A tutorial on Padé approximation, with applications to control

First part

Claude Brezinski

Université de Lille, France

- Introduction
- Padé-type approximants
- Padé approximants
- Recursive computation
- Error estimation
- Convergence
- Gibbs phenomenon
- Generalizations
- Continued fractions
- Applications to control
- Henri Padé

Introduction

Various approaches :

- ➜ direct approach : Johann Henrich Lambert, 1758
- ➜ via continued fractions : Joseph Louis Lagrange, 1776
- ➜ via **formal quadrature methods**

Let us consider the **formal power series**

$$f(t) = c_0 + c_1 t + c_2 t^2 + \cdots$$

Padé approximation consists in constructing a rational function whose series expansion in ascending power of t agrees with f up to a certain degree.

Let c be the **linear functional** on the space of polynomials defined by

$$c(x^i) = \begin{cases} c_i, & i = 0, 1, \cdots \\ 0, & i < 0. \end{cases}$$

The **main** result that will be used is a very simple one :

we **formally** have ($\mathbf{c}$ acts on $\mathbf{x}$ and $\mathbf{t}$ is a parameter)

$$
\begin{aligned}
c\left(\frac{1}{1-xt}\right) &= c(1+xt+x^2t^2+\cdots) \\
&= c(1)+c(x)t+c(x^2)t^2+\cdots \\
&= c_0+c_1t+c_2t^2+\cdots \\
&= f(t).
\end{aligned}
$$

We want to obtain an **approximation** of $\mathbf{c(g(x))}$ where $\mathbf{g}$ depends on $\mathbf{t}$ and is given by

$$\mathbf{g(x)=\frac{1}{1-xt}}$$

A similar situation is well known in numerical analysis : **numerical quadrature** where we have to obtain an approximation of

$$c(\mathbf{g}(\mathbf{x})) = \int_a^b \mathbf{g}(\mathbf{x})\mathbf{w}(\mathbf{x})\,dx.$$

A numerical quadrature method consists in replacing $\mathbf{g}$ by an **interpolation polynomial** $\mathbf{P}$ at $\mathbf{k}$ (usually distinct) points and to integrate it, that is computing $\mathbf{c}(\mathbf{P}(\mathbf{x})) \simeq \mathbf{c}(\mathbf{g}(\mathbf{x}))$.

Such an **interpolation–type quadrature formula** is exact on $\mathcal{P}_{\mathbf{k}}$.

If the interpolation points are the zeros of the **orthogonal polynomial** of degree $\mathbf{k}$ with respect to **w(x)** on $[a, b]$, the formula is exact on $\mathcal{P}_{\mathbf{2k-1}}$.

In this case, it is called a **Gaussian quadrature formula**.

The same idea will now be applied with

$$\mathbf{g(x) = c\left(\frac{1}{1-xt}\right)}$$

thus leading to **formal** quadrature formulae

$$\mathbf{c(P(x)) \simeq c\left(\frac{1}{1-xt}\right) = f(t).}$$

Padé–type approximants

Let $\mathbf{v_k}$ be any polynomial of **degree** k.

The Hermite interpolation polynomial $\mathbf{P}$ of $\mathbf{g(x) = 1/(1 - xt)}$ at the zeros of v_k is given by

$$\mathbf{P(x) = \frac{1}{1 - xt}\left(1 - \frac{v_k(x)}{v_k(1/t)}\right)}$$

Indeed, P is a polynomial of degree $k - 1$ and, if

$$v_k(x) = (x - x_1)^{k_1} \cdots (x - x_n)^{k_n}$$

with $k_1 + \cdots + k_n = k$, then

$$P^{(j)}(x_i) = \left.\frac{d^j}{dx^j}(1 - xt)^{-1}\right|_{x=x_i}$$

for $i = 1, \ldots, n$ and $j = 0, \ldots, k_i - 1$.

We will now approximate $f(t) = c(1/(1-xt))$ by $c(P)$.

We have

$$c(P) = \frac{1}{v_k(1/t)} c\left(\frac{v_k(1/t) - v_k(x)}{1 - xt}\right). \qquad (1)$$

Let us set

$$w_k(t) = c\left(\frac{v_k(x) - v_k(t)}{x - t}\right).$$

w_k is a polynomial of degree $k-1$ in t and we have

$$\mathbf{c(P) = \frac{\widetilde{w}_k(t)}{\widetilde{v}_k(t)}}$$

where $\mathbf{\widetilde{w}_k(t) = t^{k-1} w_k(1/t)}$ and $\mathbf{\widetilde{v}_k(t) = t^k v_k(1/t)}$ (reversal of the coefficients).

So, $\mathbf{c(P)}$ is a **rational function** with a numerator of degree at most $\mathbf{k-1}$ and a denominator of degree at most $\mathbf{k}$.

From (1), we obtain

$$
\begin{aligned}
\mathbf{c(P)} &= \frac{\widetilde{w}_k(t)}{\widetilde{v}_k(t)} \\
&= c(1/(1-xt)) - \frac{t^k}{\widetilde{v}_k(t)} c(v_k(x)/(1-xt)) \\
&= \mathbf{f(t)} + \mathcal{O}(\mathbf{t^k}).
\end{aligned}
$$

The rational function $c(P) = \widetilde{w}_k(t)/\widetilde{v}_k(t)$ is called a

Padé–type approximant of f

and it is denoted by

$$(\mathbf{k-1/k})_{\mathbf{f}}(\mathbf{t})$$

$\mathbf{v_k}$ is called the **generating polynomial** of the approximant.

It is possible to construct PTA with arbitrary degrees by setting

$$f(t) = c_0 + \cdots + c_n t^n + t^{n+1} \underbrace{(c_{n+1} + c_{n+2} t + \cdots)}_{f_n}.$$

Then, we will obtain

$$\begin{aligned} c_0 + \cdots + c_n t^n + t^{n+1} (k-1/k)_{f_n}(t) = \\ f(t) + \mathcal{O}(t^{n+k+1}) = \mathbf{(n+k/k)_f(t)}. \end{aligned}$$

Similarly, we set

$$f_{-n}(t) = 0 + 0t + \cdots + 0t^{n-2} + t^{n-1} f(t)$$

and

$$\begin{aligned} \mathbf{(k/n+k)_f(t)} &= t^{-n+1} (n+k-1/n+k)_{f_{-n}}(t) \\ &= f(t) + \mathcal{O}(t^{k+1}). \end{aligned}$$

Thus, we are able to construct all the PTAs (p/q), $p, q \geq 0$ with

$$\mathbf{(p/q)_f(t) = f(t) + \mathcal{O}(t^{p+1})}.$$

The computation of $\mathbf{(p/q)}$ needs the knowledge of $\mathbf{c_0, \ldots, c_p}$.

✘ **Difficult problem :** choice of v_k.

But a **good choice** can lead to **better convergence and approximation results** than with Padé approximants **(Eiermann, 1984)**.

Padé approximants

Gaussian quadrature methods are obtained when the interpolation points are the zeros of some family of **orthogonal polynomials**.

✘ Here, the same situation holds.

We already saw that

$$(k-1/k)_f(t) = f(t) - \frac{t^k}{\widetilde{v}_k(t)} c\left(\frac{v_k(x)}{1-xt}\right).$$

But

$$c\left(\frac{v_k(t)}{1-xt}\right) = c\left(v_k(x)(1+xt+\cdots+x^{k-1}t^{k-1}+\frac{x^k t^k}{1-xt}\right).$$

So, if we choose v_k such that

$$\mathbf{c(x^i v_k(x)) = 0} \text{ for } \mathbf{i = 0, 1, \ldots} \tag{2}$$

the first terms in the preceding formula will be cancelled out, and the order of approximation will be improved.

Since v_k has degree k, it has $k+1$ unknown coefficients.
But, on the other hand, a rational function is defined apart a multiplying factor.
So, we can only impose $\mathbf{k}$ conditions (2) on v_k, that is for $\mathbf{i = 0, \ldots, k-1}$.

The family of polynomials satisfying the conditions (2) for $\mathbf{i = 0, \ldots, k-1}$ is called the family of **formal orthogonal polynomials** with respect to $\mathbf{c}$. They will be denoted by $\mathbf{P_k}$ instead of v_k.

In this case, the approximant $\widetilde{\mathbf{w}}_{\mathbf{k}}(\mathbf{t})/\widetilde{\mathbf{v}}_{\mathbf{k}}(\mathbf{t})$ will be called the

Padé approximant of f

and it will be denoted by (notice the **square** brackets)

$$[\mathbf{k}-\mathbf{1}/\mathbf{k}]_{\mathbf{f}}(\mathbf{t})$$

We have

$$[k-1/k]_f(t) = f(t) + \mathcal{O}(t^{2k}).$$

Similarly $[p/q]_f(t) = f(t) + \mathcal{O}(t^{p+q+1})$, instead of $\mathcal{O}(t^{p+1})$ for PTA.

The computation of $[\mathbf{p}/\mathbf{q}]$ needs the knowledge of $\mathbf{c_0}, \ldots, \mathbf{c_{p+q}}$ (instead of $c_0, \ldots, c_p$ for (p/q)).

Expression of Padé approximants

Let us set

$$[p/q]_f(t) = N(t)/D(t)$$

with

$$\begin{aligned} N(t) &= a_0 + a_1 t + \cdots + a_p t^p \\ D(t) &= b_0 + b_1 t + \cdots + b_q t^q. \end{aligned}$$

Then, we have

$$N(t) - D(t) f(t) = \mathcal{O}(t^{p+q+1}).$$

Identifying the coefficients of the powers of t, we get

$$\begin{aligned}
a_0 &= c_0 b_0 \\
a_1 &= c_1 b_0 + c_0 b_1 \\
&\vdots \\
a_p &= c_p b_0 + c_{p-1} b_1 + \cdots + c_{p-q} b_q
\end{aligned}$$

$$\begin{aligned}
0 &= c_{p+1} b_0 + c_p b_1 + \cdots + c_{p-q+1} b_q \\
&\vdots \\
0 &= c_{p+q} b_0 + c_{p+q-1} b_1 + \cdots + c_p b_q
\end{aligned}$$

with $c_i = 0$ for $i < 0$.

The **last q equations** contain $q + 1$ unknowns.

Since a rational function is defined apart a multiplying factor, we will take $\mathbf{b_0 = 1}$ and solve the remaining $\mathbf{q \times q}$ system (assuming that it is nonsingular). It gives $\mathbf{b_1, \ldots, b_q}$.

Then, the coefficients $\mathbf{a_0, \ldots, a_p}$ of the numerator are obtained directly by the **first** $p+1$ **equations**.

The following determinantal expression holds (**Jacobi, 1846**).

$$[p/q]_f(t) = \frac{\begin{vmatrix} z^q f_{p-q}(t) & z^{q-1} f_{p-q+1}(t) & \cdots & f_p(t) \\ c_{p-q+1} & c_{p-q+2} & \cdots & c_{p+1} \\ \vdots & & & \vdots \\ c_p & c_{p+1} & \cdots & c_{p+q} \end{vmatrix}}{\begin{vmatrix} t^q & t^{q-1} & \cdots & 1 \\ c_{p-q+1} & c_{p-q+2} & \cdots & c_{p+1} \\ \vdots & & & \vdots \\ c_p & c_{p+1} & \cdots & c_{p+q} \end{vmatrix}}$$

with

$$\begin{aligned} f_k(t) &= \sum_{i=0}^{k} c_i t^i \quad \text{for } k \geq 0, \\ &= 0 \qquad\qquad \text{for } k < 0. \end{aligned}$$

Padé approximants are displayed in a double array called the **Padé table**

$$\begin{array}{cccc} [0/0] & [0/1] & [0/2] & \cdots \\ [1/0] & [1/1] & [1/2] & \cdots \\ [2/0] & [2/1] & [2/2] & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{array}$$

Let us give a first example.

We consider the **Stieltjes** series

$$\ln(1+t) = t - \frac{t^2}{2} + \frac{t^3}{3} - \cdots$$

We denote by $\mathbf{S_i(t)}$ its **partial sum** up to the term of degree $\mathbf{i+1}$ inclusively.

The series converges for t complex, $|t| \leq 1, t \neq -1$.

The computation of $\mathbf{S_{2k}}$ and $\mathbf{[k/k]}$ both **need** the knowledge of

$$\mathbf{c_0 = 0, c_1, \cdots, c_{2k+1}}$$

$$\ln 2 = 0.6931471805599453\ldots$$

k	$S_{2k}(1)$	$[k/k]_f(1)$
1	0.833	0.7
2	0.783	0.6933
3	0.759	0.693152
4	0.745	0.69314733
5	0.736	0.6931471849
6	0.730	0.693147180688
7	0.725	0.69314718056363
8	0.721	0.6931471805600 0
9	0.718	0.6931471805599485
10	0.716	0.6931471805599454

$$\ln 3 = 1.098612288668110\ldots$$

k	$S_{2k}(2)$	$[k/k]_f(2)$
1	$0.266.10^1$	1.14
2	$0.506.10^1$	1.101
3	$0.126.10^2$	1.0988
4	$0.375.10^2$	1.098625
5	$0.121.10^3$	1.0986132
6	$0.410.10^3$	1.09861235
7	$0.142.10^4$	1.098612293
8	$0.504.10^4$	1.0986122890
9	$0.181.10^5$	1.098612288692
10	$0.655.10^5$	1.0986122886698

Recursive computation of Padé approximants

Let $\mathbf{c}^{(\mathbf{n})}$ be the linear functional on the space of polynomials defined by

$$\mathbf{c}^{(\mathbf{n})}(\mathbf{x}^{\mathbf{i}}) = \mathbf{c}_{\mathbf{n}+\mathbf{i}}$$

and let $\left\{\mathbf{P}_{\mathbf{k}}^{(\mathbf{n})}\right\}$ be the family of **FOP** with respect to $\mathbf{c}^{(\mathbf{n})}$.

Then

$$[\mathbf{p}/\mathbf{q}]_{\mathbf{f}}(\mathbf{t}) = \sum_{\mathbf{i}=0}^{\mathbf{p}-\mathbf{q}} \mathbf{c}_{\mathbf{i}}\mathbf{t}^{\mathbf{i}} + \mathbf{t}^{\mathbf{p}-\mathbf{q}+1} \frac{\widetilde{\mathbf{Q}}_{\mathbf{q}}^{(\mathbf{p}-\mathbf{q}+1)}(\mathbf{t})}{\widetilde{\mathbf{P}}_{\mathbf{q}}^{(\mathbf{p}-\mathbf{q}+1)}(\mathbf{t})}$$

where

$$\begin{aligned}
\widetilde{P}_q^{(p-q+1)}(t) &= t^q P_q^{(p-q+1)}(1/t) \\
\widetilde{Q}_q^{(p-q+1)}(t) &= t^{q-1} Q_q^{(p-q+1)}(1/t) \\
Q_q^{(p-q+1)}(t) &= c^{(p-q+1)}\left(\frac{P_q(x) - P_q(t)}{x - t}\right).
\end{aligned}$$

These polynomials are displayed in a table

$$\begin{array}{cccc} P_{-1}^{(0)} & P_0^{(-1)} & P_1^{(-2)} & \cdots \\ P_{-1}^{(1)} & P_0^{(0)} & P_1^{(-1)} & \cdots \\ P_{-1}^{(2)} & P_0^{(1)} & P_1^{(0)} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{array}$$

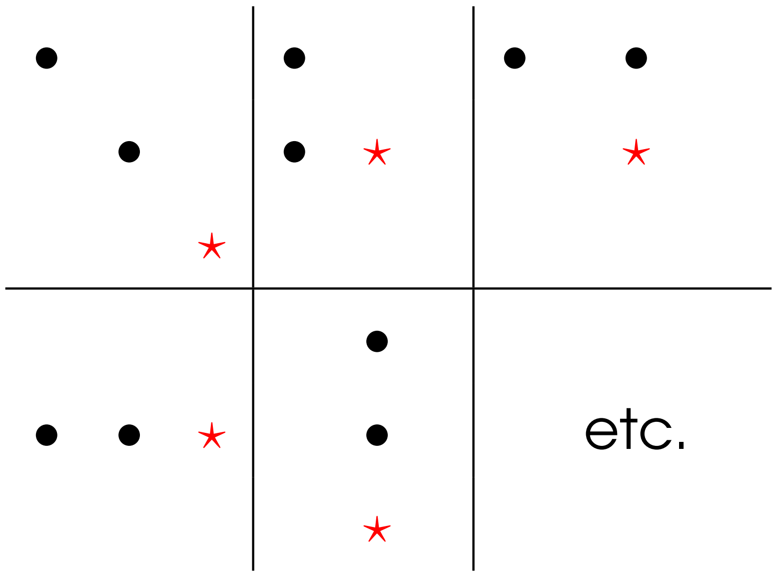
Each family lives on a **diagonal**.

Two families located on **adjacent diagonals** are called **adjacent families** of FOP.

As the usual orthogonal polynomials, adjacent families satisfy **recurrence relationships** (assuming all polynomials exist and have a degree equal to their index).

In particular, they satisfy a **three–term recurrence relationship**.

But other recurrences also exist



They leads to relations for computing recursively **any** sequence of adjacent Padé approximants.

The ε–algorithm:

Let $(\mathbf{S_n})$ be a sequence of number. The ε**–algorithm** is a convergence acceleration algorithm which consists in computing

$$\varepsilon_{\mathbf{k+1}}^{(\mathbf{n})} = \varepsilon_{\mathbf{k-1}}^{(\mathbf{n+1})} + (\varepsilon_{\mathbf{k}}^{(\mathbf{n+1})} - \varepsilon_{\mathbf{k}}^{(\mathbf{n})})^{-\mathbf{1}}, \quad \mathbf{k}, \mathbf{n}, = \mathbf{0}, \mathbf{1}, \ldots$$

with $\varepsilon_{-\mathbf{1}}^{(\mathbf{n})} = \mathbf{0}$ and $\varepsilon_{\mathbf{0}}^{(\mathbf{n})} = \mathbf{S_n}$.

The quantities $\varepsilon_{2k+1}^{(n)}$ are intermediate computations.

Under some assumptions, the sequences $(\varepsilon_{\mathbf{2k}}^{(\mathbf{n})})$, when $\mathbf{k}$ is fixed and $\mathbf{n}$ tends to infinity, or vice-versa, converge faster than the initial sequence (S_n).

If the ε**–algorithm** is applied to the partial sums of the series $\mathbf{f}$, that is

$$\mathbf{S_n = \sum_{i=0}^{n} c_i t^i}$$

then

$$\varepsilon_{\mathbf{2k}}^{(\mathbf{n})} = [\mathbf{n+k/k}]_{\mathbf{f}}(\mathbf{t}).$$

Thus the ε**–algorithm** allows to compute recursively **half** of the Padé table for a **fixed** value of $\mathbf{t}$.

The other half is obtained thanks to the relation

$$[p/q]_f(t) = 1/[q/p]_g(t)$$

where $g(t) = 1/f(t)$ is the reciprocal series of f.

Error estimation

The error of Gaussian quadratures can be estimated by **Kronrod procedure**.

Since Padé approximants can be interpreted as **formal Gaussian quadratures**, their error can also be estimated via Kronrod procedure.

This procedure consists in constructing **another quadrature** formula using the nodes of the Gaussian formula **plus** additional nodes chosen in an **optimal** way (that is to achieve the highest possible degree of exactness).

Hence, for Padé approximants, we will construct the Padé–type approximant $(n+k-1/n+k)$ with the generating polynomial $v(x) = P_k(x)\mathbf{V_n(x)}$, where V_n is chosen so that

$$(n+k-1/n+k)_f(t) = f(t) + \mathcal{O}(t^{2n+k})$$

If $n < k$, $(n+k-1/n+k)$ reduces to $[k-1/k]$. So, we have to take $\mathbf{n = k+1}$.

$\mathbf{V_{k+1}}$ is chosen so that (Stieltjes polynomial)

$$c(x^i P_k V_{k+1}) = 0, \quad i = 0, \ldots, k.$$

We obtain

$$\frac{\mathbf{f(t) - [k-1/k]_f(t)}}{\mathbf{(2k/2k+1)_f(t) - [k-1/k]_f(t)}} = \mathbf{1} + \mathcal{O}(\mathbf{t^{k+2}})$$

thus showing that $\mathbf{(2k/2k+1)_f(t) - [k-1/k]_f(t)}$ is a good **approximation of** the error $\mathbf{f(t) - [k-1/k]_f(t)}$.

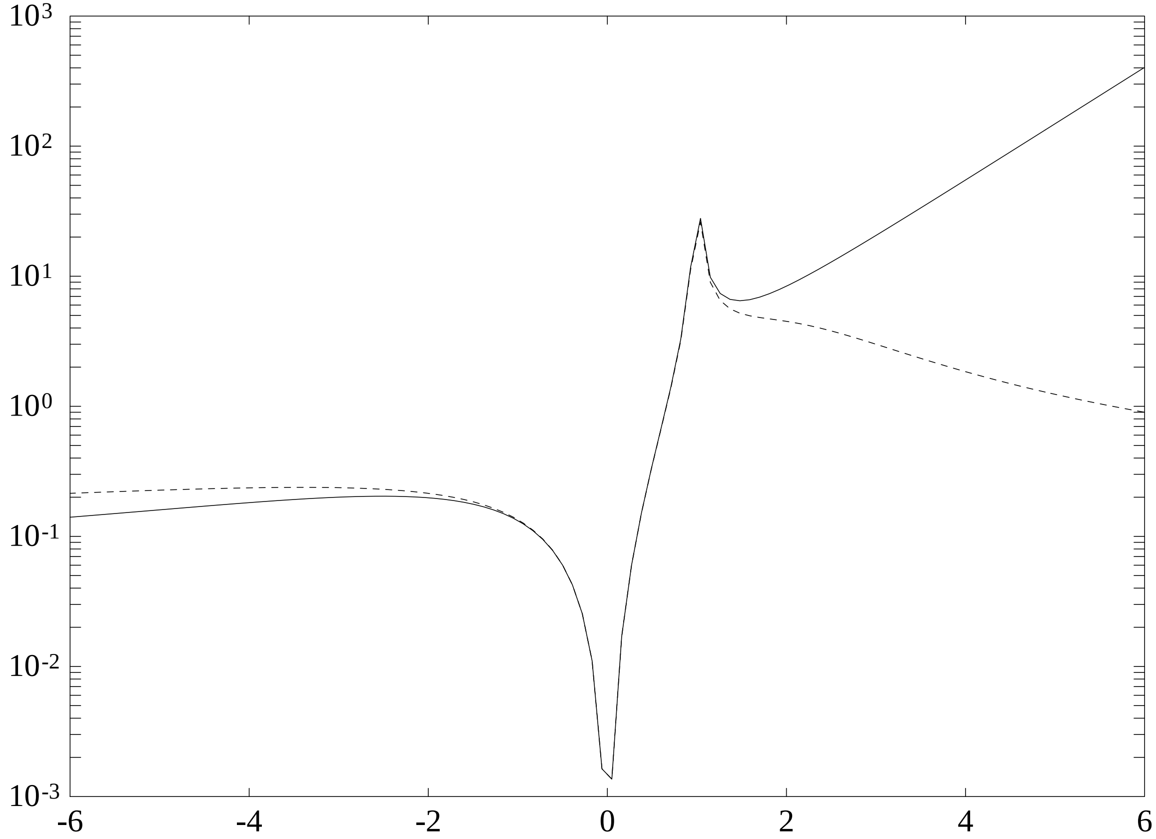
Example 1 : $f(t) = e^t$

$$
\begin{aligned}
[0/1] &= \frac{1}{1-t} \\
(2/3) &= \frac{-11t^2 - 24t + 36}{(1-t)(7t^2 - 24t + 36)} \\
[1/2] &= \frac{6 + 2t}{6 - 4t + t^2} \\
(4/5) &= \frac{379t^4/4 + 468t^3 - 630t^2 - 4200t + 15750}{(6 - 4t + t^2)(-62t^3 + 420t^2 - 1575t + 2625)}
\end{aligned}
$$

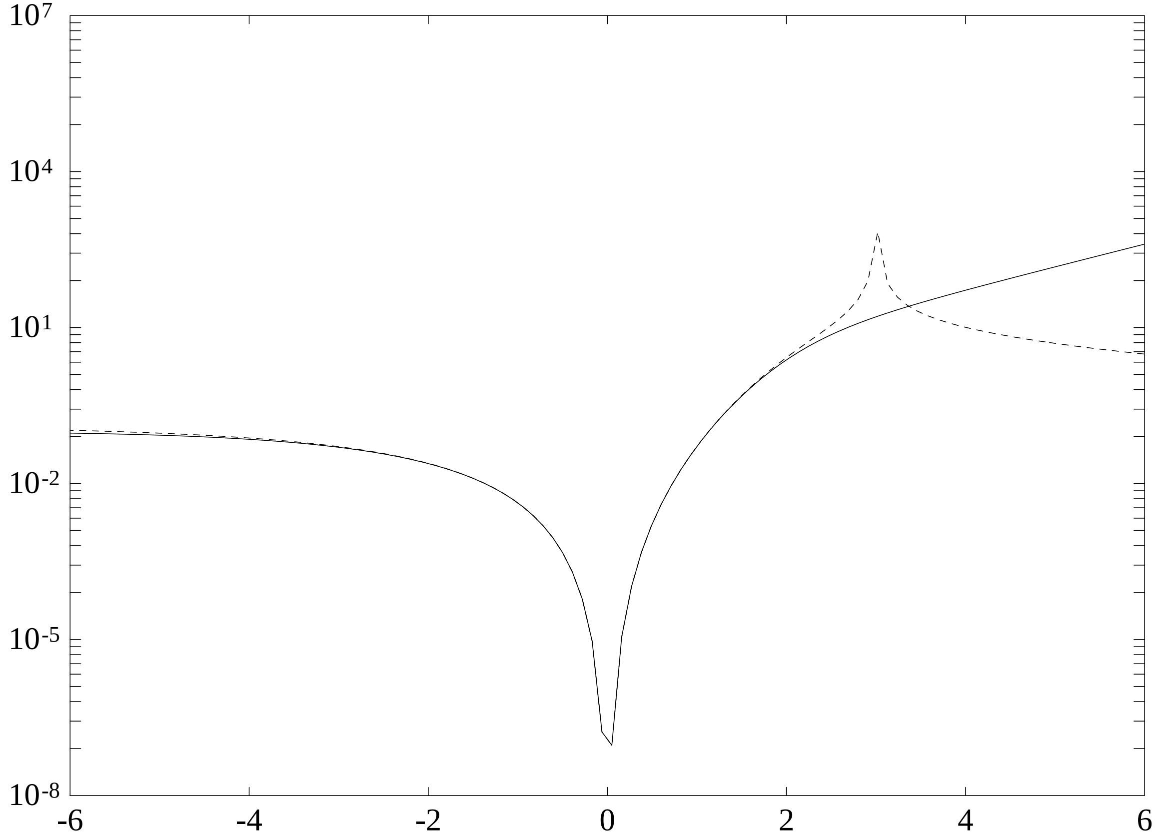
Solid line = **error**

Dashed line = **estimates** of the error

$f(t) = e^t$: error estimate for $[0/1]$



$f(t) = e^t$: error estimate for $[1/2]$



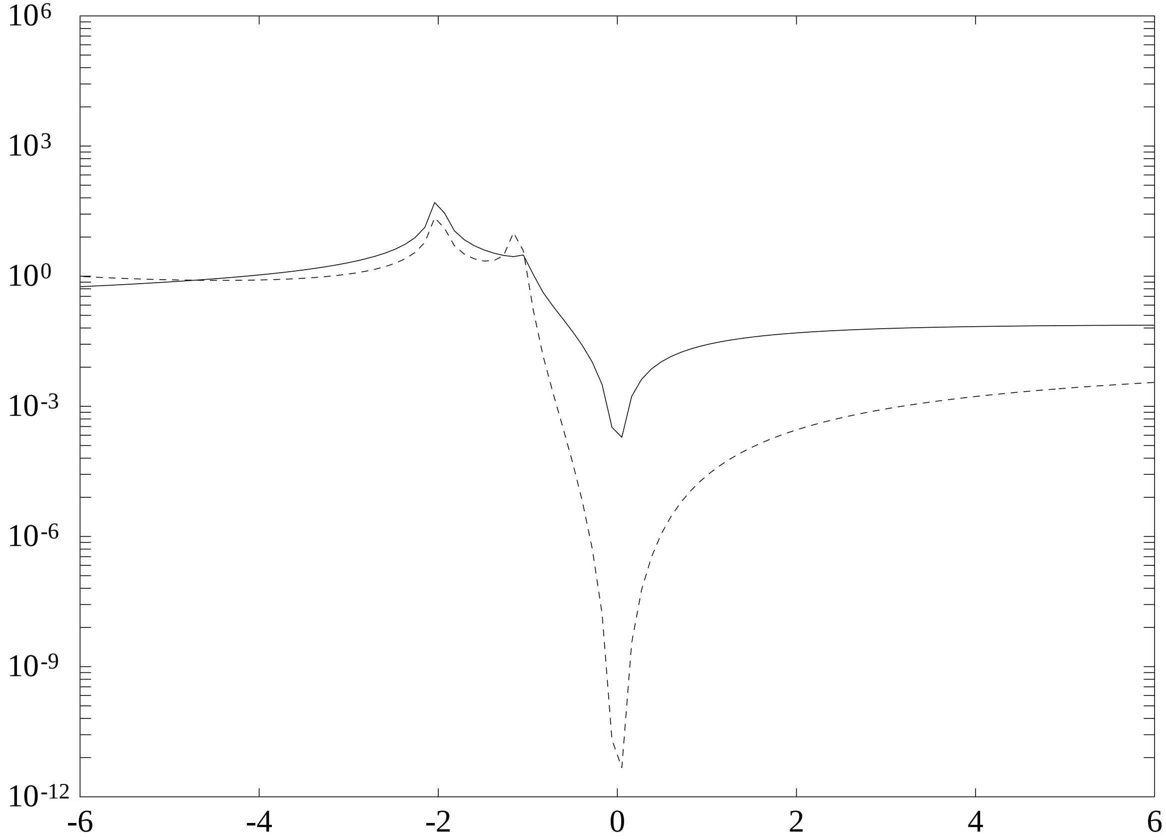
Example 2 : $f(t) = \ln(1+t)/t$

$$
\begin{aligned}
[0/1] &= \frac{2}{2+t} \\
(2/3) &= \frac{11t^2 + 60t + 60}{(2+t)(3t^2 + 30t + 30)} \\
[1/2] &= \frac{6+3t}{6+6t+t^2} \\
(4/5) &= \frac{73t^4 + 1440t^3 + 6480t^2 + 10080t + 5040}{15(6+6t+t^2)(t^3 + 30t^2 + 84t + 56)}
\end{aligned}
$$

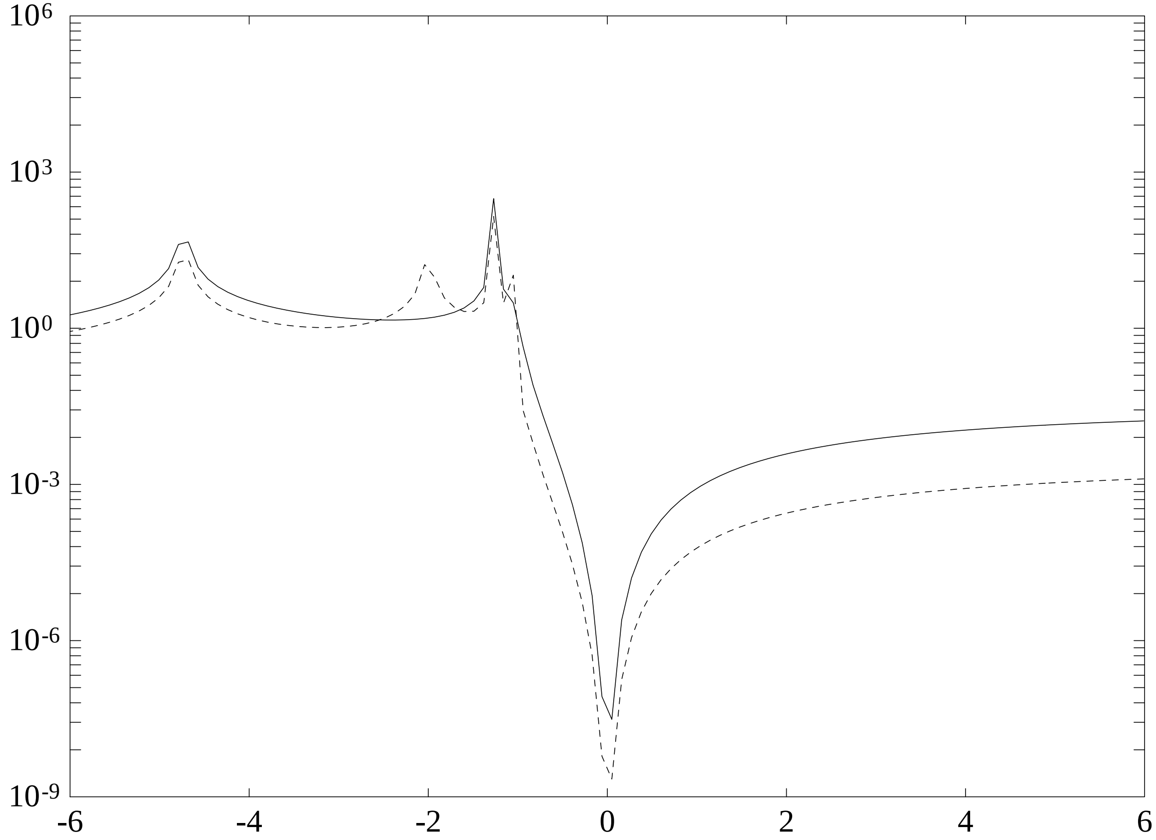
Solid line = **error**

Dashed line = **estimates** of the error

$f(t) = \ln(1+t)/t$: error estimate for $[0/1]$



$f(t) = \ln(1+t)/t$: error estimate for $[1/2]$



Convergence

Meaning : convergence **in some sense** of **some** sequence of PA.

✘ **Difficult problem ! Only few results !**

Let us give a very simple example showing the difficulties.

$$f(t) = \frac{10+t}{1-t^2} = \sum_{i=0}^{\infty} c_i t^i$$

with $c_{2i} = 10$ and $c_{2i+1} = 1$. It **converges for $|t| < 1$**.

$$[k/1]_f(t) = \sum_{i=0}^{k-1} c_i t^i + \frac{c_k t^k}{1 - c_{k+1} t / c_k}.$$

k is odd : $[k/1]$ has a simple **pole** at $t = 1/10$ while f has no pole. Thus $([k/1])$ **cannot converge to f in $|t| < 1$**.

But a paradoxical situation can also arise:
the **zeros** of the Padé approximants can prevent convergence.

This is the case for the reciprocal series of f

$$g(t) = \frac{1-t^2}{10+t}.$$

It converges in $|t| < 10$.

Since $[1/k]_g(t) = 1/[k/1]_f(t)$, we have $[1/2k+1]_g(0.1) = 0$, while $g(0.1) \neq 0$.

Thus, the sequence $([1/k]_g)$ cannot converge in $|t| < 10$ where the series g does.

Theorem

Let (m_k) and (n_k) be two sequences of nonnegative integers so that

$$\lim_{k\to\infty} \max\ (m_k, n_k) = \infty.$$

Let $\mathbf{R_k(t) = [m_k/n_k]_f(t)}$ and let $\mathbf{D}$ be a domain of the complex plane containing the origin.

Then

- (R_k) **converges uniformly on every compact subset of** D **iff** $\{R_k(t)\}$ **is uniformly bounded on every compact subset of** D**,**
- **if** (R_k) **converges uniformly on every compact subset of** D**, then the function** $f(t) = \lim_{k\to\infty} R_k(t)$ **is holomorphic in** D **and the series** f **is the Taylor expansion of the function** f **about the origin.**

Theorem (de Montessus de Ballore, 1902)

Let f be analytic at $t = 0$ and meromorphic with exactly $\mathbf{k}$ poles $\alpha_1, \ldots, \alpha_k$ (counted with their multiplicity) in the disc $D_R = \{t \mid |t| < R\}$. Let $D = D_R - \{\alpha_1, \ldots, \alpha_k\}$.

When n tends to infinity, the sequence $([n/k])$ converges uniformly on every compact subset of D.

The poles of $[n/k]$ tend to those of f.

Stieltjes series : $f(t) = \int_a^b \frac{d\alpha(x)}{1+xt}$, α bounded, nondecreasing, with infinitely many different values and $-\infty < a \leq 0 \leq b < \infty$.

Theorem (Markov, 1884)

For all fixed $\mathbf{n \geq -1}$, the sequence $\mathbf{([n+k/k])}$ converges uniformly and geometrically, when $\mathbf{k}$ tends to infinity, in any open set of the complex plane cut along $\mathbf{(-\infty, -b^{-1}] \cup [-a^{-1}, \infty)}$.

Pólya frequency series :

$$f(t) = a_0 e^{\gamma t} \prod_{i \geq 0} \frac{1+\alpha_i t}{1-\beta_i t}$$

with $a_0 > 0, \gamma \geq 0, \alpha_i \geq 0, \beta_i \geq 0, \sum_i (\alpha_i + \beta_i) < \infty.$

Let $[m_k/n_k] = N_k/D_k$

with $\lim_{k\to\infty} m_k = \infty, \quad \lim_{k\to\infty} m_k/n_k = a, \quad 0 \leq a \leq \infty.$

Then, uniformly on any compact subset of complex plane,

$$\begin{aligned} \lim_{k\to\infty} N_k &= a_0 e^{a\gamma t/(1+a)} \prod_{i\geq 0} (1+\alpha_i t) \\ \lim_{k\to\infty} D_k &= e^{-\gamma t/(1+a)} \prod_{i \geq 0} (1-\beta_i t). \end{aligned}$$

AN APPLICATION

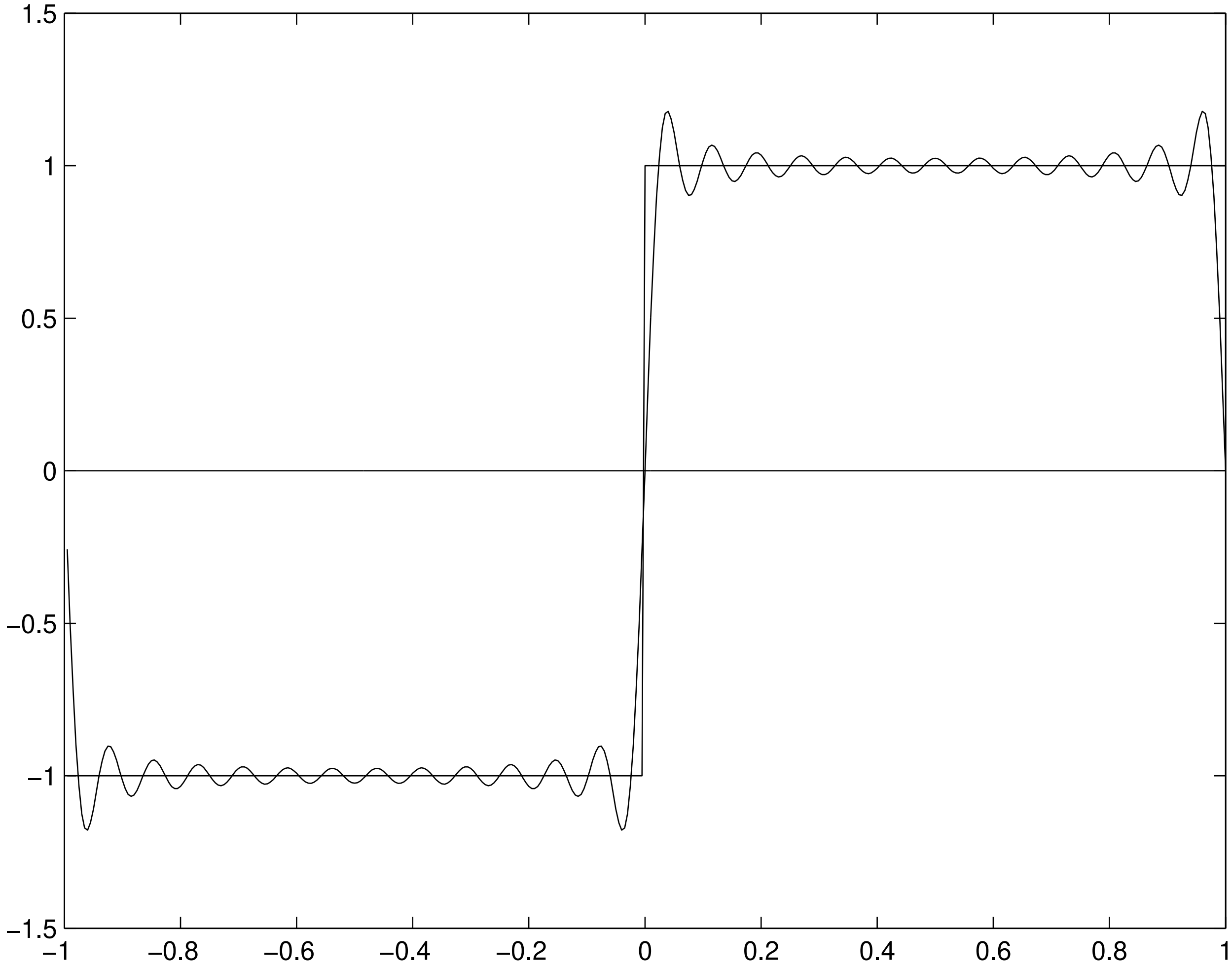
TREATMENT OF THE GIBBS PHENOMENON

Let us consider the partial sums of the Fourier series of the sign function

$$f(t) = \frac{4}{\pi} \sum_{k=1}^{\infty} \frac{1}{2k-1} \cos((2k-1)\pi t) = \begin{cases} -1, & t < 0 \\ +1, & t > 0. \end{cases}$$

They exhibit a **Gibbs phenomenon**.

For the 13th partial sum, we have



We will now discuss is the application of the

ε-algorithm

to the partial sums of a Fourier series in order to

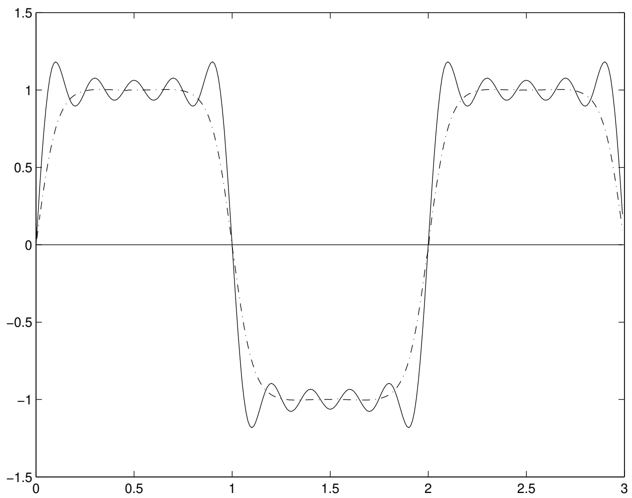
- **accelerate the convergence,**
- **locate the discontinuities,**
- **reduce the Gibbs phenomenon.**

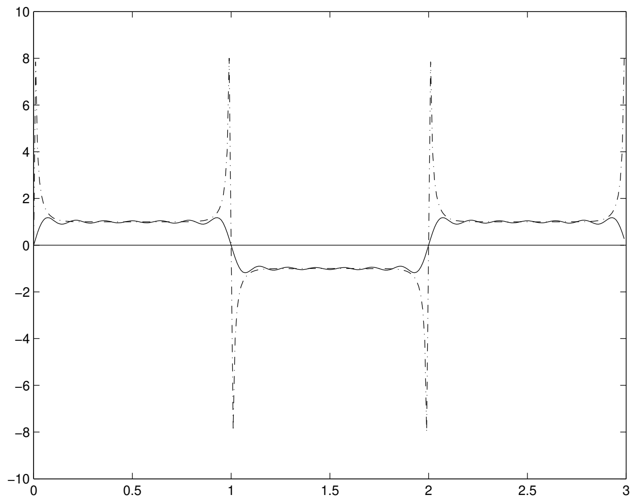
Fourier series

We consider the Fourier series

$$S(t) = \frac{4}{\pi} \sum_{k=0}^{\infty} \frac{1}{2k+1} \sin((2k+1)\pi t).$$

It has discontinuities at the integer points and its partial sums exhibits a Gibbs phenomenon at these points.





The graph on the left represents $S(t)$ (dash-dotted line) and $S_4(t)$ (solid line). The graph on the right shows $S_6(t)$ and $\varepsilon_6^{(0)}$.

The ε-algorithm **reveals the location of the singularities** and **reduces the Gibbs phenomenon** away from discontinuities.

We consider a general Fourier series

$$S(t) = \sum_{k=1}^{\infty} a_k \cos kt + \sum_{k=1}^{\infty} b_k \sin kt.$$

If the ε-algorithm is applied to the partial sums of $S(t)$, then

$$\varepsilon_{2k}^{(n)} - S(t) = \mathcal{O}(\text{Re } e^{i(n+k+1)t}).$$

Thus, $\varepsilon_{2k}^{(n)}$ is Padé-type approximant of S.

Fourier series and their conjugate series

Adding it its conjugate series

$$\widetilde{S}(t) = \sum_{k=1}^{\infty} a_k \sin kt - \sum_{k=1}^{\infty} b_k \cos kt,$$

as an imaginary part, we get

$$\mathbf{F(t) = S(t) + i\widetilde{S}(t)} = \sum_{k=1}^{\infty} (a_k - ib_k) e^{ikt}.$$

We make the change of variable $z = e^{it}$, and we set $c_k = a_k - ib_k$. Then

$$\mathbf{F(z) = \sum_{k=1}^{\infty} c_k z^k.}$$

We apply the ε-algorithm to the **partial sums of** $F(t)$, and then we take the **real part** of $\varepsilon_{\mathbf{2k}}^{(\mathbf{n})}$.

Thanks to the connection between Padé approximants and the ε-algorithm we have

$$\text{Re } \varepsilon_{2k}^{(n)} = \text{Re } [n+k/k]_F(e^{it}).$$

Then

$$\text{Re } \varepsilon_{2k}^{(n)} = S(t) + \mathcal{O}(\text{Re } e^{i(n+2k+1)t}).$$

So we have now **Padé** approximants instead of **Padé-type** approximants (order of approx. $\mathbf{n+2k+1}$ instead of $\mathbf{n+k+1}$).

Such a strategy was first evoked by P. Wynn (1967), and some numerical examples were already given by CB (1978) and CB-M. Redivo Zaglia (1991).

An example:

We consider

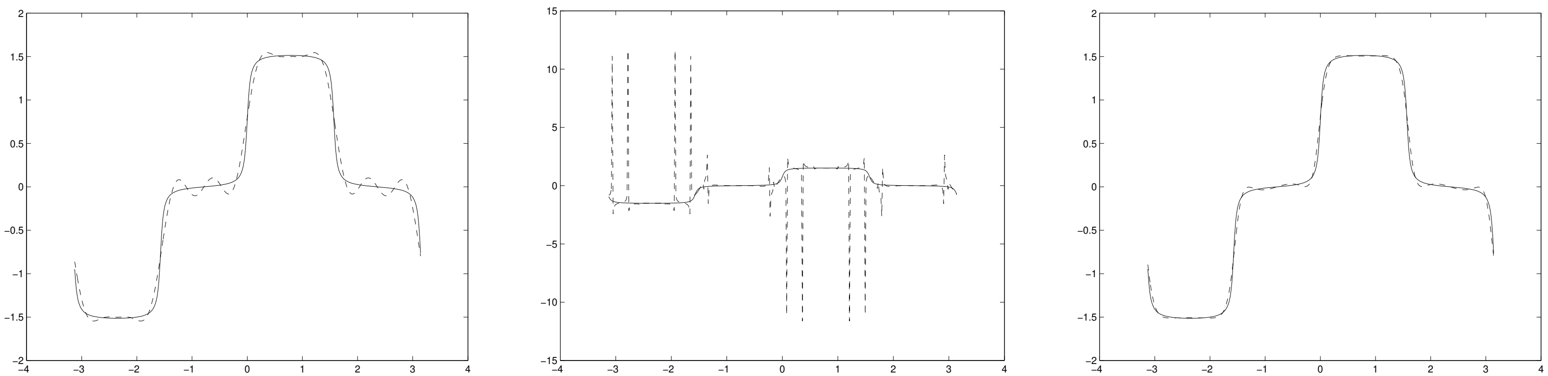
$$
\begin{aligned}
S(t) &= \frac{1}{2}\left(\arctan\frac{2a\cos t}{1-a^2} + \arctan\frac{2a\sin t}{1-a^2}\right) \\
&= \sum_{k=1}^{\infty}(-1)^{k+1}\frac{a^{2k-1}}{2k-1}\cos(2k-1)t + \sum_{k=1}^{\infty}\frac{a^{2k-1}}{2k-1}\sin(2k-1)t.
\end{aligned}
$$

The solid line is the exact result.

The graph on the left represents $S_4(t)$ for $a = 0.96$.

The graph in the middle shows $\varepsilon_4^{(0)}$ when applied to the partial sums of $S(t)$.

The graph on the right gives the real part of $\varepsilon_4^{(0)}$ obtained from the partial sums of $F(t)$.



Generalizations of Padé approximants

- Partial PA
- Multipoint PA
- Cauchy-type approximants
- Approximants for series of functions
- Padé-Hermite approximants
- Vector PA
- Non-commutative PA
- Multivariate PA

Continued fractions

Rafael Bombelli (Bologne, 1526 - 1572)

Pietro Antonio Cataldi (Bologne, 1548 - Bologne, 1626)

A **continued fraction** is an expression of the form

$$C = b_0 + \cfrac{a_1}{b_1 + \cfrac{a_2}{b_2 + \cfrac{a_3}{b_3 + \cfrac{a_4}{\ddots}}}}.$$

For evident typographical reasons, it will be written as

$$C = b_0 + \frac{a_1}{b_1+}\frac{a_2}{b_2+}\frac{a_3}{b_3+}\cdots$$

a_k and b_k are the kth **partial numerator** and **partial denominator**.

a_k/b_k is the kth **partial quotient**.

$$C_n = b_0 + \frac{a_1}{b_1+} \cdots \frac{a_{n-1}}{b_{n-1}+} \frac{a_n}{b_n}$$

is the nth **convergent** of the continued fraction C.

The continued fraction is said to converge if the sequence (C_n) converges when n goes to infinity.

The convergence of a continued fraction can be **accelerated** by various procedures.

C_n can be written as $\mathbf{C_n = A_n / B_n}$ and it holds (Bhascara, 1150)

$$\begin{aligned} A_k &= b_k A_{k-1} + a_k A_{k-2} \\ B_k &= b_k B_{k-1} + a_k B_{k-2}, \quad k = 1, 2, \ldots \end{aligned}$$

with

$$A_0 = b_0, \quad A_{-1} = 1$$
$$B_0 = 1, \quad B_{-1} = 0.$$

Let us consider the continued fraction

$$C = b_0 + \frac{a_1 t}{1+} \frac{a_2 t}{1+} \frac{a_3 t}{1+} \cdots$$

$\mathbf{A_{2k-1}}, \mathbf{A_{2k}}$ and $\mathbf{B_{2k}}$ are polynomials of degree $\mathbf{k}$ in t and that $\mathbf{B_{2k-1}}$ is a polynomial of degree $\mathbf{k-1}$.

It is possible to choose $b_0, a_1, a_2, \ldots$ so that the expansion of $\mathbf{C_k}$ agrees with that of a given series $f(t) = c_0 + c_1 t + c_2 t^2 + \cdots$ up to the term of degree $\mathbf{k}$.

This continued fraction is called the continued fraction **corresponding** to the series f.

Thus, $C_{2k} = [k/k]_f(t)$ and $C_{2k+1} = [k+1/k]_f(t)$.

Related to the QD-**algorithm** of Heinz Rutishauser.