

8-2-2010

Bilinear Programming and Protein Structure Alignment

J. Cain

D. Kamenetsky

N. Lavine

Advisors:

Allen Holder

Follow this and additional works at: http://scholar.rose-hulman.edu/math_mstr

 Part of the [Applied Mathematics Commons](#), [Bioinformatics Commons](#), and the [Molecular Biology Commons](#)

Recommended Citation

Cain, J.; Kamenetsky, D.; and Lavine, N., "Bilinear Programming and Protein Structure Alignment" (2010). *Mathematical Sciences Technical Reports (MSTR)*. 29.

http://scholar.rose-hulman.edu/math_mstr/29

MSTR 10-08

This Article is brought to you for free and open access by the Mathematics at Rose-Hulman Scholar. It has been accepted for inclusion in Mathematical Sciences Technical Reports (MSTR) by an authorized administrator of Rose-Hulman Scholar. For more information, please contact weir1@rose-hulman.edu.

Bilinear Programming and Protein Structure Alignment

J. Cain, D. Kamenetsky, N. Lavine

Adviser: Allen G. Holder

**Mathematical Sciences Technical Report Series
MSTR 10-08**

August 2, 2010

**Department of Mathematics
Rose-Hulman Institute of Technology
<http://www.rose-hulman.edu/math>**

Fax (812)-877-8333

Phone (812)-877-8193

Bilinear Programming and Protein Structure Alignment

J. Cain, D. Kamenetsky, N. Lavine
Faculty Sponsor: Allen Holder

August 2, 2010

Abstract

Proteins are a primary functional component of organic life, and understanding their function is integral to many areas of research in biochemistry. The three-dimensional structure of a protein largely determines this function. Protein structure alignment compares the structure of a protein with known function to that of a protein with unknown function. A protein's three-dimensional structure can be transformed through a smooth piecewise-linear sigmoid function to a real symmetric contact matrix that represents the functional significance of certain parts of the protein. We address the protein alignment problem as a minimization of the 2-norm difference of two proteins' contact matrices. The minimization is presented as a bilinear program, and spectral bounds for best- and worst-case alignments are provided, which are continuous with respect to small changes in the protein's structure. Further conditions for a perfect alignment and heuristics for finding quality solutions are given.

1 Introduction

Proteins are long chains of amino acids that form unique, tightly packed, globular structures called folds. The three-dimensional structure of a protein molecule, defined by its fold, largely determines its biological function [3]. Applying this property, protein alignment techniques seek to draw useful conclusions about the biological function of a protein given data about its fold. Specifically, the available data is the locations in three-dimensional space of components of the protein. With current technology, these components are generally amino acids, but in the future the location of every atom might be known. Our mathematical model is the same, and so, a generalized term for the components is used, namely residues.

The fold of a protein can be described by the relative distances between residues rather than by a list of the three-dimensional coordinates. In this way, proteins can be compared independently of the coordinate system. One common alignment technique compares the relative distances between the residues directly. A second common technique compares contact matrices, or contact maps, that describe the closeness of any two residues relative to a cut-off distance. In both cases, the goal is to compare the three-dimensional structure of a protein whose function is unknown to the three-dimensional structure of a protein whose biological function is known.

Methods of comparing protein structures that use distances directly have a disadvantage, which stems from the biology of the problem. At distances greater than a certain length, no chemical bonding occurs [3]. Contact matrices are used to compare proteins because they provide information about which residues are chemically bonded to others. While contact map overlap alignment problems use discrete contact maps [5], defining contact discretely limits flexibility. We follow the model presented in [7], which smooths the contact information and reformulates the problem in n -dimensional Euclidean space.

2 Notation

Consider an $n \times 3$ coordinate matrix whose i th row is the coordinates of the i th residue of the protein, and let M be the $n \times n$ distance matrix whose (i, j) th element is the distance between residue i and residue j . This gives a symmetric matrix whose diagonal elements are zero. When comparing two different proteins, we refer to their distance matrices as M' and M'' .

Let C be a smooth contact map, or matrix, based on a distance matrix M . Let B_1 and B_2 , with $0 \leq B_1 \leq B_2$, be the breakpoints of a piecewise-linear sigmoid function. Define C by

$$[C(B_1, B_2)]_{i,j} = \begin{cases} 1 & \text{if } M_{ij} \leq B_1 \\ 0 & \text{if } M_{ij} \geq B_2 \\ 1 - \frac{M_{ij} - B_1}{B_2 - B_1} & \text{if } B_1 < M_{ij} < B_2. \end{cases} \quad (1)$$

The entry in the (i, j) th position of C gives a measurement of the strength of the bond between the i th and j th residues. A value of one is the strongest bond, and a value of zero is the weakest. Since a distance of zero produces a contact value of one, the diagonal entries of C are one. The breakpoints, B_1 and B_2 , permit flexibility with regard to the exact cut-off distance. For any distance between B_1 and B_2 , the greater the distance, the smaller the value in the smooth contact map. For all distances greater than B_2 , the contact has a value of zero.

Let C' and C'' be the contact matrices for the two proteins being compared, and assume that C' and C'' are the same size, i.e., assume that two proteins contain the same number of residues.¹ The eigenvalues of C' and C'' are denoted λ'_i and λ''_i , respectively, where i indexes the eigenvalues in order of decreasing magnitude. Since C' and C'' are real symmetric matrices, C' and C'' are normal. It follows that each is unitarily similar to a diagonal matrix of its eigenvalues. Specifically,

$$C' = UD'U^T \text{ and } C'' = WD''W^T,$$

where D' and D'' are the diagonal matrices with λ'_i and λ''_i , respectively, in the i th row, and U and W are unitary matrices whose i th column is the eigenvector associated with λ'_i and λ''_i .

¹C.f. the matrix Γ in [7] for a way to ameliorate the problem of differing numbers of residues per protein and how to judiciously select which ones are relevant for protein function and therefore comparison.

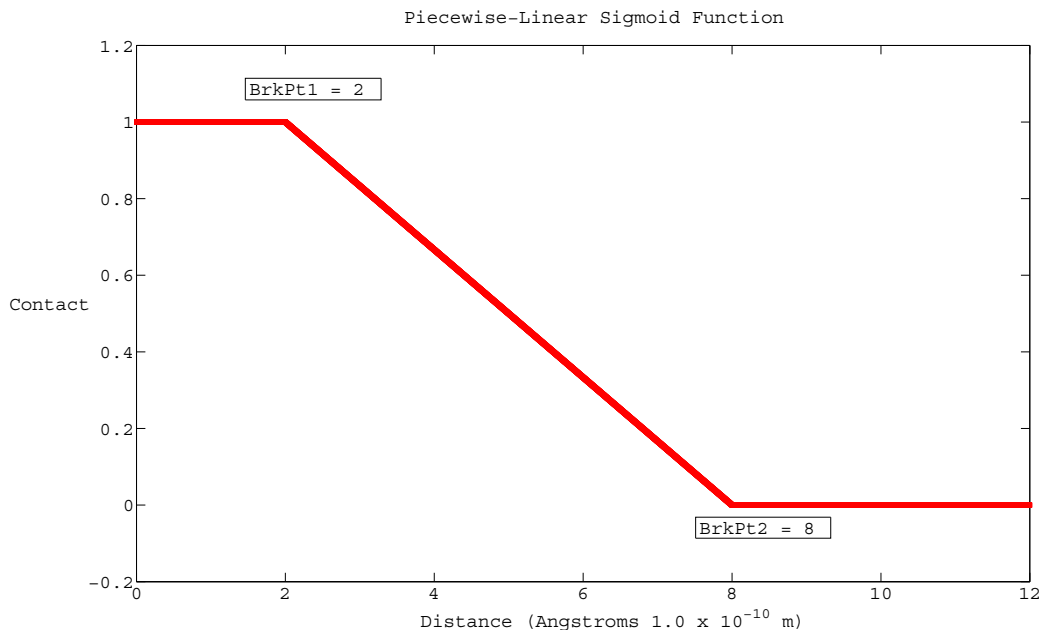


Figure 1: Graph of $C(2, 8)$ as defined in (1)

2.1 Example of determining a contact matrix

Consider the coordinate matrix X , corresponding distance matrix M , and corresponding contact matrix C below.

$$X = \begin{bmatrix} 2 & 2 & 0 \\ 8 & -6 & 0 \\ 5 & -2 & 0 \end{bmatrix} \implies M = \begin{bmatrix} 0 & 10 & 5 \\ 10 & 0 & 5 \\ 5 & 5 & 0 \end{bmatrix} \implies C = \begin{bmatrix} 1 & 0 & \frac{1}{2} \\ 0 & 1 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 1 \end{bmatrix}$$

The element in the first row and second column of M is calculated as follows:

$$M_{1,2} = \|(2, 2, 0) - (8, -6, 0)\| = \sqrt{(2-8)^2 + (2-(-6))^2 + (0-0)^2} = 10$$

All other elements of M are calculated similarly. The element in the first row and third column of C is calculated as defined in (1):

$$C_{1,3}(2, 8) = 1 - \frac{M_{1,3} - 2}{8 - 2} = 1 - \frac{5 - 2}{8 - 2} = \frac{1}{2}$$

In this example, $C_{1,1}(2, 8) = 1$ means that residue 1 has full contact with itself, $C_{1,2}(2, 8) = 0$ means that residue 1 has no contact with residue 2, and $C_{1,3}(2, 8) = \frac{1}{2}$ means that residue 1 has some contact with residue 3. Figure 1 provides further intuition about the meaning of each contact value.

3 Problem Statement

We define a search space over three collections of linear operators. Let

- \mathcal{P} be the collection of all permutation matrices,
- \mathcal{R} be the collection of all unitary matrices, and
- \mathcal{I} be the collection of all axial reflections, i.e., \mathcal{I} is the set of all diagonal matrices I^\pm for which each diagonal element is either 1 or -1 . The diagonal element in the i th row of I^\pm is denoted η_i .

Let $\|\cdot\|$ represent the matrix 2-norm (or equivalently the Frobenius norm) unless indicated otherwise. The alignment problem is

$$\min\{\|C' - \Theta C''\Omega\| : \Theta\Omega^T W = UI^\pm, I^\pm \in \mathcal{I}, \Theta \in \mathcal{R}, \Omega \in \mathcal{P}\}. \quad (2)$$

As stated, the optimization problem studies the deviation between the smooth contact maps, and we refer to it as the deviation problem statement. Two proteins align well if their contact matrices align well, allowing for permutation, by Ω , and signing, by I^\pm , of the columns. The role of Θ is to align the eigenspaces of the contact matrices. We rewrite the optimization problem in three distinct forms that remove Θ by substitution and focus on the remaining two parameters Ω and I^\pm . Each form lends itself to a different approach to solving the problem.

3.1 Matrix form of a bilinear program

The first form in which we state the problem considers a minimization of the matrix 2-norm as in (2). We alter the expression inside the norm. First, substituting $\Theta = UI^\pm W^T$, $C' = UD'U^T$ and $C'' = WD''W^T$, we have

$$\begin{aligned} C' - \Theta C''\Omega &= UD'U^T - UI^\pm W^T W D'' W^T \Omega \\ &= U(D'U^T - I^\pm D''W^T\Omega). \end{aligned}$$

Since the 2-norm is invariant under unitary transformations,

$$\|U(D'U^T - I^\pm D''W^T\Omega)\| = \|D'U^T - I^\pm D''W^T\Omega\| = \|D' - I^\pm D''W^T\Omega U\|.$$

We rewrite the deviation problem in the left matrix form as

$$\min\{\|D'U^T - I^\pm D''W^T\Omega\| : I^\pm \in \mathcal{I}, \Omega \in \mathcal{P}\}, \quad (3)$$

and in the right matrix form as

$$\min\{\|D' - I^\pm D''W^T\Omega U\| : I^\pm \in \mathcal{I}, \Omega \in \mathcal{P}\}. \quad (4)$$

The matrix forms of a bilinear program are useful for calculating a worst-case I^\pm .²

²See Section 7, A Simplified Calculation of I^\pm .

3.2 Angle calculation

To obtain the second form of the problem, we exploit properties of the 2-norm and simplify the right matrix form in (4). Specifically, we modify a simplification of the problem shown in [7] to rewrite the minimization of the 2-norm as a maximization of a finite sum. Since we are considering an optimization problem over I^\pm and Ω , minimizing $\|\cdot\|^2$ minimizes $\|\cdot\|$. Then, by a property of the 2-norm,

$$\begin{aligned}
& \|D' - I^\pm D'' W^T \Omega U\|^2 \\
&= \text{tr}((D' - I^\pm D'' W^T \Omega U)^T (D' - I^\pm D'' W^T \Omega U)) \\
&= \text{tr}((D' - U^T \Omega^T W D'' I^\pm)(D' - I^\pm D'' W^T \Omega U)) \\
&= \text{tr}((D')^2 - D' I^\pm D'' W^T \Omega U - U^T \Omega^T W D'' I^\pm D' + U^T \Omega^T W D'' I^\pm I^\pm D'' W^T \Omega U) \\
&= \text{tr}((D')^2 - D' I^\pm D'' W^T \Omega U - U^T \Omega^T W D'' I^\pm D' + U^T \Omega^T W (D'')^2 W^T \Omega U) \\
&= \text{tr}((D')^2) - \text{tr}(D' I^\pm D'' W^T \Omega U) - \text{tr}(U^T \Omega^T W D'' I^\pm D') + \text{tr}(U^T \Omega^T W (D'')^2 W^T \Omega U).
\end{aligned}$$

Since $\text{tr}(A) = \text{tr}(A^T)$ for any matrix A ,

$$\|D' - I^\pm D'' W^T \Omega U\|^2 = \text{tr}(D'^2) + \text{tr}(U^T \Omega^T W (D'')^2 W^T \Omega U) - 2\text{tr}(D' I^\pm D'' W^T \Omega U).$$

Optimizing over I^\pm and Ω allows us to ignore the first two terms. The first term is clearly invariant under the optimization. For the second term, note that $\text{tr}(U^T \Omega^T W (D'')^2 W^T \Omega U) = \|D'' W^T \Omega U\|^2$. Then since W , U , Ω and their respective transpose matrices are unitary, $\|D'' W^T \Omega U\|^2 = \|D''\|^2 = \text{tr}((D'')^2)$. Therefore, the second term is invariant under the optimization as well. The problem reduces to

$$\max\{\text{tr}(D' I^\pm D'' W^T \Omega U) : I^\pm \in \mathcal{I}, \Omega \in \mathcal{P}\}. \quad (5)$$

The problem as stated in (5), provides geometric intuition about the angle between paired eigenvectors. The terms in the trace sum are dot products of rows of W^T and columns of ΩU , both of which are eigenvectors, and the dot product of two vectors is inversely proportional to the angle between them. So, by maximizing the trace calculation, we are minimizing the angles between the paired eigenvectors.

3.3 A bilinear program based on the assignment problem

The trace calculation of an n -dimensional matrix is a finite sum of n terms. In order to maximize the trace calculation, we want each of these n terms to be as large as possible. Choosing the best combination of terms is directly related to the choice of either 1 or 0 as the

value of Ω_{ij} for each pair (i, j) . We consider the assignment problem:

$$\begin{aligned} & \max \sum_{ij}^n c_{ij} \Omega_{ij} \\ \text{subject to } & \sum_i^n \Omega_{ij} = 1 \\ & \sum_j^n \Omega_{ij} = 1 \\ & \Omega_{ij} \in \{0, 1\}. \end{aligned}$$

In this model, c_{ij} represents the benefit of choosing 1 as the value of Ω_{ij} . In the expression in the interior of our trace calculation, the permutation matrix Ω is acting on the rows of U . Since Ω is pairing a row of U with a column of W^T , we say that Ω is pairing a row of U with a row of W . Specifically, if the value of Ω_{ij} is 1, then the j th row of U is paired with the i th row of W . We must also incorporate I^\pm , D' and D'' . Note that all three matrices act on the columns of W . In particular, η_k , λ'_k and λ''_k act on the k th column of W . We thus define c_{ij} by

$$c_{ij} = \sum_k^n \eta_k \lambda'_k \lambda''_k W_{ik} U_{jk}$$

The bilinear optimization problem associated with (4) is

$$\begin{aligned} & \max \sum_{ij}^n \left(\sum_k^n \eta_k \lambda'_k \lambda''_k W_{ik} U_{jk} \right) \Omega_{ij} \tag{6} \\ \text{subject to } & \sum_i^n \Omega_{ij} = 1 \\ & \sum_j^n \Omega_{ij} = 1 \\ & \Omega_{ij} \in \{0, 1\}, \eta_k \in \{-1, 1\}. \end{aligned}$$

The assignment problem has been studied in detail, and so, modelling our problem in this way provides us with a tool for finding a quality solution. Previous work with this protein alignment technique has shown that if one knows the optimal I^\pm , then finding the optimal Ω is possible in polynomial time. When the problem is written as in (6), it is clear that if we know the value of c_{ij} for all (i, j) pairs, then we can calculate Ω_{ij} for all (i, j) pairs in polynomial time. As defined above, each c_{ij} depends on I^\pm . Note that if one could find an optimal Ω , then one could find the optimal I^\pm in linear time.

Both the calculation of I^\pm and the calculation of Ω involve large search spaces, 2^n and $n!$, respectively. Reformulating the alignment problem, however, yields two useful results. In Section 4, we bound the optimization problem independent of both I^\pm and Ω . In Section 7, we will explain a more efficient calculation of I^\pm than previously presented in [7].

4 Bounding the Problem

In this section we derive both an upper and lower bound for the problem dependent only on the eigenvalues of the contact matrices, C' and C'' . A lower spectral bound allows us to analyse the implications of a perfect alignment with regard to the eigenvalues.

4.1 Upper and lower bounds

Recall the equality developed in 3.2:

$$\|D' - I^\pm D'' W^T \Omega U\|^2 = \text{tr}((D')^2) + \text{tr}((D'')^2) - 2\text{tr}(D' I^\pm D'' W^T \Omega U).$$

Also recall that W , U , Ω and their respective transpose matrices are unitary, and any product of unitary matrices is also unitary. Since each diagonal element of a unitary matrix is less than or equal to 1 and greater than or equal to -1 , we bound the expression $\text{tr}(D' I^\pm D'' W^T \Omega U)$ as follows:

$$-\sum_{i=1}^n (|\lambda'_i| |\lambda''_i|) \leq \text{tr}(D' I^\pm D'' W^T \Omega U) \leq \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|).$$

Then,

$$2 \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|) \geq -2\text{tr}(D' I^\pm D'' W^T \Omega U) \geq -2 \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|)$$

Adding $\text{tr}((D')^2) + \text{tr}((D'')^2)$ through the inequality, an upper bound for the left matrix form and thus the deviation problem in (2) is

$$\sum_{i=1}^n ((\lambda'_i)^2 + (\lambda''_i)^2) + 2 \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|), \quad (7)$$

and a lower bound for (2) is

$$\sum_{i=1}^n ((\lambda'_i)^2 + (\lambda''_i)^2) - 2 \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|). \quad (8)$$

Factoring this out, we have the equivalent form of this lower bound,

$$\sum_{i=1}^n (|\lambda'_i| - |\lambda''_i|)^2.$$

This form shows that lower bound (8) is always non-negative.

4.2 Tightening the bounds

These bounds give the possible range of the equation

$$\|D' - I^\pm D'' W^T \Omega U\|$$

in the situation that D' and D'' are known, but W , U , Ω and I^\pm could be any matrices in their respective domains. In our situation, however, we intend to choose Ω and I^\pm to minimize the result, and it is possible to use this fact to achieve a tighter bound on our overall problem.

Consider Equation (4.1):

$$-\sum_{i=1}^n (|\lambda'_i| |\lambda''_i|) \leq \text{tr}(D' I^\pm D'' W^T \Omega U) \leq \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|).$$

The upper bound is as small as it can be, because it is possible that $W = U$, in which case we set $\Omega = I$ to get $W^T \Omega U = U^T U = I$, and we can choose I^\pm to make the diagonal elements of the $D' I^\pm D''$ positive and achieve the upper bound.

However, we will never achieve the lower bound, because we can always choose I^\pm so that the signs of the elements on the diagonal of $D' I^\pm D'' W^T \Omega U$ are non-negative, which will make the trace the sum of non-negative real numbers, which means it will always be at least zero.

Furthermore, zero is the least upper bound on this equation when we consider all W and U , because it is possible that

$$W^T = \begin{bmatrix} \frac{1}{\sqrt{n}} & \cdots & \frac{1}{\sqrt{n}} & \frac{-1}{\sqrt{n}} & \cdots & \frac{-1}{\sqrt{n}} \\ \vdots & \ddots & \frac{1}{\sqrt{n}} & \frac{-1}{\sqrt{n}} & \ddots & \vdots \\ \frac{1}{\sqrt{n}} & \cdots & \frac{1}{\sqrt{n}} & \frac{-1}{\sqrt{n}} & \cdots & \frac{-1}{\sqrt{n}} \end{bmatrix}, \quad U = \begin{bmatrix} \frac{1}{\sqrt{n}} & \cdots & \frac{1}{\sqrt{n}} \\ \vdots & \ddots & \vdots \\ \frac{1}{\sqrt{n}} & \cdots & \frac{1}{\sqrt{n}} \end{bmatrix}$$

where W^T has $\frac{n}{2}$ columns of $\frac{1}{\sqrt{n}}$ and $\frac{n}{2}$ columns of $\frac{-1}{\sqrt{n}}$.

In this situation, $\Omega U = U \forall \Omega \Rightarrow W^T \Omega U = \frac{n}{2\sqrt{n}} + \frac{-n}{2\sqrt{n}} = 0$, so $\text{tr}(D' I^\pm D'' W^T \Omega U) = \text{tr}(0) = 0 \forall \Omega$.

This discussion has proven the following theorem.

Theorem 4.2.1. *For any known D' and D'' ,*

$$0 \leq \text{tr}(D' I^\pm D'' W^T \Omega U) \leq \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|),$$

and there exist W and U such that the trace expression will attain both of those bounds.

This means the overall minimization problem (4) is

$$\begin{aligned} & \sum_{i=1}^n ((\lambda'_i)^2 + (\lambda''_i)^2) - 2 \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|) \\ & \leq \min \{ \|D' - I^\pm D'' W^T \Omega U\| : I^\pm \in \mathcal{I}, \Omega \in \mathcal{P} \} \\ & \leq \sum_{i=1}^n ((\lambda'_i)^2 + (\lambda''_i)^2). \end{aligned}$$

Note that is the smallest range we can get without considering W and U .

4.3 A perfect alignment

Definition 4.3.1. We say that two proteins are in perfect alignment if

$$\|D' - I^\pm D'' W^T \Omega U\| = 0.$$

Lemma 4.3.1. If two proteins are in perfect alignment, then the eigenvalues of their respective contact matrices have the same magnitude.

Proof. Suppose two proteins are in perfect alignment. Then by definition 4.3.1,

$$\|D' - I^\pm D'' W^T \Omega U\| = 0.$$

Since (8) is a non-negative lower bound for $\|D' - I^\pm D'' W^T \Omega U\|$, (8) has a value of zero. Note that

$$\sum_{i=1}^n ((\lambda'_i)^2 + (\lambda''_i)^2) - 2 \sum_{i=1}^n (|\lambda'_i| |\lambda''_i|) = 0$$

if and only if $|\lambda'_i| = |\lambda''_i|$ for all i . □

Lemma 4.3.2. If two proteins are in perfect alignment, then $W^T \Omega U = \hat{I}^\pm$, where $\hat{I}^\pm \in \mathcal{I}$.

Proof. Suppose two proteins are in perfect alignment. By Definition 4.3.1,

$$\|D' - I^\pm D'' W^T \Omega U\| = 0$$

$$\iff$$

$$D' - I^\pm D'' W^T \Omega U = 0$$

$$\iff$$

$$D' = I^\pm D'' W^T \Omega U$$

Then since D' is a diagonal matrix, $I^\pm D'' W^T \Omega U$ is diagonal as well. Since I^\pm and D'' are diagonal by definition, $W^T \Omega U$ is diagonal, and thus, each column of $W^T \Omega U$ contains only one non-zero element, namely the diagonal element. Since $W^T \Omega U$ is the product of unitary matrices, the Euclidean norm of each of its columns is equal to 1. Therefore, every diagonal element of $W^T \Omega U$ is either 1 or -1 . □

Theorem 4.3.1. Two proteins are in perfect alignment if and only if $|\lambda'_i| = |\lambda''_i|$ for all i , and $W \hat{I}^\pm U^T = \Omega$, where $\hat{I}^\pm \in \mathcal{I}$ and $\Omega \in \mathcal{P}$.

Proof. (\Rightarrow) Suppose two proteins are in perfect alignment. By Lemma 4.3.1, $|\lambda'_i| = |\lambda''_i|$ for all i , and by Lemma 4.3.2, $W^T \Omega U = \hat{I}^\pm$. Using properties of unitary matrices, the equation is rearranged as $W \hat{I}^\pm U^T = \Omega$.

(\Leftarrow) Suppose $|\lambda'_i| = |\lambda''_i| \forall i$, and $W^T \Omega U = \hat{I}^\pm$. Then $\|D' - I^\pm D'' W^T \Omega U\| = \|D' - I^\pm D'' \hat{I}^\pm\|$. D' and D'' are the same in absolute value, and we can choose any I^\pm , so we will choose the I^\pm that makes their signs the same. Therefore $\|D' - I^\pm D'' \hat{I}^\pm\| = \|D' - D'\| = 0$. □

5 Continuity of the Bounds for the Deviation Problem

Unlike discrete approaches to protein structure alignment such as the contact map overlap problem, the bounds of the bilinear program and its reformulations are continuous under small perturbations in the breakpoints, B_1 and B_2 , that define our contact map, $C(B_1, B_2)$. This is biologically significant because certain physical conditions may alter the contact of residues. Consider ϵ , not necessarily positive, so that if B_1 and B_2 are our original break points, then $\mathfrak{B}_1 = B_1 + \epsilon$ and $\mathfrak{B}_2 = B_2 + \epsilon$ are our perturbed breakpoints. If given a contact map $C'(B_1, B_2)$, define a perturbed contact matrix $\mathfrak{C}' = C' + E'$ so that

$$[\mathfrak{C}'(\mathfrak{B}_1, \mathfrak{B}_2)]_{i,j} = \begin{cases} 1 & \text{if } M_{ij} \leq \mathfrak{B}_1 \\ 0 & \text{if } M_{ij} \geq \mathfrak{B}_2 \\ 1 - \frac{M_{ij} - \mathfrak{B}_1}{\mathfrak{B}_2 - \mathfrak{B}_1} & \text{if } \mathfrak{B}_1 < M_{ij} < \mathfrak{B}_2. \end{cases}$$

Substituting $\mathfrak{B}_1 = B_1 + \epsilon$ and $\mathfrak{B}_2 = B_2 + \epsilon$, and noting that $\mathfrak{C}'(\mathfrak{B}_1, \mathfrak{B}_2) = \mathfrak{C}'(B_1, B_2, \epsilon)$,

$$[\mathfrak{C}'(B_1, B_2, \epsilon)]_{i,j} = \begin{cases} 1 & \text{if } M_{ij} \leq B_1 + \epsilon \\ 0 & \text{if } M_{ij} \geq B_2 + \epsilon \\ 1 - \frac{M_{ij} - B_1}{B_2 - B_1} + \frac{\epsilon}{B_2 - B_1} & \text{if } B_1 + \epsilon < M_{ij} < B_2 + \epsilon. \end{cases}$$

We now have $\mathfrak{C}'(B_1, B_2, \epsilon) = C'(B_1, B_2) + E'(\epsilon)$, where C' is our original contact matrix (1) and

$$[E'(\epsilon)]_{i,j} = \begin{cases} \frac{M_{ij} - B_1}{B_2 - B_1} & \text{if } B_1 < M_{ij} \leq B_1 + \epsilon \\ \frac{\epsilon}{B_2 - B_1} & \text{if } B_1 + \epsilon < M_{ij} < B_2 \\ 1 - \frac{M_{ij} - B_1}{B_2 - B_1} + \frac{\epsilon}{B_2 - B_1} & \text{if } B_2 \leq M_{ij} < B_2 + \epsilon \\ 0 & \text{otherwise.} \end{cases}$$

For the same reason that C' was normal, \mathfrak{C}' is also normal. Let it have the eigen decomposition $\mathfrak{C}' = \mathfrak{U}\mathfrak{D}'\mathfrak{U}^T$. Similarly, for our other perturbed contact matrix, we have $\mathfrak{C}'' = \mathfrak{W}\mathfrak{D}''\mathfrak{W}^T$. Thus, our right matrix form (4), reformulated for the perturbed contact matrix is

$$\|\mathfrak{D}' - I^\pm \mathfrak{D}'' \mathfrak{W}^T \mathfrak{U}\|. \quad (9)$$

5.1 Perturbed spectral bounds

In order to analyse the stability of the spectrum of our original contact matrix C' , we develop an expression similar to our spectral upper bound (7). First, note the following relationship between the matrix norm presented above and the spectral norm used to analyse the perturbed matrix, see [1] for this and other results.

Proposition 5.1.1. *For any matrix A , we have that*

$$\|A\|_2 \leq \|A\|_F \leq \sqrt{r} \|A\|_2$$

where r is the rank of the matrix.

If C', E' , and \mathcal{C}' are normal, then they have full row rank; so, $r = n$. However, it is conceivable that a perturbation in the breakpoints of our sigmoid function will make two columns of Q' linearly dependent and therefore reduce the row rank. We discuss this later as we consider the limitations on the breakpoints. Now that we have established a norm relationship, we use the following spectral result to give an upper bound for our perturbed problem. Let C' and E' have eigenvalues $\lambda_1, \dots, \lambda_n$ and μ_1, \dots, μ_n , respectively.

Lemma 5.1.1. *If C' and E' are normal and our norm $\|\cdot\|$ is spectral, then for every i and any fixed $\alpha \in \mathbb{R}$, the eigenvalues of $\mathcal{C}' = C' + E'$ are bounded by*

$$\{z : |z - (\alpha + \lambda_i)| \leq \max_i |\mu_i - \alpha|\}.$$

A proof of this lemma is found in [4]. Let ψ_1, \dots, ψ_n be the eigenvalues of \mathcal{C}' . Then, the preceding lemma implies that for every i ,

$$|\psi_i - (\alpha + \lambda_i)| \leq \max_i |\mu_i - \alpha|.$$

Since $\psi_i \in \mathbb{R}$, we have

$$(\alpha + \lambda_i) - \max_i |\mu_i - \alpha| \leq \psi_i \leq (\alpha + \lambda_i) + \max_i |\mu_i - \alpha|.$$

Thus, for every $\alpha \in \mathbb{R}$,

$$|\psi_i| \leq \min\{|\alpha + \lambda_i - \max_i |\mu_i - \alpha||, |\alpha + \lambda_i + \max_i |\mu_i - \alpha||\}.$$

We define

$$\rho_i(\alpha) := \min\{|\alpha + \lambda_i - \max_i |\mu_i - \alpha||, |\alpha + \lambda_i + \max_i |\mu_i - \alpha||\}.$$

Applying the same logic to find the unperturbed spectral upper bounds (7), by Definition 5.1.1 the upper bounds for the perturbed problem is

$$\sqrt{n} \left[\sum_{i=1}^n ((\rho'_i)^2 + (\rho''_i)^2) + 2 \sum_{i=1}^n (|\rho'_i| |\rho''_i|) \right]. \quad (10)$$

Similarly, the lower bound is,

$$\sum_{i=1}^n ((\rho'_i)^2 + (\rho''_i)^2) - 2 \sum_{i=1}^n (|\rho'_i| |\rho''_i|) \quad (11)$$

Note that this is never negative by an earlier calculation.

5.2 Continuity of perturbed bounds

We now show that small changes in ϵ produce small changes in the perturbed bounds (10) and (11). Clearly, any change in ϵ for fixed B_1 and B_2 change the eigenvalues μ_1, \dots, μ_n of $[E'(\epsilon)]_{i,j}$ and therefore the bounds. However, this is a continuous change.

Lemma 5.2.1. *Let A be an $n \times n$ symmetric matrix with eigenvalues $\lambda_1 \leq \dots \leq \lambda_n$. Then, λ_i is continuous under perturbations of A .*

A proof of this Lemma is found in [2]. Since the eigenvalues of $E'(\epsilon)$ are continuous in ϵ , our bounds (10) and (11) are continuous with respect to ϵ . However, if ϵ changes too drastically, there is the potential for the perturbed contact matrix \mathfrak{C}' to lose row rank. We give a condition on ϵ in terms of B_1 and B_2 to guarantee this cannot happen. First, we require a lemma and corollary to the lemma. Let A^\dagger represent the Moore-Penrose generalized inverse of $A \in \mathbb{C}^{m \times n}$ and $\|\cdot\|$ be any matrix norm.

Lemma 5.2.2. *If $E \in \mathbb{C}^{m \times n}$ and*

$$\|E\| < \frac{1}{\|A^\dagger\|},$$

then $\text{rank}(A + E) \geq \text{rank}(A)$.

A proof of this lemma is given in [6]. Assume $\dim(C') = \text{rank}(C') = n$.

Corollary 5.2.1. *If $\mathfrak{C}' = C' + E'$, $\text{rank}(C') = n$, and*

$$\|E'\| < \frac{1}{\|C'^{-1}\|},$$

then $\text{rank}(\mathfrak{C}') = n = \text{rank}(C')$.

Proof. Since C' is a normal matrix, it is invertible. So, $C'^\dagger = C'^{-1}$. Now, the result follows from Lemma 5.2.2 because $\text{rank}(\mathfrak{C}') \neq n$ by definition of \mathfrak{C}' . \square

We are now ready to give a condition for which perturbations in ϵ will not produce loss of row rank. Let $\max_{ij} |[C'^{-1}]_{i,j}|$ denote the largest entry in absolute value of C'^{-1} .

Theorem 5.2.1. *For any distance matrix $M_{i,j}$, if*

$$\epsilon < \left| \frac{(B_2 - B_1)}{\max_{ij} |[C'^{-1}]_{i,j}|} \right| \tag{12}$$

then $\text{rank}(\mathfrak{C}') = \text{rank}(C')$.

Proof. Consider the max norm defined for some matrix A as $\|A\| = \max_{ij} |A_{ij}|$. Then,

$$\|[C'^{-1}]_{i,j}\| = \max_{ij} |[C'^{-1}]_{i,j}|.$$

Because C' is not the zero matrix, there is some $c_{ij} \in C'^{-1}$ so that $|c_{ij}| > 0$. So,

$$\max_{ij} |[C'^{-1}]_{i,j}| \neq 0$$

and the max norm for C'^{-1} is well defined. Now, by the definition of E' ,

$$\|E'(\epsilon)\| = \max_{ij} |E'_{i,j}| \leq \frac{\epsilon}{B_2 - B_1}.$$

Condition (12) implies that

$$\frac{\epsilon}{B_2 - B_1} < \frac{1}{\max_{ij} |[C'^{-1}]_{i,j}|}.$$

So, using the max norm, we have that

$$\|E'\| \leq \frac{\epsilon}{B_2 - B_1} < \frac{1}{\max_{ij} |[C'^{-1}]_{i,j}|} = \frac{1}{\|C'^{-1}\|}.$$

Thus, Corollary 5.2.1 implies that $\text{rank}(\mathfrak{C}') = n = \text{rank}(C')$. \square

This theorem shows that as long as condition (12) holds, then by Lemma 5.2.1, our bounds are continuous.

6 Conditional Bounds for the Program

As shown by the bilinear program developed in Section 3.3, if we know either I^\pm or Ω , we can calculate the other. In this section, we show that if we know either I^\pm or Ω , then we can establish worst-case bounds on the problem by considering all possibilities for the other.

6.1 Known I^\pm

Let us consider the problem in the left matrix form shown in (3), where I^\pm is fixed:

$$\min\{\|D'U^T - I^\pm D''W^T\Omega\| : \Omega \in \mathcal{P}\}.$$

We can distribute the matrix 2-norm over the entries in the matrix. We know that Ω permutes the columns of $I^\pm D''W^T$ but does not change the set of elements in each row, so first we distribute the norm over the rows:

$$\min\left\{\sum_k \|(D'U^T)_{k:} - (I^\pm D''W^T\Omega)_{k:}\| : \Omega \in \mathcal{P}\right\}.$$

To find a lower bound we will consider subtracting any element of the k th row of $D'U^T$ from any element of the k th row of $I^\pm D''W^T\Omega$. The greatest that the norm of such a subtraction could be is

$$S_k = \max\left\{\max_i \{(D'U^T)_{ki}\} - \min_j \{(I^\pm D''W^T)_{kj}\}, \min_i \{(D'U^T)_{ki}\} - \max_j \{(I^\pm D''W^T)_{kj}\}\right\}.$$

Since there are n element-by-element subtractions per row, we take the worst-case bound for the row to be $n \cdot S_k$. We bound the left matrix form with fixed I^\pm as follows:

$$\min\{\|D'U^T - I^\pm D''W^T\Omega\| : \Omega \in \mathcal{P}\} \leq \sum_k n \cdot S_k.$$

When we consider different I^\pm matrices, the bound for each row depends on our choice of I^\pm . In this context, we let S_k^+ be the bound on row k when $I_{kk}^\pm = 1$, and S_k^- be the bound on row k when $I_{kk}^\pm = -1$.

6.1.1 An approximation algorithm for I^\pm

This upper bound is the basis for the approximation algorithm given in [7]. The algorithm considers this upper bound as a function of I^\pm and chooses an I^\pm that minimizes that function, and then finds the best Ω for the constructed I^\pm . This is not guaranteed to produce the best (I^\pm, Ω) pair out of all possible pairs, but numerical tests on a data set with known matchings show that this method produces results close to the correct ones [7]. We describe the algorithm with our notation as follows:

Algorithm 1 The approximation algorithm from [7] for choosing I^\pm

```

for all  $k \in \{1 \dots n\}$  do
  if  $S_k^+ > S_k^-$  then
     $I_{kk}^\pm = 1$ 
  else if  $S_k^+ < S_k^-$  then
     $I_{kk}^\pm = -1$ 
  else
     $I_{kk}^\pm = 0$ 
  end if
end for

```

The algorithm assigns $I_{kk}^\pm = 0$ in the case where $S_k^+ = S_k^-$ because it was considered better to ignore the information from row k rather than to make a potentially non-optimal choice of I_{kk}^\pm that could have adversely affected the later choice of Ω .

Each S_k only depends on $(I^\pm)_{kk}$, not the other elements of I^\pm , so by considering the $2n$ S_k^+ and S_k^- , we have actually minimized our bound over all 2^n possible I^\pm matrices.

6.2 Known Ω

By analogy to the previous subsection, it is possible to derive a bound on our problem in the case that Ω is fixed. In this case, we use the fact that I^\pm changes the sign of the rows of $D''W^T\Omega$ but makes no other change. First, we distribute the norm over every entry in the result matrix:

$$\{\|D'U^T - I^\pm D''W^T\Omega\| : I^\pm \in \mathcal{I}\} = \sum_{i,j} \{\|(D'U^T - I^\pm D''W^T\Omega)_{ij}\|, I^\pm \in \mathcal{I}\}.$$

To find a lower bound, we observe that $\|a - b\| \leq \|a\| + \|b\|$, and the worst-case element-by-element subtraction occurs when the quantities being subtracted have different signs. We bound the left matrix form with fixed Ω as follows:

$$\{\|D'U^T - I^\pm D''W^T\Omega\| : I^\pm \in \mathcal{I}\} \leq \sum_{i,j} \{ |(D'U^T)_{i,j}| + |(I^\pm D''W^T\Omega)_{ij}| \}.$$

6.2.1 An approximation algorithm for Ω

We can also use the bound in (??) to create an approximation algorithm. We choose an Ω matrix that minimizes the bound, and then choose an I^\pm that minimizes the deviation problem given this Ω .

Since Ω pairs columns of $D'U^T$ with columns of $I^\pm D''W^T$, we regroup our sum as a sum by columns. When we consider different choices for Ω , we let S_{kp} be the upper bound on the portion of the norm that comes from the subtraction of columns k and p :

$$\| |D'U_{:k}^T| - |D''W_{:p}^T| \|^2.$$

Given any C' and C'' , there are n^2 such bounds S_{kp} , and the problem is to choose n of them such that each possible value of k and p is included exactly once. This is the assignment problem, and we can solve it with standard techniques.

Once we have this Ω , we can find an optimal I^\pm by using the S^+ and S^- technique from the previous section.

7 A Simplified Calculation of I^\pm

The algorithm shown in 6.1.1 can be improved. We want to calculate I^\pm both optimally and efficiently. If we orient an eigenvector incorrectly, we will have poor alignment results. Furthermore, we want to accurately determine as many elements of I^\pm as possible so that we do not lose any information. In order to improve the calculation of I^\pm , we simplify the right matrix form in (4).

Specifically, we consider (4) ignoring D' and D'' , so we have $\|I - I^\pm W^T \Omega U\|$. As a result, we consider only the eigenvector information. We let $V = W^T \Omega U$. Note that $V^T = U^T \Omega^T W$, and so

$$\begin{aligned} VV^T &= U^T \Omega^T W W^T \Omega U \\ &= I. \end{aligned}$$

Thus, V is a unitary matrix. We call this the unscaled version of the problem:

$$\min\{\|I - I^\pm V\|_p^2 : I^\pm \in \mathcal{I}, \Omega \in \mathcal{P}\}. \quad (13)$$

7.1 Defining a signal

Even with our simplifications, each diagonal element of I^\pm must be determined individually, and we will do so by comparing the elements in the i th row of I with the elements in the i th row of V . Notationally, e_{ij} is the component in the i th row and j th column of I , and similarly, v_{ij} is the component in the i th row and j th column of V .

For each row i , let

$$B_i^+ = \max_j (\{e_{ij}\} \cup \{v_{ij}\}) - \min_j (\{e_{ij}\} \cup \{v_{ij}\})$$

$$B_i^- = \max_j (\{e_{ij}\} \cup \{-v_{ij}\}) - \min_j (\{e_{ij}\} \cup \{-v_{ij}\}).$$

If $B_i^+ > B_i^-$, we prefer B_i^- and choose $I_{ii}^\pm = -1$. If $B_i^- > B_i^+$, we prefer B_i^+ and choose $I_{ii}^\pm = 1$. If $B_i^+ = B_i^-$, we get no information from this test and are forced to set $I_{ii}^\pm = 0$.

Definition 7.1.1. We say that there is a signal in row i if $B_i^+ \neq B_i^-$.

A signal by our definition tells us that choosing either 1 or -1 for I_{ii}^\pm will improve our minimization problem. We are interested in defining the necessary condition for no signal as well. Since V is unitary and thus orthonormal, we know that

$$|v_{ij}| \leq 1 \quad \forall i, j.$$

Therefore, $\max_j (\{e_{ij}\} \cup \{v_{ij}\}) = 1$, and so, $B_i^+ = 1 - \min_j (\{e_{ij}\} \cup \{v_{ij}\})$. Similarly, $B_i^- = 1 - \min_j (\{e_{ij}\} \cup \{-v_{ij}\})$. We note that $\min_j (\{e_{ij}\} \cup \{v_{ij}\})$ will be zero unless $\min_j \{v_{ij}\}$ is negative. Similarly, we note that $\min_j (\{e_{ij}\} \cup \{-v_{ij}\})$ will be zero unless $\max_j \{v_{ij}\}$ is positive. In particular, if $\min_j \{v_{ij}\} = -\max_j \{v_{ij}\}$, then $B_i^+ = B_i^-$. The condition for having no signal is then

$$B_i^+ = B_i^- \Leftrightarrow -v_{ij} = v_{ik} \text{ and } \max_m \{v_{im}\} = v_{ik},$$

where v_{ij} and v_{ik} are elements of the i th row of V .

7.2 Probability of a signal in every row

We would prefer that we never have $B_i^+ = B_i^-$ because in that case we set $I_{kk}^\pm = 0$ for some k , which means we ignore information in our algorithm. We will show that after restating the problem as shown in (13), $P(B_i \neq B_i^-) = 1$ for all i , i.e, we get a signal for every row i .

In order to prove this proposition we will use a decomposition of orthonormal matrices using Givens rotations. A Givens rotation is a matrix

$$G(i, j, \theta) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & c & \cdots & s & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & -s & \cdots & c & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix},$$

where $c = \cos(\theta)$ is in entry (i, i) and (j, j) and $s = \sin(\theta)$ is in entry (i, j) and its negative in (j, i) [1], or equivalently a rotation of angle θ through a plane defined by the i th and j th coordinate axes.

The central fact about Givens rotations is that any orthonormal matrix O can be written as

$$O = I^\pm G_1 \cdots G_m,$$

where I^\pm is the identity matrix with some of its diagonal entries being -1 , and G_1 through G_m Givens rotations. In particular, we can write the matrix V that way.

Lemma 7.2.1. *A Givens rotation only affects the coordinates of the axes on which it rotates.*

Proof. Let G be a Givens matrix that rotates through axes i and j , A be any matrix, and $k \neq i, j$. For any l ,

$$[GA]_{kl} = G_{k;A;l} = (0 \cdots 0 1 0 \cdots 0) \begin{pmatrix} a_{1l} \\ \vdots \\ a_{nl} \end{pmatrix} = a_{kl}.$$

Since this holds for all $k \neq i, j$ and all l , we see that $GA = A$ in all rows except the i th and j th. \square

Consider a Givens rotation G around axes i and j acting on matrix A , and choose two columns of A , k and l . Now consider what would be necessary for the resulting matrix to have the sort of symmetry in the k th and l th columns that could give us no signal along the i th axis.

We know that

$$[GA]_{i;} = (0 \cdots 0 \ c \ 0 \cdots 0 \ s \ 0 \cdots 0)A = cA_{i;} + sA_{j;},$$

so

$$[GA]_{ik} = ca_{ik} + sa_{jk} \text{ and } GA_{il} = ca_{il} + sa_{jl}.$$

In order for them to have the symmetry we want, we have

$$\begin{aligned} GA_{ik} &= -GA_{il} \\ \Leftrightarrow ca_{ik} + sa_{jk} &= -(ca_{il} + sa_{jl}) \\ \Leftrightarrow c(a_{ik} + a_{il}) &= -s(a_{jk} + a_{jl}) \\ \Leftrightarrow \frac{s}{c} &= \frac{-(a_{ik} + a_{il})}{a_{jk} + a_{jl}} = \tan(\theta). \end{aligned}$$

For any value of A , there will be at most two θ that satisfy this equation.

Theorem 7.2.1. *Symmetry occurs with probability zero in any continuous probability distribution.*

Proof. For any Givens rotation G acting on a matrix A , we know there are only two θ that can introduce a symmetry into columns we like. If we assume that the θ come from a continuous probability distribution, the probability that we introduce a symmetry into any two chosen columns of A is 0.

Finally, consider the n by n matrix V from the optimization problem written as

$$V = I^\pm G_1 \cdots G_m,$$

where G_1 through G_m are Givens rotations, and the angles of the Givens rotations come from continuous probability distributions. Then for each rotation,

$$\begin{aligned} & P(\text{that rotation introduces a symmetry}) \\ & \leq \binom{n}{2} P(\text{that rotation introduces a symmetry for two given columns of } V) \\ & = \binom{n}{2} 0 = 0, \end{aligned}$$

so

$$\begin{aligned} & P(\text{all the rotations together leave us with a symmetry}) \\ & \leq m \cdot 0 = 0. \end{aligned}$$

□

The actual distribution of the θ is unknown, but we believe it is reasonable to assume that the distribution is continuous, which makes this theorem a good sign for our algorithm.

7.3 Feasibility of no signal for every row

Another question that naturally arises is whether or not it is possible for there to be no signal in every row, i.e., $\forall i, B_i^+ = B_i^-$. This subsection shows that this is impossible; i.e., there is at least one row that gets a signal. To this end, we propose the following mathematical program that we will show is infeasible. Let n be the size of our matrix V . Consider

$$\min\{0\} \tag{14}$$

s.t.

$$\begin{aligned} \sum_j w_j &= n, w_j \in \{0, 1\} \\ \sum_i v_{ij} v_{ih} &= 0, j \neq h \\ \sum_i v_{ij} v_{ij} &= 1, \forall j \\ y_i &\leq v_{ij}, \forall i, j \\ z_i &\leq -v_{ij}, \forall i, j \\ -z_i - y_i &\leq 2(1 - w_i), \forall i. \end{aligned}$$

The parameters $\sum_i v_{ij}v_{ih} = 0$ and $\sum_i v_{ij}v_{ij} = 1$ insure orthogonality and $y_i \leq v_{ij} \leq -z_i$ force the v_{ij} term to zero in every row since w_i forces $-z_i - y_i \geq 0$ to zero. The Lagrangian of the above program (14) is

$$\begin{aligned} \mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) = & \alpha \left(\sum_i w_i - n \right) - \sum_{j \neq h} \lambda_{jh} \left(\sum_i v_{ij}v_{ih} \right) \\ & - \sum_j \sigma_j \left(\sum_i v_{ij}v_{ij} - 1 \right) \\ & - \sum_{j,i} \rho_{j,i} (v_{kj} - y_i) \\ & - \sum_{j,i} \eta_{j,i} (-v_{ij} - z_i) \\ & - \beta \sum_i 2(1 - w_i) + z_i + y_i, \end{aligned}$$

where ρ, η and β are non-negative and the other Lagrangian multipliers are unsigned. To show the infeasibility of (14), we show that the Lagrangian dual tends to infinity. The idea is to let the Lagrangian multipliers be sequences that iterate on t so that as t tends to infinity, the Lagrangian also tends to infinity. The proof follows.

Proof. Case 1: $\sum_i w_i \neq n$. Then, for some \hat{i} , $w_{\hat{i}} = 1$. For this index, let $\alpha = -t$ and the other Lagrangian multipliers be zero. So,

$$\begin{aligned} \lim_{t \rightarrow \infty} \mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) &= \lim_{t \rightarrow \infty} -t \sum_i w_i - n \\ &= \lim_{t \rightarrow \infty} -t(w_{\hat{i}} - n) \\ &= \lim_{t \rightarrow \infty} t(n - 1) \\ &= \infty. \end{aligned}$$

Thus, when $\sum_i w_i \neq n$, our mathematical program (14) is infeasible. **Case 2:** $\sum_i w_i = n$. Let $\lambda_{jh} = 0$ and $\sigma_j = 0$ for every j and h . Then, the Lagrangian reduces to

$$\begin{aligned} \mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) = & - \sum_{j,i} \rho_{j,i} (v_{kj} - y_i) \\ & - \sum_{j,i} \eta_{j,i} (-v_{ij} - z_i) \\ & - \beta \sum_i 2(1 - w_i) + z_i + y_i. \end{aligned}$$

Splitting the sums, we have

$$\begin{aligned}
\mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) = & - \sum_{j,i} \rho_{j,i} v_{kj} + \sum_{j,i} \rho_{j,i} y_i \\
& + \sum_{j,i} \eta_{j,i} v_{ij} + \sum_{j,i} \eta_{j,i} z_i \\
& - \beta \sum_i 2(1 - w_i) - \beta \sum_i z_i - \beta \sum_i y_i.
\end{aligned}$$

Combining the v_{ij} , y_i , and z_i terms, we find

$$\begin{aligned}
\mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) = & + \sum_{j,i} (\eta_{j,i} - \rho_{j,i}) v_{kj} \\
& + \sum_{j,i} \rho_{j,i} y_i - \beta \sum_i y_i \\
& + \sum_{j,i} \eta_{j,i} z_i - \beta \sum_i z_i \\
& - \beta \sum_i 2(1 - w_i).
\end{aligned}$$

Since $\sum_j -\frac{\beta}{n} \sum_i y_i = -\beta \sum_i y_i$,

$$\begin{aligned}
\mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) = & + \sum_{j,i} (\eta_{j,i} - \rho_{j,i}) v_{kj} \\
& + \sum_{j,i} (\rho_{j,i} - \frac{\beta}{n}) y_i \\
& + \sum_{j,i} (\eta_{j,i} - \frac{\beta}{n}) z_i \\
& - \beta \sum_i 2(1 - w_i).
\end{aligned}$$

Now we use our assumption that $\sum_i w_i = n$ and let $\rho_{j,i} = \eta_{j,i} = \frac{\beta}{n}$ for every i and j . Then,

$$\begin{aligned}
\mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) = & + \sum_{j,i} (0) v_{kj} \\
& + \sum_{j,i} (0) y_i \\
& + \sum_{j,i} (0) z_i \\
& + 2\beta(n - 1).
\end{aligned}$$

Let $\beta = t$. Then,

$$\begin{aligned}
\lim_{t \rightarrow \infty} \mathcal{L}(w, y, z, v, \alpha, \lambda, \sigma, \rho, \eta, \beta) & = \lim_{t \rightarrow \infty} 2t(n - 1) \\
& = \infty.
\end{aligned}$$

Thus, when $\sum_i w_i = n$, our mathematical program (14) is infeasible. This completes the proof. \square

8 Conclusion

The reformulations of the eigensystem-based protein structure alignment algorithm and the associated bounds described in this article provide a better understanding of the protein alignment problem and offer new tools to solve it optimally and efficiently. Our model, based on the assignment problem, is bilinear in the sense that if either Ω or I^\pm are known, then the problem reduces to a linear program; this offers intuition about the complexity of the problem. The exact complexity of the problem remains unknown, however, showing that there is possibly room to make the deviation problem solve faster and more optimally.

The bounds on the deviation problem show that error arises from two sources. The first source of error is associated with the difference in magnitude of the eigenvalues of two respective contact matrices. If the eigenvalues differ greatly enough, then the proteins will be dissimilar. This might provide a valuable tool in database wide alignments. If the eigenvalues are the same, then the lower bound is zero, but the minimization is not guaranteed to attain the bound. The fact that the deviation problem may not attain zero is due to a secondary source of error that arises if given an optimal \hat{I}^\pm , then $W\hat{I}^\pm U^T$ is not a permutation matrix. Understanding the error in the model engenders two key results: First, we can eliminate bad alignments based solely on the eigenvalues. Second, if the eigenvalues are similar, then we can reduce the problem to the calculation of the secondary source of error that is similar to an assignment problem.

9 Acknowledgements

We wish to express our deepest and most sincere gratitude to Professor Allen Holder for his guidance and enthusiasm in helping us complete a project of this magnitude. We would also like to thank the Rose-Hulman Institute of Technology Mathematics Department, specifically Profs. David Finn and Joshua Holden, for their comments and encouragement. Finally, we thank Rose-Hulman Institute of Technology and the National Science Foundation for giving us the opportunity to conduct such research and the catalyst for all three of our careers as practitioners of mathematics.

References

- [1] G.H Golub and C.F. Van Loan. *Matrix Computations*, Second Edition. The John Hopkins University Press, Baltimore and London, 1989.
- [2] Eugene E. Tyrtyshnikov. *A Brief Introduction to Numerical Analysis*. 3(9): 26. Birkhäuser Boston, 1997.
- [3] Mark T Oakley, Daniel Barthel, Turi Bykov, Jonathan M Garibaldi, Edmund K Burke, Natalio Krasnogor, and Jonathan D Hirst. Search Strategies in Structural Bioinformatics. *Curr Protein Pept Sci*, 9(3):260-274, Jun 2008.
- [4] Peter Lancaster and Miron Tismenetsky. *The Theory of Matrices*, Second Edition with Applications. 11(3): 390. Academic Press Inc, 1985.
- [5] R. J. Forrester and H. J. Greenberg. Quadratic binary programming models in computational biology. *Algorithmic Operations Research*, 3:110-129, 2008.
- [6] S.L. Campbell and C.D. Meyer, Jr. *Generalized Inverses of Linear Transformations*. 10(4): 218. Dover Publications, Inc., New York, 1979.
- [7] Yosi Shibberu, Allen Holder, and Kyla Lutz. Fast Protein Structure Alignment. *M. Borodovsky, ed. Springer-Verlag, ISBRA 2010. LNCS (LNBI)*, vol. 6053: 152-165, 2010.