Protein Sequencing and Identification by Mass Spectrometry
Outline

• Tandem Mass Spectrometry
• De Novo Peptide Sequencing
• Spectrum Graph
• Protein Identification via Database Search
• Identifying Post Translationally Modified Peptides
• Spectral Convolution
• Spectral Alignment
Masses of Amino Acid Residues

Aspartate

Leucine
Protein Backbone

H...-HN-CH-CO-NH-CH-CO-NH-CH-CO-...OH

N-terminus

AA residue_{i-1}  AA residue_i  AA residue_{i+1}

C-terminus
Peptide Fragmentation

- Peptides tend to fragment along the backbone.
- Fragments can also lose neutral chemical groups like NH$_3$ and H$_2$O.
Breaking Protein into Peptides and Peptides into Fragment Ions

- Proteases, e.g. trypsin, break protein into peptides.
- A Tandem Mass Spectrometer further breaks the peptides down into fragment ions and measures the mass of each piece.
- Mass Spectrometer accelerates the fragmented ions; heavier ions accelerate slower than lighter ones.
- Mass Spectrometer measures mass/charge ratio of an ion.
N- and C-terminal Peptides

N-terminal peptides: GPPF

C-terminal peptides: PFNA
Terminal peptides and ion types

Peptide: G P F N H$_2$O
Mass (D): $57 + 97 + 147 + 114 = 415$

Peptide: G P F N without H$_2$O
Mass (D): $57 + 97 + 147 + 114 - 18 = 397$
N- and C-terminal Peptides

N-terminal peptides

C-terminal peptides
N- and C-terminal Peptides

486

415

301

154

57

N-terminal peptides

C-terminal peptides

71

185

332

429
N- and C-terminal Peptides
N- and C-terminal Peptides

Reconstruct peptide from the set of masses of fragment ions (mass-spectrum)
Peptide Fragmentation

\[ \text{H} - \text{N} \cdots \text{C} \cdots \text{C} \cdots \text{N} \cdots \text{C} \cdots \text{C} \cdots \text{N} \cdots \text{C} \cdots \text{C} \cdots \text{N} \cdots \text{C} - \text{COOH} \]

- \( b_2 - \text{H}_2\text{O} \)
- \( a_3 - \text{NH}_3 \)
- \( a_2 \)
- \( b_2 \)
- \( b_3 - \text{NH}_3 \)
- \( y_1 \)
- \( y_2 - \text{NH}_3 \)
- \( y_3 - \text{H}_2\text{O} \)
Mass Spectra

- The peaks in the mass spectrum:
  - Prefix and Suffix Fragments.
  - Fragments with neutral losses (-H$_2$O, -NH$_3$)
  - Noise and missing peaks.
Protein Identification with MS/MS

MS/MS

Peptide Identification:

Intensity

mass

0

G V D L K
Tandem Mass-Spectrometry
Breaking Proteins into Peptides

Protein → Trypsin → Peptides

MPSER GTDIMR PAKID → MPSER GTDIMR PAKID

HPLC → To MS/MS
Mass Spectrometry

Matrix-Assisted Laser Desorption/Ionization (MALDI)

Figure 1. The soft laser desorption process.

From lectures by Vineet Bafna (UCSD)
Tandem Mass Spectrometry

**LC**

- Ion Source
- MS-1
- Collision Cell
- MS-2

**MS**

- Scan 1707
- Scan 1708

**MS/MS**
Protein Identification by Tandem Mass Spectrometry

MS/MS instrument

Database search
- Sequest
- de Novo interpretation
- Sherenga
Tandem Mass Spectrum

- Tandem Mass Spectrometry (MS/MS): mainly generates partial N- and C-terminal peptides
- Spectrum consists of different ion types because peptides can be broken in several places.
- Chemical noise often complicates the spectrum.
- Represented in 2-D: mass/charge axis vs. intensity axis
De Novo vs. Database Search

**Database Search**

Database of known peptides

MDERHILNM, KLQWVCSDL, PTYWASDL, ENQIKRSACVM, TLACHGgem, NGALPQWRT, HLLERTKMNVv, GGPassda, GGLITGmqsd, MQPLMNWE, AAKUNMRR, AVGELTK, HEWAILF, GHNWAMNAC, GVFGSLRA, EKLNKAATYIN...

**De Novo**

Database of all peptides = 20^n

DB: AAAAAAAA, AAAAAAAC, AAAAAAAD, AAAAAAAE, AAAAAAAG, AAAAAAAAF, AAAAAAAAH, AAAAAAAAI, AVGELTI, AVGELTK, AVGELTL, AVGELTM, YYYYYYYS, YYYYYYYT, YYYYYYYYV, YYYYYYYY

AVGELTK
De Novo vs. Database Search: A Paradox

• The database of all peptides is huge $\approx O(20^n)$.

• The database of all known peptides is much smaller $\approx O(10^8)$.

• However, *de novo* algorithms can be much faster, even though their search space is much larger!

• A database search scans all peptides in the *database of all known peptides* search space to find best one.

• De novo eliminates the need to scan *database of all peptides* by modeling the problem as a graph search.
De novo Peptide Sequencing
Theoretical Spectrum

N-terminal peptide ladder:
- G-57
- P-97
- F-147
- N-114
- A-71

C-terminal peptide ladder:
- A-71
- N-114
- F-147
- P-97
- G-57
Theoretical Spectrum (cont’d)
Theoretical Spectrum (cont’d)
Building Spectrum Graph

- How to create vertices (from masses)
- How to create edges (from mass differences)
- How to score paths
- How to find best path
<table>
<thead>
<tr>
<th>b</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
</tr>
</tbody>
</table>

<p>| Mass/Charge (M/Z) |</p>
<table>
<thead>
<tr>
<th>S</th>
<th>E</th>
<th>Q</th>
<th>U</th>
<th>E</th>
<th>N</th>
<th>C</th>
<th>E</th>
</tr>
</thead>
</table>

Mass/Charge (M/Z)

mass/charge (m/z)
a is an ion type shift in b
<table>
<thead>
<tr>
<th>Mass/Charge (M/Z)</th>
</tr>
</thead>
<tbody>
<tr>
<td>E</td>
</tr>
<tr>
<td>C</td>
</tr>
<tr>
<td>N</td>
</tr>
<tr>
<td>E</td>
</tr>
<tr>
<td>U</td>
</tr>
<tr>
<td>Q</td>
</tr>
<tr>
<td>S</td>
</tr>
</tbody>
</table>
noise
Some Mass Differences between Peaks Correspond to Amino Acids
Ion Types

- Some masses correspond to fragment ions, others are just random noise.
- Knowing ion types $\Delta = \{\delta_1, \delta_2, \ldots, \delta_k\}$ lets us distinguish fragment ions from noise.
- We can learn ion types $\delta_i$ and their probabilities $q_i$ by analyzing a large test sample of annotated spectra.
Example of Ion Type

- \( \Delta = \{ \delta_1, \delta_2, \ldots, \delta_k \} \)
- Ion types
  \[\{ b, b-{NH}_3, b-{H}_2{O}\}\]
  correspond to
  \[\Delta = \{0, 17, 18\}\]

*Note: In reality the \( \delta \) value of ion type \( b \) is -1 but we will “hide” it for the sake of simplicity.
Match between Spectra and the Shared Peak Count

- The match between two spectra is the number of masses (peaks) they share (Shared Peak Count or SPC)
- In practice mass-spectrometrists use the weighted SPC that reflects intensities of the peaks
- Match between experimental and theoretical spectra is defined similarly
Peptide Sequencing Problem

Goal: Find a peptide with maximal match between an experimental and theoretical spectrum.

Input:
- $S$: experimental spectrum
- $\Delta$: set of possible ion types
- $m$: parent mass

Output:
- $P$: peptide with mass $m$, whose theoretical spectrum matches the experimental $S$ spectrum the best
Vertices of Spectrum Graph

- Masses of potential N-terminal peptides
- Vertices are generated by reverse shifts corresponding to ion types
  \[ \Delta = \{ \delta_1, \delta_2, \ldots, \delta_k \} \]
- Every N-terminal peptide can generate up to \( k \) ions
  \[ m-\delta_1, m-\delta_2, \ldots, m-\delta_k \]
- Every mass \( s \) in an MS/MS spectrum generates \( k \) vertices
  \[ V(s) = \{ s+\delta_1, s+\delta_2, \ldots, s+\delta_k \} \]
corresponding to potential N-terminal peptides

- **Vertices of the spectrum graph:**
  \[ \{ \text{initial vertex} \} \cup V(s_1) \cup V(s_2) \cup \ldots \cup V(s_m) \cup \{ \text{terminal vertex} \} \]
Reverse Shifts

Shift in H$_2$O

Shift in H$_2$O+NH$_3$
Edges of Spectrum Graph

• Two vertices with mass difference corresponding to an amino acid $A$:
  • Connect with an edge labeled by $A$
• Gap edges for di- and tri-peptides
Paths

- Path in the labeled graph spell out amino acid sequences
- There are many paths, how to find the correct one?
- We need **scoring** to evaluate paths
Path Score

- \( p(P, S) = \) probability that peptide \( P \) produces spectrum \( S = \{s_1, s_2, \ldots, s_q\} \)

- \( p(P, s) = \) the probability that peptide \( P \) generates a peak \( s \)

- **Scoring = computing probabilities**

- \( p(P, S) = \prod_{s \in S} p(P, s) \)
Peak Score

• For a position $t$ that represents ion type $d_j$:

$$p(P, s_t) = \begin{cases} 
q_j, & \text{if peak is generated at } t \\
1 - q_j, & \text{otherwise}
\end{cases}$$
Peak Score (cont’d)

• For a position $t$ that is not associated with an ion type:

\[
p_R(P,s_t) = \begin{cases} 
q_R, & \text{if peak is generated at } t \\
1-q_R, & \text{otherwise}
\end{cases}
\]

• $q_R = \text{the probability of a noisy peak that does not correspond to any ion type}$
Finding Optimal Paths in the Spectrum Graph

• For a given MS/MS spectrum $S$, find a peptide $P'$ maximizing $p(P,S)$ over all possible peptides $P$:

$$p(P',S) = \max_P p(P,S)$$

• Peptides = paths in the spectrum graph

• $P'$ = the optimal path in the spectrum graph
Ions and Probabilities

- Tandem mass spectrometry is characterized by a set of ion types \( \{\delta_1, \delta_2, \ldots, \delta_k\} \) and their probabilities \( \{q_1, \ldots, q_k\} \)

- \( \delta_i \)-ions of a partial peptide are produced independently with probabilities \( q_i \)
Ions and Probabilities

- A peptide has all \( k \) peaks with probability \( \prod_{i=1}^{k} q_i \)
- and no peaks with probability \( \prod_{i=1}^{k} (1 - q_i) \)
- A peptide also produces a "random noise" with \textit{uniform} probability \( q_R \) in any position.
Ratio Test Scoring for Partial Peptides

• Incorporates **premiums** for observed ions and **penalties** for missing ions.

• Example: for \( k=4 \), assume that for a partial peptide \( P' \) we only see ions \( \delta_1, \delta_2, \delta_4 \).

The score is calculated as:

\[
\frac{q_1}{q_R} \cdot \frac{q_2}{q_R} \cdot \frac{(1-q_3)}{(1-q_R)} \cdot \frac{q_4}{q_R}
\]
Scoring Peptides

- $T$: set of all positions.

- $T_i = \{t_{\delta_1}, t_{\delta_2}, \ldots, t_{\delta_k}\}$: set of positions that represent ions of partial peptides $P_i$.

- A peak at position $t_{\delta_j}$ is generated with probability $q_j$.

- $R = T - U T_i$: set of positions that are not associated with any partial peptides (noise).
Probabilistic Model

- For a position $t_{\delta_j} \in T_i$ the probability $p(t, P, S)$ that peptide $P$ produces a peak at position $t$.

  $$P(t, P, S) = \begin{cases} 
  q_j & \text{if a peak is generated at position } t_{\delta_j} \\
  1 - q_j & \text{otherwise}
  \end{cases}$$

- Similarly, for $t \in R$, the probability that $P$ produces a random noise peak at $t$ is:

  $$P_R(t) = \begin{cases} 
  q_R & \text{if a peak is generated at position } t \\
  1 - q_R & \text{otherwise}
  \end{cases}$$
Probabilistic Score

• For a peptide $P$ with $n$ amino acids, the score for the whole peptides is expressed by the following ratio test:

$$\frac{p(P, S)}{p_R(S)} = \prod_{i=1}^{n} \prod_{j=1}^{k} \frac{p(t_i \delta_j, P, S)}{p_R(t_i \delta_j)}$$
De-novo sequencing: how?

How can we find the amino acid sequence that best explains the spectrum?

a) Explains the largest number of peaks
b) Explains the most intensity

• Exhaustive enumeration?
  1. Generate every possible sequence with the same peptide mass
  2. Match each sequence to the spectrum
  3. Choose the sequence that explains the most intensity in the spectrum

Takes too long!
Initial attempts

Computer programs for \textit{de-novo} interpretation of MS/MS spectra date as far back as 1966 when Biemman et al. proposed a prefix extension algorithm:
- Tries every possible prefix extension
- Eliminates solutions with missing peaks
- Outputs every peptide with a matching parent mass
- ALL prefix peaks must be present in the spectrum

In an attempt to include interpretations with missing peaks, Sakurai et al. (1984) proposed:
- Exhaustive search over the space of all possible permutations of amino acid multisets where the total mass equals the parent mass of the MS/MS spectrum
- Score peptide/spectrum matches by counting prefix \textit{and} suffix mass matches
- Naturally more sensitive but also \textit{very} slow
Prefix extension revisited

The second half of the eighties saw a few better designed approaches to this problem, based on the same type of algorithm:

- Prefix extension, one amino acid at a time.
- Tolerate missing peaks.
- Include both prefix and suffix peaks in the score.
- User-specified maximum number of candidates in memory at any point of the execution. (sub-optimal)

Ishikawa and Niwa’86, Siegel and Baumann’88, Johnson and Biemann’89, Zidarov et al.’90
Spectrum graphs

In 1990 Bartels introduced a graph representation of an MS/MS spectrum
- Every peak in the spectrum defines a vertex
- Vertices connected by an edge if peak mass difference is an amino acid mass

- Best peptide is defined as the best path between the two endpoint vertices: $v_0$ to $v_{M(S)}$
- No detailed algorithm was given for finding the best peptide; interactive exploration tool was made available.
What is the DP recursion?
Score(i) = intensity(i) + max( Score(j) ),
for all j with mass(i)-mass(j) ∈ Amino acid masses

Recovered de-novo sequence? ESESE
Mass array sequencing

• Note that spectral graph approaches use local peak mass tolerances resulting in cumulative peptide mass errors. Could be off by as much as 6 Da in a 12aa peptide (0.5 Da per aa)!

• Alternative approach to sequencing
  • Represent the spectrum as an array of 0.1Da bins.
  • The intensity in a bin B is the sum of the peak intensities for all peaks with rounded masses equal to B.
  • What is the recursion?
  • How accurate is the parent mass of the recovered peptide?
  • How can we generate all suboptimal solutions with a score higher than a chosen threshold?
DP de-novo sequencing: EDTES

DP recursion:
Score(i) = intensity(i) + max( Score(j) ),
for all j with mass(i) - mass(j) ∈ Amino acid masses

Recovered de-novo sequence? ESESE

Why was the correct peptide missed?
Forbidden pairs

The exclusion of symmetric peaks in a maximal scoring path through a spectrum graph was first proposed by Dančík et al. in 1999:

- Peaks in the spectrum are called forbidden pairs if their mass adds up to the parent mass – either vertex can be used in the output path but not both.
- A path is anti-symmetric if it uses at most one vertex from every forbidden pair.
- Objective function becomes: find maximal scoring anti-symmetric path.
- NP-Hard in the general case

Solution:
1. Jump from the mass closest to the start/end of the spectrum
2. Avoid reusing the pairing mass (highlighted in red)

Note that this ordering is always possible (Chen’01, Bafna and Edwards’03)

What is the recursion?
Maximal scoring anti-symmetric path

Chen et al’01 provided a dynamic programming recursion to find a maximal scoring anti-symmetric path:

- A peak $s_i$ *precedes* a peak $s_j$ if it is closer to one of the ends of the spectrum: $\min(s_i,m(S)-s_i)<\min(s_j,m(S)-s_j)$

- Let $Sc[i,j]$ be the score of the maximal scoring anti-symmetric path from $v_0$ to $v_i$ and from $v_j$ to $v_{m(S)}$, including $v_i$ and $v_j$ (all initialized to $-\infty$)

- Then from $Sc[i,j] =$
  - If $s_j$ precedes $s_k$, $s_k-s_i$ is an amino acid mass and $v_k$ and $v_j$ are not a forbidden pair
    - $Sc[k,j] = \max(Sc[k,j], \text{score}(k)+Sc[i,j])$  (prefix extension)
  - If $s_i$ precedes $s_k$, $s_j-s_k$ is an amino acid mass and $v_i$ and $v_k$ are not a forbidden pair
    - $Sc[i,k] = \max(Sc[i,k], \text{score}(k)+Sc[i,j])$  (suffix extension)
  - If $s_j-s_i$ is an amino acid mass and $v_i$ and $v_j$ are not a forbidden pair
    - Mark as possible solution  (prefix/suffix connection)
Optimality and extensions

The anti-symmetric dynamic programming solutions are

- Correct: no symmetric peak can be reused and every anti-symmetric path is considered
- Optimal: a maximal scoring anti-symmetric path is constructed
- “Efficient”: runtime efficiency is $O(n^2)$, $n=$# peaks

Other algorithms have also been proposed:

- Ma’03, Frank’05, same algorithm, different scoring
- Bafna and Edwards’03, same principle, extended the concept of forbidden pairs to avoid peak reusage between any pair of ion-types
De Novo vs. Database Search

Database Search

Database of known peptides

MDERHILNM, KLQVVCSDL, PTYWASDL, ENQIKRSACVM, TLACHGGEM, NGALPQWRT, HLLERTKMNVV, GGPASSDA, GGLITGMQSD, MQPLMNWE, ALKIIMNVRT, AVGELTK, HEWAILF, GHNLWAMNAC, GVFGSVLRA, EKLNKAATYIN...

De Novo

AVGELTK
Peptide Sequencing Problem

**Goal**: Find a peptide with maximal match between an experimental and theoretical spectrum.

**Input**:
- $S$: experimental spectrum
- $\Delta$: set of possible ion types
- $m$: parent mass

**Output**:
- A peptide with mass $m$, whose theoretical spectrum matches the experimental $S$ spectrum the best
Peptide Identification Problem

Goal: Find a peptide *from the database* with maximal match between an experimental and theoretical spectrum.

Input:
- $S$: experimental spectrum
- *database of peptides*
- $\Delta$: set of possible ion types
- $m$: parent mass

Output:
- A peptide of mass $m$ *from the database* whose theoretical spectrum matches the experimental $S$ spectrum the best
MS/MS Database Search

Database search in mass-spectrometry has been very successful in identification of already known proteins.

Experimental spectrum can be compared with theoretical spectra of database peptides to find the best fit.

SEQUEST (Yates et al., 1995)

But reliable algorithms for identification of modified peptides is a much more difficult problem.
The proteome of the cell is changing.
Various extra-cellular, and other signals activate pathways of proteins.
A key mechanism of protein activation is **post-translational modification (PTM)**.
These pathways may lead to other genes being switched on or off.
Mass spectrometry is key to probing the proteome and detecting PTMs.
Post-Translational Modifications

Proteins are involved in cellular signaling and metabolic regulation.

They are subject to a large number of biological modifications.

Almost all protein sequences are post-translationally modified and **200 types of modifications** of amino acid residues are known.
Post-translational modifications increase the number of “letters” in amino acid alphabet and lead to a combinatorial explosion in both database search and de novo approaches.
Search for Modified Peptides: Virtual Database Approach

Yates et al., 1995: an exhaustive search in a virtual database of all modified peptides.

Exhaustive search leads to a large combinatorial problem, even for a small set of modifications types.

Problem (Yates et al., 1995). Extend the virtual database approach to a large set of modifications.
Exhaustive Search for modified peptides.

- For each peptide, generate all modifications.
- Score each modification.

\[ 2^5 = 32 \text{ possibilities, with 2 types of modifications!} \]
Peptide Identification Problem Revisited

**Goal**: Find a peptide from the database with maximal match between an experimental and theoretical spectrum.

**Input**:
- $S$: experimental spectrum
- database of peptides
- $\Delta$: set of possible ion types
- $m$: parent mass

**Output**:
- A peptide of mass $m$ from the database whose theoretical spectrum matches the experimental $S$ spectrum the best
Modified Peptide Identification Problem

**Goal**: Find a modified peptide from the database with maximal match between an experimental and theoretical spectrum.

**Input**:
- $S$: experimental spectrum
- database of peptides
- $\Delta$: set of possible ion types
- $m$: parent mass
- Parameter $k$ (# of mutations/modifications)

**Output**:
- A peptide of mass $m$ that is at most $k$ mutations/modifications apart from a database peptide and whose theoretical spectrum matches the experimental $S$ spectrum the best
Database Search:
Sequence Analysis vs. MS/MS Analysis

Sequence analysis:

similar peptides (that a few mutations apart) have similar sequences

MS/MS analysis:

similar peptides (that a few mutations apart) have dissimilar spectra
Peptide Identification Problem: Challenge

Very similar peptides may have very different spectra!

**Goal:** Define a notion of spectral similarity that correlates well with the sequence similarity.

If peptides are a few mutations/modifications apart, the spectral similarity between their spectra should be high.
Deficiency of the Shared Peaks Count

**Shared peaks count (SPC):** intuitive measure of spectral similarity.

**Problem:** SPC diminishes very quickly as the number of mutations increases.

Only a small portion of correlations between the spectra of mutated peptides is captured by SPC.
SPC Diminishes Quickly

\[ S(\text{PRTEIN}) = \{98, 133, 246, 254, 355, 375, 476, 484, 597, 632\} \]

\[ S(\text{PRTEYN}) = \{98, 133, 254, 296, 355, 425, 484, 526, 647, 682\} \]

\[ S(\text{PGTEYN}) = \{98, 133, 155, 256, 296, 385, 425, 526, 548, 583\} \]
Spectral Convolution

\[ S_2 \ominus S_1 = \{ s_2 - s_1 : s_1 \in S_1, s_2 \in S_2 \} \]

Number of pairs \( s_1 \in S_1, s_2 \in S_2 \) with \( s_2 - s_1 = x \):

\[ (S_2 \ominus S_1)(x) \]

The shared peaks count (SPC peak):

\[ (S_2 \ominus S_1)(0) \]
Elements of $S_2 \otimes S_1$ represented as elements of a **difference matrix**. The elements with multiplicity >2 are colored; the elements with multiplicity =2 are circled. The SPC takes into account only the red entries.
Spectral Convolution: An Example

![Spectral Convolution Graph]

<table>
<thead>
<tr>
<th>x</th>
<th>150</th>
<th>-100</th>
<th>-50</th>
<th>0</th>
<th>50</th>
<th>100</th>
</tr>
</thead>
<tbody>
<tr>
<td>y</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Protein**

**Protein**
Spectral Comparison: Difficult Case

\[ S = \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\} \]

Which of the spectra
\[ S' = \{10, 20, 30, 40, 50, 55, 65, 75, 85, 95\} \]
or
\[ S'' = \{10, 15, 30, 35, 50, 55, 70, 75, 90, 95\} \]
fits the spectrum \( S \) the best?

SPC: both \( S' \) and \( S'' \) have 5 peaks in common with \( S \). Spectral Convolution: reveals the peaks at 0 and 5.
### Spectral Comparison: Difficult Case

<table>
<thead>
<tr>
<th></th>
<th>S₀</th>
<th>S₁</th>
<th>S₂</th>
<th>S₃</th>
<th>S₄</th>
<th>S₅</th>
<th>S₆</th>
<th>S₇</th>
<th>S₈</th>
<th>S₉</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td>-70</td>
<td>-80</td>
<td>-90</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td>-70</td>
<td>-80</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td>-70</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>20</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>30</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td>-35</td>
<td>-45</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td>-35</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>55</td>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td></td>
</tr>
<tr>
<td>75</td>
<td>65</td>
<td>55</td>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>75</td>
<td>65</td>
<td>55</td>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>S₀</th>
<th>S₁</th>
<th>S₂</th>
<th>S₃</th>
<th>S₄</th>
<th>S₅</th>
<th>S₆</th>
<th>S₇</th>
<th>S₈</th>
<th>S₉</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td>-70</td>
<td>-80</td>
<td>-90</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td>-35</td>
<td>-45</td>
<td>-55</td>
<td>-65</td>
<td>-75</td>
<td>-85</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td>-60</td>
<td>-70</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td>-35</td>
<td>-45</td>
<td>-55</td>
<td>-65</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>30</td>
<td>20</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td>-40</td>
<td>-50</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td>-35</td>
<td>-45</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>50</td>
<td>40</td>
<td>30</td>
<td>20</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td>-20</td>
<td>-30</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>55</td>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td>-15</td>
<td>-25</td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>70</td>
<td>60</td>
<td>50</td>
<td>40</td>
<td>30</td>
<td>20</td>
<td>10</td>
<td>0</td>
<td>-10</td>
<td></td>
</tr>
<tr>
<td>85</td>
<td>75</td>
<td>65</td>
<td>55</td>
<td>45</td>
<td>35</td>
<td>25</td>
<td>15</td>
<td>5</td>
<td>-5</td>
<td></td>
</tr>
</tbody>
</table>
Limitations of the Spectrum Convolutions

Spectral convolution does not reveal that spectra $S$ and $S'$ are similar, while spectra $S$ and $S''$ are not.

**Clumps of shared peaks:** the matching positions in $S'$ come in clumps while the matching positions in $S''$ don't.

This important property was not captured by spectral convolution.
Shifts

$A = \{a_1 < ... < a_n\}$: an ordered set of natural numbers.

A shift $(i, \Delta)$ is characterized by two parameters, the position $(i)$ and the length $(\Delta)$.

The shift $(i, \Delta)$ transforms

$\{a_1, \ldots, a_n\}$

into

$\{a_1, \ldots, a_{i-1}, a_i + \Delta, \ldots, a_n + \Delta\}$
Shifts: An Example

The shift \((i, \Delta)\) transforms \(\{a_1, \ldots, a_n\}\) into \(\{a_1, \ldots, a_{i-1}, a_i + \Delta, \ldots, a_n + \Delta\}\)

\[\text{e.g.}\]

\[
\begin{array}{cccccccccc}
10 & 20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 \\
\end{array}
\]

\[
\begin{array}{cccccccccc}
10 & 20 & 30 & 35 & 45 & 55 & 65 & 75 & 85 \\
\end{array}
\]

\[
\begin{array}{cccccccccc}
10 & 20 & 30 & 35 & 45 & 55 & 62 & 72 & 82 \\
\end{array}
\]

shift \((4, -5)\)

shift \((7, -3)\)
Spectral Alignment Problem

• Find a series of $k$ shifts that make the sets $A=\{a_1, \ldots, a_n\}$ and $B=\{b_1, \ldots, b_n\}$ as similar as possible.

• $k$-similarity between sets

• $D(k)$ - the maximum number of elements in common between sets after $k$ shifts.
Representing Spectra in 0-1 Alphabet

- Convert spectrum to a 0-1 string with 1s corresponding to the positions of the peaks.
Comparing Spectra=Comparing 0-1 Strings

- A modification with positive offset corresponds to inserting a block of 0s
- A modification with negative offset corresponds to deleting a block of 0s
- Comparison of theoretical and experimental spectra (represented as 0-1 strings) corresponds to a (somewhat unusual) edit distance/alignment problem where elementary edit operations are insertions/deletions of blocks of 0s
- Use sequence alignment algorithms!
Spectral Alignment vs. Sequence Alignment

- Manhattan-like graph with different alphabet and scoring.
- Movement can be diagonal (matching masses) or horizontal/vertical (insertions/deletions corresponding to PTMs).
- At most $k$ horizontal/vertical moves.
Spectral Product

\[ A=\{a_1, \ldots, a_n\} \text{ and } B=\{b_1, \ldots, b_n\} \]

*Spectral product* \( A \otimes B \): two-dimensional matrix with \( nm \) 1s corresponding to all pairs of indices \((a_i, b_j)\) and remaining elements being 0s.

SPC: the number of 1s at the main diagonal.

\( \delta \)-shifted SPC: the number of 1s on the diagonal \((i, i+ \delta)\).
Spectral Alignment: \( k \)-similarity

\( k \)-similarity between spectra: the maximum number of 1s on a path through this graph that uses at most \( k+1 \) diagonals.

\( k \)-optimal spectral alignment = a path.

The spectral alignment allows one to detect more and more subtle similarities between spectra by increasing \( k \).
SPC reveals only $D(0)=3$ matching peaks.

Spectral Alignment reveals more hidden similarities between spectra: $D(1)=5$ and $D(2)=8$ and detects corresponding mutations.
**Black line** represent the path for $k=0$

**Red lines** represent the path for $k=1$

**Blue lines** (right) represents the path for $k=2$
Spectral Convolution’ Limitation

The spectral convolution considers diagonals separately without combining them into feasible mutation scenarios.

\[ D(1) = 10 \quad \text{shift function score} = 10 \quad D(1) = 6 \]
Dynamic Programming for Spectral Alignment

\( D_{ij}(k) \): the maximum number of 1s on a path to \((a_i, b_j)\) that uses at most \(k+1\) diagonals.

\[
D_{ij}(k) = \max_{(i', j') < (i, j)} \begin{cases} 
D_{i' j'}(k) + 1, & \text{if } (i', j') \sim (i, j) \\
D_{i' j'}(k-1) + 1, & \text{otherwise}
\end{cases}
\]

\[
D(k) = \max_{ij} D_{ij}(k)
\]

Running time: \(O(n^4 k)\)
Edit Graph for Fast Spectral Alignment

$\text{diag}(i,j)$ – the position of previous 1 on the same diagonal as $(i,j)$
Fast Spectral Alignment Algorithm

\[
M_{ij}(k) = \max_{(i', j') < (i, j)} D_{i', j'}(k)
\]

\[
D_{ij}(k) = \max \begin{cases} 
D_{\text{diag}(i,j)}(k) + 1 \\
M_{i-1, j-1}(k - 1) + 1 \\
D_{ij}(k)
\end{cases}
\]

\[
M_{ij}(k) = \max \begin{cases} 
D_{ij}(k) \\
M_{i-1, j}(k) \\
M_{i, j-1}(k)
\end{cases}
\]

Running time: \(O(n^2 k)\)
Spectral Alignment: Complications

Spectra are combinations of an increasing (N-terminal ions) and a decreasing (C-terminal ions) number series.

These series form two diagonals in the spectral product, the main diagonal and a complementary diagonal.

The described algorithm deals with the main diagonal only.
Spectral Alignment: Complications

- Simultaneous analysis of N- and C-terminal ions
- Taking into account the intensities and charges
- Analysis of minor ions
Filtration: Combining de novo and Database Search in Mass-Spectrometry

- So far de novo and database search were presented as two separate techniques

- **Database search is rather slow:** many labs generate more than 100,000 spectra per day. SEQUEST takes approximately 1 minute to compare a single spectrum against SWISS-PROT (54Mb) on a desktop.

- It will take SEQUEST more than 2 months to analyze the MS/MS data produced in a single day.

- **Can slow database search be combined with fast de novo analysis?**
Why Filtration?

Sequence Alignment – BLAST

- BLAST filters out very few correct matches and is almost as accurate as Smith – Waterman algorithm.
Filtration and MS/MS

Peptide Sequencing – SEQUEST / Mascot

MS/MS spectrum

Filtration

Database

MDERHILNMKIQWVCSDLPT
YWASDLENIKRSACVMILA
CHGEMNGALPQWRTHLLE
RTYMNVVGGPASSDALITG
MQSDPVLLVCATRGHEWAILF
GHNLWACVMLETAIKLEGV
FGSVLRSAELNKAAPETYIN..
Filtration in MS/MS Sequencing

- Filtration in MS/MS is more difficult than in BLAST.

- Early approaches using Peptide Sequence Tags were not able to substitute the complete database search.

- Current filtration approaches are mostly used to generate additional identifications rather than replace the database search.

- Can we design a filtration based search that can replace the database search, and is orders of magnitude faster?
Asking the Old Question Again: Why Not Sequence De Novo?

- **De novo** sequencing is still not very accurate!

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Amino Acid Accuracy</th>
<th>Whole Peptide Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lutefisk (Taylor and Johnson, 1997).</td>
<td>0.566</td>
<td>0.189</td>
</tr>
<tr>
<td>SHERENGA (Dancik et. al., 1999).</td>
<td>0.690</td>
<td>0.289</td>
</tr>
<tr>
<td>Peaks (Ma et al., 2003).</td>
<td>0.673</td>
<td>0.246</td>
</tr>
<tr>
<td><strong>PepNovo</strong> (Frank and Pevzner, 2005).</td>
<td><strong>0.727</strong></td>
<td><strong>0.296</strong></td>
</tr>
</tbody>
</table>
So What Can be Done with De Novo?

• Given an MS/MS spectrum:
  • Can de novo predict the entire peptide sequence? - No! (accuracy is less than 30%).
  • Can de novo predict partial sequences? - No! (accuracy is 50% for GutenTag and 80% for PepNovo)
  • Can de novo predict a set of partial sequences, that with high probability, contains at least one correct tag? - Yes!

A Covering Set of Tags
A Peptide Sequence Tag is short substring of a peptide.

Example:  
\[
\begin{align*}
G & \ V & \ D & \ L & \ K \\
G & \ V & \ D \\
V & \ D & \ L \\
D & \ L & \ K
\end{align*}
\]

Tags:  
\[
\left\{
\begin{align*}
G & \ V & \ D & \ L & \ K \\
G & \ V & \ D \\
V & \ D & \ L \\
D & \ L & \ K
\end{align*}
\right\}
\]
Filtration with Peptide Sequence Tags

- Peptide sequence tags can be used as filters in database searches.

- **The Filtration:** Consider only database peptides that contain the tag (in its correct relative mass location).

- First suggested by Mann and Wilm (1994).

- Similar concepts also used by:
  - MultiTag - Sunayev et. al. 2003.
Why Filter Database Candidates?

- Filtration makes genomic database searches practical (BLAST).

- Effective filtration can greatly speed-up the process, enabling expensive searches involving post-translational modifications.

- Goal: *generate a small set of covering tags and use them to filter the database peptides.*
Tag Generation - Global Tags

- Parse tags from *de novo* reconstruction.
- Only a small number of tags can be generated.
- If the *de novo* sequence is completely incorrect, none of the tags will be correct.

<table>
<thead>
<tr>
<th>TAG</th>
<th>Prefix Mass</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>0.0</td>
</tr>
<tr>
<td>VGE</td>
<td>71.0</td>
</tr>
<tr>
<td>GEL</td>
<td>170.1</td>
</tr>
<tr>
<td>ELT</td>
<td>227.1</td>
</tr>
<tr>
<td>LTK</td>
<td>356.2</td>
</tr>
</tbody>
</table>
Tag Generation - Local Tags

- Extract the highest scoring subpaths from the spectrum graph.
- Sometimes gets misled by locally promising-looking “garden paths”.

<table>
<thead>
<tr>
<th>TAG</th>
<th>Prefix Mass</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVG</td>
<td>0.0</td>
</tr>
<tr>
<td>WTD</td>
<td>120.2</td>
</tr>
<tr>
<td>PET</td>
<td>211.4</td>
</tr>
</tbody>
</table>
Ranking Tags

- Each additional tag used to filter increases the number of database hits and slows down the database search.

- Tags can be ranked according to their scores, however this ranking is not very accurate.

- It is better to determine for each tag the “probability” that it is correct, and choose most probable tags.
Reliability of Amino Acids in Tags

- For each amino acid in a tag we want to assign a probability that it is correct.

- Each amino acid, which corresponds to an edge in the spectrum graph, is mapped to a feature space that consists of the features that correlate with reliability of amino acid prediction, e.g. score reduction due to edge removal.
The removal of an edge corresponding to a genuine amino acid usually leads to a reduction in the score of the *de novo* path.

However, the removal of an edge that *does not* correspond to a genuine amino acid tends to leave the score unchanged.
Probabilities of Tags

- How do we determine the probability of a predicted tag?

- We use the predicted probabilities of its amino acids and follow the concept:

  *a chain is only as strong as its weakest link*
## Experimental Results

<table>
<thead>
<tr>
<th>Algorithm \ #tags</th>
<th>Length 3</th>
<th>Length 4</th>
<th>Length 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>GlobalTag</td>
<td>0.80</td>
<td>0.73</td>
<td>0.66</td>
</tr>
<tr>
<td>LocalTag+</td>
<td>0.75</td>
<td>0.70</td>
<td>0.57</td>
</tr>
<tr>
<td>GutenTag</td>
<td>0.49</td>
<td>0.41</td>
<td>0.31</td>
</tr>
</tbody>
</table>

- Results are for 280 spectra of doubly charged tryptic peptides from the ISB and OPD datasets.
Tag-based Database Search

Db 55M peptides

Tag filter → Tag extension → Score → Significance

Candidate Peptides (700)

De novo
Matching Multiple Tags

• Matching of a sequence tag against a database is fast
• Even matching many tags against a database is fast
• $k$ tags can be matched against a database in time proportional to database size, but independent of the number of tags.
  • *keyword trees* (Aho-Corasick algorithm)
• Scan time can be amortized by combining scans for many spectra all at once.
  • *build one keyword tree from multiple spectra*
Keyword Trees

.....Y F R A Y F N T A.....
Tag Extension

Filter → Extension → Score → Significance

Db 55M peptides

De novo

Candidate Peptides (700)
Fast Extension

• Given:
  • tag with prefix and suffix masses \(<m_p> xyz <m_s>\)
  • match in the database

\[
\begin{array}{c}
\langle m_p \rangle xyz \langle m_s \rangle \\
\hline
xyz & xyz
\end{array}
\]

• Compute if a suffix and prefix match with allowable modifications.
• Compute a candidate peptide with most likely positions of modifications (attachment points).
Scoring Modified Peptides

Db 55M peptides

Filter → Extension → Score → Significance

De novo
Scoring

• **Input:**
  - Candidate peptide with attached modifications
  - Spectrum

• **Output:**
  - Score function that normalizes for length, as variable modifications can change peptide length.
Assessing Reliability of Identifications

Db 55M peptides

Filter → extension → Score → Significance

De novo
Selecting Features for Separating Correct and Incorrect Predictions

• Features:
  • Score $S$: as computed
  • Explained Intensity $I$: fraction of total intensity explained by annotated peaks.
  • b-y score $B$: fraction of b+y ions annotated
  • Explained peaks $P$: fraction of top 25 peaks annotated.
  • Each of $I, S, B, P$ features is normalized (subtract mean and divide by s.d.)

• Problem: separate correct and incorrect identifications using $I, S, B, P$
Separating power of features
Separating power of features

Quality scores:
\[ Q = w_I I + w_S S + w_B B + w_P P \]

The weights are chosen to minimize the mis-classification error.
Distribution of Quality Scores
Results on ISB data-set

- All ISB spectra were searched.
  - The top match is valid for 2978 spectra (2765 for Sequest)
  - InsPecT-Sequest: 644 spectra (I-S dataset)
  - Sequest-InsPecT: 422 spectra (S-I dataset)
  - Average explained intensity of I-S = 52%
  - Average explained intensity of S-I = 28%
  - Average explained intensity I ∩ S = 58%
  - ~70 Met. Oxidations
  - Run time is 0.7 secs. per spectrum (2.7 secs. for Sequest)
Results for Mus-IMAC data-sets

<table>
<thead>
<tr>
<th>Novel?</th>
<th>Occurrence</th>
<th>p-value</th>
<th>Peptide</th>
<th>Protein (GI number)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>8</td>
<td>0.0022</td>
<td>PAAPAAPAPVEKTpPVKK</td>
<td>Histone H1d (gi</td>
</tr>
<tr>
<td>Yes</td>
<td>1</td>
<td>0.003</td>
<td>RPSpVYLPTR</td>
<td>150003401Rik protein (gi</td>
</tr>
<tr>
<td>1</td>
<td>0.0031</td>
<td></td>
<td>MASINFIPpPEENPPSWAK</td>
<td>(gi</td>
</tr>
<tr>
<td>2</td>
<td>0.0012</td>
<td></td>
<td>DKLHPSpPEEKK</td>
<td>40S ribosomal protein S20 (gi</td>
</tr>
<tr>
<td>36</td>
<td>0.0057</td>
<td></td>
<td>DTGKIpPVEPEVAIHR</td>
<td>40S ribosomal protein S20 (gi</td>
</tr>
<tr>
<td>Yes</td>
<td>3</td>
<td>0.007</td>
<td>KTEAYpLEAIR</td>
<td>Secretogranin 3 (gi</td>
</tr>
<tr>
<td>1</td>
<td>0.007</td>
<td></td>
<td>APTSTDlpPIYSQVAPR</td>
<td>Protein tyrosine phosphatase, non-receptor type 1B (gi</td>
</tr>
<tr>
<td>1</td>
<td>0.007</td>
<td></td>
<td>ASGQAFELILspPR</td>
<td>Stathmin 1 (gi</td>
</tr>
<tr>
<td>5</td>
<td>0.007</td>
<td></td>
<td>SKEvsPDPISpPPK</td>
<td>Stathmin 1 (gi</td>
</tr>
<tr>
<td>Yes</td>
<td>1</td>
<td>0.0167</td>
<td>PAAPAAPAPAETpPVKK</td>
<td>Histone H1e (gi</td>
</tr>
<tr>
<td>Yes</td>
<td>7</td>
<td>0.02</td>
<td>PNQQAVLPKKTESpHHK</td>
<td>Histone H2a(gi</td>
</tr>
<tr>
<td>1</td>
<td>0.026</td>
<td></td>
<td>SQETYETpLKHEKPPQ</td>
<td>Fc receptor, IgE (gi</td>
</tr>
</tbody>
</table>

Table 3: Phosphoepitopes identified on the Tripure dataset.

- The Alliance for Cellular signalling is looking at proteins phosphorylated in specific signal transduction pathways.
- 6500 spectra are searched with up to 4 modifications (up to 3 Met. Oxidation and up to 2 Phos.)
- 281 phosphopeptides with P-value < 0.05
The search was done against SWISS-PROT (54Mb).

- With 10 tags of length 3:
  - The filtration is 1500 more efficient.
  - Less than 4% of spectra are filtered out.
  - The search time per spectrum is reduced by two orders of magnitude as compared to SEQUEST.
Conclusion

• With 10 tags of length 3:
  • The filtration is 1500 more efficient than using only the parent mass alone.
  • Less than 4% of the positive peptides are filtered out.
  • The search time per spectrum is reduced from over a minute (SEQUEST) to 0.4 seconds.
SPIDER: Yet Another Application of \textit{de novo} Sequencing

- Suppose you have a good MS/MS spectrum of an elephant peptide
- Suppose you even have a good \textit{de novo} reconstruction of this spectra
- However, until elephant genome is sequenced, it is hard to verify this \textit{de novo} reconstruction
- Can you search \textit{de novo} reconstruction of a peptide from elephant against human protein database?
- SPIDER (Han, Ma, Zhang) addresses this \textit{comparative proteomics} problem
Common de novo sequencing errors

N and GG have the same mass
From *de novo* Reconstruction to Database Candidate through Real Sequence

- Given a sequence with errors, search for the similar sequences in a DB.

\[
\begin{array}{c}
\text{Seq) } X: \text{ LSCFAV} \\
\text{Real) } Y: \text{ SLCFAV} \\
\text{Match) } Z: \text{ SLCF-V}
\end{array}
\]

sequencing error

\[
\begin{array}{c}
\text{Seq) } X: \text{ LSCF-AV} \\
\text{Real) } Y: \text{ EACF-AV} \\
\text{Match) } Z: \text{ DACFKAV}
\end{array}
\]

Homology mutations

mass(LS)=mass(EA)
Alignment between de novo Candidate and Database Candidate

- If real sequence $Y$ is known then:

$$d(X,Z) = \text{seqError}(X,Y) + \text{editDist}(Y,Z)$$
Alignment between de novo Candidate and Database Candidate

- If real sequence $Y$ is known then:
  \[ d(X,Z) = \text{seqError}(X,Y) + \text{editDist}(Y,Z) \]
- If real sequence $Y$ is unknown then the distance between de novo candidate $X$ and database candidate $Z$:
  - $d(X,Z) = \min_Y ( \text{seqError}(X,Y) + \text{editDist}(Y,Z) )$
Alignment between de novo Candidate and Database Candidate

- If real sequence $Y$ is known then:
  $$d(X,Z) = \text{seqError}(X,Y) + \text{editDist}(Y,Z)$$

- If real sequence $Y$ is unknown then the distance between de novo candidate $X$ and database candidate $Z$:
  $$d(X,Z) = \min_Y ( \text{seqError}(X,Y) + \text{editDist}(Y,Z) )$$

- **Problem**: search a database for $Z$ that minimizes $d(X,Z)$
- The core problem is to compute $d(X,Z)$ for given $X$ and $Z$. 

<table>
<thead>
<tr>
<th>Seq</th>
<th>X: LS CF-AV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real</td>
<td>Y: EACF-AV</td>
</tr>
<tr>
<td>Match</td>
<td>Z: DACF AV</td>
</tr>
</tbody>
</table>
Computing $\text{seqError}(X, Y)$

- Align $X$ and $Y$ (according to mass).

- A segment of $X$ can be aligned to a segment of $Y$ only if their mass is the same!

- For each erroneous mass block $(X_i, Y_i)$, the cost is $f(X_i, Y_i) = f(\text{mass}(X_i))$.
  - $f(m)$ depends on how often de novo sequencing makes errors on a segment with mass $m$.
  - $\text{seqError}(X, Y)$ is the sum of all $f(\text{mass}(X_i))$. 
Computing $d(X,Z)$

- Dynamic Programming:
  - Let $D[i,j] = d(X[1..i], Z[1..j])$
  - We examine the last block of the alignment of $X[1..i]$ and $Z[1..j]$. 

(Seq) $X$: LS CF-AV

(Real) $Y$: EACF-AV

(Match) $Z$: DACF AV
Dynamic Programming: Four Cases

- **Cases A, B, C** - no de novo sequencing errors
- **Case D**: de novo sequencing error

\[
D[i,j] = \begin{cases} 
D[i,j-1] + \text{indel} \\
D[i-1,j] + \text{indel} \\
D[i-1,j-1] + \text{dist}(X[i], Z[j]) \\
D[i',j'] + \alpha(X[i'..i], Z[j'..j]) 
\end{cases}
\]

- **D[i,j]** is the minimum of the four cases.
**Computing \( \alpha(.,.) \)**

- \( \alpha(X[i’..i],Z[j’..j]) \)
  \[
  = \min_{m(y)=m(X[i’..i])} [\text{seqError}(X[i’..i], y) + \text{editDist}(y, Z[j’..j])]
  \]
  \[
  = \min_{m(y)=m[i’..i]} [f(m[i’..i]) + \text{editDist}(y, Z[j’..j])].
  \]
  \[
  = f(m[i’..i]) + \min_{m(y)=m[i’..i]} \text{editDist}(y, Z[j’..j]).
  \]

- This is like to align a mass with a string.
- **Mass-alignment Problem:** Given a mass \( m \) and a peptide \( P \), find a peptide of mass \( m \) that is most similar to \( P \) (among all possible peptides)
Solving Mass-Alignment Problem

\[
\begin{align*}
\alpha(m, Z[i..j]) &= \min \left\{ \begin{array}{l}
\min_y \alpha(m - m(y), Z[i..j]) + \text{indel} \\
\alpha(m, Z[i..(j-1)]) + \text{indel} \\
\min_y \alpha(m - m(y), Z[i..(j-1)]) + \text{dist}(y, Z[j])
\end{array} \right. \\
\end{align*}
\]
Improving the Efficiency

• Homology Match mode:
  • Assumes tagging (only peptides that share a tag of length 3 with de novo reconstruction are considered) and extension of found hits by dynamic programming around the hits.

• Non-gapped homology match mode:
  • Sequencing error and homology mutations do not overlap.

• Segment Match mode:
  • No homology mutations.

• Exact Match mode:
  • No sequencing errors and homology mutations.
Experiment Result

- The correct peptide sequence for each spectrum is known.
- The proteins are all in Swissprot but not in Human database.
- SPIDER searches 144 spectra against both Swissprot and human databases

<table>
<thead>
<tr>
<th>Method</th>
<th>SwissProt</th>
<th>Human</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPIDER exact</td>
<td>35</td>
<td>13</td>
</tr>
<tr>
<td>SPIDER segment</td>
<td>90</td>
<td>36</td>
</tr>
<tr>
<td>SPIDER non-gapped homology</td>
<td>76</td>
<td>49</td>
</tr>
<tr>
<td>SPIDER homology</td>
<td>78</td>
<td>52</td>
</tr>
<tr>
<td>MS-BLAST</td>
<td>52</td>
<td>24</td>
</tr>
</tbody>
</table>

Table 1. The comparison of SPIDER and MS-BLAST
Example

- Using de novo reconstruction $X = \text{CCQWDAEACAFNNPGK}$, the homolog $Z$ was found in human database. At the same time, the correct sequence $Y$, was found in SwissProt database.