A Kernel Representation for Exponential Splines with Global Tension

Sven Barendt[a] and Bernd Fischer[a] and Jan Modersitzki[b]

[a]Institute of Mathematics, University of Luebeck, Luebeck (Germany);
[b]Department of Computing and Software, McMaster University, Hamilton, ON (Canada)

ABSTRACT

Interpolation is a key ingredient in many imaging routines. In this note, we present a thorough evaluation of an interpolation method based on exponential splines in tension. They are based on so-called tension parameters, which allow for a tuning of their properties. As it turns out, these interpolants have very many nice features, which are, however, not born out in the literature. We intend to close this gap. We present for the first time an *analytic representation* of their *kernel* which enables one to come up with a space and frequency domain analysis. It is shown that the exponential splines in tension, as a function of the tension parameter, bridging the gap between linear and cubic B-Spline interpolation. For example, with a certain tension parameter, one is able to suppress ringing artefacts in the interpolant. On the other hand, the analysis in the frequency domain shows that one derives a superior signal reconstruction quality as known from the cubic B-Spline interpolation, which, however, suffers from ringing artifacts. With the ability to offer a trade-off between opposing features of interpolation methods we advocate the use of the exponential spline in tension from a practical point of view and use the new kernel representation to qualify the trade-off.

Keywords: Interpolation, Exponential Splines, Tension, Kernel Representation, Ringing, Aliasing, Signal Reconstruction

1. INTRODUCTION

Interpolation routines often constitute a mayor ingredient in imaging applications. Typically, the goal is to embed given discrete data into a continuous setting. The choice of the specific interpolation scheme is highly application dependent and encompasses fundamental properties like signal reconstruction, approximation order, differentiability and computational complexity. The present work deals with a specific interpolation class, the so-called *exponential splines in tension*.[1] This class enables one to tune the interpolants towards the wanted properties with the help of so-called *tension parameter*. In a 1D setting this interpolation scheme ranges inbetween two outstanding interpolation approaches: the simple linear and the cubic B-Spline interpolation. To see this, note that the exponential spline in tension of order four has to fulfill the differential equation

$$(D^4 - \tau_i D^2)s = 0 \tag{1}$$

in each interval I_i, which are based upon predefined knots. Here, D denotes the differentiation operator and $\tau_i \geq 0$ the tension parameter. For $\tau_i \to 0$ the term $D^4 s$ dominates and consequently the solution of (1) is nearly a cubic polynomial. On the other hand, if $\tau_i \to \infty$ the term $D^2 s$ leads to a close approximation of a linear polynomial. We refer to Ref. 2 for a detailed discussion.

To set up the notation, the thought after interpolant t may be phrased in terms of the basis functions $B_i(u)$ via

$$t(u) = \sum_{i=0}^{n-1} c_i B_i(u). \tag{2}$$

In Section 2 the linear and the cubic B-Spline basis functions and their kernel representations are defined and discussed in terms of the interpolant's inherited features. Later, in Section 3, the exponential spline in tension basis functions, as presented by Koch and Lyche,[1] are shown. Based on their definition, we give an analytic derivation of a kernel representation of an exponential spline in *global* tension. Finally in Section 4 this kernel representation is used to examine and compare the exponential spline's features under different tensions to the one by linear and cubic B-Spline interpolants, respectively.
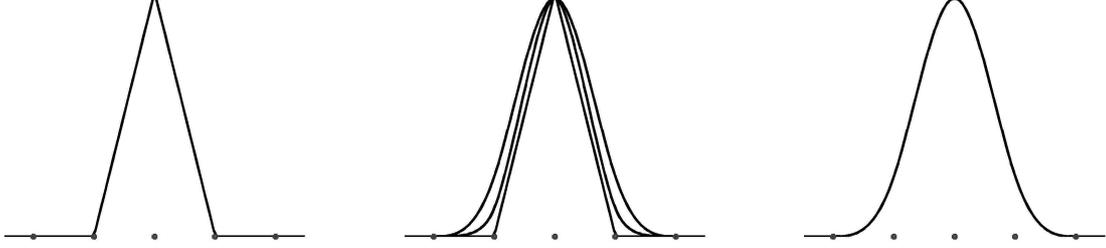
Figure 1. Linear, exponential spline with different tension $(7 \cdot 10^2, 8, 10^{-4})$ and cubic B-Spline 1D basis functions (from left to right). To have consistent plots the basis functions are scaled independently.

2. LINEAR AND CUBIC B-SPLINE INTERPOLATION

In the linear case, the basis functions are given by the hat functions

$$B^{Lin}(u) = \begin{cases} 1 - |u| & , |u| < 1 \\ 0 & , |u| \geq 1, \end{cases} \tag{3}$$

that is $B_i(u) = B^{Lin}(u - i - 1)$. For this special case the coefficients c_i in the basis representation are simply the data points. For cubic B-Spline interpolation, the basis functions in (2) are based on the following piecewise cubic polynomial

$$B^{Cub}(u) = \begin{cases} u^3 & , u \in [0,1), \\ 4 - 12u + 12u^2 - 3u^3 & , u \in [1,2), \\ -44 + 60u - 24u^2 + 3u^3 & , u \in [2,3), \\ 64 - 48u + 12u^2 - u^3 & , u \in [3,4), \\ 0 & , \text{otherwise}, \end{cases} \tag{4}$$

where $B_i(u) = B^{Cub}(u - i - 2)$. In Figure 1, a linear basis function (left) and a cubic B-Spline basis function (right) is shown. In addition, some exponential splines with respect to different tensions are depicted, which will be discussed later. One quality measure for interpolation schemes is their signal reconstruction capability,[3] i.e., the ability to match the underlying function as exact as possible. Here, the cubic B-Spline interpolation scheme is among the top performing methods. This may be shown by interpreting (2) as a convolution

$$t(u) = \sum_{i=0}^{n-1} c_i B^{Cubic}(u - i - 2) = c * B^{Cubic}(u - 2). \tag{5}$$

A reformulation of the coefficient vector c and a little analysis uncovers the cubic B-Spline interpolation kernel

$$K^{Cubic}(u) = B^{Cubic}(u - 2) * \sum_{m=-\infty}^{\infty} \sqrt{3}(\sqrt{3} - 2)^{|m|} \delta(u + m), \tag{6}$$

see Ref. 3 for details. This kernel and its single sided magnitude spectrum is depicted in Figure 2. As it is apparent, the kernel is very similar to the $\mathrm{sinc}(u) = \frac{\sin(\pi u)}{\pi u}$ function and therefore close to the ideal interpolation kernel. The same observation is true for the magnitude spectrum, as it nicely resembles the rect function

$$\mathrm{rect}(u) = \begin{cases} 1 & , |u| < \frac{1}{2} \\ 0 & , |u| \geq \frac{1}{2} \end{cases}, \tag{7}$$

which is the magnitude spectrum of the sinc function. Furthermore, the sidelobes in the stopband and the magnitude at the cutoff point are very low for the cubic B-Spline kernel magnitude spectrum.

On top, as it is well-known (see Ref. 3), the cubic B-Spline may be evaluated faster than any other method with a similar signal reconstruction quality. The linear interpolation method, on the other hand, is less exact. Its kernel $K^{Lin}(u) = B^{Lin}(u)$ and its magnitude spectrum is shown in Figure 3. Here, one observes a
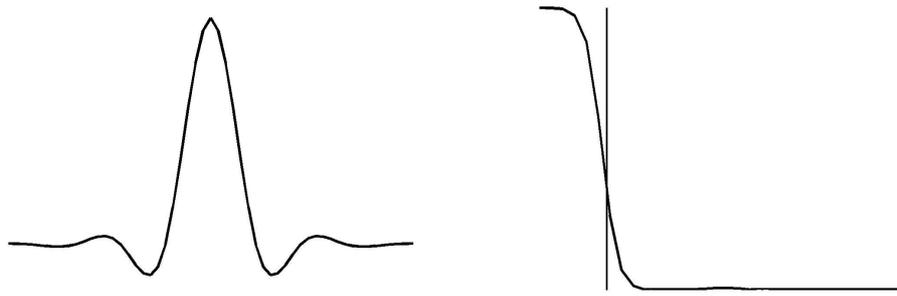
Figure 2. 1D cubic B-Spline kernel representation ($|u| < 5$) and its single sided magnitude spectrum (linear scaling). The vertical line indicates the cutoff point, that is the area left to the vertical line is the passband; the area right to the vertical line is the stopband.
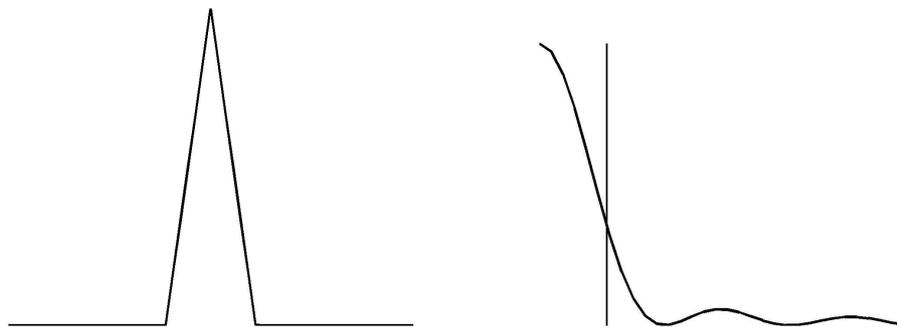

Figure 3. 1D linear Kernel representation ($|u| < 5$) and its single sided magnitude spectrum (linear scaling). The vertical line indicates the cutoff point.

significant higher amplitude of the sidelobes, resulting in more aliasing in the reconstructed signal. In addition higher frequencies in the passband are more absorbed than in the B-Spline Kernel case. This leads to a inferior reproduction of higher frequencies. However, due to its simplicity, the linear interpolation is one of the fastest interpolation methods. The price paid for this, is the fact, that the interpolant is not differentiable at the interpolation points, which can be a drawback, in particular in conjunction with the employment of fast optimization schemes. A feature of the linear interpolation method which has often been neglected is the fact, that its interpolant is free of ringing. This is not a true artefact and almost every interpolation method with a more exact signal reconstruction may produce ringing.[4] Figure 4 (left) shows ringing occuring in the cubic B-Spline interpolant. Note that the under- and overshooting can produce pixel values that are beyond the reasonable range of values. As seen in Figure 4 (right), no ringing is observable for the same data for the linear interpolation scheme.

In Table 1 the outlined features of linear versus cubic B-Spline interpolation are listed.

	Linear	Cubic B-Spline
ringing	inexisting	existing
reconstruction	less exact	more exact
differentialbility	not at data points	twice continuous differentiable

Table 1. Inherited features of linear versus cubic B-Spline interpolation schemes

Figure 4. The interpolated data values (left) and the cubic B-Spline interpolant (middle) as well as the linear interpolant (right) is shown. Note the ringing in the B-Spline interpolant.

Ideally, one is looking for an interpolant, which is easy to compute, exhibits no ringing artefacts, is differentiable and does nicely reconstruct the given function. Neither a linear nor a cubic B-Spline interpolant does fulfill all these features. On the other hand, as mentioned above, the tension parameter associated with the exponential spline in tension may be used to tune these basis functions towards a linear interpolant or to a cubic B-Spline interpolant or to any situation inbetween. This enables the user to choose "the right dose" of features that are inherited in interpolation schemes by simply changing one parameter. The exponential spline in tension will be closely examined in the next section.

3. EXPONENTIAL SPLINE IN TENSION

3.1 B-Spline representation

As mentioned in the introduction, the general interpolation setting (2) is used for the locally supported exponential B-Spline basis introduced by Koch and Lyche.[1] They defined the basis in question recursively for an arbitrary order $k > 2$ by successive integration:

$$B_{i,k} = \Phi_{i,k-1} - \Phi_{i+1,k-1}, \tag{8}$$

with

$$\Phi_{j,k-1}(u) = \begin{cases} 0, & \text{if } u < t_j \\ \frac{\int_{t_j}^{u} B_{j,k-1}(v)dv}{\sigma_{j,k-1}}, & \text{if } t_j \leq u < t_{j+k-1} \\ 1, & \text{otherwise,} \end{cases} \tag{9}$$

and

$$\sigma_{j,k-1} = \int_{t_j}^{t_{j+k-1}} B_{j,k-1}(v)dv \tag{10}$$

Here, the $t_j, j = 0, 1, ..., m-1$, denote an increasing knot sequence. For our interpolation purposes we assume that the t_j are equidistantly distributed with $t_j - t_{j-1} = 1$. Moreover we use a global tension parameter τ to keep the issues of interest clear. Furthermore, this choice allows for a straightforward calculation of a basis in higher space dimensions by simply envoking tensor products. The recursion starts with

$$B_{i,2}(u) = \begin{cases} \frac{\sinh(\tau(x-t_i))}{\sinh(\tau)} & \text{if } t_i \leq u < t_{i+1}, \\ \frac{\sinh(\tau(t_{i+2}-u))}{\sinh(\tau)} & \text{if } t_{i+1} \leq u < t_{i+2}, \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

Thus, after successive integration, an explicit form for a basis of order four (assuming four nontrivial intervals)

looks like

$$B_{0,4}^\tau(u) = \begin{cases} \varphi_1^\tau(u) & , u \in [0,1) \\ \varphi_2^\tau(u) & , u \in [1,2) \\ \varphi_2^\tau(-u+4) & , u \in [2,3) \\ \varphi_1^\tau(-u+4) & , u \in [3,4) \\ 0 & , \text{otherwise,} \end{cases} \tag{12}$$

with

$$\begin{aligned} \varphi_1^\tau(u) &= (\sin(\tau x) - x)\left(\frac{1}{(2\cosh(\tau)-2)\tau} - \frac{1}{2\sinh(\tau)+\tau+1}\right) \\ \varphi_2^\tau(u) &= (2\cosh(\tau)(u-1)\tau + \sinh(\tau(2-x)) - 2\sinh(\tau(x-1)) - x) \\ &\quad \left(\frac{1}{(2\cosh(\tau)-2)\tau} - \frac{1}{2\sinh(\tau)+\tau+1}\right) \end{aligned} \tag{13}$$

As a result, we can now use these basis functions in the general interpolation problem (2) with the setting $B_i(u) := B_{0,4}^\tau(u-i-2)$.

3.2 Kernel representation

To derive a kernel representation we first interpret (2) as a convolution

$$t(u) = \sum_{i=0}^{n-1} c_i B_{0,4}^\tau(u-i-2) = c * B_{0,4}^\tau(u-2). \tag{14}$$

To reformulate c_i, we evaluate (14) at the interpolation points

$$t(j) = \sum_{i=0}^{n-1} c_i B_i(j) = \sum_{i=j-2}^{j+2} c_i B_i(j), \tag{15}$$

and observe

$$t(j) = c(j) * \frac{1}{2b_{-1}^\tau + b_0^\tau}(b_{-1}^\tau \delta(j-1) + b_0^\tau \delta(j) + b_{-1}^\tau \delta(j+1)), \tag{16}$$

which follows from $B_{0,4}^\tau(1) = b_{-1}^\tau = B_{0,4}^\tau(3)$, $B_{0,4}^\tau(2) = b_0^\tau$. A Fourier transformation of (16) leads to

$$\hat{t}(f) = \hat{c}(f)\frac{1}{2b_{-1}^\tau + b_0^\tau}(b_0^\tau + 2b_{-1}^\tau \cos(2\pi f)). \tag{17}$$

A reformulation of (17) and a Fourier sum representation is used to express $\hat{c}(f)$ as

$$\hat{c}(f) = \hat{t}(f)\left(\frac{-2\log(-\sqrt{-1+2b_2}) + \log(-1+2b_2)}{2\sqrt{-1+2b_2}\pi} + \sum_{m=1}^\infty d_m \cos(2\pi m f)\right) \tag{18}$$

with the coefficients

$$d_m = \frac{2\sqrt{-1+2b_2}(a_1^n - a_2^n)\pi - 2\sqrt{b_1^2 - b_2^2}(a_1^n + a_2^n)\log(-\sqrt{-1+2b_2}) + \sqrt{b_1^2 - b_2^2}(a_1^n + a_2^n)\log(-1+2b_2)}{2\sqrt{-1+2b_2}\sqrt{b_1^2 + b_2^2}\pi} \tag{19}$$

and

$$a_1 = \frac{-b_1 + \sqrt{b_1^2 - b_2^2}}{b_2}, a_2 = -\frac{b_1 + \sqrt{b_1^2 - b_2^2}}{b_2} \tag{20}$$

and the constants $b_1 = \frac{b_0^\tau}{2b_{-1}^\tau + b_0^\tau}$ and $b_2 = \frac{2b_{-1}^\tau}{2b_{-1}^\tau + b_0^\tau}$. With an inverse Fourier transform it follows that

$$c(j) = t(j) * \left(\frac{-2\log(-\sqrt{-1+2b_2}) + \log(-1+2b_2)}{2\sqrt{-1+2b_2}\pi}\delta(j) + \sum_{m=1}^\infty d_m(\delta(j-m) + \delta(j+m))\right). \tag{21}$$
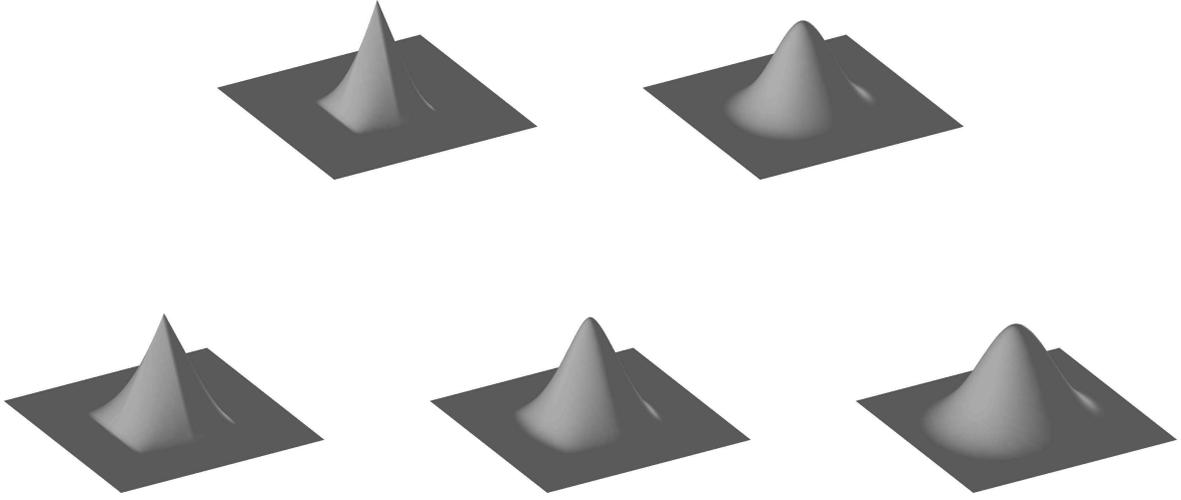
Figure 5. Upper row: Linear basis function and cubic B-Spline basis function; lower row: Exponential Spline with different *global* tension $(10^{-4}, 8, 7 \cdot 10^2)$.

Finally, by taking (14) into account, we obtain an explicit expression for the interpolation kernel which is summarized in the following theorem.

THEOREM 3.1. (Kernel Representation of exponential spline in global tension) *Let $u \in \mathbb{R}$ and $B_{0,4}^\tau$ the exponential B-Spline in global tension $\tau \in \mathbb{R}^{>0}$ (12). Furthermore let the constants $b_{-1}^\tau = B_{0,4}^\tau(1)$ and $b_0^\tau = B_{0,4}^\tau(2)$ and $b_1 = \frac{b_0^\tau}{2b_{-1}^\tau + b_0^\tau}$, $b_2 = \frac{2b_{-1}^\tau}{2b_{-1}^\tau + b_0^\tau}$ be given. Then the kernel representation of an exponential spline in global tension may be written as*

$$K^\tau(u) = B_{0,4}^\tau(u-2) * \left(\frac{-2\log(-\sqrt{-1+2b_2}) + \log(-1+2b_2)}{2\sqrt{-1+2b_2}\pi} \delta(j) + \sum_{m=1}^{\infty} d_m(\delta(j-m) + \delta(j+m)) \right) \quad (22)$$

with

$$d_m = \frac{2\sqrt{-1+2b_2}(a_1^n - a_2^n)\pi - 2\sqrt{b_1^2 - b_2^2}(a_1^n + a_2^n)\log(-\sqrt{-1+2b_2}) + \sqrt{b_1^2 - b_2^2}(a_1^n + a_2^n)\log(-1+2b_2)}{2\sqrt{-1+2b_2}\sqrt{b_1^2 + b_2^2}\pi} \quad (23)$$

and

$$a_1 = \frac{-b_1 + \sqrt{b_1^2 - b_2^2}}{b_2}, a_2 = -\frac{b_1 + \sqrt{b_1^2 - b_2^2}}{b_2} \quad (24)$$

 Thus the kernel of an exponential B-Spline in global tension can be described as a convolution with its B-Spline basis. In conjunction with the cubic B-Spline kernel in (6), this kernel exhibits an infinite support. Consequently, for an interpolation based on an exponential spline in tension an infinite support kernel has to be taken into account. This is a mayor advantage as opposed to interpolation methods based on a truncated sinc function, for example.

4. RESULTS

In Section 2 we presented some results for the linear and cubic B-Spline interpolation method based on their kernel representation. In particular the respective magnitude spectrum has been used to investigate the signal reconstruction quality by analyzing the passband and stopband characteristics. Based on the novel kernel
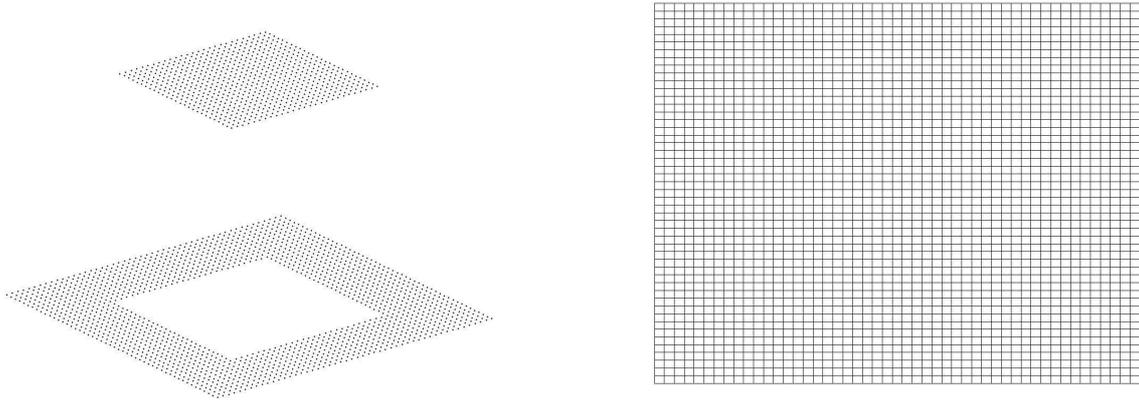
Figure 6. The data points (left) and the grid where the points are located (right) is shown.

representation for the exponential spline in global tension, we are now in a position to compare it with the linear and cubic B-Spline kernel in order to quantify its properties. Furthermore a simple example is given which shows the different interpolation methods on duty. here the interpolation data as shown in Figure 6 (left) is used. The 50^2 data points are located at the equidistant distributed grid, shown in Figure 6 (right). As we have a two-dimensional grid, we need two dimensional basis functions. These can be easily derived from the 1D basis functions by a straightforward tensor approach (see Appendix A for details). The two-dimensional basis functions for the linear, cubic B-Spline and exponential spline in global tension are shown in Figure 5.

The interpolants corresponding to the aforementioned data points of the three interpolation methods are shown in Figure 7 (first column). One may see that with the linear interpolation method no ringing occured in the interpolant in opposition to the cubic B-Spline method. The exponential Spline in global tension interpolant reduces the ringing significant by increasing the tension. However, ringing close to the cubic B-Spline case is observable for low tension. Note the similarity of the exponential spline kernel to the linear kernel (resp. cubic B-Spline kernel) for high tension (resp. low tension) in the third column of Figure 7.

On the other hand, if we have a look at the magnitude spectrum of the different kernel representations (fourth column of Figure 7), the linear interpolation method's attenuation of high frequencies in the passband is observable, as previously discussed. In addition we discussed the magnitude spectrum of the cubic B-Spline interpolation which resembles nicely the rect-function. Again we can observe a trade-off in the passband characteristics in the magnitude spectrum of the exponential spline's kernel in different tension situation. The same is true for the stopband characteristics: The sidelobes in the stopband are high and close to the linear case for high tension and hardly existing for a low tension and therefore close to the cubic B-Spline case.

The effect of an attenuation of high frequencies in the passband and high sidelobes can be seen in the second column of figure 7. Here the interpolant is successivly evaluated at a rotated grid as shown in Figure 6 (right). it has been rotated 30 times with an angle of $2\pi/30$. Thus, after 30 rotations, the interpolation result should be in theory the start configuration. Here, one observes a significant smoothed interpolant for the linear interpolation method. In contrast the cubic B-Spline's interpolant is less smoothed and the exponential spline's interpolant again offers a nice trade-off.

Table 2 underscores this statement by presenting the relative Euclidian distance between the data presented in Figure 6 (left) and the interpolant after 30 interpolations evaluated at the grid shown in Figure 6 (right). As a reference, the distance in the linear case is set to 100%.

It is worth noticing, that the exponential spline is, independent of the tension parameter, two times continuous differentiable in contrast to the linear case (see Ref.[5]). So, if one suppresses the ringing further by choosing a
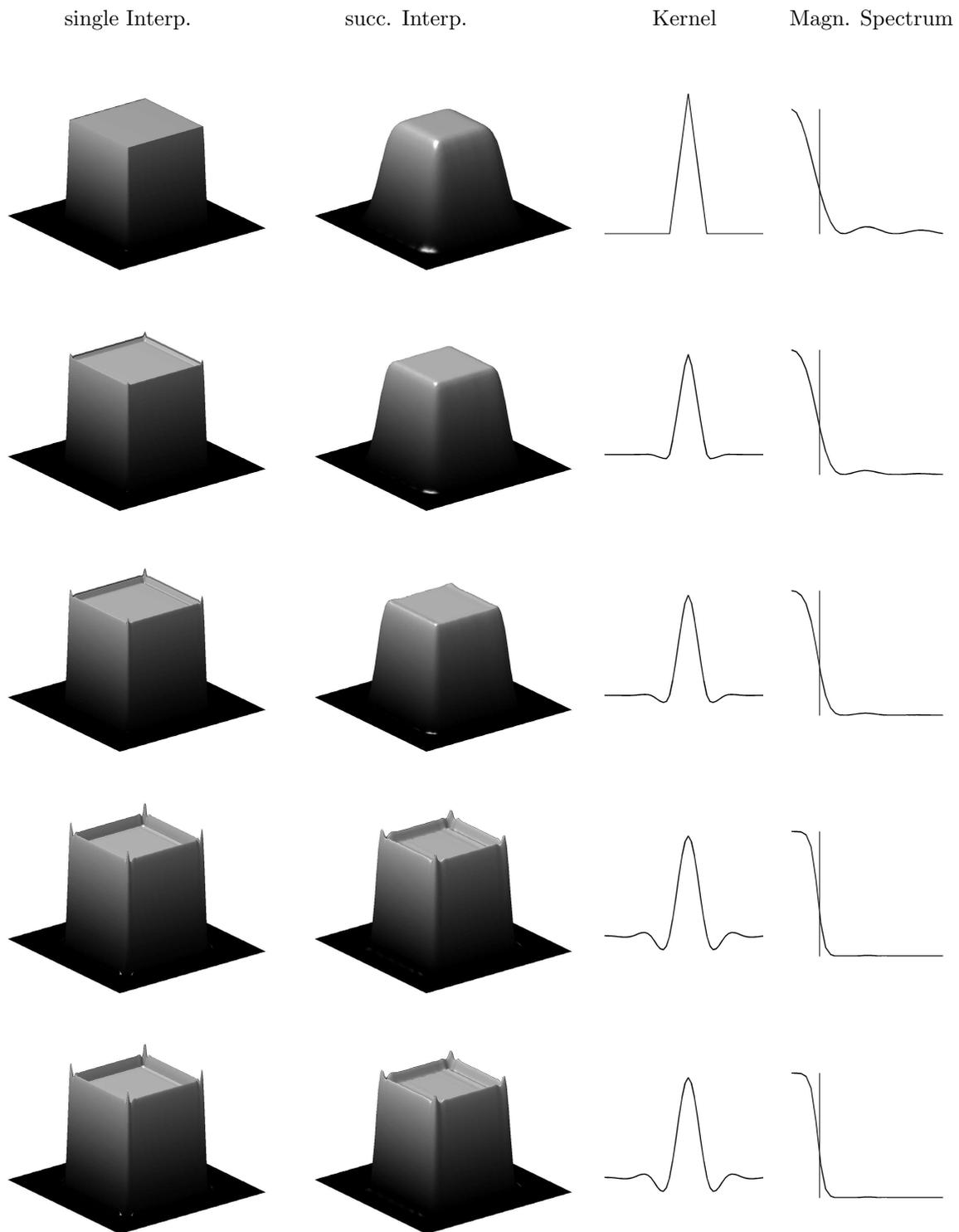
Figure 7. The plot shows the different interpolation methods for one single interpolation, 30 successive interpolations as well as the kernel representation and its single sided magnitude spectrums (column 1-4). The first row shows the linear interpolation method. The second to forth row show the exponential spline in different tension (row 2: $\tau = 10$, row 3: $\tau = 5$, row 4: $\tau = 0.1$). The last row is associated with the cubic B-Spline. Again, in the plots of the magnitude spectrums the vertical bar denotes the cutoff point, that is, the area to the left indicates the passband and the area to the right indicates the stopband.

Interpolation Method	Euclidian Distance (Linear interp. set to 100 %)
Linear	100%
Exp. Spline ($\tau = 10$)	71.79%
Exp. Spline ($\tau = 5$)	51.64%
Exp. Spline ($\tau = 0.1$)	20.16%
Cubic Spline	20.13%

Table 2. This table shows the Euclidean distance between the data shown in Figure 6 exemplarily and the interpolant after 30 successive interpolations (second column in Figure 7). The data is denoted here with $a_{i,j}$, $i, j = 1, ..., 50$. If I denotes the current interpolant and $x_{i,j}$ the two dimensional location of $a_{i,j}$, then the Euclidean distance is $E = \sqrt{\sum_{i=1}^{50} \sum_{j=1}^{50} (a_{i,j} - I(x_{i,j}))^2}$. The Euclidean distance for the linear case is set to 100%.

higher tension τ the exponential spline behaves more and more like the linear interpolant and has less pronounced features than a cubic B-Spline interpolant and vice versa.

5. DISCUSSION

The outlined analysis shows, from a practical point of view, that the exponential splines in tension bridge the gap between two opposing interpolation schemes: The linear and the cubic B-Spline interpolation. Both have features that are more or less desired, depending on the application. We derive an analytical kernel representation of the exponential spline and show that the tension parameter enables the user to determine the best trade-off for a given application with respect to signal reconstruction versus ringing. Nevertheless we still have a two times differentiable interpolant which can be useful for fast optimization schemes.

As a conclusion, a variation of tension in the exponential spline in global tension results in a specific accentuation of features depending on the application. An analysis of the passband and stopband characteristics of the magnitude spectrum of the exponential spline's kernel can be used to qualify the accentuation.

APPENDIX A. TENSOR APPROACH

With the linear, the cubic B-Spline, and exponential B-Spline in tension one obtains an enhancement to higher dimensional interpolation by using a tensor product approach as follows (cf. (2)):

$$t(u) = \sum_{i_d=0}^{n_d-1} \cdots \sum_{i_1=0}^{n_1-1} c_{i_1,...,i_d} B_{i_1}(u^1) \cdots B_{i_d}(u^d), \tag{25}$$

with u^i as the evaluation point in the i-th dimension. To calculate the coefficients $c_{i_1,...,i_d}$ for the cubic B-Spline and the exponential spline in tension we rewrite (25) and obtain for instance for $d = 2$

$$T(i_1, i_2) = \sum_{k_2=0}^{n_2-1} \sum_{k_1=0}^{n_1-1} c_{k_1,k_2} B_{k_1}(t_{i_1}^1) B_{k_2}(t_{i_2}^2), \tag{26}$$

with $t_{i_1}^1$ and $t_{i_2}^1$ as the knot sequences in the first and second dimension respectively. The corresponding interpolated value is denoted by $T(i_1, i_2)$. By introducing the matrix $A_{n_j} = [B_k(t_i^j)]_{k,i=0}^{n_j-1}$, equation (26) can be written as

$$T(i_1, i_2) = \sum_{k_2=0}^{n_2-1} \sum_{k_1=0}^{n_1-1} c_{k_1,k_2} A_{n_1}(k_1, i_1) A_{n_2}(k_2, i_2). \tag{27}$$

This can again be reformulated with the linearized orderings $r = i_1 + i_2 n_1$, $s = k_1 + k_2 n_1$ and the Kronecker product $A_n = A_{n_2} \otimes A_{n_1}$ with $n = (n_1 - 1)(n_2 - 1)$:

$$T(r) = \sum_{s=0}^{n} A_n(r, s) c_s. \tag{28}$$

Thus, the coefficients can be computed by

$$c = A_{n_1}^{-1} T A_{n_2}^{-1}. \tag{29}$$

Next, we make use of the underlying spline basis. In order to use the local support of the basis one only needs to evaluate the spline (cf. (25)) on a few basis functions once the coefficients are computed. If we want to evaluate the spline for an $(u^1, u^2) \in [t_{l_1}, t_{l_1+1}] \times [t_{l_2}, t_{l_2+1}]$, $l_1, l_2 \in \mathbf{Z}$, we only have to evaluate the sums

$$t(u) = \sum_{i_2=l_2-2}^{l_2+2} \sum_{i_1=l_1-2}^{l_1+2} c_{i_1,i_2} B_{i_1}(u^1) B_{i_2}(u^2), \tag{30}$$

by assuming that the interpolation values $T(i_1, i_2)$ are zero for $i_1 \notin [0, n_1]$ and $i_2 \notin [0, n_2]$ and therefore c_{i_1,i_2} equals zero. This leads to a fast evaluation of the spline interpolant.

In Figure 5 the basis functions (linear, cubic B-Spline, exponential spline in tension) are shown for d=2. For all 2D representations the tensor approach is used with the aforementioned 1D basis functions (cf. (3), (4), (12)).

REFERENCES

[1] Koch, P. E. and Lyche, T., "Exponential b-splines in tension," *Approximation Theory VI*, 361–364 (1989).

[2] Prenter, P. M., [*Splines & Variational Methods*], John Wiley & Sons (1989).

[3] Lehmann, T. M., Gönner, C., and Spitzer, K., "Survey: Interpolation methods in medical image processing," *IEEE Transactions on Medical Imaging* **18**(11), 1049–1075 (1999).

[4] Thévenaz, P., Blu, T., and Unser, M., [*Image Interpolation and Resampling*], 393–420, Academic Press, Inc., Orlando, FL, USA (2000).

[5] Koch, P. E. and Lyche, T., "Construction of exponential tension b-splines of arbitrary order," *Curves and Surfaces, P.-J. Laurent, A. Le Méhauté, and L. L. Schumaker (eds.), Academic Press, New York*, 255–258 (1991).