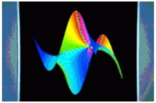
Algorithmique semi-numérique rapide des séries de Tchebychev

Alexandre Benoit

Projet Algorithms



INRIA

Soutenance de thèse
Directeur de thèse : Bruno Salvy
18 juillet 2012

Summary

Goal

- Using computer algebra, find fast algorithms to compute guaranteed polynomial approximations of functions.

Summary

Goal

- Using computer algebra, find fast algorithms to compute guaranteed polynomial approximations of functions.

Framework

- Approximation theory.
- Computer algebra:

Summary

Goal

- Using computer algebra, find fast algorithms to compute guaranteed polynomial approximations of functions.

Framework

- Approximation theory.
- Computer algebra:
  - Solutions of linear differential equations.

Summary

Goal

- Using computer algebra, find fast algorithms to compute guaranteed polynomial approximations of functions.

Framework

- Approximation theory.
- Computer algebra:
    - Solutions of linear differential equations.
    - Fast algorithms.

Summary

Goal

- Using computer algebra, find fast algorithms to compute guaranteed polynomial approximations of functions.

Framework

- Approximation theory.
- Computer algebra:
  - Solutions of linear differential equations.
  - Fast algorithms.

How

- Chebyshev expansion: very good approximation on an interval with good properties for computer algebra.

Basic Properties of Chebyshev Polynomials

$$T_n(\cos(\theta)) = \cos(n\theta)$$

Basic Properties of Chebyshev Polynomials

$$T_n(\cos(\theta)) = \cos(n\theta)$$

$$\int_{-1}^{1} \frac{T_n(x)T_m(x)}{\sqrt{1-x^2}}dx = \begin{cases} 0 & \text{if } m \neq n \\ \pi & \text{if } m = 0 \\ \frac{\pi}{2} & \text{otherwise} \end{cases}$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \qquad (1-x^2)T_n''(x) - xT_n'(x) + n^2T_n(x) = 0$$

Basic Properties of Chebyshev Polynomials

$$T_n(\cos(\theta)) = \cos(n\theta)$$

$$\int_{-1}^{1} \frac{T_n(x)T_m(x)}{\sqrt{1-x^2}}dx = \begin{cases} 0 & \text{if } m \neq n \\ \pi & \text{if } m = 0 \\ \frac{\pi}{2} & \text{otherwise} \end{cases}$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x) \qquad (1-x^2)T_n''(x) - xT_n'(x) + n^2T_n(x) = 0$$

$$T_n(x) = \sum_{k=0}^{\lfloor n/2 \rfloor} (-1)^k 2^{n-2k-1} \frac{n}{n-k} \binom{n-k}{k} x^{n-2k}$$

$$T_0(x) = 1$$
$$T_1(x) = x$$
$$T_2(x) = 2x^2 - 1$$
$$T_3(x) = 4x^3 - 3x$$
$$T_4(x) = 8x^4 - 8x^2 + 1$$

Approximation by Taylor or Chebyshev Series

Two approximations of a function f:

by a Taylor series

$$f = \sum_{n=0}^{+\infty} u_n x^n,$$

$$u_n = \frac{f^{(n)}(0)}{n!},$$

Approximation by Taylor or Chebyshev Series

Two approximations of a function f:

by a Taylor series

$$f = \sum_{n=0}^{+\infty} u_n x^n,$$

$$u_n = \frac{f^{(n)}(0)}{n!},$$

or by Chebyshev series

$$f = \sum_{n=0}^{+\infty} c_n T_n(x),$$

$$c_n = \frac{1}{\pi} \int_{-1}^{1} T_n(t) \frac{f(t)}{\sqrt{1-t^2}} dt.$$

Chebyshev Truncations are Near-Best

Let f be continuous on $[-1,1]$ with degree n Chebyshev truncation f_n and best approximant p_n (the polynomial of degree at most n that minimizes $\|f-p\|_\infty = \sup_{-1\le x\le 1}|f(x)-p(x)|$), $n \ge 1$. Then

$$\|f - f_n\|_\infty \le \underbrace{\left(4 + \frac{4}{\pi^2}\log(n+1)\right)}_{\Lambda_n} \|f - p_n\|_\infty.$$

Chebyshev Truncations are Near-Best

Let f be continuous on $[-1,1]$ with degree n Chebyshev truncation f_n and best approximant p_n (the polynomial of degree at most n that minimizes $\|f-p\|_\infty = \sup_{-1\le x\le 1}|f(x)-p(x)|$), $n \ge 1$. Then

$$\|f-f_n\|_\infty \le \underbrace{\left(4+\frac{4}{\pi^2}\log(n+1)\right)}_{\Lambda_n}\|f-p_n\|_\infty .$$

- $\Lambda_{10} = 4.93... \to < 3$ bits
- $\Lambda_{30} = 5.37... \to < 3$ bits
- $\Lambda_{100} = 5.87... \to < 3$ bits
- $\Lambda_{1000} = 6.80... \to < 3$ bits

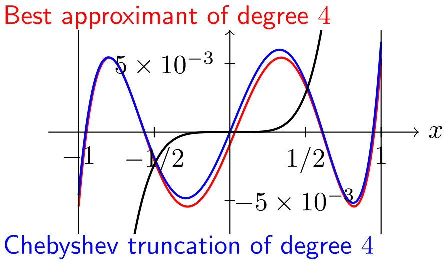
Chebyshev Truncations are Near-Best

Let f be continuous on $[-1,1]$ with degree n Chebyshev truncation f_n and best approximant p_n (the polynomial of degree at most n that minimizes $\|f-p\|_\infty = \sup_{-1\le x\le 1}|f(x)-p(x)|$), $n \ge 1$. Then

$$\|f-f_n\|_\infty \le \underbrace{\left(4+\frac{4}{\pi^2}\log(n+1)\right)}_{\Lambda_n}\|f-p_n\|_\infty.$$

- $\Lambda_{10} = 4.93... \to < 3$ bits
- $\Lambda_{30} = 5.37... \to < 3$ bits
- $\Lambda_{100} = 5.87... \to < 3$ bits
- $\Lambda_{1000} = 6.80... \to < 3$ bits

error of the approximation of $\exp$



Chebyshev Truncations are Near-Best

Let f be continuous on $[-1,1]$ with degree n Chebyshev truncation f_n and best approximant p_n (the polynomial of degree at most n that minimizes $\|f-p\|_\infty = \sup_{-1\le x\le 1}|f(x)-p(x)|$), $n \ge 1$. Then

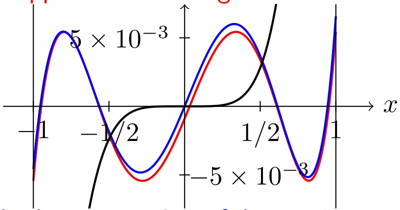
$$\|f-f_n\|_\infty \le \underbrace{\left(4+\frac{4}{\pi^2}\log(n+1)\right)}_{\Lambda_n}\|f-p_n\|_\infty.$$

- $\Lambda_{10} = 4.93\ldots \to < 3$ bits
- $\Lambda_{30} = 5.37\ldots \to < 3$ bits
- $\Lambda_{100} = 5.87\ldots \to < 3$ bits
- $\Lambda_{1000} = 6.80\ldots \to < 3$ bits

It's faster to compute f_n instead of p_n

error of the approximation of $\exp$

Best approximant of degree 4



Chebyshev truncation of degree 4

Contributions

- Fast polynomial conversion, Monomial bases $\mapsto$ Chebyshev basis (with Alin Bostan, Chap. 3)

Contributions

- Fast polynomial conversion, Monomial bases $\mapsto$ Chebyshev basis (with Alin Bostan, Chap. 3)
- When a Chebyshev series is solution of a linear differential equation, its coefficients are solutions of a linear recurrence equation
  - Fast algorithm to compute this recurrence using fractions of recurrence operators (with Bruno Salvy, ISSAC'09)

Contributions

- Fast polynomial conversion, Monomial bases $\mapsto$ Chebyshev basis (with Alin Bostan, Chap. 3)
- When a Chebyshev series is solution of a linear differential equation, its coefficients are solutions of a linear recurrence equation
  - Fast algorithm to compute this recurrence using fractions of recurrence operators (with Bruno Salvy, ISSAC'09)
  - Use of this recurrence relation to compute the coefficients numerically (with Mioara Joldeș and Marc Mezzarobba, Chap. 6 and 7)

Contributions

- Fast polynomial conversion, Monomial bases $\mapsto$ Chebyshev basis (with Alin Bostan, Chap. 3)
- When a Chebyshev series is solution of a linear differential equation, its coefficients are solutions of a linear recurrence equation
  - Fast algorithm to compute this recurrence using fractions of recurrence operators (with Bruno Salvy, ISSAC'09)
  - Use of this recurrence relation to compute the coefficients numerically (with Mioara Joldeș and Marc Mezzarobba, Chap. 6 and 7)
  - Implementation in the DDMF (ICMS'10)

Contributions

- Fast polynomial conversion, Monomial bases $\mapsto$ Chebyshev basis (with Alin Bostan, Chap. 3)
- When a Chebyshev series is solution of a linear differential equation, its coefficients are solutions of a linear recurrence equation
  - Fast algorithm to compute this recurrence using fractions of recurrence operators (with Bruno Salvy, ISSAC'09)
  - Use of this recurrence relation to compute the coefficients numerically (with Mioara Joldeș and Marc Mezzarobba, Chap. 6 and 7)
  - Implementation in the DDMF (ICMS'10)
- Generalization of the algorithm for the Chebyshev recurrence to the computation of recurrences for the coefficients of generalized Fourier series (with Bruno Salvy, Chap. 8)

Contributions

- Fast polynomial conversion, Monomial bases $\mapsto$ Chebyshev basis (with Alin Bostan, Chap. 3)
- When a Chebyshev series is solution of a linear differential equation, its coefficients are solutions of a linear recurrence equation
  - Fast algorithm to compute this recurrence using fractions of recurrence operators (with Bruno Salvy, ISSAC'09)
  - Use of this recurrence relation to compute the coefficients numerically (with Mioara Joldeș and Marc Mezzarobba, Chap. 6 and 7)
  - Implementation in the DDMF (ICMS'10)
- Generalization of the algorithm for the Chebyshev recurrence to the computation of recurrences for the coefficients of generalized Fourier series (with Bruno Salvy, Chap. 8)
- Fast algorithm for the product of differential operators (with Alin Bostan and Joris van der Hoeven, FOCS'12)

Outline

I DDMF

The Special Function $\operatorname{erf}(x)$

[-] ## 1. Differential equation

rendering link

The function $\operatorname{erf}(x)$ satisfies the differential equation

$$2\left(\frac{d}{dx}y(x)\right)x+\frac{d^2}{dx^2}y(x)=0$$

with initial values $y(0)=0$, $(y')(0)=2\frac{1}{\sqrt{\pi}}$.

[+] ## 2. Plot

[+] ## 3. Numerical Evaluation

[+] ## 4. Symmetry

[+] ## 5. Taylor Expansion at 0

[+] ## 6. Local Expansions at Singularities and at Infinity

† ICMS 2010, B., Chyzak, Darasse, Gerhold, Mezzarobba and Salvy (http://ddmf.msr-inria.inria.fr).

DDMF[†]

[-] 8. Chebyshev Expansion over $[-1, 1]$

- Chebyshev expansion:

$$\operatorname{erf}(x) = \sum_{n=0}^{\infty} 2\,\frac{4^{-n}(-1)^n\,{}_1F_1(1/2+n;\,2n+2;-1)T_{2n+1}(x)}{\sqrt{\pi}\,(2n+1)\,n!}.$$

- First terms and polynomial approximation:

$$\operatorname{erf}(x) = 0.904347\,T_1(x) - 0.0661130\,T_3(x) + 0.00472936\,T_5(x) + \dots$$

$$\operatorname{erf}(x) \approx 1.12633280\,x - 0.35903920\,x^3 + 0.07566976\,x^5.$$

order = 6 Submit

- The coefficients c_n in the Chebyshev expansion $\operatorname{erf}(x) = \sum_{n=0}^{\infty} c_n T_n(x)$ satisfy the recurrence

$$(n^2+3n)\,c(n) + (2n^3+12n^2+24n+16)\,c(n+2) + (-n^2-5n-4)\,c(n+4) = 0.$$

[†]ICMS 2010, B., Chyzak, Darasse, Gerhold, Mezzarobba and Salvy (`http://ddmf.msr-inria.inria.fr`).

II Chebyshev Recurrence

Chebyshev Recurrence

Theorem (Paszkowski 1975)

If $\sum c_n T_n(x)$ is solution of a linear differential equation with polynomial coefficients, then the coefficients c_n are solution of a linear recurrence with polynomial coefficients.

DDMF[†]

[-] ## 8. Chebyshev Expansion over $[-1, 1]$

- Chebyshev expansion:

$$\operatorname{erf}(x) = \sum_{n=0}^{\infty} 2\,\frac{4^{-n}(-1)^n\,{}_1F_1(1/2+n;\,2n+2;-1)T_{2n+1}(x)}{\sqrt{\pi}(2n+1)\,n!}.$$

- First terms and polynomial approximation:

$$\operatorname{erf}(x) = 0.904347\,T_1(x) - 0.0661130\,T_3(x) + 0.00472936\,T_5(x) + \dots$$

$$\operatorname{erf}(x) \approx 1.12633280\,x - 0.35903920\,x^3 + 0.07566976\,x^5.$$

order = 6 Submit

- The coefficients c_n in the Chebyshev expansion $\operatorname{erf}(x) = \sum_{n=0}^{\infty} c_n T_n(x)$ satisfy the recurrence

$$(n^2+3n)\,c(n) + (2n^3+12n^2+24n+16)\,c(n+2) + (-n^2-5n-4)\,c(n+4) = 0.$$

[†]ICMS 2010, B., Chyzak, Darasse, Gerhold, Mezzarobba and Salvy (`http://ddmf.msr-inria.inria.fr`).

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Taylor series ($f := \sum u_n x^n$)

$$xf = \sum u_n x^{n+1} = \sum u_{n-1} x^n,$$

$$f' = \sum n u_n x^{n-1} = \sum (n+1) u_{n+1} x^n$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Taylor series ($f := \sum u_n x^n$)

$$xf = \sum u_n x^{n+1} = \sum u_{n-1} x^n,$$

$$f' = \sum n u_n x^{n-1} = \sum (n+1) u_{n+1} x^n$$

$$x \mapsto X := S^{-1},$$

$$\frac{d}{dx} \mapsto D := (n+1)S.$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Taylor series ($f := \sum u_n x^n$)

$$xf = \sum u_n x^{n+1} = \sum u_{n-1} x^n,$$

$$f' = \sum n u_n x^{n-1} = \sum (n+1) u_{n+1} x^n$$

$$x \mapsto X := S^{-1},$$

$$\frac{d}{dx} \mapsto D := (n+1)S.$$

$$(4+x^2)\left(\frac{d}{dx}\right)^2 + 2x\frac{d}{dx}$$

$$\mapsto (4+S^{-2})(n+1)(n+2)S^2 + 2S^{-1}(n+1)S$$

$$= (n+1)\left(4(n+2)S^2 + n\right)$$

$$4(n+2)u_{n+2} + n u_n = 0$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Monomial Basis $x^n = M_n(x)$

$$xM_n(x) = M_{n+1}(x),$$
$$M_n'(x) = nM_{n-1}(x).$$

Chebyshev series

$$xT_n(x) = 1/2\left(T_{n+1}(x) + T_{n-1}(x)\right)$$
$$T_n'(x) = \frac{n\left(T_{n-1}(x) - T_{n+1}(x)\right)}{2(1-x^2)}.$$

$$x \mapsto X := S^{-1},$$
$$\frac{d}{dx} \mapsto D := (n+1)S.$$

$$x \mapsto X := \frac{S + S^{-1}}{2},$$
$$\frac{d}{dx} \mapsto D := \frac{(n+1)S - (n-1)S^{-1}}{2(1-X^2)} = \frac{2n}{S^{-1} - S}.$$

$$(4+x^2)\left(\frac{d}{dx}\right)^2 + 2x\frac{d}{dx}$$
$$\mapsto (4+S^{-2})(n+1)(n+2)S^2 + 2S^{-1}(n+1)S$$
$$= (n+1)\left(4(n+2)S^2 + n\right)$$
$$4(n+2)u_{n+2} + nu_n = 0$$

$$\frac{(n-1)(n+1)\left((n+2)S^2 + 18n + (n-2)S^{-2}\right)}{\left((n-1)S^2 - 2n + (n+1)S^{-2}\right)},$$

$$(n+2)c_{n+2} + 18nc_n + (n-2)c_{n-2} = 0.$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Monomial Basis $x^n = M_n(x)$

$$xM_n(x) = M_{n+1}(x),$$
$$M_n'(x) = nM_{n-1}(x).$$

$$x \mapsto X := S^{-1},$$
$$\frac{d}{dx} \mapsto D := (n+1)S.$$

$$(4+x^2)\left(\frac{d}{dx}\right)^2 + 2x\frac{d}{dx}$$
$$\mapsto (4+S^{-2})(n+1)(n+2)S^2 + 2S^{-1}(n+1)S$$
$$= (n+1)\left(4(n+2)S^2 + n\right)$$
$$4(n+2)u_{n+2} + nu_n = 0$$

Chebyshev series

$$xT_n(x) = 1/2\left(T_{n+1}(x) + T_{n-1}(x)\right)$$
$$T_n'(x) = \frac{n\left(T_{n-1}(x) - T_{n+1}(x)\right)}{2(1-x^2)}.$$

$$x \mapsto X := \frac{S+S^{-1}}{2},$$
$$\frac{d}{dx} \mapsto D := \frac{(n+1)S - (n-1)S^{-1}}{2(1-X^2)} = \frac{2n}{S^{-1} - S}.$$

$$\frac{(n-1)(n+1)\left((n+2)S^2 + 18n + (n-2)S^{-2}\right)}{\left((n-1)S^2 - 2n + (n+1)S^{-2}\right)},$$

$$(n+2)c_{n+2} + 18nc_n + (n-2)c_{n-2} = 0.$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Monomial Basis $x^n = M_n(x)$

$$xM_n(x) = M_{n+1}(x),$$
$$M'_n(x) = nM_{n-1}(x).$$

$$x \mapsto X := S^{-1},$$
$$\frac{d}{dx} \mapsto D := (n+1)S.$$

$$(4+x^2)\left(\frac{d}{dx}\right)^2 + 2x\frac{d}{dx}$$
$$\mapsto (4+S^{-2})(n+1)(n+2)S^2 + 2S^{-1}(n+1)S$$
$$= (n+1)\left(4(n+2)S^2 + n\right)$$
$$4(n+2)u_{n+2} + nu_n = 0$$

Chebyshev series

$$xT_n(x) = 1/2\left(T_{n+1}(x) + T_{n-1}(x)\right)$$
$$T'_n(x) = \frac{n(T_{n-1}(x) - T_{n+1}(x))}{2(1-x^2)}.$$

$$x \mapsto X := \frac{S + S^{-1}}{2},$$
$$\frac{d}{dx} \mapsto D := \frac{(n+1)S - (n-1)S^{-1}}{2(1-X^2)} = \frac{2n}{S^{-1} - S}.$$

$$\frac{(n-1)(n+1)\left((n+2)S^2 + 18n + (n-2)S^{-2}\right)}{\left((n-1)S^2 - 2n + (n+1)S^{-2}\right)},$$

$$(n+2)c_{n+2} + 18nc_n + (n-2)c_{n-2} = 0.$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Monomial Basis $x^n = M_n(x)$

$$xM_n(x) = M_{n+1}(x),$$
$$M_n'(x) = nM_{n-1}(x).$$

Chebyshev series

$$xT_n(x) = 1/2\,(T_{n+1}(x) + T_{n-1}(x))$$
$$T_n'(x) = \frac{n\,(T_{n-1}(x) - T_{n+1}(x))}{2(1-x^2)}.$$

$$x \mapsto X := S^{-1},$$
$$\frac{d}{dx} \mapsto D := (n+1)S.$$

$$x \mapsto X := \frac{S+S^{-1}}{2},$$
$$\frac{d}{dx} \mapsto D := \frac{(n+1)S-(n-1)S^{-1}}{2(1-X^2)} = \frac{2n}{S^{-1}-S}.$$

$$(4+x^2)\left(\frac{d}{dx}\right)^2 + 2x\frac{d}{dx}$$
$$\mapsto (4+S^{-2})(n+1)(n+2)S^2 + 2S^{-1}(n+1)S$$
$$= (n+1)\left(4(n+2)S^2 + n\right)$$
$$4(n+2)u_{n+2} + nu_n = 0$$

$$\frac{(n-1)(n+1)\left((n+2)S^2+18n+(n-2)S^{-2}\right)}{((n-1)S^2-2n+(n+1)S^{-2})},$$

$$(n+2)c_{n+2} + 18nc_n + (n-2)c_{n-2} = 0.$$

Morphisms of Rings of Operators ($S \cdot u_n = u_{n+1}$)

Monomial Basis $x^n = M_n(x)$

$$xM_n(x) = M_{n+1}(x),$$
$$M_n'(x) = nM_{n-1}(x).$$

Chebyshev series

$$xT_n(x) = 1/2\,(T_{n+1}(x) + T_{n-1}(x))$$
$$T_n'(x) = \frac{n\,(T_{n-1}(x) - T_{n+1}(x))}{2(1-x^2)}.$$

$$x \mapsto X := S^{-1},$$
$$\frac{d}{dx} \mapsto D := (n+1)S.$$

$$x \mapsto X := \frac{S+S^{-1}}{2},$$
$$\frac{d}{dx} \mapsto D := \frac{(n+1)S - (n-1)S^{-1}}{2(1-X^2)} = \frac{2n}{S^{-1}-S}.$$

$$(4+x^2)\left(\frac{d}{dx}\right)^2 + 2x\frac{d}{dx}$$
$$\mapsto (4+S^{-2})(n+1)(n+2)S^2 + 2S^{-1}(n+1)S$$
$$= (n+1)\left(4(n+2)S^2 + n\right)$$
$$4(n+2)u_{n+2} + nu_n = 0$$

$$\frac{(n-1)(n+1)\left((n+2)S^2 + 18n + (n-2)S^{-2}\right)}{((n-1)S^2 - 2n + (n+1)S^{-2})},$$

$$(n+2)c_{n+2} + 18nc_n + (n-2)c_{n-2} = 0.$$

Morphisms of Rings of Operators 2

Definition

The Chebyshev morphism φ is defined by:

$$\varphi(x) = \frac{1}{2}\left(S + S^{-1}\right) \text{ and } \varphi\left(\frac{d}{dx}\right) = \frac{2n}{-S + S^{-1}}.$$

Theorem (BenoitSalvy2009)

$f \in \mathcal{C}^k$, $L = \sum p_i \left(\frac{d}{dx}\right)^i$ a linear differential operator of order k such that $L \cdot f = 0$. Suppose that either of the following holds:

(i). $\displaystyle\int_{-1}^{1} \frac{f^{(k)}(x)}{\sqrt{1-x^2}}\,dx$ *is convergent;*

(ii). $\displaystyle\int_{-1}^{1} \frac{(1-x^2)^k f^{(k)}(x)}{\sqrt{1-x^2}}\,dx$ *is convergent and* $(1-x^2)^i | p_i$, $i = 0, \ldots, k$.

Then, the Chebyshev coefficients of f are cancelled by any numerator of $\varphi(L)$.

New and Fast Algorithm[†]

Theorem

If the order is at most k and the degrees of p_i are at most k,

- *Paszkowski (1975) and Lewanowicz (1976):* $\mathcal{O}(k^4)$ *arithmetic operations.*
- *New:* $\mathcal{O}(k^\omega)$ *arithmetic operations.*

ω is a feasible exponent for matrix multiplication $(2 \leq \omega \leq 3)$

---

†

[BenoitSalvy, 2012] A.B., Bruno Salvy,
Chebyshev Expansions for Solutions of Linear Differential Equations,
ISSAC 2009.

New and Fast Algorithm†

Theorem

If the order is at most k and the degrees of p_i are at most k,

- *Paszkowski (1975) and Lewanowicz (1976): $\mathcal{O}(k^4)$ arithmetic operations.*
- *New: $\mathcal{O}(k^\omega)$ arithmetic operations.*

ω is a feasible exponent for matrix multiplication $(2 \leq \omega \leq 3)$

Idea for the new algorithm: Compute the numerator of a fraction of recurrence operators using a divide-and-conquer method.

---

†

[BenoitSalvy, 2012] A.B., Bruno Salvy,
Chebyshev Expansions for Solutions of Linear Differential Equations,
ISSAC 2009.

III Computation of the Chebyshev Coefficients by Hadamard Product

Hadamard Product for Chebyshev Expansions†

Hyp: f is analytic in the closed unit disk. Then, there exists u_n and c_n such that

$$f(x) = \sum_{n \in \mathbb{N}} u_n x^n = \sum_{n \in \mathbb{N}} c_n T_n(x).$$

† extension of Thacher 1964

DDMF[†]

[-] 8. Chebyshev Expansion over $[-1, 1]$

- Chebyshev expansion:

$$\operatorname{erf}(x) = \sum_{n=0}^{\infty} 2\,\frac{4^{-n}(-1)^n\,{}_1F_1(1/2+n;\,2n+2;-1)\,T_{2n+1}(x)}{\sqrt{\pi}\,(2n+1)\,n!}.$$

- First terms and polynomial approximation:

$$\operatorname{erf}(x) = 0.904347\,T_1(x) - 0.0661130\,T_3(x) + 0.00472936\,T_5(x) + \dots$$

$$\operatorname{erf}(x) \approx 1.12633280\,x - 0.35903920\,x^3 + 0.07566976\,x^5.$$

order = 6 Submit

- The coefficients c_n in the Chebyshev expansion $\operatorname{erf}(x) = \sum_{n=0}^{\infty} c_n T_n(x)$ satisfy the recurrence

$$(n^2+3n)\,c(n) + (2n^3+12n^2+24n+16)\,c(n+2) + (-n^2-5n-4)\,c(n+4) = 0.$$

[†]ICMS 2010, B., Chyzak, Darasse, Gerhold, Mezzarobba and Salvy (`http://ddmf.msr-inria.inria.fr`).

Hadamard Product for Chebyshev Expansions[†]

Hyp: f is analytic in the closed unit disk. Then, there exists u_n and c_n such that

$$f(xt) = \sum_{n \in \mathbb{N}} u_n x^n t^n = \sum_{n \in \mathbb{N}} c_n(t) T_n(x).$$

[†] extension of Thacher 1964

Hadamard Product for Chebyshev Expansions[†]

Hyp: f is analytic in the closed unit disk. Then, there exists u_n and c_n such that

$$f(xt) = \sum_{n\in\mathbb{N}} u_n x^n t^n = \sum_{n\in\mathbb{N}} c_n(t) T_n(x).$$

Inner product with $T_n(x)$

$$c_n(t) = \frac{2}{\pi}\int_{-1}^{1} \frac{\sum_{k\in\mathbb{N}} u_k x^k t^k T_n(x)}{\sqrt{1-x^2}} dx = \frac{2}{\pi}\sum_{k\in\mathbb{N}} u_k t^k \underbrace{\int_{-1}^{1} \frac{x^k T_n(x)}{\sqrt{1-x^2}} dx}_{\text{independant of } f} = \sum_{k\in\mathbb{N}} u_k g_{n,k} t^k.$$

[†] extension of Thacher 1964

Hadamard Product for Chebyshev Expansions[†]

Hyp: f is analytic in the closed unit disk. Then, there exists u_n and c_n such that

$$f(xt) = \sum_{n\in\mathbb{N}} u_n x^n t^n = \sum_{n\in\mathbb{N}} c_n(t) T_n(x).$$

Inner product with $T_n(x)$

$$c_n(t) = \frac{2}{\pi}\int_{-1}^{1} \frac{\sum_{k\in\mathbb{N}} u_k x^k t^k T_n(x)}{\sqrt{1-x^2}} dx = \frac{2}{\pi}\sum_{k\in\mathbb{N}} u_k t^k \underbrace{\int_{-1}^{1} \frac{x^k T_n(x)}{\sqrt{1-x^2}} dx}_{\text{independant of } f} = \sum_{k\in\mathbb{N}} u_k g_{n,k} t^k.$$

$$g_n(t) = \sum_{k\in\mathbb{N}} g_{n,k} t^k = \sum_{k\in\mathbb{N}} 2^{1-2k-n} \binom{2k+n}{k+n} t^{2k+n} = \frac{2t^n}{\left(1+\sqrt{1-t^2}\right)^n \sqrt{1-t^2}}.$$

[†] extension of Thacher 1964

Hadamard Product for Chebyshev Expansions[†]

Hyp: f is analytic in the closed unit disk. Then, there exists u_n and c_n such that

$$f(xt) = \sum_{n\in\mathbb{N}} u_n x^n t^n = \sum_{n\in\mathbb{N}} c_n(t) T_n(x).$$

Inner product with $T_n(x)$

$$c_n(t) = \frac{2}{\pi}\int_{-1}^{1} \frac{\sum_{k\in\mathbb{N}} u_k x^k t^k T_n(x)}{\sqrt{1-x^2}} dx = \frac{2}{\pi}\sum_{k\in\mathbb{N}} u_k t^k \underbrace{\int_{-1}^{1} \frac{x^k T_n(x)}{\sqrt{1-x^2}} dx}_{\text{independant of } f} = \sum_{k\in\mathbb{N}} u_k g_{n,k} t^k.$$

$$g_n(t) = \sum_{k\in\mathbb{N}} g_{n,k} t^k = \sum_{k\in\mathbb{N}} 2^{1-2k-n}\binom{2k+n}{k+n} t^{2k+n} = \frac{2t^n}{\left(1+\sqrt{1-t^2}\right)^n \sqrt{1-t^2}}.$$

$$c_n(t) = f(t) \odot g_n(t)$$

[†] extension of Thacher 1964

Application to the Closed Form of Coefficients

Idea: If $f = \sum u_n x^n$ is m-hypergeometric, i.e. u_{n+m}/u_n is rational, then $f \odot g_k$ is 2m-hypergeometric.

Application to the Closed Form of Coefficients

Idea: If $f = \sum u_n x^n$ is m-hypergeometric, i.e. u_{n+m}/u_n is rational, then $f \odot g_k$ is 2m-hypergeometric.

Corollary (Luke, 1969):

$$ {}_pF_q\left(\begin{matrix} a_1;\dots;a_p \\ b_1;\dots;b_q \end{matrix} \middle| xt\right) = \sum_{k\in\mathbb{N}} c_k(t)T_k(x), $$

$$ c_k(t) = $$

$$ \frac{2}{2^k}\frac{(a_1)_k\dots(a_p)_k}{(b_1)_k\dots(b_q)_k}\frac{t^k}{k!}\,{}_{2p}F_{2q+1}\left(\begin{matrix} \frac{a_1+k}{2};\frac{a_1+k+1}{2};\dots;\frac{a_p+k}{2};\frac{a_p+k+1}{2} \\ \frac{b_1+k}{2};\frac{b_1+k+1}{2};\dots;\frac{b_q+k}{2};\frac{b_q+k+1}{2};k+1 \end{matrix} \middle| \frac{t^2}{4^{q-p+1}}\right). $$

Applications to Special Functions

Function	Chebyshev expansion
$\exp(xt)$	$\sum_{n=0}^{\infty}{}' 2\,\mathrm{I}_n(t) T_n(x)$
$\sin(xt)$	$2\sum_{n\in\mathbb{N}}(-1)^n J_{2n+1}(t) T_{2n+1}(x)$
$\operatorname{erf}(xt)$	$\sum_{n\in\mathbb{N}} \frac{2}{\sqrt{\pi}} \frac{(-1)^n}{4^n} \frac{t^{2n+1}}{(2n+1)n!}\, {}_1F_1\left(\begin{matrix} n+\frac{1}{2} \\ 2n+2 \end{matrix} \middle\vert -t^2\right) T_{2n+1}(x)$
$\mathrm{Si}(xt)$	$\sum_{n\in\mathbb{N}} \frac{4^{-n}(-1)^n}{(2n+1)!(2n+1)}\, {}_1F_1\left(\begin{matrix} n+\frac{1}{2} \\ 2n+2, n+\frac{3}{2} \end{matrix} \middle\vert -\frac{1}{4}t^2\right) T_{2n+1}(x)$

Applications to Special Functions

Function	Chebyshev expansion
$\exp(xt)$	$\sum_{n=0}^{\infty}{}' 2\,\mathrm{I}_n(t)T_n(x)$
$\sin(xt)$	$2\sum_{n\in\mathbb{N}}(-1)^n J_{2n+1}(t)T_{2n+1}(x)$
$\mathrm{erf}(xt)$	$\sum_{n\in\mathbb{N}} \frac{2}{\sqrt{\pi}} \frac{(-1)^n}{4^n} \frac{t^{2n+1}}{(2n+1)n!}\, {}_1F_1\left(\begin{matrix} n+\frac{1}{2} \\ 2n+2 \end{matrix}\,\middle|\, -t^2\right) T_{2n+1}(x)$
$\mathrm{Si}(xt)$	$\sum_{n\in\mathbb{N}} \frac{4^{-n}(-1)^n}{(2n+1)!(2n+1)}\, {}_1F_1\left(\begin{matrix} n+\frac{1}{2} \\ 2n+2, n+\frac{3}{2} \end{matrix}\,\middle|\, -\frac{1}{4}t^2\right) T_{2n+1}(x)$
$\mathrm{Ai}(xt)$	$\sum_{n\in\mathbb{N}}{}' \frac{1}{81^n n!}\Bigg(-\frac{3^{2/3}}{144}\frac{1}{\Gamma(n+4/3)}\, {}_2F_5\left(\begin{matrix} 1/2\ n+4/3; 1/2\ n+5/6 \\ 4/3;5/3;n+1;n+4/3;n+5/3 \end{matrix}\,\middle|\, \frac{t^6}{1296}\right) + \frac{\sqrt[3]{3}}{3}\frac{2}{\Gamma(n+2/3)}\, {}_2F_5\left(\begin{matrix} 1/2n+2/3; 1/2\ n+1/6 \\ 1/3;2/3;n+1;n+2/3;n+1/3 \end{matrix}\,\middle|\, \frac{t^6}{1296}\right)\Bigg) T_{3n}(x) + \cdots$

Application to Numerical Computation

$$f(x) = \sum_{n\in\mathbb{N}} c_n(1)T_n(x), \quad \text{with} \quad c_n(t) = f(t) \odot \frac{2t^n}{(1+\sqrt{1-t^2})^n\sqrt{1-t^2}}$$

Application to Numerical Computation

$$f(x) = \sum_{n\in\mathbb{N}} c_n(1)T_n(x), \text{ with } c_n(t) = f(t) \odot \frac{2t^n}{(1+\sqrt{1-t^2})^n\sqrt{1-t^2}}$$

$c_n(t) = \sum_{k=0}^{\infty} c_{n,k}t^k$ is solution of a linear differential equation, then $c_{n,k}$ are solutions of a linear recurrence

Application to Numerical Computation

$$f(x) = \sum_{n\in\mathbb{N}} c_n(1) T_n(x), \text{ with } c_n(t) = f(t) \odot \frac{2t^n}{(1+\sqrt{1-t^2})^n \sqrt{1-t^2}}$$

$c_n(t) = \sum_{k=0}^{\infty} c_{n,k} t^k$ is solution of a linear differential equation, then $c_{n,k}$ are solutions of a linear recurrence

Application: Algorithm for the computation of $c_n(1)$ in $\tilde{\mathcal{O}}(\log(\epsilon^{-1}) + n)$ bit operations
INPUT: n, ϵ
OUTPUT: $\widetilde{c}_n(1)$ such that $|\widetilde{c}_n(1) - c_n(1)| < \epsilon$

Application to Numerical Computation

$$f(x) = \sum_{n \in \mathbb{N}} c_n(1) T_n(x), \ \text{ with } \ c_n(t) = f(t) \odot \frac{2t^n}{(1+\sqrt{1-t^2})^n \sqrt{1-t^2}}$$

$c_n(t) = \sum_{k=0}^{\infty} c_{n,k} t^k$ is solution of a linear differential equation, then $c_{n,k}$ are solutions of a linear recurrence

Application: Algorithm for the computation of $c_n(1)$ in $\tilde{\mathcal{O}}(\log(\epsilon^{-1}) + n)$ bit operations

INPUT: n, ϵ

OUTPUT: $\widetilde{c}_n(1)$ such that $|\widetilde{c}_n(1) - c_n(1)| < \epsilon$

- Computation of the recurrence satisfied by $c_{n,k}$, plus initial conditions ($\tilde{\mathcal{O}}(n)$ ops) (Using binary splitting)
- Use of the Mezzarobba-Salvy algorithm to compute N such that $|\sum_{n=N+1}^{\infty} c_{n,k}| < \epsilon$ ($\mathcal{O}(\log(\epsilon^{-1}))$ ops)
- Computation of $\sum_{k=0}^{N} c_{n,k}$ using the recurrence ($\tilde{\mathcal{O}}(N) = \tilde{\mathcal{O}}(\log(\epsilon^{-1}))$ ops) (Using binary splitting)

Application to Numerical Computation

$$f(x) = \sum_{n\in\mathbb{N}} c_n(1)T_n(x), \text{ with } c_n(t) = f(t) \odot \frac{2t^n}{(1+\sqrt{1-t^2})^n\sqrt{1-t^2}}$$

$c_n(t) = \sum_{k=0}^{\infty} c_{n,k}t^k$ is solution of a linear differential equation, then $c_{n,k}$ are solutions of a linear recurrence

Application: Algorithm for the computation of $c_n(1)$ in $\tilde{\mathcal{O}}(\log(\epsilon^{-1}) + n)$ bit operations
INPUT: n, ϵ
OUTPUT: $\widetilde{c}_n(1)$ such that $|\widetilde{c}_n(1) - c_n(1)| < \epsilon$

- Computation of the recurrence satisfied by $c_{n,k}$, plus initial conditions ($\tilde{\mathcal{O}}(n)$ ops) (Using binary splitting)
- Use of the Mezzarobba-Salvy algorithm to compute N such that $|\sum_{n=N+1}^{\infty} c_{n,k}| < \epsilon$ ($\mathcal{O}(\log(\epsilon^{-1}))$ ops)
- Computation of $\sum_{k=0}^{N} c_{n,k}$ using the recurrence ($\tilde{\mathcal{O}}(N) = \tilde{\mathcal{O}}(\log(\epsilon^{-1}))$ ops) (Using binary splitting)

To compute ℓ coefficients, the complexity becomes $\tilde{\mathcal{O}}(\ell \log(\epsilon^{-1}))$.

IV Numerical Evaluation using the Recurrence Relation

Direct Application of the Recurrence

Taylor: $\exp(x) = \sum \frac{1}{n!} x^n$
Rec: $u_{n+1} = \frac{u_n}{n+1}$

$u_0 = 1$	$1/0! = 1$
$u_1 = 1$	$1/1! = 1$
$u_2 = 0.5$	$1/2! = 0.5$
$\vdots$	$\vdots$
$u_{50} \approx 3.28 \times 10^{-65}$	$1/50! \approx 3.28 \times 10^{-65}$

Direct Application of the Recurrence

Taylor: $\exp(x) = \sum \frac{1}{n!} x^n$
Rec: $u_{n+1} = \frac{u_n}{n+1}$

$u_0 = 1$ $1/0! = 1$

$u_1 = 1$ $1/1! = 1$

$u_2 = 0.5$ $1/2! = 0.5$

$\vdots$ $\vdots$

$u_{50} \approx 3.28 \times 10^{-65}$ $1/50! \approx 3.28 \times 10^{-65}$

Chebyshev: $\exp(x) = \sum \mathrm{I}_n(1) T_n(x)$
Rec: $u_{n+1} = -2nu_n + u_{n-1}$

$u_0 = 1.266$ $\mathrm{I}_0(1) \approx 1.266$

$u_1 = 0.565$ $\mathrm{I}_1(1) \approx 0.565$

$u_2 \approx 0.136$ $\mathrm{I}_2(1) \approx 0.136$

$\vdots$ $\vdots$

Direct Application of the Recurrence

Taylor: $\exp(x) = \sum \frac{1}{n!} x^n$
Rec: $u_{n+1} = \frac{u_n}{n+1}$

$$u_0 = 1 \qquad 1/0! = 1$$
$$u_1 = 1 \qquad 1/1! = 1$$
$$u_2 = 0.5 \qquad 1/2! = 0.5$$
$$\vdots \qquad \vdots$$
$$u_{50} \approx 3.28 \times 10^{-65} \qquad 1/50! \approx 3.28 \times 10^{-65}$$

Chebyshev: $\exp(x) = \sum \mathrm{I}_n(1) T_n(x)$
Rec: $u_{n+1} = -2n u_n + u_{n-1}$

$$u_0 = 1.266 \qquad \mathrm{I}_0(1) \approx 1.266$$
$$u_1 = 0.565 \qquad \mathrm{I}_1(1) \approx 0.565$$
$$u_2 \approx 0.136 \qquad \mathrm{I}_2(1) \approx 0.136$$
$$\vdots \qquad \vdots$$
$$u_{50} \approx 4.450 \times 10^{67} \qquad \mathrm{I}_{50}(1) \approx 2.934 \times 10^{-80}$$

Direct Application of the Recurrence

Taylor: $\exp(x) = \sum \frac{1}{n!} x^n$
Rec: $u_{n+1} = \frac{u_n}{n+1}$

$u_0 = 1$	$1/0! = 1$
$u_1 = 1$	$1/1! = 1$
$u_2 = 0.5$	$1/2! = 0.5$
$\vdots$	$\vdots$
$u_{50} \approx 3.28 \times 10^{-65}$	$1/50! \approx 3.28 \times 10^{-65}$

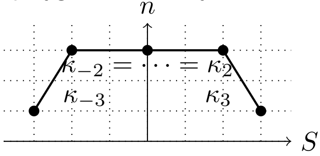
Chebyshev: $\exp(x) = \sum \mathrm{I}_n(1) T_n(x)$
Rec: $u_{n+1} = -2n u_n + u_{n-1}$

$u_0 = 1.266$	$\mathrm{I}_0(1) \approx 1.266$
$u_1 = 0.565$	$\mathrm{I}_1(1) \approx 0.565$
$u_2 \approx 0.136$	$\mathrm{I}_2(1) \approx 0.136$
$\vdots$	$\vdots$
$u_{50} \approx 4.450 \times 10^{67}$	$\mathrm{I}_{50}(1) \approx 2.934 \times 10^{-80}$

Study of the Chebyshev recurrence

If u_n is solution, then there exists another solution $v_n \sim \frac{1}{u_n}$

Newton polygon of a Chebyshev recurrence



Hadamard Product and Linear Recurrence

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t) T_n(x) = \sum_{n\in\mathbb{N}} \sum_{k\in\mathbb{N}} c_{n,k} t^k T_n(x).$$

Hadamard Product and Linear Recurrence

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t) T_n(x) = \sum_{n\in\mathbb{N}} \sum_{k\in\mathbb{N}} c_{n,k} t^k T_n(x).$$

We can compute

$$t\,\mathrm{I}_n(t) - 2(n+1)\,\mathrm{I}_{n+1}(t) - t\,\mathrm{I}_{n+2}(t) = 0.$$

Hadamard Product and Linear Recurrence

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t) T_n(x) = \sum_{n\in\mathbb{N}} \sum_{k\in\mathbb{N}} c_{n,k} t^k T_n(x).$$

We can compute

$$t \sum_{k\in\mathbb{N}} c_{n,k} t^k - 2(n+1) \sum_{k\in\mathbb{N}} c_{n+1,k} t^k - t \sum_{k\in\mathbb{N}} c_{n+2,k} t^k = 0.$$

Hadamard Product and Linear Recurrence

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t) T_n(x) = \sum_{n\in\mathbb{N}} \sum_{k\in\mathbb{N}} c_{n,k} t^k T_n(x).$$

We can compute

$$t\sum_{k\in\mathbb{N}} c_{n,k} t^k - 2(n+1) \sum_{k\in\mathbb{N}} c_{n+1,k} t^k - t \sum_{k\in\mathbb{N}} c_{n+2,k} t^k = 0 \quad \textcolor{red}{\bmod\ t^{N+2}}.$$

Hadamard Product and Linear Recurrence

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t)T_n(x) = \sum_{n\in\mathbb{N}}\sum_{k\in\mathbb{N}} c_{n,k}t^kT_n(x).$$

We can compute

$$t\sum_{k\in\mathbb{N}} c_{n,k}t^k - 2(n+1)\sum_{k\in\mathbb{N}} c_{n+1,k}t^k - t\sum_{k\in\mathbb{N}} c_{n+2,k}t^k = 0 \mod t^{N+2}.$$

We deduce a non-homogeneous linear recurrence in n

$$t\sum_{k=0}^{N} c_{n,k}t^k - 2(n+1)\sum_{k=0}^{N} c_{n+1,k}t^k - t\sum_{k=0}^{N} c_{n+2,k}t^k = 2(n+1)c_{n+1,N+1}t^{N+1}.$$

satisfied by

$$\sum_{k=0}^{N} c_{n,k}t^k.$$

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t) T_n(x) = \sum_{n\in\mathbb{N}} \sum_{k\in\mathbb{N}} c_{n,k} t^k T_n(x).$$

We can compute

$$t \sum_{k\in\mathbb{N}} c_{n,k} t^k - 2(n+1) \sum_{k\in\mathbb{N}} c_{n+1,k} t^k - t \sum_{k\in\mathbb{N}} c_{n+2,k} t^k = 0 \mod t^{N+2}.$$

We deduce a non-homogeneous linear recurrence in n

$$\sum_{k=0}^{N} c_{n,k} - 2(n+1) \sum_{k=0}^{N} c_{n+1,k} - \sum_{k=0}^{N} c_{n+2,k} = 2(n+1) c_{n+1,N+1}.$$

satisfied by

$$\sum_{k=0}^{N} c_{n,k}.$$

Hadamard Product and Linear Recurrence

$$\exp(xt) = \sum_{n\in\mathbb{N}} \mathrm{I}_n(t)T_n(x) = \sum_{n\in\mathbb{N}} \sum_{k\in\mathbb{N}} c_{n,k}t^k T_n(x).$$

We can compute

$$t\sum_{k\in\mathbb{N}} c_{n,k}t^k - 2(n+1)\sum_{k\in\mathbb{N}} c_{n+1,k}t^k - t\sum_{k\in\mathbb{N}} c_{n+2,k}t^k = 0 \mod t^{N+2}.$$

We deduce a non-homogeneous linear recurrence in n

$$\sum_{k=0}^{N} c_{n,k} - 2(n+1)\sum_{k=0}^{N} c_{n+1,k} - \sum_{k=0}^{N} c_{n+2,k} = 2(n+1)c_{n+1,N+1}.$$

satisfied by

$$\sum_{k=0}^{N} c_{n,k}.$$

Application: Algorithm to compute the first ℓ coefficients $\widetilde{c}_n(1) = \sum_{k=0}^{N} c_{n,k}$ such that $|\widetilde{c}_n(1) - f| < \epsilon$ in $\tilde{\mathcal{O}}(\ell + \log(\epsilon^{-1}))$ arithmetic operations.

Miller's Method

Example

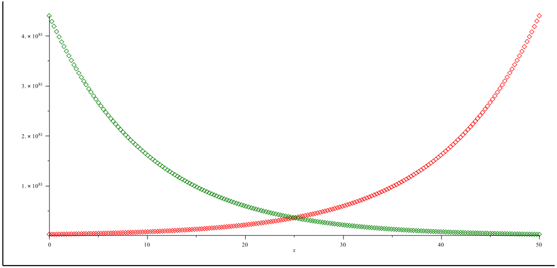
$y(x) = e^x = \sum_{n=0}^{\infty} c_n \, T_n(x)$ $\qquad$ $c_{n+1} + 2n \, c_n - c_{n-1} = 0$

Miller's Method

Example

$y(x) = e^x = \sum_{n=0}^{\infty} c_n\, T_n(x)$ $\qquad c_{n+1} + 2n\, c_n - c_{n-1} = 0$

2 solutions: $\mathrm{I}_n(1)$ and $\mathrm{K}_n(1)$



---

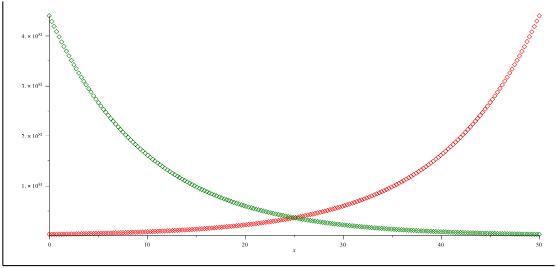
u_0	c_0
u_1	c_1
u_2	c_2
$\vdots$	$\vdots$
u_{50}	c_{50}
u_{51}	c_{51}
u_{52}	c_{52}

Miller's Method

Example

$y(x) = e^x = \sum_{n=0}^{\infty} c_n \, T_n(x)$ $c_{n+1} + 2n\, c_n - c_{n-1} = 0$

2 solutions: $\mathrm{I}_n(1)$ and $\mathrm{K}_n(1)$



---

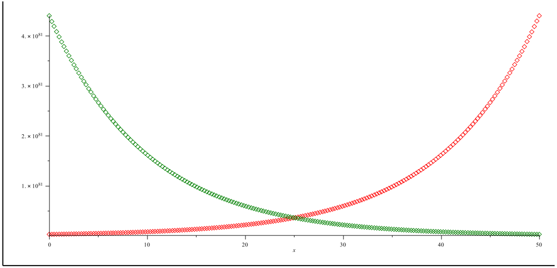
u_0			c_0
u_1			c_1
u_2			c_2
$\vdots$			$\vdots$
u_{50}	$\approx$	$1.02 \cdot 10^2$	c_{50}
u_{51}	$=$	1	c_{51}
u_{52}	$=$	0	c_{52}

Miller's Method

Example

$y(x) = e^x = \sum_{n=0}^{\infty} c_n T_n(x)$ $c_{n+1} + 2n\,c_n - c_{n-1} = 0$

2 solutions: $\mathrm{I}_n(1)$ and $\mathrm{K}_n(1)$



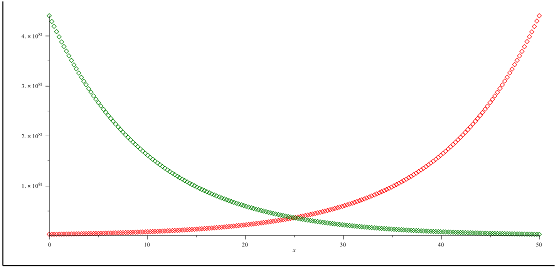
$$
\begin{aligned}
u_0 &\approx -4.40 \cdot 10^{81} \\
u_1 &\approx 1.96 \cdot 10^{81} \\
u_2 &\approx -4.72 \cdot 10^{80} \\
&\vdots \\
u_{50} &\approx 1.02 \cdot 10^{2} \\
u_{51} &= 1 \\
u_{52} &= 0
\end{aligned}
\qquad
\begin{aligned}
&c_0 \\
&c_1 \\
&c_2 \\
&\vdots \\
&c_{50} \\
&c_{51} \\
&c_{52}
\end{aligned}
$$

Miller's Method

Example

$y(x) = e^x = \sum_{n=0}^{\infty} c_n \, T_n(x)$ $\qquad c_{n+1} + 2n\, c_n - c_{n-1} = 0$

2 solutions: $\mathrm{I}_n(1)$ and $\mathrm{K}_n(1)$



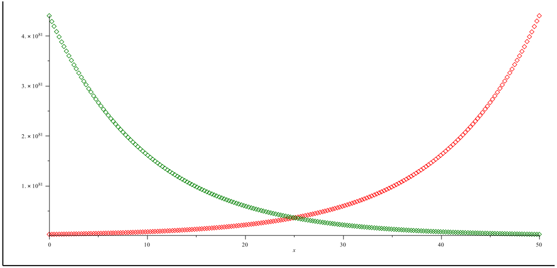
$$S = \sum_{n=0}^{50} u_n \, T_n(0) \approx -3.48 \cdot 10^{81}$$

$$\begin{aligned} u_0 &\approx -4.40 \cdot 10^{81} & c_0 \\ u_1 &\approx 1.96 \cdot 10^{81} & c_1 \\ u_2 &\approx -4.72 \cdot 10^{80} & c_2 \\ &\vdots & \vdots \\ u_{50} &\approx 1.02 \cdot 10^{2} & c_{50} \\ u_{51} &= 1 & c_{51} \\ u_{52} &= 0 & c_{52} \end{aligned}$$

Example

$y(x) = e^x = \sum_{n=0}^{\infty} c_n\, T_n(x)$ $\qquad$ $c_{n+1} + 2n\, c_n - c_{n-1} = 0$

2 solutions: $\mathrm{I}_n(1)$ and $\mathrm{K}_n(1)$



$$S = \sum_{n=0}^{50} u_n\, T_n(0) \approx -3.48 \cdot 10^{81}$$

$$\begin{aligned}
u_0 &\approx -4.40 \cdot 10^{81} \\
u_1 &\approx 1.96 \cdot 10^{81} \\
u_2 &\approx -4.72 \cdot 10^{80} \\
&\vdots \\
u_{50} &\approx 1.02 \cdot 10^{2} \\
u_{51} &= 1 \\
u_{52} &= 0
\end{aligned}$$

$g_n := u_n / S$

$$\begin{aligned}
c_0 &\approx 1.27 \\
c_1 &\approx -5.65 \cdot 10^{-1} \\
c_2 &\approx 1.36 \cdot 10^{-1} \\
&\vdots \\
c_{50} &\approx 2.93 \cdot 10^{-80} \\
c_{51} &\approx 2.88 \cdot 10^{-82} \\
c_{52} &\approx 0
\end{aligned}$$

Algorithm from the Second method†

Algorithm

Input: *a differential equation of order r with boundary conditions*
Output: *a polynomial approximation of degree N of the solution*

- *compute the Chebyshev recurrence (order $2s \geq 2r$);*
- *for i from 1 to s*
  - *using the recurrence relation backwards, compute the first N coefficients of the sequence $u^{[i]}$ starting with the initial conditions*

$$\left(u^{[i]}(N+2s), \cdots, u^{[i]}(N+i), \cdots, u^{[i]}(N+1)\right) = (0, \cdots, 1, \cdots, 0)\,;$$

- *combine the s sequences $u^{[i]}$ according to the r boundary conditions and the $s-r$ symmetry relations.*

---

†

[BenoitMezzarobbaJodeș, 2012] A.B., Mioara Joldeș and Marc Mezzarobba, Rigorous uniform approximation of D-finite functions using Chebyshev expansions, In preparation.

Algorithm from the Second method†

Algorithm

Input: *a differential equation of order r with boundary conditions*
Output: *a polynomial approximation of degree N of the solution*

- *compute the Chebyshev recurrence (order $2s \geq 2r$);*
- *for i from 1 to s*
  - *using the recurrence relation backwards, compute the first N coefficients of the sequence $u^{[i]}$ starting with the initial conditions*

$$\Big(u^{[i]}(N+2s), \cdots, u^{[i]}(N+i), \cdots, u^{[i]}(N+1)\Big) = (0, \cdots, 1, \cdots, 0)\,;$$

- *combine the s sequences $u^{[i]}$ according to the r boundary conditions and the $s - r$ symmetry relations.*

Theorem

This algorithm runs in linear time.

†

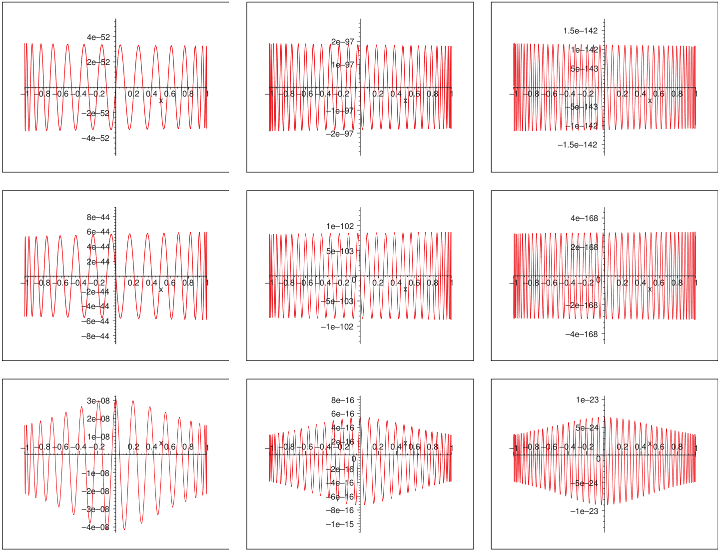
[BenoitMezzarobbaJodeș, 2012] A.B., Mioara Joldeș and Marc Mezzarobba,
Rigorous uniform approximation of D-finite functions using Chebyshev expansions,
In preparation.

Quality of the Approximation

$\frac{e^{x/2}}{\sqrt{x+16}}$

$\frac{3\cos x - \sin x}{2}$

$e^{1/(1+2x^2)}$



degree = 30 | degree = 60 | degree = 90

Validation of the Polynomial

Algorithm

- Input: Differential operator, initial conditions and a polynomial of degree d
- Output: R such that $||f - p||_\infty < R$ and R is not too large

Validation of the Polynomial

Algorithm

- Input: Differential operator, initial conditions and a polynomial of degree d
- Output: R such that $||f-p||_\infty < R$ and R is not too large

$$\log_{10} \frac{\text{(bounds computed)}}{\|f-p\|_\infty}$$

$\frac{e^{x/2}}{\sqrt{x+16}}$	4,8	0,58	0, 57
$\frac{3\cos x - \sin x}{2}$	3,1	3,7	4,1
$e^{1/(1+2x^2)}$	0,57	0,56	0,56
	degree = 30	degree = 60	degree = 90

V Generalized Fourier Series

Generalized Fourier Series

$$f(x) = \sum a_n \psi_n(x)$$

Some Examples

$$\sin(x) = 2\sum_{n=0}^{\infty} (-1)^n \mathrm{J}_{2n}(x)$$

$$\arccos(x) = \frac{1}{2\pi} T_0(x) - \sum_{n=0}^{\infty} \frac{4}{(2n+1)^2 \pi} T_{2n+1}(x)$$

$$\operatorname{erf}(x) = 2\sum_{n=0}^{\infty} \left(-\frac{1}{4}\right)^n \frac{1}{\sqrt{\pi}\,(2n+1)\,n!}\, {}_1F_1\left(\left.\begin{matrix} n+\frac{1}{2} \\ 2n+2 \end{matrix}\right| -x\right)$$

Generalized Fourier Series

$$f(x) = \sum a_n \psi_n(x)$$

Some Examples

$$\sin(x) = 2\sum_{n=0}^{\infty}(-1)^n \mathrm{J}_{2n}(x)$$

$$\arccos(x) = \frac{1}{2\pi} T_0(x) - \sum_{n=0}^{\infty} \frac{4}{(2n+1)^2 \pi} T_{2n+1}(x)$$

$$\operatorname{erf}(x) = 2\sum_{n=0}^{\infty}\left(-\frac{1}{4}\right)^n \frac{1}{\sqrt{\pi}(2n+1)n!} {}_1F_1\left(\begin{matrix} n+\frac{1}{2} \\ 2n+2 \end{matrix} \middle| -x\right)$$

Aim: general algorithm for these series

Framework

Families of functions $\psi_n(x)$ with two special properties

Mult by x ($\mathcal{P}_x$)

$$\mathcal{R}ec_{x2}\left(x\psi_n(x)\right) = \mathcal{R}ec_{x1}\left(\psi_n(x)\right)$$

Examples

- Monomial polynomials ($M_n = x^n$)
- All orthogonal polynomials
- Bessel functions
- Legendre functions
- Parabolic cylinder functions

Framework

Families of functions $\psi_n(x)$ with two special properties

Mult by x ($\mathcal{P}_x$)

$$\mathcal{R}ec_{x2}\left(x\psi_n(x)\right) = \mathcal{R}ec_{x1}\left(\psi_n(x)\right)$$

Examples

- Monomial polynomials ($M_n = x^n$) $\qquad xM_n = M_{n+1}$
- All orthogonal polynomials $\qquad 2xT_n(x) = T_{n+1}(x) + T_{n-1}(x)$
- Bessel functions $\qquad \frac{1}{n}\left(x\mathrm{J}_{n+1} - x\mathrm{J}_{n-1}\right) = 2\mathrm{J}_n$
- Legendre functions
- Parabolic cylinder functions

Framework

Families of functions $\psi_n(x)$ with two special properties

Mult by x ($\mathcal{P}_x$)

$$\mathcal{R}ec_{x2}\left(x\psi_n(x)\right) = \mathcal{R}ec_{x1}\left(\psi_n(x)\right)$$

Differentiation ($\mathcal{P}_\partial$)

$$\mathcal{R}ec_{\partial 2}\left(\psi_n'(x)\right) = \mathcal{R}ec_{\partial 1}\left(\psi_n(x)\right)$$

Examples

- Monomial polynomials
- Classical orthogonal polynomials
- Bessel functions
- Legendre functions
- Parabolic cylinder functions

Framework

Families of functions $\psi_n(x)$ with two special properties

Mult by x $(\mathcal{P}_x)$

$$\mathcal{R}ec_{x2}\left(x\psi_n(x)\right) = \mathcal{R}ec_{x1}\left(\psi_n(x)\right)$$

Differentiation $(\mathcal{P}_\partial)$

$$\mathcal{R}ec_{\partial 2}\left(\psi'_n(x)\right) = \mathcal{R}ec_{\partial 1}\left(\psi_n(x)\right)$$

Examples

- Monomial polynomials
- Classical orthogonal polynomials
- Bessel functions
- Legendre functions
- Parabolic cylinder functions

$$M'_n = nM_{n-1}$$

$$\frac{1}{n+1}T'_{n+1}(x) - \frac{1}{n-1}T'_{n-1}(x) = 2T_n(x)$$

$$2\mathrm{J}'_n(x) = \mathrm{J}_{n-1}(x) - \mathrm{J}_{n+1}(x)$$

Framework

Families of functions $\psi_n(x)$ with two special properties

Mult by x ($\mathcal{P}_x$)

$$\mathcal{R}ec_{x2}\left(x\psi_n(x)\right) = \mathcal{R}ec_{x1}\left(\psi_n(x)\right)$$

Differentiation ($\mathcal{P}_\partial$)

$$\mathcal{R}ec_{\partial 2}\left(\psi'_n(x)\right) = \mathcal{R}ec_{\partial 1}\left(\psi_n(x)\right)$$

This is our data-structure for $\psi_n(x)$

New algorithm to compute the recurrence†

Main Idea

If $\psi_n(x)$ satisfies $(\mathcal{P}_x)$ and $(\mathcal{P}_\partial)$, for any $f(x) = \sum a_n \psi_n(x)$ solution of a linear differential equation with polynomial coefficients, the coefficients a_n are solutions of a linear recurrence relation with polynomial coefficients.

---

†

[BenoitSalvy, 2012] A.B. and Bruno Salvy.
Generalized Fourier Series for Solutions of Linear Differential Equations,
In preparation.

New algorithm to compute the recurrence†

Main Idea

If $\psi_n(x)$ satisfies $(\mathcal{P}_x)$ and $(\mathcal{P}_\partial)$, for any $f(x) = \sum a_n \psi_n(x)$ solution of a linear differential equation with polynomial coefficients, the coefficients a_n are solutions of a linear recurrence relation with polynomial coefficients.

New Contribution

New general algorithm computing the recurrence relation of the coefficients of a Generalized Fourier Series when $\psi(x)$ satisfies $(\mathcal{P}_x)$ and $(\mathcal{P}_\partial)$.

---

†

[BenoitSalvy, 2012] A.B. and Bruno Salvy.
Generalized Fourier Series for Solutions of Linear Differential Equations,
In preparation.

New algorithm to compute the recurrence[†]

Main Idea

If $\psi_n(x)$ satisfies $(\mathcal{P}_x)$ and $(\mathcal{P}_\partial)$, for any $f(x) = \sum a_n \psi_n(x)$ solution of a linear differential equation with polynomial coefficients, the coefficients a_n are solutions of a linear recurrence relation with polynomial coefficients.

New Contribution

New general algorithm computing the recurrence relation of the coefficients of a Generalized Fourier Series when $\psi(x)$ satisfies $(\mathcal{P}_x)$ and $(\mathcal{P}_\partial)$.

$$L(x,\partial) \mapsto \text{numer}(L(X,D))$$

---

†

[BenoitSalvy, 2012] A.B. and Bruno Salvy.
Generalized Fourier Series for Solutions of Linear Differential Equations,
In preparation.

VI Quasi-Optimal Multiplication of Linear Differential Operators

Product of Linear Differential Operators

The product of differential operators is a complexity yardstick:
That of more involved, higher-level, operations on linear differential operators can be reduced to it:

- LCLM, GCRD (van der Hoeven 2011)
- closure properties for differential operators.

van der Hoeven's Algorithm (2002)

$$\Phi_L^{k+d,k} = \begin{pmatrix} L(1)_0 & \cdots & L(x^{k-1})_0 \\ \vdots & & \vdots \\ L(1)_{k+d-1} & \cdots & L(x^{k-1})_{k+d-1} \end{pmatrix} \in \mathbb{K}^{(k+d)\times k}$$

we clearly have

$$\Phi_{KL}^{k+2d,k} = \Phi_K^{k+2d,k+d}\Phi_L^{k+d,k}, \quad \text{for all } k \geqslant 0.$$

Algorithm: when K and L are of degrees d and order r.

Evaluation Computation of $\Phi_K^{2r+2d,2r+d}$ and $\Phi_L^{2r+d,2r}$ from K and L ($\tilde{\mathcal{O}}((r+d)^2)$ ops).

Inner multiplication Computation of the matrix product ($\mathcal{O}((r+d)^\omega)$ ops).

Interpolation Recovery of KL from $\Phi_{KL}^{2r+2d,2r}$ ($\tilde{\mathcal{O}}((r+d)^2)$ ops).

New and Fast Algorithm for the Product[†]

$$\Phi_L^{\alpha_i,k+d,k} = \begin{pmatrix} L(e^{\alpha_i x}) & \cdots & L(x^{k-1}e^{\alpha_i x}) \end{pmatrix} \in \mathbb{K}^{(k+d)\times k}$$

we clearly have

$$\Phi_{KL}^{\alpha_i,k+2d,k} = \Phi_K^{\alpha_i,k+2d,k+d}\Phi_L^{\alpha_i,k+d,k}, \quad \text{for all } k \geqslant 0.$$

Algorithm: when K and L are of degrees d and order r ($r > d$).

- **Evaluation** For $i = 0, \ldots, r/d$, computation of $\Phi_K^{\alpha_i,4d,3d}$ and $\Phi_L^{\alpha_i,3d,2d}$ from K and L.
- **Inner multiplication** Computation of $\mathcal{O}(r/d)$ products of matrices with size $d \times d$.
- **Interpolation** Recovery of KL from $\Phi_{KL}^{\alpha_i,4d,2d}$.

---

†

[BenoitBostanvanderHoeven, 2012] A.B., Alin Bostan and Joris van der Hoeven.
Quasi-Optimal Multiplication of Linear Differential Operators,
FOCS 2012.

New and Fast Algorithm for the Product†

$$\Phi_L^{\alpha_i,k+d,k} = \begin{pmatrix} L(e^{\alpha_i x}) & \cdots & L(x^{k-1}e^{\alpha_i x}) \end{pmatrix} \in \mathbb{K}^{(k+d)\times k}$$

we clearly have

$$\Phi_{KL}^{\alpha_i,k+2d,k} = \Phi_K^{\alpha_i,k+2d,k+d}\Phi_L^{\alpha_i,k+d,k}, \quad \text{for all } k \geqslant 0.$$

Algorithm: when K and L are of degrees d and order r ($r > d$).

Evaluation For $i = 0, \ldots, r/d$, computation of $\Phi_K^{\alpha_i,4d,3d}$ and $\Phi_L^{\alpha_i,3d,2d}$ from K and L ($\tilde{\mathcal{O}}(rd)$ ops).

Inner multiplication Computation of $\mathcal{O}(r/d)$ products of matrices with size $d \times d$ ($\mathcal{O}(rd^{\omega-1})$ ops).

Interpolation Recovery of KL from $\Phi_{KL}^{\alpha_i,4d,2d}$ ($\tilde{\mathcal{O}}(rd)$ ops).

---

†

[BenoitBostanvanderHoeven, 2012] A.B., Alin Bostan and Joris van der Hoeven.
Quasi-Optimal Multiplication of Linear Differential Operators,
FOCS 2012.

VII Conclusion

Summary and Perspectives

Summary

- Fast algorithm for the computation of coefficients of Chebyshev expansions.
- Use of the Chebyshev recurrence relation for this computation.
- New tools to compute this recurrence relation: Fraction of recurrence operators.
- New and fast algorithm for the product of operators.

Summary and Perspectives

Summary

- Fast algorithm for the computation of coefficients of Chebyshev expansions.
- Use of the Chebyshev recurrence relation for this computation.
- New tools to compute this recurrence relation: Fraction of recurrence operators.
- New and fast algorithm for the product of operators.

Perspectives

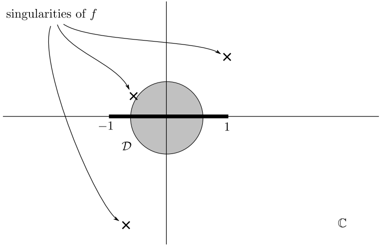
- Use the same idea to compute the coefficients of other generalized Fourier series.
- Use the fast algorithm for the product of operators to design new and fast algorithms for linear differential or recurrence operators.

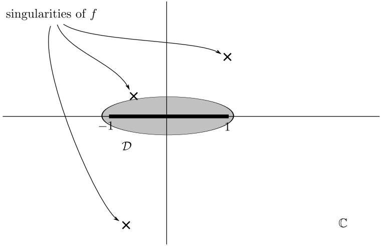
Chebyshev Series vs Taylor Series II

Convergence Domains:

Taylor series:
disk centered at $x_0 = 0$, avoiding the singularities of f

Chebyshev series:
elliptic disk with foci at ± 1, avoids the singularities of f





- Taylor series cannot converge over entire $[-1,1]$ unless all singularities lie outside the unit circle.

✓ Chebyshev series converges over entire $[-1,1]$ as soon as there are no real singularities in $[-1,1]$.