Computational Biology and Chemistry 59 (2015) 28–36

Contents lists available at ScienceDirect

Computational Biology and Chemistry

journal homepage: www.elsevier.com/locate/compbiolchem

Research Article

Impact of heuristics in clustering large biological networks☆

Md. Kishwar Shafin[a], Kazi Lutful Kabir[a], Iffatur Ridwan[a], Tasmiah Tamzid Anannya[a], Rashid Saadman Karim[a], Mohammad Mozammel Hoque[a], M. Sohel Rahman[b,*,1]

[a] Department of CSE, MIST, Mirpur Cantonment, Dhaka 1216, Bangladesh
[b] AℓEDA Group, Department of CSE, BUET, Dhaka 1215, Bangladesh

ARTICLE INFO

Article history:
Received 12 April 2015
Received in revised form 17 May 2015
Accepted 28 May 2015
Available online 26 July 2015

Keywords:
Algorithms
Biological networks
Clustering
Heuristics

ABSTRACT

Traditional clustering algorithms often exhibit poor performance for large networks. On the contrary, greedy algorithms are found to be relatively efficient while uncovering functional modules from large biological networks. The quality of the clusters produced by these greedy techniques largely depends on the underlying heuristics employed. Different heuristics based on different attributes and properties perform differently in terms of the quality of the clusters produced. This motivates us to design new heuristics for clustering large networks. In this paper, we have proposed two new heuristics and analyzed the performance thereof after incorporating those with three different combinations in a recently celebrated greedy clustering algorithm named SPICi. We have extensively analyzed the effectiveness of these new variants. The results are found to be promising.

1. Introduction

Clustering is an important tool in biological network analysis. However, traditional clustering algorithms do not perform well in the analysis of large biological networks being either extremely slow or even unable to cluster (Song and Singh, 2009). On the other hand, recent advancement of the state of the art technologies along with computational predictions have resulted in large scale biological networks for numerous organisms (Brun et al., 2004). As a result, faster clustering algorithms are of tremendous interest. There exist a number of clustering algorithms that work well on small to moderate biological networks. For instance, a number of algorithms in the literature can guarantee that they would generate clusters with some specific properties (e.g., Cfinder (Adamcsek et al., 2006; Palla et al., 2005; Colak et al., 2009; Georgii et al., 2009)). They are however computationally very intensive and hence do not scale well as the size of the biological network increases. Algorithms like WPNCA (Peng et al., 2014) and ClusterONE (Nepusz et al., 2012) are new approaches to handle weighted biological networks of moderate size. But in many cases they fail or require a large amount of time to cluster.

To this end, some more efficient approaches have been introduced most of which are based on greedy techniques (e.g., SPICi (Jiang and Singh, 2010), DPClus (Altaf-Ul-Amin et al., 2006), etc.). However, algorithms like MGclus (Frings et al., 2013) suit relatively well for clustering large biological networks with dense neighborhood. In most cases, clusters produced by greedy approaches are highly dependent on the heuristic(s) employed. It is expected that a better heuristic will yield even more improved results. This motivates us to search for a better heuristic for a well-performing greedy approach to devise an even better clustering algorithm that not only runs faster but also provides quality solutions.

SPICi (Jiang and Singh, 2010) is a relatively recent and new approach among the greedy techniques that can cluster large biological networks. After carefully studying and analyzing the implementation of SPICi, we have discovered that some essential modification in the heuristics employed can bring drastic change in the clusters' quality. In this paper, we have proposed a couple of new heuristics for SPICi with an aim to devise an even better clustering algorithm. We have implemented three new versions of SPICi and have conducted extensive experiments to analyze the performance our new implementations. Experimental results are found to be promising with respect to both speed and accuracy. Some preliminary results of this paper have been presented at Shafin et al. (2015).

2. Background

We start this section with preliminaries on some related notions followed by a discussion on the algorithmic framework of SPICi.

☆ Some preliminary results of this research work was presented at Shafin et al. (2015).
* Corresponding author.
E-mail address: msrahman@cse.buet.ac.bd (M.S. Rahman).
[1] Currently on a Sabbatical leave from BUET.

http://dx.doi.org/10.1016/j.compbiolchem.2015.05.007
1476-9271/© 2015 Elsevier Ltd. All rights reserved.

We will also briefly review the heuristics used in SPICi. A biological network is modeled as an undirected graph $G = (V, E)$ where each edge $(u, v) \in E$ has a *confidence score* $(0 < w_{u,v} \leq 1)$, also called the *weight* of the edge. We say that, $w_{u,v} = 0$, if the two vertices u, v have no edge between them. The *weighted degree* of each vertex u, denoted by $d_w(u)$, is the sum of the confidence scores of all of its incident edges, i.e., $d_w(u) = \sum_{(u,v)\in E} w_{u,v}$. Based on the confidence scores or weights of the edges, we can define the term *density* for a set of vertices $S \subseteq V$ as follows. The density $\mathcal{D}(S)$ of a set $S \subseteq V$ of vertices is defined as the sum of the weights of the edges that have both end vertices belonging to S divided by the total number of possible edges in S. Formally,

$$\mathcal{D}(S) = \frac{\sum_{u,v\in S} w_{u,v}}{|S| \times (|S| - 1)/2}$$

For each vertex u and a set $S \subseteq V$, *support* of u by S, denoted by $\mathcal{S}(u, S)$, is defined as the sum of the confidence scores of the edges of u that are incident to the vertices in S. Formally,

$$\mathcal{S}(u, S) = \sum_{v\in S} w_{u,v}$$

Given a weighted network, the goal of SPICi is to output a set of disjoint dense sub-graphs. SPICi uses a greedy heuristic approach that builds one cluster at a time and expansion of each cluster is done from an original protein seed pair. SPICi depends on two parameters, namely, the *support threshold, T_s* and the *density threshold, T_d*. The use of these two parameters will be clear shortly. Now, we briefly review how SPICi employs its heuristic strategies. In fact, SPICi first selects two seed nodes and then attempt to expand the clusters.

Seed selection

While selecting the seed vertices, SPICi uses a heuristic. Very briefly, at first it chooses a vertex u in the network that has the highest weighted degree. Then it divides the neighboring vertices of u into five bins according to their edge weights, namely, (0, 0.2], (0.2, 0.4], (0.4, 0.6], (0.6, 0.8] and (0.8, 1.0]. Then the vertex with the highest weighted degree belonging to the highest non-empty bin is chosen as the second seed, v. The edge (u, v) is referred to as the seed edge.

Cluster expansion

For cluster expansion, SPICi follows a procedure similar to that of Altaf-Ul-Amin et al. (2006). It works with a vertex set S for the cluster initially containing the two selected seed vertices. It uses a heuristic approach to build the clusters and it builds one cluster at a time. In the cluster expansion step, SPICi searches for the vertex u such that $\mathcal{S}(u, S)$ is maximum amongst all the unclustered vertices that are adjacent to a vertex in S. If $\mathcal{S}(u, S)$ is smaller than a threshold then u is not added to S and $\mathcal{D}(S)$ is updated accordingly. However, if the calculated $\mathcal{D}(S)$ turns out to be smaller than the density threshold T_d then SPICi does not include u in the cluster and output S.

3. Proposed heuristics

The heuristics employed by SPICi are developed based on an observation that two vertices are more likely to be in the same cluster if the weight of the edge between them is higher (Jiang and Singh, 2010).

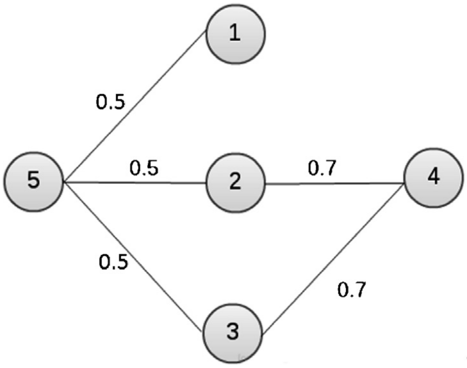

Fig. 1. An example to illustrate the necessity of a new measure.

The two heuristics that SPICi employs are implemented in the form of two procedures, namely, **Search** and **Expand**. In the **Search** procedure, node with the highest outdegree is chosen as the seed and in **Expand**, node with the highest support is selected as the candidate to be added to the cluster. In this paper, we have proposed two heuristics and we combine our heuristics with the heuristics of SPICi to have three new versions of SPICi. We will refer to these three versions as **SPICi**$_+^1$, **SPICi**$_+^2$ and **SPICi**$_+^{12}$. To be specific, we employ a new heuristic and modify the **Expand** procedure of SPICi to get **Expand+**. Similarly, we employ another new heuristic and modify the **Search** procedure of SPICi to get **Search+**. In **SPICi**$_+^1$, we combine **Expand+** with **Search** while in **SPICi**$_+^2$, we combine **Expand** with **Search+**. Both the heuristics are replaced by **Search+** and **Expand+** in **SPICi**$_+^{12}$. In essence, our first heuristic is to choose the node with the highest weighted degree among the neighbors as the first seed and second one is to choose the node with the highest average weighted degree as the candidate to join the cluster.

Note that, the heuristics employed by SPICi are developed based on an observation that two vertices are more likely to be in the same cluster if the weight of the edge between them is higher (Brohee and Helden, 2006). In what follows, we describe our heuristic strategies along with the motivation and rationale behind those.

3.1. Average edge weight

Consider Fig. 1. Here assume that, the current cluster set is $S = \{1, 2, 3\}$ and the set of candidate nodes is $\{4, 5\}$. The goal at this point

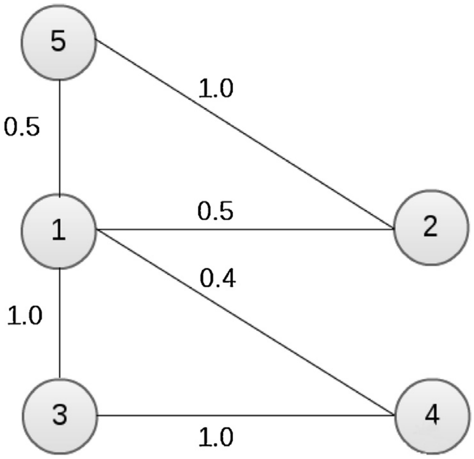

Fig. 2. Another illustration denoting the necessity of a new measure.