Reconstruction from non-uniform samples: A direct, variational approach in shift-invariant spaces

Laurent Condat[*]

Final author's version. Cite as: L. Condat, "Reconstruction from non-uniform samples: A direct, variational approach in shift-invariant spaces," Digital Signal Processing, vol. 23, no. 4, pp. 1277–1287, 2013.

Abstract

We propose a new approach for signal reconstruction from non-uniform samples, without any constraint on their locations. We look for a function that minimizes a classical regularized least-squares criterion, but with the additional constraint that the solution lies in a chosen linear shift-invariant space—typically, a spline space. In comparison with a pure variational treatment involving radial basis functions, our approach is resolution dependent; an important feature for many applications. Moreover, the solution can be computed exactly by a fast non-iterative algorithm, that exploits at best the particular structure of the problem.

Keywords: Non-uniform sampling, variational reconstruction, interpolation, shift-invariant spaces, splines

1 Introduction

In modern digital data processing and communication systems, signals and numerical data are usually available as a sequence of discrete—eventually non-uniform and/or noisy—samples. For the purpose of deriving numerical algorithms, it is sometimes desirable to represent the data by a continuously-defined parametric function. This is particularly relevant for resampling tasks: a function is fitted to the data and resampled at new locations. Non-integer sampling rate conversion, arbitrary time delay, or edge detection in images are examples where such a treatment may be required.

Numerous methods have been proposed in the literature for reconstructing a signal from non-uniform samples. Since the pioneering work of Shannon [1], it is known that any π/T-bandlimited function $s(t) \in L_2(\mathbb{R})$ can be perfectly reconstructed from its uniform samples $s(Tn), n \in \mathbb{Z}$. This is still true when

[*] Laurent Condat is with GIPSA-lab, CNRS–University of Grenoble, France. Contact: see `http://www.gipsa-lab.grenoble-inp.fr/∼laurent.condat/`

considering non-uniform samples, assuming strong limitations on the samples locations [2]. Perfect reconstruction from non-uniform samples is also possible under the weaker assumption that $s(t)$ belongs to a *linear shift-invariant* (LSI) space [3], with similar constraints on the sampling set [4]. Practical iterative algorithms have been proposed for reconstruction in the bandlimited case [5, 6] and in LSI spaces [7, 8]. Strong conditions have to be met for these algorithms to converge [9].

In this article, we propose a novel approach for one-dimensional signal reconstruction from non-uniform samples, without any constraint on their locations. We aim at reconstructing a continuous-time function $f_T(t)$ that best fits the data, while being resolution dependent, i.e. completely determined by its values $f_T(Tk), k \in \mathbb{Z}$, where the parameter T can be chosen arbitrarily. In other words, we seek a reconstructed function that is constrained to lie in a linear shift-invariant space. This ensures that the solution is parameterized by coefficients attached to the uniform reconstruction grid having the desired resolution, and not to the data locations. In this reconstruction space, we formulate reconstruction as a regularized least-squares problem; that is, the solution minimizes a cost depending on two terms: the sum of the squared errors at the sampling locations on one side, and a quadratic variational functional that enforces the solution to be smooth on the other side. This formulation circumvents the strong limitations—typically, a restriction on the maximal gap between samples—of formulations where only the fit to the data is considered. The main advantage of our formulation is that the coefficients, that parameterize the reconstructed function, can be computed using a simple, fast, and non-iterative algorithm. In essence, it performs a two-pass time-varying recursive filtering of the data.

The paper is organized as follows. We first give the necessary mathematical background and formulate the problem in Section 2. In Section 3, we derive its solution and discuss the novelty of our approach in comparison with previous related works. We then propose a practical algorithm for computing the solution in Section 4. In Section 5, we discuss the properties of the reconstructed function and illustrate our method with experimental results. In the last section, some applications are discussed.

2 Mathematical preliminaries

2.1 Definitions and notations

Throughout the paper, parentheses are used for continuous-time signals, e.g., $f(t)$, and brackets for the samples of a discrete time signal like $u = (u[n])_{n\in\mathbb{Z}}$. Continuous and discrete convolutions are denoted by $*$. We define the $z-$transform of a discrete signal u by $U(z) = \sum_{n\in\mathbb{Z}} u[n]z^{-n}$, and the convolution inverse u^{-1} of u as the sequence whose $z-$transform is $1/U(z)$. We also introduce the autocorrelation function a_φ of a function φ: $a_\varphi(t) = (\bar{\varphi} * \varphi)(t)$, using the flip operator $\bar{\varphi}(t) = \varphi(-t)$.

2.2 Linear shift-invariant spaces

In this work, we make use of functions that belong to linear shift-invariant (LSI) spaces. Let the real number $T > 0$ denote our choice of sampling step. The LSI space $V_T(\varphi)$ is the functional space spanned by the T-shifts of a generating function $\varphi(\frac{t}{T})$:

$$V_T(\varphi) = \Big\{ \sum_{k \in \mathbb{Z}} c_T[k]\, \varphi(\frac{t}{T} - k) \ : \ c_T[k] \in \mathbb{R} \ \ \forall k \in \mathbb{Z} \Big\}. \tag{1}$$

The potential of LSI spaces has been recognized for quite some time. They have been used in finite elements and approximation theory [10, 11] and for the construction of multiresolution approximations and wavelets [12]. In sampling theory, they are used extensively [13, 8]. We assume in this work that φ is bounded with compact support, and that the functions $\{\varphi(\frac{t}{T} - k)\}$ form a Riesz basis of $V_T(\varphi) \cap L_2(\mathbb{R})$, which ensures that each function $f_T \in V_T(\varphi)$ has a unique expansion of the form $f_T(t) = \sum_{k \in \mathbb{Z}} c_T[k]\, \varphi(\frac{t}{T} - k)$. This last condition is equivalent to the requirement that there exist two constants $0 < C_1$ and $C_2 < +\infty$ (the lower and upper Riesz bounds) such that
$C_1 < \sum_{k \in \mathbb{Z}} a_\varphi(k) e^{-j\omega k} < C_2$ a.e.

A classical example of LSI space is the set of π/T-bandlimited functions, obtained with $\varphi(t) = \mathrm{sinc}(t)$ [1]. Particularly useful LSI spaces are the polynomial spline spaces [14, 15]. They are obtained by choosing $\varphi = \beta^d$, the centered B-spline of degree d, that is symmetric and has compact support of width $d+1$. The B-splines are constructed by successive convolution: $\beta^d = \beta^0 * \beta^{d-1}$, from the indicator function β^0 defined by $\beta^0(t) = \big\{1 \text{ if } t \in [-\frac{1}{2}, \frac{1}{2}), 0 \text{ otherwise}\big\}$.

3 Variational reconstruction

3.1 Problem statement

We assume that we are given a finite number N of measurements $(v[n])_{n \in [0,N-1]}$ at locations $(x[n])_{n \in [0,N-1]}$ within an interval $\mathcal{I} = [0, S]$; that is, $v[n] = s(x[n])$ where $s(t)$ is some unknown process defined on $\mathcal{I}$. Let us choose a resolution step $T > 0$ (whose choice will be discussed in Sections 5, 6). Our goal is to reconstruct a continuous-time function $f_T(t)$ defined on $\mathcal{I}$, that modelizes the data and belongs to the reconstruction space $V_T(\varphi)$. That is, we look for a function having the form

$$f_T(t) = \sum_{k \in \mathbb{Z}} c_T[k] \varphi(\frac{t}{T} - k), \tag{2}$$

where the discrete coefficients $c_T[k]$ are the unknowns to be determined. We assume that the generator φ has compact support, included in the interval $(-W, W)$, with $W \in \mathbb{N}$, and that $S = KT$, for some $K \in \mathbb{N}$. Then, for every $t_0 \in \mathcal{I}$ the value $f_T(t_0)$ is determined by only a few coefficients $c_T[k]$, for k

in the interval $(\frac{t_0}{T} - W, \frac{t_0}{T} + W)$. Therefore, the function f_T is completely determined on the interval $\mathcal{I}$ by the $c_T[k], k \in [-W+1, K+W-1]$. So, for convenience, we only consider functions of $V_T(\varphi)$ such that $c_T[k] = 0$ for every $k \notin [-W+1, K+W-1]$ in (1) (except in Section 3.3). This allows to use simple notations indexed by $k \in \mathbb{Z}$, but we have to keep in mind that only a finite number of coefficients are not zero, e.g. in (2).

We define our reconstruction problem as a variational problem, whose f_T is the solution:

$$f_T = \underset{g \in V_T(\varphi)}{\operatorname{argmin}} \left(\sum_{n=0}^{N-1} \left| g(x[n]) - v[n] \right|^2 + \lambda \int_{\mathcal{I}} |g^{(r)}(t)|^2 \, dt \right). \tag{3}$$

where $g^{(r)}$ is the r^{th} derivative of g, for some integer $r \geq 1$.

This variational criterion is composed of two antagonist terms, one controlling the closeness to the data, the other one enforcing the solution to be smooth. Interestingly, these terms are similar to the so-called external forces (the function is attracted by the data) and internal forces (the bending energy of the function tends to be minimized) in the theory of *snakes* and *active contours* used in computer graphics [16]. The quantity $\lambda > 0$ is a Lagrangian parameter working as a tradeoff factor between these two terms. The integer r controls the smoothness of the reconstruction. The values $r = 1$ and $r = 2$ are the most frequently used, and correspond to searching a function that has maximum flatness, and minimum curvature, respectively. The resolution parameter $T > 0$ controls the coarseness of the representation. Finally, the generator function $\varphi(t)$ controls the space where the reconstructed function lives. Signal processing practitioners often rely on localized kernels such as B-splines [15] or MOMS [17]. Other compactly-supported kernels were recently introduced in [18]. The influence of this set of parameters is discussed and illustrated in Section 5.

Before determining the solution of (3) in the remainder of Sect. 3, we have to introduce some mathematical safeguards for the problem to be well posed. First, we suppose that φ is such that $\int_{\mathbb{R}} |\varphi^{(r)}(t)|^2 dt < \infty$. For instance, if choosing a B-spline $\varphi = \beta^d$, then $d \geq r$ is necessary. Moreover, we require that the samples are at r or more distinct locations. We will see that these conditions allow the solution of our problem to exist and to be unique.

It is important to note that the Tikhonov-like formulation (3) is not limited to the choice of the squared norm of the derivative for the regularization term. The approach could be extended without difficulty to any other quadratic shift-invariant penalty with impulse response of compact support. We exemplify the approach with the energy of the r-th derivative because it is a classical penalty for functions living in Sobolev spaces, but another regularizer may be more appropriate, for a given application. The reader may wonder if a non-quadratic criterion like the *total variation* [19] could be considered; that is, replacing the integral in (3) by $\int_{\mathcal{I}} |g^{(r)}(t)| dt$ (without the square). In that case, an iterative algorithm for convex non-smooth optimization would be necessary to solve (3), whereas the efficiency of the proposed non-iterative algorithm is due to the linear structure of the problem. Therefore, the extension of the approach to non-quadratic regularization is beyond the scope of the paper and is left for future work.

3.2 Solution to (3)

Finding the reconstructed function $f_T(t)$ amounts to determining the sequence c_T in (2), so that the cost function

$$\Psi(c_T) = \sum_{n=0}^{N-1} \left| f_T(x[n]) - v[n] \right|^2 + \lambda \int_{\mathcal{I}} |f_T^{(r)}(t)|^2 \, dt \tag{4}$$

is minimized. First, we rewrite the data fidelity term as a function of c_T:

$$\sum_{n=0}^{N-1} \left| f_T(x[n]) - v[n] \right|^2 = \sum_{n=0}^{N-1} \left(v[n] - \sum_{k\in\mathbb{Z}} c_T[k] \varphi(\frac{x[n]}{T} - k) \right)^2 . \tag{5}$$

Second, we rewrite the variational term as a function of c_T, with the integral over $\mathbb{R}$ and not $\mathcal{I}$ in a first time. Let us introduce the discrete sequence $q_{\varphi,r}$ defined by $q_{\varphi,r}[k] = \frac{(-1)^r}{T} a_\varphi^{(2r)}(k)$. Since $\varphi^{(r)}(-t) = (-1)^r \bar{\varphi}^{(r)}(t)$, and differentiations commute with convolutions, we have:

$$\begin{aligned}
\int_{\mathbb{R}} \left| f_T^{(r)}(t) \right|^2 dt &= \frac{1}{T^2} \int_{\mathbb{R}} \left(\sum_{k\in\mathbb{Z}} c_T[k] \varphi^{(r)}(\frac{t}{T} - k) \right)^2 dt \\
&= \frac{1}{T} \sum_{k,l\in\mathbb{Z}} c_T[k] c_T[l] \int_{\mathbb{R}} \varphi^{(r)}(x - (k-l)) \varphi^{(r)}(x) \, dx \\
&= \frac{(-1)^r}{T} \sum_{k,l\in\mathbb{Z}} c_T[k] c_T[l] \left(\bar{\varphi}^{(r)} * \varphi^{(r)} \right) (k-l) \\
&= \frac{(-1)^r}{T} \sum_{k,l\in\mathbb{Z}} c_T[k] c_T[l] \, a_\varphi^{(2r)}(k-l) \\
&= \sum_{k\in\mathbb{Z}} c_T[k] \, (c_T * q_{\varphi,r})[k]. \qquad (6)
\end{aligned}$$

In the case of spline reconstruction ($\varphi = \beta^d$), we can give the general form of the filter $q_{\varphi,r}$. First, B-splines verify the simple relation $a_{\beta^d}(t) = \beta^{2d+1}(t)$. Moreover, the derivative of a spline is also a spline of lower degree, since ${\beta^d}^{(1)}(t) = \beta^{d-1}(t + \frac{1}{2}) - \beta^{d-1}(t - \frac{1}{2})$. We define b^d as the discrete centered B-spline of degree d: $b^d[k] = \beta^d(k)$ for every $k \in \mathbb{Z}$ and $d \in \mathbb{N}$. We recall the expression of the first few discrete B-splines in the $z-$domain: $B^0(z) = 1$, $B^1(z) = 1, B^2(z) = \frac{1}{8}z + \frac{3}{4} + \frac{1}{8}z^{-1}, B^3(z) = \frac{1}{6}z + \frac{2}{3} + \frac{1}{6}z^{-1}$. Then the filter $q_{\varphi,r}$ has the following form:

$$Q_{\beta^d,r}(z) = \frac{1}{T}(-z + 2 - z^{-1})^r \, B^{2d+1-2r}(z). \tag{7}$$

For example, if $\varphi = \beta^1$ and $r = 1$, then $Q_{\varphi,r}(z) = (-z + 2 - z^{-1})/T$. If $\varphi = \beta^3$ and $r = 2$, then $Q_{\varphi,r}(z) = (z^3 - 9z + 16 - 9z^{-1} + z^{-3})/6T$.

In order to express the cost $\Psi(c_T)$ in terms of matrices and vectors, we introduce the following quantities ($\cdot^{\mathrm{T}}$ is the transpose operator):

$$\begin{aligned}
&\mathbf{c} = \big[c_T[-W+1] \ \cdots \ c_T[K+W-1]\big]^{\mathrm{T}}, \\
&\mathbf{v} = \big[v[0] \ v[1] \ \cdots \ v[N-1]\big]^{\mathrm{T}}, \\
&\mathbf{M} = [(M[n,k])_{n,k}] \text{ with } M[n,k] = \varphi(\frac{x[n]}{T} - k), \qquad (8) \\
&\ \text{for } n \in [0, N-1], k \in [-W+1, K+W-1], \\
&\mathbf{Q} = [(Q[k,l])_{k,l}] \text{ for } k, l \in [-W+1, K+W-1]. \qquad (9)
\end{aligned}$$

Note that performing a matrix-vector product like $\mathbf{Mc}$ is equivalent to applying a time-varying filter to the signal c_T. Using these matrices, the cost $\Psi(c_T)$ can be rewritten as:

$$\Psi(\mathbf{c}) = \|\mathbf{Mc} - \mathbf{v}\|^2 + \lambda \mathbf{c}^{\mathrm{T}} \mathbf{Qc}, \qquad (10)$$

where the values of the matrix $\mathbf{Q}$ are given by (6):

$$Q[k,l] = q_{\varphi,r}[k-l] = \frac{(-1)^r}{T} \, a_{\varphi}^{(2r)}(k-l), \qquad (11)$$

except for the first and last rows of $\mathbf{Q}$ that contain particular values because $\Psi(c_T)$ is defined with the integral over $\mathcal{I}$, and not $\mathbb{R}$ as in (6). These special values are in squares of size $(2W-1)^2$ in the lower-left and upper-right corners of the matrix. To compute them for the left boundary (this would be the same for the other boundary), we have to develop the left-hand side of the following equality, and identify the coefficients with its right-hand side:

$$\int_0^{2W-1} \Big| \sum_{k=-W+1}^{W-1} c_T[k] \varphi^{(r)}(\frac{t}{T} - k) \Big|^2 dt = \sum_{k,l=-W+1}^{W-1} Q[k,l] c_T[k] c_T[l]. \qquad (12)$$

For instance, if $\varphi = \beta^1, r = 1$ and $\varphi = \beta^3, r = 2$, $\mathbf{Q}$ takes the respective forms:

$$\begin{array}{cc}
\mathbf{Q} = \dfrac{1}{T} \times & \mathbf{Q} = \dfrac{1}{6T} \times \\
\begin{bmatrix}
\mathbf{1} & -1 & 0 & 0 & 0 & \cdots \\
-1 & 2 & -1 & 0 & 0 & \ddots \\
0 & -1 & 2 & -1 & 0 & \ddots \\
0 & 0 & -1 & 2 & -1 & \ddots \\
0 & 0 & 0 & -1 & 2 & \ddots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots
\end{bmatrix}, &
\begin{bmatrix}
\mathbf{2} & \mathbf{-3} & \mathbf{0} & 1 & 0 & \cdots \\
\mathbf{-3} & \mathbf{8} & \mathbf{-6} & 0 & 1 & \ddots \\
\mathbf{0} & \mathbf{-6} & \mathbf{14} & -9 & 0 & \ddots \\
1 & 0 & -9 & 16 & -9 & \ddots \\
0 & 1 & 0 & -9 & 16 & \ddots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots
\end{bmatrix}.
\end{array}$$

Now, let us define $\mathbf{A} \overset{\text{def}}{=} \mathbf{M}^{\mathrm{T}}\mathbf{M} + \lambda\mathbf{Q}$ and $\mathbf{y} \overset{\text{def}}{=} \mathbf{M}^{\mathrm{T}}\mathbf{v}$. Minimizing the cost $\Psi(\mathbf{c})$ amounts to solving the linear system

$$\mathbf{A}\mathbf{c} = \mathbf{y}. \tag{13}$$

When considering the rows of this system, we obtain a set of equations that can be found directly from (5) and (6) by setting the partial derivatives $\partial\Psi/\partial c_T[k]$ to zero, for each $k \in [-W+1, K+W-1]$:

$$\begin{aligned}\sum_{l\in\mathbb{Z}} \left[\sum_{n=0}^{N-1} \varphi(\frac{x[n]}{T} - k)\varphi(\frac{x[n]}{T} - l) + \lambda\, Q[k,l]\right] c_T[l] \\ = \sum_{n=0}^{N-1} \varphi(\frac{x[n]}{T} - k)v[n].\end{aligned} \tag{14}$$

We have to check that the solution to the linear system (13) is well defined. In fact, $\mathbf{M}^{\mathrm{T}}\mathbf{M}$ and $\mathbf{Q}$ are positive semi-definite matrices, and $\mathbf{A}$ is symmetric and positive definite, as proved in Section 8. As a consequence, the linear system has a unique solution. Moreover, thanks to the hypothesis of compact support of φ, $\mathbf{A}$ is band-diagonal with only $4W-1$ diagonals containing non-zero entries; more precisely, $A[k,l] = 0$ if $|k-l| \geq 2W$. This will be the key for efficiently solving this linear system, as detailed in Section 4.

3.3 Reconstruction with mirror boundary conditions

There is a second possibility to deal with reconstruction on a finite interval: considering mirror boundary conditions that allow to extend a finite signal to an infinite one by symmetry and periodicity [20]. The main advantage is that there is no overhead of coefficients as with the previous method; that is, we only have to compute $c_T[k]$ for $k \in [0, K]$: the decomposition is said to be *non-expansive*. In this subsection, we assume that φ is symmetric ($\varphi(t) = \varphi(-t)$).

The proposed method consists in solving the following problem:

$$f_T = \underset{g\in V_T(\varphi)\cap\mathcal{S}}{\operatorname{argmin}} \left(\sum_{n=0}^{N-1} \left|g(x[n]) - v[n]\right|^2 + \lambda \int_{\mathcal{I}} |g^{(r)}(t)|^2\, dt\right). \tag{15}$$

where $\mathcal{S}$ is the set of functions that are symmetric around $t = 0$ and $t = TK$. With these boundary conditions, the coefficients c_T of f_T verify the following symmetry and periodicity conditions: $c_T[-k] = c_T[k]$ and $c_T[k+2pK] = c_T[k]$ for every $k, p \in \mathbb{Z}$. In other words, f_T is completely determined by the coefficients $c_T[k], k \in [0, K]$. An interpretation of (15) is to consider that we want to fit a function on an infinite data set that is symmetric and periodic, as illustrated in Fig. 1.

The coefficients $(c_T[k])_{k\in[0,K]}$ are solution of a linear system $\widetilde{\mathbf{A}}\widetilde{\mathbf{c}} = \widetilde{\mathbf{y}}$ to be determined (the tildes are for distinguishing the matrices from the ones previously defined). Let us study the structure of this system on the left boundary

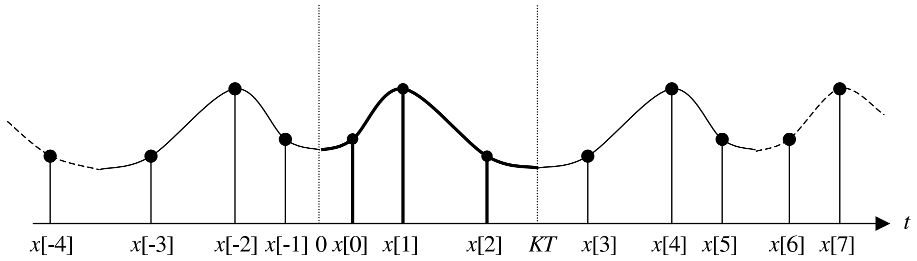


Figure 1: A finite data set of $N = 3$ samples extended to an infinite one using mirror boundary conditions

only (the treatment on the right boundary is similar). As in the previous subsection, our problem is equivalent to minimizing the criterion $\Psi(\mathbf{c})$ in (10). The vector $\mathbf{c}$ has the form $\mathbf{c} = \big[c_T[W-1] \cdots c_T[1]\ c_T[0]\ c_T[1] \cdots\big]^{\mathrm{T}}$ because of the boundary conditions $c_T[-k] = c_T[k]$. Each row of $\mathbf{Mc}$ can be expanded as $\cdots + M[k,-1]c_T[-1] + M[k,0]c_T[0] + M[k,1]c_T[1] + \cdots$, that can be rewritten $M[k,0]c_T[0] + (M[k,-1] + M[k,1])c_T[1] + \cdots$. Therefore, $\widetilde{\mathbf{M}}\widetilde{\mathbf{c}} = \mathbf{Mc}$, where $\widetilde{\mathbf{M}}$ is derived from $\mathbf{M}$ by removing its first $W-1$ columns and adding their values to the ones of the mirror columns with respect to the W^{th} column (with the similar manipulation at the right boundary). This is equivalent to "folding" the matrix $\mathbf{M}$, as in the following example with $W = 2$, where the first column is added to the third one and removed:

$$\mathbf{M} = \begin{bmatrix} a & b & c & d & \cdots \\ e & f & g & h & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad \longrightarrow \quad \widetilde{\mathbf{M}} = \begin{bmatrix} b & c+a & d & \ldots \\ f & g+e & h & \ldots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{16}$$

This folding operation has to be performed on the first and last rows and columns of the matrix $\mathbf{Q}$ and $\mathbf{A}$ in order to get $\widetilde{\mathbf{Q}}$ and $\widetilde{\mathbf{A}}$, respectively. Similarly, the first and last values of $\mathbf{y}$ have to be folded. For instance, if $\varphi = \beta^3$ and $r = 2$ (thus $W = 2$), this yields:

$$\begin{matrix} \mathbf{Q} = \dfrac{1}{6T} \times \\ \left[\begin{array}{r|rrrrr} \mathbf{2} & \mathbf{-3} & \mathbf{0} & 1 & 0 & \cdots \\ \hline \mathbf{-3} & \mathbf{8} & \mathbf{-6} & 0 & 1 & \ddots \\ \mathbf{0} & \mathbf{-6} & \mathbf{14} & -9 & 0 & \ddots \\ 1 & 0 & -9 & 16 & -9 & \ddots \\ 0 & 1 & 0 & -9 & 16 & \ddots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \end{array}\right] \end{matrix} \quad \longrightarrow \quad \begin{matrix} \widetilde{\mathbf{Q}} = \dfrac{1}{6T} \times \\ \begin{bmatrix} \mathbf{8} & -9 & 0 & 1 & \ddots \\ -9 & \mathbf{16} & \mathbf{-8} & 0 & \ddots \\ 0 & \mathbf{-8} & 16 & -9 & \ddots \\ 1 & 0 & -9 & 16 & \ddots \\ \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}. \end{matrix} \tag{17}$$

3.4 Related works

Most of the works in the literature address the non-uniform reconstruction problem with restrictions on the samples locations, usually a density constraint or a maximum gap $x[n+1]-x[n]$, so as to achieve perfect reconstruction of an unknown signal $s(t)$ from its non-uniform noiseless samples [2, 4, 21]. If noise is present, some of these methods yield a reconstructed function that approaches the data at best in the least-squares sense. The variational formulation adopted in the present paper allows to handle the problem in whole generality, whatever the distribution of samples. It is classical, in approximation theory and statistics, to reconstruct a function $f(t)$ that minimizes the criterion of (3), but over the whole Sobolev space of order r instead of over $V_T(\varphi)$ [22]. The solution to this problem can be expressed as $f(t)=\sum_{n=0}^{N-1} c[n]\,|t-x[n]|^{2r-1}+p(t)$ [14]. It is made of a polynomial $p(t)$ of degree less than r and a linear combination of *radial basis functions* (RBF) $|t|^{2r-1}$ positioned at the sampling locations $x[n]$. This implies that the solution $f(t)$ is a non-uniform polynomial spline of degree $2r-1$ with knots at the $x[n]$. These spline spaces have other basis functions, the non-uniform B-splines, which are compactly supported and, therefore, much better conditioned than the radial basis functions [14]. Non-uniform B-splines revert to classical B-splines when the samples are uniform. In comparison with the RBF framework, our approach enforces the solution to lie in $V_T(\varphi)$, and thus to be resolution-dependent. Our solution can be expressed as a linear combination of translates of a single generator φ, while in the RBF framework, a non-uniform B-spline has to be determined for each sample. Moreover, the choice of φ is free in our approach, while in the RBF framework, it is induced by the differential regularization operator, and thus constrained to be a spline of odd degree.

The idea of minimizing a regularized least-squares criterion in a LSI space has also been proposed recently in 2-D for image reconstruction [23, 24], but the associated algorithms, although relatively fast in comparison with previous works, are still iterative and not adapted to real-time implementations. In fact, our algorithm is limited to the 1-D case. This is the price to pay for having the direct and very efficient implementation based on time-varying recursive filtering, detailed in Section 4.

Another related approach has been proposed in [25]; it also provides a resolution-dependent reconstruction $f_T(t)$, in the noise-free case. To this end, the non-uniform spline $f(t)$ interpolating the data is computed, and projected orthogonally onto $V_T(\varphi)$ in the L_2 sense, so as to obtain $f_T(t)$. This method is more complicated than ours, since the non-uniform interpolation has to be performed first, and the projection step is relatively involved. Moreover, this formulation gives more credit, in the definition of $f_T(t)$, to the estimate $f(t)$ than to the data $(v[n])$. It is more reliable to directly formulate the solution $f_T(t)$ from the available data $(v[n])$, as proposed in this work.

4 A fast reconstruction algorithm

4.1 Strategy

We have seen that our problem boils down to solving a linear system. It is possible to perform this operation using standard linear algebra libraries (such as LAPACK), that implement routines to factorize matrices in various ways and to solve linear systems. Such libraries are highly optimized, but they are generic. The practitioner interested by our approach may want to have a dedicated algorithm, that exploits at best the particular structure of the problem. So, we detail in this section the best strategy for solving the linear system $\mathbf{Ac} = \mathbf{y}$. This system is symmetric, positive definite, and band-diagonal. Thus, the best way to solve it relies on the Cholesky factorization of $\mathbf{A}$ and consists in the following three steps:

1. The Cholesky decomposition of $\mathbf{A}$ is performed; that is, we compute the unique lower triangular matrix $\mathbf{L}$ such that $\mathbf{A} = \mathbf{LL}^{\mathrm{T}}$ [26, 27]. $\mathbf{L}$ is also band-diagonal, with $2W$ diagonals containing non-zero entries: $L[k,l] \neq 0$ only if $-2W < k - l \leq 0$. The Cholesky decomposition can be performed row by row by a fast version of Crout's algorithm [26] that takes advantage of the band-diagonal structure of $\mathbf{A}$.

2. The lower triangular system $\mathbf{L}\overset{\circ}{\mathbf{c}} = \mathbf{y}$ is solved by forward substitution: for k from $k_{\min}$ to $k_{\max}$,

$$\overset{\circ}{c}_T[k] = \frac{1}{L[k,k]}\Big(y[k] - \sum_{i=-2W+1}^{-1} L[k,k+i]\overset{\circ}{c}_T[k+i]\Big), \tag{18}$$

3. The upper triangular system $\mathbf{L}^{\mathrm{T}}\mathbf{c} = \overset{\circ}{\mathbf{c}}$ is finally solved by backward substitution: for k from $k_{\max}$ down to $k_{\min}$,

$$c_T[k] = \frac{1}{L[k,k]}\Big(\overset{\circ}{c}_T[k] - \sum_{i=1}^{2W-1} L[k+i,k]c_T[k+i]\Big), \tag{19}$$

4.2 Practical algorithm

The practical algorithm that computes the coefficients $c_T[k], k \in [-W+1, K+W-1]$, consists in two passes: the first pass implements the Cholesky decomposition and solves the first system (18), while the second pass solves the second linear system (19). We define the auxiliary variables $a[i] = A[k,k+i]$ and $u[k,i] = L[k+i,k]$. Then our algorithm can be written in pseudo-code as:

- First pass:

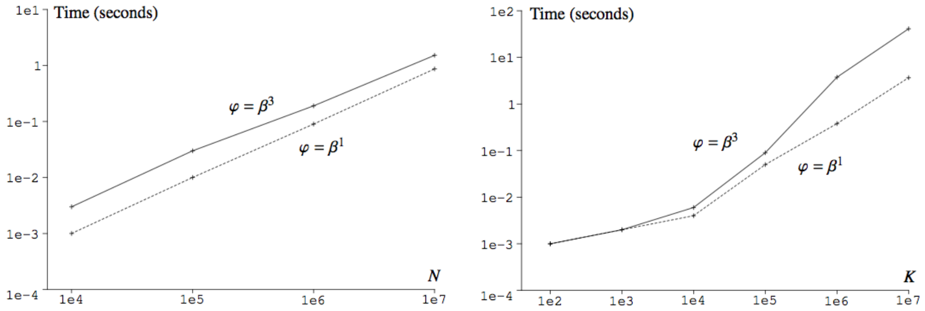
```
for $k$ from $-W+1$ to $K+W-1$ {
  $i_{\min} := \max(-2W+1, -W+1-k)$;
  $i_{\max} := \min(2W-1, K+W-1-k)$;
  for $i$ from 0 to $i_{\max}$,
    $a[i] := \sum_{n=0}^{N-1} \varphi(\frac{x[n]}{T} - k)\varphi(\frac{x[n]}{T} - k - i) + \lambda\, Q[k, k+i]$;
  $u[k,0] := \Big(a[0] - \sum_{i=i_{\min}}^{-1} u[k+i, -i]^2\Big)^{1/2}$;
  $\overset{\circ}{c}_T[k] := \frac{1}{u[k,0]}\Big(\sum_{n=0}^{N-1} \varphi(\frac{x[n]}{T} - k)v[n] - \sum_{i=i_{\min}}^{-1} u[k+i, -i]\,\overset{\circ}{c}_T[k+i]\Big)$;
  for $i$ from 1 to $i_{\max}$,
    $u[k,i] := \frac{1}{u[k,0]}\Big(a[i] - \sum_{j=\max(i-2W+1, -W+1-k)}^{-1} u[k+j, -j]\, u[k+j, i-j]\Big)$;
}
```

- Second pass:

```
for $k$ from $K+W-1$ down to $-W+1$ {
  $i_{\max} := \min(2W-1, K+W-1-k)$;
  $c_T[k] := \frac{1}{u[k,0]}\Big(\overset{\circ}{c}_T[k] - \sum_{i=1}^{i_{\max}} u[k,i]\, c_T[k+i]\Big)$;
}
```

In fact, the sums indexed by n have to be evaluated only for the few samples located in the interval $(T(k-W), T(k+W))$ that have a non-zero contribution. If the samples are ordered such that $x[n+1] \geq x[n]$ for every n, the data set $(x[n], v[n])$ can be accessed progressively. Hence the first pass can be computed on the fly, as the data are made available. For example, if $x[n]$ is the time of the measurement $v[n]$, the first pass can be performed in real-time with a delay of at most W time units. Moreover, during the second backward pass, the coefficients $c_T[k]$ can be computed in place, replacing the intermediate values $\overset{\circ}{c}_T[k]$. Note that this algorithm can be interpreted as a two-pass time-varying recursive filtering on the sequence $\mathbf{y}$, with a filter updated on the fly by the Cholesky decomposition.

If the data locations are not sorted, or if $N \gg K$, the computation time will be consumed mostly for the evaluation of the sums indexed by n. In this case,



(a) (b)

Figure 2: Computation time for the calculation of the coefficients $c_T[k], k \in [0, K]$, in log-log scale, for a reconstruction problem from N samples randomly located in the interval $[0, 100]$, as in Fig. 6 below ($T = 100/K$, $\lambda = 0.01$). (a): K is fixed to 100 and N is varying. (b): N is fixed to 10000 and K is varying. The dotted line is for linear spline reconstruction ($\varphi = \beta^1$, $r = 1$), while the solid line is for cubic spline reconstruction ($\varphi = \beta^2$, $r = 2$).

it is much more appropriate to use the following variant of the algorithm that consists in three passes. During the first pass, the upper part of the matrix $\mathbf{A}$ is stored temporarily in the coefficients $u[k, i]$. The $y[k]$ are also stored in the $\overset{\circ}{c}_T[k]$. The true values of these coefficients are computed in place in the second pass, as previously.

- First pass:

```
for $k$ from $-W + 1$ to $K + W - 1$ {
  $\overset{\circ}{c}_T[k] = 0$;
  for $i$ from 0 to $\min(2W - 1, K + W - 1 - k)$,
    $u[k, i] := \lambda\, Q[k, k + i]$;
}
for $n$ from 0 to $N - 1$,
  for $k$ from $\lfloor \frac{x[n]}{T} + 1 - W \rfloor$ to $\lceil \frac{x[n]}{T} - 1 + W \rceil$ {
    for $i$ from 0 to $\min(2W - 1, K + W - 1 - k)$,
      $u[k, i] := u[k, i] + \varphi(\frac{x[n]}{T} - k)\varphi(\frac{x[n]}{T} - k - i)$;
    $\overset{\circ}{c}_T[k] := \overset{\circ}{c}_T[k] + \varphi(\frac{x[n]}{T} - k) v[n]$;
  }
```

- Second pass:

```
for $k$ from $-W+1$ to $K+W-1$ {
    $i_{\min} := \max(-2W+1, -W+1-k)$;
    $i_{\max} := \min(2W-1, K+W-1-k)$;
    $u[k,0] := \Big(u[k,0] - \sum_{i=i_{\min}}^{-1} u[k+i,-i]^2\Big)^{1/2}$;
    $\mathring{c}_T[k] := \frac{1}{u[k,0]}\Big(\mathring{c}_T[k] - \sum_{i=i_{\min}}^{-1} u[k+i,-i]\,\mathring{c}_T[k+i]\Big)$;
    for $i$ from 1 to $i_{\max}$,
        $u[k,i] := \frac{1}{u[k,0]}\Big(u[k,i] - \sum_{j=\max(i-2W+1,-W+1-k)}^{-1} u[k+j,-j]\,u[k+j,i-j]\Big)$;
}
```

- Third pass:

```
for $k$ from $K+W-1$ down to $-W+1$ {
    $i_{\max} := \min(2W-1, K+W-1-k)$;
    $c_T[k] := \frac{1}{u[k,0]}\Big(\mathring{c}_T[k] - \sum_{i=1}^{i_{\max}} u[k,i]\,c_T[k+i]\Big)$;
}
```

These algorithms are for the strategy without boundary conditions, proposed in Section 4.2. The implementation for the method with mirror boundary conditions is very similar. In that case, the "folding" operation on the matrices is implemented by assigning each contribution $\varphi(\frac{x[n]}{T} - k)\varphi(\frac{x[n]}{T} - k - i)$ to its folded place directly, for example to $u[-k,-i]$ instead of $u[k,i]$ if $k < 0$.

4.3 Computation time and storage requirements

The computation time of the proposed algorithm can be modeled as $O(W^2N)$, $O(W^2K)$, $O(WN)$ for the calculation of the elements in $\mathbf{A}$, $\mathbf{L}$ and $\mathbf{y}$ respectively, and $O(KW)$ for the forward and backward substitutions. So, the total time reduces to $O(W^2(N+K))$; it is linear in N and K, which is the best one could have hoped for. If the reconstruction is to be performed on an interval $\mathcal{I}$ of fixed size $S = KT$, the total time may be rewritten $O(W^2(N + S/T))$, so as to let appear the linear dependence in $1/T$. Experimental computation times are reported in Fig. 2 for an implementation in C language of the algorithm proposed in Section 4.2 (second variant), running on a 1.6 GHz laptop PC. The computation time is asymptotically linear in K and N, as predicted.

Apart of the memory required to store the coefficients $c_T[k]$, auxiliary memory of size $2W(K+2W-1)$ units (or $2W(K+1)$ if using mirror boundary conditions) is needed to store the values $u[k,i]$, which are generated during the forward pass of our algorithm and used in the backward pass.

Instead of the Cholesky decomposition, it is possible to use a LU-factorization that does not exploit the symmetry of the matrix. Although this decomposition requires two times more computation, the square-root operator is not needed, and one diagonal is saved in the storage of $\mathbf{L}$; that is, $(2W-1)(K+2W-1)$ instead of $2W(K+2W-1)$ memory units are used. Another variant is the $\mathbf{LDL}^{\mathrm{T}}$ factorization, that requires one more pass on the data, but also avoids the square-root operator.

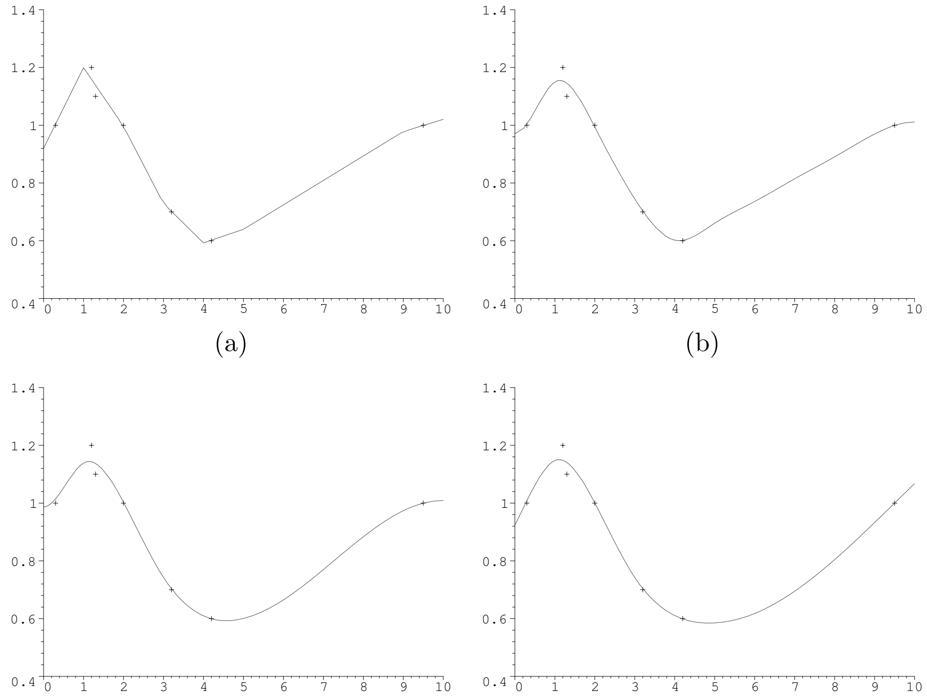
Note that our algorithm uses a Cholesky decomposition that is "extremely stable numerically" [26]. However, even if our linear system is positive definite, its condition number clearly depends on the sampling locations. In fact, our method amounts to performing the deconvolution of a time-varying filter. In large gaps without samples, this filter reduces to $q_{\varphi,r}$, which has roots on the complex unit circle. An inverse filter with poles on the unit circle is said to be *marginally stable*, because the impulse response corresponding to these poles does not decay, but does not grow either. So, if a round-off error occurs on a coefficient $c_T[k]$ during the computation, it can propagate to its neighbors inside a region without samples, but its amplitude, limited to the machine accuracy, will not grow. Therefore, this is not a problematic issue.

5 Choice of the parameters

In this section, we discuss the influence of the parameters φ, r, T, λ.

Since the reconstruction is performed in a LSI space, this space has to be chosen before hand. The "best" space depends on the characteristics of the signal to be modeled. If prior knowledge on the process that gave the samples is available, it can be used to choose a particular generator φ [28]. For instance, if the reconstructed function is required to be continuous and continuously differentiable, these properties will be enforced on φ. In concrete problems, spline spaces have shown to be particularly adequate for representing signals [15]. Their optimal approximation properties have been demonstrated theoretically [29, 17] and confirmed by practical experiments [30, 31, 32]. Part of their interest lies in the fact that a B-spline has the maximal possible approximation order, given the size of its support. The ability to reproduce high order polynomials is of primary concern for approximating arbitrary signals. Besides, for a given parameter r, choosing $\varphi = \beta^{2r-1}$ ensures that our solution f_T coincides with the RBF solution if the samples are uniform, at locations $x[n] = Tn$. This means that spline functions are the natural choice when minimizing a criterion based on a derivative.

The parameter r controls the kind of smoothness that is enforced on the solution. Three configurations are illustrated with a simple synthetic example in Fig. 3. In (a), the choice $\varphi = \beta^1, r = 1$ results in a piecewise linear recon-



(a)

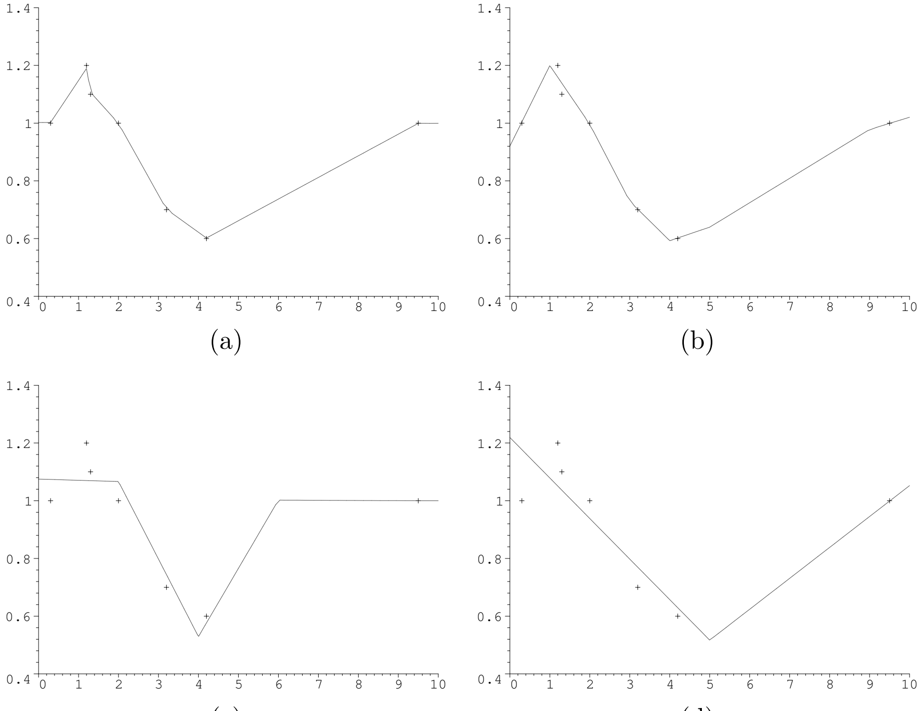
(b)

(c)

(d)

Figure 3: Uniform splines with knots at the integers ($T = 1$, $\lambda = 0.01$) fitted on 7 point samples in the interval $[0, 10]$, with different polynomial degrees and values of the regularization parameter r. (a): $\varphi = \beta^1$, $r = 1$. (b): $\varphi = \beta^3$, $r = 1$. (c), (d): $\varphi = \beta^3$, $r = 2$. Mirror boundary conditions are used for (a), (b), (c), not for (d).

struction, with knots at the Tk, $k \in \mathbb{Z}$, which means that $f_T(t)$ is linear on each interval $[Tk, T(k+1)]$. Note that with $\varphi = \beta^1$, there is no other possible choice than $r = 1$. Moreover, the two strategies for handling the reconstruction on a finite interval (no boundary conditions as in Section 4.2 or mirror boundary conditions as in Section 3.3) are equivalent in this case. In (b), (c), (d), $\varphi = \beta^3$ yields a smoother reconstruction, that is twice continuously differentiable. If $r = 1$, the variation $\int |\nabla f_T|^2$ is minimized, and the solution tends to behave like a straight line in large gaps, as in (b). If $r = 2$, the second derivative modelizes a curvature energy, and in large gaps, the solution f_T behaves like a polynomial of degree three, as in (c); in this case, f_T can go beyond the dynamics of the initial samples, which may be a disadvantage. Note that, in large gaps, we do not have information about the signal to reconstruct. So, fitting a polynomial using the data at the boundaries of the gap is, in essence, the best we can do.



(a)

(b)

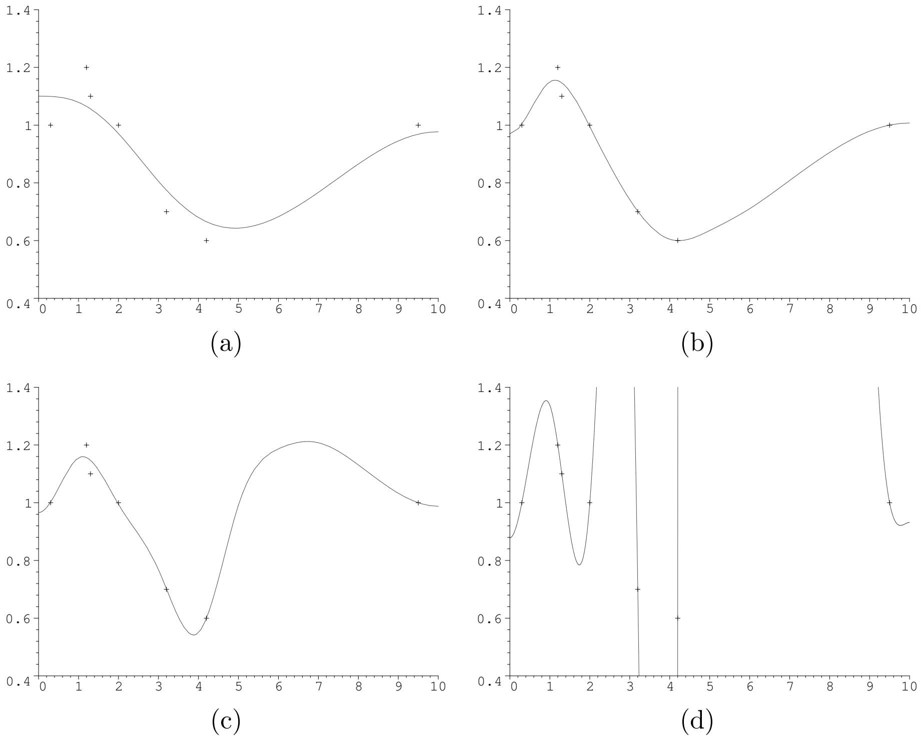
(c)

(d)

Figure 4: Uniform linear splines with different resolutions ($\varphi = \beta^1$, $r = 1$, $\lambda = 0.01$) fitted on 7 point samples in the interval $[0, 10]$. (a): $T = 0.1$. (b): $T = 1$. (c): $T = 2$. (d): $T = 5$. The splines have their knots at the Tk, $k \in [0, 10/T]$.

The choice of the boundary conditions is illustrated in Fig. 3 (c) and (d): mirror boundary conditions yield a reconstruction whose first derivative is constrained to be zero at the boundaries of the reconstruction interval. In this case, f_T is parameterized by 11 coefficients $c_T[k], k \in [0, 10]$. In (d), no boundary conditions are enforced, but the solution is now parameterized by 13 coefficients $c_T[k], k \in [-1, 11]$.

The parameter T controls the coarseness of the representation. When reconstructing a signal over an interval $[0, S]$, we obtain a parametric solution with $K + 1 = S/T + 1$ degrees of freedom. If the parsimony of the representation modeling the data is an important criterion, for instance in coding applications or if the computation time is limited, then T will be chosen relatively large. Conversely, when $T \to 0$, the solution f_T becomes closer and closer to the nonuniform solution in the RBF framework, since the space $V_T(\varphi)$ becomes dense in the whole Sobolev space of order r. The influence of T is illustrated in Fig. 4,



(a)

(b)

(c)

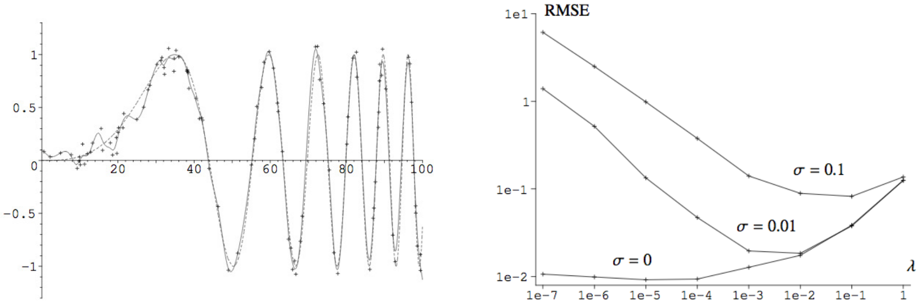
(d)

Figure 5: Uniform cubic splines with knots at the integers ($\varphi = \beta^3$, $r = 2$, $T = 1$, mirror boundary conditions) fitted on 7 point samples in the interval $[0, 10]$, for different values of the smoothing parameter λ. (a): $\lambda = 1.0$. (b): $\lambda = 0.001$. (c): $\lambda = 0.0001$. (d): limit case when $\lambda \to 0$.

with $\varphi = \beta^1, r = 1$. The function f_T is parameterized by $10/T + 1$ coefficients $c_T[k], k \in [0, 10/T]$. When $T \to 0$, f_T approaches the non-uniform smoothing spline of degree 1, which has its knots at the non-uniform sampling locations.

In practical applications, the parameter T will be matched to the cut-off frequency of the reconstruction lattice, as discussed in the next section. In fact, for each function $f_T \in V_T(\varphi)$, there is a one-to-one correspondence between its coefficients $c_T[k]$ and its point values $w_T[k] = f_T(Tk)$ at locations Tk. That is why $f_T \in V_T(\varphi)$ is said to have *resolution* $1/T$.

The regularization factor λ is a key parameter: an excessive value will oversmooth the solution, while a small value will provide a solution that is close to the data, but may have large disturbing variations. Let us consider the behavior of the reconstructed function f_T in the limit case when $\lambda \to 0$. The solution $\mathbf{c} = (\mathbf{M}^{\mathrm{T}}\mathbf{M} + \lambda\mathbf{Q})^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{s}$ has a well-defined limit. This limit corresponds to setting exactly $\lambda = 0$ in the equations only if the matrix $\mathbf{M}^{\mathrm{T}}\mathbf{M}$ is invertible.



(a) (b)

Figure 6: Reconstruction of the function $s(t) = \sin((\frac{t}{30})^3)$ (dotted line) from 100 noisy samples (additive Gaussian noise with standard deviation σ) at random locations in the interval [0,100]. (a): uniform cubic spline with knots at the integers ($\varphi = \beta^3$, $r = 2$, $T = 1$) fitted on the noisy data ($\sigma = 0.1$, $\lambda = 0.1$). (b): RMSE $e = (\frac{1}{100}\int_0^{100} |s(t) - f_T(t)|^2 dt)^{1/2}$ in function of the noise level σ and the smoothing parameter λ, in log-log scale.

This is the case, for example, if $0 < x[n+1] - x[n] < T$ for every n. In the general case, the limit solution has the following interpretation: when $\lambda \to 0$, the regularization term in (3) becomes neglictible in comparison with the fit-to-data term. So the limit function f_T minimizes the criterion $\sum_n \left|g(x[n]) - v[n]\right|^2$, and if the solution $g(t)$ is not unique, the remaining degrees of freedom are used to minimize $\int_{\mathcal{I}} |g^{(r)}(t)|^2$. Therefore, the behavior of the reconstructed function is the following: if λ is large, f_T is smooth, whatever the locations $x[n]$ of the data, as in Fig. 5 (a). If λ is very small, there are two cases: if locally the samples set is sparse (i.e. $x[n+1] - x[n] > T$ for a few successive values of n), then f_T almost passes through the samples, and is smooth inbetween (Fig. 5 (b), (c), (d), for $t > 3$). Thus large gaps are "in-painted" in a smooth way. Conversely, in a region where $x[n+1] - x[n] < T$ for a few successive values of n, f_T approaches the data at best in the least-squares sense (Fig. 5 (b), (c), (d), for $t < 2$). Thus, in the noise-free case, it is tempting to choose λ very small. However, this may result in large unexpected oscillations, going far beyond the initial range of the signal, as shown in Fig. 5 (c), (d): the exact interpolation of the samples is a too strong constraint.

In practice, the measurements are often noisy or lacking precision and it is not suitable to reconstruct a function that interpolates the data exactly. It is therefore appropriate to achieve a tradeoff between the closeness of fit and the smoothness of the solution through the parameter λ. The best value for the problem at hand has to be determined on a case by case basis. There is no rule giving an optimal value, and empirical adjustment is the most reliable technique. It is also possible to learn λ from the data by cross-validation [33, 34]. If we consider a stochastic framework where the data come from a random stationary process with additive random noise, both with known spectral characteristics, it is suggested in [35] to choose λ inversely proportional to the signal-to-noise ratio. This may serve as a heuristic in the deterministic case. This is confirmed in the example shown in Fig. 6, where $\lambda = \sigma$ yields the minimum root-mean-square error, when approximating an unknown signal from its samples contaminated by additive noise with standard deviation σ.

6 Applications

There are plenty of problems where it is useful to fit a parametric curve on discrete data. Numerous methods have been proposed, generally issued from statistical estimation theory [36]. Our approach allows to reconstruct a function f_T that is resolution dependent, a feature that offers many advantages in practical applications. For example, if a non-uniform signal is to be rendered on a display device with point-spread-function $\Gamma(t)$, it is straightforward to apply the proposed approach: we choose $\varphi = \Gamma$ and match T to the resolution of the device, so that $V_T(\varphi)$ is the set of all continuously-defined signals that can be rendered by the device. Therefore, by minimizing the criterion in (3), we ensure that our solution is optimal, given the available data.

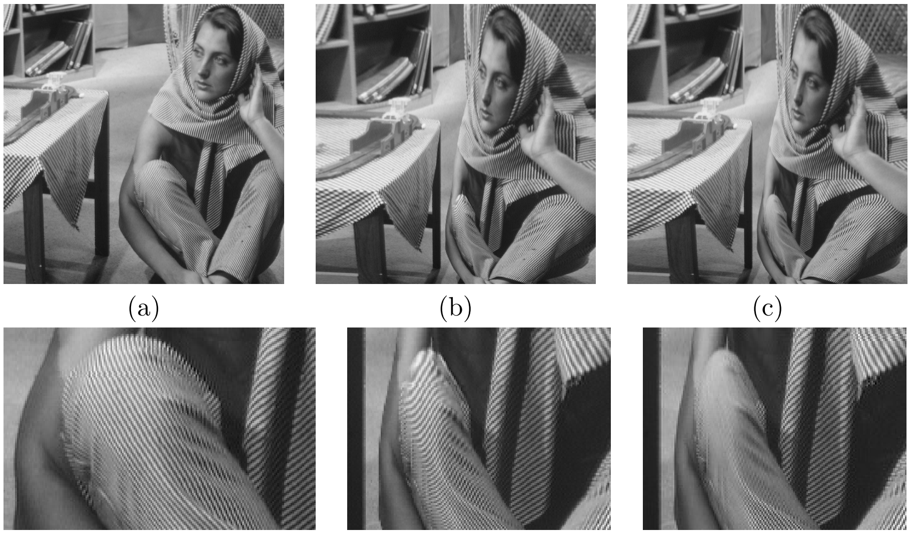
Another potential field of applications is image analysis using multiscale "pyramidal" representations. The proposed work can be used for this task, if not only a single function but a whole collection of functions $\{f_T\}$, with different values of T, is computed. For instance, the performances of procedures such as edge detection or image registration can be improved by processing from coarser to finer levels [37]. To this purpose, our approach is more general than classical dyadic representations, that only apply to uniform signals and discrete dyadic resolutions ($T = 2^n, n \in \mathbb{N}$).

Our approach is also particularly adapted to resampling problems involving rate conversions, such as image resizing. Let us present some generic problems that could benefit from our approach.

6.1 Non-uniform to uniform resampling

Given the non-uniform samples $v[n]$ at locations $x[n]$, suppose we want to obtain resampled values $w_T[k]$ located on the uniform reconstruction lattice $(Tk)_{k\in\mathbb{Z}}$. We can simply compute the function $f_T(t)$ and resample it on the uniform lattice: this yields $w_T[k] = f_T(Tk)$ for every k. In fact, once the $c_T[k]$ have been computed, the signal w_T is directly obtained by digital filtering [15]: $w_T = c_T * b^{-1}$, where $b[k] = \varphi(k)$ for every $k \in \mathbb{Z}$.

With this method, the representation capabilities of the target lattice, where the resampled signal lives, are exploited optimally: $f_T(t)$ retains at best the information contained in the non-uniform samples and representable on this lattice. Conversely, the information that is not representable is canceled out, and no irrelevant structure is introduced. As a consequence, the aliasing issue is automatically handled because, qualitatively, the frequencies higher than the Nyquist rate of the target lattice are not representable in $V_T(\varphi)$. That is why



(a) (b) (c)

(d) (e) (f)

Figure 7: Image *Barbara* (a) warped using: backward mapping (b), and proposed approach with $\lambda = 10^{-4}$ (c) (see text). (d), (e), (f): zoom-ins on (a), (b), (c) respectively.

T is chosen so that f_T has a resolution matched to the cut-off frequency of the reconstruction lattice. Note that we assume that φ is lowpass, and that λ is correctly chosen so as not to distort the frequency content of the data.

This approach is also useful for resampling a uniform ($x[n] = n$) signal v. If v is to be resampled at coarser resolution or magnified with a non-integer factor, interpolation followed by resampling can introduce severe distortions [38]. Instead, we look for a function having the resolution of the target resampled signal, and not of the source signal. This formulation has been proposed previously in [39], in the more restrictive case where T is an integer.

Interestingly, our approach amounts to formulate resampling as an inverse problem: we seek the uniform signal w_T that, when interpolated by the function $f_T(t)$ and resampled back on the source lattice ($x[n]$), is the closest to the initial signal ($v[n]$). This is the opposite of the forward approach that fits a function on the source signal, typically by interpolation, and then resample it, independently of the target lattice.

6.2 Warping

Another application is signal or image warping. Suppose we have the uniform samples $v[n] = s(n)$ of an unknown function $s(t)$ at our disposal, and we want to compute the uniform samples $(s(\mathcal{W}^{-1}(n)))_{n\in\mathbb{Z}}$ of the warped function $s(\mathcal{W}^{-1}(t))$ for some reversible transform $\mathcal{W}(t)$. The classical method used for warping, called *backward mapping*, consists in interpolating the uniform samples $v[n]$ with a function $f(t)$, that is then warped and resampled, providing the warped signal $(f(\mathcal{W}^{-1}(n)))_{n\in\mathbb{Z}}$. Artifacts may appear with this method because the spectrum of the warped function $f(\mathcal{W}^{-1}t)$ extends potentially beyond the Nyquist frequency of the target lattice, resulting in aliasing when this function is sampled. In fact, the samples $v[n] = s(\mathcal{W}^{-1}(\mathcal{W}(n)))$ of the warped function at the non-uniform locations $x[n] = \mathcal{W}(n)$ are known, so we are back to a non-uniform to uniform resampling problem, solved as previously. For example, arbitrary time-delay is a particular case of warping, with $\mathcal{W}(t) = t + \tau$, for a delay $\tau \in \mathbb{R}$.

The proposed algorithm can be used for image warping as well, as long as the 2-D warping $\mathcal{W}(x, y)$ is separable, by applying the resampling algorithm on each row and then on each column of the image. It is more rigorous to use a truly 2-D approach with a 2-D variational criterion, as proposed in [23, 24], but one has to give up our direct and fast algorithm in that case. As an example, the image *Barbara* is warped with our approach in Fig. 7 with $\mathcal{W}(x/511, y/511)/511 = (2.2x - 3.6x^2 + 2.4x^3, 0.5y^2 + 0.5y)$, for $(x, y) \in [0, 511]^2$. As can be seen in Fig. 7 (b), (e), aliasing appears on the trousers with backward mapping, because the stripes in the initial image are not representable any more in the warped image: their frequency has become higher than the Nyquist frequency of the target lattice. Our method is free from this drawback, and renders a flat zone without aliasing.

7 Conclusion

In this article, reconstruction from non-uniform samples is stated as a variational problem in a shift-invariant space. This formulation yields a well-defined solution, whatever the sample locations, which may be completely arbitrary. Moreover, the reconstructed function has a given resolution that can be tuned, for example to match the representation capabilities of a target lattice for resampling purpose, or the desired rate for coding. Taking into account the particular structure of the problem, we have proposed an efficient algorithm that computes the exact solution of the optimization problem, by solving a band-diagonal linear system, without having to represent and store any matrix explicitly.

8 Proof that A is positive definite

First, $\mathbf{M}^{\mathrm{T}}\mathbf{M}$ is positive semi-definite, i.e. $\mathbf{u}^{\mathrm{T}}\mathbf{M}^{\mathrm{T}}\mathbf{M}\mathbf{u} \geq 0$ for every vector $\mathbf{u}$, since $\mathbf{u}^{\mathrm{T}}\mathbf{M}^{\mathrm{T}}\mathbf{M}\mathbf{u} = \|\mathbf{M}\mathbf{u}\|^2_{\ell_2} \geq 0$. $\mathbf{Q}$ is also positive semi-definite: let us choose a vector $\mathbf{u}$ and define $g(t) = \sum_{k=-W+1}^{K+W-1} u[k]\varphi(\frac{t}{T} - k)$. Then $\mathbf{u}^{\mathrm{T}}\mathbf{Q}\mathbf{u} = \int_{\mathcal{I}} |g^{(r)}(t)|^2 dt$ by construction of $\mathbf{Q}$, and this integral is positive.

Now, let us show that $\mathbf{A}$ is positive definite, i.e. $\mathbf{u}^{\mathrm{T}}\mathbf{A}\mathbf{u} = 0 \Rightarrow \mathbf{u} = 0$. We suppose that $\mathbf{u}^{\mathrm{T}}\mathbf{A}\mathbf{u} = 0$. Then $\mathbf{u}^{\mathrm{T}}\mathbf{M}^{\mathrm{T}}\mathbf{M}\mathbf{u} = 0$ and $\mathbf{u}^{\mathrm{T}}\mathbf{Q}\mathbf{u} = 0$. This yields $g(x[n]) = 0$ for every $n \in [0, N-1]$ and $\int_{\mathcal{I}} |g^{(r)}(t)|^2 dt = 0$. Therefore, $g^{(r)}(t) = 0$

within $\mathcal{I}$, and $g(t)$ is a polynomial of degree less than r in this interval. This polynomial has N roots at the $x[n]$ (with at least r distinct roots by hypothesis on the samples locations). Then $g = 0$ and $u[k] = 0$ for every k. □

References

[1] C. E. Shannon, "Communication in the presence of noise," *Proc. of the Inst. of Radio Eng.*, vol. 37, no. 1, pp. 10–21, 1949.

[2] K. Gröchenig and H. Razafinjatovo, "On Landau's necessary density conditions for sampling and interpolation of band-limited functions," *J. Lond. Math. Soc.*, vol. 54, no. 3, pp. 557–567, 1996.

[3] W. Chen and S. Itoh, "A sampling theorem for shift-invariant subspaces," *IEEE Trans. Signal Processing*, vol. 46, no. 10, pp. 2822–2824, Oct. 1998.

[4] A. Aldroubi and K. Gröchenig, "Beurling-Landau-type theorems for non-uniform sampling in shift invariant spline spaces," *J. Fourier Anal. Appl.*, vol. 6, no. 1, pp. 93–103, 2000.

[5] H. G. Feichtinger, K. Gröchenig, and T. Strohmer, "Efficient numerical methods in non-uniform sampling theory," *Numer. Math.*, no. 4, pp. 423–440, 1995.

[6] T. Strohmer, "Numerical analysis of the non-uniform sampling problem," *J. Comp. Appl. Math.*, vol. 112, no. 1-2, pp. 297–316, 2000.

[7] Y. Liu, "Irregular sampling for spline wavelet subspaces," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 623–627, 1996.

[8] A. Aldroubi and K. Gröchenig, "Nonuniform sampling and reconstruction in shift-invariant spaces," *SIAM Rev.*, vol. 43, no. 4, pp. 585–620, 2001.

[9] W. Chen, B. Han, and R.-Q. Jia, "Maximal gap of a sampling set for the exact iterative reconstruction algorithm in shift-invariant spaces," *IEEE Signal Processing Lett.*, vol. 11, no. 8, pp. 655–658, Aug. 2004.

[10] C. D. Boor, R. DeVore, and A. Ron, "The structure of finitely generated shift-invariant spaces in $L_2(\mathbb{R}^d)$," *J. Funct. Anal. 119*, pp. 37–78, 1994.

[11] R.-Q. Jia, "Shift-invariant spaces and linear operator equations," *Israel J. Math.*, vol. 41, pp. 259–288, 1998.

[12] I. Daubechies, *Ten lectures on wavelets.* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1992.

[13] M. Unser, "Sampling—50 Years after Shannon," *Proc. IEEE*, vol. 88, no. 4, pp. 569–587, Apr. 2000.

[14] C. de Boor, *A Practical Guide to Splines.* New York: Springer-Verlag, 1978.

[15] M. Unser, "Splines: A perfect fit for signal and image processing," *IEEE Signal Processing Mag.*, vol. 16, no. 6, pp. 22–38, Nov. 1999.

[16] M. Jacob, T. Blu, and M. Unser, "Efficient energies and algorithms for parametric snakes," *IEEE Trans. Image Processing*, vol. 13, no. 9, pp. 1231–1244, Sep. 2004.

[17] T. Blu, P. Thévenaz, and M. Unser, "Moms: Maximal-order interpolation of minimal support," *IEEE Trans. Image Processing*, vol. 10, no. 7, pp. 1069–1080, Jul. 2001.

[18] R. Madani, A. Ayremlou, A. Amini, and F. Marvasti, "Optimized compact-support interpolation kernels," *IEEE Trans. Signal Processing*, vol. 60, no. 2, pp. 626–633, Feb. 2012.

[19] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1–4, pp. 259–268, 1992.

[20] C. M. Brislawn, "Classification of nonexpansive symmetric extension transforms for multirate filter banks," *Applied and Comp. Harmonic Anal.*, vol. 3, pp. 337–357, 1996.

[21] K. Gröchenig and H. Schwab, "Fast local reconstruction methods for nonuniform sampling in shift invariant spaces," *SIAM J. Matrix Anal. and Appl.*, vol. 24, no. 4, pp. 899–913, 2003.

[22] R. L. Eubank, *Nonparametric regression and spline smoothing.* New York, NY, USA: Marcel Dekker, 1999.

[23] M. Arigovindan, M. Sühling, P. Hunziker, and M. Unser, "Variational image reconstruction from arbitrarily spaced samples: A fast multiresolution spline solution," *IEEE Trans. Image Processing*, vol. 14, no. 4, pp. 450–460, Apr. 2005.

[24] C. Vázquez, E. Dubois, and J. Konrad, "Reconstruction of nonuniformly sampled images in spline spaces," *IEEE Trans. Image Processing*, vol. 14, no. 6, pp. 713–725, Jun. 2005.

[25] A. Muñoz Barrutia, T. Blu, and M. Unser, "Non-uniform to uniform grid conversion using least-squares splines," in *Proc. of EUSIPCO*, vol. 4, Tampere, Finland, Sep. 2000, pp. 1997–2000.

[26] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing.* New York, NY, USA: Cambridge University Press, 1992.

[27] G. H. Golub and C. F. V. Loan, *Matrix computations*, 2nd ed. Baltimore: Johns Hopkins Univ. Press, 1989.

[28] S. Ramani, D. Van De Ville, and M. Unser, "Sampling in practice: Is the best reconstruction space bandlimited?" in *Proc. of IEEE ICIP*, vol. 2, Sep. 2005, pp. 153–156.

[29] T. Blu and M. Unser, "Quantitative Fourier analysis of approximation techniques: Part I–interpolators and projectors—and part II–wavelets," *IEEE Trans. Signal Processing*, vol. 47, no. 10, pp. 2783–2806, Oct. 1999.

[30] T. M. Lehmann, C. Gönner, and K. Spitzer, "Survey: Interpolation methods in medical image processing," *IEEE Trans. Med. Imag.*, vol. 18, no. 11, pp. 1049–1075, Nov. 1999.

[31] P. Thévenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Trans. Med. Imag.*, vol. 19, no. 7, pp. 739–758, Jul. 2000.

[32] E. Meijering, W. Niessen, and M. Viergever, "Quantitative evaluation of convolution-based methods for medical image interpolation," *Medical Image analysis*, vol. 5, no. 2, pp. 111–126, Jun. 2001.

[33] P. Craven and G. Wahba, "Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation," *Numerical Mathematics*, vol. 31, pp. 377–403, 1979.

[34] M. F. Hutchinson and F. R. De Hoog, "Smoothing noisy data with spline functions," *Numer. Math.*, vol. 4, no. 7, pp. 99–106, 1985.

[35] Y. C. Eldar and M. Unser, "Non-ideal sampling and interpolation from noisy observations in shift-invariant spaces," *IEEE Trans. Signal Processing*, vol. 54, no. 7, pp. 2636–2651, Jul. 2006.

[36] G. Wahba, *Spline models for observational data*. Philadelphia, PA: Society for industrial and applied mathematics (SIAM), 1990.

[37] P. Thévenaz, U. E. Ruttimann, and M. Unser, "A pyramid approach to subpixel registration based on intensity," *IEEE Trans. Image Processing*, vol. 7, no. 1, pp. 27–41, Jan. 1998.

[38] M. Unser, A. Aldroubi, and M. Eden, "Enlargement or reduction of digital images with minimum loss of information," *IEEE Trans. Image Processing*, vol. 4, no. 3, pp. 247–258, Mar. 1995.

[39] A. Aldroubi, M. Eden, and M. Unser, "Discrete spline filters for multiresolution and wavelets of l_2," *SIAM J. Math. Anal.*, vol. 25, no. 5, pp. 1412–1432, Sep. 1994.