The dyadic iterative interpolation method and some extensions

S. De Marchi and M. Morandi Cecchi

Abstract. The Dyadic Iterative Interpolation Method (DIIM), presented in [1], concerns with the construction of an interpolating function on dyadic rationals. We characterize it in terms of Uniform Dyadic Subdivision Schemes (UDSS) [2]. We show that the DIIM is a 4-point subdivision scheme that generates almost twice differentiable functions. We also provide a proof to an unsolved problem left in [1] related to the so called *fundamental interpolation function*, namely the existence of a unique minimum in the interval $[1, 2]$. We extend the method to a two dimensional setting proving interesting properties.

§1. Introduction

In [1] the author presents a method for constructing an interpolating function extending smoothly a sequence defined on the integers. The extension is made on the set of dyadic numbers. The process is iterative and produces almost $\mathcal{C}^2$ functions defined on the whole real axis. In this paper we shall refer to this method as the Dyadic Iterative Interpolation Method.
The aim of this work is twofold. First we discuss some property of the DIIM method proving one open problem, that is the uniqueness of a minimum in the interval $[1, 2]$ of the fundamental interpolation. For instance, during the whole paper we use the name of "mother interpolant" since it is related with the mother functions in wavelets and splines literature. Secondly we place the DIIM in the framework of Uniform Dyadic Subdivision Scheme as presented in [2]. The use of this theory allows us to recover the characteristics of the DIIM in a easier way.
Finally we present a possible extension of the method to bivariate functions defined on regular grids. Indeed, we restrict ourselves to the square grid of dyadics in $\mathbb{R}^2$.

The paper is organized as follows. In Section 2 we presents the DIIM method and its fundamental characteristics. We provide a proof of an unsolved problem concerning the uniqueness of a minimum of the so called mother interpolation function. We also present an algorithm to compute the interpolation function in a efficient way. In Section 3 we analyse the method in the point of view of uniform dyadic subdivision schemes, as discussed in [2]. In Section 4 we apply the general theory for UDSS to the DIIM method proving that the DIIM method, applied to functions defined on dyadics, reproduces $\mathcal{C}^2$ interpolating functions. Finally in Section 5 we present an algorithm for computing the interpolating function in the bidimesional case.

§2. The DIIM method

Let $y(n)$ be a function defined on $\mathbb{Z}$. The set of integeral numbers can be seen as the set

$$D_0 = \{t_i^0 = \frac{i}{2^0}, \quad i \in \mathbb{Z}\},$$

that is the set of dyadic number of *depth* zero.
The method constructs the interpolation function starting from $\{y(t_i^0)\}$ in the following way.

(i) `Extend the function to` $D_{n+1}, \quad n \geq 0$.
To accomplish this we note that $y(\cdot)$ is already defined on D_0. Hence, if y is defined on D_n and taking $h = \frac{1}{2^{n+1}}$ and $t \in D_{n+1} \setminus D_n$ then

$$\{t_1 = t - 3h, t_2 = t - h, t_3 = t + h, t_4 = t + 3h\} \in D_n. \tag{1}$$

The extension is made by

$$y(t) = \frac{1}{16}\{-y(t_1) + 9y(t_2) + 9y(t_3) - y(t_4)\} \tag{2}$$

The extension continues to D_∞, that is the set of **dyadic numbers**.

(ii) Since D_∞ is dense in $\mathbb{R}$ then $y(t)$ has a continuous extension to $\mathbb{R}$ and we call it again $y(t)$. Moreover, it can be proved that $y \in \mathcal{C}^1$ (see [1]). We outline some properties of the interpolating function.

1. $y(t) \in \mathcal{C}^1$.
2. The derivative $y'(t)$ is close to be differentiable. That is for each positive integer M there exists a constant C_M such that for any $0 < h < 1$ and $|t| \leq M$ we have

$$\frac{|y'(t+h) - y'(t)|}{h} \leq C_M log(\frac{1}{h}) \tag{3}$$

3. Let $F(t)$ be the $\underline{interpolating\ function}$ of the sequence

$$\begin{array}{l} y(0) = 1 \\ y(n) = 0 \quad n \neq 0 \end{array} \tag{4}$$

We call the function $F(\cdot)$ the **mother** of the process (2). In fact this function provides a nice way to describe any interpolation defined on the dyadics.

Proposition 1. *Any interpolation function $y(t)$ is a linear combination of translates of F, $F(t-n)$.*

Moreover it is easy to prove, also by direct computations, that holds the following result.

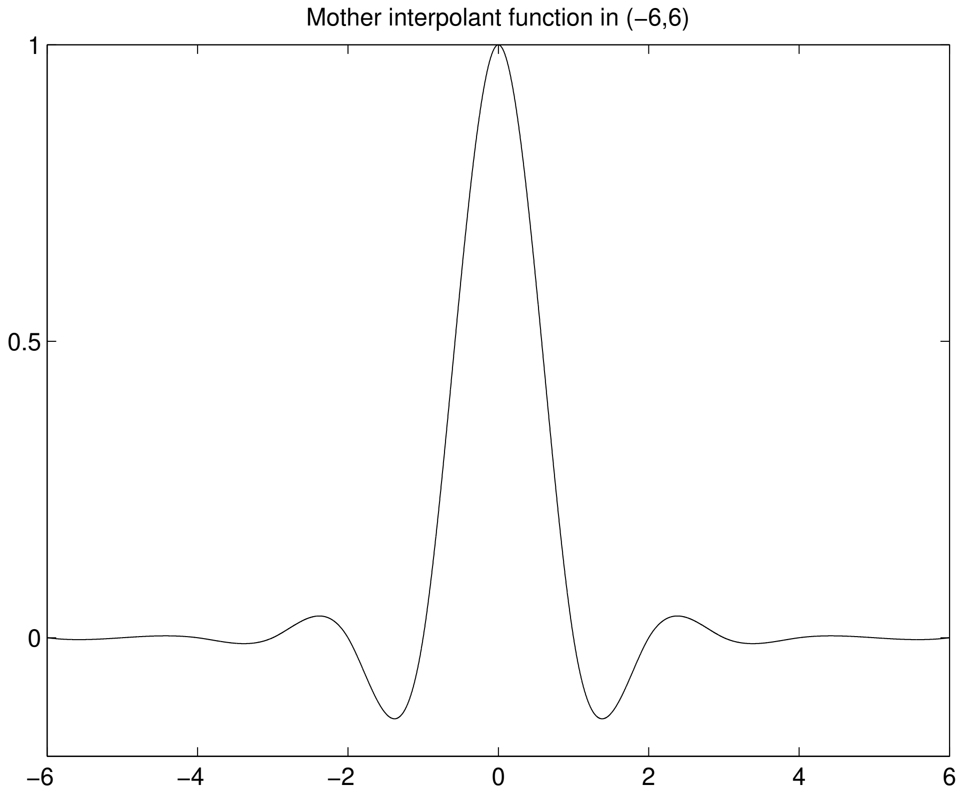

Fig. 1. The graph of the *mother* interpolant.

Proposition 2. *The mother function vanishes outside* $(-3, 3)$.

Proof: We notice that this function is an even function (see Fig. 1). Therefore, we only need to study its behaviour in the half positive interval $(0, 3)$. Any dyadic number in (0,3) can be computed by the recurrence

$$\begin{aligned} t_0 &= 0 \\ t_{n+1} &= t_n + \tfrac{3}{2^{n+1}} \quad n > 0 \end{aligned} \tag{5}$$

The solution of this recurrence is $t_n = 3 - \frac{3}{2^n}$. For each n, $F(t)$ vanishes outside the interval $[-t_n, t_n]$. Then, as n tends to infinity t_n tends to 3. ■

From the Propositions 1 and 2 follows the next theorem which says that each interpolation function defined by means of the scheme (2) can be written as a linear combination of translates of the mother function.

Theorem 3. *If* $y(n)$ *is defined on* D_0 *then* $y(t)$ *on* D_∞ *is given*

$$y(t) = \sum_{n=k-2}^{k+3} y(n)F(t-n) \tag{6}$$

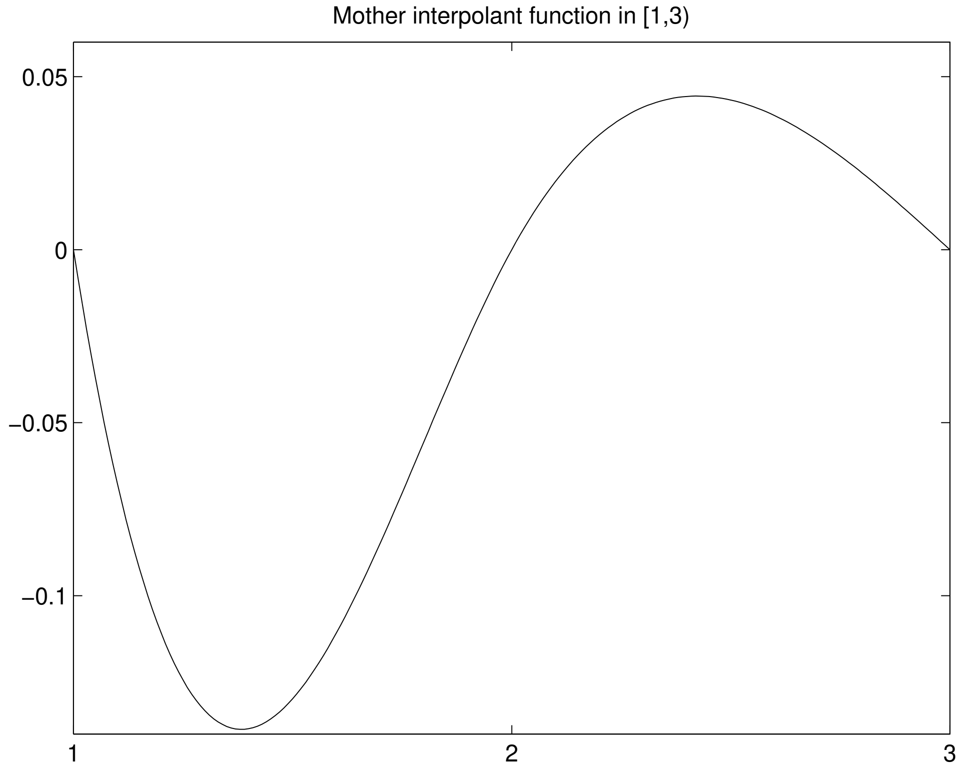

Fig. 2. The *mother* interpolant in the interval $[1,3)$.

where $k = \lfloor t \rfloor$.

With reference to the approximation property of the scheme (2), we may quote another interesting result.

Theorem 4. *If the original sequence are the values of a cubic polynomial* $p(n)$, *then* $y(t) = p(t)$.

Proof: *The proof is by recursion on the depth of the dyadic number* t. *If* p *is a cubic polynomial we can write*

$$p(0) = \sum_{k=0}^{3} p(2k-3) l_k(0)$$

where $l_k(t),\ k = 0, ..., 3$ *are the four elementary Lagrange's polynomials based on the knots* $-3, -1, 1, 3$ *(remember that we are refering to the interpolating process (2)). Since* $l_0(0) = -1, l_1(0) = l_2(0) = 9$ *and* $l_3(0) = -1$ *we have* $p(0) = \dfrac{-p(-3) + 9p(-1) + 9p(1) - p(3)}{16}$. *Setting* $y(u) = p(uh + t)$, *with* $h = 2^{-n}$, *we conclude.* ■

Remark. *Theorem 4 says that the process is exact up to order three.*

From Fig. 2 we notice that there is `one and only one minimum` in the interval $[1,2]$. This remark that follows from experimental computations can also be proven directely from theory. First of all we say something about the sign of the mother function in $[1,2]$.

Theorem 5. *The mother function is negative in* $(1,2)$ *and for all* $t \in (1,2)$ *we have*

$$F(t) = \frac{-(t-1)(t-2)(t-3)}{6} + 4F(1+t) - F(4-t) \qquad (7).$$

Proof: The interpolation of the sequence $y(n) = \frac{-(n-1)(n-2)(n-3)}{6}$ is the function $y(t) = \frac{-(t-1)(t-2)(t-3)}{6}$. Theorem 3 in the interval $[1,2]$ gives for the mother function the expression

$$F(t) = \sum_{k=-1}^{4} y(k)F(t-k).$$

The conclusion follows since $y(-1) = 4, y(0) = 1, y(1) = y(2) = y(3) = 0, y(4) = -1$ and by parity $F(t-4) = F(4-t)$. ■

Using this result and the Rolle's theorem in the interval $[1,2]$ we may conclude that there exists at least one minimum. We claim that there exists right only one. First we need a definition.

Definition 6. *Given a sequence of function values* $\{f_i^k = f(t_i^k),\ i \in I, k > 0\}$. *We say that* f *is strictly convex (concave) at a point* t_i *if the second divided difference*

$$f[t_{i-1}, t_i, t_{i+1}] > 0 \quad (resp. < 0).$$

Using this definition we claim the following theorem.

Theorem 7. *The mother interpolant has one and only one minimum in* $[1,2]$.

Proof: We note that at each step k the mother interpolant is strictly convex (see Fig. 2). Therefore by using Weierstrass' theorem on the compact $[1,2]$ we conclude. ■

It remains to point out how we can find such a minimum. We suggest two possibilities.

1. Using small mesh size and linear interpolation of the data, we can use linear programming techniques.
2. We saw that the process is exact for polynomials at most of degree 3. Let $s(t)$ the cubic spline interpolating the mother function in its support, that is $(-3,3)$. We know that

$$\|s - F\| = O(h^4) \qquad (8)$$

where, in our case, $h = \frac{1}{2^k}$, $\quad k > 0$. Thus, if we are able to find a minimum of $s(t)$ in $[1, 2]$, we can say that if differs from the "real" one up to the approximation order. Therefore, taking k sufficiently large, we can assume that the minimum of s in $[1, 2]$ coincides with the value of the minimum of F.

Concerning with the point of minimum, we can use an heuristics. As soon as we found to values $F(t_1)$ and $F(t_2)$ that differ less than the step 16^k, then the value $\tilde{t}$ of the minimum is $\tilde{t} = \frac{t_1+t_2}{2}$.

2.1 An algorithm to compute the interpolation function

The relation (2), used to compute point in D_{n+1} starting from the ones in D_n, suggests a way to compute the interpolant.
We need of matrix $(n+1) \times 6$, let's say $\{a_{i,j}\}$, $\;i = 0, ..., n;\; j = 1, ..., 6$. The element $a_{i,j}$ is given as follows:

$$a_{i,j} = y\left((m - 3 + j)2^{-i}\right) \tag{9}$$

with $m = \lfloor t2^i \rfloor$. Each successive row is computed from the preceding ones. The algorithm that evaluates the interpolant on a point in D_n, is as follows.

Algorithm 1

`Start with the set` $D_0 = \mathbb{Z}$ `of integers and take` $k \in D_0$`. More-over the sequence` $y(n),\; n \in \mathbb{Z}$ `is given.`

for $n = 1, ...$ **do** $\quad$ {n stands for the depth }
$\quad t_n = \frac{k}{2^n}$ $\quad$ {t_n is a dyadic number of depth n }
$\quad$ **for** i=0,...,n **do**
$\quad$ **begin**
$\quad\quad$ m=$\lfloor t_n 2^i \rfloor$
$\quad\quad$ **for** j=1,...,6 **do**
$\quad\quad\quad$ $a_{i,j} = y\left((m - 3 + j)2^{-i}\right)$;
$\quad\quad$ **if** $t - \lfloor t_n 2^i \rfloor \le \frac{1}{2}$ **then**
$\quad\quad$ **begin**
$\quad\quad\quad$ $a_{i+1,1} = a_{i,2}$
$\quad\quad\quad$ $a_{i+1,2} = \frac{1}{16}(-a_{i,1} + 9a_{i,2} + 9a_{i,3} - a_{i,4})$
$\quad\quad\quad$ $a_{i+1,3} = a_{i,3}$
$\quad\quad\quad$ $a_{i+1,4} = \frac{1}{16}(-a_{i,2} + 9a_{i,3} + 9a_{i,4} - a_{i,5})$
$\quad\quad\quad$ $a_{i+1,5} = a_{i,4}$
$\quad\quad\quad$ $a_{i+1,6} = \frac{1}{16}(-a_{i,3} + 9a_{i,4} + 9a_{i,5} - a_{i,6})$
$\quad\quad$ **end**
$\quad\quad$ **elseif** $t - \lfloor t_n 2^i \rfloor > \frac{1}{2}$ **then**
$\quad\quad$ **begin**
$\quad\quad\quad$ $a_{i+1,1} = \frac{1}{16}(-a_{i,1} + 9a_{i,2} + 9a_{i,3} - a_{i,4})$
$\quad\quad\quad$ $a_{i+1,2} = a_{i,3}$
$\quad\quad\quad$ $a_{i+1,3} = \frac{1}{16}(-a_{i,2} + 9a_{i,3} + 9a_{i,4} - a_{i,5})$
$\quad\quad\quad$ $a_{i+1,4} = a_{i,4}$

$$a_{i+1,5} = \frac{1}{16}(-a_{i,3} + 9a_{i,4} + 9a_{i,5} - a_{i,6})$$
$$a_{i+1,6} = a_{i,5}$$

end

end

$y(t) = a_{n,j}$

end

In the next section we shall see that this algorithm is the basic one in order to describe the "future" behaviour of the interpolant in the refined dyadic set.

§3. Analysis of DIIM in the framework of UDSS

In this section we want to present the above DIIM scheme as a Uniform Dyadic Subdivision Scheme (shortly UDSS). The idea is re-find and re-formulate the properties of the DIIM by using the theory of uniform dyadic subdivision schemes. Useful surveys on this subject are the papers [2,3,4] and the references therein.

3.1 Generality on Uniform Dyadic Subdivision Schemes (UDSS)

Let $f_i^k \in \mathbb{R}^n, \quad i \in \mathbb{Z}, \quad k \geq 0$ denote a sequence of points in $\mathbb{R}^n$. A uniform dyadic subdivision scheme is defined by

$$\begin{cases} f_{2i}^{k+1} = \sum_{j=0}^{m} a_j f_{i+j}^k \\ f_{2i+1}^{k+1} = \sum_{j=0}^{m} b_j f_{i+j}^k \end{cases} \tag{10}$$

where $m > 0$ stands for the "class" of UDSS. For instance, the DIIM is a *4-points subdivision scheme* with $m = 3$.

Let f be a function defined in $\mathbb{R}$, we may consider the sequence

$$\{f_i^k = f(t_i^k), \quad t_i^k \in D_k, \quad i \in \mathbb{Z}, k \geq 0\}.$$

Definition. *The values $\{f_i^k\}$ are called the control points. The linear interpolant to these values is called the control polygon.*

Proposition 8. *The subdivision scheme is completely defined by the vectors of the coefficients, in what follows control vectors, that is $\mathbf{a} = (a_0, a_1, ..., a_m)$ and $\mathbf{b} = (b_0, b_1, ..., b_m)$.*

Proof: It is easy to see that if we know the vectors $\mathbf{a}$ and $\mathbf{b}$ we can compute the new control points by means of (10). ■

We notice that expressions (10) can be seen as two inner products between the pair of vectors $(\mathbf{a}, \mathbf{f}_{i,m}^k)$ and $(\mathbf{b}, \mathbf{f}_{i,m}^k)$ where

$$\mathbf{f}_{i,m}^k = (f_i^k, f_{i+1}^k, ..., f_{i+m}^k)^T.$$

But we can say something more. At each step, the process (10) defines f_{2i}^{k+1} that replaces f_i^k at the mesh point $t_{2i}^k = t_i^k$ and f_{2i+1}^{k+1} is inserted at the new mesh point $t_{2i+1}^k = \frac{t_i^k + t_{i+1}^k}{2}$.

Consider the interval $J_i^k = [t_i^k, t_{i+1}^k] = [\frac{i}{2^k}, \frac{i+1}{2^k}]$. It is easy to prove that the control points that determine the future behaviour of the process in J_i^k is the vector

$$\mathbf{f}_{i,m}^k = (f_i^k, f_{i+1}^k, ..., f_{i+1+p(m)}^k)^T$$

where

$$p(m) = \begin{cases} 2m-1 & \text{if} \quad a_m \neq 0 \\ 2m-2 & \text{if} \quad a_m = 0 \end{cases} \tag{11}$$

Definition (see [2]). *We call* `generator matrix of order M`, $M = p(m) + 3$ *the matrix*

$$A = \begin{pmatrix} a_0 & \cdots & \cdots & a_m & 0 & \cdots & \cdots & 0 \\ b_0 & \cdots & \cdots & b_m & 0 & \cdots & \cdots & 0 \\ 0 & a_0 & \cdots & . & a_m & \cdots & \cdots & 0 \\ 0 & b_0 & \cdots & . & b_m & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & a_0 & \cdots & \cdots & a_m & 0 \\ 0 & \cdots & \cdots & b_0 & \cdots & \cdots & b_m & 0 \end{pmatrix} \tag{12}$$

Proposition 9. *The vectors* $\mathbf{f}_{2i,m}^{k+1}$ *and* $\mathbf{f}_{2i+1,m}^{k+1}$ *relative to the intervals* $J_{2i}^{k+1} = [t_{2i}^{k+1}, t_{2i+1}^{k+1}]$ *and* $J_{2i+1}^{k+1} = [t_{2i+1}^{k+1}, t_{2i+2}^{k+1}]$*, respectively, are defined by two linear transformations on* $\mathbf{f}_{i,m}^k$*. That is*

$$\mathbf{f}_{2i,m}^{k+1} = A_0 \mathbf{f}_{i,m}^k \tag{13}$$

$$\mathbf{f}_{2i+1,m}^{k+1} = A_1 \mathbf{f}_{i,m}^k. \tag{14}$$

The matrices A_0 and A_1 are non singular of order $M-1 \times M-1$ obtained from A in the following way

$$A_0 = A(1...M-1;\ 1...M-1)$$

$$A_1 = A(2...M;\ 1...M-1)$$

The notation $A(j...j+k;\ i...i+l)$ says that we take the row with indices $j, ..., j+k$ and the columns $i, ..., i+l$ of the matrix A. It is straightforward to see that the matrices A, A_0 and A_1 are interrelated. Indeed it holds the following result.

Proposition 10. *Let $\sigma(A)$ denote the spectrum of A, that is the set*

$$\sigma(A) = \{\lambda_1, ..., \lambda_M\} \tag{15}$$

where $\lambda_M = 0$ and $\lambda_{M-1} = a_m$ if $a_m \neq 0$ otherwise $\lambda_{M-1} = b_m$. Then

$$\sigma(A_0) = \{\lambda_1, ..., \lambda_{M-1}\} \tag{16}$$

$$\sigma(A_1) = \{\lambda_1, ..., \lambda_{M-2}, b_0\} \tag{17}.$$

Proof: It sufficies to refer to the form of the matrix (12). ■

We note that both matrix A and matrices A_0, A_1 have $\lambda_1 = 1$ as eigenvalue. The corresponding eigenvector is the vector $\mathbf{u} = (1, ..., 1)^T$.

3.2 Conditions for the $\mathcal{C}^0$ continuity of the limit function

Concerning the convergence of the subdivision scheme to a continuous function, we quote the following result whose proof can be found in [2].

Theorem 11. *The subdivision scheme (10) on the set D_∞ converges to a continuous function F,* **if and only if** *for any given initial data we have*

$$\sum_{i=0}^{m} \mathbf{a}(i) = 1$$

$$\sum_{i=0}^{m} \mathbf{b}(i) = 1.$$

About the proof of the previous theorem, we point out that the only thing we have to assume is that the limit function F does not have to vanish in zero. For example the mother function of the DIIM scheme (2) converges to a limit function F such that $F(0) = 1$. Moreover that scheme satisfies the necessary and sufficient conditions stated above in Theorem 11 for the continuity of the limit function.

3.3 Conditions for the $\mathcal{C}^1$ continuity of the limit function

In order to study the $\mathcal{C}^1$ continuity of the limit function, we have to investigate on the *first order difference process* associated with the process (10). The sequence of first order finite differences is

$$\Delta_i^k = f_{i+1}^k - f_i^k, \quad i \in \mathbb{Z}, \quad k \geq 0. \tag{18}$$

These differences are related with the control vectors $\mathbf{a}$ and $\mathbf{b}$ by means of the following recurrence.

Proposition 12. *The differences Δ_i^k satisfy the recurrence*

$$\begin{aligned} \Delta_{2i}^{k+1} &= \sum_{j=0}^{m-1} c_j \Delta_{i+j}^k \\ \Delta_{2i+1}^{k+1} &= \sum_{j=0}^{m} d_j \Delta_{i+j}^k . \end{aligned} \tag{19}$$

where

$$\begin{aligned} c_j &= \sum_{i=0}^{j} (a_i - b_i) \\ d_j &= a_j + \sum_{i=0}^{j} (b_i - a_i) \end{aligned} \tag{20}$$

that is $d_j = a_j - c_j$.

Proof: From (10) and (18) we have

$$\Delta_{2i}^{k+1} = f_{2i+1}^{k+1} - f_{2i}^{k+1} = \sum_{j=0}^{m} (b_j - a_j) f_{i+j}^k = \sum_{j=0}^{m} c_j f_{i+j}^k \qquad (21).$$

$$\begin{aligned} \Delta_{2i+1}^{k+1} = f_{2(i+1)}^{k+1} - f_{2i+1}^{k+1} = \\ &= \sum_{j=0}^{m} a_j f_{i+j+1}^k - \sum_{j=0}^{m} b_j f_{i+j}^k = \\ &= -b_0 f_i^k + \sum_{j=1}^{m} (a_{j-1} - b_j) f_{i+j}^k + a_m f_{i+m+1}^k . \end{aligned} \tag{22}$$

We note that the sums of the coefficients in (21) and (22) are zero. This says that the sums in (21) and (22) can be written as follows.

$$f_{i+m}^k - f_{i+j}^k = \sum_{s=j}^{m-1} \Delta_{i+s}^k \qquad (23).$$

Substituting in (21) we get the expression (19). Similarly for the (20). ∎

As we did for the $\mathcal{C}^0$ continuity conditions, we can associated to the first order difference scheme (19) a *generator matrix.*

Definition 13. *The* `generator matrix` *associated to the difference process (19) is the following square matrix of order $M-1 \times M-1$, obtained from the matrix A by similarity.*

$$C = \mathcal{S}A\mathcal{S}^{-1}(1...M-1, 1...M-1)$$

where the matrix $\mathcal{S}$ has order M and is given by

$$\mathcal{S} = \begin{pmatrix} -1 & 1 & & & & \\ & -1 & 1 & & & \\ 0 & & \ddots & \ddots & 0 & \\ & & & \ddots & \ddots & \\ & & & & -1 & 1 \end{pmatrix} \tag{24}$$

and $\mathcal{S}^{-1}$ is the matrix

$$\mathcal{S}^{-1} = \begin{pmatrix} -1 & -1 & \cdots & & \cdots & -1 \\ & -1 & \cdots & & \cdots & -1 \\ & & \ddots & \ddots & \ddots & \vdots \\ & & & \ddots & \ddots & \vdots \\ & & & & -1 & -1 \\ & & & & & -1 \end{pmatrix} \tag{25}$$

Thus, the future behaviour of the difference process is related to the matrix C. We have

$$\Delta_{2i}^{k+1} = C_0 \Delta_i^k \tag{26}$$

$$\Delta_{2i+1}^{k+1} = C_1 \Delta_{i+1}^k \tag{27}$$

where $C_0 = C(1...M-2, 1...M-2)$ and $C_1 = C(2...M-1, 1...M-2)$.

Immediately we have the following

Corollary 14. *The spectra of the difference process matrices are the sets*

$$\sigma(C_0) = \sigma(A_0) \setminus \{\lambda_1\} \tag{28}$$

$$\sigma(C_1) = \sigma(A_1) \setminus \{\lambda_1\} \tag{29}$$

where λ_1 is the eigenvalue corresponding to the unit eigenvector $\mathbf{u}$.

Proof: *By definition, the matrix C has spectrum $\sigma(C) = \sigma(A) \setminus \{\lambda_1\}$. Furthermore, $\sigma(C_0)$ differs from $\sigma(C)$ by the eigenvalue 0 associated to the null eigenvector, while $\sigma(C_1) = \sigma(C_0) \setminus \{d_0\} = \sigma(C_0) \setminus \{\lambda_1, d_0\}$ that by using (17) and the fact that $d_0 = b_0$ gives (29).* ■

Corollary 15. *The process (10) converges uniformly to a continuous limit function if there exits a $L > 0$ s.t. for all $i_k \in \{0, 1\}$, $k = 1, .., L$ and all matrix norm we have*

$$\|C_{i_L} \cdots C_{i_1}\| \leq \gamma \quad \gamma \in [0, 1). \tag{30}$$

From (30) and Corollary 14 we have

Corollary 16. *A necessary condition for convergence is*

$$\rho(C_0) < 1 \quad and \quad \rho(C_1) < 1. \tag{31}$$

Iterating this idea, we can assume that the difference process can be differenced once again giving the process Δ^2. In order to do this we have to require that $\sum_{j=0}^{m} c(j) = \sum_{j=0}^{m} d(j)$ (this happens when the control matrices C_0 and C_1 have common eigenvector $\mathbf{u}$). Finally we quote the general result for second order differentiabilty of a subdivision process.

Theorem 17. *Suppose that the conditions (31) hold and*

$$\sum_{j=0}^{m} \mathbf{c}(j) = \sum_{j=0}^{m} \mathbf{d}(j).$$

Then the difference process Δ converges uniformly to zero if and only if the process Δ^2 converges uniformly to zero.

Corollary 18. *Suppose that the conditions (31) hold and that the process Δ^r, $r \geq 0$ exists. Then the subdivision scheme converges uniformly to a limit function that is continuous if and only if the r-th difference process Δ^r converges uniformly to zero.*

The previuos results say that the study of the convergence of the original process is carried out in terms of the control matrices associated to the difference process Δ^r. Since these matrices are of order $(M - r) \times (M - r)$, then the analysis should be easier for r close to M.

§4. The DIIM is a UDSS

There are many other results concerning dyadic subdivision scheme proved in [2]. Other results concern high order differentiability of general dyadic subdivision scheme. We have only mentioned a few of them in order to study the main characteristics of process DIIM. In this section we shall provide a connection between the results proved in Section 1 and 2, showing that the DIIM can be studied in the framework of general uniform subdivision scheme.

Let us start with the sequence $f_i^0 = f(t_i^0)$, with $t_i^0 \in D_0$, i.e. $t_i^0 = \frac{i}{2^0}$. The DIIM process (2) can be rewritten as follows

$$\begin{cases} f_{2i}^{k+1} = f_i^k \\ f_{2i+1}^{k+1} = \dfrac{1}{16}\left(-f_{i-1}^k + 9f_i^k + 9f_{i+1}^k - f_{i+2}^k\right) \end{cases} \tag{32}$$

with $k \geq 0$ and $i \in \mathbb{Z}$. Here $f_j^{k+1} = f\left(\dfrac{j}{2^{k+1}}\right)$.

4.1 Analysis of the continuity

From (32) we observe that the scheme is a subdivision process of order 4, that is $m = 3$, whose associated control vectors are

$$\begin{aligned} \mathbf{a} &= (0, 1, 0, 0)^T \\ \mathbf{b} &= (-\frac{1}{16}, \frac{9}{16}, \frac{9}{16}, -\frac{1}{16})^T. \end{aligned} \tag{33}$$

and the corresponding of (13) and (16) are

$$\begin{aligned} \mathbf{f}_{2i,5}^{k+1} &= A_0 \mathbf{f}_{i-1,5}^k \\ \mathbf{f}_{2i+1,5}^{k+1} &= A_1 \mathbf{f}_{i-1,5}^k. \end{aligned} \tag{34}$$

Since $a(3) = 0$, from (11), we get $M = 2m + 1 = 7$. Therefore the `generator matrix` A and its submatrices A_0 ad A_1 are

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \tag{35}$$

$$A_0 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} \end{pmatrix} \tag{36}$$

$$A_1 = \begin{pmatrix} -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{16} & \frac{9}{16} & \frac{9}{16} & -\frac{1}{16} \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{37}$$

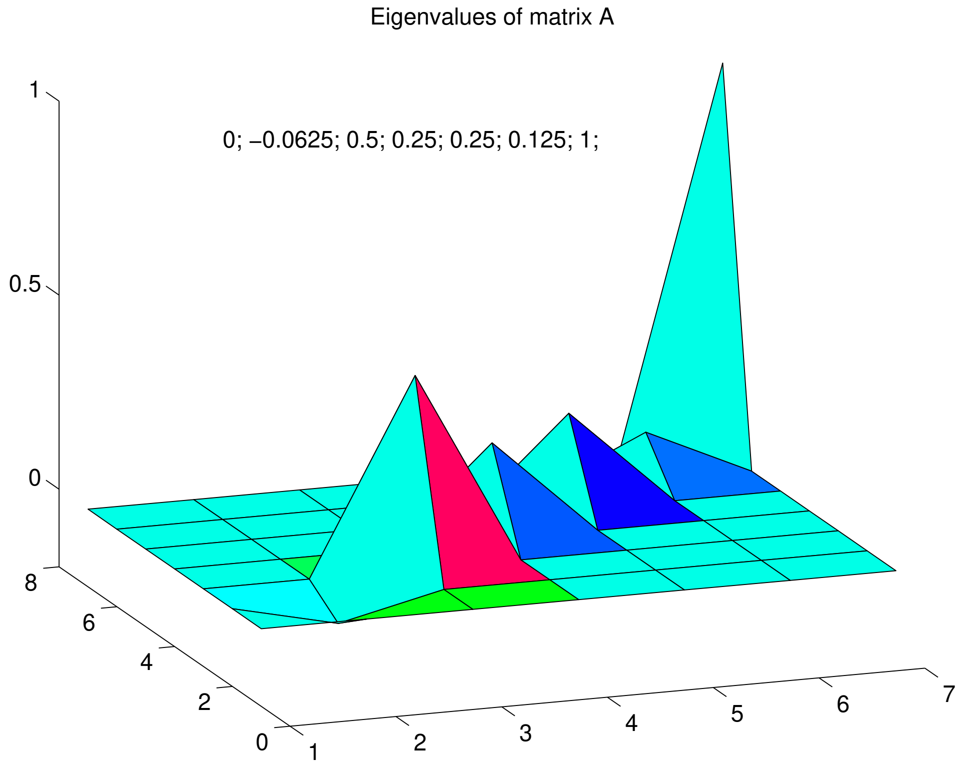

Fig. 3. The eigenvalues of the generator matrix A associated to the scheme (32).

The eigenvalues of these three matrices are displayed in Figures 3,4,5.

At first glance these eigenvalues are in modulus less than one. Furthermore we can characterized them in terms of the order of continuity of the limit function.

Proposition 19. *The eigenvalues of the matrices (35), (36) and (37) satisfy the following relations*

(a) $\lambda_{s+1} = \left(\frac{1}{2}\right)^s, \quad s = 0, 1, ..., S;$

(b) $|\lambda_s| \le \left(\frac{1}{2}\right)^S, \quad s = S+2, ..., M;$

where S stands for the `order of continuity` of the scheme.

From Section 2, we know that $S = 1$, since the scheme converges to a $\mathcal{C}^1$ function. From Fig. 3 we can also see that the eigenvalues of A, A_0, A_1 satisfy

$$\lambda_1 = 1, \quad \lambda_2 = \frac{1}{2}$$

$$|\lambda_3 \div \lambda_7| \le \frac{1}{2}.$$

and $\sigma(A_0) = \sigma(A_1)$.

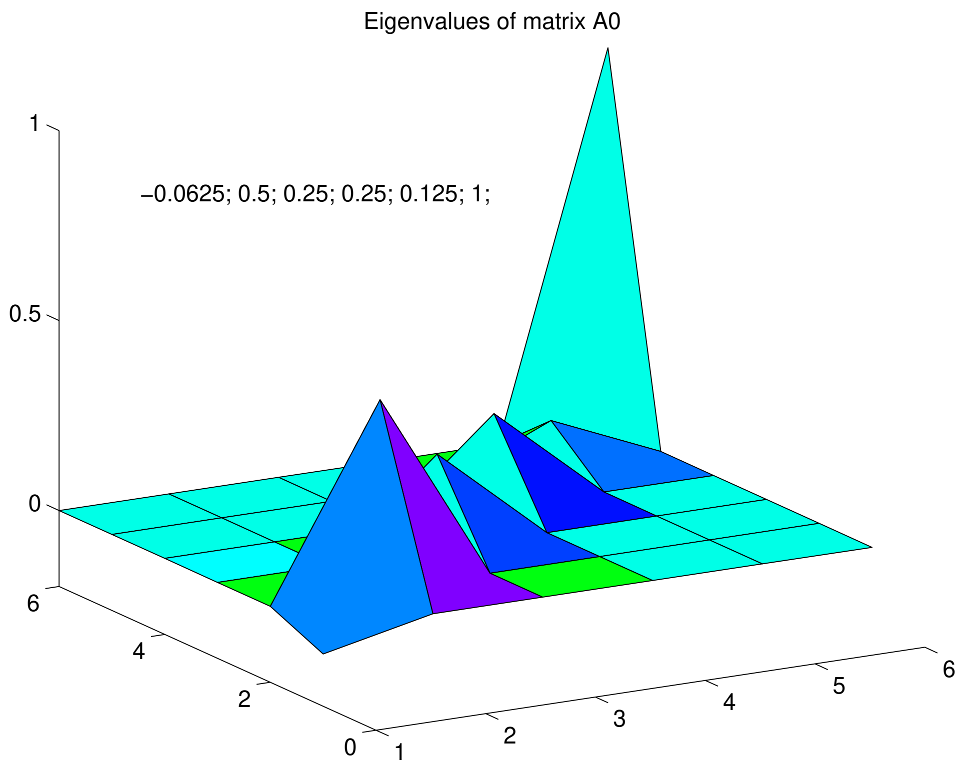

Fig. 4. The eigenvalues of the submatrix A_0 associated to the scheme (32).

We can also prove that the vectors of the coefficients

$$\sum_{j=0}^{m} \mathbf{a}(j) = \sum_{j=0}^{m} \mathbf{b}(j) = 1 \tag{38}$$

that is the process (32) satisfies the necessary condition for the uniform convergence to a $\mathcal{C}^0$ function on the real axis. Furthermore

$$\sum_{j=0}^{m} j(\mathbf{b}(j) - \mathbf{a}(j)) = \frac{1}{2}. \tag{39}$$

Putting together (38) and (39) we derive the necessary condition for the uniform convergence of the divided difference process on the dyadic numbers. Thus, our next step concerns the difference process in order to verify the convergence to a $\mathcal{C}^1$ limit function.

4.1 Analysis of first order differences

Using (24) the difference process is completely defined by the vectors

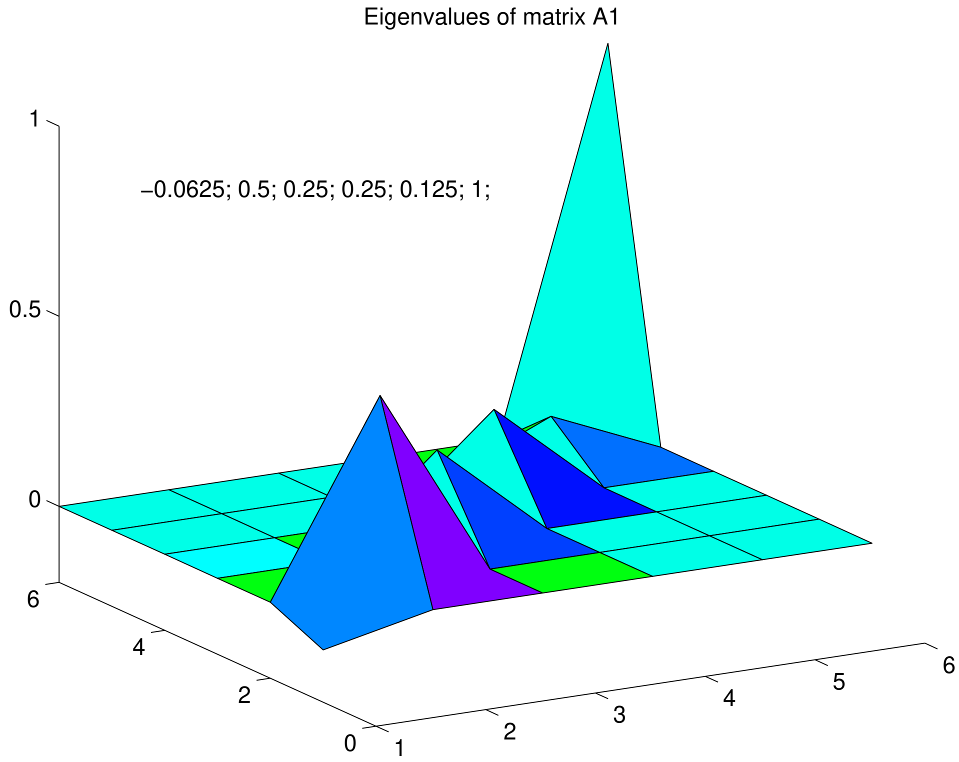

Fig. 5. The eigenvalues of the submatrix A_1 associated to the scheme (32).

$$\begin{aligned} \mathbf{c} &= (\frac{1}{16}, \frac{1}{2}, -\frac{1}{16}, 0)^T \\ \mathbf{d} &= (-\frac{1}{16}, \frac{1}{2}, \frac{1}{16}, 0)^T. \end{aligned} \tag{40}$$

The matrices C, C_0, C_1 and the corresponding spectra are displayed in Figures 6, 7, 8.

$$C = \begin{pmatrix} \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} & 0 & 0 & 0 \\ -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 & 0 & 0 \\ 0 & \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} & 0 & 0 \\ 0 & -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 & 0 \\ 0 & 0 & \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} & 0 \\ 0 & 0 & -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 \end{pmatrix} \tag{41}$$

$$C_0 = \begin{pmatrix} \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} & 0 & 0 \\ -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 & 0 \\ 0 & \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} & 0 \\ 0 & -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 \\ 0 & 0 & \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} \end{pmatrix} \tag{42}$$

$$C_1 = \begin{pmatrix} -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 & 0 \\ 0 & \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} & 0 \\ 0 & -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} & 0 \\ 0 & 0 & \frac{1}{16} & \frac{1}{2} & -\frac{1}{16} \\ 0 & 0 & -\frac{1}{16} & \frac{1}{2} & \frac{1}{16} \end{pmatrix} \tag{43}$$

As a further proof, these eigenvalues verify Corollary 14 and Corollary 16.

We observe that

1. The eigenvalues of C_0 are the same of C_1 and in modulus are strictly less than 1, that is Corollary 20holds.
2. $$2\sum_{j=0}^{m} \mathbf{c}(j) = 2\sum_{j=0}^{m} \mathbf{d}(j) = 1$$

This means that the scheme can be differentiated. Differentiating once more, it produces the following control vectors

$$\begin{aligned} \mathbf{f} &= (\frac{1}{8}, \frac{1}{8}, 0, 0)^T \\ \mathbf{g} &= (-\frac{1}{16}, \frac{6}{16}, -\frac{1}{16}, 0)^T. \end{aligned} \tag{44}$$

Furthermore, since

$$4\sum_{j=0}^{m} \mathbf{f}(j) = 4\sum_{j=0}^{m} \mathbf{g}(j) = 1 \tag{45}$$

we achieve the necessary conditions for the $\mathcal{C}^2$ continuity of the limit function. But as we have seen before the method generates interpolation functions that are not twice differentiable. This re-states that the above condition (45) is only necessary.

§5. Bidimensional setting

Let consider the function ϕ defined on the $\mathbb{Z}^2 = D_0 \times D_0$, that is

$$\begin{aligned} \phi : &\mathbb{Z}^2 \longrightarrow \mathbb{R} \\ (i, j) &\longmapsto \phi(i, j) = \phi^0_{i,j} \end{aligned} \tag{46}$$

The easiest way to extend the DIIM to the bidimensional setting is by using twice the DIIM, once in the x direction and once in the y direction. The only "problem" to solve is the computation of the barycentric point of each square. First we outline the algorithm, then we shall make a few remarks.

Algorithm 2

`Start with the sequence` $\phi(i,j) = \phi^0_{i,j}$ `given on the set` $D_0 \times D_0 = \mathbb{Z}^2$. `The aim is to compute` $\phi^k_{i,j}$ `on` $D^2_k = \{(t^k_i, t^k_j) = (\frac{i}{2^k}, \frac{j}{2^k})\}$.

for $k = 0, 1, ...$ **do** $\{k$ stands for the depth $\}$
`Work on the rows`

for $j \in \mathbb{Z}$ **do**
for $i \in \mathbb{Z}$ **do**
begin
$\phi^{k+1}_{2i,2j} = \phi^k_{i,j}$;
$\phi^{k+1}_{2i+1,2j} = \alpha_0\phi^k_{i-1,j} + \alpha_1\phi^k_{i,j} + \alpha_2\phi^k_{i+1,j} + \alpha_3\phi^k_{i+2,j}$;
end
`Work on the columns`

for $i \in \mathbb{Z}$ **do**
for $j \in \mathbb{Z}$ **do**
begin
$\phi^{k+1}_{2i,2j+1} = \alpha_0\phi^k_{i,j-1} + \alpha_1\phi^k_{i,j} + \alpha_2\phi^k_{i,j+1} + \alpha_3\phi^k_{i,j+2}$;
$A = \frac{1}{2}\left\{\phi^{k+1}_{2i+1,2j} + \phi^{k+1}_{2i+1,2j+2}\right\}$
$B = \frac{1}{2}\left\{\phi^{k+1}_{2i,2j+1} + \phi^{k+1}_{2i+2,2j+1}\right\}$
$\phi^{k+1}_{2i+1,2j+1} = \frac{1}{2}(A+B)$
end
end

The coefficients α_i are: $\alpha_0 = -\frac{1}{16}$, $\alpha_1 = \frac{9}{16}$, $\alpha_2 = \frac{9}{16}$ and $\alpha_3 = -\frac{1}{16}$.

Remarks

1. First of all we observe that the values on the sides of each subsquare, can be computed by using the matrices A_0 and A_1 exaclty in the same way we did for the univariate case. That is
$$\Phi^{k+1}_{2i,j} = A_0\Phi^k_{i,j}, \quad \Phi^{k+1}_{i,2j} = A_0\Phi^k_{i,j}$$
$$\Phi^{k+1}_{2i+1,j} = A_1\Phi^k_{i,j}, \quad \Phi^{k+1}_{i,2j+1} = A_1\Phi^k_{i,j}$$
where $\Phi^k_{i,j}$ corresponds to the vector $\mathbf{f}^k_i$ that we used in the univariate case.
2. The computation of the mid-point of each square can be done later since it depends only on the values computed in the current step. Moreover, for the value to give to this point we have chosen the average of the new four points that effect it. This assumption seems to be reasonable because the point is exactly in the barycentric position of the square and it is express as a convex combination of the mid-points on the sides of the square. This last observation can be used in preserving some characteristics of the original sequence (see [5] for a likewise scheme).

Acknowledgments. This work has been done with the support of CNR within "Progetto Strategico Sistema Lagunare Veneziano" and funds 40% of MURST on the project "Analisi Numerica e Matematica Computazionale".

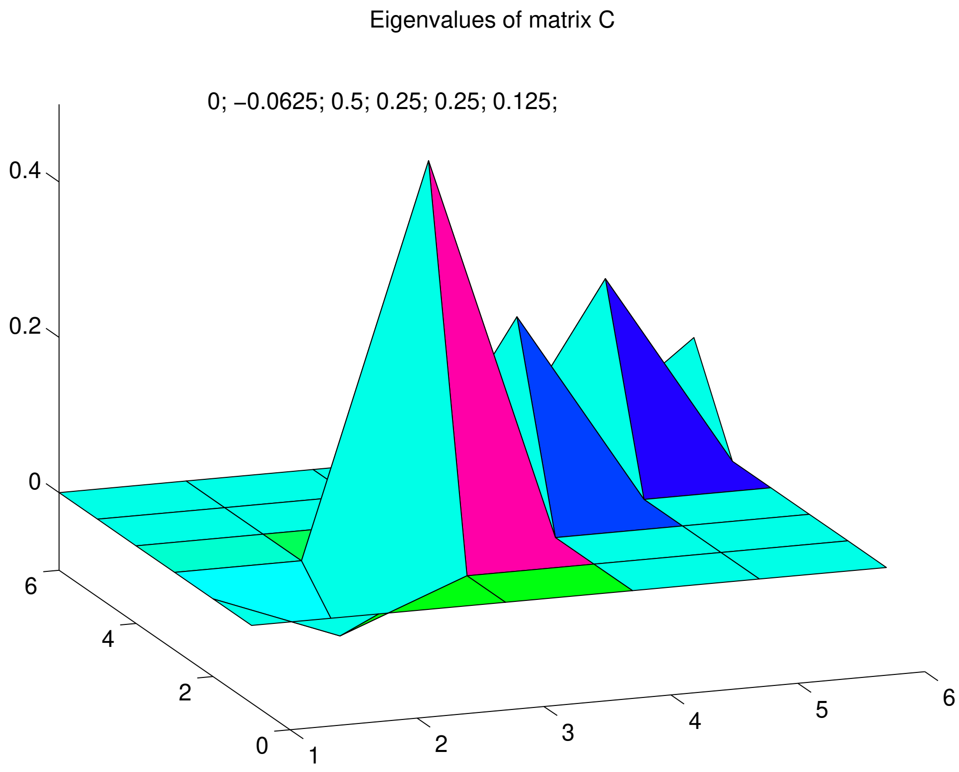

Fig. 6. The eigenvalues of the generator matrix C associated to the difference scheme (32).

References

1. Dubuc, S., Interpolation through an Iterative Scheme, Journal of Math. An. and Appl., 1986, 185–204.
2. Dyn, N., and Gregory, J. A., and Levin, D., *Analysis of Uniform Binary Subdivision Schemes for Curve Design*, Constr. Approx. **7** (1991), 127–147.
3. Dyn, N., and Sigalit, H., and Levin, D., *Subdivision Schemes for Surface Interpolation*, in *Workshop on Computational Geometry*, A. Conte et al. (eds), World Scientific, Singapore, 1993, 97–118.
4. Dyn, N., and Levin, D., *The Subdivision Experience*, in *Wavelets, Images and Surface Fitting*, P.-J. Laurent, A. Le Méhauté and L. L. Schumaker. (eds), A. K. Peters, Wellesley, 1994, 229–244.
5. Le Méhauté, A., and Utreras, F. I., Convexity-preserving interpolatory subdivision, Computer-Aided Geom. Design **11** (1994), 17–37.

Stefano De Marchi
Dipartimento di Matematica Pura ed Applicata
Università di Padova
Via Belzoni 7,

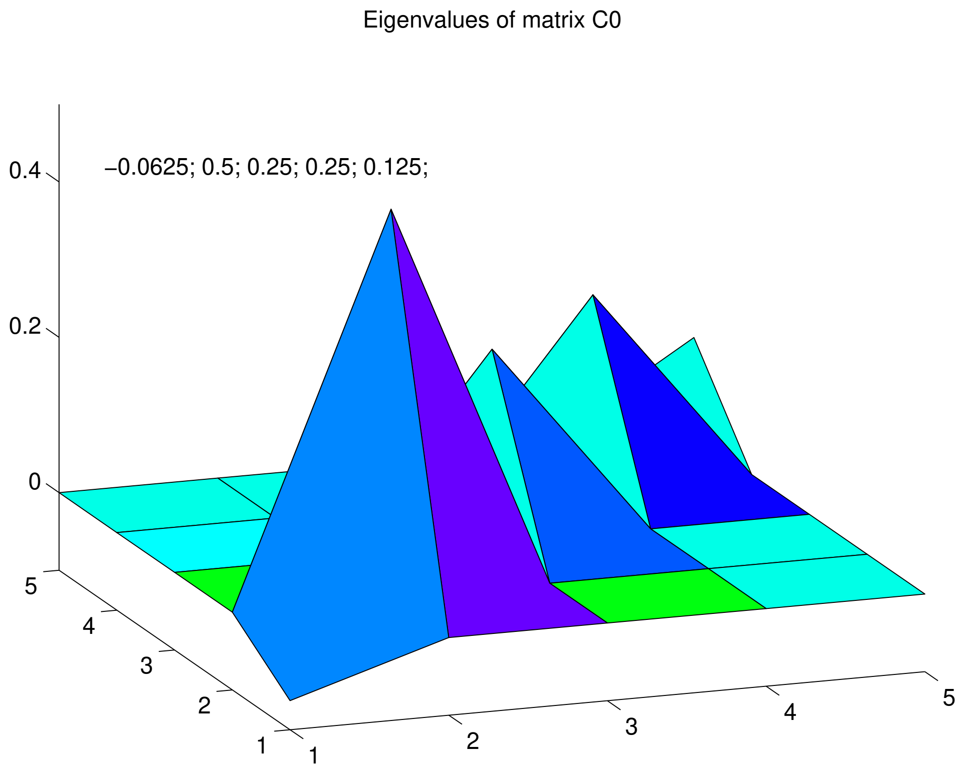

Fig. 7. The eigenvalues of the submatrix C_0.

I-35131 Padova, ITALY
stef@goedel.math.unipd.it

Maria Morandi Cecchi
Dipartimento di Matematica Pura ed Applicata
Università di Padova
Via Belzoni 7,
I-35131 Padova, ITALY
mcecchi@goedel.math.unipd.it

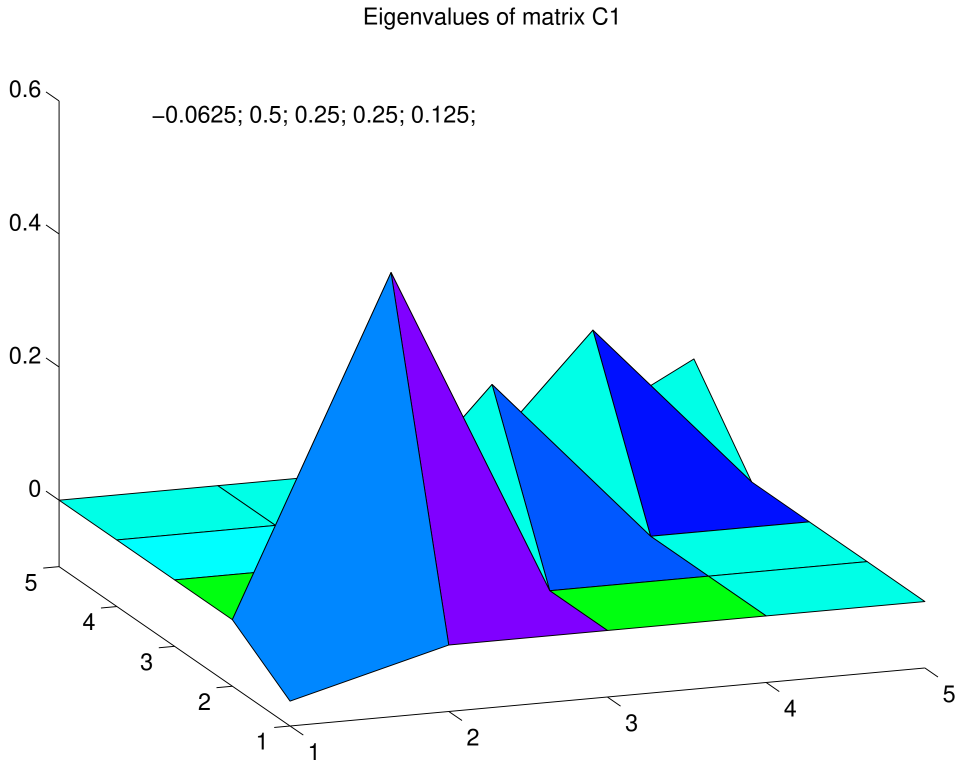

Fig. 8. The eigenvalues of the submatrix C_1.

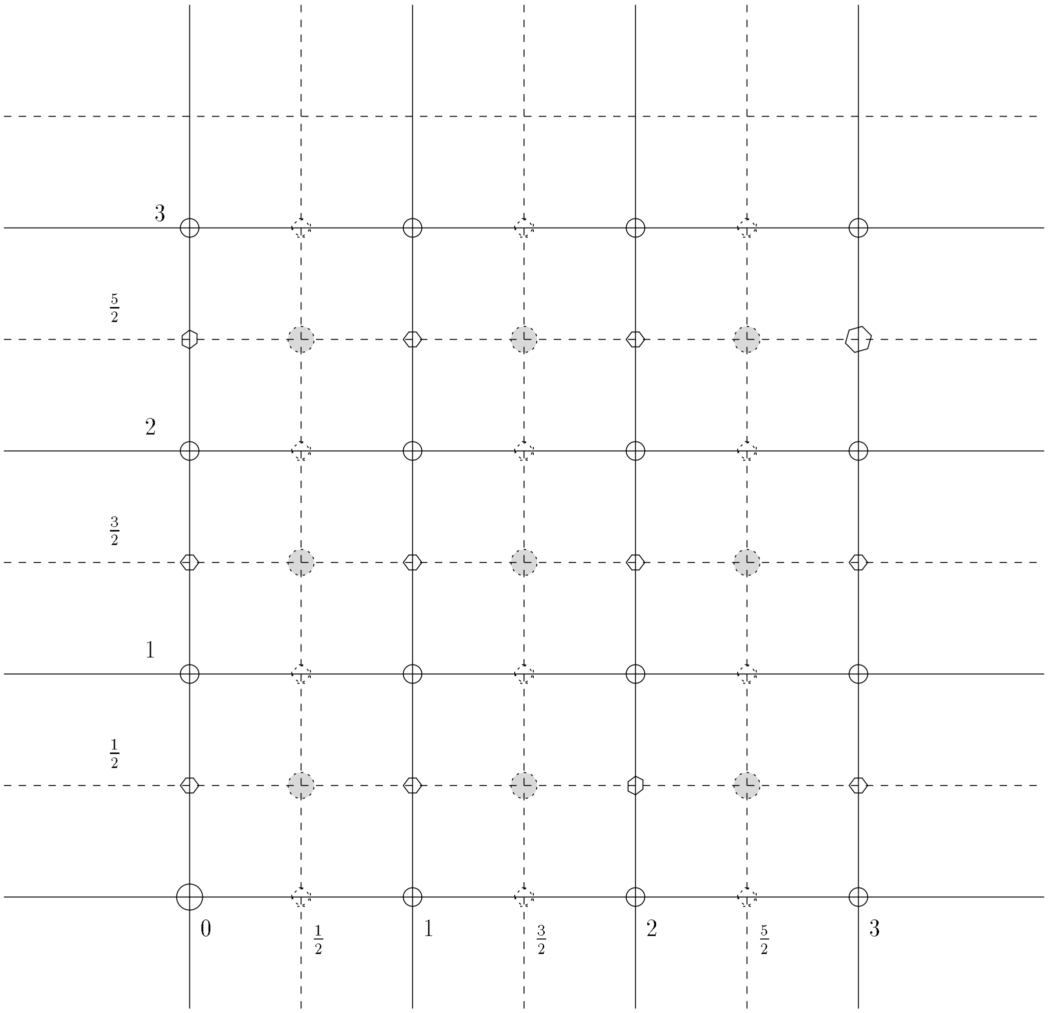

Fig. 9. Algorithm 2 after one step.