2014 13th International Conference on Machine Learning and Applications

Protein Conformation Motion Modeling using sep-CMA-ES

Maxim Buzdalov, Sergey Knyazev, Yury Porozov
ITMO University
49 Kronverkskiy prosp.
Saint-Petersburg, Russia, 197101
Email: {mbuzdalov, srknyazev}@gmail.com, porozov@ifc.cnr.it

***Abstract*—The problem of protein conformation motion modeling is an open problem in the structural computational biology. It is difficult to solve it using methods of molecular dynamics or quantum physics because these methods deal with time intervals of nanoseconds or microseconds, while conformation motions take time of millisecond order. In addition, these methods cannot take external forces into consideration. To deal with these problems, numerous approximated and coarse-grained methods are developed, which use ideas from geometry and motion planning.**

We present a new coarse-grained method of modeling the protein motion between two given conformations. The method is based on optimization of a cost function similar to the one in the Monge-Kantorovich mass transfer problem. The optimization is performed using sep-CMA-ES, which makes the running time of an iteration linear in the number of amino acids in a protein.

The proposed method is compared with some of the existing methods on several molecules. It is shown that the results of the proposed method are more accurate than of the other methods.

I. Introduction

Structural biology of biopolymers has made substantial progress on the way to understanding the spatial structure of proteins, cellular localization, and predicting their functions and interactions with other proteins and small molecules. The development of such basic instruments of structural biology as X-ray crystallography and nuclear magnetic resonance and the exponential growth in the number of recognized protein structures accumulated in the Protein Data Bank [1] has led to new methods of mathematical modeling for both the three-dimensional structures themselves and their specific properties, in particular, conformation motion. The ability to change conformations is essential for proteins. Studying protein molecule dynamics in time can help to answer questions regarding the order in which protein conformations follow each other and regarding molecular motion trajectories across stable states. It is known that many protein functions are actually implemented in motion [2]. Obviously, this is due to the fact that during such behavior different active centers (hot spots) may become exposed at the molecular surface. If we suppose that a protein molecule has several active centers responsible for interaction with different substances, then modeling the motion of these proteins may give us the key to predicting their functions [2]. Moreover plausible trajectories of protein between static conformations can serve as an input for modern techniques of flexible docking and virtual screening in modern pharmacology. Functional properties of such proteins with hidden or temporarily closed active centers may remain unclear if the structures are static and only show up in conformation motion modeling. This is extremely important for both theoretical metabolomic and signaling studies and applied drug design, as a way to predict, for instance, side effects of new active substances. This, in turn, may help us understand how proteins behave and search for the regulators of their functions.

There are several ways for prediction of protein trajectories. The most precise ones are molecular dynamics techniques [3]–[7]. However, these methods have restricted use because of high computational complexity (leading to very long simulation times even on modern supercomputers) and the low probability of escaping from an energy region close to a stable state. These drawbacks make it virtually infeasible to model conformation motion of protein molecules (especially large proteins) by protocols of molecular dynamics.

The second group of methods for proteins trajectory prediction are based on geometry analysis [8], [9]. Their advantages include a relatively low computational cost (i.e., high computation speed) and a possibility to overcome energetic barriers bounding a region close to the stable state. But at the same time implementation of these techniques can violate protein geometry. Later works [10] construct more complicated models of a conformation motion and use optimization methods to minimize cost functions. We also consider the elastic network models [11]–[14] to belong to this group.

The third group of methods originates from motion planning. These include probabilistic roadmaps [15], [16], rapidly exploring random trees [17], [18], and stochastic roadmap simulation [19]. For example, in a probabilistic roadmap, individual conformations are connected to form a graph in which well-known shortest path algorithms can be applied to approximate the optimal transition. However, to appropriately weigh the edges of this graph one needs to have some way to estimate the transition cost between conformations, albeit on a smaller scale, that is, methods from the first two groups may still be needed.

In this paper we introduce an approach for prediction and construction of plausible conformation trajectory in proteins. This technique is based on the same mass transportation problem as in [10], but uses the sep-CMA-ES algorithm [20]

978-1-4799-7415-3/14 $31.00 © 2014 IEEE
DOI 10.1109/ICMLA.2014.12

for its optimization. We describe our methods and show its efficiency on the example of calmodulin transition modeling between a compact (PDB ID: 1PRW [21]) and an open (PDB ID: 1OSA [22]) state, as well as on the example of transitions between all pairs of conformations of the 2M3M protein [23].

II. Conformation Motion and Cost Evaluation

Proteins are biopolymers constructed from multiple amino acid residues. In proteins, one considers the *backbone*, which consists of the atoms N^i, C^i_α, C^i, N_{i+1}, C^{i+1}_α, C^{i+1}, ... connected by covalent bonds to make a chain, and *sidechains*, which are unique to each amino acid. In this work, we ignore the sidechains, and their mass is added to the corresponding atoms from the backbone. This is motivated by the fact that changes in sidechains require much less energy than changes in the backbone.

The lengths of the bonds between the atoms in the backbone are nearly constant during all the conformation motion. The planar angles formed by consecutive bonds (namely, N^i–C^i_α–C^i, C^i_α–C^i–N^{i+1} and C^i–N^{i+1}–C^{i+1}_α) are also almost constant. The torsion angles between bonds C^i_α–C^i and N^{i+1}–C^{i+1}_α is always equal to π. However, the torsion angles between N^i–$C\alpha^i$ and C^i–N^{i+1}, and between C^i–N^{i+1} and C^{i+1}_α–C^{i+1} can change. This makes transformations between conformations possible, and the conformation itself can be defined, up to translations and rotations, by the values of the variable torsion angles.

Assume the protein has N amino acids. Then there are $3N$ atoms in the backbone and $3N-3$ torsion angles, out of which $2N-2$ are variable. A conformation can be determined by a vector ω of $2N-2$ values of torsion angles in the range of $(-\pi;\pi]$. More specifically, Cartesian coordinates of the atoms can be restored from the torsion angles using the method from [24]. However, this conformation can be arbitrarily rotated and translated in space.

A. Conformation Motion and its Cost

A conformation motion is modelled in this paper as a function from time to the Cartesian coordinates of the form:

$$M(t) = X(TC(\omega(t)), T(t), R(t)),$$

where $\omega(t)$ is the function from time to the torsion angle values, $TC(\omega)$ is the function that restores Cartesian coordinates, $T(t)$ is the function from time to translation vector, $R(t)$ is the function from time to rotation matrix, $X(c,t,r)$ applies the transition vector t and the rotation matrix r to the sequence of atom coordinates c. There are two constraints: $M(0)$ is the initial conformation X_0, and $M(1)$ can be achieved from the final conformation X_1 by translation and rotation only.

The *cost* of a motion is similar to the function from the Monge-Kantorovich mass transportation problem [25]:

$$C(M) = \sum_{i=1}^{3N} m_i \cdot {l_i}^p,$$

where N is the number of amino acids (so the number of atoms in the backbone is $3N$), m_i is the *associated mass* of the i-th atom (it consists of the mass of the atom and masses of all the connected atoms from the corresponding sidechain), l_i is the length of a path that is made by the i-th atom during the conformation motion M, and p is a parameter, which is typically equal to 1 or 2.

B. Discrete Version

We need to discretize the definition of the conformation motion, as well as its cost, to allow its modeling and evaluation in finite time. To do this, we define the number of *intermediate conformations* K and give the definition of the *discretized conformation motion* as follows:

$$M_j = X(TC(\omega_j), T_j, R_j),$$

where j ($0 \le j \le K+1$) is the integer index of a discrete moment of time, and the definitions of ω_j, T_j and R_j are the same as the definitions of $TC(t)$, $T(t)$, $R(t)$ above, except that they are now discrete. Again, M_0 is the initial conformation X_0, and M_{K+1} can be achieved from the final conformation X_1 by translation and rotation only.

The cost can be discretized as follows:

$$C(M) = \sum_{i=1}^{3N} m_i \left(\sum_{j=0}^{K} l_i^{j,j+1} \right)^p, \tag{1}$$

where $l_i^{j,j+1}$ is the distance travelled by atom between the discrete moments of time j and $j+1$, which is approximated as the Euclidean distance between the atom locations at these moments.

However, as we do not impose additional restrictions on $l_i^{j,j+1}$, they can be arbitrary. In fact, when $p \ge 1$, the minimum is reached if $l_i^{0,1}$ is the distance between the i-th atom in X_0 and the i-th atom in X_1 and $l_i^{j,j+1} = 0$ if $j > 0$. This, in turn, corresponds to the unrealistic motion when the protein makes all the movement during the first discrete time step and does not move during all other time steps.

Instead, we will optimize the following cost function:

$$C(M) = \sum_{i=1}^{3N} m_i \sum_{j=0}^{K} \left(l_i^{j,j+1} \right)^p. \tag{2}$$

There are two reasons to select this modification of expression. First, it can be shown that, when K goes to infinity, the motions that deliver the minimums to the expressions (1) and (2) coincide. Second, as shown below, there exist efficient algorithms which align the structures while minimizing (2), but the authors are unaware of similarly efficient algorithms that do the same for (1). We can write the expression (2) as:

$$C(M) = \sum_{j=0}^{K} D_j; D_j = \sum_{i=1}^{3N} m_i \left(l_i^{j,j+1} \right)^p,$$

where D_j is effectively the cost of transition from the conformation at step j to the conformation at step $j+1$. If all ω_j are fixed, then we can minimize D_j separately by appropriately aligning the pairs of consecutive conformations, so we compute the minimal possible cost for arbitrary fixed ω_j.

This makes it possible to optimize the cost by changing only the values of ω_j and computing the translation and rotation vectors exactly.

If $p = 2$, then minimizing D_j turns to minimizing the square of *weighed* RMSD (i-th atom has a weight of m_i), multiplied by the number of atoms. We can do it effectively using the Kabsch algorithm [26]. In the rest of the article, $p = 2$ is used.

C. Atom and Bond Collisions

There is another source of violations of physical properties of conformation motions in the models — in real life, atoms cannot collide and appear too close to each other, and the bonds between the atoms, when treated as sticks between the atoms, cannot intersect each other. It is possible to compute the number of collisions in a motion between two conformations, at least approximately, but this takes $O(N^2)$ time, while all other steps mentioned above (restoring the Cartesian coordinates, aligning the consecutive conformations and computing the weighed RMSD) take only $O(N)$ time. In this paper we do not consider collisions when optimizing conformation motions. However, to compare different methods, we check and report whether the produced motions contain collisions.

III. SEP-CMA-ES

As described in Section II-B, we can evaluate the cost function using only the values ω_j — the vectors of the values of torsion angles in the discrete moments of time. To optimize the cost function, one can use any optimization algorithm. In [10], the method of conjugate gradients was used. However, we do not know the properties of the cost function — for example, we cannot guarantee that it is unimodal — so gradient-based methods can converge to some non-global optima. To deal with this problem, we use the *evolution strategy with covariance matrix adaptation* (CMA-ES) — an evolutionary algorithm for global optimization [27].

However, the size of the conformation motion optimization problem is typically large — for example, the average number of amino acids in yeast proteins is 466, and titins can reach 27 000 amino acids [28]. This discourages the use of CMA-ES in its original version [27], which requires computing eigenvectors of a matrix with the size of $O(KN) \times O(KN)$, in total $O(K^3N^3)$ time. Instead, we use a modification which preserves only the diagonal of the covariance matrix, called sep-CMA-ES [20], which makes the complexity of all matrix updates equal to $O(KN)$. Although we cannot guarantee that our problem is separable, the authors of sep-CMA-ES state [20] that for large problem dimensions their approach is competitive with the full version of CMA-ES, when comparing results achieved in given computation budget.

A. Conformation Motion Representation

We need to represent all the ω_j vectors as a single real-valued vector. As we use sep-CMA-ES, the relative order of the elements in the vector does not matter.

The simplest approach is to simply write out the values from ω_j: $\omega_0^{(1)}$, $\omega_0^{(2)}$, …, $\omega_0^{(2N-2)}$, $\omega_1^{(1)}$, …. However, these values should be the angles from the range $(-\pi; \pi]$ wrapping around π. As CMA-ES requires adding the values taken from a normal distribution to these angles, one needs to take care of that: either truncate every value less than $-\pi$ to $-\pi$ and greater than π to π, or take every result modulo 2π and return it to the range $(-\pi; \pi]$. In the first case, a difficulty of transiting from $-\pi$ to π is introduced, which can make optimization very hard. In the second case, periodicity of the fitness landscape is introduced, which makes every function multimodal and also makes optimization harder.

We deal with these problems by constructing a vector z that is twice as long as the vector from the first approach and compute $\omega_0^{(1)} = \texttt{atan2}(z_1, z_2)$, $\omega_0^{(2)} = \texttt{atan2}(z_3, z_4)$, …. This removes all the problems connected with the periodicity of angles, but makes the search space twice as big.

To make the search space smaller, we do not optimize the angles that differ less than 0.03 between the initial and the final conformations. Instead, these angles are interpolated linearly.

B. Initialization and Parameters

To optimize the cost function using the CMA-ES algorithm, an initial approximation needs to be constructed. We construct the initial approximation by linearly interpolating the angles. The angle interpolation always uses the shortest arc between the angle values.

The covariance matrix is initialized as follows. For each diagonal index i we first locate the torsion angle it corresponds to. The angle difference d for that torsion angle is then computed between the initial and the final conformation. The maximum of 0.01 and d^2 is then used to initialize the diagonal element of the covariance matrix.

We use the default settings for the sep-CMA-ES algorithm [20]. The only free parameters left are the population size and the initial step size. We chose the population size to be 32, which is the standard population size for many experiments, and the initial step size to be 0.001.The limit on the number of iterations of the CMA-ES algorithm is 30000.

IV. EXPERIMENTS

Presentation of the experimental results has the following structure. In Section IV-A, the proposed method is compared on one conformation motion with several existing methods: MovieMaker [8], PATH-ENM [13] and PMPF [10]. In Section IV-B, the proposed method is run on one conformation motion (2M3M, model 1 – 2M3M, model 18), but with different initial approximations. Experiments show that the results depend on the initial approximation, which is an evidence of multimodality of the problem. In Section IV-C, the proposed method is compared with the PMPF algorithm [10], shown to be the best of the existing methods in Section IV-A, on conformation motions between all pairs of conformations of the 2M3M protein.

A. One Motion, Many Algorithms

In this section, we compare experimentally five algorithms for conformation motion prediction: the MovieMaker algorithm [8], which performs linear interpolation of Cartesian

TABLE I
COMPARISON OF PROTEIN CONFORMATION MODELING METHODS

Algorithm	Cost	Collisions
MovieMaker	223901.05	1778.53
PATH-ENM	508114.45	39.73
Torsion interpolation	282094.01	0.0
PMPF	204672.29	0.0
CMA-ES	191173.62	0.0

coordinates of atoms, the PATH-ENM algorithm [13] based on a elastic network model, the algorithm of linear interpolation of torsion angles, which we use as an initial approximation in our method, the PMPF tool [10] and the proposed method.

As a benchmark, we used two conformations of calmodulin: the first one has the PDB ID 1OSA [22], and the second one is 1PRW [21]. The RMSD distance between them exceeds 16Å. In fact, these conformations are from different calmodulins, but their similarity degree reaches 87%. This pair was chosen because 1OSA is a good representer of an open calmodulin structure, while 1PRW is very compact.

All the algorithms constructed a conformation motion with 44 intermediate conformations. This number is chosen for two reasons. First, all methods can compute a motion with a predefined number of intermediate conformations except for PATH-ENM, which determines it by itself. Second, when the parameter p of the cost is not equal to 1, it is impossible to compare the quality of the results with different number of intermediate conformations.

In Table I, the algorithms are compared by the conformation motion cost described in Section II-B and by collision quotient. The latter is defined as follows: for all pairs of consecutive conformations, all pairs of backbone bonds are considered in motion between the conformations. For each pair of bonds, the minimal distance d between any two points on them is computed. A collision quotient for this pair is $(D - d)/D$, where D is the minimal length of a backbone bond. The collision quotients are summed up for all bond pairs for all pairs of consecutive conformations.

One can see that, as expected, the MovieMaker algorithm produces a relatively low cost result, but the number of collisions is very high. PATH-ENM produced the result that is slightly better in collisions, but the cost is twice as big. The torsion angle interpolation produces suprisingly good results, including a relatively low cost and no intersections. The PMPF algorithm, which also uses the torsion angle interpolation as an initial approximation, optimized this result by almost a third. The proposed algorithm shows the best results by both criteria.

B. Multimodality

In this section, we compare the results of the proposed method for different initial approximations. The conformation motion studied in this section is the motion between the first and 18th model of the 2M3M protein [23]. For this motion, the torsion angles with indices 268 and 271 in the backbone differ by more than 1.5 radians between the initial and the final conformation, so it may make sense to interpolate them linearly in two possible directions each — using the shortest arc or the longest arc — resulting in four possible initial approximations in total.

For each initial approximation we conducted eight runs of the proposed method. In Table II, the results are presented. We conducted the Wilcoxon rank sum tests from the R package [29] for all pairs of configurations. All p-values appeared to be less than 0.0009. This is an experimental evidence of the fact that the considered problem is multimodal.

C. Many Motions, Two Algorithms

In this section, we compare experimentally two methods for conformation motion prediction: the proposed one and the PMPF method from [10]. The comparison is performed on all pairs of conformations of the 2M3M protein [23]. This protein has 21 conformations. We constructed conformation motions from all conformations to all other conformations using the proposed method and the PMPF method.

Due to the results of Section IV-B, in the experiments we chose the initial approximation the following way:

1) The set S of torsion angles that differ by more than 1.2 radians between the initial and the final conformations is constructed.
2) For all subsets of S, a transformation is constructed where all torsion angles from the subset are interpolated using the longest arc, whereas all other torsion angles are interpolated using the shortest arc.
3) The transformation with the smallest cost (as in Section II-B) is chosen to be the initial approximation.

The costs of the resulting transformations are presented in Table III. The proposed method produced better results in 159 cases and it was worse in 50 cases. The Wilcoxon signed rank test from the R package [29], conducted for configurations from Table III, reported that p-value is $9.16 \cdot 10^{-7}$. More detailed statistics reveal that the proposed method never loses too much, whereas PMPF can be worse up to an order of magnitude in certain cases.

For every case, we measure the ratio of $|A-B|/\min(A, B)$, where A is the result of PMPF and B is the result of the proposed method. In Fig. 1, the plot of these ratios is shown — for the case the proposed method is worse, the ratio is taken with the negative sign, and the resulting numbers are sorted. It can be seen that in the cases the proposed method loses, it loses only a small percent (4.5% in average). In the cases it wins, the cost of the PMPF motion is 171% bigger in average, and the maximum value of the ratio is 14.11.

We also compare the algorithms by the number of intersections. The motions produced by the proposed method contained no intersections. For PMPF, there were no intersections, but in two motions (3–19 and 2–7) there were the cases when the distance between two non-adjacent bonds was 87.3% and 91.3% of the bond length, correspondingly. For these motions, the cost of PMPF motion was much larger than of the proposed method (8597 vs. 1110 and 7292 vs 707, correspondingly). Our hypothesis is that for these motions PMPF is stuck in a local optimum which is far from the global one.

TABLE II

THE RESULTS OF RUNS FOR THE SAME CONFORMATION MOTION AND DIFFERENT APPROXIMATIONS. EACH ROW CORRESPONDS TO A VARIANT OF APPROXIMATION, WHICH IS DESCRIBED BY A SET OF TORSION ANGLES THAT ARE INTERPOLATED USING THE LONGEST ARC. ALL VALUES ARE ROUNDED TO THE NEAREST INTEGER.

Variant	1	2	3	4	5	6	7	8	mean	dev
$\{\}$	2737	2724	2729	2724	2699	2740	2679	2754	2723	22
$\{268\}$	2321	2185	2296	2268	2303	2161	2414	2342	2286	77
$\{271\}$	2119	2091	2143	2107	1934	1958	2007	1959	2040	78
$\{268, 271\}$	1532	1532	1526	1534	1522	1526	1543	1527	1530	6

TABLE III

COMPARISON OF PMPF AND THE PROPOSED METHOD ON ALL CONFORMATIONS OF 2M3M PROTEIN. THE PART OF THE TABLE ABOVE THE MAIN DIAGONAL CONTAINS THE ENTRIES FOR THE PROPOSED METHOD, THE PART BELOW THE DIAGONAL CONTAINS THE ENTRIES FOR PMPF. FROM EACH CELL PAIR (i, j)–(j, i), CORRESPONDING TO THE SAME MOTION COMPUTED BY DIFFERENT METHODS, THE ONE THAT HAS THE LOWER COST IS MARKED GRAY. ALL VALUES ARE ROUNDED TO THE NEAREST INTEGER.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1		1077	1216	1130	780	999	1803	977	2745	1814	2083	392	3369	1010	1354	753	1540	1548	1430	616	0
2	2280		708	789	446	1621	1919	278	1054	1520	2445	903	1408	624	665	626	665	812	587	556	1075
3	4621	10140		533	885	1469	1932	707	1442	1292	2612	1049	1907	642	1417	669	919	603	648	621	1233
4	4074	3387	518		1060	1134	1193	868	1318	673	3283	953	2249	1113	1673	1086	778	922	1106	1110	1144
5	822	415	5093	1295		1449	1686	450	1868	1650	2018	755	2278	571	815	616	1085	981	1190	303	781
6	1008	1677	8616	1991	1526		1444	1366	2253	858	3065	613	3112	1316	1759	872	2059	1911	1748	893	977
7	3857	22442	23316	2666	9262	7287		1821	2526	722	2988	1589	3362	1799	3002	1668	2042	2070	2733	1378	2169
8	983	314	7292	3975	437	1351	6698		1457	1318	2093	755	1574	494	574	515	659	833	632	338	985
9	2776	1234	13001	1274	2221	2299	9420	1666		2394	5893	2290	790	1783	2237	1901	1372	1541	1466	1994	2729
10	1919	5849	4129	2188	2223	3817	683	3749	3034		2799	873	2797	1357	2412	1380	1308	1249	1669	1090	1861
11	4163	2471	7178	4051	2029	3160	6997	2098	6574	4040		2196	5947	2328	1900	2627	2909	3054	2907	1940	2094
12	403	858	7410	1200	704	628	1698	698	2359	1411	2217		2746	712	1145	528	1439	1328	1040	478	388
13	3343	1483	6121	5061	2285	8118	15452	1628	2339	13072	6596	2775		2011	2697	2160	1811	1774	1557	2214	3388
14	2353	641	8295	6509	551	1266	5855	474	2092	2238	2333	677	2042		1058	491	925	839	700	385	998
15	1411	660	10803	1795	792	1968	6814	601	2578	2676	1856	1085	2725	1061		921	1193	1358	1156	826	1357
16	783	622	1709	1121	591	1006	4714	490	1981	1554	2616	488	2116	459	886		1209	968	946	306	763
17	3769	730	4597	1961	4179	2079	2544	827	1962	1365	3132	4696	8203	3855	3841	1152		481	879	1083	1539
18	2853	781	2883	2708	1573	1887	3922	2747	1707	1297	6451	4845	10599	1511	3317	1025	464		642	901	1547
19	2087	566	9788	1209	1133	1724	11832	597	1405	8722	2920	1004	1597	667	1078	921	3628	5667		965	1420
20	568	585	743	8598	308	1111	4265	311	2398	1656	1929	418	2221	363	877	293	3920	2872	934		617
21	0	2314	3813	3966	841	1057	4666	981	2780	1690	4131	413	3347	2325	1445	793	5073	2856	2393	568	

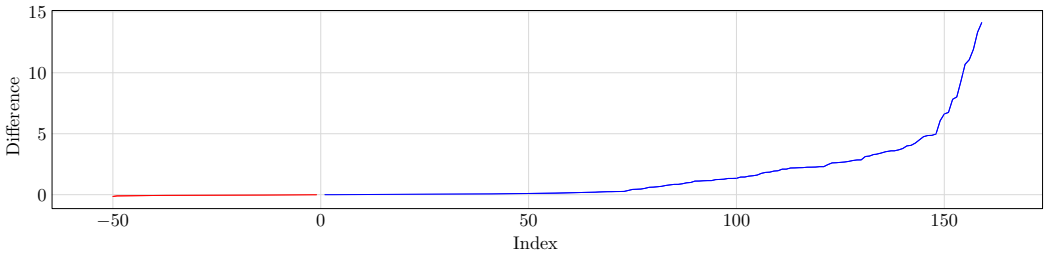

Fig. 1. The plots of differences between the motion costs of proposed method and PMPF divided by the smallest cost. In the cases where the proposed method is worst, the ratio is taken with the negative sign. The differences are sorted.

V. DISCUSSION

In this section, we discuss the problems that are not solved up to the necessary degree in the current research and outline the basis for future work on the topic of the current paper.

A. Multimodality Problem

As shown in Section IV-B, the problem of conformation motion optimization is multimodal. Different possible approximations have significant distances in the configuration space with highly undesirable solutions between them. This suggests that there are even bigger distances between different local optima.

The approach which tests many possible interpolation directions for angles that change a lot can produce accurate solutions for many cases, but this makes it necessary to perform many optimization runs for a single conformation motion. One of the possible ideas is to use generational evolutionary optimizers, like differential evolution, which are able to focus on several hot spots simultaneously.

B. Updating Initial Approximations

Constructing a suitable initial approximation for optimization is difficult. Sometimes it is needed to optimize all pairwise motions between all conformations of the given protein (as we did with 2M3M). In this case, one can benefit from optimizing simple and low-cost motions first and then updating initial approximations for the complex motions from the shortest paths in the graph of already computed motions.

C. Protein Backbone Intersections

Optimizing the mass transportation function can not, in many situations, produce a conformation motion without self-intersections of the backbone. These self-intersections violate physical constraints. However, integration of the penalty function that finds and reports intersections is not an easy task, because optimal alignment and computation of mass transportation can be made in $O(N)$, and straightforward intersection computation requires $O(N^2)$ expensive operations. A more efficient way to find intersections is needed.

D. From Backbone to Full Molecule

The proposed method generates a conformation motion, but the only atoms it outputs are the backbone atoms (C, N and C_α). Most software packages require a full set of atoms. Some programs are able to restore sidechains, but when it is done for all conformations from the motion, the motion of each sidechain may not be optimal any longer. Some heuristics are needed to restore the sidechains in their motions, which also try to minimize mass transportation and avoid collisions.

VI. Conclusion

We have presented a new algorithm for protein conformation motion modeling. In this algorithm, the problem of construction of a reasonably good conformation motion is formulated as a mass transportation problem. The mass transportation cost function is then minimized using the sep-CMA-ES algorithm. The algorithm does not violate the constrants on bond lengths and on backbone planar angles by construction and achieves low values of conformation motion cost.

The code for the experiments is published at GitHub[1]. This work was financially supported by the Government of Russian Federation, Grant 074-U01.

References

[1] H. M. Berman, G. J. Kleywegt, H. Nakamura, and J. L. Markley, "The Protein Data Bank at 40: Reflecting on the Past to Prepare for the Future," *Structure*, vol. 20, no. 3, pp. 391–396, 2012.

[2] I. Bahar, T. R. Lezon, L.-W. W. Yang, and E. Eyal, "Global Dynamics of Proteins: Bridging Between Structure and Function," *Annual review of biophysics*, vol. 39, no. 1, pp. 23–42, 2010.

[3] A. V. Finkelstein and O. B. Ptitsyn, *Protein Physics: a Course of Lectures*. San Diego, CA: Academic Press, 2002.

[4] J. R. Maple, Y. X. Cao, W. G. Damm, T. A. Halgren, G. A. Kaminski, L. Y. Zhang, and R. A. Friesner, "iGNM: a Database of Protein Functional Motions Based on Gaussian Network Model," *Journal of Chemical Theory and Applications*, vol. 1, pp. 694–715, 2005.

[5] R. A. Friesner, "Modeling Polarization in Proteins and Protein-Ligand Complexes: Methods and Preliminary Results," *Advances in Protein Chemistry*, vol. 72, pp. 79–104, 2006.

[6] Y. Dehouck, D. Gilis, and M. Rooman, "A New Generation of Statistical Potentials for Proteins," *Biophysical Journal*, vol. 90, pp. 4010–4017, 2006.

[7] F. E. Boas and P. B. Harbury, "Potential Energy Functions for Protein Design," *Current Opinions in Structural Biology*, vol. 17, pp. 199–204, 2007.

[8] R. Maiti, G. H. van Domselaar, and D. S. Wishart, "MovieMaker: a Web Server for Rapid Rendering of Protein Motions and Interactions," *Nuclear Acids Research*, vol. 33, no. Web Server issue, pp. W358–362, 2005.

[1] https://github.com/mbuzdalov/papers/tree/master/2014-icmla-cmaes-proteins

[9] W. G. Krebs and M. Gerstein, "The Morph Server: a Standardized System for Analyzing and Visualizing Macromolecular Motions in a Database Framework," *Nuclear Acids Research*, vol. 28, no. 8, pp. 1665–1675, 2000.

[10] A. Koshevoy, E.O.Stepanov, and Yu.B.Porozov, "Method of Prediction and Optimization of Conformational Motion of Proteins Based on Mass Transportation Principle," *Biophysics*, vol. 59, no. 1, pp. 28–34, 2014.

[11] I. Bahar and A. J. Rader, "Coarse-Grained Normal Mode Analysis in Structural Biology," *Current Opinions in Structural Biology*, vol. 15, no. 5, pp. 586–592, 2005.

[12] Q. Cui and I. Bahar, *Normal Mode Analysis: Theory and Applications to Biological and Chemical Systems*. Boca Raton: Chapman & Hall/CRC, 2006.

[13] W. Zheng, B. R. Brooks, and G. Hummer, "Protein Conformational Transitions Explored by Mixed Elastic Network Models," *Proteins*, vol. 69, no. 1, pp. 43–57, 2007.

[14] W. Zheng and S. Doniach, "A Comparative Study of Motor-Protein Motions by Using a Simple Elastic-Network Model," *Proceedings of the National Academy of Sciences, USA*, vol. 100, no. 23, pp. 13 253–13 258, 2003.

[15] L. E. Kavraki, P. Svetska, J.-C. Latombe, and M. H. Overmars, "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[16] N. M. Amato, K. A. Dill, and G. Song, "Using Motion Planning to Map Protein Folding Landscapes and Analyze Folding Kinetics of Known Native Structures," *Journal of Computational Biology*, vol. 10, no. 3-4, pp. 239–255, 2003.

[17] B. Raveh, A. Enosh, O. Schueler-Furman, and D. Halperin, "Rapid Sampling of Molecular Motions with Prior Information Constraints," *PLOS Computational Biology*, vol. 5, no. 2, p. e1000295, 2009.

[18] M. Zucker, J. J. Kuffner, and M. Branicky, "Multipartite RRTs for Rapid Replanning in Dynamic Environments," in *IEEE International Conference on Robotics and Automation*. New York: IEEE Press, 2007, pp. 704–710.

[19] M. S. Apaydin, D. L. Brutlag, C. Guestrin, D. Hsu, and J.-C. Latombe, "Stochastic Roadmap Simulation: An Efficient Representation and Algorithm for Analyzing Molecular Motion," in *Proceedings of RECOMB'02*, Washington D.C., 2002, pp. 12–21.

[20] R. Ros and N. Hansen, "A Simple Modification in CMA-ES Achieving Linear Time and Space Complexity," in *Parallel Problem Solving from Nature PPSN X*, ser. Lecture Notes in Computer Science. Springer, 2008, vol. 5199, pp. 296–305.

[21] J. L. Fallon and F. A. Quiocho, "A Closed Compact Structure of Native Ca^{2+}-Calmodulin," *Structure*, vol. 11, no. 10, pp. 1303–1307, 2003.

[22] C. Ban, B. Ramakrishnan, K.-Y. Ling, C. Kung, and M. Sundaralingam, "Structure of the Recombinant Paramecium Tetraurelia Calmodulin at 1.68 Å Resolution," *Acta Crystallographica Section D*, vol. 50, no. 1, pp. 50–63, 1994.

[23] A. Mischo, O. Ohlenschlager, P. Hortschansky, R. Ramachandran, and M. Gorlach, "Structural Insights into a Wildtype Domain of the Oncoprotein E6 and Its Interaction with a PDZ Domain." *PLoS One*, vol. 8, pp. e62 584–e62 584, 2013.

[24] J. Parsons, J. B. Holmes, J. M. Rojas, J. Tsai, and C. E. M. Strauss, "Practical Conversion from Torsion Space to Cartesian Space for In Silico Protein Synthesis," *Journal of Computational Chemistry*, vol. 26, no. 10, pp. 1063–1068, 2005.

[25] L. C. Evans, "Partial Differential Equations and Monge-Kantorovich Mass Transfer," in *Current Developments in Mathematics, 1997, International Press*, 1999.

[26] W. Kabsch, "A Solution for the Best Rotation to Relate Two Sets of Vectors," *Acta Crystallographica*, vol. 32, pp. 922–923, 1976.

[27] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES)," *Evolutionary Computation*, vol. 11, no. 1, pp. 1–18, 2003.

[28] A. B. Fulton and W. B. Isaacs, "Titin, a Huge, Elastic Sarcomeric Protein with a Probable Role in Morphogenesis," *BioEssays*, vol. 13, no. 4, pp. 157–161, 1991.

[29] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2013. [Online]. Available: http://www.R-project.org/