Sparse Approximation of Ion-Mobility Spectrometry Profiles by Minutely Shifted Discrete B-splines

Masaru Kamada
Department of Computer and Information Sciences
Ibaraki University
Hitachi, Ibaraki 316–8511, Japan
m.kamada@cis.ibaraki.ac.jp

Masakazu Ohno
Graduate School of Science and Engineering
Ibaraki University
Hitachi, Ibaraki 316–8511, Japan

Abstract—Employing discrete B-splines instead of the Gaussian distribution, we construct an algorithm for the analysis of ion-mobility spectrometry profiles. The algorithm is suitable for hardware implementation because the discrete B-splines are supported by a simple digital filter to compute their weighted sum and their correlations with a given signal. Minutely shifted discrete B-splines are deployed of which weighted sum is to approximate a given profile with non-negative weights. Closely neighboring discrete B-splines are almost linearly dependent so that they may cause numerical instability in the approximation process. But numerical experiments deny this anxiety at least for the final results. Varying the width of discrete B-splines, we obtain a number of different approximations. Out of sufficiently precise approximations, we choose the sparse one in the sense that it comprises few discrete B-splines with large weights.

I. Introduction

Ion-mobility spectrometry [1] is a method of discriminating chemical molecules in the atmosphere. Its capability of identifying tiny amounts of various chemicals has made it possible to analyze odor and flavor and to detect poisons, drugs and explosives. The analysis is mainly composed of physical and computational processes.

The physical process proceeds in this way: (i) Chemical molecules are ionized and injected near the cathode as shown in Fig. 1(a). (ii) The ions move toward the anode with the acceleration proportional to their charge-mass-ratio as illustrated in Fig. 1(b). Light ions reach the anode earlier than the heavier ones on the average. The ions bump and bounce against air and other molecules during their travel so that even ions of the same kind arrive at the anode in different traveling times. (iii) The ions give their charges to the anode which constitute the electric current called *profile* like the curve in Fig. 2(a). The profile is modeled as a weighted sum of several distributions as schematized in Fig. 2(b). Each distribution is traditionally supposed to be Gaussian because any random displacements of ions by their collision with other molecules amount to a Gaussian distribution if they happen infinitely many times.

The computational process identifies each different distribution in a given profile. Its weight and average tell, respectively, how much and what kind of ions are present. The standard algorithm employs the steepest descent method to search for locally optimal values of unknown parameters such as average, variance and weight of an unknown number of Gaussian distributions. This search has to be conducted sequentially so that it consumes much time even on the latest fast CPUs.

While a tiny chip from Owlstone Nanotech [2] and a system solution from ATONARP [3] have already made it possible to complete the physical process in a few milliseconds, the computational algorithm is still sequentially searching for local optima at much computational cost. In this paper, we shall approach a new algorithm which matches up to the compact and fast physical system. This approach is characterized by the following four features:

(i) Instead of the Gaussian distribution, we use the B-spline [4] of order m that is defined as the m-fold convolution integral of a uniform distribution and represents the distribution of ion position after m collisions if one causes a uniformly random displacement. The B-spline is a good substitute since it tends to the Gaussian at the limit $m \to \infty$. We can even say that the Gaussian was not the perfect choice because it has infinitely long tails that never exist in reality. We had better take a large m but do not have to make it infinity.

(ii) For the sake of simpler computation, the B-splines are further replaced by their discrete version[1] defined as the m-fold discrete convolution of the uniform discrete distribution over n sampling points. The discrete B-splines can be generated by only additions and subtractions [6]. There is also
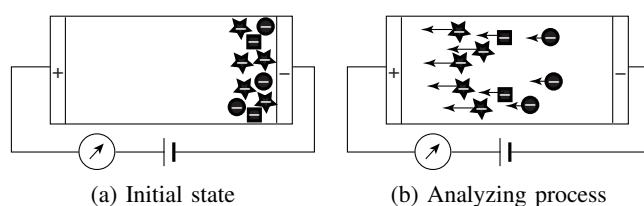

(a) Initial state   (b) Analyzing process

Fig. 1. Schematics of the ion-mobility spectrometry
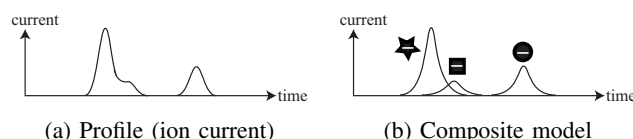

(a) Profile (ion current)   (b) Composite model

Fig. 2. Ion-mobility spectrometry profile

---

[1]The discrete B-splines tend to the original B-splines when $n \to \infty$ [5].

a fast digital filter to compute their correlations with a given signal [7].

(iii) We dare to deploy the discrete B-splines shifted by a minute interval as analyzing components of which weighted sum is to approximate a given profile even though we risk numerical instability in the approximation process due to the almost linear dependency among the overcrowding components. Otherwise, the algorithm would fall back to the slow sequential search for an unknown number of arrival times. The weights are constrained to be non-negative since ion counts cannot be negative numbers.

(iv) The other unknown parameter n, which represents how widely the arrival times distribute, is sought exhaustively. Since the above approximation process is rather simple and suitable for hardware implementation, we can try approximations with various values of n in parallel to find its best value. Among the values of n that result in good approximations with sufficiently small errors, we shall choose the one giving a sparse approximation in the sense that the approximation comprises few discrete B-splines with large weights.

The algorithm is a sort of sparse approximation method that arose from this particular application field. It works empirically fine. A proper formulation within the general theory of sparse approximation is yet to be established.

II. SUMMARY OF DISCRETE B-SPLINES

The B-spline of order m is defined as the m-fold convolution integral of a rectangle function [4]. It tends to the Gaussian distribution at the limit $m \to \infty$ by the central limit theorem. The discrete B-spline to be used in this paper is defined recursively by

$$b_m[k] = (b_{m-1} * b_1)[k] = \sum_{l=-\infty}^{\infty} b_{m-1}[k-l]b_1[l] \quad (1)$$

as the m-fold discrete convolution of a sampled rectangle

$$b_1[k] = \begin{cases} 1, & k = 0, 1, 2, \cdots, n-1 \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

It tends to the original B-spline at the limit $n \to \infty$ [5]. The z-transform of $b_m[\cdot]$ is

$$B_m(z) = \sum_{k=-\infty}^{\infty} b_m[k]z^{-k} = \left(\frac{1-z^{-n}}{1-z^{-1}}\right)^m. \quad (3)$$

The inner product or correlation of two discrete B-splines $b_m[\cdot - r]$ and $b_m[\cdot - l]$ can be rearranged in the form of discrete convolution

$$\begin{aligned} \langle b_m[\cdot - r], b_m[\cdot - l] \rangle &= \sum_{k=-\infty}^{\infty} b_m[k-r]b_m[k-l] \\ &= \sum_{k=-\infty}^{\infty} b_m[k]b_m[-(l-r-k)] \\ &= \sum_{k=-\infty}^{\infty} b_m[k]\tilde{b}_m[l-r-k] \\ &= (b_m * \tilde{b}_m)[l-r], \end{aligned}$$

where we have set $\tilde{b}_m[\cdot] = b_m[-\cdot]$. By its z-transform

$$\begin{aligned} & B_m(z)B_m(z^{-1})z^{l-r} \\ &= \left(\frac{1-z^{-n}}{1-z^{-1}}\right)^m \left(\frac{1-z^n}{1-z}\right)^m z^{l-r} \\ &= \left(\frac{1-z^{-n}}{1-z^{-1}}\right)^m \left(\frac{1-z^{-n}}{1-z^{-1}}\right)^m z^{(n-1)m+l-r} \\ &= \left(\frac{1-z^{-n}}{1-z^{-1}}\right)^{2m} z^{m(n-1)+l-r}, \end{aligned}$$

we know that

$$\langle b_m[\cdot - r], b_m[\cdot - l] \rangle = b_{2m}[l - r + m(n-1)]. \quad (4)$$

Given weighting coefficients $c[\cdot]$ of which z-transform is

$$C(z) = \sum_{l=-\infty}^{\infty} c[l]z^{-l},$$

we can express the weighted sum of discrete B-splines $b_m[\cdot - l]$ in the form of a discrete convolution

$$q[k] = \sum_{l=-\infty}^{\infty} c[l]b_m[k-l] = (c * b_m)[k] \quad (5)$$

of which the representation by the transfer functions is

$$C(z)B_m(z) = C(z)\left(\frac{1-z^{-n}}{1-z^{-1}}\right)^m. \quad (6)$$

So we can generate $q[\cdot]$ as the output of a digital filter having the transfer function $\left(\frac{1-z^{-n}}{1-z^{-1}}\right)^m$ for the input $c[\cdot]$. This digital filter can be implemented in two steps: the m-th order accumulation $\left(\frac{1}{1-z^{-1}}\right)^m$ and the m-th order difference $(1-z^{-n})^m$. Although the accumulation may overflow in the first step, it has been known that the final output $q[\cdot]$ stays correct as long as we use the integer arithmetic in the 2's complement representation of which bit-length is long enough to accommodate the theoretical range of $q[\cdot]$ [6]. Since the amplitude of the final output $q[\cdot]$ is bounded by

$$\begin{aligned} \sup_k |q[k]| &\leq \sup_l |c[l]| \sum_{k=-\infty}^{\infty} |b_m[k]| \\ &= \sup_l |c[l]| \sum_{k=-\infty}^{\infty} b_m[k] \\ &= \sup_l |c[l]| B(1) \\ &= \sup_l |c[l]| n^m, \quad (7) \end{aligned}$$

it cannot be magnified more than n^m times the amplitude of the input $c[\cdot]$. So it suffices for correct computation to add guard bits of the length $m\lceil \log_2 n \rceil$.

For a given profile $p[\cdot]$, let

$$P(z) = \sum_{k=-\infty}^{\infty} p[k]z^{-k}.$$

Then its inner products or correlations with $b_m[\cdot - l]$ can be represented in the form of discrete convolution

$$
\begin{aligned}
\langle p[\cdot], b_m[\cdot - l] \rangle &= \sum_{k=-\infty}^{\infty} p[k] b_m[k - l] \qquad (8)\\
&= \sum_{k=-\infty}^{\infty} p[k] b_m[-(l - k)]\\
&= (p * \tilde{b}_m)[l] \qquad (9)
\end{aligned}
$$

of which the representation by the transfer functions is

$$
\begin{aligned}
P(z) B_m(z^{-1}) z^l &= P(z) \left(\frac{1 - z^n}{1 - z} \right)^m z^l\\
&= P(z) \left(\frac{1 - z^{-n}}{1 - z^{-1}} \right)^m z^{(n-1)m+l}. \qquad (10)
\end{aligned}
$$

So we can compute the inner products by inputting $p[\cdot]$ to a digital filter having the transfer function $\left(\frac{1 - z^{-n}}{1 - z^{-1}} \right)^m$ and sampling its output at $(n - 1)m + l$. This transfer function is the same as the one for generating weighted sums and can also be implemented efficiently in the two steps.

III. ALGORITHM

We have to approximate a given profile $p[\cdot]$ by a weighted sum of the discrete B-splines $b_m[\cdot - l]$ deployed by the most minute interval 1 under the constraint that the weights should be non-negative.

A. Digital filter to compute inner products $\langle p[\cdot], b_m[\cdot - l] \rangle$

The inner products $\langle p[\cdot], b_m[\cdot - l] \rangle$ of a given profile $p[\cdot]$ and the discrete B-splines $b_m[\cdot - l]$ should usually be evaluated by the standard multiply-and-add architecture according to their definition (8). But, by virtue of (9) and (10), we can do the same only by using the digital filter depicted in Fig. 3.

The first half of the filter in Fig. 3 represents the m-fold accumulation free from the parameter n so that this part has to be operated just once for a given profile. We can evaluate the inner product for different n only by operating the second half. Its computational cost is almost only m subtractions per an inner product on the average.

The mutual inner products $\langle b_m[\cdot - r], b_m[\cdot - l] \rangle$ among the discrete B-splines can be precomputed by (4) and stored in a data table.

B. Non-negative least-square approximation

From the inner products, we are to determine the weighting coefficients $c[\cdot]$ so that the weighted sum

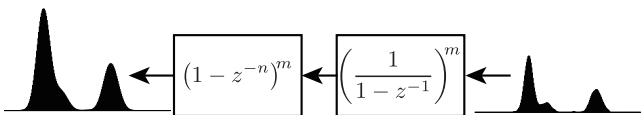$$
q[k] = \sum_{l=0}^{L-1} c[l] b_m[k - l] \qquad (11)
$$

of the discrete B-splines $b_m[\cdot - l]$ approximate the profile $p[\cdot]$ best in the sense

$$
\begin{aligned}
E &= \langle p[\cdot] - q[\cdot], p[\cdot] - q[\cdot] \rangle\\
&= \sum_k (p[k] - q[k])^2 \longrightarrow \min. \qquad (12)
\end{aligned}
$$

Such coefficients can be determined by solving the normal linear equations

$$
\sum_{l=0}^{L-1} c[l] \langle b_m[\cdot - r], b_m[\cdot - l] \rangle = \langle p[\cdot], b_m[\cdot - l] \rangle,
$$
$$
r = 0, 1, 2, ..., L - 1. \qquad (13)
$$

The resulting coefficients $c[\cdot]$ may be negative whereas they should be constrained to be non-negative.

The least-square approximation under this constraint can be solved by overwriting the negative coefficients by zero, discarding the coefficients and corresponding discrete B-splines from the linear equations, and solving the linear equations repeatedly until all the coefficients get non-negative [8].

Although all the four arithmetic operations in the floating point representation are required to solve the linear equations, this process is not so slow since the discrete B-splines are locally supported to make the equations banded and because the number of involved discrete B-splines decreases during the iterations.

The only and major concern is numerical instability in solving the linear equations. The minutely shifted discrete B-splines are so crowded that they are almost linearly dependent. The numerically obtained initial approximation result is quite imprecise despite the mathematical fact that the initial approximation must theoretically be an exact interpolation having no errors at all. It has been empirically observed that neighboring discrete B-splines are likely to have coefficients of opposite signs. Since the discrete B-splines with negative coefficients should be discarded, the discrete B-splines get sparser in the next iteration. The non-negativity constraint happened to bring in such a nice side effect. In that way, all the numerical experiments up to now with test data taken from real profiles finished successfully at the end.

C. Evaluation of mean square error

It follows from (5) that the approximate profile $q[\cdot]$ can be computed from $c[\cdot]$ by the digital filter depicted in Fig. 4. The mean square error is evaluated from the output $q[\cdot]$ and the original profile $p[\cdot]$ by

$$
E_1 = \sqrt{ \frac{\sum_k (p[k] - q[k])^2}{\sum_k (p[k])^2} }. \qquad (14)
$$

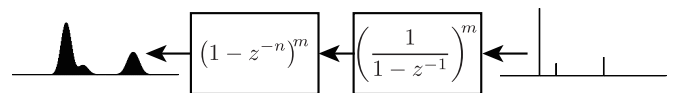Fig. 3. Digital filter to compute $\langle p[\cdot], b_m[\cdot - l] \rangle$ at $(n-1)m + l$ from $p[k]$

Fig. 4. Digital filter to compute $q[k]$ from $c[k]$

D. Evaluation of sparsity

Among a number of approximations obtained by the above processes *A*, *B* and *C* for various values of n, we choose the best one that has sufficiently small errors and is composed of few discrete B-splines with large weights.

In the case that the mean square error E_2 is very small, the absolute difference $\sum_k |p[k] - q[k]|$ is also small and the sums $\sum_k p[k]$ and $\sum_k q[k]$ of the original and approximate profiles are close to each other because both $p[\cdot]$ and $q[\cdot]$ are non-negative. In this case, the modified coefficients $n^m c[l]$ for the discrete B-splines $\frac{1}{n^m} b_m[\cdot - l]$ normalized by its sum $\sum_k b_m[k - l] = n^m$ satisfy

$$q[k] = \sum_{l=0}^{L-1} (n^m \ c[l]) \left(\frac{1}{n^m} \ b_m[k - l] \right) \qquad (15)$$

and

$$\sum_{l=0}^{L-1} n^m c[l] = \sum_k q[k] \approx \sum_k p[k] = \text{constant.} \qquad (16)$$

In this situation, sparsity in the sense that the approximation $q[\cdot]$ should comprise few large portions is translated into that the coefficients $n^m c[l]$ should comprise few and large ones because $\sum_l n^m c[l]$ is constant and $n^m c[l]$ is non-negative. An index to evaluate this sparsity is

$$E_2 = \sqrt{\sum_{l=0}^{K-1} (n^m c[l])^2}. \qquad (17)$$

We take the sparsest approximation giving the largest E_2 out of the good approximations having the mean-square error E_1 smaller than a threshold among various approximations for different values of n.

IV. NUMERICAL RESULT

The algorithm was applied to test data taken from real profiles. Figure 5 shows approximations of a profile within a short window of 128 sampling points. The order of the discrete B-splines is fixed as $m = 4$. The original profile $p[\cdot]$ is plotted in black, its approximations $q[\cdot]$ for various n is in red. The green curves represent the discrete B-splines weighted by their coefficients to compose the approximation.

The cases for $n \leq 12$ cleared the precision bar conditioned by $E_1 < 0.1$. The case $n = 12$ gave the largest E_2 to be selected as the sparsest among the precise approximations.

Figure 6 shows a whole profile. The best n gets larger as the time passes so that it was sought within each short window. The best n is 12 for the two left hills and 14 for the right one.

V. CONCLUSIONS

The discrete B-splines were employed to construct an algorithm for the analysis of ion-mobility spectrometry profiles. This application field requested modification of the standard B-spline approximations in two aspects: the deployment of B-splines by a minute shift interval and the non-negativity constraint on coefficients. The former put us in danger of numerical instability and the latter pulled us out of it.
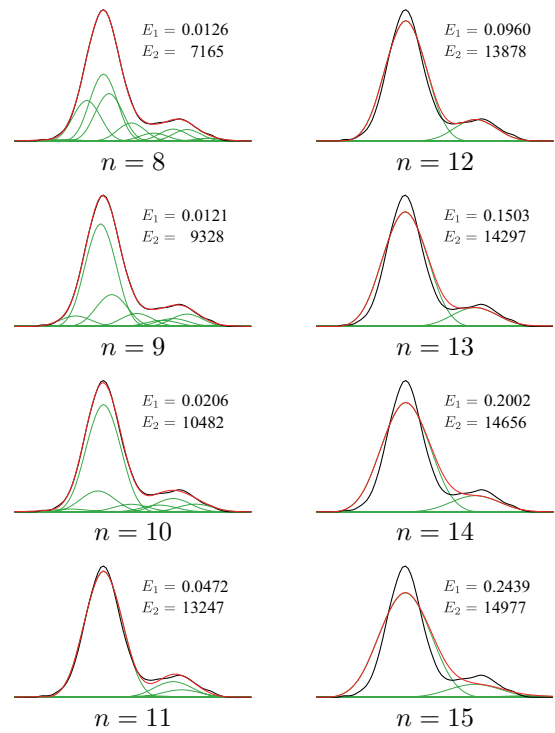

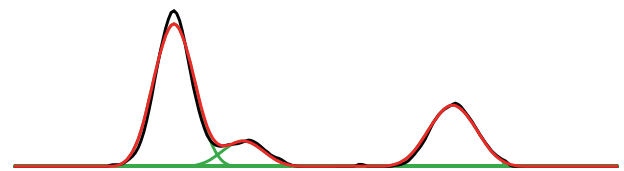Fig. 5. Approximations of a partial sample profile

Fig. 6. Approximation of a whole sample profile

A next step is to come up with a single index to balance the approximation error and the sparsity. We may probably have to reformulate the problem within the general theory of sparse approximation. Before a large scale test of the algorithm against various profile data, we should see by simulations whether an artificial profile built up of discrete B-splines is identified in the noise-free and noisy cases.

REFERENCES

[1] R. W. Purves, R. Guevremont, S. Day, C. W. Pipich and M. S. Matyjaszczyk, Mass spectrometric characterization of a high-field asymmetric waveform ion mobility spectrometer, *Rev. Sci. Instrum.*, **69**, 4094-4105, 1998.
[2] Owlstone Nanotech, Inc. http://www.owlstonenanotech.com/technology/
[3] ATONARP, Inc. http://www.atonarp.com/en/
[4] M. Unser, Splines: A perfect fit for signal and image processing, *IEEE Signal Process. Magazine*, **16**(6), 22–38,1999.
[5] K. Ichige and M. Kamada, An approximation for discrete B-splines in time domain, *IEEE Signal Process. Lett.*, **4**, 82–84, 1997.
[6] T. Saramäki, Y. Neuvo and S. K. Mitra, Design of computationally efficient interpolated FIR filters, *IEEE Trans. Circuits & Syst.*, **35**(1), 70– 88, 1988.
[7] K. Ichige, M. Kamada and R. Ishii, A simple scheme of decomposing and reconstructing continuous-time signals by B-splines, *IEICE Trans. Fundamentals*, **E81-A**, 2391–2399, 1998.
[8] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*, SIAM, 1987.