Computationally Analyzing Mass Spectra of Hydrogen Deuterium Exchange Experiments

Kevin S. Drew
University of Chicago

May 21, 2005

Abstract

Hydrogen deuterium exchange (HDX) using Mass Spectrometers (MS) has become a useful experiment in Proteomics which can probe structural characteristics of proteins. The data analysis of such experiments can be difficult and laborious to do by hand. First, peptides of the protein must first be matched to their corresponding mass spectrum. Second, mass spectra must be analyzed to determine the amount of deuterium in the peptides. Two computer algorithms which automate much of the data analysis of HDX experiments are described. The first, SpectralMatch, matches peptide mass spectra to the corresponding sequence in the subject protein. The second algorithm, HDXRates, computes the amount of deuterium in a peptide represented by a given spectrum. The two algorithms are shown to work on MS data from a quadropole time of flight mass spectrometer. Runtime analysis are shown for each algorithm and time trials are shown for implementations of the two algorithms.

Table of Contents

Chapter I: Introduction

What is Proteomics?

With the completion of the Human Genome Project many biology research questions have turned to understanding the human proteome. A proteome is the set of all proteins expressed in a cell at a given instant as well as the proteins many isoforms and modifications. [Ty03] The proteome is not limited to just proteins. The proteome also includes the interactions of the proteins, the structure and characteristics of proteins and proteins' formations of higher-order complexes. [Ty03] Proteins serve a large range of purposes in organisms such as signal, storage, structural or transport functions.

Proteomics is the study of an organism's proteome.

The task of mapping the human proteome far exceeds that of mapping the genome because any one gene may give rise to multiple proteins and each protein may present itself in a variable amount of states within a cell. [Ko05] Additionally, the set of proteins expressed in a cell may differ due to the specific state of the cell. Such cell states include tissue type, age of the cell, and disease or stress states. Unlike a genome, an organism's proteome is dynamic which allows scientists to map the course of a disease through its development and its treatment. [Ko05]

Proteins are made up of a long chain of amino acids and the arrangement of these amino acids are what gives a protein its specific characteristics. Many amino acids may undergoe a post translational modification or PTM which will change the attributes of the amino acid and therefore will change the attributes of the protein. The change may be local to the modification or at allosteric sites, remote to the modification. PTM identification and quantification of specific proteins are a major interest to proteomic researchers.

Why study Proteomics with Mass Spectrometers?

Many proteomic researchers are interested in identifying and quantifying proteins, identifying and quantifying modification to proteins, or mapping structural features of proteins. Many approaches may be used to aid proteomic researchers in their endeavors. One specific approach is mass spectrometry which has increasingly become the essential tool for proteomic research.

Mass spectrometry (MS) is the process by which molecules are ionized and their mass measured. [St04] MS is a desirable proteomic tool for its sensitivity, speed, ability to handle contaminated complex mixtures and tolerance for limited sample amounts. MS also has the capability of producing sequence information through the tandem MS (MS/MS) process of fragmentation. [St04] Using these features of an MS instrument, one is capable of identifying proteins as well as their modifications by searching a protein database or using a de novo sequencer which will be discussed in Chapter II.

Structural mapping of proteins however is commonly done by X-ray crystallography or nuclear magnetic resonance (NMR). X-ray crystallography is difficult to work with but has produced much of the structural information of proteins to date. Limiting factors of X-ray crystallography are the amount of the sample, purity of the sample and the inability to crystallize a protein. [SmM00] These factors have hindered high throughput structural experiments. NMR is another approach to gain insight into the structural characteristics of proteins but has limitations of its own. NMR is not suitable for analyzing large proteins and is less sensitive than a mass spectrometer.

Where X-ray crystallography and NMR experiments fall short, MS fills in the gaps. MS is capable of dealing with large proteins and can be used as a high throughput experiment; two advantages the other approaches do not have.

Why use Hydrogen Deuterium Exchange to discover characteristics of proteins?

Hydrogen deuterium exchange (HDX) experiments can access structural information about proteins in a high resolution fashion. [En01] Hydrogen atoms on proteins exchange with surrounding solution. These hydrogen molecules may exchange with an isotope of hydrogen, deuterium, which as a result adds mass to the protein. This additional mass may be measured by NMR or MS. HDX has been shown to give structural information on many states of proteins, including in solution, membrane bound, molten globular and other biologically relevant states. [En01]

A proteomic researcher is interested in the rates at which the hydrogen molecules exchange because this leads to structural information, dynamic behavior or thermodynamic parameters as a function of time. [En01] The experiment is carried out over multiple steps where exchanging hydrogen molecules are measured at multiple time points. This experiment is commonly done using NMR but has been extended to MS in order to increase resolution and examine larger proteins. MS HDX has been shown to provide insight into protein-protein interfaces and conformational changes during protein folding. Other studies have used MS HDX for detecting covalent modification, protein motion and enzyme functions. [Ho03]

Description of remaining chapters

Chapter II provides background material for understanding protein and peptide principles. It also covers mass spectrometry basics as well as hydrogen deuterium experiment basics. Chapter III gives motivation for designing a MS HDX toolkit as well as further motivation for which the toolkit may be extended. A review of the literature pertaining to proteomic MS tools, specifically database search techniques and de novo sequencers is discussed in Chapter IV. Chapter V states the problem of analyzing HDX MS data as well as what has been done in the field to alleviate these complications. Chapter VI presents an algorithm for matching MS data to specific segments of a protein sequence. Chapter VII presents an algorithm for determining the rates of hydrogen exchange from HDX MS data. Lastly, Chapter VIII puts forth results obtained and conclusions reached from the analysis of an HDX MS experiment. Also in Chapter VIII, future work is discussed. Appendix I lists terms used and their definitions and Appendix II lists references.

Chapter II: Background

Proteins and Peptides

Cells use proteins for a variety of purposes including structural support, storage, transport, motor/mechanical, enzyme/catalyst, signal receiving, gene regulating as well as many other purposes.

Makeup and structure of proteins

The "Central Dogma of Molecular Biology" is the concept that DNA is transcribed into RNA and RNA is translated into Proteins. In the simplest model, DNA is considered the long term storage of information which is then converted into short term storage, RNA. RNA is then used as the blueprint to make proteins. This simple model is not sufficient to describe more complex functions of a cell because of many other factors such as positive feedback loops or reverse transcription. However for our discussion, the simpler model will suffice. For more detailed background, one can reference "Essential Cell Biology" by Alberts et al. [A+]

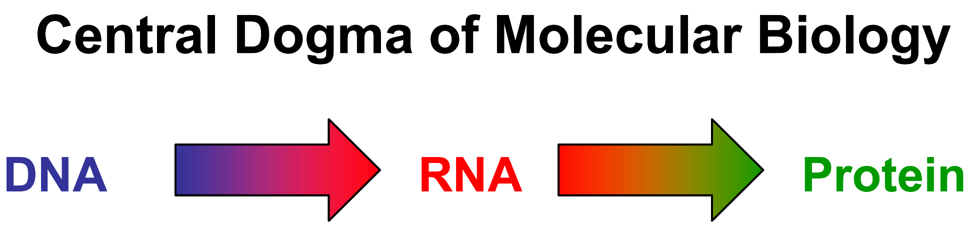


Figure 2.1: The Central Dogma of Molecular Biology is the idea that DNA is transcribed into RNA and RNA is translated into Proteins.

Amino acids are the basic building blocks of peptides. Amino acids are covalently linked together by a protein machine known as a ribosome to form peptides. A protein is a functional peptide or polypeptide. RNA serves as the blueprint for the ribosome to add additional amino acids to a growing protein strand. A protein strand is also known as the primary sequence of a protein.

There are twenty different amino acids used in creating proteins, each with a different side chain. The molecular makeup of an amino acid is the same as the other amino acids except for the side chain. It is the side chain that gives amino acids their properties and allows them to be classified into groups. Amino acids are commonly grouped into hydrophobic, non-polar, negatively charged and positively charged.

The structure of an amino acid consists of an amino group and a carboxyl group which are labeled the N-term and C-term by biologists. When a ribosome links two amino acids together, the carboxyl group of one amino acid is bonded to the amino group of the other with a net loss of a water molecule. This bond is called a peptide bond. The strands that result from peptide bonds are sometimes called the peptide backbone. Figure 2.2 describes this process.

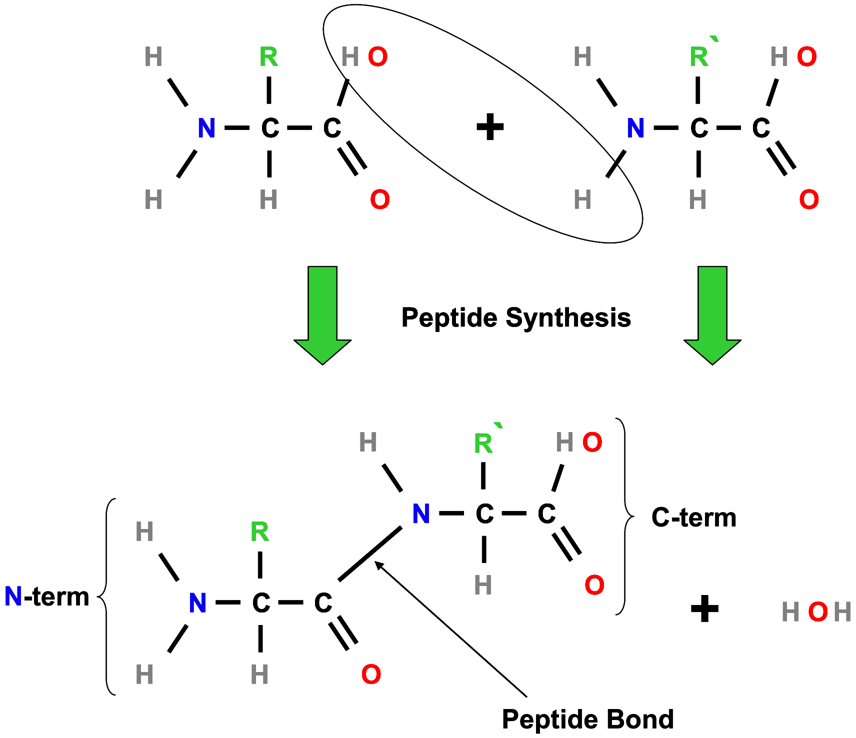


Figure 2.2: A peptide bond is formed when the C-term of one amino acid and the N-term of another amino acid bond and release a water molecule. R and R' denote the side chain of the two amino acids.

The peptide backbone is flexible enough that it is capable of folding upon itself. Peptides fold into a conformation of lowest energy. Contributions to this energy are different types of non-covalent bonds such as hydrogen bonds, ionic bonds and van der Waals attractions. When a protein has folded into a conformation of lowest energy it is said to be stable.

Proteins can be denatured or unfolded. This process of denaturing is done by exposing a protein to a substance such as Urea or an environmental condition such as heat. A denatured protein may or may not be able to refold back into its native state on its own in the absence of the denaturant. Sometimes initial folding or refolding requires the use of a chaperone, a specialized protein that guides another protein through the folding process.

The folding process may undergo several steps in which secondary structures and tertiary structures form. Common secondary structures are the alpha helix, Figure 2.5 and the beta sheet, Figure 2.4.

An alpha helix consists of multiple amino acids which interact with each other through hydrogen bonds. These hydrogen bonds cause the peptide backbone to curve into a helix shape. A beta sheet on the other hand is formed when two or more portions of the peptide backbone interact through hydrogen bonds and line up in a sheet.

Tertiary structures consist of interacting secondary structures as well as loops and random coils. Quaternary structures are when multiple proteins interact in a larger complex.

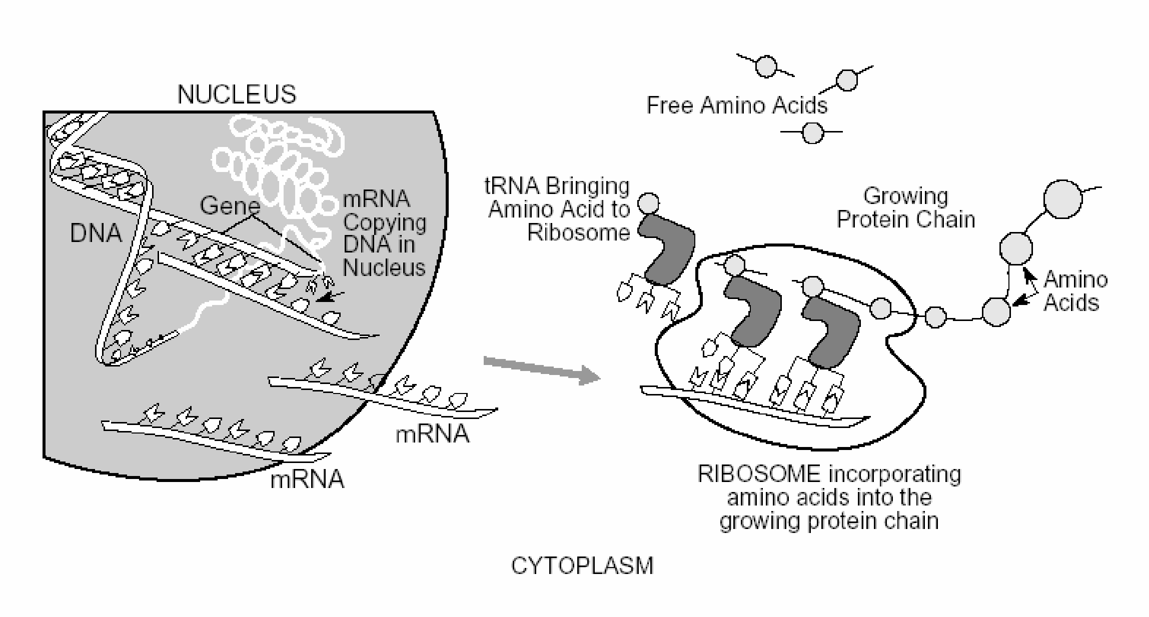

NUCLEUS
Free Amino Acids
tRNA Bringing Amino Acid to Ribosome
Growing Protein Chain
Gene
mRNA Copying DNA in Nucleus
DNA
Amino Acids
mRNA
mRNA
RIBOSOME incorporating amino acids into the growing protein chain
CYTOPLASM

Figure 2.3: DNA is transcribed in the nucleus into RNA. RNA is transported out of the nucleus and links up with a ribosome which translates the RNA into a protein. (DOE Human Genome Program "Primer on Molecular Genetics" 1992, U.S. Department of Energy, Office of Energy Research, Office of Health and Environmental Research, Washington, DC 20585)

Proteins bind to other molecules through binding sites created by their native folding. A binding site of a protein has a specific fingerprint which allows it to only accept binding with a particular molecule. Also, these binding sites are regulated depending on the protein's modifications by other unrelated molecules. These covalent modifications are known as post translational modifications (PTM). A PTM of a protein may cause the protein to be activated, deactivated or even cause an allosteric conformational change. [Al04]

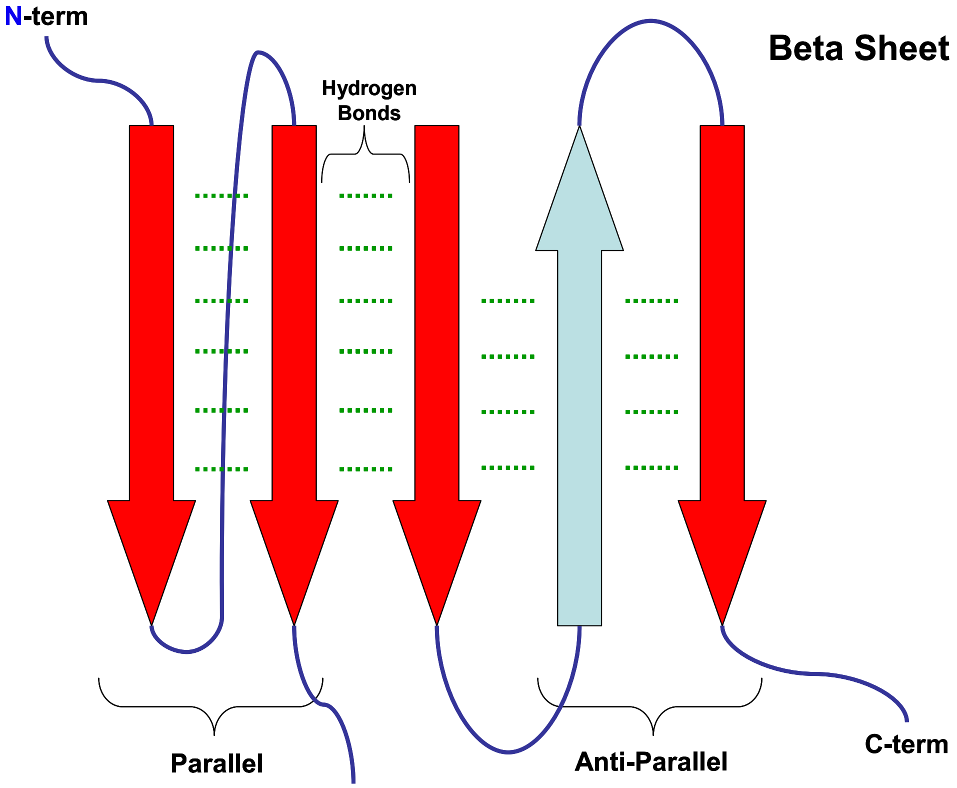


Figure 2.4: A beta sheet is a common secondary structure found in proteins. The beta sheet is formed by peptide strands hydrogen bonding between themselves. It may consist of parallel or anti-parallel strands.

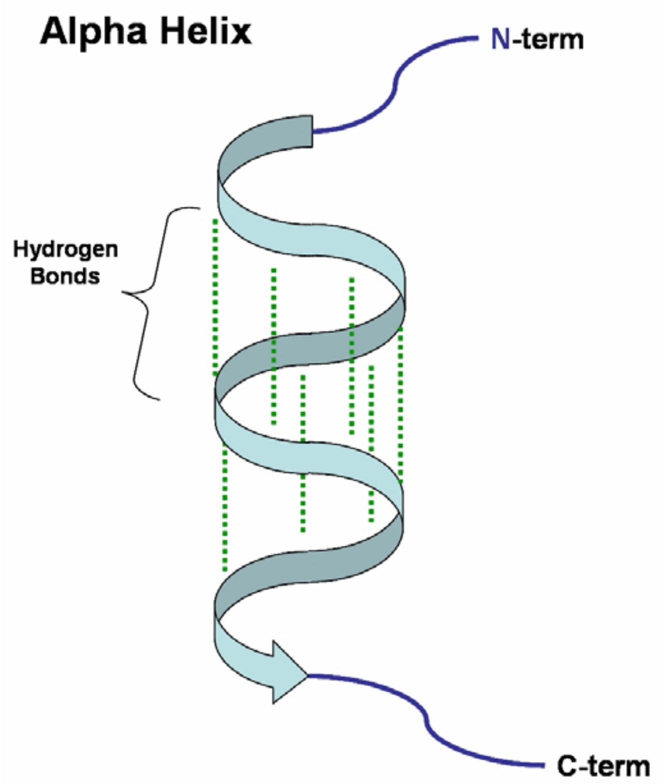


Figure 2.5: An alpha helix is a common secondary structure found in proteins. An alpha helix is formed when amino acids form hydrogen bonds with other amino acids and forms into a helix.

Mass Spectrometry Basics

Mass Spectrometry (MS) is a technique used by biologists and chemists to determine the composition of an unknown analyte. A mass spectrometer is an instrument that is used to determine the mass of an analyte by converting the analyte to a gas-phase ion and measuring the m/z (mass to charge ratio) of the ion. An analyte can be many different molecules but for our purposes it will be a protein or peptide.

An MS experiment begins with placing the analyte into the source. The source is what ionizes the analyte. The analyte must be ionized in order for it to be detected and measured. There are two common source technologies currently used in proteomics, matrix assisted laser desorption ionization (MALDI) and electrospray ionization (ESI). The MALDI source uses a laser to ionize the analyte while ESI sprays the analyte out of an electrically charged needle which ionizes the analyte. [Li02]

Once the analyte is ionized, the mass analyzer sorts the ions according to their mass to charge ratio (m/z). [Li02] It should be noted here that peptides, when ionized, may contain several charges. This causes the peptide to show up in the lower mass range of a spectrum. A simple calculation can be done to obtain the accurate mass of a peptide and will be discussed shortly.

Finally, the ions are received from the mass analyzer into the detector. The detector detects and records the m/z value and abundance of the ions. Collectively, the m/z values and abundances are what is known as a spectrum. After the ions are recorded in a spectrum, the data can then be graphed visually for human interpretation or inputted into software where it can be analyzed.

A common technique used with MS is what is known as tandem MS or MS/MS. Tandem MS involves the use of two mass analyzers or multiple stages of MS with a single mass analyzer. The first is used as described above. The second obtains specific ions from the first mass analyzer and fragments them with a high energy gas in a process known as collision induced dissociation (CID). [Li02] Other fragmentation techniques are available such as electron capture dissociation (ECD) [ZuK98] and electron transfer dissociation (ETD) [Sy04]. ECD and ETD are considered softer fragmentation techniques in comparison to CID because they do not subject the whole analyte to very high energies.

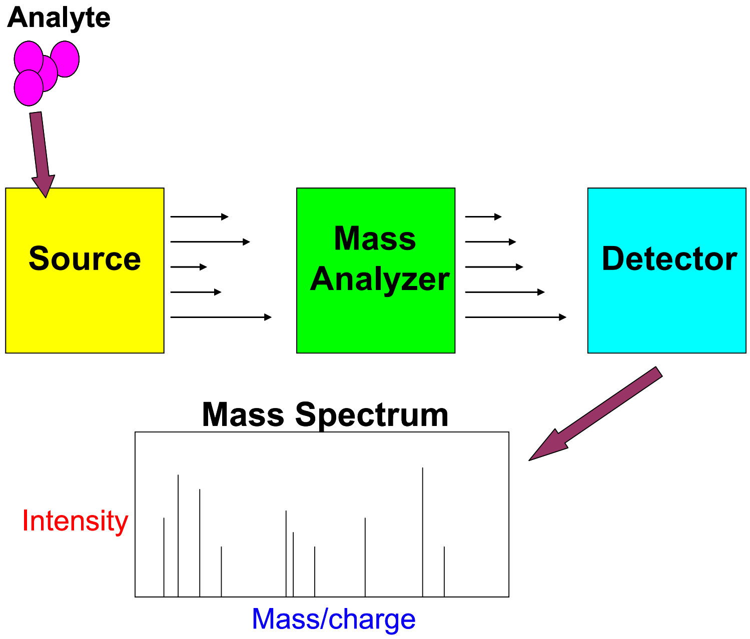


Figure 2.6: A Mass Spectrometer has three essential parts, the source, the mass analyzer and the detector. The source is where the analyte is ionized and made capable of detection. The mass analyzer sorts the ionized analyte by its mass/charge value. Finally, the detector observes the mass/charge values and returns them to the user.

The result of measuring m/z of ions is in the form of a mass spectrum. A mass spectrum is a graph with m/z on the x axis and intensity on the y axis. The peaks that make up the spectrum are representations of ions or fragment ions of the analyte. Mass spectra are used to identify unknown proteins and help determine their amino acid sequence often using a process known as de novo sequencing. De novo sequencing is the process of analyzing the peaks in a mass spectrum that represent fragment ions of the peptide. Since the masses of the amino acids are known, one can deduce from the resulting peaks the composition of the peptide.

De Novo Sequencing

De novo sequencing involves analyzing a spectrum to determine the sequence of the peptide which is represented by that spectrum. This is done by assigning ions to fragments of the peptide. To understand de novo sequencing, one must first understand the concept of b and y ions. Inside the mass spectrometer, a gas-phase ion peptide is fragmented into two separate fragment ions. One fragment ion will contain the N-terminus of the peptide and the other fragment ion will contain the C-terminus. The fragment ion that contains the N-terminus is the b ion and the one that contains the C-terminus is the y ion. This is shown in Figure 2.7. Since amino acids have known masses, mathematical relationships can be derived between sequences and their corresponding ions as shown in Equations 2.1 and 2.2.

Sequence: N-terminus, AA1, AA2, AA3, ..., AAn, C-terminus

b ions:
b2 ion: N-terminus, AA1, AA2
b3 ion: N-terminus, AA1, AA2, AA3
bn-1 ion: N-terminus, AA1, AA2, AA3, ... AAn-1

y ions: y1 ion: AAn, C-terminus
y2 ion: AAn-1, AAn, C-terminus
yn-1 ion: AA2, ..., AAn-2, AAn-1, AAn, C-terminus

Figure 2.7: b1 ion is not observed in spectra due to chemical reasons of ionization

m/z of b(i) ions:
Maa = Mass(AA1) + Mass(AA2) + ...+ Mass(AAi)
Mb = Maa + Mass(N-terminus) + Mass(charge)
m/z = Mb / charge

Equations 2.1

m/z of y(i) ions:
Maa = Mass(AAn) + Mass(AAn-1) + ...+ Mass(AAn-i+1)
My = Maa + Mass(N-terminus) + Mass(charge)
m/z = My / charge

Equations 2.2

-where Maa is mass of amino acids
-where Mb is mass of b ion fragment
-where My is mass of y ion fragment
-where Mass is a function that determines the mass of the given amino acid, amine group or carboxylic acid group. Charge generally has a mass of one Hydrogen.
-where charge is the number of protons that ionized the fragment

Equations 2.1, 2.2

This relationship between b and y ions can be expanded to determine a b ion's location in the spectrum, given a y ion m/z value and the precursor mass and vice versa.

m/z of b(i) ion:
b(i) = Mzp – y(n-i)

Equation 2.3

m/z of y(i) ion:
y(i) = Mzp – b(n-i)

Equation 2.4

-where n is the number of amino acids in the peptide
-where Mzp is the m/z value of the precursor mass in the spectrum

Equation 2.3, 2.4

These compliment equations are helpful for distinguishing false positives when an ion is discovered. One may discard an ion hit if its complement ion is missing.

The concept of compliment ions is important because some ions may be obscured by noise or other peaks and if its compliment is not obscured coverage is still possible. Another reason that this concept is fruitful is when an ion is not present at all. The b1 ion, for chemical reasons, is never present in a peptide mass spectrum. The yn-1 ion then becomes the only ion that is able to determine the first amino acid on the N-terminus end.

The fragmentation of a peptide inside the mass spectrometer can occur at any of the individual amino acids making up the peptide. This results in a spectrum with peaks that represent many different fragment ions all of which contain the summation of the composite amino acid masses.

B and y ions are not the only ions that appear in a spectrum. During fragmentation, a b or y ion may loose a water molecule or another molecule which shows up at a lower mass than the b or y. These ions are known as internal loss ions. A ions are also common in mass spectra. The a ion appears when the initial fragmentation of the peptide breaks at a bond closer to the amino end of the amino acid. This results in an ion with lower mass than the common b ion. Immonium ions are also evident in spectra and provide a de novo sequencer clues to what types of amino acids are present but not where or how many.

The de novo process can be described (simplistically) as analyzing a set of peaks (either b ions or y ions), taking the difference of two adjacent peaks in the set and determining if an amino acid mass is the result. Since a b(i) ion is a b(i-1) ion plus an additional amino acid, AAi, we are able to deduce what the additional amino acid is by taking the difference. The same can be said about the set of y ions.

HDX Experiment Basics

A technique known as hydrogen-deuterium exchange(HDX) can help distinguish the characteristics of different proteins. Each amino acid has a set of hydrogen atoms that exchange with the water molecules that surrounds the amino acid. This exchange cannot be observed directly but when the hydrogen atoms exchange with another atom, such as deuterium, the effects can be observed. A deuterium atom, an isotope of hydrogen, has twice the mass of a hydrogen atom and therefore the exchange is observed when the amino acid's mass is measured using mass spectrometry. Not all hydrogen molecules within amino acids can be observed. Only amide hydrogen molecules exchange at rates that are readily observed [SmD97].

Amide hydrogen atoms exchange at variable rates due to several factors such as interactions with the adjacent Amino Acid residue[Ba93], the degree of solvent accessibility or the position of the hydrogen atom in the secondary structure of a protein[Ho03]. For example, if the amino acid in question participates in a secondary structure known as a Beta sheet (Figure 2.8, a protein with an alpha helix and beta sheet), and is hydrogen-bonded to another amino acid in that Beta sheet, then a typically fast exchanging hydrogen atom will exchange at a slower rate or not at all.

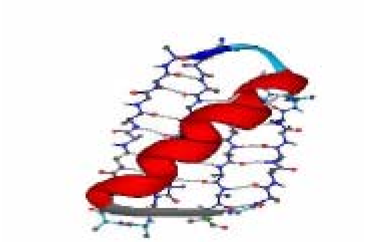


Figure 2.8: This protein is made up of an alpha helix (red) and a beta sheet (blue, behind alpha helix). (Image compliments of Andres Colubri, University of Chicago)

An HDX experiment is a series of steps that requires care to control temperature and pH. First, the protein is incubated in D20 (heavy water containing deuterium) for a period of time t. This step is when the hydrogen atoms exchange with deuterium atoms. Next, the protein is quenched in a solution of low pH and with a temperature of 0degC. This step is to preserve the exchanges that occurred in the first step. Hydrogen molecules exchange at a slower, observable rate at low pH and low temperature. The protein is then digested with pepsin, a protease which cleaves the protein into peptides. Further discussion regarding pepsin and other proteases will follow. The digestion step also needs to be done at a temperature of 0degC. Finally the protein is introduced into the mass spectrometer and run in a manner similar to other samples. This experiment is repeated for several time points, t. Again, t is the length of time the protein is subjected to D20 [Ho03].

To run an HDX experiment with MS, one is dependent on a protease to digest the protein into peptides. Many of the proteases used in MS have specific cleaving rules which make it fairly routine to determine how a protein will be digested into peptides. For example, trypsin, a common protease used in MS, cleaves proteins after the amino acids arginine and lysine. Many of the common proteases used in MS only work under specific environmental conditions. Trypsin works in conditions of alkaline pH and physiological temperatures. [Ki00]. Since HDX experimental parameters require low pH and low temperature, the only protease that can be used is pepsin which cleaves under these conditions. One characteristic of pepsin under these conditions is that it is a nonspecific protease [Ho03]. In other words it does not cleave solely at a few specific amino acids as many other proteases do. Pepsin will cleave residues uniquely specific to the protein undergoing digestion as opposed to fixed rigid cleaving rules, like trypsin. That is, pepsin will cleave in different locations for different proteins. A protein in an unfolded or denatured state subject to pepsin digestion will also produce peptic fragments different from the native state digestion [Ho03].

Chapter III: Motivation

HDX Experiment MS

HDX can help discover many characteristics of proteins ranging from binding sites to protein dynamics. Hoofnagle et al. describes several applications of hydrogen deuterium exchange experiments using mass spectrometry in a 2003 review including protein folding, stability, ligand binding, protein aggregation, protein-protein interactions, epitope mapping and dynamics of protein interactions. [Ho03] Other reviews have focused on hydrogen exchange experiments specifically for protein folding [En00, En96].

HDX with mass spectrometry has been used to examine the pathways of protein folding, examine folding rates [De98, Ey00] and examine role of chaperone proteins which assist folding [Ro94]. Hoofnagle et al. also notes practical applications of HDX with mass spectrometry in the field of protein stability studies. HDX-MS is also used in order to determine promising candidates for NMR and X-ray crystallography experiments which are expensive, time consuming structural analyses techniques. [Ho03]

Using HDX-MS to discover protein interaction and binding sites is generally straightforward for a biological researcher. One can determine from exchange rates of solvent protected areas of proteins that there is binding. The analysis of a solvent protected area can be applied to discovering ligand binding, protein aggregation, protein-protein interactions [Ma98] or epitope mapping [BO02,Ya02]. [Ho03]

HDX experiment MS/MS

Tandem mass spectrometry HDX experiments are thought to be able to give single amino acid resolution of solvent unprotected protein sites. Currently, there are conflicting reports of the validity of these experiments due to scrambling of deuterium molecules during the collision phase of tandem MS. [Ho03] The scrambling is thought to take place because of the high energy reaction that takes place during collision induced dissociation (CID). A recent article by Jorgensen et al. discusses findings of 100% scrambling during CID. [Jo04]

Alternative methods to CID have been developed known as Electron Capture Dissociation (ECD) [ZuK98] and Electron Transfer Dissociation (ETD) [Sy04]. These methods are a gentler way of fragmentation and therefore are thought to reduce or even eliminate the scrambling that takes place during fragmentation. More experimental work needs to be accomplished before these methods are verified as free of scrambling but the methods are promising.

Thiophosphorylation experiments

Thiophosphoralyation is a chemical technique that enables researchers to track the degree of phosphorylation of proteins in a cell. The phosphorylation of proteins is known to be involved in many cell functions such as cellular transcription, replication, apoptosis and signal transduction. [Kw03] At any given point of a cell's life cycle, two thirds of the proteins in the cell may be phosphorylated. To determine which proteins are phosphorylated due to a specific environmental change, one must "tag" the phosphate molecule in order to determine its origin. A researcher will then be able to determine which proteins were phosphorylated due to the environmental change and which were phosphorylated due to previous unrelated events. This technique has been proven to work with MALDI mass spectrometers using ATPgammaS as a thiophosphate. [Pa05] Future experiments will involve reducing the thiophosphorylation site to single amino acid resolution using tandem mass spectrometry.

Post Translational Modification Profiling

Many post translation modification profiling experiments are aimed at discovering whether a peptide substrate has been modified or not. The outcome may usually be expressed as a discrete state, for example the substrate is modified or it remains unmodified. Commonly in these types of experiments, the peptide substrate is exposed to an enzyme present in bodily fluid or tissue extract. The presence of a particular enzyme may represent a disease state. The enzyme interacts with the substrate and the substrate becomes modified. This modification can then be detected by a mass spectrometer. It is noted by Dr. Steve Patrie (Department of Pathology, University of Chicago), that modifications should be studied as rates of incorporation rather than the discrete outcomes in order to determine levels of disease states in bodily fluids or tissues.

Each of the above experiments is attempting to discover rates of change of mass due to addition or subtraction of a known mass quantity. The same tools or slightly modified versions which have been developed for analyzing MS HDX spectra can be applied to MS/MS HDX spectra, MS/MS Thiophosphorlation and post-translational modification profiling.

Chapter IV: Review of Literature

This chapter will contain two sections. The first will discuss what has been published on algorithms for searching genomic and protein databases with data that can be obtained with mass spectrometers. The second section will discuss what has been published on algorithms for identifying mass spectra using nothing more than the information and data from the individual MS run known as de novo sequencing.

Database Searching Techniques

The primary reason a researcher would use a database search program would be to identify an unknown sample protein which he/her is studying. There are two possibilities that may happen during a database search. One, the unknown sample protein is in the database. In this case, the database search program should extract the database entry that is of the sample protein. The second possibility is that the unknown sample protein is not in the database. The database program should then return database entries of proteins that are homologous to the sample protein.[Pe99] It is common for homologous proteins to be equivalent proteins in different species[Pe99] or a result containing a homologous protein may lead the researcher to better understand the function and characteristics of the sample protein.

There are three types of databases which are searched by database search programs. First, primary nucleotide sequence databases are made up of genomic data or DNA base pairs. A good example of a primary nucleotide sequence is GenBank.[Be04] Since nucleotides are not the same language as amino acids, in order to search for a protein, database search programs must convert nucleotide data to amino acid data. The second database is the comprehensive protein sequence database which is derived from nucleotide databases. These databases are considered an interim point for when protein databases do not have entries for the specific protein. Examples of a comprehensive protein sequence database are GenPept [Be04] and TrEMBL [Tr05]. The third and final type of database is the curated protein database. This type of database contains the sequence, function, specific characteristics and links to literature about the protein. [Ki00] An example of a curated protein database is Swiss-Prot. [SP05] According to Kinter [Ki00], there are five factors that are used to judge the quality of a database which are number of entries, errors, redundancies, annotations and frequency of updates. A researcher must weigh these factors before a database is chosen to search.

A researcher not only has to choose a database to search but also has to choose how to search the database. There are several different types programs that will search the different genomic and proteomic databases using data from mass spectrometers. The different types of searching are using amino acid sequences as a query, using peptide molecular weights as a query and using uninterpreted product ion spectra as a query.[Ki00] Each type of query can be formed from data collected from mass spectrometers in one way or another.

Uninterpreted product ion spectra and peptide molecular weights require the least amount of analysis to obtain a suitable query for database searching from the mass spectrometry data. Using amino acid sequences as a query for database searching, the mass spectrometry data needs to be analyzed and sequenced before the search can take place.[Ki00] Examples of programs that can search a database for an amino acid sequence are FASTA [Pe88] and BLAST [BL05].

Database searching using peptide molecular weights as a query require relatively little analysis compared to amino acid sequence database searching. The experimental data are peptide molecular weights from enzymatically digested proteins. [Pe99]. The proteins are digested with a protease such as trypsin, ran through a mass spectrometer and the weights are chosen from the resulting mass spectrum. Usually the weights are the most abundant peptide ions of a mass spectrum. The entries in the database are theoretically cleaved with the same protease used on the protein and the masses of the resulting fragments are compared to the input molecular weights. The database entries are then ranked in order of the number of matches to molecular weights of the sample protein. [Ki00] The quality of the matches depends on the total intensity covered, which is the sum of the intensities of the matched molecular weights. Other factors that involve a good match are the magnitude of mass difference per match, the percent of amino acid coverage of the database entry sequence, and finally the agreement with other experimental data available on the protein. [Ki00] This database search technique is also referred to as a peptide mass fingerprint. [Pe99]

One algorithm which uses peptide mass fingerprints is called MOWSE. [PaH93] MOWSE uses a scoring function based on frequencies of peptide masses in a particular database. These frequencies are used to determine a p-value or estimation that the match was found at random. Pvalues are used frequently in bioinformatics and provide estimates of expected rates of false positives. [HaH03]

Mascot is a database search program which uses several different search techniques discussed previously. [Pe99] Mascot uses probabilistic based scoring similar to MOWSE. It considers the validity of a match by comparing the probability of chance to some significance value. For example, a match may be significant if the probability that a match was by chance is less than 1 in 20. [Pe99] Using probability based scoring allows researchers to easily tell if a result is significant, allows researchers to compare results to other search results and also allows researchers to customize searches to reach maximum probability of a match. [Pe99]

The final way to search a database is to use uninterpreted product ion spectra as a query. Product ion spectra are tandem mass spectrometry data from peptides. [Pe99] The technique involves correlating sequence-specific information in ion spectra with amino acid sequences in database entries. The search will use what digestion technique (ie. protease) to determine the possible amino acid sequences of peptides to be searched. There is some ability of these searches to allow for amino acid modifications or substitutions in the sample protein. [Ki00]

There are two common ways of using uninterpreted product ion spectra to search databases, comparison of m/z values to calculated values and cross correlation of experimental spectra to theoretical database spectra. The first is similar to database searching using peptide molecular weights. An example of this type of search is the program MS-Tag. [Cl99] M/z values of a spectrum are compared to calculated values from the database and the entries are ranked according to number of matches. First, the b and y ions and possibly immonium, water loss, ammonia loss and internal cleavage ions need to be calculated for the database entry. Note that these calculated ions will lack intensity data due to the current inability to predict intensity of ions. Next, the experimental spectrum will be thresholded to only include the most abundant ions. Thresholding is performed in order to limit the ions considered to those of significant intensity and to reduce contributions from systematic noise. Finally, the threshold experimental spectrum is compared to the theoretical database spectrum peak for peak and the database entry with the most matches is ranked highest. [Ki00]

The second way of database searching using uninterpreted product ion spectra is to cross correlate the experimental spectrum and the calculated database spectrum. In this case the experimental spectrum and the theoretical spectrum have intensity data. Intensity needs to be calculated for the theoretical spectrum and usually b and y ions are given higher intensities than a ions and loss ions. It should be noted this is a very simplistic way of calculating intensity. The experimental spectrum's data undergoes segmented normalization which equalizes ions in different mass ranges of the spectrum. [Ki00] The cross-correlation is then used to compare one spectrum to the other at different offsets. The score produced is the difference of the value of the comparison at offset zero and the average of all other overlap values at offsets not equal to zero. [Ki00]

SEQUEST is one such program that uses cross-correlation as a method for scoring uninterpreted product ion spectra to database entries. [EnM94] First, the experimental spectra are put through a pre-processing step which eliminates noise and normalizes intensities. Next, the precursor mass of the experimental spectrum is matched to theoretically digested database entries to identify possible peptides. In step three, the theoretical database spectrum and the experimental spectrum are compared, summing intensity data on matches and giving continuity of an ion series an extra score. The presence of immonium ions can increase or decrease scores depending on the amino acid sequence. Finally, a fast Fourier transform is used to calculate the cross correlation of the spectra and these scores are used to rank the database entries. The scores obtained in step three are also reported to give validity of the match. [EnM94]

De Novo Sequencers

Database searching is a valuable tool for identifying sample proteins but a database search is only as good as the database it is using. [Lu04] Specifically, database searching is limited by the proteins that are in the database. If a sample protein is not listed as a database entry, then the results of a search may come up empty or inaccurate. Also, proteins often undergo modifications and are mutants of a wild type or have a post translational modification. These modifications will be evident in the resulting mass spectra but again database searches may not return correct results. When a researcher is unable to get valid results using database search techniques, they may turn to de novo sequencers. [Lu04]

De novo sequencing is the attempt to derive the sequence of a peptide through analysis of a spectrum without help from a database. [Lu04] De novo sequencing is often used as a compliment to database searching to provide extra validation of identified proteins. [Lu04] Currently, there are several approaches that are used ranging from naive approaches to sophisticated graph theory approaches.

The naive approach involves simply computing all the possible peptides according to the precursor mass of the experimental spectrum. Then, for each theoretical peptide computed, a theoretical spectrum is derived from the peptide sequence. These theoretical spectra are then compared to the experimental spectrum similar to database searches described above. [Lu04]

Another approach, subsequencing, computes short amino acid subsequences that are then tested against the experimental spectrum. When a match is found, the subsequence is extended one residue at a time until the end of the spectrum is found. [Lu04]

A graphical display approach is used to display the spectrum graphically and show peaks with differences of one or multiple amino acid masses connected by a line. This approach is used so that a researcher can view the spectrum and determine which peaks are relevant and which are not. This approach is not a viable de novo solution in high-throughput situations because a researcher must be present to validate every spectrum. [Lu04]

A common approach utilizes graph theory. A graph consists of vertices which are each peak in the spectrum and directed edges are connected between vertices whose mass difference equals the mass of one amino acid or the mass of multiple amino acids. The correct path is the path which represents the correct sequence of the experimental spectrum. There are several ways to determine a possible correct path. A de novo sequence algorithm, SeqMS [FC95], uses Dijktra's single source, shortest path. [Lu04]

Another algorithm, Lutefisk [Ta97], first finds significant ions in the experimental spectrum. The N and C terminal ions are then found in the spectrum. The algorithm then creates a list of m/z values and probability of fragmentation at the m/z value also known as a sequence spectrum. Next, all possible sequences are computed with the sequence spectrum. These possible sequences are scored and ranked. [Ta97, Lu04]

The final graph theory approach is an algorithm called Sherenga. [Da99] In this algorithm, a sample dataset is used to learn about ion types and their probabilities of appearance in spectrum. Since different mass spectrometers produce slightly different data, this approach creates a machine independent algorithm. Next, the Sherenga algorithm converts the experimental spectrum into a graph similar to the Lutefisk algorithm. One difference is that for each peak in the spectrum there is one vertex for each ion type specified such as a ions or water loss ions. The graph is then used to find the longest antisymmetric path. [Da99, Lu04]

Finding the longest antisymmetric path is an NP-complete problem. [Lu04] Chen et al. [Ch01] proposed a dynamic programming solution to this problem. This solution returns a result of the optimal solution to the longest antisymmetric path problem. Due to noise and unknown ions in experimental spectrum, the optimal solution may not be the correct amino acid sequence for the sample protein. A suboptimal algorithm is presented by Chen and Lu. [Lu03]

A scoring scheme of probabilistic methods for scoring graph theory was devised by Frank and Pevzner known as PepNovo. [Fr04] PepNovo uses a probabilistic network which reflects chemical and physical rules of peptide fragmentation to assign probabilities to vertices of a graph. The tables of probabilities are created using a training set of experimentally derived spectra. A hypothesis test is carried out at each cleavage to test whether the cleavage is more likely to be genuine cleavage of a peptide or random cleavage.

Another approach used in de novo sequencing is the divide and conquer technique. This is similar to the approach of computing all possible peptide combinations for a particular precursor mass but on a much smaller scale to reduce excessive computation. DACSIM, created by Zhang uses this method. [Zh04a] After a spectrum goes through a preprocessing stage, the spectrum is broken up until a subspectrum is small enough to compute all possible peptide combinations. These combinations are then recombined with other subspectrum to produce larger possible solutions. Once a collection of sequence candidates is generated from the divide and conquer algorithm, the sequence candidates are scored according to the similarity of their simulated spectrum to the actual experimental spectrum. The simulated spectrum is the result of a mathematical model to simulate peptide fragmentation. [Zh04b]

Finally, a pattern recognition algorithm named SALSA has been used as a high throughput method for discovering post-translational modifications in tandem mass spectra in correlation to other features of tandem mass spectra such as a ions and neutral loss ions. [Ha01] Although technically not a de novo algorithm, SALSA uses many techniques of de novo algorithms such as preprocessing spectra to reduce spurious results and correlation between ion pairs. The final score of a spectrum is the sum of normalized intensities of peaks which matched user defined characteristics.

Chapter V: Statement of the Problem

Complications of De Novo Sequencing

De novo sequencing and mass spectra in general are more complicated than I have described in the background chapter. First, b and y ions are not the only ions present in a mass spectrum. A spectrum may contain other peaks such as water loss ions, immonium ions, doubly fragmented ions, isotopic peaks, multiply charged ions and the possibility of gaps in ion sets. Post translational modifications to amino acids in peptides also make de novo sequencing a challenge. All of these add to the complexity of sequencing spectra, but all give extra information about the peptide in question.

Water loss ions are b or y ions that have loss of a water molecule thereby decreasing the mass by the mass of a water molecule, 18 mass units. One may observe a water loss by subtracting 18 mass units from an ion and look for a prominent peak in the spectrum. Immonium ions are ions in the low mass range of a spectrum that indicate the presence of a certain amino acid (Ki00). Doubly fragmented ions also known as by ions are ions that have undergone fragmentation twice and contain neither an N-terminus nor a C-terminus. There are many other such ions similar to the ones described.

Isotopic peaks are a result of the elemental composition of peptides. Some elements, such as carbon and nitrogen, have natural isotopes which have different masses than the usual form of the element. Carbon has a mass of 12.0, denoted C12, and has a natural isotope of mass 13.003, C13. Isotopes also show up in nature as certain percents relative to total amount of the element in nature. C12 has a natural abundance of 98.89%, where C13 has a natural abundance of 1.11%.

These natural abundances are apparent in mass spectra. For small natural peptides, the monoisotopic peak is the most abundant peak. The monoisotopic peak is an ion which has all the usual forms of the elements. For example, a peptide may have 50 carbon atoms and all of them will be C12 in the monoisotopic peak. Another peak, however, will be present at 1.003 mass units greater than the monoisotopic peak but with lower intensity. This is referred to as the C13 peak. The C13 is an ion which has all the usual forms of the elements but with one C13. With larger peptides, the C13 peak will become the most abundant peak. This is because the probability of a peptide containing a C13 atom increases with every carbon atom in the peptide. Peptides of around 100 carbon atoms begin to present larger C13 peaks than the monoisotopic peak. Carbon is not the only element with isotopes that we need to take into account. Hydrogen, nitrogen, oxygen and sulfur all have natural isotopes that a de novo sequencer needs to be aware of. Isotopes may complicate the de novo process but they are also used as a check to verify the accuracy of the results obtained from the process.

HYDROGEN		CARBON		NITROGEN		OXYGEN		SULFUR	
Mass	Abd.	Mass	Abd.	Mass	Abd.	Mass	Abd.	Mass	Abd.
1.00782	0.99990	12.00000	0.98890	14.00307	0.99630	15.99492	0.99760	31.97207	0.95020
2.01410	0.00010	13.00335	0.01110	15.00011	0.00370	16.99913	0.00040	32.97146	0.00760
						17.99916	0.00200	33.96786	0.04220
								35.96709	0.00010

Table 5.1 isotopes, masses, and abundances (Abd.)

Like isotopes, multiply charged ions complicate but also help to verify results of de novo sequencing. A multiply charged ion is an ion with more than one proton attached to the ion. This affects the ions in the mass spectrum in two ways. First, the extra protons add one mass unit for each proton onto the ion's mass. The second effect is the ion will be observed at an m/z value of the mass divided by the charge. (note: m/z is also called "mass to charge ratio").

Isotopic peaks and multiply charged ions are used in conjunction to determine the charge of a particular set of ions. Since isotopic peaks are close to one mass unit apart when observed on singly charged ions, isotopic peaks will be observed around .5 mass units apart on doubly charged ions, due to the mass being divided by the charge. On triply charged ions, isotopic peaks will be observed at .333 mass units apart and so on and so forth for greater charged ions. This technique may be used by de novo sequencers to more easily determine the true mass of a particular ion. Therefore multiply charged ions can be sequenced independent of other charged ions and the results may be compared to resolve discrepancies such as gaps in the ion sets.

Gaps in the ion sets are simply missing peaks or peaks that can not be distinguished from other unrelated peaks. A gap in the ion set may be due to several different causes. Instrumental noise, experimental error and overlapping ion sets, to name a few, all create gaps or indistinguishable peaks in mass spectra. Using the data given by the mass spectrum as a whole, the de novo process is generally able to give accurate sequences for each mass spectrum.

The final complexity of de novo sequencing that will be discussed is not a complexity of mass spectrometry but rather of biology. Post translational modifications (PTM) are modifications to proteins that may create a change in conformation of the protein. PTMs are present in nature and cause proteins to activate or deactivate their function. Most PTMs are caused by the addition of molecules to the amino acids of the protein. One common PTM is the addition of a phosphate group to an amino acid which is known as phosphorylation.

Investigators working in MS need to be aware of PTMs because the addition of molecules to amino acids changes the masses of the amino acids. When dealing with a modified protein in MS, the set of amino acids grows. For example, when dealing with a protein that may be phosphorylated, the usual set of 20 amino acids needs to be taken into account along with phosphoserine, phosphothreonine and phosphotyrosine. Another complexity that comes with PTMs is indistinguishable amino acid masses. An example of indistinguishable amino acid masses is a PTM named oxidized methionine (147.04da), which happens to be close to the mass of phenylalanine (147.07da). A mass spectrometer may or may not have the mass accuracy to deal with these overlaps and therefore a good de novo sequencer needs to be able to deal with these scenarios.

Difficulties of Hydrogen Deuterium Exchange

Conducting HDX experiments presents several difficulties. One such difficulty is the matching of an unknown pepsin digested peptide fragment to a portion of the known sequence. As described in the background chapter, pepsin protease cleaves proteins at amino acid locations specific to the protein (as oppose to cleaving rules determined for all proteins, like trypsin). Therefore, the amino acid composition of the resulting peptides is unknown. Before a hydrogen deuterium exchange experiment can be run, all the peptides and their resulting spectrum must be mapped to the known sequence. This is so hydrogen deuterium exchange can be resolved to a specific peptide in the protein. The complications of de novo sequencing are inherited by this first difficulty of analyzing HDX data because matching peptides to spectra involves partial de novo sequencing.

Another difficulty is determining the rates of exchange of hydrogen to deuterium from the data collected in an HDX experiment. This process is done by taking the weighted average of the isotopic cluster of the subject peptide at a given time point and determining the average number of deuterium molecules for that peptide. Since HDX experiments obtain data for several time points, one is able to plot the average number of deuterium molecules for a peptide as a function of time. The graph is then fitted to an exponential decay curve:

$$Y = N - \mathrm{Sum}(D \exp(k^*t))$$

Equation 5.1: where N is the monoisotopic mass of the peptide with no deuterium molecules, D is the number of deuterium molecules exchanged at time t, and k is the rate of exchange. [Ho03]

Once the data plot is fitted to an exponential decay curve, the rate can then be compared to the solution exposed theoretical rate for the peptide which is derived by summing the theoretical rates of each amino acid in the peptide as described by Bia. [Ba93] If the experimental rate is slower than the theoretical rate, conclusions can be drawn about the structural significance of the peptide in the protein.

Determining site resolution (ie. determining which amino acid on the peptide has exchange) of hydrogen exchange has been elusive for many current MS HDX experiments. Tandem MS experiments theoretically will give site resolution due to the fragmentation process but 100 per cent scrambling of deuterium molecules has been observed due to the high energy nature of collision induced dissociation (CID). [Jo04] It is thought that site resolution can be obtained using more advanced fragmentation techniques such as electron capture dissociation (ECD) [ZuK98] or electron transfer dissociation (ETD) [Sy04].

Previous Work

No prior work has been specifically directed at solving the problem of matching unknown pepsin digested peptide fragments to their corresponding spectra. However, there are tools currently available that can be used to aid in the matching process. One tool discussed previously in the literature review chapter is Mascot. Mascot uses a probability based scoring method to rank matches from a protein database search. It can be arranged for Mascot to only search a database of one entry. The one entry would be of the protein sequence which is the subject of the HDX experiment. Since Mascot uses a probability based scoring method, to receive accurate results one is dependent on a large number of entries in the database. With only one entry in the database, the final scoring is essentially meaningless. Mascot will however return a sequence coverage of the protein sequence which becomes useful in order to determine which spectra matched which portion of the sequence.

More work has been done on discovering the rates of hydrogen exchange but results are limited to samples run on FTMS. AutoHD [PaB01] is a hydrogen exchange analysis tool/algorithm for analysis of Fourier transform ion cyclotron resonance mass spectrometers (FTMS). This algorithm first compares isotopic clusters in mass spectra using fast Fourier transform to calculated isotopic distributions to identify peptides. The algorithm then determines the extent of deuterium incorporation of the peptide by comparing theoretical isotopic distributions to experimentally observed isotopic distributions. It was shown that this algorithm works efficiently on FTMS HDX experiments.

Chapter VI: SpectralMatch algorithm

Described in this chapter is an algorithm which attempts to solve the problem of matching mass spectra to their corresponding peptide sequence in the given protein. The SpectralMatch algorithm contains four modules, FindSeqTags, LocateSeqTagInSequence, ComputeSequenceDerivedSpectrum and SpectralMatch. The FindSeqTags module analyzes an input experimentally derived spectrum to determine possible amino acid sequence tags. The sequence tags are then used to locate a portion (or multiple portions) of the input sequence that matches the sequence tag in LocateSeqTagInSequence. Once a sequence portion has been found, the ComputeSequenceDerivedSpectrum module computes a theoretical spectrum based on the sequence portion. This is done for each sequence portion found. Finally, the SpectralMatch module compares the theoretical spectrum and the input experimental spectrum and returns a score. The score is compared to the scores obtained throughout the process and returned to the user. The running time of the algorithm is also discussed.

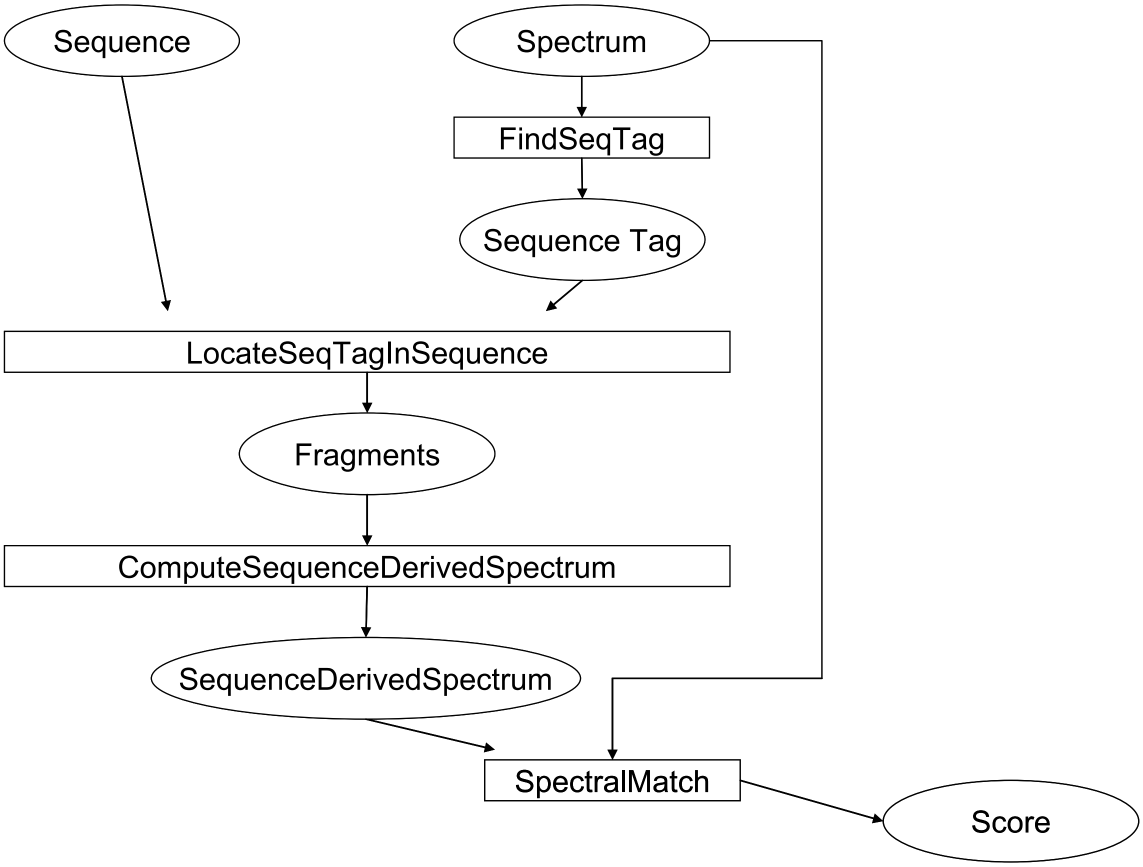


Figure 6.1: Flow of the SpectralMatch algorithm. Rectangles are modules and ovals are inputs/outputs of the modules.

FindSeqTags

The FindSeqTags algorithm searches an experimental spectrum in order to identify ions, specifically the b2 ion, yn-2 ion and the y1 ion. The result of the algorithm is to determine a short sequence of amino acids, a sequence tag, that is present in an experimental spectrum. This algorithm stops short of full de novo sequencing and concludes with a sequence tag rather than the full sequence.

First, given an experimental spectrum (derived from the lab), find all the ions that are prospect singly charged b2 ions. This is done by looking at all the ions in the range from GG to WW (G and W are the one letter codes for Glycine and Tryptophan, respectively), for each ion in the range, compare the mass to charge value to a table constructed of all possible two amino acid combination masses plus the mass of the proton added for type b ions, Equation 6.1.

b2 ion = a + b + Mass(proton)

Equation 6.1: a and b are the masses of any amino acid in the set of amino acids. If the ion's m/z value matches a table entry it is considered as a prospect b2 ion and all amino acid combinations are recorded for that entry.

Next, using the prospect b2 ions, find the compliment yn-2 ion. This can be calculated by the formula y(i) = Mzp – b(n-i) described earlier, where Mzp is the m/z of the precursor mass of the peptide represented by the spectrum.

In the next stage of finding sequence tags, the y1 ion is sought after. The y1 ion uncovers the C-terminus of a peptide sequence. Finding the y1 ion is done by searching through the peak list in the range of one amino acid mass plus the mass of H20 and the mass of a proton for the offset of a y ion. The search can be limited by the bounds laid out in Equation 6.2.

Mass(G) + Mass(H) + Mass(HO) + Mass(Charge)
< = y1
< = Mass(W) + Mass(H) + Mass(HO) + Mass(Charge)

Equation 6.2: Bounds for the y1 ion. G is glysine, W is tryptophan, H is hydrogen, O is oxygen and charge is the number of protons that ionized the fragment ion.

When a match of a y1 ion is found, it is recorded.

Finally, a gap figure is determined. A gap figure is the amount of mass units between known amino acids. The gap figure in FindSeqTags is the precursor mass minus the masses of the first two amino acids and minus the mass of the last amino acid. This is assuming ions were determined for the first two amino acids and the last amino acid. For example, a final sequence tag may look like Figure 6.2.

E L [840.514100] K

Figure 6.2: Example sequence tag. E, L and K are amino acids determined and [840.514100] is the gap between amino acid L and K

A b2-ion coupled with a yn-2 ion, a y1 ion and a gap figure are what makes up a sequence tag for our purposes. It should be noted here that this algorithm ignores peptide-cleaving enzyme rules. It is possible to limit the possible C-term amino acids by introducing cleaving rules. This algorithm is intended for all enzymes but more specifically pepsin under specific environmental conditions, as discussed above.

LocateSeqTagInSequence:

Once a sequence tag or multiple sequence tags have been found for the experimental spectrum, the next step is to find every instance of the sequence tag in the given known sequence. This is done by scanning the sequence for the first two amino acids in the sequence tag. After an instance of the first two amino acids is found in the sequence, the last amino acid in the sequence tag is scanned for until a condition is met.

The condition is based on the gap figure given by the sequence tag and the masses of the amino acids in between the instance of the first two amino acids and the instance of the last amino acid found in the sequence. If an instance of the last amino acid is found in the sequence, the sum of the masses of the amino acids in between the first two instances of amino acids found and the instance of the last amino acid found is compared to the gap figure. If the gap is greater than the sum of masses, the instance of the last amino acid is treated like other intermediate amino acids in the sequence and the scan for another instance of the last amino acid is continued.

If the gap is less than the sum of masses, the scan for the last amino acid is aborted and the instance of the first two amino acids is ignored. Finally, if the gap is equal to the sum of masses (within some accuracy figure), the portion of the sequence, from the first two amino acid to the last amino acid, is recorded in a list. The search for the first two amino acids begins again at the next amino acid in the sequence from the first matched amino acid. Each portion of the sequence that is found to match the sequence tag is recorded.

SeqTag: E L [840.514100] K

Sequence: C* Y {{E L V I S L I V E S K}} Mo P {E L} C* G Y {{E L V I L S V I S E K}} Mo P

Figure 6.3: Example of locating a sequence tag in a sequence

As you can see in this example, Figure 6.3, the sequence tag is made up of three amino acids, E, L and K and a gap figure, 840.5141. The amino acid substrings, enclosed in double curly brackets, are matches to the sequence tag. If added up, the amino acids between V and S (inclusive) in the first instance, and the amino acids between V and E in the second instance, will equal the gap figure, 840.5141, plus or minus some accuracy figure given. There is also an extra instance of the first two amino acids in the sequence tag enclosed in single curly brackets. When this instance was found, a search for the last amino acid was aborted after the gap figure was surpassed by the sum of the following amino acids.

ComputeSequenceDerivedSpectrum

Computing the theoretical spectrum from an input sequence involves using the formulas described above in Chapter II, namely Equations 2.1 and 2.2. The procedure starts with computing the b2 ion. As noted in earlier chapters, the b1 ion will not be present in spectrum of a peptide because of chemical reasons. The mass of the third amino acid is added to the b2 ion to compute the b3 ion. This process is repeated until the n-1 amino acid. A similar procedure is used to determine the y ions.

Since we are computing a theoretical spectrum, we need to give the ions computed intensity. Currently there is no reasonable way to determine the intensity by sequence alone. Many other algorithms that compute theoretical spectrum, such as the one used in Sequest [EnM94], normalize all ions. This algorithm does the same. A more sophisticated theoretical spectrum compute algorithm may include other ions such as water loss and immonium ions. In this case, ion intensities of lesser importance may be assigned lower values than the normalized value.

SpectralMatch

The final step in matching the correct portion of a known sequence to an experimental spectrum is to compare the experimental spectrum to the theoretical spectrum derived from the sequence. This is done by searching the experimental spectrum for the peaks in the theoretical spectrum. When searching for a peak, a match is determined within some error figure given. When multiple peaks are found within the given error figure, the peak with the largest intensity is the one selected as a match. Once a match has been determined for a peak, the intensity of the peak found is added to a global summation score. The raw score is then normalized to a percentage of total ion intensity for the whole spectrum and returned to the user.

Running Time

The FindSeqTags module attempts to find all b2 ions, the corresponding yn-2 ions and y1 ions. To find all possible b2 ions, the algorithm traverses the spectrum once and finding the corresponding yn-2 ion can be done in constant time. To find all possible y1 ions, the algorithm traverses the spectrum once. Each y1 ion found is combined with the b2 ions found previously. Since there is a constant number of possible b2 ions, this step takes constant time. Therefore the runtime of this module is O(n), where n is the number of peaks in the given spectrum.

The LocateSeqTagInSequence module attempts to find every instance of a sequence tag in a given protein sequence. Finding each instance of the first two amino acids in the given sequence takes m steps, where m is the number of amino acids in the given sequence. Once an instance of the first two amino acids is found, it then takes m steps to locate the last amino acid and verify whether the instance is valid according to the gap figure. Since it takes m steps to find each instance and for each instance it takes m steps to verify its validity, the runtime of this module is O(m^2).

The ComputeSequenceDerivedSpectrum module computes the theoretical spectrum from the sequence found in the LocateSeqTagInSequence module. This process takes m steps, where m is the number of amino acids in the sequence. Each peak in the b ion set and y ion set of the resulting spectrum corresponds to one amino acid of the sequence. The runtime of this module is O(m).

The SpectralMatch module which matches an experimental spectrum to a theoretical spectrum involves traversing each spectrum once. The experimental spectrum as described earlier is of size n. The theoretical spectrum is bounded by the size of the input sequence, m. The runtime of this module is O(n + m).

Combining all the modules together into a full functional unit, FindSeqTags and LocateSeqTagInSequence are both run once. ComputeSequenceDerivedSpectrum is run at most m times, where m is the largest amount of output from LocateSeqTagInSequence. Subsequently SpectralMatch is also run at most m times. The total running time of the full SpectralMatch algorithm is O((n + m)*m) which can be rewritten as O(n*m + m^2), where n is the number of peaks in the spectrum, m is the number of amino acids in the given sequence.

Chapter VII: HDX Rates algorithm

This chapter describes an algorithm which solves the problem of computing the number of hydrogen molecules which have exchanged with deuterium molecules for a given spectrum at a given timepoint. The HDX Rates algorithm contains four modules, ComputeIsotopicDistribution, ComputeMaxExchangeableHydrogens, ComputeTheoreticalDeuteratedPeaks, and ComputeHDXRate. The first module, ComputeIsotopicDistribution, computes the theoretical isotopic distribution for a given charged peptide inputted by the user. Next, ComputeMaxExchangeableHydrogens calculates the total number of possible hydrogen molecules that may undergo exchange of a given peptide sequence inputted by the user. Next, the max exchangeable hydrogen molecules value, the monoisotopic m/z value and a given charge are used to determine the theoretical m/z values of the deuterated peaks in ComputeTheoreticalDeuteratedPeaks. Finally, ComputeHDXRate computes the average number of deuterium molecules present in a given spectrum. A deuterium count is computed for all inputted spectra and can be graphed according to the retention time of the spectrum. The running time of the algorithm is also discussed.

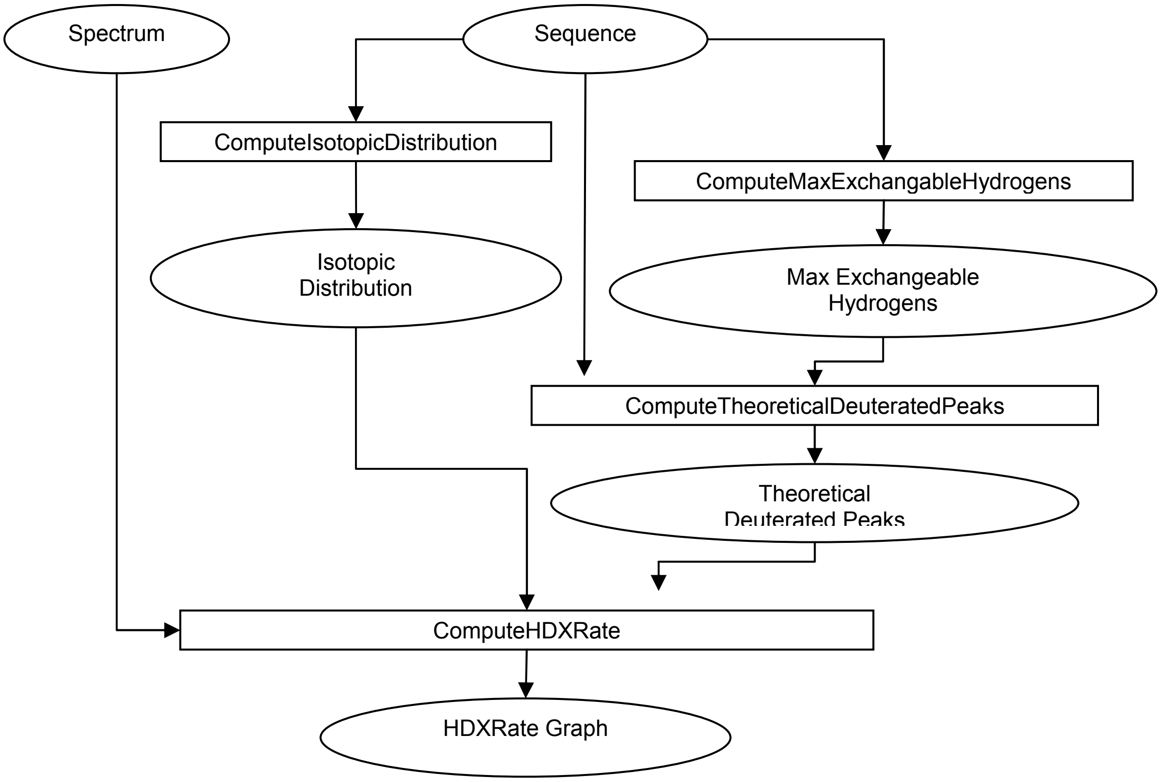

Spectrum
Sequence
ComputeIsotopicDistribution
ComputeMaxExchangableHydrogens
Isotopic Distribution
Max Exchangeable Hydrogens
ComputeTheoreticalDeuteratedPeaks
Theoretical Deuterated Peaks
ComputeHDXRate
HDXRate Graph

Figure 7.1: Flow of HDX Rates algorithm. Rectangles are modules and ovals are inputs and outputs.

ComputeIsotopicDistribution

The ComputeIsotopicDistribution algorithm is modeled off the Kubinyi algorithm which computes isotopic distributions for small molecular compounds. [Ku91] The purpose of the ComputeIsotopicDistribution algorithm is to determine the isotopic distribution of a given peptide sequence. A table of pre-calculated isotopic distributions of each of the twenty amino acids is used to construct isotopic distributions of a whole peptide. The pre-calculated table also includes the isotopic distributions of multiple amino acids combined. This aids in the efficiency of computing the isotopic distributions of peptides by reducing the number of sub-isotopic distributions needed to combine.

Once the pre-calculated table is computed, a peptide sequence is divided up into groups for which there are entries in the table. For example, if the pre-calculated table has the isotopic distributions of all twenty amino acids as well as combinations of two amino acid combination isotopic distributions, a peptide sequence of "AAGWTLK" may be divided into "AA", "GW", "TL" and "K", each of which has an entry in the table. Another improvement for efficiency is to group amino acids together so that pre-computed isotopic distributions can be used to build up to a final solution. For example, a sequence of "ABACABACABACABA" can be broken down to 8 A's, 4 B's and 3 C's. This is because the order of the computation does not matter to the final isotopic distribution. The 8 A's can be computed by first computing the isotopic distribution of 2 A's. That result can be used to compute the isotopic distribution of 4 A's and so on and so forth until the final isotopic distribution is completed.

The sub-sequences are then combined in the fashion described by Kubinyi. Given two isotopic distributions, for each peak in the first sub-sequence isotopic distribution, the mass value is added to every mass value of the second sub-sequence isotopic distribution and the abundance value is multiplied by every abundance value of the second sub-sequence isotopic distribution. This results in a third isotopic distribution which is a convolution of the first two. This step is repeated until all of the sub-isotopic distributions are convoluted into one isotopic distribution.

After the combination of two isotopic distributions, the peaks are then subject to a resolution check and a precision check. The resolution check combines all peaks that are within a certain value of each other. Since this tool is used for mass spectrometry data, the resolution value may be the same value as the mass resolution of the specific mass spectrometer. The precision check eliminates the peaks with abundances lower than a given precision value. The precision check gives the ability to ignore small abundance values for very large peptides and therefore reduce computation time. The resulting peaks' intensities in the isotopic distribution are a percentage of the highest peak.

ComputeMaxExchangeableHydrogens

The ComputeMaxExchangeableHydrogens algorithm gives a bound on how many hydrogen atoms may exchange with deuterium atoms in an HDX experiment. Given a peptide sequence, the max exchangeable hydrogen count (MEH) is the number of backbone amide hydrogen atoms in the sequence. [HX05] Every amino acid except Proline has an exchangeable amide hydrogen, as described in Equation 7.1. Also there is no exchangeable amide hydrogen on the C-terminal end of the peptide.

$$MEH = num(AA) - num(Ps) - C\text{-}terminalAA$$

Equation 7.1: The max exchangeable hydrogen molecules formula. AA is the number of amino acids in the given sequence, Ps is the number of Prolines in the given sequence and C-terminalAA is equal to one.

ComputeTheoreticalDeuteratedPeaks

The ComputeTheoreticalDeuteratedPeaks algorithm determines a set of theoretical peaks which are used to locate deuterated peaks in the experimental spectrum. The inputted values are the monoisotopic m/z value, a charge value and the max exchangeable hydrogen molecules (MEH). First, the difference of the mass of a deuterium and the mass of a hydrogen (diffHD) is calculated. A count, numOfDeut, is kept from zero to MEH. The numOfDeut is multiplied by diffHD, divided by the given charge and added to the monoisotopic m/z value to determine the m/z value of a deuterated peak. These m/z values are recorded as the theoretical deuterated peaks.

The process of determining m/z values of the theoretical deuterated peaks is done just once for the entire run of the HDX experiment. The intensities of these peaks, however, are determined by the experimental spectra and are done separately for each spectrum. Using the theoretical m/z value and an accuracy value, intensity is determined by taking the weighted averaging of all the peaks in the experimental spectrum within a range of theoretical m/z plus or minus the accuracy value.

ComputeHDXRate

The ComputeHDXRate algorithm determines the average number of deuterium molecules present in a given spectrum. The intensities determined in the ComputeTheoreticalDeuteratedPeaks module undergo a deconvolution process to remove the isotopic distribution contribution of the sequence. This is in order to compare just the mono-isotopic peaks of varying deuterated content with no isotopic contribution. The isotope contribution must therefore be subtracted from neighboring peaks to obtain accurate results.

To remove the isotopic contribution from peaks in a spectrum, every peak in the spectrum (SPeak) from the monoisotopic peak with no deuterium content (MI) to the monoisotopic peak plus MEHs (MI+MEH) is looped over. Every peak (i) in the isotopic distribution is then multiplied by SPeak in order to obtain the contribution of the isotopes. The contribution is then subtracted from the spectrum peak at position SPeak + i.

Once the peaks in the spectrum are deconvoluted, the weighted average of the peaks from the MI to MI+MEH is taken to determine the average m/z value of all the peaks. The weighted average can be formulated by Equation 7.2.

$$WA = Sum[0\text{->}n]\ (i.m/z * i.intensity) / Sum[0\text{->}n]\ (i.intensity)$$

Equation 7.2: WA is the weighted average which is in units of m/z. The Sum of all deuterated peak's m/z values multiplied by deuterated peak's intensity value. The top term gives weight to peaks with large intensities. Finally the figure is divided by the Sum of all the peaks' intensities.

The weighted average of the deuterated peaks can then be subtracted by the monoisotopic peak to give the average deuterium molecules present in the subject peaks. This value is returned to the user for each spectrum inputted. The average deuterium molecules for each spectrum can then be graphed as a function of time of when the spectrum was computed by the mass spectrometer.

Running Time

The ComputeIsotopicDistribution module calculates the isotopic distribution of a given peptide by combining pre-calculated sub-isotopic distributions. Discovering which pre-calculated sub-isotopic distributions to combine takes m steps, where m is the length of the peptide sequence. The combination of each sub-isotopic distribution can be done in p^2 steps where p is the number of peaks in each sub-isotopic distributions. The relationship of the number of peaks in the isotopic distribution of the whole peptide is described in Equation 7.3. The value of p grows linearly with m.

$$p = m * C – (m – 1)$$

Equation 7.3: m is the number of amino acids in the given peptide, C is a constant value representing the number of peaks in the first sub-isotopic distribution and p is the number of peaks in the final isotopic distribution.

There are log(m) sub-isotopic distributions that need to be combined for each group of amino acids, where again m is the count of amino acids in a group. As described in section ComputeIsotopicDistribution of this chapter, there are only log(m) sub-isotopic distributions to combine because the isotopic distribution of m amino acids can be calculated from combining two sub-isotopic distributions of sqrt(m). Equation 7.4 shows a recurrence relation for the number of steps it takes to calculate the full isotopic distribution of a peptide with m amino acids. The recurrence relation in Equation 7.4 solves to O(m^2).

T(m) = T(m/2) + (2^lg(m) * C – 2^lg(m) + 1)^2

Equation 7.4: m is the number of amino acids in the given peptide, C is a constant value representing the number of peaks in the first sub-isotopic distribution and T(m) is the recurrence relation.

The ComputeMaxExchangeableHydrogens module calculates the maximum amount of exchangeable amide hydrogen molecules in a given peptide. The sequence must be traversed once to determine the number of Prolines in the sequence. As discussed in the ComputeMaxExchangeableHydrogens section, Proline amino acids do not have an exchangeable amide hydrogen and need to be subtracted from the length of the peptide. Therefore the runtime of this module is O(m), where m is the number of amino acids in the given sequence.

The ComputeTheoreticalDeuteratedPeaks module computes the theoretical mass over charge values telling where deuterated peaks will be present in an experimental spectrum. The runtime of this module is O(k) where k is the max exchangeable hydrogen molecules, which is also the number of deuterated peaks. The max exchangeable hydrogen molecules value is bounded by the length of the peptide sequence and therefore O(k) <= O(m).

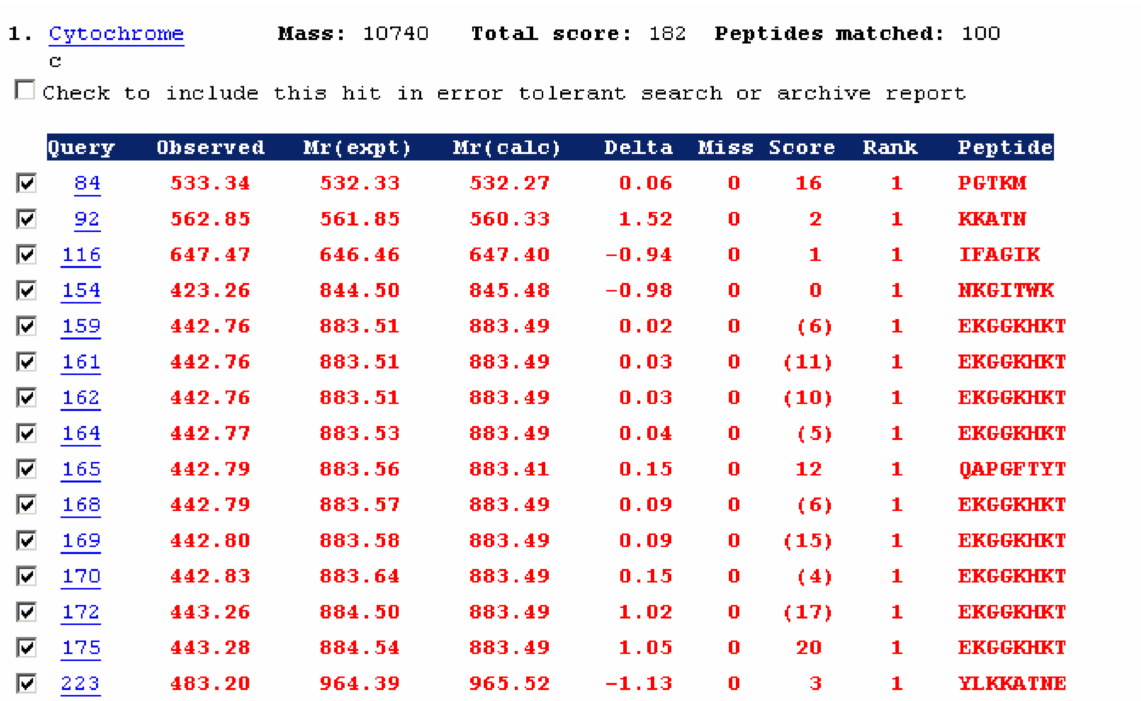
The ComputeHDXRate module deconvolutes the experimental spectrum from the peptide’s isotopic distribution and calculates the final deuterium content. Deconvolution requires p steps to deconvolute n peaks, where n is the number of deuterated peaks in the experimental spectrum and p is the number of peaks in the peptide’s isotopic distribution. The value p can be obtained using Equation 7.3. Since the value of p grows linearly with m, p can be replaced by m in the running time of this module. The weighted average requires n steps, where n is the number of deuterated peaks. The deuterium content can then be calculated in constant time using the value of the weighted average. The runtime of this module is O(n*m).

Combining all of the HDXRates modules into one functional unit, the final running time is O(m^2 + n*m), where m is the number of amino acids in the input sequence and n is the number of peaks in the number of deuterated peaks in the experimental spectrum.

Chapter VIII: Results and Conclusions

SpectralMatch vs. Mascot

As described earlier in the Previous Work section of Chapter V, the database search tool, Mascot, can be used to match pepsin digested peptide spectra to their corresponding portion of the protein sequence. A sample of the protein cytochrome-c was digested with pepsin and ran through a quadrupole time-of-flight mass spectrometer. The resulting data was then analyzed by Mascot against a database of one entry, the cytochrome-c sequence, Figure 8.5. The default parameters were used except for adjusting the digest enzyme to “no-enzyme”. The “no-enzyme” option for the digest enzyme allows Mascot to examine every possible cleavage of the protein sequence in order to compare it to the subject spectrum. The Mascot results returned showed 100 peptide hits out of 1136 spectra. Since the search was only searching one sequence and Mascot determines scores based on the size of the database, the scores returned were relatively meaningless.



1. Cytochrome c **Mass:** 10740 **Total score:** 182 **Peptides matched:** 100

□ Check to include this hit in error tolerant search or archive report

	Query	Observed	Mr(expt)	Mr(calc)	Delta	Miss	Score	Rank	Peptide
☑	84	533.34	532.33	532.27	0.06	0	16	1	PGTKM
☑	92	562.85	561.85	560.33	1.52	0	2	1	KKATN
☑	116	647.47	646.46	647.40	-0.94	0	1	1	IFAGIK
☑	154	423.26	844.50	845.48	-0.98	0	0	1	NKGITWK
☑	159	442.76	883.51	883.49	0.02	0	(6)	1	EKGGKHKT
☑	161	442.76	883.51	883.49	0.03	0	(11)	1	EKGGKHKT
☑	162	442.76	883.51	883.49	0.03	0	(10)	1	EKGGKHKT
☑	164	442.77	883.53	883.49	0.04	0	(5)	1	EKGGKHKT
☑	165	442.79	883.56	883.41	0.15	0	12	1	QAPGFTYT
☑	168	442.79	883.57	883.49	0.09	0	(6)	1	EKGGKHKT
☑	169	442.80	883.58	883.49	0.09	0	(15)	1	EKGGKHKT
☑	170	442.83	883.64	883.49	0.15	0	(4)	1	EKGGKHKT
☑	172	443.26	884.50	883.49	1.02	0	(17)	1	EKGGKHKT
☑	175	443.28	884.54	883.49	1.05	0	20	1	EKGGKHKT
☑	223	483.20	964.39	965.52	-1.13	0	3	1	YLKKATNE

Figure 8.1: Partial Mascot results run on MS data from cytochrome-c digested with pepsin protease.

Search title : Cytochrome C pepsin digest
MS data file : pepsinCytoC.dta
Database : CytC (1 sequences; 95 residues)
Type of search : MS/MS Ion Search
Enzyme : None
Mass values : Monoisotopic
Protein Mass : Unrestricted
Peptide Mass Tolerance : ± 2 Da
Fragment Mass Tolerance: ± 0.8 Da
Max Missed Cleavages : 1
Instrument type : Default
Data File Name : pepsinCytoC.dta
Number of queries : 1136

Figure 8.2: General parameters for Mascot run on MS data from cytochrome-c

The same data set was analyzed with SpectralMatch against the same Cytochrome-c sequence in Figure 8.5. SpectralMatch returned 76 probable peptide matches and 330 possible peptide matches out of the 1136 total spectra. A probable peptide match is a score which is above a given threshold for the given data set. The probable peptide matches for this data set were scores above 3 per cent and possible peptide matches were scores above 2 per cent.

Peptide Sequence	Peptide Mass (Daltons)	Peptide Sequence Position	SpectralMatch Score(s)	Mascot Hit
GDVEKGKKIF	1120.63663	1-10		
KGGKHKTGPNLHGLF	1590.88687	22-36	1.95%	
YLENPKKYIPGTKM	1681.89874	67-80	6.7%	X
YLENPKKYIPGTKMIF	1942.05122	67-82		X
IFAGIKKKTEREDL	1647.94338	81-94		
IFAGIKKKTEREDLIA	1832.06456	81-96	9.23%	X
AGIKKKTEREDL	1387.7909	83-94	1.97%	
AGIKKKTEREDLIA	1571.91208	83-96		
AYLKKATNE	1037.56313	95-104		X

Table 8.1: Known peptides produced from Cytochrome-c digested with pepsin. [Zh93] Results of SpectralMatch and Mascot are also shown next to each of the peptides.

Table 8.1 shows results of Mascot and SpectralMatch on peptides which are known to be produced from Cytochrome-c digested with pepsin. [Zh93] Two of the scores on SpectralMatch are very probable using the mentioned threshold. The other two scores are slightly below the possible peptide match threshold but still worth taking note of their presence.

Comparing the SpectralMatch output to the Mascot output, there were only 11 hits that were in both sets of output. Example output of two spectra run through an implementation of SpectralMatch is shown in the following Figures, 8.3 and 8.4. Both experimental spectra were reviewed by a mass spectrometry specialist to verify the top matches and both were deemed accurate.

```
Datafile: CHYMO1.561.4.0502.0503.3.dta
----------------------------------------------------------------------------
INFO (spectralMatch): Welcome to findSeqTags

SeqTag: (Y L) [1255.915300] M
SeqTag score: 0.577558 %
1 instances of SeqTag located

    Sequence:  Y L E N P K K Y I P G T K M
    Score of spectralMatch: 6.023102 %

SeqTag: (Y L) [1255.990500] M
SeqTag score: 1.402640 %
1 instances of SeqTag located

    Sequence:  Y L E N P K K Y I P G T K M
    Score of spectralMatch: 6.023102 %

SeqTag: (W T) [1262.061700] N
SeqTag score: 0.330033 %
1 instances of SeqTag located

    Sequence:  T W K E E T L M E Y L E N
```

```
        Score of spectralMatch: 1.732673 %

SeqTag: (R G) [1336.075100] N
SeqTag score: 0.412541 %
1 instances of SeqTag located

        Sequence:  G R K T G Q A P G F T Y T D A N
        Score of spectralMatch: 1.485149 %

 Total number of matches: 4
------------------------------------------------------------------------------
```

Figure 8.3: SpectralMatch returned 4 matches for the spectrum given in the data file. The matches with a spectralMatch score of > 6 per cent is a relatively high score and in fact YLENPKKYIPGTKM is the sequence for the given spectrum confirmed by a mass spectrometry specialist.

```
Datafile: CHYMO1.611.4.0464.0465.3.dta
------------------------------------------------------------------------------
INFO (spectralMatch): Welcome to findSeqTags

SeqTag: (F I) [1482.001100] A
SeqTag score: 3.030303 %
1 instances of SeqTag located

        Sequence:  I F A G I K K K T E R E D L I A
        Score of spectralMatch: 6.565657 %

 Total number of matches: 1
------------------------------------------------------------------------------
```

Figure 8.4: SpectralMatch returned 1 match for the spectrum given in this particular data file. Again the match is > 6 per cent and the sequence IFAGIKKKTEREDLIA was confirmed by a mass spectrometry specialist.

G D V E K G K K I F V Q K C A Q C H T V E K G G K H K T G P N L H G L F G R K T G Q A P G
F T Y T D A N K N K G I T W K E E T L M E Y L E N P K K Y I P G T K M I F A G I K K K T E R E
D L I A Y L K K A T N E

Figure 8.5: This is the amino acid sequence of the protein horse heart cytochrome c as presented by the NCBI website. [NC05]

HDX Rates vs. theoretical results

The HDX Rates algorithm was tested with a peptide of length 9 amino acids and a data set of spectra which were the results of a hydrogen deuterium exchange experiment on the 9 amino acid peptide. The sequence of the peptide is “I H N V K H K G W”. The peptide was first exposed to a solution of D20, deuterium water in which exchange was allowed to take place. The sample was then exposed to a solution with conditions of pH 2.8 and temperature of 1.1 degC which allowed back-exchange to occur, ie. deuterium molecules on peptide exchange back with hydrogen molecules in surrounding solution. This will be considered time 0 seconds. The solution with conditions of low pH and low temperature has a known relationship with the back exchange reaction and therefore it can be determined the rate at which the back exchange occurred. These rates are shown in Table 8.3 for varying temperatures. The sample was then entered into the mass spectrometer to be analyzed at time 16 seconds. At time 78 seconds, the first spectrum was recorded by the detector.

It should be noted here that the temperature of the peptide sample rose to 21degC or around room temperature and back exchange occurred at a faster rate. This was mainly due to the un-insulated 550mm capillary tube which transferred the sample from the HPLC to the electrospray needle. Once the experiment was complete, 30 spectra were then selected to be inputs into an implementation of the HDX Rates algorithm. Figures 8.7 and 8.8 were among the selected spectra. Figure 8.6 shows what a fully deuterated spectrum looks like.

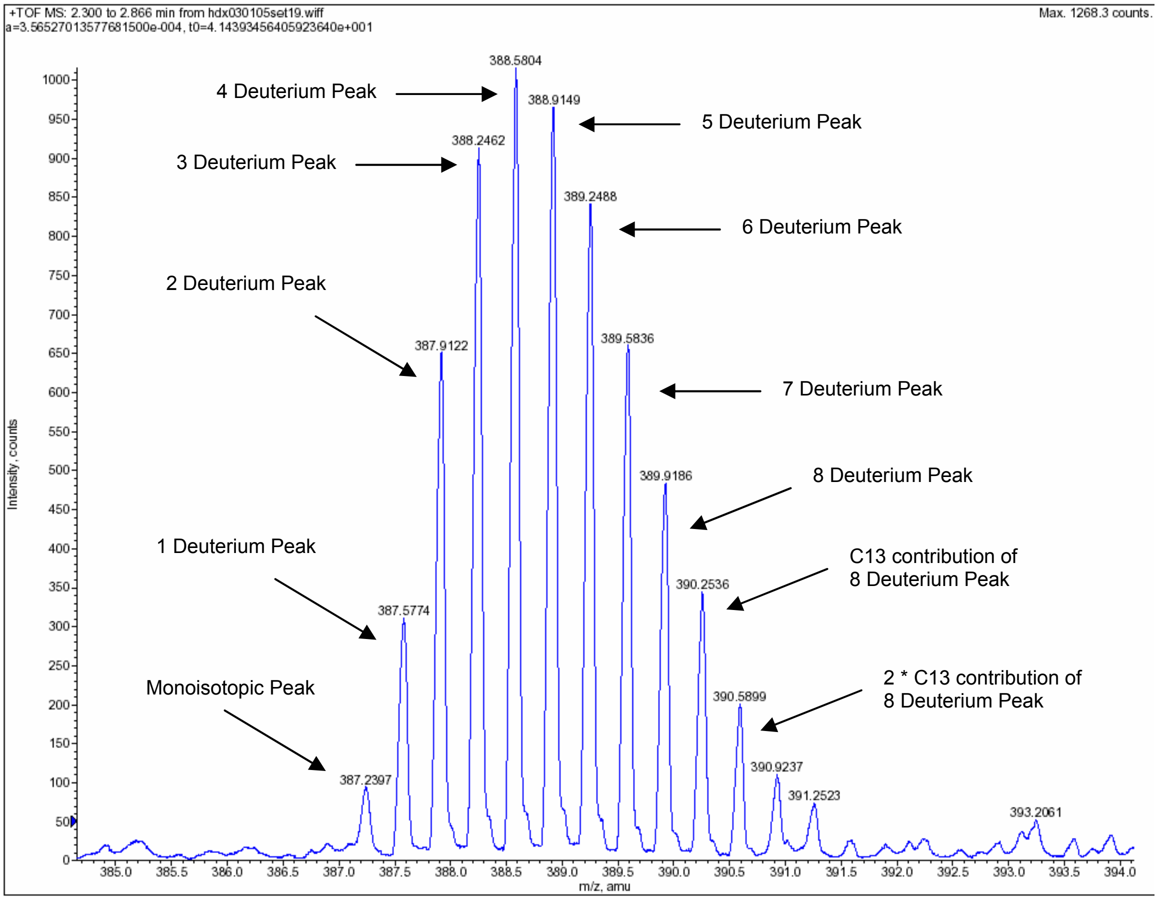


Figure 8.6: A fully deuterated spectrum of the peptide "I H N V K H K G W" showing the maximum number of deuterium molecules present, 8. This spectrum was acquired while injecting the peptide sample with D2O and not allowing back exchange. Each of the deuterated peaks labeled have a contribution from previous peaks isotopic pattern. For example, the 1 Deuterium Peak is a combination of the 1 Deuterium Peak and the C13 peak of the monoisotopic peak.

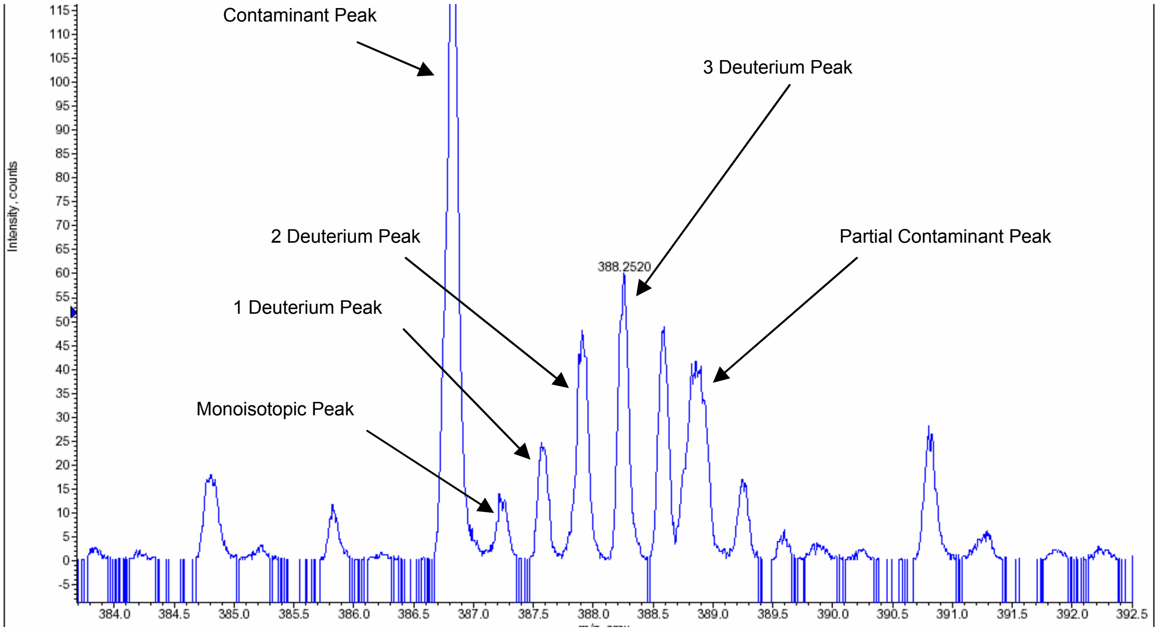


Figure 8.7: Partially deuterated peptide spectrum. This spectrum was acquired early in the back exchange experiment, ~78 seconds.

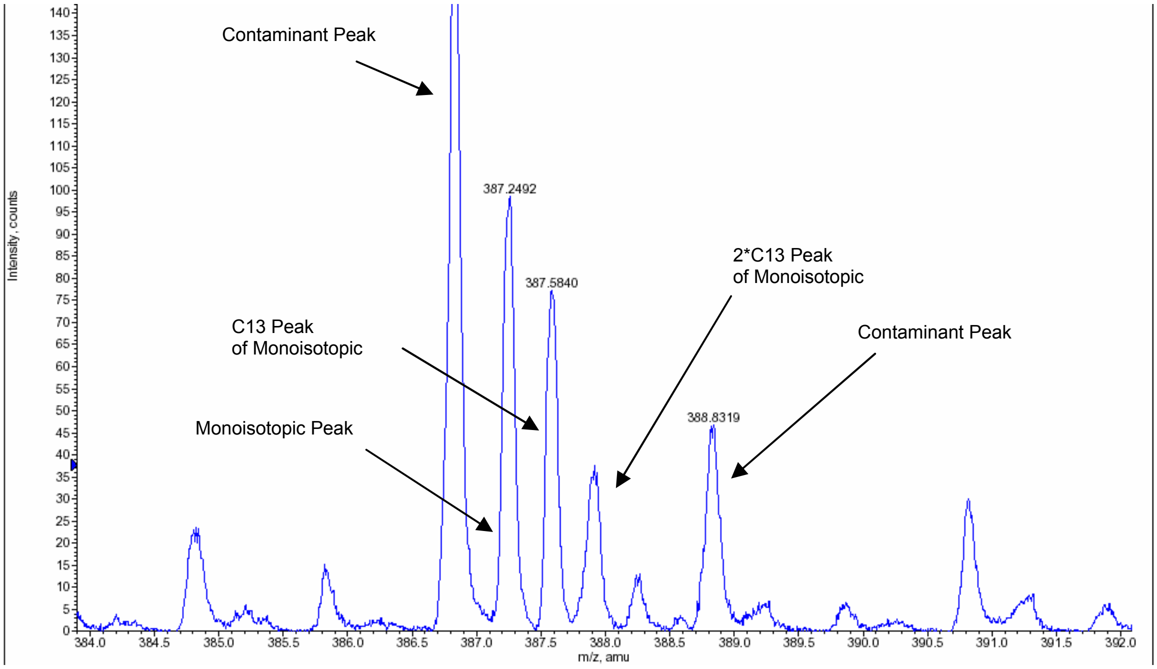


Figure 8.8: Peptide spectrum with no deuterium. This spectrum was acquired late in the back exchange experiment, ~1400 seconds.

The output of the HDX Rates implementation was then graphed, Table 8.2, Figure 8.10, a data point for each spectrum, according to the amount of time between the start of the back-exchange reaction to when the mass spectrometer recorded the spectrum. The observed data, Table 8.2, was fitted to the exponential decay equation in Figure 8.9.

$$f(x) = a * \exp(-k * t)$$

Figure 8.9: Exponential decay equation, a is the initial quantity, k is the rate and t is time.

The data fitting was done using Gnuplot's fit function which uses the least squares algorithm. [GP05] The observed rate of exchange, k, for the peptide is 0.005348 deuterium molecules per second.

The curve in the graph, Figure 8.10, was adjusted slightly by 1.04 deuterium molecules due to Mass Spectrometry noise. Evidence of this noise is apparent in the data points, Table 8.2, where at the final time points the deuterium count is above zero. Theoretically, all back exchange would have occurred by the final time points and the deuterium count should be zero.

The graph and resulting rate in Figure 8.10 can then be compared to the exponential decay graph of the predicted theoretical rate laid out by Bia. [Ba93] Table 8.3 is output from a program SPHERE [ZY05] which predicts the theoretical rates of exchange for each amino acid in a given peptide. Values are shown for 4degC, 10degC and 20degC all with a pH of 2.8. Figure 8.11 shows a side by side comparison of three exponential decay curves. The graph shows the observed exponential decay curve is bounded within the theoretical exponential decay curves of 20degC and 4degC. Again the curves are adjusted by 1.04 deuterium molecules.

In most experiments, the theoretical rate and the observed rate will be different due to structural features the subject peptide is involved with in the protein. The presented experiment was involving only one peptide with no structural features and therefore the observed rate should be bounded by the two theoretical rates.

Time (seconds)	Average Number of Deuterium molecules
0	8.0 (artificial data point)
78	2.875905
139.8	2.292047
145.8	2.148964
222	1.977205
295.98	1.672362
371.94	1.46423
445.92	1.395056
527.94	1.314119
595.92	1.220413
753.9	1.066446
821.88	1.16889
897.9	1.049438
971.88	1.107738
1131.84	0.909951
1197.84	0.883899
1273.86	0.964789
1353.84	1.068621
1423.8	0.993386
1497.84	0.896388
1645.8	1.058058
1799.76	0.977794
1945.74	1.049783

Table 8.2: Output data from HDX Rates implementation for 9 amino acid peptide. The time element in this table is relative to the time the mass spectrometer recorded the spectrum.

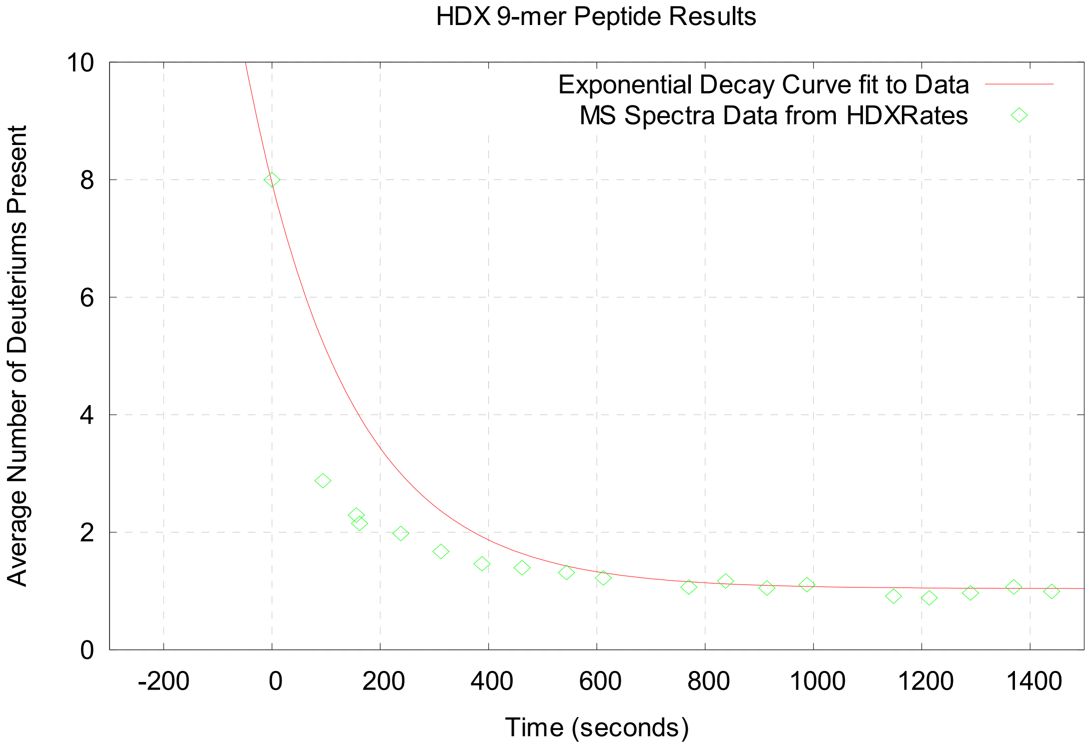


Figure 8.10: This graph shows plotted data (green) from the HDX Rates implementation. The data was fitted to an exponential decay curve (red) to compute a rate of decay of 0.005348 deuterium molecules per second. The first data point is artificial to simulate fully deuterated conditions. The subject peptide is known to have eight exchangeable deuterium molecules from previous experiments.

	IH	HN	NV	VK	KH	HK	KG	GW
4degC	0.728E-02	0.112E-02	0.178E-03	0.189E-03	0.537E-03	0.438E-03	0.289E-03	0.275E-03
10degC	0.144E-01	0.217E-02	0.302E-03	0.325E-03	0.102E-02	0.820E-03	0.519E-03	0.473E-03
20degC	0.414E-01	0.608E-02	0.696E-03	0.762E-03	0.276E-02	0.219E-02	0.131E-02	0.111E-02

Table 8.3: Theoretical exchange rates for each amide hydrogen outputted from the Sphere program which uses chemical properties of amino acids to determine theoretical exchange rates of peptides laid out by Bia. [B+, ZY05] The are shown for temperature at 4degC, 10degC and 20degC, and with a consistent pH of 2.8.

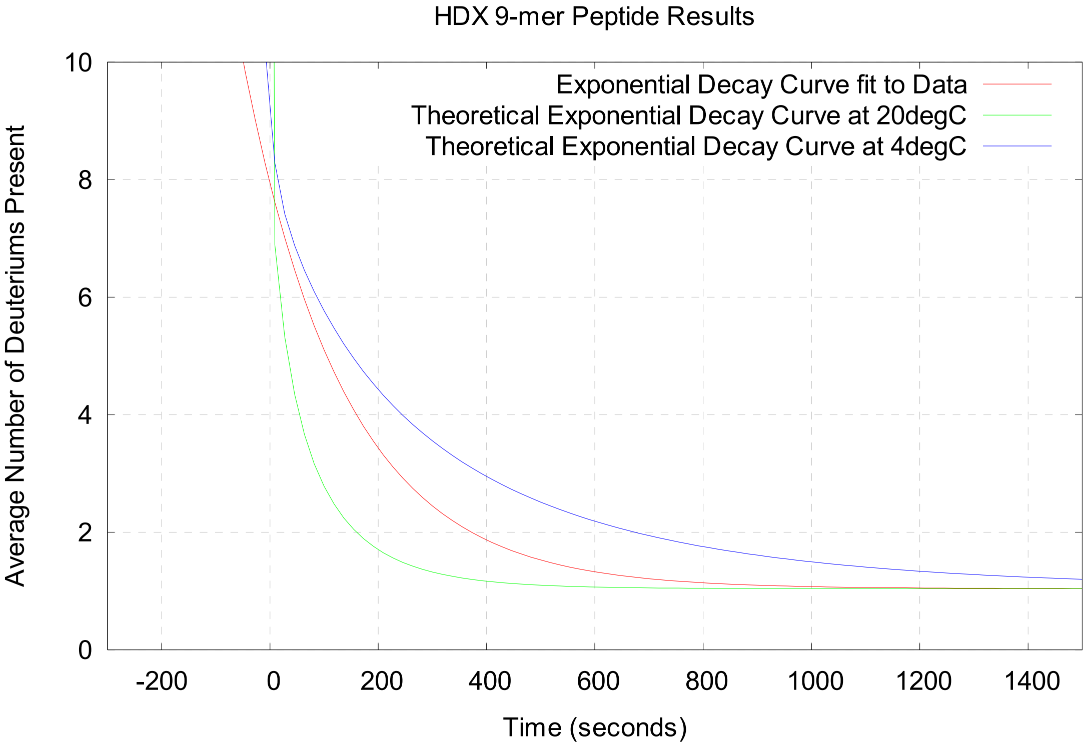


Figure 8.11: This graph shows a comparison between the observed rate of exchange (red), the theoretical rate of exchange at 20degC (green) and the theoretical rate of exchange at 4degC (blue).

Spectral Match Time Trials

The SpectralMatch implementation was tested on a Pentium IV 3.2 GHz machine with 2Gb of RAM running Debian with kernel 2.4. It was tested with several different spectra of varying sizes. The accuracy value was set to 0.5 and the protein sequence given was 104 amino acids long. Table 8.4 shows the results of the time trials. The resulting number of matches for each spectrum is also shown. It is observed that each run requires less than a second to run for even the largest spectrum.

Size of Spectrum	120	282	653	1111	3188	6002	9113	12552
Num of Matches	1	1	0	0	0	2	1	0
Run 1	0.032	0.06	0.061	0.063	0.141	0.385	0.56	0.468
Run 2	0.032	0.06	0.063	0.062	0.139	0.4	0.542	0.451
Run 3	0.031	0.06	0.061	0.062	0.139	0.382	0.55	0.466
Run 4	0.033	0.06	0.061	0.062	0.139	0.403	0.535	0.454
Run 5	0.031	0.061	0.06	0.071	0.141	0.389	0.555	0.471
Run 6	0.031	0.06	0.061	0.064	0.141	0.393	0.531	0.46
Run 7	0.033	0.06	0.061	0.062	0.141	0.384	0.555	0.465
Run 8	0.032	0.06	0.061	0.061	0.139	0.393	0.534	0.451
Run 9	0.032	0.06	0.065	0.063	0.14	0.381	0.553	0.458
Run 10	0.032	0.061	0.061	0.061	0.14	0.389	0.53	0.458
Average (sec)	**0.0319**	**0.0602**	**0.0615**	**0.0631**	**0.14**	**0.3899**	**0.5445**	**0.4602**

Table 8.4: Results of SpectralMatch implementation with spectra of varying sizes as input. Average of ten runs for each spectrum is given on the last line.

HDX Rates Time Trials

The HDXRates implementation was tested on a Pentium IV 3.2 GHz machine with 2Gb of RAM running Debian with kernel 2.4. It was tested with several different spectra of varying sizes. The accuracy value was set to 0.5, precision value was set to 0.0000001, resolution value was set to 0.05, charge was set to 3 and the protein sequence given was 9 amino acids long. Table 8.5 shows the results of the time trials. It is observed that each run requires less than a second to run for even the largest spectrum.

Size of Spectrum	29292	31405	32082	34441	35384	37938	42587	44945	48529	50596	52320	52916
Run 1	0.075	0.08	0.083	0.086	0.092	0.092	0.102	0.105	0.112	0.116	0.12	0.12
Run 2	0.116	0.122	0.125	0.131	0.134	0.144	0.157	0.168	0.115	0.182	0.121	0.19
Run 3	0.116	0.122	0.124	0.132	0.135	0.144	0.156	0.163	0.177	0.181	0.185	0.189
Run 4	0.115	0.121	0.125	0.132	0.134	0.142	0.117	0.163	0.111	0.162	0.187	0.188
Run 5	0.117	0.123	0.125	0.133	0.134	0.143	0.101	0.164	0.176	0.183	0.187	0.186
Run 6	0.116	0.122	0.124	0.13	0.135	0.143	0.157	0.165	0.174	0.182	0.186	0.19
Run 7	0.115	0.122	0.126	0.132	0.134	0.141	0.158	0.165	0.175	0.18	0.186	0.188
Run 8	0.117	0.123	0.125	0.133	0.134	0.142	0.156	0.164	0.177	0.182	0.188	0.19
Run 9	0.115	0.123	0.124	0.132	0.135	0.142	0.157	0.163	0.176	0.182	0.186	0.188
Run 10	0.116	0.08	0.123	0.131	0.135	0.142	0.102	0.163	0.175	0.116	0.185	0.185
Average	**0.1118**	**0.1138**	**0.1204**	**0.1272**	**0.1302**	**0.1375**	**0.1363**	**0.1583**	**0.1568**	**0.1666**	**0.1731**	**0.1814**

Table 8.5: Results of HDXRates implementation with spectra of varying sizes as input. Average of ten runs for each spectrum is given on the last line.

Conclusions

As the results above show, SpectralMatch is a complement rather than a replacement for Mascot and other protein database search engines. It is thought that since the implementation of SpectralMatch is heavily favored to singly charged ions, that a mass spectrometer with a MALDI ionization method would fair even better results. This is due to MALDI ionization generally only produces singly charged peptide ions.

SpectralMatch will be offered as open source software, freely downloadable and customizable to adjust to the requirements of a specific lab or experiment. SpectralMatch is also part of a larger proteomics framework developed by the Illinois Bio-Grid which its modules can be used to build larger more complex proteomic analysis software.

The HDX Rate algorithm is shown to work for HDX experiments on peptides using a tandem mass spectrometer. Like SpectralMatch, the implementation of the HDX Rates algorithm was built in a modular fashion in order to be part of a larger proteomics framework. Each module was developed with general usage in mind so that code could be reused in separate, unrelated proteomic software tools or could be used as the base of more complex software.

Future work

Future work for SpectralMatch involves creating a more accurate and flexible analysis which allows for adjustment of weighting scoring parameters. Other enhancements to the algorithm will include the incorporation of predicting intensity data from machine and sample parameters. [Zh04b] This will allow a large advantage over protein database searches which generally use only mass data to determine spectrum-peptide matches.

Future work for the HDX Rate algorithm will be to incorporate the fast Fourier transform algorithm into determining isotopic distributions. [PaB01] Also, the implementation will be improved by adding seamless graphing capabilities to allow users to fit and view exchange rate graphs immediately after analysis.

Appendix I: References

[Al04] Alberts, Bruce; Bray, Dennis; Hopkin, Karen; Johnson, Alexander; Lewis, Julian; Raff, Martin; Roberts, Keith; Walter, Peter. Essential Cell Biology, second edition. Garland Science Taylor & Francis Group, 2004.

[Ba93] Bai, Yawen; Milne, John S.; Mayne, Leland; Englader, S. Walter. (1993) Primary Structure Effects on Peptide Group Hydrogen Exchange. PROTEINS: Structure, Function, and Genetics 17:75-86 (1993)

[Be04] Dennis A. Benson, Ilene Karsch-Mizrachi, David J. Lipman, James Ostell, and David L. Wheeler. GenBank: update. Nucleic Acids Res. 2004 January 1; 32(Database isfsue): D23–D26.

[BL05] Basic Local Alignment Search Tool (BLAST). Web Page. Available from http://www.ncbi.nlm.nih.gov/blast. Accessed August 1, 2005.

[BO02] Baerga-Ortiz, Abel; Hughes, Carrie A.; Mandell, Jeffrey G.; Komives, Elizabeth A. (2002) Epitope mapping of a monoclonal antibody against human thrombin by H/D-exchange mass spectrometry reveals selection of a diverse sequence in a highly conserved protein. Protein Science, 2002, 11:1300-1308.

[Ch01] Chen, T; Kao, MY; Tepel, M; Rush, J; Church, GM. (2001) A dynamic programming approach to de novo peptide sequencing via tandem mass spectrometry. J. Comput Biol. 2001; 8 (3): 325-37.

[Cl99] Clauser K. R., Baker P. R. and Burlingame A. L., Role of accurate mass measurement (+/- 10 ppm) in protein identification strategies employing MS or MS/MS and database searching. Analytical Chemistry, Vol. 71, 14, 2871- (1999)

[Da99] Dancik, Vlado; Addona, Theresa A.; Clauser, Karl R.; Vath, James E.; Pevzner, Pavel A. (1999) De Novo Peptide Sequencing via Tandem Mass Spectrometry. Journal of Computational Biology. Vol. 6. Num. 3/4. 327-342. 1999

[De98] Deng, Y. H.; Smith, D.L. (1998) Identification of unfolding domains in large proteins by their unfolding rates. Biochemistry. 1998. 37:6256-62

[Du03] Du, Zhaohui; Rempel, Don; Fremont, David; Miley, Mike; Gross, Michael. 2003. Study of the Binding of MHC Class I Molecule and Antigenic Peptide Complex by H/D Exchange and HPLC MS/MS. Dept. Chemistry. Washington University, St. Louis.

[En01] Englander, S. Walter; Krishna, Mallela M. G. Hydrogen exchange. Nature Structural Biology. Vol. 8, Number 9, September 2001

[En00] Englander, S. Walter. (2000) Protein folding intermediates and pathways studied by hydrogen exchange. Annu. Rev. Biophys. Biomol. Struct. 2000. 29:213-38

[En96] Englander, S.Walter; Sosnick, TobinR.; Englander, J.J.; Mayne, L. (1996) Mechanisms and uses of hydrogen exchange. Curr. Opin. Struct. Biol. 1996. 6:18-23

[EnM94] Eng, Jimmy K.; McCormack, Ashley L.; Yates, John R. III. 1994. An Approach to Correlate Tandem Mass Spectral Data of Peptides with Amino Acid Sequences in a Protein Database. J Am Soc Mass Spectrom 1994, 5, 976-989

[Ey00] Eyles, S.J.; Speir, J.P.; Kruppa, G.H.; Gierasch, L.M.; Kaltashov, I.A. (2000) Protein conformational stability probed by Fourier transform ion cyclotron resonance mass spectrometry. J. Am. Chem. Soc. 2000. 122:495-500

[Pe88] W. R. Pearson and D. J. Lipman (1988), Improved Tools for Biological Sequence Analysis, PNAS 85:2444-2448

[FC95] Fernandez-de-Cossio, J. et al. (1995) A computer program to aid the sequencing of peptides in collision-activated decomposition experiments. Comput. Appl. Biosci. 11, 427-434

[Fr04] Frank, Ari; Pevzner, Pavel. (2004) PepNovo: De Novo Peptide Sequencing via Probabilistic Network Modeling. American Chemical Society.

[GP05] GNUPlot. Web Page. Available from http://www.gnuplot.info/. Accessed August 3, 2005.

[Gl03] Glish, Gary L.; Vachet, Richard W. 2003. The Basics of Mass Spectrometry In The Twenty-First Century. Nature, Vol. 2, 140-150, February 2003

[Ha01] Hansen, Beau T.; Jones, Juliet A.; Mason, Daniel E.; Liebler, Daniel C. (2001) SALSA: A Pattern Recognition Algorithm To Detect Electrophile-Adducted Peptides by Automated Evaluation of CID Spectra in LC-MS-MS Analyses. Anal. Chem. 2001, 73, 1676-1683

[HaH03] Havilio, Moshe; Haddad, Yariv; Smilansky, Zeev. (2003) Intensity-Based Statistical Scorer for Tandem Mass Spectrometry. Anal. Chem. 2003, 75, 435-444

[Ho03] Hoofnagle, Andrew N.; Resing, Katheryn A.; Ahn, Natalie G. 2003. Protein Analysis By Hydrogen Exchange Mass Spectrometry. Annu. Rev. Biophys. Biomol. Struct. 2003. 32:1-25

[HX05] HXMS.com. Web Page. Available from http://www.hxms.com. Accessed August 3, 2005.

[Jo04] Jorgensen, Thomas J. D.; Gardsvoll, Henrik; Ploug, Michael; Roepstorff, Peter. Intramolecular Migration of Amide Hydrogens in Protonated Peptides upon Collisional Activation. (2004) Journal of American Chemical Society.

[Ki00] Kinter, Michael; Sherman, Nicholas E. 2000. Protein Sequencing and Identification Using Tandem Mass Spectrometry. John Wiley & Sons, Inc. 2000

[Ko05] Kolch, Walter; Mischak, Harald; Pitt, Andrew R. The molecular make-up of a tumor: proteomics in cancer research. Clinical Science (2005) 108, 369-383

[Ku91] Kubinyi, Hugo; 1991. Calculation of isotope distributions in mass spectrometry. A trivial solution for a non-trivial problem. Anal. Chim. Acta 247, 107-119 (1991)

[Kw03] Kwon, Sung Won; Kim, Sung Chan; Jaunbergs, Janis; Falck, John R.; Zhao, Yingming. (2003) Selective Enrichment of Thiophosphorylated Polypeptides as a Tool for the Analysis of Protein Phosphorylation. Molecular & Cellular Proteomics 2:242-247, 2003.

[Lu04] Lu, Bingwen; Chen, Ting. (2004) Algorithms for de novo peptide sequencing using tandem mass spectrometry. Biosilico, Vol. 2, No. 2 March 2004

[Lu03] Lu, Bingwen; Chen, Ting. (2003) A Suboptimal Algorithm for De Novo Peptide Sequencing via Tandem Mass Spectrometry. Journal of Computational Biology. Volume 10, Number 1, 2003

[Li02] Liebler, Daniel C. Introduction to Proteomics, Tools for the New Biology. Humana Press. Totowa, NJ. 2002.

[Ma98] Mandell, Jeffrey G.; Falick, Arnold M.; Komives, Elizabeth A. (1998) Identification of protein-protein interfaces by decreased amide proton solvent accessibility. Proc. Natl. Acad. Sci. USA Vol. 95, pp. 14705-14710, December 1998 Biophysics.

[NC05] National Center for Biotechnology Information (NCBI) HomePage. Web Page. Available from: http://www.ncbi.nlm.nih.gov. Accessed August 1, 2005.

[Pa05] Parker, Laurie L.; Schilling, Alexander B.; Kron, Stephen J.; Kent, Stephen B. H. (2005) Using MALDI-TOF-MS detection to optimize thiophosphorylation in the presence of competing phosphorylation.

[PaB01] Palmblad, Magnus; Buijs, Jos; Hakansson, Per. Automatic Analysis of Hydrogen/Deuterium Exchange Mass Spectra of Peptides and Proteins Using Calculations of Isotopic Distributions. (2001) American Society for Mass Spectrometry. 2001, 12, 1153-1163.

[PaH93] Pappin DJ, Hojrup P, Bleasby AJ. Rapid identification of proteins by peptide-mass fingerprinting. Curr Biol. 1993 Jun 1;3(6):327-32.

[Pe99] Perkins, David N.; Pappin, Darryl J.; Creasy, David M.; Cottrell, John S. (1999) Probability-based protein identification by searching sequence databases using mass spectrometry data. Electrophoresis. 1999, 20, 3551-3567

[Ro94] Robinson, C.V.; Gross, M.; Eyles, S.J.; Ewbank, J.J.; Mayhew, M.; et al. (1994) Conformation of GroEL-bound alpha-lactalbumin probed by mass spectrometry. 1994. Nature 372:646-51

[SmD97] Smith, David L.; Deng, Yuzhong; Zhang, Zhongqi. 1997. Probing the Non-covalent Structure of Proteins by Amide Hydrogen Exchange Mass Spectrometry. Journal of Mass Spectrometry, Vol. 32, 135-146 (1997)

[SmM00] Smyth, M.S.; Martin, J.H.J. X Ray Crystallography. J. Clin. Pathol: Mol. Pathol. 2000; 53:8-14

[St04] Steen, Hanno; Mann, Matthias. The ABC's (and XYZ's) of Peptide Sequencing. Nature Reviews. Molecular Cell Biology. Vol. 5. September 2004

[SP05] UniProtKB/Swiss-Prot DATABASE. Web Page. Available at http://www.ebi.ac.uk/swissprot/. Accessed August 3, 2005.

[Sy04] Syka, John E. P.; Coon, Joshua J; Schroeder, Melanie J; Shabanowitz, Jeffrey; Hunt, Donald F. Peptide and protein sequence analysis by electron transfer dissociation mass spectrometry. PNAS June 29, 2004 vol. 101 no. 26 9528–9533

[Ta97] Taylor, Alex J.; Johnson, Richard S. (1997) Sequence Database Searches via de Novo Peptide Sequencing by Tandem Mass Spectrometry. Rapid Communications in Mass Spectrometry. Vol. 11. 1067-1075. 1997

[Tr05] UniProtKB/TrEMBL DATABASE. Web Page. Available at http://www.ebi.ac.uk/trembl/. Accessed August 3, 2005.

[Ty03] Tyers, Mike; Mann, Matthias. From genomics to proteomics. Nature. Vol 422. 13 March 2003

[Ya02] Yamada, N.; Suzuki, E.; Hirayama, K. (2002) Identification of the interface of a large protein-protein complex using H/D exchange and Fourier transform ion cyclotron resonance mass spectrometry. Rapid Commun. Mass Spectrom. 2002.16:293-99

[Zh04a] Zhang, Zhongqi. (2004) De Novo Peptide Sequencing Based on a Divide-and-Conquer Algorithm and Peptide Tandem Spectrum Simulation. American Chemical Society.

[Zh04b] Zhang, Zhongqi. (2004) Prediction of Low-Energy Collision-Induced Dissociation Spectra of Peptides. Anal. Chem. 2004, 76, 3908-3922.

[Zh95] Zhang, Zhongqi; Post, Carol Beth; Smith, David L. 1995. Amide Hydrogen Exchange Determined by Mass Spectrometry: Application to Rabbit Muscle Aldolase. Biochemistry 1996, 35, 779-791

[Zh93] Zhang, Zhongqi; Smith, David L. 1993. Determination of amide hydrogen exchange by mass spectrometry: A new tool for protein structure elucidation. Protein Science (1993), 2, 522-531 Cambridge University Press.

[ZuK98] Zubarev, R.; Kelleher, N.; McLafferty, F. (1998) Electron capture dissociation of multiply charged protein cations. A nonergodic process. J. Am. Chem. Soc. 1998. 120, 3265-3266

[ZY05] Yu-Zhu Zhang, Protein and peptide structure and interactions studied by hydrogen exchange and NMR. *Ph.D. Thesis*, Structural Biology and Molecular Biophysics, University of Pennsylvania, PA, USA. Available at http://www.fccc.edu/research/labs/roder/sphere/. Accessed August 19, 2005.

Appendix II: Documentation

SpectralMatch User Interface (version 0.3)

SpectralMatch is a utility for mapping tandem peptide mass spectra to a given protein sequence. The user interface described is for command line Unix systems. Functionality of the utility will be explained along with the input parameters. Examples of output, definitions of errors and warnings and where to find more help will also be discussed.

SpectralMatch fFunctionality

The purpose of the user interface of the SpectralMatch utility is to allow users to easily provide input, change runtime parameters and receive text output of the results to and from the SpectralMatch engine. The utility is run as most executable programs in UNIX are run. On the command line, the path to the executable is specified and inputs follow. The common inputs to SpectralMatch are the mass spectra and the protein sequence to map the spectra against.

When the SpectralMatch utility is executed, the inputs and parameters are parsed and displayed to the screen. Parameters with the default settings are marked as "`(default)`". Each input spectrum is then evaluated against the given protein sequence and given a score to how well the spectrum matches a portion of the protein sequence.

The results for each input spectrum is displayed in two parts. First the sequence tag(s) found in the spectrum are displayed. A score is also displayed after each sequence tag that shows the percentage of total ion intensity covered by the matched peaks in the input spectrum. The second part of the results is the portion of the peptide sequence the spectrum matched, the match. The match follows the sequence tag from which it was derived. Also the score of the match is displayed which is also the percentage of the total ion intensity covered.

Finally, at the end of each input spectrum, the total number of matches is displayed. If there were no matches found for the given input spectrum, zero(0) is displayed. Also, if there were no sequence tags found, "`No sequence tags found`" is displayed.

SpectralMatch Parameters

Parameters can be declared on the command line using the short form, beginning with a '`-`' then the parameter's single letter form or the long form, beginning with a '`--`' then the parameter's long form. Parameters can also be declared through a parameter file with the format "`<parameter_name> = <parameter_val>`" with each parameter declared on a new line and comment lines beginning with "`#`".

A user can specify five distinct parameters. The final two parameters for specifying the protein sequence are mutually exclusive. The parameters with descriptions are as follows:

- `accuracy_value` is the value for which a peak matches another peak. The command line short form is specified by '`-a`' and the long form is specified by '`—-accuracy_value=<double>`'. SpectralMatch uses this value when deciding whether a peak, prospect peak is in range of another peak, target peak. If the prospect peak is within plus or minus the accuracy_value of the target peak, a peak match is recorded. The default is set to 0.5 which means that there is a window of one dalton for matching peaks.

- `threshold_value` is the value for determining whether a peak should be considered when finding possible sequence tags. The command line short form is specified by '`-t`' and the long form is specified by '`—-threshold_value=<double>`'. The value is in terms of a percent of the largest peak in the spectrum. The default is set to 0.0 which means that all peaks in the spectrum are considered when finding possible sequence tags.

- `init_file` is the filename of the initialization file which contains the mass values of single and multiple residue combinations. The command line short form is specified by
‘`-i`’ and the long form is specified by ‘`—-init_file=<filename>`’. The format of the file is of the form:

```
(id#) (Residue Mass) (Charge) (Mass/Charge) (NULL) (NULL) (AA1) (AA2)
(AA3)
```

Each mass value is entered on a new line. The entries are space separated with a unique id number followed by the residue mass, the charge of the entry, the mass to charge ratio. The next two fields are unprocessed and are meant for future use. The final three fields are to identify the single letter abbreviations of the amino acid which makeup the residue or residue combination. The default value for the init_file is set to initFiles/DisambuableList2.csv which comes packaged with the distribution.

- `dta_filename` is the filename of the dta formatted file which holds the experimental spectrum. The command line short form is specified by ‘`-d`’ and the long form is specified by ‘`—-dta_filename=<filename>`’. Input spectra may also be passed to the program as unnamed parameter on the command line, meaning the ‘`-d`’ or
‘`—-dta_filename=<filename>`’ is unnecessary. The default value for dta_filename is dtaFiles/ELVISLIVESK.dta which comes packaged with the distribution. The file is for demo purposes only.

- `sequence` is a string representation of a protein sequence. The command line short form is specified by ‘`-s`’ and the long form is specified by ‘`—-sequence=<STRING>`’. SpectralMatch matches the given spectrum to this sequence in attempt to map the spectrum to the sequence. The format of the string should be single letter abbreviations of amino acids and should be space separated. Also this parameter should only be used for short protein sequences. For longer protein sequences, a fasta formatted file may be inputted and is mutually exclusive to the `fasta_sequence` parameter.

- `fasta_sequence` is the filename of the fasta formatted file which holds a protein sequence. The command line short form is specified by ‘`-f`’ and the long form is specified by ‘`—-fasta_sequence=<filename>`’. SpectralMatch matches the given spectrum to this sequence in attempt to map the spectrum to the sequence. The fasta_sequence parameter is mutually exclusive to the `sequence` parameter. The default value for fasta_sequence is fastaFiles/ELVISLIVESK.fsa which comes packaged with the distribution. The file is for demo purposes only.

SpectralMatch Examples

Demo Output (no parameters, no input files)

```
user@localhost:~$ ./spectralMatch

Welcome to SpectralMatch, a utility for mapping tandem mass spectra to
a given protein sequence
        accuracy_value is 0.500000 (default)
        threshold_value is 0.000000 (default)
        init_file is initFiles/DisambuableList2.csv (default)
        using fastaFiles/ELVISLIVESK.fsa (default)
        using dtaFiles/ELVISLIVESK.dta (default)

results for file dtaFiles/ELVISLIVESK.dta:

 (E L) [840.514100] K   7.745182 %

         L E I V S I L V S E K   45.535740 %

         E L V I S L I V E S K   55.783824 %

 Total number of matches: 2
```

Real Output (command line parameters, multiple input files)

```
user@localhost:~$ ./spectralMatch dtaFiles/CHYMO1*.dta -a 0.5 -f
fastaFiles/CytochromeC.fsa

Welcome to SpectralMatch, a utility for mapping tandem mass spectra to
a given protein sequence
        accuracy_value is 0.500000
        threshold_value is 0.000000 (default)
        init_file is initFiles/DisambuableList2.csv (default)
        fasta_file is fastaFiles/CytochromeC.fsa

results for file dtaFiles/CHYMO1.561.4.0502.0503.3.dta:

 (Y L) [1255.990500] M  1.402640 %

         Y L E N P K K Y I P G T K M     34.653465 %

 Total number of matches: 1

results for file dtaFiles/CHYMO1.611.4.0464.0465.3.dta:

 (F I) [1482.001100] A  3.030303 %

         I F A G I K K K T E R E D L I A         34.343434 %

 Total number of matches: 1
```

Real Output (parameter file, single input file)

```
user@localhost:~$ ./spectralMatch -p parFiles/test_CytochromeC.par
Welcome to SpectralMatch, a utility for mapping tandem mass spectra to
a given protein sequence
        accuracy_value is 0.500000
        threshold_value is 0.000000
        init_file is initFiles/DisambuableList2.csv (default)
        fasta_file is fastaFiles/CytochromeC.fsa

results for file dtaFiles/CHYMO1.561.4.0502.0503.3.dta:

 (Y L) [1255.990500] M  1.402640 %

         Y L E N P K K Y I P G T K M     34.653465 %

 Total number of matches: 1
```

SpectralMatch Errors and Warnings

Several error and warning messages may appear while executing the SpectralMatch utility. Their definitions and descriptions are as follows:

- `spectralMatch: Error opening configuration file 'nofile.par'`. The parameter file could not be found or opened. This is because the parameter filename specified is misspelled, missing or not at the path specified. Check the location of the parameter file to make sure it exists and make sure the path is properly specified.

- `2 options of group sequence group were given. At most one is required`. The parameter `fasta_sequence` and `sequence` were both given as parameters to execution. Since `fasta_sequence` and `sequence` are two different ways of inputting the same information, they are mutually exclusive and therefore are not allowed to be specified both at once. Check the parameter file to make sure there does not exist a `sequence` or `fasta_sequence` along with the command line.

- `option requires an argument --` . A parameter which requires an argument is specified but the argument is not specified. Check all of the command line and parameter file parameter specifications and make sure an argument is specified for each parameter that requires an argument.

- `option given more than once` . A parameter is specified multiple times on the command line or in the parameter file. Check for duplicate parameter specifications on the command line and the parameter file.

- `dta_filename: No such file or directory` . The file specified in the `dta_filename` parameter could not be found or opened. This is because the filename specified is misspelled, missing or not at the path specified. Check the location of the file specified to make sure it exists and make sure the path is properly specified.

- `fasta_file: No such file or directory` . The file specified in the `fasta_sequence` parameter could not be found or opened. This is because the filename specified is misspelled, missing or not at the path specified. Check the location of the file specified to make sure it exists and make sure the path is properly specified.

- `init_file: No such file or directory`. The file specified in the `init_file` parameter could not be found or opened. This is because the filename specified is misspelled, missing or not at the path specified. Check the location of the file specified to make sure it exists and make sure the path is properly specified.

- `(warning: accuracy_value is less than zero)`. The accuracy value specified by the user is less than zero. Behavior with this parameter setting is undefined but execution will proceed. It is advised to change this value to a positive value.

- `(warning: threshold_value is less than zero)`. The threshold value specified by the user is less than zero. Behavior with this parameter setting is undefined but execution will proceed. It is advised to change this value to a positive value.

- `(default)` The parameter value was not set by the user but rather is using the default set by the program. This is generally acceptable in a few cases. The user may want to run the demo in which all of the parameters are set to the default values, the user may want to use the default accuracy value and threshold value set by the program or the user may want to use the initialization file provided and distributed with the package. In most cases, outside the demo, the dta_filename and the sequence or fasta_sequence parameters should not be set to the defaults but rather the users own input data.

SpectralMatch Help

Following is the output displayed when SpectralMatch is given the help parameter. More documentation may be found at the Illinois Bio-Grid website: http://gridweb.cti.depaul.edu/twiki/bin/view/IBG/MassSpecToolkit . Also, documentation of the API and libraries produced by doxygen can be found at: http://histone.cti.depaul.edu/~kdrew/docs/. If you still have questions or comments please contact Kevin Drew at kdrew@mcs.anl.gov . He is always happy to answer questions, hear suggested improvements and know how SpectralMatch is helping a researcher.

Output (-h command line option)

```
user@localhost:~$ ./spectralMatch -h
spectralMatch 0.3

Usage: spectralMatch [OPTIONS]... [inputs]...

  -h, --help                    Print help and exit
  -V, --version                 Print version and exit
  -p, --parameter-file=filename Parameter file for run time options

MS Instrument dependent values:
  the following parameters may be set to allow MS instrument dependent
figures
  -a, --accuracy_value=DOUBLE   Accuracy for which to match peaks, ie.
+ or -
                                  accuracy value equals match
(default=`0.5')
  -t, --threshold_value=DOUBLE  Threshold value is a percent of the
largest
                                  peak in the spectrum  (default=`0.0')

Initialization files:
  the following options may be set to allow for different locations of
init
  files
  -i, --init_file=filename      Initialization filename

(default=`initFiles/DisambuableList2.csv')

Run specific parameters:
  the following parameters are used to set the values of run specific
  parameters
  -d, --dta_filename=filename   DTA Filename
                                  (default=`dtaFiles/ELVISLIVESK.dta')

 Group: sequence group
  a sequence is required
  -s, --sequence=STRING         Amino Acid Sequence String (short)
  -f, --fasta_sequence=filename FASTA formatted Amino Acid Sequence
file
```

HDXRates User Interface (version 0.3)

HDXRates is a tool which analyzes MS data from an Hydrogen Deueterium Exchange experiment to compute deuterium content of a peptide. The user interface described is for command line Unix systems. Functionality of the utility will be explained along with the input parameters. Examples of output, definitions of errors and warnings and where to find more help will also be discussed.

HDXRates Functionality

The purpose of the user interface of the HDXRates utility is to allow users to easily provide input, change runtime parameters and receive text output of the results to and from the HDXRates engine. The utility is run as most executable programs in UNIX are run. On the command line, the path to the executable is specified and inputs follow. The common inputs to HDXRates are the raw deuterated mass spectra and the sequence of the peptide represented by the mass spectra.

When the HDXRates utility is executed, the inputs and parameters are parsed and displayed to the screen. Parameters with the default settings are marked as “(default)”. Each input spectrum is then analyzed for deuterium content and a floating point number is displayed as the calculated number of deuteriums on the peptide.

The HDXRates utility can be run on deuterated spectra at different time points during an experiment and the output can be graphed using a graphing package such as *gnuplot*.

HDXRates Parameters

Parameters can be declared on the command line using the short form, beginning with a ‘-‘ then the parameter’s single letter form or the long form, beginning with a ‘--‘ then the parameter’s long form. Parameters can also be declared through a parameter file with the format “<parameter_name> = <parameter_val>” with each parameter declared on a new line and comment lines beginning with “#”.

A user can specify five distinct parameters. The final two parameters for specifying the protein sequence are mutually exclusive. The parameters with descriptions are as follows:

- accuracy_value is the value for which a peak matches another peak. The command line short form is specified by ‘-a’ and the long form is specified by ‘—-accuracy_value=<double>’. HDXRates uses this value when centroiding raw data to a defined peak. If the prospect raw data point is within plus or minus the accuracy_value of the theoretical deuterated target peak, a peak match is recorded. The default is set to 0.5 which means that there is a window of one dalton for matching peaks.

- resolution_value is the value for determining whether a peak should be combined with another peak while computing an isotopic distribution. The command line short form is specified by ‘-r’ and the long form is specified by ‘—-resolution_value=<double>’. The default is set to 0.05 which means that peaks, while computing an isotopic distribution, closer than 0.05 are combined.

- precision_value is the value for determining whether a peak should be deemed insignificant and eliminated while computing an isotopic distribution. The command line short form is specified by ‘-P’ and the long form is specified by ‘—-precision_value=<double>’. The default is set to 0.0000001 which means that peaks with abundance less than 0.0000001will be eliminated from the isotopic distribution

- `charge` is the charge value of the peptide in the mass spectra. The command line short form is specified by '`-c`' and the long form is specified by '`—charge=<INT>`'. There is no default value and it is a required parameter to the program.

- `init_file` is the filename of the initialization file which contains the isotopic mass values and abundances of common elements and amino acids. The command line short form is specified by '`-i`' and the long form is specified by '`—-init_file=<filename>`'. The format of the file is of the form:

  (Name of Element/Amino acid) (# of mass value entries)
  (Mass Value) (percentage abundance)
  (Mass Value) (percentage abundance)
  …
  Element/amino acid entries are separated by a space. The items in an entry are space separated with a unique name of the element or amino acid followed by the number of mass value entries for the element or amino acid. Each mass value entry contains the mass value and the percentage of total abundance. The default value for the init_file is set to initFiles/abundanceTable which comes packaged with the distribution.

- `dta_filename` is the filename of the dta formatted file which holds the raw deuterated spectrum. The command line short form is specified by '`-d`' and the long form is specified by '`—-dta_filename=<filename>`'. Input spectra may also be passed to the program as unnamed parameter on the command line, meaning the '`-d`' or
  '`—-dta_filename=<filename>`' is unnecessary. The default value for dta_filename is dtaFiles/hdx_dtas/30.output.dta which comes packaged with the distribution. The file is for demo purposes only.

- `sequence` is a string representation of a protein sequence. The command line short form is specified by '`-s`' and the long form is specified by '`—-sequence=<STRING>`'. HDXRates uses this sequence to compute an isotopic distribution and compute theoretical deuterated peaks. The format of the string should be single letter abbreviations of amino acids and should be space separated. Also this parameter should only be used for short protein sequences. For longer protein sequences, a fasta formatted file may be inputted and is mutually exclusive to the `fasta_sequence` parameter.

- `fasta_sequence` is the filename of the fasta formatted file which holds a protein sequence. The command line short form is specified by '`-f`' and the long form is specified by '`—-fasta_sequence=<filename>`'. HDXRates uses this sequence to compute an isotopic distribution and compute theoretical deuterated peaks. The fasta_sequence parameter is mutually exclusive to the `sequence` parameter. The default value for fasta_sequence is fastaFiles/hdx.fsa which comes packaged with the distribution. The file is for demo purposes only.

HDXRates Examples

Demo Output (required charge parameter, no input files)

```
user@localhost:~$ ./HDXRates -c 3
Welcome to HDXRates, a utility for determining deuterium content on a
peptide through mass spectrum analysis
        accuracy_value is 0.500000 (default)
        precision_value is 0.000000 (default)
        resolution_value is 0.050000 (default)
        charge is 3
        init_file is initFiles/abundanceTable (default)
        using fastaFiles/hdx.fsa (default)
        using dtaFiles/hdx_dtas/30.output.dta (default)

results for file dtaFiles/hdx_dtas/30.output.dta:
3.433706
```

Real Output (command line parameters, multiple input files)

```
user@localhost:~$./ HDXRates -c 3 -f fastaFiles/hdx.fsa \
dtaFiles/hdx_dtas/*.dta
Welcome to HDXRates, a utility for determining deuterium content on a
peptide through mass spectrum analysis
        accuracy_value is 0.500000 (default)
        precision_value is 0.000000 (default)
        resolution_value is 0.050000 (default)
        charge is 3
        init_file is initFiles/abundanceTable (default)
        using fastaFiles/hdx.fsa (default)

results for file dtaFiles/hdx_dtas/30.output.dta:
3.433706

results for file dtaFiles/hdx_dtas/39.output.dta:
2.793665

results for file dtaFiles/hdx_dtas/70.output.dta:
2.381493

results for file dtaFiles/hdx_dtas/186.output.dta:
1.936172
```

HDXRates Errors and Warnings

Several error and warning messages may appear while executing the SpectralMatch utility. Their definitions and descriptions are as follows:

- `HDXRates: Error opening configuration file 'nofile.par'`. The parameter file could not be found or opened. This is because the parameter filename specified is misspelled, missing or not at the path specified. Check the location of the parameter file to make sure it exists and make sure the path is properly specified.

- `2 options of group sequence group were given. At most one is required`. The parameter `fasta_sequence` and `sequence` were both given as parameters to execution. Since `fasta_sequence` and `sequence` are two different ways of inputting the same information, they are mutually exclusive and therefore are not allowed to be specified both at once. Check the parameter file to make sure there does not exist a `sequence` or `fasta_sequence` along with the command line.

- `option requires an argument --`. A parameter which requires an argument is specified but the argument is not specified. Check all of the command line and parameter file parameter specifications and make sure an argument is specified for each parameter that requires an argument.

- `option given more than once`. A parameter is specified multiple times on the command line or in the parameter file. Check for duplicate parameter specifications on the command line and the parameter file.

- `dta_filename: No such file or directory`. The file specified in the `dta_filename` parameter could not be found or opened. This is because the filename specified is misspelled, missing or not at the path specified. Check the location of the file specified to make sure it exists and make sure the path is properly specified.

- `fasta_file: No such file or directory`. The file specified in the `fasta_sequence` parameter could not be found or opened. This is because the filename specified is misspelled, missing or not at the path specified. Check the location of the file specified to make sure it exists and make sure the path is properly specified.

- `init_file: No such file or directory`. The file specified in the `init_file` parameter could not be found or opened. This is because the filename specified is misspelled, missing or not at the path specified. Check the location of the file specified to make sure it exists and make sure the path is properly specified.

- `(warning: accuracy_value is less than zero)`. The accuracy value specified by the user is less than zero. Behavior with this parameter setting is undefined but execution will proceed. It is advised to change this value to a positive value.

- `(warning: precision is less than zero)`. The precision value specified by the user is less than zero. Behavior with this parameter setting is undefined but execution will proceed. It is advised to change this value to a positive value or zero.

- `(warning: resolution is less than zero)`. The resolution value specified by the user is less than zero. Behavior with this parameter setting is undefined but execution will proceed. It is advised to change this value to a positive value or zero.

- `(default)`. The parameter value was not set by the user but rather is using the default set by the program. This is generally acceptable in a few cases. The user may want to run the demo in which all of the parameters are set to the default values, the user may want to use the default accuracy value, precision value and resolution value set by the program or the user may want to use the initialization file provided and distributed with the package. In most cases, outside the demo, the dta_filename and the sequence or fasta_sequence parameters should not be set to the defaults but rather the users own input data.

- `HDXRates was unable to obtain a valid result` . The program was unable to produce valid output for the given input files. This is caused by certain parameter combinations or when spectra does not contain valid data. Adjusting parameter settings may help in relieving this condition.

HDXRates Help

Following is the output displayed when HDXRates is given the help parameter. More documentation may be found at the Illinois Bio-Grid website: http://gridweb.cti.depaul.edu/twiki/bin/view/IBG/MassSpecToolkit . Also, documentation of the API and libraries produced by doxygen can be found at: http://histone.cti.depaul.edu/~kdrew/docs/. If you still have questions or comments please contact Kevin Drew at kdrew@mcs.anl.gov . He is always happy to answer questions, hear suggested improvements and know how HDXRates is helping a researcher.

Output (-h command line option)

```
user@localhost:~$ ./HDXRates -h
HDXRates 0.3

Usage: HDXRates [OPTIONS]... [inputs]...

  -h, --help                    Print help and exit
  -V, --version                 Print version and exit
  -p, --parameter-file=filename Parameter file for run time options

MS Instrument dependent values:
  the following parameters may be set to allow MS instrument dependent
figures
  -a, --accuracy_value=DOUBLE   Accuracy for which to match peaks, ie.
+ or -
                                  accuracy value equals match
(default=`0.5')
  -r, --resolution_value=DOUBLE Resolution for which to combine peaks,
ie.
                                  peaks less than resolution_value
apart are
                                  combined  (default=`0.05')
  -P, --precision_value=DOUBLE  Precision for which to eliminate
isotopes, ie.
                                  peaks less than precision are
considered
                                  insignificant  (default=`0.0000001')
  -c, --charge=INT              Charge state of the protein/peptide

Initialization files:
  the following options may be set to allow for different locations of
init
  files
  -i, --init_file=filename      Initialization (abundance) filename
                                  (default=`initFiles/abundanceTable')

Run specific parameters:
  the following parameters are used to set the values of run specific
  parameters
  -d, --dta_filename=filename   DTA Filename

(default=`dtaFiles/hdx_dtas/30.output.dta')

 Group: sequence group
  a sequence is required
  -s, --sequence=STRING         Amino Acid Sequence String (short)
  -f, --fasta_file=filename     FASTA formated Amino Acid Sequence file
```

API (version 0.3)

The application programming interface for SpectralMatch, HDXRates and the infrastructure behind these two tools can be found at:

http://histone.cti.depaul.edu/~kdrew/docs/ .

Download Programs and Source Code

The SpectralMatch program, the HDXRates program and the source code can be downloaded from:

http://histone.cti.depaul.edu/~kdrew/IBG_Workbench-0.3.tar.gz

Acknowledgements

I would like to thank David Angulo and Alex Schilling for guidance while working on my thesis project. I would also like to thank Leo Irakliotis and Janos Simon for participating on my thesis committee. Eric Puryear deserves mention for his excellent support in infrastructure programming. Tobin Sosnick was generous enough to donate peptide samples for the HDX experiments. Steve Patrie and Laurie Parker were very helpful with discussions on future motivation for the programs. TMD&RMB.