Published in final edited form as:
Proceedings (IEEE Int Conf Bioinformatics Biomed). 2017 November ; 2017: 222–229. doi:10.1109/BIBM.2017.8217653.

A Spectrum Graph-Based Protein Sequence Filtering Algorithm for Proteoform Identification by Top-Down Mass Spectrometry

Runmin Yang[1,2], **Daming Zhu**[1,*], **Qiang Kou**[2], **Poomima Bhat-Nakshatri**[3], **Harikrishna Nakshatri**[3], **Si Wu**[4], and **Xiaowen Liu**[2,5,*]

[1]School of Computer Science and Technology, Shandong University

[2]Department of BioHealth Informatics, Indiana University-Purdue University Indianapolis

[3]Department of Surgery, Indiana University School of Medicine

[4]Department of Chemistry and Biochemistry, University of Oklahoma

[5]Center for Computational Biology and Bioinformatics, Indiana University School of Medicine

Abstract

Database search is the main approach for identifying proteoforms using top-down tandem mass spectra. However, it is extremely slow to align a query spectrum against all protein sequences in a large database when the target proteoform that produced the spectrum contains post-translational modifications and/or mutations. As a result, efficient and sensitive protein sequence filtering algorithms are essential for speeding up database search. In this paper, we propose a novel filtering algorithm, which generates spectrum graphs from subspectra of the query spectrum and searches them against the protein database to find good candidates. Compared with the sequence tag and gaped tag approaches, the proposed method circumvents the step of tag extraction, thus simplifying data processing. Experimental results on real data showed that the proposed method achieved both high speed and high sensitivity in protein sequence filtration.

Keywords

Mass spectrometry; filtering algorithm; spectrum graph

I. Introduction

Because top-down mass spectrometry (MS) provides “a bird’s eye” view of whole proteoforms, it is an important technology for identifying proteoforms with primary sequence alterations, such as post-translational modifications (PTMs) and mutations [1]. Reliable identification of protein alterations, which are related to changes in cellular or tissue dynamics, is essential to understanding cellular regulatory processes and diseases [2].

Database search [3], [4], represented by tools such as ProSightPC [5] and TopPIC [6], is the dominant approach for top-down MS-based proteoform identification, in which top-down

[*]Corresponding authors, dmzhu@sdu.edu.cn xwliu@iupui.edu.

tandem mass (MS/MS) spectra are searched against a protein sequence database to identify matches between spectra and protein sequences. It is a challenging computational problem because the target proteoform that produced the query spectrum often contains multiple alterations that are not included in its corresponding database protein sequence.

There are two main approaches for identifying proteoforms with alterations. The first approach is to build an extended proteoform database [5] using known protein alterations reported in the literature. For example, phosphorylation, acetylation, and ubiquitylation sites have been reported in the tumor protein p53 [7], and the knowledge can be used to build a proteoform database of the protein. When the alterations on the target proteoform are known, using an extended database containing the target proteoform simplifies database search. However, the extended database approach has its limitations. Many proteins have not been extensively studied, and most of their PTM sites are unknown. Even if the PTM sites of a protein are known, each PTM site may or may not be present in a proteoform and the number of combinations of these PTM sites is an exponential function of the number of sites. As a result, it is impractical to build a complete extended protein database.

The second approach is to employ spectral alignment algorithms [3], [4], [6], [8]–[10] to align a top-down MS/MS spectrum from a modified proteoform against unmodified database protein sequences. While spectral alignment is efficient to align a spectrum against a protein sequence, it is extremely time-consuming to align thousands of query spectra against all protein sequences in a large database. Therefore, efficient protein sequence filtering algorithms are essential to speeding up database search in proteome-level proteomics studies [3].

Given a query MS/MS spectrum and a database of unmodified protein sequences, the objective of a protein sequence filtering algorithm is to quickly filter out most protein sequences in the database while keeping the target one that produced the query spectrum. When the target proteoform contains alterations, the precursor mass of the query spectrum does not match the molecular mass of its corresponding database protein sequence. As a result, fragment masses of the query spectrum need to be used for protein sequence filtration.

Many filtering methods have been proposed in bottom-up and top-down MS [11], [12]. One main filtering approach is to use sequence tags, which are partial protein sequences extracted from mass spectra, to scan and filter protein sequences [11], [13], [14]. Correct long sequence tags, which are substrings of the target protein sequence, are extremely useful in filtering out protein sequences, but we may fail to extract such tags from mass spectra due to missing peaks.

To address the problem, Jeong et al. introduced gapped tags, which can be extracted from spectra with missing peaks [15]. Many MS/MS spectra contain correct long gapped tags, but not correct long sequence tags. Several algorithms have been proposed to efficiently search gapped tags against a protein database [16], [17]. Because of the complexity of MS/MS spectra, it is challenging to distinguish signal from noise peaks. The gapped tag approach solves the problem of missing peaks in tag extraction, but it is highly possible that many incorrect tags are reported because of noise peaks.

In this paper, we propose a novel filtering algorithm, which extracts subspectra of the query spectrum, constructs spectrum graphs from these subspectra, and searches them against the protein database to filter protein sequences. Compared with the gapped tag approach, the proposed approach simplifies protein sequence filtration by skipping the tag extraction step. It directly searches spectrum graphs generated from subspectra with noise peaks against the protein database. Experimental results on real data demonstrated that the proposed approach achieved both high filtration efficiency and high speed in protein sequence filtration.

II. Methods

A top-down MS/MS spectrum of a proteoform consists of a precursor mass corresponding to the molecular mass of the proteoform and a list of fragment ions peaks corresponding to fragments of the proteoform. In general, top-down MS/MS spectra are complex because of the existence of highly charged fragment ions and isotopic peaks. In data preprocessing, fragment ion peaks in top-down MS/MS spectra are converted into monoisotopic fragment masses using spectral deconvolution tools [18]. The intensity of a monoisotopic mass is assigned as the sum of the intensities of its corresponding fragment ions peaks. We assume that the query spectrum is a deconvoluted one in the following analysis.

A deconvoluted top-down MS/MS spectrum may contain some noise masses. Generally speaking, high-intensity masses tend to be signal ones and low-intensity masses noise ones. We use an intensity-based method to remove noise masses from the query spectrum. For each mass x in the query spectrum, we rank all the masses in the width 200 Dalton (Da) interval $[x - 100, x + 100]$ in the decreasing order of their intensities. If x is not one of the top λ masses, x is treated as a noise peak and removed, where λ is a user-specified parameter.

A. Spectrum graphs

In sequence tag-based protein sequence filtration, spectrum graphs are often used to extract sequence tags from mass spectra [19]. A spectrum graph is generated from a query spectrum with two steps: (a) a node is added to the graph for each fragment mass in the spectrum, and (b) a directed edge from node u to node v is added to the graph if mass(v) − mass(u) matches the mass of one amino acid residue[1] within an error tolerance, where mass(u) and mass(v) are the masses corresponding to u and v, respectively. The edge is labeled with the amino acid explaining the mass difference (Fig. 1(b)). Each path in the spectrum graph corresponds to a sequence tag, which is the readout of the labeled amino acids on its edges. For example, NVRS is the sequence tag extracted from the only path from node v_0 to v_4 in the spectrum graph in Fig. 1(b). There are various methods for extracting sequence tags from the spectrum graph [14], [20]. These tags are then searched against the protein database for protein sequence filtration.

[1] The mass difference is matched to the mass of one or two amino acids in some algorithms.

Long correct sequence tags that are matched to the target protein sequence are extremely useful because the chance that they are matched to random sequences is low. When the query spectrum misses many fragment masses, many node pairs whose mass differences are explained by 2 or more amino acids in its spectrum graph cannot be connected. In this case, the spectrum graph approach described in the previous paragraph may fail to extract long correct sequence tags. To solve this problem, gapped sequence tags were introduced to allow large mass gaps between two fragment masses in tag generation [15].

To extract gapped sequence tags, we change step (b) in the generation of spectrum graphs as follows: a directed edge is added into the graph from node u to v if the mass difference mass(v) − mass(u) is no large than a predefined bound α (α is chosen between 200 and 400 Da in practice) and is explained by a combination of several amino acids; the edge is labeled with the mass difference mass(v) − mass(u) (Fig. 1(c)). Each path in the spectrum graph corresponds to a mass sequence, called a *blocked pattern* or a gapped sequence tag [16]. In protein sequence filtration, multiple blocked patterns (gapped sequence tags) are extracted from the spectrum graph and searched against the protein database. The number of blocked patterns in a spectrum graph may be an exponential function of the length of blocked patterns, making it inefficient to explicitly extract all blocked patterns in the spectrum graph. As a result, it is common that only a limited number of paths and their corresponding blocked patterns are chosen. Because of the existence of noise peaks, it is a challenging problem to determine which paths in the spectrum graph correspond to blocked patterns that match the target protein sequence. We propose to circumvent the tag generation step and search the spectrum graph directly against the protein database for protein sequence filtration.

B. The spectrum graph matching problem

Most protein databases, such as UniProt [21], contain only one unmodified reference sequence for a gene or a transcript. We assume that the protein database used in spectral identification contains only unmodified reference sequences to simplify the analysis. In addition, discretized masses are used in the definition of the spectrum graph matching problem. All amino acid residue masses are discretized by first multiplying the masses by a scaling factor (100 was used in the experiments) and then rounding the results to integers.

Notations introduced by Deng et al. [17] are used to define the spectrum graph matching problem. An amino acid string is represented as a list of discretized residue masses: the mass representation of an amino acid string a_1, a_2,…, a_n is a residue mass string $S = s_1$, s_2, ..., s_n, in which s_i is the discretized residue mass of a_i for $1 \le i \le n$. Residue mass strings are called *text strings* in the following analysis. The sum of all the masses in a substring s_i, s_{i+1}, …, s_j of S, i.e., $\sum_{k=i}^{j} s_k$, is called the *mass of the substring.* Two substrings of S are called *consecutive substrings* if the first residue mass of the second string directly follows the last residue mass of the first string. For example, s_i, s_{i+1}, …, s_j and s_{j+1}, s_{j+2}, … ,s_k are consecutive substrings of S. A sequence of consecutive substrings A_1, A_2, …, A_k including all residue masses in S is called a partition of S. The masses of the consecutive substrings in a partition is called the mass string of the partition. For example, let S = 114, 156, 99, 87, A_1 = 114, A_2 = 156, 99, and A_3 = 87, then A_1, A_2, A_3 is a partition of S and the mass string of the partition is 114, 255, 87, where 255 is the mass of A_2. A blocked pattern obtained from a path in a spectrum graph is represented by a sequence of masses, which are labels of the edges in the path. For example, the blocked pattern corresponding to the bold path in Fig. 1(c) is 114.04, 225.17, 87.03. The mass sequence is further discretized to an integer sequence using the same method for residue mass discretization. In the following analysis, we assume that all blocked patterns are integer ones. Unlike text strings, a blocked pattern contains not only single amino acid residue masses, but also those of combinations of several amino acids. A blocked pattern P matches a text string S if P is the mass string of a partition of S. For example, the blocked pattern 114, 255, 87 matches the text string 114, 156, 99, 87 because the blocked pattern is the mass string of a partition of the text string.

In protein sequence filtration, we search all blocked patterns in a spectrum graph against the protein database to find matched amino acid sequences (sequence tags). All protein sequences in the database are concatenated and represented as a text string over an alphabet Σ containing the discretized residue masses of the 20 common amino acids. Because the masses of leucine and isoleucine are the same, the size of Σ is 19 instead of 20. Given a text string T over an alphabet Σ and a blocked pattern P, the blocked pattern matching problem is to find all substrings of T that match P. The spectrum graph matching (SGM) problem is more complex than the blocked pattern matching problem.

Definition 1—Given a text string T over an alphabet Σ and a spectrum graph G, the spectrum graph matching problem is to find all substrings of T that match a blocked pattern in G.

We first study a restricted version of the SGM problem, in which a start node and an end node in G are specified and only paths from the start node to the end node are used to find matched substrings of the text string. The SGM problem is solved by enumerating all node pairs in G as the start and end nodes in the restricted spectrum graph matching (RSGM) problem.

Definition 2—Given a text string T, a spectrum graph G over an alphabet Σ, a start node s, and an end node t, the restricted spectrum graph matching problem is to find all substrings of T that match a blocked pattern corresponding to a path from s to t in G.

C. A suffix tree based algorithm for the RSGM problem

We present a suffix tree based algorithm for the RSGM problem, which extends the algorithm proposed by Deng et al. for the blocked pattern matching problem [17]. The text string T is represented by a suffix tree. To simplify the algorithm description, we assume that each edge in the suffix tree is labeled with only one letter (integer residue mass).

Below we review the algorithm for the blocked pattern matching problem [17]. A blocked pattern is represented as a spectrum graph with only one path from the start node to the end node. Let $G = \{V, E\}$ be the graph representation of a blocked pattern P, where $V = \{v_0, v_1 \ldots, v_m\}$ and $v_0, v_1, \ldots, v_m$ is the only path from v_0 to v_m in G. Each prefix $p_1, p_2, \ldots, p_k$ of the blocked pattern P corresponds to a path $v_0, v_1, \ldots, v_k$. A text string over Σ that matches the prefix $p_1, p_2, \ldots, p_k$ is called a prefix text string of v_k. A prefix text string is *identifiable* if it is a substring of T. For example, when $P = 114, 255, 87$, both 114, 156, 99 and 114, 99, 156 are prefix text string of P. When $T = 114, 156, 99, 87$, the string 114, 156, 99 is an identifiable prefix text string of P, but 114, 99, 156 is not. If a prefix text string is not identifiable, then all text strings with the prefix are not identifiable, making it not necessary to search these text strings in T. Using identifiable prefix text strings significantly improve in the speed of searching a blocked pattern against the text string represented as a suffix tree.

Let B_i be the set of nodes in the suffix tree corresponding to all identifiable prefix text strings of v_i for $0 \leq i \leq m$, where m is the length of the blocked pattern. Specifically, B_0 contains only the root node of the suffix tree. The blocking pattern matching algorithm progressively finds the node sets $B_1, \ldots, B_m$ in the suffix tree. The node set B_m contains the solution to the blocked pattern matching problem.

Unlike the blocked pattern matching problem, the spectrum graph G in the RSGM problem contains many paths from the start node to the end node. A trivial method to solve the RSGM problem is to explicitly extract all paths in the spectrum graph and search each path against the suffix tree separately. However, the number of all paths in a spectrum graph may be an exponential function of the number of nodes, making this approach inefficient.

Next, we describe how to extend the blocked pattern matching algorithm to solve the RSGM problem. Let $V = \{v_0, v_1, \ldots, v_m\}$ be the set of nodes in the increasing order of their corresponding masses, in which the start node s is v_0 and the end node t is v_m. A text string is a prefix text string of node v_i if it matches a blocked pattern corresponding to a path from v_0 to v_i. Let B_i be the set of nodes in the suffix tree corresponding to all identifiable prefix text strings of v_i, and $C(e)$ be the set of all text strings whose masses match the labeled mass of e. In practice, a precomputed table is used to quickly obtain $C(e)$. Because the fragment masses in the query spectrum have measured errors, an error tolerance ε is allowed in generating the text strings in the table. Denote $E_i \subseteq E$ the set of all edges whose tail nodes are v_i. For each edge $e = (v_j, v_i)$ in E_i, we search from each suffix tree node in B_j to find all the paths corresponding to a text string in $C(e)$ and add the end nodes of these paths to B_i.

After the last set B_m is found, the RSGM problem is solved by reporting all the text strings corresponding to a suffix tree node in B_m and their positions in T, which are stored in the suffix tree. Because the text string T is represented by a suffix tree, the space complexity and time complexity of the algorithm are both $O(n)$, where n is the length of the text string T.

The time complexity of the preprocessing step, in which the text string T is represented as a suffix tree, is $O(n)$. Let d be the maximum out-degree of the nodes in G, and N be the maximum size of $C(e)$ for all edges in G, that is, $N = \max_{e \in E} |C(e)|$. We can prove that the time complexity of the graph query is $O(dNn)$.

The RSGM algorithm finds all the substrings in T that match a path from v_0 to v_m by reporting the suffix tree nodes in B_m and gives all the substrings in T that match a path from v_0 to v_i by reporting the suffix tree nodes in B_i. That is, the algorithm reports all the substrings in T that match a path starting from v_0. The SGM problem is solved by setting v_m

as the end node t and enumerating each node in G as the start node s, and solving the corresponding RSGM problems separately.

In practice, we are interested in finding only text strings that match a long path in G. Let V_s be the set of nodes v in G such that the sum of the labeled masses on a path from v_0 to v is no larger than β, where β is a user-specified parameter. Let V_t be the set of nodes v in G such that the sum of the labeled masses on a path from v to v_m is no larger than β. We only report text strings that match a path from a node in V_s to a node in V_t.

D. Protein sequence filtration

A top-down MS/MS spectrum often has low fragment coverage, and its spectrum graph consists of several connected components. Based on this observation, we propose to extract γ mass intervals (subspectra) that are abundant with fragment masses from the query spectrum and construct a spectrum graph from each extracted mass interval, where γ is a user-specified parameter. In practice, the value of γ is chosen between 1 and 20. The spectrum graphs are searched against the protein database separately and the search results are combined to find top sequence candidates. The filtering algorithm is referred to as the spectrum graph matching (SGM) filtering algorithm.

In mass interval selection, mass intervals with the same width δ are reported and the width δ is a user-specified parameter. In practice, the value of δ is chosen between 500 and 1400 Da so that the reported mass intervals correspond to 5 – 14 amino acids. (See Section "Parameter settings.") The score of a mass interval is defined as the number of masses in it. A greedy approach is employed to find multiple top-scoring mass intervals. We first find and report a top-scoring mass interval. Next, we find and report a topscoring mass interval whose overlapping region with each reported mass interval is less than ρ, which is a user-defined parameter. The second step is performed iteratively until a total of γ mass intervals are reported or no mass intervals that satisfy the condition can be found. In addition, only mass intervals with at least 6 fragment masses are reported.

We use reversed mass intervals to find text strings that match suffix fragment masses in the query spectrum. Each extracted mass interval is further reversed to obtain a reversed mass interval in which suffix fragment masses are converted into prefix fragment masses. We search the spectrum graphs generated from the extracted and reversed mass intervals against the protein database represented by a suffix tree.

A protein sequence matches a blocked pattern P if its text string representation contains a substring that matches P. The score of the pattern P is the number of the nodes in the corresponding path in the spectrum graph. The similarity score between a protein sequence and a spectrum graph is the score of the best scoring pattern extracted from the graph that matches the protein sequence. Let $G_1, G_2, \ldots, G_k$ be the set of spectrum graphs extracted from a query spectrum, the similarity score between a protein sequence and the query spectrum is the best similarity score between the protein sequence and $G_1, G_2, \ldots, G_k$.

After finding the best scoring pattern from $G_1, G_2, \ldots, G_k$ that matches a substring of the protein sequence, we compute the mass shift between the experimental fragment masses in the spectrum and the theoretical fragment masses of the protein sequence, and extend the substring at the both ends to find more matched experimental fragment masses. The sum of the scores of the path and the matched fragment masses found by the extension is called the *extended similarity score* between the protein sequence and the query spectrum.

III. Results

The SGM filtering algorithm was implemented in Java and tested on a Linux (64-bit) desktop PC with an Intel 3.4 GHz CPU and 16 GB memory.

A. Data sets

Two data sets were used to evaluate the SGM filtering algorithm. The first is a top-down MS data set with 2027 collision-induced dissociation (CID) and 2027 electron-transfer dissociation (ETD) MS/MS spectra from *Escherichia coli* (EC) K-12 MG1655. The EC sample was analyzed by a liquid chromatography system coupled with an LTQ Orbitrap Velos mass spectrometer. MS and MS/MS spectra were collected at a 60000 resolution and the top 4 ions in each MS spectrum were selected for MS/MS analysis.

The second top-down MS data set was generated from MCF-7 cells. MCF-7 cells were obtained from American Tissue Culture Collection (ATCC) and maintained in minimum essential media with 5% charcoal-dextran treated fetal calf serum. In the MS experiment, soluble MCF-7 intact proteins were separated using the bead-beating based cell lysis approach followed by a filter-based desalting step. A liquid chromatograph system with a home-made long column was directly coupled with an LTQ Orbitrap Velos Pro mass spectrometer with a customized ion source. A total of 5174 CID MS/MS spectra were collected at a 60000 resolution.

B. Protein spectrum-matches for evaluation

The EC raw data was centroided using msconvert in ProteoWizard [22] and deconvoluted using MS-Deconv [18]. The *Escherichia coli* K-12 MG1655 proteome database (September 12, 2016 version, 4306 entries) was downloaded from the UniProt database [21] and was concatenated with a shuffled decoy database of the same size. TopPIC [6] was employed to search the deconvoluted spectra against the target-decoy database. In TopPIC, unexpected modifications were not allowed in identified proteoforms, and the error tolerances for precursor and fragment masses were set as 15 part per million (ppm). With a 1% spectrum-level false discovery rate (FDR), a total of 1866 proteoform spectrum-matches were identified. Because of the stringent FDR cutoff, we assume that all the matches are correct in the following analysis. The 1866 proteoform spectrum-matches are referred to as the EC evaluation data set.

The SGM filtering algorithm was employed to search the spectrum in each of the 1866 matches against the EC proteome database and report the 20 top scoring proteins. If the top 20 proteins contain the protein in the proteoform spectrum-match reported by TopPIC, we say the filtering is efficient. The *filtration efficiency rate* of the filtering algorithm is defined as the ratio between the number of efficiently filtered spectra and the number of all spectra in the experiment.

C. Parameter settings

We used the EC evaluation data set to test the performance of the SGM filtering algorithm with various settings of the parameters In the experiments, only one spectrum graph was extracted from each query spectrum ($\gamma = 1$), and reversed mass intervals were not included.

We first evaluated the filtration efficiency with various settings of the three parameters α, β and δ. In the filtering algorithm, no masses are removed in the spectral preprocessing ($\lambda = \infty$), the error tolerance ε was set as 0.02 Da. To test the 3 parameters, we fixed the settings of 2 parameters and compared the performances with various settings of the third parameter. First, the parameters α and β were set as $\alpha = 300$ Da and $\beta = 200$ Da, and the parameter δ was set as various values between 500 to 1400 Da. The best filtration efficient rate (29.21%) of the algorithm is obtained when $\delta = 900$ Da (Fig. 2). Although the running time decreases as the value of δ increases, there is no significant difference in the running time when $\delta \geq$ 900 Da.

Similarly, we evaluated the performance of the SGM filtering algorithm with various settings of β between 0 and 400 Da and fixed settings $\delta = 900$ Da and $\alpha = 300$ Da and the performance with various settings of α between 200 and 500 Da and fixed settings $\delta = 900$ Da and $\beta = 250$ Da. The algorithm achieved the best filtration efficiency when $\beta = 250$ Da in the first experiment and when $\alpha = 350$ Da in the second experiment. The best filtration efficiency rate 33.07% was obtained when $\alpha = 350$ Da, $\beta = 250$ Da and $\delta = 900$ Da.

In practice, the error tolerance ε in the SGM filtering algorithm is determined by the precision of the mass spectrometer. By using the following parameter settings: $\alpha = 350$ Da, $\beta = 250$ Da, $\delta = 900$ Da, $\lambda = \infty$, we compared the performance of the filtering algorithm on the EC evaluation data set with various settings of the error tolerance ε in [0, 0.05] Da. The best filtration efficient rate was achieved when $\varepsilon = 0.02$ Da, and the filtration efficient rates were similar when $\varepsilon = 0.01$ and 0.02 Da.

We also tested how the parameter λ in spectral preprocessing affects the performance of the SGM filtering algorithm. We set $\alpha = 350$ Da, $\beta = 250$ Da, $\delta = 900$ Da, $\varepsilon = 0.02$ Da, and compared the performance of the filtering algorithm on the EC evaluation data set with various settings of the parameter λ between 2 and 12. By setting $\lambda = 8$, the SGM filtering algorithm achieved the best filtration efficiency. Based on the experimental results, the default settings of the parameters are set as $\alpha = 350$ Da, $\beta = 250$ Da, $\delta = 900$ Da, $\varepsilon = 0.02$ Da, and $\lambda = 8$.

D. Multiple spectrum graphs—We used the methods described in Section “Protein sequence filtration” to extract multiple spectrum graphs with two parameters: the number γ of extracted spectrum graphs and the maximum overlapping region ρ between two mass intervals. We compared the performance of the SGM filtering algorithm with various settings of the two parameters: $\gamma = 1, 3, 5, 10, 20$, and $\rho = 0\%, 20\%, 50\%, 80\%$. Other parameters were set as the default values. Using the parameter settings $\gamma = 20$ and $\rho = 50\%$ obtained the best filtration efficiency (Table I), but the running time with the setting was about 7 times longer than that with the setting $\gamma = 20$ $\rho = 20\%$. Because the numbers of fragment masses in many query spectra are limited, the total number of spectrum graphs reported from a query spectrum often depends on the maximum allowed overlapping region. When the maximum overlapping region is 0%, that is, overlapping between two mass intervals is not allowed, only a few spectrum graphs are generated in most cases. When the maximum overlapping region is 80%, more spectrum graphs are generated, significantly increasing the running time. This is the reason why the parameter setting $\gamma = 20$ and $\rho =$ 20% achieved a good balance between the filtration efficiency and speed.

We further evaluated the performance of the SGM algorithm with reverse mass intervals. Combining spectrum graphs generated from original mass intervals of the query spectrum and their reversed mass intervals improves the filtration efficiency. When $\gamma = 20$ and $\rho =$ 20%, experiments showed the filtration efficiency rate (71.01%) of the SGM filtering algorithm with reversed mass intervals was much higher than that (54.44%) without reversed mass intervals.

E. Comparison with other filtering algorithms

We compared the SGM filtering algorithm with two tag-based algorithms for protein sequence filtration on the MCF-7 data set. The first tag-based algorithm is implemented in MS-Align+Tag [23] and referred to as TAG-1; the second tag-based algorithm is implemented in MSPathFinder [10] and referred to as TAG-2. The MCF-7 raw data was centroided using msconvert in ProteoWizard [22] and deconvolued using MS-Deconv [18]. The human proteome database (July 9, 2016 version, 20191 entries) was downloaded from the UniProt database and concatenated with a shuffled decoy database with the same size. The parameters of the SGM filtering algorithm were set as the default values. In the tag-based methods, the error tolerance for matching the difference between two fragment masses to an amino acid residue mass was set as 0.02 Da. Each filtering algorithm reported 20 candidate protein sequences for each query spectrum. The average running time (554 ms) of the SGM algorithm for a query spectrum was about 2 times faster than the tag-based algorithms (TAG-1: 1340 ms; TAG-2: 1350 ms).

The three filtering algorithms were further coupled with the spectral alignment algorithm in TopPIC for spectral identification. The parameter settings of TopPIC were the same as those in the experiment in Section 3.2 except that at most 1 unexpected modification was allowed in an identified proteoform. With a 1% spectrum-level FDR, the SGM, TAG-1, TAG-2 filtering algorithms identified 894, 342, 601 proteoform spectrum-matches, respectively. A total of 251 spectra were identified by all the three methods. The SGM method identified 601 spectra missed by TAG-1 and 385 spectra missed by TAG-2 (Fig. 3), showing that the filtration efficiency of the SGM algorithm is much better than the two tag-based algorithms.

IV. Conclusions

In this paper, we proposed an efficient spectrum graph-based filtering algorithm for top-down mass spectral identification and tested the algorithm on two real top-down MS data sets. Compared with tag-based methods, the SGM filtering algorithm circumvents the steps of tag generation and searches spectrum graph against the protein database directly, simplifying data processing and increasing filtration efficiency. The experimental results on real data demonstrate that the SGM filtering algorithm outperforms the two tag- based algorithms in speed and filtration efficiency.

Acknowledgments

The research was supported by the National Institute of General Medical Sciences, National Institutes of Health (NIH) through Grant R01GM118470, and the National Natural Science Foundation of China (NSFC) through Grants 61472222 and 61732009.

References

1. Catherman AD, Skinner OS, Kelleher NL. Top down proteomics: facts and perspectives. Biochem Bioph Res Co. 2014; 445:683–93.
2. Roth MJ, Forbes AJ, Boyne MT, Kim Y-B, Robinson DE, Kelleher NL. Precise and parallel characterization of coding polymorphisms, alternative splicing, and modifications in human proteins by mass spectrometry. Mol Cell Proteomics. 2005; 4:1002–1008. [PubMed: 15863400]
3. Liu X, Sirotkin Y, Shen Y, Anderson G, Tsai YS, Ting YS, Goodlett DR, Smith RD, Bafna V, Pevzner PA. Protein identification using top-down spectra. Mol Cell Proteomics. 2012; 11:M111-008 524.
4. Kou Q, Wu S, Tolić N, Paša-Tolić L, Liu Y, Liu X. A mass graph-based approach for the identification of modified proteoforms using top-down tandem mass spectra. Bioinformatics. 2017; 33:1309–1316. [PubMed: 28453668]
5. Zamdborg L, LeDuc RD, Glowacz KJ, Kim Y-B, Viswanathan V, Spaulding IT, Early BP, Bluhm EJ, Babai S, Kelleher NL. ProSight PTM 2.0: improved protein identification and characterization for top down mass spectrometry. Nucleic Acids Res. 2007; 35:W701–W706. [PubMed: 17586823]
6. Kou Q, Xun L, Liu X. TopPIC: a software tool for top-down mass spectrometry-based proteoform identification and characterization. Bioinformatics. 2016; 32:3495–3497. [PubMed: 27423895]
7. Bode AM, Dong Z. Post-translational modification of p53 in tumorigenesis. Nat Rev Cancer. 2004; 4:793–805. [PubMed: 15510160]
8. Frank AM, Pesavento JJ, Mizzen CA, Kelleher NL, Pevzner PA. Interpreting top-down mass spectra using spectral alignment. Anal Chem. 2008; 80:2499–2505. [PubMed: 18302345]
9. Sun RX, Luo L, Wu L, Wang RM, Zeng WF, Chi H, Liu C, He SM. pTop 1.0: a high-accuracy and highefficiency search engine for intact protein identification. Anal Chem. 2016; 88:3082–90. [PubMed: 26844380]
10. Park J, Piehowski PD, Wilkins C, Zhou M, Mendoza J, Fujimoto GM, Gibbons BC, Shaw JB, Shen Y, Shukla AK, Moore RJ, Liu T, Petyuk VA, Tolić N, Paša-Tolić L, Smith RD, Payne SH, Kim S. Informed-Proteomics: open-source software package for top-down proteomics. Nat Methods. 2017; 14:909–914. [PubMed: 28783154]
11. Tanner S, Shu H, Frank A, Wang LC, Zandi E, Mumby M, Pevzner PA, Bafna V. InsPecT: identification of posttranslationally modified peptides from tandem mass spectra. Anal Chem. 2005; 77:4626–39. [PubMed: 16013882]
12. Liu X, Mammana A, Bafna V. Speeding up tandem mass spectral identification using indexes. Bioinformatics. 2012; 28:1692–1697. [PubMed: 22543365]
13. Mann M, Wilm M. Error-tolerant identification of peptides in sequence databases by peptide sequence tags. Anal Chem. 1994; 66:4390–4399. [PubMed: 7847635]
14. Tabb DL, Ma Z-Q, Martin DB, Ham A-JL, Chambers MC. DirecTag: accurate sequence tags from peptide MS/MS through statistical scoring. J Proteome Res. 2008; 7:3838–3846. [PubMed: 18630943]
15. Jeong K, Kim S, Bandeira N, Pevzner PA. Gapped spectral dictionaries and their applications for database searches of tandem mass spectra. Mol Cell Proteomics. 2011; 10:M110.002220.
16. Ng, J., Amir, A., Pevzner, PA. The 15th Annual International Conference on Research in Computational Molecular Biology, RECOMB 2011. Springer; 2011. Blocked pattern matching problem and its applications in proteomics; p. 298-319.

17. Deng F, Wang L, Liu X. An efficient algorithm for the blocked pattern matching problem. Bioinformatics. 2015; 31:532–538. [PubMed: 25322837]

18. Liu X, Inbar Y, Dorrestein PC, Wynne C, Edwards N, Souda P, Whitelegge JP, Bafna V, Pevzner PA. Deconvolution and database search of complex tandem mass spectra of intact proteins: a combinatorial approach. Mol Cell Proteomics. 2010; 9:2772–2782. [PubMed: 20855543]

19. Frank A, Pevzner P. PepNovo: de novo peptide sequencing via probabilistic network modeling. Anal Chem. 2005; 77:964–973. [PubMed: 15858974]

20. Cao X, Nesvizhskii AI. Improved sequence tag generation method for peptide identification in tandem mass spectrometry. J Proteome Res. 2008; 7:4422–4434. [PubMed: 18785767]

21. Wu CH, Apweiler R, Bairoch A, Natale DA, Barker WC, Boeckmann B, Ferro S, Gasteiger E, Huang H, Lopez R, Magrane M, Martin MJ, Mazumder R, O'Donovan C, Redaschi N, Suzek B. The Universal Protein Resource (UniProt): an expanding universe of protein information. Nucleic Acids Res. 2006; 34:D187–D191. [PubMed: 16381842]

22. Kessner D, Chambers M, Burke R, Agus D, Mallick P. ProteoWizard: open source software for rapid proteomics tools development. Bioinformatics. 2008; 24:2534–2536. [PubMed: 18606607]

23. "MS-Align+Tag," http://bioinf.spbau.ru/proteomics/ms-align-plus-tag, 2012

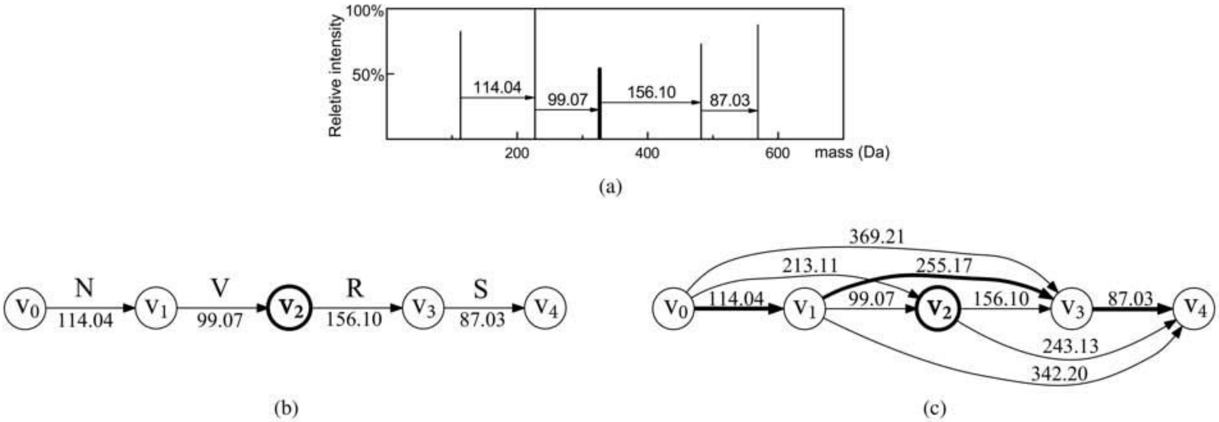


Figure 1.

Illustration of spectrum graph generation using an example deconvoluted MS/MS spectrum of the protein LNRVSG. (a) In the spectrum, the mass of the N-terminal fragment LNR is missing, and there is a noise mass peak (bold) between the fragment masses of LNR and LNRV. (b) In the spectrum graph, each node corresponds to a peak in the spectrum. Two nodes are connected by a directed edge if the difference between their corresponding masses matches the residue mass of one amino acid; the edge is labeled with the amino acid. The sequence tag NVRS extracted from the spectrum is incorrect because of the noise mass peak and its node v_2. (c) In the spectrum graph, each node corresponds to a peak in the spectrum. Two nodes are connected by a directed edge if the difference between their corresponding masses is less than 400 Da and matches the residue mass of one or several amino acids; the edge is labeled by the mass difference. The mass sequence of a path is a blocked pattern of the spectrum. For example, the bold path v_0, v_1, v_3, v_4 corresponds to a blocked pattern 114.04, 255.17, 87.03, which matches a correct sequence tag NRVS because 255.07 is the sum of the mass 156.10 of R and the mass 99.07 of V.

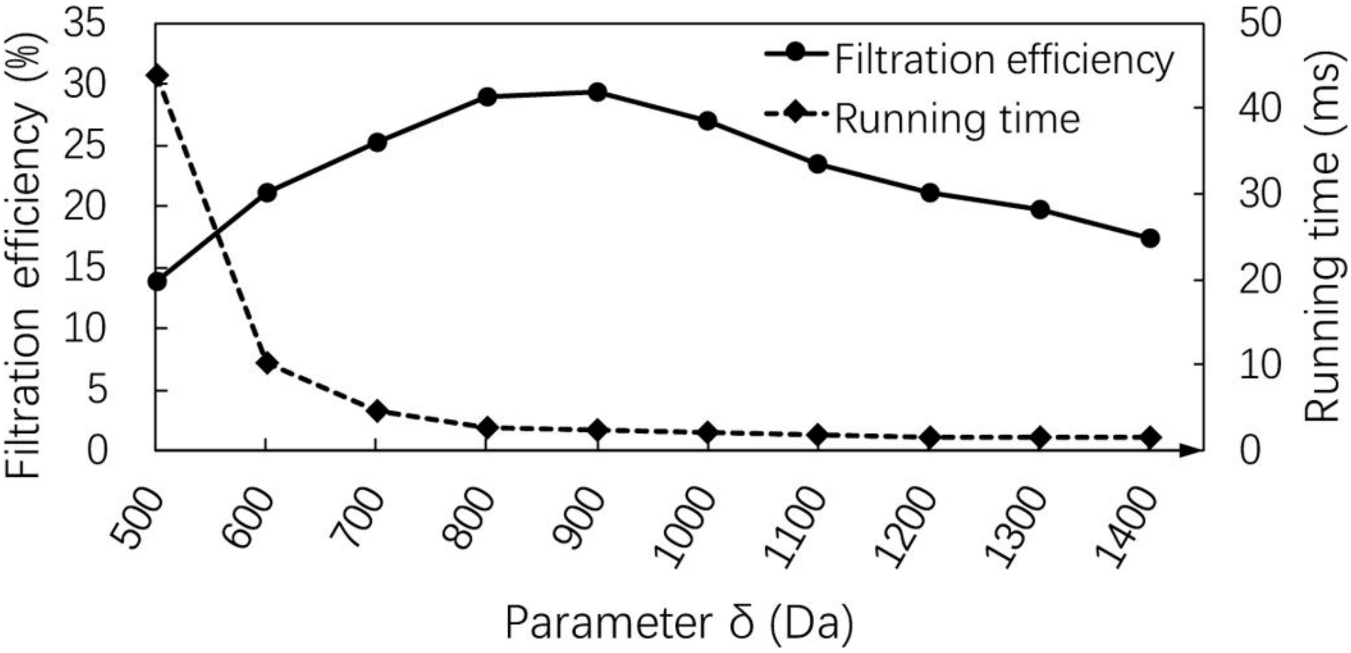


Figure 2.

The filtration efficiency and running time of the SGM filtering algorithm with various settings of the parameter δ from 500 to 1400 Da on the EC evaluation data set, when α = 300 Da, β = 200 Da, no masses are removed in the spectral preprocessing ($\lambda = \infty$), and the error tolerance ε is 0.02 Da.

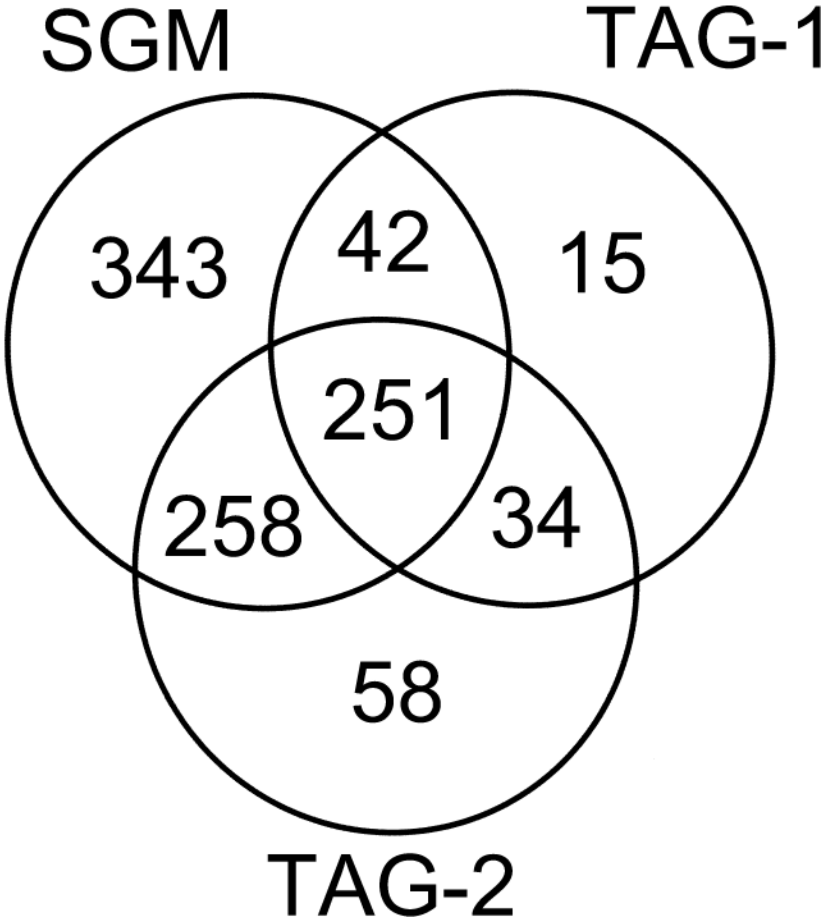
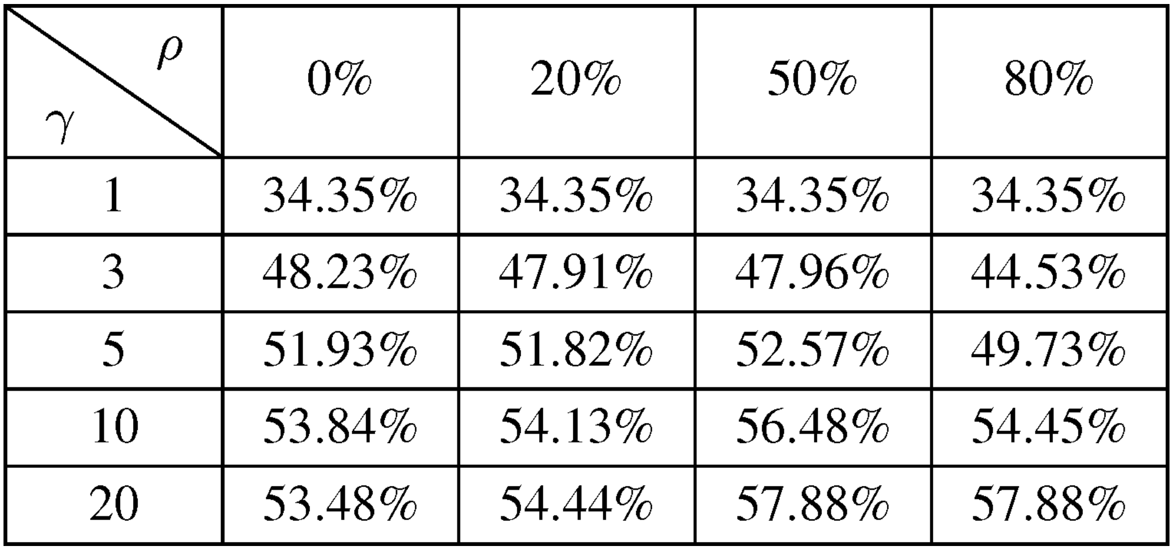


Figure 3.

Comparison of the numbers of proteoform spectrum-matches identified from the MCF-7 data set by the SGM, TAG-1, and TAG-2 filtering algorithms coupled with the spectral alignment algorithm in TopPIC with a 1% spectrum-level FDR.

Table I

FILTRATION EFFICIENCY RATES OF THE SGM FILTERING ALGORITHM WITH VARIOUS SETTINGS OF THE PARAMETERS γ AND ρ ON THE EC EVALUATION DATA SET.



ρ γ	0%	20%	50%	80%
1	34.35%	34.35%	34.35%	34.35%
3	48.23%	47.91%	47.96%	44.53%
5	51.93%	51.82%	52.57%	49.73%
10	53.84%	54.13%	56.48%	54.45%
20	53.48%	54.44%	57.88%	57.88%