

COMPUTER ALGEBRA METHODS FOR ORTHOGONAL POLYNOMIALS

W. KOEPF

University of Kassel
Dept. of Mathematics and Computer Science
Heinrich-Plett-Str. 40
34132 Kassel, Germany
E-mail: koepf@mathematik.uni-kassel.de

In this article we will show how computer algebra can be used in the study of orthogonal polynomials and special functions. The classical orthogonal polynomials named after Jacobi, Gegenbauer, Chebyshev, Legendre, Laguerre, Hermite and Bessel can be classified as the polynomial solutions of second order differential equations. Similarly the classical “discrete” orthogonal polynomials named after Hahn, Krawtchouk, Meixner and Charlier are classified as the polynomial solutions of second order difference equations. Using computer algebra one can compute the recurrence equations and hypergeometric representations of these systems, one can convert this process by computing differential and difference equations from the hypergeometric representations automatically, and one can decide whether a recurrence equation has classical orthogonal polynomial solutions. We will discuss these and related algorithms, and give some demonstrations with *Maple*.

1. 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*.

Internally the following algorithms are most prominently used: linear algebra techniques, multivariate polynomial factorization and the solution of nonlinear equations, e. g. by Gröbner basis techniques².

As an appetizer we consider the conversion between a recurrence equation and a difference equation.

In this article a difference equation is an equation involving the *forward difference operator*

$$\Delta f(x) = f(x + 1) - f(x) .$$

2

We ask the question: How can one convert a linear recurrence equation

$$a_p f(x+p) + \cdots + a_1 f(x+1) + a_0 f(x) = 0 \quad (1)$$

(involving the shift operator) to a difference equation (involving the forward difference operator)?

Here is a *Maple* worksheet for this purpose. First we load the `LREtools` package

```
> with(LREtools);
```

[*AnalyticityConditions, HypergeometricTerm, IsDesingularizable, REcontent, REcreate, REplot, REprimpart, REReduceorder, REtoDE, REtodelta, REtoproc, ValuesAtPoint, autodispersion, constcoeffsol, dAlembertiansols, delta, dispersion, divconq, firstlin, hypergeomsols, polysols, ratpolysols, riccati, shift*]

Let us start with the left hand side of the recurrence equation $n f_{n+2} - (n-1)(n+1) f_{n+1} + f_n = 0$ of type (1):

```
> RE:=n*f(n+2)-(n-1)*(n+1)*f(n+1)+f(n);
```

$$RE := n f(n+2) - (n-1)(n+1) f(n+1) + f(n)$$

The desired conversion rewrites the highest term $a_p f(x+p)$ of Eq. (1) in terms of $a_p \Delta^p f(x)$ which changes the coefficient of $f(x+p-1)$, and iterates this process. This procedure is already implemented in the `LREtools` package and can be invoked by the command `REtodelta`:

```
> L:=REtodelta(RE,f(n),{ });
```

$$L := n LREtools_{\Delta n}^2 + (-n^2 + 1 + 2n) LREtools_{\Delta n} + n + 2 - n^2$$

This results in an operator L such that $Lf(x) = 0$ is equivalent to the starting recurrence. The `LREtools` package uses its own Delta operator $LREtools_{\Delta}$ which may be replaced by the usual Δ :

```
> subs(LREtools[Delta][n]=Delta,deltaexpr);
```

$$n \Delta^2 + (2n - n^2 + 1) \Delta + 2 + n - n^2$$

so that the resulting operator looks more familiar.

Now let's ask the opposite question. How can we compute the original recurrence equation from the operator L ? This is easy: Replacing Δ and its powers by their defining equations does the job. For this purpose we implement the program `deltatoRE` and some subroutines by^a

```
deltaop:=(f,x)->subs(x=x+1,f)-f:
```

^aYou don't have to study this code in detail.

```

expanddelta:=(f,x,n)->
if n=0 then f(x)
  elif n=1 then f(x+1)-f(x)
  else
deltaop(expanddelta(f,x,n-1),x)
fi:

deltatoRE:=proc(deltaexpr,an)
local a,n,term,deg,k;
a:=op(0,an);
n:=op(1,an);
term:=deltaexpr-coeff(deltaexpr,LREtools[Delta][n],0)+
coeff(deltaexpr,LREtools[Delta][n],0)*a(n);
deg:=degree(deltaexpr,LREtools[Delta][n]);
for k from deg by -1 to 1 do
  term:=subs(LREtools[Delta][n]^k=expanddelta(a,n,k),term);
od;
collect(term,a,factor);
end:

```

Our function `deltatoRE` uses L and creates the original recurrence equation, again:

```
> deltaxoRE(L,f(n));
```

$$nf(n+2) - (n-1)(n+1)f(n+1) + f(n)$$

2. Classical Orthogonal Polynomials

Assume, 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 a real interval $[a, b]$ is given. One considers essentially the following 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} .
- discrete measure $\mu(x) = \rho(x)$ with support in $q^{\mathbb{Z}}$.

In the sequel we will only consider the first two cases, whereas the third instance is called the basic case, q denoting the base.

A family $P_n(x)$ of polynomials

$$P_n(x) = k_n x^n + k'_n x^{n-1} + k''_n x^{n-2} + \dots, \quad k_n \neq 0 \quad (2)$$

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

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

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

$$\sigma(x)P_n''(x) + \tau(x)P_n'(x) + \lambda_n P_n(x) = 0 . \quad (3)$$

Since one searches for solution families $P_n(x)$ according to Eq. (2) one gets the following 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 $\lambda_n = -n(a(n-1) + d)$

Therefore these differential equations are easily classified according to the five parameters a, b, c, d and e , the coefficients of the polynomials $\sigma(x)$ and $\tau(x)$.

By this procedure the classical systems can be classified according to the following scheme (Bochner¹):

- $\sigma(x) = 0$ 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

Every solution family $P_n(x)$ of Eq. (3) is a translate of one of the above systems. The powers are not orthogonal, whereas all other systems are. The Bessel polynomials are, however, not orthogonal in a real interval but on the unit circle of the complex plane.

It turns out that the corresponding Borel measure $d\mu(x)$ is absolutely continuous, and the corresponding weight function $\rho(x)$ fitting to the differential equation (3) satisfies *Pearson's differential equation*

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

Hence it is given as

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

The classical “discrete” orthogonal polynomials can be defined as the polynomial solutions of the difference equation

$$\sigma(x)\Delta\nabla P_n(x) + \tau(x)\Delta P_n(x) + \lambda_n P_n(x) = 0 \quad (4)$$

where $\nabla f(x) = f(x) - f(x-1)$ denotes the backward difference operator.

We get the same conclusions as before:

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

This leads—besides translations—to the classification⁷:

- $\sigma(x) = 0$ falling factorials $x^{\underline{n}} = x(x-1)\cdots(x-n+1)$
- $\sigma(x) = 1$ translated Charlier polynomials
- $\sigma(x) = x$ falling factorials, Charlier, Meixner and Krawtchouk polynomials
- $\deg(\sigma(x), x) = 2$ Hahn polynomials

Again, the falling factorials are not orthogonal, whereas the other systems are. The discrete weight function $\rho(x), x \in \mathbb{Z}$, corresponding to the difference equation (4) satisfies Pearson’s difference equation

$$\Delta(\sigma(x)\rho(x)) = \tau(x)\rho(x).$$

Hence it is given by the term ratio

$$\frac{\rho(x+1)}{\rho(x)} = \frac{\sigma(x) + \tau(x)}{\sigma(x+1)}. \quad (5)$$

3. Hypergeometric Functions

The power series

$${}_pF_q \left(\begin{matrix} a_1, \dots, a_p \\ b_1, \dots, b_q \end{matrix} \middle| z \right) = \sum_{k=0}^{\infty} A_k z^k,$$

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

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

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 .

Eq. (5) therefore states that the weight functions $\rho(x)$ of classical discrete orthogonal polynomials are hypergeometric terms w. r. t. the variable x .

For the coefficients of the hypergeometric function one gets the formula

$${}_pF_q \left(\begin{matrix} a_1, \dots, a_p \\ b_1, \dots, b_q \end{matrix} \middle| z \right) = \sum_{k=0}^{\infty} \frac{(a_1)_k \cdots (a_p)_k}{(b_1)_k \cdots (b_q)_k} \frac{z^k}{k!},$$

in terms of the *Pochhammer symbol* (or *shifted factorial*)

$$(a)_k = a(a+1) \cdots (a+k-1) = \frac{\Gamma(a+k)}{\Gamma(a)}.$$

From the differential or difference equation of a classical orthogonal system, one can determine a *hypergeometric representation*. This computation can be easily done with *Maple*. Here are the computations in the generic continuous case. We define the polynomials

```
> sigma:=a*x^2+b*x+c;
> tau:=d*x+e;
```

$$\begin{aligned} \sigma &:= ax^2 + bx + c \\ \tau &:= dx + e \end{aligned}$$

with arbitrary coefficients a, b, c, d and e and consider the differential equation

```
> DE:=
> sigma*dif(F(x),x$2)+tau*dif(F(x),x)-n*(a*n+d-a)*F(x);
```

$$DE := (ax^2 + bx + c) \left(\frac{d^2}{dx^2} F(x) \right) + (dx + e) \left(\frac{d}{dx} F(x) \right) - n(an + d - a)F(x)$$

To convert the differential equation to a recurrence equation for the series coefficients, we load the *gfun* (generating functions) package.

```
> with(gfun):
```

This package contains a function *diffeqtorec* which computes the recurrence equation for the Taylor coefficients A_j of a function

$$F(x) = \sum_{j=0}^{\infty} A_j x^j$$

satisfying a linear differential equation *DE*:

```
> RE:=diffeqtorec(DE,F(x),A(j));
```

$$RE := (aj^2 + (d-a)j - n^2a - nd + an)A(j) + (bj^2 + (e+b)j + e)A(j+1) + (cj^2 + 3cj + 2c)A(j+2)$$

In certain instances, this three-term recurrence collapses towards a two-term recurrence generating a hypergeometric representation for the family. This gives, for example, for the Laguerre polynomials

$$L_n^{(\alpha)}(x) = \binom{n+\alpha}{n} {}_1F_1\left(\begin{matrix} -n \\ \alpha+1 \end{matrix} \middle| x\right) = \sum_{k=0}^n \frac{(-1)^k}{k!} \binom{n+\alpha}{n-k} x^k.$$

In the Jacobi case, the point of development should be $x = \pm 1$ to get a hypergeometric power series. As a classical discrete example, the Hahn polynomials are given by

$$Q_n^{(\alpha,\beta)}(x, N) = {}_3F_2\left(\begin{matrix} -n, -x, n+1+\alpha+\beta \\ \alpha+1, -N \end{matrix} \middle| 1\right).$$

Note that in a similar fashion the basic discrete case can be handled and leads to a classification of the so-called q -Hahn class of orthogonal polynomials, that have a basic hypergeometric representation, see e. g. Lesky⁶.

4. Computation of the Recurrence Coefficients

Moreover, by linear algebra one can determine the coefficients of the following identities (written for the discrete case)

- (Recurrence Equation)

$$x P_n(x) = a_n P_{n+1}(x) + b_n P_n(x) + c_n P_{n-1}(x) \quad (6)$$

- (Difference Rule)

$$\sigma(x) \Delta P_n(x) = \alpha_n P_{n+1}(x) + \beta_n P_n(x) + \gamma_n P_{n-1}(x)$$

- (Structure Relation)

$$P_n(x) = \hat{a}_n \Delta P_{n+1}(x) + \hat{b}_n \Delta P_n(x) + \hat{c}_n \Delta P_{n-1}(x)$$

in terms of the given numbers a, b, c, d and e . A recurrence equation of type (6) is valid for every orthogonal system, whereas difference rule (or differential rule) and structure relation are characteristics of the *classical systems*.

With *Maple*, we will now determine the coefficients a_n, b_n and c_n of the recurrence equation (6) in terms of a, b, c, d and e . As an example, we

8

deal with the classical continuous case. Note that the discrete case can be treated in a similar fashion.

We consider the three highest coefficients of the orthogonal polynomial:

> $p := k[n] * x^n + kprime[n] * x^{(n-1)} + kprimeprime[n] * x^{(n-2)};$

$$p := k_n x^n + kprime_n x^{(n-1)} + kprimeprime_n x^{(n-2)}$$

The polynomial satisfies the differential equation $DE = 0$ with

> $DE := \sigma * \text{diff}(p, x^2) + \tau * \text{diff}(p, x) + \lambda[n] * p;$

$$\begin{aligned} DE := & (a x^2 + b x + c) \left(\frac{k_n x^n n^2}{x^2} - \frac{k_n x^n n}{x^2} + \frac{kprime_n x^{(n-1)} (n-1)^2}{x^2} \right. \\ & - \frac{kprime_n x^{(n-1)} (n-1)}{x^2} + \frac{kprimeprime_n x^{(n-2)} (n-2)^2}{x^2} \\ & \left. - \frac{kprimeprime_n x^{(n-2)} (n-2)}{x^2} \right) + (d x + e) \\ & \left(\frac{k_n x^n n}{x} + \frac{kprime_n x^{(n-1)} (n-1)}{x} + \frac{kprimeprime_n x^{(n-2)} (n-2)}{x} \right) \\ & + \lambda_n (k_n x^n + kprime_n x^{(n-1)} + kprimeprime_n x^{(n-2)}) \end{aligned}$$

After division by x^{n-4} , we collect coefficients:

> $de := \text{collect}(\text{simplify}(DE/x^{(n-4)}), x);$

$$\begin{aligned} de := & (-a k_n n + \lambda_n k_n + d k_n n + a k_n n^2) x^4 + (-3 a kprime_n n + b k_n n^2 \\ & + a kprime_n n^2 + 2 a kprime_n + \lambda_n kprime_n - d kprime_n + e k_n n \\ & - b k_n n + d kprime_n n) x^3 + (-5 a kprimeprime_n n \\ & - 2 d kprimeprime_n - e kprime_n - c k_n n - 3 b kprime_n n \\ & + 2 b kprime_n + c k_n n^2 + e kprime_n n + 6 a kprimeprime_n \\ & + d kprimeprime_n n + \lambda_n kprimeprime_n + a kprimeprime_n n^2 \\ & + b kprime_n n^2) x^2 + (c kprime_n n^2 - 5 b kprimeprime_n n \\ & + 2 c kprime_n - 3 c kprime_n + e kprimeprime_n n \\ & + b kprimeprime_n n^2 + 6 b kprimeprime_n - 2 e kprimeprime_n) x \\ & - 5 c kprimeprime_n n + 6 c kprimeprime_n + c kprimeprime_n n^2 \end{aligned}$$

Equating the highest coefficient gives the already mentioned identity for λ_n :

> $rule1 := \lambda[n] = \text{solve}(\text{coeff}(de, x, 4), \lambda[n]);$

$$rule1 := \lambda_n = -n(a n + d - a)$$

This equation can be substituted:

> `de:=expand(subs(rule1,de));`

$$\begin{aligned} de := & 2x^3 a k_{prime_n} + 6x^2 a k_{primeprime_n} + 2x^2 b k_{prime_n} \\ & + 6x b k_{primeprime_n} + 2x c k_{prime_n} + 6c k_{primeprime_n} \\ & - x^3 d k_{prime_n} - 2x^2 d k_{primeprime_n} - x^2 e k_{prime_n} \\ & - 2x e k_{primeprime_n} + x^2 c k_n n^2 - x^2 c k_n n - 2x^3 a k_{prime_n} n \\ & - 4x^2 a k_{primeprime_n} n + x^3 b k_n n^2 - x^3 b k_n n + x^2 b k_{prime_n} n^2 \\ & - 3x^2 b k_{prime_n} n + x b k_{primeprime_n} n^2 - 5x b k_{primeprime_n} n \\ & + x c k_{prime_n} n^2 - 3x c k_{prime_n} n + c k_{primeprime_n} n^2 \\ & - 5c k_{primeprime_n} n + x^3 e k_n n + x^2 e k_{prime_n} n \\ & + x e k_{primeprime_n} n \end{aligned}$$

Equating the second highest coefficient gives k'_n as rational multiple of k_n :

> `rule2:=kprime[n]=solve(coeff(de,x,3),kprime[n]);`

$$rule2 := k_{prime_n} = \frac{k_n n (e + b n - b)}{-2a + d + 2an}$$

Equating the third highest coefficient gives k''_n as rational multiple of k_n :

> `rule3:=kprimeprime[n]=`
> `solve(coeff(subs(rule2,de),x,2),kprimeprime[n]);`

$$\begin{aligned} rule3 := & k_{primeprime_n} = \frac{1}{2} k_n n (3be + 5b^2 n - 2b^2 + e^2 n + 2en^2 b \\ & - 5enb - e^2 - 4cna + cnd + 2cn^2 a + 2ca - cd + b^2 n^3 \\ & - 4b^2 n^2) / ((-2a + d + 2an)(-3a + d + 2an)) \end{aligned}$$

Without loss of generality we consider the monic case and set

> `k[n]:=1;`

$$k_n := 1$$

We would like to compute the coefficients a_n, b_n and c_n in the recurrence equation $RE = 0$ with:

> `RE:=x*P(n)-(a[n]*P(n+1)+b[n]*P(n)+c[n]*P(n-1));`

$$RE := xP(n) - a_n P(n+1) - b_n P(n) - c_n P(n-1)$$

We substitute $P_n(x)$:

> `RE:=subs(`
> `{P(n)=p,P(n+1)=subs(n=n+1,p),P(n-1)=subs(n=n-1,p)},`
> `RE);`

$$\begin{aligned}
RE &:= x(x^n + kprime_n x^{(n-1)} + kprimeprime_n x^{(n-2)}) \\
&- a_n(x^{(n+1)} + kprime_{n+1} x^n + kprimeprime_{n+1} x^{(n-1)}) \\
&- b_n(x^n + kprime_n x^{(n-1)} + kprimeprime_n x^{(n-2)}) \\
&- c_n(x^{(n-1)} + kprime_{n-1} x^{(n-2)} + kprimeprime_{n-1} x^{(n-3)})
\end{aligned}$$

We substitute the already known formulas:

```

> RE:=subs(
> {rule2,subs(n=n+1,rule2),subs(n=n-1,rule2),
> rule3,subs(n=n+1,rule3),subs(n=n-1,rule3)},
> RE);

```

$$\begin{aligned}
RE &:= x(x^n + \frac{n(e+bn-b)x^{(n-1)}}{-2a+d+2an} + n(3be+5b^2n-2b^2+e^2n \\
&+ 2en^2b-5enb-e^2-4cna+cnd+2cn^2a+2ca-cd \\
&+ b^2n^3-4b^2n^2)x^{(n-2)}/(2(-2a+d+2an)(-3a+d+2an))) \\
&- a_n(x^{(n+1)} + \frac{(n+1)(e+b(n+1)-b)x^n}{-2a+d+2a(n+1)} + (n+1)(3be \\
&+ 5b^2(n+1)-2b^2+e^2(n+1)+2e(n+1)^2b-5e(n+1)b \\
&- e^2-4c(n+1)a+c(n+1)d+2c(n+1)^2a+2ca-cd \\
&+ b^2(n+1)^3-4b^2(n+1)^2)x^{(n-1)}/(2(-2a+d+2a(n+1)) \\
&(-3a+d+2a(n+1)))) - b_n(x^n + \frac{n(e+bn-b)x^{(n-1)}}{-2a+d+2an} + n(\\
&3be+5b^2n-2b^2+e^2n+2en^2b-5enb-e^2-4cna+cnd \\
&+ 2cn^2a+2ca-cd+b^2n^3-4b^2n^2)x^{(n-2)}/(2 \\
&(-2a+d+2an)(-3a+d+2an))) - c_n(x^{(n-1)} \\
&+ \frac{(n-1)(e+b(n-1)-b)x^{(n-2)}}{-2a+d+2a(n-1)} + (n-1)(3be+5b^2(n-1) \\
&- 2b^2+e^2(n-1)+2e(n-1)^2b-5e(n-1)b-e^2 \\
&- 4c(n-1)a+c(n-1)d+2c(n-1)^2a+2ca-cd \\
&+ b^2(n-1)^3-4b^2(n-1)^2)x^{(n-3)}/(2(-2a+d+2a(n-1)) \\
&(-3a+d+2a(n-1))))
\end{aligned}$$

After putting this rational expression in the form numerator/denominator with `normal`, the numerator must vanish:

```

> re:=simplify(numer(normal(RE))/x^(n-3)):

```

Equating the highest coefficient gives for monic polynomials

```

> rule4:=a[n]=solve(coeff(re,x,4),a[n]);

```

$$\text{rule4} := a_n = 1$$

and equating the second highest coefficient yields

```
> rule5:=b[n]=factor(solve(subs(rule4,coeff(re,x,3)),b[n]));
```

$$\text{rule5} := b_n = \frac{-2bn^2a + 2bna + 2ea - 2bnd - ed}{(d + 2an)(-2a + d + 2an)}$$

Finally equating the third highest coefficient leads to

```
> rule6:=c[n]=
> factor(solve(subs(rule5,subs(rule4,coeff(re,x,2))),c[n]));
```

$$\begin{aligned} \text{rule6} := c_n = & -n(an + d - 2a)(4a^2n^2c - 8a^2cn + 4a^2c - ab^2n^2 + 4acnd \\ & + 2ab^2n + ae^2 - ab^2 - 4acd - b^2dn - bed + cd^2 + b^2d)/((d - a + 2an) \\ & (-3a + d + 2an)(-2a + d + 2an)^2) \end{aligned}$$

In a similar fashion the differential (difference) rule and structure relation can be handled.

5. Zeilberger's Algorithm

Doron Zeilberger (1990)¹⁰ 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.

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.

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

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

For this purpose, we load the package `hsum.mpl` from my book *Hypergeometric Summation*^{3b}

```
> read "hsum9.mpl";
```

*Package "Hypergeometric Summation", Maple V – Maple 9
Copyright 1998 – 2004, Wolfram Koepf, University of Kassel*

We define the hypergeometric summand of the Laguerre polynomials

```
> laguerreterm:=(-1)^k/k!*binomial(n+alpha,n-k)*x^k;
```

$$\text{laguerreterm} := \frac{(-1)^k \text{binomial}(n + \alpha, n - k) x^k}{k!}$$

and use Zeilberger's algorithm to detect a recurrence equation for the sum, hence for the Laguerre polynomials:

```
> LaguerreRE:=sumrecursion(laguerreterm,k,L(n));
```

LaguerreRE :=

$$(n + \alpha + 1)L(n) + (x - 2n - \alpha - 3)L(n + 1) + (n + 2)L(n + 2) = 0$$

Similarly, a recurrence equation w.r.t. α is obtained

```
> sumrecursion(laguerreterm,k,L(alpha));
```

$$(n + \alpha + 1)L(\alpha) - (x + \alpha + 1)L(\alpha + 1) + xL(\alpha + 2) = 0$$

Next, we compute the differential equation of the Laguerre polynomials from their hypergeometric representation:

```
> LaguerreDE:=sumdiffeq(laguerreterm,k,L(x));
```

$$\text{LaguerreDE} := x \left(\frac{d^2}{dx^2} L(x) \right) - (x - \alpha - 1) \left(\frac{d}{dx} L(x) \right) + L(x) n = 0$$

Using the holonomic algebra, it is furthermore easy to find recurrence and differential equations for the square $L_n^{(\alpha)}(x)^2$ and for the product $L_n^{(\alpha)}(x) L_m^{(\beta)}(x)$ or any other sum or product of holonomic functions.

The following computation using the `gfun` package takes the recurrence equation for $L_n^{(\alpha)}(x)$ and generates the recurrence equation valid for the square of the Laguerre polynomials (including some initial values)

```
> 'rec*rec'(LaguerreRE,LaguerreRE,L(n));
```

^bThe software can be downloaded from my home page <http://www.mathematik.uni-kassel.de/~koepf/Publikationen>.

$$\begin{aligned}
& \{(10 - 2x + 29n + 30n^2 + 25\alpha^2 - 5x\alpha + 2n^4 + 13n^3 + 27\alpha + 9\alpha^3 + \alpha^4 \\
& - xn^3 - 4xn^2 + 7n^3\alpha + 9n^2\alpha^2 + 35n^2\alpha + 31n\alpha^2 + 5n\alpha^3 \\
& - x\alpha^3 - 4x\alpha^2 - 3xn^2\alpha - 3xn\alpha^2 - 8xn\alpha - 5xn + 55n\alpha) \\
& L(n) + (-66 + 70x - 149n - 124n^2 - 47\alpha^2 + 75x\alpha - 6n^4 \\
& - 45n^3 - 22x^2 + 2x^3 - 91\alpha - 11\alpha^3 - \alpha^4 + x^3n - 6x^2n^2 \\
& - 23x^2n + 11xn^3 + 62xn^2 - 15n^3\alpha - 14n^2\alpha^2 - 84n^2\alpha \\
& - 52n\alpha^2 - 6n\alpha^3 + x^3\alpha - 3x^2\alpha^2 - 17x^2\alpha + 3x\alpha^3 + 26x\alpha^2 \\
& - 9x^2n\alpha + 22xn^2\alpha + 14xn\alpha^2 + 82xn\alpha + 115xn - 153n\alpha) \\
& L(n+1) + (110 - 102x + 219n + 160n^2 + 22\alpha^2 - 48x\alpha + 6n^4 \\
& + 51n^3 + 26x^2 - 2x^3 + 82\alpha + 2\alpha^3 - x^3n + 6x^2n^2 + 25x^2n \\
& - 11xn^3 - 70xn^2 + 9n^3\alpha + 5n^2\alpha^2 + 57n^2\alpha + 21n\alpha^2 + n\alpha^3 \\
& + 6x^2\alpha - 6x\alpha^2 + 3x^2n\alpha - 11xn^2\alpha - 3xn\alpha^2 - 46xn\alpha \\
& - 147xn + 119n\alpha)L(n+2) + (xn^3 + 8xn^2 + 21xn + 18x \\
& - 2n^4 - 19n^3 - 66n^2 - 99n - n^3\alpha - 8n^2\alpha - 21n\alpha - 18\alpha - 54) \\
& L(n+3), L(2) = \frac{1}{4} - C_0 - C_2 + \frac{9}{4} - C_1 - C_3 - \frac{3}{4} - C_0 - C_3 - \frac{3}{4} - C_1 - C_2 \\
& + \frac{1}{4} - C_1x - C_2\alpha - \frac{1}{2} - C_1\alpha - C_3x + \frac{1}{4} - C_0\alpha - C_3x + \frac{1}{4} - C_0\alpha^2 - C_2 \\
& - \frac{1}{4} - C_0\alpha^2 - C_3 - \frac{1}{4} - C_1\alpha^2 - C_2 + \frac{1}{4} - C_1\alpha^2 - C_3 + \frac{1}{4} - C_1x^2 - C_3 \\
& + \frac{1}{2} - C_0\alpha - C_2 - C_0\alpha - C_3 + \frac{1}{4} - C_0 - C_3x - C_1\alpha - C_2 \\
& + \frac{3}{2} - C_1\alpha - C_3 - \frac{3}{2} - C_1 - C_3x + \frac{1}{4} - C_1x - C_2, L(0) = -C_0 - C_2, \\
& L(1) = -C_1 - C_3\}
\end{aligned}$$

Next, we compute the differential equation for the square of the Laguerre polynomials

> 'diffeq*diffeq'(LaguerreDE, LaguerreDE, L(x));

$$\begin{aligned}
& (-4xn + 4n\alpha + 2n)L(x) \\
& + (4xn + 2x^2 + 3\alpha + 1 + 2\alpha^2 - 4x\alpha - 4x)\left(\frac{d}{dx}L(x)\right) \\
& + (-3x^2 + 3x\alpha + 3x)\left(\frac{d^2}{dx^2}L(x)\right) + \left(\frac{d^3}{dx^3}L(x)\right)x^2
\end{aligned}$$

and this is finally the (fourth order) differential equation for the product $L_n^{(\alpha)}(x)L_m^{(\beta)}(x)$ of the Laguerre polynomials.

> 'diffeq*diffeq'(LaguerreDE, \\
> subs(alpha=beta, n=m, LaguerreDE), L(x));

$$\begin{aligned}
& (5\beta^2 n^2 x + 4x n^2 \alpha \beta + 2\beta n^2 x - 2m^2 \alpha x - m\beta \alpha^2 - 2m^2 \beta x + m\alpha^2 \beta^2 \\
& + 2m\alpha \beta^2 + \alpha n \beta^2 - 2\beta n \alpha^2 - 12m x^2 n^2 - 5m^2 \alpha^2 x + m^2 \beta^2 x \\
& + 12m^2 x^2 n - 12m x^2 n \alpha + 6m x n \alpha^2 + 12m \beta x^2 n - 6m \beta^2 x n \\
& - 4m^2 x \alpha \beta - \alpha^3 \beta n - \alpha^3 m \beta + 4x^3 n \alpha - 4x^3 n \beta + 4x^3 m \alpha \\
& - 4x^3 m \beta - 4x^2 \alpha^2 n + x \alpha^3 n - 2x^2 \alpha n^2 + 4x^2 m \beta^2 - 4x^2 \alpha n \beta \\
& - x \alpha n \beta^2 + 4x^2 m \beta \alpha + x m \beta \alpha^2 + 5x \beta n \alpha^2 - 5x m \alpha \beta^2 \\
& + 8x^3 n^2 - 8x^3 m^2 - x m \beta^3 + 2x^2 m^2 \beta + 8x^2 \beta^2 n - 5x \beta^3 n \\
& - 14x^2 \beta n^2 - 8x^2 m \alpha^2 + 5x m \alpha^3 + 14x^2 m^2 \alpha + 12m^2 x^2 \\
& - 12x^2 n^2 - n \alpha^2 - n \alpha^3 - 6x^2 n \alpha + 2x n^2 \alpha + 6x n \alpha^2 - 6x^2 \alpha m \\
& + 6\beta x^2 m + 6\beta x^2 n + 6\alpha^2 m x - 6\beta^2 m x - 6\beta^2 x n + \beta^3 \alpha n \\
& + \beta^3 \alpha m - \alpha^2 n \beta^2 - \alpha^2 n^2 x + \beta^4 n + 4x^2 n^3 + n \beta^2 - m \alpha^2 + m \beta^2 \\
& + 2\beta^3 n - 4m^3 x^2 - m \alpha^4 - 2m \alpha^3 + m \beta^3) L(x) + (-\alpha^2 + \beta^2 \\
& - 2m x \beta + 2x n \beta - 9\beta x^2 \alpha^2 - 6\beta^3 x \alpha - 2x m \alpha + 4x^3 n \alpha \\
& + 20x^3 n \beta - 20x^3 m \alpha - 4x^3 m \beta + 4x^2 \alpha^2 n - 2x \alpha^3 n \\
& + 8x^2 \alpha n^2 - 4x^2 m \beta^2 - 6\beta \alpha m x - 16x^2 \alpha n \beta + 6x \alpha n \beta^2 \\
& + 16x^2 m \beta \alpha - 6x m \beta \alpha^2 + 2x \beta n \alpha^2 - 2x m \alpha \beta^2 + 8x^3 \alpha^2 \\
& - 3\beta \alpha^3 + 6\beta \alpha x n + 3\beta^3 \alpha - 16x^3 n^2 + 16x^3 m^2 - 32m x^3 \\
& + 8m x^4 + 2x m \beta^3 - 8x^2 m^2 \beta - 12x^2 \beta^2 n + 2x \beta^3 n + 8x^2 \beta n^2 \\
& + 12x^2 m \alpha^2 - 2x m \alpha^3 - 8x^2 m^2 \alpha + 21\beta^2 x^2 - 9\beta^3 x + \beta^4 \\
& - 12m^2 x^2 - 2\alpha^3 - \alpha^4 + 9x^2 \alpha \beta^2 + 6x \alpha^3 \beta + 32x^3 n + 12x^2 n^2 \\
& - 20x^2 n + 16x^3 \alpha - 21x^2 \alpha^2 - 10x^2 \alpha + 9x \alpha^3 + 10x \alpha^2 \\
& - 16x^2 n \alpha - 2x n \alpha^2 + 2x n \alpha + 10\beta x^2 - 10\beta^2 x - 2\alpha^2 \beta \\
& + 20m x^2 + 2\beta^3 - 5x^2 \alpha^3 - 8\beta^2 x^3 + 5\beta^3 x^2 - 9x \alpha \beta^2 \\
& + 28x^2 \alpha m + 9\beta x \alpha^2 + 16\beta x^2 m - 28\beta x^2 n - 8\alpha^2 m x \\
& + 2\beta^2 m x + 8\beta^2 x n - \alpha^4 \beta - \alpha^3 \beta^2 - 8x^4 n - 4x^4 \alpha + 4\beta x^4 \\
& - \beta^4 x + \beta^3 \alpha^2 + \beta^4 \alpha - 16\beta x^3 + 2\alpha \beta^2 + \alpha^4 x) (\frac{d}{dx} L(x)) + (\\
& 9\beta x^2 \alpha^2 + 3\beta^3 x \alpha - 16x^3 n \alpha - 24x^3 n \beta + 24x^3 m \alpha + 16x^3 m \beta \\
& + 2x^2 \alpha^2 n - 2x^2 m \beta^2 + 12x^2 \alpha n \beta - 12x^2 m \beta \alpha - 15x^3 \alpha^2 \\
& + 8x^3 n^2 - 8x^3 m^2 + 52m x^3 - 20m x^4 + 6x^2 \beta^2 n - 6x^2 m \alpha^2 \\
& - 25\beta^2 x^2 + 6\beta^3 x - 9x^2 \alpha \beta^2 - 3x \alpha^3 \beta - 52x^3 n + 16x^2 n \\
& - 26x^3 \alpha + 25x^2 \alpha^2 + 8x^2 \alpha - 6x \alpha^3 - 7x \alpha^2 + 18x^2 n \alpha - 8\beta x^2 \\
& + 7\beta^2 x - 16m x^2 + 7x^2 \alpha^3 + 15\beta^2 x^3 - 7\beta^3 x^2 + 6x \alpha \beta^2 \\
& - 18x^2 \alpha m - 6\beta x \alpha^2 - 18\beta x^2 m + 18\beta x^2 n + 20x^4 n + 10x^4 \alpha \\
& - 10\beta x^4 + \beta^4 x + 26\beta x^3 - \alpha^4 x) (\frac{d^2}{dx^2} L(x)) + (-8\beta^2 x^3 + 6\beta^2 x^2 \\
& + 8x^3 \alpha^2 + 20x^3 n + 16m x^4 - 16x^4 n - 20m x^3 + 10x^3 \alpha \\
& - 10\beta x^3 - 6x^2 \alpha^2 - 8x^4 \alpha + 8\beta x^4 + 2x^2 \alpha \beta^2 - 8x^3 m \alpha \\
& + 8x^3 n \alpha - 2\beta x^2 \alpha^2 - 8x^3 m \beta + 8x^3 n \beta - 2x^2 \alpha^3 + 2\beta^3 x^2) \\
& (\frac{d^3}{dx^3} L(x)) \\
& + (2x^4 \alpha - 2\beta x^4 - x^3 \alpha^2 + \beta^2 x^3 - 4m x^4 + 4x^4 n) (\frac{d^4}{dx^4} L(x))
\end{aligned}$$

6. Petkovsek-van Hoeij Algorithm

Marko Petkovsek⁸ 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 Hoeij⁹ gave a very efficient version of such an algorithm, and implemented it in Maple.

We would like to find a simple representation for

```
> s:=Sum(binomial(n-2*k,k)*(-4/27)^k,k=0..floor(n/3));
```

$$s := \sum_{k=0}^{\lfloor \frac{n}{3} \rfloor} \binom{n-2k}{k} \left(\frac{-4}{27}\right)^k$$

Hence we define the summand

```
> summand:=binomial(n-2*k,k)*(-4/27)^k;
```

$$\text{summand} := \binom{n-2k}{k} \left(\frac{-4}{27}\right)^k$$

and compute the recurrence equation for the sum s_n using Zeilberger's algorithm:

```
> RE:=sumrecursion(summand,k,S(n));
```

$$RE := 2(n+3)S(n) + 3(n+4)S(n+1) - 9(n+2)S(n+2) = 0$$

Since the recurrence RE is second order, we try to find the hypergeometric term solutions of RE using van Hoeij's algorithm:

```
> res:='LREtools/hsols'(RE,S(n));
```

$$\text{res} := \left[\left(\frac{-1}{3}\right)^n, \left(\frac{2}{3}\right)^n \left(\frac{4}{3} + n\right)\right]$$

Since our second order recurrence has two linearly independent hypergeometric term solutions, every solution of RE must be a linear combination of them. Hence we can set for s_n

```
> result:=alpha*op(1,res)+beta*op(2,res);
```

$$\text{result} := \alpha \left(\frac{-1}{3}\right)^n + \beta \left(\frac{2}{3}\right)^n \left(\frac{4}{3} + n\right)$$

and we can find the coefficients α and β using two initial values by linear algebra:

16

```
> sol:=solve({
> eval(subs(Sum=sum,s=result),n=0),
> eval(subs(Sum=sum,s=result),n=1)},
> {alpha,beta});
```

$$sol := \{\beta = \frac{2}{3}, \alpha = \frac{1}{9}\}$$

Therefore, we have finally found the simple formula for s_n :

```
> result:=subs(sol,result);
```

$$result := \frac{\left(\frac{-1}{3}\right)^n}{9} + \frac{2\left(\frac{2}{3}\right)^n\left(\frac{4}{3} + n\right)}{3}$$

7. Recurrence Operators

Assume we consider the holonomic recurrence equation

$$Rf(x) := f(x+2) - (x+1)f(x+1) + x^2f(x) = 0.$$

In the general setting the coefficients could be rational functions w.r.t. x .

Let τ denote the shift operator $\tau f(x) = f(x+1)$. Then the above recurrence equation can be rewritten as $Rf(x) = 0$ with the operator polynomial

$$R := \tau^2 - (x+1)\tau + x^2.$$

Such operators form a non-commutative algebra.

The product rule for the shift operator

$$\tau(xf(x)) = (x+1)f(x+1) = (x+1)\tau f(x)$$

is equivalent to the commutator rule

$$\tau x - x\tau = \tau$$

in this algebra.

An operator polynomial has a first order right factor iff the recurrence has a hypergeometric term solution. Hence the Petkovsek-van Hoeij 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 τ is equivalent to substitute x by $x+1$ in the recurrence equation.

Let us construct a fourth-order recurrence equation from R . To construct the equation $Sf(x) = 0$ with operator

$$S := (x(x+1)\tau^2 + x^3\tau + (x^2 + x - 1)) \cdot R,$$

we just add the equations

$$(x^2 + x - 1) \left(f(x+2) - (x+1)f(x+1) + x^2 f(x) \right) = 0$$

$$x^3 \left(f(x+3) - (x+2)f(x+2) + (x+1)^2 f(x+1) \right) = 0$$

$$x(x+1) \left(f(x+4) - (x+3)f(x+3) + (x+2)^2 f(x+2) \right) = 0.$$

This leads to

$$\begin{aligned} S := & x(x+1)\tau^4 \\ & -x(4x+3)\tau^3 \\ & + (x+1)(3x^2+6x-1)\tau^2 \\ & + (x+1)(x^4+x^3-x^2-x+1)\tau \\ & + (x^2+x-1)x^2. \end{aligned}$$

Given S , a factorization procedure by Mark van Hoeij can compute the factorization $S = LR$, again. For this purpose, we load his package:

```
> read "FactorOrder4-discrete";
```

Warning, the name delta has been redefined

```
._Env_LRE_x := x
```

```
._Env_LRE_tau := tau
```

The following command factors S and returns the right factor found:

```
> fact:=FactorOrder4(RE);
```

$$fact := \{\tau^2 + (-x-1)\tau + x^2\}$$

which, of course, equals R .

8. 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 of the classical orthogonal systems.

If one uses this information in the opposite direction, then the corresponding differential / difference equation can be obtained from a given three-term recurrence. In this algorithm a polynomial equation system containing linear and quadratic polynomials must be solved⁵.

Let the recurrence

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

be given.

Our program computes that for $\alpha = 1/4$ this corresponds to translated Laguerre polynomials, and for $\alpha < 1/4$ translated Meixner and Krawtchouk polynomial solutions occur:

```
> read "retode.mpl";
```

Package "REtoDE", Maple V – Maple 8

Copyright 2000 – 2002, Wolfram Koeopf, University of Kassel

```
> RE:=P(n+2)-(x-n-1)*P(n+1)+alpha*(n+1)^2*P(n)=0;
```

$$RE := P(n + 2) - (x - n - 1)P(n + 1) + \alpha(n + 1)^2P(n) = 0$$

The following command gives the classical continuous solutions

```
> REtoDE(RE,P(n),x);
```

Warning : parameters have the values,

$$\{b = 2c, a = 0, \alpha = \frac{1}{4}, c = c, d = -4c, e = 0\}$$

$$\left[\frac{1}{2}(2x + 1)\left(\frac{\partial^2}{\partial x^2}P(n, x)\right) - 2x\left(\frac{\partial}{\partial x}P(n, x)\right) + 2nP(n, x) = 0,\right.$$

$$\left.[I = \left[\frac{-1}{2}, \infty\right], \rho(x) = 2e^{(-2x)}, \frac{k_{n+1}}{k_n} = 1\right]$$

and finally, the following command gives the classical discrete solutions:

```
> REtodiscreteDE(RE,P(n),x);
```

Warning : parameters have the values, $\{\alpha = \frac{f^2 - 1}{4f^2}, f = f, d = d,$

$$c = -\frac{1}{4}f^2d + \frac{1}{4}d + \frac{1}{2}gdf + \frac{1}{2}gd, a = 0, g = g, e = -gd,$$

$$b = -\frac{1}{2}fd - \frac{1}{2}d\}$$

$$\begin{aligned} & \left[\frac{1}{2} \frac{(f + 2fx - 1) \Delta(\text{Nabla}(P(n, fx + g), x), x)}{f} - \frac{2x \Delta(P(n, fx + g), x)}{f + 1} \right. \\ & \left. + \frac{2n P(n, fx + g)}{(f + 1)f} = 0, \right. \\ & \left. [\sigma(x) = \frac{f}{2} + x - \frac{1}{2} - g, \sigma(x) + \tau(x) = \frac{(f - 1)(f + 2x + 1 - 2g)}{2(f + 1)}], \right. \\ & \left. \rho(x) = \left(\frac{f - 1}{f + 1}\right)^x, \frac{k_{n+1}}{k_n} = \frac{1}{f} \right] \end{aligned}$$

9. Final Remarks

I hope this survey article has shown some of the capabilities that lie in the use of computer algebra systems when working with orthogonal polynomials and special functions. I wish you much success with their use!

References

1. S. Bochner: Über Sturm-Liouvillesche Polynomsysteme. *Math. Z.* **29**, 730–736 (1929).
2. K. O. Geddes, S. R. Czapor and G. Labahn, *Algorithms for Computer Algebra*, Kluwer, Boston/Dordrecht/London, 1992.
3. W. Koepf, *Hypergeometric Summation*, Vieweg, Braunschweig/Wiesbaden, 1998.
4. W. Koepf and D. Schmersau, Representations of orthogonal polynomials, *J. Comp. Appl. Math.* **90**, 57–94 (1998).
5. W. Koepf and D. Schmersau, Recurrence equations and their classical orthogonal polynomial solutions, *Appl. Math. Comput.* **128**, 303–327 (2002).
6. P. Lesky, *Eine Charakterisierung der klassischen kontinuierlichen, diskreten und q-Orthogonalpolynome*, Shaker, Aachen, 2005.
7. A. F. Nikiforov, S. K. Suslov and V. B. Uvarov, *Classical orthogonal polynomials of a discrete variable*, Springer, Berlin, 1991.
8. M. Petkovsek: Hypergeometric solutions of linear recurrences with polynomial coefficients. *J. Symbolic Computation* **14**, 243–264 (1992).
9. M. van Hoeij: Finite singularities and hypergeometric solutions of linear recurrence equations. *J. Pure Appl. Algebra* **139**, 109–131 (1999).
10. D. Zeilberger: A fast algorithm for proving terminating hypergeometric identities. *Discrete Math.* **80**, 207–211 (1990).