Quantification and Simulation of Liquid Chromatography-Mass Spectrometry Data

Freie Universität Berlin

Dissertation zur Erlangung des Grades
eines Doktors der Naturwissenschaften (Dr. rer. nat.)
am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

von

Chris Bielow

Berlin
2012

Datum der Disputation:

29.10.2012

Dekan:

Professor Dr. Rupert Klein

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

Der Mensch hat dreierlei Wege, klug zu handeln:
Erstens durch Nachdenken, das ist der edelste;
zweitens durch Nachahmen, das ist der leichteste;
drittens durch Erfahrung, das ist der bitterste.

Kung Fu Tse (551-479 v. Chr.), chinesischer Philosoph.

Abstract

Computational mass spectrometry is a fast evolving field that has attracted increased attention over the last couple of years. The performance of software solutions determines the success of analysis to a great extent. New algorithms are required to reflect new experimental procedures and deal with new instrument generations.

One essential component of algorithm development is the validation (as well as comparison) of software on a broad range of data sets. This requires a gold standard (or so-called ground truth), which is usually obtained by manual annotation of a real data set. Comprehensive manually annotated public data sets for mass spectrometry data are labor-intensive to produce and their quality strongly depends on the skill of the human expert. Some parts of the data may even be impossible to annotate due to high levels of noise or other ambiguities. Furthermore, manually annotated data is usually not available for all steps in a typical computational analysis pipeline. We thus developed the most comprehensive simulation software to date, which allows to generate multiple levels of ground truth and features a plethora of settings to reflect experimental conditions and instrument settings. The simulator is used to generate several distinct types of data. The data are subsequently employed to evaluate existing algorithms. Additionally, we employ simulation to determine the influence of instrument attributes and sample complexity on the ability of algorithms to recover information. The results give valuable hints on how to optimize experimental setups.

Furthermore, this thesis introduces two quantitative approaches, namely a decharging algorithm based on integer linear programming and a new workflow for identification of differentially expressed proteins for a large in vitro study on toxic compounds. Decharging infers the uncharged mass of a peptide (or protein) by clustering all its charge variants. The latter occur frequently under certain experimental conditions. We employ simulation to show that decharging is robust against missing values even for high complexity data and that the algorithm outperforms other solutions in terms of mass accuracy and run time on real data. The last part of this thesis deals with a new state-of-the-art workflow for protein quantification based on isobaric tags for relative and absolute quantitation (iTRAQ). We devise a new approach to isotope correction, propose an experimental design, introduce new metrics of iTRAQ data quality, and confirm putative properties of iTRAQ data using a novel approach.

All tools developed as part of this thesis are implemented in OpenMS, a C++ library for computational mass spectrometry.

Zusammenfassung

Rechnergestützte Massenspektrometrie steht seit Jahren im Fokus von Forschungsbestrebungen und erlangt immer mehr Aufmerksamkeit. Die Güte von Software bestimmt zu einem erheblichen Teil den Erfolg oder Misserfolg einer Datenanalyse. Neue experimentelle Möglichkeiten und Instrumentengenerationen erfordern die Anpassung bzw. Neuentwicklung von Algorithmen.

Ein essentieller Gesichtspunkt der Algorithmenentwicklung ist die Validierung (oder auch der Vergleich) von Software auf einer möglichst großen Bandbreite an Eingabedaten. Eine Validierung erfordert einen Goldstandard, der meist durch manuelle Annotation eines Datensatzes erzeugt wird. Umfassende manuell annotierte, öffentliche Datensätze für Massenspektrometrie sind zeitaufwändig in der Herstellung und ihre Qualität hängt stark von den Fähigkeiten des Experten ab. Nicht alle Teile des Datensatzes sind annotierbar, da es teilweise hohe Rauschpegel und andere Störquellen gibt die eine zuverlässige Annotation verhindern. Weiterhin sind manuell annotierte Datensätze üblicherweise nicht für alle Ebenen eines Goldstandards verfügbar. Um dieses Dilemma zu beheben entwickelten wir die zurzeit umfassendste Simulationssoftware, welche viele Ebenen eines Goldstandards unterstützt, ebenso wie eine Vielzahl von Einstellungen, die es erlauben, viele experimentelle Bedingungen und Instrumenteneinstellungen nachzubilden. Der Simulator wird benutzt um mehrere verschiedenartige Datensätze zu erzeugen. Diese werden anschließend eingesetzt um existierende Algorithmen zu bewerten. Zusätzlich benutzen wir Simulationen um den Einfluss von Instrumenteneigenschaften und Probenkomplexität auf die Güte und Vollständigkeit der von Algorithmen extrahierten Informationen zu bestimmen. Die Ergebnisse geben wertvolle Hinweise für die Optimierung von Versuchsaufbauten.

Zusätzlich führt diese Arbeit zwei quantitative Ansätze ein: einen Decharging-Algorithmus basierend auf ganzzahligen linearen Programmen sowie einen neuen Workflow für die Identifizierung von differentiell exprimierten Proteinen für eine große In-vitro-Studie zur Systemtoxikologie. Decharing inferiert die ungeladene Masse eines Peptids (oder Proteins) durch Clustering aller seiner Ladungsvarianten. Letztere entstehen häufig unter bestimmten experimentellen Bedingungen. Wir verwenden Simulationen, um zu zeigen, dass Decharging robust gegen Datenlücken sogar auf hochkomplexen Datensätzen ist, und dass der Algorithmus anderen Lösungen hinsichtlich der Massengenauigkeit und Laufzeit auf realen Daten überlegen ist. Der letzte Teil der Arbeit widmet sich einem modernen Workflow für Proteinquantifizierung mit Hilfe von iTRAQ (isobaric tags for relative and absolute quantitation). Wir stellen einen neuen Ansatz für Isotopenkorrektur vor, entwerfen ein experimentelles Design, konzipieren neue Metriken für die Datenqualität von iTRAQ-Daten und verifizieren vermutete Eigenschaften dieser Art von Daten anhand von neuen Verfahren.

Alle Softwarewerkzeuge, die als Teil dieser Arbeit entstanden sind, wurden in OpenMS – einer C++-Bibliothek für Massenspektrometrie – implementiert.

Acknowledgments

This thesis would not have been possible without Knut Reinert, who piqued my curiosity in computational mass spectrometry when pursuing my Master thesis and who supported my application for the IMPRS PhD scholarship. He was always available for discussions and provided invaluable guidance throughout the thesis. I am indebted to Oliver Kohlbacher, who not only agreed to review this thesis but has also been an inexhaustible source of knowledge on a broad range of topics, which I frequently relied on.

As one of most important sources of support I'd like to thank Claudia and my parents for their love, patience, encouragement and reassurance during the last years.

Kudos to the development team of OpenMS, especially Marc, Andreas, Stephan, Sandro, Clemens, Sven, and Timo for a stimulating working atmosphere and fruitful collaboration. Special thanks to Knut and Oliver for picking remote and unique retreat locations, and their unrivaled cuisine. Regards to the whole team which made these events always fun.

Thanks to my fellow colleagues in the office, Stephan, Sandro and Dave, who made sure we tested about every XXL burger restaurant in town, and are not only a constant source of information on a broad range of interesting topics, but also increased my knowledge of soccer about a hundredfold. I'd also like to pay tribute to Lars Malmström, Andreas Quandt and Hendrik Weisser for creating an enjoyable working environment and making my stay at the ETH Zürich productive and exciting. Thanks to Gunnar Klau and Sandro Andreotti for fruitful discussions on the decharging algorithm. On the IMPRS side of things, thanks to Hannes Luz and Kirsten Kelleher for organizing retreats, soft-skill courses, and the notorious "Last Wednesday of the Month" events.

I am also indebted to all collaborators from the Predict-IV project, especially Silke Ruzek, Christian Huber, and Paul Jennings, who not only provided me with invaluable data but also got me involved in interesting side projects, and with whom I enjoyed writing papers.

Last but not least I'd like to acknowledge Julia and Anja for proofreading this thesis and ironing out the chips and cracks.

Contents

Chapter 1

Introduction

Synopsis: *We motivate the topic of this thesis: the development and application of algorithms for simulation and quantification for mass spectrometry data analysis. Since mass spectrometry is a fast-moving field, we give an overview of the current state of the art and challenges.*

1.1 Mass Spectrometry-based Proteomics

Proteomics – commonly defined as the study of the ensemble of proteins at a given point in time, especially their expression pattern, structure and function – is one of the key research areas of today enabling us to extend our knowledge of regulation and control in living systems. Many emerging or already established fields dealing with large-scale biological data are designated by adding the suffix "-omics" to previously used terms, e.g., metabonomics or genomics, where ongoing efforts are now focusing on personalized and population aspects [1]. The proteome is thought to capture the cellular processes much closer than, for example, the genome or transcriptome. Regulation on protein level includes post-translational modifications, degradation, transport and (protein) interaction. These regulation steps cannot be adequately described or modeled on the genome- or transcriptome level. Recently it was found that RNA-editing, i.e., the alteration of RNA sequence, is not a seldom process in humans [2] but occurs rather frequently. This discovery shifts our understanding of the central dogma of molecular biology in the sense that translation from DNA to protein is not faithful but merely gives the direction of information transfer, thus implying that one cannot fully explain the protein content of a cell, given genes and their splice variants alone. Not only does RNA-editing lead to diversification of the proteome, but it also implies that protein databases derived from DNA alone cannot contain the complete set of protein sequences.

Mass Spectrometry (MS) coupled to liquid chromatography (LC) is currently the major workhorse in proteomics, due its unparalleled automation and high throughput capabilities. It can be used as a "hypothesis-generating engine" [3] and is increasingly replacing techniques like 2D gels and western blotting, especially in exploratory settings, although the latter are still used for confirmatory experiments. Mass Spectrometry itself is an established technique. The introduction of soft-ionization methods, i.e., electrospray ionization (ESI) as proposed by Fenn [4] and matrix-assisted laser desorption/ionization (MALDI) as invented by Tanaka, made MS applicable to the analysis of biomolecules. Fenn and Tanaka both received the Nobel Prize in Chemistry in 2002. Soft ionization prevents larger biomolecules to break upon ionization and thus allows to measure their intact mass.

In clinical proteomics, LC-MS (and especially MALDI-MS) initially triggered a wide range of exploratory studies but has not found wide applicability yet [5], though some remarkable breakthroughs like early detection of kidney scarring have been achieved [6]. About 200 protein *biomarkers* [7] as approved by the FDA (Food and Drug Administration) are used in clinical practice, but only a small subset of these were initially discovered using mass spectrometry.
This is in parts due to the high standards required for clinical biomarkers (with 80 replicates recommended – see [5]) as well as limited reproducibility and robustness of LC-MS. Protein abundance covers more than nine orders of magnitude in human blood, whereas a mass spectrometer covers a linear *dynamic range* of 2-4 (possibly up to 6) orders of magnitude. Linear dynamic range is defined as the range over which ion signal is linear with analyte concentration [8].

Unfortunately, most biomarkers found to date typically occur at concentrations of several ng/ml, thus evading exploratory mass spectrometry [5]. Also, mass spectrometry currently lacks the standardization protocols required for wide clinical applicability. This is partly due to the fact that MS is much more variable than other omics techniques [9], such as microarrays; thus the effort of finding standard operating procedures (SOPs) is more involved. This does not only apply to experimental procedures but also very much to subsequent computational data analysis.

Advancements and gain of public interest for MS-based proteomics can be attributed to improvements in targeted proteomics and instrument design by increasing sensitivity, mass *resolution*, but also in analytical terms where new *multiplexing* methods have been developed, allowing the concurrent measurement of multiple samples. Finally, computational biology has made progress, providing better models and software in order to analyze the increasing amounts of data generated by high-throughput techniques such as LC-MS. The community has realized the importance of sustainable software solutions, which are accessible to a wide audience. MS excels in certain areas like annotation of post-translational modifications [10], and remarkable discoveries were made possible by MS [3].

Mass Spectrometric Imaging (MSI) [11], which was developed in the late nineties, is becoming more popular as spatial resolution, sample preparation methods and sensitivity are improving.

An alternative technique to LC (or LC-MS), namely conventional 2D gels with staining, has advantages in costs and when it comes to the detection of protein isoforms with diverging modifications but is labor-intensive, has low dynamic range and lacks gel-to-gel reproducibility [12]. Formerly it was thought that LC(-MS) will replace 2D gels. However, it becomes more and more clear that they complement each other [13, 14]. Immuno-assays like the enzyme-linked immunosorbent assay (ELISA) are still in wide use in the clinical setting and during validation phase, where MS is still only beginning to become the method of choice [15].

The role of computational proteomics is becoming more and more important, as data generation is currently vastly outpacing data analysis [16]. For example, several hundred patient samples (i.e., human serum) can be screened by a single MALDI platform in one day [5], a single LC-MS run on a modern *Orbitrap* mass analyzer yields many gigabytes of data per day and public data repositories like PRIDE [17] contain hundreds of millions of spectra[1].

Recently, the field has shifted toward high-throughput analysis of (so far unsequenced) organisms/cells trying to achieve maximum proteome coverage while also attempting to increase knowledge about modification states and sites of proteins. New techniques like *concurrent peptide fragmentation* (MS^{E}) [18], fragmenting all peptides, or *sequential windowed acquisition of all theoretical fragment ion mass spectra* (SWATH-MS) [19], are emerging, aiming to push the limits of identification. This also poses a challenge to algorithm development as for each new technique, new solutions are required to make optimal use of the data and avoid time-consuming and error-prone manual analysis.

Many publications have called for a set of credible benchmark data providing a gold standard [20, 21, 22, 23, 24, 25] which allows to compare and evaluate algorithmic approaches. The premise of the exact nature of this data is not necessarily unified though. Some require a gold standard (so-called ground truth) for lower-level signal processing of raw LC-MS data [20, 22, 23, 24], others require pre-processed data with protein expression values from different conditions [25]. One common solution is the use of small protein mixtures (so-called standard mixtures), which are commercially available and can be used to generate data sets with known protein content. However, these mixtures often lack the complexity inherent to real biological samples, and the data quality (e.g., in terms of contaminants from sample handling) strongly depends on the expertise of the laboratory. Inevitably, such controlled experiments cannot provide the lower levels of a gold standard, e.g., for *peak picking* (a low-level data reduction step), whereas they are useful for peptide identification and quantification problems. Of course, manual annotation can be employed for many levels of ground truth. However, manual annotation is labor-intensive and the quality of the annotation strongly depends on the skill of the human expert. Some

[1] as of PRIDE Basis Statistics at http://www.ebi.ac.uk/pride on 04/22/2012.

parts of the data may even be impossible to annotate due to high levels of noise or other ambiguities. Public data sets with comprehensive manual annotation are usually not available for many levels of ground truth. An orthogonal solution to the problem of a benchmark data set is the use of extensive simulation of mass spectrometry data. This area of research is usually neglected or is given only marginal attention when used to validate an algorithm [20, 23, 24]. The software employed is usually written for this purpose alone and neither published nor extensively described. One exception is the publication of LC-MSsim [26], a simulation software for LC-MS data, which unfortunately is not maintained anymore and lacks some desirable functionality, e.g., simulation of fragment spectra (so called MS/MS, MS^2 or tandem mass spectra) and simulation of *labeled* LC-MS data. Fragment spectra allow to identify the peptide (or protein) by sequence and/or its quantification. Fragments are created by introducing the peptide (or protein) of interest into a collision cell. A labeling procedure allows to discern identical peptides (in terms of sequence and post-translational modifications) from different samples within a single LC-MS experiment (a process called multiplexing) by using chemical or metabolic labeling of some kind to introduce a systematic mass shift. Therefore, peptides from different samples can be identified and quantified concurrently. In order to make simulation available to a wider audience we developed the most comprehensive and convenient simulation software to date, allowing for many levels of ground truth. The simulator features a plethora of settings to reflect experimental conditions and instrument configurations. We show how simulation can be used for algorithm benchmarking and validation. Additionally, we employ simulation to determine the influence of instrument configurations and sample complexity on the ability of algorithms to recover information. The results give valuable hints on how to optimize experimental setups.

Labeling can also be used for higher levels of MS, e.g., MS^2, where different chemical labels can only be discerned upon fragmentation. One example of an MS^2-based multiplexing technique is *isobaric tags for relative and absolute quantitation* (iTRAQ) [27]. Like all multiplexing approaches, iTRAQ allows to save time for sample acquisition as multiple samples are measured concurrently, and has the added benefit of easy assignment of individual peptide abundances to samples; thus, differences in expression between peptides can easily be computed without the need for complex algorithms which try to determine peptide content for each sample. Quantification is more challenging in *label-free* experiments, where each sample is measured separately, and is even more difficult for most MS^1-based labeling techniques. However, iTRAQ suffers from other disadvantages, such as ratio underestimation and isotope impurities, which are caused by impurities in chemical labeling reagents and give rise to artifactual intensity values when corrected for with the conventional method of inverse matrix multiplication as proposed in Shadforth et al. [28]. The second contribution of this thesis is the introduction of a new procedure for isotope correction of iTRAQ-based quantification values, which is more robust for low-intensity quantification values. Additionally we present new metrics of iTRAQ data quality, and confirm putative properties of iTRAQ data using a novel approach. The computational analysis is made readily available as highly automated workflow and was implemented into OpenMS [29] – a C++ library for computational mass spectrometry. In large-scale studies using labeling techniques there is often the problem of finding a suitable *experimental design*, i.e. the allocation of samples to the limited set of available labels. The set of samples usually includes technical or biological replicates. We propose an experimental design, specifically adapted to the requirements of a large-scale project using iTRAQ-based LC-MS.

A common problem, especially in ESI-MS, is the presence of multiple *charge variants*, i.e., a peptide or protein occurs in multiple charge states and possibly with different adducts. This gives rise to so-called charge ladders. This signal partitioning unfortunately leads to dense spectra, but it also allows for multiple measurement points for a single species, which can be used for a more precise mass estimation. An algorithm which can find charge variants can thus not only be applied to find the overall signal contribution originating from a single peptide or protein, thus finding a more clear representation of sample content, but also to determine peptide or protein mass more precisely. Decharging infers the uncharged mass of a peptide (or protein) by clustering all its charge variants. The latter occur frequently during ESI, but can also be observed in MALDI-based experiments under certain conditions [30]. One of the first algorithms by Mann, Meng, and Fenn [31] is able to decharge spectra from whole protein samples, but is prone to dense spectra and likely to create artifact signals. Later algorithms have drawbacks in other areas, e.g., the heuristic approach by Malyarenko et al. [30] is only applicable to MALDI spectra. The well known ZScore-Algorithm [32] supports decharging either using charge ladders or local charge information, but not both simultaneously. An algorithm by Wehofsky and Hoffmann [33] can use both local charge and charge ladders but does not take into account retention time, only considers proton adducts, and requires charge ladders without gaps. None of the algorithms mentioned above is able to model charge ladders with multiple adduct combinations, e.g., a combination of pure proton adduct species with a proton/sodium species from the same peptide or protein. Our decharging algorithm based on integer linear programming (ILP) is suitable for finding pairs of labeled data concurrently with charge variants and supports arbitrary adduct combinations. We employ simulation to show that the algorithm is robust against missing values even for high complexity data and that it outperforms other solutions in terms of mass accuracy and run time on real data. The decharging algorithm constitutes our third contribution.

1.2 Guide to this Thesis

This thesis focuses on the algorithmic aspects of mass spectrometry-based proteomics while also touching on biomarker identification and statistical evaluation.

The remainder of this thesis is structured as follows: The second chapter gives an overview of the wide field of computational proteomics, especially covering quantification of peptide signals from LC-MS samples, while also elucidating our contribution to *OpenMS* [29] – a widely used framework for computational mass spectrometry. We also cover current algorithmic problems, some of which are addressed in the following chapters. Every algorithm described in this thesis was implemented in OpenMS and is part of the official OpenMS release, readily available as binary package from the official website.

In the third chapter, we describe our simulation tool *MSSimulator*; motivate why simulation of MS^1 and MS^2 data provides a valuable tool for algorithm prototyping, benchmarking and experimental setup optimization and describe the capabilities of the simulator as well as the properties of the underlying instruments. This chapter was, in parts, published in Bielow et al. [34].

The fourth chapter introduces a deconvolution algorithm designed to cluster charge variants, differently labeled peptides, and common adducts in MS experiments. We show the theoretical details and multiple applications of this versatile approach. The algorithm is benchmarked using the simulator described in Chapter 4. An earlier version of this algorithm was published in Bielow et al. [35].

The fifth chapter represents our contribution to the analysis of iTRAQ data, including experimental design, a novel isotope correction procedure, and new metrics of iTRAQ data quality. Parts of the analysis along with the biological interpretation are currently being prepared for publication in Wilmes et al. [36].

In Chapter 6 we conclude this thesis with a summary of our work and point to future extensions.

Chapter 2

Computational Mass Spectrometry

Synopsis: *This chapter introduces common terms used in proteomics and especially mass spectrometry with the focus on their algorithmic aspects and serves as a foundation for the following chapters. We also briefly recapitulate the pros and cons of the major existing software packages as well as our contribution to the OpenMS software library.*

Computational biology, and computational mass spectrometry as a special case, deals with providing computational methods to analyze and interpret data from molecular biology. Although the idea is not new, the field has received increasing attention in the last years as data volumes increase to a point unmanageable for manual analysis.

To familiarize the reader with our terminology and the current state-of-the-art methods, we describe the most typical LC-MS setups and analysis pipelines while also covering algorithmic challenges and solutions to quantification and identification scenarios. Our contribution to the widely used software library OpenMS is also covered in this chapter.

2.1 Separation

A separation step is advisable for high-complex samples like serum or whole cell lysates before the samples are measured by MS. The former method of choice was two-dimensional gel electrophoresis (2-DE), which nowadays is seldomly used since the method is not automatable, thus very time-consuming, lagging reproducibility, thus not cost-effective [37]. Furthermore, 2-DE only offers a very limited dynamic range, preventing identification of low abundance species [38]. On top of that, it has problems with hydrophobic proteins and species of extreme low or high molecular weight or isoelectric point [39].

An alternative to gels is *capillary electrophoresis* (CE). It offers robustness against interfering substances and uses inexpensive capillaries, it is compatible with practically all buffers and analytes, delivering high separation efficiency and speed [40, 41, 42, 43]. Furthermore, it produces visible trends, which can aid in peptide identification [44]. However, decreased loading capacity, which highlights the potential problem of sensitive detection [42] and (earlier) poor reproducibility in migration time [41, 45], are drawbacks of this technology.

The term CE refers to a family of separation techniques that use narrow-bore fused-silica capillaries to separate a complex mixture of large and small charged molecules. In a high electric field, molecules are separated based on their physical-chemical properties which determine their migration time, which is further dependent on the background electrolyte (BGE) and its properties, e.g., ionic strength, pH, or type of ions [46]. The most commonly used trade of CE is capillary zone electrophoresis (CZE), for which CE will be used synonymously from now on. In CZE, the separation mechanism is largely based on differences in the charge-to-mass ratio of the analytes and requires homogeneity of the buffer solution as well as a constant field strength throughout the length of the capillary. An acetate buffer is usually employed in CE-MS experiments because it is a volatile buffer and is fully MS compatible. Neutral molecules pass down the column at the pace of water while positively charged analytes are accelerated and negatively charged analytes are retarded by the electrical field [47]. More precisely, the time taken for the analyte to migrate through the column is described as the “migration time” (MT) rather than the “retention time” (RT) as in *high-performance liquid chromatography* (HPLC). MT is the product of the electric field strength times the apparent mobility (electrophoretic mobility + electroosmotic flow mobility) in the BGE. The electrophoretic mobility of an analyte depends on charge, size, shape, and hydrophobicity properties [48]. The electroosmotic flow is the bulk flow of liquid through the capillary. It is influenced by the dielectric constant and viscosity of the buffer. The electric field strength is a function of applied voltage divided by the capillary length [46].

A model-oriented discussion on the topic of migration time can be found in Section 3.1.

The most commonly used fractionation technique is high-performance liquid chromatography (HPLC), which is highly automatable and easily coupled to a mass spectrometer. The

term “high performance” hints at its ability to cleanly separate nearby species, i.e., achieve a high resolution. HPLC instrumentation includes a pump, injector, column, detector and data system. In brief, the mixture is forced through a stationary phase by the flow of a mobile phase at high pressure, separating the mixture into its components. The stationary phase is defined as the immobile packing material in the column, whereas the mobile phase is the solvent added to promote elution. The solvent’s composition can be changed in time to change the interaction of the solute with mobile and stationary phase.
HPLC has also received attention from the machine learning community, where multiple approaches for predicting retention times have been published [49, 50]. This allows to weed out false positive peptide identifications based on a trained RT model, to predict peptide retention time for in silico experiments, and to design targeted proteomics experiments, e.g., for multiple reaction monitoring (MRM).

Both CE and HPLC are highly automatable and can be coupled directly to a mass spectrometer (so-called online LC-MS). Here, the stream of analytes eluting from the column over time is directly introduced into the mass spectrometer. Alternatively, the material eluting from the column is stored (e.g., on a plate) and data acquisition in the mass spectrometer is deferred (and might take place in another laboratory). The latter is called offline mass spectrometry.

2.2 Introduction to Mass Spectrometry and Terms

Mass spectrometry is an analytical technique that, as the name suggests, measures masses of molecular species while offering high mass accuracy and detection sensitivity (down to a single molecule). In general, a mass spectrometer consists of three components, namely an ionization source (or ion source), a mass analyzer, and a detector. Mass spectrometers can handle diverse samples (e.g., (non-)volatile, (non-)polar, solid, liquid, gaseous) and complex mixtures. As we will see later in more detail, masses of any compound can only be measured when the ions are in gas-phase and the compound is ionized, thus carries one or multiple charges, which can either be positive (usually by excess of protons due to protonation) or negative (deprotonation). Only then can the trajectory of ions be manipulated by applying an electromagnetic field for which different instrument types use different mechanisms [51]. Thus a mass spectrometer measures not mass but rather mass over charge also denoted using the unitless m/z, where m is the molecular/atomic mass in u and z is the number of elementary charges. The unit of the mass-to-charge ratio is infrequently defined as *Thomson (Th)*, with $1\,\mathrm{Th} = 1\frac{Da}{e}$, where Da is the unit dalton (also called u) and e is the elementary charge.

As any measurement technique, MS also suffers from measurement errors. These can be defined in terms of *accuracy* and *precision*-, also called bias and variance respectively. Accuracy can be corrected using calibration, as it is the difference between the measured mass of an ion and its theoretical mass, typically given in parts per million (ppm), which can be computed as

$$Acc = \frac{m_{\text{measured}} - m_{\text{theoretical}}}{m_{\text{theoretical}}} \cdot 10^6\, ppm.$$

Precision is an intrinsic property of the instrument and describes the reproducibility of a repeated mass measurement as determined by its physical limits [52]. See Section 3.1 for details.

Independent of accuracy and precision is *resolution*, which is defined as $R = m/\Delta m_{50\%}$, where m is the mass to be measured and $\Delta m_{50\%}$ is the minimal distance to the next theoretical mass which can be resolved. The distance is defined in terms of the *full width at half maximum* (FWHM), i.e., the width of a single mass peak at half its maximum height, which needs to be resolved given an adjacent peak (of the same height). Instrument resolution is typically specified at $400\,m/z$. The maximal possible resolution of an instrument is usually fixed by its design, e.g., an LTQ *Orbitrap* XL has a maximal resolution of $R = 100\,000$[1]. Resolution and sensitivity can have a trade-off, i.e., lowering resolution can improve sensitivity or reduce the time required to acquire the spectrum (as is the case for an Orbitrap). In limited scenarios accuracy follows resolution, e.g., when resolution is not sufficient to resolve a peak, accuracy will suffer, but once the peak can be clearly resolved, there is no gain in increasing resolution to increase accuracy.

The dynamic range, defined as the difference between the most and least abundant peptide, can vary considerably in biological samples. In highly complex samples like serum it spans over nine orders of magnitude. If the instrument has only low resolving power, peptide signals will overlap, thus preventing the detection of low abundant peptides [53]. The observable dynamic range is also influenced by the *ionization efficiency* and trapping capacity of the instrument. Ionization efficiency is the ratio of the number of ions formed to the number of molecules consumed in the ion source and is strongly dependent on the ion species, mostly on its pKa and molecular volume [54]. Trapping capacity is only relevant for some types of instruments and denotes the number of ions that can be analyzed concurrently.

Separation (see Section 2.1) can help in improving results by allowing the instrument to record only a subset of molecular species, thereby improving the observable dynamic range of the experiment as a whole.

2.3 MS Instruments

In general, a mass spectrometer consists of three components, namely an ionization source, a mass analyzer, and a detector. We go briefly over each component and highlight computational challenges arising for specific instrument types. An overview of the general setup is provided in Figure 2.1.

2.3.1 Ionization

The two most widely used soft ionization techniques are electrospray ionization (ESI) and matrix-assisted laser desorption/ionization (MALDI).

In brief, MALDI is an offline technique, where the sample eluting from a column is mixed with a matrix compound and spotted onto a plate. Each spot represents a certain interval of eluting compounds. Then, a pulsed laser is used to evaporate and ionize material from a certain spot. The ions are then introduced into the mass analyzer. As usually not all material on a single spot is shot exhaustively during one experiment, it is possible to redo an experiment using optimized instrument parameters or select tandem MS sites offline, which is superior to the online solution as elution shape maxima can be detected more easily. It is possible to store MALDI samples for several months, even years, with only minor influence on sample quality [55, 56]. However, sample depletion, i.e., the consumption of sample due to evaporation, has been reported as a major factor for declining spectra quality [57], leading to inferior identification results.

Due to matrix contaminations, MALDI is known to yield spectra with a baseline, especially in low mass regions, which can obstruct peptide signal detection and quantification. Ionization of peptides by MALDI typically yields ions of charge one, rarely higher.

[1] see http://sjsupport.thermofinnigan.com/techpubs/manuals/LTQ-Orbitrap-XL_Hardware.pdf

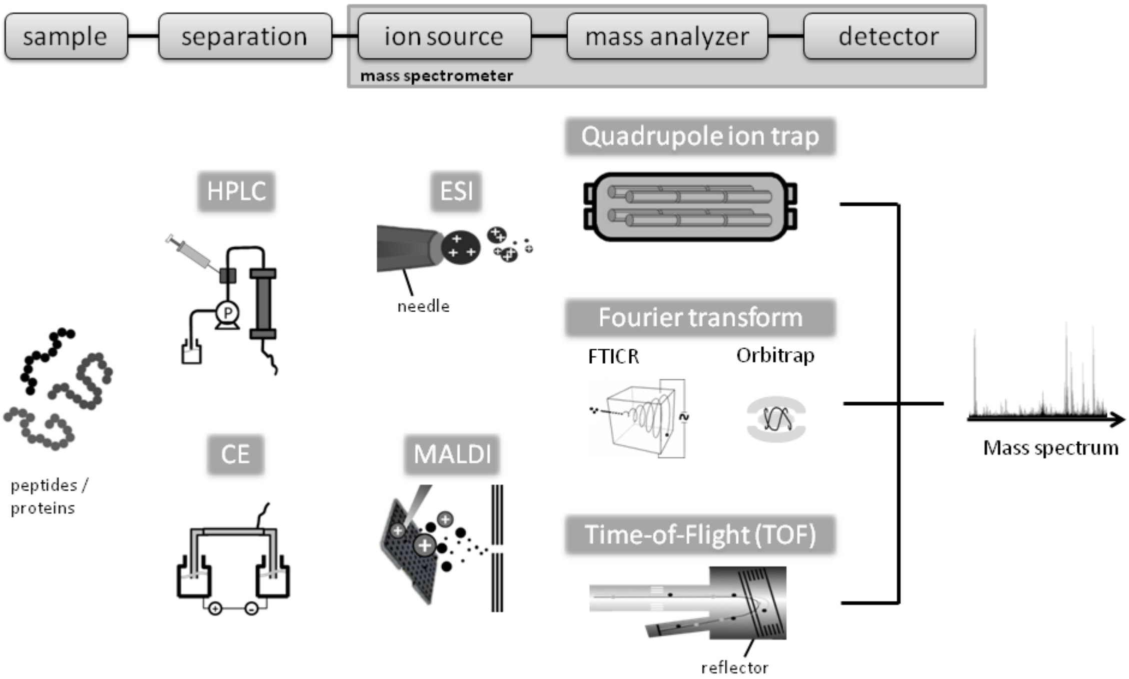


Figure 2.1: Components of a typical LC-MS setup. Common alternatives are shown for each column. Mass analyzers are aligned to ionization sources such that they reflect a typical setup. Combinations of several instruments are not shown, e.g., LTQ Orbitrap or quadrupole time-of-flight hybrid. For low complexity samples the separation step can be omitted.

ESI, on the other hand, is an online technique and can be coupled directly to an LC column. The analyte is forced through a needle at high voltage, leading to solvent evaporation. The exact mechanism of evaporation is unclear, the two most prominent theories being the Ion Evaporation Model and the Charge Residue Model. A good overview can be found in Wilm [58]. Protonation sites are usually attributed to accessible basic residues (Arg, Lys, His and N-terminus) [59, 60]. The number of charges that is taken up by a peptide/protein during ESI is highly complex and depends on a number of factors, e.g., number of basic and acidic residues [61, 62], solution pH, solvent system [63], presence of proton sponges [64, 65], supercharging additives [66], and instrumental factors. Suggestions have been made to experimentally shift and/or compress the charge distribution by ESI [62] to facilitate disentanglement of spectra. Recently, Kaltashov and Abzalimov [67] reviewed the information hidden in charge state distributions to infer macromolecular structure. A peptide species will usually be observable in multiple charge states with a charge of two being the most common and abundant. Multiple charge variants cause signal congestion and denser spectra with redundant signals. This can hamper quantification as the chance of signal overlap increases. Furthermore, algorithms which identify isotope patterns need to be able to scan a wider range (typically charges one to six, with charge two and three being the most common). For proteins, very high charge states ($z > 100$) can be observed with ESI.

Some percentage of a given peptide population can have one or more non-proton adducts. Depending on (column) conditions, this results in potentially many signals belonging to a certain state of a peptide, that have to be identified to get an accurate assessment of the total ion count.

2.3.2 Mass Analyzers

All mass analyzers separate ions by their mass-to-charge ratio in an electromagnetic field. There are many types of mass analyzers; we will describe the most common ones in terms of important characteristics such as resolution, mass accuracy, cost, and dynamic range. Common to all instruments is that from this step onwards, ions are kept under high vacuum to avoid collision (and damage) of analyte ions with other species. Exceptions are tandem MS and variants where ions are introduced into a collision cell on purpose (see below).

One of the earliest class of mass analyzers used in proteomics was the ion trap, where the linear ion trap is superior to the 3D ion trap in terms of ion capacity, scan speed, and mass resolving power. A linear trap quadrupole (LTQ) is an example of a linear ion trap and can be found in modern hybrid instruments such as the LTQ Orbitrap XL (see below). Increased resolving power can be attained by time-of-flight (TOF) analyzers with currently $R \approx 50\,000$. They are often coupled to quadrupole analyzers, which serve as a mass filter during peptide fragmentation. Both instruments also have a very high mass range, making them suitable for large molecules [53]. The analyzers providing the highest resolution R are *Fourier transform ion cyclotron resonance* (FTICR) instruments achieving $R \gg 100\,000$ using a superconducting magnet. A similar principle is used in Orbitrap analyzers where the magnet is replaced by purely electric fields, offering resolutions up to 100 000. However, due to restricted loading capacity, the elution profile shape of compounds can become distorted if the amount of overall eluting material changes, which impedes peptide signal detection on the computational side. Data obtained by FT instruments can suffer from the presence of noise signals known as *shoulder peaks*, which are artifacts of the FT function. Shoulder peaks have an intensity usually below 5% of the main peak [68].

High resolution naturally eases the task of *centroiding*, i.e., the conversion of a peak into a single mass representation, since peaks are separated more clearly and become more narrow. This aids in the analysis of highly complex samples with many interleaving isotope distributions, but also allows to determine charge states of highly charged proteins since the distance between isotopic peaks is $1/z$ and thus becomes closer the higher the charge. As a result, algorithms for the basic pre-processing of data (such as peak centroiding) become conceptionally easier and are usually much faster than their earlier counterparts which required advanced mathematical modeling, e.g., compare a wavelet algorithm [69] against a simple local maxima spline fit.

Different instrument types can produce different peak shapes, which is important for peak matching algorithms. The commonly accepted models in the literature are a Gaussian and Lorentzian shape. A more detailed discussion on peak shapes can be found in Section 3.1.

Another important characteristic is resolution behavior across the m/z range. The most desirable behavior can be seen in TOF instruments, which have a constant resolution, whereas resolution in FTICR instruments decays linearly with m/z, which is problematic for high m/z. Orbitraps show intermediate behavior (square root decay with m/z) [70]. See Figure 3.4 for an illustration. This may pose a serious problem to a peak picking (i.e., centroiding) algorithm, which is sensitive to the expected peak width as it changes with increasing m/z. With increasing resolution and improved signal-to-noise behavior, one can argue that even naive algorithms without knowledge of peak width are sufficient. However, especially for low-resolution instruments, it is desirable to have a more detailed concept for a peak to filter out noise and increase specificity.

2.3.3 Detectors

Each instrument type uses its own detection system. For TOF and quadrupole instruments, the common choice is the electron multiplier where ions are detected on impact. FTICR and Orbitraps on the other hand use contact-free detection plates which record the ions passing by, allowing multiple rounds of detection without destroying the ion, thus allowing increased sensitivity.

2.3.4 Data Terminology

We now create a small nomenclature, which is identical to the one used within our software library OpenMS and its documentation, in order to simplify the description of data and algorithms.
The rawest type of data, which has not been preprocessed and is by far the most memory consuming, is referred to as *raw data*. Here, individual isotope species still have a Gaussian-like shape, which we refer to as *peak*, i.e., one peak represents multiple datapoints.
Using a lossy signal compression process called centroiding or peak picking, one can represent a peak as a single data point called a *stick*, which is usually close or identical to the peak's apex. This representation usually reduces the amount of data by one order of magnitude. Modern instruments allow to retrieve the data directly in this format, though it might be advantageous to request the raw data and perform centroiding using custom solutions as provided by OpenMS. When the meaning is clear, we sometimes loosely refer to a centroided m/z value as peak even though stick would be more appropriate.
A *spectrum* is a set of peaks or sticks covering a certain m/z range. It can have a certain retention/migration time attached to it when recorded in an LC-MS setup. A spectrum can be recorded in different MS modes (e.g., MS^1 or MS^2 – see Section 2.5).
An *LC-MS map* is a collection of spectra covering a certain RT range; thus, each datapoint can be described by RT, m/z, and intensity.
A *feature* represents the average retention time, the monoisotopic m/z, and the integrated intensity of a (peptide) signal in a certain charge state.
For an illustration of an LC-MS map with multiple peptide signals see Figure 2.2. The process of identifying features in a (centroided) LC-MS map is called *feature finding*. A *feature map* is a collection of features representing all features of a single LC-MS map. As retention time is usually not 100% reproducible between different experiments, there is a need to superimpose corresponding entities across maps in a process called *map alignment*, which is using either raw LC-MS data or feature data (with or without peptide identifications) to identify landmarks which are used to compute an RT transformation.

To represent a set of features, we use the *consensus feature*. The common property which the features are grouped by is usually clear from the context. Groups may also represent charge variants with similar RT but different m/z of a single peptide in one map during decharging, or labeled pairs during feature finding. The most common scenario is groups in feature linking where features with similar m/z, RT, and charge across multiple feature maps are grouped, representing the (putatively) same peptide in different experiments. A consensus feature always has a centroid which stands for all features it represents, e.g., in map alignment the centroid would be the mean m/z and RT of the features.

Similarly, a *consensus map* is defined as a collection of consensus features.

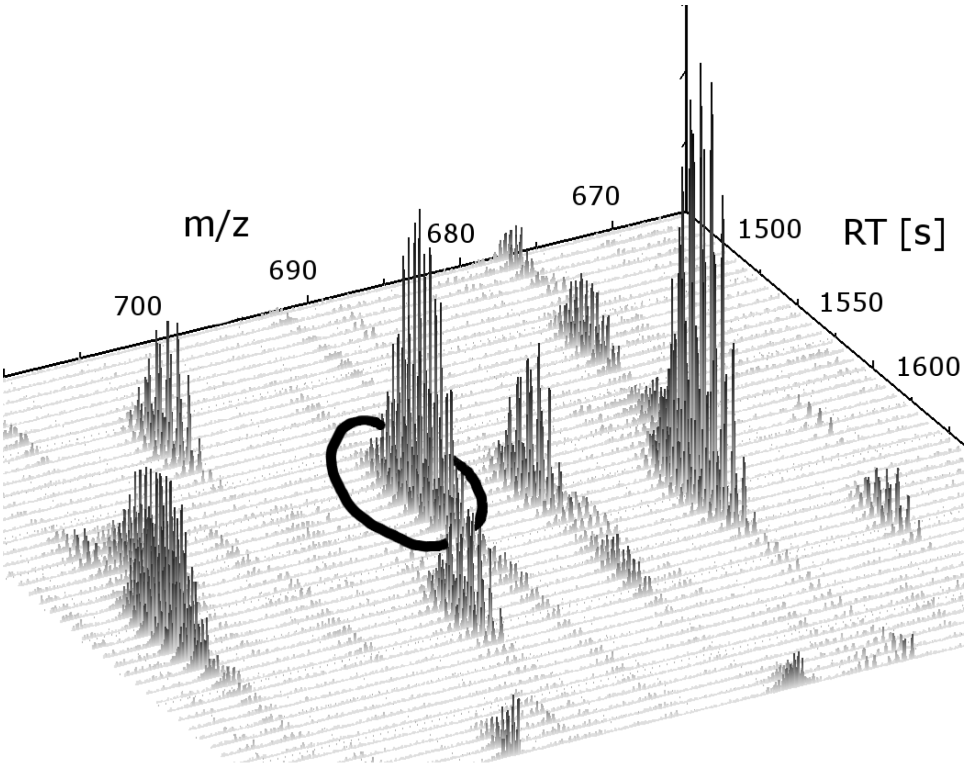


Figure 2.2: Example of an LC-MS map in 3D view with multiple peptide features, each with an elution profile in retention time (RT) and an isotope pattern in m/z. One arbitrary peptide feature is highlighted.

2.4 LC-MS for Biomolecules/Proteins/Peptides

2.4.1 Isotope Distribution

As this thesis revolves around algorithms for simulation and quantification of peptides and proteins, we will explain some of the characteristic particularities of peptides and proteins in contrast to other compounds.

In order to understand how quantification and, especially, identification work, an understanding of the nature of the data is required. For proteins and peptides, mass spectrometers are usually operated in positive ionization mode, i.e., all compounds carry one or more positive charges. In mass spectrometry, which measures mass over charge of a molecule, different notions of the mass of a molecule and its isotope distribution are the most vital concepts. Every chemical element has a number of different variants – so-called isotopes – which only differ by the number of neutrons and thus by mass. The number of protons is by definition identical for all isotopes of a chemical element, e.g., hydrogen always has one proton and may have zero or one neutrons in its stable form. The average atomic mass of an element is the sum of all its isotope masses weighted by their frequency of occurrence. An average mass of a peptide or protein is simply the sum of the average atomic masses of all elements of a molecule. Thus, the average mass needs not coincide with a mass from the isotope distribution, since it is weighted. Note that, depending on the source, elemental isotope distributions may vary slightly; thus, the average atomic mass of a certain element, e.g., carbon, might not be the same across different geographical locations.

The isotope distribution of large molecules (such as proteins) is determined by their elemental composition. For the more general class of metabolites the number of possible sum formulas, given an isotope distribution of an unknown molecule, is very high. However, due to the regular architecture of peptides and proteins featuring a limited number of 20 amino acids as building blocks, the isotope distribution is much more predictable – notwithstanding chemical modifications, e.g., post-translational modifications (PTMs). Thus, for proteomics one usually considers the elements carbon, hydrogen, nitrogen, oxygen and sulfur (short CHNOS). Given any peptide sequence, one can compute the gross isotope distribution, i.e., the probability to carry n_i additional neutrons ($i \in \{0 \ldots n\}$). By definition, the monoisotopic peak is always one containing only the most abundant isotope of each element. For CHNOS this also coincides with the lightest (stable) isotopes, i.e., ^{1}H, ^{12}C, ^{14}N, ^{16}O and ^{32}S. As mass increases, the probability of a peptide not to contain an extra neutron decreases, i.e., the monoisotopic peak vanishes. For a heavy protein with 16.9 kDa like equine apomyoglobin, it contributes only 0.4% to the total abundance.

The regular nature of peptides allows to approximate their atom composition just by knowing their mass. This can be done by simply looking at a large protein database and count the number of amino acids. From this, an average amino acid of mass 111.1254 – termed *averagine* – can be derived, having an elemental composition ($\mathrm{c_{avg}}$) of $\mathrm{C_{4.9384}}$, $\mathrm{H_{7.7583}}$, $\mathrm{N_{1.3577}}$, $\mathrm{O_{1.4773}}$ and $\mathrm{S_{0.0417}}$ [71]. To estimate the isotope distribution given a mass X, one computes $f(X) = X/111.1254 \times c_{avg}$ to get the fractional sum formula. The latter is then rounded element-wise to its nearest integer and filled with hydrogen to correct for the rounding error. See Figure 2.3 for an example of averagine isotopic gross distributions, i.e., with nominal masses only.

The fractional averagine model as an extension to the general averagine has also been proposed [23], but its use seems limited as the results are very similar and the large portion of uncertainty lies not in the truncation of the fractional sum formula but rather in the attempt to describe the wide distribution of peptides (especially the ones containing sulfur) using one average amino acid.

Given any sum formula, whether derived from the averagine model or from the actual peptide sequence, the isotope distribution can be computed in a number of ways (see Valkenborg et al. [72]). The mass defect is usually not considered as instruments typically do not resolve this rather tiny mass difference and combinatorial explosion gives rather unfavorable computation times. Thus, most algorithms only deal with computing the gross isotopic distribution, usually by convolution of two isotope distributions.

However, not only the number of extra neutrons in a molecule are important but also the fine isotope structure as different elements have different mass defects, i.e., the binding energy for the extra neutron will result in subtle mass differences between ^{13}C and ^{12}C compared to ^{2}H and ^{1}H. E.g., consider the molecule CO with the gross structures 28, 29, 30 and 31, where each number gives the summed nucleon number. The gross structure of ^{12}C^{16}O has only one fine structure whereas within the gross structure of 29 are two fine structures, ^{13}C^{16}O and ^{12}C^{17}O [73]. With today's instruments, fine structures can usually not be discerned, thus for most practical applications the fine structure is not relevant yet. This will change, however, when instrument resolution improves. See Figure 2.4 for an example. A very memory-efficient algorithm for computing isotopic fine structure is described in [73].

Direct measurement of proteins is known as *top-down* mass spectrometry. It has the advantage of being able to better locate chemical modification sites and a simpler sample handling, as a digestion step can be omitted. Due to their size, proteins can carry many charges and give rise to a broad isotopic envelope where the monoisotopic peak is hard to observe due to its low abundance. Furthermore, the higher the charge the closer two isotopic peaks will be; thus,

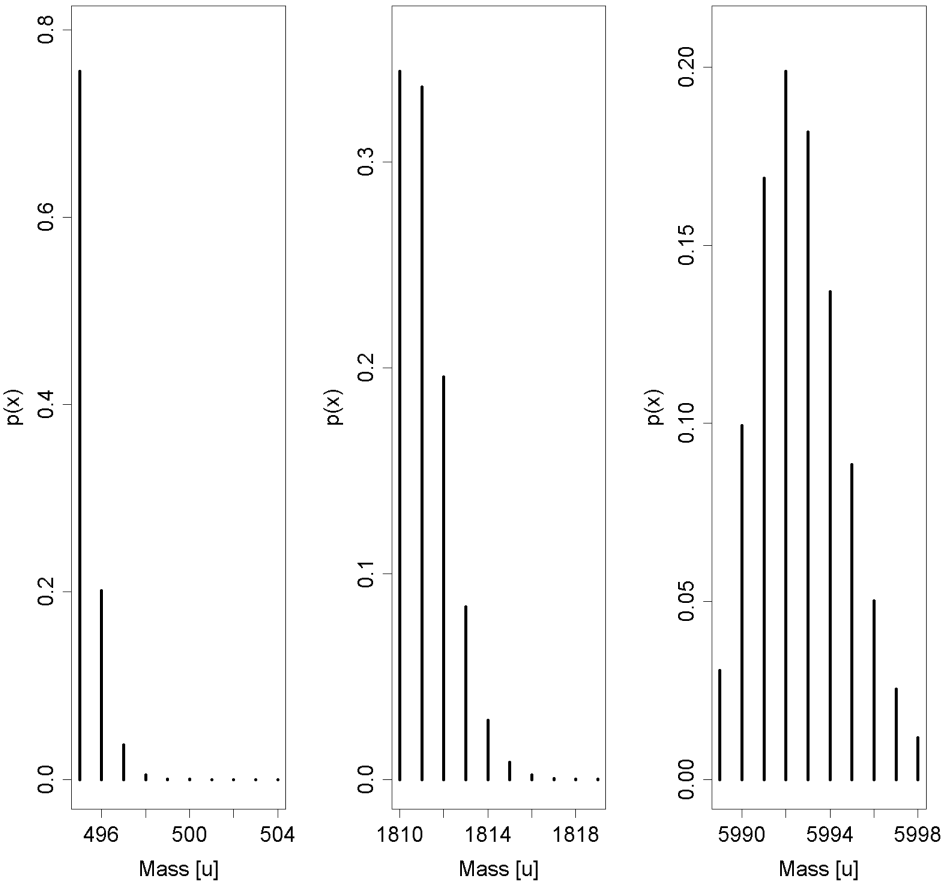


Figure 2.3: Isotope gross structure of the averagine model for three different masses. The monoisotopic peak (i.e., leftmost peak) declines with increasing mass, and the distribution becomes wider and more Gaussian-like.

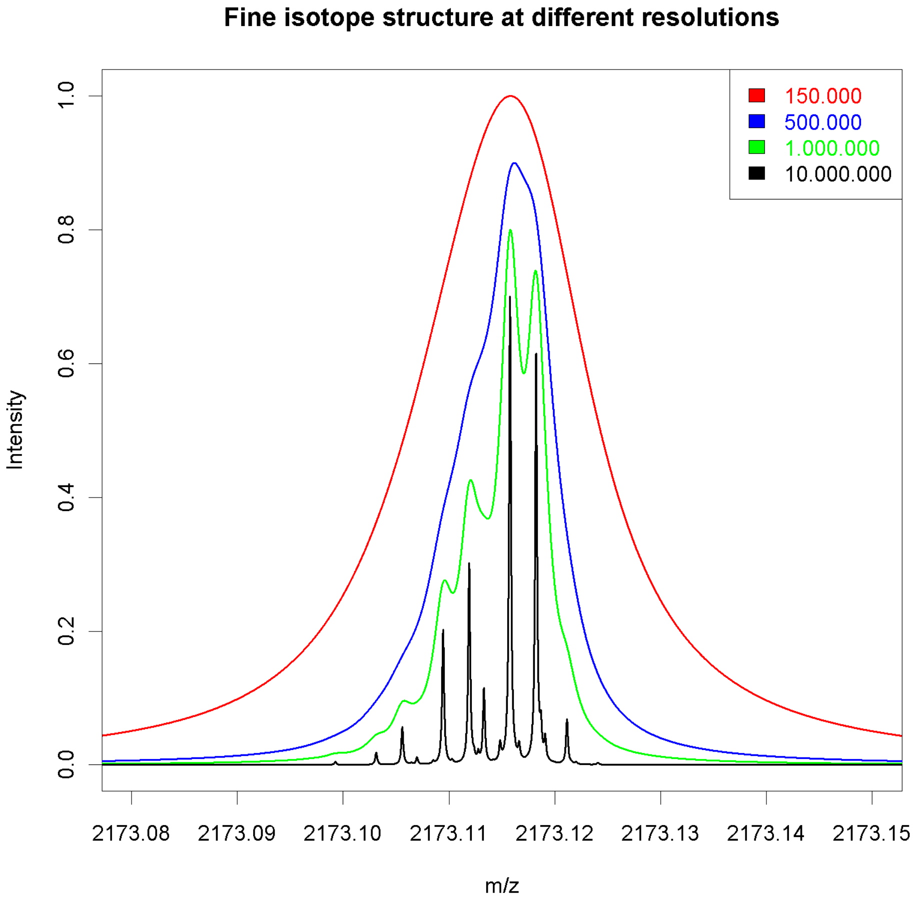


Figure 2.4: Isotopic envelope of the TRP-Cage protein with only 20 amino acids (NLYIQWLKDGGPSSGRPPPS) with monoisotopic mass of 2 168.10145 Da. Displayed is the 2 172 gross isotope peak at ≈2 173.12 Th, consisting of multiple fine structures at different resolutions ($R = M/\Delta m_{50\%}$). The masses were convolved with a Lorentzian peak shape to reflect the limited mass precision of the MS instrument.

high resolution is required. In contrast, *bottom-up* mass spectrometry deals with the analysis of peptides which are cleaved from proteins by enzymatic digestion or created by synthesis. For digestion, trypsin is the most widely used enzyme due to its high specificity to cleave after lysine (Lys) and arginine (Arg), which under the optimized experimental conditions leads to two well-defined charges for a tryptic peptide, i.e., one charge at the N-terminus and one charge at the C-terminal Lys/Arg residue. Under normal experimental conditions most peptides will carry two charges when trypsin is used for digestion. Additionally, tryptic peptides are on average ≈14 amino acids long, yielding ions in a desirable mass range [74].

2.5 Identification

Identification of peptides is feasible using two different approaches. The conceptually easier one is called peptide mass fingerprinting in which a peptide is identified solely based on an accurate mass measurement, which is compared against a list of known theoretical masses. However, modifications, insufficient mass accuracy and incomplete databases usually make unambiguous identification hard or impossible, especially in highly complex samples. A more sophisticated version uses accurate mass and time (AMT) tags where a normalized retention time is incorporated in addition to an exact m/z value. See [75] for a good overview on uniqueness of mass and other peptide parameters for different proteomes.

A more complex approach is identification by tandem mass spectrometry where all ions within a certain m/z range are selected for fragmentation within a collision cell. The mass range ideally contains only one peptide species which is also known as the *precursor ion* or parent ion. Within the collision cell the precursor fragments along the backbone, yielding peptide fragments (called *product ions* or fragment ions) which are re-introduced into a mass analyzer, yielding typical ion ladders. Depending on the position within the backbone, ions are assigned different names; see Figure 2.5 for an illustration. The dominant ion types depend on the fragmentation technique used. The most prominent techniques are collision-induced dissociation (CID), higher-energy collisional C-trap dissociation (HCD), and electron-transfer dissociation (ETD). CID and HCD result in strong b and y ion ladders, whereas in ETD c and z ions are produced primarily.

The distance on these ion ladders can be used to infer (sub-)sequences of the parent peptide. The resulting MS spectrum is also known as tandem MS or MS^2 spectrum. When the search space for the peptide sequence is unrestricted, the inference of the sequence is termed *de novo* sequencing as no database with potential peptide sequences is provided or even known. If the content of the sample can be narrowed down and a database can be used, the search strategy can be adapted as now each database entry can be used to generate a theoretical spectrum which is matched against the observed spectrum and scored using some measure of similarity. As search space is much more restricted, database search is usually faster and more powerful, yet not applicable if a database is unavailable. The most widely used algorithms for database search are Mascot [76], Sequest [77], OMSSA [78], and X!Tandem [79]. For de novo, usually PepNovo [80] is used. OpenMS includes a similarly performing algorithm named ANTILOPE [81].

It is also possible to combine multiple search engines to improve either sensitivity, or specificity, or both at the same time. Details can be found in Nahnsen et al. [82].

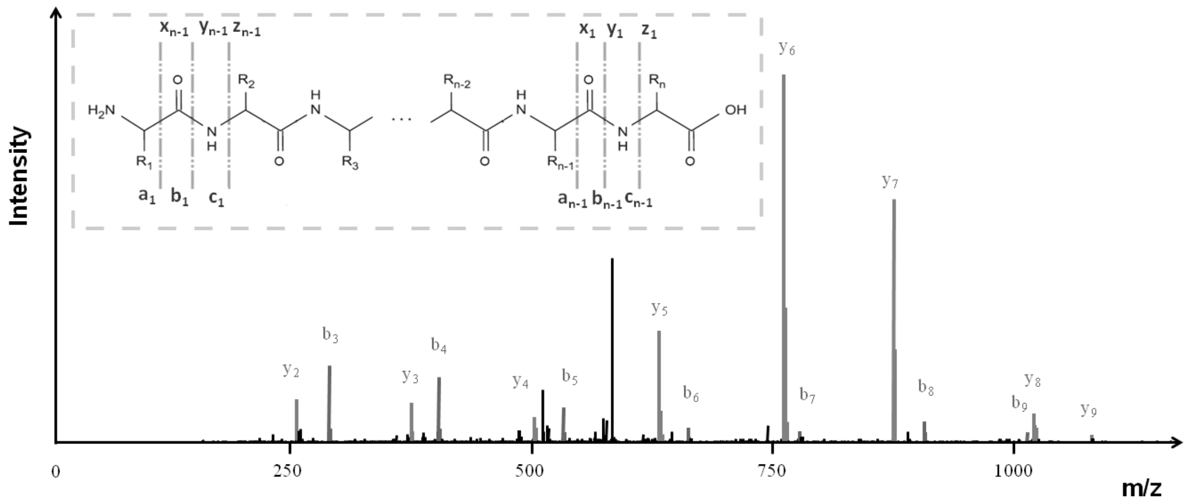


Figure 2.5: Example of a CID spectrum with b and y ion annotation. The inset shows the naming convention for fragment ions, depending on the break point within the peptide. Ion types a, b, and c denote an N-terminal fragment whereas z, y, and z ions denote C-terminal fragments. The subscript indicates the number of amino acid residues.

2.6 Quantification

Varying susceptibility to efficient ionization for different peptides/proteins is one of the major reasons why mass spectrometry is only semi-quantitative. Inferring the sample concentration of certain peptide/protein species from the measurement is not possible without the use of internal standards where concentration is known. Usually, these standards are heavy isotope versions of the molecule of interest to ensure comparable ionization efficiency. If no internal standards are used, quantitative statements are only valid to a certain extent for comparative purposes to other samples, and no absolute concentration can be inferred.

For quantification in HPLC-based proteomics, two paradigms are prevalent. In label-free quantification, each biological sample is measured separately, resulting in multiple maps containing the signals of the eluting peptides. In order to compare the signals across samples, they first have to be identified in the corresponding maps and then grouped together (applying suitable data reduction, mapping, and normalization methods).

In labeled quantification, different biological samples are measured in a single map concurrently – a procedure also termed multiplexing. In order to distinguish the states, they can be labeled with a fixed mass label, shifting the peptide along the m/z axis (see, for example, Figure 4.2a for measured data and Figure 4.2b for the respective features of two peptides in two different labeling states – indicated by filled or empty symbols). Labeling can be done on either the MS^1 or the MS^2 level, and solutions for up to eight channels are available. A *channel* is the sample assigned to one labeled state (e.g., *light*) in a labeling experiment.
Today, there is a clear trend toward multiplexing [83], e.g., for time-resolved experiments or multiple experimental conditions (see Chapter 5 for our own study).

In both paradigms, the ratio of the assigned pairs of signals can be used for subsequent data analysis (e.g., for the detection of biomarkers). We will now discuss advantages and disadvantages of the respective methods. See Figure 2.6 for a graphical overview.

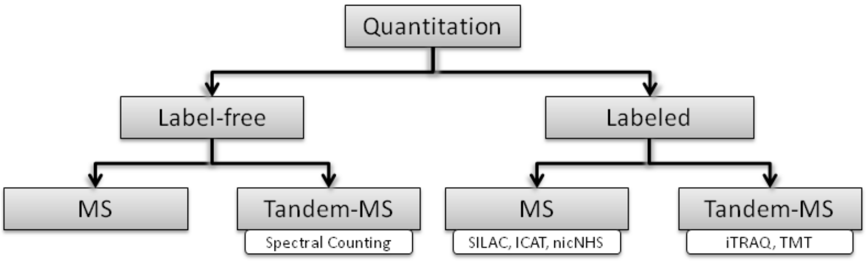


Figure 2.6: General scheme for quantification methods with respect to MS level and labeling state. If applicable, the most widely known representatives are listed.

2.6.1 Labeled vs. Label-free

If experimental conditions and purification procedures can be properly controlled, label-free quantification is an attractive strategy because it requires no additional sample preparation steps.

Two types of labeling are discerned. One is in vitro chemical labeling after sample collection, which is applicable to any sample, the is other metabolic labeling during cell growth, which can be used only for cell cultures or small animals. Popular methods for in vitro chemical labeling are isotope-coded affinity tagging (ICAT) and trypsin-catalyzed ^{18}O labeling. Chemical labeling techniques like isobaric tags for relative and absolute quantitation (iTRAQ) [27] or tandem mass tags (TMT) [84] might be cost intensive and have the disadvantage of requiring yet another biochemical step (the labeling itself). Metabolic labeling is also known as stable isotope labeling with amino acids in cell culture (SILAC). For MS^1 labeling, quantification with one label absent is hard or even impossible as one can not determine if the light or heavy species is observed when its partner is missing unless an identification is present or PMF is used. Labeling on the MS^1 level will lead to increased signal congestion, which is especially problematic for high complex samples as the amount of overlapping peptides increases. This complicates quantification as signal contributions from different channels need to be disentangled.

For all labeling methods it holds that *labeling efficiency* might not be perfect, labeling might be biased towards one type of label, or only applicable to certain peptide species. ICAT, for example, can only be applied to peptides and proteins containing a cystein residue. SILAC suffers from metabolic conversion of the stable isotope-labeled peptide [85].

Due to multiplexing, each sample can be measured in a fraction of the time (depending on the number of channels) required for label-free measurements. For a long gradient (e.g., 5 hours and more), this can be a critical advantage when measurement time is limited.

Some labeling techniques are more suited for certain instrument types; e.g., quadrupoles and TOF instruments excel at measuring low m/z ions whereas for ion traps, the recovery of fragment ions below 30% of the precursor ion mass is very poor, which makes them unusable for iTRAQ or TMT. These two labeling techniques give rise to light reporter ions which are used for quantification [83].

The advantage of labeling techniques is that they naturally control for instrument variability from the point of channel mixing onwards (e.g., column condition, instrument settings, instrument performance) as each channel is affected simultaneously, whereas in label-free experiments, instrument calibration might change in time (as shown in [86]), i.e., the earlier channels are mixed, the smaller the experimental error component will be. The timepoint will differ between labeling methods, e.g., for bottom-up approaches, SILAC allows mixing before digestion whereas iTRAQ does not.

Noirel et al. [87] provide an insight into the popularity of different methods (label free, iTRAQ, SILAC, and ICAT) in their recent paper [87] – see Fig. 1 therein. By this measure, iTRAQ is clearly in the lead, followed by SILAC and label-free.

2.6.2 MS^1 Quantification

Quantification in MS^1 is achieved by finding peptide or protein signals, and using the signal intensity to assign a quantitative value to the entity (which is not necessarily identified). It is important to note that comparing intensities across multiple experiments usually requires some kind of normalization. Even the intensity of different peptide species within one experiment cannot be used to directly infer their abundance in the sample as peptides have different ionization efficiency. On the software level quantification can be implemented on the raw data or stick level, and the derivation of peptide abundance can be based on different schemes, e.g., by summing all datapoints deemed to belong to the peptide, or by fitting a model and using its theoretical area or maximum.

If a label-free strategy is adopted, quantification is usually done on the MS^1 level (see Subsection 2.6.3 for alternatives in MS^2).

Popular multiplexing techniques used for quantification on the MS^1 level include SILAC and isotope-coded affinity tagging (ICAT) or labeling with nicotinoyloxy succinimide (nicNHS).

2.6.3 MS^2 Quantification

MS^2 based methods obviously require MS^2 acquisition. In the case of iTRAQ only higher-energy collisional C-trap dissociation (HCD) and pulsed-Q dissociation (PQD) [83] are feasible. These are usually marginally inferior to collision-induced dissociation (CID) in terms of identification performance thus offsetting the multiplex advantage slightly by requiring longer gradients to achieve the same coverage in terms of MS^2 identifications.

Methods for label-free MS^2 (pseudo) quantification are controversial in the community and include Spectral Counting, exponentially modified protein abundance index (emPAI) [88], Robust Intensity Based Averaged Ratio (RIBAR), and Extended Robust Intensity Based Averaged Ratio (xRIBAR) [89]. A benchmark can be found in [89].

Robust and sensitive quantification in MS^2 can be achieved using *multiple reaction monitoring* (MRM). MRM in particular has the advantage of high dynamic range, combined with reliable acquisition of the targeted species, for up to 100 proteins. MRM uses selected (and specific) precursor and fragment ions of a peptide (a so-called transition) as it elutes off the LC column. Quantification is performed on the chromatogram obtained from the fragment ion. Transitions are designed to be very specific for certain (unique) peptides, thus allowing identification as well. MRM works for highly complex samples and is usually performed on triple quadrupole mass spectrometers where the first quadrupole isolates the precursor, the second quadrupole acts as collision cell, and the last quadrupole records the fragment ion(s).

Methods allowing multiplexing in MS^2 include tandem mass tags (TMT) and isobaric tags for relative and absolute quantitation (iTRAQ). The latter has become a popular multiplexing technique, which we will now describe in more detail: iTRAQ is an in vitro chemical labeling procedure, consisting of either four or eight isobaric (equal nominal mass) tags, each of which can be used to label a specific peptide sample. Prior to labeling with iTRAQ reagent, protein samples are digested proteolytically to allow labeling of the peptide's N-terminus. After labeling,

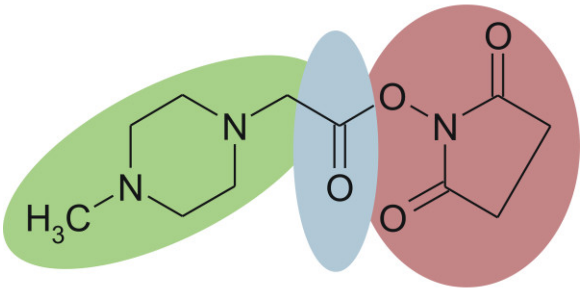


Figure 2.7: Structure of the 4-plex iTRAQ tag. Green) Reporter group, Blue) Balancer group, Red) Reactive NHS ester group.

the samples are mixed and then subjected to (LC-)MS analysis. Identical peptides from each sample will have identical masses in MS^1, and quantification will only be possible in MS^2. A tag consists of a peptide reactive region, a reporter region, and a balance region (see Figure 2.7). The intensity of the reporter ions reflects the relative amount of peptides in each channel.

The NHS ester of the peptide reactive region is designed to react with the N-terminus and lysines of peptides after protease digestions, but might also attach to tyrosine. As usual in MS^2, fragmentation takes place along the peptide backbone, allowing for qualitative analysis while simultaneously affecting the link between the reporter and balance region of the tag, resulting in intense reporter ions in the tandem mass spectrum. For the 4-plex version, the reporter groups appear in the MS^2 spectrum at m/z 114.1, 115.1, 116.1, and 117.1. The attached balancer group is designed to create tags of a total mass of 145 Da, requiring balancer group weights of 31, 30, 29, and 28 Da, respectively. There is also an 8-plex iTRAQ kit where four more reporter tags with masses at about 113, 118, 119 and 121 Da are available. The 8-plex tags themselves are heavier though, having 305 Da instead of 145 Da as in the 4-plex case. Due to the low-mass reporter ions, not all instrument types and fragmentation technique combinations lend themselves equally well to iTRAQ analysis, e.g., due to the 1/3 rule (i.e., the lowest m/z value recordable is about 1/3 of the precursor mass), an ion trap in CID mode cannot observe reporter ions [90].

For some more study specific details, see Chapter 5.

2.7 Software for Analyzing MS Data

2.7.1 Widely-used Software Packages

The role of software for data analysis and processing is becoming more and more important. A recent study [91] conducted by the Human Proteome Organization (HUPO) showed that, the raw data quality obtained from the instrument is usually a very good basis to start from, but it is the choice of the software and algorithms therein that determine the success of the analysis.

Every instrument is shipped with the vendor software which enables users to analyze their data. Analysis options differ: Most vendors offer built-in support for search engines (like Mascot or SEQUEST) and a software for visualization. However, the user is usually restricted to certain scenarios as analysis pipelines are monolithic and not flexible. Also, implementation details of the algorithms are rarely made public, and comparison against other algorithms is difficult because of different data formats, e.g., it is not possible to benchmark most vendor software when only an mzML file is available, as converters to mzML are usually available but not the other way around.

In addition to vendor software, a number of free and (usually) open source packages have been developed, which allow data analysis across different platforms and flexible workflow construction.

Each package has a slightly different focus, and not all operating systems (OS) platforms are supported. We will briefly describe the most important packages.

One of the most comprehensive packages available for all major platforms is the Trans-Proteomic Pipeline (TPP) [92], which focuses on the analysis of MS^2 data sets. It includes a web server, thus allowing the user to use the browser as a user interface, but can also be used from the command line. The TPP includes widely known tools like PeptideProphet, ProteinProphet and ASAPRatio which can be used to analyze labeled data sets.

MaxQuant [93] is a quantitative proteomics software package specifically aimed at high-resolution MS data (e.g., Orbitrap). It supports several labeling techniques but is mostly known for its SILAC pipeline. Recently, the search engine Andromeda was incorporated. MaxQuant is freeware but closed source and requires a Windows PC.

MZMine [94] and MzMine2 [68] are Java-based, open source packages originally aimed at analyzing metabolite data but with capabilities for peak picking, advanced visualization, and map alignment.

VIPER (Visual Inspection of Peak/Elution Relationships) [95] includes a GUI and supports feature detection, calibration, feature alignment, and identification mapping. Its main focus is AMT processing. It is written in VB (version 6) and thus runs on Windows OS only. It can read the outdated mzData and mzXML formats.

ProteoWizard [96] is a C++ open source, cross platform suite primarily known for its ability to convert most vendor formats into HUPO-PSI mass spectrometry data formats like mzML or mzXML. It also includes Skyline, an editor for creating and analyzing selected reaction monitoring (SRM) experiments.

msInspect [97] uses Java and the R language and is thus platform-independent. As input mzXML and pepXML are accepted, mzML support is available in the development version. Supported are basic signal processing, feature detection, label-free and labeled quantification, MRM data analysis (via the MRMer tool), alignment, and identification mapping. It also includes msInspect/AMT, a suite of tools for Accurate Mass and Time analysis.

OpenMS is an open source library written in C++ for label-free and labeled quantification and identification, supporting all major platforms. The OpenMS Proteomics Pipeline (TOPP) [98] is a set of executables, chainable in modular fashion for a wide set of analysis scenarios, and covers common tasks like peak picking, map alignment, identification (via wrappers for common identification engines like Mascot, X!Tandem and OMSSA), filtering, and quantification of labeled and label-free data. It supports the HUPO-PSI standards mzML and mzIdentML as well as the widely used pepXML and protXML formats, enabling data exchange between collaborators based on open platform-independent formats.

2.7.2 OpenMS in Detail

As the algorithms presented in this thesis are integrated into OpenMS/TOPP, we devote some pages to the library itself, its design principles, and to how it can be used by developers and users. A description focusing on on library design and project management can be found in Sturm [99].

OpenMS itself currently consists of a core OpenMS library and the OpenMS-GUI library. The core library implements data structures for data points, spectra, LC-MS maps, features,

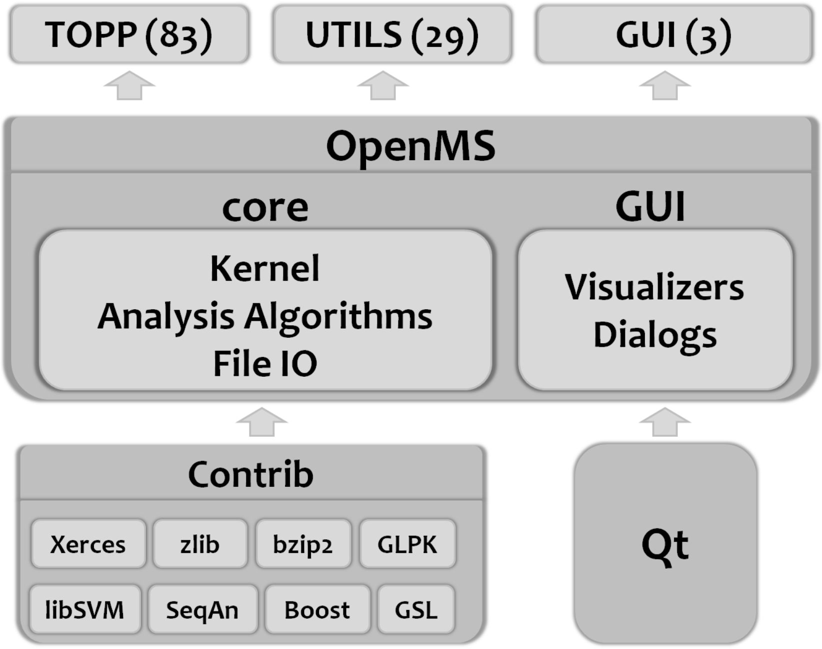


Figure 2.8: OpenMS dependencies (bottom) and OpenMS library structure (center). Executables for the end user are shown at the top. Numbers in brackets give the number of tools in each category as of OpenMS release 1.9.

consensus features, and a wide range of algorithms, e.g. for signal processing, file handling, peptide identification and quantification.

The OpenMS-GUI library contains all graphical user interface (GUI) components, like spectrum widgets and dialogs, used by the GUI tools TOPPView [100], TOPPAS [101] and INIFileEditor shipped with OpenMS. The split library reduces compile/link time overhead and avoids linker restrictions on Windows OS with respect to library size.

OpenMS has dependencies on other libraries which enable parsing of XML files (via Xerces-C), handling of GUI components (via Qt), numerical processing (via GSL), machine learning (via libSVM), integer linear programming (ILP) solvers (via GLPK), sequence alignment (SeqAn) and more. All support libraries except Qt are bundled in a contrib package provided by the developers of OpenMS, which is essential for Windows platform support. On Linux and MacOS, the native packages can be used as an alternative. This allows easy installation of OpenMS' dependencies (on Windows) and contains patches for some packages which might otherwise lead to fatal errors and incompatibilities. For an overview of the OpenMS dependencies and library structure itself, see Figure 2.8.

Algorithm prototyping is a key feature of OpenMS as the developer has access to a powerful toolbox for all kinds of data processing steps.

As the library itself is only of use for developers of C++ or programming languages capable of wrapping C++ libraries, TOPP tools are provided to enable non-programmers to connect algorithms in a flexible and comprehensive way. TOPP is meant to provide small building blocks for a wide range of analysis pipelines, which can be chained together to fit custom needs.

Build System and Installers

The build system of OpenMS relies on CMake [102], an open source cross platform build system using simple platform-independent and compiler-independent configuration files, e.g., CMake-

Lists.txt. CMake generates native makefiles and workspaces via so-called generators for many environments like Visual Studio, nmake, make, XCode, Eclipse and QtCreator. OpenMS also makes use of CTest, a testing software tightly integrated into CMake, which allows to set up automated testing. The results can be submitted to CDash, an open source, web-based software testing server, which aggregates and displays the results of software testing processes submitted from one more clients, usually running different compilers and/or platforms. Finally, binary installers for two major platforms (MacOS, Linux) and architectures (32bit and/or 64bit) are generated by CPack, a package creation tool, to ease installation of TOPP and GUI tools within minutes. For Windows OS, we use the more powerful Nullsoft Installer System (NSIS)[2]. The NSIS-based Windows installer features user account control (UAC) bridging, file extension registration (e.g., for files formats supported by OpenMS' GUI tools), and PATH manipulation. It also includes a release of ProteoWizard, enabling the user to convert many vendor formats into the mzML format, which is supported by OpenMS.

Porting OpenMS to Windows OS

Supporting multiple platforms and thus giving the user the ability to use their favorite operating system (OS), is a desirable property. Indeed, most users of OpenMS are most familiar with the Windows OS, in part because Windows is favored by most vendor software and delivered with the instrument hardware. Supporting Windows allows users to install OpenMS/TOPP on already existing systems.

From a software development perspective, advantages of supporting multiple platforms for a C++ application outweigh the disadvantages in the case of OpenMS and its dependencies. Disadvantages include the need for more maintenance and testing code on multiple platforms. Also, support libraries need to be available for all platforms supported by OpenMS or need to be ported by the OpenMS developers. Platform-specific code is also unavoidable in some cases, e.g. attributes for specifying storage-class information for Windows dynamic link libraries (dll) (i.e., *__declspec(dllimport)* and *__declspec(dllexport)*), platform-specific API calls (e.g., locating executable paths, process ids, time measurements, etc.). However, other cases can be abstracted by use of other libraries, such as Qt. Compiler-specific extensions not covered by a C++ standard (such as the *round()* function) can be used only conditionally or must be avoided, thus making the code more standard-compliant. Advantages of multiple platforms go beyond using multiple compilers on one platform. E.g., debugging and memory-leak checking tools differ for each platform and have different strengths and usability advantages. Usually compilers are not cross-platform and have complementary warning and error messages. Thus finding a bug is easier when using multiple platforms. Bugs might go unnoticed entirely if the code is not tested on multiple platforms. This is especially true for memory access violations. We collected some of the most malicious cases that were detected during the Windows port and testing of clang compiler in Table 2.1.

[2] http://nsis.sourceforge.net

Table 2.1: Coding errors found by using multiple compilers and running tests on different platforms.

code		detected by	explanation
faulty	corrected		
`for (Size j = 0; i < i; ++j)`	`for (Size j = 0; j < i; ++j)`	clang	typing error

`map<int,int> m;`
`m[1] = m.size();`
`cout << m[1]; //VS gives '0', GCC gives '1'`
	`map<int,int> m;`
	`int i = m.size();`
	`m[1] = i;`
		GCC, VS	order of default construction

`long long r = numeric_limits<long long>::max();` [a]
`long long r2 = fabs(r); //GCC: "ok", VS: nc`
`cout << (r==r2); // GCC: false, VS: nc`
	`long long r = numeric_limits<long long>::max();`
	`long long r2 = abs(r);`
	`cout << (r==r2); // VS&GCC: true`
		VS	faulty implicit type conversion

[a] nc: not compiling

Own Contribution

We developed major parts of the current build system for OpenMS and OpenMS Contrib using CMake, CTest and CDash, providing a platform-independent build system, testing, and nightly regression tests.

Five new TOPP tools were written (MSSimulator, Decharger and ITRAQAnalyzer, EICExtractor, GenericWrapper). MSSimulator, Decharger and ITRAQAnalyzer will be described in the following chapters.
EICExtractor was developed for targeted metabolite feature finding and quantification (manuscript in preparation).
The *TOPP Pipeline Assistant* (TOPPAS) was extended to allow more flexible workflows, error checking, file list recycling, online download of workflow files, and more (see [101] for details).
The *GenericWrapper* tool was written to enable external program support via TOPP tool descriptions (TTD) which describe parameter mappings, values and types from external third-party tools to a TOPP-conformant format. This enables the usage of these tools within TOPPAS or other workflow engines if the TOPP interface can be parsed (e.g., via KNIME [103]).

We ported OpenMS/TOPP to work on the Windows platform (i.e., Visual Studio 2005 to 2010) and wrote the binary packaging system which enables the creation of a full-grown installer package.

Chapter 3

Simulation of LC-MS Data

Synopsis: *We introduce the most comprehensive simulation software for LC-MS data, provide statistics on the realism of the generated data, and finally show its usefulness for algorithm benchmarking and how parameters of the instrument influence the computational analysis.*

This chapter subsumes and extends the work presented in Bielow et al. [34].

During the development of software for LC-MS data processing the algorithm should be continuously evaluated against suitable benchmark data sets. This allows to refine the algorithm (and its internal parameters) and to create a robust and sensitive solution. Ideally, the benchmark data should be diverse in terms of instrument type, sample complexity, instrument resolution, noise, and other key parameters in order to make the algorithm applicable to a wide range of data. Obviously, each benchmark data set needs to be annotated according to a gold standard (so-called ground truth). This annotation then represents the ideal solution, which the algorithm should be able to reconstruct. Depending on the type of algorithm, different levels of ground truth (GT) are required: feature detection, for example, requires the position, charge, and intensity of peptide signals. Peak picking requires peak annotation and map alignment the positions of corresponding points between two (or more) maps. Furthermore, knowledge of ground truth allows a developer to find subtle errors in the program (i.e., when not all signals contained in the data are identified) which are hard to trace otherwise.

One common solution to establish a ground truth is the use of standard protein mixtures, which are commercially available and can be used to generate data sets with known protein content. These mixtures, however, often lack the complexity inherent to real biological samples, and the data quality (e.g., in terms of contaminants from sample handling) strongly depends on the expertise of the laboratory. Inevitably, such controlled experiments cannot provide the lower levels of a gold standard, e.g., for peak picking, whereas they are useful for peptide identification and quantification problems. Of course, manual annotation can be employed for many levels of ground truth. However, manual annotation is labor-intensive, and the quality of the annotation strongly depends on the skill of the human expert. Some parts of the data may even be impossible to annotate due to high levels of noise or other ambiguities. Public data sets with comprehensive manual annotation are usually not available for many levels of ground truth.

An orthogonal solution to the problem of a benchmark data set is the use of extensive simulation of mass spectrometry data. The use of simulation in the mass spectrometry community began more than 30 years ago. A popular software for ion optics simulation is SIMION [104], which was used to provide the understanding required for the development of new instrument generations. For software development, Morris et al. [20] used simulation to benchmark their new approach for feature extraction and quantification. The Cromwell software, as presented by Coombes et al. [105], was used to create a simulated data set which was then fed to the feature extraction algorithm. Renard et al. [23] implemented a quite simple simulation approach to validate the NITPICK feature finding algorithm. Ipsen and Ebbels [106] published a promising statistical model specific to LC-MS data obtained from TOF instruments. Simulation was used to validate the results. Unfortunately, no software was made available to the community. In 2009, Yang, He, and Yu [24] used a simulated data set from Morris et al. [20] to benchmark different peak picking algorithms. However, simulation is usually given only marginal attention when used to validate an algorithm [20, 23, 24]. The software employed is usually written for this purpose alone and not extensively described. In 2008, Schulz-Trieglaff et al. [26] presented a comprehensive approach to simulating LC-MS data and used it to benchmark different feature detection approaches. Unfortunately, LC-MSsim is not maintained anymore, lacks some desirable functionality (e.g., simulation of fragment spectra and simulation of labeled LC-MS data), and has a few undesirable properties. For example, the retention times of charge variants of a single peptide (or protein) are determined for each feature separately. Since this calculation includes an RT variance term, the charge variants have (significantly) different retention times. This is not the case in real data and should thus be avoided.

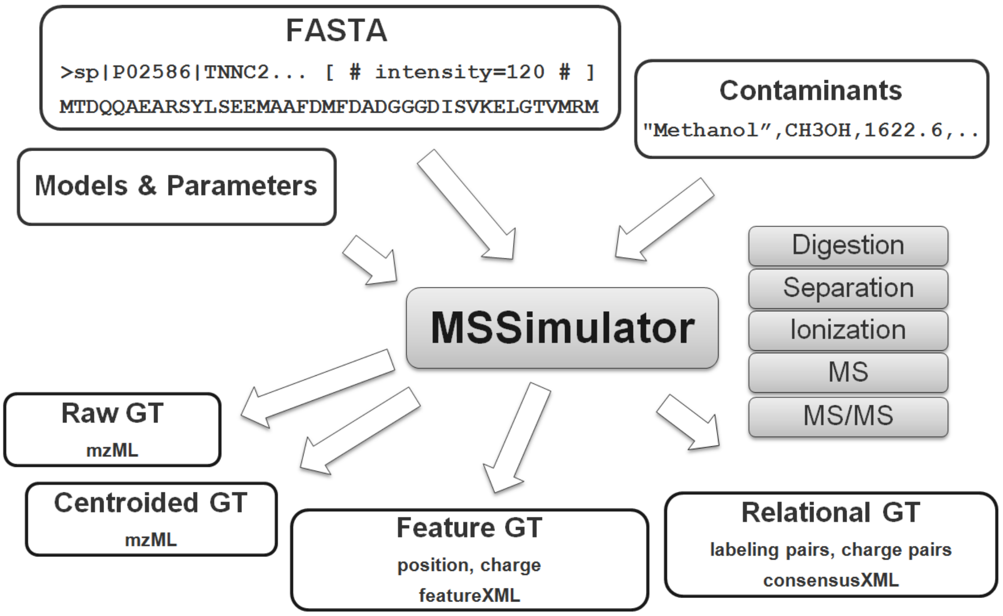


Figure 3.1: Overview of our simulator concept. Peptides/proteins with modification (optional) serve as input, along with parameter settings and models. Result files cover multiple levels of ground truth.

We developed MSSimulator, a simulation software for LC-MS and LC-MS^2 experiments, which is based on LC-MSsim [26] and extends it in many respects. The input for the simulator is a list of proteins (or peptides) and optionally a list of contaminants. For a detailed description of the contaminants file format, see Appendix 6.5. The simulator performs in silico digestion, retention time prediction, ionization prediction, and raw signal simulation (including MS^2) while providing many options to change the properties of the resulting data, such as column conditions, resolution, noise levels, and sampling rate. Protocols for SILAC, ICAT, ^{18}O labeling, iTRAQ or MS^E are available in addition to the usual label-free approach, making MSSimulator the most comprehensive simulator for LC-MS and LC-MS^2 data. As output, the program provides ground truth on multiple levels, which can be used for easy benchmarking and prototyping of algorithms. These levels include raw MS data, centroided MS data, feature positions including peptide sequence, feature relational data for labeled experiments, positions of contaminants, and charge ladder groups. Compared to experimental data, simulation thus not only gives valuable ground truth but is also much faster (usually completed within minutes). Simulation is also unaffected by experimental errors. Since we make heavy use of random number generation, two simulations using the same input data can be configured to yield slightly different results.

An overview of the simulation workflow is illustrated in 3.1.

In the following sections we will describe the basic steps which can be simulated using MSSimulator and the underlying theoretical models. Then we give some examples of how MSSimulator can be used to benchmark algorithms or conduct a large scale simulation on experimental robustness and optimal setup. Parts of this chapter have been published in Bielow et al. [34].

3.1 Methods

MSSimulator is written in C++ as part of the OpenMS [29] framework and is integrated into The OpenMS Proteomics Pipeline (TOPP) [98]. The simulator is configurable via a parameter file, which can be edited using a dedicated GUI shipped with OpenMS. As input we use one or more FASTA files. Optionally, a parameter file containing a (non-default) configuration and a list of contaminants can be provided. A FASTA file contains protein or peptide sequences including modifications[1] and can also be used to provide protein/peptide-specific information like the abundance or a specific retention time. For labeled experiments, one FASTA file for each channel must be present. This allows to specify a different set of proteins and their concentrations for each channel.

The simulation is divided into several submodules, which account for the different steps carried out in a classical LC-MS experiment and will be explained in detail in the following sections.

3.1.1 Digestion

Digestion can be performed in two modes or can be switched off. The first mode does a complete in silico digest using regular expressions, also modeling missed cleavages[2]. To add another level of realism, the second mode uses a model from Siepen et al. [108], which was reimplemented in OpenMS to predict missed cleavages. The current model is based on trypsin data but can be easily adapted, simply by substituting a text file containing the model parameters. To extend the model to other enzymes, the log-likelihood ratio data matrix described in the original paper needs to be computed.

3.1.2 Peptide Separation

As prefractionation techniques, two widely used approaches are available in MSSimulator: Capillary Electrophoresis (CE) and High Performance Liquid Chromatography (HPLC). Both techniques yield separation of peptides according to different properties therefore complementing each other. In CE mode, MSSimulator will predict a migration time based on a theoretical linear model described below whereas for HPLC simulation we use a machine learning approach based on support vector regression.

3.1.3 A Model for Capillary Electrophoresis

In a strong electric field, molecules are separated based on their physicochemical properties which determine their migration time (MT). A molecules' MT is further dependent on the background electrolyte and its properties, e.g., ionic strength, pH, type of ions.

Our migration time model concentrates on simulating the electrophoretic mobility (μ_{ep}) of analytes while electroosmotic flow (μ_{eo}), which is mainly governed by the viscosity of the buffer and the capillary itself, is a parameter provided by the user.

Electrophoretic mobilities and separations are predicted from physicochemical properties of the peptide species, namely net charge and mass. A common model for electrophoretic mobility is

$$\mu_{ep} = q/MW^{\alpha}, \tag{3.1}$$

[1] We support all modifications contained in UniMod [107].

[2] Note that when missed cleavages are used, the completely cleaved peptides will be contained in the sample as well.

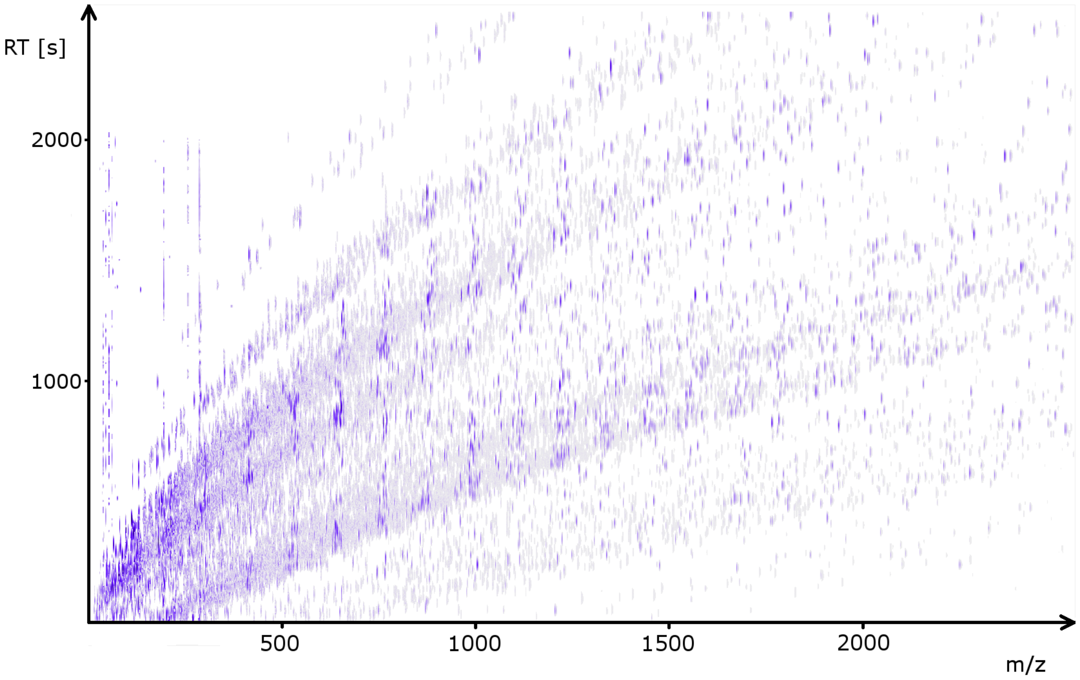


Figure 3.2: Raw CE-MS map of 100 proteins using default CE settings.

where q is the net charge of the ion, MW is its molecular weight, and α is some constant. In a vacuum, an ion's speed is proportional to its net charge when an electric field is applied. In a medium, however, we need to correct for frictional drag (MW^{α} term). The choice of α has been the topic of extensive discussion. The most common values include $\frac{1}{3}$, $\frac{1}{2}$, $\frac{2}{3}$, which all relate to theoretical models. For details on choices of α and charge determination, see Appendix 6.4.

To determine the migration time we compute

$$t = \frac{L_d L_t}{(\mu_{ep} + \mu_{eo})V}, \tag{3.2}$$

where L_d is the distance between injection site and detector, L_t is the total capillary length, and V is the applied voltage (see McLaughlin et al. [109]). Peptides with negative migration times are discarded (but mentioned in a summary statistic).

In contrast to HPLC where elution profiles remain constant across the RT dimension, in CE the peak width increases as a function of migration time due to dispersion factors and decreased mobility. We use a linear model to account for this effect. Figure 3.2 shows an exemplary CE-MS map using our CE model. The typical charge bands can be observed easily.

Prediction of Retention Times in Liquid Chromatography: Schulz-Trieglaff et al. [26] already applied the paired oligo-border kernel (POBK) presented by Pfeifer et al. [49] to accurately predict the retention times for peptides in their simulation. We use the same approach in MSSimulator. A trained model is provided with our software, but training a custom model using MS^2 identifications is easy using the RTModel tool, which is part of TOPP.

A Model for Elution Profile Shape: Peptides eluting from an HPLC- or CE column will usually display an elution profile, which has a Gaussian-like shape. Asymmetric shapes due to fronting or tailing (defined as the widening of the left or right tail of the Gaussian, respectively) are commonly observed. Tailing is more common and has many possible causes [110]. The exponential Gaussian hybrid [111] (EGH) function allows to model asymmetric peaks conveniently.

$$f_{egh}\left(t\right)=\begin{cases}H\exp\left(\frac{-(t-t_R)^2}{2\sigma_g^2+\tau(t-t_R)}\right), & 2\sigma_g^2+\tau\left(t-t_R\right)>0\\ 0, & 2\sigma_g^2+\tau\left(t-t_R\right)\leq 0\end{cases}, \tag{3.3}$$

where t is the retention time, t_R the center of the chromatographic peak, H the peak height, σ_g the standard deviation of the peak, and τ the time constant of the exponential decay.

MSSimulator comes with a set of default values for σ_g and τ as well as the possibility to vary them using a Lorentzian distribution. For more details, see Bielow et al. [34].

To reflect poor chromatographic conditions, the user can also customize the quality of the generated elution profiles by adding uniformly distributed noise.

3.1.4 Peptide Detectability Filter

Although detectability and ionization are closely coupled, we treat them as separate steps during simulation. To account for the effect that not necessarily all peptides ionize with the same efficiency, we include the peptide detectability filter presented by Schulz-Trieglaff et al. [26]. It uses a support vector machine combined with a paired oligo-border kernel to compute the likelihood of each peptide to create a signal in a mass spectrum. The user can define a threshold value – every peptide below the threshold will be discarded. MSSimulator is shipped with a trained model. Customized models can be trained using TOPP's PTModel.

3.1.5 Ionization

We support the two common ionization methods electrospray ionization (ESI) and matrix-assisted laser desorption/ionization (MALDI). For ESI we sample charge states for each peptide entity from a binomial distribution $B(n,p)$ where n is equal to the number of basic residues, plus one for the N-terminal charge, and p is set to 0.8 by default. We also support custom adducts like Na^+ or K^+.

For MALDI we have chosen a discrete distribution of the charge states, with default probability values of $P(q=1)=0.9$ for charge 1 and $P(q=2)=0.1$ for charge 2. The user can customize the charge probabilities according to their needs, specifying as many charge states as desired.

3.1.6 Modeling Peptide Signals in the Mass Spectrum

At this point, a list of peptides annotated with charge, retention time and an elution profile shape was generated. Based on this list, MSSimulator computes the signals for each peptide ion. Each signal has two components, i.e., the shape in the retention time dimension, which has been defined during the simulation of the chromatographic column, and the signal in m/z dimension.

To compute the complete isotopic envelope, MSSimulator uses a fast algorithm [112] implemented in OpenMS. The shape of each individual isotopic peak is a topic of discussion in the literature [113] and can therefore be modeled during the simulation by either a truncated Gaussian or Lorentzian distribution (see Figure 3.3).

The width of the peaks can be controlled by the user with regard to resolution. We additionally provide three models of resolution behavior, which are present in common instruments. Resolution is usually specified at a fixed m/z position p, which we denote R_p, where p is usually 400 Th. Depending on instrument class, resolution might change for other m/z positions q.

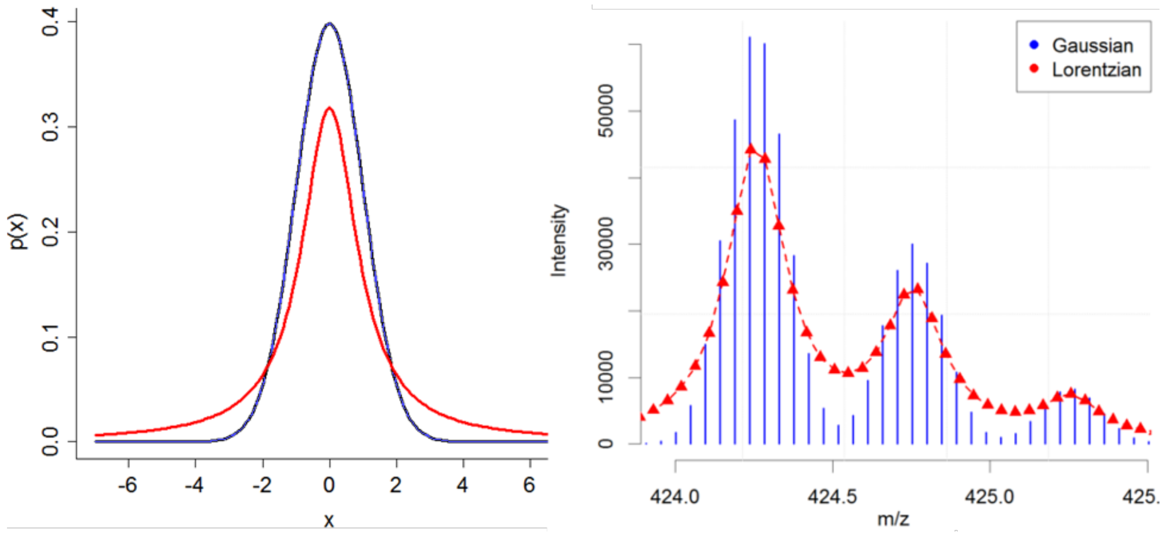


Figure 3.3: Comparison of Gaussian (blue) and Lorentzian (red) peak shape. Left) Theoretical model, Right) A peptide signal at equal resolution, the Gaussian as vertical sticks, Lorentzian as connected lines. Note that due to the broader tails of the Lorentzian, peak height is reduced.

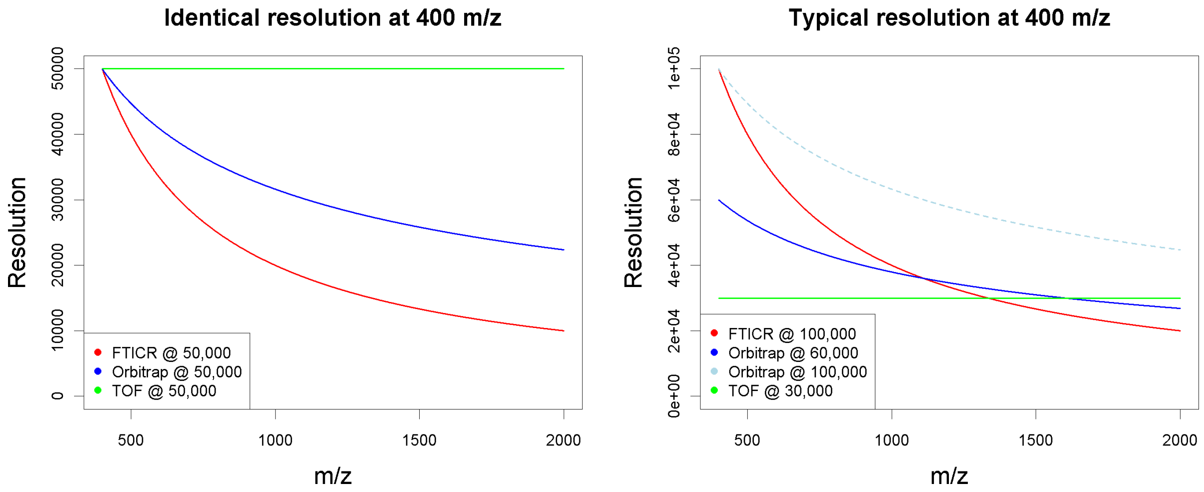


Figure 3.4: Resolution behavior of three common MS instrument types. Left) at the same resolution, Right) at typical resolutions.

Resolution is constant in time-of-flight (TOF) instruments (i.e., $R_q = R_p$); in Fourier transform ion cyclotron resonance (FTICR) instruments it is known to degrade linearly with m/z (i.e., $R_q = R_p \cdot (p/q)$); in Orbitrap mass spectrometers it degrades with the square root of m/z (i.e., $R_q = R_p \cdot (\sqrt{p}/\sqrt{q})$) [70].

See Figure 3.4 for an illustration. Note that at ≈1 500 Th, a TOF with $R = 30\,000$ will create peaks of similar FWHM to an FTICR with $R = 100\,000$.

The estimation of peak width from real Orbitrap data confirms that a square root model fits much better than a linear model (see Figure 3.5). The data was obtained from peak width estimation of the high-resolution peak picker on an Orbitrap data set.

3.1.7 MS^2 Sampling

MSSimulator supports three MS^2 modes: The naive mode creates fixed-intensity peaks for selected ion types (a, b, c, x, y, z), neutral losses, and immonium ions. The second mode uses

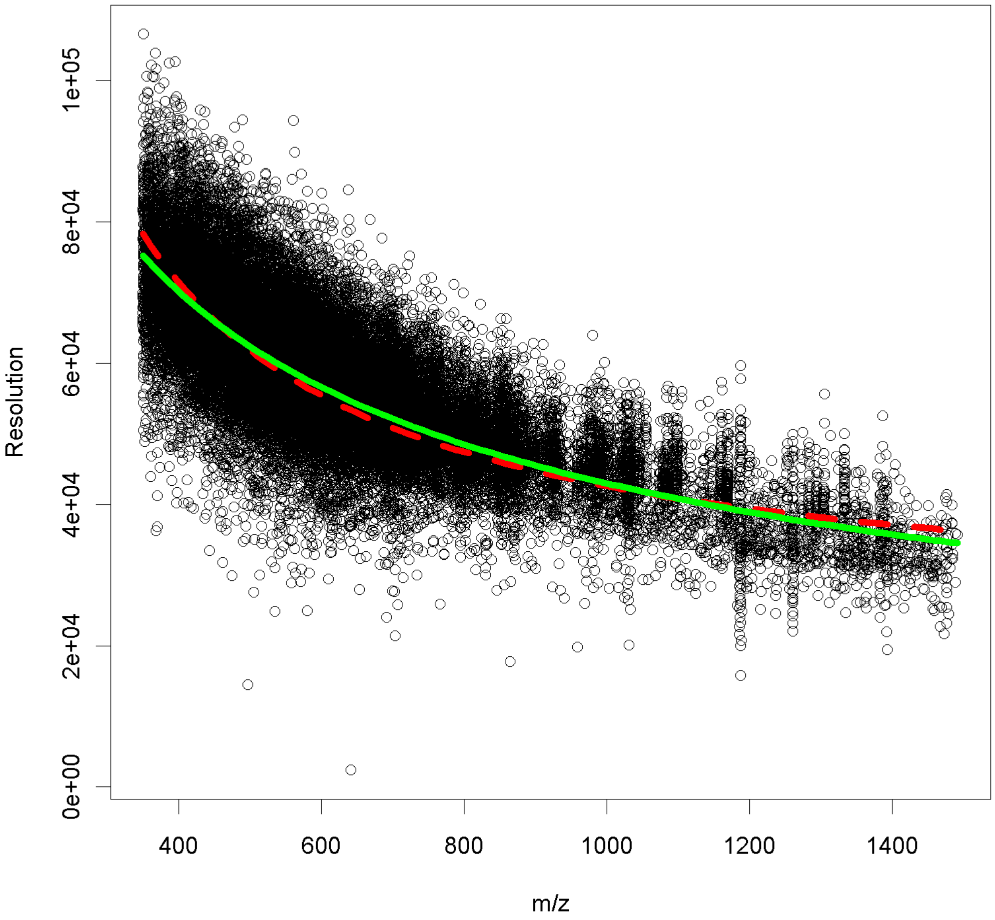


Figure 3.5: Resolution behavior of an Orbitrap. The dashed red line represents the fit of a linear model (as expected from an FTICR instrument), the green line shows a square root model (as expected from an Orbitrap).

a support vector machine-based classifier to predict if a primary ion type is present or not. Neutral losses and charge variants are added using a trained Bayesian model. A support vector regression (SVR) constitutes the third mode and additionally allows to predict the intensity of primary ion types (within five intensity bins). We provide models (pre-trained on CID data) for the two support vector machine-based modes which support precursor charges from one to three. For more details, see [34].

MS^2 precursor selection is based on data-dependent acquisition. A user-defined number of high-intensity precursors are automatically selected from the preceding MS^1 scan. Accepted precursor charges and the width of the isolation window can be changed by the user.

Simulating MS^E Data: Concurrent peptide fragmentation (i.e., MS^E) is an emerging technique in proteomics which could revolutionize the way peptides are identified and quantified. Currently there are very few algorithms capable of analyzing MS^E data, e.g., Elution Time Ion Sequencing (ETISEQ) [18]. This fragmentation technique has been proposed in the metabonomics community several years ago [114], but manual analysis is still prevalent. By providing simulated data we hope to facilitate algorithm development as the simulator provides an easy means to benchmark the results. MS^E data is generated by alternatively recording data in MS and MS^2 mode. The latter has no restriction on the precursor mass; thus, all ions are fragmented simultaneously. This has the advantage that suboptimal precursor selection is no longer an issue, but it also leads to congested MS^2 spectra which need to be disentangled for proper peptide identification. The simulator will create MS^2 spectra for each peptide currently eluting from the HPLC/CE column according to our fragmentation model. Spectra are scaled in intensity such that MS and MS^2 spectra will display proper elution profiles, which can be used to correlate MS^2 peaks with MS features. Subsequently, the single MS^2 spectra are merged to form the final MS^E spectrum. An example can be seen in Figure 3.6. The peaks are color-coded by precursor.

3.1.8 Labeled Experiments

The simulator contains a framework which allows an easy and fast incorporation of any labeling technique used in mass spectrometry. We currently provide four widely used techniques, namely iTRAQ (isobaric tag for relative and absolute quantitation) [27], SILAC (stable isotope labeling by amino acids in cell culture) [115], ^{18}O labeling [116], and isotope-coded protein label (ICPL) [117] in addition to the usual label-free setup. For each labeled channel a FASTA input file must be given. This allows to model different protein/peptide sets. Optionally retention times, abundances, and modification states can be provided for each channel.

iTRAQ Labeling: The software can be used to simulate iTRAQ MS^2 spectra with arbitrary channel allocation (using 4-plex or 8-plex) and customizable isotope correction matrices (the default being the matrix provided by Applied Biosystems). The labeling efficiency of tyrosine residues can be changed as desired and has a default efficiency of 30%. A peptide containing a Y residue will be split into two sibling peptides with different masses, each with an abundance reflecting labeling efficiency. N-terminus and lysine residues are assumed to be fully labeled. The MS^2 spectra generated in iTRAQ mode differ from normal MS^2 spectra in that they contain the reporter ions in the m/z range from 113-121 Th and that the fragment ions are 145 Da heavier for every iTRAQ-modified amino acid they contain. Fragment ions with partially or even completely cleaved iTRAQ tags seem to be missing from the iTRAQ spectra we examined.

Stable Isotope Labeling by Amino Acids in Cell Culture:

MSSimulator currently supports two- and three-channel SILAC labeling. In the following, mass shifts will be shown in brackets. By default, the medium SILAC channel features a modified

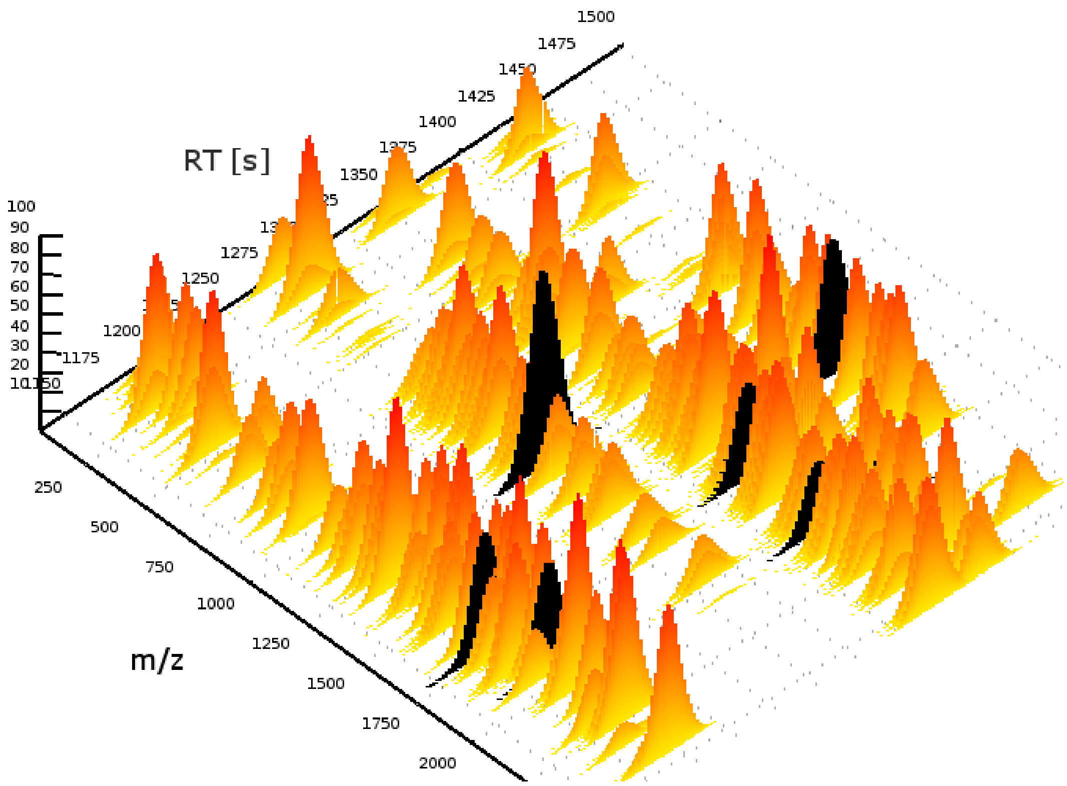


Figure 3.6: Color-coded detail of MS^E spectra containing seven precursor species (black). Intensities are scaled to 100% for MS and MS^2 spectra.

lysine (≈4.02 Da) and arginine (≈6.02 Da), the (optional) heavy SILAC channel an even heavier modified lysine (≈8.01 Da) and arginine (≈10 Da). Complete incorporation of the label into the labeled channel is assumed.

Isotope-coded Protein Label Labeling: ICPL labeling is usually performed on the protein level and yields a mass shift visible on the MS^1 level. Up to three channels are supported, in which all lysine residues and the N-terminal are labeled. The mass shift is therefore sequence-dependent, but upon tryptic digestion without missed cleavages, only one lysine should be present. The protein's N-terminal peptide carries an additional modification. We also allow ICPL labeling after digestion such that all peptides carry an N-terminal modification.

^{18}O Stable Isotope Labeling:

The ^{18}O labeling protocol uses inexpensive, stable ^{18}O isotopes, which are incorporated into the C-terminal of a peptide during protein digestion in exchange for ^{16}O. Complete labeling is achieved when two heavy isotopes are incorporated, introducing a mass shift of 4 Da. Incomplete labeling or back-exchange can lead to mono- (mass shift of 2 Da) or unlabeled peptides. Given a labeling efficiency f, the concentration of unlabeled B_0, mono- B_1, and dilabeled B_2 peptides is computed from the total concentration B of the labeled channel using the kinetic model of Ramos-Fernández, López-Ferrer, and Vázquez [118]:

$$B_2 = Bf^2 \tag{3.4}$$

$$B_0 = B(1-f)^2 \tag{3.5}$$

$$B_1 = B2f(1-f) \tag{3.6}$$

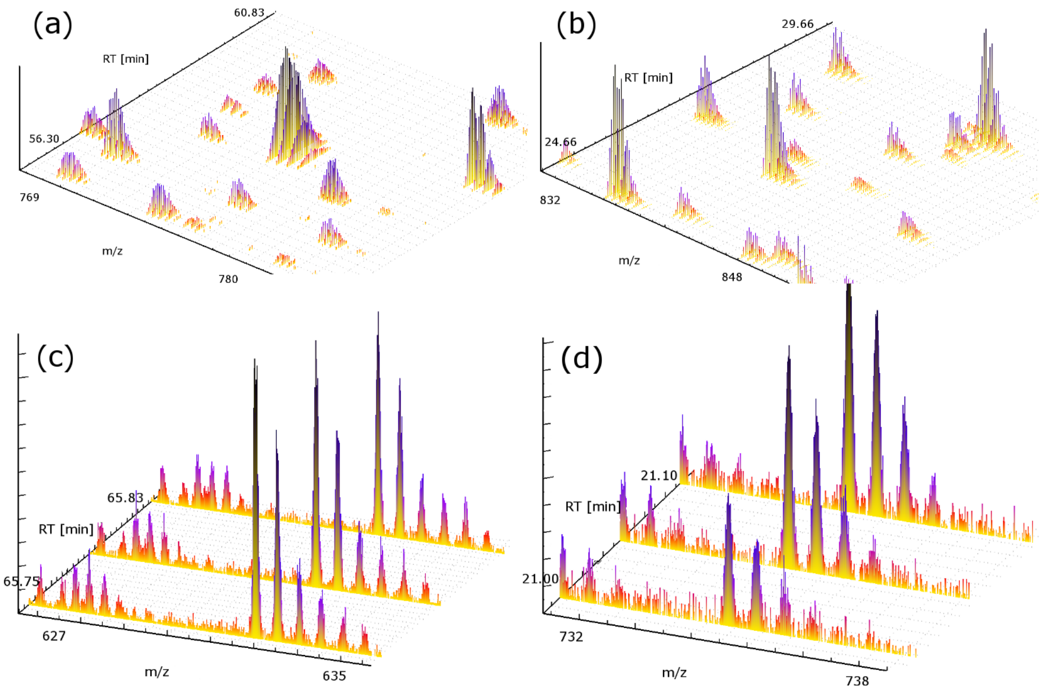


Figure 3.7: Comparison of real and simulated data for FT and Q-TOF instruments. For clarity, data is shown on zoomed regions of an LC-MS map. a) real FT data, b) simulated FT data, c) real Q-TOF data, d) simulated Q-TOF data.

3.1.9 Output

The user can specify one or multiple output files which provide different layers of ground truth. One output file contains the raw MS data in mzML [119] format. A corresponding centroided version of the raw MS data allows to benchmark peak picking algorithms. A feature map (in featureXML format) containing all simulated peptides annotated with charge, charge adducts, and sequence can also be generated. The featureXML file can easily be converted into an Excel sheet or csv (comma-separated values) file. Also, a list of features describing the contaminants in the data set can be requested by the user. Last but not least, MSSimulator can provide files containing the correct associations between the different charge variants of a single peptide and the correct associations between the labeled and label-free versions of the simulated peptides.

3.1.10 Our Contribution

We re-implemented major parts of the predecessor tool LC-MSsim and integrated it into the OpenMS library. The simulation was extended to support more levels of ground truth (charge groups, centroided data, contaminant features), a trained digestion model, a CE model, user-defined retention times and intensities, ionization with adducts, different peak shapes (Gaussian and Lorentzian), resolution and peak widening model (linear, square root, constant), iTRAQ labeling, MS^{E} fragmentation, simulation of contaminants, and detector sampling heuristics.

3.2 Results

Since MSSimulator is highly configurable, it can be adapted to mimic certain instrument types (e.g., Q-TOF or FT instruments). To asses the level of realism of the simulated data we compare simulated data to data sets from the Standard Protein Mix Database [21] (Mix 3, low-res Q-TOF and high-res Fourier Transform (FT) data). The simulation parameters were adapted to closely resemble the experimental conditions of the real data (in terms of protein mix, instrument settings, etc.). After applying the same analysis pipeline (centroiding, feature finding) to both data sets, we find that the number of peptide signals, charge distribution and intensity range are highly comparable. For a visual comparison, see Figure 3.7.

3.2.1 Algorithm Benchmarking

We display several examples of how simulation can be used to benchmark algorithms. The list of applications is numerous, and some more scenarios, carried out by the co-authors, are provided in the accompanying paper [34].

ETISEQ

We used MSSimulator in $\mathrm{MS^E}$ mode to benchmark the ETISEQ software which to our knowledge is the only software publicly available for the analysis of $\mathrm{MS^E}$ data. Since $\mathrm{MS^E}$ data has the inherent property of containing a mixture of fragment ions of possibly hundreds of precursor ions, an interesting criterion for any algorithm trying to reconstruct single-precursor spectra is the number of precursors that can be successfully extracted, such that a search engine is capable of identifying the peptide.

A very simple data set consisting of one protein (P02769, bovine serum albumin) was generated, yielding 114 peptide signals in different charge states (1-3). We disabled simulation of contaminants to make the spectra as clean as possible. MS and $\mathrm{MS^E}$ spectra were generated alternatingly. Additionally, the simulator was configured to create "debug" $\mathrm{MS^2}$ spectra, which can be used as a ground truth when assessing the disentangled ETISEQ spectra. All spectra ($\mathrm{MS^1}$ and $\mathrm{MS^E}$) were generated as centroided data with no missing peaks. Elution profile distortion was disabled, which ensures perfect elution profiles for all features in both $\mathrm{MS^1}$ and $\mathrm{MS^E}$. The generation of neutral loss ions (water and ammonia), immonium ions, and precursor ions was enabled during simulation of $\mathrm{MS^2}$ spectra in order to increase identification rates during database search (using X!Tandem).

Initial tests using the ETISEQ web interface[3] revealed faulty reconstruction of precursor positions when the input spectra were not sorted by RT in addition to XML formatting issues of the ETISEQ output file. Fortunately, the authors of ETISEQ provided a patched version of the ETISEQ algorithm which fixes the aforementioned issues and allows access to more algorithm parameters.

In order to assess the ability of ETISEQ to reconstruct $\mathrm{MS^2}$ spectra from complex $\mathrm{MS^E}$ spectra, the following parameters were modified: We disabled ion exclusion (i.e., $x = 0$) and chose a high value of $n = 15$ (the maximum number of spectra to be reconstructed from a single $\mathrm{MS^E}$ spectrum). To allow for reconstruction of low intensity spectra, we set the minimum relative intensity for parent and fragment ions to zero.

On our dataset ETISEQ successfully reconstructed precursor positions for every simulated feature. Since ETISEQ has no knowledge about the true peptide sequence, usually multiple putative monoisotopic peaks are reconstructed. Many of these spectra yield identical sequences iff a wide precursor tolerance window is selected during spectra identification. It is not clear if ETISEQ uses an averagine approach to match isotope pattern. The data suggests that the

[3] http://www.cancerresearch.unsw.edu.au/CRCWeb.nsf/page/Elution+time+ion+sequencing

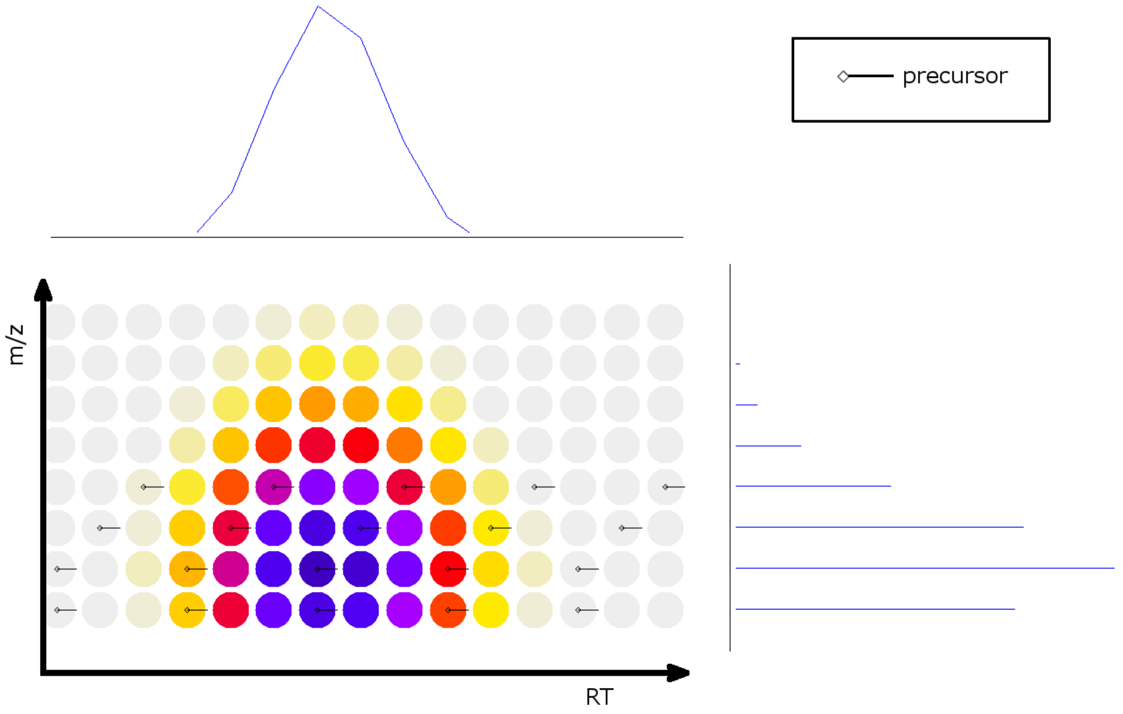


Figure 3.8: Location of precursor positions as reconstructed by ETISEQ for a single feature of charge two and weight of ≈2 434 Da. Projections in both RT and m/z are shown on top and right side. Not only the true monoisotopic peak (lowest position) is chosen, but also the first, second, and even third isotope peak. For this rather heavy peptide the deviation of all but the true isotope pattern to an averagine model are rather large.

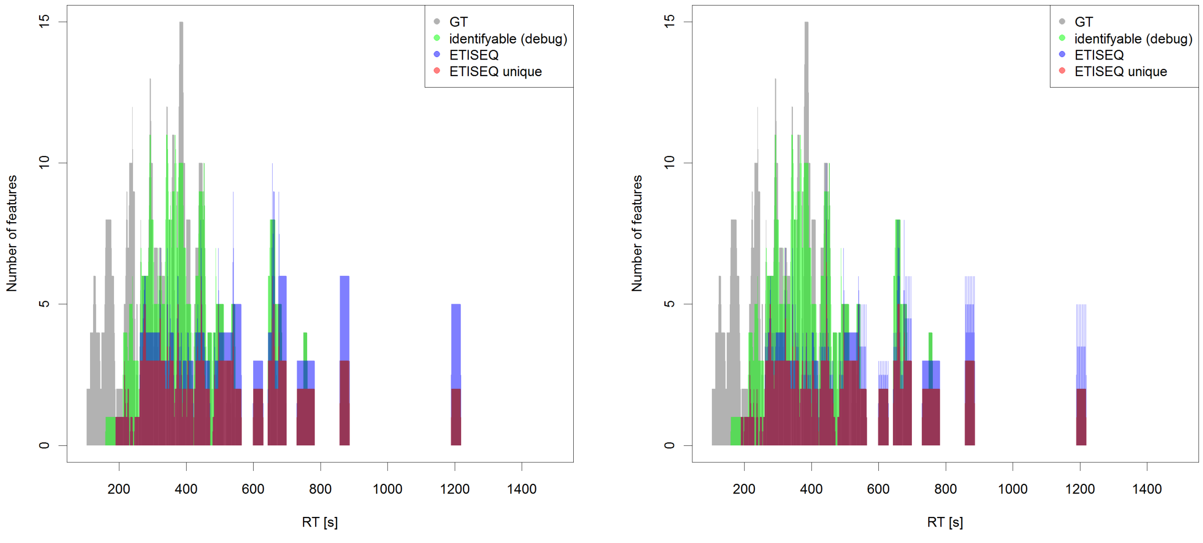
acceptance threshold for any isotope pattern is low, since many putative precursors are reconstructed. See Figure 3.8 for an example of precursor positions.
To reduce the number of putative monoisotopic peaks in MS^1 (and thus the number of redundant reconstructed spectra) an improved algorithm could use a heuristic which searches for a maximal pairing of b-y ions (or any other dominant ion types), or use an averagine model with appropriate model fitting thresholds.

Even though we have simulated MS^2 spectra available as ground truth, not all spectra are identifiable by a search engine. Therefore, we compute an unbiased set of identifiable spectra by submitting all debug spectra to the same search engine (X!Tandem) that is later used for the spectra reconstructed by ETISEQ. For X!Tandem we chose a precursor mass tolerance window of 2.5 Da to allow for small errors in precursor reconstruction. All resulting spectra were searched against a combined database containing 38 717 proteins, as described in the ETISEQ publication, and filtered at 1% FDR using a decoy database approach. The overall identification rate of debug spectra was 75% after 1% FDR filtering. This can serve as the optimal result obtainable by ETISEQ if reconstruction is perfect. We report recall, defined as

$$recall = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}, \tag{3.7}$$

and precision, defined as

$$precision = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}. \tag{3.8}$$



(a)

(b)

Figure 3.9: Number of concurrently eluting peptides over time. a) shows an overlay of ground truth (grey), identifiable peptides (green), peptides identfied from spectra reconstructed by ETISEQ (blue), and unique peptides in terms of sequence and charge of the latter (red). b) shows the results using $x = 1$ for ETISEQ.

of the peptide sequences reconstructed by ETISEQ (while requiring charge to be correct as well in order to be counted as true positive).

Figure 3.9a shows that the density of features in RT is highly heterogeneous, but never exceeds our selection of $n = 15$. One can clearly see that some debug spectra cannot be identified (especially in lower RT regions). Thus, computing an unbiased set of identifiable spectra is clearly advisable. ETISEQ was not able to successfully reconstruct more than seven unique peptide sequences from a single MS^E spectrum (counting distinct charge states as separate results). In most cases only three unique peptide sequences were reconstructed. The number of reconstructed MS^2 spectra is higher, though, since redundant spectra were generated which only differ in the position of the precursor (see Figure 3.8). Overall recall of ETISEQ-reconstructed spectra compared to ground truth was ≈32% (less than half of the identifiable spectra). Precision was 100%, i.e., no wrong peptide sequences were reconstructed. As already shown, selecting a global number n of precursors to reconstruct per MS^E spectrum for the whole experiment is not trivial, because feature density varies significantly with RT. To test if the ion exclusion parameter x can improve performance, we set $x = 1$. The results are shown in Figure 3.9b. Excluding ions in subsequent scans does not improve performance at any RT position. It rather leads to zigzag patterns due to exclusion of previously successful candidates.

Investigating which spectra could not be reconstructed by ETISEQ, we found no dependency on RT or peptide length (data not shown). Fragment ion intensity seemed to have a significant influence iff many high and low abundant precursors were eluting at the same time. However, for only few precursors with large intensity span ETISEQ is able to reconstruct all charge variants, e.g., see Figure 3.9a (≈870 Th) for a single peptide (no other peptides in RT range) in multiple charge states. In more dense regions low abundance features are not reconstructed, despite a high n and no intensity thresholds. See Figure 3.10 for the intensity distribution of identified and unidentified spectra. In dense regions most features' apexes will inherently be close to each other. For our dataset the fraction of features (disregarding charge variants) whose apex is at

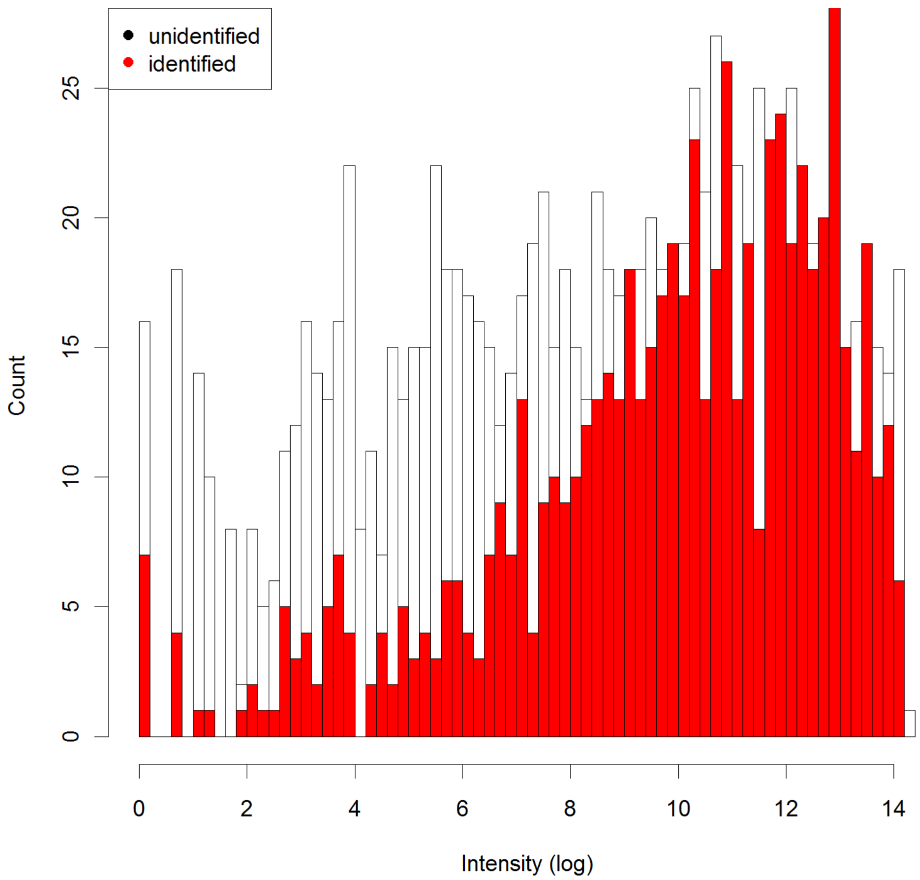


Figure 3.10: Intensity distribution of spectra whose reconstruction was successful (red) versus all simulated spectra. A preference for high intensity spectra is clearly visible.

least 1 s from its closest neighbor is 82%, for 0.5 s this number rises to 97%. Thus, most features should be distinguishable by apex alone. Additionally, the simulator uses random elution profile shapes, which should enable the Pearson correlation analysis to distinguish features which are close in RT. Nevertheless, changing the threshold for the Fourier transform lag to 0 and Pearson correlation to 0.9 did not change the ETISEQ results (defaults are 3 and 0.7, respectively). This is unfortunate, since the performance of the algorithm is critically dependent on its ability to distinguish features which are close in RT, possibly with similar elution profile shapes.

To summarize, our data suggests that ETISEQ is currently limited in the number of peptides that can be reconstructed from a single MS^E spectrum. Also, high abundance precursors are preferred over low abundance precursors in dense regions. Furthermore, the precursor selection strategy has potential for improvement, since isotope peaks are often selected in addition to the monoisotopic peak. According to the ETISEQ paradigm for peak assignment, all fragment ions of a peptide with multiple precursors are copied to all precursors from other peptides. This could introduce a serious amount of noise, therefore decreasing identification rates.

Map Alignment

In this study we aim to benchmark the ability of a map alignment strategy to correct for a retention time distortion between two simulated data sets when the overlap of sample content is varied. We used the simulator to create feature maps of decreasing overlap in terms of protein content but constant number of features (≈4 000) and applied the TOPP MapAligner

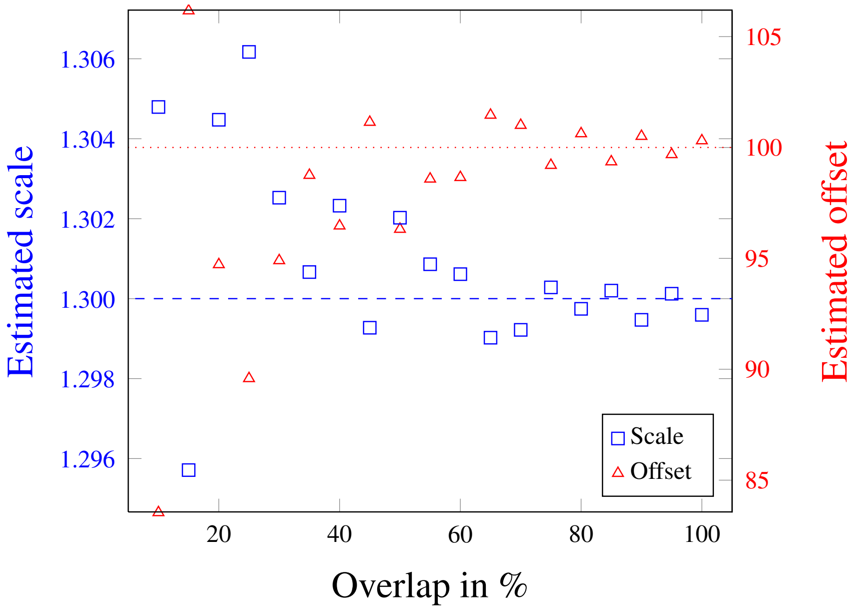


Figure 3.11: Quality of alignment when altering peptide overlap between the two data sets. Red triangles indicate the reconstructed offset in comparison to the simulated offset (red dotted line). Blue squares indicate the reconstructed scale in comparison to the simulated scale (blue dashed line).

tool to reconstruct the affine retention time shift plus a local Gaussian distributed distortion. We chose $offset = 100$, $scale = 1.3$, and a local Gaussian distortion with $sd = 3$ for each feature. This scenario can provide insight on how many corresponding features (i.e., alignment anchors) are needed to reconstruct the correct alignment. Inefficient feature finding and/or poor chromatographic conditions may lead to poor overlap of (replicate) experiments, thus a robust algorithm is required to reconstruct the RT shift. The results show that even a very small overlap does allow for a reliable estimation of the true transformation (see Figure 3.11).

3.2.2 Experimental Settings Optimization

The exact conditions under which LC-MS experiments are performed, e.g., which gradient length, column type, resolution, etc. are used, are important for the success of the scientific endeavor. Simulation cannot only be used for algorithm debugging, optimization and performance evaluation but also for predictive purposes, namely for the optimization of experimental settings. Given a certain sample of known complexity, experimentalists choose LC and MS settings based on previous experience. In this case simulation can help to determine if any significant improvement can be gained by increasing resolution, LC time, or by using replicates with exclusion list.

Starting from a default configuration for an analysis pipeline (see Figure 3.12), we are going to change prominent parameters and evaluate their influence on precision and recall.

The list of parameters along with all tested values for simulation is shown in Table 3.1. Values in bold show the default setting used for all dimensions except for the one dimension that is used for iteration.

To speed up evaluation, one could opt to work on a subset of the map; however, this can bias the results for two reasons. First, the feature spread in RT and m/z dimension is nonhomogeneous and secondly, evaluation software might depend on a map large enough to estimate

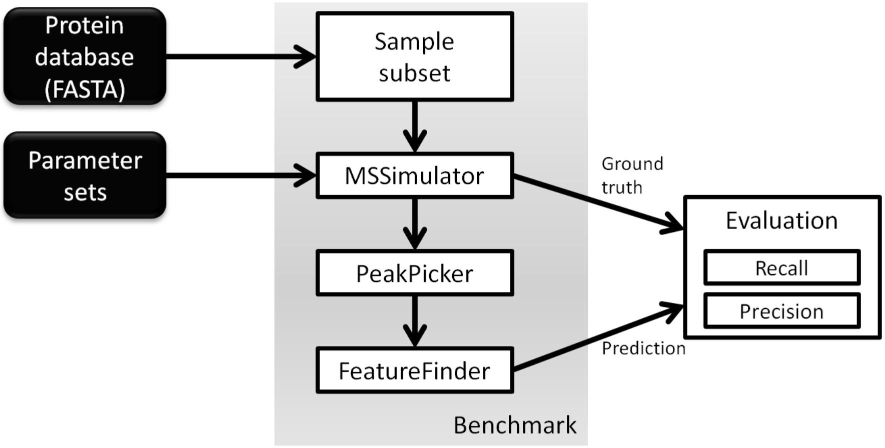


Figure 3.12: Evaluation workflow allowing to test multiple simulation parameters and evaluate their effect on precision and recall.

Table 3.1: Iteratable simulation parameters. Values in bold represent the default.

parameter	values
protein count	10 50 100 500 **1 000** 3 000 10 000
dyn. range	0.1 0.2 0.5 1 1.5 **3** 6 9
resolution	5 000 7 500 10 000 20 000 40 000 **60 000** 100 000 200 000
RT (min)	5 30 60 **90** 120 240
shot noise	0 **0.01** 0.02 0.05 0.1 0.5 1 2 3

signal-to-noise ratios robustly. Thus the location and number of features found might differ (as is the case for the OpenMS centroided FeatureFinder – data not shown).

We report recall (see Eq. 3.7) and precision (see Eq. 3.8), since only true positives, false positives, and false negatives are available. The number of true negatives is infinite.

Figure 3.13 summarizes the results.

According to the simulation, dynamic range and noise levels do not negatively influence precision and recall of the OpenMS FeatureFinder (centroided). The computation of signal strength from peptide abundances uses a linear model during simulation. The dynamic range is therefore linear as well. This indicates that the feature finding algorithm can identify weak features amidst high-intensity features. However, higher spectral density where more overlapping features occur will drastically decrease performance. One way to avoid overlapping features is to increase the gradient length of the HPLC. Not surprisingly, the algorithm is then able to recover more features and suffers less from false positives. Larger gradient lengths can balance higher sample complexity. Interestingly, increasing resolution does not change precision but increases recall due to more narrow raw peaks, which, once they are centroided, allow the algorithm to disentangle interleaved features. If overlapping peaks cannot be resolved, the observed isotope distribution will deviate strongly from the expected averagine distribution, leading to rejection of the putative feature as peptide signal. This increases the false negative count; thus, recall is decreasing.

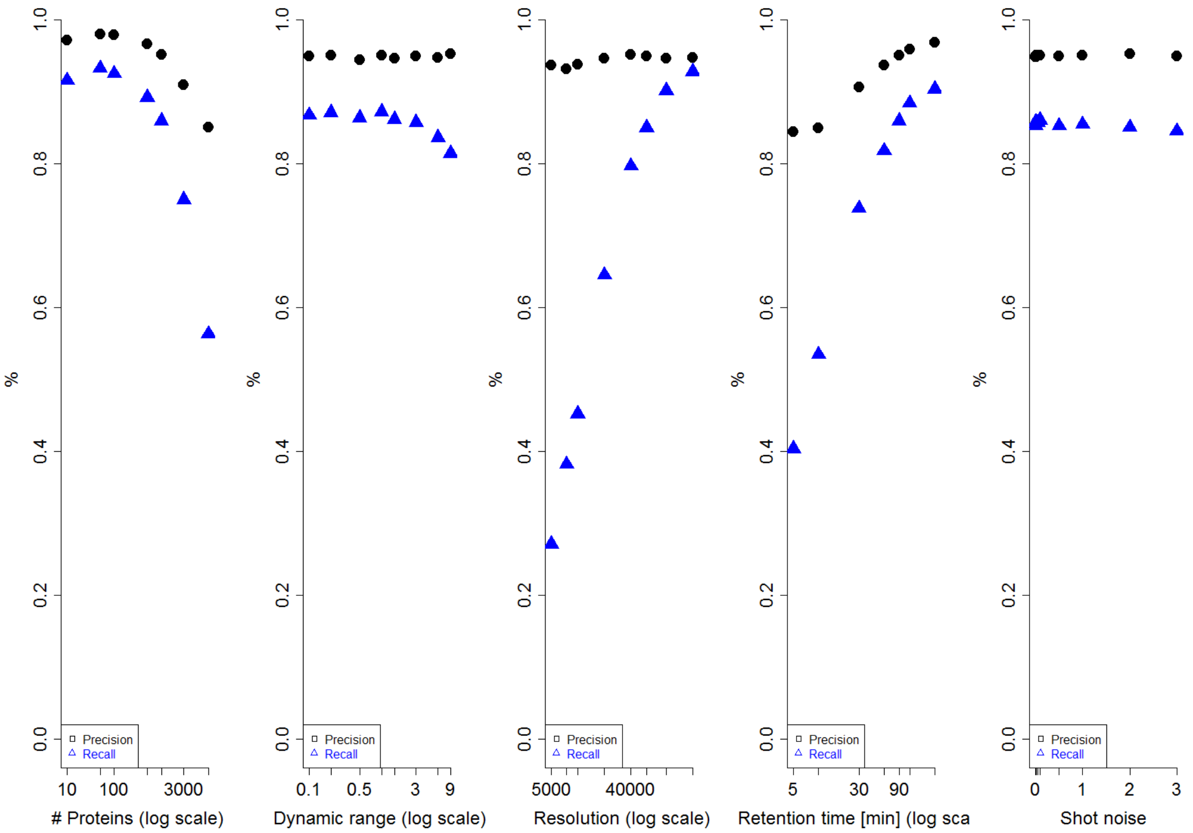


Figure 3.13: Recall and precision for multiple parameter sweeps.

3.3 Discussion

MSSimulator is the most extensive collection of algorithms and models for MS simulation and allows for easy algorithm validation on a broad range of conditions, opening a wide range of benchmarking scenarios that can easily be automated. The availability of a ground truth reduces the need for expensive manual validation on real data sets. For each simulation step the simulator has several models (e.g., CE and HPLC for separation, MALDI and ESI for ionization, three resolution models for mass measurement), it allows for arbitrary modifications and contaminants. In addition to the four labeling techniques which are currently supported, future labeling techniques can be added quickly by implementing a powerful labeling interface.

We have shown that our simulated data is very similar to real data and allows easy validation of existing algorithms. Compared to experimental data, simulation thus not only provides valuable ground truth but is also much faster to generate and unaffected by experimental errors. Simulation of different experimental conditions can be used to predict which parameters have the largest influence on a subsequent computational analysis (e.g., feature finding). However, in order to find the optimal experimental parameters a cost function is required (e.g., is increasing the gradient from four to six hours worth a gain of 20% in feature count). This cost function is most certainly subject to change for every laboratory or project. Missing support for ionization efficiency, trapping capacity, etc., during simulation is currently the reason why the parameter ranges will most likely not be comparable (i.e., under identical conditions a real dataset with 5 000 proteins will have a different amount of feature overlap than the corresponding simulated dataset). Another model trained on real data would be required to account for this during simulation.

Future extensions might include, but are not limited to, automatic estimation of simulation parameters (e.g., resolution, sampling rate, noise level) from real data allowing to quickly generate benchmark data for analysis software, quantitative prediction of ionizability, incorporation of additional noise models and ion statistics and more instrument-specific properties (e.g., shoulder peaks on FT instruments – see Subsection 2.3.2). Due to the broad support of different levels of ground truth and a wide variety of models, the simulator could also be used to re-evaluate published algorithms whose performance was assessed using a feature-limited and special purpose simulation tool. A comparison might reveal significant differences in the performance of the algorithm, pointing to violated model assumptions (e.g., shape models, data complexity).

Chapter 4

Decharging of Charge Variants

Synopsis: *A new algorithm for charge variant detection based on integer linear programming is introduced and evaluated. We show that it outperforms existing methods and is robust to missing data in simulated experiments.*

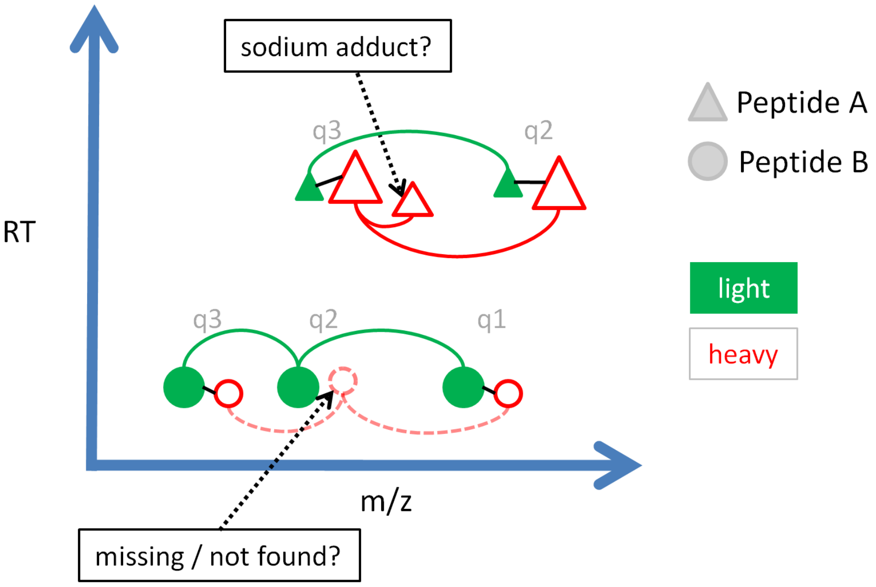


Figure 4.1: Schematic illustration of (simple) decharging problem. Two peptide species in different, rather small charge states are shown. One peptide has an additional variant featuring a sodium adduct. In addition, both peptides exist in a light and heavy state, i.e., peptides are labeled.

This chapter subsumes and extends the work presented in Bielow et al. [35].

In electrospray ionization mass spectrometry (ESI-MS), peptide and protein ions are usually observed in multiple charge states. Peptides and proteins are measured in positive mode, making protonation the primary mechanism of charge acquisition. The number of charges depends on experimental conditions and on the length of the amino acid sequence. Therefore, peptides usually carry a small number of charges in ESI, usually two, whereas proteins can reach charges above 50. Adduction with other ions such as sodium or potassium leads to further partitioning and more complex signal patterns for a single species, adding to spectra density and complicating the derivation of quantitative information from the mass spectra. Labeling strategies targeting the MS^1 level further aggravate this situation since multiple samples must be represented simultaneously. For an example, see Figure 4.1.

We developed an *integer linear programming* (ILP) approach which can cluster signals belonging to the same peptide or protein. Our widely applicable and general approach models all possible shifts of signals along the m/z axis, taking into account different charge states of the compound, the presence of adducts (e.g., potassium or sodium), and/or a fixed mass label (e.g., from ICAT or SILAC), or any combination of the above. We show that our approach can be used to infer more features in labeled data sets, correct wrong charge assignments even for high-resolution MS data, improve mass precision, compute intact protein masses from large protein charge ladders in complex mixtures, cluster charged species with several adduct types, and is robust against missing values in simulation studies.

Figure 4.2a shows the raw map of two peptides of similar mass but different retention time. Usually, the raw data is subjected to algorithms for data reduction, and the measured signal is reduced to a single data point – a feature (see Section 2.6).

The problem of multiply charged peptide species is usually only present in ESI whereas in MALDI, peptides mostly receive one charge, rarely two. However, even in MALDI, experimental conditions can be changed such that higher charges are observed [30]. Obviously, it is crucial to find all signals originating from a peptide. Unfortunately, this task is more difficult in practice

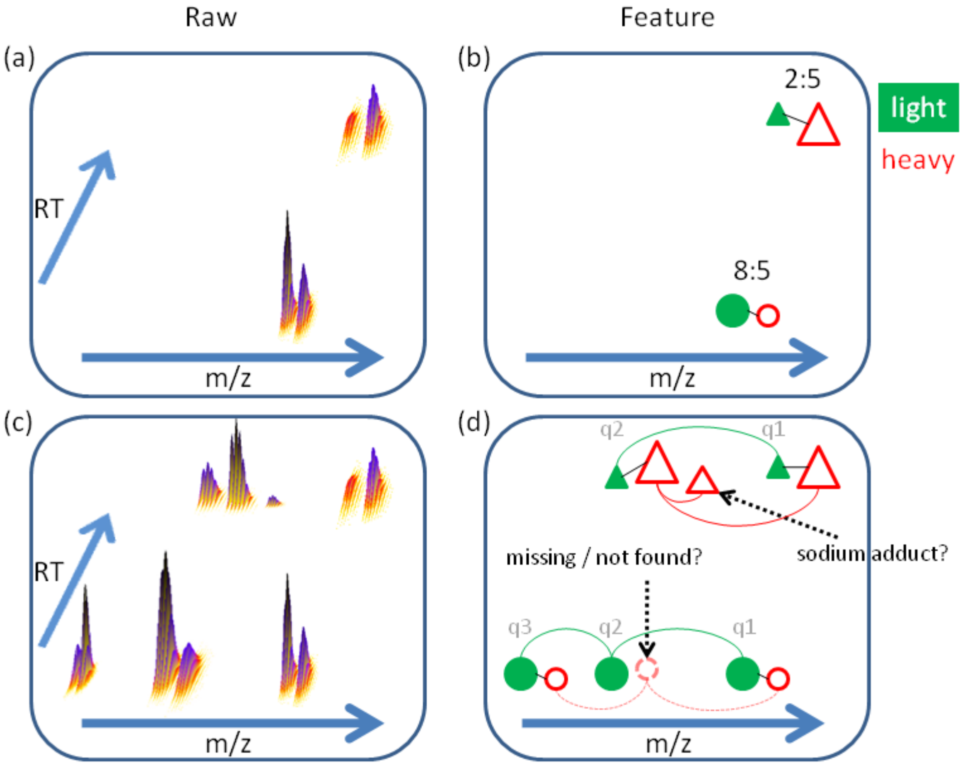


Figure 4.2: Schematic illustration of ESI spectral cluttering due to multiple charge states and adduct formation in experiments involving light and heavy-labeled species. a) Two ideal peptide signals (top and bottom) eluting from a chromatographic system, each showing a light and heavy analog. b) Features identified from raw signal on the left with the resulting intensity ratio between labeled and unlabeled compounds. c) The same peptides as in sub-figure A but spread across several charge states. Even an adduct can be observed, which is related to the high-intensity heavy peptide at higher retention time. d) Charge ladders and charge states indicated at the feature level for sub-figure C.

than implied in Figure 4.2 since samples are usually highly complex and ambiguities need to be resolved. Figure 4.2c, for example, shows the (realistic) case of two and three charge variants of each peptide with the addition of a sodium adduct. Figure 4.2d then shows the respective feature map in which one feature is missing, which is also quite common in practice due to noisy data or algorithms not being able to detect all signals correctly. While charge ladders of peptides are usually rather small (a few charge variants), proteins give rise to a much broader charge distribution due to their size and can easily reach charges of 30 and even above 50. As for peptides, proteins can carry adducts such as sodium.

The problem of clustering differently charged species from the same compound in ESI spectra is often referred to as deconvolution (although this is misleading – mathematically speaking), decharging (although experimentalists usually interpret this as a reduction of the average charge state [61, 65]), or simply disentanglement. Deconvolution is also sometimes used synonymously for deisotoping [120] or resolving overlapping shapes [121]. We thus suggest the name *decharging* for a reduction of multiple (deisotoped) species of the same analyte with different charge adducts to a single zero-charge signal.

In labeling approaches (see Subsection 2.6.2), usually no decharging is applied. Instead, signals of different labeling states with equal charge are grouped and compared directly, which results in redundant information if multiple charge states are present. In both label and label-free approaches, the quantification is further aggravated by the presence of adducts with ubiquitous ions, such as sodium and potassium, whose occurrence depends on experimental conditions, e.g., usage of salts during HPLC or capillary electrophoresis (CE). Peptide signals incorporating such adducts are usually low in abundance but will nevertheless reduce the ion count of proton-only signals.

Inferring the correct mass of a peptide or protein from charge states and charge ladders has been an active research topic from the onset of application of ESI-MS in proteomics. Early approaches targeting undigested protein samples use “global” information, i.e., multiple signals of different charge states, to infer the mass and are best suited for mass spectra containing only a few analytes. With the emergence of high resolution instruments it became possible to use “local” information, i.e., isotope patterns, to infer charge, which is sometimes the only option to infer mass if an analyte is only present in a single charge species. For protein spectra, Mann, Meng, and Fenn [31] proposed an algorithm to fold a spectrum into mass space, thus eliminating charge ladders. Although this greatly improved mass precision, the algorithm can only deal with few analytes in one spectrum and gives rise to artifact peaks. This algorithm was further improved by Reinhold and Reinhold [122], who reduced artifact peaks by using an entropy-based measure at the cost of requiring a model distribution of charge ladders which is applied to all masses under investigation and a loss of the peak height-abundance relationship. For broad-range MALDI spectra, a heuristic approach [30] working on single spectra was devised, which can cluster multiple charge states considering only H^+ but relies on MALDI-specific rules not applicable to ESI. The widely known ZScore-Algorithm [32] features either local or global decharging but not both simultaneously. It can deal with complex single (stick) spectra but might also produce artifact peaks due to spectral noise. A similar algorithm along with a brief review was published by Zheng et al. [123]. Du and Angeletti [120] infer charge from local isotope peaks and cluster all species projecting onto the same mass. This approach, however, is prone to incorrect charge assignment during charge estimation and requires a threshold parameter. One algorithm that attempts to make use of global and local information was published by Wehofsky in 2002 [33]. It rewards features with charge q when their sibling of charge $q-1$ is also found. Unfortunately, the algorithm has no notion of retention time, only considers adducts of type

H^+, and relies on identifying gapless charge ladders. Furthermore, if charge and thus mass are estimated incorrectly, the decharged spectrum is neither likely to contain the wrong signal nor the correct one, because wrong charges are not fed back into the input spectrum. MaxQuant [93] creates charge pairs based on retention time correlation and a peptide mass estimate threshold for SILAC-based experiments. ASAPRatio [124] also uses charge pairs in ICAT-type LC-ESI-MS data to improve quantification results. In addition to the two algorithms above, another tool capable of analyzing labeled data is VIPER [95]. It supports arbitrary mass differences (e.g., from ICAT or $^{16}O/^{18}O$ labeling) and can deal with pairs in multiple charge states.

None of the algorithms mentioned above is able to model charge ladders with multiple adduct combinations, e.g., a combination of pure proton adduct species with a proton/sodium species from the same peptide or protein. And except for the more recent ones, they were all designed for undigested analytes producing long charge ladders. In tryptic digests, however, one rarely encounters species with a charge of five or higher. There is a solution to cluster undigested protein degradation products based on an EM algorithm, which is not publicly available [125]. For metabolites an approach using database search accounting for different adducts was recently devised [126]. Additionally, there is the CAMERA software package[1], which groups metabolite mass signals based on rules for mass differences and peak shape comparison [127].

In this work we propose a method for identifying groups of signals belonging to the same compound in labeled or label-free MS data. The algorithm is general in that it models all possible shifts of signals along the m/z axis. These shifts can be induced by a different charge state of the compound, the presence of adducts (e.g., potassium or sodium), the presence of a mass shift due to isotope labeling, or any combination of the above. It allows for an iterative approach (rerunning feature detection on missing charge states or missing pairs in labeled experiments) and can deal with missing data (e.g., gapped charge ladders). We show that by applying our algorithm, several types of errors can be corrected in a feature map, e.g., wrong charge assignment or missing features. Additionally, we can achieve a reduction of data volume, improvements in mass precision, and prevent manual annotation errors and incomplete annotation.

Parts of this chapter have been published in Bielow et al. [35].

4.1 Mathematical Preliminaries

Our approach uses a widely known optimization technique called integer linear programming (ILP). To facilitate understanding, we now give a brief introduction to linear programming (LP), its special form integer linear programming, and some standard notation. This section is mostly based on Bertsimas and Tsitsiklis [128]. To ease reading we will use bold letters for vectors, bold capital letters for matrices, and plain letters for scalars.

In linear programming we seek to minimize an objective function $\boldsymbol{c}'\boldsymbol{x} = \sum_{i=1}^{n} c_i x_i$ where $\boldsymbol{c}$ is a given cost vector and $\boldsymbol{x}$ is an unknown vector of decision variables subject to a set of linear equality and inequality constraints of the form $\boldsymbol{a}'\boldsymbol{x} = b$, $\boldsymbol{a}'\boldsymbol{x} \leq b$ or $\boldsymbol{a}'\boldsymbol{x} \geq b$ where b is a scalar. We can write a set of constraints as $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ where $\boldsymbol{b} = (b_1, ..., b_m)$ is a vector of m constraint bounds and $\boldsymbol{A}$ is a $m \times n$ matrix whose rows are row vectors $\boldsymbol{a}_1, ..., \boldsymbol{a}_m$. Every LP can be brought to standard form, i.e.,

[1] http://www.bioconductor.org/packages/bioc/html/CAMERA.html

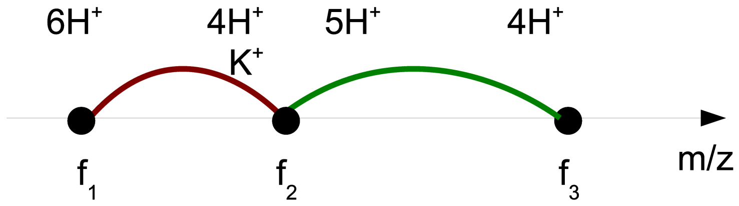


Figure 4.3: Two conflicting edges inducing a constraint due to inconsistent annotation of feature f_2 (implicit H^+ are shown as well).

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{c}'\boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0. \end{aligned}$$

A vector $\boldsymbol{x}$ satisfying all constraints is called a *feasible solution.* If additionally some $\boldsymbol{x}$ minimizes the objective function, $\boldsymbol{x}$ is called an *optimal feasible solution.* Depending on the problem, there can be no, one or multiple optimal solutions. Maximizing $\boldsymbol{c}'\boldsymbol{x}$ is obviously equivalent to minimizing $-\boldsymbol{c}'\boldsymbol{x}$.

The first and most widely used algorithm described by Dantzig [129] to solve an LP is called the simplex algorithm, which has exponential worst-case complexity but performs well in practice.

ILP formulations are exactly the same as LP formulations with the additional constraint that some variables are restricted to take integer values, i.e.,

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{c}'\boldsymbol{x} \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \\ & \boldsymbol{x} \geq 0 \quad , x \text{ integer}. \end{aligned}$$

In general ILP problems are NP-hard and can be solved using a number of algorithms, e.g., branch-and-bound or cutting-plane methods.

4.2 Methods

The input data set is a feature map F as generated by a feature finding algorithm, each feature having at least a retention time and m/z (monoisotopic or average). In addition, it can be advantageous to have an initial charge estimate and an intensity value.

We model our problem as a graph, which lends itself to an ILP formulation. The nodes in the graph correspond to features at a certain RT and m/z and hence to a peptide with a certain charge state, possibly with adducts and/or mass labels. Edges are inserted between pairs of nodes if a certain combination of adducts and charge assignment of the nodes explain the mass difference between the nodes. Each edge carries information on the potential charge of its adjacent nodes and the adducts which are required to explain the resulting mass difference. In Figure 4.3, for example, the edge between f_1 and f_2 is inserted because the mass difference can be explained by the assumption that f_1 has charge 6, f_2 has charge 5, and f_2 has a potassium adduct. The inserted edge hence induces charge states on f_1 and f_2.

For building the graph, we only use the most commonly occurring adducts listed in Table 4.1. Note that this table can be easily modified by the user if required. Simple protonation is the

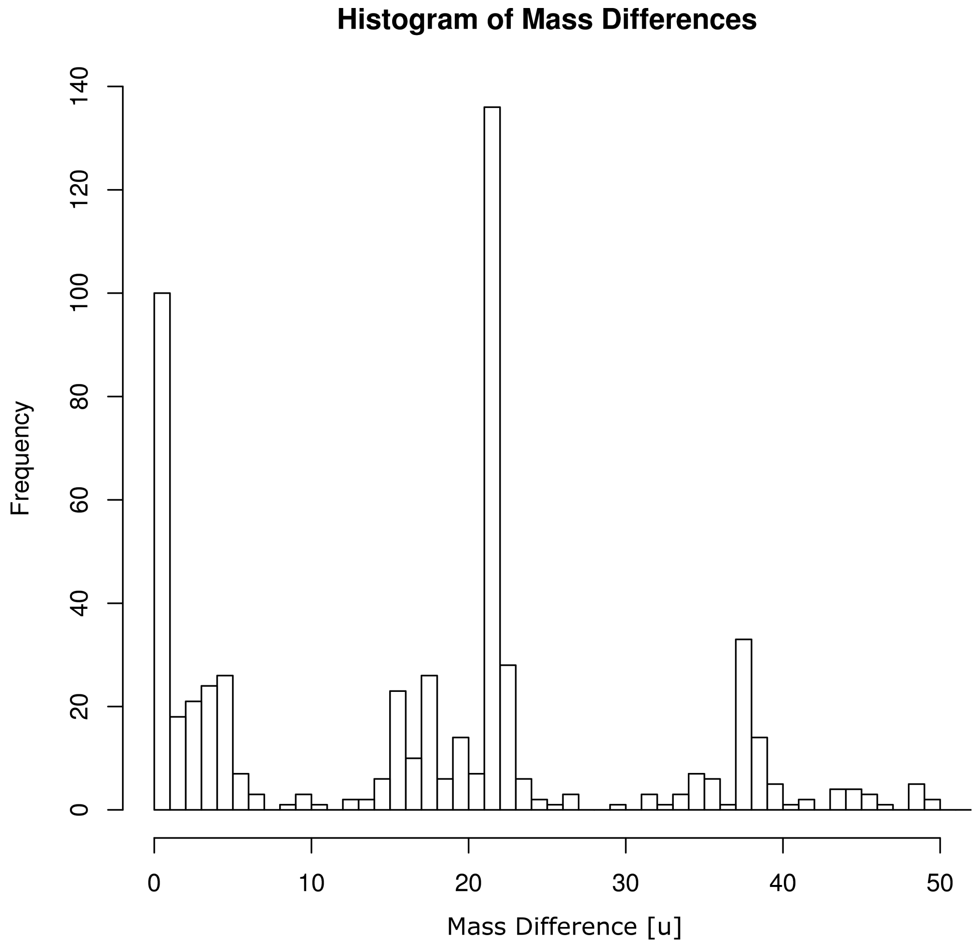


Figure 4.4: Histogram of pairwise mass differences showing evidence of presence of sodium and potassium (obtained from the SPC data set used below).

most common effect (and desirable due to better fragmentation behavior and decreased signal congestion [130]). Non-proton adducts are usually a result of prior prefractionation via CE or HPLC.

Adduct frequencies can be estimated by looking at the histogram of pairwise mass differences of all features (see Figure 4.4). At the masses of Na, K and NH_4 (minus a proton mass each), one can clearly observe clusters which indicate their presence (see Table 4.1 for adduct masses).

Constructing the graph in such a fashion obviously results in conflicting edge descriptions. Each pair of edges adjacent to a node might induce a constraint due to a conflicting charge/adduct combination. For example, Figure 4.3 shows two edges adjacent to f_2 where the left edge assigns four protons and a positively charged potassium ion to f_2 while the right edge assigns 5 protons to f_2. Obviously, only one of these conflicting annotations can be fulfilled at a time. Hence, our goal is to choose a subset of the edges with an overall maximal weight which does not contain any pair of conflicting edges. We compute the optimal subset by solving an ILP. In the following, the method is specified in further detail.

4.2.1 Generating the Adduct Transition Graph

Initially, our algorithm generates a table L of all feasible net adduct transitions, i.e., the subset "lost" on one side and the subset "gained" on the other. Losing a proton and gaining a sodium adduct, for example, can serve as an explanation for an edge between $[M + 2H]^{2+}$ and $[M +$

Table 4.1: Adducts commonly observed in ESI-MS. All adducts occur singly charged, i.e., they are lacking one electron.

name	formula	monoisotopic mass (Da)
hydrogen	H	1.0078250319
ammonium	NH_4	18.05
sodium	Na	22.98976928
potassium	K	38.96370668

Table 4.2: Example for adduct transition table L.

loss	gain	net charge	mass
Na^+	H^+	0	-21.9819
-	H^+	+1	1.0078
-	Na^+	+1	22.9892
H^+	$2Na^+$	+1	44.9712
Na^+	$3H^+$	+2	-19.9674
$2Na^+$	$4H^+$	+2	-41.9493
-	$2H^+$	+2	2.0146
-	HNa^+	+2	23.9965

$H + Na]^{2+}$ ions. The table contains net mass and net charge differences. An example using proton and sodium adducts can be found in Table 4.2. Note that we do not model redundant transitions, i.e., elements that occur on both sides and would cancel each other out. Additionally, each adduct is assigned an a priori probability (e.g., using Figure 4.4), which allows to compute adduct transitions up to a probability threshold which can be chosen generously but avoids adduct transitions which are unlikely to occur (e.g., all-sodium adducts in a charge 5 feature). Furthermore, the list is bound by the charge difference q_{span}, which is the maximum number of charge states that can be bridged by edges in the graph. By default, q_{span} is set to 4, which allows bridging q_3 (charge 3) and q_6 (charge 6) but would not allow to join two nodes with q_3 and q_7. The size of L depends very much on the number of adducts allowed and q_{span} but rarely exceeds 400 entries.

We now construct the adduct transition graph $G = (V, E)$ where V is a set of nodes n_i corresponding to features f_i from the set F. E is a set of undirected edges $e_j = \{n_k, n_l\}$. To generate edges between nodes, the algorithm enumerates all pairwise features within a small RT delta $delta_{RT}$ since charge ladders are a property of ESI and thus have similar RT. However, if method-specific RT shifts are known (e.g., in ICAT pairs), this can easily be accounted for by specifying an adduct's intrinsic RT shift. During enumeration, mass differences are looked up in L, and for all matches an edge containing the putative charge and adduct of the left and right node is inserted as well as a score which serves as an edge weight. All charges not explicitly explained by the adduct transition are implicitly modeled as H^+ and stored in the edge as well. Obviously, edges with adduct transitions that require a feature to take up more charges than allocated are not realized. Edges are weighted by the product of probabilities of adducts which are required to explain the mass difference (see Table 4.2). However, a more involved scoring scheme can be easily implemented, which could, for example, account for mass and RT deltas, feature quality, and violation of a feature's local charge prediction.

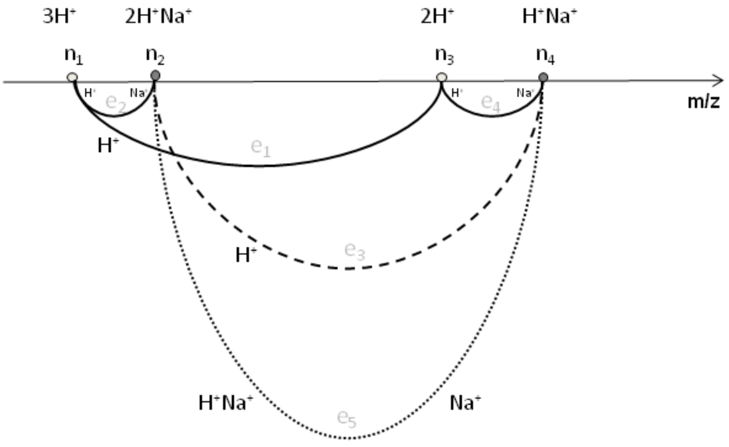


Figure 4.5: Example for edge inference. e_5 is inferred by using the adducts induced by e_2 and e_4. (Note that for clarity only edges important for edge inference are shown, e.g., $e = \{n_1, n_4\}$ is missing.)

To reduce the number of false positives in highly complex maps, an additional filter which reduces the number of edges in the graph can be used. In case that $\mathrm{ch}(e_k, n_i) = \mathrm{ch}(e_k, n_j)$, we add an edge $e_k = \{n_i, n_j\}$ only if $sign(\mathrm{int}(n_i) - \mathrm{int}(n_j)) = sign(\mathrm{pr}(e_k, n_i) - \mathrm{pr}(e_k, n_j))$ where n_i and n_j are the nodes connected by edge e_k; $\mathrm{int}(n)$, $\mathrm{pr}(e, n)$ and $\mathrm{ch}(e, n)$ are the intensity, probability, and charge of node n induced by edge e. In other words, we enforce that features with lower probability also have a lower abundance. We only enable this constraint for equal charge states since it is very hard to predict ionization behavior across multiple charges.

As table L only contains non-redundant adduct transitions, it is sometimes necessary to infer sibling edges that contain explicit redundant adducts from already existing edges. An example is given in Figure 4.5. As edge e_3 induces purely protonated nodes n_2, n_4, it is in conflict with edges e_2 and e_4, each inducing a sodium adduct at n_2, n_4 respectively. To enable the final solution to contain a fully connected subgraph $\{n_1, n_2, n_3, n_4\}$, another edge e_5 needs to be created. These inferred edges are created between any two nodes n_i, n_j for any pair of edges e_k, e_l with either $e_k = \{n_i, n_m\}$ or $e_k = \{n_o, n_i\}$ and $e_l = \{n_j, n_p\}$ or $e_l = \{n_q, n_j\}$, using the adducts induced for n_i, n_j by e_k and e_l.

The graph construction algorithm can be summarized in pseudo code as follows, assuming the input F is sorted by RT:

```
for $f_i$ in $F[start : end]$ do
  for $f_j$ in $F[i+1 : index(rt(f_i) + delta_{RT})]$ do
    for $(q_1, q_2)$ in Q $\times$Q do
      adduct_candidates = massDeltaLookup($f_i \cdot q_1, f_j \cdot q_2, mz\_tol$)
      for $ac_i$ in adduct_candidates do
        if (not intensityFilter($f_i, f_j, ac_i$)) then
          continue
        end if
        insertEdge($E, ac_i, f_i, f_j$)
      end for
    end for
  end for
end for
edgeInference($E$)
```

The for-loop enumerating all feature and charge combinations is optional. It is only used when the algorithm is in "discovery" mode, i.e., when searching for edges without relying on the annotated charge of the feature but instead enumerating all possible values.

4.2.2 Constructing the ILP

Having constructed the adduct transition graph for our problem, it is straightforward to define the corresponding ILP. During this phase, all edges sharing one or more nodes are checked for consistency, i.e., whether any pair of edges induces an inconsistent adduct annotation for the shared feature. Consistency requires

1. identical charge,

2. identical adduct composition.

An example for two inconsistent edges can be found in Figure 4.3. Feature f_2 is assigned adducts K^+4H^+ by the left edge whereas the right edge induces $5H^+$, both of which cannot be true simultaneously. We introduce x_i to indicate the presence/absence of edge e_i from the solution and c_i as the score of edge e_i.

The ILP is defined as

$$\begin{aligned} \max \quad & \boldsymbol{c}'\boldsymbol{x} \\ \text{s.t.} \quad & x_i + x_j \leq 1 \\ & x_i, x_j \in \{0,1\} \text{ for all pairs of inconsistent edges.} \end{aligned} \tag{4.1}$$

A more advanced formulation can be established when we model the configurations (charge, adducts) of each feature i as a set y_i with size m_i and force one configuration to be chosen. The configurations are induced by the adjacent edges of each feature. We thus arrive at the following formulation:

$$\begin{aligned}
\max \quad & \sum_{k=1}^{n} c_k x_k & & \\
\text{s.t.} \quad & \sum_{j=1}^{m_i} y_{ij} = 1 & & , \forall i \\
& x_k \leq y_{ij} & & , x_k \in E(y_{ij}), \forall i, j \\
& x_k, y_i \in \{0, 1\}, & &
\end{aligned} \tag{4.2}$$

where $E(y_{ij})$ is the set of edges inducing configuration j of feature i. In other words, an edge is chosen if and only if the two feature configurations induced by this edge are active. We use Equation 4.2 to solve the decharging problem, as in practice it is much faster to solve than Equation 4.1, especially for larger problem sizes. The reason for this is most likely due to the large number of pairwise constraints induced by Equation 4.1.

The ILP's output is a set of active edges. Thus, finding all connected components will automatically cluster nodes (features) into groups representing charge ladders with adducts and/or labeled pairs.

Due to the problem structure, the ILP solver employed (COIN-MP) achieves runtime improvements of ≈40% just by ordering the pairs as connected components when constructing the ILP columns even though a clique heuristic is active within the solver. As a further optimization we split the ILP into child ILPs by determining connected components in the graph and only feeding the edges for one (or a few) connected components at a time. The overall result is not affected by the concurrent amount of connected components fed to the ILP solver, but runtime can improve about five-fold.

Also, the resulting child ILP's lend themselves to parallelization in a very straightforward manner. Thus, the algorithm has been parallelized by OpenMP [131] using dynamic thread allocation. The resulting runtime improvements can vary considerably, depending on the hardest sub-ILP. Usually improvements of another 40-60% can be observed with four threads.

4.2.3 Post-Processing

During post-processing clusters can be discarded using a filter which reduces spurious hits. The "backbone" filter will only allow clusters which have at least one feature whose charge can be explained by protons only, i.e., that is part of the backbone of a charge ladder. Otherwise wrong or very unlikely clusters might be found, e.g., $([3K]^{3+}, [5Na]^{5+})$. Without the backbone filter (especially in complex maps) these spurious hits are common when all possible feature charges are enumerated by our algorithm.

4.3 Results

The algorithm was applied to several real data sets. On all data sets analyzed here, the running times for our algorithm were below five seconds (2.26 GHz Core2Duo), and memory requirements did not exceed 500 Mb. Time and memory requirements can increase, however, if many adduct types are allowed and the feature finder charge is not fixed.

We will now show some practical cases where decharging can help increase data quality. We compare our approach to a commercially available tool (Xtract) and to a pair-finding algorithm implemented in OpenMS. Comparison with other packages is difficult since they are partially specialized for certain labeling methods like SILAC or ICAT. If not indicated otherwise, we used the OpenMS PeakPicker for centroiding raw data and the OpenMS FeatureFinder for generating feature maps.

4.3.1 Increasing Mass Precision

We applied our decharging algorithm to one of the SPC data sets (Mix1, LTQ-FT, 20060502data08) [21]. This data set stems from a tryptic digest of 18 proteins measured in an LTQ-FT mass spectrometer and is available at http://regis-web.systemsbiology.net/Publicdatasets/. The interesting region of 500-4 000 s and 400-1 400 Th was excised and only every second scan was retained from the MS^1 data, as only they contained the FT scans. The OpenMS PeakPicker and FeatureFinder were applied to the raw data, resulting in 1 064 features. Subsequent internal calibration using high-confidence MS^2 identifications was applied to enable the calculation of a standard deviation between monoisotopic feature position in m/z and MS^2 identifications. Decharging was applied to find clusters of corresponding features stemming from the same peptide with different adducts. We found evidence for adducts (see Figure 4.4) and thus allowed H^+, Na^+, K^+, and $\text{NH}_4{}^+$. With the data set being a high-resolution measurement, the charges assigned by the FeatureFinder are mostly correct, nevertheless misassignments did occur (especially when the isotope pattern deviated strongly from the averagine model). Hence, we allowed the decharging algorithm to alter the FeatureFinder charge. The adduct transition graph had 1 064 nodes, 344 edges, inducing 167 constraints. About 35% of all features (371) were grouped into 155 clusters during decharging, and their monoisotopic m/z position was corrected using the average mass predicted by all members of the cluster. For all other features (693), no partner was identified. For 20 clusters, an MS^2 identification was available. This allowed to calculate the mass deviation between the predicted MS^2 mass and the feature mass. The standard deviation between the features' monoisotopic m/z and the theoretical m/z position predicted by MS^2 identifications prior to decharging was 1.044 ppm. In contrast, it was significantly reduced to 0.527 ppm after decharging due to the fact that feature masses are averaged over all members of a cluster by our algorithm. The increase in mass precision by clustering obviously only applies to features which are members of a cluster and can thus benefit from decharging. Furthermore, we examined those features whose charge (as assigned by the feature finder) was altered in the ILP solution. In total, the charge of eight features was changed by our algorithm, and, except for one, these reassignments were found to be correct by manual verification of the raw data.

4.3.2 Finding Pairs in Labeled Data

We applied our algorithm to a centroided data set of MHC peptides [132] which contains 4 117 scans and 3 083 features. Nicotinic acid labeling was used to tag two samples with either a light or heavy label (in which four hydrogen atoms were replaced by deuterium) prior to mixing and LC-MS analysis. Our algorithm generally supports any kind of labeling as long as the mass difference can be expressed as an empirical formula (see below).

As the charge estimation using the OpenMS FeatureFinder was very reliable for this data set, feature charges were not altered. The set of possible adducts was set to $^+$H and D4-4H, the former being simply protonation, the latter being an uncharged adduct describing the net mass gain of 4 Da for the heavy analog due to deuterium exchange. We allowed up to two uncharged adducts for the computation of L. The adduct transition graph had 3 083 nodes, 653 edges, inducing 349 constraints.

To compare our results we tested the labeled pair finder of OpenMS. It allows the user to supply an arbitrary list of allowed masses and RT shifts. We found 293 pairs using this standard

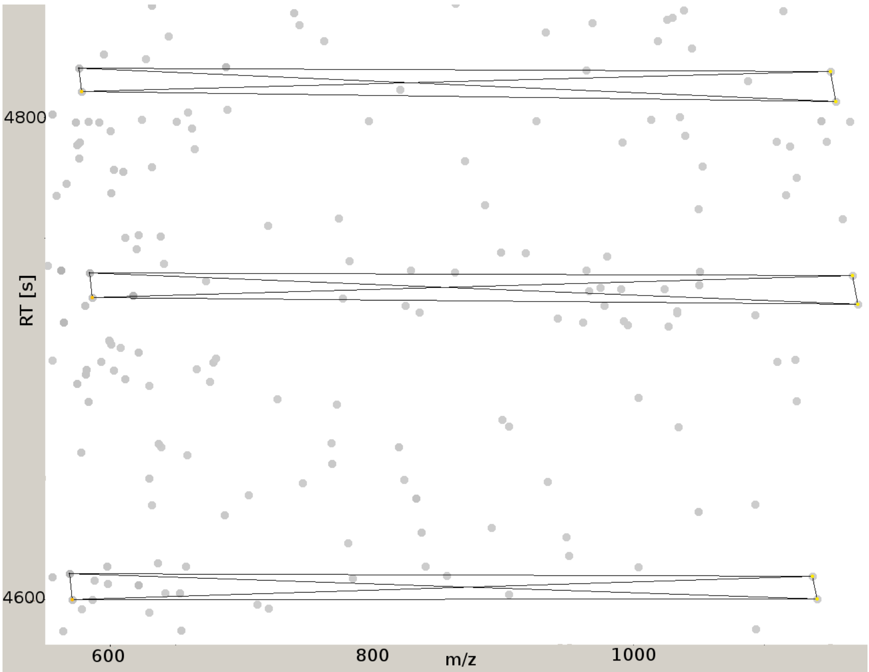


Figure 4.6: Example of charge ladders spanning two charge states (including light and heavy partners). Edges connect all features of the cluster as found during graph construction.

approach. Using the decharging algorithm we found 307 pairs, 16 of which have a partner pair in a different charge state (see Figure 4.6 for an example). These 16 pairs can be condensed into 8, because they represent the same peptide. Moreover, it allows to compute the average of two intensity ratios (see Figure 4.7).

Sixty-four clusters of size three were also found, 11 of which contained features of the same charge state, thus indicating a potential conflict in uniquely identifying the light and heavy pair: when ordered by m/z, it is possible that either feature 1 and 2 or feature 2 and 3 represent the light and heavy peptide. Other pair-finding algorithms will most likely just pick one greedily. Even the common precaution in standard pair finding algorithms requiring that any third feature must lie x Da further away from a pair would not be beneficial here as x would need to be larger than the pair mass difference in order to avoid ambiguous pairing. Choosing x that way would result in many pairs remaining undiscovered in the data set. Our approach will detect these ambiguities and allows to discard/mark those clusters. Another constellation for clusters of size three occurred 53 times: one light/heavy pair is identified, and additionally, a third feature (either light or heavy) of a different charge. The missing fourth feature was either not discovered during feature finding or is simply not detected by the instrument. An example of the former case is given in Figure 4.8. Without reference to another charge pair or MS^2 identification it is difficult to infer which of the two partners is present or if the identified feature even has a partner. Manual inspection of the data set suggests that about 60% of the 53 clusters indeed have a fourth feature, which was simply not detected by feature finding. Reiterating the feature finding step using the 53 seeded positions suggested by our algorithm yielded 29 new features (10% increase in pair count).

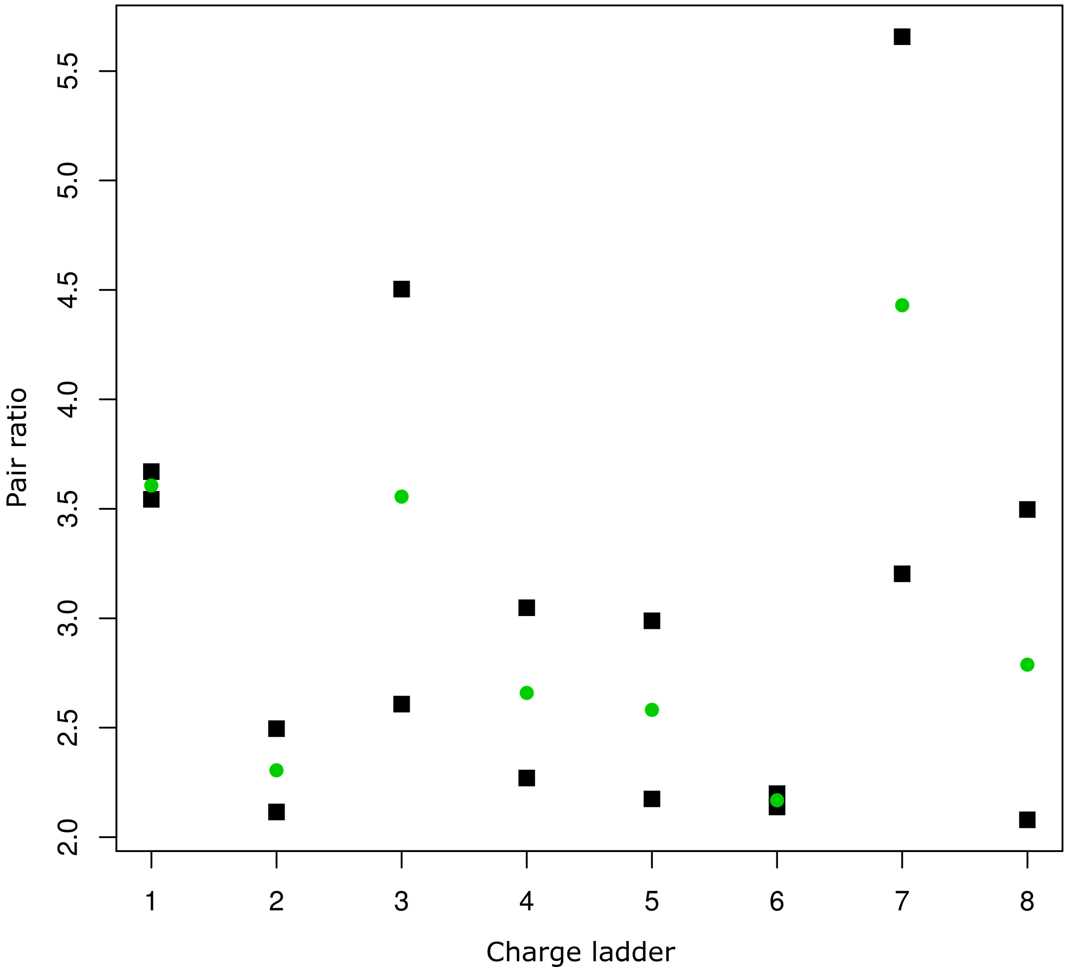


Figure 4.7: Intensity ratios (light vs. heavy feature) from nicNHS pairs of different charge states. Ratios are depicted as squares, averages as circles. Some pairs (e.g., #6) show very similar intensity ratios in different charge states whereas other charge ladders (e.g., #7) show a two-fold difference. These differences can aid in determining the confidence in the observed ratios.

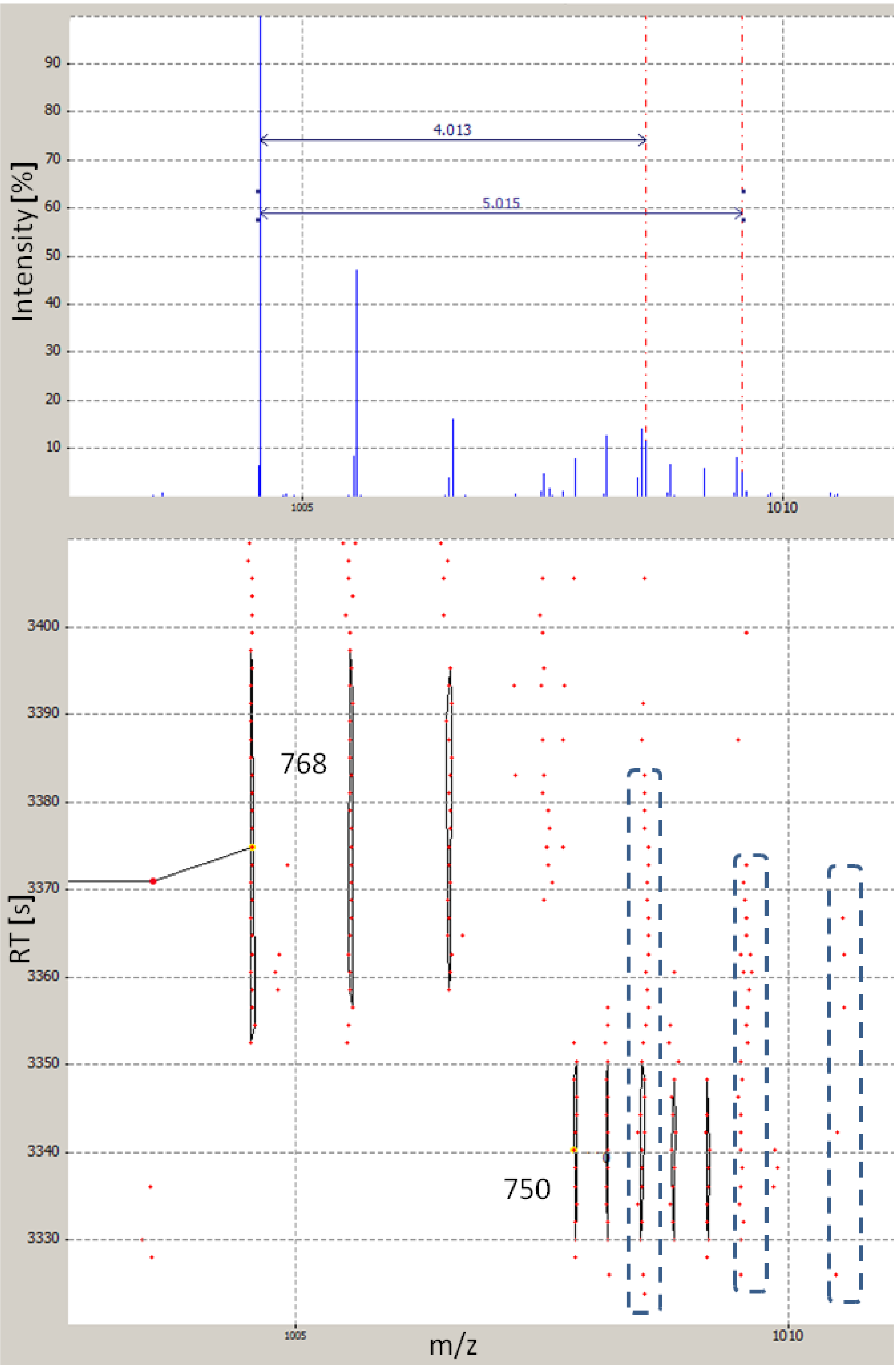


Figure 4.8: By evidence from a triple (2x q_2, 1x q_1) we can infer the presence of the heavy partner of the q_1 feature #768. The top section shows the projection of the m/z dimension. One can clearly see signals at 4 and 5 Da from the monoisotopic mass trace of feature #768. The missing feature's mass traces are indicated by dashed boxes in the map view (lower section). The reason for not identifying this feature is probably the presence of feature #750.

4.3.3 Calculating Intact Protein Masses

Hemoglobin

We analyzed a hemoglobin HPLC/ESI-MS raw data set consisting of ten scans containing HBA1 and HBB measured on an LTQ Orbitrap XL mass spectrometer with a resolution of 100 000. We compared the results of our algorithm with the Xtract module of Thermo's Proteome Discoverer 1.0. This module operates scanwise allows decharging at the raw data level. The maximum charge was set to 30, and we enabled the reporting of monoisotopic masses only. Xtract finished after 237 s of CPU time (2.26 GHz Core2Duo). Note that Xtract reports the monoisotopic protein mass as singly charged.

We used Hardkloer (v1.22) [133] to identify features scanwise and the postprocessing tool Kroenik (v1.3) to summarize features occurring in multiple scans. Minimum and maximum charge were set to 4 and 30, union and intersection mode were enabled, and S/N was set to 1. Decharging was set to consider sodium and potassium adducts and to correct for monoisotopic shifts of up to one position to the left or right. Alteration of charge values was disabled. The adduct transition graph had 104 nodes, 315 edges, inducing 2 590 constraints.

CPU time from raw data to features took 15 s, subsequent decharging one second (2.26 GHz Core2Duo). Our algorithm found 68 distinct masses (clusters). The two largest-sized clusters represent the hemoglobin subunits – cluster A for HBA1 (size 14 ranging from charge 8-18 with 11 proton-only features, 2 potassium and 1 sodium adducted features, average measured monoisotopic mass was 15 116.92939 Da, molecular mass calculated from the sequence was 15 116.88510 Da), cluster B for HBB (size 14 ranging from charge 9-17 with 10 proton-only features, charge 11 occurring split into two proton-only features with different RT, 2 sodium and potassium adducted features each, average measured monoisotopic mass was 15 857.29186 Da, molecular mass calculated from the sequence was 15 857.24969 Da).

As Xtract reports several masses (one per scan) for each hemoglobin subunit, we extracted the relevant regions to obtain an overall of ten molecular mass values for each subunit. By averaging these ten mass estimates we obtained masses 15 116.95 and 15 857.31 Da for HBA1 and HBB, respectively. Figure 4.9 and 4.10 show the relative mass deviations for both methods, the horizontal lines indicating the relative mass deviation from the theoretical mass for our approach and Xtract. Note that our approach is closer to the predicted theoretical mass (at 0 ppm) for both subunits. However, the instrument seems not to be optimally calibrated as both methods' standard deviation (0.4987 and 0.3858 ppm for our method with 2 and 1 outliers removed, 0.3510 and 0.8808 ppm for Xtract with 1 and 0 outliers removed) is lower than the gap to the theoretical mass of the protein. Outliers were removed using z-scores and a p-value threshold of 0.95. Hence, our decharging algorithm can also be utilized to recalibrate the mass spectrometer with signals of multiply charged ion species. Our approach can additionally group all sodium and potassium peaks into the main cluster (if desired by the user), which further disentangles the results. Furthermore, our approach also works with single spectra, allowing us to estimate the mass error from multiple charges – an information not provided by Xtract.

Wheat Extract Protein Mass

Mohr et al. [134] analyzed a wheat extract containing intact protein masses using an LTQ Orbitrap XL mass spectrometer. In order to obtain protein masses, the data was analyzed using the Xtract tool supplied with the instrument software and ProMass [134]. In brief, multiple scans of interesting regions as determined by manual data inspection where summed to increase the

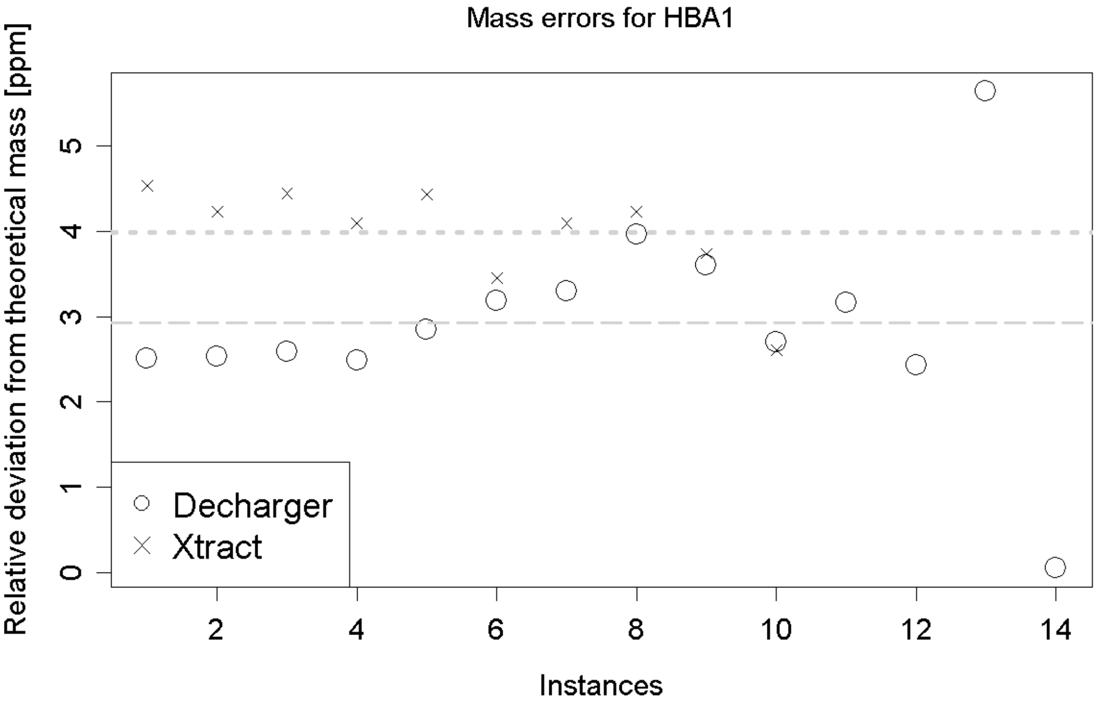


Figure 4.9: Deviation of observed masses for HBA1 (in ppm). Circles represent charged features clustered into A, crosses are the mass estimate errors by Xtract from the ten scans. Dashed (our) and dotted (Xtract) lines are the mean values.

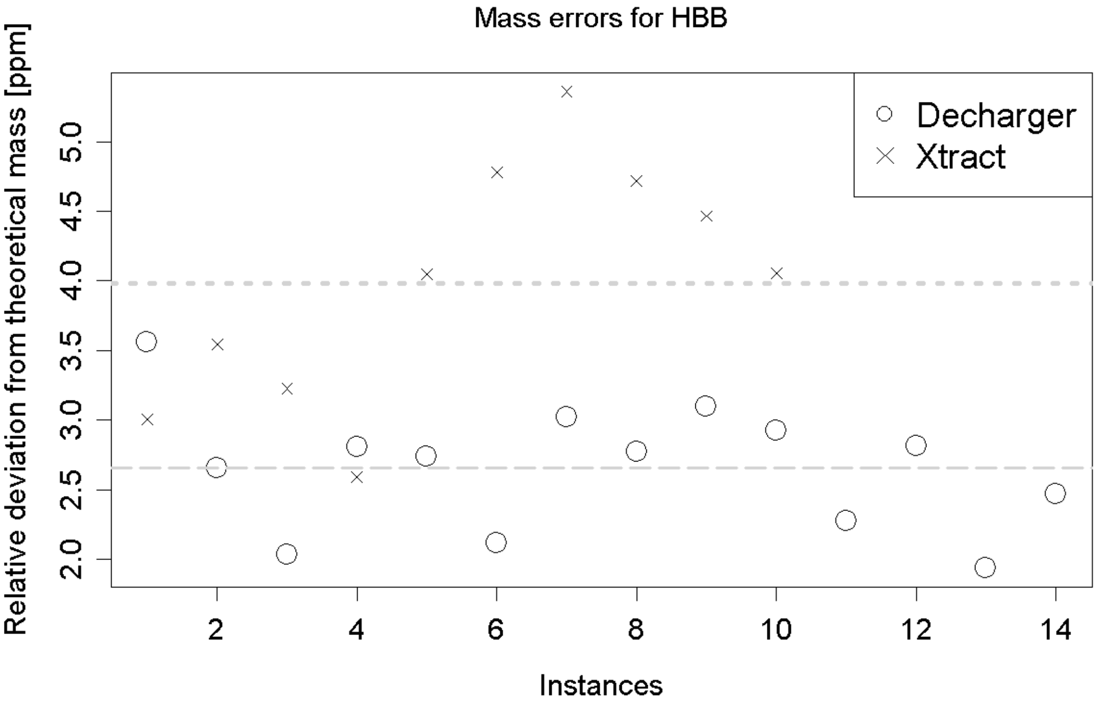


Figure 4.10: Deviation of observed masses for HBB (in ppm). Circles represent charged features clustered into B, crosses are the mass estimate errors by Xtract from the ten scans. Dashed (our) and dotted (Xtract) lines are the mean values.

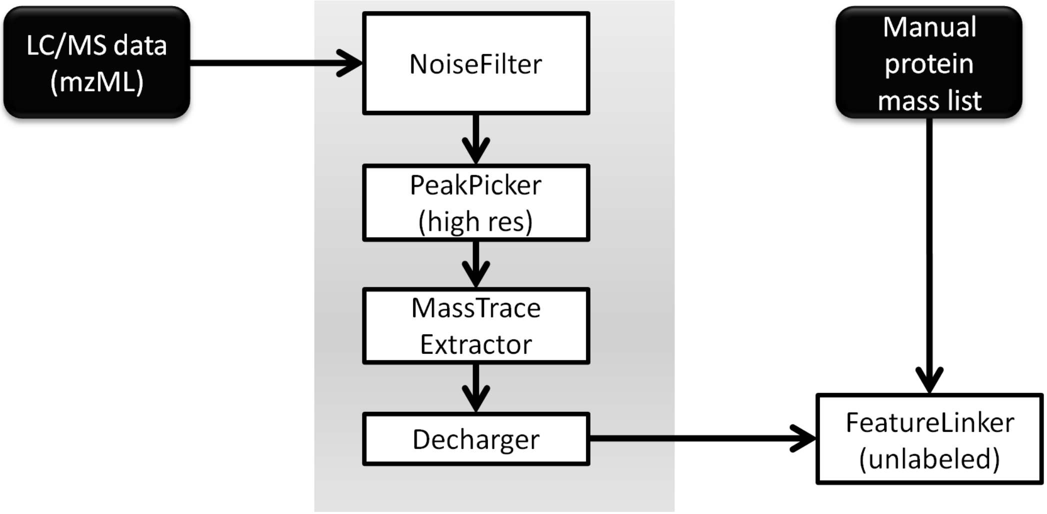


Figure 4.11: Illustration of decharging workflow for highly charged protein data with unresolvable isotope patterns. The resulting protein list was compared to the manually annotated list by matching protein masses and retention times.

signal-to-noise ratio and submitted to Xtract and ProMass. According to the authors, ProMass was used for summed spectra where no isotopically resolved signals were available (thus inferring protein mass by charge ladders only). Xtract uses the distance between isotopically resolved peaks to infer charge. Subsequently an averagine model is fitted to the isotope pattern to infer the average mass. Decharged spectra where then manually annotated to obtain protein masses. The supplemental material if [134] contains a table listing 53 protein masses.

To show the automation and high-throughput capabilities of our approach, we designed a robust and widely applicable pipeline to automatically extract protein masses from the whole raw LC-MS map. As the data set contains large proteins which cannot be isotopically resolved, we replace the feature finding step with data smoothing, peak picking and subsequent mass trace finding, using the respective TOPP tools (see Figure 4.11). This yields features that represent the protein in a certain (but unknown) charge state and its average mass.

Processing time for the complete pipeline, starting from raw data (≈4.7 GiB mzML file) to a list of protein masses, was about 20 min on an Intel Xeon server using 8 GB of RAM at most. Decharging itself takes only a few seconds.

Our algorithm yields 126 protein masses (after removing spurious hits with less than three charge variants). We obtain a charge distribution as shown in Figure 4.12 with charges ranging from 11 to 59.

Figure 4.13 shows the most dense section of the smoothed and peak-picked raw data, superimposed with the charge-annotated features. Features in red were assigned to a protein mass and thus have a charge. Features in blue could not be annotated since no charge ladder was identified.

To compare the results, we matched the manually created list of 53 protein masses with associated retention times from Mohr et al. [134] using the FeatureLinkerUnlabeled TOPP tool, allowing for mass tolerances up to 5 Da and RT shifts of 60 s. We were able to recover 30 proteins (57%) of the manually curated list. 23 protein masses were not found which were manually annotated. Another 96 proteins were uniquely found by our approach.

In order to explain the missing matches, we manually compared the two protein mass lists.

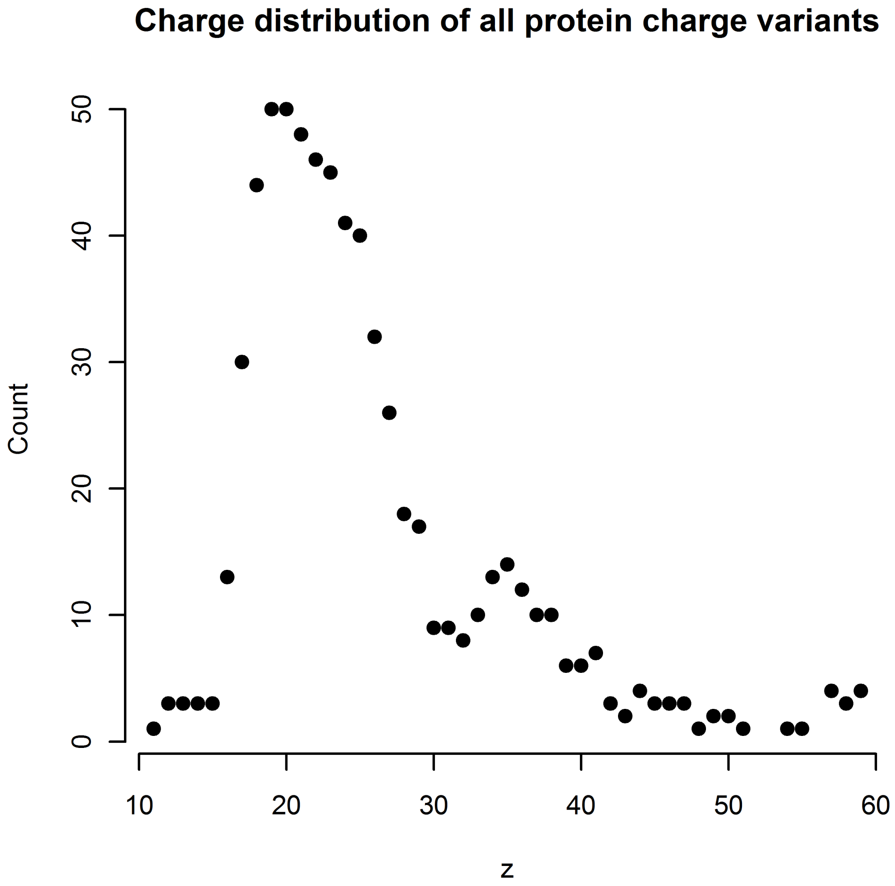


Figure 4.12: Charge distribution for all protein charge variants of the gliadin extract sample.

For the protein with manually annotated mass 31 543.54 Da at 38.03-38.63 min, we found a close match in our list of 31 534.12 Da at 38.45 min. By inspecting the raw data (see Suppl. Mat. of [134], p.38) we found the true mass to be 31 534.54 Da, which is very close to our result. The mismatch was caused by a typing error in the manual list of [134].

Two masses in the manually curated list which were apparently computed using ProMass (instead of Xtract) are found at 54 863 Da, 3.88-4.02 min (see p.13, Suppl. Mat.) and 54 977 Da, 4.51-4.68 min (see p.18, Suppl. Mat.). We find close matches at 54 837 Da, 3.96 min and 54 950.55 Da, 4.58 min respectively. In both cases, a mass shift of about 25 Da can be observed, with the protein annotated by ProMass being heavier. Since we are already computing average masses, the difference between monoisotopic and average mass cannot serve as an explanation. When looking at the corresponding spectra computed by Xtract (compare p. 11 versus p. 13 for the first protein, and p. 15 versus p. 16 for the second), we find the Xtract spectra to agree very well with our estimated mass. Thus for ProMass spectra, there seems to be a mass calibration procedure in place, making the protein masses incomparable. This not only prevents us from finding a match in both cases but also leads to an inconsistent mass list in the Supplemental Material of [134], since some masses are reported using ProMass and others using Xtract. Since the sample is of high complexity and the identity of most proteins remains unknown, we cannot assess if the mass correction of ProMass is meaningful.

To explain the excess of protein masses identified by our algorithm, we tried to find the masses in the deconvoluted spectra supplied in the Supplemental Material of [134]. Multiple

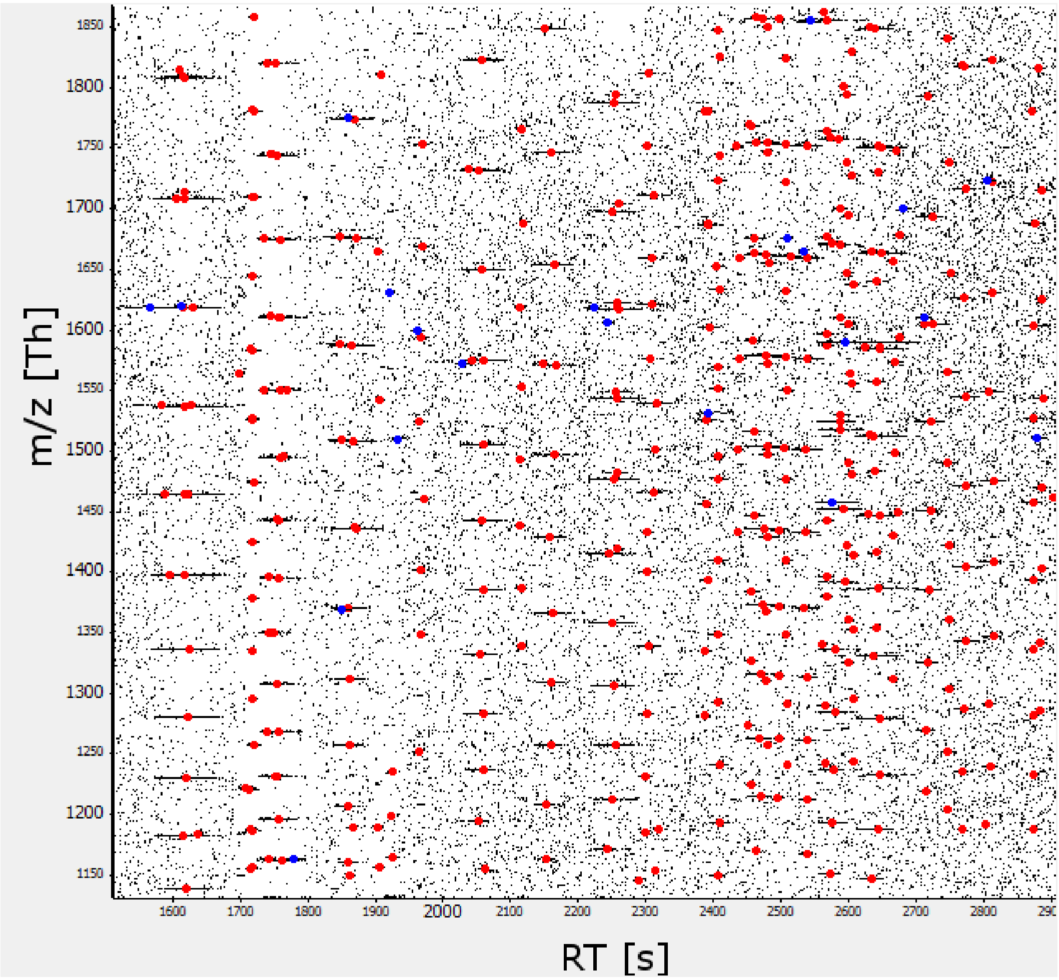


Figure 4.13: Dense section of the picked raw data, superimposed with charge-annotated features (red) and unassigned features (blue).

matches were found, which can be identified from these spectra, but were not mentioned in the final protein mass table; e.g., on p. 48, a protein of mass 31 112.48 Da was identified, which our algorithm also found at 31 112.12 Da and RT 2 606.13 s. On the same slide covering 43.18-44.15 min we found three other masses not mentioned in the table, but matching very well to entries in our list, namely, mass 30 498.02 Da at 43.27 min (versus 30 499.16 Da), 30 615.08 Da at 43.14 min (versus 30 613.26 Da) and 31 296.43 Da at 43.33 min (versus 31 296.59 Da). Similarly, this analysis could be done for other retention time windows (data not shown).

4.3.4 Benchmarking Using Simulated Data

In silico data lends itself very well to study algorithm performance under different conditions. We use a set of proteins with varying complexity, modeling data sets ranging from simple protein mixtures to complex samples, and apply the decharging algorithm. The feature list which serves as input for our algorithm depends on a few preprocessing steps, which – if performed carelessly – will prevent our algorithm from fully reconstructing the complete protein list (false negatives) or even lead to wrong results (false positives). Thus, we vary the degree of missing data and study its effect on precision and recall. This data reduction step can be done in multiple ways. We argue that the most useful approach is random sampling from all features. In real data sets, some charge ladders are affected more by undersampling than others, especially when peak

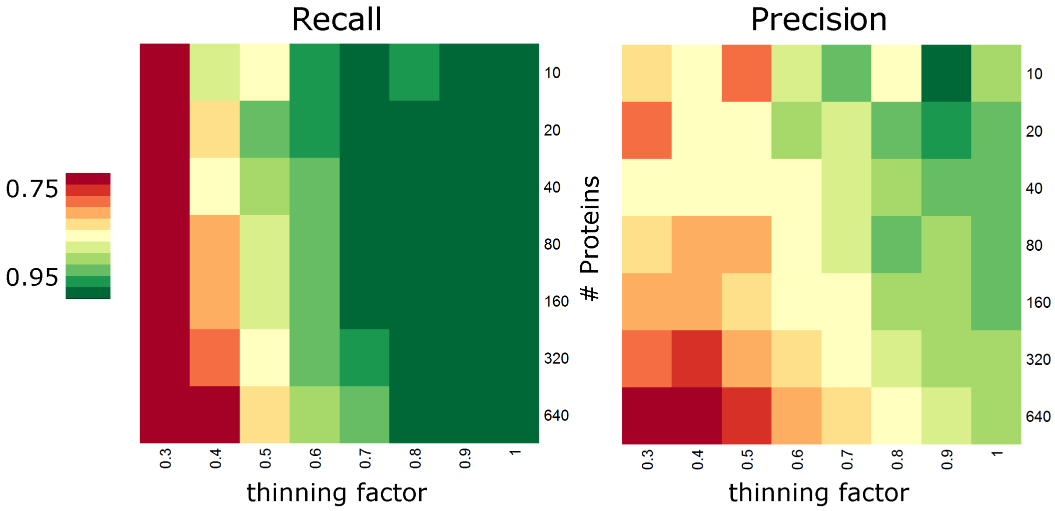


Figure 4.14: Heapmap displaying sample complexity and missing data rate for recall (left) and precision (right) for simulated protein charge ladders.

intensity is low. Thus, real data is always a mixture of varying degrees of missing data for each protein. However, this mixture effect strongly depends on the data set and is hard to quantify. The result is much more interpretable if we create a homogeneous data set.

We model a set of sample complexities, ranging from only a few proteins (10 proteins) to a complex mixture (640 proteins) randomly sampled from the human proteome for proteins with a weight up to 100 000 Da.

We report recall (see Equation 3.7) and precision (see Equation 3.8) as we only have true positives, false positives and false negatives available. The number of true negatives is infinite.

We count true positives as entities which have the correct retention time and mass as well as internally fully correct charge assignments. The charge ladder is not required to be complete. False positives are proteins not found in the simulated ground truth (by a small RT and m/z delta) but reported by the decharging algorithm. False negatives are proteins which were expected but not reported by our algorithm (e.g., due to missing data).

Figure 4.14 summarizes the performance for varying sample complexity and missing data. The data was averaged over multiple runs to smooth out outliers, especially in low complexity mixtures. Not surprisingly, increasing complexity and the amount of missing data reduce the precision and recall of our algorithm. Note that neither recall nor precision dropped below 75%. Recall shows a critical point at a thinning factor of about 0.4 where suddenly many protein masses cannot be reconstructed. This happens when gaps in the charge ladders become too big such that the algorithm cannot bridge them any longer. The largest bridgeable gap in this experiment was two (which is the default), i.e., two intermediate charges are allowed to be missing. Recall is not significantly affected by the number of proteins. Precision is dependent on protein count and thinning factor simultaneously as precision improves when protein count declines and there is less missing data. Intuitively, thinning out data might split charge ladders into multiple sub ladders which cannot be bridged. As the scoring function favors easy explanations over more complicated ones, singletons from one charge ladder might be clustered with singletons from others when the required adduct annotation is simpler, e.g., a charge difference of one instead of five.

Overall, we find the algorithm to be robust up to and even beyond 50% missing data and capable of handling very complex protein mixtures.

4.4 Discussion

We demonstrated that decharging is useful for many applications in quantitative proteomics. The algorithm is not restricted to a specific instrument or resolution, and although it is intended for ESI data, it should also be applicable to MALDI data when multiply charged ions are observed (e.g., for whole protein measurements [30]). The algorithm was optimized by splitting the ILP into subproblems, which can be solved more efficiently by the solver, but also lend themselves to parallel processing. Both measures reduce running time significantly such that for a complex data set with adducts, the runtime is only a few minutes. Without adducts, the solution is usually obtained within seconds.

Decharging was able to improve mass precision on the SPC data set from 1.044 ppm to 0.527 ppm, as verified using MS^2 identifications. On labeled data, the additional information of charge ladders was used to resolve ambiguous pairs as well as to infer more pairs. In top down mass spectrometry, decharging was found to be superior to the Xtract software on the hemoglobin data set. The protein mass estimated by our software was closer to the theoretical mass for both subunits (HBA1 and HBB). Furthermore, in case of the gliadin data set, we showed that manual annotation is not only cumbersome and time-consuming, but also prone to misannotation. Many protein masses which are readily available from the data were not added to the manual protein list. This is also reflected by the fact that automatic annotation detected twice as many protein masses. A few protein masses were missed by our automatic annotation, usually because the mass traces were not detected due to irregular signal of the smoothed data, and were thus lost for charge estimation. Using in silico data, we showed that decharging is robust to missing features and can handle very complex samples. Recall was found be sensitive to missing features, whereas precision is affected by the number of proteins in the sample and the amount of missing features. Even in the worst case scenario with 70% missing features and the most complex mixture neither recall nor precision dropped below 75%.

A desirable extension of our approach is the estimation of scoring function parameters from the data. Also, missing features could be alleviated by a hypothesis-driven feature finding heuristic which searches for a strong signal (e.g., using signal-to-noise ratio) at putative feature positions to infer missing features or resolve ambiguous explanations.

Chapter 5

iTRAQ Biomarker Discovery

Synopsis: *iTRAQ data from multiple studies within a large project is evaluated. We devise a new approach to isotope correction, propose an experimental design, introduce new measures of iTRAQ data quality and confirm known properties of iTRAQ data.*

5.1 Introduction

5.1.1 Predict-IV Project

This chapter describes our biomarker discovery approach using multiple LC-MS iTRAQ experiments in the context of the *Predict-IV* (Predict-In Vitro) project, which aims at characterizing the dynamics and kinetics of cellular responses to toxic effects in vitro. The project is titled *Profiling the toxicity of new drugs: a non animal-based approach integrating toxicodynamics and biokinetics* and is funded by the European Union (EU) as part of the Seventh Framework Programme (FP7) and involves twenty international partners.

Undesired toxic effects of drugs are observed frequently in specific organs, namely liver and kidney, leading to early termination of compounds and their derivatives during drug development. Furthermore, the neuronal system is a frequent target of drug side effects. Thus, model systems for kidney, liver and the central nervous system (CNS) are used in this project. The project aims for an integrated test system/test strategy with specific and early markers to predict toxicity based on in vitro data before entering in vivo testing. The dynamics and kinetics of cellular responses to toxic effects in vitro, specifically for kidney, liver and CNS were examined. Improved dynamic and kinetic models for in vitro systems are expected to be delivered. The in vitro system model is used as a predictor for in vivo systems. This is a common and effective approach as in vitro systems based on human tissue might hold more predictive power for pharmaceutical safety evaluation than animal-based models. Also sample collection from animals is avoided (animal testing, ethic commission approvals). However, the approach has drawbacks as usually only a single (cancer) cell line is optimized, which might not be representative of a population of healthy cells. In addition the role of the microenvironment is neglected [51]. The full list of 29 final compounds for each subgroup of toxicity (hepatotoxic, nephrotoxic and neurotoxic) is currently restricted to project members only. Most compounds induce organ-specific toxicity. Cyclosporin A was chosen as a control compound, as it affects all three systems.

Our focus is on proteomics experimental design and identification of potential biomarkers specific for the induced toxic effect, including feature identification and differential display.

A biomarker can either be a single molecule (usually a protein) or a combination of several traits. Several classes of biomarkers can be discerned, i.e., diagnostic, prognostic (risk for a disease, no treatment) and predictive (in response to a treatment) biomarkers. The following criteria provide a measure of the performance of a biomarker. (1) high specificity for a given disease (few false positives); (2) high sensitivity (few false negatives); (3) ease of use; (4) standardization; and (5) clarity and readability of the results for the clinicians [12].

The aim of this project is the identification of putative biomarkers; thus, exhaustive quantitative protein data needs to be acquired in an untargeted approach. If putative biomarkers are identified across multiple toxic compounds and cell systems, a targeted follow-up study can be performed, which, however, is out of the scope of this project.

Multiple studies were conducted within the Predict-IV project. A *study* is defined as a set of experiments using exactly one cell system and one toxic compound (in different concentrations). Multiple biological replicates are usually generated, finally yielding a set of LC-MS data sets.

Parts of this chapter are being prepared for publication [36].

5.1.2 Labeling Preliminaries

In addition to the brief overview in Subsection 2.6.1, we focus on more specific details of iTRAQ at this point, in order to motivate the analysis procedures described in Section 5.2.

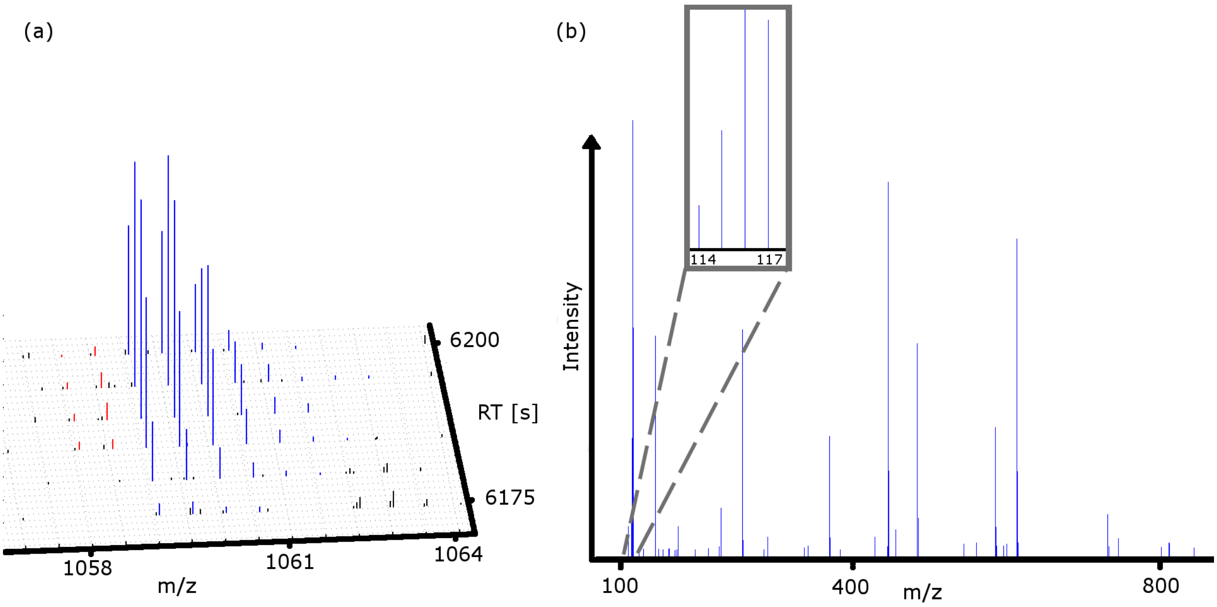


Figure 5.1: Exemplary iTRAQ signal at MS^1 and corresponding MS^2 level. a) peptide feature at the MS^1 level with isotopic envelope (blue) and pre-peaks (red) due to isotope impurity of the iTRAQ 4-plex reagent. b) corresponding MS^2 spectrum recorded in HCD mode. The inset shows the four iTRAQ reporter ions in the 114-117 Th range.

MS^1-based labeling techniques (such as SILAC) suffer from signal congestion (the amount of signal approximately doubles for two-channel labeling) and signal overlap, which lead to biased quantification results, usually towards higher intensities for heavy labeled peptides. The size of this effect depends on the mass distance between the labeled channels, peptide mass (as isotope distributions get wider – see Figure 5.1), and peptide intensity [135].

Isotope widening, i.e., additional isotopic peaks in front of the theoretical monoisotopic peak, in survey spectra cannot only be observed in MS^1 labeling techniques but also in MS^2 based methods like iTRAQ. These additional isotopic peaks are observable in MS^1 as isotope impurities lead to non-isobaric reagents, skewing the isotope distribution and causing isotope peaks below the theoretical monoisotopic peak of the peptide. Empirically, we observed that about 85% of all identified features have at least one pre-peak (data not shown). Not surprisingly, the intensity of a pre-peak is correlated with the intensity of the feature (correlation of ≈0.68 – data not shown).

In general, labeling techniques – independent of their MS level – tend to underestimate the true ratio. This effect has been described in the literature and is also true for iTRAQ in particular [83, 12]. Usually, this effect is attributed to background signal, especially in highly complex samples. See Subsection 5.3.5 for results on our iTRAQ data. The effect is more pronounced when the isolation window is widened. Thus, a tradeoff between signal intensity and signal contamination is required. Karp et al. [136] suggested to globally correct for ratio underestimation by a linear factor. Bantscheff et al. [83] showed that using wide isolation windows (2-5 Th) results in increased background signal from coeluting peptides, thus biasing the iTRAQ reporter ion counts – usually towards a ratio of one in a complex mixture when we assume that most peptides are not differentially regulated. To minimize this effect, the isolation window should be narrowed as much as possible, which comes at the cost of sensitivity as the reporter ion intensity will decrease due to fewer reporter fragments.

iTRAQ (and other MS^2 based techniques) usually require isotope correction of MS^2 reporter ions as the labeling reagents are not 100% isotopically clean. Using a correction matrix, isotope impurity is removed via inverse matrix multiplication or, in our case, via non-negative least squares (see Subsection 5.2.4). Isotope impurity also affects the MS^1 level; signal congestion is much less pronounced than in MS^1 multiplex techniques, though.

Multi-experiment designs have difficulties with missing values as a missing MS^2 scan cannot be reconstructed from the data; for purely MS^1-based methods, however, this is feasible. To minimize this problem, extensive fragmentation with long gradients or inclusion lists can be used. The latter require stable column conditions to allow for reproducible retention times.

5.1.3 Experimental Design

The following subsection closely follows and subsumes the work of Oberg and Vitek [137].

An experimental design describes the protocol that selects and allocates individuals to treatment/disease groups and arranges the experimental setup in space and time, e.g., allocation of iTRAQ channels, number of biological replicates, inference across multiple samples when $m > n$ ($m = |\text{samples}|, n = |\text{channels}|$) and has been researched extensively (e.g., [137]). Limited resources of laboratories for sample preparation, instrument acquisition time and material cost must be taken into consideration as well.

Three important aspects need to be considered, namely *replication*, *randomization* and *blocking*. Replication is the use of (biological) replicates, which allows to assess if an observation is purely by chance and can help to determine if a difference is significant. See Figure 5.2 for an example.

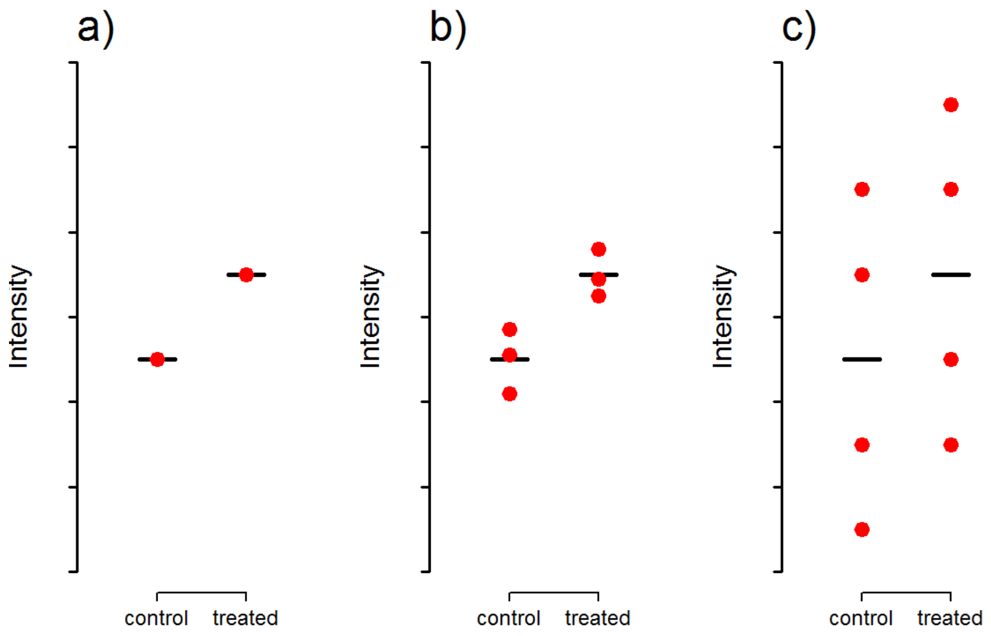


Figure 5.2: Importance of replication exemplified by experiments yielding identical mean expression values for each group. a) Experimental design without replication. It is impossible to determine if the difference is significant. b) Replication with small intra-group variation, indicating group difference. c) Replication with large intra-group variation, indicating no group difference. This figure is inspired by Fig. 2 in Oberg and Vitek [137].

What fold change is detectable is dependent on multiple factors, such as biological variation, number of replicates and experimental conditions. Thus, no general answer on the number of required biological replicates can be given. According to Noirel et al. [87], the fraction of published studies on iTRAQ suggests that using replicates is becoming more and more popular since replication is required for sound statistical conclusions.

Randomization helps avoid undesirable artifacts, e.g., instrument drift in time. This bias occurs if all samples from treatment control are prepared or measured prior to samples from the treatment group. Additional bias might occur if the instrument was cleaned or the LC column exchanged after one group was measured. A highly cited article written by Petricoin et al. [138], where experimental artifacts led to wrong scientific conclusions, prompted scientific concerns [139]. In general this is known as confounding effect: the observed difference between groups is (in parts) attributable to experimental conditions or other factors, rather than biological state. Complete randomization aims to remove this bias. However, especially for a small number of measurements, random sampling of the order of acquisition can also lead to unbalanced allocations, e.g., most control samples before treatment samples. In this case a manual allocation schema can be devised.

Blocking refers to measuring a specific subset of samples concurrently (or at least consecutively in label-free settings). Such a subset constitutes a block. The underlying assumption is that differences between samples can be assessed more effectively within blocks than between blocks. As one is usually interested in differences between control and treatment groups, these samples are blocked rather than blocking replicates of a single group. The size of the block is determined by the labeling technique, e.g., iTRAQ allows for block sizes up to four for 4-plex kits and eight for 8-plex kits. In a complete block design, every treatment appears in each block (i.e., one LC-MS run). If the number of channels is larger than the block size, only a subset can be allocated, leading to an incomplete block design. Alternatively, a reference design can be used, i.e., a common (pooled) reference is assigned to one channel in every block, which can be used as reference for intra-block normalization and comparison between blocks. A loop design is similar to a reference design, but groups are systematically shuffled through each block. Blocking and randomization can be combined into a block-randomized design where a member of each treatment group is part of one block and the channels are allocated randomly in labeled scenarios, or the order of acquisition within the block is randomized for label-free scenarios. A randomized complete block design is the most robust against run failures and requires the smallest number of runs, in contrast to a reference design where the reference channel is measured multiple times, or the loop design where a run failure destroys the link between the preceding and following block.
Of particular importance is a balanced design, i.e., every treatment should appear equally often, and within a block, each two treatments should appear about the same number of times [140, 137].

Pooling is an attractive strategy to reduce the number of samples and thus the number of runs, but has severe drawbacks. If all samples from one group are pooled, biological variability cannot be assessed, hence determining a statistical difference is impossible. Furthermore, pooling is particularly vulnerable to contaminated/outlier samples as the pool will be affected too. However, in a reference design a pool might be useful as it allows to collect enough sample by mixing and is a stable reference which contains all proteins from all groups, thus circumventing the problem of infinite ratios, i.e., if a protein is present in a treatment sample but not in the reference, the log ratio will not be computable and usually be lost in most software solutions.

5.1.4 Existing Software for iTRAQ Data Analysis

In order to be suitable for our purposes, an analysis tool needs to provide multiple features – most importantly and integrated analysis of multiple iTRAQ experiments which follow a designated experimental design. Furthermore, it must be able to deal with centroided high-resolution data from an LTQ Orbitrap XL instrument. Obtaining raw data from the instrument would be a major bottleneck for file transfer and storage as data generation and data analysis sites are located in different countries.

Existing software solutions usually provide methods for the analysis of a single experiment only. The most common approach for the analysis of iTRAQ data is to force a design that uses at most four (or in some cases up to eight) conditions and replicates, which avoids the need for combining multiple iTRAQ experiments [141], but usually limits the answers that can be given. The experimental design described above (Subsection 5.2.1) necessitates the combination of technical and, optionally, biological replicates. The most popular tools are described below.

i-Tracker [28] requires raw, i.e., uncentroided data since reporter ion intensity is computed using the trapezoid method and is limited to peptide level quantification; no protein level analysis is provided.

iQuantitator [142] is based on a Monte-Carlo-Markov-Chain approach and allows for almost arbitrary experimental designs. Accordingly, it is theoretically applicable to the Predict-IV design. The software is freely available for download and runs on Linux OS. Unfortunately, beyond a few experiments, runtime and memory requirements become unfeasible. For three experiments, the computation was stopped after 24 hours while using 12 GB of RAM. A minor problem is that isotope correction is assumed to be already complete.

Multi-Q [143] only supports Mascot search results, or PeptideProphet/ProteinProphet results, which complicates the use of multiple search engines. During our tests, search results from Mascot 2.2.4 and Mascot 2.3 yielded a fatal exception, probably because the tools' latest version from 2008 does not support the format. Also, the isotope correction matrix described in the paper (see Eq. 1 therein) seems to promote false diagonal values. Diagonal values should be computed as $1 - \sum_{p \in \{-2,-1,+1,+2\}} k_{ch,p}$ (see Subsection 5.2.4 for details) instead of just setting the diagonal to one. If isotope correction is implemented as described, the results will be incorrect.

ProteomeDiscoverer, the native solution provided by the instrument vendor for Thermo instruments, cannot combine runs and is only commercially available. Furthermore, it suffers from wrong isotope correction, as explained in Subsection 5.2.4.

An analysis of variance (ANOVA) approach described in [140] is computationally infeasible for a study of this size, and even the approach described in the paper requires the stage-wise fitting of parameters, which somewhat offsets the advantage of ANOVA to simultaneously fit all model parameters. ANOVA models cannot deal with data heteroscedasticity, i.e., noise is essentially multiplicative and varies with signal intensity. Additionally, the software is based on the commercial SAS package and not readily available.

The *isobar* package [144] accepts input as Mascot XML file, mzIdentML, tab-separated

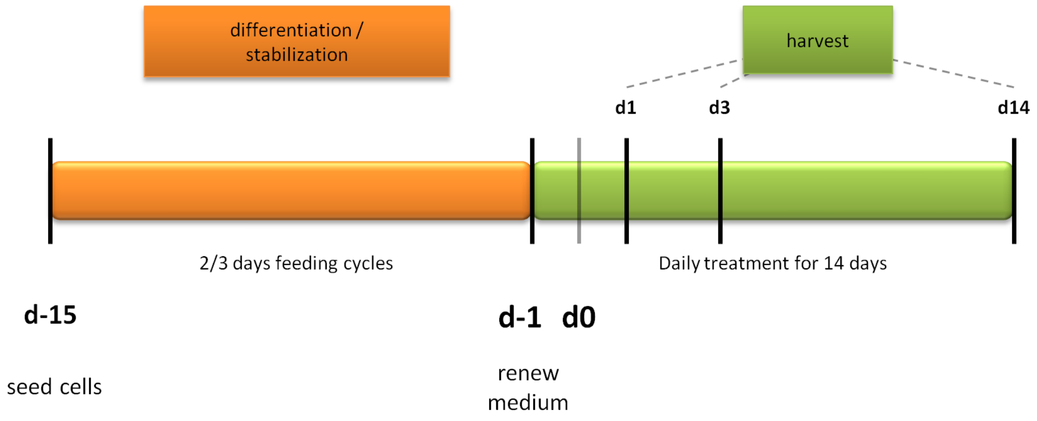


Figure 5.3: Differentiation and treatment scheme for cells from IMU studies. Treatment is either low dose, high dose or none (as control).

formats or directly via an R interface, permitting easy integration into an existing pipeline. isobar also features a flexible R interface, noise models to account for data heteroscedasticity, and statistically sound significance estimation for protein over- or underexpression, circumventing ad hoc thresholds for fold changes.

5.2 Methods

The amount of data expected during the course of the project necessitates the deployment of an efficient, adaptable and fast analysis pipeline. We thus decided to use the capabilities of OpenMS/TOPP/TOPPAS to build a custom iTRAQ analysis pipeline, covering identification and quantification, coupled with statistical analysis in R [145], an open source statistical software package.

Here, we will describe our experimental design and the analysis pipeline. The latter features multiple peptide identification engines, isotope correction via non-negative least squares, exclusion lists, and isobar for the estimation of protein fold changes and p-values to determine significant protein over- or underexpression.

5.2.1 Experimental Design for Predict-IV

Within the Predict-IV project it was decided that for all cell systems to be analyzed by proteomics, three dosage groups are collected. A control group, not treated with a toxic compound; a low dose group, receiving a mild dose; and a high dose group. The dose depends on the cell system and toxic compound and is determined separately for each. To observe a time effect, three harvesting points were agreed on, namely days one, three and fourteen after daily treatment with the respective toxin. See Figure 5.3 for an overview of cell differentiation and harvesting times for each dose and biological replicate cell sample.

Thus, in combination, this results in $3 \times 3 = 9$ groups, and therefore, a block size of nine would be desirable if all conditions were to be compared to every other. Unfortunately, iTRAQ does not offer blocks beyond four, or eight for the 8-plex kit. The latter is reported to yield similarly stable results as the 4-plex version with the added advantage of larger block size [146]. However, the probability of an experiment failure due to contaminated sample or experimental error increases when eight channels are used instead of four.

The number of biological replicates is bounded by laboratory capacity. For the studies at hand, three replicates are available. Enough material was delivered to allow for another three technical replicates.

As the number of groups (nine) exceeds the block size of four, we decided to block by concentration only, i.e., control, low dose, high dose. This allows to easily track changes in dose response but makes analysis of time effects hard as time is not blocked. Blocking of time would have resulted in one third of the experiments only containing control samples (i.e., day one, three and 14 samples from control in one block), which can at best answer how control samples evolve in time; this information is of limited interest, though. What is more, cells were grown for several weeks before being allocated to control or treatment groups, thus the 14 day harvesting period should not result in a change of protein expression in control samples. Without biological or technical replicates, this results in three LC-MS experiments, one for each harvest time point. Since three biological replicates are available per sample, they are distributed randomly across different blocks, resulting in nine LC-MS experiments. To increase the coverage of the proteome under investigation, three "technical" replicates were run for each iTRAQ mix. For details, see Subsection 5.2.3.

As one channel is still unallocated, we decided to add a pooled reference as a fallback option for later analysis of time series in addition to concentration series. The pool is created from the 27 real samples (three timepoints, three dosages, three biological replicates) to ensure that it contains all proteins and infinite ratios can be avoided. The pooled sample will always be assigned to channel 117 as the iTRAQ labeling procedure is another source of variation, and using the exact same sample avoids unwanted side effects and decreases laboratory overhead.

For channels 114 to 116, our design incorporates random channel allocation, i.e., random assignment of samples to iTRAQ channels and thus tags, as a safety measure. However, recent literature indicates that the tag effect is negligible and the direction of the effect is not reproducible [136, 147, 148]. Thus a tag effect needs not be accounted for during data analysis.

An illustration of the design can be found in Figure 5.4.

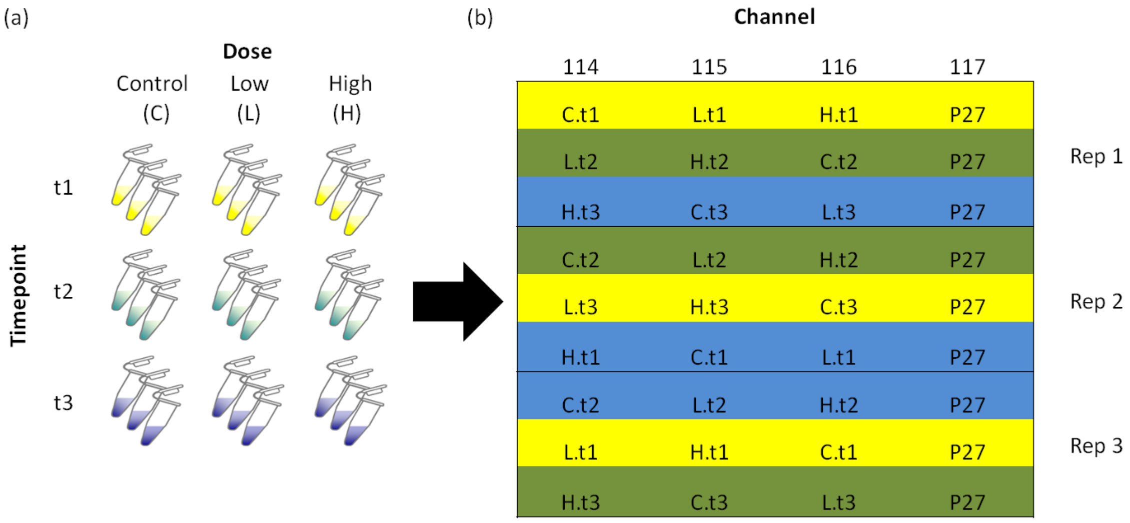
5.2.2 Cell Growth and Data Acquisition

All cell cultures (human and mouse) were grown under control (C), low dose (L), and/or high dose (H) conditions. One study (URO001) additionally used hypoxia (h) conditions (low oxygen) to simulate complications associated with pre-damaged and compromised tissue. Cell samples were analyzed by partners in Salzburg using an LTQ Orbitrap XL instrument operated in parallel mode. Survey MS scans were acquired in the Orbitrap between 450 and 2 000 m/z at a resolution of 60 000. CID spectra were acquired using the LTQ for identification, and corresponding HCD spectra with the same precursor were acquired in the Orbitrap for identification and quantification via iTRAQ. Samples were analyzed in triplicates in which peptides identified in a previous run were excluded from MS^2 triggering using the exclusion list option in the instrument software.

The raw data files were transferred from Salzburg to Berlin via file transfer protocol (FTP) and fed into our analysis pipeline as described in Section 5.2.3.

5.2.3 Analysis Pipeline

We will now describe the workflow used to convert the raw mzML files from the mass spectrometer to Excel sheets containing protein names, expression values for conditions, and significance estimates. An identification/quantification workflow was implemented in OpenMS/TOPPAS,



114	115	116	117	
C.t1	L.t1	H.t1	P27	
L.t2	H.t2	C.t2	P27	Rep 1
H.t3	C.t3	L.t3	P27	
C.t2	L.t2	H.t2	P27	
L.t3	H.t3	C.t3	P27	Rep 2
H.t1	C.t1	L.t1	P27	
C.t2	L.t2	H.t2	P27	
L.t1	H.t1	C.t1	P27	Rep 3
H.t3	C.t3	L.t3	P27	

Figure 5.4: Experimental design for the analysis of 27 samples using iTRAQ. a) For each analyzed compound, nine conditions comprising three dosages (control, low and high dosage) and three time points are compared. For each condition, three biological replicates are produced, resulting in 27 samples per compound. The proteomes of these samples are analyzed in nine 4-plex iTRAQ experiments as listed in (b). A pool of all 27 samples (P27) is used as reference and ensures high comparability between ratios across runs.

statistical analysis was done using the isobar package [144] and custom R code. In short, quantification is based on iTRAQ reporter ions extracted from HCD spectra. Reporter intensities are isotope-corrected using a non-negative least squares procedure. Peptide identification uses separate searches for CID and HCD spectra on multiple search engines (X!Tandem, OMSSA, Mascot) and subsequent ConsensusID [82] scoring with FDR filtering at 5%, using a decoy database approach. HCD and CID spectra are searched in separate runs to allow for optimized search parameters in terms of mass tolerances, for HCD spectra are acquired in the Orbitrap with high mass accuracy whereas CID spectra are acquired in the IonTrap at low accuracy. Split searching reduces false positive hits in HCD spectra due to more restrictive mass tolerances, thus increasing the number of identifications after FDR filtering (data not shown). However, too narrow precursor tolerances will lead to an increased FDR as the reliability of the scoring is reduced due to fewer candidates [149, 150].

We use UniProtKB [151] as database because it is actively maintained and provides stable identifiers [152]. Depending on the cell type, the corresponding species-specific database is used and concatenated with the common Repository of Adventitious Proteins (cRAP)[1] to ensure common contaminant peptides are identified, thereby reducing false positives. To control the false positive rate, we employ a decoy database which is created by reversing each sequence of the above database. The search is done with target and decoy database concatenated.

Channel normalization is based on median-of-pairs normalization. Protein ratios are computed from unique peptides after outlier elimination using a weighted average based on peptide noise level. Using only unique peptides, i.e., peptides matching only a single protein, will avoid skewed results from (homologous) proteins. Here, uniqueness is defined in the realm of the database, i.e., the whole known proteome of the respective species (human or mouse), in ad-

[1] http://www.thegpm.org/crap/index.html

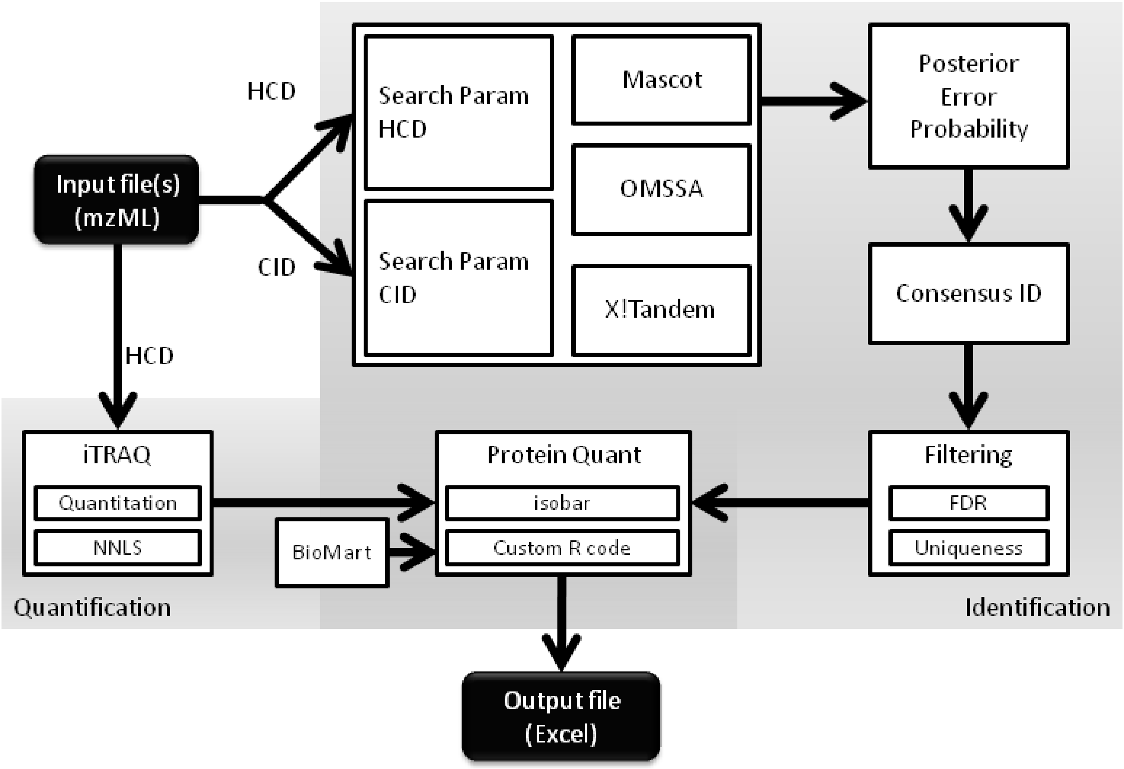


Figure 5.5: TOPPAS workflow to process raw mzML files as acquired by the instrument, resulting in Excel spreadsheets containing lists of identified and quantified proteins with significance values attached. Entrez gene IDs are queried via BioMart based on protein identifiers.

dition to common contaminants (see above). Significantly over-/underexpressed proteins are reported using a p-value threshold of 0.05.

Entrez gene IDs are queried via a BioMart R package [153] based on protein identifiers in order to ease downstream analysis for transcriptomics and proteomics data integration.

The workflow (see Figure 5.5) was modeled in TOPPAS and is executed with all 27 input files from one study.

Labeling Efficiency

MS^2 spectra without iTRAQ reporter ions cannot contribute to the final protein ratio computation. Those spectra are thus lost for quantification, even if they can be identified. Missing reporter ions can have multiple causes, e.g., the amount of sample was not sufficient, wrong precursor selection (e.g., noise peak), failed iTRAQ labeling, etc.

Therefore, labeling efficiency, i.e., the relative fraction of iTRAQ spectra in which a reporter ion signal is present in any channel, can be used as a criterion for data quality.

If labeling efficiency is to be determined for a specific channel ch_x, the presence of reporter ion signals should be evaluated after isotope correction, since isotope impurities can lead to the presence of noise signals in neighboring channels. When truly no peptide is present in channel ch_x, this noise must be accounted for, as ch_x would be counted as occupied otherwise. In comparison to a complete identification run (usually with multiple search engines), labeling efficiency can be computed very quickly because the algorithm solely needs to quantify the iTRAQ signals.

Exclusion List

With the current generation of instruments for LC-MS/MS employed in shotgun proteomics, not all coeluting peptides at a given retention time are selected for MS^2 during *data-dependent acquisition* (DDA) as spectral acquisition is too slow and not sensitive enough for very low abundant peptide species [154].

This leads to an undersampling problem that includes a random component as technical replicates are not completely identical in terms of chromatography conditions, amount of sample, and other confounding factors.

As shown in multiple publications, pure technical replicates lack reproducibility to varying degrees, depending on the platform and sample complexity [9] in terms of peptide and protein coverage as obtained from DDA in MS^2. The degree a protein list varies is usually smaller than the peptide list. Protein list overlap of 70-80% is common [57]. Nevertheless, technical replicates can be used to increase the protein coverage by relying on semi-random DDA sampling in high complex samples. However, many peptides, especially the highly abundant ones, are typically sampled in all replicates, thus wasting acquisition time. Also, FDR on the protein level will increase since false (random) identifications tend to be unique across runs whereas true identifications target the same protein in multiple runs [57].

For achieving a wider coverage of the proteome, a common method is to employ *exclusion lists*. Peptides identified in previous runs are not considered for acquisition in subsequent runs under the same technical conditions. As shown by Wang and Li [154], different exclusion strategies are possible. Adding retention time windows, for example, increases the number of identified proteins.

The Thermo software supports exclusion lists with some restrictions. Precursor positions of peptides identified in previous runs can be used to generate exclusion windows with a certain RT range. The m/z position is taken directly from the precursor. We call this single-charge exclusion.

However, using ESI, one peptide will have multiple charge variants (charge two and three are the most common), which are likely to be targeted in subsequent runs but will not contribute to identifying more peptides and proteins. See Figure 5.6 for a distribution of the number of charges a peptide was observed in, i.e., was identified by MS^2. Thus an overhead of up to $\approx$25% for targeting a charge variant of an already identified peptide can be expected when using single-charge exclusion. This data was derived from single LC-MS runs without exclusion lists. Thus, during ESI, no dynamic exclusion of charge variants is performed by the instrument. This is unfortunate as it leads to overhead, but using multi-charge exclusion (i.e., excluding other potential charge variants of the observed peptide), it is possible to avoid reacquisition in the next run. Exclusion of charge variants can unfortunately not be incorporated into the Thermo solution. We thus devised an OpenMS-based tool called InclusionExclusionListCreator, which allows to increase coverage by using exclusion lists accounting for charge variants.

For every technical replicate run, exclusion lists derived from all previous runs are computed and used during acquisition. We use the same identification pipeline that is used for the final analysis, including multiple search engine identification via ConsensusID [82] and FDR filtering. This strategy allows for a reacquisition of peptides with marginal scores as well as unidentified spectra.

The tool was tuned for usage with LTQ Orbitrap devices, which have certain restrictions on the exclusion data. For example, the exclusion windows are not allowed to overlap in m/z, and only a certain number of windows are allowed to overlap in RT (over the whole m/z range)

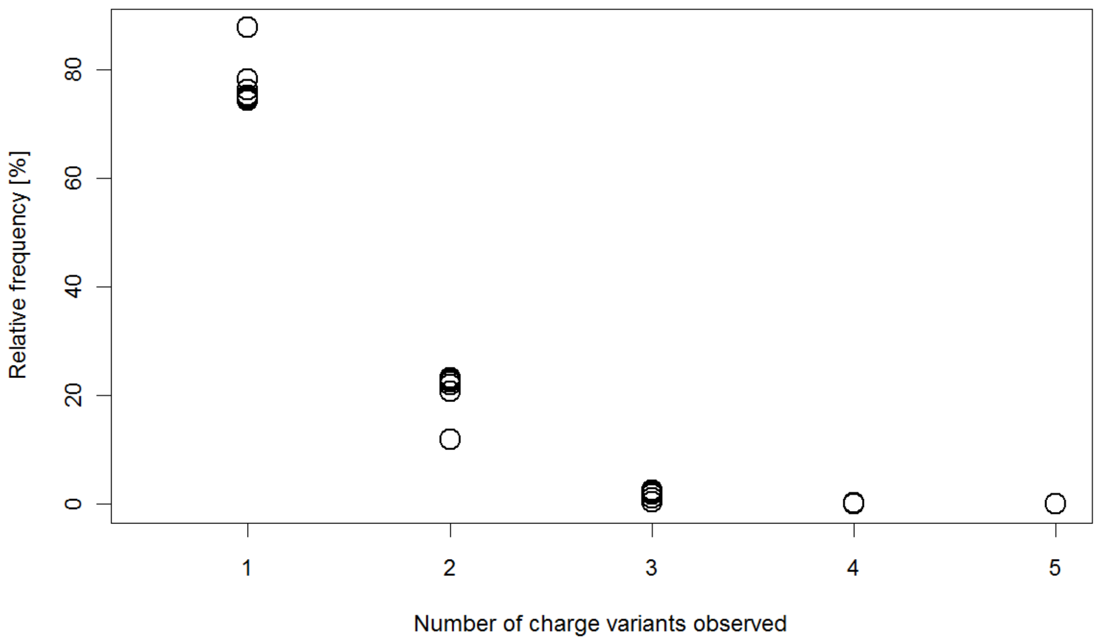


Figure 5.6: Distributions of the number of charges a peptide was identified in based on MS^2 data of IMU006. Multiple empirical distributions are shown, each one from a single LC-MS experiment where no exclusion list was used.

at any point in time. To resolve this issue, we use a hierachical clustering approach to merge overlapping windows after generating the exclusion list.

5.2.4 The ITRAQAnalyzer Tool

We implemented a new TOPP tool named ITRAQAnalyzer, which is capable of quantifying reporter ions in MS^2 spectra for 4-plex and 8-plex iTRAQ experiments.

As input, an mzML file must be provided. The output format is the OpenMS format consensusXML, which can be easily converted into a text file readable by statistics packages and/or Excel using the TextExporter TOPP tool.

Quantification is achieved by summing ion intensities at the expected reporter positions (e.g., 114.11123 m/z (monoisotopic) for channel 114) while allowing for a user definable mass delta, as reporter peak position depends on instrument precision and accuracy. Within this m/z interval, all peak intensities are summed up. Due to the flexible design of TOPP it is also possible to use a peak picker prior to ITRAQAnalyzer, but it is usually not necessary. Data for Predict-IV is retrieved from the instrument in centroided format.

ITRAQAnalyzer also supports normalization of each channel by a factor using median of ratios (see Subsection 5.2.5 for more details).

Isotope Correction via Non-Negative Least Squares

iTRAQ reagents cause the transfer of signal from one channel to neighboring channels on either side due to reagent impurities affecting the mass of the iTRAQ tag (see Figure 5.1 for an

Table 5.1: Default isotope correction table for 4-plex iTRAQ. The values at offset 0 were inferred by computing $1 - \sum_{p \in \{-2,-1,+1,+2\}} k_{ch,p}$. For simplicity we report values in percent instead of relative frequencies.

	mass offset				
channel	-2	-1	0	+1	+2
114	0.0	1.0	92.9	5.9	0.2
115	0.0	2.0	92.3	5.6	0.1
116	0.0	3.0	92.4	4.5	0.1
117	0.1	4.0	92.3	3.5	0.1

Table 5.2: Default isotope correction table for 8-plex iTRAQ. The values at offset 0 were inferred by computing $1 - \sum_{p \in \{-2,-1,+1,+2\}} k_{ch,p}$. For simplicity we report values in percent instead of relative frequencies.

	mass offset				
channel	-2	-1	0	+1	+2
113	0.00	0.00	92.89	6.89	0.22
114	0.00	0.94	93.00	5.90	0.16
115	0.00	1.88	93.12	4.90	0.10
116	0.00	2.82	93.21	3.90	0.07
117	0.06	3.77	93.29	2.99	0.00
118	0.09	4.71	93.32	1.88	0.00
119	0.14	5.66	93.33	0.87	0.00
121	0.27	7.44	92.11	0.18	0.00

example). A table can be constructed which lists the relative amount of signal that is lost to the two immediate neighboring channels.

Default isotope correction tables for 4-plex and 8-plex kits are provided but can be changed by the user. Table 5.1 and 5.2 give an overview of the defaults in ITRAQAnalyzer. $k_{ch,p}$ denotes the relative loss in channel ch to position $p, p \in \{-2, -1, +1, +2\}$. The 8-plex table was obtained directly from AB Sciex[2]. The 4-plex table is taken from the Certificate of Analysis provided with all 4-plex iTRAQ kits. Both matrices are stable (personal communication with AB Sciex support); thus, modifying the defaults should not be required.

Isotope impurity will not lead to excessive loss of signal for the respective channel, i.e., loss $< 8\%$ (see Tables 5.1 and 5.2) but might cause severe signal gain in neighboring channels, which should be corrected. Table 5.3 provides an example where failure to correct for isotope impurity results in a peptide ratio of ≈1:24, whereas the true ratio is 1:100. As this can happen for high reporter ion intensities, no noise model will be able to correct for this. Thus, isotope correction should always be applied.

Almost all iTRAQ software packages available are capable of isotope impurity correction. Some ignore the problem and expect isotopically corrected values, e.g., iQuantitator [142]. There are two solutions for isotope correction we are aware of, which we will describe in the following. To ease reading, we will use bold letters for vectors, bold capital letters for matrices, and plain

[2] http://www.absciex.com/Documents/Support/AB_SCIEX_Question_and_Answer.xls

Table 5.3: Isotope impurity example resulting in skewed ratios.

	channel				115/116
condition	114	115	116	117	
real	0.00	100	10 000	0.00	1:100
observed	2	392	9 246	450	1:24

Table 5.4: Construction of the matrix $\boldsymbol{A}$ used for isotope impurity correction where $f(ch) = \sum_{p\in\{-2,-1,+1,+2\}} k_{ch,p}$.

$$\boldsymbol{A} = \begin{array}{cccc} 1-f(114) & k_{115,-1} & k_{116,-2} & 0 \\ k_{114,+1} & 1-f(115) & k_{116,-1} & k_{117,-2} \\ k_{114,+2} & k_{115,+1} & 1-f(116) & k_{117,-1} \\ 0 & k_{115,+2} & k_{116,+1} & 1-f(117) \end{array}$$

letters for scalars. The solution used by the majority of open source software packages is to simply solve

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \tag{5.1}$$

where $\boldsymbol{A}$ is an $n \times n$ isotope correction matrix, $\boldsymbol{b} \in \mathbb{R}^n$ is the vector of observed reporter intensities (also called raw intensity values), and $\boldsymbol{x} \in \mathbb{R}^n$ is the isotope-corrected solution. We denote this method as the *naive approach.*

$\boldsymbol{A}$ can be computed from the isotope impurity table given in the Certificate of Analysis as provided by the supplier *AB Sciex*. See Table 5.4 for its construction for 4-plex iTRAQ. The procedure is similar for the 8-plex case.

As a negative peptide abundance is not possible, negative intensities in $\boldsymbol{x}$ are subsequently set to zero, i.e., reporters are treated as "not observed" or "ignored". Software packages which use this strategy include IsobariQ [155][3], isobar [144][4], and Multi-Q [143]. In Mascot [76], negative ratios are flagged and not used for quantification.

The second method used by Thermo in the ProteomeDiscover software (version 1.2) yields rather different results. The algorithm employed is unknown, but we could reproduce consistent underestimation (about 8%) of reporter intensities when compared to the *non-negative least squares* (NNLS) and naive approach across all spectra we examined from one experiment. We only have Thermo isotope correction data available where NNLS (see below) and naive solution are identical. See Table 5.5 for an example.

The constant overestimation points to the fact that the algorithm used by Thermo is also a matrix inversion, as described above, but with different matrix values. We implemented a proper naive approach using diagonal values of $1 - \sum_{p\in\{-2,-1,+1,+2\}} k_{ch,p}$ to reflect the loss of native ions of channel ch to neighboring channels. If we substitute the diagonal values from the matrix of the naive approach with entries of 1, the solutions are exactly the same, i.e., the ratio is one for all channels. Thus we conclude that the Proteome Discoverer software uses a wrong correction matrix $\boldsymbol{A}$.

We propose a third method, non-negative least squares (NNLS), to correct for isotope impurities. Our approach solves

[3] Treatment of negative intensities as zero is not explicitly mentioned in the paper [155], but can be found in the source code (version 1.2) available from http://folk.uio.no/magnusar/isobariq/files/IsobariQ_1.2_source.zip

[4] Negative intensities are ignored altogether, excluding them from protein ratio computations.

Table 5.5: Thermo isotope correction result of one arbitrary MS^2 spectrum compared to the NNLS/naive solution. The ratios between the Thermo solution and NNLS are stable for all examined spectra.

channel	**intensity**		**ratio**
	Thermo	NNLS/naive	
114	3 253	3 500	1.076751
115	5 456	5 904	1.080432
116	5 854	6 329	1.082083
117	4 292	4 651	1.084052

$$\min ||\boldsymbol{Ax} - \boldsymbol{b}||_2, \tag{5.2}$$

with the constraint $\boldsymbol{x} \geq 0$. $\boldsymbol{A}$ and $\boldsymbol{b}$ are the correction matrix and the raw reporter intensities as in Equation (5.1). If the solution to Equation (5.1) is positive the NNLS solution is identical; if it yields negative values, the results will differ.

5.2.5 Protein Ratio Computation

Published papers on iTRAQ dedicate different amounts of effort to describing and carrying out protein ratio computations and a statistical analyses to determine whether the observed changes are significant. About 50% of all studies choose arbitrary thresholds like 1.5 as fold-change cutoffs [87]. However, statistical modeling which provides a p-value is a desirable approach [148].

To properly analyze our data, we use a combination of in-house R code and the isobar package [144], also implemented in R. Even though isobar cannot combine multiple iTRAQ experiments, it offers the most flexible interface with a sound statistical for protein quantification and significance estimation approach, allows for a very high degree of automation, and is sufficiently fast, i.e., data from 27 experiments can be analyzed in about 15 min after identification and quantification results from our TOPPAS pipeline are exported in a comma-separated values (CSV) format readable by R.

Due to sampling bias of highly abundant peptide species in data-dependent LC-MS experiments, simple global normalization is not advisable for MS^1-based methods, as pointed out by Wang et al. [156]. As abundance in MS is cumulative over all channels when using iTRAQ, global normalization is feasible here. Normalization using matched pairs, which are available at no cost in iTRAQ-type experiments, is a viable strategy as well. Both yield similar results (data not shown), and we use median of ratios for normalization across channels to correct for different amounts of sample loaded between channels. The rationale behind normalization is the assumption that most proteins in a sample are not regulated [157], and that thus a global offset should be corrected. The model is fairly simple, yet avoids overfitting of data as we do not know the extent of proteomic changes induced by the toxic compounds. The laboratory will usually ensure that protein concentration in each iTRAQ channel is equal by determining protein concentration using a Bradford assay, but no mixture can be expected to be a precise 1:1 mix [157]. Hence normalization is advisable, in order to remove the systematic shift. We use all iTRAQ spectra where reporter ions are present, even if no identification is available, as this increases the number of pairs and allows for a more robust estimation. After normalization, three technical replicates acquired using exclusion lists (as described in Subsection 5.2.3) are combined.

isobar is then used to compute protein ratios for control versus low dose and control versus high dose conditions for each harvest time point. In detail, protein ratios are computed based on all unique peptides (as filtered in TOPPAS pipeline) and weighted by signal strength, which ensures that ratios from high-abundant peptides receive more weight as they are less influenced by noise. Weighting is not only applied in iTRAQ workflows but also in other labeling techniques such as SILAC [93]. The following formulas used by isobar to compute protein ratios and p-values are taken from the paper by Breitwieser et al. [144] and its Supplemental Material with minor notational modifications. A protein log ratio is computed as

$$c(p_i) = \sum_{j \in S'_i} \alpha_{i,j} c(s_{i,j}), \tag{5.3}$$

where $c(p_i)$ is the protein log ratio for protein i; $S^{'}$ is the set of identified, non-outlier peptide ratios for protein i; $c(s_{i,j})$ is the log ratio of peptide j belonging to protein i identified from spectrum s, and $\alpha_{i,j} = k \cdot \beta_{i,j}$ with k as scaling factor such that $\sum_j \alpha_{i,j} = 1$. A peptide log ratio is computed as

$$c(s_{i,j}) = log(I_{ch2,i,j}/I_{ch1,i,j}). \tag{5.4}$$

Finally $\beta_{i,j} = 1/Var(c(s_{i,j})) = 1/(n(log(I_{ch1,i,j})) + n(log(I_{ch2,i,j})))$. ch_1 and ch_2 represent the channels, i.e., $ch_1, ch_2 \in \{114, 115, 116, 117\}$ (for 4-plex iTRAQ). $I_{ch,i,j}$ is the abundance of a given channel ch, for protein i, with peptide j.

The noise function $n()$ is computed as

$$n(x) = a + re^{-\lambda x}, \tag{5.5}$$

with x as the signal log intensity and a, r, λ as parameters estimated from a dedicated 1:1 experiment (see Subsection 5.2.5). The final noise model is an average over all noise models from channel pairs deemed of high quality.

Weighting peptide ratios is important since low intensity ratios exhibit more variance due to noise. High abundance ratios are thus more stable and should receive more weight [136, 144].

Protein ratio variance $\text{Var}(c(p_i))$ is determined as

$$\text{Var}(c(p_i)) = \max(\text{Var}_{estim,i}, \text{Var}_{spectrum,i}), \tag{5.6}$$

where

$$\text{Var}_{estim,i} = \frac{1}{\sum_j \beta_{i,j}}, \tag{5.7}$$

and

$$\text{Var}_{spectrum,i} = \frac{\sum_j \beta_{i,j}}{(\sum_j \beta_{i,j})^2 - \sum_j \beta_{i,j}^2} \sum_j \beta_{i,j}(c_{i,j} - c(p_i))^2. \tag{5.8}$$

The reason for using max() in (5.6) is that $\text{Var}_{estim,i}$ can become very small if a protein has many peptides; thus, the overall sample variance is taken as fallback (see isobar paper for details). $\text{Var}_{spectrum,i}$ is not computable if only one spectrum is available; thus, $\text{Var}'_{spectrum,i} = (\text{Var}_{estim,i})^{0.75}$ is used as heuristics. Likewise, for two spectra, $\text{Var}'_{spectrum,i} = \max(\text{Var}_{spectrum,i}, \text{Var}_{estim,i})^{0.75}$ is used.

To determine if a protein is differentially expressed, a Cauchy distribution is fitted to the global protein ratio distribution. Per default, a 5% cutoff is selected where proteins are deemed significantly over-/underexpressed. This avoids the need for an arbitrary fold change cutoff as the model adapts to the data naturally.

Additionally, we augment the results with an Entrez Gene ID mapped from the given Swiss-Prot identifier to ease linking of transcriptomics results later on. We also determine the average expression ratio from up to three biological replicate measurements and the number of times a protein was deemed significantly over-/underexpressed by isobar in each dose versus control scenario.

Noise Model

In isobar it is possible to compute a noise model which determines the weight of a peptide ratio when computing the protein ratio. Ideally, the model is derived from a dedicated experiment with 1:1:1:1 channel allocation, measured on the same instrument as the real samples. Alternatively it is possible to derive a model from study data if channel conditions are similar. All data is isotope-corrected beforehand. We tested three different models, two from dedicated 1:1 experiments and one derived from study data, and found them all to be very similar. See Figure 5.7 for a comparison. They all yield almost exactly the same list of significantly expressed proteins (data not shown). However, care has to be taken as experimental errors can severely alter the model – see Figure 5.8 for an example.

In the published version of isobar, normalized reporter intensities are used for creating and evaluating the noise model. As normalization can alter reporter intensities quite significantly, we suggest to use raw reporter intensities instead of normalized intensities since noise is a property of the instrument. We obtained and used a version of isobar which supports this strategy.

5.2.6 Our Contribution

In some areas it might not be explicitly clear where our contribution starts and ends. We use this section to clarify the details. We conceptualized and implemented the ITRAQAnalyzer tool as well as the custom R scripts (for combining multiple biological and technical replicates and wrapping isobar), and extended/rewrote TOPPAS to support the workflow required for this project (in terms of pipeline design restrictions, parallel execution, “dry run” functionality, and input recycling). We constructed the analysis pipelines for the exclusion list workflow as well as the iTRAQ identification and quantification workflow, including parameter tuning and fixing of contributing TOPP tools (e.g., PeptideIndexer to deal with ambiguous amino acids). We helped write the InclusionExlcusionListCreator and extended it to support restrictions on exclusion lists of the Thermo instrument software. We had fruitful discussions with the developers of isobar, who implemented the modified noise model scheme.

5.2.7 Acknowledgments

The Predict-IV project is a joint effort of many international partners, and the data analysis described here would not have been possible without them. In particular, we would like to thank Silke Ruzek (Universität Salzburg) for LC-MS data generation, Paul Jennings (Medizinische Universität Innsbruck) and Martin Leonard (University College Dublin) for handling of cell systems, Philip Hewitt (Merck KGaA) and Hannelore Popa-Henning (Universität Würzburg) for coordination, and many others involved in advancing the project and generating data.

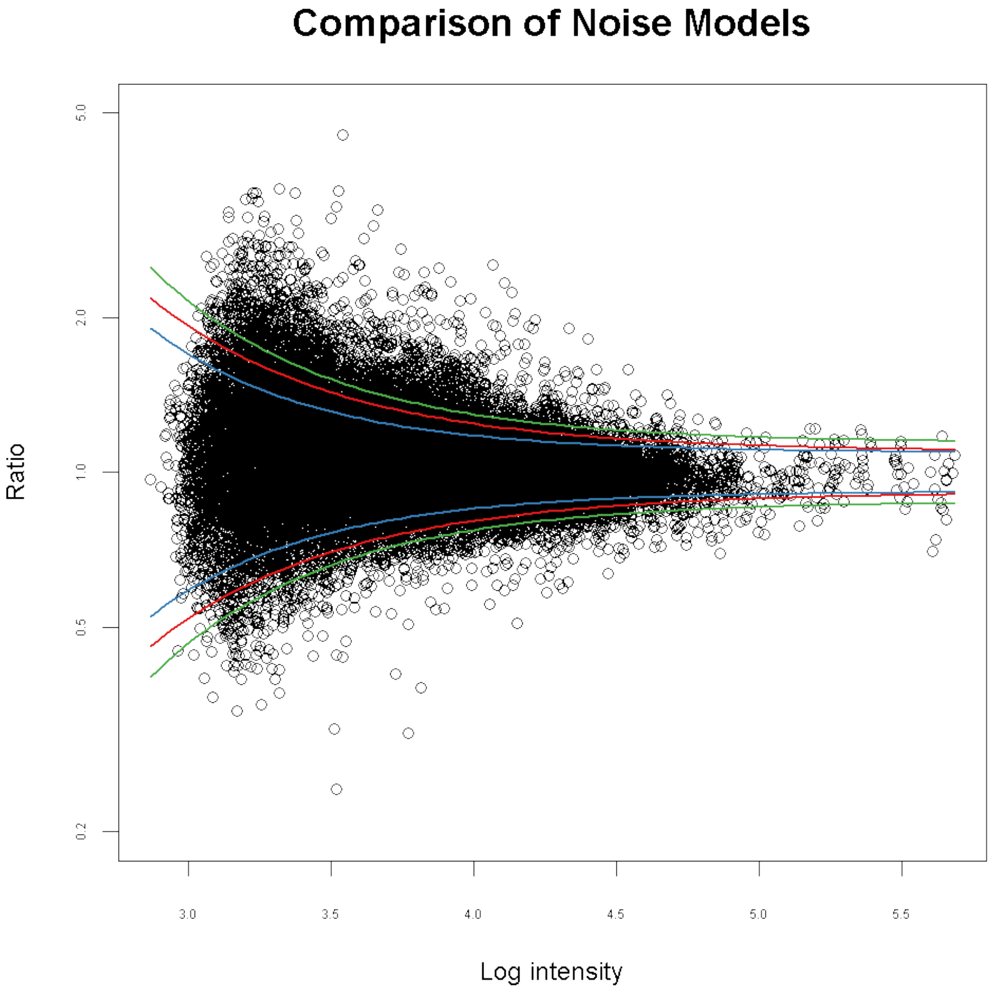


Figure 5.7: Three noise models from 1:1 noise experiments with cells from NUI002 study (red) and IMU006 (blue), and real data from IMU006 (green). All channels were used, except 117 in NUI002.

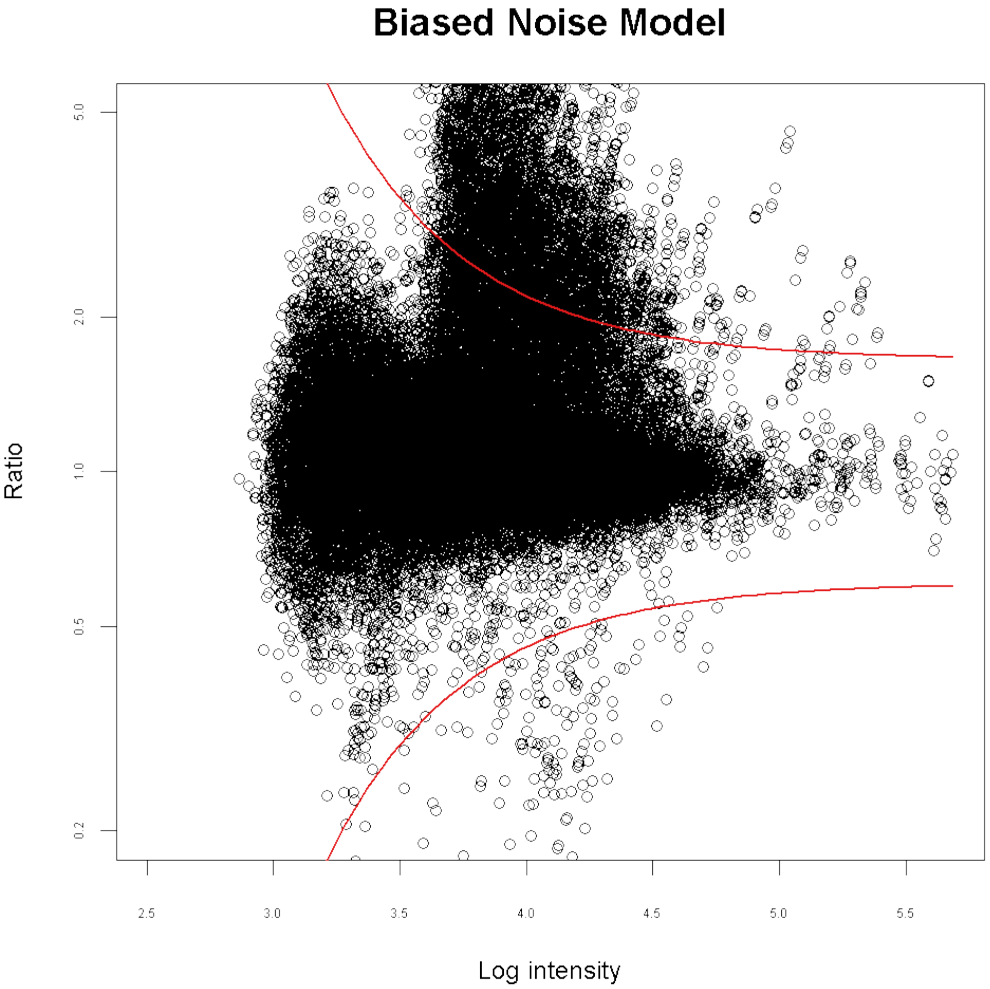


Figure 5.8: Noise model averaged over all channels from 1:1 experiment in NUI002 study, with channel 117 only having 10% intensity of the others.

Table 5.6: Overview of Predict-IV studies analyzed.

study name	species (organ)	toxic compound	conditions[a]	spectra[b]	qtfy. proteins
IMU002	human (kidney)	cyclosporin A	C, L, H	238 818	945
IMU006	human (kidney)	cyclosporin A	C, L, H	389 614	2 642
IMU007	human (kidney)	ifosfamide	C, L, H	519 822	3 656
NUI002-A	human (kidney)	adefovir	C, L, H	?[c]	2 160
NUI002-Hyp-A	human (kidney)	adefovir	C, hC, hL	258 222	2 210
NUI002-Hyp-Z	human (kidney)	zoledronate	C, hC, hL	198 098	1 575
URO001	mouse (brain)	cyclosporin A	C, L, H	330 321	1 674

[a] C: control, L: low dose, H: high dose, h: hypoxia

[b] total number of MS^2 spectra of all 27 LC-MS maps, HCD and CID each contributing 50%

[c] raw data not available

5.3 Results

We analyzed multiple studies from the Predict-IV project and demonstrated properties of iTRAQ data, some of which have already been shown in the literature and some of which are novel. We also use this data to show the performance of our algorithmic approach in order to maximize information gain in terms of quantification and identification results.

An overview of the data analyzed can be found in Table 5.6. Identification and quantification performance varied greatly between studies. Especially initial studies suffered from errors during iTRAQ labeling. Also, insufficient amount of protein material occurred sporadically. Both affected the number of identifiable/quantifiable proteins.

5.3.1 Labeling Efficiency

We evaluated labeling efficiency and compared it to the number of identified peptides. Data from a preliminary study and third-party iTRAQ data (not shown) suggested that a labeling efficiency of ≈80% is feasible.

IMU002 yielded a rather small result set due to errors during iTRAQ labeling (see Table 5.6). Therefore, the entire analysis, starting from cell growth, was repeated as IMU006, yielding about thrice as many quantifiable proteins.

To elucidate if labeling efficiency can serve as a quality control measure, we compare the number of identified peptides (after FDR filtering) with labeling efficiency, which can be computed very quickly (see Figure 5.9). In most cases, labeling efficiency is good at ≈80%, and peptide count only slightly declines for subsequent replicate runs with exclusion list. When labeling efficiency is very poor ($< 50\%$), no peptides can be identified. This is the case for two samples from IMU006, where protein content was not sufficient for a third technical replicate run. Labeling efficiency of IMU002 is poor in general as already reflected by the number of quantifiable proteins (cf. Table 5.6).

Thus, labeling efficiency can be used to exclude low quality data sets from further time-consuming analysis. We note that the TIC on the MS level is not indicative of labeling efficiency (data not shown).

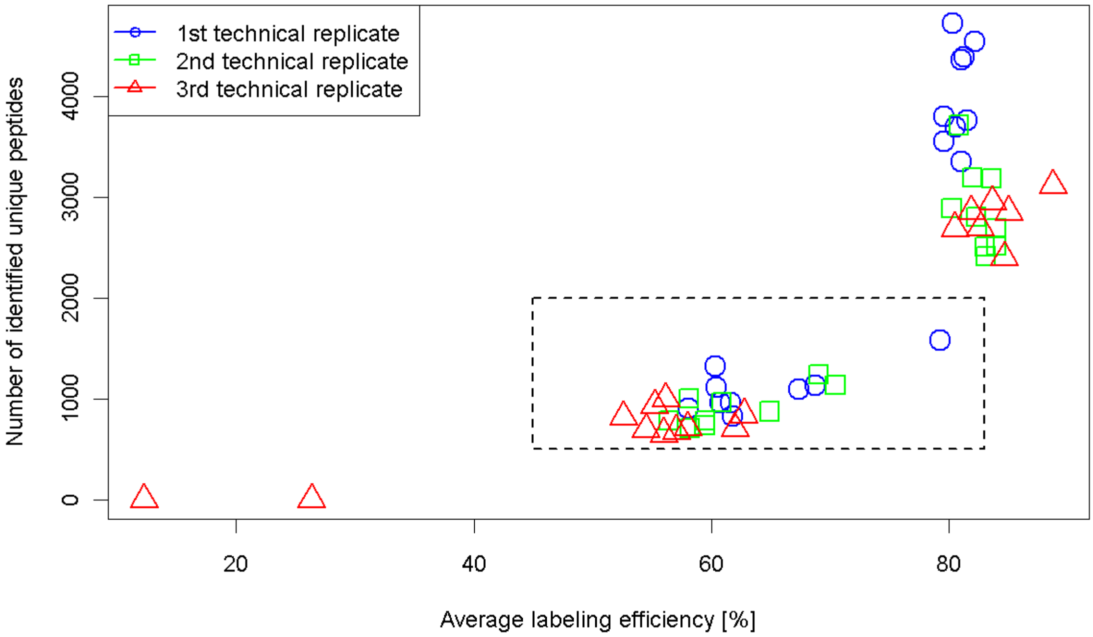


Figure 5.9: Labeling efficiency vs. the number of identified peptides uniquely matching a single protein after FDR filtering. Each point represents one LC-MS experiment from either IMU002 (inside dashed rectangle) or IMU006 (outside dashed rectangle).

5.3.2 NNLS Isotope Impurity Correction

We now motivate why NNLS isotope impurity correction is superior to naive inverse matrix multiplication. Theoretically, the absolute difference between Equation (5.1) and (5.2) can become arbitrarily large, e.g., consider $\boldsymbol{b} = \{1e5, 0, 1e3, 0\}$ and $\boldsymbol{A}$ as in Table 5.4. Due to the nature of $\boldsymbol{A}$, we get

$$\begin{aligned}\boldsymbol{x}_{naive} &= \{107\,792, -6\,932, 1\,271, -54\}\\ \boldsymbol{x}_{NNLS} &= \{107\,209, 0, 626, 0\}\end{aligned}$$

for the naive and NNLS approach respectively. Note that we report the immediate results of the inverse matrix multiplication directly. We do not set negative intensities to zero to show the implicit error made by the naive approach.

The difference becomes more pronounced the more diverse we chose the entries for i_{114} and i_{116}. When $\boldsymbol{b}_1$ is large enough, $\boldsymbol{x}_3$ will become zero for NNLS, but will increase for the naive approach. For example, for $\boldsymbol{b} = \{1e7, 0, 1e3, 0\}$ we get

$$\begin{aligned}\boldsymbol{x}_{naive} &= \{10\,779\,110, -689\,658, 19\,557, -206\}\\ \boldsymbol{x}_{NNLS} &= \{10\,720\,973, 0, 0, 0\}\,.\end{aligned}$$

A realistic input might be $\boldsymbol{b} = \{100, 0, 1, 0\}$, giving

$$\begin{aligned}\boldsymbol{x}_{naive} &= \{108, -7, 1.27, -0.05\}\\ \boldsymbol{x}_{NNLS} &= \{107, 0, 0.63, 0\}\,.\end{aligned}$$

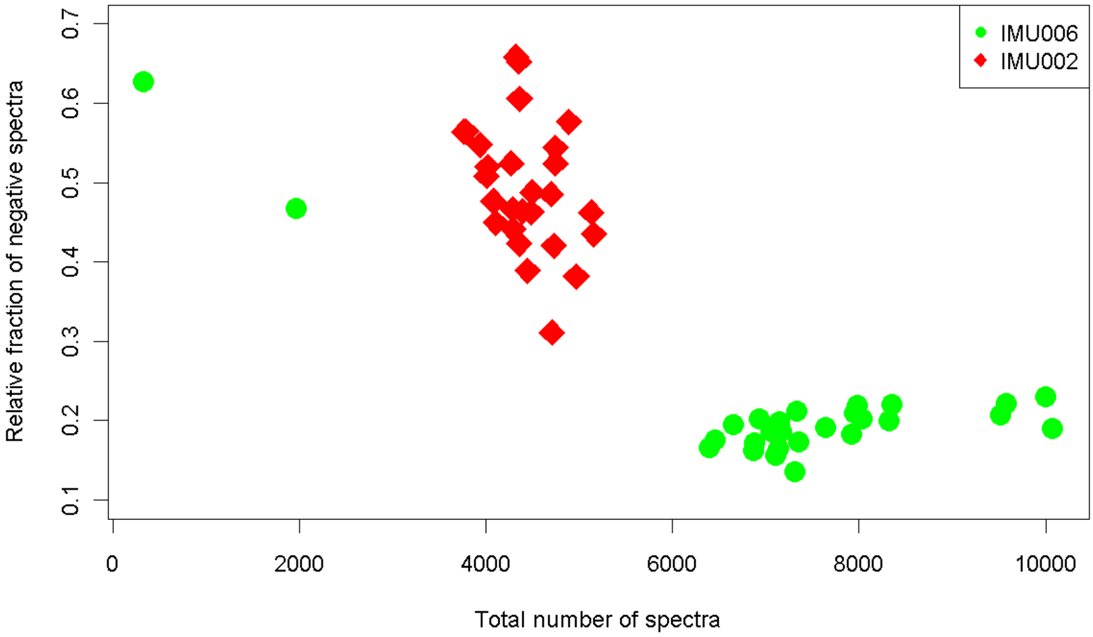


Figure 5.10: Relative frequency of HCD MS^2 spectra from all data sets from two studies which have one or more intermediate negative reporter intensities during isotope correction.

As can be seen, in relative terms, the difference for i_{116} is about 2:1. Ratios between i_{114} and i_{116} would thus yield very unstable results. Fortunately, most software packages use outlier filtering and/or take reporter intensity into account by using a noise model where low intensities receive a low weight for the protein ratio computation.

Also note that any spectra with just one reporter channel present will result in the naive approach predicting two non-zero channel intensities. One at the initial non-zero position ch_x, the other at the next but one reporter position ch_y ($|ch_x - ch_y| = 2$), which we call a *ghost peak*. NNLS will only report the initial position ch_x as non-zero. For example, take $\boldsymbol{b} = \{0, 0, 1\,000, 0\}$, giving

$$\boldsymbol{x}_{naive} = \{1, -35, 1\,087, -53\}$$
$$\boldsymbol{x}_{NNLS} = \{0, 0, 1\,079, 0\}\,.$$

In terms of least squares, our solution is more appropriate and not ad hoc as negative values need not be set to zero, and it does not suffer from ghost peaks.

The reason for obtaining negative intensities is most likely due to noise affecting the reporter ions. As noise is not accounted for beforehand, the naive solution can yield negative results.

Figure 5.10 shows the relative frequency of all HCD MS^2 spectra which have one or more intermediate negative intensities when using the naive solution. This statistic is provided by default by our ITRAQAnalyzer tool for each data set. Studies can be grouped very efficiently by this measure alone (similarly to labeling efficiency – see above). Thus, for high quality data as in IMU006, we can expect ≈20% of spectra to show this behavior (except for two outlier data sets where injection volume was insufficient); for low quality spectra with poor iTRAQ labeling as in IMU002, the values are between 30-68%.

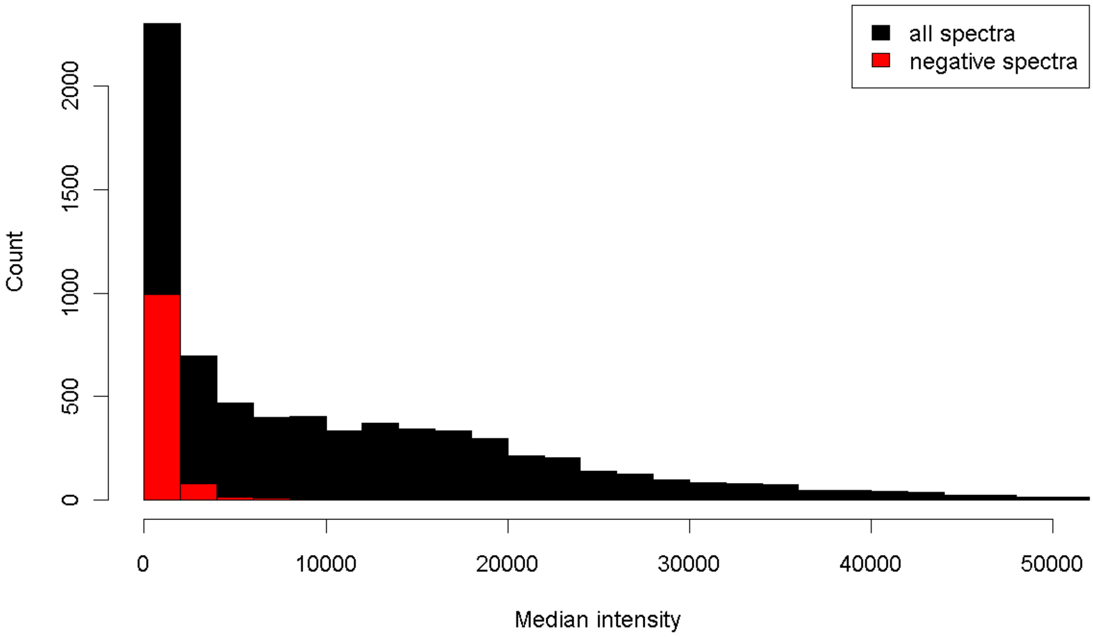


Figure 5.11: Median raw reporter abundance of spectra where at least one negative intermediate reporter abundance occurred vs. all spectra, as observed in one IMU006 data set. Negative intensities are prevalent in spectra where raw abundance is low.

This suggests that mostly low-abundance spectra and/or badly labeled spectra are affected. We found that the total ion count (TIC) is not indicative of this behavior, as for both studies of IMU002 and IMU006, the TICs are comparable. However, there is a big difference (approximately ×5) in the median of reporter ion intensities between the studies, which points to an error during iTRAQ labeling (data not shown).

For spectra with low reporter counts (e.g., due to bad labeling efficiency or simply low abundance of the fragmented peptide), one can expect that some channels have zero as raw intensity. This will inevitably lead to different results of the naive versus NNLS solution as according to the isotope correction matrix for any channel i with abundance $\boldsymbol{b}_i > 0$, every neighboring channel should have a positive intensity. If the neighboring channel has raw intensity zero, inverse matrix multiplication will correct this to a negative value. Figure 5.11 confirms this hypothesis. Data for other data sets are similar (not shown).

The missing peak (with low expected raw abundance) can be explained by the lack of sensitivity of the instrument or too stringent internal noise filtering algorithms. Given that the isotope correction table is correct, the solution of NNLS is thus more realistic since it implicitly takes the missing values into account.

We found unusually high ratios in iTRAQ data, which was corrected using the naive method. For the 1:1 mix used as part of the noise model estimation, we obtain more extreme ratio distributions using the naive method than with NNLS. In the naive method, ratios go up to 1:8 914, whereas for NNLS, the maximum ratio is 1:312. For an overview of ratios, see Figure 5.12 with dependence on abundance. The outlier ratios for the naive approach are clearly visible. Due to the nature of the naive approach, certain ratios occur more frequently, visible as horizontal

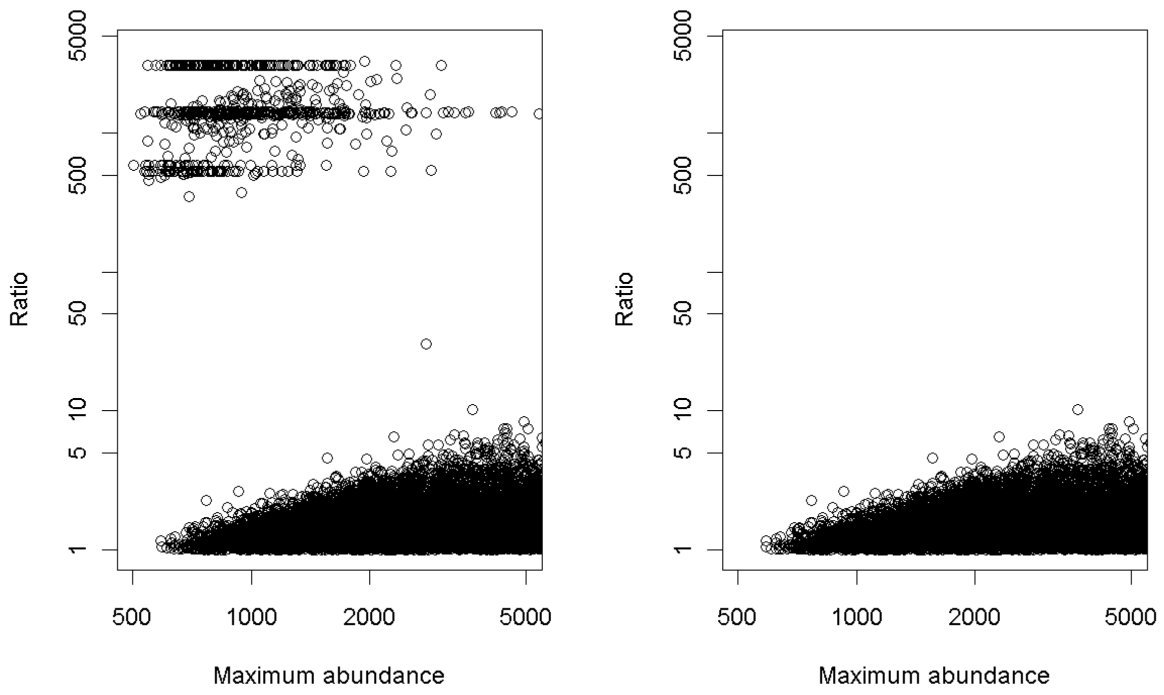


Figure 5.12: Zoomed view of reporter abundance (maximum of two values used for ratio) vs. the resulting ratio after isotope correction. For convenience, if the ratio x:y is below one, we show the ratio y:x such that only ratios larger than one are plotted. Left) Correction via naive method. Right) Correction via NNLS method.

traces in Figure 5.12.

NNLS yields a different solution for about 20-68% of the spectra (see Figure 5.10), depending on data quality. Differing solutions are predominantly observed in low-intensity spectra. Similar to labeling efficiency, the fraction of conflicting solutions between naive and NNLS approach can be used as a data quality criterion, i.e., a spectrum with diverse solutions to isotope correction is most likely suffering from lack of instrument sensitivity. Fortunately, low intensity reporter ions are usually weighted down when protein abundance ratios are determined, for example, by using a noise model. However, to our knowledge, all noise models published to date estimate their parameters after isotope correction has been performed [158, 159, 144]. If the instrument was truly sensitive in the low-intensity range, however, we would expect small intensity values (due to isotope impurities) in the raw data instead of zero intensity values. Reasons for the absence of the peak are unknown, but as the data was acquired on an Orbitrap, limited trap capacity or acquisition time could play a role.

To summarize, in contrast to the naive method, NNLS does not create ghost peaks and reduces the need for outlier detection since no extreme ratios are generated; ratios where zero-intensities are involved are usually handled by all quantification approaches correctly by being excluded from further computations.

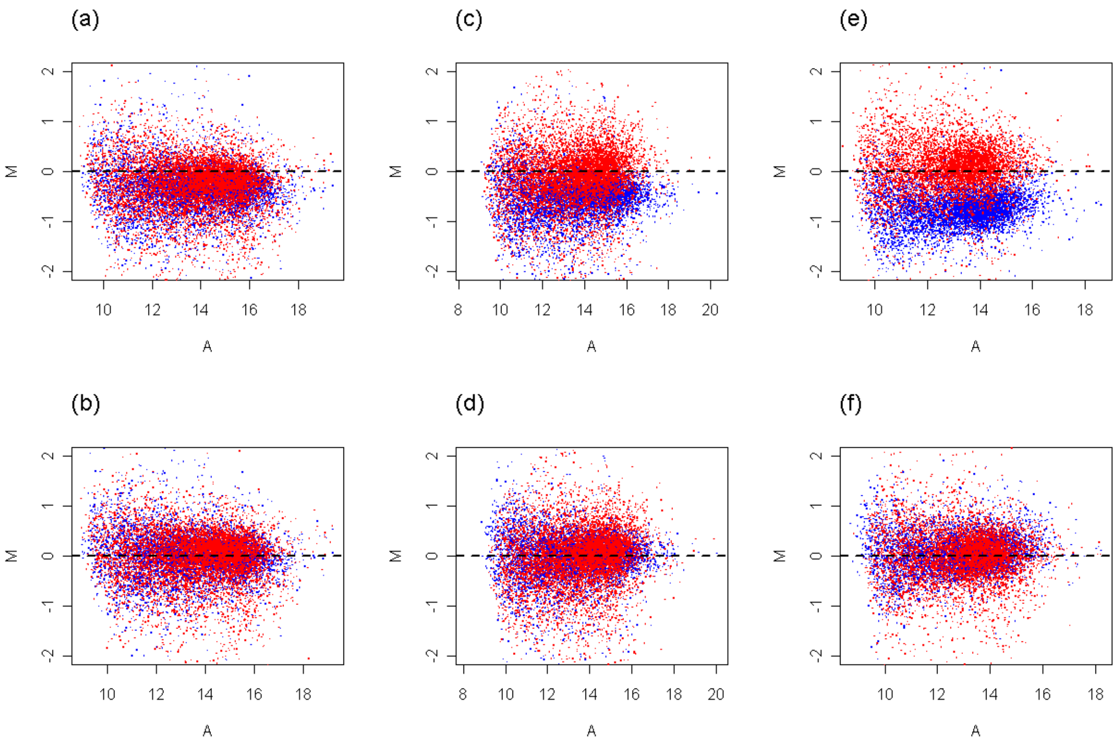


Figure 5.13: MA plots of three random experiments from the IMU006 study. The upper plots (a, c, e) show MA plots of intensity values before normalization, the bottom plots (b, d, f) show the intensity values after normalization. Channel 114 was used as reference channel. Color-coded ratios of channel 115 (blue) and channel 116 (red) shown.

5.3.3 Normalization

We used median-of-ratios normalization to correct for global bias in protein amount. To check the result visually, we use an MA plot (minus versus average plot) where $M = log_2(I_{ch1})) - log_2(I_{ch2})$ and $A = (log_2(I_{ch1})) + log_2(I_{ch2})) \times 0.5$. For normalized data we expect the majority of points on the y-axis to be located at 0 ($= log(1)$) with no noticeable dependency on the intensity (x-axis). Results for three random experiments from IMU006 are shown in Figure 5.13.

The distribution of normalization factors for IMU006 is shown in Figure 5.14. The most extreme factor was 2.64, which shows that normalization is absolutely critical to avoid wrong protein quantification values. Reasons for deviation from the expected 1:1 mixture are manifold and include pipetting errors, lack of sensitivity of Bradford assay and incomplete labeling.

As a consistency check, we compared the normalization factors of all three technical replicates as the sample mixture is identical for all three data sets. The normalization factors are almost identical for all data sets we examined (data not shown).

5.3.4 Proteome Coverage

Exclusion List

We compare the number of proteins identified from a list of FDR-filtered, unique peptides for two data sets, namely IMU002 and IMU006. The former was acquired using the built-in exclusion functionality of the Thermo software, the latter using our exclusion list approach as described in Subsection 5.2.3.

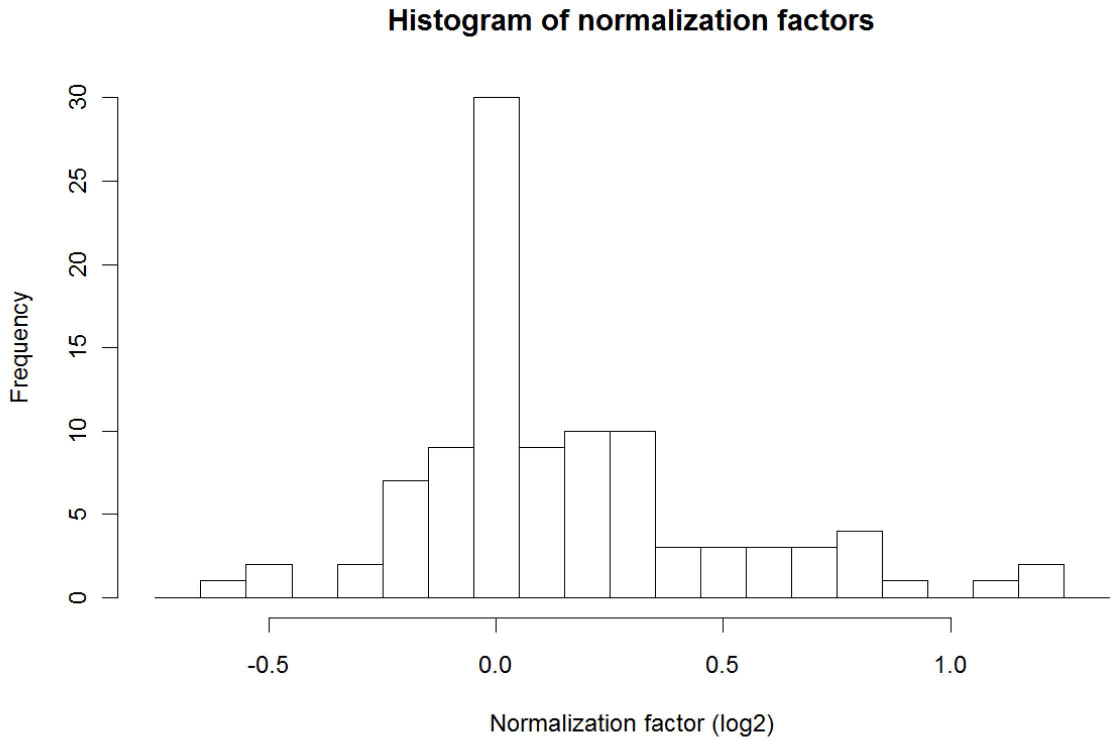


Figure 5.14: Histogram of normalization factors for IMU006. The most extreme factor is 2.64.

Figure 5.15 shows the gain in the number of identified proteins in the second and third replicate measurement with cumulative exclusion list. The first measurement is always performed in DDA mode without exclusion. For both methods the gain is about 27% for the second replicate and 44% for the third, when compared to the first initial measurement. Unfortunately, there is no common data set where both approaches were applied; thus, it is very hard to assess which performs better.

With the data at hand, it is possible to compute the gain in the number of proteins for biological replicates without exclusion lists from the first technical replicates, which were acquired using pure DDA. Note that the exclusion list approach uses technical replicates and thus has no biological variation. Using biological replicates without exclusion will most likely trigger DDA acquisition to focus on different proteins as expression values vary between biological replicates. Using only the first technical replicate from each of the nine iTRAQ samples, we obtain three groups with three LC-MS experiments each. As the order of evaluation can play a significant role, we use the mean of all permutations of the group in order to estimate the gain in protein numbers. Results are superimposed in Figure 5.15 for studies IMU002 and IMU006. Performance of this pure-DDA approach is comparable to both exclusion list approaches. Even though biological replicates should favor a deeper sampling of the proteome compared to technical replicates, the gain (if any) of exclusion lists is much smaller than expected.

We investigated the overlap of identical peptide identifications between subsequent runs using exclusion lists. The results are shown in Figure 5.16.

Evidently, masses and retentions times which were present on the exclusion list were nevertheless selected for fragmentation. We conclude that the instrument software was either misconfigured during acquisition and/or ignored the provided exclusion list. This phenomenon is not

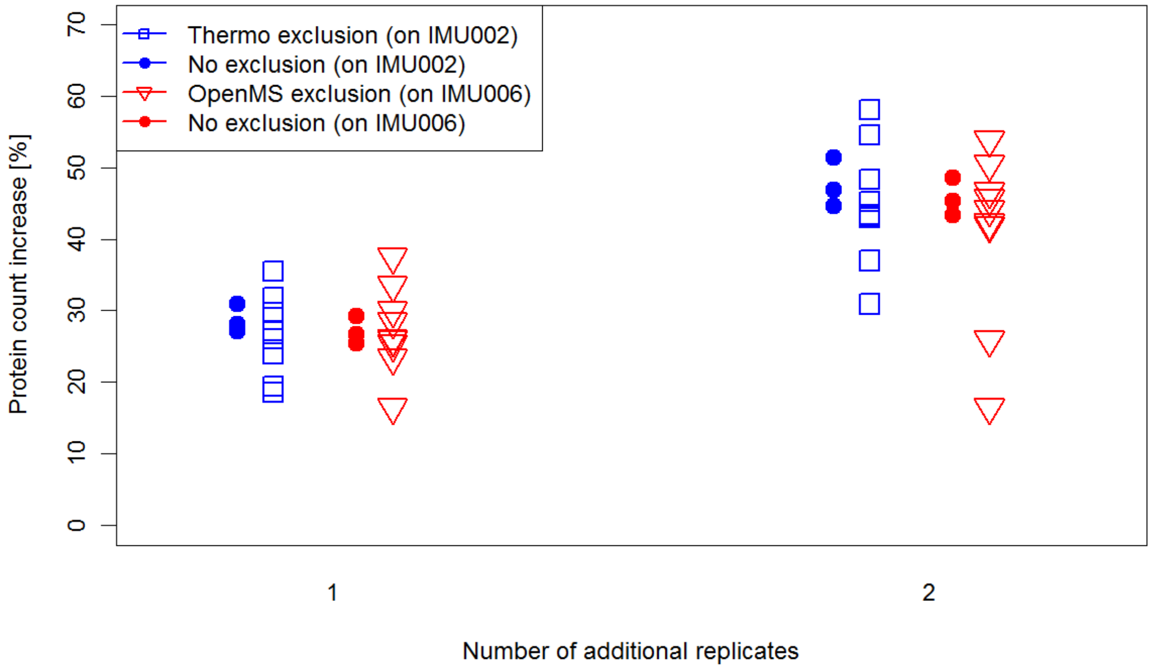


Figure 5.15: Gain in protein count for Thermo exclusion lists (without charge exclusion) and OpenMS exclusion lists (with charge exclusion). Performance without usage of exclusion list (circles) computes the gain in the number of proteins from three pure-DDA experiments of the same data set.

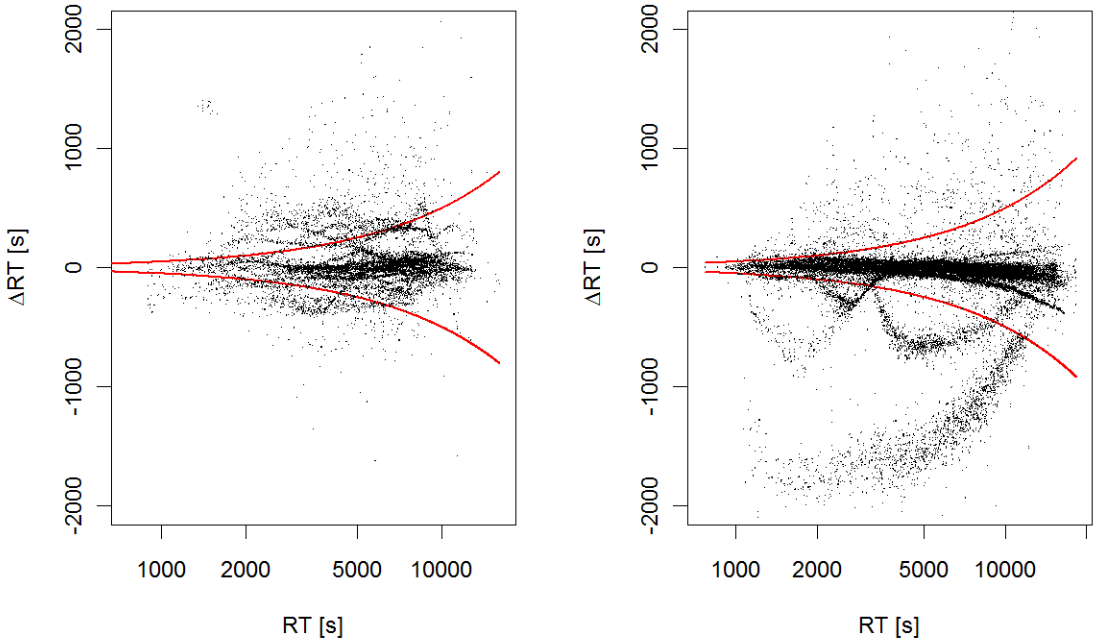


Figure 5.16: Offset in RT of identical peptides between replicate measurements for IMU002 (left) and IMU006 (right) for all replicates. IMU002 uses native Thermo exclusion lists whereas IMU006 uses our approach. The red boundaries mark the RT window which is (theoretically) excluded from fragmentation, i.e., in the area between the boundaries no points are expected if exclusion were perfect. Nevertheless, within the red boundaries multiple traces can be identified in both studies, each trace representing one replicate experiment.

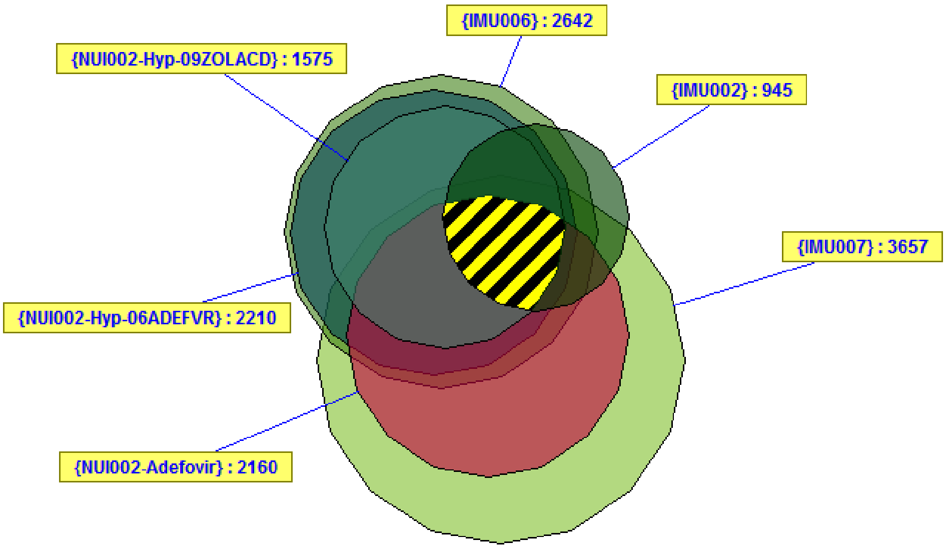


Figure 5.17: Euler diagram of the identified protein sets of six Predict-IV studies, each comprising 27 LC-MS experiments. The total number of unique proteins per study is reported in yellow labels. The size of overlapping regions is proportional to number of proteins common to the respective studies. The intersection of all studies contains 369 proteins (yellow striped region). In total, 6 549 unique proteins were identified.

restricted to the exclusion list we generated for IMU006, but is also apparent in IMU002 where native Thermo exclusion lists were used.

Based on this data we cannot prove our method to be superior as there is no common data set where precursor exclusion was successfully carried out. However, based on the statistics shown in Figure 5.6, our approach should reduce redundancy by about 20% compared to the native approach.

Studywide Overlap

In addition to investigating the gain in protein identifications due to technical/biological replicates, we looked at the overlap of protein lists between studies in order to assess the coverage of the whole proteome of the respective cell systems. Results from literature indicate a possible range of approximately 20 300 to approximately 83 800 proteins resembling the human proteome, but this figure does neither take into account the unknown extent of pseudogenes, nor alternative splicing or cell type specific expression [160]. The UniProt database has a size of 35 230 proteins (including predicted).

See Figure 5.17 for an Euler diagram of six human renal proximal tubular cell (RPTEC-TERT1) studies. The intersection of protein lists from all studies contains 369 proteins, the union contains 6 549 unique proteins. This number is far below the most modest estimate of proteome size and illustrates that full proteome coverage is not easily achievable due to various reasons, such as limited acquisition speed and dynamic range of instruments [161].

The overlap can be expected to further decrease with increasing numbers of studies – a common property of the stochastic data-dependent acquisition strategy that was used. Once candidate proteins with expression values indicative of the condition have been identified, targeted proteomics methods, e.g., multiple reaction monitoring (MRM), can be employed for validation.

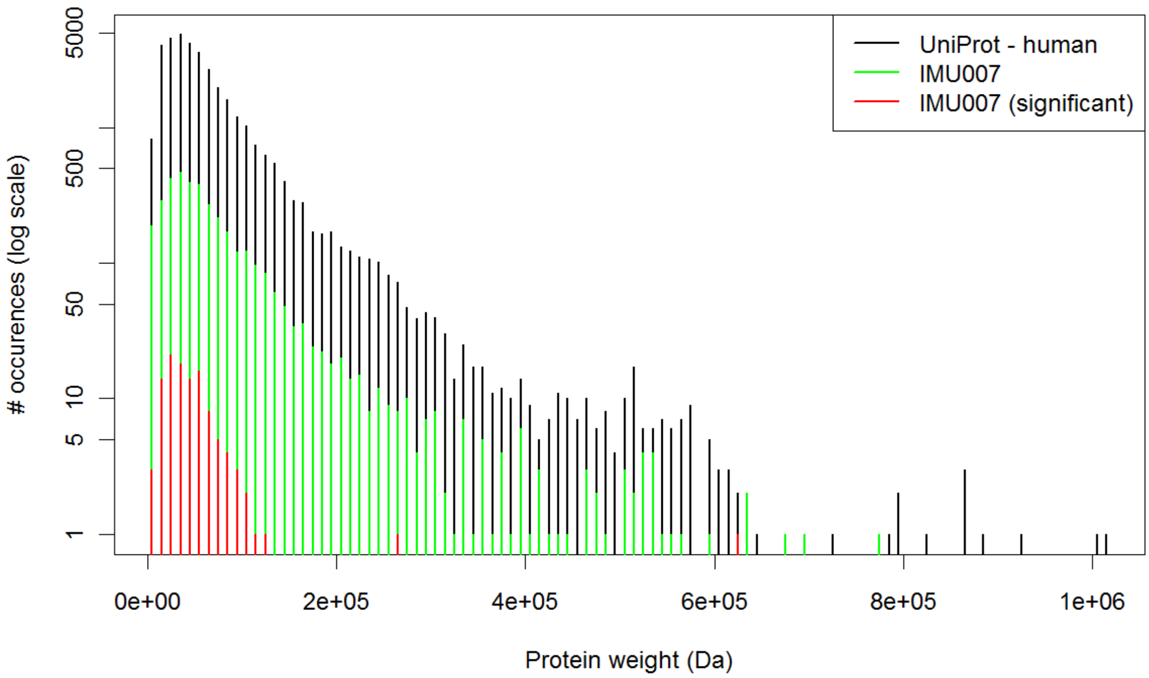


Figure 5.18: Histogram of protein masses from IMU007 study vs. the entire human UniProt database.

Protein Length Bias

We compared the weight of proteins identified in the IMU007 study to all proteins in the UniProt database (35 230 proteins). IMU007 was chosen because this study features the highest number of proteins identified. Results (cf. Figure 5.18) show a small trend towards the identification of lighter proteins. A preference of heavier proteins cannot be observed (as reported in [148]), but as our sampling strategy favors highly expressed proteins, the expression pattern is probably the determining factor for the observed distribution.

5.3.5 Ratio Underestimation

Global ratio underestimation is a known problem in iTRAQ experiments and has been observed in a number of publications. One possible explanation is contamination of the precursor selection window [136, 83] due to overlapping MS^1 features, which all carry the iTRAQ label [12, 83]. This assumption is reasonable since a tryptic digest contains a large number of (non-tryptic/partly-tryptic) peptides, which give rise to a dense proteolytic background [162]. Assuming that most peptides are not differentially expressed, a tendency towards the null hypothesis (i.e., log ratio equals zero) will be observed for a differentially expressed peptide in case the MS^2 isolation window includes a non-/adversely-regulated peptide. Thus, the ratio underestimation affects accuracy since ratios are compressed towards zero (on log scale) [136], even though the overall trend of over-/underexpression is preserved.

For iTRAQ quantification we acquired HCD spectra using an isolation window of ±1 or ±2 Th (depending on the data set). The Thermo software for Orbitraps will always select the monoisotopic peak as precursor m/z and extend the isolation window equally to left and right.

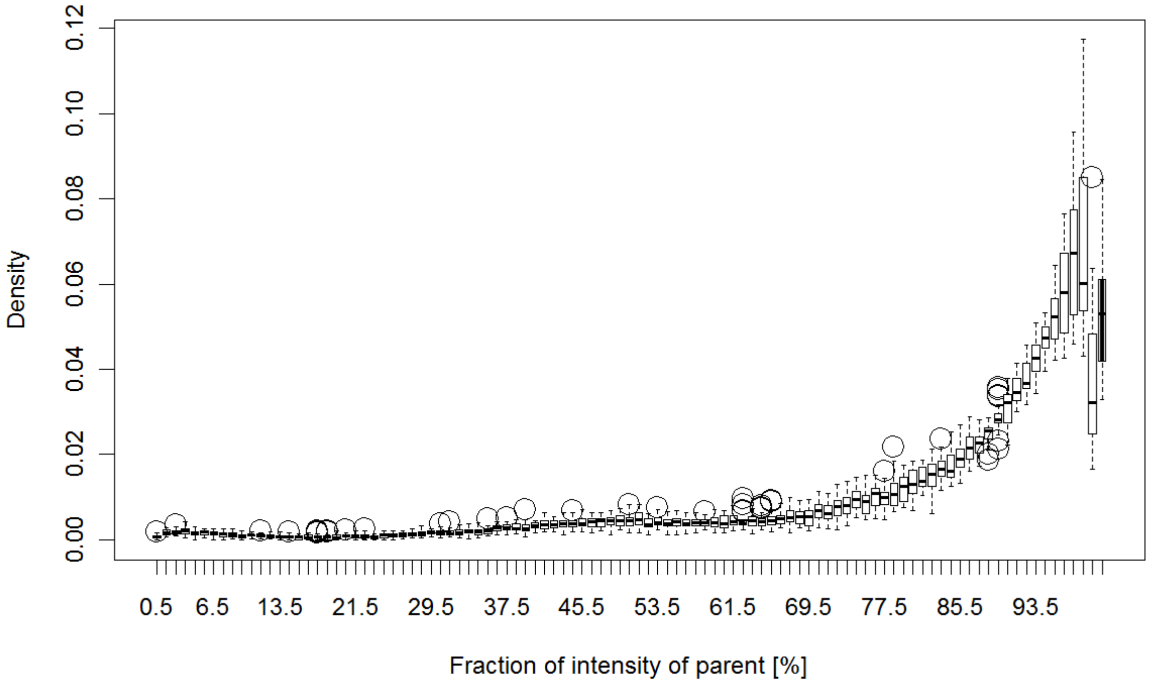


Figure 5.19: Density plot of the fraction of intensity that can be explained by peaks that belong to the precursor ion, compared to the total intensity observed in the precursor isolation window. Values for each interval represent data from 27 data sets from the IMU006 study. The rightmost bar (black) represents the set of precursors with no background signal at all (≈5% of all precursors).

On the left side, however, nothing is to be gained; only the chances of background inclusion increase.

Within the precursor isolation window (2-4 Th) we computed (on the MS^1 level) the fraction of signal, that can be attributed to the isotope pattern of the identified peptide compared to the overall signal. Isotope peaks below the monoisotopic peak resulting from iTRAQ impurities were also included. All other “rogue” peaks with m/z values between isotope peaks (outside of mass tolerance of 0.02 Da) were classified as background noise. An example can be seen in Figure 5.1a: peaks colored in blue or red are counted towards the foreground peptide signal, peaks colored in black are classified as rogue peaks. To our knowledge, this is the first work using this approach. See Figure 5.19 for the results from IMU006. Data for other studies look similar (not shown). Thus, only a small subset of precursors (≈5%) have no other background signal within the precursor window. Most precursors have background noise, contributing about 20% to the total intensity in the isolation window. Very few precursors are more severely affected. This could explain the slight underestimation of iTRAQ ratios when assuming that most overlapping background signals are peptides which are not differentially expressed and carry an iTRAQ label, thus causing a trend towards ratio underestimation.

We also investigated in what way overlap in MS^1 influences the probability of a peptide ratio as being tagged as an outlier during protein quantification. We found that outliers do not show a different overlap behavior from non-outliers (see Figure 5.20). It is important to note here

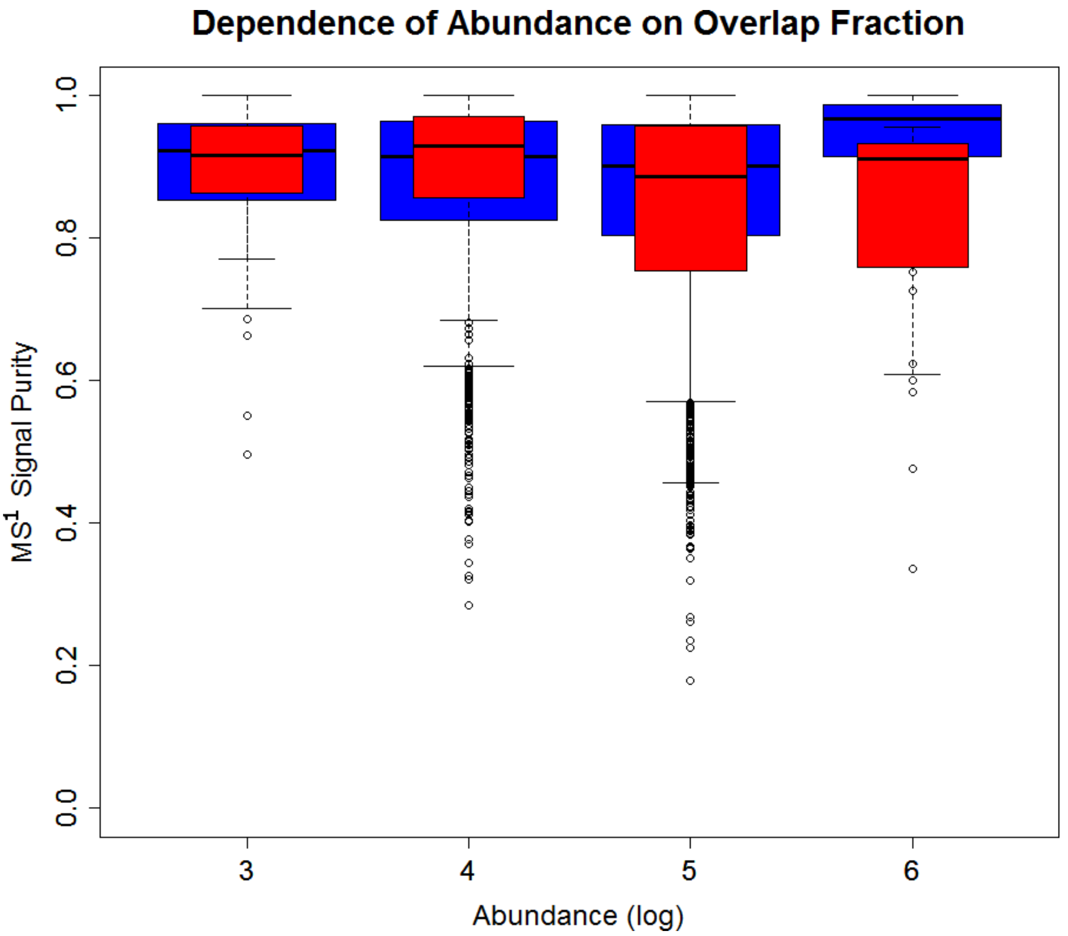


Figure 5.20: MS signal purity in each abundance bucket for non-outlier ratios (blue) and outlier ratios (red). No dependency of intensity on overlap is present for outliers.

that we used the reporter ion signal as a reference for abundance. Karp et al. [136] found that fragmentation efficiency of the iTRAQ tag is peptide-dependent by looking at the 145 Da peak in MS^2 for different peptides and comparing it to the reporter peak intensity. Thus, reporter ion abundance does not necessarily correlate with feature abundance. This is also backed by our data (not shown).

Finally, we looked at the distribution of outlier peptides (as tagged by isobar) depending on their MS^2 reporter abundance. Outliers are predominant for low abundance reporters. This is not surprising as ratios of low reporter ions are less stable (see Figure 5.21).

5.3.6 Detection of Potential Biomarkers

For a powerful biomarker analysis, proteome coverage should be as high as possible while especially considering the low intensity regions since current results indicate that established biomarkers are usually in the low abundance domain [15]. Our analysis of proteome coverage (see Subsection 5.3.4) indicated that only a rather small subset of the proteome was covered, most likely biased towards more high abundance proteins, as DDA (with exclusion lists) was used, where preferentially highly abundant precursors are selected for fragmentation. For exploratory studies where the goal is to find a list of potential biomarkers, the field currently lacks a more powerful alternative which is feasible for our purposes in terms of time constraints and proteome coverage [163]. However, the approach could be modified to achieve even higher

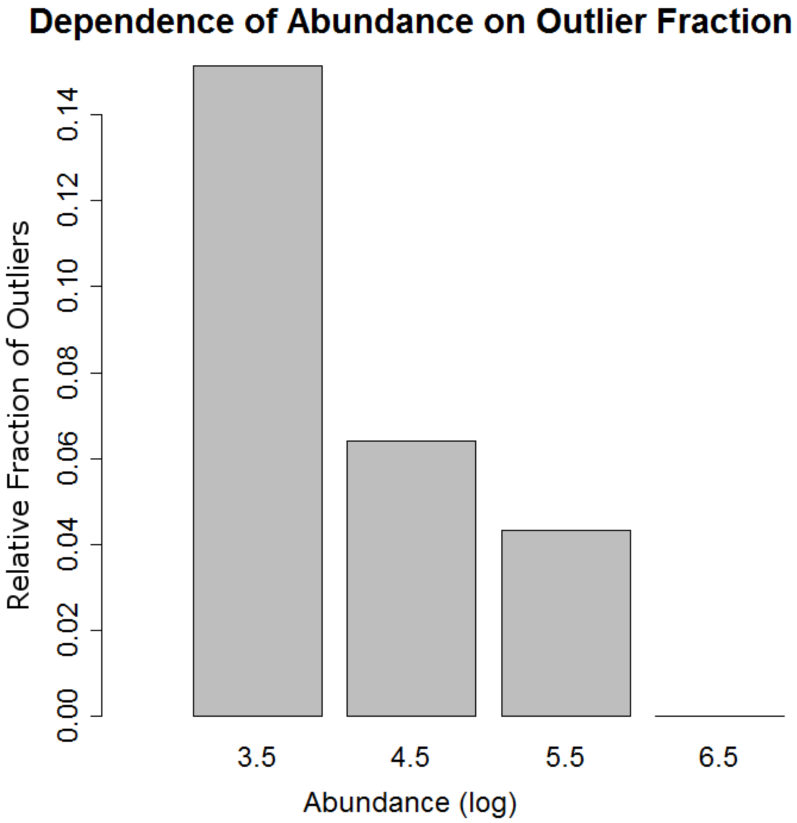


Figure 5.21: Relative fraction of outliers per reporter ion abundance bucket in IMU006. Outliers appear more often in spectra with low iTRAQ reporter abundance.

Table 5.7: Number of proteins classified as significantly expressed per study.

study	# significant proteins
IMU002	20
IMU006	218
IMU007	112
NUI002-Adefovir	164
NUI002-Hyp-06ADEFVR	156
NUI002-Hyp-09ZOLACD	114
URO001	128

proteome coverage, e.g., by using extensive prefractionation at the cost of acquisition and analysis time, and/or using the latest instrument generation which allows for acquisition of more precursor ions.

Biologically interpreted and confirmed results are not readily available for all but one study. For IMU006, extensive data analysis and interpretation is currently carried out and prepared for publication in Wilmes et al. [36]. In brief, IMU006 investigates the effect of the immunosuppressive compound cyclosporin A (CsA) on the proteome, transcriptome and metabolome level. CsA at high dosage (15 μM) caused significant alterations on all omics platforms. The protein ubiquitination pathway and Nrf2-mediated oxidative stress response were significantly altered in proteomics and transcriptomics analysis, thus supporting each other. Results were validated using western blots. At low dosage (5 μM), no major cellular perturbations are observed. A more detailed evaluation will be reported elsewhere [36].

As one of the main goals of Predict-IV is to find common markers of toxicity across compounds and cell systems, we will now report on the results of an integrated analysis using all available proteomics data.

Comparative Approach

Table 5.7 provides an overview of the number of proteins classified as significantly expressed per study. Our goal is to find candidate proteins which are consistently over- or underexpressed in all studies and can be used as predictive markers for toxicity.

We compared protein expression ratios for control versus high dose samples from day 14 of IMU002 with the values from IMU006, which is a repetition of IMU002. Results are shown in Figure 5.22. The Pearson correlation of all common protein log ratios is 0.69 and rises to 0.77 for proteins which have been deemed significantly expressed in either study. As data from IMU002 was confirmed to be of low quality in terms of proteome coverage and iTRAQ reporter intensity, results might be overly pessimistic.

Looking at all studies combined, computing a simple intersection of significant proteins across studies is not advisable since proteome coverage varies greatly between studies; thus, a protein (whether significant or not) is not likely to be observable in all studies and the probability of excluding a promising candidate is increasing rapidly with every additional study that is added. Not surprisingly, the intersection of significant proteins of all six studies with human tissue is empty.

We thus use an approach on the pathway level which allows for missing values (i.e., protein quantifications) while still being robust. We submitted the protein lists for each study to

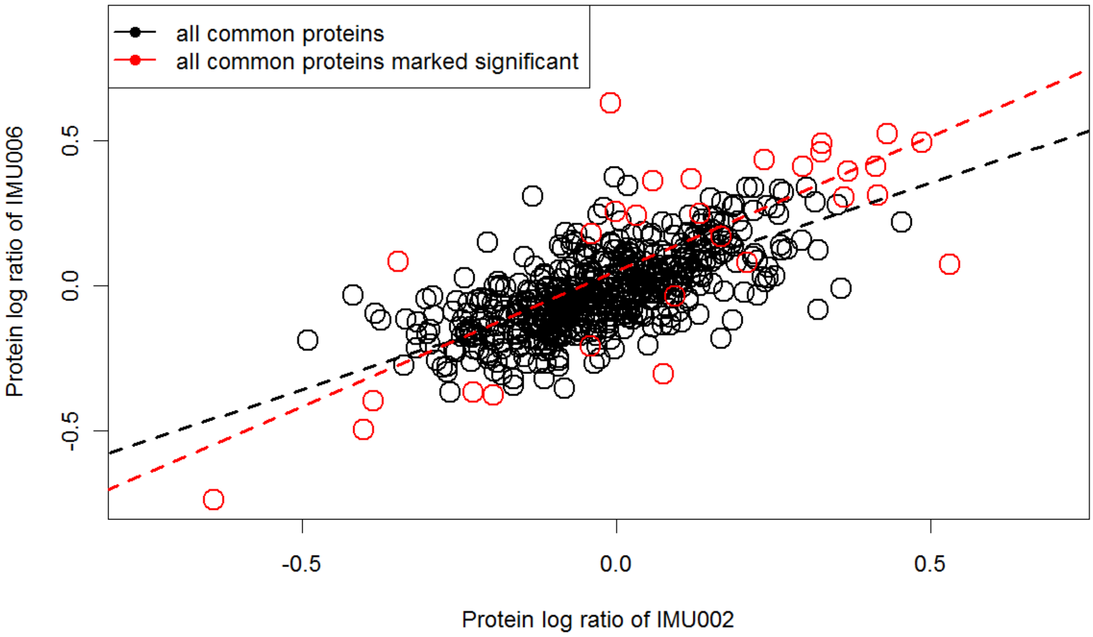


Figure 5.22: Comparison of protein expression ratios (C vs. H, day 14) between IMU002 and IMU006 (repetition of IMU002). The Pearson correlation of all common protein log ratios (black and red circles) is 0.69 and rises to 0.77 for proteins which have been deemed significantly expressed in either study (red circles only). A dashed regression line is shown for each case.

ConsensusPathDB [164] (release 22), an online tool which allows to identify enriched networks using a paired Wilcoxon signed-rank test for each pathway based on the provided expression values. Currently, ConsensusPathDB aggregates pathway information from twelve databases. We provide expression values for every observed protein in two different phenotypes, i.e., control versus high-dose samples at day 14. ConsensusPathDB returns a p-value for each functional set, based on the probability that the combined expression differences of genes in the functional set between the phenotypes have appeared by chance. Also, q-values (p-values corrected for multiple testing) are returned.

The q-values of IMU002 and IMU006 give a Pearson correlation of 0.79, whereas q-values of IMU006 and IMU007 have no correlation (0.03).

There is again a missing values problem since not all studies hit every pathway due to semi-random protein sampling. Only pathways with quantified proteins are shown in the result list of ConsensusPathDB for any p-value threshold. We use the median of all available q-values per pathway to determine a list of pathways with consistently low q-values. Note that the median is a very conservative statistic here since we allow almost half of the data to be outliers. The top result (lowest median q-value) is > 0.145 (activation of DNA fragmentation factor; three hits); thus, according to this method no pathway is consistently enriched across studies.

5.4 Discussion

We have described an approach to designing and evaluating iTRAQ-based proteomics experiments in the Predict-IV project. We successfully formulated an experimental design which allowed analysis of dose effects for all studies. The design is flexible enough for analysis in the time domain by using a pooled channel as reference between experiments. We developed software as part of OpenMS/TOPP to enable analysis of iTRAQ-labeled LC-MS data sets, and adapted and integrated the isobar package into a comprehensive TOPPAS-driven analysis pipeline which allows for highly automated analysis of large data sets.

We devised new measures of data quality for iTRAQ spectra, namely labeling efficiency and proportion of negative reporter ion counts after naive isotope correction. In general, a low reporter ion signal strength will decrease the number of confident peptide ratios used for protein ratio computation or will prevent quantification altogether. For certain data sets based on iTRAQ reporter strength alone, we can determine that the MS^2 search need not be carried out as we cannot expect to identify (or quantify) any peptides.

Isotope correction via NNLS was shown to be superior to the established naive inverse matrix multiplication and subsequent elimination of negative reporter ion intensities. NNLS avoids this ad hoc procedure, does not produce ghost peaks, and avoids extreme ratios as demonstrated for the naive approach.

We showed that normalization using median of ratios results in MA plots with no visible intensity bias. The normalization factors were usually between 0.6 and 1.6, but reached up to 2.64 in extreme cases, showing that normalization is absolutely critical.

iTRAQ is known to suffer in terms of accuracy because of ratio underestimation for complex samples due to precursor overlap, and precision issues due to variance heterogeneity, as low signal data have higher relative variability. A global correction for lack of accuracy as suggested by Karp et al. [136] seems only necessary when comparing proteomics results with other technologies, but is not necessary for biomarker discovery within a sample since the compression affects all ratios (this picture is somewhat simplistic as ratio compression depends on the actual amount of overlap for each precursor). For the first time, we have investigated the influence of ions which will be concurrently fragmented with the selected precursor peptide due to overlapping signals. We found that about 95% of all precursors are affected by overlapping signals, which contribute about 20% to the abundance within the isolation window. This again supports the theory of ratio underestimation by overlapping peptides.

Peptide ratios tagged as outlier are predominantly found in iTRAQ spectra with low reporter ion counts, which is not surprising since other studies have found variance in low-abundance spectra to be larger than in high-abundance spectra.

The extended exclusion list approach developed for this project and implemented into OpenMS can only be theoretically motivated as the data suggests that, while exclusion lists were successfully generated, the instrument software nevertheless allowed for precursor acquisition at the exact same position, indicating an instrument error or false parameter settings on the instrument.

The detection of common potential biomarkers indicative of toxic adverse effects across all studies was expected to be difficult due to limited and partly diverse proteome coverage of each study. As there is no significantly over-/underexpressed potential biomarker across all studies, a network approach was chosen to alleviate the missing values problem. Unfortunately, the method employed yielded no common enriched network across all studies. If such a panel of biomarkers is biologically feasible remains an open question.

Chapter 6

Summary and Future Directions

Synopsis: *We summarize the results of this thesis and provide an overview of possible future extensions.*

6.1 Novel Contributions of this Thesis

This thesis has made novel contributions in three areas, namely simulation, decharging, and iTRAQ-based differential protein expression.

We have presented MSSimulator [34], an extensive collection of algorithms and models for MS simulation supporting multiple levels of ground truth, which reduces the need for expensive manual validation on real data sets. Most notably, the simulator supports multiple models of chromatography (CE and HPLC), enzymatic digestion (regular expression and linear model), peak shapes (Gaussian and Lorentzian), label-free and labeled experiments (via a modular labeling framework), resolution (constant, linear, square root), amongst other features such as MS^2 and MS^E simulation. We illustrated that simulation is invaluable to assess algorithm performance (e.g., ETISEQ for MS^E data and map alignment using the OpenMS MapAligner). Furthermore, we used simulation in a pilot study as a means to optimize experimental settings, e.g., to answer the question of what is the gain in label-free feature identification, if resolution is doubled when measuring a complex mixture data set.

Our second contribution is the decharging [35] of peptide or protein charge ladders while allowing for multiplexing and different adduct compositions. The problem is modeled as an ILP and allows for a fast and robust solution. We demonstrated that decharging is useful for many applications in quantitative proteomics. The algorithm is not restricted to a specific instrument or resolution, and although it is intended for ESI data, it should also be applicable to MALDI data when multiply-charged ions are observed (e.g., for whole protein measurements [30]). The algorithm was optimized by splitting the ILP into subproblems, which, as a result, can be solved more efficiently. The split also allows for parallel processing on multiple CPU cores. Both measures reduce running time significantly such that for a complex data set with adducts the runtime is only a few minutes. Without adducts, the solution is usually obtained within seconds.

Decharging was able to improve mass precision on an 18 protein mix data (tryptic digest). On a whole protein measurement of hemoglobin, our algorithm showed better mass precision when compared to the Xtract software, which is shipped with the instrument. Decharging is also applicable to labeled data. Ambiguous pairs can be resolved by the additional information provided by charge ladders. Also, missing partners of singletons can be inferred, which is not possible with other pair finders. For the gliadin data set, our reanalysis revealed some misannotation, introduced by manual analysis. We also found more protein masses which are readily available in the data but were missed by manual annotation. Using our simulator, we could show that our decharging algorithm maintains high levels of recall and precision (above 75%) even when the input data is complex or contains many missing values (even beyond 50%).

Our third contribution is a novel, highly automatable analysis workflow in TOPPAS for iTRAQ-based proteomics experiments. The workflow was adapted to the experimental design which we proposed for the Predict-IV project. A new isotope correction procedure based on non-negative least squares was introduced. We showed that NNLS avoids ghost peaks and extreme ratios and is therefore superior to the commonly used inverse matrix multiplication with subsequent elimination of negative reporter ion intensities. Labeling efficiency was introduced as a new metric for iTRAQ data quality. The metric can be computed very fast, is easy to implement, and provides a powerful means to assess prospects for high-quality quantification and identification results. NNLS and the labeling efficiency metric are available in our ITRAQAnalyzer tool, which is part of OpenMS/TOPP. Normalization of iTRAQ channels using median of ratios was shown to be essential to avoid biased quantification results. Normalization factors usually ranged between 0.6 and 1.6, but reached up to 2.64 in extreme cases. The extended exclusion list approach that was developed for this project can only be theoretically motivated. Despite the fact that exclusion lists were successfully generated, the instrument software nevertheless allowed for precursor acquisition at the exact same position. For the first time, we have investigated the influence of background ions which will be concurrently fragmented with the selected precursor peptide. We found that about 95% of all precursors are affected by overlapping signals. The latter contribute about 20% to the abundance within the isolation window. This supports the theory of ratio underestimation by overlapping peptides as already reported in the literature [12, 83, 136].

All algorithms presented in this thesis were implemented in OpenMS/TOPP and are part of the current stable release of OpenMS 1.9.

6.2 Future Extensions

Even though all algorithms presented in this thesis have reached a mature state, have been published (or are being prepared for publication), and are part of OpenMS/TOPP, there are extensions to each algorithm, which can be beneficial to extend their applicability.

Our simulation software would benefit from an automatic estimation of simulation parameters (e.g., resolution, sampling rate, noise level) from real data, which is currently a manual process and thus time-consuming. Furthermore, the current model of proteotypicity is not quantitative. By using machine learning, it should be possible to create a quantitative model that could be easily integrated into the current framework. Current approaches (including MSSimulator) only classify peptides as either proteotypic or not proteotypic [165, 166]. Incorporation of ion statistics, detectors, and instrument specific properties (e.g., “shoulder peaks” on FT instruments – see Subsection 2.3.2) should also increase the level of realism. Due to the broad support of different levels of ground truth and a wide variety of models, the simulator could also be used to re-evaluate published algorithms whose performances were assessed using a feature-limited and special purpose simulation tool. A comparison might reveal significant differences in performance of the algorithm, pointing to violated model assumptions (e.g., shape models, data complexity). Last but not least, to estimate the performance of peak picking or feature finding algorithms on existing data, one could embed simulated signals into real data and use the recovered proportion as a proxy for sensitivity.

Decharging has been shown to be robust towards high-complex data sets and missing values up to 50% and even beyond. If the amount of missing features is higher, missing features could be attenuated by coupling decharging to a hypothesis-driven feature finding heuristic (already implemented in OpenMS) which searches for a strong signal (e.g., using signal-to-noise ratio) at putative feature positions to infer missing features or resolve ambiguous explanations. Furthermore, automatic estimation of putative adducts from the data would remove the need for their manual specification. This should be possible by searching for edges with a restricted set of adducts and selecting those which significantly increase the number of edges.

iTRAQ-based quantification can be re-evaluated on the biological modeling level. Currently, only proteins with fold changes in the tails of the Cauchy distribution are tagged as significant, which is the default behavior of the isobar package. This is a conservative measure in the sense that a lower fold change can be very stable and consistent across conditions but would never be tagged as significant, which will decrease the number of candidate biomarkers in exploratory studies. Furthermore, in order to avoid extreme normalization ratios in iTRAQ-based quantification it should be possible to measure an aliquot of the sample in the wet lab to determine a preliminary normalization ratio. On strong deviation from a ratio of one the mixture can be adjusted before a full measurement is performed. Also, inclusion lists are a reliable method to increase protein overlap across studies, but they require additional effort during acquisition. Increased overlap would also aid in establishing ratio comparisons across iTRAQ experiments via a pooled reference (which already exists), thus allowing to track not only changes with dosage but also with time more reliably.

To sum up, we have introduced a comprehensive simulation framework and have shown its applicability to algorithm benchmarking and validation. Our decharging algorithm is capable of handling an arbitrary set of adducts, is suitable for peptide- as well as protein data sets, and can handle labeled data. It achieves mass precision that is superior to other decharging algorithms. Its robustness towards missing values and complex data sets was validated using simulation. Last but not least, our iTRAQ analysis pipeline allows for a fast computation of lists of differentially expressed proteins. The pipeline incorporates a non-negative least squares procedure for isotope correction which outperforms inverse matrix multiplication. We have introduced labeling efficiency as a new measure for iTRAQ data quality and also found supporting evidence for iTRAQ ratio underestimation by overlapping signals.

Appendix

6.3 Availability and Implementation

The TOPP tools and UTILS developed as part of this thesis are available free of charge in the current release of OpenMS/TOPP, available from http://www.OpenMS.de, running on all major operating systems (Windows, Linux, MacOSX). The classes on which the TOPP tools are based are implemented in the OpenMS library and have accompanying regression tests.

6.4 Simulation: Capillary Electrophoresis

6.4.1 Explanations for Choices of α

- $\alpha = \frac{1}{3}$: Based on Strokes' law for frictional drag in non-conducting media. The model predicts the mobility to be proportional to the radius of a sphere (volume of sphere: $\frac{4}{3} \cdot pi \cdot r^3$). As volume is proportional to mass (MW) and a peptide's shape can arguably be approximated as a sphere, the radius of the sphere is proportional to the cube root of the peptide's mass.

- $\alpha = \frac{1}{2}$: Classical polymer model. Assuming that frictional drag is proportional to the average radius of gyration, it has been shown (for synthetic polymers) that mobility is proportional to the square root of the number of polymers times the length of a unit [167]. This translates to the number of AA, which is approximately the peptide mass.

- $\alpha = \frac{2}{3}$: Offord model [168], which is similar to Strokes' law, but assumes the mobility of an ion in conducting media to be inversely proportional to the surface of of a sphere which is proportional to the square of the radius ($surface = 4 \cdot pi \cdot r^2$).

However, all these models are only approximations and neglect certain properties of CE and the analytes (e.g., peptide shapes, electrostatical interactions with separation buffer and other peptides). See Rickard [169] for a more elaborate discussion. Authors report different empirical values of α, e.g., 0.52 [44] or 0.46 [170] to produce the best correlation with observed migration times; others use the theoretical values $\alpha = \frac{1}{2}$ or $\alpha = \frac{2}{2}$ [169]. As several different values for α have been found adequate, depending on the researched peptides, the software allows manual adjustment for α. The default, however, is set to $\frac{1}{2}$, which is the exact value for the classical polymer model and also close to the empirical findings by Williams, Russell, and Russell [44] and Kim, Zand, and Lubman [170].

6.4.2 Charge Determination

The net charge value q for a peptide depends on the buffer pH and its amino acid composition. Depending on the pH and other factors, an amino acid will become positively or negatively charged (or might even be neutral) if its pI equals the buffer pH. According to Benavente [171], q can be estimated reliably from average AA pK_a values from Rickard [169]. This leads us to employ the widely used Rickard's pK_a values and the charge equation given by Winzor [172] and Benavente [171]:

$$q = \sum_{n=1}^{4} \frac{P_n}{1 + 10^{pH - pK(P_n)}} - \sum_{n=1}^{5} \frac{N_n}{1 + 10^{pK(N_n) - pH}}, \tag{6.1}$$

where P_n are the number of basic residues (P_1 = terminal $NH_2 = 1$; $P_2 = \#$His; $P_3 = \#$Arg; $P_4 = \#$Lys) and N_n are the number of acidic residues (N_1 = terminal COOH = 1; $N_2 = \#$Asp; $N_3 = \#$Glu; $N_4 = \#$Cys; $N_5 = \#$Tyr).

The equation allows the user to provide a pH value (which is typically rather low at about pH 2-3). The model assumes that each amino acid's charge is independent of the other groups in the peptide. However, real pK_a values are additionally governed by peptide bonds and ionizing groups in the neighborhood, and are further shifted due to secondary and tertiary structural elements [169, 171]. Net-charge-based “trends” in CE-MS data have been observed, seemingly dependent on the number of lysine [44], arginine, and histidine [173].

6.4.3 Migration Time Computation

For practical purposes we allow an automatic scaling where the 95th MT percentile is projected to 95% of a given gradient time. The reason for not scaling the whole MT range to the given gradient is that outliers (with large MT) will otherwise compress the rest of the features into a small MT range.

6.5 Simulation: Contaminants Input Format

MSSimulator ships with a default text file containing a list of contaminations. The user can easily modify or extend this list. The file is in comma separated value format with the following columns:

- Name of the contaminant (this will be included into the generated featureXML file to help the user identify the simulated contaminant).
- Elemental composition of the contaminant (e.g. CH3OH for Methanol).
- Retention time at which the contaminant starts to elute.
- Retention time at which the contaminant stops to elute.
- The intensity of the contaminant.
- The charge of the contaminant.
- The shape of the elution profile of the contaminant. Valid choices are `gauss`, which gives a Gaussian-like elution profile, and `rec`, which gives a rectangular elution profile (immediate start and end of elution).

- The ion source which can ionize the contaminant. Valid choices are `ESI`, `MALDI`, and `ALL`. The contaminant will be visible if the respective ion source is used during simulation.

Listing 6.1 shows an example of a typical contaminants file.

Listing 6.1: Contaminants example (shortened).

```
”Methanol”, CH3OH, 1622.67007796, 1636.41953782, 9.2213957831, 1, rec, ESI
”ACN”, CH3CN, 124.649458627, 214.635926495, 2.16273553493, 1, gauss, ESI
”ACN”, CH3CN, 1898.59871684, 1907.43645208, 5.83229369713, 1, gauss, ESI
”PEG”, C2H4OH2O, 248.114284788, 408.870937779, 26.5073181419, 1, gauss, ESI
”ACN”, CH3CN, 1315.05880119, 1323.44605547, 3.58052185195, 1, gauss, ESI
”Methanol”, CH3OH, 335.244027089, 363.211414194, 13.7931809344, 1, gauss, ESI
”DMSO”, C2H6OS, 1033.09825892, 1047.95569948, 7.10154401272, 1, gauss, ESI
”PPG”, C3H6OH2O, 1416.76207731, 1424.20882302, 1.75685771789, 1, gauss, ESI
”Acetonitrile”, CH3CN, 1427.9342627, 1471.10917918, 3.03404557045, 1, rec, ESI
”PEG”, C2H4OH2O, 841.010029756, 844.145059911, 2.08297782253, 1, rec, ESI
”d6–DMSO”, C2(2)H6OS, 1040.65301474, 1050.06202825, 9.47611582171, 1, gauss, ESI
”Tween”, C22H42O6C2H4O, 1291.34392383, 1345.04027583, 5.6043075182, 1, rec, ALL
”Tween”, C24H44O6C2H4O, 1138.92462328, 1170.23236234, 11.4304013533, 1, gauss, ALL
”Tween”, C24H46O6C2H4O, 815.201065785, 906.178998869, 7.02337746262, 1, gauss, ALL
”Tween”, C22H42O6C2H4O, 592.070179209, 615.839682222, 8.38995498978, 1, rec, ALL
”Tween”, C24H44O6C2H4O, 420.416093615, 429.92926345, 8.65663328873, 1, gauss, ALL
”Tween”, C24H46O6C2H4O, 506.491681671, 590.456315632, 4.04299987225, 1, gauss, ALL
```

Glossary

Accuracy

Accuracy is the difference between the measured mass of an ion and its theoretical mass, typically given in parts per million (ppm), which can be computed as

$$Acc = \frac{m_{\text{measured}} - m_{\text{theoretical}}}{m_{\text{theoretical}}} \cdot 10^6 \, ppm.$$

Biomarker

A characteristic allowing to discern normal from pathogenic biological processes, or a pharmacological response to therapeutic treatment or an *in vivo* molecule whose altered abundance is connected to a specific condition of health [12].

Bottom-up Mass Spectrometry

Analysis of digested proteins, i.e., peptides. Requires a smaller m/z range and creates a narrow charge distribution. Protein quantification and identification values are inferred from the detected peptides. Not all peptides may be viable for quantification/identification of proteins if they are shared between multiple proteins (non-unique).

Capillary electrophoresis (CE)

The term CE refers to a family of separation techniques that use narrow-bore fused-silica capillaries to separate a complex mixture of large and small charged molecules. In a high electric field, molecules are separated based on their physical-chemical properties which determine their migration time, which is further dependent on the background electrolyte and its properties, e.g., ionic strength, pH, or type of ions.

Centroiding

See peak picking.

DDA

Data-dependent acquisition, also known as IDA (intensity-dependent acquisition) is an operation mode available in MS^2-capable mass spectrometers for selecting precursor ions for MS^2 fragmentation. The approach is untargeted (thus no prior knowledge of peptide content is required) and usually intensity dependent, i.e., the most intense precursor ions from a preceding survey scan will be selected as precursors.

Exclusion list

In subsequent LC-MS runs of the same (or similar) analytes with reproducible chromatography conditions, exclusion lists can be used to exclude certain precursor ions from being selected for MS^2 fragmentation in DDA mode. Usually, the list will be populated based on previous runs and contains positions in RT and m/z where a peptide has already been identified, thus avoiding redundant identification and allowing for a deeper proteome coverage.

FDA

The Food and Drug Administration (FDA) is an agency within the U.S. Department of Health and Human Services instantiated to protect public health and approve safe and effective medicines and drugs.

FWHM

Full width at half maximum (FWHM) is a metric used to describe the width of a peak, which is measured from its left to right flank at half the maximum peak intensity.

Ghost peak

Artefactual peak which is erroneously reconstructed from a zero-intensity peak by the naive inverse matrix multiplication procedure during isotope correction of raw iTRAQ reporter ions.

High-performance liquid chromatography (HPLC)

HPLC is a chromatographic technique to separate a (complex) mixture for eased downstream analysis. HPLC instrumentation includes a pump, injector, column, detector, and data system. The analyte mixture is forced through a stationary phase by the flow of a mobile phase at high pressure, separating the mixture into its components. The stationary phase is defined as the immobile packing material in the column, whereas the mobile phase is the solvent added to promote elution and whose composition can be changed in time to change the interaction of the solute with mobile and stationary phase.

Sacred Birman

Admirable cat breed with blue eyes, native to central Asia and German rental apartments. Most individuals prefer cozy roosts such as human interface devices (HID), keyboards in particular, especially when the aforementioned HID is currently in use.

iTRAQ

The isobaric tag for relative and absolute quantitation (iTRAQ) is an MS^2-based labeling technique allowing for multiplexing with up to eight isobaric tags. Reporter ions allowing to quantify individual sample contributions show in the MS^2 spectrum in the 113-121 Th range for the 8-plex kit, and 114-117 Th for the 4-plex kit.

isobar

The isobar package [144] is a statistical software package written in R [145] for the analysis of iTRAQ data. It features a flexible R interface, noise models to account for data heteroscedasticity, and statistical significance estimation for protein over- or underexpression, circumventing ad hoc thresholds for fold changes.

Labeling efficiency

Labeling efficiency denotes the relative fraction of iTRAQ spectra where a reporter ion signal is present in any channel and can be used as a criterion of data quality. Optionally, when stated explicitly, the term can also refer to the relative fraction of reporter ion presence in a single iTRAQ channel only.

Multiple reaction monitoring (MRM)

MRM, also called selected reaction monitoring (SRM), is a targeted quantification strategy which uses selected (and specific) precursor and product ions of a peptide (a so-called transition) as it elutes off the LC column. MRM works for highly complex samples and requires an instrument with MS^2 capabilities, usually a Q-TOF or a Triple-Q. MRM has the advantage of high sensitivity and dynamic range combined with reliable acquisition of the targeted species, but with a limited amount of peptides/proteins that can be quantified in one experiment.

Multiplexing

Multiplexing via labeling allows to discern identical peptides (in terms of sequence and post-translational modifications) from different samples within one LC-MS experiment by using chemical or metabolic labeling of some kind to introduce a systematic mass shift. This allows to discern and finally quantify peptides from different samples concurrently.

Non-negative Least Squares (NNLS)

Non-negative Least Squares solves the problem min $||Ax{-}b||_2$, subject to $x \geq 0$. The NNLS problem is solved iteratively, and it can be shown that the iteration always converges and terminates.

OpenMS

OpenMS is a C++-based open-source library for label-free and labeled quantification and identification, supporting all major platforms. It supports the HUPO-PSI standards mzML and mzIdentML as well as the widely used pepXML and protXML formats, enabling data exchange between collaborators based on open platform-independent formats. TOPP is a set of application based on OpenMS.

TOPP

The OpenMS Proteomics Pipeline (TOPP) [98] is a set of executables, chainable in modular fashion for a wide set of analysis scenarios and covers common tasks like peak picking, map alignment, identification (via wrappers for common identification engines like Mascot, X!Tandem and OMSSA), filtering, and quantification of labeled and label-free data.

Peak

A peak is the signal produced by a single ion species with a fixed number of neutrons and a fixed charge. The shape of a peak is usually Gaussian- or Lorentzian-like. Its width depends on the resolution of the mass spectrometer at the designated m/z position. The width is usually measured as FWHM.

Peak picking

One of the first data reduction steps for processing raw LC-MS data. Peak picking aims at representing a true peak by single datapoint (the centroid), describing its m/z (usually somewhere close to the center of the peak or its apex) and cumulative or apex intensity, while discarding single noise data points.

Precision

Precision is an intrinsic property of the instrument and describes the reproducibility of a repeated mass measurement as determined by its physical limits.

Precursor ion

In shotgun proteomics, a precursor ion (or parent ion) describes an unfragmented peptide species which, when selected for fragmentation, will give rise to multiple product ions which allow peptide identification.

Predict-IV

Predict-InVitro (Predict-IV) is a project funded by the European Union within its Seventh Framework Programme. The project aims at characterizing the dynamics and kinetics of cellular responses to toxic effects in vitro.

Product ion

See Precursor ion.

Reporter ion

An ion observed in MS^2 iTRAQ spectra at designated m/z positions (114, 115, 116, 117 Th for 4-plex iTRAQ and 113, 114, 115, 116, 117, 118, 119, 121 Th for 8-plex iTRAQ). We use the term raw abundance or raw intensity to denote reporter ion intensities as observed in the data, i.e., prior to isotope correction.

Resolution

Resolution is defined as $R = m/\Delta m_{50\%}$ where m is the mass to be measured and $\Delta m_{50\%}$ is the minimal distance to the next theoretical mass which can be resolved. The distance is defined in terms of the full width at half maximum (FWHM).

Study

In the realm of the Predict-IV project, a study designates a set of experiments involving exactly one cell type and one toxic compound, but with different dosages and collection time points. This results in a set of iTRAQ LC-MS data sets according to the experimental design.

Top-down Mass Spectrometry

Analysis of whole proteins, without prior digestion. Not feasible with all mass spectrometers. Allows easier localization of modification sites but usually creates large charge ladders and congested mass spectra. Depending on protein size, only high-resolution instruments are capable of resolving isotopic envelopes.

Index

Curriculum Vitae

For reasons of data protection, the Curriculum vitae is not published in the online version.

List of Publications

[P.1] Chris Bielow et al. “Optimal decharging and clustering of charge ladders generated in ESI-MS.” In: *Journal of Proteome Research* 9.5 (May 2010), pp. 2688–95.

[P.2] Chris Bielow et al. “Bioinformatics for qualitative and quantitative proteomics.” In: *Methods in Molecular Biology (Clifton, N.J.)* 719 (Jan. 2011), pp. 331–49.

[P.3] Chris Bielow et al. “MSSimulator: Simulation of mass spectrometry data.” In: *Journal of Proteome Research* 10.7 (July 2011), pp. 2922–9.

[P.4] Johannes Junker et al. “TOPPAS: A graphical workflow editor for the analysis of high-throughput proteomics data.” In: *Journal of Proteome Research* (May 2012), DOI: 10.1021/pr300187f.

[P.5] Anja Wilmes et al. “An integrated omics approach for the assessment of compound induced cell stress in cultured human renal proximal tubular cells”. In: *manuscript in preparation* (2012).

Bibliography

[1] Guy H. Fernald et al. "Bioinformatics challenges for personalized medicine." In: *Bioinformatics (Oxford, England)* 27.13 (July 2011), pp. 1741–8.

[2] Mingyao Li et al. "Widespread RNA and DNA sequence differences in the human transcriptome." In: *Science (New York, N.Y.)* 333.6038 (July 2011), pp. 53–8.

[3] Benjamin F. Cravatt, Gabriel M. Simon, and John R. Yates. "The biological impact of mass-spectrometry-based proteomics." In: *Nature* 450.7172 (Dec. 2007), pp. 991–1000.

[4] John B. Fenn et al. "Electrospray ionization for mass spectrometry of large biomolecules". In: *Science* 246.4926 (Oct. 1989), pp. 64–71.

[5] Jakob Albrethsen. "The first decade of MALDI protein profiling: A lesson in translational biomarker research." In: *Journal of Proteomics* 74.6 (May 2011), pp. 765–73.

[6] Marion Haubitz et al. "Identification and validation of urinary biomarkers for differential diagnosis and evaluation of therapeutic intervention in anti-neutrophil cytoplasmic antibody-associated vasculitis." In: *Molecular & Cellular Proteomics* 8.10 (Oct. 2009), pp. 2296–307.

[7] N. Leigh Anderson. "The clinical plasma proteome: a survey of clinical assays for proteins in plasma and serum." In: *Clinical Chemistry* 56.2 (Feb. 2010), pp. 177–85.

[8] Scott A. McLuckey and J. Mitchell Wells. "Mass Analysis at the Advent of the 21st Century". In: *Chemical Reviews* 101.2 (Feb. 2001), pp. 571–606.

[9] David L. Tabb et al. "Repeatability and reproducibility in proteomic identifications by liquid chromatography-tandem mass spectrometry." In: *Journal of Proteome Research* 9.2 (Feb. 2010), pp. 761–76.

[10] Ole N. Jensen. "Interpreting the protein language using proteomics." In: *Nature Reviews. Molecular Cell Biology* 7.6 (June 2006), pp. 391–403.

[11] Ron M. A. Heeren et al. "Imaging mass spectrometry: hype or hope?" In: *Journal of the American Society for Mass Spectrometry* 20.6 (June 2009), pp. 1006–14.

[12] Emily Boja et al. "Evolution of clinical proteomics and its role in medicine." In: *Journal of Proteome Research* 10.1 (Jan. 2011), pp. 66–84.

[13] Annemieke Kolkman et al. "Double standards in quantitative proteomics: direct comparative assessment of difference in gel electrophoresis and metabolic stable isotope labeling." In: *Molecular & Cellular Proteomics* 4.3 (Mar. 2005), pp. 255–66.

[14] Edgar Nägele et al. "2D-LC/MS techniques for the identification of proteins in highly complex mixtures." In: *Expert Review of Proteomics* 1.1 (June 2004), pp. 37–46.

[15] Silvia Surinova et al. "On the Development of Plasma Protein Biomarkers." In: *Journal of Proteome Research* (Dec. 2010).

[16] Nuno Bandeira, Alexey Nesvizhskii, and Martin McIntosh. “Advancing next-generation proteomics through computational research.” In: *Journal of Proteome Research* 10.7 (July 2011), p. 2895.

[17] Juan Antonio Vizcaíno et al. “A guide to the Proteomics Identifications Database proteomics data repository.” In: *Proteomics* 9.18 (Sept. 2009), pp. 4276–83.

[18] Jason W. H. Wong, Alexander B. Schwahn, and Kevin M. Downard. “ETISEQ–an algorithm for automated elution time ion sequencing of concurrently fragmented peptides for mass spectrometry-based proteomics.” In: *BMC Bioinformatics* 10 (Jan. 2009), p. 244.

[19] Ludovic Gillet et al. “Swath MS: a novel data independent acquisition method with sequential precursor isolation windows allowing unlimited SRM-like data analysis and quantification”. In: *Proceedings of the 59th ASMS Conference on Mass Spectrometry and Allied Topics*. Denver, 2011.

[20] Jeffrey S. Morris et al. “Feature extraction and quantification for mass spectrometry in biomedical applications using the mean spectrum.” In: *Bioinformatics (Oxford, England)* 21.9 (May 2005), pp. 1764–75.

[21] John Klimek et al. “The standard protein mix database: a diverse data set to assist in the production of improved Peptide and protein identification software tools.” In: *Journal of Proteome Research* 7.1 (Jan. 2008), pp. 96–103.

[22] Eva Lange et al. “Critical assessment of alignment procedures for LC-MS proteomics and metabolomics measurements.” In: *BMC Bioinformatics* 9 (Jan. 2008), p. 375.

[23] Bernhard Y. Renard et al. “NITPICK: peak identification for mass spectrometry data.” In: *BMC Bioinformatics* 9 (Jan. 2008), p. 355.

[24] Chao Yang, Zengyou He, and Weichuan Yu. “Comparison of public peak detection algorithms for MALDI mass spectrometry data analysis.” In: *BMC Bioinformatics* 10 (Jan. 2009), p. 4.

[25] Kang Ning, Damian Fermin, and Alexey I. Nesvizhskii. “Comparative Analysis of Different Label-Free Mass Spectrometry Based Protein Abundance Estimates and Their Correlation with RNA-Seq Gene Expression Data.” In: *Journal of Proteome Research* (Feb. 2012).

[26] Ole Schulz-Trieglaff et al. “LC-MSsim–a simulation software for liquid chromatography mass spectrometry data.” In: *BMC Bioinformatics* 9 (Jan. 2008), p. 423.

[27] Philip L. Ross et al. “Multiplexed protein quantitation in Saccharomyces cerevisiae using amine-reactive isobaric tagging reagents.” In: *Molecular & Cellular Proteomics* 3.12 (Dec. 2004), pp. 1154–69.

[28] Ian P. Shadforth et al. “i-Tracker: for quantitative proteomics using iTRAQ.” In: *BMC Genomics* 6 (Jan. 2005), p. 145.

[29] Marc Sturm et al. “OpenMS - an open-source software framework for mass spectrometry.” In: *BMC Bioinformatics* 9 (Jan. 2008), p. 163.

[30] Dariya I. Malyarenko et al. “Automated assignment of ionization states in broad-mass matrix-assisted laser desorption/ionization spectra of protein mixtures.” In: *Rapid Communications in Mass Spectrometry* 24.1 (Jan. 2010), pp. 138–46.

[31] Matthias Mann, Chin Kai Meng, and John B. Fenn. “Interpreting mass spectra of multiply charged ions”. In: *Analytical Chemistry* 61.15 (Aug. 1989), pp. 1702–1708.

[32] Z. Zhang and A. G. Marshall. “A universal algorithm for fast and automated charge state deconvolution of electrospray mass-to-charge ratio spectra.” In: *Journal of the American Society for Mass Spectrometry* 9.3 (Mar. 1998), pp. 225–33.

[33] Marco Wehofsky and Ralf Hoffmann. “Automated deconvolution and deisotoping of electrospray mass spectra.” In: *Journal of Mass Spectrometry* 37.2 (Feb. 2002), pp. 223–9.

[34] Chris Bielow et al. “MSSimulator: Simulation of mass spectrometry data.” In: *Journal of Proteome Research* 10.7 (July 2011), pp. 2922–9.

[35] Chris Bielow et al. “Optimal decharging and clustering of charge ladders generated in ESI-MS.” In: *Journal of Proteome Research* 9.5 (May 2010), pp. 2688–95.

[36] Anja Wilmes et al. “An integrated omics approach for the assessment of compound induced cell stress in cultured human renal proximal tubular cells”. In: *manuscript in preparation* (2012).

[37] Brittan N. Clark and Howard B. Gutstein. “The myth of automated, high-throughput two-dimensional gel analysis.” In: *Proteomics* 8.6 (Mar. 2008), pp. 1197–203.

[38] Steven P. Gygi et al. “Evaluation of two-dimensional gel electrophoresis-based proteome analysis technology.” In: *Proceedings of the National Academy of Sciences of the United States of America* 97.17 (Aug. 2000), pp. 9390–5.

[39] Yong Yang et al. “Evaluation of different multidimensional LC-MS/MS pipelines for iTRAQ-based proteomic analysis of potato tubers in response to cold storage.” In: *Journal of Proteome Research* (Aug. 2011).

[40] Yufeng Shen and Richard D. Smith. “Proteomics based on high-efficiency capillary separations.” In: *Electrophoresis* 23.18 (Sept. 2002), pp. 3106–24.

[41] Javier Hernández-Borges et al. “On-line capillary electrophoresis-mass spectrometry for the analysis of biomolecules.” In: *Electrophoresis* 25.14 (July 2004), pp. 2257–81.

[42] Walter Kolch et al. “Capillary electrophoresis-mass spectrometry as a powerful tool in clinical diagnosis and biomarker discovery.” In: *Mass Spectrometry Reviews* 24.6 (), pp. 959–77.

[43] Dan Theodorescu et al. “Pilot study of capillary electrophoresis coupled to mass spectrometry as a tool to define potential prostate cancer biomarkers in urine.” In: *Electrophoresis* 26.14 (July 2005), pp. 2797–808.

[44] Brad J. Williams, William K. Russell, and David H. Russell. “Utility of CE-MS data in protein identification.” In: *Analytical Chemistry* 79.10 (May 2007), pp. 3850–5.

[45] Ju Yang, Sahana Bose, and David S. Hage. “Improved reproducibility in capillary electrophoresis through the use of mobility and migration time ratios”. In: *Journal of Chromatography A* 735.1-2 (May 1996), pp. 209–220.

[46] Bezhan Chankvetadze. *Capillary Electrophoresis in Chiral Analysis.* John Wiley and Sons, 1997.

[47] Václav Kašička. “Recent advances in capillary electrophoresis of peptides.” In: *Electrophoresis* 22.19 (Nov. 2001), pp. 4139–62.

[48] Zak K. Shihabi. “Effect of sample composition on electrophoretic migration application to hemoglobin analysis by capillary electrophoresis and agarose electrophoresis.” In: *Journal of Chromatography A* 1027.1-2 (Mar. 2004), pp. 179–84.

[49] Nico Pfeifer et al. “Statistical learning of peptide retention behavior in chromatographic separations: a new kernel-based approach for computational proteomics.” In: *BMC Bioinformatics* 8 (2007), p. 468.

[50] Luminita Moruz, Daniela Tomazela, and Lukas Käll. “Training, selection, and robust calibration of retention time models for targeted proteomics.” In: *Journal of Proteome Research* 9.10 (Oct. 2010), pp. 5209–16.

[51] Vathany Kulasingam and Eleftherios P. Diamandis. “Tissue culture-based breast cancer biomarker discovery platform.” In: *International Journal of Cancer* 123.9 (Nov. 2008), pp. 2007–12.

[52] Edmond de Hoffmann and Vincent Stroobant. *Mass Spectrometry: Principles and Applications.* 2001, p. 420.

[53] Matthias Mann and Neil L. Kelleher. “Precision proteomics: the case for high resolution and high mass accuracy.” In: *Proceedings of the National Academy of Sciences of the United States of America* 105.47 (Nov. 2008), pp. 18132–8.

[54] Merit Oss et al. “Electrospray ionization efficiency scale of organic compounds.” In: *Analytical Chemistry* 82.7 (Apr. 2010), pp. 2865–72.

[55] Michael A. Kuzyk et al. “A comparison of MS/MS-based, stable-isotope-labeled, quantitation performance on ESI-quadrupole TOF and MALDI-TOF/TOF mass spectrometers.” In: *Proteomics* 9.12 (June 2009), pp. 3328–3340.

[56] Christine Luebbert, Christian Ziegmann, and Martin Schuerenberg. “Long-term Archiving of Proteomics Samples on Disposable Prespotted AnchorChip MALDI Targets”. In: HUPO 4th Annual World Congress, 2005, TP135.

[57] Hsuan-shen Chen et al. “Enhanced characterization of complex proteomic samples using LC-MALDI MS/MS: exclusion of redundant peptides from MS/MS analysis in replicate runs.” In: *Analytical Chemistry* 77.23 (Dec. 2005), pp. 7816–25.

[58] Matthias Wilm. “Principles of electrospray ionization.” In: *Molecular & Cellular Proteomics* 10.7 (July 2011), p. M111.009407.

[59] Victor J. Nesatyy et al. “On the acquisition of +1 charge states during high-throughput proteomics: Implications on reproducibility, number and confidence of protein identifications.” In: *Journal of Proteomics* 72.5 (July 2009), pp. 761–70.

[60] Halan Prakash and Shyamalava Mazumdar. “Direct correlation of the crystal structure of proteins with the maximum positive and negative charge states of gaseous protein ions produced by electrospray ionization.” In: *Journal of the American Society for Mass Spectrometry* 16.9 (Sept. 2005), pp. 1409–21.

[61] Michal Svoboda et al. “The influence of strongly acidic groups on the protonation of peptides in electrospray MS”. In: *Journal of Mass Spectrometry* 32.10 (Nov. 1997), pp. 1117–1123.

[62] Casey J. Krusemark et al. “Modifying the charge state distribution of proteins in electrospray ionization mass spectrometry by chemical derivatization.” In: *Journal of the American Society for Mass Spectrometry* 20.9 (Sept. 2009), pp. 1617–25.

[63] Anthony T. Iavarone, John C. Jurchen, and Evan R. Williams. “Supercharged Protein and Peptide Ions Formed by Electrospray Ionization”. In: *Analytical Chemistry* 73.7 (Apr. 2001), pp. 1455–1460.

[64] M. Isabel Catalina et al. “Decharging of globular proteins and protein complexes in electrospray.” In: *Chemistry (Weinheim an der Bergstrasse, Germany)* 11.3 (Jan. 2005), pp. 960–8.

[65] Justin L. P. Benesch and Carol V. Robinson. “Mass spectrometry of macromolecular assemblies: preservation and dissociation.” In: *Current Opinion in Structural Biology* 16.2 (Apr. 2006), pp. 245–51.

[66] Harry J. Sterling and Evan R. Williams. “Origin of supercharging in electrospray ionization of noncovalent complexes from aqueous solution.” In: *Journal of the American Society for Mass Spectrometry* 20.10 (Oct. 2009), pp. 1933–43.

[67] Igor A. Kaltashov and Rinat R. Abzalimov. “Do ionic charges in ESI MS provide useful information on macromolecular structure?” In: *Journal of the American Society for Mass Spectrometry* 19.9 (Sept. 2008), pp. 1239–46.

[68] Tomás Pluskal et al. “MZmine 2: modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data.” In: *BMC Bioinformatics* 11 (Jan. 2010), p. 395.

[69] Eva Lange et al. “High-accuracy peak picking of proteomics data using wavelet techniques.” In: *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* (Jan. 2006), pp. 243–54.

[70] Alexander Makarov et al. “Performance evaluation of a hybrid linear ion trap/orbitrap mass spectrometer.” In: *Analytical Chemistry* 78.7 (Apr. 2006), pp. 2113–20.

[71] Michael W. Senko, Steven C. Beu, and Fred W. McLaffertycor. “Determination of monoisotopic masses and ion populations for large biomolecules from resolved isotopic distributions”. In: *Journal of the American Society for Mass Spectrometry* 6.4 (Apr. 1995), pp. 229–233.

[72] Dirk Valkenborg et al. “The isotopic distribution conundrum.” In: *Mass Spectrometry Reviews* 31.1 (May 2011), pp. 96–109.

[73] Long Li et al. “Memory-efficient calculation of the isotopic mass states of a molecule.” In: *Rapid Communications in Mass Spectrometry* 24.18 (Sept. 2010), pp. 2689–96.

[74] Julia Maria Burkhart et al. “Systematic and quantitative comparison of digest efficiency and specificity reveals the impact of trypsin quality on MS-based proteomics.” In: *Journal of Proteomics* (Nov. 2011).

[75] Yasset Perez-Riverol et al. “In silico analysis of accurate proteomics, complemented by selective isolation of peptides.” In: *Journal of Proteomics* (May 2011).

[76] David N. Perkins et al. “Probability-based protein identification by searching sequence databases using mass spectrometry data”. In: *Electrophoresis* 20.18 (1999), pp. 3551–3567.

[77] Jimmy K. Eng, Ashley L. McCormack, and John R. Yates. “An approach to correlate tandem mass spectral data of peptides with amino acid sequences in a protein database”. In: *Journal of the American Society for Mass Spectrometry* 5.11 (Nov. 1994), pp. 976–989.

[78] Lewis Y. Geer et al. “Open mass spectrometry search algorithm.” In: *Journal of Proteome Research* 3.5 (2004), pp. 958–64.

[79] Robertson Craig and Ronald C. Beavis. “A method for reducing the time required to match protein sequences with tandem mass spectra.” In: *Rapid Communications in Mass Spectrometry* 17.20 (Jan. 2003), pp. 2310–6.

[80] Ari Frank and Pavel Pevzner. “PepNovo: de novo peptide sequencing via probabilistic network modeling.” In: *Analytical Chemistry* 77.4 (Feb. 2005), pp. 964–73.

[81] Sandro Andreotti, Gunnar W. Klau, and Knut Reinert. “Antilope - A Lagrangian Relaxation Approach to the de novo Peptide Sequencing Problem.” In: *IEEE/ACM Transactions on Computational Biology and Bioinformatics / IEEE, ACM* (Mar. 2011).

[82] Sven Nahnsen et al. “Probabilistic Consensus Scoring Improves Tandem Mass Spectrometry Peptide Identification.” In: *Journal of Proteome Research* (June 2011).

[83] Marcus Bantscheff et al. “Robust and sensitive iTRAQ quantification on an LTQ Orbitrap mass spectrometer.” In: *Molecular & Cellular Proteomics* 7.9 (Sept. 2008), pp. 1702–13.

[84] Andrew Thompson et al. “Tandem mass tags: a novel quantification strategy for comparative analysis of complex protein mixtures by MS/MS.” In: *Analytical Chemistry* 75.8 (Apr. 2003), pp. 1895–904.

[85] Dennis Van Hoof et al. “An experimental correction for arginine-to-proline conversion artifacts in SILAC-based quantitative proteomics.” In: *Nature Methods* 4.9 (Sept. 2007), pp. 677–8.

[86] Amol Prakash et al. “Assessing bias in experiment design for large scale mass spectrometry-based quantitative proteomics.” In: *Molecular & Cellular Proteomics* 6.10 (Oct. 2007), pp. 1741–8.

[87] Josselin Noirel et al. “Methods in Quantitative Proteomics: Setting iTRAQ on the Right Track”. In: *Current Proteomics* 8.1 (Apr. 2011), pp. 17–30.

[88] Yasushi Ishihama et al. “Exponentially modified protein abundance index (emPAI) for estimation of absolute protein amount in proteomics by the number of sequenced peptides per protein.” In: *Molecular & Cellular Proteomics* 4.9 (2005), pp. 1265–72.

[89] Niklaas Colaert, Kris Gevaert, and Lennart Martens. “RIBAR and xRIBAR: Methods for reproducible relative MS/MS-based label-free protein quantification.” In: *Journal of Proteome Research* 10.7 (July 2011), pp. 3183–9.

[90] Thomas Köcher et al. “High precision quantitative proteomics using iTRAQ on an LTQ Orbitrap: a new mass spectrometric method combining the benefits of all.” In: *Journal of Proteome Research* 8.10 (Oct. 2009), pp. 4743–52.

[91] Alexander W. Bell et al. “A HUPO test sample study reveals common problems in mass spectrometry-based proteomics.” In: *Nature Methods* 6.6 (June 2009), pp. 423–30.

[92] Andrew Keller et al. “A uniform proteomics MS/MS analysis platform utilizing open XML file formats.” In: *Molecular Systems Biology* 1 (Jan. 2005), p. 2005.0017.

[93] Jürgen Cox and Matthias Mann. “MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification.” In: *Nature Biotechnology* 26.12 (Dec. 2008), pp. 1367–72.

[94] Mikko Katajamaa, Jarkko Miettinen, and Matej Oresic. “MZmine: toolbox for processing and visualization of mass spectrometry based molecular profile data.” In: *Bioinformatics (Oxford, England)* 22.5 (2006), pp. 634–6.

[95] Matthew E. Monroe et al. "VIPER: an advanced software package to support high-throughput LC-MS peptide identification." In: *Bioinformatics (Oxford, England)* 23.15 (2007), pp. 2021–3.

[96] Darren Kessner et al. "ProteoWizard: open source software for rapid proteomics tools development." In: *Bioinformatics (Oxford, England)* 24.21 (Nov. 2008), pp. 2534–6.

[97] Matthew Bellew et al. "A suite of algorithms for the comprehensive analysis of complex protein mixtures using high-resolution LC-MS." In: *Bioinformatics (Oxford, England)* 22.15 (2006), pp. 1902–9.

[98] Oliver Kohlbacher et al. "TOPP–the OpenMS proteomics pipeline." In: *Bioinformatics (Oxford, England)* 23.2 (2007), e191–7.

[99] Marc Sturm. "OpenMS - A framework for computational mass spectrometry". PhD thesis. Wilhelmstr. 32, 72074 Tübingen: Universität Tübingen, 2010.

[100] Marc Sturm and Oliver Kohlbacher. "TOPPView: an open-source viewer for mass spectrometry data." In: *Journal of Proteome Research* 8.7 (July 2009), pp. 3760–3.

[101] Johannes Junker et al. "TOPPAS: A graphical workflow editor for the analysis of high-throughput proteomics data." In: *Journal of Proteome Research* (May 2012), DOI: 10.1021/pr300187f.

[102] Ken Martin and Bill Hoffman. "An Open Source Approach to Developing Software in a Small Organization". In: *IEEE Software* 24.1 (Jan. 2007), pp. 46–53.

[103] Michael R. Berthold et al. "KNIME - the Konstanz information miner". In: *ACM SIGKDD Explorations Newsletter* 11.1 (Nov. 2009), p. 26.

[104] David A. Dahl. "SIMION for the personal computer in reflection". In: *International Journal of Mass Spectrometry* 200.1-3 (Dec. 2000), pp. 3–25.

[105] Kevin R. Coombes et al. "Understanding the characteristics of mass spectrometry data through the use of simulation." In: *Cancer informatics* 1 (Jan. 2005), pp. 41–52.

[106] Andreas Ipsen and Timothy M. D. Ebbels. "Prospects for a statistical theory of LC/TOFMS data." In: *Journal of the American Society for Mass Spectrometry* 23.5 (May 2012), pp. 779–91.

[107] David M. Creasy and John S. Cottrell. "Unimod: Protein modifications for mass spectrometry." In: *Proteomics* 4.6 (June 2004), pp. 1534–6.

[108] Jennifer A. Siepen et al. "Prediction of missed cleavage sites in tryptic peptides aids protein identification in proteomics." In: *Journal of Proteome Research* 6.1 (Jan. 2007), pp. 399–408.

[109] G. M. McLaughlin et al. "Pharmaceutical Drug Separations by HPCE: Practical Guidelines". In: *Journal of Liquid Chromatography* 15.6-7 (Apr. 1992), pp. 961–1021.

[110] John R. Conder. "Peak distortion in chromatography. Part 1: Concentration-dependent behavior". In: *Journal of High Resolution Chromatography* 5.7 (July 1982), pp. 341–348.

[111] Kevin Lan and James W. Jorgenson. "A hybrid of exponential and gaussian functions as a simple model of asymmetric chromatographic peaks". In: *Journal of Chromatography A* 915.1-2 (2001), pp. 1–13.

[112] Hugo Kubinyi. "Calculation of isotope distributions in mass spectrometry. A trivial solution for a non-trivial problem". In: *Analytica Chimica Acta* 247.1 (1991), pp. 107–119.

[113] Rune Matthiesen, ed. *Mass Spectrometry Data Analysis in Proteomics (Methods in Molecular Biology).* Humana Press, 2007, p. 336.

[114] Mark Wrona et al. "'All-in-one' analysis for metabolite identification using liquid chromatography/hybrid quadrupole time-of-flight mass spectrometry with collision energy switching." In: *Rapid Communications in Mass Spectrometry* 19.18 (Jan. 2005), pp. 2597–602.

[115] Shao-En Ong et al. "Stable Isotope Labeling by Amino Acids in Cell Culture, SILAC, as a Simple and Accurate Approach to Expression Proteomics". In: *Molecular & Cellular Proteomics* 1.5 (May 2002), pp. 376–386.

[116] Olga A. Mirgorodskaya et al. "Quantitation of peptides and proteins by matrix-assisted laser desorption/ionization mass spectrometry using (18)O-labeled internal standards." In: *Rapid Communications in Mass Spectrometry* 14.14 (Jan. 2000), pp. 1226–32.

[117] Alexander Schmidt, Josef Kellermann, and Friedrich Lottspeich. "A novel strategy for quantitative proteomics using isotope-coded protein labels." In: *Proteomics* 5.1 (Jan. 2005), pp. 4–15.

[118] Antonio Ramos-Fernández, Daniel López-Ferrer, and Jesús Vázquez. "Improved method for differential expression proteomics using trypsin-catalyzed 18O labeling with a correction for labeling efficiency." In: *Molecular & Cellular Proteomics* 6.7 (July 2007), pp. 1274–86.

[119] Lennart Martens et al. "mzML - a Community Standard for Mass Spectrometry Data." In: *Molecular & Cellular Proteomics* (Aug. 2010), R110.000133–.

[120] Peicheng Du and Ruth H. Angeletti. "Automatic deconvolution of isotope-resolved mass spectra using variable selection and quantized peptide mass distribution." In: *Analytical Chemistry* 78.10 (May 2006), pp. 3385–92.

[121] David M. Horn, Roman A. Zubarev, and Fred W. McLafferty. "Automated reduction and interpretation of". In: *Journal of the American Society for Mass Spectrometry* 11.4 (Apr. 2000), pp. 320–332.

[122] Bruce B. Reinhold and Vernon N. Reinhold. "Electrospray ionization mass spectrometry: Deconvolution by an Entropy-Based algorithm". In: *Journal of the American Society for Mass Spectrometry* 3.3 (Mar. 1992), pp. 207–215.

[123] Huiru Zheng et al. "Heuristic charge assignment for deconvolution of electrospray ionization mass spectra." In: *Rapid Communications in Mass Spectrometry* 17.5 (Jan. 2003), pp. 429–36.

[124] Xiao-Jun Li et al. "Automated statistical analysis of protein abundance ratios from data generated by stable-isotope dilution and tandem mass spectrometry." In: *Analytical Chemistry* 75.23 (Dec. 2003), pp. 6648–57.

[125] Stefan Wittke, Thorsten Kaiser, and Harald Mischak. "Differential polypeptide display: the search for the elusive target." In: *Journal of chromatography. B, Analytical technologies in the biomedical and life sciences* 803.1 (Apr. 2004), pp. 17–26.

[126] John Draper et al. "Metabolite signal identification in accurate mass metabolomics data with MZedDB, an interactive m/z annotation tool utilising predicted ionisation behaviour 'rules'." In: *BMC Bioinformatics* 10.1 (Jan. 2009), p. 227.

[127] Ralf Tautenhahn. “Bioinformatics Research and Development”. In: *Annotation of LC/ESI-MS Mass Signals*. Ed. by S. Hochreiter and R. Wagner. Vol. 4414. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 371–380.

[128] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, 1997, p. 608.

[129] George B. Dantzig. “Programming of Interdependent Activities. II. Mathematical Model.” In: *Econometrica* 17 (1949), pp. 200–211.

[130] Bernd O. Keller et al. “Interferences and contaminants encountered in modern mass spectrometry.” In: *Analytica Chimica Acta* 627.1 (Oct. 2008), pp. 71–81.

[131] Leonardo Dagum and Ramesh Menon. “OpenMP: an industry standard API for shared-memory programming”. In: *IEEE Computational Science and Engineering* 5.1 (1998), pp. 46–55.

[132] Claudia Lemmel et al. “Differential quantitative analysis of MHC ligands by mass spectrometry using stable isotope labeling.” In: *Nature Biotechnology* 22.4 (Apr. 2004), pp. 450–4.

[133] Michael R. Hoopmann, Gregory L. Finney, and Michael J. MacCoss. “High-speed data reduction, feature detection, and MS/MS spectrum quality assessment of shotgun proteomics data sets using high-resolution mass spectrometry.” In: *Analytical Chemistry* 79.15 (Aug. 2007), pp. 5620–32.

[134] Jens Mohr et al. “High-efficiency nano- and micro-HPLC–high-resolution Orbitrap-MS platform for top-down proteomics.” In: *Proteomics* 10.20 (Oct. 2010), pp. 3598–609.

[135] Salvatore Cappadona et al. “Deconvolution of overlapping isotopic clusters improves quantification of stable isotope-labeled peptides.” In: *Journal of Proteomics* 74.10 (Sept. 2011), pp. 2204–9.

[136] Natasha A. Karp et al. “Addressing accuracy and precision issues in iTRAQ quantitation.” In: *Molecular & Cellular Proteomics* 9.9 (Sept. 2010), pp. 1885–97.

[137] Ann L. Oberg and Olga Vitek. “Statistical design of quantitative mass spectrometry-based proteomic experiments.” In: *Journal of Proteome Research* 8.5 (2009), pp. 2144–56.

[138] Emanuel F. Petricoin et al. “Use of proteomic patterns in serum to identify ovarian cancer.” In: *Lancet* 359.9306 (Feb. 2002), pp. 572–7.

[139] Keith A. Baggerly, Jeffrey S. Morris, and Kevin R. Coombes. “Reproducibility of SELDI-TOF protein patterns in serum: comparing datasets from different experiments.” In: *Bioinformatics (Oxford, England)* 20.5 (Mar. 2004), pp. 777–85.

[140] Ann L. Oberg et al. “Statistical analysis of relative labeled mass spectrometry data from complex samples using ANOVA.” In: *Journal of Proteome Research* 7.1 (Jan. 2008), pp. 225–33.

[141] Dipanjana Ghosh et al. “Identification of key players for colorectal cancer metastasis by iTRAQ quantitative proteomics profiling of isogenic SW480 and SW620 cell lines.” In: *Journal of Proteome Research* (Aug. 2011).

[142] John H. Schwacke et al. “iQuantitator: a tool for protein expression inference using iTRAQ.” In: *BMC Bioinformatics* 10 (Jan. 2009), p. 342.

[143] Wen-Ting Lin et al. “Multi-Q: a fully automated tool for multiplexed protein quantitation.” In: *Journal of Proteome Research* 5.9 (Sept. 2006), pp. 2328–38.

[144] Florian P. Breitwieser et al. “General statistical modeling of data from protein relative expression isobaric tags.” In: *Journal of Proteome Research* 10.6 (June 2011), pp. 2758–66.

[145] R Development Core Team. “R: A Language and Environment for Statistical Computing”. In: *Vienna Austria R Foundation for Statistical Computing* 1.09/18/2009 (2008), ISBN 3–900051–07–0.

[146] Saw Yen Ow et al. “Quantitative shotgun proteomics of enriched heterocysts from Nostoc sp. PCC 7120 using 8-plex isobaric peptide tags.” In: *Journal of Proteome Research* 7.4 (Apr. 2008), pp. 1615–28.

[147] Elizabeth G. Hill et al. “A statistical model for iTRAQ data analysis.” In: *Journal of Proteome Research* 7.8 (Aug. 2008), pp. 3091–101.

[148] Douglas W. Mahoney et al. “Relative Quantification: Characterization of bias, variability and fold changes in mass spectrometry data from iTRAQ labeled peptides.” In: *Journal of Proteome Research* (July 2011).

[149] John S. Cottrell and David M. Creasy. “Response to: The Problem with Peptide Presumption and Low Mascot Scoring.” In: *Journal of Proteome Research* (Sept. 2011).

[150] Markus Brosch et al. “Comparison of Mascot and X!Tandem performance for low and high accuracy mass spectrometry and the development of an adjusted Mascot threshold.” In: *Molecular & Cellular Proteomics* 7.5 (May 2008), pp. 962–70.

[151] Cathy H. Wu et al. “The Universal Protein Resource (UniProt): an expanding universe of protein information.” In: *Nucleic Acids Research* 34.Database issue (Jan. 2006), pp. D187–91.

[152] Johannes Griss et al. “Published and Perished? the influence of the searched protein database on the long-term storage of proteomics data.” In: *Molecular & Cellular Proteomics* (June 2011).

[153] Steffen Durinck et al. “BioMart and Bioconductor: a powerful link between biological databases and microarray data analysis.” In: *Bioinformatics (Oxford, England)* 21.16 (Aug. 2005), pp. 3439–40.

[154] Nan Wang and Liang 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.” In: *Analytical Chemistry* 80.12 (June 2008), pp. 4696–710.

[155] Magnus Ø Arntzen et al. “IsobariQ: software for isobaric quantitative proteomics using IPTL, iTRAQ, and TMT.” In: *Journal of Proteome Research* 10.2 (Feb. 2011), pp. 913–20.

[156] Pei Wang et al. “Normalization regarding non-random missing values in high-throughput mass spectrometry data.” In: *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing* (Jan. 2006), pp. 315–26.

[157] Andreas M. Boehm et al. “Precise protein quantification based on peptide quantification using iTRAQ.” In: *BMC Bioinformatics* 8 (Jan. 2007), p. 214.

[158] Claudia Hundertmark et al. “MS-specific noise model reveals the potential of iTRAQ in quantitative proteomics.” In: *Bioinformatics (Oxford, England)* 25.8 (Apr. 2009), pp. 1004–11.

[159] Yi Zhang et al. “A robust error model for iTRAQ quantification reveals divergent signaling between oncogenic FLT3 mutants in acute myeloid leukemia.” In: *Molecular & Cellular Proteomics* 9.5 (May 2010), pp. 780–90.

[160] Paul M. Harrison et al. “A question of size: the eukaryotic proteome and the problems in defining it.” In: *Nucleic Acids Research* 30.5 (Mar. 2002), pp. 1083–90.

[161] Linfeng Wu and David K. Han. “Overcoming the dynamic range problem in mass spectrometry-based shotgun proteomics.” In: *Expert Review of Proteomics* 3.6 (Dec. 2006), pp. 611–9.

[162] Paola Picotti, Ruedi Aebersold, and Bruno Domon. “The implications of proteolytic background for shotgun proteomics.” In: *Molecular & Cellular Proteomics* 6.9 (Sept. 2007), pp. 1589–98.

[163] Stephanie M. Pütz et al. “iTRAQ Analysis of a Cell Culture Model for Malignant Transformation, Including Comparison with 2D-PAGE and SILAC.” In: *Journal of Proteome Research* 11.4 (Apr. 2012), pp. 2140–53.

[164] Atanas Kamburov et al. “ConsensusPathDB: toward a more complete picture of cell biology.” In: *Nucleic Acids Research* 39.Database issue (Jan. 2011), pp. D712–7.

[165] Parag Mallick et al. “Computational prediction of proteotypic peptides for quantitative proteomics.” In: *Nature Biotechnology* 25.1 (Jan. 2007), pp. 125–31.

[166] Claire E. Eyers et al. “CONSeQuence: prediction of reference peptides for absolute quantitative proteomics using consensus machine learning approaches.” In: *Molecular & Cellular Proteomics* 10.11 (Nov. 2011), p. M110.003384.

[167] C. H. Tanford. *Physical Chemistry of Macromolecules.* Wiley, 1961.

[168] Robin E. Offord. “Electrophoretic mobilities of peptides on paper and their use in the determination of amide groups.” In: *Nature* 211.5049 (Aug. 1966), pp. 591–3.

[169] Eugene C. Rickard, M. M. Strohl, and R. G. Nielsen. “Correlation of electrophoretic mobilities from capillary electrophoresis with physicochemical properties of proteins and peptides.” In: *Analytical biochemistry* 197.1 (Aug. 1991), pp. 197–207.

[170] Jeongkwon Kim, Robert Zand, and David M. Lubman. “Electrophoretic mobility for peptides with post-translational modifications in capillary electrophoresis.” In: *Electrophoresis* 24.5 (Mar. 2003), pp. 782–93.

[171] Fernando Benavente et al. “Modelling migration behavior of peptide hormones in capillary electrophoresis-electrospray mass spectrometry.” In: *Journal of Chromatography A* 1117.1 (June 2006), pp. 94–102.

[172] Donald J. Winzor. “Classical approach to interpretation of the charge-dependence of peptide mobilities obtained by capillary zone electrophoresis.” In: *Journal of Chromatography A* 1015.1-2 (Oct. 2003), pp. 199–204.

[173] Petra Zürbig et al. “Biomarker discovery by CE-MS enables sequence analysis via MS/MS with platform-independent separation.” In: *Electrophoresis* 27.11 (June 2006), pp. 2111–25.

Erklärung

Ich versichere, daß ich die von mir vorgelegte Dissertation selbstständig angefertigt, und die benutzten Quellen und Hilfsmittel vollständig angegeben sind.

Ein Promotionsverfahren wurde zu keinem früheren Zeitpunkt an einer anderen in- oder ausländischen Hochschule oder bei einem anderen Fachbereich beantragt. Die Bestimmungen der Promotionsordnung sind mir bekannt.

Berlin, 8. Juni 2012

Chris Bielow