Optimal precursor ion selection for LC-MS/MS based proteomics

von

Alexandra Zerck

Mai 2013

Dissertation zur Erlangung des Grades

Doktor der Naturwissenschaften (Dr. rer. nat.)

eingereicht am Fachbereich Mathematik und Informatik,

Freie Universität Berlin

Datum der Disputation:

29.11.2013

Betreuer:

Professor Dr. Knut Reinert
Freie Universität Berlin
Institut für Informatik
Algorithmische Bioinformatik
Takustraße 9
D-14195 Berlin

Gutachter:

Professor Dr. Knut Reinert, Freie Universität Berlin
Professor Dr. Oliver Kohlbacher, Eberhard Karls Universität Tübingen

Abstract

Shotgun proteomics with Liquid Chromatography (LC) coupled to Tandem Mass Spectrometry (MS/MS) is a key technology for protein identification and quantitation. Protein identification is done indirectly: detected peptide signals are fragmented by MS/MS and their sequence is reconstructed. Afterwards, the identified peptides are used to infer the proteins present in a sample. The problem of choosing the peptide signals that shall be identified with MS/MS is called *precursor ion selection*. Most workflows use data-dependent acquisition for precursor ion selection despite known drawbacks like data redundancy, limited reproducibility or a bias towards high-abundance proteins. In this thesis, we formulate optimization problems for different aspects of precursor ion selection to overcome these weaknesses.

In the first part of this work we develop inclusion lists aiming at optimal precursor ion selection given different input information. We trace precursor ion selection back to known combinatorial problems and develop linear program (LP) formulations. The first method creates an inclusion list given a set of detected features in an LC-MS map. We show that this setting is an instance of the Knapsack Problem. The corresponding LP can be solved efficiently and yields inclusion lists that schedule more precursors than standard methods when the number of precursors per fraction is limited. Furthermore, we develop a method for inclusion list creation based on a list of proteins of interest. We employ retention time and detectability prediction to infer LC-MS features. Based on peptide detectability, we introduce protein detectabilities that reflect the likelihood of detecting and identifying a protein. By maximizing the sum of protein detectabilities we create an inclusion list of limited size that covers a maximum number of proteins.

In the second part of the thesis, we focus on iterative precursor ion selection (IPS) with LC-MALDI MS/MS. Here, after a fixed number of acquired MS/MS spectra their identification results are evaluated and are used for the next round of precursor ion selection. We develop a heuristic which creates a ranked precursor list. The second method, IPS_LP, is a combination of the two inclusion list scenarios presented in the first part. Additionally, a protein-based exclusion is part of the objective function. For evaluation, we compared both IPS methods to a static inclusion list (SPS) created before the beginning of MS/MS acquisition. We simulated precursor ion selection on three data sets of different complexity and show that IPS_LP can identify the same number of proteins with fewer selected precursors. This improvement is especially pronounced for low abundance proteins. Additionally, we show that IPS_LP decreases the bias to high abundance proteins.

All presented algorithms were implemented in OpenMS, a software library for mass spectrometry. Finally, we present an online tool for IPS that has direct access to the instrument and controls the measurement.

Zusammenfassung

Flüssigkeitschromatographie (LC) gekoppelt mit Tandemmassenspektrometrie (MS/MS) ist eine Schlüsseltechnologie für die Proteinidentifikation und Quantifizierung in proteomischen Proben. Dabei werden Proteine indirekt identifiziert: detektierte Peptidsignale werden durch MS/MS fragmentiert und anschließend wird die Peptidsequenz rekonstruiert. Über die identifizierten Peptide werden schließlich die Proteine in der Probe identifiziert. Das Problem der Auswahl der Peptidsignale, die über MS/MS sequenziert werden sollen, heißt *Precursor-Ionen-Selektion* (PS). Die meisten Selektionsverfahren benutzen rein intensitätsbasierte Ansätze – sogenannte Datenabhängige Akquisition (DDA) – trotz bekannter Schwächen wie Datenredundanz, begrenzter Reproduzierbarkeit oder einer Neigung zur Identifikation häufiger Proteine. In dieser Arbeit entwickeln wir für unterschiedliche Aspekte der PS Formulierungen als Optimierungsprobleme mit dem Ziel den bekannten Schwächen entgegenzusteuern.

Im ersten Teil der Arbeit werden für unterschiedliche Anfangsinformationen optimale Inklusionslisten erstellt. Dabei führen wir PS auf bekannte kombinatorische Probleme zurück und entwickeln Formulierungen als Lineare Programme (LP) zur Lösung der Probleme. Die erste Methode basiert auf einer Liste von LC-MS-Features. Wir zeigen, dass sich diese Situation auf das Rucksackproblem zurückführen läßt. Das zugehörige LP erstellt effiziente Inklusionslisten, die mehr Precursor enthalten als Standardmethoden, wenn die Anzahl an Precursor-Ionen pro Fraktion begrenzt ist. Außerdem entwickeln wir eine Methode basierend auf einer Liste an zu identifizierenden Proteinsequenzen. Wir benutzen Schätzverfahren für RT und Detektierbarkeit um repräsentative LC-MS-Features für diese Proteine vorherzusagen. Basierend auf der Peptiddetektierbarkeit führen wir eine Proteindetektierbarkeit ein. Indem wir die Summe dieser maximieren, erstellen wir eine größenbeschränkte Inklusionsliste, die eine maximale Anzahl an Proteinen abdeckt.

Im zweiten Teil der Arbeit beschäftigen wir uns mit iterativer PS (IPS) mit LC-MALDI MS/MS. Dabei werden nach einer bestimmten Anzahl an aufgenommenen MS/MS-Spektren deren Identifikationsergebnisse ausgewertet und diese zur weiteren PS benutzt. Wir entwickeln einerseits eine Heuristik, die eine priorisierte Inklusionsliste erstellt. Für die zweite Methode, IPS_LP, kombinieren wir die beiden LP-Formulierungen aus dem ersten Teil und erweitern sie um eine proteinbasierte Exklusion. Für die Auswertung vergleichen wir unsere IPS-Methoden mit einer statischen Inklusionsliste (SPS), die vor Beginn der MS/MS-Messung erstellt wurde. Wir simulieren die PS auf drei Datensätzen mit unterschiedlicher Komplexität und zeigen, dass IPS_LP die gleiche Proteinanzahl wie SPS identifiziert, dabei aber weniger MS/MS-Messungen benötigt. Diese Verbesserung wird insbesondere für Proteine mit geringer Abundanz deutlich. Außerdem können wir zeigen, dass die Neigung zur Identifikation häufiger Proteine gesenkt wird.

Unsere Algorithmen wurden als Teil von OpenMS, einer Softwarebibliothek für Massenspektrometrie, implementiert. Im letzten Teil stellen wir außerdem ein Onlinetool vor, dass direkten Zugriff auf das Massenspektrometer hat und die Messungen steuert.

Acknowledgments

The work for this thesis was carried out at the Max Planck Institut für Molekulare Genetik in Berlin and at the Freie Universität Berlin. I want to thank all people who helped and supported me during the last years.

First, I would like to thank my advisors: I thank Johan Gobom for his support especially during the first years and for bringing my attention to this topic. I am grateful for the opportunity to gain insight into proteomics from the experimental perspective. I thank Johan and Hans Lehrach for the opportunity to work at MPI. I want to thank Eckhard Nordhoff for filling in as supervisor and for his guidance especially during the writing of my first paper. I thank Knut Reinert for his support in the last years, shelter at the FU and for showing me the mathematical view of precursor ion selection. Furthermore, I thank Oliver Kohlbacher for reviewing this thesis.

A special thanks goes to my colleagues at MPI and FU. Thanks to the last of the Mohicans of the MS group: Klaus-Dieter Klöppel and Beata Lukaszewska-McGreal for their support, joyful lunch sessions and coffee breaks. I am indebted to KDK for his guidance during all the years at the MPI. I thank Beata, Gabriela Thiele and Christine Lübbert for all the work in the lab. Furthermore, I want to thank the whole Algorithmic Bioinformatics group for the nice working atmosphere. Thanks to everyone from the OpenMS team for all the effort they put into the development of OpenMS and for enjoyable retreats. Especially, I am grateful to Nico Pfeifer for his support with RT and detectability prediction, and for providing evaluation scripts that were used in this thesis.

I want to thank all my friends for great distraction especially throughout the tougher times of the last years. I am indebted to Samira Jaeger and Lars Petzold who slogged through earlier versions of this thesis and helped improving it a lot by their comments. Furthermore, I thank Mara Pankau for proofreading parts of the thesis.

Ein besonderer Dank gilt meinen Eltern, die mich immer unterstützt haben. Ihnen und meinen Schwiegereltern möchte ich für die zahlreichen Babysitterdienste danken, die es mir ermöglicht haben diese Arbeit tatsächlich fertigzustellen. Schließlich möchte ich meiner eigenen wundervollen Familie danken: Lars, der mit viel Geduld und Verständnis immer für mich da war, und unserer Tochter Charlotte. Gemeinsam zeigt ihr mir jeden Tag aufs Neue, was im Leben wirklich wichtig ist.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The publication of a first version of the human genome by the Human Genome Project [1] was an important milestone at the beginning of this century. Along with the genome sequence it became obvious that former estimations concerning the number of protein-coding genes had to be adjusted downwards from 30,000 - 40,000 genes estimated with the draft version of the genome [2, 3] to around 20,000 - 25,000 [1][1]. The number of proteins these genes are translated into is several orders of magnitude larger due to post-transcriptional and post-translational events such as alternative splicing and various post-translational modifications. This high number reflects the key role that proteins play in virtual every important biological process: they catalyze biochemical reactions, act as structural components of cells, and participate, amongst others, in cell signaling and immune responses.

In analogy to the notion of genome, the term proteome was proposed. The proteome is defined as "the **PROTE**in complement expressed by a gen**OME**" [5]. As such, it comprises the set of proteins expressed in a given biological system at a specific time point. In contrast to the genome, which is essentially the same in all cells and does not change during the life span of an organism, the proteome is highly dynamic. The expressed proteins vary between cell types, different environmental conditions, cell cycle states etc.

In the following, we give a short introduction into protein structure and the analysis of samples by Liquid Chromatography Tandem Mass Spectrometry (LC-MS/MS). This provides the background needed for the motivation of the thesis.

1.1. Proteomics

1.1.1. Protein structure

Proteins are chains of amino acids whose sequence is coded in the genome. Protein synthesis is a two-step process. First, DNA is translated into messenger RNA (mRNA) which is, after some processing, translated into the protein sequence.

[1] The current Gencode version 14 lists 20,078 protein-coding genes [4].

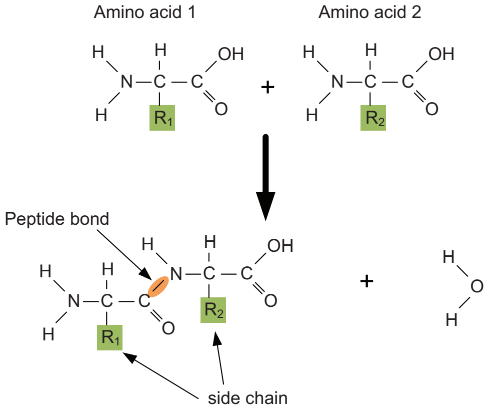


Figure 1.1.: Peptide bond. Two amino acids are covalently bonded in a dehydration reaction that includes a loss of water.

There are 20 amino acids that occur in proteins and are encoded in the genome. All of them consist of three functional groups: the carboxyl group $COOH$, the amino group NH_2 and the side chain R. The side chain is specific for each amino acid and determines its physico-chemical properties like charge, hydrophobicity, and size to name only a few. In a peptide, the amino acid chain is built by peptide bonds which link the amino group of one amino acid to the carboxyl group of another amino acid (Figure 1.1). The peptide end with a free carboxyl group is called *C terminus*, the amino end is denoted as *N terminus*. A linear chain of amino acids is called a polypeptide. A protein consists of one or more polypeptides whose C, N and O atoms linked in the peptide bonds form the protein backbone. The combination of all amino acid side chains defines the three-dimensional structure of a protein and its functional properties.

1.1.2. Proteomic workflows

The term proteomics describes the analysis of proteins and whole proteomes, their expression profiles, functions, structures, and interactions. Initially, protein analysis focused on the study of single proteins. However, proteomics is a strongly technology-driven research area, where developments, especially in the field of biological mass spectrometry and separation technology, have enabled detection of several thousand proteins in small quantities of biological samples.

In early proteomic workflows, much effort was invested into separating the sample proteins prior to MS analysis. Particularly, high-resolving two-dimensional gel electrophoresis became the core technique for protein separation. As mass spectrometers evolved, especially with regards to sensitivity and speed for peptide sequencing with tandem mass spectrometry (MS/MS), it became possible to efficiently identify proteins based on fragment ion analysis of individual proteolytic peptides in mixtures, thus alleviating the need for protein fractionation

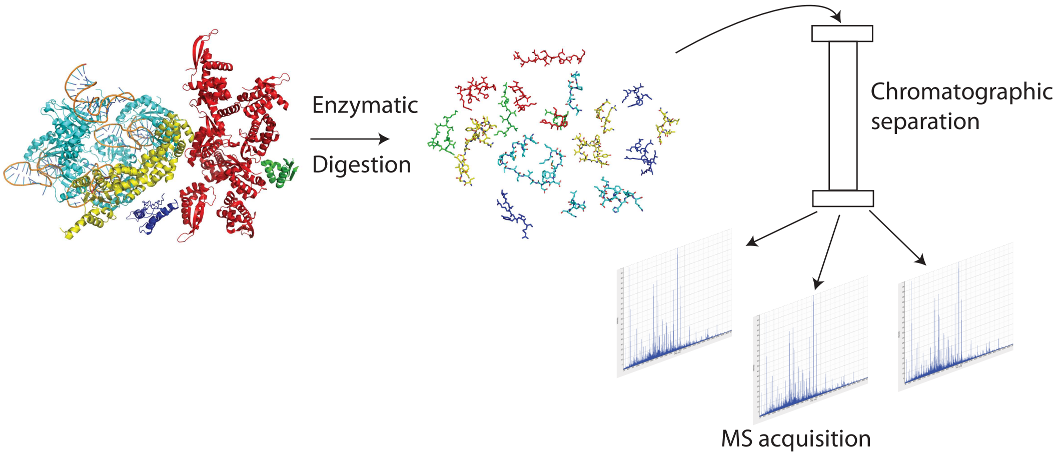
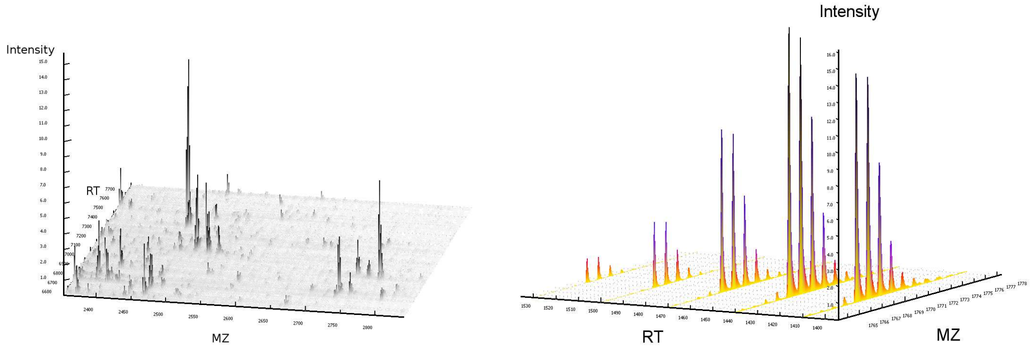


Figure 1.2.: Workflow for shotgun proteomics. The protein mixture to be analyzed is first enzymatically digested, usually with trypsin. The resulting peptide mixture is then fractionated via liquid chromatography. After chromatographic separation the peptides are ionized and separated by their mass-to-charge (m/z) ratio in the mass spectrometer. The m/z-ratios of the ions are recorded, resulting in mass spectra where the signal intensity reflects the amount of ions detected at each m/z-value.

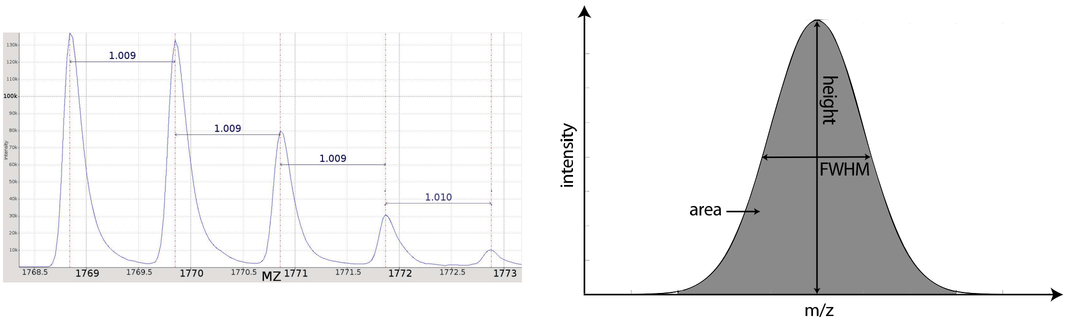
prior to proteolytic digestion and MS analysis. Instead, crude protein mixtures are subjected to enzymatic digestion, and the produced complex peptide mixtures are separated by liquid-chromatography (LC) coupled to MS. This analytical approach has been termed “shotgun proteomics” in analogy to shotgun genomics and is now a standard approach to gain information about the identity and quantity of the proteins in a specific sample.

In a typical LC-MS setup (illustrated in Figure 1.2), peptides in a sample are first separated by liquid chromatography based on their physico-chemical properties like hydrophobicity. The LC system is coupled to a mass spectrometer, either directly or indirectly via fractionation onto a target plate which is inserted into the mass spectrometer. Inside, the peptides are ionized and their mass-to-charge-ratios (m/z) are determined. The signal intensity at a specific m/z-value depends on the amount of ions present with this m/z which makes it possible to measure the peptide quantity present in the sample. In order to obtain structural information for a peptide ion, it is fragmented in the mass spectrometer and the m/z-values of the fragment ions are recorded in a mass spectrometer. This process is called tandem mass spectrometry (MS/MS). By means of these fragment ions it is possible to partially derive the peptide sequence. Afterwards, the peptides are mapped onto proteins. Thus, the proteins present in the original sample can be reconstructed. In this thesis we focus on how to decide which peptide ion signals shall be sequenced. This decision is called *precursor ion selection*. We demonstrate why it is an important issue in proteomics after a short excursus into the nature of MS data.



(a) (b)

Figure 1.3.: LC-MS map. (a) A zoom into a LC-MS map. (b) LC-MS map of a peptide feature.



(a) (b)

Figure 1.4.: Peak characteristics. (a) Isotope pattern. (b) Peak parameters.

The nature of MS data

LC-MS analysis of a sample results in a three-dimensional map, see Figure 1.3 (a) for a zoom into one. Each data point is characterized by three values: the RT at which the MS spectrum was recorded, the measured mass-to-charge ratio m/z and the number of detected ions, i.e., the signal intensity.

Peptide signals usually occur in several consecutive scans, building an RT elution profile. In m/z dimension, the peptide signal consists of several isotopic peaks, their distance depending on the peptide ion's charge z, Figure 1.4 (a) shows a charge one isotope pattern. Each peak can be described by certain characteristics like its m/z-value, its maximal height, its area under the curve (which usually builds the peak intensity after signal processing) or its full-width-at-half-max (FWHM). These parameters are displayed in Figure 1.4 (b). All MS peaks belonging to the same peptide signal at a distinct charge form a so-called LC-MS feature, see Figure 1.3 (b) for a zoom into an LC-MS map showing a feature.

1.2. Motivation

For peptide sequencing via MS/MS the peptide ions of interest are isolated and further fragmented. The isolation window is mainly as small as possible to isolate all ions in question so that the interference with other ions with a similar m/z-value is minimized. The standard workflow uses an MS spectrum, a so-called survey scan, to determine the m/z-values of all compounds present in the analyzed fraction. Usually, many more signals are detected in the survey scan than can be selected for MS/MS. Even low complexity samples, like a standard containing 20 proteins, produce more peptide ions than can be fragmented in a single run [6, 7]. The two different ionization techniques Electrospray Ionization (ESI) and Matrix-assisted Laser Desorption/Ionization (MALDI) pose different constraints on the sample usage: The main limitation with ESI-MS/MS is time, as the sample is analyzed on the fly during elution from the column. In contrast, MALDI-MS/MS, being performed off-line, is limited by the sample available for each fraction. In a standard workflow the highest signals in the spectrum are selected for fragmentation via MS/MS. This procedure is one possible implementation of the so-called data-dependent acquisition (DDA). Some companies use "Data directed analysis" or "information dependent acquisition" to denote the same procedure as Thermo Fisher trademarked the term "Data-Dependent" [8]. With DDA, first a survey MS spectrum is acquired and processed. Then, based on some predefined rules such as a specific charge state or a minimal signal intensity ions for MS/MS are selected [8].

One of the main problems when using DDA in LC-MS/MS is the limited reproducibility of replicates. Small variations in the signal intensities of peptide ions might result in different sets of selected precursors and thus lead to different peptide and protein identifications. For instance, a systematic study from Tabb et al. [6] showed an overlap in peptide identifications of only 35-60% in technical replicates. The reproducibility at protein level was similar. Liu et al. [9] analyzed 9 LC-LC-MS/MS samples recorded with the same settings. In the cumulative protein set only 35% of the proteins were found in all runs, while 24% were identified in only one run.

Another important problem is the high dynamic range of protein abundances in biological samples. For instance, the plasma proteome spans 12 orders of magnitude between the most abundant protein serum albumin and cytokines, which are of great relevance as they drive disease processes [10]. LC-MS/MS with DDA has the tendency to identify peptides from high abundant proteins [6, 9]. However, in many cases it is not the high abundance proteins we are interested in. Additionally, proteins that are used as biomarkers for diseases (e.g., prostate-specific antigen) are usually present in a concentration several magnitudes lower than the most abundant serum proteins [11]. There are experimental procedures such as immunoaffinity precipitation to deplete these high abundant but analytical mainly unimportant proteins [12–15]. However, they are time-consuming and

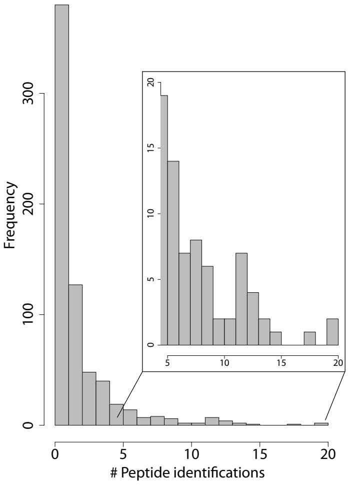


Figure 1.5.: Distribution of significant peptide evidences. Human cell lysate with 13,546 detected features, 1,074 significant peptide identifications matching 670 proteins. While most proteins have only one peptide identification there are a few proteins with more than 10 peptide matches.

expensive, and change the sample stoichiometry, posing a problem for protein quantification. Additionally, the removal of proteins like albumin, which bind a variety of compounds including other proteins, might result in the loss of low abundant proteins [11].

Furthermore, the high redundancy achieved with DDA is unnecessary. Once a protein is identified the detection of additional peptides usually does not yield further information. Figure 1.5 shows the number of significant peptide evidences per protein for an LC-MS/MS analysis of a complex human cell lysate. It can be seen that a few proteins assemble each more than 10 peptide identifications, whereas the majority is identified by only one peptide. Limiting this redundancy might lead to an increased number of protein identifications.

As a peptide usually elutes from the column over several fractions, there are often several possibilities to fragment it. Hence, high abundant peptides would be selected several times with normal DDA, without providing more information. This also means, that we can decide whether to fragment a specific peptide depending on other signals present in the same fraction in order to optimize the number or the set of selected precursors.

An advantage of DDA is that no additional information about the sample is required. It can be used straight away to discover unknown proteins in a sample. However, there are many cases where prior information about the signals in the sample is available. Here, directed approaches that "search" for signals of interest are usually more suitable.

In the last paragraphs we illustrated common problems with the standard precursor ion selection strategies. These problems show why it is important and promising to apply more elaborate precursor ion selection strategies. Especially MALDI, where LC and MS are decoupled and there is no time constraint, is well suited for more sophisticated approaches. In the following, we describe two different precursor ion selection scenarios and how they can be traced back to combinatorial problems.

1.3. Precursor ion selection as Knapsack Problem

Precursor ion selection based on an existing LC-MS feature map can be seen as an adaptation of the *Knapsack Problem*. This is a well-known combinatorial problem: Given a set of items with each having a weight and a value assigned and a knapsack with a weight limit, we want to find a set of items that does not exceed the knapsack's weight limit and has the highest possible value.

Loosely speaking, imagine we want to board a plane with only hand luggage. Now, we have a set of cosmetics we would like to take, that each have a certain volume and a price as illustrated in Figure 1.6. Safety regulations at the airport allow only liquids that fit into a one liter bag, our knapsack. So we want to find a set of cosmetics that fits into the bag and reaches the maximal possible value (and we thus have to invest the smallest possible amount of money to buy the rest we cannot take). [2]

In our LC-MS/MS setup, each spectrum corresponds to a knapsack. Here, the weight limit is the number of possible MS/MS spectra per RT bin. The value of a precursor is its intensity and each precursor has the same weight 1. An LC-MS/MS run consists of multiple spectra, thus we have a multi Knapsack Problem. In our setup, we have the additional constraint that each observed feature shall be selected only in one spectrum as a precursor. The goal is to find a maximal number of precursors given our set of features. By formulating this task as a combinatorial problem, we can develop a *linear program* (LP). Solving the LP yields the demanded precursor set.

1.4. Precursor ion selection as Hitting Set Problem

In proteomic studies one is often interested in protein identification and/or quantification. However, as protein identification is done indirectly by inferring proteins from sequenced peptides, often only the number of peptide identifications

[2] For hand luggage at the airport more constraints apply, but for simplicity we leave it at the ones stated above.

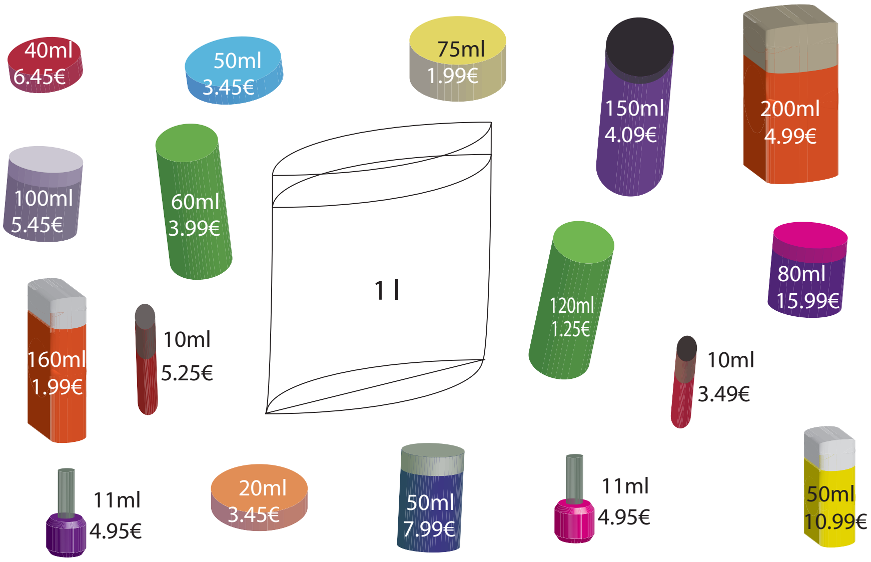


Figure 1.6.: The Knapsack Problem. Illustrated through cosmetics when packing the hand luggage for airplane travel. The plastic bag represents the knapsack with the volume limit of 1 l. The cosmetics each have a weight and a value. The goal is to find a set of cosmetics with maximal weight that does not exceed the volume limit of the knapsack.

is tried to maximize. Incorporating peptide-protein relations into the precursor ion selection might prevent identification of a few abundant proteins with many peptides while the protein majority lacks peptide evidences.

In our approach, we trace precursor ion selection back to the *Hitting Set Problem*, another well-known combinatorial problem. As illustrated in Figure 1.7, we have given a set of circles and a set of rectangles which separate the circles into groups that may partially overlap. In our application, peptides correspond to circles and proteins are represented by rectangles. The aim is to find a minimal set of peptides or circles so that each protein or rectangle is "hit" by at least one peptide or circle. Again, we can formulate a linear program for this precursor ion selection problem. This way, we achieve a targeted selection of peptides for all proteins of interest. In practice, we need to adapt the original *Hitting Set Problem* as peptides shared by several proteins are favored over unique peptides and thus, we cannot distinguish between proteins that share peptides. Additionally, for precursor ion selection, the selected peptides need to be translated into precursor ions. Thus, m/z and RT need to be reliably predicted. Furthermore, not all tryptic peptides of a protein can be observed in a given experimental setup. For instance, very hydrophobic peptides strongly interact with the LC column and thus might never elute with standard gradients [16]. Other peptides might have a low ionization efficiency. Thus, for each setup and protein one can define a set of proteotypic peptides that can be observed frequently. With machine learning techniques, weights can be predicted for each peptide reflecting its proteotypicity. By incorporating these weights into the LP formulation, the selected precursors correspond to a representative set of peptides for each protein.

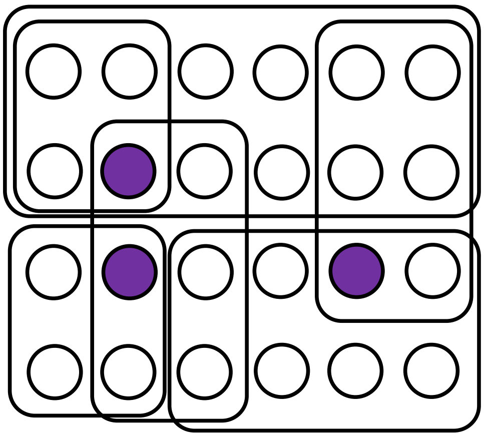


Figure 1.7.: Hitting Set Problem. Given a set of rectangles and a set of circles lying in the rectangles, find a minimal set of circles so that each of the rounded rectangles is hit by at least one circle. One possible minimal set is shown in purple.

1.5. Iterative precursor ion selection

In the last sections, we briefly described two scenarios for inclusion list creation prior to MS/MS acquisition. Now, we are introducing a different concept: iterative precursor ion selection. As MALDI allows to interrupt the MS/MS acquisition, it is possible to incorporate the information about peptide and protein identifications obtained so far into the current precursor ion selection step and then continue with MS/MS acquisition. We combine the two presented inclusion list problems with an exclusion strategy aiming at avoiding the selection of precursors possibly belonging to already identified proteins.

1.6. Contributions

In this thesis, we address the described problems of DDA and develop tools that help to circumvent them.

- We develop strategies for inclusion list creation based on a formulation of the selection process as optimization problem. We exemplarily explain two different scenarios for inclusion lists and show that these can be traced back to known combinatorial problems. Our framework allows for easy adaption of the selection depending on the aim of the study. We make use of protein-peptide relations and the 3D nature of peptide signals in LC-MS in order to select an optimal set of precursors. We develop protein detectabilities as a measure for protein coverage achieved with predicted precursors and utilize them in our setup.
- We introduce an iterative precursor ion selection procedure that combines

the discovery nature of DDA with directed MS/MS. This approach is especially suited for LC-MALDI MS/MS where the sample is “frozen in time” on the target plate. We develop a simple proof-of-concept heuristic and show that applying this approach leads to protein identifications using significantly less selected precursors than standard procedures.

- Thereupon, we develop a mathematical formulation for the iterative precursor ion selection addressing the problems observed with the heuristic.
- The presented methods are implemented as part of OpenMS, an open-source C++ software library for mass spectrometry.
- We implemented an online version of the iterative precursor ion selection that has direct access to the mass spectrometer and controls the measurement. This tool has a graphical user interface that allows to easily adapt the parameters for both the acquisition as well as for the processing of MS/MS spectra.

1.7. Thesis outline

Following this introduction, we present an overview of the background needed for the rest of this thesis in **Chapter 2**. We start with a description of LC-MS instrumentation. Afterwards, we explain how to derive peptide and protein identifications from MS/MS spectra. Then, the prediction of peptide characteristics is briefly introduced. Finally, we give an introduction to linear programming.

Chapter 3 summarizes the current state of the art in precursor ion selection for LC-MS/MS and presents the different approaches to peptide sequencing with MS/MS.

In **Chapter 4** we describe the samples used for algorithm evaluation. This is followed by a short overview of the sample preparation, data acquisition and processing. Model training for prediction of peptide characteristics is explained and finally evaluated.

In **Chapter 5** we present different problems for inclusion list creation with LC-MALDI MS/MS, translate them into optimization problems and evaluate the solutions. First, we show how to formulate the selection of LC-MS features as maximization problem and compare that to other methods. This is followed by targeted inclusion lists created based solely on protein sequences without prior MS acquisition.

Chapter 6 describes how precursor ion selection can be adapted during MS/MS acquisition depending on the results achieved so far. We present two iterative algorithms that we developed. On the one hand a heuristic that proceeds on a ranked list of precursors. The ranking is adapted throughout the measurement based on the identification results. Then, we combine the two problems presented in Chapter 5 and use them in an LP formulation that maximizes the number of selected features and targets at confirming protein hits (proteins with peptide evidences that did not yet exceed a given significance threshold).

Chapter 7 presents details of the implementation of the tools described in this thesis. Furthermore the OnlinePrecursorIonSelector is presented, a graphical tool that has access to the mass spectrometer and controls the measurements. This is followed by a conclusion in **Chapter 8**.

1.8. Related publications

The heuristic iterative precursor ion selection presented in Chapter 6 was described previously in a publication in the Journal of Proteome Research [17]. The contributions were as follows: Zerck and Gobom developed the main idea. Zerck, Nordhoff and Gobom designed the experiments and performed the evaluation. Lukaszewska, Resemann and Zerck performed the measurements. Zerck implemented the algorithm. Reinert provided supervision.

Chapter 5 was described in a publication in BMC Bioinformatics [18]. The iterative precursor ion selection with linear programs presented in Chapter 6 was introduced in the same manuscript. Here, the contributions were: Zerck developed the LP formulations, did the implementation and evaluation under the supervision of Reinert. Nordhoff was involved in the conception of the study.

Chapter 2 Background

In this chapter, we present the experimental and mathematical background needed in the following chapters. First, we give a short introduction to liquid chromatography and mass spectrometry. Afterwards, we describe how to retrieve information about the peptides and proteins present in the sample and how to determine statistical significance of the identifications. This is followed by an overview of the prediction of peptide properties using machine learning techniques. Finally, we present an introduction to linear programming and a few well-known combinatorial problems, onto which the problems described in the next chapters can be traced back.

2.1. Liquid chromatography-Mass spectrometry

2.1.1. Liquid chromatography

Because proteomic samples are highly complex, it is not possible to analyze them directly via mass spectrometry. A preceding separation step is added to reduce the complexity, e.g., 2D electrophoresis or chromatographic techniques. Liquid chromatography (LC) is the most widely used approach in MS-based proteomics, hence we are focusing on it.

In LC, analytes are separated by their different interaction behavior with the *mobile phase* (solvent) and the *stationary phase* (the solid material of the chromatographic media). The stationary phase exhibits functional groups that interact with the analyte molecules and the mobile phase. Depending on the physico-chemical properties of the analyte, the mobile phase, and the stationary phase, the analyte components take different times to flow through the column. The time a molecule needs to elute from the column is called retention time (RT). It is specific for the molecule in a given setup, molecules with similar physico-chemical properties elute at similar RTs. The LC system can be coupled directly to the mass spectrometer. This is most commonly done with ESI-MS. For MALDI-MS, usually discrete fractions are collected by time onto a MALDI sample plate.

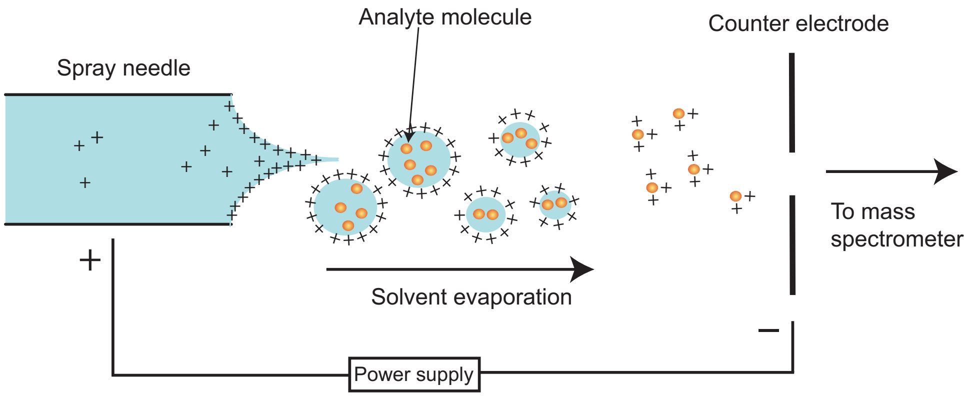


Figure 2.1.: Electrospray ionization process.

2.1.2. Mass spectrometry

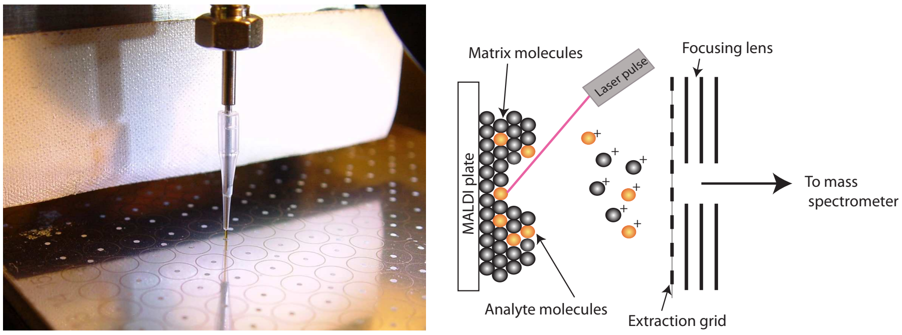
A mass spectrometer consists of the three main components: ion source, mass analyzer, and detector. In the ion source the conversion from neutral molecules to gaseous ions takes place. These ions are separated according to the ratio of their mass to charge (m/z) in the mass analyzer and afterwards the detector records the mass spectrum, which contains the information in what quantity ions were detected [19–21].

While several different mass spectrometric ionization techniques have been discovered, there are two which are used in proteomics: Electrospray Ionization (ESI) and Matrix-assisted Laser Desorption/Ionization (MALDI), which are briefly introduced in the following sections.

Electrospray Ionization

Electrospray Ionization (ESI) for MS was developed in the lab of John Fenn in 1984 [22]. This study was based on the work of Malcolm Dole [23] from 1968, who proposed Electrospray Ionization to produce beams of charged macromolecules. Figure 2.1 gives an illustration of ESI.

When an LC system is directly coupled to the ESI source, the eluent flows through the electrospray needle, which has a high potential difference to the counter electrode applied to it. Thereby, positively charged droplets form, consisting of analyte and solvent molecules. The solvent evaporates while the droplets move to the counter electrode. This means an instability of the droplets, that finally leads to singly and multiply charged analyte molecules.



(a) MALDI spotting

(b) Ionization with MALDI

Figure 2.2.: MALDI ionization and fractionation.(a) A fractionator spotting the sample after LC separation onto the MALDI target plate. (The photo was taken by Klaus-Dieter Kloeppel.) (b) MALDI ionization process.

Matrix-assisted Laser Desorption/Ionization

Matrix-assisted Laser Desorption/Ionization (MALDI) was developed simultaneously by Michael Karas and Franz Hillenkamp [24] at the University of Muenster (Germany) and Koichi Tanaka at Shimadzu Corporation (Japan) in 1987 [25, 26]. Here, the analyte is embedded into the crystal lattice of an organic solution called matrix, with a high molar excess of matrix typically between 100:1 and 10,000:1 [26]. Usually, the sample is either prepared using the thin-layer method onto previously prepared microcrystalline layer of the matrix, or both solutions are mixed and afterwards applied onto the target plate (dried droplet). Figure 2.2 shows a fractionator spotting the fractionated sample after LC separation onto the MALDI target plate.

For ionization, the crystalline sample is irradiated with a brief laser pulse, by which the matrix molecules absorb most of the energy, leading to desorption. In this process, matrix molecules entrain the embedded analyte molecules, which are also transferred into gas phase. Ionization of the analyte can occur at any time during this process [26]. For peptides, mainly singly-charged ions are produced, while larger biomolecules yield more multiply charged species.

Mass analyzer

MALDI is most frequently used in conjunction with a time-of-flight analyzer (TOF). Figure 2.3 gives a schematic overview of a TOF analyzer. Here, after ionization the ions are accelerated in a strong electric field, then enter a field-free region, where they drift freely until they hit the detector. The ions are separated according to their molecular weight, as lighter ions are faster than heavier ones. The flight time t of an ion can be converted into an m/z-value using the following

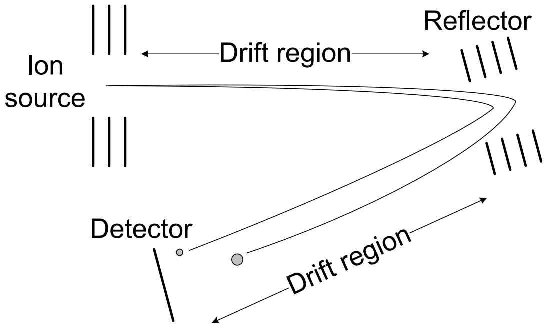


Figure 2.3.: Time-of-flight mass spectrometer with reflector. The reflector corrects for small kinetic energy differences of ions with the same m/z-value as faster ions penetrate deeper into the reflector.

equation:

$$t = a\sqrt{\frac{m}{z}}, \tag{2.1}$$

where a is an instrument specific constant. In order to increase the resolution, Reflector-TOF instruments use an electrostatic mirror after the drift region to correct for different energies of ions with the same m/z-value. Higher energetic ions penetrate deeper into the reflector, thus having a longer path to the detector [27].

The resolution of an instrument is defined at a specific m/z-value as ratio of the m/z of a peak and its full-width-at-half-max (FWHM). It is a measure of how good an instrument can separate (isotopic) peaks. Depending on the type of instrument, the resolution can be approximately constant over the mass range as for Quadrupoles and Ion traps, linear for QTOFs and TOFs or for Orbitraps inversely proportional to the square root of m/z [28].

An important parameter for the analysis of an LC-MS run is mass accuracy, which can be calculated as absolute value using the theoretical m/z of a compound m_{theo} and the observed value m_{obs}

$$ma_{abs} = |m_{obs} - m_{theo}|, \tag{2.2}$$

or, which is commonly used nowadays, relatively as parts-per-million (ppm):

$$ma_{rel} = 10^6 \cdot \frac{|m_{obs} - m_{theo}|}{m_{theo}}. \tag{2.3}$$

2.1.3. Tandem Mass Spectrometry

In tandem mass spectrometry, a second MS step is performed on previously isolated peptide ions. Usually, peptide ions, the so-called precursor ions, are isolated within a small m/z window. These ions are then subjected to further fragmenta-

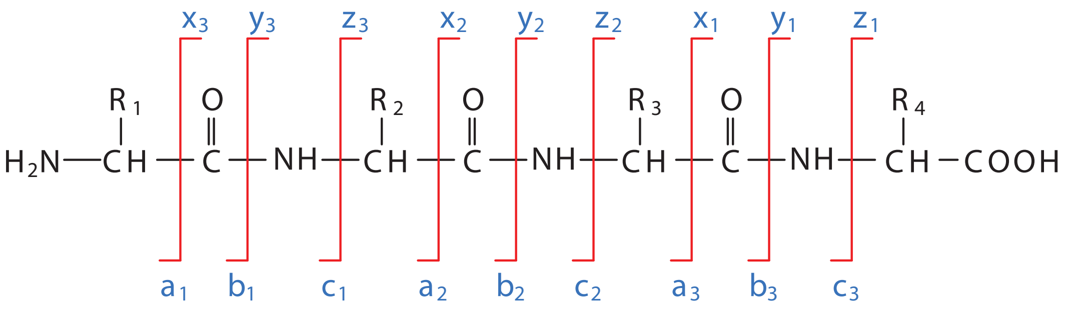


Figure 2.4.: Peptide fragmentation with MS/MS. The fragment ion types according to Roepstorff's nomenclature [31].

Table 2.1.: Fragmentation techniques for MS/MS and their primary ions [29, 34].

Name	Abbreviation	Primary ions
Collision-induced dissociation	CID	b, y
Electron-capture dissociation	ECD	c, z
Electron-transfer dissociation	ETD	c, z
Electron-detachment dissociation	EDD	a, x
MALDI-Post source decay	PSD	b, y
MALDI-In-source decay	ISD	c, z

tion. Depending on the instrumentation this step is performed in an additional analyzer as in Triple quadrupoles and TOF/TOF instruments, or consecutively within the same analyzer like in ion traps [29]. For peptide ions, the most widely used fragmentation method is collision-induced dissociation (CID). After isolation of the precursor ions, they get accelerated using an energy potential. Then the precursor ions collide with neutral gas molecules like helium, nitrogen or argon. During the collision the internal energy increases, which leads to peptide fragmentation at specific bonds [29, 30]. Typical fragment ion types occurring after MS/MS are illustrated in Figure 2.4. According to Roepstorff's nomenclature [31], ions with the charge retained on the N-terminal side are denoted as *a, b* or *c* ions, depending on the position of the fragmentation with respect to the peptide bond. Analogously, fragment ions with the charge retained on the C-Terminus are called *x, y* and *z* ions.

Different fragmentation methods produce different types of ions, for an overview see Table 2.1. Thus, the use of complementary ionization techniques can help to improve peptide identification rates [32, 33].

2.2. Peptide identification

2.2.1. Generation of peptide-spectrum matches

After signal processing of the MS/MS spectra, we want to assign peptide sequences to them. There exist two complementary approaches: database searching and *de novo* sequencing.

Given protein sequences of a given species, database search methods compile a set of peptides that lie in the m/z-range of the precursor of the specific MS/MS spectrum. Theoretical spectra are generated for this peptide set and matched to the MS/MS spectrum. Various database search tools were developed, examples are Sequest [35], Mascot [36], X!Tandem [37], and OMSSA [38]. See [39, 40] for an overview and evaluation of some of the most prominent database search tools.

There exist several scenarios where a database search is not sufficient to identify peptides in a sample. These include samples from species with unsequenced genome, protein sequence variants or splice isoforms and the analysis of peptides with non-proteogenic or modified amino acids as they appear in bacteria or fungi [29]. Examples for the various *de novo* sequencing tools are Lutefisk [41, 42], SeqMS [43, 44], and Pepnovo [45]. There also exist *de novo* approaches which exploit the complementary nature of different ionization methods like CID and ETD [32, 33]. See [46] for an evaluation of different *de novo* tools.

Figure 2.5 shows an MS/MS spectrum with the theoretical spectrum of the best matching peptide. The matching peaks of the b- and y-ion series are annotated.

2.2.2. Scoring of PSMs

All tools that generate peptide-spectrum matches (PSM) rank these for each spectrum according to some scoring scheme. However, these scores cannot easily answer the question which PSM is correct, as score distributions for correct and incorrect matches overlap. There has been much effort in the last years to assign statistical scores to PSMs, facilitating the decision which PSM can be considered correct. The task to decide whether a PSM is a true or a random match is a classification task. In the following sections we are shortly presenting different statistical measures used for scoring of PSMs.

Construction of incorrect PSMs

In order to develop statistical scores we need to study the occurrence of random PSMs. This is often done using decoy databases which contain amino acid sequences that should have similar properties as the original database (e.g., AA

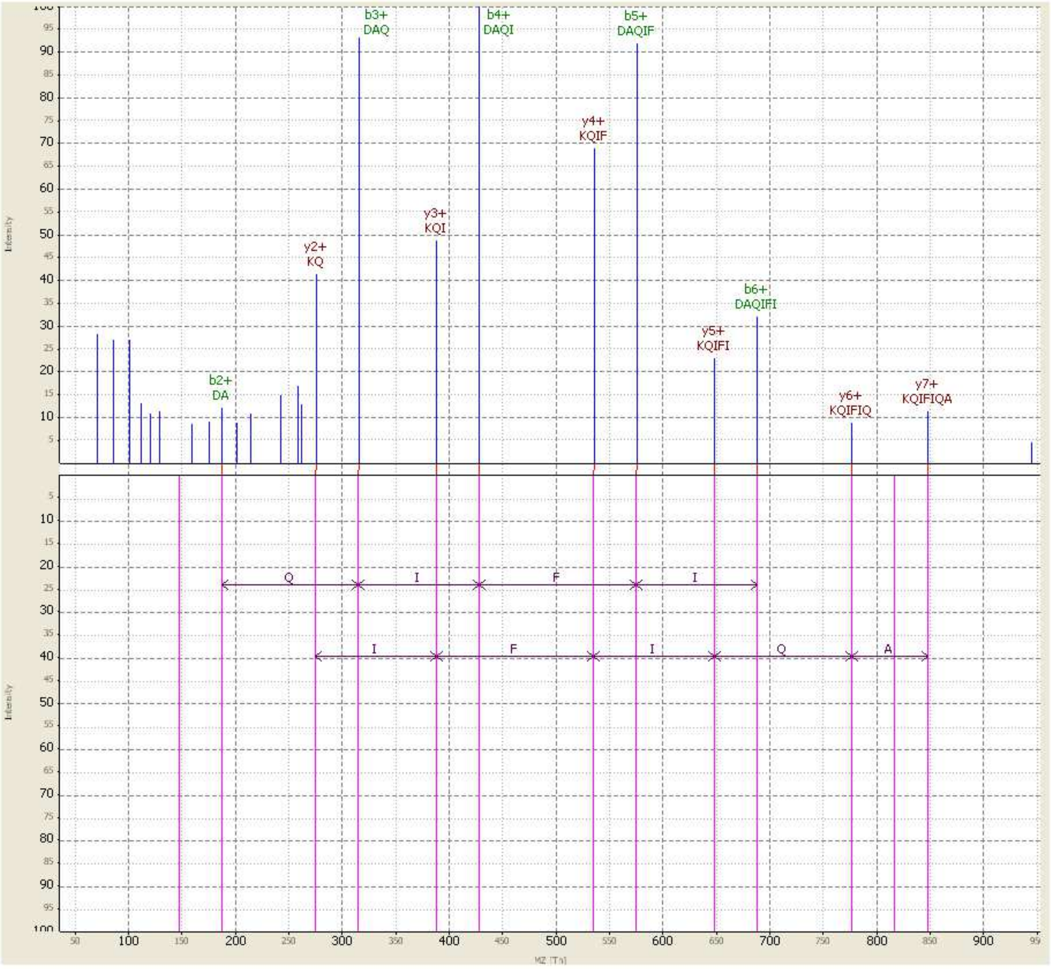


Figure 2.5.: Annotated MS/MS spectrum with the theoretical spectrum of the highest scoring peptide (DAQIFIQK) underneath. Visualized with TOPPView [47]. The matching peaks of the *b*- and *y*-ion series are annotated with their corresponding peptide fragment.

composition, number of tryptic peptides, peptide length and mass) but contain sequences that do not occur in the original database. Hits in such a database are all false positives. Decoy databases can be constructed by shuffling or reversing the original sequences, also random sequence construction approaches exist. However, there is no consensus which method is the best. See Bianco et al. [48] for a comparison of different construction methods. There are also disadvantages of the decoy database approach. First, the search space is increased which also increases the search time. Besides, such a database cannot be constructed for all applications such as error-tolerant searches or *de novo* sequencing.

p- and *E*-Values

One of the most commonly used statistical measures is the *p*-value. Given a null hypothesis, it is the probability to achieve a result at least as extreme as the observed. In other words, it describes the probability that a result occurs simply by chance given a true null hypothesis. In our case the null hypothesis would be that a given peptide is not represented by the assigned MS/MS spectrum. Without a loss of generality we assume in the following a scoring scheme where higher scores indicate better scores. Following Käll et al. [49], the *p*-value for a PSM with score s can be calculated as

$$p(s) = \frac{\text{\#decoy PSMs with score} \geq \text{s}}{\text{\#decoy PSMs}}. \tag{2.4}$$

However, as we want to calculate a score for all PSMs, this test is performed many times and thus needs to be corrected for multiple testing. Otherwise, with several thousand PSMs the percentage of small *p*-values simply by chance is not negligible, thus the number of correct PSMs would be overestimated.

Similar to the *p*-value, several search engines calculate a so-called *E*-value which can be interpreted as the expected number of peptides with a score at least as high as the observed score simply by chance [50]. This way the *E*-value corrects for the number of candidate peptides in the database. However, the *E*-value also does not account for the number of spectra being matched [50].

A simple correction method for multiple testing is the *Bonferroni correction*, where *p*-values are divided by the number of tests that are performed. However, the corrected *p*-value is very conservative and overestimates the fraction of spurious hits.

The two statistical measures we briefly introduce in the next sections account for multiple testing.

False discovery rates and q-values

Storey and Tibshirani [51] propose a method to calculate false discovery rates (FDR) and q-values based on p-values. They approximate the FDR for a given p-value threshold t as

$$FDR(t) \approx \frac{E\left[\#\{\text{null } p_i \leq t; i = 1, ..., m\}\right]}{E\left[\#\{p_i \leq t; i = 1, ..., m\}\right]}. \tag{2.5}$$

$p_1 \ldots p_m$ are the m p-values we are considering, null p_i is a p-value of a feature for which the null hypothesis is true. In the case of PSMs this corresponds to an incorrect PSM. The denominator can be simply estimated by the number of observed p-values $\leq t$. When correctly calculated, the null p-values are uniformly distributed. Thus, the probability of a null p-value $\leq t$ is given by t [51]. Hence, the numerator can be estimated as $\hat{\pi}_0 \cdot m \cdot t$, with $\hat{\pi}_0$ being the estimated proportion of truly null features. This leads to an estimated FDR:

$$\widehat{FDR}(t) = \frac{\hat{\pi}_0 \cdot m \cdot t}{\#\{p_i \leq t; i = 1, ..., m\}}. \tag{2.6}$$

In the literature, there exist two similar ways to calculate the FDR for PSMs. The above FDR definition requires information about the number of incorrect PSMs which is often acquired via decoy databases as hits in this database ideally are truly random. Target-decoy searches can either be done in two separate searches, once searching the target database and once searching the decoy database, or in one search using a combined target-decoy database. Following the FDR calculation from Equation 2.5, this leads to the estimation of the FDR as

$$\widehat{FDR}(s) = \frac{2 \cdot N_d}{N_d + N_t}, \tag{2.7}$$

where N_d is the number of hits to the decoy database passing threshold s and N_t the number of target hits passing the threshold [52–54]. The numerator should correspond to the number of incorrect hits, which is unknown. However, it is assumed that there are as many false hits to the normal database as there are hits to the decoy database.

A similar estimation is:

$$\widehat{FDR}(s) = \frac{N_d}{N_t} \cdot \hat{\pi}_0, \tag{2.8}$$

which is used by Käll et al. [49, 55] for separate target-decoy searches. Here, $\hat{\pi}_0$ is used to correct for the overestimation of incorrect matches given by N_d.

There also exist approaches which calculate the FDR without decoy databases, e.g., with spectral probabilities [56] or a mixture modeling approach [57].

A drawback of the FDR is that a smaller score threshold can lead to a smaller estimated FDR [49]. Storey and Tibshirani [51] addressed this problem in the context of genome-wide studies by proposing the *q-value*. Käll et al. [49] introduced this term in the context of PSMs. Here, the q-value for a given PSM with score s is defined as the minimal FDR-threshold at which the PSM is accepted.

Posterior (Error) Probabilities

False discovery rates are suitable when one is interested in a group of proteins, e.g., when determining which proteins are expressed in a cell type or when one is looking at sets of PSMs [50]. In contrast, if we are interested in a specific peptide or protein, calculating posterior error probabilities (PEP) is the method of choice [50]. Sometimes the PEP is also referred to as local FDR [50, 54].

The PEP for a PSM of peptide p and spectrum s gives the probability that the observed PSM is incorrect [50]. Or as Käll et al. [50] state a PEP of 0.01 implies a probability of 99% that peptide p was in the mass spectrometer during the creation of s. This posterior probability (PP) is calculated as:

$$PP = 1 - PEP \tag{2.9}$$

The basic assumption in the calculation of PEPs is that the distribution of search engine scores actually consists of two parts: one distribution for incorrect PSMs and another one for correct matches. Typical distributions used for incorrect matches in the mixture model are Gumbel or Gaussians [58].

Parameters for the mixture model are learned using labeled training data, in our case the labels are target or decoy, reflecting in which part of the database the best PSM is found. Learning is done with an Expectation-Maximization (EM) approach. In the expectation step, posterior probabilities are estimated using Bayesian statistics and initial guesses for the mixture model parameters. In the maximization step, estimated probabilities are used to fit the distributions and thus to adapt the model parameters [58].

The posterior probabilities are calculated using Bayes' law. The PEP for a peptide with score s is:

$$p(-|s) = \frac{p(s|-)p(-)}{p(s|-)p(-) + p(s|+)p(+)}. \tag{2.10}$$

$p(-)$ and $p(+)$ are the prior probabilities of a false and a correct match. The probabilities of achieving score s given that a match is false or correct are denoted as $p(s|-)$ and $p(s|+)$. They can be calculated using the score distributions of the correct and incorrect matches.

A widely used tool that computes posterior probabilities, not PEPs, is PeptideProphet [59]. In this thesis we used the tool IDPosteriorErrorProbability [58],

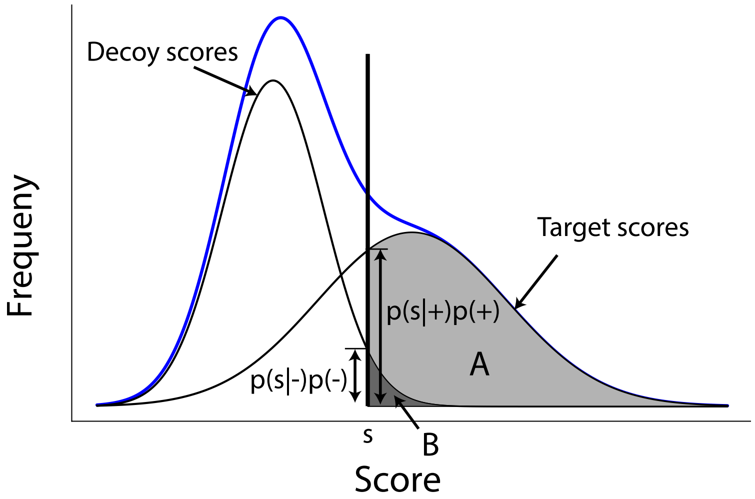


Figure 2.6.: The relation between FDR and PEP. The blue line represents a histogram of peptide scores. The two black lines represent the distribution of the target and decoy scores. The FDR is calculated using the areas under the scoring distributions as $FDR = \frac{B}{A+B}$, where A and B are the number of target and decoy scores $> s$. When calculating the PEP, the heights of the distributions are used : $PEP = \frac{p(s|-)p(-)}{p(s|-)p(-)+p(s|+)p(+)}$. Figure reproduced from [50] and [61].

available as part of TOPP [60].

The relation between FDR and PEP is shown in Figure 2.6. See Käll et al. [50] for a detailed comparison of FDR and PEP.

2.3. Protein identification

In the last sections we introduced measures for peptide identification significance. In the following, we address the problem of deriving protein identifications from given peptide identifications.

2.3.1. Protein inference

The *protein inference problem* describes the task of assigning peptide matches to protein identifications. This is particularly challenging since peptides might occur in more than one protein. Thus, more than one correct solution exists. These peptides are called *degenerate* or *shared peptides*. In general, we are interested in a minimal protein list explaining all given peptide identifications. Such a scenario is often referred to as Occam's razor [62], which is a principle that prefers to select the solution among several possible solutions which makes the smallest number of assumptions. Fig 2.7 shows the protein inference problem for a small set of peptide identifications a_1 to a_8. The minimal set of proteins covering all peptide

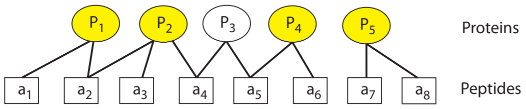


Figure 2.7.: The protein inference problem. Given a peptide set $\{a_1, \ldots, a_8\}$ and a protein set $\{P_1, \ldots, P_5\}$, we want to find a minimal set of proteins that covers all peptides. Here, the set $\{P_1, P_2, P_4, P_5\}$ is the minimal set.

identifications is $\{P_1, P_2, P_4, P_5\}$.

2.3.2. Protein identification measures

Even if we have found such a minimal solution of protein identifications, it remains unclear which identifications should be trusted. Peptide identification algorithms usually provide a measure of confidence such as score or probability as presented in the last section. Peptide confidences need to be combined to yield a measure of protein identification confidence. Although several different methods have been already proposed up to now, there exist no established criteria for determining whether a protein has been identified in an experiment. In the following, we introduce common strategies, including simple (unique) peptide counting and probability-based criteria.

Unique peptide counting

In 2004, the journal *Molecular & Cellular Proteomics* published its MCP guidelines that included publications standards [63]. Due to the rising number of publications containing peptide and protein identification data, Carr et al. proposed guidelines for authors about the information that should be included in their manuscripts. An important aspect addressed proteins identified by a single peptide hit. The authors claimed that proteins supported by only one peptide hit are more likely to be incorrectly assigned than proteins with two or more peptide hits. Today this is known as "two-peptide rule" [64], a widely accepted recommendation among experimentalists to require at least two unique peptide matches for a protein identification [65, 66]. However, by discarding single-hit proteins many high-quality protein identifications are lost [65, 66] as in a typical high-throughput experiment several hundred proteins are "one-hit wonders" [65]. As Higdon and Kolker [65] show, the false-discovery rate decreases with the number of peptide identifications required for a protein identification, making it hard to legitimate the need for two and not three or more peptide identifications per protein. Gupta and Pevzner [66] showed that the one-peptide rule outperforms the two-peptide rule in terms of FDR, i.e., they show that two medium score hits are more likely to occur simply by chance than one high scoring hit.

False Discovery Rates

Similar to the FDR calculation for peptides, see Section 2.2.2, an FDR on protein level can be computed using a decoy database [67, 68]. However, again this does not tell us anything about the probability of a single protein to be present in a sample as the FDR provides a measure of global error rate. Evaluating the FDR on different scores and determining an optimal threshold can be used for filtering.

Probability based protein identification

Several approaches exist to compute protein probabilities based on peptide identifications [65, 69–72]. Probably the best-known method is ProteinProphet developed by Nesvizhskii et al. [70] in 2003. ProteinProphet constructs minimal protein lists explaining all peptide identifications using the expectation-maximization (EM) algorithm. It computes a protein probability as the probability that at least one of the corresponding peptide identifications is correct. In Figure 2.8 an example is shown. First, peptides are grouped according to their corresponding proteins. Then, the probability that protein $prot_i$ is in the sample can be computed as

$$P(prot_i) = 1 - \prod_{pep_k \in prot_i} (1 - P(pep_k)), \tag{2.11}$$

with $P(pep_k)$ being the posterior probability of peptide pep_k. Peptide probabilities are adjusted by using the number of peptides corresponding to the same protein. Degenerate peptides are given an iteratively determined weight, thereby keeping the protein list minimal [70].

Li et al. [71] use Gibbs Sampling to calculate a protein list that maximizes the joint probability of protein indicator variables given peptide indicator variables. An indicator variable can only have the values 0 or 1, in our case a peptide or protein indicator variable is 1 if the peptide/protein was identified and 0 otherwise. As prior probabilities the authors use predicted peptide detectabilities adjusted by estimated protein abundances.

2.4. Prediction of peptide characteristics

Based on their amino acid composition peptides display different characteristics upon LC-MS/MS analysis. One property discussed in this thesis is the detectability of a peptide in a given LC-MS/MS setup. The detectability is the probability to detect and to identify a peptide by LC-MS/MS. This includes the probability of the peptide eluting from the LC column in the observed time frame, being

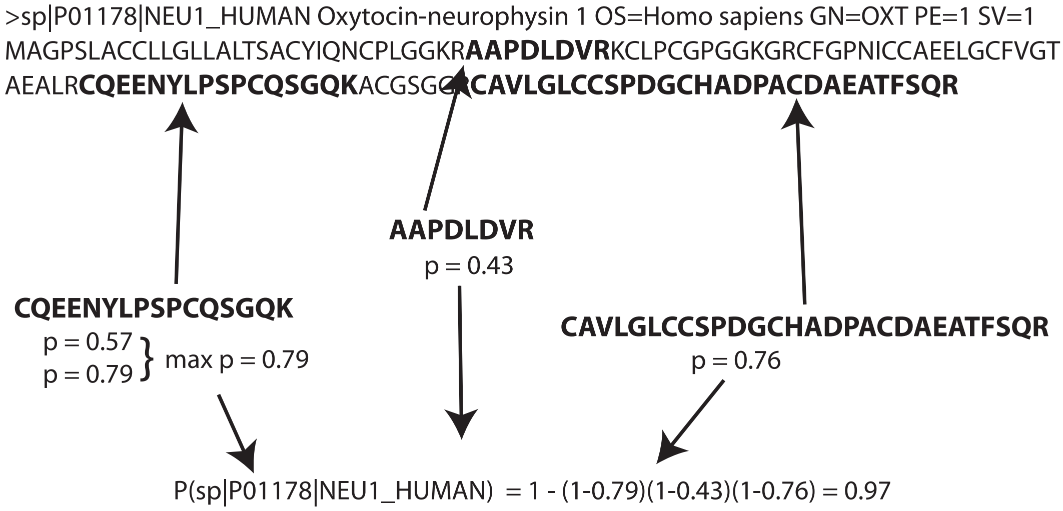

>sp|P01178|NEU1_HUMAN Oxytocin-neurophysin 1 OS=Homo sapiens GN=OXT PE=1 SV=1
MAGPSLACCLLGLLALTSACYIQNCPLGGKRAAPDLDVRKCLPCGPGGKGRCFGPNICCAEELGCFVGT
AEALRCQEENYLPSPCQSGQKACGSGGRCAVLGLCCSPDGCHADPACDAEATFSQR
AAPDLDVR
p = 0.43
CQEENYLPSPCQSGQK
p = 0.57
p = 0.79
max p = 0.79
CAVLGLCCSPDGCHADPACDAEATFSQR
p = 0.76
P(sp|P01178|NEU1_HUMAN) = 1 - (1-0.79)(1-0.43)(1-0.76) = 0.97

Figure 2.8.: Protein probability calculation. Given peptide identifications with corresponding peptide probabilities, the protein probability is calculated as the complementary probability that none of the peptides is identified correctly. Reproduced from [70].

ionized in the ion source, having an m/z-value that can be detected by the mass spectrometer, the ion being selected as precursor, the ion's suitability for fragment ion analysis, and the correct identification of the peptide.

Another important characteristic is the retention time of a peptide in a specific LC setup. Many different types of chromatographic media exist that separate peptides depending on, e.g., size, charge, polarity, or hydrophobicity.

There exist machine learning tools for the prediction of these peptide characteristics. In the next section we shortly describe the tools used in this thesis.

2.4.1. RT Prediction

Different approaches have been developed for predicting retention times. Often machine learning techniques like support vector machines (SVMs) [73–75] or artificial neural networks [76] are used. In our study, we used the approach by Pfeifer et al. [74] as it showed a good performance, requires a relatively small number of training peptides, and is easily available as part of TOPP [60].

In their study, Pfeifer et al. developed the (paired) oligo-border kernel ((P)OBK). The approach directly works on the amino acid sequence of peptides and also distinguishes between different post-translational modifications (PTMs). The OBK tries to identify signals or motifs in the borders of peptides, where border corresponds to the leftmost and rightmost residues and is of fixed length. The POBK used in this work considers the left and right border in one common oligo function and thus can detect similarities between opposite borders. The POBK is used in a Support Vector Regression (SVR) as the labels (in our case the RTs) are continuous. Using a training set of high-confidence peptide identifications, a model is learned with the TOPPtool RTModel. The trained model can then be used to predict RTs for new peptide sequences.

2.4.2. Prediction of peptide detectabilities

Similar to the RT, the detectability of a peptide can be predicted. This can either be a classification problem, where we try to distinguish between observable and unobservable peptides.[1] Or, as in our case, the labels are continuous likelihoods reflecting whether a certain peptide can be detected, so again SVR in conjunction with the POBK is applied. We use the TOPPtool PTModel [77] which needs a number of high confidence peptide identifications and additionally a set of undetectable peptides to train a model.

2.5. Linear programming

2.5.1. Introduction to linear programming

Linear programming is an optimization technique where a linear objective function should be optimized subject to linear equality and/or inequality constraints. Bertsimas and Tsitsiklis [78] define a linear programming problem (LP) as: Given a cost vector $c = (c_1, ..., c_n)$, we want to minimize the linear objective function $c'x = \sum_{i=1}^{n} c_i \cdot x_i$ over all vectors $x = (x_1, ..., x_n)$ subject to linear constraints. For each constraint i a vector a_i and a scalar b_i are given. The three kinds of constraints ($\geq$,$\leq$,$=$) are formed using three index sets S_1, S_2 and S_3. Additionally, size constraints on the variables x_j can be given. Thus, an LP can be written as:

$$\begin{aligned}
\min \quad & \sum_i c_i \cdot x_i && (2.12)\\
\text{subject to:} \quad & a_i'x \geq b_i, \quad i \in S_1, && (2.13)\\
& a_i'x \leq b_i, \quad i \in S_2, && (2.14)\\
& a_i'x = b_i, \quad i \in S_3, && (2.15)\\
& x_j \geq 0, \quad j \in S_4, && (2.16)\\
& x_j \leq 0, \quad j \in S_5. && (2.17)
\end{aligned}$$

The variables $x_1, ..., x_n$ are the decision variables and every vector x that satisfies the constraints is called a feasible solution. A vector x that is a feasible solution and that minimizes the objective function is an optimal solution. In short the

[1] The observable peptides are also often called proteotypic peptides.

LP can be written as:

$$\min \quad \sum_i c_i \cdot x_i \tag{2.18}$$

$$\text{s.t.:} \quad Ax \geq b, \tag{2.19}$$

where A is a $m \times n$ matrix and the rows $a'_1, ..., a'_m$ build the constraints as $a'_i x \geq b_i$. A constraint of the form $a'_i x \leq b_i$ can be reformulated as $(-a_i)'x \geq -b_i$. Equality constraints $a'_i x = b_i$ can be reformulated using the two constraints $a'_i x \geq b_i$ and $a'_i x \leq b_i$.

During this thesis we are mainly dealing with maximization problems which can be easily converted into minimization problems as minimizing $c'x$ is equivalent to maximizing $-c'x$. Problems where the variables x_i are required to be integer are called *integer linear programming problems* (ILP). Problems with both integer and continuous variables are *mixed integer programming problems* (MIP).

In the following sections we introduce three combinatorial problems, which are part of Richard Karp's 21 problems, for which Karp showed the NP-completeness in 1972 [79].

2.5.2. Hitting Set

In Section 1.4, we gave a short introduction to the Hitting Set Problem. In the following, we are deriving a mathematical formulation.

An instance of the Hitting Set problem is given by a universe U and a family of sets $S = S_1, .., S_n$, with $S_i \subset U \; \forall i$. The goal is to find a subset P of U so that $|P|$ is minimal and $P \cap S_i \neq \emptyset \; \forall i$ [79]. The verbal formulation can be translated to the following ILP:

$$\min \quad \sum_j x_j \tag{2.20}$$

$$\text{s.t.:} \quad \forall_i : \quad \sum_{j \in S_i} \quad x_j \geq \quad 1 \tag{2.21}$$

$$\forall_j : \quad x_j \in \{0, 1\}. \tag{2.22}$$

Here, x_j is an indicator variable which is one, if circle j is part of the minimal set P and zero otherwise. In Figure 1.7, U is built by all circles, the subsets S_i are displayed via the rounded rectangles. A possible solution P is given by the purple circles.

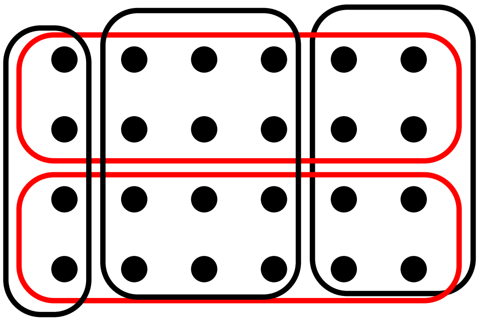


Figure 2.9.: Set-Covering Problem. Given a set of circles and a set of rectangles covering subsets of the circles, find a minimal number of rounded rectangles that covers all dots. The red rectangles are an optimal solution.

2.5.3. Set Cover

An instance of the *Set-Covering Problem* is given by a universe U and a family of sets $S = S_1, .., S_n$, with $S_i \subset U \ \forall i$ and every element u_j of U belongs to at least one set S_i. The goal is to find a set C of subsets of U, so that the number of subsets in C is minimal while C still covers all elements of U [79, 80]. An example is illustrated in the Figure 2.9.

The Set-Covering Problem can be formulated as ILP:

$$\min \quad \sum_i y_i \tag{2.23}$$

$$\text{s.t.:} \quad \forall_{j \in U} : \quad \sum_{i:j \in S_i} y_i \geq 1 \tag{2.24}$$

$$\forall_i : \quad y_i \in \{0, 1\}. \tag{2.25}$$

y_i is an indicator variable which is one, if set S_i is part of the minimal cover and zero otherwise.

The Set Cover and the Hitting Set Problem are equivalent and can be transformed into one another.

2.5.4. Knapsack

The Knapsack Problem is another well known combinatorial problem. We introduced a real life example in Section 1.3. In the following, we give a more technical introduction.

Mathematically speaking we have given a set of items $I = i_1, ..., i_n$. Each item has a weight w_i and a value v_i and an indicator variable x_i, which is 1 if item i is part of the solution and 0 otherwise. Now, the goal is to maximize the sum of the selected items' values while the sum of item weights does not exceed *cap*. The optimization problem looks as follows:

$$\max \quad \sum_i x_i \cdot v_i \tag{2.26}$$

$$\text{s.t.:} \quad \sum_i x_i \cdot w_i {<} cap \tag{2.27}$$

$$\forall_i : \quad x_i \quad {\in} \{0, 1\}. \tag{2.28}$$

Chapter 3 Related work

In a standard LC-MS/MS workflow especially when using ESI-MS the most frequent precursor ion selection strategy is data-dependent acquisition (DDA), where after each survey MS scan the highest signals are selected for further fragmentation [81, 82]. This precursor ion selection is incorporated into most of the machine vendor's software packages. As already pointed out in the motivation, DDA yields only a limited reproducibility in technical and biological replicates.

In this chapter, we present an overview of existing precursor ion selection strategies. These can be categorized in the following main classes:

- DDA, as a tool for discovery proteomics it requires no prior information about the analyzed sample and it can be easily applied as it is implemented as standard procedure in most mass spectrometers;
- Exclusion list approaches that prevent fragmentation of uninteresting or redundant signals;
- Directed MS/MS based on inclusion lists requires knowledge about the peptide signals, for instance
  - based on a map of detected LC-MS features,
  - based on interesting signals that show a difference in their abundance between samples,
  - or they target known proteins using their proteotypic peptides;
- Iterative procedures or real-time precursor ion selection that change the precursor ion selection during the MS/MS run.

Furthermore, in the last years data-independent acquisition was developed where no precursor ion selection is performed prior to fragmentation. In the following sections we present the approaches beside DDA more elaborately.

3.1. Exclusion lists

Peptides elute over a certain time from the LC system and thus occur principally in more than one MS scan. In consequence, with normal DDA, high abundant peptide signals are selected several times. Through a simple approach called dynamic exclusion (DEX) this redundancy can be circumvented by excluding the m/z-values of already fragmented precursors. Usually, this is done in conjunction with (absolute or relative) retention time windows. Additionally, exclusion lists can contain m/z-values of certain widespread contaminants like keratin or of internal standards used for calibration.

Exclusion lists are often used in conjunction with replicate analyses of a sample [82–87]. Here, the exclusion list is updated after each analysis and contains the fragmented signals or identified precursors of earlier analyses. This approach often leads to a higher number of unique peptide identifications in replicate runs [84] and an overall higher number of protein identifications than simple repetitions [87]. The additional peptides identified by using exclusion lists are often among the low abundant signals [82]. Rudomin et al. [82] could additionally observe an increased sequence coverage of the identified proteins. Yet, exclusion lists in conjunction with DDA still select only high abundant signals which is problematic with complex biological samples where the dynamic range often spans several orders of magnitude. Furthermore, Claassen et al. [88] showed, based on predictions, that after a certain number of repetitions only the number of false positive peptide discoveries increases while the number of true positives remains the same.

3.2. Directed MS/MS

A complementary concept to excluding uninteresting signals is directed MS/MS [89–92] where one is looking for specific signals of interest. One possibility for a directed precursor ion selection are inclusion lists that contain m/z-values (and often an RT window) of the peptides of interest. Usually, inclusion lists are static, meaning that they are fixed prior to MS/MS analysis. The interesting signals can be based on MS data from one or more LC-MS analyses that are used to determine the molecular mass profile of all features in the sample. This profile can then constitute the basis for precursor ion selection. This approach is typically used with LC-MALDI MS due to its offline nature where MS and MS/MS acquisition can be performed separately in time.

Gandhi et al. [93] used an inclusion list based strategy to reduce redundancy for 2D-LC-MALDI-MS/MS. Peptide signals were clustered according to their first dimension elution profile and the most promising fraction was chosen for fragmentation. This decision was based on the signal-to-noise ratio (SNR). This way the authors could identify the same number of unique peptides as DDA with a smaller set of precursors. Juhasz et al. [10] combined experimental depletion of high abundant proteins with 2D-LC-MALDI-MS/MS. They utilized an inclusion list of detected LC-MS features and combined it with the exclusion of unwanted signals. The authors applied this approach to monitor peptide abundance levels for cardiovascular disease markers.

There are also several studies where inclusion lists were applied to LC-ESI MS/MS: Different groups showed that inclusion lists created from a consensus map of the detectable LC-MS features can yield various improvements. Rinner et al. [90] used the so created inclusion lists for the study of protein interactions. Hoopmann et al. [94] and Schmidt and co-workers [92] showed that this approach can lead to a higher number of identified peptides, especially for precursors of low abundance, compared to DDA. Picotti et al. [91] showed that for tryptic digests of single protein samples the number of peptide identifications per protein can be drastically increased. Sandhu et al. [95] compared DDA, directed MS/MS using inclusion lists, and Multiple Reaction Monitoring (MRM). In their study, transcription factors (TF) and bovine serum albumin (BSA) were spiked in known concentration into a complex tryptic digest of lysated breast cancer cells in order to analyze the limits of detection for the different methods. Sandhu et al. could show that inclusion lists based on known peptides lower the required amount of spiked BSA or TFs significantly in order to identify the protein of interest when comparing to DDA. Jaffe et al. [96] used inclusion lists as a first step in biomarker detection. With their help long lists of biomarker candidates can be shortened to the peptides that are detectable in a specific setup. For these candidates MRM assays can then be developed for verification.

Hattan and Parker [97] proposed a precursor ion selection based on a consensus LC-MS map of several replicates. Additionally, the authors used statistical tests to detect significant differences in different sample groups. Their proposition was to target precursor ion selection specifically at sample differences and similarities. In this way, the efficiency of MS/MS acquisition in the context of information retrieval can be improved as less sample, time and effort is spent on uninformative signals. Neubert et al. [98] used the method of Hattan and Parker to detect differentially expressed proteins in E. coli with label-free LC-MALDI MS/MS.

Recently, Yan et al. [99] developed Index-ion Triggered Analysis (ITA) where for each targeted peptide, a heavy index peptide is synthesized which triggers the MS/MS of the light target ion independent of the light ion's abundance. Additionally, for each target peptide a reference peptide is synthesized which is used for quantification. This approach is more sensitive than inclusion list approaches especially for low abundant target peptides and it does not rely on a highly reproducible LC run. However, a clear drawback is the need of synthesizing two peptides per target peptide which makes it probably not suitable for high-throughput analyses.

In a recent study, Schmidt et al. [100] used an repetitive directed selection strategy with LC-ESI-MS/MS to monitor protein abundances at different cell states of a microorganism. Two initial DDA runs were used to create a map of detectable features. The detectable but yet unsequenced features were then inserted into inclusion lists. After these inclusion runs, additional inclusion lists were created based on proteotypic peptides for this organism observed in previous studies and on predictions. With the protein and peptide identifications achieved with this procedure a set of proteotypic peptides per protein was selected and together with a set of labeled peptides inserted into a new inclusion list. This allowed quantitative time course measurements of perturbed cells with a relatively small number of precursors.

3.3. Data-independent acquisition

A complementary approach for MS/MS is the so-called MS^E technology, concurrent peptide fragmentation or data-independent acquisition [101–106]. In MS^E each survey MS scan is usually followed by a fragmentation spectrum where all peptide ions are concurrently dissociated. Thus, practically no precursor ion selection is done. This results in highly complex fragment spectra. Algorithms for deconvolution of mixture spectra were developed that use LC elution profiles of precursor and product ions to construct MS/MS-like spectra for all simultaneously fragmented peptides [105, 107]. Blackburn et al. [106] compared MS^E to DDA and showed that MS^E can yield a higher protein sequence coverage especially for low abundance proteins. Geromanos et al. [108] argued that MS^E is more suitable for quantification than DDA as all precursor and product ions are recorded during the peptide's entire chromatographic elution leading to more comprehensive product ion spectra.

3.4. Iterative and real-time precursor ion selection

The presented approaches for inclusion and exclusion list generation are often applied in repetitive analyses where previously acquired LC-MS/MS data are used to guide the precursor ion selection of the current run. In contrast to that, with iterative precursor ion selection (IPS) not all tandem spectra are recorded at once. Rather acquisition is suspended after a certain number of MS/MS spectra. Then, information from identification results obtained so far can be used to guide the selection in the following iterations. This means that with IPS the same LC-MS data is used for the whole analysis, whereas with repetitive analysis replications are used with the associated drawbacks like limited reproducibility (see Section 1.2).

The advantage of an iterative exclusion of unwanted signals was shown by Scherl et al. [109] for protein digests fractionated on gels. The authors included m/z-values of tryptic peptides of already identified proteins into the DEX list, thus preventing fragmentation of signals pointing to already identified proteins.

Recently, Liu et al. [110] presented an iterative MS/MS acquisition (IMMA) tool. Similar to a study conducted for this thesis [17], Liu et al. exploited the offline nature of LC-MALDI MS/MS and changed the precursor ion selection during ongoing MS/MS acquisition. Unlike our approach, Liu et al. concentrated on excluding ions from the precursor list with different filters. First, a peptide fractional mass filter that classifies m/z features as peptides or non-peptides based on their excess to nominal mass ratio. This filter makes use of the observation that peptide masses are unevenly distributed and can be clustered into narrow equidistant regions separated by approximately 1 Da. [1] Besides, proteotypic peptides of previously identified proteins are set onto an exclusion list with predicted RTs and computed m/z-values. The proteotypicity prediction is used to increase the specificity of the exclusion.

Lately, real-time peptide identification was applied for targeted precursor ion selection with LC-ESI-MS/MS [111, 112]. Graumann et al. [111] incorporated a so-called “intelligent data acquisition” together with real-time database search into MaxQuant [113]. Their tool detects features or SILAC pairs while the corresponding peptide is eluting and triggers fragmentation of these on the fly. A real-time version of the search engine Andromeda [114] was developed and used for mass calibration during the measurement. Their work describes some proof-of-principle examples like resequencing of a peptide feature based on the intensity development of the eluting peptide. Bailey et al. [112] showed different applications of real-time peptide identification: the authors used RT predictions to create inclusion lists on the fly thereby targeting 30 times more peptides per RT window than with offline scheduling. In their study, Bailey et al. [112] also observed significant improvements of quantification results by resequencing the targeted peptide. Besides, Bailey and co-workers improved localization of PTMs by triggering an ETD MS/MS scan of peptides whose PTM could not be localized with HCD MS/MS.

[1] This pattern is also illustrated in Fig. 6.2.

Chapter 4 Sample preparation and data processing

In this chapter, we describe the samples used in the evaluation of our algorithms. Sample preparation is explained and how the resulting LC-MS/MS data were processed.

We used three samples of different complexity to evaluate the different approaches which are listed in Table 4.1. A protein standard sample containing 48 human proteins in equimolar concentrations provides a well-defined basis for the evaluation. The proteins are known, so we have a gold standard to work with. As pointed out in Section 1.2, biological samples have a high dynamic range of protein abundances. In order to investigate the influence of the high dynamic range on our algorithms we used two biological samples for evaluation, one of medium and one of high complexity.

In the following, we give a description of the sample preparation. This is followed by data processing and model training for PT and RT prediction. Model training is exemplarily evaluated on one of the samples (figures for the other samples are given in the supplement A.2).

4.1. Sample description

Sample 1 was the Universal Proteomics Standard (UPS1, Sigma-Aldrich), consisting of 5 pmol each of 48 human proteins. The protein standard was dissolved in 25 μL 50 mM NH_4HCO_3/10 mM nOGP. After adding 5 μL 25 mM DTT the sample was incubated for 30 min at 37°C. Then 5 μL 50 mM IAA were added and the mixture was again incubated for 30 min at 37°C. The sample was diluted by adding 85 μL H_2O. 2μL of trypsin (100 ng/μL) were added and the sample was incubated at 37°C over night. The digest was acidified and diluted by addition of 380 μL of 0.1% TFA and stored in 10μL aliquots, containing 100 fmol of each of the 48 proteins, at -20°C. We analyzed four technical replicates of UPS.

Sample 2 was the 50S ribosomal subunit, consisting of 33 different proteins, and isolated from *Escherichia coli* as described previously [115]. It was a gift from Dr. Fucini (Max Planck Institute for Molecular Genetics, Berlin). The sample

Table 4.1.: Sample overview

Name	Description
UPS	Universal protein standard consisting of 48 human protein in equimolar concentration.
50S	50S ribosomal subunit of *E. coli*, consisting of 33 proteins.
HEK293	Tryptic digest of cell lysate of HEK293 cells.

was subjected to tryptic digestion as previously described [116]. 6 μL sample, corresponding to 1 pmol 50S subunits, were used for each LC-MS analysis. We measured this sample in four replicates.

The third sample is a tryptic digest of the total proteome of 10,000 HEK293 cells. This sample was analyzed in the contest of the 13th Workshop for micro methods in protein chemistry in Martinsried. It was prepared and provided by the group of Prof. H. Meyer (Medical Proteome Center, Ruhr University Bochum, Germany). The peptide lyophilisate was dissolved in 20 μL 0.1% TFA.

4.2. LC-MS sample preparation

All samples except Sample 3 were analyzed on an 1100 Series Nanoflow LC system (Agilent Technologies, Waldbronn, Germany). The mobile phases were Buffer A: 1% acetonitrile and 0.05% TFA and Buffer B: 90% acetonitrile and 0.04% TFA. The samples were separated using a 100 min gradient. The Agilent 1100 fraction collector spotted fractions of LC-effluent onto MALDI sample plates from min 14 to 77 every 30 seconds. The gradient started with 100% Buffer A, after which the concentration of Buffer B was set to 3% after 5 min and increased to 15% after 8 min. Then Buffer B was linearly increased to 45% over 60 min. At min 73 Buffer B was set to 95% and held at 95% for 5 min.

Prior to HPLC analysis AnchorChip 800/384 targets (Bruker Daltonics, Bremen, Germany) were prepared with thin layer of CHCA matrix as previously described [116]. All mass spectra were acquired on a Bruker Ultraflex III MALDI TOF-TOF equipped with a 200 Hz solid state smartbeam laser. Positively charged ions of *m/z* 800-4000 were detected, for Sample 3 this window was extended to *m/z* 700-5000, and thousand single-shot spectra were accumulated at ten different positions. Monoisotopic peaks were determined using the algorithm SNAP, implemented in the FlexAnalysis 3.0 software (Bruker Daltonics). Except for Sample 3 all spectra were internally calibrated using two peptides present in the matrix solution (Angiotensin I 1296.6853 Da and ACTH (18-39) 2465.1989 Da). Monoisotopic peaks in successive spectra were combined to compounds and selected for MS/MS analysis using the software Warp-LC 1.1 (Bruker Daltonics).

Sample 3 was analyzed on an Easy-nanoLC (Bruker). Mobile phases were Buffer

A, consisting of 0.5% TFA, and Buffer B with 90% acetonitrile and 0.05% TFA. We used a 205 min gradient for the first ten minutes 98% Buffer A and 2% Buffer B. Afterwards, Buffer B was linearly increased to 35% over 120 min. Then, it was further increased to 70% over 60 min and finally it was increased to 100% over 10 min. Fractions were spotted from the 37th to the 165th min every 10 seconds, resulting in 768 spots on two targets. Half of the sample (10 μL) was injected.

4.3. Peptide identification

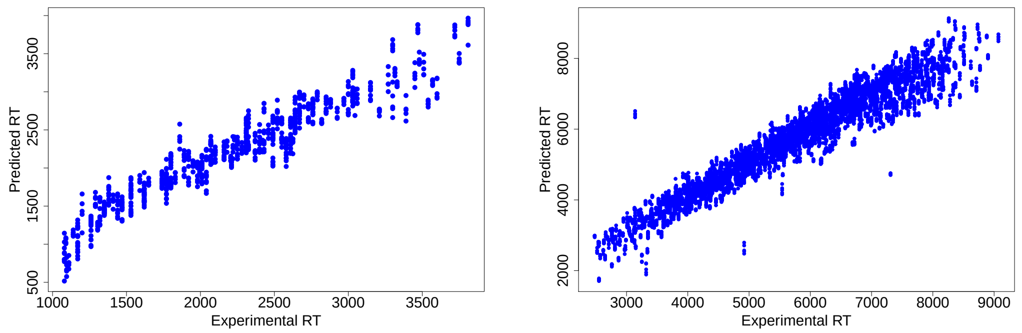
For peptide identification, we performed database searches using X!Tandem [37] (release CYCLONE (2010.12.01)) via XTandemAdapter from TOPP [60] as wrapper of the search engine. We searched the Swiss-Prot protein sequence database in Release 2011_08 with the taxonomy limited to E. coli for sample 2 and human for the other samples, unless otherwise stated. A combined database of a decoy and a normal version was used for searching. The other search settings were:

- 25 ppm precursor mass tolerance,
- 0.3 Da fragment mass tolerance,
- +1 as minimal and maximal precursor charge,
- carbamidomethylation as fixed modification (except for Sample 3),
- methionine and tryptophane oxidation as variable modification,
- 1 allowed missed cleavage and
- a tryptic cleavage site.

After the search, the peptide hits were annotated as target or decoy hits using TOPP's PeptideIndexer. Then, PEPs were computed using IDPosteriorErrorProbability. Finally, peptides were filtered to retain only the target hits. All tools were used in version 1.9. Afterwards the posterior error probabilities (PEP) were transformed into identification probabilities using P = 1 - PEP.

4.4. RT and detectability model training

Before we can apply our algorithms, we require certain information about every peptide in the underlying database. This includes the m/z, which can be easily computed for all peptides using the molecular masses of the amino acids, the RT and the detectability. Incorporating predicted RTs and detectabilities into our setup allows to reduce the risk of erroneously assigning a peptide in the database to an observed LC-MS feature. The RT limits the search space for matching features in the LC-MS map, the detectability limits the set of peptides



(a) UPS

(b) HEK293

Figure 4.1.: Experimental vs. predicted retention time for (a) the protein standard and (b) the HEK293 sample. The Pearson correlation is 0.94 (UPS) and 0.96 (HEK293).

to be considered. We used SVMs to predict both RT and detectability for our setup. Therefore, it was necessary to train models as explained in Sections 2.4.1 and 2.4.2.

The training set for the RT model consisted of peptides identified with a probability of at least 0.99. For samples measured in replicates, the training set consisted of merged IDs from all but one run. We performed a 10-fold cross validation to determine the best parameter set. In Figure 4.1 experimental and predicted RT are plotted for the UPS and HEK293 sample. We can see a high correlation of experimental and predicted RTs leading to Pearson's correlation coefficients of 0.94 and 0.96, respectively.

Two peptide sets are required for detectability model training, a positive one containing the proteotypic peptides and a negative set with unobserved or undetectable peptides. The positive set was composed of peptides identified with a PEP < 0.01 (for replicates merged from all but one run). We know the sample composition for UPS and the 50s ribosomal subunit, so these identifications were filtered for protein sequences contained in the UPS and 50s sample. In order to create the negative set of undetectable peptides, protein sequences of the positive set were assembled. Then, these were *in-silico* digested and filtered for the exclusion of all peptide hits found in any of the runs irrespective of the identification score. Furthermore, negative peptides that are substrings of identified peptides or that contain substrings of identified peptides are filtered out. After filtering we thus retrieve a set of peptides that belongs to the observed proteins but the peptides itself were not observed and can therefore serve as negative peptide set. Additionally, the negative peptide set was filtered for size, as in our setup we only observe peptides with an m/z between 700 Da and 5000 Da (complex data set) or between 780 Da and 3600 Da (all other data sets). Besides, the HEK293 data set was filtered for proteins with at least four peptide identifications to keep the negative sequence set at a reasonable size.

We used balanced data sets for model training. Thus, negative peptides were randomly drawn from the whole negative sequence set as the number of negative peptides exceeded the number of positive ones. We used a 10-fold cross-validation to learn the model parameters.

4.4.1. Evaluation of the detectability model

We validated the detectability models with a method proposed by Pfeifer [117]. As explained in section 2.4.2, the SVM learned which amino acids are important to distinguish detectable from undetectable peptides. When evaluating the model training we first show these important amino acids at the different positions of the peptide termini in a heatmap. Then, we compare this heatmap with a Two Sample Logo (TSL) [118] which determines enriched and depleted AAs of the positive sequence set using a statistical test. Enrichment or depletion in this context means that an AA is over- or underrepresented in the positive set. The TSL requires two multiple sequence alignments, one of the positive and one of the negative sequences. Hence, peptide sequences were aligned on their C-Terminus as the peptides differ in length. In this section, we focus on the evaluation of the UPS sample, complete figures for the remaining samples can be found in the supplement A.2.

We applied a POBK which considers both peptide ends simultaneously, thus a strong signal in the heatmap at position i corresponds to peptide positions i and $n - i + 1$, where n is the peptide length. The SVM showed a strong depletion of arginine and lysine at the borders (Figure 4.2) what is confirmed by the TSL (Figure 4.3). Another strong signal in the heatmap is the enrichment of the aromatic AAs phenylalanine and tyrosine which is also visible in the TSL. The enrichment of aromatic AAs for MALDI experiments was also detected by Pfeifer [117] and is confirmed by the literature [119]. The strong depletion of the same AAs at the high positions in the heatmap is interesting as most peptides are shorter than 22 AAs and can not produce a signal at these positions. However, a bias to longer negative sequences was observed in the training data. The longest positive peptide consists of 20 AAs, the longest negative of 34 AAs, which might explain this phenomenon.

Finally, we compared the differences in peptide probability and predicted detectability (Figure 4.4). The predicted detectability is mostly smaller than the peptide probability but the histogram shows that the detectability can indeed be a predictor for the ability of a peptide to be identified. The mean difference is 0.05 with a standard deviation of 0.37, the median is 0.18.

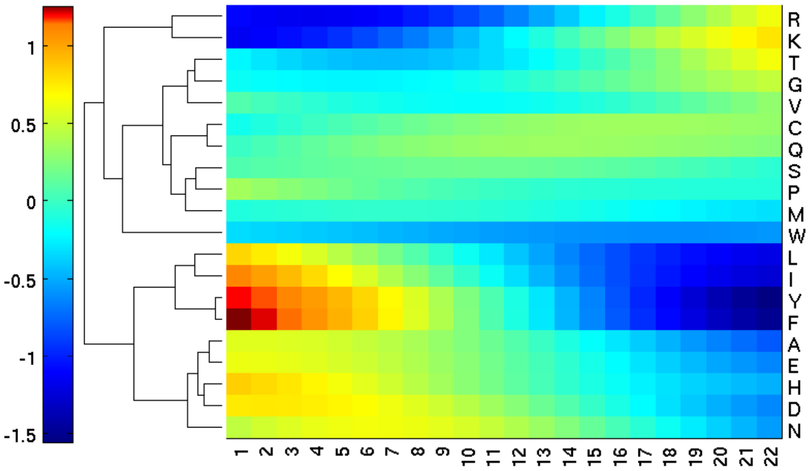


Figure 4.2.: Visualization of POBK for UPS. Produced with MATLAB scripts from Nico Pfeifer[117]. The plot shows the signals for both termini together, hence position i corresponds to AAs at position i and $n - i + 1$ (where n refers to the peptide length).

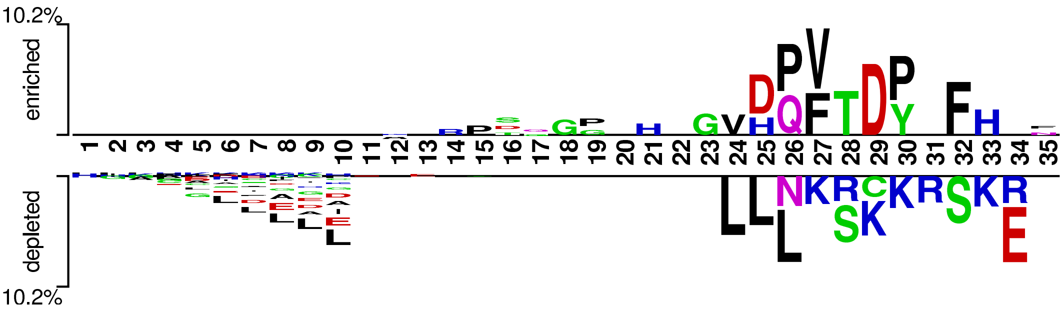


Figure 4.3.: Two Sample logo [118] for the high-scoring peptide identifications and the unobserved peptide sequences of the protein standard. Enriched AAs are shown at the top, depleted AAs at the bottom. Sequences were aligned at their C-Terminus and the position is given with respect to the longest peptide.

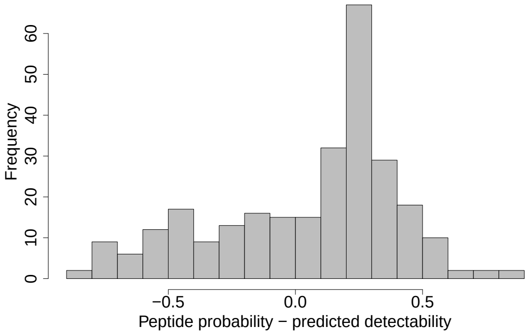


Figure 4.4.: Histogram of the difference between peptide probability and predicted detectability for UPS.

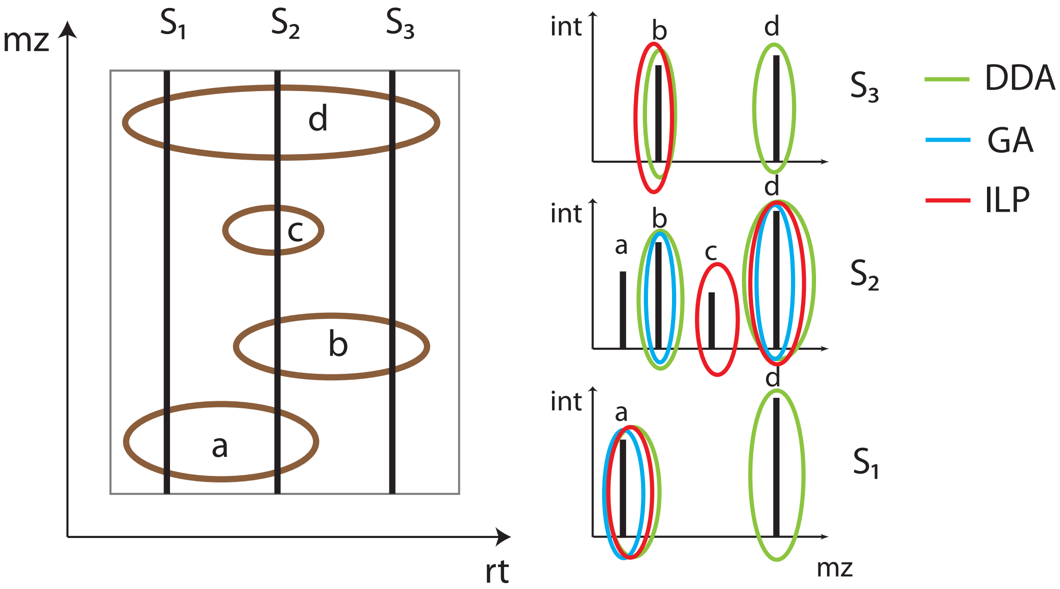
Chapter 5 Inclusion list creation as optimization problem

Inclusion lists are widely used for directed LC-MS/MS analyses as pointed out in Chapter 3. Depending on the aim of a study, several approaches are conceivable. In this chapter we introduce two strategies. First, given a survey MS feature map, e.g., as obtained from a first LC-MS run, we construct an inclusion list that maximizes the number of selected precursors. Thereupon, we develop an inclusion list solely based on protein sequences of interest in the sample to be analyzed. In both approaches we are interested in the optimal set of precursors, thus we develop an objective function and formulate the inclusion list creation as linear program (LP).

5.1. Inclusion lists for a given feature map

Assume we have recorded an LC-MS feature map, e.g., as is typically the case for LC-MALDI analyses due to the decoupled steps of LC and MS. Standard data-dependent precursor selection (DDA) chooses the highest signals in each spectrum, even if this means selecting the same feature again and again at different retention times. A more sophisticated selection would account for the 3D nature of the LC-MS feature and contains each feature only once, ideally at the RT with its maximal signal intensity. However, such a greedy approach (GA) might lead in total to a lower number of selected precursors than a global strategy as shown in a mock example in Figure 5.1. Here, a frequently occurring problem is that feature maxima are not equally distributed over the spectra. In spectra crowded with feature maxima, the MALDI sample may be depleted before MS/MS spectra of all selected precursors can be recorded. Additionally, in crowded spectra there is also an increased risk of occurrence of features with m/z-values too close to permit clean isolation of one precursor for MS/MS.

In the following section, we develop a formulation of the feature based inclusion list creation as optimization problem.



(a) (b)

Figure 5.1.: Illustration of precursor ion selection strategies. (a) LC-MS map of four features. (b) MS spectral view of the map. The colored markers show the selected precursors for each of the strategies, green with DDA, blue with GA and red with the ILP. Assuming a limited number of precursors per spectrum, here 2, feature c is never chosen by DDA and with GA again only features a, b and d are selected while in spectrum S_3 no MS/MS spectrum is acquired. Only ILP allows to select all features at once.

Table 5.1.: Variables and constants used in the LP formulations throughout this chapter.

Variable	Explanation
$x_{j,s}$	Indicator variable, 1 if feature j is selected in spectrum s, 0 otherwise
x_j	Indicator variable, 1 if feature j is part of the solution, 0 otherwise
$int_{j,s}$	Normalized signal intensity of feature j in spectrum s
cap_s	Maximal number of MS/MS precursors in spectrum s
h	Maximal number of times a feature is selected as precursor
dp_i	Detectability of protein i
z_i	$-log(1-dp_i)$, higher values reflect a better detectability
d_k	Detectability of peptide k
$a_{i,k}$	Indicator variable, 1 if peptide k is part of protein i, 0 otherwise
ws	RT window size
t_p	Predicted RT
max_list_size	Maximal number of elements in inclusion list
p_k	Probability that peptide k was identified correctly
c	Minimal protein probability to declare a protein identified

5.1.1. Problem formulation

Given a set of detected LC-MS features, our goal is to select a maximal number of these as precursors for fragmentation. Two constraints have to be fulfilled: first, for each spectrum the maximal possible number of precursors, also referred to as spot capacity, may not be exceeded. Second, the number of times a feature is selected as precursor is limited by a specified number h. This problem is related to the *Knapsack problem*, as pointed out in Section 1.3. However, now we are dealing with features potentially spanning more than one fraction. Our goal is to make a global precursor ion selection, and not a separate selection for each fraction.

For each feature we have a set of indicator variables $x_{j,s}$ that are 1 if feature j is selected in spectrum s as precursor and 0 otherwise. The x-variables are weighted by $int_{j,s}$ which corresponds to the intensity of feature j in spectrum s normalized by the maximal intensity of feature j in any spectrum (see Table 5.1 for an overview on ILP variables and constants used throughout this chapter.). This way, all features have normalized intensity values between 0 and 1, thus high intensity features are not favored over low intensity ones. Yet, for each feature, a spectrum with higher signal intensity is more likely to be chosen than a lower intensity spectrum. Absolute feature intensities can be considered instead of normalized intensities as well. However, in this case the sum of signal intensities of the precursors is maximized and not the number of precursors. Our constraints are that each feature must not lead to more than h precursors and that each RT bin has at most *cap* precursors. The ILP formulation looks as follows:

$$\max \sum_{j,s} x_{j,s} \cdot int_{j,s} \tag{5.1}$$

$$\text{s.t.:} \quad \forall_s : \sum_{j} x_{j,s} \leq cap_s \tag{5.2}$$

$$\forall_j : \sum_{s} x_{j,s} \leq h. \tag{5.3}$$

Inequation 5.2 ensures that the maximal number of selected precursors, cap_s, for spectrum s is not exceeded. Due to Inequation 5.3 each feature will only be selected in h spectra or less.

In our implementation, we solve the ILP formulation using the GNU Linear Programming Kit (GLPK, `www.gnu.org/software/glpk/`). The solution provides values for all $x_{j,s}$ and all features j where $x_{j,s} = 1$ are part of the final inclusion list. Due to Constraint 5.3, $x_{j,s}$ can only be 1 for at most h spectra s for each precursor j. In our standard settings we set $h = 1$, thus each precursor is scheduled in a specified fraction.

5.1.2. Results

Evaluation workflow

We want to evaluate a variety of settings for inclusion list creation, so a simulation study is best suited for this purpose. However, the spectra themselves are not simulated, only the precursor ion selection. This means that an LC-MS sample was exhaustively measured including all possible MS/MS spectra. Afterwards, different settings were applied for the inclusion list creation. The evaluation workflow is illustrated in Figure 5.2.

In the evaluation, inclusion lists were mapped onto observed LC-MS feature maps. If a feature from the inclusion list overlaps with an observed feature we assumed that the inclusion list feature can generate the same MS/MS spectrum as the observed feature. This is a strong assumption, as it also implies that for a given feature the fragmentation works with (almost) equivalent quality in all fractions, that it occurs in. However, as the reproducibility even of “simple” technical replicates is limited [6], this approach is the only possibility to differentiate real performance differences from differences resulting from replication issues.

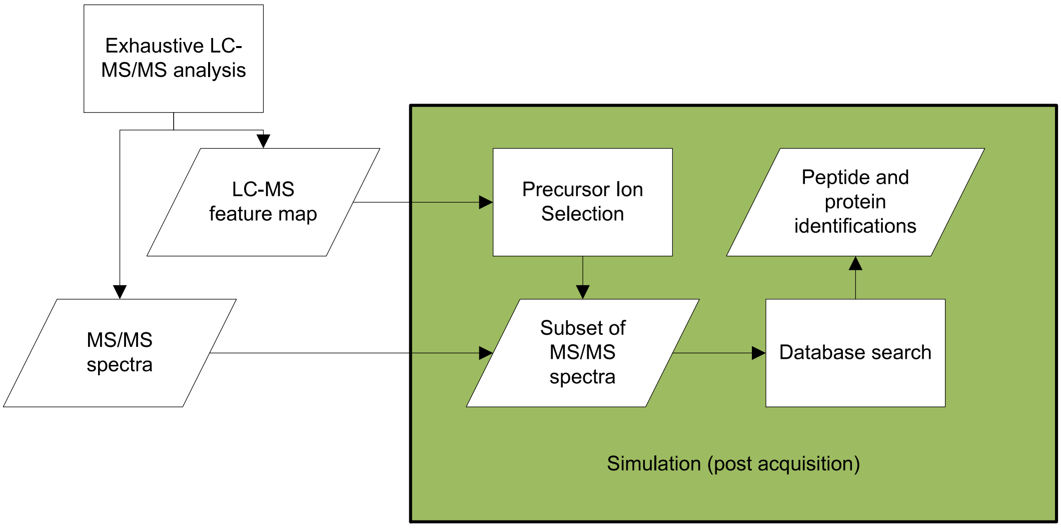
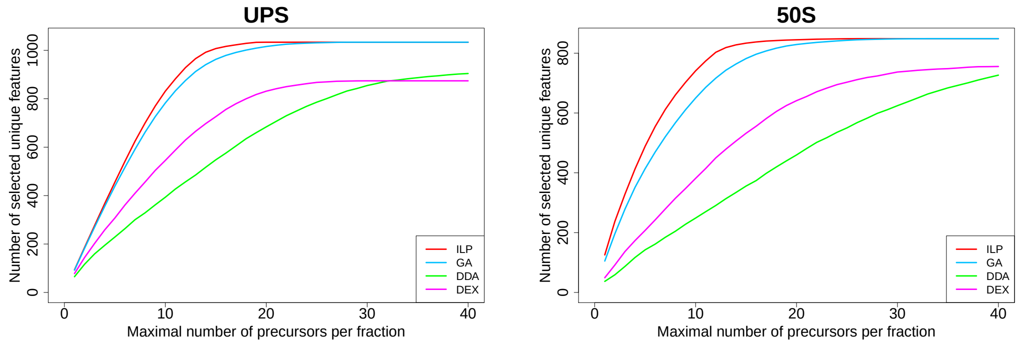

Exhaustive LC-MS/MS analysis
LC-MS feature map
MS/MS spectra
Precursor Ion Selection
Subset of MS/MS spectra
Peptide and protein identifications
Database search
Simulation (post acquisition)

Figure 5.2.: Evaluation workflow. First, the samples are analyzed by extensive LC-MS/MS, resulting in an LC-MS feature map and a number of MS/MS spectra. These build the data pool for all evaluation experiments that simulate precursor ion selection upon the data.

Algorithm evaluation

We evaluated four different strategies, namely GA, DDA and ILP that were illustrated in Figure 5.1 and DDA with dynamic exclusion of each scheduled precursor for the following two fractions enabled (DEX). We applied the selection strategies to the UPS, the 50S and the HEK293 sample. The maximal number of precursors per RT bin varied from 1 to 40, leading to inclusion lists of increasing size for each approach. For each strategy we counted the number of selected unique features to ensure that features which are selected more than once as precursor are considered only once. Figure 5.3 shows the results for UPS and 50S. As expected, ILP and GA, the two methods that make use of the feature information, clearly outperform DDA and DEX. ILP is also considerably better than GA: with about 18-20 precursors per RT bin all possible features can be selected as precursors while GA requires around 25 precursor per RT bin to do so. In turn, DDA and DEX do not allow to select all features present in the data set within the limit of 40 precursors per RT bin. Although the toy example in Figure 5.1 appears to be fictitious, the results show that there is a clear performance difference between ILP and GA. Especially for the biological relevant 50S sample this difference is significant.

In Figure 5.4 we can see the results for the complex HEK293 sample. Here, the difference of DDA and DEX compared to GA and ILP is even more significant than in the previous example. Only less than half of the LC-MS features are selected for fragmentation. Interestingly, GA and ILP perform similar up to a capacity of fifteen precursors per fraction, where ILP starts to perform better. At the maximal capacities of 20 and 25 GA selects around 400 and 650 precursors



(a) (b)

Figure 5.3.: Evaluation of feature based selection. For four different strategies the number of selected LC-MS features (each features counted once, even if selected several times) is shown against the number of maximal precursors per fraction for (a) the UPS sample and (b) the 50S sample. The results with the ILP are in red, for GA in blue, for DDA in green, and for DEX in magenta.

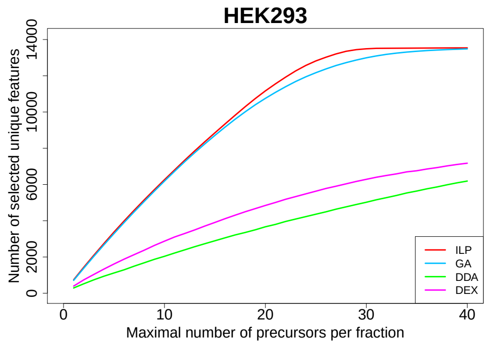


Figure 5.4.: Evaluation of feature based selection for the HEK293 sample. For four different strategies the number of selected LC-MS features (each features counted once, even if selected several times), is shown against the number of maximal precursors per fraction using the HEK293 data set. The ILP results are given in red, DDA in green, DEX in magenta and GA in blue.

less than the ILP. At the capacity limit of 40 none of the strategies selects all of the 13,546 features. The GA selection consists of 13,484 features while ILP selects 13,539 features. Hence, for all evaluated samples in all tested settings the ILP yields the maximal number of scheduled features.

Figure 5.5 (a) shows the number of LC-MS feature maxima in each RT bin for the HEK293 sample. Clearly, there are many RT fractions where the number of feature maxima exceeds 20, which is a realistic spot capacity in our setup. The histogram in Figure 5.5 (b) of the number of fractions with a given number of feature maxima gives a brief overview about the number of spectra exceeding a certain capacity.

As next step, we want to consider the run times. The CPU times for solving the

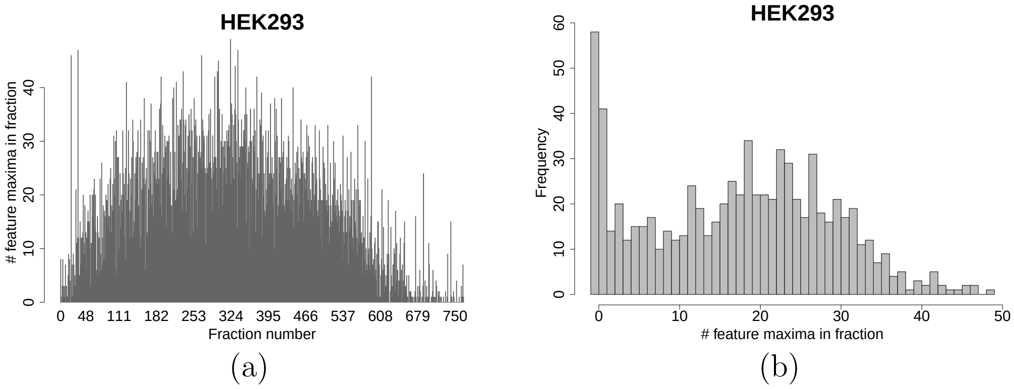


Figure 5.5.: Distribution of feature maxima for HEK293 sample. (a) shows a histogram of feature maxima per fraction. There are many fractions where the number of features exceeds 20, which is a realistic spot capacity in our setup. (b) shows how many fractions exceed a given spot capacity.

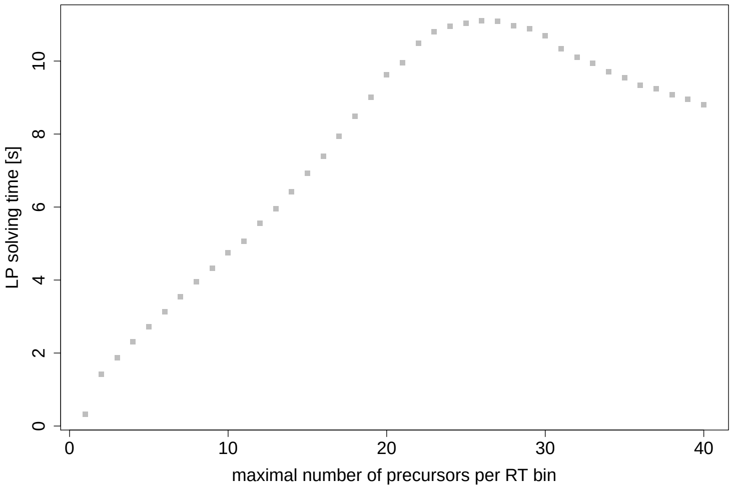


Figure 5.6.: Times for solving the feature-based ILP, measured by evaluating the HEK293 sample.

ILP with the solver from GNU Linear Programming Kit (GLPK) were measured in 15 experiments on an Intel Xeon X5550 with 2.67 GHz. In each of the experiments the maximal RT bin capacity ran from 1 to 40 as shown in the previous figures. For UPS, the CPU times for solving the ILP varied between 0.04 and 0.05 seconds, no dependency on the parameter settings has been observed. For the 50S data set the CPU time were below 0.01 s. Whereas for the HEK293 data the solving time clearly increased with a higher number of allowed precursors per RT bin up to 27 allowed precursors per fraction where $\approx$ 11 s are needed for the solution (Figure 5.6). Interestingly, for RT bin capacities higher than 27 the CPU times start decreasing again down to $\approx$ 8.8 s. A possible reason for this decrease is that the number of conflicts is smaller with a higher bin capacity as there less spectra remain that exceed their capacity than with a smaller limit. Another interesting observation is that maximal running time coincides with the beginning of the plateau in the number of protein identifications (see Figure 5.4). In summary, the times for solving the ILP are acceptable for all of the tested samples.

5.2. Inclusion lists for a given list of protein sequences

There are many experimental setups where researchers are not interested in maximizing the number of identified features, but want to observe a defined set of proteins under various conditions. This can also be done using inclusion lists, even without previous LC-MS runs of each protein set where the LC-MS signature of the sample is determined. Thus, we are now interested in optimizing the selection given a set of proteins of interest, but without prior knowledge of the LC-MS data. Ideally, we want to find a set of precursors such that each protein of interest is sufficiently characterized. We explain in Section 5.2.1 what this exactly means and how we compute this. As we have no previously acquired LC-MS data to base our precursor selection on, we have to predict LC-MS features as explained in the next paragraph.

Figure 5.7 shows the three layers of the problem: the highest layer presents the proteins of interest. Using their sequences, an *in silico* digestion leads to a set of tryptic peptide sequences. As shown in section 4.4 it is possible to reliably predict the RT and the detectability of a peptide given only its sequence if well trained models for the used experimental setup exist. After the prediction, we retrieve a set of candidate features. Now, we use an LP formulation to select a subset and to define an RT window for each feature.

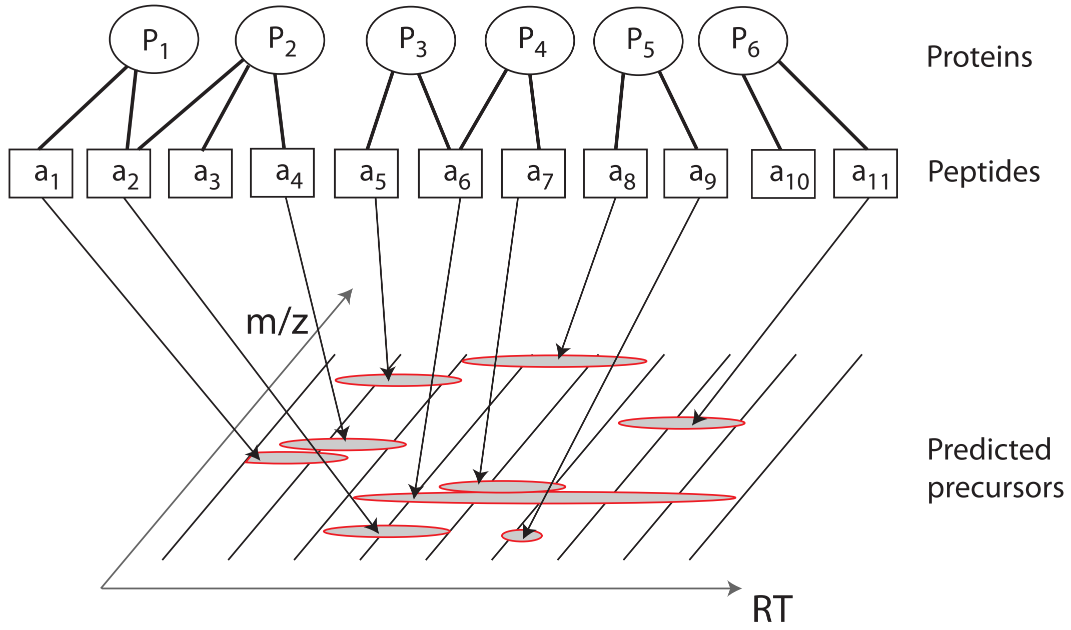


Figure 5.7.: The protein sequence based ILP inclusion list creation. Given a set of protein sequences P_1 to P_6 we can calculate the tryptic peptides a_1 to a_{11}. For all peptides we can calculate their m/z-values, predict their RT and whether they are detectable in a given LC-MS setup. In our example peptides a_3 and a_{10} are not detectable. The goal is to select a set of features that yields the best protein detectability.

5.2.1. Protein detectabilities

First, we need to find a measure to determine when a protein is sufficiently characterized. In Section 2.3.1 we dealt with different methods for protein identification. Here, we want to use a probabilistic formulation similar to the basic formula used in ProteinProphet [70].

The probability that protein i is identified correctly (in the following shortly called protein probability) can be computed via the probabilities of the corresponding peptides to be identified incorrectly, as shown in Equation 2.11 in Section 2.3.1. Accordingly, we can calculate the protein probability as probability that at least one of the peptides is identified correctly, see Figure 2.8 for an example.

However, in our case we do not have peptide probabilities. We use peptide detectabilities as analogies as they represent the likelihood that a peptide is detectable and identifiable in a given experimental setup. Thus, we can define a protein detectability of protein i as:

$$dp_i = 1 - \prod_k (1 - a_{i,k} x_k d_k), \tag{5.4}$$

where $a_{i,k}$ is an indicator term which equals 1 if peptide k is part of protein i and 0 otherwise. d_k is the detectability of peptide k. Additionally, we have an indicator variable x_k which is 1 if peptide k is part of the solution and 0 otherwise.

Finally, we want to formulate a problem with linear constraints, thus we need to reformulate the product term using the logarithm:

$$1 - dp_i = \prod_k (1 - a_{i,k} x_k d_k) \tag{5.5}$$

$$\Rightarrow \log(1 - dp_i) = \sum_k \log(1 - a_{i,k} x_k d_k) \tag{5.6}$$

$$\Rightarrow \log(1 - dp_i) = \sum_k x_k \cdot \log(1 - a_{i,k} d_k). \tag{5.7}$$

The last conversion is valid as x_k can only have the values 0 or 1. If it is 0, in both equations 5.6 and 5.7 we add 0 and if it is 1, we add $\log(1 - a_{i,k}d_k)$ in equations.

In the following section we use the protein detectability calculation in our formulation of the protein sequence-based precursor ion selection as optimization problem.

5.2.2. Problem formulation

In Section 1.4 we introduced an approach for a protein-based precursor ion selection using the *Hitting Set Problem.* This means we select a minimal set of peptides that covers the whole protein set. This approach has two problems in practice. First, by construction, it favors shared peptides over peptides that are unique for each protein as the number of selected peptides is minimized. This can be circumvented by maximizing the number of proteins and penalizing for the number of selected peptides. This way we retrieve a minimal peptide set covering a maximum number of proteins. The second point is that we cannot select peptides directly, as not all theoretical tryptic peptides are observed and identified in practice. As explained before we use the detectability to account for that. Altogether, this means we want to find a set of peptides, so that the sum of protein detectabilities is maximal, the inclusion list does not contain more than *max_list_size* precursors in total and each RT bin has at most *cap* precursors. This yields the following ILP formulation:

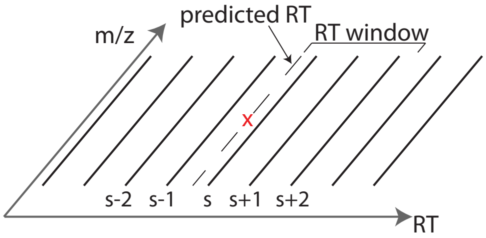


Figure 5.8.: RT window constraint. The predicted RT of a peptide is indicated by the dashed line. The solid lines depict the RTs of the survey MS spectra. The nearest spectra to the predicted RT has index s. The RT window shows how many spectra “left” and “right” of spectrum s are included in the ILP formulation.

$$max \sum_i z_i \tag{5.8}$$

$$s.t.: \quad \forall_s: \quad \sum_k x_{k,s} \leq cap_s \tag{5.9}$$

$$\forall_{k,s}: \quad x_{k,s} \leq x_k \tag{5.10}$$

$$\sum_k x_k \leq max\_list\_size \tag{5.11}$$

$$\forall_k: \quad \sum_{s \notin [t_p - ws, t_p + ws]} x_{k,s} = 0 \tag{5.12}$$

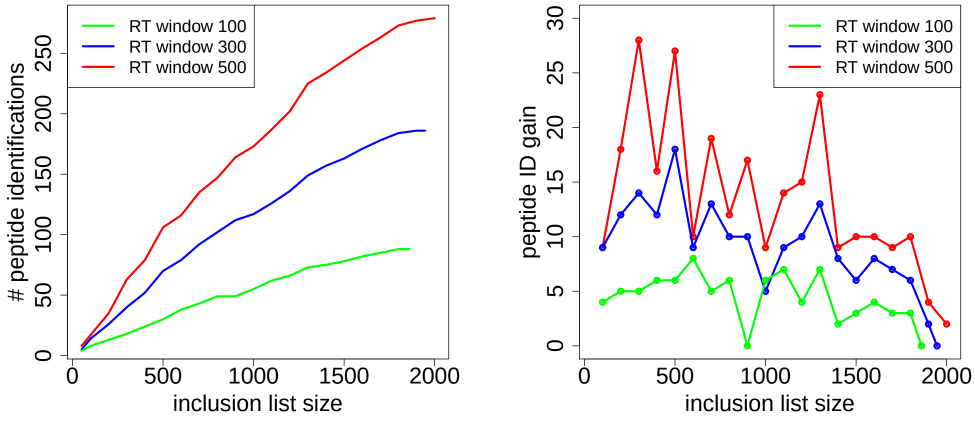
$$\forall_i: \quad z_i = -\sum_{k,s} x_{k,s} \cdot \log(1 - a_{i,k} d_k) \tag{5.13}$$

$$\forall_{k,s}: \quad x_{k,s}, x_k \in \{0,1\}. \tag{5.14}$$

z_i is depending on the protein detectability dp_i as explained in the previous section. From $dp_i \in [0,1]$ follows that $log(1 - dp_i) \leq 0$. For high protein detectabilities $log(1 - dp_i)$ is approaching $-\infty$. Thus, by maximizing the sum of z_i, the additive inverse of $log(1 - dp_i)$, we maximize the sum of protein detectabilities.

Constraint 5.12 ensures that only those spectra s can be chosen for peptide k that lie in an RT window of size ws around the predicted RT t_p, hence that lie in the interval $[t_p - ws, t_p + ws]$, see Figure 5.8 for an illustration.

By solving the ILP formulation we receive a set of variables $x_{k,s} = 1$ that build the inclusion list. In this setup, we provide RT windows for each precursor in the inclusion list. Thus, for each peptide k there can be multiple $x_{k,s} = 1$.



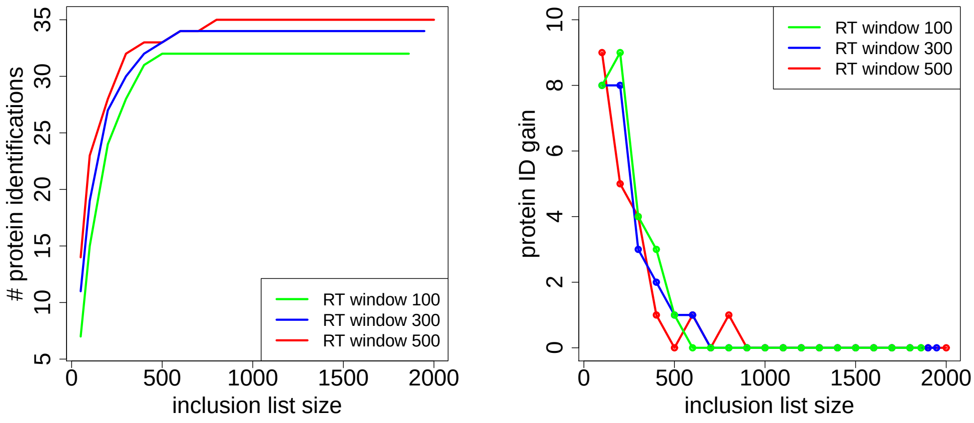
(a) (b)

Figure 5.9.: Peptide IDs obtained with protein sequence-based LP. Inclusion list creation via a protein based ILP formulation for the protein standard, (a) the inclusion list size vs. the number of peptide identifications, (b) the gain in peptide identifications with the increasing inclusion list size. The gain is the number of additional peptide IDs obtained with the last size limit increase. The RT window varied from 100 to 500.

5.2.3. Results

The inclusion list creation with protein sequence-based ILP was evaluated on the protein standard. We trained RT and PT models as described in section 4.4. The training set consisted of peptide identifications from three LC-MS/MS experiments. The fourth LC-MS/MS run that has not been considered for model training was used in the evaluation. Inclusion lists were created using the ILP formulation. During the evaluation, we compared the precursors of the inclusion list with the actually observed features. If an observed feature overlapped with a predicted precursor, the peptide annotation of this feature was assigned to the predicted feature. This way, we evaluated the number of peptide and protein identifications an inclusion list would deliver. In this context, a protein was declared as identified if the protein probability calculated using Equation 2.11 is at least 0.99.

Figure 5.9 (a) shows the absolute number of peptide identifications against the inclusion list size. We used RT window sizes of 100, 300 and 500 seconds, illustrated in green, blue and red. The figure shows that the increase in the number of peptide identifications correlates with the inclusion list size. Interestingly, this effect depends strongly on the RT window size. Using a smaller window clearly reduces the gain of the increase in inclusion list size. Figure 5.9 (b) explicitly emphasizes this effect. Here, we show the additional number of obtained peptide IDs for each stepwise increase of the inclusion list size. The highest gain can always be achieved for an RT window of 500 s.



(a) (b)

Figure 5.10.: Protein IDs obtained with protein sequence-based LP. Inclusion list creation via a protein based ILP formulation for the protein standard, (a) the inclusion list size vs. the number of protein identifications, (b) the gain in protein identifications with the increasing inclusion list size. The RT window was varied from 100 to 500.

As we are evaluating the protein based inclusion list creation, the more important aspect is the number of protein identifications. Figure 5.10 (a) shows the number of protein identifications against the maximal inclusion list size. Again, we assessed the performance of the inclusion list using different RT window sizes, 100, 300 and 500 s. For all RT window sizes we see that the maximal number of protein identifications is achieved with about 900 precursors. A further increase in inclusion list size does not yield an improvement. The absolute number of identified proteins decreases with a decreasing RT window size. An inclusion list with around 500 precursor already yields 32 or 33 protein identifications for all window sizes. Figure 5.10 (b) shows the gain in protein identifications with increasing the inclusion list size. An interesting aspect is that the number of protein identifications is partly higher than the number of peptide IDs (using the same threshold). This is due to the computation of the protein probability where several medium quality peptide IDs, for themselves not significant, can be added up to a significant protein ID.

The effect of the RT window size is shown in Figure 5.11. We can see that the number of identified peptides increases almost linear with the RT window. The RT range of the underlying experiment was only 2880 s, thus an RT window size of 1000 seconds covers more than two third of the whole experiment rendering the RT prediction somewhat irrelevant. We compared two settings in Figure 5.11 (b): an inclusion list size containing maximally 1000 precursors (green) and an inclusion list not limited in its size (red). Both settings yield very similar results. The plot shows that already an RT window of 150 seconds yields 34 identified proteins. Any further increase of the RT window only leads to 1 or 2 more protein

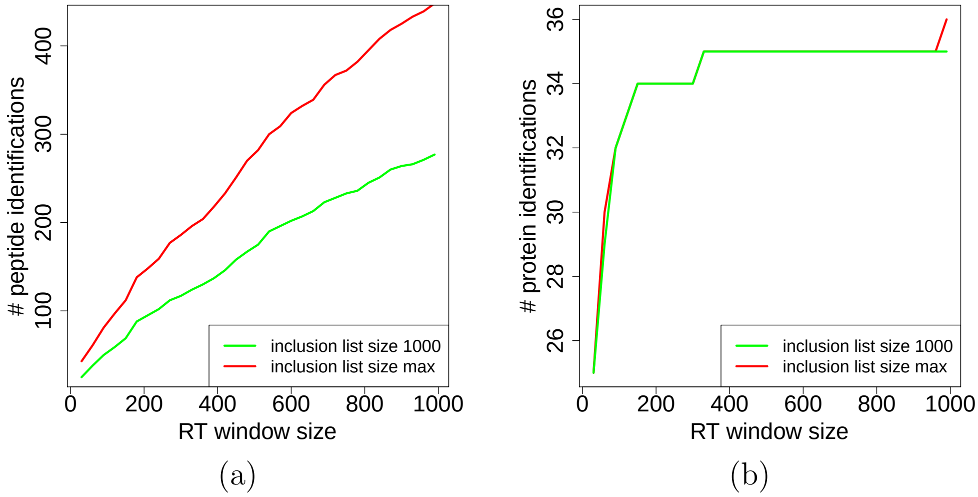


Figure 5.11.: Effect of RT window size. Inclusion list creation via a protein based ILP formulation for the protein standard, (a) the RT window size vs. the number of peptide identifications, (b) the RT window size vs. the number of protein identifications. The inclusion list size was either unlimited or set to 1000.

identifications. These results justify smaller RT windows.

Next, we wanted to determine the value of a good detectability prediction. We compared the results obtained with our trained model with inclusion lists created with either a constant detectability set to 1 for all peptides or with a randomly assigned detectability (Figure 5.12). Both inclusion lists perform considerably worse than the one obtained with the trained model. In the end, all settings lead to the same number of protein identifications, yet the required number of precursors is very different. Especially, with complex samples where the number of theoretical tryptic peptides clearly outranges the number of possible precursors the usage of the detectability might make a clear difference.

So far, we only considered the well defined UPS sample for evaluation, now we apply the LP-based inclusion list to a biological relevant sample, the 50S ribosomal subunit of *E. coli*. The proteins building the ribosomal subunit are known which is a prerequisite of the protein sequence-based selection. However, in contrast to the UPS sample the proteins are not equimolar and thus represent a more realistic setting. The number of observed features was smaller than with the UPS sample, so already with around 600 precursors a maximal number of proteins is identified (Figure 5.13) in all tested settings. Now, the performance of small RT windows of 100 s is considerably worse than the one of larger windows. However, again RT windows of 200 s yield a maximal number of identified proteins.

We measured the running times for solving the ILP again on a Xeon X5550 (Figure 5.14). The solving times are clearly increasing with larger RT windows. Another time-relevant factor is the maximal inclusion list size: a smaller limit

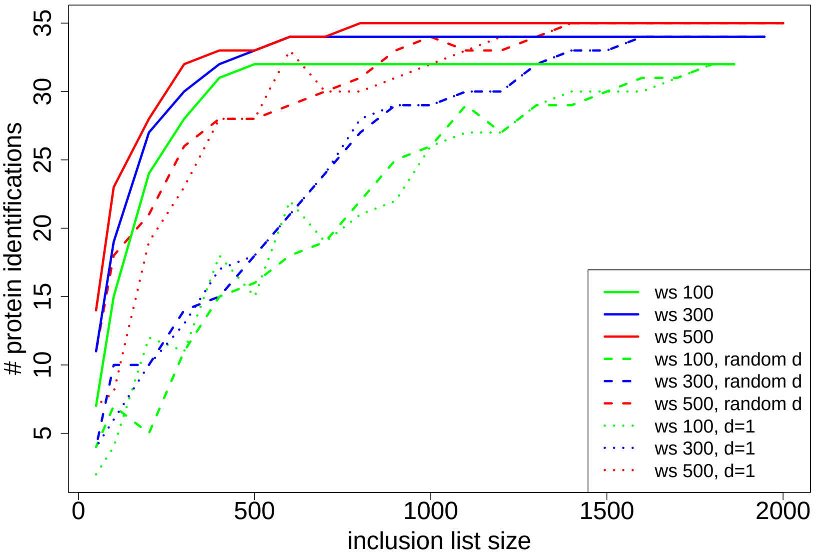
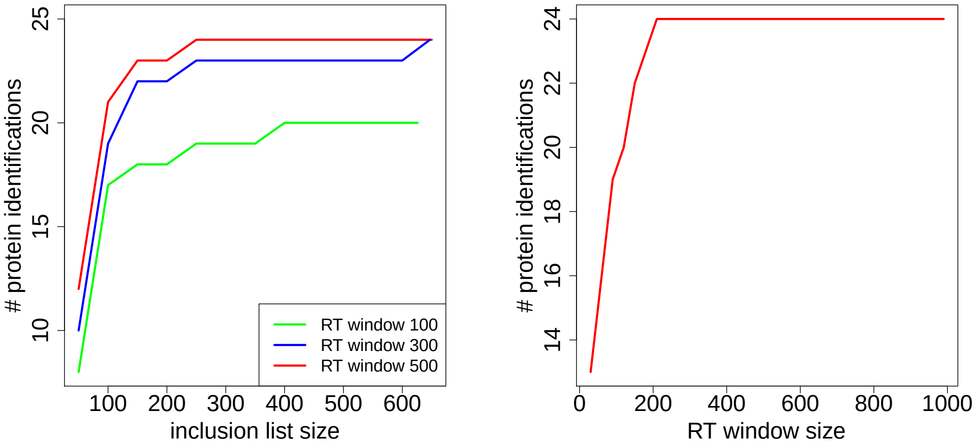


Figure 5.12.: Results with random or constant detectability.



(a) (b)

Figure 5.13.: Inclusion list creation using a protein sequence-based ILP formulation for the 50S sample. (a) The number of protein identifications against the inclusion list size, (b) the RT window size vs. the number of protein identifications.

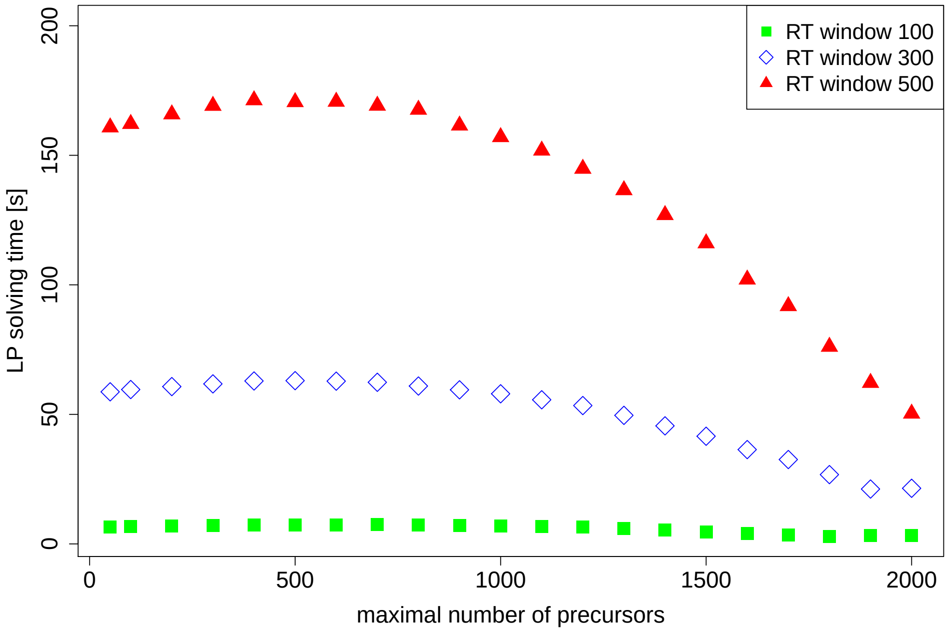


Figure 5.14.: CPU times for solving the protein sequence-based ILP, measured by evaluating the UPS sample.

implies more conflicts and thus requires more time to solve. However, again all times are feasible.

These results show that our ILP formulation delivers very efficient inclusion lists solely based on predictions. It enables a direct control of the amount of "protein confidence" by optimizing the protein detectability. This way, we retrieve an optimal precursor set for each parameter setting. The ILP formulation can be easily adapted to consider not all peptides of a protein (weighted by their predicted detectability), but a specific set of predefined peptides that can be used for quantification. For instance, Schmidt et al. [100] used such a set of around 5,000 peptides belonging to 1,680 proteins of a human pathogen to monitor their expression levels at 25 different states.

Chapter 6 Iterative precursor ion selection

In the last chapter, we described different inclusion list problems and how to solve them with ILPs. However, especially with MALDI-MS/MS, it is possible to change the inclusion list during MS/MS acquisition as the sample is “frozen in time”. We are able to perform analyses on the MS/MS data we got so far and let the results influence the next precursor ion selection. So in this chapter, we introduce *iterative precursor ion selection* (IPS).

In each iteration a specified number of MS/MS spectra is recorded and a database search is performed in order to identify the peptide signals. Afterwards, peptides are matched onto proteins. Here, we distinguish between already identified proteins which exceed a given probability c and protein candidate hits with a probability $< c$. IPS has two goals: on the one hand, to find more peptide hits for protein candidates so that they exceed the significance threshold with one of the next selected precursors. On the other hand, we want to identify as many proteins as possible, hence sequencing peptides from already identified proteins yields only redundant information and is uninteresting. Thus, these signals shall be excluded.

In the next paragraph, we briefly explain how, given a set of peptide identifications, a minimal protein set is determined. Thereafter, we introduce a heuristic strategy for iterative precursor ion selection. Following that, we show how IPS can be formulated as linear program using a combination of the problems presented in Chapter 5. We evaluate both IPS strategies regarding different aspects like mass accuracy and sample complexity. Additionally, we discuss different termination criteria and finally present exemplarily two adaptations of the original LP formulation.

6.1. Protein inference

The protein inference problem, explained in section 2.3.1, is an instance of the set-covering problem presented in section 2.5.3 what is used in several protein inference approaches [120, 121]. Here, all peptide identifications form the universe U and sets of peptide IDs being part of the same protein build the subsets S_i.

Now, we want to find the minimal list of proteins, the set C, explaining all peptide identifications. Therefore, we have indicator variables y_i, which are 1 if protein i is part of the minimal list and 0 otherwise. Then, the ILP formulation looks as follows:

$$\min \quad \sum_i y_i \tag{6.1}$$

$$\text{s.t.:} \quad \forall_j : \sum_i a_{i,j} \cdot y_i \geq 1 \tag{6.2}$$

$$\forall_i : \quad y_i \in \{0, 1\}. \tag{6.3}$$

$a_{i,j}$ is an indicator variable, it is 1 if peptide j is part of protein i and 0 otherwise. Constraint 6.2 ensures that every peptide j is part of at least one protein i. Solving the ILP leads to a minimal protein list for which protein probabilities can calculated using one of the basic formulas of ProteinProphet as described in section 2.3.2.

In the next section, we introduce a heuristic that works on a ranked list of precursors. Subsequently, we present a formulation of IPS as linear program.

6.2. Heuristic

The heuristic iterative precursor ion selection (HIPS) presented in this section was published in the Journal of Proteome Research [17]. Figure 6.1 gives an overview on the workflow that is described in the next subsections. The following two subsections are adapted from [17].

6.2.1. Method

HIPS retrieves an LC-MS feature map and starts by ranking the features according to their score (see Figure 6.1 for the complete workflow). In our setting, the score reflects the ability of a feature to produce interpretable fragment spectra. Thus, it considers signal intensity and the existence of neighboring peaks which fall into the isolation window and therefore result in hard-to-interpret mixture spectra. It is computed by Bruker's WarpLC software. After feature ranking, the top scoring features are fragmented by MS/MS. A database search is performed and the retrieved proteins are categorized as *identified* or *uncertain candidates*. Afterwards, the feature map is compared to the m/z-values of tryptic peptides of all retrieved protein sequences. The score of features with m/z-values that match the *in silico* calculated peptides of already identified proteins is decreased as their selection is less likely to result in newly identified proteins than the fragmentation of other features. Conversely, fragmentation of features that match *in silico* cal-

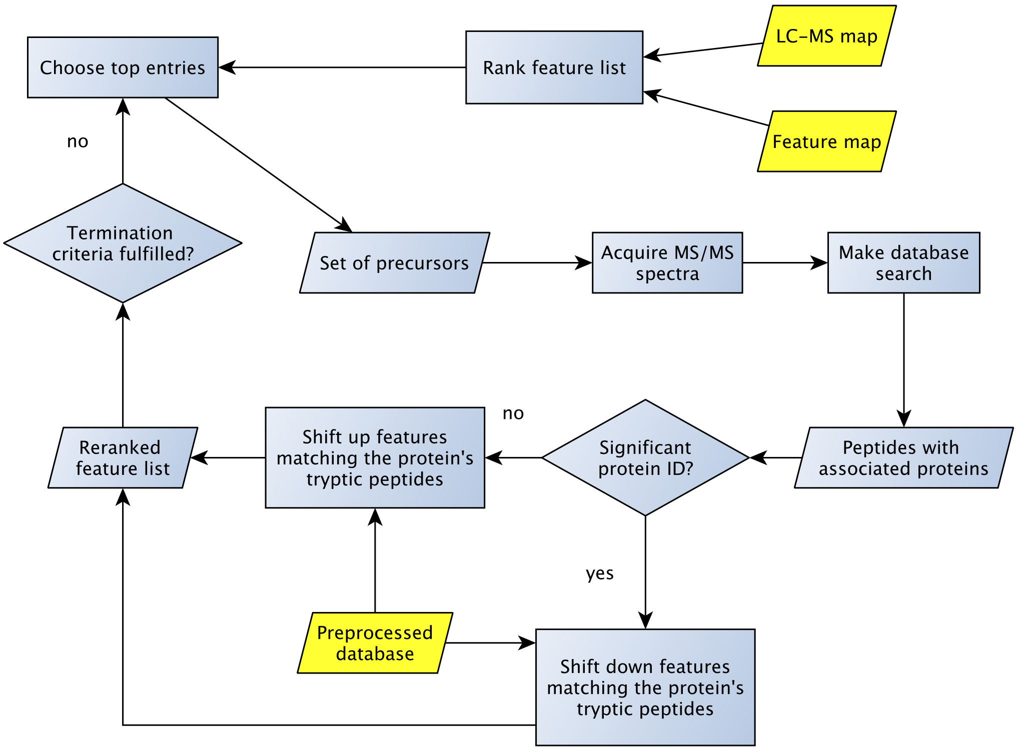


Figure 6.1.: Workflow of heuristic IPS. HIPS receives a feature map, an LC-MS map and a preprocessed database. It ranks the features and chooses the top entries for fragmentation. After MS/MS acquisition, a database search is performed. When a new significant protein ID was retrieved, the masses of its tryptic peptides are queried from the preprocessed database and matching features are shifted down in the feature list. When only a protein candidate was found, all its matching features are shifted up with the intention to safely identify the protein within the next iterations.

culated peptides of uncertain candidates are more likely to result in identifications than fragmentation of other features, and thus their score is increased.

After recalculating the scores of the features, MS/MS analysis is performed on the next top entries in the list. A new database search is performed with this MS/MS data set, and the identification results are combined with the previously retrieved results. This process is repeated until the set termination criteria have been fulfilled (see Section 6.4). The number of acquired MS/MS spectra per iteration, referred to in the following as *step size*, was set to 1 unless otherwise stated.

6.2.2. Rescoring

HIPS uses a simple strategy for changing the score of the features: if a feature has a mass matching a peptide of an already identified protein its score is basically halved, and if it matches an uncertain candidate, its score is set to the maximal score present in the list. However, often more than one peptide matches a given experimental *m/z*-value within the tolerated error range. The number of matching peptides varies depending on the *m/z*, the searched database, and the error tolerance. To account for this ambiguity, a weighting factor was used when rescoring the entries in the feature list. It is based on the frequency of peptide masses in the sequence database used for protein identification. To decrease the influence of the database size, the weights are scaled to the maximum relative frequency.

The weighting factor for a peptide with mass m is calculated as

$$w(m) = 1 - \frac{f(m)}{f_{max}}, \tag{6.4}$$

where $f(m)$ is the frequency of mass m in the database (within a specified error range) and f_{max} the maximal frequency. If m is very common in the database, i.e., the mass matches many different peptides, the weighting factor will be close to 0. For low-frequency masses it will be close to 1.

If a feature c with mass m is shifted down in the list its new score s_{down} is calculated as follows:

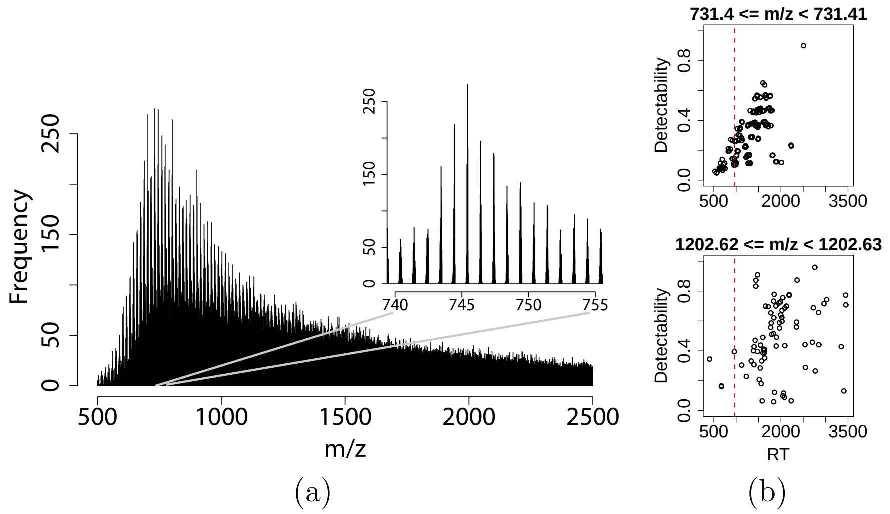
$$s_{down}(c) = s(c) - \frac{s(c)}{2} \cdot w(m) = s(c)\left(1 - \frac{w(m)}{2}\right). \tag{6.5}$$

For a very common mass, w is small and hence the score of the feature is decreased by only a small amount. Conversely, with a high weighting factor the score is approximately halved.

Analogously, the new score of a feature c that matches an uncertain protein candidate is increased:

$$s_{up}(c) = s(c) + (s_{max} - s(c)) \cdot w(m) = s(c)\,(1 - w(m)) + s_{max} \cdot w(m). \tag{6.6}$$

Here, a low weighting factor, i.e., a low frequency of mass m, leads to a new and higher score. The score can maximally be s_{max}, which is the maximum score found in the initial feature list. With the new score the feature is among the top entries. As the order of features is based on their initial score and the frequency of their masses in the database, the features that are most likely to give good identification results are at the top.



(a) (b)

Figure 6.2.: Distribution of tryptic peptides with 1 allowed missed cleavages computed using Swiss-Prot with taxonomy limited to human. (a) Mass distribution of charge 1 peptides, (b) RT and detectability distribution for two selected m/z bins. The minimal experimental RT is given by the dashed line.

6.2.3. Peptide mass distribution

An obvious drawback of HIPS is that the matching of features and peptides is solely based on their m/z-values. When large databases or complex samples are analyzed this inherently leads to a high number of erroneous assignments of theoretical peptides to observed features. For illustration, Figure 6.2 shows the distribution of peptide m/z-values in bins of 0.01 Da width[1] for Swiss-Prot with taxonomy human and 1 allowed missed cleavage. In extreme cases more than 250 distinct peptides fall in the same bin and are indistinguishable using only their m/z. Considering the largest bin containing 275 peptides, RT prediction eliminates already 34 peptides which have a predicted RT below the minimal RT in the experiment (Figure 6.2 (b) upper part). This bin contains peptides of lengths between 5 and 7 AAs, the majority has low detectability values. When we take a closer look at a second bin with m/z-values between 1202.62 and 1202.63, we can see that the RT distribution is wider, thus enabling a better resolution when RT and m/z are both considered for peptide-feature matching. Again, this bin contains many peptides with low detectability values what limits the possible number of matching peptides even further.

After introducing this heuristic approach to IPS and presenting the potential problem of erroneous peptide-feature assignments, we are now describing a formulation of IPS as optimization problem. It incorporates RT and peptide detectability to overcome the presented drawback of HIPS.

[1] This bin width corresponds to a mass accuracy of 10 ppm for m/z-values around 1,000 Da.

6.3. IPS as mixed integer linear program

In Chapter 5, we introduced different inclusion list creation problems that use an ILP formulation. We want to adapt these approaches to an iterative precursor ion selection and want to combine the feature map-based approach with the protein sequence-based approach into one iterative selection strategy. The goal is twofold: first, we want to identify as many proteins as possible and second, we want to maximize the number of selected features. As we have both integer and non-integer variables, we are now dealing with a mixed integer program (MIP).

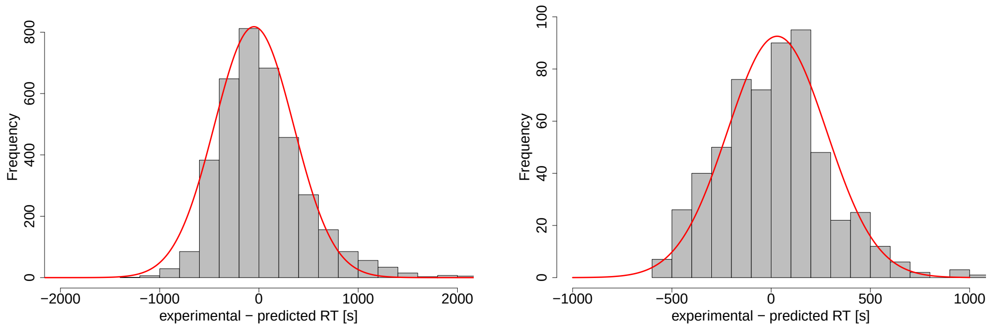
We start with the feature map-based ILP as presented in section 5.1 extended by an additional constraint limiting the number of selected precursors per iteration. For each feature j, we have several indicator variables $x_{j,s}$ that are 1 if feature j is selected as precursor in spectrum s and 0 otherwise. After solving the MIP, we retrieve a precursor set for which we trigger the acquisition. All $x_{j,s}$ corresponding to a feature selected in spectrum s are fixed to 1 for all future iterations. Afterwards, the MS/MS spectra are subjected to a database search and each resulting PSM is assigned to its corresponding proteins. Here, we distinguish between different cases:

- A match to a new protein not yet exceeding the protein probability threshold c. We want this protein to exceed c as soon as possible, so we aim at selecting precursors for this protein. We add a new variable for this protein to the MIP formulation and consider all features within a certain m/z-range of its tryptic peptides in the corresponding coverage constraint.

- A match to a new protein exceeding c. Again, we add a new protein variable to the MIP and consider all corresponding LC-MS features in its coverage constraint. However, as we have found enough evidence for this protein, any new peptide match only yields redundant information. Hence, we want to exclude these peptides from future selections. Therefore, the contribution to the objective function is decreased for all corresponding features.

- A match to a known protein not yet exceeding c. The coverage constraint of the protein is updated to contain the peptide probability of the newly identified peptide.

- A match to a known protein now exceeding c. Again the coverage constraint is updated with the peptide probability. Additionally, as in the second item, the contribution to the objective function is decreased for other features corresponding to the newly identified protein.



(a)

(b)

Figure 6.3.: Deviation of predicted and experimental RT (a) for HEK293, and (b) for UPS. The histograms show the observed deviations, the red curves represent an approximated Gaussian.

6.3.1. Calculating probabilities for the matching of theoretical peptides and LC-MS features

In the last section, we vaguely spoke of corresponding features which denote the set of features matching theoretical tryptic peptides of a protein determined by *in silico* digestion. In the following, we describe how we calculate matching probabilities for theoretical peptides and observed LC-MS features.

With the machine learning tools presented in sections 2.4.1 and 2.4.2 we are able to predict RT and detectability (PT) of a peptide given only its sequence. Using these two values, we want to estimate a probability that a certain feature in an LC-MS feature map corresponds to a theoretical peptide, both have m/z-values within a predefined mass range. As simplification we consider RT and PT to be independent. Mass accuracy is not directly included in the probability, it is only used to derive a set of peptides matching the particular feature. Then, matching probabilities are computed for this set.

The RT deviation can be approximated by a Gaussian distribution as shown exemplarily in Figure 6.3 for two data sets. Thus, the probability that a predicted RT t_{pred} is truly shifted by x spectra can be calculated as:

$$P(t_{pred} - t_{obs} = x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2}\left(\frac{t_{pred}-x-\mu}{\sigma}\right)^2}. \tag{6.7}$$

LC-MS features occur in several consecutive spectra which are all considered for RT probability calculation. As shown in Figure 6.4, the probability that a feature f corresponds to a predicted RT t_{pred} can be determined as the probability that the predicted RT deviates at least x_1 and at most x_2, where x_1 and x_2 denote the difference between predicted RT and maximal or minimal observed

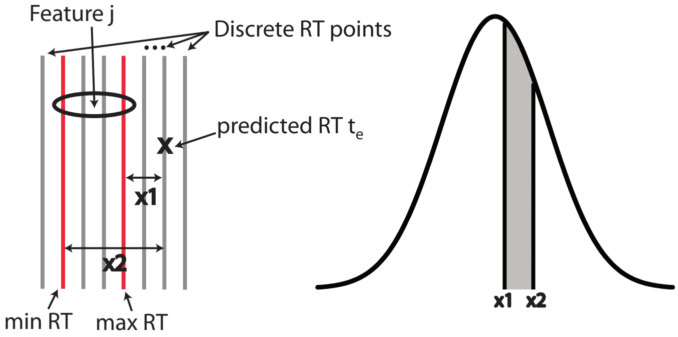


Figure 6.4.: Probability calculation for the matching of theoretical peptides and LC-MS features. For feature j its maximal and minimal observed RT are determined and their distance to the predicted RT is denoted by x_1 and x_2, respectively. Then the area under a Gaussian distribution, with preset mean and standard deviation, between x_1 and x_2 gives the probability that the RT prediction error lies between x_1 and x_2.

RT, respectively. Thus, they can be computed as

$$x_1 = t_{pred} - \max t_{obs} \quad \text{and} \quad x_2 = t_{pred} - \min t_{obs}. \tag{6.8}$$

This leads to the probability $r_{p,j}$ that the predicted RT of peptide p is truly shifted so that it lies within the RT range of the observed feature j as indicated by the gray area in Figure 6.4:

$$r_{p,j} = P(x_1 \leq t_{pred} - t_{obs} \leq x_2) \tag{6.9}$$

$$= P(t_{pred} - t_{obs} \geq x_2) \qquad - P(t_{pred} - t_{obs} \geq x_1) \tag{6.10}$$

$$= \int_{-\infty}^{x_2} \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2}(\frac{x_2-\mu}{\sigma})^2} \quad - \int_{-\infty}^{x_1} \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{1}{2}(\frac{x_1-\mu}{\sigma})^2}. \tag{6.11}$$

As said before, we assume RT and PT to be independent. Thus, combining the RT probability with the detectability of a peptide leads to the probability $m_{p,j}$ that an observed feature j corresponds to a predicted peptide p:

$$m_{p,j} = r_{p,j} \cdot d_p. \tag{6.12}$$

$m_{p,j}$ is computed for all features j with an m/z within the specified error range around the theoretical m/z of peptide p, this set of features is denoted as M_p.

6.3.2. MIP formulation

In the following, we want to incorporate the significance of a protein identification into the MIP. Therefore, we need the protein probability calculation as explained in Section 2.3.2, which gives us the probability P_i of protein i to be correctly identified, and a minimal protein probability c to declare a protein identified. Thus, we demand

$$P_i \geq c \tag{6.13}$$

$$\Rightarrow \log(1 - P_i) \leq \log(1 - c) \tag{6.14}$$

$$\Rightarrow \frac{log(1 - P_i)}{log(1 - c)} \geq 1. \tag{6.15}$$

The transformation above is only valid for P_i and $c < 1$, otherwise we enter a pseudocount instead. This way, we can define the indicator b_i which is 1 if $P_i \geq c$ and 0 otherwise:

$$b_i = \left\lfloor \frac{log(1 - P_i)}{log(1 - c)} \right\rfloor. \tag{6.16}$$

b_i is used in the exclusion part of the objective function. It indicates for which proteins, and thereby for which features matching theoretical peptides of these proteins, the contribution to the objective function is decreased as their probability already exceeds the threshold c.

This leads to a formulation of the combined MIP with an objective function composed of three parts: An inclusion and an exclusion part accounting for the number of identified proteins and a third part which maximizes the number of selected LC/MS features. The constraints account for the protein coverage (Constraints 6.18, 6.19), the maximal number of precursors per fraction (Constraint 6.21), the number of times a feature can be selected as precursor (Constraint 6.22), and the number of selected precursors in each iteration (Constraint 6.23). The protein coverage constraint (Inequation 6.18) consists of two parts: one is computed by the peptide probabilities and the other part by considering matching theoretical peptides for unidentified and so far not selected features.

$$\begin{aligned}
\max \quad & \overbrace{k_1 \sum_i z_i}^{\text{protein-based inclusion}} + \overbrace{k_2 \sum_{j,s} x_{j,s} \cdot int_{j,s}}^{\text{feature-based inclusion}} \\
& - \overbrace{k_3 \sum_i b_i \cdot \sum_p \sum_{j \in M_p} \sum_s m_{p,j} \cdot a_{i,p} \cdot x_{j,s}}^{\text{exclusion}} \qquad (6.17)
\end{aligned}$$

s.t.:

$$\begin{aligned}
\forall_i : \quad & z_i \leq \frac{log(1-P_i)}{log(1-c)} \\
& + \frac{\sum_p \sum_{j \in M_p} \sum_s x_{j,s} \cdot \log(1 - a_{i,j} \cdot m_{p,j})}{log(1-c)} \qquad (6.18)
\end{aligned}$$

$$\forall_i : \quad z_i \in [0,1] \qquad (6.19)$$

$$\forall_{j,s} : \quad x_{j,s} \in \{0,1\} \qquad (6.20)$$

$$\forall_s : \quad \sum_j x_{j,s} \leq cap_s \qquad (6.21)$$

$$\forall_j : \quad \sum_s x_{j,s} \leq 1 \qquad (6.22)$$

$$\sum_{j,s} x_{j,s} \leq precs + step\_size. \qquad (6.23)$$

The workflow for the iterative precursor ion selection with MIP (IPS_LP) is shown in Figure 6.5. The algorithm starts with a feature-based ILP formulation and during ongoing analysis fills in the protein coverage constraints and adds the protein-based parts to the objective function. The pseudocode for IPS_LP is shown in Algorithm 1.

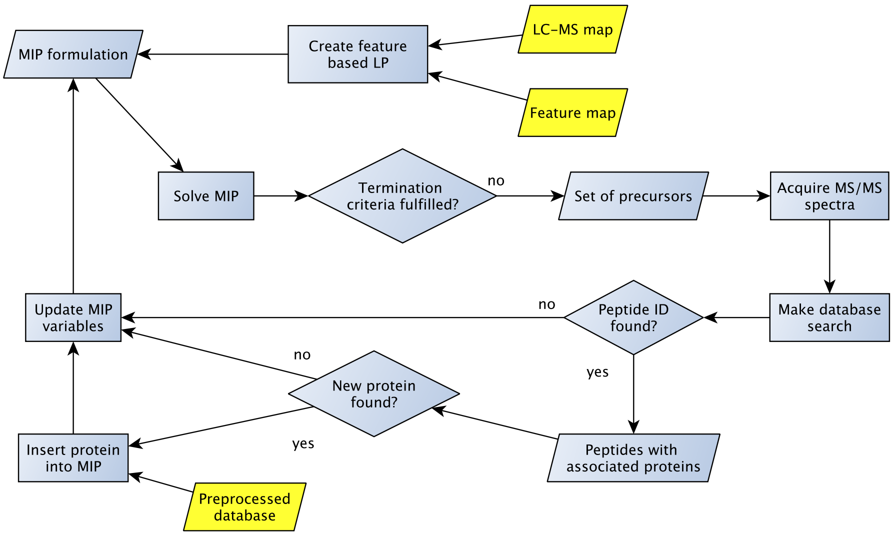


Figure 6.5.: Workflow for the iterative precursor ion selection with MIPs. Starting from an LC-MS map and a feature map, the iterative precursor ion selection creates a feature-based MIP and solves it. This way, a set of precursors is selected for which MS/MS acquisition is triggered. After a database search new protein hits are inserted into the MIP formulation and all MIP variables are updated. Afterwards, the MIP is solved again, leading to a new set of selected precursors.

Algorithm 1 Iterative precursor ion selection

```
createInitialLP(feature_map)
solveLP()
$solution\_indices \leftarrow$ getLPSolution()
$all\_protein\_ids \leftarrow \{\}$
$i \leftarrow 1$
while $\neg$ terminate() do
    for $s \in solution\_indices$ do
        $f \leftarrow$getFeature($s$)
        acquireMSMS($f$)
        $prot\_ids \leftarrow$ getProteinIds($f$)
        for $p \in prot\_ids$ do
            if $p \notin all\_protein\_ids$ then
                $all\_protein\_ids$.insert($p$)
                addProteinCoverageConstraint(p)
            end if
        end for
        updateLP()
    end for
    solveLP()
    $solution\_indices \leftarrow$getLPSolution()
    $i \leftarrow i + 1$
end while
```

6.4. Termination of iterative acquisition

A major goal of the presented iterative methods is to save sample and analysis time by completing the MS/MS analysis earlier. Thus, we need to define criteria when to stop the acquisition. Possible termination criteria are:

- **Maximal time/spectra:** A user defined maximal analysis time or maximal number of MS/MS spectra is reached. This is completely independent of the identification results.
- **Maximal number of protein/peptide IDs:** A user defined maximal number of protein or peptide identifications is reached. This is algorithm-dependent and can result in significantly different numbers of acquired MS/MS spectra and thus analysis time.
- **Maximal number of MS/MS spectra without peptide/protein ID:** For a given number of spectra, no new identification was achieved, either on peptide or on protein level.
- **Minimal level of efficiency:** The efficiency of the MS/MS analysis falls below a user defined minimal value. Efficiency can be defined as the number of identifications per MS/MS spectrum. Again, this can be done on protein or peptide level.
- **Minimal level of "local" efficiency:** The local efficiency of the MS/MS analysis falls below a user defined minimal value. In contrast to the efficiency defined in the last point, this is the number of identifications in the last x MS/MS spectra. This value depends heavily on x, if x is chosen too small the variation is quite high what might result in early termination.

6.5. Optimal solution

In this chapter, we present strategies for precursor selection made during MS/MS acquisition, which is influenced by the results of previously acquired MS/MS spectra during the same experiment. Thus, the presented methods are online algorithms which receive their input, the results of MS/MS processing, not as a complete set but as a sequence of input portions. Hence, the future input is not known to the system yet and the algorithm can only act based on the knowledge given by the previous input. A typical performance evaluation of online algorithms is done with competitive analysis [122, 123], where a given online algorithm is compared to an optimal offline algorithm, the adversary. Similar to that, we want to compare the performance of IPS with the optimal offline precursor ion selection that knows all peptide and protein identifications in advance. This optimal solution can be computed after the acquisition of all LC-MS/MS data and is presented in the next section.

6.5.1. Problem formulation

We want to find a minimal set of precursors such that all proteins of interest are identified, each feature is selected not more than h-times as precursor and each RT bin has not more than *cap* precursors. Similar to the inclusion list strategy presented in Section 5.2 this is an extension of the *Hitting Set Problem* as presented in section 2.5.2. However, in contrast to the original problem, where a minimal hitting set is sought-after, for our problem such a minimal set would usually mean that we cannot distinguish between proteins sharing the same peptide and thus the same feature. This is addressed in the protein inference, where indistinguishable proteins are grouped together and are counted as one protein ID. By maximizing the number of protein IDs, we aim for peptides separating these protein groups.

The complete MIP formulation looks as follows:

$$\max \quad k_1 \sum_i y_i - k_2 \sum_{k,s} x_{k,s} \tag{6.24}$$

$$\text{s.t.:} \quad \forall_s : \quad \sum_k x_{k,s} \leq cap_s \tag{6.25}$$

$$\forall_k : \quad \sum_s x_{k,s} \leq h \tag{6.26}$$

$$\forall_i : \quad y_i \leq \frac{\sum_{k,s} x_{k,s} \cdot \log(1 - a_{i,k} p_k)}{log(1-c)} \tag{6.27}$$

$$\forall_i : \quad y_i \in [0,1] \tag{6.28}$$

$$\sum_{j,s} x_{j,s} \leq precs + step\_size \tag{6.29}$$

$$\forall_{k,s} : \quad x_{k,s} \in \{0,1\}. \tag{6.30}$$

In the following results section, we compare performances of the iterative approaches to the optimal solution.

6.6. Results

In this part we evaluate both IPS strategies and compare them to the optimal solution and a static precursor ion selection (SPS), an inclusion list created before starting the MS/MS acquisition. This inclusion list is ranked by a score reflecting amongst other things the feature's intensity and the existence of nearby peaks that may cause interferences in the MS/MS spectrum. It is created using WarpLC from Bruker Daltonics. In the evaluation, we focus on the following subjects:

- Mass accuracy
- Sample complexity
- Abundance of identifications
- RT bin capacity
- Parameter robustness
- Step size
- Database size
- Termination criteria
- Run times

Afterwards, we present two adaptations of the MIP formulation. First, we are using a different ID criterion for proteins, the *two-peptide rule* which was introduced in Section 2.3.2. We show that it can be easily incorporated into the MIP and evaluate the performance of the different strategies when this ID criterion is applied. Next, we adapt the precursor ion selection to process RT fractions in a sequential order. This can also be done with minor changes to the MIP formulation. Finally, we evaluate the sequential MIP on a complex sample. Unless noted otherwise, we use the following weights for IPS_LP: $k_1 = 10$, $k_2 = 1$ and $k_3 = 10$.

6.6.1. Mass accuracy

We evaluated IPS with varying mass accuracy on the UPS sample. Figures 6.6 (a), (c) and (e) show the number of identified proteins over the number of selected precursors for decreasing mass accuracy. The three selection strategies are shown in blue (SPS), green (HIPS) and red (IPS_LP). Both iterative methods perform better than SPS for 10 and 25 ppm mass accuracy. With a low mass accuracy of 50 ppm HIPS is to some extent worse than SPS. This is due to erroneous assignments of hypothetical peptides to observed LC-MS features. This risk rises with the allowed mass error tolerance. For IPS_LP this dependence is less pronounced. This is expected, as the incorporation of RT and PT prediction reduces the number of false assignments.

The performance difference of the IPS approaches compared with SPS is more explicitly shown in Figures 6.6 (b), (d) and (f), where the difference in the number of precursors required to identify a given number of proteins is shown in percent with respect to SPS. For 10 ppm mass accuracy both IPS methods perform very similar, except one outlier of HIPS. For 25 ppm the performances divide: although both methods can save up to 40% precursors compared to SPS, with ongoing analysis IPS_LP performs superior. For 50 ppm, HIPS is partly significantly worse than SPS, requiring around 40% more precursors.

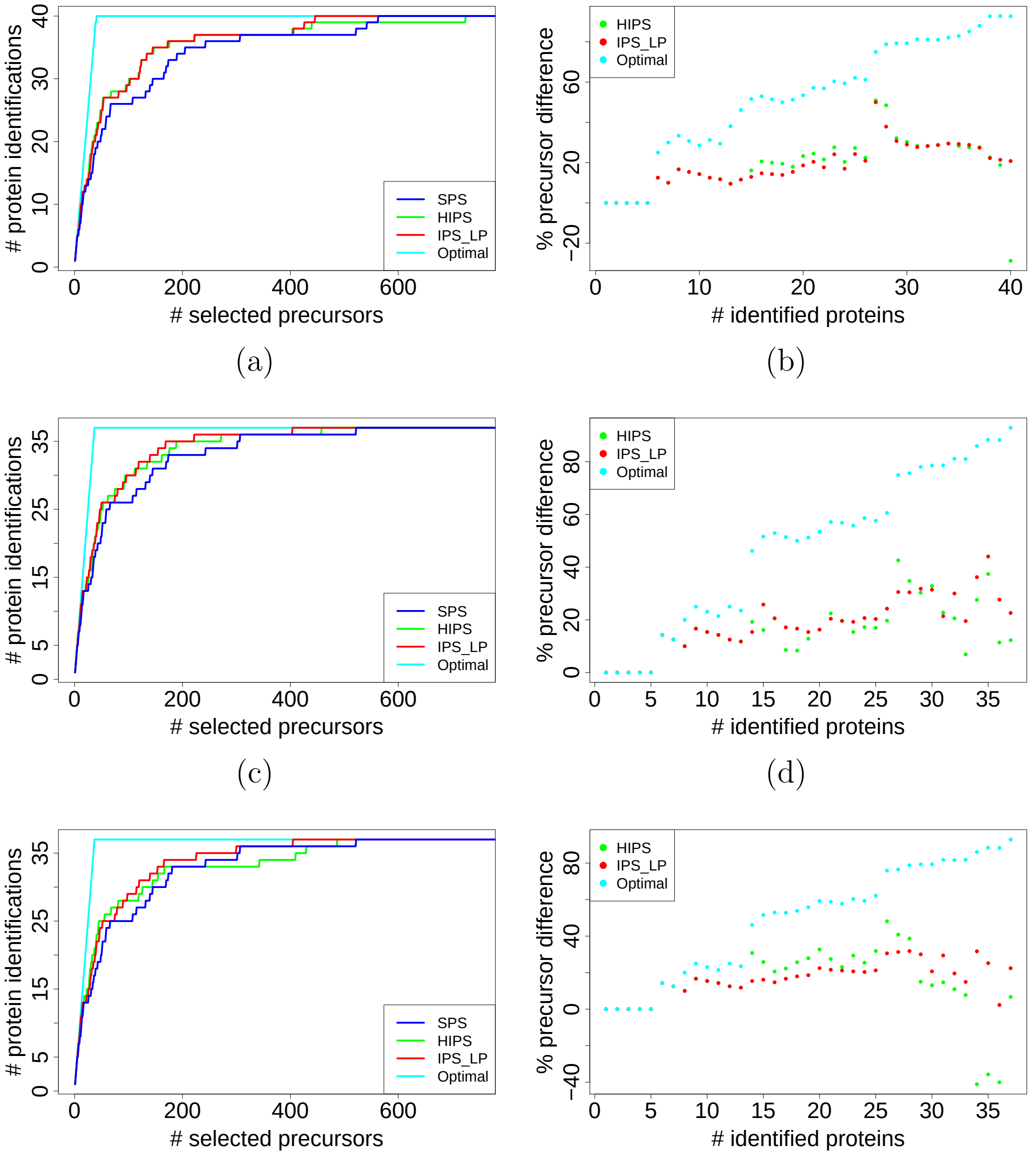
For comparison, the optimal solution (OPT), computed after acquisition of all MS/MS spectra and their processing, is included in Figure 6.6. This perfect competitor, that knows all peptide IDs and which proteins they are part of, shows the minimal number of spectra necessary to identify all proteins. Its performance is therefore independent of the mass accuracy.[2] With 10 ppm mass accuracy, it selects 41 precursors to identify 40 proteins. For both 25 and 50 ppm, 37 precursors are required to identify all 37 proteins. For all three tested mass accuracies, the online methods perform comparable to OPT up to around 10 identified proteins. However, for the final number of protein IDs the precursor saving for IPS_LP is around 1/4th of the one for OPT. This is expected as OPT is constructed so that every precursor contributes directly to the protein identifications. Both IPS methods try to select precursors that are likely to contribute to a protein identification. However, there are several reasons such as bad fragmentation or wrong peptide-precursor assignment that might lead to an unidentified peptide or a different peptide identification than expected.

As a next step, we compared the order in which the precursors were selected with the different selection strategies. So for each feature, we compared the iteration in which it was selected for the different strategies. In Figure 6.7 the ranks are shown for 10 ppm mass accuracy. For clarity, the diagonal is plotted in gray. Dots below it refer to precursors that are chosen earlier with IPS than with SPS. Negative values for IPS_LP indicate that these precursors are never selected with IPS_LP. For both IPS methods, we can see two trends. First, a large portion of precursors are selected later with IPS due to the exclusion part of the algorithms. A second trend is the line below the diagonal which basically follows the order of SPS. These precursors are not shifted by HIPS or are selected based on the feature-based inclusion part of IPS_LP, respectively. However, due to exclusion of other precursors they are selected earlier with IPS than with SPS.

6.6.2. Sample complexity

In the last paragraph we analyzed the performance of IPS on the UPS sample, an equimolar protein standard. In the following, we apply the methods to biologically relevant samples that contain proteins in varying abundances. Figure 6.8 (a) shows the results for the 50S sample, Figure 6.8 (b) for the HEK293 sample. In both cases the mass accuracy was set to 10 ppm.

[2] The slightly results are due to different database search results obtained for varying mass accuracies.



(a)

(b)

(c)

(d)

(e)

(f)

Figure 6.6.: Iterative precursor ion selection for UPS: (a), (b) 10ppm, (c), (d) 25 ppm, (e), (f) 50 ppm. (b), (d) and (f) show the relative difference in the number of precursors needed to identify a given number of proteins with respect to SPS.

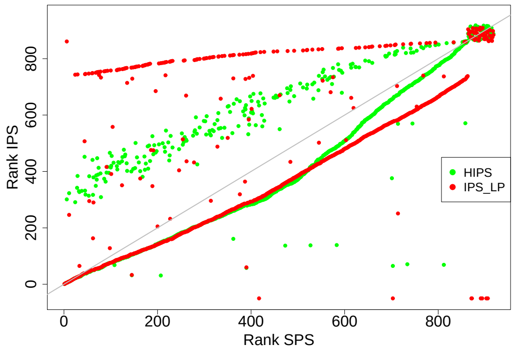


Figure 6.7.: Precursor rank comparison: Analyzed on UPS with 10 ppm mass accuracy. Ranks of HIPS are indicated by green dots, those of IPS_LP by red ones. The gray line shows the identity diagonal. Dots below the diagonal refer to features selected earlier as precursors with IPS than with SPS.

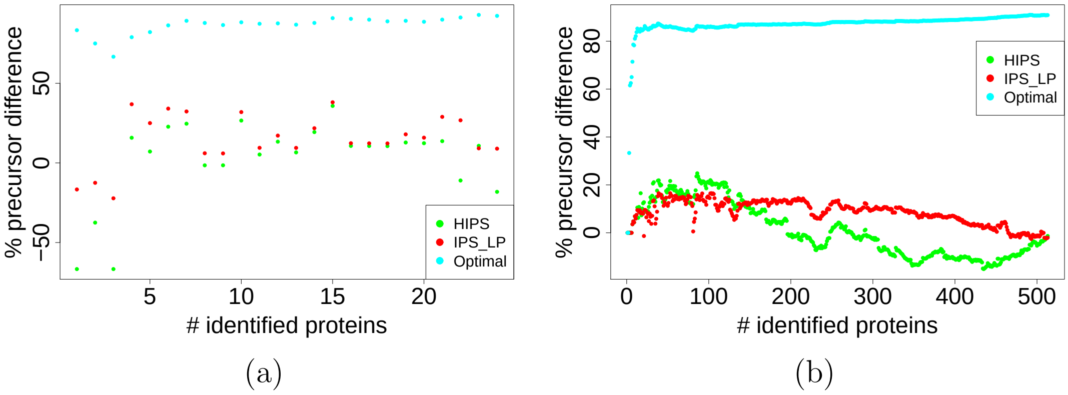


Figure 6.8.: IPS on biological samples: Iterative precursor ion selection with 10 ppm mass accuracy. The relative difference in the number of precursors needed to identify a given number of proteins with respect to SPS is shown for (a) 50S and (b) HEK293.

For the medium complexity 50S sample, IPS_LP can save up to 40% precursors, on average it saves 15%. In order to identify the first three proteins both IPS methods require more precursors than SPS, however the absolute values are -1 and -2 for IPS_LP and between -3 and -6 for HIPS, so this represents no drastic difference. Yet, HIPS also gets worse for higher number of protein IDs (22 and 24). Here, the relative difference is around -20% what translates to an absolute value of -33 and -71, respectively.

When looking at the difference in the number of required precursors for the high complexity HEK293 sample, we can see that at the beginning the heuristic works better than the MIP, which results in a maximal saving of 25% for HIPS and around 17% for IPS_LP.

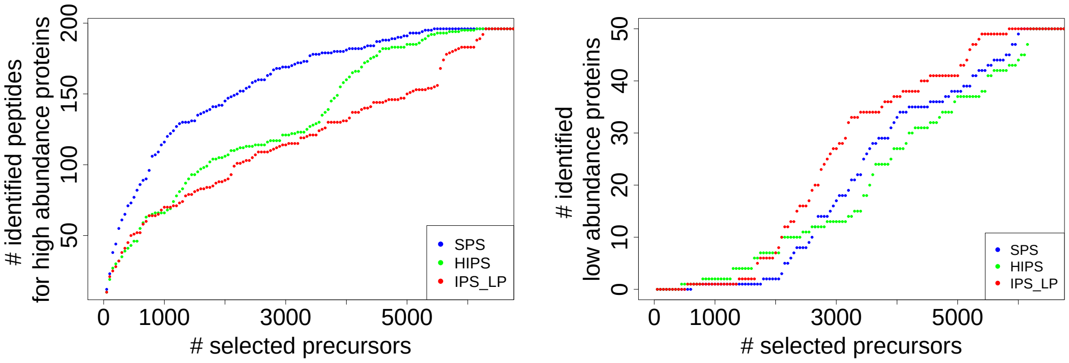
It is clearly visible, that for both samples the performance of both IPS strategies decreases with increasing number of identified proteins. This is expected as both are constructed in a way that in later stages of the experiment precursors are chosen which are less likely to improve the result. However, with IPS_LP this decrease is much less pronounced than with HIPS. This can be explained on the one hand by the reduction of erroneous precursor-peptide assignments through RT and PT prediction. Additionally, IPS_LP is looking for a global optimum, whereas HIPS selects its precursors in a greedy fashion which at the beginning yields good results but finally performs inferior.

For both biological samples, the difference between OPT and both IPS methods is more explicit than with UPS showing that there is further room for improvement for IPS.

6.6.3. Abundance of identifications

As pointed out before, intensity-based selection by construction is biased towards high abundance proteins and peptides. With our method, we aim at limiting the identified peptides for high abundance proteins to the number necessary for protein identification. This restriction shall increase the number of identified low abundance proteins. In the following analysis, we estimated protein abundance as mean feature intensity of all corresponding peptide identifications.

In Figure 6.9, we focus on two aspects. First, we compare the number of peptide identifications covering the 10% most abundant proteins. Here, we observe that for both IPS methods the total peptide number is smaller than for SPS for a large part of the experiment. For HIPS, the total peptide number starts to rise significantly after around 3,500 selected precursors. This steep increase can be explained by previously downranked precursors that are selected at that stage. A similar effect can be seen for IPS_LP, however, the increase occurs after 5,500 spectra. Again, it is probably a result of the exclusion part of the LP formulation. These results show that for the biggest part of the experiment the identification



(a) (b)

Figure 6.9.: Abundance of identifications: For the HEK293 sample with 10 ppm mass accuracy high and low abundance protein identifications are analyzed. Protein abundance is estimated as the mean intensity of all features with a corresponding peptide identification. (a) The number of identified peptides for the 10% most abundant proteins. (b) The number of protein identifications among the 10% least abundant proteins.

bias towards high-abundance proteins is less pronounced with IPS_LP than with intensity-based selection methods.

In Figure 6.9 (b), we analyzed the number of identified low intensity proteins. We considered the 10% least abundant proteins and counted the number of protein identifications. We observe that IPS_LP identifies the low abundant proteins earlier than SPS. Similar to the situation when looking at all protein identifications (Figure 6.8 (b)), HIPS is the best method at the beginning. After around 2,000 iterations its performance drops and HIPS is worse than the other two evaluated methods.

6.6.4. RT bin capacity

In a next step, we analyzed the influence of the maximal number of precursors per fraction, in the following called RT bin capacity, on the performance of the different selection methods. We varied the maximal capacity between 3 and 20 and show the number of protein identifications in Figure 6.10. The optimal selection identifies the maximal possible number of proteins already at a capacity of 3 precursors, thus varying the threshold does not change the performance and so these results are not shown in Figure 6.10.

For the three other methods the total number of identified proteins is similar only for capacities above 10 precursors per spot. When the spot capacity is very limited, IPS_LP is able to identify more proteins in total than the other two methods. This implies, that IPS_LP might especially be of value in situations where the sample amount is limited.

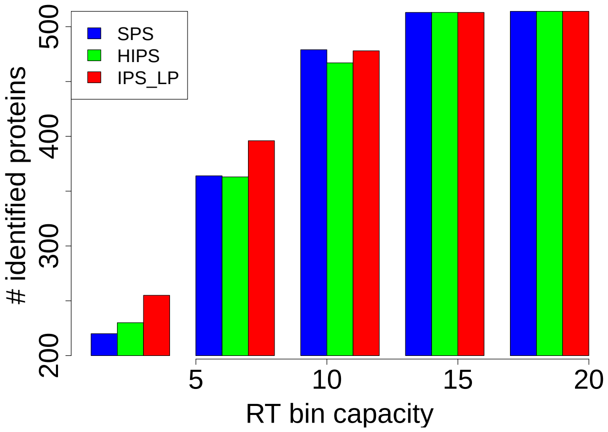


Figure 6.10.: Influence of RT bin capacity: Iterative precursor ion selection for HEK293 for 10 ppm mass accuracy. The total number of proteins IDs with a limited rt bin capacity is shown.

Interestingly, for a bin capacity of 10 precursors IPS_LP identifies one protein less than SPS. Looking closer at the identified proteins revealed that IPS_LP identified 25 proteins that were not identified with SPS which in turn could identify 26 proteins not found by IPS_LP. The majority of these protein differences are due to different selected precursors yielding different peptide IDs. However, we also observed differences due to shared peptides: One of these IDs is O15020 which has two peptide IDs, p_1 and p_2, assigned. p_1 was identified with low probability not exceeding the significance threshold. p_2, whose precursor was not chosen by IPS_LP, was identified with a significant probability. However, it is a shared peptide and other proteins that it is part of were identified before O15020. Thus, the contribution to the objective of the corresponding precursor of p_2 was decreased due to the MIP's exclusion part. This illustrates the potential problem of shared peptides with IPS_LP.

6.6.5. Parameter robustness

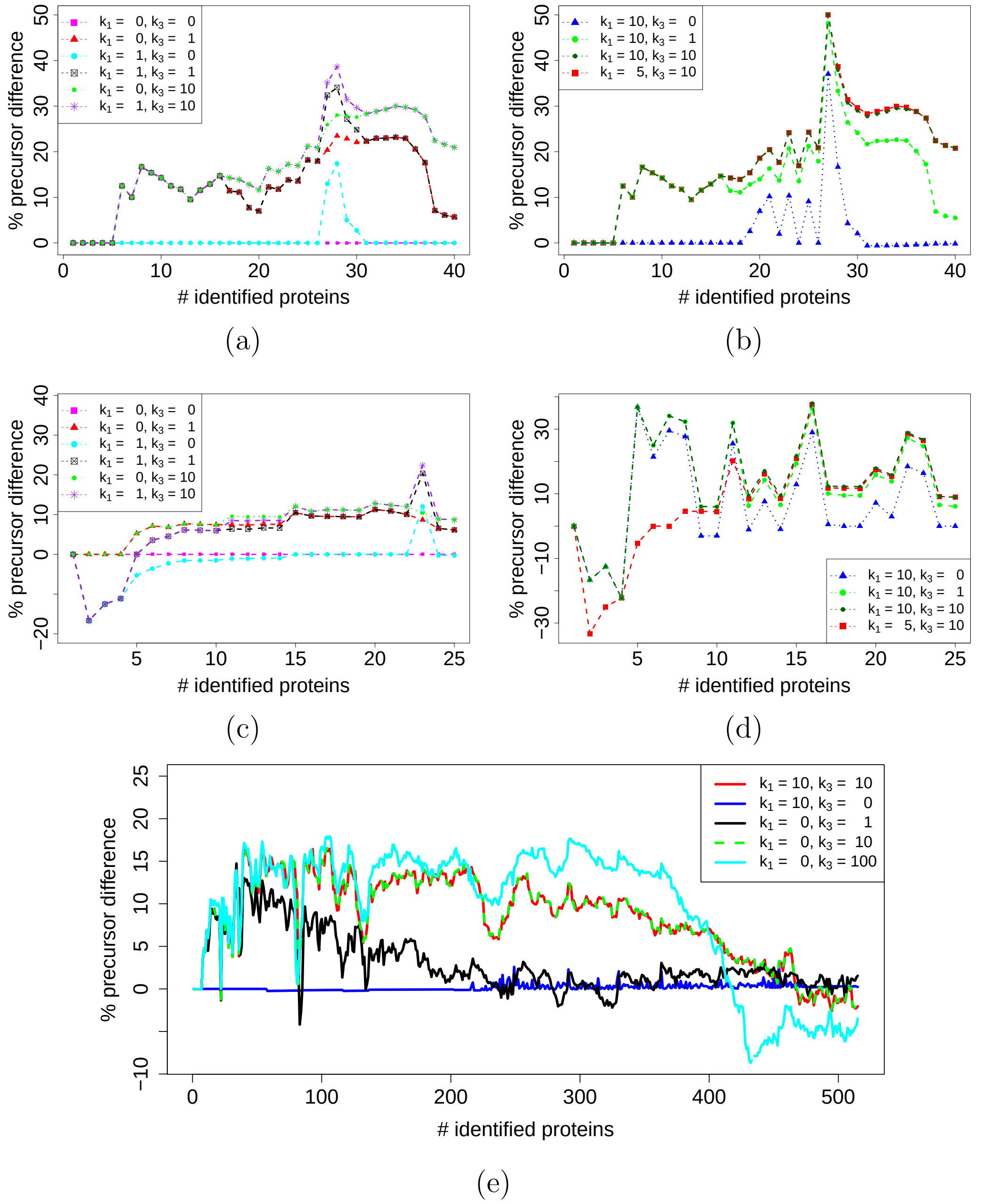
As shown in Section 6.3.2, the MIP formulation consists of three parts, protein-based inclusion, feature-based inclusion, and protein-based exclusion, which are weighted by terms k_1, k_2 and k_3, respectively. An obvious question is how robust is the system against different values for these weights and if each kind of sample requires a specific set of values. Thus, we analyzed our three samples for various parameter sets and show the results in Figure 6.11 (a) and (b) for UPS, in (c) and (d) for 50S, and in (e) for HEK293. In our analysis, we fixed $k_2 = 1$ as setting it to 0 results in early termination due to the absence of positive contributions to the objective function. As expected, setting $k_1 = k_3 = 0$ leads to the same performance as SPS as only the feature-based inclusion is switched on. For UPS, we can see that additionally switching on the protein-based inclusion with $k_1 = 1$ only leads to a temporary improvement between protein IDs No. 26 and 30. A similar pattern can be seen for 50S, here the protein inclusion leads to a performance decrease for the first 4 protein IDs. However, as discussed in Section 6.6.2 this is insignificant in terms of absolute precursor number differences. In general, setting $k_1 = 1$ does not yield a great performance improvement as the weight of 1 is too low to compensate for the weight of all features. A protein adds a contribution of z_i, which is maximally 1, weighted by k_1. Whereas each feature adds a contribution in the same range as z_i weighted by k_2 and the number of features can easily exceed the number of proteins by an order of magnitude.

For 50S, in the region between protein ID No. 4 and 10 especially instances with $k_1 = 10$ reach good results showing that here the protein-inclusion dominates the MIP and yields a good performance. Thereafter, a medium value of $k_1 = 5$ yields similar results. For this sample, switching on only the exclusion yields a constant performance improvement of approximately 10%. Comparing $k_1 = 10, k_3 = 0$ (blue curve in Figure 6.11 (d)) and $k_1 = 0, k_3 = 10$ (green curve in Figure 6.11 (c)) to $k_1 = 10, k_3 = 10$ (dark green curve in Figure 6.11 (d)) shows that the combination of inclusion and exclusion yields a better performance than both individually. Additionally, after each performance improvement due to protein inclusion follows a decrease, which is partly compensated if exclusion is switched on. Switching on the exclusion in general leads to a smoother curve compared to switching on inclusion. Similar observations can be made for UPS, Figure 6.11 (a) and (b), however, here all tested IPS_LP instances perform never worse than SPS.

The complex HEK293 sample behaves differently: switching on the protein exclusion part of the LP results in a performance almost completely similar to the one achieved with both inclusion and exclusion enabled. Thus, for this sample protein-based exclusion has more influence on the precursor ion selection than the protein-based inclusion. A lower weight for exclusion ($k_3 = 1$) produces inferior results to $k_3 = 10$, whereas a higher value of $k_3 = 100$ saves more precursors up to 400 identified proteins. Afterwards, this instance requires around 5% more precursors than SPS. This effect is probably due to erroneous precursor-peptide assignments during the exclusion and shows that too large values for k_3 might impair the results. In general, we observe that $k_1 = 10, k_2 = 1, k_3 = 10$ yields good results on all tested samples.

6.6.6. Step size

During IPS, in each iteration a database search has to be performed, the minimal protein list updated and the MIP formulation updated and solved. This results in a run time overhead as it is analyzed in Section 6.6.9. Choosing larger step sizes means decreasing the number of times these computations have to be made. In order to analyze the influence of the step size on the performance, step sizes were varied from 1 to 1000 for both iterative methods (Figure 6.12) using the



(a) (b)

(c) (d)

(e)

Figure 6.11.: Iterative precursor ion selection with varying weights. (a) and (b) UPS, (c) and (d) 50S, and (e) HEK293.

HEK293 sample.

Both methods show a different behavior with varying step sizes. For small to medium step sizes (1 to 100) HIPS performs similar up to around 100 protein identifications where step size 100 starts to perform inferior to the others. In the region between 250 and 450 protein IDs step sizes 1 and 10 are clearly superior to larger step sizes. Whereas for the last 50 protein IDs there are only minor differences between the step sizes, here the performance of HIPS approaches the one of SPS. While step size 500 is never better than lower step sizes, the largest tested size 1000 is in the region between 300 and 500 protein IDs partly better than smaller sizes. With this large number of precursors per iteration the erroneous assignments of theoretical peptides to observed features is less influential.

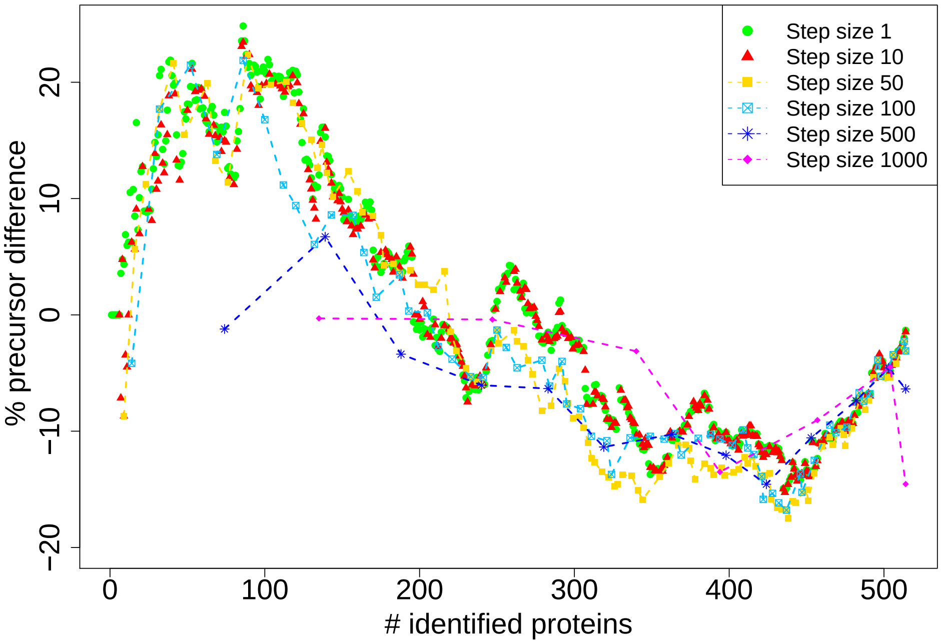
With IPS_LP the biggest differences can be observed for the first 120 protein IDs. Afterwards, differences between step sizes 1 to 100 are negligible. Here, probably the feature-based selection part dominates the objective function and therefore the performance of IPS_LP approaches the performance of SPS for all step sizes. In contrast to HIPS, large step sizes of 500 or 1000 are never better than smaller ones as one would expect if the assignment of features and theoretical peptides works reasonably well.

In summary, we observe that a step size of 10 seems to be a good trade-off between run time overhead and performance for both methods.

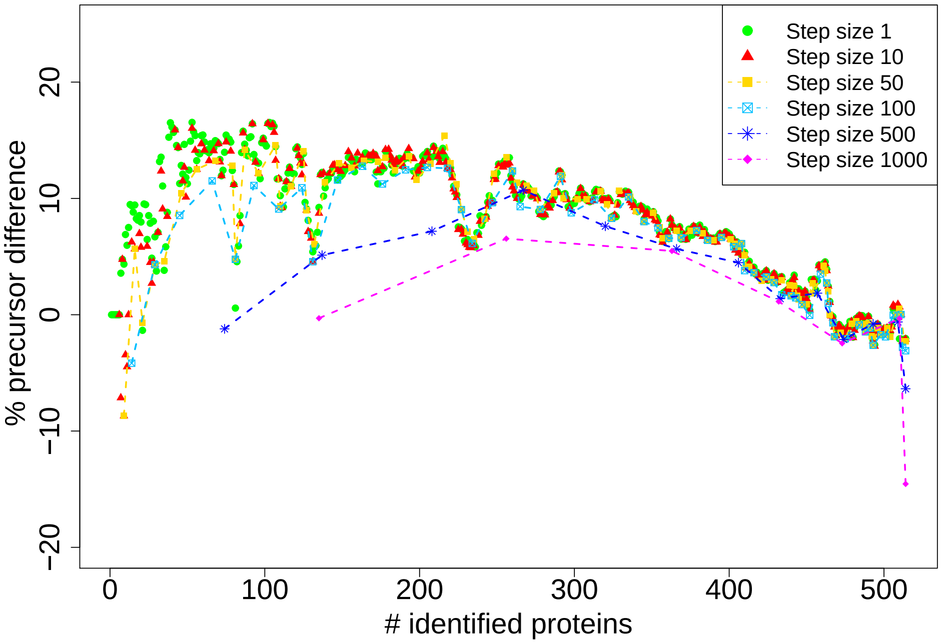
6.6.7. Database size

For the previous analyses, we used Swiss-Prot with limited taxonomy as database for peptide identification. In the following, we use databases with higher number of protein entries, namely IPI human (version 3.87 with 91,464 entries) and the complete Swiss-Prot database (Release 2011_08 with 531,473 entries), and evaluate the results of IPS on the UPS and HEK293 data sets (Figure 6.13).

For the UPS sample with IPI human, we can see that HIPS performs very comparable as before with Swiss-Prot human (Figure 6.6 (b)). The same holds for IPS_LP apart from the last three protein identifications. As we have seen before, changing weights for the exclusion part can have a big influence. Thus, we tested lower values for k_3 and observed that these do not clearly improve the performance. When the complete Swiss Prot database is used for peptide identification on the UPS sample, up to 30 identified proteins all IPS methods save between 15 and 20% precursors with respect to SPS. Thereafter, all IPS instances perform worse than SPS. This is obviously due to erroneous exclusion of precursors as using lowing values for k_3 partly compensates for that. This behavior is expected as Swiss-Prot contains homologous proteins of different species and thus more shared peptides than the Swiss-Prot database limited to human. As we have seen in Section 6.6.4 shared peptides can cause problems with our IPS

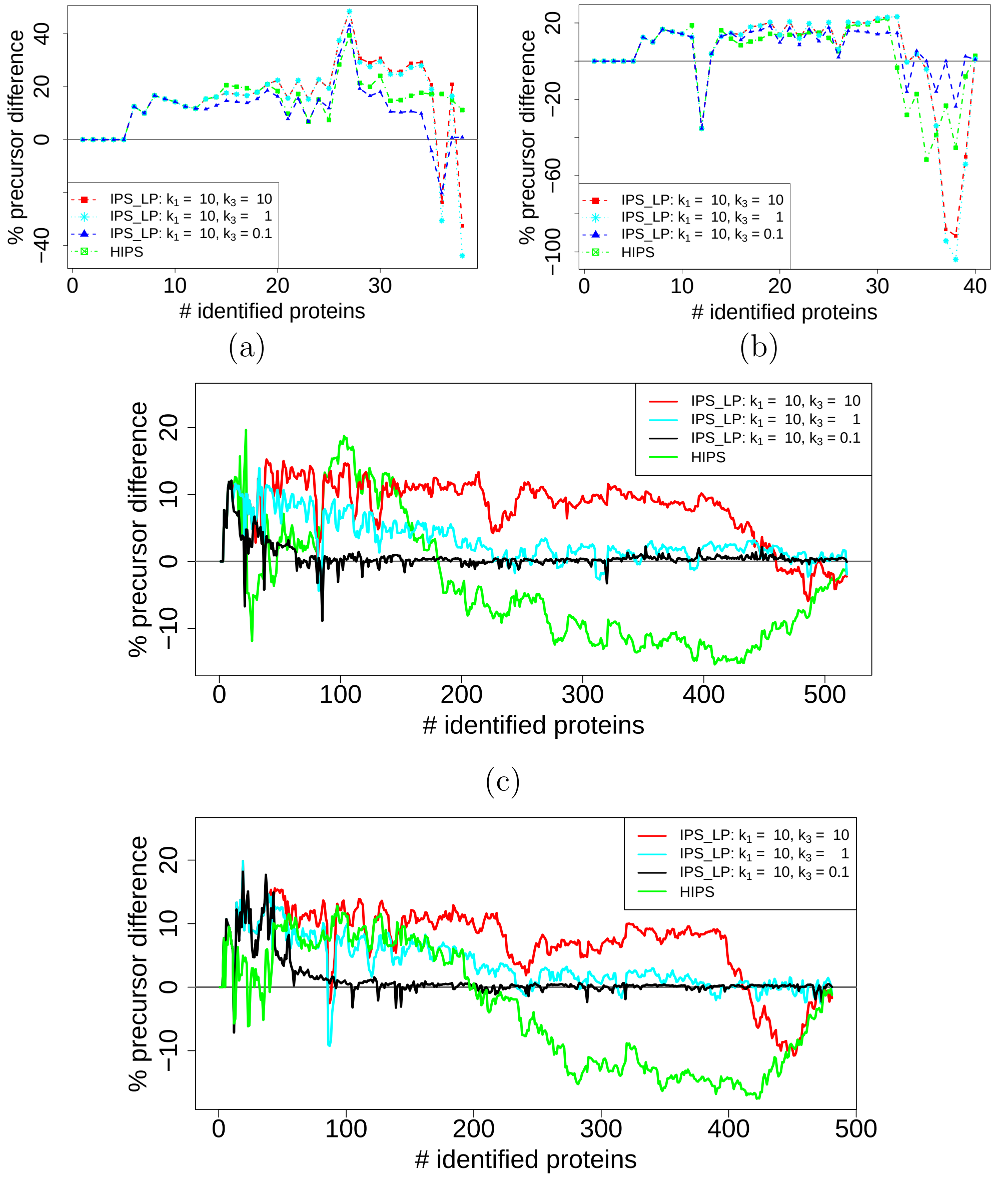


(a)



(b)

Figure 6.12.: Iterative precursor ion selection with varying step sizes for HEK293 with 10 ppm mass accuracy. (a) HIPS, (b) IPS_LP. For both iterative methods the step size was varied from 1 to 1000.



(a)

(b)

(c)

(d)

Figure 6.13.: Database size: Iterative precursor ion selection for 10 ppm mass accuracy with (a) UPS & IPI human, (b) UPS & Swiss-Prot, (c) HEK293 & IPI human, (b) HEK293 & Swiss-Prot.

approaches.

Figures 6.13 (c) and (d) show the results obtained with HEK293 on IPI human and Swiss-Prot, respectively. In both cases, HIPS performs worse than SPS for a large part of the experiment. For IPI human, IPS_LP with standard values $k_1 = 10, k_3 = 10$ performs around 10% better than SPS except for the last 50 protein IDs. Choosing lower values for k_3 compensates for the late performance breakdown, however, it also results in an overall worse performance. When Swiss-Prot is used as database, IPS_LP saves less precursors than with the other databases and the breakdown in late experiment stages is bigger. However, in a real experiment the analyzed organism is usually known and thus, the database taxonomy can be limited.

6.6.8. Termination criteria

As the goal of IPS is an earlier termination of MS/MS analysis in order to save time and/or sample amount, we now evaluate the performance of different termination criteria for the HEK293 sample. First, we are looking more closely at the local efficiency as defined in Section 6.4. Therefore, we tested different window sizes and show the results in Figure 6.14 (a) for IPS_LP with varying window sizes. As expected, a relatively small window size of 100 precursors has big fluctuations. With larger window sizes, the efficiency curves are smoothed. In Figure 6.14 (b) the local efficiency with a window of 1,500 spectra is shown for all three methods together with a gray line indicating a threshold of 0.05. Looking closer at the results for HIPS in the region between 5,000 and 6,000 selected precursors we can observe a problem of this termination criterion. For more than 1,000 spectra, the local efficiency of HIPS remains around 0.05, showing that setting the threshold to 0.05 results in early termination and thus a bad performance for HIPS. However, that is in a large part due to erroneous assignments of peptides to LC-MS features that were receiving a lower priority for selection in early iterations. When they get selected in later steps they increase the efficiency again. This can also be seen in Figure 6.8 where the performance of HIPS improves with higher numbers of identified proteins. When looking at the total efficiency shown in Figure 6.15, we can see that HIPS has the highest efficiency up to around 1,000 precursors. Afterwards, it decreases and is below the line for SPS.

We tested all termination criteria presented in section 6.4 and show the number of identified proteins and selected precursors in Table 6.1. When limiting either the number of acquired MS/MS spectra or the number of identified proteins, the results are very similar: HIPS performs worst, IPS_LP best and SPS between both but closer to IPS_LP. When applying result-dependent termination criteria, the results show a higher variability and are less predictable. For instance, when the number of spectra without a protein ID is limited to 100 (number (3)), the number of identified proteins is between 329 for SPS and 492 for IPS_LP. With this

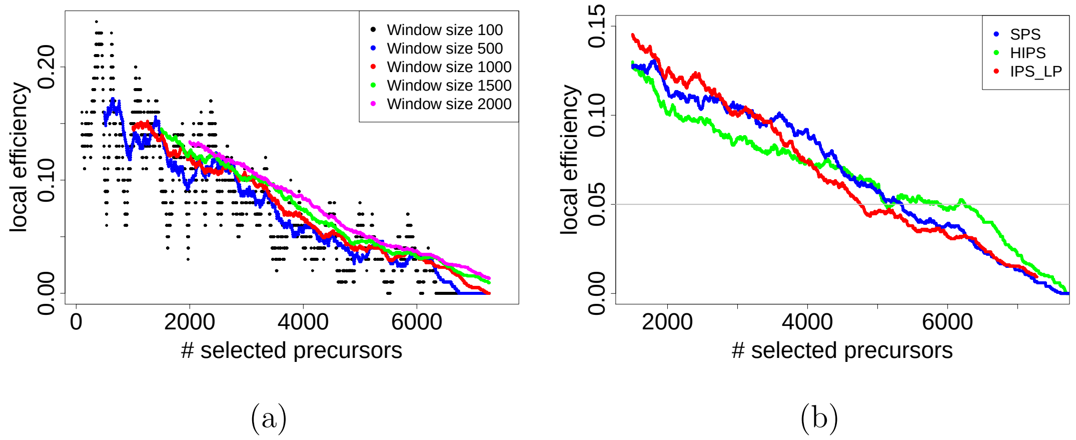


Figure 6.14.: Local efficiency of IPS with 10 ppm mass accuracy for HEK293 sample. (a) IPS_LP with varying window sizes. (b) for all methods with window size of 1500.

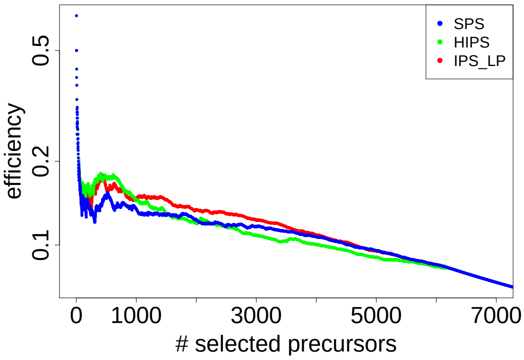


Figure 6.15.: Efficiency of IPS with 10 ppm mass accuracy for HEK293 sample.

Table 6.1.: Results for different termination criteria.

#	Termination criterion	Method	Threshold	# identified proteins	# precursors
(1)	# spectra	SPS	4,000	428	4,000
		HIPS	4,000	401	4,000
		IPS_LP	4,000	434	4,000
(2)	# protein IDs	SPS	400	400	3,582
		HIPS	400	400	3,962
		IPS_LP	400	400	3,333
(3)	# spectra w/o protein ID	SPS	100	329	2,854
		HIPS	100	435	4,689
		IPS_LP	100	492	5,529
(4)	efficiency	SPS	0.1	452	4,521
		HIPS	0.1	405	4,051
		IPS_LP	0.1	464	4,641
(5)	local efficiency (window size 1,500)	SPS	0.05	491	5,350
		HIPS	0.05	454	5,119
		IPS_LP	0.05	466	4,762

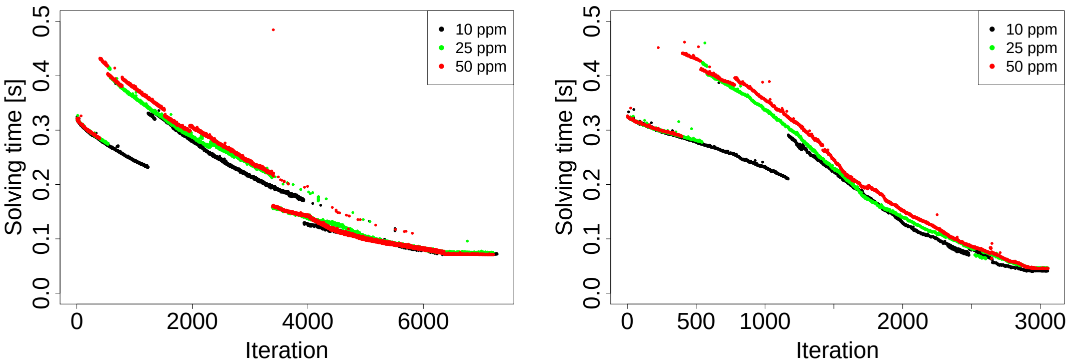
termination criterion, the latter approach selects almost twice as many precursors as SPS. The local efficiency, number (5) in Table 6.1, shows a similar performance. The total efficiency yields results comparable to the ones obtained with criteria (1) and (2).

These results show that termination criteria have to be chosen with care. Result-dependent methods like (3)-(5) can lead to an early termination due to local fluctuations.

6.6.9. Run times

In the following, we are analyzing times needed to solve the MIP in each iteration. All experiments were done on a machine with 72 GB RAM running with Intel Xeon X5550 processors with 2.67GHz. All run time experiments were using the HEK293 data set, which was the most complex in this study.

First, we measured run times for experiments with varying mass accuracy for RT capacities of 25 and 5 precursors per fraction, see Figure 6.16. In general, we can observe only a small difference in solving times between the tested mass accuracies. In all runs, we see at least one leap in solving times. A closer look at these leaps reveals, that all these are caused by a new protein hit which did not exceed the significance threshold for a save protein identification. Hence, several peptides are targeted by the inclusion part of the objective function. However, in all observed cases this leap is not the first incidence of such a protein hit, all



(a) (b)

Figure 6.16.: Run times of IPS with varying mass accuracy for HEK293 sample. (a) RT Capacity 25, (b) RT Capacity 5. Mass accuracy of 10 ppm, 25 ppm, and 50 ppm are indicated by black, green, and red dots, respectively.

instances have several protein hits not resulting in a steep increase of MIP solving time.

Limiting the number of precursors in each fraction results in a faster decrease of solving times. Another effect is that after the first leap the solving times are steadily decreasing without another leap as it can be observed for an RT capacity of 25 precursors. But again, a smaller RT bin capacity does not result in higher solving times although one would expect more conflicts as the total number of realized precursors decreases down to around 3,000.

In the next step, we varied the number of selected precursors in each iteration, also referred to as step size. In Figure 6.17 we show the solving times for 10 and 100 precursors per iteration for 50 ppm mass accuracy. Again, we tested RT capacities of 5 and 25. In general, we notice a very similar behavior as with step size 1. Thus, increasing the step size does not result in longer solving times. For a fraction capacity of 25 for both step sizes we can observe a solving time outlier: for a step size of 100 precursors solving the LP in this iteration takes nearly 5 seconds, more than 10 times the time than for all other iterations. Note that this outlier is not due to a measurement error. It was consistently observed in each of 10 separate runs. A similar outlier was already recognizable for a step size of 1, see Figure 6.16 (a). For all three step sizes, the same feature was in the selection set in the iteration leading to this long solving time. This feature leads to a manipulation of the LP formulation that must have triggered the application of heuristics enabled in GLPK. These heuristics resulted in longer solving times.

In summary, we can state that the main parameters of IPS like mass accuracy, RT capacity and step size have minor influence on the time needed to solve the MIP. In each iteration, before solving the LP formulation, it is manipulated, a database search is necessary, and eventually the target plate moved to a distant position. Each of these steps additionally influences the total time needed for an

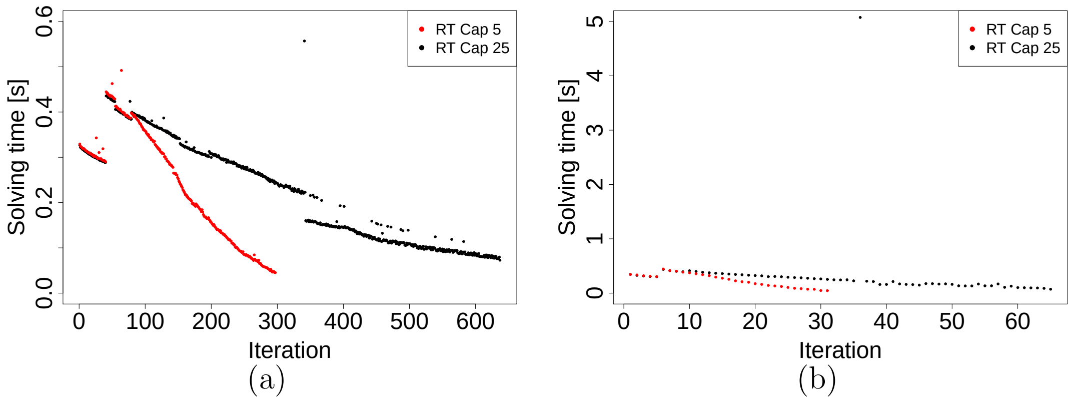


Figure 6.17.: Run times of IPS with varying step size for HEK293 sample with 50 ppm mass accuracy. (a) Step size 10, (b) Step size 100. RT capacities of 5 and 25 precursors per fraction are indicated by red and black dots, respectively.

iteration. Thus, in practice, especially the step size results in larger differences in running time, for instance, because database searches of many spectra can be parallelized, the LP is solved fewer times, and the target plate is moved less often as precursors of the same fraction can be selected sequentially.

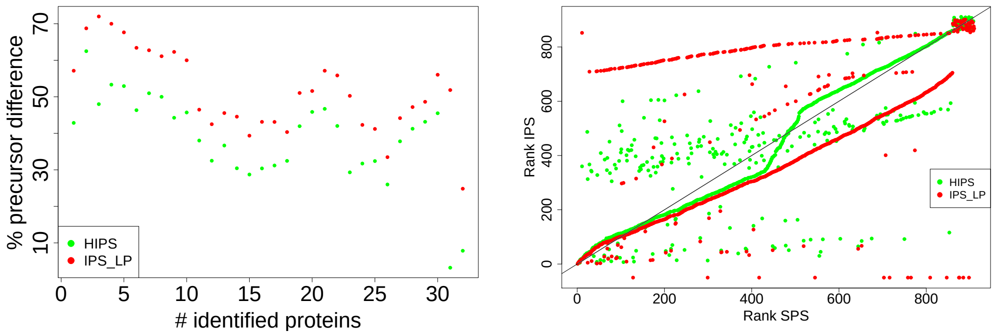
6.7. Adaptations

The MIP formulation can be easily adapted to variations of the precursor ion selection problem. This is shown exemplarily for two scenarios in the next sections. First, we use a different protein identification criterion, peptide counting, show the adapted LP and briefly evaluate it. Afterwards, we formulate a sequential precursor ion selection that chooses precursors following the order in RT dimension. This scenario is of special interest as it results in shorter analysis time in practice because the MALDI target plate is not moved after each fragmentation step.

6.7.1. ID criteria

There are various protein identification measures, as pointed out in section 2.3.2. So far, we used protein probabilities in the MIP formulation, but it can be adapted to incorporate other measures. In the following section we modify the formulation for a peptide counting approach, the *two-peptide rule*. This means, we demand at least two significant peptide IDs for an identified protein to exclude one-hit wonders.

Instead of requiring a minimal protein probability, we now want to achieve a



(a) (b)

Figure 6.18.: Iterative precursor ion selection for UPS with two peptide rule. (a) Percentage of saved precursors with iterative PS compared to SPS. (b) Rank of precursors in SPS compared to rank in iterative PS, HIPS in green, IPS_LP in red. For comparison, the gray line shows the identity diagonal.

minimal number m of peptide identifications that exceed a given peptide probability threshold p_{thr}. Therefore, we need to adapt constraints 6.18 and 6.19 in the following way:

$$\forall_i : \quad z_i \leq \sum_{j,s;a_{i,j}\cdot p_j \geq p_{thr}} x_{j,s} + \sum_{j,s;a_{i,p}\cdot m_{p,j} \geq m_{thr}} x_{j,s} \tag{6.31}$$

$$\forall_i : \quad z_i \in [0, m] \tag{6.32}$$

Inequation 6.31 counts the peptide IDs per protein that exceed the peptide ID threshold p_{thr}. $a_{i,j}$ is an indicator variable, which is 1 if peptide j is part of protein i and 0 otherwise. Thus, it ensures that only peptides of protein i are counted for its identification. The second part of Inequation 6.31 includes unfragmented precursors that potentially contribute to protein i: all precursors that have a predicted weight $m_{p,j} \geq m_{thr}$ are considered. This triggers the selection of precursors that are likely to stem from a peptide belonging to protein i. Constraint 6.32 ensures that at most m peptides are contributing to z_i for each protein i, so additional peptide identifications do not enhance the significance of a protein.

We evaluated the adapted iterative LP with the UPS sample using a mass accuracy of 10 ppm. In Figure 6.18 (a) the percentage of saved precursors with the iterative strategies in comparison with SPS is shown. IPS_LP requires on average around half of the precursors that SPS needs to identify a certain number of proteins, the maximum saving is 72%. HIPS saves on average around 40% and maximally 62%. The requirement of a certain number of peptide IDs per protein is well suited for the targeted precursor ion selection with an LP, everytime a peptide of a new protein is found this triggers targeting a certain set of peptides of which at least one is necessary for protein identification.

For analyzing further when the selection of different precursors is triggered, we plotted the ranks of the precursors in IPS against their rank in SPS in Figure 6.18 (b). As in Figure 6.7, we included the diagonal in gray. Thus, points below the diagonal correspond to precursors selected earlier with IPS than with SPS. The ranks of IPS_LP follow three trends: on the one hand, we have a certain number of precursors over the whole rank range of SPS that are selected at late stages or never with IPS_LP. This behavior can be explained by means of the exclusion part of the objective function. Second, we have a few points considerably below the diagonal, which indicate precursors with a high weight in the inclusion part of the objective function. Thus, these are precursors probably belonging to peptides that shall support a protein hit. The majority of IPS_LP precursor ranks follows a line close to the diagonal but below it which corresponds to the feature based inclusion part dominating the selection. When looking at the HIPS precursors, we observe a similar division in three parts. Although here, the exclusion of precursors is less strict: compared to IPS_LP the downranked precursors are selected earlier. Compared to the ranks obtained with a probability based identification criterion as shown in Figure 6.7, we can see that more precursors are selected due to the protein-based inclusion. This is expected as two peptides are necessary for a protein ID which always triggers protein inclusion after the first observed peptide.

6.7.2. Online approach for sequential order of target positions

An advantage of LC-MALDI-MS/MS is that the sample is fixed on a sample plate so that precursors can be chosen independently of their RT. However, when varying the RT the sample plate has to be moved. As this takes time, varying the RT after each MS/MS acquisition might not be feasible when analysis time is limited. Thus, in the following, we adapt the MIP formulation so that it proceeds through the precursor set in a sequential order according to the fraction number.

We start with spectrum $s^* = 0$. Only the capacity constraint of the MIP formulation (Inequation 6.21) has to be adapted to account for the sequential selection:

$$\forall_{s>s^*} : \sum_j x_{j,s} = 0 \tag{6.33}$$

$$\forall_{s<s^*} : \sum_j x_{j,s} = cap_s^* \tag{6.34}$$

$$\sum_j x_{j,s^*} = cap_{s^*} \tag{6.35}$$

Capacities of all fraction with lower number than s^* are fixed at the number of realized precursors in the fraction (cap_s^*). The capacities of all fractions with

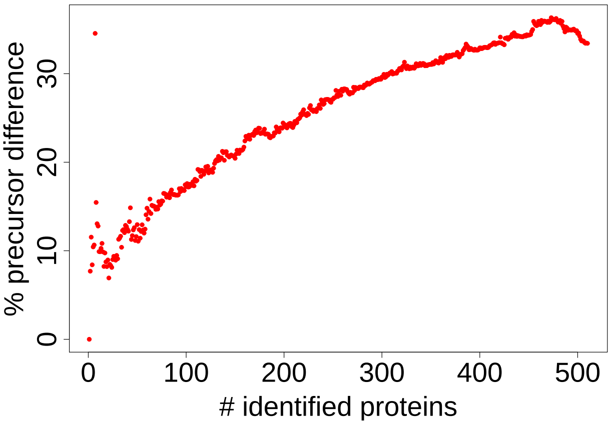


Figure 6.19.: Iterative precursor ion selection with sequential precursor ion selection for HEK293 with 10 ppm mass accuracy.

a higher number than s^* are set to 0. When all precursors in s^* were selected or when its capacity is reached, the next fraction is set as s^*. We evaluated the sequential IPS and illustrated the results in Figure 6.19. Obviously, the percentage in the difference of required precursors for a certain number of protein identifications rises with ongoing analysis and reaches a maximum of around 35% precursor saving after which it slightly drops again. Finally, IPS_LP saves more than 30% of the precursors. The steady performance increase is a result of IPS_LP selecting fewer precursors than SPS in most fractions. In the end, this sums up to more than 4,000 saved MS/MS spectra without a loss in protein identifications. Figure 6.19 shows an overview of the number of selected precursors per fraction for SPS and IPS_LP. With SPS, in the RT range between 3400 s and 7200 s almost all RT bins are used to their full capacity. Whereas, with IPS_LP only very few bins are completely used. The large amount of saved precursors becomes obvious for the sequential LP, however, it was already there for the non-sequential experiments presented in previous sections. As with IPS_LP, only precursors are chosen that contribute a positive weight to the objective function the selection stops if there are no more precursors with such a positive weight. This shows that with IPS_LP additional termination criteria as presented in Section 6.6.8 are not essential for its performance.

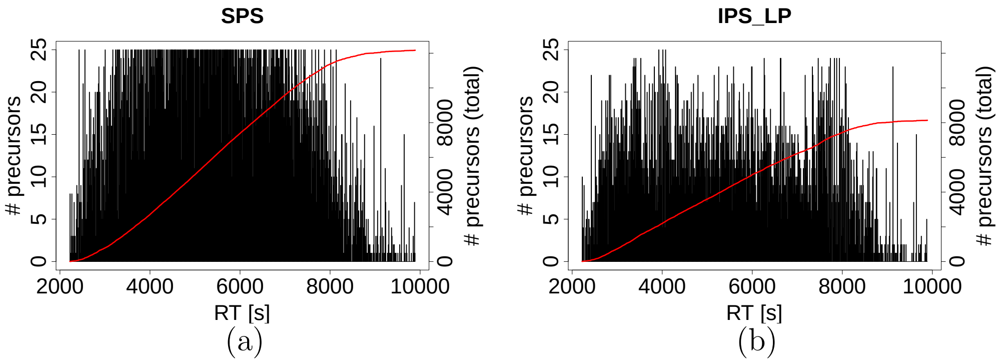


Figure 6.20.: Histogram showing the number of selected precursors per fraction for HEK293 for 10 ppm mass accuracy and a sequential precursor ion selection. (a) SPS, (b) ILP_IPS. The red line show the total number of selected precursors.

Chapter 7 Tools and Implementation

Throughout the last chapters, we focused on the algorithmic details and the evaluation. In this chapter we describe the implementation of the algorithms and tools that were developed for this thesis. First, we describe OpenMS, a C++ software library for LC/MS analyses, in which all tools are implemented. Afterwards, the tools *InclusionExclusionListCreator* and *PrecursorIonSelector* are introduced, which provide implementations of the algorithms presented in Chapters 5 and 6, respectively. Following that, we present *OnlinePrecursorIonSelector*, a tool that directly communicates with the mass spectrometer and controls the measurements. It has a user-friendly graphical interface for easily setting up all required parameters.

7.1. OpenMS

OpenMS is a C++ software library developed mainly by groups from the Eberhard-Karls Universität Tübingen, the Freie Universität Berlin, the Universität des Saarlandes, and the ETH Zürich. It provides implementations of efficient algorithms for common tasks in proteomics data analysis as signal processing, quantitation, identification and file conversion. It is freely available at `www.openms.de`. OpenMS provides data structures for efficient storing of basic MS data objects like raw data points, peaks, features or spectra. It supports standard data formats such as mzML, mzData or mzXML. Additionally, OpenMS includes TOPPView, a viewer for MS data. Built upon the OpenMS library, The OpenMS Proteomics Pipeline (TOPP) is a selection of tools for the main tasks in LC/MS data conversion and analysis which can be combined in workflows [60]. These workflows can be created using TOPPAS [124], which was used for MS/MS processing done for this thesis. InclusionExclusionlistCreator and PrecursorIonSelector, that are presented in the following sections, are available as TOPP tools.

7.2. InclusionExclusionlistCreator

The InclusionExclusionCreator can create both inclusion and exclusion list from various input sources. Inclusion lists are created from:

- featureXML: When the tool receives a featureXML file as input, either all features can be put into the inclusion list, or a selection based on the feature-based ILP formulation as presented in section 5.1 can be performed.
- fasta: For a fasta file input either all tryptic peptides of the sequences can be scheduled in specified charge states or a subset of these determined by the protein sequence-based ILP formulation as presented in section 5.2.

MSSimulator [125], a tool for MS and MS/MS simulation, uses the feature-based precursor ion selection in MALDI mode.

Similar to the inclusion list creation part also exclusion lists can be written for different input types: additional to featureXML and fasta, exclusion lists can be build upon identification results provided in an IdXML file. This can be used for excluding already identified signals in replicate analyses of the same sample.

7.3. PrecursorIonSelector

The algorithms for iterative precursor ion selection as described in Chapter 6 are implemented in the tool PrecursorIonSelector. For both HIPS and IPS_LP, a preprocessing of the database used for peptide identification is necessary. HIPS requires only the m/z values of all tryptic peptides and their frequency in the database. This frequency is used to scale the heuristic rescoring. IPS_LP additionally requires a trained RT and PT model. These can be created on a sample representative for the used experimental setup of the sample to be analyzed. The preprocessing for IPS_LP contains m/z values, predicted RTs and detectability values for all tryptic peptides present in the database. It needs to be created only once for each experimental setup and can be reused for later analyses.

IPS_LP creates an MIP formulation of the precursor ion selection problem. The implementation uses GNU Linear Programming Kit (GLPK, `www.gnu.org/software/glpk/`). First, an initial MIP formulation based on the feature-based ILP is created. Throughout ongoing analysis it is filled with protein information and solved in each iteration. Variables that turned 1 in the current iteration are traced back to the corresponding precursor and then can be returned in an inclusion list file.

PrecursorIonSelector offers a simulation mode that was used in the evaluation in Chapter 6. In this mode, all peptide IDs are given as input and matched onto the feature map. Hence, for each selected precursor the corresponding peptide is immediately known. Then, the whole IPS analysis is performed and the results, the number of identified peptides and proteins per iteration, are returned in a text file.

7.4. OnlinePrecursorIonSelector

The OnlinePrecursorIonSelector allows direct application of the PrecursorIonSelection tools on the MS instrument. It was developed to work on an in-house Bruker Ultraflex III mass spectrometer.

7.4.1. Implementation

Bruker Daltonics provided access to the software components for instrument control through their C++ library. An additional OpenMS dependency was created so that these components could be used directly out of OpenMS data structures and algorithms.

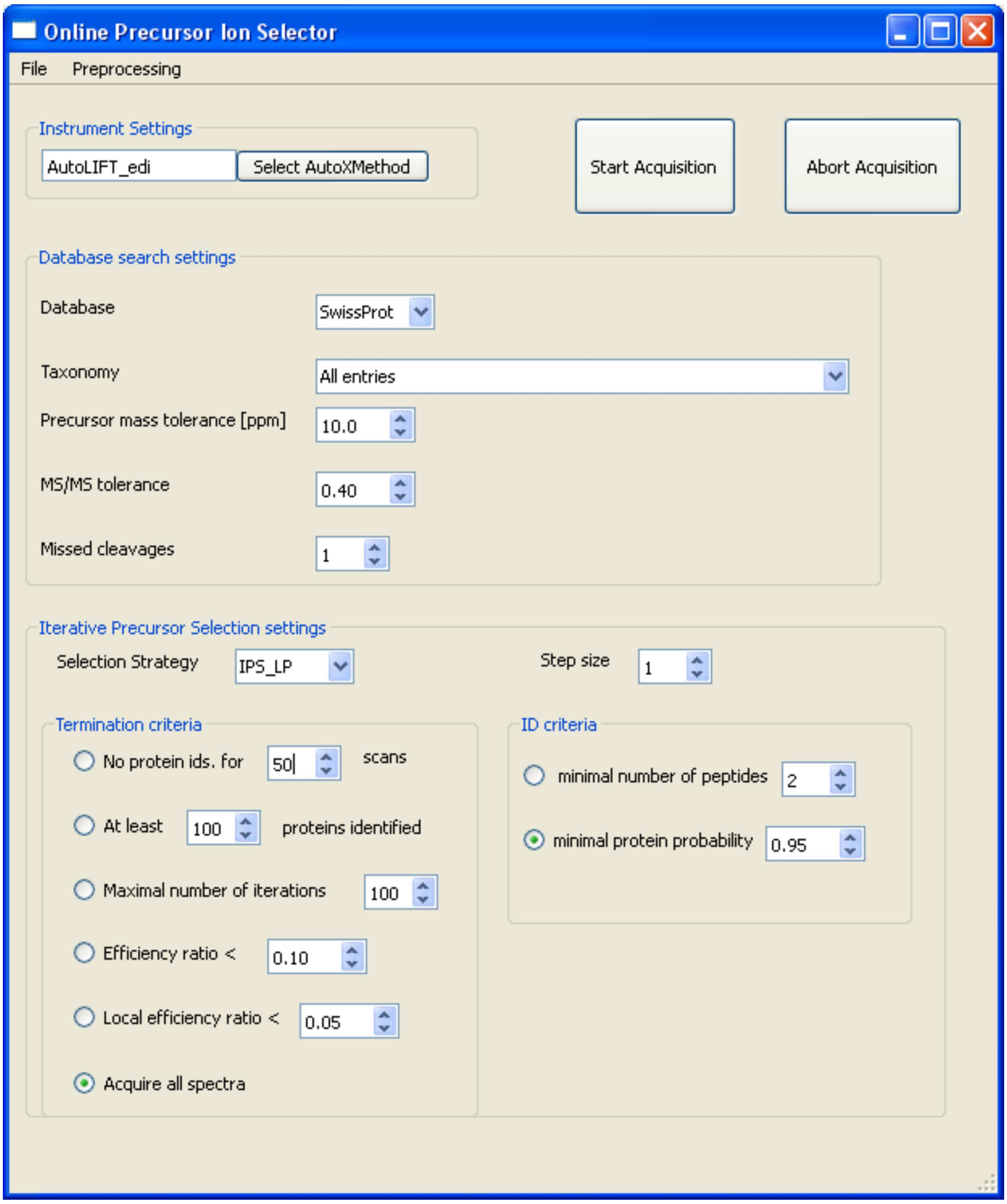
Then, in each iteration the set of selected precursors is translated into Bruker specific objects, the target plate moved to the current spot and the precursors' fragmentation is triggered. After this step a database search is performed using MascotOnlineAdapter and the MIP formulation is updated based on the identifications as it is done in offline mode.

The tool works directly on MS data acquired with the same instrument and processed with Bruker software. Thus, file adapters were written to handle Bruker's feature map and peak list XML formats.

7.4.2. GUI

OnlinePrecursorIonSelector offers a graphical user interface (GUI) to easily load the required data and configuration files and to tune the main algorithm parameters. It was created using Qt (`http://qt-project.org`). Figure 7.1 shows the GUI with its three main parts: *instrument settings*, *database search settings* and *iterative precursor ion selection settings*. In the *instrument settings* part the file containing instrument and MS/MS method specific parameters is chosen. These parameters are tuned for each sample before the run. The main *database search settings* can be changed directly, this includes the searched database, taxonomy, precursor and fragment mass tolerances, and missed cleavages. In the *iterative precursor ion selection settings* part there are the subsections termination and identification criteria. Here the user can choose, if the MS/MS acquisition should be stopped for instance when a certain number of proteins is identified or a maximal number of iterations is achieved. There are also efficiency related constraints like no protein identification for the last x MS/MS spectra or a minimal efficiency ratio. See section 6.4 for a detailed description of the termination criteria. For protein identification, the user can choose between unique peptide counting and a minimal protein probability calculated as described in section 5.2.1.

Figure 7.2 shows the File and Preprocessing dialogs. In the File dialog the user can load required files like the CompoundList file, a Bruker specific XML file similar to the OpenMS feature map file containing all features detected in the MS data. The AutoXSequence file used for MS acquisition is also loaded here. This file contains instrument and sample specific information and is needed for the instrument control. Besides, previously acquired MS/MS spectra can be loaded, e.g., for continuation of a stopped run. The preprocessing dialog allows to load, create and save the database specific preprocessing. For preprocessing creation the necessary RT and PT models can be specified.



Instrument Settings

AutoLIFT_edi

Database search settings

Database: SwissProt

Taxonomy: All entries

Precursor mass tolerance [ppm]: 10.0

MS/MS tolerance: 0.40

Missed cleavages: 1

Iterative Precursor Selection settings

Selection Strategy: IPS_LP

Step size: 1

Termination criteria

○ No protein ids. for 50 scans

○ At least 100 proteins identified

○ Maximal number of iterations 100

○ Efficiency ratio < 0.10

○ Local efficiency ratio < 0.05

● Acquire all spectra

ID criteria

○ minimal number of peptides 2

● minimal protein probability 0.95

Figure 7.1.: The GUI of the OnlinePrecursorIonSelector.

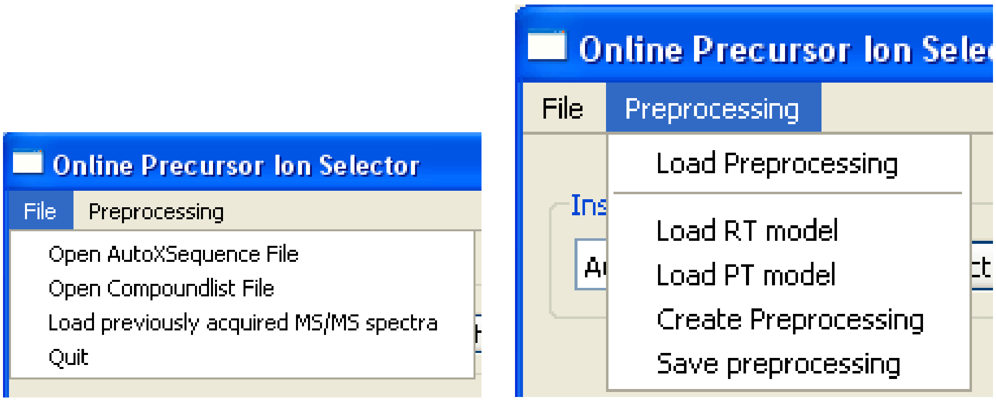


Figure 7.2.: Dialogs used in the OnlinePrecursorIonSelector.

Chapter 8 Conclusion

Precursor ion selection for MS/MS is an often disregarded topic. A typical workflow uses data-dependent acquisition provided by the mass spectrometer's manufacturer software despite its known drawbacks like limited reproducibility. In this thesis and the related publications we were among the first to systematically address iterative precursor ion selection with LC-MALDI MS/MS (together with Liu et al. [110]). Our aim is to go beyond maximizing the pure number of peptide identifications towards a more protein centric view of precursor ion selection.

In the last years, a complementary development for LC-ESI MS/MS took place, away from precursor ion selection to a simultaneous fragmentation of all ions in a broader m/z window, the so-called data-independent acquisition or MS^E which we presented in Section 3.3. [1] This development may lead to the question why to bother at all with precursor ion selection. However, these techniques pose major problems to data processing as MS/MS spectra are composed of fragments from different peptides. Typical processing approaches apply database searching either using the mixture spectra or using artificial MS/MS spectra created on the basis of elution profiles of fragment and precursor ions [126]. However, this analysis is very error-prone. Additionally, large selection windows in m/z and low fragment ion mass accuracy lead to overlapping fragment ions of different precursors, thus making the analysis of mixture spectra even harder [126]. To overcome this, some MS^E studies used smaller window sizes, however, then multiple LC injections are necessary to cover the full mass range. This is not suitable for high-throughput experiments.

In this thesis we developed formulations of several precursor ion selection scenarios as optimization problems and showed that they can be efficiently solved with LPs. As we demonstrated with different adaptations, our methods can be easily customized for different study requirements. For instance, Bertsch et al. [127] developed an LP formulation for the related MRM scheduling problem.

[1] The window size can vary from the full mass range to a few Daltons [126].

8.1. Inclusion lists

In this thesis, we presented methods for inclusion list creation based on a different amount of available information. Given an LC-MS feature map, we showed how to formulate a multiple Knapsack Problem for selecting a maximal number of precursors given common constraints such as the maximal fraction capacity. This way, we select more precursors for fragmentation than data-dependent or greedy methods.

In protein quantification, often the proteins of interest are known. Thus, we can use this information for inclusion list creation. Here, we showed that this precursor ion selection problem is related to the Hitting Set Problem and can be efficiently solved via LPs. We demonstrated that once a certain inclusion list size is achieved a plateau in the number of protein IDs is reached. Larger inclusion lists only increase the number of peptide identifications.

In our approach, a likelihood value for a protein identification is directly included in the precursor ion selection: using peptide detectabilities, we calculate a detectability value for the corresponding protein. By maximizing the sum of protein detectabilities, we ensure that precursors are matching peptides of many different proteins. This is of practical value for studying protein quantification for large protein samples. For instance, Schmidt et al. [100] used a set of 5,000 proteotypic peptides to observe the expression levels of 1,680 proteins of a human pathogen at 25 different states. Our method can be used to select such a set of peptides and create an inclusion list for them.

Creating inclusion lists with LPs can facilitate a change in the order of the analytical workflow: the goal can be to look for differentially expressed signals first and then target these for precursor ion selection given constraints as the maximal number of precursors per fraction. As our method does not rely on a previous LC-MS run it is also suited for LC-ESI MS/MS analysis when additional constraints for considered charge states are included.

8.2. Iterative precursor ion selection

In Chapter 6, we developed two different approaches for iterative precursor ion selection where not the entire precursors are scheduled before MS/MS acquisition starts. Instead, in each iteration a database search is performed and the information obtained there guides the selection in subsequent iterations.

The first method, HIPS, is a heuristic that requires only the feature map and knowledge about the database which is used for peptide identification. Then, precursors that are likely to support a protein candidate are assigned a high priority. Whereas, precursors matching peptides of already safely identified proteins receive a low priority. This method identifies proteins using less precursors than necessary with a static inclusion list created before the start of the analysis. Its advantage is the limited amount of information needed for its application. Only the database used for peptide identification is needed for preprocessing where m/z-values of all tryptic peptides are computed. However, it has clear limitations with respect to complex samples or bad mass accuracies where it suffers from erroneous peptide-precursor assignments.

The second presented method, IPS_LP, addresses this problem by incorporating predictions for RT and peptide detectability. This way, IPS_LP is less dependent on the mass accuracy than HIPS as we showed in Chapter 6.6.1. IPS_LP is a combination of the two inclusion list approaches presented in Chapter 5 plus an additional exclusion of peptides of already identified proteins. Although IPS_LP requires specified weights for the three parts of the objective function, our analysis showed that similar values could be used for various samples. In our case, setting $k_1 = 10, k_2 = 1, k_3 = 10$ worked well for all tested samples. Additionally, analyzing different weights showed that the exclusion part of the objective function has a bigger impact on the performance than the inclusion part. This is not surprising, as many proteins were identified already by the first matching peptide. So in many cases no further targeting of other peptides was necessary to support a protein ID. The adaptation of the LP to require minimally 2 significant peptides for a protein ID showed that in this case more precursors were selected based on the inclusion part of the objective function.

We showed that IPS_LP requires less precursors than standard DDA and HIPS in almost all tested settings. We evaluated our algorithms on a well-defined protein standard and two biological samples of very different complexity. We analyzed the influence of different parameters on the performance of IPS. In Chapter 6.6.1, we could show that IPS_LP performs superior to standard DDA for all tested mass accuracies. When the number of precursors per fraction is small, for instance because of a limited amount of sample, we observed that IPS_LP identifies more proteins than the other methods. Furthermore, With IPS_LP we are able to limit the number of peptide identifications covering high abundance proteins. This way, we can overcome the inherent bias of intensity-based selection methods to find many peptides for a few frequently occurring proteins. A side effect of this limitation is that lower abundance proteins are identified with IPS_LP considerably earlier than with SPS. We observed in Section 6.7.1 that especially if more than one peptide is required for protein identification, IPS_LP performs superior to DDA and HIPS. This shows the potential of IPS_LP for quantitative analyses where usually several peptides are required per protein.

In Sections 6.4 and 6.6.8 we introduced and evaluated different termination criteria which can be applied to iterative or standard DDA precursor ion selection. Additionally, IPS_LP has an intrinsic termination criteria as acquisition stops when no variable has a positive contribution to the objective function. When reliable models for RT and detectability are used, and thus the risk of false exclusions is minimized, this leads to a significant number of saved precursors without performance loss.

We analyzed the solving times of the MIP formulation for different mass accuracies, fraction capacities and step sizes. In general, we observed solving times below 1 second and none of the analyzed parameters showed a clear difference in MIP solving times. However, altogether an iteration includes more than just solving the MIP so that in practice larger step sizes might be beneficial. In section 6.7.2 we examined the performance of a sequential order in terms of RT. This way times for moving the target plate are minimized. We observed that this sequential IPS_LP selects over 4,000 precursors less than SPS to identify the same number of proteins. Combining the sequential MIP with larger step sizes, for instance by selecting all precursors for a fraction at once, might lead to a good tradeoff between analysis time and the number of protein identifications.

As we have seen in Section 6.6.4, shared peptides, e.g., peptides that are part of more than one protein, can represent an obstacle for IPS. Limiting the database to the species of interest helps to decrease the amount of shared peptides by reducing the number of protein homologues. It is questionable whether one can decide which protein a shared peptide belongs to before all other peptide evidence in the sample are analyzed. In our case we implemented an approach where a minimal protein list is created. Thus, if a peptide is shared by proteins A and B and there are other identified peptides for A but not for B, the MIP presented in Chapter 6.1 chooses A over B. If no other peptides are available for A and B, both are of equal value and one is chosen randomly. It is possible to include other strategies for protein inference. There are many approaches that solve the problem of peptide degeneracy in different ways. The widely used tool ProteinProphet learns the weight for each protein using an EM algorithm [70]. Recently, Huang and He [128] presented a linear programming approach that uses the joint probability that both a protein and its constituent peptide are present in the sample.

8.3. Future directions

In our evaluation, we compared SPS and IPS to the optimal solution which can be determined after the experiment when all MS/MS measurements are done. Although a difference between online algorithms like IPS and the offline optimal solution is expected as not all precursors yield the predicted identifications, this comparison showed that there is still room for improvement. One possible extension would be the inclusion of a fractional mass filter. We showed in Figure 6.2 that peptide m/z values appear in clusters with approximately 1 Da distance. Between these clusters no peptides occur. This characteristic can be used to discriminate non-peptide from peptide signals. Additional to RT and PT prediction, Liu et al. [110] applied such a filter successfully for their IPS method. Another possible extension would be to include a peptide mass fingerprinting (PMF) step prior to MS/MS analysis. PMF was developed independently by several groups in 1993 [129–133]. It is a technique used for protein identification based on the peptide masses determined with an MS run. Applying PMF after the initial LC-MS run yields a list of proteins whose corresponding peptides can be targeted with LC-MS/MS. This might improve the efficiency of IPS and could result in a greater impact of the protein-inclusion part of IPS_LP.

In our current setup, we only allow fixed modifications and not variable PTMs. However, it was shown that the number of modified peptides rises with decreasing protein abundance [134, 135]. Thus, including variable PTMs into our methods might lead to a higher number of identified low abundance proteins. Incorporation of variable modifications has several consequences. First, the methods for RT and PT prediction need to be able to cope with modifications which is the case for the machine learning techniques we applied. However, a good training set containing a representative set of PTMs is also required. A drawback is that by including variable PTMs in our methods the set of theoretical peptides for a protein grows exponentially. This implies a higher chance for false assignments of observed precursor ions and theoretical peptides and might impair the overall performance of our precursor ion selection. The higher number of candidate peptides also increases the running time of the database search for peptide identification. A compromise might be the inclusion of variable PTMs only at late experiment stages when already a large number of high abundance proteins is identified. Then, the PTMs might enhance the identification of otherwise hard to identify low abundance proteins.

As pointed out repeatedly in this thesis, the amount of detectable precursors often dramatically exceeds the amount of possible MS/MS measurements. This problem can be addressed by running multiple repeat measurements and focusing each time on different precursor ion sets. Thus, a possible extension of the methods presented in Chapter 5 would be the simultaneous creation of inclusion lists for multiple experiments. This can be achieved by introducing a third index to the precursor indicator variable which points to the experiment. This way $x_{e,j,s}$ would be 1 if feature j is chosen in experiment e in fraction s. However, the number of experiments has to be limited or needs to be considered in the objective function. Otherwise, the feature-based ILP formulation would create inclusion lists for new experiments until no unscheduled feature is left. Alternatively, results of previous runs can be considered while creating the inclusion list. For the feature-based inclusion this can be done by aligning the previous feature maps to the current one and afterwards forbidding the selection of already identified features.

In the future, it will be interesting to use iterative, result-driven precursor ion selection for LC-ESI MS/MS. Therefore, a fast online database search is necessary. Lately, Graumann et al. [111] and Bailey et al. [112] presented tools that incorporate such a database search on the fly and used the results for mass calibration during the measurement or targeted resequencing of peptides. Recently, Webber et al. [136] published an open source framework for Thermo Fisher instruments that hides the complexity of the instrument firmware from the user and enables customized data acquisition via python scripts. This way, the development of targeted selection strategies is significantly simplified. In order to use IPS_LP for ESI, multiple charge states of peptides have to be included into the LP formulation. This increases the possible number of peptide matches in the database for a precursor what might lead to more erroneous assignments and consequently to a worse performance. However, the sequential LP formulation that was presented in Section 6.7.2 showed a good performance and was a significant improvement over data-dependent precursor ion selection. This is a promising result motivating the development of a similar method for LC-ESI MS/MS.

Bibliography

[1] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431:931–945, Oct 2004.

[2] E. S. Lander et al. Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, Feb 2001.

[3] J. C. Venter et al. The sequence of the human genome. *Science*, 291(5507): 1304–1351, Feb 2001.

[4] `http://www.gencodegenes.org/stats.html`, 2012. [Online; accessed 02-January-2013].

[5] M. R. Wilkins, C. Pasquali, R. D. Appel, K. Ou, O. Golaz, J. C. Sanchez, J. X. Yan, A. A. Gooley, G. Hughes, I. Humphery-Smith, K. L. Williams, and D. F. Hochstrasser. From proteins to proteomes: large scale protein identification by two-dimensional electrophoresis and amino acid analysis. *Biotechnology (N.Y.)*, 14:61–65, Jan 1996.

[6] D. L. Tabb, L. Vega-Montoto, P.A. Rudnick, A.M. Variyath, A. J. Ham, D. M. Bunk, L. E. Kilpatrick, D. D. Billheimer, R. K. Blackman, H. L. Cardasis, S. A. Carr, K. R. Clauser, J. D. Jaffe, K. A. Kowalski, T. A. Neubert, F. E. Regnier, B. Schilling, T. J. Tegeler, M. Wang, P. Wang, J. R. Whiteaker, L. J. Zimmerman, S. J. Fisher, B. W. Gibson, C. R. Kinsinger, M. Mesri, H. Rodriguez, S. E. Stein, P. Tempst, A. G. Paulovich, D. C. Liebler, and C. Spiegelman. Repeatability and reproducibility in proteomic identifications by liquid chromatography-tandem mass spectrometry. *J. Proteome Res.*, 9:761–776, Feb 2010.

[7] M. Mann, A. Michalski, and J. Cox. More than 100,000 detectable peptide species elute in single shotgun proteomics runs but the majority is inaccessible to data dependent LC MS/MS. *J Proteome Res*, Feb 2011.

[8] J. T. Watson and O. D. Sparkman. *Introduction to Mass Spectrometry: Instrumentation, Applications, and Strategies for Data Interpretation*, page 199. John Wiley & Sons, 2008. ISBN 9780470516881.

[9] H. Liu, R. G. Sadygov, and J. R. Yates. A model for random sampling and estimation of relative protein abundance in shotgun proteomics. *Anal. Chem.*, 76:4193–4201, Jul 2004.

[10] P. Juhasz, M. Lynch, M. Sethuraman, J. Campbell, W. Hines, M. Paniagua, L. Song, M. Kulkarni, A. Adourian, Y. Guo, X. Li, S. Martin, and N. Gordon. Semi-targeted plasma proteomics discovery workflow utilizing two-stage protein depletion and off-line LC-MALDI MS/MS. *J. Proteome Res.*, 10:34–45, Jan 2011.

[11] J. N. Adkins, S. M. Varnum, K. J. Auberry, R. J. Moore, N. H. Angell, R. D. Smith, D. L. Springer, and J. G. Pounds. Toward a human blood serum proteome: analysis by multidimensional separation coupled with mass spectrometry. *Mol. Cell Proteomics*, 1:947–955, Dec 2002.

[12] D. L. Rothemund, V. L. Locke, A. Liew, T. M. Thomas, V. Wasinger, and D. B. Rylatt. Depletion of the highly abundant protein albumin from human plasma using the Gradiflow. *Proteomics*, 3:279–287, Mar 2003.

[13] Y. Y. Chen, S. Y. Lin, Y. Y. Yeh, H. H. Hsiao, C. Y. Wu, S. T. Chen, and A. H. Wang. A modified protein precipitation procedure for efficient removal of albumin from serum. *Electrophoresis*, 26:2117–2127, Jun 2005.

[14] Y. Gong, X. Li, B. Yang, W. Ying, D. Li, Y. Zhang, S. Dai, Y. Cai, J. Wang, F. He, and X. Qian. Different immunoaffinity fractionation strategies to characterize the human plasma proteome. *J. Proteome Res.*, 5:1379–1387, Jun 2006.

[15] B. R. Fonslow, P. C. Carvalho, K. Academia, S. Freeby, T. Xu, A. Nakorchevsky, A. Paulus, and J. R. Yates. Improvements in Proteomic Metrics of Low Abundance Proteins through Proteome Equalization Using ProteoMiner Prior to MudPIT. *J Proteome Res*, Jun 2011.

[16] H. Steen and M. Mann. The ABC's (and XYZ's) of peptide sequencing. *Nat. Rev. Mol. Cell Biol.*, 5(9):699–711, Sep 2004.

[17] A. Zerck, E. Nordhoff, A. Resemann, E. Mirgorodskaya, D. Suckau, K. Reinert, H. Lehrach, and J. Gobom. An iterative strategy for precursor ion selection for LC-MS/MS based shotgun proteomics. *J. Proteome Res.*, 8: 3239–3251, Jul 2009.

[18] A. Zerck, E. Nordhoff, H. Lehrach, and K. Reinert. Optimal precursor ion selection for LC-MALDI MS/MS. *BMC Bioinformatics*, 14(1):56, Feb 2013.

[19] F. W. McLafferty. Tandem mass spectrometry. *Science*, 214:280–287, Oct 1981.

[20] R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422:198–207, Mar 2003.

[21] B. F. Cravatt, G. M. Simon, and J. R. Yates. The biological impact of mass-spectrometry-based proteomics. *Nature*, 450:991–1000, Dec 2007.

[22] C. M. Whitehouse, R. N. Dreyer, M. Yamashita, and J. B. Fenn. Electrospray interface for liquid chromatographs and mass spectrometers. *Anal. Chem.*, 57:675–679, Mar 1985.

[23] M. Dole, L. L. Mack, R. L. Hines, R. C. Mobley, L. D. Ferguson, and M. B. Alice. Molecular Beams of Macroions. *J. Chem. Phys.*, 49:2240–2249, Sep 1968.

[24] M. Karas and F. Hillenkamp. Laser desorption ionization of proteins with molecular masses exceeding 10,000 daltons. *Anal. Chem.*, 60:2299–2301, Oct 1988.

[25] K. Tanaka, H. Waki, Y. Ido, S. Akita, Y. Yoshida, and T. Yoshida. Protein and polymer analyses up to m/z 100 000 by laser ionization time-of-flight mass spectrometry. *Rapid Commun. Mass Spectrom.*, 2:151–153, Aug 1988.

[26] R. J. Cotter. *Time-of-Flight Mass Spectrometry.* ACS Professional Reference Books, 1997.

[27] B. Canas, D. Lopez-Ferrer, A. Ramos-Fernandez, E. Camafeita, and E. Calvo. Mass spectrometry technologies for proteomics. *Brief Funct Genomic Proteomic*, 4:295–320, Feb 2006.

[28] F. Suits, B. Hoekman, T. Rosenling, R. Bischoff, and P. Horvatovich. Threshold-avoiding proteomics pipeline. *Anal. Chem.*, 83:7786–7794, Oct 2011.

[29] J. Seidler, N. Zinn, M. E. Boehm, and W. D. Lehmann. De novo sequencing of peptides by MS/MS. *Proteomics*, 10:634–649, Feb 2010.

[30] L. Sleno and D. A. Volmer. Ion activation methods for tandem mass spectrometry. *J Mass Spectrom*, 39:1091–1112, Oct 2004.

[31] P. Roepstorff and J. Fohlman. Proposal for a common nomenclature for sequence ions in mass spectra of peptides. *Biomed. Mass Spectrom.*, 11:601, Nov 1984.

[32] M. M. Savitski, M. L. Nielsen, F. Kjeldsen, and R. A. Zubarev. Proteomics-grade de novo sequencing approach. *J. Proteome Res.*, 4:2348–2354, 2005.

[33] A. Bertsch, A. Leinenbach, A. Pervukhin, M. Lubeck, R. Hartmer, C. Baessmann, Y. A. Elnakady, R. Muller, S. Bocker, C. G. Huber, and O. Kohlbacher. De novo peptide sequencing by tandem MS using complementary CID and electron transfer dissociation. *Electrophoresis*, 30: 3736–3747, Nov 2009.

[34] F. Kjeldsen, O. A. Silivra, I. A. Ivonin, K. F. Haselmann, M. Gorshkov, and R.A. Zubarev. C alpha-C backbone fragmentation dominates in electron

detachment dissociation of gas-phase polypeptide polyanions. *Chemistry*, 11:1803–1812, Mar 2005.

[35] J. K. Eng, A. L. McCormack, and J. R. Yates. An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database . *J. Am. Soc. Mass Spectrom.*, 5:976–989, 1994.

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

[37] R. Craig and R. C. Beavis. TANDEM: matching proteins with tandem mass spectra. *Bioinformatics*, 20:1466–1467, Jun 2004.

[38] L. Y. Geer, S. P. Markey, J. A. Kowalak, L. Wagner, M. Xu, D. M. Maynard, X. Yang, W. Shi, and S. H. Bryant. Open mass spectrometry search algorithm. *J. Proteome Res.*, 3:958–964, 2004.

[39] E. A. Kapp, F. Schutz, L. M. Connolly, J. A. Chakel, J. E. Meza, C. A. Miller, D. Fenyo, J. K. Eng, J. N. Adkins, G. S. Omenn, and R. J. Simpson. An evaluation, comparison, and accurate benchmarking of several publicly available MS/MS search algorithms: sensitivity and specificity analysis. *Proteomics*, 5:3475–3490, Aug 2005.

[40] E. Kapp and F. Schutz. Overview of tandem mass spectrometry (MS/MS) database search algorithms. *Curr Protoc Protein Sci*, Chapter 25:Unit25.2, Aug 2007.

[41] J. A. Taylor and R. S. Johnson. Sequence database searches via de novo peptide sequencing by tandem mass spectrometry. *Rapid Commun. Mass Spectrom.*, 11:1067–1075, 1997.

[42] J. A. Taylor and R. S. Johnson. Implementation and uses of automated de novo peptide sequencing by tandem mass spectrometry. *Anal. Chem.*, 73: 2594–2604, Jun 2001.

[43] J. Fernandez-de Cossio, J. Gonzalez, L. Betancourt, V. Besada, G. Padron, Y. Shimonishi, and T. Takao. Automated interpretation of high-energy collision-induced dissociation spectra of singly protonated peptides by 'SeqMS', a software aid for de novo sequencing by tandem mass spectrometry. *Rapid Commun. Mass Spectrom.*, 12:1867–1878, 1998.

[44] J. Fernandez-de Cossio, J. Gonzalez, Y. Satomi, T. Shima, N. Okumura, V. Besada, L. Betancourt, G. Padron, Y. Shimonishi, and T. Takao. Automated interpretation of low-energy collision-induced dissociation spectra by SeqMS, a software aid for de novo sequencing by tandem mass spectrometry. *Electrophoresis*, 21:1694–1699, May 2000.

[45] A. Frank and P. Pevzner. PepNovo: de novo peptide sequencing via probabilistic network modeling. *Anal. Chem.*, 77:964–973, Feb 2005.

[46] S. Pevtsov, I. Fedulova, H. Mirzaei, C. Buck, and X. Zhang. Performance evaluation of existing de novo sequencing algorithms. *J. Proteome Res.*, 5: 3018–3028, Nov 2006.

[47] M. Sturm and O. Kohlbacher. TOPPView: an open-source viewer for mass spectrometry data. *J. Proteome Res.*, 8:3760–3763, Jul 2009.

[48] L. Bianco, J. Mead, and C. Bessant. Comparison of novel decoy database designs for optimizing protein identification searches using ABRF sPRG2006 standard MS/MS datasets. *J. Proteome Res.*, Feb 2009.

[49] L. Käll, J. D. Storey, M. J. MacCoss, and W. S. Noble. Assigning significance to peptides identified by tandem mass spectrometry using decoy databases. *J. Proteome Res.*, 7:29–34, Jan 2008.

[50] L. Käll, J. D. Storey, M. J. MacCoss, and W. S. Noble. Posterior error probabilities and false discovery rates: two sides of the same coin. *J. Proteome Res.*, 7:40–44, Jan 2008.

[51] J. D. Storey and R. Tibshirani. Statistical significance for genomewide studies. *Proc. Natl. Acad. Sci. U.S.A.*, 100:9440–9445, Aug 2003.

[52] J. E. Elias and S. P. Gygi. Target-decoy search strategy for increased confidence in large-scale protein identifications by mass spectrometry. *Nat. Methods*, 4:207–214, Mar 2007.

[53] M. Fitzgibbon, Q. Li, and M. McIntosh. Modes of inference for evaluating the confidence of peptide identifications. *J. Proteome Res.*, 7:35–39, Jan 2008.

[54] H. Choi and A. I. Nesvizhskii. False discovery rates and related statistical concepts in mass spectrometry-based proteomics. *J. Proteome Res.*, 7:47–50, Jan 2008.

[55] V. Granholm and L. Käll. Quality assessments of peptide-spectrum matches in shotgun proteomics. *Proteomics*, 11:1086–1093, Mar 2011.

[56] S. Kim, N. Gupta, and P. A. Pevzner. Spectral probabilities and generating functions of tandem mass spectra: a strike against decoy databases. *J. Proteome Res.*, 7:3354–3363, Aug 2008.

[57] B. Y. Renard, W. Timm, M. Kirchner, J. A. Steen, F. A. Hamprecht, and H. Steen. Estimating the confidence of peptide identifications without decoy databases. *Anal. Chem.*, 82:4314–4318, Jun 2010.

[58] S. Nahnsen, A. Bertsch, J. Rahnenfuhrer, A. Nordheim, and O. Kohlbacher. Probabilistic consensus scoring improves tandem mass spectrometry peptide identification. *J. Proteome Res.*, 10:3332–3343, Aug 2011.

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

[60] O. Kohlbacher, K. Reinert, C. Gröpl, E. Lange, N. Pfeifer, O. Schulz-Trieglaff, and M. Sturm. TOPP–the OpenMS proteomics pipeline. *Bioinformatics*, 23(2):e191–197, 2007. doi: 10.1093/bioinformatics/btl299. URL `http://bioinformatics.oxfordjournals.org/cgi/content/abstract/23/2/e191`.

[61] B. C. Searle. Peptideprophet Explained. `http://www.proteomesoftware.com/pdf_files/peptide_prophet_edited.pdf`, 2009. [Online; accessed 02-February-2012].

[62] A. I. Nesvizhskii and R. Aebersold. Interpretation of shotgun proteomic data: the protein inference problem. *Mol. Cell Proteomics*, 4:1419–1440, Oct 2005.

[63] S. Carr, R. Aebersold, M. Baldwin, A. Burlingame, K. Clauser, and A. Nesvizhskii. The need for guidelines in publication of peptide and protein identification data: Working Group on Publication Guidelines for Peptide and Protein Identification Data. *Mol. Cell Proteomics*, 3:531–533, Jun 2004.

[64] K. Cottingham. Two are not always better than one. *J. Proteome Res.*, 8: 4172, Sep 2009.

[65] R. Higdon and E. Kolker. A predictive model for identifying proteins by a single peptide match. *Bioinformatics*, 23:277–280, Feb 2007.

[66] N. Gupta and P. A. Pevzner. False discovery rates of protein identifications: a strike against the two-peptide rule. *J. Proteome Res.*, 8:4173–4181, Sep 2009.

[67] D. B. Weatherly, J. A. Atwood, T. A. Minning, C. Cavola, R. L. Tarleton, and R. Orlando. A Heuristic method for assigning a false-discovery rate for protein identifications from Mascot database search results. *Mol. Cell Proteomics*, 4:762–772, Jun 2005.

[68] L. Reiter, M. Claassen, S. P. Schrimpf, M. Jovanovic, A. Schmidt, J. M. Buhmann, M. O. Hengartner, and R. Aebersold. Protein identification false discovery rates for very large proteomics data sets generated by tandem mass spectrometry. *Mol. Cell Proteomics*, 8:2405–2417, Nov 2009.

[69] A. Ramos-Fernandez, A. Paradela, R. Navajas, and J. P. Albar. Generalized method for probability-based peptide and protein identification from tandem mass spectrometry data and sequence database searching. *Mol. Cell Proteomics*, 7:1748–1754, Sep 2008.

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

[71] Y. F. Li, R. J. Arnold, Y. Li, P. Radivojac, Q. Sheng, and H. Tang. A bayesian approach to protein inference problem in shotgun proteomics. *J. Comput. Biol.*, 16:1183–1193, Aug 2009.

[72] P. Alves, R. J. Arnold, M. V. Novotny, P. Radivojac, J. P. Reilly, and H. Tang. Advancement in protein inference from shotgun proteomics using peptide detectability. *Pac Symp Biocomput*, pages 409–420, 2007.

[73] A. A. Klammer, X. Yi, M. J. MacCoss, and W. S. Noble. Improving tandem mass spectrum identification using peptide retention time prediction across diverse chromatography conditions. *Anal. Chem.*, 79:6111–6118, Aug 2007.

[74] N. Pfeifer, A. Leinenbach, C. G. Huber, and O. Kohlbacher. Statistical learning of peptide retention behavior in chromatographic separations: a new kernel-based approach for computational proteomics. *BMC Bioinformatics*, 8, 2007.

[75] L. Moruz, D. Tomazela, and L. Käll. Training, selection, and robust calibration of retention time models for targeted proteomics. *J. Proteome Res.*, 9:5209–5216, Oct 2010.

[76] K. Petritis, L. J. Kangas, P. L. Ferguson, G. A. Anderson, L. Paša-Tolić, M. S. Lipton, K. J. Auberry, E. F. Strittmatter, Y. Shen, R. Zhao, and R. D. Smith. Use of artificial neural networks for the accurate prediction of peptide liquid chromatography elution times in proteome analyses. *Anal. Chem.*, 75:1039–1048, Mar 2003.

[77] O. Schulz-Trieglaff, N. Pfeifer, C. Gropl, O. Kohlbacher, and K. Reinert. LC-MSsim–a simulation software for liquid chromatography mass spectrometry data. *BMC Bioinformatics*, 9:423, 2008.

[78] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, Belmont, Massachusetts, 1997.

[79] R. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[80] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 2001.

[81] C. C. Wong, D. Cociorva, J. D. Venable, T. Xu, and J. R. Yates. Comparison of different signal thresholds on data dependent sampling in Orbitrap and LTQ mass spectrometry for the identification of peptides and proteins in complex mixtures. *J. Am. Soc. Mass Spectrom.*, 20:1405–1414, Aug 2009.

[82] E. L. Rudomin, S. A. Carr, and J. D. Jaffe. Directed sample interrogation utilizing an accurate mass exclusion-based data-dependent acquisition strategy (AMEx). *J. Proteome Res.*, 8:3154–3160, Jun 2009.

[83] J. P. M. Hui, S. Tessier, H. Butler, B. Jonathan, P. Kearney, A. Carrier, and P. Thibault. In *Proceedings of the 51st ASMS Conference on Mass Spectrometry and Allied Topics*, Montreal, Quebec, Canada, 2003.

[84] H.-S. Chen, T. Rejtar, V. Andreev, E. Moskovets, and B. L. Karger. Enhanced characterization of complex proteomic samples using lc-maldi ms/ms : Exclusion of redundant peptides from ms/ms analysis in replicate runs. *Anal Chem*, 77:7816–7825, 2005.

[85] N. Wang, J. Zheng, R. Whittal, and L. Li. In *Proceedings of the 54th ASMS Conference on Mass Spectrometry and Allied Topics*, Seattle, WA, 2006.

[86] N. Wang and L. Li. Exploring the Precursor Ion Exclusion Feature of Liquid Chromatography-Electrospray Ionization Quadrupole Time-of-Flight Mass Spectrometry for Improving Protein Identification in Shotgun Proteome Analysis. *Anal. Chem.*, 80:4696–4710, Jun 2008.

[87] S. C. Bendall, C. Hughes, J. L. Campbell, M. H. Stewart, P. Pittock, S. Liu, E. Bonneil, P. Thibault, M. Bhatia, and G. A. Lajoie. An enhanced mass spectrometry approach reveals human embryonic stem cell growth factors in culture. *Mol. Cell Proteomics*, 8(3):421–432, Mar 2009.

[88] M. Claassen, R. Aebersold, and J. M. Buhmann. Proteome coverage prediction with infinite Markov models. *Bioinformatics*, 25:i154–160, Jun 2009.

[89] A. Schmidt, M. Claassen, and R. Aebersold. Directed mass spectrometry: towards hypothesis-driven proteomics. *Curr Opin Chem Biol*, 13:510–517, Dec 2009.

[90] O. Rinner, L. N. Mueller, M. Hubalek, M. Müller, M. Gstaiger, and R. Aebersold. An integrated mass spectrometric and computational framework for the analysis of protein interaction networks. *Nat. Biotechnol.*, 25:345–352, Mar 2007.

[91] P. Picotti, R. Aebersold, and B. Domon. The implications of proteolytic background for shotgun proteomics. *Mol. Cell Proteomics*, 6:1589–1598, Sep 2007.

[92] A. Schmidt, N. Gehlenborg, B. Bodenmiller, L. N. Mueller, D. Campbell, M. Mueller, R. Aebersold, and B. Domon. An integrated, directed mass spectrometric approach for in-depth characterization of complex peptide mixtures. *Mol. Cell Proteomics*, 7:2138–2150, Nov 2008.

[93] T. Gandhi, F. Fusetti, E. Wiederhold, R. Breitling, B. Poolman, and H. P. Permentier. Apex peptide elution chain selection: a new strategy for selecting precursors in 2D-LC-MALDI-TOF/TOF experiments on complex biological samples. *J. Proteome Res.*, 9:5922–5928, Nov 2010.

[94] M. R. Hoopmann, G. E. Merrihew, P. D. von Haller, and M. J. MacCoss. Post analysis data acquisition for the iterative MS/MS sampling of proteomics mixtures. *J. Proteome Res.*, 8:1870–1875, Apr 2009.

[95] C. Sandhu, J. A. Hewel, G. Badis, S. Talukder, J. Liu, T. R. Hughes, and A. Emili. Evaluation of data-dependent versus targeted shotgun proteomic approaches for monitoring transcription factor expression in breast cancer. *J. Proteome Res.*, 7:1529–1541, Apr 2008.

[96] J. D. Jaffe, H. Keshishian, B. Chang, T. A. Addona, M. A. Gillette, and S. A. Carr. Accurate inclusion mass screening: a bridge from unbiased discovery to targeted assay development for biomarker verification. *Mol. Cell Proteomics*, 7:1952–1962, Oct 2008.

[97] S. J. Hattan and K. C. Parker. Methodology utilizing MS signal intensity and LC retention time for quantitative analysis and precursor ion selection in proteomic LC-MALDI analyses. *Anal. Chem.*, 78:7986–7996, Dec 2006.

[98] H. Neubert, T. P. Bonnert, K. Rumpel, B. T. Hunt, E. S. Henle, and I. T. James. Label-free detection of differential protein expression by LC/MALDI mass spectrometry. *J. Proteome Res.*, 7:2270–2279, Jun 2008.

[99] W. Yan, J. Luo, M. Robinson, J. Eng, R. H. Aebersold, and J. Ranish. Index-ion triggered MS2 Ion quantification: A novel proteomics approach for reproducible detection and quantification of targeted proteins in complex mixtures. *Mol Cell Proteomics*, Dec 2010.

[100] A. Schmidt, M. Beck, J. Malmstrom, H. Lam, M. Claassen, D. Campbell, and R. Aebersold. Absolute quantification of microbial proteomes at different states by directed mass spectrometry. *Mol. Syst. Biol.*, 7:510, 2011.

[101] S. Purvine, J. T. Eppel, E. C. Yi, and D. R. Goodlett. Shotgun collision-induced dissociation of peptides using a time of flight mass analyzer. *Proteomics*, 3:847–850, Jun 2003.

[102] J. D. Venable, M. Q. Dong, J. Wohlschlegel, A. Dillin, and J. R. Yates. Automated approach for quantitative analysis of complex peptide mixtures from tandem mass spectra. *Nat. Methods*, 1:39–45, Oct 2004.

[103] A. A. Ramos, H. Yang, L. E. Rosen, and X. Yao. Tandem parallel fragmentation of peptides for mass spectrometry. *Anal. Chem.*, 78:6391–6397, Sep 2006.

[104] A. B. Chakraborty, S. J. Berger, and J. C. Gebler. Use of an integrated MS–multiplexed MS/MS data acquisition strategy for high-coverage peptide mapping studies. *Rapid Commun. Mass Spectrom.*, 21:730–744, 2007.

[105] J. W. Wong, A. B. Schwahn, and K. M. Downard. ETISEQ–an algorithm for automated elution time ion sequencing of concurrently fragmented peptides for mass spectrometry-based proteomics. *BMC Bioinformatics*, 10: 244, 2009.

[106] R. K. Blackburn, F. Mbeunkui, S. K. Mitra, T. Mentzel, and M. B. Goshe. Improving Protein and Proteome Coverage through Data-Independent Multiplexed Peptide Fragmentation. *J Proteome Res*, May 2010.

[107] M. Bern, G. Finney, M. R. Hoopmann, G. Merrihew, M. J. Toth, and M. J. MacCoss. Deconvolution of mixture spectra from ion-trap data-independent-acquisition tandem mass spectrometry. *Anal. Chem.*, 82:833–841, Feb 2010.

[108] S. J. Geromanos, J. P. Vissers, J. C. Silva, C. A. Dorschel, G. Z. Li, M. V. Gorenstein, R. H. Bateman, and J. I. Langridge. The detection, correlation, and comparison of peptide precursor and product ions from data independent LC-MS with data dependant LC-MS/MS. *Proteomics*, 9:1683–1695, Mar 2009.

[109] A. Scherl, P. Francois, V. Converset, M. Bento, J. A. Burgess, J.-C. Sanchez, D. F. Hochstrasser, J. Schrenzel, and G. L. Corthals. Nonredundant mass spectrometry: A strategy to integrate mass spectrometry acquisition and analysis. *Proteomics*, 4:917–927, 2004.

[110] H. Liu, L. Yang, N. Khainovski, M. Dong, S. C. Hall, S. J. Fisher, M. D. Biggin, J. Jin, and H. E. Witkowska. Automated Iterative MS/MS Acquisition: A Tool for Improving Efficiency of Protein Identification Using a LC-MALDI MS Workflow. *Anal Chem*, 83(16):6286–6293, Aug 2011.

[111] J. Graumann, R. A. Scheltema, Y. Zhang, J. Cox, and M. Mann. A framework for intelligent data acquisition and real-time database searching for shotgun proteomics. *Mol Cell Proteomics*, Dec 2011.

[112] D. J. Bailey, C. M. Rose, G. C. McAlister, J. Brumbaugh, P. Yu, C. D. Wenger, M. S. Westphall, J. A. Thomson, and J. J. Coon. Instant spectral assignment for advanced decision tree-driven mass spectrometry. *Proc. Natl. Acad. Sci. U.S.A.*, 109(22):8411–8416, May 2012.

[113] J. Cox and M. Mann. MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. *Nat. Biotechnol.*, 26:1367–1372, Dec 2008.

[114] J. Cox, N. Neuhauser, A. Michalski, R. A. Scheltema, J. V. Olsen, and M. Mann. Andromeda: a peptide search engine integrated into the MaxQuant environment. *J. Proteome Res.*, 10:1794–1805, Apr 2011.

[115] U. Bommer, N. Burkhardt, R. Jünemann, C. M. T. Spahn, F. J. Triana-Alonso, and K. H. Nierhaus. *Subcellular Fractionation: A Practical Approach*, pages 271–301. IRL Press, Oxford, 1997.

[116] E. Mirgorodskaya, C. Braeuer, P. Fucini, H. Lehrach, and J. Gobom. Nanoflow liquid chromatography coupled to matrix-assisted laser desorption/ionization mass spectrometry: Sample preparation, data analysis, and application to the analysis of complex peptide mixtures. *Proteomics*, 5: 399–408, 2005.

[117] N. Pfeifer. *Kernel-based Machine Learning on Sequence Data from Proteomics and Immunomics*. PhD thesis, Eberhard-Karls-Universität Tübingen, Tübingen, Germany, 2009.

[118] V. Vacic, L. M. Iakoucheva, and P. Radivojac. Two Sample Logo: a graphical representation of the differences between two sets of sequence alignments. *Bioinformatics*, 22:1536–1537, Jun 2006.

[119] E. Giralt, M.-L. Valero, and D. Andreu. An evaluation of some structural determinants for peptide desorption in MALDI-TOF mass spectrometry. In *Peptides 1996*, pages 855–856. Mayflower Scientific Ltd., 1998.

[120] B. Zhang, M. C. Chambers, and D. L. Tabb. Proteomic parsimony through bipartite graph analysis improves accuracy and transparency. *J. Proteome Res.*, 6(9):3549–3557, Sep 2007.

[121] V. R. Koskinen, P. A. Emery, D. M. Creasy, and J. S. Cottrell. Hierarchical clustering of shotgun proteomics data. *Mol. Cell Proteomics*, 10(6): M110.003822, Jun 2011.

[122] A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[123] S. Albers. Online algorithms. In D. Goldin, S. A. Smolka, and P. Wegner, editors, *Interactive Computation*, pages 143–164. Springer Berlin Heidelberg, 2006. ISBN 978-3-540-34874-0. URL `http://dx.doi.org/10.1007/3-540-34874-3_7`.

[124] J. Junker, C. Bielow, A. Bertsch, M. Sturm, K. Reinert, and O. Kohlbacher. TOPPAS: A Graphical Workflow Editor for the Analysis of High-Throughput Proteomics Data. *J. Proteome Res.*, 11(7):3914–3920, Jul 2012.

[125] C. Bielow, S. Aiche, S. Andreotti, and K. Reinert. MSSimulator: Simulation of mass spectrometry data. *J. Proteome Res.*, 10(7):2922–2929, Jul 2011.

[126] L. C. Gillet, P. Navarro, S. Tate, H. Rost, N. Selevsek, L. Reiter, R. Bonner, and R. Aebersold. Targeted data extraction of the MS/MS spectra generated by data-independent acquisition: a new concept for consistent and accurate proteome analysis. *Mol. Cell Proteomics*, 11(6):O111.016717, Jun 2012.

[127] A. Bertsch, S. Jung, A. Zerck, N. Pfeifer, S. Nahnsen, C. Henneges, A. Nordheim, and O. Kohlbacher. Optimal de novo Design of MRM Experiments for Rapid Assay Development in Targeted Proteomics. *J Proteome Res*, Mar 2010.

[128] T. Huang and Z. He. A linear programming model for protein inference problem in shotgun proteomics. *Bioinformatics*, Sep 2012.

[129] W. J. Henzel, T. M. Billeci, J. T. Stults, S. C. Wong, C. Grimley, and C. Watanabe. Identifying proteins from two-dimensional gels by molecular mass searching of peptide fragments in protein sequence databases. *Proc. Natl. Acad. Sci. U.S.A.*, 90(11):5011–5015, Jun 1993.

[130] D. J. Pappin, P. Hojrup, and A. J. Bleasby. Rapid identification of proteins by peptide-mass fingerprinting. *Curr. Biol.*, 3(6):327–332, Jun 1993.

[131] P. James, M. Quadroni, E. Carafoli, and G. Gonnet. Protein identification by mass profile fingerprinting. *Biochem. Biophys. Res. Commun.*, 195(1): 58–64, Aug 1993.

[132] J. R. Yates, S. Speicher, P. R. Griffin, and T. Hunkapiller. Peptide mass maps: a highly informative approach to protein identification. *Anal. Biochem.*, 214(2):397–408, Nov 1993.

[133] M. Mann, P. Højrup, and P. Roepstorff. Use of mass spectrometric molecular weight information to identify proteins in sequence databases. *Biol. Mass Spectrom.*, 22(6):338–345, Jun 1993.

[134] M. L. Nielsen, M. M. Savitski, and R. A. Zubarev. Extent of modifications in human proteome samples and their effect on dynamic range of analysis in shotgun proteomics. *Mol. Cell Proteomics*, 5(12):2384–2391, Dec 2006.

[135] R. A. Zubarev. The challenge of the proteome dynamic range and its implications for in-depth proteomics. *Proteomics*, Jan 2013.

[136] J. T. Webber, M. Askenazi, S. B. Ficarro, M. A. Iglehart, and J. A. Marto. Library dependent LC-MS/MS acquisition via mzAPI/Live. *Proteomics*, Mar 2013.

Appendix A Data

A.1. RT prediction

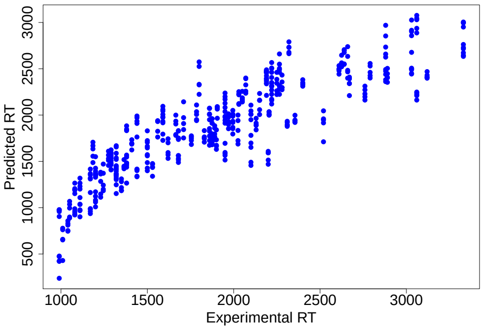


Figure A.1.: Experimental RT vs. predicted RT for the 50s sample.

A.2. PT prediction

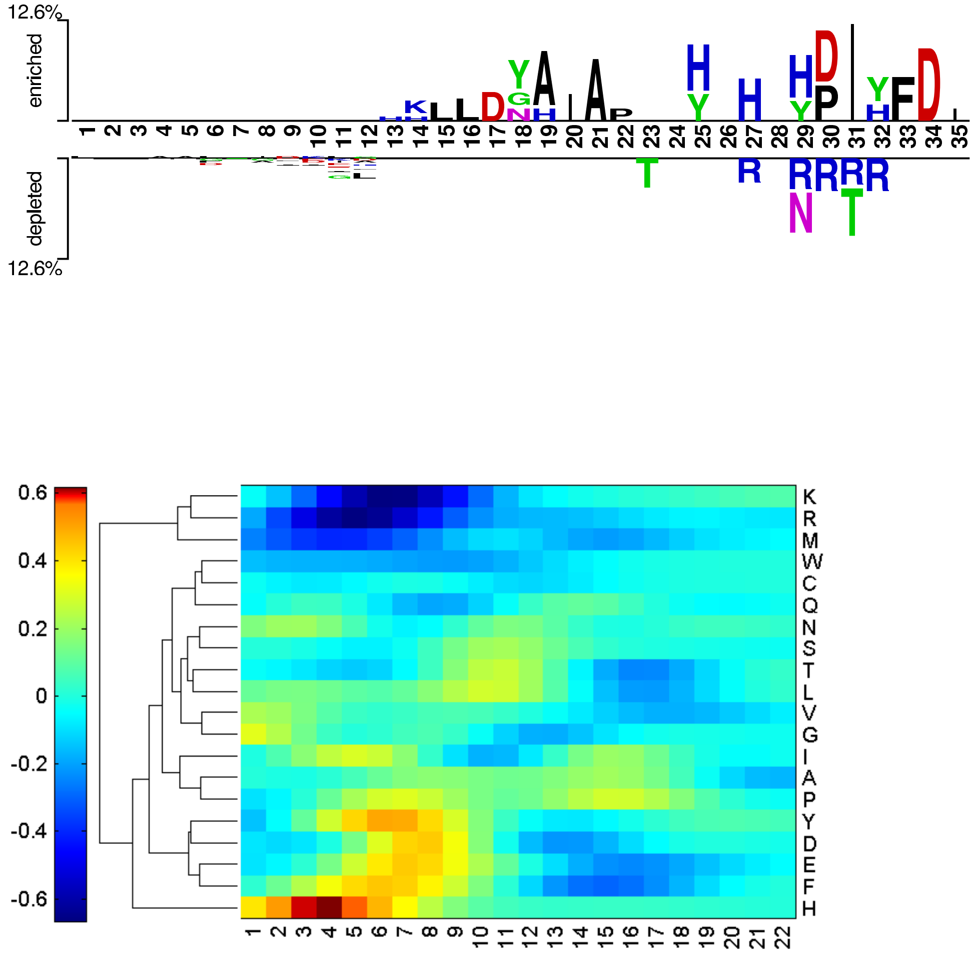


Figure A.2.: PT model evaluation for 50s sample: Two sample logo and heatmap.

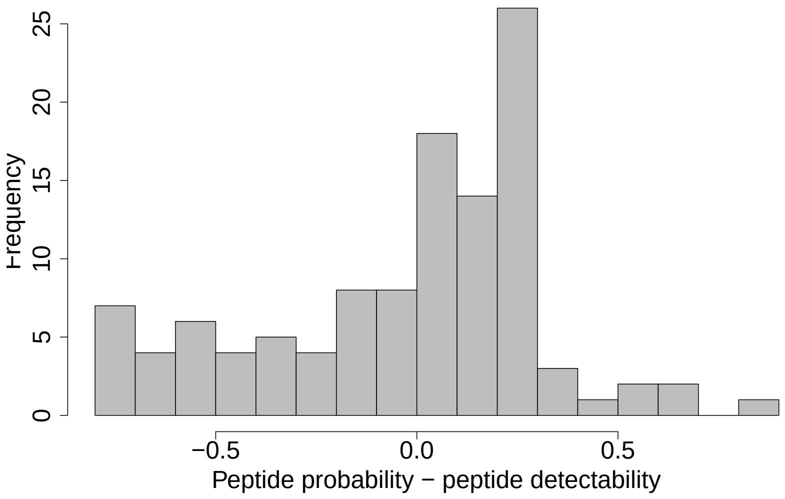


Figure A.3.: PT prediction evaluation for 50s sample: Histogram of differences of peptide probabilities and detectabilities.

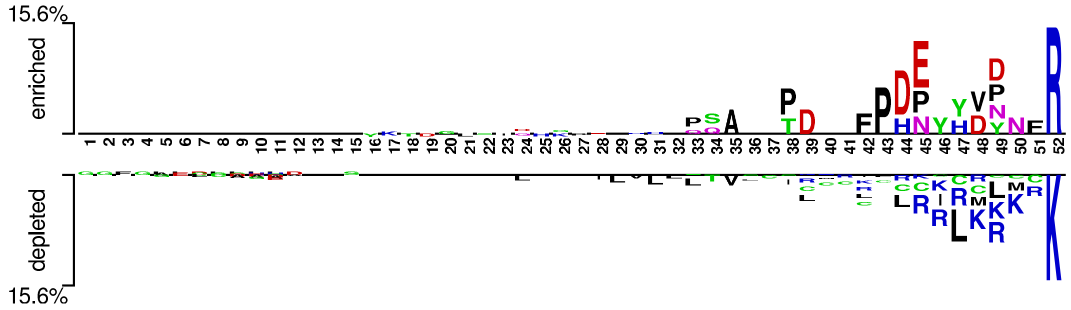


Figure A.4.: Two Sample logo [118] for the high-scoring peptide identifications and the unobserved peptide sequences of the complex dataset. Enriched AAs are shown at the top, depleted AAs at the bottom. The sequences were aligned at their C-Terminus and the position is given with respect to the longest peptide.

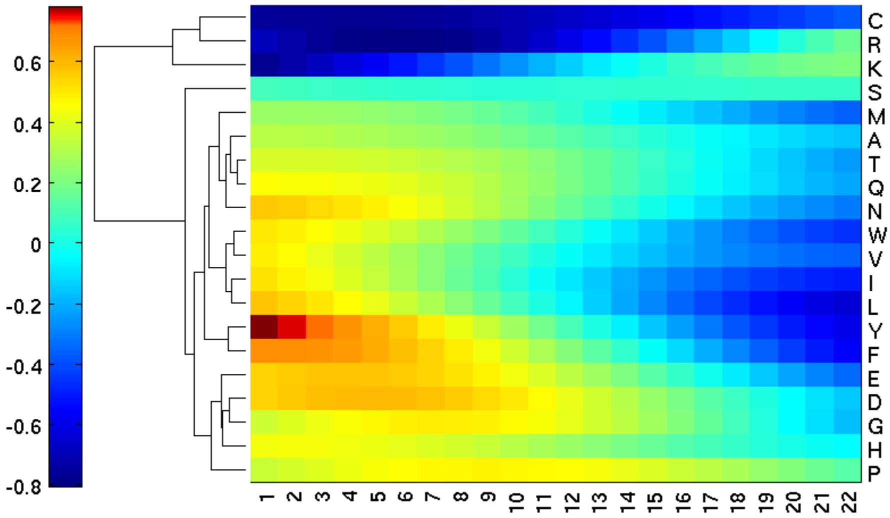


Figure A.5.: Visualization of POBK for complex dataset. Inspired by [117] and produced with MATLAB scripts from Nico Pfeifer. The plot shows the signals for both termini together, hence position i corresponds to AAs at position i and $n - i + 1$ (where n refers to the peptide length).

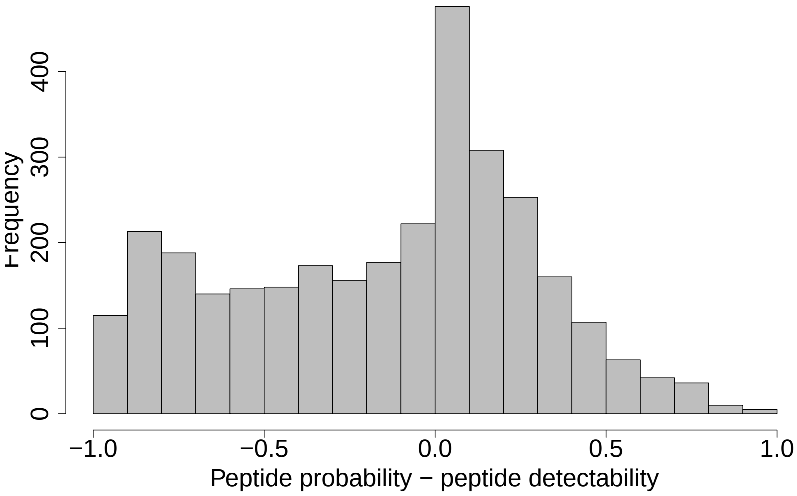


Figure A.6.: Histogram of the difference between peptide probability and predicted detectability for HEK293.

Appendix B Abbrevations

AA	Amino acid
AIMS	accurate inclusion mass screening
BSA	Bovine serum albumin
cdf	Cumulative distribution function
CID	Collision induced dissociation
DDA	Data dependent acquisition
DEX	Dynamic exclusion
EM	Expectation-maximization
ESI	Electrospray Ionization
FDR	False-discovery rate
FWHM	Full-width-at-half-max
GA	Greedy approach
GLPK	GNU Linear Programming Kit
HIPS	Heuristic iterative precursor ion selection
HPLC	High Performance Liquid Chromatography
HSA	Human serum albumin
ILP	Integer Linear Program
IPS	Iterative precursor ion selection
IPS_LP	Iterative precursor ion selection with Linear Programming
ITA	Index-ion Triggered Analysis
LC	Liquid Chromatography
LP	Linear Program
MALDI	Matrix Assisted Laser Desorption/Ionization
MIP	Mixed Integer Program
MRM	Multiple Reaction Monitoring
MS	Mass Spectrometry
MS/MS	Tandem Mass Spectrometry
m/z	mass-to-charge ratio
OPT	Optimal solution
PEP	Posterior error probability
PMF	Peptide mass fingerprinting
POBK	Paired Oligo-Border Kernel
ppm	parts-per-million
PSM	Peptide-spectrum match

PT	Proteotypicity or detectability
PTM	Posttranslational modification
RT	Retention Time
SNR	signal-to-noise ratio
SPS	Static precursor ion selection
SVM	Support Vector Machine
SVR	Support Vector Regression
TOF	Time-of-flight
TOPP	The OpenMS Proteomics Pipeline
TSL	Two Sample Logo
UPS	Universal proteomics standard

Selbständigkeitserklärung

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel in Anspruch genommen habe. Ich versichere, dass diese Arbeit in dieser oder anderer Form keiner anderen Prüfungsbehörde vorgelegt wurde.

____

Alexandra Zerck
Berlin, Mai 2013

Curriculum Vitae

For privacy reasons, the curriculum vitae is not contained in the online version of this thesis.

For privacy reasons, the curriculum vitae is not contained in the online version of this thesis.