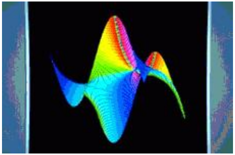
Common Multiples of Linear Differential and Difference Operators

Frédéric Chyzak
(joint work in progress with A. Bostan, Z. Li & B. Salvy)

Algorithms Project, INRIA



November 21, 2007

Motivation: Iterated LCLMs are bad!

$A = \mathbb{Q}\langle x, \partial; \partial x = x\partial + 1\rangle$ (differential op., polynomial coeffs.)
LCLM $=$ common left multiple with least order, with no content
$L = \mathrm{LCLM}(L_1, \ldots, L_k) = C_i L_i$ for given $L_i \in A$

$\deg_\partial L_i = r_i, \quad \deg_x L_i = d_i \quad \longrightarrow \quad \deg_\partial L \le R, \quad \deg_x L \le B$

k	R	B	generic	Fuchsian
1	3	9		
2	6	72	72 / 1.2s	63 / <1s
3	9	189	189 / 42s	162 / 23s
4	12	360	360 / 519s	306 / 267s

$\mathrm{LCLM}\big(\mathrm{LCLM}(L_1, L_2), \mathrm{LCLM}(L_3, L_4)\big)$ takes 1120s / 796s,
$\mathrm{LCLM}(\mathrm{LCLM}(L_1, L_2), L_3)$ takes 61s / 53s,
$\mathrm{LCLM}(\mathrm{LCLM}(\mathrm{LCLM}(L_1, L_2), L_3), L_4)$ takes 978s / 677s.
(Maple 11's `DEtools[LCLM]` by Van Hoeij; $-100 <$ coeffs. < 100.)

Related Works for LCLMs in Specific Classes

LCLMs of first-order operators in Van Hoeij's `DEtools[DFactor]` (1997).

LCLMs of two operators by *A subresultant theory for linear ordinary differential polynomials*, Z. Li (1998).

LCLMs of the form $\mathrm{LCLM}\big(\{L(\omega x, \omega^{-1}\partial); \omega^k = 1\}\big)$ appear for the extraction of k-sections in multisummable D-finite series in *Remarques algorithmiques liées au rang d'un opérateur différentiel linéaire*, Barkatou, C., Loday-Richaud (2003).

LCLMs of telescopers are used for the summation of rational functions, after a partial fraction decomposition, in *A direct algorithm to construct the minimal Z-pairs for rational functions*, H. Q. Le (2003).

Contributions

- Analysis of worst-case arithmetic complexity of several existing algorithms for computing LCLMs.
- A tight bound $B \approx k^2 rd$ on the degree in x of the LCLM.
- A first algorithm, asymptotically better for large r.
- A bound $B' \approx 2kr$ on the total degree in (x, ∂) in which common left multiples exist.
- A faster algorithm which computes the LCLM as the GCRD of two common multiples of small size.

Inspired by our work on algebraic series (ISSAC'07):
by allowing orders twice as large as the LCLM, $B' \approx 2R$,
the total size is much reduced, $B'^2/2 \approx 2k^2r^2 \ll RB \approx k^3r^2d$.

Poole's Method ($k = 2$)

1. Set $R = r_1 + r_2$ and $D = \max\{d_1, d_2\}$.
2. Compute $M_{i,j,1} = x^i \partial^j L_1$ for $0 \leq i \leq (R+2)D$, $0 \leq j \leq r_2$.
3. Compute $M_{i,j,2} = x^i \partial^j L_2$ for $0 \leq i \leq (R+2)D$, $0 \leq j \leq r_1$.
4. Find a suitable linear combination of the $M_{i,j,k}$ over $\mathbb{Q}$.

$$\#\text{unkowns} = (R+2)\big((R+2)D+1\big),$$
$$\#\text{equations} = (R+1)\big((R+2)D+D+1\big),$$
$$\dim \text{kernel} \geq D+1.$$

Complexity at least $O(R^{2\omega} D^{\omega})$ for $k = 2$.
Iterative LCLMs in $O(kR^{3\omega} D^{\omega})$; by balancing inputs in $O(R^{3\omega} D^{\omega})$.

Ore's Method: Extended GCRD Algorithm ($k = 2$)

1. Set $R_0 = L_1$, $R_1 = L_2$, $U_0 = 1$, $V_0 = 0$, $U_1 = 0$, $V_1 = 1$.
2. Perform right Euclidean divisions $R_{i-1} = Q_i R_i + R_{i+1}$ and compute $U_{i+1} = Q_i U_i - U_{i-1}$, $V_{i+1} = Q_i V_i - V_{i-1}$, to ensure $R_i = U_i L_1 + V_i L_2$.
3. Let m be the index of the last non-zero remainder R_m (a GCRD). Output $U_{m+1} L_1 = -V_{m+1} L_2$.

Generically, $\deg_\partial Q_i = 1$, $\deg_x Q_i = \deg_x R_i = (i+1)D$, so the ith Euclidean division costs $\tilde{O}\big((r-i)iD\big)$, if $r = \max\{r_1, r_2\}$.

Total complexity is $\tilde{O}(R^3 D)$ for $k = 2$.
Iterative LCLMs in $\tilde{O}(kR^4 D)$; by balancing inputs in $\tilde{O}(R^4 D)$.

Van Hoeij's Method Revisited ($k \geq 2$)

1. Compute the $v_{i,j} = \text{rem}(\partial^i, L_j)$ for $0 \leq i \leq R = kr$, $1 \leq j \leq k$, keeping $\text{lc}(L_j)^i$ as denominators, by the formula
$$v_{i,j} = \text{rem}(\partial v_{i-1,j}, L_j).$$
2. Write the numerator of each $v_{i,j}$ as a row vector $n_{i,j}$ in $\mathbb{Q}[x]^{r_j}$ and determine the rank ρ of $N = (n_{i,j})$ by Storjohann and Villard's 2005 algorithm.
3. Extract (randomly) an invertible submatrix of N from the first ρ rows, and prolong it consistently with the next row to get a $(\rho + 1) \times \rho$ matrix M.
4. Determine the (rank 1) polynomial kernel of M by Storjohann and Villard's 2005 algorithm.
5. Simplify the (single) basis element so as to take the denominators into account.

Complexity: $\tilde{O}(R^3 D)$ for filling in V; $\tilde{O}(R^{\omega+1} D)$ for computing the kernel; $\tilde{O}(R^3 D)$ for simplifying.

Our Linear-Algebraic Approach to Computing LCLMs

Setting: $\partial a = \sigma(a)\partial + \delta(a)$ where σ and δ do not increase the degrees in x and can be computed in linearly-many operations.

$$S_N(P) = \begin{pmatrix} \partial^{N-\deg_\partial P} P \\ \vdots \\ \partial P \\ P \end{pmatrix}, \quad M_N = \begin{pmatrix} S_N(L_1) & & \\ & \ddots & \\ & & S_N(L_k) \\ -S_{N+1}(1) & \dots & -S_{N+1}(1) \end{pmatrix}.$$

Left kernel of M_N $\quad\longleftrightarrow\quad$ Common multiples and their cofactors.

Fast polynomial-kernel algorithm for solving (Storjohann–Villard).
Tight degree bound for the analysis (from Hadamard's bound).

Degree Bound for Common Multiples of Minimal Order

L_i of order r_i and degree d_i, $R = r_1 + \cdots + r_k$, $D = \max_{1 \le i \le k} d_i$.
$L = \mathrm{LCLM}(L_1, \ldots, L_k)$ or order $\ell \le R$.

M_N has size $m_N \times n_N$ and rank ρ_N given by:

$$m_N = (k+1)(N+1) - R, \qquad n_N = k(N+1),$$
$$m_N - n_N = N + 1 - R \le 1,$$
$$(k+1)(N+1) - R = (N - \ell + 1) + \rho_N.$$

All coefficients in M_N have degree $\le D$.

Hadamard's bound is $B = (k\ell + k - R)D$, where
$\ell = \rho_R + R - k(R+1)$.

Algorithm for Common Multiples of Minimal Order

1. Compute M_R and determine its rank ρ by S. & V.'s algorithm.
2. Set $\ell = \rho + R - k(R+1)$ and extract M_ℓ from M_R by suppressing rows and columns.
3. Compute the kernel of M_ℓ by S. & V.'s algorithm, that is, find one polynomial solution $(C_1, \dots, C_k, L)$, and return it.

Complexity: $O(k^2 R^2 \rho^{\omega-2} D) = O(k^\omega R^\omega D)$

Towards Common Multiples of Small Total Degree

In the spirit of Poole's method:

$$S'_N(P) = \left(x^i \partial^j P\right)_{i+j=0}^{N-\deg_{x,\partial} P}, \quad M'_N = \begin{pmatrix} S'_N(L_1) & & \\ & \ddots & \\ & & S'_N(L_k) \\ -S'_{N+1}(1) & \dots & -S'_{N+1}(1) \end{pmatrix}.$$

For $\deg_{x,\partial} L_i \leq \delta$, a left kernel is ensured by the bound

$$N \geq B' = \left\lceil k\delta + \frac{\left(4k(k-1)\delta^2 + 1\right)^{1/2} - 3}{2} \right\rceil \approx 2k\delta.$$

Matrix of size $O(k^3\delta^2) \rightarrow$ linear algebra in $O(k^{3\omega}\delta^{2\omega})$?

Final Algorithm for Common Multiples of Minimal Order

Series of size $O(B'^2) \subset O(k^2\delta^2)$ instead of operators of size $O(k^3 r^2 d) \subset O(k^3\delta^3)$!

1. Compute truncated series solutions of the L_i at order ${B'}^2 + B'$ — $O(k{B'}^2) \subset O(k^3\delta^2)$
2. Take a random linear combination of them — (same)
3. Derive its first B' derivatives — $O(k^3\delta^3)$
4. Compute a Hermite–Padé approximant — $O\big({B'}^\omega M(B')\big) \subset \tilde{O}(k^{\omega+1}\delta^{\omega+1})$
5. Take any two CLMs and compute their GCRD — $O\big(r^\omega M(d)\big) \subset \tilde{O}(\delta^{\omega+1})$

Conclusions

Algorithm	Complexity	Best for
Poole's	$k^{3\omega} r^{3\omega} D$	
Ore's	$k^4 r^4 D$	
Van Hoeij's	$k^{\omega+1} r^{\omega+1} D$	large (k, D), fixed r
Ours by S.V.	$k^{2\omega} r^{\omega} D$	large r, fixed (k, D)
Ours by P.H.	$k^{\omega+1} \delta^{\omega+1}$	large δ when $\delta \approx r \approx D$

$$r, D \leq \delta \leq r + D$$

When $k = 2$, Li's subresultants give LCLM in $O(r^\omega D)$.

For comparison's sake: product in $O(\delta^\omega)$.

Eigenrings in same complexity?