

## TOPP – The OpenMS Proteomics Pipeline

Oliver Kohlbacher<sup>a</sup>, Knut Reinert<sup>b</sup>, Clemens Gröpl<sup>b</sup>,  
Eva Lange<sup>b</sup>, Nico Pfeifer<sup>a</sup>, Ole Schulz-Trieglaff<sup>b</sup>,  
Marc Sturm<sup>a</sup>

<sup>a</sup> Simulation of Biological Systems, Eberhard Karls University Tübingen, Sand 14, 72076 Tübingen, Germany

<sup>b</sup> Algorithmic Bioinformatics, Free University Berlin, Takustr. 9, 14195 Berlin, Germany

### ABSTRACT

**Motivation:** Experimental techniques in proteomics have seen rapid development over the last few years. Volume and complexity of the data have both been growing at a similar rate. Accordingly, data management and analysis are one of the major challenges in proteomics. Flexible algorithms are required to handle changing experimental setups and to assist in developing and validating new methods. In order to facilitate these studies, it would be desirable to have a flexible “toolbox” of versatile and user-friendly applications allowing for rapid construction of computational workflows in proteomics.

**Results:** We describe a set of tools for proteomics data analysis – TOPP, The OpenMS Proteomics Pipeline. TOPP provides a set of computational tools which can be easily combined into analysis pipelines even by non-experts and can be used in proteomics workflows. These applications range from useful utilities (file format conversion, peak picking) over wrapper applications for known applications (e.g. Mascot) to completely new algorithmic techniques for data reduction and data analysis. We anticipate that TOPP will greatly facilitate rapid prototyping of proteomics data evaluation pipelines. As such, we describe the basic concepts and the current abilities of TOPP and illustrate these concepts in the context of two example applications: the identification of peptides from a raw data set through database search and the complex analysis of a standard addition experiment for the absolute quantitation of biomarkers. The latter example demonstrates TOPP's ability to construct flexible analysis pipelines in support of complex experimental setups.

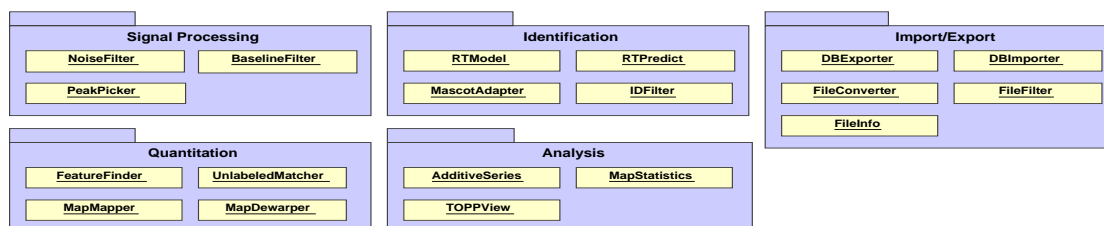
**Availability:** The TOPP components are available as open-source software under the lesser GNU public license (LGPL). Source code is available from the project web site at [www.OpenMS.de](http://www.OpenMS.de)

**Contact:** [oliver.kohlbacher@uni-tuebingen.de](mailto:oliver.kohlbacher@uni-tuebingen.de)

### 1 INTRODUCTION

HPLC-MS-based proteomics applications require the management of large amounts of data in quite complex ways. While some key steps in the data analysis pipeline are common to all applications, the arrangement of these steps and the context of the data analysis is highly dependent on the kind of experiment being conducted. Hence, the overall data analysis pipeline is often subjected to large changes, while the underlying analysis algorithms remain mostly identical. We propose a new set of proteomics tools built upon our software framework OpenMS which addresses this problem using a set of predefined tools that can be combined into a proteomics workflow in a simple file-driven way. Adapting a proteomics analysis pipeline to a new experiment or a new data analysis strategy then only requires minor modifications to this workflow. Each tool handles a well-defined functionality in the area of proteomics data analysis. While the individual applications range from very trivial to rather complex tasks (e.g. file format conversion, peak picking, noise reduction, spectrum identification, quantitation, etc.) their combined value arises from the fact that they share a common interface, common formats, and common configuration files. They can thus be combined like building blocks to perform more complex analysis tasks, an idea already used in similar toolboxes in bioinformatics, e.g. in EMBOSS [15]. The analysis pipeline then defines the wiring of these building blocks by executing small scripts connecting individual building blocks. Manual analyses during the development of a pipeline are supported through a system of log files allowing the reconstruction of every processing step. The debugging output can be turned off as soon as the pipeline works as intended.

Some of the tasks above can be performed with the vendor software of the mass spectrometer as well. An example is the commercial software MassLynx (Waters, Inc.) which performs relative quantitation of proteins without the use of labeling methods. However, we believe that OpenMS offers a much higher degree of flexibility since the user is allowed to



**Fig. 1.** The TOPP tools are grouped into five packages, each addressing a major area of functionality.

combine the individual components of TOPP according to his individual needs. Furthermore all algorithmic details are either published or can be found in the documentation. The user is always in full control of the data workflow and not dependent on any manufacturer, just to name another advantage TOPP has over most commercial software.

There are also some academic software projects with aims similar to OpenMS. Closest to our idea is the Trans-Proteomic Pipeline (TPP) [5] developed at the Institute of Systems Biology (ISB) in Seattle (USA). The TPP makes use of open XML file formats for storage of data at the raw data, peptide, and protein levels. It integrates other tools developed at the ISB into a coherent framework. Among these tools are *PeptideProphet* [6] which validates peptides assigned to MS/MS spectra, *XPRESS* [3] and *ASAPRatio* [11] that quantify peptides and proteins in differentially labeled samples. *Pep3D* [9] visualizes the raw spectral data, and *ProteinProphet* [13] infers sample proteins. At its current status, the main emphasis of the TPP is on peptide identification and quantitation. Only limited preprocessing of the data is possible and the software deals with the spectra as they leave the mass spectrometer.

Several groups such as [4, 8, 10, 18, 23, 24] have developed other working systems for proteomic data analyses. Most of them focus on a single task such as protein identification or quantitation. However, in some situations, it might be preferable to have more control over the workflow of the data analysis in order to build customized applications or even implement own algorithms using existing data structures. TOPP offers all of these possibilities. In its current version, it can perform the main computational tasks occurring in proteomics experiments such as visualization, protein identification, quantitation, alignment (mapping) of samples and a basic statistical analysis. Besides, it comes with a comprehensible documentation and simple interfaces. Moreover, it is possible to develop new applications and to contribute ideas to the OpenMS framework.

By using standardized data exchange formats it is possible to combine TOPP components into complex workflows. In contrast to other academic projects, we did not only develop sophisticated algorithms but also aim for a software which is easy to use and most of all extensible.

In the following section we give a short overview of the main components implemented in TOPP and then demonstrate the

versatility of our system by using the components in some pipelines that can be used for complex proteomics analysis.

## 2 TOPP COMPONENTS

The individual tools (components) of TOPP can be grouped into several distinct packages: import/export, signal processing, identification, quantitation, and analysis (see Fig. 1). We will now briefly discuss the major components of each area.

### 2.1 Import/Export

File handling is important for all proteomics data analysis tools as there are at least two standard file formats and a lot of proprietary formats. The **FileConverter** converts several commonly used MS formats into each other. Supported formats are mzData (HUPO - PSI) [17], mzXML (Sashimi project) [19], ANDI/MS, and several text-based formats.

The **FileFilter** extracts parts of a file such as specific types of spectra or data intervals. It allows the specification of a set of rules and extracts all data matching these rules from the input file. The filtering can be based on spectrum type, e.g. extract all MS/MS spectra from a combined MS-MS/MS run or on geometric criteria. In the latter case, it allows simple range operations on the data, such as the extraction of rectangles with respect to retention time (RT) and mass-to-charge (m/z) from an HPLC-MS run, or the extraction of a specific m/z region from a set of MS spectra. The **FileInfo** shows basic information about an MS data file, i.e. m/z range, RT range, intensity range and the type of the spectra in the file.

As the TOPP components support file-based operations only the two auxiliary components **DBImporter** and **DBExporter** are provided for database connectivity. **DBExporter** exports experimental data from an OpenMS database to one or several files. After processing the data **DBImporter** is used to store the results in the database again. Database connectivity is especially useful to distribute data for grid and workflow applications.

### 2.2 Signal Processing

MS raw data is always disturbed by baseline fluctuations and two types of noise: chemical (colored) noise, and random (white) noise. To improve the reliability of the data for further analysis steps, noise and baseline should be removed.

For noise reduction we implemented two different smoothing filters in the signal processing component **NoiseFilter**, which are a peak area preserving Gaussian low-pass filter and a Savitzky-Golay low-pass filter recommended for spectrometric data [22, 25].

Baseline correction of raw MS data can be performed using the **BaselineFilter** component. For the baseline in MS experiments no universally accepted analytical expression exists. Therefore, we decided to implement a nonlinear filter, known as the top-hat operator in morphological mathematics [26], as it does not depend on the underlying baseline shape.

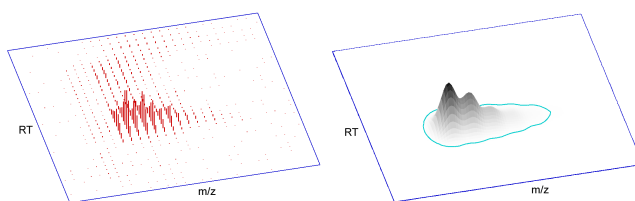
Many of the higher level analysis steps, like identification, rely on precise information about mass spectrometric peaks. The **PeakPicker** component extracts this information, that means converts the "raw" ion count acquired by the mass spectrometer into peak lists. Our peak picking approach [7] uses the multi-scale nature of spectrometric data and first detects the mass peaks in the wavelet-transformed signal. Afterwards important peak parameters (centroid, area, height, signal-to-noise ratio, asymmetric peak shape) are extracted by fitting an asymmetric peak function to the raw data. In an optional third step, the resulting fit can be further improved by using techniques from nonlinear optimization. In contrast to currently established techniques our algorithm yields precise peak positions even for data with low resolution and is able to separate overlapping peaks of multiply charged peptides.

### 2.3 Identification

By sending MS/MS spectra to an identification tool one can determine the peptides present in a sample, which in turn can be used to identify the proteins. TOPP will provide a number of adapters for identification tools. So far **MascotAdapter**, an adapter for the database-driven search algorithm Mascot [20], is available. Adapters for other popular search engines such as SEQUEST [27] and InSpecT [28] as well as an adapter to the *de novo* sequencing code LuteFisk [29] are under development.

These adapters facilitate the integration of identification tools by handling both their input and output. They transform a set of MS/MS spectra to the specific formats required for input. After analysis by the identification tool, the resulting output is parsed and converted to analysisXML, a Proteomics Standards Initiative (PSI [16]) compliant format for identification. Since there is often more than one candidate peptide hit for a certain spectrum we provide the **IDFilter** component to filter out relevant hits by different filter criteria. One such filter criterion is that the score of the peptide hit exceeds a significance threshold.

To distinguish furthermore between correct and incorrect peptide hits the retention time can be taken into account [21]. This is done by the **RTModel** and the **RTPredict** components. **RTModel**, which uses a support vector machine [1], is trained using high quality peptide - retention time pairs. Afterwards



**Fig. 2.** A small part of the raw data (left) and a model adjusted to it by FeatureFinder (right).

the model can be used in the **RTPredict** component to predict retention times for peptide hits. These predicted retention times can then be used to filter the peptide hits since a large difference between measured and predicted retention time suggests a false peptide identification.

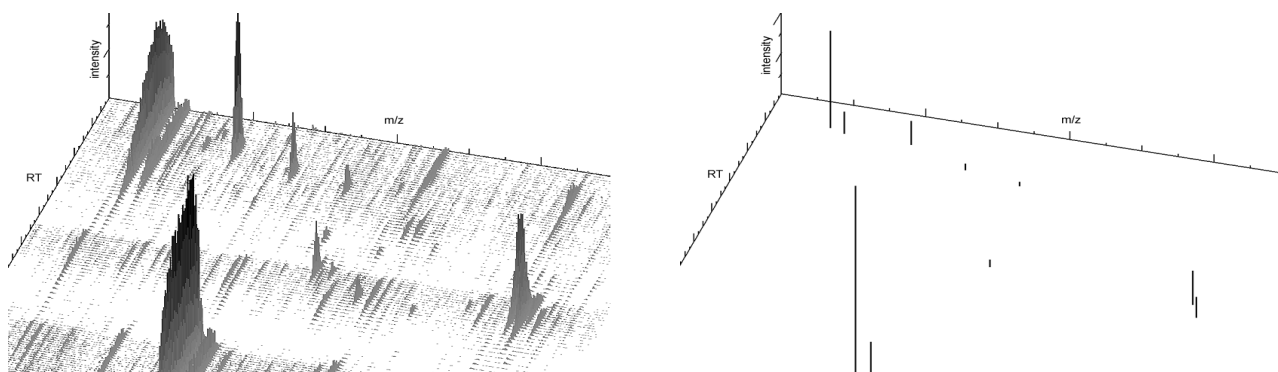
### 2.4 Quantitation

HPLC-MS experiments produce a flood of data that is difficult to handle and analyze. It is necessary to reduce the raw instrument data to the essential features therein: the retention time, mass-to-charge ratio, and intensity of each peptide (or any other component) eluting from the column. The transformation of raw instrument data to so-called *feature maps* reduces the data volume and improves the running time of further analysis steps [2]. Besides, it yields valuable secondary information which is not immediately evident from the raw data such as charge estimates of peptides. The idea of two-dimensional raw data maps and the concept of peptide features is not novel [8, 23] but has just started to emerge as a basis of quantitation and visualization of MS data.

Another important step, especially for differential quantitation, is the alignment of two maps. To calculate an alignment matching features between the two maps are determined. The list of matching features forms the basis for the computation of a suitable transformation between the two maps. The transformation is then applied to correct the coordinates of one of the maps such that both can be superimposed.

Initially a feature map is created from HPLC raw data by a run of the **FeatureFinder** component. It identifies the raw data points belonging to a feature and fits a two-dimensional model to the extracted region of the input data. The model is based on a bi-Gaussian elution profile along the retention time axis (or any other appropriate function) and a theoretical isotope pattern along the  $m/z$  axis. The output of the feature finder is a map of features, each identified by its RT and  $m/z$  coordinates and its intensity. Fig. 3 shows a part of the raw data file and the features found in it. Fig. 2 shows an example of how a feature model is adjusted to a (small) segment of the input data. The details of the algorithm have been described elsewhere [2, 12]. Features can have annotations like the charge state and the quality of the model fit to the data.

The **UnlabeledMatcher**'s task is the matching of features across maps. It takes two feature maps as input and creates a



**Fig. 3.** Feature finding from a global perspective. A section of a LC/MS raw data map (left) and the features extracted by the FeatureFinder (right). Visualization was done using TOPPView.

list of feature pairs. In its simplest form it will find all pairs of matching features according to some user-specified distance criteria.

Due to calibration issues and changes in the experiment settings both mass-to-charge ratio and HPLC retention time can vary between two experiments. That is why features arising from the same species often need to be shifted along the RT or  $m/z$  dimension before matching. Thus we also developed a more sophisticated algorithm based on *geometric hashing* [30] to produce possible pairs of features after a suitable translation.

The transformation itself is computed by **MapMapper**. MapMapper takes as input a bijection between some features in these maps. It then computes a *dewarping function* that corrects for coordinate shifts between the two maps. The dewarping function is typically a piecewise linear or quadratic function. Its application to the coordinates of the second map will move all features in this map as close as possible to the coordinates of their corresponding features in the first map.

The transformation function is then applied to a map by **MapDewarper**. As described above, we decided to divide the process of MS sample alignment into three stages: computation of potential feature partners, estimation of a transformation of the feature coordinates, and application of this transformation (dewarping). The reason for this decision was to achieve a high flexibility. This architecture allows us to use different computational approaches in each step and to improve the individual TOPP components independently from each other.

## 2.5 Analysis

Having preprocessed the data, one might want to perform further analyses. TOPP offers several components for this purpose. **AdditiveSeries** can be used to conduct an absolute quantitation of peptides. It uses the feature intensities as computed by the FeatureFinder application and evaluates the data of an additive measurement (see Section 4.2).

**TOPPView** can be used for visual inspection of MS data. It is able to visualize one-dimensional spectra, two-dimensional maps and the results of our peak picking and feature finding algorithms. TOPPView supports all the file formats used by TOPP and can visualize data from OpenMS databases as well.

Finally, the tool **MapStatistics** computes a five-number summary of a feature map. This summary consists of median, minimum, maximum and the quartiles of the feature intensities and qualities in a map. These values provide a measure of location and spread and can be used to estimate the quality of the preprocessing steps. Feature maps with highly unusual statistics might be excluded from the further analysis workflow.

## 3 DESIGN AND IMPLEMENTATION

All the components of TOPP are designed to be versatile. They can be used both individually as command line tools and chained into linear or more complex pipelines. Chaining is done through makefiles, simple shell scripts or as components of complex workflow systems in distributed or GRID environments, e.g. by workflow systems like Taverna [14].

In order to make the TOPP components easy to combine, we use standard file formats such as *mzData* and *analysisXML* only. This also facilitates the integration of external tools supporting standard formats. A pipeline-specific control file provides parameters to all components and directs the data flow between them. In the control file a set of parameters for each individual invocation of a tool can be provided. For tasks which cannot be done with TOPP, wrapper components are provided to integrate commonly used applications. As an example, Mascot can be integrated to perform peptide identification by database search.

One of the design goals is user-friendliness. Hence, all TOPP components share a common base interface and provide a detailed description for all parameters. Additionally, a full documentation of all components and examples are available on our web site.

TOPP is based on OpenMS, an object-oriented software platform for shotgun proteomics. OpenMS makes extensive use of generic programming techniques in C++ and thereby provides fast execution of programs and portable code. It is tested on different Linux platforms (e.g. Fedora Core 4, Scientific Linux 4 and Suse Linux 9.0-10.1) using 32 bit and 64 bit architectures. OpenMS itself is based on several other open-source libraries such as QT (TrollTech Inc.) and the GNU Scientific Library.

The TOPP components use OpenMS datastructures and algorithms and provide a coherent interface for them. As TOPP shows, OpenMS can easily be used or extended in order to create new applications in the field of proteomics data analysis. It has already been used in other projects [7, 12] and will be developed further continuously. All future extensions of OpenMS functionality will be turned into new TOPP components making the pipeline more powerful. Just as TOPP, OpenMS comes with a Doxygen documentation of all the classes. Additionally, a tutorial and several example applications offer a starting point for new users. OpenMS and TOPP are available as open-source software under the lesser GNU public license (LGPL) from [www.OpenMS.de](http://www.OpenMS.de).

## 4 EXAMPLE APPLICATIONS

Using the TOPP components, one can easily set up simple, yet powerful proteomics workflows. In its simplest form, a TOPP workflow merely consists of a shell script calling the individual components in a well-defined order. The output of each component is passed on to the next component. All components obtain their settings from a common configuration file, which contains individual sections for each component. The settings can be passed through the command line as well, but this method is more error-prone. In this section we will present two examples of analyses that we successfully implemented and ran using TOPP.

### 4.1 Simple Tandem-MS Identification Pipeline

In this example pipeline the task is to identify all the peptides in an HPLC-MS/MS run. The starting point is the raw data exported from an MS machine (in one of the supported raw data formats). In the first step MS/MS spectra are extracted from the HPLC-MS file, as only these spectra are needed for the identification. In order to improve the quality of the data a noise filter is applied before reducing the raw data to stick spectra with the PeakPicker. Finally, Mascot is used to produce a list of peptide identification candidates for each spectrum. These candidates are then validated using the IDFilter. Unlikely identifications are removed in this step. The output of the pipeline consists of a list of identified peptides and the reliability of each identification in analysisXML format.

The shell script executing the pipeline is given below:

```
## Simple Tandem MS Identification Pipeline
# Convert raw data to mzData format.
FileConverter -in $1 -out id.mzData
# Extract MS/MS spectra only.
FileFilter -in id.mzData -out raw.mzData -level 2
# Noise filtering.
NoiseFilter -in raw.mzData -out rawf.mzData -ini ID.ini
# Reduction to stick spectra.
PeakPicker -in rawf.mzData -out peaks.mzData -ini ID.ini
# Identify spectra.
MascotAdapter -in peaks.mzData -out id.analysisXML -ini ID.ini
# Filter out reliable identifications.
IDFilter -in id.analysisXML -out result.analysisXML -ini ID.ini
```

The result of this particular workflow is an XML document adhering to the standards proposed by the PSI, which can be displayed in every standard web browser using appropriate style sheets. Except for the intermediary raw data files, all files produced in the pipeline are human-readable formats and – as far as possible – adhere to PSI standards.

We tested the pipeline on 1371 MS/MS spectra of a sample with five different proteins having a total of 234 possible tryptic peptides. These 1371 spectra led to 74 identified peptides, the majority of which could be assigned to the five proteins. The whole pipeline took about twelve minutes on a two processor AMD Opteron 250.

### 4.2 Absolute Quantitation

The goal of our second example workflow is to determine the concentration of myoglobin, a marker for myocardial infarction, in human serum. The experimental setup has already been described in detail [12] as well as the computational approach [2]. In this section, we will therefore merely summarize the key concepts and then describe how we implemented this workflow in TOPP.

Myoglobin is a protein of low-molecular mass present in the cytosol of the cardiac and skeletal muscle. It quickly appears in blood after tissue injury and therefore represents a well-known biochemical marker for myocardial infarction. Nevertheless, results from different analytical procedures for myoglobin quantitation showed significant bias due to a lack of assay standardization. The aim of the project described in [12] was to develop a reference measurement procedure for myoglobin in human serum. Strong anion-exchange (SAX) chromatography was used to separate highly abundant serum proteins from the myoglobin fraction, which was then trypsinized and analyzed by reversed-phase high-performance liquid chromatography in combination with electrospray ionization mass spectrometry (RP-HPLC-ESI-MS).

To reduce the quantitation error, a constant amount of horse myoglobin was added as internal standard. Furthermore, an additive series was performed by adding known amounts of human myoglobin to aliquots of the sample. The absolute quantitation was conducted by determining the x-intercept of a

linear regression using the ratio of the eleventh tryptic peptide of human myoglobin (*T11hu*) and the tenth tryptic peptide of horse myoglobin (*T10ho*).

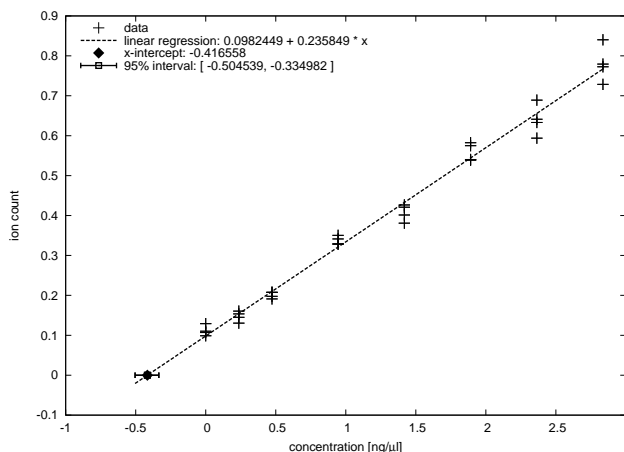
To speed up the computational analysis, all raw data files were truncated to a retention time range of 900 to 1600 s and an m/z range of 600 to 1000 Th. These ranges were chosen such that all myoglobin peptides were included in the truncated raw data maps. In addition, data points with very low intensities were filtered out using a threshold well below the noise level. No further preprocessing or peak picking was performed. We used our feature finding algorithm as it was described in Section 2.4 to identify and quantify the myoglobin peptides.

The workflow was this time implemented in TOPP, in contrast to the procedure described in [2], and allows a fully automated execution of the analysis. By using the TOPP components described above we were able to perform the myoglobin quantitation by executing a simple shell script. Furthermore, we used the TOPP components to estimate the shift in retention time and m/z between the different LC/MS maps and to map the myoglobin features from different maps directly onto each other (see Section 2.4 for details). This allows a direct comparison of different samples and reduces the likelihood of errors.

The shell script executing the pipeline is given below. First, all maps are truncated and the FeatureFinder is executed on each data set. In the second loop, all feature maps are mapped on one reference map in a star-like alignment. Each TOPP module reads its parameters from the file `AddSeries.ini` which holds a separate section for each data set. The number of the current section is given by parameter `-n`. Executing this pipeline took less than an hour for each map. Our feature detection algorithm found on average 300 features in each map. The size of the raw data set was 258 MB after the truncation, and the feature detection led to a reduction by 90 % to 26 MB in total.

```
## Pipeline for Myoglobin Absolute Quantitation
# Find features in all 32 individual maps.
for i in `seq 1 32`; do
  # Truncate raw data maps to save time.
  FileFilter -ini AddSeries.ini -n $i
  # Collect peptide features.
  FeatureFinder -ini AddSeries.ini -n $i
done
# Star-like matching (31 edges).
for i in `seq 1 31`; do
  # Map features across different maps.
  UnlabeledMatcher -ini AddSeries.ini -n $i
  MapMatcher -ini AddSeries.ini -n $i
  Dewarper -ini AddSeries.ini -n $i
done
# Compute final concentration (lin. regression).
AdditiveSeries -ini AddSeries.ini
```

The results of this experiment are shown in Fig. 4. Our quantitation gives an estimate of the absolute myoglobin concentration of 0.417 ng/ $\mu$ l (true value is 0.463 ng/ $\mu$ l) whereas a manual expert analysis yielded a result of 0.382 ng/ $\mu$ l.



**Fig. 4.** Regression results for the automated analysis of myoglobin in the serum samples.

## 5 DISCUSSION AND CONCLUSION

We have presented TOPP (The OpenMS Proteomics Pipeline) – a set of practical tools that can easily be combined into proteomics pipelines. The TOPP components are based on OpenMS, the underlying C++ framework. Two standard proteomics workflows (a simple tandem-MS identification pipeline, and an absolute quantitation by an additive series) have been described in detail to show its functionality.

The development of software platforms for mass spectrometry based proteomics is currently a very active field of research. We believe that OpenMS, being an open-source project under the lesser GNU public license, fills a gap between commercial software products supplied by vendors of mass spectrometers and academic software projects that are often not very user friendly and offer a much smaller range of functions than TOPP.

TOPP and OpenMS are being developed further in ongoing research projects. For example, we plan to provide a TOPP pipeline for the detection and analysis of feature pairs resulting from labeling techniques like e.g. ICAT. Another planned extension is to improve upon the current capabilities of peptide identification algorithms.

## ACKNOWLEDGMENTS

The authors wish to thank Prof. Dr. Christian Huber (Saarland University, Germany) for providing the experimental data used to test the example pipelines.

## REFERENCES

- [1] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- [2] C. Gröpl, E. Lange, K. Reinert, O. Kohlbacher, M. Sturm, C. G. Huber, B. M. Mayr, and C. L. Klein. Algorithms for the automated absolute quantification of diagnostic markers in complex proteomics samples. In Michael Berthold, editor, *Proceedings of CompLife 2005*, Lecture Notes in Bioinformatics, pages 151–163. Springer, Heidelberg, 2005.
- [3] D. K. Han, J. Eng, H. Zhou, and R. Aebersold. Quantitative profiling of differentiation-induced microsomal proteins using isotope-coded affinity tags and mass spectrometry. *Nat. Biotechnol.*, 19:946–951, 2001.
- [4] Mikko Katajamaa, Jarkko Miettinen, and Matej Oresic. MZmine: Toolbox for processing and visualization of mass spectrometry based molecular profile data. *BMC Bioinformatics*, 6(179):634–636, 2006.
- [5] A. Keller, J. Eng, N. Zhang, Xiao jun Li, and R. Aebersold. A uniform proteomics MS/MS analysis platform utilizing open XML file formats. *Mol. Syst. Biol.*, 1:1744–4292, 2005.
- [6] Andrew Keller, A Nesvizhskii, E Kolker, and R Aebersold. Empirical statistical model to estimate the accuracy of peptide identifications made by MS/MS and database search. *Anal. Chem.*, 74:5383–5392, 2002.
- [7] E. Lange, C. Gröpl, K. Reinert, O. Kohlbacher, and A. Hildebrandt. High accuracy peak-picking of proteomics data using wavelet techniques. In *Proceedings of PSB 2006*, pages 243–254, 2006.
- [8] Kyriacos C. Leptos, David A. Sarracino, Jacob D. Jaffe, Bryan Krastins, and George M. Church. MapQuant: Open-source software for large-scale protein quantification. *Proteomics*, 6(6):1770–1782, 2006.
- [9] Xiao-Jun Li, PGA Pedrioli, J Eng, D Martin, EC Yi, H Lee, and R Aebersold. A tool to visualize and evaluate data obtained by liquid chromatography/electrospray ionization/mass spectrometry. *Anal. Chem.*, 76:3856–3860, 2004.
- [10] Xiao-Jun Li, Eugene C. Yi, Christopher J. Kemp, Hui Zhang, and Ruedi Aebersold. A Software Suite for the Generation and Comparison of Peptide Arrays from Sets of Data Collected by Liquid Chromatography-Mass Spectrometry. *Mol. Cell Proteomics*, 4(9):1328–1340, 2005.
- [11] Xiao-Jun Li, H Zhang, JR Ranish, and R Aebersold. Automated statistical analysis of protein abundance ratios from data generated by stable isotope dilution and tandem mass spectrometry. *Anal. Chem.*, 75:6648–6657, 2003.
- [12] B. M. Mayr, O. Kohlbacher, K. Reinert, M. Sturm, C. Groepel, E. Lange, C. Klein, and C. Huber. Absolute myoglobin quantitation in serum by combining two-dimensional liquid chromatography-electrospray ionization mass spectrometry and novel data analysis algorithms. *J. Proteome Res.*, 5:414–421, 2006.
- [13] AI Nesvizhskii, A Keller, E Kolker, and R Aebersold. A statistical model for identifying proteins by tandem mass spectrometry. *Anal. Chem.*, 75:4646–4658, 2003.
- [14] Tom Oinn, Matthew Addis, Justin Ferris, Darren Marvin, Martin Senger, Mark Greenwood, Tim Carver, Kevin Glover, Matthew R. Pocock, Anil Wipat, and Peter Li. Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics*, 20(17):3045–54, 2004.
- [15] Sue A Olson. EMBOSS opens up sequence analysis. European Molecular Biology Open Software Suite. *Brief Bioinform.*, 3(1):87–91, Mar 2002.
- [16] S. Orchard, H. Hermjakob, and R. Apweiler. The proteomics standards initiative. *Proteomics*, 3(7):1374–6, Jul 2003.
- [17] S. Orchard, H. Hermjakob, C. Taylor, Pierre-Alain Binz, C. Hoogland, R. Julian, John S. Garavelli, R. Aebersold, and R. Apweiler. Autumn 2005 Workshop of the Human Proteome Organisation Proteomics Standards Initiative (HUPO-PSI) Geneva, September, 4-6, 2005. *Proteomics*, 6(3):738–41, 2006.
- [18] Patricia M. Palagi, Daniel Walther, et al. MSight: An image analysis software for liquid chromatography-mass spectrometry. *Proteomics*, 5(9):2381–2384, 2005.
- [19] Patrick G. A. Pedrioli, Jimmy K. Eng, R. Hubley, et al. A common open representation of mass spectrometry data and its application to proteomics research. *Nat Biotech.*, 22(11):1459–1466, 2004.
- [20] D. N. Perkins, D. J. C. Pappin, D. M. Creasy, and J. S. Cottrell. Probability-based protein identification by searching sequence databases using mass spectrometry data. *Electrophoresis*, 20:3551–3567, 1999.
- [21] K. Petritis, L. J. Kangas, P. L. Ferguson, G. A. Anderson, L. Pasatolić, M. S. Lipton, K. J. Auberry, E. F. Strittmatter, Y. Shen, R. Zhao, and R. D. Smith. Use of artificial neural networks for the accurate prediction of peptide liquid chromatography elution times in proteome analyses. *Anal. Chem.*, 75:1039–1048, 2003.
- [22] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C++: The art of scientific computing*. Cambridge University Press, 2002.
- [23] Dragan Radulovic, Salomeh Jelveh, Soyoung Ryu, T. Guy Hamilton, Eric Foss, Yongyi Mao, and Andrew Emili. Informatics platform for global proteomic profiling and biomarker discovery using liquid chromatography-tandem mass spectrometry. *Mol. Cell Proteomics*, 3(10):984–997, 2004.
- [24] Jim Samuelsson, Daniel Dalevi, Fredrik Levander, and Thorsteinn Rognvaldsson. Modular, scriptable and automated analysis tools for high-throughput peptide mass fingerprinting. *Bioinformatics*, 20(18):3628–3635, 2004.
- [25] A. Savitzky and M.J.E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Anal. Chem.*, 36:1627–1639, 1964.
- [26] P. Soille. *Morphological Image Analysis*. Springer, 1999.
- [27] D. L. Tabb, J. K. Eng, and J. R. Yates. *Protein Identification by SEQUEST*, volume 1, pages 125–142. Springer, 2001.
- [28] S. Tanner, H. Shu, A. Frank, L.-C. Wang, E. Zandi, M. Mumby, P. A. Pevzner, and V. Bafna. Inspect: Fast and accurate identification of post-translationally modified peptides from tandem mass spectra. *Anal. Chem.*, 77(14):4626–39, 2005.
- [29] J. A. Taylor and R. S. Johnson. Implementation and uses of automated de novo peptide sequencing by tandem mass spectrometry. *Anal. Chem.*, 73:2594–2604, 2000.
- [30] Haim J. Wolfson and Isidore Rigoutsos. Geometric hashing: An overview. *IEEE Computational Science and Engineering*, 4(4):10–21, 1997.