Gene expression

Retention time alignment algorithms for LC/MS data must consider non-linear shifts

Katharina Podwojski1,2,*,†, Arno Fritsch1,†, Daniel C. Chamrao3, Wolfgang Paul2,4, Barbara Sitek2,5, Kai Stühler1,2,5, Petra Mutzel4, Christian Stephan5, Helmut E. Meyer5, Wolfgang Urfer1, Katja Ickstadt1 and Jörg Rahnenführer1

1Fakultät Statistik, Technische Universität Dortmund, 44221 Dortmund, 2Zentrum für Angewandte Proteomik, Dortmund, 3Protagen AG, Otto-Hahn-Str. 15, 44227 Dortmund, 4Fakultät für Informatik, Technische Universität Dortmund, 44221 Dortmund and 5Medizinisches Proteom-Center (MPC), Ruhr-Universität Bochum, 44801 Bochum, Germany

Received on October 7, 2008; revised and accepted on January 22, 2009
Advance Access publication January 28, 2009
Associate Editor: John Quackenbush

ABSTRACT

Motivation: Proteomics has particularly evolved to become of high interest for the field of biomarker discovery and drug development. Especially the combination of liquid chromatography and mass spectrometry (LC/MS) has proven to be a powerful technique for analyzing protein mixtures. Clinically oriented proteomic studies will have to compare hundreds of LC/MS runs at a time. In order to compare different runs, sophisticated preprocessing steps have to be performed. An important step is the retention time (rt) alignment of LC/MS runs. Especially non-linear shifts in the rt between pairs of LC/MS runs make this a crucial and non-trivial problem.

Results: For the purpose of demonstrating the particular importance of correcting non-linear rt shifts, we evaluate and compare different alignment algorithms. We present and analyze two versions of a new algorithm that is based on regression techniques, once assuming and estimating only linear shifts and once also allowing for the estimation of non-linear shifts. As an example for another type of alignment method we use an established alignment algorithm based on shifting vectors that we adapted to allow for correcting non-linear shifts also. In a simulation study, we show that rt alignment procedures that can estimate non-linear shifts yield clearly better alignments. This is even true under mild non-linear deviations.

Availability: R code for the regression-based alignment methods and simulated datasets are available at http://www.statistik.tu-dortmund.de/genetik-publikationen-alignment.html
Contact: katharina.podwojski@tu-dortmund.de
Supplementary information: Supplementary data are available at Bioinformatics online.

1 INTRODUCTION

Proteomics is the large-scale study of proteins aiming at elucidating proteins’ structure and function in complex biological systems. Quantitative differential studies that compare the proteomes of two or more different groups of samples offer the opportunity to identify new biomarkers for diagnosis or therapy control. For the analysis of complex samples, different techniques have been established and are applied (Meyer and Stühler, 2007).

Mass spectrometry (MS) is a technique that has advanced much in the last years and is now frequently used for this type of differential study. Particularly liquid chromatography (LC) coupled to MS is highly suitable for the analysis of complex protein mixtures. In recent years, LC/MS has become a very sensitive technique that is well able to separate the different components of a protein mixture. Hence, it allows for both qualitative and quantitative analyses of protein mixtures. Due to easy automation, this is a suitable high-throughput method.

LC/MS is a 2D separation procedure. Before analyzing a protein sample with LC/MS, the proteins are digested into peptides. Then a first separation by LC takes place. Here, the digested sample is applied to the LC-column. The column has certain properties (e.g. hydrophobicity) so that the different peptides are retained according to their physiochemical properties. A solvent is pumped through the column such that some of the peptides elute from the column. By gradually changing the hydrophobicity of the solvent, peptides with different properties elute from the column at different times, leading to a separation of the peptide constituents of the protein sample. The time point at which a certain fraction of peptides elutes from the column is called the retention time (rt). The resulting fractions are then inserted online into the mass spectrometer. Here the second separation takes place. By ionizing the peptides and accelerating them to fly through a field-free tube, the peptides are separated by their mass-to-charge (m/z) ratio. The measured data can then be depicted in a 2D or 3D image. One axis depicts the rt, the second represents the m/z value and the third axis represents the intensity or quantity of a peptide. In 2D images, the intensity is depicted by color. The complete image is called a LC/MS run.

As data is first only available as an image, usually several preprocessing steps have to be performed to translate the image into analyzable data (Listgarten and Emili, 2005; Radulovic et al., 2004). For each LC/MS run, peak detection and quantification,
calibration and noise reduction are typically required preprocessing steps. Also, single peaks are sometimes combined according to isotopic distributions of peptides or different charge states.

Yet, especially for clinical studies, not only one run, but often hundreds of LC/MS runs need to be measured in order to find differences between, e.g., diseased and healthy patients. In this case, further preprocessing steps have to be performed to make the different runs comparable. For differential analysis, the different runs have to be combined as follows. Peaks representing the same peptide in different runs have to be matched and differences in their abundances have to be determined. The biggest obstacle here is the potential rt deviation between different LC/MS runs. While differences in m/z are typically small and the magnitude is known for the different mass spectrometers, the differences in rt can become quite large and often cannot be captured by linear shifts. Moreover, the rt deviations tend to become larger with increasing periods of time between the experiments and even larger between measurements obtained from different LC/MS instruments. As such problems will arise especially in large clinical studies, sophisticated alignment algorithms that are able to correct large and non-linear shifts in rt between runs are necessary for a reasonable analysis of such studies.

Many alignment methods have been proposed in the literature. A recent and detailed overview of alignment methods is given in Vandenbergobaert et al. (2008). Simple algorithms only use total ion chromatograms (TIC) for the alignment (Tomasi et al., 2004; van Nederkassel et al., 2006). For TIC, the total ion count of LC across rt is measured. Also, various algorithms have been implemented that work on the raw LC/MS runs (Christin et al., 2008; Listgarten et al., 2006). However, for huge clinical studies, algorithms that work on peak-picked LC/MS runs (Lange et al., 2007; Suits et al., 2008; Wang et al., 2007) should be preferred as TIC data does not use the MS part also containing information on the rt deviation. Also, alignment methods using raw LC/MS runs need huge amounts of computer memory due to the size of the input data, thus making it impractical to process hundreds of runs with these methods.

Some state-of-the-art algorithms working on peak-picked data (Lange et al., 2007) do not correct non-linear rt shifts. Yet, in real experiments, we have often observed such nonlinearities. Thus, we compare several algorithms working on peak-picked data where some correct non-linear rt shifts and some do not. We will show that correcting for such shifts yields better alignment results. For the evaluation, we use a simulation study and introduce several measures of performance that are suited to evaluate the quality of an alignment method.

The article is organized as follows. In Section 2, the methods employed in this article are explained. This section is divided into illustrations of the different alignment algorithms, a presentation of the simulation study used for the evaluation, and an introduction of suitable measures for the performance evaluation. In Section 3, the alignment algorithms are compared and their ability to correct for the non-linear shifts in rt is evaluated. Finally in Section 4, a discussion of the results is presented.

## 2 MATERIALS AND METHODS

To be able to analyze MS data, a couple of preprocessing steps have to be performed. The most important steps are peak detection, alignment, binning and normalization. Here we focus on the alignment of LC/MS runs.

<table>
<thead>
<tr>
<th>Table 1. Possible methods of alignment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Alignment method</strong></td>
</tr>
<tr>
<td>Linear regression (2.1.1)</td>
</tr>
<tr>
<td>Loess regression (2.1.2)</td>
</tr>
<tr>
<td>Local vectors (2.1.3)</td>
</tr>
</tbody>
</table>

Enclosed in parentheses are the sections where the individual methods are explained.

## 2.1 Alignment Algorithms

Often non-linear shifts in rt can be observed between different LC/MS runs. Hence, sophisticated alignment algorithms that correct for these distortions are very important for a correct analysis of such runs. We want to compare the performance of algorithms that especially take into account non-linear shifts in rt and those that do not. For this purpose, we apply three different algorithms. We first introduce two versions of an alignment algorithm based on statistical regression methods. The estimated rt deviation is once modeled by a linear function, thus only correcting linear shifts and once by a local linear function that also corrects non-linear shifts. This approach is particularly useful as the effect of the non-linear generalization compared with the linear version can directly be captured. A variant of the local version was previously introduced in Podwojski et al. (2008). An established alternative alignment algorithm correcting only linear shifts (Lange et al., 2007) is based on affine transformations, which can also be interpreted as shifting vectors. This algorithm was shown to be highly competitive to other freely available alignment algorithms (Lange et al., 2008). We use the concept of these shifting vectors to derive a third alignment method that corrects non-linear shifts for further comparison. Table 1 presents an overview of the alignment abilities of the different methods used in this article.

### 2.1.1 Linear regression method

The pseudocode of the linear regression method is shown in Figure 1. The alignment method is a two-step procedure. In the first step, groups of peaks that are easy to align are detected using cluster analysis. In a second step, these groups are then used for estimating the rt deviation line for each run with a linear regression, which only estimates global trends. Hence, we refer to this type of alignment as a ‘global’ alignment procedure. We implemented this algorithm in the statistical programming language R (R Development Core Team, 2008).

For this algorithm, the raw LC/MS runs need to be preprocessed. In each run the different peaks need to be detected, and for each peak the corresponding rt, m/z ratio and intensity or volume have to be recorded. This data is needed as input for the alignment algorithm. The peaks of all n runs to be aligned are first combined into one map. Next, slightly overlapping and fixed mass windows are laid over the whole rt domain of the combined map.

The clustering step of the algorithm works as follows. In each mass window, the p peaks with the highest intensities are picked. The idea behind this is that every type of organism has a number of proteins that are present in every individual with usually very high abundance. Those proteins are involved in the basic functions of a cell and are called housekeeping proteins. Such proteins can usually be found in every complex protein mixture like serum or urine. Due to their high abundance, the corresponding peptides can be measured very accurately in all runs and can thus be used as first anchor points for an alignment. Pairwise distances between the p peaks from the combined map are calculated according to the absolute distance in mass measured in p.p.m. together with penalties for a high deviation in rt or intensity. The mass of a peptide is well defined and can be measured with high accuracy depending on the type of mass spectrometer. Hence, the mass is a good indicator for peaks from different runs representing the same peptide. On the other hand, the rt of a peak depends on the hydrophobicity of the
Preliminaries

combine all n LC/MS runs
build overlapping mass-windows across combined runs

1. Cluster Analysis

for each mass-window do
use p peaks with highest intensities
calculate distance matrix of pairs of peaks \((j, h)\)
\[
d_{j,h} = \begin{cases} 
\text{diff}([\text{mass}]), & \text{if } \text{diff}(rt) < k_1 \land \\
\text{diff}[\text{log}_{10}(\text{intensity})] < k_2 \\
\infty, & \text{if } \text{diff}(rt) \geq k_1 \lor \\
\text{diff}[\text{log}_{10}(\text{intensity})] \geq k_2 
\end{cases}
\]
hierarchical average linkage cluster analysis
cut cluster-tree at mass accuracy \(\Delta_m\)
if \(n_{\text{dup}} < \text{threshold}_1\) \land \(n_{\text{miss}} < \text{threshold}_2\) then
cluster is ‘well-behaved’
delete duplicated ‘well-behaved’ clusters
for each ‘well-behaved’ cluster do
\(rt = \text{median}(rt)\)
for each peak \(i\) do
\(\text{dev}_i = rt_i - rt\)

2. Regression

for each run \(s\) do
take only peaks from ‘well-behaved’ clusters
fit regression line \(\text{dev}_{s,i} = a_s + b_s \cdot rt_i\)
by minimizing \(\sum (\text{dev}_{i} - \text{dev}_{s,i})^2\)

Correction

for each run \(s\) do
for each peak \(i\) do
\(rt_{\text{cor},i} = rt_i - \text{dev}_{s,i}\)

Fig. 1. Pseudocode of linear regression alignment algorithm.

corresponding peptide. If the rt of two peaks is far apart, the corresponding peptides will not be the same, as they have different chemical properties.

Also, the amount of a housekeeping protein should be roughly the same in each of our algorithm. In analogy to their approach, the median rt for each of the remaining ‘well-behaved’ clusters is calculated as well as the deviation of each single peak in the cluster from this corresponding median.

In the second step of the algorithm, the peaks from the ‘well-behaved’ clusters are now used for estimating the rt deviation of each peak. For each LC/MS run, the rt and the calculated rt deviation of each peak in a ‘well-behaved’ cluster are employed to fit a linear regression line, where the retention time is the explanatory variable and the rt deviation is the response variable. The regression line fitted to the peaks from the ‘well-behaved’ clusters can now be taken to correct the rt deviation for all peaks of the run. This is done by subtracting the estimated differences over the whole rt domain.

2.1.2 Local linear regression method

Linear regression is only able to fit global linear trends. The commonly seen non-linear shifts in rt between different LC/MS runs can, hence, not be corrected. However, there exist regression methods that are able to cope with such shifts. The well-established locally weighted scatterplot smoother (loess) (Cleveland, 1992, p. 309–376) is such a regression procedure.

The fundamental concept of loess regression is to estimate the regression function at each point separately. At every fixed point \(x\), only points close to this point enter into the calculation of the regression function. Typically, points enter with higher weight the closer they are to the point \(x\). The subset of points with positive weight can be characterized by the span, i.e. the percentage of all data points to be used for each local fit. Each regression function is fit with weighted least squares. A popular weight function is the tri-cube function (Hastie et al., 2001, p. 168).

A crucial parameter in loess regression is the span. This parameter determines the smoothness of the curve. A very small span takes into account only a small number of points at each rt and thus results in a jagged curve, while a large span will lead to a very smooth curve. A small span will result in high goodness-of-fit but due to high-model complexity it is also prone to overfitting. A large span on the other hand might not reproduce all local trends and hence result in a low goodness-of-fit.

Choosing the right span is a typical model selection problem. Hurvich et al. (1998) proposed an improved Akaike Information Criterion (AIC) for an optimal selection of the span. Generally, AIC tries to find an optimal trade-off between the goodness-of-fit of a model and its complexity by minimizing a weighted sum of goodness-of-fit and a measure for model complexity. However, it is known that under certain conditions AIC tends to prefer models with high complexity. A criterion very similar to the AIC but without the drawback of preferring models with higher complexity is the so-called Bayesian Information Criterion (BIC) (Schwarz, 1978). The two criteria only differ in the way the model complexity is penalized.

For the local linear regression alignment method, we again first employ the clustering procedure to derive ‘well-behaved’ clusters. Similar to Smith et al. (2006), loess regression replaces the linear regression to estimate the retention-time deviations in the second step. A linear polynomial weighted by the tri-cube weight function is used in the loess procedure. However, differing from Smith et al. (2006) the optimal span is chosen with the BIC for each run individually. This way, the local linear regression method adaptively corrects non-linear shifts in retention time between runs. We call this a ‘local’ alignment procedure.

2.1.3 Shifting vector method

A state-of-the-art algorithm working on peak-picked data is described by Lange et al. (2007). This algorithm does not correct non-linear rt shifts. The algorithm determines an affine transformation between two runs, which can also be seen as a shifting vector. The parameters of the affine transformation used to align one run to another (master) run are derived with so-called pose clustering. First, the shifts between all possible pairs of peaks from the two runs are calculated and recorded in a hash table.

Each of the shifts describes a possible solution for the parameters of the affine transformation. The true transformation parameters are then expected to be found most often among all shifts, and thus will cluster within the set of all shifts. This cluster can be detected with pose clustering.
In our comparison, the concept of shifting vectors is used for an alignment algorithm that corrects nonlinear rt deviations. The shifts between all possible pairs of peaks from a run and a master run are calculated. At each rt point only the recorded shifts corresponding to close-by peaks in the run are used to calculate a local transformation. This is done by calculating a weighted average of the possible shifts, where shifts coming from peaks close to the current retention time get higher weight. Thus, this is a ‘local’ alignment algorithm correcting non-linear shifts.

### 2.2 Simulation study

#### 2.2.1 Generation of simulated datasets

When the performance of different preprocessing algorithms is to be evaluated, the general problem with LC/MS runs is that the true locations of the peaks and the true abundances of the corresponding peptides are not known. Neither is it known, whether two peaks from different runs represent the same peptide. While the first problem can be avoided in the evaluation of alignment algorithms by using the same peak picking algorithm before each of the alignment methods and defining this as the truth, the second problem is still immanent. A popular solution is to take only single proteins or mixtures of only few known proteins where a theoretical digest can be made and used as a reference. The problem with this approach is that a sample with very few proteins is often easy to align due to the only few peaks that are present. If the algorithm is originally designed to cope with complex protein mixtures, a meaningful evaluation cannot be done with such data. A possibility to deal with real complex mixtures on the other hand would be to perform MS/MS experiments and identify the peaks in the run. Once the peptide corresponding to a peak is known, it can also be assessed if the peptide has been aligned correctly. Unfortunately, MS/MS measurements can only be obtained from a very small subset of the peaks in an LC/MS experiment. As the majority of the peaks cannot be measured, it may occur that for one peptide an MS/MS measurement is present for only a subset of the LC/MS runs. In this case, it is unknown if that peptide is not present in the other LC/MS runs or if just no MS/MS measurement was made of that specific peptide.

As both of these procedures to obtain data for performance evaluation have their drawbacks, we use simulated datasets in this study. For simulated data it is known in advance, which peaks from different runs represent the same peptide and this information can be used for an objective evaluation.

The simulation procedure is based on *Escherichia Coli* samples analyzed by LC/MS in an electrospray ionization (ESI) quadrupole time-of-flight (Q-TOF) mass spectrometer connected online with a nanoflow high pressure liquid chromatography (HPLC) including the reversed phase column and a 2 cm precolumn. Subsequent peak detection was performed. To simulate a real complex sample, like it would be expected from measurements of blood or urine, we combined five of the peak detected LC/MS runs from the real experiment. This resulted in a complete run with more than 35,000 peaks. This run was defined as the truth that contains all the peptides originally local, i.e. non-linear shifts, typical deviation curves are added to each run. The global shift is a time offset added to the whole run. For the first run while others might hardly ever be detected. Thus, we simulated 20 LC/MS runs for each simulated dataset based on this complete and true run.

First, for each of the 20 simulated runs, a subset of the 35,000 peaks to be chosen. In the simulation, we select peaks with high abundance with higher probability than those with low abundance. The probability for a peak to be included in the run rises with its abundance. Furthermore, the probabilities are adapted such that on average about 10,000 peaks are chosen for each simulated LC/MS run.

For simulating realistic rt deviations both local and global shifts are added to each run. The global shift is a time offset added to the whole run. For the local, i.e. non-linear shifts, typical deviation curves are added to each run.

The typical curves are derived from an experiment where a single protein (phosphorylase b) was measured ten times on the same type of LC/MS machinery to obtain ten LC/MS runs. Due to the small amount of peptides, peptides can easily be matched across the different runs. Thus, the necessary shifts can be estimated by loess regression and displayed with a rt deviation curve. Experts easily assess the correctness of these curves through visual inspection. Further typical curves were derived by reflection and dilations of the original 10 curves. It is also possible to derive typical random errors from this experiment that remain after an optimal alignment. A typical curve is thus added to each run.

Now, random errors are added to the three measurements (i.e. intensity, m/z ratio and rt) of each of the peaks to mimic typical measurement errors in LC/MS experiments. For intensity, both an individual error for each peak and a total error for the whole run are added. Errors for m/z are chosen to mimic the mass precision of a mass spectrometer. The random errors for rt are taken from the experiment with phosphorylase b.

During the whole process of variation of the different simulated runs, a numbering of the peaks ensures that an allocation of the peaks across the simulated runs in one dataset is always possible.

#### 2.2.2 Choice of parameters

In this simulation study, we simulate two datasets for the evaluation of the different alignment methods. For both datasets rt deviations are simulated to be maximally 3.5 min. The difference between the datasets lies in mass accuracy. In the first dataset, mass accuracy is simulated to be 100 p.p.m., in the second 10 p.p.m. The first dataset is harder to align, since the larger deviation in m/z makes many peaks ambiguous. The second dataset with mass accuracy of only several p.p.m. mimics the high quality of modern state-of-the-art mass spectrometers used in LC/MS.

For the regression alignment methods several parameters have to be set.

Primary parameters controlling the cluster analysis are \(k_1\), \(k_2\) and \(\Delta_m\), These parameters have to be set by the user depending on the LC/MS data. \(k_1\) has to be set to the largest expected rt deviation between LC/MS runs. The \(\Delta_m\) has to be set to the expected mass accuracy. The \(k_2\) is a criterion for exclusion and not very critical. We set this parameter to 1. This way, peaks whose intensities differ by factor 10 are not put into the same cluster. The influence of the two critical parameters \(k_1\) and \(\Delta_m\) will be a major aspect of our simulation study.

Secondary parameters controlling the choice of ‘well-behaved’ clusters are \(p\), \(\text{threshold}_1\) and \(\text{threshold}_2\). We generally choose \(p\) to be 400. This way, the cluster analyses run in reasonable time while at the same time many ‘well-behaved’ clusters are found. For very noisy LC/MS runs the amount can be increased, yet this will also increase the running time of the algorithm. We set \(\text{threshold}_1\) to 0 and \(\text{threshold}_2\) to 8. The first threshold should always be handled very strictly, while the second threshold has to be increased with increasing numbers of LC/MS runs. We found, that a choice of \(\sim n\) yields good results, when the LC/MS runs are not too noisy. With huge noise in the data, this threshold could be increased in order to still find enough ‘well-behaved’ clusters. However, LC/MS runs with much larger noise than in this simulation study in practice would be detected as bad LC/MS runs and excluded from an analysis.

### 2.3 Quality measures for alignment algorithms

To assess the performance of the different algorithms, we introduce several measures for performance evaluation. In contrast to real data it is known which peaks across the simulated runs represent the same peptide. While this information is blinded during the alignment procedure we are now able to use it for the performance assessment of the different algorithms.

#### 2.3.1 Pairwise comparisons

A first possibility to assess the quality of an alignment is to compare pairs of runs. For each pair of runs, all peaks present in both runs are used to calculate the ‘Median Retention Time Difference’ (MedRD) between run \(i\) and \(j\), which is defined by

\[
\text{MedRD}_{i,j} = \text{median}(rt_{ij0} - rt_{ij}),
\]

where \(rt_{ij0}\) is the vector of rt from run \(i\) of peaks that are also present in run \(j\). Analogously, the ‘Mean Retention Time Difference’ of run \(i\) and \(j\) (\(\text{MeanRD}_{i,j}\)) is derived by replacing the median with the mean in
Equation (1). The values of all possible pairs of runs can be combined to a matrix MedRD or MeanRD, respectively. The closer the individual values in this matrix are to zero the better is the alignment, as then there is no remaining systematic shift in rt.

However, the magnitude of the remaining shifts cannot be measured well with these quantities. For this purpose the ‘Median Absolute Retention Time Difference’ of run i and j can be calculated as

\[ \text{MedARD}_{ij} = \text{median}(|rt_{ij} - rt_{ij}|). \] (2)

Again, the ‘Mean Absolute Retention Time Difference’ of run i and j (MeanARD_{i}) can also be derived and the values can be combined to matrices MedARD and MeanARD. The smaller the individual values are, the lower the remaining shifts in rt are, and hence the better is the alignment between the corresponding pair of runs.

The pairwise comparisons especially help to evaluate, if there are any particular runs that could not be aligned well with an alignment procedure, and hence allow to compare the quality of different alignment procedures on the basis of single runs.

2.3.2 Multiple comparisons Besides the pairwise comparisons, we also want to assess the quality for the alignment procedure across all runs simultaneously. Therefore, for each peak h, that is present in at least two of the runs, the ‘Standard Deviation of Retention Time (SDRT)’ is defined by

\[ SDRT_{h} = \sqrt{\text{var}(rt_{h})}, \] (3)

where \( I_{h} \) is the vector of indices of runs that contain peak h and \( rt_{h} \) is the corresponding vector of rt. Note that due to the simulation process, a certain peak may only be present in a subset of the runs, and hence the vector of \( I_{h} \) may have different lengths for different peaks. The vector SDRT for all peaks h present in at least two runs can either be analyzed visually with boxplots or histograms or with a summarizing measure of location. Suitable measures include the ‘Median Standard Deviation of Retention Time’, defined by

\[ \text{MedSDRT} = \text{median}(SDRT), \] (4)

and the ‘Mean Standard Deviation of Retention Time’ (MeanSDRT).

An alternative is to derive the ‘Maximum Distance of Retention Time’ for each peak h, that is present in at least two runs, by

\[ MDRT_{h} = \max(rt_{h}) - \min(rt_{h}). \] (5)

The ‘Median Maximum Distance of Retention Time’ (MedMDRT) or ‘Mean Maximum Distance of Retention Time’ (MeanMDRT) are again suitable measures for the quality assessment of the complete alignment procedure.

3 RESULTS AND DISCUSSION

For the assessment of the different alignment methods, we use the two different datasets simulated with the procedure described in Section 2.2. We use the first dataset to assess the two regression methods and the robustness of the procedures with respect to user-specified parameter settings. The two critical parameters are the expected mass accuracy and maximal rt deviation between runs. We will henceforth call these two user-specified parameters ‘parameter setting’. We run the two alignment procedures both with the ‘true’ setting of 3.5 min and 100 p.p.m. and with smaller and larger values for the rt deviation and for the mass accuracy in order to mimic possible misjudgments of a user. The usage of the first dataset, which is harder to align, will ensure the ability to detect even slight differences in the alignment accuracy of different parameter settings.

For each of the settings both regression methods find a large number of ‘well-behaved’ clusters evenly spread across the rt range. Typical fitted curves derived with the two regression alignment methods are shown in Figure 2. The alignment curves for further runs look similar (see supplementary Material). The quality of the alignments is assessed with the measures introduced in Section 2.3. Due to the simulation procedure, most peaks are only present in a subset of the 20 simulated runs. However, with over 28,000 peaks being present in at least two runs, enough peaks can be used to calculate the different measures of performance. In Table 2 the SDRT is shown for all different parameter settings. It can be seen that the algorithms are very robust to the different settings in mass accuracy. Only when the user choice of rt deviation is much smaller (1 min) than the truth, MedSDRT for the linear regression method becomes 6 times as high as for the ‘true’ setting and for the loess regression becomes 6 times as high. The procedures thus are very robust to the parameter settings, as long as rt deviation is not largely underestimated by the user.

Comparing the two methods, the non-linear method consistently gives better alignment results. Except for the 1 min rt deviation parameter setting, the MedSDRT are more than 62% better for the loess regression method than for the linear regression method. MDRT shows the same results (data not shown).

In order to analyze why a large underestimation of rt deviation leads to worse alignment results, we consider the measures for pairwise comparisons between runs. In Figure 3, results of MedARD are shown. We see four outliers (runs 1, 4, 12 and 15) in the 1 min parameter setting. This means, that the corresponding runs have not been aligned correctly. These runs have large ‘true’ deviations in retention time. Parameter settings with larger rt deviations than the ‘true’ ones give similar results of MedARD as the true parameter settings (results not shown). Thus, the user should rather choose larger rt deviations than expected in order to obtain a good alignment.

We use the second dataset to compare the regression alignment methods to the shifting vector method. The dataset is very close to typical data from LC/MS experiments with good mass accuracy, and thus the results will be directly transferable to real LC/MS experiments. We already showed that parameter settings deviating...
Table 2. SDRT of peaks across all runs with 100 p.p.m. mass accuracy

<table>
<thead>
<tr>
<th>User-chosen parameter settings</th>
<th>Regression method</th>
<th>Minimum</th>
<th>First quartile</th>
<th>MedSDRT</th>
<th>MeanSDRT</th>
<th>Third quartile</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before alignment</td>
<td>–</td>
<td>7.043e-05</td>
<td>0.3608</td>
<td>0.4815</td>
<td>0.4730</td>
<td>0.5775</td>
<td>1.9340</td>
</tr>
<tr>
<td>80 p.p.m.; 3.5 min</td>
<td>Linear</td>
<td>1.876e-05</td>
<td>0.0371</td>
<td>0.0571</td>
<td>0.0778</td>
<td>0.0847</td>
<td>1.1920</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>1.252e-06</td>
<td>0.0249</td>
<td>0.0354</td>
<td>0.0566</td>
<td>0.0499</td>
<td>1.1400</td>
</tr>
<tr>
<td>100 p.p.m.; 1 min</td>
<td>Linear</td>
<td>1.966e-05</td>
<td>0.0896</td>
<td>0.2012</td>
<td>0.1997</td>
<td>0.2789</td>
<td>1.4820</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>9.433e-06</td>
<td>0.0662</td>
<td>0.2203</td>
<td>0.2282</td>
<td>0.3492</td>
<td>1.5650</td>
</tr>
<tr>
<td>100 p.p.m.; 2 min</td>
<td>Linear</td>
<td>8.061e-06</td>
<td>0.0375</td>
<td>0.0577</td>
<td>0.0782</td>
<td>0.0856</td>
<td>1.1970</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>3.026e-06</td>
<td>0.0255</td>
<td>0.0361</td>
<td>0.0568</td>
<td>0.0505</td>
<td>1.1460</td>
</tr>
<tr>
<td>100 p.p.m.; 3.5 min*</td>
<td>Linear</td>
<td>9.355e-06</td>
<td>0.0371</td>
<td>0.0571</td>
<td>0.0778</td>
<td>0.0848</td>
<td>1.1970</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>1.850e-05</td>
<td>0.0251</td>
<td>0.0356</td>
<td>0.0568</td>
<td>0.0500</td>
<td>1.1460</td>
</tr>
<tr>
<td>100 p.p.m.; 5 min</td>
<td>Linear</td>
<td>5.374e-07</td>
<td>0.0376</td>
<td>0.0575</td>
<td>0.0781</td>
<td>0.0851</td>
<td>1.1910</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>6.124e-06</td>
<td>0.0254</td>
<td>0.0359</td>
<td>0.0570</td>
<td>0.0504</td>
<td>1.1450</td>
</tr>
<tr>
<td>100 p.p.m.; 10 min</td>
<td>Linear</td>
<td>1.088e-05</td>
<td>0.0382</td>
<td>0.0589</td>
<td>0.0789</td>
<td>0.0867</td>
<td>1.2010</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>3.613e-06</td>
<td>0.0279</td>
<td>0.0390</td>
<td>0.0596</td>
<td>0.0539</td>
<td>1.1680</td>
</tr>
<tr>
<td>120 p.p.m.; 3.5 min</td>
<td>Linear</td>
<td>5.798e-07</td>
<td>0.0371</td>
<td>0.0572</td>
<td>0.0778</td>
<td>0.0850</td>
<td>1.1970</td>
</tr>
<tr>
<td></td>
<td>Loess</td>
<td>1.061e-06</td>
<td>0.0251</td>
<td>0.0356</td>
<td>0.0568</td>
<td>0.0500</td>
<td>1.1480</td>
</tr>
</tbody>
</table>

*The ‘true’ setting is highlighted in bold face.

Fig. 3. Heatmap of MedARD matrix for dataset with 100 p.p.m. mass accuracy. The color at position \((i,j)\) represents the value of \(\text{MedARD}_{ij}\), i.e. dark boxes represent large deviations. Top: Loess regression alignment with maximal rt deviation set to 1 min. Bottom: Loess regression alignment with maximal rt deviation set to 3.5 min.

only slightly from the ‘true’ settings do not have an effect on the alignment. Thus, only the ‘true’ parameter settings are considered in this part of the evaluation. The results of the different alignment methods are shown in Table 3 and Figure 4. Again the alignment algorithms that incorporate non-linear rt shifts outperform the linear regression method. The two non-linear alignment methods give comparable alignment accuracies. Pairwise comparisons show that the linear regression alignment method is not able to align all runs equally well (results not shown). This leads to larger values for SDRT and MDRT.

4 CONCLUSIONS

We have compared different alignment algorithms for peak-picked data that either correct non-linear shifts in rt or that only correct linear shifts. Especially, we evaluated one algorithm that is available both in a linear and in a non-linear version. Thus, a direct comparison is possible and is not influenced by differences between alignment algorithms regarding other aspects. A simulation study was employed to evaluate the alignment algorithms. The simulated data exhibited typical non-linear rt shifts as they are often seen in real LC/MS experiments. The advantage of simulated data is that it is known which peaks represent the same peptide across runs. We introduced several measures of performance evaluation that are suitable for simulated data and can be used to compare the results of different algorithms. The non-linear alignment algorithm results in better alignment of runs, even when the non-linear deviations are only small. Another non-linear approach, the shifting vector algorithm, shows that these results are consistent. Hence, correcting non-linear shifts in rt is crucial, independent of other properties of the algorithm.

A next step for the evaluation of performance of alignment algorithms is to use real LC/MS data. Lange et al. (2008) have already evaluated several freely available alignment algorithms on real LC/MS data. They used MS/MS experiments that enable the identification of single peptides in LC/MS runs. However, to compare the peptides across runs, a binning procedure is needed after the alignment that combines peptides across runs.
Table 3. SDRT of peaks across all runs with 10 p.p.m. mass accuracy

<table>
<thead>
<tr>
<th>Alignment method</th>
<th>Minimum</th>
<th>First quartile</th>
<th>MedSDRT</th>
<th>MeanSDRT</th>
<th>Third quartile</th>
<th>maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before alignment</td>
<td>2.965e-04</td>
<td>0.3769</td>
<td>0.5043</td>
<td>0.4960</td>
<td>0.6065</td>
<td>1.7570</td>
</tr>
<tr>
<td>Linear regression</td>
<td>1.005e-05</td>
<td>0.0366</td>
<td>0.0567</td>
<td>0.0773</td>
<td>0.0840</td>
<td>0.8001</td>
</tr>
<tr>
<td>Loess regression</td>
<td>5.692e-06</td>
<td>0.0243</td>
<td>0.0344</td>
<td>0.0555</td>
<td>0.0486</td>
<td>0.7108</td>
</tr>
<tr>
<td>Shifting vector</td>
<td>5.869e-06</td>
<td>0.0226</td>
<td>0.0326</td>
<td>0.0538</td>
<td>0.0467</td>
<td>0.7060</td>
</tr>
</tbody>
</table>

Fig. 4. Boxplots of MDRT for dataset with 10 p.p.m. mass accuracy aligned with different alignment methods.

Lange et al. (2008) introduced measures for performance evaluation that take the combined procedure of alignment and binning into account. They could show that their combined procedure of alignment and binning is highly competitive to many other freely available algorithms. However, with this type of performance evaluation the resulting errors cannot be allocated to the different parts of the alignment and binning procedure anymore. A good binning procedure is able to cope with some of the remaining variations in rt remaining after an alignment. However, the better an alignment is, the easier it will be for a following binning procedure to achieve sensible results. Thus, it can be expected that algorithms correcting non-linear shifts in rt will in general yield better overall results than their non-linear counterparts.

ACKNOWLEDGEMENTS

We thank Sebastian Link and Birgit Korte from the Medizinisches Proteom-Center (MPC) and Carsten Bäßmann and his group from Bruker Daltonics for providing the real LC/MS datasets used in the simulation study.


Conflict of Interest: none declared.

REFERENCES