International Journal of Computational Intelligence and Informatics, Vol. 2: No. 1, April - June 2012

Design and analysis of Genetic Clustering Bee Colony Optimizaiton for Flexible Protein-Ligand Docking

C. P. Chandran
Associate Professor of Computer Science and
Head, Post Graduate Department of Bioinformatics
Ayya Nadar Janaki Ammal College
Sivakasi, India
drcpchandran@gmail.com

E. Kiruba Nesamalar
Full-time Research Scholar (M.Phil)
Post Graduate Department of Computer Science and Information Technology
Ayya Nadar Janaki Ammal College
Sivakasi, India
kirubanesamalar@gmail.com

***Abstract*-In this paper, the design and analysis of Genetic Clustering Bee Colony Optimization for Flexible Protein-Ligand docking is carried out. The molecular docking problem is to find a good position and orientation for docking and a small molecule ligand to a large receptor molecule. It is originated as an optimization problem consists of optimization method and the clustering technique. Clustering is a data mining task which groups the data on the basis of similarities among the data. Genetic Algorithm (GA) is one of the evolutionary algorithms inspired by biological evolution and utilized in the field of clustering. K-median clustering is a variation of K-means clustering where instead of calculating the mean for each cluster to determine its centroid, one instead calculates the median. A Genetic Clustering algorithm combine a GA with the K-medians clustering algorithm. Genetic Clustering is combined with Bee Colony Optimization (BCO) algorithm to solve Molecular docking problem. BCO is a Swarm Intelligent algorithm that was first introduced by Karaboga. It is based on the Fuzzy Clustering with Artificial Bee Colony Optimization algorithm proposed by Dervis Karaboga and Celal Ozturk. In this work, a new algorithm called Genetic Clustering Bee Colony Optimization (GCBCO) is proposed. The performance of GCBCO is tested in 10 docking instances from the PDB bind core set and compared the performance with PSO and ACO algorithms. The result shows that the GCBCO could find ligand poses with best energy levels than the existing search algorithms.**

Keywords-Data Mining, BCO, K-medians, Genetic algorithms, Molecular docking, Protein-Ligand docking.

I. INTRODUCTION

A. Data Mining

Data mining is defined as the non-trivial process of searching and analyzing data in order to find implicit but potentially useful information. Let D = {d_1... d_n} be the dataset to be analyzed. The data mining process is described as the process of finding a subset D' of D and hypotheses H_U(D',C) about D' that a user U considers useful in an application context C. D' have fewer data elements than D, but it also have a lower dimensionality (m'). In databases the data is partitioned into relations or object classes [1]. D is considered as a union of relations R_1... R_k each has its own dimensionality (m_1... m_k).The hypotheses expressing interesting aspects of the data deal with the whole database or with a single relation (D'=D or D'= R_i) they deal with real subsets of the database (D'⊂ D with |D'| < |D| and |D'| sufficiently large) or with single exceptional data items, (D'⊂ D with |D'|=1 or sufficiently small when compared to |D|). Among others, hypotheses that hold for all or most $e_i \in D'$, (D'⊆ D), classifications of D' into classes C_i with different properties P_i [$P_i(e_1) \neq P_j(e_2) \Rightarrow e_1 \in C_i \wedge e_2 \in C_j \wedge i \neq j$], functional dependencies F or relationships R between two or more dimensions [$d_{i1} = F(d_{i2}, ..., d_{il})$ or $R(d_{i1}, ..., d_{il})$, $l \leq m$] [2].

Clustering

Clustering is a data mining technique used to place data elements into related groups without prior knowledge of the group definitions. It is the process of grouping data objects into a set of disjoint classes called clusters, so that objects within a class have high similarity to each other while objects in separate classes are more dissimilar. We consider a dataset X consisting of data points x_i ($1 \leq i \leq N$) where xi = {a_{i1}, a_{i2},

ISSN: 2349 - 6363

a_{i3}.....a_{id}}and $a_{ii} \in A$ is a numerical or nominal attribute from the attribute space A. Clustering is an unsupervised classification. It refers to a procedure that assigns data objects to a set of classes. Unsupervised means clustering does not depend on predefined classes while classifying the data objects [3].

K-medians Clustering

Given a dataset N of nodes a distance function d: $N^2 \rightarrow R$ and an integer k, find k element subset of N as medians such that sum of distances from each node to its nearest median is minimal. The nodes that are closer to a median form a cluster. The node is assigned to its nearest median. K-median clustering is a variation of K-means clustering where instead of calculating the mean for each cluster to determine its centroid one instead calculates the median. This has the effect of minimizing error over all clusters with respect to the 1-norm distance metric, as opposed to the square of the 2-norm distance metric. This relates directly to the K-medians problem which is the problem of finding k centers such that the clusters formed by them are the most compact. A set of data points x, the k centers c_i are to be chosen so as to minimize the sum of the distances from each x to the nearest c_i. The criterion function formulated in this way is sometimes a better criterion used in the k-means clustering algorithm in which the sum of the squared distances is used [4].

Genetic Algorithm (GA)

GA is a population based search technique used to find appropriate solutions to optimization and search problems. A random population of individuals is generated by initializing each individual as a vector composed of a set of uniformly distributed random values between the minimum and maximum x, y and z values. The genes representing torsion angles are given random values between -180 and +180. The fitness value of an individual is the binding energy between ligand and the target protein [5]. Mutation operator is performed by adding a random real number that has a Cauchy distribution to the variable where α and β are parameters that affect the mean and spread of the distribution. Combining respective characteristic of the GA and the clustering algorithm the key of the clustering is find out using cluster centers but GA is provided with characteristic of global optimum search. Firstly the clustering centers which are keeping with the global characteristic are automatically selected by utilizing GA and then other data points are distinguished by the clustering algorithm. The clustering analysis results corresponding to the global distribution characteristic is produced [6].

Swarm Intelligence (SI)

The behavior of a single ant, bee, termite and wasp often is too simple but their collective and social behavior is of paramount significance. The phrase Swarm Intelligence (SI) was proposed by Beny and Wang in the context of cellular robotics. SI systems are typically made-up of a population of simple agents and entity capable of performing or executing certain operations [7]. Although there is normally no centralized control structure dictating and individual agents should behave local interactions between such agents often lead to the emergence of global behavior. Many biological creatures such as fish schools and bird flocks clearly display structural order with the behavior of the organisms so integrated that even though they may change shape and direction they appear to move as a single coherent entity proposed by Couzin.

Ant Colony Optimization (ACO)

The ants behavior is shown in Fig.1. ACO is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs. This algorithm is a member of the ant colony algorithms, in SI methods, and it constitutes some metaheuristic optimizations. Initially proposed by Marco Dorigo, the first algorithm was aiming to search for an optimal path in a graph, based on the behavior of ants seeking a path between their colony and a source of food. The original idea has since diversified to solve a wider class of numerical problems, and as a result, several problems have emerged, drawing on various aspects of the behavior of ants.

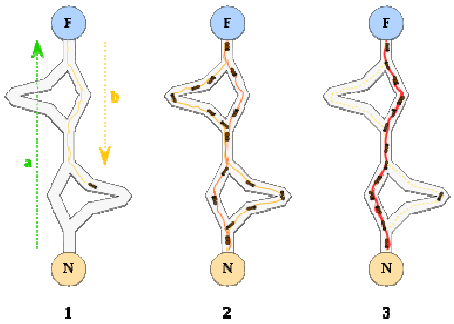

Figure 1. Behavior of Ants

a) The first ant finds the food source, then returns to the nest, leaving behind a trail pheromone

b) Ants indiscriminately follow four possible ways, but the strengthening of the runway makes it more attractive as the shortest route.

c) Ants take the shortest route; long portions of other ways lose their trail pheromones.

The basic idea of the algorithm involves the movement of a colony of ants through the different states of the problem influenced by two local decision policies, trails and attractiveness.

Particle Swarm Optimizaiton (PSO)

PSO is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. PSO optimizes a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple particle's position and velocity. Each particle's movement is influenced by its local best known position and is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions. PSO is originally designed by Kennedy and Eberhart for simulating social behaviour, as a stylized representation of the movement of organisms in a bird flock or fish school. The algorithm was simplified and it was observed to be performing optimization.

Bee Colon y Optimizaiton (BCO)

The general BCO algorithm is given in Fig. 2. BCO is an optimization algorithm based on the intelligent for aging behaviour of honey bee swarm proposed by Karaboga [8]. In ABC model the colony consists of three groups of bees: employed bees, onlookers and scouts. It is assumed that there is only one artificial employed bee for each food source. The number of employed bees in the colony is equal to the number of food sources around the hive. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source [9].

Algorithm:-

1. Initialization: Determine the number of bees B, and the number of iterations I. ST = $\{s_{t1}, s_{t2}\ldots s_{tm}\}$. Find any feasible solution x of the problem. This solution is the initial best solution.
2. Set i: = 1. Until i = I,
3. Set j = 1. Until j = m,
4. Forward pass: Allow bees to fly from the hive and to choose B partial solutions from the set of partial solutions S_j at stage s_{tj}
5. Backward pass: Send all bees back to the hive. Allow bees to exchange information about quality of the partial solutions created. Set j = j+1.
6. If the best solution x_i obtained during the i^{th} iteration is better than the best- known solution, update the best solution x: = x_i
7. Set i= i+1;

Figure 2. General BCO Algorithm

ABC is a population based algorithm the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality fitness of the associated solution. At the first step, a randomly distributed initial population food source positions is generated. After initialization, the population is subjected to repeat the cycles of the search processes of the employed, onlooker and scout bees. After all employed bees complete the search process they share the position information of the sources with the onlookers area. Each onlooker evaluates the nectar information taken from all employed bees and then chooses a food source depending on the nectar amounts of sources. As in the case of the employed bee it produces a modification on the source position in memory and checks its nectar amount. Providing that nectar is higher than that of the previous one the bee memorizes the new position and forgets the old one [10].

Molecular Docking

Docking is a method which predicts the preferred orientation of one molecule to a second when bound to each other to form a stable complex [11]. Knowledge of the preferred orientation in turn may be used to predict

the strength of association or binding affinity between two molecules. The associations between biologically relevant molecules such as proteins, nucleic acids, carbohydrates and lipids play central role in signal transduction. Docking is frequently used to predict the binding orientation of drug candidates to their protein targets in order to predict the affinity and activity of the small molecule. Hence docking plays an important role in the rational design of drugs [12]. The aim of molecular docking is to achieve an optimized conformation for both the protein and ligand and relative orientation between protein and ligand so that the free energy of the overall system is minimized.

Molecular Docking method can be divided into two categories rigid docking and flexible docking. Both the ligand and the protein are treated as rigid objects in the process of rigid docking, while flexible docking treat the ligand as an articulated object and the protein as a rigid object. Docking method consists of two components one is scoring and the other one is optimization problem. Scoring functions are fast approximate mathematical methods used to predict the strength of the non-covalent interaction between two molecules after they have been docked. It is referred to as binding affinity [13]. Finding functional sites on protein molecular surfaces is crucial for revealing the mechanisms of molecular signaling involving target proteins. Ligand binding sites are among the most promising targets for drug candidates, whose actions depend upon the inhibition or regulation of the target protein functions. However, in some cases such ligand-binding sites must be predicted because little or no experimental information about the protein's functional sites exists [14].

Protein-Ligand Docking

Protein-Ligand docking problem was first formulated by Fischer using the famous lock-and-key metaphor. The key ligand must fit exactly into the lock protein. But molecules are not rigid objects this description is too limited and at least the flexibility of the ligand must be taken into account which is done by almost all state-of-the-art docking algorithms. But even the protein conformation can adapt to an incoming ligand leading to the so-called induced fit model [15]. Protein docking is a method that predicts the bound conformation of one protein to another protein or a ligand. A docking algorithm aims to find the best orientation of these two molecules such that they have the minimum binding energy as scored by a predefined scoring function. Good scoring function with high selectivity and efficiency that distinguishes between correctly or incorrectly docked structures and a search algorithm that can efficiently do global minimization of the scoring function [16].

In early docking algorithms, both protein and ligand are considered as rigid bodies and they have only six degrees of translational and rotational freedom to search for best orientations. Since the number of degrees of freedom is large if the proteins are modeled as flexible, it is impractical to perform exhaustive conformational search. Most of current docking algorithms consider the flexibility of ligands to find the best binding position between small molecules (ligands) such as substrates or drug candidates and structurally known target proteins in Fig. 3.

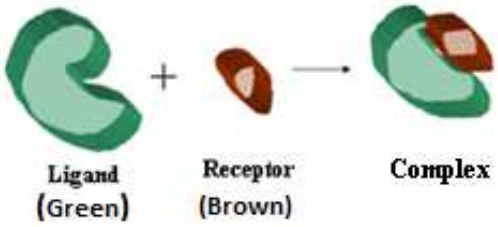

Figure 3. The docking of molecule ligand (green) to a protein receptor (brown) to produce a complex.

The number of optimization variables is composed of six degrees of freedom for rotation and translation plus the number of torsion angels. The ligand finds its position into the protein's active site after a certain number of moves in its conformational space. Flexibility modeling allows the ligand to change its structure with the torsions angles. Each move costs energy and after moves are completed total energy is computed by the system. The goal is to minimize this binding energy to find the best conformation. Very large databases ranging in size from thousands to millions of ligands can be tested by ligand or structure-based virtual screening techniques and different lead-optimization routes can be tried without ever synthesizing most of the compounds. Structure-based approaches start with a known 3D structure of a protein. These structures are obtained by experimental techniques like X-ray crystallography or NMR-spectroscopy and are,publicly available from the Protein Data Bank. Then a docking algorithm tries to solve the pose prediction problem. This problem consists in finding the correct orientation and conformation of the ligand within a priori known active site of the protein [17].

II. PREVIOUS WORKS

Guha et.al., [18] proposed Fast Genetic K-means cluster technique (FGKA). It is a faster version of GKA and FGKA that features several improvements over GKA including an efficient evaluation of the objective

value TWCV (Total Within-Cluster Variation), avoiding illegal string elimination overhead, and a simplification of the mutation operator. Mohammad Ali has proposed [19] a hybrid genetic k-means algorithm (HGKA). HGKA combines the benefits of FGKA and performs well in smaller and larger mutation probability. Basic foundations of these GA based clustering techniques are k-means clustering and it can deal only numeric datasets. The book by Kennedy and Eberhart describes many philosophical aspects of PSO and Swarm Intelligence. An extensive survey of PSO applications is made by Poli. Artificial Bee Colony (ABC) algorithm is a SI method proposed by Karaboga which simulates intelligent foraging behavior of honey bees. Dervis Karaboga and Celal Ozturk [20] have proposed Fuzzy Clustering with BCO. ABC algorithm fuzzy clustering classifies the different datasets. Cancer, Diabetes and Heart from UCI database, a collection of classification benchmark problems. Bonding as a swarm Applying Bee Nest Site Selection behavior to Protein Docking algorithm proposed by Konrad Diwlod, Daniel Himmelbach and Rene Meier [21] Bee Nest-Site Selection Optimization, which solves optimisation problems using a novel scheme inspired by the nest-site selection behaviour found in honeybees Moreover, this is the first BNSO algorithm.CABC algorithm is used for optimizing six widely used benchmark functions and the comparative results produced by ABC and PSO. E.Kiruba Nesamalar and C.P Chandran [22] proposed Fuzzy clustering with Ant Colony Optimization for Protein-Ligand Docking. The ACO is a probabilistic technique for solving computational problems which can be reduced to finding good paths through graphs. FCACO is based on the grouping behaviour of ants.

III. PROBLEM DEFINITION

The overview of docking is shown in Fig. 4. The optimization algorithm travel through the search space and finding the lowest binding energy for this problem. The requirements for docking are ligand structure (For E.g. XK-263), Protein Structure (For E.g. HIV-1 Protease (1hvr)), and docking algorithm (GCBCO).

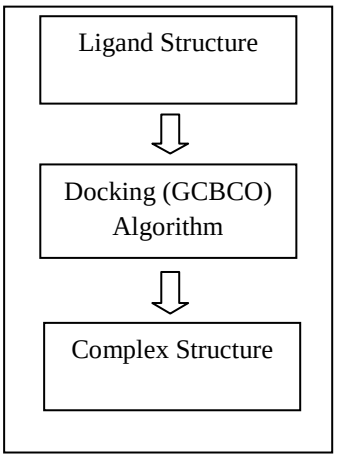

Figure 4. Overview of Docking

IV. METHODOLOGY USED

The methodology used for Protein-Ligand Docking is given in Fig. 5. GCBCO combines a K-medians clustering algorithm and BCO. The K-medians is chosen, because it can be computed using certain mathematical operations so that it can deal with very large datasets and is more robust than the k-means. Optimization problem is the problem of finding the best solution from all feasible solutions. BCO is a relatively best approach to solving optimization problems based on the foraging behaviour of bees. These algorithms are applied to solve the Flexible Protein-Ligand docking problem to find the best target protein.

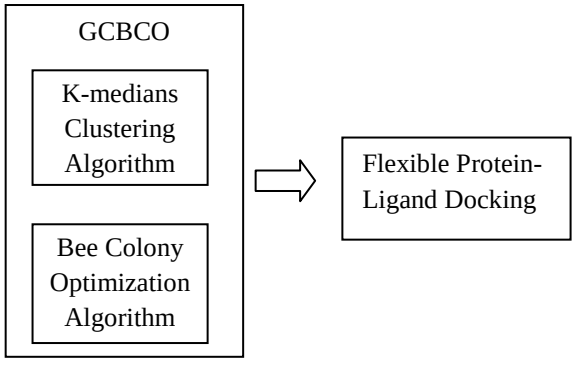

Figure 5. Methodology used for Molecular Docking

V. ALGORITHM USED

A. K-medians Clustering Algorithm

General K-medians algorithm is shown in Fig. 6. K-median clustering is a variation of K-means clustering where instead of calculating the mean for each cluster to determine its centroid, one instead calculates the median. This has the effect of minimizing error over all clusters with respect to the 1-norm distance metric, as opposed to the square of the 2-norm distance metric which k-means does.

Algorithm:-

1) Let N be the set of input data
2) Let d_j be the demand of $j \in N$. This demand may be considered as the number of points at node j. In some problem definitions d_j is not taken into account in the formulation, which can be formulated as letting $d_j = 1$ for all j.
3) Let c_{ij} be the cost of assigning $j \in N$ to the median $i \in N$
4) Let x_{ij} be the final assignment of $j \in N$ to the median $i \in N$. It is a 0-1 variable, and when $x_{ij} = 1$ means that node j is assigned to i.
5) Let y_i be a 0-1 variable indicating whether node i is selected as a median.

Figure 6. Original K-medians Algorithm

B. Genetic K-medians Algorithm

Genetic K-medians clustering is shown in Fig. 7. Genetic K-medians algorithm maintains a population of coded solutions. The population is initialized randomly and is evolved over generations; the population in the next generation is obtained by applying genetic operators on the current population. The evolution takes place until a terminating condition is reached. The genetic operators that are used in Genetic K-medians algorithm are the selection, the distance based mutation and the K-medians operator.

Algorithm :-

```
Input: Mutation Probability, P_m;
       Population size, N;
       Maximum number of generation, MAX_GEN;
Output: Solution string, s*
 {   Initialize the population, P;
     geno = MAX_GEN;
     s*= P_1; (P_i is the i^th string in P)
while (geno > 0)
{    Calculate Fitness values of strings in P;
     P = Selection (P);
     for i = 1 to N, P_i = Mutation(P_i);
     for i = 1 to N, K-medians(P_i);
     s = string in P such that the corresponding weight
     matrix Ws has the minimum SE measure;
     if (S(W_s) > S(W_s)), s*= s;
    geno = geno - 1;}
output s*;}
```

Fig. 7 Original Genetic K-medians Algorithm

C. Proposed GCBCO Algorithm

The BCO is a new population-based metaheuristic approach, initially proposed by Karaboga and further developed by Karaboga and Basturk. It has been used in various complex problems. The algorithm simulates the intelligent foraging behavior of honey bee swarms.

Flowchart for GCBCO is shown in Fig. 8. In the initialization phase, the GCBCO algorithm generates a randomly distributed initial food source positions of S solutions, where S denotes the size of employed bees or onlooker bees. Each solution x^i (i=1, 2... S) is a D-dimensional vector where, D is the number of optimization parameters. And then evaluate each nectar amount fit_i. In the employed bees' phase, each employed bee finds a new food source v_i in the neighborhood of its current source $x_{i.}$ using (1).

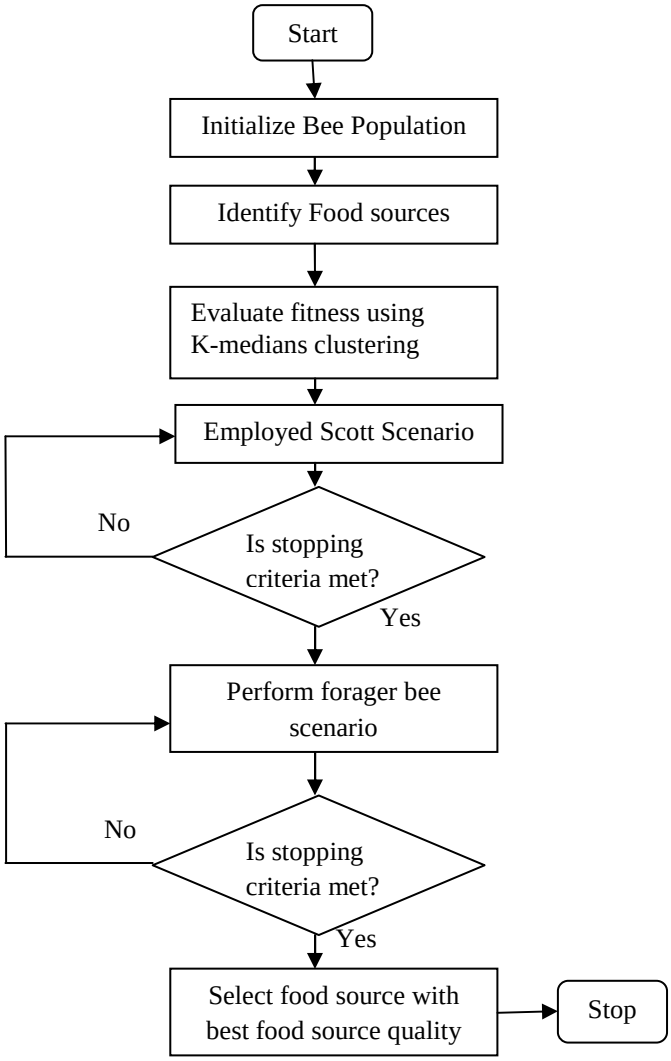

Figure 8. Flowchart for GCBCO Algorithm

$$v_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (1)$$

where $k \in (1, 2... S)$ and $j \in (1, 2... D)$ are randomly chosen indexes, and $k \neq i$.φ_{ij} is a random number between [−1, 1]. And then employed bee compares the new one against the current solution and memorizes the better one by means of a greedy selection mechanism. In the onlooker bee's phase, each onlooker chooses a food source with a probability which is related to the nectar amount fitness of a food source shared by employed bees. Probability is calculated using (2).

$$p^i = \frac{fit_i}{\sum_{n=1}^{SN} fit_i} \quad (2)$$

In the scout bee phase, if a food source cannot be improved through a predetermined c, called "limit", it is removed from the population, and the employed bee of that food source becomes scout. The scout bee finds a new random food source position using (3) and (4).

$$d = \sum_{i=1}^{n} | x_i - y_i | \tag{3}$$

$$perf(X, c) = \sum_{i=1}^{n} Min\{|| x_i - y_i ||^2 1 = 1.K\} \tag{4}$$

These steps are repeated through a predetermined number of cycles, called Maximum C Number or until a termination criterion is satisfied. proposed GCBCO Algorithm is given in Fig. 9.

Algorithm:-

C=1
Initialize the food source positions x^i, i = 1...S
Evaluate the nectar amount (fitness function fit_i) of food sources.

Main steps of the Fitness Function:
For data vector x_i
Calculate the Manhattan distance by using d
Assign x_i to the closest median cluster c_j.
Calculate the measure function using perf.
End For.
Return value of the fitness function.
Repeat

Employed Bees' Phase
For each employed bee
Replace the j component of the g best by using the j component of bee i
Calculate the f [new g best]
End For
For employed bee i produce new food source positions v_i
Calculate the value fit_i
Apply greedy selection mechanism
End For.
Calculate the probability values p_i for the solution.

Onlooker Bees' Phase
For each onlooker bee
Chooses a food source depending on p_i
Produce new food source positions v_i Calculate the value fit_i
Apply greedy selection mechanism
End For

Scout Bee Phase
If there is an employed bee becomes scout
Then replace it with a new random source positions
Memorize the best solution achieved so far
c=c+1.
Until c = Maximum C Number

Figure 9. Proposed GCBCO Algorithm

VI. IMPLEMENTATION

Most of the docking tools (AutoDock, ClusDock, FlexX) are implemented in C or C++. GCBCO is implemented using C#. Only three steps are required to start docking. Users must characterize a protein structure, one or several ligands and docking type. Several sample files are supplied to users and can be directly uploaded into the form simply by clicking on a link. Three steps are involved to submit docking. In target selection, users upload the coordinate files of protein structures through GCBCO interface, or enter the PDB codes of the respective structures. In Ligand selection, a ligand can be selected either by specifying its identifier from the ZINC database or by uploading the structure files. Three docking parameters can be used. Accurate, Fast and very fast. These parameters are adjusted to reach the desired docking time.

GCBCO Algorithm starts with an initialization of the food source positions x_i, i=1…s and then evaluate the fitness function using Manhattan distance. Manhattan distance is used to find the ligand and the receptor pose value. Three Phases are used in this algorithm Employed Bees, Onlooker Bee, and Scout Bee Phase. At the first step, a randomly distributed initial population is generated. After initialization, the population is subjected to repeat the cycles of the search processes of the employed, onlooker, and scout bees. Here the employed bee represents an environment and each position in the search space corresponds to a potential solution. Using the principles of Greedy selection the colony tries to find a better quality than its current location. In the onlooker bee phase chooses a docked conformation from the employed bee (p_i) and produce new value (v_i). The selection process begins with scouts trying to find potential solution in the surroundings of the receptor's current location. If the scouts are able to find a location that is of acceptable quality, they report it to the employed bee. The GCBCO repeats the selection process until the correct docked conformation is found.

The method for finding lowest binding energy includes three parts, intramolecular energies for ligand, intramolecular energies for protein and intermolecular energy. The force field is showed in (5).

$$\Delta G = (V_{bound}^{L-L} - V_{unbound}^{L-L}) + (V_{bound}^{P-P} - V_{unbound}^{P-P}) + (V_{bound}^{P-L} - V_{unbound}^{P-L} + \square S_{conf}) \quad (5)$$

L refers to the "ligand" and P refers to the "protein" in a protein-ligand complex. The first two terms are intramolecular energies for the bound and unbound states of the ligand, and the other two terms are intramolecular energies. The last term denotes the change in intermolecular energy between the bound and unbound states.

VII. Results And Discussion

The GCBCO algorithm was evaluated using 10 test cases. Three steps are involved to start a docking process Target selection, Ligand selection and the docking parameters. A target protein structure can be used either by PDB code or by uploading structure files. A ligand can be selected either by zinc code database or uploading the ligand structure files. Docking parameters are used to adjust the desired docking time and search. The best results in terms of binding energy obtained by various algorithms are given in Table I. Docking accuracy is given in Table II. We have obtained the lowest energy in all the 10 test cases. This GCBCO algorithm gives a better performance than previous algorithms. The screenshot of ligand and protein selection is shown in Fig.10. Docking result of 1hvr protein is given in Fig.11. Graphical representation of the average convergence performance for the docking problems is shown in Fig.12. Docking accuracy of four methods is shown in Fig.13.

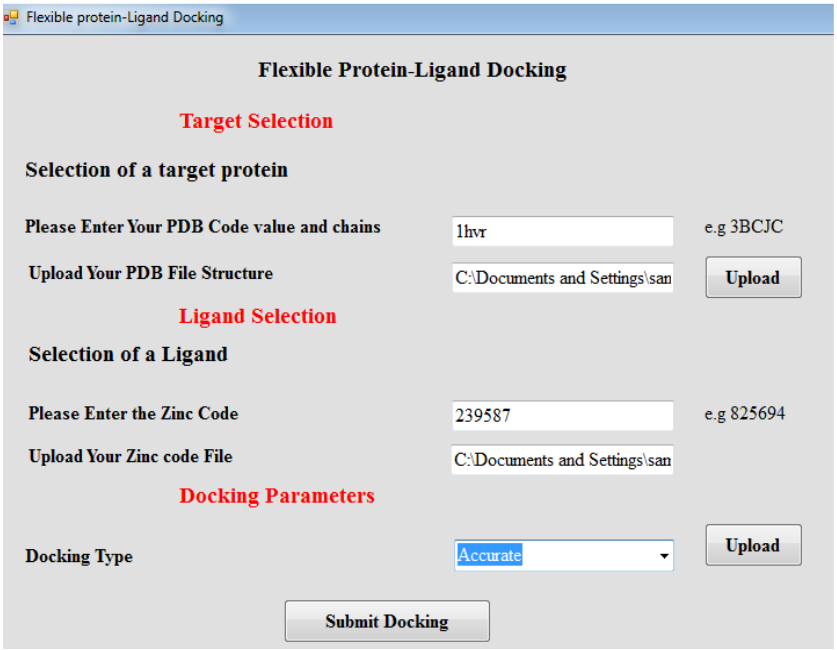

Figure 10. Screenshot of Ligand and Protein selection

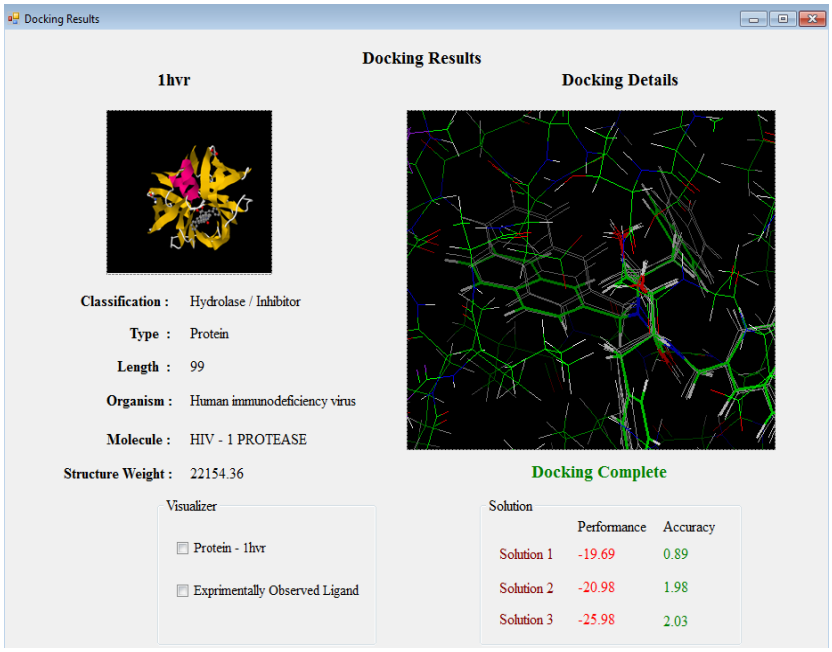

Figure 11. Docking result of 1hvr protein

TABLE I. COMPARISON OF LOWEST BINDING ENERGY

PDB	Torsion	Binding Energy			
		Methods			
		ACO	PSO	FCBCO	GCBCO
1hvr	10	-17.52	-16.73	-17.51	-19.69
1nnb	9	-8.71	8.15	-8.07	-9.11
2dgl	6	-12.56	-12.32	-12.49	-12.85
1stp	5	-7.23	-7.78	-7.62	-7.86
7abp	4	-8.40	-9.93	-10.58	-11.77
1tnh	2	-5.92	-6.07	-7.89	-7.47
3ptb	0	-6.28	-6.31	-6.46	-7.78
1cdg	12	-9.08	-10.84	-11.97	-12.84
1qbt	12	-11.45	12.8	-13.6	-13.59
hmg	11	-8.92	-9.49	10.96	-11.11

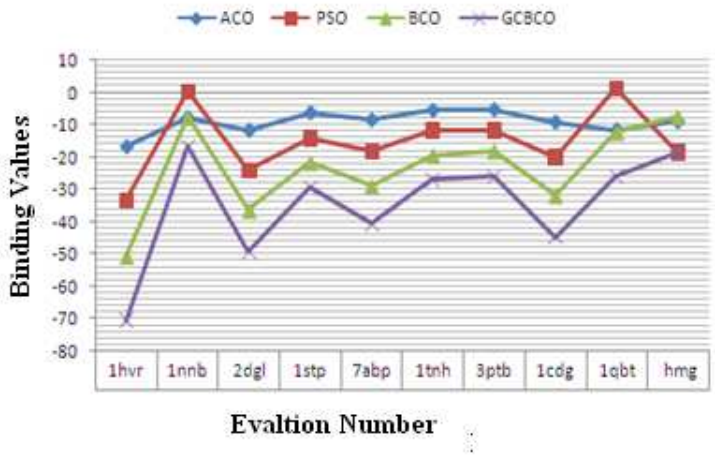

Figure 12. Comparison of Average Converge Performance of ACO, PSO, BCO and GCBCO methods

TABLE II. Docking Accuracy of PSO, ACO, FCBCO and GCBCO

PDB	Torsion	Docking Accuracy			
		GCBCO	FCBCO	ACO	PSO
1hvr	10	0.66	0.78	6.47	9.06
1nnb	9	0.72	0.84	8.58	3.44
2dgl	6	0.75	1.09	7.79	2.61
1stp	5	0.66	0.89	1.49	1.82
7abp	4	0.88	0.97	1.67	1.93
1tnh	2	0.85	0.51	1.56	0.19
3ptb	0	0.55	0.40	1.00	0.39
1cdg	12	0.96	0.37	3.98	1.51
1qbt	12	0.33	0.54	1.33	0.76
hmg	11	0.50	0.62	0.75	0.79

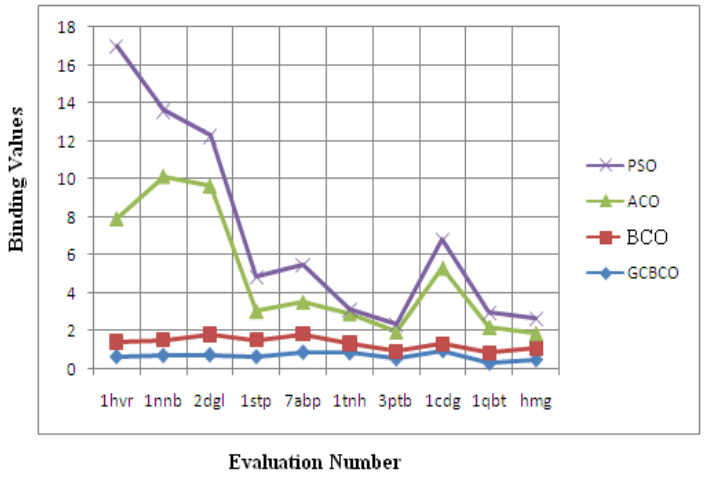

Figure 13. Docking accuracy of PSO, ACO, FCBCO and GCBCO algorithms

VIII. CONCLUSION

The performance of GCBCO is tested in 10 docking instances from the PDB bind core set and compared to the performance of three algorithms. The result shows that the GCBCO could find Ligand poses with best energy levels than the existing search algorithms implemented. The results of GCBCO algorithm are compared with PSO and ACO algorithms and the outcome shows that the GCBCO is successful on optimization of Genetic clustering. This work gets the lowest energy in all the 10 test cases. This GCBCO algorithm gives a better performance than other methods.

REFERENCES

[1] H. David, M. Heikki and S. Padhraic, “Principles of Data Mining,” J. MIT Press, vol. 1, pp. 567-650, 2001.

[2] L. Kantardzic and J. Mehmed, “Data Mining: Concepts, Models, Methods, and Algorithms”, John Wiley & Sons, vol. 3, pp. 356-400, 2003.

[3] R. Pradeep, and S. Singh, “A Survey of Clustering Techniques”, International Journal of Computer Applications, vol. 7, pp.156-162, 2010.

[4] R. David Musicant, M. Anna Ritz and G. Thomos, “Adapting K-medians to generate Normalized cluster centers,” Society for Industrial and Applied Mathematics, vol. 14, pp. 51-57, 2006.

[5] M. Morris, G. Goodsell, R. Halliday, R. Huey, "Automated docking using a Lamarckian genetic algorithm and an empirical binding free energy function," J. Computational Chemistry, vol. 19, no. 14, pp. 1639-1662, 1998.

[6] L.O. Hall. B. Ozvurt and J.C. Bezdek, “Clustering with a Genetically Optimized Approach,” Thesis (PhD) Universitˊe Blaise Pascal Clermon, 2002.

[7] D. Swagatam, A. Abraham and K. Amit, “Swarm Intelligence Algorithms in Bioinformatics,” Jadavpur University, 2007.

[8] D. Karaboga, “An Idea Based On Honey Bee Swarm for Numerical Optimization,” Technical Report-TR06, Erciyes University., 2005.

[9] A. Hadidi, S. Kazemzadeh, and S. Azad, “Structural optimization using Artificial Bee Colony Optimization,” 2 nd International Conference on Engineering optimization, 2010.

[10] D. Karaboga, and B. Basturk, "Artificical Bee Colony (ABC) optimization Algorithm for solving Constrained optimization Problems," LNCS: Advances in Soft Computing: Foundations of Fuzzy Logic and Soft Computing, vol. 10, pp.145-152, 2007.

[11] T. Lengauer, M. Rarey, "Computational methods for bio molecular docking", Curr Opin Struct Biol, vol .3, pp. 40-50, 1996.

[12] D.B. Kitchen, H. Decornez, J.R. Furr and Bajorath, J "Docking and scoring in virtual screening for drud discovery: methods and applications", Nature reviews Drug Discovery, vol. 11, pp.178-182, 2004.

[13] A. Jain, "Scoring Functions for Protein-Ligand Docking", Curr. Protein Pept. Sci, vol. 5, pp.200-215, 2006.

[14] T. Fukunishi, and N. Haruki, "Prediction of ligand binding sites of proteins by molecular docking calculation for a random ligand library", Journal of Protein science, vol. 9, pp.558-562, 2001.

[15] K. Oliver, S. Thomas, and E. Thomas, "An Ant Colony Optimization Approach to Flexible Protein-Ligand Docking," Journal of Bioinformatics., vol. 13, pp. 145-166, 2006.

[16] R.D. Taylor, P.J. Jewsbury and J.W. Essex, "A review of protein-small molecule docking methods," J. Comput. Aided Mol. Des, vol. 16, no. 3, pp. 151-166, 2002.

[17] E. Kiruba Nesamalar, C.P. Chandran, "Genetic Clustering with Bee Colony Optimization for Flexible Protein-Ligand Docking", Proceedings of the International Conference on Pattern Recognition, Informatics and Medical Engineering (PRIME-2012), pp. 82-87, ISBN 978-1-4673-1038-3, 2012.

[18] S. Guha, R. Rastogi, and K. Shim, "ROCK: Arobust clustering algorithm for categorical attributes," Information System, vol.25, pp.345-366, 2000.

[19] S. Mohammad Ali and R. Mohammad, "A Hybrid algorithm for data clustering using Honey Bee algorithm, Genetic algorithm and K-means method," Journal of advanced Computer science and Technology Research, vol.1, pp. 110-125, 2011.

[20] K. Dervis, and C. Ozturk, "Fuzzy Clustering with Bee Colony Optimization," Scientific Research, vol. 1, pp. 600-615, 2010.

[21] D. Konrad, H. Daniel, and Rene Meier, "Bonding as a swarm: Applying Bee Nest Site Selection behavior to Protein Docking Algorithm," Journal of GECCO, vol. 3, pp.142-152, 2011.

[22] E. Kiruba Nesamalar and C.P. Chandran, "Fuzzy Clustering with Ant Colony Optimization for Flexible Protein-Ligand Docking"., Proceedings of International Conference on Mathematics in Engineering and Business Management., pp. 381-386, IBSN 978-81-8286-015-5, 2012.

Dr.C.P.Chandran is an Associate Professor of Computer Science and Head of Post-Graduate Department of Bioinformatics, Co-ordinator, Center for Technology Enhanced Learning (CTEL), Ayya Nadar Janaki Ammal College, Sivakasi, India, (Madurai Kamaraj University). He received his Doctoral degree in Computer Science from Madurai Kamaraj University, India. He has about 16 years of teaching experience in Computer Science. He is an Alumni of Department of Physics, NGM College, Pollachi and Department of Computer Science, Bharathiar University, Coimbatore, India. He is an editor/reviewer of various international journals. His research focuses on data mining in bioinformatics, rough sets, swarm intelligence and granular computing. He has authored 10 international journal papers. He has published 30 research papers in national and international conference proceedings

E.Kiruba Nesamalar was born in Sivakasi, Tamil Nadu (TN), India, in 1988. He received the Bachelor of Computer Science (B.Sc.) degree from the Standard Fireworks Rajaratnam College for Women, Sivakasi, TN, India, in 2008 and the Master of Computer Science (M.Sc.) degree from the Ayya Nadar Janaki Ammal College (ANJAC), in 2010 and Master of Philosophy (M.Phil.) of Computer Science degree from the ANJAC, Sivakasi, TN, India, in 2012. Her research interests include data mining, and bioinformatics.