Identification of Post-translational Modifications via Blind Search of Mass-Spectra

Dekel Tsur
Computer Science and Engineering
UC San Diego
dtsur@cs.ucsd.edu

Stephen Tanner
Bioinformatics
UC San Diego
stanner@ucsd.edu

Ebrahim Zandi
School of Medicine
Univ. Southern California
zandi@usc.edu

Vineet Bafna
Computer Science and Engineering
UC San Diego
vineet@cs.ucsd.edu

Pavel A. Pevzner
Computer Science and Engineering
UC San Diego
ppevzner@cs.ucsd.edu

Abstract

Post-translational modifications (PTMs) are of great biological importance. Most existing approaches perform a restrictive *search that can only take into account a few types of PTMs and ignore all others. We describe an* unrestrictive *PTM search algorithm that searches for* all *types of PTMs at once in a* blind *mode, i.e., without knowing which PTMs exist in a sample. The blind PTM identification opens a possibility to study the extent and frequencies of different types of PTMs, still an open problem in proteomics. Using our new algorithm, we were able to construct a two-dimensional* PTM frequency matrix *that reflects the number of MS/MS spectra in a sample for each putative PTM type and each amino acid. Application of this approach to a large IKKb dataset resulted in the largest set of PTMs reported for a single MS/MS sample so far. We demonstrate an excellent correlation between high values in the PTM frequency matrix and known PTMs thus validating our approach. We further argue that the PTM frequency matrix may reveal some still unknown modifications that warrant further experimental validation.*

1 Introduction

Fueled by recent improvements in instrumentation and software, tandem mass spectrometry has become the tool of choice for protein identification. However, while the popular algorithms like Sequest [8] and Mascot [16] are used extensively for peptide identification, many spectra remain unidentified by these algorithms. This can be attributed to several factors including poor quality of fragmentation/ionization, and the presence of post-translational modifications *(PTMs)* and mutations.

Identifying PTMs at a global level is undoubtedly the next big step for proteomics (Cantin and Yates, 2004 [5]). Post translational modifications greatly increase the complexity of the proteome and recent reports [3, 22] suggest that the extent of modifications is much larger than some earlier estimates. However, reliable computational identification of PTMs and mutations remains a formidable challenge. In theory, one could enumerate all of the combinatorial modifications for each peptide, score them, and find the best scoring peptide. In practice, however, the combinatorial explosion makes it harder to enumerate all but a few choice modifications. Mutations complicate the picture even further, as each residue can be mutated into a different residue.

The first approach to PTM identification was proposed by Yates et al., 1995 [25], who advocated the enumeration and scoring of all possible candidates. This exhaustive search approach has serious limitations since it can only take into account a few modifications and would be prohibitively slow for mutation detection. In other words, a researcher has to "guess" in advance which PTMs are present in the sample, an unrealistic expectation. As a result, the current practice is to perform a *restrictive* search for a small set of PTMs (such as phosphorylation) and ignore all other PTMs. The question arises whether one can design an *unrestrictive* PTM search algorithm that searches for *all* types of PTMs at once in a *blind* mode, i.e., without knowing which PTMs exist in a sample. Another, even more ambitious question is whether one can predict PTMs that are not known yet by data mining large MS/MS datasets, something that was never done before. The blind PTM identification

approach would open a possibility to study the extent and frequencies of different types of PTMs, still an open problem in proteomics.

The first blind approach to PTM identification (*spectral alignment*) was proposed by Pevzner et al., 2000, 2001, [17, 18]. Recently, Searle et al., 2004 (OpenSea) [21] and Han et al., 2004 [10] (SPIDER) proposed yet another approach to blind PTM identification. In contrast to spectral alignment, these approaches rely on de novo interpretation of MS/MS spectra. For example, Han et al. [10] formulate the problem as the identification of a modified peptide that best matches both the de novo interpretation and the database peptide. While this elegant formulation accommodates some de novo sequencing errors, the approach depends critically on a good de novo interpretation and thus is vulnerable to de novo sequencing errors.

In parallel, there has been extensive research of late on improving the scoring of unmodified mass spectra against peptides [2, 6, 9, 13, 16, 19, 20, 23, 25, 26]. These score functions consider fragmentation propensities, peptide composition, ion dependencies, completeness of b-y ladders, etc. Finally, the reliability of peptide assignments is considered by comparisons of the score distribution of correctly and incorrectly assigned peptides [11, 14, 15]. Unfortunately, these extensions do not carry over to the identification of modified peptides, and most algorithms for identifying PTMs still consider simpler score functions.

Recently, Tanner et al. [24] revisited PTM search in restricted mode (i.e, with a short list of putative modifications). They start with a de novo approach to tag generation, and combine tag generation and extension in the presence of modifications with a novel scoring function. This approach revealed many previously unidentified PTMs in large samples of MS/MS spectra (Alliance for Cellular Signalling) and is about 2 orders of magnitude faster than other restricted PTM searches. Tanner et al. [24] separate candidate generation from the scoring step thus allowing the scoring step to take advantage over recent improvements in scoring. However, their algorithm also has to "guess" the types of PTMs in advance and cannot be run in a blind mode. Moreover, the time complexity of their algorithm depends exponentially on the size of the set of allowed modifications. In particular, this makes the algorithm impractical for handling mutations.

In this paper, we revisit the blind approach to PTM identification that directly aligns the spectra against the database thus eliminating the dependence on accurate de novo interpretations. We describe a new algorithm that extends the dynamic programming approach of Pevzner et al. [17, 18] and addresses some limitations of spectral alignment. Our alignment is a local or *fitting* version of spectral alignment instead of global alignment as in Pevzner et al. [17, 18]. Additionally, we use a more realistic scoring function that accumulates evidence for present ions, penalizes missing ions, and uses a more general scoring to account for PTMs. Although these improvements further complicate the spectral alignment approach, we were successful in keeping the running time of the fitting spectral alignment low. We apply our algorithm only to get an initial list of candidates, which are then rigorously scored using a Support Vector Machine (SVM) approach to obtain p-values on the hits. Finally, we address the issue of robustness of identification, by combining information from *overlapping*, but identically modified peptides, and identical, but unmodified peptides to generate a list of reliable PTMs.

Identification of all types of PTMs present in a large collection of MS/MS is a difficult task. An even more difficult task (that still requires manual case-by-case analysis) is to distinguish between real PTMs and artifacts of protein identification and to estimate the relative frequencies of different types of PTMs. While many PTMs are known, there is no study of their relative frequencies in protein samples. In this paper we address both problems (finding many diverse PTMs and estimating their frequencies) by studying a large sample of MS/MS spectra generated at USC Medical School. The key bottleneck in studying such samples is the limited speed and accuracy of blind PTM identification. Using our approach we were able, for the first time, to construct a *PTM frequency matrix* PTM(Δ, a) that reflects the number of MS/MS spectra in a sample with predicted PTM Δ on amino acid a for all possible shifts Δ and all amino acids a (Table 2). Admittedly, many entries in this table represent interpretation artifacts rather than real PTMs. However, one can notice that while most entries in this table are small, some entries are very large (shown in bold). We argue that since the noise in the PTM frequency matrix is "random", the large values are likely to represent the real PTMs rather than artifacts. For example, the four largest entries in the PTM frequency matrix of Table 2 (PTM(16,M)=614, PTM(32,M)=376, PTM(28,K)=239, and PTM(14,C)=233) match common PTMs (oxidation and double oxidation of Methionine, dimethylation of Lysine, and carboxamidomethyl of Cysteine). These four PTMs alone increase the number of interpreted MS/MS spectra in the sample by $\approx 15\%$, a significant increase. Below we demonstrate an excellent correlation between high values in the PTM frequency matrix and known PTMs thus validating our approach. Moreover, some high values in the PTM frequency matrix (e.g., PTM(53,E)) may point to some still unknown modifications and provide multiple supporting evidence that they indeed may correspond to previously unknown PTMs rather than artifacts of our approach.

2 The algorithm

In this section, we formulate the *Modified Peptide Identification Problem* and describe two algorithms that solve it. We first describe a general algorithm that can solve the problem with any number of modifications. Then, we show how to solve the problem more efficiently when there are at most 2 modifications.

2.1 Preliminaries

We begin with some definitions. Let $A = \{a_1, \ldots, a_{20}\}$ be the set of amino acids, each with molecular mass $m(a_i)$. A *peptide* $P = p_1 \cdots p_n$ is a sequence of amino acids, with mass $m(P) = \sum_{i=1}^{n} m(p_i)$. For an experimental spectrum S, $m(S)$ is the mass of the spectrum, which is equal to the mass of the peptide that generated the spectrum.

Peptide fragmentation in a tandem mass spectrometer can be characterized by a set of numbers $\{s_1, \ldots, s_k\}$ representing the different types of ions that correspond to the removal of a certain chemical group from a peptide fragment (for example, $s = 17$ corresponds to loss of water). For tandem mass spectrometry, the *theoretical* spectrum $T(P)$ of a peptide P can be calculated by subtracting all possible ion types $s_1, \ldots, s_k$ from the masses of all prefixes and suffixes of P.

The Shared Peak Count between an experimentally measured spectrum S and a peptide P is the number of masses in S that are equal to masses in $T(P)$. In reality, peptide sequencing algorithms use more sophisticated scoring functions than a simple shared peaks count, incorporating different weighting functions for the matching masses and taking into account intensities of peaks. Let $\mathrm{Match}(S, P)$ be a function that scores the likelihood that a spectrum S is produced by a peptide P.

Computationally, a *modification* Δ of the peptide $P = p_1 \cdots p_n$ at position i results in a modified peptide $\hat{P}$ with the mass of residue p_i increased by Δ. We emphasize that this operation defines a theoretical spectrum of any modified peptide $\hat{P}$ and allows one to compute $\mathrm{Match}(S, \hat{P})$. We study the following problem:

Modified Protein Identification Problem
Input: A database of proteins, an experimental spectrum S, and a parameter k capping the number of modifications.
Output: A modified peptide $\hat{P}$ with the best match $\mathrm{Match}(S, \hat{P})$ to the spectrum S that is at most k modifications away from a peptide P that appears in the database (namely, P is a substring of some protein in the database).

To simplify the presentation, we make the following assumptions on the input. We assume that if two modifications appear on two consecutive amino acids of the peptide, then either the b-ion or the y-ion that corresponds to the cleavage site between these two amino acids appears in the spectrum S. Moreover, we assume that there are no two consecutive cleavage sites whose b and y ions are missing from S. We also assume that the masses of all amino acids and all modifications are integers, and that there are no measurement errors in the spectrum S. We note that our actual algorithm does not need these assumptions (due to lack of space we do not describe how to remove these assumptions).

2.2 Algorithm for arbitrary number of modifications

Any modified peptide that is at most k modifications away from a peptide in the database is called a *candidate*. Our approach is to find the highest scoring candidates according to some scoring function Match_1. These candidates are then rescored in a second stage using a more sophisticated scoring function Match_2 as described in Section 3. To start, low-intensity peaks are filtered from spectra as described in Bandeira et al. [4].

The score $\mathrm{Match}_1(S, \hat{P})$ of a candidate $\hat{P}$ consists of two parts: scores for the masses of the prefixes of $\hat{P}$, and scores for the modifications in $\hat{P}$. More precisely, let $\mathrm{MassScore}(v)$ be a scoring function for every mass v. Let $\mathrm{PTMScore}(\Delta, a) \leq 0$ be a penalty for having a modification Δ on the amino acid a. For every candidate $\hat{P}$, the score of $\hat{P}$ is the sum of $\mathrm{MassScore}(v)$ for every mass v of a prefix of $\hat{P}$ (including the entire peptide $\hat{P}$ and the empty prefix), plus the sum of scores of the modifications of $\hat{P}$ according to $\mathrm{PTMScore}$. In our implementation of the algorithm, we use the mass scoring function from Tanner et al. [24]. The function $\mathrm{PTMScore}$ is defined as follows:

$$\mathrm{PTMScore}(\Delta, a) = \begin{cases} C & \text{if } -M_1 \leq \Delta \leq M_2 \\ & \text{and } m(a) + \Delta \geq 50 \\ -\infty & \text{otherwise} \end{cases},$$

where M_1 (resp., M_2) is the maximum (resp., minimum) allowed mass offset of one modification, and $C < 0$ is some constant. This function forbids implausible interpretations, and gives better results than a constant penalty function.

We now show how to compute the score of the highest scoring candidate. From S we build a prefix residue mass (PRM) spectrum S', namely, for every mass $v \in S$, we add to S' the masses $v - 1$ and $m(S) - (v - 19)$, Furthermore, we add to S' the masses 0 and $m(S)$.

Denote the protein database q as a single sequence $p_1 \cdots p_n$, and let m be the size of the set S'. For every $j \leq n$ and $v \in S'$, let $D_k(j, v)$ be the maximum score of a peptide $\hat{P}$ with exactly k modifications whose unmodified peptide P is a substring of q that ends at p_j, and whose mass is v. Note that the size of the table D_k is $n \times m$.

The table D_0 can be easily computed in $O(nm)$ time. To compute a value $D_k(j, v)$ for $k \geq 1$ we need to consider five cases: (1) The optimal peptide $\hat{P}$ for $D_k(j, v)$ does not have a modification on its last amino acid, and the mass v' of the prefix of $\hat{P}$ of length $|\hat{P}| - 1$ is in S' (2) $\hat{P}$ does not have a modification on its last two amino acids and $v' \notin S'$, (3) $\hat{P}$ has a modification on its last amino acid and $v' \in S'$, (4) $\hat{P}$ has a modification on its last amino acid and $v' \notin S'$, and (5) $\hat{P}$ has a modification on its penultimate amino acid but not on its last, and $v' \notin S'$. Formally, the recurrence formula for $D_k(j, v)$ is given by the following lemma.

Lemma 2.1. $D_k(j, v) = \max\{d_{j,v,k,1}, d_{j,v,k,2}, d_{j,v,k,3}, d_{j,k,v,4}, d_{j,k,v,5}\} + \mathrm{MassScore}(v)$, *where* $d_{j,v,k,1}$, $d_{j,v,k,2}$, $d_{j,v,k,3}$, $d_{j,k,v,4}$, *and* $d_{j,k,v,5}$ *are defined in Figure 1.*

After computing $D_k(j, v)$ for all j and v, we can find the score of the highest scoring candidate with at most k modifications by computing $\max_{k' \leq k, j} D_{k'}(j, m(S))$. Each value $D_{k'}(j, m(S))$ is the maximum score of a candidate with k' modifications that ends at p_j. This candidate can be found by traversing the dynamic programming tables starting at $D_{k'}(j, m(S))$. By performing this process on the T k', j pairs with highest $D_{k'}(j, m(S))$ values (for some parameter T), we obtain a set of candidates, which is passed to the second stage.

The time complexity of the algorithm above is $O(knm^2)$. This is expensive for typical values of the parameters, and can be improved for two special cases of practical importance. The first case is when $\mathrm{PTMScore}(\Delta, a)$ is a constant C, namely, it does not dependent on the modification or the residue a. In that case, we can compute cases 3–5 of the algorithm in constant time by maintaining additional information. Define $M_k(j, v) = \max_{w<v}\{D_k(j, w)\}$. It is easy to see that in case 3, $d_{j,v,k,3} = M_{k-1}(j-1, w)$. Cases 4 and 5 can be modified in a similar fashion. M_k can be computed in constant time per entry, leading to an $O(knm)$ time algorithm.

In the next section, we describe an efficient algorithm for an arbitrary function PTMScore, but we limit the number of modifications to 2. As the results with 3 or more modifications are not very reliable, this restriction is reasonable. The handling of 0 or 1 modifications is simpler, and is not described here.

2.3 Algorithm for two modifications

We denote by M_1 (resp., M_2) the maximum (resp., minimum) mass offset of one modification. Instead of using the tables D_0, D_1, and D_2, our algorithm will use D_1 and the following two tables: For every $i \leq j$ such that $m(p_i \cdots p_j) \leq m(S) + M_1$, $\mathrm{PrefixScore}(i, j)$ is the score of $p_i \cdots p_j$, namely $\mathrm{PrefixScore}(i, j) = \mathrm{MassScore}(0) + \sum_{k=i}^{j} \mathrm{MassScore}(m(p_i \cdots p_k))$. Similarly, for every $j \leq i$ such that $m(p_j \cdots p_i) \leq m(S) - M_2$, $\mathrm{SuffixScore}(j, i) = \mathrm{MassScore}(m(S)) + \sum_{k=j}^{i} \mathrm{MassScore}(m(S) - m(p_k \cdots p_i))$. Computing the table PrefixScore is done by going over all i, and accumulating the scores $\mathrm{MassScore}(m(p_i \cdots p_j))$ for all j. Computing SuffixScore is done similarly.

Define $\Delta_{i,j,v} = v - m(p_i \cdots p_j)$ for every $i \leq j \leq n$ and $v \in S'$. To compute the table D_1, we use the following lemma (note that there are only four cases in the lemma, while Lemma 2.1 has five cases).

Lemma 2.2. $D_1(j, v) = \max\{d_{j,v,1,1}, d_{j,v,1,2}, d_{j,v,3}, d_{j,v,4}\} + \mathrm{MassScore}(v)$, *where* $d_{j,v,1,1}$ *and* $d_{j,v,1,2}$ *are defined in Figure 1, and* $d_{j,v,3}$ *and* $d_{j,v,4}$ *are defined in Figure 2.*

After computing $D_1(j, v)$ for all j and v, we can find the maximum score of a candidate as follows: Define $\hat{\Delta}_{j,i,v} = m(S) - m(p_{j+1} \cdots p_i) - v$ for all j, i, and v. The maximum score of a candidate is $\max(\cup_{j,v}\{b_{j,v,1}, b_{j,v,2}\})$, where $b_{j,v,1}$ and $b_{j,v,2}$ are defined in Figure 2. Like in the previous algorithm, we generate the highest scoring candidates by traversing the table D_1.

Time complexity The time complexity of the algorithm is $O(dnm)$ where $d = \lfloor (M_1 - M_2)/57 \rfloor + 1$ is an upper bound on the number of values of i we need to consider in order to compute a single value of $d_{j,v,3}$, $d_{j,v,4}$, $b_{j,v,1}$, or $b_{j,v,2}$ (this upper bound follows from the fact that for all i, $\Delta_{i+1,j,v} - \Delta_{i,j,v}$ is the mass of some amino acid, and the minimum mass of an amino acid is 57 Da.) On average, the number of i values is even smaller: $\bar{d} \simeq \lfloor (M_1 - M_2)/100 \rfloor + 1$. Typical values are $M_1 = 150, M_2 = -150$, implying $\bar{d} \simeq 4$.

3 Scoring and P-value computation

An important part of our approach is that we dissociate candidate generation from the final scoring, and P-value computation. To score modified peptides, we follow the approach of Tanner et al. [24] that is built upon Dancik et al., 1999 [7], and Bafna and Edwards, 2001 [2]. We also address the question of validity of the top-scoring peptide. While every spectrum returns a top-scoring peptide, the top scoring peptide is not always the correct one even if the score function is accurate. This could happen, for example, if the correct peptide was not in the database, or the spectrum was of low quality etc. The common approach to validation [1, 11, 14, 15, 24] is to combine a number of features, including the assignment score, a δ-score (difference in match score between the top match and the nearest runner-up), similar to Δ_{cn} feature used by Sequest. Because δ-score is sensitive to database size, we also include

$$d_{j,v,k,1} = \begin{cases} D_k(j-1, v - m(p_j)) & \text{if } v - m(p_j) \in S' \\ -\infty & \text{otherwise} \end{cases}$$

$$d_{j,v,k,2} = \begin{cases} D_k(j-2, v - m(p_{j-1}p_j)) + \text{MassScore}(v - m(p_j)) & \text{if } v - m(p_{j-1}p_j) \in S' \\ -\infty & \text{otherwise} \end{cases}$$

$$d_{j,v,k,3} = \max\left(\{D_{k-1}(j-1, w) + \text{PTMScore}(v - (w + m(p_j)), p_j) \middle| \forall w \in S', w < v\} \cup \{-\infty\}\right)$$

$$d_{j,v,k,4} = \max\left(\left\{ \begin{array}{l} D_{k-1}(j-2, w) + \text{PTMScore}(v - (w + m(p_{j-1}p_j)), p_j) \\ \quad + \text{MassScore}(w + m(p_{j-1})) \end{array} \middle| \forall w \in S', w < v \right\} \cup \{-\infty\}\right)$$

$$d_{j,v,k,5} = \max\left(\left\{ \begin{array}{l} D_{k-1}(j-2, w) + \text{PTMScore}(v - (w + m(p_{j-1}p_j)), p_{j-1}) \\ \quad + \text{MassScore}(v - m(p_j)) \end{array} \middle| \forall w \in S', w < v \right\} \cup \{-\infty\}\right)$$

Figure 1. Definitions of $d_{j,v,k,1}$, $d_{j,v,k,2}$, $d_{j,v,k,3}$, $d_{j,k,v,4}$, and $d_{j,k,v,5}$.

$$d_{j,v,3} = \max\left(\{\text{PrefixScore}(i, j-1) + \text{PTMScore}(\Delta_{i,j,v}, p_j) \middle| \forall i \text{ s.t. } M_1 \le \Delta_{i,j,v} \le M_2\} \cup \{-\infty\}\right)$$

$$d_{j,v,4} = \max\left(\left\{ \begin{array}{l} \text{PrefixScore}(i, j-2) + \text{PTMScore}(\Delta_{i,j,v}, p_{j-1}) \\ \quad + \text{MassScore}(v - m(p_j)) \end{array} \middle| \forall i \text{ s.t. } M_1 \le \Delta_{i,j,v} \le M_2 \right\} \cup \{-\infty\}\right)$$

$$b_{j,v,1} = \max\left(\left\{ \begin{array}{l} D_1(j, v) + \text{PTMScore}(\hat{\Delta}_{j,i,v}, p_{j+1}) \\ \quad + \text{SuffixScore}(j+2, i) \end{array} \middle| \forall i \text{ s.t. } M_1 \le \hat{\Delta}_{j,i,v} \le M_2 \right\} \cup \{-\infty\}\right)$$

$$b_{j,v,2} = \max\left(\left\{ \begin{array}{l} D_1(j, v) + \text{PTMScore}(\hat{\Delta}_{j,i,v}, p_{j+2}) \\ \quad + \text{SuffixScore}(j+3, i)\} + \text{MassScore}(v + m(p_{j+1})) \end{array} \middle| \forall i \text{ s.t. } M_1 \le \hat{\Delta}_{j,i,v} \le M_2 \right\} \cup \{-\infty\}\right)$$

Figure 2. Definitions of $d_{j,v,3}$, $d_{j,v,4}$, $b_{j,v,1}$, and $b_{j,v,2}$.

as a feature the total number of peptide candidates considered. Other features considered include the percentage of b and y fragments found, the percentage of spectral peaks annotated, and the percentage of total ions present in the annotated peaks. We use an SVM based approach, similar to Anderson et., 2003 [1]. We have chosen the ISB dataset (see below) as a training sample containing top-scoring correct, and incorrect peptide assignments, and an SVM was used to optimally classify the correct and incorrectly assigned peptides using the features described above. Due to space limitations, we omit the details. The distribution of the incorrect peptide assignments can be used as a P-value for the hit.

Different modifications may produce similar theoretical spectra, and a single spectrum often does not provide enough information to confidently choose among the alternatives. Consider a peptide with two consecutive Methionines, one of which is oxidized. Unless the spectrum is of particularly high quality, the two candidate peptides that place the oxidation on either residue will receive very similar scores. However, if these candidates greatly outscore any others, we can confidently assign a peptide annotation (with some uncertainty on the oxidation position). Therefore, we categorize search results as being either *incorrect*, *correct*, or *exact*. A *correct* result recovers the correct peptide (possibly with misplaced modifications), while an *exact* result recaptures the original peptide sequence exactly. In the case with two consecutive Methionines, and one oxidation, we would have high confidence that the match was correct, but lower confidence that it was exact.

Correspondingly, for the SVM training, we compute two δ-scores: The first is difference from the runner-up, and the second is difference in scores between the top scoring peptide and the highest scoring distinct peptide. The first δ-score is most useful for classifying matches as exact, and the second for classifying them as correct.

4 Results

Although there are multiple approaches for identifying blind PTMs [10, 17, 21], there was no direct comparison of these approaches yet. We also were not able to benchmark these tools against our approach for a variety of rea-

sons (PEDANTA was developed for in-house use within a company, OpenSea was not available for licensing at the time this paper was written and SPIDER's stand-alone version is not available). However, we make a few remarks that justify our approach. Our tool is an extension and improvement over the spectral alignment approach in PEDANTA. Both SPIDER and OpenSea require a good de novo interpretation as a starting point for the alignment, which is a challenging research problem. Further, both approaches use a scoring scheme that requires a manual validation of the results, thereby making it difficult to mine large datasets for interesting modifications. Our PTM frequency matrix approach provides reliable PTM identifications that bypasses this problem.

4.1 Datasets

We apply our analysis in three computational tests. In the first test, we choose previously identified spectra, but mutate the database. In the second, we shift the peaks of the spectra to simulate modifications. As large datasets of annotated spectra of modified peptides is currently missing, these datasets are valuable in testing the performance of our approach. Finally, we consider previously unannotated PTMs in a large dataset of 45079 spectra. The results on both simulated and real datasets validate our approach to identifying modifications and mutations.

We use the following MS/MS datasets in our computational experiments:

- **ISB dataset** Annotated high-quality spectra from the ISB dataset (charge 2, Sequest Xcorr score $>$ 2), a public collection of MS/MS spectra from 22 separate LC-MS runs on a ThermoFinnigan ESI-ITMS [12].

- **IKKb dataset** 45079 spectra acquired at USC Medical School from a non-specific digestion of the inhibitor of nuclear factor kappa B kinase beta subunit (shortly IKKb protein). GST-IKKb was produced in E. coli and purified on glutathione sepharose (T. Higashimoto and E. Zandi, unpublished data). IKKb was digested using multiple proteases (Trypsin, V6 Protease, Elastase) to produce overlapping peptides.

The ISB datasets were used in the following simulations:

SimMod We constructed a dataset of modified spectra by adding feasible modifications to each peptide as described in Tanner et al., 2005 [24]. SimMod$_i$ refers to the dataset with i modifications randomly selected from the set of feasible modifications. The set of allowable modifications was hydroxylation of Proline or Lysine, sulfation of Tyrosine, and oxidation of Methionine.

MutDb We selected 168 spectra from the ISB dataset, constructed a non-redundant database of the protein sequences containing the relevant peptides, and mutated this database to a sequence identity level of 90%. Mutations are chosen at random, using parameters from the BLOSUM90 matrices.

4.2 Results on SimMod and MutDb

We searched the MutDb spectra in blind mode, allowing up to two modifications per peptide. As shown in Table 1(a), accuracy is affected by the number of mutations. As the number of modifications increase further, the advantages of database search over *de novo* sequencing are attenuated. Results on the SimMod$_1$ and SimMod$_2$ datasets were quite similar to those obtained for the MutDb spectra — see Table1(b). This is not surprising, as searching for mutations is similar to the search for post-translational modifications, when an unrestricted search is made.

4.3 Results on IKKb

We analyzed all 45079 spectra in the IKKb dataset using our algorithm in blind mode allowing up to 2 modifications. In addition to 9477 unmodified peptides, we identified 5246 spectra with a single PTM and 354 spectra with with two PTMs. We admit that some of our PTM assignments may be wrong and below we describe a procedure to distinguish between reliable and questionable PTM identifications.

For every peptide with found modification of Δ on amino acid a, we incremented the count PTM(Δ, a) in the PTM frequency matrix (Table 2). Large entries in the PTM frequency matrix indeed correspond to known and common modifications (shown in gray) thus validating our approach.

We note that the modifications with mass difference 1 should be taken with caution since they may represent an artifact caused by errors in parent mass. However, one of the entries PTM(1,N)=180 is particularly large and we suggest that it is a reflection either of a well-known modification (deamidation of Asparagine) or a mutation rather than a parent mass artifact. We also remark that some large counts in the PTM frequency matrix (e.g., PTM(16,M)=614) may have a "shadow" (like PTM(17,M)=83) that most likely represent a parent mass artifact rather than a separate PTM.

There are many small values in the PTM frequency matrix that likely are noise rather than real PTMs. We argue that there exists an evidence for a PTM Δ at an amino acid in the database if at least two MS/MS spectra support this PTM at this position (these two spectra may be interpreted by either the same peptides or by overlapping peptides). To further increase the signal to noise ratio, we label a predicted modified peptide as *reliable* if there exists another predicted modified peptide with exactly the same modifi-

(a)

	0 mutations	1 mutation	2 mutations	3+ mutations	Overall
Exact	97.1%	54.8%	8.1%	0.0%	42.5%
Correct:	2.9%	35.5%	70.3%	45.5%	38.3%
Incorrect:	0.00%	9.7%	21.6%	54.6%	19.2%

(b)

	1 PTM	2 PTMs
Exact	57.3%	15.6%
Correct:	35.4%	67.2%
Incorrect:	7.3%	17.2%

Table 1. (a) Search results against MutDB. Accuracy and exactness are reported for spectra whose corresponding peptides contain zero, one, two, or more than two mutations. A search allowing up to two unrestricted modifications was run, so exact matches for a peptide with three or more mutations were not found. (b) Search results against SimMod$_1$ and SimMod$_2$.

offset	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y	total
-32	0	6*	1	0*	0*	0	0	0	0	0	**133***	0	0	0	0	0	0	0	0	0*	140
-18	0	0	**49***	15	7*	0	1	7	5	0	0*	1	6	6	5	21	**58**	1	0	2	184
1	13	1	24	25	23	**35**	21	**36***	3*	**72***	3	**180***	**37**	**30***	3	13	**31**	21	10	23	604
14	5	**233**	2*	4	0	4*	0	1	**84**	5	**37**	1*	1	0	1	3*	0*	17*	0	1	399
16	2*	2	3*	**56**	5*	15	0	**34***	0	**35***	**614***	7	6*	9	0	8*	3	4*	**83**	4	890
17	1	0	1	2	1	5	0	6	0	11	**83**	6*	0*	4	0	7	1	0	17	1	146
22	10	1	16*	23	8	28	4	16	1	16	1	22	7	22	0	23	20	16	0	10	244
28	4*	2*	0	1	0	0	0	4	**239***	29	0	0	1	3*	0	**39***	2*	**52**	0	1	377
32	0*	0	0*	0	0	3	0	0	0	3	**376***	2	1*	1	0	1	0	2*	20	0	409
53	0	0*	21	**39**	0	4	0	15	1	14	1	5	3	2	0	3	3	7	0	2	120

Table 2. PTM frequency matrix for the IKKb dataset (45079 spectra). The first column describes the mass shift. The entry for modification Δ and amino acid a refers to the number of times a appeared with a modification of mass Δ in a top-scoring spectral interpretation. Only rows with total count of at least 100 are shown. Entries with frequency greater than 30 are shown in bold and entries that can be explained by mutations are starred. Entries corresponding to known types of modifications are shown in gray.

cation on the same position in the database (the first and second peptides are either identical or overlap). We further filter the database of modified peptide by retaining only the reliable peptides and recompute the PTM frequency matrix for the filtered dataset. This procedure greatly increases the signal to noise ratio and reduces most entries in the PTM frequency matrix to zeros (data are not shown).

One can argue that two overlapping peptides validating the same PTM is a more reliable evidence of PTM than two identical peptides validating the same PTM. To find the most reliable PTMs we further reduce noise in the PTM frequency matrix by applying the second filtering step and retaining only modified peptides that have an overlapping modified peptide with the same modification on the same position. The resulting matrix is shown in Table 3. If one assumes that the noise cause by incorrect assignments is distributed randomly across the PTM frequency matrix then the probability that the bold entries appear simply by chance is negligent.

The entries that do not correspond to popular PTMs (e.g., PTM(53,E)), but still appear in significant numbers, may prove to be interesting since they may correspond to still unknown PTMs. Table 3 provides additional evidence that some of these putative PTMs are not artifacts. Another approach to validation is if the same peptide appears multiple number of times, with and without the modification. We list some of the modified peptides with multiple support in Table 4. Remarkably, most of the the modifications appear in both modified and unmodified forms or on overlapping peptides. Although many modifications in Table 4 are common chemical modifications (like oxidation of Methionine), there are some less common ones, such Tryptophan oxidation, and double oxidation, addition of Sodium (22), dimethylation of K (28), and yet unconfirmed PTM of mass 53 on E. Moreover, many of these peptides either have multiple spectra (of the same peptide) confirming the found PTMs, or identical modifications were predicted on overlapping peptides, but distinct peptides. Such independent cross-validations greatly increase our confidence in the assignments. In most cases, the modified peptide appeared a number of times in its unmodified form. However, the modified peptide was the dominant form for Cys-methylation, and Lys-dimethylation, implying that these were constitutive modifications. We remark that these peptides are missed by traditional database search.

Some of the entries in Table 2 may correspond to PTMs

offset	A	C	D	E	F	G	H	I	K	L	M	N	P	Q	R	S	T	V	W	Y	total
-32	0	0*	0	0*	0*	0	0	0	0	0	0*	0	0	0	0	0	0	0	0	0*	0
-18	0	0	5*	4	0*	0	0	0	0	0	0*	0	0	0	0	2	**47**	0	0	0	58
1	0	0	0	16	14	16	3	17*	0*	27*	0	**139***	6	10*	0	2	8	5	0	2	265
14	4	**216**	0*	0	0	0*	0	0	**84**	0	0	0*	0	0	0	0*	0*	0*	0	0	304
16	0*	0	2*	**50**	0*	10	0	**34***	0	**34***	**474***	3	0*	8	0	6*	0	4*	**65**	0	690
17	0	0	0	0	0	3	0	4	0	8	**70**	0*	0*	0	0	0	0	0	3	0	88
22	0	0	0*	11	0	14	0	0	0	9	0	3	3	2	0	0	2	2	0	10	56
28	2*	0*	0	0	0	0	0	2	**239***	29	0	0	0	0*	0	**38***	0*	0	0	0	310
32	0*	0	0*	0	0	0	0	0	0	0	**190***	0	0*	0	0	0	0	0*	0	0	190
53	0	0*	18	19	0	0	0	6	0	5	0	3	0	0	0	0	0	0	0	0	51

Table 3. Filtered PTM frequency matrix for the IKKb dataset.

while other may correspond to mutated proteins (starred entries). Can we distinguish between different modifications/mutations that result in the same mass difference? Currently, we treat these simply as mass differences, and do not try to distinguish between the two. Indeed, we identify few false PTMs in the MutDb search, and correspondingly few mutations in the SimMod search. However, this issue can be addressed. If we see peptides with and without the mass difference, then modification is the more likely scenario. Alternatively, if we see peptides with a shift but do not see peptides without a shift (like 28 on K), then mutation may be a more likely scenario.

5 Conclusions

We expect that even larger datasets will provide further validation of our PTM frequency matrix approach and will generate independent pieces of evidence to support our conclusions (work in progress). As the tools mature, and more modified and mutated peptides are identified, we can begin to differentiate between different types of modifications, by mining these datasets. As an example, Tanner et al. [24] successfully promote the use of phosphate-loss ions as signatures of phosphorylation. Other modifications undoubtedly will lead to differences in fragment ion propensities which can be used to train appropriate score functions. Mutations present a more challenging scenario. However, certain mutations (for example, a mutation to Proline) can cause significant change in ionization propensities, which could be mined to improve identifications.

5.1 Acknowledgments

This project was supported by NIH grant NIGMS 1-R01-RR16522.

References

[1] D. C. Anderson, W. Li, D. Payan, and W. Noble. A new algorithm for the evaluation of shotgun peptide sequencing in proteomics: Support vector machine classification of peptide MS/MS spectra and SEQUEST scores. *Journal of Proteome Research*, 2(2):137–46, 2003.

[2] V. Bafna and N. Edwards. SCOPE: a probabilistic model for scoring tandem mass spectra against a peptide database. *Bioinformatics*, 17 Suppl 1:13–21, 2001.

[3] B. Ballif, J. Villen, S. Beausoleil, D. Schwartz, and G. S.P. Phosphoproteomic analysis of the developing mouse brain. *Mol Cell Proteomics*, 3(11):1093–101, 2004.

[4] N. Bandeira, H. Tang, V. Bafna, and P. Pevzner. Shotgun protein sequencing by tandem mass spectra assembly. *Analytical Chemistry*, (in press), 2004.

[5] G. Cantin and J. Yates. Strategies for shotgun identification of post-translational modifications by mass spectrometry. *Journal of Chromatography A*, 1053:7–14, 2004.

[6] D. Creasy and J. Cottrell. Error tolerant searching of uninterpreted tandem mass spectrometry data. *Proteomics*, 2(10):1426–1434, Oct 2002.

[7] V. Dancík, T. Addona, K. Clauser, J. Vath, and P. Pevzner. De novo peptide sequencing via tandem mass spectrometry. *J Comput Biol*, 6(3-4):327–342, Fall-Winter 1999.

[8] J. Eng, A. McCormack, and J. Yates. An Approach to Correlate Tandem Mass-Spectral Data of Peptides with Amino Acid Sequences in a Protein Database. *Journal Of The American Society For Mass Spectrometry*, 5(11):976–989, Nov 1994.

[9] A. Frank, T. Tanner, and P. Pevzner. Peptide sequence tags for fast database search in mass-spectrometry. *to appear in proceedings of Ninth Annual International Conference on Research in Computational Molecular Biology (RECOMB)*, 2005.

[10] Y. Han, B. Ma, and K. Zhang. SPIDER: software for protein identification from sequence tags with *de novo* sequencing error. In *IEEE Computational Systems Bioinformatics Conference (CSB)*, pages 206–215, 2004.

[11] A. Keller, A. Nesvizhskii, E. Kolker, and R. Aebersold. Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search. *Anal Chem*, 74(20):5383–5392, Oct 2002.

Peptide	count	
-18 on T (Water loss)		
LGT*GGFGNVIR	5	179
T*GGFGNVIR	35	0
1 on N (Deamidation)		
LTHPN*VV	2	111
LTHPN*VVAAR	1	32
LGEHN*IDVL	2	71
LGEHN*IDVLE	1	137
K[28]IITHPNFN*GN	3	0
K[28]IITHPNFN*GNTL	3	0
IITHPNFN*GN	10	40
IITHPNFN*GNTL	15	11
IITHPNFN*GNTLDND	2	0
IITHPNFN*GNTLDNDI[16]M	2	0
IITHPNFN*GNTLDNDIM[16]LI	5	0
IITHPNFN*GNTLDNDIML[16]I	1	0
THPNFN*GNTL	7	15
THPNFN*GNTLDNDI[16]M	1	0
THPNFN*GNTLDNDIM[16]LI	3	3
FN*GNTLDNDIM[16]LIK	1	0
AFPSAINQDNYPN*GGFTAE	20	4
AFPSAINQDNYPN*GGFTAEQITK	1	0
TYLN*GDHVTHPDF[16]M	6	0
TYLN*GDHVTHPDFM[16]L	8	0
14 on C (PAM-Cys)		
AGLALIPDKPATQC*ISDGK	11	0
AGLALIPDKPATQC*ISDGKLNE	1	0
LIPDKPATQC*ISDGK	2	0
IPDKPATQC*ISDGK	22	0
PDKPATQC*ISDGK	1	0
C*ISDGKLNEGH	8	0
PQPESVSC*ILQEPK	3	4
SVSC*ILQEPK	66	0
SVSC*ILQEPKR	1	0
SC*ILQEPK	46	38
C*ILQEPK	4	0
AHNLC*TLLE	47	0
HNLC*TL	1	0
14 on K (Methylation)		
LGEHNIDVLEGNEQFINAAK*	2	4
NIDVLEGNEQFINAAK*	10	3
EGNEQFINAAK*	7	4
GNEQFINAAK*	2	3
K*LSSPATL	16	0
K*LSSPATLN	2	0
K*LSSPATLNS	37	0
K*LSSPATLNSR	8	0

Peptide	count	
16 on M (Oxidation)		
RDVPEGM*QNLAPNDLPL	13	0
RDVPEGM*QNLAPNDLPLLA	14	0
DVPEGM*QNLAPN	4	0
DVPEGM*QNLAPNDLPL	3	0
DVPEGM*QNLAPNDLPLLA	10	0
LVHILNM*VTGT	11	0
LVHILNM*VTGTI	1	0
L[42]VHILNM*VTGTIH	4	0
LVHILNM*VTGTIH	13	0
LVHILNM*VTGTIHTYPVTE	2	2
LNM*VTGTIH	13	0
NM*VTGTIH	1	3
LNEGHTLDM*DLV	1	0
LNEGHTLDM*DLVFL	2	0
GHTLDM*DLVFLFDNSK	2	0
TLDM*DLVFL	1	0
M*DLVFLFDNSK	5	0
VEEVVSLM*NEDEK	6	10
EVVSLM*NEDEKTVVR	1	0
VVSLM*NEDEK	2	3
SLM*NEDEK	1	0
SLM*NEDEKTVV	54	4
M*NEDEKTVV	7	0
VRGPVSGSPDSM*	2	1
VRGPVSGSPDSM*NA	10	0
VRGPVSGSPDSM*NAS	6	5
GPVSGSPDSM*NASR	4	11
SRLSQPGQLM*SQPS	1	0
SRLSQPGQLM*SQPSTA	7	0
LSQPGQLM*SQPSTA	13	4
LSQPGQLM*SQPSTASNSLPEPAK	3	6
PGQLM*SQPSTASNSLPEPAKK	1	0
K[28]IITHPNFNGNTLDNDIM*LI	4	0
IITHPNFNGNTLDNDIM*	8	0
IITHPNFNGNTLDNDIM*LI	30	0
IITHPNFN[1]GNTLDNDIM*LI	5	0
IITHPNFNGNTLDNDIM*LIK	11	0
IITH[34]PNFNGNTLDNDIM*LIK	1	0
THPNFNGNTLDNDIM*	2	0
THPNFNGNTLDNDIM*L	2	0
THPNFNGNTLDNDIM*LI	5	3
THPNFN[1]GNTLDNDIM*LI	3	3
FN[1]GNTLDNDIM*LIK	1	0
GNTLDNDIM*LI	2	0
TLDNDIM*LI	5	0
TLDNDIM*LIK	11	0
DFLSKLPEM*	9	0
DFLSKLPEM*L	43	35
LPEM*LK	3	0
16 on W (Hydroxylation)		
GFRPFLPNW*Q	1	0
GFRPFLPNW*QPV	55	150
RPFLPNW*QPV	1	75
QKELW*NLL	3	81
QKELW*NLLK	1	16
ELW*NLLK	1	27

Peptide	count	
22 on E (Sodium)		
IQQDTGIPEEDQE*LL	1	15
IQQDTGIPEEDQE*LLQ	1	2
KQGGTLDDLEE*Q	1	52
KQGGTLDDLEE*QA	3	72
NIDVLEGNE*Q	1	6
NIDVLEGNE*QFI	3	17
NIDVLEGNE*QFINAA	1	15
28 on K (Dimethylation)		
NIDVLEGNEQFINAAK*II	2	0
K*IITHPNFNGN	10	0
K*IITHPNFN[1]GN	3	0
K*IITHPNFNGNTL	4	0
K*IITHPNFN[1]GNTL	3	0
K*IITHPNFNGNTLDNDIM[17]L	1	0
K*IITHPNFNGNTLDNDIM[16]LI	4	0
IMLIK*LSSPATLNSR	5	0
IK*LSSPATL	1	0
IK*LSSPATLNSR	1	0
K*LSSPATL	39	0
K*LSSPATLN	20	0
K*LSSPA[1]TLNS	1	0
K*LSSPATL[1]NS	3	0
K*LSSPATLNS	97	0
K*LSSPATLN[1]S	6	0
K*LSSPAT[1]LNS	2	0
K*LSSPATLNSR	13	0
K*LSSP[1]ATLNSR	1	0
K*LSSPATLN[1]SR	5	0
28 on S (???)		
DIFGPGTS*ILSTWIGGSTR	21	0
DIFGPGTS*ILSTWIGGSTRSISGT	4	0
FGPGTS*ILSTWIGGSTR	1	0
GPGTS*ILSTWIGGSTR	4	0
32 on M (Oxidation)		
M*MALQTD[53]IVDLQ	1	0
M*MALQTDIVDLQ	10	0
M*MALQTDIVD[22]LQ	3	0
M*MALQTDI[53]VDLQ	1	0
M*MALQTD[1]IVDLQR	2	4
M*MALQTDIVDLQR	160	4
53 on D (???)		
KQGGTLDD*LEEQA	7	72
KQGGTLDD*LEEQAR	1	21
QGGTLDD*LEEQA	3	7
53 on E (???)		
DLKPE*NIV	1	51
DLKPE*NIVLQ	3	41
IQQDTGIPE*EDQE	1	0
IQQDTGIPE*EDQELL	7	15
53 on I (???)		
DLKPENI*VLQ	1	41
DLKPENI*VLQQ	5	32
DLKPENI*VLQQGEQ	1	28
ALDDI*LNL	6	8
ALDDI*LNLK	3	108

Table 4. PTM validation, described by multiple occurrences of identical or overlapping peptides. The second column is the number of occurrences of the modified peptide, while the third column is the count of the unmodified peptides. It is possible that some of the PTMs listed in the table are incorrect either in the peptide assignment, or in the assignment of PTM positions.

[12] A. Keller, S. Purvine, A. Nesvizhskii, S. Stolyar, D. R. Goodlett, and E. Kolker. Experimental Protein Mixture for Validating Tandem Mass Spectral Analysis. *OMICS*, 6(2):207–212, 2002.

[13] B. Lu and T. Chen. A suffix tree approach to the interpretation of tandem mass spectra: applications to peptides of non-specific digestion and post-translational modifications. *Bioinformatics*, 19 Suppl 2:113–113, Oct 2003.

[14] M. MacCoss, C. Wu, and J. Yates. Probability-based validation of protein identifications using a modified SEQUEST algorithm. *Anal Chem*, 74(21):5593–5599, Nov 2002.

[15] A. Nesvizhskii, A. Keller, E. Kolker, and R. Aebersold. A statistical model for identifying proteins by tandem mass spectrometry. *Anal Chem*, 75(17):4646–4658, Sep 2003.

[16] D. Perkins, D. Pappin, D. Creasy, and J. Cottrell. Probability-based protein identification by searching sequence databases using mass spectrometry data. *Electrophoresis*, 20(18):3551–3567, Dec 1999.

[17] P. Pevzner, V. Dancík, and C. Tang. Mutation-tolerant protein identification by mass spectrometry. *J Comput Biol*, 7(6):777–787, 2000.

[18] P. Pevzner, Z. Mulyukov, V. Dancik, and C. Tang. Efficiency of database search for identification of mutated and modified proteins via mass spectrometry. *Genome Res*, 11(2):290–299, Feb 2001.

[19] J. Razumovskaya, V. Olman, D. Xu, E. Uberbacher, N. VerBerkmoes, R. Hettich, and Y. Xu. A computational method for assessing peptide-identification reliability in tandem mass spectrometry analysis with sequest. *Proteomics*, 4:961–969, 2004.

[20] R. Sadygov and J. Yates. A hypergeometric probability model for protein identification and validation using tandem mass spectral data and protein sequence databases. *Anal Chem*, 75(15):3792–3798, Aug 2003.

[21] B. Searle, S. Dasari, M. Turner, A. Reddy, D. Choi, P. Wilmarth, A. McCormack, L. David, and S. Nagalla. High-throughput identification of proteins and unanticipated sequence modifications using a mass-based alignment algorithm for MS/MS de novo sequencing results. *Anal Chem*, 76(8):2220–2230, Apr 2004.

[22] H. Shu, S. Chen, Q. Bi, M. Mumby, and D. Brekken. Identification of phosphoproteins and their phosphorylation sites in the wehi-231 b lymphoma cell line. *Molecular and Cellular Proteomics*, 3:279–286, 2004.

[23] D. Tabb, L. Smith, L. Breci, V. Wysocki, D. Lin, and J. Yates. Statistical characterization of ion trap tandem mass spectra from doubly charged tryptic peptides. *Anal Chem*, 75(5):1155–1163, Mar 2003.

[24] S. Tanner, H. Shu, A. Frank, M. Mumby, P. Pevzner, and V. Bafna. InsPecT: fast and accurate identification of post-translationally modified peptides from tandem mass spectra. submitted.

[25] J. Yates, J. Eng, and A. McCormack. Mining genomes: correlating tandem mass spectra of modified and unmodified peptides to sequences in nucleotide databases. *Anal Chem*, 67(18):3202–3210, Sep 1995.

[26] J. Yates, J. Eng, A. McCormack, and D. Schieltz. Method to correlate tandem mass spectra of modified peptides to amino acid sequences in the protein database. *Anal Chem*, 67(8):1426–1436, Apr 1995.