Journal of Network and Innovative Computing
ISSN 2160-2174 Volume 1 (2013) pp. 270-289

A Multi-Objective Evolutionary Approach to Evaluate the Designing Perspective of Protein-Protein Interaction Network

Archana Chowdhury[1], Amit Konar[1] , Pratyusha Rakshit[1], Atulya K. Nagar[2]

[1]Department of Electronics and Tele-Communication Engineering,
Jadavpur University, Kolkata, India
chowdhuryarchana@gmail.com, konaramit@yahoo.co.in, pratyushar1@gmail.com

[2]Department of Mathematics and Computer Science, Liverpool Hope University, Liverpool, UK
nagara@hope.ac.uk

Abstract— **Proteins interact with each other in a highly specific manner, and protein interactions play a key role in many cellular processes. Since protein interactions determine the outcome of most cellular processes, so identifying and characterizing Protein–Protein interactions and their networks are essential for understanding the mechanisms of biological processes on a molecular level. This paper explores the application of Non-dominated Sorting Bee Colony (NSBC) optimization algorithm to the Protein- Protein Interaction (PPI) identification problem. In this work, PPI is formulated as a multi-objective optimization problem. The proposed scheme determines an optimal solution based on the binding energy, mismatch in phylogenetic profiles of two bound proteins and clustering coefficients. Results are demonstrated for three different networks both numerically and pictorially. Experimental results reveal that the proposed method outperforms Differential Evolution for Multi-objective Optimization (DEMO), Multi-Objective Particle Swarm Optimization (MOPSO), Non-dominated Sorting Genetic Algorithm-II (NSGA-II), Artificial Bee Colony (ABC), and Differential Evolution (DE).**

Keywords-protein-protein interaction; phylogenetic profile; CHARMM energy; non dominated sorting bee colony optimization;clustering coefficient.

I. INTRODUCTION

Knowledge of protein-protein interactions provides crucial insights into their functions within a cell. Various high throughput experimental techniques such as mass spectrometry, yeast two hybrid, and tandem affinity purification have generated a significant amount of large-scale high throughput protein interaction data [1], [2], [3], [4], [5], [6], [7], [8]. Advances in experimental techniques are paralleled by the rapid development of computational approaches designed to detect protein–protein interactions [9], [10], [11], [12], [13], [14], [15], [16]. Computational inference of protein-protein interactions is an interesting and challenging area of research in modern biology. Computational methods infer potential interactions using one or more genomic features related to the protein pairs as predictor attributes. Recently hybrid machine learning approach has been used to predict Protein-Protein interaction network [17]. In [18], [19], [20] one can find a few recent reviews regarding experimental and computational methods for protein-protein interaction prediction. Protein-Protein interaction network can be used to predict the function of a protein using the subgraph approach [22], overlapping functional modules using clustering technique [23]. These approaches complement experimental techniques and, if proven to be successful in predicting interactions provide insights into principles governing protein interactions.

In the past there have been many approaches to infer interactions between proteins. Most of the in-silico methods for predicting interaction partners are based on simple sequence and genome features intuitively related with functional relations between the corresponding proteins. The general relation behind these approaches is that the functional or structural interactions between proteins have potentially modeled their sequences to better fulfill their potential functions in the corresponding organisms.

If the genes that encode two proteins are neighbors on the chromosome in several genomes, the corresponding proteins are likely to be functionally linked [24]. This method is particularly useful in case of prokaryotes, where operons commonly exist, or in organisms where operon-like clusters are observed. The obvious drawback of this approach is its limitation to the bacterial genomes as a source of information, where the tendency to put together functionally related proteins in operons is clear. Thus this methodology can be applied to eukaryotic proteins only if they have homologues in bacteria.

In [41], it has been shown that the members of some pairs of functionally related proteins tend to be fused in the same polypeptide in a number of organisms, known as 'Rosetta-Stone' protein. A model of gene neighboring method for PPI network problem has been proposed in [42]. The idea is based on the fact that if the genes that encode two proteins are

neighbors on the chromosome in several genomes, the corresponding proteins are functionally linked. Gene expression data has also been shown to be useful in understanding the dynamics of PPI networks [24]. Lu and collaborators [25] integrated gene expression profiles (from a mice model of asthma) into a network of mouse PPIs derived from the BIND database. They found that highly connected proteins, or hub proteins in the network have less variable gene expression profiles compared to proteins at the network periphery.

An important area under focus in many research projects is to infer protein interactions by looking at their domain compositions. Domains are evolutionarily conserved sequence units which are believed to be responsible for the interactions among the proteins to which they belong. There are many different methods which infer protein interactions using information on their domain composition. A protein pair is thought to be physically interacting if at least one of their constituent domain pair interacts. Most of the proteins in organisms like S. Cerevisiae are assigned one or more domains and information about the domains pairs in high confidence experimentally determined protein interaction data sets can be used to infer domain-domain and hence, protein-protein interaction. As there are no specific domain interaction data available, many methods have been developed for finding potential domain interaction from available experimentally determined high confidence protein-protein interaction datasets and then that information is used to predict back the novel protein-protein interactions as well [26], [27], [28], [29]. Majority of these models often have limitations in providing detailed information on which single domain pair actually interact for the predicted protein interaction, so computational model based on multidomain based approach were proposed [30].

There are also computational methods for the prediction of interaction partners that use structural information. Most of these methods are intended to predict whether the homologues of two proteins known to interact will interact too or not. Aloy et al. derived statistical potentials from known interactions and then used them to score the possible interactions between the homologues of the members of a given complex [31], [32]. In a similar way, the energetic feasibility of different complexes between members of the Ras family and different families of Ras effectors was evaluated using the FOLD-X [33].

The patterns of presence or absence of proteins across multiple genomes (phylogenetic or phyletic profiles) can be used to infer interactions between proteins [21]. Identification of functional linkages between proteins using phylogenetic profiles is based on the idea that functionally linked proteins would co-occur in genomes. The phylogenetic profile of a given protein family reflects the presence or absence of that family in a set of organisms and as such, it represents the species distribution of the protein family. The phylogenetic profile of a protein can be represented as a 'bit string', encoding the presence or absence of the protein in each of the genomes considered. Proteins having matching or similar phylogenetic profiles tend to be strongly functionally linked [34].Phylogenetic profiles created using DNA sequence and RNA can be used project evolutionary scenario [35], [36].

Ramani and Marcotte [37] established a mapping between the leaves of the two similar phylogenetic trees resulting in a one-to-one mapping between the members of one family and those of others. The similarity of the phylogenetic trees of interacting protein families can possibly be explained by the similar evolutionary pressure exerted on interacting and functionally related proteins, given that they are involved in the same cellular process, and by the fact that they are forced to co-adapt to each other. Both these factors would result in a coordinated evolutionary history or co-evolution, which in turn is reflected in the similarity of the corresponding trees. Correlation of phylogenetic trees can be used to predict specific interaction partners between members of two families.

Jothi proposed a new algorithm called MORPH [38] to detect interacting pairs based on the coevolution hypothesis using topological information, entropy and information content of the evolutionary trees. An important drawback of phylogenetic tree and related approaches is that they can only be applied to pairs of proteins with orthologues in many common species. Only the leaves of the trees corresponding to species where both proteins are present can be used.

Recent research aimed at solving the protein-protein interaction (PPI) problem considers designing algorithms which can balance the efficient search of possible binding mode conformations of the interacting proteins with reasonable computational complexity. A lot of effort is being spent on reducing the protein-protein interaction problem to a single-objective optimization problem by aggregating of all the necessary objectives affecting the formation of protein protein interaction network. A single objective optimization algorithm using Artificial Bee Colony (ABC) was proposed in [44] to solve PPI network problem. Different solutions, i.e. the PPI network created, may involve a tradeoff among different objectives. An optimum solution with respect to one objective may not be optimum with respect to another objective. Hence, one cannot choose a solution which is optimal with respect to only one objective. In general, in problems with more than one conflicting objective, there is no single optimum solution. There exists, instead, a set of solutions which are all optimal, called the optimal Pareto front. Among the most popular algorithms used in Pareto-based approaches are evolutionary algorithms (EAs) [39]. EAs are a class of stochastic, heuristic-based approaches to objective optimization that are designed with biological evolutionary principles in mind and are especially suitable for exploring large search spaces [40]. Typically, such algorithms are based on populations of individuals that are evolved through a set of genetic operators such as reproduction, mutation, crossover (an analog of biological recombination) and selection of the fittest for further evolution. In the case of single objectives, selection of solutions involves ranking the individual solutions according to their fitness and choosing a subset. Multi-objective EAs (MOEAs) are an extension of traditional EAs that can address multiple objectives simultaneously. MOEAs exploit the

availability of a population of individual solutions to map the entire Pareto front in a single run. Selection of the solutions involves fitness assessment of each individual solution to all objectives and Pareto ranking. MOEAs employ several techniques to achieve faster convergence to the Pareto-front and often employ niching to identify solutions representative of the Pareto-optimal set.

In this paper a multi-objective optimization (MOO) approach that simultaneously minimizes three different objectives to generate Pareto solutions corresponding to the optimal protein-protein interaction network is presented. Here, we have studied the scope of the well-known optimization algorithm namely Non-dominated Sorting Bee Colony (NSBC) optimization algorithm [45] to judiciously determine the PPI network structure. The work proposed in this paper considers the formulation of protein protein interaction problem as a multi-objective optimization problem, concerning minimization of three objective functions. The first part is based on identification of functional linkage between proteins using phylogenetic profiles. The second part of the fitness function is an energy function which determines the stable connectivity between the interacting proteins using CHARMM energy function [43]. The third part of the fitness function is the clustering coefficient which is a popular metric from graph theory. It is observed that PPI networks are characterized by a high average clustering coefficient, indicating the large fraction of interaction partners of a protein that interact among themselves. Thus the third part of the fitness function specifies the reciprocal of the 'clustering coefficient' of the PPI network.

The rest of the paper is organized as follows. Section II provides the formulation of PPI network design problem. Section III depicts the principles used to predict the PPI network structures. Section IV and V give an overview of Artificial Bee Colony and Non Dominated Sorted Bee Colony optimization algorithm respectively. Section VI is used to describe the application of NSBC algorithm to find an optimized PPI network. The pseudo-code for solving the given constrained optimization function is also provided in the section. Experiment results for three known PPI network is provided in section VII. Section VIII concludes the paper.

II. FORMULATION OF THE PROBLEM

In this section the problem of PPI network identification is presented in a framework of multi-objective optimization. To accomplish this we have constructed three objective functions and the minimization of each yields a possible solution to the PPI identification problem.

A. Predicting protein interactions using phylogenetic analysis

Phylogenetic profiles describe patterns of presence or absence of proteins in a collection of organisms. The construction of phylogenetic profiles begins with a collection of k completely sequenced genomes **G** from different organisms and a collection of *l* proteins **P** of interests. For each protein p_i, a phylogenetic profile is represented as a k-length binary string s = $s_1 s_2 \cdots s_k$ where $s_j = 1$ if protein p_i is present in genome g_j and $s_j = 0$ if protein pi is absent in genome g_j.

Table-I shows an example of the construction of phylogenetic profiles. In this example, phylogenetic profiles are constructed for 8 proteins (p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8) indicating their presence (or absence) in 5 genomes (g_1, g_2, g_3, g_4, g_5) from different organisms. Note that all proteins $p_1 \ldots p_8$ are present in genome g_1.

Functional coupling of proteins is then inferred by clustering proteins according to the intrinsic similarities of the underlying phylogenetic profile patterns. It is often concluded that proteins associated to the same cluster are functionally related. For the example shown in Table-I, a functional association between proteins p_2 and p_7 would be inferred by this method as they possess identical phylogenetic profiles.

The logic underlying this reasoning is that proteins with similar phylogenetic profiles are likely to interact in performing some biological process. In effect, there should be evolutionary pressure acting on a group of proteins in order to preserve a function that confers an advantage to the organisms.

TABLE-I
Example of phylogenetic sequence of proteins

Protein	g_1	g_2	g_3	g_4	g_5
p_1	1	1	0	1	1
p_2	1	1	1	0	1
p_3	1	0	1	1	1
p_4	1	1	0	0	0
p_5	1	1	1	1	1
p_6	1	0	1	1	1
p_7	1	1	1	0	1
p_8	1	0	0	1	1

To meet this issue, we evaluate the accuracy of the produced PPI network by comparing the phylogenetic profiles of two bonded proteins in the network with the hope that if the two proteins interact with each other in reality then the difference (error) between these two phylogenetic profiles will be less. That error for a PPI network has been calculated using the equation (1).

$$C1 = \frac{1}{K} \sum_{i=1}^{N} \sum_{\forall j \in Set_i} \sum_{k=1}^{K} (s_{k,i} - s_{k,j})^2 \qquad (1)$$

Here, N is the total number of proteins in the network and K is the length of phylogenetic profile of proteins p_i and p_j. $s_{k,i}$ and $s_{k,j}$ represent the presence or absence of proteins p_i and p_j in genome g_k respectively. Set_i symbolizes the set of proteins interacting with protein p_i.

B. CHARMM (Chemistry at HARvard Macromolecular Mechanism) force fields

In PPI identification problem, the second objective is to minimize the energy. Hence, in order to perform a qualitative analysis of the conformation of PPI network in large space, there is a need of some cost or energy functions, commonly known as force fields. In this work the CHARMM force fields

are considered to evaluate the cost of the conformations which is commonly known as Chemistry at HARvard Macromolecular Mechanism. CHARMM models the dynamics and mechanism of macromolecular system using empirical and mixed empirical quantum mechanical force fields. CHARMM uses potential functions that approximate the total potential as a sum of bond stretching, bond bending, bond twisting, improper potentials which are used to maintain planar bonds, plus potentials representing the nonbonded van der Waals and electrostatic interactions.

The energy of the bond stretching is approximated as

$$V_{bond} = K_b(b-b_0)^2 \tag{2}$$

where K_b is a constant that depends on the identity of the two atoms sharing the bond in a protein, b is the length of the bond and b_0 is the unstrained bond length in equilibrium.

The energy of the bond bending is approximated as

$$V_{angle} = K_\theta(\theta-\theta_0)^2 \tag{3}$$

where K_θ is a constant that depends on the three atoms defining the angle θ within a protein, θ is the angle between the atoms and θ_0 is the unstrained angle in equilibrium.

Determination of the energy of bond twisting (dihedral energy) requires four atoms of a protein to define the bond and the amount it is twisted. It is approximated as

$$V_{dihedral} = K_\chi(1+\cos(n\chi-\delta)) \tag{4}$$

where Kχ and δ are constants that depend on the adjacent atoms, n does an integer that depends on the number of bonds made by atoms, and χ is the value of the dihedral angle.

Improper forces or potentials are artificial forces or potentials that are used to hold a group consisting of one central atom that is bonded to three others in a particular configuration. The potential that is used in CHARMM for improper dihedrals is

$$V_{improper} = K_\psi(\psi-\psi_0)^2 \tag{5}$$

where Kψ is a constant, ψ_0 is the equilibrium improper angle, and ψ is the improper angle that depends on the coordinates of the atoms.

More elaborate force field may include the Urey-Bradley term given as

$$V_{Urey-Bradley} = K_{UB}(S-S_0)^2 \tag{6}$$

where K_{UB} is the Urey-Bradley force constant, S is the distance between two atoms separated by two covalent bonds (1, 3 distance) and S_0 is the equilibrium distance.

Therefore, the bonded energy is given by

$$V_{bond} = \sum_{bond} K_b(b-b_0)^2 + \sum_{angle} K_\theta(\theta-\theta_0)^2 + \sum_{dihedral} K_\chi(1+\cos(n\chi-\delta)) + \sum_{improper} K_\psi(\psi-\psi_0)^2 + \sum_{UB} K_{UB}(S-S_0)^2 \tag{7}$$

Van der Waals interactions between two atoms within the active site of two proteins are approximated with a Lennard-Jones potential as

$$V_{Lennard-Jones} = \varepsilon_{i,j}\left[\left(\frac{R_{min,i,j}}{r}\right)^{12} - 2\left(\frac{R_{min,i,j}}{r}\right)^{6}\right] \tag{8}$$

where $\varepsilon_{i,j}$ is the Lennard-Jones well depth, r is the distance between atoms i and j, $R_{min,i,j}$ is the minimum interaction radius.

The electrostatic interaction between two atoms is

$$V_{electrostatic} = \frac{q_i q_j}{4\pi\varepsilon r} \tag{9}$$

where q_i and q_j are the charges of the two atoms, r is the separation, and ε is the dielectric constant of the surrounding medium.

Hence, the nonbonded energy is given as

$$V_{non-bond} = \sum_{non-bond}\left[\varepsilon_{i,j}\left[\left(\frac{R_{min,i,j}}{r}\right)^{12} - 2\left(\frac{R_{min,i,j}}{r}\right)^{6}\right] + \frac{q_i q_j}{4\pi\varepsilon r}\right] \tag{10}$$

So the basic functional form of CHARMM force field to perform qualitative analysis of conformations of a PPI network is represented as follows.

$$\begin{aligned} C_2 &= \sum_{i=1}^{N} \sum_{\forall j \in Set_i} V_{CHARMM,i,j} \\ &= \sum_{i=1}^{N} \sum_{\forall j \in Set_i} \left(V_{bond,i} + V_{bond,j} + V_{non-bond,i,j}\right) \end{aligned} \tag{11}$$

Here $V_{bond,i}$ is the bonded or intra-molecular energy of protein p_i. $V_{non\text{-}bond,i,j}$ represents the non-bonded or the inter-molecular energy between proteins p_i and p_j. A few constraints for PPI network identification are incorporated in the objective function for the proposed optimization problem. Non-bonding interaction energies are calculated for the protein constituting the PPI having residues with distance not more than 5Å (range for the force to be applicable) and not less than 0.65Å (to avoid steric hindrance), from the interacting atoms of other interacting protein molecule. A few chemical preferences are considered. For example, a polar hydroxyl group should be oriented in a

way so that it lies close to positively charged groups on the protein active site.

C. *Clustering Coefficients*

Clustering coefficient is a property of a node in a network. It tells how well the neighborhood of the node is connected among them and quantifies how close its neighbors are to being a clique (complete graph). If the neighborhood of a node is fully connected, the clustering coefficient is 1 and a value close to 0 means that there are hardly any connections in the neighborhood of the node under question. Clustering coefficient of a node is the ratio of number of connections in the neighborhood of a node and the number of connections if the neighborhood was fully connected. Here neighborhood of node A means the nodes that are connected to A but does not include A itself. A fully connected group of N nodes has N*(N-1)/2 connections.

The degree k_v of a node v is defined as the cardinality of the node v with its first order neighborhood set i.e., the number of arcs connected to node v. The clustering coefficient of a vertex v with degree k_v can be defined as

$$CC(v) = \frac{2\,n_v}{k_v(k_v - 1)} \tag{12}$$

where n_v denotes the number of triangles that go through node v.

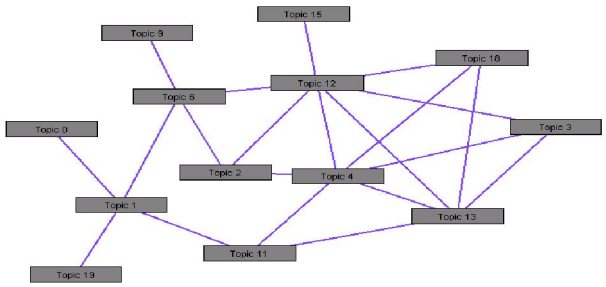


Figure 1. Example of calculation of clustering coefficient of a network

For example, the neighborhood of topic 6 consists of topics 9, 12, 2 and 1. Between these topics there is only one connection, from topic 2 to topic 12. If the four topics were fully connected, that is there would be a connection from each topic to every other topic, there would be 4*3/2=6 connections. Clustering coefficient of topic 6 is therefore 1/6=0.17. Clustering coefficient of topic 1 is 0 because there is no connection at all between topics 0, 6, 11 and 19. Clustering coefficient of topic 3 is 1 because the neighborhood consisting of topics 12, 4 and 13 is fully connected.

For a PPI network with N proteins of interest the average clustering coefficient of the network is given as follows.

$$CC = \frac{1}{N}\sum_{i=1}^{N} CC(p_i) \tag{13}$$

Hence, minimization of equation (14) would yield a dense PPI network with 'epsilon' as a very small positive integer.

$$C_3 = \frac{1}{CC + \text{epsilon}} \tag{14}$$

As mentioned before, a low objective function value corresponds to a better solution with greater stability. Accordingly, the corresponding fitness value, fit_k, k= [1, 2, 3], of a conformation is defined in this work as

$$fit_k = \frac{1}{C_k}, k = [1,2,3] \tag{15}$$

III. Formation of a Protein-Protein Interaction Network

In the proposed method for N proteins of interest, each with K dimensional phylogenetic sequence, a solution is represented by a two dimensional binary matrix **X**= $[x_{j,k}]$, $\forall$ j, k $\in$ [1,N] of dimension N × N. It describes the presence or absence of an interaction between two proteins. Hence

$$x_{j,k} = \begin{cases} 1 & \text{if proteins } p_j \text{ and } p_k \text{ interact with each other} \\ 0 & \text{if there is no interaction between } p_j \text{ and } p_k \end{cases} \tag{16}$$

For example, the solution matrix **X** for the PPI network with N=4 as shown in Fig. 2 can be represented by equation (17).

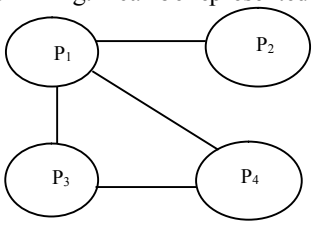


Figure 2. Example of a PPI network with 4 proteins

$$\mathbf{X} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \tag{17}$$

This implies p_1 is connected to p_2, p_3 and p_4. There exists another interaction between p_3 and p_4.

IV. An Overview of Artificial Bee Colony (ABC) Optimization Algorithm

The Artificial Bee Colony (ABC) [50], [51], [52] optimization is a population based algorithm for numerical function optimization. It draws inspiration from the stochastic behavior of foraging in bees. In case of real honey bees the self-organization dynamic is based on four properties: positive feedback (exploration through waggle dance), negative feedback (prevention of exploitation of poor food sources), fluctuations (scouting for new food source) and multiple interactions [46]. Recent studies have suggested that ABC is better suited in terms of convergence speed to solve optimization problems than other EAs like Differential Evolution (DE) [47], [48] and Particle Swarm Optimization technique (PSO) [49].

In ABC algorithm, the colony of artificial bees contains three groups of bees:

- Onlooker- waiting on a dance area to choose a food source
- Employed- going to the food source visited by it previously
- Scout- carrying out random search of food sources

In ABC algorithm, the position of a food source represents a possible solution of the optimization problem and the nectar amount of a food source corresponds to the fitness of the associated solution. The number of employed bees and onlooker bees is equal to the number of solutions in the population. ABC consists of following steps:

A. Initialization

ABC generates a randomly distributed initial population P (t=0) of NP solutions (food source positions) where NP denotes the size of population. Each solution $\vec{X}_i$ (i=1, 2… NP) is a D dimensional vector.

B. Placement of employed bees on the food sources

An employed bee produces a modification on the position in her memory depending on the local information (visual information) as stated by equation (19) and tests the nectar amount of the new source. Provided that the nectar amount of the new one is higher than that of the previous one, the bee memorizes the new position and forgets the old one. Otherwise she keeps the position of the previous one in her memory.

C. Placement of onlooker bees on the food sources

An onlooker bee evaluates the nectar information from all employed bees and chooses a food source depending on the probability value associated with that food source, p_i, given as

$$p_i = \frac{fit_i}{\sum_{j=0}^{j=N_p-1} fit_j} \quad (18)$$

where fit_i is the fitness value of the solution i evaluated by its employed bee. After that, as in case of employed bee, onlooker bee produces a modification on the position in her memory and memorizes the position of better food source only.

In order to find a solution $\vec{X}'_i$ in the neighborhood of $\vec{X}_i$, a solution parameter j and another solution $\vec{X}_k$ are selected on random basis. Except for the value of chosen parameter j, all other parameter values of $\vec{X}'_i$ are same as in the solution $\vec{X}_i$, for example, $\vec{X}'_i = (x_{i1}\ldots\ x_{i(j-1)}, x_{ij}', x_{i(j+1)}, \ldots, x_{i(D-1)}, x_{iD})$. The value of x_{ij}' parameter in X_i' solution is computed as follows:

$$x_{ij}' = x_{ij} + u(x_{ij} - x_{kj}) \quad (19)$$

where u is a uniform variable in [-1, 1] and k is any number between 1 to NP but not equal to i.

D. Placement of scout bee on the abandoned food sources

In the ABC algorithm, if a position cannot be improved further through a predefined number of cycles called 'limit', the food source is abandoned. This abandoned food source is replaced by the scouts by randomly producing a position.

After that again steps (B), (C) and (D) will be repeated until the stopping criteria is met.

V. Non-dominated Sorting Bee Colony (NSBC) Optimization Algorithm

Evolutionary algorithms are used to determine the best solution in a single objective optimization problem. In case of multi-objective optimization, it is hard to obtain a unique solution, capable of satisfying the objective functions jointly. The primary motivation of evolutionary multi-objective optimization (EMOO) algorithms is to obtain Pareto optimal solutions in a single run. One of the most popular members of the EMOO family is Non-dominated Sorting Bee Colony (NSBC) optimization algorithm. The following definitions will be referred to frequently to explain NSBC and its extension.

Definition 1. Let $\vec{X}_i$ and $\vec{X}_j$ be two food sources of a multi-objective optimization (MOO) problem. $\vec{X}_i$ is said to **dominate** the other food source $\vec{X}_j$, denoted by $\vec{X}_i \prec \vec{X}_j$, if both conditions 1 and 2 are true.

1. The food source $\vec{X}_i$ is no worse than $\vec{X}_j$ in all objective functions.

2. The food source $\vec{X}_i$ is strictly better than $\vec{X}_j$ in at least one objective.

Definition 2. Let *P* be a set of solution (food sources) to a MOO problem, and $P' \subseteq P$, such that the members of P' are not dominated by any member of P .Then P' is called the **non-dominated set** of solutions.

Definition 3. Crowding distance of a member of non-dominated set attempts to approximate the perimeter of a hypercube formed by considering the nearest neighbors of that member at the vertices of the objective space. For example, let f_1 and f_2 be two objective functions, and $\vec{X}_i$ and $\vec{X}_j$ are the nearest neighbor of $\vec{X}$, where $\vec{X}$, $\vec{X}_i$ and $\vec{X}_j$ are the members of the non-dominated list of solutions, then the crowding distance of $\vec{X}$ is computed by $\left|f_1(\vec{X}_i)-f_1(\vec{X}_j)\right|+\left|f_2(\vec{X}_i)-f_2(\vec{X}_j)\right|$.

NSBC [45] is an evolutionary strategy that utilizes the advantages of ABC with the mechanisms of Pareto-based ranking and crowding distance sorting. NSBC shares its three main steps with the classical ABC algorithm, namely employed bee, onlooker bee and scout bee. The selection step of employed and onlooker bees in NSBC, however, is different from classical ABC and will be briefly outlined below.

In NSBC, neighborhood food sources are initially generated from each of the target food sources (of the current population of size NP) using (19). Now the neighborhood food source replaces the corresponding target food source if the neighborhood food source dominates the target one. Otherwise, if the target food source dominates the neighborhood food source, it is discarded. However, when both the neighborhood and the target food sources are non-dominated with regard to each other, the neighborhood food source is added to the current population. This enables a faster convergence to true Pareto front. This step is repeated for all the food sources and hence, a population of solution food sources is obtained with size in between NP and 2NP. Now the extended population is truncated to keep only the best NP individuals. This is performed by using the concept of non-dominated sorting and evaluating each solution in the same Pareto front using crowding distance metric. This mechanism stimulates the uniform spread of solutions.

VI. Solving the Constraint Optimization Problem Using NSBC

Non-dominated Sorting Bee Colony (NSBC) [45] has earned wide publicity for its simple structure with few lines of codes, fewer parameters and its excellent performance in numerical optimization with respect to speed and accuracy. In this section we propose a solution to the PPI network identification problem using NSBC. A potential network is encoded by a food source in NSBC. In every step of the optimization algorithm, bond length, bond angles and dihedral angles of interacting proteins are calculated for fitness evaluation. In NSBC algorithm, the nectar amount of a food source corresponds to the fitness of the associated solution. The number of employed bees and onlooker bees is equal to the number of solutions in the population. An algorithm outlining the scheme is discussed below:

The main process consists of six phases, namely, initialization, fitness evaluation, employed bee, probability calculation, onlooker bee and scout bee phase. These phases are described below in details:

A. Initialization

NSBC starts with a population of NP $N \times N$-dimensional solution matrices (food sources) representing the candidate solutions of the PPI network for N proteins. We shall denote subsequent generations in NSBC by $t = 0,1,..., t_{max}$. Since the food-sources are likely to be changed over different generations, we may adopt the following notation for representing the i-th food-source (matrix) of the population at the current generation as

$$P_t = \{\mathbf{X}_1(t), \mathbf{X}_2(t),..., \mathbf{X}_{NP}(t)\}$$

Here each $\mathbf{X_i(t)}$ for $i = [1,2,..., NP]$ is a $N \times N$ -dimensional symmetric matrix where

$$X_{i,j,k} = X_{i,k,j} = \begin{cases} 1 & \text{if proteins } p_j \text{ and } p_k \text{ interact with each other} \\ 0 & \text{if there is no interaction between } p_j \text{ and } p_k \end{cases},$$

for $j, k = [1,2,..., N]$

Hence, we may initialize $X_{i,j,k}$ at generation t=0 as

$$X_{i,j,k} = X_{i,k,j} = \begin{cases} 1 \text{ if } rand_{j,k}(0,1) > 0.5 \\ 0 \text{ otherwise} \end{cases}$$

where $rand_{j,k}(0,1)$ is a uniformly distributed random number lying between 0 and 1 and is instantiated independently for each component of the i-th food-source.

B. Initialization of Mixed Population

The entries for the mixed population R_t in the t-th generation are initialized with the target vectors in the current population P_t.

C. Fitness Evaluation

Each food source $\mathbf{X_i(t)}$ is assigned three fitness values (nectar amount) $fit_k(\mathbf{X_i(t)})$ computed using (11), (12), (14) and (15) with k= [1, 3] as described in section IV.

D. Employed Bee Phase

An employed bee produces a modification $\mathbf{X'_i(t)}$ on the position (solution) in her memory $\mathbf{X_i(t)}$ depending on the local information (visual information) as stated by equation (19) and tests $fit(\mathbf{X'_i(t)})$,the nectar amount (fitness value) of the new source (new solution).

Neighborhood food source $\mathbf{X'_i(t)}$ has being generated by altering the value of one randomly chosen solution parameter of $\mathbf{X_i(t)}$ and keeping other parameters of $\mathbf{X_i(t)}$ unchanged.

Let us denote solution $\mathbf{X_i(t)}$ as

$$\mathbf{X_i(t)} = \begin{bmatrix} x_{i,1,1}(t) & x_{i,1,2}(t) & ..... & x_{i,1,N}(t) \\ x_{i,2,1}(t) & x_{i,2,2}(t) & ..... & x_{i,1,N}(t) \\ . & . & . & . \\ . & . & x_{i,j,k}(t) & . \\ . & . & . & . \\ x_{i,N,1}(t) & x_{i,N,2}(t) & ..... & x_{i,N,N}(t) \end{bmatrix}$$

In order to find a solution $\mathbf{X'_i(t)}$ in the neighborhood of $\mathbf{X_i(t)}$, a solution parameter $x_{i,j,k}(t)$ and another solution $\mathbf{X_m(t)}$ are selected on random basis. Except for the value of chosen parameter $x_{i,j,k}(t)$, all other parameter values of $\mathbf{X'_i(t)}$ are same as in the solution $\mathbf{X_i(t)}$, for example,

$$\mathbf{X'_i(t)} = \begin{bmatrix} x_{i,1,1}(t) & x_{i,1,2}(t) & ..... & x_{i,1,N}(t) \\ x_{i,2,1}(t) & x_{i,2,2}(t) & ..... & x_{i,1,N}(t) \\ . & . & . & . \\ . & . & x'_{i,j,k}(t) & . \\ . & . & . & . \\ x_{i,N,1}(t) & x_{i,N,2}(t) & ..... & x_{i,N,N}(t) \end{bmatrix}$$

The value of $x'_{i,j,k}(t)$ parameter in $\mathbf{X'_i(t)}$ solution is computed using the following expression:

$$x'_{i,j,k}(t) = \begin{cases} 1 & \text{if } x_{i,j,k}(t) + \text{rand}_{j,k}(-1,1) \times (x_{i,j,k}(t) - x_{i,j,m}(t)) > 0.5 \\ 0 & \text{otherwise} \end{cases} \quad (20)$$

where $\text{rand}_{j,k}(-1,1)$ is a uniform variable in [-1, 1] and m is any number between 1 to NP but not equal to i. If a parameter produced by this operation exceeds its predetermined limit [0, 1], the parameter can be set to an acceptable value.

E. Selection byEmployed Bees

For all objectives to be minimized by the multi-objective optimization problem, the selection operator is defined as

$$\begin{aligned} &\text{If } \mathbf{X'_i(t)} \prec \mathbf{X_i(t)} \text{ then do} \\ &\text{Begin} \\ &\quad \mathbf{X_i(t+1)} \leftarrow \mathbf{X'_i(t)}; \\ &\quad \text{fit}_k(\mathbf{X_i(t+1)}) \leftarrow \text{fit}_k(\mathbf{X'_i(t)}) \text{ for } k = [1, \text{nobj}] \\ &\text{Else If } \mathbf{X_i(t)} \text{ and } \mathbf{X'_i(t)} \text{ are non - dominated then} \\ &\quad \text{Set } R_t \leftarrow R_t \cup \mathbf{X'_i(t)}. \\ &\text{End If.} \end{aligned} \quad (21)$$

Equation (21) indicates that if the new food source $\mathbf{X'_i(t)}$ is dominates the previous one $\mathbf{X_i(t)}$, the bee memorizes the new position $\mathbf{X'_i(t)}$ and forgets the old one i.e., $\mathbf{X_i(t)}$. However if both the parent and the child are non-dominated, she keeps the position of the both solutions in her memory.

F. Non-dominated Sorting

The obtained mixed population R_t of size N (NP<N<2NP) is sorted into a number of Pareto fronts according to non-domination. All the non-dominated solutions of the current population are ranked one (named *Pareto front* PF_1). A set S corresponding to the current front is maintained which contains the solutions of P_t being dominated by the non-dominated solutions of the present front. The second front is constructed by identifying the non-dominated solutions of S. The ranking process continues until all the non-dominated sets are identified and ranked as $PF_1, PF_2, PF_3 \ldots$

G. Truncation of the extended Population using Crowding Distance-Based Ranking

The parent population for the next iteration denoted by P_{t+1} is formed by choosing the non-dominated sets of solutions according to the ascending order of their ranking. Let PF_l be the set beyond which no other set can be accommodated in P_{t+1}(i.e., by adding PF_l its size exceeds NP). If such is the case, then the solutions in PF_l are sorted in descending order of crowding distance. The solutions with the highest crowding distances are included in P_{t+1} until its size becomes NP i.e., $P_{t+1} = PF_1(t) \cup PF_2(t) \cup \ldots \cup PF_{l-1}(t) \cup \textit{part of } PF_l(t)$.

H. Probability Calculation

Calculate the probability of each food source $\mathbf{X_i(t)}$ to be selected by the onlooker by given by

$$\text{prob(i)} = \frac{|\text{Set}_i|}{\text{NP}} \quad (22)$$

where Set_i is the set of all solutions that are dominated by solution $\mathbf{X_i(t)}$ based on the fitness value as evaluated by its employed bee NP is the number of food sources which is equal to the number of employed bees.

I. Onlooker Bee Phase

After all employed bees complete the search process; they share the nectar information of the food sources (solutions) and their position information with the onlooker bees on the dance area. An onlooker bee evaluates the nectar information from all employed bees and chooses a food source depending on the probability value associated with that food source, prob(i), as calculated by the expression (22). After that, as in case of employed bee, onlooker bee produces a modification on the position (solution) in her memory as described in (19) and checks the nectar amount of the candidate source (solution).

The merged population R_t of size between NP and 2NP is formed by following the principle as stated in (19). As in case of employed bees, here also, using the methodology of non-dominated sorting and crowding distance, the non-dominated food sources are found out from merged population to form the resulting population P_{t+1} of size NP.

J. Scout Bee Phase

In the ABC algorithm, if a position cannot be improved further through a predefined number of cycles called 'limit', the food source is abandoned. This abandoned food source is replaced by the scouts by randomly producing a position.

After each evolution, we repeat from step B until one of the following conditions for convergence is satisfied. The conditions include restricting the number of iterations, maintaining error limits, or the both, whichever occurs earlier.

Pseudo Code:

Input: K dimensional phylogenetic profiles for each of N proteins of interest (PF_j, $\forall j \in [1, N]$) given by the matrix **PF**= $[PF_j]$ of dimension $N \times K$.
Output: Desired PPI network P of dimension $N \times N$.
Begin
Call NSBC (PF);
End

Procedure NSBC (PF)
Begin
I. Set the generation number $t = 0$ and randomly initialize a population of NP individuals (food sources) $P_t = \{\mathbf{X}_1(t), \mathbf{X}_2(t), ..., \mathbf{X}_{NP}(t)\}$ for $i = [1,2,..., NP]$ each of dimension $N \times N$ as in Fig. 3. Here $X_{i,j,k}(t) \in [0,1]$ for $j,k = [1,2,..., N]$ as described in section 8.1.
II. Evaluate $fit_k(\mathbf{X_i(t)})$, for food source $\mathbf{X_i(t)}$, $i = [1,2,..., NP]$ using equation (15) after decoding each $\mathbf{X_i(t)}$. Set trial(i) $\leftarrow 0$. Here t denotes t-th iteration.
While stopping cri**ter**ion is not reached, **do**
Begin
III. Employed Bee Phase:
For i=1 to NP **do**
Begin
Produce a new food source $\mathbf{X'_i(t)}$ as shown in (19).
Evaluate fit($\mathbf{X'_i(t)}$)) using (15) after decoding each $\mathbf{X'_i(t)}$.
If $\mathbf{X'_i(t)} \prec \mathbf{X_i(t)}$ then do
Begin
$\mathbf{X_i(t+1)} \leftarrow \mathbf{X'_i(t)}$;
$fit_k(\mathbf{X_i(t+1)}) \leftarrow fit_k(\mathbf{X'_i(t)})$ for $k = [1, nobj]$;
$trial_i = 0$;
Else If $\mathbf{X_i(t)}$ and $\mathbf{X'_i(t)}$ are non - dominated then
Set $R_t \leftarrow R_t \cup \mathbf{X'_i(t)}$
Else $trial_i = trial_i + 1$.
End If
End If.
End.
IV. Call **non-dominated-sort** (R_t; Front – Set).
V. Set $P_{t+1} \leftarrow$ NULL and $i \leftarrow 1$.
Repeat
$P_{t+1} \leftarrow P_{t+1} \cup Front\_Set(i)$;
$i \leftarrow i + 1$;
Until $|P_{t+1}| + |\text{Front\_Set}(i)| > NP$.
Set $P_{t+1} \leftarrow P_t \cup (N - |P_{t+1}|)$ elements of Front _ Set(i) which is already sorted in descending order of crowding distance.
VI. Calculate Probability of Selection:
For i=1 to NP **do**
Begin

$$prob(i) = \frac{|Set_i|}{NP}$$

End.
VII. Onlooker Bee Phase:
Set $i \leftarrow 1$ and $k \leftarrow 1$.
While $k \leq$ NP **do**
Begin
If rand(0,1) $\leq$ prob(i)
Then do
Begin
$k \leftarrow k + 1$.
Produce a new food source $\mathbf{X'_i(t)}$ as shown in (19).
Evaluate fit($\mathbf{X'_i(t)}$) using (15) after decoding each $\mathbf{X'_i(t)}$.
If $\mathbf{X'_i(t)} \prec \mathbf{X_i(t)}$ then do
Begin
$\mathbf{X_i(t+1)} \leftarrow \mathbf{X'_i(t)}$;
$fit_k(\mathbf{X_i(t+1)}) \leftarrow fit_k(\mathbf{X'_i(t)})$ for $k = [1, nobj]$;
$trial_i = 0$;
Else If $\mathbf{X_i(t)}$ and $\mathbf{X'_i(t)}$ are non - dominated then
Set $R_t \leftarrow R_t \cup \mathbf{X'_i(t)}$
Else $trial_i = trial_i + 1$.
End If
End If.

$i \leftarrow i+1$.
End.
End.

VIII. Call **non-dominated-sort** (R_t; Front − Set).

IX. Set $P_{t+1} \leftarrow$ NULL and $i \leftarrow 1$.

Repeat

$P_{t+1} \leftarrow P_{t+1} \cup$ Front _ Set(i);

$i \leftarrow i+1$;

Until $|P_{t+1}| + |\text{Front\_Set}(i)| > NP$.

Set $P_{t+1} \leftarrow P_t \cup (N - |P_{t+1}|)$ elements of Front _ Set(i) which is already sorted in descending order of crowding distance.

X. Scout Bee Phase:

$$\text{index} \leftarrow \arg\left(\max_{i=1,2,\ldots,NP}(\text{trial}(i))\right)$$

If trial(index) > limit

Then

Reinitialize the food source $\mathbf{X_{index}(t)}$;

Set trial(i) ← 0.

End.

XI. Update PPI network: $\mathbf{P} \leftarrow \mathbf{X_{best}(t)}$. $\mathbf{X_{best}(t)}$ is the solution with minimum value of $\prod_{k=1}^{3} \text{fit}_k(\mathbf{X_i(t)})$ for i= [1, NP].

Return P.

Procedure non-dominated-sort (R_t; Front _ Set)

Begin

I. Set Front _ Set ← NULL.

II. **For** each $\mathbf{X_i(t)} \in R_t$ **do begin**

Set $\text{Set}_i \leftarrow$ NULL and $n_i \leftarrow 0$.

For each $\mathbf{X}_j(\mathbf{t}) \in R_t$ **do begin**

If $\mathbf{X_i(t)} \prec \mathbf{X_j(t)}$ **then** $\text{Set}_i \leftarrow \text{Set}_i \cup \{\mathbf{X_j(t)}\}$;

Else $n_i \leftarrow n_i + 1$;

End If.

End For.

If $n_i == 0$ **then** Front _ Set(1) ← Front _ Set(1) $\cup \{\mathbf{X_i(t)}\}$;

End If.

End For.

III. Set $i \leftarrow 1$.

IV. **While** Front _ Set(i) ≠ NULL **do begin**

Set H ← NULL.

For each $\mathbf{X_j(t)} \in$ Front _ Set(i) **do begin**

For each $\mathbf{X_k(t)} \in \text{Set}_j$ **do begin**

$n_k \leftarrow n_k - 1$;

If $n_k == 0$ **then** H ← H $\cup \{\mathbf{X_k(t)}\}$;

End If.

End For.

End For.

Set i ← i + 1 and Front _ Set(i) ← H.

End While.

V. **Return** Front _ Set.

End.

VII. EXPERIMENTS AND RESULTS

The experiment was carried out on a simulated environment on Intel Core 2 Duo processor architecture with clock speed of 2GHz using MATLAB. The NSBC algorithm was run for 200 generations where population size was initialized to 50. In each generation, each of the food sources is decoded to obtain the corresponding PPI network. Results are taken for different possible positions of a protein within the binding site of interacting proteins, and the evolved PPI network having the lowest energy value is taken as the solution.

In order to identify the PPI network, satisfying all three objectives as stated in (11), (12) and (14), we determine the best food source in a single step of movement of the box. The best food source is obtained from the Pareto front after the NSBC converges. Since all food sources in the Pareto front are equally good, to select the one among many possible solutions, we normalize all the three objectives for the individual food source in the Pareto front. Let ${}^iC_1$, ${}^iC_2$ and ${}^iC_3$ be the measure of three objective functions of the i-th food source ($\mathbf{X_i}$) in the Pareto front. The normalization here has been accomplished by using the following operation. Let ${}^iC^*_1$, ${}^iC^*_2$ and ${}^iC^*_3$ be the respective normalized value of the fitness measures for $\mathbf{X_i}$. Then, we define

$$ {}^iC^*_1 = \frac{{}^iC_1}{\sum_{j=1}^{NP} {}^jC_1},\quad {}^iC^*_2 = \frac{{}^iC_2}{\sum_{j=1}^{NP} {}^jC_1},\quad {}^iC^*_3 = \frac{{}^iC_3}{\sum_{j=1}^{NP} {}^jC_1} \qquad (23)$$

The above process is repeated for all food sources in the Pareto front. Now to determine the effective food source to be used for PPI network identification, we take a new measure by taking the product of normalized ${}^iC^*_1$, ${}^iC^*_2$ and ${}^iC^*_3$, for the food sources present in the Pareto front. Let

$$\text{Prod}_i = ({}^iC^*_1 \times {}^iC^*_2 \times {}^iC^*_3) \qquad (24)$$

be a composite measure of time and energy optimality of the i-th food source. The effective food source $\mathbf{X_j}$ having the smallest Prod_j for j=1 to NP is now identified for determining the connectivity between proteins in the network.

For experiments, we have considered the proteins of three well-known existing PPI networks available in the literature:

1. The first network consists of eleven proteins: PDGFRB, PIK3R1, PTPN11, RASA1, PDGFB, PLCG1, NCK1, GRB2, SLC9A3R1, PIK3R2, and FYN. These proteins

regulate cell growth and division. In particular, they play a significant role in blood vessel formation (angiogenesis), the growth of blood vessels from already-existing blood vessel tissue. Uncontrolled angiogenesis is a characteristic of cancer.

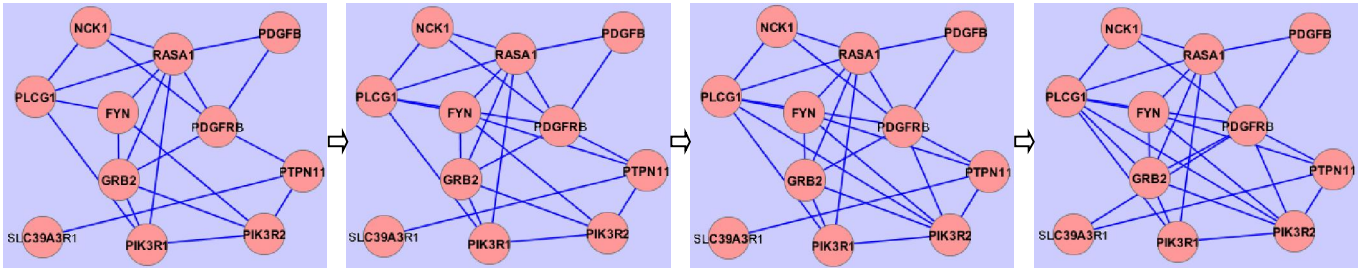
2. The second network consists of eleven proteins: AKT1, FOXO1, MTOR, MDM2, CDKN1B, RAC1, FOXO3, TSC2, BAD, NOS3, and INS. They play a crucial role in Huntington's disease, Parkinson's disease, bipolar disease and schizophrenia.
3. The final network also consists of eleven proteins: BRCA1, RBBP8, TP53, ATM, UIMC1, BRCA2, ESR1, BRIP1, BARD1, FANCD2, and H2AFX. These proteins lead to an increased risk for breast cancer as part of a hereditary breast-ovarian cancer syndrome.

The active sites conformations of the proteins in the networks of interest are obtained from Protein Data Bank [http://www.rcsb.org/pdb/home/home.do] and Active Site Prediction Server [http://www.scfbio-iitd.res.in/dock/ActiveSite.jsp]. In this section we test the relative performance of our algorithm with classical Differential Evolution for Multi-objective Optimization (DEMO) [53], Multi-Objective Particle Swarm Optimization (MOPSO) [54], [55], Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [56], and Artificial Bee Colony (ABC) [44], and Differential Evolution (DE) [47] algorithms for identifying the PPI network problem.

The structure evolved using NSBC, DEMO, MOPSO and NSGA-II are given respectively in Fig 3(a)-(c), Fig 4(a)-(c), Fig 5(a)-(c), and Fig 6(a)-(c).The evolvable structures of the PPI networks using ABC- based simulations are shown in Fig. 7(a)-(c). The structures of PPI networks evolved using DE are pictorially represented in Fig. 8(a)-(c). Fig 9 (a)-(c) represent the three dimensional structure of the identified PPI network obtained in PyMOL using NSBC. For the sake of comparison, the energy values of the PPI networks (obtained by the NSBA, DEMO, MOPSO, NSGA-II, ABC and DE based method) as well as the difference in phylogenetic sequences of interacting proteins are computed and are presented in Tables II and III respectively. The phylogenetic profiles of these proteins are obtained from PhyloPat: Phylogenetic Patterns (http://www.cmbi.ru.nl/cdd/phylopat/52/).

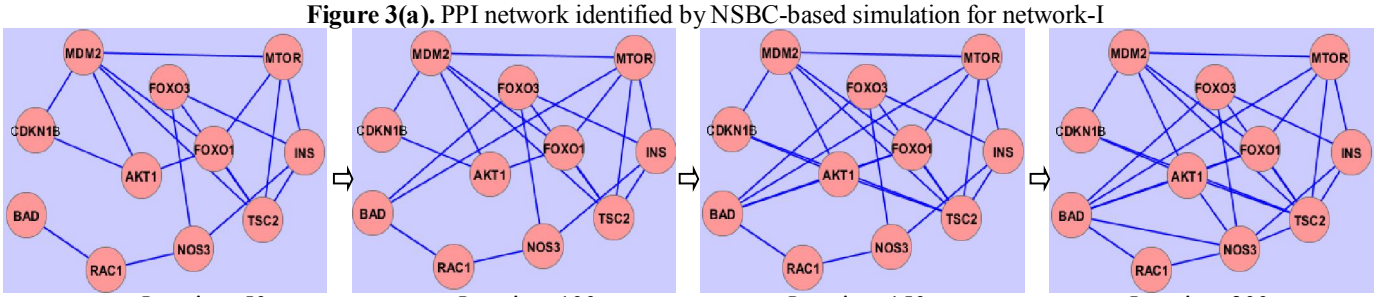
Significantly smaller difference in phylogenetic sequences of interacting proteins using the proposed NSBC algorithm is apparent. It indicates that the interacting proteins found by NSBC-based simulation are more strongly linked than those obtained by other simulation. Lower energy values of PPI suggest better stability of the networks. As seen from Table II, in all the cases NSBC provides more stable PPI networks that are associated with lower energy values.

In order to determine the correctness in predicting the PPI network structure, we refer to the STRING Database (http://string-db.org/newstring_cgi/show_input_page.pl?UserId=FEV0IvKGkenc&sessionId=FA5Pvj9O43Hr). The structures of the networks obtained from STRING of NSBC, DEMO, MOPSO, NSGA-II, ABC and DE-based simulations are listed in Table-IV along with their corresponding energy values. As evident, the structures as well as the docking energy values of the networks designed using NSBC are closer to those proposed by the database. This indicates that the PPI network conformations obtained by NSBC- based simulation are more stable.



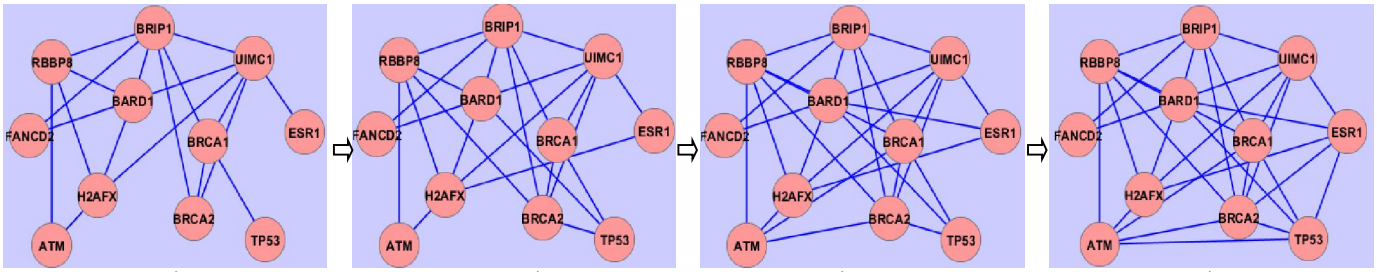
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 3(a). PPI network identified by NSBC-based simulation for network-I



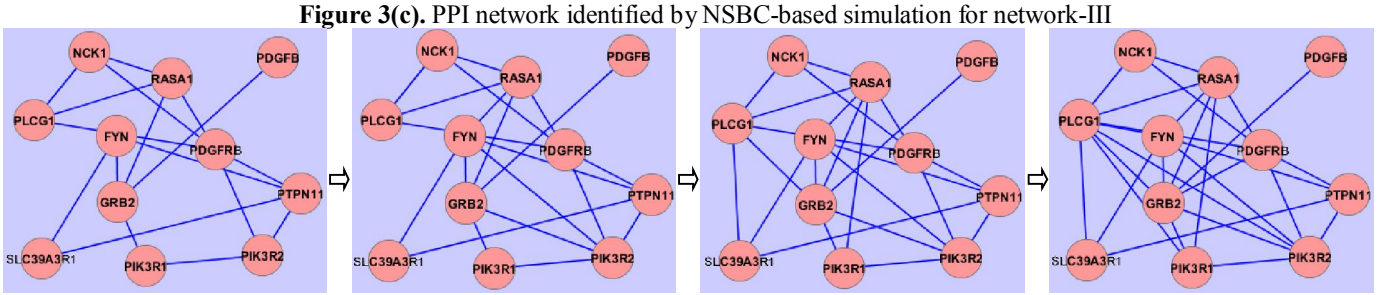
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 3(b). PPI network identified by NSBC-based simulation for network-II



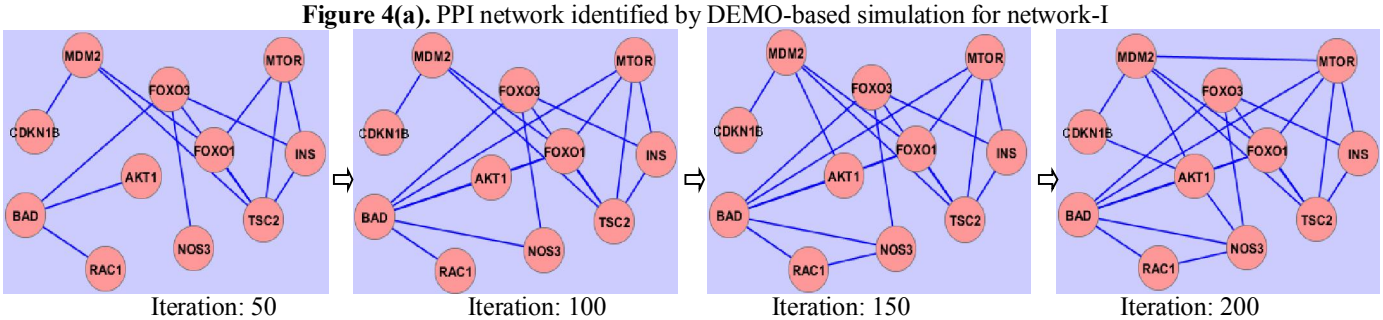
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 3(c). PPI network identified by NSBC-based simulation for network-III



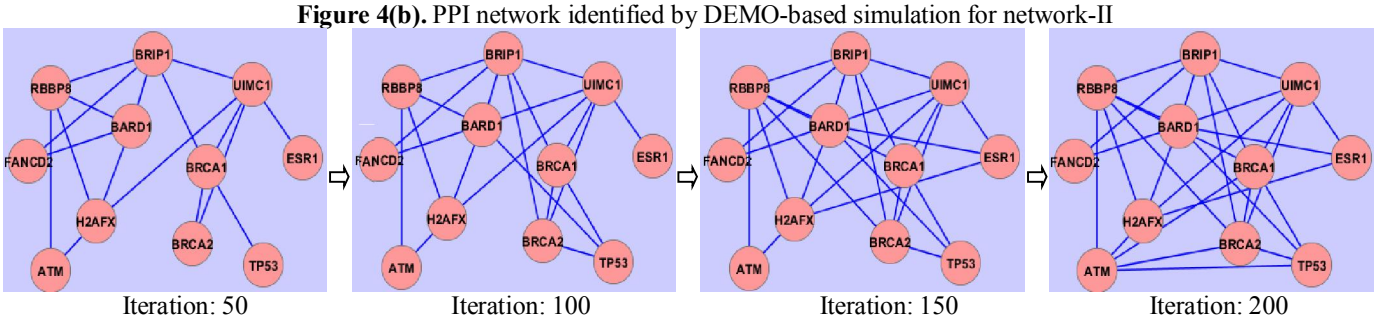
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 4(a). PPI network identified by DEMO-based simulation for network-I



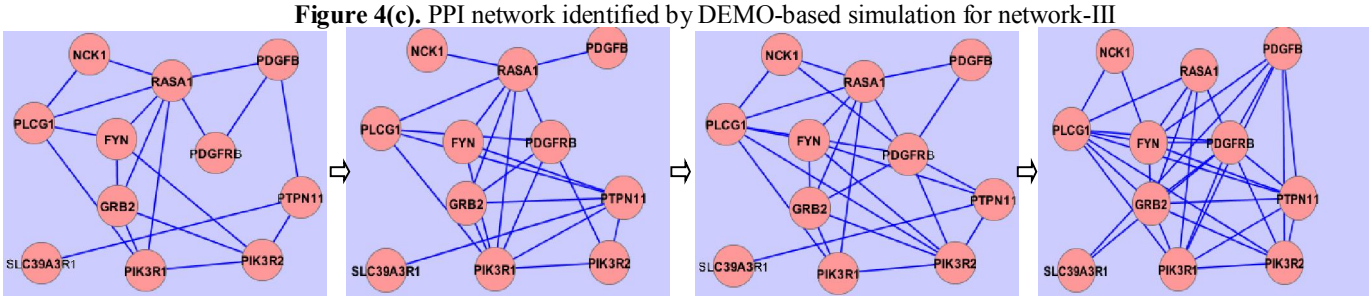
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 4(b). PPI network identified by DEMO-based simulation for network-II



Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 4(c). PPI network identified by DEMO-based simulation for network-III



Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 5(a). PPI network identified by MOPSO-based simulation for network-I

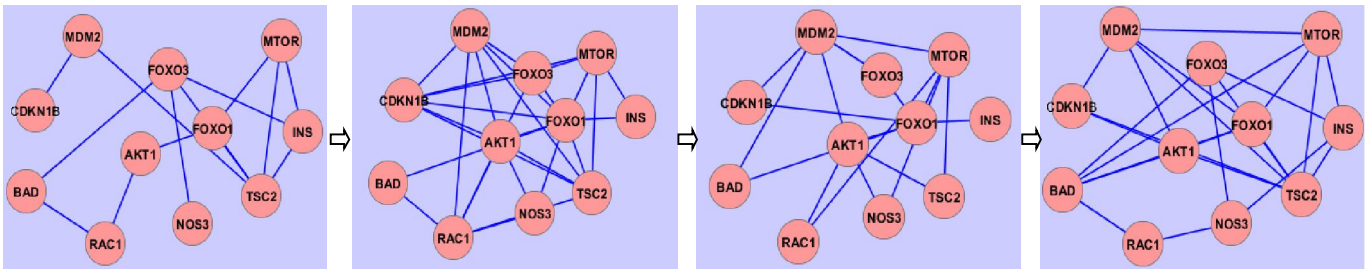


Figure 5(b). PPI network identified by MOPSO-based simulation for network-II

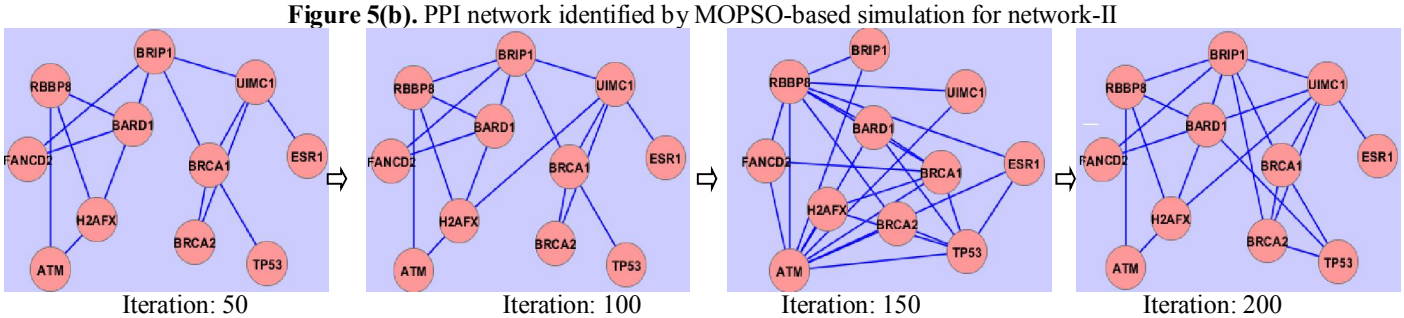


Figure 5(c). PPI network identified by MOPSO-based simulation for network-III

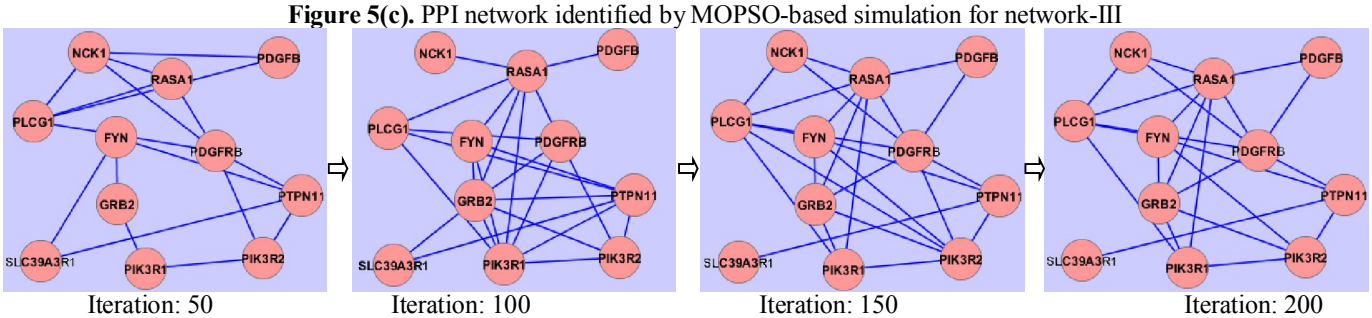


Figure 6(a). PPI network identified by NSGA-II-based simulation for network-I

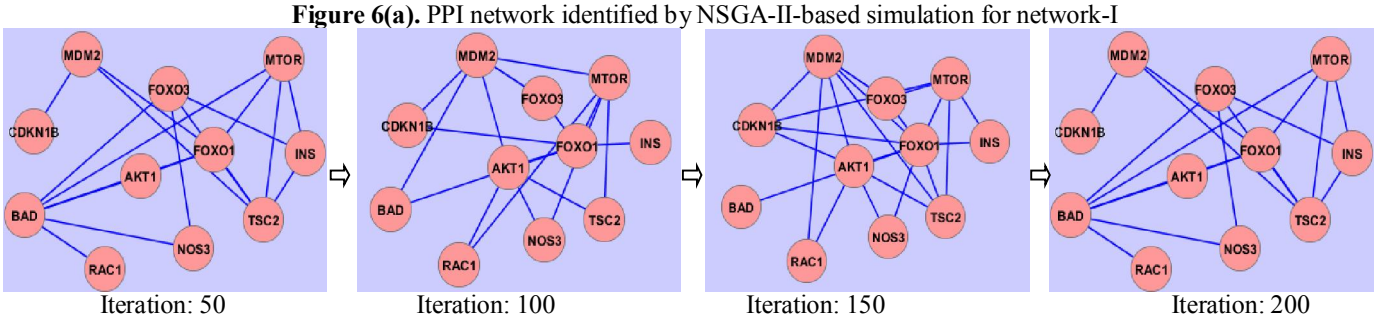


Figure 6(b). PPI network identified by NSGA-II-based simulation for network-II

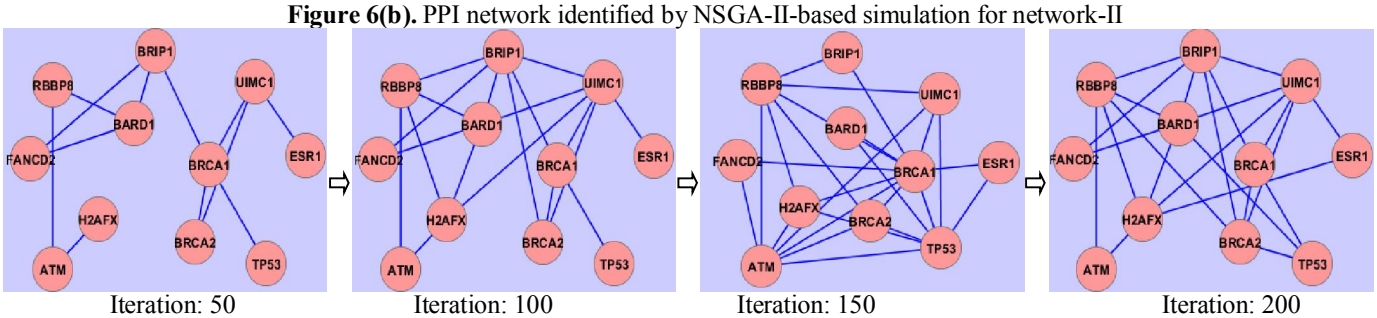
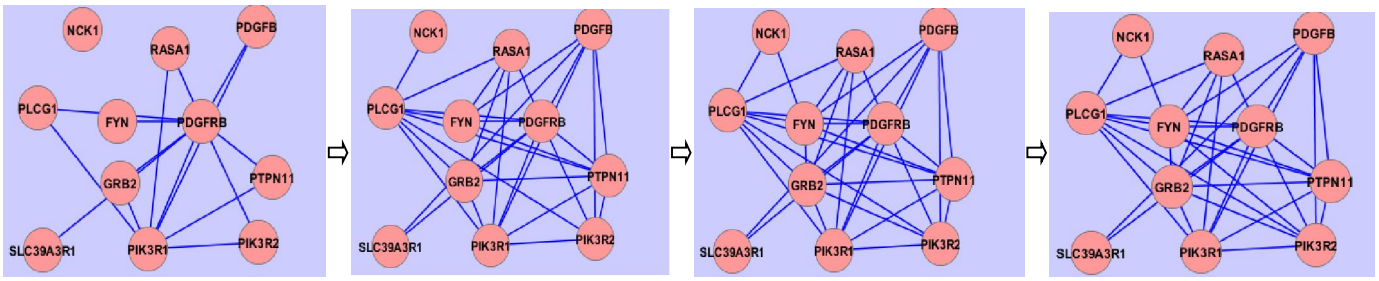
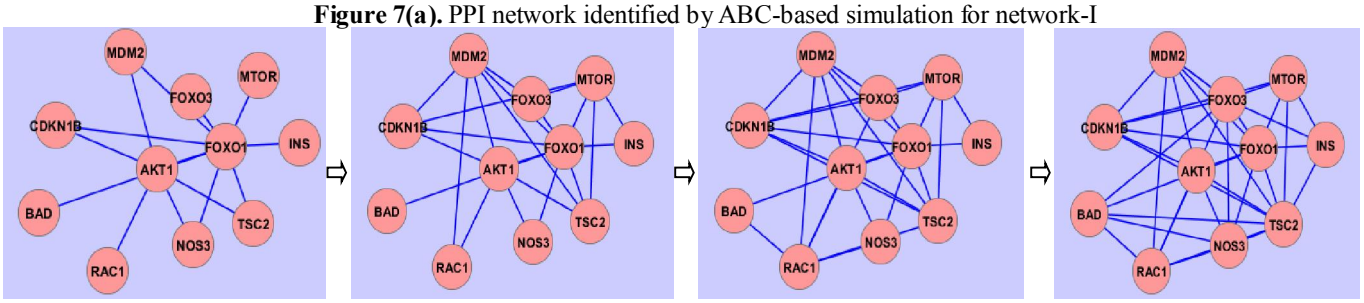


Figure 6(c). PPI network identified by NSGA-II-based simulation for network-III



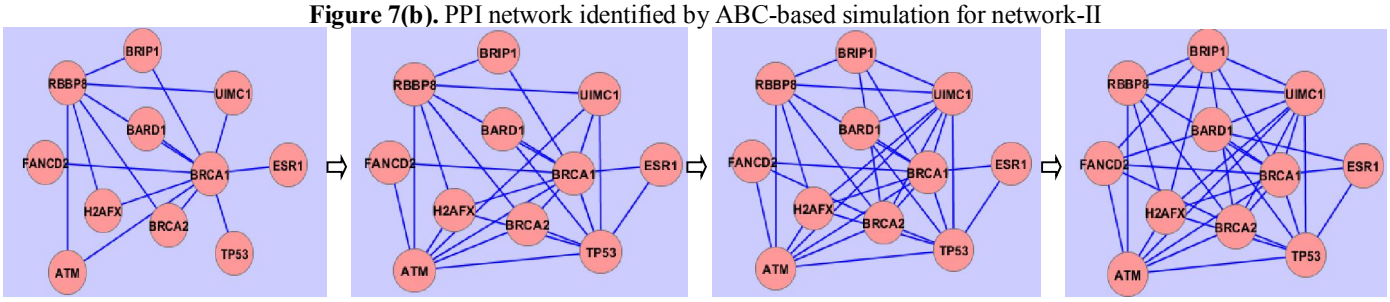
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 7(a). PPI network identified by ABC-based simulation for network-I



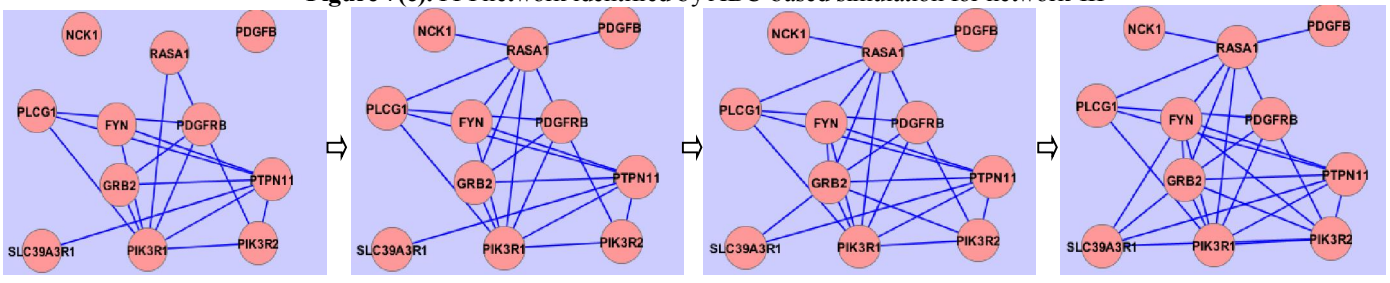
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 7(b). PPI network identified by ABC-based simulation for network-II



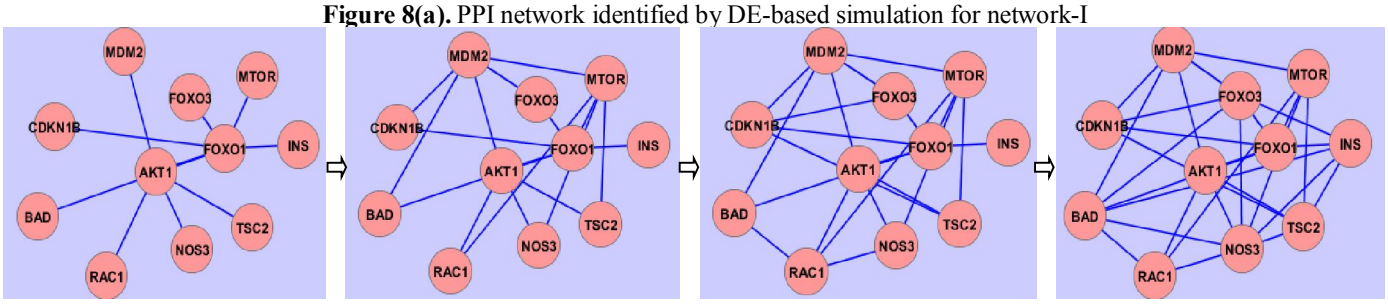
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 7(c). PPI network identified by ABC-based simulation for network-III



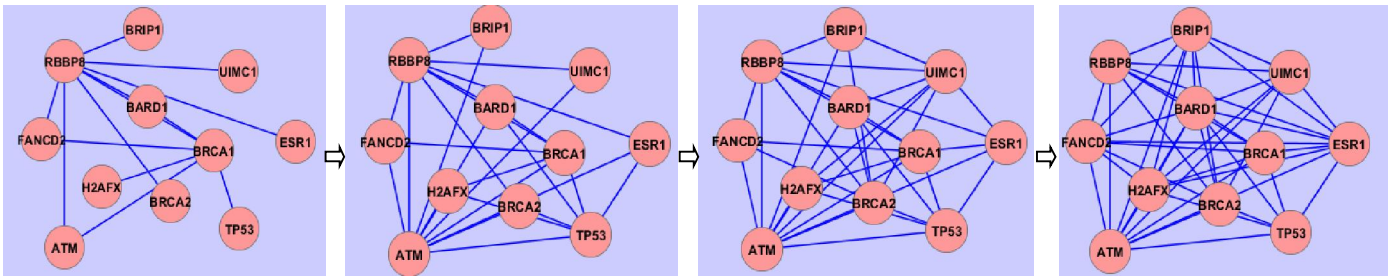
Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 8(a). PPI network identified by DE-based simulation for network-I



Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 8(b). PPI network identified by DE-based simulation for network-II



Iteration: 50 Iteration: 100 Iteration: 150 Iteration: 200

Figure 8(c). PPI network identified by DE-based simulation for network-III

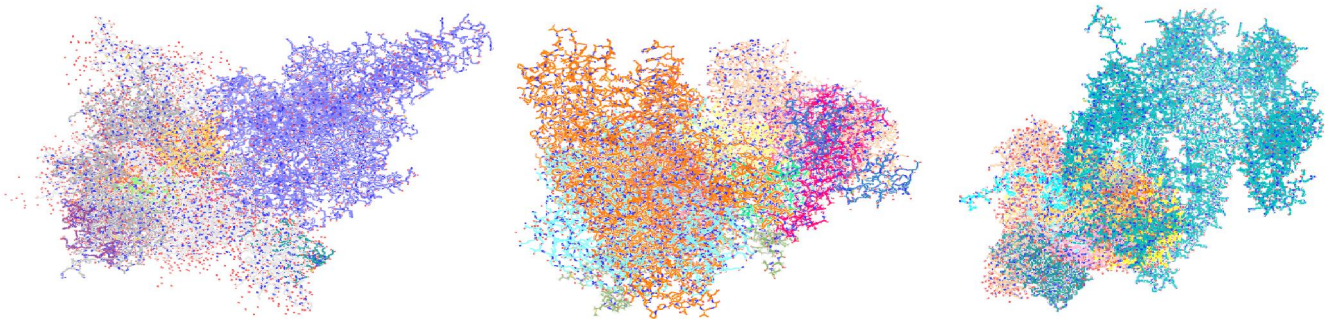


Figure 9. Three dimensional geometry of the PPI recognized by NSBC-based simulation for (a) network-I (b) network-II (c) network-III

TABLE-II

Difference in phylogenetic sequences of interacting proteins in the identified PPI

Process	Network-I	Network-II	Network-III
NSBC	1953	1862	2059
DEMO	2007	1884	2105
MOPSO	2091	1895	2168
NSGA-	2105	1927	2437
ABC	2125	1943	2762
DE	2590	2169	3038

TABLE-III

CHARMM energy values of the identified PPI networks in Kcal/mol

Process	Network-I	Network-II	Network-III
NSBC	16.857	11.373	30.748
DEMO	19.945	13.919	31.288
MOPSO	21.221	14.939	31.339
NSGA-II	22.767	16.172	31.464
ABC	23.177	17.237	30.579
DE	36.924	40.402	41.664

TABLE-IV

Comparison of structures and energies (kcal/mole) of PPI networks obtained from Binding Database, NSBC, DEMO, MOPSO, NSGA-II, ABC and DE-based simulations

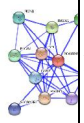
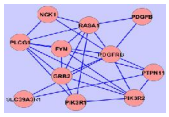
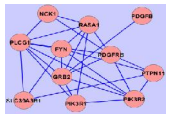
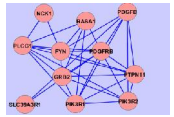
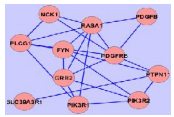
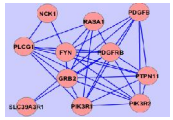
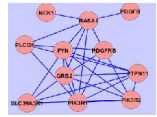
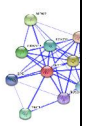
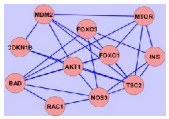
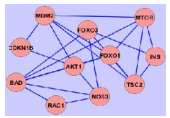
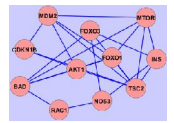
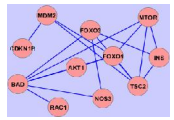
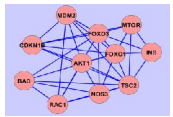
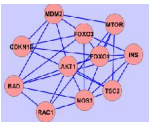
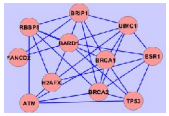
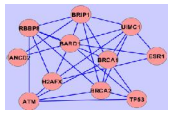
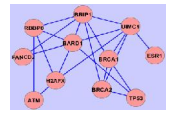
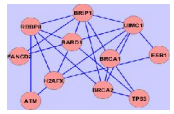
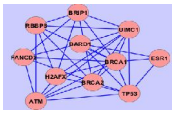
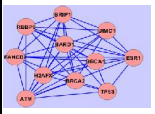
Networks	Network Conformation													
	STRING Database		NSBC-Based Simulation		DEMO-Based Simulation		MOPSO-Based Simulation		NSGA-II-Based Simulation		ABC-Based Simulation		DE-Based Simulation	
	Stru-cture	Energy	Structure	Energy	Structure	Energy	Structure	Energy	Structure	Energy	Structure	Energy	Structure	Energy
1		14.957		16.857		19.945		21.221		22.767		23.177		36.924
2		11.262		11.373		13.919		14.939		16.172		17.237		40.402
3		30.579		30.748		31.288		31.339		31.464		31.539		41.664

TABLE-V-A
Statistical Comparison

Reference Algorithm = NSBC				
Classifier Algorithm used for comparison = DEMO	Parameters used for McNemar Test		Zj	Comments on acceptance / rejection of hypothesis
	n_{01}	n_{10}		
Network-1	5	8	0.30	Accepted
Network-II	6	9	0.27	Accepted
Network-III	4	14	4.50	Rejected

TABLE-V-B
Statistical Comparison

Reference Algorithm = NSBC				
Classifier Algorithm used for comparison = MOPSO	Parameters used for McNemar Test		Zj	Comments on acceptance/ rejection of hypothesis
	n_{01}	n_{10}		
Network-1	4	8	0.75	Accepted
Network-II	3	12	4.26	Rejected
Network-III	3	14	5.88	Rejected

TABLE-V-C
Statistical Comparison

Reference Algorithm = NSBC				
Classifier Algorithm used for comparison = NSGA-II	Parameters used for McNemar Test		Zj	Comments on acceptance / rejection of hypothesis
	n_{01}	n_{10}		
Network-1	3	12	4.26	Rejected
Network-II	2	13	6.67	Rejected
Network-III	3	15	6.72	Rejected

TABLE-V-D
Statistical Comparison

Reference Algorithm = NSBC				
Classifier Algorithm used for comparison = ABC	Parameters used for McNemar Test		Zj	Comments on acceptance / rejection of hypothesis
	n_{01}	n_{10}		
Network-1	2	13	6.67	Rejected
Network-II	2	14	7.56	Rejected
Network-III	3	16	7.58	Rejected

TABLE-V-E
Statistical Comparison

Reference Algorithm = NSBC				
Classifier Algorithm used for comparison= DE	Parameters used for McNemar Test		Zj	Comments on acceptance / rejection of hypothesis
	n_{01}	n_{10}		
Network-1	2	15	8.47	Rejected
Network-II	2	16	9.38	Rejected
Network-III	2	18	11.25	Rejected

For determining the performance of the above mentioned algorithms for detecting the proper interaction between proteins we have applied McNemar's test. Let f_A and f_B be two classifiers ('1' and '0') outputs obtained by algorithm A and B, when both the algorithms used a common input phylogenetic sequences R. We now define a null hypothesis:

$$P_{r\,R,\,x}[f_A(x) = f(x)] = P_{r\,R,\,x}[f_B(x) = f(x)], \quad (25)$$

where, f(x) be the experimentally induced sign to map any data point x on to specific sign classes K, where f(x) is one of K=2 classes.

Let, n_{01} be the number of examples misclassified by f_A but not by f_B and n_{10} be the number of examples misclassified by f_B but not by f_A.

Then following (26), we define a statistic,

$$Z = \frac{\left(|n_{01} - n_{10}| - 1\right)^2}{n_{01} + n_{10}} \quad (26)$$

Let, A is our NSBC algorithm and B is any one of the remaining five competitor algorithms (DEMO, MOPSO, NSGA-II, ABC, and DE) algorithm. We thus evaluate Z_j which denotes the comparator statistic of misclassification between the NSBC (Algorithm: A) and the competitor algorithm (Algorithm: B) for the j-th network.

Tables V-A to V-E are evaluated to obtain Z_1 through Z_3 and the hypothesis is rejected if $Z_j > \chi^2_{1,\,0.95} = 3.841459$, which indicates that the probability of the null hypothesis is correct only to a level of 5% and so, we reject it. It is apparent from Tables V (A- E) NSBC has outperformed other algorithms in inferring connection between proteins on the networks in most of the cases.

VIII. CONCLUSION

In this paper, we have tried to propose a multi-objective evolutionary optimization based method to design PPI network. Non-dominated Sorting Bee Colony (NSBC) seems particularly suitable for this problem since it provides better solution quality, it does not require large number of colony size to solve optimization problem having high dimensions as it uses the exploitive process efficiently to converge minima and explorative process to provide sufficient diversity in the population. It has increased search efficiency as it is not being

sensitive to search ranges. It has also been verified in the paper that NSBC has outperformed the single objective version of Artificial Bee Colony (ABC), Differential Evolution (DE) and some multi-objective optimization algorithms such as Differential Evolution for Multi-objective Optimization (DEMO), Multi-Objective Particle Swarm Optimization (MOPSO), Non-dominated Sorted Genetic Algorithm-II (NSGA-II). Nowadays a major hope of biomedical research is to generate models explaining the clinical phenotypes observed in diseases and to develop specific cures. For these efforts, protein interaction networks are widely considered crucial contributors in two different ways: (a) by contributing to the characterization and understanding of biological processes and their aberrant function in diseases and (b) by providing a framework for the design of specific drugs. The potential of protein interaction networks in aiding the design of specific drugs have shown that disease genes tend to be hubs in protein interaction networks, exhibiting an elevated number of links, therefore making them primary targets for treatments of the respective diseases. Hence, major researches are focused to generate disease protein networks for Alzheimer disease, Huntington disease, Purkinje cell degeneration, and breast cancer. However, it remains to be seen if drugs can be efficiently designed based upon knowledge derived from protein interaction networks or networks of whole biological systems.

REFERENCE

[1] Butland G et al.,"Interaction network containing conserved and Essential protein complexes in escherichia coli," Nature 2005; 433(7025), pp.531–537.

[2] Gavin AC et al.," Functional organization of the yeast proteome by systematic analysis of protein complexes. Nature 2002; 415(6868), pp. 141–147.

[3] Giot L et al.," A protein interaction map of drosophila melanogaster," Science 2003; 30295651), pp. 1727–1736.

[4] Ho Y. et al.,"Systematic identification of protein complexes in saccharomyces cerevisiae by mass spectrometry," Nature 2002; 415 (6868),pp. 180–183.

[5] Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, Sakaki Y.," A comprehensive two hybrid analysis to explore the yeast protein interactome," Proc Natl Acad Sci USA 2001;98(8),pp. 4569–4574.

[6] Krogan NJ. et al.," Global landscape of protein complexes in the yeast Saccharomyces cerevisiae," Nature 2006; 440(7084).pp.637–643.

[7] Li S. et al.," A map of the interactome network of the metazoan c. elegans," Science 2004; 303(5657), pp. 540–543.

[8] Uetz P. et al.,"A comprehensive analysis of protein-protein Interactions in saccharomyces cerevisiae," Nature 2000; 403(6770),pp. 623–627.

[9] Dandekar T, Snel B, Huynen M, Bork P.," Conservation of gene order: a fingerprint of proteins that physically interact," Trends Biochem Sci 1998; 23(9), pp. 324–328.

[10] Enright AJ, Iliopoulos I, Kyrpides NC, Ouzounis CA.," Protein interaction maps for complete genomes based on gene fusion events," Nature 1999; 402(6757),pp. 86–90.

[11] Goh CS, Bogan AA, Joachimiak M, Walther D, Cohen FE. Co-evolution of proteins with their interaction partners. J Mol Biol 2000; 299(2): pp. 283–293.

[12] Marcotte EM, Pellegrini M, Ng HL, Rice DW, Yeates TO, Eisenberg D. Detecting protein function and protein–protein interactions from genome sequences. Science 1999; 285(5428): pp 751–753.

[13] Overbeek R, Fonstein M, D'Souza M, Pusch GD, Maltsev N. Use of contiguity on the chromosome to predict functional coupling. In Silico Biol 1999; 1(2):pp. 93–108.

[14] Pazos F, Helmer-Citterich M, Ausiello G, Valencia A. Correlated mutations contain information about protein–protein interaction. J Mol Biol 1997; 271(4): pp. 511–523.

[15] Pazos F, Valencia A. Similarity of phylogenetic trees as indicator of protein–protein interaction. Protein Eng 2001; 14(9):609–614.

[16] Pellegrini M, Marcotte EM, Thompson MJ, Eisenberg D, Yeates TO. Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. Proc Natl Acad Sci USA 1999; 96(8):4285–4288.

[17] Jung-Hsien Chiang, Tsung-Lu Michael Lee," In Silico Prediction of Human Protein Interactions Using Fuzzy–SVM Mixture Models and Its Application to Cancer Research," IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 16, NO. 4, page 1087-1095, AUGUST 2008.

[18] A Benjamin et. al. De ciphering Protein Protein Interactions. Part I. Experimental Techniques and Databases. PLoS Comput Biol 3(3):e42, 2007.

[19] A Benjamin et. al. Deciphering Protein Protein Interactions. Part II. Computational Methods to Predict Protein and Domain Interaction Partners. PLoS Comput Biol 3(4):e43, 2007.

[20] A Valencia, F. Pazos, Computational methods for the prediction of protein interactions. Current Opinion in Structural Biology 12, 368-373, 2002.

[21] Gaasterland T, Ragan MA. Microbial gene scapes: phyletic and functional patterns of ORF distribution among prokaryotes. Microb Comp Genomics 1998; 3(4):199–217.

[22] Young-Rae Cho, Aidong Zhang,"Predicting Protein Function by Frequent Functional Association Pattern Mining in Protein Interaction Networks," IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE, VOL. 14, NO. 1, Page 30-36, JANUARY 2010

[23] Jianxin Wang, Jun Ren, Min Li, Fang Xiang Wu," Identification of Hierarchical and Overlapping Functional Modules in PPI Networks," IEEE

TRANSACTIONS ON NANOBIOSCIENCE, VOL. 11, NO. 4, page 386-393, DECEMBER2012.

[24] Karthik Raman, "Construction and Analysis of Protein–Protein Interaction Networks", Automated Experimentation, February 15, 2010, Vol. 2.

[25] Nitin Bharadwaj, and Hui Lu, "Correlation between Gene Expression Profiles and Protein-Protein Interactions within and across Genomes", Bioinformatics, March 29, 2005, Vol. 21(11), pp. 2730-2738.

[26] Riley R., Lee C., Sabatti C., and Eisenberg D., "Inferring Protein-Domain Interactions from Databases of Interacting Proteins", Genome Biology, September 19, 2005, Vol. 6(10).

[27] Deng M., Mehta S., Sun F., and Chen T., "Inferring Domain-Domain Interactions from Protein-Protein Interactions", Genome Research, October, 2002, Vol. 12(10), pp. 1540-1548.

[28] Lee H., Deng M., Sun F., and Chen T., "An Integrated Approach to the Prediction of Domain-Domain Interactions", BMC Bioinformatics, May 25, 2006, Vol 7, pp. 269.

[29] Li X. L., Tan S. H., Ng S. K., "Improving Domain-Based Protein Interaction Prediction using Biologically-Significant Negative Dataset", International Journal of Data Mining and Bioinformatics, 2006, Vol. 1(2), 138-149.

[30] Woo-Hyuk Jang, Suk-Hoon Jung, Dong-Soo Han," A Computational Model for PredictingProtein Interactions Based on Multidomain Collaboration,"IEEE/ACM TRANSACTIONS ON COMPUTATIONAL BIOLOGY AND BIOINFORMATICS, VOL. 9, NO. 4,page 1081-1090, , JULY/AUGUST 2012.

[31] Patrick Aloy, and Robert B. Russell, "Interrogating Protein Interaction Networks through Structural Biology", PNAS, April 23, 2002, Vol. 99(9), pp. 5896-5901.

[32] Patrick Aloy, and Robert B. Russell, "InterPreTS: Protein Interaction Prediction through Tertiary Structure", Bioinformatics, July 15, 2002, Vol. 19(1), pp. 161-162.

[33] Kiel C., Foglierini M., Kuemmerer N., Beltrao P, and Serrano L., "A Genome-wide Ras-Effector Interaction Network", Journal of Molecular Biology, July 27, 2007, Vol. 370(5), pp. 1020-1032.

[34] Pellegrini M., Macrotte E. M., Thompson M. J., Eisenberg D., and Yeates T. O., "Assigning Protein Functions by Comparative Genome Analysis: Protein Phylogenetic Profiles", Proceedings of the National Academy of Science USA, April 13, 1999. Vol. 96(8), pp. 4285-4288.

[35] Martyn_Kennedy, Scott A. Taylor, Petr Nádvorník, Hamish G. Spencer," The phylogenetic relationships of the extant pelicans inferred from DNA sequence data," Molecular Phylogenetics and Evolution, 66(1),pp. 215-222,January 2013.

[36] Michael DJ Lynch, Andrea K Bartram, Josh D Neufeld," Targeted recovery of novel phylogenetic diversity from next-generation sequence data," The ISME Journal (2012) 6,pp. 2067–2077.

[37] Ramani A. K., and Marcotte E. M., "Exploiting the Co-evolution of Interacting Proteins to Discover Interaction Specificity", Journal of Molecular Biology, March 14, 2003, Vol. 327(1), pp. 273-284.

[38] Jothi R., Kann M. G., Przytycka T. M., "Predicting Protein-Protein Interaction by Searching Evolutionary Tree Amorphism Space", Bioinformatics, June 21, 2005.

[39] Yann C, Siarry P (Eds): Multiobjective Optimization: Principles and Case Studies. Springer-Verlag, Berlin, Germany (2004).

[40] Gillet VJ, Willett, P, Bradshaw J, and Green DVS: Selecting combinatorial libraries to optimize diversity and physical properties. J Chem Inf Comput Sci (1999) 39(1):169-177.

[41] Tsoka, S. and Ouzounis, C. A. (2000) "Prediction of protein interactions: metabolic enzymes are frequently involved in gene fusion". Nature Genet, 26, 141–142.

[42] Dandekar, T., Snel, B., Huynen, M. and Bork, P. (1998) "Conservation of gene order: a fingerprint of proteins that physically interact". Trends Biochem Sci, 23, 324–328.

[43] A. Datta, V. Talukdar, A. Konar, and L. C. Jain, "Neuro-swarm hybridization for protein tertiary structure prediction", International Journal of Hybrid Intelligent Systems 5 (2008) 153-159.

[44] P. Rakshit, A. Chowdhury, A. Konar and A. K. Nagar, "Evaluating the designing Perspective of Protein-protein Interaction Network using Evolutionary Algorithm", in IEEE Fourth World Nature and Biologically inspired Computing (NaBIC), November, pp.130-137,2012.

[45] P. Rakshit, A. K Sandhu, P. Bhattacharjee, A. Konar and R.J anarthanan, "Multi-Robot Box Pushing Using Non-Domonated Sorting Bee Colony Optimization Algorithm", in Swarm, Evolutionary and Memetic computing, lecture notes in Computer science, Vol 7076,pp601-609,2011.

[46] P. Bhattacharjee, P. Rakshit, I. Goswami, A. Konar, and A. K. Nagar, "Multi-Robot Path-Planning Using Artificial Bee Colony Optimization Algorithm", Nature and Biologically Inspired Computing, October, 2011, pp. 219-224.

[47] R. Storn, K. V. Price, and J. Lampinen, "Differential Evolution–A Practical Approach to Global Optimization", Berlin, Germany: Springer- Verlag, 2005.

[48] U.K. Chakraborty, "Advances in Differential Evolution", Springer, Heidelberg, New York, 2008.

[49] A. Chakraborty and A. Konar, "A Distributed Multi – Robot Path Planning Using Particle Swarm Optimization," in 2nd National Conference on Recent Trends in Information Systems, pp216-221, 2008.

[50] D. Karaboga, "An idea based on honey bee swarm for numerical optimisation," Technical Report-TR06,

Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[51] B. Basturk, D. Karaboga, "An Artificial Bee Colony (ABC) Algorithm for Numeric function Optimization," IEEE Swarm Intelligence Symposium 2006, May 12-14, 2006, Indianapolis, Indiana, USA.

[52] D. Karaboga, B. Basturk, On the performance of artificial bee colony (ABC) algorithm, In: Applied Soft Computing 8 (2008) 687-697.

[53] T. Robic, and B. Philipic, "DEMO: Differential Evolution for Multiobjective Optimization" In Evolutionary Multi-Criterion Optimization, Third International Conference, EMO 2005 Coello Coello, C.A., Aguirre, A.H., Zitzler, E., Eds.; Springer Lecture Notes in Computer Science: Guanajuato, Mexico, 2005; Vol. 3410, pp. 520-533.

[54] C.A. Coello Coello, and M. Lechuga, "MOPSO: A Proposal for Multiple Objective Particle Swarm Optimization", in: Proceedings of IEEE Congress of Evolutionary Computation, vol. 2, May 2002, pp. 1051–1056.

[55] C. A. Coello, G. T. Pulido and M. S. Lechuga, "Handling Multiple Objectives with Particle Swarm Optimization", IEEE Transactions on Evolutionary Computation, 2004, vol. 3, pp. 256-279.

[56] A.Deb, A. P. S. Agarwal and T. Meyarivan,(1998) "A fast and elitist multi-objective genetic algorithm: NSGA II", IEEE Transactions on Evolutionary Computation, vol. 2, pp. 162-197.

Archana Chowdhury received the degree of B.E in Computer Science and Engineering from Maharana Pratap College of Technology, Gwalior, India and M.E in Computer Engineering from Jadavpur University, Kolkata in 2008 .She is currently working towards her PhD degree from Jadavpur University under the guidance of Prof. Amit Konar.
She has served as a reviewer for many conference and journal papers. Her research interests include Artificial Intelligence, Fuzzy Logic, Evolutionary Computation, Pattern Recognition, Bioinformatics and Data Mining.

Amit Konar received the B.E. degree from Bengal Engineering and Science University (B.E. College), Howrah, India, in 1983 and the M.E. Tel E, M. Phil., and Ph.D. (Engineering) degrees from Jadavpur University, Calcutta-700032, India, in 1985, 1988, and 1994, respectively. In 2006, he was a Visiting Professor with the University of Missouri, St. Louis.
He is currently a Professor with the Department of Electronics and Tele-Communication Engineering (ETCE), Jadavpur University, where he is the Founding Coordinator of the M.Tech. program on intelligent automation and robotics. He has supervised fifteen Ph.D. theses. He has over 200 publications in international journal and conference proceedings. He is the author of eight books, including two popular texts Artificial Intelligence and Soft Computing (CRC Press, 2000) and Computational Intelligence: Principles, Techniques and Applications (Springer, 2005). He serves as the Associate Editor of IEEE Transactions on Systems, Man and Cybernetics, Part-A, and IEEE Transactions on Fuzzy Systems. His research areas include the study of computational intelligence algorithms and their applications to the various domains of electrical engineering and computer science. Specifically, he worked on fuzzy sets and logic, neurocomputing, evolutionary algorithms, Dempster–Shafer theory, and Kalman filtering, and applied the principles of computational intelligence in image understanding, VLSI design, mobile robotics, pattern recognition, brain-computer interfacing and computational biology. He was the recipient of All India Council for Technical Education (AICTE)-accredited 1997–2000 Career Award for Young Teachers for his significant contribution in teaching and research.

Pratyusha Rakshit received the B. Tech degree in Electronics and Communication Engineering (ETCE) from Institute of Engineering and Management, Salt Lake, Sector-V, India, and M.E. degree in Control Engineering from Electronics and Telecommunication Engineering Department, Jadavpur University, Kolkata-700032, India in 2010 and 2012 respectively. She is currently pursuing her Ph.D. degree in identification of protein and gene regulatory networks to control neuronal firing in mammals under the guidance of Prof. Amit Konar at Jadavpur University, India.
Her principal research interests include Artificial Intelligence, Evolutionary Computation, Bioinformatics, Pattern Recognition, Fuzzy Logic, Cognitive Science and Human-Computer-Interaction. She is an author of over 15 papers published in top international journals and conference proceedings. She served as a reviewer in IEEE-ReTIS 2011, IEEE-TFS 2012, and IJAISC 2012.

Atulya K. Nagar holds the Foundation Chair, as Professor of Computer and Mathematical Sciences, at Liverpool Hope University and is Head of the Department of Mathematics and Computer Science. A mathematician by training, Professor Nagar possesses multi-disciplinary expertise in Natural Computing, Bioinformatics, Operations Research and Systems Engineering. He has an extensive background and experience of working in Universities in the UK and India.
He has been an expert reviewer for the Biotechnology and Biological Sciences Research Council (BBSRC) grants peer-review committee for Bioinformatics Panel and serves on Peer-Review College of the Arts and Humanities Research Council (AHRC) as a scientific expert member
He has coedited volumes on Intelligent Systems, and Applied Mathematics; he is the Editor-in-Chief of the International Journal of Artificial Intelligence and Soft Computing (IJAISC) and serves on editorial boards for a number of prestigious journals as well as on International Programme Committee (IPC) for several international conferences. He received a prestigious Commonwealth Fellowship for pursuing his Doctorate in Applied Non-Linear Mathematics, which he earned from the University of York in 1996. He holds BSc (Hons.), MSc, and MPhil (with Distinction) from the MDS University of Ajmer, India. Prior to joining Liverpool Hope, Prof. Nagar was with the Department of Mathematical Sciences, and later at the Department of Systems Engineering, at Brunel University, London.