ANZIAM J. 55(2014), 299–326
doi:10.1017/S1446181114000145

DOUGLAS–RACHFORD FEASIBILITY METHODS FOR MATRIX COMPLETION PROBLEMS

FRANCISCO J. ARAGÓN ARTACHO[1,2], JONATHAN M. BORWEIN[3] and
MATTHEW K. TAM✉[3]

(Received 12 September, 2013; revised 2 April, 2014; first published online 25 July 2014)

Abstract

In this paper, we give general recommendations for successful application of the Douglas–Rachford reflection method to convex and nonconvex real matrix completion problems. These guidelines are demonstrated by various illustrative examples.

2010 *Mathematics subject classification*: primary 90C59; secondary 65K05, 47N10.

Keywords and phrases: Douglas–Rachford projections, reflections, matrix completion, feasibility problems, protein reconstruction, Hadamard matrices.

1. Introduction

Matrix completion may be posed as an inverse problem in which a matrix possessing certain properties is to be reconstructed knowing only a subset of its entries. A great many problems can be usefully cast within this framework [34, 37].

By encoding each of the properties which the matrix possesses along with its known entries as constraint sets, matrix completion can be cast as a *feasibility problem*. That is, it is reduced to the problem of finding a point contained in the intersection of a (finite) family of sets.

Projection algorithms comprise a class of general purpose iterative methods which are frequently used to solve feasibility problems [7]. At each step, these methods utilize the nearest point projection onto each of the individual constraint sets. The philosophy here is that it is simpler to consider each constraint separately (through their nearest point projections), rather than the intersection directly.

[1] Systems Biochemistry Group, Luxembourg Centre for Systems Biomedicine, University of Luxembourg, Campus Belval, L-4362 Esch-sur-Alzette, Luxembourg; e-mail: francisco.aragon@ua.es.
[2] Address when work was performed: CARMA Centre, University of Newcastle, Callaghan, NSW 2308, Australia; e-mail: jon.borwein@gmail.com.
[3] CARMA Centre, University of Newcastle, Callaghan, NSW 2308, Australia;
e-mail: matthew.k.tam@gmail.com.

Applied to closed convex sets in Euclidean space, the behaviour of projection algorithms is quite well understood. Moreover, their simplicity and ease of implementation have ensured continued popularity for successful applications in a variety of nonconvex optimization and reconstruction problems [3, 9, 10]. This popularity is despite the absence of sufficient theoretical justification, although useful beginnings exist [2, 15, 29]. In many of these settings the *Douglas–Rachford* method (see Section 2.1) has been observed, empirically, to perform better than other projection algorithms and hence will be the projection algorithm of choice for this paper. A striking example of its better behaviour is detailed in Section 4.3.

We do note that there are many other useful projection algorithms and many applicable variants. See, for example, the *method of cyclic projections* [5, 8], *Dykstra's method* [6, 12, 17], the *cyclic Douglas–Rachford method* [16; Borwein and Tam, "The cyclic Douglas–Rachford method for inconsistent feasibility problems", *J. Nonlinear Convex Anal.* (accepted March 2014)] and many references contained in these papers.

In a recent paper [3], the present authors observed that many successful nonconvex applications of the Douglas–Rachford method can be considered as matrix completion problems. The aim of this paper is to give general guidelines for successful application of the Douglas–Rachford method to a variety of (real) matrix reconstruction problems, both convex and nonconvex.

The remainder of the paper is organized as follows. In Section 2, we first describe what is known about the Douglas–Rachford method and then discuss our modelling philosophy. In Section 3, we consider several matrix completion problems in which all the constraint sets are convex: positive-semidefinite matrices, doubly stochastic matrices and Euclidean distance matrices; before discussing adjunction of noise. This is followed in Section 4 by a more detailed description of several classes in which some of the constraint sets are nonconvex. In the first two subsections, we first look at low-rank constraints and then at low-rank Euclidean distance problems. In Section 4.3, we present a detailed application by viewing protein reconstruction from nuclear magnetic resonance (NMR) data as a low-rank Euclidean distance problem. The final three subsections of Section 4 carefully consider Hadamard, skew-Hadamard and circulant-Hadamard matrix problems, respectively. We end with various concluding remarks in Section 5.

2. Preliminaries

Let E denote a finite-dimensional Hilbert space – a *Euclidean space*. We will typically be considering the Hilbert space of real $m \times n$ matrices whose inner product is given by

$$\langle A, B\rangle = \mathrm{tr}(A^T B),$$

where the superscript T denotes the transpose and $\mathrm{tr}(\cdot)$ the trace of an $n \times n$ square matrix. The induced norm is the *Frobenius norm* and can be expressed as

$$\|A\|_F = \sqrt{\mathrm{tr}(A^T A)} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}^2}.$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

A *partial (real) matrix* is an $m \times n$ array for which only entries in certain locations are known. A *completion* of the partial matrix $A = (a_{ij}) \in \mathbb{R}^{m\times n}$ is a matrix $B = (b_{ij}) \in \mathbb{R}^{m\times n}$ such that if a_{ij} is specified, then $b_{ij} = a_{ij}$. The problem of *(real) matrix completion* is the following: *given a partial matrix, find a completion having certain properties of interest (for example, positive semidefiniteness).*

Throughout this paper, we formulate matrix completion as a *feasibility problem*. That is,

$$\text{find } X \in \bigcap_{i=1}^{N} C_i \subseteq \mathbb{R}^{m\times n}. \tag{2.1}$$

Let A be the partial matrix to be completed. We take C_1 to be the set of all completions of A; the sets $C_2, \ldots, C_N$ are chosen such that their intersection has the properties of interest. In this case, (2.1) is precisely the problem of matrix completion.

2.1. The Douglas–Rachford method Recall that the *nearest point projection onto* $S \subseteq E$ is the set-valued mapping $P_S : E \to 2^S$ defined by

$$P_S x = \Big\{ s \in S \mid \|s - x\| = \inf_{y\in S} \|y - x\| \Big\}.$$

The *reflection with respect to* S is the set-valued mapping $R_S : E \to 2^E$ defined by

$$R_S = 2P_S - I,$$

where I denotes the identity map.

In an abuse of notation, if $P_S x$ (respectively $R_S x$) is a singleton, we use $P_S x$ (respectively $R_S x$) to denote the unique nearest point.

We now recall what is known about the Douglas–Rachford method, specialized to finite-dimensional spaces.

THEOREM 2.1 (Convex Douglas–Rachford iterations [11, Theorem 3.13]). *Suppose $A, B \subseteq E$ are closed and convex sets. For any $x_0 \in E$, define*

$$x_{n+1} = T_{A,B} x_n, \text{ where } T_{A,B} = \frac{I + R_B R_A}{2}.$$

Then one of the following holds:

(a) *$A \cap B \neq \emptyset$ and (x_n) converges to a point x such that $P_A x \in A \cap B$;*
(b) *$A \cap B = \emptyset$ and $\|x_n\| \to \infty$.*

The results of Theorem 2.1 can only be directly applied to the problem of finding a point in the intersection of two sets. For matrix completion problems formulated as feasibility problems with greater than two sets, we use a well-known *product-space reformulation.*

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

EXAMPLE 2.2 (Product-space reformulation). For constraint sets $C_1, C_2, \ldots, C_N$, define

$$D = \{(x, x, \ldots, x) \in E^N \mid x \in E\}, \quad C = \prod_{i=1}^{N} C_i.$$

The set D is sometimes called the *diagonal*. We now have an equivalent feasibility problem, since

$$x \in \bigcap_{i=1}^{N} C_i \Leftrightarrow (x, x, \ldots, x) \in D \cap C.$$

Moreover, $T_{D,C}$ can be readily computed whenever $P_{C_1}, P_{C_2}, \ldots, P_{C_N}$ can be, since

$$P_D x = \left(\frac{1}{N}\sum_{i=1}^{N} x_i\right)^N, \quad P_C x = \prod_{i=1}^{N} P_{C_i} x_i.$$

For further details see, for example, [3, Section 3].

There are some useful theoretical beginnings which explain the behaviour of the Douglas–Rachford method in certain nonconvex settings. For a Euclidean sphere and an affine subspace, with the reflection performed first with respect to the sphere, Borwein and Sims [15] showed that, appropriately viewed, the Douglas–Rachford scheme converges locally. An explicit region of convergence was given by Aragón Artacho and Borwein [2] for $\mathbb{R}^2$. Hesse and Luke [29] obtained local convergence results using a relaxed local version of firm nonexpansivity and appropriate regularity conditions, assuming that the reflection is performed first with respect to a subspace. We note that varying the order of the reflection does not make a substantive difference. In a recent preprint, Bauschke and Noll proved that the method is locally convergent, applied to constraints which are finite unions of convex sets.

2.2. Modelling philosophy As illustrated for Sudoku and other NP-complete combinatorial problems [3], there are typically many ways to formulate the constraint set for a given matrix completion problem; for example, by choosing different sets $C_2, C_3, \ldots, C_N$, in (2.1), such that $\bigcap_{i=2}^{N} C_i$ has the properties of interest. To apply the Douglas–Rachford method, these sets will be chosen in such a way that their individual nearest point projections are succinctly simple to compute – ideally in closed form. There is frequently a trade-off between the number of sets in the intersection and the simplicity of their projections. For example, one extreme would be to encode the property of interest in a single constraint set. In this case, it is likely that its projection is difficult to compute.

To illustrate this philosophy, consider the following example, which we revisit in Section 3.2. Suppose that the property of interest is the constraint

$$\left\{X \in \mathbb{R}^{m\times n} \,\middle|\, X_{ij} \geq 0, \sum_{k=1}^{n} X_{kj} = 1 \text{ for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n\right\}.$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

This set is equal to the intersection of C_2 and C_3, where

$$C_2 = \{X \in \mathbb{R}^{m\times n} \mid X_{ij} \geq 0 \text{ for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n\},$$
$$C_3 = \left\{X \in \mathbb{R}^{m\times n} \,\middle|\, \sum_{i=1}^{n} X_{ij} = 1 \text{ for } j = 1, \ldots, n\right\}.$$

Here the projections onto the cone C_2 and the affine space C_3 can be easily computed (see Section 3.2). In contrast, the projection directly onto $C_2 \cap C_3$ amounts to finding the nearest point in the convex hull of the set of matrices having a one in each row and remaining entries zero. This projection is less straightforward, and has no explicit form [20].

The order of the constraint sets in (2.1) also requires some consideration. For matrix completion problems with two constraints, we directly apply the Douglas–Rachford method to $C_1 \cap C_2$, with the reflection first performed with respect to the set C_1. For matrix completion problems with more than two constraints, we apply the Douglas–Rachford method to the product formulation of Example 2.2, with the reflection with respect to D performed first. In this case, the solution is obtained by projecting onto D and, thus, can be monitored by considering only a single product coordinate.

In nonconvex applications, the problem formulation chosen often determines whether or not the Douglas–Rachford scheme can successfully solve the problem at hand always, frequently or never [3]. Hence, in the rest of this paper we focus on *naive* or direct implementation of the Douglas–Rachford method while focusing on the choice of an appropriate model and the computation of the requisite projections or reflections. In a followup paper, we will look at more refined variants for our two capstone applications, that is, to protein reconstruction and to Hadamard matrix problems.

3. Convex problems

We now look, in order, at positive-definite matrices and correlation matrices, doubly stochastic matrices and Euclidean distance matrices, before discussing adjunction of noise.

3.1. Positive-semidefinite matrices Let S_n denote the set of real $n \times n$ symmetric matrices. Recall that a matrix $A = (A_{ij}) \in \mathbb{R}^{n\times n}$ is said to be *positive semidefinite* if

$$A \in S_n \quad \text{and} \quad x^T A x \geq 0 \text{ for all } x \in \mathbb{R}^n. \tag{3.1}$$

The set of all such matrices forms a closed convex cone [13, Exercise 1, Section 1.2] and is denoted by S^n_+. The *Loewner partial order* is defined on S_n by taking $A \geq B$ if $A - B \in S^n_+$. Recall that a symmetric matrix is positive semidefinite if and only if all its eigenvalues are nonnegative.

Let us consider the matrix completion problem where only some entries of the positive-semidefinite matrix A are known, and denote by Ω the location of these entries

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

(that is, $(i, j) \in \Omega$ if A_{ij} is known). Without loss of generality, we may assume that Ω is symmetric in the sense that $(i, j) \in \Omega$ if and only if $(j, i) \in \Omega$. Consider the convex sets

$$C_1 = \{X \in \mathbb{R}^{n\times n} \mid X_{ij} = A_{ij} \text{ for all } (i, j) \in \Omega\}, \quad C_2 = S_n^+. \tag{3.2}$$

Then X is a positive-semidefinite matrix that completes A if and only if $X \in C_1 \cap C_2$.

The set C_1 is a closed affine subspace. Its projection is straightforward and given pointwise by

$$P_{C_1}(X)_{ij} = \begin{cases} A_{ij} & \text{if } (i, j) \in \Omega, \\ X_{ij} & \text{otherwise} \end{cases} \tag{3.3}$$

for all $i, j = 1, \dots, n$.

Theorem 3.1 [30, Theorem 2.1]. *Let $X \in \mathbb{R}^{n\times n}$. Define $Y = (A + A^T)/2$ and let $Y = UP$ be a polar decomposition* [30, Theorem 1.1]. *Then*

$$P_{C_2}(X) = \frac{Y + P}{2}. \tag{3.4}$$

Remark 3.2. For $X \in S_n$, $Y = X$ in the statement of Theorem 3.1. If this is the case, the computation of P_{C_2} is also simplified.

If the initial matrix is symmetric, then the corresponding Douglas–Rachford iterates are, too. This condition can be easily satisfied. For instance, if $X \in \mathbb{R}^{n\times n}$, then the iterates can instead be computed starting at $P_{S_n}(X) = (X + X^T)/2$ or $XX^T \in S_n$.

Of course, for symmetric iterates only the upper (or lower) triangular matrix needs to be computed.

Remark 3.3. The matrices U and P can also be easily obtained from a *singular value decomposition* [33, p. 205], because, if $Y = WSV^T$ is a singular value decomposition, then

$$P = VSV^T, \quad U = WV^T$$

produce P and U.

Remark 3.4. (Positive-definite matrices) Recall that a real symmetric $n \times n$ matrix is said to be *positive definite* if the inequality in (3.1) holds strictly whenever $x \neq 0$. Denote the set of all such matrices by S^n_{++}. Since S^n_{++} is not closed, the problem of positive-definite matrix completion cannot be directly cast within this framework by setting $C_2 = S^n_{++}$.

In practice, one might wish to consider a closed convex subset of S^n_{++}. For example, one could instead define

$$C_2 = \{X \in \mathbb{R}^{n\times n} \mid X^T = X, x^T X x \geq \epsilon \|x\|^2 \text{ for all } x \in \mathbb{R}^n\} \tag{3.5}$$

for some small $\epsilon > 0$. Then (3.5) is equivalent to requiring that the eigenvalues be not less than ϵ.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

One can apply these methods to find semidefinite solutions to matrix equations. For example, we have considered the problem

$$\text{find } X \in S_n^+ \text{ such that } AX = B,$$

where $A, B \in \mathbb{R}^{m\times n}$.

By taking the constraint sets to be the affine subspace and the positive-semidefinite matrices, we obtain a two-set feasibility problem. Test instances were generated by choosing random $A \in \mathbb{R}^{m\times n}$ and setting $B = AP_{S_n^+}(Y)$ for random $Y \in \mathbb{R}^{m\times n}$. Using the Douglas–Rachford method, solutions could be found in 17 s for $m = 50$, $n = 100$ and within 5 min for $m = 100$, $n = 500$.

It would also be interesting to incorporate these ideas in finding semidefinite solutions to matrix Riccati equations, as discussed by Ammar et al. [1].

3.1.1. *Correlation matrices* An important class of positive-semidefinite matrices is the *correlation matrices*. Given random variables $Z_1, Z_2, \ldots, Z_n$, the associated correlation matrix is an element of $[-1, 1]^{n\times n}$ whose (i, j)th entry is the correlation between variables Z_i and Z_j. Since any random variable perfectly correlates with itself, the entries along the main diagonal of any correlation matrix are all ones. Consequently,

$$\{(i,i) \mid i = 1, \ldots, n\} \subseteq \Omega, \text{ and } A_{ii} = 1 \quad \text{for } i = 1, \ldots, n. \tag{3.6}$$

Moreover, whenever (3.6) holds, A is necessarily contained in $[-1, 1]^{n\times n}$. This is a consequence of the following inequality.

Proposition 3.5 [33, p. 398]. *Let $A = (a_{ij}) \in S_+^n$. Then*

$$a_{ii}a_{jj} \geq a_{ij}^2.$$

Thus, if A is an incomplete correlation matrix, without loss of generality we may assume that (3.6) holds. In this case, X is the correlation matrix that completes A if and only if $X \in C_1 \cap C_2$, as defined in (3.2).

Consider now the problem of generating a random sample of correlation matrices. This is the case, for example, when one uses simulation to determine an unknown probability distribution [40].

The Douglas–Rachford method provides a heuristic for generating such a sample by applying the method to initial points chosen according to some probability distribution. In this case, the set of known indices, and their corresponding values, are

$$\Omega = \{(i,i) \mid i = 1, \ldots, n\}, \text{ and } A_{ii} = 1 \text{ for } i = 1, \ldots, n.$$

The distribution of the entries in 100 000 matrices of size 5 × 5, obtained from three different sets of choices of initial point distribution, is shown in Figure 1.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

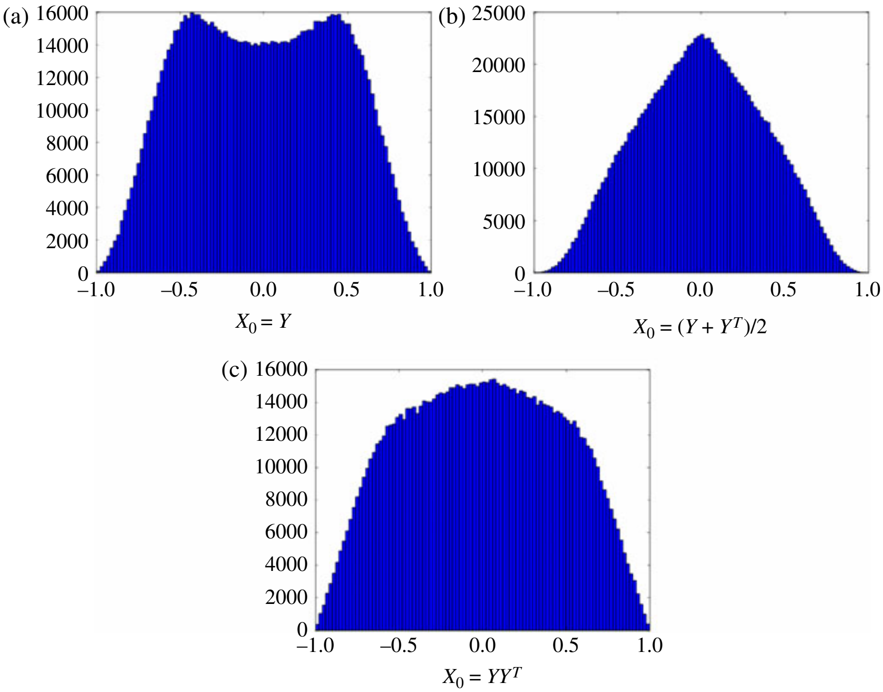


FIGURE 1. Distribution of entries in the collections of correlation matrices generated by different initializations of the Douglas–Rachford algorithm. The initial point is X_0 and Y is a random matrix in $[-1, 1]^{5\times5}$. Note that $(Y + Y^T)/2, YY^T \in S_5$.

3.2. Doubly stochastic matrices Recall that a matrix $A = (A_{ij}) \in \mathbb{R}^{m\times n}$ is said to be *doubly stochastic* if

$$\sum_{i=1}^{m} A_{ij} = \sum_{j=1}^{n} A_{ij} = 1, A_{ij} \geq 0 \quad \text{for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n. \tag{3.7}$$

The set of all doubly stochastic matrices is known as the *Birkhoff polytope*, and can be realized as the convex hull of the set of permutation matrices [13, Theorem 1.25].

Let us now consider the matrix completion problem, where only some entries of a doubly stochastic matrix A are known, and denote by Ω the location of these entries (that is, $(i, j) \in \Omega$ if A_{ij} is known). The set of all such candidates is given by

$$C_1 = \{X \in \mathbb{R}^{m\times n} \mid X_{ij} = A_{ij} \text{ for all } (i, j) \in \Omega\}, \tag{3.8}$$

which is similar to (3.2). The Birkhoff polytope may be expressed as the intersection of the three convex sets

$$C_2 = \left\{X \in \mathbb{R}^{m\times n} \,\middle|\, \sum_{i=1}^{m} X_{ij} = 1 \text{ for } j = 1, \ldots, n\right\}, \tag{3.9}$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

$$C_3 = \Big\{X \in \mathbb{R}^{m\times n} \;\Big|\; \sum_{j=1}^{n} X_{ij} = 1 \text{ for } i = 1, \ldots, m\Big\}, \tag{3.10}$$

$$C_4 = \{X \in \mathbb{R}^{m\times n} \mid X_{ij} \geq 0 \text{ for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n\}. \tag{3.11}$$

Then X is a double stochastic matrix that completes A if and only if $X \in C_1 \cap C_2 \cap C_3 \cap C_4$.

As in (3.3), the set C_1 is a closed affine subspace whose projection is given pointwise by

$$P_{C_1}(X)_{ij} = \begin{cases} A_{ij} & \text{if } (i, j) \in \Omega, \\ X_{ij} & \text{otherwise} \end{cases} \tag{3.12}$$

for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

The projection onto C_2 (respectively, C_3) can be easily computed by applying the following proposition row-wise (respectively, column-wise).

Proposition 3.6. *Let* $S = \{x \in \mathbb{R}^m \mid \sum_{i=1}^m x_i = 1\}$. *For any* $x \in \mathbb{R}^m$,

$$P_S(x) = x + \frac{1}{m}\Big(1 - \sum_{i=1}^{m} x_i\Big)e, \quad \text{where } e = [1, 1, \ldots, 1]^T.$$

Proof. Since $S = \{x \in \mathbb{R}^n \mid \langle e, x\rangle = 1\}$, the result follows from the standard formula for the orthogonal projection onto a hyperplane [23, Section 4.2.1]. □

The projection of A onto C_4 is given pointwise by

$$P_{C_4}(A)_{ij} = \max\{0, A_{ij}\}$$

for $i = 1, \ldots, m$ and $j = 1, \ldots, n$.

Remark 3.7. One can also address the problem of singly stochastic matrix completion. The problem of row (respectively, column) stochastic matrix completion is formulated by dropping the constraint C_3 (respectively, C_2).

Finally, let us mention a related work, where Takouda [43] applied Dykstra's algorithm to find the closest square doubly stochastic matrix to a given one in $\mathbb{R}^{n\times n}$ by considering the intersection of the two sets $C_2 \cap C_3$ and C_4. In particular, he showed that $P_{C_2\cap C_3}(X) = WXW + J$, where $W = I - J$ and $J = ee^T/n$ [43, Proposition 4.4].

3.3. Euclidean distance matrices

A matrix $D = (D_{ij}) \in \mathbb{R}^{n\times n}$ is said to be a *distance matrix* if

$$D_{ij} = D_{ji} \begin{cases} = 0 & i = j \\ \geq 0 & i \neq j \end{cases} \quad \text{for } i, j = 1, \ldots, n.$$

Furthermore, D is called a *Euclidean distance matrix* (EDM) if there are points $p_1, \ldots, p_n \in \mathbb{R}^r$ (with $r \leq n$) such that

$$D_{ij} = \|p_i - p_j\|^2 \quad \text{for } i, j = 1, \ldots, n. \tag{3.13}$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

If (3.13) holds for a set of points in $\mathbb{R}^r$, then D is said to be *embeddable* in $\mathbb{R}^r$. If D is embeddable in $\mathbb{R}^r$ but not in $\mathbb{R}^{r-1}$, then it is said to be *irreducibly embeddable* in $\mathbb{R}^r$.

The following result by Hayden and Wells, based on Schoenberg's criterion [41, Theorem 1], provides a useful characterization of Euclidean distance matrices.

Theorem 3.8. [28, Theorem 3.3]. *Let Q be the Householder matrix defined by*

$$Q = I - \frac{2vv^T}{v^Tv}, \quad \text{where } v = [1, 1, \ldots, 1, 1 + \sqrt{n}]^T \in \mathbb{R}^n.$$

Then a distance matrix D is a Euclidean distance matrix if and only if the $(n-1) \times (n-1)$ block $\widehat{D}$ in

$$Q(-D)Q = \begin{bmatrix} \widehat{D} & d \\ d^T & \delta \end{bmatrix} \tag{3.14}$$

is positive semidefinite. In this case, D is irreducibly embeddable in $\mathbb{R}^r$, where $r = \operatorname{rank}(\widehat{D}) \leq n-1$.

Remark 3.9. As a consequence of Theorem 3.8, the set of Euclidean distance matrices is convex.

Let us consider now the matrix completion problem where only some entries of a Euclidean distance matrix D are known, and denote by Ω the location of these entries (that is, $(i, j) \in \Omega$ if D_{ij} is known). Without loss of generality, we assume D and Ω to be symmetric. Consider the convex sets

$$C_1 = \{X \in \mathbb{R}^{n\times n} \mid X \text{ is a distance matrix}, X_{ij} = D_{ij} \text{ for all } (i, j) \in \Omega\}, \tag{3.15}$$

$$C_2 = \{X \in \mathbb{R}^{n\times n} \mid \widehat{X} \succeq 0, \text{ where } \widehat{X} \text{ is the block in } Q(-X)Q \text{ in (3.14)}\}. \tag{3.16}$$

Then X is a Euclidean distance matrix that completes D if and only if $X \in C_1 \cap C_2$.

The projection of any symmetric matrix $A = (A_{ij}) \in \mathbb{R}^{n\times n}$ onto C_1 can be easily computed as

$$P_{C_1}(A) = \begin{cases} 0 & \text{if } i = j, \\ D_{ij} & \text{if } (i, j) \in \Omega, \\ \max\{0, A_{ij}\} & \text{otherwise.} \end{cases} \tag{3.17}$$

The projection of A onto C_2 is the unique solution to the problem

$$\min_{X \in C_2} \|A - X\|_F.$$

Using (3.14), if we denote

$$Q(-A)Q = \begin{bmatrix} \widehat{A} & a \\ a^T & \alpha \end{bmatrix} \quad \text{and} \quad Q(-X)Q = \begin{bmatrix} \widehat{X} & x \\ x^T & \lambda \end{bmatrix},$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

then

$$\min_{X\in C_2}\|A-X\|_F=\min_{X\in C_2}\|Q(A-X)Q\|_F=\min_{X\in C_2}\|Q(-A)Q-Q(-X)Q\|_F$$
$$=\min_{\substack{x\in\mathbb{R}^n,\lambda\in\mathbb{R}\\ \widehat{X}=\widehat{X}^T,\widehat{X}\succeq 0}}\left\|\begin{matrix}\widehat{A}-\widehat{X} & a-x\\ (a-x)^T & (\alpha-\lambda)\end{matrix}\right\|_F.$$

As a consequence of a result of Hayden and Wells [28, Theorem 2.1], the unique best approximation is given by

$$\begin{bmatrix}U\Lambda_+U^T & a\\ a^T & \alpha\end{bmatrix},$$

where $U\Lambda U^T=\widehat{A}$ is the spectral decomposition [28, p. 116] of $\widehat{A}$, with $\Lambda=\operatorname{diag}(\lambda_1,\ldots,\lambda_{n-1})$ and $\Lambda_+=\operatorname{diag}(\max\{0,\lambda_1\},\ldots,\max\{0,\lambda_{n-1}\})$. Therefore,

$$P_{C_2}(A)=-Q\begin{bmatrix}U\Lambda_+U^T & a\\ a^T & \alpha\end{bmatrix}Q. \tag{3.18}$$

Remark 3.10. The problem of finding the nearest EDM to a given matrix is related to the problem of EDM completion. The former was considered by Glunt et al. [27] who gave Dykstra's method approach, based on the same EDM characterization used here.

3.3.1. *Noise* In many practical situations, the distances that are initially known have some errors in their measurements, and the Euclidean matrix completion problem may not even have a solution. In these situations, a model that allows errors in the distances needs to be considered.

Given some error $\varepsilon\geq 0$, consider the convex set

$$C_1^\varepsilon=\{X\in\mathbb{R}^{n\times n}\mid X \text{ is a distance matrix and } |X_{ij}-D_{ij}|\leq\varepsilon \text{ for all } (i,j)\in\Omega\}. \tag{3.19}$$

Notice that $C_1^0=C_1$. The projection of any symmetric matrix $A=(A_{ij})\in\mathbb{R}^{n\times n}$ onto C_1^ε can be easily computed as

$$P_{C_1^\varepsilon}(A)_{ij}=\begin{cases}0 & \text{if } i=j,\\ D_{ij}+\varepsilon & \text{if } (i,j)\in\Omega \text{ and } A_{ij}>D_{ij}+\varepsilon,\\ \max\{0,D_{ij}-\varepsilon\} & \text{if } (i,j)\in\Omega \text{ and } A_{ij}<D_{ij}-\varepsilon,\\ \max\{0,A_{ij}\} & \text{otherwise.}\end{cases} \tag{3.20}$$

This model could be easily modified to include different upper and lower bounds on each distance D_{ij} for $(i,j)\in\Omega$.

4. Nonconvex problems

We now turn to the more difficult case of nonconvex matrix completion problems.

4.1. Low-rank matrices In many practical scenarios, one would like to recover a matrix that is known to be low rank from only a subset of its entries. This is the case,

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

for example, in various compressed sensing applications [14]. The main problem here is that the low-rank constraint makes the problem nonconvex. For example, if we consider

$$S = \{A \in \mathbb{R}^{2\times 2} \mid \operatorname{rank}(A) \le 1\},$$

then

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \in S,$$

but, for all $\lambda \in (0, 1)$,

$$\lambda \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + (1-\lambda) \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \notin S.$$

4.1.1. *Rank constraints* Let us consider the problem of finding a matrix of minimal rank, given that some of the entries are known. For given r, we define a *rank constraint*

$$C_2^r = \{X \in \mathbb{R}^{m\times n} \mid \operatorname{rank}(X) \le r\}.$$

Then X is a matrix completion of A with rank at most r if and only if $X \in C_1 \cap C_2^r$.

The set of possible ranks of A is finite and bounded above by $\min\{m, n\}$. Furthermore, $C_2^r \subseteq C_2^s$ for $r \le s$. It follows that X is a completion of A with minimal rank if and only if

$$X \in C_1 \cap C_2^{r_0} \quad \text{and} \quad X \notin C_1 \cap C_2^r \text{ for any } r < r_0.$$

In this case, $\operatorname{rank}(X) = r_0$.

This suggests a *binary search* heuristic for finding the rank of a matrix. For convenience, denote by $P^{(r)}$ the relaxation

$$\text{find } x \in C_1 \cap C_2^r. \tag{4.1}$$

The iteration can now be implemented as shown in Algorithm 1.

Algorithm 1: Heuristic for minimum-rank matrix completion.

```
input  : $\Omega$, $A_{ij}$ for all $(i, j) \in \Omega$, MaxIterations
$r_{lb} = 0, r_{ub} = \min\{m, n\}$ and choose any $r \in [r_{lb}, r_{ub}] \cap \mathbb{N}$;
while $r_{lb} < r_{ub}$ do
    if Douglas–Rachford method solves $P^{(r)}$ within MaxIterations iterations then
        $r_{ub} = r$;
    else
        $r_{lb} = r + 1$;
    end
    $r = \lfloor (r_{lb} + r_{ub})/2 \rfloor$;
end
output: $r$
```

Of course there are many applicable variants on this idea. For instance, one could instead perform a ternary search.

Remark 4.1 (Connection to minimum-rank completion). A problem closely related to (4.1) is the *minimum-rank matrix completion problem*. That is, the optimization problem

$$\min\{\operatorname{rank}(X) \mid X \in C_1\}. \tag{4.2}$$

Indeed, (4.1) is a nonconvex relaxation of (4.2) in which one seeks a matrix with rank less than some prescribed upper bound rather than a matrix having smallest rank.

In the literature, another popular relaxation of (4.2) is the so-called *nuclear-norm relaxation* [19], which is given by

$$\min\{\|X\|_* \mid X \in C_1\}. \tag{4.3}$$

Here $\|X\|_*$ denotes the *nuclear norm*, whose value is given by the sum of the singular values of the matrix X. Since the nuclear norm is a convex function, this relaxation is a convex optimization problem.

Some comments regarding the two different relaxations are in order. Since nuclear-norm relaxation is convex, it can be solved by established convex optimization tools (the most appropriate tool depends on various factors including the dimension of the matrix to be recovered). Algorithms specific to this relaxation also exist [18]. While this approach does not require an *a priori* rank estimate on the matrix, rank minimization is achieved by replacing the objective function with the nuclear norm – a surrogate objective function.

The Douglas–Rachford approach directly solves a nonconvex feasibility problem in which the rank objective is incorporated as a constraint, thus staying closer to the original formulation of (4.2). While this method does require an *a priori* rank estimate, initial overestimates are not necessarily detrimental to the reconstruction. This is because the Douglas–Rachford algorithm can yield a matrix $X^* \in C_1 \cap C_2^r$ having $\operatorname{rank}(X) < r$.

4.2. Low-rank Euclidean distance matrices

In many situations, the Euclidean distance matrix D that one aims to complete is known to be embeddable in $\mathbb{R}^r$, say with $r = 3$. This is the case, for example, in the molecular conformation problem in which one would like to compute the relative atomic positions within a molecule. Nuclear magnetic resonance (NMR) spectroscopy can be employed to measure short-range interatomic distances (that is, those less than 5–6 Å where 1 Å = 10^{-10} m stands for *Ångström*) without structural modification of the molecule [44].

These types of problems are known as *low-rank Euclidean distance matrix* problems. For any given positive integer r, we can modify the set C_2 in (3.16) as follows:

$$C_2^r = \{X \in \mathbb{R}^{n\times n} \mid \widehat{X} \succeq 0, \text{ where } \widehat{X} \text{ is the block in } Q(-X)Q \text{ in (3.14) and } \operatorname{rank}(\widehat{X}) \le r\}.$$

Unfortunately, as noted in [25, Section 5.3], the set C_2^r is no longer convex unless $r \geq n-1$, in which case the rank condition is always satisfied and $C_2^r = C_2$. Nevertheless, a projection of any symmetric matrix A onto C_2^r can be easily computed (since C_2^r is not convex, the projection need not be unique). Indeed, let us assume without loss of generality that the eigenvalues $\lambda_1, \ldots, \lambda_{n-1}$ of the submatrix $\widehat{X}$ are given in nondecreasing order $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{n-1}$ in the spectral decomposition $\widehat{X} = U\Lambda U^T$, where $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_{n-1})$. Then $P_{C_2^r}(X)$ can be computed as in (3.18) but with Λ_+ replaced by

$$\Lambda_+^r = \operatorname{diag}(0, \ldots, 0, \max\{0, \lambda_{n-r}\}, \ldots, \max\{0, \lambda_{n-1}\}).$$

4.3. Protein reconstruction Once more, despite the absence of convexity in one of the constraints, we have observed the Douglas–Rachford algorithm to converge. Computational experiments have been performed on various protein molecules obtained from the *Research Collaboratory for Structural Bioinformatics (RCSB) Protein Data Bank* (available at http://www.rcsb.org/pdb/). The complete structure of these proteins is contained in the respective data files as a list of points in $\mathbb{R}^3$, each representing an individual atom. The corresponding complete Euclidean distance matrix is computed using (3.13). A realistic partial Euclidean distance matrix is then obtained by removing all entries which correspond to distances greater than 6 Å. From this partial matrix, we seek to reconstruct the molecular conformation.

In Algorithm 2, we give details regarding our *Python* implementation for finding the distance matrix and, in Algorithm 3, we reconstruct the positions from the matrix completion.

Algorithm 2: Douglas–Rachford component of our Python implementation.

```
input   : $D \in \mathbb{R}^{n\times n}$ (the partial Euclidean distance matrix)
$\epsilon = 0.1,\ r = 3,\ N = 5000,\ k = 0$;
$X = (Y + Y^T)/2 \in S_n$ for random $Y \in [-1, 1]^{n\times n}$;
while $k \leq N$ do
    $X = T_{C_1^\epsilon, C_2^r} X$;
    $k = k + 1$;
end
output  : $X$ (the reconstructed Euclidean distance matrix).
```

Algorithm 3: Converting a Euclidean distance matrix to points in $\mathbb{R}^q$ [21, Section 5.12].

```
input : $X \in \mathbb{R}^{n\times n}$ (the reconstructed distance matrix)
$L = I - ee^T/n$, where $e = (1, 1, \ldots, 1)^T$;
$\tau = -(1/2)LDL$;
$USV^T = SingularValueDecomposition(\tau)$;
$Z =$ first $q$ columns of $U\sqrt{S}$;
$p_i = i$th row of $Z$ for $i = 1, 2, \ldots, n$;
output: $p_1, p_2, \ldots, p_n$ (positions of the points in $\mathbb{R}^q$).
```

TABLE 1. Six proteins: average (maximum) errors from five replications (5000 iterations).

Protein	#Atoms	Relative error (dB)	RMSE	Max error
1PTQ	404	–83.6 (–83.7)	0.0200 (0.0219)	0.0802 (0.0923)
1HOE	581	–72.7 (–69.3)	0.191 (0.257)	2.88 (5.49)
1LFB	641	–47.6 (–45.3)	3.24 (3.53)	21.7 (24.0)
1PHT	988	–60.5 (–58.1)	1.03 (1.18)	12.7 (13.8)
1POA	1067	–49.3 (–48.1)	34.1 (34.3)	81.9 (87.6)
1AX8	1074	–46.7 (–43.5)	9.69 (10.36)	58.6 (62.6)

The quality of the solution is then assessed using various error measurements. The *relative error*, reported in *decibels* (dB), is given by

$$\text{Relative error} = 10\log_{10}\left(\frac{\|P_{C_2^r}P_{C_1^\epsilon}X_N - P_{C_1^\epsilon}X_N\|_F^2}{\|P_{C_1^\epsilon}X_N\|_F^2}\right), \quad \text{where } \epsilon = 0.1, r = 3.$$

Let $p_1, p_2, \ldots, p_n \in \mathbb{R}^3$ denote the positions of the n atoms obtained from the distance matrix X_N, and let $p_1^{\text{true}}, p_2^{\text{true}}, \ldots, p_n^{\text{true}}$ denote the true positions of the n atoms (both relative to the same coordinate system). It is possible for both sets of points to represent the same molecular conformation without occupying the same positions in space. Thus, to compare the two sets, a *Procrustes analysis* is performed. That is, we (collectively) translate, rotate and reflect the points $p_1, p_2, \ldots, p_n$ to obtain the points $\hat{p}_1, \hat{p}_2, \ldots, \hat{p}_n$ which minimize the least squared error to the true positions.

Using the fitted points, we compute the *root-mean-square error (RMSE)* defined by

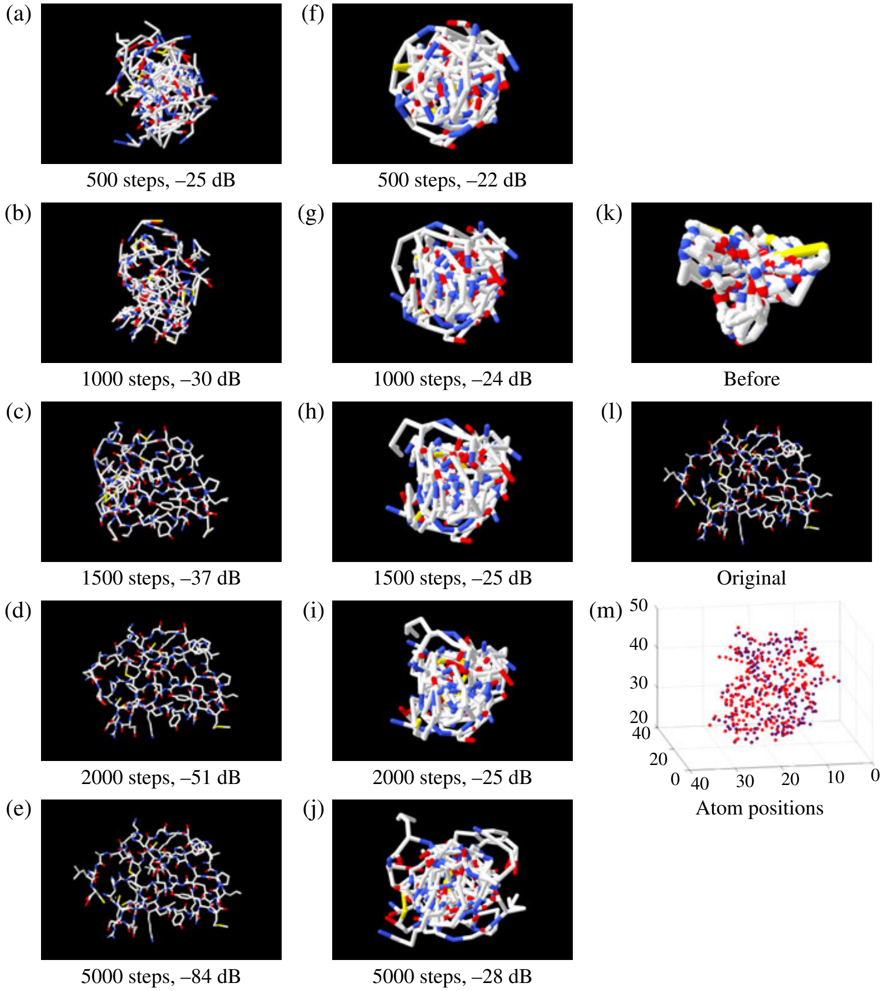
$$\text{RMSE} = \sqrt{\frac{1}{\text{number of atoms}}\sum_{i=1}^{m}\|\hat{p}_i - p_i^{\text{true}}\|_2^2}$$

and the *maximum error* defined by

$$\text{Max error} = \max_{1\le i\le m}\|\hat{p}_i - p_i^{\text{true}}\|_2.$$

Our computational results are summarized in Table 1. An animation of the algorithm at work constructing the protein 1PTQ can be viewed at http://carma.newcastle.edu.au/DRmethods/1PTQ.html. Next we make some general comments regarding the performance of our method.

- 1PTQ and 1HOE, the two proteins with fewer than 600 atoms, could be reliably reconstructed to within a small error. A visual comparison of the reconstructed and original molecules shows that they match well; they are indistinguishable. See Figures 2 and 3.
- The reconstructions of 1LFB and 1PHT, the next two smallest proteins examined, were both satisfactory although not as good as their smaller counterparts. A careful comparison of the original and reconstructed images in Figure 3 shows that a large proportion of the proteins have been faithfully



500 steps, –25 dB

500 steps, –22 dB

1000 steps, –30 dB

1000 steps, –24 dB

Before

1500 steps, –37 dB

1500 steps, –25 dB

Original

2000 steps, –51 dB

2000 steps, –25 dB

Atom positions

5000 steps, –84 dB

5000 steps, –28 dB

FIGURE 2. (Colour available online) Reconstructions (displayed in Swiss-PdbViewer) of the protein 1PTQ obtained from the Douglas–Rachford algorithm (a)–(e) and from the method of cyclic projections (f)–(j), together with their relative errors after given numbers of steps. The protein prior to the reconstruction is shown in (k) and its actual structure in (l). Only interatomic distances below 6 Å have been used as input. This represents 14,370/162,812 distances (8.83% of the nonzero entries of the distance matrix). Entry (m) shows the positions of the original (respectively, reconstructed) atoms in red (respectively, blue) – coincidence is frequent.

reconstructed, although some finer details are missing. For instance, one should look at the top right corners of the 1PHT images.

- The reconstructions of 1POA and 1AX8, the largest two proteins examined, were poor. The images of the reconstructed proteins show that some bond lengths are abnormally large. We discuss possible approaches to this issue in Remarks 4.2 and 4.3.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

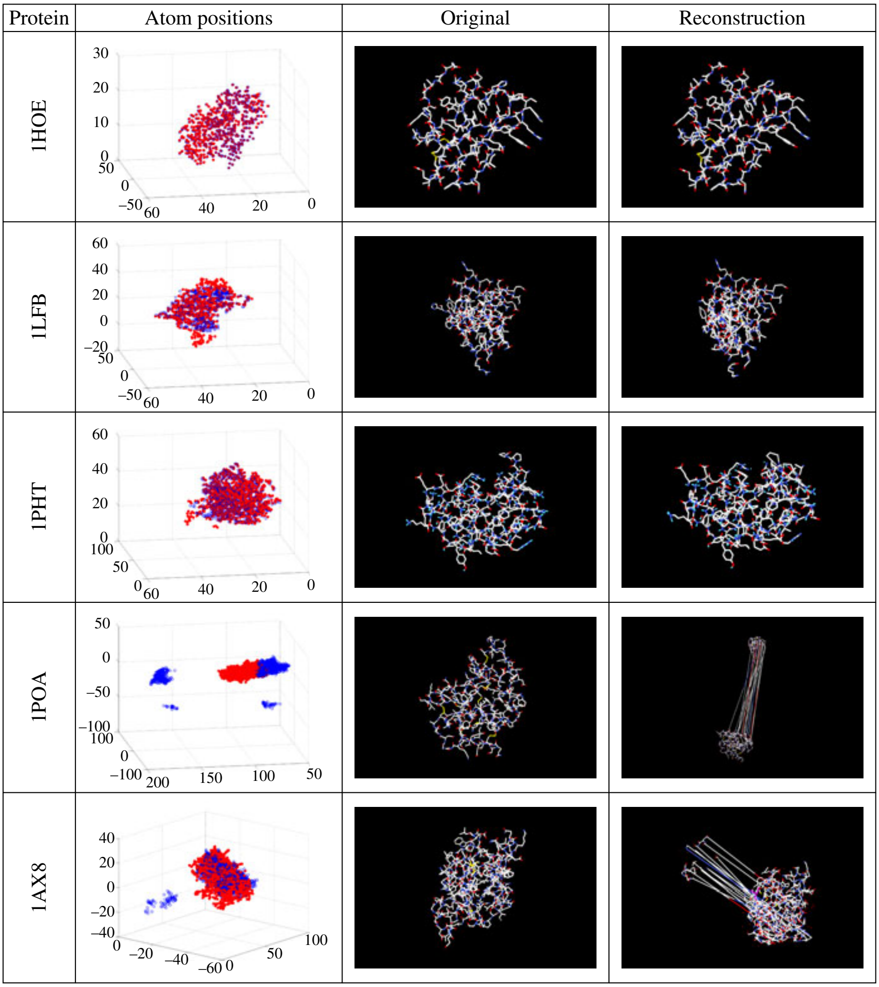


Figure 3. (Colour available online) The five proteins not shown in Figure 2. The first column shows positions of original (respectively, reconstructed) atoms in red (respectively, blue), the second and third columns show the original protein and a reconstructed instance (displayed in Swiss-PdbViewer), as reported in Table 1.

- Some alternative approaches to protein reconstruction are reported in [24]. Three of them are:
  - a 'build-up' algorithm placing atoms sequentially (*Buildup*),
  - a classical multidimensional scaling approach (*CMDSCALE*),
  - global continuation on Gaussian smoothing of the error function (*DGSOL*).

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

For 1PTQ and HOE, the RMS error of the Douglas–Rachford reconstruction was slightly smaller than the reconstruction obtained from either the buildup algorithm or CMDSCALE. For 1LFB and 1PHT, the RMS errors were comparable, and for 1POA and 1AXE they still had the same order of magnitude. DGSOL performed better than all three approaches (Douglas–Rachford, Buildup and CMSCALE).

- For the proteins examined, computational times for the full 5000 iterations, except for 1POA, ranged from 6 to 18 h. This time is mostly consumed by eigendecompositions performed as part of computing $P_{C_2^r}$, and could perhaps be dramatically reduced by using a cheaper approximate projection. For 1POA we used up to 50 h for a full reconstruction.

Remark 4.2 (An upper bound on distances). The constraint C_1 can be easily modified to incorporate additional distance information. For instance, upper and lower bounds could be placed on the distance between (not necessarily adjacent) carbon atoms on a carbon chain. Note that each carbon–carbon bond is approximately 1.5 Å in length.

Remark 4.3 (Two-phase approach). In our implementation, the Douglas–Rachford method encountered difficulties applied to the reconstruction of the two larger proteins. Therefore, it would be reasonable to consider an approach in which one partitions the atoms into sets and applies the Douglas–Rachford method to these subproblems. The reconstructed distances obtained from these subproblems can then be used as the initial estimates for distances in the original master problem (which considers all the atoms). An iterative version is outlined in Algorithm 4.

Algorithm 4: A two-phase algorithm for protein reconstruction.

```
input  : $D \in \mathbb{R}^{n\times n}$ (the partial Euclidean distance matrix)
Choose random $X \in [-1, 1]^{n\times n}$;
$\Gamma = \{1, 2, \dots, n\}$ (each index represents an atom);
while continue do
    if doPhase1 then    // generate and solve subproblems (phase 1)
        Choose a partition of $\Gamma$ into the sets $\Gamma_1, \Gamma_2, \dots, \Gamma_m$;
        for $k = 1, 2, \dots, m$ do
            Apply Algorithm 2 to atoms indexed by $\Gamma_k$ to obtain $X_k$ (that is, the
            distance matrix for the atoms indexed by $\Gamma_k$); Update $X$ with the
            reconstructed distances in $X_k$;
        end
    else                                   // solve master problem (phase 2)
        Apply Algorithm 2 to all atoms (that is, index by $\Gamma$) to obtain $X$;
    end
end
output: $X$ (the reconstructed distance matrix).
```

We continue to work on such problem-specific refinements of the Douglas–Rachford method: in most of our example problems a natural splitting is less accessible.

It would also be interesting to apply the methods of this section to *sensor network localization* problems requiring the reconstruction of an incomplete distance matrix [22, 26, 35].

4.4. Hadamard matrices Recall that a matrix $H = (H_{ij}) \in \{-1, 1\}^{n\times n}$ is said to be a *Hadamard matrix* of *order n* if

$$H^T H = nI. \tag{4.4}$$

We note that there are many equivalent characterizations. For instance, (4.4) is equivalent to asserting that H has maximal determinant ($|\det H| = n^{n/2}$) [32, Ch. 2]. A classical result of Hadamard asserts that Hadamard matrices exist only if $n = 1, 2$ or a multiple of 4. For orders one and two, such matrices are easy to find. For multiples of four, the *Hadamard conjecture* asks the converse: *If n is a multiple of four, does there exist a Hadamard matrix of order n?* Background on Hadamard matrices can be found in [32]. Thus, an important completion problem starts with structure restrictions, but with no fixed entries.

Consider now the problem of finding a Hadamard matrix of a given order. We define the constraints

$$C_1 = \{X \in \mathbb{R}^{n\times n} \mid X_{ij} = \pm 1 \text{ for } i,\ j = 1, \dots, n\}, \tag{4.5}$$
$$C_2 = \{X \in \mathbb{R}^{n\times n} \mid X^T X = nI\}. \tag{4.6}$$

Then X is a Hadamard matrix if and only if $X \in C_1 \cap C_2$.

The first constraint, C_1, is clearly nonconvex. However, its projection is simple and is given pointwise by

$$P_{C_1}(X)_{ij} = \begin{cases} -1 & \text{if } X_{ij} < 0, \\ \pm 1 & \text{if } X_{ij} = 0, \\ 1 & \text{otherwise.} \end{cases} \tag{4.7}$$

The second constraint, C_2, is also nonconvex. To see this, consider the mid point of the two matrices

$$\begin{pmatrix} \sqrt{2} & 0 \\ 0 & \sqrt{2} \end{pmatrix}, \begin{pmatrix} 0 & \sqrt{2} \\ \sqrt{2} & 0 \end{pmatrix} \in C_2.$$

Nevertheless, a projection can be found by solving the equivalent problem of finding a projection onto the set of orthogonal matrices, which is a special case of the *Procrustes problem* described above.

PROPOSITION 4.4. *Let $X = USV^T$ be a singular value decomposition. Then*

$$\sqrt{n}UV^T \in P_{C_2}(X).$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

Proof. Let $Y = X/\sqrt{n}$. Then

$$\min_{\substack{X\in\mathbb{R}^{n\times n}\\ A^TA=nI}} \|X - A\|_F = \sqrt{n}\Big(\min_{\substack{Y\in\mathbb{R}^{n\times n}\\ B^TB=I}} \|Y - B\|_F\Big).$$

Any solution to the latter is a projection of Y onto the set of orthogonal matrices. One such matrix can be obtained by replacing all singular values of Y by '1' [42]. Since

$$Y = U\hat{S}V^T, \quad \text{where } \hat{S} = S/\sqrt{n},$$

is a singular value decomposition, it follows that UV is a projection of Y onto the set of orthogonal matrices. The result now follows. □

Remark 4.5. Any $A \in P_{C_2}(X)$ is such that $\operatorname{tr}(A^TX) = \max_{B\in C_2} \operatorname{tr}(B^TX)$.

Remark 4.6. Consider instead the matrix completion problem of finding a Hadamard matrix with some known entries. This can be cast within the above framework by appropriate modification of C_1. The projection onto C_1 only differs by leaving the known entries unchanged.

We next give a second useful formulation for the problem of finding a Hadamard matrix of a given order. We take C_1 as in (4.4) and define

$$C_3 = \{X \in \mathbb{R}^{n\times n} \mid X^TX = \|X\|_F\, I\}.$$

If $X \in C_1$, then $\|X\|_F = n$; hence, $C_1 \cap C_2 = C_1 \cap C_3$. It follows that X is a Hadamard matrix if and only if $X \in C_1 \cap C_3$. A projection onto C_3 can be computed using a similar approach to the projection onto C_2.

Proposition 4.7. *Let $X = USV^T$ be a singular value decomposition. Then*

$$\sqrt{\|X\|_F}\, UV^T \in P_{C_3}(X).$$

Proof. This is a straightforward modification of Proposition 4.4. □

Remark 4.8. (Complex Hadamard matrices) It is also possible to consider *complex Hadamard matrices*. In this case,

$$C_1 = \{X \in \mathbb{C}^{n\times n} \mid |X_{ij}| = 1\}.$$

The projection onto C_1 is straightforward, and is given by

$$P_{C_1}(X)_{ij} = \begin{cases} X_{ij}/|X_{ij}| & \text{if } X_{ij} \neq 0,\\ z \in \mathbb{C}, |z| = 1 & \text{otherwise.}\end{cases}$$

Note that the real solutions to $|X_{ij}| = 1$ are ±1.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

TABLE 2. Number of Hadamard matrices found from 1000 instances.

Order	Ave time (s)	Solved	Distinct	Inequivalent
	Proposition 4.4 formulation			
2	1.1371	534	8	1
4	1.0791	627	422	1
8	0.7368	996	996	1
12	7.1298	0	0	0
16	9.4228	0	0	0
20	20.6674	0	0	0
	Proposition 4.7 formulation			
2	1.1970	505	8	1
4	0.2647	921	541	1
8	0.0117	1000	1000	1
12	0.8337	1000	1000	1
16	11.7096	16	16	4
20	22.6034	0	0	0

EXAMPLE 4.9. (Experiments with Hadamard matrices) Let H_1 and H_2 be Hadamard matrices. We say that H_1 and H_2 are *distinct* if $H_1 \neq H_2$. We say that H_1 and H_2 are *equivalent* if H_2 can be obtained from H_1 by performing a sequence of row or column permutations, and/or multiplying rows or columns by -1. The number of distinct (respectively, inequivalent) Hadamard matrices of order $4n$ is given in the *On-Line Encyclopedia of Integer Sequences (OEIS)* sequence A206712: 768, 4954521600, 20251509535014912000, ... (respectively, A007299: 1, 1, 1, 1, 5, 3, 60, 487, 13710027, ...). With increasing order, the number of Hadamard matrices is a faster than exponentially decreasing proportion of the total number of $\{+1, -1\}$ matrices (of which there are 2^{n^2} for order n). This is reflected in the observed rapid increase in difficulty of finding Hadamard matrices using the Douglas–Rachford scheme, as order increases.

We applied the Douglas–Rachford algorithm to 1000 random replications, for each of the above formulations. Our computational experience is summarized in Table 2 and Figure 4. To determine if two Hadamard matrices are equivalent, we used a *Sage* implementation of the graph isomorphism approach outlined in [39].

We make some brief comments on our results.

- The formulation based on Proposition 4.7 was found to be faster and more successful than the formulation based on Proposition 4.4, especially for orders eight and 12, where it was successful in every replication. For orders less than or equal to 12, the Douglas–Rachford scheme was able to find the unique inequivalent Hadamard matrix under either formulation (except for $n = 12$,

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

Proposition 4.4 formulation). Moreover, the Proposition 4.7 formulation was able to find four of the five inequivalent Hadamard matrices of order 16 (all five can be found at http://www.uow.edu.au/~jennie/hadamard.html).

- From Figure 4, we observed that if a Hadamard matrix was found, it was usually found within the first few thousand iterations. The frequency histogram for order 16, shown in Figure 4(f), varied significantly from the corresponding histograms for lower orders.

For orders 20 and above, it is possible that another formulation might be more fruitful, but almost certainly better and more problem-specific heuristics will again be needed.

Remark 4.10. Since C_2 is nonconvex, when computing its projection we are forced to make a selection from the set of nearest points. In our experiments we have always chosen the nearest point in the same way. It may be possible to benefit from making the selection according to some other criterion.

We now turn our attention to some special classes of Hadamard matrices.

4.4.1. *Skew-Hadamard matrices* Recall that a matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ is *skew symmetric* if $A^T = -A$. A *skew-Hadamard matrix* is a Hadamard matrix, H, such that $I - H$ is skew symmetric. That is,

$$H + H^T = 2I.$$

Skew-Hadamard matrices are of interest, for example, in the construction of *combinatorial designs* [36]. The number of inequivalent skew-Hadamard matrices of order $4n$ is given in OEIS sequence A001119: $1, 1, 2, 2, 16, 54, \ldots$ (for $n = 2, 3, \ldots$).

In addition to the constraints C_1 and C_2 from the previous section, we define the affine constraint

$$C_3 = \{X \in \mathbb{R}^{n\times n} \mid X + X^T = 2I\}.$$

A projection onto $C_1 \cap C_3$ is given by

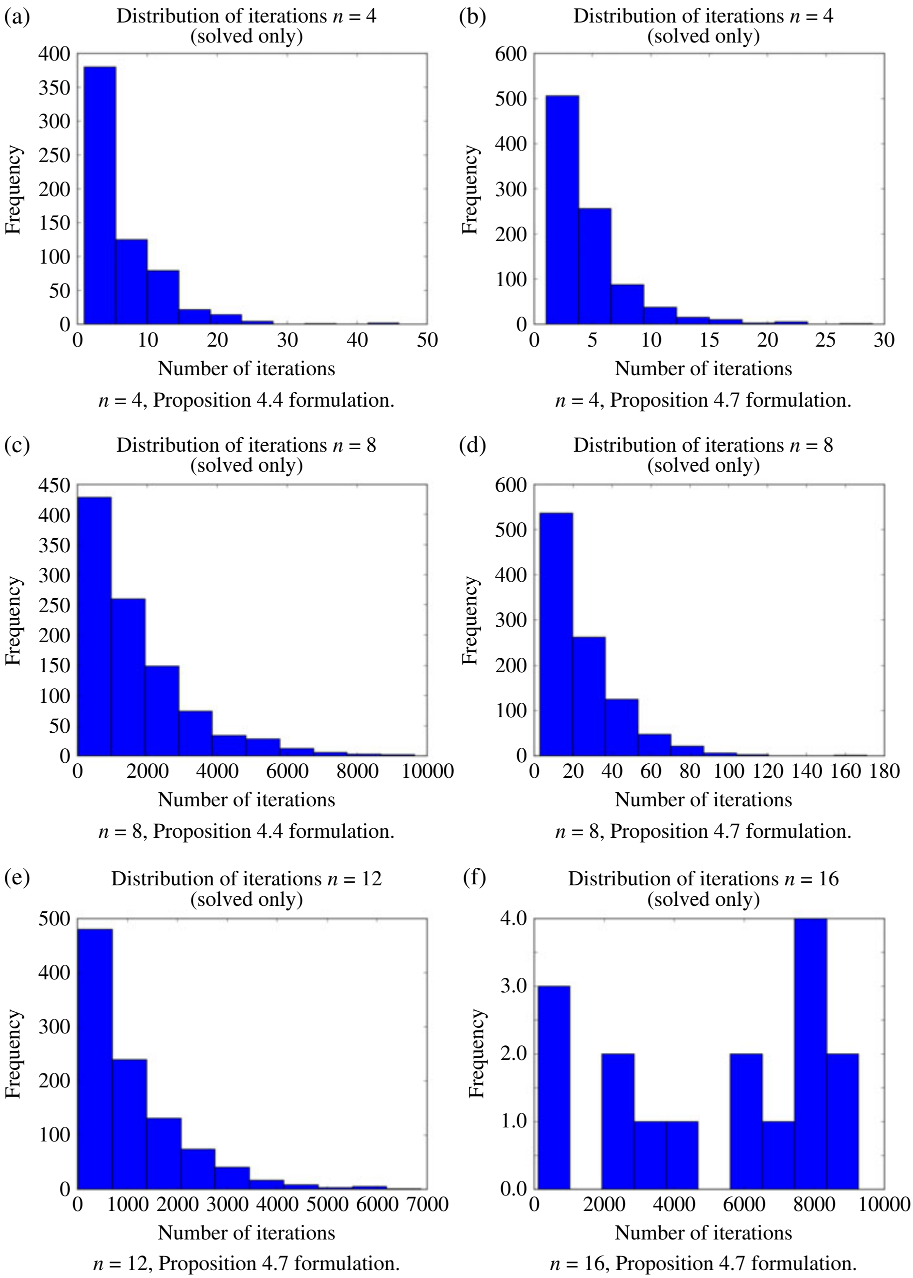
$$P_{C_1\cap C_3}(X)_{ij} = \begin{cases} 1 & \text{if } i \neq j \text{ and } X_{ij} \geq X_{ji}, \\ -1 & \text{if } i \neq j \text{ and } X_{ij} < X_{ji}, \\ 1 & \text{otherwise.} \end{cases}$$

Then X is a skew-Hadamard matrix if and only if $X \in (C_1 \cap C_3) \cap C_2$.

Table 3 shows the results of the same experiment as Section 4.4, but with the skew constraint incorporated.

Remark 4.11. Comparing the results of Table 3 with those of Table 2, it is notable that by placing additional constraints on the problem, both methods now succeed at higher orders, method two is faster than before and we can successfully find all inequivalent skew matrices of order 20 or less.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press



$n = 4$, Proposition 4.4 formulation.

$n = 4$, Proposition 4.7 formulation.

$n = 8$, Proposition 4.4 formulation.

$n = 8$, Proposition 4.7 formulation.

$n = 12$, Proposition 4.7 formulation.

$n = 16$, Proposition 4.7 formulation.

FIGURE 4. Frequency histograms showing the number of iterations required to find a Hadamard matrix, for different orders and formulations (solved instances only).

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

TABLE 3. Number of skew-Hadamard matrices found from 1000 instances.

Order	Ave time (s)	Solved	Distinct	Inequivalent
	Proposition 4.4 formulation			
2	0.0003	1000	2	1
4	1.1095	719	16	1
8	0.7039	902	889	1
12	14.1835	43	43	1
16	19.3462	0	0	0
20	29.0383	0	0	0
	Proposition 4.7 formulation			
2	0.0004	1000	2	1
4	1.6381	634	16	1
8	0.0991	986	968	1
12	0.0497	999	999	1
16	0.2298	1000	1000	2
20	20.0296	495	495	2

In contrast, the three-set feasibility problem $C_1 \cap C_2 \cap C_3$ was unsuccessful, except for order two. This is despite the projection onto the affine set C_3 having the simple formula

$$P_{C_3}(X) = I + \frac{X - X^T}{2}, \tag{4.8}$$

which allows for its efficient computation.

4.4.2. *Circulant Hadamard matrices* Recall that a matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$ is *circulant* if it can be expressed as

$$A = \begin{pmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_n \\ \lambda_n & \lambda_1 & \dots & \lambda_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_2 & \lambda_3 & \dots & \lambda_1 \end{pmatrix}$$

for some vector $\lambda \in \mathbb{R}^n$.

The set of circulant matrices forms a subspace of $\mathbb{R}^{n\times n}$. The set $\{P^k \mid k = 1, 2, \dots, n\}$, where P is the cyclic permutation matrix

$$P = \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix},$$

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

forms a basis. Consequently, any circulant matrix, A, can be expressed as a linear combination of the form

$$A = \sum_{k=1}^{n} \lambda_k P^k.$$

Remark 4.12. Right (respectively, left) multiplication by P results in a cyclic permutation of rows (respectively, columns). Hence, $P^2, P^3, \dots, P^n$ represent all cyclic permutations of the rows (respectively, columns) of P. In particular, P^n is the identity matrix.

Proposition 4.13. *[31, Exercise 6.7] For $X \in \mathbb{R}^{n\times n}$, the nearest circulant matrix is given by*

$$\sum_{k=1}^{n} \lambda_k P^k, \quad \textit{where } \lambda_k = \frac{1}{n}\sum_{i,j} P^k_{ij} X_{ij}.$$

A *circulant Hadamard matrix* is a Hadamard matrix which is also circulant.

The circulant Hadamard conjecture asserts: *No circulant Hadamard matrix of order larger than four exists.* For recent progress on the conjecture, we refer the reader to a paper of Leung and Schmidt [38]. Consistent with this conjecture, our Douglas–Rachford implementation can successfully find circulant matrices of order four, but fails for higher orders.

5. Conclusion

We have provided general guidelines for successful application of the Douglas–Rachford method to (real) matrix completion problems, both convex and nonconvex. The message of the previous two sections is the following. When presented with a new (potentially nonconvex) feasibility problem, it is well worth seeing if Douglas–Rachford can deal with it, since it is both conceptually very simple and usually relatively easy to implement. If it works, one may then think about refinements if performance is less than desired.

Moreover, this approach allows for the intuition developed for continuous optimization in Euclidean space to be usefully repurposed. This also lets one profitably consider nonexpansive fixed point methods in the class of so-called *CAT(0) metric spaces* – a far-ranging concept introduced 20 years ago in algebraic topology, but now finding applications to optimization and fixed point algorithms. The convergence of various projection-type algorithms to feasible points is under investigation by Searston and Sims among others in such spaces [4] – thereby broadening the constraint structures to which projection-type algorithms apply to include metrically rather than algebraically convex sets.

Ongoing and future computational experiments could include the following aspects.

- Implementation of the modified protein reconstruction formulation outlined in Remarks 4.2 and 4.3.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

- Considering similar reconstruction problems arising in the context of ionic liquid chemistry and, as mentioned, in sensor location problems.
- Likewise, for the discovery of larger Hadamard matrices to be tractable by Douglas–Rachford methods, a more efficient implementation is needed and a more puissant model.

Acknowledgements

We wish to thank Judy-anne Osborn and Richard Brent for useful discussions about Hadamard matrices, David Allingham for useful discussions on correlation matrices, Henry Wolkowicz for directing us to many useful matrix completion resources and Brailey Sims for careful reading of the manuscript. Many resources can be found at the paper's companion website: http://carma.newcastle.edu.au/DRmethods.

References

[1] G. Ammar, P. Benner and V. Mehrmann, "A multishift algorithm for the numerical solution of algebraic Riccati equations", *Electron. Trans. Numer. Anal.* **1** (1993) 33–48.

[2] F. J. Aragón Artacho and J. M. Borwein, "Global convergence of a non-convex Douglas–Rachford iteration", *J. Global Optim.* **57** (2013) 753–769; doi:10.1007/s10898-012-9958-4.

[3] F. J. Aragón Artacho, J. M. Borwein and M. K. Tam, "Recent results on Douglas–Rachford method for combinatorial optimization problems", *J. Optim. Theory Appl.* (2013); doi:10.1007/s10957-013-0488-0.

[4] M. Bačák, M. I. Searston and I. B. Sims, "Alternating projections in CAT(0) spaces", *J. Math. Anal. Appl.* **385** (2012) 599–607.

[5] H. H. Bauschke and J. M. Borwein, "On the convergence of von Neumann's alternating projection algorithm for two sets", *Set-Valued Anal.* **1** (1993) 185–212; doi:10.1007/BF01027691.

[6] H. H. Bauschke and J. M. Borwein, "Dykstra's alternating projection algorithm for two sets", *J. Approx. Theory* **79** (1994) 418–443; doi:10.1006/jath.1994.1136.

[7] H. H. Bauschke and J. M. Borwein, "On projection algorithms for solving convex feasibility problems", *SIAM Rev.* **38** (1996) 367–426; doi:10.1137/S0036144593251710.

[8] H. H. Bauschke, J. M. Borwein and A. Lewis, "The method of cyclic projections for closed convex sets in Hilbert space", *Contemp. Math.* **204** (1997) 1–38; doi:10.1090/conm/204/02620.

[9] H. H. Bauschke, P. L. Combettes and D. R. Luke, "Phase retrieval, error reduction algorithm, and Fienup variants: a view from convex optimization", *J. Opt. Soc. Amer.* A **19** (2002) 1334–1345; doi:10.1364/JOSAA.19.001334.

[10] H. H. Bauschke, P. L. Combettes and D. R. Luke, "Hybrid projection–reflection method for phased retrieval", *J. Opt. Soc. Amer.* A **20** (2003) 1025–1034; doi:10.1364/JOSAA.20.001025.

[11] H. H. Bauschke, P. L. Combettes and D. R. Luke, "Finding best approximation pairs relative to two closed convex sets in Hilbert space", *J. Approx. Theory* **127** (2004) 178–192; doi:10.1016/j.jat.2004.02.006.

[12] E. G. Birgin and M. Raydan, "Robust stopping criteria for Dykstra's algorithm", *SIAM J. Sci. Comput.* **26** (2005) 1405–1414; doi:10.1137/03060062X.

[13] J. M. Borwein and A. S. Lewis, *Convex analysis and nonlinear optimization* (Springer, New York, 2006).

[14] J. M. Borwein and D. R. Luke, "Entropic regularization of the ℓ_0 function", in: *Fixed-point algorithms for inverse problems in science and engineering* (Springer, New York, 2011), 65–92.

[15] J. M. Borwein and B. Sims, "The Douglas–Rachford algorithm in the absence of convexity", in: *Fixed-point algorithms for inverse problems in science and engineering* (Springer, New York, 2011), 93–109.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press

[16] J. M. Borwein and M. K. Tam, "A cyclic Douglas–Rachford iteration scheme", *J. Optim. Theory Appl.* **160** (2014) 1–29; doi:10.1007/s10957-013-0381-x.

[17] J. Boyle and R. Dykstra, "A method for finding projections onto the intersection of convex sets in Hilbert spaces", *Lecture Notes in Statist.* **37** (1986) 28–47; doi:10.1007/978-1-4613-9940-7_3.

[18] J. F. Cai, E. J. Candès and Z. Shen, "A singular value thresholding algorithm for matrix completion", *SIAM J. Optim.* **20** (2010) 717–772; doi:10.1137/080738970.

[19] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization", *Found. Comput. Math.* **9** (2009) 717–772; doi:10.1007/s10208-009-9045-5.

[20] Y. Chen and X. Ye, "Projection onto a simplex", arXiv:1101.6081v2, 2011.

[21] J. Dattorro, *Convex optimization and Euclidean distance geometry* (Meboo Publishing, California, 2013).

[22] P. Drineas, A. Javed, M. Magdon-Ismail, G. Pandurangant, R. Virrankoski and A. Savvides, "Distance matrix reconstruction from incomplete distance information for sensor network localization", *Proc. IEEE Int. Conf. Sensor, Mesh and Ad Hoc Communications and Networks (SECON'06)*, 2006, 536–544; doi:10.1109/SAHCN.2006.288510.

[23] R. Escalante and M. Raydan, *Alternating Projection Methods* (SIAM, Philadelphia, 2011); doi:10.1137/19781611971941.

[24] L. Fabry-Asztalos, I. Lorentz and R. Andonie, "Molecular distance geometry optimization using geometric build-up and evolutionary techniques on GPU", *Comput. Intell. Bioinform. Comput. Biol.* (2012) 321–328; doi:10.1109/CIBCB.2012.6217247.

[25] N. Gaffke and R. Mathar, "A cyclic projection algorithm via duality", *Metrika* **36** (1989) 29–54; doi:10.1007/BF02614077.

[26] M. R. Gholami, L. Tetruashvili, E. G. Strm and Y. Censor, "Cooperative wireless sensor network positioning via implicit convex feasibility", *IEEE Trans. Signal Process.* **61** (2013) 5830–5840; doi:10.1109/TSP.2013.2279770.

[27] 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) 589–600; doi:10.1137/0611042.

[28] T. L. Hayden and J. Wells, "Approximation by matrices positive semidefinite on a subspace", *Linear Algebra Appl.* **109** (1988) 115–130; doi:10.1016/0024-3795(88)90202-9.

[29] R. Hesse and D. R. Luke, "Nonconvex notions of regularity and convergence of fundamental algorithms for feasibility problems", *SIAM J. Optim.* **23** (2013) 2397–2419; doi:10.1137/120902653.

[30] N. J. Higham, "Computing the polar decomposition—with applications", *SIAM J. Sci. Stat. Comput.* **7** (1986) 1160–1174; doi:10.1137/0907079.

[31] A. Hjørungnes, *Complex-Valued Matrix Derivatives: with Applications in Signal Processing and Communications* (Cambridge University Press, New York, 2011); doi:10.1017/CBO9780511921490.

[32] K. J. Horadam, *Hadamard Matrices and their Applications* (Princeton University Press, New Jersey, 2007).

[33] R. A. Horn and C. R. Johnson, *Matrix Analysis* (Cambridge University Press, New York, 1985); doi:101017/CBO9780511810817.

[34] C. R. Johnson, "Matrix completion problems: a survey", *Proc. Sympos. Appl. Math.* **40** (1990) 171–198; doi:10.1090/psapm/040/1059486.

[35] N. Kirslock and H. Wolkowicz, "Explicit sensor network localization using semidefinite representations and facial reductions", *SIAM J. Optim.* **20** (2010) 2679–2708.

[36] C. Koukouvinos and S. Stylianou, "On skew-Hadamard matrices", *Discrete Math.* **309** (2008) 2723–2731; doi:10.1016/j.disc.2006.06.037.

[37] M. Laurent, "Matrix completion problems", *Encyclopedia Optim.* **271** (2001) 221–229; doi:10.1007/0-306-48332-7_271.

[38] K. H. Leung and B. Schmidt, "New restrictions on possible order of circulant Hadamard matrices", *Des. Codes Cryptogr.* **64** (2012) 143–151; doi:10.1007/s10623-011-9493-1.

[39] B. D. McKay, “Hadamard equivalence via graph isomorphism”, *Discrete Math.* **27** (1979) 213–214; doi:10.1016/0012-365X(79)90113-4.

[40] J. Piantadosi, P. Howlett and J. M. Borwein, “Modelling and simulation of seasonal rainfall using the principle of maximum entropy”, *Entropy* **16** (2014) 747–769; doi:10.3390/e16020747.

[41] I. J. Schoenberg, “Remarks to Maurice Fréchet’s article ‘Sur la définition axiomatique d’une classe d’espace distanciés vectoriellement applicable sur l’espace de Hilbert’”, *Ann. of Math. (2)* **36** (1935) 724–732.

[42] P. H. Schönemann, “A generalized solution to the orthogonal Procrustes problem”, *Psychometrika* **31** (1966) 1–10; doi:10.1007/BF02289451.

[43] P. L. Takouda, “Un probléme d’approximation matricielle: quelle est la matrice bistochastique la plus proche d’une matrice donnée?”, *RAIRO Oper. Res.* **39** (2005) 35–54.

[44] A. K. Yuen, O. Lafon, O. T. Charpentier, M. Roy, F. Brunet, P. Berthault, D. Sakellariou, B. Robert, S. Rimsky, F. Pillon, J. C. Cintrat and B. Rousseau, “Measurement of long-range interatomic distances by solid-state tritium-NMR spectroscopy”, *J. Amer. Chem. Soc.* **132** (2010) 1734–1735; doi:10.1021/ja908915v.

https://doi.org/10.1017/S1446181114000145 Published online by Cambridge University Press