Fast Algorithms for Differential Equations in Positive Characteristic

Alin Bostan (INRIA, France)

joint work with

Éric Schost (University of Western Ontario, Canada)

ISSAC 2009, KIAS, Seoul, Korea, July 30, 2009

Main objects & Aim

- $\mathbb{F}_p$ = the finite field with p elements (p = prime number)
- $L = \ell_0(x) + \ell_1(x)\partial + \cdots + \ell_r(x)\partial^r$ of order r in $\mathbb{F}_p[x]\langle\partial\rangle$

Def: *p*-curvature $\mathbf{A}_p$ of L = the matrix in $\mathscr{M}_r(\mathbb{F}_p(x))$ whose j-th column contains the coefficients of ∂^{p+j-1} mod L for $1 \leq j \leq r$

Aim of this paper: design efficient algorithms for computing

- the *polynomial solutions* of L
- the *p-curvature* $\mathbf{A}_p$ of L

- Efficiency = complexity estimates with a low exponent in p

Basics on differential equations in characteristic p

- Main differences between characteristic zero and p
  1. (Honda 1981) solutions are simpler in characteristic p

$$\mathcal{S}_L(\mathbb{F}_p[x]) = \mathcal{S}_L(\mathbb{F}_p(x)) = \mathcal{S}_L(\mathbb{F}_p[[x]]) =: \mathcal{S}_L$$

  2. Cauchy's theorem does not hold: the dimension of $\mathcal{S}_L$ over the field of constants $C = \mathbb{F}_p(x^p)$ is generally $< r = \mathrm{ord}(L)$

Example: $y' = y$ has no solution in $\mathbb{F}_p[[x]]$

- Connection between solutions and p-curvature

Theorem. (Katz & Cartier 1970) $\mathrm{rank}(\mathbf{A}_p) = r - \dim_C(\mathcal{S}_L)$

$\longrightarrow$ p-curvature measures to what extent $\dim_C(\mathcal{S}_L)$ is close to r

Motivation

- van der Put 1995: p-curvature publicised in computer algebra, as a tool for factoring operators in $\mathbb{F}_p(x)\langle\partial\rangle$
- Cluzeau 2003: first complexity analysis and implementation of van der Put's algorithms; extension to systems
- Cluzeau, van Hoeij 2004: polynomial solutions mod p and p-curvature used as filters in modular algorithms for $\mathbb{Q}(x)\langle\partial\rangle$

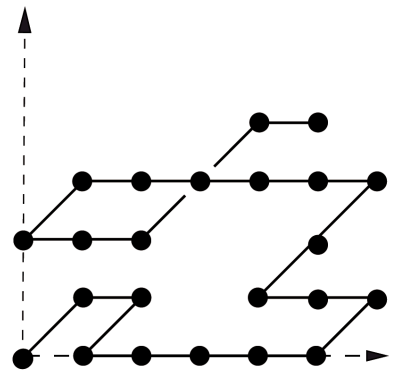
- Concrete applications:
  - enumerative combinatorics (classification of lattice walks)
  - statistical physics (square lattice Ising model)

Combinatorial application: Gessel's conjecture

- Gessel walks: walks in $\mathbb{N}^2$ using only steps in $\mathcal{S} = \{\nearrow, \swarrow, \leftarrow, \rightarrow\}$
- $g(i, j, n)$ = number of walks from $(0, 0)$ to (i, j) with n steps in $\mathcal{S}$

Question: Nature of the generating function

$$G(x, y, t) = \sum_{i,j,n=0}^{\infty} g(i, j, n)\, x^i y^j t^n \in \mathbb{Q}[[x, y, t]]$$



Theorem. (B. & Kauers 2008) $G(x, y, t)$ is an algebraic function.[†]

$\longrightarrow$ Effective, computer-driven discovery and proof

$\longrightarrow$ Key step in discovery: p-curvature computation of two 11th order (guessed) differential operators for $G(x, 0, t)$, and $G(0, y, t)$

[†] The MinPoly $P(x, y, t, G(x, y, t)) = 0$ has $> 10^{11}$ monomials; $\approx$30Gb (!)

Previous work

① p-curvature: size $\mathcal{O}(p)$, but complexity $\mathcal{O}(p^2)$

▶ Main difficulty: non-commutativity of $\mathbb{F}_p(x)\langle\partial\rangle$ prevents from using binary powering techniques for $\mathbf{A}_p$ via $\partial^p \bmod L$

• Katz 1982: first algorithm, based on the matrix recurrence

$$\mathbf{A}_1 = \mathbf{A}, \quad \mathbf{A}_{k+1} = \mathbf{A}'_k + \mathbf{A} \cdot \mathbf{A}_k,$$

where $\mathbf{A} \in \mathscr{M}_r(\mathbb{F}_p(x))$ is the companion matrix associated to L

• van der Put, Cluzeau: variants, all of complexity $\mathcal{O}(p^2)$

② Polynomial solutions mod p: few algorithms in the literature

• Cluzeau 2003: general algorithm of complexity $\mathcal{O}(p^3)$; different $\mathcal{O}(p^2)$ algorithm in the special case $\mathbf{A}_p = 0$

New results

1. on the space $\mathcal{S}_L$ of polynomial solutions

(1.a) degree bound linear in p for all elements in a basis of $\mathcal{S}_L$

(1.b) testing if $\mathcal{S}_L = 0$ in time $\tilde{\mathcal{O}}(\sqrt{p})$

(1.c) computing a whole basis of $\mathcal{S}_L$ in time $\tilde{\mathcal{O}}(p)$

2. on computing the p-curvature $\mathbf{A}_p$

(2.a) for first order operators in time $\mathcal{O}(\log(p))$

(2.b) for certain second order operators in time $\tilde{\mathcal{O}}(p)$

(2.c) for any operator in time $\tilde{\mathcal{O}}(p^{1.8})$

(2.d) deciding nilpotency of $\mathbf{A}_p$ for second order operators in $\tilde{\mathcal{O}}(\sqrt{p})$

p-curvature of arbitrary operators

Theorem. The p-curvature of any L in $\mathbb{F}_p[x]\langle\partial\rangle$ can be computed in subquadratic time $\tilde{\mathcal{O}}(p^{1+\frac{\omega}{3}}) \subset \mathcal{O}(p^{1.8})$.

If $\mathbf{A} = \text{CompanionMatrix}(L)$ and $\Lambda = \partial + \mathbf{A}$, then $\mathbf{A}_p = \Lambda^{p-1}(\mathbf{A})$.

---

① Compute $\Gamma = \Lambda^k$ by binary powering.

Basic operation: product in bidegree (k, k) in $\mathbb{F}_p(x)\langle\partial\rangle$.

Cost: $\mathcal{O}(k^\omega)$ (B., Chyzak & Le Roux'08)

② Compute $\mathbf{A}_{(1)} = \mathbf{A}$, $\mathbf{A}_{(i)} = \Gamma\mathbf{A}_{(i-1)}$, $i = 2, \ldots, \ell = (p-1)/k$.

Basic operation: $L(f)$ with $\text{bideg}_{(x,\partial)}(L) = (k, k)$ and $\deg(f) \leq ik$.

Cost: $\tilde{\mathcal{O}}(\ell^{\omega-1}k^2)$ (see next slide)

③ Return $\mathbf{A}_p = \mathbf{A}_{(\ell)}$.

Total cost: $\tilde{\mathcal{O}}(p^{1+\frac{\omega}{3}})$ obtained for $k \approx p^{2/3}$.

Fast evaluation of differential operators

Theorem. Given $L \in \mathbb{F}_p[x]\langle\partial\rangle$ of bidegree (k,k) and $f \in \mathbb{F}_p[x]$ of degree ik, $(i \le s := \sqrt{k})$, one can compute Lf in time $\tilde{\mathcal{O}}(i^{\omega-2}k^2)$.

Algo [baby steps / giant steps strategy inspired by Brent-Kung'78]

① (baby steps) Compute $f_0 = f,\ f_1 = \partial f,\ \ldots,\ f_{s-1} = \partial^{s-1} f$

② (rewriting) Cut L into s slices of bidegree (k,s) in (x,∂):

$$L = L_0 + \partial^s L_1 + \cdots + \partial^{(s-1)s} L_{s-1}$$

③ (recombination) Deduce $L_0 f, \ldots, L_{s-1} f$ at once by a product of polynomial matrices of sizes $(s,s) \times (s,i)$ and degree k

④ (giant steps) Compute and return $Lf = \sum_{0 \le j < s} \partial^{js} L_j f$

Cost: $\tilde{\mathcal{O}}(ik^{3/2})$ for ① and ④; $\tilde{\mathcal{O}}(k^2)$ for ② and $\tilde{\mathcal{O}}(i^{\omega-2}k^2)$ for ③

Computing polynomial solutions (I)

- $\mathcal{S}_L =$ the C-vector space of polynomial solutions of $Lf = 0$
- $\mathcal{G} =$ the $\mathbb{F}_p$-vector space $\mathcal{S}_L \cap \mathbb{F}_p[x]_{<pd}$ where $d = \max(\deg(\ell_i))$

Theorem. $\mathcal{S}_L$ admits a C-basis included in $\mathcal{G}$.

Algorithm for computing $\mathcal{S}_L$:

---

① Decide if $\mathcal{S}_L = 0$ ($\Leftrightarrow$ decide if $\mathcal{G} = 0$). If so, stop.
② If not, compute a $\mathbb{F}_p$-basis $(f_1, \ldots, f_k)$ of $\mathcal{G}$.
③ $(f_1, \ldots, f_k)$ generates $\mathcal{S}_L$ over C. Extract a basis.

Cost: $\tilde{\mathcal{O}}(\sqrt{p})$ for ① and $\tilde{\mathcal{O}}(p)$ for ② and ③

Computing polynomial solutions (II)

Pb: Compute an $\mathbb{F}_p$-basis of sols $f = \sum_{i=0}^{pd-1} c_i x^i \in \mathbb{F}_p[x]$ of $Lf = 0$.

- Band-diagonal linear system (S1), width $\mathcal{O}(1) \rightarrow$ gradual solving
- Technical difficulty: some rightmost band elements can be zero!

Algorithm [generalization of (ABP1995) & (BCluzeauSalvy2005)]

① From (S1), deduce an equivalent system (S2) of size $\mathcal{O}(1)$

Basic operation: *matrix factorial* $C(p-1)\cdots C(r)$

Cost: $\tilde{\mathcal{O}}(\sqrt{p})$ (Chudnovsky^2 1989)

② From a basis of (S2), deduce a basis of (S1)

Basic operation: forward substitution

Cost: $\mathcal{O}(p)$

Example

$$L = (5x^2+4)\partial^2 + (4x^2+6x+5)\partial + 2x + 2 \in \mathbb{F}_7[x]\langle\partial\rangle$$

Polynomial solution $f = \sum_{i=0}^{7\cdot 2-1} c_i x^i$ in $\mathbb{F}_7[x]$ s.t. $Lf = 0$:

$$\text{(S1)} \left\{ \begin{array}{c} 2c_0+5c_1+c_2=0, \\ 2c_0+c_1+3c_2+3c_3=0, \\ 6c_1+3c_2+c_3+6c_4=0, \\ 3c_2+c_3+6c_4+3c_5=0, \\ 2c_4+4c_5+c_6=0, \\ 4c_4+6c_5+2c_6=0, \\ c_5+6c_6=0, \\ 5c_6+2c_7+5c_8+c_9=0, \\ 2c_7+c_8+3c_9+3c_{10}=0, \\ 6c_8+3c_9+c_{10}+6c_{11}=0, \\ 3c_9+c_{10}+6c_{11}+3c_{12}=0, \\ 2c_{11}+4c_{12}+c_{13}=0, \\ 4c_{11}+6c_{12}+2c_{13}=0, \\ c_{12}+6c_{13}=0, \\ 5c_{13}=0. \end{array} \right. \Leftrightarrow \begin{bmatrix} 2 & 5 & 1 & & & & & & & & & & & \\ 2 & 1 & 3 & 3 & & & & & & & & & & \\ 0 & 6 & 3 & 1 & 6 & & & & & & & & & \\ & 0 & 3 & 1 & 6 & 3 & & & & & & & & \\ & & 0 & 0 & 2 & 4 & 1 & & & & & & & \\ & & & 0 & 4 & 6 & 2 & 0 & & & & & & \\ & & & & 0 & 1 & 6 & 0 & 0 & & & & & \\ & & & & & 0 & 5 & 2 & 5 & 1 & & & & \\ & & & & & & 0 & 2 & 1 & 3 & 3 & & & \\ & & & & & & & 0 & 6 & 3 & 1 & 6 & & \\ & & & & & & & & 0 & 3 & 1 & 6 & 3 & \\ & & & & & & & & & 0 & 0 & 2 & 4 & 1 \\ & & & & & & & & & & 0 & 4 & 6 & 2 \\ & & & & & & & & & & & 0 & 1 & 6 \\ & & & & & & & & & & & & 0 & 5 \end{bmatrix} \times \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \\ c_8 \\ c_9 \\ c_{10} \\ c_{11} \\ c_{12} \\ c_{13} \end{bmatrix} = 0$$

➥ Rewriting the equations in red using the unknowns in blue yields

$$c_4 = 5c_1,\ c_5 = 2c_1,\ c_6 = 3c_1,\ c_{11} = 5c_8 + 5c_1,\ c_{12} = 2c_8,\ c_{13} = 3c_8 + 4c_1$$

➥ Plugging them back into the red equations gives

$$\text{(S2)}: \qquad 3c_1 = 0, 6c_1 = 0, 3c_8 = 0, c_8 + 6c_1 = 0, 6c_8 + 3c_1 = 0$$

➥ $\dim(\text{S1}) = \dim(\text{S2}) = 2$ and Basis(S1) is obtained by subst. from Basis(S2) $= \{(c_0, c_1, c_7, c_8) = (1,0,0,0), (c_0, c_1, c_7, c_8) = (0,0,1,0)\}$

p-curvature of 2nd order operators

- p-curvature $\mathbf{A}_p(L) = (a_{i,j})$ satisfies a *matrix differential equation*:

$$\mathbf{A}_p' = \mathbf{A}_p \cdot \mathbf{A} - \mathbf{A} \cdot \mathbf{A}_p \qquad \text{(Cluzeau, 2003)}$$

where $\mathbf{A}$ is the companion matrix of L.

- If $L = v\partial^2 + w\partial + u \in \mathbb{F}_p[x]\langle\partial\rangle$, this is equivalent to the system

$$\begin{aligned} v^3 a_{2,1}''' + A a_{2,1}' + B a_{2,1} &= 0, \qquad (\dagger) \\ v^2 a_{1,2} + R a_{2,1}'' + S a_{2,1}' + T a_{2,1} &= 0, \\ v(a_{1,1} - a_{2,2}) + v a_{2,1}' - w a_{2,1} &= 0, \\ a_{1,1}' + a_{2,2}' &= 0, \end{aligned}$$

where $A, B, R, S, T \in \mathbb{F}_p[x]$ are explicit in terms of v, w, u.

- computing $\mathbf{A}_p$ amounts to solving Eq $(\dagger)$, in time $\tilde{\mathcal{O}}(p)$.

Conclusion, open questions

So far:

- algorithm of quasi-optimal complexity for solving $Lf = 0$.

Still open:

- compute the p-curvature in quasi-linear time (at least for second order operators!)
- decide in sublinear time, e.g. in $\tilde{\mathcal{O}}(\sqrt{p})$, if $\mathbf{A}_p(L)$ is nilpotent when $\mathrm{ord}(L) > 2$.
- are the problems of computing p-curvature and computing polynomial solutions computationally equivalent?