



# Identification of Functional Modules in Dynamic Weighted PPI Networks by a Novel Clustering Algorithm

Yimin Mao<sup>(✉)</sup>, Xin Yu, and Haiwan Zhu

School of Information Engineering, Jiangxi University of Science  
and Technology, Ganzhou 341000, Jiangxi, China  
mymlyc@163.com

**Abstract.** The Density-Based Spatial Clustering of Application with Noise algorithm (DBSCAN) suffers the limitations of selecting global parameters and having the low accuracy in recognizing overlapping protein complexes. In order to overcome the disadvantage of slow convergence and being vulnerable to trap in local optima in Artificial Bee Colony algorithm (ABC), we designed a method with novel weights and distance calculated which is suitable for network topology and the interaction between proteins. Furthermore, a truncation-championship selection mechanism (TCSM) was proposed to avoid local optimum when onlooker bees search nectar source. Meanwhile, we present the adaptive step strategy (ASS) to improve the clustering speed in ABC algorithm. Finally, in order to overcome the shortcoming which is unable to identify protein complexes in the DBSCAN algorithm, a strategy is proposed to optimize the clustering result. The experimental results on superior precision and recall parameters demonstrate that our method has competitive performance for identifying protein complexes.

**Keywords:** DBSCAN · ABC · Dynamic weighted PPI network · TCSM · Adaptive step strategy · Protein complexes

## 1 Introduction

Large-scale experiments are producing huge data sets of protein-protein interactions making it increasingly difficult to visualize and analyze the information contained in these data [1]. Being able to apply computational methods can alleviate a lot of problems in this regard. Therefore, a general trend is to represent the interactions as a network/graph and to apply suitable graph algorithms to extract necessary information. In the post-genomic era, one of the most important issues is to find protein complexes from the protein-protein interaction (PPI) networks. Protein complexes can help us to predict the functions of proteins [2], and they are also useful to understand and explain certain biological processes [3].

In recent years, several algorithms based on graph clustering, dense region finding or clique finding have been developed to discover protein complexes from PPI networks, including Clustering-based Maximal Cliques (CMC) [4] and Core-attachment Based Method (COACH) [5]. Although the above methods were shown to effectively

identify protein complexes, their results are sensitive to noisy data, which is still a challenge we have to face [6]. Density-Based Spatial Clustering of Applications with Noise [7] proposed by Ester implemented the density-based strategy with the considerations of noises in network data. It can efficiently discover the clusters with arbitrary size, shape and number in a large dataset. It is noise tolerant and independent of ordering of data objects. However, it needs two input parameters,  $\epsilon$ —the radius of neighborhood and *Minpts*—the density threshold, which are domain specific and thus hard to be determined [8]. For density-based methods, it is difficult to decide the input parameters that the algorithm is sensitive to, hence detecting global parameters of DBSCAN for the best clustering result has become an essential and challenging research problem.

Swarm intelligence optimization algorithms are commonly used to solve optimization problems by simulating the collective behavior of social insects. Thus, swarm intelligence has also attracted the interest of many research scientists of related fields. Such as in 2014, Lei et al. proposed F-MCL clustering model [9, 10] which automatically adjusts the parameters of Markov clustering method by using Firefly algorithm. Ding et al. proposed a method named P-DBSCAN [11] which combines DBSCAN and Pigeon-Inspired Optimization Algorithm (PIO) algorithm, for addressing the shortcoming of DBSCAN such as the difficulties of determining the parameters and the unreasonable strategy of using a set of global static parameters for a large varying dense database. The performance of above methods based on swarm intelligence optimization is better than existing clustering algorithms. Artificial Bee Colony (ABC) algorithm [12–14] is one of the swarm-based algorithms. Experiments show that the performance of the ABC is better than or similar to those of other population-based algorithms with the advantage of employing fewer control parameters. However, the results of DBSCAN are sensitive to global parameters, cannot identify protein complexes which are still challenges we have to face.

In order to address the aforementioned challenge, in this paper, a novel algorithm (IABC-DBSCAN) is proposed. The main contributions of this paper are summarized as follows:

- (1) The originally weighted strategy and foot distance method (FD) were proposed.
- (2) The performance of ABC algorithm is improved through adaptive step size strategy (ASS) and truncation-champion selection mechanism (TCSM).
- (3) The parameters of the improved ABC are served as the input values in DBSCAN algorithm.
- (4) The tentative strategy (TS) is designed for overcoming the defect in DBSCAN algorithm, which cannot identify protein complexes.

## 2 Methods

### 2.1 Artificial Bee Colony Optimization Algorithm

The ABC algorithm is inspired from intelligent foraging behavior of honey bee swarm searching for food [15, 16]. In bee colony algorithm, position of each nectar source

corresponds to a possible solution for the optimization problem and the amount of nectar source depicts the fitness value of the associated solution. The total bees in colony can be categorized into three groups: employed, onlooker, and scout. The number of employed and onlooker corresponds to the number of solutions in population.

Employed and onlooker bees search neighborhoods by Eq. (1):

$$new\_x_{ij} = x_{ij} + r_{ij}(x_{ij} - x_{kj}) \tag{1}$$

where  $k(k \in \{1, 2, \dots, N\})$  and  $j(j \in \{1, 2, \dots, d\})$  are random numbers and  $k \neq i$ ,  $r_{ij}$  is a random number from 0 to 1.

Onlooker bees select nectar source according to the probability  $P_i$ , which is calculated as follows:

$$p_i = \frac{fval(i)}{\sum_{j=1}^N fval(j)} \tag{2}$$

If a nectar source has not been updated after *limit* times searching, we should abandon it and make scouts search novel nectar source as follows:

$$new\_pop(i) = (upbond - lbond) \bullet rand + lbond \tag{3}$$

where *upbond* and *lbond* are the upper and lower bound of the neighborhood, respectively.

### 2.2 DBSCAN Algorithm

Density Based Spatial Clustering of Applications with Noise (DBSCAN) is the precursor of density-based clustering method which can detect clusters of arbitrary shape and also handles noise or outliers effectively. DBSCAN algorithm has a quadratic time complexity with dataset size. The algorithm can be extended to large datasets by reducing its time complexity using spatial index structures like R-trees for finding neighbors of a pattern.

### 2.3 Module Similarity

The similarity of two modules is defined as the interaction degree of nodes, it is calculated as follows:

$$Sim(I_i, I_j) = \frac{\sum_{x \in I_i, y \in I_j} c(x, y)}{\max(|I_i|, |I_j|)} \tag{4}$$

where  $C(x, y)$  is:

$$c(x, y) = \begin{cases} 1 & \text{if } x = y \\ w(x, y) & \text{if } x \neq y, \text{ and } \langle x, y \rangle \in E \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

## 2.4 Highest K-Core

The highest  $k$ -core of a graph is the one with the maximal  $k$  value among all the  $k$ -cores and is denoted as  $hkc(G)$  [17]. It is the central most densely connected sub-graph. The highest  $k$ -core of  $G$  can be found in the following way: suppose the lowest degree of nodes in  $G$  is  $d$ , delete all nodes with degree  $d$ , if all remaining nodes have a least degree  $d_1$  ( $d_1 < d$ ), we will get the  $d_1$ -core of  $G$ ; if some of the remaining nodes have degree lower than or equals to  $d$ , continue delete all nodes with the least degree until all remaining nodes have a degree higher than  $d$ , or until all nodes have been deleted. In this way we can find all  $k$ -cores and the one with the maximal  $k$  value is the highest  $k$ -core

## 3 Dynamic Network

### 3.1 Construction of Dynamic Network

The construction of dynamic networks is a process of continuously adding, filtering out and adjusting protein interactions based on static PPI networks and gene expression profile [17, 18]. The dynamic information of proteins can be obtained from the gene expression profile, which contains the gene expression values of each protein in three cycles. In this paper, the average of gene expression values of the three cycles at a timestamp is used as the final gene expression value at the time point. Subsequently, we construct the dynamic PPI network based on the static PPI network and dynamic characteristic of the proteins.

### 3.2 Weighted Dynamic Network

Based on the dynamic model described, we obtain the dynamic PPI network which contains 12 static PPI sub-networks at each time point. However, there are a lot of false positives and false negatives in high throughput protein interaction, and some interaction relationships are unreliable. Therefore, a novel strategy is presented to optimize the network. In this strategy, the edge weight is evaluated by combining the GO annotations with the co-neighbor numbers of protein.

## 4 IABC-DBSCAN Algorithm

There are several defects in the process of mining protein complexes by DBSCAN. For example, the setting and selection of global parameters are difficult. In addition, the interaction of proteins cannot be measured by distance measure methods and overlapping function modules cannot be identified. In this paper, for the above problems, we propose IABC-DBSCAN algorithm based on dynamic weighted PPI network. We first construct TCSM and ASS method for avoiding premature convergence and improving the global detection capability of ABC algorithm. Then, we use improved ABC algorithm to optimize the parameters of DBSCAN. Finally, we present the hypothetical strategy to overcome the problem of overlapping complexes which cannot be identified in DBSCAN.

### 4.1 Improved Artificial Bee Colony Algorithm

**TCSM.** In the ABC algorithm, onlookers merely adopt the roulette selection mechanism (RSM) [19] to search nectar sources, and a major issue of RSM is that the entire solution space cannot be searched effectively, in addition, the RSM is weak to detect the fittest nectar source. In consequence, individuals with higher fitness value are eliminated with randomness, which affects the result of ABC algorithm. Therefore, in this paper, RSM is replaced by the truncation selection mechanism (TSM) [20], whose selection capability is stronger. In TSM, individuals first are in descending order of fitness value, and then, top  $t\%$  individuals are selected to produce next generation. The TSM is defined as:

$$P_k = \begin{cases} 1 & k \leq M \times t\% \\ 0 & k > M \times t\% \end{cases} \quad (6)$$

where  $P_i$  denotes the probability of next generation that individual  $i$  produces,  $M$  is population size, and  $t\%$  is truncation threshold, which controls the parameters of selecting strength. In order to improve the diversity of population, an adaptive truncation selection mechanism is constructed, which employs different truncation thresholds in different phase. In the preliminary phase of iteration, we set a greater threshold to ensure that the poor nectar source can be searched. While in the later phase, the threshold is set a lower value to avoid the result falling into local optimum. The truncation threshold  $t\%$  is defined by the following equation:

$$t = t_{min} + \frac{(t_{max} - t_{min})(cyc - 1)}{Maxiter - 1} \quad (7)$$

where  $t_{max}$  and  $t_{min}$  are the maximal and minimal truncation threshold, respectively. In addition,  $cyc$  is the remaining detection times by employed and onlooker bees and  $Maxiter$  is the maximal detection times by them. The diversity of population still declines by using TSM. Therefore, we combine champion selection mechanism with TSM for keeping the diversity of population. The specific implementation steps are as follows:

**Step 1:** Firstly, the individuals are in descending order of fitness value and  $k$  individuals are selected randomly from the population for constructing a group. And then select individual  $i$  with the optimal fitness value, which is calculated according to the TSM. If the fitness value of  $i$  is greater than  $M \times t\%$ , then onlookers search the neighborhoods of  $i$  and produce next generation, else implement Step.1.

**Step 2:** Step 1 is implemented for  $m$  times to produce next population.

**ASS Method.** The efficiency of finding locally optimal nectar source by the searching method in ABC algorithm is not satisfied. Therefore, inspired by the ASS method in the drosophila optimization algorithm [21], the method of onlookers searching nectar source is dynamically adjusted for improving the global searching capability of onlookers. The updated strategy is as follows:

$$x_{ij} = x_{ij} + step_{ij}r_{ij}(x_{ij} - x_{kj}) \tag{8}$$

$$step_{ij} = h_{min} + (h_{max} - h_{min}) \times \frac{\|f_i - f_{best}\|}{f_{max}} + \sigma \times randn() \tag{9}$$

where  $step_{ij}$  is adaptive step size,  $r_{ij}$  denotes a random number from 0 to 1,  $h_{max}$  and  $h_{min}$  are the maximal and minimal step size, respectively.  $|f_i - f_{best}|$  is the difference in fitness value between nectar source  $i$  and globally optimal nectar source,  $f_{max}$  is the maximal fitness value of globally optimal nectar source,  $\sigma$  is the difference between  $step_{ij}$  and the expectation of  $step_{ij}$ ,  $randn()$  is a normally distributed random number. Onlookers find the locally optimal nectar source based on the globally optimal solution, which will reduce searching times and blindness.

## 4.2 Improved DBSCAN Algorithm

**FD Method.** In the PPI network, the Euclidean Distance and Hamming Distance are not suitable to calculate the degree of interaction between nodes, because the interaction between nodes is uncertain. Therefore, in this paper,  $1 - W$  is used as the equation of distance measured, furthermore, the  $\varepsilon$ -neighborhood and core nodes are redefined. The equation of FD is defined as follows:

$$FD(u, v) = 1 - W(u, v) \tag{10}$$

For making the DBSCAN algorithm can be applied to the PPI network, the core node and  $\varepsilon$ -neighborhood are redefined according to the topological structure of PPI network. Some definitions are described firstly.

**Definition 1** ( $\varepsilon$ -neighborhood). For a point  $x \in D$ , the  $\varepsilon$ -neighborhood denotes the points whose distance from  $x$  is less than  $\varepsilon$  and interaction with  $x$ .

**Definition 2** (core point). For a point  $x \in D$ , at least  $M$  nodes whose distance from  $x$  is less than 0.65 and score  $k$  is greater than or equal to 2 in the  $\varepsilon$ -neighborhood of  $x$ .

**TS.** As the function module of protein complexes cannot be identified by the DBSCAN algorithm, we propose the TS strategy to search it by defining fuzzy nodes after clustering.

**Definition 3** (fuzzy node). The protein node is in the function module  $I_i$  and it interacts with proteins which are in other function modules.

The elaborate of TS strategy is as follows: if protein  $v$  is in the function module  $I_i$  and it interacts with proteins which are in the function module  $I_j$ , then we calculate the fitness value of protein  $v$  which is in the function module  $I_i$  and  $I_j$ , simultaneously. If the fitness value is higher than before, indicating protein  $v$  is in  $I_i$  and  $I_j$  at same time, otherwise end the process.

### 4.3 Search of Optimal Parameters by IABC

For DBSCAN algorithm, the global parameter is set unreasonable and selected difficultly. Therefore, we propose the IABC algorithm to search the optimal parameter in every dynamic sub-network. In the IABC algorithm, we first set the optimal range of parameter  $\varepsilon$ , and then initialize  $k$  nectar sources, which positions are the value of parameter  $\varepsilon$  and fitness values are identical with the fitness values in DBSCAN algorithm. Secondly, onlookers select nectar source based on TCSM method and search nectar source in local by ASS method.

### 4.4 Implementation Steps of IABC-DBSCAN Algorithm

The specific implementation steps are as follows:

**Step 1:** Produce  $n$  dynamic weighted sub-networks  $i(i = 1)$  through dynamic weighted model, which is based on expressive data of gene and static PPI data.

**Step 2:** Set parameters in sub-network  $i$ :  $M$  and  $limit$  represent the number of bees and parameter of solution, respectively. What's more,  $maxiter$  is the maximal iteration times. Initialize a group of nectar sources  $(v_1, v_2, \dots, v_k)$ , which is the position of nectar sources. Subsequently, the fitness value is calculated based on Eq. (5) and ordered in descending. The bees at corresponding position of nectar source in top  $M/2$  of fitness value are used as employed bees, and the other half are onlookers.

**Step 3:** Go into the loop and set the loop time  $iter = 1$ .

**Step 4:** By Eq. (1), employed bees search the neighborhoods and obtain the new solution, then calculate the fitness value of new solution. If  $new - fval(i) > fval(i)$ , update the position of nectar source.

**Step 5:** The onlookers select nectar source by TCMS and search the neighborhoods by ASS method. Subsequently, calculate the fitness value of new solution. If  $new - fval(i) > fval(i)$ , update the position of nectar source.

**Step 6:** Record the currently optimal solution.

**Step 7:** If the fitness value of the  $i$ -th nectar source is constant, then  $s(i) = s(i) + 1$ .

**Step 8:** If  $s(i) \geq limit$ , scouts search global by Eq. (3) for updating solution, additionally, set  $s(i) = 0$ , else implement step 10.

**Step 9:** If iteration time achieves maximum value, output the optimal solution, else  $iter = iter + 1$ .

**Step 10:** The position of optimal nectar source is used as the input value of DBSCAN. In addition, the  $i$ -th dynamic sub-network is processed by the clustering algorithm in DBSCAN. Protein complexes are searched by TS strategy for optimizing the clustering result. Subsequently, put the clustering function module in total function module.

**Step 11:** If ( $n_i < n$  is the number of dynamic sub-network), go back to Step.2, else output the function modules of protein at every time point.

## 5 Experiments and Discussion

### 5.1 Material

An experimental computer is configured with the windows 7 ultimate operating system, an Intel i5 dual-core processor, 2.5-GHz frequency and 6.0 GB of memory. The algorithm is programmed in python.

We download dynamic gene expression data set GSE3431 about the yeast metabolic cycle from GEO dataset. This dataset includes 6777 genes that cover 95% protein in the static DIP network. By using the methods for construction, we get the DPIN of DIP which contains 12 static PPI sub-networks at 12 time points. Different sub-network has different scales, shown in Table 1.

**Table 1.** The number of active proteins and interactions in dynamic PPI networks

Time point	1	2	3	4	5	6
The number of active proteins	1638	1742	1659	1444	1368	1211
The number of interaction	7574	8497	8262	6697	6250	5264
Time point	7	8	9	10	11	12
The number of active proteins	1221	1444	1756	1285	1410	1249
The number of interaction	5438	7109	8698	5999	6598	5306

Benchmark complex sets which are respectively derived from CYC2008, which contains 408 standard compounds, the maximum and minimum size of the cluster are 81 and 2, respectively.

### 5.2 Effect of IABC-DBSCAN Algorithm

To evaluate the clustering effect of IABC-DBSCAN, the algorithm was tested on every sub-networks generated from DIP dataset. We picked the average of running result for ten times. Table 2 shows the comparative results of two algorithms over 12 different time points in terms of  $f$ -measure. It is clear that the optimal parameter and clustering performance is different in the different sub-networks of IABC-DBSCAN. The results

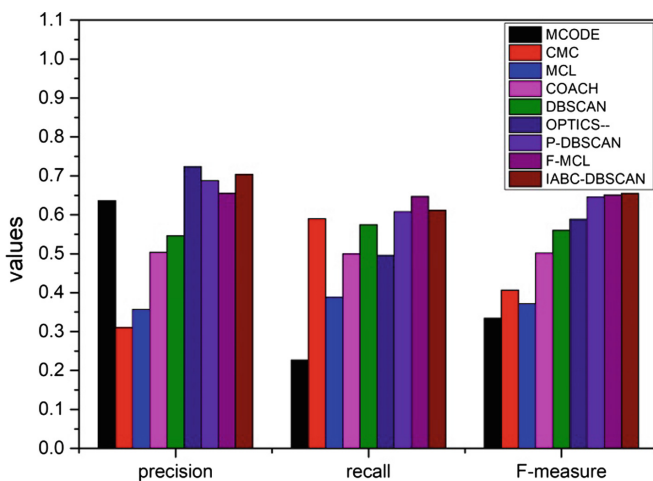


demonstrate that PPI network can be divided to numbers of dynamic sub-network based on genic data, which will improve the clustering performance of every sub-network.

**Table 2.** Optimal parameters and clustering performance in the 12 dynamic sub-networks

Time point	1	2	3	4	5	6
$\epsilon$	0.578	0.625	0.697	0.524	0.663	0.586
<i>MinPts</i>	3	3	3	3	3	3
<i>precision</i>	0.6795	0.7735	0.8675	0.6859	0.5644	0.7139
<i>recall</i>	0.4755	0.5887	0.4891	0.7023	0.5091	0.7354
<i>f - measure</i>	0.5595	0.6685	0.6255	0.6940	0.5353	0.7244
Time point	7	8	9	10	11	12
$\epsilon$	0.575	0.487	0.653	0.559	0.514	0.601
<i>MinPts</i>	3	3	3	3	3	3
<i>precision</i>	0.8012	0.6715	0.7523	0.7267	0.6218	0.7428
<i>recall</i>	0.6745	0.5841	0.6754	0.6021	0.6179	0.6453
<i>f - measure</i>	0.7324	0.6247	0.7117	0.6585	0.6198	0.7428

In order to compare IABC-DBSCAN with traditional clustering method MCODE [3], CMC [4], MCL [10], COACH [5], DBSCAN [7] and OPTICS [8], and clustering algorithm based on swarm intelligence optimization F-MCL [9] and P-DBSCAN [11]. All those algorithms are compared with each other on dynamic PPI dynamic weighted network using the CYC2008 gold standard. The performances of all clustering algorithms are reported in Fig. 1, which contains the precision, recall and *f-measure*. For the value of *f-measure*, P-DBSCAN and F-MCL is 15.34% and 75.195% higher than



**Fig. 1.** The performance of clustering algorithms

DBSCAN and MCL. The IABC-DBSCAN algorithm has the highest values in *F-measure*, which is 95.92%, 61.117%, 76.158%, 30.46%, 16.925%, 11.34%, 1.374% and 0.549% higher than MCODE, CMC, MCL, COACH, DBSCAN, OPTICS, P-DBSCAN and F-MCL algorithm, respectively.

To test further the performance of algorithm, Table 3 provides the basic information of the detection results for nine algorithms. For each algorithm, we have listed the number of clustering, the average size of protein complexes and the coverage ratio. From the Table 3, it can be seen that the number of clustering of IABC-DBSCAN is only lower than DBSCAN and P-DBSCAN, but the average size of function module in IABC-DBSCAN is much more close to the average size of gold standard. In addition, the reliability is measured based on the degree of nodes in DBSCAN, and the negative impact of false positives and false negatives on clustering result is not considered in P-DBSCAN. However, IABC-DBSCAN use the dynamic weighted network, TCSM method and ASS method to optimize the clustering process, which makes the interaction between proteins more reliable and the identified function module closer to gold standard. Therefore, it has much better clustering performance than the other algorithms.

**Table 3.** The clustering performance of algorithms

Algorithm	Type	The number of clustering	Average size	Coverage ratio
Standard clustering	–	408	4.71	1628
MCODE	Density	63	19	1032
CMC	Density	1263	4.39	2048
MCL	Stream	623	6.57	4096
COACH	Core-attachment	903	3.89	2133
DBSCAN	Density	492	6.26	1817
OPTICS	Density	107	5.9	630
P-DBSCAN	Swarm intelligence	642	4.98	1652
F-MCL	Swarm intelligence	1588	4.62	2808
ABC-DBSCAN	Swarm intelligence	639	4.95	1645

## 6 Conclusion

In this paper, the IABC-DBSCAN algorithm was obtained by modifying the ABC and DBSCAN algorithm to apply to dynamic weighted PPI network. In the clustering process, in order to overcome the effect of false positives, the GO annotations and genic expression data were used to construct the dynamic weighted PPI network; In order to overcome the defect of easily into local optimum in the density clustering algorithm,

the clustering algorithm was modified using TCSM method to search the optimal parameter; Furthermore, in order to improve the searching efficiency of optimal parameter, ASS method was used to solve the problem that the efficiency of searching optimal nectar source is not satisfied; Finally, the FD and TS strategy were used to optimize the performance of clustering result in the DBSCAN algorithm. The comparative results show that the IABC-DBSCAN algorithm for mining protein complexes is superior in terms of *precision*, *recall* and *f-measure*. In the future, we will keep trying to improve the accuracy of protein function module recognition and solve the problem of parameter setting in the DBSCAN algorithm completely.

## References

1. Ji, J., Zhang, A., Liu, C., et al.: Survey: functional module detection from protein-protein interaction networks. *IEEE Trans. Knowl. Data Eng.* **26**(2), 261–277 (2014)
2. Zhao, B., Wang, J., Li, M., et al.: A new method for predicting protein functions from dynamic weighted interactome networks. *IEEE Trans. Nanobiosci.* **15**(2), 131–139 (2016)
3. Bader, C., Hogue, C.: An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinform.* **4**(1), 2 (2003)
4. Liu, G., Wong, L., Chua, H.N.: Complex discovery from weighted PPI networks. *Bioinformatics* **25**(15), 1891–1897 (2009)
5. Leung, H.C., Xiang, Q., Yiu, S.M., et al.: Predicting protein complexes from PPI data: a core-attachment approach. *J. Comput. Biol.* **16**(2), 133–144 (2009)
6. Kessler, J., Andrushchenko, V., Kapitan, J., et al.: Insight into vibrational circular dichroism of proteins by density functional modeling. *Phys. Chem. Chem. Phys.* **20**(7), 4926–4935 (2018)
7. Ester, M., Kriegel, H.P., Xu, X.: A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: *International Conference on Knowledge Discovery and Data Mining*, pp. 226–231. AAAI Press (1996)
8. Ankerst, M., Breunig, M.M., Kriegel, H.P.: OPTICS: ordering points to identify the clustering structure. *ACM Sigmod Rec.* **28**(2), 49–60 (1999)
9. Lei, X., Wang, F., Wu, F.X., et al.: Protein complex identification through Markov clustering with firefly algorithm on dynamic protein–protein interaction networks. *Inf. Sci.* **329**(6), 303–316 (2016)
10. Lei, X., Ying, C., Wu, F.X., et al.: Clustering PPI data by combining FA and SHC method. *BMC Genom.* **16**(S3), S3 (2015)
11. Lei, X., Ding, Y., Wu, F.X.: Detecting protein complexes from DPINs by density based clustering with pigeon-inspired optimization algorithm. *Sci. China Inf. Sci.* **59**(7), 070103 (2016)
12. Singh, K., Sundar, S.: Artificial bee colony algorithm using problem-specific neighborhood strategies for the tree t-spanner problem. *Appl. Soft Comput.* **62**, 110–118 (2018)
13. Lei, X., Tian, J.: The information flow clustering model and algorithm based on the artificial bee colony mechanism of PPI network. *Chin. J. Comput.* **35**(1), 134–145 (2012)
14. Tian, J.F., Lei, X.J.: PPI network clustering based on artificial bee colony and breadth first traverse algorithm. *PR&AI* **25**(3), 481–490 (2012)

15. Wu, S., Lei, X., Tian, J.: An improved bacteria foraging optimization algorithm based on intuition fuzzy set for clustering PPI network. In: The 3rd International Conference on Quantitative Logic and Soft Computing, Xi'an, vol. 5, pp. 362–369, 12–15 May 2012
16. Karaboga, D., Akay, B.: A comparative study of artificial bee colony algorithm. *Appl. Math. Comput.* **214**(1), 108–132 (2009)
17. Amaya, A.F.F.: *Protein Interaction Network*. Springer, New York (2013)
18. Hu, S., Xiong, H., Zhao, B.: Construction of dynamic-weighted protein interactome network and its application. *Acta Autom. Sinica.* **41**(11), 1893–1900 (2015)
19. Letovsky, S., Kasif, S.: Predicting protein function from protein-protein interaction data: a probabilistic approach. *Bioinformatics* **19**(6), 197–204 (2003)
20. Li, P.: *Artificial plant optimization algorithm with different selection strategy*. Taiyuan University of Science and Technology (2014)
21. Lei, X., Ding, Y., Fujita, H., et al.: Identification of dynamic protein complexes based on fruit fly optimization algorithm. *Knowl. Based Syst.* **105**(C), 270–277 (2016)
22. Wang, J., Peng, X., Li, M., et al.: Construction and application of dynamic protein interaction network based on time course gene expression data. *Proteomics* **13**(2), 301–312 (2013)
23. Bihai, Z., Jianxin, W., Min, L., et al.: Prediction of essential proteins based on overlapping essential modules. *IEEE Trans. Nanobiosci.* **13**(4), 415–424 (2014)
24. Tu, B.P., Mcknight, S.L.: Logic of the yeast metabolic cycle: temporal compartmentalization of cellular processes. *Science* **310**(5751), 1152–1158 (2005)
25. Junzhong, J., Zhijun, L., Hongxin, L., et al.: An overview of research on functional module detection for protein-protein interaction networks. *Acta Autom. Sinica* **40**(4), 577–593 (2014)
26. Bihai, Z., Jianxin, W., Min, L., et al.: Detecting protein complexes based on uncertain graph model. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **11**(3), 486–497 (2014)
27. Ester, M.: A density-based algorithm for discovering clusters in large spatial databases with noise, pp. 226–231 (1996)
28. Enright, A.J., Dongen, S.V., Ouzounis, C.A.: An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Res.* **30**(7), 1575–1584 (2002)