HEURISTIC-BASED ANT COLONY OPTIMIZATION ALGORITHM FOR PROTEIN FUNCTIONAL MODULE DETECTION IN PROTEIN INTERACTION NETWORK

JAMALUDIN BIN SALLIM

UNIVERSITI SAINS MALAYSIA

2017

HEURISTIC-BASED ANT COLONY OPTIMIZATION ALGORITHM FOR PROTEIN FUNCTIONAL MODULE DETECTION IN PROTEIN INTERACTION NETWORK

by

JAMALUDIN BIN SALLIM

Thesis submitted in fulfillment of the requirements

for the degree of

Doctor of Philosophy

July 2017

ACKNOWLEDGEMENT

All the praise and thanks be to Allah SWT, the Most Beneficent and the Most Merciful and the Holy Prophet Muhammad SAW, peace upon him.

Ph.D. program has turned out to be a long and challenging, but rewarding journey for me. I would not have made it this far without the help of many people. My supervisor Prof. Dr. Rosni Abdullah has taught me to be a hardworking and effective researcher. She is always supportive of every idea I come up with for my research and helps me to improve and refine it. I wish to thank my co-supervisor, Prof. Dr. Ahamad Tajudin Khader for providing helpful suggestions and enriching my knowledge.

I also would like to thank to the following bodies and personnel for their generous support: Ministry of Higher Education and Universiti Malaysia Pahang for sponsoring my study, Universiti Sains Malaysia for the research grant (1001/PKOMP/841007), Prof. Minoru Kaneheisa for giving me the opportunity to perform research attachment at Bioinformatics Center, Kyoto University, Dr Alex Guttenridge and Dr Thomas Stutzle for their fruitful discussion on the research topics.

Special thanks to Dr Adib and Iznan for a great friendship, PDCC lab Members, *Fazilah, Ibrahim, Hesham, Khalid, Ali Kattan, Abu Marwan, Nizam, Mohanned, Mubarak, Najihah, Ezzedin, Hadri,* and *Amirah* for their technical and moral support.

Last but not least, I would like to express my heartiest gratitude and special regards to my mother, *Hjh Sapora*, father, *Hj Sallim,* wife, *Khomariah* and children, *Nur Dini, Nur Izzah, Nur Sakina, Nur Hidayah* and *Anwar Fahri* for their never ending bonds and support, huge patient that always reminds me to fight hard in completing my study.

TABLE OF CONTENTS

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ALGORITHMS

Page

LIST OF ABBREVIATIONS

ABC	Artificial Bee Colony
ACC	Accuracy
ACO	Ant Colony Optimization
ACS	Ant Colony System
AP	Affinity Propagation
AS	Ant System
BIND	Biomolecular Interaction Network Database
CD-Distance	Czekanowski-Dice distance
ÇoIP	Co-Immunoprecipitation`
DIP	Database of Interacting Proteins
EAS	Elitist Ant System
FS-Weight	Functional Similarity Weight
GA	Genetic Algorithm
GFA	Graph Fragmentation Algorithm
GO	Gene Ontology
HPRD	Human Protein Reference Database
I2H	In Silico 2 Hybrid
IFC	Intuitionistic Fuzzy Clustering
MC	Monte Carlo
MCL	Markov Clustering
MCODE	Molecular Complex Detection
MD	Molecular Dynamics
MINT	Molecular Interaction Database

MIPS	Munich Information Center for Protein Sequences
MMAS	Min-Max Ant System
NNH	Nearest Neighbor Heuristic
PCP	Protein Complex Prediction
PFMD	Protein Functional Module Detection
PIN	Protein Interaction Network
PPI	Protein-Protein Interaction
PPV	Positive Predictive Value
RNSC	Restricted Neighborhood Search Clustering
SA	Simulated Annealing
SPC	Super Paramagnetic Clustering
STRING	Search Tool for The Retrieval of Interacting Genes
TSP	Traveling Salesman Problem
VRP	Vehicle Routing Problem
VWP	Vertex Weight Parameter
Y2H	Yeast Two-Hybrid

ALGORITMA PENGOPTIMUMAN KOLONI SEMUT BERASASKAN HEURISTIK BAGI MENGESAN MODUL KEFUNGSIAN PROTEIN DALAM RANGKAIAN INTERAKSI PROTEIN

ABSTRAK

Algoritma Pengoptimuman Semut (ACO) merupakan suatu metaheuristik yang telah sukses digunapakai terhadap beberapa jenis masalah pengoptimuman seperti penjadualan, pengarahan dan yang terkini untuk menyelesaikan masalah mengesan modul kefungsian protein (PFMD) di dalam rangkaian interaksi antara protein (PPI). Bagi data PPI bersaiz kecil, ACO telah digunapakai dengan sukses tetapi ia tidak sesuai untuk data PPI bersaiz besar dan bersifat kebisingan yang telah menyebabkan proses pencarian menjadi penumpuan pra-matang dan terhenti. Di dalam penyelidikan ini, bagi mengatasi keterbatasan tersebut, kami mencadangkan dua penambahbaikan yang baru terhadap ACO untuk menyelesaikan masalah PFMD. Pertama, kami menggabungkan ACO dengan heuristic jiran terhampir (diistilahkan ACOPFMD-NN) yang menggunakan senarai calon sebagai strategi pemilihan oleh kecerdikan semut apabila membina solusi. Kedua, kami menggunapakai konsep teori maklumat, geraf entropi yang digabungkan dengan ACO (diistilahkan ACOPFMD-IE) untuk menangani pemilihan laluan dengan mengawal dua perimeter ACO yang penting: jejak pati α dan maklumat heuristik β. Eksperimen ke atas set data berukuran piawaian emas "*Saccharomyces cerevisiae*" daripada dua pangkalan data yang popular DIP dan MIPS telah menunjukkan bahawa kedua-dua penambaikan kami telah meningkatkan prestasi terhadap algorithma ACO versi asal, dua algorithma metaheuristik terkini dan algorithma kebiasaan. Dari segi keputusan yang berbentuk kuantitatif, ACOPFMD-NN telah meningkatkan ketepatan sehingga 67% (DIP), 80% (MIPS) manakala

ACOPFMD-IE telah meningkatkan ketepatan sehingga 73.8% (DIP), 87.3% (MIPS). Dari segi keputusan yang berbentuk kualitatif, ACOPFMD-NN telah meningkatkan ketepatan sehingga 32% (DIP), 33% (MIPS) manakala ACOPFMD-IE telah meningkatkan ketepatan sehingga 69% (DIP), 59% (MIPS). ACOPFMD-IE juga memperolehi lebih ketepatan ke atas dua algorithma metaheuristik; 80% (DIP – yang dibandingkan dengan algorithma IGA), 74% (MIPS – yang dibandingkan dengan algorithma ABC-IF

HEURISTIC-BASED ANT COLONY OPTIMIZATION ALGORITHM FOR PROTEIN FUNCTIONAL MODULE DETECTION IN PROTEIN INTERACTION NETWORK

ABSTRACT

Ant colony optimization (ACO) is a metaheuristic algorithm that has been successfully applied to several types of optimization problems such as scheduling, routing, and more recently for solving protein functional module detection (PFMD) problem in protein-protein interaction (PPI) networks. For a small PPI data size, ACO has been successfully applied to but it is not suitable for large and noisy PPI data, which has caused to premature convergence and stagnation in the searching process. To cope with the aforementioned limitations, we propose two new enhancements of ACO to solve PFMD problem. First, we combine ACO with nearest neighbor heuristic (termed ACOPFMD-NN) that utilized the candidate lists as a selection strategy used by artificial ants when they construct the solution. Second, we apply the information theory concept, information entropy combined with ACO (termed ACOPFMD-IE) to handle the path selection by controlling two important parameters of the ACO; pheromone trail α and heuristic information β. The experiments on a gold standard benchmark dataset "*Saccharomyces cerevisiae*" from two popular databases DIP and MIPS has shown that our two enhancements have improved the performance of basic ACO, two recent metaheuristics and state-of-the-art of PFMD algorithms. In terms of quantitative results, ACOPFMD-NN has improved the accuracy up to 67% (DIP), 80% (MIPS) while ACOPFMD-IE has improved the accuracy up to 73.8% (DIP), 87.3% (MIPS). In terms of qualitative result, ACOPFMD-NN has improved the accuracy up to 32% (DIP), 33% (MIPS) while ACOPFMD-IE has improved the accuracy up to

69% (DIP), 59% (MIPS). ACOPFMD-IE has also obtained a better accuracy over two metaheuristics algorithms; 80% (DIP – compared with IGA algorithm), 74% (MIPS – compared with ABC-IFC algorithm).

CHAPTER 1

INTRODUCTION

1.1 Research Background

In the past decades, the rapid growth of genomic technologies and molecular biology fields has led a biologist to interpret, analyze and utilize that a huge amount of data has made the field of bioinformatics become more important. As an interdisciplinary field that involving biology, statistics, mathematics and computer science, bioinformatics aim to achieve faster and accurate results in performing those tasks (Cohen, 2004). Most of the bioinformatic tasks involve large data and they are formulated as hard combinatorial problems. Therefore, the implementation of metaheuristics and other approximate techniques is very useful compared to exact techniques (Blum & Roli, 2003; Gendreau & Potvin, 2010).

Defined as a top-level general strategy, metaheuristic guides other heuristics to find for better solutions. The main goal of metaheuristic is to explore the search space in efficient way for finding optimal solutions. In order to avoid getting trapped in local optimum, some mechanisms may also combined with metaheuristic (Marco & Stützle, 2010).

One of the more recent and actively studied in bioinformatics is proteomics, which is defined as a systematical study on proteins data that describes the functions, structure and the biological systems control in disease and health (Patterson & Aebersold, 2003). Scientifically, proteins rarely act as single isolated components; proteins that involved in similar cellular processes have interacted with each other to form a large molecule and the biological functions have been accomplished. For

example, the processes and activities of cellular signal transduction, metabolism, cell propagation and gene expression control depend on the interactions among proteins (Schwikowski, Uetz, & Fields, 2000). Therefore, the analysis of protein-protein interactions (PPI) network naturally serves as the basis for a better understanding of biological functions, cellular organization and processes (Graves & Haystead, 2002; Hartwell, Hopfield, Leibler, & Murray, 1999). One of the analysis tasks is the process of detecting protein functional modules (or protein clusters based on common functions) in the given PPI data.

Even though some metaheuristics algorithms have been developed to solve PFMD problem (Ji, Zhang, Liu, Quan, & Liu, 2014), however, each metaheuristic algorithm has its own strength and weakness. Therefore, many researchers have tried to design and develop the hybrid algorithms to improve the performances of PFMD. One of them, Ant Colony Optimization (ACO) has been applied to solve PFMD problem (Ji, Liu, Zhang, Jiao, & Liu, 2012; Shi & Zhang, 2011). Since PFMD problem is very complicated due to its large and noisy PPI data, the application of the basic ACO algorithm was facing problems such as premature convergence and stagnation. Like other combinatorial optimization problems such as Traveling Salesman Problem (TSP) and Vehicle Routing Problem (VRP), the basic ACO algorithm can effectively work on small size of data (Bullnheimer, Hartl, & Strauss, 1999a; Marco Dorigo & Gambardella, 1997). For tackling the large size of data, the developers have improved the basic ACO implementation by combining or hybridizing with other methods such combining ACO with local search for solving TSP (Bai, Yang, Chen, Hu, & Pan, 2013; Hlaing & Khine, 2011; Stützle & Hoos, 1999; Stützle & Hoos, 1996) and hybridizing ACO for VRP (Bullnheimer, Hartl, & Strauss, 1999b; X. Zhang & Tang, 2009).

Similar to these trends, this research intends to combine and hybrid the basic ACO with other methods to solve the large-scale PFMD problem.

1.2 Research Problem Statement

Research on protein functional modules detection (PFMD) in a protein interaction network has contributed a great understanding of biological functions and mechanism. Many computational approaches have acquired a large amount of PPI data, therefore the PFMD has presented significant challenges. In recent years, many computational methods have been proposed to solve PFMD problem based on certain models and hypotheses. However, with complex, huge and increasing volumes of PPI data, how to efficiently detect the protein functional modules become a vital scientific problem and an important research topic in the post-genomic era.

Recently, the ACO based algorithm has been applied to solve PFMD problem (Shi & Zhang, 2011). The design idea was based on taking the process of PFMD as the combinatorial optimization problem to solve the Traveling Salesman Problem (TSP). Based on the optimal tour constructed by artificial ants, a short distance between the proteins are likely to be a member in a protein functional module. This algorithm has been combined the topological weight with additional protein functional information from the Gene Ontology database but still apply the similar basic ACO solution rules for artificial ants to find the optimal paths (Ji et al., 2012).

The limitation of this solution was that not all proteins have the required information in the Gene Ontology database. This ACO algorithm solution has easily led to the premature convergence when applied to larger PPI data and has influenced

the clustering performances (Ji et al., 2014). Furthermore, their selection strategy based on the noisy PPI data that contain complex connection patterns in PPI network has caused the stagnation behavior of ants searching process. This phenomenon has limited the accuracy of predicted protein functional modules.

1.3 Research Goal and Objectives

The goal of this research is to overcome the limitation of the basic ACO algorithm for solving the PFMD problem, which is premature convergence and stagnation when dealing with large PPI and noisy data. In order to achieve this goal, several objectives are required to be met, as follows:

i. To enhance the ACO algorithm by inserting the nearest neighbor heuristic into the algorithm when dealing with large PPI data.

ii. To enhance the ACO algorithm in (i) by deploying information entropy into the algorithm when dealing with noisy data.

1.4 Research Scope

In this research, a well-studied yeast protein interaction data, *Saccharomyces cerevisiae* (*S.cerevisiae*) is used to evaluate the capability and the effectiveness of the proposed algorithms. There are two kinds of ACO-based methods for optimization process specifically for clustering. One is based on ACO algorithm inspired by behaviors of searching the shortest path by ant colonies from their nest to food source, called *foraging model*. The other one is inspired by the behavior of assembling the corpses and sorting the larvae by ant colonies, called *Piling Model*. The optimization

process for PFMD will only focused on searching the shortest path, inspired by *foraging* model. Therefore, the comparison analysis will not be done with those using *Piling Model*.

1.5 Research Significance

The significances of this research can be addressed in terms of two contributions. The first contribution is for the computer science knowledge, which is the design and development of an improved ACO-based algorithm for the PFMD optimization problem. The proposed algorithm utilized the ACO algorithm, Nearest Neighbor heuristic and information entropy in improving PFMD problem.

Secondly, the outcome of this research is beneficial to the system biology community because this research has proposed a method that produce proteins functional can help the biology community to use this data for designing medicine, drug etc. via computer simulation without involving laboratory work. As a conclusion, the improved ACO-based method proposed in this research enables scientists in the field of computer science to produce computational tools to help and facilitate the system biology community in producing an accurate design of medicine and drugs.

1.6 Thesis Organization

The structure of the thesis is outlined as follows:

Chapter 1 presents the introduction of this research, which encompasses the problem statement, goal, objectives, scope, and significance of the research.

Chapter 2 discusses the literature review of the research, which covers the PPI networks, PFMD, ACO, and other related metaheuristics.

Chapter 3 provides the research methodology. The research methodology covers the PFMD and ACO research framework, description of the experimental data and the evaluation metrics used.

Chapter 4 describes the enhancement of the ACO-based algorithm for PFMD problem that involves Nearest Neighbor heuristic for solving large PPI data.

Chapter 5 describes the enhancement of the proposed algorithm in Chapter 4, by deploying information entropy for solving large and noisy PPI data.

Chapter 6 concludes the study and presents the contributions. The future work of the study is also discussed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Recently, the vast amount of the generated PPI data have provided the great opportunity to analyze a large living system in a systematic way. The essential principles such as protein complexes, cellular pathways, genetic interactions and protein functional modules also can be understood when using these PPI data (Bolin, Weiwei, Juan, & Fang-Xiang, 2014; Chen & Yuan, 2006a; Cho, Hwang, Ramanathan, & Zhang, 2007). Therefore, the PFMD research based on PPI networks was considered quite active because the unknown functional correlation between proteins can be revealed and the unknown function for protein can be predicted. (Dittrich, Klau, Rosenwald, Dandekar, & Müller, 2008). Currently, a number of metaheuristics algorithms have been proposed to solve PFMD problem based on protein interaction network and one of them is ACO, which has been applied to DIP datasets.

In this chapter, a literature review that related to this research is presented. It contains the background knowledge and related work for PFMD problem and the survey of ACO algorithms as well as the related metaheuristics. In this review, we first discuss the PPI network and protein functional modules. Second, the general framework for solving PFMD and related methods is presented. Third, we discuss the ACO algorithms and its application to various problems. Fourth, we present the concept of an uncertain graph, information entropy. Finally, we discuss the research trends and directions from the computational viewpoint before this chapter is summarized.

2.2 PPI Data, PPI Network and Protein Functional Module Detection

The large-scale PPI datasets have been produced by high-throughput profiling techniques such as yeast two-hybrid (Y2H) (Ito et al., 2001), tandem affinity purification (TAP)(Puig et al., 2001), mass spectrometry (Gavin et al., 2002), phage display (Willats, 2002), pull down assays (Vikis & Guan, 2004) and microarrays (Stoll, Templin, Bachmann, & Joos, 2005). Most of the information about PPI data was an organism specific and already available in a variety of large PPI databases. Proteins in these databases are functionally classified using well established functional catalogue FunCat (Ruepp et al., 2004). Table 2.1 summarizes some well-known public PPI databases.

Based on these PPI data, the interaction among proteins data can be represented in a network-fashion called protein-protein interaction (PPI) network (Rahman, Islam, Chowdhury, & Karim, 2013). Mathematically, a protein interaction network is often modeled as an edge-weighted undirected graph where each vertex denotes a protein and each edge represents an interaction between a pair of proteins (Gabr, Dobra, & Kahveci; Pavlopoulos et al., 2011).

Table 2.1: Public PPI Databases

Abbreviation	Full Name	Author/Developer	Year	URL
BioGRID	General Repository for interaction Datasets	(Stark et al., 2006)	2006	http://www.thebiogrid.org
DIP	Database of Interacting Proteins	(Xenarios et al., 2002)	2004	http://dip.doe-mbl.ucla.edu
BIND	The Biomolecular Interaction Network Database	(Gary D Bader, Betel, & Hogue, 2003)	2005	http://bind.ca
MIPS	The MIPS (Munich Information Center for Protein Sequences)	(Pagel et al., 2005)	2005	http://mips.gsf.de/services/ppi
HPRD	The Human Protein Reference Database	(Mishra et al., 2006)	2006	http://www.hprd.org
MINT	Molecular INTeraction Database	(Chatr-Aryamontri et al., 2006)	2007	http://mint.bio.uniroma2.it/mint
IntAct	Protein InterAction Database	(Kerrien et al., 2006)	2007	http://www.ebi.ac.uk/intact

Figure 2.1, 2.2 and 2.3 illustrate the scientific, schematic and large view of protein interaction networks, accordingly. Systematic analysis of the large-scale PPI data based on their graph representations has the potential to yield a better understanding of protein functions computationally (De Las Rivas & Fontanillo, 2010). One way to chart out the underlying cellular functional organization is to detect protein functional modules in these networks by grouping the proteins sharing similar biological functions into the same modules (Navlakha, Schatz, & Kingsford, 2009; Nepusz, Yu, & Paccanaro, 2012; Pinkert, Schultz, & Reichardt, 2010; Royer, Reimann, Andreopoulos, & Schroeder, 2008).

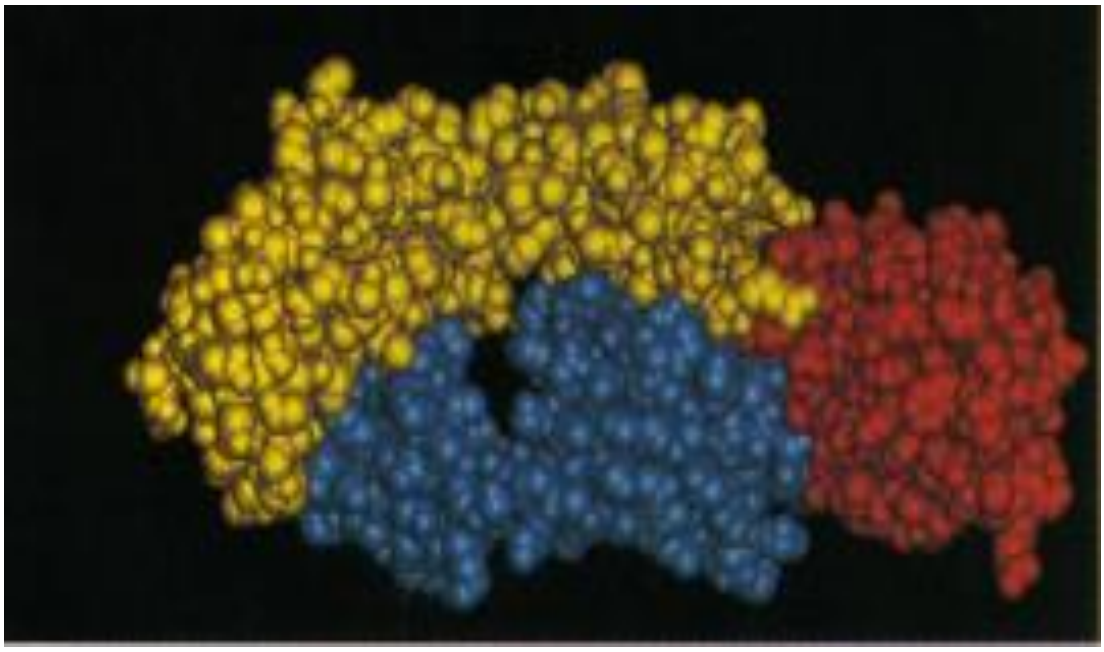

Figure 2.1: Scientific view of PPI (Jones & Thornton, 1996)

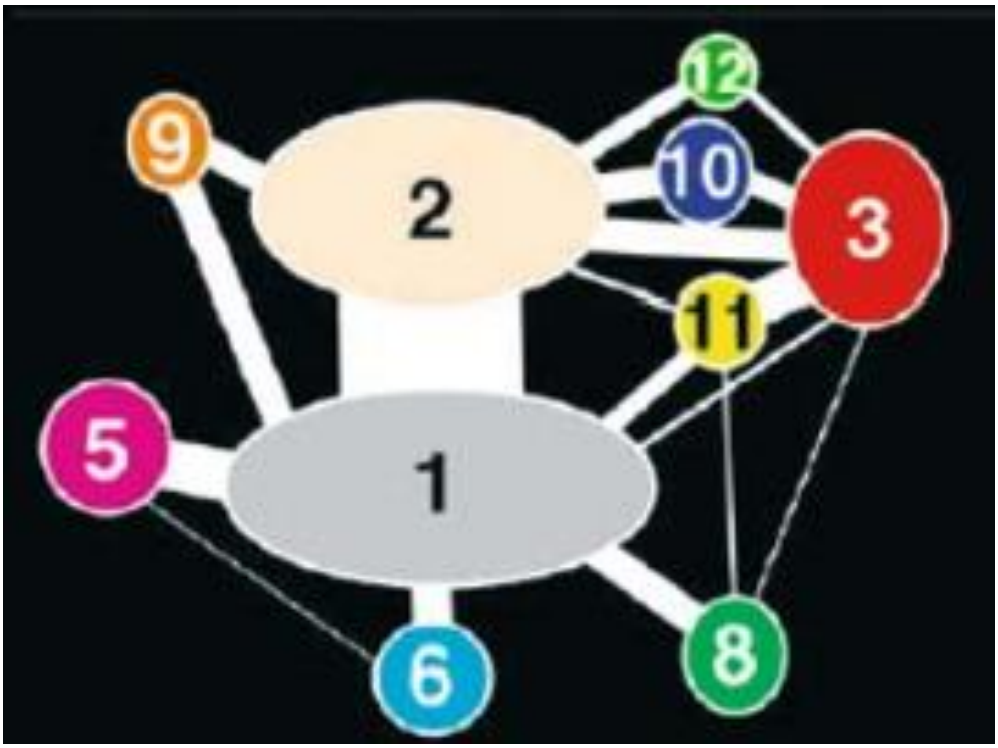

Figure 2.2: Schematic PPI view for the 10 subunits (Uetz & Vollert, 2005)

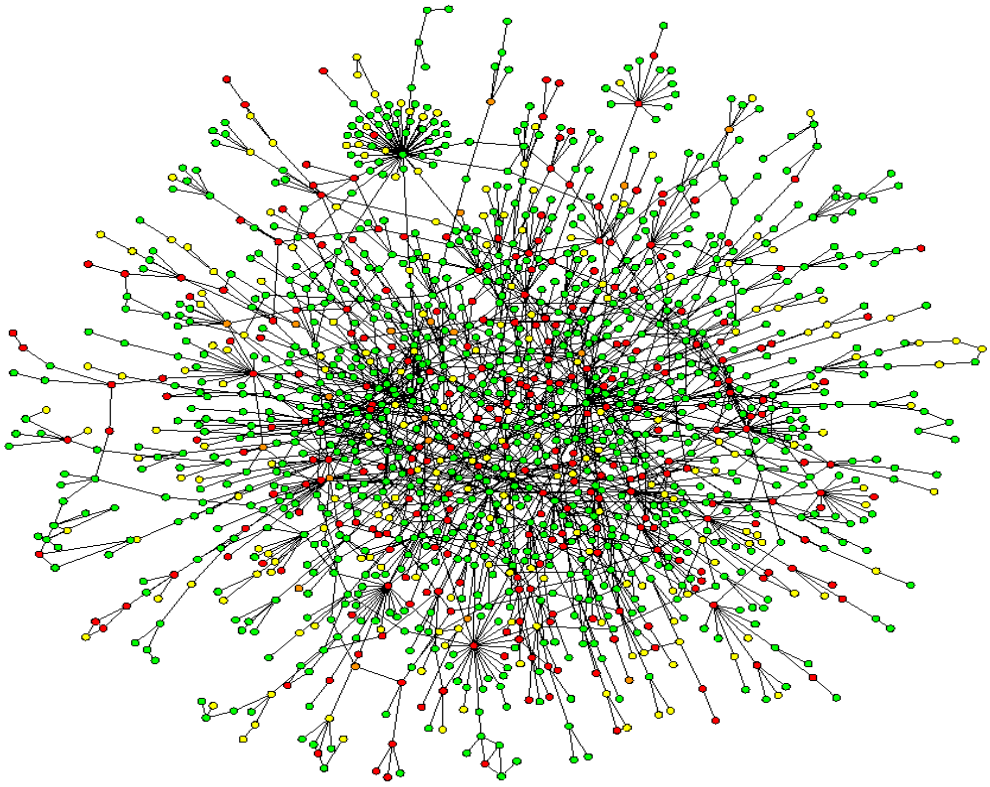

Figure 2.3: Large protein interaction network (PBworks, 2007)

In a standard definition, a protein functional module is defined as "*a group of proteins that participate in the same biological process or perform the same molecular function while binding each other even at a different time and place*" (Chen & Yuan, 2006b). Most proteins have formed the macromolecular complexes to execute their biological functions, and there are still a large number of protein functional modules undiscovered yet (Y. Wang & Qian, 2013). However, the experimental data from the high-throughput technologies have provided biologists an opportunity to detect possible protein functional modules through clustering a protein interaction network (Asur, Ucar, & Parthasarathy, 2007; Gao, Sun, & Song, 2009; Lin, Cho, Hwang, Pei, & Zhang, 2007). The detection of these modules, known as protein functional module detection (PFMD) is an area of active research and become very important to understand the fundamental function and structure of PPI networks. Therefore, many computational approaches have been developed (Gao et al., 2009; Srihari & Leong, 2013) to solve PFMD problem and facilitated the researchers to gain better understanding about the PPI networks in terms topological structure and relationships among proteins.

2.3 The General Framework of PFMD Processes

Generally, a complete PFMD process is composed of four steps: PFMD problem modelling, data pre-processing, clustering, and post-processing (Dittrich et al., 2008; Ji et al., 2014; Z. Wu, Zhao, & Chen, 2009). We illustrate the general framework used for PFMD process in Figure 2.4. The detail explanation of this figure is discussed in the following sub-sections.

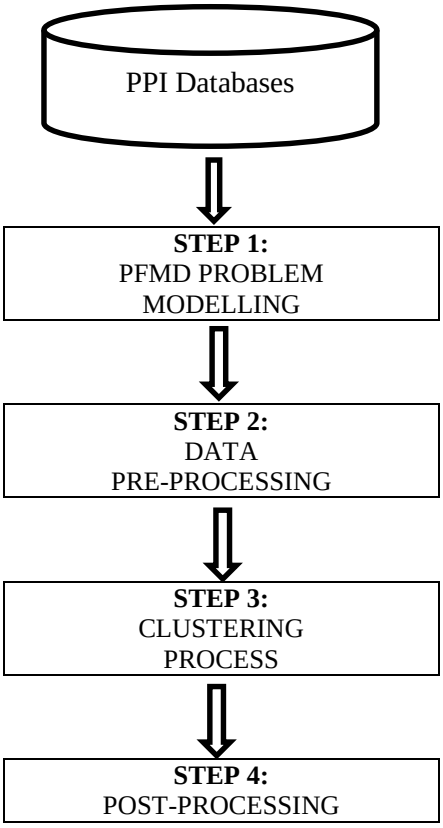

Figure 2.4: The General Framework Used for PFMD process.

2.3.1 Step 1: PFMD Problem Modeling

The modelling of PFMD problem is very important to gain a better understanding of the function and structure of the PPI network. In general, PFMD problem modeling is a task representing the clustering process in a PPI network using a mathematical model and can be categorized into three approaches: Cliques (Luce & Perry, 1949), *k*-core (Seidman, 1983; Seidman & Foster, 1978) and Distance (Shortest Paths)-Based Index (Consul & Jain, 1973). The following subsections discuss these models.

2.3.1.(a) Clique Model

A clique model was introduced by Luce and Perry (1949) and has been applied to perform social networks modelling. A clique is generally defined as an induced complete subgraph within a graph, with essential vertices that are entirely connected to each other. The edges are shared among all members of a clique. In a graph $G = (V, E)$, a clique C is considered a maximal clique if and only if there is no clique C in G with $C \subset C'$. In other words, a maximal clique is a complete subgraph that is not confined within any other complete subgraph. From the algorithmic view, the maximal cliques detection in a graph was considered as an NP-complete problem (Karp, 1972). Some methods (Adamcsek, Palla, & Farkas, 2006; B. Chen, Shi, Zhang, & Wu, 2013; Jianxin, Zhao, & Min, 2008; X.-L. Li, Foo, Tan, & Ng, 2005; X. L. Li, Tan, & Foo, 2005; Xiong, He, & Ding, 2005; S. Zhang, Ning, & Zhang, 2006) have utilized the Clique model to solve PFMD problem, however, the incompleteness PPI data and the sparse protein connected in the PPI network has limited the utilization of the clique model.

2.3.1.(b) *k*-core Model

Seidman and Foster (1978) has introduced the *k*-core model and Bollobás (1984) has utilized this model for network analysis and visualization purpose. In the protein interaction network, a *k*-core is a subgraph, which each protein is associated with at least *k* proteins of this subgraph. *k*-core was defined by Batagelj and Zaveršnik (2002) as follows. In a graph $G = (V,E)$, The formation of the *k*-core is by removing all vertices and their occurrence edges with degrees are less than *k*. Most existing PFMD methods mainly focus on detecting highly connected subgraphs in protein interaction networks as protein functional modules but their inherent organization has been ignored. However, scientific experiments that detected protein functional modules recently have discovered their inherent organization. In other words, a protein functional module generally contains a core, such that the proteins are highly co-expressed and highly functional similarity is shared, and some proteins were often attached surround the core (Gavin et al., 2006). Recently, some methods based on a *k*-model are developed (Leung, Xiang, Yiu, & Chin, 2009; Lubovac, Gamalielsson, & Olsson, 2006; Ulitsky & Shamir, 2009; M. Wu, Li, Kwoh, & Ng, 2009). Based on the survey by (Ji et al., 2014), the PFMD methods using *k*-core model have demonstrated very well matching with existing biological knowledge. However, the limitation of this model is when working on large and dense networks, which discard so many important proteins.

2.3.1.(c) Distance (Shortest Paths)-Based Index Model

Some models find a subnetwork based on the distance between vertices. The first molecular graph distance-based model is Wiener index model, *W* (Wiener, 1947). The

mathematical formula for this model consists of a simple summation of weight (distance) between all vertex pairs, as follows:

$$W(G) = \sum_{u \neq v \in V} dist(u, v). \tag{2.1}$$

When the total of distance decreases, the density of a graph *G* will increase. In (Wiener, 1947), Wiener had analyzed the total distances of a molecular graph of the molecule that revealed similarities between the subgraphs. In a graph *G*, the average path length *APL*(*G*) is the average of the shortest paths length between all vertex pairs:

$$APL(G) = \frac{\sum_{u \neq v \in V} dist(u, v)}{\frac{1}{2}(n^2 - n)}. \tag{2.2}$$

Since the shortest paths model is only well defined for the connected vertex pairs, therefore for the disconnected vertices, this model requires management of those disconnected vertices to suit with the semantics of each application. There are many PFMD methods such as (G. D. Bader & Hogue, 2003; A. J. Enright, S. Van Dongen, & C. A. Ouzounis, 2002; Hwang, Cho, & Zhang, 2006, 2008; Ji et al., 2012; A. D. King, N. Pržulj, & I. Jurisica, 2004; Lei, Wu, Tian, & Zhao, 2014; Pizzuti & Rombo, 2014; Ravaee, Masoudi-Nejad, Omidi, & Moeini, 2010; Shi & Zhang, 2011; Shuang, Xiujuan, & Jianfang, 2011) that utilize distance (shortest path)-based model, which totally depends on the weight (distance) between two proteins in PPI networks. In other words, if any PFMD method wants to utilize this model, the topological distance or the similarity measurement between two proteins should have high reliability to ensure the detected protein functional modules to be biologically meaningful.

In summary, when comparing these three approaches for PFMD, Distance (Shortest Path)-based Index Model has the advantage of working with any type of the structure protein interactions networks, either sparse or dense network. As discussed, Clique and *k*-core models are having limitations when utilized into dense and sparse protein interaction networks, which causing many important proteins have been discarded.

2.3.2 Step 2: PPI Data Pre-Processing

A comprehensive PPI data that provided by many open databases for several different organisms has given the availability for PFMD process. PPI data from different databases and from different research institutes have a unique format, mode of description and data structure. Therefore, the standardization process has been done however, the unified benchmark is still not available because the development of these databases using different computational approaches. DIP is one of the earliest and the most commonly databases used in PMFD research (Pizzuti & Rombo, 2014; Rao, Srinivas, Sujini, & Kumar, 2014).

There are many ways of data pre-processing tasks, which depend on the requirement of clustering algorithms (Ucar, Parthasarathy, Asur, & Chao, 2005). For example, distance-based clustering algorithms require the inter-proteins distance (weight) in PPI networks. Several methods have been proposed to calculate the inter-protein distances and will be used to detect the protein functional modules in PPI networks. (Glazko, Gordon, & Mushegian, 2005; Gursoy, Keskin, & Nussinov, 2008; Jain & Bader, 2010; Lord, Stevens, Brass, & Goble, 2003; Pei & Zhang, 2005; Schlicker & Albrecht, 2008).

Coefficient formula is the simplest distance computation method for two interacting proteins in a PPI networks. For example, Glazko et al. (2005) has used the topological properties in PPI networks by considering that the more two interacting proteins share common interacting partners, the more these two proteins functionally related (Pržulj, Wigle, & Jurisica, 2004; Spirin & Mirny, 2003; J. Wang, Li, Deng, & Pan, 2010; Yook, Oltvai, & Barabási, 2004). This principle has been utilized by Brun, Herrmann, and Guénoche (2004) with proposing the Czekanowski-Dice distance calculation for interacting proteins in PPI networks.

2.3.3 Step 3: Clustering Process

The most important and essential step in PFMD is the clustering process. This is because the PFMD problem formulation and post-processing step is determined in a specific ways based on the clustering tasks. The specific clustering algorithms for solving PFMD will be discussed in Section 2.4.

2.3.4 Step 4: Post-processing

There are few approaches to perform post-processing, depends on how the algorithm works. In this research, we survey the post-processing that related to our work. The optimal tour (the best solution) is obtained after a number of iterations, and those proteins in the list of optimal tour will be used for the PFMD problem. Each path between two proteins on the optimal tour has a different distance. Different protein functional modules in PPI networks are connected by longer distance in the path. Therefore, preliminary clusters could be generated by removing those longer distance paths to form the preliminary modules. To do that, we need to disconnect some paths with long distance by using a cut-off value, $\hbar$ is defined as follows:

$$\text{⋔} = \sigma \cdot d_{ave} \tag{2.3}$$

where

σ = a real parameter, only will be used if there is negative influence.

d_{ave} = average distance between proteins in all paths.

The corresponding path will be cut if a distance d between proteins in the optimal tour is bigger than ⋔, which form the preliminary modules. Next, these preliminary modules will be merged and to do that, we deploy the following formula:

$$S(M_1, M_2) = \frac{\sum_{i \in M_1, j \in M_2} s\,(i,j)}{min(|M_1|, |M_2|)}, \tag{2.4}$$

where

$$s(i,j) = \begin{cases} 1, & \text{if } \ i = j\,. \\ f_{ij,} & \text{if } \ i \neq j \ and \ \langle i,j \rangle \in E \\ 0, & otherwise \end{cases} \tag{2.5}$$

and $S(M_1, M_2)$ is the similarity score between protein functional module 1 and 2.

The merging processes are iteratively done by using Formula 2.4 and stop when the highest similarity score is not greater than merging threshold value. The filtering step will be done if the connected proteins are too sparse or the number of formed modules that contain protein members are too small (for instance, less than 3 members), by measuring the detected modules in terms of its topological density. The topological density is measured by:

$$D_M = \frac{e}{V\,.\,(V-1)/2} \tag{2.6}$$

where

V = number of vertices (proteins),

e = number of edges (interactions)

If $D_M < \delta$, the final clusters will be formed.

2.4 Algorithms for Solving PFMD

In this section, we review the PFMD algorithms that have been developed in recent years to solve PFMD problem. For this research, we categorize the PFMD algorithms into two types: state-of-the-art and metaheuristic algorithms. In a standard practice, most of the researchers will benchmark their experimental results with the state-of-the-art PFMD algorithms. Moreover, since our proposed ACO-based is one of metaheuristic algorithms, this section will only review the related algorithms that are related to our research.

2.4.1 State-of-the-art PFMD Algorithms

The first state-of-the-art algorithm applied to PFMD problem is Markov clustering (MCL). It is a flow simulation graph clustering developed by Dongen (2000). The MCL application software, TRIBE-MCL has been developed by Anton J Enright, Stijn Van Dongen, and Christos A Ouzounis (2002) to solve PFMD problem. TRIBE-MCL has used the Markov matrices (Krenk & Gluver, 1989) which simulate the random walks through the PPI network graph. There are two operators involved in the random walks: expansion and inflation where these two operators promote the flow of walking. The stronger flow will be captured and the weaker flow will be removed. MCL algorithm has been proven by many researches that it has a very good robustness however, based on the recent review (Lei et al., 2014), it has been observed that the MCL has obtained low precision and recall even though it is suitable for PPI networks.

Bader and Hogue (2003) have introduced MCODE (Molecular complex detection) algorithm. First, every protein vertex in PPI networks will be assigned a weight by calculating their density of local neighbor. Then the vertices with high

weights of will be taken as the vertices seed of initial clusters and the preliminary protein functional modules will be formed based on the further augments. MCODE has two post-processing steps where it filters the non-dense subgraphs and generates the overlapping clusters for the final protein functional modules formation. The advantage of MCODE is that it can generate overlapping protein functional modules. However, MCODE has detected a small the quantity of the protein functional modules even though it is applied in some large protein functional modules. Another drawback of MCODE is the capability is not guaranteed to detect the protein functional modules that must be the close connection in PPI networks.

King, Natasha, and Igor (2004) have developed the Restricted Neighborhood Search Clustering (RNSC) algorithm that combine the Gene Ontology and topological information to solve PFMD. There are two steps in this algorithm: Firstly, it starts with clustering PPI network based on the functional homogeneity and cluster properties. The PPI network is initialized with the random separated protein into different sub networks and it is an important partition of the vertex *V*. Secondly, the vertex moves from one cluster to the next cluster randomly and stops when the value for cost function is optimized. A drawback of RNSC algorithm is it has high potential to be trapped in the local minima solution.

2.4.2 Metaheuristic PFMD Algorithms

Olin and Liu (2006) have utilized the idea of classical Traveling Salesman Problem to study the concept of global optimization and used greedy technique to cluster the yeast proteins based on the global protein interaction information. The major drawbacks of their implementation is the calculation of distance (weight) for PPI network graph is

based on binary interaction matrix that has produced so many identical distances. Therefore, many proteins have been discarded during the searching process.

Ulitsky and Shamir (2009) also use a greedy algorithm to solve PFMD problem by optimizing the initial seeds. A tour is gradually constructed by the Greedy heuristic which repeatedly selects the shortest edge and adds the selected edge to the tour until the selecting cycle is not greater than *N* edges. The advantage of greedy algorithm is very simple to be implemented for solving PFMD (He, Li, Ye, & Zhong, 2012a; Saiyed, 2012). Later, He, Li, Ye, and Zhong (2012b) proposed a Greedy Search Method based on Core-Attachment structure (GSM-CA) that detects the dense subnetwork in large PPI networks. However, the sparse PPI network has limited the implementation of this algorithm.

Based on efficient vaccination approach, Ravaee et al. (2010) have introduced an immune genetic algorithm to search subnetworks in PPI networks, termed IGA. It is defined as a schema of variable-length antibody and the new mutations of global and local. A new selection strategy is used based on the involvement of scientific antibodies' probability reproduction in the population. This selection strategy is applied in the antibody cloning procedure to preserve the outstanding antibodies. In addition, an efficient objective function is declared for each antibody evaluation. Their results have outperformed other well-known dense-based approaches such as MCODE. However, the complexity of preprocessing steps has made this approach become more complicated.

Shuang et al. (2011) have proposed the Artificial Bee Colony (ABC) algorithm, a population-based clustering method to solve PFMD problem. In the data pre-processing step, the number of protein functional modules is determined and the noise spots is eliminated by specific algorithm. Next, the cluster centers are determined based on the vertices clustering coefficient and the cluster centers are taken as the sources of food. The searching capability of ABC has greatly improved the performances of original algorithm for functional flow clustering. The limitation of ABC is it has a difficulty in the setting of initial parameters in random optimization process.

A hybridization of Artificial Bee Colony (ABC) with intuitionistic fuzzy clustering (IFC) has been proposed by Lei et al. (2014). It has two major steps: Firstly, the ABC mechanism is used to search the optimum cluster centers and these clusters is set randomly. Secondly, IFC uses fuzzy membership matrix to form the cluster. Artificial bees with different functions update the cluster centers based on the new optimized cluster center. The ABC-IFC has greatly improved the performances of original ABC algorithm however, the number of initial clusters that are required by ABC-IFC has made the process of searching become more complex.

An ACO-based algorithm for solving PFMD has been proposed by Ji et al. (2012), which deploys a basic ACO probabilistic formula which similar to greedy technique style. In general, for each iteration, each ant constructs one tour through all protein interaction networks. Starting from a random protein, then the ant proceeds to next protein from current position until the tour is complete by returning to starting protein. The PPI graph weightage is solely based on the topology of protein interaction

networks. A short distance among proteins in the optimal path searched by artificial ants can be grouped together using a determined threshold to form a protein functional module. Based on the experimental results, ACO has outperformed several existing PFMD algorithms. However, the way ACO work during searching solutions is similar to greedy technique that causes to stagnation and premature convergence. This phenomenon has influenced the results of detected protein functional modules.

2.5 Heuristic and Metaheuristic Algorithms for Traveling Salesman Problem

One of the well-known combinatorial optimization problems is traveling salesman problem (TSP), which is generally considered as a typical example of a very hard combinatorial optimization problem (Colorni et al., 1996; Gilmore, Lawler, Shmoys, & Lawler, 1986). Generally, TSP is modeled as an undirected weighted graph, where the cities represent the vertices of graph, the paths represent the edges of graph and the distance of inter-cities represent the length of the edges. The problem of TSP is defined as follows: Starting from one city, the salesman has to visit all cities only once and returns back to the starting city with minimum total distance (Gilmore et al., 1986; Saiyed, 2012).

Nilsson (2003) has evaluated some heuristics algorithms used to find the optimal tours in TSP. Based on his observation, the nearest neighbor heuristics (NNH) is the simplest and fastest heuristic to be applied for TSP. NNH starts with an arbitrarily chosen city c_1 as partial tour. It is a constructive algorithm for the TSP using nearest-neighbor procedure, which treats the cities as components. The procedure works by randomly choosing an initial city and by iteratively adding the closest among the remaining cities to the solution under construction (ties are broken randomly). The