Fast peak error correction algorithms for proteoform identification using top-down tandem mass spectra

Zhan, Zhaohui; Wang, Lusheng

Published in:
Bioinformatics

Published: 01/04/2024

Document Version:
Final Published version, also known as Publisher's PDF, Publisher's Final version or Version of Record

License:
CC BY

Publication record in CityU Scholars:
Go to record

Published version (DOI):
10.1093/bioinformatics/btae149

Publication details:

Citing this paper
Please note that where the full-text provided on CityU Scholars is the Post-print version (also known as Accepted Author Manuscript, Peer-reviewed or Author Final version), it may differ from the Final Published version. When citing, ensure that you check and use the publisher's definitive version for pagination and other details.

General rights
Copyright for the publications made accessible via the CityU Scholars portal is retained by the author(s) and/or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights. Users may not further distribute the material or use it for any profit-making activity or commercial gain.

Publisher permission
Permission for previously published items are in accordance with publisher's copyright policies sourced from the SHERPA RoMEO database. Links to full text versions (either Published or Post-print) are only available if corresponding publishers allow open access.

Take down policy
Contact lbscholars@cityu.edu.hk if you believe that this document breaches copyright and provide us with details. We will remove access to the work immediately and investigate your claim.
Sequence analysis

Fast peak error correction algorithms for proteoform identification using top-down tandem mass spectra

Zhaohui Zhan 1 and Lusheng Wang 1,2,*

1 Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong, China
2 City University of Hong Kong Shenzhen Research Institution, Shenzhen, 518057, China

*Corresponding author. Department of Computer Science, 83 Tat Chee Ave, City University of Hong Kong, Hong Kong, China. E-mail: cswangl@cityu.edu.hk (L.W.)

Abstract

Motivation: Proteoform identification is an important problem in proteomics. The main task is to find a modified protein that best fits the input spectrum. To overcome the combinatorial explosion of possible proteoforms, the proteoform mass graph and spectrum mass graph are used to represent the protein database and the spectrum, respectively. The problem becomes finding an optimal alignment between the proteoform mass graph and the spectrum mass graph. Peak error correction is an important issue for computing an optimal alignment between the two input mass graphs.

Results: We propose a faster algorithm for the error correction alignment of spectrum mass graph and proteoform mass graph problem and produce a program package TopMGFast. The newly designed algorithms require less space and running time so that we are able to compute global optimal alignments for the two input mass graphs in a reasonable time. For the local alignment version, experiments show that the running time of the new algorithm is reduced by 2.5 times. For the global alignment version, experiments show that the maximum mass errors between any pair of matched nodes in the alignments obtained by our method are within a small range as designed, while the alignments produced by the state-of-the-art method, TopMG, have very large maximum mass errors for many cases. The obtained alignment sizes are roughly the same for both TopMG and TopMGFast. Of course, TopMGFast needs more running time than TopMG. Therefore, our new algorithm can obtain more reliable global alignments within a reasonable time. This is the first time that global optimal error correction alignments can be obtained using real datasets.

Availability and implementation: The source code of the algorithm is available at https://github.com/Zeirdo/TopMGFast.

1 Introduction

Recent studies show that changes in protein isoform expression and post-translation modifications (PTMs) have gained recognition for their roles in underlying disease mechanisms (Brown et al. 2020). Besides, all protein products that arise from a single gene due to genetic variations, alternative splicing, and PTMs can significantly diversify the proteome for basic and clinical research (Melby et al. 2021). Proteoform identification is a challenging problem that has been attracting lots of attention.

Traditional ‘bottom-up’ mass spectrometry (MS)-based proteomics has become less practical for high-throughput proteoform identification due to the ‘peptide-to-protein’ inference problem with the low protein sequence coverage and a loss in proteoform information (Toby et al. 2016, Schaffer et al. 2019). For this reason, ‘top-down’ MS-based proteomics was coined by McLafferty and co-researchers in 1999. This ‘top-down’ MS-based proteomics analyzes intact proteins instead of digesting them into peptides, which enables accurate proteoform identification, PTM localization, and relative quantification for well understanding essential biological functions, unraveling disease mechanisms and discovering new biomarkers (Shaw et al. 2013, Riley et al. 2017, Schaffer et al. 2018, Dai et al. 2019, Arauz-Garofalo et al. 2021, Melby et al. 2021). Besides, when multiple PTMs coexist on single protein molecules, ‘top-down’ MS-based proteomics becomes the only feasible method for characterization (Holt et al. 2019).

For proteoform identification, we need to align a top-down spectrum against a protein database. To handle the combinatorial explosion of possible proteoforms, the proteoform mass graph was proposed (Kou et al. 2017), and a program package TopMG was produced. Moreover, a top-down spectrum can be represented as a spectrum mass graph. The problem of searching a spectrum against a database is formulated as finding an optimal alignment between the two mass graphs. Here we design new algorithms for aligning the two mass graphs with special treatment on spectrum peak error handling.

When aligning the peaks of the spectrum with proteoform mass graphs, the theoretical mass values on the proteoform mass graphs are not necessarily identical to the mass values of corresponding peaks since the mass values of peaks have errors. In previous methods, an error tolerance value δ is used to handle this issue and for any two consecutive nodes in the obtained alignment, the masses between the two consecutive nodes in the spectrum mass graph is within the range [x − δ, x + δ], where x is the theoretical mass between the two consecutive nodes in the proteoform mass graph. This
A spectrum is also formulated as a *spectrum mass graph* (SMG for short), where there is a special node \( y_0 \), each peak \( p_i \) with mass \( m_i \) in the spectrum corresponds to a node \( y_i \) for \( i > 0 \) in the SMG, there is a directed edge connecting \( y_i \) and \( y_{i+1} \) with mass \( m_{i+1} - m_i \) such that the length of the path from \( y_0 \) to \( y_i \) is \( m_i \). For a pair of peaks \( p_i \) and \( p_j \) with \( i < j \) in the spectrum, if \( m_j - m_i \) is the same as an amino acid or the modification of an amino acid in a protein, we can match \( y_i \) and \( y_j \) to the two nodes corresponding to the amino acid. Besides, following the same way as in Kou et al. (2017), we convert each mass value \( m \) in both PMG and SMG into an integer by using the formula \( \lfloor m + 274.335215 \rfloor \).

However, since the masses of peaks have errors, we cannot assume that \( m_j - m_i \) is identical to the theoretical mass of an amino acid. Some kind of peak error handling is required. Usually, people use a value \( \delta \) for error tolerance.

**Two ways to set error tolerance value** \( \delta \): There are two ways to set the error tolerance value \( \delta \) (Kou et al. 2016, 2017, Zhan and Wang 2022). One way is to simply set \( \delta = 27 \) for every peak. The second method sets a value \( \delta_i \) to be the error for \( m_i \) for each peak \( y_i \) and we call this kind of error tolerance peak-dependent error tolerance. We can calculate the peak dependent error tolerance for each peak \( y_i \) as follows.

1. For an original peak \( y_i \) with mass \( m_i \), \( \delta_i = 27 + \frac{15}{1000}m_i \).
2. For a complementary peak \( y_i \) with mass \( m_i \), \( \delta_i = 27 + \frac{3}{1000}m_i + M \), where \( M \) is the mass of the whole protein.
3. For any peak \( y_i \) with mass \( m_i \) larger than 5000, the corresponding \( \delta_i \) should be further enlarged. An alternative way (Kou et al. 2016, 2017) to handle this kind of peaks is to add another two peaks with masses \( m_i - 1.00235 \) and \( m_i + 1.00235 \) in pre-process stage. After multiplying the factor of 274.335215, there are three peaks that are about 274.335215 away in the spectrum.

For item 3, there is a trade off between increasing the number of peaks in the spectrum and increasing the value of \( \delta_i \). We observe that increasing the number of peaks is a better choice in terms of the speed of algorithms since the value of \( \delta_i \) plays an important role for the algorithm with error correction. We denote \( \delta_{\max} = \max_{i \geq 0}\delta_i \).

The traditional method (Kou et al. 2017) computes an alignment between the two graphs (PMG and SMG). The alignment contains a list of nodes \( x_{1}, x_{2}, \ldots, x_{n} \) from the proteoform mass graph and a list of nodes \( y_{1}, y_{2}, \ldots, y_{k} \) from the spectrum mass graph such that for any two consecutive nodes \( y_{i} \) and \( y_{i+1} \) in the alignment, \( m_{i+1} - m_{i} \) is in the range \([m_{\text{th}} - \delta, m_{\text{th}} + \delta]\) where \( m_{\text{th}} \) is the theoretical mass for a path between \( x_{i} \) and \( x_{i+1} \) in PMG. We refer to such kind of error tolerance method as local edge tolerance methods. The local edge tolerance methods may suffer from error accumulation if positive/negative errors occur for many edges in the proteoform mass graph (PMG for short) (Kou et al. 2017). Each amino acid in a protein has a left node and a right node corresponding to the bonds left and right to the amino acid. An edge connecting the two nodes is assigned the mass of the amino acid. Each edge corresponding to an original residue has a black color. For each modification of an amino acid, there is a new edge with the modified mass connecting the two nodes of the original amino acid. The edges corresponding to modifications of amino acids have a red color. The PMG has a unique starting node and an ending node.
Fast proteoform identification with peak error corrections for computation of alignments. Let $s$ be the spectrum. They propose to have an error correction for spectrum mass graph (SMG) to represent the database and they still use the proteoform mass graph (PMG) and the Zhan and Wang 2022. 

Let $x_{i_0}, x_{i_1}, ..., x_{i_n}$ be the $n$ nodes in a PMG and $y_{j_0}, y_{j_1}, ..., y_{j_m}$ be the $m$ nodes in a SMG. An error correction alignment of $G$ and $H$ with size $r$ is a sequence of $r$ triples $(x_{i_0}, y_{j_0}, k_0), (x_{i_1}, y_{j_1}, k_1), ..., (x_{i_{r-1}}, y_{j_{r-1}}, k_{r-1})$ such that $m_{i_0} + k_0 - (m_{i_1} + k_1) = M_{i_1} - L_{i_1}$, where $k_0 \in (-\delta_i, \delta_i)$ is the error correction value for peak $y_{j_0}$, $m_{i_0}$ is the mass for the peak $y_{j_0}$ and $M_{i_1}$ is the mass of path between node $x_{i_{r-1}}$ and node $x_{i_r}$. Note that $\delta$ is the error tolerance value for the mass of the peak. The error correction alignment problem is to compute an error correction alignment between $G$ and $H$ with the maximum size.

Let $T(i, j, k)$ be the maximum size of the alignments between the first $i$ peaks in $H$ and the first $j$ nodes in $G$ such that the corrected peak of $y_i$ has the mass value $m_{i+k}$, and $y_i$ in $H$ matches $x_j$ in $G$. Note that the initial value of $T(i, j, k)$ is set to be 1 for all $0 \leq i \leq m$, $0 \leq j \leq n$ and $-\delta_i \leq k \leq \delta_i$.

When computing $T(i, j, k)$ for every $0 \leq i \leq m$, $0 \leq j \leq n$ and $-\delta_i \leq k \leq \delta_i$, a dynamic programming algorithm can be used to simplify the process. Let $d(i, j)$ be the set of distinct masses for paths from $x_i$ to $x_j$ in $G$. For each mass $m \in d(i, j)$, there is a list of nodes corresponding to peaks in $H$ with mass values in the range $[m_{i+k} - \delta_{max}, m_{i+k} + \delta_{max}]$. Let $l(i, j, k, m)$ be such a sorted list. The peaks in this list can be matched to the node $x_j$ in $G$ under the condition that the corrected peak of $y_i$ with mass $m_{i+k}$ matches the node $x_j$ in $G$. Therefore, the following dynamic programming equation can be used for computing $T(i, j, k)$:

$$T(i, j, k) = \max_{\begin{array}{l}
\delta_{i}^{-} \leq k \leq \delta_{i}^{+} \\
\forall (i', j') \text{ satisfying (2)}
\end{array}} \max_{i' \leq i, j' \leq j} \{T(i', j') + 1\}, \quad (1)$$

where condition (2) is as follows:

$$m - (m_{i+k} - m_{i-k}) = k'. \quad (2)$$

When computing $l(i, j, k, m)$, let $C(i, j)$ be the set of all $l(i, j, k, m)$ and can be formulated as $C(i, j) = \{l(i, j, k, m)\}$ $m \in \bigcup_{i'=0}^{i'} d(i', j')$. A node $n_{max} \in [\delta_{max}, \delta_{max}]$. Let $D = \bigcup_{i'=0}^{i'} \bigcup_{j'=0}^{j'} d(i', j')$ be the set of masses or subpaths in $G$ that we consider for computation of alignments. Let $M = \{(m, i, i') | i' < i, y_i, y_{i'} \in H, m = m_{i-k} - m_{i-k'}\}$ be the set of triples for masses differences $m$ between any pair of nodes $(y_i, y_{i'})$ in $H$. Then, $D$ and $M$ are both sorted in nondecreasing order. Going through the elements in sorted $D$ and $M$ once, we can create $C(i, j)$ with all the lists $l(i, j, k, m)$ sorted.

Moreover, for a specific peak $y_i$, a node $x_i$ and a path mass $m \in \bigcup_{i'=0}^{i'} d(i', j')$ in $G$, one element in $l(i, j, k, m)$ is enough to compute $T(i, j, k)$ for all $k \in (-\delta_i, \delta_i)$ instead of using equation (1) for all $k \in (-\delta_i, \delta_i)$. [See Theorem 1 (Zhan and Wang 2022).] Thus, the total running time of computing all $T(i, j, k)$ and finding the largest one is $O(nmq\delta_{max} + L)$, where $L$ is the total size of $\bigcup_{i'=0}^{i'} d(i', j')$.

The above algorithm is still too slow in practice. Thus, Zhan and Wang designed a local algorithm that can only compute an optimal alignment when the starting positions in both $G$ and $H$ are given. In this case, a sub-matrix of $T(i, j, k)$ around the diagonal will be computed so that the running time and the memory usage are further reduced. The detailed algorithm can be found in the Supplementary Section S1A.

2.2 New fast algorithms

When computing $T(i, j, k)$, for a peak $y_i$, the corrected position $m_i + k$ of $y_i$ is an integer in the range $[m_{i-\delta_i}, m_{i+\delta_i}]$. When the next peak $y_{i+1}$ in the spectrum is close to $y_i$, the two ranges $[m_{i-\delta_i}, m_{i+\delta_i}]$ and $[m_{i+1-\delta_i}, m_{i+1+\delta_i}]$ may have overlap. In this case, $T(i, j, k)$ and $T(i+1, j, k')$, where $m_i+k = m_{i+1}+k'$, mean the same thing, i.e. a peak in the spectrum with mass $m_{i+k} = m_{i+1}+k'$ matches $x_j$ in $G$. It does not matter whether $y_i$ or $y_{i+1}$ is corrected to the position $m_{i+k} = m_{i+1}+k'$ since Theorem 1 in Zhan and Wang (2022) ensures that $T(i, j, k)$ is always equal to $T(i+1, j, k_1)$ under the condition $m_{i+k} > \delta_{max}$, where $m_{i+k}$ is the smallest mass of all the (modified or unmodified) residues. This condition is true in practice for different error tolerance settings (Zhan and Wang 2022).

Thus, each integer in the overlapped range $[m_{i-\delta_i}, m_{i+\delta_i}] \cap [m_{i+1-\delta_i}, m_{i+1+\delta_i}]$ is used twice in the computation process of $T(i, j, k)$. See Fig. 1. Therefore, we can make the algorithm faster if every integer in the overlapped range is used once during the whole process of computing $T(i, j, k)$.

To reduce such redundant computing, we propose an algorithm to delete overlaps when computing $T(i, j, k)$ and ensure every available position for peaks is computed only once. In the new algorithm, for each peak, we define two variables to indicate the lower and upper bounds of the range, where the correct peak position should be. Let $\delta_i'$ denote the largest negative error tolerance for peak $y_i$. Similarly, let $\delta_i''$ denote the largest positive error tolerance for peak $y_i$. The initial values of $\delta_i'$ and $\delta_i''$ for peak $y_i$ are both $\delta_i$ before deleting overlaps.

For any consecutive peaks $y_i$ and $y_{i+1}$, if the range $[m_i-\delta_i', m_{i+1}+\delta_i'']$ is completely included in the range $[m_i-\delta_i', m_{i+1}+\delta_i'']$, then $y_i$ can be deleted from the spectrum at the beginning of this algorithm.

We then re-calculate the ranges for peak $y_i$ and $y_{i+1}$ according to the principles illustrated in the Supplementary Section S1B.

Now, when we compute $T(i, j, k)$, the range of the element $k$ is reduced and the total size of $T(i, j, k)$ we need to compute is also decreased. We still use Equation (1) to compute $T(i, j, k)$. Thus, the total running time is $O(nmq\delta_{max} + L)$, where $q$ is the largest size of $\delta_i'$ and $\delta_i''$ for all peaks.

Besides, in the process creating $C(i, j)$, we also change the formulation of $C(i, j)$ by the updated error tolerance for
4 Results
3.1 Datasets
Here, we use a dataset generated from Escherichia coli (EC) K-12 MG1655 cells. The protein database was downloaded from UniProt (Proteome ID: UP000000625) and included 4438 protein entries associated with this proteome with F plasmid removed. For MS and MS/MS spectra, we use the raw features downloaded from Kou et al. (2017) and further process the data following the methods described in Kou et al. (2017). After processing, we obtained 4054 top-down MS/MS spectra, where 2027 are collision-induced dissociation (CID) MS/MS spectra and the other 2027 are electron-transfer dissociation (ETD) MS/MS spectra. For this EC dataset (database), three mutations were used as variable PTMs and the three modifications are: lysine (K) to cysteine (C) (UNIMOD Accession number: 1132), threonine (T) to alanine (A) (UNIMOD Accession number: 659) and valine (V) to glycine (G) (UNIMOD Accession number: 659). A txt format file was generated based on these three pre-defined mutations as part of the input. Users can provide their own predefined mutations to replace the txt format file. The original EC database was modified based on the txt format file to form the final proteoform database for database search. For the EC protein database, the protein size (the total number of amino acids) ranges from 31 to 2001, while the spectrum size (the total number of peaks) ranges from 22 to 4604 among 4054 top-down MS/MS spectra.

3.2 Speed and memory of (local) diagonal alignment version
The best-known method with error correction is TopMGRefine (Zhan and Wang 2022), where they can only provide a diagonal alignment version reporting an optimal local alignment due to the high time/space complexity of their algorithm. The diagonal alignment version requires users to input the starting positions of the alignment and has a constraint that the two masses from the two starting positions in the alignment to any pair of aligned nodes are roughly the same. To illustrate the time/space complexity of the new algorithm, we start with the diagonal alignment version and follow the experiment processes in Zhan and Wang (2022).

Similar to TopMGRefine, since diagonal alignment needs the starting positions as part of the input, we use TopMG (Kou et al. 2017) to align all the 4054 spectra with all the proteins for the whole EC dataset. Note that, TopMG first uses a filtering method to obtain a few candidate proteins with high scores for each spectrum and then uses an alignment algorithm to further align the spectrum with each of the selected candidates and report the protein and the corresponding alignment with the best score. We refer to this version as TopMG with filtering.

Among 4054 spectra, TopMG reported 2817 spectra that can be successfully aligned to some proteins in the database. Then we choose the protein with the largest alignment size generated by TopMG for each of the 2817 spectra to form protein and spectrum pairs for further refined alignments using TopMGRefine and TopMGFast.

The experiments were performed on a cluster with 200 GB of memory. For the 2817 protein and spectrum pairs, both TopMGRefine and TopMGFast obtain identical local alignment results. The total running time of the 2817 protein and spectrum pairs for TopMGRefine and TopMGFast is 4760 and 1715 min, respectively.

The largest input instance is the protein sp/P76347|YEEJ_ECOLI with 2001 residues and the spectrum (ID: 3260) with 4604 peaks. For this largest input instance, the starting position for protein and the starting position for the spectrum given by TopMG are 813 and 0, respectively. In this case, the space required by TopMGRefine and TopMGFast is 102G and 80G, respectively. The alignment size obtained by both TopMGRefine and TopMGFast is 83. The running time for TopMGFast and TopMGRefine is 21.4 and 56.9 min, respectively. Again, the running time of TopMGFast is about 1/3 of that for TopMGRefine. Therefore, we can see that TopMGFast and TopMGRefine can obtain the same results, where TopMGFast needs much less running time and memory space.

3.3 Preliminary performance of TopMGFast for global alignment
Since TopMGFast needs less memory space, it is possible to do global alignment even for the largest input instance, where TopMGRefine cannot do that. For global alignment, we do not need to give the starting positions of the alignment for a

Figure 1. The figure illustration example for the error tolerance overlaps. The range between the position \(mass_{i+1} - \delta_{i+1}\) and \(mass_{i} + \delta_{i}\) is the overlap and the positions like the point A in it will be computed twice.
pair of protein and spectrum. Therefore, we can try to compute the final accurate alignment result without using the roughly estimated starting positions of the alignment.

In the rest of this subsection, we will compare the alignment algorithms for TopMG and TopMGFast using 4054 top-down MS/MS spectra and 4438 proteins in the whole EC database. Since both the protein and spectrum databases are huge compared to the slow speeds of the alignment algorithms for both TopMG and TopMGFast, we first run the filtering algorithm in TopMG to obtain a few candidates for each spectrum, and for the same set of obtained candidate proteins, run the alignment algorithms for both TopMG and TopMGFast with the same spectrum to get the best alignment results.

Among 4054 spectra in the whole EC dataset, TopMG reported 2817 spectra with at least one corresponding protein candidate and the total number of reported corresponding protein candidates is 31481. Each reported spectrum corresponds to a few candidate proteins with large alignment scores. We then align them with each other and report the final alignments by using TopMGFast directly. This time, no starting points are required. Besides, we used two kinds of error tolerances for peaks when computing alignment results for both TopMG and TopMGFast, one is constantly equal to 27 and the other is the peak-dependent error tolerance computed in Section Methods. The experiments were also performed on a cluster with 200 GB of memory.

3.3.1 Results for error tolerance 27

The results with error tolerances 27 are shown in Tables 1 and 2. As shown in Table 1, the running time of TopMGFast is 8702 min which is about 6.6 times that of TopMG when setting the alignment error tolerance to be 27. Among 2817 selected spectra, TopMG and TopMGFast can report the same proteins for 1396 spectra and different proteins for the other 1421 spectra, respectively. Furthermore, among 1396 spectra reporting the same proteins from TopMG and TopMGFast, 1082 spectra report roughly the same alignment locations (resulting in alignments with overlaps), while 314 spectra report completely different alignment locations.

For both 1396 (reporting identical proteins) and 1421 (reporting different proteins) spectra groups, the numbers of matches obtained from TopMGFast are slightly larger than that of TopMG. (24.61 versus 23.19 and 17.75 versus 13.34, respectively.) Consequently, the average numbers of one residue and two residue matches for reported alignments obtained from TopMGFast are also slightly larger than those obtained from TopMG. Since mass matches of one or two residues are more reliable than mass matches corresponding to a sum of a large number of residues, this might be evidence that the alignments reported by TopMGFast are more reliable than those of TopMG. It is interesting to observe that the number of 1 residue matches is slightly <50% of total matches.

The reason that TopMGFast can obtain more matches than TopMG is that TopMGFast requires that the difference between the locations of a peak and the corrected peak is bounded by 27, where TopMG requires that the mass between two consecutive matched peaks is at most 27 away from the corresponding mass in the protein. Thus, it is possible that the tolerated error of the mass between two consecutive matched peaks for TopMGFast is larger than that of TopMG. However, TopMGFast can ensure that there is no error for a mass between any pair of matched peaks after error correction.

To further compare the quality of resulting alignments, we use two measures, the maximum mass error (MME) between two matched peaks in an alignment and the average mass error (AME) between all pairs of matched peaks. The maximum mass error (MME) between two matched peaks in an alignment is the largest error among all the $\binom{n}{2}$ pairs of $n$ matched peaks in an alignment. The average mass error (AME) between two matched peaks in an alignment is the average error (comparing to the theoretical masses in the protein database) among all the $\binom{n}{2}$ pairs of $n$ matched peaks in an alignment.

Here, we plot the differences between TopMG’s and TopMGFast’s MME and AME values (MME/AME_diff) for those spectra, where both methods report the same proteins in Fig. 2. MME/AME_diff can be computed by the formula $MME/AME_{diff} = TopMGsMME/AME - TopMGFast/sMME/AME$. As shown in Fig. 2, 1396 spectra that report the same proteins have been sorted by MME in nondecreasing orders. For these 1396 spectra, there are 347 spectra in which $-35 \leq MME_{diff} < 0$ in the left figure. This means that TopMGFast’s alignment results have worse MME values compared with TopMG’s alignment results for these 347 spectra. The worst case is the spectrum that $MME_{diff} = -35$. Similarly, the numbers of spectra for the conditions $MME_{diff} = 0$, $0 < MME_{diff} \leq 35$ and

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Running time (min)</th>
<th>Alignment size</th>
<th>Same protein</th>
<th>Different proteins</th>
</tr>
</thead>
<tbody>
<tr>
<td>TopMG</td>
<td>1320</td>
<td>23.19</td>
<td>13.34</td>
<td></td>
</tr>
<tr>
<td>TopMG</td>
<td></td>
<td>11.04</td>
<td>2.75</td>
<td></td>
</tr>
<tr>
<td>TopMG</td>
<td></td>
<td>2.97</td>
<td>1.31</td>
<td></td>
</tr>
<tr>
<td>TopMG</td>
<td></td>
<td>9.18</td>
<td>9.28</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td>8702</td>
<td>24.61</td>
<td>17.57</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td></td>
<td>11.49</td>
<td>3.08</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td></td>
<td>3.33</td>
<td>1.78</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td></td>
<td>9.79</td>
<td>12.71</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Average MMEs</th>
<th>Total</th>
<th>Same protein</th>
<th>Different proteins</th>
</tr>
</thead>
<tbody>
<tr>
<td>TopMG</td>
<td>49.43</td>
<td>36.06</td>
<td>66.16</td>
<td></td>
</tr>
<tr>
<td>TopMG</td>
<td>17.35</td>
<td>11.56</td>
<td>24.37</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td>35.14</td>
<td>26.87</td>
<td>45.49</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td>11.42</td>
<td>7.77</td>
<td>15.99</td>
<td></td>
</tr>
</tbody>
</table>
MME\(_{\text{diff}}\) > 35 are 258, 668, and 123, respectively. Hence, there are 791 spectra in which TopMGFast’s alignment results are better than TopMG’s alignment results. The best case is the spectrum that MME\(_{\text{diff}}\) = 252. The MME\(_{\text{diff}}\) is unreasonably large since TopMG has no constraint for any two matched peaks. This makes the TopMG’s alignment result unreliable. For AME\(_{\text{diff}}\), it also exhibits the same pattern.

Besides, for all 2817 spectra, the average MME and AME values are given in Table 2. The detailed diagrams are illustrated in Supplementary Section S2A. As shown in Supplementary Fig. S5, there is an upper bound \(28 = 2 \times 27 = 54\) for the MMEs reported by TopMGFast for all the 2817 alignments, while we can clearly observe that TopMG reported many alignments with unreasonably large MMEs. Note that, one pair of peaks with an unreasonably large error indicates that the alignment is not reliable.

### 3.3.2 Results for peak-dependent error tolerance

Tables 3 and 4 illustrate the results for the complex error tolerances. As we can see, the running time of TopMGFast without diagonal optimization is 34 240 min which is about 3.3 times that of TopMG. Among 2817 selected spectra, TopMG and TopMGFast can report the same proteins for 1580 spectra and different proteins for 1237 spectra, respectively. Furthermore, among 1580 spectra reporting the same proteins from TopMG and TopMGFast, 1150 spectra report roughly the same locations while 430 spectra report completely different locations. However, the alignment size obtained by TopMGFast is slightly smaller than that obtained from TopMG. The reason is that TopMGFast requires that the mass between any matched peaks (after peak error correction) be identical to the corresponding theoretical mass in the protein database, while TopMG requires any two consecutive matched peaks to have a mass error at most \(\delta\). TopMG has a high chance to generate alignments with huge MMEs. Figure 3 illustrates MME/AME\(_{\text{diff}}\) for those spectra, where both methods report the same proteins when we use the peak-dependent error tolerance. The worst cases are MME = −80 and AME = −17.8. The best cases are MME = 16 755 and AME = 6058.9. The huge MME/AME\(_{\text{diff}}\) values indicate the unreliability of TopMG, and our algorithm TopMGFast can generate more accurate alignments.

Besides, for all 2817 spectra, the average MME and AME values are given in Table 4. The detailed diagrams are illustrated in Supplementary Section S2B.

### 3.3.3 Comparison of reported alignment locations with the two error tolerance methods

We also compared the reported proteins with the two different error tolerance methods. The results are shown in Table 5. As we can see, both TopMG and TopMGFast reported different proteins for large amounts of spectra when using two kinds of error tolerances. Besides, according to Tables 1 and 3, for the averaged total number of reported matches obtained from all spectrum and proteoform candidate pairs, both TopMG and TopMGFast can report more matches when using peak-dependent error tolerance. This may be evidence that using the peak-dependent error tolerance rather than the immutable value 27 is a better choice for handling the peak errors.

### 3.4 Prediction accuracy evaluation

The preliminary performance analysis in the above subsection shows that TopMG and TopMGFast report different proteins in many cases for the same spectrum even if they use the same set of filtered candidate proteins. To illustrate the quality of the alignments for the two different alignment algorithms, we delete the filter step and directly use the two alignment algorithms to search the protein database. Without the filtering algorithm, both TopMG and TopMGFast are slow. Thus, we reduce the size of the database as follows. We randomly select 700 proteins from the same protein database mentioned above to form the testing protein database. Then we randomly select 100 proteins from the testing protein database and generate a simulated spectrum with 5 modifications for each of them using MaSS-Simulator (Awan and Saed 2018). In this way, for each of the simulated spectra, we know the real corresponding (modified) protein.

For the MaSS-Simulator, a protein sequence file and a modification file should be provided as the input files. Here, we use the sample modification file given by the paper (Awan and Saed 2018) with four kinds of mutations: Oxidation (UNIMOD Accession number: 35), Deamidation (UNIMOD...
Accession number: 7), Phosphorylation (UNIMOD Accession number: 21), and Carbamidomethylation (UNIMOD Accession number: 4). Besides, each protein is set to have five modifications in the generated spectrum.

Now, we have 100 simulated spectra with their known real corresponding proteins and a protein database with 700 proteins. The original database has >4000 proteins. The reason that we do not use the whole database is that the speeds of both TopMG and our method are very slow to directly search the whole database.

We use both TopMG and our method to directly align the 100 simulated spectra against the 700 proteins in the database and report the proteins with the best alignment score as the search results. According to the alignment results, TopMG reported 94 real corresponding proteins for the 100 cases. Thus, the prediction accuracy for TopMG is 94%.

Table 3. The comparisons between TopMG and TopMGFast using the same peak-dependent error tolerances without diagonal optimization on the filtered EC dataset.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Running time (min)</th>
<th>Alignment size</th>
<th>Reporting the same protein</th>
<th>Reporting different proteins</th>
</tr>
</thead>
<tbody>
<tr>
<td>TopMG</td>
<td>10 371</td>
<td>Total no. of matches 35.09</td>
<td>34.20</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No. of 1 residue matches 16.96</td>
<td>10.71</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No. of 2 residues matches 5.11</td>
<td>4.97</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No. of ≥3 residues matches 13.02</td>
<td>18.52</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td>34 240</td>
<td>Total no. of matches 32.64</td>
<td>29.78</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No. of 1 residue matches 15.02</td>
<td>6.46</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No. of 2 residues matches 4.73</td>
<td>3.95</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>No. of ≥3 residues matches 12.89</td>
<td>19.37</td>
<td></td>
</tr>
</tbody>
</table>

a For the reported proteins from TopMG and TopMGFast among 2817 spectra, the same reported protein can be obtained for 1580 spectra and different reported proteins can be obtained for 1237 spectra, respectively.

Among 1580 spectra reporting the same proteins from TopMG and TopMGFast, 1150 spectra report overlaps alignment results while 430 spectra report completely different alignment results.

Table 4. The error comparisons between TopMG and TopMGFast for 2817 alignments when using the peak-dependent error tolerance with overlaps deleted.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Mass error</th>
<th>Total</th>
<th>The same protein</th>
<th>Different proteins</th>
</tr>
</thead>
<tbody>
<tr>
<td>TopMG</td>
<td>Average MMEs 1043.95</td>
<td>823.05</td>
<td>1351.19</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Average AMEs 326.73</td>
<td>250.55</td>
<td>432.68</td>
<td></td>
</tr>
<tr>
<td>TopMGFast</td>
<td>Average MMEs 159.35</td>
<td>115.36</td>
<td>220.53</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Average AMEs 43.20</td>
<td>28.84</td>
<td>63.18</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3. The differences between TopMG’s and TopMGFast’s MME and AME values for those spectra reporting the same proteins when using the peak-dependent error tolerance. Here, MME/AME_diff = TopMG’s MME/AME−TopMGFast’s MME/AME. The 1580 spectra have been sorted by MME/AME_diff in nondecreasing order.

Table 5. The comparisons of reported proteins between two kinds of error tolerances for TopMG and TopMGFast.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Error tolerance</th>
<th>No. of matches</th>
<th>RT(min)</th>
<th>Different proteins</th>
<th>The same protein</th>
</tr>
</thead>
<tbody>
<tr>
<td>TopMG</td>
<td>27</td>
<td>18.22</td>
<td>1320</td>
<td>1343</td>
<td>1474</td>
</tr>
<tr>
<td></td>
<td>Complex</td>
<td>34.70</td>
<td>10371</td>
<td>966</td>
<td>1851</td>
</tr>
<tr>
<td>TopMGFast</td>
<td>27</td>
<td>21.06</td>
<td>8702</td>
<td>966</td>
<td>1851</td>
</tr>
<tr>
<td></td>
<td>Complex</td>
<td>31.38</td>
<td>34240</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Our method, TopMGFast, reported 100 real corresponding proteins for the 100 cases and the prediction accuracy is 100%.

Case study: TopMG failed the case, where the real corresponding protein is \( sp \mid P0A235 \mid RFC\_SALTY \). For this case, TopMG reported protein \( sp \mid P26465 \mid FLII\_SALTY \) with 53 matched peaks in the alignment. We checked the errors between every two adjacent matched peaks in this alignment for both TopMG and TopMGFast. We found that there are many consecutive negative errors (the first few errors are: \{4, −1, −20, −9, −16, −24, ... \}) in the alignment reported by TopMG. These consecutive negative errors make the maximum mass error for TopMG become 121 which is much larger than the user-defined error tolerance 27. However, the error between any two matched peaks in the alignment reported by TopMGFast is strictly no more than 2×27 = 54. Maybe this is the reason why TopMG reported the wrong protein and the evidence that error correction methods such as TopMGFast are more reliable than TopMG.

To test the performance accuracy for local alignment, we randomly select a peptide from each of the above 100 selected proteins. The lengths of those selected peptides are from 23 to 154. Each peptide contains three modifications. We then generate 100 simulated spectra for the 100 peptides and use both TopMG and TopMGFast to search the database containing 700 proteins. TopMG reported 98 real corresponding proteins with correct locations for the 100 cases. The prediction accuracy for TopMG is 98%. TopMGFast reported 100 correct spectra and the prediction accuracy is 100%.

4 Conclusion

We propose a faster algorithm for the error correction alignment of spectrum mass graph and proteoform mass graph problem and produce a program package TopMGFast in C++. The newly designed algorithms require less space and running time so that we are able to compute global optimal alignments for the two input mass graphs in a reasonable time. According to the experiment results for both local alignments and global alignments, our new algorithm can obtain more reliable alignments within a reasonable time. Besides, this is the first time that global optimal error correction alignments can be obtained using real datasets. Still, it is an interesting open problem to design better error handling methods.

Author contributions

Both Z.Z. and L.W. are involved in all aspects of the project. In particular, Z.Z. focuses on the implementation of the software package and reporting the experiment results. L.W. is responsible for algorithm design, experiment design, manuscript writing, etc.

Supplementary data

Supplementary data are available at Bioinformatics online.

Conflict of interest

None declared.

Funding

This work was fully supported by funds from the National Science Foundation [NSF: 61972329] and GRF grants for Hong Kong Special Administrative Region, P. R. China [CityU 11210119, CityU 11206120].

Data availability

The data underlying this article are available at https://github.com/Zeirdo/TopMGFast.

References


