Linear MALDI-ToF simultaneous spectrum deconvolution and baseline removal

Vincent Picaud[1*] , Jean-Francois Giovannelli[2], Caroline Truntzer[3], Jean-Philippe Charrier[4], Audrey Giremus[2], Pierre Grangeat[5,6] and Catherine Mercier[7,8,9]

Abstract

Background: Thanks to a reasonable cost and simple sample preparation procedure, linear MALDI-ToF spectrometry is a growing technology for clinical microbiology. With appropriate spectrum databases, this technology can be used for early identification of pathogens in body fluids. However, due to the low resolution of linear MALDI-ToF instruments, robust and accurate peak picking remains a challenging task. In this context we propose a new peak extraction algorithm from raw spectrum. With this method the spectrum baseline and spectrum peaks are processed jointly. The approach relies on an additive model constituted by a smooth baseline part plus a sparse peak list convolved with a known peak shape. The model is then fitted under a Gaussian noise model. The proposed method is well suited to process low resolution spectra with important baseline and unresolved peaks.

Results: We developed a new peak deconvolution procedure. The paper describes the method derivation and discusses some of its interpretations. The algorithm is then described in a pseudo-code form where the required optimization procedure is detailed. For synthetic data the method is compared to a more conventional approach. The new method reduces artifacts caused by the usual two-steps procedure, baseline removal then peak extraction. Finally some results on real linear MALDI-ToF spectra are provided.

Conclusions: We introduced a new method for peak picking, where peak deconvolution and baseline computation are performed jointly. On simulated data we showed that this global approach performs better than a classical one where baseline and peaks are processed sequentially. A dedicated experiment has been conducted on real spectra. In this study a collection of spectra of spiked proteins were acquired and then analyzed. Better performances of the proposed method, in term of accuracy and reproductibility, have been observed and validated by an extended statistical analysis.

Keywords: Mass spectrometry, Peak picking, Deconvolution, Baseline

Background

Linear matrix-assisted laser desorption/ionization time-of-flight mass spectrometry (MALDI-ToF MS) has now revolutionized identification of bacteria, yeasts and molds in clinical microbiology [1]. The technology is simple, accurate, fast, and for large laboratories less expensive than conventional methods. Despite a lower resolution than other analyzers used in modern proteomics, linear ToF are preferred in microbiology because of a better sensitivity in the 2-20 kDa mass range, where proteins contain

phylogenic information. Moreover, the lower cost of linear instruments favored a wider adoption by health institutions. In essence, identifications are performed in minutes by simply acquiring an experimental spectrum of the whole microorganism cells and comparing the resulting peak list with a database [2, 3]. In this context we propose a new method for peak extraction especially adapted to linear MALDI-ToF spectra. The usual approach for MALDI mass spectra processing generally consists of chaining several procedures. Most of the times we have a smoothing step, a baseline correction step and only then the final the peak extraction [4]. The main idea of our new method is to jointly perform these steps with the aim of reducing the potential unrecoverable artifacts introduced by

*Correspondence: vincent.picaud@cea.fr
[1]CEA, LIST, Laboratoire d'Intégration de Systèmes et des Technologies, DIGITEO Bât 565 - Point Courrier 192, 91191 Gif-sur-Yvette Cedex, France
Full list of author information is available at the end of the article

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 2 of 20

a sequential processing. In the next sections we briefly present the three main steps of the usual approaches. We then describe our method and its detailed derivation.

Smoothing

A popular [5] smoothing technique in the spectrometry community is the use of Savitzky-Golay linear filters [6, 7]. These moving average filters perform a least squares fit of a small set of consecutive data points to a polynomial. The value of this fitted polynomial at the window central point is the filter output. One can also compute a smoothed derivative by using the derivative of the fitted polynomial to compute the central point value. This smoothed derivative can also be used by peak picking algorithms [4]. This method is versatile and efficient. However, its main drawbacks are that we have to manually choose the polynomial degree and the window length. Some studies to automatically choose the former [8] or later [9] exist but to the best of our knowledge they are not used as often as the original approach.

A more recent approach to smooth spectra is the use the wavelet transform. The Undecimated Wavelet Transform (UDWT) [10, 11] is generally preferred to the Discrete Wavelet Transform (DWT) as it produces less artifacts after coefficient thresholding. The UDWT is equivalent to an averaged DWT computed for all integer shifts of the signal and is thus a redundant and shift invariant transform. In applications it has been reported to yield better qualitative denoising [12].

Baseline correction

Baseline correction is a difficult problem that potentially also introduces artifacts [13]. There are at least two kinds of approaches for baseline correction. One category of methods is close to mathematical morphology. In these methods a lower envelope of the spectrum [14, 15] is computed. Methods of this category generally need a smoothed signal (see "Smoothing" section). The other category contains methods using an asymmetric loss function to fit spectrum baseline without being biased by peaks [16, 17]. Finally some other methods mix the two previous approaches [18].

Peak picking

After baseline removal the next step is generally a peak picking procedure. Several approaches are possible. Perhaps the most intuitive approach is to compute a regularized second order derivative (using Savitzky-Golay for instance) of the spectrum and to extract local minima [4]. The use of second order derivative minima instead of the zero-crossing of the first order derivative allows, to some extend, to detect overlapping peaks [19].

A second kind of approach, especially useful in case of overlapping peaks, is peak deconvolution. Overlapping of complex peak patterns can be deconvolved if one uses specially tuned point spread function and judicious regularizations (positivity constraint and sparsity-inducing norms, like the l_1 norm) [20–23]. Further generalizations can be obtained in case of blind-deconvolution [24]. However these kinds of approaches are much more computationally intensive and are not widely used in mass spectrometry.

Finally we can mention the Continuous Wavelet Transform (CWT) [25, 26] which can be efficiently computed using the Fast Fourier Transform. The idea is to follow wavelet modulus maximum. Theses ridges characterize the regularity of the signal [27] and can be used to detect peaks.

Contributions

Computing the baseline correction and finding peaks are two strongly linked problems, it is thus natural to perform these two operations jointly. In this work we propose such an approach.

In the first part of the paper we describe a direct model with an additive noise where the spectrum is modelized by a smooth baseline plus a sparse peak list convolved by a given peak shape function. We describe how we chose our priors to enforce baseline smoothness and sparsity of the peak list. We then assume a Gaussian distribution for the noise. This allows us to use Euclidean distance to quantify the error between our model and the measured spectrum.

Next we show how the unknown baseline can be eliminated from the model. This manipulation leads to a modified problem very close to the classical deconvolution one. We underline this similarity and rigorously describe the two limiting cases, zero or infinite penalization for the baseline smoothness. As a by product we can interpret that our new deconvolution method is equivalent in some way to deconvolve a regularized second order derivative of the initial spectrum.

We then carefully examine the behavior of the computed baseline at the spectrum boundaries. We observed that when the smoothness penalty is too strong, the computed baseline can become overly flat. To avoid this effect a correction allowing to define baseline values at boundaries is proposed. With this modification the behavior of the baseline at the boundaries is no more affected by strong baseline smoothness penalty.

An effective optimization method to compute the solution of the deconvolution problem is exposed. This optimization algorithm is used twice in our two-passes deconvolution procedure. In the first pass a sparsity prior is used and a first optimization problem is solved to find peak centers. In the second pass the sparsity prior is replaced by the previously found peak positions. This second optimization problem is solved to compute peak height values.

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 3 of 20

Finally the new method is compared to one instance of the smoothing/baseline correction/deconvolution classical approach. The advantage of the joint baseline computation and peak deconvolution is demonstrated on synthetic data. An example on "real" data is shown and a reference to a more detailed comparison between our method and classical ones is given.

Method

Problem definition

The proposed developments are founded on a natural model for the observed spectrum $\mathbf{y}$ as a spiky signal $\mathbf{x}_p$ convolved with a peak shape p superimposed onto a smooth baseline $\mathbf{x}_b$

$$\mathbf{y} = \mathbf{x}_b + \mathbf{x}_p * p + \mathbf{e} \tag{1}$$

where $\mathbf{e}$ includes measurement and model errors. It is a common linear model with additive uncertainties. These quantities are represented by vectors of size n, where n is the number of m/z channels of the original spectrum $\mathbf{y}$.

The problem at stake is to recover both the signal of interest $\mathbf{x}_p$ and the baseline $\mathbf{x}_b$. It is a difficult task at least for three reasons.

- It is underdetermined since the number of unknowns is twice the number of data.
- The convolution reduces the resolution due to peak enlargement and possible overlap.
- Measurement noise and possible model inadequacy induce additional uncertainties.

As a consequence, information must be accounted for regarding the expected signals $\mathbf{x}_p$ and $\mathbf{x}_b$. In the following developments, $\mathbf{x}_b$ is expected to be smooth while $\mathbf{x}_p$ is expected to be spiky and positive. This knowledge will be included in the next sections.

$\mathbf{x}_b$ smoothness

A simple way to account for smoothness of $\mathbf{x}_b$ is to penalize its fluctuations through

$$\mathcal{P}_b(\mathbf{x}_b) = \frac{\mu}{2} \sum_{k=1}^{n-1} (\mathbf{x}_b[k+1] - \mathbf{x}_b[k])^2 = \frac{\mu}{2} \|\mathbf{D}\mathbf{x}_b\|_2^2 \tag{2}$$

where $\mathbf{D}$ is a finite differences matrix of size $(n-1) \times n$ (given in Appendix "Smoothness and convolution matrix") and $\mu > 0$.

$\mathbf{x}_p$ sparsity and positivity

In order to favor sparsity for $\mathbf{x}_p$ (spiky property) an elastic-net penalty is introduced

$$\mathcal{P}_p(\mathbf{x}_p) = \lambda_1 \|\mathbf{x}_p\|_1 + \frac{\lambda_2}{2} \|\mathbf{x}_p\|_2^2 \tag{3}$$

The degree of sparsity is controlled by λ_1. The coefficient λ_2 is generally set to zero or to a very small value. The reason why we have introduced this extra regularization is that a small positive value can sometimes improve convergence speed of the algorithm. In practice this only happens for spectra of several thousand of m/z channels and always has a limited impact on the obtained solution.

In addition, the solution also requires positivity of $\mathbf{x}_p$, i.e. positivity for each component $\mathbf{x}_p[k]$. The l_1 norm then simplifies

$$\|\mathbf{x}_p\|_1 = \sum_{k=1}^{n} \mathbf{x}_p[k] = \mathbb{1}_n^t \mathbf{x}_p$$

where $\mathbb{1}_n$ is the n dimensional column vector with each entry to 1. This substitution turns the convex, but nonsmooth, penalty $\mathcal{P}_p(\mathbf{x}_p)$ into a smooth quadratic one. This idea has already been used in [23, 28].

Data-fidelity term

A common and natural way to quantify data - model discrepancy founded on Eq. 1 relies on a squared Euclidean norm:

$$\mathcal{J}_{\text{obs}}(\mathbf{x}_b, \mathbf{x}_p) = \frac{1}{2} \|\mathbf{y} - (\mathbf{x}_b + \mathbf{L}\mathbf{x}_p)\|_2^2 \tag{4}$$

where the $n \times n$ band matrix $\mathbf{L}$ represents the convolution with the peak shape p.

Complete objective

Using Eqs. 2, 3 and 4, we get a first expression of the complete objective:

$$\mathcal{J}(\mathbf{x}_b, \mathbf{x}_p) = \mathcal{J}_{\text{obs}}(\mathbf{x}_b, \mathbf{x}_p) + \mathcal{P}_b(\mathbf{x}_b) + \mathcal{P}_p(\mathbf{x}_p) \tag{5}$$

that is to say:

$$\mathcal{J}(\mathbf{x}_b, \mathbf{x}_p) = \frac{1}{2} \|\mathbf{y} - (\mathbf{x}_b + \mathbf{L}\mathbf{x}_p)\|_2^2 \tag{6}$$
$$+ \frac{\mu}{2} \|\mathbf{D}\mathbf{x}_b\|_2^2 + \lambda_1 \|\mathbf{x}_p\|_1 + \frac{\lambda_2}{2} \|\mathbf{x}_p\|_2^2$$

The minimization of this objective function

$$(\widehat{\mathbf{x}}_p, \widehat{\mathbf{x}}_b) = \underset{\mathbf{x}_p \geq 0, \, \mathbf{x}_b}{\arg\min} \mathcal{J}(\mathbf{x}_b, \mathbf{x}_p) \tag{7}$$

gives the desired solution $(\widehat{\mathbf{x}}_p, \widehat{\mathbf{x}}_b)$.

Elimination of $\mathbf{x}_b$

To find the solution of problem Eq. 7 we begin by solving it for the $\mathbf{x}_b$ vector. This is an unconstrained quadratic problem and an analytical solution can be found. We shall first write down the gradient of $\mathcal{J}$ with respect to $\mathbf{x}_b$:

$$\nabla_{\mathbf{x}_b} \mathcal{J} = (\mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D}) \mathbf{x}_b - (\mathbf{y} - \mathbf{L}\mathbf{x}_p) \tag{8}$$

Solving $\nabla_{\mathbf{x}_b} \mathcal{J} = 0$ yields the minimizer:

$$\widehat{\mathbf{x}}_b = (\mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D})^{-1} (\mathbf{y} - \mathbf{L}\mathbf{x}_p)$$
$$= \mathbf{B}_\mu^{-1} (\mathbf{y} - \mathbf{L}\mathbf{x}_p) \tag{9}$$

where $\mathbf{B}_\mu = \mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D}$. This operation is always possible since $\mathbf{B}_\mu$ is invertible (sum of the identity matrix and a semi-positive matrix).

It is then possible to substitute $\mathbf{x}_b$ for $\widehat{\mathbf{x}}_b$ in objective Eq. 5 to obtain a reduced objective. After some algebra (proof is given in Appendix "Derivation of reduced objective"), we have

$$\widetilde{\mathcal{J}}(\mathbf{x}_p) = \mathcal{J}\left(\widehat{\mathbf{x}}_b, \mathbf{x}_p\right) + \mathbf{C}$$

where $\mathbf{C}$ is a constant, and $\widetilde{\mathcal{J}}(\mathbf{x}_p)$ is defined by:

$$\widetilde{\mathcal{J}}(\mathbf{x}_p) = \frac{1}{2}\mathbf{x}_p^t \left(\mathbf{L}^t \mathbf{A}_\mu \mathbf{L} + \lambda_2 \mathbf{I}_b\right) \mathbf{x}_p \\ - \mathbf{x}_p^t \left(\mathbf{L}^t \mathbf{A}_\mu \mathbf{y} - \lambda_1 \mathbb{1}_n\right) \quad (10)$$

The matrix $\mathbf{A}_\mu$ is defined by:

$$\mathbf{A}_\mu = \mathbf{I}_b - \mathbf{B}_\mu^{-1} = \mathbf{I}_b - \left(\mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D}\right)^{-1} \quad (11)$$

and its interpretation is discussed in detail in "Analysis of the $\mathbf{A}_\mu$ matrix" section).

This quadratic form $\widetilde{\mathcal{J}}(\mathbf{x}_p)$ is our objective function and the solution $\widehat{\mathbf{x}}_p$ is its minimizer subject to positivity:

$$\widehat{\mathbf{x}}_p = \arg\min_{\mathbf{x}_p \geq 0} \widetilde{\mathcal{J}}(\mathbf{x}_p) \quad (12)$$

Once this constrained quadratic problem is solved, we can retrieve $\widehat{\mathbf{x}}_b$ using Eq. 9:

$$\widehat{\mathbf{x}}_b = \mathbf{B}_\mu^{-1} \left(\mathbf{y} - \mathbf{L}\widehat{\mathbf{x}}_p\right)$$

then the initial problem Eq. 7 is solved.

The gradient of $\widetilde{\mathcal{J}}(\mathbf{x}_p)$ is easily deduced form Eq. 10 and reads:

$$\nabla_{\mathbf{x}_p}\widetilde{\mathcal{J}} = \left(\mathbf{L}^t \mathbf{A}_\mu \mathbf{L} + \lambda_2 \mathbf{I}_p\right) \mathbf{x}_p - \left(\mathbf{L}^t \mathbf{A}_\mu \mathbf{y} - \lambda_1 \mathbb{1}_p\right) \quad (13)$$

Interpretation of the reduced criterion

We can make a pause here to interpret Eq. 12. If there was no baseline a natural way to deconvolve the spectrum would be to solve:

$$\arg\min_{\mathbf{x}_p \geq 0} \frac{1}{2}\|\mathbf{y} - \mathbf{L}\mathbf{x}_p\|_2^2 + \lambda \|\mathbf{x}_p\|_1$$

If we use the positivity constraint and expand this objective function we get the following equivalent problem:

$$\arg\min_{\mathbf{x}_p \geq 0} \frac{1}{2}\mathbf{x}_p^t \mathbf{L}^t \mathbf{L}\mathbf{x}_p + \mathbf{x}_p^t \left(\lambda_1 \mathbb{1}_p - \mathbf{L}^t \mathbf{y}\right) \quad (14)$$

Now if we look Eq. 10 and set $\lambda_2 = 0$ we get:

$$\arg\min_{\mathbf{x}_p \geq 0} \frac{1}{2}\mathbf{x}_p^t \mathbf{L}^t \mathbf{A}_\mu \mathbf{L}\mathbf{x}_p + \mathbf{x}_p^t (\lambda_1 \mathbb{1}_p - \mathbf{L}^t \mathbf{A}_\mu \mathbf{y}) \quad (15)$$

The two equations are very similar apart from the fact that the operator $\mathbf{A}_\mu$ is applied to the reconstructed peaks $\mathbf{L}\mathbf{x}_p$ and to the raw spectrum $\mathbf{y}$. It could be interpreted as the precision (inverse of the covariance) matrix in a correlated noise framework. Instead of this classical approach and to better understand its role in our deconvolution context we study the evolution of $\mathbf{A}_\mu$ in the two limiting cases $\mu \to 0$ and $\mu \to \infty$.

Analysis of the $\mathbf{A}_\mu$ matrix

Figure 1 shows the column $j = \lfloor \frac{n}{2} \rfloor$ of $\mathbf{A}_\mu$ for two different values of μ.

Fig. 1 $\mathbf{A}_\mu$ operator. $\mathbf{A}_\mu\left(:, \lfloor \frac{n}{2} \rfloor\right)$ column plot for $\mu = 1$ and $\mu = 100$. $\mathbf{A}_\mu$ operator acts like a regularized second order derivation. The regularization strength increases as μ increases (μ is the background smoothness parameter)
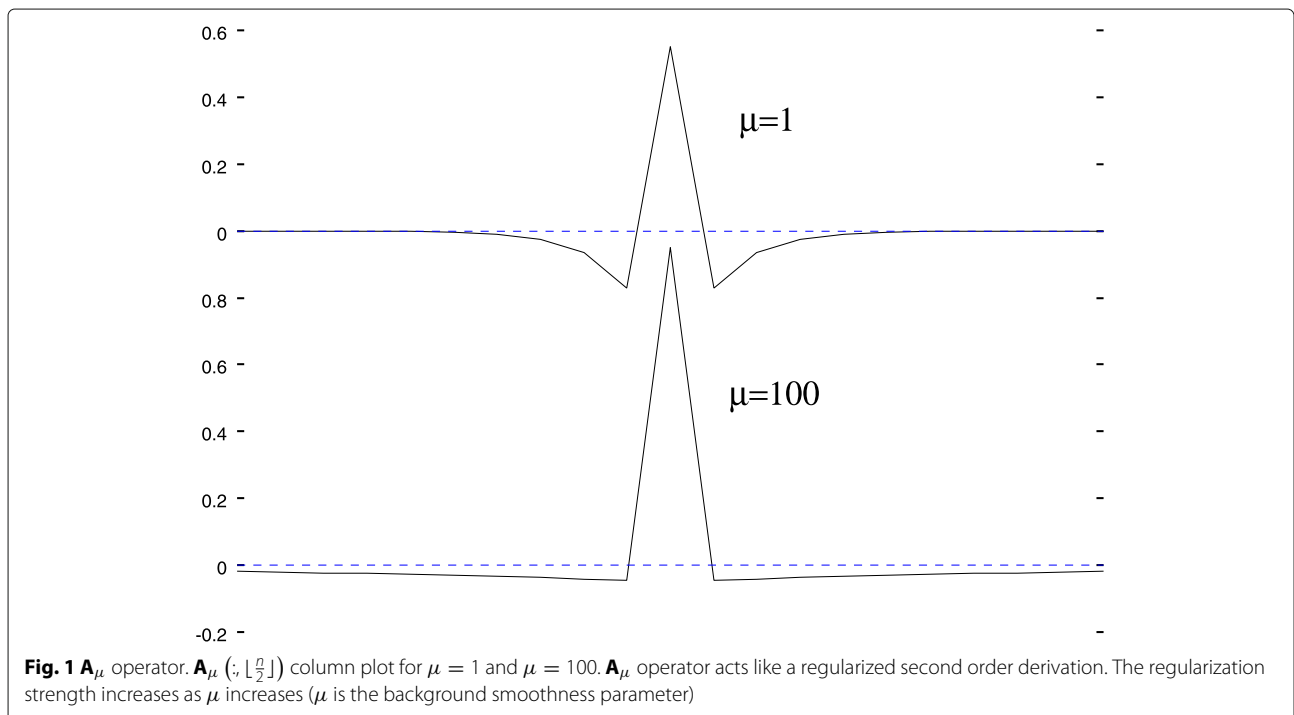
For small μ, the $\mathbf{A}_\mu$ operator acts like a regularized second order derivation. For a sufficiently small μ (such that the spectral radius satisfies $\rho(\mu\mathbf{D}^t\mathbf{D}) < 1$) a Taylor expansion gives:

$$\begin{aligned}
\mathbf{A}_\mu &= \mathbf{I}_b - \left(\mathbf{I}_b + \mu\mathbf{D}^t\mathbf{D}\right)^{-1} \\
&= \mathbf{I}_b - \left(\mathbf{I}_b - \mu\mathbf{D}^t\mathbf{D} + o(\mu)\right) \\
&= \mu\mathbf{D}^t\mathbf{D} + o(\mu)
\end{aligned}$$

Hence compared to Eq. 14, one can interpret Eq. 15 as an usual deconvolution applied on the second order derivative $\mathbf{A}_\mu\mathbf{y}$ of the initial spectrum $\mathbf{y}$. The used point spread function is also the second order derivative $\mathbf{A}_\mu\mathbf{L}$ of the initial peak shape p introduced in Eq. 1. The overall effect of the $\mathbf{A}_\mu$ operator is to cancel the slow varying component of the signal. In another terms, the baseline is removed thanks to a derivation.

The regularization strength increases as the baseline parameter μ increases. At the limit $\mu \to \infty$ (proof given in Appendix "Expression of $\lim_{\mu\to\infty} \mathbf{A}_\mu$") we get:

$$\lim_{\mu\to\infty} \left[\mathbf{A}_\mu\right]_{i,j} = \delta_{i,j} - \frac{1}{n}$$

When applied to any vector $\mathbf{y}$ we get

$$\lim_{\mu\to\infty} \mathbf{A}_\mu\mathbf{y} = \mathbf{y} - \bar{\mathbf{y}}\mathbb{1}_n$$

Thus the action of $\mathbf{A}_\mu$ centers the signal $\mathbf{y}$ by subtracting the constant $\bar{\mathbf{y}}\mathbb{1}_n$ vector, where the scalar $\bar{\mathbf{y}}$ is the mean value of the $\mathbf{y}$ vector's components. The same centering holds for the peak shape function p. It follows that in this limiting case the solved problem is the usual deconvolution procedure applied on the centered spectrum. The computed baseline is simply a constant equal to the spectrum mean.

Debiasing
The l_1 penalty acts as a soft threshold to select peaks ([29], Section 10). This leads to a bias in peak intensity estimation. These intensities are artificially reduced when the l_1 penalty increases. We use the ideas introduced in [28] to get corrected peak intensities. This yields a resolution procedure involving two stages. The first stage selects the peaks, the second one corrects their intensities.

First stage: peak support selection
Given μ, λ_1 and λ_2, we solve the minimization problem Eq. 12. The obtained solution $\widehat{\mathbf{x}}_p$ is a sparse vector containing peak intensities. The intensities are biased if the λ hyper-parameters are not null. However we can use $\widehat{\mathbf{x}}_p$ to

define the peak support Ω. In the present work the peak support is simply defined by keeping the local maxima of $\widehat{\mathbf{x}}_p$.

$$\begin{aligned}
\Omega = \big\{ i, &\left((\widehat{\mathbf{x}}_p[i] > \widehat{\mathbf{x}}_p[i-1]) \wedge (\widehat{\mathbf{x}}_p[i] \geq \widehat{\mathbf{x}}_p[i+1])\right) \vee \\
&\left((\widehat{\mathbf{x}}_p[i] \geq \widehat{\mathbf{x}}_p[i-1]) \wedge (\widehat{\mathbf{x}}_p[i] > \widehat{\mathbf{x}}_p[i+1])\right) \big\} \quad (16)
\end{aligned}$$

The peak intensities are going to be corrected in the second stage, but their final positions are defined by the condition Eq. 16. More complex procedures can be used to find the peak support. One such example is the procedure presented in [22], *Post-processing and thresholding*. These more refined methods can be introduced in a straightforward way in our approach by using them instead of the basic condition Eq. 16.

Second stage: peak intensity correction
In the second step, as we have peak support Ω, we do not need the l_1 regularization anymore. Hence we simply ignore it and solve the simplified optimization problem:

$$\begin{aligned}
\mathbf{x}_p &= \arg\min \frac{1}{2}\mathbf{x}_p^t\mathbf{L}^t\mathbf{A}_\mu\mathbf{L}\mathbf{x}_p - \mathbf{x}_p^t\mathbf{L}^t\mathbf{A}_\mu\mathbf{y} \\
&\text{s.t.} \quad \begin{cases} \mathbf{x}_p[k] = 0 \text{ if } k \notin \Omega \\ \mathbf{x}_p[k] \geq 0 \text{ if } k \in \Omega \end{cases}
\end{aligned} \quad (17)$$

Despite its more complex appearance, this problem is no more complicated than Eq. 12. Solving this problem will correct peak intensities by removing the bias induced by the previously used l_1 penalty.

Boundary conditions
There is a possible improvement of the method concerning boundary conditions. As Eq. 2 suggests, a strong μ penalty forces the baseline to be constant. In Fig. 2 we see that this phenomenon is especially present at the domain boundaries.

We solved this problem by imposing baseline values at boundaries. This corrected solution is also shown in Fig. 2 and we can see that the corrected solution does not suffer from boundary effect anymore. Appendix "Boundary correction", page 11, provides all the details on how to modify Eqs. 9 and 13 to introduce some constraints on the baseline values $\mathbf{x}_b$. These modified equations will constitute our final model formulation.

Final model formulation
After boundary correction, the modified Eq. 9 is

$$\mathbf{x}_b = \widetilde{\mathbf{B}}_\mu^{-1}\left(\widetilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_p\right) \quad (18)$$

and the modified Eq. 10 is

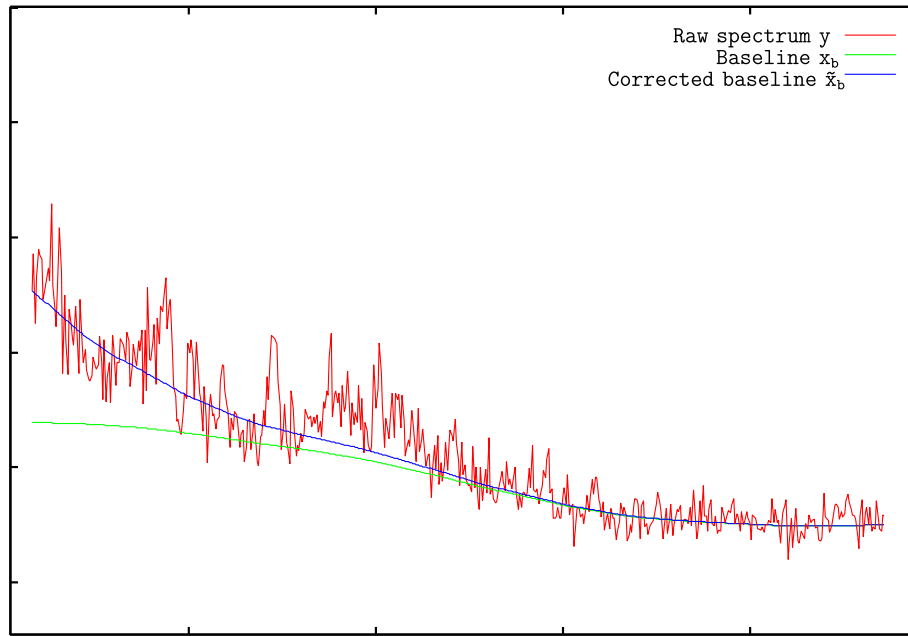Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 6 of 20

Fig. 2 The boundary problem and its correction The green curve is the baseline $\mathbf{x}_b$ as computed by Eq. 2, the blue curve is the baseline $\tilde{x}_b$ after boundary condition correction

$$\widetilde{\mathcal{J}}_2(\mathbf{x}_p) = \frac{1}{2}\mathbf{x}_p^t \left(\lambda_2 \mathbf{I}_p + \mathbf{L}^t \widetilde{\mathbf{A}}_\mu \mathbf{L}\right) \mathbf{x}_p + \\ \mathbf{x}_p^t \left(\lambda_1 \mathbb{1}_p - \mathbf{L}^t \widetilde{\mathbf{A}}_\mu \widetilde{\mathbf{y}} - \mathbf{L}^t(\mathbf{y} - \widetilde{\mathbf{y}})\right) \quad (19)$$

The solution $\widehat{\mathbf{x}}_p$ is the unique minimizer of $\widetilde{\mathcal{J}}_2$ subject to positivity constraint:

$$\widehat{\mathbf{x}}_p = \underset{\mathbf{x}_p \geq 0}{\arg\min}\widetilde{\mathcal{J}}_2(\mathbf{x}_p) \quad (20)$$

The gradient is obtained by a straightforward computation:

$$\nabla_{\mathbf{x}_p}\widetilde{\mathcal{J}}_2 = \left(\lambda_2 \mathbf{I}_p + \mathbf{L}^t \widetilde{\mathbf{A}}_\mu \mathbf{L}\right)\mathbf{x}_p + \lambda_1 \mathbb{1}_p - \mathbf{L}^t \widetilde{\mathbf{A}}_\mu \widetilde{\mathbf{y}} - \mathbf{L}^t(\mathbf{y} - \widetilde{\mathbf{y}}) \quad (21)$$

As before, $\widehat{\mathbf{x}}_b$ is computed from $\widehat{\mathbf{x}}_p$ using Eq. 18. The explicit forms of $\widetilde{\mathbf{B}}_\mu$, $\widetilde{\mathbf{y}}$ and $\widetilde{\mathbf{A}}_\mu$ are given in Appendix "Boundary correction", respectively by Eqs. 31, 32 and 33.

Algorithm summary

For ease of reading Algorithm 1 recapitulates the main steps of the proposed method in its final formulation. The optimization procedure used to solve efficiently the two minimization problems will be described in detail in the next section.

Algorithm 1 Global view of the proposed method

Require:
∗ the $\mathbf{y}$ spectrum
∗ its baseline value at boundaries $\bar{\mathbf{y}}[1]$ and $\bar{\mathbf{y}}[n]$
∗ the penalty parameters λ_1, λ_2 and μ

▷ **Boundary correction**
Compute $\widetilde{\mathbf{A}}_\mu$, $\widetilde{\mathbf{B}}_\mu$ and $\widetilde{\mathbf{y}}$, defined by Eqs. 31, 32 and 33.

▷ **Stage 1: deconvolution with l_1 penalty**
Minimize (with Algorithm 2)

$$\widehat{\mathbf{x}}_p = \underset{\mathbf{x}_p \geq 0}{\arg\min}\widetilde{\mathcal{J}}_2(\mathbf{x}_p)$$

($\widetilde{\mathcal{J}}_2$ and $\nabla_{\mathbf{x}_p}\widetilde{\mathcal{J}}_2$ are defined by Eqs. 19, 21)

▷ **Stage 2: deconvolution with peak support**
Find peak support Ω (see Eq. 16)
Set $\lambda_1 \leftarrow 0$ and $\lambda_2 \leftarrow 0$ (this modifies $\widetilde{\mathcal{J}}_2$)
Minimize (with Algorithm 2)

$$\widehat{\mathbf{x}}_p = \underset{\mathbf{x}_p}{\arg\min}\widetilde{\mathcal{J}}_2(\mathbf{x}_p)$$

$$\text{s.t.} \begin{cases} \mathbf{x}_p[k] = 0 \ \text{if} \ k \notin \Omega \\ \mathbf{x}_p[k] \geq 0 \ \text{if} \ k \in \Omega \end{cases}$$

▷ **Post-processing**
Compute:
∗ baseline $\widehat{\mathbf{x}}_b = \widetilde{\mathbf{B}}_\mu^{-1}(\widetilde{\mathbf{y}} - \mathbf{L}\widehat{\mathbf{x}}_p)$ (Eq. 18)
∗ reconstructed peaks $\mathbf{L}\widehat{\mathbf{x}}_p$
∗ reconstructed spectrum $\widehat{\mathbf{y}} = \widehat{\mathbf{x}}_b + \mathbf{L}\widehat{\mathbf{x}}_p$

Picaud *et al. BMC Bioinformatics*   (2018) 19:123

Page 7 of 20

Effective minimization

Quadratic programming with bound constraints

To solve the optimization problems Eq. 12 or Eq. 20 and their associated debiasing step Eq. 17 we use the projected Barzilai-Borwein method described in [30]. The Barzilai-Borwein method [31] dramatically improves the classical steepest descent with Cauchy step.

For a convex quadratic form

$$\mathbf{x}_p = \arg\min_{\mathbf{x}} \mathcal{J}(\mathbf{x})$$
$$= \arg\min_{\mathbf{x}} \frac{1}{2}\mathbf{x}^t\mathbf{Q}\mathbf{x} + \mathbf{q}^t\mathbf{x}$$

the steepest descent direction is defined by

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} - \alpha \nabla_{\mathbf{x}^{(k)}}\mathcal{J}$$

with

$$\nabla_{\mathbf{x}^{(k)}}\mathcal{J} = \mathbf{Q}\mathbf{x}^{(k)} + \mathbf{q}$$

The associated Cauchy step is defined by

$$\alpha_{\text{Cauchy}} = \arg\min_{\alpha} \mathcal{J}\left(\mathbf{x}^{(k)} - \alpha \nabla_{\mathbf{x}^{(k)}}\mathcal{J}\right)$$

A straightforward calculation leads to:

$$\alpha_{\text{Cauchy}} = \frac{\|\nabla_{\mathbf{x}^{(k)}}\mathcal{J}\|_2^2}{\nabla_{\mathbf{x}^{(k)}}\mathcal{J}^t \, \mathbf{Q} \, \nabla_{\mathbf{x}^{(k)}}\mathcal{J}}$$

The Barzilai-Borwein method uses the same steepest descent direction but replaces the Cauchy step by one (or an alternating sequence) of the two so-called Barzilai-Borwein steps. These step lengths are defined by [30]:

$$\alpha_{BB1} = \frac{\left\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right\|_2^2}{\left(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right)^t \left(\nabla_{\mathbf{x}^{(k)}}\mathcal{J} - \nabla_{\mathbf{x}^{(k-1)}}\mathcal{J}\right)} \quad (22)$$

$$\alpha_{BB2} = \frac{\left(\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\right)^t \left(\nabla_{\mathbf{x}^{(k)}}\mathcal{J} - \nabla_{\mathbf{x}^{(k-1)}}\mathcal{J}\right)}{\left\|\nabla_{\mathbf{x}^{(k)}}\mathcal{J} - \nabla_{\mathbf{x}^{(k-1)}}\mathcal{J}\right\|_2^2} \quad (23)$$

Despite its simple update formula the Barzilai-Borwein method works surprisingly well [32, 33]. This method has no line search and exhibits a non-monotonic convergence. It has been shown to be globally convergent for the strictly convex quadratic case [34]. A direct extension to the constrained case is obtained by replacing the iterate $\mathbf{x}^{(k+1)}$ by its projection on the feasible domain:

$$\mathbf{x}^{(k+1)} = P\left(\mathbf{x}^{(k)} - \alpha \nabla_{\mathbf{x}^{(k)}}\mathcal{J}\right) \quad (24)$$

For bound constraints $\mathbf{x} \in [\mathbf{l}, \mathbf{u}]$ the projection operator P is simply defined (in a component wise fashion) by

$$P(\mathbf{x}) = \min\left(\mathbf{u}, \max(\mathbf{l}, \mathbf{x})\right)$$

Without any line search [30] gives a counter-example where this method is not convergent. However in practice the method is generally successful, especially for badly conditioned problems where it can outperform the constrained conjugate gradients algorithm [35].

For its simplicity and good behavior in practice we have chosen to use this method. Our implementation is given in pseudo-code in Algorithm 2.

Algorithm 2 Projected Barzilai-Borwein algorithm used to minimize $\arg\min\limits_{\mathbf{x}\in[\mathbf{l},\mathbf{u}]} \frac{1}{2}\mathbf{x}^t\mathbf{Q}\mathbf{x} + \mathbf{q}^t\mathbf{x}$

Require:
 ∗ the $\mathbf{Q}$ matrix and the $\mathbf{q}, \mathbf{l}$ and $\mathbf{u}$ vectors
 ∗ the initial $\mathbf{x}$ vector
 ∗ 3 extra storages $\mathbf{x}_m, \mathbf{g}, \mathbf{g}_m$
 for *iter* $= 0 \ldots$ *maxIter* **do**
   ▷ **Project and compute gradient Eq. 24**
   $\mathbf{x} \leftarrow P(\mathbf{x})$
   $\mathbf{g} \leftarrow \mathbf{Q}\mathbf{x} + \mathbf{q}$
   ▷ **Stopping criterion Eq. 25**
   if $Stop(\mathbf{x}, \mathbf{g}, \mathbf{l}, \mathbf{u}) < \epsilon$ **then**
     Exit on success
   end if
   ▷ **Compute α step Eqs. 22, 23**
   if *iter* $= 0$ **then**
     $\alpha \leftarrow \frac{\|\mathbf{g}\|^2}{\mathbf{g}^t\mathbf{Q}\mathbf{g}}$
   else
     if *iter* is odd **then**
       $\alpha \leftarrow \frac{\|\mathbf{x}-\mathbf{x}_m\|^2}{(\mathbf{x}-\mathbf{x}_m)^t(\mathbf{g}-\mathbf{g}_m)}$
     else
       $\alpha \leftarrow \frac{(\mathbf{x}-\mathbf{x}_m)^t(\mathbf{g}-\mathbf{g}_m)}{\|\mathbf{g}-\mathbf{g}_m\|^2}$
     end if
   end if
   ▷ **Prepare next iteration**
   $\alpha \leftarrow \max(\alpha_{min}, \min(\alpha_{max}, \alpha))$
   $\mathbf{x}_m \leftarrow \mathbf{x}$
   $\mathbf{g}_m \leftarrow \mathbf{g}$
   $\mathbf{x} \leftarrow \mathbf{x}_m - \alpha\mathbf{g}_m$
 end for
 ▷ **Did not converged**
 Error message & Exit

Variants of this method with non-monotone line search [36, 37] have been tested. Theoretically this allows to prove global convergence, but in practice we have observed a performance degradation compared to the simpler method presented in [30].

Stopping criterion

Algorithm 2 stopping criterion needs to be defined. We use a rigorous one based on Karush-Kuhn-Tucker (KKT) conditions [38]. For a smooth convex optimization problem

$$\mathbf{x}_p = \arg\min_{\mathbf{x}\in[\mathbf{l},\mathbf{u}]} \mathcal{F}(\mathbf{x})$$

necessary and sufficient conditions are:

1　Stationarity $\nabla_{\mathbf{x}_p}\mathcal{F} + \lambda^{\mathbf{u}} - \lambda^{\mathbf{l}} = \mathbf{0}$

2　Primal feasibility $\mathbf{x}_p \in [\mathbf{l}, \mathbf{u}]$

3　Dual feasibility

- Lower bounds $\lambda^{\mathbf{l}} \geq \mathbf{0}$
- Upper bounds $\lambda^{\mathbf{u}} \geq \mathbf{0}$

4　Complementary slackness

- Lower bounds $\forall i,\ \lambda^{\mathbf{l}}[i]\,(\mathbf{l}[i] - \mathbf{x}_p[i]) = 0$
- Upper bounds $\forall i,\ \lambda^{\mathbf{u}}[i]\,(\mathbf{x}_p[i] - \mathbf{u}[i]) = 0$

Our stopping criterion reflects KKT conditions violation. Enumerating all possible cases this criterion is computed as follow

$$\text{stopping criterion} = \sum_i |\mathbf{s}[i]|$$

with $\mathbf{s}[i]$ defined by:

$$\mathbf{s}[i] = \begin{cases} +\infty & \text{if} & \mathbf{x}[i] \notin [\mathbf{l}[i], \mathbf{u}[i]] \\ \min\left(0, \nabla_{\mathbf{x}_p}\mathcal{F}[i]\right) & \text{if} & (\mathbf{x}[i] = \mathbf{l}[i]) \wedge (\mathbf{l}[i] < \mathbf{u}[i]) \\ \max\left(0, \nabla_{\mathbf{x}_p}\mathcal{F}[i]\right) & \text{if} & (\mathbf{x}[i] = \mathbf{u}[i]) \wedge (\mathbf{l}[i] < \mathbf{u}[i]) \\ \nabla_{\mathbf{x}_p}\mathcal{F}[i] & \text{if} & \mathbf{x}[i] \in]\,\mathbf{l}[i], \mathbf{u}[i]\,[\\ 0 & \text{if} & \mathbf{l}[i] = \mathbf{x}[i] = \mathbf{u}[i] \end{cases} \tag{25}$$

Illustration of the method

We give in Fig. 3 typical convergence behavior for the proposed method applied to problem Eq. 12 or Eq. 17.

We observe the non-monotonic convergence behavior of the Barzilai-Borwein method. As reported by [30], we observe that it is more effective to alternate between BB1 Eq. 22 and BB2 Eq. 23 steps than using only one type of step update.

Results and discussion

Synthetic data

The presented algorithm has been tested on synthetic data. The synthetic spectra consist of $\mathbf{y}$ vectors of $n = 500$ components. Each $\mathbf{y}$ is generated by summing contributions from a synthetic baseline $\mathbf{x}_b$, a synthetic peak list $\mathbf{x}_p$ convolved by a Gaussian peak shape p and a Gaussian noise ϵ:

$$\mathbf{y} = \mathbf{x}_b + \mathbf{x}_p * p + \epsilon$$

The detailed description of these three contributions follows.

- **Synthetic baseline**
  The analytic expression of the baseline is

  $$\mathbf{x}_b[i] = C(s) + s \exp\left(-3\frac{i}{n}\right) - 2\frac{i}{n}$$

  where s can take one of the following values $s = -1$, 0, or 1.
  For $s = 0$ the baseline is a straight line, for $s = -1$ the baseline is concave, for $s = +1$ it is convex.
  The constant $C(s)$ insure a positive spectrum. Its value is $C(s) = 5$ for $s = 1$ and $C(s) = 2$ otherwise.
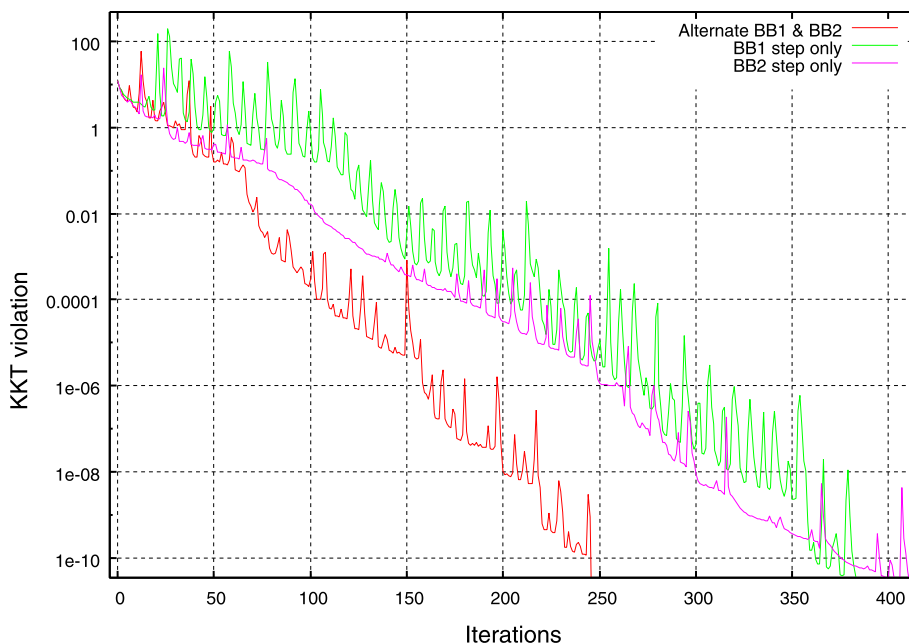

Fig. 3 Typical convergence behavior of the **PBB** algorithm given in Algorithm 2. Alternating between BB1 and BB2 steps leads to faster convergence

Picaud *et al. BMC Bioinformatics*   (2018) 19:123

Page 9 of 20

- **Synthetic peak list**
  The analytic expression of the peak list contribution is:

  $$\left(\mathbf{x}_p * p\right)[i] = \sum_{k=1}^{n_p} \alpha_k p\left(\mu_k, \sigma_p, i\right)$$

  where we have taken $n_p = 10$ and

  $$p\left(\mu_k, \sigma_p, i\right) = \exp\left(-\frac{1}{2}\left(\frac{i - \mu_k}{\sigma_p}\right)^2\right)$$

  These n_p Gaussian peaks are defined by a constant shape factor $\sigma_p = 10$. The individual heights α_k and centers μ_k are tabulated in Table 1.

- **Simulated zero mean Gaussian noise**
  The noise ϵ follows a Normal law of zero mean and σ_{noise} standard deviation:

  $$\epsilon[i] \sim \mathcal{N}\left(0, \sigma_{noise}\right)$$

Algorithm implementation

We provide a reference implementation [1] that can be used to reproduce the results of the following sections. This implementation is coded in C++ and runs under Linux. The main page of the project details all the steps to reproduce the results of the "Joint baseline computation-peak deconvolution", "Comparison with the usual sequential approach" and "Real data" sections. Typical run-times are one second for the synthetic data and three seconds for the examples using real spectra data. To keep this implementation as simple as possible a basic projected gradient descent is used instead of the more effective Algorithm 2. Also note that this implementation requires CSV input files. A more versatile version of our algorithm, using Algorithm 2, is used in [39]. However, this implementation is not yet publicly available.

Joint baseline computation-peak deconvolution

As explained in "Final model formulation" section, page 8, the joint baseline computation and peak deconvolution is performed in two steps:

1. a first resolution of Eq. 20 is performed with a strong λ_1 penalty. The role of this step, is to obtain peak centers from the noisy spectrum. This set of peak centers is denoted by Ω. This step is illustrated in Figs. 4 and 5.
2. a second resolution, is then performed with a null λ_1 penalty. This penalty is replaced by the restricted support Ω for peak centers we found in the first step. The role of this step is to correct peak heights and baseline values which were biased by the presence of the strong λ_1 penalty of the first step. Illustrations are given in Figs. 6 and 7.

The baseline approximation obtained at the end of the first step is given in Fig. 4. This figure shows a common flaw of most of the baseline removal methods which is an ascent of the baseline under the peaks .

The deconvolved peaks obtained after the first step are given in Fig. 5. We notice the negative impact of a strong λ_1 penalty which leads to an underestimation of the peak heights. The figure also shows that the deconvolved solution is less spiky in regions of strong peak overlaps (right part of the spectrum).

The baseline computed after the second step is shown in Fig. 6. During this second stage, the λ_1 penalty is removed and replaced by a restricted support Ω computed using Eq. 16. We see that the ascent of the baseline under the peaks has been corrected (c.g. x-axis ranges from 150 to 250 and from 350 to 400).

The new peak heights are shown in Fig. 7. The main role of this second stage is to debias peak heights. Compared to Fig. 5 we can see that this objective is quite well fulfilled.

Comparison with the usual sequential approach

In this second part we provide a comparison between our new method and a common procedure used to extract peaks. The objective is to validate the approach and to see if there are some advantages to use the joint baseline computation and peak deconvolution over more traditional approaches. Our method essentially depends on 3 hyperparameters λ_1, λ_2 and μ. The parameter λ_2 is of minor importance and is constantly set to 0.1. The two other hyper-parameters are:

1. the λ_1 parameter enforcing the sparsity of the solution $\mathbf{x}_p$,
2. the μ parameter enforcing the smoothness of the baseline $\mathbf{x}_b$.

Table 1 True peak centers μ_k and intensities α_k. All peaks have a common shape factor $\sigma_p = 10$

k	μ_k	α_k
1	50	1
2	90	0.5
3	170	0.5
4	200	3
5	230	2
6	260	1
7	350	0.5
8	370	3
9	390	2
10	410	1

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 10 of 20
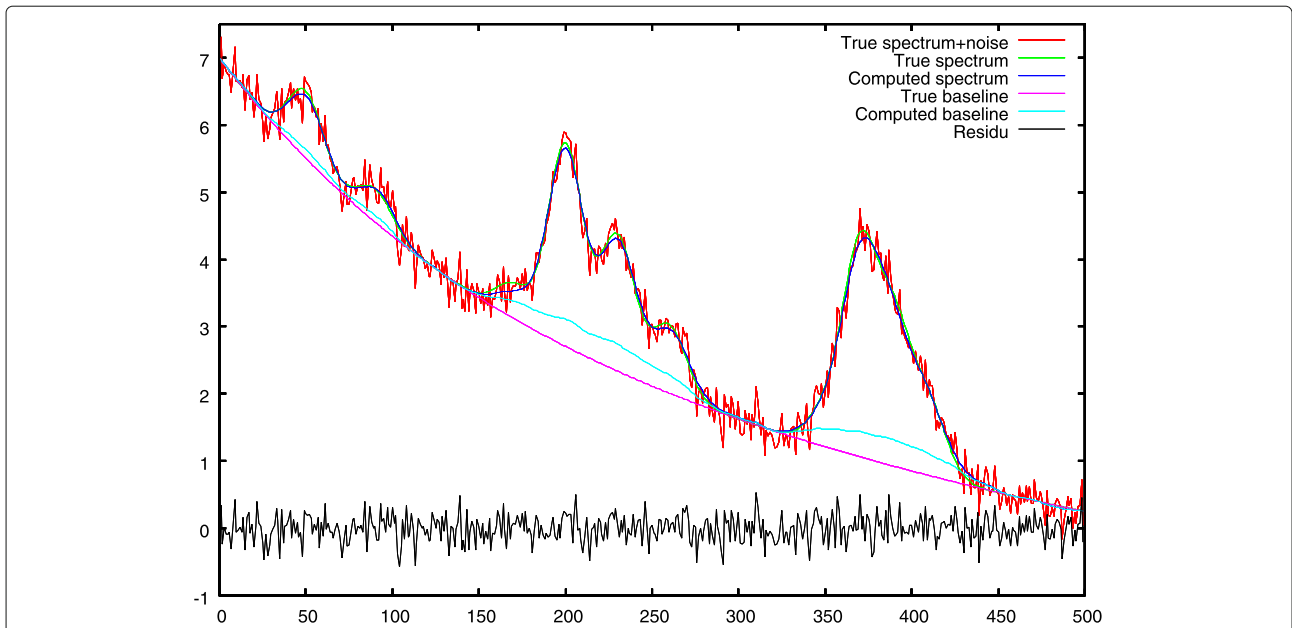
Fig. 4 Step 1: first baseline approximation with $\lambda_1 > 0$, comparison between the true baseline. We notice an ascent of the baseline under the peaks

In order to perform a fair comparison with another approach we tried to stick to an approach that also only uses two hyper-parameters. The usual approaches in spectra processing generally chain at least two procedures [4]. In peculiar, the baseline subtraction procedure is followed by a peak picking procedure.

1 Baseline subtraction: the Statistics-sensitive Non-linear Iterative Peak-clipping (SNIP) algorithm [40–42] is an efficient algorithm to compute spectrum baseline. It generally gives good results and is easy to implement. It uses only one parameter, the window width m_{SNIP}, but requires a smoothed
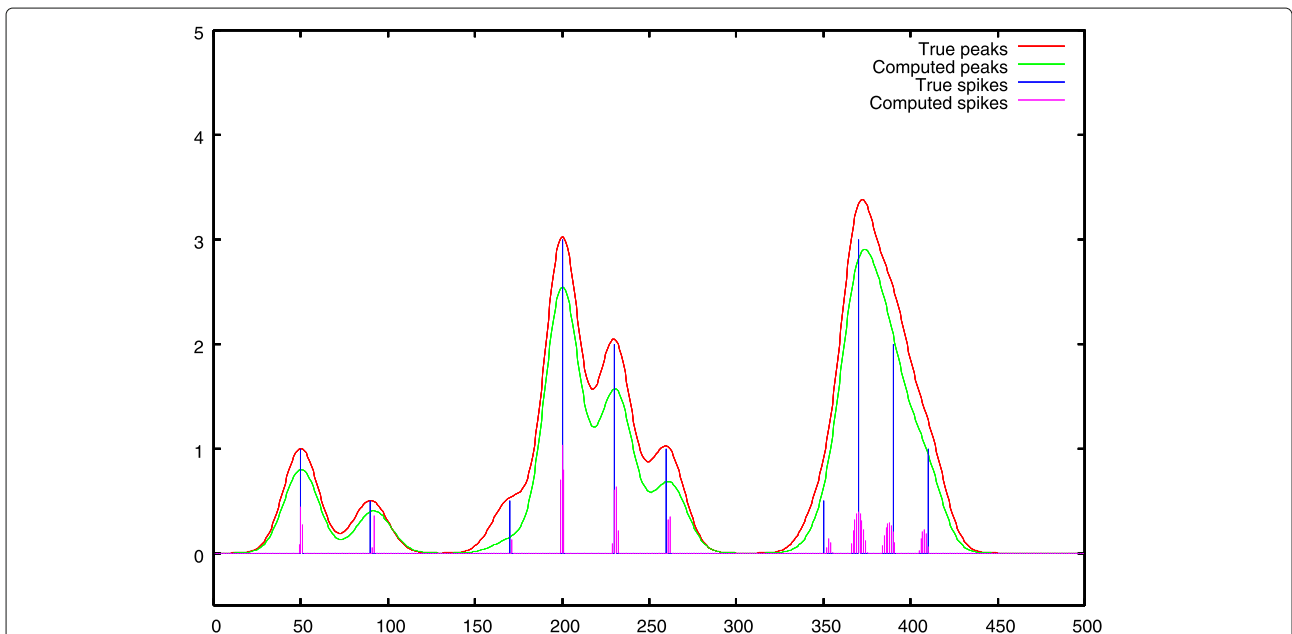
Fig. 5 Step 1: peak selection, comparison between the true peak support and the deconvolved peaks. The negative impact of a strong λ_1 penalty is showed, the peak heights are underestimated

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 11 of 20
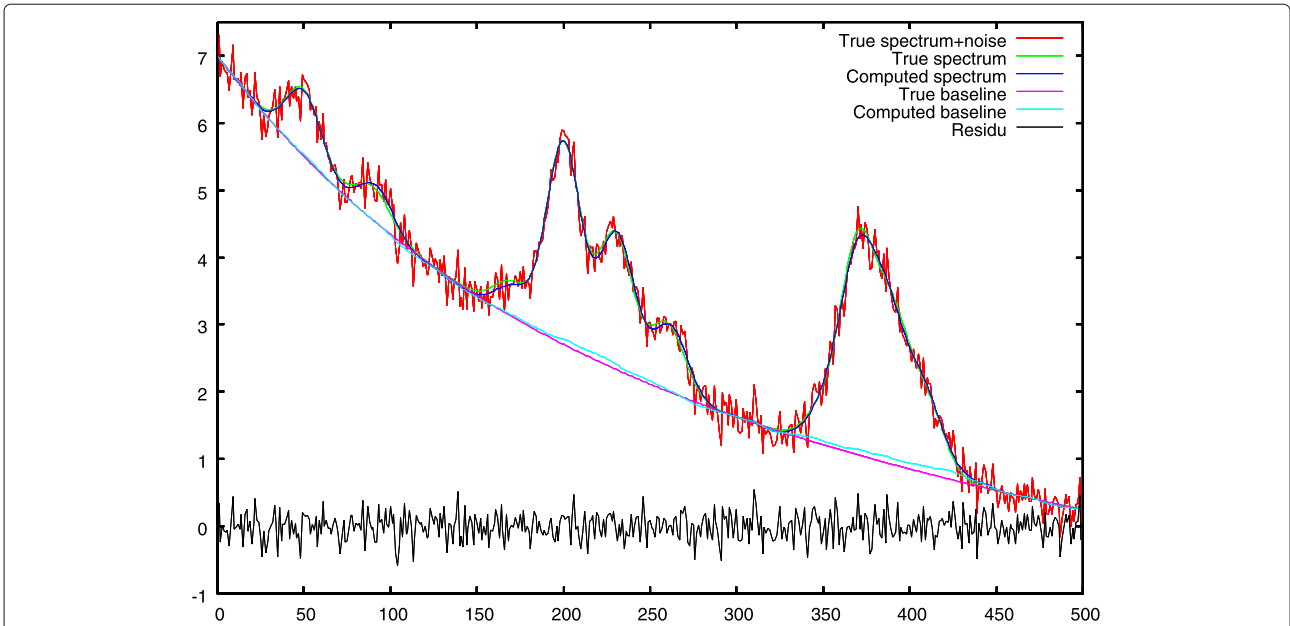
Fig. 6 Step 2: final baseline approximation with $\lambda_1 = 0$ and Ω peak support regularization. The baseline under the peaks has been corrected

spectrum. To smooth the spectrum we use a Savitzky-Golay filter [6, 7]. The idea is to locally fit the spectrum by a polynomial. This least-squares fitted polynomial of degree d is computed using m_{SG} points of a running window. For each window position, the spectrum value at the window center is then replaced by the polynomial value. In order to

have only one parameter, a fixed window width $m_{SG} = 39$ is used for the smoothing step.

2 Peak picking: once the baseline has been subtracted from the original spectrum, there are a large panel of methods to extract peaks [4]. We can mention simple thresholding [42], second derivative computation using Savitzky-Golay filters, extraction from wavelet
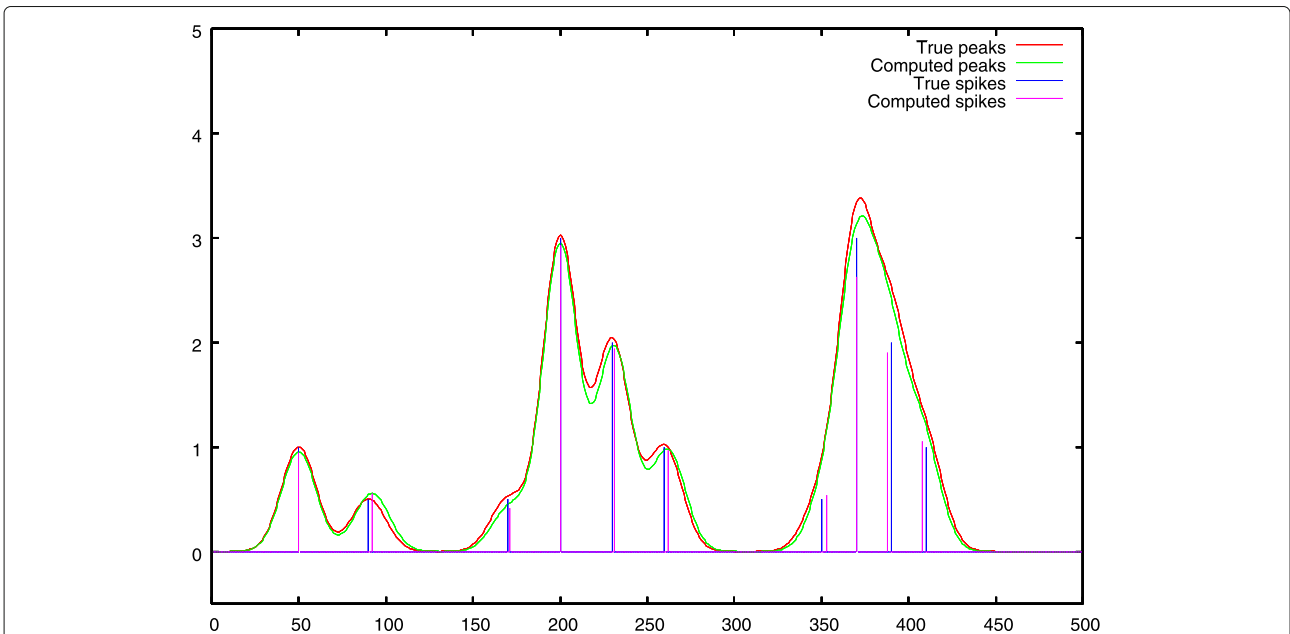
Fig. 7 Step 2: peak debiasing, comparison between the true peak support and the deconvolved peaks

Picaud *et al. BMC Bioinformatics*   (2018) 19:123

Page 12 of 20

coefficients or deconvolution. For our comparison we are more interested by deconvolution-like methods [21, 22] which are closer to our approach. By consequence we have decided to extract peaks thanks to the usual sparse deconvolution problem:

$$\arg\min_{\mathbf{x}_p \geq 0} \frac{1}{2} \left\| \mathbf{y} - \mathbf{L}\mathbf{x}_p \right\|_2^2 + \lambda_1 \left\| \mathbf{x}_p \right\|_1 + \frac{\lambda_2}{2} \left\| \mathbf{x}_p \right\|_2^2 \quad (26)$$

As for our method, the coefficient λ_2 is not critical and we set it to a constant value $\lambda_2 = 0.1$. The remaining λ_1 is our second free parameter. To solve Eq. 26 we use the same Barzilai-Borwein solver, detailed in Algorithm 2 and the same two-steps procedure which consists in peak selection (high λ_1) and peak heights debiasing ($\lambda_1 = 0$).

To compare the two approaches we perform a systematic grid search for the two hyper-parameters. The grid for our new method is given Table 2, whereas Table 3 is used by the usual method, that sequentially removes the baseline and then performs the peak deconvolution.

For each processed spectrum we perform a systematic parameter grid search and report the best result obtained by the two methods. To find the best solution we compare the deconvolved peaks with the ground truth of our synthetic data. We use the following procedure to compare these two peak lists. As defined in our model, a peak list is stored in a sparse vector $\mathbf{x}_p$. To each nonzero component $\mathbf{x}_p[i]$ corresponds a peak at position i and height $x_p[i]$. Like computed peak centers and ground truth peak centers might not be perfectly aligned we reconstruct a continuous signal by convolving the two peak lists by the known peak shape p. This blurring is required later to compute the Euclidean distance between these two peak lists. This reconstruction is performed by the following matrix-vector products: $L\mathbf{x}_p^\star$ for the ground truth peak list and $L\hat{\mathbf{x}}_p$ for the peaks extracted by the algorithm. Using these vectors we can now compute a normalized Euclidean distance defined by:

$$\mathcal{E}_e = \frac{\left\| \mathbf{L}\left(\hat{\mathbf{x}}_p - \mathbf{x}_p^* \right) \right\|_2}{\left\| \mathbf{L}\mathbf{x}_p^* \right\|_2}$$

For each noise level and each algorithm the computations are performed 10 times, each time with a different noise realization. From these 10 replica we report the obtained

Table 2 Our method: used grid for parameter search

Parameters	Min	Max	Step
μ	100	4000	100
λ_1	0	4	0.2

$\lambda_2 = 0.1$ is constant

Table 3 Usual method, baseline removal then peak deconvolution: used grid for parameter search

Parameters	min	max	step
m_{SNIP}	20	40	2
λ_1	1	15	0.2

$\lambda_2 = 0.1$ and $m_{\text{SG}} = 39$ are constant

mean and standard deviation of the $\mathcal{E}_e$ merit factor. The results are given in Table 4 and plotted in Fig. 8.

On this example the joint evaluation of the baseline and of the deconvolved peaks outperforms the sequential approach in nearly all configurations except for the noise-free case. In this condition, as we are using synthetic data, a perfect reconstruction of the "ground truth" spectrum is possible. By consequence, the reconstruction error is very small. To explain this result, we think that our iterative solver had stopped its iterative resolution too early.

To get more intuition on what is happening we provide Fig. 9 illustrating the baseline computation using the SNIP algorithm.

As explained the SNIP algorithm needs to work on a previously smoothed spectrum. This smoothing is performed using a Savitzky-Golay filter. This sequential approach, smoothing then SNIP baseline, can introduce an important bias on the computed baseline. This error is then transferred to the peak picking algorithm. The result can be a poor peak extraction.

Baselines obtained by our method and by the classical SNIP algorithm are shown in Fig. 10.

We see that the simultaneous baseline and deconvolution approach allows a nearly perfect reconstruction of the

Table 4 Comparison between the two methods, for convex baseline $s = -1$

σ_{noise}	Sequential processing	Proposed method
	$\mathcal{E}_e \times 10^{-3}$	$\mathcal{E}_e \times 10^{-3}$
0.	15.	27.6
0.2	47.8 ± 8.76	45.2 ± 9.12
0.4	88.6 ± 15.6	81.6 ± 11.9
0.6	131. ± 21.4	118. ± 18.
0.8	176. ± 26.	154. ± 20.1
1.	217. ± 33.7	184. ± 21.3
1.2	262. ± 45.7	215. ± 25.5
1.4	295. ± 48.2	245. ± 29.4
1.6	330. ± 52.5	275. ± 32.7
1.8	360. ± 59.8	300. ± 38.2
2.	390. ± 70.4	322. ± 44.3

$\mathcal{E}_e$ merit factor, lower is better. Apart from the noise free signal, the joint approach outperforms the sequential one for all noise levels

The italic means the best result (when comparing the two methods)

Fig. 8 Box plot of the results presented in Table 4. $\mathcal{E}_e$ merit factor, lower is better
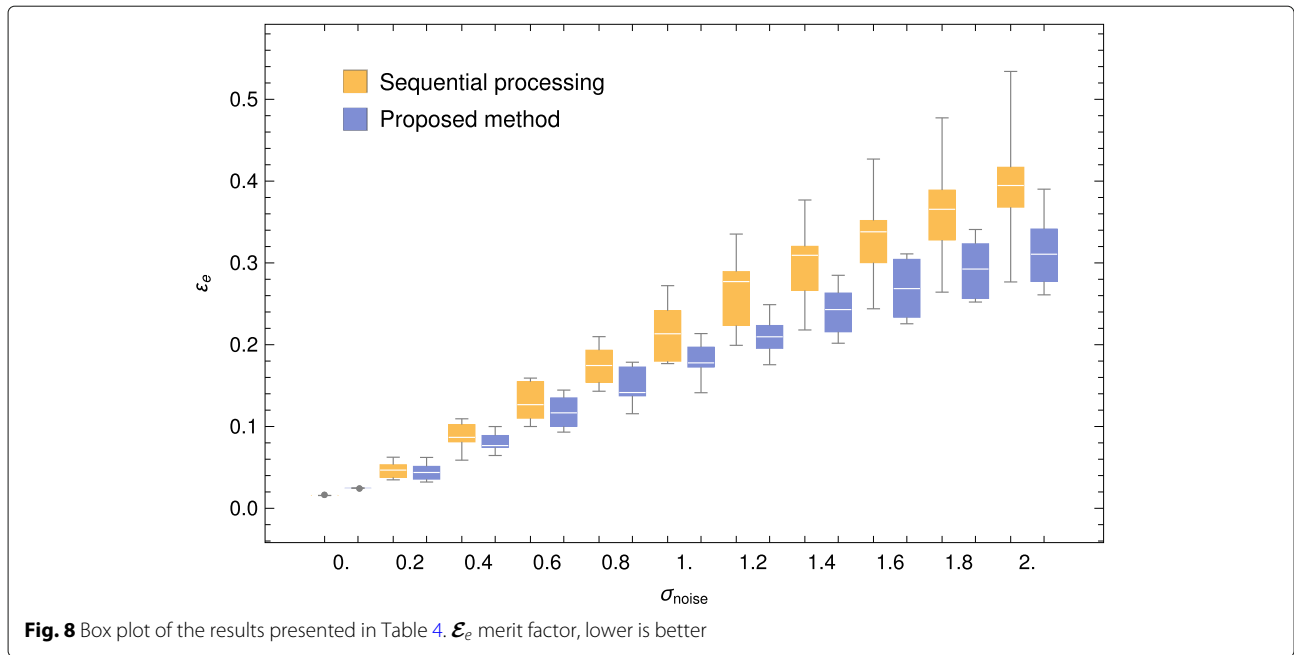
baseline. Unlike our approach, the SNIP algorithm suffers from initial filtering of the spectrum and presents baseline ascent below peaks.

Real data
We present some results obtained by our algorithm on real MALDI-ToF mass spectra. The Fig. 11 shows the peaks

obtained when the low resolution spectrum is deconvolved with a tight peak shape. We have used a Gaussian peak shape with $\sigma_{\text{peak}} = 0.4$ m/z. The goal is to evaluate the behavior of the algorithm when we try to deconvolve isotopic motifs. The notable result here is to see that the returned peak centers are approximately spaced by 1 m/z which is the expected value. This result is encouraging

Fig. 9 Details of the Savitzky-Golay smoothing + SNIP baseline, $\sigma_{noise} = 1$, $m_{\text{SNIP}} = 28$. The baseline removal can introduce serious artifacts. In a sequential approach these artifacts directly affect the peak picking algorithm

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 14 of 20

Fig. 10 Computed baselines comparison. The blue curve is the baseline obtained by our method whereas the black one is the baseline computed by the SNIP algorithm. Our method is able the recover a nearly exact baseline unlike the usual one that presents artifacts and ascent below peaks. This result is for $\sigma_{noise} = 1, s = -1$ (convex baseline) and parameters given in Table 4

as this information was not provided to the algorithm to perform its deconvolution.

However isotopic motif deconvolution without using any extra information (like an expected 1 m/z spacing between peaks) can lack robustness. That is the reason why it is certainly safer to use a wider peak shape to model the unresolved isotopic motif as a whole. This is illustrated by Fig. 12. We have run the algorithm twice. The first run used a sparsity promoting regularization of $\lambda_1 = 1$, the second run used $\lambda_1 = 0.5$.

The developed algorithm can also be used with high resolution MALDI-ToF mass spectrum (reflectron mode). Figure 13 gives such an example. A Gaussian peak shape with $\sigma_{peak} = 0.15$ m/z is used for the deconvolution. As before, we have used two values for the sparsity promoting regularization. The first run used $\lambda_1 = 0.5$, the second run used $\lambda_1 = 0.2$. We clearly see the impact on the results, in the first case only the main peaks are extracted, in the second one very small peaks are also extracted. For the moment this λ_1 parameter has to be manually tuned by the user.

It is generally quite difficult to evaluate peak picking methods on real spectra because we do not have the ground truth at our disposal. However, in our case we have expensively used the proposed approach in the BHI-PRO project. The study [39] compares our method against a more usual one on mass spectra of spiked proteins. This study quantifies the algorithmic part of the variance of the

measured protein abundances and shows a clear gain in favor of our algorithm.

Conclusions

We have introduced a new method for peak deconvolution. This new approach jointly performs baseline computation and peak deconvolution. The baseline equation can be solved in a closed form and is substituted into the deconvolution equation. The new deconvolution equation contains a linear operator $\mathbf{A}_\mu$ acting as a smoothed second order derivator. The problem of boundaries is exposed and tackled rigorously leading to a modified equation. The problem is efficiently solved by the projected Barzilai-Borwein algorithm. A comparison with a traditional approach relying on a sequential baseline removal and peak picking is detailed. The benefits of our new approach are put in evidence, including a better baseline approximation avoiding ascent below the large peaks, and a better reconstruction of the deconvolved peaks. Finally the method has been tested on real data.

Perspectives

There are two main directions we want to explore. The first direction is to go one step further in a joint processing approach by allowing automatic adjustment of the peak shape function. Ideally, with such an approach, the baseline, the peak shape and the deconvolved peaks would be computed jointly. The second direction would be to devise

Fig. 11 Low resolution MALDI-ToF spectrum, isotopic motifs deconvolution obtained for tight Gaussian peak shape of $\sigma_{\text{peak}} = 0.4 \, m/z$

a procedure for automatic tuning of the two main hyper-parameters. The resulting algorithm would be of great value to process a large number of spectra in batch mode.

Endnotes

[1] https://github.com/vincent-picaud/Joint_Baseline_PeakDeconv

[2] No off-diagonal element is null

Appendix

Smoothness and convolution matrix

To enforce baseline $\mathbf{x}_b$ smoothness the first order finite differences matrix $\mathbf{D}$ of dimension $(n-1) \times n$

$$\mathbf{D} = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \\ & & & & 1 & -1 \end{pmatrix}$$

has been introduced. For ease of reading we also provide the $\mathbf{D}^t\mathbf{D}$ matrix expression. Its dimension is $n \times n$ and its components are:

$$\mathbf{D}^t\mathbf{D} = \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix}$$

The matrices $\mathbf{D}^t\mathbf{D}$ and $\mathbf{B}_\mu = \mathbf{I}_b + \mu\mathbf{D}^t\mathbf{D}$ are tridiagonal matrices. Hence the associated matrix-vector products can be computed efficiently. The Thomas algorithm [43], which is a Gaussian elimination optimized for tridiagonal matrices, also provides an efficient way to compute the matrix-vector products associated to the $\mathbf{A}_\mu = \mathbf{I}_b - \mathbf{B}_\mu^{-1}$ operator.

In the presentation the convolution product between the peak list and peak shape function, $\mathbf{x}_p * p$, is often represented by the $\mathbf{L}\mathbf{x}_p$ matrix-vector product. However,

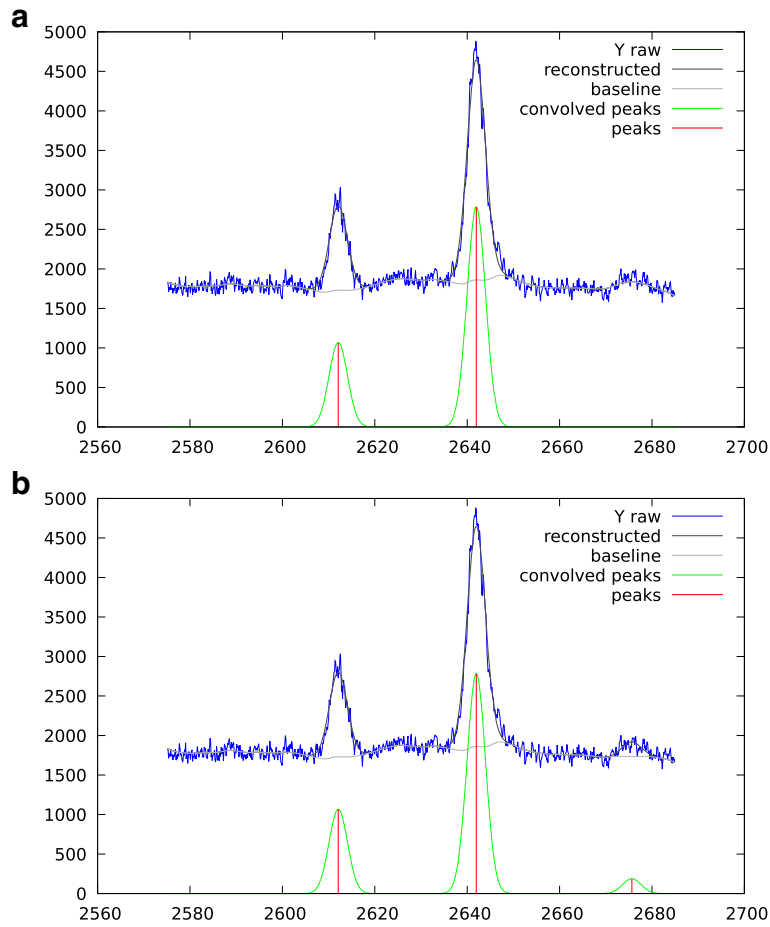Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 16 of 20

Fig. 12 Low resolution MALDI-ToF spectrum, deconvolution obtained when considering unresolved isotopic motifs as an unique peak. The Gaussian peak shape is $\sigma_{\text{peak}} = 2\ m/z$. The regularization is $\lambda_1 = 1$ for the top plot and $\lambda_1 = 0.5$ for the bottom one

in practical computations the **L** matrix is never explicitly formed and the convolution product is computed directly using a specialized subroutine.

Derivation of reduced objective

This appendix focuses on the reduced criterion Eq. 10.

Let φ be a function which takes as input a vector **z** of size n and produces as output the scalar quantity

$$\varphi(\mathbf{z}) = \frac{1}{2}\mathbf{z}^t\mathbf{Q}\mathbf{z} + \mathbf{q}^t\mathbf{z} + q_0$$

where q_0 is a scalar, **q** is a vector of size n and **Q** is a strictly positive-definite symmetric matrix of size $n \times n$. The minimizer of φ is $\hat{\mathbf{z}} = -\mathbf{Q}^{-1}\mathbf{q}$ and the minimum is

$$\varphi\left(\hat{\mathbf{z}}\right) = -\frac{1}{2}\mathbf{q}^t\mathbf{Q}^{-1}\mathbf{q} + q_0$$

If we introduce the gradient $\nabla\varphi(\mathbf{z}) = \mathbf{Q}\mathbf{z} + \mathbf{q}$ and the Hessian $\nabla^2\varphi(\mathbf{z}) = \mathbf{Q}$ we notice that the previous relation takes the following form:

$$\varphi(\hat{\mathbf{z}}) = -\frac{1}{2}\nabla\varphi(\mathbf{0})^t\left[\nabla^2\varphi(\mathbf{0})\right]^{-1}\nabla\varphi(\mathbf{0}) + \varphi(\mathbf{0}) \quad (27)$$

If we consider Eq. 6 as a function of $\mathbf{x}_b$

$$\mathbf{x}_b \rightarrow \mathcal{J}\left(\mathbf{x}_b, \mathbf{x}_p\right) = \frac{1}{2}\left\|\bar{\mathbf{y}} - \mathbf{x}_b\right\|_2^2 + \frac{\mu}{2}\left\|\mathbf{D}\mathbf{x}_b\right\|_2^2 + q_0$$

where

$$\bar{\mathbf{y}} = \mathbf{y} - \mathbf{L}\mathbf{x}_p$$

$$q_0 = \lambda_1\left\|\mathbf{x}_p\right\|_1 + \frac{\lambda_2}{2}\left\|\mathbf{x}_p\right\|_2^2$$

and

$$\mathcal{J}(\mathbf{0}, \mathbf{x}_p) = \frac{1}{2}\|\bar{\mathbf{y}}\|_2^2 + q_0$$

$$\nabla_{\mathbf{x}_b}\mathcal{J}(\mathbf{0}, \mathbf{x}_p) = -\bar{\mathbf{y}}$$

$$\nabla_{\mathbf{x}_b}^2\mathcal{J}(\mathbf{0}, \mathbf{x}_p) = \mathbf{B}_\mu$$

then application of Eq. 27 directly leads to Eq. 10:

$$\widetilde{\mathcal{J}}\left(\mathbf{x}_p\right) = \mathcal{J}\left(\hat{\mathbf{x}}_b, \mathbf{x}_p\right) + \mathbf{C}$$

with a constant term $\mathbf{C} = -\frac{1}{2}\mathbf{y}^t\mathbf{A}_\mu\mathbf{y}$.
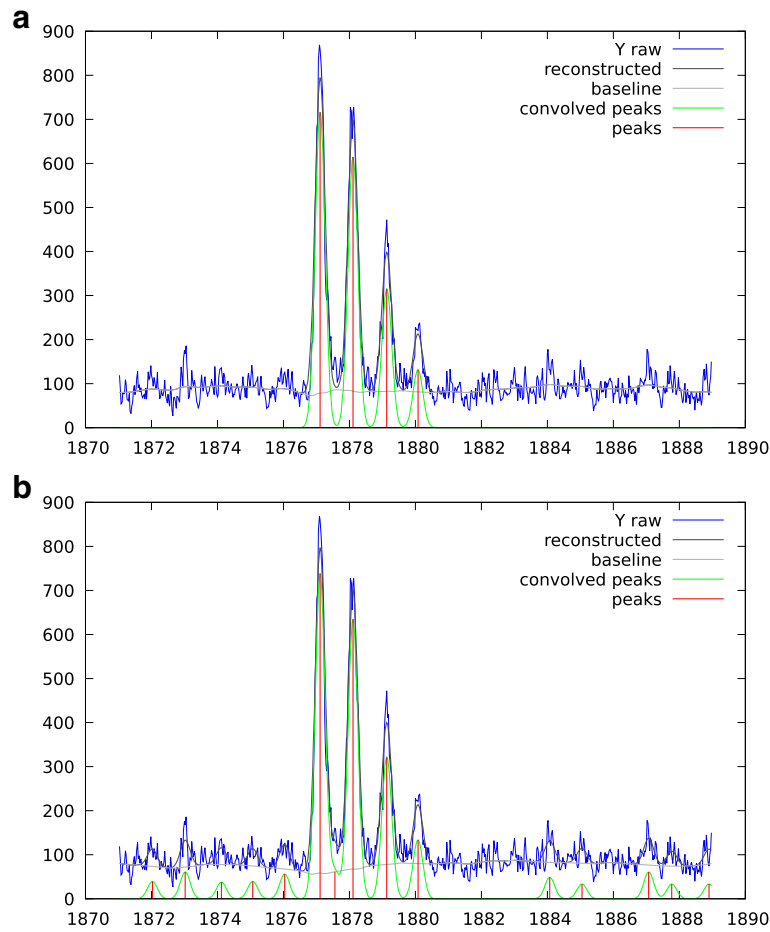
Fig. 13 High resolution MALDI-ToF spectrum, peak extraction with a Gaussian peak shape of $\sigma_{peak} = 0.15\ m/z$. The regularization is $\lambda_1 = 0.5$ for the top plot and $\lambda_1 = 0.2$ for the bottom one

Expression of $\lim_{\mu \to \infty} \mathbf{A}_\mu$

The matrix $\mathbf{D}^t\mathbf{D}$ is real and symmetric, hence it admits an eigen decomposition

$$\mathbf{D}^t\mathbf{D} = \mathbf{V}\Delta\mathbf{V}^* = \sum_{k=1}^{n} \nu_k \mathbf{v}_k \otimes \mathbf{v}_k$$

where Δ is a diagonal matrix of eigenvalues ν_k and $\mathbf{V}$ is an orthonormal matrix. Each column k of V is an eigenvector $\mathbf{v}_k$.

Let's write down two basic statements about eigenpair $(\nu, \mathbf{v})$ of any square matrix M:

- the first one, where I denotes the identity matrix, is:

$$M\mathbf{v} = \nu\mathbf{v} \Leftrightarrow (I + M)\mathbf{v} = (1 + \nu)\mathbf{v}$$

- the second one, that requires M to be invertible (hence $\nu \neq 0$), is:

$$M\mathbf{v} = \nu\mathbf{v} \Leftrightarrow M^{-1}\mathbf{v} = \frac{1}{\nu}\mathbf{v}$$

Now observe that like $\mathbf{I}_b + \mu\mathbf{D}^t\mathbf{D}$ is a strictly diagonally dominant matrix it is invertible [44]. Using the two previously enumerated relations it is easy to show that this matrix admits the following eigen decomposition:

$$\left(\mathbf{I}_b + \mu\mathbf{D}^t\mathbf{D}\right)^{-1} = \sum_{k=1}^{n} \frac{1}{1 + \mu\nu_k} \mathbf{v}_k \otimes \mathbf{v}_k$$

By construction (first order finite differences) the product of the matrix $\mathbf{D}$ with a constant vector is the null vector. It follows that we have $\mathbf{D}^t\mathbf{D}\,\mathbf{v}_1 = \mathbf{0}$ where $\mathbf{v}_1 = \frac{1}{\sqrt{n}}\mathbb{1}_n$. Thus $\mathbf{v}_1$ is the normalized eigen vector associated to the $\nu_1 = 0$ eigenvalue. Moreover, as $\mathbf{D}^t\mathbf{D}$ is an *unreduced*[2] real symmetric tridiagonal matrix, all its eigenvalues are simples ([45], Lemma 7.7.1). By consequence we can write:

Picaud *et al. BMC Bioinformatics*   (2018) 19:123

Page 18 of 20

$$\left(\mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D}\right)^{-1} = \mathbf{v}_1 \otimes \mathbf{v}_1 + \sum_{k=2}^{n} \frac{1}{1 + \mu \nu_k} \mathbf{v}_k \otimes \mathbf{v}_k$$

with $\nu_k \neq 0$ for all $k \geq 2$. It is then trivial to compute the limit:

$$\lim_{\mu \to \infty} \left(\mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D}\right)^{-1} = \mathbf{v}_1 \otimes \mathbf{v}_1 = \frac{1}{n} \mathbb{1}_{n \times n}$$

which is a square matrix of dimensions $n \times n$ with all entries equal to 1.

From the definition of $\mathbf{A}_\mu$ in Eq. 11 we get the expected result:

$$\lim_{\mu \to \infty} [\mathbf{A}_\mu]_{i,j} = \delta_{i,j} - \frac{1}{n}$$

with $\delta_{i,j}$ equals to 1 if $i = j$ and 0 otherwise.

Boundary correction

Our boundary correction approach exploits the opportunity to convert an equality constrained problem

$$\arg \min_{\mathbf{x}, \, \mathbf{x}[k] = \bar{y}_k} \frac{1}{2} \mathbf{x}^t \mathbf{Q} \mathbf{x} + \mathbf{x}^t \mathbf{q} \qquad (28)$$

into a unconstrained modified problem

$$\arg \min_{\mathbf{x}} \frac{1}{2} \mathbf{x}^t \widetilde{\mathbf{Q}} \mathbf{x} + \mathbf{x}^t \widetilde{\mathbf{q}} \qquad (29)$$

Let begin by writing the Lagrangian [38] associated to Eq. 28.

$$\mathcal{L}(\mathbf{x}, \lambda_k) = \frac{1}{2} \mathbf{x}^t \widetilde{\mathbf{Q}} \mathbf{x} + \mathbf{x}^t \widetilde{\mathbf{q}} + \lambda_k \mathbf{e}_k^t (\mathbf{x}[k] - \bar{y}_k)$$

The KKT conditions are then easily obtained by differentiating the Lagrangian and by imposing constraint satisfaction:

$$(\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda_k) = \mathbf{Q}\mathbf{x} + \mathbf{q} + \lambda_k \mathbf{e}_k = 0) \wedge (\mathbf{x}[k] = \bar{y}_k)$$

where λ_k is the Lagrangian multiplier associated to the constraint $\mathbf{x}[k] = \bar{y}_k$ and $\mathbf{e}_k$ the vector of component $\delta_k(i)$. We can expand the $\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda_k) = 0$ equation to get:

$$\left(\begin{array}{c|c|c} \mathbf{Q}^- & \mathbf{v} & \mathbf{U}^t \\ \hline \mathbf{v}^t & \mathbf{Q}(k,k) & \mathbf{w}^t \\ \hline \mathbf{U} & \mathbf{w} & \mathbf{Q}^+ \end{array}\right) \left(\begin{array}{c} \mathbf{x}^- \\ \mathbf{x}[k] \\ \mathbf{x}^+ \end{array}\right) + \left(\begin{array}{c} \mathbf{q}^- \\ \mathbf{q}[k] \\ \mathbf{q}^+ \end{array}\right) + \left(\begin{array}{c} 0 \\ \lambda_k \\ 0 \end{array}\right) = 0$$

It is then obvious that if we take:

$$\lambda_k = -\left(\mathbf{v}^t \mathbf{x}^- + \bar{y}_k \mathbf{Q}(k,k) + \mathbf{w}^t \mathbf{x}^+ + \mathbf{q}[k]\right)$$

then the constraint $\mathbf{x}[k] = \bar{y}_k$ is fulfilled. Substituting this into the $\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda_k) = 0$ equation we get:

$$\left(\begin{array}{c|c|c} \mathbf{Q}^- & \mathbf{v} & \mathbf{U}^t \\ \hline 0 & \mathbf{Q}(k,k) & 0 \\ \hline \mathbf{U} & \mathbf{w} & \mathbf{Q}^+ \end{array}\right) \left(\begin{array}{c} \mathbf{x}^- \\ \mathbf{x}[k] \\ \mathbf{x}^+ \end{array}\right) + \left(\begin{array}{c} \mathbf{q}^- \\ -\bar{y}_k \mathbf{Q}(k,k) \\ \mathbf{q}^+ \end{array}\right) = 0$$

This equation is usable but its drawback is that we have lost the symmetry of the modified $\mathbf{Q}$ matrix. This can be fixed by another modification of the vector term $\mathbf{q}$. Exploiting the fact that by construction $\mathbf{x}[k] = \bar{y}_k$ we have the following equivalent system:

$$\left(\begin{array}{c|c|c} \mathbf{Q}^- & 0 & \mathbf{U}^t \\ \hline 0 & \mathbf{Q}(k,k) & 0 \\ \hline \mathbf{U} & 0 & \mathbf{Q}^+ \end{array}\right) \left(\begin{array}{c} \mathbf{x}^- \\ \mathbf{x}[k] \\ \mathbf{x}^+ \end{array}\right) + \left(\begin{array}{c} \mathbf{q}^- + \bar{y}_k \mathbf{v} \\ -\bar{y}_k \mathbf{Q}(k,k) \\ \mathbf{q}^+ + \bar{y}_k \mathbf{w} \end{array}\right) = 0 \qquad (30)$$

By identification we can write down the announced result:

$$\arg \min_{\mathbf{x}} \frac{1}{2} \mathbf{x}^t \widetilde{\mathbf{Q}} \mathbf{x} + \mathbf{x}^t \widetilde{\mathbf{q}}$$

which is an unconstrained system for which the minimizer satisfies $\mathbf{x}[k] = \bar{y}_k$. For presentation clarity we only have considered one constraint, but this scheme can be used sequentially for any set of constraints

$$\left\{\mathbf{x}[k] = \bar{y}_k, \ k \in K\right\}$$

Now we can see how to use this technique to our deconvolution problem. We must go back to Eq. 8:

$$\nabla_{\mathbf{x}_b} \widetilde{J} = \underbrace{\left(\mathbf{I}_b + \mu \mathbf{D}^t \mathbf{D}\right)}_{\mathbf{B}_\mu} \mathbf{x}_b + \mathbf{L} \mathbf{x}_p - \mathbf{y}$$

By identification the role of the matrix $\mathbf{Q}$ is played by the $\mathbf{B}_\mu$ matrix. Furthermore all the modifications done on the vector $\mathbf{q}$ can be transposed to modifications associated to the $-\mathbf{y}$ vector.

As an illustrative example, we consider the most usual case that consists of imposing the left $\bar{y}_1$ and right $\bar{y}_n$ baseline values at boundaries. Direct application of Eq. 30 provides the result. The modified $\mathbf{B}_\mu$ matrix is:

$$\widetilde{\mathbf{B}}_\mu = \begin{pmatrix} 1+\mu & 0 & & & & & \\ 0 & 1+2\mu & -\mu & & & & \\ & -\mu & 1+2\mu & -\mu & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & -\mu & 1+2\mu & -\mu & \\ & & & & -\mu & 1+2\mu & 0 \\ & & & & & 0 & 1+\mu \end{pmatrix} \qquad (31)$$

The modified $\mathbf{y}$ vector is:

$$\widetilde{\mathbf{y}} = \begin{pmatrix} (1+\mu)\bar{y}_1 \\ \mathbf{y}[2] + \mu \bar{y}_1 \\ \mathbf{y}[3] \\ \vdots \\ \mathbf{y}[n-2] \\ \mathbf{y}[n-1] + \mu \bar{y}_n \\ (1+\mu)\bar{y}_n \end{pmatrix} \qquad (32)$$

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 19 of 20

With these modifications, the modified Eq. 9 that allows to retrieve the baseline $\mathbf{x}_b$ from the deconvolved peaks $\mathbf{x}_p$ is now:

$$\mathbf{x}_b = \widetilde{\mathbf{B}}_\mu^{-1}\left(\widetilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_p\right)$$

The modified $\mathbf{A}_\mu$ operator Eq. 11 is computed as before

$$\widetilde{\mathbf{A}}_\mu = \mathbf{I}_b - \widetilde{\mathbf{B}}_\mu^{-1} \tag{33}$$

The final quadratic form Eq. 13 must be replaced by:

$$\widetilde{\mathcal{J}}_2(\mathbf{x}_p) = \frac{1}{2}\mathbf{x}_p^t\left(\lambda_2\mathbf{I}_p + \mathbf{L}^t\widetilde{\mathbf{A}}_\mu\mathbf{L}\right)\mathbf{x}_p + \\ \mathbf{x}_p^t\left(\lambda_1\mathbb{1}_p - \mathbf{L}^t\widetilde{\mathbf{A}}_\mu\widetilde{\mathbf{y}} - \mathbf{L}^t\left(\mathbf{y} - \widetilde{\mathbf{y}}\right)\right) \tag{34}$$

Abbreviations
BB: Barzilai-Borwein; BHI-PRO: Bayesian hierarchical inversion for mass spectrometry. Application to discovery and validation of new PROtein biomarkers; CWT: Continuous wavelet transform; DWT: Discrete wavelet transform; KKT: Karush-Kuhn-Tucker; LC: Liquid chromatography; MALDI-ToF: Matrix-assisted laser desorption/ionization-time of flight; MS: Mass spectrometry; SG: Savitzky-golay; SNIP: Statistics-sensitive non-linear iterative peak-clipping; UDWT: Undecimated wavelet transform

Availability of data and materials
The datasets used to generate the figures and a basic implementation of the proposed algorithm are avalaible under the https://github.com/vincent-picaud/Joint_Baseline_PeakDeconv GitHub repository.

Authors' contributions
VP initial idea of the method, theoretical developments and C++ implementation. The contributions of JFG and AG regard (a) the design of the criterion and its optimization and (b) the writing of the paper. CT read and approved the final manuscript and proposed valuable comments about pre-processing of spectra in the context of mass spectrometry. CM participated in the problem definition and evaluated the methods on real spectra. JPC provided input in the MALDI-ToF field and the result interpretation. He acquired and provided real MALDI-ToF spectra to test the algorithm. PG was the BHI-PRO project manager. He has coordinated the conception and design of the processing algorithms, in particular the design of the acquisition chain MALDI and MRM models and inversion algorithms, and the interpretation of the data. He has revised the manuscript. All authors read and approved the final manuscript.

Ethics approval and consent to participate
Not applicable

Consent for publication
Not applicable.

Competing interests
JPC is employed by bioMérieux. The other authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details
[1]CEA, LIST, Laboratoire d'Intégration de Systèmes et des Technologies, DIGITEO Bât 565 - Point Courrier 192, 91191 Gif-sur-Yvette Cedex, France. [2]University of Bordeaux, IMS, UMR 5218, 33400 Talence, France. [3]CLIPP, Pôle de Recherche Université de Bourgogne, 21000 Dijon, France. [4]Technology Research Department, Innovation Unit, Marcy l'Étoile, bioMérieux SA, 69280 Marcy l'Étoile, France. [5]Université Grenoble Alpes, 38000 Grenoble, France. [6]CEA, LETI, MINATEC Campus, DTBS, 17 Rue des Martyrs, 38054 Grenoble, France. [7]Service de Biostatistique, Hospices Civils de Lyon, 69000 Lyon, France. [8]Université Lyon 1, 69100 Villeurbanne, France. [9]CNRS UMR5558, Laboratoire de Biométrie et Biologie Évolutive, Équipe Biostatistique-Santé, 69100 Villeurbanne, France.

References
1. van Belkum A, Welker M, Pincus D, Charrier JP, Girard V. Matrix-assisted laser desorption ionization time-of-flight mass spectrometry in clinical microbiology: What are the current issues? Ann Lab Med. 2017;37(6):475.
2. Ilina EN, Borovskaya AD, Malakhova MM, Vereshchagin VA, Kubanova AA, Kruglov AN, Svistunova TS, Gazarian AO, Maier T, Kostrzewa M, Govorun VM. Direct bacterial profiling by matrix-assisted laser desorption-ionization time-of-flight mass spectrometry for identification of pathogenic neisseria. J Mol Diagn. 2009;11(1):75–86.
3. Singhal N, Kumar M, Kanaujia PK, Virdi JS. Maldi-tof mass spectrometry: an emerging technology for microbial identification and diagnosis. Front Microbiol. 2015;6:791. https://doi.org/10.3389/fmicb.2015.00791.
4. Yang C, He Z, Yu W. Comparison of public peak detection algorithms for maldi mass spectrometry data analysis. BMC Bioinformatics. 2009;10(1):4.
5. Larive CK, Sweedler JV. Celebrating the 75th anniversary of the acs division of analytical chemistry: A special collection of the most highly cited analytical chemistry papers published between 1938 and 2012. Anal Chem. 2013;85(9):4201–2.
6. Savitzky A, Golay MJ. Smoothing and differentiation of data by simplified least squares procedures. Anal Chem. 1964;36(8):1627–39.
7. Gorry PA. General least-squares smoothing and differentiation by the convolution (savitzky-golay) method. Anal Chem. 1990;62(6):570–3.
8. Browne M, Mayer N, Cutmore TR. A multiscale polynomial filter for adaptive smoothing. Digit Signal Process. 2007;17(1):69–75.
9. Barak P. Smoothing and differentiation by an adaptive-degree polynomial filter. Anal Chem. 1995;67(17):2758–62.
10. Shensa MJ. The discrete wavelet transform: wedding the a trous and mallat algorithms. IEEE Trans Signal Process. 1992;40(10):2464–82.
11. Nason GP, Silverman BW. The stationary wavelet transform and some statistical applications. In: Antoniadis A, Oppenheim G, editors. Wavelets and Statistics. New York: Springer; 1995. p. 281–99. https://doi.org/10.1007/978-1-4612-2544-7_17.
12. Coombes KR, Tsavachidis S, Morris JS, Baggerly KA, Kuerer HM. Improved peak detection and quantification of mass spectrometry data acquired from surface-enhanced laser desorption and ionization by denoising spectra with the undecimated discrete wavelet transform. Proteomics. 2005;5:4107–17.
13. Antoniadis A, Bigot J, Lambert-Lacroix S. Peaks detection and alignment for mass spectrometry data. J Société Française Stat. 2010;151(1):17–37.
14. Perez-Pueyo R, Soneira MJ, Ruiz-Moreno S. Morphology-based automated baseline removal for raman spectra of artistic pigments. Appl Spectrosc. 2010;64(6):595–600.
15. Morháč M, Matoušek V. Peak clipping algorithms for background estimation in spectroscopic data. Appl Spectrosc. 2008;62(1):91–106.
16. Mazet V, Carteret C, Brie D, Idier J, Humbert B. Background removal from spectra by designing and minimising a non-quadratic cost function. Chemometr Intell Lab Syst. 2005;76(2):121–33.

Picaud *et al. BMC Bioinformatics* (2018) 19:123

Page 20 of 20

17. Ruckstuhl AF, Jacobson MP, Field RW, Dodd JA. Baseline subtraction using robust local regression estimation. J Quant Spectrosc Radiative Transf. 2001;68(2):179–93.
18. Li Z, Zhan DJ, Wang JJ, Huang J, Xu QS, Zhang ZM, Zheng YB, Liang YZ, Wang H. Morphological weighted penalized least squares for background correction. Analyst. 2013;138(16):4483–92.
19. Dubrovkin J. Evaluation of the peak location uncertainty in second-order derivative spectra. Case study: symmetrical lines. J Emerg Technol Comput Appl Sci. 2014;3:9.
20. Mohammad-Djafari A, Giovannelli JF, Demoment G, Idier J. Regularization, maximum entropy and probabilistic methods in mass spectrometry data processing problems. Int J Mass Spectrom. 2002;215(1):175–93.
21. Renard BY, Kirchner M, Steen H, Steen JA, Hamprecht FA. Nitpick: peak identification for mass spectrometry data. BMC Bioinformatics. 2008;9(1):355.
22. Slawski M, Hussong R, Tholey A, Jakoby T, Gregorius B, Hildebrandt A, Hein M. Isotope pattern deconvolution for peptide mass spectrometry by non-negative least squares/least absolute deviation template matching. BMC Bioinformatics. 2012;13(1):291.
23. Giovannelli JF, Coulais A. Positive deconvolution for superimposed extended source and point sources. Astron Astrophys. 2005;439: 401–12.
24. Hirsch M, Schölkopf B, Habeck M. A blind deconvolution approach for improving the resolution of cryo-em density maps. J Comput Biol. 2011;18(3):335–46.
25. Lange E, Gröpl C, Reinert K, Kohlbacher O, Hildebrandt A. High-Accuracy Peak Picking Of Proteomics Data Using Wavelet Techniques. In: Biocomputing 2006. World Scientific; 2012. p. 243–54. https://www.worldscientific.com/doi/abs/10.1142/9789812701626_0023.
26. Du P, Kibbe WA, Lin SM. Improved peak detection in mass spectrum by incorporating continuous wavelet transform-based pattern matching. Bioinformatics. 2006;22(17):2059–65.
27. Mallat S, Zhong S. Characterization of signals from multiscale edges. IEEE Trans Pattern Anal Mach Intell. 1992;7:710–32.
28. Figueiredo MA, Nowak RD, Wright SJ. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems. IEEE J Sel Top Sign Process. 2007;1(4):586–97.
29. Tibshirani R. Regression shrinkage and selection via the lasso. J R Stat Soc Ser B Methodol. 1996;58(1):267–88. http://www.jstor.org/stable/2346178.
30. Dai YH, Fletcher R. Projected barzilai-borwein methods for large-scale box-constrained quadratic programming. Numer Math. 2005;100(1): 21–47.
31. Barzilai J, Borwein JM. Two-point step size gradient methods. IMA J Numer Anal. 1988;8(1):141–8.
32. Fletcher R. On the barzilai-borwein method. In: Qi L, Teo K, Yang X, editors. Optimization and Control with Applications. Boston: Springer US; 2005. p. 235–56.
33. Raydan M. The barzilai and borwein gradient method for the large scale unconstrained minimization problem. SIAM J Optim. 1997;7(1):26–33.
34. Raydan M. On the barzilai and borwein choice of steplength for the gradient method. IMA J Numer Anal. 1993;13(3):321–6.
35. Moré JJ, Toraldo G. On the solution of large quadratic programming problems with bound constraints. SIAM J Optim. 1991;1(1):93–113.
36. Kim D, Sra S, Dhillon IS. A non-monotonic method for large-scale non-negative least squares. Optim Methods Softw (OMS). 2012.
37. Birgin EG, Martínez JM, Raydan M. Nonmonotone spectral projected gradient methods on convex sets. SIAM J Optim. 2000;10(4):1196–211.
38. Boyd S, Vandenberghe L. Convex optimization. Cambridge university press; 2009.
39. Mercier C, Klich A, Truntzer C, Picaud V, Giovannelli J-F, Ducoroy P, Grangeat P, Maucort-Boulch D, Roy P. Variance component analysis to assess protein quantification in biomarker discovery. application to MALDI-TOF mass spectrometry. Biom J. 2017;60(2):262–74. https://doi.org/10.1002/bimj.201600198.
40. Ryan C, Clayton E, Griffin W, Sie S, Cousens D. Snip, a statistics-sensitive background treatment for the quantitative analysis of pixe spectra in geoscience applications. Nucl Inst Methods Phys Res Sec B: Beam Interactions with Mater Atoms. 1988;34(3):396–402.
41. Morháč M, Kliman J, Matoušek V, Veselský M, Turzo I. Background elimination methods for multidimensional coincidence γ-ray spectra.
Nucl Inst Methods Phys Res Sec A: Accelerators, Spectrometers, Detectors and Assoc Equip. 1997;401(1):113–32.
42. Gibb S, Strimmer K. Maldiquant: a versatile R package for the analysis of mass spectrometry data. Bioinformatics. 2012;28(17):2270–1.
43. Conte SD, Boor CWD. Elementary numerical analysis: an algorithmic approach: McGraw-Hill High Educ. 1980.
44. Varga RS. Geršgorin and his circles: Springer Berlin Heidelberg; 2004. https://doi.org/10.1007/978-3-642-17798-9.
45. Parlett BN. The symmetric eigenvalue problem. SIAM. 1980;7:134.