Anal. Chem. **2005,** *77,* 964−973

PepNovo: De Novo Peptide Sequencing via Probabilistic Network Modeling

Ari Frank* and Pavel Pevzner

Department of Computer Science & Engineering, University of California, San Diego, La Jolla, California 92093-0114

We present a novel scoring method for de novo interpretation of peptides from tandem mass spectrometry data. Our scoring method uses a probabilistic network whose structure reflects the chemical and physical rules that govern the peptide fragmentation. We use a likelihood ratio hypothesis test to determine whether the peaks observed in the mass spectrum are more likely to have been produced under our fragmentation model than under a model that treats peaks as random events. We tested our de novo algorithm PepNovo on ion trap data and achieved results that are superior to popular de novo peptide sequencing algorithms. PepNovo can be accessed via the URL http://www-cse.ucsd.edu/groups/bioinformatics/software.html.

In recent years, tandem mass spectrometry has become a leading technology responsible for many of the advances in the field of proteomics.[1,2] Samples for mass spectrometry experiments can represent mixtures of thousands of proteins, some in very low quantities. These mixtures are treated with proteolytic enzymes to break the proteins down into short peptides. Each sample can generate thousands of spectra, with each spectrum ideally being created by a peptide from one of the proteins in the organism's proteome. Performing the task of matching spectra to peptides manually is very labor intensive, and much research has been done to automate this process.

The most popular approach to peptide identification was pioneered by Yates in the early 1990s. In this approach, the mass spectrum is scored against a database of all candidate peptides to detect significant matches. Popular algorithms such as Sequest[3] and Mascot[4] use this approach. Though they offer an automated high-throughput method for peptide identification, the current database search techniques do not give a complete solution to this problem. Database algorithms implicitly assume that the genome is accurately sequenced, and *all* protein coding genes are annotated. The latter condition is hardly ever met due to many alternatively spliced genes, most of which are not adequately represented in the existing databases.[5] In addition, database search algorithms may miss the identification of some peptide−spectrum matches due to limitations of the relatively simple scoring methods they use. In other instances, matches are missed due to mutations/polymorphysims or the presence of modified amino acids in the peptide that are not considered by the database search algorithm (the consideration of many modified amino acids usually renders database searches prohibitive as far as running time is concerned).

For the reasons mentioned above, much effort has been invested into development of another type of algorithms for identifying peptides, de novo sequencing methods.[6−14] With de novo sequencing, a reconstruction of the original peptide sequence is done without knowledge of the genomic sequences or even the organism from which the sample was taken. In addition, the introduction of modified amino acids into the reconstruction is usually less prohibitive than it is in database searches. Among the de novo algorithms being used today are Lutefisk[8,11] (a publicly available de novo tool), Sherenga[9] (part of the Spectrum Mill software suite by Agilent), and Peaks[14] (available from Bioinformatics Solutions, Inc.). There is also a variation of the de novo sequencing, the tag-based methods.[15−18] In this hybrid approach, putative very short peptide sequences (sequence tags) are recovered from the spectra using de novo methods and are used to filter the candidate peptides from a database. A review of several of the common de novo algorithms is given in ref 19.

The goal of a de novo peptide sequencing algorithm is to sequence the peptide whose fragmentation created the experimental spectrum. A key component in de novo peptide sequencing

* To whom correspondence should be addressed. E-mail: arf@cs.ucsd.edu.

(1) Washburn, M. P.; Wolters, D.; Yates, J. R., III. *Nat. Biotechnol.* **2001**, *19*, 242−17.
(2) Patterson, S. D.; Aebersold, R. H. *Nat. Genet.* **2003**, *33*, 311−323.
(3) Eng, J. K.; McCormack, A. L.; Yates, J. R., III. *J. Am. Soc. Mass Spectrom.* **1994**, *5*, 976−989.
(4) Perkins, D.; Pappin, D.; Creasy, D.; Cottrell, J. *Electrophoresis* **1997**, *20*, 3551−3567.
(5) Leipzig, J.; Pevzner, P.; Heber, S. *Nucleic Acids Res.* **2004**, *32*, 3977−83.
(6) Bartels, C. *Biomed. Environ. Mass Spectrom.* **1990**, *19*, 363−368.
(7) Fernandez-de-Cossio, J.; Gonzalez, J.; Besada, V. *Comput. Appl. Biosci.* **1995**, *11*, 427−434.
(8) Taylor, J. A.; Johnson, R. S. *Rapid Commun. Mass Spectrom.* **1997**, *11*, 1067−1075.
(9) Dancik, V.; Addona, T. A.; Clauser, K. R.; Vath, J. E.; Pevzner, P. A. *J. Comput. Biol.* **1999**, *6*, 327−42.
(10) Chen, T.; Kao, M. Y.; Tepel, M.; Rush, J.; Church, G. M. *J. Comput. Biol.* **2001**, *8*, 325−337.
(11) Taylor, J. A.; Johnson, R. S. *Anal. Chem.* **2001**, *73*, 2594−2604.
(12) Lubeck, O.; Sewell, C.; Gu, S.; Chen, X.; Cai, D. M. *Proc. IEEE* **2002**, *90*, 1868−1874.
(13) Bafna, V.; Edwards, N. *Proceedings of the seventh annual international conference on Computational molecular biology*; 2003; pp 9−18.
(14) Ma, B.; Zhang, K.; Lajoie, G.; Doherty-Kirby, A.; Hendrie, C.; Liang, C.; Li, M. *Rapid Commun. Mass Spectrom.* **2003**, *17*, 2337−2342.
(15) Mann, M.; Wilm, M. *Anal. Chem.* **1994**, *66*, 4390−4399.
(16) Sunyaev, S.; Liska, A. J.; Golod, A.; Shevchenko, A.; Shevchenko, A. *Anal. Chem.* **2003**, *75*, 1307−1315.
(17) Tabb, D. L.; Saraf, A.; Yates, J. R., III. *Anal. Chem.* **2003**, *75*, 6415−21.
(18) Day, R.; Borziak, A.; Gorin, A. *Proceedings of IEEE Computational Systems in Bioinformatics (CSB);* 2004; pp 505−508.
(19) Lu, B.; Chen, T. *Drug Discovery Today: BioSilico* **2004**, *2*, 85−90.

10.1021/ac048788h CCC: $30.25
Published on Web 01/13/2005

Table 1. Fragment Ions[a]

prefix fragments			suffix fragments		
fragment type	offset[b]	probability[c]	fragment type	offset[b]	probability[c]
b	$M + 1$	0.83 (0.66)	y	$M + 19$	0.87 (0.71)
b − H_2O	$M - 17$	0.39 (0.30)	y − H_2O	$M + 1$	0.26 (0.21)
b − NH_3	$M - 16$	0.36 (0.28)	y − NH_3	$M + 2$	0.24 (0.19)
b − H_2O − H_2O	$M - 35$	0.13 (0.10)	y − H_2O − H_2O	$M - 17$	0.11 (0.09)
b − H_2O − NH_3	$M - 34$	0.12 (0.09)	y − H_2O − NH_3	$M - 16$	0.13 (0.10)
b^{2+}	$(M + 2)/2$	0.13 (0.08)	y^{2+}	$(M + 20)/2$	0.23 (0.18)
a (b − CO)	$M - 27$	0.34 (0.26)			
a − H_2O	$M - 45$	0.17 (0.13)			
a − NH_3	$M - 44$	0.20 (0.15)			

[a] The value of M used in the table depends on the type of fragment examined. When examining a prefix fragment, M is defined as the mass $M = \sum_{j=1}^{i} m(p_j)$, and when a suffix fragment is examined, M is defined as the mass $M = \sum_{j=i+1}^{n} m(p_j)$. [b] The offsets are rounded to the nearest integer value. [c] This is the probability of observing expected fragments that have a mass in the "visible" portions of the spectra (for each spectrum, this is the range of masses between the peak with the lowest mass and the peak with the highest mass). The probability for all expected fragment ions (for the whole range of masses) appears in parentheses.

algorithms (as well as database search algorithms) is a scoring function, which is used to evaluate the matches between candidate peptides and the given experimental spectrum. Some scoring methods use the correlation between the observed spectrum and a theoretical spectrum generated from candidate peptides.[3,20] A second popular approach to scoring is based on assigning probabilities to the observed fragment peaks. Several different scoring schemes have been proposed for this purpose.[9,21−23] However, most of them are designed specifically for database search, and their incorporation into de novo algorithms has not been widely investigated. One noted exception is the scoring method due to Dancik et al.,[9] which is specifically designed for de novo sequencing of peptides and is implemented in the Sherenga algorithm. This scoring method employs a likelihood test that compares between two explanations for the observed peaks. The first is a statistical model that assumes the peaks are a result of a peptide's fragmentation, and the second is a model that presumes that the peaks were created by a random process (not governed by fragmentation rules). The Dancik et al. scoring was further improved by Havilio et al.,[22] who introduced a general framework for designing scoring functions that can incorporate many experimental observations and prior mass spectrometry knowledge. Our approach has some similarities to their method, mainly the possibility to introduce dependencies between different observations. However, their score is presented solely in the context of scoring candidate peptides in database searches and is applicable to a limited set of correlations between fragments. In our work, we describe several extensions that significantly improve the performance of the Dancik et al. scoring function. Other examples of scoring schemes intended for database searches include the Scope scoring by Bafna and Edwards,[21] which is a probabilistic model for scoring spectra against a peptide database, which takes into account factors such as fragment ion probabilities, noisy spectra, and instrument errors. Elias et al.[23] employed an intensity-based scoring model that uses decision trees to determine the probability of observed fragment intensities. Colinge et al.[24] introduced the OLAV family of scoring schemes, which use probabilistic models and hidden Markov models to assess the quality of a database match.

This paper is structured as follows. We first introduce the terminology, along with some basic mass spectrometry concepts. We then describe our novel scoring method and its application to ion trap MS/MS data. Following that we present our experimental results.

BACKGROUND AND TERMINOLOGY

In this section, we describe some of the basic concepts used in this paper.

Fragment Ions. A peptide P is a sequence of n amino acids, $P = p_1p_2p_3 \ldots p_n$, in an alphabet of 20 amino acids, each amino acid having a mass $m(p_i)$. The parent mass of peptide P, is defined as $PM(P) = \sum_{i=1}^{n} m(p_i)$ + mass of H_2O. Generally, in mass spectrometry experiments, peptides break along their backbones between successive amino acids during the stage of collision-induced dissociation (CID). This results in $n + 1$ possible cleavage sites in a peptide (this count includes the empty peptide with mass 0, and the full peptide with mass PM).

A common event in the CID stage is a single cleavage along the peptide's backbone. Such a breakage results in a prefix fragment $p_1, \ldots, p_i$ (also called an N-terminal fragment) and suffix fragment $p_{i+1}, \ldots, p_n$ (also called a C-terminal fragment). Since the original whole peptide, called the precursor ion, is charged, it is also possible for its fragments to retain a charge. Such charged fragments are also called ion fragments, and only they can be detected by a mass spectrometer. During the fragmentation process it is common for ion fragments to have neutral losses, which are chemical groups such as H_2O or NH_3 that get detached from the peptide fragments.

Table 1 lists some of the common fragment ions detected in low-energy ion trap MS/MS, which we chose to include in our model, along with their offsets from the cleavage site and the probabilities of detecting them in our data set. In typical ion trap mass spectra, ion fragment peaks are not detected in the low- and

(20) Fu, Y.; Yang, Q.; Sun, R.; Li, D.; Zeng, R.; Ling, C. X.; Gao, W. *Bioinformatics* **2004**, *20*, 1948−1954.
(21) Bafna, V.; Edwards, N. *Bioinformatics* **2001**, *17*, S13−S21.
(22) Havilio, M.; Haddad, Y.; Smilansky, Z. *Anal. Chem.* **2003**, *75*, 435−44.
(23) Elias, J. E.; Gibbons, F. D.; King, O. D.; Roth, F. P.; Gygi, S. P. *Nat. Biotechnol.* **2004**, *22*, 214−219.
(24) Colinge, J.; Masselot, A.; Giron, M.; Dessingy, T.; Magnin, J. *Proteomics* **2003**, *3*, 1454−63.

high-mass ranges. We therefore define the visible spectrum as the mass range in which intensity peaks appear, which corresponds to the masses between the spectrum's peak with the lowest mass and the spectrum's peak with the highest mass (in our data set, the visible range covers 77% of an average spectrum). Table 1 reports the probabilities of detecting fragment ions both in the visible spectrum range and in the entire spectrum range.

Our data are derived from doubly charged precursor ions, so the doubly charged ions b^{2+}, y^{2+} are possible fragments and are included in our model. The fragments can be classified into two groups: prefix N-terminal fragments (b- and a-ions and their derivatives), and suffix C-terminal fragments (y-ions and their derivatives). If a cleavage of the peptide occurs at mass M between amino acids i and $i + 1$, we can define the expected position for each of the fragment ions according to their offsets that appear in Table 1.

The mass spectrum data obtained from the mass spectrometer consist of a list (m_1, i_1), (m_2, i_2), ..., (m_j, i_j), of peak masses and their corresponding intensities, coupled with the experimental parent mass. Though the charge of the precursor ion is not always reported by the mass spectrometer, it can usually be derived from the data.[25] It is more difficult to determine the appropriate charge z for each peak, and therefore, the recorded masses m_1, ..., m_j are in fact the ratios m/z of the fragments. Our training data consists of labeled spectra (i.e., MS/MS spectra with their known peptide sequences).

De Novo Peptide Sequencing Problem and Spectrum Graphs. When the mass spectrometer is given a sample containing molecules of a peptide P, it fragments them using CID, and records the observed fragment masses and intensities in a mass spectrum S. This process can be viewed as drawing the spectrum S from the space of all mass spectra, according to a complex probability distribution Prob(S|P), where Prob(S|P) is governed by many factors such as the chemical composition of P, the mass spectrometer's properties, the experimental conditions, etc. The goal of sequencing algorithms is to find the peptide P that is the most likely source of S, i.e., the peptide P that maximizes Prob(S|P) among all possible peptides. Since the distribution Prob(S|P) is not available to us and is too complex to model, sequencing algorithms resort to using rough approximations in the form of scoring functions.

The space of all peptides is extremely large, making it inappropriate for an exhaustive case-by-case analysis. Database search algorithms reduce the size of the search space, by restricting their candidate peptides to ones that belong to the set of proteins present in the database. Most de novo algorithms restrict their search space to peptides that are paths in a spectrum graph, rather than all sequences in a database. A spectrum graph[9] is a directed acyclic graph; its vertexes correspond to putative prefix masses (cleavage sites) of the peptide. Two vertexes are connected by a directed edge from the vertex with the lower mass to the one with a higher mass if the difference between them equals the mass of an amino acid. The Sherenga algorithm,[9] uses a spectrum graph to sequence peptides by finding the highest scoring paths in the graph. The algorithm assumes that there is a set of k ion fragment types {y, b, y $-$ H_2O, ...}, with a set of corresponding offsets from the cleavage site $\Delta = \{\delta_1, ..., \delta_k\}$. The vertexes in the spectrum graph are assigned by creating for each mass s_i in the experimental spectrum a set of k vertexes at masses $s_i + \delta_1$, ..., $s_i + \delta_k$. Vertexes $s_i + \delta_j$ and $s_{i'} + \delta_{j'}$ having similar masses are merged (since it is likely that they are created by different ion fragments from the same cleavage site). The vertexes are scored according to a probability-based score that gives premiums for present fragment ions, and penalties for missing ones.

Peak Offset Tolerance and Noise. The measurements reported by mass spectrometers are not always accurate. It is often the case that fragments are detected at slight offsets from their theoretical positions. In our scoring scheme, we tolerate offsets of up to $\pm\epsilon = 0.5$ Da of peak locations from their expected positions. The intensity of a fragment with expected mass x is determined by examining the peaks detected in the interval $[x - \epsilon, x + \epsilon]$ in the spectrum. Using $M = m$ as a putative cleavage site in the peptide, the fragment offsets define a set of intervals or bins $B_m = \{[b - \epsilon, b + \epsilon], [y - \epsilon, y + \epsilon], ...\}$, which corresponds to the possible locations of the fragment peaks. Each interval in B_m is centered at its fragment's expected offset. For example, assuming $\epsilon = 0.5$, we get that the bin for the b-ion is $[m + 0.5, m + 1.5]$, the bin for the y-ion is $[(PM - 18) - m + 18.5, (PM - 18) - m + 19.5]$, etc. When examining a cleavage at mass m, our scoring method requires us to know the intensity levels for each of the possible fragment ions. We define the vector of ion fragment intensities $\vec{I} = \langle I_b, I_y, ... \rangle$ to be the maximal intensity detected in each of the fragments' bins in B_m. If for some fragment bin there is no peak that falls within it, we assign that bin's intensity to be 0.

Mass spectra typically contain many peaks for which there is no interpretation. In fact, in a typical mass spectrum, most of the peaks are not annotated (though the majority of the high-intensity peaks usually are). Some of these peaks are not annotated due to the limitations of our models. For example, they can belong to rare fragments (like x ions, or a $-$ H_2O $-$ H_2O, etc.). They can also be the result of complex events that are not covered by our model such as fragments from multiple cleavage sites on the same peptide (internal fragments). Another likely source for unannotated peaks are chemical contaminants and machine error. All these unannotated peaks are considered noise in the spectrum. The presence of many noisy peaks makes de novo sequencing difficult, since the noisy peaks can cause random false matches with ion fragments. In our data, the probability that a random peak matches an ion fragment's position is ~0.1 (for the visible region of the spectra). Though it might seem that this means that some of the low-probability ion fragments mentioned in Table 1 are not distinguishable from noise (e.g., y $-$ H_2O $-$ H_2O, b $-$ H_2O $-$ H_2O, etc.), this not always the case. As explained below where the probabilistic network is described, there are certain situations in which these fragments do contribute to the score, such as when we consider them in a combination with other ion fragments, or if they appear in sparse regions of the spectrum.

Discretizing Intensities and Cleavage Positions. Our scoring uses two types of continuous values that need to be transformed into discrete values, which are more convenient to use with our models. We consider the spectrum peak intensities to have continuous values; therefore, we experimentally derived

(25) Sadygov, R. G.; Eng, J.; Durr, E.; Saraf, A.; McDonald, H.; MacCoss, M. J.; Yates, J. R., III. *J. Proteome Res.* **2002**, *1*, 211−215.

thresholds to transform these intensities into k discrete intensity levels. Since depending on the experimental conditions, spectra can have total intensities that span several orders of magnitude, we assigned the peaks' relative intensity levels. This is done by calculating for each spectrum a baseline grass intensity, which equals the average of the intensities of the weakest 33% of the peaks in the spectrum. We then divide each peak's intensity by the grass level, to determine its normalized intensity. Using the training data, we experimented with several different numbers of intensity levels and different threshold values to separate between intensity levels. Let I denote the normalized intensity level of a peak; we obtained optimal results using the following four intensity levels: 0 (zero) is assigned to peaks with $I < 0.05$, 1 (low) is assigned to peaks with $0.05 \leq I < 2$ (62% of the peaks in the training data), 2 (medium) is assigned to peaks with $2 \leq I < 10$ (26% of the peaks), and 3 (high) is assigned to peaks with $I \geq 10$ (12% of the peaks).

The other type of value discretized in our models is the relative position of a cleavage site m in a peptide of mass PM. The relative position is defined as $\text{pos}(m) = m/\text{PM}$. We discretized the values of $\text{pos}(m)$ into $k = 5$ equally sized regions labeled 0, ..., $k - 1$; i.e., $\text{pos}(m) = 0$ denotes a cleavage in the first fifth of the peptide near the N-terminal, $\text{pos}(m) = 1$ denotes a cleavage in the second fifth, etc. We added this variable to our model because the intensity of observed peaks is correlated with the region in the peptide in which the peaks appear. On average, peaks are stronger in the center of the peptide and are weak or missing near the terminal ends.

NEW LIKELIHOOD SCORING METHOD

In this section, we propose a scoring scheme that assigns a relevance score to peptide prefix masses (which are the vertexes of the spectrum graph). For each mass m our scoring function determines by examining the peaks in spectrum S, how likely it is that there was a cleavage of a peptide at mass m, i.e., that m is the mass of a prefix of the peptide P that created the spectrum S.

Hypothesis Test. At the heart of our scoring function is a hypothesis test. Hypothesis tests are used by several existing scoring functions.[9,22−24] Our hypothesis test compares between two competing hypotheses concerning a spectrum S and a mass m of a possible cleavage site. The first hypothesis is the CID hypothesis, which states that m is a genuine cleavage in the peptide that created S. According to this hypothesis, there are rules that govern the outcome of a fragmentation. In particular, there are certain combinations of fragments and intensities that are more probable than others. We use a probabilistic network that models these fragmentation rules to determine the probability $P_{\text{CID}}(\vec{I}|m, \text{S})$ of detecting an observed set of fragment intensities $\vec{I}$, given that mass m is a cleavage site in the peptide that created S. The competing hypothesis is the random peaks hypothesis (RAND), which assumes that the peaks in the spectrum are caused by a random process. Thus, the intensities that appear in $\vec{I}$, which supposedly belong to fragment ions due to a cleavage at mass m, are in fact only random peaks that happen to fall into the designated bins. We describe how to compute the probability $P_{\text{RAND}}(\vec{I}|m, \text{S})$ in this scenario.

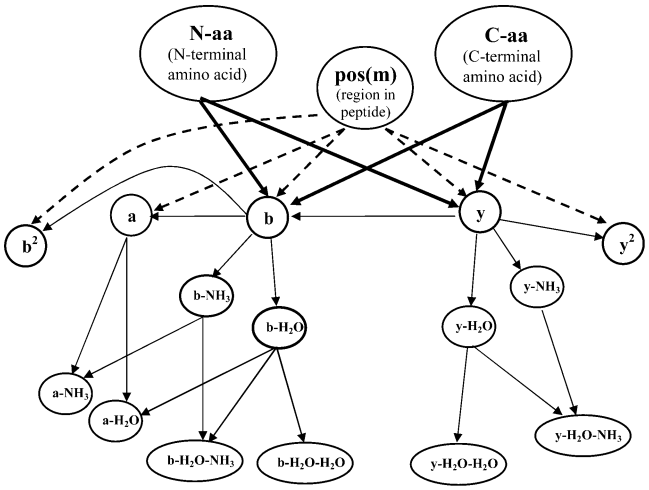

Figure 1. Probabilistic network for the CID fragmentation model of doubly charged tryptic peptides measured in an ion trap mass spectrometer. Three different types of relations are modeled in this network: (1) correlations between fragment ions (regular arrows); (2) dependencies due to the relative position of the cleavage site in the peptide (dashed arrows); (3) influence of flanking amino acids to the cleavage site (bold arrows).

The score given to a mass m and spectrum S is the logarithm of the likelihood ratio of these two hypotheses,

$$\text{Score}(m, \text{S}) = \log \frac{P_{\text{CID}}(\vec{I}|m, \text{S})}{P_{\text{RAND}}(\vec{I}|m, \text{S})} \quad (1)$$

A positive score in eq 1 means that, according to our models, it is more likely that the peak intensities $\vec{I}$ were caused by a genuine cleavage event (the higher the score, the likelier this hypothesis is, compared to the competing random hypothesis). Likewise, a negative score means that the observed intensities $\vec{I}$ are probably due to random peaks, and they give no credence to a cleavage at mass m. We now describe in detail how to compute the probabilities $P_{\text{CID}}(\vec{I}|m, \text{S})$ and $P_{\text{RAND}}(\vec{I}|m, \text{S})$, under these two different hypotheses.

Collision-Induced Dissociation Hypothesis. According to the CID hypothesis, there are rules that govern the outcome of the peptide's fragmentation process in the mass spectrometer. These rules define which ion fragments and which peak intensities are more likely to be observed. We chose to include in our CID model three types of factors that are a result of mass spectrometry fragmentation rules: (1) dependencies and correlations between types of fragment ions; (2) the positional influence of the cleavage site (the influence of the relative region in which the fragmentation occurs); (3) the influence of the type of amino acids directly N-terminal and C-terminal to the proposed cleavage site.

At this stage, we restrict our discussion of the scoring method to include only the first two factors mentioned above. The incorporation of the effect of the flanking amino acids to the cleavage site is treated separately in a section below.

Figure 1 illustrates the probabilistic network (described by a directed acyclic graph) that we use to model the common fragments resulting from a peptide cleavage. A vertex u in the graph is called a parent of vertex v if the there is a directed edge (u, v) in the graph. There are three vertexes in our graph without

parents, two which involve the amino acids flanking the cleavage site (the vertexes N-aa and C-aa), and the vertex pos(m), which holds the relative region in the peptide in which the cleavage occurs. All other vertexes are labeled with the fragment types from Table 1. Each of these vertexes holds a conditional probability table of the value of the vertex given the values of its parents. For instance the vertex b holds a table which gives the probability $P(b = I_b|y = I_y, \text{pos}(m) = r)$, where I_b is the intensity detected for the b-ion, I_y is the intensity detected for the y-ion, and the cleavage occurred in the peptide at region r. We explain the method in which we learned the model's probability tables in the Experimental Section. After exploring several network configurations, which included different types of fragments, and various ways to connect between them, we found that the structure depicted in Figure 1 gives the best results. Note that this structure reflects fragmentation rules that arise from our training data, which consists of spectra of doubly charged tryptic peptides obtained from an ion trap mass spectrometer. Spectra from other types of mass spectrometers, charge states, or proteolytic enzymes can lead to significantly different fragmentation rules.

The edges that appear in the graph reflect two types of dependencies and causal relations (at this stage we ignore edges emitting from the vertexes N-aa, C-aa). The first type of dependencies modeled are the correlations between the intensity levels of the ion fragments. Though to some extent there is a correlation between all ion types, some combinations tend to display higher correlation in their intensities. For instance, the b- and y-ions are highly correlated. In ion trap data, when a cleavage in a doubly charged tryptic peptide creates a high-intensity y-ion, there is usually also a high-intensity b-ion. We model this phenomenon by adding an edge between the vertex y and the vertex b (the direction of the edge in this case is arbitrary). The extent of this dependence can be seen when we examine the probability tables. For instance, the probability of seeing a strong b-ion in the center of the peptide, given that there is a strong y-ion, is $P_{\text{CID}}(I_b = \text{high}|I_y = \text{high}, \text{pos}(m) = 2) = 0.36$, and it drops to 0.03, if instead of the strong y-ion, a weak y-ion is detected (I_y = low). This large difference in probabilities is due to the fact that, in spectra of tryptic peptides, the y-ions are usually stronger than their b counterparts.[26] It is therefore unlikely to detect a case where the b-ion is much stronger than the y-ion. Dependencies of this type were not accounted for in previous de novo scoring models and adding them to our score led to improved performance. The complete set of probability tables used in our model can be found in the Supporting Information.

There are also correlations between the intensities of ion fragments and their neutral losses. For instance, if we do not detect a b-ion, we are less likely to detect a b $-$ H_2O ion. This is reflected in the probability tables by the values $P_{\text{CID}}(I_{b-H_2O} > \text{zero}|I_b = \text{high}) = 0.496$ for detecting a b $-$ H_2O ion when a strong b-ion was also detected, compared to the probability $P_{\text{CID}}(I_{b-H_2O} > \text{zero}|I_b = \text{zero}) = 0.242$ of detecting a b $-$ H_2O ion when no b-ion was detected. Not all the correlations between ion fragments have edges in our model's graph. For instance, a strong y-ion can indicate that the intensity of other prefix fragments will also be high (besides the b-ion, which is correlated with y). In this case, it might be reasonable to add edges from y to other prefix fragments; however, the information about y can be mediated quite well by the value of b (since a strong y is likely to be coupled with a strong b). In the interest of simplifying our model, we chose not to add those edges.

The second type of dependency modeled in our graph is the effect of the region in which the cleavage occurs (the vertex pos(m)). There are edges from the vertex pos(m) to the vertexes b, y, a, y^{2+} and that b^{2+}, because the intensity of these fragments depends on where in the peptide the cleavage occurred. For instance, y- and b-ions tend to have higher intensities in the middle of the peptide, whereas they are hardly detected near its ends.[22] a-Ions tend to be detected more when cleavages occur in the first half of the peptide. Since it is more likely for larger fragments to retain both charges in doubly charged peptides, the b^{2+} ions are observed more often when the cleavage occurs toward the C-terminal, whereas the y^{2+} ions are observed more often when the cleavage is closer to the N-terminal. Of course the cleavage location also has a strong influence on the rest of the fragment ions, but for the benefit of a simpler model, we chose to omit these edges.

The reason it is beneficial to simplify probabilistic networks becomes clear when we examine how the model complexity is affected by the addition of an edge. Each additional edge that points to a vertex adds a dimension to the probability table of that vertex. Assuming there are x edges entering a vertex and there are k discrete intensity levels, the size of the probability table at the vertex is k^{x+1}. As described in the Experimental Section, we only had a limited number of labeled spectra to train our model. Therefore, we could not add many edges between vertexes that describe true dependencies for fear of complicating our model. When limited amount of data is used to train a complex model, there is a chance that overfitting will occur. When this happens, the model's parameters are too biased toward fitting the training data and do not generalize well to accommodate data that are different from the samples in the training set.

We use the probabilistic network of Figure 1 to compute $P_{\text{CID}}(\vec{I}|m, \text{S})$, the probability of observing ion fragment intensities $\vec{I}$ given that the putative cleavage occurred at mass m in spectra S. We denote by $V = \{\text{b, y, ...}\}$ the vertexes in the probabilistic network, excluding the vertexes pos(m), N-aa and C-aa. For each vertex $v \in V$, $\pi(v)$ denotes the set of v's parents in the graph ($\pi(v)$ are the vertexes that have edges pointed from them to v), and $\vec{\pi}(v)$ denotes the set of values assigned to the vertexes $\pi(v)$. $P_{\text{CID}}(I_v = i|\vec{\pi}(v) = \{i_1, i_2, ...\})$ is the probability of detecting the intensity i at fragment ion v given the intensities detected at its parents. According to the properties of this type of probabilistic network, a vertex v is independent of the other vertexes in the graph given that the values of its parents are known (this network is a casual network with all the vertexes instantiated[27]). This leads to the following decomposition for the probability of the intensities $\vec{I}$.

$$P_{\text{CID}}(\vec{I}|m, \text{S}) = \prod_{v \in V} P_{\text{CID}}(I_v|\vec{\pi}(v), m, \text{S}) \qquad (2)$$

(26) Tabb, D. L.; Smith, L. L.; Breci, L. A.; Wysocki, V. H.; Lin, D.; Yates, J. R., III. *Anal. Chem.* **2003**, *75*, 1155−1163.

Since the values in the conditional probability tables in our model were derived from true mass spectrometry data and represent some of the rules governing the fragmentation process, the probability P_{CID} can help distinguish between likely combinations of ions (that are frequent in real cleavage sites) and unlikely combinations. For instance, the probability assigned to instances where both ions and their neutral losses are detected should be higher than unlikely instances such as ion combinations where neutral losses are detected without the b- or y-ions registering any intensity.

The fact that our model considers combinations of ion fragments makes it possible, in certain situations, for low-probability fragments to contribute to the scoring. This happens because our model considers the fragment's intensity in context with other fragments and can identify combinations that have a probability that deviates from the random background probability. For instance, the average probability of detecting a y $-$ H_2O $-$ H_2O ion fragment is 0.11 (see Table 1) and thus should be virtually indistinguishable from random noise peaks (that have probability 0.1). However, when it is considered together with the y $-$ H_2O fragment, there are combinations for which the probability of detecting the y $-$ H_2O $-$ H_2O is higher, for example, P(y $-$ H_2O $-$ H_2O = medium|y $-$ H_2O = high) = 0.17. Thus, most of the time, the intensity of y $-$ H_2O $-$ H_2O does not contribute to the score, since its probability is similar to the random noise. However, using our model, we are able to identify the conditions in which it significantly deviates from the random probability and exploit these few occasions to our benefit. We demonstrate how using such low-probability fragment ions improves the performance of PepNovo in the Experimental section.

Random Peak Hypothesis. The random model assumes that peaks are distributed according to some simple prior distribution throughout the spectrum, without there being any special cleavage sites or fragmentation rules that influence the detection of peaks at certain offsets. When we observe the intensities of $\vec{I}$ from a cleavage at mass m, any peak matches with fragment bins are considered to be due to chance. Under this random model, each peak is distributed independently of the others. Thus, the probability $P_{RAND}(\vec{I}|m, S)$ can be computed as the product of the probabilities of seeing the individual peaks in their bins.

To compute the probability of randomly seeing a peak with intensity level t in a bin of width 2ϵ around mass m', we use an empirical estimate of the peak density in the vicinity of m'. This local density estimation is used because peaks are not distributed uniformly throughout the spectrum mass range. For instance, peaks tend to be stronger and denser toward the center of the spectrum and sparser and weaker near the terminal ends. The density estimation is done by looking at a window of width w around the mass m' and counting how many peaks of each intensity level i appear in this window. Assuming there are d different intensity levels, we denote these counts by n_i, $1 \le i \le d$. Figure 2 illustrates such a count.

Let $\alpha = 1 - (2\epsilon/w)$ be the probability of uniformly selecting a random location for a single peak in a window w and having it fall outside a specified bin of width 2ϵ. The probability that the highest intensity level for a peak detected in a bin centered at m' is $t \ge 1$, given the peaks counts n_1, ..., n_d in a window of width w around m', is given by the following equation

$$P_{RAND}(I = t|n_1, n_2, ..., n_d) = (1 - \alpha^{n_t}) \cdot \alpha^{\sum_{i=t+1}^{d} n_i} \quad (3)$$

Equation 3 can be explained as follows. If the maximal intensity in the bin is t, we want to avoid the case where all the peaks of intensity t in the window w miss the bin (we do not mind if several peaks with intensity t or lower happen to also fall in the bin). The probability that a random placement of all peaks with intensity t misses the bin is α^{n_t}, so the complimentary event where at least one peak with intensity t falls in the bin has probability $1 - \alpha^{n_t}$. As for the higher intensity peaks, we want them all to miss the bin, and the probability that that occurs is $\alpha^{\sum_{i=t+1}^{d} n_i}$. Following this reasoning, the probability that no peak falls in a bin is given by

$$P_{RAND}(I = 0|n_1, n_2, ..., n_d) = \alpha^{\sum_{i=1}^{d} n_i} \quad (4)$$

Equation 3 together with eq 4 defines a probability density function for which

$$\sum_{i=0}^{d} P_{RAND}(I = i|n_1, n_2, ..., n_d) = 1 \quad (5)$$

We assume that, in the random model, the events of detecting peaks in bins are independent of each other. Therefore, we can factor the probability $P_{RAND}(\vec{I}|m, S)$ of detecting a combination of our model's k fragments' intensities into the product of the individual probabilities, as follows

$$P_{RAND}(\vec{I}|m, S) = \prod_{i=1}^{k} P_{RAND}(I_i|n_{i1}, n_{i2}, ..., n_{id}) \quad (6)$$

By examining eq 3, we can gain insight on how the random model helps to balance the effects of noise. When many noisy peaks are present (typically having low intensity), they can cause random matches and thus supposedly increase the score for a

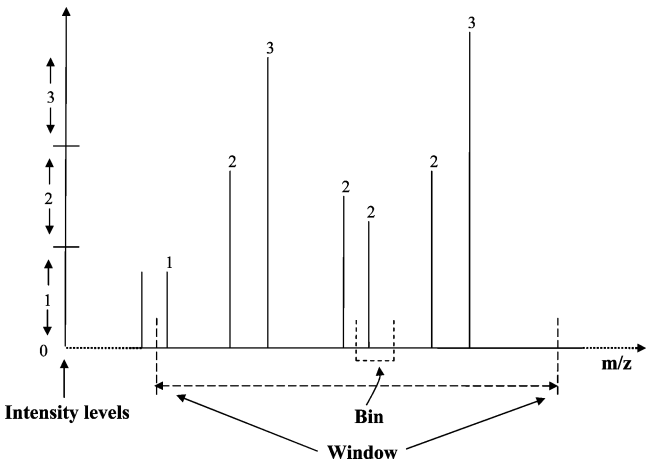

Figure 2. Window placed around a bin. There are $n_1 = 1$, $n_2 = 4$, and $n_3 = 2$, peaks of the respective intensity levels in the window. The designated bin contains a single peak with intensity 2.

(27) Jensen, F. V. *Bayesian Networks and Decision Graphs*; Springer-Verlag: New York, 2001.

cleavage site. However, if we look at eq 3, we see that increasing the peak count for the low-intensity peaks also increases the probability of detecting such a peak by chance. Since the probabilities of the random model appear in the denominator of the score equation (see eq 1), the result is a decrease in the score. Thus, if an ion fragment is detected in a dense region of the spectrum, it contributes less to the score compared to the contribution it would bring had there been only a few peaks in its vicinity. This correction does not occur when a simple random model is used, such as the one used by Dancik,[9] where the same constant random probability is used for all regions in the spectrum.

Modeling the Influence of Flanking Amino Acids. Recent research has uncovered many chemical properties and pathways that influence the outcome of the CID fragmentation process. It has been suggested that incorporating such information can improve scoring function.[26,28] A recent scoring function that uses this type of information obtained high accuracy for database searches;[23] however, incorporating such information into de novo sequencing algorithms is an open problem.

An amino acid is said to have an N-terminal bias if on average, the b and y peaks at the cleavage site N-terminal to the amino acid are stronger than the peaks from the cleavage on its C-terminal side. Similarly, an amino acid exhibits C-terminal bias if the average intensity of peaks from the cleavage C-terminal to the amino acid is stronger than the N-terminal ones. Some of the prominent amino acid biases and preferred cleavage sites that have been mentioned in the literature are as follows: (1) N-terminal bias of proline, glycine, and serine;[29,26](2) C-terminal bias of aspartic acid[30] (especially in proteins with no mobile protons[31]); (3) influence of histidine on cleavage C-terminal to acidic residues.[32]

A qualitative measurement of some of the aforementioned phenomena is given in refs 26 and 33. Some of these biases are very strong, for instance the b and y peaks N-terminal to proline are typically at least 5 times stronger than their C-terminal counterparts. Adding this information into the model can help to determine genuine cleavage sites.

We incorporate the amino acid biases into our model by adding the vertexes N-aa and C-aa (see Figure 1) and adding directed edges from them to the vertexes *b* and *y*. These edges add two conditioning variables to the conditional probability tables for *b* and *y*. Since there are 20 different amino acids, adding these variables makes the conditional probability tables for *b* and *y* 400 times larger. This large increase in table size requires much more training data than we have available to us. To reduce the number of parameters needed to train, we grouped the different amino acid combinations into 16 equivalence sets. The assignment of amino acid pairs to equivalence sets is done according to the order rank of the sets; i.e., any two amino acids are assigned to the highest ranking set to which they can belong. The equivalence sets we use are as follows (X denotes any amino acid, we start our list from the highest ranked set): X-Pro, Pro-X, X-Gly, Gly-X, X-Arg/Lys, His-X, X-His, Asp/Glu-X, X-Asp/Glu, Ile/Leu/Val-X, X-Ile/Leu/Val, Ser/Thr-X, X-Ser/Thr, Asn-X, X-Asn, and X−X (any combination of two amino acids). A table describing this assignment of amino acids to equivalence sets is given in the Supporting Information. If both amino acids in the pair are either glycine or proline, we assign the combination to the X−X set (since there is less cleavage in these cases[33]). The set's order was determined according to the extent each amino acid influences the intensities of the peaks at a cleavage site and causes a deviation from the typical cleavage intensities (we determined this based on the results mentioned in refs 28 and 33). For instance, in most cases, proline and glycine have a stronger influence than the other amino acids; therefore, they are placed at the top of the list. Note that, by using such equivalence sets, we actually model the influence of only one of the flanking amino acids each time, though it is usually the dominant one (since the sets with the dominant amino acids appear higher in our ranking). A more accurate approach might be to model the contribution of both flanking amino acids;[28] however, as mentioned above, this requires a larger training set than the one that was available to us.

Using the expanded conditional probability tables, we can replace the probability $P_{\mathrm{CID}}(\vec{I}|m, \mathrm{S})$ of eq 2 with $P_{\mathrm{CID}}(\vec{I}|m, \mathrm{S}, \text{N-aa}, \text{C-aa})$. Note that adding the conditioning on the N- and C-terminal amino acids only affects the probabilities of the fragments b and y. The other fragments' tables are not affected by this, for instance, $P_{\mathrm{CID}}(I_{\mathrm{y-H_2O}} = i_1|I_{\mathrm{y}} = i_2, \text{N-aa}, \text{C-aa}) = P_{\mathrm{CID}}(I_{\mathrm{y-H_2O}} = i_1|Iy = i_2)$. Furthermore, there is no need to make any changes to the random model because of the added conditioning on the flanking amino acids, since it is assumed in that model that the peaks are created in a random process that is not governed by the fragmentation of any source peptide.

The addition of the N-aa and C-aa vertexes to our model changes the way we score vertexes in the spectrum graph. Before the addition, each vertex in the spectrum graph had a single score. Now, each vertex can have 16 different scores (for the different combinations of flanking amino acids). When searching for the high-scoring path, the de novo algorithm must select for each vertex its appropriate score, depending on the edges that enter and exit the vertex.

EXPERIMENTAL SECTION

In this section, we describe how we constructed our experiments. We also provide benchmarks that compare PepNovo with three existing de novo programs: Sherenga,[9] Lutefisk,[11] and Peaks.[14]

Mass Spectra Data Set. The data set we use is composed of doubly charged tryptic peptides obtained from low-energy ion trap LC/MS/MS runs. We limited our experiments to only dealing with spectra of doubly charged precursor ions since this charge state is the most common in many mass spectrometry experiments. In total, we obtained 1252 spectra of peptides with unique sequences which were identified with high confidence by Sequest (these spectra had Xcorr > 2.5 and came from proteins with multiple hits). Our data came from two sources, the ISB protein

(28) Schutz, F.; Kapp, E. A.; Simpson, R. J.; Speed, T. P. *Biochem. Soc. Trans.* **2003**, *31*, 1479−83.
(29) Berci, L. A.; Tabb, D. L.; Yates, J. R.; Wysocki, V. H. *Anal. Chem.* **2003**, *75*, 1963−1971.
(30) Gu, C.; Tsaprailis, G.; Breci, L.; Wysocki, V. H. *Anal. Chem.* **2000**, *72*, 5804−5813.
(31) Wysocki, V. H.; Tsaprailis, G.; Smith, L. L.; Breci, L. A. *J. Mass Spectrom.* **2000**, *35*, 1399−1406.
(32) Huang, Y.; Wysocki, V. H.; Tabb, D. L.; Yates, J. R., III. *Int. J. Mass Spectrom.* **2002**, *219*, 233−244.
(33) Huang, Y.; Triscari, J. M.; Wysocki, V. H.; Pasa-Tolic, L.; Anderson, G. A.; Lipton, M. S.; Smith, R. D. *J. Am. Chem. Soc.* **2004**, *126*, 3034−3035.

mixture data set,[34] which used an ESI-ITMS mass spectrometer made by ThermoFinnigan (San Jose, CA), and the Open Proteomics Database (OPD),[35] which used a ESI-ion trap Dexa XP Plus mass spectrometer, also from ThermoFinnigan. Ideally, a scoring model should be trained using spectra from a single type of machine. However, to create a sufficiently large training set, we resorted to using spectra from these two different sources (although both are ESI-ion trap machines that produced spectra with similar characteristics).

For our test set, we selected from the data set described above 280 spectra of peptides with a molecular mass of up to 1400 Da (which corresponds to peptides with 7−16 amino acids, with an average length of 10.5). The peptide sequence assignments to the 280 spectra were verified by an independent run of Sequest against a 20 Mb nonredundant protein database with nonspecific digestion.

PepNovo Sequencing Algorithm. In this section, we briefly describe parameter learning in our models, the construction of the spectrum graph, and the dynamic programming algorithm for finding the highest scoring path in the spectrum graph.

Training the Probabilistic Model. After setting aside 280 spectra for a test set, we were left with 972 annotated spectra as a training set that were used to learn the probability tables. The tables were filled by counting in the training data the number of appearances of each possible combination of variables in the table. Some variable combinations did not appear, resulting in zero counts. We smoothed these zero counts by adding a small uniform count to all combinations.

It is worth noting that since the training data we used all came from the same type of source (all doubly charge tryptic peptides from ion trap machines), the models that are trained are most appropriate for spectra from this type of machine. Mass spectra that are generated under different experimental conditions, for example, other types of mass spectrometers such as Q-TOF, are likely to have different fragmentation rules and different probabilities for observing combinations of fragments. It is preferable to train separate scoring models for different types of data.

Constructing the Spectrum Graph. The vertexes in the spectrum graph represent possible cleavage sites, and the solution interpretations correspond to high-scoring paths in the graph. For this reason, selecting the appropriate number of vertexes for the spectrum graph is essential for obtaining optimal results. On one hand, if too few vertexes are selected, many cleavage sites can be missed, and the graph might contain several disconnected subpaths of the correct solution. On the other hand, if too many vertexes are used, this causes many spurious edges, which create high-scoring incorrect subpaths that add noise, which masks the correct path.

Our method for determining the graph's vertexes is as follows. Given a query spectrum S, we first select part of the peaks in the spectrum, choosing only the strongest peaks in each region. This is done by sliding a window of width w across the spectrum and keeping any peaks that are in the top k peaks, for some window location. For $w = 56$ Da and $k = 3$, this selects on average 62 peaks per spectrum, which is a density of 5.2 peaks for every 100 Da. Since the highest peaks in the spectrum tend to be b- and y-ions, we create vertexes for both of these interpretations: Given a peak at mass x, we create a vertex at mass $m = x - 1$ (by interpreting the peak as a b-ion) and also create a vertex at mass $m = \mathrm{PM} - x + 1$ (by interpreting the peak as a y-ion). To these vertexes we add the vertexes for mass 0 (the empty peptide), and mass PM − 18 (which is the mass of the entire peptide). We merge vertexes that are within 0.5 Da of each other (since they are likely created from b- and y-ions of the same cleavage site). When following this procedure, the average number of vertexes in a spectrum graph for the test set is 110. Note that this method is different from the method used by Dancik et al., where all peaks (and all their interpretations) were used to select the vertexes in the spectrum graph. The edges in the graph are created by connecting vertexes that have a mass which approximately equals the mass of an amino acid (we used a tolerance of ±0.5 Da).

Scoring vertexes in the spectrum graph is done by taking each vertex's mass m and finding the intensities $\vec{I}$ of the fragment ions for a cleavage at mass m in the original spectrum S (containing all peaks). We then score the vertex according to the log-likelihood score of eq 1. Note that each vertex has several scores computed for it according to the different combinations of flanking amino acids. When performing its search for a high-scoring path, our search algorithm selects the appropriate score for the vertex, according to the combination of edges it uses in the path that goes through that vertex.

It is common in mass spectra for peaks to have isotopic peaks that appear at increments of 1 Da after the peak. The isotopic peaks are caused by peptide fragments that contain isotopic atoms (the most common is isotope ^{13}C, but N, O, and S can also contribute to this). Isotopic peaks are usually detected for strong peaks; therefore, it is common for the b- and y-ions to have additional peaks at offsets of +1 and +2 Da. These isotopic peaks can create additional vertexes in the spectrum graph that can lead to sequencing errors. One approach to deal with isotopic peaks is to remove their vertexes from the graph. This, however, can lead to the removal of genuine vertexes that were created from peaks that happen to fall in the isotopic peak positions. Instead of using this approach, we chose to give vertexes a premium to the score if their b or y peaks had isotopic peaks ahead of them and give the vertexes a score penalty if their b or y peaks seemed to be isotopic peaks themselves (that is, they had strong peaks at an offset of −1 Da).

A point that needs to be kept in mind when constructing spectrum graphs is that the experimental parent mass measured in mass spectra machines is often inaccurate and can thus lead to mistakes in the de novo sequencing. To solve this problem, we use the combinatorial parent mass correction procedure from by Dancik et al.[9]

Dynamic Programming Algorithm. Once the spectrum graph is created and scored, we need to find a highest scoring antisymmetric path in it.[9] Since every peak we use from the spectrum contributes two vertexes to the spectrum graph, we could end up with symmetric paths that use both vertexes attributed to a peak. This leads to incorrect interpretations. Therefore, we restrict our solutions to paths containing at most one vertex from each of these "forbidden pairs" of vertexes. Though this problem is generally intractable, the unique structure of the forbidden pairs in the spectrum graphs leads to a polynomial

(34) Keller, A.; Purvine, S.; Nesvizhskii, A. I.; Stolyar, S.; Goodlett, D. R.; Kolker, E. *OMICS* **2002**, *6*, 207−212.

(35) Prince, J. T.; Carlson, M. W.; Wang, R.; Lu, P.; Marcotte, E. M. *Nat. Biotechnol.* In press.

Table 2. Comparison of De Novo Peptide Sequencing Algorithms[a]

algorithm	average accuracy	average length	predictions with correct subsequences of length at least x							
			$x = 3$	$x = 4$	$x = 5$	$x = 6$	$x = 7$	$x = 8$	$x = 9$	$x = 10$
PepNovo	**0.727**	10.30	**0.946**	**0.871**	**0.800**	**0.654**	**0.525**	**0.411**	0.271	**0.193**
Sherenga	0.690	8.65	0.821	0.711	0.564	0.364	0.279	0.207	0.121	0.071
Peaks	0.673	10.32	0.889	0.814	0.689	0.575	0.482	0.371	**0.275**	0.179
Lutefisk	0.566	8.79	0.661	0.521	0.425	0.339	0.268	0.200	0.104	0.057

[a] Cumulative results for 280 test spectra: the average accuracy of predicted amino acids, average prediction length, and proportions of predictions that had a correct subsequence of length at least x, for $3 \leq x \leq 10$.

time algorithm for the antisymmetric path problem.[10] To find the highest scoring path in the graph, we used a dynamic programming algorithm similar to the one due to Chen et al.[10,36] that is modified to take into account the particulars of our scoring function (see section on constructing the spectrum graph).

Assessing the Efficiency of De Novo Sequencing Algorithms. We desired a metric by which the success of de novo reconstructions could be evaluated and compared with other algorithms. The natural parameter we can look at is the prediction accuracy, which is defined as

$$\text{prediction accuracy} = \frac{\text{number of correct amino acids}}{\text{number of predicted amino acids}} \tag{7}$$

However, de novo sequencing algorithms often predict partial, rather than complete peptides, so a high score on this parameter can be obtained by only predicting high-scoring short partial peptides. Usually, this includes the portion in the center of the peptide that has stronger peaks, while amino acids near the terminals are ignored. We therefore also look at the capability of the algorithms to reconstruct correct consecutive subsequences of amino acids (that appear in the prediction in their expected position according to the correct peptide). For each prediction made by the algorithms, we determined the maximal correct subsequence and tallied the counts for the entire test set. Note that a predicted amino acid (or subsequence) is considered correct if its position in the predicted de novo sequence is within 2.5 Da from its expected position according to the correct sequence. We use this large margin to account for offsets in amino acid locations that occur due to both inaccurate peak m/z measurements and an incorrect parent mass (even after parent mass correction is used). In addition, we do not make a distinction between the amino acids leucine and isoleucine (which have identical masses) and between lysine and glutamine (which have a small difference of 0.04 Da in their masses).

PepNovo Benchmarking. We compared the performance of PepNovo with the following popular de novo sequencing algorithms: Lutefisk XP v1.0,[11] Peaks v2.4,[14] and Sherenga[9] (which is included in the Spectrum Mill v3.01 software suite).

We ran the algorithms on each of the 280 test spectra and kept the highest scoring interpretation they returned. The following parameters and settings were used for this benchmark. Lutefisk was run with the default parameters for doubly charged tryptic peptides on ion trap mass spectrometers. Peaks was run with an error tolerance of 0.6 Da, Trypsin digestion, and treating Q/K and I/L as identical amino acids. Sherenga was run with ESI ion trap scoring, minimum vertex score 0, and treating I/L and Q/K as identical amino acids.

The results of the four de novo algorithms are given in Table 2. PepNovo, Peaks, and Sherenga all achieve results superior to Lutefisk's, both in terms of accuracy and in terms of the longest correct subsequences predicted. As far as the prediction accuracy is concerned, PepNovo has the highest accuracy even though on average Sherenga makes shorter predictions and thus has an advantage since it is making more selective predictions (this enables it to get a higher accuracy than Peaks). When we examine the prediction of correct consecutive amino acid sequences, we see that PepNovo obtained the best results, with Peaks coming in a close second, especially when the longer subsequences are concerned.

We also ran additional experiments with deficient versions of PepNovo, where each variant of the algorithm was lacking one of the components that are incorporated into the PepNovo scoring model (e.g., dependencies between fragments, information on flanking amino acids, intensity thresholds, etc.) The results are given in the Supporting Information. Each of the tested components proved to have a positive influence on PepNovo's performance (since all deficient versions of PepNovo had inferior success rates). For instance, a version of PepNovo that did not use information on the flanking amino acids showed a reduction of 1.6% to the prediction accuracy. It is likely that the improvement in performance due to adding flanking amino acids to the model would be greater than 1.6% if more training data were available, enabling the inclusion of more equivalence sets, possibly to the degree of having a separate probability table for each pair of flanking amino acids. The lack of other components in the model, such not having intensity thresholds or using a simple random model, caused a larger decrease in the performance (see table in Supporting Information for more details). We also evaluated our de novo algorithm with Dancik scoring (which lacks many of PepNovo's enhancements) and found that PepNovo's scoring performs much better both in terms of the prediction accuracy (72.7% vs 61.2%) and in terms of the counts of the maximal lengths of correct subsequences in the predictions.

Future Work. The results obtained for PepNovo demonstrate the power of our new scoring model, which enabled our algorithm to outperform popular de novo algorithms. There are several possibilities for future related work in this area. Our algorithm can be extended to include modified amino acids in the predicted peptides. We intend to examine ways to add to our models'

(36) Lu, B.; Chen, T. *J. Comput. Biol.* **2003**, *10*, 1−12.

additional mass spectrometry "wisdom", such as the influence of the amino acid composition on the intensity of the neutral losses.[26,37] We also plan to expand our score models to include additional charge states (which might require more sophisticated methods for constructing the spectrum graph) and to create models for data from additional types of mass spectrometers such as Q-TOF and additional proteolytic enzymes (which will require additional annotated training sets).

ACKNOWLEDGMENT

We thank Andrew Keller for supplying the protein mixture data and to Edward Marcotte and his colleagues at the University of Texas for making the Open Proteomics Database publicly available. We also thank Richard Johnson for supplying the Lutefisk software, Bin Ma for running the Peaks benchmarks, and Karl Clauser for running the Sherenga benchmarks. We benefitted a lot from our discussions with Sanjoy Dasgupta, Vineet Bafna, Nuno Bandeira, Tim Chen, and Haixu Tang. This project was supported by NIH Grant NIGMS 1-R01-RR16522.

SUPPORTING INFORMATION AVAILABLE

Additional information as noted in text. This material is available free of charge via the Internet at http://pubs.acs.org.

Received for review August 14, 2004. Accepted October 29, 2004.

AC048788H

(37) Colinge, J.; Masselot, A.; Magnin, J. *Algorithms in Bioinformatics, Third International Workshop, WABI*; 2003; pp 25−38.