SIAM J. MATRIX ANAL. APPL.
Vol. 36, No. 4, pp. 1691–1717

DECOMPOSITION METHODS FOR SPARSE MATRIX NEARNESS PROBLEMS*

YIFAN SUN† AND LIEVEN VANDENBERGHE†

Abstract. We discuss three types of sparse matrix nearness problems: given a sparse symmetric matrix, find the matrix with the same sparsity pattern that is closest to it in Frobenius norm and (1) is positive semidefinite, (2) has a positive semidefinite completion, or (3) has a Euclidean distance matrix completion. Several proximal splitting and decomposition algorithms for these problems are presented and their performance is compared on a set of test problems. A key feature of the methods is that they involve a series of projections on small dense positive semidefinite or Euclidean distance matrix cones, corresponding to the cliques in a triangulation of the sparsity graph. The methods discussed include the dual block coordinate ascent algorithm (or Dykstra's method), the dual projected gradient and accelerated projected gradient algorithms, and a primal and a dual application of the Douglas–Rachford splitting algorithm.

Key words. positive semidefinite completion, Euclidean distance matrices, chordal graphs, convex optimization, first order methods

AMS subject classifications. 65F50, 90C22, 90C25

DOI. 10.1137/15M1011020

1. Introduction.

We discuss matrix optimization problems of the form

$$\begin{array}{ll} \text{minimize} & \|X - C\|_F^2 \\ \text{subject to} & X \in \mathcal{S}, \end{array} \tag{1.1}$$

where the variable X is a symmetric $p \times p$ matrix with a given sparsity pattern E, and C is a given symmetric matrix with the same sparsity pattern as X. We focus on the following three choices for the constraint set $\mathcal{S}$.

I. $\mathcal{S}$ is the set of sparse positive semidefinite (PSD) matrices with sparsity pattern E:

$$\mathcal{S} = \mathbf{S}_E^p \cap \mathbf{S}_+^p.$$

Here $\mathbf{S}_E^p$ denotes the set of symmetric $p \times p$ matrices with sparsity pattern E. The set $\mathbf{S}_+^p = \{X \in \mathbf{S}^p \mid X \succeq 0\}$ is the set of $p \times p$ PSD matrices. In this problem we approximate a matrix C by the nearest PSD matrix with sparsity pattern E.

II. $\mathcal{S}$ is the set of sparse matrices with sparsity pattern E that have a PSD completion:

$$\mathcal{S} = \Pi_E(\mathbf{S}_+^p) = \{\Pi_E(S) \mid S \in \mathbf{S}_+^p\}.$$

Here, $\Pi_E(S)$ denotes Euclidean projection on $\mathbf{S}_E^p$. In applications of this problem, the nonzeros in C are observations of a subset of the entries of a PSD matrix. The PSD matrices that are closest to C in a least squares sense are the PSD completions of X, the solution of the problem (1.1) using this choice of $\mathcal{S}$. These completions S are dense matrices in general and the sparse

*Received by the editors March 4, 2015; accepted for publication (in revised form) by M. P. Friedlander September 29, 2015; published electronically December 8, 2015.
http://www.siam.org/journals/simax/36-4/M101102.html

†Electrical Engineering Department, UCLA, Los Angeles, CA 90095 (ysun01@ucla.edu, vandenbe@ucla.edu). The research of the first author was partially supppprted by NSF Graduate Research Fellowship. The research of the second author was partially supported by NSF grants DMS-1115963 and ECCS-1128817.

matrix X is useful as a compact representation from which a full completion can be computed if needed.

III. $\mathcal{S}$ is the set of sparse matrices with sparsity pattern E that have a Euclidean distance matrix (EDM) completion:

$$\mathcal{S} = \Pi_E(\mathbf{D}^p).$$

Here, $\mathbf{D}^p$ is the set of $p \times p$ EDMs, i.e., matrices with entries that can be expressed as squared pairwise Euclidean distances between pairs of points. The interpretation of this problem is similar to problem II, but with the EDM property replacing positive semidefiniteness.

Note that the assumption that C is sparse can be made without loss of generality. If C is dense or not symmetric, it can be replaced by $(1/2)\Pi_E(C + C^T)$ in the objective function of (1.1).

We also discuss the following generalization of problems II and III:

$$\begin{array}{ll} \text{minimize} & \|\Pi_E(X) - C\|_F^2 \\ \text{subject to} & X \in \mathcal{S} \end{array} \tag{1.2}$$

with variable $X \in \mathbf{S}^p_{E'}$ where E' is an extension of E (i.e., the nonzero positions in E are a subset of the nonzeros in E') and $\mathcal{S} = \Pi_{E'}(\mathbf{S}^p_+)$ or $\mathcal{S} = \Pi_{E'}(\mathbf{D}^p)$. In the extended problem (1.2) we approximate C by a sparse matrix $\Pi_E(X)$ that has a PSD or EDM completion, and we compute some of the entries (indexed by $E' \setminus E$) of a completion of $\Pi_E(X)$.

The techniques presented in the paper are easily extended to similar problems with simple additional constraints on X (for example, problems I and II with a constraint $\mathrm{diag}(X) = \mathbf{1}$). The methods we discuss for problem (1.2) also apply to weighted Frobenius norms, i.e., the minimization of functions $\sum_{\{i,j\}\in E'} W_{ij}(X_{ij} - C_{ij})^2$ with nonnegative weights W_{ij}.

1.1. Matrix nearness problems. Problem (1.1) belongs to the general class of *matrix nearness* problems [38]. In a matrix nearness problem one projects a given matrix on the set of matrices that satisfy a certain property. Problems of this type arise in a wide range of applications, including statistics, machine learning, finance, signal processing, control, and computational biology, and can take a variety of different forms. The matrix C may be symmetric or nonsymmetric, square or rectangular. The constraint set may be convex or not, and the objective function may be a Frobenius norm or another norm, or a nonmetric distance function. In this paper, we restrict our attention to problems involving symmetric matrix variables and convex constraints.

Matrix nearness problems with PSD and EDM constraints are among the most widely studied types of matrix nearness problems. The nearest PSD matrix and nearest correlation matrix problems have applications in statistics, finance, and biology, and are discussed in [2, 39, 42, 47, 48, 52]. The nearest EDM problem is studied in multidimensional scaling techniques (see [66] for an overview), and is used in chemistry and molecular biology to compute molecular structure [6, 29, 59, 64], and for node localization in sensor networks [3, 40].

These applications all involve fairly simple convex sets and, in principle, they can be solved by general-purpose convex optimization algorithms. However, due to the large number of variables (order p^2 for dense matrix variables) the complexity of general-purpose methods grows rapidly with the matrix dimension p. Research

on matrix nearness problems has therefore concentrated on specialized first-order algorithms, or, when higher accuracy is required, quasi-Newton algorithms. Examples of such first-order methods are alternating projection methods [2, 28, 29, 39], dual first-order methods [17, 36], augmented Lagrangian methods [49], and alternating direction methods [16]. Algorithms based on quasi-Newton methods in combination with dual reformulations are described in [42, 47, 51].

In comparison, algorithms tailored to *sparse* matrix nearness problems, i.e., problems with an additional sparsity constraint, have received less attention in the literature. Sparsity constraints arise naturally in large matrix nearness problems and can have a different meaning depending on the application and the interpretation of the data matrix C. In a first type of problem, we are interested in recovering a sparse matrix with the properties represented as a set $\mathcal{S}$, from a noisy measurement or estimate of its nonzero elements. Most applications of problem I are of this type. A typical example is the estimation of a sparse covariance or inverse covariance matrix from estimates of its nonzero entries. In other situations, the sparse matrix C represents a noisy estimate of a subset of the entries of a dense matrix, which is known to have the structural properties represented by a set $\hat{\mathcal{S}}$. In this case the matrices in $\hat{\mathcal{S}}$ that best fit the measurements are dense. When the matrix dimensions are large it is of interest to avoid working with a dense matrix variable and represent it implicitly as a sparse matrix, with the added constraint that the matrix has a completion with the desired properties. The sparse partial matrix variable is therefore constrained to lie in the set $\mathcal{S} = \Pi_E(\hat{\mathcal{S}})$. Problems II and III are motivated by applications of this kind, with $\hat{\mathcal{S}}$ the set of PSD matrices or EDMs, respectively. In some applications, partial knowledge of the solution (in the form of a sparse completable matrix) is sufficient. If the full dense matrix is desired, it can be constructed from the partial solution by matrix completion algorithms [7, 31].

Most algorithms for sparse matrix nearness problems in the literature are extensions of algorithms for dense matrix nearness problems. In problem I one can use a dense matrix variable and impose the sparsity constraint by adding linear equality constraints. Similarly, problems II and III can be solved by using a dense matrix variable and masking irrelevant entries in the objective [4, 5, 39, 49, 50]. Extensions of this type are still computationally expensive. In particular, they involve eigenvalue decompositions of order p, used for projecting on the set of PSD matrices or the set of EDMs. In contrast, the approach taken in this paper is to avoid eigenvalue decompositions of order p and only use eigenvalue decompositions of smaller dense matrices. By applying decomposition techniques for cones of sparse matrices with chordal structure, we write the three types of sparse matrix cones in terms of small dense PSD or EDM cones [1, 7, 30, 31]. In combination with first-order methods, these chordal decomposition techniques allow us to solve the matrix nearness problems without using eigenvalue decompositions of order p.

1.2. Outline and notation. The rest of the paper is organized as follows. In section 2, we discuss some fundamental geometrical properties of cones with partially separable structure. In section 3 we present decomposition results for the three sparse matrix nearness problems. We formulate the problems as Euclidean projections on convex cones with partially separable structure. Section 4 presents two types of dual algorithms that apply when E is a chordal sparsity pattern. In section 5 we discuss extensions that apply when the pattern E is not necessarily chordal. These extensions also handle the more general problem (1.2). Section 6 gives numerical results with test problems that range from size $p \sim 1000$ to 100000.

Notation. If β is an index set (ordered subset of $\{1,\ldots,p\}$), we define P_β as the $|\beta| \times p$-matrix

$$(P_\beta)_{ij} = \begin{cases} 1, & j = \beta(i), \\ 0, & \text{otherwise.} \end{cases}$$

Multiplying a vector with P_β selects the subvector indexed by β:

$$P_\beta x = (x_{\beta(1)}, x_{\beta(2)}, \ldots, x_{\beta(r)}),$$

where β has r elements, denoted $\beta(1), \beta(2), \ldots, \beta(r)$. Similarly, the matrix P_β can be used to select a principal submatrix of a $p \times p$ matrix:

$$(P_\beta X P_\beta^T)_{ij} = X_{\beta(i)\beta(j)}.$$

The multiplication $x = P_\beta^T y$ of a $|\beta|$-vector y with the transpose of P_β gives a p-vector x with $P_\beta x = y$ and $x_j = 0$ for $j \notin \beta$. The operation $X = P_\beta^T Y P_\beta$ creates a $p \times p$ matrix X from an $r \times r$ matrix, with $P_\beta X P_\beta^T = Y$ and $X_{ij} = 0$ if $i \notin \beta$ or $j \notin \beta$. For example, if $p = 5$ and $\beta = \{1, 3, 4\}$, then

$$P_\beta = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad P_\beta^T y = \begin{bmatrix} y_1 \\ 0 \\ y_2 \\ y_3 \\ 0 \end{bmatrix}, \quad P_\beta^T Y P_\beta = \begin{bmatrix} Y_{11} & 0 & Y_{12} & Y_{13} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ Y_{21} & 0 & Y_{22} & Y_{33} & 0 \\ Y_{31} & 0 & Y_{32} & Y_{33} & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

We use the notation $\delta_{\mathcal{S}}(x)$ for the indicator function of a set $\mathcal{S}$:

$$\delta_{\mathcal{S}}(x) = \begin{cases} 0, & x \in \mathcal{S}, \\ +\infty, & \text{otherwise.} \end{cases}$$

The notation $\Pi_{\mathcal{S}}$ is used for the Euclidean projection operator on a closed set $\mathcal{S}$.

2. Partially separable convex cones. In this section we discuss the standard convex optimization form to which the matrix nearness problems will be reduced and work out some basic results from convex duality.

2.1. Projection on partially separable cone. The general problem we consider is

$$\begin{array}{ll} \text{minimize} & \|P_\eta x - a\|^2 \\ \text{subject to} & P_{\gamma_k} x \in \mathcal{C}_k, \quad k = 1, \ldots, l, \end{array} \tag{2.1}$$

where $\|\cdot\|$ denotes the Euclidean norm. The optimization variable is an n-vector x. The sets $\eta, \gamma_1, \ldots, \gamma_l$ are index sets (ordered subsets of $\{1, 2, \ldots, n\}$) and $\mathcal{C}_k, k = 1, \ldots, l$, are closed convex cones. We will use the notation

$$\mathcal{K} = \{x \in \mathbb{R}^n \mid P_{\gamma_k} x \in \mathcal{C}_k,\ k = 1, \ldots, l\}$$

for the feasible set in (2.1). This is a closed convex cone and equal to the intersection $\mathcal{K} = \bigcap_{k=1}^l \mathcal{K}_k$ of the closed convex cones

$$\mathcal{K}_k = \{x \in \mathbb{R}^n \mid P_{\gamma_k} x \in \mathcal{C}_k\}, \quad k = 1, \ldots, l.$$

The objective in (2.1) is the squared distance between the subvector $P_\eta x = x_\eta$ and a

given $|\eta|$-vector a:

$$\|P_\eta x - a\|^2 = \sum_{k=1}^{|\eta|} (x_{\eta(k)} - a_k)^2.$$

An important special case will be $\eta = \{1, 2, \ldots, n\}$. Then $P_\eta = I$ and the problem has a unique solution $x^\star$, the projection of a on the closed convex cone $\mathcal{K}$.

When $\eta \neq \{1, 2, \ldots, n\}$ the problem is to project a on $P_\eta\mathcal{K}$, the projection of $\mathcal{K}$ on the subspace of vectors with support η. In order to ensure that $P_\eta\mathcal{K}$ is a closed set, we assume that $P_\eta x = 0$, $x \in \mathcal{K}$, only holds for $x = 0$. This assumption is satisfied in the applications discussed in this paper (section 3). If the assumption holds, closedness of $P_\eta\mathcal{K}$ follows from Theorem 9.1 in [53], which states that if $\mathcal{K}$ is a closed convex cone and has a trivial intersection with the nullspace of A, then $A\mathcal{K}$ is a closed convex cone. Since $P_\eta\mathcal{K}$ is closed, problem (2.1) has a unique optimal solution for the subvector $P_\eta x^\star$. However, the components of $x^\star$ outside η are not necessarily unique.

In the problems of interest the cones $\mathcal{C}_k$ will be subspaces or closed convex cones with nonempty interior. We will assume that the problem (2.1) is strictly feasible, i.e., there exists an $\bar{x}$ with $P_{\gamma_k}\bar{x} \in \mathrm{ri}\,\mathcal{C}_k$ for $k = 1, \ldots, l$, where ri stands for relative interior. This assumption is satisfied in the applications we discuss, and will be needed in section 2.2.

A key property of problem (2.1) is the *partial separability* of the cone $\mathcal{K}$, i.e., the indicator function $\sum_{k=1}^{l} \delta_{\mathcal{C}_k}(P_{\gamma_k} x)$ of $\mathcal{K}$ is a partially separable function [30]. Partial separability with small index sets ($|\gamma_k| \ll n$) is a very useful type of structure for decomposition algorithms in conic optimization [58]. It is crucial to the efficiency of the algorithms presented in this paper and allows us to solve problems of the form (2.1) via a sequence of projections on the lower-dimensional cones $\mathcal{C}_k$.

2.2. Conic duality. The Lagrange dual of problem (2.1) is

$$\begin{array}{ll} \text{maximize} & -\|s + a\|^2 + \|a\|^2 \\ \text{subject to} & P_\eta^T s = \displaystyle\sum_{k=1}^{l} P_{\gamma_k}^T z_k, \\ & z_k \in \mathcal{C}_k^*, \quad k = 1, \ldots, l. \end{array} \tag{2.2}$$

The variables are $s \in \mathbb{R}^{|\eta|}$ and multipliers $z_k \in \mathbb{R}^{|\gamma_k|}$ for the l constraints in (2.1). The sets $\mathcal{C}_k^* = \{z_k \mid s_k^T z_k \geq 0 \ \forall s_k \in \mathcal{C}_k\}$ are the dual cones of $\mathcal{C}_k$. If $\mathcal{C}_k$ is a subspace, then $\mathcal{C}_k^*$ is the orthogonal complement of the subspace. Projections on $\mathcal{C}_k^*$ can be computed via projections on $\mathcal{C}_k$ and vice versa, via the formulas

$$y = \Pi_{\mathcal{C}_k}(y) + \Pi_{-\mathcal{C}_k^*}(y) = \Pi_{\mathcal{C}_k}(y) - \Pi_{\mathcal{C}_k^*}(-y).$$

(This is a special case, with $f = \delta_{\mathcal{C}_k}$, of the *Moreau decomposition* $y = \mathrm{prox}_f(y) + \mathrm{prox}_{f^*}(y)$, which relates the proximal operators of a function f and its conjugate [43].)

The variable s in (2.2) can be eliminated by noting that $(P_\eta^T s)_i = 0$ for $i \notin \eta$, and therefore $\|s + a\| = \|P_\eta^T(s + a)\|$. This gives an equivalent form of the dual problem

$$\begin{array}{ll} \text{maximize} & -\|\displaystyle\sum_{k=1}^{l} P_{\gamma_k}^T z_k + P_\eta^T a\|^2 + \|a\|^2 \\ \text{subject to} & \left(\displaystyle\sum_{k=1}^{l} P_{\gamma_k}^T z_k\right)_i = 0, \quad i \notin \eta, \\ & z_k \in \mathcal{C}_k^*, \quad k = 1, \ldots, l. \end{array} \tag{2.3}$$

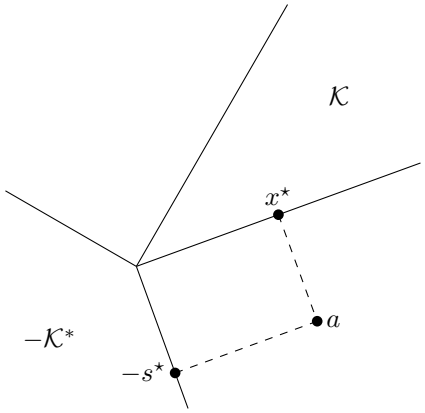

FIG. 1. *Projection on a convex cone $\mathcal{K}$ and on the polar cone $-\mathcal{K}^*$.*

The dual problem has a simple geometric interpretation. First assume that $\eta = \{1, 2, \ldots, n\}$ and $P_\eta = I$. Recall our assumption that there exists a point $\bar{x}$ in the intersection of the sets $\operatorname{ri}\mathcal{K}_k$, $k = 1, \ldots, l$. The assumption implies that the dual cone of $\mathcal{K}$ is $\mathcal{K}^* = \mathcal{K}_1^* + \cdots + \mathcal{K}_l^*$ [53, Corollary 16.4.2]. Here $\mathcal{K}_k^* = \{P_{\gamma_k}^T z_k \mid z_k \in \mathcal{C}_k^*\}$, so

$$\mathcal{K}^* = \left\{ \sum_{k=1}^{l} P_{\gamma_k}^T z_k \mid z_k \in \mathcal{C}_k^*,\ k = 1, \ldots, l \right\}.$$

The solution s of problem (2.2) is the projection of $-a$ on $\mathcal{K}^*$. (Equivalently, $-s$ is the projection of a on the polar cone $-\mathcal{K}^*$.) The solutions $x^\star$ and $s^\star$ of the two projection problems

$$\begin{array}{lll} \text{minimize} & \|x - a\|^2 & \qquad \text{maximize} \quad -\|s + a\|^2 + \|a\|^2 \\ \text{subject to} & x \in \mathcal{K}, & \qquad \text{subject to} \quad s \in \mathcal{K}^* \end{array}$$

are unique and related by the optimality conditions

$$a = x^\star - s^\star, \qquad x^\star \in \mathcal{K}, \qquad s^\star \in \mathcal{K}^*, \qquad {s^\star}^T x^\star = 0.$$

This shows one can solve either the primal or the dual problem and compute the other solution from the relation $a = x^\star - s^\star$. The geometry is illustrated in Figure 1. Note that although $x^\star$ and $s^\star$ are unique, the dual problem (2.3) is not strictly convex in the variables $z_1, \ldots, z_l$, so the optimal solution of this problem is not necessarily unique. (In the dual decomposition algorithms discussed in section 4, we also use the relation $a = x - s$ to recover an approximate primal solution $x = s + a$ from a nonoptimal dual feasible s. Clearly, convergence bounds on $s - s^\star$ also apply to $x - x^\star = s - s^\star$.)

When $\eta \neq \{1, 2, \ldots, n\}$, we can interpret the problem (2.2) in a similar way. Its solution $s^\star$ is the projection of $-a$ on the dual cone of $P_\eta\mathcal{K}$, which is given by $(P_\eta\mathcal{K})^* = \{s \mid P_\eta^T s \in \mathcal{K}^*\}$. The optimality conditions that relate the two projections are

$$a = P_\eta x^\star - s^\star, \qquad x^\star \in \mathcal{K}, \qquad P_\eta^T s^\star \in \mathcal{K}^*, \qquad {s^\star}^T P_\eta x^\star = 0.$$

Note that here the primal and dual approaches are not completely equivalent. From the solution $x^\star$ of the primal problem (2.1) one obtains the dual solution $s^\star = P_\eta x^\star - a$. However from the dual solution $s^\star$ one only finds a partial primal solution $P_\eta x^\star$ and not the values $x_i^\star$ for $i \notin \eta$.

3. Decomposition of sparse matrix nearness problems. In sections 3.2–3.4 we review three fundamental decomposition results for classes of matrices with chordal sparsity patterns, and apply them in order to reformulate the matrix nearness problems listed in the introduction in the standard forms (2.1) and (2.2). We first give some background on chordal sparsity.

3.1. Chordal sparsity patterns. A symmetric sparsity pattern E of order p is defined as a set of index pairs

$$E \subseteq \{\{i,j\} \mid i,j \in \{1,\ldots,p\},\ i \neq j\}.$$

A symmetric $p \times p$ matrix X is said to have sparsity pattern E if $X_{ij} = 0$ for all $\{i,j\} \notin E$ and $i \neq j$. The diagonal entries X_{ii} and the off-diagonal entries X_{ij} for $\{i,j\} \in E$ may or may not be zero. The set of symmetric $p \times p$ matrices with sparsity pattern E is denoted $\mathbf{S}^p_E$.

With every sparsity pattern E we can associate an undirected graph $\mathcal{G}_E$ with vertex set $\{1,\ldots,p\}$ and edge set E. A *clique* (also called *maximal clique*) of a graph $\mathcal{G}_E$ is a maximal set of vertices that are all pairwise adjacent. A clique β in $\mathcal{G}_E$ corresponds to a dense principal submatrix $X_{\beta\beta}$ of a matrix $X \in \mathbf{S}^p_E$. (Here, "dense" means that all entries may be nonzero.)

The sparsity pattern E is *chordal* if the graph $\mathcal{G}_E$ is chordal, i.e., all cycles in $\mathcal{G}_E$ with more than three vertices have a chord (an edge between nonconsecutive vertices in the cycle). Chordal graphs are important in linear algebra because every positive definite matrix X with a chordal sparsity pattern has a zero-fill Cholesky factorization, i.e., a factorization $P_\sigma X P_\sigma^T = LL^T$, where P_σ is a permutation matrix, L is lower triangular, and $P_\sigma^T(L+L^T)P_\sigma \in \mathbf{S}^p_E$ [54]. The reordering by the permutation matrix P_σ is called a *perfect elimination ordering* for E. A chordal graph with p vertices has at most p cliques and they are easily identified from a perfect elimination ordering; see [15] for a survey on chordal graphs.

A sparsity pattern E' with $E \subset E'$ is called an *extension* of the sparsity pattern E. The graph $\mathcal{G}_{E'}$ is the graph $\mathcal{G}_E$ with additional edges $E' \setminus E$. Clearly, $\mathbf{S}^p_E \subset \mathbf{S}^p_{E'}$. The extension is a *chordal* extension if $\mathcal{G}_{E'}$ is a chordal graph. A chordal extension is also called a *triangulation*. The problem of computing the minimum chordal extension (i.e., with minimum $|E'| - |E|$) is NP-complete [65]. The problem of computing a *minimal* chordal extension, i.e., a chordal extension with the property that there exists no chordal extension $\hat{E} \subset E'$, is much more tractable and several algorithms for it have been developed; see [35] for a survey. Chordal extensions can also be computed from a *fill-reducing ordering*, such as the (approximate) minimum degree reordering [23]. Although such extensions are not guaranteed to be minimal, they are often smaller than extensions computed by minimal ordering algorithms.

It will be convenient to represent sparse or dense symmetric matrices as vectors. For a dense matrix $A \in \mathbf{S}^p$, we define $\mathbf{vec}(A)$ as the $p(p+1)/2$-vector containing the elements of the lower-triangular part of A and with off-diagonal entries scaled by $\sqrt{2}$, so that $\mathbf{tr}(AB) = \mathbf{vec}(A)^T\mathbf{vec}(B)$ for all $A, B \in \mathbf{S}^p$. Similarily, for a sparse matrix $A \in \mathbf{S}^p_E$, we define $\mathbf{vec}_E(A)$ as an $(|E|+p)$-vector containing the diagonal elements and the lower-triangular elements A_{ij} for $\{i,j\} \in E$, again with the off-diagonal elements scaled by $\sqrt{2}$, so that $\mathbf{tr}(AB) = \mathbf{vec}_E(A)^T\mathbf{vec}_E(B)$ for all $A, B \in \mathbf{S}^p_E$.

3.2. PSD matrices. We now consider the first problem from the introduction: the projection of a sparse matrix $A \in \mathbf{S}^p_E$ on the set of PSD matrices in $\mathbf{S}^p_E$,

$$\mathcal{S} = \mathbf{S}^p_E \cap \mathbf{S}^p_+.$$

This is a closed convex cone (for any E). The following theorem from [1, 30] provides a characterization when the sparsity pattern is chordal.

THEOREM 3.1. *Let E' be a chordal sparsity pattern of order p, with cliques β_k, $k = 1, \ldots, m$. A matrix $S \in \mathbf{S}^p_{E'}$ is PSD if and only if it can be expressed as*

$$S = \sum_{k=1}^{m} P_{\beta_k}^T Z_k P_{\beta_k}$$

with $Z_k \succeq 0$ for $k = 1, \ldots, m$.

When E' is not chordal, the condition in the theorem is sufficient for positive semidefiniteness of S, but not necessary.

Using Theorem 3.1, we can formulate the problem of projecting a matrix $A \in \mathbf{S}^p_E$ on $\mathbf{S}^p_E \cap \mathbf{S}^p_+$, for a general sparsity pattern E, as

$$\begin{array}{ll} \text{minimize} & \left\| \sum_{k=1}^{m} P_{\beta_k}^T Z_k P_{\beta_k} - A \right\|_F^2 \\ \text{subject to} & \left(\sum_{k=1}^{m} P_{\beta_k}^T Z_k P_{\beta_k} \right)_{ij} = 0, \quad \{i,j\} \in E' \setminus E, \\ & Z_k \succeq 0, \quad k = 1, \ldots, m. \end{array} \tag{3.1}$$

Here E' is a chordal extension of E and $\beta_1, \ldots, \beta_m$ are the cliques of $\mathcal{G}_{E'}$. The variables $Z_k \in \mathbf{S}^{|\beta_k|}$ are used to parameterize a general matrix $S = \sum_k P_{\beta_k}^T Z_k P_{\beta_k}$ in $\mathbf{S}^p_{E'} \cap \mathbf{S}^p_+$, according to Theorem 3.1. The equality constraints force the off-diagonal elements S_{ij} with $\{i,j\} \in E' \setminus E$ to be zero. If E is already chordal, we take $E' = E$ and omit the equality constraints. The size of the optimization problem (3.1) depends on the efficiency of the chordal extension: the sizes of the cliques of $\mathcal{G}_{E'}$ determine the dimensions of the variables Z_k and the amount of fill $E' \setminus E$ determines the number of equality constraints. Note that the problem (3.1) is strictly feasible, since $Z_k = I$, $k = 1, \ldots, m$, is strictly feasible.

Problem (3.1) can be written in vector notation in the form (2.3). We define the variables $z_k = \mathbf{vec}(Z_k)$, $k = 1, \ldots, l$. We take $l = m$, $n = p + |E'|$, and $a = -\mathbf{vec}_E(A)$. The index sets η and γ_k are defined by imposing that the identities

$$P_\eta \mathbf{vec}_{E'}(X) = \mathbf{vec}_E(X), \qquad P_{\gamma_k} \mathbf{vec}_{E'}(X) = \mathbf{vec}(P_{\beta_k} X P_{\beta_k}^T), \quad k = 1, \ldots, l, \tag{3.2}$$

hold for all $X \in \mathbf{S}^p_{E'}$. The cone $\mathcal{C}_k^*$ is the cone of dense $|\beta_k| \times |\beta_k|$ PSD matrices converted to vectors:

$$\mathcal{C}_k^* = \{\mathbf{vec}(U) \mid U \in \mathbf{S}_+^{|\beta_k|}\}, \quad k = 1, \ldots, m.$$

For later purposes we note that the Euclidean projection on $\mathcal{C}_k^*$ can be computed by an eigenvalue decomposition: if $U \in \mathbf{S}^{|\beta_k|}$ has eigenvalue decomposition $U = \sum_i \lambda_i q_i q_i^T$ then $\Pi_{\mathcal{C}_k^*}(\mathbf{vec}(U)) = \sum_i \max\{\lambda_i, 0\} \mathbf{vec}(q_i q_i^T)$.

3.3. Matrices with PSD completions. Next we consider the projection of a sparse matrix on the cone of sparse symmetric matrices that have a PSD completion.

This set is the projection of the cone $\mathbf{S}_+^p$ on $\mathbf{S}_E^p$,

$$\Pi_E(\mathbf{S}_+^p) = \{\Pi_E(X) \mid X \in \mathbf{S}_+^p\}.$$

The projection $\Pi_E(\mathbf{S}_+^p)$ is a closed convex cone, for the following reason. If $\Pi_E(X) = 0$ then the diagonal of X is zero. Therefore $\Pi_E(X) = 0$ and $X \in \mathbf{S}_+^p$ imply $X = 0$. This is a sufficient condition for the projection $\Pi_E(\mathbf{S}_+^p)$ to be closed [53, Theorem 9.1]; see also [25, Theorem 3.1].

The following theorem is the dual counterpart of Theorem 3.1 and gives a characterization of $\Pi_{E'}(\mathbf{S}_+^p)$ when the sparsity pattern E' is chordal [31].

THEOREM 3.2. *Let E' be a chordal sparsity pattern of order p, with cliques β_k, $k = 1, \ldots, m$. A matrix $X \in \mathbf{S}_{E'}^p$ is in $\Pi_{E'}(\mathbf{S}_+^p)$ if and only if*

$$P_{\beta_k} X P_{\beta_k}^T \succeq 0, \quad k = 1, \ldots, m.$$

If E' is not chordal, the condition in the theorem is necessary for $X \in \Pi_{E'}(\mathbf{S}_+^p)$ but not sufficient.

Using Theorem 3.2, the problem of projecting a matrix $B \in \mathbf{S}_E^p$ on $\Pi_E(\mathbf{S}_+^p)$ can be written as

$$\begin{array}{ll} \text{minimize} & \|\Pi_E(X) - B\|_F^2 \\ \text{subject to} & P_{\beta_k} X P_{\beta_k}^T \succeq 0, \quad k = 1, \ldots, m. \end{array} \tag{3.3}$$

The variable is $X \in \mathbf{S}_{E'}^p$, where E' is a chordal extension of E, and $\beta_1, \ldots, \beta_m$ are the cliques of E'. When $E' = E$, the projection Π_E is the identity operator and we replace $\Pi_E(X)$ by X in the objective. We note that problem (3.3) is strictly feasible, since $X = I$ is strictly feasible.

Problem (3.3) can be put in the form (2.1) with variable $x = \mathbf{vec}_{E'}(X)$. The problem dimensions are $n = p + |E'|$ and $l = m$, and the vector $a = \mathbf{vec}_E(B)$. The index sets η, $\gamma_1, \ldots, \gamma_l$ are defined as in the previous section. Since the dense PSD cone is self-dual, the cones $\mathcal{C}_k$ are equal to $\mathcal{C}_k^*$.

If $B = -A$, problems (3.1) and (3.3) are duals, and special cases of the primal and dual pair (2.1) and (2.2). The optimal solutions $S^\star = \sum_k P_{\beta_k}^T Z_k^\star P_{\beta_k}$ of (3.1) and $X^\star$ of (3.3) are related as

$$S^\star = \Pi_E(X^\star) + A.$$

This shows that if $E' = E$, the two problems are equivalent (with $B = -A$), since $S^\star = X^\star + A$. When $E \subset E'$, one can compute the solution $S^\star$ from the solution $X^\star$. However, from $S^\star$, one only obtains a partial solution $\Pi_E(X^\star)$ of (3.3).

3.4. Matrices with an EDM completion. A matrix $X \in \mathbf{S}^p$ is an EDM if its elements can be expressed as $X_{ij} = \|u_i - u_j\|^2$ for some set of points u_k, $k = 1, \ldots, p$. Schoenberg's condition [55] states that X is an EDM if and only if

$$\mathrm{diag}(X) = 0 \quad \text{and} \quad c^T X c \leq 0 \text{ for all } c \text{ with } \mathbf{1}^T c = 0, \tag{3.4}$$

where $\mathbf{1}$ represents the p-vector with all elements equal to one. From this it is clear that the set of EDMs is the intersection of closed cones, and therefore is a closed

convex cone. The notation $\mathbf{D}^p$ will be used for the set of $p \times p$ EDMs, and $\mathbf{D}_0^p$ for the matrices that satisfy only the second condition in (3.4):

$$\begin{aligned}\mathbf{D}_0^p &= \{X \in \mathbf{S}^p \mid c^T X c \leq 0 \text{ for all } c \text{ with } \mathbf{1}^T c = 0\} \\ &= \{X \in \mathbf{S}^p \mid V^T X V \preceq 0\},\end{aligned}$$

where V is any $p \times (p-1)$ matrix whose columns span the orthogonal complement of the vector $\mathbf{1}$ in $\mathbb{R}^p$. The EDMs are the matrices in $\mathbf{D}_0^p$ with zero diagonal. The cones $\mathbf{D}_0^p$ and $\mathbf{D}^p$ are closed, convex cones, with two noticeable differences. The cone $\mathbf{D}^p$ has empty interior in $\mathbf{S}^p$, while the interior of $\mathbf{D}_0^p$ is nonempty (it contains the matrix $X = -I$). Additionally, $\mathbf{D}^p$ is pointed (since it contains only nonnegative matrices), while $\mathbf{D}_0^p$ is not, as it contains the subspace $\{a\mathbf{1}^T + \mathbf{1}a^T \mid a \in \mathbb{R}^p\}$.

A sparse matrix $X \in \mathbf{S}_E^p$ has an EDM completion if it is the projection $X = \Pi_E(D)$ of some EDM D. The cone

$$\Pi_E(\mathbf{D}^p) = \{\Pi_E(X) \mid X \in \mathbf{D}^p\}$$

is a closed convex cone, as can be seen as follows. First suppose the sparsity graph $\mathcal{G}_E$ is connected. If $\Pi_E(X) = 0$ for some $X \in \mathbf{D}^p$, with $X_{ij} = \|u_i - u_j\|^2$ for $i, j = 1, \ldots, p$, then $u_i = u_j$ for all $\{i, j\} \in E$. If the graph is connected, this implies that the position vectors u_i are all equal, i.e., $X = 0$. Hence $\Pi_E(X) = 0$, $X \in \mathbf{D}^p$ only holds if $X = 0$. It then follows from [53, Theorem 9.1] that $\Pi_E(\mathbf{D}^p)$ is closed. Next, assume $\mathcal{G}_E$ has d connected components, with vertex sets $\alpha_1, \ldots, \alpha_d \subset \{1, 2, \ldots, p\}$. For $k = 1, \ldots, d$, let $E_k = \{\{i, j\} \mid \{\alpha_k(i), \alpha_k(j)\} \in E\}$ be the edge sets of the connected components of $\mathcal{G}_E$. Since for each k, the graph with vertex set $\{1, 2, \ldots, |\alpha_k|\}$ and edge set E_k is connected, the sets $\Pi_{E_k}(\mathbf{D}^{|\alpha_k|})$ are all closed convex cones. Additionally, $X \in \Pi_E(\mathbf{D}^p)$ if and only if $X \in \mathbf{S}_E^p$ and $X_{\alpha_k \alpha_k} \in \Pi_{E_k}(\mathbf{D}^{|\alpha_k|})$ for $k = 1, \ldots, d$. Hence $\Pi_E(\mathbf{D}^p)$ is the intersection of closed convex sets. This result is also given in [25, Theorem 3.2].

Bakonyi and Johnson have formulated a clique decomposition theorem for EDM completions analogous to Theorem 3.2 for PSD completions [7].

THEOREM 3.3. *Let E' be a chordal sparsity pattern of order p, with cliques β_k, $k = 1, \ldots, m$. A matrix $X \in \mathbf{S}_{E'}^p$ is in $\Pi_{E'}(\mathbf{D}^p)$ if and only if*

$$P_{\beta_k} X P_{\beta_k}^T \in \mathbf{D}^{|\beta_k|}, \quad k = 1, \ldots, m.$$

If E' is not chordal, the condition in the theorem is necessary for X to be in $\Pi_{E'}(\mathbf{D}^p)$ but not sufficient.

In combination with Schoenberg's condition, this theorem allows us to formulate the problem of projecting a matrix $B \in \mathbf{S}_E^p$ on $\Pi_E(\mathbf{D}^p)$:

$$\begin{array}{ll}\text{minimize} & \|\Pi_E(X) - B\|_F^2 \\ \text{subject to} & P_{\beta_k} X P_{\beta_k}^T \in \mathbf{D}^{|\beta_k|}, \quad k = 1, \ldots, m,\end{array}$$

with variable $X \in \mathbf{S}_{E'}^p$, where E' is a chordal extension of E, and $\beta_1, \ldots, \beta_m$ are the cliques of E'. Equivalently, we can write the problem as

$$\begin{array}{ll}\text{minimize} & \|\Pi_E(X) - B\|_F^2 \\ \text{subject to} & P_{\beta_k} X P_{\beta_k}^T \in \mathbf{D}_0^{|\beta_k|}, \quad k = 1, \ldots, m, \\ & \operatorname{diag}(X) = 0.\end{array} \tag{3.5}$$

Problem (3.5) is strictly feasible. (For example, the matrix $X = \mathbf{1}\mathbf{1}^T - I$ is strictly feasible.) We will use (3.5) because projections on $\mathbf{D}_0^{|\beta_k|}$ are simpler than projections on $\mathbf{D}^{|\beta_k|}$, as will be discussed shortly. Problem (3.5) can be written in vector form as (2.1), with variable $x = \mathbf{vec}_{E'}(X)$. Here, $n = p+|E'|$, $a = \mathbf{vec}_E(B)$, and $l = m+1$. The index sets η, $\gamma_1, \ldots, \gamma_m$ are defined by the identity (3.2) for all $X \in \mathbf{S}^p_{E'}$. The index set γ_{m+1} is defined by

$$P_{\gamma_{m+1}} \mathbf{vec}_{E'}(X) = \mathrm{diag}(X) \quad \forall X \in \mathbf{S}^p_{E'}.$$

The cones $\mathcal{C}_k$ are defined as $\mathcal{C}_k = \{\mathbf{vec}(X) \mid X \in \mathbf{D}_0^{|\beta_k|}\}$, $k = 1, \ldots, m$, and $\mathcal{C}_{m+1} = \{0\} \subset \mathbb{R}^p$.

Formulas for projecting on $\mathbf{D}_0^p$ can be found in [28, 29, 34]. Let V be a $p \times (p-1)$ matrix with orthonormal columns that span the orthogonal complement of $\mathbf{1}$. An example is the matrix

$$V = \frac{1}{p+\sqrt{p}} \begin{bmatrix} (1+\sqrt{p})\mathbf{1}^T \\ \mathbf{1}\mathbf{1}^T - (p+\sqrt{p})I \end{bmatrix}.$$

Define $e = (1/\sqrt{p})\mathbf{1} \in \mathbb{R}^p$ and $Q = [\ V \quad e\ ]$. In [29, 34] the projection $\Pi_{\mathbf{D}_0^p}(D)$ of a matrix $D \in \mathbf{S}^p$ on $\mathbf{D}_0^p$ is computed directly as the solution of

$$\begin{array}{ll} \text{minimize} & \|X - D\|_F^2 \\ \text{subject to} & V^T X V \preceq 0. \end{array} \tag{3.6}$$

The solution is

$$\Pi_{\mathbf{D}_0^p}(D) = Q \begin{bmatrix} -\Pi_{\mathbf{S}_+^p}(-V^T D V) & V^T D e \\ e^T D v & e^T D e \end{bmatrix} Q^T,$$

where $-\Pi_{\mathbf{S}_+^p}(-V^T D V)$ is the projection of $V^T D V$ on the negative semidefinite cone, obtained by replacing the positive eigenvalues of $V^T D V$ by zero.

The method in [28] computes the projection indirectly, via the projection on the dual cone of $\mathbf{D}_0^p$ and the formula

$$\Pi_{\mathbf{D}_0^p}(D) = D - \Pi_{-(\mathbf{D}_0^p)^*}(D). \tag{3.7}$$

The dual cone is $(\mathbf{D}_0^p)^* = \{VZV^T \mid Z \preceq 0\}$, so the projection of D on $-(\mathbf{D}_0^p)^*$ is the solution of

$$\begin{array}{ll} \text{minimize} & \|VZV^T - D\|_F^2 \\ \text{subject to} & Z \succeq 0. \end{array}$$

The solution is $Z = \Pi_{\mathbf{S}_+^{p-1}}(V^T D V)$ and substituting in (3.7) gives the expression

$$\Pi_{\mathbf{D}_0^p}(D) = D - V \Pi_{\mathbf{S}_+^{p-1}}(V^T D V) V^T.$$

4. Projection via dual decomposition. In this section we present algorithms for the primal and dual problems (2.1) and (2.2) when $\eta = \{1, 2, \ldots, n\}$ and $P_\eta = I$. The primal and dual problems are

$$\begin{array}{ll} \text{minimize} & \|x - a\|^2 \\ \text{subject to} & P_{\gamma_k} x \in \mathcal{C}_k, \quad k = 1, \ldots, l, \end{array} \tag{4.1}$$

and

$$\begin{array}{ll} \text{maximize} & -\|\sum_{k=1}^{l} P_{\gamma_k}^T z_k + a\|^2 + \|a\|^2 \\ \text{subject to} & z_k \in \mathcal{C}_k^*, \quad k = 1, \ldots, l. \end{array} \tag{4.2}$$

As explained in the previous section, the three sparse matrix nearness problems can be written in this form when $E' = E$ and E is a chordal sparsity pattern.

There is a rich literature on decomposition methods for computing the projection on an intersection of closed convex sets via a sequence of projections on each set; see, for example, [26, 33, 60, 61] and the books [12, 19]. We will discuss two approaches based on duality. In the first approach (section 4.2) the gradient projection method [46, section 7.2.1] and accelerated gradient projection method [9, 44, 45] are applied to the dual problem. These algorithms can be viewed as applications of Tseng's *alternating minimization method* for minimizing a strongly convex function over the intersection of convex sets [10, 60, 61]. The second approach (section 4.3) is the dual block coordinate ascent method [62]. This method can be interpreted as a generalization of Dykstra's cyclic projection algorithm [18, 26] or Han's successive projection algorithm [32], and also as a dual block coordinate gradient projection method [11]. In the next sections, we first review some duality properties that underlie the two approaches, and then present the two algorithms.

4.1. Dual decomposition. For notational convenience, we introduce the notation

$$P = \begin{bmatrix} P_{\gamma_1}^T & P_{\gamma_2}^T & \cdots & P_{\gamma_l}^T \end{bmatrix}^T, \quad \mathcal{C} = \mathcal{C}_1 \times \mathcal{C}_2 \times \cdots \times \mathcal{C}_l, \quad \mathcal{C}^* = \mathcal{C}_1^* \times \mathcal{C}_2^* \times \cdots \times \mathcal{C}_l^*.$$

With this notation we can write (4.1) and (4.2) as

$$\begin{array}{lll} \text{minimize} & \|x - a\|^2 & \\ \text{subject to} & Px \in \mathcal{C}, & \end{array} \qquad \begin{array}{lll} \text{maximize} & -\|P^T z + a\|^2 + \|a\|^2 \\ \text{subject to} & z \in \mathcal{C}^*, \end{array}$$

where $x = (x_1, x_2, \ldots, x_l)$ and $z = (z_1, z_2, \ldots, z_l)$. Note that the matrix $P^T P$ is diagonal and that the jth diagonal entry is the number of index sets γ_k that contain the index j:

$$(P^T P)_{jj} = |\{k \mid j \in \gamma_k\}|, \quad j = 1, \ldots, n. \tag{4.3}$$

The dual decomposition methods we discuss in the next two sections are descent methods for minimizing

$$f(z) = \frac{1}{2}\|P^T z + a\|^2 = \frac{1}{2}\left\|\sum_{k=1}^{l} P_{\gamma_k}^T z_k + a\right\|^2 \tag{4.4}$$

over the product cone $\mathcal{C}^* = \mathcal{C}_1^* \times \mathcal{C}_2^* \times \cdots \times \mathcal{C}_l^*$. The methods generate a sequence of dual feasible suboptimal points z. From a dual feasible $z \in \mathcal{C}^*$, approximate primal and dual projections x and s are computed as $x = P^T z + a$ and $s = P^T z$. The distances to optimality $\|x - x^\star\|$ and $\|s - s^\star\|$ can be bounded in terms of the dual suboptimality $f(z) - f(z^\star)$. To see this, we note that for any dual optimal solution $z^\star$ and any $z \in \mathcal{C}^*$, we have

$$\begin{aligned} \|P^T z - P^T z^\star\|^2 &= \|P^T z + a\|^2 - \|P^T z^\star + a\|^2 - 2(Pz^\star + a)^T P^T (z - z^\star) \\ &\leq \|P^T z + a\|^2 - \|P^T z^\star + a\|^2. \end{aligned}$$

The inequality holds because

$$\nabla f(z^\star)^T(z - z^\star) = (P^T z^\star + a)^T P^T(z - z^\star) \geq 0$$

for all $z \in \mathcal{C}^*$ if $z^\star$ is optimal. Hence, if z is dual feasible and we define $s = P^T z$, $x = P^T z + a$, then

$$\begin{aligned}\frac{1}{2}\|x - x^\star\|^2 = \frac{1}{2}\|s - s^\star\|^2 &= \frac{1}{2}\|P^T(z - z^\star)\|^2 \\ &\leq \frac{1}{2}\|P^T z + a\|^2 - \frac{1}{2}\|P^T z^\star + a\|^2 \\ &= f(z) - f(z^\star).\end{aligned}$$

This simple inequality allows us to translate convergence results for the dual objective function to convergence rates for the distances of the primal and dual solutions to optimality.

4.2. Gradient projection method. The gradient projection method for minimizing the function (4.4) over $\mathcal{C}^*$ uses the iteration

$$\begin{aligned}z^i &= \Pi_{\mathcal{C}^*}\left(z^{i-1} - t\nabla f(z^{i-1})\right) \\ &= \Pi_{\mathcal{C}^*}\left(z^{i-1} - tP(P^T z^{i-1} + a)\right),\end{aligned} \tag{4.5}$$

where t is a positive step size. In terms of the block vectors,

$$\begin{aligned}z_k^i &= \Pi_{\mathcal{C}_k^*}\left(z_k^{i-1} - tP_{\gamma_k}\left(\sum_{j=1}^{l} P_{\gamma_j}^T z_j^{i-1} + a\right)\right) \\ &= \Pi_{\mathcal{C}_k^*}\left((1-t)z_k^{i-1} - tP_{\gamma_k}\left(\sum_{j\neq k} P_{\gamma_j}^T z_j^{i-1} + a\right)\right).\end{aligned}$$

On line 2 we use the fact that $P_{\gamma_k} P_{\gamma_k}^T = I$. The projections on $\mathcal{C}_k^*$ can also be expressed in terms of projections on $\mathcal{C}_k$ via the identity $u = \Pi_{\mathcal{C}_k}(u) - \Pi_{\mathcal{C}_k^*}(-u)$.

We use a fixed step size $t = 1/L$ where L is the largest eigenvalue of the Hessian $\nabla^2 f(z) = PP^T$:

$$L = \lambda_{\max}(PP^T) = \lambda_{\max}(P^T P) = \max_{j=1,\ldots,n} |\{k \mid j \in \gamma_k\}|. \tag{4.6}$$

(Recall that $P^T P$ is diagonal; see (4.3).) A standard convergence result states that with this step size the sequence z^i converges to a minimizer of f over $\mathcal{C}^*$, even when the minimizers are not unique [46, page 207]. Moreover the dual optimality gap decreases as

$$f(z^i) - f(z^\star) \leq \frac{L}{2i}\|z^0 - z^\star\|^2,$$

where $z^\star$ is any optimal solution [9, Theorem 3.1]. From (4.5) it then follows that the sequences $x^i = P^T z^i + a$ and $s^i = P^T z^i$ converge to the projections $x^\star$ and $s^\star$ at a rate $O(1/\sqrt{i})$:

$$\|x^i - x^\star\| = \|s^i - s^\star\| \leq \sqrt{\frac{L}{i}}\|z^0 - z^\star\|.$$

The gradient projection algorithm is summarized in Algorithm 1. It is important to keep in mind that the projection on line 4 reduces to l projections on $\mathcal{C}_k^*$ or $\mathcal{C}_k$, and can be computed in parallel:

$$z_k^i = \Pi_{\mathcal{C}_k^*}(y_k^i) = y_k^i + \Pi_{\mathcal{C}_k}(-y_k^i), \quad k = 1, \ldots, l.$$

The stopping condition of the algorithm is left open for now and will be discussed at the end of this section.

ALGORITHM 1. GRADIENT PROJECTION METHOD FOR PROBLEMS (4.1) AND (4.2).

1 **Initialize:** Set $t = 1/L$ with L defined in (4.6). Choose an initial $z^0 = (z_1^0, \ldots, z_l^0)$, and take $s^0 = P^T z^0$ and $x^0 = s^0 + a$.

2 **for** $i = 1, 2, \ldots$ until convergence **do**

3 *Gradient step.* Compute $y^i = z^{i-1} - tPx^{i-1}$.

4 *Projection step.* Compute $z^i = \Pi_{\mathcal{C}^*}(y^i)$, $s^i = P^T z^i$, and $x^i = s^i + a$.

5 **end for**

An important advantage of the gradient projection method is the availability of Nesterov-type accelerations [9, 44, 45, 63]. We will use Nesterov's first accelerated gradient projection method [44], which was later generalized to an accelerated proximal gradient method in [9] and is widely known under the acronym FISTA used in [9]; see also [63]. FISTA applies a gradient projection update after an extrapolation step:

$$z^i = \Pi_{\mathcal{C}^*}(v^i - t\nabla f(v^i)), \quad \text{where } v^i = z^{i-1} + \frac{i-2}{i+1}(z^{i-1} - z^{i-2})$$

(with the assumption $z^{-1} = z^0$, so the first iteration is the standard gradient projection update). The same step size $1/L$ is used as in the gradient projection method. The accelerated gradient projection algorithm has the same complexity per iteration as the basic algorithm.

As for the gradient projection method, the iterates z^i of the fast gradient projection method can be shown to converge, even when the optimal solutions are not unique. This is discussed in the recent paper [20]. The dual optimality gap decreases as $1/i^2$:

$$f(z^i) - f(z^\star) \leq \frac{2L}{(i+1)^2}\|z^0 - z^\star\|^2,$$

where $z^\star$ is any optimal solution [9, Theorem 4.4]. By the same argument as used for the gradient projection method, this leads to the following convergence result for the sequences $x^i = P^T z^i + a$ and $s^i = P^T z^i$:

$$\|x^i - x^\star\| = \|s^i - s^\star\| \leq \frac{2\sqrt{L}}{i+1}\left\|z^0 - z^\star\right\|.$$

The $O(1/i)$ convergence rate is an improvement over the $O(1/\sqrt{i})$ rate of the gradient projection method. The method is summarized in Algorithm 2.

Various stopping criteria can be used for Algorithms 1 and 2. For example, one can bound the error with which the iterates satisfy the optimality condition $z = \Pi_{\mathcal{C}^*}(z - t\nabla f(z))$ for problem (4.4), where t is any positive number. In the gradient projection algorithm (Algorithm 1) we have

$$z^{i-1} = \Pi_{\mathcal{C}^*}(z^{i-1} - t\nabla f(z^{i-1})) + r^i$$

ALGORITHM 2. FAST GRADIENT PROJECTION METHOD FOR PROBLEMS (4.1) AND (4.2).

Initialize: Set $t = 1/L$ with L defined in (4.6). Choose an initial $z^{-1} = z^0 = (z_1^0, \ldots, z_l^0)$, and take $s^0 = P^T z^0$ and $x^0 = s^0 + a$.
for $i = 1, 2, \ldots$ until convergence **do**
  Extrapolation step. Compute $v^i = z^{i-1} + \frac{i-2}{i+1}\left(z^{i-1} - z^{i-2}\right)$.
  Gradient step. Compute $y^i = v^i - tP(P^T v^i + a)$.
  Projection step. Compute $z^i = \Pi_{\mathcal{C}^*}(y^i)$, $s^i = P^T z^i$, and $x^i = s^i + a$.
end for

with $r^i = z^{i-1} - z^i$. In the fast gradient projection algorithm (Algorithm 2) we have

$$v^i = \Pi_{\mathcal{C}^*}(v^i - t\nabla f(v^i)) + r^i$$

with $r^i = v^i - z^i$. This suggests using stopping conditions

$$\frac{\|z^i - z^{i-1}\|}{\max\{\|z^i\|, 1\}} \leq \epsilon, \qquad \frac{\|v^i - z^i\|}{\max\{\|z^i\|, 1\}} \leq \epsilon$$

for the proximal gradient method and the fast proximal gradient method, respectively.

4.3. Dual block coordinate ascent. Dykstra's algorithm [18, 26], also known as the *successive projection algorithm* [32], is a method for computing the Euclidean projection of a point on the intersection of convex sets, via a cyclic sequence of projections on each set. It can be interpreted as a dual block coordinate ascent algorithm [28, 32, 62].

Dykstra's method has been applied to several types of dense matrix nearness problems in the literature. The problem of finding the nearest EDM matrix to a given matrix is discussed in [29] and [28, section 5.3]. The approach taken in these papers is to formulate the projection on $\mathbf{D}^p$ as a projection on the intersection of $\mathbf{D}_0^p$ and $\{X \in \mathbf{S}^p \mid \mathrm{diag}(X) = 0\}$. The nearest correlation matrix problem, i.e., projection on the intersection of $\mathbf{S}_+^p$ and $\{X \in \mathbf{S}^p \mid \mathrm{diag}(X) = \mathbf{1}\}$, is discussed in [37, 39, 42].

Following the dual interpretation of Dykstra's method we apply block coordinate ascent to the dual problem (4.2). If we fix z_j for $j \neq k$ and minimize $f(z)$ over z_k only, the problem reduces to

$$\begin{array}{ll} \text{minimize} & \left\| z_k + P_{\gamma_k}\left(\sum_{j \neq k} P_{\gamma_j}^T z_j + a\right)\right\|^2 \\ \text{subject to} & z_k \in \mathcal{C}_k^*. \end{array}$$

This is a Euclidean projection of the point $w = -P_{\gamma_k}(\sum_{j\neq k} P_{\gamma_j}^T z_j + a)$ on $\mathcal{C}_k^*$. The unique solution can be expressed in two equivalent forms:

$$z_k = \Pi_{\mathcal{C}_k^*}(w) = \Pi_{\mathcal{C}_k}(-w) + w. \tag{4.7}$$

Alternatively, we can view this as a block coordinate gradient projection update. The gradient of $f(z)$ with respect to z_k is

$$\nabla_{z_k} f(z) = P_{\gamma_k}\left(\sum_{j=1}^{l} P_{\gamma_j}^T z_j + a\right) = z_k + P_{\gamma_k}\left(\sum_{j \neq k} P_{\gamma_j}^T z_j + a\right),$$

because $P_{\gamma_k} P_{\gamma_k}^T = I$. The vector w is therefore equal to $w = z_k - \nabla_{z_k} f(z)$ and $\Pi_{\mathcal{C}_k^*}(w)$ is a block coordinate gradient projection step with step size one.

In Algorithm 3 we minimize over $z_1, \ldots, z_l$ cyclically, using the second expression in (4.7). We also maintain a primal variable $x^i = \sum_{j=1}^l P_{\gamma_j}^T z_j^i + a$. The simplest initialization is to take $z^0 = 0$ and $x^0 = a$. On line 4 we project the point

$$P_{\gamma_k}\left(\sum_{j\neq k} P_{\gamma_j}^T z_j^{i-1} + a\right) = P_{\gamma_k} x^{i-1} - z_k^{i-1}$$

on $\mathcal{C}_k$. The update on line 5 is easier to describe in words: x^i is equal to x^{i-1} with the subvector $x_{\gamma_k}^{i-1}$ replaced by v. Line 6 can also be written as $z_k^i = z_k^{i-1} + v - P_{\gamma_k} x^{i-1}$. Hence

$$z_k^i = z_k^{i-1} + v - P_{\gamma_k}\left(\sum_{j=1}^l P_{\gamma_j}^T z_j^{i-1} + a\right) = v - P_{\gamma_k}\left(\sum_{j\neq k} P_{\gamma_j}^T z_j^{i-1} + a\right).$$

ALGORITHM 3. DUAL BLOCK COORDINATE ASCENT FOR PROBLEMS (4.1) AND (4.2).

1 **Initialize:** Choose an initial $(z_1^0, \ldots, z_l^0) \in \mathcal{C}^*$ and set $x^0 = P^T z^0 + a$.
2 **for** $i = 1, 2, \ldots$ until convergence **do**
3 *Select the next index:* $k = (i-1) \bmod l + 1$.
4 *Projection.* Compute $v = \Pi_{\mathcal{C}_k}(P_{\gamma_k} x^{i-1} - z_k^{i-1})$.
5 *Primal variable.* Compute $x^i = P_{\gamma_k}^T v + (I - P_{\gamma_k}^T P_{\gamma_k}) x^{i-1}$.
6 *Dual variables.* Compute $z_k^i = z_k^{i-1} + P_{\gamma_k}(x^i - x^{i-1})$ and $z_j^i = z_j^{i-1}$ for $j \neq k$.
7 **end for**

When $P_{\gamma_k} = I$ for $k = 1, \ldots, l$, and with the initialization $x^0 = a$, $z^0 = 0$, this is Dykstra's algorithm for computing the projection on $\bigcap_{k=1}^l \mathcal{K}_k$ [26]. Algorithm 3 is a special case of Tseng's *dual block coordinate ascent algorithm* [62, section 3] and convergence follows from [62, Theorem 3.1]. We can also give a convergence rate by applying a recent result by Beck and Tetruashvili on block coordinate gradient projection algorithms [11, Theorem 6.3]: we have $f(z^i) - f(z^\star) \leq c/i$, where c is a constant. It then follows from (4.5) that x^i and $s^i = P^T z^i$ satisfy $\|x^i - x^\star\| = \|z^i - z^\star\| = O(1/\sqrt{i})$.

In general, deciding when to terminate the block coordinate ascent method can be difficult, since the iterates may remain constant for several successive iterations. The stopping condition proposed in [13, eq. (13)] is based on measuring the residual $r^j = z^{jl} - z^{(j-1)l}$, i.e., the difference between the values of z at the end of two successive cycles. We will use a similar stopping condition of the form

$$\frac{\|z^{jl} - z^{(j-1)l}\|}{\max\{\|z^{jl}\|, 1\}} \leq \epsilon.$$

It can be shown that if $r^j = 0$ then the iterates x^i have remained constant during cycle j. This can be seen from the expression

$$\|r^j\|^2 = \sum_{k=1}^l \|z_k^{(j-1)l+k} - z_k^{(j-1)l+k-1}\|^2 = \sum_{k=1}^l \|x^{(j-1)l+k} - x^{(j-1)l+k-1}\|^2.$$

The first step follows because z_k changes only in iteration k of cycle j; the second step follows from line 6 in the algorithm. This observation implies that x^{jl} and z^{jl} are optimal if $r^j = 0$. By construction, $x^i - a = P^T z^i$ after each step of the algorithm. Additionally, if $k = ((i-1) \bmod l) + 1$ then the two relations $z_k^i = \Pi_{\mathcal{C}_k^*}(w)$ and $P_{\gamma_k} x^i = -\Pi_{\mathcal{C}_k}(w)$ hold, where $w = -P_{\gamma_k}(\sum_{j \neq k} P_{\gamma_j}^T z_j^i + a)$. This shows that z^i is always dual feasible. Moreover if x^i remains unchanged for an entire cycle then it is primal feasible and $(z_k^i)^T P_{\gamma_k} x^i = 0$ for $k = 1, \ldots, l$.

5. Projection via Douglas–Rachford splitting. We now turn to the general problem (2.1) for arbitrary index sets η. The assumption that $\eta = \{1, 2, \ldots, n\}$ and $P_\eta = I$ was crucial in the dual decomposition approaches of the previous section 4. When $\eta \neq \{1, 2, \ldots, n\}$ the dual problem (2.3) includes a coupling equality constraint and the separable structure exploited by the dual gradient projection and block coordinate ascent methods is no longer available. In this section, we present decomposition methods based on the Douglas–Rachford splitting method [8, 27, 41].

5.1. Douglas–Rachford algorithm. The Douglas–Rachford method is a popular method for minimizing a sum $f(\tilde{x}) = g(\tilde{x}) + h(\tilde{x})$ of two closed, convex, and possibly nonsmooth functions g and h. In our applications, the second function will be the indicator function of a subspace $\mathcal{V}$, i.e., we solve

$$\begin{array}{ll} \text{minimize} & g(\tilde{x}) \\ \text{subject to} & \tilde{x} \in \mathcal{V}. \end{array} \tag{5.1}$$

The Douglas–Rachford method specialized to this problem is also known as the *method of partial inverses* [27, 56, 57]. The algorithm starts at an arbitrary $\tilde{z}^0$ and repeats the following iteration:

$$\begin{aligned} \tilde{x}^{i+1} &= \text{prox}_{tg}(\tilde{z}^i), \\ \tilde{y}^{i+1} &= \Pi_{\mathcal{V}}(2\tilde{x}^{i+1} - \tilde{z}^i), \\ \tilde{z}^{i+1} &= \tilde{z}^i + \rho(\tilde{y}^{i+1} - \tilde{x}^{i+1}). \end{aligned} \tag{5.2}$$

There are two algorithm parameters: a step size $t > 0$ and a relaxation parameter $\rho \in (0, 2)$. The function prox_{tg} in the first step is the *proximal operator* of $tg(x)$, defined as

$$\text{prox}_{tg}(z) = \operatorname*{argmin}_u \left(g(u) + \frac{1}{2t}\|u - z\|^2 \right).$$

In this paper, only two types of proximal operators will be encountered. If $g(\tilde{x}) = \delta_{\mathcal{S}}(\tilde{x})$ is the indicator function of a closed convex set $\mathcal{S}$, then $\text{prox}_{tg}(\tilde{x}) = \Pi_{\mathcal{S}}(\tilde{x})$ is the Euclidean projection on $\mathcal{S}$. If g is a convex quadratic function $g(\tilde{x}) = (1/2)\tilde{x}^T Q \tilde{x} + q^T \tilde{x}$, then

$$\text{prox}_{t\tilde{g}}(\tilde{x}) = (I + tQ)^{-1}(\tilde{x} - tq).$$

The Douglas–Rachford method (5.2) is of interest when the projection on $\mathcal{V}$ and the proximal operator of g are inexpensive.

The convergence of the Douglas–Rachford algorithm has been studied extensively. It can be shown that if g is a closed convex function, with $\mathcal{V} \cap \text{ri}\,\mathbf{dom}\, g \neq \emptyset$, and the problem (5.1) has a solution, then the sequences $\tilde{x}^i$ and $\tilde{y}^i$ converge to a solution [8, Corollary 27.2]. Convergence rates are usually expressed in terms of fixed point

residuals. Recent results by Davis and Yin [21, Theorem 7] also give a convergence rate for the objective value: it is shown that $|f(\tilde{x}^i) - f(\tilde{x}^\star)| = o(1/\sqrt{i})$. The bound can be improved if g satisfies additional properties, e.g., if its gradient is Lipschitz continuous [22].

We will use the simple stopping condition described in [58],

$$\frac{\|r_{\mathrm{p}}^i\|}{\max\{\|\tilde{x}^i\|, 1\}} \leq \epsilon_{\mathrm{p}}, \qquad \frac{\|r_{\mathrm{d}}^i\|}{\max\{\|\tilde{s}^i\|, 1\}} \leq \epsilon_{\mathrm{d}},$$

for some primal and dual tolerances ϵ_{p} and ϵ_{d}, where $\tilde{s} = t^{-1}(\tilde{z}^{i-1} - \tilde{x}^i)$ and

$$r_{\mathrm{p}}^i = \Pi_{\mathcal{V}}(\tilde{x}^i) - \tilde{x}^i, \qquad r_{\mathrm{d}}^i = -\Pi_{\mathcal{V}}(\tilde{s})$$

are primal and dual residuals.

5.2. Primal splitting. To apply the Douglas–Rachford algorithm to the primal problem (2.1), we first reformulate it as

$$\begin{array}{ll} \text{minimize} & \dfrac{1}{2}\|P_\eta x - a\|^2 + \displaystyle\sum_{k=1}^{l} \delta_{\mathcal{C}_k}(y_k) \\ \text{subject to} & P_{\gamma_k} x = y_k, \quad k = 1, \ldots, l. \end{array} \tag{5.3}$$

The variables are $x \in \mathbb{R}^n$ and an additional splitting variable $y = (y_1, y_2, \ldots, y_l)$. This problem has the form (5.1) if we take $\tilde{x} = (x, y)$,

$$g(x, y) = \frac{1}{2}\|P_\eta x - a\|^2 + \delta_{\mathcal{C}}(y), \qquad \mathcal{V} = \{(x, y) \mid y = Px\}.$$

The function g is separable with proximal operator

$$\mathrm{prox}_{tg}(x, y) = \begin{bmatrix} (I + tP_\eta^T P_\eta)^{-1}(x + tP_\eta^T a) \\ \Pi_{\mathcal{C}}(y) \end{bmatrix}. \tag{5.4}$$

Note that the inverse in the first block is the inverse of a strictly positive diagonal matrix, since $P_\eta^T P_\eta$ is diagonal with $(P_\eta^T P_\eta)_{ii} = 1$ if $i \in \eta$ and $(P_\eta^T P_\eta)_{ii} = 0$ otherwise. The projection on $\mathcal{C}$ in the second block reduces to l independent projections $\Pi_{\mathcal{C}}(y) = (\Pi_{\mathcal{C}_1}(y_1), \ldots, \Pi_{\mathcal{C}_l}(y_l))$. The projection on the subspace $\mathcal{V}$ in (5.2) is

$$\Pi_{\mathcal{V}}(x, y) = \begin{bmatrix} I \\ P \end{bmatrix} (I + P^T P)^{-1}(x + P^T y), \tag{5.5}$$

which is also simple to compute since $P^T P$ is diagonal; see (4.3).

A summary of the Douglas–Rachford method is given in Algorithm 4.

The algorithm is easily extended to generalizations of problems (5.3) with simple equality constraints $Ax = b$ added. An example is the constraint $\mathrm{diag}(X) = \mathbf{1}$ in (3.1) to compute the projection on the matrices with a correlation matrix completion. The additional constraint can be handled by adding the indicator function $\delta_{\{0\}}(Ax - b)$ to the objective function of (5.3). The proximal operator of

$$\frac{1}{2}\|P_\eta x - a\|^2 + \delta_{\{0\}}(Ax - b)$$

is a weighted projection on an affine set and is inexpensive if A has few rows.

Algorithm 4. Douglas–Rachford method for primal problem (2.1).

Initialize: Choose parameters $t > 0$, $\rho \in (0, 2)$, initial u^0, $v_1^0, \ldots, v_l^0$.
for $i = 1, 2, \ldots$ until convergence **do**
  Compute $(x^i, y^i) = \mathrm{prox}_{tg}(u^{i-1}, v^{i-1})$ using (5.4).
  Compute $(w^i, z^i) = \Pi_{\mathcal{V}}(2x^i - u^{i-1}, 2y^i - v^{i-1})$ using (5.5).
  Update $u^i = u^{i-1} + \rho(w^i - x^i)$ and $v_k^i = v_k^{i-1} + \rho(z_k^i - y_k^i), \quad k = 1, \ldots, l.$
end for

5.3. Dual splitting. To apply the Douglas–Rachford method to the dual problem (2.2) we write it as

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|s + a\|^2 + \sum_{k=1}^{l} \delta_{\mathcal{C}_k^*}(z_k) \\ \text{subject to} & \sum_{k=1}^{l} P_{\gamma_k}^T z_k = P_\eta^T s. \end{array} \tag{5.6}$$

This is in the form (5.1) with $\tilde{x} = (s, z)$,

$$g(s, z) = \frac{1}{2}\|s + a\|^2 + \delta_{\mathcal{C}^*}(z), \qquad \mathcal{V} = \{(s, z) \mid P_\eta^T s = P^T z\}.$$

The proximal operator of g is

$$\mathrm{prox}_{tg}(s, z) = \begin{bmatrix} (1 + t)^{-1}(s - ta) \\ \Pi_{\mathcal{C}^*}(z) \end{bmatrix}. \tag{5.7}$$

The projection on $\mathcal{V}$ can be written as

$$\Pi_{\mathcal{V}}(s, z) = \begin{bmatrix} s \\ z \end{bmatrix} + \begin{bmatrix} -P_\eta \\ P \end{bmatrix} (P_\eta^T P_\eta + P^T P)^{-1}(P_\eta^T s - P^T z). \tag{5.8}$$

A summary of the Douglas–Rachform method using splitting is given in Algorithm 5.

Algorithm 5. Douglas–Rachford method for dual problem (2.2).

Initialize: Choose parameters $t > 0$, $\rho \in (0, 2)$, initial u^0, $v_1^0, \ldots, v_l^0$.
for $i = 1, 2, \ldots$ until convergence **do**
  Compute $(s^i, z^i) = \mathrm{prox}_{tg}(u^{i-1}, v^{i-1})$ using (5.7).
  Compute $(w^i, y^i) = \Pi_{\mathcal{V}}(2s^i - u^{i-1}, 2z^i - v^{i-1})$ using (5.8).
  Update $u^i = u^{i-1} + \rho(w^i - s^i)$ and $v_k^i = v_k^{i-1} + \rho(y_k^i - z_k^i), \quad k = 1, \ldots, l.$
end for

6. Numerical results. In this section we test the proposed algorithms on problems with sizes ranging from 1000 to 100000. The problems in the first set of experiments are constructed from thirteen symmetric sparsity patterns in the University of Florida sparse matrix collection [24]. In the second set of experiments, we consider a family of randomly generated sensor network node localization problems [3, 14, 40].

The experiments are performed on an Intel Xeon CPU E31225 processor with 32 GB RAM, running Ubuntu 14.04 (Trusty) and using MATLAB version 8.3.0 (2014a). In the dual gradient projection and Douglas–Rachford algorithms the main step per iteration is the projection on l dense PSD or EDM cones, and these projections can be

TABLE 1
Thirteen symmetric sparsity patterns from the University of Florida sparse matrix collection. For each pattern we give p, the number of rows and columns of the matrix, and the density defined as $(p + 2|E|)/p^2$.

Pattern	p	Density
ex4	1601	1.24e-2
c-26	4307	1.86e-3
g3rmt3m3	5357	7.24e-3
barth4	6019	1.13e-3
c-37	8204	1.11e-3
tuma2	12992	2.92e-4
crack_dual	20141	1.98e-4
biplane-9	21701	1.79e-4
mario001	38434	1.39e-4
c-60	43640	1.57e-4
c-67	57975	1.58e-4
rail_79841	79841	8.69e-5
luxembourg_osm	114599	1.82e-5

done in parallel. In the dual block coordinate ascent method, some of the projections can be computed in parallel, if they are scheduled using a topological ordering on a clique tree [15]. Since the projections are the most expensive part of the algorithms, exploiting parallelism would result in a significant speedup. This possibility was not exploited in the code used for the experiments, which computes the projections sequentially.

6.1. Chordal sparse matrix cones. We consider the three matrix nearness problems in sections 3.2–3.4 with chordal patterns E. The patterns are chordal extensions of thirteen nonchordal sparsity patterns from the University of Florida sparse matrix collection [24]. Table 1 gives some statistics for the patterns before the extension. The sparsity graphs of three of the larger patterns (`mario001`, `c-60`, `c-67`) are not connected, but since the largest connected component contains almost all the vertices, we did not remove the smaller connected components.

The chordal extensions are computed in two steps. We first use graph elimination (or symbolic Cholesky factorization) using a fill-reducing reordering (the MATLAB `amd` reordering) to generate a first chordal extension. We then merge some of the smaller cliques of this extension according to heuristics discussed in [58, section 6.2]. Table 2 gives the statistics of the two chordal extensions, before and after clique merging.

For each pattern E, we generate five instances B with lower-triangular nonzero values generated from independent normalized Gaussian distributions. The numerical results in the following tables are averages over the five instances.

We first consider projections on the PSD and PSD completable cones. When E is a chordal pattern, the formulation (3.3) of the projection of B on $\Pi_E(\mathbf{S}^p_+)$ simplifies to

$$\begin{array}{ll} \text{minimize} & \|X - B\|_F^2 \\ \text{subject to} & P_{\beta_k} X P_{\beta_k}^T \succeq 0, \quad k = 1, \ldots, m, \end{array} \tag{6.1}$$

where $\beta_1, \ldots, \beta_m$ are the cliques of the sparsity pattern. The dual of this problem is

$$\begin{array}{ll} \text{maximize} & -\left\| \sum_{k=1}^m P_{\beta_k}^T Z_k P_{\beta_k} + B \right\|_F^2 + \|B\|_F^2 \\ \text{subject to} & Z_k \succeq 0, \quad k = 1, \ldots, m, \end{array} \tag{6.2}$$

TABLE 2

Chordal extensions of the patterns of Table 1. *The table shows the density, the number of cliques* (m), *and the average and maximum clique size, after a chordal extension and after a further clique merging step.*

p	After extension				After clique merging			
	Density	m	Avg. clique size	Max. clique size	Density	m	Avg. clique size	Max. clique size
1601	3.24e-2	598	18.0	74	4.94e-2	94	46.3	74
4307	3.69e-3	3740	6.1	66	1.07e-2	556	17.4	66
5357	2.97e-2	577	52.9	261	3.27e-2	267	80.8	261
6019	6.12e-3	3637	11.1	89	1.11e-2	317	42.2	89
8204	4.05e-3	7158	7.1	257	9.54e-3	1121	21.5	257
12992	2.90e-3	11051	5.7	241	5.22e-3	704	37.2	241
20141	1.21e-3	17053	6.5	168	2.80e-3	1098	35.7	168
21701	1.41e-3	16755	8.0	147	2.99e-3	1099	40.7	147
38434	5.36e-4	30917	6.0	188	1.25e-3	2365	28.1	188
43640	1.37e-3	39468	6.2	954	2.56e-3	6175	19.5	954
57975	2.45e-4	52404	5.5	132	9.04e-4	8875	14.9	132
79841	5.31e-4	61059	8.7	337	9.71e-4	4247	44.4	337
114599	4.34e-5	113361	2.9	45	2.02e-4	7035	18.9	58

TABLE 3

Projection on chordal PSD completable matrices. CPU times (in seconds) for the projection on $\Pi_E(\mathbf{S}^p_+)$. *The total runtimes and times per iteration are given. The algorithms are: dual fast projected gradient method (F-PG), dual block coordinate ascent (BCD), primal Douglas–Rachford method (P-DR), and dual Douglas–Rachford method (D-DR).*

p	Total runtime				Time/iteration			
	F-PG	BCD	P-DR	D-DR	F-PG	BCD	P-DR	D-DR
1601	2.7e1	3.5	4.4	5.0	1.2e-1	1.4e-1	1.8e-1	2.0e-1
4307	5.5e2	6.8	3.3e1	3.6e1	2.0e-1	2.7e-1	3.3e-1	3.6e-1
5357	4.8e2	8.4e1	4.6e1	5.2e1	1.3	1.7	1.8	2.1
6019	1.2e2	9.6	1.3e1	1.6e1	3.4e-1	3.8e-1	5.1e-1	6.2e-1
8204	2.6e3	7.1e1	1.1e2	1.1e2	9.5e-1	1.4	1.4	1.5
12992	2.4e2	6.2e1	3.6e1	4.2e1	9.5e-1	1.2	1.4	1.7
20141	2.5e2	3.9e1	3.8e1	4.6e1	9.9e-1	1.6	1.5	1.9
21701	3.3e2	3.4e1	4.6e1	5.8e1	1.2	1.4	1.8	2.3
38434	4.7e2	4.7e1	6.2e1	7.8e1	2.1	1.9	2.5	3.1
43640	$>$ 4hr	1.9e3	1.6e3	1.5e3	1.0e1	1.9e1	1.6e1	1.5e1
57975	$>$ 4hr	1.4e2	1.1e3	1.1e3	3.5	5.7	6.4	6.2
79841	2.4e3	3.0e2	2.4e2	3.0e2	6.3	7.6	9.7	1.2e1
114599	5.3e2	5.5e1	1.0e2	1.2e2	2.6	2.2	4.0	4.6

and is equivalent to the projection of $-B$ on $\mathbf{S}^p_E \cap \mathbf{S}^p_+$, i.e., problem (3.1) for $A = -B$. We apply the fast projected gradient and block coordinate ascent methods to the dual problem (6.2), and the Douglas–Rachford method to the primal problem (written in the form (5.3)) and the dual problem (in the form (5.6)).

The step size in the fast projected gradient method is $t = 1/L$. In the Douglas–Rachford methods we used $t = 1$, $\rho = 1.75$. A tolerance $\epsilon = 10^{-3}$ was used in the various stopping conditions given in section 2. The algorithms were terminated when the CPU time exceeded a maximum four hours. The results are given in Table 3. The runtimes are averages over five instances; we did not observe a significant variance in runtime or number of iterations for the different instances. To reduce computational overhead, the stopping conditions are tested every 25 iterations.

TABLE 4

Projection on chordal EDM completable matrices. CPU times (in seconds) for the projection onto $\Pi_E(\mathbf{D}^p)$. The total runtimes and times per iteration are given. The algorithms are: dual fast projected gradient method (F-PG), dual block coordinate ascent (BCD), primal Douglas–Rachford method (P-DR), and dual Douglas–Rachford method (D-DR).

p	Total runtime				Time/iteration			
	F-PG	BCD	P-DR	D-DR	F-PG	BCD	P-DR	D-DR
1601	7.6e1	1.2e1	5.7	5.2	1.5e-1	1.6e-1	2.3e-1	2.1e-1
4307	1.9e3	3.2e1	2.1e1	2.0e1	2.9e-1	4.1e-1	4.2e-1	3.9e-1
5357	1.1e3	2.3e2	5.6e1	5.3e1	1.5	1.9	2.2	2.1
6019	3.9e2	2.7e1	1.7e1	1.6e1	4.4e-1	5.0e-1	6.9e-1	6.2e-1
8204	8.6e3	2.7e2	8.5e1	8.3e1	1.2	2.7	1.7	1.7
12992	6.8e2	1.8e2	4.5e1	4.1e1	1.2	1.5	1.8	1.6
20141	9.1e2	1.3e2	5.0e1	4.4e1	1.3	1.7	2.0	1.8
21701	1.1e3	1.6e2	6.2e1	5.4e1	1.6	2.1	2.5	2.2
38434	1.4e3	8.6e2	8.3e1	7.2e1	2.1	6.1	3.3	2.9
43640	$>$ 4hr	1.1e4	9.1e2	9.3e2	1.4e1	7.1e1	1.8e1	1.9e1
57975	$>$ 4hr	4.6e3	5.5e2	5.6e2	4.8	3.7e1	7.3	7.5
79841	8.8e3	9.8e2	3.2e2	2.8e2	9.5	9.8	1.3e1	1.1e1
114599	2.5e3	1.3e2	1.3e2	1.1e2	3.3	5.4	5.1	4.5

Table 4 shows the results of a similar experiment for the projection on the EDM completable cone (problem III) by solving (3.5), via the formulation

$$\begin{array}{ll} \text{minimize} & \|X - B\|_F^2 \\ \text{subject to} & P_{\beta_k} X P_{\beta_k}^T \in \mathbf{D}_0^{|\beta_k|},\ k = 1, \ldots, m, \\ & \operatorname{diag}(X) = 0, \end{array} \tag{6.3}$$

where $\beta_1, \ldots, \beta_m$ are the cliques in the chordal pattern E. The strictly lower-triangular nonzero values of B are assigned according to a uniform distribution in $[0, 1]$. The diagonal of B is set to zero.

In all cases when the method converged, the final objective values for the different algorithms are equal to two or three significant digits. (The unaccelerated projected gradient method in general took much longer than all other methods to converge, and the results are not included.) In general, the fast projected gradient method converged more slowly than the other methods. In all but four instances, the dual fast projected gradient method took between 200 and 1000 iterations and in two instances exceeded the time limit. In comparison, the dual block coordinate ascent and Douglas–Rachford algorithms took between 25 and 150 iterations to converge. (For the block coordinate ascent algorithm, we count one cycle through all l cones as one iteration.)

6.2. Nonchordal sparse matrix cones. In the next two sets of experiments we consider problems with nonchordal sparsity patterns. We first consider the projection on the PSD completable cone (problem (3.3)) and its dual

$$\begin{array}{ll} \text{maximize} & -\|S + B\|_F^2 + \|B\|_F^2 \\ \text{subject to} & S = \displaystyle\sum_{k=1}^{m} P_{\beta_k}^T Z_k P_{\beta_k}, \\ & S_{ij} = 0, \quad \forall \{i, j\} \in E' \setminus E, \\ & Z_k \succeq 0, \quad k = 1, \ldots, m. \end{array}$$

This also computes the projection on the PSD cone (problem (3.1)) of $A = -B$. We use the patterns listed in Table 1 as E and the chordal extensions listed in Table 2

TABLE 5

Projection on nonchordal PSD completable matrices. CPU times (in seconds) for the projection on $\Pi_E(\mathbf{S}^p_+)$. The Douglas–Rachford method is applied to the primal and dual problem form (P-DR and D-DR). The total runtimes and times per iteration are given.

p	Total runtime		Time/iteration	
	P-DR	D-DR	P-DR	D-DR
1601	3.5e1	2.8e1	1.6e-1	1.6e-1
4307	5.8e1	4.8e1	2.9e-1	2.9e-1
5357	4.4e2	3.8e2	1.6	1.7
6019	1.3e2	1.1e2	4.5e-1	4.8e-1
8204	4.1e2	3.6e2	1.3	1.3
12992	1.9e2	1.3e2	1.2	1.3
20141	2.6e2	2.1e2	1.3	1.4
21701	5.2e2	4.3e2	1.6	1.7
38434	2.4e2	1.7e2	2.4	2.3
43640	4.1e3	3.6e3	1.3e1	1.3e1
57975	1.4e3	1.2e3	5.6	5.3
79841	2.1e3	1.8e3	8.6	9.1
114599	7.1e2	4.5e2	3.7	3.6

as E'. For each sparsity pattern E, we consider five randomly generated matrices $B \in \mathbf{S}^p_E$, with lower-diagonal nonzero values chosen from a normal Gaussian distribution. The results for the primal and dual Douglas–Rachford methods (Algorithms 4 and 5) are given in Table 5.

Compared to the chordal problems (6.1), solving the nonchordal problems in general took more iterations (between 75 to 300 iterations). The final objective values for the primal and dual Douglas–Rachford methods are equal to around 2 or 3 significant digits.

In the next experiment we consider projections on EDM completable cones for a family of randomly generated sparsity graphs. An EDM $D \in \mathbf{D}^p$ is constructed by choosing p points uniformly in a three-dimensional cube. The sparsity pattern E is defined as $E = \{\{i,j\} \mid i \neq j,\ D_{ij} \leq R\}$, where R is a positive parameter. For some problems with $p \geq 10000$, this resulted in graphs that were not connected. However the largest connected component contained over 95% of the vertices, so we did not remove these instances.

Next we add noise to the elements of D to construct a matrix $B \in \mathbf{S}^p_E$:

$$B_{ij} = \begin{cases} D_{ij} + N_{ij}, & \{i,j\} \in E, \\ 0, & \text{otherwise}, \end{cases}$$

where for each i,j, the noise $N_{ij} = N_{ji}$ is drawn from a Gaussian distribution with mean zero and standard deviation 0.1. This type of problem is similar to the sensor network node localization problems studied in [3, 14, 40], where R represents the radio range of the sensors. We solve problem (6.3) using the Douglas–Rachford method for the primal formulation (Algorithm 4).

Table 6 gives the problem statistics and runtime results, averaged over five instances. In general, each problem converged after 200 to 400 iterations.

7. Conclusion. We have presented decomposition methods for projections on sparse PSD, PSD completable, and EDM completable cones. By combining clique decomposition theorems for chordal sparsity patterns and first-order convex optimization algorithms, we are able to solve large problems with sizes ranging from $p \sim 1000$ to 100000. As mentioned briefly in section 5, it is straightforward to extend the algo-

TABLE 6

Projection on the EDM completable cone, with randomly generated nonchordal patterns. Problem statistics and CPU runtimes (total time and time per iteration, in seconds) for the primal Douglas–Rachford method are given.

p	Range (R)	Density	Density (extens.)	avg. # cliques	Avg. Cclique size	Total runtime	Time/ iteration
1000	5.00e-2	3.67e-2	2.56e-1	46.2	122.8	1.5e2	6.9e-1
1000	1.00e-1	9.15e-2	4.39e-1	36.6	206.3	4.1e2	1.6
5000	1.00e-2	3.93e-3	6.89e-2	228.0	121.6	1.6e3	5.3
5000	2.00e-2	1.03e-2	1.58e-1	191.4	240.7	5.0e3	1.9e1
10000	5.00e-3	1.47e-3	3.20e-2	474.4	102.1	3.6e3	1.0e1
25000	2.00e-3	3.96e-4	9.22e-3	1277.4	71.3	6.4e3	1.8e1
50000	1.00e-3	1.48e-4	2.70e-3	2821.4	46.3	5.4e3	1.8e1
75000	6.67e-4	8.34e-5	1.17e-3	4779.4	32.9	3.7e3	1.6e1
100000	5.00e-4	5.56e-5	6.04e-4	7214.4	24.4	2.6e3	1.3e1

rithms, for example, to problems with a small set of linear equality constraints. This only requires an additional projection step on the affine set defined by the constraint equations.

The key feature of the algorithms is that they involve only small dense eigenvalue decompositions, corresponding to the cliques in the chordal extension of the sparsity pattern. To underscore the importance of this property, we briefly outline some alternative first-order methods that do not use the clique decomposition. The first problem (a projection on the sparse PSD cone) can be viewed as a projection on the intersection $\mathbf{S}_E^p \cap \mathbf{S}_+^p$ of two sets. One can apply Dykstra's algorithm and solve the problem by alternately projecting on the set of sparse matrices $\mathbf{S}_E^p$ and the dense PSD cone $\mathbf{S}_+^p$. Similarily, the projections on the PSD completable and EDM completable cones (problems II and III) can be viewed as minimizing a sum $f(X) + g(X)$, where

$$f(X) = \sum_{\{i,j\}\in E} (X_{ij} - A_{ij})^2, \qquad g(X) = \delta_{\mathcal{C}}(X),$$

where $\mathcal{C} = \mathbf{S}_+^p$ or $\mathbf{D}^p$. This can be solved using the Douglas–Rachford method, which for this choice of g will require projections on $\mathcal{C}$. Hence, at every iteration of these algorithms, a single eigenvalue decomposition of order p is needed. However, a full eigenvalue decomposition quickly becomes impractical for p greater than 10000. On the machine used for our experiments, a single dense eigenvalue decomposition of order $p = 20000$ takes 15 minutes to compute, and exceeds memory for $p \geq 50000$. Sparse eigenvalue decomposition methods also pose difficulties. Even when the initial and final matrix variables in the algorithms are sparse, the intermediate variables (and in particular, the arguments to the projections on $\mathbf{S}_+^p$ and $\mathbf{D}^p$) are dense and, unless the method has almost converged, have close to $p/2$ positive and negative eigenvalues. On the same machine, a single full matrix projection using a sparse eigenvalue decomposition took more than an hour for a problem of size $p = 8204$ and more than eight hours for a problem of size $p = 12992$. In comparison, the runtimes of the decomposition methods discussed in the paper depend less strongly on p and more on the sparsity pattern and density; a test problem of size $p = 21701$ converged in 3 minutes, using about 2 seconds per iteration. In none of the test problems were we close to running out of memory.

There are some interesting differences among the decomposition methods. The dual block coordinate ascent method (Dykstra's method) and the Douglas–Rachford method seem to converge in fewer iterations than the accelerated dual projection

method. One possible explanation is, as described in section 4.3, Dykstra's method can be interpreted as a block coordinate gradient projection method with step size 1. In comparison, the projected gradient method described in section 4.2 uses a step size $1/L$, which is usually much smaller than 1. We also noticed that the Douglas–Rachford methods are more general and can be applied to the problems with nonchordal sparsity patterns. However they converged more slowly on the test problems with nonchordal patterns. We did not observe a difference in efficiency between the primal and dual Douglas–Rachford methods. A general difficulty when applying the Douglas–Rachford algorithm is the sensitivity to the choice of the problem parameters t and ρ. We used the same fixed values for all the experiments, and it is possible that the performance can be further improved by tuning the parameters.

REFERENCES

[1] J. Agler, J. W. Helton, S. McCullough, and L. Rodman, *Positive semidefinite matrices with a given sparsity pattern*, Linear Algebra Appl., 107 (1988), pp. 101–149.

[2] S. Al-Homidan and M. AlQarni, *Structure methods for solving the nearest correlation matrix problem*, Positivity, 16 (2012), pp. 497–508.

[3] S. Al-Homidan and H. Wolkowicz, *Approximate and exact completion problems for Euclidean distance matrices using semidefinite programming*, Linear Algebra Appl., 406 (2005), pp. 109–141.

[4] C. M. Alaíz, F. Dinuzzo, and S. Sra, *Correlation matrix nearness and completion under observation uncertainty*, IMA J. Numer. Anal., 35 (2015), pp. 325–340.

[5] A. Y. Alfakih, A. Khandani, and H. Wolkowicz, *Solving Euclidean distance matrix completion problems via semidefinite programming*, Comput. Optim. Appl., 12 (1999), pp. 13–30.

[6] B. Alipanahi, N. Krislock, A. Ghodsi, H. Wolkowicz, L. Donaldson, and M. Li, *Determining protein structures from NOESY distance constraints by semidefinite programming*, J. Comput. Biol., 20 (2013), pp. 296–310.

[7] M. Bakonyi and C. R. Johnson, *The Euclidian distance matrix completion problem*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 646–654.

[8] H. H. Bauschke and P. L. Combettes, *Convex Analysis and Monotone Operator Theory in Hilbert Spaces*, Springer, New York, 2011.

[9] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.

[10] A. Beck and M. Teboulle, *A fast dual proximal gradient algorithm for convex minimization and applications*, Oper. Res. Lett., 42 (2014), pp. 1–6.

[11] A. Beck and L. Tetruashvili, *On the convergence of block coordinate descent type methods*, SIAM J. Optim., 23 (2013), pp. 2037–2060.

[12] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Belmont, MA, 1997.

[13] E. G. Birgin and M. Raydan, *Robust stopping criteria for Dykstra's algorithm*, SIAM J. Sci. Comput., 26 (2005), pp. 1405–1414.

[14] P. Biswas and Y. Ye, *Semidefinite programming for ad hoc wireless sensor network localization*, in Third International Symposium on Information Processing in Sensor Networks, IPSN'04, ACM, New York, 2004, pp. 46–54.

[15] J. R. S. Blair and B. Peyton, *An introduction to chordal graphs and clique trees*, in Graph Theory and Sparse Matrix Computation, A. George, J. R. Gilbert, and J. W. H. Liu, eds., Springer-Verlag, New York, 1993.

[16] R. Borsdorf, N. J. Higham, and M. Raydan, *Computing a nearest correlation matrix with factor structure*, SIAM J. Matrix Anal. Appl., 31 (2010), pp. 2603–2622.

[17] S. Boyd and L. Xiao, *Least-squares covariance matrix adjustment*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 532–546.

[18] J. P. Boyle and R. L. Dykstra, *A method for finding projections onto the intersection of convex sets in Hilbert spaces*, in Advances in Order Restricted Statistical Inference, R. Dykstra, T. Robertson, and F. T. Wright, eds., Lecture Notes in Statist. 37, Springer-Verlag, New York, 1986, pp. 28–47.

[19] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*, Numerical Math. Sci. Comput., Oxford University Press, New York, 1997.

[20] A. Chambolle and C. Dossal, *On the convergence of the iterates of the "Fast Iterative Shrinkage/Thresholding Algorithm"*. J. Optim. Theory Appl., 166 (2015), pp. 968–982.
[21] D. Davis and W. Yin, *Convergence Rate Analysis of Several Splitting Schemes*, preprint, arXiv:1406.4834, 2014.
[22] D. Davis and W. Yin, *Faster Convergence Rates of Relaxed Peaceman-Rachford and ADMM Under Regularity Assumptions*, preprint, arXiv:1407.5210, 2014.
[23] T. A. Davis, *Direct Methods for Sparse Linear Systems*, Fundam. Algorithms, SIAM, Philadelphia, 2006.
[24] T. A. Davis and Y. Hu, *The University of Florida sparse matrix collection*, ACM Trans. Math. Software, 38 (2011), 1.
[25] D. Drusvyatskiy, G. Pataki, and H. Wolkowicz, *Coordinate shadows of semidefinite and Euclidean distance matrices*, SIAM J. Optim., 25 (2015), pp. 1160–1178.
[26] R. L. Dykstra, *An algorithm for restricted least squares regression*, J. Amer. Statist. Assoc., 78 (1983), pp. 837–842.
[27] J. Eckstein and D. Bertsekas, *On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators*, Math. Program., 55 (1992), pp. 293–318.
[28] N. Gaffke and R. Mathar, *A cyclic projection algorithm via duality*, Metrika, 36 (1989), pp. 29–54.
[29] W. Glunt, T. L. Hayden, S. Hong, and J. Wells, *An alternating projection algorithm for computing the nearest Euclidean distance matrix*, SIAM J. Matrix Anal. Appl., 11 (1990), pp. 589–600.
[30] A. Griewank and P. L. Toint, *On the existence of convex decompositions of partially separable functions*, Math. Program., 28 (1984), pp. 25–49.
[31] R. Grone, C. R. Johnson, E. M. Sá, and H. Wolkowicz, *Positive definite completions of partial Hermitian matrices*, Linear Algebra Appl., 58 (1984), pp. 109–124.
[32] S.-P. Han, *A successive projection method*, Math. Program., 40 (1988), pp. 1–14.
[33] S.-P. Han and G. Lou, *A parallel algorithm for a class of convex programs*, SIAM J. Control Optim., 26 (1988), pp. 345–355.
[34] T. Hayden and J. Wells, *Approximation by matrices positive semidefinite on a subspace*, Linear Algebra Appl., 109 (1988), pp. 115–130.
[35] P. Heggernes, *Minimal triangulation of graphs: A survey*, Discrete Math., 306 (2006), pp. 297–317.
[36] D. Henrion and J. Malick, *Projection methods for conic feasibility problems: applications to polynomial sum-of-squares decompositions*, Optim. Methods Softw., 26 (2011), pp. 23–46.
[37] D. Henrion and J. Malick, *Projection methods in conic optimization*, in Handbook on Semidefinite, Conic and Polynomial Optimization, M. F. Anjos and J. B. Lasserre, eds., Springer, New York, 2012, pp. 565–600.
[38] N. Higham, *Computing a nearest symmetric positive semidefinite matrix*, Linear Algebra Appl., 103 (1988), pp. 103–118.
[39] N. Higham, *Computing the nearest correlation matrix—a problem from finance*, IMA J. Numer. Anal., 22 (2002), pp. 329–343.
[40] N. Krislock and H. Wolkowicz, *Euclidean distance matrices and applications*, in Handbook on Semidefinite, Conic and Polynomial Optimization, M. F. Angos and J. B. Lasserre, eds., Springer, New York, 2012, pp. 879–914.
[41] P. L. Lions and B. Mercier, *Splitting algorithms for the sum of two nonlinear operators*, SIAM J. Numer. Anal., 16 (1979), pp. 964–979.
[42] J. Malick, *A dual approach to semidefinite least-squares problems*, SIAM J. Matrix Anal. Appl., 26 (2004), pp. 272–284.
[43] J. J. Moreau, *Proximité et dualité dans un espace hilbertien*, Bull. Soc. Math. France, 93 (1965), pp. 273–299.
[44] Y. Nesterov, *A method of solving a convex programming problem with convergence rate* $O(1/k^2)$, Sov. Math. Dokl., 27 (1983), pp. 372–376.
[45] Y. Nesterov, *Introductory Lectures on Convex Optimization*, Kluwer Academic, Dordrecht, The Netherlands, 2004.
[46] B. T. Polyak, *Introduction to Optimization*, Optimization Software, New York, 1987.
[47] H. Qi and D. Sun, *A quadratically convergent Newton method for computing the nearest correlation matrix*, SIAM J. Matrix Anal. Appl., 28 (2006), pp. 360–385.
[48] H. Qi and D. Sun, *Correlation stress testing for value-at-risk: An unconstrained convex optimization approach*, Comput. Optim. Appl., 45 (2010), pp. 427–462.
[49] H. Qi and D. Sun, *An augmented Lagrangian dual approach for the H-weighted nearest correlation matrix problem*, IMA J. Numer. Anal., 31 (2011), pp. 491–511.
[50] H.-D. Qi, *A semismooth Newton method for the nearest Euclidean distance matrix problem*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 67–93.

[51] H.-D. Qi, N. Xiu, and X. Yuan, *A Lagrangian dual approach to the single-source localization problem*, IEEE Trans. Signal Process., 61 (2013), pp. 3815–3826.
[52] R. Rebonato and P. Jäckel, *The Most General Methodology to Create a Valid Correlation Matrix for Risk Management and Option Pricing Purposes*, Quantitative Research Centre of the NatWest Group, Royal Bank of Scotland, 1999.
[53] R. T. Rockafellar, *Convex Analysis*, 2nd ed., Princeton University Press, Princeton, NJ, 1970.
[54] D. J. Rose, *Triangulated graphs and the elimination process*, J. Math. Anal. Appl., 32 (1970), pp. 597–609.
[55] I. J. Schoenberg, *Remarks to Maurice Fréchet's article "Sur la définition axiomatique d'une classe d'espaces vectoriels distanciés applicables vectoriellement sur l'espace de Hilbert"*, Ann. Math., 36 (1935), pp. 724–732.
[56] J. E. Spingarn, *Partial inverse of a monotone operator*, Appl. Math. Optim., 10 (1983), pp. 247–265.
[57] J. E. Spingarn, *Applications of the method of partial inverses to convex programming: Decomposition*, Math. Program., 32 (1985), pp. 199–223.
[58] Y. Sun, M. S. Andersen, and L. Vandenberghe, *Decomposition in conic optimization with partially separable structure*, SIAM J. Optim., 24 (2014), pp. 873–897.
[59] M. W. Trosset, *Applications of Multidimensional Scaling to Molecular Conformation*, Technical report, Rice University, Houston, TX, 1997.
[60] P. Tseng, *Further applications of a splitting algorithm to decomposition in variational inequalities and convex programming*, Math. Program., 48 (1990), pp. 249–263.
[61] P. Tseng, *Applications of a splitting algorithm to decomposition in convex programming and variational inequalities*, SIAM J. Control Optim., 29 (1991), pp. 119–138.
[62] P. Tseng, *Dual coordinate ascent methods for non-strictly convex minimization*, Math. Program., 59 (1993), pp. 231–247.
[63] P. Tseng, *On Accelerated Proximal Gradient Methods for Convex-Concave Optimization*, manuscript, 2008.
[64] K. Wüthrich, *Protein structure determination in solution by nuclear magnetic resonance spectroscopy*, Science, 243 (1989), pp. 45–50.
[65] M. Yannakakis, *Computing the minimum fill-in is NP-complete*, SIAM J. Algebraic Discrete Meth., 2 (1981), pp. 77–79.
[66] F. W. Young, *Multidimensional Scaling: History, Theory, and Applications*, Psychology Press, Hoboken, NJ, 2013.