



HAL
open science

A Reinforcement Learning Approach to Protein Loop Modeling

Kevin Molloy, Nicolas Buhours, Marc Vaisset, Thierry Simeon, Étienne Ferré,
Juan Cortés

► **To cite this version:**

Kevin Molloy, Nicolas Buhours, Marc Vaisset, Thierry Simeon, Étienne Ferré, et al.. A Reinforcement Learning Approach to Protein Loop Modeling. IEEE/RSJ International Conference on Intelligent Robots and Systems, Sep 2015, Hamburg, Germany. hal-01206128

HAL Id: hal-01206128

<https://hal.science/hal-01206128>

Submitted on 28 Sep 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Reinforcement Learning Approach to Protein Loop Modeling

Kevin Molloy^{*†}, Nicolas Buhours^{*†‡}, Marc Vaisset^{*†}, Thierry Siméon^{*†}, Étienne Ferré[‡], Juan Cortés^{*†}

^{*}CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France

[†]Univ de Toulouse, LAAS, F-31400 Toulouse, France

[‡]Siemens Industry Software, 478 rue de la découverte, 31670 Labège, France

Abstract—Modeling the loop regions of proteins is an active area of research due to their significance in defining how the protein interacts with other molecular partners. The high structural flexibility of loops presents formidable challenges for both experimental and computational approaches. In this work, we combine a robotics approach with reinforcement learning (RL) to compute an ensemble of loop configurations. We are actively performing experiments on well known benchmark sets to illustrate how RL improves the efficiency and effectiveness of our approach.

I. INTRODUCTION

Protein loops can exhibit high flexibility which can prove challenging to model with experimental techniques such as X-Ray crystallography. The loop regions of proteins can control how the protein interacts with other molecular partners, and as a result, a large number of computational approaches have been proposed. We define the loop modeling problem as, given a protein P and a loop defined by its starting and ending residue (L_{start} and L_{end}), compute a representative ensemble of conformations Ω . Early applications of loop modeling include supplementing data from experimentally solved protein structures and computing loops in the context of homology and ab-initio protein structure prediction protocols. In our approach, we conduct an extensive search and generate an ensemble of loop configurations that effectively map the feasible conformational space for the loop region. This provides insight into the dynamics and energy surface associated with loop configurations, which can be utilized to better understand their functional roles.

Proteins can be represented using simplified models where the degrees of freedom (DOFs) are the dihedral bond angles. The resulting search space is still vast even for relatively small loops. The protein loop modeling problem resembles the inverse kinematics (IK) problem in robotics and computer graphics where the protein’s backbone can be viewed as an articulated linkage. The goal then becomes to assign values to each of the dihedral angles such that the two ends of the loop keep connected to the rest of the protein, in effect closing the loop, while avoiding collisions with itself and the protein. Several robotics-inspired approaches to loop modeling have been proposed over the years [1], [2], [3]. Shehu and Kavraki provide a good review of the techniques applied to loop sampling [4].

In this paper, we propose a method for assembling an ensemble of valid loop structures. To address the vast con-

formation space, we propose discretizing the search space via a database of small, contiguous segments of experimentally observed loop configurations organized by the corresponding amino acid sequence. We further organize this database into a multi-dimensional grid and employ a reinforcement learning (RL) [5] strategy to bias the selection of configurations from the database.

II. METHODS

First, we describe our method for solving the loop closure problem. We then proceed to describe how we incorporate RL into this approach.

A. Loop Construction

We represent the protein using an all-atom model. Proteins share a common backbone or scaffold that allow amino acids to be joined together to form long polypeptide chains. Typically, the backbone is modeled with 2 DOFs per amino acid (the ϕ and ψ angles). Our method first searches for a collision free backbone configuration that closes the loop, and then adds the residue specific sidechains.

The loop region is decomposed into k tripeptides (continuous segments of the protein loop consisting of 3 amino acids). We utilize a database of tripeptides, which are excised from a set of more than 10,000 non-redundant experimentally solved proteins from the SCOP database [6], and index them by the tripeptide’s amino-acid sequence. This approach discretizes the search space and capitalizes on the prior knowledge encoded within experimentally determined proteins. Others have proposed combining databases and inverse kinematics in the context of loop sampling [7]. For each iteration of our method, we identify one tripeptide, s , that we will solve using IK. For each of the remaining positions, we will draw random samples from the database. The loop is constructed starting from the tripeptide at position 1 to position $s - 1$, and then from the end of the loop at position K backwards to position $s + 1$.

Each tripeptide sampled from the database is attached to the linkage. If collisions are detected, we redraw a random sample (for a maximum of $MAXTRIES$ times per position). When $MAXTRIES$ is reached for tripeptide i , we reset the failure count to zero for tripeptide i , increment the failure counter for the prior position $i - 1$ and draw a new random sample for position $i - 1$. The sampling terminates when:

1) we reached *MAXTRIES* for the first position; 2) a collision free loop has been constructed for the $K-1$ tripeptide positions. We solve for the last tripeptide position using IK and attempt to place the loop's sidechains in a collision free configuration. If the sidechains are successfully placed, a short Monte Carlo minimization is performed to help improve the potential energy and the conformation is added to Ω . An ensemble of conformations is created by repeating this process many times.

B. Using Reinforced Learning for Tripeptide Selection

The method described in the previous section employs a naïve strategy for selecting tripeptides (randomly selecting configurations). In this work, we investigate tripeptide attributes and how we can utilize this information in a RL strategy.

We first organize the tripeptides for each position k by constructing an n dimensional feature vector. We then discretize the feature space using an n dimensional multi-resolution grid. Each cell in the grid effectively clusters together tripeptides with similar features. For each cell, we record the number of times a tripeptide from that cell participated in a successful or failed loop closure. When a cell's heterogeneity (with respect to success/failures) exceeds a threshold, the cell is subdivided into 2^m neighboring cells, which in turn provide a higher resolution for these regions. This scheme resembles an octree (except in n dimensions), where the hierarchy of cells allow the grid to be viewed as a tree (with the root representing the entire grid).

When selecting a tripeptide for position k , the success of this selection is dependent on the previous $k-1$ selections. To capture this dependency, each grid cell points to an entire new grid structure for the next loop position to be sampled. Each grid cell (at all levels in hierarchy) is assign a score, which captures downstream success or failures as shown in Eq. 1.

$$score_c^k = score_c^k \times \prod_{m=k+1}^K score^m \quad (1)$$

For position k , the score for cell c is equal to its score times the score of the grids for the remaining positions $k+1 \dots K-1$. The scores of each cell are then used by a selection strategy, that is biased to select cells based on their participations in successful loop closures.

III. RESULTS

We are applying our RL loop sampling method to several benchmark datasets including the ones proposed by Canutescu and Dunbrack [3], and the set utilized in work by Wang et al [7]. Our proposed approach investigates several alternative methods to organize our database of tripeptides and several selection techniques.

A. Tripeptide Features

We are investigating different feature vectors to project our tripeptides into a n dimensional multi-resolution grid and how each of these projections impact the effectiveness of the RL

strategy. Some features being considered are: relative position of the last atom (within a tripeptide), orientation (of the last rigid body of the tripeptide), orientation and the length of the tripeptide, axis and angular rotation (with respect to the beginning and end of the tripeptide).

B. Selection Strategy

Each hierarchical cell in our multi-resolution grid is assigned a score which is used to guide future selections. In this work, we will explore different methods to select cells from this grid. Two examples of selection methods we employ are a greedy scheme and a probabilistic scheme. In the greedy scheme, we select the cell with the highest score. In the probabilistic scheme, each cell is given a normalized weight and is sampled with respect to these weights.

IV. CONCLUSIONS

Loop modeling is an important open problem in the field of structural biology. Computational methods, like the one proposed here, help supplement the available experimental data and augment our understanding of the role of protein loops. Understanding the flexibility of loops is important in many fields such as pharmacology and protein design, and we believe the techniques proposed here will help guide our sampling procedure to provide an efficient and meaning representation of the loop's conformational space. We have obtained interesting preliminary results which will be presented in our corresponding poster. Our proposed technique can also be used to augment robotic motion planning problems that involve generating samples in high dimensional space.

V. ACKNOWLEDGMENT

This work has been partially supported by the French National Research Agency (ANR) under project ProtiCAD (project number ANR-12-MONU-0015).

REFERENCES

- [1] J. Cortés, T. Siméon, M. Remaud-Simon, and V. Tran, "Geometric algorithms for the conformational analysis of long protein loops," *Journal of Computational Chemistry*, vol. 25, no. 7, pp. 956–967, May 2004.
- [2] P. Yao, A. Dhanik, N. Marz, R. Propper, C. Kou, G. Liu, H. van den Bedom, J.-C. Latombe, I. Halperin-Landsberg, and R. B. Altman, "Efficient Algorithms to Explore Conformation Spaces of Flexible Protein Loops," *Ieee/Acm Transactions on Computational Biology and Bioinformatics*, vol. 5, no. 4, pp. 534–545, 2008.
- [3] A. A. Canutescu and R. L. Dunbrack, "Cyclic coordinate descent: A robotics algorithm for protein loop closure," *Protein Science : A Publication of the Protein Society*, vol. 12, no. 5, pp. 963–972, May 2003. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC2323867/>
- [4] A. Shehu and L. E. Kaviraki, "Modeling Structures and Motions of Loops in Protein Molecules," *Entropy*, vol. 14, no. 2, pp. 252–290, Feb. 2012.
- [5] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.
- [6] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia, "SCOP: a structural classification of proteins database for the investigation of sequences and structures," vol. 247, pp. 536–540, 1995.
- [7] C. Wang, P. Bradley, and D. Baker, "Protein Protein Docking with Backbone Flexibility," *Journal of Molecular Biology*, vol. 373, no. 2, pp. 503–519, Oct. 2007.