Reflection Methods for Inverse Problems

Matthew K. Tam
Joint work with Francisco Aragón Artacho and Jonathan Borwein

School of Mathematical and Physical Sciences
University of Newcastle, Australia

With generous support from the CSIRO/ANZIAM SSS

Computational Techniques and Applications Conference (ANU)
December 1–3, 2014

Introduction: Projection Methods

Projection methods are a family of iterative algorithms useful for solving the feasibility problem which asks:

$$\text{find } x \in C_1 \cap C_2 \subseteq \mathcal{H},$$

where C_1 and C_2 are constraint sets in a Hilbert space $\mathcal{H}$.

The focus of this talk is application of the Douglas–Rachford method as a heuristic for non-convex feasibility problems guided by convex theory.

Recall that a set S is convex if, $\lambda x + (1-\lambda)y \in S$, $(\forall x, y \in S)(\forall \lambda \in [0,1])$.

Introduction: Projection Methods

Projection methods are a family of iterative algorithms useful for solving the feasibility problem which asks:

$$\text{find } x \in C_1 \cap C_2 \subseteq \mathcal{H},$$

where C_1 and C_2 are constraint sets in a Hilbert space $\mathcal{H}$.

- At each stage, employ (nearest point) projections w.r.t. the individual constraint sets. The solution is obtained in the limit.

The focus of this talk is application of the Douglas–Rachford method as a heuristic for non-convex feasibility problems guided by convex theory.

Recall that a set S is convex if, $\lambda x + (1-\lambda)y \in S$, $(\forall x, y \in S)(\forall \lambda \in [0,1])$.

Introduction: Projection Methods

Projection methods are a family of iterative algorithms useful for solving the feasibility problem which asks:

$$\text{find } x \in C_1 \cap C_2 \subseteq \mathcal{H},$$

where C_1 and C_2 are constraint sets in a Hilbert space $\mathcal{H}$.

- At each stage, employ (nearest point) projections w.r.t. the individual constraint sets. The solution is obtained in the limit.
- For (closed) convex constraint sets, behavior is *fairly* well understood – the methods can be analyzed using non-expansivity properties of convex projection operators.

The focus of this talk is application of the Douglas–Rachford method as a heuristic for non-convex feasibility problems guided by convex theory.

Recall that a set S is convex if, $\lambda x + (1-\lambda)y \in S$, $(\forall x, y \in S)(\forall \lambda \in [0,1])$.

Introduction: Projection Methods

Projection methods are a family of iterative algorithms useful for solving the feasibility problem which asks:

$$\text{find } x \in C_1 \cap C_2 \subseteq \mathcal{H},$$

where C_1 and C_2 are constraint sets in a Hilbert space $\mathcal{H}$.

- At each stage, employ (nearest point) projections w.r.t. the individual constraint sets. The solution is obtained in the limit.
- For (closed) convex constraint sets, behavior is *fairly* well understood – the methods can be analyzed using non-expansivity properties of convex projection operators.
- When one or more of the constraint sets are non-convex, theory is largely unknown. However, one particular projection method, the Douglas–Rachford method, has been (experimentally) observed to successfully solve a large range of non-convex problems.

The focus of this talk is application of the Douglas–Rachford method as a heuristic for non-convex feasibility problems guided by convex theory.

Recall that a set S is convex if, $\lambda x + (1-\lambda)y \in S$, $(\forall x, y \in S)(\forall \lambda \in [0,1])$.

Introduction: Projection Methods

Projection methods are a family of iterative algorithms useful for solving the feasibility problem which asks:

$$\text{find } x \in C_1 \cap C_2 \subseteq \mathcal{H},$$

where C_1 and C_2 are constraint sets in a Hilbert space $\mathcal{H}$.

- At each stage, employ (nearest point) projections w.r.t. the individual constraint sets. The solution is obtained in the limit.
- For (closed) convex constraint sets, behavior is *fairly* well understood – the methods can be analyzed using non-expansivity properties of convex projection operators.
- When one or more of the constraint sets are non-convex, theory is largely unknown. However, one particular projection method, the Douglas–Rachford method, has been (experimentally) observed to successfully solve a large range of non-convex problems. Examples:
  - Solving Sudoku and nonogram puzzles, 8-queens and generalizations, enumerating Hadamard matrices, phase retrieval & ptychography, ...

The focus of this talk is application of the Douglas–Rachford method as a heuristic for non-convex feasibility problems guided by convex theory.

Recall that a set S is convex if, $\lambda x + (1-\lambda)y \in S$, $(\forall x, y \in S)(\forall \lambda \in [0,1])$.

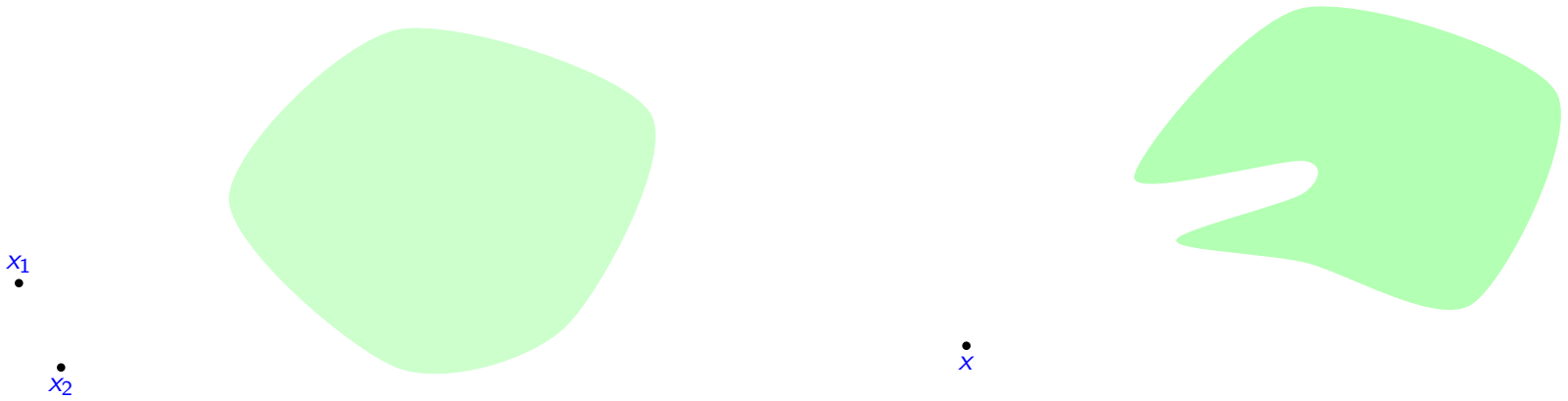
Introduction: Variational Tools

Let $S \subseteq \mathcal{H}$. The (nearest point) projection onto S is the (set-valued) mapping,

$$P_S x := \arg\min_{s \in S} \|s - x\|.$$

The reflection w.r.t. S is the (set-valued) mapping,

$$R_S := 2P_S - I.$$



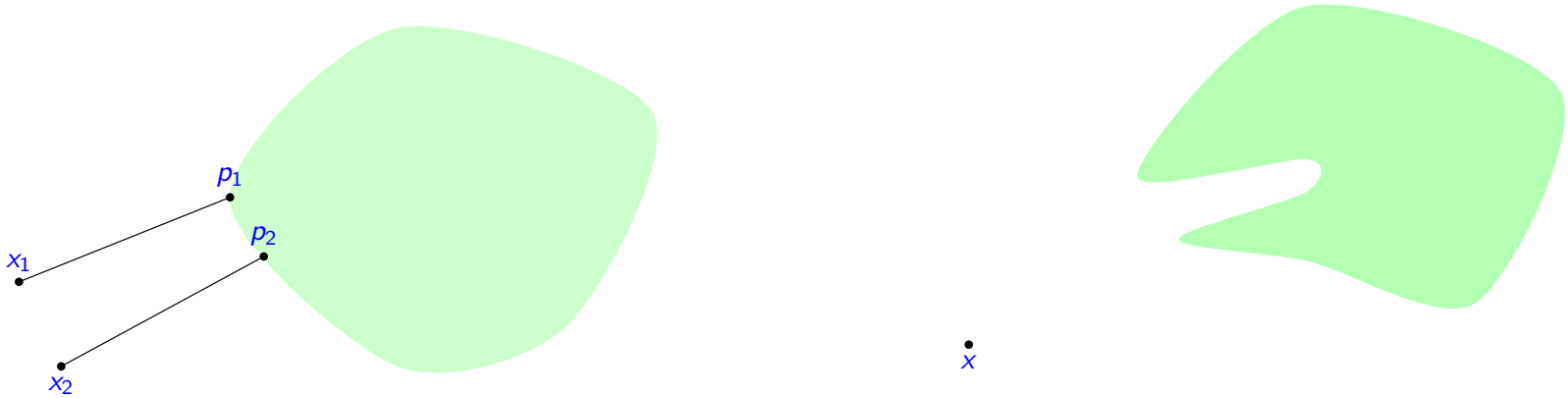
Introduction: Variational Tools

Let $S \subseteq \mathcal{H}$. The (nearest point) projection onto S is the (set-valued) mapping,

$$P_S x := \arg\min_{s \in S} \|s - x\|.$$

The reflection w.r.t. S is the (set-valued) mapping,

$$R_S := 2P_S - I.$$



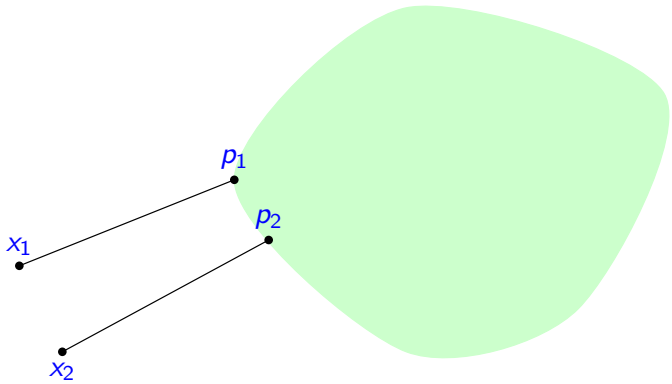
Introduction: Variational Tools

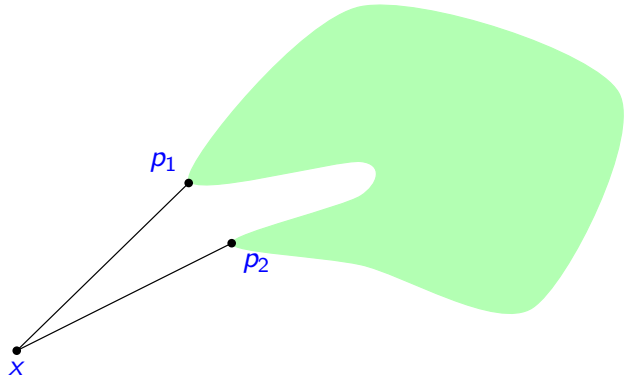
Let $S \subseteq \mathcal{H}$. The (nearest point) projection onto S is the (set-valued) mapping,

$$P_S x := \arg\min_{s \in S} \|s - x\|.$$

The reflection w.r.t. S is the (set-valued) mapping,

$$R_S := 2P_S - I.$$





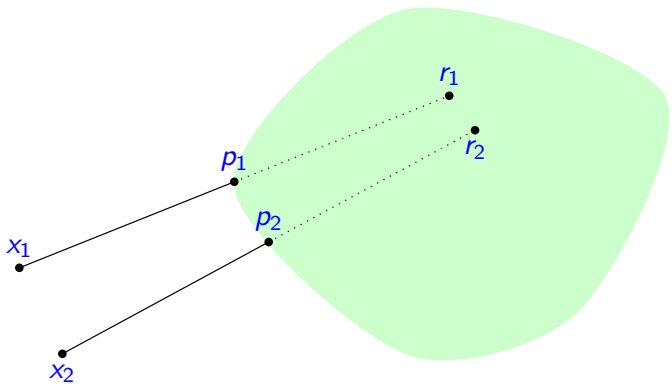
Introduction: Variational Tools

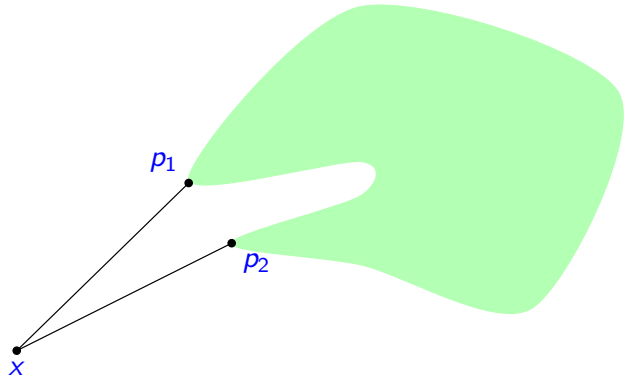
Let $S \subseteq \mathcal{H}$. The (nearest point) projection onto S is the (set-valued) mapping,

$$P_S x := \arg\min_{s \in S} \|s - x\|.$$

The reflection w.r.t. S is the (set-valued) mapping,

$$R_S := 2P_S - I.$$





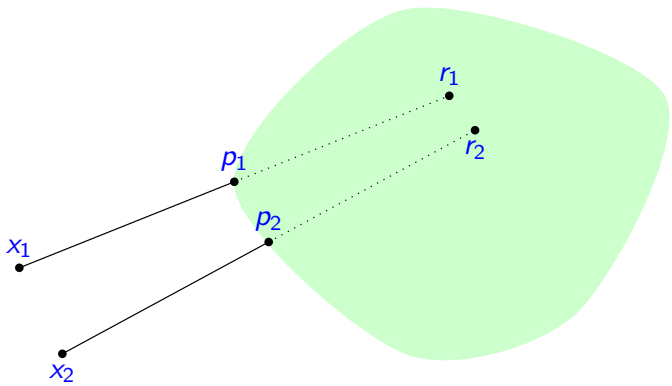
Introduction: Variational Tools

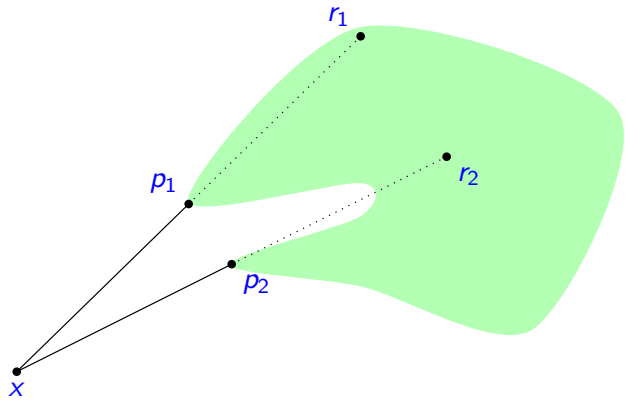
Let $S \subseteq \mathcal{H}$. The (nearest point) projection onto S is the (set-valued) mapping,

$$P_S x := \arg\min_{s \in S} \|s - x\|.$$

The reflection w.r.t. S is the (set-valued) mapping,

$$R_S := 2P_S - I.$$



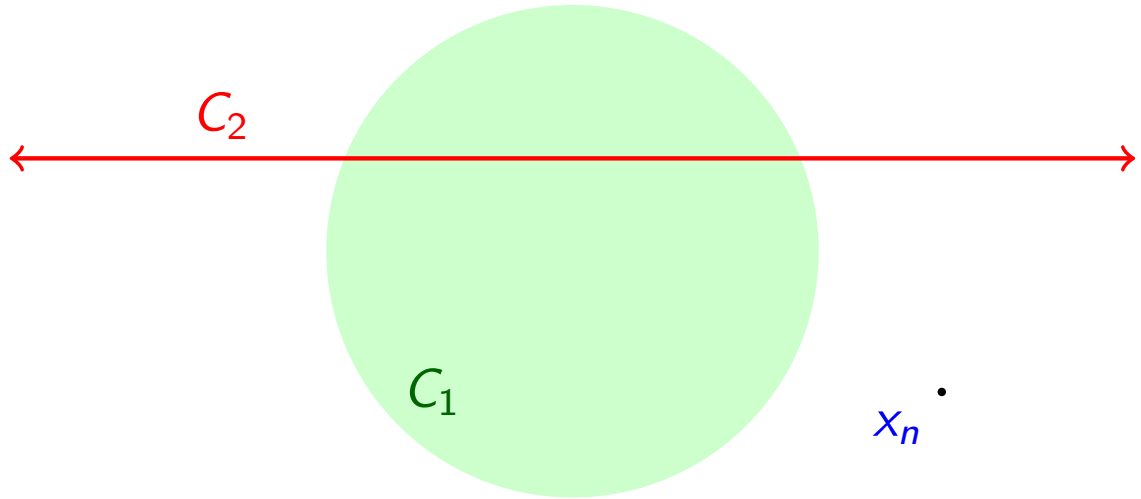


The Douglas–Rachford Algorithm

Given an initial point $x_0 \in \mathcal{H}$, the Douglas–Rachford method is the fixed-point iteration given by

$$x_{n+1} = T_{C_1,C_2} x_n \quad \text{where} \quad T_{C_1,C_2} := \frac{Id + R_{C_2} R_{C_1}}{2}.$$

- If x is a fixed point of T_{C_1,C_2} then $P_{C_1} x \in C_1 \cap C_2$.



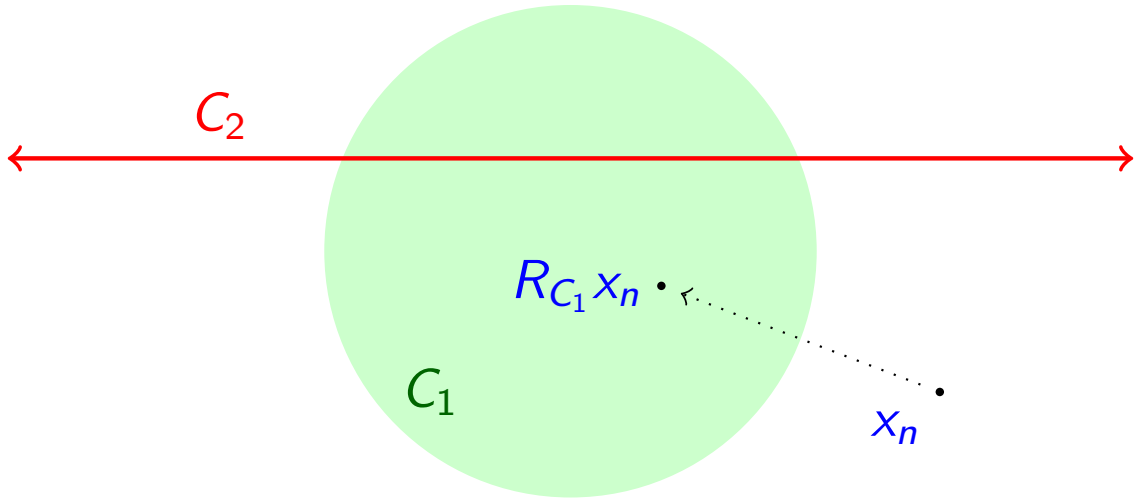
$$C_1 = \{x \in \mathcal{H} : \|x\| \leq 1\}, \quad C_2 = \{x \in \mathcal{H} : \langle a, x \rangle = b\}.$$

The Douglas–Rachford Algorithm

Given an initial point $x_0 \in \mathcal{H}$, the Douglas–Rachford method is the fixed-point iteration given by

$$x_{n+1} = T_{C_1,C_2} x_n \quad \text{where} \quad T_{C_1,C_2} := \frac{Id + R_{C_2} R_{C_1}}{2}.$$

- If x is a fixed point of T_{C_1,C_2} then $P_{C_1} x \in C_1 \cap C_2$.



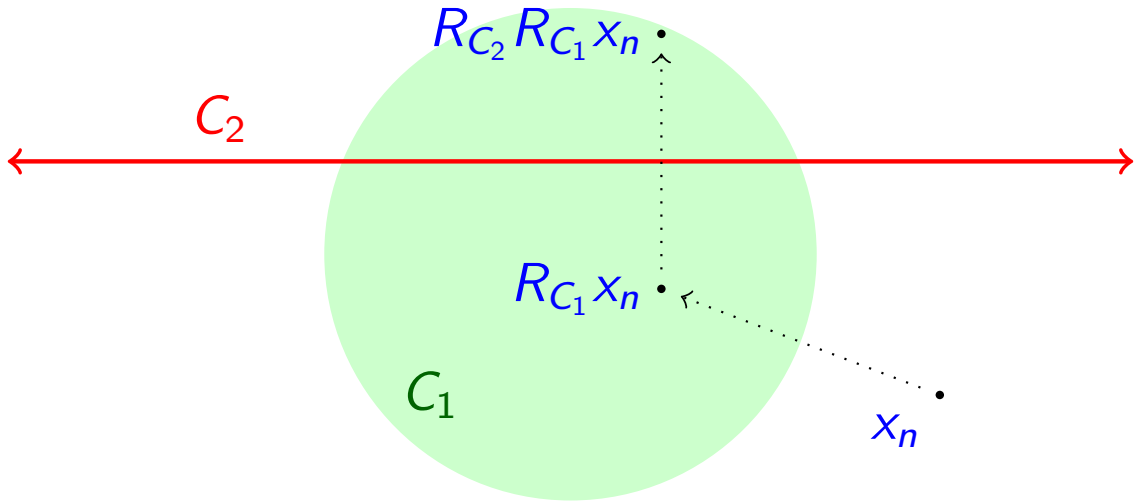
$$C_1 = \{x \in \mathcal{H} : \|x\| \leq 1\}, \quad C_2 = \{x \in \mathcal{H} : \langle a, x \rangle = b\}.$$

The Douglas–Rachford Algorithm

Given an initial point $x_0 \in \mathcal{H}$, the Douglas–Rachford method is the fixed-point iteration given by

$$x_{n+1} = T_{C_1,C_2} x_n \quad \text{where} \quad T_{C_1,C_2} := \frac{Id + R_{C_2} R_{C_1}}{2}.$$

- If x is a fixed point of T_{C_1,C_2} then $P_{C_1} x \in C_1 \cap C_2$.



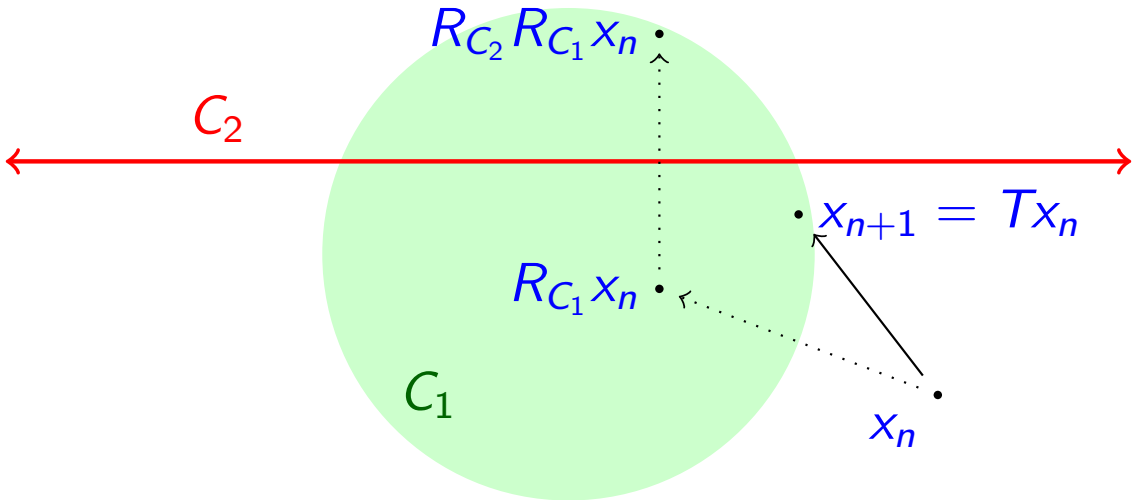
$$C_1 = \{x \in \mathcal{H} : \|x\| \leq 1\}, \quad C_2 = \{x \in \mathcal{H} : \langle a, x \rangle = b\}.$$

The Douglas–Rachford Algorithm

Given an initial point $x_0 \in \mathcal{H}$, the Douglas–Rachford method is the fixed-point iteration given by

$$x_{n+1} = T_{C_1,C_2} x_n \quad \text{where} \quad T_{C_1,C_2} := \frac{Id + R_{C_2} R_{C_1}}{2}.$$

- If x is a fixed point of T_{C_1,C_2} then $P_{C_1} x \in C_1 \cap C_2$.



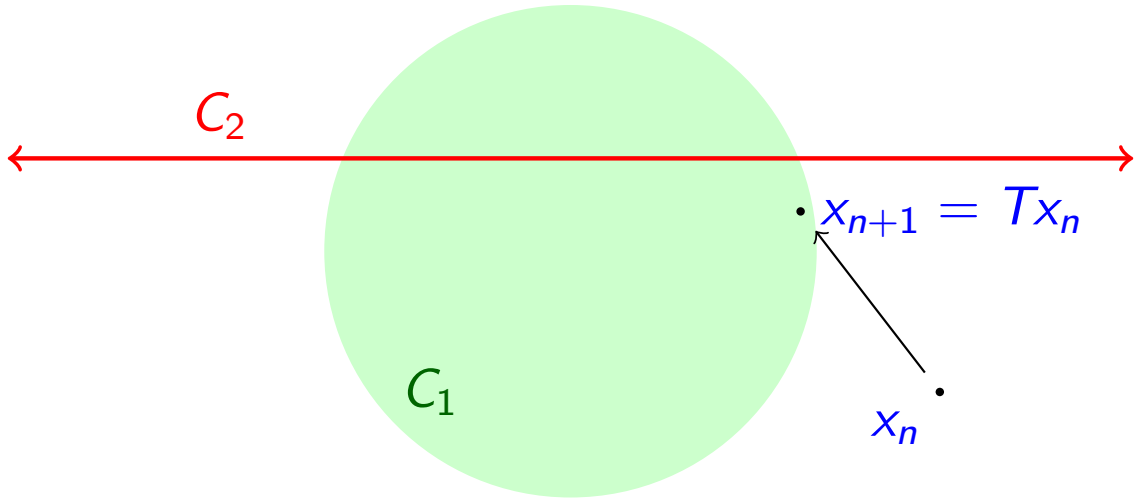
$$C_1 = \{x \in \mathcal{H} : \|x\| \leq 1\}, \quad C_2 = \{x \in \mathcal{H} : \langle a, x \rangle = b\}.$$

The Douglas–Rachford Algorithm

Given an initial point $x_0 \in \mathcal{H}$, the Douglas–Rachford method is the fixed-point iteration given by

$$x_{n+1} = T_{C_1,C_2} x_n \quad \text{where} \quad T_{C_1,C_2} := \frac{Id + R_{C_2} R_{C_1}}{2}.$$

- If x is a fixed point of T_{C_1,C_2} then $P_{C_1} x \in C_1 \cap C_2$.



$$C_1 = \{x \in \mathcal{H} : \|x\| \leq 1\}, \quad C_2 = \{x \in \mathcal{H} : \langle a, x \rangle = b\}.$$

The Douglas–Rachford Algorithm

- First studied by Douglas & Rachford (1956) in connection with heat conduction problems, and later by Lions & Mercier (1979) for finding a zero in the sum of two maximal monotone operators.

Theorem (Basic behaviour of the Douglas–Rachford method)

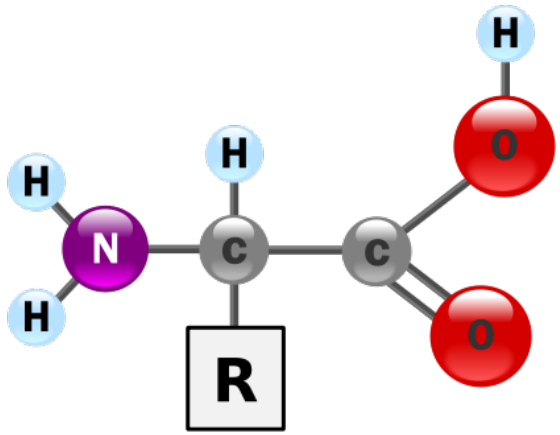
Suppose C_1, C_2 are closed convex subsets of a finite dimensional Hilbert space $\mathcal{H}$. For any $x_0 \in \mathcal{H}$, define $x_{n+1} = T_{C_1,C_2} x_n$.

1. If $C_1 \cap C_2 \neq \emptyset$, then $x_n \to x$ such that $P_{C_1} x \in C_1 \cap C_2$.
2. If $C_1 \cap C_2 = \emptyset$, then $\|x_n\| \to +\infty$.

- It is important to monitor the shadow sequence $(P_{C_1} x_n)_{n=1}^{\infty}$, not just the iterates $(x_n)_{n=1}^{\infty}$.

Protein Confirmation Determination and EDMs

Proteins are large biomolecules comprising of multiple amino acid chains.



Generic amino acid

Myoglobin

They participate in virtually every cellular process, and knowledge of structural conformation gives insights into the mechanisms by which they perform.

Protein Confirmation Determination and EDMs

One technique that can be used to determine conformation is nuclear magnetic resonance (NMR) spectroscopy. However, NMR is only able to resolve short inter-atomic distances (*i.e.*, $< 6\text{Å}$).

Protein Confirmation Determination and EDMs

One technique that can be used to determine conformation is nuclear magnetic resonance (NMR) spectroscopy. However, NMR is only able to resolve short inter-atomic distances (*i.e.*, < 6Å). For 1PTQ (404 atoms) this corresponds to $< 8\%$ of the total inter-atomic distances.

Protein Confirmation Determination and EDMs

One technique that can be used to determine conformation is nuclear magnetic resonance (NMR) spectroscopy. However, NMR is only able to resolve short inter-atomic distances (*i.e.*, < 6Å). For 1PTQ (404 atoms) this corresponds to $< 8\%$ of the total inter-atomic distances.

We say $D = (D_{ij}) \in \mathbb{R}^{m\times m}$ is a Euclidean distance matrix (EDM) if there exists points $p_1, \ldots, p_m \in \mathbb{R}^q$ such that

$$D_{ij} = \|p_i - p_j\|^2.$$

When this holds for points in $\mathbb{R}^q$, we say that D is embeddable in $\mathbb{R}^q$.

Protein Confirmation Determination and EDMs

One technique that can be used to determine conformation is nuclear magnetic resonance (NMR) spectroscopy. However, NMR is only able to resolve short inter-atomic distances (*i.e.*, < 6Å). For 1PTQ (404 atoms) this corresponds to $< 8\%$ of the total inter-atomic distances.

We say $D = (D_{ij}) \in \mathbb{R}^{m\times m}$ is a Euclidean distance matrix (EDM) if there exists points $p_1, \ldots, p_m \in \mathbb{R}^q$ such that

$$D_{ij} = \|p_i - p_j\|^2.$$

When this holds for points in $\mathbb{R}^q$, we say that D is embeddable in $\mathbb{R}^q$.

We formulate protein reconstruction as a matrix completion problem:

> *Find a member from a given family of matrices,*
> *knowing only a subset of its entries.*

Find a EDM, embeddable in $\mathbb{R}^3$, knowing only short inter-atomic distances.

A Feasibility Problem Formulation

Denote by Q the Householder matrix defined by

$$Q := I - \frac{2vv^T}{v^T v}, \text{ where } v = \left[1, 1, \ldots, 1, 1 + \sqrt{m}\right]^T \in \mathbb{R}^m.$$

Theorem (Hayden–Wells 1988)

A nonnegative, symmetric, hollow matrix X, is a EDM iff $\widehat{X} \in \mathbb{R}^{(m-1)\times(m-1)}$ in

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

is positive semi-definite (PSD). In this case, X is embeddable in $\mathbb{R}^q$ where $q = \text{rank}(\widehat{X}) \leq m - 1$ but not in $\mathbb{R}^{q-1}$.

Let D denote the partial EDM (obtained from NMR), and $\Omega \subset \mathbb{N} \times \mathbb{N}$ the set of indices for known entries. In light of the above characterization, the protein reconstruction problem is the feasibility problem with constraints:

$$C_1 = \{X \in \mathbb{R}^{m\times m} : X \geq 0, X_{ij} = D_{ij} \text{ for } (i,j) \in \Omega\},$$

$$C_2 = \{X \in \mathbb{R}^{m\times m} : \widehat{X} \text{ in } (*) \text{ is PSD with } \text{rank}\,\widehat{X} \leq 3\}.$$

A Feasibility Problem Formulation

Recall the constraint sets:

$$C_1 = \{X \in \mathbb{R}^{m\times m} : X \geq 0, X_{ij} = D_{ij} \text{ for } (i,j) \in \Omega\},$$

$$C_2 = \{X \in \mathbb{R}^{m\times m} : \widehat{X} \text{ in } (*) \text{ is PSD with } \operatorname{rank} \widehat{X} \leq 3\}.$$

A Feasibility Problem Formulation

Recall the constraint sets:

$$C_1 = \{X \in \mathbb{R}^{m\times m} : X \geq 0, X_{ij} = D_{ij} \text{ for } (i,j) \in \Omega\},$$
$$C_2 = \{X \in \mathbb{R}^{m\times m} : \widehat{X} \text{ in } (*) \text{ is PSD with } \operatorname{rank} \widehat{X} \leq 3\}.$$

Now,

- C_1 is a convex set (intersection of cone and affine subspace).
- C_2 is convex iff $m \leq 2$ (in which case $C_2 = \mathbb{R}^{m\times m}$).

For interesting problems, C_2 is **never convex**.

Computing Projections and Reflections

Recall the constraint sets:

$$C_1 = \{X \in \mathbb{R}^{m\times m} : X \geq 0, X_{ij} = D_{ij} \text{ for } (i,j) \in \Omega\},$$
$$C_2 = \{X \in \mathbb{R}^{m\times m} : \widehat{X} \text{ in } (*) \text{ is PSD with } \operatorname{rank} \widehat{X} \leq 3\}.$$

Computing Projections and Reflections

Recall the constraint sets:

$$C_1 = \{X \in \mathbb{R}^{m\times m} : X \geq 0, X_{ij} = D_{ij} \text{ for } (i,j) \in \Omega\},$$
$$C_2 = \{X \in \mathbb{R}^{m\times m} : \widehat{X} \text{ in } (*) \text{ is PSD with } \operatorname{rank} \widehat{X} \leq 3\}.$$

The projection onto C_1 is given (point-wise) by

$$P_{C_1}(X)_{ij} = \begin{cases} D_{ij} & \text{if } (i,j) \in \Omega, \\ \max\{0, X_{ij}\} & \text{otherwise.} \end{cases}$$

Computing Projections and Reflections

Recall the constraint sets:

$$C_1 = \{X \in \mathbb{R}^{m\times m} : X \geq 0, X_{ij} = D_{ij} \text{ for } (i,j) \in \Omega\},$$
$$C_2 = \{X \in \mathbb{R}^{m\times m} : \widehat{X} \text{ in } (*) \text{ is PSD with } \operatorname{rank} \widehat{X} \leq 3\}.$$

The projection onto C_1 is given (point-wise) by

$$P_{C_1}(X)_{ij} = \begin{cases} D_{ij} & \text{if } (i,j) \in \Omega, \\ \max\{0, X_{ij}\} & \text{otherwise.} \end{cases}$$

The projection onto C_2 is the set

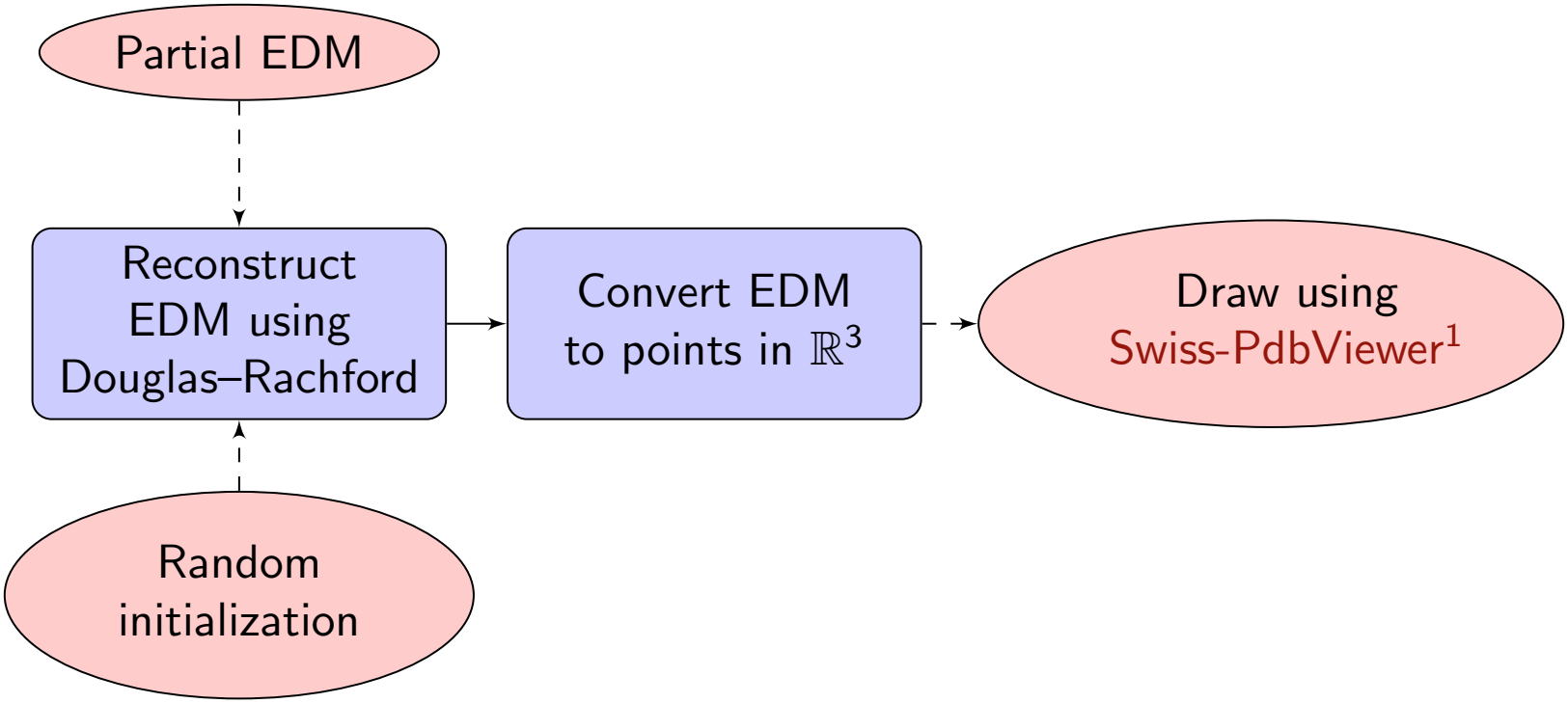
$$P_{C_2}(X) = \left\{ -Q \begin{bmatrix} \widehat{Y} & d \\ d^T & \delta \end{bmatrix} Q : Q(-X)Q = \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix}, \begin{array}{l} \widehat{X} \in \mathbb{R}^{(m-1)\times(m-1)}, \\ d \in \mathbb{R}^{m-1}, \delta \in \mathbb{R}, \end{array} \widehat{Y}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

where S is the set of PSD matrices of rank 3 or less.

- Computing $P_S(\widehat{X})$ = spectral decomposition $\rightarrow$ threshold eigenvalues.

Numerical and Visual Experiments

The reconstruction approach can be summarised as follows:



[1] http://spdbv.vital-it.ch/

Experiment: Six Test Proteins

Experiment: We consider the simplest realistic protein conformation determination problem.

NMR experiments were simulated for proteins with known conformation by computing the partial EDM containing all inter-atomic distances < 6Å.

Table : Six proteins from the RCSB Protein Data Bank.[2]

Protein	# Atoms	# Residues	Known Distances
1PTQ	404	50	8.83%
1HOE	581	74	6.35%
1LFB	641	99	5.57%
1PHT	988	85	4.57%
1POA	1067	118	3.61%
1AX8	1074	146	3.54%

[2] `http://www.rcsb.org/`

Experiment: Six Test Proteins

Table : Average (worst) results: **5,000** iterations, five random initializations.

Protein	Problem Size	Rel. Error (dB)	RMS Error	Max Error
1PTQ	81,406	-83.6 (-83.7)	0.02 (0.02)	0.08 (0.09)
1HOE	168,490	-72.7 (-69.3)	0.19 (0.26)	2.88 (5.49)
1LFB	205,120	-47.6 (-45.3)	3.24 (3.53)	21.68 (24.00)
1PHT	236,328	-60.5 (-58.1)	1.03 (1.18)	12.71 (13.89)
1POA	568,711	-49.3 (-48.1)	34.09 (34.32)	81.88 (87.60)
1AX8	576,201	-46.7 (-43.5)	9.69 (10.36)	58.55 (62.65)

- The reconstructed EDM is compared to the actual EDM using:

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

- The reconstructed points in $\mathbb{R}^3$ are then compared using:

$$\text{RMS Error} = \left(\sum_{k=1}^{m}\|z_k - z_k^{\text{actual}}\|^2\right)^{1/2}, \quad \text{Max Error} = \max_{k=1,\ldots,m}\|z_k - z_k^{\text{actual}}\|,$$

which are computed up to translation, reflection and rotation.

Experiment: Six Test Proteins

Table : Average (worst) results: 5,000 iterations, five random initializations.

Protein	Problem Size	Rel. Error (dB)	RMS Error	Max Error
1PTQ	81,406	-83.6 (-83.7)	0.02 (0.02)	0.08 (0.09)
1HOE	168,490	-72.7 (-69.3)	0.19 (0.26)	2.88 (5.49)
1LFB	205,120	-47.6 (-45.3)	3.24 (3.53)	21.68 (24.00)
1PHT	236,328	-60.5 (-58.1)	1.03 (1.18)	12.71 (13.89)
1POA	568,711	-49.3 (-48.1)	34.09 (34.32)	81.88 (87.60)
1AX8	576,201	-46.7 (-43.5)	9.69 (10.36)	58.55 (62.65)

- The reconstructed EDM is compared to the actual EDM using:

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

- The reconstructed points in $\mathbb{R}^3$ are then compared using:

$$\text{RMS Error} = \left(\sum_{k=1}^{m}\|z_k - z_k^{\text{actual}}\|^2\right)^{1/2}, \quad \text{Max Error} = \max_{k=1,\ldots,m}\|z_k - z_k^{\text{actual}}\|,$$

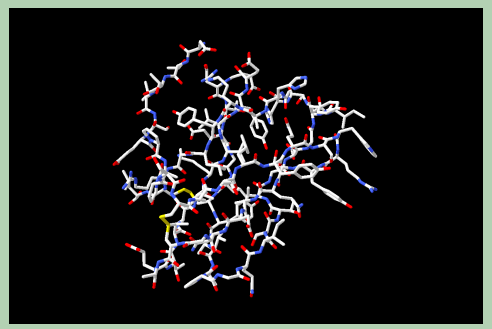
which are computed up to translation, reflection and rotation.

Experiment: Six Test Proteins

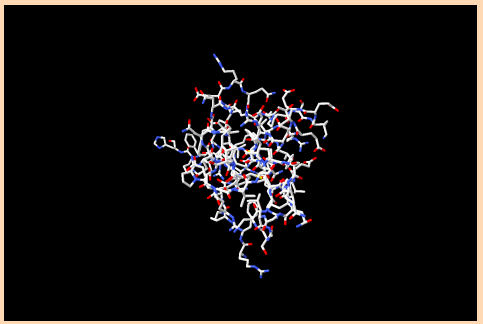
How do these error metrics compare visually to our expectations?

Experiment: Six Test Proteins

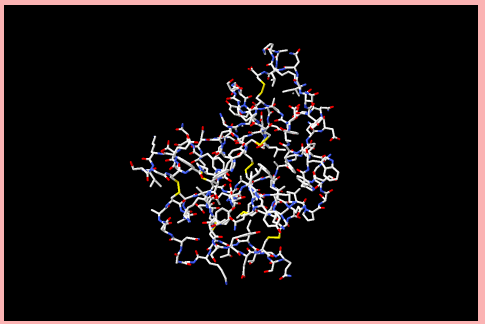
How do these error metrics compare visually to our expectations?



1HOE (actual)



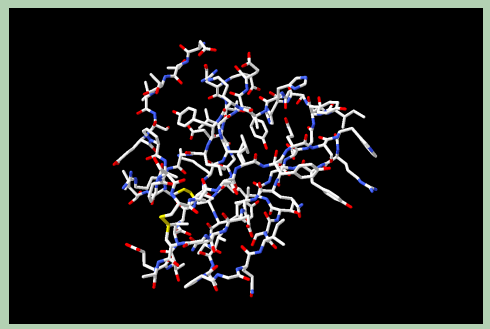
1LFB (actual)



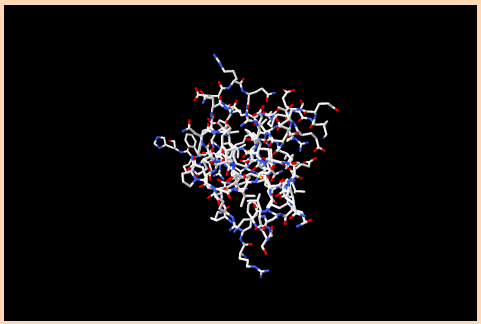
1POA (actual)

Experiment: Six Test Proteins

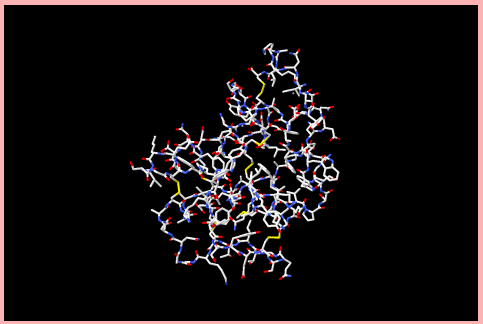
How do these error metrics compare visually to our expectations?



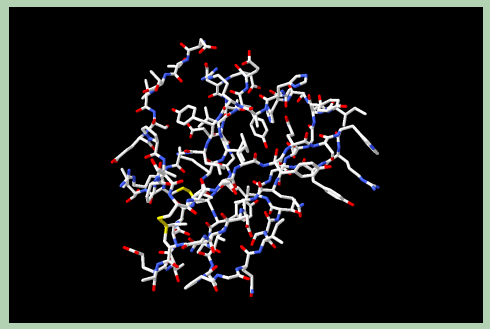
1HOE (actual)



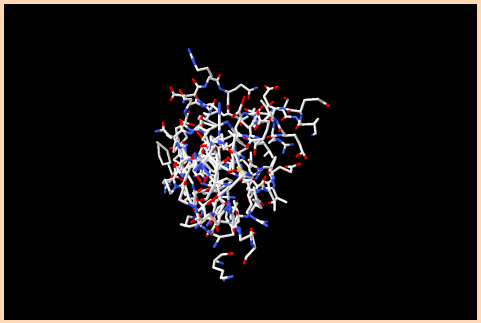
1LFB (actual)



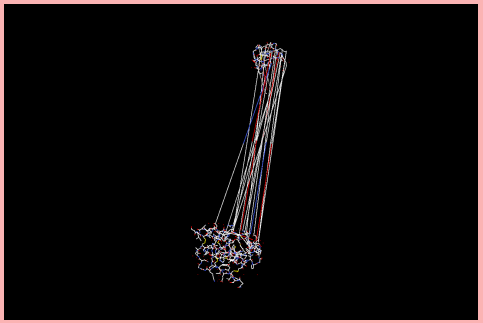
1POA (actual)



1HOE (-72.7dB)



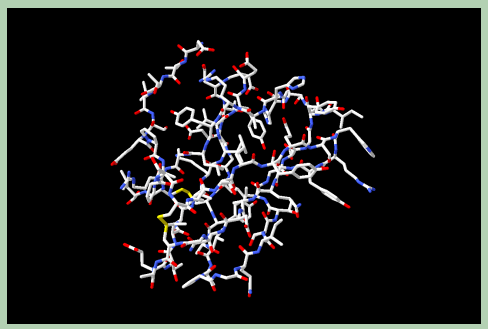
1LFB (-60.5dB)



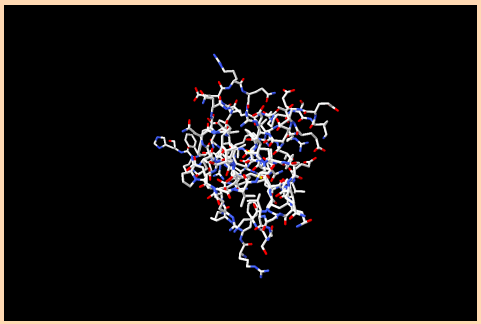
1POA (-49.3dB)

Experiment: Six Test Proteins

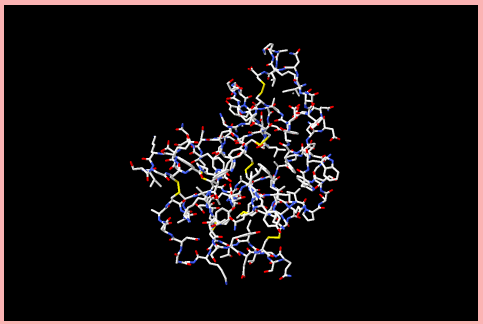
How do these error metrics compare visually to our expectations?



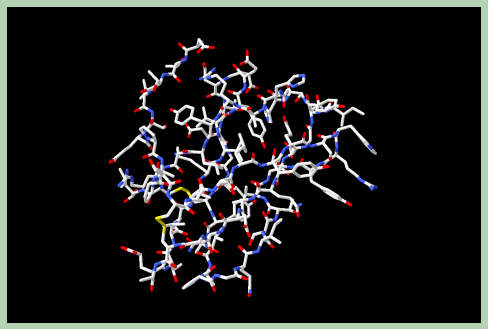
1HOE (actual)



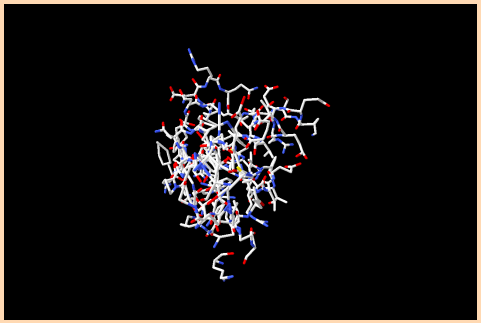
1LFB (actual)



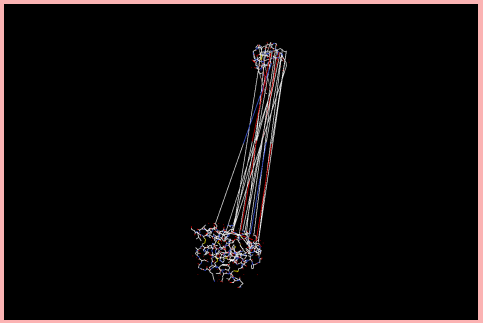
1POA (actual)



1HOE (-72.7dB)



1LFB (-60.5dB)



1POA (-49.3dB)

1HOE is good, 1LFB is mostly good, and 1POA has two good pieces.

Experiment: A Better Stopping Criterion?

On revisiting the problem, an optimised implementation gave a ten-fold speed-up.

Experiment: A Better Stopping Criterion?

On revisiting the problem, an optimised implementation gave a ten-fold speed-up. This allowed for the following experiment to be performed:

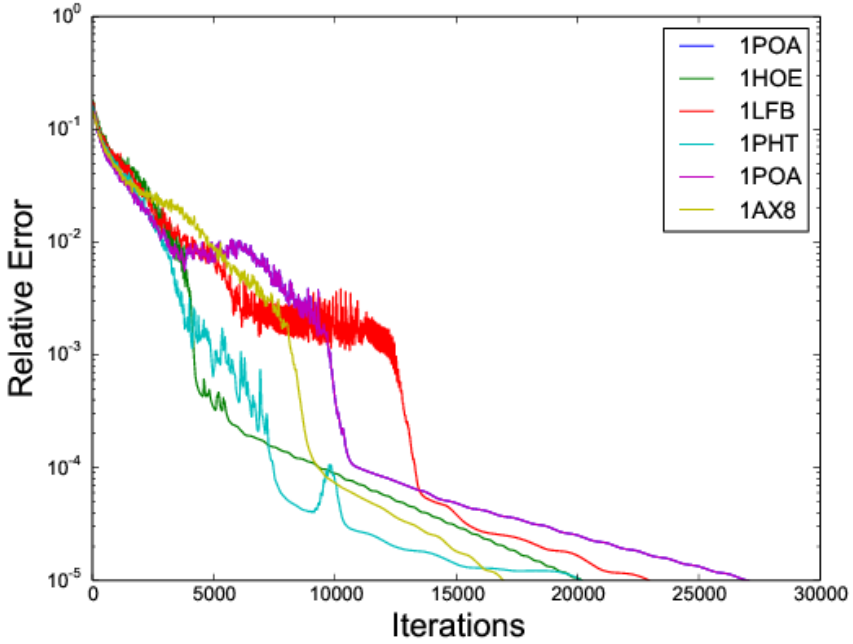


Figure: Relative error by iterations (vertical axis logarithmic).

Experiment: A Better Stopping Criterion?

On revisiting the problem, an optimised implementation gave a ten-fold speed-up. This allowed for the following experiment to be performed:

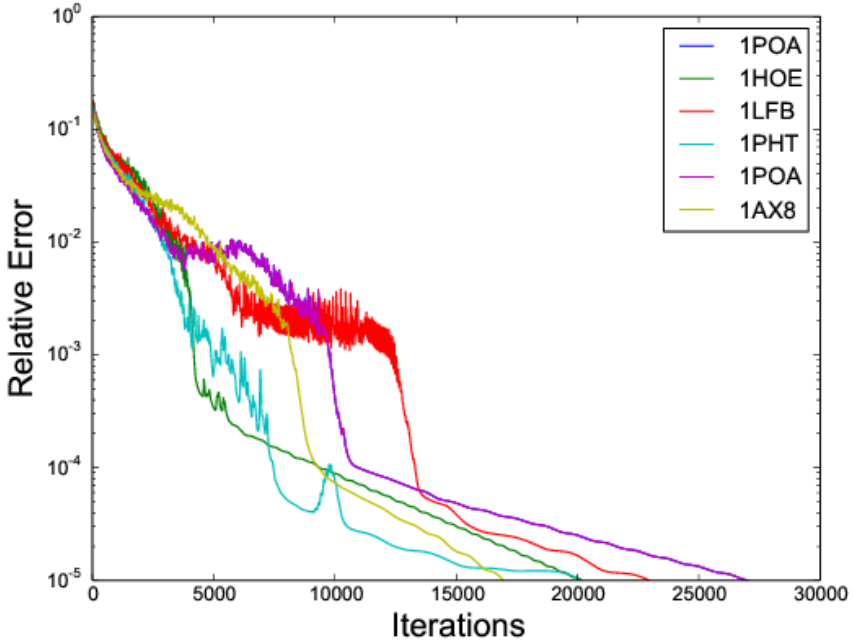
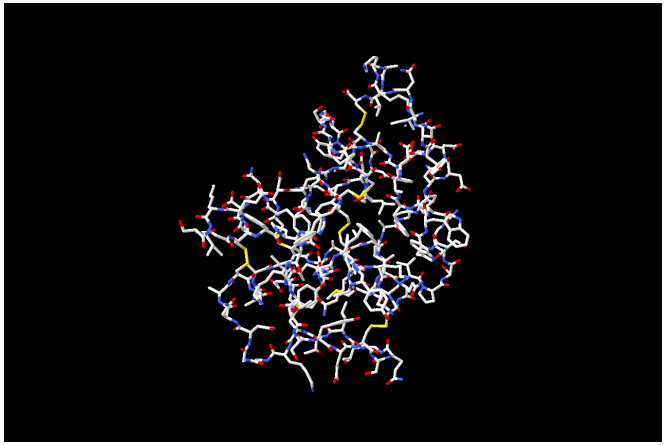


Figure: Relative error by iterations (vertical axis logarithmic).

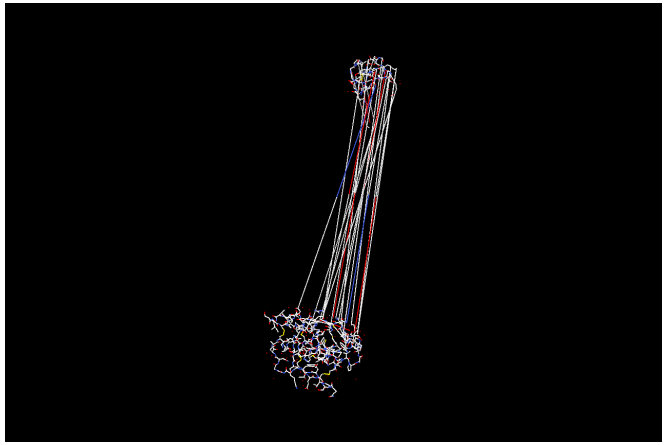
- For $< 5,000$ iterations, the error exhibits non-monotone oscillatory behaviour. It then decreases sharply. Beyond this progress is slower.
- Is early termination to blame? Terminate when error < -100dB.

A More Robust Stopping Criterion

The "un-tuned" implementation (worst reconstruction from previous slide):



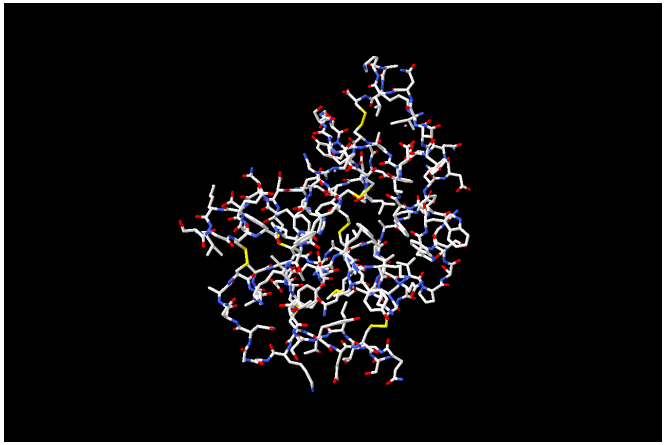
1POA (actual)



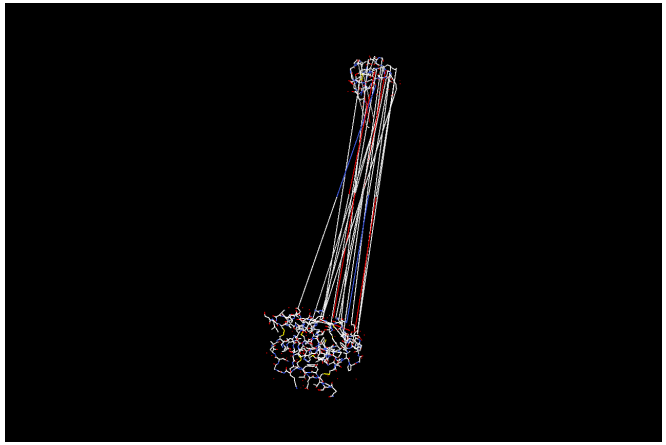
5,000 steps ($\sim$2d), -49.3dB

A More Robust Stopping Criterion

The "un-tuned" implementation (worst reconstruction from previous slide):

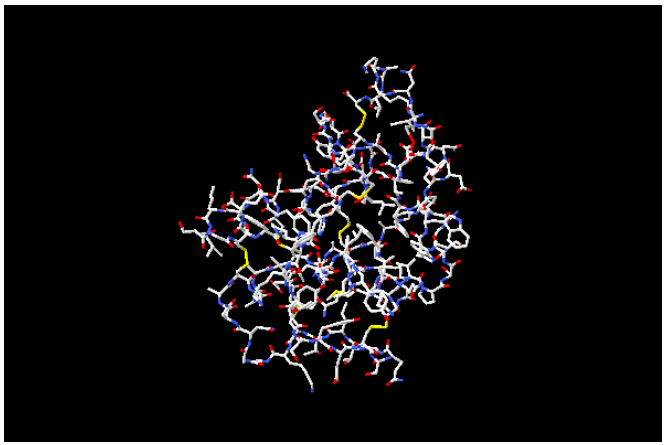


1POA (actual)

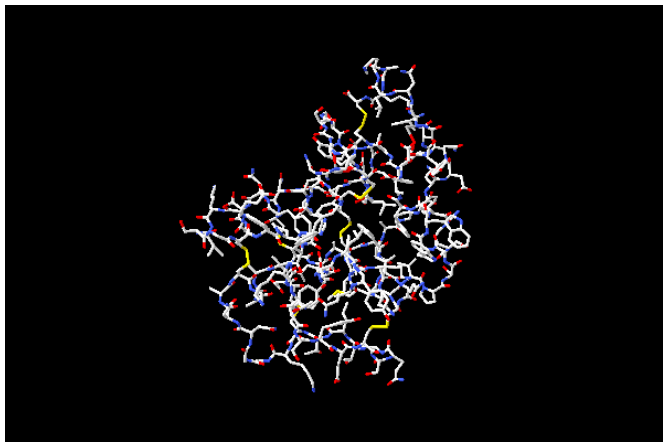


5,000 steps ($\sim$2d), -49.3dB

The optimised implementation:



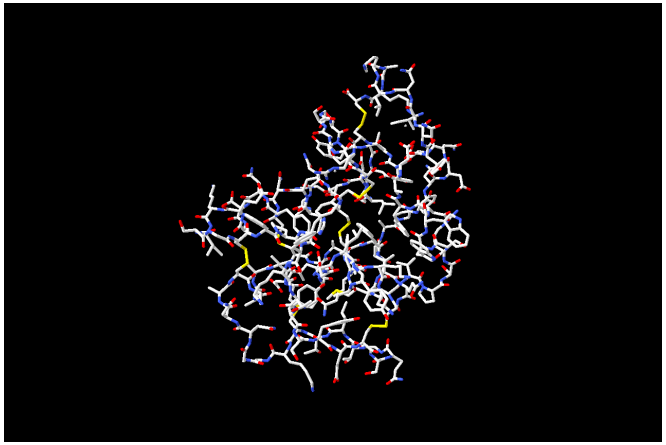
1POA (actual)



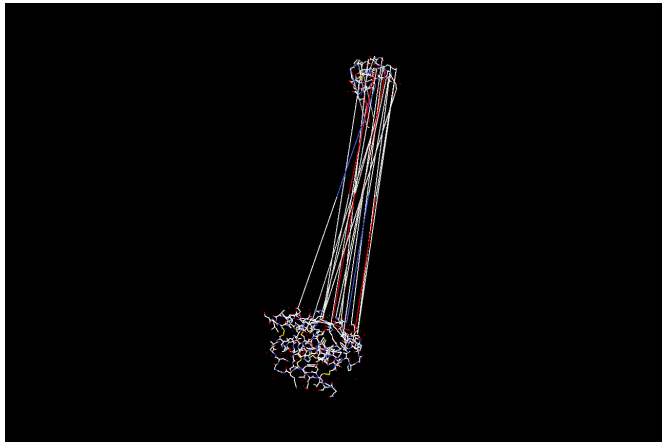
28,500 steps ($\sim$1d), -100dB (perfect!)

A More Robust Stopping Criterion

The "un-tuned" implementation (worst reconstruction from previous slide):

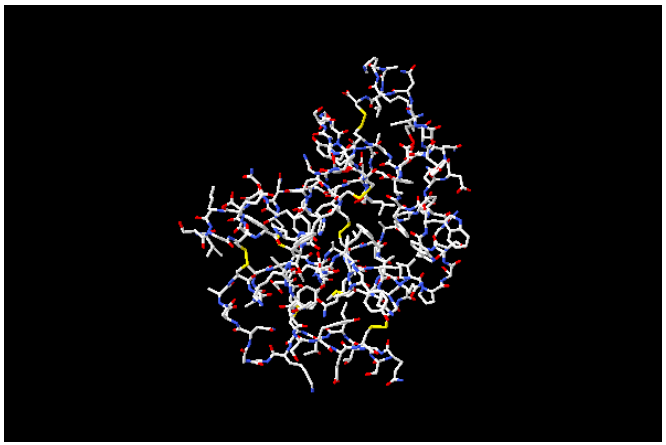


1POA (actual)

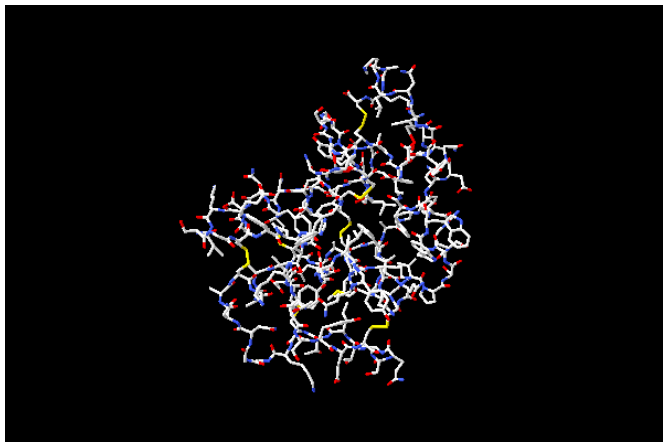


5,000 steps ($\sim$2d), -49.3dB

The optimised implementation:



1POA (actual)



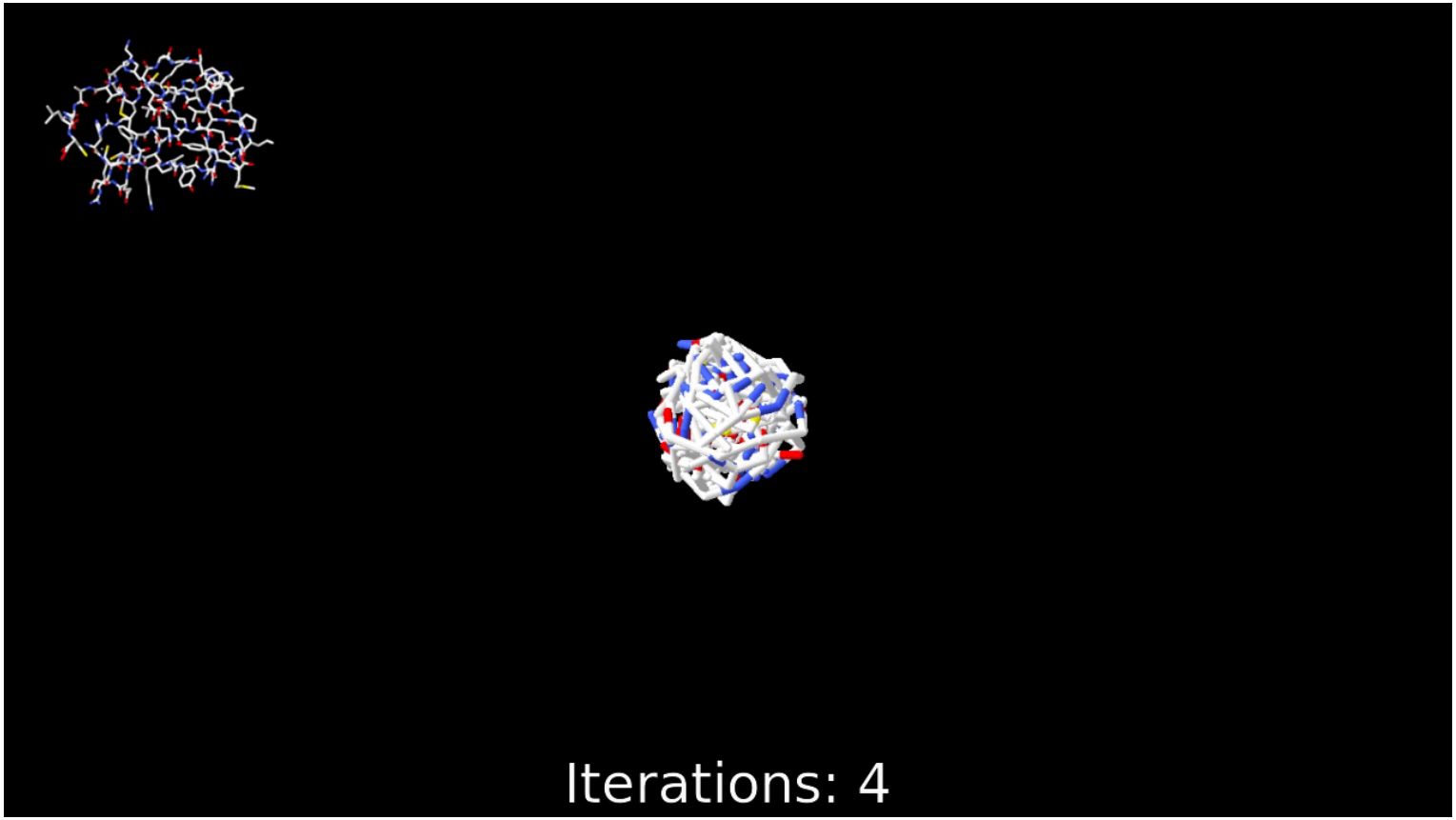
28,500 steps ($\sim$1d), -100dB (perfect!)

- Similar results observed for the other test proteins.

Experiment: Why Use the Douglas–Rachford Method?

Experiment: There are many projection methods, so why should we use the Douglas–Rachford method?

Experiment: Why Use the Douglas–Rachford Method?



First 3,000 steps of the 1PTQ reconstruction

Experiment: Why Use the Douglas–Rachford Method?

Experiment: There are many projection methods, so why should we use the Douglas–Rachford method?

Experiment: Why Use the Douglas–Rachford Method?

Experiment: There are many projection methods, so why should we use the Douglas–Rachford method?

A simpler projection method is the method of alternating projections. Given a point $y_0 \in \mathcal{H}$ is given by the fixed-point iteration

$$y_{n+1} := P_{C_2} P_{C_1} y_n.$$

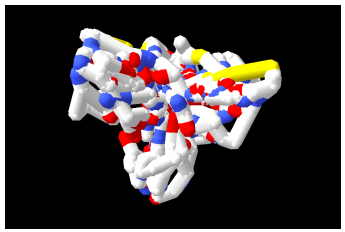
Experiment: Why Use the Douglas–Rachford Method?

Experiment: There are many projection methods, so why should we use the Douglas–Rachford method?

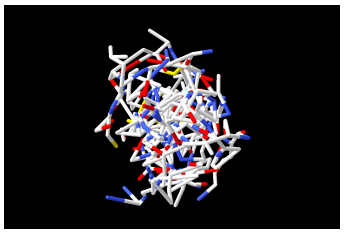
A simpler projection method is the method of alternating projections. Given a point $y_0 \in \mathcal{H}$ is given by the fixed-point iteration

$$y_{n+1} := P_{C_2} P_{C_1} y_n.$$

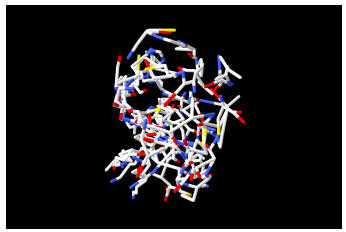
Before reconstruction



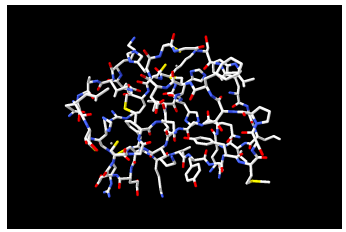
Douglas–Rachford method reconstruction:



500 steps, -25 dB

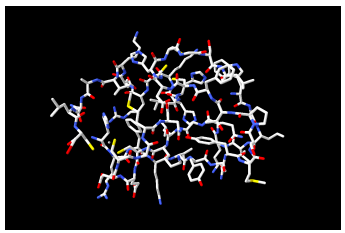


1,000 steps,-30 dB

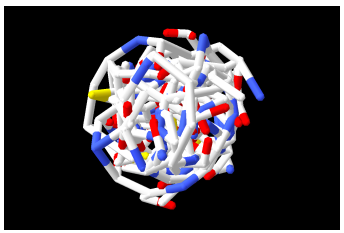


2,000 steps, -51 dB

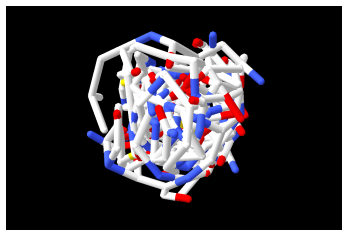
1PTQ (actual)



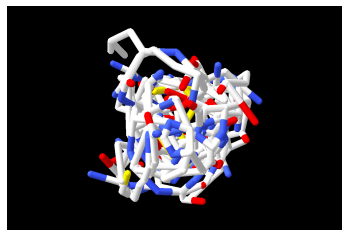
Method of alternating projections reconstruction:



500 steps,-22 dB



1,000 steps, -24 dB



2,000 steps, -25 dB

Experiment: Why Use the Douglas–Rachford Method?

Recall:

Theorem (Basic behaviour of the Douglas–Rachford method)

Suppose C_1, C_2 are closed convex subsets of a finite dimensional Hilbert space $\mathcal{H}$. For any $x_0 \in \mathcal{H}$, define $x_{n+1} = T_{C_1,C_2} x_n$.

1. If $C_1 \cap C_2 \neq \emptyset$, then $x_n \to x$ such that $P_{C_1} x \in C_1 \cap C_2$.
2. If $C_1 \cap C_2 = \emptyset$, then $\|x_n\| \to +\infty$.

Experiment: Why Use the Douglas–Rachford Method?

Recall:

Theorem (Basic behaviour of the Douglas–Rachford method)

Suppose C_1, C_2 are closed convex subsets of a finite dimensional Hilbert space $\mathcal{H}$. For any $x_0 \in \mathcal{H}$, define $x_{n+1} = T_{C_1,C_2} x_n$.

1. If $C_1 \cap C_2 \neq \emptyset$, then $x_n \to x$ such that $P_{C_1} x \in C_1 \cap C_2$.
2. If $C_1 \cap C_2 = \emptyset$, then $\|x_n\| \to +\infty$.

- The Douglas–Rachford method can be sensitive to perturbations in the constraint sets.

Experiment: Why Use the Douglas–Rachford Method?

Recall:

Theorem (Basic behaviour of the Douglas–Rachford method)

Suppose C_1, C_2 are closed convex subsets of a finite dimensional Hilbert space $\mathcal{H}$. For any $x_0 \in \mathcal{H}$, define $x_{n+1} = T_{C_1,C_2} x_n$.

1. If $C_1 \cap C_2 \neq \emptyset$, then $x_n \to x$ such that $P_{C_1} x \in C_1 \cap C_2$.
2. If $C_1 \cap C_2 = \emptyset$, then $\|x_n\| \to +\infty$.

- The Douglas–Rachford method can be sensitive to perturbations in the constraint sets.
- In contrast the alternating projections sequence might still converge even if the intersection section is empty.

Experiment: Why Use the Douglas–Rachford Method?

Recall:

Theorem (Basic behaviour of the Douglas–Rachford method)

Suppose C_1, C_2 are closed convex subsets of a finite dimensional Hilbert space $\mathcal{H}$. For any $x_0 \in \mathcal{H}$, define $x_{n+1} = T_{C_1,C_2} x_n$.

1. If $C_1 \cap C_2 \neq \emptyset$, then $x_n \to x$ such that $P_{C_1} x \in C_1 \cap C_2$.
2. If $C_1 \cap C_2 = \emptyset$, then $\|x_n\| \to +\infty$.

- The Douglas–Rachford method can be sensitive to perturbations in the constraint sets.
- In contrast the alternating projections sequence might still converge even if the intersection section is empty.
- **Conjecture:** The Douglas–Rachford method's instability stops it from getting 'stuck' in local minima.

Ongoing Work: Towards Theoretical Justification

A recent result of Phan arXiv:1401.6509v2, shows local convergence of the Douglas–Rachford method assuming the constraints satisfy regularity properties (*i.e.*, super-regularity, strongly regular intersection). With some work, specialising to our application, we deduce:

- A sufficient condition for local convergence can be "checked" by showing that there is no non-zero solution to a certain linear system.
- This linear system is defined by parameters which depend on the experimentally collected data.

Ongoing Work: Towards Theoretical Justification

A recent result of Phan arXiv:1401.6509v2, shows local convergence of the Douglas–Rachford method assuming the constraints satisfy regularity properties (*i.e.*, super-regularity, strongly regular intersection). With some work, specialising to our application, we deduce:

- A sufficient condition for local convergence can be "checked" by showing that there is no non-zero solution to a certain linear system.
- This linear system is defined by parameters which depend on the experimentally collected data.

Probable Local Convergence (T)

Suppose $\overline{X} \in C_1 \cap C_2 \subseteq \mathbb{R}^{m\times m}$ is embeddable in $\mathbb{R}^s$ but not $\mathbb{R}^{s-1}$ where

$$\overline{X} \equiv -Q \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix} Q,$$

and the directions of eigenvectors of $\widehat{X}$ are distributed uniformly at random (on the sphere). Further, suppose $s = m - 2$. For initial point sufficiently close to $\overline{X}$, the Douglas–Rachford method almost surely converges to a solution.

Ongoing Work: Towards Theoretical Justification

A recent result of Phan arXiv:1401.6509v2, shows local convergence of the Douglas–Rachford method assuming the constraints satisfy regularity properties (*i.e.*, super-regularity, strongly regular intersection). With some work, specialising to our application, we deduce:

- A sufficient condition for local convergence can be "checked" by showing that there is no non-zero solution to a certain linear system.
- This linear system is defined by parameters which depend on the experimentally collected data.

Probable Local Convergence (T)

Suppose $\overline{X} \in C_1 \cap C_2 \subseteq \mathbb{R}^{m\times m}$ is embeddable in $\mathbb{R}^s$ but not $\mathbb{R}^{s-1}$ where

$$\overline{X} \equiv -Q \begin{bmatrix} \widehat{X} & d \\ d^T & \delta \end{bmatrix} Q,$$

and the directions of eigenvectors of $\widehat{X}$ are distributed uniformly at random (on the sphere). Further, suppose $s = m - 2$. For initial point sufficiently close to $\overline{X}$, the Douglas–Rachford method almost surely converges to a solution.

While this **does not** yet explain our experimental observations, it nevertheless provides a useful template and starting point.

Concluding Remarks and Future Work

- The Douglas–Rachford method can predict protein conformation using only short-range distances. It performs better than theory suggests.
- Ongoing work is focusing on conditions for local convergence.
- The Douglas–Rachford method is a general purpose algorithm. There is potential for problem specific improvements which exploit special structure in the constraint sets.
- Other fruitful applications? We have also investigated an analogous bulk structure determination problem arising in ionic liquid chemistry.

Douglas–Rachford feasibility methods for matrix completion problems with F.J. Aragón Artacho & J.M. Borwein. *ANZIAM J.*, 55(4):299–325, 2014. arXiv:1312.7323

Reflection methods for inverse problems with application to protein conformation determination with J.M. Borwein. Springer volume on the *CIMPA school Generalized Nash Equilibrium Problems, Bilevel programming and MPEC* New Delhi, India, Dec. 2012. In press, Aug. 2014. arXiv:1312.7323

On the regularity of nonnegative sparsity sets. In preparation (near complete).

Many resources can be found at the companion website:

http://carma.newcastle.edu.au/DRmethods/

Concluding Remarks and Future Work

- The Douglas–Rachford method can predict protein conformation using only short-range distances. It performs better than theory suggests.
- Ongoing work is focusing on conditions for local convergence.
- The Douglas–Rachford method is a general purpose algorithm. There is potential for problem specific improvements which exploit special structure in the constraint sets.
- Other fruitful applications? We have also investigated an analogous bulk structure determination problem arising in ionic liquid chemistry.

When presented with a feasibility problem, it is well worth investigating if the Douglas–Rachford method can deal with it – the method is conceptually simple and easy to implement.

Douglas–Rachford feasibility methods for matrix completion problems with F.J. Aragón Artacho & J.M. Borwein. *ANZIAM J.*, 55(4):299–325, 2014. arXiv:1312.7323

Reflection methods for inverse problems with application to protein conformation determination with J.M. Borwein. Springer volume on the *CIMPA school Generalized Nash Equilibrium Problems, Bilevel programming and MPEC* New Delhi, India, Dec. 2012. In press, Aug. 2014. arXiv:1312.7323

On the regularity of nonnegative sparsity sets. In preparation (near complete).

Many resources can be found at the companion website:

http://carma.newcastle.edu.au/DRmethods/