DOI: 10.24996/ijs.2024.65.3.42

ISSN: 0067-2904

Biological versus Topological Domains in Improving the Reliability of Evolutionary-Based Protein Complex Detection Algorithms

Isra H. Abdulateef[1,2*], Bara'a Ali Attea[2], and Dhia A. Alzubaydi[1,3]

[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] *Uruk Private University*

Received: 28/10/2022 Accepted: 14/4/2023 Published: 30/3/2024

Abstract

By definition, the detection of protein complexes that form protein-protein interaction networks (PPINs) is an NP-hard problem. Evolutionary algorithms (EAs), as global search methods, are proven in the literature to be more successful than greedy methods in detecting protein complexes. However, the design of most of these EA-based approaches relies on the topological information of the proteins in the PPIN. Biological information, as a key resource for molecular profiles, on the other hand, acquired a little interest in the design of the components in these EA-based methods. The main aim of this paper is to redesign two operators in the EA based on the functional domain rather than the graph topological domain. The perturbation mechanism of both crossover and mutation operators is designed based on the direct gene ontology annotations and Jaccard similarity coefficients for the proteins. The results on yeast Saccharomyces cerevisiae PPIN provide a useful perspective that the functional domain of the proteins, as compared with the topological domain, is more consistent with the true information reported in the Munich Information Center for Protein Sequence (MIPS) catalog. The evaluation at both complex and protein levels reveals that feeding the components of the EA with biological information will imply more accurate complex structures, whereas topological information may mislead the algorithm towards a faulty structure.

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

المجال البيولوجي مقابل المجال الطوبولوجي في تحسين موثوقية خوارزميات اكتشاف المركبات البروتينية القائمة على التطور

أسراء هيثم عبد اللطيف[1و2*]، براء علي عطية[2]، ضياء الزبيدي[3,1]

[1]قسم علوم الحاسوب ، كلية العلوم ، الجامعة المستنصرية ، بغداد ، العراق
[2]قسم علوم الحاسوب ، كلية العلوم ، جامعة بغداد ، بغداد ، العراق
[3]جامعة اوروك الاهلية

[*] Email: isra.h@sc.uobaghdad.edu.iq

الخلاصة:

حسب التعريف، فإن اكتشاف المركبات البروتينية والتي تشكل شبكات تفاعل البروتين البروتين (PPINs) تعد مشكلة من نوع NP–hard. وكما أثبتت الأدبيات ذات العلاقة، أن الخوارزميات التطورية (EAs)، كخوارزميات بحث شاملة، تعتبر أكثر نجاحا من الخوارزميات المحلية في اكتشاف المركبات البروتينية. ومع ذلك، فإن تصميم معظم الخوارزميات القائمة على مفهوم ال EA يعتمد على المعلومات الطوبولوجية للبروتينات في PPIN. ومن ناحية أخرى، اكتسبت المعلومات البيولوجية، والتي تعد كمورد رئيسي للملفات الجزيئية، القليل من الاهتمام بتصميم هكذا خوارزميات. الهدف الرئيسي من هذا البحث هو أعادة صياغة جزئين من مكونات خوارزمية ال EA باعتماد المعلومات البيولوجية بدلا من المعلومات الطوبولوجية. تم تعديل آلية التغيير في عملية المزج وعملية الطفرة باعتماد مجال وظيفي يعتمد على التعليقات التوضيحية المباشرة لعلم الوجود الجيني ومعامل تشابه Jaccard للبروتينات. وفرت النتائج على شبكات الخمائر Saccharomyces cerevisiae PPIN بأن المجال الوظيفي للبروتينات، مقارنة بالمجال الطوبولوجي، أكثر اتساقًا مع المعلومات الحقيقية الواردة في كتالوج مركز معلومات ميونيخ لتسلسل البروتين (MIPS).حيث يكشف تقييم النتائج على مستوى المركب ومستوى البروتين بأن تغذية مكونات EA بمعلومات بيولوجية ستعني ضمنيًا الوصول الى مركبات بروتينية ذات هيكلية أكثر دقة، في حين أن المعلومات الطوبولوجية قد تضلل الخوارزمية نحو هيكلية مركبات خاطئة.

1. Introduction

Thanks to the genomics revolution, we have witnessed, over the years, the birth of new protein-coding genes and RNA genes with novel functions in all organisms. However, some of these functional proteins, such as those in Severe Acute Respiratory Syndrome (SARS) and Coronavirus Disease 19 (COVID-19), have very harmful effects. While proteins rarely act alone, they team up into networks of complexes as described as biological functions at both cellular and systemic levels in all living organisms. In each living organism, the complete map of protein interactions is known as the interactome. Large protein-protein interaction networks (PPINs) are naturally complex and form highly inhomogeneous scale-free architectures in which a few highly connected proteins play a central role in mediating interactions among numerous, less connected proteins. Thus, the detection of protein complexes and understanding of the 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 [1].

A variety of topology-based complex augmentation/division, and cluster refinement approaches have been proposed in the literature. However, all of these approaches can be defined in the context of the topological information of n proteins being represented by the domain for both the heuristic rule and the heuristic operator. As protein complexes persist within specific topological characteristics, all the proposed heuristic rules and operators in this category are geared toward satisfying some of these topology-based objectives. Different studies acquired and developed different levels of topology-based relatedness or cost functions. These functions are used to iteratively evaluate how the generated solution should improve. For example, the cost function is based on the numbers of intra-cluster and inter-cluster connections in [2], local neighborhood density, and partition density in [3], [4], [5], and [6]. In these methods, scholars focused on a narrow aspect of the heuristic framework to only underscore the merits of topological characteristics to generate interconnected sub-graphs. Thus, different speculations of the adjacency matrix identify different levels of interconnected complexes.

Unfortunately, like many real-world optimization problems, the computational complexity of complex detection problems falls into the category of non-deterministic polynomial time hard (NP-hard) problems. For such a combinatorial optimization problem with n parameters, an exhaustive greedy search for the optimal solution becomes computationally prohibitive, particularly when n is large. Evolutionary algorithms (EAs) have been proven to be a viable and often superior alternative to greedy in solving NP-hard problems while accommodating their combinatorial explosion [7].

The building blocks of many evolutionary-based complex detection algorithms, provided in the literature, are designed based on the topological domain of the protein network to be decomposed. The key contribution of this paper is to revisit the design of evolutionary algorithms and extend their framework to cope with biological information. Two components of the EA framework, crossover and mutation, will be viewed as different manifestations of GO-based heuristic operators.

The remainder of this paper starts with a related work to our proposed EA-based complex detection algorithm and a brief introduction to the graph, topology, and ontology of PPINs. This is followed by introducing the proposed GO-based formulations that are used in regulating the flow of the proposed evolutionary-based complex detection algorithms. The discussions and results provided demonstrate that it is curious enough to develop only non-ontology-based complex detection algorithms. Finally, this paper ends by briefly concluding the work and giving recommendations for further research to start such new ramifications.

2. Related work

One of the earliest evolutionary-based complex detection algorithms is suggested in 2014 [8]. All the components of the designed evolutionary algorithm are based on the adjacency matrix (i.e., topology domain). Further, they modeled the problem as a single objective function with different complex structure formulations (modularity, community fitness, conductance, community score, expansion, internal density, normalized cut, negative ratio association, and ratio cut). Again, all these structure formulations are based on the simplified graph approaches. They proved that evolutionary-based complex detection methods are more robust than other state-of-the-art greedy methods.

In [9], a multi-objective complex detection model is introduced as two objective functions to be optimized by a multi-objective evolutionary algorithm. Further a dual-heuristic mutation operator (known as “protein complex attraction and repulsion”) is also suggested to improve the performance of the single-objective evolutionary-based algorithm of [8] and their multi-objective evolutionary algorithms. Unlike the traditional non-heuristic mutation operator, more inherent topological properties at both the complex level and the protein level are exposed by the dual-heuristic operator. At the complex level, the proposed heuristic operator aims at revealing the complex’s sparse interactions while delimiting its proper boundary as closely as possible. On the other hand, at the protein level, the heuristic mutation resembles the heuristic mutation operator proposed in [10] to re-assign the protein-complex attribution in such a way that more protein intra-connections are perceived. Another example of a topological-based heuristic mutation operator is proposed in [11, 12] to breakdown the coexistence of a pair of proteins according to their topological similarity. Their interactions can serve for either intra-delineation topology or inter-delineation topology.

However, when complex PPINs are simplified in graph theory as topology domains composed of proteins (nodes) and interactions (edges), the functional richness of each protein is lost. Unfortunately, to our knowledge, only a few works in the literature, such as [13, 14], have examined the incorporation of the functional domain in the design of evolutionary-based complex detection algorithms. Biological information as reflected by the gene ontology (GO) drives similarity into the design of a multi-objective evolutionary algorithm. However, they only applied a direct way of modeling it in the optimization function, with the aim of maximizing functional modules. Both [13] and [14] developed a topological-based protein-cluster contribution objective and closeness centrality objectives. In [13], the proposed GO-based function is formulated with respect to the measure of [15]. In [14], another objective function based on the direct GO annotation with the average of pairwise GO semantic similarity is formulated.

Recent studies have investigated the injection of biological information into the flow of EA operators, i.e., crossover and mutation [16] and [17]. In [16], a new crossover operator is proposed that incorporates biological information in the form of both gene semantic similarity and protein functional similarity in its design. The proposed GO-based crossover operator has the ability to partition proteins into more accurate complex structures than the counterpart canonical crossover operator. In [17], an EA with a GO-based mutation operator is proposed in an attempt to be more effective than a canonical EA in detecting protein complexes. The operator works with the three GO categories and their fusion.

3. Background

Mathematically, a PPIN is defined as an interaction graph with finite sets of n nodes (proteins) and m edges (interactions). Figure 1 depicts a PPIN from the yeast Saccharomyces cerevisiae interactome, which contains up to 4687 potential interactions for 990 different proteins. In Figure 1, the network (top left) is decomposed into several complexes (top right) on the basis of 81 known complexes annotated from the collection of protein complexes in the Munich Information Center for Protein Sequence (MIPS) database [18].

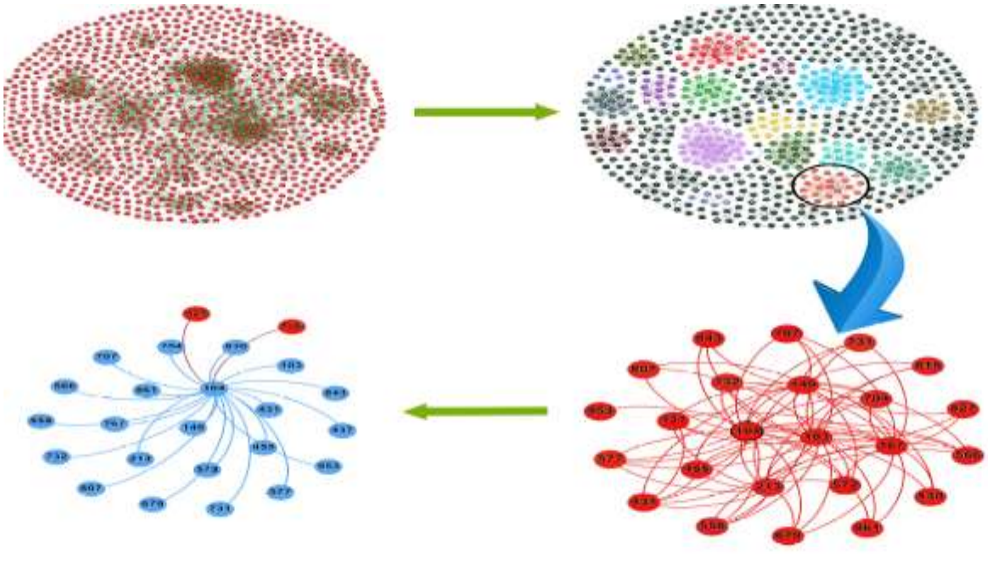

Figure 1: An illustrative example for an interaction graph representing a yeast PPIN with 990 different proteins and 4687 potential interactions (top left).

One of the identified complexes and one of its proteins are zoomed out at the bottom. In bottom right, the complex structure with all its proteins and all its intra-connections. Protein 'YCR046C' (protein #104) is zoomed out with all its intra-connections (blue edges) and its two inter-connections (red edges).

3.1 Topology domain

Usually, the interaction graph can be represented by a symmetric adjacency matrix $A = [a_{ij}]^{n\times n}$, where protein pair P_i and P_j are said to be adjacent (i.e. $a_{ij} = 1$ and $a_{ji} = 1$) if there is an interaction between them. Otherwise, $a_{ij} = 0$ and $a_{ji} = 0$.

3.2 Gene ontology and functional domain

Gene ontology (GO) is the guide for describing gene and protein functions. Actually, the correlation of a gene product to a GO term is called GO annotation. One widespread use is to derive commonalities in the location or function of genes that are over- or under-expressed. The goal of GO is to create strict shared vocabularies and clarify its role across different organisms [19]. These strict vocabularies are separated into three sub-ontologies: Molecular Function (MF), Cellular Component (CC), and Biological Process (BP). Functional similarity (FS) quantifies protein pairwise functional similarity scores and thus infers their relationships based on their GOAs. Also, from FS, we can formulate a functional-based similarity matrix FS.

4. Functional domain for designing the proposed EA

The proposed evolutionary-based complex detection algorithm aims to conceptually relate its components to more biological knowledge. Beside the topological domain A, we call for another, but functional, domain (F).

4.1 F matrix versus A matrix

The adopted functional domain is based on the direct GO annotations for the proteins. In this domain, the Jaccard similarity metric (Eq. 1) is used:

$$F = [FS_{ij}]^{n\times n} \tag{1}$$

where FS_{ij} is the functional similarity (Eq. 2) between the direct GO Slim terms T_{Pi} and T_{Pj} of a protein pair P_i and P_j, respectively.

$$FS_{ij} = \frac{|T_{Pi} \cap T_{Pj}|}{|T_{Pi} \cup T_{Pj}|} = \frac{|T_{Pi} \cap T_{Pj}|}{|T_{Pi}| + |T_{Pj}| - |T_{Pi} \cap T_{Pj}|} \tag{2}$$

where T_{Pi} and T_{Pj} are the set of the three sub-ontology terms for, respectively, protein T_{Pi} and T_{Pj}:

$$T_{Pi} = \{MF_i \cup BP_i \cup CC_i\} \tag{3}$$

$$T_{Pj} = \{MF_j \cup BP_j \cup CC_j\} \tag{4}$$

Let us consider a set of protein pairs depicted in Figure 1 (bottom right), where the common protein that pairs all other proteins is protein \#104 'YCR046C'. In the following example, the considered pairs are of protein \#104 ('YCR046C') with: protein \#167 ('YML025C'), protein \#572 ('YJL063C'), and protein \#425 ('YNL284C'). Further, let us consider another uncoupled protein, \#827 ('YBR122C') with protein \#104 ('YCR046C'). Thus, protein \#827 ('YBR122C') is apart from the sub-PPIN depicted in Figure 1 (bottom right). The topological information of these pairs is as advised by the adjacency matrix from the yeast Saccharomyces cerevisiae intractome. This intractome entails the neighboring or existence of interactions (i.e., $a_{ij} = 1$ and $a_{ji} = 1$) between protein 'YCR046C' and each of protein \#167 ('YML025C'), protein \#572 ('YJL063C'), and protein \#425 ('YNL284C'). On the other hand, there is no interaction between 'YCR046C' and 'YBR122C' (i.e. $a_{ij} = 0$ and $a_{ji} = 0$). Referring to the MIPS catalog, however, we found that both protein \#167 ('YML025C') and protein \#572 ('YJL063C') are defined to be paired with protein \#104 ('YCR046C') in the same complex (i.e., their connections are defined to be intra-connections with protein \#104 'YCR046C'). However, protein \#425 ('YNL284C') is defined to be interconnected with protein \#104 ('YCR046C'). In other words, although they are neighbors, they are located at two different complexes. Finally, the detached protein \#425 ('YNL284C') to protein \#104 ('YCR046C'), with respect to the MIPS catalog, is found to be located at the same complex. Shortly speaking, one can see that the topological information (as provided by the adjacency matrix for the pair of neighbor proteins \#104 and \#425 and for the pair of detached proteins \#104 and \#827) tends to deceive or mislead the unmasking ability of the complex detection algorithm in such a way as to faultily gather proteins \#104 and \#425 at the same complex, while scattering away proteins \#104 and \#827 at two different complexes.

Unlike the topological information provided by the domain of the adjacency matrix, consider the biological domain as supplied by the GO terms. The common and tuncommon GO terms for protein 'YCR046C' with the three neighbor proteins: protein \#167 ('YML025C'), protein \#572 ('YJL063C'), and protein \#425 ('YNL284C') and with the disjoint protein \#827 ('YBR122C') are depicted in Figures 2 - 5.

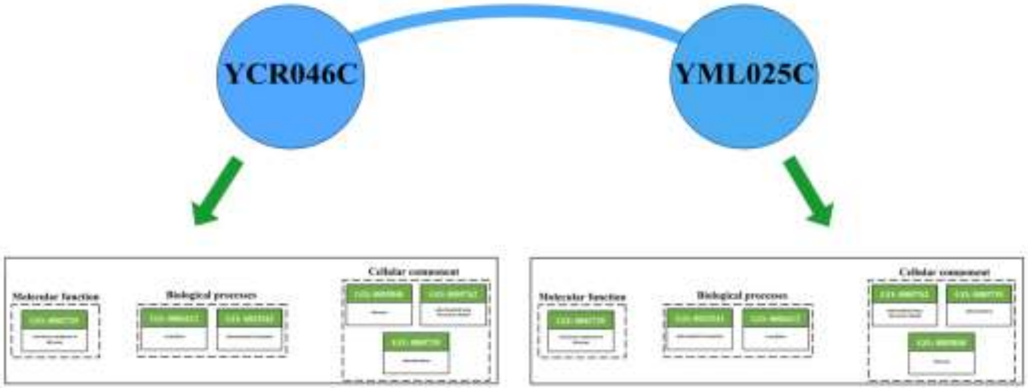

Figure 2: Direct GO terms of both protein #104: ’YCR046C’ and protein #167: ’YML025C’. Similar MF, BP, and CC terms are clarified with green color.

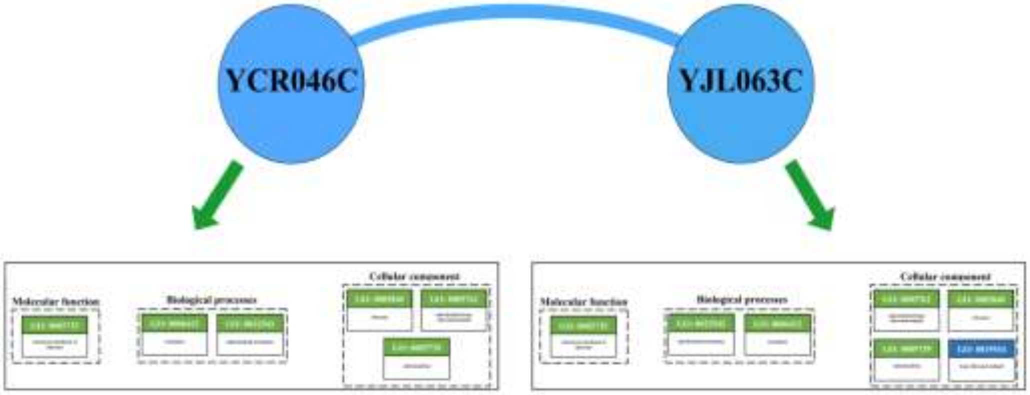

Figure 3: Direct GO terms of both protein #104: ’YCR046C’ and protein #572: ’YJL063C’. Similar MF, BP, and CC terms are clarified with green color.

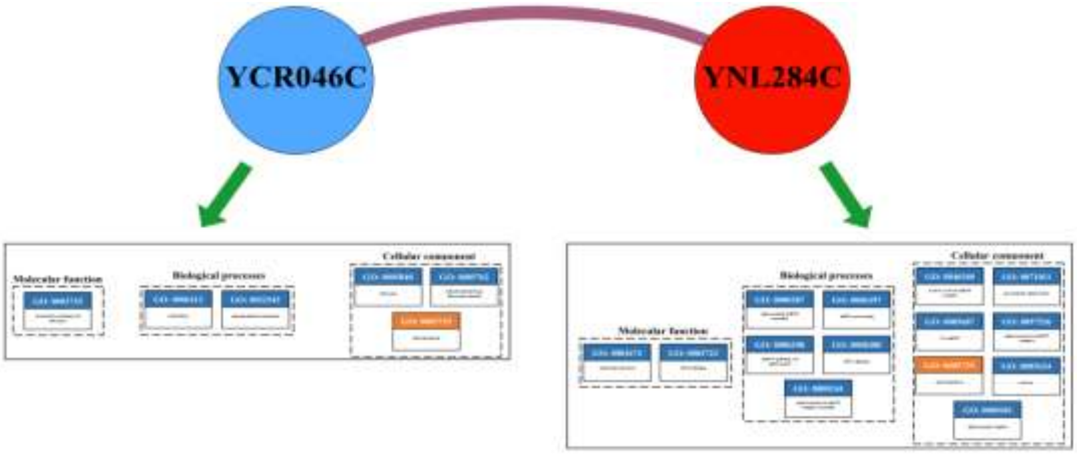

Figure 4: Direct GO terms of both protein #104:'YCR046C' and protein #425: 'YNL284C'. Similar MF, BP, and CC terms are clarified with orange color.

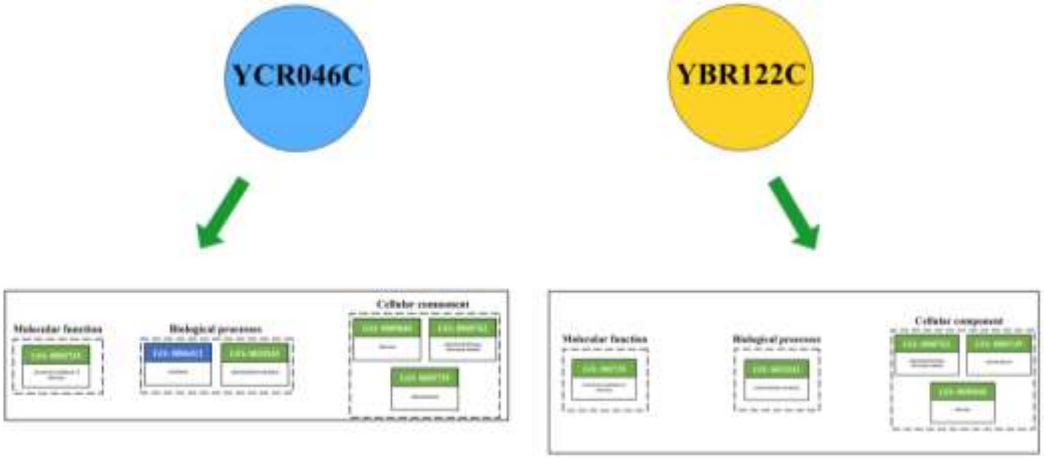

Figure 5: Direct GO terms of both protein #104: 'YCR046C' and protein #827: 'YBR122C'. Similar MF, BP, and CC terms are clarified with green color.

In these figures, protein \#104 ('YCR046C') and each of the intra-neighbor protein \#167 ('YML025C') and protein \#572 ('YJL063C') are depicted in blue color with their intra-connections. However, the inter-neighbor protein \#425 ('YNL284C') with its interconnection to protein \#104 ('YCR046C') is colored red. Finally, the disjointed protein \#827 ('YBR122C') is depicted in yellow. Also, in the figures, the common terms between protein \#104 ('YCR046C') and each of the intra-complex proteins, i.e. the intra-connected protein \#167 ('YML025C') and protein \#572 ('YJL063C'), and the disjoint protein \#827 ('YBR122C'), are colored green, while the uncommon terms are given in blue color. However, the common/uncommon GO terms with respect to the inter-neighbor protein \#425 ('YNL284C'), are given, respectively, in orange and blue colors. These figures simply provide a useful perspective that the individual functional similarities between protein \#104 ('YCR046C') and each of its neighbors: protein \#167 ('YML025C'), protein \#572 ('YJL063C'), and even the disjoint protein \#827 are greater than the functional similarity with the counterpart inter-neighbor protein \#425 ('YNL284C'). This can be seen by the number of common and uncommon GO terms in each figure.

4.2 The proposed GO-based EA

The proposed GO-based EA is defined as an iterated transformation function that is composed of a sequence of sub-functions. It starts with an initial population of encoded solutions. The locus-based individual representation is used to encode each partitioning solution. In locus-based representation, here, each individual chromosome I in the population $\mathbb{I} = \{I_1, I_2, \dots, I_{pop-size}\}$ is defined as a collection of n proteins–proteins fitting in the same complex. Each chromosome, P_i, has n entities, i.e. $I_i = \{I_{i,1}, I_{i,2}, \dots, I_{i,n}\}$, where the locus, j, points to the protein, P_j, and the allele value, $I_{i,j}$, points to one of the neighboring of proteins P_j should fit in the complex formation. The decoding function δ of an individual chromosome sketches different complex structures, and thus a different number of complexes, for the network.

Good solutions, and thus good areas of the search space, are quantitatively evaluated using the modularity function, Q (Eq. 5) [20].

$$Q(\mathcal{C}) = \sum_{i=1}^{K}\left[\frac{m_{C_i}}{|m|} - (\frac{\sum_{v\in C_i}|d_v|}{2|m|})^2\right] \tag{5}$$

Then, another set of solutions will be generated by the iterative composition of three main evolution operators. These are selection, the proposed GO-based crossover, and the proposed GO-based mutation. A set of parent solutions is selected, using binary tournament selection, based on their modularity values. The canonical uniform crossover, r_{pc}, with crossover probability, pc, forms an offspring individual by uniformly inheriting the *topological* information from the two individual parents I_1 and I_2. Thus, uniform crossover works as follows:

$\forall i|\ 1 \le i \le pop-size\ \wedge \forall j|\ 1 \le j \le n$:

$$I'_{i,j} = \begin{cases} I_{1,j} & if\ rand \le 0.5 \\ I_{2,j} & otherwise \end{cases} \tag{6}$$

where $rand$ is a uniform random value, sampled a new for each protein P_j. The canonical crossover operator uniformly inherits the topological information from two individual parents. Figure 7 depicts an example of a uniform crossover.

In this paper, we proposed a GO-based crossover as follows:

$\forall i|\ 1 \le i \le pop-size\ \wedge \forall j|\ 1 \le j \le n$:

$$I'_{i,j} = \begin{cases} I_{1,j} & if\ FS_{P_j,I_{1,j}} > FS_{P_j,I_{2,j}} \\ I_{2,j} & otherwise \end{cases} \tag{7}$$

The essential rule of the proposed mutation operator is based on the migration operator of [10]. The operator works on allele values (i.e., protein neighboring) and alters, with a specified normally low migration probability p_m, its neighbor. In this paper, a redesign of the topological-based mutation operator proposed in [10] to work under the functional domain is offered. A protein GO-based mutation operator, m_{pm} when activated for a given protein, will change the complex of this protein to a new complex where it could maintain the maximum function homogeneity. The general sketch for the proposed GO-based mutation operator is outlined in Algorithm 1.

Algorithm 1 General sketch for the proposed GO-based mutation

```
input $Ii \leftarrow \{Ii,1, Ii,2, . . . , Ii,n\}$;/* Genotype of an individual $i$ */
input $Ci \leftarrow \{C1,C2, . . . ,CK \}$;/* Phenotype $Ci$ as a set of complexes */
input $A$, $F$;/* Topological and functional information for $n$ proteins */
output $Ii$ , $Ci$ ;/* Genotype and phenotype of the mutated individual $i$ */
/* For each protein $j$ in $Ii$ */
for $j \in \{1, . . . , n\}$ do
    if ($\chi j \leq pm$) then
        $Cj \leftarrow C|C \in Ci \wedge Pj \in C$;/* Home complex to which protein $j$ belongs */
        $SumFS \leftarrow \sum_{p \in Cj} FS_{jp}$;/* Functional similarity of protein $j$ in $Cj$ */
        /* For each complex $Ck \neq Cj$ and there is a neighbor protein to protein $j$ in $Ck$ */
        for $k \in \{1, . . . ,K\}$ do
            if ($Ck \neq Cj$ ) $\wedge$ ($\exists p|a\ j\ p \leftarrow 1$) then
                $NewSumFS \leftarrow \sum_{p \in Cj} FS_{jp}$;
                if ($NewSumFS > SumFS$) then
                    $SumFS \leftarrow NewSumFS$;/* Update the functional similarity */
                    $Ii, j \leftarrow p$;/* Update the genotype at locus $j$ */
                    $Cj \leftarrow Ck$ ;/* Update the complex to which protein $j$ migrates*/
                end if
            end if
        end for
    end if
end for
```

5. Results and discussions

In this section, we aim to compare the performance of these GO-based EAs against the state-of-the-art topological-based EA used in the literature [8]. Two yeast Saccharomyces cerevisiae PPINs are used in the performance evaluation. The GO terms assigned to the proteins were downloaded from the Saccharomyces Genome Database (SGD) at http://genome-www.stanford.edu/Saccharomyces/ in the period June 2021-April 2022. The networks are denoted as PPI-D1 and PPI-D2. PPI-D1 contains 4687 interactions for 990 proteins annotated with a total of 1245 BP, 452 CC, and 541 MF GO terms. PPI-D2 contains 6993 interactions for 1443 proteins annotated with a total of 1570 BP terms, 566CC terms, and 659 MF terms.

To validate the quality of the generated solutions for the proposed algorithms, two benchmark gold standard complex sets are considered, drawn from the Munich Information Center for Protein Sequence (MIPS) catalog (Complex-D1 and Complex-D2) [16]. A predicted complex C_i matches one of the true complexes from the benchmark set in S^* (say S_j), if the proteins of both complexes overlap or intersect with overlapping score (Eq. 8 and Eq. 9) that is equal to or greater than a specified threshold σ_{OS}. In the experiments, we set OS to range from 0.1 to 0.8, in an incremental step of 0.05.

$$OS(C_i, S_j) = \frac{|C_i \cap S_j|^2}{|C_i||S_j|} \tag{8}$$

where |·| is refer to the number of common proteins to both a predicted complex and a true standard complex.

$$match\left(C_i, S_j\right) = \begin{cases} 1 \;\; if \;\; OS(C_i, S_j) \geq \sigma_{OS} \\ 0 \;\; otherwisex \end{cases} \tag{9}$$

The percentage of the true benchmark complexes that match (with respect to the overlapping score) any of the detected complexes is known as $recall$. On the other hand, $precision$ refers to the fraction of the detected complexes that match any of the true complexes. The F score, then, represents the harmonic mean of both recall and precision.

$$recall = \frac{|S_i|S_i \in S^* \wedge \exists C_j \in C \rightarrow match(S_i, C_j)}{K^*} \tag{10}$$

$$precision = \frac{|C_i|C_i \in C \wedge \exists S_j \in S^* \rightarrow match(C_i, S_j)}{K} \tag{11}$$

$$F = \frac{2 \times recall \times precision}{recall + precision} \tag{12}$$

While recall and precision measure the cumulative quality of the detected complexes for an algorithmic prediction at the complex level, $\mathrm{recall_N}$ and $\mathrm{precision_N}$ can estimate the detection accuracy at the protein level. Finally, $\mathrm{F_N}$ score imitates the F score but at the protein level for both $recall_N$ and $precision_N$ measures.

$$recall_N = \frac{\sum_{i=1}^{K_S} |m_i|}{\sum_{j=1}^{K_S} |S_i|} \tag{13}$$

where $|m_i| = max_{|C_i \cap S_j|} \{\forall\, S_j \in \; S^* \wedge match\big(S_i, C_j\big) \geq \sigma_{OS}\}$

$$precision_N = \frac{\sum_{i=1}^{K_C} |m_i|}{\sum_{i=1}^{K_C} |C_i|} \tag{14}$$

where $|m_i| = max_{|C_j \cap S_i|} \{\forall\, C_j \in \; C^* \wedge match\big(C_j, S_i\big) \geq \sigma_{OS}\}$

$$F_N = \frac{2 \, \times \, recall_N \, \times \, precision_N}{recall_n + \, precision_N} \tag{15}$$

The setting of the parameters is allowed to match, more or less, the settings used in the literature [8–12], [16], and [17]. The population size is set to 100. The maximum number of generations used to stop the evolutionary process is set to 100 (i.e., 10,000 function evaluations). Control parameters for the main evolutionary operators are set to the following: the probability of uniform crossover, $p_c = 0.8$, the probability of the mutation operator, $p_m = 0.2$, and the probability of the

GO-based mutation operator is also set to $p_m = 0.2$. The evaluation metrics are reported in Tables 1–6 for the average of 30 different simulation runs for the best solutions obtained (in terms of modularity). For ease of comparison, the results of the proposed GO-based EAs are highlighted in bold when they outperform the counterpart canonical EA.

Table 1: Performance comparison for PPI-D1 in terms of $recall$, and $recall_N$ for an average of 30 runs.

OS	$recall$			$recall_N$		
	EA	EA_{GOx}	EA_{GOm}	EA	EA_{GOx}	EA_{GOm}
0.10	0.9029	**0.9457**	0.8363	0.8349	0.8223	**0.8647**
0.15	0.8504	**0.8979**	0.7786	0.8056	0.7961	**0.8212**
0.20	0.7978	**0.8572**	0.7329	0.7676	**0.7743**	**0.7816**
0.25	0.7568	**0.8222**	0.6816	0.7174	**0.7428**	0.7004
0.30	0.7183	**0.7885**	0.6491	0.6640	**0.7053**	0.6331
0.35	0.6871	**0.7534**	0.6329	0.6349	**0.6805**	0.6145
0.40	0.6598	**0.7278**	0.6085	0.5976	**0.6517**	0.5759
0.45	0.6192	**0.6885**	0.5756	0.5489	**0.6079**	0.5256
0.50	0.5987	**0.6769**	0.5624	0.5161	**0.5904**	0.5056
0.55	0.5581	**0.6346**	0.5479	0.4793	**0.5485**	**0.4876**
0.60	0.5384	**0.6047**	0.5325	0.4572	**0.5121**	**0.4732**
0.65	0.5106	**0.5662**	**0.5252**	0.4281	**0.4718**	**0.4583**
0.70	0.4837	**0.5282**	**0.5171**	0.3987	**0.4364**	**0.4489**
0.75	0.4431	**0.4752**	**0.4932**	0.3676	**0.3897**	**0.4252**
0.80	0.4170	**0.4291**	**0.4662**	0.3499	**0.3516**	**0.4045**

Table 2: Performance comparison for PPI-D2 in terms of $recall$, and $recall_N$ for an average of 30 runs.

OS	$recall$			$recall_N$		
	EA	EA_{GOx}	EA_{GOm}	EA	EA_{GOx}	EA_{GOm}
0.10	0.9598	**0.9820**	0.9344	0.5744	0.5398	**0.6275**
0.15	0.8956	**0.9353**	0.8518	0.5535	0.5263	**0.5985**
0.20	0.8344	**0.8687**	0.7704	0.5290	0.4960	**0.5651**
0.25	0.7622	**0.8064**	0.6991	0.4804	0.4672	**0.5187**
0.30	0.6900	**0.7411**	0.6451	0.4416	0.4355	**0.4911**
0.35	0.6158	**0.6687**	0.5744	0.3968	**0.4010**	**0.4346**
0.40	0.5676	**0.6173**	0.5316	0.3550	**0.3566**	**0.3864**
0.45	0.4993	**0.5433**	0.4649	0.3064	**0.3158**	**0.3229**
0.50	0.4784	**0.5180**	0.4396	0.2792	**0.2862**	**0.2850**
0.55	0.4027	**0.4367**	0.3884	0.2403	**0.2472**	**0.2579**
0.60	0.3662	**0.3987**	0.3582	0.2151	**0.2217**	**0.2370**
0.65	0.3207	**0.3476**	**0.3364**	0.1899	**0.1921**	**0.2103**
0.70	0.2662	**0.2951**	**0.2798**	0.1488	**0.1613**	**0.1730**
0.75	0.2367	**0.2644**	**0.2620**	0.1322	**0.1413**	**0.1488**
0.80	0.2040	**0.2249**	**0.2449**	0.1083	**0.1167**	**0.1362**

Table 3: Performance comparison for PPI-D1 in terms of $precesion$, and $precesion_N$ for an average of 30 runs.

OS	$precesion$			$precesion_N$		
	EA	EA_{GOx}	EA_{GOm}	EA	EA_{GOx}	EA_{GOm}
0.10	0.7776	0.7632	**0.8001**	0.6868	**0.7522**	0.6530
0.15	0.7304	0.7102	**0.7691**	0.6802	**0.7417**	0.6501
0.20	0.7148	0.6872	**0.7607**	0.6755	**0.7352**	0.6454
0.25	0.7015	0.6703	**0.7534**	0.6629	**0.7261**	0.6298
0.30	0.6710	0.6313	**0.7353**	0.6367	**0.6995**	0.5943
0.35	0.6456	0.5994	**0.7249**	0.6154	**0.6780**	0.5808
0.40	0.6257	0.5807	**0.7140**	0.5895	**0.6556**	0.5603
0.45	0.5894	0.5406	**0.6887**	0.5459	**0.6120**	0.5203
0.50	0.5716	0.5305	**0.6800**	0.5160	**0.5932**	0.5046
0.55	0.5294	0.4924	**0.6629**	0.4793	**0.5485**	**0.4876**
0.60	0.5107	0.4696	**0.6443**	0.4572	**0.5121**	**0.4732**
0.65	0.4843	0.4395	**0.6355**	0.4281	**0.4718**	**0.4583**
0.70	0.4591	0.4096	**0.6257**	0.3987	**0.4364**	**0.4489**
0.75	0.4207	0.3687	**0.5967**	0.3676	**0.3897**	**0.4252**
0.80	0.3960	0.3329	**0.5642**	0.3499	**0.3516**	**0.4045**

Table 4: Performance comparison for PPI-D2 in terms of $precesion$, and $precesion_N$ for an average of 30 runs.

OS	$precesion$			$precesion_N$		
	EA	EA_{GOx}	EA_{GOm}	EA	EA_{GOx}	EA_{GOm}
0.10	0.6059	0.5897	0.5803	0.7061	**0.7535**	0.6960
0.15	0.5749	0.5506	0.5672	0.6976	**0.7399**	0.6932
0.20	0.5456	0.5191	0.5454	0.6846	**0.7199**	0.6823
0.25	0.4935	0.4666	**0.5043**	0.6640	**0.6928**	**0.6640**
0.30	0.4721	0.4422	**0.4991**	0.6472	**0.6743**	**0.6562**
0.35	0.4369	0.4112	**0.4776**	0.6105	**0.6458**	**0.6438**
0.40	0.4175	0.3936	**0.4569**	0.5626	**0.6083**	**0.5968**
0.45	0.3818	0.3566	**0.4264**	0.5227	**0.5644**	**0.5509**
0.50	0.3737	0.3486	**0.4156**	0.5008	**0.5411**	**0.5221**
0.55	0.3214	0.2983	**0.3802**	0.4641	**0.4962**	**0.4907**
0.60	0.3013	0.2793	**0.3609**	0.4366	**0.4628**	**0.4709**
0.65	0.2773	0.2562	**0.3499**	0.4007	**0.4186**	**0.4456**
0.70	0.2392	0.2230	**0.3225**	0.3401	**0.3729**	**0.4196**
0.75	0.2148	0.2010	**0.3022**	0.3076	**0.3292**	**0.3611**
0.80	0.1929	0.1791	**0.2799**	0.2614	**0.2798**	**0.3259**

Table 5: Performance comparison for PPI-D1 in terms of F, and F_N for an average of 30 runs.

OS	F			F_N		
	EA	EA_{GOx}	EA_{GOm}	EA	EA_{GOx}	EA_{GOm}
0.10	0.8352	**0.8444**	0.8174	0.7534	**0.7853**	0.7439
0.15	0.7856	**0.7928**	0.7733	0.7373	**0.7676**	0.7255
0.20	0.7539	**0.7623**	0.7461	0.7182	**0.7540**	0.7068
0.25	0.7279	**0.7378**	0.7152	0.6889	**0.7342**	0.6630
0.30	0.6936	**0.7006**	0.6892	0.6500	**0.7023**	0.6130
0.35	0.6654	**0.6671**	**0.6755**	0.6249	**0.6791**	0.5971
0.40	0.6420	**0.6454**	**0.6568**	0.5935	**0.6536**	0.5679
0.45	0.6037	**0.6050**	**0.6269**	0.5474	**0.6100**	0.5229
0.50	0.5847	**0.5943**	**0.6154**	0.5161	**0.5918**	0.5051
0.55	0.5432	**0.5541**	**0.5997**	0.4793	**0.5483**	**0.4876**
0.60	0.5241	**0.5282**	**0.5829**	0.4573	**0.5121**	**0.4732**
0.65	0.4970	**0.4944**	**0.5749**	0.4282	**0.4718**	**0.4583**
0.70	0.4710	0.4611	**0.5660**	0.3988	**0.4364**	**0.4489**
0.75	0.4315	0.4148	**0.5398**	0.3677	**0.3897**	**0.4252**
0.80	0.4062	0.3746	**0.5104**	0.3499	**0.3516**	**0.4045**

Table 6: Performance comparison for PPI-D2 in terms of F, and F_N for an average of 30 runs.

OS	F			F_N		
	EA	EA_{GOx}	EA_{GOm}	EA	EA_{GOx}	EA_{GOm}
0.10	0.7426	0.7367	0.7159	0.6332	0.6285	**0.6598**
0.15	0.6998	0.6930	0.6808	0.6170	0.6147	**0.6423**
0.20	0.6594	0.6496	0.6385	0.5965	0.5869	**0.6180**
0.25	0.5989	0.5910	0.5858	0.5572	**0.5577**	**0.5822**
0.30	0.5604	0.5537	**0.5627**	0.5247	**0.5288**	**0.5616**
0.35	0.5108	0.5090	**0.5213**	0.4806	**0.4944**	**0.5183**
0.40	0.4808	0.4803	**0.4911**	0.4316	**0.4491**	**0.4684**
0.45	0.4323	0.4301	**0.4445**	0.3860	**0.4045**	**0.4066**
0.50	0.4193	0.4163	**0.4270**	0.3582	**0.3739**	**0.3681**
0.55	0.3571	0.3541	**0.3841**	0.3163	**0.3297**	**0.3376**
0.60	0.3303	0.3281	**0.3593**	0.2878	**0.2994**	**0.3146**
0.65	0.2971	0.2947	**0.3428**	0.2572	**0.2628**	**0.2852**
0.70	0.2518	**0.2537**	**0.2996**	0.2067	**0.2249**	**0.2449**
0.75	0.2250	**0.2281**	**0.2806**	0.1848	**0.1974**	**0.2107**
0.80	0.1981	**0.1992**	**0.2612**	0.1530	**0.1645**	**0.1921**

Recall that by the definition of recall, precision, and F measures on the one hand, and by the definition of $\mathrm{recall}_\mathrm{N}$, $\mathrm{precision}_\mathrm{N}$, and F_N measures on the other hand, one can say that the group of complex-level measures is interested in the quantity of the matched complexes. On the other hand, the protein-level measures are used to assess the quality of the matched complexes. In other words, while recall assesses the quantity of matched benchmark reference (MIPS) complexes, $\mathrm{recall_N}$ assesses their quality. Similarly, $\mathrm{precision}$ is used to assess the quantity of matched predicted complexes, while $\mathrm{precision_N}$ assesses their quality. Finally, F and $\mathrm{F_N}$ assess, respectively, the overall performance for the quantity and quality of matched complexes. First, for the quantity of the matched complexes, based on the results reported in Tables 1–6, one can observe that the proposed GO-based crossover operator and the proposed GO-based mutation operator work as opposite operators. The GO-based crossover operator works as a fine-grained operator to yield relatively many small-sized complexes, which in turn results in rediscovering more known MIPS complexes by the proposed EA_{GOx} (as reflected by recall values in Table 1 and Table 2) than the counterpart canonical EA. The GO-based mutation operator, on the other hand, works as a coarse-grained operator to return relatively a limited number of large-sized complexes, and thus more matched predicted complexes are obtained by EA_{GOm} (as reflected by $\mathrm{precision}$ values in Tables 3 and 4) than the counterpart canonical EA. On overall, the positive impact of the GO-based crossover operator and the GO-based mutation operator, individually, enable their GO-based EAs (i.e. EA_{GOx} and EA_{GOm}) to achieve more matched real and predicted complexes than the canonical EA (as reflected by F values in Table 3).

Second, for the quality of the matched complexes, one can also observe that the proposed GO-based crossover operator and the proposed GO-based mutation operator work as opposite operators. High qualified MIPS complexes and high qualified predicted complexes are matched better by, respectively, the proposed EA_{GOx} and EA_{GOm} than the counterpart canonical EA (as reflected by, respectively, $\mathrm{recall_N}$, and $\mathrm{precision_N}$ in Tables 1 - 6). This in turns yields high correct matched complexes for the proposed EA_{GOx} and generally for higher values of overlapping score for the proposed EA_{GOm} (as reflected by $\mathrm{F_N}$ values in Table 5 and Table 6).

Finally, interested behavior can also be obtained by the proposed GO-based crossover operator and the proposed GO-based mutation operator, which will empower the proposed GO-based EAs to be more robust. This is reflected by the more reliable results obtained after the proposed EA_{GOx} and EA_{GOm} than the counterpart canonical EA (in Tables 1-6) when the detection problem becomes harder to solve by increasing the matching score (i.e., OS) to more than 0.5.

6. Conclusions and future works

The design of one or more GO-based components in the framework of the EA gives it the green light to easily outperform the state-of-the-art heuristic algorithms. The injection of the GO information into the evolutionary operators of the EA (crossover and mutation) has proven to improve the performance of the algorithm. The injection of the GO information into the evolutionary operators of the EA is proven to improve the performance of the algorithm. The injection of the GO information empowers the crossover operator to freeze out the main lack of the traditional modularity model (i.e., the resolution limit) and to generate fine-grain complexes with homogeneous structures. The additional biological information enables the GO-based mutation to generate coarse-grain complexes with more homogeneous structures. The collaboration of the modularity with the GO-based mutation operator or the GO-based crossover operator promotes the whole EA framework to further adjust the functional structures of the coarse and fine-grain complexes.

One of the main ramifications of the current work is the investigation of the impact of GO on the performance of multi-objective evolutionary algorithms. Likewise, many real-world problems, such as complex detection in PPINs, can be better delineated when depicted as a multi-objective optimization problem (MOP) and tackled by a multi-objective optimization algorithm (MOEA). Unfortunately, insufficient investigation room is indicated in the literature for EAs, particularly MOEAs. In designing a multi-objective optimization model to control the work of any multi-objective evolutionary algorithm, it should be wise to pay special attention to the contrasting modes for coping with the contradictory requirements ascribed to the intra-complex and inter-complex structures.

Declaration

The authors declare that all figures associated with the paper were prepared by them.

References

[1] 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, p. 1230002, 2013.

[2] King, A D et al., “Protein complex prediction via cost-based clustering,” *Bioinformatics (Oxford, England)*, vol. 20, No. 17, pp. 3013-3020, (2004). doi:10.1093/bioinformatics/bth351

[3] Bader, G.D., Hogue, C.W, “An automated method for finding molecular complexes in large protein interaction networks,” *BMC Bioinformatics*, vol. 4, no. 2, pp. 1-27, 2003. https://doi.org/10.1186/1471-2105-4-2

[4] X.-L. Li, C.-S. Foo, S.-H. Tan, and S.-K. Ng., “Interaction graph mining for protein complexes using local clique merging,” *Genome Informatics*, vol. 16, no. 2, pp. 260–269, 2005.

[5] Li, Xiao-Li et al., “Discovering protein complexes in dense reliable neighborhoods of protein interaction networks,” *Computational systems bioinformatics*, *Computational Systems Bioinformatics Conference*, vol. 6, pp. 157-168, 2007.

[6] Ahn, YY., Bagrow, J. & Lehmann, S., "Link communities reveal multiscale complexity in networks,” *Nature*, Vol. 466, pp. 761–764, 2010. https://doi.org/10.1038/nature09182

[7] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, “Complexity and approximation: Combinatorial optimization problems and their approximability properties,” *Springer Science & Business Media*, 2012.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] A. H. H. Abdulateef, B. A. Attea, and A. N. Rashid, “Heuristic Modularity for Complex Identification in Protein-Protein Interaction Networks”, *Iraqi Journal of Science*, vol. 60, no. 8, pp. 1846–1859, Aug. 2019.

[13] 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.

[14] 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.

[15] D. Lin, "An information-theoretic definition of similarity," in *Icml*, vol. 98, no. 1998, pp. 296-304, 1998.

[16] I. H. Alani, D. A. Alzubaydi, and B. A. Attea, "An Evolutionary Algorithm with Gene Ontology-aware Crossover Operator for Protein Complex Detection, " *unpublished, Iraqi Journal of Science,* vol. 64, No. 4, 2023.

[17] I. H. Abdulateef, D. A. J. Alzubaydi, and B. A. Attea, “A Tri-Gene Ontology Migration Operator for Improving the Performance of Meta-heuristics in Complex Detection Problems”, *Iraqi Journal of Science*, vol. 64, no. 3, pp. 1426–1441, Mar. 2023.

[18] D. F. H.-W. Mewes, U. Gu¨ldener, G. Mannhaupt, K. Mayer, M. Mokrejs, B. Morgenstern, M. Mu¨nsterko¨tter, S. Rudd, and B. Weil, "Mips: a database for genomes and protein sequences," Nucleic acids research, vol. 30, no. 1, pp. 31-34, 2002.

[19] 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.

[20] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821-7826, 2002.