Sample-based Models of Protein Structural Transitions

Tatiana Maximova
Dept. of Computer Science,
George Mason University,
Fairfax, VA 22030
tmaximov@gmu.edu

Daniel Carr
Dept. of Statistics,
George Mason University,
Fairfax, VA 22030
dcarr@gmu.edu

Erion Plaku*
Dept. of Electrical Engineering
and Computer Science,
The Catholic University of
America,
Washington, DC 20064
plaku@gmu.edu

Amarda Shehu†
Dept. of Computer Science,
Dept. of Bioengineering
School of Systems Biology
George Mason University,
Fairfax, VA 22030
amarda@gmu.edu

ABSTRACT

Modeling structural transitions of a protein at equilibrium is central to understanding function modulation but challenging due to the disparate spatio-temporal scales involved. Of particular interest are sampling-based methods that embed sampled structures in discrete, graph-based models of dynamics to answer path queries. These methods have to balance between further exploiting low-energy regions and exploring unpopulated, possibly high-energy regions needed for a transition. We recently presented a strategy that leverages experimentally-known structures to improve sampling. Here we demonstrate how such structures can further be leveraged to improve both exploitation and exploration and obtain paths of very high granularity. We show that such improvement is key to accurate sample-based modeling of structural transitions. We further demonstrate that ranking methods by the best transition cost obtained can be deceptive, as denser sampling, which follows a rugged landscape more faithfully, may result in higher costs. The work presented here improves understanding of the current capabilities and limitations of sampling-based methods. Proposing strategies to address some of these limitations in this paper is a first step towards sampling-based methods becoming reliable tools for modeling protein structural transitions.

CCS Concepts

•**Applied computing** → **Molecular structural biology;**

*Corresponding Author
†Corresponding Author

ACM BCB 2016 Seattle, WA

DOI: xxx

Keywords

protein structure; energy landscape; transition modeling; sample-based representation; discrete models of dynamics

1. INTRODUCTION

Characterizing protein dynamics at equilibrium is central to elucidating the precise role played by structure in molecular recognition events in the cell. Particularly important instances of protein dynamics are exquisitely regulated transitions among different thermodynamically-stable and semi-stable structures used to bind different molecular partners and thus modulate biological activity in the cell [7].

Characterizing equilibrium protein dynamics involves disparate spatio-temporal scales, and bridging such scales is challenging both in wet and dry laboratories [29]. Elucidating the step-by-step succession of structures constituting the transition of a protein between two function-carrying structures of interest often involves long spatial and/or temporal scales of several angstroms and microseconds [20]. Given the central role that structural transitions play in regulating molecular recognition events, a significant body of research focuses on elucidating transitions between two known structures or structural states of a protein [21].

Given the hardware improvements and algorithmic sophistication in the last decade, computational methods are gaining renewed interest in complementing wet laboratories on modeling structural transitions [21]. All these methods implement a mechanism for navigating the protein structure space (and underlying energy landscape) in search of one or more paths that connect the two structures of interest. A path of higher granularity shows the transition in more detail, via more intermediate structures. Since the protein structure space is multi-dimensional, the focus is either on the most probable transition route, which corresponds to the minimum-energy path, or the similarly-probable routes, which correspond to the set of comparably low-energy paths.

Currently, the more popular computational methods are based on the Molecular Dynamics (MD) approach. There is a great variety of such methods, and the interested reader is referred to the review in [21]. In essence, MD-based methods follow the atoms constituting the protein of interest along

the slope of the energy landscape by iteratively solving Newton's equation of motion on a finely-discretized time scale [3]. Many enhancement strategies have been proposed to address premature convergence in MD-based methods, also known as the multiple minima issue; protein energy landscapes are rich in shallow and deep minima (manifesting in the disparate temporal scales). MD-based methods often operate in a random restart mode so as to explore more of the structure space and possibly discover lower-energy paths.

Methods based on the Monte Carlo (MC) approach are also popular; these methods launch random walks in the structure space biased towards lower energies. They are often known as (transition) path sampling methods and are generally recognized as having higher exploration capability than MD-based methods. A detailed review is presented in [13]. Similarly to MD-based methods, MC-based ones have to employ strategies to enhance exploration. Strategies that allow a random walk/path to escape a local minimum enhance exploration capability, but they have to be carefully implemented so as to balance between seeing more of the structure space (known as exploration) and digging deeper in low-energy regions containing stable and semi-stable structures (known as exploitation).

Of particular interest in this regard are sampling-based methods inspired from robot motion planning; in the latter, a path is sought connecting a start to a goal configuration in the feasible robot configuration space [11]. Robotics-inspired methods build a discrete, sample-based representation of the connectivity of the structure space. They typically consist of two stages. A discrete, sample-based representation of the relevant structure space is built in the first stage. Samples are then embedded in a tree or graph that supports queries for paths connecting two structures of interest. In tree-based methods, the two stages are coupled; a sample is immediately added onto the growing tree of samples. In roadmap-based methods, the stages are decoupled, and samples, once obtained, are embedded in a nearest-neighbor graph.

Significant early contributions from the Latombe, Amato, Kavraki, and Siméon labs have now made it possible to further improve upon the robotics-inspired, sampling-based framework to elucidate credible folding, unfolding, loop, and structural transition motions of small- and medium-size proteins [4–6,12,35] and even obtain highly-detailed yet expansive views of transition routes in small peptides [14,18]. Detailed reviews can be found in [1,16,21].

The body of research on robotics-inspired methods is growing [2,4,10,14,17–19,24–27,30–36], in part due to the remaining challenge of limited sampling. Sampling-based methods have to balance between further exploiting low-energy regions and exploring unpopulated, possibly high-energy regions needed for a transition. Exploring both the breadth (of the structure space) and the depth of the energy landscape associated with the structure space is central to sampling-based methods that rely on embedding samples in graph-based data structures.

In [22,23], we present a roadmap-based method that leverages experimentally-known structures of a protein to improve sampling. The strategy is to expand into the search space from regions populated by the known structures. A statistical analysis technique additionally extracts from the known structures a search space of reduced dimensionality that makes it feasible to explore more of the structure space with a practical computational budget, a few days on one CPU on medium-size proteins $90-166$ amino acids long.

While the work in [22,23] represents the state of the art of robotics-inspired methods for structural transition modeling and indeed reproduces transition routes that reconcile diverse wet-laboratory and computational hypotheses, it also spends a significant portion of its time in climbing out of the low-energy regions populated by the experimentally-known structures. A strategy is devised to remedy this issue by penalizing exploring well-populated regions, but the initialization mechanism favors exploitation over exploration. In particular, it becomes increasingly hard to sample regions of high energy that may represent an energy barrier, as all sampling-based methods have an energy bias incorporated in them to avoid computing physically-unrealistic structures.

Currently, it is not clear how to balance exploitation and exploration so as to sample the regions of relevance for a sought transition, as such regions are not known a priori. Typically, this issue of energy-biased, non-uniform sampling is hard to expose, as a nearest-neighbor graph that connects a sample/structure to its k closest neighbors will mask away scarcely-sampled regions, unless an additional length threshold is imposed on edges. Without such care, path queries will be answered, but the obtained paths may not be realistic; in particular, an edge in a path may effectively tunnel through a "hill" in the landscape and report no energetic barrier if indeed there are no samples on the hill. Disproportionately-long edges will be observed connecting energy basins and tunneling through scarcely-sampled energetic barriers, betraying a limited exploration capability.

In this paper, we demonstrate how the initialization mechanism can be leveraged to address some of these issues. In particular, we demonstrate how experimentally-known structures can further be employed to improve both exploitation and exploration and indeed allow obtaining paths of very high granularity with successive structures no further than $0.1\mathring{A}$ in structure space. We show that such improvement is key to accurate modeling of transitions on discrete models of dynamics. We further demonstrate that ranking methods by the lowest transition cost obtained can be deceptive, as denser sampling exposes hills and possibly higher costs. The work presented here improves understanding of current capabilities and limitations of sampling-based methods for protein structure transition modeling. Proposing strategies to address some of the limitations is a first step towards sampling-based methods becoming reliable tools for modeling protein structural transitions.

2. METHODS

The roadmap-based framework we investigate here first obtains a sample-based representation of the relevant structure space (and underlying energy landscape), and then embeds the samples in a nearest-neighbor graph to answer least-cost path queries. In particular, we build on the `SoPriM` algorithm presented and evaluated in [22,23] to model structural transitions of medium-size proteins. `SoPriM` is able to handle protein chains up to 166 amino acids long, as it generates samples in a variable space of reduced dimensionality; the latter is extracted from a statistical analysis of experimentally-known structures of a protein.

The leveraging of diverse experimentally-known structures of a protein is key to `SoPriM`'s exploration capability. In addition to defining the variable space, the known structures directly provide `SoPriM` with initial samples, effectively ex-

posing local minima in the energy landscape at no computational cost. The initialization is one mechanism to control or bias the sampling stage in roadmap-based algorithms towards the "relevant" structure space. The variation operators that are then employed to generate more samples based on the initial ones have to implement a trade-off between exploration and exploitation. The actual interplay between the initialization mechanism and the variation operators determines the extent to which a sampling-based algorithm will achieve a computationally-feasible trade-off that allows it to then answer path queries reliably. In the following, we first summarize the initialization mechanism as employed in `SoPriM`. We then discuss in detail the variation operators and how they balance between the two conflicting objectives of exploration and exploitation and expose pertinent issues. The rest of this section then describes two alternative initializations designed to address some of these issues.

2.1 Known Structures Initialize the Sample-based Representation of Structure Space

Known structures of a protein can be collected from the Protein Data Bank. In [22], the collection is expanded to include structures of variants no more than 3 point mutations away from the sequence of interest. Doing so is warranted by the conformation selection/ population shift principle [7]. The latter states that mutations change the probability with which structures are populated at equilibrium; that is, structures collected for a variant may be semi-stable or, at worst, high-energy for the sequence of interest, but they are precious seeds for any sampling-based algorithm. The collected structures are stripped down to their CA atoms and subjected to principal component analysis (PCA); the top m eigenvectors/principal components (PCs) that cumulatively capture more than 90% of the variance are employed as variable axes for the exploration; that is, samples are m-dimensional points in the space of the top m PCs (the process is detailed in [22]).

The left panel of Figure 1(a) shows the (initial) ensemble Ω of experimentally-known structures; these are threaded onto the protein sequence of interest, and SCWRL 4.0 is used to pack in side chains at mutated sites. A conjugate gradient descent protocol in Amber is used to obtain local minima structures for the sequence of interest. The left panel of Figure 1(a) shows the resulting structures projected onto the top two PCs, color-coded based on their Amber energies (the Generalized Born solvation model is used in the minimization protocol; details can be found in [22]).

2.2 Beyond Initialization: Growing the Sample-based Representation of Structure Space

In the early robotics-inspired algorithms, new samples were obtained uniformly at random in the variable space; this has a high probability of yielding structures with self collisions, as motions of protein chains are highly constrained. More successful strategies rely on biased sampling; while details vary, the main idea is that the growing ensemble is iteratively subjected to a variation operator; initially, the ensemble consists of one structure (when unfolding routes are sought, as in [34,35], the start and the goal structure (when transitions are sought, as in [17]), or many experimentally-known structures, as in [22] and this paper. The iterative-based application of the variation operator is also intrinsic to tree-based algorithms, which additionally add the obtained sample to the tree after each iteration. In roadmap-based algorithms, only after the sampling stage is terminated are all the samples embedded in a nearest-neighbor graph.

In an iterative-based application of the variation operator, a selection mechanism is needed to select at each iteration a sample to be subjected to the variation operator. The selection can be uniformly at random over all samples in the current ensemble Ω, or biased and employ one or more weighting functions that balance the different objectives of exploration and exploitation. We first summarize the selection mechanism and the variation operator in `SoPriM`, and then expose the interplay and issues that arise.

2.2.1 Selection Mechanism

A grid-based discretization of the variable space (along PC1 and PC2) is used so that regions/cells can be defined; At every iteration, a cell is selected per the weighting function shown in Equation 1:

$$w(\gamma) = \frac{e^{-minE(\gamma)\cdot\alpha}}{(nrConfs(\gamma)\cdot nrSel(\gamma)\cdot nrFailures(\gamma))^2}, \qquad (1)$$

where $minE(\gamma)$, $nrConfs(\gamma)$, $nrSel(\gamma)$, and $nrFailures(\gamma)$ denote the minimum energy over samples that map to a grid cell γ, the number of samples that map to γ, the number of times γ has been selected, and the number of times the variation operator has failed to obtain a successor sample when selecting a sample mapped to γ, respectively. Once a cell is selected per a probability distribution over the weighting function, any sample in the selected cell is then selected uniformly at random to be subjected to the variation operator.

The weighting function penalizes cells of high energy and cells that have been selected before. While the functional formulas that determine the role of energy over other statistics recorded for sampled cells can be different, the general idea of combining the different terms is so as to steer sampling away from high-energy and populated regions; that is, promote exploration of regions of possible relevance for a sought transition. The employment of a grid-based selection mechanism is familiar both in robot motion planning and in robotics-inspired algorithms for protein structural transition modeling, though it has mainly been used in the context of tree-based algorithms [25,26,30] rather than roadmap-based ones. `SoPriM` is one of the first to incorporate the mechanism in the roadmap-based framework.

2.2.2 Variation Operator

The variation operator in `SoPriM` modifies the selected sample along each of its m coordinates. The modification to the coordinate corresponding to the top PC, PC1, is sampled uniformly at random inside a given interval $[-\delta_{max}, \delta_{max}]$; modifications to the other coordinates corresponding to the other PCs, PCi, are computed as $v_i = v_1\lambda_i/\lambda_1$, where λ_i is the eigenvalue of PC_i. The obtained sample is typically followed up with a structure/energetic improvement protocol to correct possible structure deformations (including intra-chain collisions). In `SoPriM`, the obtained sample (which is just an m-dimensional point) is mapped to an all-atom structure space by a protocol detailed in [22]; in summary, the point is mapped to a CA trace, and then the rest of the atoms (backbone and side chains) are packed in and then followed by a conjugate gradient descent protocol (summarized above in Section 2.1).

2.2.3 Interplay between the Selection Mechanism and the Variation Operator

Selection mechanisms are indirect; they attempt to control where samples are generated by the variation operator by instead controlling which samples are selected for variation. This indirect strategy is more likely to succeed if indeed the variation operator yields samples that are adjacent to selected samples in the structure space. If variation operators cannot provide some structural correlation between a selected sample and the generated successor sample, the control strategy is ineffective and degenerates to unbiased random sampling; the latter has been demonstrated in the context of a simple iterative improvement algorithm in [28].

The demand for adjacency ensures that the sampling stage will expand rather gradually from already-visited regions in the structure space. This is visually illustrated in the right panel of Figure 1(a) by showing Ω after 500 iterations of selection and variation (followed by structure correction) in `SoPriM`. Exploration is further slowed down by structure-correcting protocols, which apportion a significant portion of the computational budget in digging deeper in (exploiting) already-populated regions. It is worth noting that structure corrections cannot be avoided, as otherwise the ensemble would be dominated by unreasonable structures with significant deformations and self collisions.

The selection mechanism is the main contributor to exploration, whereas the variation and structure correction operators contribute to exploitation. Figure 1(a) shows `SoPriM` slowly expanding in structure space away from the low-energy regions populated by the experimentally-known structures. Energy barriers between such regions are less likely to be sampled; while the selection mechanism favors further populating such regions once a few samples are obtained in them, the variation (and structure correction) operator drive samples from ridges down to the valleys in the landscape.

2.2.4 Interplay between Selection, Variation, and Initialization

The leveraging of structures is key to an effective initialization mechanism that provides `SoPriM` with a non-local view of the structure space. However, the structures also tilt the computational budget towards exploitation more than exploration, as they lie in local minima from which it takes `SoPriM` many iterations to climb out with the variation (and structure correction) operator. The result of this tilted scale is that many iterations are needed to obtain samples on the possible ridges in the energy landscape that are crucial to connect two energy basins housing the start and goal structures sought to be connected. Even if the ridges are sampled, samples will be scarce and disproportionately reside in valleys in the landscape.

Sampling impacts the quality of the path(s) that can be offered to model the transition between two given start and goal structures. Typically, after the sampling stage is terminated (exhausting a fixed computational budget or reaching some other termination criterion), the samples are embedded in a nearest-neighbor graph, where each sample in Ω is connected to its k nearest neighbors in Ω. If the start and goal structures are in a connected component, paths can be found. Moreover, a cost $c(u, v)$ can be associated with a directed edge (u, v) to then obtain a lowest-cost path via shortest path algorithms. In `SoPriM`, $c(u, v) = \max\{E(v) - E(u), 0\}$. This cost implements the concept of work; only uphill moves in the landscape are recorded.

Finding paths is not a measure of success. Indeed, any setting of k (even if a range r is considered to remove edges connecting structures beyond r units in the structure space) can be employed to obtain a connected graph so that path queries can be answered. A deeper inspection of these paths will betray limited sampling on the ridges. Longer edges will disproportionately be found connecting the scarce samples on the high-energy regions crossed by a structural transition. Moreover, reported path costs may be optimistic, as undersampling effectively hides hills (tunneling through them). More samples would reveal the actual ruggedness and possibly increase the cost of a path.

2.3 Leveraging the Initialization Mechanism for Exploration-Exploitation Trade-off

Sampling-based algorithms like `SoPriM` delegate path quality to the sampling stage. Uniformly-dense sampling is generally very challenging to guarantee, particularly considering the high dimensionality of the protein structure space. Moreover, the quality of sampling depends on the exploration-exploitation trade-off, which, as described above, is affected by the interplay between the selection mechanism, the variation operator(s), and the initialization mechanism. We propose to leverage the initialization mechanism to improve the quality of sampling and, in turn, the quality of paths modeling structural transitions. We describe two strategies to do so. We refer to `SoPriM` with either of these strategies as `SoPriMp` ('p' standing for paths) and `SoPriMo` ('o' standing for orthogonal paths).

2.3.1 SoPriMp: Structures Along Direct Paths

The key idea is to initialize the sampling stage with additional structures. The experimentally-known structures are likely to reside in basins; generating structures on direct paths connecting basins would seed sampling with samples likely to reside on or near ridges in the landscape. This is implemented as follows. The known structures are grouped; clustering can be used, but here we rely on visualization over PC1-PC2 projections. Only a few structures are used per group/state. These can be canonical structures (other criteria can be used). For every structure u in group U and every structure v in group V, the normalized vector $\hat{uv}$ is defined in the m-dimensional space. A new sample $u' = u + \delta_{max} \cdot \hat{uv}$ is first generated. The sample is mapped to an all-atom structure via the structure correction operator, projected back to the variable space to obtain u'^*, and the process is repeated, using the normalized vector $\hat{u'^*v}$ from u'^*. This continues until either the structure correction fails (too many deformations have been accumulated), or the current structure is less than δ_{max} away from v. When no more advances can be made toward v, the reverse direction vu is attempted. Figure 1(b) shows the experimentally-known structures and the additional ones obtained as described projected onto the top two PCs and color-coded by their Amber ff14SB energies.

2.3.2 SoPriMo: Structures Along Orthogonal Paths

The additional initial structures are now generated exploiting ideas from the Conjugate Peak Refinement algorithm [15], where it is assumed that the saddle point along a direct (straight line) path has the highest energy relative to those along all other paths connecting two minima of in-

terest; thus it follows that the orthogonal directions from the saddle point may be the shortest way to find other low-energy regions. SoPriMo first invokes SoPriMp to obtain all intermediate structures between structure pairs u and v. For a given pair, the highest-energy intermediate structure uv^h is recorded. The initial structures added by SoPriMo to the ensemble Ω (in addition to the experimentally-known structures) are obtained by modifying uv^h along vectors orthogonal to $\hat{uv}$ at uv^h; these are limited to the PC1-PC2 and PC1-PC3 planes, as these three dimensions contain most of the structural variation in investigated systems (more planes can be used). In addition, the magnitudes of the orthogonal vectors are set to that of the uv vector. New structures along an orthogonal vector are then generated (at increments of δ_{max}) until structure deformations cannot be corrected or the length limit has been reached. The structures generated by SoPriMo are shown in Figure 1(c).

2.4 Implementation Details and Setup

SoPriM, SoPriMp, and SoPriMo are only different in the initial structures they use to initialize the Ω ensemble before the sampling stage begins. In SoPriMp and SoPriMo, more initial structures are added to the set of experimentally-known ones, computed as described above. The sampling stage in each proceeds until Ω contains $3,000$ structures. Under each algorithm, sampling is repeated a total of 15 times, 5 times for each value of δ_{max} varied in $\{1.0, 2.0, 3.0\}$. The structures obtained from all 15 runs of an algorithm are collected and compared across the three algorithms. The structures collected for an algorithm are then embedded in a nearest-neighbor graph, where a structure is connected to at most $k = 50$ nearest neighbors; the neighbors are additionally restricted to be no more than rÅ away in the structure space. Different values are considered for r (from 0.250 to 0.1Å in least root-mean-squared-deviation – lRMSD – over the CA atoms), and the lowest-cost path obtained at each value of r is extracted and compared among the three algorithms.

A detailed, comparative evaluation is conducted over the 166 amino-acid long wildtype sequence of H-Ras. This enzyme is central to human biology and is known to switch between different structures to regulate recognition of partners. Structure switching spans 2.5Å all-atom lRMSD and 1.5Å CA lRMSD. H-Ras has been studied by various laboratories, including ours; in particular, H-Ras has been employed to test the baseline capability of SoPriM, and details regarding how many known structures are collected and the PCs obtained can be found in [22]. Implementation of the algorithms is carried out in C/C++, and testing is done on Intel Xeon E5-2670 2.6GHz CPU nodes with 3.5TB of RAM. Running times vary from 24 to 48 hours on one CPU.

3. RESULTS

We first compare the algorithms on the quality of their sampling of the H-Ras structure space, and then on the quality of the lowest-cost path they obtain at different values of r to model the active to inactive structure switching in H-Ras.

3.1 Comparison of Quality of Sampling

3.1.1 *Visual Comparison over 2D Embeddings*

The Ω ensembles generated by each algorithm can be compared via two-dimensional (2D) embeddings over selected pairs of PCs. Projections of the structures are color-coded

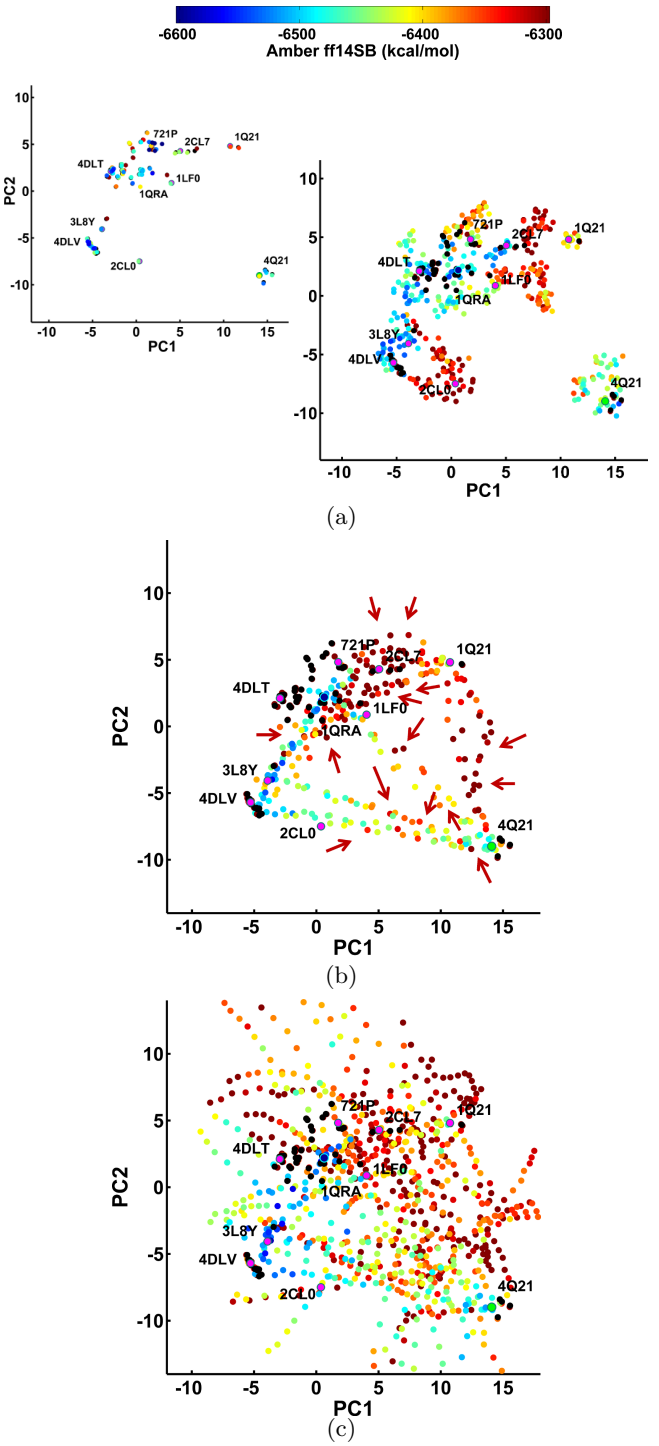


Figure 1: Color-coding scheme is shown at the top. (a) shows PC1-PC2 projections of structures that initialize SoPriM on the left and the additional samples obtained after 500 iterations of sampling on the right. (b) and (c) show projections of additional structures generated by SoPriMp and SoPriMo, respectively, to initialize sampling. Arrows point to structures of highest energy at which orthogonal vectors are computed. Experimentally-known structures are drawn as black dots. (a)-(c) show data on H-Ras, where top two PCs capture $> 50\%$ of the dynamics.

based on their Amber ff14SB energies. All pairs of the 10 PCs that capture more than 90% of the variance among the collected experimentally-known structures of H-Ras are considered, but only two pairs, PC1-PC2 and PC1-PC4, are selected to relate representative observations regarding differences in sampling attributed to the different initialization strategies. The energy color-coded embeddings effectively show 2D, sample-based representations of the energy landscape as put together by the three different algorithms. The embeddings are shown in Figure 2.

The top row in Figure 2 compares the PC1-PC2 embeddings. All three algorithms largely agree on the location of the lowest-energy regions, also referred to as basins here, associated with the active (left) and the inactive (right) structural states of H-Ras. The majority of the experimentally-known structures, drawn as black dots with four-letter code identifiers, project on these basins, suggesting the algorithms reasonably reconstruct the H-Ras energy landscape. Differences can be observed. `SoPriM` underperforms `SoPriMp` and `SoPriMo` in sampling low-energy structures between the two basins. This is due to the fact that `SoPriM` overly exploits the regions populated by the experimentally-known structures and needs many iterations to escape those basins, whereas `SoPriMp` and `SoPriMo` are provided with more diverse regions for exploitation from the initialization mechanism.

Comparison of the PC1-PC4 embeddings confirms that the initialization mechanism in `SoPriM` leads to under-exploring regions between basins. Few structures are sampled along PC4 that would allow connecting the different low-energy regions. In contrast, `SoPriMp` and `SoPriMo` are able to explore this region, and show that the region is of high energy and represents transition barriers between subregions of the active basin and the inactive basin. `SoPriMo` generates fewer structures than `SoPriMp` in this regions (see subregion around known structure with identifier 4Q21).

3.1.2 Comparison of Exploration Capability

We now compare the three algorithms on their exploration capability by showing the number of structures they generate in each cell of the PC1-PC2 grid used by the selection mechanism in each algorithm. Figure 3 color-codes cells by their population, using the same color-coding scheme for all three algorithms, effectively relating density maps. Figure 3 shows that `SoPriM` focuses its exploration to regions containing the experimentally-known structures, as these are the ones initializing its exploration. In contrast, `SoPriMp` has more cells of high density. `SoPriMo` samples away from the experimentally-known structures, as its tendency is to explore directions orthogonal to those connecting the experimentally-known structures. A visual comparison among the three algorithms in terms of the amount of green to red cells (cells of high density) suggests `SoPriMp` and `SoPriMo` have higher exploration capability than `SoPriM`; they explore more of the structure space than `SoPriM`.

Simple statistics support these observation. In `SoPriM`, the median number of samples in cells populated by the initial (experimentally-known) structures is 115, whereas the median over all populated cells is only 28. As also related in Figure 2, cells with very few samples in them lie between the active and inactive basins; populating them is crucial to model structural between-basin transitions. Figure 3 shows that, in contrast to the decidedly non-uniformly dense sampling in `SoPriM`, `SoPriMp` performs better in a compact area encompassing the known structures, and `SoPriMo` covers the structure space more broadly due to its initialization (`SoPriMo` populates 11% more cells in PC1-PC2 grid and 30% more cells in the PC1-PC4 grid over `SoPriM`). In particular, the median number of samples over all populated cells for `SoPriMp` and `SoPriMo` is 39, higher than that of `SoPriM`.

Taken together, these results suggest that both `SoPriMp` and `SoPriMo` have higher exploration capability over `SoPriM`. Moreover, `SoPriMp` uniformly fills and expands the regions of interest outlined in the initialization, whereas `SoPriMo`, by exploring also orthogonal directions, considers more of the structure space. While `SoPriMp` seems better equipped to produce transitions between experimentally-known structures, `SoPriMo` may discover unknown minima or confirm their non-existence (the latter is the case for H-Ras here).

3.1.3 Comparison of Exploitation Capability

We compare the algorithms on their exploitation capability by relying on a hexagonal discretization of the PC1-PC2 embedding of the structure space (based on statistical graphics research showing such binning is more robust [8,9]). The lowest energy over structures projecting to a cell is recorded for each algorithm, and differences between such values for corresponding cells are calculated. Figure 4 color-codes cells based on such differences, to show `SoPriMp` − `SoPriM` in the top row (the lowest energy reached by `SoPriM` in a cell is subtracted from the lowest energy reached by `SoPriMp` in the same cell), `SoPriMp` − `SoPriMo` in the second row, and `SoPriMp` − `SoPriMo` in the third row. These exploitation maps shows that both `SoPriMp` and `SoPriMo` populate the structure space with much lower-energy structures over `SoPriM`, and so have higher exploitation capability. Due to its initialization, `SoPriM`'s higher exploitation capability is limited to regions populated by the experimentally-known structures. The third row in Figure 4 shows that `SoPriMp` has higher exploitation capability than `SoPriMo`. These observations are confirmed in Table 1, which shows counts of the number of populated cells with negative, equivalent (within 10 kcal/mol), or positive differences.

Table 1: Counts of hexagonal cells with different categories of lowest-energy differences.

Comparison	$<$	$\sim$	$>$	populated by both
`SoPriMp - SoPriM`	363	70	53	486
`SoPriMo - SoPriM`	283	94	113	490
`SoPriMp - SoPriMo`	323	57	143	523

3.2 Comparison of Quality of Lowest Cost Path

The three algorithms are now compared on the quality of the lowest-cost path they find at different range values r to connect a canonical, representative structure of the active state in H-Ras (identifier 1QRA) to a canonical, representative of the inactive state (4Q21); note that r corresponds to the maximum allowed edge length. Table 2 shows that sampling in `SoPriM` is not dense enough to be able to obtain a connected graph at values lower than 0.250Å, whereas `SoPriMo` only fails at the lowest value of 0.1Å. Even when all algorithms report a path at a given value of r (note that r corresponds to the maximum allowed edge length), the average and median edge lengths ($\langle el \rangle$, $\tilde{el}$) in `SoPriM` are higher than those in `SoPriMp` and `SoPriMo`, indicating sparser sampling in `SoPriM`. In addition, the maximum, average, and

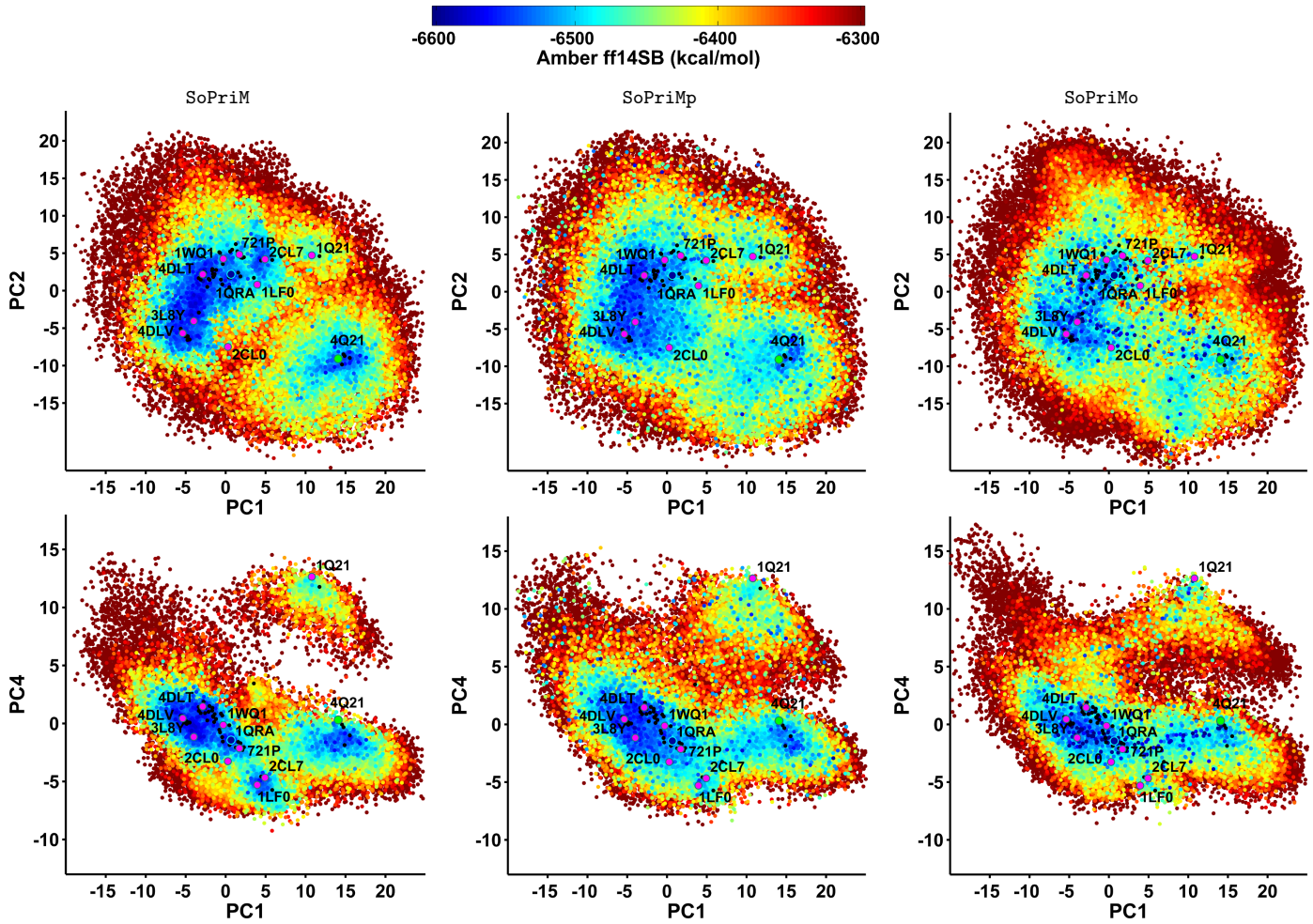


Figure 2: The algorithms are compared visually, by projecting samples on PC1-PC2 and PC1-PC4. Black dots indicate experimentally-known structures . The color coding, which follows the Amber ff14SB values, is shown at the top.

median edge costs ($ec_{\max}$, $\langle ec \rangle$, $\tilde{ec}$) along the lowest-cost path are also higher in `SoPriM` over `SoPriMp` and `SoPriMp` at a given value of r, indicating that the better exploration and better exploitation in the latter two algorithms provide alternative routes with both shorter and less costly edges. While path costs initially go down at lower values of r, indicating a phase where less costly routes are found, at the smallest values possible to find paths, as in 0.124 and 0.100Å, the path cost goes up. What occurs in this case is that insisting that edges be short forces a path to go over small hills in the landscape, and thus follow the ruggedness much more closely, resulting in higher cost.

Figure 5 shows the lowest-cost path generated by each algorithm at 0.250Å, the lowest-cost path generated by two of the algorithms, `SoPriMp` and `SoPriMo` at the lowest possible value of r to do so, and then the lowest-cost path generated only by `SoPriMp` at the lowest value of r where a path can be found, 0.1Å. We draw attention to the fact that at 0.250Å, all three algorithms report a similar path in structure space, but only `SoPriMp` and `SoPriMo` achieve lower costs due to their better exploration. When r decreases to 0.125Å, `SoPriMo` is unable to find a path along the same route as before and instead reports a suboptimal one, as well, in terms of cost. Only `SoPriMp` is able to largely retain the structural characteristics of the path at these different r values and even at the lowest value of 0.100Å, though the cost increases due to the path following the landscape much more closely.

4. CONCLUSION

The results presented here demonstrate that the initialization mechanism has a great influence on the quality of sampling and paths reported by robotics-inspired sampling-based methods. Strategies inspired by the principle of conformational selection and the conjugate peak refinement algorithm are presented here to tilt the scale towards more exploration in the exploration-exploitation trade-off. The analysis demonstrates that such strategies (which can even be combined), interleaved with exploration-driven selection mechanisms and exploitation-driven variation operator(s), improve both exploration and exploitation

As demonstrated, improvements in sampling translate to paths of higher granularity that follow the energy landscape more faithfully. The paths shown to model the active to inactive structural transition in H-Ras additionally suggest that great care has to be taken to provide dense sampling; paths change both structurally and energetically depending

Table 2: Comparison of lowest-cost paths at varying r in terms of average and median edge lengths ($\langle el \rangle$, $\tilde{el}$) and maximum, average, median edge costs ($ec_{\max}$, $\langle ec \rangle$, $\tilde{ec}$), nr. of vertices and overall path cost.

r	SoPriM edge stats $\langle el \rangle$, $\tilde{el}$	$ec_{\max}$, $\langle ec \rangle$, $\tilde{ec}$	#vert	cost
0.250	0.14, 0.15	33.1, 2.90, 6.37	31	129.92

r	SoPriMp edge stats $\langle el \rangle$, $\tilde{el}$	$ec_{\max}$, $\langle ec \rangle$, $\tilde{ec}$	#vert	cost
0.250	0.17, 0.17	15.91, 2.45, 3.17	28	83.61
0.201	0.13, 0.14	15.91, 1.51, 2.32	40	86.10
0.167	0.14, 0.14	15.91, 1.63, 2.15	46	89.16
0.124	0.11, 0.11	32.80, 2.52, 4.21	41	126.95
0.100	0.09, 0.09	56.80, 6.56, 8.33	49	240.6

r	SoPriMo edge stats $\langle el \rangle$, $\tilde{el}$	$ec_{\max}$, $\langle ec \rangle$, $\tilde{ec}$	#vert	cost
0.250	0.19, 0.19	18.29, 4.36, 4.60	20	84.44
0.201	0.18, 0.17	43.71, 1.89, 4.46	22	86.40
0.167	0.15, 0.14	37.25, 2.41, 5.45	29	117.47
0.124	0.11, 0.11	56.15, 5.26, 9.56	49	274.49

on the restrictions imposed on the graph-based models embedding samples to model dynamics.

While this paper has focused on sampling and its influence on paths, it is worth noting that the structurally-persistent lowest-cost paths obtained by SoPriMp at varying r suggest that the known structure with identifier 1LF0 is an intermediate in the transition. This structure has been suggested as a possible intermediate, but it has only been isolated in wet laboratories as a stable structure of the A59G variant and not the wildtype. Further observations of interest to wet-laboratory investigations can be made from visualization of the landscape and paths, but the focus in this paper is on improving current understanding of the capabilities and limitations of sampling-based methods for protein structure transition modeling. We note that related questions regarding efficacy of sampling and its effect on the accuracy of modeled transitions are being raised among computational biophysicists embedding structures obtained from many MD trajectories in Markov state models [16]. The analysis and strategies proposed in this paper to expose and address current limitations are a first step towards making robotics-inspired sampling-based methods reliable tools for modeling protein structural transitions.

5. ACKNOWLEDGMENT

This work is supported in part by NSF SI2 No. 1440581 and NSF IIS CAREER Award No. 1144106. Computations were run on the ARGO research computing cluster at George Mason University.

6. REFERENCES

[1] I. Al-Bluwi, T. Siméon, and J. Cortés. Motion planning algorithms for molecular simulations: A survey. *Computer Science Reviews*, 6(4):125–143, 2012.

[2] I. Al-Bluwi, M. Vaisset, T. Siméon, and J. Cortés. Modeling protein conformational transitions by a

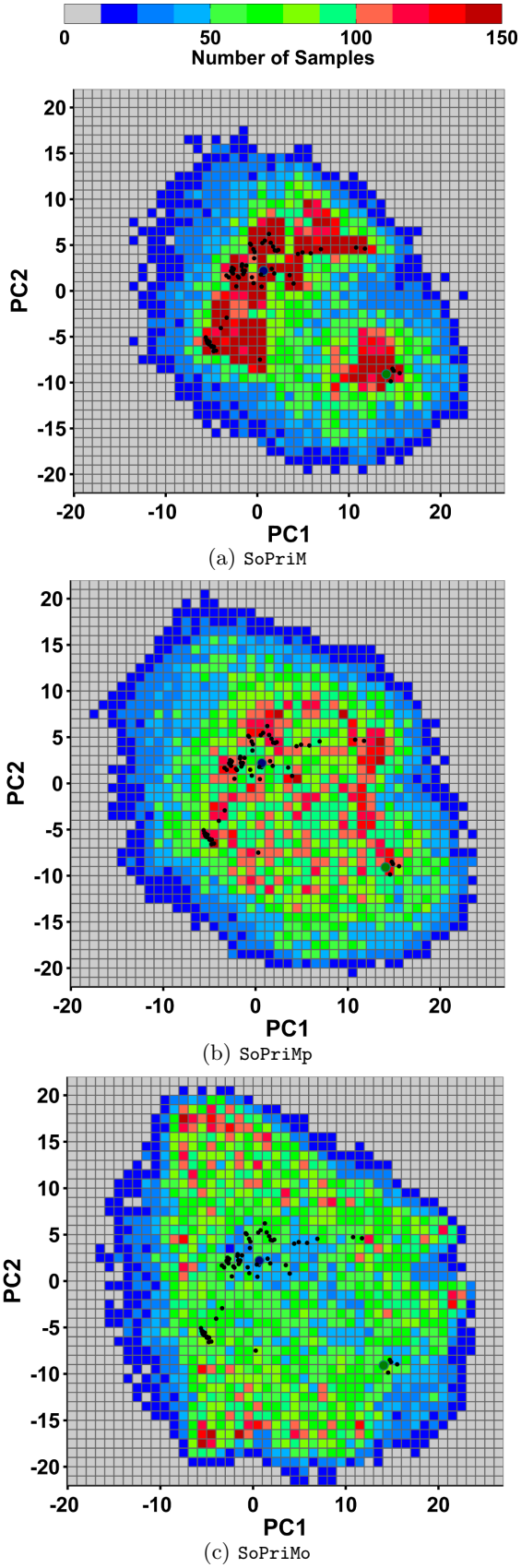
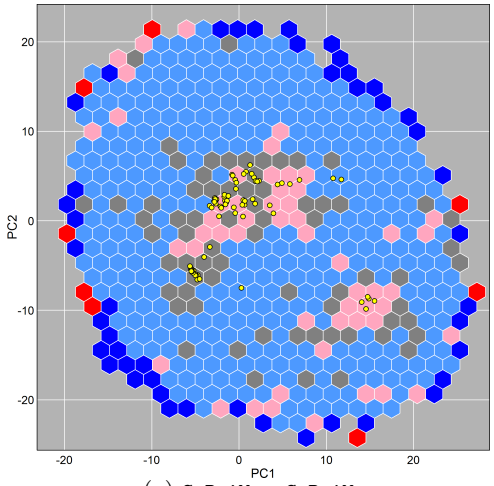
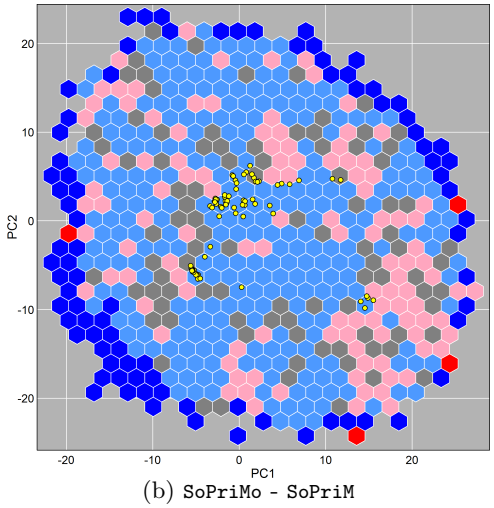


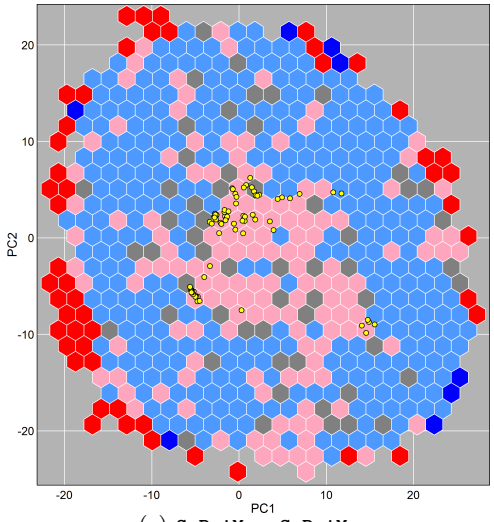
Figure 3: PC1-PC2 grid cells are color-coded (scheme shows at top) based on the number of samples per cell. Experimentally-known structures are drawn as black dots. Cell width corresponds to 0.08Å in CA lRMSD.



(a) SoPriMp - SoPriM

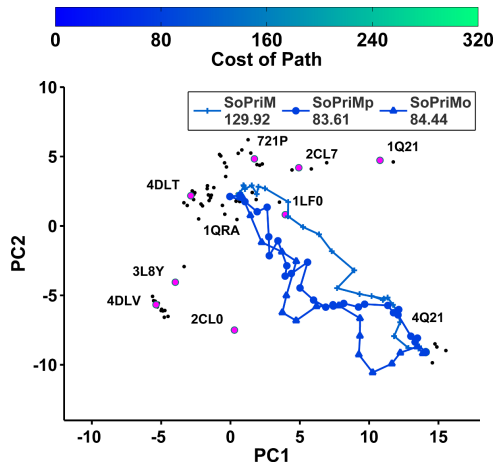


(b) SoPriMo - SoPriM

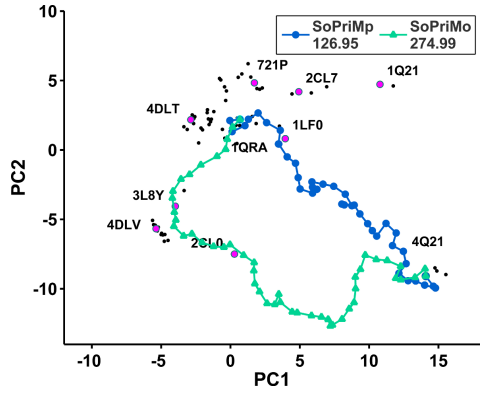


(c) SoPriMp - SoPriMo

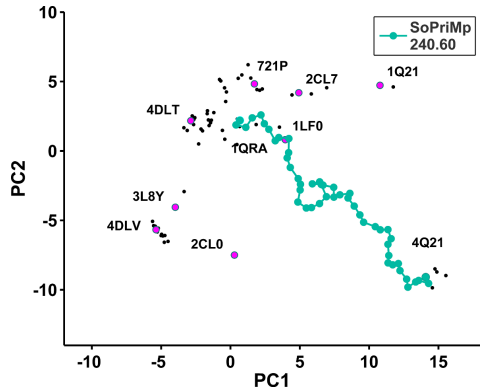
Figure 4: Hexagonal cells are color-coded based on the difference between the lowest-energy sample in the cell generated by the algorithms under comparison. Cells with differences $\leq$ 10 kcal/mol are in light blue, those with differences in $(-10, 10)$kcal/mol are in gray, and those with higher differences are in light pink. In an A $-$ B comparison, dark blue cells indicate those unpopulated by algorithm B, and dark red cells indicate those unpopulated by algorithm A. Experimentally-known structures are drawn as yellow dots.



$r = 0.250$



$r = 0.125$



$r = 0.100$

Figure 5: The lowest-cost path returned by each algorithm as r is varied is drawn in the PC1-PC2 embedding. The color-coding scheme is shown at the top.

combination of coarse-grained normal mode analysis and robotics-inspired methods. *BMC Struct. Biol.*, 13(Suppl 1):S8, 2013.

[3] R. E. Amaro and M. Bansai. Editorial overview: Theory and simulation: Tools for solving the insolvable. *Curr. Opinion Struct. Biol.*, 25:4–5, 2014.

[4] 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. *J. Comp. Biol.*, 10(3-4):239–255, 2002.

[5] 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. *J. Comp. Biol.*, 10(3-4):257–281, 2003.

[6] M. S. Apaydin, A. P. Singh, D. L. Brutlag, and J.-C. Latombe. Capturing molecular energy landscapes with probabilistic conformational roadmaps. *Proc. IEEE Int. Conf. Robot. Autom.*, 1:932–939, 2001.

[7] D. D. Boehr, R. Nussinov, and P. E. Wright. The role of dynamic conformational ensembles in biomolecular recognition. *Nature Chem Biol*, 5(11):789–96, 2009.

[8] D. B. Carr. Looking at large data sets using binned data plots. In A. Buja and P. Tukey, editors, *Computing and Graphics in Statistics*, pages 7–39. Springer-Verlag, New York, New York, 1991.

[9] D. B. Carr. Scanning a 4-D domain for local minima: A protein folding example. *Topics in Scientific Visualization*, 6(2):9–12, 1995.

[10] T. H. Chiang, M. S. Apaydin, D. L. Brutlag, D. Hsu, and J.-C. Latombe. Using stochastic roadmap simulation to predict experimental quantities in protein folding kinetics: folding rates and phi-values. *J. Comp. Biol.*, 14(5):578–593, 2007.

[11] H. Choset et al. *Principles of Robot Motion: Theory, Algorithms, and Implementations*. MIT Press, Cambridge, MA, 1st edition, 2005.

[12] J. Cortés et al. A path planning approach for computing large-amplitude motions of flexible molecules. *Bioinformatics*, 21(S1):116–125, 2005.

[13] C. Dellago, P. G. Bolhuis, and P. L. Geissler. Transition path sampling methods. In *Computer Simulations in Condensed Matter Systems: From Materials to Chemical Biology*, volume 703 of *Lecture Notes in Physics*, pages 349–391. Springer, 2005.

[14] D. Devaurs, K. Molloy, M. Vaisset, and A. Shehu. Characterizing energy landscapes of peptides using a combination of stochastic algorithms. *IEEE Trans. NanoBioSci.*, 14(5):545–552, 2015.

[15] S. Fischer and M. Karplus. Conjugate peak refinement: an algorithm for finding reaction paths and accurate transition states in systems with many degrees of freedom. *Chem. Phys. Lett.*, 194(3):252–261, 1992.

[16] B. Gipson, D. Hsu, L. E. Kavraki, and J. Latombe. Computational models of protein kinematics and dynamics: Beyond simulation. *Annu. Rev. Anal. Chem.*, 5:273–291, 2012.

[17] N. Haspel, M. Moll, M. L. Baker, W. Chiu, and L. E. Kavraki. Tracing conformational changes in proteins. *BMC Struct. Biol.*, 10(Suppl1):S1, 2010.

[18] L. Jaillet, F. J. Corcho, J.-J. Perez, and J. Cortés. Randomized tree construction algorithm to explore energy landscapes. *J. Comput. Chem.*, 32(16):3464–3474, 2011.

[19] L. Jaillet, J. Cortés, and T. Siméon. Transition-based RRT for path planning in continuous cost spaces. In *IEEE/RSJ Int. Conf. Intel. Rob. Sys.*, pages 22–26, Stanford, CA, 2008. AAAI.

[20] M. Karplus, Y. Q. Gao, J. Ma, A. van der Vaart, and W. Yang. Protein structural transitions and their functional role. *Philos. Trans. A Math. Phys. Eng.*, 363(1827):331–355, 2005.

[21] T. Maximova, R. Moffatt, B. Ma, R. Nussinov, and A. Shehu. Principles and overview of sampling methods for modeling macromolecular structure and dynamics. *PLoS Comp. Biol.*, 12(4):e1004619, 2016.

[22] T. Maximova, E. Plaku, and A. Shehu. Computing transition paths in multiple-basin proteins with a probabilistic roadmap algorithm guided by structure data. In *IEEE Intl. Conf. Bioinf. and Biomed.*, pages 35–42, 2015.

[23] T. Maximova, E. Plaku, and A. Shehu. Structure-guided protein transition modeling with a probabilistic roadmap algorithm. *IEEE/ACM Trans. Bioinf. and Comp. Biol.*, 2016. in press.

[24] K. Molloy, R. Clausen, and A. Shehu. A stochastic roadmap method to model protein structural transitions. *Robotica*, 34(8):1705–1733, 2016.

[25] K. Molloy, S. Saleh, and A. Shehu. Probabilistic search and energy guidance for biased decoy sampling in ab-initio protein structure prediction. *IEEE/ACM Trans. Bioinf. and Comp. Biol.*, 10(5):1162–1175, 2013.

[26] K. Molloy and A. Shehu. Elucidating the ensemble of functionally-relevant transitions in protein systems with a robotics-inspired method. *BMC Struct. Biol.*, 13(Suppl 1):S8, 2013.

[27] K. Molloy and A. Shehu. A general, adaptive, roadmap-based algorithm for protein motion computation. *IEEE Trans. NanoBioSci.*, 2(15):158–165, 2016.

[28] B. Olson, I. Hashmi, K. Molloy, and A. Shehu. Basin hopping as a general and versatile optimization framework for the characterization of biological macromolecules. *Advances in AI J*, (674832), 2012.

[29] D. Russel et al. The structural dynamics of macromolecular processes. *Curr. Opinion Cell. Biol.*, 21(1):97–108, 2009.

[30] A. Shehu and B. Olson. Guiding the search for native-like protein conformations with an ab-initio tree-based exploration. *Intl. J. Robot. Res.*, 29(8):1106–1127, 2010.

[31] A. P. Singh, J.-C. Latombe, and D. L. Brutlag. A motion planning approach to flexible ligand binding. In R. Schneider, P. Bork, D. L. Brutlag, J. I. Glasgow, H.-W. Mewes, and R. Zimmer, editors, *Proc Int Conf Intell Sys Mol Biol (ISMB)*, volume 7, pages 252–261, Heidelberg, Germany, 1999. AAAI.

[32] X. Tang, S. Thomas, L. Tapia, D. P. Giedroc, and N. Amato. Simulating rna folding kinetics on approximated energy landscapes. *J. Mol. Biol.*, 381(4):1055–1067, 2008.

[33] L. Tapia, X. Tang, S. Thomas, and N. Amato. Kinetics analysis methods for approximate folding landscapes. *Bioinformatics*, 23:i539–i548, 2007.

[34] L. Tapia, S. Thomas, and N. Amato. A motion planning approach to studying molecular motions. *Commun Inf Sys*, 10(1):53–68, 2010.

[35] S. Thomas, G. Song, and N. M. Amato. Protein folding by motion planning. *J. Phys. Biol.*, 2(4):148, 2005.

[36] S. Thomas, X. Tang, L. Tapia, and N. M. Amato. Simulating protein motions with rigidity analysis. *J. Comput. Biol.*, 14(6):839–855, 2007.