Semidefinite Facial Reduction and Rigid Cluster Interpolation in Protein Structure Elastic Network Models

Xiao-Bo Li*, Forbes J. Burkowski, and Henry Wolkowicz

Abstract—Elastic network models have been used to interpolate two conformations of the same protein to create a transition pathway. This interpolation is called “elastic network interpolation (ENI)”. This procedure can be modified to accommodate *rigid clusters*, groups of atoms that move concurrently in a protein. The modified procedure is then referred to as “rigid cluster ENI”. Rank 3 positive semidefinite (PSD) matrix manifolds have *faces* that are defined by these rigid clusters. This facial structure strongly suggest these matrix manifolds are a natural choice for modelling rigid cluster transitions. However, such structure is hidden in the original classical formulation which does not use the PSD matrix manifold.

Index Terms—coarse grain, elastic network model, Euclidean distance matrix, facial reduction, Gram matrix, positive semidefinite matrix manifold, protein structure, Riemannian manifold.

1 Introduction

Elastic network models (ENMs) were introduced by Tirion [21] and Bahar et al. [4], where these authors demonstrated they are an efficient tool for studying low frequency protein dynamics. Elastic network interpolation (ENI) was used by Kim et al. to interpolate a transition pathway between two protein conformations [12]. It was shown in [9], [11], [13] how to interpolate between rigid clusters, groups of atoms that move concurrently. This method is referred to as *rigid cluster ENI*.

In this paper, we show rigid cluster ENI can be formulated as a facially reduced semidefinite optimization problem. In Section 2 we review Kim et al.’s formulation of rigid cluster ENI. Classically, ENI has used a potential energy that is a function of distance. However, the Euclidean distance matrix (EDM) which contains distance-squared values, is bijectively related to positive semidefinite (PSD) matrices due to a linear mapping discussed in, for example, [6], [7], [14]. Rigid clusters allow the PSD matrix representing the protein’s ENM to be facially reduced [14], [15]; such geometry is hidden in the classical potential energy defined using distance. This observation suggests distance-squared may be the more natural choice when defining the potential energy. We review the necessary mathematical background to facial reduction in Section 3. Finally, in Section 4, we describe the potential energy for rigid cluster interpolation on the rank 3 PSD matrix manifold; this potential energy explicitly shows the face, which is hidden in the classical formulation by Kim et al..

2 Classical Rigid Clustser Elastic Network Models

We now review the rigid cluster ENI method proposed by Kim et al. [9], [11], [13]. Our discussion assumes we have a protein structure represented by n α-carbon atoms distributed among m *disjoint* rigid clusters $\mathcal{C}_1, \ldots, \mathcal{C}_m$ such that

$$\mathcal{C}_1 \bigcup \cdots \bigcup \mathcal{C}_m = 1:n \tag{1}$$

Consider an arbitrary atom, with coordinates at time t denoted by $p_a(t) \in \mathbb{R}^3$, which belongs to some arbitrary rigid cluster $\mathcal{C}_i$. Let the center of this rigid cluster at time t be denoted by $c_i(t) \in \mathbb{R}^3$. Let $v_i(t) = c_i(t) - c_i(t-1)$ denote the translation of the cluster center, and hence translation of the entire rigid cluster, between time t and $t-1$. Also, let $R(\omega_i(t))$, a 3×3 rotation matrix, denote the relative rotation of cluster i at time t from time $t-1$, with $\omega_i(t) \in \mathbb{R}^3$ a vector parallel to the axis of rotation. Then, $p_a(t)$, can be expressed by:

$$p_a(t) = R(\omega_i(t))\,(p_a(t-1) - c_i(t-1)) + c_i(t-1) + v_i(t) \tag{2}$$

Assuming the rotation is very small, we can approximate the rotation matrix as:

$$R(\omega_i(t)) \approx I_3 + mat(\omega_i(t)) \ , \tag{3}$$

where I_3 is the 3×3 identity matrix. $mat(\cdot)$ turns a vector $v = (x, y, z)^T \in \mathbb{R}^3$ to a skew-symmetric matrix:

- *Asterisk indicates corresponding author.*
- *XB Li is with the Cheriton School of Computer Science, University of Waterloo, ON Canada, N2L 3G1. Email: x22li@uwaterloo.ca.*
- *FJ Burkowski is with the Cheriton School of Computer Science, University of Waterloo, ON, Canada, N2L 3G1. E-mail: fjburkowski@uwaterloo.ca. Website: http://www.structuralbioinformatics.com/ and https://cs.uwaterloo.ca/about/people/fjburkowski.*
- *H. Wolkowicz is with the Department of Combinatorics and Optimization, University of Waterloo, ON, Canada, N2L 3G1. E-mail: hwolkowicz@uwaterloo.ca and website http://www.math.uwaterloo.ca/ hwolkowi/*

Manuscript received Month xx, 2016; revised Month xx, 2016.

$$mat(v) = \begin{pmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{pmatrix} \tag{4}$$

This approximation gives us a new expression for $p_a(t)$ for small rotations :

$$\begin{aligned} p_a(t) &\approx p_a(t-1) - c_i(t-1) \\ &\quad + mat(\omega_i(t))(p_a(t-1) - c_i(t-1)) \\ &\quad + c_i(t-1) + v_i(t) \\ &\quad = p_a(t-1) + mat(\omega_i(t))(p_a(t-1) - c_i(t-1)) + v_i(t) \ . \end{aligned} \tag{5}$$

Note that for vectors $v, w \in \mathbb{R}^3$,

$$mat(w)v = -mat(v)w \ , \tag{6}$$

Therefore, we can expression equation(5) in matrix form as:

$$\begin{aligned} p_a(t) &= p_a(t-1) - mat(p_a(t-1) - c_i(t-1))\omega_i(t) + v_i(t) \\ &= p_a(t-1) + H_{ai}(t-1)\delta_i(t) \ , \end{aligned} \tag{7}$$

where $H_{ai}(t-1)$ is a 3×6 matrix:

$$H_{ai}(t-1) = \begin{pmatrix} -mat(p_a(t-1) - c_i(t-1)) & I_3 \end{pmatrix} \ , \tag{8}$$

and $\delta_i(t)$ is a 6×1 vector:

$$\delta_i(t) = \begin{pmatrix} \omega_i(t) \\ v_i(t) \end{pmatrix} \tag{9}$$

Kim et al. proposes to model the movement of rigid clusters from a starting protein conformation to an ending conformation using the objective function:

$$U_t(\delta(t)) = \sum_{(a,b)\in\mathcal{D}} \frac{1}{2} \left(\parallel p_a(t) - p_b(t) \parallel - l_{ab}(t)\right)^2 \tag{10}$$

$p_a(t)$ and $p_b(t)$ are functions of $\delta_i(t)$ as given by equation (7). $\mathcal{D}$ is a set of pairs of indices that represent pairs of α-carbons in different rigid clusters that interact, and $\delta(t) = (\delta_1(t)^T, \ldots, \delta_m(t)^T)^T \in \mathbb{R}^{6m}$. $l_{ab}(t)$ for $0 < t < 1$ is the *linearly interpolated* targeted distance at time t between α-carbons indexed by a and b:

$$l_{ab}(t) = (1-t) \parallel p_a(0) - p_b(0) \parallel + t \parallel p_a(1) - p_b(1) \parallel \ , \tag{11}$$

In order to find the optimal $\delta_i(t)$ to advance to the next time step, we take the second order Taylor series expansion of the potential energy given by equation (10). We now give a quick review of the needed formulas.

2.1 Second Order Expansion for Distances

Let $d \in \mathbb{R}$ be a scalar and $x \in \mathbb{R}^n$ be a vector, and the function to be expanded be:

$$f(\delta) = \frac{1}{2} \left(\parallel x + \delta \parallel - d\right)^2 \ . \tag{12}$$

The second order expansion is:

$$f(\delta) \approx f(0) + \mathrm{grad} f(0)^T \delta + \frac{1}{2} \delta^T \mathrm{Hess} f(0) \delta \ . \tag{13}$$

The constant term $f(0)$ is given by:

$$f(0) = \frac{1}{2} \left(\parallel x \parallel - d\right)^2 \ . \tag{14}$$

$\mathrm{grad} f(0)$ is an $n \times 1$ vector given by:

$$\mathrm{grad} f(0) = (\parallel x \parallel - d) \frac{x}{\parallel x \parallel} \ . \tag{15}$$

$\mathrm{Hess} f(0)$ is an $n \times n$ matrix given by, using the product rule on $\mathrm{grad} f(0)$:

$$\begin{aligned} \mathrm{Hess} f(0) &= (\parallel x \parallel - d) \left(\frac{\parallel x \parallel I_n - \frac{xx^T}{\|x\|^2}}{\parallel x \parallel^2} \right) + \frac{xx^T}{\parallel x \parallel^2} \\ &= I_n - \frac{d}{\parallel x \parallel} \left(I_n - \frac{xx^T}{\parallel x \parallel^2} \right) \ . \end{aligned} \tag{16}$$

2.2 Second Order Expansion for Rigid Clusters Using Distance

The objective function, equation (10), has the following second order expansions. Let $p_{ab}(t) = p_a(t) - p_b(t)$, then:

$$\begin{aligned} U_t(\delta) \approx & \frac{1}{2} \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} (\delta_i(t)^T \delta_j(t)^T) A_{ij} \begin{pmatrix} \delta_i(t) \\ \delta_j(t) \end{pmatrix} \\ & + \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} B_{ij} \begin{pmatrix} \delta_i(t) \\ \delta_j(t) \end{pmatrix} \\ & + \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} C_{ij} \ . \end{aligned} \tag{17}$$

where A_{ij} is a 12×12 matrix given by:

$$A_{ij} = \sum_{\substack{a\in\mathcal{C}_i \\ b\in\mathcal{C}_j \\ (a,b)\in\mathcal{D}}} \begin{bmatrix} H_{ia}^T \\ -H_{jb}^T \end{bmatrix} \mathcal{A}_{ab} \left[H_{ia} - H_{jb} \right] \tag{18}$$

$$\mathcal{A}_{ab} = I_3 - l_{ab}(t) \frac{g(p_{ab}(t-1))}{\parallel p_{ab}(t-1) \parallel} \tag{19}$$

and $g(v)$ is a function on $v \in \mathbb{R}^3$ is given by:

$$g(v) = I_3 - \frac{vv^T}{\parallel v \parallel^2} \tag{20}$$

and B_{ij} is a 1×12 matrix given by:

$$Bij = \sum_{\substack{a\in\mathcal{C}_i \\ b\in\mathcal{C}_j \\ (a,b)\in\mathcal{D}}} \mathcal{B}_{ab} \left[H_{ia}(t-1) - H_{jb}(t-1) \right] \tag{21}$$

where:

$$\mathcal{B}_{ab} = \left(1 - \frac{l_{ab}(t)}{\parallel p_{ab}(t-1) \parallel} \right) p_{ab}(t-1)^T \tag{22}$$

$$Cij = \sum_{\substack{a\in\mathcal{C}_i \\ b\in\mathcal{C}_j \\ (a,b)\in\mathcal{D}}} \frac{1}{2} \left(\parallel p_a(t-1) - p_b(t-1) \parallel - l_{ab}(t)\right)^2 \tag{23}$$

Note that in the above formulas,

$$\left[H_{ia}(t-1) - H_{jb}(t-1) \right] \ , \tag{24}$$

is a block matrix, not a matrix subtraction.

Now define the blocks of A_{ij} as:

$$A_{ij} = \begin{pmatrix} P_{ij} & Q_{ij} \\ Q_{ji} & S_{ij} \end{pmatrix} \ , \tag{25}$$

and also define the blocks of B_{ij} as:

$$B_{ij} = \begin{pmatrix} u_{ij} & v_{ij} \end{pmatrix} \ . \tag{26}$$

Then we define a large $6m \times 6m$ matrix M, where m is the number of rigid clusters, whose i, j-th block is given by:

$$M_{ij} = \begin{cases} \sum_{a=1}^{i-1} S_{ai} + \sum_{b=i+1}^{m} P_{ib} & \text{if } i = j \\ Q_{ij} & \text{if } i \neq j \end{cases} \tag{27}$$

and a large $1 \times 6m$ matrix N, whose i-th block is given by:

$$N_i = \sum_{a=1}^{i-1} v_{ai} + \sum_{b=i+1}^{m} u_{ib} \tag{28}$$

These matrices allow us to express equation (17) as:

$$\frac{1}{2}\delta(t)^T M \delta(t) + N\delta(t) + O \ , \tag{29}$$

where $O = \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} C_{ij}$ is a constant. The derivative of equation (29) is set to zero to find the optimal $\delta(t)$.

3 Facial Reduction

3.1 The PSD Matrix Manifold

For an arbitrary time t, define the $n \times 3$ matrix P_t which contains all n α-carbon coordinates of a protein:

$$P_t = \begin{pmatrix} p_1^T(t) \\ \vdots \\ p_n^T(t) \end{pmatrix} \tag{30}$$

The *Gram matrix*, a rank 3 PSD matrix is given by P_t multiplied by its transpose:

$$X_t = P_t {P_t}^T \tag{31}$$

The set of points $p_1(t), \ldots, p_n(t)$ also has a Euclidean distance matrix (EDM). This is the matrix of distance-squared between points. The Gram matrix X_t is related to the EDM D_t via the following linear operator

$$D_t = \mathcal{K}(X_t) = \text{diag}(X_t)e^T + e\text{diag}(X_t)^T - 2X_t \tag{32}$$

diag$(\cdot)$ is a linear operator extracting the diagonal of the matrix. *Centered* Gram matrices have points all centered at the origin, this condition can be expressed as $Xe = 0$, where e is an $n \times 1$ vector of ones. The mapping $\mathcal{K}(\cdot)$ is a bijective linear mapping between EDMs and centered Gram matrices. Krislock provides a more indepth discussion of this mapping [14]. This bijection means the choice to use distance-squared is equivalent to the choice to use Gram matrices to model protein dynamics.

Certain mathematical properties of the Gram matrix suggests they are a more natural choice for modelling ENMs.

For example, X_t is invariant to rotation. For a rotation matrix Q, $P_t Q Q^T {P_t}^T = P_t {P_t}^T$.

Classical dynamics defines a potential energy as a function on a Riemannian manifold [3], [5]. The set of rank 3 PSD matrices is a Riemannian manifold [10], [18], [19], [22], it has many geometries, see [22]. However, the rotational invariance of the Gram matrix means the quotient geometry seen in [10], [19], and Section 6.62 of [22] is the geometry relevant to ENMs. Under this geometry, P_t is usually specified, rather than the entire X_t.

In classical ENI, the potential energy is a function of distance. Such potential energy thus does not involve the mapping $\mathcal{K}(\cdot)$ and is not a function of the PSD Gram matrix. It is a function of a vector of all the atomic coordinates, which does not have the rotational invariance property of Gram matrices; the set of such vectors is called the *linear manifold* [1]. The rank 3 PSD matrix manifold is an alternative choice to the linear manifold.

Secondly, the set of $n \times n$ EDMs, ignoring the rank constraint of the points, is a convex cone because of the bijective mapping with the convex cone of PSD matrices; see Section 2.5 page 25 of Krislock [14] . ENI requires the targeted distance to be interpolated, but the set of $n \times n$ distance matrices is not convex for $n > 3$, see a discussion in [8].

Finally, the strongest evidence Gram matrices, and EDMs are the natural choice for modelling ENMs is the property that rigid clusters within a protein structure describe *faces* of the PSD cone.

3.2 A Face of the PSD Cone

From Proposition 2.15 of Krislock [14], we have that a face of the $n \times n$ PSD cone is a convex cone given by:

$$F = U\mathbf{S}_+^k U^T \ , \tag{33}$$

where U has orthonormal columns and $\mathbf{S}_+^k$ is the set of $k \times k$ PSD matrices, where k is known after U is known.

Facial reduction is the process of finding the face matrix U.

3.3 A Motivation for Facial Reduction

We now provide an intuitive description for what facial reduction does.

Consider a rigid cluster $\mathcal{C}$ containing points $p_1, \ldots, p_k \in \mathbb{R}^3$ and place these points in an $k \times 3$ matrix P_0

$$P_0 = \begin{pmatrix} p_1^T \\ \vdots \\ p_k^T \end{pmatrix} \tag{34}$$

If we wish to model the rotation and translation of these points to arrive at a new configuration P_t, we would form the following expression:

$$P_t = P_0 Q_t + e{v_t}^T \ , \tag{35}$$

Where Q_t is a 3×3 rotation matrix, and v_t is a 3×1 translation vector, so ${v_t}^T$ is 1×3; e is an $n \times 1$ vector of all ones. P_0 can be assumed to be centered at the origin, with P_t moving relative to P_0. Expressing equation (35) using block matrices gives:

$$P_t = \begin{pmatrix} P_0 & e \end{pmatrix} \begin{pmatrix} Q_t \\ {v_t}^T \end{pmatrix} \ . \tag{36}$$

Let $V_0 = \begin{pmatrix} P_0 & e \end{pmatrix}$ and $S_t = \begin{pmatrix} Q_t \\ {v_t}^T \end{pmatrix}$. We can then write the Gram matrix concisely as:

$$X = P_t {P_t}^T = V_0 S_t {S_t}^T {V_0}^T = V_0 R_t {V_0}^T \tag{37}$$

Equation (37) illustrates the facially reduced smaller Gram matrix $R_t = S_t {S_t}^T$ contains information about what needs to change to arrive at the new configuration. That which does not need to change has been factored out into the face V_0. In other words, we only need to determine the smaller rank 3 PSD matrix R_t.

3.4 Theorems for Facial Reduction

We now formally show how to construct the face of a single rigid cluster, and the face of disjoint rigid clusters. The construction for a single rigid cluster makes use of Krislock Theorem 4.1 (Single Clique Facial Reduction Theorem) [14]; the construction for disjoint rigid clusters make use of Krislock Theorem 4.5 (Disjoint Subsets Facial Reduction Theorem) [14].

3.5 Single Rigid Cluster Face Construction

Consider a protein with n α-carbon coordinates represented by $p_1, \dots, p_n \in \mathbb{R}^3$ embedded in r dimensions. Let the indices of the one rigid cluster be denoted by the set $\mathcal{C} \subseteq 1 : n$. Form the $n \times 3$ matrix P:

$$P = \begin{pmatrix} p_1^T \\ \vdots \\ p_n^T \end{pmatrix} \tag{38}$$

Let $P[\mathcal{C}, :]$ denote the rows of P indexed by $\mathcal{C}$. Ensure $P[\mathcal{C}, :]$ is centered at the origin. Then the centered Gram matrix of the rigid cluster is given by $X_{\mathcal{C}} = P[\mathcal{C}, :]P[\mathcal{C}, :]^T$. Perform singular value decomposition (SVD), or eigendecomposition of $X_{\mathcal{C}}$, to find a $|\mathcal{C}| \times r$ orthogonal matrix $U_{\mathcal{C}}$ such that $\text{range}(X_{\mathcal{C}}) = \text{range}(U_{\mathcal{C}})$. Then, the face for these $\mathcal{C}$ points is given by the $|\mathcal{C}| \times (r+1)$ matrix:

$$\bar{V} = \begin{pmatrix} U_{\mathcal{C}} & \frac{e}{\sqrt{k}} \end{pmatrix} \tag{39}$$

where e is a vector of $|\mathcal{C}|$ $1'$s and division by $\sqrt{k}$ ensures $\bar{V}$ is orthogonal.

The rest of the points, which are not indexed by $\mathcal{C}$, do not belong to any clique in this current construction. They receive the *trivial* face, an 1×1 matrix with one entry equal to 1. In light of the motivation from Section 3.3, this is equivalent to these points having only a translational variation. This is consistent with *hybrid ENMs* modelling point masses (single atoms) as rigid clusters with a trivial (no) rotation, see [9], [11], [13].

Therefore, the face of the entire protein with n α-carbon atoms, and one rigid cluster indexed by $\mathcal{C}$ is the matrix given by:

$$V = \begin{array}{r} \\ |\mathcal{C}| \\ n - |\mathcal{C}| \end{array} \begin{array}{c} \begin{array}{cc} r+1 & n-|\mathcal{C}| \end{array} \\ \begin{pmatrix} \bar{V} & 0 \\ 0 & I \end{pmatrix} \end{array} \tag{40}$$

Without loss of generality, we have assumed indices in $\mathcal{C}$ are the first $|\mathcal{C}|$ rows of V; if this is not so, the rows may be permutated to those positions.

This completes the construction of the face for an n α-carbon protein with one rigid cluster. The matrix P can now be expressed as the factorization:

$$P = VS \tag{41}$$

Any problem requiring P to be determined can now be reduced to determining S. If we known P, the corresponding S can be found by:

$$V^T P = V^T V S = I_n S = S \tag{42}$$

3.6 Disjoint Clique Face Reduction

We now show how to construct the face matrix for a protein with more than one, disjoint, rigid cluster. If the rigid clusters are not disjoint, the intersection must be non-rigid, because rigid intersections will not allow the two rigid clusters to move relative to each other, which is the modelling goal. Non-rigid intersections in three-dimensions will have at most two atoms in common. The intersection can be treated as disjoint by absorbing the two atoms simultaneously into one of the rigid clusters, chosen arbitrarily. See Krislock Section 4.7-4.9 [14] for a discussion of rigid and non-rigid intersections.

Let $\mathcal{C}_1, \dots, \mathcal{C}_m \subseteq 1 : n$ be the index set of disjoint rigid cluster α-carbon atoms, with embedding dimensions $r_1, \dots, r_m$. Assume without loss of generality that these index sets are consecutive, so their union has indices $1 : |\mathcal{C}|$.

$$\mathcal{C} = \bigcup_{i=1}^{m} \mathcal{C}_i = 1 : |\mathcal{C}| \tag{43}$$

Let there be $n - |\mathcal{C}|$ remaining atoms not associated with any rigid cluster. Finally, let $\bar{V}_i$ denote the face of rigid cluster $\mathcal{C}_i$, constructed as described in Section 3.5 given by equation(39). The face for all n α-carbon atoms is given by:

$$U = \begin{array}{r} \\ |\mathcal{C}_1| \\ \vdots \\ |\mathcal{C}_m| \\ n - |\mathcal{C}| \end{array} \begin{array}{c} \begin{array}{cccc} r_1 + 1 & \dots & r_m + 1 & n - |\mathcal{C}| \end{array} \\ \left(\begin{array}{cccc} \bar{V}_1 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & \bar{V}_m & 0 \\ 0 & \dots & 0 & I \end{array} \right) \end{array} \tag{44}$$

4 Facially Reduced Positive Semidefinite Elastic Network Models

4.1 Facial Reduction and Rigid Cluster ENI

We now show that rigid cluster ENMs are in fact using facially reduced Gram matrices, and thus rigid cluster ENI can be formulated as a facially reduced rank 3 PSD matrix manifold optimization problem.

Consider the transpose of equation (5):

$$\begin{aligned} p_a(t)^T &= p_a(t-1)^T \\ &+ (p_a(t-1) - c_i(t-1))^T mat(\omega_i(t))^T + v_i(t)^T \ . \end{aligned} \tag{45}$$

For all atoms that belong to one rigid cluster, they are rotated by the same matrix $mat(\omega_i(t))^T$ and translated by the same vector $v_i(t)^T$. Let rigid cluster $\mathcal{C}_i$ have $m(i)$ atoms. We can thus place all $m(i)$ equations for each rigid cluster into a matrix as follows:

$$\begin{pmatrix} p_1(t)^T \\ \vdots \\ p_{m(i)}(t)^T \end{pmatrix} = \begin{pmatrix} p_1(t-1)^T \\ \vdots \\ p_{m(i)}(t-1)^T \end{pmatrix} + \begin{pmatrix} (p_1(t-1) - c_i(t-1))^T & 1 \\ \vdots & \\ (p_{m(i)}(t-1) - c_i(t-1))^T & 1 \end{pmatrix} \begin{pmatrix} mat(\omega_i(t))^T \\ v_i(t)^T \end{pmatrix} . \tag{46}$$

Consider the matrix:

$$V_i = \begin{pmatrix} (p_1(t-1) - c_i(t-1))^T & 1 \\ \vdots & \\ (p_{m(i)}(t-1) - c_i(t-1))^T & 1 \end{pmatrix} \tag{47}$$

V_p is actually the *point representation* of the face of this rigid cluster. See Section 4.12 of Krislock [14] for a discussion of point representations of faces. This point representation was also used in equation (36). Define:

$$\Delta_i = \begin{pmatrix} mat(\omega_i(t))^T \\ v_i(t)^T \end{pmatrix} . \tag{48}$$

Then equation (46) can be written more concisely as:

$$P_i(t) = P_i(t-1) + V_i \Delta_i \tag{49}$$

The matrix Δ_i is what needs to change to arrive at the new time t $P_i(t)$, the matrix V_i does not change. If we can further factorize $P_i(t-1) = V_i S_{t-1}$, then we see equation (49) is:

$$P_i(t) = V_i S_{t-1} + V_i \Delta_i = V_i(S_{t-1} + \Delta_i) \tag{50}$$

The facially reduced Gram matrix for $P_i(t)$ is thus:

$$R_t = (S_{t-1} + \Delta_i)(S_{t-1} + \Delta_i)^T \tag{51}$$

Thus, rigid cluster ENI is perturbing the *facially reduced coordinates*.

4.2 The Facially Reduced Positive Semidefinite Potential Energy

We now introduce the facially reduced potential energy for ENI on the rank 3 PSD matrix manifold. This potential energy has already been used in semidefinite optimization problems as objective functions, see [2], [18], [19]

Let P_t be the matrix containing all n α-carbons of a protein structure, as defined in equation (30). Suppose there are m disjoint rigid clusters, and their face is denoted V, constructed as described in Section 3.6. Then, using facial reduction:

$$P_t = V S_t , \tag{52}$$

Following the same argument as in Section 4.1, we can show to arrive at P_t from P_{t-1}, we only need to perturb the facially reduced Gram matrix as follows. Since:

$$P_t = V(S_{t-1} + \Delta) \tag{53}$$

The perturbed Gram matrix is thus:

$$P_t {P_t}^T = V(S_{t-1} + \Delta)(S_{t-1} + \Delta)^T V^T \tag{54}$$

We now define the potential energy in terms of equation (54).

Firstly, note that squaring the distance in the classical ENI potential energy given by equation (10) gives the following potential energy:

$$\begin{aligned} f(P_t) &= \sum_{(a,b) \in \mathcal{D}} \frac{1}{2}((p_a(t) - p_b(t))^T((p_a(t) - p_b(t)) - d_{ab}(t))^2 \\ &= \sum_{(a,b) \in \mathcal{D}} \frac{1}{2}((e_a - e_b)^T P_t {P_t}^T (e_a - e_b) - d_{ab}(t))^2 . \end{aligned} \tag{55}$$

Equation (55) is summed over pairs in $\mathcal{D}$, which are all the interactions between different rigid clusters, the same set as used in equation (10). $e_a \in \mathbb{R}^n$ is an $n \times 1$ vector with a 1 at the a-th position. $d_{ab}(t)$ is given by:

$$d_{ab}(t) = (1-t) \parallel p_a(0) - p_b(0) \parallel^2 + t \parallel p_a(1) - p_b(1) \parallel^2 , \tag{56}$$

Whereas in equation (11) we interpolated distance, we now interpolate between distance-squared. Equation (56) can be expressed explicitly as an interpolated EDM in the following equivalent expression using the $\mathcal{K}(\cdot)$ map given by equation (32):

$$f(P_t) = \frac{1}{2} \parallel H \circ (\mathcal{K}(P_t {P_t}^T) - D(t)) \parallel_F^2 . \tag{57}$$

$\parallel \cdot \parallel_F$ is the Frobenius norm. H is a matrix whose entries are defined as:

$$H_{ab} = \begin{cases} 1 & \text{if } (a,b) \in \mathcal{D} \\ 0 & \text{otherwise} \end{cases} . \tag{58}$$

$\circ$ is element-wise multiplication. $D(t)$ has entries $d_{ab}(t)$ given by equation (56); this matrix can be written explicitly as a convex combination of elements from a convex set:

$$D(t) = (1-t)D(0) + tD(1) , \tag{59}$$

where the entries of $D(0)$ are $d_{ab}(0) = \parallel p_a(0) - p_b(0) \parallel^2$ and the entries of $D(1)$ are $d_{ab}(1) = \parallel p_a(1) - p_b(1) \parallel^2$.

The interpolated EDM $D(t)$ may not represent points in 3 dimensions, this is because the rank constraint is not convex. However, the potential energy is finding the best set of points in 3 dimensions because it is only searching over the set of rank 3 Gram matrices.

The objective function introduced by equation (55) is the same form as that presented in Section 6.3.1 of [18]. The equivalent matrix expression given by equation (57) was introduced in [2]. Both functions are objective functions for solving the EDM completion problem. These objective functions show ENI can be formulated as a semidefinite optimization problem. To accommodate rigid clusters in the protein, we substitute equation (52) into these equation. For example, with equation (57), we have:

$$\begin{aligned} f_V(S_t) &= \frac{1}{2} \parallel H \circ (\mathcal{K}(V S_t {S_t}^T V^T) - D(t)) \parallel_F^2 \\ &= \frac{1}{2} \parallel H \circ (\mathcal{K}_V(S_t S_t) - D(t)) \parallel_F^2 . \end{aligned} \tag{60}$$

where we have defined the operator $\mathcal{K}_V(\cdot)$ as:

$$\mathcal{K}_V(X) = \mathcal{K}(V X V^T) , \tag{61}$$

This operator was first used in [2] to show how facial reduction can be incorporated into EDM completion problems.

Similarly, we can substitute equation (52) into equation (55)

$$f_V(S_t) = \sum_{(a,b)\in\mathcal{D}} \frac{1}{2}((e_a-e_b)^T V S_t {S_t}^T V^T (e_a-e_b) - d_{ab}(t))^2 \; . \tag{62}$$

Equations (53) and (54) then show we are searching for the optimal perturbation to go from S_{t-1} to the current time t S_t.

4.3 Finding the Optimal Perturbation

Equation (53) shows we need to perturbed the reduced coordinates to get the new Gram matrix. In Kim's formulation of ENI, a second order approximation of the potential energy is minimized to find the optimal pertubration [9], [11], [12], [13]. We follow a similar approach here. However, since we are using the quotient geometry for the rank 3 PSD matrix manifold, we use the tCG algorithm adopted to this geometry [1]; the gradient and Hessian used here have been discussed in [16], [18], [19]

The second order Taylor series expansion for a stepsize η to arrive at S_t from S_{t-1} given by:

$$\begin{aligned} &f_V(S_t) = f_V(S_{t-1} + \eta) \\ &\approx f_V(S_{t-1}) + \langle \eta, \mathrm{grad} f_V(S_{t-1})\rangle + \frac{1}{2}\langle \eta, \mathrm{Hess} f_V(S_{t-1})[\eta]\rangle \; , \end{aligned} \tag{63}$$

where $\langle \xi, \eta\rangle = tr(\xi^T \eta)$ is the Riemannian metric defined on the tangent space at S_t. The η in equation (63) is a matrix in the tangent space centered at S_{t-1}. Since the expansion is for time t, the targeted distance in all the above $f_V(\cdot)$ functions is $d_{ab}(t)$, and not $d_{ab}(t-1)$.

As shown by Meyer [18], the gradient, grad$f_V(S_{t-1})$, and the Hessian, Hess$f_V(S_{t-1})[\eta]$, in equation (63) can be expressed as the product of sparse matrices. These are convenient for implementation. We modify Meyer's formulas to accommodate the face V. For gradient, this is:

$$\mathrm{grad} f_V(S_{t-1}) = 2(V^T E \Sigma E^T V) S_t \; . \tag{64}$$

E is an $n \times |\mathcal{D}|$ sparse matrix with columns given by $(e_a - e_b)$ for all $(a,b) \in \mathcal{D}$. $e_a, e_b \in \mathbb{R}^n$ have a 1 in the a-th and b-th position respectively, and zero elsewhere. Σ is the $|\mathcal{D}| \times |\mathcal{D}|$ diagonal matrix whose diagonal entries are the errors:

$$\mathcal{D}_{ii} = (d_{ab}(t-1) - d_{ab}(t)) \quad i = 1, \ldots, |\mathcal{D}| \; , \tag{65}$$

For the Hessian, we have:

$$\mathrm{Hess} f_V(S_{t-1})[\eta] = 2P^H_{S_{t-1}}(V^T E \Sigma E^T V \eta + V^T E \widetilde{\Sigma} E^T V S_{t-1}) \; , \tag{66}$$

where $\widetilde{\Sigma}$ is a diagonal matrix, whose diagonal are the dot products given by:

$$\mathrm{diag}(\widetilde{\Sigma}) = 2\mathrm{diag}((E^T V S_{t-1})(\eta^T V^T E)) \tag{67}$$

$P^H_{S_{t-1}}$ is a projection operator that maps tangent vectors in the total space onto the horizontal space [1], [18], [19]. It is given by:

$$P^H_{S_{t-1}}(\eta) = \eta - \Omega S_{t-1} \; . \tag{68}$$

η is a matrix on the tangent space, and Ω solves the Sylvester equation:

$$\Omega {S_{t-1}}^T S_{t-1} + {S_{t-1}}^T S_{t-1} \Omega = {S_{t-1}}^T \eta - \eta^T S_{t-1} \tag{69}$$

4.4 Sample Interpolation: Lactoferrin

In this section, we use lactoferrin to demonstrate the facially reduced interpolation process. Lactoferrin has 3 disjoint rigid clusters. The "head": Gly321 $\sim$ Lys691, the left lobe: His91 $\sim$ Val 250, and the right lobe: Gly1 $\sim$ Thr90, Pro251 $\sim$ Leu320. We implemented the tCG algorithm in the python environment provided by UCSF Chimera [20]. Although an interpolation of only the α-carbons is enough to generate the transition, we found that Chimera required more atoms to render the α-helices correctly; therefore, all atoms common to both structures, modulo amino acid mutation, were interpolated. Figure 1 presents the result of interpolation from 1LFG to 1LFH. A smooth transition is observed, and at time $t = 1$ the interpolated structure can be seen to align closely with the targeted structure in black. Figure 1 is consistent with the interpolation presented by Kim et al. in [11], [13].

4.5 Discussion

The range of motion between the two lactoferrin conformations is small, and in this case, a geodesic connecting the matrix S_t would also produce a realistic transition, this was done in our previous publication [17]. We discussed in [16] that using a geodesic to connect the beginning and ending S_0 and S_1 matrices and interpolating these matrices directly (range interpolation), both may lead to degeneracy problems; Kim [11] had also discussed this degeneracy problem. ENI, formulated using both distance and distance-squared, is able to overcome this degeneracy problem.

In [16], we observed some anomalies when interpolating lattice structures using the matlab code provided by Kim [1]. When we implemented Kim's formulation in python, those anomalies were not seen, or were not as severe. Thus, those anomalies may be due to the linear algebra libraries used, and not caused by using distance in the potential energy.

5 Conclusion

Rigid cluster ENI is an efficient method for generating transitional conformations for proteins with rigid clusters. Rigid clusters in a protein restrict the protein to a certain face of the PSD cone. Due to the bijective mapping between PSD Gram matrices and EDMs, this face is explicit in the potential energy when distance-squared is used to formulate ENI. The use of distance in the potential energy hides the facial structure of the rigid clusters. This observation suggests distance-squared may be a more natural choice than distance when modelling rigid cluster transitions.

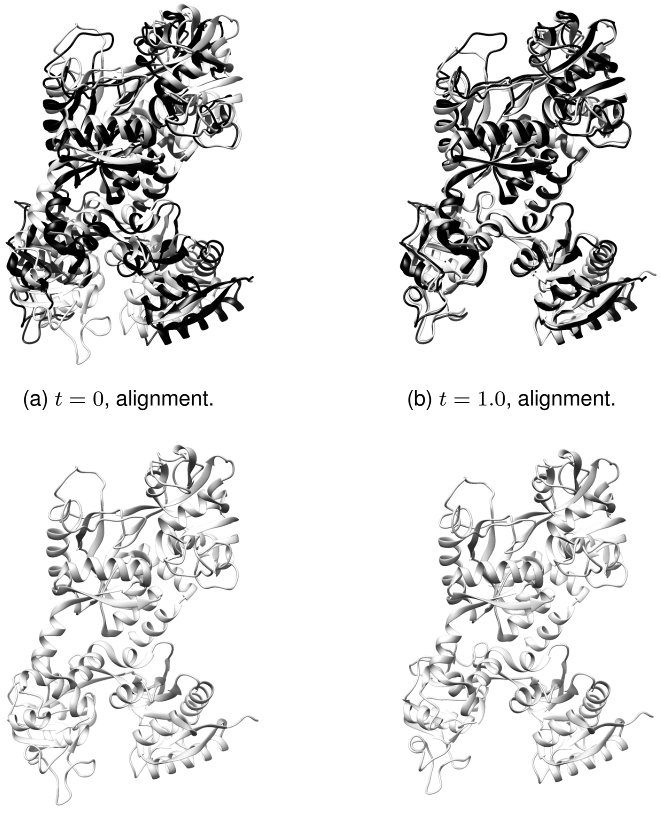
References

[1] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ: Princeton University Press, 2008.

[2] A. Alfakih, A. Khandani, and H. Wolkowicz, "Solving Euclidean distance matrix completion problems via semidefinite programming," *Computational Optimization and Applications*, vol. 12, no. 1-3, pp. 13–30, January 1999.

[3] V. Arnold, *Mathematical Methods of Classical Mechanics*. Springer-Verlag, 1978.

[1] Matlab code for ENI is available from the KOSMOS website http://bioengineering.skku.ac.kr/kosmos/tutorial.php. This code does not handle rigid clusters.



(a) $t = 0$, alignment.

(b) $t = 1.0$, alignment.

(c) $t = 0.25$

(d) $t = 0.75$

Fig. 1. 1LFG (light gray) is interpolated to 1LFH (black).

[4] I. Bahar, A. Atilgan, and B. Erman, "Direct evaluation of thermal fluctuations in poteins using a single-parameter harmonic potential," *Folding and Design*, vol. 2, no. 3, pp. 173–181, 1997.

[5] O. Calin and D.-C. Chang, *Geometric mechanics on Riemannian manifolds: applications to partial differential equations*. Springer Science & Business Media, 2006.

[6] F. Critchley, "On certain linear mappings between inner-product and squared-distance matrices," *Linear Algebra and its Applications*, vol. 105, pp. 91–107, 1988.

[7] J. Dattorro, "Equality relating Euclidean distance cone to positive semidefinite cone," *Linear Algebra and its Applications*, vol. 428, no. 11, pp. 2597–2600, 2008.

[8] ——, *Convex optimization and Euclidean distance geometry*. MeBoo, 2014.

[9] Y. Jang, "Hybrid elastic network model for macromolecular dynamics," Ph.D. dissertation, University of Massachusetts Amherst, 2008.

[10] M. Journée, F. Bach, P.-A. Absil, and R. Sepulchre, "Low-rank optimization on the cone of positive semidefinite matrices," *SIAM J. OPTIM*, vol. 20, no. 5, pp. 2327–2351, May 2010.

[11] M. K. Kim, "Elastic network models of biomolecular structure and dynamics," Ph.D. dissertation, The Johns Hopkins University, 2004.

[12] M. K. Kim, R. L. Jernigan, and G. S. Chirikjian, "Efficient generation of feasible pathways for protein conformational transitions," *Biophysical Journal*, vol. 83, no. 3, pp. 1620–1630, 2002.

[13] ——, "Rigid-cluster models of conformational transitions in macromolecular machines and assemblies," *Biophysical journal*, vol. 89, no. 1, pp. 43–55, 2005.

[14] N. Krislock, "Semidefinite facial reduction for low-rank Euclidean distance matrix completion," Ph.D. dissertation, School of Computer Science, University of Waterloo, 2010.

[15] N. Krislock and H.Wolkowicz, "Explicit sensor network localization using semidefinite representations and facial reductions," *SIAM Journal on Optimization*, vol. 20, no. 5, pp. 2679–2708, 2010.

[16] X. Li and F. Burkowski, "Generating conformational transitions using the Euclidean distance matrix." *IEEE transactions on nanobioscience*, 2015.

[17] X.-B. Li and F. J. Burkowski, "Conformational transitions and principal geodesic analysis on the positive semidefinite matrix manifold," in *Bioinformatics Research and Applications*, ser. Lecture Notes in Computer Science, M. Basu, Y. Pan, and J. Wang, Eds. Springer International Publishing, 2014, vol. 8492, pp. 334–345.

[18] G. Meyer, "Geometric optimization algorithms for linear regression on fixed-rank matrices," Ph.D. dissertation, University of Liège, 2011.

[19] B. Mishra, G. Meyer, and R. Sepulchre, "Low-rank optimization for distance matrix completion," in *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, Orlando, FL, USA, December 12-15 2011, pp. 4455–4460.

[20] E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng, and T. E. Ferrin, "UCSF chimera a visualization system for exploratory research and analysis," *J Comp Chem*, vol. 25, no. 13, pp. 1605–1612, 2004.

[21] M. Tirion, "Large amplitude elastic motions in proteins from a single-parameter, atomic analysis," *Physical Review Letters*, vol. 77, no. 9, pp. 1905–1908, August 1996.

[22] B. Vandereycken, "Riemannian and multilevel optimization for rank-constrained matrix problems," Ph.D. dissertation, Department of Computer Science, KU Leuven, 2010.