CRANFIELD UNIVERSITY

Oshiobugie Dokpesi

Development of a database and its use in the Investigation of Interferences in SRM assay design

School of Health
Bioinformatics

Msc
Academic Year: 2011 - 2012

Supervisor: Dr Conrad Bessant
March 2013

CRANFIELD UNIVERSITY

School of Health
Bioinformatics

Msc

Academic Year 2011 - 2012

Oshiobugie Dokpesi

Development of a database and its use in the Investigation of Interferences in SRM assay design

Supervisor: Dr Conrad Bessant
April 2013

This thesis is submitted in partial fulfilment of the requirements for the degree of Msc by Research

ABSTRACT

Selected Reaction Monitoring (SRM), is a form of mass spectrometry that guarantees high throughput and also a high level of selectivity and specificity. Performing SRM experiments requires the development of assays to aid in peptide identification. This is a time consuming and expensive process thus biological researchers have come up with bioinformatics solutions for the design of SRM assay. The accuracy of these bioinformatics methods is quite high and the next step is to optimise the process by tackling the interference issue. As various analytes may have the same signals within an SRM experiment and thus interfere with each other's signals, different solutions are being derived to tackle the issue.

This thesis describes the development of a SRM transition database to store peptide and transition data, software to populate the database and also software to retrieve the data from the database. Finally the database is tested with the MRMaid transitions for the human proteome which were mined from the PRIDE database and the results analysed to investigate the transition interference issue.

The database currently contains data for 20220 proteins and approximately 870,000 tryptic peptides from the human proteome.

Keywords: Bioinformatics, Proteomics, Peptides, Transitions, Spectrometry

ACKNOWLEDGEMENTS

I would like to thank God.

My Supervisor Prof Conrad Bessant has been a source of inspiration, knowledge and insight. I am grateful for his patience with me as the project commenced and for giving me the chance to learn a lot on my own and for guiding me when I absolutely needed guidance.

I will like to thank Dr Jun Fan and Andrew Bullimore for opening the world of programming to me. I will like to thank Dr Michael Cauchi and Dr Fady Mohareb as part of the Cranfield's bioinformatics team they have proved very helpful to me in advancing my education.

My Parents, Siblings, Cousins and Friends have been very supportive and without them this would not have been possible. Many thanks Guys

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

CAD	Collision Activated Dissociation
CID	Collision Induced Dissociation
ESI	Electrospray Ionisation
MALDI	Matrix-Assisted Laser Desorption/Ionization
MRM	Multiple Reaction Monitoring
MS	Mass Spectrometer
QQQ	Triple Quadrupole
SRM	Selected Reaction Monitoring

1 Introduction

1.1 Proteomics

Following the completion of genome sequencing efforts for several prokaryotic and eukaryotic organisms, (Aebersold and Cravatt, 2002), biological researchers began to tackle the task of interpreting the results of the translation of these sequences into proteins. In response to this, a new field has emerged. This field following on from the fields of genomics and transcriptomics is known as proteomics (Yates et al., 2009). According to (Ray et al., 2012) proteomics is the systematic study of proteins encoded by a genome for their expression, localization, interaction and post-translational modifications. The term proteome was first introduced in the mid-1990s and it was used to describe the functional complement of a genome (Zybailov et al., 2005) and can be defined as the complete set of proteins, expressed in the lifetime of a cell including those that have been post-translationally modified, (Forner et al., 2007).

The idea is that proteomics will enable researchers develop new technologies that will bring forth the discovery of new therapies and medical advancements and thus further increase our understanding of cellular biology. In order for these goals to be met there are several technical challenges that must be overcome. Challenges that have in the past limited efforts in the quantification and identification of proteins from highly complex samples (Aebersold and Cravatt, 2002).

At the beginning of the proteomics era more than 10 years ago (Messana et al., 2013), the main method of studying proteins was 2D gel electrophoresis. The advancement in proteomics from that has led to the utilization of many different methods coupled together to form an efficient system whose main goal is to solve the many different issues concerned with the study of proteins. Examples of such systems, include the mass spectrometry technique in conjunction with high-throughput separation techniques such as liquid chromatography (LC).

Though the methods with which to study proteins and retrieve valuable information about their nature are not as advanced as those used to study

genes, much progress has been made in their development. These methods can be divided into two types, qualitative and quantitative proteomics. They have become more complex and advanced with time and this is fuelled by the desire to describe the proteins and understand their precise functions. They have developed from the more basic gel-based methods such as gel electrophoresis to currently high – throughput methods such as mass spectrometry. Mass spectrometry allows the quantitative and qualitative study of proteins in far greater measures than previously possible.

The primary goal of qualitative proteomics is to define the complete set of proteins present in a sample including post –translational modifications (Messana et al., 2013) without particular concern for quantity. Though this in itself is significant, there is the chance of suppression of low abundance proteins within a sample thus preventing them from being detected. This causes difficulties when specific proteins such as biomarkers, within a sample may be of great importance but due to concentration levels they cannot be measured adequately. Qualitative studies focused on protein identification relying on shotgun strategies are now being complemented with large scale quantitative experiments, (Kiyonami et al., 2011). This has come about as a result of the demand for identification of the proteins supplemented with the need to acquire enough information about their quantities in order to make modelling of these proteins in systems biology possible.

Quantitative proteomics on the other hand is defined by (Karp and Lilley., 2007), as the comparison of distinct proteomes which enables the identification of protein species. These protein species are identified by the changes in expression or post-translational state in response to a given stimulus. General approaches of quantitative proteomics can be further divided into relative and absolute quantitation.

1.2 Methods Used in Proteomics Analysis

A proteomic analysis include the top down methods where the proteins are analysed intact or the bottom up methods proteins are enzymatically digested into peptides according to (Yates et al., 2009). A detailed understanding of

protein digestion is very important where quantitative proteomics is concerned. This is because the efficiency of digestion is proportional to results which are easily reproduced and enables a standardization of empirical practice (Switzar et al., 2013).

Protein digestion can be performed with the use of enzymes or by chemical means. It is an important process in proteomics as a vast majority of the proteomics experiments rely on the digestion of protein into peptides (Switzar et al., 2013). The analysis of peptides are preferred to protein in many proteomic experiments due to the factors such as the length of the polypeptide, lower molecular mass, charge states and more efficient separation by liquid chromatography which result in an increase in sensitivity in experiments. The most common method for digesting proteins is by the use of proteolytic enzymes. There are many proteases available for this purpose (Switzar et al., 2013) and they differ in their modes of action on proteins which means that a researcher can tailor his experiment through the use of a specific enzyme. The most widely used enzyme is trypsin due to its predictable action and the size of peptides it produces after digestion. The peptides produced by trypsin are ideal for MS and thus it has become the gold standard in MS experiments.

Among the first methods used to study proteins were gel – based techniques which were as a result of the emergent understanding of chemistry at that time period. These techniques were mainly based around the electrochemical properties of the proteins and their behaviour in chemical matrices.

They were focused not only on the visual observance of the proteins in mediums but also on their electrochemical properties. Later methods take into account these electrochemical properties but due to larger amounts of information produced use more sophisticated methods to observe and measure the actions of the proteins. This need to analyse these experimental results resulted in biologists using computational tools to expedite the process.

Nowadays it is possible for proteins to not only be measured, observed and analysed by complicated tools, but also for extrapolations such as the prediction of the structure of certain proteins to be made from sequence alone. This is the

area of bioinformatics and computational biology where scientists now have the ability to use the power of computing to create solutions to problems in biology by simulating biological environments and conditions.

There are several proteomic tools available and they fall under several proteomic methods such as separation methods, MS methods and relative and absolute quantification methods. Tools used for separation methods include capillary and gel electrophoresis, micro channel, protein chips and Liquid Chromatography (LC) and High-Performance Liquid Chromatography (HPLC) (Palagi et al, 2006). MS methods include Electrospray ionisation (ESI), Matrix assisted laser desorption ionisation (MALDI) as ionisation sources, Triple quadrupole (QQQ) and Fourier Transform-Ion Cyclotron resonance FT-ICR Shotgun approach.

Relative quantitative methods include label-free and labelled methods. The labelled methods consist of metabolic labelling such as the Stable isotope labelling by amino acids in cell culture (SILAC) method. Enzymatic labelling such as ^{18}O- labelling and chemical labelling methods such as Isobaric tags for relative and absolute quantification (iTRAQ), tandem mass tags (TMT), difference gel electrophoresis (DIGE) and isotope-coded affinity tags (ICAT). The label-free approach includes methods such as spectral counting, peak intensity and densitometry where quantification involves comparing the peak intensity of the same peptide or the spectral count of the same protein (Zhu et al., 2010).

Absolute quantification methods include methods such as Absolute Quantification peptides (AQUA) (Gerber et al., 2003) which is a method that allows for the absolute quantification of proteins and translationally modified proteins via a two stage process. The first stage involves choosing an internal standard which is a stable isotope incorporated into a synthetic peptide. The second stage involves the use of the internal standard, for absolute quantification. As an absolute amount of the AQUA peptide is added to the sample, the absolute quantification can then be calculated by the comparison of the abundance of the AQUA internal peptide against that of the peptide being

measured (Gerber et al., 2003). The Quantification Concatamers (QconCAT) method (Rivers et al., 2007), Is a method which involves concatenating proteolytic peptides chosen from several proteins to be quantified. The concatenated peptides are then assembled into an artificial protein. An artificial QconCAT gene is then designed (Brun et al., 2007) which encodes this concatamer. The QconCAT gene is inserted into a high level expression vector which is then expressed in *Escherichia coli* (Benyon et al., 2005). The bacteria is grown in a growth medium that contains heavy amino acids. The purified QconCAT protein is digested with the sample to be studied and the mixture of QconCAT peptides and peptides of the protein to be quantified are then analysed (Holman et al., 2012). Protein Standard Absolute Quantification (PSAQ) (Brun et al., 2007) is a method that 'uses in vitro synthesized isotope labelled proteins as standards for absolute quantification'. According to (Holman et al., 2012) PSAQ proteins are added to the sample pre-digestion to facilitate quantification. The PSAQ protein standards used are a very close biochemical match to the target proteins and thus can be directly added into the samples to be analysed (Brun et al., 2007).

Choosing the right separation methods is often the first step in designing the proteomic application. The major separation methods widely used in proteomics are gel based and gel free. (Yates et al., 2009).

1.2.1 Separation

1.2.1.1 2D SDS PAGE

Sodium dodecyl sulphate poly acrylamide electrophoresis (SDS – PAGE), is a technique whereby proteins are separated in gel electrophoresis according to their charge state and/or size. When Sodium Dodecyl Sulphate (SDS) binds to proteins it makes the protein molecules linear and imparts a negative charge on it. 2 Dimensional SDS PAGE is a valuable tool in proteomics and its use enables the separation of thousands of proteins. The first dimension is an isoelectric focusing (IEF) step that is performed that allows the proteins to migrate on the gel along a neutral pH gradient until they reach a position where they contain no charge. The next step is then to transfer the IEF strip with the

proteins on it to a polyacrylamide gel. This is the second dimension as it employs the molecular mass of the proteins. SDS is bound to the proteins in order to impart a uniform charge on the proteins. A voltage is applied to the gel and the negatively charged proteins then migrate across the gel toward the positive end. The protein's migration across the gel is dependent on the molecular mass of the molecules. The smaller proteins move further through the gel matrix as a result of less resistance encountered within the gel matrix. After the 2D SDS PAGE is complete the gel is stained using several different kinds of staining methods. There are dye-based staining methods such as Coomasie brilliant blue, zinc imidazole staining methods, silver stains such as silver nitrate and silver ammonia and fluorescent stains such as SYPRO Ruby protein gel stain from the family of SYPRO dyes (Chevalier, 2010). This results in a two dimensional map of hundreds or thousands of proteins. Proteins of interest within the gel can also be excised and used in various other techniques such as mass spectrometry for further analysis. A disadvantage of 2D SDS PAGE is that spots on a given 2D gel often contain more than one protein (Zhu et al., 2010). This may make the quantification of the protein of interest difficult as it may be unclear which protein is being observed.

1.2.1.2 Liquid Chromatography

Chromatography is a technique that can be used to separate thousands of compounds. There are two types of chromatography. Gas Chromatography (GC) and Liquid Chromatography (LC). Liquid Chromatography is one of the main techniques used in proteomics for proteins and peptides analysis. It involves passing the protein or peptide in solution through a solid phase usually a column at high pressure. The interaction of the column with the proteins in the solution, causes the proteins or peptides to separate along the column. As they emerge from the end of the column a detector measures the molecules. Liquid chromatography is often used in conjunction with mass spectrometry LC-MS. This ensures greater selectivity and specificity.

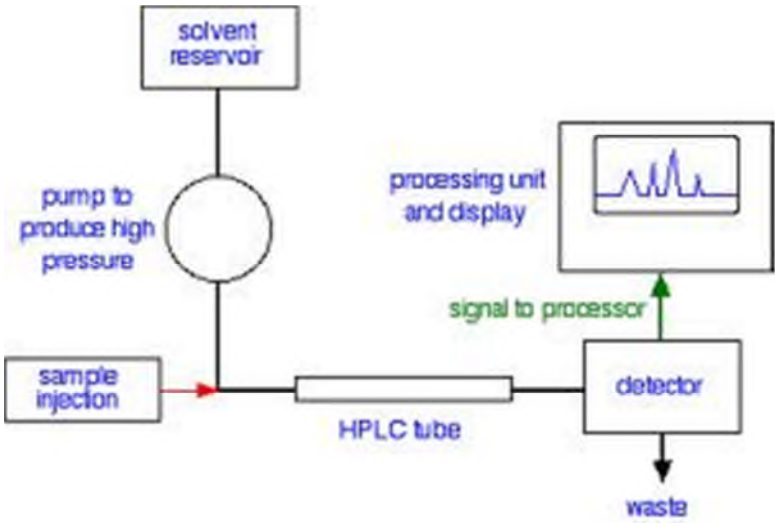


Figure 1. Simple diagram to show the process of liquid chromatography. The sample and a solvent are mixed into a solution and then run through the fractionation column. The proteins within the column elute at a predictable rate proportional to their hydrophobicity. The detector identifies the substance as it passes form the column. http://www.chemguide.co.uk

1.2.2 Mass Spectrometry (MS)

At its simplest a mass spectrometer measures the mass – to – charge ratio (m/z) of ionised molecules (Forner et al., 2007). (Yates et al., 2009) describe the method as the most comprehensive and adaptable tool in large – scale - proteins on a large scale (Kito and Ito, 2008). Mass spectrometers are usually made up of the ion source and optics, the mass analyser and the data processing electronics (Yates et al., 2009).

Initially in the proteomics field the main tool of choice was 2D PAGE, the issue with 2D PAGE is that its resolution that can only analyse the most abundant proteins in a sample (Kumar and Mann, 2009). A new method was needed to perform more high through – put proteomics. Mass spectrometry has long been used in the field of chemistry to identify and quantify molecules. Proteomic mass spectrometry has experienced rapid growth in the past two decades due to important developments in experimental methods, instrumentation and data analysis approaches (Yates et al., 2009). The ability to present proteins or

peptides in an ionised and gaseous state and the maintenance of the protein/peptide molecule without any degradation was a previous challenge in proteomics mass spectrometry. MS also has an ability to provide quantitative information in proteome analysis.

Two breakthrough techniques which were very important for the ability of the Mass spectrometer to measure proteins are electrospray ionisation (ESI) and matrix – assisted laser desorption ionisation (MALDI).

Limitations of biological MS require several different methodologies to be applied to protein analysis. The approaches include sample preparation, ionization, data acquisition and data analysis. Their application may differ depending on how complex the sample is and the purpose of the analysis. Front-end separation is also necessary to detect the signals of low-abundance proteins that would otherwise be obscured by a higher abundance signal. Therefore efficient separation is important to the accuracy and sensitivity of a mass spectrometric experiment (Yates et al., 2009).

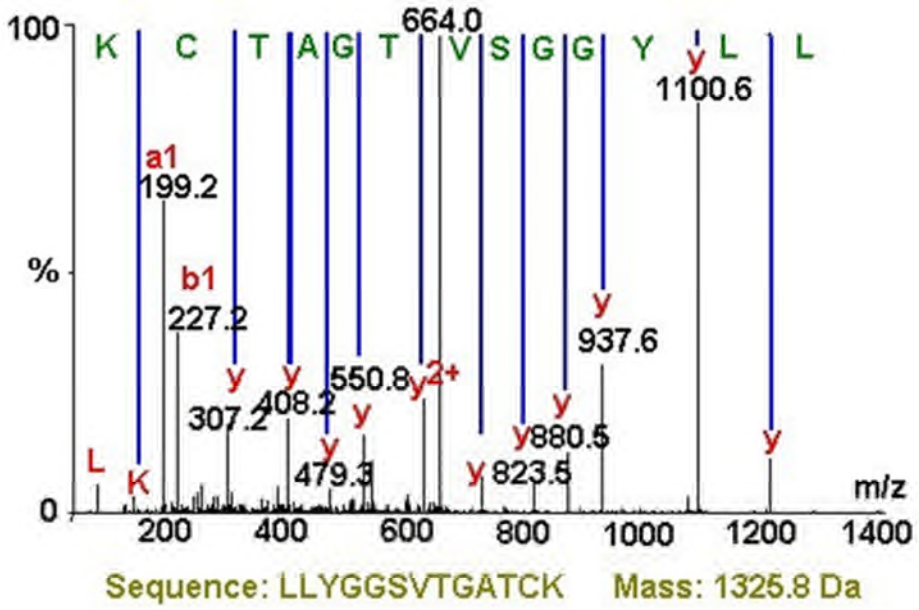


Figure 2. Shows the mass spectrum of the peptide LLYGGSVTGATCK. The individual amino acids can be elucidated by measuring the peaks and the distance between the peaks. The m/z of an amino acid is usually an identifier and the identification can be validated by comparing with against a library of mass spectra. www.astbury.leeds .ac.uk/facil/mass.htm

1.2.2.1 Electrospray Ionisation (ESI)

One of the most important developments in peptide based mass spectrometry was the introduction of ionisation methods that allow for proteins and peptides to be reliably measured by MS. These methods were ESI and MALDI. The ESI method produces ions from solution. The peptide ions in solution are forced through a needle and the solution is evaporated until the peptides get into a state whereby they repel one another. Once this occurs they are dispersed into a fine mist and the charged ions are collected to be fed into a mass spectrometer for further analysis.

1.2.2.2 Matrix – assisted Laser Desorption Ionisation (MALDI)

MALDI is the production of ions from the laser activation of a target analyte which has been fixed on a solid matrix. The resulting ions are mostly singly charged. This makes MALDI more suitable for top-down analysis of high-

molecular weight proteins using pulsed analysis instruments (Yates et al., 2009). MALDI has also led to the use of other techniques such as surface-enhanced laser desorption ionization (SELDI) which introduces the concept of surface affinity to the ionisation of protein and peptide molecules.

1.2.2.3 Mass Analysers

Mass analysers are an important part of each MS instrument because they are able store ions and separate them based on their mass-to-charge (m/z) ratios. The most widespread mass analysers are quadrupoles (Q), Ion traps (IT), time of flight (TOF) and (FT-ICR) analysers (Forner et al., 2007). These mass analysers can be separated into two major categories: the scanning and ion-beam mass spectrometers, such as TOF and Q; and the trapping mass spectrometers such as Ion Trap, Orbitrap and (FT-ICR) (Yocum and Chinnaiyan, 2009). Mass analysers separate ions based on their m/z resonance frequency, quadrupoles (Q) use m/z stability, and time-of-flight (TOF) analysers use flight time (Yates et al., 2009). To achieve maximum performance hybrid combinations of these analysers have been developed such as the (QQQ), ion trap/FT-ICR and time of flight-time of flight (TOF-TOF). (Forner et al., 2007).

1.2.2.4 Detectors

Fragment ions that pass through the mass analyser are detected by the detector. The electromagnetic signal determines the number of ions present at each m/z value. The result is a mass spectrum or chart with a series of spikes or peaks, each representing a charged protein fragment from the sample. The height of each peak represents the amount of that particular protein or fragment that is present in the sample. The size of the peaks and the distance between them is the protein pattern or array of the entire sample. Each spectrum contains enough data points for every protein and protein fragment through which their molecular weight and intensity values may be used to reflect their relative abundance within the sample. Detectors may consist of multiple channel plates detectors, photo-multiplier detectors or electron-multiplier detectors.

1.2.2.5 Tandem Mass Spectrometry

Tandem mass spectrometry (MS/MS) has been widely used in proteome analysis, where a peptide ion to be to be analysed is selectively isolated and fragmented to obtain an MS/MS spectrum (Kito and Ito, 2008). In tandem MS, mass analysis can be carried out on intact molecular ions or on fragmented precursor ions. In most cases full scans produce masses of the proteins or peptides and fragmentation scans yield the primary sequence information. (Yates et al., 2009).

The fragmented precursor ions are generally referred to as product ions. The fragmentation is usually by ion dissociation. The most commonly applied fragmentation method used for proteome identification and quantification analysis is Collision-induced dissociation (CID) (Quan and Liu, 2013). There are other dissociation methods such as Electron-transfer dissociation (ETD), Electron-capture dissociation (ECD) and Infrared Multi-photon dissociation. In CID conditions the precursor ion undergoes multiple collisions with collision gas such as helium or argon. The increased ion energy and impact of the collisions causes predictable fragmentation. CID generally yields b and y type fragment ions. Complementary to CID fragmentation, ECD which generates radical cations or ETD which transfers electrons, (Quan and Liu, 2013) result in the formation of c and z type fragment ions.

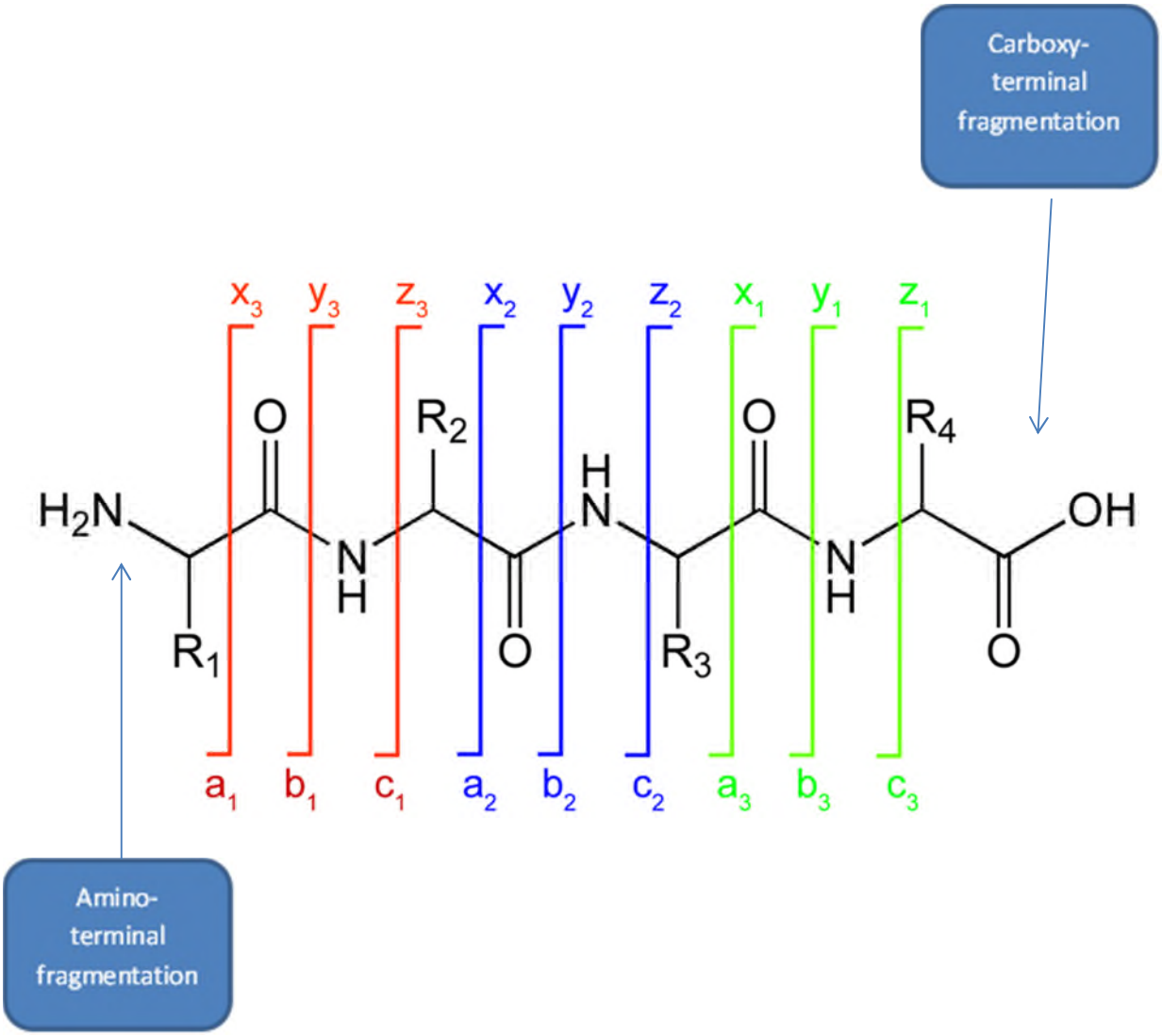


Figure 3. Peptide Fragmentation. a, b and y ions are the most common ions observed in low energy collisions. C ions are observed in high energy collisions. Sourced from Wikipedia

1.2.3 Data Analysis

The raw data retrieved from the mass spectrometry experiment must be processed and converted to the final result according to (Jones and Hubbard, 2010). Tandem mass spectrometers can now process several tens of thousands peptide ions per hour. This process requires the use of advanced

algorithms for proper interpretation of the data and also without the cost of too much computational time. The final result following the conversion can either be protein or peptide identifications. There are three approaches which have been developed for analysing mass spectrometric data according to (Forner et al., 2007). These are the knowledge-based approach, an ab-initio approach and a sequence-tag methods approach.

The knowledge based approach involves comparing the spectra acquired from the mass spectrometry experiment with a database of theoretical fragments this approach is relatively simple and robust. SEQUEST, (Eng et al., 1994) is a commercial online mass spectrometry data analysis tool, which has been around for a long time and performs protein and peptide identifications using knowledge based approach and is a non-probability based model. Another tool that uses the knowledge based approach is Mascot, (Perkins et al., 1999) which is another popular tool that has been around for use in proteomics for a while. It is available as a web tool and a standalone version (Cham Mead et al., 2010). According to (Forner et al., 2007) it utilises a probability based method of mass spectrometry data analysis. Other probability based tools for protein/peptide identification utilising the knowledge based approach include X!TANDEM which was developed as an alternative to SEQUEST and Mascot and has the advantage of being able to search for post translationally modified peptides already observed thus reducing the computational time spent calculating all possibilities of peptide modifications. OMMSA like X!TANDEM is an open source tool and freely available where identification is based on the probability of the match being random.

De Novo sequencing is an ab-initio approach that involves the direct elucidation of the peptide sequence from the fragment ion spectrum. The de novo method is a mainly database –independent method. Originally this method was carried out manually but now software tools such as the PEAKS software which is one of the first and most widely used algorithms, are used to assist in the process. An advantage of the de novo sequencing method is that it does not require prior

knowledge of sequence. De novo sequencing is mainly used in situations whereby species with limited genome information available are being studied.

The approaches that involve the sequence-tag methods are usually hybrid in nature and can combine database searching with de novo sequencing. Short sequence tags from MS/MS spectra are used in a protein database search together with other information. The search allows for one or more mismatches between acquired spectra and those in the database. By limiting the search space to those only containing the sequence tag extracted from the spectrum, the search time is significantly reduced. This approach is useful for identifying peptides with post-translational modifications or unknown sequence variations according to (Forner et al., 2007).

These methods generally utilise the spectra generated by the mass spectrometry experiments to measure peptide sequence and quantity. According to (Lai et al., 2013) information which may be obtained from the spectra include, peptide peak intensity derived from the height or area of a peak, the peptide precursor ion peak height and the peak height of product ions. This information can then be used to quantify peptides through methods such as spectral counting and peptide peak intensity measurement (Wasinger et al., 2013). This quantification based on peak areas or heights of peptide ions must take into account several factors such as chemical modifications of the peptide ions, post translational modifications, uniqueness of the peptide and complete digestion. Complete digestion is important as every protein within the sample must be digestible by the enzyme used. This allows shotgun techniques such as MS to be able to detect every protein present as the proteins produce peptides that can be detected in terms of size and sensitivity (Elliott et al., 2009). Though the peak height or the peak area can both be used to measure peptide quantity, there are advantages of using either one. Using the peak height to measure quantity is of advantage when the width of the peak does not vary between samples and there is a strong signal with minimal noise (Zhang et al., 2010). On the other hand it is preferable to use the peak area when there is a lot more noise as more information is derived from more data points.

Interference from other peaks becomes a problem when peak area is used to measure quantity as a result of the larger area in the m/z and retention time space used according to (Zhang et al., 2010). This issue of interference is very important in mass spectrometry based data generation.

This may help the researcher choose to use either peak area or peak height to measure quantity when designing their own experiment.

1.2.4 Bioinformatics Tools in Proteomics

(Kumar and Mann, 2009) define bioinformatics as a means for functional analysis and data mining of data sets leading to biologically interpretable results and insights. Following on from that, the current tools used in proteomics have generated a very large amount of data. The data poses a lot of challenges for scientists trying to understand them. The field of bioinformatics is well practiced in the manipulation of biological data from experience in genomics. Bioinformatics has evolved to deal with a multitude of different data types and should now be well – equipped to aid proteomics (Kumar and Mann, 2009). The number of protein and proteome databases being developed is increasing at a rapid rate. There is an abundance of information about specific proteins which can be found in databases that store protein sequences. One such database is Swiss-Prot. Swiss-Prot is a database that provides access to protein sequences and to information such as descriptions of a protein's function, its domain structure, and post-translational modifications and links to other databases.

Peptide identification via Peptide Mass Fingerprinting (PMF) involves the comparison of experimental peaks from mass spectrometry with theoretically digested proteins to elucidate the peptide sequence. This uses a number of algorithms to try and find the most accurate match of the experimental spectra to theoretical spectra within databases. There are many computer programs and databases which have been built to help analyse MS data for the identification of proteins. One of the most widely used tools for PMF is Mascot which utilises the MOWSE algorithm to provide a score for the most adequate

match. There are a number of these databases which provide a repository of MS spectra.

1.3 Selected Reaction Monitoring

There has been a recent trend in proteomics toward the development and application of technologies for the targeted analysis of proteins within complex mixtures (Prakash et al., 2009). One such technique is called Selected Reaction Monitoring (SRM). Selected reaction monitoring (SRM) is a non – scanning, targeted mass spectrometry technique that can be used to accurately quantify proteins in complex biological mixtures, (Chang et al., 2012); (Cham Mead et al., 2010). SRM can be used to overcome the limitations of shotgun – MS/MS (Holman et al., 2012). While its application is relatively new in proteomics, it is a method that has been used extensively in the toxicology and pharmacokinetics disciplines to determine and analyse small molecules for decades (Yocum and Chinnaiyan, 2009); (Abbatiello et al., 2010). It is only within the last few years that it has been applied to proteomic research and then mainly to quantitative proteomics. SRM exploits the capabilities of tandem quadrupole spectrometers also called triple (QQQ) MS for quantitative analysis (Lange et al., 2008). In SRM, the first (Q1) and third quadrupoles (Q3) are mass analysers which are used as filters to monitor selected precursor ion and fragment ions. The first quadrupole monitors and filters the precursor ion, which in a quantitative proteomics experiment is the mass – to – charge (m/z) value of an ionised peptide of interest (Holman et al., 2012). The precursor ion that is selected is introduced into the second quadrupole (Q2) which serves as a collision cell and is useful in increasing selectivity. Collision induced dissociation (CID) occurs within the second quadrupole and the precursor ion is fragmented to produce product ions. The product or fragment ions are then introduced into the third quadrupole which detects a preselected fragment ion by is m/z value. See figure 4 for a workflow of the SRM experiment.

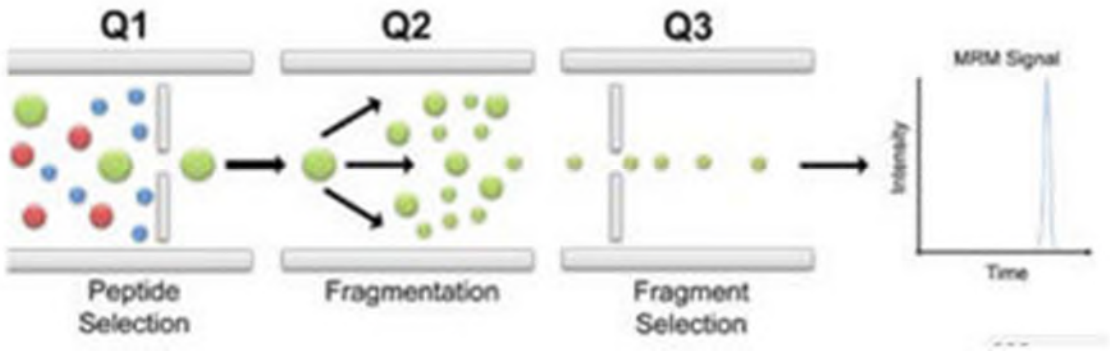


Figure 4. Workflow of a typical SRM experiment the precursor ion is selected in the first mass analyser (Q1) then in the collision cell (Q2) the collision energy is optimized to produce the desired fragment ions. The third mass analyser (Q3) is set to selects the only fragment ion of interest. http://www.mrmproteomics.com

The SRM technique is a very good candidate for hypothesis driven proteomics, due to its specificity and sensitivity/ high selectivity of the SRM technique as well as its multiplexing ability make it a very good candidate for hypothesis driven proteomics (Chandramouli and Qian, 2009). The detection of biomarkers can be of utmost importance in medicine. Therefore analytical methods suitable for accurate research need to be as robust as possible. Selectivity is crucial in such methods (Sauvage et al., 2008). The SRM technique is equipped to significantly reduce interferences, thereby allowing a dramatic increase in selectivity, a very low baseline, very good limits of quantitation and a very good linearity (Hunter, 2010). The specifically selected and predefined mass – over – charge (m/z) values of the precursor ion against the fragment ion is known as a transition. The ionisation efficiency of the parent ion (Q1 transmission), the fragmentation efficiency of the parent ion and subsequently the intensity of fragment ion (Q3 transmission) are very important in achieving successful MRM transitions (Yocum and Chinnaiyan, 2009).

In an SRM experiment several of such transitions are observed over time, yielding a set of chromatographic traces with the retention time and signal

intensity for a specific transition as coordinates (Lange et al., 2008). Using the selectivity of multiple stages of mass selection of a tandem mass spectrometer, these targeted SRM assays are the mass spectrometry equivalent of a western blot. (Prakash et al., 2009).

In comparison to discovery centred methods, methods which are SRM based and use the triple quadrupole mass spectrometers are known to have a high specificity and sensitivity within a complex mixture and therefore can be performed in a fraction of the instrument time. (Prakash et al., 2009).

A multiplexed approach can be taken whereby LC-SRM peptide – based assays can be used for quantifying panels of proteins. This involves measuring multiple SRM transitions within the same experiment and there is an assumption that the individual SRM do not interfere with each other within the assay (Mead et al., 2009). Due to the multiplexed nature of this assay it is usually referred to as Multiple reaction Monitoring (MRM) and centres on balancing productivity against sensitivity. Although the SRM term is widely used most people prefer to represent the method as MRM.

The MRM analysis of numerous peptides enables detection of the different proteins in a single sample, as opposed to reproductions of the same proteins and this set of assays could be further expanded to include other endogenous proteins of interest. (Remily-Wood et al., 2011). The terms SRM/MRM can also be used to describe experiments that are conducted in instruments other than triple quadrupole MS (Lange et al., 2008).

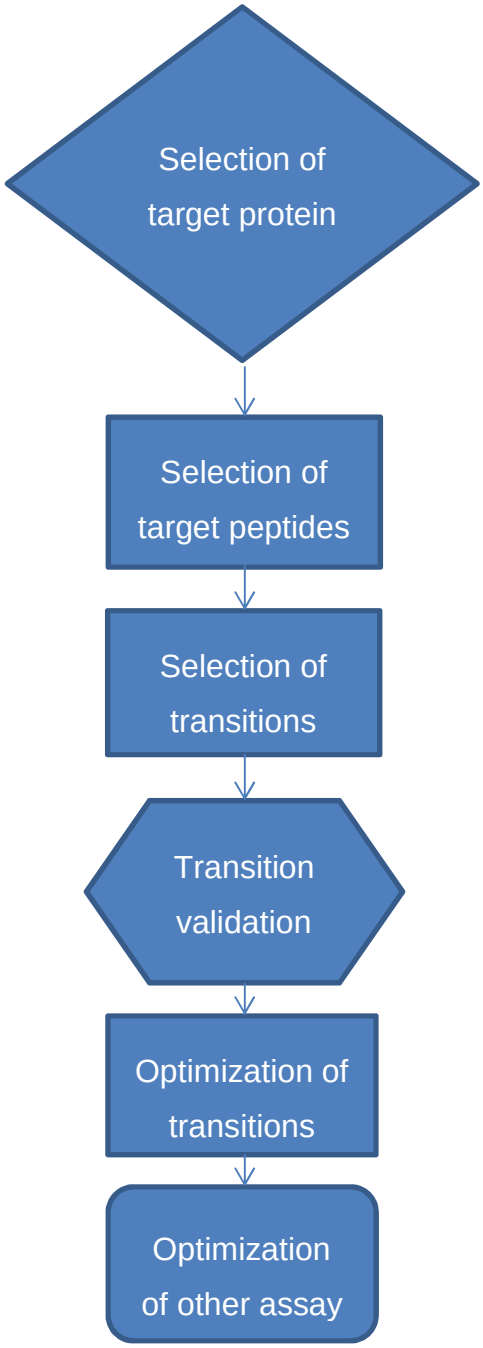


Figure 5. A SRM design workflow. The experiment starts with the selection of a target protein. This can be based on the focus of the study (biological or clinical), experiments or from scientific literature. Following this step is the selection of target peptides (unique, proteotypic). Then for each peptide the most optimal transitions are selected and then validated. Other assay parameters are also optimized. These assays are then ready for application in proteomics experiments to quantify or detect proteins.

1.3.1 SRM assay design

The development of an SRM assay for an individual protein is not an easy and straight forward process. According to (Mead et al., 2009) the critical part of performing SRM is designing suitable transitions to monitor the peptide and by extension, the protein of interest. To achieve this goal, there are certain conditions to be met in order to conduct a SRM experiment.

First a protein of interest selected as a target for quantification is cleaved/digested into peptides. Then out of all the peptides relating to the protein of interest a peptide that shows uniqueness for the protein is chosen. Peptides such as these are termed proteotypic peptides. Then the right transitions have to be chosen and these transitions are a derivation of fragment ions specific to the peptide (precursor) ion. The sensitivity of SRM experiments is highly dependent on these transitions. Though this allows for very good quantification results the price to pay is in the amount of time required to pick and choose the right transitions. Following this the transitions derived have to be confirmed for the identification of the peptide of interest. After these conditions have been fulfilled other factors such as the collision energy within the collision cell and the retention time, have to be taken into account. Choosing the right parameters will help to improve the sensitivity and signal response.

In designing SRM transitions certain rules apply. For Q1 doubly charged parent ions are favoured over singly charged ions as the doubly charged parent ions yield higher quality MS/MS fragmentation and y-ions are mostly preferred to b-ions, (Prakash et al., 2009) found that when choosing SRM methods a higher performance can be derived from choosing the most intense y-ions found in the spectrum of a tryptic peptide compared to the normal experimental approach for selecting product ions and that these methods can be used as a part of automated development workflows. They also state that the b2 ion is frequently one of the largest but least selective product ions in the spectrum. (Prakash et al., 2009). The Q3 fragment should have a greater m/z than the selected Q1 parent ion and if the sequence contains a proline residue, the high abundance

y-ion created from fragmentation closest to the N-terminal to the proline is selected (Yocum and Chinnaiyan, 2009) Higher m/z fragments should also be chosen over lower m/z fragments as they are more informative (Han and Higgs, 2008), this is due to the fact that there is a lower likelihood of obtaining the same m/z value from an unrelated combination of amino acid residues when a higher m/z value is used than when a lower m/z value is used.

(Yocum and Chinnaiyan, 2009) have recommended that in addition to choosing the most intense transition for quantification, two additional transitions should also be chosen as this will provide the most selectivity and combined reduced interference from other peptides.

There are other considerations which are taken into account when designing SRM peptides. These include the length of the peptide which should be between 8 – 25 amino acid residues. The presence of multiple aliphatic amino acids such as Alanine (A), Leucine (L), Isoleucine (I) and Valine (V) as these factors influence the hydrophobicity of the peptide and may cause synthesis problems. According to (Deutsch et al., 2012) extreme hydrophobicity (low or high) will also cause inconsistency in observation by LC-MS methods. Factors which affect the stability of the peptide should be taken into account such as the presence of oxidation sensitive residues such as, Cysteine (C), Tryptophan (W) and Methionine (M). N-Terminal Glutamine (Q) residues should also be avoided as they are converted to pyroglutamic acid. Genetic variants, modified residues, peptides also showing the best signal intensity and chromatographic peak shape for a given parent protein are selected (Anderson and Hunter, 2006).

1.3.2 Bioinformatics and SRM Design

Figure 6. Screenshot for the MRMaid SRM assay design tool. A protein accession number of interest is entered and the most optimised transitions are computed and displayed on this interface. The transitions are also ranked according to their suitability for use in an SRM experiment.

Many SRM experiments aim at comparing protein abundance across conditions or time points of interest (Chang et al., 2012). SRM experiments however require a lot of preparation in the form of selecting appropriate signatures for the proteins and peptides that are to be targeted (Deutsch et al., 2012). In order to facilitate this, computational and statistical tools have been developed. These tools allow for faster and more accurate analysis and also help to increase throughput. The tools assist in SRM assay development, signal processing and protein significance analysis (Chang et al., 2012). Currently a lot of the tools available focus on the assay development side of SRM experiments. A

challenge of performing these targeted protein analysis is the amount of time spent preparing methods to measure specific transitions for specific peptides that are both selective and sensitive (Prakash et al., 2009).

The different software resources developed to assist in the design of SRM assays have their own specific qualities. Predicting the right transitions from peptide sequence alone requires absolute knowledge of the behaviour of peptides in various situations such as the cleavage/digestion products of protein sequences, fragmentation patterns in a collision cell and the ionisation of the peptides and products. Though there are certain algorithms that return results with a high degree of confidence, the data is still not conclusive. Therefore many of the software tools available have followed the route of generating transition data from mining spectral libraries (Lam et al., 2007).

Various tools such as PRIDE (Vizcaino et al, 2010), PeptideAtlas (Farrah et al., 2011) and Global Proteome Machine database (GPMDB) act as such spectral libraries. They have accelerated proteomics research by facilitating more efficient cross – analysis of datasets, supporting the creation of protein and peptide collection of experimental data and supporting the development of software tools (Farrah et al., 2011). These repositories contain millions of spectral data for peptides, which is a valuable source of information in the prediction of useful transitions for SRM experiments. These spectral libraries highlight the importance of high quality annotated data produced by laboratories and the benefits of maintaining such data sets in a structured way that can be shared between research laboratories in the proteomics community (Prakash et al., 2009).

Though some of these repositories also provide the means to design transitions, there are other more specialised tools that have been developed for that sole purpose. These software tools come in two forms; web-based and standalone. The web-based tools are easily accessed via any web server and are available to anyone while the standalone tools require downloading and installing on a local machine. The two different avenues have their advantages

and disadvantages but the end goal is to design transitions using the most efficient algorithms that the authors can come up with.

Among the web-based tools is the SRM transition design tool MRMaid (Cham et al., 2010), (Mead et al., 2009). This software resource was originally designed with SRM assay design as its main focus. It uses spectra data mined from the PRIDE database to enhance its calculations and recommend transitions (Fan et al., 2012). It is able to recommend transitions for multiple proteins and will return results in a manner that will allow researchers the design SRM assay via the web interface. A number of properties are used to inform the researcher about the usefulness of the peptide being studied and the transitions returned. These properties include the (a) Peptide score which is a score that measures suitability of the peptide for SRM. (b) Retention time (RT) is the estimated retention time attributed to the peptide sequence and its movement through a liquid chromatography set up. This is calculated theoretically using the SSRCalc algorithm (Krokhin et al., 2004). (c) Product ion which is the ion to monitor and (d) The PRIDE data which is the number of experiments in which the peptide has been previously observed. Just by entering the peptide/protein sequence/s to be queried into the web interface these metrics mentioned above including others are taken into consideration and a transition table is returned with the most appropriate suggested. This makes MRMaid a very powerful tool and due to its ease of access a very convenient resource. The next step for the MRMaid software resource is use interferent information to enable researchers design the most optimal assays.

Following on from an example of a web-based software resource an example of a standalone tool is Skyline (Maclean et al., 2010). This is a very popular tool amongst proteomic researchers wishing to design SRM assays. It is a tool that must be downloaded and installed on a local machine and works on windows platform. Instead of initially using peptide libraries to inform its transition selection process, Skyline offers the option of theoretically calculating transitions and also allowing researchers to use their own peptide libraries to

develop their assays. The option is also there to compare these results with a public repository of MS/MS spectra and returns a result of the most abundant transitions. Skyline ranks the transitions according to intensity and together with that uses a dotproduct function which calculates how similar reference spectra is to your transition to present the best transition. In addition to the interface which offers more options due to its architecture the ability of researchers to keep their data within their control makes it a very attractive tool for many researchers. Other tools include the Global Proteome Machine Database (GPMBD) which allows researchers to use information obtained by the GPM servers to validate MS/MS spectra. It also compares the information within the server to determine the number of times other people have observed the peptide. This SRMCollider database developed by (Rost et al., 2012) also allows for the design of theoretical transitions. It also compares the inputted transitions to other transitions in a selected proteome. The transitions that interfere with the input transition are noted and a count of the interferences is returned.

1.3.3 SRM assays and Interference

The high selectivity associated with SRM is actually achieved at two levels (Yan et al., 2008) the Q1 and Q3 parts of the triple quadrupole MS. The precursor ions are first selected at the Q1 mass analyser stage by the mass to charge ratios and then at the Q3 stage any isobaric components are further resolved by selecting for the product ions of these precursor ions. The formation of these product ions occurs by collision-induced fragmentation at the Q2 level within a collision cell and is highly structure specific (Yan et al., 2008). As a result of this it is expected that the scenario whereby another component with the same signals at Q1 and Q3 as that of a transition being monitored will be rare. The actual experimental evidence suggests otherwise and this can be understood by the fact that there are over 20,000 proteins in the human genome minus the post transitional modifications. And when digested into tryptic peptides the number rises into millions. So the probability of finding peptide species with the same retention times and precursor and product m/z is quite high. The ability to

distinguish between correct identifications and false positives is a very important condition for SRM assays (Yocum and Chinnaiyan, 2009). The SRM method was originally designed and used to study small molecules. Scientists familiar with the SRM method are aware of the fact that the identification of analytes by the detection and monitoring of only a few fragment ions is subject to false-positive identification and imprecise quantification (Abbatiello et al., 2010). This makes it necessary to have measures in place to ensure the right results. The main reasons for this false – positive identification and imprecise quantification may be credited to problems such as interference and ion suppression by the constituents of the biological matrix (Abbatiello et al, 2010). Peptides are more complex and larger than small molecules and analytes which the SRM method was originally designed to study, they share considerable homology and are measured in very complex matrices. This complexity results in a lot of peptides with the same ion signals which makes them hard to tell apart by mass spectrometry data analysis. The interference experienced in a biological matrix may be as a result of each fragmentation product of peptide only providing information about one position in that peptide and this may result in the other peptides resulting in the same transition as the peptide analyte in question (Yocum and Chinnaiyan, 2009).

Another potential source of interference is in-source fragmentation of abundant peptides where the fragment ions rather than the precursors are the source of interference. This is caused by the primary or secondary fragment of the precursor having the same or nearly the same mass as the transition of interest. This can be a significant issue for quantification and depends on the level of sensitivity to be achieved. It is crucial to select transition ions that maximise specificity and potentially minimise interferences from co-eluting species that fall within the mass windows and tolerances of the detector (Yocum and Chinnaiyan, 2009). So the main idea is to be able to confidently select for the transitions you want with full knowledge of what to expect if there are any interfering signals or if the researcher is not too sure about the signal he is observing.

Originally interferences were by manual examination of the raw data. Now for many of the software tools, interference data is incorporated into the workflows to aid transition design for example GPMDB has incorporated information on interfering peptides along with their transition results. Other software tools choose to use software resources that have been designed specifically for dealing with interference. These tools have various strategies for coping with the interference issue. (Rost et al., 2012) have developed the SRMCollider which uses the idea of Unique Ion Signatures (UIS) to detect and avoid redundancy in transition design. mProphet (Reiter et al., 2011) uses decoy transitions to create a probability scoring model that filters out interfering transitions. AuDIT the algorithm for automated detection of inaccurate and imprecise transitions developed by (Abbatiello et al., 2010) automatically filters data sets from SID-MRM-MS assays to identify problematic data to the researcher. Its main aim is to use reference peptides and technical replicates to detect transitions with interferences (Reiter et al., 2011). The reason for awareness of interference is most poignant when there are not enough transitions to reliably monitor a peptide thus it will be very advantageous to be aware of all interferences within the vicinity. (Sauvage et al., 2008).

1.4 Aim of the thesis

In complex mixtures, chromatographic signals from isobaric or nearly isobaric precursor peptides might overlap with the specific precursor signals to be measured. If the product ion signals and the hydrophobicity of the precursor also overlap, these isobaric or nearly isobaric peptides might hamper reliable quantification of the specific signals. Performing an SRM assay with higher resolution can result in improved selectivity but to the detriment of analytical throughput. This is due to the fact that a higher resolution is likely to increase dwell time. As mentioned earlier there have been many tools created to aid in the design of SRM transitions. The problem is that most of these tools do not have a means of increasing selectivity by addressing the problem of interferences and the only way researchers tackle this issue is by painstaking

and subjective manual examination of the raw data (Yan et al., 2008). With the increasingly multiplexed nature of the SRM experiments it is obvious that this is a situation that is redundant in the SRM practice.

The overall aim of the thesis is to produce a database of peptides that may be observed in human proteomics experiments. This database will then be used within the MRMaid selected reaction monitoring design tool to indicate peptides that may interfere with those selected by the user. The detection of these interfering peptides and to be more specific theoretically interfering transitions will allow a reasonable resolution to be set without compromising analytical throughput. Considering the number of peptides available to be observed in the dataset used (Human Proteome). The database tool can help predict the presence of interference and help experimentally identify them and to also refine the conditions for the identification of the investigated peptides with the SRM techniques (Sauvage et al., 2008). Also finding possible interferences may lead the researcher to resolve or prevent the issue by improving liquid chromatography separation sample purification finding as using non-interfering MRM transitions. The tool can help establish the uniqueness of a transition as the researcher may be monitoring a more abundant protein instead of their selected protein.

Is the problem of interference a common occurrence and what the probability is of encountering theoretically interfering transitions when performing your SRM experiment.

Objective 1: Create Interferent Database

Data containing information about peptides within a proteome needs to be retrieved related and analysed. The data is then used to find out information that about transitions that may produce interfering signals during SRM experiments. The data from the proteome is to be retrieved from the Swiss-Prot

database and stored on an in house database built in the MySQL database server.

Objective 2: Populate the database

The interferent database is to be populated with all the peptide and transition values derived from the proteome. The means of population is via a perl program that extracts the necessary transitions and compared them with other transitions within the proteome. The transitions with similar or close values within a range are then stored as interferents.

Objective 3: Interrogate the database to query the significance of interferents

An analysis of the database content is then to be performed to reveal the importance and occurrence of interferents. Information such as the relationship between precursor and product m/z tolerances and detectable transitions.

2 Materials and Methods

2.1 Software Development

2.1.1 Database Design

As described in chapter 1, the tracking of interferences to peptide signals in the Q1 and Q3 phase of an SRM assay is of importance as using a very high resolution can affect throughput. Storage of all the protein and peptide information to use in the validation of this process requires the necessary storage mechanism that has the flexibility and capability to handle such data. The system of storage requires the ability to keep track of all the relationships between the data types. The database should also flexible enough to allow for changes such as expansion and addition of new fields. The database should also have referential integrity and data integrity as this ensures the quality of the data

A relational database is the most useful tool for this project as it facilitates the storage of information in various ways that can enable the organization of data in a most efficient manner especially as a storage and retrieval tool. For example relational databases can be easily adjusted, the data within can be queried in very sophisticated ways with ease and simplicity and this is made very possible and easy with the application of the appropriate query utilising the Structured Query Language (SQL). Also connecting the database information to a web interface is possible which then makes the information stored accessible to a wider audience.

The design of the database used for this thesis included the understanding of the requirements for the database, the nature of the data that being stored, creating the appropriate database-entity relationship diagram, normalising the entity-relationship diagram and then generating the necessary SQL script to construct the database.

As shown in figure 7 the MRMinter is a database consisting of 5 tables:

Peptide – A table of protein accession numbers, peptide sequences and the hydrophobicity of the peptide transitions to be monitored. This is the parent table for the precursor and product tables.

Precursor – A table that contains the precursor mass to charge ratio of the peptide sequence. It also contains the charge states of the peptide. The three charge states derived from the perl script are stored within these tables. It is related to the peptide table in a one to many relationship with the peptide table as the parent table.

Product – A table that contains the product mass to charge ratio of the theoretically fragmented peptide sequence. It contains the fragment type and number typically referred to as the product type and product number. It is related to the peptide table in a one to many relationship and also with the peptide table as the parent table.

Transition – A table that holds the theoretical transitions derived from the joining of the above three tables to yield a unique transition. This table is created specifically to hold any transitions observed within the MRMaid database.

Interference – A tables that holds the interfering signals related to the transitions in the transition table. This table is related to the transitions table in a one to many relationship and is the child table.

Required data fields for the database are

Protein accession number - This is the identification of the protein in the proteome. This information is retrieved along with the protein sequence from the curated database Swiss-Prot.

Peptide sequence – After digestion the peptide sequences are stored in the database.

Hydrophobicity- The hydrophobicity of the protein and peptide is strongly linked to the amount of time the peptide takes to pass through the LC column.

This elution time is commonly called Retention time and is a very important factor in mass spectrometry as it adds to the specific identification of a peptide.

Precursor m/z – following ionization by ESI and MALDI as described in chapter 1 the ion mass to charge ratio (m/z) of the peptide is stored in the database singly, doubly and triply charged precursor ions are stored as these are the most readily observed.

Precursor charge – This field identifies the charge state of the precursor ion to which it is linked.

Product m/z – following the theoretical fragmentation of the peptide using the principles for fragmentation in a collision cell the resulting fragment ions of type b and y are stored in the database.

Product type – The type of fragment or product ion is stored in this field. The product types can either be b or y ions.

Product number – According to the principles the number of amino acids along the backbone where the peptide was fragmented is stored as the product number.

All these fields are then taken and stored in the appropriate tables. The Entity – Relationship diagram in figure 7 shows the tables and their relationships. This diagram shows the relationship between the different fields and tables. The peptide, precursor and product tables are normalised to limit redundancy. Due to the nature of the data and the parameters needed to obtain a range for the transitions and interferences for the purpose of obtaining the right query conditions, the transition and interference tables were denormalised.

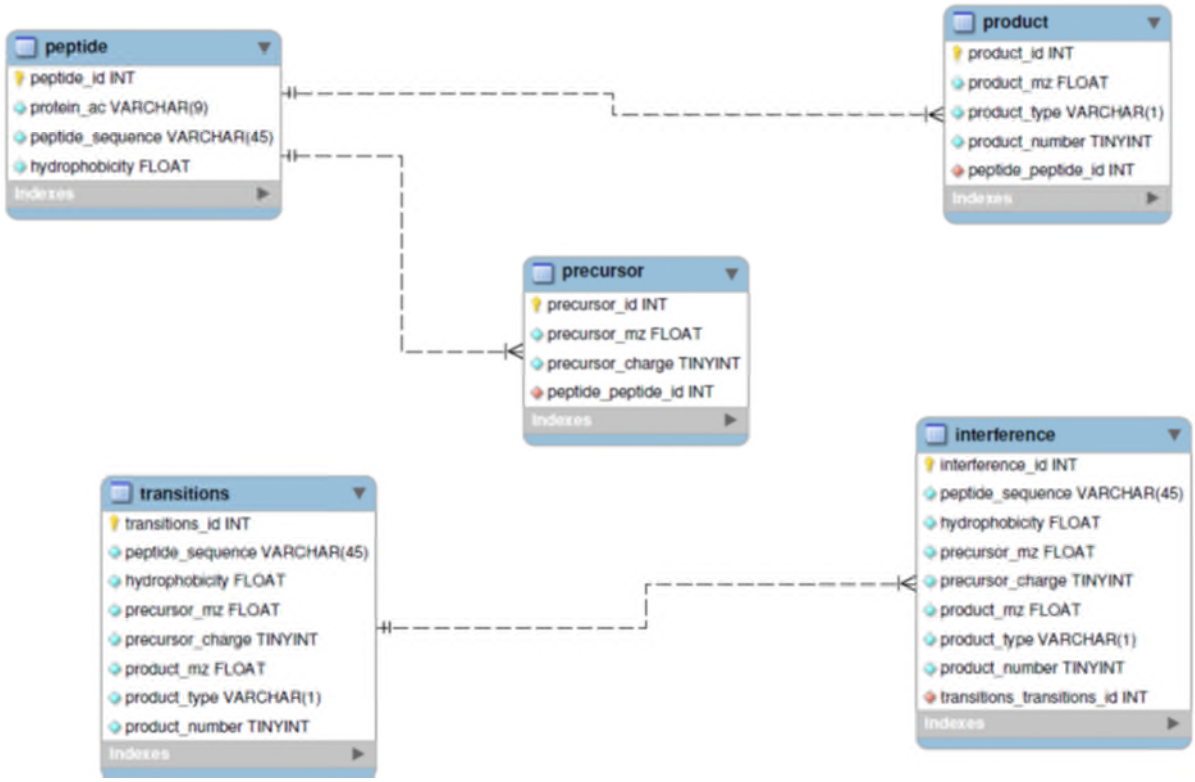


Figure 7. Entity – Relationship diagram for the MRMinter database showing the Schema of the database.

2.1.2 Population of Database

Using the LAMP platform is the preferred course for bioinformaticians and biologists as it is open source and hence access to a wealth of information that can enhance any project due to the information being free. The database management system used is MySQL. It is an SQL based database system. It is freely downloaded from the main MySQL website. The MySQL database allows one to query data between two values. These values can be either decimal or integer values. The Perl programming language is then used for data input and to query the database. Though there is a host of perl tools in the public domain the scripts were generated in house. The information for the proteins were

downloaded from the Swiss-Prot database and this ensures richly annotated and robust data. An organisation of the data is highlighted below.

Perl Scripts – Perl is the scripting language of choice within the bioinformatics community due to its text manipulation abilities and the fact that it is open source. The main perl scripts written for this project are Populate_mrminter.pl, Populate_mrminter_transitions.pl, retrieve_interference.pl and retrieve_interference2.pl .

Populate_mrminter.pl is the program that populates the MRMinter database with the peptide, precursor and product information. The raw data is downloaded from the latest Swiss-Prot release The Perl program then performs a conversion of the peptide amino acid sequence into a mass to charge ratio and also calculates the charge and hydrophobicity of the peptide. There are two ways by which the mass of peptides can be calculated (Jones and Hubbard, 2010), by the average mass or mono-isotopic mass. In this case the mono-isotopic mass was option was taken as MS identifies amino acids by their mono-isotopic masses.

Data Source – Peptides are derived from proteins and the proteins used in populating the MRMinter database were retrieved from the Swiss-Prot database. The Swiss-Prot database is part of the UniProt database. This database is maintained up of the Swiss institute of Bioinformatics (SIB), the European Bioinformatics Institute (EBI) and the Protein Information Resource (PIR). The UniProt database consists of four different main databases namely Uniprot knowledgebase (UniProt KB) of which consists of two sections: Swiss-Prot and Translations of EMBL (TrEMBL), UniMes, UniParc and UniRef. Swiss-Prot which is the source of the proteome data is a protein sequence database that is manually annotated, reviewed and provides high quality non-redundant data. According to UniProt about 98% of the protein sequences available from them are derived from the translation of coding sequences that have been submitted to the public nucleic databases.

The complete proteome sets containing the protein sequences can be retrieved from the UniProt website by searching the taxonomic identifier of your target

organism, with the keyword “complete proteome and reviewed” in the query box. This brings up a list of proteins and clicking the orange download button take the user to a new screen with a list of possible formats with which to download the information. The formats available are Tab-Delimited, Excel, FASTA, GFF, Flat text, XML and RDF/XML. The format chosen is the FASTA format as it is suitable for parsing using the perl programming language. Otherwise it is also possible to download the complete proteome programmatically at every UniProt release and on the website is an example of how to do it using the perl programming language. It is also possible to download the information directly by FTP as the UniProt FTP server caters for downloading expanded FASTA sets which contain both the canonical and manually reviewed isoforms available for the most widely used complete proteomes. Although the data obtained from Swiss-Prot is of high quality it is worthy to note that there are protein sequences which contain ambiguous amino acids.

Peptide sequence digestion – The tryptic digest is performed on the data of the human proteome obtained from Swiss-Prot in FASTA format. This digestion is carried out according to the rules of the trypsin enzyme proteolysis (Switzar et al., 2013), it cuts the sequence after Lysine (K) and Arginine (R) but not before Proline (P). That is to say it cleaves the protein sequence after each lysine or arginine unless lysine or arginine is followed by proline. The main reason for the use of trypsin is that mass spectrometry experiments works more efficiently with peptides up to 3Kda and trypsin usually produces peptides of this size. Also as trypsin cuts the peptide sequence on the C-terminal side of arginine and lysine, it guarantees that the peptides will fragment in a across the whole peptide backbone in a predictable manner. This is because it has been observed that when a basic residue such as arginine is present in the middle of the peptide sequence the fragmentation of the sequence is discontinuous with some bonds remaining intact.

Sorting the peptides- Post digestion, peptide sequences from 3 to 30 amino acids long are chosen. The usual maximum length of peptides used in many

SRM experiments is 25 amino acids. In order to make sure information within the database was as robust as possible and as also according to (Han and Higgs, 2008) peptides which contain approximately 7-30 amino acid residues are generally optimal for MRM analysis. Therefore the maximum peptide length threshold was increased to 30 amino acids. Within the theoretical protein sequences submitted to Swiss-Prot are ambiguous amino acids (X B Z J). Any peptides containing these amino acids are discarded.

Hydrophobicity- The hydrophobicity of the peptide is calculated according to rules laid out by (Krokhin et al., 2004) in their Sequence-Specific Retention calculator (SSRcalc) peptide retention prediction algorithm.

The hydrophobicity of the protein and peptide is correlated with the amount of time the peptide takes to pass through the LC column. This elution time is commonly called Retention time and is a very important factor in mass spectrometry as it allows specific identification of a certain peptide to a certain degree. The calculation for hydrophobicity used in this thesis is derived from (Krokhin et al., 2004) (SSRcalc). This formula for calculating the hydrophobicity of an amino acid sequence also takes into account the influence of peptide length (N) on the hydrophobicity of the peptide (Krokhin et al., 2004) and is calculated as follows

$$H = KL^{*}(\sum R_c + 0.42R^{1}_{cNt} + 0.22R^{2}_{cNt} + 0.05R^{3}_{cNt}) \text{ if } H < 38$$

$$\text{And } H = KL^{*}(\sum R_c + 0.42R^{1}_{cNt} + 0.22R^{2}_{cNt} + 0.05R^{3}_{cNt}) - 0.3(KL^{*}(\sum R_c + 0.42R^{1}_{cNt} + 0.22R^{2}_{cNt} + 0.05R^{3}_{cNt}) - 38) \text{ if } H \geq 38.$$

Where H = hydrophobicity, if N < 10, KL = 1-0.027*(10-N); if N > 20, KL = 1-0.014*(N-20); Otherwise KL = 1, R_c = Retention coefficient and R_{cNt} = a second set of retention coefficients.

Equation 1

Precursor m/z – following ionization by ESI and MALDI as described in chapter 1, the ion mass to charge ratio (m/z) of the peptide is stored in the database. Singly, doubly and triply charged precursor ions are stored as these are the most readily observed ions. The m/z is calculated by the following formula

$$m/z = ([M + (nX)])/n$$

Equation 2

“M is the molecular mass of the peptide, X is the mass of the cation added to the molecule and n is the integer representing the number of charges” (Simpson, 2002).

Precursor charge – This field identifies the charge state of the precursor ion linked to it.

Product m/z – The peptide amino acid sequence is also broken down into theoretical fragments according to rules that calculate the fragmentation of a peptide sequence into b and y ions. As mentioned earlier this fragmentation is meant to duplicate the process that takes place in a collision cell during collision induced dissociation. In experimental situations during collision induced dissociation mostly y ions and low m/z b ions are readily observed. In the database all the theoretical fragment pieces are stored.

Following the theoretical fragmentation of the peptide using the principles for fragmentation in a collision cell the resulting fragment ions of type b and y are stored in the database in the product table. The b ions are the sum of the residue masses from the first AA on the Amino (NH_2) -> Carboxy (COOH) direction in the peptide sequence to the last AA plus the mass of the hydrogen ion while the y ions are the sum of the residue masses from the first AA on the Carboxy (COOH) -> Amino (NH_2) direction in the peptide sequence to the last AA plus the mass of water and a hydrogen ion.

Other perl scripts are written to extract the necessary information from the database such as count of interference, various peptide precursor and product ion combinations to form a transition.

Populate_mrminter_transitions.pl

The purpose of this perl script is to populate the transition and interference tables of the MRMinter database. The information for which transition to insert into the transition table is obtained from the MRMaid database. It can also be obtained from a text file if the information is stored in text format. These transitions are then run through the MRMinter database to make sure they match, are within the database and also to obtained the hydrophobicity, precursor m/z and product m/z values which are used to query the database for interferences. Once the correct transitions are found they are then inserted into the transition table and then a query is run to search for the interferences. The resulting transitions that are flagged are inserted into the interference table as interferences to the target transition.

It is possible to retrieve information about interfering transitions from the database without using the transition table. The transition table offers advantages such as speed as the information does not have to be retrieved from a join on the peptide, precursor and product table as it has been stored earlier and updated when necessary.

In future the script can be modified to check the regular Swiss-Prot update via ftp transfer. The main reason for the design of this system is to assist researchers in making an informed choice when selecting transitions for their own SRM assay design. The system is intended to be coupled with the MRMaid transition design tool the in house MRM transition design tool in order to help identify any interferences pertaining to those transitions.

2.1.3 The MRMinter output

Retrieve_interference.pl - This perl script retrieves the interfering transitions from the transition table. This information can be returned as a .csv file or directly to a web interface.

Retrieve_interference2.pl - This perl script returns a count of the number of interferences and is suitable if that is the only information required. It can also be returned as a .csv file or directly to a web interface.

The results generated can be printed directly onto screen or exported or converted to CSV format to enable integration to other analysis programs.

2.1.4 The Resolution parameters

According to (Abbatiello et al., 2010), an interference occurs when a transition other than that being monitored appears within ±0.5-1.0Da of the monitored transition. The resolution parameters are the different resolutions entered by researchers into the MRMaid database for m/z tolerance and retention time (RT) tolerance. The MRMinter database only holds the raw information for hydrophobicity and is not converted to RT. A formula was derived taking into account the relationship of hydrophobicity with RT and using this to give a tolerance value for constant hydrophobicity. According to (Krokhin and Spicer, 2009) the observed retention time of peptides is correlated with their calculated hydrophobicity through their SSRcalc method. Using the SSRcalc algorithm for the calculation of hydrophobicity where,

Retention Time = a + b*hydrophobicity where intercept (a) is the gradient delay time (differs for individual HPLC systems) and slope (b) is a value related to the slope of acetonitrile gradient (Krokhin and Spicer, 2009).

Using the above equation to estimate tolerance,

$\Delta H = \Delta RT/b$. where ΔH is the hydrophobicity tolerance, ΔRT is the Retention Time tolerance.

To determine which mass resolutions to consider, the PRIDE database (Vizcaino et al., 2013), was interrogated to see which combination of precursor and product tolerance had been used in submitted experimental data. A summary of this information is shown in Table 1.

Table 1. A table showing the resolution parameters (m/z) used to test the MRMinter tool. These parameters were derived from the PRIDE database.

Precursor Tolerance (m/z)	Product Tolerance (m/z)
0	0
0.04	0.8
0.6	0.4
0.6	0.6
1	0.7
1	0.8
1.5	0.7
1.5	1
2.5	0.7
3	0

2.1.5 Evaluation of the database

The database content was evaluated and the system was tested using data from (Anderson and Hunter 2006) and datasets from the MRMaid transition database. The Anderson and Hunter study was aimed at studying proteotypic

peptides and showing the importance of choosing the right peptides to use in SRM based transition design. The proteotypic peptides were run through the database with a query to identify the transition interferences using the tolerance parameters shown above. The Anderson and Hunter study involved 57 proteins and 137 transitions. 18 transitions were stable isotope labelled transitions and these were analysed to judge how useful this tool would be in supporting common absolute quantification methods. Using a SIS labelled transition is a great tool for quantification and is an important factor along with the measurement of the intensity of interferences in validating transitions.

The datasets from the MRMaid transition database were used also to test the MRMinter database. These transitions were peptide transitions from the human proteome and of the 404,000 transitions available the system was able to recognize and test 402,975 transitions.

Using a database populated with the peptides obtained from theoretically digesting proteins from a proteome sourced from a database repository, can sometimes mean that there is the assumption that the peptide to be queried against is available within the database. Therefore the probability that there will be no result from the query exists.

3 Results and Discussion

3.1 The MRMinter database

This thesis aim is to investigate the issue of interfering transitions in SRM experiments. This is achieved through the process of building a peptide database (MRMinter) housing theoretical transitions and using the information to select transitions with precursor and product ions with signals close to the original transitions being designed. MRMinter will be able to assist the researcher performing an SRM assay experiment, in determining the uniqueness of the peptide being monitored. This is achieved by cross-checking the transitions being monitored against the proteome from which the sample is derived, highlighting the count of observed interferences, their description; peptide sequence, hydrophobicity, precursor m/z, precursor charge, product m/z, product type and product number. In order for the investigation of interferences to be performed, a theoretical tryptic digest of the proteome was conducted and this yielded peptides which provided the information for the transitions. This information is in the form of precursor ion m/z and product ion m/z. To increase the specificity of the results, the hydrophobicity of the parent peptide is considered. This monitoring of the hydrophobicity and by extension the elution time of the peptide reduces the number of false positives that the algorithm would raise. The representation of the tryptic peptides between 3 to 30 amino acids is shown in Table 2. An SRM experiment is ideal for peptides between 8-25 amino acids residues. The reason for choosing 3-30 amino acid residues within the database is to serve as an adequate background to search against.

Table 2. A summary of the content of the different tables in the MRMinter database. Most of the data is a derivation from information in peptide table. Shown is the count of Proteins, Peptides, Precursor ions and Product ions.

Protein	Peptide	Precursor	Product
20,220	870,192	2,610,576	17,015,452

The definition of the transitions that occur in a QQQ peptide fragmentation is an essential step in SRM assay development, (de Graaf et al., 2011). The interference issue is a measure of the specificity and selectivity of a transition. This can be described also in terms of the uniqueness of a transition and may be related to the characteristics of the parent peptide. Thus the knowledge of the probability of another peptide possessing the same characteristics as the target peptide is very important. This information also helps in the choice of method in the SRM experiment. The MRMinter database is able to assist researchers increase the selectivity of their methods by having previous knowledge of possible peptides sharing similar properties. The selectivity of the precursor ions, the product ions and the hydrophobicity within a proteome is important. This can be shown as the probability of the occurrence of peptides showing identical values in the form of the same precursor ions, product ions and hydrophobicity. The selectivity of the above values in table 2 within the MRMinter database is measured to give a frequency distribution of transitions within the MRMaid database. MRMinter can help researchers to obtain a picture of the probability of occurrence of a transition having the same SRM behaviour as their transition of interest. This is accomplished by taking into account the individual probabilities of the precursor m/z, the product m/z and hydrophobicity, to give the total probability of having an interfering transition within a result set. According to (Berendsen et al., 2013) the disadvantage of analysing these data by using independent probability statistics is that dependency of the data is assumed to be non-existent. One of the main dependencies that exist is the charge ratio of the precursor ion and the fact that singly charged precursor ions produce product ions with a lower m/z. As a result of this using independent probability statistics in this analysis does not allow for precise measurement of probability but only an indication of the value of selectivity.

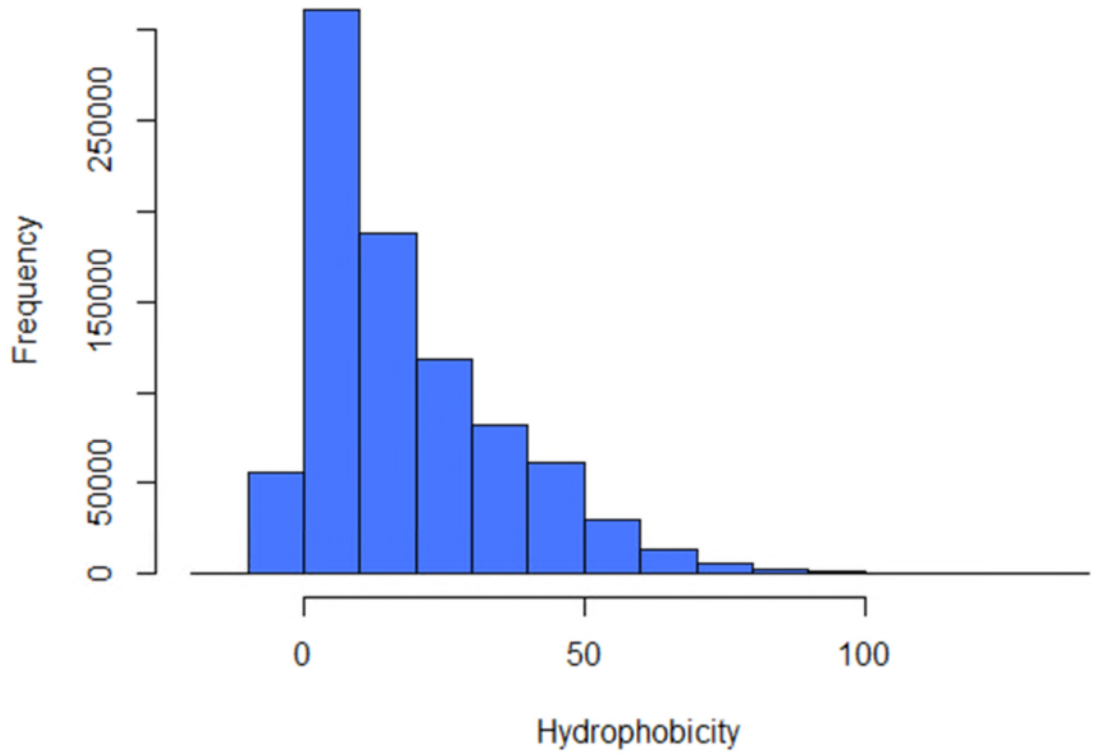


Figure 8. The probability distribution of the hydrophobicity of the peptides. The hydrophobicity was calculated according to Krokhin et al, 2004. Using SSRcalc resulted in some peptides with negative hydrophobicity values.

Figure 8 shows the distribution of the hydrophobicity values within the database. It is observed that the majority of peptides carry a hydrophobicity of 10 units which ensures that elution through a high performance liquid chromatography column is possible and can be used to aid in the selectivity process of SRM. Using the SSRcalc algorithm results in negative values for hydrophobicity for some tryptic peptides. For example tryptic peptide QPPSNPPPRPPAEAR from the protein Cytochrome b-245 light chain has a hydrophobicity value of -1 when calculated with SSRcalc. This should not present a problem in the SRM perspective. The reason is that an offset is added when converting

hydrophobicity to retention time (RT) which means we do not have to worry about negative RTs.

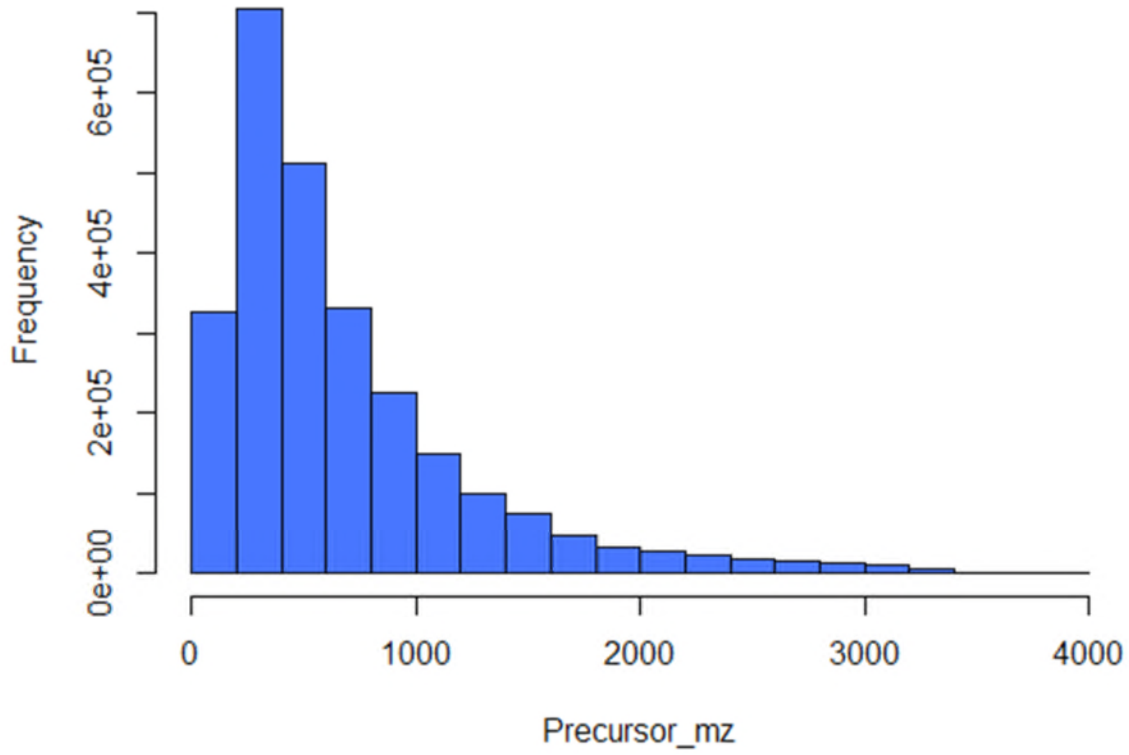


Figure 9. The probability distribution of the precursor ions m/z.

Figure 9 shows the precursor m/z ratio and given the operation of a mass spectrometer is limited to approximately 3kDa, this shows that the SRM technique is suitable for working on tryptic peptides within the human proteome

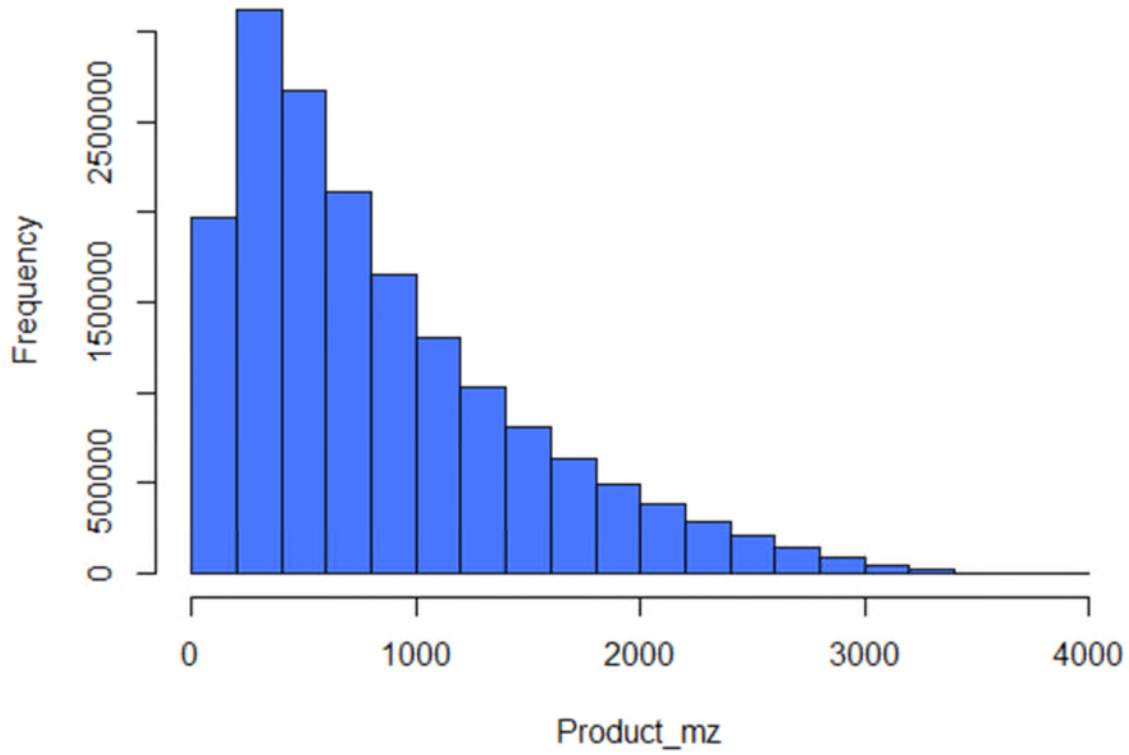


Figure 10. The frequency distribution of the m/z of the product ions

Figure 10 shows the product ions and their distribution within the human proteome. The spread of the m/z ratios also show that detecting the product ion fragments within a SRM experiment is well within the capabilities of the SRM equipment.

The knowledge of peptides showing the same behaviour in an SRM experiment and the probability of them occurring together can be estimated theoretically. This can be of great assistance to researchers by helping them choose the appropriate SRM parameters in tune with their method and equipment.

As explained earlier a transition is the combination of a peptide's precursor ion m/z and its corresponding fragment ion. Figure 11 below shows a bitmapped image of the relationship of all possible transitions within the MRMinter database. This relationship is reflective of the tryptic nature of the peptides. The

total possible number of transitions within the database come to a total of 51,046,356. This figure represents the combination of number of precursor ions, charge state 1, 2 and 3 with the number of product ions, Table 2. This is encouraging as it means that detecting most of the possible interferences that a transition encounters is theoretically possible. The charge states within the database are observed as three spikes, peaking at approximately 3000 product m/z show a correlation to a distribution at 3000, 1500 and 1000 product m/z. Figure 11 shows the hotspots to avoid when seeking appropriate transitions.

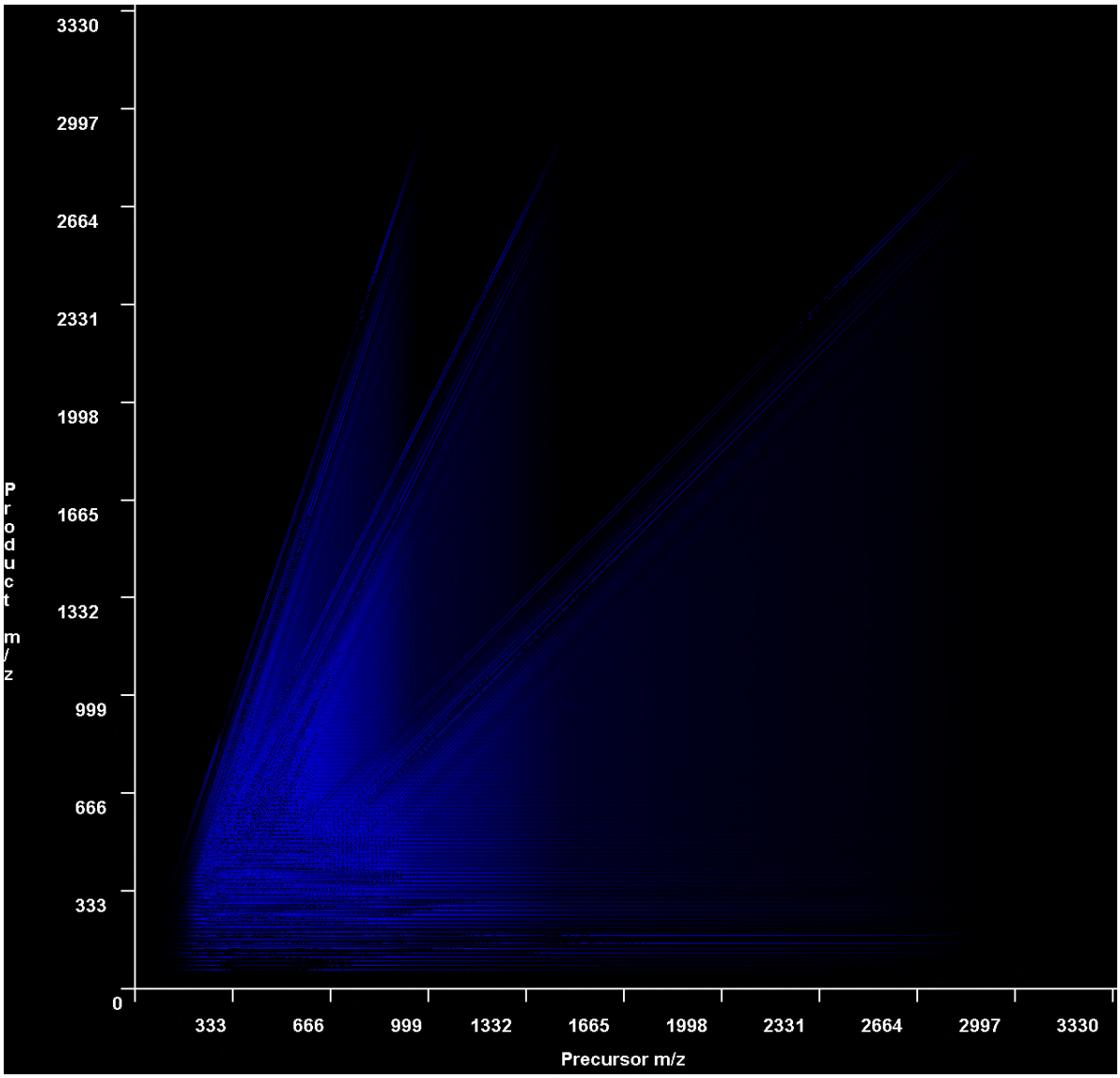


Figure 11. A bitmapped image of the precursor m/z value against the product m/z value of all the peptides contained within the MRMinter database.

3.2 Effect of mass tolerance and chromatographic resolution on interference (Using Anderson and Hunter 2006 transitions).

Anderson and Hunter carried out a study to assess the performance of the MRMs in various typical digest experiments. In the study the value of high and medium abundance plasma proteins as biomarkers makes a case for the application of specific MRM assays. The aim of the study was to find out if MRM assays could be used to quantify the plasma proteins count and to determine the measurement precision. Assays were designed for the purpose of measuring tryptic peptides. The peptides represented 53 high to medium abundance proteins (see Table 4) in human plasma. 137 transitions were derived from the 60 tryptic peptides. Of those 137 transitions, 18 were Stable isotope labelled Standards (SIS). The study was able to demonstrate the accuracy possible using the MRM technique by producing significant quantitative data.

The computational approach to predicting the unique existence of a transition according to (Picotti et al., 2013) is an alternative to experimental methods. By comparing the target transitions with all the other transitions present within the database and generating a list of matches, MRMinter aims to contribute to assay design by increasing specificity through the identification of possible interferences for a given transition.

The search is performed by looking at a range of values within the specified tolerance and any matches indicate an interfering transition. The peptide tolerance, (the error window on experimental peptide mass values) is dependent on the mass spectrometer used and the accuracy of the calibration. This is a case whereby error tolerant searches are permitted in order to find the most optimal tolerance setting that SRM equipment can utilize without losing throughput. The m/z windows which make up the resolution in which the SRM scans each individual transition, allows for increased sensitivity or a dynamic range of one to two orders of magnitude (Shi et al., 2012) compared with the full scan mode of LC-MS/MS. It is of even greater advantage to get the best resolution to maximise the specificity and sensitivity without losing efficiency.

Resolutions queried were those most commonly used in PRIDE during submission of spectra. Below is an example output for a transition and its interferents.

Table 3. An example output of a transition for peptide EIGELYLPK and its interferents. The resolution used is tolerance of 1 for the precursor m/z, 0.7 for the product m/z and 1 hydrophobic unit.

Peptide sequence	hydrophobicity	Precursor m/z	Precursor charge	Product m/z	Product type	Product number
Target : EIGELYLPK	29.6084	531.3	2	633.4	Y	5
interferents						
GMEEGEDNLLCNLR	29.04	531.6	2	633.22	B	6
EQIDLAAR	29.35	531.78	2	633.29	B	5
VYAVEASAIWQQAR	29.66	531.28	3	633.32	B	6
ALEFATLAAR	29.74	571.80	2	633.32	B	6

Table 3 shows an example of a target transition and the interferences it encounters at a particular resolution. Due to the nature of isobaric peptides, filtering out interferences from transitions requires more than just the knowledge of the precursor m/z and product m/z. The hydrophobicity of the peptide sequence plays a very important role in the process. The Retention time is defined as the time it takes for the peptide sequence to elute through the chromatography column and as it is a function of the hydrophobicity of the peptide, it is a major defining factor in distinguishing the transition being observed from any similar signals. Early bioinformatics forays into predicting the interference encountered by transitions, did not take into account the chromatographic behaviour or fragmentation properties of individual peptides (Picotti and Aebersold, 2012).

In the (Anderson and Hunter, 2006) study, 53 proteins were studied. Out of these proteins six of them were not reliably observed. These peptides and their transitions have been validated and studied using stable – isotope labelled peptides. The transitions were run through the database for the ten different

precursor and product tolerance settings most queried in the MRMaid pride database search. For the precursor and product tolerance settings three different hydrophobicity tolerance settings were queried, a tolerance of 1, 2 and 3. These hydrophobicity tolerances are the windows of hydrophobicity searched within the database for the peptide from which the transition is derived. Figure 12 is a graph which highlights the total number of interferences for the different precursor, product and hydrophobicity tolerances.

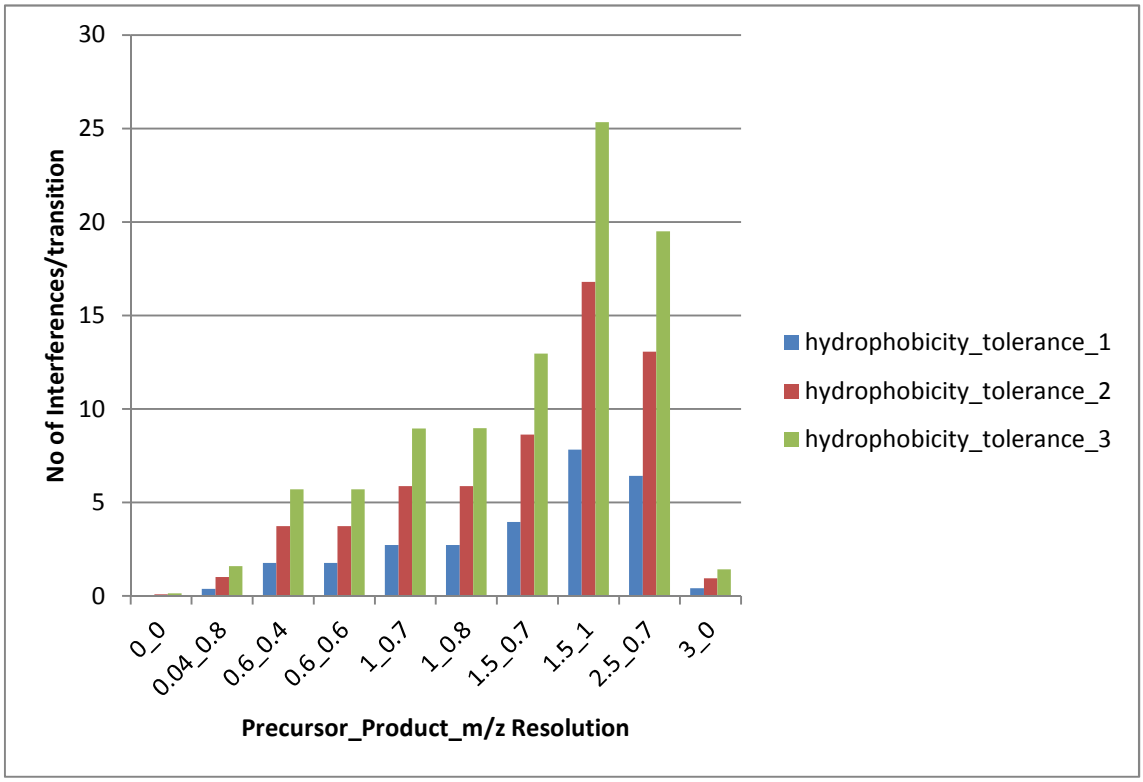


Figure 12. Total number of interferences per transition measured by MRMinter for the different precursor m/z, product m/z and hydrophobicity resolutions using the Anderson and Hunter 2006 dataset. The hydrophobicity tolerance 1, 2 and 3 units.

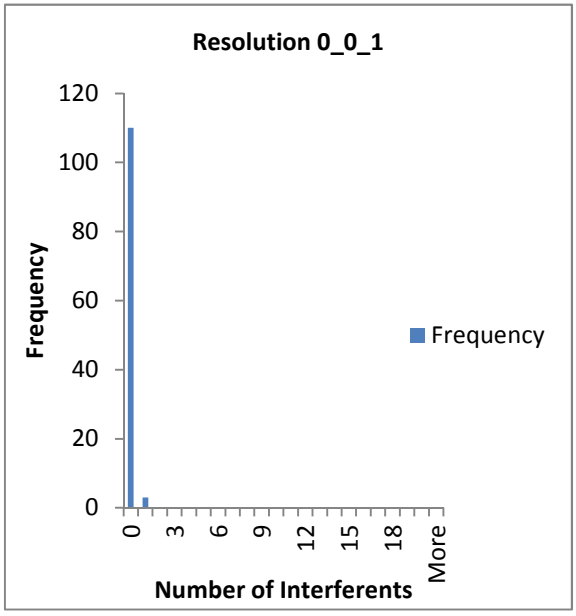
As expected the result shows that the higher the resolution the less the number of interferents. Also it can be assumed that hydrophobicity has a direct co-relation as the interferent counts increased almost exponentially with an increase in tolerance. This same effect was not as pronounced for the precursor m/z and product m/z.

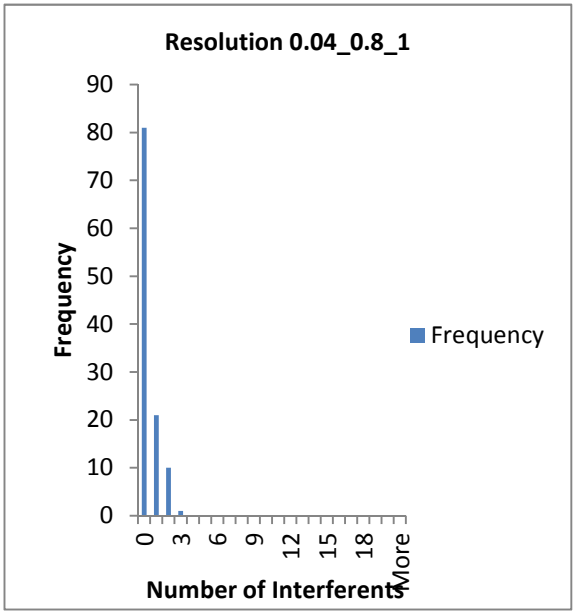
Table 4. A set of the 53 proteins used in the Anderson and Hunter, 2006 study. Also included are the peptides. The transitions used in the study were derived from these peptides.

No	Protein	Peptide
1	Afamin	DADPDTFFAK
2	Alpha-1-acid glycoprotein 1	NWGLSVYADKPETTK
3	Alpha-1-antichymotrypsin	EIGELYLPK
4	Alpha-1B-glycoprotein	LETPDFQLFK
5	Alpha-2-antiplasmin	LGNQEPGGQTALK
6	Alpha-1-antitrypsin	DTEEEDFHVDQVTTVK
7	Alpha-2-macroglbulin	LLIYAVLPTGDVIGDSAK
8	Angiotensinogen	ALQDQLVLVAAK
9	Angiotensinogen	PKDPTFIPAPIQAK
10	Antithrombin-III	DDLYVSDAFHK
11	Apolipoprotein A-I	ATEHLSTLSEK
12	Apolipoprotein A-II precursor	SPELQAEAK
13	Apolipoprotein A-IV	SLAPYAQDTQEK
14	Apolipoprotein B-100	FPEVDVLTK
15	Apolipoprotein B-100	TEVIPPLIENR
16	Apolipoprotein C-I lipoprotein	TPDVSSALDK
17	Apolipoprotein C-II lipoprotein	STAAMSTYTGIFTDQVLSVLK
18	Apolipoprotein C-III	DALSSVQESQVAQQAR
19	Apolipoprotein E	LGPLVEQGR
20	Beta-2-glycoprotein I	ATVVYQGER
21	Beta-2-glycoprotein I	EHSSLAFWK

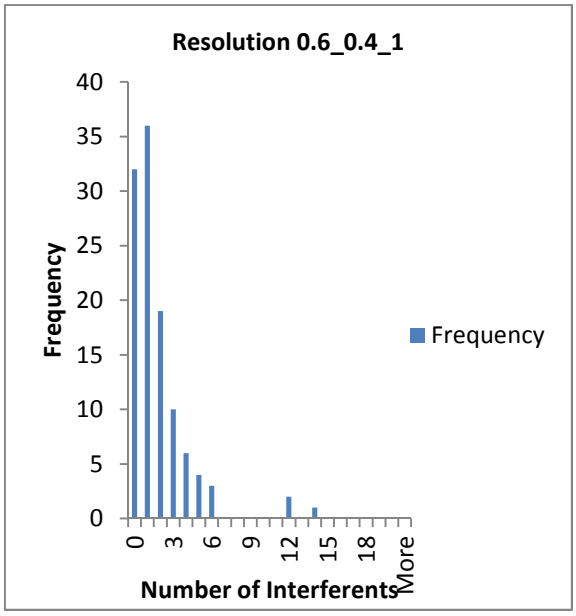
22	C4b-binding protein alpha	LSLEIEQLELQR
23	Ceruloplasmin	EYTDASFTNR
24	Clusterin	LFDSDPITVTVPVEVSR
25	Coagulation factor V	DPPSDLLLLK
26	Coagulation factor XIIa light chain	VVGGLVALR
27	Complement C3	TGLQEVEVK
28	Complement C4 gamma chain	ITQVLHFTK
29	Complement C4 beta chain	VGDTLNLNLR
30	Complement C9	AIEDYINEFSVR
31	Complement factor B	EELLPAQDIK
32	Complement factor H	SPDVINGSPISQK
33	Fibrinogen alpha chain	TVIGPDGHK
34	Fibrinogen alpha chain	GSESGIFTNTK
35	Fibrinogen beta chain	QGFGNVATNTDGK
36	Fibrinogen gamma chain	DTVQIHDITGK
37	Fibronectin	DLQFVEVTDVK
38	Fibronectin	VTWAPPPSIDLTNFLVR
39	Gelsolin isoform I	TGAQELLR
40	Haptoglobin beta chain	VGYVSGWGR
41	Hemopexin	NFPSPVDAAFR
42	Heparin cofactor II	TLEAQLTPR
43	Histidine-rich glycoprotein	DSPVLIDFFEDTER
44	Inter-alpha-trypsin inhibitor heavy chain	AAISGENAGLVR
45	Inter-alpha-trypsin inhibitor light chain	AFIQLWAFDAVK
46	Kininogen	TVGSDTFYSFK

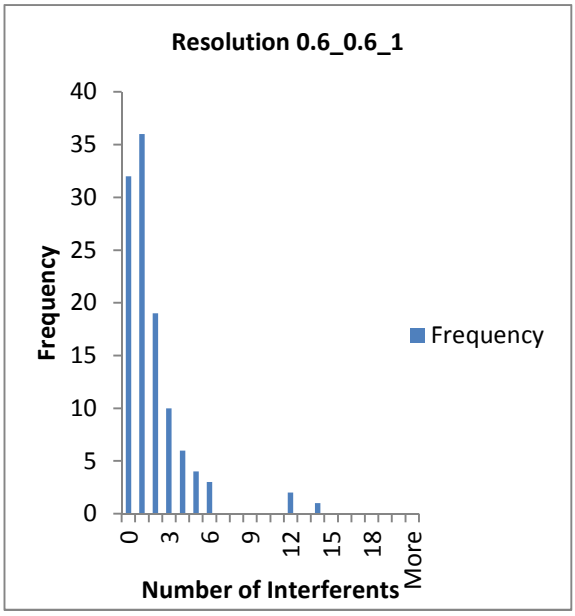
47	L-selectin	AEIEYLEK
48	Plasma retinol-binding protein	YWGVASFLQK
49	Plasminogen	LSSPAVITDK
50	Plasminogen	LFLEPTR
51	Prothrombin	ETAASLLQAGYK
52	Serum albumin	LVNEVTEFAK
53	Serum amyloid P-component	VGEYSLYIGR
54	Transferrin	EDPQTFYYAVAVVK
55	Transthyretin	AADDTWEPFASGK
56	Vitamin D-binding protein	THLPEVFLSK
57	Vitamin K-dependent protein C	WELDLDIK
58	Vitronectin	DVWGIEGPIDAAFTR
59	Vitronectin	FEDGVLDPDYPR
60	Zinc-alpha-2-glycoprotein	EIPAWVPFDPAAQITK



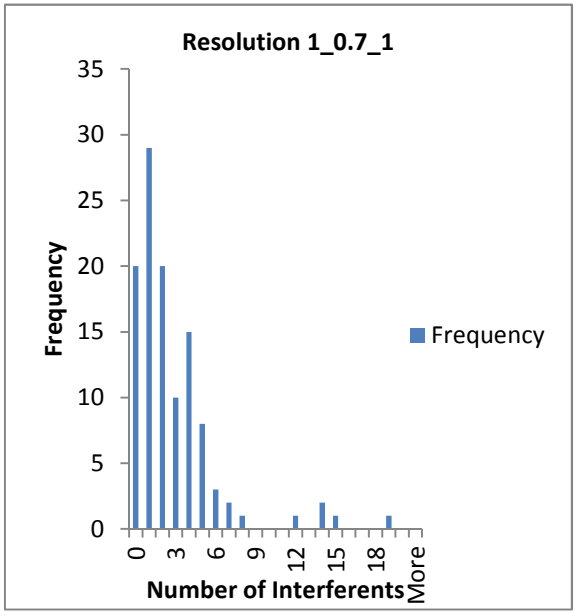


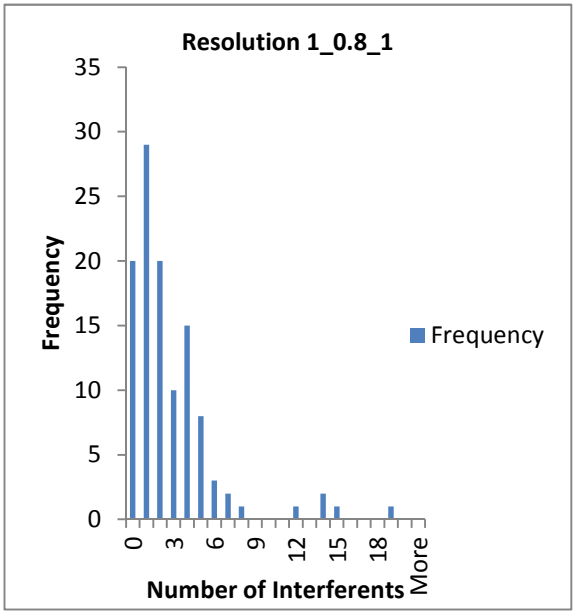
13a 13b



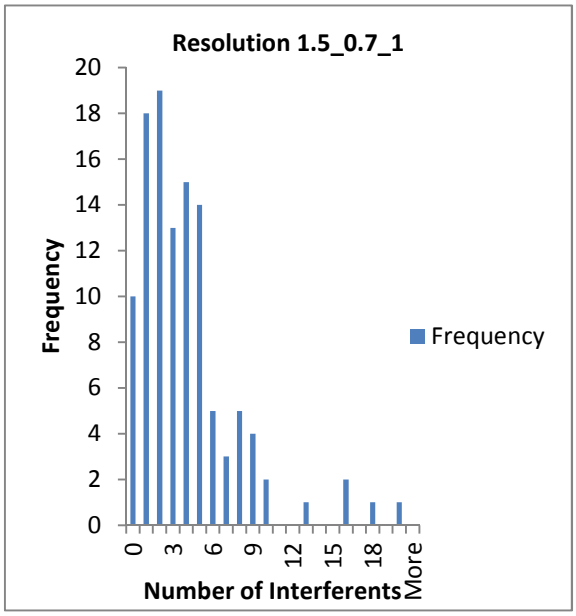


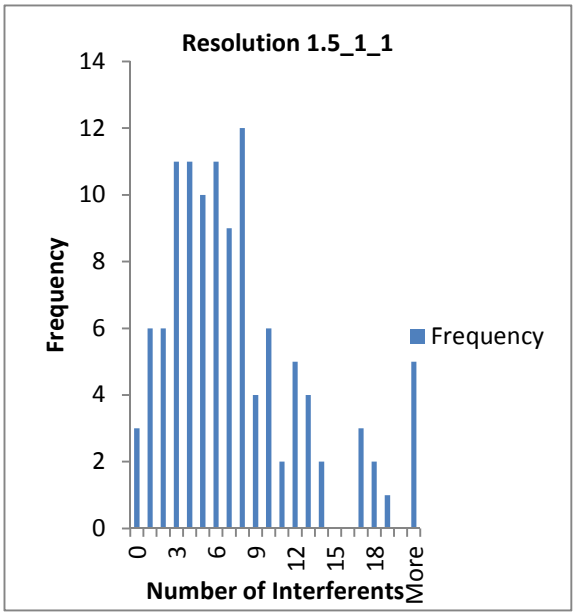
13c 13d



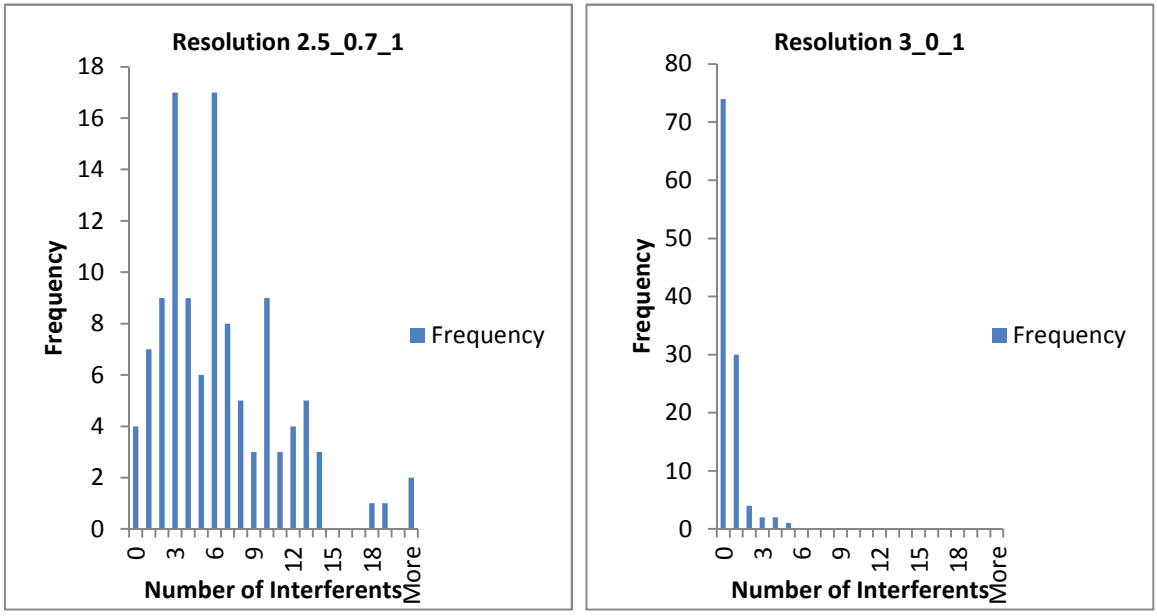


13e 13f



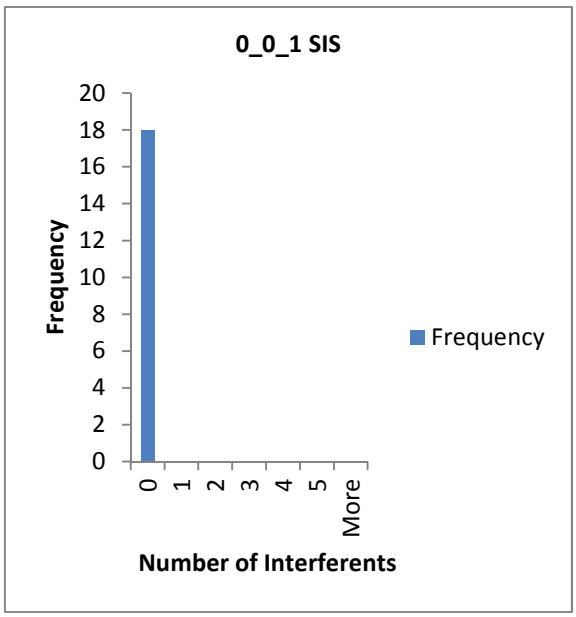


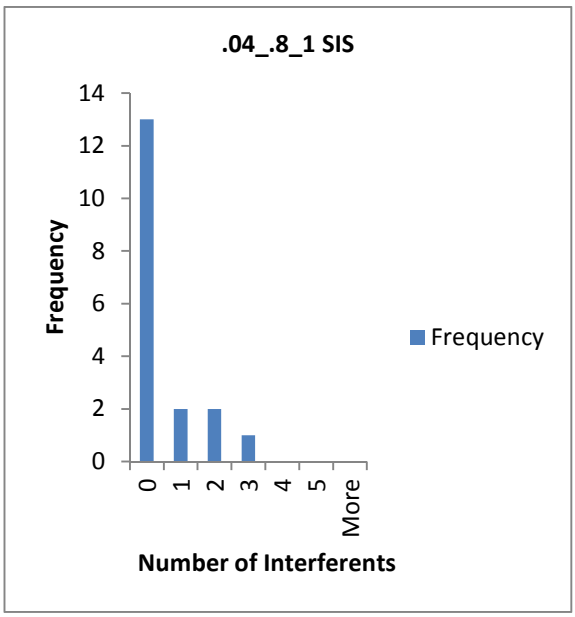
13f 13g



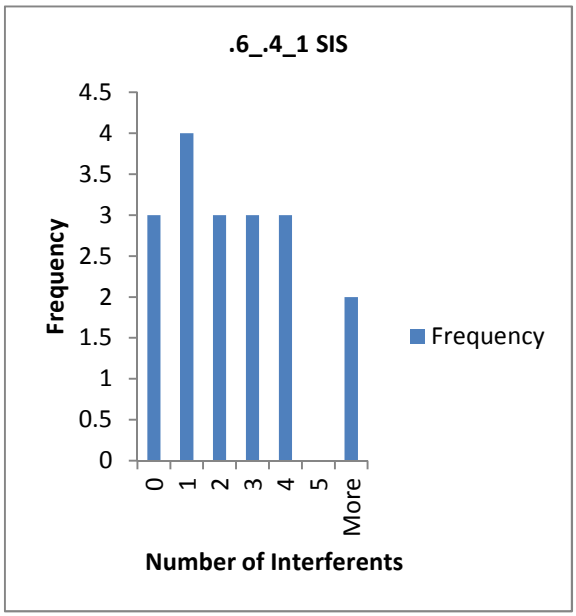
13h	13i

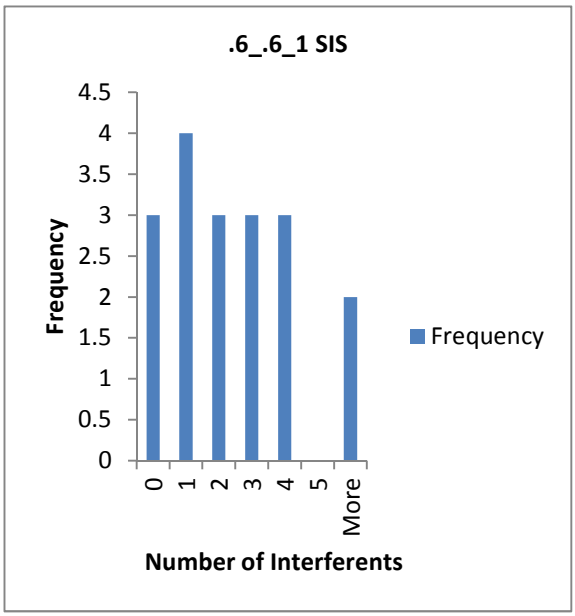
Figure 13. The number of interferents and the frequency with which they occur across the test sample. In this case 119 transitions were taken from Anderson and Hunter 2006. The measurements are for 1 hydrophobicity unit and all ten resolutions specified in Chapter 2, page 39.



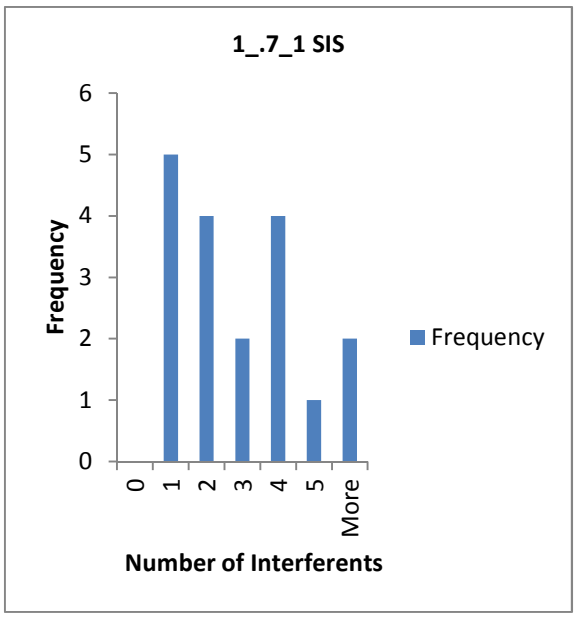


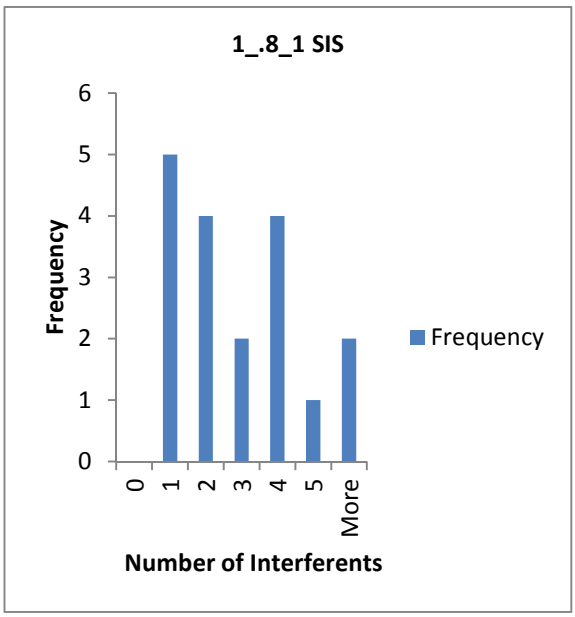
14a 14b



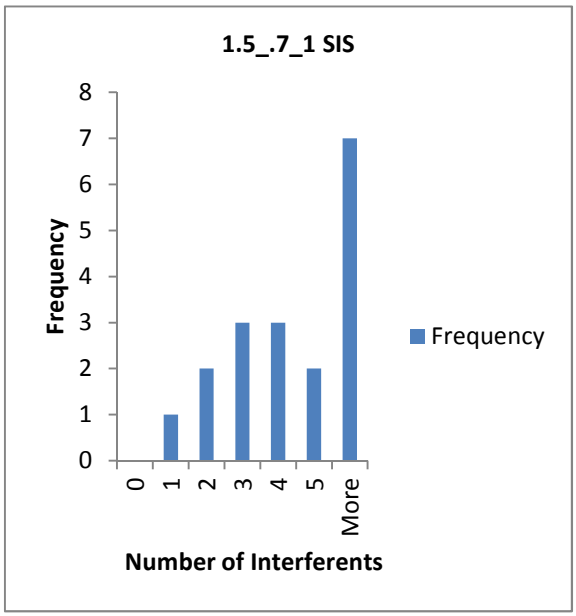


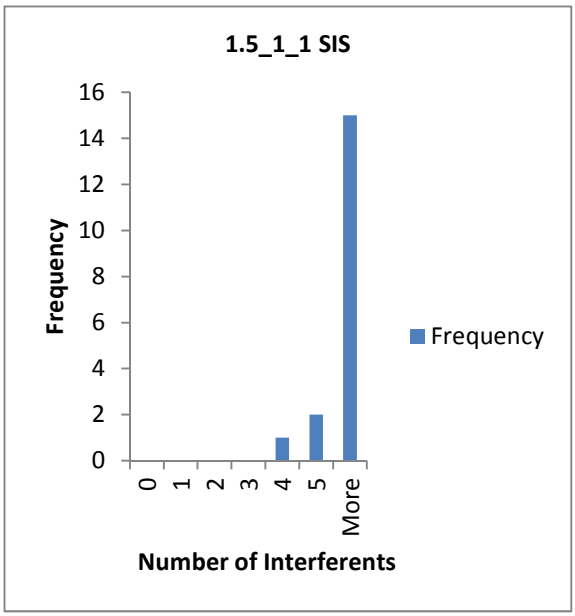
14c 14d



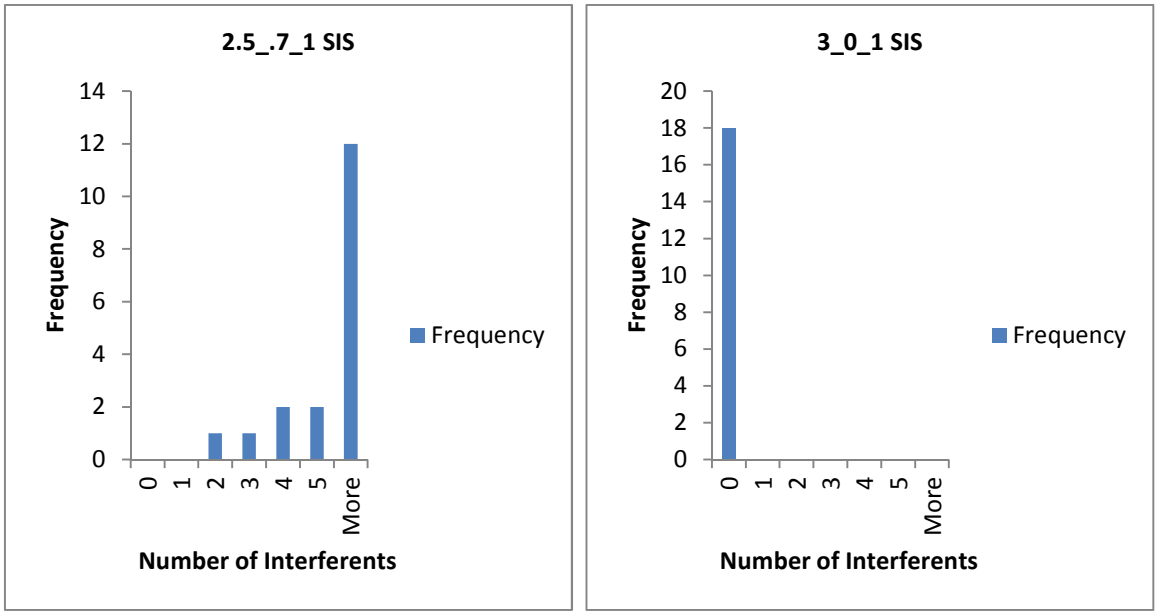


14e 14f





14g 14h



14i 14j

Figure 14. The number of interferents and the frequency with which they occur across the test sample. The transitions are 18 Stable isotope-labelled internal standard (SIS) peptide transitions taken from Anderson and Hunter 2006. The measurements are for 1 hydrophobicity unit and all ten resolutions specified in Chapter 2, page 39.

The first aspect of this study was to find out if the database was able to return any interferences for transitions given a resolution. In an experimental setting a transition's interference will be easily quantified as the desired transition because it will demonstrate close precursor m/z and product m/z values and elute at a retention time very close to that of the specified transition (Yan et al., 2008).

As expected we see almost no interferents for the high resolution setting of (0 precursor and 0 product), figure 13a. Although the tolerance settings of 0 m/z for the product and precursor ions is practically impossible, it has been included as a theoretical case. The most used resolutions, fig 13c and 13d and figure

13b return more interferences but the 0.04 precursor tolerance 0.8 product (figure 13b) resolution's interference values are a lot less showing the importance of the precursor tolerance settings. The (1 precursor and 0.7 and 0.8 product) tolerance, fig 13 e and f also highlight the significance of precursor tolerance as compared to the previous tolerance values a higher precursor tolerance results in more interferences. The (3 precursor and 0 product) tolerance figure 13j also shows very low interference numbers. Considering the sensitivity of SRM equipment it is highly likely that that is a setting that will not be viable to many researchers. The (1.5 precursor and 0.7 and 1 product) tolerances figures 13g and 13h and (2.5 precursor and 0.7 product) tolerance figure 13i show that low resolutions which is normal for older instruments, increase the possibility of introducing errors into the results through inaccurate selection of transitions. This highlights the balance that researchers are trying to keep between sensitivity of the instrument settings and throughput. There might also be a case for the importance of the product m/z tolerance settings as the increase in this is more significant than the increase in the precursor tolerance. This can be seen in the tolerance of (1.5 precursor and 1 product) which gives the highest output of interferences of all the resolutions queried. When the precursor tolerance is increased to (2.5 and product tolerance is .7) the output of interferences is still far below that of the (1.5 precursor and 1 product) resolution. An interesting resolution used is that of tolerance precursor of 3 and product tolerance of 0. There is still a very high output of interferences which means that the upper range of the product m/z tolerance of 0.7 to 1 is where a lot of interferences can be observed.

The analysis was also carried out with 18 Stable isotope-labelled internal standard (SIS) peptides. The distribution of the graphs show that the precursor tolerance is still a very important factor in the observance of interferences. Figure 14i shows that even with a product tolerance of 0.7, the precursor tolerance of 2.5 results in the most interferences in the analysis of the SIS peptide transitions derived from the (Anderson and Hunter, 2006) study. The main point is the fact that the algorithm is able to find transitions that interfere

with SIS transitions. The fact that the number of interferents are strongly related to the resolution tolerance gives more confidence in the tool.

3.3 Effect of mass tolerance and chromatographic resolution on interference (Using MRMaid transitions).

Figure 12 clearly shows that the tolerances increase at an almost exponential level with every increase in tolerance by 1 unit. Therefore the decision was taken to study the effect of hydrophobicity tolerance on the MRMaid data only up to a tolerance of 2 hydrophobic units.

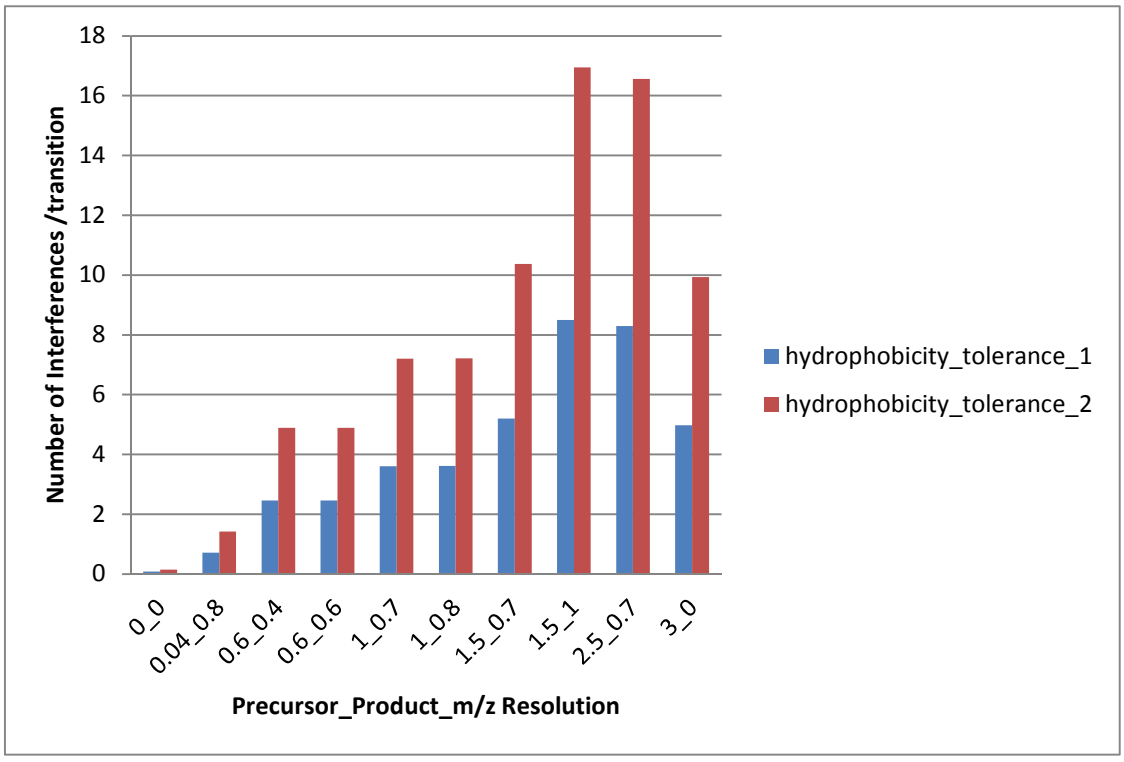
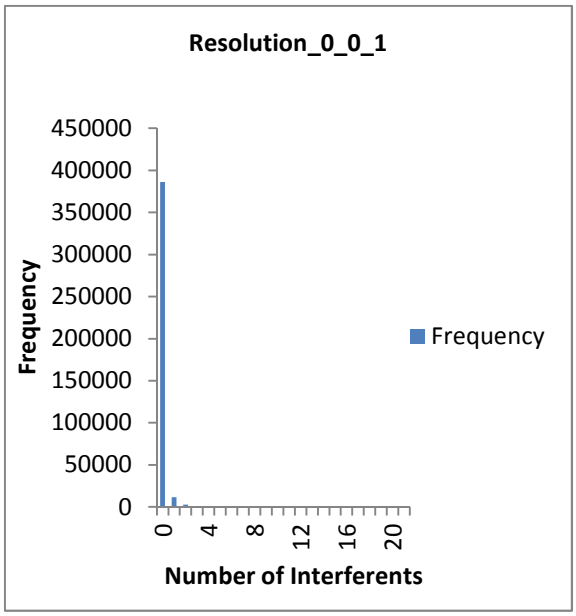


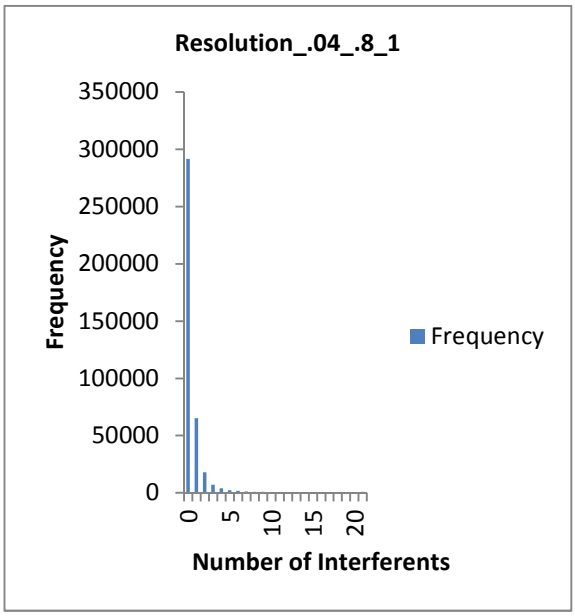
Figure 15. Total number of interferences per transition measured by MRMinter for the different precursor m/z, product m/z and hydrophobicity resolutions using the MRMaid dataset.

Using the Anderson and Hunter transitions only covered a minute fraction of possible transitions. For example in the MRMaid transition database there are 473,373 transitions for the species Arabidopsis (Fan et al., 2012). The human proteome is the subject of study in this thesis and from the MRMaid transition

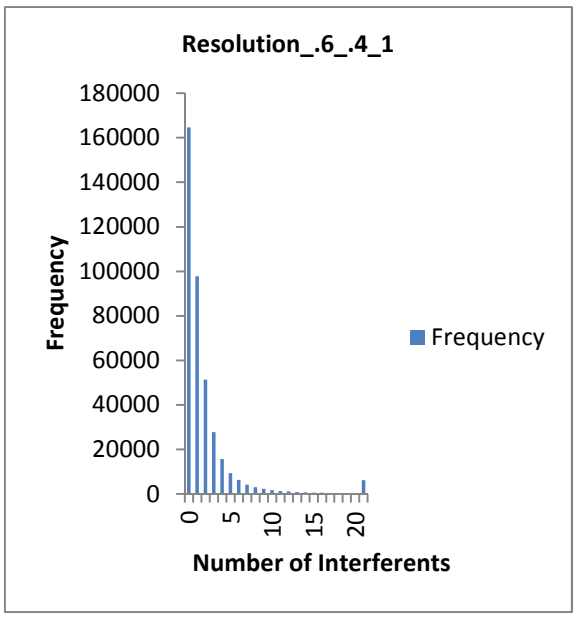
database 404,000 transitions were retrieved. Of these, 402,975 transitions matched with the MRMinter database. The reason for the difference is that the remainder transitions had precursor ions with a charge state of 4 or higher. While this is useful in certain circumstances SRM experiments performed with precursor ions of higher charge states do not give the most accurate results. Transitions with peptide lengths of 7-25 amino acids were queried against the MRMinter database to retrieve interferences. 397757 transitions were analysed. The results are shown in Figure 16a-j.

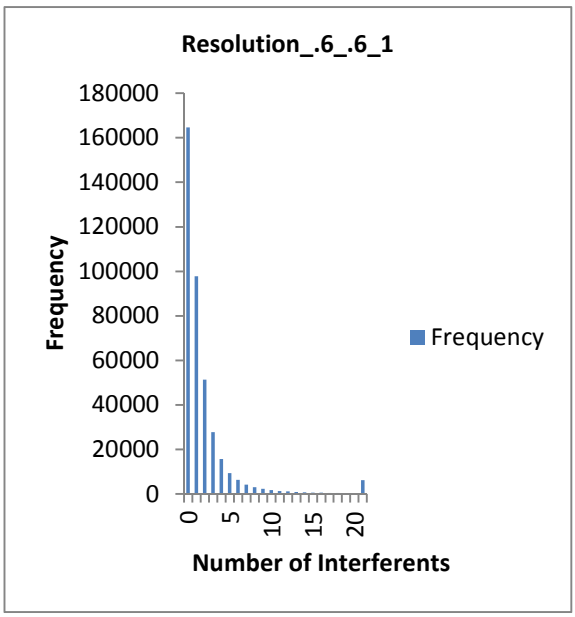
Using the same resolution as that used against the Anderson and Hunter data, it is observed that the resolution with the least number of interferences is 0 precursor and 0 product (figure 16a) which is to be expected as these values are theoretical and only serve to provide a standard to measure against. The 0.04 precursor and 0.8 product (figure 16b) resolution returned the second lowest number of interferences. Of the remainder resolutions the 0.6 precursor and 0.4 product (figure 16c) and 0.6 precursor and 0.6 product (figure 16d) returned the same number of interferences. The use of the 3 precursor and 0 product resolution which seemed to show in the Anderson and Hunter data set, that it was possible to allow such a high tolerance of 3 for the precursor with a product tolerance of 0, returned a higher number of interferences. This is consistent with expectation. For example in the MRMinter human proteome database of 20,220 proteins there are over 870,000 peptides with missed cleavages not considered. In a similar study with a database of approximately the same size, (Li et al., 2009) found that a single precursor ion will match to about 600 peptides within the database if the query is based on a tolerance of 1Da. Therefore even when the product tolerance is still set to 0, the high precursor tolerance will result in a high number of interferences. The best recommended resolutions figure 16b, c and d also contain the most transitions with 0 number of interferences. This is useful when designing SRM experiments as a researcher will like to choose peptides that have at least one transition with 0 interferences.



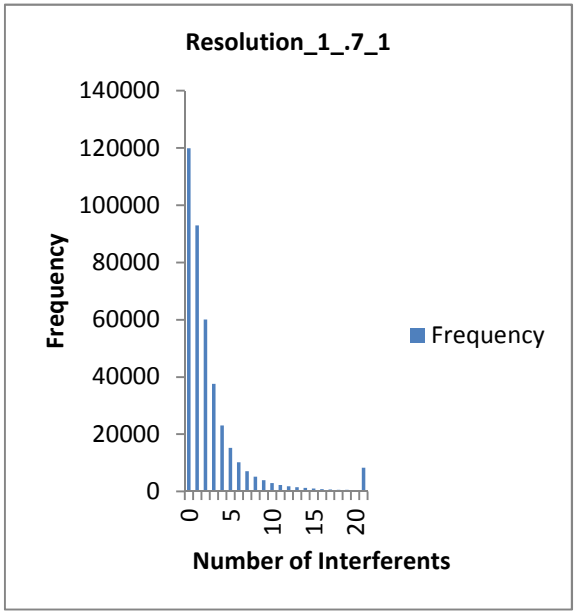


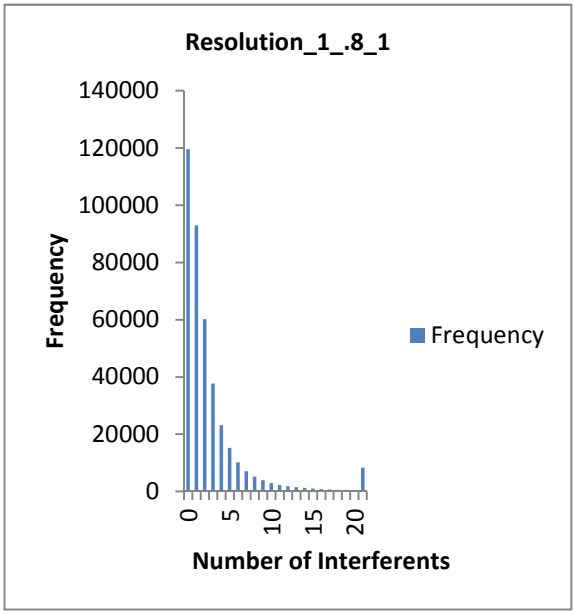
16a 16b



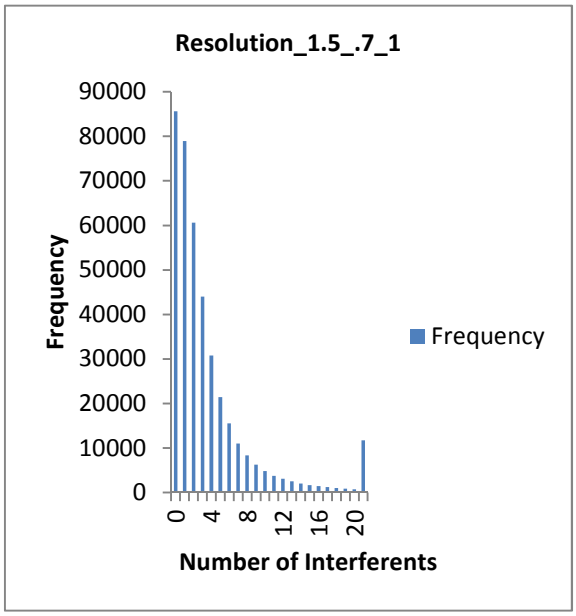


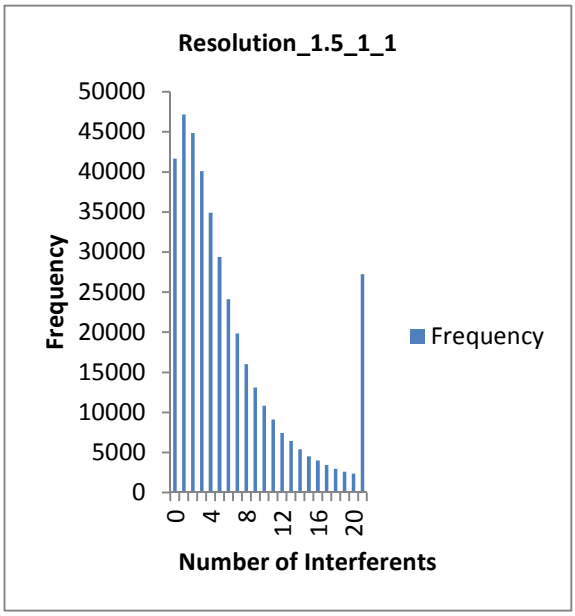
16c 16d



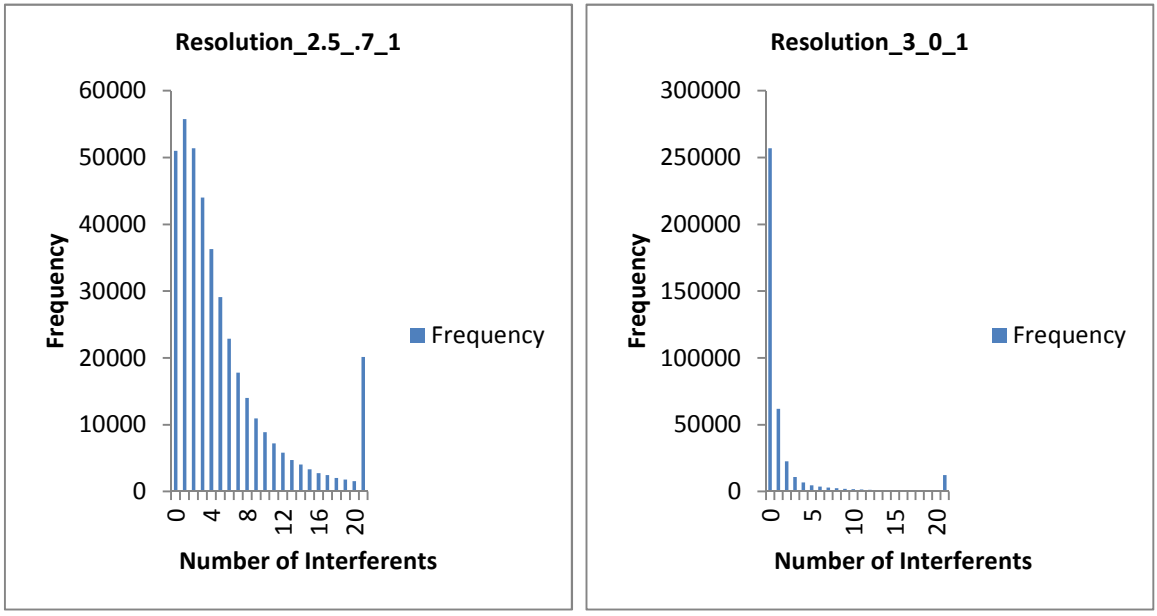


16e 16f





16g 16h



16i	16j

Figure 16. The number of interferences and the frequency with which they occur across the test sample. Transitions in this dataset were obtained from MRMaid. The measurements are for all ten resolutions specified in Chapter 2, page 39.

3.4 MRMInter a step toward better transition selection.

The MRMinter system was developed from a need to increase the selectivity and specificity of SRM assays.

Generally in order to increase specificity multiple fragment ions are selected for each peptide as this reduces the possibility of interferences. MRMinter can help this process by removing the need to select multiple fragment ions as the interferences can be filtered out directly. This is discussed by (Rost et al., 2012) and (Sherman et al., 2009). (Sherman et al., 2009) have tried to explain the need for directed selection of SRM by dealing with interferences.

The results described in 3.3 have shown that the SRM method is possible but researchers have to be careful to choose the right transitions.

The hope is that this work will allow the MRMaid tool to be able to provide a better SRM design platform for researchers and allow the researchers measure the suitability of transitions based on the interferences they encounter from other peptide transitions.

Knowing the coordinates of the interferences will also allow the instrument makers to pre-program those data points for the assay and maximize dwell time (Holman et al., 2012).

(Holman et al., 2012) make a case for selecting transitions. They argue that rather than pre-programming a SRM instrument with the coordinates for interfering peptides in order to filter them out. The best way is to originally choose transitions where the product ion m/z is greater than the precursor ion m/z. The main premise for this is that co-eluting singly charged precursor ions that may interfere at the first quadrupole will inevitably be filtered out in the second quadrupole. Thus increasing selectivity and removing interference.

As observed by (Rost et al., 2012) the understanding of the interference problem can help researchers better understand the fact that identifying a peptide just by monitoring one transition may not by feasible. This may be due to the fact that assay redundancy can become a hurdle when designing and measuring SRM-assays for a given peptide in a highly complex background. The analysis of data in the MRMinter database has shown that there is a great probability of encountering interferences within a sample matrix of proteins and having prior information to aid with the assay design phase will afford researchers a higher degree of freedom.

4 Conclusion

A database has been built and accompanying software algorithms have been developed to assist the MRMaid transition software design program. By virtue of developing the database I was able to discover that the SRM method is a viable method that can be improved with knowledge of interfering transitions. In addition to providing optimal transitions for SRM assays, with this new addition MRMinter, more information regarding interfering peptides can be provided. SRM tools are using spectral libraries to generate their transition environments and the advantages of using this are quite clear.

Chapter 1 provided an introduction to the SRM method and its importance in the field of proteomics. It showed how bioinformatics can be used to make the process quicker and more accurate. Chapter 2 showed the method by which project was to be carried out and highlighted the process of building and manipulating a database to return theoretical results that closely mirror experimental settings. An analysis of the results from the database, software tool and datasets used to test the tool were presented and analysed. As has been mentioned MRMinter shows clearly that the SRM procedure is a viable procedure and any errors that may accompany the method can be clearly controlled with careful observation of the results and experimental conditions.

As long as the peptide is tryptic and between 3-30 sequences long then it will be within the database along with its corresponding data. As mentioned by (Rost et al., 2012) results that are returned by the database are subject to the data acquired from Swiss-Prot thus increasing the chances of false negatives. The false negatives are interferences that cannot be detected due to factors outside our control. Decreasing the occurrence of false negative hits might involve allowing for missed cleavages modifications within the peptide sequence and monitoring other ions other than the b and y ions.

The tool helps move towards completely automated SRM assay design and data analysis. The use of the various tools such as Perl programming language have made the manipulation of data and the ability to connect to the database

using the Perl DBD module has allowed the generation of results sets in formats that are easy to manipulate and analyse. For example connecting to MySQL through Perl has allowed the downloading of .csv files that would not have been easy through MySQL alone. A drawback with Perl is that it does not have the speed of languages such as java and C++. This can affect the time taken to generate the required data.

5 Future Work

For the real generation of an adequate system to aid the design of SRM assays all the scenarios that can occur with respect to the manipulation of proteins via mass spectrometry should be taken into account for example, missed cleavages are a common occurrence in experimental settings as the digestive enzyme might not get 100% coverage of the sample to purification. This can affect protein abundance and by extension quantitation. Immonium ions observed under high energy dissociation must also be taken into account. A lot of SRM experiments are carried out on triple quadrupole spectrometers and low energy CID is used to fragment the precursor ions and though we can expect only y ions and low m/z b ions, other ion fragments can be programmed as an option for researchers to choose from especially if they are working with other types of mass spectrometers and fragmentation takes place by Electron Capture Dissociation etc. They then encounter fragment ions like a, z and c fragment ions. Having a database of interferences for peptides from post translational modified proteins may be appropriate. This may mean that the MRMinter database may have to deviate from its original function as a tool to assist the MRMaid software due to the fact that the MRMaid software does not consider factors like missed cleavages or modified peptides. The MRMinter database can then be further developed into a stand-alone web-based software that can be queried online for interference information.

The database can also be extended to cater for other species and this would also have the added benefit of more robust analysis of the transition interference issue and selectivity. In Mass spectrometry there are peptides which are undetectable and some peptides which though detectable have transitions which cannot be measured. The effect of these peptides can be studied with the inteferent database.

The interferent database is originally designed as a tool to work with the web-based MRMaid search program and its main objective is to assist researchers using the MRMaid program to increase the selectivity of their MRM experiments by having prior knowledge about the uniqueness of their own transitions.

The data generated in this thesis only shows the presence or absence of interferences within the database or proteome. Future work could include generating enough data to show a clear difference between signal and background noise. This will provide enough information to allow the instrument manufacturers to design instruments, with the technology to reduce matrix interferences without losing signal intensity.

REFERENCES

Abbatiello, S. E., Mani, D. R., Keshishian, H. and Carr, S. A. (2010), "Automated detection of inaccurate and imprecise transitions in peptide quantification by multiple reaction monitoring mass spectrometry", *Clinical Chemistry*, vol. 56, no. 2, pp. 291-305.

Aebersold, R. and Cravatt, B. F. (2002), "Proteomics--advances, applications and the challenges that remain", *Trends in Biotechnology*, vol. 20, no. 12, pp. S1-2.

Anderson, L. and Hunter, C. L. (2006), "Quantitative mass spectrometric multiple reaction monitoring assays for major plasma proteins", *Molecular & Cellular Proteomics*, vol. 5, no. 4, pp. 573-588.

Berendsen, B. J., Stolker, L. A. and Nielen, M. W. (2013), "The (un)certainty of selectivity in liquid chromatography tandem mass spectrometry", *Journal of the American Society for Mass Spectrometry*, vol. 24, no. 1, pp. 154-163.

Benyon, R. J., Doherty, M. K., Pratt, J. M. and Gaskell, S. J. (2005), “Multiplexed absolute quantification in proteomics using artificial QCAT proteins of concatenated peptides”, *Nature Methods* vol. 2, no. 8, pp 587-589.

Brun, B., Dupuis, A., Adrait, A., Marcellin, M., Thomas, D., Court, M., Vandenesch, F. and Garin, J. (2007), “Isotope-labeled protein standards: toward absolute quantitative proteomics”, *Molecular & Cellular Proteomics*, vol.6, no. 12, pp. 2139-2149.

Chandramouli, K. and Qian, P. Y. (2009), "Proteomics: challenges, techniques and possibilities to overcome biological sample complexity", *Human Genomics and Proteomics*, vol. 2009, doi: 10.4061/2009/239204.

Chang, C. Y., Picotti, P., Huttenhain, R., Heinzelmann-Schwarz, V., Jovanovic, M., Aebersold, R. and Vitek, O. (2012), "Protein significance analysis in selected reaction monitoring (SRM) measurements", *Molecular & Cellular Proteomics*, vol. 11, no. 4, doi: 10.1074/mcp.M111.014662.

Cham Mead, J. A., Bianco, L. and Bessant, C. (2010), "Mining proteomic MS/MS data for MRM transitions", *Methods in Molecular Biology*, vol. 604, pp. 187-199.

Chevalier, F. (2010), “Standard dyes for total protein staining in gel-based proteomic analysis”, *Materials*, vol. 3, pp. 4784-4792.

De Graaf, E. L., Altelaar, A. F., van Breukelen, B., Mohammed, S. and Heck, A. J. (2011), "Improving SRM assay development: a global comparison between triple quadrupole, ion trap, and higher energy CID peptide fragmentation spectra", *Journal of Proteome Research*, vol. 10, no. 9, pp. 4334-4341.

Deutsch, E. W., et al. (2012), "TraML--a standard format for exchange of selected reaction monitoring transition lists", *Molecular & Cellular Proteomics*, vol. 11, no. 4, doi: 10.1074/mcp.R111.015040.

Eng, J. K., McCormack, A. L., and Yates, J. R. (1994). "An Approach to Correlate Tandem Mass Spectral Data of peptides with Amino Acid Sequences in a Protein Database", *Journal of American Society for Mass Spectrometry*, vol. 5, no. 11, 976-989.

Elliott, M. H., Smith, D. S., Parker, C.E and Borchers, C. (2009), “Current trends in quantitative proteomics”, *Journal of Mass Spectrometry*, vol. 44, no. 12, pp.1637-1660.

Fan, J., Mohareb, F., Jones, A. M. and Bessant, C. (2012), "MRMaid: The SRM Assay Design Tool for Arabidopsis and Other Species", *Frontiers in Plant Science*, vol. 3, pp. 164.

Farrah, T., et al (2011), "A high-confidence human plasma proteome reference set with estimated concentrations in PeptideAtlas", *Molecular & Cellular Proteomics*, vol. 10, no. 9, doi: 10.1074/mcp.M110.006353.

Forner, F., Foster, L. J. and Toppo, S. (2007), "Mass spectrometry data analysis in the proteomics era", *Current Bioinformatics*, vol. 2, no. 1, pp. 63-93.

Gerber, S.A., Rush, J., Stemman, O., Kirschner, M. W. and Gygi, S. P (2003) Absolute quantification of proteins and phosphoproteins from cell lysates by tandem MS. *The Proceedings of the National Academy of Sciences USA*, vol. 100, 6940-6945.

Han, B. and Higgs, R. E. (2008), "Proteomics: from hypothesis to quantitative assay on a single platform. Guidelines for developing MRM assays using ion trap mass spectrometers", *Briefings in Functional Genomics & Proteomics*, vol. 7, no. 5, pp. 340-354.

Holman, S. W., Sims, P. F. and Eyers, C. E. (2012), "The use of selected reaction monitoring in quantitative proteomics", *Bioanalysis*, vol. 4, no. 14, pp. 1763-1786.

Hunter, C. (2010), "MRM^3 Quantitation for Highest Selectivity of Proteins in Complex Matrices", *Journal of .Biomolecular Techniques*, vol. 21, no. 3 Suppl, pp. S34-5.

Jones, A. R. and Hubbard, S. J. (2010), "An introduction to proteome bioinformatics", *Methods in Molecular Biology*, vol. 604, pp. 1-5.

Karp, N. A. and Lilley, K. S. (2007), "Design and analysis issues in quantitative proteomics studies", *Proteomics*, vol. 7 Suppl 1, pp. 42-50.

Kito, K. and Ito, T. (2008), "Mass spectrometry-based approaches toward absolute quantitative proteomics", *Current Genomics*, vol. 9, no. 4, pp. 263-274.

Kiyonami, R., Schoen, A., Prakash, A., Peterman, S., Zabrouskov, V., Picotti, P., Aebersold, R., Huhmer, A. and Domon, B. (2011), "Increased selectivity, analytical precision, and throughput in targeted proteomics", *Molecular & Cellular Proteomics*, vol. 10, no. 2, doi: 10.1074/mcp.M110.002931.

Krokhin, O. V., Craig, R., Spicer, V., Ens, W., Standing, K. G., Beavis, R. C. and Wilkins, J. A. (2004), "An improved model for prediction of retention times of tryptic peptides in ion pair reversed-phase HPLC: its application to protein peptide mapping by off-line HPLC-MALDI MS", *Molecular & Cellular Proteomics*, vol. 3, no. 9, pp. 908-919.

Krokhin, O. V. and Spicer, V. (2009), "Peptide retention standards and hydrophobicity indexes in reversed-phase high-performance liquid chromatography of peptides", *Analytical Chemistry*, vol. 81, no. 22, pp. 9522-9530.

Kumar, C. and Mann, M. (2009), "Bioinformatics analysis of mass spectrometry-based proteomics data sets", *FEBS Letters*, vol. 583, no. 11, pp. 1703-1712.

Lai, X., Wang, L and Witzmann, F. A. (2013), “Issues and applications in label-free quantitative mass spectrometry”, *International Journal of Proteomics*, vol. 2013, 756039.

Lam, H., Deutsch, E. W., Eddes, J. S., Eng, J. K., King, N., Stein, S. E. and Aebersold, R. (2007), "Development and validation of a spectral library searching method for peptide identification from MS/MS", *Proteomics*, vol. 7, no. 5, pp. 655-667.

Lange, V., Picotti, P., Domon, B. and Aebersold, R. (2008), "Selected reaction monitoring for quantitative proteomics: a tutorial", *Molecular Systems Biology*, vol. 4, pp. 222.

Li, G, Vissers, J. P. C., Silva, J. C., Golick, D., Gorenstein, M. V. and Geromanos, S. J. (2009), "Database searching and accounting of multiplexed precursor and product ion spectra from the data independent analysis of simple and complex peptide mixtures", *Proteomics*, vol. 9, no. 6, pp. 1696-1719.

MacLean, B., Tomazela, D. M., Shulman, N., Chambers, M., Finney, G. L., Frewen, B., Kern, R., Tabb, D. L., Liebler, D. C. and MacCoss, M. J. (2010), "Skyline: an open source document editor for creating and analysing targeted proteomics experiments", *Bioinformatics*, vol. 26, no. 7, pp. 966-968.

Mead, J. A., Bianco, L., Ottone, V., Barton, C., Kay, R. G., Lilley, K. S., Bond, N. J. and Bessant, C. (2009), "MRMaid, the web-based tool for designing multiple reaction monitoring (MRM) transitions.", *Molecular & Cellular Proteomics*, vol. 8, no. 4, pp. 696-705.

Messana, I., Cabras, T., Iavarone, F., Vincenzoni, F., Urbani, A. and Castagnola, M. (2013), "Unraveling the different proteomic platforms", *Journal of Separation Science*, vol. 36, no. 1, pp. 128-139.

Perkins, D. N., Pappin, D. J. C., Creasy, D. M., and Cottrell, J. S. (1999). "Probability-based protein identification by searching sequence databases using mass spectrometry data", *Electrophoresis* vol 20, no.18, 3551–3567.

Picotti, P. and Aebersold, R. (2012), "Selected reaction monitoring-based proteomics: workflows, potential, pitfalls and future directions", *Nature Methods*, vol. 9, no. 6, pp. 555-566.

Picotti, P., Bodenmiller, B. and Aebersold, R. (2013), "Proteomics meets the scientific method", *Nature Methods*, vol. 10, no. 1, pp. 24-27.

Prakash, A., Tomazela, D. M., Frewen, B., Maclean, B., Merrihew, G., Peterman, S. and Maccoss, M. J. (2009), "Expediting the development of targeted SRM assays: using data from shotgun proteomics to automate method development", *Journal of Proteome Research*, vol. 8, no. 6, pp. 2733-2739.

Quan, L. and Liu, Miao. (2013), "CID, ETD and HCD Fragmentation to Study Protein Post-Translational Modifications", *Modern Chemistry and Applications*, vol. 1, no. 1, 1:e102. Doi:10.4172/mca.1000e102.

Ray, S., Koshy, N. R., Reddy, P. J. and Srivastava, S. (2012), "Virtual Labs in proteomics: new E-learning tools", *Journal of Proteomics*, vol. 75, no. 9, pp. 2515-2525.

Reiter, L., Rinner, O., Picotti, P., Huttenhain, R., Beck, M., Brusniak, M. Y., Hengartner, M. O. and Aebersold, R. (2011), "mProphet: automated data processing and statistical validation for large-scale SRM experiments", *Nature Methods*, vol. 8, no. 5, pp. 430-435.

Remily-Wood, E. R., et al. (2011), "A database of reaction monitoring mass spectrometry assays for elucidating therapeutic response in cancer", *Proteomics.Clinical Applications*, vol. 5, no. 7-8, pp. 383-396.

Rivers, J., Simpson, D. M., Robertson, D. H., Gaskell, S. J., Benyon, R. J. (2007). “Absolute multiplexed quantitative analysis of protein expression during muscle development using QconCAT”, *Molecular and Cellular Proteomics*, vol. 6, no. 8, pp. 1416-1427.

Rost, H., Malmstrom, L. and Aebersold, R. (2012), "A computational tool to detect and avoid redundancy in selected reaction monitoring", *Molecular & Cellular Proteomics*, vol. 11, no. 8, pp. 540-549.

Sauvage, F. L., Gaulier, J. M., Lachatre, G. and Marquet, P. (2008), "Pitfalls and prevention strategies for liquid chromatography-tandem mass spectrometry in the selected reaction-monitoring mode for drug analysis", *Clinical chemistry*, vol. 54, no. 9, pp. 1519-1527.

Sherman, J., McKay, M. J., Ashman, K. and Molloy, M. P. (2009), "Unique ion signature mass spectrometry, a deterministic method to assign peptide identity", *Molecular & Cellular Proteomics*, vol. 8, no. 9, pp. 2051-2062.

Shi, T., Su, D., Liu, T., Tang, K., Camp, D. G., Qian, W. J. and Smith, R. D. (2012), "Advancing the sensitivity of selected reaction monitoring-based targeted quantitative proteomics", *Proteomics*, vol. 12, no. 8, pp. 1074-1092.

Switzar, L., Giera, M. and Niessen, W. M. (2013), "Protein digestion: An overview of the available techniques and recent developments", *Journal of Proteome Research*, vol. 12, no. 3, 1067-1077.

Vizcaino, J. A., Cote, R., Reisinger, F., Barsnes, H., Foster, J. M., Rameseder, J., Hermjakob, H. and Martens, L. (2010), "The Proteomics Identifications database: 2010 update", *Nucleic Acids Research*, vol. 38, pp. D736-42.

Vizcaino, J. A., et al. (2013), "The PRoteomics IDEntifications (PRIDE) database and associated tools: status in 2013", *Nucleic Acids Research*, vol. 41, pp. D1063-9.

Wasinger, V. C., Zeng, M and Yunki, Y. (2013), “Current status and advances in quantitative proteomic mass spectrometry”, *International Journal of Proteomics*, vol. 2013, 180605.

Yan, Z., Maher, N., Torres, R., Cotto, C., Hastings, B., Dasgupta, M., Hyman, R., Huebert, N. and Caldwell, G. W. (2008), "Isobaric metabolite interferences and the requirement for close examination of raw data in addition to stringent chromatographic separations in liquid chromatography/tandem mass spectrometric analysis of drugs in biological matrix", *Rapid Communications in Mass Spectrometry*, vol. 22, no. 13, pp. 2021-2028.

Yates, J. R., Ruse, C. I. and Nakorchevsky, A. (2009), "Proteomics by mass spectrometry: approaches, advances, and applications", *Annual Review of Biomedical Engineering*, vol. 11, pp. 49-79.

Yocum, A. K. and Chinnaiyan, A. M. (2009), "Current affairs in quantitative targeted proteomics: multiple reaction monitoring-mass spectrometry", *Briefings in Functional Genomics & Proteomics*, vol. 8, no. 2, pp. 145-157.

Zhang, G., Ueberheide, B. M., Waldermarson, S., Myung, S., Molloy, K., Eriksson, J., Chait, B. T., Neubert, T. A. and Fenyo, D. (2010), “Protein quantitation using mass spectrometry”, *Methods in Molecular Biology*, vol. 673, pp. 211-222.

Zhu, W., Smith, J. W. and Huang, C. M. (2010), "Mass spectrometry-based label-free quantitative proteomics", *Journal of Biomedicine & Biotechnology*, 2010: 840518.

Zybailov, B., Coleman, M. K., Florens, L. and Washburn, M. P. (2005), "Correlation of relative abundance ratios derived from peptide ion chromatograms and spectrum counting for quantitative proteomic analysis using stable isotope labeling", *Analytical Chemistry*, vol. 77, no. 19, pp. 6218-6224.

APPENDICES

Appendix A Scripts

A.1 SQL Script for the schema for MRMinter

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL';

CREATE SCHEMA IF NOT EXISTS `MRMInter` DEFAULT CHARACTER SET latin1
COLLATE latin1_swedish_ci ;
USE `MRMInter`;

-- -----------------------------------------------------
-- Table `MRMInter`.`peptide`
-- -----------------------------------------------------
CREATE  TABLE IF NOT EXISTS `MRMInter`.`peptide` (
  `peptide_id` INT NOT NULL AUTO_INCREMENT ,
  `protein_ac` VARCHAR(9) NOT NULL ,
  `peptide_sequence` VARCHAR(45) NOT NULL ,
  `hydrophobicity` FLOAT NOT NULL ,
  PRIMARY KEY (`peptide_id`) )
ENGINE = InnoDB
COMMENT = 'This table contains peptide sequences';


-- -----------------------------------------------------
-- Table `MRMInter`.`precursor`
-- -----------------------------------------------------
CREATE  TABLE IF NOT EXISTS `MRMInter`.`precursor` (
  `precursor_id` INT NOT NULL AUTO_INCREMENT ,
  `precursor_m/z` FLOAT NOT NULL ,
  `precursor_charge` TINYINT NOT NULL ,
  `peptide_peptide_id` INT NOT NULL ,
  PRIMARY KEY (`precursor_id`, `peptide_peptide_id`) ,
  INDEX `fk_precursor_peptide` (`peptide_peptide_id` ASC) ,
  CONSTRAINT `fk_precursor_peptide`
    FOREIGN KEY (`peptide_peptide_id` )
    REFERENCES `MRMInter`.`peptide` (`peptide_id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
COMMENT = 'All values related to the precursor ion ';


-- -----------------------------------------------------
-- Table `MRMInter`.`product`
-- -----------------------------------------------------
CREATE  TABLE IF NOT EXISTS `MRMInter`.`product` (
  `product_id` INT NOT NULL AUTO_INCREMENT ,
  `product_m/z` FLOAT NOT NULL ,
  `product_type` VARCHAR(1) NOT NULL ,
  `product_number` TINYINT NOT NULL ,
  `peptide_peptide_id` INT NOT NULL ,
  PRIMARY KEY (`product_id`, `peptide_peptide_id`) ,
```

```
  INDEX `fk_product_peptide1` (`peptide_peptide_id` ASC) ,
  CONSTRAINT `fk_product_peptide1`
    FOREIGN KEY (`peptide_peptide_id` )
    REFERENCES `MRMInter`.`peptide` (`peptide_id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB
COMMENT = 'All values related to product ions.';


-- -----------------------------------------------------
-- Table `MRMInter`.`transition`
-- -----------------------------------------------------
CREATE  TABLE IF NOT EXISTS `MRMInter`.`transition` (
  `transition_id` INT NOT NULL AUTO_INCREMENT ,
  `peptide_sequence` VARCHAR(30) NOT NULL ,
  `hydrophobicity` FLOAT NOT NULL ,
  `precursor_m/z` FLOAT NOT NULL ,
  `precursor_charge` TINYINT NOT NULL ,
  `product_m/z` FLOAT NOT NULL ,
  `product_type` VARCHAR(1) NOT NULL ,
  `product_number` TINYINT NOT NULL ,
  PRIMARY KEY (`transition_id`) )
ENGINE = InnoDB
COMMENT = 'peptide transitions';


-- -----------------------------------------------------
-- Table `MRMInter`.`interference`
-- -----------------------------------------------------
CREATE  TABLE IF NOT EXISTS `MRMInter`.`interference` (
  `interference_id` INT NOT NULL AUTO_INCREMENT ,
  `peptide_sequence` VARCHAR(30) NOT NULL ,
  `hydrophobicity` FLOAT NOT NULL ,
  `precursor_m/z` FLOAT NOT NULL ,
  `precursor_charge` TINYINT NOT NULL ,
  `product_m/z` FLOAT NOT NULL ,
  `product_type` VARCHAR(1) NOT NULL ,
  `product_number` TINYINT NOT NULL ,
  `transition_transition_id` INT NOT NULL ,
  PRIMARY KEY (`interference_id`, `transition_transition_id`) ,
  INDEX `fk_interference_transition1` (`transition_transition_id` ASC)
,
  CONSTRAINT `fk_interference_transition1`
    FOREIGN KEY (`transition_transition_id` )
    REFERENCES `MRMInter`.`transition` (`transition_id` )
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;



SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

A.2 Perl Scripts

A.2.1 Populate_mrminter.pl

```
#! /usr/bin/perl -w



# Author:           Oshiobugie Dokpesi
# Date:             February 2013
# File:             populate_mrminter.pl

# Code Description: Program retrieves FASTA data from file and performs a
theoretical
#                   trypsin digest on the sequence. From the resulting
peptides only
#                   those from 3 to 30 amino acids long are selected. The
hydrophobicity,
#                   precursor charge, precursor m/z, product ion m/z,
production type and number
#                   values are then calculated and with the protein accession
number and peptide sequence
#                   inserted into the MRMinter database.
#



use strict;
use Data::Dumper;
use DBI;
use DBD::mysql;

my $hat;
my $i;
my $pep_hydro;
my $pepmass;
my $pep;
my $mz_1;
my $mz_2;
my $mz_3;
my $seq = '';
my $hot;

my %sequence = %{ read_fasta_as_hash('humans.fasta') };

foreach my $id ( keys %sequence ) {

       my ($tmp, $Access, $ID) = split(/\|/, $id);

       if ($ID =~ /(\d+\w_\w+)/){

              my $name = $1;

       }

# The protein sequence is digested using a regex that simulates the action
#of the enzyme trypsin
```

```
	my @seq = split/(?!P)(?<=[RK])/,  $sequence{$id});

#  peptides that are greater than 2 AAs and less than or equal to 24 AAs are
chosen.
	@seq = grep { length ($_) >= 3 && length ($_) <= 24 } @seq;
#The peptides are then sorted in a list from shortest to longest.

	foreach $seq(@seq){

		chomp ($seq);
		$hat = hydro($seq);

		if ($hat < 38) {
			$hot = $hat;
		}elsif ($hat >= 38) {
			$hot = $hat - 0.3*($hat - 38);
		}

		print " hydrophobicity is $hot \n";

		my $pepmaster = mz_ratio($seq);

		print "  The mass $seq is $pepmaster g   \n";

# mass-charge ratio is sum of residue masses + hydrogen divided by charge
# CSID:1010, http://www.chemspider.com/Chemical-Structure.1010.html (accessed
23:06, Oct 26, 2012)for mass of hydrogen atom (da)
		$mz_1 = ($pepmaster+1.007276)/1;
		$mz_2 = ($pepmaster+2.014552)/2;
		$mz_3 = ($pepmaster+3.021828)/3;

		print " charge1 :$mz_1 , charge2: $mz_2,  charge3: $mz_3 \n";

# convert the peptide sequence to mass of ions using subroutines.
# change mass of ions in array to hash

# change my b ions from array to hash value with the array position + 1 to
# represent the ion number. This number is then made into a key with the mass
as value.
		my @b_ions = pep_mass_bions($seq);

		my %hb_ions = ();
		for ($i=0; $i<scalar(@b_ions); $i++){
			$hb_ions{$i+1} = $b_ions[$i];
		}

# change my y ions from array to hash value with the array position + 1 to
# represent the ion number. This number is then made into a key with the mass
as value.
		my @y_ions = pep_mass_yions($seq);

		my %hy_ions = ();
		for ($i=0; $i<scalar(@y_ions); $i++){
			$hy_ions{$i +1} =  $y_ions[$i];
		}

# The hash for B ions
```

```
            print "\n This is the hash for my b ions: \n\n";

            while ( my ($key, $value) = each(%hb_ions) ) {
                        print "B|$key|$value\n";
            }

# The hash for Y ions
            print "\n This is the hash for my y ions: \n\n";


            while ( my ($key, $value) = each(%hy_ions) ) {
                  print "Y|$key|$value\n";

            }

            my $ds = "DBI:mysql:mrminterest:localhost";
            my $user = "root";
            my $passwd = "Peptide5";

            my $dbh = DBI->connect($ds,$user,$passwd) || die "Can't
Connect!";

# prepare an SQL statement


# insert into peptide table
            my $sth = $dbh->prepare("insert into peptide (protein_ac,
peptide_sequence, hydrophobicity) values (?,?,?)");

            $sth->execute( $Access, $seq, $hot);

# to get the primary key from the peptide table and insert it into the
foriegn keys of your child tables
            my $table_key = $dbh->{'mysql_insertid'};

# insert into precursor table
            my $sth2 = $dbh->prepare("insert into precursor (precursor_m/z,
precursor_charge, peptide_peptide_id) values (?,?,?)");

            $sth2->execute( $mz_1, 1, $table_key);
            $sth2->execute( $mz_2, 2, $table_key);
            $sth2->execute( $mz_3, 3, $table_key);

# insert into product table a while loop is used to loop over the hash and
retrieve values from it
            my $sth3 = $dbh->prepare("insert into product ( product_m/z,
product_type, product_number, peptide_peptide_id) values (?,?,?,?)");


            while ( my ($key, $value) = each(%hb_ions) ) {
                  $sth3->execute ($value, 'B', $key, $table_key );
            }
            while ( my ($key, $value) = each(%hy_ions) ) {
                  $sth3->execute ($value, 'Y', $key, $table_key );
            }
```

```
# finish in order to

             $sth->finish;
             $sth2->finish;
             $sth3->finish;
             $dbh->disconnect;



      }

}


sub read_fasta_as_hash {
    my $fn = shift;

    my $current_id = '';
    my %seqs;
    open FILE, "<$fn" or die $!;
    while ( my $line = <FILE> ) {
        chomp $line;
        if ( $line =~ /^(>.*)$/ ) {
            $current_id  = $1;
        } elsif ( $line !~ /^\s*$/ ) { # skip blank lines
            $seqs{$current_id} .= $line

        }
    }
    close FILE or die $!;

    return \%seqs;
}

# To calculate the hydrophobicity, the hydrophobicity model developed by
Krokhin et al is used. The values for the retention cooefficients are
substituted for the amino acids and calculated as follows.

sub hydro {

       my ($pep_seq) = @_;
       my $pep;
       my $pep_hydro = 0;
       my $Kl = 0;
       my $sumRc = 0;
       my $len = length ($pep_seq);


# To calculate the sum of all the amino acids
       for (my $i =0; $i< $len; $i += 1) {
             $pep = substr ($pep_seq, $i, 1);
             $sumRc += hyd($pep);

       }
# To calculate the hydrophobicity H = Kl*(sumRc + 0.42(R1cNt) + 0.22(R2cNt) +
0.05(R3cNt))
       my $R1 = .42 * Rcnt(substr($pep_seq, 0, 1));
```

```
	my $R2 = .22 * Rcnt(substr($pep_seq, 1, 1));

	my $R3 = .05 * Rcnt(substr($pep_seq, 2, 1));


# Taking into account the correction coefficient and reflecting the influence
of peptide length
	if ($len < 10) {
		$Kl = 1-0.027*(10-$len);
	}elsif ($len > 20) {
		$Kl = 1-0.014*($len-20);
	}else {
		$Kl = 1;
	}

	$pep_hydro = $Kl*($sumRc + $R1 + $R2 + $R3 );

	return  $pep_hydro;
}


# To obtain the mass charge ratio, the mass of the peptide is first
calculated
# Take a count of the respective amino acids and multiply the total count of
each individual AA by the mass of the amino acids.
# add all together plus the mass of water molecule to give the total mass for
the peptide.

sub m/z_ratio {
	my ($pep_seq) = @_;
	my $pep;
	my $pepmass;
	for (my $i =0; $i< length($pep_seq) ; $i += 1) {

		$pep = substr ($pep_seq, $i, 1);
		$pepmass += AAmass($pep);

	}
	$pepmass = $pepmass + 18.01056;
	return  $pepmass ;

}

# the b ions are the sum of the residue masses from the first AA (amino acid)
on the N->C direction in the peptide sequence to the last AA plus (1( the
mass of the hydrogen atom)).
# this can be achieved by taking the first AA add 1 and store as b1, then b2
= b1+(2nd AA), b(n) = bn-1+(last AA)
# subroutine to calculate the B product_ion  masses given a peptide sequence.
sub pep_mass_bions {

	my ($pep_seq) = @_;
	  my @AAs;
	  my $bmass;
	  my @bmass;
```

```
# use the split function to break up the string into an array
# loop through the array and calculate the ions.
      @AAs = split( "", $pep_seq);
      for ($i = 0; $i< scalar(@AAs); $i++){
            if ($i == 0) {
                  $bmass = AAmass($AAs[$i]) + 1.007276;
                        push (@bmass, $bmass);

            } else{
                  $bmass = $bmass + AAmass($AAs[$i]);
                      push (@bmass, $bmass);

            }
      }
      return @bmass ;
}

# the y ions are the sum of the residue masses from the first AA (amino acid)
on the C->N direction in the peptide sequence to the lasr AA plus (19 ( the
mass of water and hydrogen atom)).

# this can be achieved by reversing the peptide sequence first, then taking
the first AA add 19 and store as y1, the y2 = y1+(2nd AA, y(n) = y(n-1)+(last
AA).
# subroutine to calculate the Y product_ion masses given a peptide sequence.


sub pep_mass_yions {
# my input peptide sequence
      my ($pep_seq) = @_;
# my reversed peptide sequence to generate y_ions
        my $ypep_seq = reverse($pep_seq);
        my @AAs;
        my $ymass;
        my @ymass;

# use the split function to break up the string into an array
# loop through the array and calculate the ions.
      @AAs = split( "", $ypep_seq);
      for ($i = 0; $i< scalar(@AAs); $i++){
            if ($i == 0) {
                  $ymass = AAmass($AAs[$i]) + 19.0178407;
                  push (@ymass, $ymass);

            } else{
                  $ymass = $ymass + AAmass($AAs[$i]);
                      push (@ymass, $ymass);

            }
      }
      return @ymass ;
}


# subroutine to store the total monisotopic masses of the Amino acids. These
values can be used to calculate
```

```
# the mass of the peptides

sub AAmass {
    my ($AAcid) = @_;

    my (%AAmass) = (

   'A'        => 71.037114,  #alanine
   'C'        => 103.00919,  #cysteine
   'D'        => 115.02694,  #aspartic_acid
   'E'        => 129.04259,  #glutamic_acid
   'F'        => 147.06841,  #phenyalanine
   'H'        => 137.05891,  #histidine
   'I'        => 113.08406,  #isoleucine
   'K'        => 128.09496,  #lysine
   'L'        => 113.08406,  #leucine
   'M'        => 131.04049,  #methionine
   'N'        => 114.04293,  #asparagine
   'P'        => 97.052764,  #proline
   'Q'        => 128.05858,  #glutamine
   'R'        => 156.10111,  #arginine
   'S'        => 87.032029,  #serine
   'G'        => 57.021464,  #glycine
   'T'        => 101.04768,  #threonine
   'V'        => 99.068414,  #valine
   'W'        => 186.07931,  #tryptophan
   'Y'        => 163.06333,  #tyrosine
   'X'        => 0,          # unknown
   'B'        => 114.04293,  # Asparagine or aspartic acid
   'Z'        => 128.05858,  # Glutamine or glutamic acid
   'U'       => 150.95364,  # Selenocysteine
   'O'        => 237.14773,  # Pyrrolysine
   'J'        => 113.08406,  # leucine or isoleucine
 );

# clean up the input

$AAcid =~ s/\n//g;

   if (exists $AAmass{$AAcid}) {
        return $AAmass{$AAcid};
   }else{
           print STDERR "bad amino_acid \"$AAcid\"!!\n";
           exit;
   }
}


# subroutine to store the Retention coefficient values of the Amino acids.
These values are used to calculate
# the hydrophobicity values of the peptides

sub hyd {
    my ($AAcid) = @_;

    my (%hydro) = (
```

```
    'A'         => .80,   #alanine
    'C'         => -.80,  #cysteine
    'D'         => -.50,  #aspartic_acid
    'E'         => 0.0,   #glutamic_acid
    'F'         => 10.5,  #phenyalanine
    'H'         => -1.3,  #histidine
    'I'         => 8.4,   #isoleucine
    'K'         => -1.9,  #lysine
    'L'         => 9.6,   #leucine
    'M'         => 5.8,   #methionine
    'N'         => -1.2,  #asparagine
    'P'         => 0.2,   #proline
    'Q'         => -0.9,  #glutamine
    'R'         => -1.3,  #arginine
    'S'         => -0.80, #serine
    'G'         => -0.90, #glycine
    'T'         => 0.4,   #threonine
    'V'         => 5.0,   #valine
    'W'         => 11.0,  #tryptophan
    'Y'         => 4.0,   #tyrosine
    'X'         => 0,     # unknown
    'B'         => -1.2,  # Asparagine or aspartic acid
    'Z'         => -0.9,  # Glutamine or glutamic acid.
    'U'       => 0,     # Selenocysteine
    'O'         => 0,     # Pyrrolysine
    'J'         => 8.4,   # leucine or isoleucine
);

$AAcid =~ s/\n//g;
    if (exists $hydro{$AAcid}) {
         return $hydro{$AAcid};
    }else{
            print STDERR "bad amino_acid \"$AAcid\"!!\n";
            exit;
    }
}

# subroutine to store the weighted retention coefficients which reflect the
influence of the distance from the N-terminus
# of the Amino acids. These values are used to calculate the hydrophobicity
values of the peptides.

sub Rcnt {
    my ($AAcid) = @_;

    my (%hydro) = (

    'A'         => -1.5,  #alanine
    'C'         => 4.0,   #cysteine
    'D'         => 9.0,   #aspartic_acid
    'E'         => 7.0,   #glutamic_acid
    'F'         => -7.0,  #phenyalanine
    'H'         => 4.0,   #histidine
    'I'         => -8.0,  #isoleucine
    'K'         => 4.6,   #lysine
    'L'         => -9.0,  #leucine
    'M'         => -5.5,  #methionine
```

```
    'N'         => 5.0,   #asparagine
    'P'         => 4.0,   #proline
    'Q'         => 1.0,   #glutamine
    'R'         => 8.0,   #arginine
    'S'         => 5.0,   #serine
    'G'         => 5.0,   #glycine
    'T'         => 5.0,   #threonine
    'V'         => -5.5,  #valine
    'W'         => -4.0,  #tryptophan
    'Y'         => -3.0,  #tyrosine
    'X'         => 0,     # unknown
    'B'         => 5.0,   # Asparagine or aspartic acid
    'Z'         => 1.0,   # Glutamine or glutamic acid
    'U'       => 0,     # Selenocysteine
    'O'         => 0,      # Pyrrolysine
    'J'         => -8.0,   # leucine or isoleucine
);

$AAcid =~ s/\n//g;
    if (exists $hydro{$AAcid}) {
         return $hydro{$AAcid};
    }else{
            print STDERR "bad amino_acid \"$AAcid\"!\n";
            exit;
    }
}
```

A.2.2 Populate_mrminter_transitions.pl

```
#! /usr/bin/perl -w



# Author:           Oshiobugie Dokpesi
# Date:             February 2013
# File:             populate_mrminter_transitions.pl

# Code Description: Program retrieves transitions data from Mrmaid and uses
the data to populate the transitions table
#                   and find interferences for those transitions. those
interferences are then inserted into the interference table.
#
#
#
```

```
use strict;
use DBI;
use DBD::mysql;



my $peptide_sequencess;
my $RTs;
my $prec_m/z;
my $prec_charge;
my $prod_m/z;
my $product_typ;
my $product_numbe;
my $Peptide;

my $upper_tolerancepre;
my $lower_tolerancepre;

my $upper_tolerancepro;
my $lower_tolerancepro;

my $upper_toleranceRT;
my $lower_toleranceRT ;

# get the search terms to add to the select statement
print " please enter your precursor tolerance \n";
my $tolerance_precursor = <STDIN>;

print " please enter your product tolerance  \n";


my $tolerance_product =  <STDIN>;


print " please enter your hydrophobicity tolerance  \n";
my $tolerance_RT = <STDIN>;

print "Please wait while your data is being processed \n";

# Connect to MRMaid database
# Retrieve transition data from MRMaid database
my $dsn = "DBI:mysql:mrmaid:globe.ccc.cranfield.ac.uk";
my $user1 = "password";
my $passwdd = "password";

my $dbh = DBI->connect($dsn,$user1,$passwdd) || die "Can't Connect!";

# Prepare select statement.

 my $statemente = qq{select distinct Peptide.sequence, Precursor.charge,
Fragment.type, Fragment.serial from Peptide, PeptideFragment, Fragment,
```

```
Precursor, Transition where Peptide.species = 'Human' and Peptide.id =
PeptideFragment.peptideID and PeptideFragment.fragmentID = Fragment.id and
Fragment.id = Transition.fragmentID and Transition.precursorID =
Precursor.id;};

	my $sth1 = $dbh->prepare($statemente) || die "Cannot prepare
statement:" . DBI->errstr;

	$sth1->execute() || die "Cannot execute statement:" . DBI->errstr;


	# retrieve required values from database
	while (my @response = $sth1->fetchrow_array()){

		my $peptide_sequences = $response[0];
		my $prec_charge= $response[1];
		my $f_type = $response[2];
		my $f_serial = $response[3];

print "$peptide_sequences\t$prec_charge\t$f_type\t$f_serial\n";


# connect to MRMinter database
		   my $ds = "DBI:mysql:mrminter:localhost";
		my $user = "root";
		my $passwd = "password";
		my $dbh = DBI->connect($ds,$user,$passwd) || die "Can't
Connect!";

# Prepare select statement to retrieve hydrophobicity, precursor charge
precursor m/z and product m/z values from MRMinter using
# transition information from MRMaid
my $statement = qq{select distinct peptide.peptide_sequence,
peptide.hydrophobicity, precursor.precursor_m/z, precursor.precursor_charge,
product.product_m/z, product.product_type, product.product_number from
	peptide join precursor on peptide.peptide_id =
precursor.peptide_peptide_id join product on peptide.peptide_id =
product.peptide_peptide_id where peptide_sequence = '$peptide_sequences' and
precursor_charge = '$prec_charge' and product_type = '$f_type' and
product_number = '$f_serial';};

		my $sth2 = $dbh->prepare($statement) || die "Cannot prepare
statement:" . DBI->errstr;

		$sth2->execute() || die "Cannot execute statement:" . DBI-
>errstr;


# retrieve required values from database
		while (my @response = $sth2->fetchrow_array()){

			$peptide_sequencess = $response[0];
			$RTs = $response[1];
			$prec_m/z = $response[2];
			$prec_charge = $response[3];
			$prod_m/z = $response[4];
			$product_typ = $response[5];
```

```
                    $product_numbe = $response[6];

# Insert theoretical transition values matching MRMaid transtion values inot
transitions table in MRMinter
      my $sth3 = $dbh->prepare("insert into transitions(peptide_sequence,
hydrophobicity, precursor_m/z, precursor_charge, product_m/z, product_type,
product_number) values(?,?,?,?,?,?,?)");
      $sth3->execute( $peptide_sequencess, $RTs, $prec_m/z, $prec_charge,
$prod_m/z, $product_typ, $product_numbe);

# to get the primary key from the transitions table and insert it into the
foriegn keys of your child tables

                    my $table_key = $dbh->{'mysql_insertid'};

# using the tolerance values inputted by to produce upper range and lower
range of values a required by select statement
# for this project tolerance values used are +-3Da for precursor ion, +-1.1Da
for product ion and 3 units for hydrophobicity
# to cover the resolutions used by scientists when inputting spectra into
PRIDE.
            my $upper_tolerancepre = $prec_m/z + $tolerance_precursor;
            my $lower_tolerancepre = $prec_m/z - $tolerance_precursor;

            my $upper_tolerancepro = $prod_m/z + $tolerance_product;
            my $lower_tolerancepro = $prod_m/z - $tolerance_product;

            my $upper_toleranceRT = $RTs + $tolerance_RT;
            my $lower_toleranceRT =  $RTs - $tolerance_RT;

# my select statement to select the interference values
my $statements = qq{select distinct peptide.peptide_sequence,
peptide.hydrophobicity, precursor.precursor_m/z, precursor.precursor_charge,
product.product_m/z, product.product_type, product.product_number from
peptide join precursor on peptide.peptide_id = precursor.peptide_peptide_id
join product on peptide.peptide_id = product.peptide_peptide_id where
hydrophobicity >= '$lower_toleranceRT' and hydrophobicity <=
'$upper_toleranceRT' and precursor_m/z >= '$lower_tolerancepre' and
precursor_m/z <= '$upper_tolerancepre' and product_m/z >=
'$lower_tolerancepro' and product_m/z <= '$upper_tolerancepro'  and
peptide_sequence != '$peptide_sequencess' and product_m/z != '$prod_m/z';};

                    my $sth4 = $dbh->prepare($statements) || die "Cannot
prepare statement:" . DBI->errstr;

                    $sth4->execute() || die "Cannot execute statement:" .
DBI->errstr;


# retrieve require values from database
                    while (my @response = $sth4->fetchrow_array()){

      #my $protein_ac = $response[0];
                          my $peptide_sequence = $response[0];
                          my $RT = $response[1];
                          my $precursor_m/z = $response[2];
                          my $precursor_charge = $response[3];
```

```
                        my $product_m/z = $response[4];
                        my $product_type = $response[5];
                        my $product_number = $response[6];


# insert interference values into interference tables these values represent
the transitions that interfere with MRMaid transitions
#
      my $sth5 = $dbh->prepare("insert into interference (peptide_sequence,
hydrophobicity, precursor_m/z, precursor_charge, product_m/z, product_type,
product_number, transitions_transitions_id) values(?,?,?,?,?,?,?,?)");
      $sth5->execute( $peptide_sequence, $RT, $precursor_m/z,
$precursor_charge, $product_m/z, $product_type, $product_number, $table_key);

      }




}
}

exit;
```

A.2.3 Retrieve_interference.pl

```
#! /usr/bin/perl -w
use strict;

use DBI;
use DBD::mysql;


my @datax;
my @datay;
my $peptide_sequencess;
my $RTs;
my $prec_m/z;
my $prec_charge;
my $prod_m/z;
my $product_typ;
my $product_numbe;
my $Peptide;

my $temp_q1 = 1;
my $temp_q3 = 1;
my $upper_tolerancepre;
my $lower_tolerancepre;

my $upper_tolerancepro;
my $lower_tolerancepro;

my $upper_toleranceRT;
my $lower_toleranceRT ;
```

```
my $outputfile = 'transitions_MRMInter_1_.7_2.csv';
open (OUTFILE, ">$outputfile");



# get the search terms to add to the select statement
print " please enter your precursor tolerance \n";
my $tolerance_precursor = <STDIN>;

print " please enter your product tolerance  \n";
#my $tolerance_product = <STDIN>;

my $tolerance_product =  <STDIN>;


print " please enter your hydrophobicity tolerance  \n";
my $tolerance_RT = <STDIN>;

print "Please wait while your data is being processed \n";




               my $ds = "DBI:mysql:MRMInter:localhost";
             my $user = "root";
             my $passwd = "Peptide5";
             my $dbh = DBI->connect($ds,$user,$passwd) || die "Can't
Connect!";


my $statement = qq{select distinct transition.peptide_sequence,
transition.hydrophobicity, transition.precursor_m/z,
transition.precursor_charge, transition.product_m/z, transition.product_type,
transition.product_number from transition;};

      my $sth = $dbh->prepare($statement) || die "Cannot prepare statement:"
. DBI->errstr;

      $sth->execute() || die "Cannot execute statement:" . DBI->errstr;


      # retrieve required values from database
      while (my $response = $sth->fetchrow_arrayref()){

            $peptide_sequencess = $response->[0];
            $RTs = $response->[1];
            $prec_m/z = $response->[2];
            $prec_charge = $response->[3];
            $prod_m/z = $response->[4];
            $product_typ = $response->[5];
            $product_numbe = $response->[6];
```

```
            #print OUTFILE "Theoretical transition:
$peptide_sequencess\t$RTs\t$prec_m/z\t$prec_charge\t$prod_m/z\t$product_typ\t
$product_numbe \n";


            my $upper_tolerancepre = $prec_m/z + $tolerance_precursor;
            my $lower_tolerancepre = $prec_m/z - $tolerance_precursor;

            my $upper_tolerancepro = $prod_m/z + $tolerance_product;
            my $lower_tolerancepro = $prod_m/z - $tolerance_product;

            my $upper_toleranceRT = $RTs + $tolerance_RT;
            my $lower_toleranceRT =  $RTs - $tolerance_RT;

            # my select statement to select the interference values
my $statements = qq{ select interference.peptide_sequence,
interference.product_type, interference.product_number,
count(interference.interference_id) as NumberOfInterferences from transition,
interference where transition.peptide_sequence = '$peptide_sequencess' and
transition.product_type = '$product_typ' and transition.product_number =
'$product_numbe' and interference.hydrophobicity >= '$lower_toleranceRT' and
interference.hydrophobicity <= '$upper_toleranceRT' and
interference.precursor_m/z >= '$lower_tolerancepre' and
interference.precursor_m/z <=    '$upper_tolerancepre' and
interference.product_m/z >= '$lower_tolerancepro' and
interference.product_m/z <= '$upper_tolerancepro' and
transition_transition_id = transition_id ;};

my $sth2 = $dbh->prepare($statements) || die "Cannot prepare statement:" .
DBI->errstr;

$sth2->execute() || die "Cannot execute statement:" . DBI->errstr;

#print OUTFILE "Pep-sequence\thydrophobicity\tPrec-m/z\tPrec-charge\tProd-
m/z\tProd-type\tProd-number \n";

# retrieve require values from database
while (my $response = $sth2->fetchrow_arrayref()){

      #my $protein_ac = $response[0];
      my $peptide_sequence = $response->[0];
      my $RT = $response->[1];
      my $precursor_m/z = $response->[2];
      my $count = $response->[3];
      my $product_m/z = $response->[4];
      my $product_type = $response->[5];
      my $product_number = $response->[6];


      print OUTFILE " $peptide_sequencess,$product_typ,$product_numbe,$count
\n";

      print " $peptide_sequencess\t$product_typ\t$product_numbe\t$count \n";
```

```
}
}
close (OUTFILE);
#close (FILE);
exit;
```

A.2.4 Retrieve_interference2.pl

```
#! /usr/bin/perl -w


# Author:           Oshiobugie Dokpesi
# Date:             February 2013
# File:             retrieve_interference2.pl

# Code Description: Program retrieves the interferences for transitions from
the interference table in MRMinter



use strict;
use DBI;
use DBD::mysql;


my @datax;
my @datay;
my $peptide_sequencess;
my $RTs;
my $prec_m/z;
my $prec_charge;
my $prod_m/z;
my $product_typ;
my $product_numbe;
my $Peptide;

my $upper_tolerancepre;
my $lower_tolerancepre;

my $upper_tolerancepro;
my $lower_tolerancepro;

my $upper_toleranceRT;
my $lower_toleranceRT ;

my $tolerance_precursor = .0001;
my $tolerance_product = .0001;
my $tolerance_RT = .0001;



# Get the search terms to add to the select statement
# as the select query will not allow a tolerance of 0 due to a neccesity for
a range
```

```
# in order to search default setting for zero is 0.0001.
print " please enter your precursor tolerance \n";
my $temp_tolerance_prec = <STDIN>;
if($temp_tolerance_prec > .0001) {

    $tolerance_precursor = $temp_tolerance_prec;
}



print " please enter your product tolerance  \n";
my $temp_tolerance_pro =  <STDIN>;
if($temp_tolerance_pro > .0001) {

    $tolerance_product = $temp_tolerance_pro;
}


print " please enter your hydrophobicity tolerance  \n";
my $temp_tolerance_RT = <STDIN>;
if($temp_tolerance_RT > .0001) {

    $tolerance_RT = $temp_tolerance_RT;
}


print "Please wait while your data is being processed \n";

# output file to store the interference data can aslo be in .csv format
my $outputfile = 'transitions_mrmaid.txt';
open (OUTFILE, ">$outputfile");




my $ds = "DBI:mysql:MRMInter:localhost";
my $user = "root";
my $passwd = "Peptide5";
my $dbh = DBI->connect($ds,$user,$passwd) || die "Can't Connect!";

# select statement to search through all the transitions in the database.
this can be modified to search for only a subset of transtions.
my $statement = qq{select distinct transition.peptide_sequence,
transition.hydrophobicity, transition.precursor_m/z,
transition.precursor_charge, transition.product_m/z, transition.product_type,
transition.product_number from transition where transition.peptide_sequence =
'EIGELYLPK' and transition.precursor_m/z >= '531.2' and
transition.precursor_m/z <= '5531.30' and transition.product_m/z >= '633.3'
and transition.product_m/z <= '633.40';};

      my $sth2 = $dbh->prepare($statement) || die "Cannot prepare
statement:" . DBI->errstr;

      $sth2->execute() || die "Cannot execute statement:" . DBI->errstr;
```

```
        # retrieve required values from database
        while (my @response = $sth2->fetchrow_array()){

                $peptide_sequencess = $response[0];
                $RTs = $response[1];
                $prec_m/z = $response[2];
                $prec_charge = $response[3];
                $prod_m/z = $response[4];
                $product_typ = $response[5];
                $product_numbe = $response[6];
                print OUTFILE " Interference for peptide :
$peptide_sequencess\t$RTs\t$prec_mz\t$prec_charge\t$prod_mz\t$product_typ\t$p
roduct_numbe \n";


# using the tolerance values inputted by to produce upper range and lower
range of values a required by select statement

                $upper_tolerancepre = $prec_m/z + $tolerance_precursor;
                $lower_tolerancepre = $prec_m/z - $tolerance_precursor;

                $upper_tolerancepro = $prod_m/z + $tolerance_product;
                $lower_tolerancepro = $prod_m/z - $tolerance_product;

                  $upper_toleranceRT = $RTs + $tolerance_RT;
                $lower_toleranceRT =  $RTs - $tolerance_RT;

# my select statement to select the interference values
# my select statement to select the interference values
my $statements = qq{ select distinct transition.peptide_sequence,
interference.peptide_sequence, interference.hydrophobicity,
interference.precursor_m/z, interference.precursor_charge,
interference.product_m/z, interference.product_type,
interference.product_number from transition join interference where
transition.peptide_sequence = '$peptide_sequencess' and
transition.product_type = '$product_typ' and transition.product_number =
'$product_numbe' and interference.hydrophobicity >= '$lower_toleranceRT' and
interference.hydrophobicity <= '$upper_toleranceRT' and
interference.precursor_m/z >= '$lower_tolerancepre' and
interference.precursor_m/z <= '$upper_tolerancepre' and
interference.product_m/z >= '$lower_tolerancepro' and
interference.product_m/z <= '$upper_tolerancepro' and
transition_transition_id = transition_id ;};


                my $sth2 = $dbh->prepare($statements) || die "Cannot prepare
statement:" . DBI->errstr;

$sth2->execute() || die "Cannot execute statement:" . DBI->errstr;

#print OUTFILE "Pep-sequence\thydrophobicity\tPrec-m/z\tPrec-charge\tProd-
m/z\tProd-type\tProd-number \n";

# retrieve require values from database
                while (my @response = $sth2->fetchrow_array()){
```

```
                    my $peptide_sequencess = $response[0];
                    my $peptide_sequences = $response[1];
                    my $RT = $response[2];
                    my $precursor_m/z = $response[3];
                    my $precursor_charge = $response[4];
                    my $product_m/z = $response[5];
                    my $product_type = $response[6];
                    my $product_number = $response[7];


      print OUTFILE "
$peptide_sequences\t$RT\t$precursor_m/z\t$precursor_charge\t$product_m/z\t$pr
oduct_type\t$product_number\n";


}
}
close (OUTFILE);
#close (FILE);
exit;
```