A Family of 4-Point Dyadic Multistep Subdivision Schemes

Daniel Lemire

Abstract. We present a new family of multistep iterative interpolation schemes and a 4-point high resolution scheme reproducing quartic polynomials. Interpolation requires two steps: a coarse scale interpolation followed by a fine scale interpolation. The interpolants are C^1, have good local properties, and no additional memory requirement.

§1. Introduction

Subdivision schemes interpolate a discrete set of data points in a local manner, that is, the value of the interpolation function at a given point depends on a small number of nearby data points. The classical dyadic subdivision scheme [6,3] finds the midpoint values by fitting a Lagrange polynomial through the $2N$ closest data points. By repeating this algorithm iteratively, we have a dense set of data points and determine uniquely a smooth interpolation function. Because subdivision schemes relate data points from one scale to another, it is not surprising that they are a key ingredient in the construction of compactly supported wavelets [1,4].

Often, interpolation schemes can be made more local by using more memory. Merrien [15,16,7] introduced Hermite subdivision schemes which have twice the approximation order and better regularity for a given support, and vector subdivision schemes in general have received a lot of attention ever since [11,17,13]. In spline theory, adding extra nodes can make spline interpolation local [5]. However, in this paper, instead of using more memory, we want to make better use of the memory we already have. We propose to use the upcoming nodes at least one step earlier to record coarse scale guesses (see Fig. 1). Because the new placeholders are used

Curve and Surface Fitting: Saint-Malo 2002
XXX (eds.), pp. 1–10.

ISBN XXX.

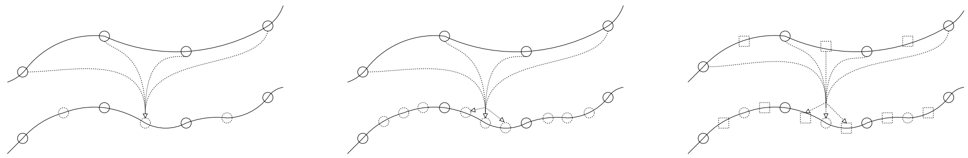

Fig. 1. Diagrams of 4-point subdivision schemes in the dyadic (left) and 4-adic (center) cases, and our multistep scheme (right). Arrows symbolize the interpolation process. The circles are data samples, and the squares represent placeholders recording "guesses" .

scheme	regularity	support	polynomials
Dubuc	C^1	$[-3,3]$	cubic
Dyn-Gregory-Levin	up to C^1	$[-3,3]$	up to cubic
Hassan et al.	C^2	$[-5/2,5/2]$	quadratic
presented multistep	C^1	$[-3,3]$ or $[-3,4]$	cubic or quartic

Tab. 1. Comparison between some 4-point iterative interpolation schemes. The support of the fundamental functions is given. The quartic multistep scheme is slightly less local because it requires initialization by a one-step 5-point scheme.

as predictors, the new schemes will be as local as the usual subdivision schemes.

The main result of this paper is that by summing up the 4-adic interpolation and dyadic interpolation, we get a range of smooth (C^1) multistep schemes reproducing at least cubic but also quartic polynomials (see Tab. 1). While there exists 5-point quartic subdivision schemes, they are not as local as our 4-point multistep scheme.

The paper is organized as follows. We begin with a brief review of subdivision schemes, and discuss both the dyadic and 4-adic 4-point Deslauriers-Dubuc schemes. Combining these subdivision schemes, we present a family of multistep schemes reproducing cubic polynomials and prove that some of these schemes are smooth (C^1).

§2. Subdivision Schemes

Given an integer $b > 1$, a b-adic number is of the form $x_{j,k} = k/b^j$ for some integers k, j. For a fixed J, given some data $\{y_{J,k}\}_{k\in\mathbb{Z}}$, we want a smooth function f such that $f(x_{J,k}) = y_{J,k}$, for every $k \in \mathbb{Z}$. Starting with $\{y_{J,k}\}_{k\in\mathbb{Z}}$ and using the formula

$$y_{j+1,l} = \sum_{k\in\mathbb{Z}} \gamma_{bk-l} y_{j,k}, \tag{1}$$

for some array γ, we get values $y_{j,k}$ for any $j > J$. Since b-adic numbers form a dense set in $\mathbb{R}$, there is at most one continuous function such that $f(x_{j,k}) = y_{j,k}$ for all $k \in \mathbb{Z}, j > J$.

A subdivision scheme is interpolatory and satisfies $f(x_{J,k}) = y_{J,k}$ if $\gamma_{bk} = 0$, for every $k \in \mathbb{Z}$ except for $\gamma_0 = 1$. We say that a subdivision scheme is stationary if the array γ is constant (does not depend on j). Because γ does not depend explicitly on l but rather on $bk-l$, the scheme is translation invariant or homogeneous. A subdivision scheme is a $2N-$point scheme if $\gamma_l = 0$ for $|l| \geq Nb$. The fundamental function of an interpolatory $2N-$point b-adic scheme has initial data $y_{0,l} = \delta_{l,0}$; it has a compact support of $[-(Nb-1)/(b-1), (Nb-1)/(b-1)]$ (or $[1-2N, 2N-1]$ when $b = 2$). For general stationary schemes, we define the k^{th} fundamental function as the interpolant with initial data $y_{0,l} = \delta_{l,k}$.

For $N = 1, 2, 3, ...$ there are corresponding interpolatory $2N-$point Deslauriers-Dubuc subdivision schemes (DD) built from the midpoint evaluation of Lagrange polynomials of degree $2N-1$. For $b = 2$ (dyadic case), the 4-point Deslauriers-Dubuc scheme can be defined with the array γ^{DD2} given by $\gamma_0^{DD2} = 1, \gamma_{\pm1}^{DD2} = -9/16, \gamma_{\pm3}^{DD2} = -1/16$ with $\gamma_k^{DD2} = 0$ otherwise; and for $b = 4$ (4-adic case), the scheme is defined with the array γ^{DD4} given by $\gamma_{2k}^{DD4} = \gamma_k^{DD2} \, \forall k \in \mathbb{Z}$, $\gamma_{\pm1}^{DD4} = 105/128, \gamma_{\pm3}^{DD4} = 35/128$, $\gamma_{\pm5}^{DD4} = -7/128, \gamma_{\pm7}^{DD4} = -5/128$, with $\gamma_k^{DD2} = 0$ otherwise.

Because 4-point DD schemes are derived from cubic Lagrange polynomials, they reproduce cubic polynomials, that is, if the initial data $y_{j,k}$ satisfies $y_{j,k} = p(x_{j,k})$, for every $k \in \mathbb{Z}$ for some cubic polynomial p, then the interpolation function f is this same cubic polynomial: $f = p$. The two cases presented above (γ^{DD2} and γ^{DD4}) converge to differentiable (C^1) interpolation functions [6,3].

§3. Multistep Subdivision Schemes

We define stationary multistep schemes by the equation

$$y_{j+1,l} = \sum_{m=1}^{M} \sum_{k \in \mathbb{Z}} \gamma_{Mbk+m-1-l}^{(m)} y_{j,Mk+m-1}, \tag{2}$$

where $\gamma^{(1)}, ..., \gamma^{(M)}$ are constant arrays (independent from j). They form a b-adic scheme because the number of nodes increases by a factor of b with each iteration. However, because we have $M > 1$ arrays γ, the scheme is said to be a multistep scheme. It is an interpolatory scheme if $y_{j+1,Mbk} = y_{j,Mk}$, and it is a $2N-$point scheme if $\gamma_l^{(m)} = 0$ for $|l| \geq MNb$ and $m = 1, ..., M$. In [2], a similar idea was proposed as the poly-scale framework, but it differs from our approach: Dekel and Dyn proved that within the poly-scale framework, convergent processes with initial data

$y_{0,k}$ converge to $\sum_k y_{0,k}\phi(x-k)$ for some continuous function ϕ, whereas no such thing is possible in general with multistep schemes.

For $b = M = 2$, the general equation (2) becomes

$$y_{j+1,l} = \sum_{k\in\mathbb{Z}} \gamma^{(1)}_{4k-l} y_{j,2k} + \gamma^{(2)}_{4k+1-l} y_{j,2k+1}. \tag{3}$$

A value $y_{j,k}$ is a stable value if $y_{j,k} = y_{j+1,2k}$; other nodes are said to be temporary or are referred to as placeholders. A multistep scheme on a dyadic grid is an interpolatory scheme if all $y_{j,2k}$ values on even nodes $(x_{j,2k})$ are stable so that $y_{j,2k} = y_{j+1,4k}$, for every $k \in \mathbb{Z}$.

For the rest of the paper, we will consider the schemes MS_α for some parameter $\alpha \in \mathbb{R}$, where $\gamma^{(1)}$ and $\gamma^{(2)}$ are chosen to be

$$\begin{aligned} \gamma^{(1)}_{2k} &= \gamma^{DD2}_{k}(1-\alpha) + \alpha\delta_{k,0}, \\ \gamma^{(1)}_{2k+1} &= \gamma^{DD4}_{2k+1}, \forall k \in \mathbb{Z}, \end{aligned} \tag{4}$$

and $\gamma^{(2)}_{-1} = \alpha$, $\gamma^{(2)}_{k} = 0$ for $k \neq -1$. Since $\gamma^{DD4}_{2k} = \gamma^{DD2}_{k}\, \forall k \in \mathbb{Z}$, we can rewrite equation (3) for even and odd terms. Firstly, setting $l = 2s$ (l even), we have

$$y_{j+1,2s} = \sum_{k\in\mathbb{Z}} \left((1-\alpha)\gamma^{DD2}_{2k-s} + \alpha\delta_{2k,s}\right) y_{j,2k} + \delta_{2k+1,s}\alpha y_{j,2k+1},$$

so that when s is even ($l = 2s = 4r$), we have the interpolatory condition

$$y_{j+1,4r} = y_{j,2r}, \tag{5}$$

otherwise, when s is odd ($l = 2s = 4r+2$)

$$y_{j+1,4r+2} = \alpha y_{j,2r+1} + (1-\alpha)\sum_{k\in\mathbb{Z}} \gamma^{DD2}_{2k-2r-1} y_{j,2k}. \tag{6}$$

Secondly, if l is odd ($l = 2s+1$), we have

$$y_{j+1,2s+1} = \sum_{k\in\mathbb{Z}} \gamma^{DD4}_{4k-2s-1} y_{j,2k}. \tag{7}$$

Equations (5), (6), and (7) can be used to describe MS_α: while equation (5) is the interpolatory condition, equation (7) fills the placeholders with 4-adic (coarse scale) interpolated values, whereas equation (6) combines the value stored in the placeholder with the newly available interpolated value (fine scale) given by the summation term which we recognize from the dyadic DD interpolation.

In the simplest case, $\alpha = 0 \Rightarrow \gamma^{(2)} = 0$, and equation (3) becomes $y_{j+1,l} = \sum_{k\in\mathbb{Z}} \gamma^{DD4}_{4k-l} y_{j,2k}$. In this last equation, $y_{j+1,l}$ depends only on even nodes $(y_{j,2k})$. Hence, we have $y_{j+1,2l} = \sum_{k\in\mathbb{Z}} \gamma^{DD4}_{4k-2l} y_{j,2k}$, but because $\gamma^{DD4}_{2k} = \gamma^{DD2}_{k}$, this last equation becomes $y_{j+1,2l} = \sum_{k\in\mathbb{Z}} \gamma^{DD2}_{2k-l} y_{j,2k}$, and if we define $\widetilde{y}_{j,k} = y_{j,2k}$ we have that MS_0 is equivalent to the 4-point dyadic DD subdivision scheme.

§4. Reproduced Polynomials

Assume that for some j, $y_{j,k} = p_3\left(x_{j,k}\right)$, for every $k \in \mathbb{Z}$, where p_3 is a cubic polynomial. Because 4-point DD schemes reproduce cubic polynomials, we have

$$\sum_{k\in\mathbb{Z}} \gamma^{DD2}_{2k-2r-1} y_{j,2k} = y_{j,2r+1} = p_3\left(x_{j,2r+1}\right),$$

and thus equation (6) becomes $y_{j+1,4r+2} = p_3\left(x_{j,2r+1}\right)$ for any $\alpha \in \mathbb{R}$. Similarly, equation (7) implies $y_{j+1,2s+1} = p_3\left(x_{j+1,2s+1}\right)$. We conclude that $y_{j+1,k} = p_3\left(x_{j+1,k}\right)$, for every $k \in \mathbb{Z}$ if $y_{j,k} = p_3\left(x_{j,k}\right)$, for every $k \in \mathbb{Z}$, and thus MS_α schemes reproduce cubic polynomials. For practical implementations of a multistep scheme, it is necessary to first apply a one-step subdivision scheme. Let $\{y_{j,k}\}_k$ be some initial data. As a first step, we apply DD's equation

$$y_{j+1,l} = \sum_{k\in\mathbb{Z}} \gamma^{DD2}_{2k-l} y_{j,2k}, \tag{8}$$

followed by MS_α. This algorithm is as local as the corresponding DD subdivision scheme in the sense that the fundamental function has support $[-3, 3]$. By induction on j, we get the following lemma.

Lemma 1. *MS_α schemes reproduce cubic polynomials and are interpolatory when using a one step interpolatory 4-point dyadic DD interpolation as initialization.*

We get a stronger result by choosing a specific α. We can write any quartic polynomial p_4 as $p_4(x) = a_4x^4 + p_3(x)$, where p_3 is some cubic polynomial. Because of the Generalized Rolle's theorem and because $\frac{p_4^{(4)}}{4!} = a_4$, given any 4 points ξ_1,ξ_2,ξ_3, and ξ_4, the corresponding cubic polynomial p_{L3} approximates p_4 with error

$$p_4(x) - p_{L3}(x) = a_4\left(x-\xi_1\right)\left(x-\xi_2\right)\left(x-\xi_3\right)\left(x-\xi_4\right), \tag{9}$$

for some ξ. In other words, the error depends only on a_4 and the geometry of the sample points ξ_i with respect to x. This makes the task of canceling out the errors convenient as we shall see.

Suppose that for some j, $y_{j,2k} = p_4\left(x_{j,2k}\right)$ and $y_{j-1,k} = p_4\left(x_{j-1,k}\right)$ for every $k \in \mathbb{Z}$. We can write $y_{j+1,4r+2}$ for any $r \in \mathbb{Z}$ in terms of this initial data (y_j and y_{j-1}) by substituting equation (7) into (6) to get

$$y_{j+1,4r+2} = \alpha \sum_{k\in\mathbb{Z}} \gamma^{DD4}_{4k-2r-1} y_{j-1,2k} + (1-\alpha) \sum_{k\in\mathbb{Z}} \gamma^{DD2}_{2k-2r-1} y_{j,2k}. \tag{10}$$

We want to show that $y_{j+1,4r+2} = p_4(x_{j,2r+1})$ for some $\alpha \in \mathbb{R}$, and so we substitute $y_{j,2k} = p_4(x_{j,2k})$ and $y_{j-1,k} = p_4(x_{j-1,k})$ into the two sums of this last equation. We compute both sums in equation (10) explicitly using equation (9) (Rolle's):

$$\sum_{k\in\mathbb{Z}} \gamma^{DD4}_{4k-2r-1} y_{j-1,2k} = p_4(x_{j,2r+1}) - \frac{105a_4}{2^{4j}}, \tag{11}$$

and similarly $\sum_{k\in\mathbb{Z}} \gamma^{DD2}_{2k-2r-1} y_{j,2k} = p_4(x_{j,2r+1}) - \frac{9a_4}{2^{4j}}$. Hence, setting $\alpha = -3/32$ in equation (10), we get

$$y_{j+1,4r+2} = p_4(x_{j,2r+1}) - \frac{105\alpha + 9(1-\alpha)}{2^{4j}} a_4 = p_4(x_{j+1,4r+2}).$$

Therefore, $MS_{-3/32}$ reproduces quartic polynomials once the data has been properly initialized. While there are no 4-point subdivision schemes capable of interpolating $y_{j-1,k} = p_4(x_{j,k})$ to $y_{j,2k+1} = p_4(x_{j,2k+1}) - \frac{105a_4}{16\times 2^{4j}}$ and $y_{j,2k} = p_4(x_{j,2k})$ for all $k \in \mathbb{Z}$, there exist 5-point subdivision schemes such as the subdivision scheme described by the next algorithm.

Algorithm 2. *For a given integer j, begin with some initial y values $y_{j,k}$ $k \in \mathbb{Z}$ over dyadic numbers $x_{j,k} = k/2^j$,*

1) *recopy data at $x_{j+1,2k} = x_{j,k}$: $y_{j+1,2k} = y_{j,k}$ $\forall k \in \mathbb{Z}$;*
2) *extrapolate $y_{j,k+4}$ using $y_{j,k-2}, y_{j,k-1}, y_{j,k}, y_{j,k+1}, y_{j,k+2}$ by the formula $\gamma_{j,k} = 5y_{j,k-2} - 24y_{j,k-1} + 45y_{j,k} - 40y_{j,k+1} + 15y_{j,k+1}$, $\forall k \in \mathbb{Z}$;*
3) *interpolate midpoint value using the 4-adic DD formula $y_{j+1,2k+1} = \frac{-7y_{j,k-2} + 105y_{j,k} + 35y_{j,k+2} - 5\gamma_{j,k}}{128}$, $\forall k \in \mathbb{Z}$.*

To see that algorithm 2 properly initializes the placeholders, observe that if we assume $y_{J,k} = p_4(x_{J,k})$, then we only need to check that $y_{J+1,2k+1} = p_4(x_{J+1,2k+1}) - \frac{105a_4}{16\times 2^{4(J+1)}}$. However, if $y_{J,k} = p_4(x_{J,k})$ is satisfied, then $\gamma_{J,k} = p_4(x_{J,k+4})$ since it can be derived by finding the quartic polynomial $p_{J,k}$ satisfying $p_{J,k}(x_{j,l}) = y_{J,l}$ for $l = k-2, ..., k+2$ and setting $\gamma_{J,k} = p_{J,k}(x_{J,k+4})$. Hence, by equation (11), we have the following lemma.

Lemma 3. *Algorithm 2 describes a 5-point dyadic subdivision scheme such that with $y_{j-1,k} = p_4(x_{j-1,k})$ and $p_4(x) = a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$ is a quartic polynomial, $y_{j,2k+1} = p_4(x_{j,2k+1}) - \frac{105a_4}{16\times 2^{4j}}$ and $y_{j,2k} = p_4(x_{j,2k})$ for all $k \in \mathbb{Z}$.*

Because we have a proper initialization scheme, we can reproduce quartic polynomials as shown in the next proposition and we say that $MS_{-3/32}$ is a high resolution subdivision scheme.

Proposition 4. *With Algorithm 2 as an initialization step, $MS_{-3/32}$ reproduces quartic polynomials.*

Only subdivision schemes using at least 5 points can interpolate quartic polynomials, and the support of the fundamental function is at least of *size 8*, whereas the algorithm described by Proposition 4 ($MS_{-3/32}$) leads to fundamental functions having compact support of *size 7* taking into account the 5-point initialization scheme.

§5. Sufficient Conditions for Regularity

Given that MS_0 is equivalent to the DD subdivision which is C^1, it is reasonable to expect MS_α to be C^1 for some range of α values. Motivated by Proposition 4, we need to show that this range of values includes $\alpha = -3/32$. At this point, it is convenient to rewrite equation (3) in terms of Laurent polynomials. Given some data $y_{j,k}$, define $P^j(z) = \sum_{k\in\mathbb{Z}} y_{j,k} z^k$. If $P_2(z) = \sum_{k\in\mathbb{Z}} \gamma_k^{DD2} z^k$, then the equation of the 4-point dyadic DD scheme (equation (8)), can be rewritten $P^{j+1}(z) = P_2(z)P^j(z^2)$. Similarly, if $P_4(z) = \sum_{k\in\mathbb{Z}} \gamma_k^{DD4} z^k$, then the 4-adic subdivision scheme is given by $P^{j+1}(z) = P_4(z)P^j(z^2)$. We can rewrite the general equation for $b-$ adic multistep schemes as

$$P^{j+1}(z) = \sum_{k=1}^{M} \Gamma_k(z) P^j\left(e^{2\pi i k/b} z^b\right),$$

where the Γ_k must be Laurent polynomials. For dyadic ($b = 2$) and two-step schemes ($M = 2$), this equation becomes

$$P^{j+1}(z) = \Gamma_1(z) P^j\left(z^2\right) + \Gamma_2(z) P^j\left(-z^2\right). \tag{12}$$

Observe that $E^j(z^2) = (P^j(z^2) + P^j(-z^2))/2$ selects only the stable values (even indices), whereas $O^j(z^2) = (P^j(z^2) - P^j(-z^2))/2$ depends only on the guesses (odd indices) of y_j. Hence, we can rewrite equation (4) as $P^{j+1}(z) = (\tau(z) + \alpha)\, E^j(z^2) + \alpha O^j(z^2)$, where $\tau(z) = P_4(z) - \alpha P_2(z^2)$. After rearranging the terms, we get $P^{j+1}(z) = (\frac{\tau(z)}{2} + \alpha) P^j(z^2) + \frac{\tau(z)}{2} P^j(z^2)$. Because $\tau(z) = (P_4(z) - P_4(-z))/2 + (1-\alpha)(P_4(z) + P_4(-z))/2$, the MS_α symbols are

$$\Gamma_1(z) = \Gamma_2(z) + \alpha,$$
$$\Gamma_2(z) = \frac{P_4(z) - P_4(-z)}{4} + (1-\alpha)\frac{P_4(z) + P_4(-z)}{4}.$$

Following Dyn [8], we want to find corresponding schemes for the (forward) finite differences. Let $dx_j = 1/2^j$ and write $D_{j,k} = \frac{dy_{j,k}}{dx_j} = 2^j(y_{j,k+1} -$

$y_{j,k}$), and define higher order finite differences recursively $D^n_{j,k} = d^{(n)} y_{j,k} 2^{jn}$. Because $\sum_{k\in\mathbb{Z}} 2^j \left(y_{j,k+1} - y_{j,k}\right) z^k = \sum_{k\in\mathbb{Z}} 2^j y_{j,k}\left(z^{k-1} - z^k\right)$, we have

$$H^j_n(z) = \frac{2(1-z)}{z} H^j_{n-1}(z) = \left(\frac{2(1-z)}{z}\right)^n P^j(z),$$

where $H_0(z) = P(z)$. They can be computed by

$$\begin{aligned} H^{j+1}_n(z) = {} & \left(\frac{2z}{1+z}\right)^n \Gamma_1(z) H^j_n\left(z^2\right) \\ & + \left(\frac{-2z(1-z)}{1+z^2}\right)^n \Gamma_2(z) H^j_n\left(-z^2\right). \end{aligned} \tag{13}$$

H_n is the symbol of a multistep scheme if $n = 1,2,3,4$ because $\Gamma_1(z)/(1+z)^n$ and $\Gamma_2(z)/(1+z^2)^n$ are Laurent polynomials

We define $dH^j_n(z) = H^j_{n+1}(z)/2^j$ as symbols of $dD^{n-1}_{j,k} = d\frac{dy_{j,k}}{dx_j}$ and since $dH^j_n(z) = H^j_{n+1}(z)/2^j$, dH_{n-1} is the symbol of a multistep scheme for $n = 1,2,3,4$. Using results from Dyn [8], we have the following theorem.

Theorem 5. *Given Laurent polynomials $\Gamma_1(z)$ and $\Gamma_2(z)$, the multistep scheme defined by $P^{j+1}(z) = \Gamma_1(z) P^j\left(z^2\right) + \Gamma_2(z) P^j\left(-z^2\right)$ is C^n if the symbol corresponding to finite differences of order $n+1$, $dH^j_n(z) = \frac{2^{jn}(1-z)^{n+1}}{z^{n+1}} P^j(z)$ is the symbol of a multistep scheme converging uniformly to zero for all bounded initial data.*

Proof: See the proof of Theorem 3.4 in [8] or Section 4.2 in [10] as it applies to multistep schemes. The key point is that for an iterative interpolation scheme (even a nonstationary one) to be C^n, it is sufficient for the finite differences $d^{n+1} y_{j,k}/\left(dx_j\right)^n$ to converge uniformly to zero. □

In general, given $y_{j+1,l} = \sum_{k\in\mathbb{Z}} \gamma_{2k-l} y_{j,k}$, a sufficient condition for $y_{j,k} \to 0$ uniformly as $j \to \infty$ is that $\lambda = \max_{l=0,1}\left\{\sum_{k\in\mathbb{Z}} |\gamma_{2k-l}|\right\} < 1$. For a multistep scheme given by $y_{j+1,l} = \sum_{k\in\mathbb{Z}} \gamma^{(1)}_{4k-l} y_{j,2k} + \gamma^{(2)}_{4k+1-l} y_{j,2k+1}$, $\lambda_{HR} = \max_{l=0,1}\left\{\sum_{k\in\mathbb{Z}} \left|\gamma^{(1)}_{2k-l}\right| + \left|\gamma^{(2)}_{2k+1-l}\right|\right\}$ implies $M_{j+1} \leq \lambda_{HR} M_j$ where $M_j = \sup\{|y_{j,k}| : k \in \mathbb{Z}\}$. To write this statement in symbols, define $\|Q(z)\|_{sup} = \sup_k\{|q_k|\}$ and $\|Q(z)\|_\Sigma = \sum_k |q_k|$ where $Q(z) = \sum_k q_k z^k$. Now, if $P^{j+1}(z) = \tilde{\Gamma}_1(z) P^j\left(z^2\right) + \tilde{\Gamma}_2(z) P^j\left(-z^2\right)$ and

$$\lambda_{HR} = \max\left\{\lambda_1, \lambda_2\right\}, \tag{14}$$

where $2\lambda_{1/2} = \left\|\tilde{\Gamma}_1(z) \pm \tilde{\Gamma}_1(-z) + \tilde{\Gamma}_2(z) \mp \tilde{\Gamma}_2(-z)\right\|_\Sigma$, then

$$\left\|P^{j+1}(z)\right\|_{sup} \leq \lambda_{HR}\left\|P^j(z)\right\|_{sup}.$$

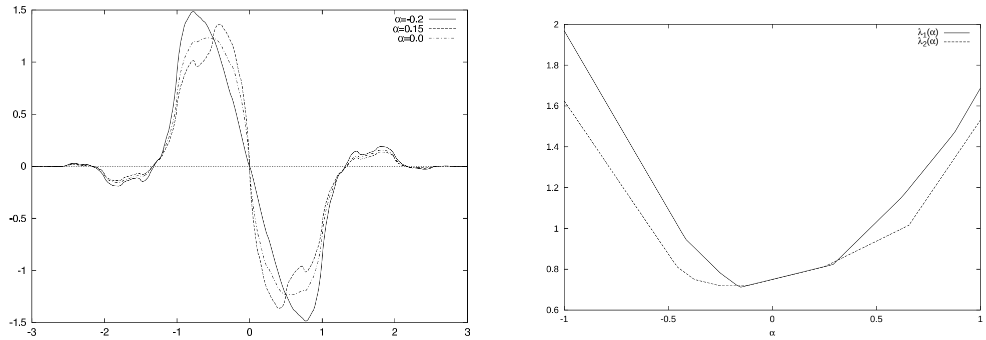

Fig. 2. Derivatives of MS_α fundamental functions (left). MS_α is differentiable if $\lambda_{HR}(\alpha) = \max\{\lambda_1(\alpha), \lambda_2(\alpha)\} < 1$ (right).

Lemma 6. *A multistep scheme given by the symbol equation* $P^{j+1}(z) = \tilde{\Gamma}_1(z)P^j\left(z^2\right) + \tilde{\Gamma}_2(z)P^j\left(-z^2\right)$ *converges uniformly to zero for all bounded initial values if* $\lambda_{HR} < 1$, *where* λ_{HR} *is as in equation (14).*

We are now ready to prove the following theorem which shows that MS_α are smooth for α near 0.

Theorem 7. *For* $-25/56 < \alpha < 15/32$, MS_α *interpolants are* C^1.

Proof: By Theorem 5, it is enough to show that $dD^1_{j,k}$ converges uniformly to zero for all bounded initial data. The symbol of the multistep scheme $dD^1_{j,k}$, dH_1 is given by

$$dH_1^{j+1}(z) = \tilde{\Gamma}_1 dH_1^j\left(z^2\right) + \tilde{\Gamma}_2 dH_1^j\left(-z^2\right),$$

where $\tilde{\Gamma}_1(z) = 2z^2\Gamma_1(z)/\left(1+z\right)^2$ and $\tilde{\Gamma}_2(z) = 2z^2(1-z)^2\Gamma_2(z)/\left(1+z^2\right)^2$ (see equation (13)). By Lemma 6, it is sufficient to prove that $\lambda_{HR} < 1$. Note $f_{a,b} = |a\alpha + b|$, then we have $\lambda_1 = \frac{1}{64}(5 + 2f_{4,1} + 2f_{-8,7} + 2f_{12,5} + f_{32,5} + f_{-8,5} + f_{24,-7}$ and $\lambda_2 = \frac{1}{64}(5 + 2f_{4,1} + 2f_{8,3} + 2f_{-4,1} + f_{32,5} + f_{-32,21} + f_{8,1} + f_{24,11}$, and so we have that $\lambda_1 < 1$ for $-25/56 < \alpha < 15/32$ and $\lambda_2 < 1$ for $-7/12 < \alpha < 5/8$, $\lambda_2 < 1$ hence the result (see Fig. 2). $\square$

References

1. I. Daubechies, Orthonormal bases of compactly supported wavelets, Comm. Pure & Appl. Math. **41** (1988), 909–996.
2. S. Dekel and N. Dyn, Poly-scale refinability and subdivision, Applied and Computational Harmonic Analysis **13** (2002), 35–62.
3. G. Deslauriers and S. Dubuc, Symmetric iterative interpolation processes, Constr. Approx. **5** (1989), 49–68.

4. G. Deslauriers, S. Dubuc, and D. Lemire, Une famille d'ondelettes biorthogonales sur l'intervalle obtenue par un schéma d'interpolation itérative, Ann. Sci. Math. Québec **23** no. 1 (1999), 37–48.
5. F. Dubeau and R. Gervais, Procédures locale et non locale d'interpolation à l'aide de fonctions splines quadratiques, Ann. Sci. Math. Québec **23** no. 1 (1999), 49–61, 1999.
6. S. Dubuc, Interpolation through an iterative scheme, J. Math. Anal. Appl. **114** (1986), 185–204.
7. S. Dubuc, D. Lemire, J.-L. Merrien, Fourier analysis of 2-point Hermite interpolatory subdivision schemes, J. Fourier Anal. Appl. **7** no. 5 (2001).
8. N. Dyn, Subdivision schemes in computer-aided geometric design, in *Advances in Numerical Analysis* (W. Light, ed.), vol. 2, Clarendon Press (1992), 36–104.
9. N. Dyn, J. A. Gregory, and D. Levin, A 4-point interpolatory subdivision scheme for curve design, Comput. Aided Geom. Design **4** (1987), 257–268.
10. N. Dyn, D. Levin, Subdivision schemes in geometric modelling, Acta Numerica **12** (2002), 1–72.
11. B. Han, Approximation properties and construction of Hermite interpolants and biorthogonal multiwavelets, J. Approx. Theory **110** no.1 (2001), 18–53.
12. M. F. Hassan, I. P. Ivrissimitzis, N. A. Dodgson, and M. A. Sabin, An interpolating 4-point C^2 ternary stationary subdivision scheme, submitted.
13. C. Heil and D. Colella, Matrix refinement equations and subdivision schemes, J. Fourier Anal. Appl. **2** (1996), 363–377.
14. F. Kuijt and R. van Damme, Stability of subdivision schemes, Memorandum no. 1469, Faculty of Applied Mathematics, University of Twente, Netherlands.
15. J.-L. Merrien, A family of Hermite interpolants by bisection algorithms. Numer. Algorithms **2** (1992), 187–200.
16. J.-L. Merrien, Interpolants d'Hermite C^2 obtenus par subdivision. Math. Model. Numer. Anal. **33** (1999), 55–65.
17. G. Plonka, Approximation order provided by refinable function vectors, Constr. Approx. **13** (1997), 221–244.

Daniel Lemire, National Research Council of Canada (NRC)
46 Dineen Drive, Fredericton, N.B.
E3B 5A3 Canada
`lemire@ondelette.com`, `http://www.ondelette.com/acadia/`