Automatic Feature Selection in Markov State Models Using Genetic Algorithm

Qihua Chen*
University of Illinois at Urbana-Champaign
Urbana, IL
Qihua.chen06@gmail.com

Jiangyan Feng*
University of Illinois at Urbana-Champaign
Urbana, IL
jf8@illinois.edu

Shriyaa Mittal
University of Illinois at Urbana-Champaign
Urbana, IL
smittal6@illinois.edu

Diwakar Shukla†
University of Illinois at Urbana-Champaign
Urbana, IL
diwakar@illinois.edu

ABSTRACT

Markov State Models (MSMs) are a powerful framework to reproduce the long-time conformational dynamics of biomolecules using a set of short Molecular Dynamics (MD) simulations. However, precise kinetics predictions of MSMs heavily rely on the features selected to describe the system. Despite the importance of feature selection for large system, determining an optimal set of features remains a difficult unsolved problem. Here, we introduce an automatic approach to optimize feature selection based on genetic algorithms (GA), which adaptively evolves the most fitted solution according to natural selection laws. The power of the GA-based method is illustrated on long atomistic folding simulations of four proteins, varying in length from 28 to 80 residues. Due to the diversity of tested proteins, we expect that our method will be extensible to other proteins and drive MSM building to a more objective protocol.

KEYWORDS

Genetic algorithm, feature selection, markov state model, molecular dynamics simulation, generalized matrix Rayleigh quotient

1 INTRODUCTION

Molecular Dynamics (MD) simulation, first introduced by Alder and Wainwright[2] in the late 1950's, has evolved into a major technique to study the detailed actions and mechanisms of proteins[10, 23, 26, 35, 39]. Based on Newton's equations of motion, MD simulations can describe protein dynamics in unprecedented spatial and temporal resolution. However, one of the major challenges for MD simulations are the analysis of high dimensional data and the incompatibility between timescales accessible to MD simulation and that are functionally relevant[22, 25, 45, 46, 50]. Markov State Models (MSMs)[20, 37, 44] have recently been used to address the aforementioned issues by predicting protein dynamics at long timescales from a pool of short MD simulations. The MSM itself is a "transition probability matrix"[6], describing mathematically the memoryless transitions between metastable states. To construct a MSM, raw MD trajectories are first transformed from their Cartesian coordinates to features, such as dihedral angles[18, 33] or pairwise contact distances of a protein. This step is often called "featurization". The dimensionality of these features may be further reduced through dimensionality reduction step. One commonly used method is time-structure independent components analysis (tICA), which creates linear combinations of input features by maximizing their decorrelation time[24, 27, 38, 41, 42]. With a properly constructed MSM, useful thermodynamic and kinetic properties of the dynamic process can be extracted. Despite the attractive feature of MSMs, the thermodynamics and kinetics predicted by MSMs are highly sensitive to which features are selected to discretize the configuration space[4, 10, 28]. Ideally, features should be chosen to capture the slowest motions of the protein, which are usually the most interesting or important processes. However, determining an optimal set of features remains a considerable challenge especially when a protein system is sufficiently complex.

Currently, there are two major ways of selecting features in terms of "contact featurization", where pairwise contact distances of a protein are used as features. One is using all pairwise contact distances of a protein as features. In principle, no important information about the system is missed out since all the contact distances are considered. However, it is costly to calculate all distances even for a small protein. For a protein system with R residues, the total number of distances among each other will be $R(R\text{-}1)/2$, which creates a heavy load of calculation on computers. In addition, irrelevant features that do not contribute to the dynamics process may lead to the poor generalization performance of the model. Thus, using all available features may degrade the performance of the MSM both in speed (due to high dimensionality) and accuracy (due to irrelevant information). Alternatively, the most commonly used method is choosing a subset of contact distances based on human intuition [20]. Consequently, the thermodynamics and kinetics extracted from MSMs can be biased by the manually chosen features. In summary, either way is not appropriate for the selection of features and a more convenient, accurate and automated method for feature selection is necessary. A variety of machine learning methods have been recently reported for dimensionality reduction and/or feature

[*] Qihua Chen and Jiangyan Feng contributed equally to this work.
[†] Correspondence and requests for materials should be addressed to D.S. (email: diwakar.shukla@shuklagroup.org)

© 2018 Journal of Computational Science Education
https://doi.org/10.22369/issn.2153-4136/9/2/2

selection for molecular dynamics datasets[33]. However, the use of these ideas for automatic feature selection in building MSMs has not been explored.

Here, we present a genetic-algorithm based method to select an optimal set of residue pair distances for contact featurization. Genetic algorithm (GA) is one of the advanced methods to help with dealing feature selection problems in data science. First proposed by John H. Holland[13, 15], GA is a heuristic and adaptive simulation algorithm that evolves the most fitted solution to a problem based on Darwinian natural selection laws. GA has been broadly applied to help with function optimization[48], protein folding prediction[21, 49], multiple sequence alignment[14] and more scientific investigations[16]. In nature, useful traits in genes tend to be preserved in offspring because of a higher survival probability. Like the real cases in nature, better solutions to a problem can be derived by GA according to this principal. In our case, each "gene" represents the alpha carbon distance between a residue pair, and "chromosomes" are combinations of residue pair distances. To seed the whole process, we randomly select one residue pair distance as the starting point of the GA. The adaptability of each "chromosome" (a set of residue pair distances) is quantitatively expressed as fitness scores in GA. In this study, we use generalized matrix Rayleigh quotient (GMRQ) score as the fitness score. GMRQ was recently introduced to quantitatively evaluate MSMs based on its distance from a theoretical upper limit[19, 30, 36]. The higher the GMRQ score is, the more prominent the MSM is to capture the slow underlying dynamical motions while a low GMRQ score indicates that the MSM is not able to reveal the slow dynamics of the system. Therefore, the goal of our method is optimizing a set of residue pairs that gives the highest GMRQ score. The framework of our GA-based method is adapted from the "Optimal Probes" method proposed by Mittal and Shukla[32]. In their study, an optimal choice of residue pairs, capturing the slow conformational dynamics, is successfully predicted for double electron-election resonance spectroscopy, an experimental technique capable of detecting conformational changes by monitoring the distance between electron spins.

In this method, we (1) perform contact featurization for each set of residue pair alpha carbon distances, (2) use tICA to further reduce the dimensionality of the data, (3) construct MSMs based on the reduced dimensionality, and (4) calculate GMRQ for each set of residue pair distances to evaluate the MSMs. Based on the GMRQ score, the combination of residue pair distances will be updated. The algorithm will then go back to step (1) to repeat the whole process until reaching user specified number of iterations. In the end, the set of distances with the maximum GMRQ score is chosen as an optimal set of residues for the construction of the "best MSM". To evaluate of our method, we test the GA-based method on four folding proteins with the size ranging from 28 residues to 80 residues. Our experimental results show that the method yields comparable and even better accuracy compared with using all available features. To our knowledge, this is the first attempt to automatically select proper MSM features for analysis. The GA-based method described here can be extended to larger proteins undergoing conformational changes.

2 THEORY AND METHODS

Molecular Dynamics (MD) Simulation Dataset. MD simulation datasets of the four folding proteins for analysis were generated by Lindorff-Larsen *et al*[26]. The four proteins (BBA, Villin, WW domain and λ-repressor) vary in length from 28 to 80 amino acids. More details of the simulations are summarized in Table 1. For the analysis, we retain all the trajectory frames. Three small proteins (BBA, Villin and WW domain) are chosen to evaluate the proposed method and the best GMRQ achieved using all contact distances serves as the benchmark. The 80-amino-acid λ-repressor is used to test the feasibility of the method on large proteins, as using all distances is impractical.

Table 1: Protein and Trajectory Information.

Protein	PDB	Residues	Total simulation time (μs)
BBA	1FME	28	325
Villin	2F4K	47	429
WW domain	2F21	35	1137
λ-repressor	1LMB	80	643

Markov State Models (MSMs). In this study, the goal is optimizing a set of residue pair distances to build the best MSM based GMRQ. MSMs are kinetic models that reveal the dynamics of a system[6, 17, 37, 39, 40]. An MSM describes a network of metastable conformational states and reveals the probabilities of each state performing jumps from one to another over an appropriate time resolution (τ, also called lag time). The jumps are memoryless, which means the probability to transit to the present state is not dependent on the previous ones. Such information is presented in a "transition probability matrix" by MSM, where an $n \times n$ square matrix depicts the transitions among n states[6]. The probability of each jump can be expressed according to the equation below:

$$p_j(t+\tau) = \sum_{i=1}^{n} p_i(t) T_{ij}(\tau) \tag{1}$$

The equation can also be expressed in a matrix form:

$$p^T(t+\tau) = p^T(t) T(\tau) \tag{2}$$

where $p_i(t)$ is a population vector whose elements show the probability at time t, $p_j(t+\tau)$ is a population vector after time τ, $T_{ij}(\tau)$ is the probability to jump from state i to state j and $T(\tau)$ is the transition probability matrix that $T(\tau) \in \mathbb{R}^{n \times n}$. Further details of the transition matrix can be found in literatures[6, 44].

The transition probability matrix can be decomposed into eigenfunctions and eigenvalues shown below:

$$T(\tau) \circ \psi_i = \lambda_i \psi_i \tag{3}$$

where ψ_i is the eigenfunction and λ_i are the real eigenvalues that $\lambda_i \leq 1$, arranged in descending order.

Here, each step of the MSM building process used in this study is described in detail. All the hyperparameters (e.g. the number of tICA components, tICA lagtime, the number of clusters, the number of MSM timescales and MSM lagtime) are shown in Table 2.

(1) Featurization. To construct an MSM, the first step is to process the datasets that we plan to work on. In our case, we use the MD simulation data sets listed in Table 1. The datasets are given in the form of MD trajectories, which present series of motion of the protein atoms in a frame-wise arrangement. Because the simulated movements recorded in Cartesian coordinates are not ideal for analysis, and too much noise not relevant to our study may be included, it is better to interpret the data in other ways. As a result, a lot of reasonable metrics such as dihedral angle[18] and contact distances between residue pairs are used to featurize the data. The featurization method we choose here is contact distance analysis. By using such technique, more useful information can be extracted from the redundant MD trajectories. Again, our goal of this study is to optimize the choice of residue pairs for contact distance calculation, so that an MSM with more information and less noise can be found by this method. The method outlined in this study could be applied to any chosen set of features calculated using simulation data.

(2) Dimensionality reduction. We further processed our featurized data by tICA so as to reduce the dimensionality of the data. After featurization, the featurized data were projected onto linear subspaces of the slowest dynamics. The components of tICA are termed time structure-based independent components (tICs), which are linear combination of the input features (a set of contact distances in our case). Top tICs capture the slowest motion captured by tICA and usually represent the most interesting dynamics[24, 27, 38, 41, 42].

(3) Clustering. We perform mini-batch k-means clustering on the processed data. Clustering refers to the coarse graining analysis that groups certain datasets based on their similarities, so that macrostates can be formed to be better understood. Commonly used clustering algorithms, such as mini-batch k-means[6, 30, 34], mini-batch k-medoids[8, 12] and k-centers[5, 24], have shown similar performance when the data is preprocessed with tICA[27, 33, 38, 41].

(4) MSM construction. After the clustering, a MSM can be built based on the processed datasets. The process was implemented in a Python environment and the software involved to produce the analysis above include Numpy[3], MDTraj[28] and MSMBuilder3.8[4].

Generalized Rayleigh Quotient (GMRQ). In short, an ideal MSM should successfully identify the slowest dynamics of the protein. Because the state decomposition mentioned above reveals the dynamical processes in the system, the identification of true eigenfunction and eigenvalues become the major problem for scientists to solve. A more quantitative method is needed to help evaluate and find the true state decomposition, which is directly related to the choice of metrics in the featurization stage.

To help solve this problem, GMRQ was introduced as a quantitative way of evaluating the quality of an MSM[30, 30, 36]. GMRQ is derived from the variational principle that adds up the first m eigenvalues, which denote the slowest m dynamical processes in the system. The variational principles set an upper boundary[19, 30, 36] for the total sum of real eigenvalues shown below:

$$GMRQ \equiv \sum_{i=1}^{m} \hat{\lambda}_i \leq \sum_{i=1}^{m} \lambda_i \tag{4}$$

where the $\hat{\lambda}_i$ is the estimated eigenvalue and the λ_i is the real eigenvalue. In this study, as we try to maximize GMRQ score to approach the upper boundary, the larger the GMRQ score we get, the closer we are to the slowest dynamics of the protein.

To help avoid overfitting, cross-validation must be applied to evaluate our GMRQ scores. The dataset from the MD simulation is split into a training set and a test set. The training set is first used to estimate the model parameters such as the eigenvalues, then the estimated model is applied to score its performance in the test set. In this way, the model will not be biased by overfitting the data onto the model. The process of deriving GMRQ scores is achieved by Osprey package[29] and the recruited parameters are shown in Table 2. Mean GMRQ of five cross-validation iterations are used for the analysis.

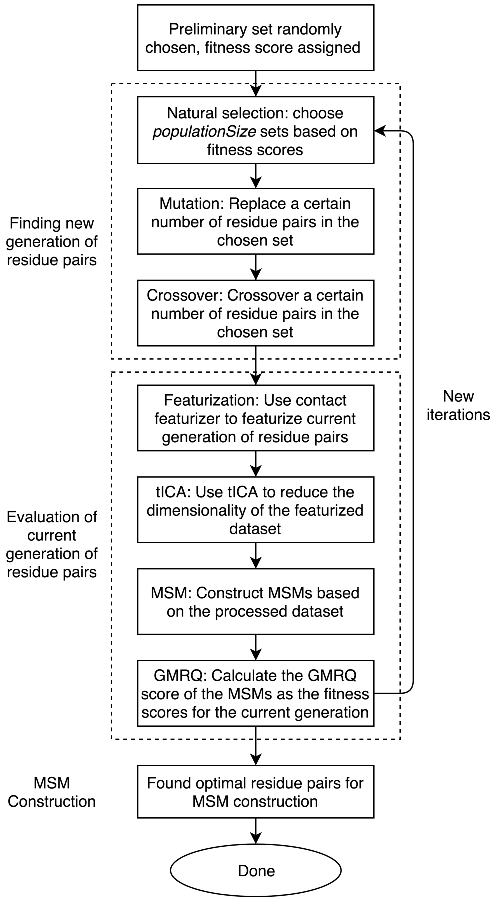
Preliminary set randomly chosen, fitness score assigned
Finding new generation of residue pairs
Natural selection: choose *populationSize* sets based on fitness scores
Mutation: Replace a certain number of residue pairs in the chosen set
Crossover: Crossover a certain number of residue pairs in the chosen set
New iterations
Evaluation of current generation of residue pairs
Featurization: Use contact featurizer to featurize current generation of residue pairs
tICA: Use tICA to reduce the dimensionality of the featurized dataset
MSM: Construct MSMs based on the processed dataset
GMRQ: Calculate the GMRQ score of the MSMs as the fitness scores for the current generation
MSM Construction
Found optimal residue pairs for MSM construction
Done

Figure 1: The flow chart showing the whole process of our GA-based method.

Genetic-algorithm-based Method for automatic feature selection in Markov State Models (MSMs). To simulate the natural selection process according to the Darwinian law, we must decide how the natural selection principles are implemented in our algorithm. In this section, we introduce our basic operators of GA, the framework that we follow to perform GA, and the protocol we adopt to finally generate optimal residue pairs. The construction of the GA is based on the work of Mittal and Shukla[32].

In the field of programming, operators refer to the actions to take during each step of execution of the algorithm. The basic operators in our study are composed of natural selection, mutation, and crossover. In the following section, we discuss our method to help predict an optimized set of residue pair distances for MSM construction using genetic algorithm. We also provide the series of steps as a flow chart shown in Figure 1. Some important parameters that are involved in these steps are: *populationSize*, *percentMutation* and *percentCrossover*. These parameters can be changed according to user's need.

(1) A set of all possible residue pairs is identified. $R(R\text{-}1)/2$ residue pairs for a protein with R residues.
(2) *populationSize* preliminary sets of residue pair are randomly selected from the set of all possible residue pairs for the first iteration. Each set contains only one residue pair as the starting point for selection. These sets of residue pairs serve as the initial generation G_0 and are assigned fitness scores of 0.
(3) Natural selection is performed to choose the new generation of residue pairs according to their fitness scores. The natural selection operator corresponds to the reproductive process in nature, which selects genomes with ideal traits for breeding offspring. In our case, we define a parameter *populationSize* that describes the number of elements randomly chosen from the parental set for a new generation G_{new}.
(4) Mutation is performed to maintain diversity to the current generation of residue pair selections. The mutation operator corresponds to the mutation process in nature to increase genetic diversity. In our version of GA, we define a parameter *percentMutation* to maintain a ratio of mutation in our combination of residue pairs. During the mutation step, the number of residue pairs to be mutated are generated by (*percentMutation* × *populationSize*)/100 from G_{new} and those residue pairs are randomly replaced by other residue pairs that are excluded in the G_{new}.
(5) Crossover is performed to add more diversity to the current generation. The crossover operator corresponds to the natural recombination process of chromosomes. Here, we define another parameter *percentCrossover* as the percentage of crossover in our combination of residue pairs. The number of residue pairs to perform crossover is generated by (*percentCrossover* × *populationSize*)/100 from G_{new}. The residue pair distance sets will then be swapped according to the number calculated before to create a new combination of residue pair distances.
(6) Evaluations are performed to assign fitness scores to the newly generated residue pairs. MSMs are constructed based on contact featurization using the current generation of residue pairs, and GMRQ scores are calculated accordingly to serve as fitness scores.
(7) If more iterations are designed to be finished, the next iteration should restart at step (3) and use the current generation of residue pairs as G_0. As the iteration number increases, the fitness scores for the selection of residue pairs should show a convergence of fitness scores.

All the parameters used in this study are organized in Table 2.

Table 2: Model Hyperparameteres.

Featurization		
α-carbon contact distances		
Decomposition	Components	Lag time (ns)
tICA	5	0.2
Clustering	Clusters	
Mini-batch k-means	200	
Model fitting	N_timescales	Lag time (ns)
MSM	5	50
Scoring		
GMRQ		
Cross-validation	Iterations	Test set size
Shuffle & Split	5	0.5
Genetic algorithm		
Iterations	40	
populationSize	20%	
percentMutation	50%	
percentCrossover	20%	

3 RESULTS

In this section, we discuss the optimized set of residue pair distances obtained from our GA-based approach. As described in the method part, the unbiased and extensive MD simulation data (>100μs) simulating the folding process of the proteins is taken from literature[26]. Preliminary sets of residue pair distance are randomly selected from the set of all the possible residue pairs as the starting point of the genetic algorithms. These sets go through selection, mutation and crossover steps to provide a new generation of residue pair distances. In the setting of GA, we choose a population size of 10%, mutation percentage of 50% and crossover percentage of 20%. Next, the newly generated residue pair distances are used to build MSMs and assign new GMRQ scores (fitness scores) for evaluation. The next iteration will then go back to the selection step and select according to the newly assigned fitness scores. As the process goes through more iterations, the GMRQ scores will converge and a best GMRQ score can be found.

This method is applied to 4 proteins for demonstration of its functionality: BBA, Villin, WW domain and λ-repressor. Among the proteins, 3 proteins (BBA, Villin and WW domain) are small proteins, each of which has a residue number that smaller than 40 (R < 40). To examine the effectiveness of our method, we compare the GMRQ scores and implied timescales with their corresponding benchmark values (using all contact distances as features). In the end, we show the ability of our GA-based method to process larger

Table 3: Comparison of the best GMRQ scores generated and benchmark GMRQ scores from all contact featurization. The fraction of all residue pairs is the fraction of chosen residue pairs in all residue pairs. The best GMRQ refers to the highest GMRQ score that we obtain from MSMs using residue pair distance features given by our genetic algorithm approach, and the benchmark GMRQ score is the GMRQ provided by the MSM constructed with all contact featurization. The deviation column if the deviation of our best GMRQ score from the benchmark GMRQ score.

Protein	Residues	Number of chosen distances	Fraction of pairs (%)	Best GMRQ	Benchmark GMRQ	Deviation (%)
BBA (1FME)	28	47	12.43	4.445	4.239	+4.80
Villin (2F4K)	35	61	10.25	3.203	3.705	-13.5
WW domain (2F21)	35	4	0.67	4.198	4.111	+2.12
λ-repressor (1LMB)	80	60	1.90	4.956	N/A	N/A

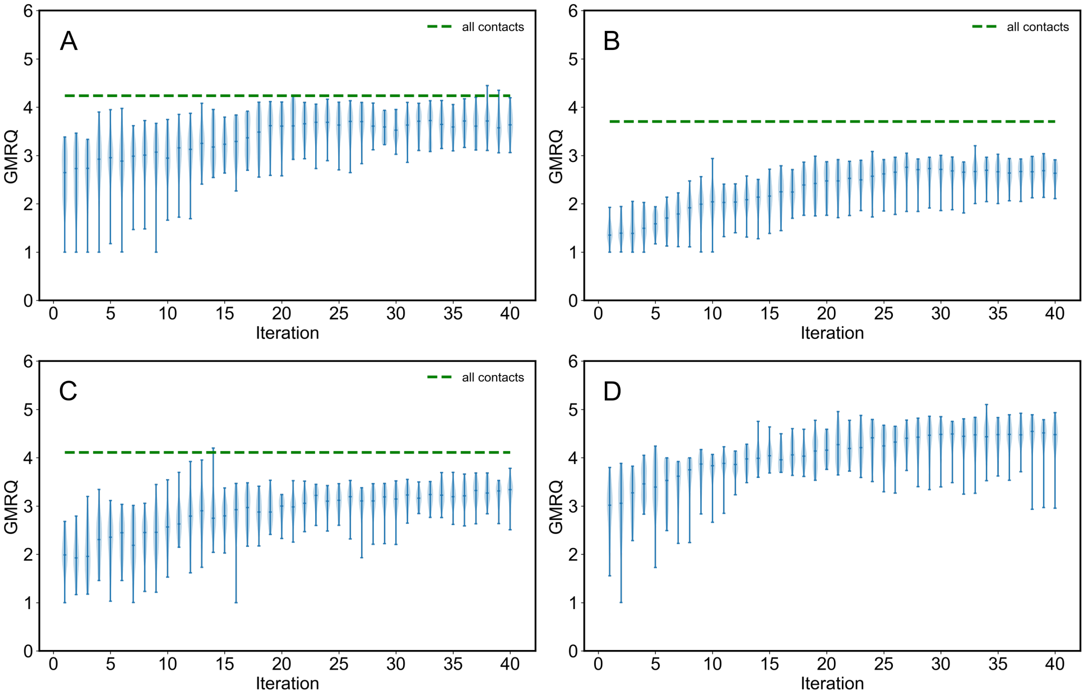

Figure 2: GMRQ scores reflecting the MSMs based on the GA-predicted residue pairs. (A) BBA (PDB ID: 1FME), (B) Villin (PDB ID: 2F4K), (C) WW domain (PDB ID: 2F21), (D) λ-repressor (PDB ID: 1LMB). Green, dashed lines indicate the best GMRQ score corresponding to MSMs based on all contact featurization. Each violin plot shows the increase of GMRQ scores over 40 iterations. In each set of data, the center dot shows the mean values and the vertical line shows the range of this GMRQ data set.

proteins such as λ-repressor, a protein with 80 residues, which cannot be featurized using all contact distances.

3.1 Our GA-based method proved effectiveness in generating GMRQ scores that are close to the highest possible values given by all contacts featurizer.

We featurize the small proteins (BBA, Villin, WW domain) using all contacts featurization to produce benchmark GMRQ scores for comparison. Benchmark GMRQ scores will serve as a comparable reference to evaluate the performance of our method of using GA to generate optimal residue pairs as featurization metrics. By comparing the best GMRQ scores from our GA-based method to the benchmark GMRQ scores, we are able to check whether our method successfully provides the residue pair sets that depict the slowest process of the protein dynamics. We also apply this method to λ-repressor, a medium sized protein with 80 residues, to show its ability to process larger proteins. The GMRQ scores are calculated by adding up the eigenvalues of the transition probability matrix provided by MSMs[30, 36]. The theoretical upper limit of GMRQ score is 6 in all cases[19, 30, 36], due to the fact that the number of MSM timescales is chosen to be 5 in the MSM settings. Therefore, in our case, high GMRQ score that approaches 6 usually

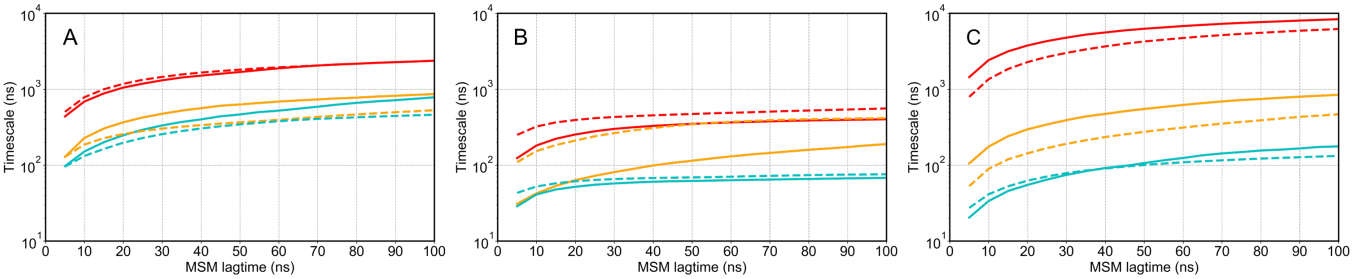

Figure 3: The first three slowest implied timescales as a function of MSM lag time. (A) BBA (PDB ID: 1FME), (B) Villin (PDB ID: 2F4K), (C) WW domain (PDB ID: 2F21). The red, yellow and cyan colored lines indicate the slowest, second slowest and third slowest implied timescales, respectively. Dashed lines correspond to the reference value given by the MSMs built on all contacts featurization. Solid lines correspond to the implied timescales given by the MSMs achieved by using the set of distances optimally chosen by our GA-based method.

suggests a better ability of an MSM to capture the slowest process, whereas low GMRQ score implies ineffective state decomposition during the MSM construction process. All information regarding the GMRQ scores and residue pair selection is summarized in Table 3. As shown in Figure 2, all GMRQ scores converged over 40 iterations. In Figure 2A, the highest GMRQ score for BBA is around 4.445, which is higher than the benchmark GMRQ score (4.239). Similar traits are shown by WW domain in Figure 2C that the best GMRQ from GA (4.198) is higher than the benchmark (4.111). However, one exception happens in Villin, shown in Figure 2B. In Figure 2B, the best predicted GMRQ (3.203) does not reach the benchmark (3.705). More iterations for Villin are needed to reach a best GMRQ score that is higher than the benchmark, but there exists a trade-off between the accuracy and computational resource needed. Overall, the percent variances between our predicted GMRQ score and the benchmark GMRQ score are +4.8% for BBA, -13.5% for Villin and +2.12% for WW domain, in which Villin has the highest difference compared to the other two proteins.

Similar analysis is applied to λ-repressor, except that λ-repressor lacks a benchmark GMRQ score due to its higher number of residues. Hence, there is no reference value to compare in this case. The best GMRQ score given over 40 iterations is around 4.956. Considering that the upper limit of the GMRQ score in this system is 6, we believe that a score of 4.956 is a relatively high GMRQ that effectively captures the slow dynamics of the protein folding mechanisms. Therefore, we can conclude that the method proves its ability to provide the optimal selection of residue pairs for the construction of the best MSM.

3.2 Implied timescale plots show that predicted optimal sets of residue pair distances are able to successfully capture the slowest dynamics in the proteins.

By plotting lag time dependent implied timescale plots, we can quantitatively visualize the slow modes of protein dynamics. Figure 3 shows the comparison between the converged slowest implied timescales provided by all contact featurization and our GA-based method. Again, the reference values are provided by utilizing all residue pair distances as features. Since λ-repressor is too big for all contacts featurization, there is no benchmark data available and its implied timescale is not shown. In Figure 3A, the slowest implied timescales (solid and dashed red lines) of BBA nearly overlap with each other, indicating that our method has chosen a set residue pair distances that captures the slowest process. In addition, the predicted second and third slowest implied timescales (yellow and cyan) are slower than the corresponding timescale for the benchmarks. In Figure 3B, the predicted implied timescales of Villin has a larger deviation. This inconsistency will be explained and justified in the next paragraph. In the case of WW domain (Figure 3C), we capture a slower timescale than the benchmarks. We find that inclusion of all residue pair distances can add noises to the model, and our GA-based method helps improve the MSM construction by excluding those irrelevant features.

3.3 Number of selected distances may reflect the degree of complexity of the protein folding mechanism.

Other than the GMRQ scores and implied timescale plots, more information can be obtained from the sets of residue pair distances. In Table 2, we collect and summarize the number of distances selected by GA and the actual residue numbers in each protein. One interesting thing is that the number of residues in a protein is not necessary correlated to the number of distances needed to capture its slowest dynamics. For example, it can be observed in Table 2 that although both Villin and WW domain have 35 residues in their sequences, WW domain only needs 4 distances of residue pairs while Villin requires 61 distances. This may be due to the complex folding mechanism of Villin. Though both proteins are fast-folding proteins with small numbers of residues, the secondary structure elements in Villin fold more independently without much interactions[31]. Such minimized interaction or minimal frustration makes the folding kinetics fast for Villin, according to the folding funnel theory[7]. Consequently, because the protein folds quickly, this phenomenon suggests a continuous reduction in energy in the folding funnel[7, 9], which implies multiple parallel pathways during the folding hypothesis[51]. On the other hand, WW domain folds much slower than Villin[6] and has more consistent folding

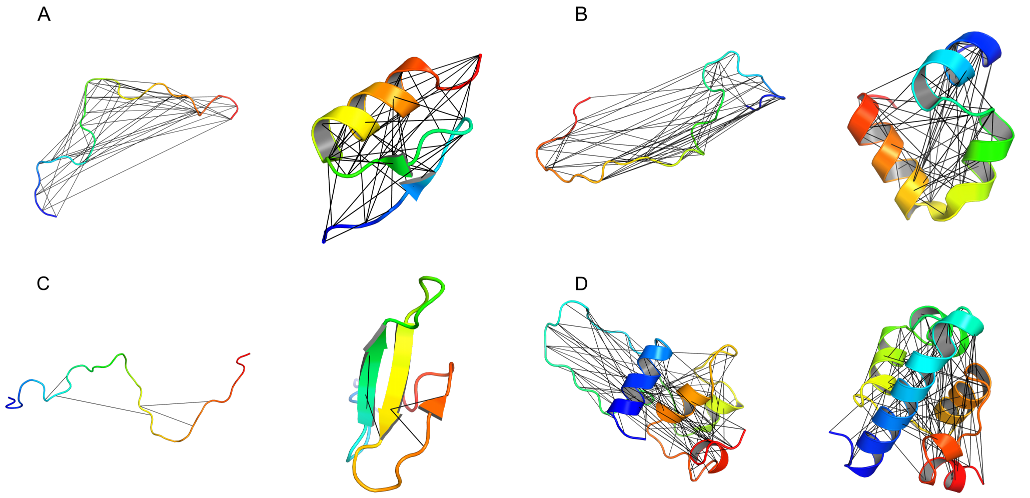

Figure 4: GA-chosen residue pairs visualized on the unfolded MD structures and folded crystal structures. (A) BBA (PDB ID: 1FME), (B) Villin (PDB ID: 2F4K), (C) WW domain (PDB ID: 2F21), (D) λ-repressor (PDB ID: 1LMB). The black lines specify the distances between residue pairs chosen by our GA-based method, which capture the slowest dynamics of the proteins.

pathways[1]. The independent features in Villin make it hard for GA to fully capture its slowest dynamics.

In a previous study, Feng and Shukla[11] utilized evolution couplings (ECs) as functional features to capture protein folding and conformational dynamics, which gives the similar results for Villin and WW domain. Their work identified that Villin needs 73 ECs and WW domain only needs 5 ECs to fully describe the protein dynamics. They stated that more ECs are needed if the ECs has low correlation. Here, our results show the same trait that Villin requires more features for identification of its slowest dynamic processes, which is reasonable due to the folding complexity of Villin comparing to other fast-folding small proteins. To fully capture the slow dynamics of proteins like Villin, a large number of features should be included from its whole dataset. This is a different scenario comparing to capturing the dynamics of the proteins that needs small numbers of features, which is a problem easier for GA to solve. For proteins like Villin, other methods needs to be explored for a more efficient way to capture the slowest dynamics. Although our method results in some degrees of deviations from the benchmarks (shown in Figure 2B and 3B), it still shows effectiveness in dealing with proteins with complicated kinetics.

To present our predicted results in a more understandable way, we visualize the optimal sets of residue pairs for all four proteins in Figure 4. Each section (A, B, C and D) of Figure 4 consists of two parts, representing the unfolded and folded structure of the protein respectively. It is easy to notice that the residue pair distances chosen by our method spread out in the protein to capture the complex dynamics of protein folding.

4 CONCLUSIONS

Feature selection of MSM construction determines the accuracy of predicted kinetics properties. Currently, the selection of features is done using trial and error. The utilization of GMRQ score enables a quantitative description of the accuracy of MSMs in representing the molecular dynamics observed in a simulation dataset. Using GMRQ score as fitness score, we introduce a GA-based method in order to optimize a set of residue pair distances that produce superior MSMs. In this study, we have shown that our method can provide an automatic, efficient and accurate way to choose the optimal residue pair distances as features for MSMs construction. This significantly improves the efficiency in the overall process of building MSMs while still guarantees the quality of MSMs to capture the slowest protein dynamics. Due to the diversity of tested proteins, our method can be widely applied to other proteins to help with the feature selection process and we anticipate that this method will shift MSM building one step closer to a systemic and objective protocol. It is important to be aware that the underlying assumption of this approach is that the slowest dynamic processes correspond to the process of interest. However, this assumption can be challenged in the case of insufficient sampling or inaccurate force field.

However, the method also has some limitations. The proposed method belongs to the class of wrapper methods for feature selection that find the “optimal" feature subset by iteratively selecting features based on the classifier performance. The performance of these methods drops significantly for datasets with large number of important but uncorrelated features. Our method also does not perform well on systems with complex dynamics that requires a large number of features to capture the underlying dynamics. In other words, the effectiveness partially depends on the complexity of the conformational changes in the protein, which is shown in the discussion of Villin. As the folding complexity increases, more pathways are available for the protein, so the selection of residue pairs may not fully depict the slowest dynamics of the protein. However, a large number of biologically relevant dynamic processes have been shown to involve only a few important features[11, 22, 25, 43, 45–47, 50]. In addition, sequence information and crystal structure of the protein should be known, and sufficient amount of MD simulation data should be generated to apply our method. In conclusion, the proposed algorithm, can help identify essential residue pair distances for featurization and exclude noises for MSM construction with high efficiency.

5 REFLECTION

The year-long Blue Waters Internship enriched my experience in many aspects. This opportunity was rare and precious, especially because I can utilize one of the leading-edge petascale computational resources on the Blue Waters Supercomputer. I was excited to be offered the opportunity to meet other interns to study and practice computational skills together. Starting last summer, I have been involved in a variety of activities, including a two-week educational workshop at University of Illinois at Urbana-Champaign, regular webinars, monthly reports and preparing a manuscript. Majoring in Material Science, I joined the internship with limited computational experience. However, I quickly gained essential skills and became adept with the help of internship coordinators, my research advisor and mentors in the lab. In addition, working on the projects helped me to be familiar with the life in a research group and be better prepared for the graduate school. My presentation skills were improved through attending group meetings and poster sessions. I also practiced my writing skills through regular progress reports and writing this manuscript. Overall, the past year was a busy year, but it has became a unique experience in my undergraduate studies.

ACKNOWLEDGMENTS

The authors thank D. E. Shaw Research for MD simulation trajectories (BBA, Villin, WW domain and λ-repressor). Q.C. would like to thank the support of the Shodor Education Foundation and the Blue Waters Student Internship Program. J.F. would like to thank Chia-chen Chu fellowship for support. S.M. is supported by the CSE Fellows Program, Computational Science and Engineering at University of Illinois, Urbana-Champaign, Urbana, IL. D. S. acknowledges support from the Foundation for Food and Agriculture Research via the new innovator program.

REFERENCES

[1] Silvio a Beccara, Tatjana Škrbić, Roberto Covino, and Pietro Faccioli. 2012. Dominant folding pathways of a WW domain. *Proc. Natl. Acad. Sci. U.S.A* 109, 7 (2012), 2330–2335.

[2] BJ Alder and TEf Wainwright. 1957. Phase transition for a hard sphere system. *J. Chem. Phys.* 27, 5 (1957), 1208–1209.

[3] David Ascher, Paul F Dubois, Konrad Hinsen, Jim Hugunin, Travis Oliphant, et al. 2001. Numerical python.

[4] Kyle A Beauchamp, Gregory R Bowman, Thomas J Lane, Lutz Maibaum, Imran S Haque, and Vijay S Pande. 2011. MSMBuilder2: modeling conformational dynamics on the picosecond to millisecond scale. *J. Chem. Theory. Comput.* 7, 10 (2011), 3412–3419.

[5] Kyle A Beauchamp, Daniel L Ensign, Rhiju Das, and Vijay S Pande. 2011. Quantitative comparison of villin headpiece subdomain simulations and triplet–triplet energy transfer experiments. *Proc. Natl. Acad. Sci. U.S.A* 108, 31 (2011), 12734–12739.

[6] Gregory R Bowman, Vijay S Pande, and Frank Noé. 2014. Introduction and overview of this book. In *An introduction to Markov state models and their application to long timescale molecular simulation.* Springer, 1–6.

[7] Joseph D Bryngelson, Jose Nelson Onuchic, Nicholas D Socci, and Peter G Wolynes. 1995. Funnels, pathways, and the energy landscape of protein folding: a synthesis. *Proteins: Struct., Funct., Bioinf.* 21, 3 (1995), 167–195.

[8] John D Chodera, Nina Singhal, Vijay S Pande, Ken A Dill, and William C Swope. 2007. Automatic discovery of metastable states for the construction of Markov models of macromolecular conformational dynamics. *J. Chem. Phys.* 126, 15 (2007), 04B616.

[9] Ken A Dill and Hue Sun Chan. 1997. From Levinthal to pathways to funnels. *Nat. Struct. Mol. Biol.* 4, 1 (1997), 10.

[10] S Doerr, MJ Harvey, Frank Noé, and G De Fabritiis. 2016. HTMD: high-throughput molecular dynamics for molecular discovery. *J. Chem. Theory. Comput.* 12, 4 (2016), 1845–1852.

[11] Jiangyan Feng and Diwakar Shukla. 2018. Characterizing Conformational Dynamics of Proteins Using Evolutionary Couplings. *J. Phys. Chem. B* (2018).

[12] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. 2001. *The elements of statistical learning.* Vol. 1. Springer Series in Statistics New York.

[13] David E Goldberg and John H Holland. 1988. Genetic algorithms and machine learning. *Mach. Learn.* 3, 2 (1988), 95–99.

[14] Cedric Gondro and Brian P Kinghorn. 2007. A simple genetic algorithm for multiple sequence alignment. *Genet. Mol. Res.* 6, 4 (2007), 964–982.

[15] John Henry Holland. 1992. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* MIT press.

[16] Md Tamjidul Hoque and Sumaiya Iqbal. 2017. Genetic algorithm-based improved sampling for protein structure prediction. *Int. J. Bio. Inspir. Com.* 9, 3 (2017), 129–141.

[17] Gerhard Hummer and Attila Szabo. 2014. Optimal dimensionality reduction of multistate kinetic and Markov-state models. *J. Phys. Chem. B* 119, 29 (2014), 9029–9037.

[18] Brooke E Husic, Robert T McGibbon, Mohammad M Sultan, and Vijay S Pande. 2016. Optimized parameter selection reveals trends in Markov state models for protein folding. *J. Chem. Phys.* 145, 19 (2016), 194103.

[19] Brooke E Husic and Vijay S Pande. 2017. Note: MSM lag time cannot be used for variational model selection. *J. Chem. Phys.* 147, 17 (2017), 176101.

[20] Brooke E Husic and Vijay S Pande. 2018. Markov state models: From an art to a science. *J. Am. Chem. Soc.* 140, 7 (2018), 2386–2396.

[21] M Kenneth Jr, Scott M LeGrand, et al. 2012. *The protein folding problem and tertiary structure prediction.* Springer Science & Business Media.

[22] Kai J Kohlhoff, Diwakar Shukla, Morgan Lawrenz, Gregory R Bowman, David E Konerding, Dan Belov, Russ B Altman, and Vijay S Pande. 2014. Cloud-based simulations on Google Exacycle reveal ligand modulation of GPCR activation pathways. *Nature chemistry* 6, 1 (2014), 15.

[23] Thomas J Lane, Diwakar Shukla, Kyle A Beauchamp, and Vijay S Pande. 2013. To milliseconds and beyond: challenges in the simulation of protein folding. *Curr. Opin. Struct. Biol.* 23, 1 (2013), 58–65.

[24] Lisa J Lapidus, Srabasti Acharya, Christian R Schwantes, Ling Wu, Diwakar Shukla, Michael King, Stephen J DeCamp, and Vijay S Pande. 2014. Complex pathways in folding of protein G explored by simulation and experiment. *Biophys. J.* 107, 4 (2014), 947–955.

[25] Morgan Lawrenz, Diwakar Shukla, and Vijay S Pande. 2015. Cloud computing approaches for prediction of ligand binding poses and pathways. *Scientific reports* 5 (2015), 7918.

[26] Kresten Lindorff-Larsen, Stefano Piana, Ron O Dror, and David E Shaw. 2011. How fast-folding proteins fold. *Science* 334, 6055 (2011), 517–520.

[27] Mohammad M. Sultan and Vijay S Pande. 2017. tICA-Metadynamics: Accelerating Metadynamics by using kinetically selected collective variables. *J. Chem. Theory. Comput.* 13, 6 (2017), 2440–2447.

[28] Robert T McGibbon, Kyle A Beauchamp, Matthew P Harrigan, Christoph Klein, Jason M Swails, Carlos X Hernández, Christian R Schwantes, Lee-Ping Wang, Thomas J Lane, and Vijay S Pande. 2015. MDTraj: a modern open library for the analysis of molecular dynamics trajectories. *Biophys. J.* 109, 8 (2015), 1528–1532.

[29] Robert T McGibbon, Carlos X Hernández, Matthew P Harrigan, Steven Kearnes, Mohammad M Sultan, Stanislaw Jastrzebski, Brooke E Husic, and Vijay S Pande. 2016. Osprey: Hyperparameter optimization for machine learning. *J. O. S. S* 1 (2016), 00034.

[30] Robert T McGibbon and Vijay S Pande. 2015. Variational cross-validation of slow dynamical modes in molecular kinetics. *J. Chem. Phys.* 142, 12 (2015), 03B621_1.

[31] James C McKnight, Don S Doering, Paul T Matsudaira, and Peter S Kim. 1996. A thermostable 35-residue subdomain within villin headpiece. *Mol. Biol.* (1996).

[32] Shriyaa Mittal and Diwakar Shukla. 2017. Predicting Optimal DEER Label Positions to Study Protein Conformational Heterogeneity. *J. Phys. Chem. B* 121, 42 (2017), 9761–9770.

[33] Shriyaa Mittal and Diwakar Shukla. 2018. Recruiting machine learning methods for molecular simulations of proteins. *Mol. Simul.* 44, 11 (2018), 891–904.

[34] Alexander S Moffett, Kyle W Bender, Steven C Huber, and Diwakar Shukla. 2017. Allosteric Control of a Plant Receptor Kinase through S-Glutathionylation. *Biophys. J.* 113, 11 (2017), 2354–2363.

[35] Alexander S Moffett and Diwakar Shukla. 2018. Using molecular simulation to explore the nanoscale dynamics of the plant kinome. *Biochem J* 475, 5 (2018), 905–921.

[36] Frank Noé and Feliks Nuske. 2013. A variational approach to modeling slow processes in stochastic dynamical systems. *Multiscale Model Simul.* 11, 2 (2013), 635–655.

[37] Vijay S Pande, Kyle Beauchamp, and Gregory R Bowman. 2010. Everything you wanted to know about Markov State Models but were afraid to ask. *Methods* 52, 1 (2010), 99–105.

[38] Guillermo Pérez-Hernández, Fabian Paul, Toni Giorgino, Gianni De Fabritiis, and Frank Noé. 2013. Identification of slow molecular order parameters for Markov model construction. *J. Chem. Phys.* 139, 1 (2013), 015102.

[39] Nuria Plattner, Stefan Doerr, Gianni De Fabritiis, and Frank Noé. 2017. Complete protein–protein association kinetics in atomic detail revealed by molecular dynamics simulations and Markov modelling. *Nat. Chem.* 9, 10 (2017), 1005.

[40] Jan-Hendrik Prinz, Hao Wu, Marco Sarich, Bettina Keller, Martin Senne, Martin Held, John D Chodera, Christof Schütte, and Frank Noé. 2011. Markov models of molecular kinetics: Generation and validation. *J. Chem. Phys.* 134, 17 (2011), 174105.

[41] Christian R Schwantes and Vijay S Pande. 2013. Improvements in Markov state model construction reveal many non-native interactions in the folding of NTL9. *J. Chem. Theory. Comput.* 9, 4 (2013), 2000–2009.

[42] Christian R Schwantes, Diwakar Shukla, and Vijay S Pande. 2016. Markov state models and tICA reveal a nonnative folding nucleus in simulations of NUG2. *Biophysical journal* 110, 8 (2016), 1716–1719.

[43] Zahra Shamsi, Alexander S Moffett, and Diwakar Shukla. 2017. Enhanced unbiased sampling of protein dynamics using evolutionary coupling information. *Scientific Reports* 7, 1 (2017), 12700.

[44] Diwakar Shukla, Carlos X Hernández, Jeffrey K Weber, and Vijay S Pande. 2015. Markov state models provide insights into dynamic modulation of protein function. *Acc. Chem. Res.* 48, 2 (2015), 414–422.

[45] Diwakar Shukla, Yilin Meng, Benoît Roux, and Vijay S Pande. 2014. Activation pathway of Src kinase reveals intermediate states as targets for drug design. *Nat. Commun.* 5 (2014), 3397.

[46] Diwakar Shukla, Ariana Peck, and Vijay S Pande. 2016. Conformational heterogeneity of the calmodulin binding interface. *Nat. Commun.* 7 (2016), 10910.

[47] Mohammad M Sultan, Gert Kiss, Diwakar Shukla, and Vijay S Pande. 2014. Automatic selection of order parameters in the analysis of large scale molecular dynamics simulations. *J. Chem. Theory. Comput.* 10, 12 (2014), 5217–5223.

[48] Mohammad Taherdangkoo, Mahsa Paziresh, Mehran Yazdi, and Mohammad Hadi Bagheri. 2013. An efficient algorithm for function optimization: modified stem cells algorithm. *Cent. Eur. J. Eng.* 1, 3 (2013), 36–50.

[49] Ron Unger and John Moult. 1993. Genetic algorithms for protein folding simulations. *Biochem Mol Biol J.* 231, 1 (1993), 75–81.

[50] Dan K Vanatta, Diwakar Shukla, Morgan Lawrenz, and Vijay S Pande. 2015. A network of molecular switches controls the activation of the two-component response regulator NtrC. *Nat. Commun.* 6 (2015), 7283.

[51] Li Zhu, Kingshuk Ghosh, Michael King, Troy Cellmer, Olgica Bakajin, and Lisa J Lapidus. 2011. Evidence of multiple folding pathways for the villin headpiece subdomain. *J. Phys. Chem. B* 115, 43 (2011), 12632–12637.