, H., Noé, F.: Maximum a posteriori estimation for Markov chains based on gaussian Markov random fields. Procedia Computer Science **1**(1), 1665–1673 (2010)
57. Xie, W., Deng, X.: Scalable algorithms for the sparse ridge regression. SIAM J. Optim. **30**, 3359–3386 (2020)
58. Ziniel, J., Potter, L.C., Schniter, P.: Tracking and smoothing of time-varying sparse signals via approximate belief propagation. In: 2010 Conference Record of the Forty Fourth Asilomar Conference on Signals, Systems and Computers, pages 808–812. IEEE, (2010)