SIMA: Simultaneous Multiple Alignment of LC/MS Peak Lists

Björn Voss[1,‡], Michael Hanselmann[1,‡], Bernhard Y. Renard[1,†], Martin S. Lindner[1], Ullrich Köthe[1], Marc Kirchner[1,¶], Fred A. Hamprecht[1,⋆]

[1] Heidelberg Collaboratory for Image Processing (HCI), Interdisciplinary Center for Scientific Computing (IWR), University of Heidelberg, Germany. [†] Current address: Department of Computational Medicine. The Institute for Translational Oncology and Immunology, Mainz, Germany. [¶] Current address: Proteomics Center, Departments of Pathology, Children's Hospital Boston and Harvard Medical School, Boston, MA, USA.

[‡] Authors contributed equally. [⋆] Corresponding author, fred.hamprecht@iwr.uni-heidelberg.de.

Abstract

Motivation: Alignment of multiple liquid chromatography/mass spectrometry (LC/MS) experiments is a necessity today, which arises from the need for biological and technical repeats. Due to limits in sampling frequency and poor reproducibility of retention times, current LC systems suffer from missing observations and nonlinear distortions of the retention times across runs. Existing approaches for peak correspondence estimation focus almost exclusively on solving the *pairwise* alignment problem, yielding straightforward but suboptimal results for *multiple* alignment problems.

Results: We propose SIMA, a novel automated procedure for alignment of peak lists from multiple LC/MS runs. SIMA combines hierarchical pairwise correspondence estimation with *simultaneous* alignment and *global* retention time correction. It employs a tailored multidimensional kernel function and a procedure based on maximum likelihood estimation to find the retention time distortion function that best fits the observed data. SIMA does not require a dedicated reference spectrum, is robust with regard to outliers, needs only two intuitive parameters, and naturally incorporates incomplete correspondence information. In a comparison with 7 alternative methods on 4 different datasets, we show that SIMA yields competitive and superior performance on real-world data.

SIMA Software: Free binaries and free C++ source code are available from our website http://hci.iwr.uni-heidelberg.de/MIP/Software.

1 Introduction

Recent developments in liquid chromatography/mass spectrometry (LC/MS) have afforded insight into the dynamics of biological systems at unprecedented levels of detail. High-resolution MS–based protein identification and quantitative MS are now established methodologies in fields as diverse as proteomics (Aebersold and Mann, 2003), glycomics (Zaia, 2010), lipidomics (Shevchenko and Simons, 2010) and metabolomics (Dettmer *et al.*, 2007).

Robust Alignment of LC/MS Experiments. Current LC/MS experiments often investigate complex biological systems over a set of different environmental conditions and/or time-courses. The associated data are routinely split into multiple fractions and acquired in technical and biological replicates, yielding tens to hundreds of LC/MS runs. Each of these runs delivers a snapshot of the system of interest and to enable their joint analysis, the common components in different measurements need to be related to each other. In practical LC/MS applications, two major factors complicate the determination of component correspondences across multiple runs: (i) the limited reproducibility attained on LC systems which gives rise to nonlinear distortions of the retention time domain; and (ii) the limited sampling frequency inherent to data-driven MS/MS acquisition as a notorious cause for missing observations. To obtain quantitative estimates or to increase peptide identification rates over a series of experiments, LC/MS data analysis frameworks (Mueller *et al.*, 2007; Khan *et al.*, 2009) rely on accurate mass and retention time alignment to propagate correspondence information between runs, experiments and samples. Accounting for LC distortions is a necessary prerequisite for such cross-experiment inference (America and Cordewener, 2008; Podwojski *et al.*, 2009).

Although numerous pairwise alignment methods have been proposed, the question of *simultaneous* alignment of multiple datasets is still a particularly challenging task: as the number of potential correspondences grows exponentially, false initial multiple correspondence estimates are more likely, and estimation procedures for the associated warping functions need to be robust to potential outliers. Even more, in the light of practical application, multiple alignment methods should naturally cope with incomplete correspondences where peaks are only observed in a subset of runs and no obvious missing value imputation strategy is available.

Types of LC/MS Alignment Algorithms. Published alignment algorithms work on different representations: either peaks extracted from raw measurements, or the raw measurements themselves (Prakash *et al.*, 2006; Vandenbogaert *et al.*, 2008; Clifford *et al.*, 2009). This contribution focuses on the alignment of sparse sets of samples, i.e. peaks $\boldsymbol{p_i} = [(m/z)_i, (rt)_i, z_i]$, which lie in a three-dimensional feature space (ion mass-to-charge ratio, ion elution time, and ion charge) as proposed in (Cox and Mann, 2008; Lange *et al.*, 2008; Khan *et al.*, 2009). Existing approaches can be divided into three categories:

1. Alignment based on a static reference, where all observed measurements are aligned to a single reference peak list (Zhang *et al.*, 2005; Bellew *et al.*, 2006; Lange *et al.*, 2007, 2008; Sturm *et al.*, 2008). Because these approaches single out one measured reference run, they perform well if there is at least one run of exceptional quality. Peaks that are not present in the reference cannot be used for alignment. This can be a substantial drawback, especially in low signal-to-noise ratio (SNR) situations or if suboptimal reproducibility is an issue.

2. Complete pairwise correspondence-based alignment, where the pairwise distances between all observed measurements in all runs yield pairwise alignments (Li *et al.*, 2005). Based on these correspondence pairs, global correspondence groups are computed by iteratively linking similar peaks. Although this approach overcomes the single reference problem, it is computationally expensive since it requires the calculation of all similarity measurements between all extracted peaks and performs a retention time correction in each iteration.

3. Hierarchical progressive alignment, where a similarity measure based on peak distances determines the merging sequence between different peak lists (Prakash *et al.*, 2006; Mueller *et al.*, 2007). The algorithm starts with an arbitrary (e.g. the most complete) list and subsequently merges the most similar lists until all correspondences are computed. Like in pairwise alignment, the retention times are corrected in each step which is is suboptimal (Smith *et al.*, 2006). Few methods exist that work without a similarity measure (Pluskal *et al.*, 2010).

In all categories, existing multiple alignment methods are straightforward extensions of pairwise alignments (Mueller *et al.*, 2007; Lange *et al.*, 2007, 2008; Khan *et al.*, 2009). While some of them use heuristics to deal with peaks that are not present in all available peak lists, i.e. missing correspondences, others completely discard incomplete correspondence information. However, the probability that a peak is visible in *all* peak lists decreases with increasing numbers of runs that have to be aligned. Simultaneous correction of all peak lists is superior but rarely considered (Smith *et al.*, 2006).

Simultaneous Multiple LC/MS Alignment. We propose SIMA, a novel approach that performs a single global retention time correction based on the multiple correspondence information obtained from all peak lists and naturally deals with missing correspondences. SIMA: (i) uses a pairwise greedy hierarchical strategy to determine all (potentially incomplete) correspondences across D peak lists (without performing a retention time correction in each step; section 2.1); and then (ii) uses a kernel density estimation type nonparametric approach to simultaneously work on all correspondence groups and derive a D-dimensional *retention time ridge*. Its highest path

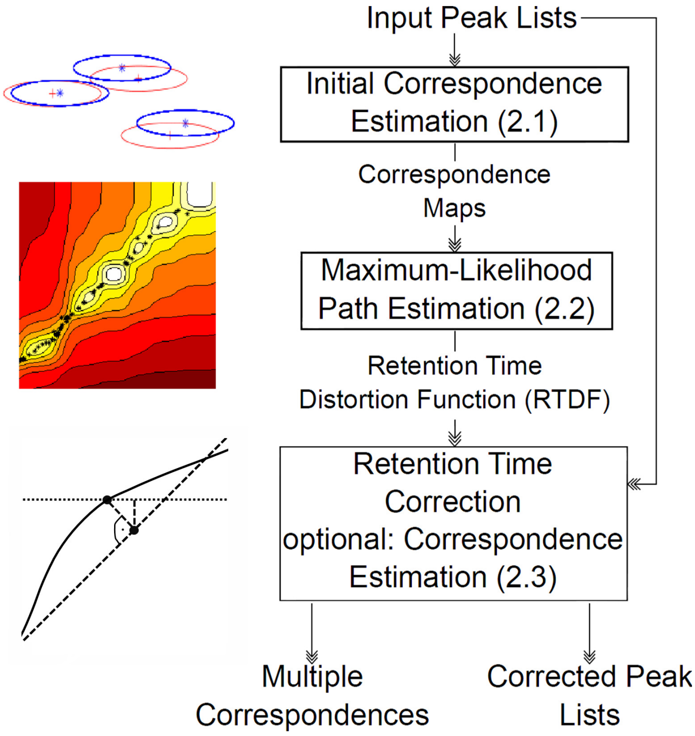
Input Peak Lists
Initial Correspondence Estimation (2.1)
Correspondence Maps
Maximum-Likelihood Path Estimation (2.2)
Retention Time Distortion Function (RTDF)
Retention Time Correction optional: Correspondence Estimation (2.3)
Multiple Correspondences
Corrected Peak Lists

Figure 1: SIMA workflow: Starting from a set of D LC/MS peak lists, SIMA conducts an initial correspondence estimation that yields groups of corresponding peaks (section 2.1). Based on these groups, the method calculates the retention time distortion function that is most likely to explain the observed retention time differences across the D peak lists (section 2.2). Given this function, all retention time deviations are corrected, and peak correspondences are reestimated (optional) (section 2.3).

is found by maximum likelihood estimation and approximates the global *retention time distortion function* that describes retention time shifts across all runs (section 2.2). Finally, (iii) it uses this function to correct the individual peak lists for retention time shifts and optionally performs a second hierarchical correspondence estimation (section 2.3). Step (ii) relies on a customized kernel that is inspired by *signal maps* (Prakash *et al.*, 2006) and that has specifically been tailored to make direct use of complete *and* incomplete correspondence information.

The remainder of this contribution is organized as follows: section 2 introduces the proposed workflow (see fig. 1) and its mathematical framework. Section 3 describes the experimental setup and error statistics used to judge SIMA performance and section 4 reports and discusses the outcomes. We end with conclusions in section 5.

2 Methods

2.1 Correspondence Estimation

The correspondence estimation is based on: (i) a measure for estimating the distance of peaks from two different peak lists, (ii) an algorithm for establishing peak correspondence pairs based on this measure, (iii) a distance measure for quantifying the dissimilarity of two peak lists based on their peak correspondences, and (iv) a hierarchical iteration scheme that successively combines the sparse individual peak lists into a more complete master peak list, while storing all established peak correspondences.

Notation. Let $\mathcal{P} = \{P_d\}, d = 1, \ldots, D$, be the set of the D LC/MS peak lists to be aligned. The dth peak list P_d comprises $|P_d|$ peaks, $P_d = \{\boldsymbol{p_{d,1}}, ..., \boldsymbol{p_{d,|P_d|}}\}$, and each peak $\boldsymbol{p_{d,i}} = [(m/z)_{d,i}, (rt)_{d,i}, z_{d,i}]$ is described by its mass over charge position (m/z), retention time (rt), and charge state z. To simplify notation, we discard the index indicating the membership of a peak to a peak list throughout the remainder of the derivations. Scalars are printed in standard font, vectors in bold.

Distance Measure for Peak Correspondence Estimation. The definition of an adequate peak distance measure is fundamental for identifying peak correspondences between different LC/MS runs. We quantify the distance between two peaks $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$ by the diagonal thresholded squared Mahalanobis distance ϕ given by

$$\phi(\boldsymbol{p_i}, \boldsymbol{p_j}) = \Psi\left((\boldsymbol{p_i} - \boldsymbol{p_j}) * W * (\boldsymbol{p_i} - \boldsymbol{p_j})'\right) \tag{1}$$

where we define $\Psi(\psi) = \psi$ for $0 \leq \psi \leq 1$, $\Psi(\psi) = \infty$ for $\psi > 1$, and the weight matrix W as $W = diag^{-1}(T^2_{(m/z)}, T^2_{(rt)}, T^2_z)$. $T_{(m/z)}$ and $T_{(rt)}$ are user-defined threshold parameters for the upper bounds on (m/z) and (rt) shift tolerance. In practice, their choice depends on measurement precision and is determined by the experimental instrument setup. Choosing very small values for T_z disallows correspondences between peaks with different charge states (default). If reliable charge state information is not available, deviations in charge state may be allowed by using larger values for T_z. Furthermore, W may easily be adapted to also take other features like intensity differences into account (cf. Supplementary Material A). In the two-dimensional $[(m/z), (rt)]$ domain, eq. (1) yields elliptical equidistance lines within the feasible area in which peaks may correspond to each other (cf. Supplementary Material B).

Establishing Correspondence Groups. Given $\phi(\cdot, \cdot)$, the problem of finding correspondences between peaks from two peak lists P_d and P_e can efficiently be solved by an algorithm that is best known for solving the “stable marriage” problem (Gale and Shapley, 1962). Initially designed for graph-matching problems, this method computes an optimal matching between elements from two disjunct sets, such that peak pairs with small distances are preferred. The resulting set $\mathcal{F}_{de}$ contains all peak correspondences of P_d and P_e, that is each peak pair $(i, j) \in \mathcal{F}_{de}$ contains exactly one peak from both P_d and P_e. Note that the two peaks forming a pair may differ in (m/z), (rt), and z. Some peaks might not find a partner.

Distance Measure for Peak List Dissimilarity. Given $\mathcal{F}_{de}$, the dissimilarity $\Phi(P_d, P_e)$ of two peak lists is obtained from averaging the finite truncated squared Mahalanobis distances of the assigned peak pairs. Denoting the peaks in pair (i, j) as $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$, we obtain

$$\Phi(P_d, P_e) = \frac{1}{|\mathcal{F}_{de}|} \sum_{(i,j) \in \mathcal{F}_{de}} \phi(\boldsymbol{p}_i, \boldsymbol{p_j}). \tag{2}$$

Hierarchical Correspondence Estimation. Rather than relying on one predetermined reference peak list for the alignment, we follow the idea of (Mueller *et al.*, 2007) and apply a greedy pairwise hierarchical iterative approach: This strategy eliminates the bias towards a single LC/MS run which occurs when using a reference peak list in the correspondence estimation. Nonetheless, SIMA can also use a single reference peak list to compute multiple peak correspondences. This may be beneficial if one of the peak lists is a priori known to be correct.

We successively combine the peak lists until all individual peak lists have been absorbed in one master peak list. In parallel, we construct *correspondence groups*, i.e. sets of peaks from the individual peak lists that match (see Supplementary Material C for details). Let $\mathcal{P}(t)$ be the set of peak lists that still have to be combined in iteration t. We initialize $\mathcal{P}(0) = \mathcal{P}$, that is with all original peak lists. During the course of the iterations, $\mathcal{P}(t)$ may contain both members of the set of original peak lists and/or representatives for previously combined lists. In each iteration, the two most similar peak lists according to eq. (2) are combined. Assume that in iteration step t these are $P_d \in \mathcal{P}(t)$ and $P_e \in \mathcal{P}(t)$. First, an empty peak list P_{de} is created, and all peaks that are unique to either P_d or P_e are added to it. Then, all peak correspondences $(i, j) \in \mathcal{F}_{de}$ between P_d and P_e are considered, and one representative peak is added for each correspondence pair. Its (rt) and (m/z) values are set to the mean over the respective values of all peaks that in previous iterations have contributed to the two merging peaks. Finally, P_d and P_e are removed from $\mathcal{P}(t)$ and replaced by the combined peak list P_{de} yielding $\mathcal{P}(t + 1)$. The correspondence groups are updated accordingly. After $D - 1$ steps, all peak lists have been combined, i.e. $|\mathcal{P}(D - 1)| = 1$.

We note that the greedy nature of the correspondence estimation allows for an efficient implementation. However, once merged, peaks cannot be split at later iterations which may suggest that the respective peaks should rather be kept separate. This typically does not pose a practical problem, since by setting the thresholds in W the experimentalist can control the merging behavior of peaks.

Let N be the number of resulting correspondence groups. After iteration $D-1$, the retention times associated with the peaks in the N groups are stored in a *retention time correspondence map* $C \in \mathbb{R}^{N \times D}$. More precisely, element $c_{n,d}$ of C holds the retention time of the peak in P_d that is a member in correspondence group n. If no such peak exists, the respective entry is flagged to indicate a *missing correspondence.* Each row vector $\boldsymbol{c_n} \in \mathbb{R}^D, n = 1, \ldots, N$, in C can be interpreted as a *correspondence point* in the D-dimensional *retention time space* (see figs. 4 and 6).

2.2 Maximum Likelihood Path Estimation

Retention Time Distortion Function. Define a *master time scale (MTS)* in the retention time space by equidistant sampling of the line of unit slope (that is the angle bisection line for $D = 2$). Further, define the *retention time distortion function (RTDF)* as the function that describes the retention time shifts for all peak lists. Its trajectory in retention time space thus explains the observed correspondence points $\boldsymbol{c_n}, n = 1, \ldots, N$. For a set of perfectly reproducible LC/MS measurements, all $\boldsymbol{c_n}$ lie on the line of unit slope such that the RTDF is equivalent to the latter. In practice, however, retention time measurements are subject to correlated noise and nonlinearly deviate from the ideal case. Given an estimate for the RTDF, the distortion of a peak can be identified by back-projecting its retention time onto the MTS (cf. fig. 5).

The behavior of the estimate should be in agreement with fundamental physical properties of LC/MS. We argue that a suitable estimation procedure should: (i) yield a RTDF that features a certain degree of *smoothness* since we do not expect abrupt changes in the elution process, (ii) ensure that the RTDF is *monotonous* such that the elution order is preserved across runs (Kirchner *et al.*, 2007), (iii) be *robust* with respect to measurement errors and naturally deal with outliers that may originate from incorrect matches in the initial correspondence estimation, and (iv) be *independent* of the input order of peak lists. We cast the problem of estimating the RTDF into a maximum likelihood (MFL) estimation framework, i.e. we find the RTDF as the function that best explains the correspondence points and at the same time fulfills the above constraints. To this end, we define a customized kernel that incorporates all prior assumptions. Its convolution with the (partially incomplete) observed peak correspondences yields a *retention time ridge.* The path along the highest points of this "height profile", i.e. the maximum likelihood path, is the RTDF that describes the nonlinear retention time distortions across the set of peak lists.

Constructing a Retention Time Ridge using a Sigmoid Kernel. Whereas smoothness

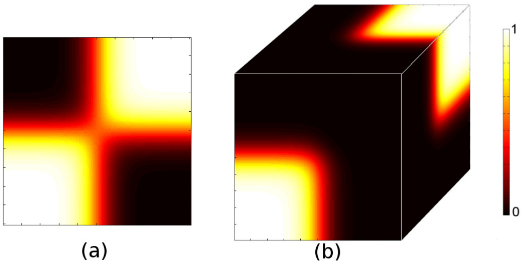

Figure 2: Plot of the sigmoid kernel defined in eq. (3) for (a) $D = 2$ and (b) $D = 3$. In both cases, the kernel has two preferred areas along the angle bisection line (light areas) and $(2^D - 2)$ "forbidden regions" (dark areas).

is guaranteed by employing a smooth kernel, the monotonicity of the ridge in retention times is more difficult to achieve. Figuratively speaking, the kernel (cf. fig. 2) should induce two preferred areas (lower left, upper right) and two "forbidden regions": When convolving it with all correspondence points in C, the response at a point $\boldsymbol{x} \in \mathbb{R}^2$ merely depends on the contribution of correspondence points whose retention times are not both lower *or* both higher than the ones of $\boldsymbol{x}$. This encourages the monotonicity of the ridge. Whereas, theoretically speaking, sets of correspondence points can be constructed that lead to paths that violate the monotonicity constraint, all point sets that can be considered a reasonable outcome of a set of LC/MS measurements yield a monotonous result. To make our approach more robust against measurement errors, an adaptive kernel parameter is used that controls the slope and hence the smoothness of the kernel and decreases the influence of outliers. Finally, our method is independent of the input order of the peak lists, since we use an equal kernel profile along all dimensions. By construction, all discussed properties carry over to higher dimensions.

More formally, let the kernel $K : \boldsymbol{x} \in \mathbb{R}^D \to (0, 1)$ be an outer product of sigmoid functions $k(x, \alpha) = 1/(1 + e^{-\alpha x})$ where

$$K(\boldsymbol{x}) = \prod_{d=1}^{D} k(x_d, \alpha) + \prod_{d=1}^{D} k(-x_d, \alpha). \tag{3}$$

Here, $x_d \in \mathbb{R}$ denotes the dth component of $\boldsymbol{x}$, and $\alpha \in \mathbb{R}^+$ is a parameter that controls the influence of the estimated correspondences in the *retention time space* (see fig. 4(c)). A higher value for α yields a steeper slope of the sigmoid function $k(\cdot, \cdot)$ and thus increases the local influence of the kernel (see below). For an arbitrary point $\boldsymbol{x} \in \mathbb{R}^D$ we obtain the cumulative kernel response

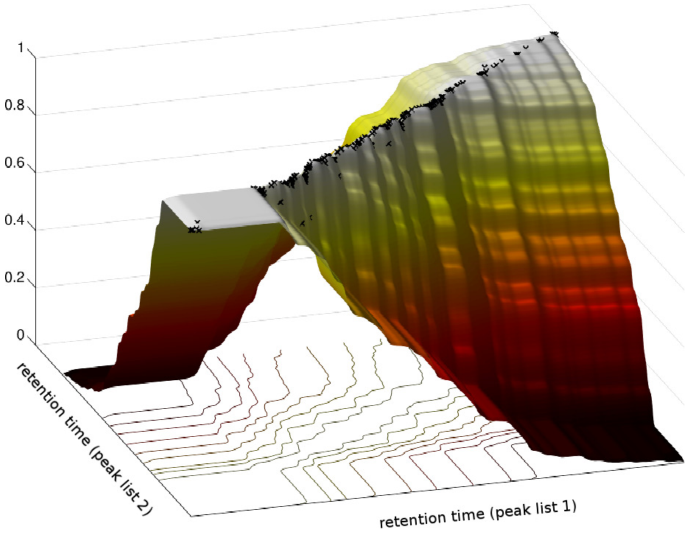

Figure 3: Three-dimensional plot of a retention time ridge formed by convolution of the correspondence points from two peak lists (dots) with the sigmoid kernel (cf. fig. 2(a)).

H with regard to all correspondence points $\boldsymbol{c_n}$ with the convolution

$$H(\boldsymbol{x}) = \frac{1}{\Omega}\sum_{n=1}^{N}\omega(\boldsymbol{c_n})K(\boldsymbol{x}-\boldsymbol{c_n}) = \tag{4}$$

$$\frac{1}{\Omega}\sum_{n=1}^{N}\omega(\boldsymbol{c_n})\left[\prod_{d=1}^{D}k(x_d - c_{n,d},\alpha) + \prod_{d=1}^{D}k(-x_d + c_{n,d},\alpha)\right] \tag{5}$$

where $\omega(\boldsymbol{c_n})$ is a weighting factor and $\Omega = \sum_{n=1}^{N}\omega(\boldsymbol{c_n})$. To deal with missing correspondences, we replace $k(\cdot,\cdot)$ with the adapted version $\tilde{k}(\cdot,\cdot)$, given by $\tilde{k}(x_d - c_{n,d},\alpha) = k(x_d - c_{n,d},\alpha)$ if $c_{n,d} \neq 0$ and 1 otherwise. In both cases, an analytical solution for the derivative $H'(\boldsymbol{x})$ exists (cf. Supplementary Material D-F). The intuition behind the adaption is as follows: In case of missing correspondences, the correspondence points degenerate to correspondence hyperplanes (cf. fig. 6). Although incomplete, these correspondences still constrain the RTDF in the orthogonal subspace within the retention time domain. We hence adapt the kernel to be uniform along the missing dimensions. This way, dimensions for which no correspondence information is available are simply ignored whereas all other information is used whenever available. Here, we use equal weights for all correspondence points, that is $\omega(\boldsymbol{c_n}) = 1\ \forall n = 1,\dots,N$. However, different weighting schemes are possible (cf. Supplementary Material G). An exemplary retention time ridge is shown in figs. 3 and 4.

ML-Estimation of the Retention Time Distortion Function. Using the sigmoidal kernel from above, the RTDF can be estimated by finding the points on the highest path through the retention time ridge that correspond to the time points of the MTS. We start with sampling a set of L equidistant points $\boldsymbol{y_l} \in \mathbb{R}^D, l = 1,\dots,L$ from the line of unit slope that constitute the

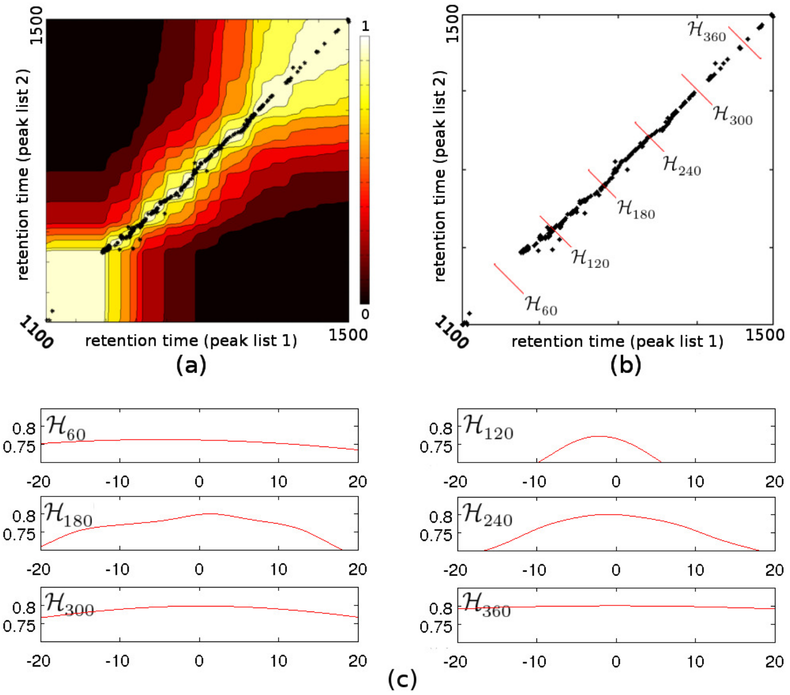

Figure 4: (a) Contour plot of the retention time ridge (cf. fig. 3). (b) Correspondence points (dots) and selected hyperplanes $\mathcal{H}_l$. (c) The intersections of the retention time ridge with these hyperplanes yield height profiles on which the gradient ascents are performed. The profiles of $\mathcal{H}_{120}$ and $\mathcal{H}_{240}$ show distinct bumps. There, the correspondence point density is high, leading to a larger kernel parameter α, i.e. a steeper kernel that emphasizes the influence of local points. The opposite holds, e.g., for $\mathcal{H}_{60}$.

MTS, and perform L gradient ascents toward the retention time ridge, starting from each of the $\boldsymbol{y}_l$. Each gradient ascent is performed on a subspace of $\mathbb{R}^D$ (cf. fig. 4). These subspaces $\mathcal{H}_l$ are hyperplanes perpendicular to the line of unit slope and contain the $\boldsymbol{y}_l$, that is, they are described by the normal vector $[1, \ldots, 1]$ and support vectors $\boldsymbol{y}_l$. The gradient ascents yield L points $\boldsymbol{x}_l \in \mathbb{R}^D$ whose piecewise linear interpolation approximates the RTDF. Mathematically, this procedure is similar to a maximum likelihood (ML) estimation where $K(\boldsymbol{x}) \in (0, 1)$ acts as a prior and we determine the $\boldsymbol{x}_l$ by

$$\underset{(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_L)}{\arg\max} \sum_{l=1}^{L} H(\boldsymbol{x}_l) \text{ subject to } \boldsymbol{x}_l \in \mathcal{H}_l. \tag{6}$$

Formulas for the normalized gradient directions along which we search for the maxima and for the update of the current estimate of $\boldsymbol{x}_l$ in iteration t are derived in Supplementary Data D. We propose to use an adaptive step size for the gradient ascents based on the Powell-Wolfe conditions (Powell, 1976) for increased robustness (Supplementary Material F). Note that when performing the gradient ascents, our method never computes the retention time ridge in its entirety but only evaluates $H(\boldsymbol{x})$ at a few points.

Adaptive Kernel Bandwidth. Naturally, the density of the correspondence points varies

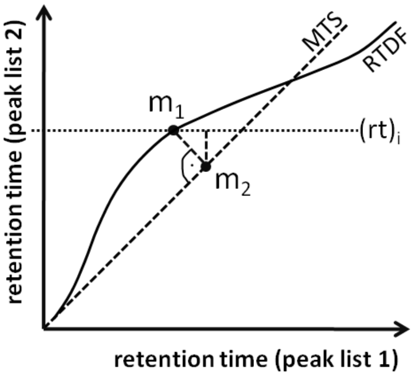

Figure 5: Retention time correction for $D = 2$ where we correct the retention time for peak $\boldsymbol{p_i} = [(m/z)_i, (rt)_i, z_i]$ in peak list P_2. We obtain $\boldsymbol{m_1}$ as the intersection of $y = (rt)_i$ with the *retention time distortion function* (RTDF). By back-projection onto the *master time scale* (MTS) we obtain $\boldsymbol{m_2}$. The difference in the vertical distance between $\boldsymbol{m_1}$ and $\boldsymbol{m_2}$ constitutes the amount by which the retention time of $\boldsymbol{p_i}$ needs to be shifted.

throughout retention time space. In such scenarios, the performance of nonparametric kernel methods can be improved by introducing an adaptive kernel bandwidth (Brockmann *et al.*, 1993). Thus, we locally adapt the kernel parameter α as follows: In low density areas, α is set to low values to achieve a higher robustness and avoid artefacts in the RTDF caused by single observations. In areas of high density, the smoothness of the RTDF is reduced by using larger values for α (cf. Supplementary Material H). This trades off bias and variance and reduces the overall error compared to a non-adaptive scheme.

2.3 Retention Time Correction

After performing the L gradient ascents and subsequent linear interpolation we obtain the piecewise linear RTDF which we use for correcting the retention times of the peaks observed in the D peak lists. Assume we want to correct the retention time for peak $\boldsymbol{p_i} = [(m/z)_i, (rt)_i, z_i]$ in peak list P_d. We first find $\boldsymbol{m_1}$, the intersection of the RTDF with the hyperplane given by support vector $\boldsymbol{a}$ with $a_d = (rt)_i$ and 0 elsewhere and normal vector $\boldsymbol{a}/||\boldsymbol{a}||$. We then identify that point on the line of unit slope that is closest to this intersection point ($\boldsymbol{m_2}$). The distance of those two points in the dth dimension constitutes the amount by which the retention time of $\boldsymbol{p_i}$ needs to be corrected. The procedure is repeated for all peaks and all peak lists (see fig. 5).

Second Correspondence Estimation. Correction of the retention times after the first iteration may give more correspondences. Hence, a second correspondence estimation (cf. 2.1) may yield a slightly more complete correspondence map C. However, practical impact is limited due to the overall robust nature of SIMA.

3 Experiments

Real-world Data. Lange et al. (Lange *et al.*, 2008) compared a total of six alignment algorithms on four publicly available proteomics ($P1$, $P2$) and metabolomics ($M1$, $M2$) datasets, including msInspect (May *et al.*, 2007), MZmine (Katajamaa and Oresic, 2005; Katajamaa *et al.*, 2006), OpenMS (Lange *et al.*, 2007; Sturm *et al.*, 2008), SpecArray (Li *et al.*, 2005), XAlign (Zhang *et al.*, 2005) and XCMS (Smith *et al.*, 2006). In addition, Pluskal et al. recently proposed MZmine 2 with RANSAC aligner (Pluskal *et al.*, 2010). We obtained the proteomics datasets used by Lange et al. (Lange *et al.*, 2008) from the Open Proteomics Database (OPD) (Prince *et al.*, 2004). Dataset $P1$ originates from an E. coli sample and contains two LC/MS runs of six fractions. Dataset $P2$ represents three LC/MS runs of five fractions of different cell states of Mycobacterium smegmatis. The datasets were analyzed by LC/MS/MS on an ESI ion trap mass spectrometer (ThermoFinnigan Dexa XP Plus), exported in centroid mode and preprocessed using TOPP tools (Kohlbacher *et al.*, 2007) resulting in a peak list of (m/z) and (rt) positions, which served as input for all alignment procedures. Each run of a fraction contains between 400 and 5800 peaks. Lange et al. (Lange *et al.*, 2008) optimized parameters for all approaches on the first fraction of each dataset and generated a partial ground truth by linking MS/MS search results from SEQUEST to the LC/MS spectra. A detailed description of all steps and the parameterization of the algorithms is given in Supplementary Material I and (Lange *et al.*, 2008).

For the metabolomics samples, Lange et al. (Lange *et al.*, 2008) analyzed Arabidopsis thaliana leaf tissue using an API QSTAR Pulsar i (Applied Biosystems/MDS Sciex) for the $M1$ dataset and a MicrOTOF-Q (Bruker Daltonics) for the $M2$ dataset resulting in 44 and 24 LC/MS spectra respectively. Peaks were identified using XCMS (Smith *et al.*, 2006) resulting in 4000 to 17600 data points per LC/MS spectrum. They generated ground truth by identifying highly confident peak groups which were reproducible over at least four runs and did not only have the same retention time, but also showed high correlation in their chromatographic peak shapes. Parameters were optimized on the complete datasets for all algorithms since no separate fractions were available. Even though SIMA can operate without a predetermined alignment order, it was run in the same starwise manner (i.e. using one predefined reference against which all remaining runs were aligned) that was used for the other algorithms and the ground truth generation in order to enable a fair comparison (Otherwise, SIMA might benefit from *not* using a potentially incomplete reference). Again we refer to Lange et al. (Lange *et al.*, 2008) and the supplementary materials for a detailed description and the parameterization of the algorithms.

Performance Measures. To measure the performance of an approach, we compute its precision (PR) and recall (RE). Precision is the fraction of correctly aligned peaks among all peaks

data	measure	ms-Inspect	MZ-mine	Open-MS	Spec-Array	X-Align	XCMS	MZ-mine 2 (RANSAC)	SIMA
$P1$	RE	0.66	0.85	0.93	0.70	0.88	0.81	**0.94**	0.92
	PR	0.50	0.89	0.93	0.70	0.88	0.80	**0.94**	**0.94**
	F	0.57	0.87	0.93	0.70	0.88	0.80	**0.94**	0.93
$P2$	RE	0.58	0.77	**0.83**	0.50	0.73	0.70	0.75	0.76
	PR	0.26	0.66	**0.72**	0.35	0.63	0.59	0.68	**0.72**
	F	0.36	0.71	**0.77**	0.41	0.67	0.64	0.71	0.74
$M1$	RE	0.27	0.89	0.87	-	0.88	**0.94**	0.91	0.92
	PR	0.46	0.74	0.69	-	0.70	0.70	0.74	**0.75**
	F	0.34	0.81	0.77	-	0.78	0.80	0.82	**0.83**
$M2$	RE	0.23	0.98	0.93	-	0.93	0.98	0.98	**0.99**
	PR	0.47	0.84	0.79	-	0.79	0.78	0.83	**0.84**
	F	0.31	0.90	0.85	-	0.85	0.87	0.90	**0.91**
All	RE	0.43	0.87	0.89	-	0.85	0.86	**0.90**	**0.90**
	PR	0.42	0.78	0.78	-	0.75	0.72	0.80	**0.81**
	F	0.39	0.82	0.83	-	0.80	0.78	0.84	**0.85**

Table 1: Comparison of the results of seven current alignment approaches with SIMA based on the datasets of the comparative studies by Lange (Lange *et al.*, 2008) and Pluskal (Pluskal *et al.*, 2010). Recall (RE), Precision (PR) and the F-measure (F) are reported as an average over various runs on two proteomics ($P1$, $P2$) and two metabolomics ($M1$, $M2$) datasets as well as an overall average of all datasets (*All*). Bold print highlights the overall best values for each dataset. MZ-mine 2 (RANSAC) and SIMA show the best overall recall, while SIMA features the highest values for precision and the F-measure.

aligned by one approach, $PR = \frac{\text{\# correctly aligned peaks}}{\text{\# aligned peaks}}$, whereas recall corresponds to the fraction of correctly aligned peaks by one approach among all correct peaks according to the ground truth, $RE = \frac{\text{\# correctly aligned peaks}}{\text{\# correct peaks}}$. To simplify comparison, we use the F-measure $F = \frac{2 \cdot PR \cdot RE}{PR+RE}$, which summarizes precision and recall value by computing their harmonic mean (Gay *et al.*, 2002).

4 Results and Discussion

SIMA Yields Competitive or Superior Results. In table 1, the results of the comparison are detailed. With regard to recall, SIMA and MZ-mine 2 (RANSAC) tie at the best performance with an average recall of 0.90. While MZ-mine 2 (RANSAC) and OpenMS feature better recall values for the $P1$ and $P2$ datasets, SIMA shows better recall performance on the $M1$ and $M2$ data. With regard to precision, i.e. the likelihood of results being correct, SIMA shows the best performance in all datasets with an average precision value of 0.81. This is also reflected in the F-measure, which combines recall and precision. Here, SIMA shows the best overall performance with an average value of 0.85. SIMA always is among the best two methods with respect to the F-measure ($P1$, $P2$) or even performs best ($M1$, $M2$). A complete list of the results on all fractions of all datasets is given in Supplementary Material J.

It is important to note that the comparison favors the algorithms that require a reference peak list: the ground truth generated by Lange et al. (Lange *et al.*, 2008) is based on the same reference run as used for the alignment. For independently generated ground truth results for any reference-spectrum based approach are bound to deteriorate since not all peaks present in the ground truth necessarily need to be in the reference peak list. The performance measurements of hierarchical approaches such as SIMA are not affected by this kind of ground truth generation.

SIMA is Especially Powerful when Aligning Numerous Spectra. The strength of the SIMA approach is particularly visible on the metabolomics ($M1$ and $M2$) datasets. Here, it outperforms the other methods with regard to precision as well as recall. The metabolomics datasets contain more LC/MS spectra (44 and 24, respectively) than the proteomics datasets (2 and 3, respectively) and, thus, also show significantly more missing correspondences. Further, visual inspection confirms that these datasets are less perturbed by noise and show a more characteristic structure, which benefits more from the nonlinear fitting of SIMA than the proteomics set, for which linear methods already show good results.

Exploiting Incomplete Correspondence Information is Feasible. Visual examples of SIMA alignment results are given in fig. 6 and in Supplementary Material K. Fig. 6 shows an (m/z) 500–800 subrange of the first three peak lists in the $M1$ dataset, in which 8 correspondence groups are complete, and incomplete information is available for an additional 14 groups (omitting single entry correspondence groups for visual clarity). The exploitation of partial correspondence groups yields valuable constraints for guiding the RTDF curve through retention time space and provides robustness in cases where single complete retention time observations show extreme values. The latter is especially prevalent with the obvious outliers present in the mass range (m/z) 800–1100, as illustrated in Supplementary Material K. SIMA uses all information available from the data by including missing correspondences and can thus base estimates on larger effective numbers of observations compared to other approaches (also cf. Supplementary Material K and L).

Hierarchical Correspondence Estimation Renders Distinguished Reference Spectra Oobsolete. SIMA eliminates the problem of selecting a distinguished reference spectrum or peak list, respectively. In practical applications, obvious reference candidates are neither easily obtained nor guaranteed to exist. Consequently, SIMA based alignment is not subject to a reference bias and independent of the peak list processing order.

SIMA is Robust with regard to Parameter Settings. Considering that for the proteomics datasets the first fraction was used for parameter optimization, it is interesting to observe that SIMA is not performing as well as, e.g., OpenMS on these fractions. Still, SIMA shows superior performance on the remaining fractions, for which the parameters identified on the first sections were used (cf. Supplementary Material J). This indicates that SIMA is not overly dependent

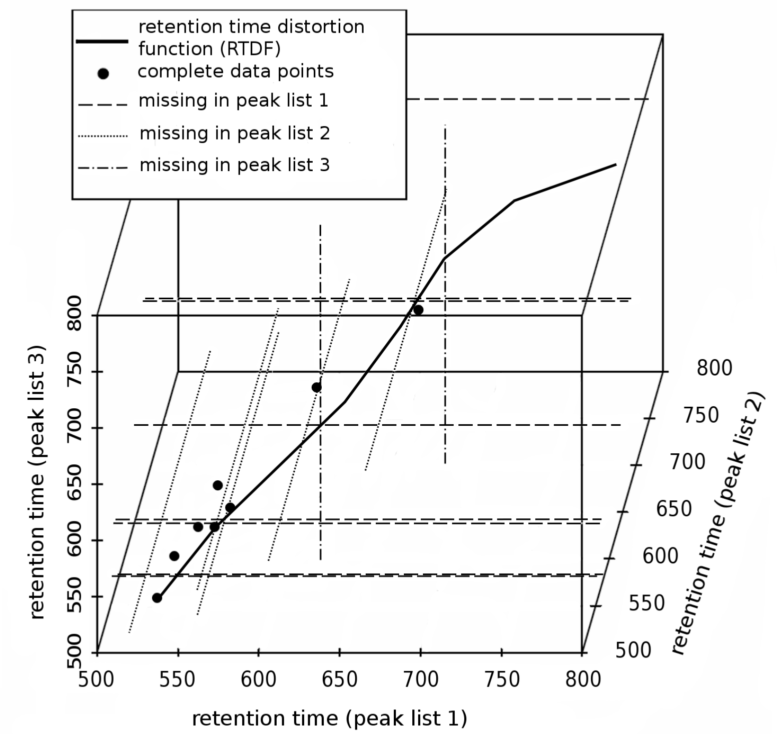

Figure 6: Visualization of the alignment approach on the peak lists of the first three spectra of the $M1$ dataset for the range of 500–800s in retention time: Only 8 peaks (dots) in this range could be matched in all peak lists, whereas 14 peaks were additionally available in two of the three peak lists only. Since these peaks contain information for two dimensions, but are non-informative for the third, they are displayed by straight lines parallel to the coordinate axis of the missing information. The computed *retention time distortion function* (red) still benefits from these straight lines as they help pinpointing its optimal path through this area of few observations.

on parameter settings since it does not benefit from the overfitting on the first fractions where parameters were adjusted to give optimal results, but also shows high quality results on datasets where parameter optimization was not performed. This can at least partially be explained by the fact that (given reliable charge state information) SIMA only requires two parameters in the matrix W for correspondence estimation, which in our experiments have proven robust to changes. Choosing a larger region for the correspondence estimation results in additional random data points for the kernel regression. However, as long as those additional points are unstructured, they do not bias the estimate for the RTDF, since our approach is robust to outliers. Choosing a smaller region results in fewer data points and additional missing correspondences, which can be handled as long as not all data points of a region are removed.

5 Conclusion

We introduced SIMA, a novel approach for the simultaneous alignment of multiple LC/MS peak lists. SIMA is specifically tailored to the problems arising from large-scale experiments where only few peaks are consistently present in all runs. Thus, in contrast to many competing algorithms, SIMA can naturally handle missing correspondences. In addition, it does not rely on a single, error-free reference run as basis for an alignment, but weights the inherent measurement errors of each run against each other.

SIMA requires only very limited user interaction, since it is robust with respect to its two parameters, the thresholds for the tolerated retention time and m/z difference between two peaks. Moreover, these parameters can typically directly be inferred from the expected measurement error in the experiment.

An experimental comparison on real-world proteomics and metabolomics data to seven state-of-the-art approaches demonstrates excellent performance of our method. While matching the recall of MZ-mine 2 (RANSAC), the best performing method from previous comparisons (Lange *et al.*, 2008; Pluskal *et al.*, 2010), it delivers the best precision and overall F-score values.

Conceptually, SIMA is not limited to the alignment of LC/MS data: By redefinition of the thresholded squared Mahalanobis distance function, it can easily be adapted to any time series with discrete events (cf. Supplementary Data A). SIMA is freely available from *http://hci.iwr.uni-heidelberg.de/MIP/Software.*

6 Acknowledgments

We gratefully acknowledge financial support by the DFG under grant no. HA4364/2-1 (M.H, B.Y.R.), the Robert Bosch GmbH (F.A.H.), and the DFG GRK-1114 (B.V.). We further like to thank Bernhard Kausler, Anna Kreshuk, Sebastian Boppel, and Xinghua Lou (all University of Heidelberg) for fruitful discussions, as well as both of our reviewers for constructive criticism and comments.

Supplementary Material

A: Adapting the Weight Matrix

In equation (1) of the manuscript, the diagonal thresholded squared Mahalanobis distance is defined. This distance relies on a weight matrix W, given by $W = diag^{-1}(T^2_{(m/z)}, T^2_{(rt)}, T^2_z)$. Note that mass-to-charge ratio (m/z), retention time (rt), and charge state (z) are *scale-invariant* peak features. As a consequence, SIMA can be applied to a broad range of data. For instance, it may even be used to align LC/MS measurements of samples that exhibit different levels of up- or down-regulation of certain proteins. In that case, scale-dependent features such as the difference in peak intensity are suboptimal.

Nevertheless, some applications may require additional (potentially scale-dependent) features. For instance, peak intensity might be useful if we can assume that the (relative) peak intensities are constant over different runs (innately or after proper normalization). Integration of additional peak features into SIMA is straightforward: We only need to redefine the weight matrix W as

$$W = diag^{-1}(T^2_{(m/z)}, T^2_{(rt)}, T^2_z, T^2_{(intensity)}) \tag{7}$$

where $T^2_{(intensity)}$ controls the maximum permissible deviation in peak intensity between two measurements (note that this parameter is scale-dependent). Thresholds on other peak features that may be available from the previously used peak picking routine can be added analogously.

B: Elliptical Equidistant Lines

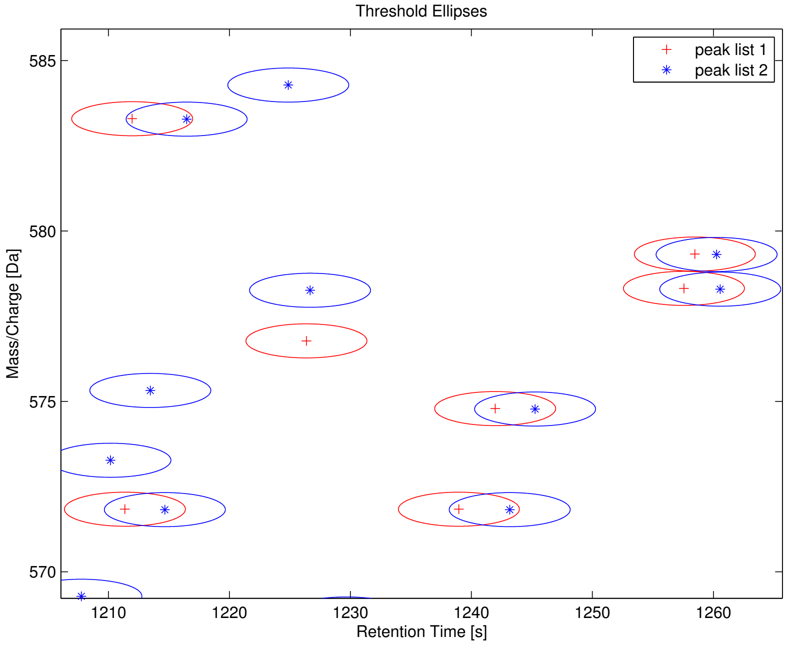

Figure 7: The figure shows peaks from two peak lists P_d (red crosses) and P_e (blue stars) in the two-dimensional $[(m/z),(rt)]$ domain. The threshold for the thresholded squared Mahalanobis distance θ between peaks can be visualized by equidistant lines which in this case correspond to ellipses centered at the peaks. A matching between peak $p \in P_d$ and $q \in P_e$ is feasible if both peaks are located within the two corresponding ellipses.

C: Storing the Correspondence Group Memberships

The hierarchical correspondence estimation described in the manuscript is an iterative procedure that constructs peak correspondence groups, i.e. sets of peaks from the D original peak lists that match each other. In the following, we describe the data structure that is used to protocol the memberships of the original peaks in these correspondence groups.

Let $\mathcal{G}(t)$ be the set of correspondence groups in iteration t where $\mathcal{G}(0) = \emptyset$. Assume that in iteration t we combine the two most similar peak lists P_d and P_e in the current set $\mathcal{P}(t)$ of peak lists that still have to be merged into a new peak list P_{de} that replaces the other two in $\mathcal{P}(t+1)$ (see manuscript for details). For each peak correspondence pair, a new peak is constructed and added to P_{de}. Let the peaks $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$ form one such pair (i, j) in the set of peak correspondences $\mathcal{F}_{de}$ of P_d and P_e. We store the correspondence of $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$ by updating $\mathcal{G}(t)$ in the following way:

- If neither $\boldsymbol{p_i}$ nor $\boldsymbol{p_j}$ are the result of a previous merging step, we create a new correspondence group $G_{\boldsymbol{p_i p_j}} = \{\boldsymbol{p_i}, \boldsymbol{p_j}\}$ and add it to $\mathcal{G}(t)$, that is $\mathcal{G}(t) = \mathcal{G}(t) \cup G_{\boldsymbol{p_i p_j}}$.

- If, in contrast, exactly one of these peaks was created by one or more merging steps in preceding iterations, $\mathcal{G}(t)$ is updated as follows. Without loss of generality, assume that $\boldsymbol{p_i}$ was created by merging two or more peaks, whereas $\boldsymbol{p_j}$ is one of the peaks in the original peak lists. Further assume that the peaks that have been absorbed into $\boldsymbol{p}$ constitute correspondence group $G_{\boldsymbol{p_i}} \in \mathcal{G}(t)$. We then merely set $G_{\boldsymbol{p_i}} = G_{\boldsymbol{p_i}} \cup \boldsymbol{p_j}$.

- If both peaks $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$ are the result of merging steps in preceding iterations, $\mathcal{G}(t)$ is updated as follows: Assume that the peaks that have been absorbed into $\boldsymbol{p_i}$ constitute correspondence group $G_{\boldsymbol{p_i}}$ and that the peaks that have been absorbed in $\boldsymbol{p_i}$ constitute correspondence group $G_{\boldsymbol{p_j}}$. We then set $G_{\boldsymbol{p_i p_j}} = G_{\boldsymbol{p_i}} \cup G_{\boldsymbol{p_j}}$ as well as $\mathcal{G}(t) = \mathcal{G}(t) \setminus (G_{\boldsymbol{p_i}} \cup G_{\boldsymbol{p_j}}) \cup G_{\boldsymbol{p_i p_j}}$.

These changes to $\mathcal{G}(t)$ are performed for all peak pairs in $\mathcal{F}_{de}$, yielding $\mathcal{G}(t + 1)$. After the last iteration of the hierarchical correspondence estimation $\mathcal{G}$ is used to construct the retention time correspondence map C as described in the manuscript.

D: Gradient of the Retention Time Ridge

The adaptive kernel $K : \boldsymbol{x} \in \mathbb{R}^D \rightarrow (0,1)$ is defined as the outer product of two sigmoid functions such that the estimated retention time distortion function fulfills the properties described in section 2.2 of the manuscript (monotonicity, order-independence, smoothness and robustness):

$$k(x,\alpha) = \frac{1}{1+e^{-\alpha x}}. \tag{8}$$

We note that $k(x,\alpha)$ is differentiable. It holds that

$$\begin{aligned}
\frac{\partial k(x,\alpha)}{\partial x} &= \frac{\partial}{\partial x}\frac{1}{1+e^{-\alpha x}} \\
&= -\frac{\alpha\ e^{-\alpha x}}{(1+e^{-\alpha x})^2} \\
&= \alpha\ \left(\frac{1+e^{-\alpha x}}{(1+e^{-\alpha x})^2} - \frac{1}{(1+e^{-\alpha x})^2}\right) \\
&= \frac{\alpha}{1+e^{-\alpha x}}(1 - \frac{1}{1+e^{-\alpha x}}) \\
&= \alpha\ k(x,\alpha)(1-k(x,\alpha)) \\
&= \alpha\ k(x,\alpha)k(-x,\alpha)
\end{aligned} \tag{9}$$

As described in section 2.2 of the manuscript, we perform gradient ascents on the retention time ridge H (cf. eq. (5) of the manuscript). More precisely, these gradient ascents are performed within the L hyperplanes given by the support vectors $\boldsymbol{y_l}, l = 1, \ldots, L$. Confer to Supplementary Material E for a more detailed description.

Given a single hyperplane $\mathcal{H}_l$ and iteration t we have to calculate the $(D-1)$-dimensional gradient

$\nabla g(x_{l(t),1}, \ldots, x_{l(t),D-1})$. In the following we discard index t to keep the notation uncluttered. It holds that

$$g(x_{l,1},\ldots,x_{l,D-1}) = \tilde{H}(x_{l,1},\ldots,x_{l,D-1},\vartheta(x_{l,1},\ldots,x_{l,D-1}))$$

$$= \tfrac{1}{\Omega}\textstyle\sum_{n=1}^{N}\omega(\boldsymbol{c_n})\Big[\underbrace{\prod_{d=1}^{D-1}\tilde{k}(x_d - c_{n,d},\alpha)}_{S_1}\cdot\underbrace{\tilde{k}(\vartheta(x_{l,1},\ldots,x_{l,D-1}) - c_{n,D},\alpha)}_{S_2} \tag{10}$$

$$+\underbrace{\prod_{d=1}^{D-1}\tilde{k}(-\left[x_d - c_{n,d}\right],\alpha)}_{S_3}\cdot\underbrace{\tilde{k}(-[\vartheta(x_{l,1},\ldots,x_{l,D-1}) - c_{n,D}],\alpha)}_{S_4}\Big].$$

Here, $\vartheta(\cdot)$ is defined as in Supplementary Material E. With eqs. (9) and (10) the components of the gradient can be obtained from

$$\nabla g(x_{l,1},\ldots,x^{l,D-1})_i = \frac{\partial\tilde{H}}{\partial x_i} = \frac{1}{\Omega}\sum_{n=1}^{N}\omega(\boldsymbol{c_n})\Big[S_1'\cdot S_2 + S_1\cdot S_2' + S_3'\cdot S_4 + S_4'\cdot S_3\Big] \tag{11}$$

where

$$\begin{aligned}
S_1' &= \alpha\ \tilde{k}(-x_{l,i}+c_{n,i},\alpha)\textstyle\prod_{d=1}^{D-1}\tilde{k}(x_d - c_{n,d},\alpha)\\
S_2' &= -\eta\ \alpha\ \tilde{k}(-\vartheta(x_{l,1},\ldots,x_{l,D-1})+c_{n,D},\alpha)\ \tilde{k}(\vartheta(x_{l,1},\ldots,x_{l,D-1})-c_{n,D},\alpha)\\
S_3' &= -\alpha\ \tilde{k}(x_{l,i}+c_{n,i},\alpha)\ \textstyle\prod_{d=1}^{D-1}\tilde{k}(-x_d - c_{n,d},\alpha)\\
S_4' &= \eta\ \alpha\ \tilde{k}(\vartheta(x_{l,1},\ldots,x_{l,D-1})+c_{n,D},\alpha)\ \tilde{k}(-\vartheta(x_{l,1},\ldots,x_{l,D-1})-c_{n,D},\alpha).
\end{aligned} \tag{12}$$

E: Formulas for the Gradient Ascent

Let η be the normal vector of the hyperplanes $\mathcal{H}_l, l = 1, \dots, L$ which is parallel to the line of unit slope, and $\boldsymbol{y_l} \in \mathbb{R}^D, l = 1, \dots, L$ be equidistant points on the line of unit slope that lie within the $\mathcal{H}_l$.

A point $\boldsymbol{x_l} \in \mathbb{R}^D$ lies on hyperplane $\mathcal{H}_l$ if $x_{l,D} = \vartheta(x_{l,1}, \dots, x_{l,D-1})$, where $x_{l,D}$ is the Dth component of vector $\boldsymbol{x_l}$ and $\vartheta : \mathbb{R}^{D-1} \to \mathbb{R}$ is given by the projection on the hyperplane

$$\vartheta(x_{l,1}, \dots, x_{l,D-1}) = \frac{1}{\eta_D} \left(\boldsymbol{y_l} \cdot \boldsymbol{\eta} - \sum_{d=1}^{D-1} x_{l,d} \eta_d \right). \tag{13}$$

For this choice of $x_{l,D}$, the plane equation $\boldsymbol{y_l} \cdot \boldsymbol{\eta} = \boldsymbol{x_l} \cdot \boldsymbol{\eta}$ holds. For every point $\boldsymbol{x_l} \in \mathcal{H}_l$ the height g of the retention time ridge can then be calculated using eq. 5 from the manuscript in $g : \mathbb{R}^{D-1} \to \mathbb{R} : g(x_{l,1}, \dots, x_{l,D-1}) = H(x_{l,1}, \dots, x_{l,D-1}, \vartheta(x_{l,1}, \dots, x_{l,D-1}))$.

In iteration step t of the gradient ascent with current estimate $\boldsymbol{x_l}(t)$ (where we set $\boldsymbol{x_l}(0) = \boldsymbol{y_l}$), the normalized gradient direction $\boldsymbol{\delta_l}(t)$ along which we search for the maximum is given by

$$\boldsymbol{\delta_l}(t) = \frac{\nabla g(x_{l,1}(t), \dots, x_{l,D-1}(t))}{||\nabla g(x_{l,1}(t), \dots, x_{l,D-1}(t))||}. \tag{14}$$

We further estimate a feasible step size $\Psi_l(t)$ with respect to the Powell-Wolfe conditions Powell (1976) which leads to a more robust detection of the maxima (see Supplementary Material F). The next estimate is then obtained from

$$\boldsymbol{x_l}(t+1) = \boldsymbol{x_l}(t) + \Psi_l(t)\boldsymbol{\delta_l}(t) \tag{15}$$

and the iterations are stopped as soon as $\Psi_l(t)$ falls below a user-defined threshold.

F: Gradient Ascent Step Size Estimation

The step size for the gradient ascent in step t is estimated with an iterative approach where the step sizes $\Psi_l(t)$ are forced to fulfill the Powell-Wolfe conditions Powell (1976). For given $\beta \in (0, 0.5)$, $\gamma \in (0.5, 1)$, $\boldsymbol{x_l}(t) \in \mathcal{H}_l$, $g(\boldsymbol{x_l}(t))$, $\nabla g(\boldsymbol{x_l}(t))$, and search direction $\boldsymbol{\delta_l}(t)$ the step size $\Psi_l(t)$ has to fulfill the following inequalities:

$$g(\boldsymbol{x_l}(t) + \Psi_l(t)\boldsymbol{\delta_l}(t)) \leq g(\boldsymbol{x_l}(t)) + \Psi_l(t)\beta\nabla g(\boldsymbol{x_l}(t))^T\boldsymbol{\delta_l}(t) \tag{16}$$

$$\nabla g(\boldsymbol{x_l}(t) + \Psi_l(t)\boldsymbol{\delta_l}(t))^T\boldsymbol{\delta_l}(t) \geq \gamma\nabla g(\boldsymbol{x_l}(t))^T\boldsymbol{\delta_l}(t) \tag{17}$$

Setting $G(\Psi_l(t)) := g(\boldsymbol{x_l}(t) + \Psi_l(t)\boldsymbol{\delta_l}(t))$ we can rewrite these inequalities as

$$G(\Psi_l(t)) \leq G(0) + \beta\Psi_l(t)G'(0) \tag{18}$$

and

$$G'(\Psi_l(t)) \geq \gamma G'(0). \tag{19}$$

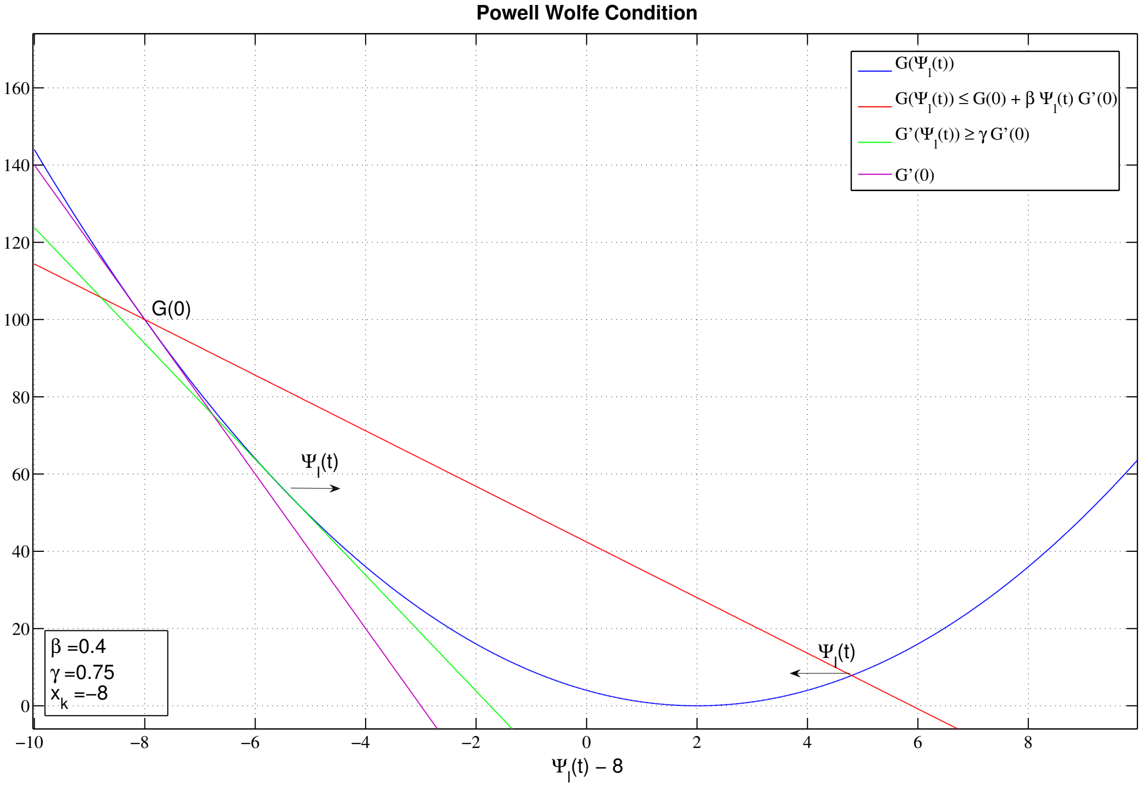

Figure 8: Powell–Wolfe conditions: The lower bound for the step size estimator is the point where the slope of $G(\Psi_l(t))$ is larger than $\gamma G(\Psi_l(0))$. The upper bound is the intersection of $G(\Psi_l(t))$ with the line $\beta\Psi_l(t)G'(\Psi_l(0))$.

The first inequality (16) describes an upper bound for the step size to avoid overly large steps in the gradient line search. This bound is given by the line with slope $\beta G'(0)$ that passes through the point $G(0)$ (see fig. 8). If the value of $G(\Psi_l(t))$ is above this line, $\Psi_l(k)$ is too large to fulfill the Powell–Wolfe condition. Values of $\Psi_l(t)$ for which the slope of $G(\Psi_l(t))$ is larger than $\gamma G'(0)$ are rejected as too small. We estimate the step size with an iterative algorithm that starts with step size $\Psi_l(t_1) = 1$ and adds or subtracts the value $\frac{\Psi_l(t)}{2}$ until the Wolfe condition is fulfilled. The result $\Psi_l(t_\Theta)$ that satisfies the Wolfe Condition is used as step size in the gradient ascent algorithm.

G: Assigning Weights to Correspondence Groups

In equations 5 and 6 of the manuscript, the influence of the correspondence points $\boldsymbol{c_n}$ is controlled by the weights $\omega(\boldsymbol{c_n})$. Different choices are possible. In the manuscript we have used a constant weight of 1 for all correspondence points. Alternatively, the $\boldsymbol{c_n}$ can be weighted according to their "completeness", that is $\omega(\boldsymbol{c_n}) = 1/|\boldsymbol{c_n}|$. Consequently, a correspondence group that comprises peaks from four LC/MS runs will have twice the influence of a correspondence point that comprises two peaks.

On one hand, weighting correspondence points according to their completeness seems to be more natural, since correspondence groups to which more measurements have contributed intuitively should receive higher weights. On the other hand, down-weighting of incomplete correspondence groups only seems to be appropriate if the missing correspondences do not arise from measurement inconsistencies or artifacts introduced by the peak picker. In that sense, it sometimes may be more beneficial to rely on more local but less complete correspondence groups than on far away but complete groups.

We have thus tested both strategies and have found that equal weights indeed work slightly better on the Lange et al. Lange *et al.* (2008) data which was used in our experiments (see section 3 and 4 of our manuscript). Results are summarized in table 2.

data	weighting scheme	RE	PR	F
$P1$	equal	**0.92**	**0.94**	**0.93**
	completeness	**0.92**	**0.94**	**0.93**
$P2$	equal	**0.76**	**0.72**	**0.74**
	completeness	**0.76**	0.70	0.72
$M1$	equal	**0.92**	**0.75**	**0.83**
	completeness	0.91	0.74	0.82
$M2$	equal	**0.99**	**0.84**	**0.91**
	completeness	0.98	**0.84**	0.90

Table 2: Performance on the Lange et al. Lange *et al.* (2008) datasets for equal weights for correspondence groups and weights according to their completeness. The former strategy performs slightly better.

H: Local Adaption of Kernel Parameter α

In the manuscript, we implicitly assume that the projections of the retention time ridge onto the hyperplanes $\mathcal{H}_l$ are convex for all $l = 1, \ldots, L$ such that we can find global optima. This assumption can be violated if the mean variation of the data points in a local neighborhood is large. We therefore propose to locally adjust the parameter α that determines the sphere of influence of the sigmoid kernel $k(x, \alpha)$ such that it is large where the data density is high and low otherwise.

The area of influence of $k(x, \alpha)$ is defined by the interval $I(k(x, \alpha), \xi) = \left[-\frac{\lambda}{2}, \frac{\lambda}{2}\right]$ which is centered at zero and has diameter λ for which

$$k(-\tfrac{\lambda}{2}, \alpha) - k(\tfrac{\lambda}{2}, \alpha) \quad = \quad \xi, \quad \xi \in (0, 1) \tag{20}$$

holds. It can be shown that the dependence between the diameter λ of the sphere of influence and the kernel parameter α is

$$\begin{aligned} \frac{1}{1+e^{\alpha\frac{\lambda}{2}}} - \frac{1}{1+e^{-\alpha\frac{\lambda}{2}}} &= \xi \\ \Rightarrow \frac{\lambda}{2} &= -\frac{ln(-\frac{\xi+1}{\xi-1})}{\alpha} \qquad (21) \\ \alpha &\sim \frac{1}{\lambda}. \end{aligned}$$

We propose to estimate the kernel parameter $\alpha(\boldsymbol{x}, C)$ for a given point $\boldsymbol{x} \in \mathbb{R}^D$ from the retention time correspondence map $C \in \mathbb{R}^{N \times D}$ introduced in section 2.1 of the manuscript. Here α is computed from the mean of the euclidean distances $|\cdot|_{rt}$ of the 20 nearest points (i.e. rows $\boldsymbol{c_n}$ of C) in the D-dimensional retention time space. Missing correspondences, represented by zero entries in c_n are ignored during the computation of $|\cdot|_{rt}$.

$$\alpha(\boldsymbol{x}, C) = \frac{1}{\frac{1}{20}\sum_{\boldsymbol{c_n} \in 20NN} |\boldsymbol{x} - \boldsymbol{c_n}|_{rt}} \tag{22}$$

Note that the adaptive kernel parameter α is defined by the reciprocal of the mean distance which can be seen as an estimate for the variance of the selected points in a local region. Consequently, α becomes larger in areas of high data density (that have low variance) and smaller in areas of low data density (that have higher variance). This ensures that the maximum likelihood estimation of the retention time distortion function is precise in all scenarios. The adaptive method should thus be preferred over methods using a global parameter α (also see Brockmann et al. Brockmann *et al.* (1993)).

Empirically, the result proved to be robust with respect to the exact number of nearest neighbors that were used in the computation. Note that selecting a larger number of neighbors increases the average distance of $\boldsymbol{x}$ to its neighbors. However, at the same time this also holds for all other points $\tilde{\boldsymbol{x}}$ in the retention time space. Thus, the algorithm is rather robust with respect to this parameter, which does not have to be tuned by the user and was fixed for all experiments.

I: Parameter Optimization

Parameters for the proteomics and metabolomics were optimized as described in Lange et al. Lange *et al.* (2008). For the proteomics datasets ($P1$ and $P2$), the first fraction of the dataset was used for parameter optimization. For the metabolomics datasets ($M1$ and $M2$), parameters were optimized on the complete datasets.

Using a grid search we identified the parameters in table 3 as optimal for the various datasets for the SIMA algorithm. Parameters for the other algorithms are described in Lange et al. Lange *et al.* (2008) and Pluskal et al. Pluskal *et al.* (2010).

parameter	$P1$	$P2$	$M1$	$M2$
$T_{(rt)}$	125	350	40	40
$T_{(m/z)}$	2.1	1.9	0.03	0.03

Table 3: Optimal parameters for the SIMA approach for the two proteomics and the two metabolomics datasets of Lange et al. Lange *et al.* (2008).

J: Detailed Results for the Proteomics Datasets

fraction	measure	ms-Inspect	MZ-mine	OpenMS	SpecArray	XAlign	XCMS	MZ-mine 2 (RANSAC)	SIMA
00	RE	0.52	0.75	**0.86**	0.61	0.82	0.62	**0.86**	0.83
	PR	0.38	0.81	**0.86**	0.61	0.82	0.58	**0.86**	**0.86**
	F	0.44	0.78	**0.86**	0.61	0.82	0.60	**0.86**	0.85
20	RE	0.56	0.87	0.92	0.62	0.85	0.81	0.93	**0.94**
	PR	0.45	0.88	0.92	0.62	0.85	0.80	0.93	**0.97**
	F	0.50	0.87	0.92	0.62	0.85	0.80	0.93	**0.95**
40	RE	0.63	0.87	**0.94**	0.75	0.87	0.81	**0.94**	0.91
	PR	0.48	0.90	**0.94**	0.75	0.87	0.80	**0.94**	**0.94**
	F	0.54	0.88	**0.94**	0.75	0.87	0.80	**0.94**	0.92
60	RE	0.73	0.79	0.96	0.71	0.87	0.78	**0.97**	0.92
	PR	0.54	0.84	0.96	0.71	0.87	0.75	**0.97**	0.94
	F	0.62	0.81	0.96	0.71	0.87	0.76	**0.97**	0.93
80	RE	0.70	0.92	0.96	0.74	0.90	0.89	**0.97**	0.96
	PR	0.57	0.94	0.96	0.74	0.90	0.88	0.97	**0.98**
	F	0.63	0.93	0.96	0.74	0.90	0.88	**0.97**	**0.97**
100	RE	0.82	0.92	0.94	0.77	**0.96**	**0.96**	**0.96**	0.95
	PR	0.56	0.94	0.94	0.77	**0.96**	**0.96**	**0.96**	**0.96**
	F	0.67	0.93	0.94	0.77	**0.96**	**0.96**	**0.97**	0.95
All	RE	0.66	0.85	0.93	0.70	0.88	0.81	**0.94**	0.92
	PR	0.50	0.89	0.93	0.70	0.88	0.80	**0.94**	**0.94**
	F	0.57	0.87	0.93	0.70	0.88	0.80	**0.94**	0.93

Table 4: Comparison of the results of seven current alignment approaches with SIMA based on the $P1$ dataset of Lange et al. Lange *et al.* (2008) and Pluskal et al. Pluskal *et al.* (2010). Recall (RE), Precision (PR) and the F-measure (F) are reported for each fraction of the dataset as well as an overall average of all fractions (*All*). MZ-mine 2 (RANSAC) shows the best overall recall, and MZ-mine 2 (RANSAC) and SIMA tie for the highest values for precision. Although SIMA was specifically designed for the alignment of many LC/MS runs, it performs second best on this pairwise problem.

fraction	measure	ms-Inspect	MZ-mine	OpenMS	SpecArray	XAlign	XCMS	MZ-mine 2 (RANSAC)	SIMA
00	RE	0.23	**0.77**	**0.77**	0.07	0.65	0.58	0.56	0.61
	PR	0.07	0.60	**0.65**	0.05	0.49	0.44	0.49	0.55
	F	0.11	0.67	**0.70**	0.06	0.56	0.50	0.52	0.58
20	RE	0.67	0.87	0.92	0.57	0.84	0.86	**0.93**	0.89
	PR	0.24	0.71	0.77	0.42	0.70	0.66	**0.78**	0.75
	F	0.35	0.78	0.84	0.48	0.76	0.75	**0.85**	0.82
40	RE	0.44	**0.79**	0.76	0.60	0.71	0.72	0.78	0.75
	PR	0.26	0.76	0.74	0.41	0.69	0.69	0.77	**0.81**
	F	0.33	0.77	0.75	0.49	0.70	0.70	0.77	**0.78**
80	RE	0.73	0.61	**0.80**	0.65	0.58	0.49	0.61	0.63
	PR	0.34	0.56	0.70	0.44	0.56	0.45	0.61	**0.74**
	F	0.46	0.58	**0.75**	0.52	0.57	0.47	0.61	0.68
100	RE	0.82	0.80	**0.90**	0.63	0.85	0.85	0.88	0.89
	PR	0.39	0.65	0.75	0.44	0.69	0.69	0.75	**0.77**
	F	0.53	0.72	**0.82**	0.52	0.76	0.76	0.81	**0.82**
All	RE	0.58	0.77	**0.83**	0.50	0.73	0.70	0.75	0.76
	PR	0.26	0.66	**0.72**	0.35	0.63	0.59	0.68	**0.72**
	F	0.36	0.71	**0.77**	0.41	0.67	0.64	0.71	0.74

Table 5: Comparison of the results of seven current alignment approaches with SIMA based on the $P2$ dataset of Lange et al. Lange *et al.* (2008) and Pluskal et a. Pluskal *et al.* (2010). Recall (RE), Precision (PR) and the F-measure (F) are reported for each fraction of the dataset as well as an overall average over all fractions (*All*). OpenMS shows the best overall recall, while OpenMS and SIMA tie for the highest values for precision. OpenMS shows the overall best value in the F-measure. OpenMS clearly outperforms our approach on the first fraction of the dataset, which was used for the parameter optimization, while OpenMS and SIMA show similar behavior on the remaining fractions.

K: Visualization of the Alignment Approach for the Range of 800–1100s

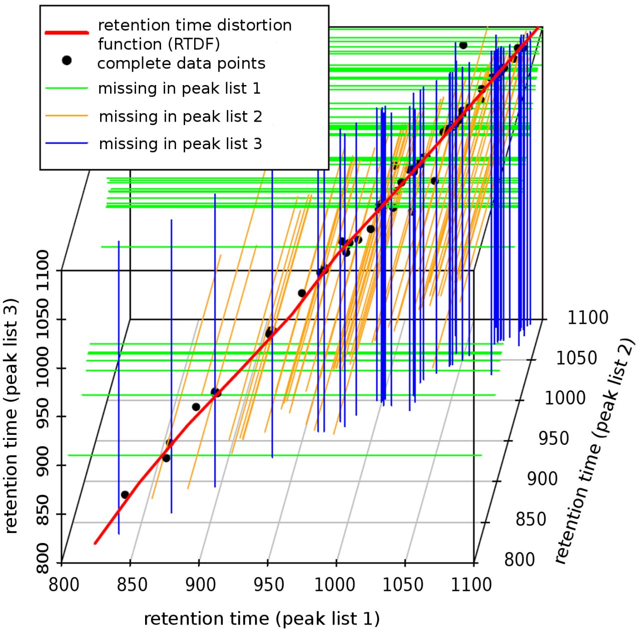

Figure 9: Visualization of the alignment approach on the peak lists of the first three spectra of the $M1$ dataset for the range of 800–1100s in retention time. Within this densely populated region with many data points with complete information (black dots) as well as incomplete information (straight lines) the *retention time distortion function* closely follows the bisecting line without being influenced by obvious outliers.

L: Histograms over Correspondence Groups

As table 6 and figure 10 show, many of the peaks can only be observed in few runs. Consequently, many correspondence groups are incomplete, especially in cases where many different runs have to be aligned. Discarding all these correspondence groups when aligning LC/MS datasets leads to a significant loss of information. In contrast to most competing alignment algorithms, SIMA can make direct use of incomplete correspondence information.

The overall numbers of correspondence groups for the sets $P1$, $P2$, $M1$, and $M2$ are 7139, 6669, 43696, and 73073. Note that correspondence groups of size one (single peaks) are not used for alignment.

correspondence group size	$P1$	$P2$
1	3670	1599
2	**6938**	5716
3	-	**6636**

Table 6: The table gives the total number of peaks in the two proteomics datasets $P1$ and $P2$. Peaks are arranged by the size of the correspondence group to which they were assigned. Note that $P1$ and $P2$ consist of two respectively three runs, so the maximum correspondence group sizes are two and three. Peaks that belong to complete groups are printed in bold. As it can be seen, many peaks are part of incomplete groups.

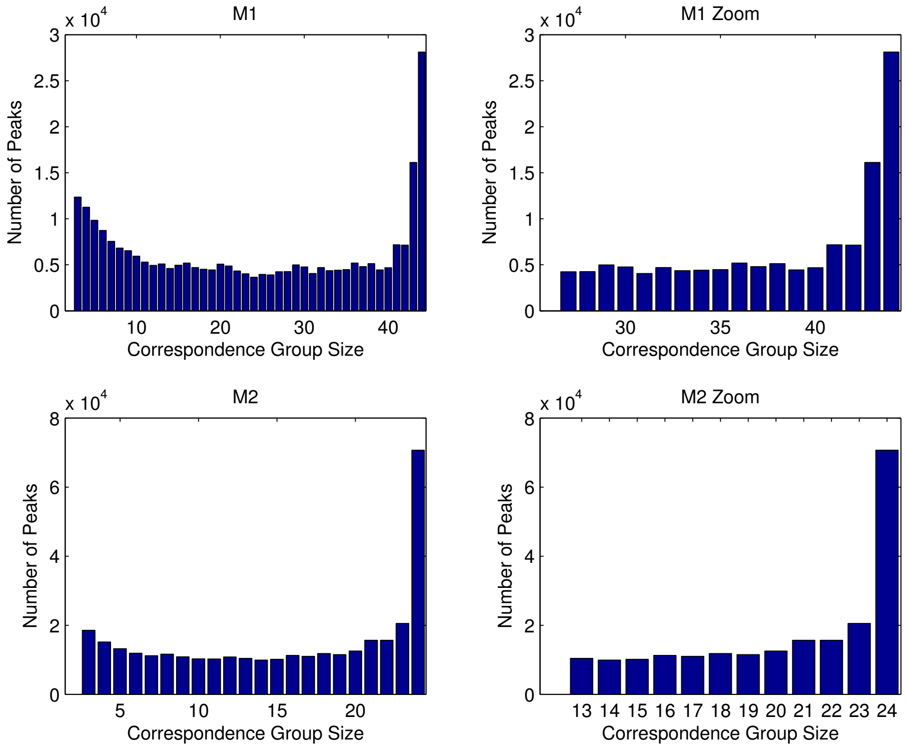

Figure 10: The figures on the left show the total number of peaks in the metabolomics datasets $M1$ and $M2$ (featuring 44 respectively 24 LC/MS runs). Again, peaks are arranged according to the correspondence group sizes to which they were assigned. The peak numbers for groups of sizes one (15683 and 29258) and two (27016 and 45552) are left out for improved visualization. On the right, the histogram over peaks that occur in at least 50% of all runs are given.

M: Performance in Case of Broad Eluting Peaks

As pointed out by one of our reviewers, broad eluting peaks might pose a challenging task for alignment algorithms that work on peak lists, if the peak picker returns several peaks $\boldsymbol{p_i}, i = 1, \ldots, M$ instead of one (that in reality all represent the same broad peak). This set of peaks may highly differ between runs, rendering correct peak matching a difficult task. Although we argue that this problem should rather be tackled by the peak picking software and not by the alignment tool which does not have access to the raw data (which may be needed to decide which peaks to merge), this problem may indeed arise in practice. It is thus interesting to discuss how SIMA performs in such settings and how its parameters may be tuned to obtain satisfactory results.

By selecting a large threshold $T_{(rt)}$, it can be ensured that even peak correspondences with large retention time deviations can be found such that the corresponding peaks can be matched. However, this may come at a high cost, since other peaks which, for instance, stem from different but close-by peaks may be wrongly matched due to the large threshold.

Thus, it may be more suitable to chose a smaller threshold $T_{(rt)}$ (e.g. according to the resolution of the instrument) to ensure that different peaks are not combined. As a result, the peaks $\boldsymbol{p_i}$ will not all be merged into the same correspondence group, but multiple groups will evolve instead, each of which will be incomplete. In contrast to other alignment approaches, SIMA can make direct use of incomplete correspondence information. It is thus likely that SIMA would still provide robust alignment results.

References

Aebersold, R. and Mann, M. (2003). Mass spectrometry-based proteomics. *Nature*, **422**(6928), 198–207.

America, A. H. P. and Cordewener, J. H. G. (2008). Comparative LC-MS: A landscape of peaks and valleys. *Proteomics*, **8**(4), 731–749.

Bellew, M., Coram, M., Fitzgibbon, M., Igra, M., Randolph, T., Wang, P., May, D., Eng, J., Fang, R., Lin, C., Chen, J., Goodlett, D., Whiteaker, J., Paulovich, A., and McIntosh, M. (2006). A suite of algorithms for the comprehensive analysis of complex protein mixtures using high-resolution LC-MS. *Bioinformatics*, **22**(15), 1902–1909.

Brockmann, M., Gasser, T., and Herrmann, E. (1993). Locally adaptive bandwidth choice for kernel regression estimators. *J Am Stat Assoc*, **88**(424), 1302.

Clifford, D., Stone, G., Montoliu, I., Rezzi, S., Martin, F.-P., Guy, P., Bruce, S., and Kochhar, S. (2009). Alignment using variable penalty dynamic time warping. *Anal Chem*, **81**(3), 1000–1007.

Cox, J. and Mann, M. (2008). MaxQuant enables high peptide identification rates, individualized p.p.b.-range mass accuracies and proteome-wide protein quantification. *Nat Biotechnol*, **26**(12), 1367–1372.

Dettmer, K., Aronov, P. A., and Hammock, B. D. (2007). Mass spectrometry-based metabolomics. *Mass Spectrom Rev*, **26**(1), 51–78.

Gale, D. and Shapley, L. (1962). College admissions and the stability of marriage. *Am Math Mon*, **69**(1), 15, 9.

Gay, S., Binz, P.-A., Hochstrasser, D. F., and Appel, R. D. (2002). Peptide mass fingerprinting peak intensity prediction: Extracting knowledge from spectra. *Proteomics*, **2**(10), 1374–1391.

Katajamaa, M. and Oresic, M. (2005). Processing methods for differential analysis of LC/MS profile data. *BMC Bioinformatics*, **6**, 179.

Katajamaa, M., Miettinen, J., and Oresic, M. (2006). MZmine: Toolbox for processing and visualization of mass spectrometry based molecular profile data. *Bioinformatics*, **22**(5), 634–6.

Khan, Z., Bloom, J. S., Garcia, B. A., Singh, M., and Kruglyak, L. (2009). Protein quantification across hundreds of experimental conditions. *PNAS*, **106**(37), 15544–15548.

Kirchner, M., Saussen, B., Steen, H., Steen, J. A. J., and Hamprecht, F. A. (2007). amsrpm: Robust point matching for retention time aligment of LC/MS data with R.

Kohlbacher, O., Reinert, K., Gropl, C., Lange, E., Pfeifer, N., Schulz-Trieglaff, O., and Sturm, M. (2007). TOPP - the OpenMS proteomics pipeline. *Bioinformatics*, **23**(2), e191–e197.

Lange, E., Gropl, C., Schulz-Trieglaff, O., Leinenbach, A., Huber, C., and Reinert, K. (2007). A geometric approach for the alignment of liquid chromatography mass spectrometry data. *Bioinformatics*, **23**(13), 273–281.

Lange, E., Tautenhahn, R., Neumann, S., and Gropl, C. (2008). Critical assessment of alignment procedures for LC-MS proteomics and metabolomics measurements. *BMC Bioinformatics*, **9**(1), 375.

Li, X., Yi, E. C., Kemp, C. J., Zhang, H., and Aebersold, R. (2005). 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), 1328–40.

May, D., Fitzgibbon, M., Liu, Y., Holzman, T., Eng, J., Kemp, C. J., Whiteaker, J., Paulovich, A., and McIntosh, M. (2007). A platform for accurate mass and time analyses of mass spectrometry data. *J Prot Res*, **6**(7), 2685–94.

Mueller, L. N., Rinner, O., Schmidt, A., Letarte, S., Bodenmiller, B., Brusniak, M.-Y., Vitek, O., Aebersold, R., and Müller, M. (2007). SuperHirn - a novel tool for high resolution LC-MS-based peptide/protein profiling. *Proteomics*, **7**(19), 3470–3480.

Pluskal, T., Castillo, S., Villar-Briones, A., and Oresic, M. (2010). MZmine 2: Modular framework for processing, visualizing, and analyzing mass spectrometry-based molecular profile data. *BMC Bioinformatics*, **11**, 395.

Podwojski, K., Fritsch, A., Chamrad, D. C., Paul, W., Sitek, B., Stuhler, K., Mutzel, P., Stephan, C., Meyer, H. E., Urfer, W., Ickstadt, K., and Rahnenfuhrer, J. (2009). Retention time alignment algorithms for LC/MS data must consider non-linear shifts. *Bioinformatics*, **25**(6), 758–764.

Powell, M. J. D. (1976). Some global convergence properties of a variable metric algorithm for minimization without exact line searches. *Nonlinear Programming, SIAMS-AMS Proc.*, **9**.

Prakash, A., Mallick, P., Whiteaker, J., Zhang, H., Paulovich, A., Flory, M., Lee, H., Aebersold, R., and Schwikowski, B. (2006). Signal maps for mass spectrometry-based comparative proteomics. *Mol Cell Proteomics*, **5**(3), 423–432.

Prince, J. T., Carlson, M. W., Wang, R., Lu, P., and Marcotte, E. M. (2004). The need for a public proteomics repository. *Nat Biotechnol*, **22**(4), 471–472.

Shevchenko, A. and Simons, K. (2010). Lipidomics: Coming to grips with lipid diversity. *Nat Rev Mol Cell Biol*, **11**(8), 593–598.

Smith, C. A., Want, E. J., O'Maille, G., Abagyan, R., and Siuzdak, G. (2006). XCMS: Processing mass spectrometry data for metabolite profiling using nonlinear peak alignment, matching, and identification. *Anal Chem*, **78**(3), 779–787.

Sturm, M., Bertsch, A., Groepl, C., Hildebrandt, A., Hussong, R., Lange, E., Pfeifer, N., Trieglaff, O., Zerck, A., Reinert, K., and Kohlbacher, O. (2008). OpenMS - an open-source software framework for mass spectrometry. *BMC Bioinformatics*, **9**(1).

Vandenbogaert, M., Li-Thiao-Té, S., Kaltenbach, H.-M., Zhang, R., Aittokallio, T., and Schwikowski, B. (2008). Alignment of LC-MS images, with applications to biomarker discovery and protein identification. *Proteomics*, **8**(4), 650–72.

Zaia, J. (2010). Mass spectrometry and glycomics. *OMICS*, **14**(4), 401–18.

Zhang, X., Asara, J. M., Adamec, J., Ouzzani, M., and Elmagarmid, A. K. (2005). Data pre-processing in liquid chromatography-mass spectrometry-based proteomics. *Bioinformatics*, **21**(21), 4054–9.