International Journal of Advanced Research in IT and Engineering

ISSN: 2278-6244
Impact Factor: 6.111

Survey on Approaches for Clustering of Protein Structure Decoys

Luibaiba Muhammad Kunhi, Department of CS & E, NMAMIT,Nitte, Karnataka, India

Raju K, Department of CS & E, NMAMIT, Nitte, Karnataka, India

Abstract*—Docking problem in drug design is generally solved (binding of a ligand to its receptor) by generating a large number of decoys of the putative complexes and then selecting the representative decoys by clustering. Knowledge of the tertiary structure of the protein is important in understanding its function. Computational approach for protein structure prediction involves generating a large number of decoy candidates from which representatives can be selected. Clustering based approach can be used for selecting representative candidates in drug designing and protein structure prediction systems. As the number of decoy structures becomes very large, faster approaches are required. Recently, several GPU-based parallel algorithms have been proposed by researchers. This paper is a survey on the existing approaches for clustering protein structure decoys.

***Keywords*—docking; protein structure**

I. INTRODUCTION

Clustering based approach can be used in protein structure prediction system and computer aided drug-designing systems for predicting the 3-dimensional structure of proteins. Computational approach is used for faster prediction instead of the traditional laboratory based techniques. The computational approach for predicting protein structure involves selecting representative structures from the large number of decoys that are generated. Clustering is method of assigning similar objects into groups based on similarity measure. The idea behind using clustering is that the nearest native structure has a high probability of being closer to the cluster with maximum population.

RMSD is the most commonly used similarity measure for measuring similarity between proteins. Consider 2 decoys d1 and d2 of a protein with N atoms and each atom of the decoy represented by a 3-Dimensional vector as given below.

$$d1: < x_{1i}, y_{1i}, z_{1i} > (i = 1, \dots, N)$$

$$d2: < x_{2i}, y_{2i}, z_{2i} > (i = 1, \dots, N)$$

The pairwise RMSD can be calculated using equation given below: $RMSD(d1, d2) =$

$$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_{1i}-x_{2i})^2 + (y_{1i}-y_{2i})^2 + (z_{1i}-z_{2i})^2} \qquad (1)$$

In most cases the number of decoy candidates can be very large necessitating development of faster algorithms. Recently, several parallel mechanisms based on GPU have been proposed by researchers. This paper is a survey on the existing approaches for protein clustering.

II. BACKGROUND

A. Issues in cluster based techniques

Several issues need to be considered while designing a cluster based approach for protein decoys.

- Selection of appropriate similarity measure for measuring protein structure similarity
- The number of representative candidates required i.e. number of clusters
- The threshold distance d, for assigning a decoy to a cluster.

III. LITERATURE SURVEY

In the I-TASSER[1] structure prediction approach, prediction steps involve selecting the cutoff value and then finding the decoy with the highest neighbors within the cutoff distance. This decoy is reported as the decoy with the highest rank. The same steps are then repeated by removing this decoy and its neighbors to find the next high ranked and so on. This method requires pairwise RMSD computation. If there are m decoys then a $m \times m$ matrix is required to store the pairwise RMSD. If the number of decoy structures is very large then it can exceed the memory.

SPICKER [2] provides a solution to the memory limitation by applying a shrinking technique. When the value of m is too large to fit into memory the decoy set is shrunk. Assuming that only 10^4 structures can be stored in memory, the decoy set is divided into subsets with each subset containing $m * 10^{-4}$ decoys. The decoy with lowest energy conformation is then selected from each subset. An iterative method is used for selecting the RMSD threshold. An initial threshold is set first and thereafter the threshold is determined by the current threshold and the ratio of number of decoys in the cluster with highest population to the total number of decoy structures in the set.

Wang.Q, Shang. Yi and Xu. D [3] has improvised SPICKER by applying a scoring function based filtering mechanism. Outlier decoys are filtered out using existing scoring functions like OPUS-C_α [4] and Model Evaluator.

Calibur [5] describes an approximate approach that is completely different from the previous approaches and uses 3 strategies to enable the clustering of large number of decoys. Instead of calculating the pairwise RMSD, Calibur uses approximate distance measures. Decoys are grouped based on their proximity and are considered collectively in determining if it is within the cutoff distance from a decoy in another grouping. This reduces the number of pairwise RMSD computations. Another strategy is the use of upper bound and lowers bounds for deciding if a decoy is within the cutoff distance from another decoy rather than computing the exact C_α-RMSD. This strategy is applied as a preliminary check to reduce the number of RMSD computations. Several ways of determining the bounds have been described. One such method requires the use of reference decoys and pre-computed RMSD between the decoy and the reference. With n references, n upper bounds and n lower bounds can be calculated. The third strategy applied is the filtering of outlier decoys. This is done by taking a random sample of 100 decoys from the set and removing those decoys that are not within a distance of twice the cutoff from any other decoys in the sample. By applying these 3 strategies, Calibur has obtained an improvement in performance over the SPICKER approach.

Zhang.J and Xu. Dong [6] has proposed a faster algorithm for clustering large number of decoys using an alternative distance measure instead of the pairwise RMSD. A new distance measure called as C-score is used which is based on contact map vector. Contact map vector is determined using the Euclidean distance between the atoms of the proteins. Given that A and B are the contact map vectors and ||A|| and ||B|| are the norm of A and B, C-score is calculated follows:

$$C_{AB} = \sqrt{1 - dot(\frac{A}{||A||}, \frac{B}{||B||})/2} \quad (2)$$

C-Score measure is highly correlated to RMSD and is much faster than the pairwise RMSD computation.

ONION [7]describes an algorithm for clustering large number of decoys in minutes. The algorithm tries to find the suitable cluster number based on the decoy information by evaluating all possibilities. Initially, centroids are constructed by random sampling of decoys. In the subsequent stages, assignment to the centroid is carried out.

HS-Forest [8] is an approach that is a combination of clustering and scoring function. However, clustering is done only partially. Every decoy is not assigned to a cluster. Only the representatives are selected based on similarity measures. HS-Forest (H-height and S-size) uses a tree based approach. Initially the entire decoy set is divided into 2 nodes of the HS-Tree. A random hashing function is used for the division. These nodes are further divided until the required number of nodes is obtained.

ClusCo [9] is tool that allows the comparison of protein structures based on several similarity measures. The computation of RMSD is done using CUDA based parallel method. ClusCouses parallel K-means algorithm implemented with OpenMP and also a serial version of hierarchical agglomerative clustering.

Fast_protein_cluster [10] provides a faster method for clustering of nutritious rice protein data generated by the world project. It is an OpenCL based parallel approach for parallelizing RMSD computation and TM_Score Computation. It also uses K-means algorithm and hierarchical agglomerative clustering and is significantly faster and more accurate compared to ClusCo.

Dang. H et. al. [11] has proposed a CUDA based approach for parallelizing RMSD computation. [11] uses hierarchical ward clustering. Three approaches of parallelizing the computation of ward distance have been proposed.

We can analyze these approaches based on the following criterion:

A. Clustering Methods

Exact clustering is the approach wherein prediction is done by selecting a threshold value and then finding the decoy with maximum number of neighbors within the threshold distance. The same steps are then repeated by removing this decoy and its neighbors to find the next high ranked and so on. I-TASSER, SPICKER and [3] uses this approach.

ClusCoand Fast_protein_cluster has used K-Means algorithm and hierarchical agglomerative clustering.[11] has used hierarchical ward clustering. HS-Forest is based on partial clustering which avoids the overhead of assigning each decoy to a cluster.

B. Filtering Methods Applied

When the number of decoy structures becomes very large, pairwise RMSD computation can become time consuming and exceed the memory. I-TASSER does not apply any filtering techniques. SPICKER applies a decoy shrinking technique to filter outliers. [3] uses existing scoring functions to filter outliers. Calibur uses 2 strategies to filter out bad structures. One based on upper and lower bounds and other based on random sampling.

C. Similarity Measure

Pairwise RMSD is the most common distance measure and is used by almost all the approaches. [6] uses a distance measure called as the C-score which is highly correlated to RMSD. Calibur uses an approximation of the pairwise RMSD instead of calculating the actual value.

ClusCo also provides comparison of protein structures using GDT_TS, TM-Score, MaxSub and dRMSD in addition to pairwise RMSD.Fast_protein_cluster also provides comparison using TM-Score.

D. Selection of the threshold value

SPICKER provides an iterative approach for determining the threshold that is used to determine the membership of a decoy to cluster. [3] has used the same approach as used by SPICKER. However, an initial cutoff needs to be experimentally determined.

Calibur automatically discovers the threshold distance from the input decoys, if it is not explicitly stated. Several methods for discovering threshold is provided by Calibur. The default strategy used is to find the cutoff in such a way that only a certain percentage of pairwise RMSD is below the cutoff. A heuristic approach is used to determine this percentage. Other strategies include the use of most frequently occurring RMSD to determine this threshold.

In [6], it is possible that any 2 structures with a large difference in RMSD to have closer values for score. Therefore a refinement strategy is used wherein a centroid is calculated from decoys with similar C-score values. Assignment to cluster is done only for decoys that are closer to the centroid.

ONION eliminates the need to compute the threshold by determining optimal cluster number using distance information. ClusCo does not specify any method for selecting threshold and therefore it has to be specified along with the input.[11] uses minimum distance criteria for assigning a decoy to cluster and therefore there is no need to select a threshold. However, there is a need to find optimal cluster number. This applies for ClusCo and [10] as well.

E. Use of parallel methods

GPUs have the ability to provide a good improvement in performance for applications have large number of computations. RMSD computation is an ideal candidate for parallelization. ClusCo, Fast_protein_cluster and [11] have applied parallel solutions for the same.

F. Handling large number of protein decoy structures

SPICKER cannot handle large number of decoy structures due to memory limitation. It can handle only up to 13,000 decoy structures. Calibur is capable of handling a large number of decoy structures. [11] and Fast_protein_cluster can handle more than 500,000 decoy structures.

IV. CONCLUSIONS AND FUTURE WORK

Faster computational approach for clustering of protein decoys is essential in many structure prediction systems. This paper is an extensive survey on the several existing approaches. The future work would be development of an efficient parallel approach for the same.

REFERENCES

[1] Wu S, Skolnick J, and Zhang Y. “Ab initio modeling of small proteins byiterative TASSER simulations.” BMC Biology 2007, 5(17).

[2] Zhang Y, Skolnick J: SPICKER.“Approach to clustering protein structures for near-native model selection.” J ComputChem 2004, 25:865-871.

[3] Qingguo Wang, Yi Shang and Dong Xu, “A New Clustering-Based Method for Protein Structure Selection.” International Joint Conference on Neural Networks, IEEE 2008

[4] Y. Wu, M. Lu, M. Chen, J. Li and J. Ma, ”OPUS-Ca: A knowledge-based potential function requiring only Ca positions,” Protein Science,vol. 16, pp. 1449-1463, 2007.

[5] Li and Ng: Calibur. “A tool for clustering large numbers of protein decoys.” BMC Bioinformatics 2010 11:25.

[6] JingfenZhang and Dong Xu.“Fast algorithm for clustering a large number of protein structural decoys.”IEEE International Conference on Bioinformatics and Biomedicine, 2011

[7] Shuai Cheng Li, Dongbo Bu, and Ming Li, “Clustering 100,000 Protein Structure Decoys in Minutes.” IEEE/ACM Transactions on Computational Biology and bioinformatics, vol. 9, 2012.

[8] Zhou and Wishart. “An improved method to detect correct protein folds using partial clustering.” BMC Bioinformatics 201314:11.

[9] Jamroz and Kolinski. “ClusCo: clustering and comparison of protein models.” BMC Bioinformatics 2013 14:62.

[10] Ling-Hong Hung* and Ram Samudrala. “fast_protein_cluster: parallel and optimized clustering of large-scale protein modeling data.” BMC Bioinformatics 2014.

[11] Hoang-Vu Dang, Bertil Schmidt, Andreas Hildebrand, Anna Katharina Hildebrandt. “Parallelized Clustering of Protein Structures on CUDA-enabled GPUs.” 22nd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing. IEEE 2014.