Chapter 1
Reflection methods for inverse problems with applications to protein conformation determination

Jonathan M. Borwein and Matthew K. Tam

Abstract The Douglas–Rachford reflection method is a general purpose algorithm useful for solving the feasibility problem of finding a point in the intersection of finitely many sets. In this chapter we demonstrate that applied to a specific problem, the method can benefit from heuristics specific to said problem which exploit its special structure. In particular, we focus on the problem of protein conformation determination formulated within the framework of matrix completion, as was considered in a recent paper of the present authors.

Key words: reflection methods; inverse problems; protein conformation

1.1 Techniques of Variational Analysis

This chapter builds on a series of seven lectures titled *Techniques of Variational Analysis* given by the first author at the CIMPA school *Generalized Nash Equilibrium Problems, Bilevel Programming and MPEC* held November 25 to December 6, 2013, University of Delhi, New Delhi, India. In this written presentation we focus on *reflection methods* for *protein conformation determination*, as was discussed in the seventh and final lecture of the series. The complete lectures — one through six taken from [13] — can be found online at:

  `http://www.carma.newcastle.edu.au/jon/ToVA/links.html`

Before turning our attention to reflection methods, we briefly outline the content of the first six lectures.

J.M. Borwein

CARMA Centre, University of Newcastle, Callaghan, NSW 2308, Australia. e-mail: `jon.borwein@gmail.com`

M.K. Tam

CARMA Centre, University of Newcastle, Callaghan, NSW 2308, Australia. e-mail: `matthew.tam@uon.edu.au`

- **Lectures 1 & 2** provided an introduction to variational analysis and variational principles [13, §1-§2].
- **Lectures 3 & 4** introduced nonsmooth analysis: normal cones and subdifferentials of lower semi-continuous functions, Fréchet and limiting calculus [13, §3.1-§3.4], and discussed convex functions and their calculus rules [13, §4.1-§4.4].
- **Lecture 5** turned to multifunction analysis: sequences of sets, continuity of maps, minimality and maximal monotonicity, and distance functions [13, §5.1-§5.3].
- **Lecture 6** focussed on convex feasibility problems and the method of alternating projections [13, §4.7], and therefore providing the preliminary background for the rest of this chapter.

1.2 Introduction to Reflection Methods

Given a (finite) family of sets, the corresponding *feasibility problem* is to find a point contained in their intersection. *Douglas–Rachford reflection methods* form a class of general-purpose iterative algorithms which are useful for solving such problems. At each iteration, these methods perform *(metric) reflections* and *(metric/nearest point) projections* with respect to the individual constraint sets in a prescribed fashion. Such methods are most useful when applied to feasibility problems whose constraint sets have more easily computable reflections and projections than does the intersection.

When the underlying constraint sets are all convex, Douglas–Rachford methods are relatively well understood [6, 12, 11, 7] — their behaviour can be analysed using nonexpansivity properties of convex projections and reflections. In the absence of convexity, recent result have assumed the constraint sets to possess other structural and regularity properties [10, 1, 20]. However, at present, this developing theoretical foundation is not sufficiently rich to explain many of the successful applications in which one or more of the constraint sets lacks convexity [3, 2, 17, 18]. In these cases, the method can be viewed as a heuristic inspired by its behaviour within fully convex settings.

More generally, with any algorithm there is typically a trade-off between the scope of their applicability and tailoring of performance to particular instances. Douglas–Rachford reflection methods are no different. Owing to these methods' broad applicability, potential for further problem specific refinements when applied to special classes of feasibility problems is possible.

In this chapter, we investigate and develop one such refinement with a focus on application of the Douglas–Rachford method to *protein conformation determination*. This application was previously considered as part of [3]. We now propose problem specific heuristics, and also study the effect of increasing problem size. We finish by demonstrating a complementary application of the approach arising in the context of *ionic liquid chemistry*.

The remainder of this chapter is organized as follows. In Sections 1.3, 1.4, 1.5 & 1.6 we introduce the necessary mathematical preliminaries along with the Douglas–

Rachford reflection method, before formulating the protein conformation determination problem. Substantial numerical and graphical results are given in Section 1.7, and concluding remarks in Section 1.8.

1.3 Mathematical Preliminaries

Let $\mathbb{E}$ denote a Euclidean space, that is, a finite dimensional Hilbert space. We will mainly be concerned with the space $\mathbb{R}^{m \times m}$ (i.e., real $m \times m$ matrices) equipped with the inner-product given by

$$\langle A, B \rangle := \text{tr}(A^T B).$$

Here the symbol $\text{tr}(X)$ (resp. X^T) denotes the trace (resp. transpose) of the matrix X. The induced norm is the *Frobenius norm* and can be expressed as

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

The subspace of real symmetric $m \times m$ matrices is denoted S^m, and the cone of positive semi-definite $m \times m$ matrices by S_+^m.

Given sets $C_1, C_2, \ldots, C_N \subseteq \mathbb{E}$, the *feasibility problem* is

$$\text{find } x \in \bigcap_{i=1}^{N} C_i. \tag{1.1}$$

When the intersection in (1.1) is empty, one often seeks a "good" surrogate for a point in the intersection. When $N = 2$, a useful surrogate is a pair of points, one from each set, which minimize the distance between the sets – a *best approximation pair* [6].

1.4 Matrix Completion

A *partial (real) matrix* is an $m \times m$ array for which entries only in certain locations are known. Given a partial matrix $A = (a_{ij}) \in \mathbb{R}^{m \times m}$, a matrix $B = (b_{ij}) \in \mathbb{R}^{m \times m}$ is a *completion* of A if $b_{ij} = a_{ij}$ whenever a_{ij} is known. The problem of *(real) matrix completion* is the following: *Given a partial matrix find a completion belonging to a specified family of matrices.*

Matrix completion can be naturally formulated as a feasibility problem. Let A be the partial matrix to be completed. Choose $C_1, C_2, \ldots, C_N$ such that their intersection is equal to the intersection of completions of A with the specified matrix family. Then (1.1) is precisely the problem of matrix completion for A. The simplest such

case is when C_1 is the set of all completions of A and the intersection of $C_2, \dots, C_N$ equals the desired matrix class.

Remark 1. More generally, one may profitably consider matrix completion for rectangular matrices [3], for example with doubly stochastic matrices. However, since the partial matrices in the discussed protein application are always square, for the purposes of this discussion, we only concern ourselves with the square case.

1.5 The Douglas–Rachford Reflection Method

The *projection onto* $C \subseteq \mathbb{E}$ is the set-valued mapping $P_C : \mathbb{E} \rightrightarrows C$ which maps any point $x \in \mathbb{E}$ to its sets of nearest points in C. More precisely,

$$P_C(x) = \left\{ c \in C : \|x - c\| \leq \inf_{y \in C} \|x - y\| \right\}.$$

The *reflection with respect to* C is the set-valued mapping $R_C : \mathbb{E} \rightrightarrows \mathbb{E}$ given by $R_C = 2P_C - I$, where I denotes the identity mapping.

When C is non-empty, closed, and convex, its corresponding projection operator (and hence its reflection) is single-valued (see, for example, [15, Ch. 1.2]).
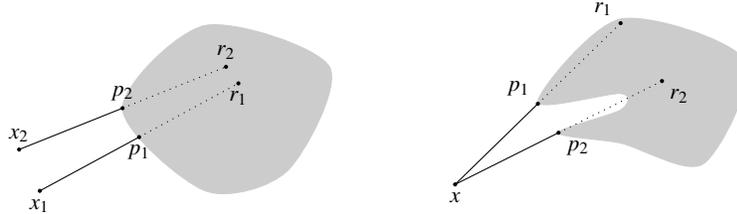

Fig. 1.1 (Left) The (single-valued) projection, p_i, and reflection, r_i, of the point x_i onto a convex set, for $i = 1, 2$. (Right) The (set-valued) projection, $\{p_1, p_2\}$, and reflection, $\{r_1, r_2\}$, of the point x onto a non-convex set. Note the non-expansivity of the reflection in the convex case.

Given $A, B \subseteq \mathbb{E}$ and $x_0 \in \mathbb{E}$, the *Douglas–Rachford reflection method* is the fixed point iteration given by

$$x_{n+1} \in T_{A,B} x_n \text{ where } T_{A,B} = \frac{I + R_B R_A}{2}. \tag{1.2}$$

We refer to the sequence $(x_n)_{n=1}^{\infty}$ as a *Douglas–Rachford sequence*, and to the mapping $T_{A,B}$ as the *Douglas–Rachford operator*.

We now recall the behavior of the Douglas–Rachford method in the classical convex setting. In this case, $T_{A,B}$ is single-valued as a consequence of the single-

valuedness of each of P_A, P_B, R_A and R_B. We denote the set of *fixed points* of a single-valued mapping T by $\mathrm{Fix}\,T = \{x \in \mathbb{E} : Tx = x\}$, and the *normal cone* of a convex set C at the point x by

$$N_C(x) = \begin{cases} \{y \in \mathbb{E} : \langle C - x, u \rangle \leq 0\} & \text{if } x \in C, \\ \emptyset & \text{otherwise.} \end{cases}$$

For convenience, we also introduce the two sets

$$E = \left\{ x \in A : \inf_{a \in A} \|a - x\| \leq \inf_{a \in A, b \in B} \|a - b\| \right\},$$

$$F = \left\{ x \in B : \inf_{b \in B} \|x - b\| \leq \inf_{a \in A, b \in B} \|a - b\| \right\},$$

and the vector $v = P_{\overline{B-A}}(0)$. Here the overline denotes the closure of the set.

Theorem 1 (Convex Douglas–Rachford in finite dimensions [6]). *Suppose $A, B \subseteq \mathbb{E}$ are closed and convex. For any $x_0 \in \mathbb{E}$ define $x_{n+1} = T_{A,B}x_n$. Then there is some $v \in \mathbb{E}$ such that:*

(i) $x_{n+1} - x_n = P_B R_A x_n - P_A x_n \to v$ and $P_B P_A x_n - P_A x_n \to v$.
(ii) If $A \cap B \neq \emptyset$ then $(x_n)_{n=1}^{\infty}$ converges to a point in

$$\mathrm{Fix}(T_{A,B}) = (A \cap B) + N_{\overline{A-B}}(0);$$

otherwise, $\|x_n\| \to +\infty$.
(iii) Exactly one of the following two alternatives holds.

(a) $E = \emptyset$, $\|P_A x_n\| \to +\infty$, and $\|P_B P_A x_n\| \to +\infty$.
(b) $E \neq \emptyset$, the sequences $(P_A x_n)_{n=1}^{\infty}$ and $(P_B P_A x_n)_{n=1}^{\infty}$ are bounded, and their cluster points belong to E and F, respectively; in fact, the cluster points of

$$((P_A x_n, P_B R_A x_n))_{n=1}^{\infty} \text{ and } ((P_A x_n, P_B P_A x_n))_{n=1}^{\infty}$$

are a best approximation pairs relative to (A, B).

Theorem 1 provides the template for application of the Douglas–Rachford method as a heuristic for non-convex feasibility problems. Furthermore, this theorem also shows that for the Douglas–Rachford method the sequence of primary interest is not the fixed point iterates $(x_n)_{n=1}^{\infty}$ themselves, but their *shadows* $(P_A x_n)_{n=1}^{\infty}$.

Remark 2 (Douglas–Rachford splitting). The Douglas–Rachford reflection method can be viewed as a special case of the *Douglas–Rachford splitting algorithm* for finding a zero of the sum of two maximally monotone operators. This more general splitting method iterates by using *resolvents* of the given maximally monotone operators rather than projection operators of sets. The reflection method is obtained in the special case in which the maximal monotone operators are normal cones to the feasibility problem sets. For details, we refer the reader to [5].
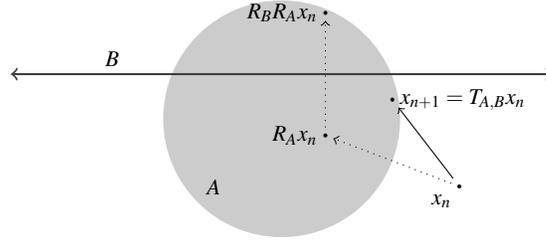
Fig. 1.2 One iteration of the Douglas–Rachford method for the sets $A = \{x \in \mathbb{E} : \|x\| \leq 1\}$ and $B = \{x \in \mathbb{E} : \langle a, x \rangle = b\}$.

Within an implementation of the Douglas–Rachford method, computation of the projection operators are the component typically requiring the most resources. It is therefore beneficial to store two additional sequences in memory; the *shadow sequence* $(P_A x_n)_{n=1}^{\infty}$, and the sequence $(P_B R_A x_n)_{n=1}^{\infty}$. This is because iteration (1.2) is expressible as

$$\begin{aligned} x_{n+1} &\in x_n + P_B R_A x_n - P_A x_n \\ &= x_n + P_B(2P_A x_n - x_n) - P_A x_n. \end{aligned} \tag{1.3}$$

An implementation utilizing this approach is given in Algorithm 1.3. The stopping criterion uses a relative error and is discussed in Section 1.7.

Fig. 1.3 Implementation of the basic Douglas–Rachford algorithm.

Input: $x_0 \in \mathbb{E}$ and $\varepsilon > 0$
$n = 0$;
$p_0 \in P_A(x_0)$;
while $n = 0$ **or** $\|r_n - p_n\| > \varepsilon \|p_n\|$ **do**
$\quad r_n \in P_B(2p_n - x_n)$;
$\quad x_{n+1} = x_n + r_n - p_n$;
$\quad p_{n+1} \in P_A(x_{n+1})$;
$\quad n = n + 1$;
end
Output: $p_n \in \mathbb{E}$

1.6 Protein Conformation Determination

Proteins are large biomolecules which are comprised of multiple amino acid residues,[1] each of which typically consists of between 10 and 25 atoms. Proteins participate is virtually every cellar process, and knowledge of their structural conformation gives insight into the mechanisms by which they perform.

One of many techniques that can be used to determine conformation is *nuclear magnetic resonance (NMR)*. Currently NMR is only able to non-destructively re-solve relatively short distances (*i.e.,* those less than $\sim 6\text{Å}$). In the proteins we con-sider, this corresponds to less than 9% of all non-zero inter-atom distances.

We now formulate the problem of protein conformation determination as a com-putationally tractable matrix completion problem. In fact, our formulation is a *low-rank Euclidean distance matrix completion problem*. We next introduce the neces-sary definitions.

We say that a matrix $D = (D_{ij}) \in \mathbb{R}^{m \times m}$ is a *Euclidean distance matrix (EDM)* if there exists points $z_1, z_2, \ldots, z_n \in \mathbb{R}^m$ such that

$$D_{ij} = \|z_i - z_j\|^2 \text{ for } i, j = 1, 2, \ldots, m. \tag{1.4}$$

Clearly any EDM is symmetric, non-negative, and hollow (*i.e.,* contains only zeros along its main diagonal). When (1.4) holds for a set of points in $\mathbb{R}^q$, we say D is *embeddable* in $\mathbb{R}^q$. If D is embeddable in $\mathbb{R}^q$ but not in $\mathbb{R}^{q-1}$, then we say that D is *irreducibly embeddable* in $\mathbb{R}^q$.

We now recall a useful characterization of EDMs, due to Hayden and Wells [19]. In what follows, the matrix $Q \in \mathbb{R}^{m \times m}$ is the *Householder matrix* given by

$$Q = I - \frac{2vv^T}{v^T v}, \text{ where } v = \begin{bmatrix} 1, & 1, & \ldots & 1, & 1 + \sqrt{m} \end{bmatrix}^T \in \mathbb{R}^m.$$

Theorem 2 (EDM characterization [19, Th. 3.3]). *A non-negative, symmetric, hollow matrix $X \in \mathbb{R}^{m \times m}$ is a Euclidean distance matrix if and only if the block $\widehat{X} \in \mathbb{R}^{(m-1) \times (m-1)}$ in*

$$Q(-X)Q = \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix} \tag{1.5}$$

is positive semi-definite. In this case, X is irreducibly embeddable in $\mathbb{R}^q$ where $q = \text{rank}(\widehat{X}) \leq m - 1$.

The problem of *low-rank Euclidean distance matrix completion* can now be for-mulated. Let D denote a partial Euclidean distance matrix, with entry D_{ij} known whenever $(i, j) \in \Omega$ for some index set Ω, which is embeddable in $\mathbb{R}^q$. Without loss of generality, we make the following three simplifying assumptions on the partial matrix D and index set Ω.

1. (non-negative) $D \geq 0$ (*i.e.,* $D_{ij} \geq 0$ for all $i, j = 1, 2, \ldots, m$);

---

[1] When two amino acids form a peptide bond, a water molecule is formed. An *amino acid residue* is what remains of each amino acid after this reaction.

2. (hollow) $D_{ii} = 0$ and $(i,i) \in \Omega$ for $i = 1,2,\ldots,m$;
3. (symmetric) $(i,j) \in \Omega \iff (j,i) \in \Omega$, and $D_{ij} = D_{ji}$ for all $(i,j) \in \Omega$.

We define two constraint sets

$$C_1 = \left\{ X \in S^m : X \geq 0, X_{ij} = D_{ij} \text{ for all } (i,j) \in \Omega \right\},$$

$$C_2 = \left\{ X \in S^m : Q(-X)Q = \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix}, \begin{array}{c} \widehat{X} \in S_+^{m-1}, \ d \in \mathbb{R}^{m-1} \\ \operatorname{rank}\widehat{X} \leq q, \ \delta \in \mathbb{R} \end{array} \right\}. \tag{1.6}$$

In light of Theorem 2, the problem of *low-rank Euclidean distance matrix completion* can be cast as the two-set feasibility problem

$$\text{find } X \in C_1 \cap C_2.$$

That is, a matrix X is a low-rank Euclidean distance matrix which completes D if and only if $X \in C_1 \cap C_2$. Some comments regarding the constraint sets in (1.6) are in order.

The set C_1 encodes the experimental data obtained from NMR, and the *a priori* knowledge that the matrix is non-negative, symmetric and hollow. Its projection has a simple formulae, as we now show.

Proposition 1 (Projection onto C_1). *Let $X \in \mathbb{R}^{m \times m}$. Then $P_{C_1}X$ is given element-wise by*

$$(P_{C_1}X)_{ij} = \begin{cases} D_{ij}, & (i,j) \in \Omega \\ \max\{0, X_{ij}\}, & (i,j) \notin \Omega \end{cases} \quad \text{for} \quad i,j = 1,2,\ldots,m.$$

Proof. Let Y be any matrix in C_1. We have

$$\begin{aligned} \|X-Y\|_F^2 &= \sum_{(i,j) \in \Omega} (X_{ij}-Y_{ij})^2 + \sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} < 0}} (X_{ij}-Y_{ij})^2 + \sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} \geq 0}} (X_{ij}-Y_{ij})^2 \\ &= \sum_{(i,j) \in \Omega} (X_{ij}-D_{ij})^2 + \sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} < 0}} X_{ij}^2 + \sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} \geq 0}} (X_{ij}-Y_{ij})^2. \end{aligned} \tag{1.7}$$

Let P be the matrix given by the proposed projection formula (clearly $P \in C_1$). Then

$$\sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} \geq 0}} (X_{ij}-Y_{ij})^2 \geq \sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} \geq 0}} (X_{ij}-X_{ij})^2 = \sum_{\substack{(i,j) \notin \Omega \\ \text{s.t. } X_{ij} \geq 0}} (X_{ij}-P_{ij})^2. \tag{1.8}$$

By combining (1.7) and (1.8) we see that

$$\|X-Y\|_F^2 \geq \|X-P\|_F^2 \text{ for all } Y \in C_1.$$

Since C_1 is closed and convex, P is the unique nearest point to X in C_1. $\qquad\square$

Remark 3. Since C_1 is a closed convex set, an alternative (less direct) proof of Proposition 1 can be given using the standard variational characterization of convex projections [15, Th. 1.2.4].

Using the necessary condition given by Theorem 2, the non-convex set C_2 encodes the *a priori* knowledge that the matrix of interest is a EDM together with the dimension of the space in which the corresponding points generating the matrix are contained. We now derive the projection onto C_2.

Theorem 3 (Nearest low-rank EDMs [3]). *Let $X \in S^m$ be a non-negative, hollow matrix. Then*

$$P_{C_2}(X) = \left\{ -Q \begin{bmatrix} \widehat{Y} & d \\ d^T & \delta \end{bmatrix} Q : Q(-X)Q = \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix}, \begin{matrix} \widehat{X} \in \mathbb{R}^{(m-1)\times(m-1)}, \\ d \in \mathbb{R}^{m-1}, \ \delta \in \mathbb{R}, \end{matrix} \widehat{Y} \in P_M\widehat{X} \right\},$$

where M is the set of positive semi-definite matrices with rank q or less. In particular, $P_{C_2}(X)$ is a singleton if and only if $P_M\widehat{X}$ is a singleton.

Proof. Let Y be any matrix in C_2. That is,

$$Y = \begin{bmatrix} \widehat{Y} & c \\ c^T & \beta \end{bmatrix}, \quad \text{for some } c \in \mathbb{R}^{m-1}, \beta \in \mathbb{R}, \widehat{Y} \in S.$$

Using the orthogonality of Q, we compute

$$\begin{aligned}
\|X - Y\|_F^2 &= \|Q(X-Y)Q\|_F^2 = \|Q(-X)Q - Q(-Y)Q\|_F^2 \\
&= \left\| \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix} - \begin{bmatrix} \widehat{Y} & c \\ c^T & \beta \end{bmatrix} \right\|_F^2 = \left\| \begin{bmatrix} \widehat{X} - \widehat{Y} & (d-c) \\ (d-c)^T & (\delta - \beta) \end{bmatrix} \right\|_F^2 \qquad (1.9) \\
&= \|\widehat{X} - \widehat{Y}\|_F^2 + 2\|d - c\|^2 + |\gamma - \beta|^2.
\end{aligned}$$

To complete the proof we observe that (1.9) is minimized if and only if $c = d, \gamma = \beta$ and $\widehat{Y} \in P_M\widehat{X}$. $\qquad \square$

The set M in Theorem 3 is a set of low-rank positive semi-definite matrices. One method to compute its projection (and the one we will use) is by exploiting the eigen-decomposition of $\widehat{X}$. Denote by $\mathrm{diag}(\lambda)$ the diagonal matrix given by placing the elements of the vector $\lambda \in \mathbb{R}^m$ along the main diagonal. Let $\widehat{X} = U\,\mathrm{diag}(\lambda)U^T$ be an eigen-decomposition (of $\widehat{X}$) with

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_q^+ \geq \cdots \geq \lambda_m.$$

A projection onto the set is then given by

$$U\,\mathrm{diag}((\lambda_1^+, \lambda_2^+, \ldots, \lambda_q^+, 0\ldots, 0, 0))U^T,$$

where x^+ denotes $\max\{0, x\}$.

1.7 Computational Experiments

We apply the formulation of Section 1.6 to six proteins, shown in Table 1.1, obtained from the *RCSB Protein Data Bank*[2]. As part of [3], reconstructions of the same six proteins were attempted using a partial EDM containing only distances less than 6Å. Here we attempt reconstructions using partial EDMs which, in addition to these short-range distances, incorporate other *a priori* information. In particular, we include inter-atomic distances greater than 6Å for atoms from within the same residue in the partial EDM. This is reasonable since the structure of the individual residues is known. For 1PTQ, this information gives approximately a further 0.2% of the total non-zero inter-atomic distances.

Table 1.1 Number of atoms, residues, known, and total non-zero inter-atomic distances in our six test proteins.

Protein	Atoms	Residues	Total Non-Zero Distances	Known Non-Zero Distances
1PTQ	404	50	81,406	8.9207%
1HOE	581	74	168,490	6.4105%
1LFB	641	99	205,120	5.6362%
1PHT	988	85	236,328	4.6501%
1POA	1067	118	568,711	3.6375%
1AX8	1074	146	576,201	3.5606%

Our experiments were implemented in *Cython* and performed on a machine having an Intel Xeon E5540 @ 2.83GHz running Red Hat Enterprise Linux 6.5. A combination of the Cython platform, and optimized code gave approximately a ten-fold speed up compared to [3]. This allowed for a greater number of iterations to be performed and hence the use of the more robust (albeit still heuristic) stopping criterion given in Algorithm 1.3 as opposed to simply performing a fixed number of iterations. The reconstructed EDM, x, was converted to points $z_1, z_2, \ldots, z_m \in \mathbb{R}^3$ using Algorithm 1.4.

Remark 4. It is worth emphasing that our primary concern is the quality of the reconstruction, rather than the time required to perform the reconstruction. This is because, if done well, one only needs to determine the conformation once.

We report two error metrics, which we now explain. Denote the actual EDM by x^{actual}. The first error metric is a measure of the *error in the reconstructed EDM*, and is given by

$$\text{EDM-error} = \|x^{\text{actual}} - x\|_F = \sqrt{\sum_{i,j=1}^{m} \left| x_{ij}^{\text{actual}} - x_{ij} \right|^2}.$$

---

[2] RCSB Protein Data Bank: `www.rcsb.org/pdb`

Fig. 1.4 Conversion of EDM to points in $\mathbb{R}^q$.

```
Input: x ∈ X ;                        /* a Euclidean distance matrix */
L = I − ee^T /n where e = (1,1,...,1)^T ;
τ = −LDL/2;
USV^T = SingularValueDecomposition (τ);
Z = first q columns of U√S;
z_i = ith row of Z for i = 1,2,...,m;
Output: z_1,z_2,...,z_q ∈ ℝ^q ;       /* points corresponding to x */
```

Denote the actual atom positions by $z_1^{\text{actual}}, z_2^{\text{actual}}, \ldots, z_m^{\text{actual}} \in \mathbb{R}^3$. The second error metric measures the *error in the reconstructed atom positions* $z_1, z_2, \ldots, z_m \in \mathbb{R}^3$. Since EDMs are invariant under translation, reflection, and rotation of the points by which they are induced, we first perform a *Procrustes analysis* [16] to obtain $\widetilde{z_1}, \widetilde{z_2}, \ldots, \widetilde{z_m} \in \mathbb{R}^3$. These points are a best fit of the reconstructed points when the aforementioned transformations are allowed. The second error metric is given by

$$\text{Position-error} = \sqrt{\sum_{k=1}^{m} \|z_k^{\text{actual}} - \widetilde{z_k}\|_2^2}.$$

When comparing the relative size of these two errors, it is worth noting that the summation in the EDM-error contains m^2 terms whereas the summation in the position-error contains only $3m$.

Remark 5 (Decibel error). It is also common to consider the relative error in *decibels (dB)*, as was reported in [3]. That is,

$$\text{Relative error (dB)} = 10\log_{10}\left(\frac{\|P_B R_A x - P_A x\|_F^2}{\|P_A x\|_F^2}\right).$$

In this study the relative error in decibels is not reported. This is unnecessary because the stopping criterion used in Algorithm 1.3 is equivalent to requiring that the decibel error be less than $10\log_{10}(\varepsilon^2)$. Requiring that $\varepsilon = 10^{-5}$ corresponds to aiming at a relative error of -100dB.

Remark 6 (Stopping criterion and tolerance). In the computational experiments that follow, the stopping tolerance is taken to be $\varepsilon = 10^{-5}$. We now provide some justification for this choice.

For each of the six proteins, Figure 1.5 shows the relative error as a function of the number of iterations starting from a given initial point for the Douglas–Rachford method.

- When the number of iteration is less than 5000 the relative error exhibits non-monotone oscillatory behaviour — which seems to provide much of the potency
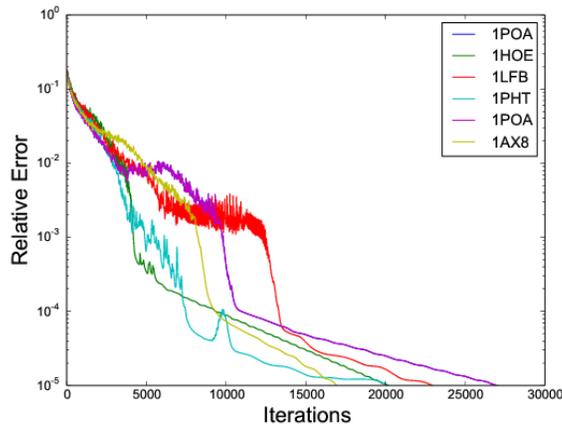
Fig. 1.5 The relative error as a function of iterations (vertical axis is logarithmic).

of the method. It seems to allow the reflection method to sample regions and avoid settling at an inferior local minimum of the configuration space. In [3] we observed that the alternating projection method, which is monotonic, fails to produce good reconstructions.

- When the relative error is between 10^{-3} and 10^{-4}, it decreases sharply after which a period of more predictable decrease is observed.
- Beyond this point slower progress is made. We therefore choose our stopping tolerance to be $\varepsilon = 10^{-5}$ so that the algorithm will terminate in this region.

The change in successive iterates was found to also exhibit similar behavior (not shown), so is another suitable candidate for a stopping criterion.

It is worth noting that there are many other techniques for solving (variants of) the protein conformation problem (see for instance [21]). Such a discussion, however, is beyond the scope of this chapter.

1.7.1 Basic Douglas–Rachford Algorithm Results

Table 1.2 gives results for the basic Douglas–Rachford algorithm presented in Algorithm 1.3. We make some comments regarding these results.

The EDM-error increases with increasing problem size; yet the same trend is not observed for the position-error for which 1PHT reported the largest error. For all of the proteins studied, the differences between the average and worst case results for the position-errors were small. This strongly suggests that the method can consistently produce a EDM which gives the desired atomic positions.

The second column of Figure 1.6 shows the conformation of the basic Douglas–Rachford reconstructions, which are visually indistinguishable from the actual conformation shown in the first column. This is an improvement from what was reported in [3] whose Douglas–Rachford reconstructions of two of the larger proteins, 1POA and 1AX8, gave unrealistic conformations consisting of disjoint blocks of atoms. In light of Remark 6 it is likely that this was due to premature algorithm termination.

Table 1.2 Average (worst) results from five random replications of the basic Douglas–Rachford algorithm with $\varepsilon = 10^{-5}$.

Protein	EDM-Error		Position-Error		Iterations		Time (h)	
1PTQ	3.6816	(4.0938)	0.1307	(0.1457)	4339.6	(4686)	0.28	(0.30)
1HOE	9.7475	(13.8503)	0.1781	(0.2636)	20794.4	(21776)	3.50	(3.67)
1LFB	9.8728	(17.2860)	1.1388	(2.1177)	22346.2	(23295)	4.64	(4.85)
1PHT	10.3709	(12.9557)	12.8782	(13.0056)	20103.0	(20251)	13.90	(14.00)
1POA	25.4225	(46.5804)	0.5844	(1.1639)	28426.0	(29766)	23.33	(24.47)
1AX8	25.7369	(39.4586)	0.6592	(0.9160)	17969.8	(19059)	15.04	(15.95)

1.7.2 Douglas–Rachford Algorithm with Periodic Rank Projections

In our formulation of the protein confirmation problem, the most expensive step is the computation of the projection onto the rank constraint C_2. Thus requires the eigen-decomposition of a $(m-1) \times (m-1)$ symmetric matrix. In this section we propose problem specific heuristics which allow for this computation to sometimes be avoided.

One idea to avoid performing the eigen-decomposition is to not update the sequence $(r_n)_{n=1}^{\infty}$ in Algorithm 1.3 at every iteration but only periodically. This approach is described in Algorithm 3, and results, with updates only every third time, in Table 1.3.

We now compare the results of this section to those of Section 1.7.1. A small increase in the position-errors, and a larger increase in the EDM-errors was observed. The number of iterations required also increased, with this number almost doubling for 1PTQ. For all six test proteins, the total time required was less. The biggest improvement was 1POA whose total time was more than halved. The quality of the reconstructed conformations seem not to be adversely effected by the use of periodic rank projections, as can be seen in Figure 1.6.
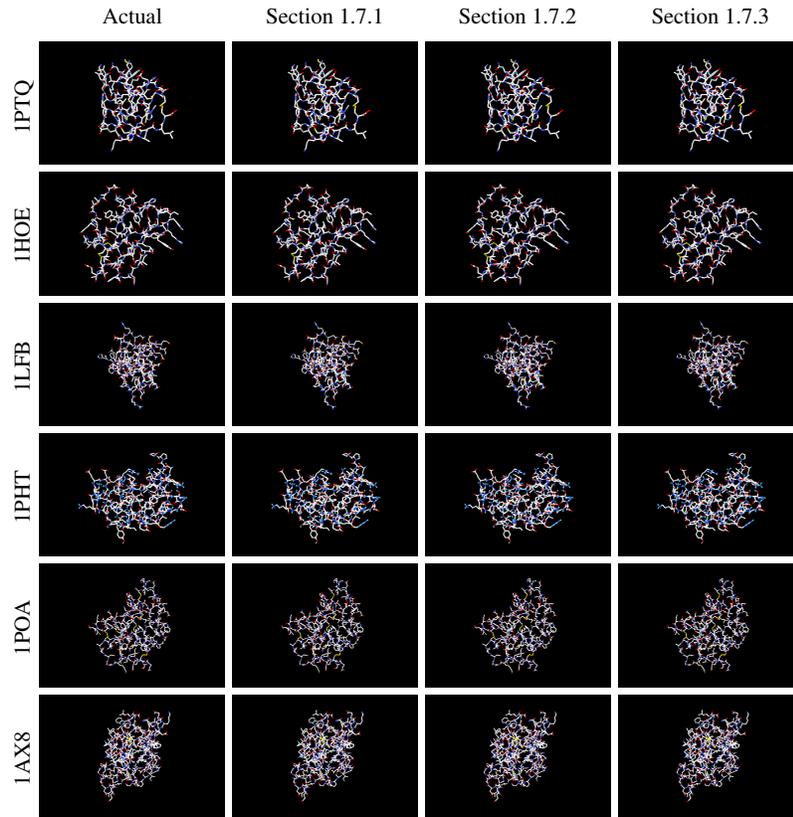
	Actual	Section 1.7.1	Section 1.7.2	Section 1.7.3

Fig. 1.6 The conformations of the six proteins, and their three Douglas–Rachford reconstructions.

Fig. 1.7 The Douglas–Rachford algorithm with T-periodic projections onto the set B.

```
Input: x_0 ∈ X, T ∈ ℕ and ε > 0
n = 0;
p_0 ∈ P_A(x_0);
while n = 0 or ||r_n − p_n|| > ε||p_n|| do
    if n mod T = 0 then
        | r_n ∈ P_B(2p_n − x_n);
    else
        | r_n = r_{n−1};
    end
    x_{n+1} = x_n + r_n − p_n;
    p_{n+1} ∈ P_A(x_{n+1});
    n = n + 1;
end
Output: p_n ∈ X
```

Table 1.3 Average (worst) results from five random replications of the Douglas–Rachford algorithm with periodic rank projections with $T = 3$ and $\varepsilon = 10^{-5}$.

Protein	EDM-Error		Position-Error		Iterations		Time (h)	
1PTQ	4.3709	(4.7200)	0.1919	(0.2240)	7160.6	(7595)	0.16	(0.17)
1HOE	10.1790	(12.1089)	0.2603	(0.2933)	20305.4	(22550)	1.21	(1.35)
1LFB	17.6532	(19.0984)	1.2709	(1.7243)	28983.8	(31211)	2.15	(2.31)
1PHT	23.8594	(25.9794)	13.1358	(13.2805)	20559.2	(20981)	5.03	(5.13)
1POA	49.8406	(51.3411)	1.0948	(1.2084)	33150.8	(39083)	9.55	(11.25)
1AX8	45.5203	(49.1866)	1.1696	(1.4482)	27080.6	(31250)	7.96	(9.20)

1.7.3 Reconstructions with Additional Distance Data

In Sections 1.7.1 & 1.7.2 we considered the physically realistic setting in which distances below the threshold of 6Å were known. As noted, when the number of atoms in a protein increases, the proportion of inter-atomic distances below this threshold compared to the total number of (non-zero) distances decreases.

To better understand the Douglas–Rachford method applied to larger problem instances, we performed the same reconstruction as in Section 1.7.1 but with the percentage of known non-zero distances constant. More precisely, we assumed that the smallest 10% of inter-atomic distances were known.

Table 1.4 Average (worst) results from five random replications of the basic Douglas–Rachford algorithm from the smallest 10% of inter-atomic distances with $\varepsilon = 10^{-5}$.

Protein	EDM-Error		Position-Error		Iterations		Time (h)	
1PTQ	3.1924	(3.5936)	0.0963	(0.1213)	4014.4	(4184)	0.26	(0.27)
1HOE	8.0905	(10.4357)	0.0960	(0.1265)	15110.4	(15709)	2.54	(2.64)
1LFB	7.2941	(13.9893)	0.4647	(0.9182)	11060.6	(11912)	2.29	(2.46)
1PHT	14.1302	(20.2476)	0.3542	(0.4326)	6071.0	(6512)	4.19	(4.49)
1POA	19.5619	(31.1987)	0.1624	(0.2665)	11555.8	(13244)	9.44	(10.81)
1AX8	14.0747	(29.7259)	0.0940	(0.1922)	10099.2	(11125)	8.38	(9.23)

As could perhaps be predicted, when more distance information is incorporated the error metrics, and the number of iterations decrease. Problem size and EDM-error do not correlate as strongly compared to the results of Section 1.7.1. However, the general trend that larger problem sizes give larger EDM-errors is still observed. The most notable improvement, when compared to Section 1.7.1, is the position-error for 1PHT. This suggests that in the realistic setting of Section 1.7.1 the underlying protein's conformation (*e.g.,* a compact or a dispersed conformation) is an important factor in the difficulty of the reconstruction problem.

1.7.4 Ionic Liquid Bulk Structure Determination

Ionic liquids (ILs) are salts (*i.e.,* they are comprised of positively and negatively charged ions) having low melting points, typically occupying the liquid state at room temperature. An analogous reconstruction problem arising in the context of ionic liquid chemistry is to determine a given ionic liquid's *bulk structure*. That is, the configuration of its ions with respect to each other (the structure of the individual ions is known).

In this section, we applied the Douglas–Rachford method to a simplified version of this problem. Entries of the partial EDM are assumed to be known whenever the two atoms are bonded (*i.e.,* when their *Van der Waals radii* taken from [8] overlap).

Table 1.5 reports results for a *propylammonium nitrate (PAN)* data set consisting of 180 atoms. The corresponding rank-3 EDM completion problem has a total of 32,220 non-zero inter-atomic distances of which 5.95% form the partial EDM.

Table 1.5 Average (worst) results from five random replications of the basic Douglas–Rachford algorithm, applied to ionic liquid bulk structure determination, with $\varepsilon = 10^{-5}$.

EDM-Error		Position-Error		Iterations		Time (h)	
0.6323	(0.6918)	2.0374	(2.5039)	41553.2	(82062)	0.22	(0.43)

As was the case in the protein conformation application, the difference between the average and worst case results for the two error metrics is observed to be small. The actual conformation of PAN, and its Douglas–Rachford reconstruction are shown in Figure 1.8. A high degree of visual coincidence is observed, although a small amount of the finer detail is missing.
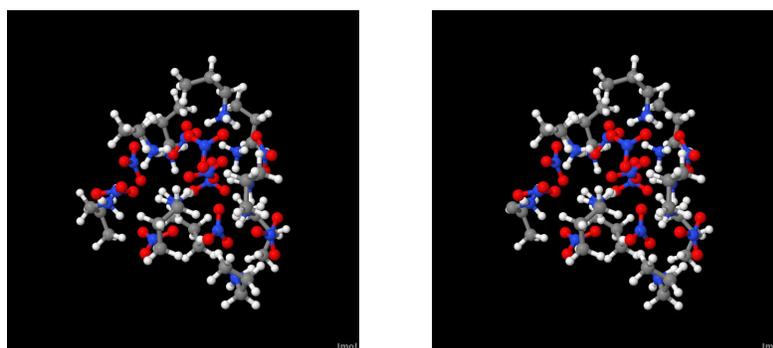

Fig. 1.8 The actual conformation (left) and Douglas–Rachford reconstruction (right) of PAN. Note the two poorly reconstructed hydrogen atoms (white) in the left configuration.

1.8 Concluding Remarks

We have shown that the Douglas–Rachford reflection method can successfully solve the protein conformation determination problem by directly addressing a non-convex matrix completion problem. This is also the case for an analogous ionic liquid bulk structure determination problem. It is worth emphasising again that the current literature provides no theoretical justification for the method to work at all, let alone so well. Modifications of the method have also been shown to reduce computational times without significantly effecting the quality of the results. This promising demonstration of the method begs further attention, both in improving theoretical understanding, and in the refinement and investigation of these and further applications.

References

1. Aragón Artacho, F., Borwein, J.: Global convergence of a non-convex Douglas–Rachford iteration. J. Glob. Optim. **57**(3), 753–769 (2013).
2. Aragón Artacho, F., Borwein, J., Tam, M.: Recent results on Douglas–Rachford methods for combinatorial optimization problems. J. Optim. Theory Appl. (in press, 2013).
3. Aragón Artacho, F., Borwein, J., Tam, M.: Douglas–Rachford feasibility methods for matrix completion problems. ANZIAM J. (in press, 2014).
4. Bauschke, H., Bello Cruz, J., Nghia, T., Phan, H., Wang, X.: The rate of linear convergence of the Douglas–Rachford algorithm for subspaces is the cosine of the Friedrichs angle. J. Approx. Theory 185, 63–79 (2014).
5. Bauschke, H., and Combettes, P.: Convex analysis and monotone operator theory in Hilbert space. Springer, New York (2011).
6. Bauschke, H., Combettes, P., Luke, D.: Finding best approximation pairs relative to two closed convex sets in Hilbert spaces. J. Approx. Theory **127**(2), 178–192 (2004).
7. Bauschke, H., Noll, D., Phan, H.: Linear and strong convergence of algorithms involving averaged nonexpansive operators. arXiv preprint arXiv:1402.5460 (2014).
8. Bondi, A.: Van der Waals Volumes and Radii. J. Phys. Chem. **68**(3):441–51 (1964).
9. Borwein, J., Lewis, A.: Convex analysis and nonlinear optimization. Springer (2006).
10. Borwein, J., Sims, B.: The Douglas–Rachford algorithm in the absence of convexity. In: Fixed-Point Algorithms for Inverse Problems in Science and Engineering, pp. 93–109. Springer (2011).
11. Borwein, J., Tam, M.: The cyclic Douglas–Rachford method for inconsistent feasibility problems. J. Nonlinear Convex Analysis, accepted March 2014. arXiv preprint arXiv:1310.2195 (2013).
12. Borwein, J., Tam, M.: A cyclic Douglas–Rachford iteration scheme. J. Optim. Theory Appl. **160**(1), 1–29 (2014).
13. Borwein, J., Zhu, Q.: Techniques of Variational Analysis, *CMS Books in Mathematics*, vol. 20. Springer-Verlag, New York (2005, Paperback, 2010).
14. Berman, A., Shaked-Monderer, N.: Completely positive matrices. World Scientific, Singapore (2003).

15. Cegielski, A.: Iterative methods for fixed point problems in Hilbert space, *Lecture Notes in Mathematics*, vol. 2057. Springer, London (2012).
16. Dattorro, J.: Convex optimization & Euclidean distance geometry. Meboo Publishing USA (2005).
17. Elser, V., Rankenburg, I., Thibault, P.: Searching with iterated maps. Proc. Natl. Acad. Sci. **104**(2), 418–423 (2007).
18. Gravel, S., Elser, V.: Divide and concur: A general approach to constraint satisfaction. Phys. Rev. E **78**(3), 036,706 (2008).
19. Hayden, T., Wells, J.: Approximation by matrices positive semidefinite on a subspace. Linear Algebra Appl. **109**, 115–130 (1988).
20. Hesse, R., Luke, D.: Nonconvex notions of regularity and convergence of fundamental algorithms for feasibility problems. SIAM J. Optim. **23**(4), 2397–2419 (2013).
21. Seo, J., Kim, J.-K., Ryu, J., Lavor, C., Mucherino, A., and Kim, D.-S.: BetaMDGP: Protein structure determination algorithm based on the Beta-complex. Trans. Comput. Sc. **8360**, 130–155 (2014).