Computer Algebra Methods for Orthogonal Polynomials

Prof. Dr. Wolfram Koepf
University of Kassel
koepf@mathematik.uni-kassel.de
http://www.mathematik.uni-kassel.de/~koepf

International Conference on Difference Equations, Special Functions and Applications
Munich, July 25-30, 2005
Online Demonstrations with Computer Algebra

- I will use the computer algebra system *Maple* to demonstrate and program the algorithms presented.

- Of course, we could also easily use any other general purpose system like *Mathematica*, MuPAD or Reduce.

- The following algorithms are most prominently used (internally): linear algebra techniques, multivariate polynomial factorization and the solution of nonlinear equations, e.g. by Gröbner basis techniques.
An Appetizer

• As an appetizer we consider the conversion between a recurrence equation and a difference equation.
• In this talk a difference equation is an equation involving the forward difference operator
  \[ \Delta f(x) = f(x + 1) - f(x) \, . \]
• Question: How can one convert a recurrence equation
  \[ a_p f(x + p) + \cdots + a_1 f(x + 1) + a_0 f(x) = 0 \]
  (involving the shift operator) to a difference equation (involving the forward difference operator)?
Scalar Products

• Given: a scalar product

\[ \langle f, g \rangle := \int_{a}^{b} f(x)g(x) \, d\mu(x) \]

with non-negative Borel measure \( \mu(x) \) supported in an interval \([a, b]\).

• Particular cases:
  – absolutely continuous measure \( d\mu(x) = \rho(x) \, dx \) with weight function \( \rho(x) \),
  – discrete measure \( \mu(x) = \rho(x) \) with support in \( \mathbb{Z} \).
Orthogonal Polynomials

- A family $P_n(x)$ of polynomials
  
  $$P_n(x) = k_n x^n + k'_n x^{n-1} + k''_n x^{n-2} + \cdots, \quad k_n \neq 0$$

  is called orthogonal w. r. t. the positive definite measure $\mu(x)$, if

  $$\langle P_m, P_n \rangle = \begin{cases} 
    0 & \text{if } m \neq n \\
    h_n > 0 & \text{if } m = n
  \end{cases}$$
Classical Families

- The classical orthogonal polynomials can be defined as the polynomial solutions of the differential equation:

\[ \sigma(x)P''_n(x) + \tau(x)P'_n(x) + \lambda_n P_n(x) = 0. \]

- Conclusions:
  - \( n = 1 \) implies \( \tau(x) = dx + e, d \neq 0 \)
  - \( n = 2 \) implies \( \sigma(x) = ax^2 + bx + c \)
  - coefficient of \( x^n \) implies \( \lambda_n = -n(a(n - 1) + d) \)
Classification

- The classical systems can be classified according to the following scheme (Bochner 1929):
  - $\sigma(x) = 0$ powers $x^n$
  - $\sigma(x) = 1$ Hermite polynomials
  - $\sigma(x) = x$ Laguerre polynomials
  - $\sigma(x) = x^2$ powers, Bessel polynomials
  - $\sigma(x) = x^2 - 1$ Jacobi polynomials
Hermite, Laguerre, Jacobi and Bessel
The weight function $\rho(x)$ corresponding to the differential equation satisfies Pearson’s differential equation

$$
\frac{d}{dx} \left( \sigma(x) \rho(x) \right) = \tau(x) \rho(x).
$$

Hence it is given as

$$
\rho(x) = \frac{C}{\sigma(x)} e^{\int \frac{\tau(x)}{\sigma(x)} dx}.
$$
Classical Discrete Families

- The classical “discrete” orthogonal polynomials can be defined as the polynomial solutions of the difference equation: \( (\nabla f(x) = f(x) - f(x - 1)) \)

\[
\sigma(x) \Delta^2 P_n(x) + \tau(x) \Delta P_n(x) + \lambda_n P_n(x) = 0.
\]

- Conclusions:
  - \( n = 1 \) implies \( \tau(x) = dx + e, d \neq 0 \)
  - \( n = 2 \) implies \( \sigma(x) = ax^2 + bx + c \)
  - coefficient of \( x^n \) implies \( \lambda_n = -n(a(n-1) + d) \)
Classification

- The classical discrete systems can be classified according to the scheme (Nikiforov, Suslov, Uvarov 1991):
  - $\sigma(x) = 0$  
    - falling factorials  
    - $x^n = x(x - 1) \cdots (x - n + 1)$
  - $\sigma(x) = 1$  
    - translated Charlier polynomials
  - $\sigma(x) = x$  
    - falling factorials, Charlier, Meixner, Krawtchouk polynomials
  - $\deg(\sigma(x), x) = 2$  
    - Hahn polynomials
Weight function

• The weight function \( \rho(x) \) corresponding to the difference equation satisfies Pearson’s difference equation

\[
\Delta \left( \sigma(x) \rho(x) \right) = \tau(x) \rho(x).
\]

• Hence it is given by

\[
\frac{\rho(x + 1)}{\rho(x)} = \frac{\sigma(x) + \tau(x)}{\sigma(x + 1)}.
\]
Hypergeometric Functions

The power series

\[ pF_q\left(\begin{array}{c} a_1, \ldots, a_p \\ b_1, \ldots, b_q \end{array} \bigg| z \right) = \sum_{k=0}^{\infty} A_k z^k, \]

whose summands \( \alpha_k = A_k z^k \) have rational term ratio

\[ \frac{\alpha_{k+1}}{\alpha_k} = \frac{A_{k+1} z^{k+1}}{A_k z^k} = \frac{(k + a_1) \cdots (k + a_p) z}{(k + b_1) \cdots (k + b_q) (k + 1)} \]

is called the generalized hypergeometric function.
The summand $\alpha_k = A_k z^k$ of a hypergeometric series is called a hypergeometric term w. r. t. $k$.

The relation
\[
\frac{\rho(x + 1)}{\rho(x)} = \frac{\sigma(x) + \tau(x)}{\sigma(x + 1)}
\]
therefore states that the weight functions $\rho(x)$ of classical discrete orthogonal polynomials are hypergeometric terms w. r. t. the variable $x$. 
Formula for Hypergeometric Terms

For the coefficients of the hypergeometric function one gets the formula

\[
pFq\left(\begin{array}{c} a_1, \ldots, a_p \\ b_1, \ldots, b_q \end{array} \right| z \right) = \sum_{k=0}^{\infty} \frac{(a_1)_k \cdots (a_p)_k}{(b_1)_k \cdots (b_q)_k} \frac{z^k}{k!},
\]

in terms of the Pochhammer symbol (or shifted factorial)

\[(a)_k = a(a + 1) \cdots (a + k - 1) = \frac{\Gamma(a + k)}{\Gamma(a)}.
\]
Classical Orthogonal Polynomials of Hahn Class as Hypergeometric Functions

- From the differential or difference equation, one can determine a hypergeometric representation. Maple

- One gets, for example, for the Laguerre polynomials

\[ L_\alpha^n(x) = \binom{n + \alpha}{n} \, _1F_1\left(\begin{array}{c} -n \\ \alpha + 1 \end{array} \right| x \right) = \sum_{k=0}^{n} \frac{(-1)^k}{k!} \binom{n + \alpha}{n - k} x^n, \]

and the Hahn polynomials are given by

\[ Q_{n}^{(\alpha, \beta)}(x, N) = \,_3F_2\left(\begin{array}{c} -n, -x, n + 1 + \alpha + \beta \\ \alpha + 1, -N \end{array} \right| 1 \right). \]
Properties of Classical Discrete Orthogonal Polynomials

Moreover, by linear algebra one can determine the coefficients of the following identities

\[(RE) \quad x \, P_n(x) = a_n \, P_{n+1}(x) + b_n \, P_n(x) + c_n \, P_{n-1}(x)\]

\[(DR) \quad \sigma(x) \, \Delta P_n(x) = \alpha_n \, P_{n+1}(x) + \beta_n \, P_n(x) + \gamma_n \, P_{n-1}(x)\]

\[(SR) \quad P_n(x) = \hat{a}_n \, \Delta P_{n+1}(x) + \hat{b}_n \, \Delta P_{n}(x) + \hat{c}_n \, \Delta P_{n-1}(x)\]

in terms of the given numbers \(a, b, c, d\) and \(e\).  

Maple
Zeilberger’s Algorithm

• Doron Zeilberger (1990) developed an algorithm to detect a holonomic recurrence equation for hypergeometric sums

\[ s_n = \sum_{k=\infty}^{\infty} F(n, k). \]

• A recurrence equation is called holonomic, if it is homogeneous, linear and has polynomial coefficients.
Zeilberger’s Algorithm

• A similar algorithm detects a holonomic differential equation for sums of the form

\[ s(x) = \sum_{k=-\infty}^{\infty} F(x, k). \]

• Holonomic functions form an algebra, i.e. sum and product of holonomic functions are holonomic, and there are linear algebra algorithms to compute the corresponding differential / recurrence equations.
Application to Orthogonal Polynomials

• As an example, we apply Zeilberger’s algorithm to the Laguerre polynomials

\[ L_\alpha^n(x) = \sum_{k=0}^{n} \frac{(-1)^k}{k!} \binom{n + \alpha}{n - k} x^n. \]

• Using the holonomic algebra, it is also easy to find recurrence and differential equations for the square \( L_\alpha^n(x)^2 \) and for the product \( L_\alpha^n(x) L_\beta^m(x) \).
The software used was developed for my book

Hypergeometric Summation, Vieweg, 1998, Braunschweig/Wiesbaden

and can be downloaded from my home page:

http://www.mathematik.uni-kassel.de/~koepf
Petkovsek-van Hoeij Algorithm

- Marko Petkovsek (1992) developed an algorithm to find all hypergeometric term solutions of a holonomic recurrence equation.

- This algorithm is not very efficient, but finishes the problem to find hypergeometric term representations of hypergeometric sums $s_n = \sum_{k=\infty}^{\infty} F(n, k)$ like $\sum_{k=0}^{n} (\binom{n}{k})^2$ algorithmically.

- Mark van Hoeij (1999) gave a very efficient version of such an algorithm, and implemented it in Maple.
Recurrence Operators

• Assume we consider the holonomic recurrence equation

\[ R f(x) := f(x + 2) - (x + 1) f(x + 1) + x^2 f(x) = 0. \]

• In the general setting the coefficients could be rational functions w.r.t. \( x \).

• Let \( \tau \) denote the shift operator \( \tau f(x) = f(x + 1) \).

Then the above recurrence equation can be rewritten as \( R f(x) = 0 \) with the operator polynomial

\[ R := \tau^2 - (x + 1) \tau + x^2. \]
Recurrence Operators

• Such operators form a non-commutative algebra.

• The product rule for the shift operator

\[
\tau \left( x f(x) \right) = (x + 1) f(x + 1) = (x + 1) \tau f(x)
\]

is equivalent to the commutator rule

\[
\tau x - x \tau = \tau
\]

in this algebra.
Some Facts

- An operator polynomial has a first order right factor iff the recurrence has a hypergeometric term solution.
- Hence Petkovsek’s algorithm finds first order right factors of operator polynomials.
- Multiplying an operator polynomial from the left by a rational function in $x$ is equivalent to multiply the recurrence equation by this rational function.
- Multiplying an operator polynomial from the left by $\tau$ is equivalent to substitute $x$ by $x + 1$ in the recurrence equation.
Construction of Fourth Order Recurrence

- Let us construct a fourth-order recurrence equation.
- To construct the equation \( S f(x) = 0 \) with operator
  \[ S := (x (x + 1) \tau^2 + x^3 \tau + (x^2 + x - 1)) \cdot R, \]
  we just add the equations
  \[
  (x^2 + x - 1) \left( f(x + 2) - (x + 1) f(x + 1) + x^2 f(x) \right) = 0
  \]
  \[
  x^3 \left( f(x + 3) - (x + 2) f(x + 2) + (x + 1)^2 f(x + 1) \right) = 0
  \]
  \[
  x (x+1) \left( f(x+4) - (x+3) f(x+3) + (x+2)^2 f(x+2) \right) = 0.
  \]
Factorization of Recurrence Equations

- This leads to

\[ S := x (x + 1) \tau^4 \]
\[ -x (4x + 3) \tau^3 \]
\[ + (x + 1) (3x^2 + 6x - 1) \tau^2 \]
\[ + (x + 1) (x^4 + x^3 - x^2 - x + 1) \tau \]
\[ + (x^2 + x - 1) x^2. \]

- Given \( S \), a factorization procedure by Mark van Hoeij can compute the factorization \( S = LR \), again.
Classical Orthogonal Polynomial Solutions of Recurrence Equations

- Previously we had shown how the recurrence equation can be explicitly expressed in terms of the coefficients of the differential / difference equation.

- If one uses this information in the opposite direction, then the corresponding differential / difference equation can be obtained from a given three-term recurrence.
Example

- Let the recurrence

\[ P_{n+2}(x) - (x - n - 1) P_{n+1}(x) + \alpha(n + 1)^2 P_n(x) = 0 \]

be given.

- We can compute that for \( \alpha = 1/4 \) this corresponds to translated Laguerre polynomials, and for \( \alpha < 1/4 \) Meixner and Krawtchouk polynomial solutions occur.
The End

Thank you very much for your attention!