REPORTS ON COMPUTATIONAL MATHEMATICS, ANL/MCS-P1542-0908
MATHEMATICS AND COMPUTER SCIENCE DIVISION, ARGONNE NATIONAL LABORATORY

Mathematical Programming manuscript No.
(will be inserted by the editor)

Mihai Anitescu · Sanghyun Park

Short Communication: A Linear Programming Approach for the Least-Squares Protein Morphing Problem

the date of receipt and acceptance should be inserted later

Abstract This work addresses the computation of free-energy differences between protein conformations by using morphing (i.e., transformation) of a source conformation into a target conformation. To enhance the morphing procedure, we employ permutations of atoms; we transform atom n in the source conformation into atom $\sigma(n)$ in the target conformation rather than directly transforming atom n into atom n. Because shorter morphing paths reduce the cost of the free-energy computation, we seek to find the permutation σ that minimizes the mean-square distance traveled by the atoms. Instead of performing this combinatorial search in the space of permutations, we relax the search onto the space of bistochastic matrices and solve the relaxed problem by linear programming. Using Birkhoff's theorem, we show that the solution of the relaxed problem is indeed identical to the solution of the original problem. We demonstrate that the use of such optimal permutations significantly improves the efficiency of the free-energy computation.

Keywords Linear programming, Birkhoff's theorem, protein conformations, free-energy calculation

Mathematics Subject Classification (2000) 65K10, 90C33

1 Introduction

Many important phenomena in molecular biology involve conformational changes of proteins. For quantitative understanding of these phenomena, it is imperative to know the free-energy changes associated with the conformational changes [4,5]. Conformational free energy is also an essential notion

Mathematics and Computer Science Division, Building 221, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439, U.S.A. email: anitescu@mcs.anl.gov

in predicting protein structures from their sequences, as conformational free-energy differences provide the relative stability of different protein structures [1,9]. This paper addresses the computation of conformational free-energy differences through least-squares protein morphing. The key challenge of this technique is to determine a permutation of the labels of atoms such that the sum of distances between atoms of a given structure and their targets is minimized.

Optimization problems over the set of all permutations, such as discussed here, are combinatorial in nature. These problems often belong to the NP-complete complexity class and may be difficult to solve even for small values of N [12]. An exhaustive brute search would result in $N!$ steps and would be impossible to carry out even for N in the low tens.

This paper shows that an optimal permutation can be determined as a solution of a linear program with N^2 variables and $2N$ constraints, which results in the least-squares morphing problem having a much lower complexity and a practical large-scale solution. The key element of the proof is the use of Birkhoff's theorem in a fashion reminiscent of its application for linear assignment problems [11,8]. Note, however, that our problem is radically different, both in the objective function and in its interpretation, from the linear assignment problem.

We demonstrate the power of least-squares morphing over direct morphing by applying it to the computation of conformational free energy of deca-alanine. We also demonstrate that our linear programming approach is a practical method for determining the solution to the least-squares protein morphing problem.

2 Protein Morphing

We briefly describe the basics of free-energy calculations. For further details, the reader can consult [6].

Consider a system, composed of a protein surrounded by solvent such as water, described by a potential energy function $U(\mathbf{X}, \mathbf{Y})$, where $\mathbf{X} := (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ and $\mathbf{Y} := (\mathbf{y}_1, \ldots, \mathbf{y}_{N'})$ denote the atomic coordinates of the protein and the solvent, respectively. For a protein conformation $\mathbf{X}$, the conformational free energy $F(\mathbf{X})$ is defined by

$$e^{-\beta F(\mathbf{X})} = \int d\mathbf{Y}\, e^{-\beta U(\mathbf{X},\mathbf{Y})} \;, \tag{1}$$

where $\beta := 1/k_{\mathrm{B}}T$, k_{B} is the Boltzmann constant, and T is the temperature.[1] Given two different conformations, $\mathbf{A} := (\mathbf{a}_1, \ldots, \mathbf{a}_N)$ and $\mathbf{B} := (\mathbf{b}_1, \ldots, \mathbf{b}_N)$, the objective is to find the free-energy difference

$$\Delta F := F(\mathbf{B}) - F(\mathbf{A}) \;. \tag{2}$$

[1] In general, conformational states of proteins are more adequately defined as ensembles of conformations rather than single conformations. In such cases, the free-energy difference between two conformational states should include the relative free-energy cost for transforming each conformational ensemble into a respective reference conformation [13].

Direct computations of the integral in Eq. 1 are impractical because of the high dimensionality of the problem. One practical approach for calculating ΔF is to perform free-energy perturbation [18, 2] along a transformation (i.e., morphing) path between the two conformations. In previous work [13], the following morphing path, parametrized with λ, was employed for such a purpose:

$$\mathbf{x}_n(\lambda) = (1-\lambda)\,\mathbf{a}_n + \lambda\,\mathbf{b}_n\ , \quad 0 \le \lambda \le 1\ , \tag{3}$$

in which the nth atom is transformed from its position $\mathbf{a}_n$ in conformation $\mathbf{A}$ into its corresponding position $\mathbf{b}_n$ in conformation $\mathbf{B}$.

In this work, we enhance the morphing procedure by incorporating permutations of atoms. Instead of transforming the nth atom onto itself, we transform it onto the $\sigma(n)$th atom:

$$\mathbf{x}_n(\lambda) = (1-\lambda)\,\mathbf{a}_n + \lambda\,\mathbf{b}_{\sigma(n)}\ , \quad 0 \le \lambda \le 1\ , \tag{4}$$

where σ is a permutation. Since the farther the atoms travel, the more costly is the morphing path for the free-energy computation, we seek to find the permutation σ that minimizes the mean-square distance traveled by the atoms,

$$\frac{1}{N}\sum_{n=1}^{N} ||\mathbf{a}_n - \mathbf{b}_{\sigma(n)}||^2\ , \tag{5}$$

where $||\cdots||$ is the Euclidean norm.

In this work, we investigate the problem of determining the permutation that minimizes this sum. We call this problem the least-squares protein morphing problem. In the next section, we show that this combinatorial problem can essentially be solved in polynomial time by linear programming.

3 Linear Programming Formulation

The least-squares morphing problem determines the permutation of the labels of the target points that minimizes the mean-square distance between the current position and the target position. It can be expressed formally as follows.

Problem Π: Given $\mathbf{A} := (\mathbf{a}_1, \ldots, \mathbf{a}_N) \in \mathbb{R}^{Np}$ and $\mathbf{B} := (\mathbf{b}_1, \ldots, \mathbf{b}_N) \in \mathbb{R}^{Np}$, solve

$$\min_{\sigma \in \Pi_N} \sum_{i=1}^{N} ||\mathbf{a}_i - \mathbf{b}_{\sigma(i)}||^2. \tag{6}$$

Here, Π_N is the set of all permutations with N elements, and $\mathbf{a}_i$ and $\mathbf{b}_i$ ($i = 1, 2, \ldots, N$) are elements of $\mathbb{R}^p$. The definition of the Euclidean norm in terms of inner products results in the relationship

$$||\mathbf{x}-\mathbf{y}||^2 = \langle \mathbf{x}, \mathbf{x}\rangle + \langle \mathbf{y}, \mathbf{y}\rangle - 2\,\langle \mathbf{x}, \mathbf{y}\rangle\ , \quad \mathbf{x}, \mathbf{y} \in \mathbb{R}^p.$$

Recall that the definition of the inner product is $\langle \mathbf{x}, \mathbf{y}\rangle = \sum_{k=1}^{p} x^k y^k$. Here and in the following, we denote the entries of a vector $\mathbf{x} \in \mathbb{R}^p$ by $x^k \in \mathbb{R}$, $k = 1, 2, \ldots, p$.

We now define the permutation matrix $\mathbb{P}(\sigma) \in \mathbb{R}^{N\times N}$ to be the matrix whose entries $\{p_{ij}(\sigma)\}_{i,j=1,2,\ldots,N}$ satisfy

$$p_{ij}(\sigma) = \begin{cases} 1, & j = \sigma(i) \\ 0, & \text{otherwise} \end{cases}. \tag{7}$$

The permutation matrix thus operates as

$$\mathbb{P}(\sigma)\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} x_{\sigma(1)} \\ x_{\sigma(2)} \\ \vdots \\ x_{\sigma(N)} \end{pmatrix}. \tag{8}$$

We define the matrices $\mathbb{A}, \mathbb{B} \in \mathbb{R}^{p\times N}$:

$$\mathbb{A} = \begin{pmatrix} a_1^1 & a_2^1 & \ldots & a_N^1 \\ a_1^2 & a_2^2 & \ldots & a_N^2 \\ \vdots & \vdots & \vdots & \vdots \\ a_1^p & a_2^p & \ldots & a_N^p \end{pmatrix}, \quad \mathbb{B} = \begin{pmatrix} b_1^1 & b_2^1 & \ldots & b_N^1 \\ b_1^2 & b_2^2 & \ldots & b_N^2 \\ \vdots & \vdots & \vdots & \vdots \\ b_1^p & b_2^p & \ldots & b_N^p \end{pmatrix}. \tag{9}$$

The definition of these matrices allows us to reformulate Problem Π in terms of permutation matrices.

Lemma 1 *Let the permutation σ^* be a solution of the optimization problem*

$$\max_{\sigma\in\Pi_n} \text{trace}\left[\mathbb{A}\mathbb{P}(\sigma)\mathbb{B}^T\right]. \tag{10}$$

Then, σ^* is a solution of Problem Π. Here the trace *operator returns the sum of the diagonal entries of a square matrix.*

Proof First, we rewrite the objective function of the Problem Π in terms of inner products:

$$\sum_{i=1}^{N} ||\mathbf{a}_i - \mathbf{b}_{\sigma(i)}||^2 = \sum_{i=1}^{N}\left[\langle \mathbf{a}_i, \mathbf{a}_i\rangle + \langle \mathbf{b}_{\sigma(i)}, \mathbf{b}_{\sigma(i)}\rangle - 2\langle \mathbf{a}_i, \mathbf{b}_{\sigma(i)}\rangle\right]. \tag{11}$$

On the other hand, we have

$$\text{trace}\left[\mathbb{A}\mathbb{P}(\sigma)\mathbb{B}^T\right] \overset{(8)}{=} \sum_{k=1}^{p}\sum_{i=1}^{N} a_i^k b_{\sigma(i)}^k = \sum_{i=1}^{N}\langle \mathbf{a}_i, \mathbf{b}_{\sigma(i)}\rangle. \tag{12}$$

From (12) and (11), it follows that $\sigma \in \Pi_n$ minimizes $\sum_{i=1}^{N} ||\mathbf{a}_i - \mathbf{b}_{\sigma(i)}||^2$ if and only if it maximizes trace $\left[\mathbb{A}\mathbb{P}(\sigma)\mathbb{B}^T\right]$, since the term $\sum_{i=1}^{N}\langle \mathbf{b}_{\sigma(i)}, \mathbf{b}_{\sigma(i)}\rangle$ in (11) does not depend on σ. The proof is complete. □

We now define two convex sets that are important in the proof of Theorem 1.

$$\mathcal{F}_N = \left\{ w_{ij} \geq 0,\ i,j = 1,2,\ldots,N \;\middle|\; \begin{array}{l} \sum_{j=1}^{N} w_{ij} = 1,\ i = 1,2,\ldots,N; \\ \sum_{i=1}^{N} w_{ij} = 1,\ j = 1,2,\ldots,N \end{array} \right\}$$

$$\mathcal{S}_N = \left\{ \mathbb{W} \in \mathbb{R}^{N\times N} \;\middle|\; \{w_{ij}\}_{i,j=1,2,\ldots,N} \in \mathcal{F}_N \right\}.$$

A matrix is an element of $\mathcal{S}_N$ if and only if its entries belong to the set $\mathcal{F}_N$. Such matrices are called *bistochastic* matrices. They have nonnegative entries that add up to 1 on both rows and columns. Note that permutation matrices are bistochastic matrices. The relationship between bistochastic matrices and permutation matrices is further clarified by Birkhoff's theorem. Recall that a vertex of a closed convex set is a point that cannot be expressed as a nontrivial convex combination of two distinct points in the set [10].

Birkhoff's theorem [3,11] A matrix is a vertex of the set of bistochastic matrices $\mathcal{S}_N$ if and only if it is a permutation matrix.

We are now ready to state our main result.

Theorem 1 *Consider the linear programming problem,*

$$\max_{\{w_{ij}\}_{i,j=1,\ldots,N} \in \mathcal{F}_N} \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} \langle \mathbf{a}_i, \mathbf{b}_j \rangle \ . \tag{13}$$

The problem has a solution w_{ij}^*, $i, j = 1, 2, \ldots, N$, which represents the entries of a permutation matrix, $\mathbb{P}(\sigma^*)$. The permutation σ^* is a solution of the least-squares morphing problem, Problem Π.

Proof The problem stated in the theorem can be considered a linear assignment problem with costs $\langle \mathbf{a}_i, \mathbf{b}_j \rangle$. That problem is known to have a permutation matrix solution [8], which would address the first part of the statement. For completeness and for the computational biology audience, we include the proof of that fact, in the context of this work.

The entries of any bistochastic matrix are feasible for the linear program (13). From the definition of $\mathcal{F}_N$, it follows that $0 \leq w_{ij} \leq 1$. Therefore the feasible set is also bounded. Following the Fundamental Theorem of Linear Programming [10], the linear program (13) has a solution $\left\{w_{ij}^*\right\}_{i,j=1,2,\ldots,N}$ that is a vertex of its feasible set $\mathcal{F}_N$. The mapping $\mathbb{W} \to \{w_{ij}\}_{i,j=1,2,\ldots,N}$, which maps a bistochastic matrix to the set of its entries, is a linear isomorphism between $\mathcal{F}_N$ and $\mathcal{S}_N$. Therefore the solution $\left\{w_{ij}^*\right\}_{i,j=1,2,\ldots,N}$, a vertex of $\mathcal{F}_N$, is the image of a vertex of the set of bistochastic matrices. From Birkhoff's theorem, there exists a permutation σ^* such that that vertex of $\mathcal{S}_N$ is the permutation matrix $\mathbb{P}(\sigma^*)$ whose entries are $\left\{w_{ij}^*\right\}_{i,j=1,2,\ldots,N}$.

For the second part of the proof, denote by $\mathbb{W}$ the bistochastic matrix with entries $\{w_{ij}\}_{i,j=1,2,\ldots,N}$. It is immediate that the objective function of the problem (13) can be written as trace $\left(\mathbb{A}\mathbb{W}\mathbb{B}^T\right)$. Therefore the linear program (13) is equivalent to the problem

$$\max_{\mathbb{W} \in \mathcal{S}_N} \text{trace} \left(\mathbb{A}\mathbb{W}\mathbb{B}^T\right) \ .$$

A solution of this problem is $\mathbb{P}(\sigma^*)$. Since the set $\mathcal{S}_N$ contains all permutation matrices, the permutation matrix $\mathbb{P}(\sigma^*)$ is a solution of the same problem restricted over the set of permutation matrices,

$$\max_{\mathbb{W} \in \Pi_N} \text{trace} \left(\mathbb{A}\mathbb{W}\mathbb{B}^T\right),$$

which is precisely problem (10). From Lemma 1, the permutation σ^* solves Problem Π. The proof is complete. □

Linear Programming problems have polynomial complexity when solved with interior-point algorithms [17]. On the other hand, Theorem 1 requires an optimal vertex, not merely an optimal solution, which is what the interior-point algorithms are guaranteed to find in polynomial time. Although a vertex solution of (13) can be found by the simplex algorithm [10], the simplex algorithms can take an exponentially large number of steps [12].

We point out, however, that primal nondegeneracy, the existence of unique primal solutions, is a generic property of linear programs [14]. This means that the set of problems for which the solution returned by the interior-point algorithm is not unique is, in some sense, of measure zero. In that case, from Theorem 1, the unique solution will be precisely the required permutation matrix. While this does not completely settle the issue of complexity, it does indicate that the polynomial complexity of interior-point methods can be expected.

Of course, the linear program (13) that we solve in order to determine our permutation is in a linear programming class narrower than the one for which the genericity result [14] holds. Therefore, using such a result as an argument for expecting low complexity, while encouraging, must be considered with some skepticism before actual numerical demonstrations are provided. We point out, however, that this is probably true of any instance of linear programmng applications. In addition, we provide numerical demonstrations of the effectiveness of the method based on (13) in Section 4.

4 Numerical Demonstration of Least-Squares Morphing for Free-Energy Calculations

In this section we compare the efficiencies of the free-energy computation using the direct and the least-squares morphing schemes. We also analyze the performance of the linear programming method for minimizing the mean-square distance.

Note that the the linear programming approach posited by Theorem 1 is an essential part of a new method of computing conformational free energy of proteins. Since no other free-energy method we are aware of uses least-squares morphing, our linear programming approach does not compete with any existing baseline method for solving Problem Π.

Therefore our performance assessments will focus on two issues. First, we will assess whether the linear programming approach is sufficiently fast to determine the least-squares permutation so as not to slow the rest of the calculation. In particular, since we use an interior-point method, we do not reach the solution permutation matrix in a finite number of steps. In addition, given the possible degeneracy risk outlined at the end of Section 3, the solution returned by the interior-point algorithm may be a point other than a vertex of the feasible set. We thus wish to determine whether the method converges sufficiently fast to a point from which the optimal permutation σ^* can be unambiguously determined. The second issue we will

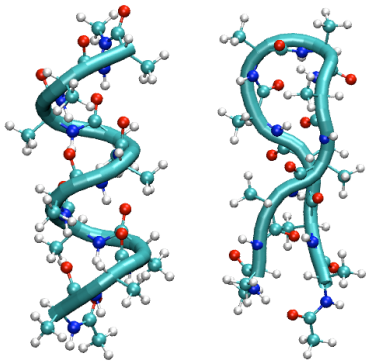

Fig. 1 Two conformations of deca-alanine: helix and hairpin. The traces of backbone are shown as tubes.

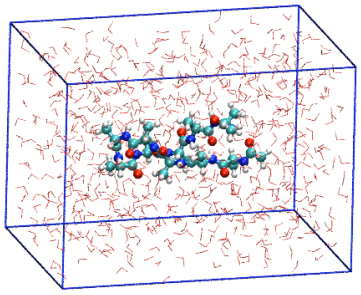

Fig. 2 Simulation box containing a deca-alanine molecule submerged in water.

assess is whether least-squares morphing is superior to direct morphing for free-energy computations.

4.1 The Free Energy Calculation Example

We use as an example deca-alanine [7]. Deca-alanine is a polypeptide chain composed of ten alanine residues. It is often used as a model system in computational studies of proteins. We selected two different conformations of deca-alanine, helix (conformation **A**) and hairpin (conformation **B**) as shown in Fig. 1, and aligned them such that their centers of mass and principal axes coincide, respectively. Molecular dynamics simulations were then performed with deca-alanine in a box of water (Fig. 2) along a direct morphing path and along a least-squares morphing path between the two conformations, and free-energy differences were computed by free-energy perturbation (FEP). We used the same simulation parameters as in Ref. [13], where further details of the molecular dynamics simulations can be found.

While direct morphing is entirely described in [13], our implementation of least-squares morphing requires additional detail. A least-squares morphing

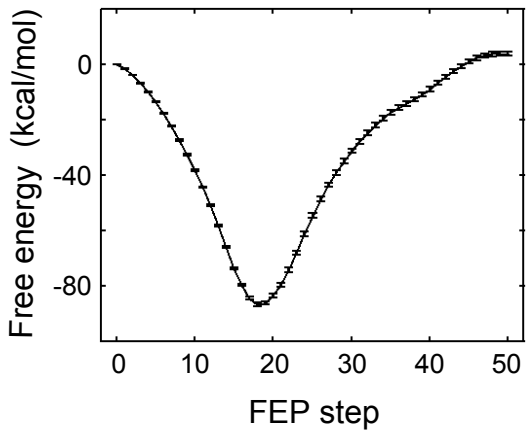

Fig. 3 The free-energy profile computed by the direct morphing scheme. 50 FEP steps were used along the morphing path. The total free-energy difference is estimated to be $\Delta F = 3.6 \pm 0.69\,\mathrm{kcal/mol}$.

path, which involves a permutation of the protein atoms, is not a complete transformation of conformation **A** into **B**, because of the chemical diversity of the atoms. To be specific, the protein atoms have different charges, radii, and so forth. Thus, for each conformation, we performed an alchemical FEP [16] in which the protein atoms were converted into chemically identical dummy atoms. Dummying at **A** followed by a least-squares morphing from **A** to **B** and then inverse-dummying at **B** constitutes a complete transformation of conformation **A** into **B**.

In addition, in designing the morphing procedure, we grouped the protein atoms into two sets, hydrogens and heavy atoms, and applied the dummying and the least-square morphing procedures separately for each set. We find that this separation significantly reduces the cost of the dummying steps without much sacrifice in the morphing step. But the significance for linear programming is that *two* instances of Problem Π need to be resolved by linear programming: one for the heavy atoms and one for the hydrogen atoms. (In the case of deca-alanine, we had 57 hydrogen atoms and 55 heavy atoms.) All these choices include some amount of trial and error and heuristic, typical of the development of a new technique. Further study needs to be done to find out which separation scheme is the most efficient in general. Note, however, that the solving of the optimal permutation problem itself is on solid theoretical ground provided by Theorem 1.

4.2 Numerical Results

Linear Programming Performance. We used MATLAB for the linear programming calculation. The matrices obtained by the linear programming resolution of (13) and the use of Theorem 1 were unmistakably permutation matrices, with their elements close to either zero or one within 10^{-7}.

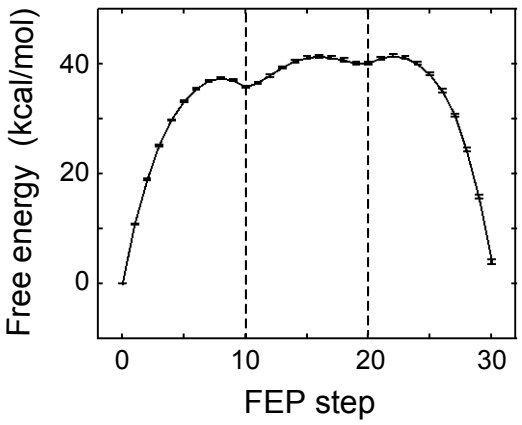

Fig. 4 The free-energy profile computed by the least-square morphing scheme. FEP steps 1 to 10 correspond to the dummying at the helix conformation, 11 to 20 the least-square morphing path, and 21 to 30 the inverse-dummying at the hairpin conformation. The total free-energy difference is estimated to be $\Delta F = 3.8 \pm 0.44\,\mathrm{kcal/mol}$.

The computational cost of the linear programming calculation is trivial compared to that of FEP; each linear programming calculation took a few wallclock seconds using the interior-point `linprog` solver in MATLAB with the default tolerance parameters. By comparison, each FEP step, which uses C-compiled software NAMD [15], took 20 wallclock hours on a machine with a comparable processor and memory configuration. Therefore we find that the linear programming approach is an efficient method to design optimal procedures for free-energy calculations.

Least-Squares Morphing Versus Direct Morphing. Along the direct morphing path constructed by Eq. 3, the atoms must travel quite large distances (the root-mean-square distance amounts to 8.4 Å), and we used 50 FEP steps for the free-energy computation. Figure 3 shows the free-energy profile computed along the direct path. The total free-energy difference is estimated to be $\Delta F = F(\mathbf{B}) - F(\mathbf{A}) = 3.6 \pm 0.69\,\mathrm{kcal/mol}$, where the uncertainty was obtained by block averaging as in Ref. [13].

Shown in Fig. 4 is the free-energy profile along the entire transformation path consisting of the dummying step at the helix conformation, the least-squares morphing, and the inverse-dummying step at the hairpin conformation. The total free-energy difference estimated by this scheme is $\Delta F = 3.8 \pm 0.44\,\mathrm{kcal/mol}$. Compared to the direct morphing path, the atoms move much shorter distances along the least-squares path (the root-mean-square distance is 2.1 Å), and we were able to use only 10 FEP steps for the morphing. Even if we include the 20 FEP steps that we used for the two dummying procedures (10 for each), it is still much lower than the 50 steps needed in the direct morphing scheme. The number of steps needed to reduce the uncertainty of the direct morphing approach to the level of least-squares morphing

approach would be higher than 50, making the latter even more convincingly a winner.

With the least-squares morphing scheme we were able to achieve a smaller numerical uncertainty with fewer FEP steps, that is, with lower computational cost. In this test case, the least-squares morphing scheme proves to be significantly more efficient than the direct scheme. The saving in computing time amounts to hundreds of hours because of the very expensive FEP steps.

5 Conclusions and Future Work

Morphing, the continuous virtual transformation of a protein conformation into another, is an important aid in efficient calculations of conformational free-energy differences, which are essential for the understanding of the function of proteins. We introduce a particular type of morphing, which we call least-squares morphing, that has the potential to reduce the lengths of the paths traveled by atoms between the two conformations. When this reduction occurs, the number of very costly free-energy perturbation (FEP) steps needed to complete such a calculation radically decreases. Moreover, the use of least-squares morphing mostly eliminates the possibility of trapping of water inside protein, which could be disastrous for free-energy computations [13].

Least-squares morphing searches for the permutation of the labels of the atoms in the target conformation that minimizes the mean-square distance traveled. We show, by using Birkhoff's theorem, that this permutation can be obtained by solving a linear program with N^2 nonnegative unknowns with $2N$ constraints, where N is the number of atoms involved. We demonstrate, using calculations for deca-alanine, that the cost of the linear program is insignificant compared to the cost of an FEP step. In addition, least-squares morphing results in substantially fewer FEP steps and better free-energy uncertainty compared to direct morphing.

Future research will be devoted to the application of least-squares morphing to complex conformational changes of proteins, which typically involve hundreds or thousands of atoms. We will also investigate the relationship between different parametric choices, for example, between the number of FEP steps and the uncertainty in free-energy differences estimated.

Acknowledgments

Sanghyun Park thanks Peter Freddolino for many insights and discussions. Mihai Anitescu and Sanghyun Park have been supported by the Department of Energy through contract DE-AC02-06CH11357.

References

1. Baker, D., Sali, A.: Protein structure prediction and structural genomics. Science **294**(5540), 93–96 (2001)

2. Bennett, C.H.: Efficient estimation of free energy differences from Monte Carlo data. J. Comp. Phys. **22**, 245–268 (1976)
3. Birkhoff, G.: Tres observaciones sobre el algebra lineal. Univ. Nac. Tucuman Rev. Ser. A **5**, 147–151 (1946)
4. Dobson, C.: Protein folding and misfolding. Nature **426**(6968), 884–890 (2003)
5. Frederick, K., Marlow, M., Valentine, K., Wand, A.: Conformational entropy in molecular recognition by proteins. Nature **448**, 325–329 (2007)
6. Frenkel, D., Smit, B.: Understanding Molecular Simulation: From Algorithms to Applications. Academic Press, New York (2002)
7. Kitao, A., Go, N.: Conformational dynamics of polypeptides and proteins in the dihedral angle space and in the Cartesian coordinate space: Normal mode analysis of deca-alanine. J. Comp. Chem **12**, 359–368 (1991)
8. Koopmans, T., Beckmann, M.: Assignment problems and the location of economic activities. Econometrica **25**(1), 53–76 (1957)
9. Lee, M., Tsai, J., Baker, D., Kollman, P.: Molecular dynamics in the endgame of protein structure prediction. Journal of Molecular Biology **313**(2), 417–430 (2001)
10. Luenberger, D.: Linear and Nonlinear Programming. Springer, Berlin (2003)
11. von Neumann, J.: A certain zero-sum two-person game equivalent to the optimal assignment problem. In: H. Kuhn, A.W. Tucker (eds.) Contributions to the Theory of Games, vol. 2, pp. 5–12. Princeton University Press (1953)
12. Papadimitriou, C., Steiglitz, K.: Combinatorial Optimization: Algorithms and Complexity. Dover Publications, Mineola, New York (1998)
13. Park, S., Lau, A.Y., Roux, B.: Computing conformational free energy by deactivated morphing. J. Chem. Phys. (in press)
14. Pataki, G., Tunçel, L.: On the generic properties of convex optimization problems in conic form. Mathematical Programming **89**(3), 449–457 (2001)
15. Phillips, J.C., Braun, R., Wang, W., Gumbart, J., Tajkhorshid, E., Villa, E., Chipot, C., Skeel, R.D., Kale, L., Schulten, K.: Scalable molecular dynamics with NAMD. J. Comput. Chem. **26**, 1781–1802 (2005)
16. Straatsma, T., McCammon, J.: Computational alchemy. Annual Reviews in Physical Chemistry **43**(1), 407–435 (1992)
17. Wright, S.: Primal-Dual Interior-Point Methods. Society for Industrial Mathematics, Philadelphia (1997)
18. Zwanzig, R.W.: High-temperature equation of state by a perturbation method. J. Chem. Phys. **22**, 1420–1426 (1954)