A RECURRENCE ALGORITHM FOR QUADRATIC HERMITE-PADÉ FORMS

by

R.G. Brookes and A.W. McInnes

No. 48 March, 1989.

Abstract

A simple recurrence algorithm to generate diagonal quadratic Hermite-Padé forms is presented. It requires $O(n^2)$ operations to calculate all the diagonal forms up to the $[n, n, n]$ form.

AMS Classification: 41A30, 41A21
Key words and phrases: Hermite-Padé approximation, quadratic function approximation, recurrence algorithm.

1 Introduction

A recurrence algorithm for calculating diagonal quadratic Hermite-Padé forms is derived. This algorithm has less restrictive conditions and applies to a larger class of functions than just those with normal systems as required by Paszkowski in [5], [6].

The Paszkowski algorithm is an algorithm for the more general algebraic Hermite-Padé form. It is a generalisation of the work of Padé [4], and the analysis of the algorithm is based on the Padé determinant [5]. Furthermore, the algorithm depends on the solution of a linear system of equations at each step.

Other recurrence relations have been derived by Della Dora and di Crescenzo [3], whose derivation is based on determinental identities, and by Borwein [1], [2], whose formulas are specific to the diagonal forms for exp(z) and log(z).

This algorithm is simply derived, is not complicated by the requirement to solve linear systems, and, since it requires less restrictive conditions applies to a much larger class of functions. The number of operations involved in each step is $O(n)$ (for the $[(n,n,n)]$ form), and hence the algorithm requires $O(n^2)$ operations to calculate all the diagonal forms up to the $[(n,n,n)]$ form. In §5, some examples are computed using the the algorithm. While this algorithm produces the correct results in symbolic computation, it should be noted that it appears to be unstable for numerical computation.

2 Notation

Let $f(x)$ be a given function with a known power series expansion about the origin.

(i) A quadratic Hermite-Padé form defined by

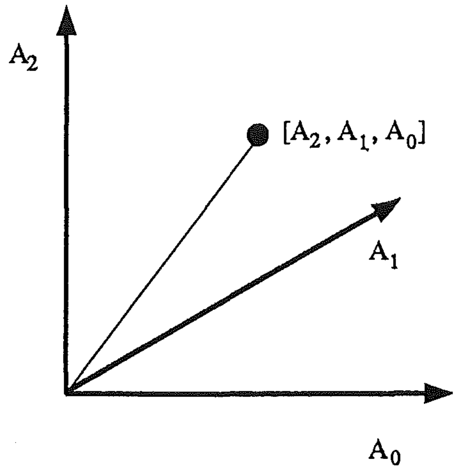
$$\sum_{i=0}^{2} a_i(x)f(x)^i = O(x^{A_0+A_1+A_2+2})$$

where $\deg(a_i(x)) \leq A_i$ will be expressed as $r(x,y) = a_2(x)y^2 + a_1(x)y + a_0(x)$ and often denoted by $[(A_2, A_1, A_0)]$.

(ii) The coefficient of $x^{A_0+A_1+A_2+2}$ in $\sum_{i=0}^{2} a_i(x)f(x)^i$ will be denoted by $E([(A_2, A_1, A_0)])$.

(iii) The coefficient of $x^i y^j$ in the polynomial $r(x,y)$ will be denoted by $\text{coeff}(r, x^i y^j)$.

(iv) To aid understanding we will imagine the quadratic Hermite-Padé forms to be elements of a 3–D lattice in the following arrangement.



It will be assumed throughout that the function $f(x)$ satisfies the following property.

Definition : Property A.

The function $f(x)$ satisfies :

(i) $E[(-1,0,0)] \neq 0$ for all such forms (where the only polynomial of degree -1 is the zero polynomial)

(ii) $E([(i,i,i)]), E([(i,i,i+1)]), E([(i,i+1,i+1)]) \neq 0$, for all possible such forms, $\forall i \in$ N.

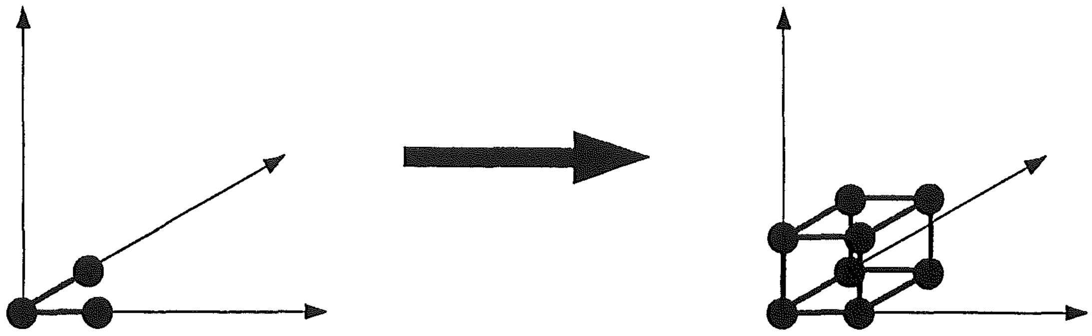
Note that normal systems (see Paszkowski [6]) satisfy the condition $E([(i,j,k)]) \neq 0, \forall i,j,k \in$ N, and hence are included in the class of systems with Property A.

3 Discussion

This section will discuss the basis of the algorithm.

Suppose $[(0,0,0)], [(0,1,0)], [(0,0,1)]$ forms and their corresponding order coefficients, E, are known. Then forms of the type $[(i,j,k)]$ can be found $\forall i,j,k \in \{0,1\}$.

i.e. If these three forms are known then the remainder of the "cube" can be generated.



(i) A [(1, 0, 0)] form.

Consider $r(x,y) = x[(0,0,0)] + \alpha_1[(0,1,0)] + \alpha_2[(0,0,1)]$ where :

α_1 is chosen so that $\mathrm{coeff}(r, xy) = 0$

α_2 is chosen so that $\mathrm{coeff}(r, x) = 0$.

Since $E([(0,0,0)]) \neq 0$ for all [(0,0,0)] forms the coefficient of xy in [(0,1,0)] and the coefficient of x in [(0,0,1)] are both non-zero, so such α_1, α_2 exist. This follows since if these coefficients were zero, then [(0,1,0)] and [(0,0,1)] would be [(0,0,0)] forms with order $O(x^3)$, which implies that $E([(0,0,0)]) = 0$ for these forms, contradicting Property A. Similarly, if the coefficient of y^2 in [(0,0,0)] were zero then this would be a [(-1,0,0)] form with order $O(x^2)$, again contradicting Property A. Hence $r(x,y) \not\equiv 0$ and since $r(x,y) = O(x^3)$, it follows that $r(x,y)$ is a $[(1,0,0)]$ form.

(ii) A [(1, 0, 1)] form .

Set $r(x,y) = [(1,0,0)] + \alpha_1[(0,0,1)]$ where α_1 is chosen so that $E([(1,0,0)]) + \alpha_1 E([(0,0,1)]) = 0$.

Since $E([(0,0,1)]) \neq 0$ such an α_1 exists and since $E([(0,0,0)]) \neq 0$ the coefficient of xy^2 in the [(1,0,0)] form is non-zero by the same argument as above, and so $r(x,y) \not\equiv 0$. It follows easily that $r(x,y)$ is a $[(1,0,1)]$ form.

(iii) A [(0, 1, 1)] form .

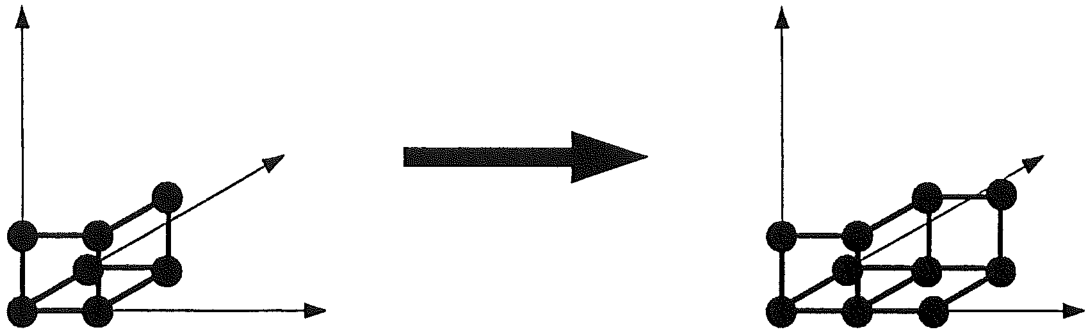
Set $[(0,1,1)] = [(0,1,0)] + \alpha_1[(0,0,1)]$ where α_1 is chosen so that $E([(0,1,0)]) + \alpha_1 E([(0,0,1)]) = 0$.

(iv) A [(1, 1, 1)] form .

Set $[(1,1,1)] = [(1,0,1)] + \alpha_1[(0,1,1)]$ where α_1 is chosen so that $E([(1,0,1)]) + \alpha_1 E([(0,1,1)]) = 0$.

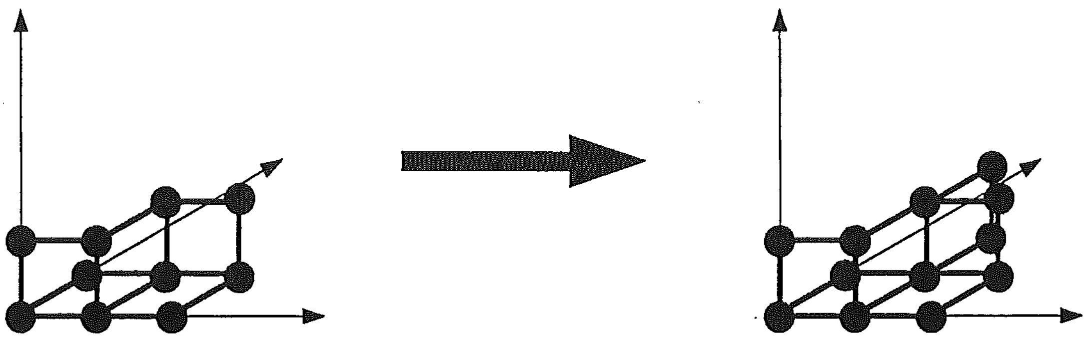
One can also generate a $[(1,1,0)]$ form in the same way although it will not be needed.

By symmetry it is clear that by using the same process the cube can be extended in the following way.



i.e. generate the forms $[(0,0,2)], [(1,1,2)], [(0,1,2)]$.

This can then be extended in the following way :



i.e. generate the forms $[(0,2,1)]$ and $[(1,2,1)]$.

It is now apparent that the initial pattern of three forms has been generated, but now centred at $(1,1,1)$ and that a simple recursion of this process will yield all the diagonal quadratic approximants. It is also clear that the number of operations involved at each step is only $O(n)$ so that it requires only $O(n^2)$ operations to generate the sequence $[(0,0,0)], [(1,1,1)], ..., [(n,n,n)]$.

4 Algorithm

Assume that the power series coefficients for $f(x)$ and $f(x)^2$ are given, and that the initial forms $[(0,0,0)], [(0,1,0)]$, and $[(0,0,1)]$ together with their corresponding order coefficients, E, have been independently calculated.

Step 1 :Set $[(0,1,1)] = [(0,1,0)] - \dfrac{E([(0,1,0)]}{E([(0,0,1)]}\,[(0,0,1)]$

$E([(0,1,1)]$ =coeff$([(0,1,1)], x^4)$

Set $n = 0$

Step 2 :Set $\mathbf{v} = (n,n,n), \mathbf{r} = (1,0,0), \mathbf{s} = (0,1,0), \mathbf{t} = (0,0,1)$

$z_1 = yx^n, z_2 = x^n, z_3 = x^{3n+3}$

Step 3 :Subroutine (see below)

Step 4 :Set $\mathbf{v} = (n,n,n+1), \mathbf{r} = (0,0,1), \mathbf{s} = (1,0,0), \mathbf{t} = (0,1,0)$

$z_1 = y^2x^n, z_2 = yx^n, z_3 = x^{3n+4}$

Step 5 :Subroutine

Step 6 :Set $\mathbf{v} = (n,n+1,n+1), \mathbf{r} = (0,1,0), \mathbf{s} = (0,0,1), \mathbf{t} = (1,0,0)$

$z_1 = x^{n+1}, z_2 = y^2x^n, z_3 = x^{3n+5}$

Step 7 :Subroutine

Step 8 :Set $n = n+1$

Go to Step 2

Subroutine

Step 1s :Set $[\mathbf{v}+\mathbf{r}] = x\,[\mathbf{v}] - \dfrac{\text{coeff}\,([\mathbf{v}], z_1)}{\text{coeff}\,([\mathbf{v}+\mathbf{s}], xz_1)}\,[\mathbf{v}+\mathbf{s}] - \dfrac{\text{coeff}\,([\mathbf{v}], z_2)}{\text{coeff}\,([\mathbf{v}+\mathbf{t}], xz_2)}\,[\mathbf{v}+\mathbf{t}]$

$E([\mathbf{v}+\mathbf{r}])$ =coeff$([\mathbf{v}+\mathbf{r}], z_3)$

Step 2s :Set $[\mathbf{v}+\mathbf{r}+\mathbf{t}] = [\mathbf{v}+\mathbf{r}] - \dfrac{E([\mathbf{v}+\mathbf{r}])}{E([\mathbf{v}+\mathbf{t}])}\,[\mathbf{v}+\mathbf{t}]$

$E([\mathbf{v}+\mathbf{r}+\mathbf{t}])$ =coeff$([\mathbf{v}+\mathbf{r}+\mathbf{t}], xz_3)$

Step 3s :Set $[\mathbf{v}+\mathbf{r}+\mathbf{s}+\mathbf{t}] = [\mathbf{v}+\mathbf{r}+\mathbf{t}] - \dfrac{E([\mathbf{v}+\mathbf{r}+\mathbf{t}])}{E([\mathbf{v}+\mathbf{s}+\mathbf{t}])}\,[\mathbf{v}+\mathbf{s}+\mathbf{t}]$

$E([\mathbf{v}+\mathbf{r}+\mathbf{s}+\mathbf{t}])$ =coeff$([\mathbf{v}+\mathbf{r}+\mathbf{s}+\mathbf{t}], x^2z_3)$

Note that coeff([v + s], xz_1) $\neq 0$, since if it were zero then [v + s] would be a [v] form with $E([\mathrm{v}]) = 0$, contradicting Property A. Similarly coeff([v + t]) $\neq 0$, and hence the conditions of Property A are sufficient for the operation of this algorithm.

5 Examples

The algorithm has been used, coded in MACSYMA, on the functions $\exp(z), \log(1+z), \sin(z)$ and has been found to work satisfactorily. Following are the calculations for Steps 1–3 in some detail for $f(x) = \exp(-x)$ (note that the forms have been normalised at each step to give integer coefficients).

Given

$[0,0,0] = y^2 - 2y + 1$

$[0,1,0] = y^2 + 2xy - 1$

$[0,0,1] = y^2 - 4y - 2x + 3$

and

E([(0,1,0)]) = -1/3

E([(0,0,1)]) = -2/3

Then

Step 1

$[0,1,1] = 2[y^2 + 2xy - 1 - \frac{1/3}{2/3}\,(y^2 - 4y - 2x + 3)] = y^2 + (4x+4)y + 2x - 5$

$E([0,1,1)] = 1/6$

Step 1s

$[1,0,0] = 2[x(y^2 - 2y + 1) + (y^2 + 2xy - 1) + (1/2)(y^2 - 4y - 2x + 3)] = (2x+3)y^2 - 4y + 1$

$E([1,0,0)]) = 2/3$

Step 2s

$[1,0,1] = (1/2)[(2x+3)y^2 - 4y + 1 + \frac{2/3}{2/3}\,(y^2 - 4y - 2x + 3)] = (x+2)y^2 - 4y - x + 2$

$E([1,0,1]) = -1/6$

Step 3s

$$[1,1,1] = (x+2)y^2 - 4y - x + 2 + \frac{1/6}{1/6}\,(y^2 + (4x+4)y + 2x - 5) = (x+3)y^2 + 4xy + x - 3$$

$E([1,1,1)] = 1/30$

When the calculations are performed numerically rather than symbolically however, a rapid build up of calculation error is evident. Several diagonal forms for $\log(1+z)$ are given here, calculated both exactly using MACSYMA and numerically using PC-Matlab with 16 digit precision. Although all the values are available to at least 16 digits only enough precision is given to indicate the numerical error. Note also that the normalisation $\|[n,n,n]\|_2 = 1$ is used (where $\|[n,n,n]\|_2$ means the ℓ_2 norm of the vector formed from the coefficients of $[n,n,n]$).

The $[2,2,2]$ form

Coefficient	PC-Matlab	MACSYMA
y^2	$-1.84812331090 \times 10^{-1}$	$1.84812331090 \times 10^{-1}$
xy^2	$-1.84812331090 \times 10^{-1}$	$-1.84812331090 \times 10^{-1}$
x^2y^2	$3.08020551818 \times 10^{-2}$	$3.08020551816 \times 10^{-2}$
y	2.90×10^{-12}	0
xy	$-5.54436993269 \times 10^{-1}$	$-5.54436993270 \times 10^{-1}$
x^2y	$-2.77218496635 \times 10^{-1}$	$-2.77218496635 \times 10^{-1}$
1	0	0
x	-2.90×10^{-12}	0
x^2	$7.39249324361 \times 10^{-1}$	$7.39249324360 \times 10^{-1}$

The $[4,4,4]$ form

Coefficient	PC-Matlab	MACSYMA
y^2	6.10111 $\times 10^{-2}$	6.10109 $\times 10^{-2}$
xy^2	1.22022 $\times 10^{-1}$	1.22022 $\times 10^{-1}$
x^2y^2	2.03374 $\times 10^{-2}$	2.03369 $\times 10^{-2}$
x^3y^2	-4.06739 $\times 10^{-2}$	-4.06739 $\times 10^{-2}$
x^4y^2	6.77897 $\times^{-4}$	6.77898 $\times 10^{-4}$
y	-1.56 $\times 10^{-6}$	0
xy	3.99958 $\times 10^{-1}$	3.99960 $\times 10^{-1}$
x^2y	5.99940 $\times 10^{-1}$	5.99940 $\times 10^{-1}$
x^2y	1.83033 $\times 10^{-1}$	1.83033 $\times 10^{-1}$
x^4y	-8.47371 $\times 10^{-3}$	-8.47373 $\times 10^{-3}$
1	0	0
x	1.56 $\times 10^{-6}$	0
x^2	-4.60970 $\times 10^{-1}$	-4.60971 $\times 10^{-1}$
x^3	-4.60972 $\times 10^{-1}$	-4.60971 $\times 10^{-1}$
x^4	2.93756 $\times 10^{-2}$	2.93756 $\times 10^{-2}$

The $[5,5,5]$ form

Coefficient	PC-Matlab	MACSYMA
y^2	3.76×10^{-5}	·0
xy^2	6.672×10^{-2}	6.660×10^{-2}
x^2y^2	1.333×10^{-1}	1.332×10^{-1}
x^3y^2	5.392×10^{-2}	5.391×10^{-2}
x^4y^2	-1.270×10^{-2}	-1.269×10^{-2}
x^5y^2	1.059×10^{-4}	1.057×10^{-4}
y	-2.541×10^{-1}	-2.540×10^{-1}
xy	-6.349×10^{-1}	-6.349×10^{-1}
x^2y	-3.445×10^{-1}	-3.448×10^{-1}
x^3y	1.180×10^{-1}	1.177×10^{-1}
x^4y	7.874×10^{-2}	7.866×10^{-2}
x^5y	-1.451×10^{-3}	-1.448×10^{-3}
1	0	0
x	2.540×10^{-1}	2.540×10^{-1}
x^2	5.079×10^{-1}	5.080×10^{-1}
x^3	4.503×10^{-2}	4.544×10^{-2}
x^4	-2.087×10^{-1}	-2.085×10^{-1}
x^5	5.435×10^{-3}	5.424×10^{-3}

6 Conclusion

Numerical problems aside, this algorithm gives a recursive procedure for calculating quadratic Hermite-Padé forms for many functions. The algorithm operates successfully for functions whose systems are not necessarily normal, such as $\log(1+x)$ and $\sin(x)$.

Although a procedure for calculating diagonal forms is given, it is clear that the algorithm could be easily generalised to achieve any [(l,m,n)] form by constructing the cubes in an appropriate manner to reach this point in the lattice. Similar concepts are mentioned by Padé [4], and Paszkowski [5], [6]. Property A would have to be modified in this case, to take account of the path through the lattice, but it is clear that a full normal system would not be required unless every point in the lattice was to be computed.

It is also clear that these principles could easily be extended to the more general algebraic Hermite-Padé forms studied by Paszkowski [6].

It is of interest to note that the relationships obtained in the algorithm generalise in a natural way some of the Frobenius identities familiar in Padé approximation, as was noted by Paszkowski [6].

7 References

1. P.B. Borwein, (1986) : *Quadratic Hermite-Padé Approximation to the Exponential Function.* Constr. Approx. **2**:291–302.
2. P.B. Borwein, (1987) : *Quadratic and Higher Order Padé Approximants.* In: Colloquia Mathematica Societatis Janos Bolyai, **49** Alfred Haar Memorial Conference, Budapest (Hungary) 1985, : 213–224.
3. J. Della Dora, C. di Crescenzo, (1984) : *Approximants de Padé-Hermite.* Numer. Math. **43**:23–57.
4. H. Padé, (1894) : *Sur la généralisation des fractions continues algebriques.* J. Pures et Appl. **10**:291–329.
5. S. Paszkowski, (1982) : *Quelques Algorithms de l'Approximation de Padé-Hermite.* Publ. ANO-89, Univ. Lille I.
6. S. Paszkowski, (1987) : *Recurrence Relations in Padé-Hermite Approximation.* J. Comput. Appl. Math. **19**:99–107.