Yan *et al. Proteome Science* 2011, **9**(Suppl 1):S17
http://www.proteomesci.com/content/9/S1/S17

PROTEOME SCIENCE

PROCEEDINGS Open Access

Applications of graph theory in protein structure identification

Yan Yan[1,2], Shenggui Zhang[1], Fang-Xiang Wu[2*]

From International Workshop on Computational Proteomics
Hong Kong, China. 18-21 December 2010

Abstract

There is a growing interest in the identification of proteins on the proteome wide scale. Among different kinds of protein structure identification methods, graph-theoretic methods are very sharp ones. Due to their lower costs, higher effectiveness and many other advantages, they have drawn more and more researchers' attention nowadays. Specifically, graph-theoretic methods have been widely used in homology identification, side-chain cluster identification, peptide sequencing and so on. This paper reviews several methods in solving protein structure identification problems using graph theory. We mainly introduce classical methods and mathematical models including homology modeling based on clique finding, identification of side-chain clusters in protein structures upon graph spectrum, and *de novo* peptide sequencing via tandem mass spectrometry using the spectrum graph model. In addition, concluding remarks and future priorities of each method are given.

Background

Protein structure identification is a central research area in proteomics [1]. Proteins, as we know, are complex organic compounds, which consist of series of amino acids. Protein structures are usually considered as four different levels from amino acids sequences to various folding patterns. They are very important in proteomics since they usually determine the function, homology and other features of proteins. Therefore, increasing number of researchers are focusing on protein structure identification problems. Usually, biological experiments for identifying protein structures produce huge quantity of data. Facing these molecular biology data, researchers aim to find perspective relationships of proteins through effective analyzing and then, focusing on further biological relationships and functions of them [2]. In order to deal with these, biological ways have been used at first time. However, due to various limitations such as strict environment request and high experiment cost, these methods have encountered tough difficulties. Mathematical methods, by contrast, are effective in summarizing and predicting biological characteristics with lower cost, which are drawing increasing attention and being widely used in this area. Among different kinds of mathematical methods, graph theory is an essential one [3], which owns advantages in various protein structure identification problems including predicting protein structure, identification of side-chain clusters in protein structures, *de novo* sequencing, and so on [4,5].

In this paper, we summarize current applications and development of graph theory modeling in protein identification, mainly introducing three classical methods and mathematical models including homology modeling based on clique finding, identification of side-chain clusters in protein structures upon graph spectrum, and *de novo* peptide sequencing via tandem mass spectrometry using the spectrum graph model. Besides, we briefly analyze the advantages and disadvantages of these methods and give some possible directions for future research.

Review

Basic knowledge of graph theory

In order to understand the problem modeling, we need to know some basic concepts and background knowledge in graph theory. A *graph* G is an ordered pair (V

* Correspondence: faw341@mail.usask.ca
[2]Division of Biomedical Engineering, University of Saskatchewan, Saskatoon, SK S7N 5A9, Canada
Full list of author information is available at the end of the article

(G), $E(G)$) consisting of a set $V(G)$ of *vertices* and a set $E(G)$, disjoint from $V(G)$, of *edges*, together with an *incident function* ψ_G that associates with each edge of G an unordered pair of vertices (not necessary distinct), if e is an edge and u and v are vertices such that ψG(e) ={u, v}, then the edge e is said to *join* the vertices u and v, and u and v are called the *ends* of e[6]. We denote the numbers of vertices and edges in G by $v(G)$ and $e(G)$, which are called the *order* and *size* of G, respectively. In this paper, we always use G to represent a graph we are concerning.

The following is an example of a graph to clarify the definition. For notational simplicity, we use uv for the unordered pair $\{u,v\}$. Let $G = (V(G), E(G))$, where $V(G) = \{u, v, w, x, y\}$, $E(G) = \{a, b, c, d, e, f, g, h\}$. The function ψ_G is defined as: $\psi_G(a) = uv$, $\psi_G(b) = uu$, $\psi_G(c) = vw$, $\psi_G(d) = wx$, $\psi_G(e) = vx$, $\psi_G(f) = wx$, $\psi_G(g) = ux$, $\psi_G(h) = xy$. The graph G could be drawn as in Figure 1.

An edge with identical ends is called a *loop*, and an edge with distinct ends a *link* . Two or more links with the same pair of ends are said to be *parallel edges*. A graph is *simple* if it has no loops or parallel edges. In this paper, all the graphs we concern are simple graphs.

A *complete graph* is a simple graph in which any two vertices are adjacent, an *empty graph* one in which no two vertices are adjacent (that is, one whose edge set is empty). A *path* is a simple graph whose vertices can be arranged in a linear sequence in such a way that two vertices are adjacent if they are consecutive in the sequence, and are nonadjacent otherwise. The *length* of a path is the number of its edges. In a graph G, the *degree* of a vertex v, denoted by $d_G(v)$, is the number of edges of G incident with v, each loop counting as two edges. The set of all vertices incident with v is denoted by $N_G(v)$ [6].

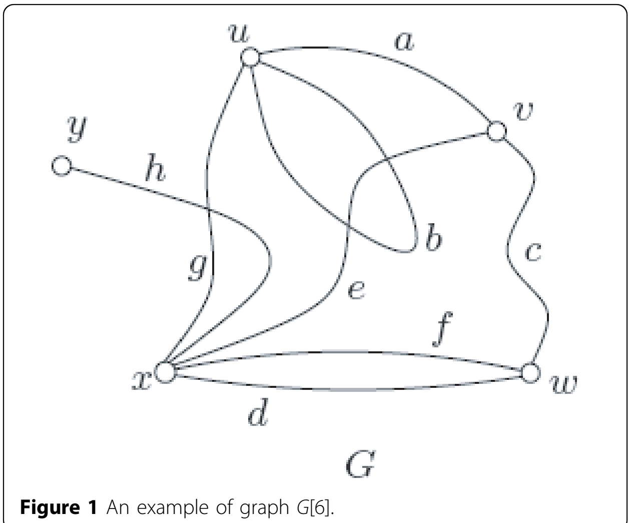

Figure 1 An example of graph G[6].

In a graph, a *clique* is a set of mutually adjacent vertices, in other words, a subset of $V(G)$ that has completely connected vertices. So in a clique, arbitrarily choosing two vertices, they are connected with each other. A clique in a graph is *maximum* if the graph contains no larger cliques. If a subgraph S in a graph G is a clique, then the clique center is a vertex v in S satisfying that, $\forall u \in V(S) \setminus v$, $maxd(u, v)$ is minimal. The clique center is *weighted* if G is weighted in calculating distance.

Adjacency matrix of a graph G is the $n \times n$ matrix $A_G := (a_{uv})$, where a_{uv} is the number of edges joining vertices u and v. Each loop is counted as two edges [6]. A set of points in space can be represented in the form of a graph where the points represent the vertices of the graph and the distances between the points represent edges. The constructed graph can be represented mathematically in the form of a matrix called the *Laplacian matrix*[7]. *Graph spectrum* is the information on analyzing the eigenvalues and eigenvectors related to Laplacian matrix in the graph spectrum research. It can gain information on cliques and clique centers in the graph.

Construction of homology modeling upon best-weight clique finding

Problem description

Homology modeling is a key aspect in preteome study. When we say that sequence A has high homology to sequence B, we claim that not only sequence A looks much the same as sequence B, but also all of their ancestors look the same, going all the way back to a common ancestor [8]. Identification of homological sequences enables us to assign information from one known sequence to another unknown sequence, which enables to save lots of time and energy in research, too. However, homology modeling is facing many difficulties nowadays. One problem is that it is usually hard to find acceptable conformations of proteins because many conformations are highly dependent on experiment environment which would definitely limit the experiment design. Another problem is that there is no much effective algorithm available to cope with biological methods. Therefore, researchers are thinking of different mathematical approaches to solve these problems. Among them, the graph-theoretic method is a typical one. In this section, we will introduce a graph-theoretic method that constructs homology modeling upon best-weight clique finding. We first introduce some concepts, followed by modeling process, and then evaluate this method, giving some future research directions at last.

Homology modeling, also known as comparative modeling of proteins, is a technique that identifies approximate structure of a target protein from a related known homologous protein. When the target sequence is closely related to some known sequence, their overall folds are similar [9], so we can reconstruct the structure of target protein (from sequence) if we recognize its folding way by the known protein.

The steps of *homology modeling* can be arranged as follows. First, identifying an alignment between the target and related protein sequences [10]. Second, copying the main-chain coordinates from the related protein for equivalent residues and inferring some side-chain conformations. Last, building other structures left. In this procedure, current numerical methods encounter difficulties because it is hard to find suitable models [11-16]. A good model should not only satisfies the polypeptide chain property that steric exclusive effect makes energy surface discontinuous and that the conformation is context-dependent, but also has effective algorithms in implementing. Here, a graph-theoretic method can be applied to solve this problem well [17].

Graph-theoretic modeling

In 1998, Samudrala and Moult transferred homology modeling into a clique finding problem in graph theory and used an effective algorithm to solve it [18]. The vertices and edges of the graph are defined as follows.

Vertex: Each possible conformation of an amino acid residue in the sequence stands for a vertex in the graph. The *weight* of the vertex depends on interaction strength between local main-chain atoms and side-chain atoms. The main-chain atoms up to four residues on each side of the residue position, and the main-chain atoms of this residue, should be considered to calculate the weight.

Edge: Edges would be drawn when vertices present residue conformations within the same main-chain segment but not between clash atoms or different possible side-chain conformations of the same residue. The *weight* of an edge stands for interaction strength between two differen vertices (which represent residues).

Once the qualified graph has been drawn, all the maximal sets of cliques can be found using a clique finding algorithm [19,20]. Here, we propose an algorithm developed by Bron and Kerbosch [21].

This algorithm uses a recursive backtracking procedure and a branch-bound technique to achieve quick time clique finding [22]. There are three sets that play key roles in the algorithm: (1) *potential clique;* in this set, all the vertices are connected to each other, so this set can be extended by some new qualified vertices and has the potential to be the maximal clique. (2) *candidates;* this set consists of the vertices that can be added into the *potential clique* set. (3) *not;* this is a set of vertices that not belong to either of the former two sets, which means that the vertex has already served as an extension to the current *potential clique* set but not qualified.

At the beginning of the algorithm, *potential clique* and *not* are both empty while *candidates* consists of all the vertices of graph G, which represents all the possible conformations and their interactions. After that, choosing vertex v in *candidates* with maximal degree to the *potential clique* set. This kind of strategy makes larger cliques being found faster. Then, the vertices in *candidates* should be the vertices connected to v, and the vertices in *not* be the vertices disconnected to v. After that, choosing vertex u with maximal degree in the current *candidates* set, and repeating the procedure till the *candidates* set is empty. The procedure can also be written as the following steps. We use P, C, N to represent the sets *potential clique*, *candidates*, and *not*, respectively.

step 1: Set $C = V(G)$, $P = \emptyset$, $N = \emptyset$;

step 2: If $C \neq \emptyset$, calculate $d(v) = \max_{u \in C} d(u)$, go to step 3; else go to step 4.

step 3:$P = P \cup \{v\}$, $C = C \cap N_G\{v\}$, $N = V(G) \setminus (P \cup C)$, go to step 2.

step 4: Output P, stop.

Following this procedure, we can find (one of) the maximal cliques in G. Since each of the cliques represents a possible conformation of the sequence, the maximal one with the best weight would be considered as the most similar one to the native protein structure.

The *score* of each clique used to find maximal one with best weight is defined as

$$S(d_{ab}) = -\ln \frac{P(d_{ab} \mid C)}{P(d_{ab})} \quad - \ln P(d_{ab} \mid C) \qquad (1)$$

where $S(d_{ab})$ represents the score of atoms type a and b with distance d, $P(d_{ab}|C)$ represents the probability of observing a distance d between atom type a and b in a correct structure, and $P(d_{ab})$ represents the probability of observing such a distance in all conditions without considering it is correct or not. The value of $P(d_{ab}|C)/P(d_{ab})$ is calculated by

$$\frac{P(d_{ab} \mid C)}{P(d_{ab})} = \frac{N(d_{ab}) / \sum_{d} N(d_{ab})}{\sum_{ab} N(d_{ab}) / \sum_{d} \sum_{ab} N(d_{ab})} \qquad (2)$$

where $N(d_{ab})$ represents the number of observations of atom types a and b in a particular distance d, $\Sigma_d N(d_{ab})$ represents the number of $a - b$ contacts observed for all distances, $\Sigma_{ab} N(d_{ab})$ represents the total number of contacts between all pair of atom types in a particular distance d, and $\Sigma_d \Sigma_{ab} N(d_{ab})$ represents the total number of contacts between all pair of atom types observed for all distances.

Given a weighted clique with n vertices and m edges representing a possible conformation, its score that represents the correctness of the probability can be calculated by

$$S(clique) = \sum S(vertex) + \sum S(edge) \tag{3}$$

where $S(vertex)$ is the sum of the scores for distances between all atoms p of the side-chain and atoms q of the total main-chain. Therefore, we have

$$S(vertex) = \sum S(d_{ab}^{pq}) \tag{4}$$

and $S(edge)$ is the sum of the scores for the distance between an atom r of one residue and an atom s of the other, which can be calculated by

$$S(edgs) = \sum S(d_{ab}^{rs}). \tag{5}$$

If the distance between r and s is no more than four residues, only side-chain atoms are used to calculate scores. All $S(vertex)$ and $S(edge)$ are calculated only once. By this means, the calculating cost can be reduced a lot.

Discussion and further improvement

This section gives a typical graph-theoretic method which solves homology modeling problem. It has mainly three advantages. First, it transfers a protein structure identification problem to a graph theory one, uses the algorithm of graph theory (clique finding) to solve it and makes the original problem easier to handle. Second, in this model, each score can be calculated fast, which makes the computation easy to accomplish. At last, this method excludes impossible conformation before giving weight, which eliminates the number of edges and reduces the computation scale.

However, we can also see that there are some disadvantages in this method. One is that clique finding in a given graph is an NP-hard problem that the computation time of the worst case is $O(3^{n/3})$ [21], so it cannot be applied to large proteins. The other is that the function used to calculating weights of vertices and edges eliminates that the weight must be independent from other vertices and edges.

This method showed its effectiveness in the experiments done by Samudrala and Moult [18]. When the scoring function is appropriate and the CF algorithm is suitable, it can find out the native-like conformations and native structure. This method successfully calculates the fitness of a conformation, excluding a large number of unacceptable conformations, then finds the conformations represented by the cliques independently. However, if the scale of the graph is extremely large, the clique finding algorithm would be timing consuming. Further improvements of the proposed method can be focused on at least two aspects. One is improving the algorithm and the other is modifying the model. For the former one, we can try to find other advanced clique finding (CF) algorithms to reduce the computation time and broaden the range of protein size, or we may use some parallel approaches to fasten the speed. For the latter one, we can modify the original model in selection part, adding filters to exclude more unacceptable conformations to reduce the scale of the graph.

Identification of side-chain clusters in protein structures upon graph spectrum

problem description

Side-chain interactions are essential to protein stability, function and folding. In protein secondary structures, the role of non-covalent side-chain interactions in stabilizing the mutual orientation has been studied well [23-25]. It is well known that clusters of hydrophobic side-chains on the surface are important for protein-protein recognition [26-30], protein oligomerization [31-33] and protein DNA interactions [34]. However, identifying side-chain interactions by experimental ways is very difficult, thus researchers prefer mathematical methods. In 1999, Kannan and Vishveswara explored a method to detect side-chain clusters in protein three-dimensional structures using a graph spectral approach [7].

Graph-theoretic modeling

The protein structure can be represented by a weighted graph being made up of residues. The vertices and edges are defined as follows.

Vertex: The C^{β} atoms of the interacting residues are represented by vertices in a graph. Since atoms are labeled by Greek alphabetic order, C^{α} is the carbon closest to the hydroxyl group($-OH$), and C^{β} is the second closest one.

Edge: If the distance between two C^{β} atoms satisfies specific interaction, we draw an edge between them.

In protein structure, side-chain interactions are represented by a weighted graph and the constructed graph is represented by its Laplacian matrix. Clusters are obtained directly from the eigenvector associated with the second lowest eigenvalue of the Laplacian matrix, and the side-chains which make the largest number of interactions in a cluster (cluster centers) are obtained from the eigenvectors associated with the top eigenvalues [7]. Particularly, clustering information is sorted in the vector components of the second lowest eigenvalue, for example, all vector components in the same cluster have the same value [35], and the vector components of the top eigenvalues carry the information regarding the branching of the points forming the cluster [36] and cluster centers [37,38]. This methodology, also been used in other disciplines like electrical engineering for obtaining clusters in circuit net-lists [39], has been used here for the identification of clusters in protein structures.

An easy way to construct an adjacency matrix is to assign 1 or 0 to a_{ij} according vertex i and j are adjacent or not in the graph. Here, we use the following weight to construct adjacency matrix.

$$a_{ij} = \begin{cases} 1/d_{ij}, & \begin{array}{l}\text{side-chains residues } i \text{ and } j\\ \text{above interaction criteria}\end{array} \\ 1/100 & \text{else} \end{cases}$$

where d_{ij} is the distance between C^{β} atoms of the residues i and j.

A distance of 100 is assigned to the two side-chains not satisfying the interaction criteria, hence their corresponding weight (1/100) are close to zero. The degree matrix $D := (d_{ij})$ is constructed as:

$$d_{ij} = \begin{cases} \sum_{j=1}^{n} a_{ij}, & i = j \\ 0 & \text{else} \end{cases}$$

thus, the Laplacian matrix B can be calculated as:

$$B = D - A. \tag{6}$$

Here, we also need to define a function that evaluates side-chain interactions since the definition of A uses it. The interaction can be calculated as

$$Int(R_i, R_j) = \frac{N(R_i, R_j)}{Normal(type(R_i))} \times 100 \tag{7}$$

where R_i, R_j are two different residues, $Int(R_i, R_j)$ is the side-chain interaction of residues R_i and R_j, and $N(R_i, R_j)$ is the number of all pairs of interacting side-chain atoms. Here only those atoms of residues have distance within 4.5 *Å* are calculated. $Normal(type(R_i))$ is the normalization value of residue R_i that can be calculated in advance. Here we do not concern the way of calculating this value, but only show the $Normal(type(R_i))$ for all 20 residues (see Table 1). Detailed calculation process can be found in [7].

After that, we can define the side-chain interaction criteria in different values. Noticing that when R_i and R_j are fixed, $Int(R_i, R_j)$ is fixed, too. When the side-chain interaction threshold becomes higher, fewer residues will be considered, which leads to fewer clusters being found. However, if the threshold is too low, it will result in large expanded clusters. Therefore, there is a tradeoff of setting the proper threshold in this method.

Table 1 The normal(type(*R_i*)) for 20 residues

Residue type	Normal value
Ala	55.7551
Arg	93.7891
Asn	73.4097
Asp	75.1507
Cys	54.9528
Gln	78.1301
Glu	78.8288
Gly	47.3129
His	83.7357
Ile	67.9452
Leu	72.3517
Lys	69.6096
Met	69.2569
Phe	93.3082
Pro	51.331
Ser	61.3946
Thr	63.7075
Trp	106.703
Tyr	100.719
Val	62.3673

The following table shows the *Normal*(*type*(R_i)) for 20 residues.

Since side-chain information can be calculated through the clique and clique center, our goal here is to find them. Specifically, Clusters are acquired from the eigenvectors associated with the second lowest eigenvalue of the Laplacian matrix, and side-chains that have the most interaction in cluster (cluster center) are acquired from the eigenvectors associated with the top eigenvalues. Therefore, the Laplacian matrix B contains the information of cliques and clique centers, and useful side-chains in the protein structure can be found by the above method. The detailed approach of calculating clique center upon graph spectrum and an example can be found in the Appendix of [7].

Discussion and further improvement

This section discusses the aspects of graph spectral approach that used for identification of side-chain clusters. Clusters are obtained directly from the eigenvectors associated with the second lowest eigenvalue of the Laplacian matrix and the side-chains which make the largest number of interactions in a cluster (cluster centers) are obtained from the eigenvectors associated with the top eigenvalues. This approach detects clusters by using different side-chain interaction criteria which can be changed by users easily. Higher side-chain interaction threshold results in less clusters while lower threshold leads to expanded clusters. Users may change the threshold to fit the specific problem they are concerning. Also, this approach can be implemented by numerical methods and the output is a simple two-dimensional cluster plot which contains the cluster and cluster center information.

However, this approach also has some disadvantages. One is that the side-chain interaction criteria is defined by researchers without any deep analysis on why this criteria is suitable, the other is that the way of constructing adjacency matrix A may be still simple and does not reflect interaction properly. Therefore, main issues in future can be the improvement of side-chain criteria and ways of constructing A.

De novo peptide sequencing via tandem mass spectrometry

Tandem mass spectrometry

Nowadays, *tandem mass spectrometry* (*MS/MS*) plays an important role in protein identification problems [40,41]. It breaks a peptide into smaller fragments and measures the mass of each fragment. A typical procedure of MS/MS contains the following steps. Protein mixtures are first digested into suitable sized peptides for mass spectrometric analysis using site-specific proteases (usually trypsin). Then the peptides are ionized during a ionization process. After that, Some of the peptides are fragmented by collision-induced dissociation (CID) and their tandem mass spectra are collected then [42-45].

A tandem mass spectrometry works like a charged sieve, we can only get a series of charged fragments from it [46,47]. Large molecules are broken into small pieces, and the problem of peptide sequencing is to find out the whole sequence of the peptide from these fragments [48]. A schematic of MS/MS is shown in Figure 2. More introduction about mass spectrometry and tandem mass spectrometry can be found in [49-54].

Problem of peptide sequencing

In the following subsection, we will provide the method of modeling peptide sequencing based on [5]. Let A be the set of amino acids, since there are 20 different amino acids in nature, A can be defined as:

$$A = \{a_1, a_2, \ldots, a_{20}\}. \tag{8}$$

Then, the mass of each amino acid can be denoted as $m(a_i)$, where $i \in [1, 2,..., 20]$.

Let $P = p_1...p_n$be a sequence of amino acids. The mass of each amino acid and the mass of parent peptide P are denoted as $m(\ p_i)$ and $m(P) = \sum_{i=1}^{n} m(p_i)$, respectively. A protein can be viewed as a chain of amino acids, which connected by a peptide bound. A peptide bound starts at a nitrogen(N) and ends at a carbon(C). We use P_i to represent N-terminal peptide $p_1...p_i$, and its mass can be calculated by $m_i = \sum_{j=1}^{i} m(p_j)$. Similarly, We use P_i^- to represent C-terminal peptide p_{i+1} ... p_n with mass $m(P) - m_i$.

When the peptide breaks down during MS/MS, it loses small pieces of molecules like water (H_2O), CO–group and NH–$group$[55-57]. Assuming that there are k

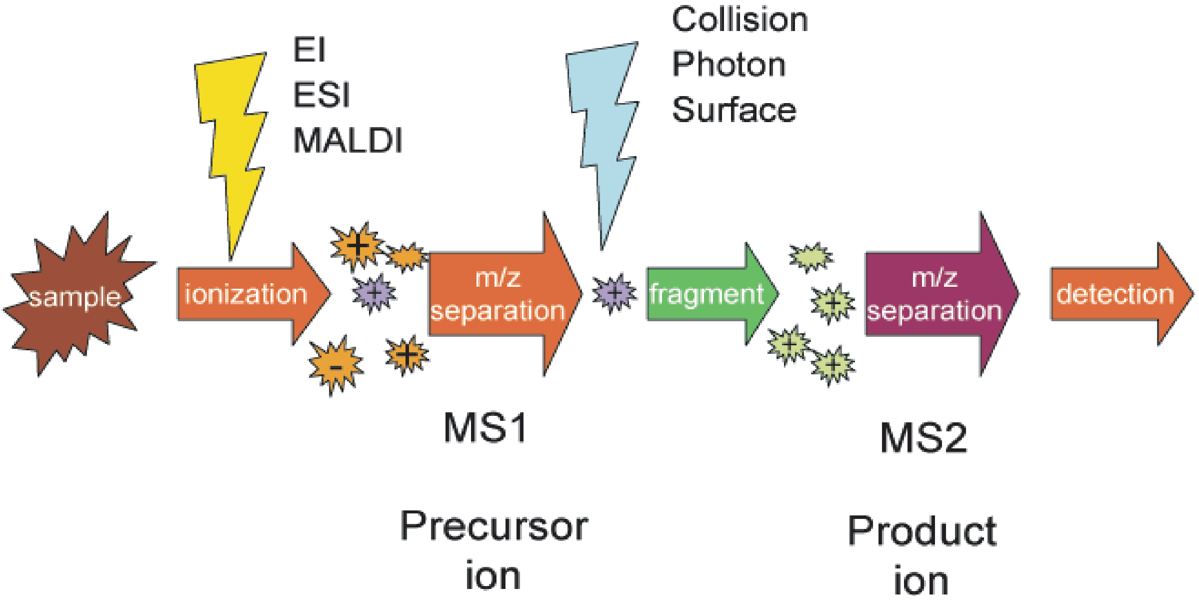

Figure 2 schematic of tandem mass spectrometry (from wikipedia)

different types of ions that correspond to the removal of k chemical groups, the set of ions can be defined as

$$\Delta = \{ \ _1, \ _2, \ldots, \ _k\}. \tag{9}$$

We also use δ_j to represent its mass, where j = 1, 2,..., k. A $\delta - ion$ of an N–terminal partial peptide P_i is a modification of P_i losing a small molecule of mass δ, and its mass is $m_i - \delta$. Similarly, we can define $\delta - ion$ of the C–terminal partial peptides [58,59].

We denote the theoretical spectrum of peptide P as T (P), it can be calculated by subtracting all possible ion types $\delta_1,\delta_2,\ldots,\delta_k$ from the masses of all partial peptide of P, such that every partial peptide generates k masses in the theoretical spectrum.

An experimental spectrum, denoted by S, is what we get from MS/MS, which can be defined as

$$S = \{s_1, s_2, \ldots, s_q\} \tag{10}$$

where s_t is a fragment ion (peak) in S, t = 1, 2,..., q. In the following, we also use s_t to represent its mass. The experimental spectrum usually includes loss of some small fragments and chemical noises. Actually, MS/MS measures m/z ratio, where m stands for mass and z stands for charge value (typically, it is 1, 2, or 3). Here, we assume that $z = 1$ for simplicity. The distinction of the theoretical spectrum $T(P)$ and the experimental spectrum S is the mathematical results ($T(P)$) given the peptide sequence P, and the experimental spectrum (S) without knowing what the peptide sequence is behind this spectrum (S). A match of $T(P)$ and S can be used to measure the relationship between the two as well as to predict peptide sequence of S. Therefore, the problem of peptide sequencing can be described as below.

Problem of Peptide Sequencing

Finding a peptide whose theoretical spectrum has a maximum match to a measured experimental spectrum.

Input: Experimental spectrum S, the set of possible ion types Δ, and the parent mass m.

Output: A peptide P of mass m whose theoretical spectrum matches S better than any other peptide of mass m

De novo peptide sequencing method

There are mainly two ways to solve peptide sequencing problems, one is database search, and the other is *de novo* method [57,60]. The former one involves generating all 20l amino acid sequences of a certain length l and the theoretical spectrum related to each sequence, finding the maximal match among all the spectra [61-63]. Considering the number of possible sequences grows exponentially with the length of peptide sequences, the computing time would also increase exponentially. *De novo* sequencing which usually uses a spectrum graph model, on the other hand, dose not need to generate all the amino acid sequences, thus developing fast and drawing increasing attention in recent years [64-66]. Here, we introduce basic models and principles of this kind of method [5,65]. Some recent improvements and advanced approaches can be found in [67-70].

In this method, a spectrum graph representing the experimental spectrum is constructed. Assuming that experimental spectrum $S = s_1,\ldots,s_q$ consists of N–terminal ions. Here, we ignore C–terminal ions because we can build a similar model of C–terminal ions by changing N–terminal ions into C–terminal ions. Every mass of $s_t \in S$ (t = 1, 2,..., q) may have been created from a partial peptide by one of the k different ion types. In other words, each s_t (t = 1, 2,..., q) corresponds to a spectrum of an ion, which is derived from some peptide P_i (i = 1, 2,..., n) losing some small group δ_j (j = 1, 2,..., k). However, we do not know what ion type of $\Delta = \{\delta_1, \delta_2,\ldots, \delta_k\}$ brings the mass of s_t, so we need to generate k different *guesses* for each mass in the experimental spectrum. Every guess corresponds to a hypothesis that, let x be the mass of some partial peptide, then $s_t = x - \delta_j$, where t = 1, 2,..., q and j = 1, 2,..., k. Therefore, there are k different guesses of a partial peptide with mass x that $s_t + \delta_1 s_t + \delta_2,\ldots, s_t + \delta_k$ corresponding to the mass s_t in experimental spectrum. That is to say, a partial peptide with mass x has k different possible conformations in this model.

After that, each mass in the experimental spectrum is transferred into a set consisting of k vertices in spectrum graph, corresponding to each possible ion type. The problem now can be solved by using graph theory. In particular, we use a directed acyclic graph (DAG) to represent the experimental spectrum. The vertices and edges of the graph are defined as follows.

Vertex: Each possible conformation of a partial peptide is represented by a vertex. The vertex for δ_j of the mass s_t is labeled with mass $s_t + \delta_j$.

Edge: An directed edge is drawn from vertex u to v if the mass of v is larger than that of u by the mass of a single amino acid.

Now, if we add a vertex at 0 representing the starting vertex (with mass 0) and a vertex at m representing the parent peptide (with mass M), the peptide sequencing problem can be translated into a path (from 0 to m) finding problem in the resulting DAG. Specifically, if there exists an edge from u to v, the chain of amino acids will be extended by adding a chemical group whose mass is the mass difference between vertex u and v. Therefore, by finding a path from 0 to m in the DAG, amino acid chain increases gradually and the peptide sequence can be found eventually.

In addition, vertices of the resulting spectrum graph is a set of numbers $s_t + \delta_j$ representing potential masses of N–terminal peptides adjusted by the ion type δ_j . Every mass s_t generates k different vertices, denoted by $V_t(s)$, then

$$V_t(s) = \{s_t + {}_1, s_t + {}_2, \ldots, s_t + {}_k\}. \tag{11}$$

There is the possibility that $V_t(s)$ and $V_\tau(s)$ may overlap when s_t and s_τ are close, where s_t, $s_\tau \in S$. The set of vertices in a spectrum graph is therefore $\{s_{initial}\} \cup V_1 \cup \ldots \cup V_q \cup \{s_{final}\}$, where $s_{initial} = 0$ and $s_{final} = m$.

The spectrum graph has at most $qk + 2$ vertices. We label the edge of the spectrum graph by amino acid whose mass is equal to the mass difference between two possible conformations (vertices). If we view vertices as putative N–terminal peptides, the edge from u to v implies that the N–terminal sequence corresponding to v can be obtained by extending the sequence at u by the amino acid that labels on the edge from u to v, where u, $v \in V(G)$.

For any $i \in [1, n]$, if S contains at least one ion type corresponding to every N–terminal partial peptide P_i , we say that the spectrum S of a peptide sequence $P = p_l \ldots p_n$ is complete. The use of a spectrum graph is based on the fact that, for a complete spectrum, there exists a path of length $n + 1$ from $s_{initial}$ to s_{final} in the spectrum graph that is labeled by P. This observation casts the peptide sequencing problem as one of finding the correct path in the set of all paths between two given vertices in a DAG. In addition, if the spectrum is complete, the correct path that we are finding will be the longest path in the graph usually [5].

Discussion and further improvement

In this section, we describe the *de novo* peptide sequencing problem and give an effective solution by a graph-theoretic method. The *de novo* method aims at inferring peptide sequences without using database, and the spectrum graph model solves this problem in a mathematical way. The solution successfully solves the problem by finding a longest path in a given spectrum graph. This kind of approach involves automatically interpreting the spectrum using the table of amino acids masses, and not relies on the completeness of database and effectiveness of searching algorithm, which the database method just relies on. Therefore, it usually costs less computation time, especially when the spectrum is with good quality.

However, this approach still has limitations. First, the success of finding the longest path in the graph relies on the completeness of mass spectrum, but in experiments, spectrum is always incomplete and combines with different kinds of noises, which makes the proposed approach hard to achieve. Second, finding the longest path in a given graph is an NP-complete problem which is difficult to find optimal solution. Third, when peptide breaks into MS/MS, it loses different kinds of small molecules, and considering all these losses needs a lot of vertices been created in the spectrum graph. When the number of vertices of the graph increases, computation time of solving this problem increases too, and even faster. At last, this kind of approach does not pay much attention to the peak intensity but using the *m/z* value only.

The performance of *de novo* peptide sequencing depends on the quality of the MS/MS spectra and the algorithms. When the spectra is complete or with high quality, *de novo* algorithm can find the correct sequences faster than the database search method, and also has the ability of finding new peptide which is not in the current database. Also, with advanced algorithm, *de novo* method could handle with spectra containing much noise, with missing peaks and so on. However, due to the limitation of tandem mass spectrometry, the database method is still the most popular and widely used one today. Some possible ways of improvements of de novo method are given below. First, when the spectrum is incomplete, we can add the missing ones by their complementary ions. Since any ion with a mass X in MS/MS, there should be an ion with mass Y such that $X + Y = M$, where M is the mass of the parent peptide. Thus we can add complementary ions back in an experimental spectral data set [71]. Second, we can consider effective algorithms on finding the longest path in a given graph such as dynamic programming and parallel approach. Third, this method can be partly solved by modifying the original model from finding global solution to possible local solutions. Some suboptimal algorithms can be considered, too [69]. Last but not least, a meaningful issue for the future research can be the combination of *de novo* method and other approaches, for example, database search [72].

Conclusions

This paper reviews several methods in solving protein structure identification problems using graph theory. We first introduce the development of protein structure identification and existing problems, then giving basic knowledge of graph theory, and focusing on three typical methods using graph theory to solve protein identification problems. These methods are effective but still have problems or some inadequacy, so we also give concluding remarks of them.

In homology modeling based on clique finding, a graph that represents all the possible conformations of residues in amino acids and their interactions is drawn. We use a clique finding algorithm to find out the cliques with the best weight that are viewed as the optimal combinations of various side-chain and main-chain conformations. In identification of side-chain clusters in protein structures, graph spectral method is used. Clusters are obtained directly from the eigenvectors associated with the second lowest eigenvalue of the Laplacian matrix and the side-chains which make the largest number of interactions in a cluster (cluster centers) are obtained from the eigenvectors associated with the top eigenvalues. In *de novo* peptide sequencing via tandem mass spectrometry, the spectrum graph represents all the possible conformation of the partial peptide and the mass difference between each pair of conformations is drawn first. Then by finding the longest path in the spectrum graph, we can obtain the peptide sequence.

The above three methods all change protein identification problems into graph-theoretical ones and find effective ways of solving them. They give novel methods for handling proteomics problems and can be improved in various aspects in future. There are mainly two directions of improvements. One is the algorithm, such as improving CF algorithm and the longest path algorithm; the other is the model, for example, modifying side-chain interaction criteria. These improvements will enhance the computation ability and make the graph scale an acceptable size. We have seen that in recent literature, researchers are focusing on some of the improvements and have already done partial work successfully. However, there are still a vast amount of work for us to do to improve the current modified methods and find better ways to solve different protein identification problems in graph theoretical methods.

Acknowledgements
This work was supported by Natural Sciences and Engineering Research council of Canada (NSERC) and National Natural Science Foundation of China (No. 10871158).
This article has been published as part of *Proteome Science* Volume 9 Supplement 1, 2011: Proceedings of the International Workshop on Computational Proteomics. The full contents of the supplement are available online at http://www.proteomesci.com/supplements/9/S1.

Author details
[1]Department of Applied Mathematics, Northwestern Polytechnical University, Xi'an, Shaanxi 710072, P.R. China. [2]Division of Biomedical Engineering, University of Saskatchewan, Saskatoon, SK S7N 5A9, Canada.

Authors' contributions
YY wrote the first draft of the review. SGZ intensively revised the manuscript. FXW supervised and gave suggestions of modifications of the manuscript. All authors read and approved the manuscript.

Competing interests
The authors declare that they have no competing interests.

Published: 14 October 2011

References

1. Williams KL, Gooley AA, Packer NH: **Proteome: not just a make-up name.** *Today's Life Science* 1996, 16-21.
2. Searls DB: **The roots of bioinformatics.** *PLoS Computational Biology* 2010, **6**:1-7.
3. González-Díaz H, González-Díaz Y, Santana L, Ubeira FM, Uriarte E: **Proteomics, networks and connectivity indices.** *Proteomics* 2008, **8**:750-778.
4. Pevzner PA: **Computational Molecular Biology: An Alogorithmic Approach.** Cambridge, Massachusetts: The MIT Press; 2000.
5. Jones NC, Pevzner PA: **An Introduction to Bioinformatics Algorithms.** Cambridge, Massachusetts: MIT press; 2004.
6. Bondy JA, Murty USR: **Graph Theory.** New York: Springer; 2008.
7. Kannan N, Vishveshwara S: **Identification of side-chain clusters in protein structures by a graph spectral method.** *J. Mol. Biol* 1999, **292**:441-464.
8. Pertsemlidis A, Fondon I, John W: **Having a BLAST with bioinformatics (and avoiding BLASTphemy).** *Genome Biology* 2001, **2(10)**:1-10.
9. Chothia C, Lesk A: **The relation between the divergence of sequence and structure in proteins.** *EMBO Journal* 1986, **5**:823-826.
10. Greer J: **Comparative modeling methods: application to the family of the mammalian serine proteases.** *Proteins: Struct. Funct. Genet* 1990, **7**:317-334.
11. Chen R: **Monte Carlo simulations for the study of hemoglobin fragment conformations.** *J. Comput. Chem* 1989, **10**:448-494.
12. Skolnick J, Kolinski A: **Simulations of the folding of a globular protein.** *Science* 1990, **250**:1121-1125.
13. Wilson S, Cui W: **Applications of simulated annealing to peptides.** *Biopolymers* 1990, **29**:225-235.
14. Venclovas C, Zemla A, Fidelis K, Moult J: **Numerical criteria for evaluating protein structures derived from comparative modeling.** *Proteins: Struct. Funct. Genet* 1997, , **Suppl 1**: 7-13.
15. Abagyan R, Totrov M: **Biased probability Monte Carlo conformational searches and electrostatic calculations for peptides and proteins.** *J. Mol. Biol* 1994, **235**:983-1002.
16. Avbelj F, Moult J: **Determination of the conformation of folding initiation sites in proteins by computer simulation.** *Proteins: Struct. Funct. Genet* 1995, **23**:129-141.
17. Harel D: **Algorithmics: The Spirit of Computing.** New York: Pearson Education; 1992.
18. Samudrala R, Moult J: **A graph-theoretic algorithm for comparative modeling of protein structure.** *J. Mol. Biol* 1998, **279**:287-302.
19. Moon J, Moser L: **On cliques in graphs.** *Israel J. Math* 1965, **3**:23-28.
20. Augustson JG, Minker J: **An analysis of some graph theoretical cluster techniques.** *Journal of the ACM* 1970, **17**:571-588.
21. Bron C, Kerbosch J: **Algorithm 457: finding all cliques of an undirected graph.** *Communications of the ACM* 1973, **16**:575-577.
22. Little , John D, *et al*: **An algorithm for the traveling salesman problem.** *Oper. Res* 1963, **11**:972-989.
23. Chou KC, Nemeth G, Scheraga HA: **Energetics of interactions of regular structural elements in proteins.** *Accts. Chem. Res* 1990, **23**:134-141.
24. Nemethy G, Scheraga HA: **A possible folding pathway of bovine pancreatic RNase.** *Proc. Natl. Acad. Sci. USA* 1979, **76**:6050-6054.
25. Creighton TE, Chothia C: **Electing buried residues.** *Nature* 1989, **339**:14-15.
26. Young L, Jernigan BL, Covell DG: **A role for surface hydrophobicity in protein-protein recognition.** *Protein Sci* 1994, **3**:717-729.
27. Guss JM, Freeman HC: **Structure of oxidized polar plastocyanin at 1.6 Å resolution [abstract].** *J. Mol. Biol* 1983, **169**:521-563.
28. Vam de Kamp M, Silvestrini MC, Brunoir M, Van Beumen J, Hali FC, Canters GW: **Involvement of the hydrophobic patch of azurin in the electron transfer reactions with cytochrome c551 and nitrite reductase.** *Eur. J. Biochem* 1990, **194**:109-118.
29. Pelletier H, Kraut J: **Crystal structure of a complex between electron transfer partners, cytochrome c peroxidase and cytochrome c.** *Science* 1992, **258**:1744-1755.
30. Chen L, Durley RCE, Mathews FS, Davidson VL: **Structure of an electron transfer complex: methylamine dehydrogenase, amicyanin and cytochrome c551i.** *Science* 1994, **264**:86-89.
31. Jones DH, McMillan AJ, Fersht AR: **Reversible dissociation of dimeric tyrosil-tRNA synthetase by mutagenesis at the subunit interface.** *Biochemistry* 1985, **245**:852-857.
32. Ponder JW, Richards FM: **Tertiary templates for proteins. Use of packing criteria in the enumeration of allowed sequences for different structural classes.** *J. Mol. Biol* 1987, **193**:775-791.
33. Mossing MC, Sauer RT: **Stable, monomeric variants of lambda-Cro obtained by insertion of a designed beta-hairpin sequence.** *Science* 1990, **250**:1712-1715.

34. Anderson JE, Ptashne M, Harrison SC: **Structure of the repressor-operator complex of bacteriophage 434.** *Nature* 1987, **326**:846-852.
35. Hall KM: **An r-dimensional quadratic placement algorithm.** *Manag. Sci* 1970, **17**:219-229.
36. Randic M: **Unique numbering of atoms and unique codes for molecular graphs.** *J. Chem. Inf. Comp. Sci* 1975, **15**:105-108.
37. Cvetkovic DM, Gutman I: **Note on branching.** *Croat. Chem. Acta* 1977, **49**:105-121.
38. Patra SM, Vishveshwara S: **Classification of polymer structures by a graph theory.** *Int. J. Quantum Chem* 1998, **71**:349-356.
39. Hagen L, Kahng AB: **New spectral methods for ratio cut partitioning and clustering.** *IEEE Trans. Comp.Design* 1992, **11**:1074-1084.
40. Johoson GJ, Biemann K: **Computer program (DEQPEP) to aid in the interpretation of high-energy collision tandem mass spectra of peptides.** *Biomed. Environ. Mass Spectrom* 1989, **18**:945-957.
41. McHugh L, Arthur JW: **Computational methods for protein identification from mass spectrometry data.** *PLoS Computational Biology* 2008, **4(2)**:1-12.
42. Wysockia VH, Resingb KA, Zhang QF, Cheng GL: **Mass spectrometry of peptides and proteins.** *Methods* 2005, **35**:211-222.
43. McLafferty FW, Turecek F: **Interpretation of Mass Spectra(Fourth Edition).** California: United Science Books; 1993.
44. Pitt JJ: **Principles and aplications of liquid chromatography mass spectrometry in clinical biochemistry.** *Clin. Biochem. Rev* 2009, **30**:19-34.
45. Marshall AG, Hendrickson CL, Jackson GS: **Fourier transform ion cyclotron resonance mass spectrometry: a primer.** *Mass Spectrom. Rev* 1998, **17**:1-35.
46. March RE: **Quadrupole ion trap mass spectrometry: theory, simulation, recent developments and applications.** *Rapid Commun. Mass Spectrom* 1998, **12**:1543-1554.
47. Na S, Paek E, Cheolju L: **CIFTER: automated charge-state determination for peptide tandem mass spectra.** *Anal. Chem* 2008, **80**:1520-1528.
48. Wang P, Polce MJ, Bleiholder C, Paizs B, Wesdemiotis C: **Structural characterization of peptides via tandem mass spectrometry of their dilithiated monocations.** *Int. J. Mass Spectrom* 2006, **249-250**:45-59.
49. Thomson JJ: **Rays of positive electricity and their application to chemical analysis.** *Proc. Roy. Soc* 1913, **89**:1-20.
50. Beynon J: **The use of the mass spectrometer for the identification of organic compounds.** *Microchimica Acta* 1956, **44**:437-453.
51. Biemann K, Cone C, Webster BR, Arsenault GP: **Determination of the amino acid sequence in oligopeptides by computer interpretation of their high-resolution mass spectra.** *J. Am. Chem. Soc* 1966, **88**:5598-5606.
52. Chamrad DC, Korting G, Stuhler K, Meyer HE, Klose J, *et al*: **Evaluation of algorithms for protein identification from sequence databases using mass spectrometry data.** *Proteomics* 2004, **4**:619-628.
53. Wong J, Sullivan M, Cartwright H, Cagney G: **msmsEval: tandem mass spectral quality assignment for high-throughput proteomics.** *BMC Bioinformatics* 2007, **8**:51.
54. Futrell JH: **Development of tandem mass spectrometry: one perspective.** *Int. J. Mass Spectrom* 2000, **200**:495-508.
55. Gray AL, Williams JG, Ince AT, Liezers M: **Noise sources in inductively coupled plasma mass spectrometry: an investigation of their importance to the precision of isotope ratio measurements.** *J. Anal. At. Spectrom* 1994, **9**:1179-1181.
56. Zhang JF, He SM, Ling CX, Cao XJ, Zeng R, Gao W: **PeakSelect: preprocessing tandem mass spectra for better peptide identification.** *Rapid Commun. Mass Spectrom* 2008, **22**:1203-1212.
57. Resing KA, Ahn NG: **Proteomics strategies for protein identification.** *FEBS Letters* 2005, **579**:885-889.
58. Wysocki VH, Tsaprailis G, Simth LL, Mobile B, Protons L: **A framework for understanding peptide dissociation.** *J. Mass Spectrom* 2000, **35**:1399-1406.
59. Aebersold R, Goodlett DR: **Mass spectrometry in proteomics.** *Chem. Rev* 2001, **101**:269-295.
60. **Protein ID: comparing de novo based and database search methods** [http://www.bioinformaticssolutions.com/functions db download.php?id=3558]. .
61. Eng J, McCormack A, Yates J: **An approach to correlate tandem mass spectral data of peptides with amoni acid sequences in a protein database.** *J. Am. Soc. Mass Spectrom* 1994, **5**:976-989.
62. Mann M, Wilm M: **Error-tolerant identification of peptides in sequence tags.** *Anal. chem* 1994, **66**:4390-4399.
63. Sadygov RG, Cociorva D, Yates III JR: **Large-scale database searching using tandem mass spectra: looking up the answer in the back of the book.** *Nature methods* 2004, **1(3)**:195-202.
64. Bassil I, Dahiyat , Mayo SL: ***De novo* protein design: fully automated sequence selection.** *Science* 1997, **278**:82-87.
65. Dancik V, Addona TA, Clauser KR, *et al*: ***De novo* peptide sequencing via tandem mass spectrometry.** *J. Comput. Biol* 1999, **6**:327-342.
66. Lu BW, Chen T: **Algorithms for de novo peptide sequencing using tandem mass spectrometry.** *BIOSILICO* 2004, **2**:85-90.
67. Chen T, Kao MY, Tepel M, *et al*: **A dynamic programming approach to *de novo* peptide sequencing via tandem mass spectrometry.** *J. Comput. Biol* 2001, **8(3)**:325-337.
68. Ma B, Zhang K, Hendrie C, *et al*: **PEAKS: powerful software for peptide *de novo* sequencing by tandem mass spectrometry.** *Rapid Commun. Mass Spectrom* 2003, **17**:2337-1342.
69. Lu BW, Chen T: **A suboptimal algorithm for *de novo* peptide sequencing via tandem mass spectrometry.** *J. Comput. Biol* 2003, **10**:1-12.
70. Frank A, Pevzner PA: **PepNovo: *de novo* peptide sequencing via probabilistic network modeling.** *Anal. Chem* 2005, **77**:964-973.
71. Yan B, Pan CL, Olman VN, Hettich RL, Xu Y: **A graph-theoretic approach for the separation of b and y ions in tandem mass spectra.** *Bioinformatics* 2005, **21**:563-574.
72. Taylor JA, Johnson RS: **Sequence database searches via *de novo* peptide sequencing by tandem mass spectrometry.** *Rapid Commun. Mass Spectrom* 1997, **1(9)**:1067-1075.

doi:10.1186/1477-5956-9-S1-S17
Cite this article as: Yan *et al.*: **Applications of graph theory in protein structure identification.** *Proteome Science* 2011 **9**(Suppl 1):S17.