

Fast Computation of Exponential B-splines

Takeshi Asahi, Koichi Ichige and Rokuya Ishii

Division of Electrical and Computer Engineering, Yokohama National University

Tokiwadai 79-5, Hodogaya-ku, Yokohama 240-8501, Japan

Tel: +81-45-339-4138, Fax: +81-45-338-1157

E-mail: takeshiasahi@netscape.net, koichi@dnj.ynu.ac.jp, ishii@dnj.ynu.ac.jp

Abstract: This paper proposes a fast method for the calculation of exponential B-splines sampled at regular intervals. This algorithm is based on a combination of FIR and IIR filters which enables a fast decomposition and reconstruction of a signal. When complex values are selected for the parameters of the exponentials, complex trigonometric functions are obtained. Only the real part of these functions are used for the interpolation of real signals, leading less bandlimited signals when they are compared with the polynomial B-spline counterparts. These characteristics were verified with 1-D and 2-D examples.

1. Introduction

Splines consist of piecewise functions, which have a certain continuity at some joint points called knots. Until now, in the family of splines, the polynomial ones have been extensively developed [1]. However, a more general formulation of these splines are the exponential ones [2, 3], where polynomial splines can be considered as a specific case.

In the present work, a fast algorithm is proposed in order to calculate discrete exponential B-splines at equally spaced knots. This algorithm is an extension of a fast algorithm for calculating the polynomial splines [4], where its calculation is mainly based on the application of IIR filters in the decomposition, as well as the reconstruction of the interpolated signal. Therefore, the calculation of discrete polynomial B-splines is a particular case, when the parameters of the exponents are set to be zero. Besides, the use of a complex parameter in the exponent leads to one kind of trigonometric splines. In this way, less band-limited functions can be obtained in order to carry out an interpolation. This fact may prove advantageous for the processing of digital images.

2. Exponential B-splines and cone splines

A family of continuous piecewise basic spline functions can be obtained by multifold convolution of functions $w^i(x), i = 1, \dots, n$:

$$\gamma^n(x) = \underbrace{w^1 \circledast \dots \circledast w^n}_{n}(x),$$

where \circledast denotes the convolution integral operation. These weights are defined in the domain $x \in [-1/2, 1/2]$ and being 0 otherwise for the case of the centered basic splines and the domain $x \in [0, 1]$ for the shifted ones.

In the case of polynomial B-splines, $w^i(\cdot)$ is a rectangular function of height 1. An example of a polynomial B-spline (order $n = 2$) can be seen in fig.1(a).

In order to obtain the exponential B-splines, the weights will be exponential functions $w^{\mu_i}(x) = w_1^{\mu_i} \exp(\mu_i x)$, with $\mu_i \in \mathbf{C}$ being the parameter of the exponential function and $w_1^{\mu_i} \in \mathbf{C}$ a normalization factor. So, by defining the vector $\vec{\mu}_r = (\mu_1, \dots, \mu_r)^T$, the exponential B-spline can also be expressed as [2]:

$$\mathcal{T}^{\vec{\mu}_n}(x) = \sum_{k \in \mathbf{Z}} d^{\vec{\mu}_n}[k] \sigma^{\vec{\mu}_n}(x - k),$$

where the single and multiple discrete difference functions are given by

$$\begin{aligned} d^{\mu_r}[k] &= \delta[k] - e^{\mu_r} \delta[k - 1] \\ d^{\vec{\mu}_n}[k] &= \underbrace{d^{\mu_1} * d^{\mu_2} * \dots * d^{\mu_n}}_n[k] \end{aligned}$$

respectively, with $\delta[k]$ denoting the Kronecker's delta function. Also, $\sigma^{\vec{\mu}_n}(\cdot)$ is the continuous exponential truncated power or exponential cone spline. The latter can be defined by

$$\sigma^{\mu_1}(x) = \begin{cases} w^1(x - 1/2) & x \in [0, \infty) \\ 0 & \text{otherwise} \end{cases}$$

and then recursively calculated by

$$\sigma^{\vec{\mu}_r}(x) = \int_0^\infty w^r(x_1 - 1/2) \sigma^{\vec{\mu}_{(r-1)}}(x - x_1) dx_1.$$

It is convenient to introduce the notation for the expanded exponential B-spline:

$$\begin{aligned} \mathcal{T}_m^{\vec{\mu}_n}(x) &= \mathcal{T}^{\vec{\mu}_n}(x/m) \\ &= \lambda_m^{\vec{\mu}_n} \sum_{k \in \mathbf{Z}} d^{\vec{\mu}_n}[k] \sigma^{\vec{\mu}_n/m}(x - mk) \end{aligned} \quad (1)$$

where m is a positive integer and

$$\lambda_m^{\vec{\mu}_n} = m \prod_{r=1}^n \frac{\sinh(\mu_r/2m)}{\sinh(\mu_r/2)} \exp\left(-\frac{m-1}{2m} \mu_r\right)$$

is a scalar factor necessary for the expansion.

The continuous exponential cone spline $\sigma^{\vec{\mu}_n}(\cdot)$ can be related to the discrete exponential cone spline $s^{\vec{\mu}_n}[\cdot]$ by the following equation [3]:

$$\sigma^{\vec{\mu}_n}(x) = \sum_{\ell \geq 0} s^{\vec{\mu}_n}[\ell] \mathcal{T}^{\vec{\mu}_n}(x - \ell),$$

where the inverse of the discrete function $d^{\vec{\mu}_n}[\cdot]$ is defined as

$$s^{\vec{\mu}_n}[\ell] \xrightarrow{z} S^{\vec{\mu}_n}(z) = \prod_{r=1}^n \frac{1}{1 - e^{\mu_r} z^{-1}}. \quad (2)$$

By using this relation, equation (1) becomes

$$\mathcal{T}_m^{\vec{\mu}_n}(x) = \lambda_m^{\vec{\mu}_n} \sum_{\ell \in \mathbf{Z}} \left((d^{\vec{\mu}_n})_{\uparrow m} * s^{\vec{\mu}_n/m} \right) [\ell] \mathcal{T}^{\vec{\mu}_n/m}(x - \ell) \quad (3)$$

where the up-sampling operator can be written as

$$(f)_{\uparrow m}[\tilde{\ell}] = \begin{cases} f[\ell], & \tilde{\ell} = m\ell, \\ 0, & \text{otherwise.} \end{cases}$$

Note that if the expanded exponential B-spline is sampled at integer points $x = k, k \in \mathbf{Z}$, the discrete exponential B-spline $t_m^{\vec{\mu}_n}[\cdot]$ is obtained:

$$t_m^{\vec{\mu}_n}[k] = \lambda_m^{\vec{\mu}_n} \underbrace{(t_m^{\mu_1} * \dots * t_m^{\mu_n})}_n * t_1^{\vec{\mu}_n/m}[k], \quad (4)$$

where $t_m^{\mu_r}[\ell] = ((d^{\mu_r})_{\uparrow m} * s^{\mu_r/m})[\ell]$ is the weight $w^{\mu_r}(\cdot)$ sampled at an interval of $1/m$.

3. Decomposition/Reconstruction of a Signal

3.1 Calculation of the spline coefficients

An exponential spline function can be constructed by the integer shift of a base exponential B-spline function $\mathcal{T}^{\vec{\mu}_n}(\cdot)$:

$$f(x) = \sum_{\ell \in \mathbf{Z}} c[\ell] \mathcal{T}^{\vec{\mu}_n}(x - \ell) \quad (5)$$

where $c[\cdot]$ is referred as the B-spline coefficients.

When a discrete signal $y[k], k \in \mathbf{Z}$ is interpolated, the condition will be the function of eq. (5) evaluated at integer points ($x = k, k \in \mathbf{Z}$) has the same values as the discrete signal $y[\cdot]$.

To obtain the continuous model of the signal, the values of the coefficients are necessary. Therefore, by using the following equation

$$c[k] = \left(t^{\vec{\mu}_n} \right)^{-1} * y[k], \quad (6)$$

the fast calculation algorithm of the inverse filter $(t^{\vec{\mu}_n})^{-1}$ can be implemented by means of recursive IIR causal and anti-causal filters. This procedure corresponds to the decomposition of a discrete signal.

3.2 Fast calculation of the discrete exponential B-spline

In order to quickly calculate the exponential B-spline, we express eq. (4) in terms of the discrete exponential cone spline:

$$t_m^{\vec{\mu}_n}[k] = \lambda_m^{\vec{\mu}_n} s^{\vec{\mu}_n/m} * \left((d^{\vec{\mu}_n})_{\uparrow m} * t_1^{\vec{\mu}_n/m} \right) [k]. \quad (7)$$

From the z -transform of the discrete function $s^{\vec{\mu}_n}[\cdot]$ given in eq. (2), it can also be seen as an IIR filter that can be expressed as follows:

$$(f)_{\mu_i}[k] = \sum_{\ell=0}^{\infty} f[k - \ell] e^{\mu_i \ell} \xrightarrow{z} S^{\mu_i}(z) F(z)$$

where $f[\cdot]$ is a causal function ($f[k] = 0$ for $k < 0$) and $(\cdot)_{\mu_i}$ is the operator of the IIR filter. This procedure can be implemented recursively as

$$(f)_{\mu_i}[k] = f[k] + e^{\mu_i} (f)_{\mu_i}[k - 1].$$

Finally, the definition for multiple weights is

$$(f)_{\vec{\mu}_i}[k] = \left((f)_{\vec{\mu}_{(i-1)}} \right)_{\mu_i}[k],$$

where the IIR filters are applied in cascade, in order to express eq.(7) as

$$t_m^{\vec{\mu}_n}[k] = \lambda_m^{\vec{\mu}_n} \left((d^{\vec{\mu}_n})_{\uparrow m} * t_1^{\vec{\mu}_n/m} \right)_{\vec{\mu}_n/m}[k]. \quad (8)$$

For the reconstruction of signals, it is assumed that the continuous signal is oversampled by an integer multiple (or sampled at $1/k, k \in \mathbf{Z}$, intervals). Therefore, the signal $f_m[k] = f(k/m)$ can be obtained:

$$f_m[k] = (c)_{\uparrow m} * t_m^{\vec{\mu}_n}[k],$$

with $t_m^{\vec{\mu}_n}[\cdot]$ given in eq. (8). In the IIR filter form, the previous equation becomes:

$$f_m[k] = \lambda_m^{\vec{\mu}_n} \left((d^{\vec{\mu}_n} * c)_{\uparrow m} * t_1^{\vec{\mu}_n/m} \right)_{\vec{\mu}_n/m}[k]. \quad (9)$$

3.3 Complete algorithm

By using equations (6) and (9), a Difference/IIR Filter (DIF) algorithm can be composed as follows:

(DIF1) Obtain the coefficients of the exponential spline $\{c[\cdot]\}$ by using inverse filter as shown in eq. (6), convolve it with the sequence $\{d^{\vec{\mu}_n}[\cdot]\}$ and up-sample the result by a factor of m .

(DIF2) Convolute the result of **(DIF1)** with the exponential B-spline $t_1^{\vec{\mu}_n/m}[\cdot]$ of weight $\vec{\mu}_n/m$ sampled at the knot points.

(DIF3) Apply the IIR filters in cascade $(\cdot)_{\vec{\mu}_n/m}$ to the result in **(DIF2)**. Finally, multiply by the resulting function by $\lambda_m^{\vec{\mu}_n}$.

4. Implementation

A general solution of the exponential B-spline of order $n = 4$ is given in table 1 where the factors are defined as $w_m^{\vec{\mu}_n} = \prod_{\ell=1}^n \frac{\mu_\ell}{2m \sinh(\mu_\ell/2m)}$ and $\varepsilon_\ell^{\vec{\mu}_n} = \sum_{\substack{r=1 \\ r \neq \ell}}^n \exp(\mu_r)$. Also, the notation for the sum of the components of a vector is $|\vec{\mu}_n|_1 = \sum_{\ell=1}^n \mu_\ell$.

When the values of the different μ_r are imaginary, one kind of trigonometric splines is obtained. Because

Table 1. Piecewise functions for the exponential B-splines of order $n = 4$.

$\mathcal{T}^{\vec{\mu}_4}(x)$	domain
$w_1^{\vec{\mu}_4} \exp(-\frac{1}{2} \vec{\mu}_4 _1) \sum_{\ell=1}^4 \frac{\exp \mu_\ell(x+2)}{v_\ell^{\vec{\mu}_4}}$	$-2 \leq x < -1$
$-w_1^{\vec{\mu}_4} \exp(-\frac{1}{2} \vec{\mu}_4 _1) \sum_{\ell=1}^4 \frac{\varepsilon_\ell^{\vec{\mu}_4}}{v_\ell^{\vec{\mu}_4}} \exp \mu_\ell(x+1)$	$-1 \leq x < 0$
$w_1^{\vec{\mu}_4} \exp(\frac{1}{2} \vec{\mu}_4 _1) \sum_{\ell=1}^4 \frac{\varepsilon_\ell^{-\vec{\mu}_4}}{v_\ell^{\vec{\mu}_4}} \exp \mu_\ell(x-1)$	$0 \leq x < 1$
$-w_1^{\vec{\mu}_4} \exp(\frac{1}{2} \vec{\mu}_4 _1) \sum_{\ell=1}^4 \frac{\exp \mu_\ell(x-2)}{v_\ell^{\vec{\mu}_4}}$	$1 \leq x < 2$

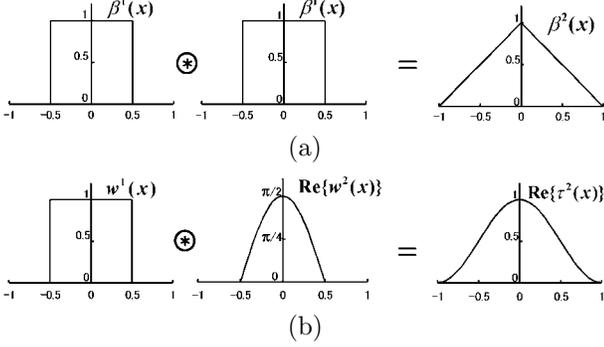


Figure 1. Generation of (a) polynomial and (b) exponential B-splines by convolution ($n = 2$).

we are dealing with real signals, the real part of the resulting B-spline is extracted from the reconstructed function. As an example, a 2nd order exponential B-spline can be seen in fig.1(b).

Here we propose some values:

$$\mu_r = \begin{cases} 0 & \text{for } r = 1 \\ j\frac{2\pi}{r} & \text{for } r \geq 2, \end{cases} \quad (10)$$

because the periods of the exponentials are r for $r > 1$. The weight $\mu_1 = 0$ enables the property of the partition of unity

$$\sum_k \mathcal{T}^{\vec{\mu}_n}(x-k) = 1, \quad \forall x \in \mathbf{R},$$

regardless the values of the other weights.

In the present work, the exponential B-spline of order $n = 4$, corresponding to a vector of weights $\vec{\mu}_4 = j\pi(0, 1, \frac{2}{3}, \frac{1}{2})$ is considered. The necessary values of discrete function $t^{\vec{\mu}_4/m}[\cdot]$ are shown in table 2.

For the case where $m = 1$, the z -transform of the $t_1^{\vec{\mu}_n}[\cdot]$ filter will be

$$T^{\vec{\mu}_n}(z) = a_1^* z^{-1} + a_0 + a_1 z$$

with $a_0 = 1.07735$ and $a_1 = -0.03868 + j0.30662$. However, because we are more interested in the real part of $T_m^{\vec{\mu}_n}$, the inverse filter of $\tilde{T}^{\vec{\mu}_n}(z) = \Re\{a_1\}z^{-1} + a_0 + \Re\{a_1\}z$ becomes:

$$\left(\tilde{T}^{\vec{\mu}_n}(z)\right)^{-1} = 0.93062 \left(\frac{1}{1 + \alpha z^{-1}} + \frac{1}{1 + \alpha z} - 1 \right)$$

Table 2. Values of the exponential B-splines of order $n = 4$ at the knot points.

k	$t^{\vec{\mu}_4/m}[k]$
-1	$j\frac{3m^3}{\pi^3} w_m^{\vec{\mu}_4} \exp(-j\frac{13\pi}{12m}) \left\{ 8 \exp j\frac{\pi}{2m} - 9 \exp j\frac{2\pi}{3m} + 2 \exp j\frac{\pi}{m} - 1 \right\}$
0	$\frac{6m^3}{\pi^3} w_m^{\vec{\mu}_4} \left\{ 10 \sin \frac{5\pi}{12m} - 7 \sin \frac{7\pi}{12m} - \sin \frac{\pi}{12m} \right\}$
1	$-j\frac{3m^3}{\pi^3} w_m^{\vec{\mu}_4} \exp(j\frac{13\pi}{12m}) \left\{ 8 \exp(-j\frac{\pi}{2m}) - 9 \exp(-j\frac{2\pi}{3m}) + 2 \exp(-j\frac{\pi}{m}) - 1 \right\}$

where $\alpha = -0.03595$. This inverse filter can be divided into stable causal and anti-causal IIR filters ($|\alpha| < 1$), which enables a fast calculation. This method is the same as the one used for the inverse filter of polynomial B-splines [5].

The decomposition of the exponential B-spline based signal ($n = 4$) requires the same amount of operations compared to the decomposition of the polynomial case. That is, for each sample point 4 real floating point additions and 3 multiplications are necessary. However, for the case of the reconstruction, an expansion with a factor $m > 2$ will require $4(m+1)$ complex additions and $4m+7$ complex multiplications per input sample point. This is more computationally expensive than the $4(m+1)$ real additions and 3 multiplications required for the polynomial case.

5. Simulations and Results

The first 1-D example is the interpolation of the discrete delta $\delta[\cdot]$ function. In other words, is the shape of the cardinal B-spline. Fig.2(a) shows that the impulse response of the 4th order ($n = 4$) cardinal exponential B-spline decays faster than the polynomial one. In terms of the frequency characteristics shown in fig.2(b), the exponential cardinal spline is less band-limited than the polynomial one, even though they have the same continuity C^2 .

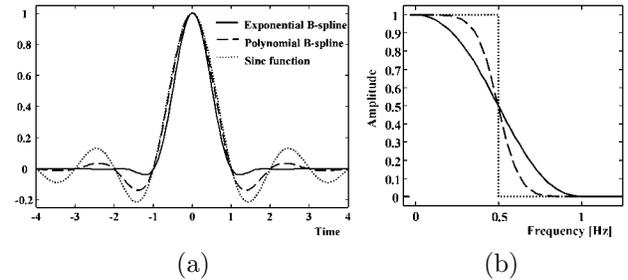


Figure 2. (a) Cardinal B-spline ($n = 4$) functions and (b) their frequency characteristics.

The second example consists of the interpolation of the discrete step function $u[k] = 1$ for $k \geq 0$ and 0 otherwise. From fig.3, the real part of the exponential B-spline results in less oscillations in comparison with the polynomial one when interpolating digital signals such as the discrete step function. This may be due

to the frequency characteristics of the cardinal spline in fig.2(b), where the exponential B-spline ($n=4$) is less band-limited than polynomial counterpart and also the ideal interpolator.

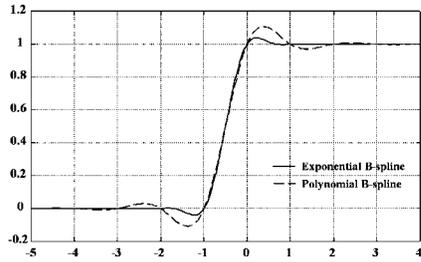


Figure 3. Interpolation of the discrete step function using exponential and polynomial B-splines functions ($n = 4$).

The third example is the expansion by a factor of 8, of an artificial image that consists of a digitalized text and can be seen in fig.4(a). Because images involve 2-D data, the tensor product of the real part of exponential B-spline was used as the base function. In the case of the expansion using the polynomial B-spline ($n = 4$), some ringing near the edges of the letters can be observed in fig.4(b). This ringing effect is diminished by using exponential B-splines, as can be seen in fig. 4(c).

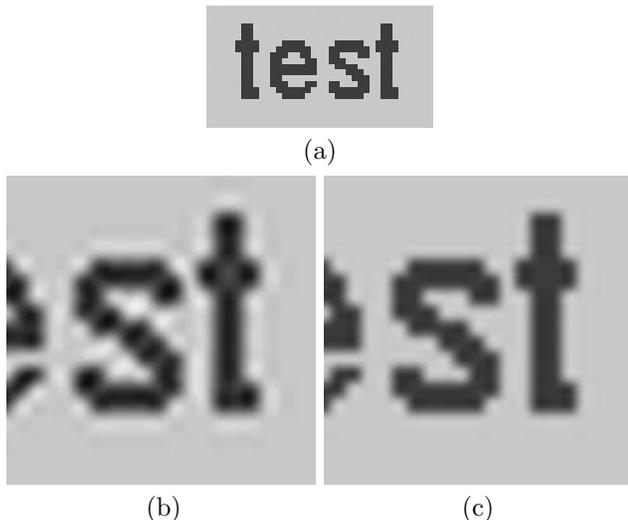


Figure 4. Expansion of an artificial image by polynomial and exponential tensor product B-splines, (a) original image (b) by polynomial (c) exponential.

The last example is the expansion by a factor of 8 from a section of the image of Barbara. This image consists of smooth areas corresponding to the face of Barbara, and higher frequency parts which are the background and part of the dress. In fig.5 it can be observed that the resulting interpolated images are almost the same at a glance, but more jaggedness appears in the high frequency areas in the exponential B-spline case due to its less band-limited characteristics.

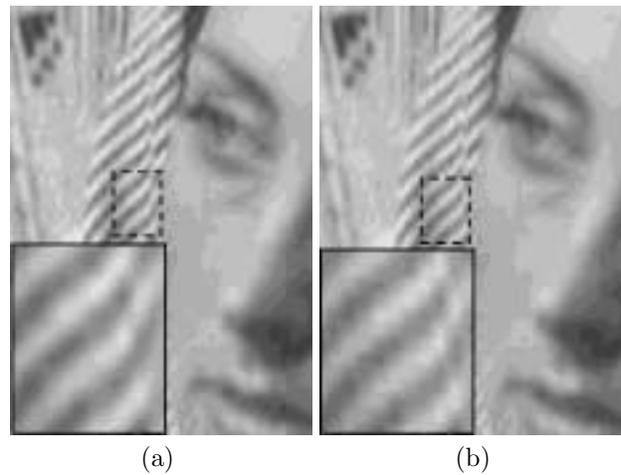


Figure 5. Expansion of Barbara image by (a) polynomial and (b) exponential tensor product B-splines.

6. Conclusions

In this work, a fast calculation method for exponential B-splines was proposed (where the calculation of B-splines is a particular case). When complex parameters are used in the exponential functions, trigonometric splines can be obtained. Based on some simulation results, less band-limited functions can be achieved by using the real part of this later functions and comparing them with the polynomial B-spline counterparts. Although more amount of computation is required, it is specially useful for images containing high frequency components.

Because the parameters $\tilde{\mu}_n$ of the exponentials are variable, these values can be selected to obtain B-splines with different space/frequency characteristics. For the implementation in different classes of images, the optimum parameters are yet to be determined in a future research.

References

- [1] M. Unser, "Splines: A perfect fit for signal and image processing," *IEEE Signal Processing Magazine*, vol.16, no.6, pp. 22-38, November 1999.
- [2] W. Dahmen and C.A. Micchelli, "On theory and application of Exponential Splines," in *Topics in Multivariate Approximation*, C.K. Chui, L.L. Schumaker and F.I. Utreras (eds.), Academic Press, New York, 1987, pp.37-46.
- [3] A. Ron, "Exponential Box Splines," *Constructive Approximation*, no.4, pp.357-378, 1998.
- [4] L. Ferrari, J. Park, A. Healey, and S. Leeman, "Interpolation using a fast spline transform (FST)," *IEEE Trans. on Circuits and Systems I*, vol.46, no.8, pp.891-906, Aug. 1999.
- [5] M. Unser, A. Aldroubi and M. Eden, "Fast B-Spline transforms for continuous image representation and interpolation," *IEEE Trans. on Pattern Anal. & Machine Intell.*, vol.13, no.3, pp.277- 285, March 1991.