A ROBOTICS APPROACH TO ENHANCE CONFORMATIONAL SAMPLING OF PROTEINS

Juan Cortés*, **Ibrahim Al-Bluwi**
CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
Univ de Toulouse, LAAS, F-31400 Toulouse, France
Email: juan.cortes@laas.fr , ialbluwi@laas.fr

ABSTRACT

Proteins are biological macromolecules that play essential roles in living organisms. Furthermore, the study of proteins and their function is of interest in other fields in addition to biology, such as pharmacology and biotechnology. Understanding the relationship between protein structure, dynamics and function is indispensable for advances in all these areas. This requires a combination of experimental and computational methods, whose development is the object of very active interdisciplinary research. In such a context, this paper presents a technique to enhance conformational sampling of proteins carried out with computational methods such as molecular dynamics simulations or Monte Carlo methods. Our approach is based on a mechanistic representation of proteins that enables the application of efficient methods originating from robotics. The paper explains the generalities of the approach, and gives details on its application to devise Monte Carlo move classes. Results show the good performance of the method for sampling the conformational space of different types of proteins.

INTRODUCTION

Proteins are essential components of living organisms. They have a wide range of functions into cells such as catalysis, regulation, signaling, transport, storage and structural functions. In addition to their primary importance in biology, proteins are also key items in other domains. Indeed, proteins are pharmaceutical targets and drugs, their catalytic properties are exploited in biotechnology, and they are used as components of nano-devices in the rising field of bionanotechnology. The study of the relationship between structural and dynamic features of proteins and their function is fundamental in all these domains. Unfortunately, experimental methods to provide accurate, atomic-scale data for such studies are limited and expensive. Therefore, computational methods are being developed since several decades ago to model proteins and to simulate their behavior.

Protein modeling is very challenging because of the large size and the flexibility of these biological molecules. Indeed, an appropriate model of a protein should not involve a single structure, but a set of conformational ensembles. Given a conformational state of a protein, obtained by experimental techniques like X-ray crystallography [1] or by structure prediction algorithms [2], several methods can be applied to explore the conformational space aiming to provide a conformational ensemble suitable for the analysis of physico-chemical properties. Most of these methods are based on molecular dynamics simulations or on the Monte Carlo method [3, 4]. Nevertheless, alternative methods have been proposed in recent years, some of which are originating from robotics [5].

This paper presents an approach to enhance conformational exploration methods. The approach is based on a mechanistic view of proteins. The idea is to cut the protein into small fragments, called tripeptides, that can be represented as kinematic chains, similar to robotic manipulators. Such a representation enables to efficiently perform local deformations of the protein using semi-analytical inverse kinematics methods. The present work focuses on a specific application of this approach for devising Monte Carlo move classes. Nevertheless, the tripeptide-based protein representation introduced below could be exploited within other types of methods.

The Monte Carlo (MC) method [4, 6], explores the conformational space through a random walk. At each iteration, the protein conformation is randomly perturbed, and the trial move is accepted or rejected with a probability that depends on the potential energies of the old and the new states. The main difficulty involving the application of the MC method to proteins consists

*Address all correspondence to this author.

in devising suitable trial move classes for complex chain-like molecules. An effective move class would yield a good acceptance rate (therefore avoiding futile, expensive energy evaluations), while enabling the exploration of large regions of the conformational space. Several types of trial move classes have been proposed over the years to enhance the efficiency of MC methods applied to proteins and other chain-like polymers (e.g. [7–11]). The approach presented in this paper permits to devise different types of move classes and to implement them easily using a unique representation and a single solver.

The paper presents the general aspects on the mechanistic protein representation using the tripeptide-based decomposition. Then, it explains how to implement several move classes based on this representation. The performance of these move classes is then analyzed through the application to different types of proteins.

PROTEIN MODEL

Some Basic Notions of Biochemistry

A protein is a biological macromolecule composed by one or several long polypeptide chains, generally folded in a globular manner (see molecular modeling textbooks - e.g. [12] - for a detailed structural description of proteins). A polypeptide chain is a sequence of amino-acid residues connected by covalent bonds, called peptide bonds, between the amine group of residue i and the the carboxylic acid group of residue $i-1$. The concatenation of these groups, together with the $C_\alpha H$ groups that attach them, forms the main-chain, or *backbone*, of the protein. This backbone is decorated with *side-chains*, which are specific to each amino-acid type.

The conformation (i.e. spatial arrangement) of a protein can be defined by the *Cartesian coordinates* of all its constituent atoms, or by a vector of *internal coordinates* that represent the relative position of bonded atoms. These internal coordinates correspond to the bond lengths, bond angles and bond torsions. A bond length is the distance between two bonded atoms and a bond angle is the angle between two consecutive bonds. The bond torsion between atoms A_{i-1} and A_i is measured by the dihedral angle formed by planes A_{i-2}-A_{i-1}-A_i and A_{i-1}-A_i-A_{i+1}. Since the bond lengths and bond angles vary very slightly at room temperature, they are often considered to be constant parameters in molecular simulations [13]. Under such assumption, the bond torsions are the only degrees of freedom of the molecule. An additional simplification of molecular models is to consider that double bonds, such as peptide bonds in proteins, are rigid connections (i.e. the dihedral angle ω associated with the peptide bond torsion is constant). In summary, the variable parameters that define the conformation of a protein backbone are the pairs of dihedral angles, ϕ and ψ, of all its constituent amino-acid residues. The conformation of the side-chains is determined by a variable number of dihedral angles χ_i for each residue.

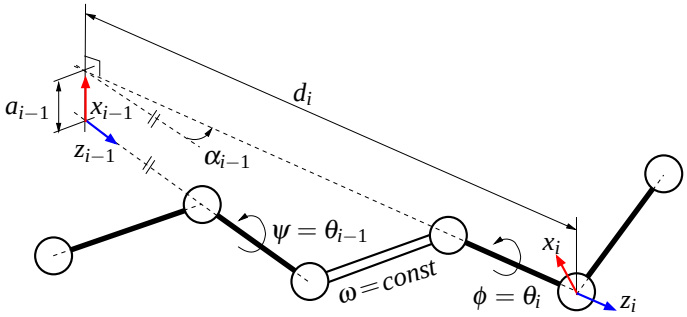

FIGURE 1. GEOMETRIC MODEL OF THE PROTEIN BACKBONE AROUND A RIGID PEPTIDE BOND.

Mechanistic Model

Using the internal coordinate representation described above, proteins can be modeled as articulated mechanisms. The bodies of the mechanism correspond to rigidly-bonded atoms, and the joints are the bond torsions. The kinematic chains corresponding to the protein backbone and side-chains can then be modeled using standard conventions usually applied in robotics. In this work, we have used the *modified Denavit-Hartenberg* (mDH) convention described in [14]. Following this convention, a Cartesian coordinate system F_i is attached to each rigid atom group. The relative location of consecutive frames in a kinematic chain can be then defined by a homogeneous transformation matrix of the form:

$$ {}^{i-1}T_i = \begin{pmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{i-1} \\ \sin\theta_i\cos\alpha_{i-1} & \cos\theta_i\cos\alpha_{i-1} & -\sin\alpha_{i-1} & -d_i\sin\alpha_{i-1} \\ \sin\theta_i\sin\alpha_{i-1} & \cos\theta_i\sin\alpha_{i-1} & \cos\alpha_{i-1} & d_i\cos\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

The elements of ${}^{i-1}T_i$ depend on the bond geometry, being the bond torsion angle θ_i the only variable parameter. Fig. 1 illustrates the method to assign the frames and to obtain the mDH parameters when peptide bond torsion angles (ω) are considered to have an arbitrary constant value.

Decomposition Into Tripeptides

The main idea behind the approach proposed in this paper is to cut the protein chain into fragments of three amino acid residues, which we refer to as *tripeptides*. The reason for choosing such a subdivision is that each tripeptide backbone involves 6 degrees of freedom (three pairs of angles ϕ, ψ), therefore corresponding to the shortest fragment with full mobility of the end-frame relatively to the base-frame. Figure 2 illustrates this idea. Figure 2.a shows a protein model with a ribbon representing the backbone embedded in the model of the protein surface. Figure 2.b represents the protein backbone trace with the frames

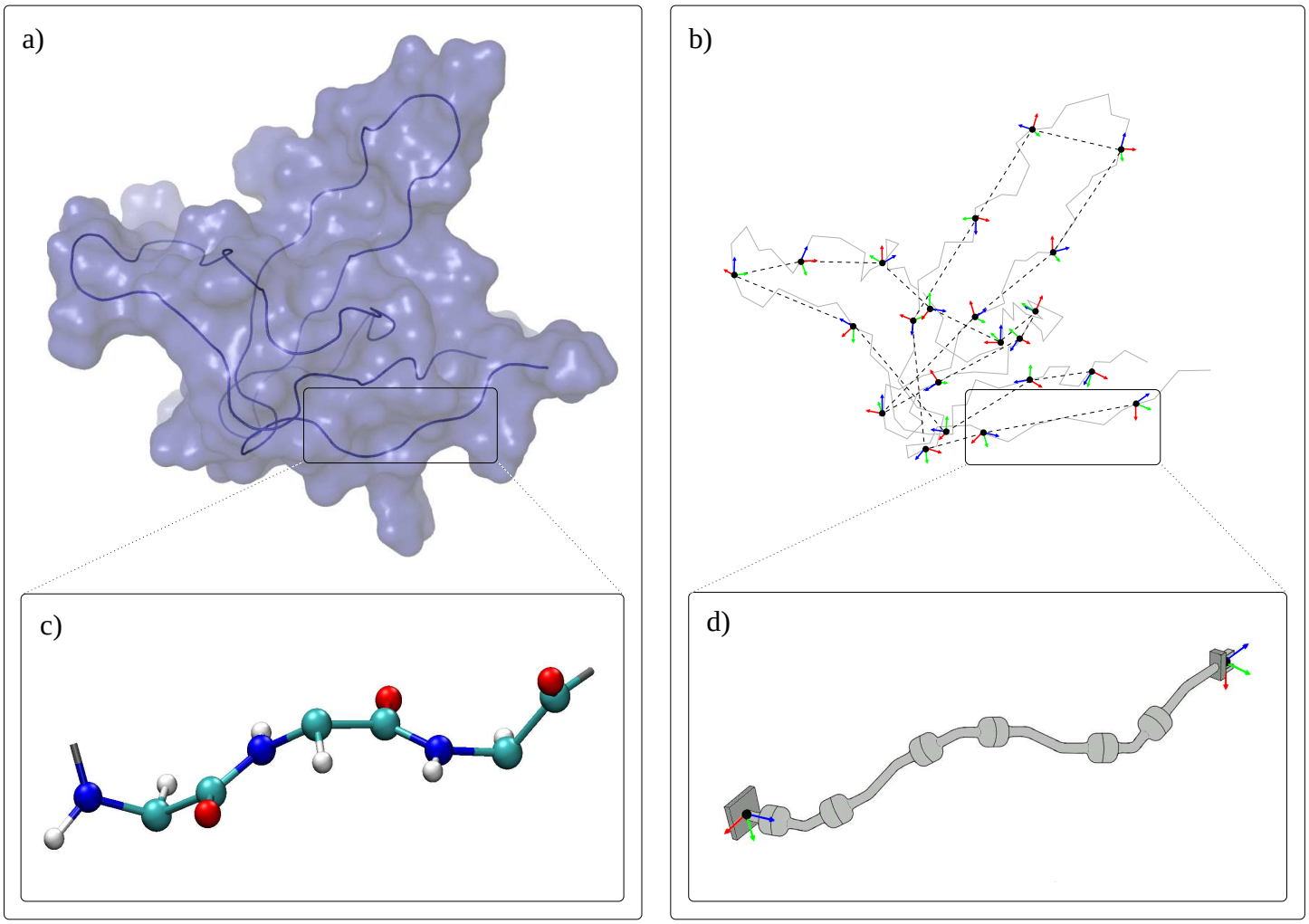

FIGURE 2. ILLUSTRATION OF THE PROTEIN SUBDIVISION APPROACH. FRAGMENTS OF THREE AMINO ACIDS ARE TREATED AS KINEMATIC CHAINS, SIMILAR TO ROBOTIC MANIPULATORS.

corresponding to the ends of the tripeptides. Figures 2.c and 2.d represent respectively the chemical and the mechanistic models of the backbone of a tripeptide. As depicted in the figure, the tripeptide backbone can be seen as a robotic manipulator with six revolute joints. The base of the manipulator corresponds to the first body of the tripeptide backbone (i.e. the first rigid atom group in the backbone of the first amino-acid residue). Since tripeptides are linked through rigid peptide bonds, the location of the end effector of tripeptide i can be determined from the base-frame of tripeptide $i+1$ by a constant transformation. Given the location of the base-frame and the end-frame, the conformation of a tripeptide backbone can be determined by *inverse kinematics* (IK). The IK solver applied in this work is described in the next section. Consequently, the conformation of the whole protein backbone can be determined from the pose of a single reference frame for each tripeptide, the one attached to the first body of each tripeptide backbone. In the following, we will refer to these reference frames as (oriented) *particles*. The last affirmation is true for all the protein backbone except two short fragments at the N-terminal and C-terminal ends of the chain. Indeed, since the choice of the first residue for the decomposition into tripeptides is arbitrary (and may change during the conformational exploration process) the polypeptide chain model involves two terminal fragments, containing up to three residues, which require a particular treatment. The conformation of these terminal fragment can be simply defined by their internal bond torsions.

DEVISING MOVE CLASSES

This section presents a unified approach for devising different move classes using the tripeptide-based representation described above. The principle consists in perturbing the pose (position and orientation) of particles, and then to adapt the conformation of the tripeptides in order to keep the integrity of the molecular chain while maintaining the local geometry of the bonds (i.e. constant bond lengths and bond angles). Several strategies can be considered for perturbing the pose of particles. The number of particles selected for perturbation and the correlation/uncorrelation between the motion direction of several particles will lead to different move classes, more or less local, and more or less collective.

Next we explain how to implement simple, general-purpose move classes. Here, we only deal with unbiased random moves. However, our approach is also suitable for devising biased moves. In such case, the selection of the particles to be perturbed and the motion directions would be determined depending on the specific context. For instance, moves could be devised to deform proteins while simulating the interaction with other molecules, or for applications such as structure fitting into electron density maps.

Note that the move classes presented below are compatible with respect to each other, and with other moves such as *pivot moves* applied to a single bond torsion. Therefore, they could be combined within a higher-level sampling protocol that selects a move class at each iteration. In addition, the constraints on bond lengths, bond angles, and peptide bond torsions imposed by the tripeptide-based model can be relaxed by performing separate MC moves on these parameters. Finally mention that side-chain moves, that are usually performed separately from backbone moves, are not treated in this section. Side chain conformations can be sampled by simple perturbations of the bond torsion angles χ_i or following more sophisticated approaches (e.g. [10,15]). Different strategies can also be adopted to combine backbone and side chain trial moves in a correct and efficient manner (see for instance [16]).

Perturbing Particles

Figure 3 illustrates three move classes that can be easily implemented from the proposed tripeptide-based model. They involve perturbations of one or several particles as explained next.

One Particle Moves. Local moves can be implemented by perturbing the pose of a single particle, as depicted in Figure 3.a. Such perturbation implies that the two tripeptides linked through this particle (i.e. with end-frame or base-frame defined from it) change their backbone conformation. In other words, 12 consecutive bond torsions are modified, while the rest of the protein conformation remains unchanged. This type of moves will have a similar effect to other local, fixed-end move classes

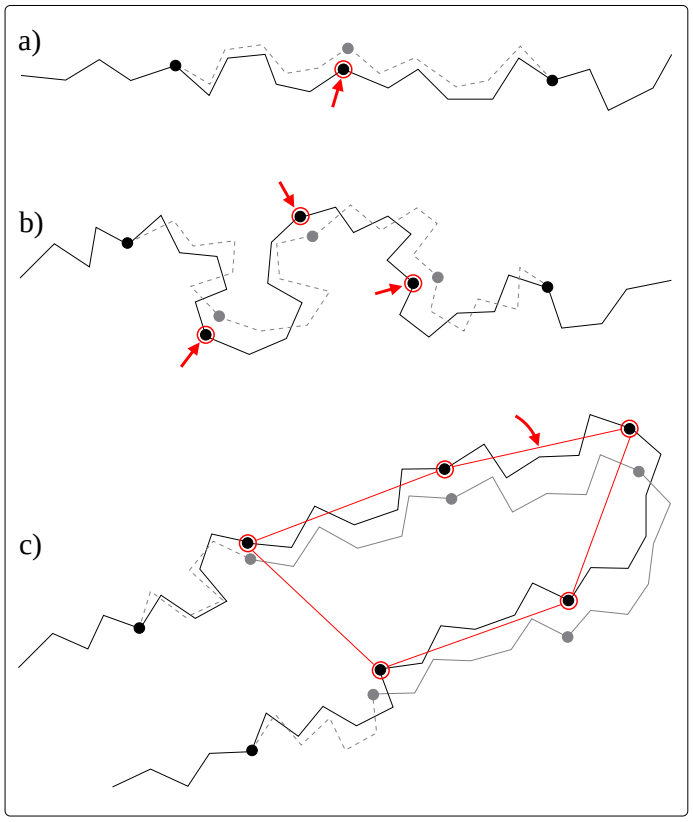

FIGURE 3. ILLUSTRATION OF THREE MOVE CLASSES DEVISED FROM A TRIPEPTIDE-BASED REPRESENTATION OF PROTEINS.

(e.g. [8,9,17]) proposed from the seminal work of Go and Scheraga [7].

Flexible Fragment Moves. The previous move class can be extended to larger fragments by applying perturbations to a set of n consecutive particles. The move class is illustrated in Figure 3.b for the case of three particles. If the particles are perturbed independently (i.e. in different random directions), the backbone of $n+1$ tripeptides is affected by the move. This move class will have a similar effect to moves based on the cyclic coordinate descent method (CCD) [18]. Such moves consist in breaking the chain at an arbitrary point, performing a random perturbation of several bond torsions at one of the sides, and applying CCD to close the chain again (but with a different, perturbed conformation). The expected advantages of the the proposed move class with respect to CCD-based moves are that, if desired, it permits to modulate the deformation of sub-fragments (e.g. larger perturbations for the middle particles in the fragment and smaller perturbations near the ends), and that it is not subject to numerical convergence issues.

Rigid-body Block Moves. A simple variant of the previous move class may produce a very different effect, as illustrated in Figure 3.c. In this case, n particles are also perturbed, but the perturbations are correlated in such a way that the particles do not move with respect to each other. Indeed, the perturbation is applied to a fictitious rigid body formed by the set of n particles. In principle, a random translation and rotation around an arbitrary axis could be tried. Nevertheless, it may be more interesting to apply moves that simulate hinge motions. Note that only two "hinge" tripeptides, the ones preceding and following the selected particle sequence, change their conformation. Hinge-like moves, called closed rigid-body rotation under bond-angle restraints (CRRUBAR) moves [11], have been shown to be particularly efficient for sampling conformations of proteins. Although the proposed method involves more complex algebraic operations than CRRUBAR, it presents the advantage that bond angles do not need to be perturbed.

Solving Inverse Kinematics for a Tripeptide

Once the location of the particles is set, obtaining the conformation of each tripeptide requires to solve an inverse kinematics (IK) problem for the kinematic chain corresponding to its backbone. As explained above, the model of a tripeptide backbone is similar to a six-revolute (*6R*) serial manipulator with general geometry. The method applied in this work for solving the IK problem for a general 6R serial kinematic chain has been adapted from the solver proposed by Renaud [19,20]. This solver is based on algebraic elimination theory, and develops an ad-hoc resultant formulation inspired by the work of Lie and Liang [21,22]. Starting from a system of equations representing the IK problem (the formulation involves the product of homogeneous transformation matrices), the elimination procedure leads to an 8-by-8 quadratic polynomial matrix in one variable. The problem can then be treated as a generalized eigenvalue problem, as was previously proposed by Manocha and Canny [23], for which efficient and robust solutions are available [24]. Our implementation applies the Schur factorization from LAPACK [25]. Technical details on the applied IK solver are provided in the technical report of Renaud [20].

This solver has been successfully applied in previous works on protein and polymer modeling [26, 27]. The advantage of this semi-analytical method with respect to numerical (optimization-based) methods, such as CCD, is that it provides the exact solution in a single iteration, not suffering from numerical convergence issues. The solver is very computationally efficient, requiring about 0.2 milliseconds on a single processor. Note however that our approach is not dependent on this solver, so that other IK methods (e.g. [23, 28]) could be applied.

In general, the IK problem for a 6R serial kinematic chain has a finite number of solutions (up to 16 in the most general case). All the solutions correspond to geometrically valid conformations of the tripeptide backbone with fixed ends defined by the pose of the particles. Depending on the type of applications, several strategies can be adopted to select one of the solutions. The simplest strategy within a MC method consists in selecting one at random. However, if the moves are performed for finding energy minima, all the conformations can be evaluated in order to keep the best one (in terms of the Boltzmann factor, for instance). If detailed balance needs to be satisfied for a correct sampling of equilibrium fluctuations in the canonical ensemble, some works recommend to take one of the solutions with a probability that depends on the Boltzmann factor and on a term, called Jacobian, which is aimed to correct the non-uniformity in the distribution of the torsion angles introduced by the closed-chain moves [8, 10]. Otherwise, when the goal is to simulate continuous motions, the closest conformation to the one before the perturbation has to be selected in order to avoid jumps in the conformational space (if none of the solutions remains within a distance threshold that depends on the perturbation step-size, the local move has to be rejected).

RESULTS

This section presents first results aimed to validate our approach. We have implemented three simple move classes, illustrated in Figure 4:

- The simplest class of trial moves, largely applied to sample the conformation of chain-like molecules, consists in perturbing a randomly selected bond torsion and then propagating the motion toward the end of the chain. Such moves, usually called pivot moves, are named here *OneTorsion* moves.
- The second move class is named *ConRot*, since it is inspired from the *concerted rotations* proposed by Dodd *et al.* [8]. It has been implemented using the tripeptide-based model as follows: an amino-acid residue is randomly selected and one of its bond torsions (ϕ or ψ) is randomly perturbed; the backbone conformation of the next three residues (the next tripeptide) is modified by inverse kinematics in order to maintain fixed ends.
- The third move class, called *OneParticle* moves, corresponds to the simplest move class involving particle perturbations, as described in previous section.

These three move classes have been applied within a basic MC method, using the Metropolis criterion [6] to accept or to reject trial moves. At each iteration, the algorithm randomly chooses between performing either a backbone move (i.e. *OneTorsion*, *ConRot*, or *OneParticle*) or a side-chain move. A side-chain move consists in randomly choosing one side-chain and perturbing all of its dihedral angles χ_i. The conformational parameters (bond torsions or oriented particle poses) are perturbed by adding a random value to their old value, sampled in the inter-

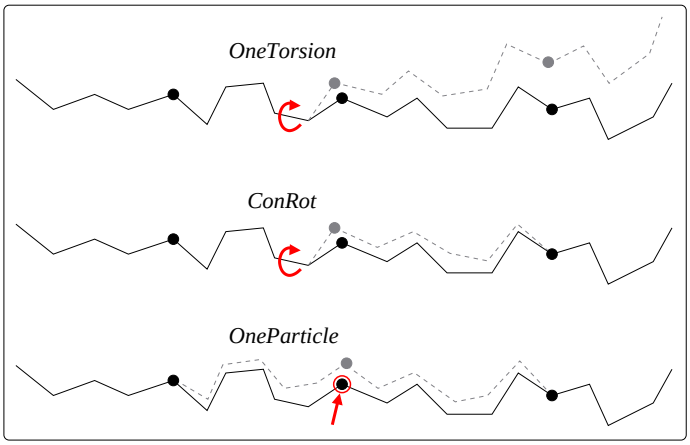

FIGURE 4. ILLUSTRATION OF THE THREE MOVE CLASSES IMPLEMENTED FOR COMPARATIVE ANALYSIS.

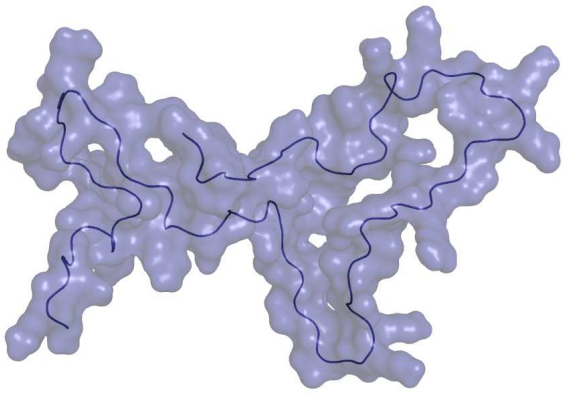

FIGURE 5. MODEL OF THE SIC1 PROTEIN.

TABLE 1. SETS OF PERTURBATION STEP-SIZES

		δ_b	δ_{pt}	δ_{pr}
SH3	Step I	0.03 rad.	0.1 Å	0.01 rad.
	Step II	0.09 rad.	0.3 Å	0.03 rad.
Sic1	Step I	0.09 rad.	0.3 Å	0.03 rad.
	Step II	0.15 rad.	0.5 Å	0.05 rad.

val $[-\delta, \delta]$. Thus, the parameter δ defines the maximum perturbation step-size. Two sets of values have been used for the perturbation step-size of bond torsions (δ_b), particle translations (δ_{pt}) and particle rotations (δ_{pr}). For each set, the values have been chosen in such a way that they will produce comparable atom displacements in a chain fragment. The values considered for the two test systems presented below are given in Table 1. All the simulations have been performed at a temperature of 300 K. For energy evaluation, we have used the OPLS-AA force-field [29] together with an implicit representation of the solvent using the Generalized Born approximation. Note that a geometric filter is applied before energy evaluation with the aim of reducing the number of calls to such a costly function. After applying each trial move, the model is checked for atom overlaps. A trial move is rejected if the distance between two non-bonded atoms is less than 70% of the van der Waals equilibrium distance [30]. In addition, for *ConRot* and *OneParticle* move classes, a trial move is rejected if the IK solver fails to find a solution.

We have chosen two different types of proteins to evaluate the performance of the move classes. The first one is the *SH3 domain* of obscurin, represented in Figure 2.a. This is a small globular protein composed of 68 amino-acid residues. Its crystal structure is available in the Protein Data Bank (PDB ID: 1V1C). The second test system is an intrinsically disordered protein called *Sic1 protein*, which contains 77 residues. A model of this protein is shown in Figure 5. The model was generated using the Flexible-Meccano method [31] for sampling a statistically probable backbone conformation, and SCWRL4 [32] for the side chains. The results presented in this section are not aimed to provide new insights into these biological systems, but to serve as a proof of concept and to show the interest of the proposed method.

Before running the tests, the molecules were equilibrated by running 10^5 MC steps, using *OneParticle* moves with *step-size I* (see Table 1). Starting from the equilibrated conformations, the MC method was iterated until the generation of 10^6 accepted conformations for each molecule, using the three move classes and the two step-size sets. Results are presented in Table 2 and Figure 6. Table 2 contains results on the computational performance of the methods. It shows the total move acceptance rate, the number of calls to the energy evaluation function and the overall CPU time for each system and each setting. Note that simulations were run on a single AMD Opteron 148 processor at 2.6 GHz. Figure 6 shows plots aimed at comparing the performance of the different move classes in terms of conformational space coverage. It shows the projection of the sampled states on two-dimensional plots, where the two variables are the distance with respect to a reference conformation (the initial conformation of the molecule before equilibration) and the potential energy. The distance is measured as the root mean square deviation (RMSD) of the bond torsion angles. For clarity reasons, only one every 1000 conformations has been plotted (i.e. 1000 samples for each experiment). Note that the plot for the Sic1 protein and step-size II is not presented since the overall shapes of the projections are very similar to those obtained with step-size I.

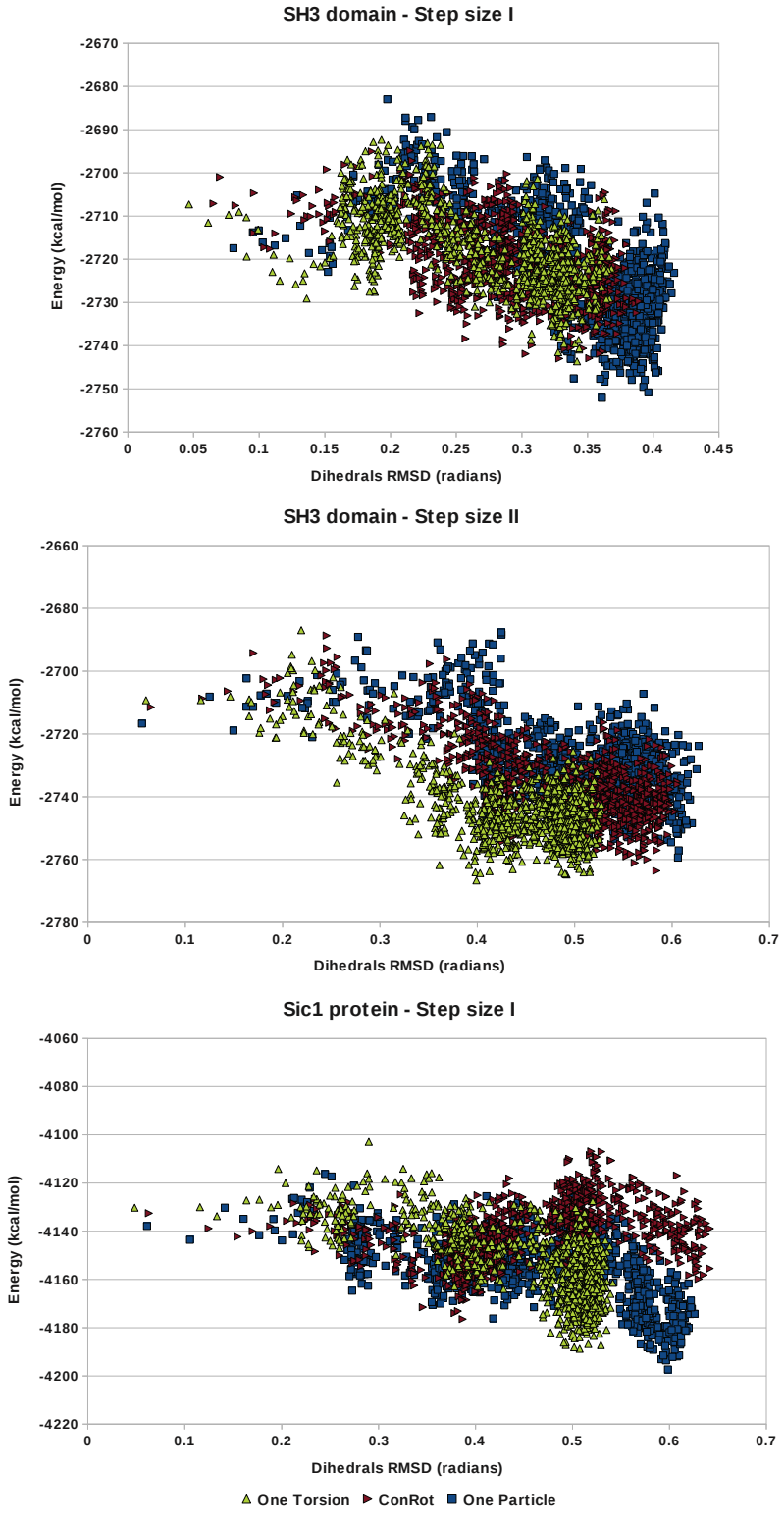

FIGURE 6. PROJECTION OF SAMPLED STATES ON DISTANCE VS. ENERGY PLOTS.

TABLE 2. COMPUTATIONAL PERFORMANCE

	Move Class	Rate	# E-eval.	T_{CPU}
SH3 - Step I	*OneTorsion*	0.58	1.7×10^6	49 h.
	ConRot	0.70	1.3×10^6	39 h.
	OneParticle	0.56	1.5×10^6	45 h.
SH3 - Step II	*OneTorsion*	0.40	2.3×10^6	68 h.
	ConRot	0.48	1.7×10^6	51 h.
	OneParticle	0.45	1.6×10^6	53 h.
Sic1 - Step I	*OneTorsion*	0.44	2.2×10^6	57 h.
	ConRot	0.46	1.7×10^6	46 h.
	OneParticle	0.46	1.5×10^6	40 h.
Sic1 - Step II	*OneTorsion*	0.32	2.9×10^6	74 h.
	ConRot	0.35	2.1×10^6	56 h.
	OneParticle	0.43	1.6×10^6	42 h.

Results show that the *OneTorsion* move class is clearly outperformed by the other two move classes, which perform local fixed-end motions. In all the experiments, the MC method using this move class requires more computing time for sampling the 10^6 valid states, and the samples cover a smaller region of the conformational space. The reason is that perturbations of one bond torsion, even if they are very small, may lead to large displacements of atoms far away from the selected bond, which usually produces high-energy conformation. Also note that although this move class does not require solving inverse kinematics, a significant amount of computing time is needed for propagating atom motions along the chain by *forward kinematics*. In other words, the computing time need by the *ConRot* and *OneParticle* move classes to solve inverse kinematics is largely compensated since only the positions of a small number of atoms need to be updated after each move.

As it was expected, the performance of *ConRot* and *OneParticle* moves is comparable. Nevertheless, several advantages of *OneParticle* moves can be outlined. In all the experiments but the first one (the SH3 domain with the smaller step-size I), the number of rejections due to the Metropolis test is smaller for *OneParticle* moves than for *ConRot* moves. Therefore, although the number of calls to the IK solver is larger for the former move class (twice at each iteration), the overall number of (costly) energy evaluations is smaller, which will be an important advantage when applied to larger proteins. This feature is more visible in the experiment for the Sic1 protein with step-size I. For a similar acceptance rate, the number of calls to the energy evaluation function is significantly smaller for *OneParticle* moves, and therefore, the CPU time is notably reduced. A conclusion that can be extracted from Table 2 is that the *OneParticle* move class provides notably better results for the disordered protein than for the globular one. The reason is that local fixed-end moves involving a larger number of atoms can be more easily accepted in disordered regions of a protein than in well-packed regions. Overall, *ConRot* and *OneParticle* moves provide similar results in terms of coverage. Note however that *OneParticle* moves yield a better coverage in the first experiment, where the poorer performance of this move class in terms of acceptance rate is counterbalanced by a significantly larger area covered by the samples. Another advantage of *OneParticle* moves that can not be illustrated in the present experiments, but which will be very interesting for some applications, is that biasing moves in order to deform regions of the protein with respect to directions given in Cartesian coordinates will be straightforward with this move class, while it would be difficult with the other two move classes.

CONCLUSION

We have presented a unified approach to devise efficient Monte Carlo move classes for chain-like molecules. Based on a subdivision of the protein into tripeptides and on the application of 6R inverse kinematics, many of the move classes that have been proposed in the last decades to enhance protein backbone sampling can be easily implemented. Related works have shown that moves similar to the ones described in this paper improve the efficiency of the exploration while maintaining a correct distribution of sampled states (i.e. the Boltzmann distribution). The approach is general, being applicable to proteins on any size and topology. First results for proteins with different topologies have been presented as a proof of concept.

Future work will involve the implementation of a more sophisticated sampling algorithm that performs different types of trial moves selected among a variety of parameterized move classes. We also envisage to investigate the application of our approach to enhance other types of conformational exploration methods. More precisely, we plan to introduce the tripeptide-based representation within path-planning-based methods to compute motions associated with protein-ligand interactions [33]. Another research direction will concern the development of methods to refine predictions of protein-protein docking methods using flexible tripeptide-based protein models.

ACKNOWLEDGMENT

The authors would like to thank Pau Bernadó for providing us with the model of the intrinsically disordered protein Sic1, and Marc Vaisset for his help with software developments.

REFERENCES

[1] Woolfson, M. M., 1997. *An introduction to X-ray crystallography*. Cambridge University Press.

[2] Zaki, M., and Bystroff, C., 2008. *Protein Structure Prediction*. Methods in Molecular Biology, 413. Humana Press / Springer.

[3] Leach, A. R., 2001. *Molecular Modelling: Principles and Applications*. Pearson Education.

[4] Frenkel, D., and Smit, B., 2002. *Understanding Molecular Simulations: From Algorihtms to Applications*. Academic Press.

[5] Al-Bluwi, I., Siméon, T., and Cortés, J., 2012. Motion planning algorithms for molecular simulations: A survey. *Submitted*.

[6] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and E.Teller, 1953. "Equation of state calculations by fast computing machines". *Journal of Chemical Physics,* ***21***, pp. 1087–1092.

[7] Gō, N., and Scheraga, H., 1970. "Ring closure and local conformational deformations of chain molecules". *Macromolecules,* ***3***, pp. 178–187.

[8] Dodd, L. R., Boone, T. D., and Theodorou, D. N., 1993. "A concerted rotation algorithm for atomistic Monte Carlo simulation of polymer melts and glasses". *Molecular Physics,* ***78***(4), pp. 961–996.

[9] Leontidis, E., de Pablo, J. J., Laso, M., and Suter, U. W., 1994. "A critical evaluation of novel algorithms for the off-lattice Monte Carlo simulation of condensed polymer phases". *Advances in Polymer Science,* ***116***, pp. 283–318.

[10] Wu, N. G., and Deem, M. W., 1999. "Efficient Monte Carlo methods for cyclic peptides". *Molecular Physics,* ***94***(4), pp. 559–580.

[11] Betancourt, M. R., 2005. "Efficient Monte Carlo trial moves for polypeptide simulations". *Journal of Chemical Physics,* ***123***, p. 174905.

[12] Branden, C., and Tooze, J., 1999. *Introduction to Protein Structure*. Garland Science.

[13] Scott, R., and Scheraga, H., 1966. "Conformational analysis of. macromolecules. ii. the rotational isomeric states of the normal hydrocarbons". *Journal of Chemical Physics,* ***44***, pp. 3054–3069.

[14] Craig, J. J., 1989. *Introduction to Robotics*. Addison-Wesley.

[15] Nilmeier, J., and Jacobson, M. P., 2008. "Multiscale Monte Carlo sampling of protein sidechains: Application to binding pocket flexibility". *Journal of Chemical Theory and Computation,* ***4***(5), pp. 835–846.

[16] Nilmeier, J., and Jacobson, M. P., 2009. "Monte Carlo sampling with hierarchical move sets: POSH Monte Carlo". *Journal of Chemical Theory and Computation,* ***5***(8), pp. 1968–1984.

[17] Wu, N. G., and Deem, M. W., 1999. "Analytical rebridging Monte Carlo: Application to cis/trans isomerization in proline-containing, cyclic peptides". *Journal of Chemical Physics,* ***111***, pp. 6625–6632.

[18] Canutescu, A. A., and Dunbrack Jr., R. L., 2003. "Cyclic coordinate descent: A robotics algorithm for protein loop closure". *Protein Science,* ***12***(5), pp. 963–972.

[19] Renaud, M., 2000. "A simplified inverse kinematic model calculation method for all 6R type manipulators". In *Current Advances in Mechanical Design and Production VII*, M. F. Hassan and S. M. Megahed, eds. Pergamon, New York, pp. 57–66.

[20] Renaud, M., 2006. Calcul des modèles géométriques inverses des robots manipulateurs 6R. Rapport LAAS 06332, LAAS, Toulouse.

[21] Lee, H. Y., and Liang, C. G., 1988. "A new vector theory for the analysis of spatial mechanisms". *Mechanism and Machine Theory,* ***23***(3), pp. 209–217.

[22] Lee, H. Y., and Liang, C. G., 1988. "Displacement analysis of the general spatial 7-link 7R mechanisms". *Mechanism and Machine Theory,* ***23***(3), pp. 219–226.

[23] Manocha, D., and Canny, J. F., 1994. "Efficient inverse kinematics for general 6r manipulators". *IEEE Transactions on Robotics and Automation,* ***10***(5), pp. 648–657.

[24] Golub, G. H., and Van Loan, C. F., 1996. *Matrix Computations*, 3rd edition ed. Johns Hopkins University Press.

[25] Anderson, E., Bai, Z., Bischof, C., Blackford, S., Demmel, J., Dongarra, J., Du Croz, J., Greenbaum, A., Hammarling, S., McKenney, A., and Sorensen, D., 1999. *LAPACK Users' Guide*, 3rd edition ed. Society for Industrial and Applied Mathematics, Philadelphia, PA.

[26] Cortés, J., Siméon, T., Remaud-Siméon, M., and Tran, V., 2004. "Geometric algorithms for the conformational analysis of long protein loops". *Journal of Computational Chemistry,* ***25(7)***, pp. 956–967.

[27] Cortés, J., Carrion, S., Curco, D., Renaud, M., and Aleman, C., 2010. "Relaxation of amorphous multichain polymer systems using inverse kinematics". *Polymer,* ***51***(17), pp. 4008–4014.

[28] Coutsias, E. A., Seok, C., Jacobson, M. P., and Dill, K. A., 2004. "A kinematic view of loop closure". *Journal of Computational Chemistry,* ***25***(4), pp. 510–528.

[29] Jorgensen, W. L., Maxwell, D. S., and Tirado-Rives, J., 1996. "Development and testing of the opls all-atom force field on conformational energetics and properties of organic liquids". *Journal of the American Chemical Society,* ***118***(45), pp. 11225–11236.

[30] Bondi, A., 1964. "Van der Waals volumes and radii". *Journal of Physical Chemistry,* ***68***, pp. 441–451.

[31] Bernadó, P., Blanchard, L., Timmins, P., Marion, D., Ruigrok, R., and Blackledge, M., 2005. "A structural model for unfolded proteins from residual dipolar couplings and small-angle x-ray scattering". *Proceedings of the National*

Academy of Sciences of the U.S.A., ***102***, pp. 17002–17005.

[32] Krivov, G. G., Shapovalov, M. V., and Dunbrack Jr., R. L., 2009. "Improved prediction of protein side-chain conformations with SCWRL4". *Proteins: Structure, Function, and Bioinformatics,* ***77***(4), pp. 778–795.

[33] Cortés, J., Le, D., Iehl, R., and Siméon, T., 2010. "Simulating ligand-induced conformational changes in proteins using a mechanical disassembly method". *Physical Chemistry Chemical Physics,* ***12***(29), pp. 8268–8276.