Iraqi Journal of Science, 2023, Vol. 64, No. 3, pp: 1426-1441
DOI: 10.24996/ijs.2023.64.3.33

ISSN: 0067-2904

A Tri-Gene Ontology Migration Operator for Improving the Performance of Meta-heuristics in Complex Detection Problems

Isra H. Abdulateef [1], Dhia A. Jumaa Alzubaydi [3], Bara'a Ali Attea[2*]
[1] Department of Computer Science, College of Science, Al-Mustansiriyah University, Baghdad, Iraq.
[2] Department of Computer Science, College of Science, University of Baghdad, Baghdad, Iraq.
[3]Al-Rasheed University College, Baghdad, Iraq

Received: 29/1/2022 Accepted: 13/8/2022 Published: 30/3/2023

Abstract

Detecting protein complexes in protein-protein interaction (PPI) networks is a challenging problem in computational biology. To uncover a PPI network into a complex structure, different meta-heuristic algorithms have been proposed in the literature. Unfortunately, many of such methods, including evolutionary algorithms (EAs), are based solely on the topological information of the network rather than on biological information. Despite the effectiveness of EAs over heuristic methods, more inherent biological properties of proteins are rarely investigated and exploited in these approaches. In this paper, we proposed an EA with a new mutation operator for complex detection problems. The proposed mutation operator is formulated under four expressions depending on the type of gene sub-ontology. To demonstrate the performance of the proposed evolutionary based complex detection algorithm, the Saccharomyces Cerevisiae (yeast) PPI network is used in the evaluation. The results reveal that the proposed algorithm achieves more accurate complex structures than the counterpart heuristic algorithms and the canonical evolutionary algorithm based on the topological-aware mutation operator.

Keywords: Evolutionary algorithm; functional similarity; gene sub-ontology; protein-protein interaction network; semantic similarity.

مشغل ترحيل مبني على علم الوجود ثلاثي الجينات لتحسين أداء الاستدلال الفوقي لأكتشاف المركبات البروتينية

أسراء هيثم عبد اللطيف[1]، ضياء الزبيدي[3]،.براء علي عطية[2*]
[1] قسم الحاسبات ، كلية العلوم ، الجامعة المستنصرية ، بغداد ، العراق
[2] قسم الحاسبات ، كلية العلوم ، جامعة بغداد ، بغداد ، العراق
[3] كلية الرشيد الجامعة ، بغداد ، العراق

الخلاصه

يعد اكتشاف المركبات البروتينة في شبكات التفاعل البروتين–البروتين (PPI) مشكلة صعبة في علم الأحياء الحسابي. للكشف عن شبكة PPI في بنية معقدة ، تم اقتراح خوارزميات وصفية مختلفة في الأدبيات. لسوء

*Email: bara.a@sc.uobaghdad.edu.iq

الحظ ، فإن العديد من هذه الأساليب ، بما في ذلك الخوارزميات التطورية (EAs) ، تعتمد فقط على المعلومات الطوبولوجية للشبكة بدلاً من المعلومات البيولوجية. على الرغم من فعالية EAs على الطرق الاستكشافية ، نادرًا ما يتم التحقيق والاستفادة من الخصائص البيولوجية الأكثر جوهرية للبروتين في هذه الأساليب. في هذا البحث ، اقترحنا مشغل ترحيل لحل مشكلة اكتشاف المركبات البروتينية. تمت صياغة عامل الطفرة المقترح تحت أربعة تعبيرات اعتمادًا على نوع الجينات الأنطولوجية الفرعية. لإثبات أداء خوارزمية أكتشاف المركبات البروتينية المقترحة ، تم استعمال شبكة Saccharomyces Cerevisiae (الخميرة) PPI للتقييم. كشفت النتائج أن الخوارزمية المقترحة تحقق مركبات بروتينية أكثر دقة من الخوارزميات المضادة الموجهة والخوارزمية التطورية القانونية والتي تعتمد على مشغل ترحيل مبني على المعلومات الطوبولوجية فقط.

1. Introduction

Within one cell, multi-biological processes are carried out by proteins, which are grouped into complexes. These complexes work, even in the world of microorganisms (such as yeast), in a dense model [1, 2]. Protein complexes interact with each other or with proteins as a unit alone, and the interaction formulates a functional model that identifies cellular mechanisms. A Protein-Protein Interaction (PPI) network is a kind of biological network where proteins are nodes and the interactions between proteins are the network's edges. Consequently, detection of protein complexes and understanding of complete reconstruction of physical interactions within protein complexes will be very useful to get a clear idea about cellular organization, mechanisms regulating cell life, even therapeutic purposes, and more [2, 3].

Unlike heuristic or deterministic based complex detection algorithms, metaheuristics and evolutionary algorithms (EAs) are proved to be a sustainable alternative to solve NP-hard problems while accommodating their combinatorial explosion. For complex detection problems, Pizzuti and Rombo in 2014 [3] were the first to show that evolutionary-based complex detection methods are more robust than other state-of-the-art heuristic-based complex detection methods. Unfortunately, almost all the design of the main components of all these meta-heuristic algorithms is generally directed by several topological structures of the complexes. For example, Pizzuti and Rombo [3] expressed a canonical EA to detect protein complexes and showed the encouraging performance of EAs to outperform the counterpart heuristic methods. Unfortunately, the current effort to design evolutionary-based complex detection methods with gene ontology (GO) aware components is still lagging behind in the literature.

The key contribution of this paper is to design an evolutionary-based complex detection algorithm with a gene ontology mutation operator. The proposed mutation operator (the so-called migration operator) is formulated based on three different gene sub-ontology types: Molecular Function (MF), Cellular Component (CC), and Biological Process (BP) and their combinations. The remainder of this paper is organized as follows. A literature review and preliminary concepts are presented in the following two sections. This is followed by Section 4 introducing the proposed EA with the proposed migration operator. Four formulations are suggested for the proposed migration operator. The results and discussions are provided in Section 5, demonstrating that it is interesting enough to develop an EA with gene ontology-based components. Finally, conclusions and future directions are provided in Section 6.

2. Metaheuristic based complex detection algorithms: A review

As far as we know, developing GO-based evolutionary algorithms for the complex detection problem is still insufficiently explored in the literature. Only a few works have

examined the incorporation of GO information into the framework of evolutionary algorithms. For example, Mukhopadhyay et al. [4] and Bandyopadhyay et al. [5] applied a direct way of modeling GO semantic similarity in the optimization function, however, with the main aim of maximizing functional modules. In both [4] and [5], a multi-objective optimization problem is formulated reflecting a topologically based protein-cluster contribution and closeness centrality objectives. However, in [5], the proposed GO-based function is formulated with respect to the direct annotation and average of pairwise GO semantic similarity.

A multi-objective evolutionary based complex detection algorithm is proposed in [6]. The algorithm defines the problem as a multi-objective optimization model. Further, in [6, 7], a heuristic mutation operator (the so-called protein-complex attraction and repulsion operator) is also proposed to harness the strength of the proposed optimization model and other state-of-the-art single and multi-objective optimization models. Unlike the canonical mutation operator, more inherent topological properties at both the complex level and protein level are reflected by the proposed protein-complex attraction and repulsion operator.
However, the main interest of many metaheuristic-based complex detection algorithms is to formulate their components from topological structure only. For example, a topological-based mutation operator is proposed in [8, 9]. The basic idea of the proposed operator is to break up the coexistence of a pair of proteins according to their topological similarity. Their similarity will determine the existence of this pair within one complex or distinct complexes. The detection ability of several single and multi-objective topology-based optimization models is demonstrated using this operator.

3. PPI networks and complex graphs

A PPI network, N, is a random but finite complex graph of n nodes (proteins) and m edges (interactions). Usually in graph notations, we can describe N as a graph with cardinality n and volume m. The general expression of any network can be modeled as an undirected graph $G = (V, E)$ of a finite set of n vertices $V = (v_1, v_2, \dots, v_n)$ V and a finite set of m edges E $\subseteq$V×V.

When two nodes, v_1 and v_2, have an edge in E, then we can use the term "neighbor" to describe them. Overall, $G = (V, E)$ can be described by a symmetric binary square matrix, usually known as an adjacency matrix $\boldsymbol{A}$, where each entry (i, j) in the matrix is assigned to 1 (similarly $(j, i) = 1$) when v_1 and v_2 are neighbors, otherwise (i, j) is set to 0.
From the adjacency matrix, $\boldsymbol{A} = [i, j]^{n \times n}$, we can visualize the search space, Ω, for the network decomposition problem. The search space, Ω, contains all possible or candidate decomposition solutions of the matrix, $\boldsymbol{A}$ into square sub-matrices, $(A_1, A_2, \dots, A_K)$. In complex detection problem and for PPI network, N, each candidate solution renders a possible partitioning for the n proteins into K complexes $\mathcal{C} = (C_1, C_2, \dots, C_K)$. The main characteristic of complexes (or modules), even though there is no firm rule to define them, is that they can be mapped onto highly dense sub-matrices $(A_1 A_2, \dots, A_K)$.
Representation of conditional independence relations among variables and causal relationships could use directed graphs as a useful tool. A directed graph G consists of a set of vertices and a set of edges. When (X, Y) belongs to the edge set E, then there is an arrow pointing from Y to X (see Figure 1).

Figure 1: A directed graph with vertices V = {X, Y, Z} and edges E = {(Y, X), (Y, Z)}.

4. Gene Ontology

Ontology is the part of philosophy that specializes in studying existence and beginnings. It tries to build norms to define and characterize entities, their properties, events that may occur on them, processes that take place on them, and relationships in all aspect of the real world, taking into account all details and different domains [10, 11]. This was the main motivation behind Gene Ontology (GO). GO is a guide for describing gene and protein functions. Actually, the correlation of a gene product to a GO term is a GO annotation. The GO Consortium realized that the availability of supporting information besides these correlations is important, so each GO annotation constantly makes mention of the evidence that is based upon it. One widespread use is to derive commonalities in the location or function of genes that are over-or under-expressed [12]. The goal of GO is to create strict shared vocabularies and clarify its role across different organisms [12, 13]. These strict vocabularies are separated into three sub-ontologies:

1. *Molecular Function (MF)*: Gene products perform molecular-level activities. These activities at the molecular-level (such as catalysis or transport) are described by Molecular Function terms. These terms also represent entities, molecules, or complexes, to perform the actions. However, it is not defined where, when, or even under which conditions the action occurs. Molecular Function is usually correlated with an activity that is performed by one gene product (i.e., protein or RNA) [14]. In short, we can say that MF terms are terms that depict actions that they perform [15].

2. *Cellular Component (CC)*: Terms in this ontology depict the position of a gene product in the cell [15]. Gene product functions are performed in one of the following cellular structures: i) cellular compartments (*e.g.*, the *mitochondrion*), or ii) stable macromolecular complexes of which they are part (*e.g.*, the *ribosome*). Contrary to what is known about aspects of GO, cellular component classes represent a cellular anatomy, not processes [16].

3. *Biological Process (BP)*: Multiple molecular activities are needed to accomplish larger processes or "biological programs" which are described by the term "Biological Process (BP)". A simple example of a broad Biological Process ontology is *cell growth and maintenance.* Note that Biological Process is not equivalent to pathway [16]. Briefly speaking, Biological Process (BP) terms are terms that depict the broadest role that could be played [15].

Each GO sub-ontology is represented as a hierarchically structured graph (known as directed acyclic graphs (DAGs)). Consider the three DAGs in Figure 2, of three GO terms. The top left represents a CC term, the top right represents an MF term, and the bottom DAG represents a BP term. In each DAG, the terms (i.e., functional feature descriptions) are the nodes in these DAGs, and edges form relations between terms. Because of the hierarchically structured nature of GO, the annotated proteins can be compared in terms of the different terms in the ontology graph [12, 14, 17]. Mainly, relationships come in two types. The first type is sub-typing (or simple class-subclass) relations (i.e., "is a"); where term A "is a" term B means that A is a subclass of B. The second type is the partition (or partial ownership) relation (i.e., "part of"); where the term C "part of" D means that whenever C is present, it is always a part of D, but C need not always be present [14, 16]. In ontology graph, terms which

are semantically more similar are arranged close together than those away from each other [12]. A gene product (i.e., protein) is annotated with any set of terms from all or any of the three ontologies, depending on the available data [15].
The GO project is composed of two main parts: the biological aspects models (i.e., Gene Ontology itself) and the annotations. The annotation part links genes or gene products to terms from the Gene Ontology. Thus, all functions that a gene could have are explained in GO annotation. Table 1 presents ten different gene products (proteins) with their GO annotations in MF, CC, and BP. These proteins and their annotations in GO terms were taken from the Saccharomyces Genome Database (SGD) in the live version (URL: http://www.yeastgenome.org).

Table 1 explains that each protein could consist of one, two, or three types of ontologies. Additionally, each ontology can have a different number of GO terms. For example, the annotation of protein 'YNL317W' consists of three types of ontology (MF, BP, and CC). However, the annotation of protein 'YMR091C' has two types of ontology, while protein 'YLR312W' has no term in any ontology.

5. The proposed meta-heuristic with tri gene ontology migration operator

In this section, we introduce the proposed meta-heuristic-based complex detection algorithm. As evolutionary algorithms (EAs) are known to be the most dominant meta-heuristics, we will consider them as the search algorithm. The presentation includes the formulation of the main components of the proposed EA with the modularity as optimization model and the proposed tri-gene ontology migration operator. To improve the quality of the population evolved by EA, four formulations for the migration operator are formulated. The formulations are based on the three main gene ontology types and their combinations.

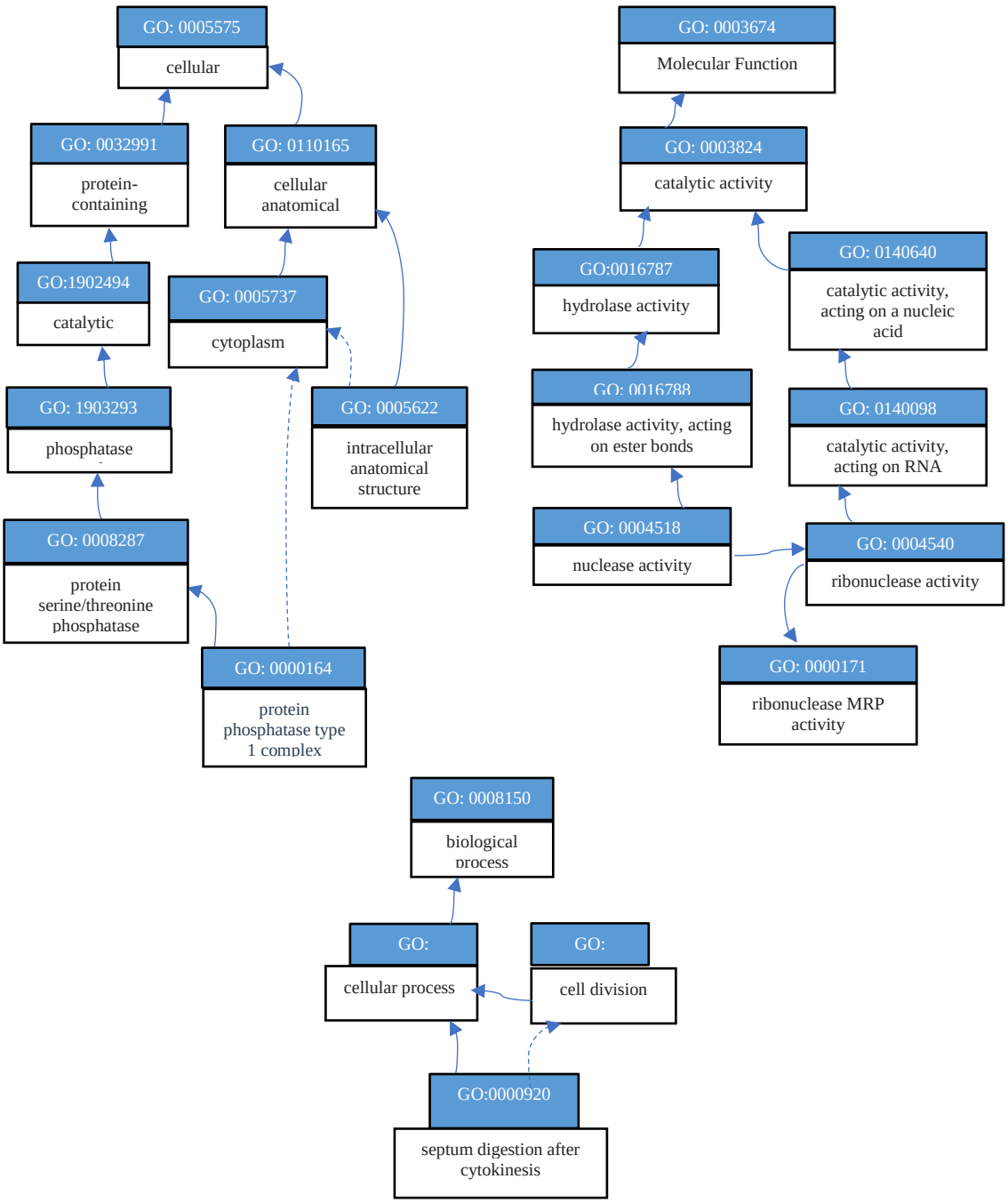


Figure 2: DAGs for three different GO terms. Top left: The DAG for a GO term of *CC* sub-ontology (protein phosphatase type 1 complex: 0000164). Top right: The DAG for a GO term of *MF* sub-ontology (ribonuclease MRP activity: 0000171). Bottom: The DAG for a GO term of *BP* sub-ontology (septum digestion after cytokinesis: 0000920).

Table 1: Annotation of ten different gene products (i.e. proteins) with their sub-ontology MF, BP, and/or CC terms. The annotation in GO terms is an updated version taken from Saccharomyces Genome Database (SGD) in the live version (URL: http://www.yeastgenome.org).

Protein	MF	CC	BP
YNL317W	[GO:0003723]	[GO:0005847, GO:0005634, GO:0005634, GO:0005847, GO:0005847, GO:0005847]	[GO:0006378, GO:0006378, GO:0098789, GO:0098789, GO:0006397, GO:0006378, GO:0006378]
YDL213C	[GO:0003723, GO:0019843, GO:0030515, GO:003676, GO:003676]	[GO:0030686, GO:0005730, GO:0030686, GO:0005730 , GO:0005634, GO:0005730]	[GO:0042274, GO:0042254, GO:0006364]
YDR381W	[GO:0005515, GO:0003723, GO:1990119, GO:0003723, GO:0003676, GO:0003676]	[GO:0005634, GO:0000346, GO:0005634]	[GO:0043462, GO:0006283, GO:0006406, GO:1902281]
'YLR221C	[GO:0003674]	[GO:0005730, GO:0005634, GO:00030687, GO:0005730, GO:0030687, GO:0030687]	[GO:0042254, GO:0000027, GO:0000027, GO:0000027]
YPR093C	[GO:0004842, GO:0046872]	[GO:0005634, GO:0005737, GO:0005737, GO:0005737, GO:0005634, GO:0005634]	[GO:0000921, GO:0016567, GO:0016567, GO:0045471]
YDR240C	[GO:0003723, GO:0003729, GO:0003729]	[GO:0000243, GO:0005685, GO:0071004, GO:0005634, GO:0005634, GO:0005681]	[3 GO:00098, GO:000398, GO:0006397, GO:0008380]
YDR073W	[]	[GO:0005634, GO:0005829, GO:0016514, GO:0016514, GO:0005634, GO:005634]	[GO:0045944]
YMR091C	[]	[GO:0016586, GO:0016586, GO:0016586, GO:0005634, GO:0005634, GO:0005634, GO:0005634]	[GO:0006337, GO:0006368, GO:0043044, GO:0006325]
YLR312W	[]	[]	[]
YFR024C	[]	[]	[]

5.1 Individual representation and search space size

The formulations of all components of the proposed EA are expressed to meet the complex detection problem. First, the locus-based individual representation is used to encode each partitioning solution. In locus-based representation, here, each individual chromosome P in the population $\mathbb{P} = \{P_1, P_2, \dots, P_{pop-size}\}$ is defined as a collection of n protein–protein fitting in the same complex. Each chromosome, P_i, has n entities, i.e. , $P_i = \{P_{i,1}, P_{i,2}, \dots, P_{i,n}\}$, where the locus, j, points to the protein, j, and the allele, $P_{i,j}$, points to one of the neighboring proteins of j within which protein, j, should fit in the complex formation. For this representation, one can compute the total search space size, $|\Omega|$, that should be uncovered well by the proposed EA to reach an acceptable partitioning solution. Consider that $|d_i|$ is the number of neighbor proteins to protein, i. Then for n proteins, the size of the search space $|\Omega| = d_1 \times d_2 \times \dots \times d_n$.

The decoding function δ of an individual chromosome sketches different complex structures, and thus different number of complexes, for the network. To uncover the promising area of the search space, the EA starts with a small subset of $|\Omega|$, called an initial random population $\mathbb{P}$ and evolves it iteratively towards a better and better population.

5.2 Modularity as a fitness function

Good solutions, and thus good areas of the search space, are quantitatively evaluated using the modularity function, Q. Modularity Q (Eq. 1), positively treats a partitioning solution $\mathcal{C}$ with a higher fraction of intra-neighboring within the K complexes of $\mathcal{C}$ and, simultaneously, with sparse inter-neighboring between these K complexes [18]. The left term in Eq.1 (i.e. $(\frac{m_{C_i}}{|m|})$), expresses the ratio of the intra-volume of the complex C_i (which is noted as m_{C_i}) to the total volume of the network ($|m|$). In other words, it biases towards a collection of densely intra-connected modules (or complexes). On the other hand, the right term $(\frac{\sum_{v \in C_i} |d_v|}{2|m|})$ expresses that the expected value of the same connection density in $\mathcal{C}$ with the same community structure, but fall at random, between the proteins should be small.

$$Q(\mathcal{C}) = \sum_{i=1}^{K} \left[\frac{m_{C_i}}{|m|} - \left(\frac{\sum_{v \in C_i} |d_v|}{2|m|} \right)^2 \right] \quad (1)$$

5.3 Evolution operators

Once the chromosome solutions are evaluated in terms of modularity, another set of $pop - size$, but better, solutions is generated by iterative composition of three the main evolution operators. These are selection (s), crossover (Θ_c), and mutation. A set of parent solutions is selected, using tournament selection, based on their modularity values. Then, a uniform crossover operator, with relatively high crossover probability, p_c, is performed to perform a locus-wise crossing between the alleles of each pair of parent solutions. The uniform crossover uniformly mixes the alleles of the two parents to generate a child solution. Thus, the child solution would normally contain equivalent traits from the two parents.

5.4 The proposed Tri GO-based migration operator

For the mutation operator, four formulations are suggested. The essential rule of the proposed mutation operator is based on the migration operator of [19, 20]. This topological-based operator was proved in [19, 20] to harness the performance of the evolutionary based community detection algorithms. The migration operator, Ψ, works on allele values (i.e. protein neighboring) and alters, with a specified, normally low migration probability p_m, its neighbor. In other words, the proposed operator would change the selected protein from its home complex to a new one if the protein has denser connections with the proteins of the second complex than with the home complex.

In this paper, we relaxed this migration operator to meet the requirement for the gene semantic similarity of the migrated proteins (i.e., the migrated gene products). By this means, we introduce four types of gene ontology-based migration operators. The first three formulations are based on the three types of gene sub-ontology, i.e., MF, CC, and BP. In each type, the migration operator alters the complex-belonging of a mutated protein if it has more semantic similarity (in terms of its sub-ontology gene terms) with the proteins of the destination complex than with the source or home complex. The fourth formulation is then based on the combination of the semantic similarity of the three types.

Let us consider an individual solution $P_i = \{P_{i,1}, P_{i,2}, \dots, P_{i,n}\}$ corresponding to a candidate complex structure $\mathcal{C} = \{C_1, \dots, C_K\}$ with K complexes. Let protein j that corresponds to locus j being positioned in complex k, where $1 \leq k \leq K$. Then, protein j can be represented as a gene product with $Slim_j$, such that this slim combines the GO terms in the three sub-ontologies (Eq. 2). Further, protein j inside complex k has a gene product similarity, in terms of MF terms, CC terms, BP terms, and a combination of the three sub-ontology terms with

other proteins in the same complex k as expressed in Eq. 3, 4, 5, and 6, respectively. Note that the similarity of the combination of the three sub-ontology terms is called functional similarity (FS).

$$Slim_j = MF_j \cup CC_j \cup BP_j \tag{2}$$

$$MF(j, C_k) = \sum_{v \in C_k} \frac{|MF_j \cap MF_v|}{|MF_j \cup MF_v|} \tag{3}$$

$$CC(j, C_k) = \sum_{v \in C_k} \frac{|CC_j \cap CC_v|}{|CC_j \cup CC_v|} \tag{4}$$

$$BP(j, C_k) = \sum_{v \in C_k} \frac{|BP_j \cap BP_v|}{|BP_j \cup BP_v|} \tag{5}$$

$$FS(j, C_k) = \sum_{v \in C_k} \frac{|Slim_j \cap Slim_v|}{|Slim_j \cup Slim_v|} \tag{6}$$

Note that Jaccard similarity is computed for the GO terms of the pair of proteins j and each protein v in C_k. For example, consider the two first proteins (YNL317W and YDL213C) in Table 1 with their GO terms. By their sub-ontologies terms, we have Jaccard similarity equals to 0.2500, 0.2500, 0, and 0.1429 for, respectively, MF, CC, BP and FS.
The proposed migration operators, then, moves j from its home complex to another complex k', $1 \leq k' \leq K$ and $k' \neq k$ where protein j could maintain, in complex k', the maximum semantic similarity (as in Eq. 6, 7, and 8) and the maximum functional similarity as in Eq. 9. Note that when there is more than one complex to accept protein j with equal similarity value, then Ψ randomly selects any one of these destination complexes.

$$\Psi_{MF}(j \in C_k, p_m) = \max_{C_{k'} \in \mathcal{C}} MF(j, C_{k'}) \tag{7}$$

$$\Psi_{CC}(j \in C_k, p_m) = \max_{C_{k'} \in \mathcal{C}} CC(j, C_{k'}) \tag{8}$$

$$\Psi_{BP}(j \in C_k, p_m) = \max_{C_{k'} \in \mathcal{C}} BP(j, C_{k'}) \tag{9}$$

$$\Psi_{FS}(j \in C_k, p_m) = \max_{C_{k'} \in \mathcal{C}} FS(j, C_{k'}) \tag{10}$$

5.5 Algorithm layout

Algorithm 1 outlines the main steps of the proposed algorithm. In the algorithm, each individual solution is evolved via three main operators. The parent selection operator s selects the parents' population. Afterwards, both recombination and mutation operators are applied to generate modified individuals.

Algorithm 1: GO-based EA

```
t ← 0;
initialize population ℙ(t) ← (P_1, P_2, …, P_{pop−size});
for i ← 1 to pop − size do
    evaluate Q(P_i(t));
End
while (ι(ℙ(t)) ≠ true) do
    for i ← 1 to pop − size do
```

$P_{i,1}(t) \leftarrow s(G_i(t))$; // select parent 1 for P_i
$P_{i,2}(t) \leftarrow s(G_i(t))$; // select parent 2 for P_i
$P_i(t) \leftarrow \Theta_c(P_{i,1}(t), P_{i,2}(t), p_c)$
$P_i(t) \leftarrow \Psi(P_i(t), p_m)$; // either Ψ_{MF}, Ψ_{CC}, Ψ_{BP}, or Ψ_{FS}
evaluate $Q(P_i(t))$;
end
$t \leftarrow t + 1$;
end
return $P^*(t)$ // the best solution in terms of Q

6. Results and discussions

The filtered version of the Yeast Saccharomyces cerevisiae PPI network [21] is proven to be a highly effective PPI network for modeling organisms for mammalian biological functions and diseases. It contains 4687 interactions for 990 proteins. The MF, CC, and BP sub-ontology terms assigned to these 990 proteins are taken from the Saccharomyces Genome Database (SGD). They are annotated with a total of 541 MF, 452 CC, and 1245 BP. To validate the quality of the predicted complexes, a benchmark gold standard complex set drawn from the Munich Information Center for Protein Sequence (MIPS) catalog is used in the experiments. This benchmark contains 859 proteins partitioned into 81 protein complexes. The common measures of recall, precision, and cumulative F measure at both complex and protein levels are used in the evaluation. The detected set of complexes $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$ obtained from the best solution of each of the tested algorithms is compared with the standard true complexes $\mathcal{C}^* = \{C_1^*, C_2^*, \dots, C_{K^*}^*\}$ obtained from the MIPS. A detected complex C_i in the solution $\mathcal{C}$ overlaps, in terms of protein, a true complex C_j^* with an overlapping score (OS) (Eq. 10). Complex C_i *matches* the true complex C_j^* if OS is equal or larger than a specified threshold, σ_{OS}.

$$OS(C_i, C_j^*) = \frac{|C_i \cap C_j^*|^2}{|C_i||C_j^*|} \tag{11}$$

where $|\cdot|$ is the number of proteins common to both C_i and C_j^*.

$$match(C_i, C_j^*) = \begin{cases} 1 \; if \; OS(C_i, C_j^*) \geq \sigma_{OS} \\ 0 \; otherwise \end{cases} \tag{12}$$

$$Recall = \frac{|C_i^* : C_i^* \in \mathcal{C}^* \wedge \quad \exists C_j \in \mathcal{C} \rightarrow match(C_i^*, C_j)|}{K^*} \tag{13}$$

$$Precision = \frac{\left|C_i : C_i \in \mathcal{C} \wedge \exists C_j^* \in \mathcal{C}^* \rightarrow match\left(C_i, C_j^*\right)\right|}{K_{\mathcal{C}}} \tag{14}$$

$$F\text{-}measure = \frac{2 * recall * precision}{recall + precision} \tag{15}$$

$$Recall_N = \frac{\sum_{i=1}^{K^*} |m_i|}{\sum_{i=1}^{K^*} |C_i^*|} \tag{16}$$

where $|m_i| = max_{C_j \in \mathcal{C}} |match(C_i^*, C_j)$.

$$Precision_N = \frac{\sum_{i=1}^{K_{\mathcal{C}}} |m_i|}{\sum_{i=1}^{K_{\mathcal{C}}} |C_i|} \quad (17)$$

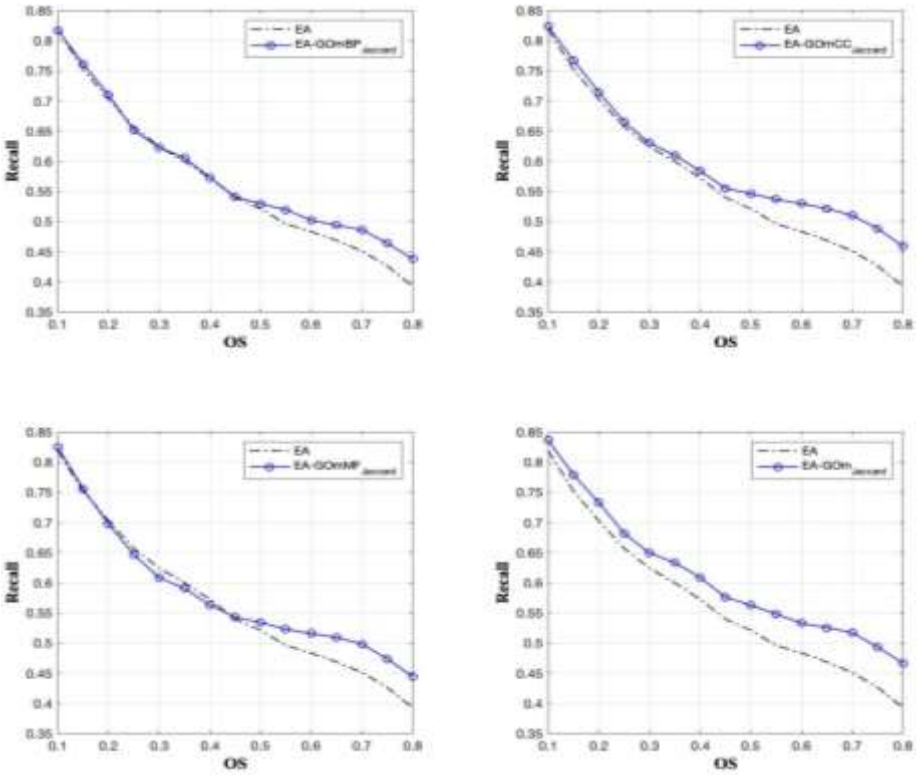
where $|m_i| = max_{C_j^* \in \mathcal{C}^*} |match(C_i, C_j^*)$.

$$Fmeasure_N = \frac{2 * recall_N * precision_N}{recall_N + precision_N} \quad (18)$$

Figures 3-5 depict the performance of the EA with the proposed migration operators based on semantic similarities (EA_{GOmMF}, EA_{GOmCC}, and EA_{GOmBP}) and functional similarity based on Jaccard ($EA_{GOmJaccard}$) against the canonical EA (EA) for an average of 30 different runs. Here, all algorithms were tested while setting their parameters to one common setting. Population size was set to 100. The evolutionary process was stopped at a maximum number of 100 generations. The Probability of uniform crossover was $p_c = 0.6$, and the probability of the canonical mutation operator and the proposed heuristic GO-based migration operators were $p_m = 0.2$.

The results show the positive impact of exploiting the gene ontology information in the formulation of the mutation operator at both the complex level and protein level. This can be turned back to the additional information injected into the algorithm. Further, the results clarify the additional improvement capabilities introduced by injecting the functional information of the three ontology types (i.e., $EA_{GOmJaccard}$).

Additional results are also provided in Table 2. The results report the performance of the proposed evolutionary algorithm against some of the well-known heuristic methods and the canonical EA. These are Molecular Complex Detection (MCODE) [22], Restricted Neighborhood Search Clustering (RNSC) [23], Clique Percolation Method (CPM) [24], link clustering (LC) [25], Markov Cluster Algorithm (MCL) [26], Overlapping Cluster Generator (OCG) [27], Extended Link Clustering method (ELC) [28], and network decomposition for overlapping community detection algorithm (NDOCD) [29].



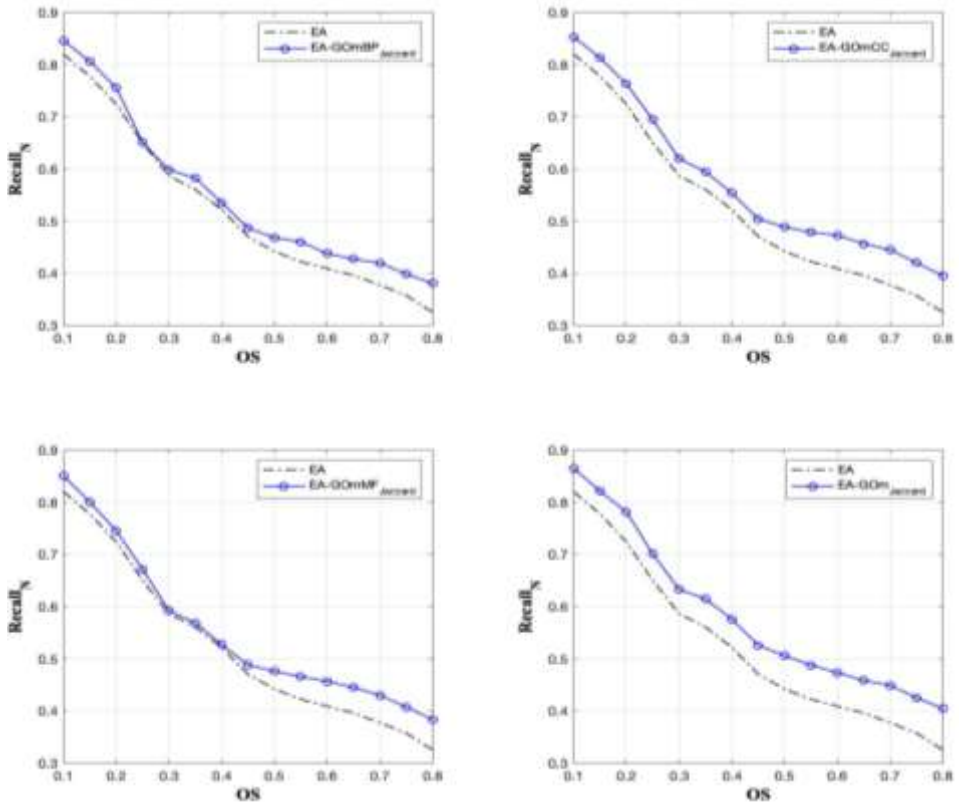
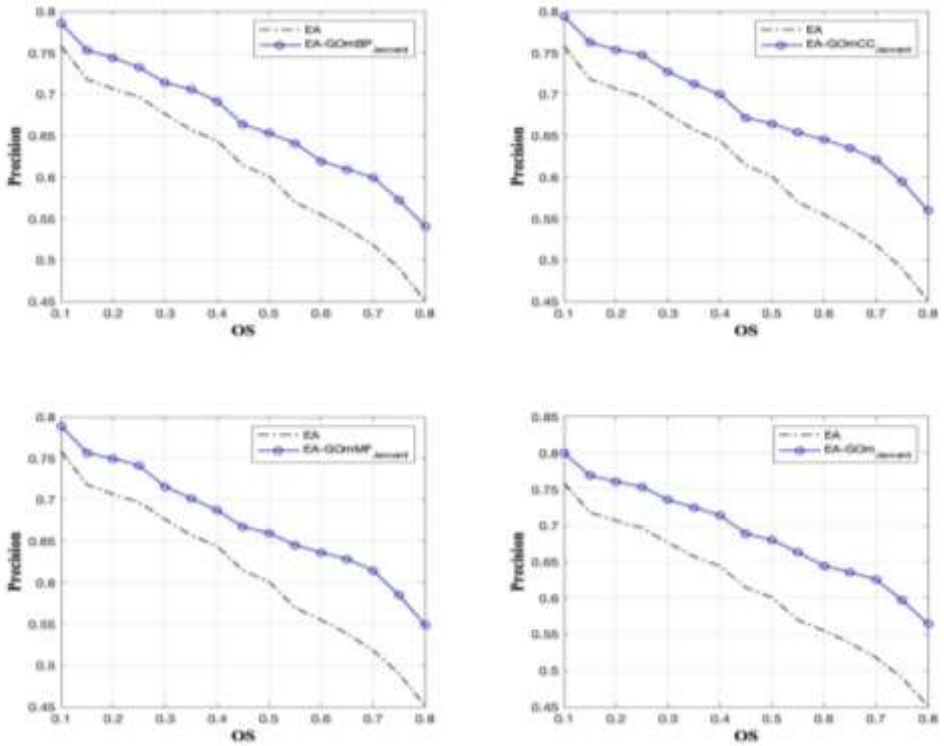


Figure 3: Performance comparison at both complex level (top) and protein level (bottom) in terms of Recall (top) and Recall_{N} (bottom) for an average of 30 different runs of the proposed EA_{GOmMF}, EA_{GOmCC}, EA_{GOmBP} and $EA_{GOmJaccard}$ against the EA.



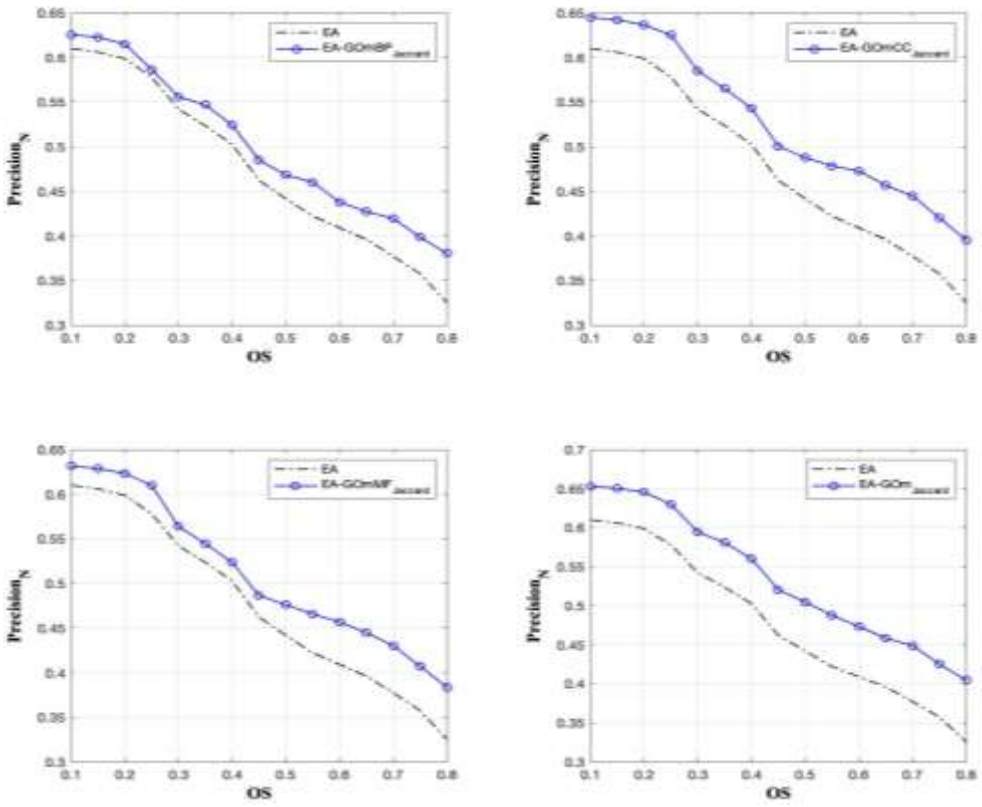
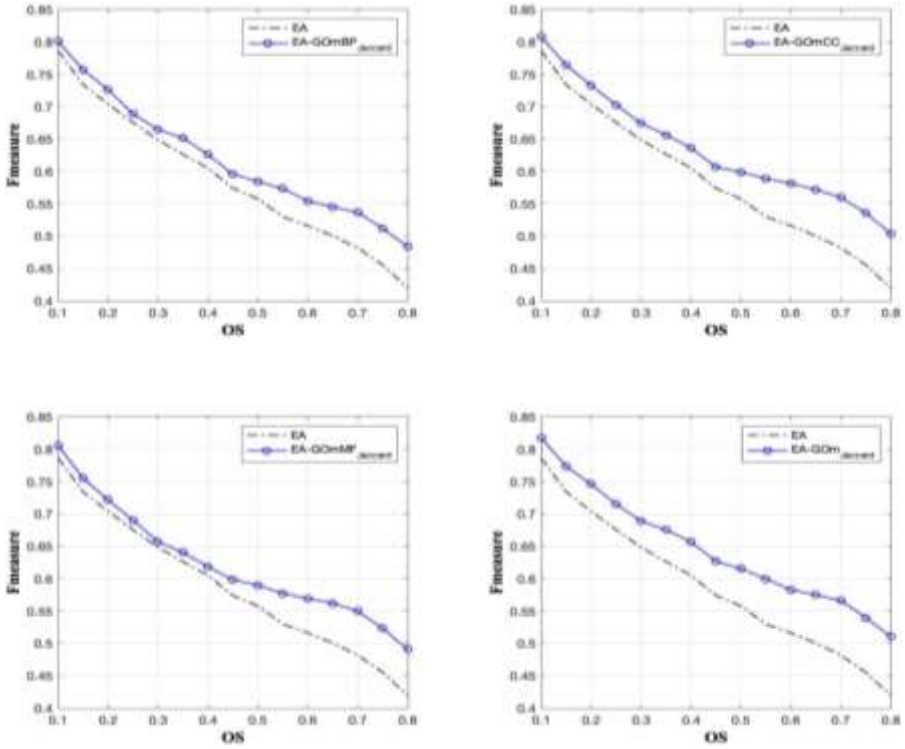


Figure 4: Performance comparison at both complex level (top) and protein level (bottom) in terms of Precision (top) and $\text{Precision}_{\text{N}}$ (bottom) for an average of 30 different runs of the proposed EA_{GOmMF}, EA_{GOmCC}, EA_{GOmBP} and $EA_{GOmJaccard}$ against the EA.



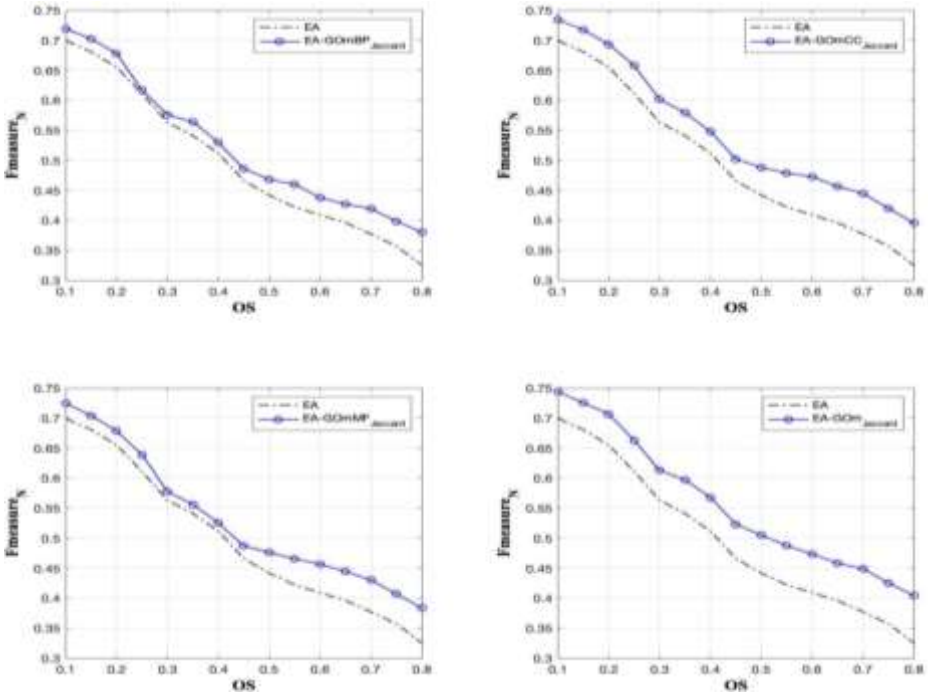


Figure 5: Performance comparison at both complex level (top) and protein level (bottom) in terms of Recall (top) and $\text{Fmeasure}_{\text{N}}$ (bottom) for an average of 30 different runs of the proposed EA_{GOmMF}, EA_{GOmCC}, EA_{GOmBP} and $EA_{GOmJaccard}$ against the EA.

The results clearly reflect the positive investment of the GO information and the functional domain to improve the detection ability of the single objective evolutionary algorithm (EA_{GOm}). In all metrics, $EA_{GOmJaccard}$ outperforms the canonical EA. Further, in $Precision$ and $F\text{-}measure$, the proposed GO based evolutionary algorithm outperforms all the heuristic based complex detection algorithms. For $Recall$, on the other hand, EA_{GOm} performs better than almost all the heuristic methods with exception to the performance of RNSC, MCL, and OCG. These algorithms, however, provide inferior solutions in terms of $Precision$, and $F\text{-}measure$.

Table 2: Comparison of performance in terms of $Recall$, $Precision$, and $F\text{-}measure$ where overlapping score with $OS = 0.2$. The best result in each validation metric is provided in bold.

Algorithm	***Recall***	***Precision***	***F-measure***
MCODE	0.6700	0.6250	0.6467
RNSC	**0.8490**	0.2650	0.4039
CPM	0.5850	0.6170	0.6006
LC	0.4950	0.0410	0.0757
MCL	0.8230	0.5390	0.6514
OCG	0.8380	0.6150	0.7094
ELC	0.5910	0.6479	0.6181
NDOCD	0.7830	0.7000	0.7392
EA	0.7050	0.7100	0.7050
$\boldsymbol{EA_{GOmJaccard}}$	0.7490	**0.7750**	**0.7550**

7. Conclusions

Complex detection is proved to be an NP-hard combinatorial optimization problem where meta-heuristic algorithms are proved to be beneficial over heuristic based approaches. One of the main contributions of bioinformatics to molecular biology is the introduction of ontologies for genome annotation. However, we found a lack of interest in the literature in designing an effective meta-heuristic algorithm. In this paper, a gene ontology based migration operator is proposed to further fine tune the EA-based generated complex structure. Four types of migration operators are formulated. The formulations are based on the three different types of gene sub-ontology (MF, CC, and BP) and their combinations. The results report a positive argument in favor of the proposed formulations against the canonical EA. Further improvement can be achieved while examining other EA components, e.g., objective function and crossover operator.

References

[1] J. Zahiri, A.Emamjomeh , S.Bagheri , A.Ivazeh , G.Mahdevar , H.S. Tehrani, M.Mirzaie, B.A. Fakheri and M.Mohammad-Noori, , "Protein complex prediction: A survey," *Genomics*, vol. 112, no. 1, pp.174-183, 2020.

[2] S. Srihari and H. W. Leong, "A survey of computational methods for protein complex prediction from protein interaction networks," *Journal of bioinformatics and computational biology,* vol. 11, no. 02, pp. 1230002, 2013.

[3] C. Pizzuti and S. E. Rombo, "Algorithms and tools for protein–protein interaction networks clustering, with a special focus on population-based stochastic methods," *Bioinformatics,* vol. 30, no. 10, pp. 1343-1352, 2014.

[4] A. Mukhopadhyay, S. Ray, and M. De, "Detecting protein complexes in a PPI network: a gene ontology based multi-objective evolutionary approach," *Molecular BioSystems,* vol. 8, no. 11, pp. 3036-3048, 2012.

[5] S. Bandyopadhyay, S. Ray, A. Mukhopadhyay, and U. Maulik, "A multiobjective approach for identifying protein complexes and studying their association in multiple disorders," *Algorithms for Molecular Biology,* vol. 10, no. 1, pp. 1-15, 2015.

[6] B. a. A. Attea and Q. Z. Abdullah, "Improving the performance of evolutionary-based complex detection models in protein–protein interaction networks," *Soft Computing,* vol. 22, no. 11, pp. 3721-3744, 2018.

[7] Q.Z. Abdullah, B.A. Attea, "Aheuristic strategy for improving the performance of evolutionary based complex detection in protein-protein interaction networks," *Iraqi Journal of Science* vol.57, no. 4A, pp. 2513–2528, 2016.

[8] A. H. Abdulateef, A. A. Bara'a, A. N. Rashid, and M. Al-Ani, "A new evolutionary algorithm with locally assisted heuristic for complex detection in protein interaction networks," *Applied Soft Computing,* vol. 73, pp. 1004-1025, 2018.

[9] A. H. Abdulateef, A. A. Bara'a, and A. N. Rashid, "Heuristic modularity for complex identification

[10] D.P. Hill, B . Smith, M.S. McAndrews-Hill, J.A. Blake. "Gene Ontology annotations: what they mean and where they come from," *BMC bioinformatics*, vol. 9, no. 5, pp. 1-9, 2008 BioMed Central.

[11] D.Faria, C. Pesquita , F.M. Couto, and A.O. Falcão, "GOclasses: molecular function as viewed by proteins". *The 12th Annual Bio-Ontologies*. 2009.

[12] C. Dessimoz, and N. Škunca , "*The gene ontology handbook*," Springer Nature, 2017, Ch2, Ch3.

[13] C. Pesquita, "Improving semantic similarity for proteins based on the gene ontology," M.S. thesis, Department of Informatics, University of Lisbon, Faculty of Sciences, Portugal 2007.

[14] P. Pinoli, D. Chicco, and M. Masseroli, "Computational algorithms to predict Gene Ontology annotations," *BMC bioinformatics*, vol. 16, no.6, pp.1-15, 2015

[15] M.B. Arnaud, M.C. Costanzo, P. Shah, M.S Skrzypek and G. Sherlock, , "Gene Ontology and the annotation of pathogen genomes: the case of Candida albicans," *Trends in microbiology*, vol.17, no. 7, pp.295-303, 2009.

[16] J.Z. Wang, Z. Du, R. Payattakool, P.S. Yu and C.F. Chen, "A new method to measure the semantic similarity of GO terms," *Bioinformatics*, vol. 23, no.10, pp.1274-1281, 2007.
[17] M. Kejriwal, C.A. Knoblock and P. Szekely, "*Knowledge Graphs: Fundamentals, Techniques, and Applications,"* MIT Press, 2021.
[18] Girvan, M. and Newman, M.E., "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp.7821-7826, 2002.
[19] B.A. Attea, W. A. Hariz and M. F. Abdulhalim, "Improving the performance of evolutionary multi-objective co-clustering models for community detection in complex social networks," *Swarm and Evolutionary Computation*, vol. 26, pp. 137-156, 2016.
[20] B.A. Attea, W. A. Hariz, "A Heuristic Multi-objective Community Detection Algorithm for Complex Social Networks," *Iraqi Journal of Science*, vol. 56 , 3533-3545, 2015.
[21] N. Zaki, J. Berengueres, and D. Efimov, "Detection of protein complexes using a protein ranking algorithm," *Proteins: Structure, Function, and Bioinformatics*, vol. 80, no.10, pp. 2459-2468, 2012.
[22] G.D. Bader, C.W. Hogue, "An automated method for finding molecular complexes in large protein interaction networks," *BMC bioinformatics* , vol.4, pp.1-27, 2003.
[23] A.D. King, N. Pržulj, I. Jurisica, "Protein complex prediction via cost-based clustering," *Bioinformatics*, vol.20, pp. 3013–3020, 2004.
[24] G. Palla, I. Derényi, I. Farkas, T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society." *Nature*, vol.435, pp.814–818, 2005.
[25] Y.Y. Ahn, J.P. Bagrow, S. Lehmann, "Link communities reveal multiscale complexity in networks," *Nature*, vol.466, pp.761–764, 2010.
[26] S. Ray, A. Hossain, U. Maulik, "Disease associated protein complex detection: a multi objective evolutionary approach," *International conference on microelectronics, computing and communications (MicroCom)*, IEEE. pp. 1–6, 2016
[27] E. Becker, B. Robisson, C.E. Chapple, A. Guénoche, C. Brun, "Multifunctional proteins revealed by overlapping clustering in protein interaction network," *Bioinformatics*, vol.28, pp.84–90, 2012.
[28] L. Huang, G. Wang, Y. Wang, E. Blanzieri, C. Su, "Link clustering with extended link similarity and eq evaluation division," *PLoS ONE* vol.8, No.6 e66005, 2013. https://doi.org /10.1371/journal.pone.0066005PloS one 8, e66005.
[29] Z. Ding, X. Zhang, D. Sun, B. Luo, "Overlapping community detection based on network decomposition," *Scientific reports*, vol.6, pp.1–11, 2016.