2019 22nd International Conference of Computer and Information Technology (ICCIT), 18-20 December 2019

3DIGARS-PSP: A Novel Statistical Energy Function and Effective Conformational Search Strategy based *ab initio* Protein Structure Prediction

Avdesh Mishra
Department of Electrical Engineering
and Computer Science
Texas A&M University-Kingsville
Kingsville, Texas
avdesh.mishra@tamuk.edu

Md Tamjidul Hoque
Department of Computer Science
University of New Orleans
New Orleans, Lousiana
thoque@uno.edu

***Abstract*—To solve protein structure prediction (PSP) problems computationally, a plethora of template-based methods exist. However, there are very few *ab initio* models for PSP. Template-based modeling relies on the existing structures and therefore is not effective for non-homologous sequence-based structure prediction. Thus, *ab initio* modeling is indispensable in such cases, even though it is a challenging optimization problem. To cope, we utilize an effective energy function (called 3DIGARS) and an advanced search algorithm (called KGA) based *ab initio* PSP, called 3DIGARS-PSP. To address critical search, the proposed genetic algorithm deploys two effective operators: angle rotation and segment translation. Further, propensities of torsion angle and secondary structure distribution have been utilized to guide the conformation search. Crucial features, such as sequence-specific accessibility, hydrophobic-hydrophilic properties and torsion angles of protein residues are mined to formulate an optimized energy function, which is then combined with the advanced sampling algorithm to explore critical conformational space. Consequently, 3DIGARS-PSP performed well compared to the state-of-the-art method for a set of low TMscore models from CASP data.**

Keywords—ab initio, conformational search, energy function, genetic algorithm, protein structure prediction

I. INTRODUCTION

Protein tertiary structure prediction is one of the most challenging problems in molecular and structural biology. The goal of protein tertiary structure prediction is to accurately predict the spatial position of each atom in a 3D protein from only the sequence of amino acid residues. There exist experimental approaches for protein structure prediction (PSP), e.g., X-ray crystallography and nuclear magnetic resonance (NMR) but, these methods are far too slow and expensive for PSP. Moreover, there are computational approaches available for the PSP problem. The existing computational approaches can be categorized into two broad categories *i)* homology modeling or template-based modeling and *ii) ab initio* or *de novo* modeling depending on whether similar proteins have already been experimentally solved. If proteins of the similar structure are recognized from the PDB [1] library, and the templates of the similar proteins are utilized to construct the target model, then this approach is called "homology modeling or template-based modeling" [2], [3], [4], [5], [6]. However, if protein templates are not available, the 3D structure is built from only the sequence of amino acid residues, and this approach is called "*ab initio* or *de novo* modeling" [7], [8], [9], [10], [11], [12], [13], [14].

The approach of homology modeling for PSP has achieved significant success, and the reason is the growing number of experimentally solved structures available in the PDB library. Nevertheless, this approach fails to produce an effective structure in the absence of similar proteins. This necessitates the development of the *ab initio* method for PSP. Typically, *ab initio* modeling comprises two essential components *i*) an accurate energy function and *ii*) effective conformational search. The energy function is used to evaluate the fitness of a given conformation and in general, distinguish the native structure from native-like decoys [15], [16], [17], [18], [19]. Likewise, a search algorithm is used to explore the protein's conformational space by generating diverse and effective conformational samples.

In this study, we develop a new algorithm, 3DIGARS-PSP for *ab initio* protein structure prediction, with the focus on an elegant design of the energy function as well as the search algorithm. Our design of an energy function involves the generation of multiple 3D structural and sequence-specific energetic features using multiple data sets of known proteins and two different reference states. Subsequently, the energetic features are ranked based on the *Pearson Correlation Coefficient* (PCC) and their optimal combination is obtained using the Genetic Algorithm (GA) [20], [21]. During optimization, the feature selection technique is used and only the features which helped improve the fitness of the GA are considered in the energy calculation of the structure. The optimized energy function is then used to evaluate the structures generated during the *ab initio* PSP process. Moreover, the design of the search process involves conformational change in the structure. We achieve the conformational change in the structure by applying the GA with novel mutation and crossover operators based on angular rotation and translation capabilities.

This work is supported by the Louisiana Board of Regents through the Board of Regents Support Fund, LEQSF (2016-19)-RD-B-07.

978-1-7281-5842-6/19/$31.00 ©2019 IEEE

Further, we conducted a systematic test and analysis of the proposed 3DIGARS-PSP method on the target proteins from the *Critical Assessment of Protein Structure Prediction* 8 (CASP8) [22]. We compare the models predicted by 3DIGARS-PSP with the models predicted by one of the existing, top-performing PSP method, called Rosetta [23] in terms of the TM-score [24], [25] performance measure.

II. METHOD

A. Design of Optimal Energy Function

The energy function plays a key role in evaluating the fitness of a given conformation and guiding the conformational search process by discriminating native-like structures from an ensemble of decoy structures, generated by conformational sampling during the *ab initio* PSP process. In this work, we designed an optimized energy function whose total energy is the sum of the 6 energy features as shown in (1).

$$\begin{aligned} E_{tot} &= E_{ASA\_TDS3\_RS1} + w_1 E_{3DIGARS} + w_2 E_{ASA\_REGAd^3p} \\ &+ w_3 E_{Psi\_TDS4\_RS1} + w_4 E_{Psi\_TDS2\_RS1} \\ &+ w_5 E_{Psi\_Triplet\_TDS4\_RS1} \end{aligned} \quad (1)$$

where, energy features are computed from four different datasets (DS1, DS2, DS3, and DS4) which are described under "Datasets for Constructing Libraries for Energy Function" in *Section D, Subsection 1* and two different reference states (RS1 and RS2) which are also described later in this section. Note that $E_{3DIGARS}$ and $E_{ASA\_REGAd^3p}$ energy features were computed from DS1 and DS2 respectively, in our prior work [16], [15] and are extracted from 3DIGARS3.0 [17] energy function for the purpose of this work. Furthermore, w_1 = 1.98, w_2 = 0.70, w_3 = 1.16, w_4 = 0.03, and w_5 = 0.25 are the weighting factors to balance the energy features, which were tuned using a GA based on a multi-objective fitness function as define in (2).

$$\begin{aligned} Obj\_fxn &= (Avg\_PCC + (-1.0)Avg\_TMscore \\ &+ (-1.0)Avg\_Native\_Count + Avg\_Zscore) \end{aligned} \quad (2)$$

where, *Avg_PCC* is an average of the correlation coefficient for all the proteins in a dataset whereas, PCC is computed from the total energy and the structural accuracy (or TM-score) of the models in a protein, *Avg_TMscore* is an average of the TM-score of the low energy models, *Avg_Native_Count* is an average number of correctly selected native structures out of decoys and *Avg_Zscore* is an average of Z-scores (more negative Z-scores indicate that the energy function is able to clearly separate natives from decoys). The average for all of the components of the objective function is computed by dividing the features by the count of proteins present in the optimization dataset which are described under "Optimization Datasets for Energy Function" in *Section D, Subsection 2*. The GA parameters used for energy function optimization were set to *i)* maximum generation of 20,000; *ii)* population size of 200; *iii)* elite rate of 5%; *iv)* crossover rate of 90%; and *v)* mutation rate of 50%. Additionally, each of the weight variables, w_1 through w_5 were represented by an 11-bit binary (1/0) encoding system.

Initially, we designed 41 energy features among which 17 of the energy features were obtained by a sequence-specific *accessible surface area* (ASA) energy calculation method, 20 of the energy features were obtained by a sequence-specific torsion angle (Phi and Psi) energy calculation method and 4 of the energy features were obtained from the 3DIGARS3.0 [17] energy function (1 feature for the hydrophobic-hydrophilic energy term, 1 feature for the sequence-specific ASA energy term where predicted ASA energy is computed using the REGAd^{3}p tool [15] and 2 features for uPhi and uPsi energy terms). All other features except the features extracted from 3DIGARS3.0 energy function were generated using the outputs from DSSP [26] and Spider2 [27] programs. DSSP program provides the real value assignment of the phi-psi angle pair and ASA from the structure of the protein. Whereas, Spider2 is a program which provides predicted phi-psi angle pair and ASA from the sequence of amino acids (fasta sequence). Among 41 features, only 6 of the features were finally considered for total energy calculation and the rest of the features were ruled out using incremental feature selection technique.

1) Sequence-specific ASA Energy Features

The sequence-specific ASA energy feature, E_{ASA} is computed from the probability $P(\Delta ASA_i|AA_i)$ of the prediction error of ASA ($\Delta ASA_i = ASA_i^{Real} - ASA_i^{Pred}$) for a given amino acid type, AA_i over all the ASA along the sequence. The sequence-specific ASA energy feature is mathematically represented as:

$$E_{ASA} = -RT\sum_i \ln P(\Delta ASA_i \mid AA_i) \quad (3)$$

where, R is the gas constant and T is the temperature. The ASA_i^{Real} and ASA_i^{Pred} terms in the prediction error calculation are obtained from DSSP [26] and Spider2 [27] for a given amino acid type, AA_i. Two different probability functions $P(\Delta ASA_i|AA_i)$ and $P(\Delta ASA_i|AA_i,SS_i)$ were obtained from four different datasets (DS1, DS2, DS3, and DS4, discussed under "Dataset Collection" in *Section D, Subsection 1*) and two different reference states. The reference state indicates the distribution of atoms in a protein system when the interaction is turned off. To test the influence of different reference states, we employed two different reference states *i*) based on conditional probability proposed by Samudrala and Moult [28] and *ii*) based on averaging technique proposed by Hoque *et al*. [29]. We generated 16 sequence-specific ASA based features by using Spider2 ASA predictor, two different probability functions, four different datasets, and two different reference states.

2) Sequence-specific Torsion Angle Energy Features

The sequence-specific torsion angle energy feature, E_θ is computed from the probability $P(\Delta\theta_i|AA_i, SS_i)$ of the prediction angle error ($\Delta\theta_i = \theta_i^{Real} - \theta_i^{Pred}$) for a given amino acid type, AA_i and predicted secondary structure, SS_i over all the torsion angles along the sequence. The sequence-specific torsion energy feature is mathematically represented as in (4).

$$E_\theta = -RT\sum_i \ln P(\Delta\theta_i \mid AA_i, SS_i) \quad (4)$$

where, R is the gas constant and T is the temperature. The θ_i^{Real} and θ_i^{Pred} terms in the prediction angle error calculation are obtained from DSSP and Spider2 programs for a given amino acid type, AA_i and predicted secondary structure, SS_i. The probability function $P(\Delta\theta_i|AA_i,SS_i)$ was obtained from four different datasets and two different reference states

similar to the ASA energy feature extraction as stated above. We generated 16 sequence-specific torsion angle (phi and psi) based features by using four different datasets and two different reference states. Two additional features based on uPhi and uPsi were reproduced using reference states based on the averaging technique proposed by Hoque *et al.* [29] whereas, these features were initially generated using a conditional probability based reference state in 3DIGARS3.0 [17].

3) Sequence-specific ASA and Torsion Angle Energy Feature Computed from Amino Acid Triplets

The sequence-specific ASA energy feature from amino acid triplets, $E_{ASA\_Triplet}$ is computed from the probability $P(\Delta ASA_i|AA_{i-1}\text{-}AA_i\text{-}AA_{i+1})$ of the error of ASA ($\Delta ASA_i = ASA_i^{Real} - ASA_i^{Pred}$) for a given amino acid type, AA_i over all the ASA computed along the sequence. Similarly, the sequence-specific torsion angle energy feature for amino acid triplets, $E_{\theta\_Triplet}$ are computed from the probability $P(\Delta\theta_i|AA_{i-1}\text{-}AA_i\text{-}AA_{i+1},SS_i)$ of the prediction angle error ($\Delta\theta_i=\theta_i^{Real} - \theta_i^{Pred}$) for a given amino acid type, AA_i and predicted secondary structure, SS_i over all the torsion angles computed along the sequence. The sequence-specific ASA energy feature for an amino acid triplet is mathematically represented as in (5).

$$E_\theta = -RT\sum_i \ln P(\Delta ASA_i \mid (AA_{i-1} - AA_i - AA_{i+1})) \quad (5)$$

where, $(AA_{i-1}\text{-}AA_i\text{-}AA_{i+1})$ represents an amino acid triplet at position '*i*' in the sequence. Similarly, the sequence-specific torsion angle energy feature for amino acid triplets is mathematically represented as in (6):

$$E_\theta = -RT\sum_i \ln P(\Delta\theta_i \mid (AA_{i-1} - AA_i - AA_{i+1}), SS_i) \quad (6)$$

where, $(AA_{i-1}\text{-}AA_i\text{-}AA_{i+1})$ again represents an amino acid triplet at position '*i*' in the sequence. By this approach, we generated 1 feature for sequence-specific ASA energy and 2 features for sequence-specific torsion (1 for phi and 1 for psi) energies.

B. Design of Conformational Search

Effective conformational search is another critical component of *ab initio* protein structure prediction, where the design of conformational change operators which can effectively sample the energy hyper-surface of the protein folding process, looking for the global minimum or the native fold of the protein is essential for improving the efficiency of search algorithms. Towards this goal, we designed a memory assisted GA which involves two types of conformational change operators *i*) angle rotation; and *ii*) segment translation. Our mutation operation involves phi or psi angle rotation and crossover operation involves segment translation followed by phi or psi angle rotation at the crossover point. Rotation of phi and psi angles involves rotation about an arbitrary axis. We consider this arbitrary axis to pass through the atoms that are involved in phi and psi angle formation. Torsion angle phi involves the backbone atoms $C(O)_{n-1}\text{-}N_n\text{-}C(\alpha)_n\text{-}C(O)_n$ and psi involves the backbone atoms $N_n\text{-}C(\alpha)_n\text{-}C(O)_n\text{-}N_{n+1}$. To perform phi angle rotation we follow steps described in Algorithm 1. In a like manner, psi angles are rotated by the similar steps described in Algorithm 1. However, the points p1 and p2 here instead represent atoms $C(\alpha)_n$ and $C(O)_n$, respectively. Moreover, to generate child structures of GA by crossing over parent structures, the segment translation technique is employed. A set of possible crossover points are selected based on the secondary structure information. All amino acid indexes except the amino acids belonging to the beta-sheet secondary structure type (either E or B) are considered as possible crossover points. This is because we want to preserve beta-sheet regions in the structure from random changes during the crossover operation and perform more careful changes in the beta-sheet region while performing mutation operation.

Algorithm 1: Phi Angle Rotation

1. Select an axis passing through two points p1 and p2 (atoms N_n and $C(\alpha)_n$).
2. Translate point p1 (atom N_n) to the origin.
3. Rotate point p2 (atom $C(\alpha)_n$) onto the Z-axis.
4. Rotate the segment of the structure after point p2 around the Z-axis.
5. Rotate the axis passing through two points p1 and p2 to the original orientation.
6. Translate the structure to the original position.

During the crossover process we generate four child structures from two-parent structures and a structure with the best fitness saved in the memory [20]. After selecting a crossover point, the *first child* structure is created by copying atoms starting at position one to the crossover point from *first parent* and the translated atoms starting at crossover point plus one to the last atom from the *second parent*. Similarly, *second child* structure is created by copying atoms starting at position one to the crossover point from *first parent* and the translated atoms starting at crossover point plus one to the last atom from the *structure in memory*. Alternatively, the *third child* structure is created by copying the translated atoms starting at position one to one less than the crossover point from *second parent* and the atoms starting at the crossover point to the last atom from *first parent*. Similarly, the *fourth child* structure is created by copying the translated atoms starting at position one to one less than the crossover point from the *structure in memory* and the atoms starting at the crossover point to the last atom from *first parent*. After segment translation is complete the torsion angles of the child structure at the crossover point are rotated back to the original torsion angles of parent structures. This is done to ensure that the secondary structure type before crossover and after crossover remains consistent. Furthermore, we update the fragments of the structure in the memory with the fragments that result in better fitness during the crossover process. This ensures that the segment that yields better fitness is preserved and used in the next round of crossover operation during the search process. The memory assisted GA presented in this work is an extension of KGA implemented specifically for the purpose of *ab initio* PSP. For the basics on KGA please refer [20] and for the detailed implementation of memory assisted GA please refer to the 3DIGARS-PSP Software code available freely online at http://cs.uno.edu/~tamjid/Software/ab_initio/v2/PSP.zip.

C. Ab initio Protein Structure Prediction Method (3DIGARS-PSP)

Protein structure in 3DIGARS-PSP is represented by backbone atoms N, Cα, C and O. We start by initializing some of the chromosomes of the GA population with the Cartesian coordinates of the backbone atoms of the models

obtained from Rosetta [23] server. The rest of the chromosomes are initialized by single point torsion angle changes (rotation). To change the phi or psi angles effectively, we collected the frequency of occurrence of 20 different amino acids with different phi-psi torsion angle pairs. Both phi and psi angles are divided into 120 bins with an interval of 3 degrees, summarized from the 4,332 high-resolution experimental structures. An example that shows how the frequency of occurrence is computed is as follows: if amino acid "ALA" has phi angle of -178 degrees and psi angle of 179 degrees, the frequency count for amino acid "ALA" at psi index zero and phi index zero will be increased by one. The frequency distribution obtained for each amino acid is further categorized into zones by looking at the cluster of the frequency values. To update the phi or psi angle of a certain amino acid type (*aa_type*) first, the torsion angle type (*tor_type*) is selected randomly. Next, the zone index (*zone_ind*) belonging to *aa_type* is selected randomly. Then, the roulette wheel selection method is applied to select the most probable torsion angles (namely, pPhi or pPsi) belonging to the *zone_ind*. Later, if *tor_type* = phi angle, we select a random phi (say, rPhi) between pPhi-3 and pPhi and rotate the current phi angle to achieve rPhi angle. Whereas, if *tor_type* = psi angle, we select a random psi (rPsi) between pPsi and pPsi+3 and rotate the current psi angle to achieve rPsi angle.

The changes of the torsion angles are also guided by the secondary structure (SS) types of the amino acids which are mined from the 4,332 high-resolution experimental structures. To mine the SS types, first, we run the DSSP [26] program on the experimental structures to obtain the phi-psi angle pair and the SS type for each of the amino acids in each of the proteins. DSSP output gives eight different SS types (E, B, H, G, I, T, S, and U) which are broadly categorized into four different SS types (H, E, T, and U). The SS types "E and B" are considered as "E", "H, G and I" are considered as "H", "T and S" are considered as "T" and a blank is considered as "U or undefined". Using phi-psi angle pair and SS types, we obtain the index in our SS frequency table and increase the frequency count of the cell in the frequency table by one. E.g. if amino acid "ALA" has a phi angle of -178 degrees, a psi angle of 179 degrees, and the SS type as "H" the frequency count for amino acid "ALA" at psi index zero, phi index zero and SS index zero is increased by one. Later, the SS type which has the largest frequency count is assigned to the given amino acid having a certain phi-psi angle. Additionally, we collect the phi-psi angle pairs belonging to the H and E secondary structure types and group them into helix and beta groups. We utilize the phi-psi angle pairs belonging to the helix or sheet group to update the phi or psi angles that result in the clash within the structure.

Moreover, the random change of phi or psi angles within the structure could produce low-resolution structures. In other words, random changes could destroy the conserved beta-sheet regions of the structure. To overcome this issue, we apply a beta smoothing technique. An amino acid (AA_i) is considered to satisfy the beta condition if any of the following conditions are satisfied: *i*) AA_{i-1} and AA_{i+1} both have SS type "E"; *ii*) AA_{i-1} and AA_{i-2} both have SS type "E"; and *iii*) AA_{i+1} and AA_{i+2} both have SS type "E". AA_i is the amino acid that is selected for change. To change the phi or psi angle of the AA_i, we follow the steps shown in Algorithm 2. Furthermore, changes in phi or psi angles could result in a clash between atoms within the structure. To prevent clashes, we check the distance between all possible Cα atom pairs within the structure and discard the change if Cα-Cα distance is less than 3.6 Å. If the change in phi or psi angles of the current residue results in a clash then a new residue position is selected for the change.

Algorithm 2: Change in Phi or Psi Angles Constrained by Beta Condition

1. Check if AA_i belongs to SS type "E".
2. Check if AA_i satisfies Beta Condition
   - If TRUE
     - Accept the change in Phi or Psi angle if new Phi-Psi angle pair belongs to SS type "E".
       - If new Phi-Psi angle pair belongs to SS type other than "E".
         - Update the angle under consideration (Phi or Psi) with the most probable torsion angles from the beta group based on the roulette wheel selection mechanism.
   - If FALSE
     - If new Phi-Psi angle pair belongs to SS type "H"
       - Update the angle under consideration (Phi or Psi) with the most probable torsion angles from the helix group based on the roulette wheel selection mechanism.
     - If new Phi-Psi angle pair belongs to SS type other than "H"
       - The rotation of Phi or Psi angle is performed to achieve the new Phi-Psi angle pairs.

In our implementation, before applying the energy function to evaluate the fitness of the structure, we obtain the full atomic model from the backbone model using Oscar-star [30]. The flowchart of the 3DIGARS-PSP method is shown in Fig. 1.

For effective conformational search, the parameters of the GA were configured as: *i)* maximum generation of 300; *ii)* population size of 100; *iii)* elite rate of 5%; *iv)* crossover rate of 70%; and *v)* mutation rate of 60%.

D. Data Collection

This section first discusses the dataset used for constructing libraries and the optimization of the energy function. Then, it discusses the test dataset collected for testing of our *ab initio* method.

1) Datasets for Constructing Libraries for Energy Function

We collected three different sets of data and created the fourth set by combining the three, to construct energy score libraries and obtain multiple features.

a) Datasets1 (DS1)

The experimental structures (proteins) in this set were obtained from the PDB server. The proteins with unknown residues as well as with missing residues anywhere except for five terminal residues on either side were rejected to avoid any noise in the data. The final dataset consists of 4,332 proteins with resolution ≤ 2.5Å, single-chain proteins, and a sequence identity cutoff of 100%. This dataset was published previously and used for constructing energy score

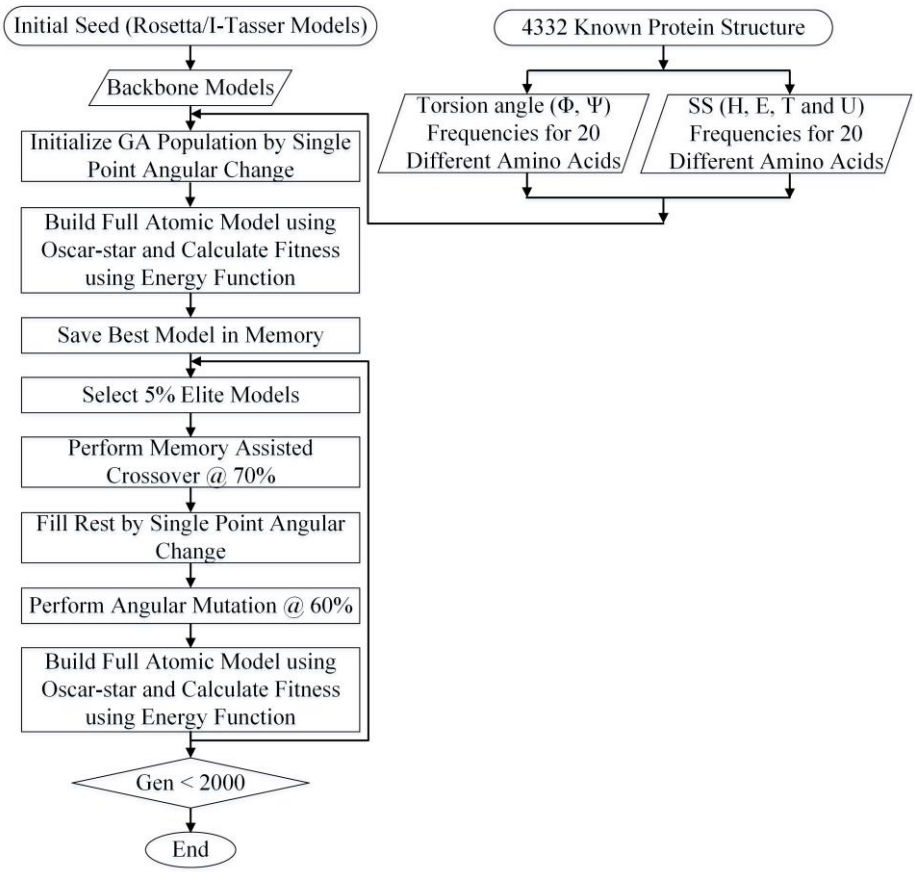

Fig. 1. Flowchart of the 3DIGARS-PSP ab initio prediction.

libraries in 3DIGARS [16] energy function. In this work, we discarded 8 of the proteins which consisted of only alpha carbon atoms as the DSSP [26] program requires the full atomic structure to compute phi and psi angles and the SS type for each amino acid. The resulting dataset consists of 4,324 proteins, which were used to generate the energy function features in this work.

b) Datasets2 (DS2)

DS2 was prepared from the PDB and consists of 1299 proteins after data purification. Initially, 2,793 proteins (both single and multiple chains) were collected from the PDB with the following data collection parameters: *i)* solved by X-ray crystallography; *ii)* resolution ≤ 1.5Å; *iii)* residue length ≥ 40; and *iv)* 30% sequence identity cutoff. Then, the proteins were refined to keep only those with a 25% sequence identity cutoff. Next, the proteins with unknown residues and missing Cartesian coordinates were discarded to avoid any noise in the data. This dataset is the same as the dataset used to generate the sequence-specific ASA based energy feature in the 3DIGARS2.0 [15] energy function.

c) Datasets3 (DS3)

DS3 consists of 2,479 high-resolution (resolution lower than 3Å), non-redundant (sequence identity < 25%) proteins taken from the protein sequence culling server, PISCES [31]. The proteins in this set have 500 or fewer amino acid residues. This dataset is the same as the dataset implemented in training and testing of the SPINE X [32] server.

d) Datasets4 (DS4)

DS4 is a combination of DS1, DS2, and DS3.

2) Optimization Datasets for Energy Function

To optimize the weights of the energy function, we collected the structures submitted in four Critical Assessment of Protein Structure Prediction's (CASP); CASP8 [22], CASP9 [33], CASP10 [34] and CASP11 [35]. Furthermore, the native structures for the proteins were obtained from Zhang Lab [36], [37], [38], [39]. The native structures were only used for TMscore based structure assessment of our *ab initio* method. We carried out the following two-step refinement to ensure quality optimization set collection: *i*) proteins that have missing residues were removed from the optimization set; *ii*) if the models contain an additional number of residues at the beginning and end of the structure compared to the native structure, the additional residues at the beginning and end were chopped off from the models. After filtration, the CASP8 set consists of 73 proteins, CASP9 set consists of 82 proteins, CASP10 set consists of 67 proteins and CASP11 set consists of 59 proteins. Furthermore, CASP8, CASP9 and CASP10 consists of 300 models per protein and CASP11 consists of 200 models per proteins on average.

3) Test Datasets for Ab initio PSP (TAI16)

To assess the robustness of the 3DIGARS-PSP method, we collected the models with TMscore < 0.5 submitted by the Rosetta server in the CASP8 challenge. We found that, among 73 proteins, 16 of the proteins have TMscores < 0.5 for the models submitted by Rosetta. We name this benchmark set of 16 proteins as TAI16. We compared our method with the Rosetta method based on these 16 proteins as we believe that they represent the true *ab initio* predictions by the Rosetta method.

III. RESULTS

Here we discuss the robustness of our approach based on obtained results and analysis.

1) Results of the Energy Function Optimization

After ranking the features based on average PCC between the total energy and the model's structural accuracy (TMscore), we sequentially added and ruled out the features based on their importance in improving the objective fitness during energy function optimization. In Table 1 and Table 2, we show the improvements we achieved in our energy

function based on the components of the objective function: *i*) Average PCC; *ii*) Average TMscore; *iii*) Native Count; and *iv*) Average Zscore. The "Average PCC" column of Table 1 and Table 2 shows that there is a slight decrement in the average PCC. Nonetheless, from "Native Counts" column of Table 1 and Table 2, we can clearly see that the optimized energy function with 6 energy features results in 111.39% improvement and is able to select more natives from the dataset of decoy structures (CASP8, CASP9, CASP10, and CASP11), which is the primary objective of the energy function. Furthermore, from the "Average TMscore" column of Table 1 and Table 2, it is evident that the improved energy function can select the best models from an ensemble of decoys based on the average TM-score with a percentage improvement of 2.08%. Similarly, based on the "Average Zscore", the optimized energy function is improved by 60.44%. This shows the significance of our multi-objective optimization technique in improving the accuracy of the energy function.

2) Results of the 3DIGARS-PSP Method

We evaluated the performance of the 3DIGARS-PSP method on the challenging benchmark set TAI16, which consists of 16 proteins. Each of the proteins in TAI16 consists of low TMscore models (TMscore < 0.5) submitted by the Rosetta server in CASP8. In Table 3, we compare the performance of the 3DIGARS-PSP method with Rosetta based on the TM-score (structure assessment criteria). Based on the average TM-score of the first model out of the five, in set TAI16 submitted by Rosetta, the average TM-score of the 3DIGARS-PSP models is 3.11% better than Rosetta (see Table 3, column "Rosetta (First Model)"). Moreover, based on the average of the average TM-score of 5 models, in TAI16 set submitted by Rosetta, 3DIGARS-PSP achieves a 5.56% improvement over Rosetta (see Table 3, column "Rosetta (Average of 5 Models)"). From Table 3 it is evident that 3DIGARS-PSP provides superior performance over Rosetta.

IV. CONCLUSIONS

We have proposed a new and advanced algorithm, 3DIGARS-PSP, for *ab initio* protein structure prediction. In 3DIGARS-PSP, the backbone atoms (N, Cα, C and O) in a Cartesian coordinate system define protein conformations. Representing protein conformation by only backbone atoms is our first step to reduce the large search space. In a subsequent step, we reduce the search space by deploying a memory assisted GA which involves two types of conformational change operators *i)* angle rotation; and *ii)* segment translation. Moreover, we perform a torsion angle and secondary structure distribution guided changes instead of random sampling to generate lower energy conformations.

We show that our optimized energy function consisting of 6 energy features, computed from sequence-specific accessibility, hydrophobic-hydrophilic properties, and torsion angles is able to select a higher number of native structures from the CASP decoy sets. Also, when tested on the CASP decoy set, our energy function is found to select the low energy conformation decoys more accurately based on TM-score and Z-scores.

TABLE I. VALUES OF OBJECTIVE FUNCTION COMPONENT WHILE USING THE HIGHEST RANKED FEATURE

Dataset	Objective Function Components			
	Average PCC	*Average TMscore*	*Native Counts*	*Average Zscore*
CASP8 (73)[a]	-0.7003	0.6628	23	-0.9736
CASP9 (82) [a]	-0.7049	0.6305	15	-0.8083
CASP10 (67) [a]	-0.6654	0.6614	24	-1.3768
CASP11 (59) [a]	-0.6450	0.5626	17	-1.1600
Average	-0.6789	0.6293	19.75	-1.0797

[a.] Total number of proteins available in corresponding dataset.

TABLE II. VALUES OF OBJECTIVE FUNCTION COMPONENT WHILE USING SIX OF THE FINAL ENERGY FEATURES

Dataset	Objective Function Components			
	Average PCC	*Average TMscore*	*Native Counts*	*Average Zscore*
CASP8 (73)[a]	-0.7189	0.6735	44	-1.6242
CASP9 (82) [a]	-0.6888	0.6482	46	-1.4988
CASP10 (67) [a]	-0.6008	0.6739	43	-2.0135
CASP11 (59) [a]	-0.6062	0.5740	34	-1.7927
Average	-0.6537 (-3.71%)[b]	0.6424 (2.08%) [b]	41.75 (111.39%) [b]	-1.7323 (60.44%) [b]

[a.] Total number of proteins available in corresponding dataset.
[b.] Percentage of improvement while using six best energy features

TABLE III. PERFORMANCE OF 3DIGARS-PSP ROSETTA METHODS ON CASP8 LOW TMSCORE MODELS

Protein ID	TMscores		
	3DIGARS-PSP	*Rosetta (Average of 5 Models)*	*Rosetta (First Model)*
T0397	0.3934	0.35636	0.3566
T0409	0.4523	0.40528	0.4407
T0460	0.3017	0.26156	0.2624
T0466	0.2239	0.26246	0.3259
T0467	0.2912	0.27222	0.3031
T0468	0.236	0.20982	0.2529
T0474	0.4435	0.4839	0.5029
T0476	0.3402	0.2932	0.2793
T0478	0.2467	0.2436	0.2461
T0480	0.2736	0.2303	0.2077
T0482	0.3663	0.36222	0.3516
T0484	0.2651	0.24994	0.2527
T0495	0.4017	0.40656	0.4091
T0496	0.3179	0.25928	0.2158
T0498	0.2376	0.2387	0.2387
T0504	0.2738	0.24766	0.262
Average	0.316556	0.298941 (5.56%) [a]	0.306719 (3.11%) [a]

[a.] Percentage of improvement of 3DIGARS-PSP over Rosetta based on average TMscore of 5 models and average of TMscore of the first models respectively.

Combining improved sampling and an optimized energy function attains improvement over Rosetta template-based method, based on the test performed on the low TMscore models, selected from CASP8 dataset (TAI16). Our method showed 5.56% and 3.11% improvement over Rosetta based on the average of the average TM-scores of the top 5 models and average of the first models, on the benchmark set TAI16, respectively. Despite notable improvement in this work, continuous efforts in both aspects of energy function development and conformational search improvement are still necessary to improve the accuracy of the *ab initio* protein structure prediction.

ACKNOWLEDGMENT

We are thankful to Glenn Robert McLellan for proof-reading this manuscript.

REFERENCES

[1] R. PDB. (February 2014). *Advanced Search Interface.* Available: http://www.rcsb.org/pdb/search/advSearch.do

[2] K. Ginalski, J. Pas, L. S. Wyrwicz, M. v. Grotthuss, J. M. Bujnicki, and L. Rychlewskia, "ORFeus: detection of distant homology using sequence profiles and predicted secondary structure," *Nucleic Acids Res.,* vol. 31, pp. 3804-3807, 2003.

[3] D. T. Jones, "GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences," *Journal of Molecular Biology,* vol. 287, pp. 797-815, 1999.

[4] K. Karplus, C. Barrett, and R. Hughey, "Hidden Markov models for detecting remote protein homologies," *Bioinformatics,* vol. 14, pp. 846-56, 1998.

[5] J. Skolnick, D. Kihara, and Y. Zhang, "Development and large scale benchmark testing of the PROSPECTOR 3.0 threading algorithm," *Proteins: Struct., Funct., Bioinf.,* vol. 56, 2004.

[6] S. Wu and Y. Zhang, "MUSTER: improving protein sequence profile-profile alignments by using multiple sources of structure information," *Proteins,* vol. 72, pp. 547-556, 2008.

[7] D. Bhattacharya, R. Cao, and J. Cheng, "UniCon3D: de novo protein structure prediction using united-residue conformational search via stepwise, probabilistic sampling.," *Bioinformatics,* vol. 32, pp. 2791-2799, 2016.

[8] D. Bhattacharya and J. Cheng, "i3Drefine software for protein 3D structure refinement and its assessment in CASP10," *PloS one,* vol. 8, p. e69648, 2013.

[9] P. Bradley, K. M. S. Misura, and D. Baker, "Toward high-resolution de novo structure prediction for small proteins," *Science,* vol. 309, pp. 1868-1871, 2005.

[10] R. Cao, D. Bhattacharya, B. Adhikari, J. Li, and J. Cheng, "Large-scale model quality assessment for improving protein tertiary structure prediction.," *Bioinformatics,* vol. 31, pp. i116-i123, 2015.

[11] R. Jauch, H. C. Yeo, P. R. Kolatkar, and N. D. Clarke, "Assessment of CASP7 structure predictions for template free targets.," *Proteins: Struct., Funct., Bioinf.,* vol. 69, pp. 57-67, 2007.

[12] J. L. Klepeis, Y. Wei, M. H. Hecht, and C. A. Floudas, "Ab initio prediction of the three-dimensional structure of a de novo designed protein: A double-blind case study," *Proteins: Struct., Funct., Bioinf.,* vol. 58, pp. 560-570, 2005.

[13] A. Liwo, M. Khalili, and H. A. Scheraga, "Ab initio simulations of protein-folding pathways by molecular dynamics with the united-residue model of polypeptide chains.," *Proceedings of the National Academy of Sciences U S A,* vol. 102, pp. 2362-2367, 2005.

[14] S. Wu, J. Skolnick, and Y. Zhang, "Ab initio modeling of small proteins by iterative TASSER simulations," *BMC Biology,* vol. 5:17, 2007.

[15] S. Iqbal, A. Mishra, and T. Hoque, "Improved Prediction of Accessible Surface Area Results in Efficient Energy Function Application," *Journal of Theoretical Biology,* vol. 380, pp. 380-391, 2015.

[16] A. Mishra and M. T. Hoque, "Three-Dimensional Ideal Gas Reference State Based Energy Function," *Current Bioinformatics,* vol. 12, pp. 171-180, 2017.

[17] A. Mishra, S. Iqbal, and M. T. Hoque, "Discriminate protein decoys from native by using a scoring function based on ubiquitous Phi and Psi angles computed for all atom.," *Journal of theoretical biology,* vol. 398, pp. 112-121, 2016.

[18] Y. Yang and Y. Zhou, "Specific interactions for ab initio folding of protein terminal regions with secondary structures.," *Proteins,* vol. 72, pp. 793-803, 2008.

[19] H. Zhou and J. Skolnick, "GOAP: A Generalized Orientation-Dependent, All-Atom Statistical Potential for Protein Structure Prediction.," *Biophys. J*. vol. 101, pp. 2043-2052, 2011.

[20] M. T. Hoque and S. Iqbal, "Genetic algorithm-based improved sampling for protein structure prediction," *International Journal of Bio-Inspired Computation,* vol. 9, pp. 129-141, 2017.

[21] D. Bhandari, C. A. Murthy, and S. K. Pal, "Genetic Algorithm with Elitist Model and its Convergence," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 10, pp. 731-747, 1996.

[22] D. University of California. *Critical Assessment of Protein Structure Prediction 8 (CASP8)*. Available: http://predictioncenter.org/casp8/index.cgi

[23] U. o. Washington. (February 2017). *Robetta Full-chain Protein Structure Prediction Server*. Available: http://robetta.bakerlab.org/

[24] J. Xu and Y. Zhang, "How significant is a protein structure similarity with TM-score=0.5?," *Bioinformatics,* vol. 26, pp. 889-895, 2010.

[25] Y. Zhang and J. Skolnick, "Scoring function for automated assessment of protein structure template quality," *Proteins: Struct., Funct., Bioinf.,* vol. 57, pp. 702-710, 2004.

[26] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers,* vol. 22, pp. 2577-2637, 1983.

[27] R. Heffernan, K. Paliwal, J. Lyons, A. Dehzangi, A. Sharma, J. Wang*, et al.*, "Improving prediction of secondary structure, local backbone angles, and solvent accessible surface area of proteins by iterative deep learning," *Scientific Reports,* vol. 5, 2015.

[28] R. Samudrala and J. Moult, "An All-atom Distance-dependent Conditional Probability Discriminatory Function for Protein Structure Prediction.," *Journal of Molecular Biology,* vol. 275, pp. 895-916, 1998.

[29] M. T. Hoque, Y. Yang, A. Mishra, and Y. Zhou, "sDFIRE: Sequence-specific statistical energy function for protein structure prediction by decoy selections.," *Journal of Computational Chemistry,* vol. 37, pp. 1119-1124, 2016.

[30] S. Liang, D. Zheng, C. Zhang, and D. M. Standley, "Fast and accurate prediction of protein side-chain conformations," *Bioinformatics,* vol. 27, pp. 2913-2914, 2011.

[31] D. Lab. (1969, February 2014). *Taking Input Parameters for Culling Whole PDB*. Available: http://dunbrack.fccc.edu/Guoli/PISCES_ChooseInputPage.php

[32] E. Faraggi, Y. Yang, S. Zhang, and Y. Zhou, "Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction," *Structure,* vol. 17, pp. 1515-1527, 2010.

[33] D. University of California. (2017). *Critical Assessment of Protein Structure Prediction 9 (CASP9)*. Available: http://predictioncenter.org/casp9/index.cgi

[34] D. University of California. *Critical Assessment of Protein Structure Prediction 10 (CASP10)*. Available: http://predictioncenter.org/casp10/index.cgi

[35] D. University of California. *Critical Assessment of Protein Structure Prediction 11 (CASP11)*. Available: http://predictioncenter.org/casp11/index.cgi

[36] Y. Zhang. (January 2017). *CASP8 Native Structures*. Available: https://zhanglab.ccmb.med.umich.edu/casp8/native.html

[37] Y. Zhang. (January 2017). *CASP9 Native Structures*. Available: https://zhanglab.ccmb.med.umich.edu/casp9/native.html

[38] Y. Zhang. (January 2017). *CASP10 Native Structures*. Available: https://zhanglab.ccmb.med.umich.edu/casp10/native.html

[39] Y. Zhang. (January 2017). *CASP11 Native Structures*. Available: https://zhanglab.ccmb.med.umich.edu/casp11/native.html