Factorization and hypergeometric solutions of linear recurrence systems.
Manuel Bronstein Conference, July 13, 2006

Factorization. Three cases:

1. Factor in $K[x]$ where K is a field.
2. Factor in $\mathbb{C}(x)[\partial]$ where $\partial = \mathrm{d}/\mathrm{d}x$
3. Factor in $\mathbb{C}(x)[\tau]$ where τ is the shift operator.

The goal is an algorithm for case 3.

1. $K[x]$: For illustration during the talk.
2. $\mathbb{C}(x)[\partial]$: Discuss Manuel's ISSAC'1994 paper.
3. $\mathbb{C}(x)[\tau]$: Was to become joint work. Ingredients:

   - Manuel's '94 paper also works for $\mathbb{C}(x)[\tau]$.
   - To make this as efficient as possible we need a direct algorithm for computing hypergeometric solutions of systems. I will discuss Manuel's (unpublished) ideas for this algorithm.

If $f \in K[x]$ and you can factor $f = f_1 f_2$ then solving $f(x) = 0$ reduces to solving equations f_1 and f_2 of lower degree. So a factorization makes solving easier.

If $L \in \mathbb{C}(x)[\partial]$ then $L = a_n\partial^n + \ldots + a_0\partial^0$ with $a_i \in \mathbb{C}(x)$ and L acts as follows:

$$L(y) = a_n y^{(n)} + \cdots + a_1 y' + a_0 y$$

Corresponding differential equation $L(y) = 0$.

If $L = L_1 L_2$ then solutions of $L_2(y) = 0$ are also solutions of $L(y) = 0$. So having a right-hand factor of L makes it easier to find solutions of $L(y) = 0$.

An operator $L = a_n\tau^n + \ldots + a_0\tau^0 \in \mathbb{C}(x)[\tau]$ acts as follows:
If $u = u(x)$ is a function then

$$L(u) = a_n u(x+i) + \cdots + a_1 u(x+1) + a_0 u(x)$$

If

$$L(u) = 0 \quad \text{and} \quad \frac{u(x+1)}{u(x)} \in \mathbb{C}(x)$$

then u is called a **hypergeometric solution** and corresponds to a **right-hand factor** $\tau - r$ of L where

$$r = \frac{u(x+1)}{u(x)}.$$

A hypergeometric solution u of $L = a_n\tau^n + \cdots + a_0\tau^0$ can always be written as

$$u(x) = c^x P(x) \prod_i \Gamma(x - \alpha_i)^{e_i}$$

for some

$$c \in \mathbb{C}^*, \quad P(x) \in \mathbb{C}[x], \quad \alpha_i \in \mathbb{C}, \quad e_i \in \mathbb{Z}.$$

Write

$$A = \prod_{e_i>0} (x - \alpha_i)^{e_i} \text{ and } B = \prod_{e_i<0} (x - \alpha_i)^{-e_i}$$

so that

$$u(x) = c^x P(x) \mathrm{Sol}(\tau - \frac{A}{B}).$$

Petkovšek (1992) gave a criterium for $A, B \in \mathbb{C}[x]$, namely:

$$A \text{ divides } a_0(x)$$
$$B \text{ divides } a_n(x - n + 1)$$

This leaves only a finite (but exponential) number of potential A, B in the hypergeometric solutions:

$$\begin{aligned} u(x) &= c^x P(x) \mathrm{Sol}(\tau - \frac{A}{B}) \\ &= c^x P(x) \prod_i \Gamma(x - \alpha_i)^{e_i} \end{aligned}$$

Petkovšek's algorithm (1992):

For all possible combinations of:

- a monic factor $A \in \mathbb{C}[x]$ of a_0,
- a monic factor B of $a_n(x-n+1)$,
- and c in some finite list

compute a recurrence $L_{A,B,c}$ that has $P(x)$ as a solution, and solve it to find hypergeometric solutions:

$$c^x P(x) \mathrm{Sol}(\tau - \frac{A}{B})$$

Another algorithm was given by Cluzeau and v.H.

Computing hypergeometric solutions is equivalent to computing first order factors. So:

- We can compute factors of order 1.
- How to use this to compute factors of higher order?

Beke (1894) has shown that you can reduce finding d'th order factors of an n'th order operator to computing first order factors of operators of order $\binom{n}{d}$.

Manuel's ISSAC'1994 paper gives significant practical improvements to this process.

L has order n. To compute: factors of order d. In Beke's approach, this means computing first order factors of operators of order $\binom{n}{d}$.

To simplify notations, we take

$$n = 4,\ d = 2 \text{ so } \binom{n}{d} = 6.$$

However, what follows also works for general n, d.

Reducing computing higher order factors to computing first order factors (Beke 1894).

If $L = \partial^4 + a_3\partial^3 + a_2\partial^2 + a_1\partial + a_0$ in $\mathbb{C}(x)[\partial]$ then one can compute a 6'th order differential operator L_a such that $\partial + a$ is a right-hand factor of L_a for every right-hand factor $\partial^2 + a\partial + b$ of L.

Potential b's are obtained likewise (compute first order factors of some operator L_b). Then by trying combinations of the potential a's and potential b's, one finds all second order right-hand factors of L.

Manuel significantly improved this (next slide).

- Instead of solving **two operators** of order 6 (L_a and L_b) we need to solve only **one system** of order 6. So the problem of how to combine the data from L_a with that of L_b disappears.
- This 6'th order system has much smaller coefficients in $\mathbb{C}(x)$ than the operators L_a and L_b.

This makes the algorithm much more elegant. Moreover, the smaller coefficient sizes in Manuel's approach can lead to significantly improved performance (recall the combinatorial search in Petkovšek's algorithm).

To take advantage of the smaller coefficient sizes, one must find hypergeometric solutions of a **system** of order 6 instead of **operators** of order 6.

For this reason, Manuel and I had planned to write a program for computing hypergeometric solutions of a system directly (without reducing the system to an operator because that increases coefficient size, which would eliminate the efficiency advantage).

Remaining slides:
Explain Manuel's ISSAC'1994 paper and his ideas for hypergeometric solutions of systems.

Lets use the case $K[x]$ to illustrate Manuel's ISSAC'1994 paper and do the case $\mathbb{C}(x)[\tau]$ later.

We take $n = 4$ and $d = 2$ (the general case works in the same way).

Suppose $f = x^4 + a_3x^3 + a_2x^2 + a_1x + a_0 \in K[x]$. Degree $n = 4$. Suppose we have an algorithm for computing factors of degree 1 but not for degree 2.

f has four roots $\alpha_1, \ldots, \alpha_4$. If $x^2 + ax + b$ is a factor of f then

$$-a = \alpha_i + \alpha_j \quad \text{and} \quad b = \alpha_i\alpha_j$$

for some i, j. So a is a root of

$$F_a = \prod_{i<j} x + \alpha_i + \alpha_j$$

and b is a root of

$$F_b = \prod_{i<j} x - \alpha_i\alpha_j.$$

Beke (1894) described factoring in $\mathbb{C}(x)[\partial]$ but if we applied the same idea to $K[x]$ then we'd compute quadratic factors of $x^2 + ax + b$ of f as follows:

1. Compute $F_a, F_b \in K[x]$ (degrees are 6).
2. Compute linear factors of F_a to find candidate a's.
3. Same for F_b to get the candidate b's.
4. For each combination of some candidate a and b, try if $x^2 + ax + b$ divides f.

Manuel's improvements are as follows:

- Only one degree 6 problem instead of two.
- Smaller coefficient sizes.
- Combinatorial problem in item 4 above disappears.

Illustration of Manuel's ISSAC'94 paper to $K[x]$. Let $f \in K[x]$ have degree 4. Now $M := K[x]/(f)$ is a $K[x]$-module of K-dimension 4 and its second exterior power $\wedge^2 M$ is a $K[x]$-module of K-dimension 6.

If g is a factor of f of degree 2 then $N := Kxg + Kg$ is a submodule of M. Write $w := xg \wedge g$. Then $\wedge^2 N = Kw$ is a submodule of $\wedge^2 M$.
The action of x on $\wedge^2 M$ is given by a 6 by 6 matrix, and w must be an eigenvector of this matrix.

Thus, finding w and hence g reduces to an eigenvector computation of a system of dimension 6.

If $f = x^4 - a_3x^3 - a_2x^2 - a_1x - a_0$ then the action of multiplying by x on a basis of $\wedge^2 M$ is:

$x^0 \wedge x^1 \to x^1 \wedge x^2$
$x^0 \wedge x^2 \to x^1 \wedge x^3$
$x^0 \wedge x^3 \to x^1 \wedge x^4 = x^1 \wedge (a_3x^3 + a_2x^2 + a_0x^0)$
$x^1 \wedge x^2 \to x^2 \wedge x^3$
$x^1 \wedge x^3 \to x^2 \wedge x^4 = x^2 \wedge (a_3x^3 + a_1x^1 + a_0x^0)$
$x^2 \wedge x^3 \to x^3 \wedge x^4 = x^3 \wedge (a_2x^2 + a_1x^1 + a_0x^0)$

This action can be described by the following matrix.

$$\begin{pmatrix} 0 & 0 & -a_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -a_0 \\ 1 & 0 & a_2 & 0 & -a_1 & 0 \\ 0 & 1 & a_3 & 0 & 0 & -a_1 \\ 0 & 0 & 0 & 1 & a_3 & -a_2 \end{pmatrix}$$

Now a 1-dimensional submodule of $\wedge^2 M$ corresponds to an eigenvector of the (transpose) of this matrix.

For $\mathbb{C}(x)[\tau]$ with $L = \tau^4 - a_3\tau^3 - a_2\tau^2 - a_1\tau - a_0$ one finds the same matrix as on the previous slide.

Replacing $K[x]$ modules by $\mathbb{C}(x)[\partial]$-modules, and computing the action of ∂ on $\wedge^2 M$ one finds the same matrix in Manuel's ISSAC'1994 paper. This matrix looks slightly different because the action of ∂ satisfies the Leibniz rule

$$\partial(a \wedge b) = \partial(a) \wedge b \ + \ a \wedge \partial(b)$$

which introduces a few more terms.

To find a factor of order 2 of an operator L of order 4, we must compute

- If $L \in K[x]$: an eigenvector with eigenvalue in K.
- If $L \in \mathbb{C}(x)[\partial]$: an exponential solution.
- If $L \in \mathbb{C}(x)[\tau]$: a hypergeometric solution.

The advantage that the entries in the matrix are small, in the $K[x]$-case this advantage disappears when we compute the characteristic polynomial.

But for $\mathbb{C}(x)[\partial]$ or $\mathbb{C}(x)[\tau]$ we can benefit from this advantage if we do not compute the analogue of the characteristic polynomial (a cyclic vector operator).

So if $L = \tau^4 - a_3\tau^3 - a_2\tau^2 - a_1\tau - a_0$ we need to compute a hypergeometric solution of

$$\tau(Y) = \begin{pmatrix} 0 & 0 & -a_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -a_0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -a_0 \\ 1 & 0 & a_2 & 0 & -a_1 & 0 \\ 0 & 1 & a_3 & 0 & 0 & -a_1 \\ 0 & 0 & 0 & 1 & a_3 & -a_2 \end{pmatrix} Y$$

(again, τ is the shift operator).

So the system to solve looks like

$$\begin{pmatrix} y_1(x+1) \\ \vdots \\ y_6(x+1) \end{pmatrix} = M \begin{pmatrix} y_1(x) \\ \vdots \\ y_6(x) \end{pmatrix}$$

where M is a 6 by 6 matrix over $\mathbb{C}(x)$. The solution we search for is hypergeometric, meaning that it can be written as

$$\begin{pmatrix} y_1(x) \\ \vdots \\ y_6(x) \end{pmatrix} = c^x \begin{pmatrix} P_1(x) \\ \vdots \\ P_6(x) \end{pmatrix} \mathrm{Sol}(\tau - \frac{A}{B})$$

where $A, B, P_1, \ldots, P_6 \in \mathbb{C}[x]$ and $c \in \mathbb{C}^*$.

The goal: Find a solution

$$Y = c^x \begin{pmatrix} P_1(x) \\ \vdots \\ P_6(x) \end{pmatrix} \mathrm{Sol}(\tau - \frac{A}{B})$$

of the system $\tau(Y) = MY$. Once you have A, B and c, then you can compute $P_1, \ldots, P_6$ with software that Manuel had already implemented. The question is: How to find A, B? Manuel had found the answer:

$$Y = c^x \begin{pmatrix} P_1(x) \\ \vdots \\ P_6(x) \end{pmatrix} \mathrm{Sol}(\tau - \frac{A}{B})$$

Manuel's idea for finding A, B:

We may assume that the gcd of $P_1, \ldots, P_6$ is 1. Now look at

$$\tau(Y) = \frac{A}{B} c^{x+1} \begin{pmatrix} P_1(x+1) \\ \vdots \\ P_6(x+1) \end{pmatrix} \mathrm{Sol}(\tau - \frac{A}{B})$$

The denominator B in $\tau(Y)$ does not appear in Y, and since $\tau(Y) = MY$ it follows that the denominator B must divide the denominator of M.

A similar argument shows that A divides the denominator of M^{-1}.

So Manuel's idea shows how you can generalize Petkovšek's algorithm to systems: B has to divide denominator(M), and A has to divide denominator(M^{-1}).

We started from an operator

$$L = a_4\partial^4 + \cdots + a_1\partial + a_0$$

for which we wanted to compute the factors of order 2. One would expect that this should be harder than computing factors of order 1.

The operator L lead (through the process from Manuel's ISSAC'1994 paper) to the matrix M. Now the denominators of M resp. M^{-1} are a_4 resp. a_0.

So A resp. B is a monic factor of a_0 resp. a_4, the same bound as in Petkovšek's algorithm. That's a very nice and surprising result because it indicates that computing second order factors is not intrinsically harder than computing first order factors.