Erdelyi Addendum PhL 1.20.04

These notes contain derivations and facts on orthogonal polynomials which Erdelyi did not provide. These things were used in Jim Ball's Gaussian Quadrature papers, but are really ortho poly facts.

Contents:

Theorem A1: (The X Determinant)
Theorem A2: (Angular Momentum Product Rule for Ortho Polys)

1. About the scaling freedom of the ortho polys.
2. How to get the ortho poly "A,B,C" recursion relation into "symmetric form" (via orthonormals)
3. How to get the "A,B,C" recursion relation into "Gautschi α',β' form" (via monics)
4. How to compute the ortho polys from the symmetric recursion relation coefficients.
5. How to compute the symmetric recursion relation coefficients *and* the ortho polys *at the same time*, given knowledge of the w(x) and (a,b). [This is the 1884 **Stieltjes** procedure]

6a. Compute the ortho poly recursion coefficients A,B,C as integrals of the polys.
6b. Repeat previous section for the "symmetric" form.
6c. Repeat previous section for the "Gautschi α',β' form".

On page 158 we find the following claim as part of (3):

Theorem A1: (The X Determinant) Consider this n+1 x n+1 matrix X,

$$\begin{matrix} x_0^0, x_0^1, ... x_0^n \\ x_1^0, x_1^1, ... x_1^n \\ ..... \\ x_n^0, x_n^1, ... x_n^n \end{matrix} = X$$

We claim that $\det(X) = \Pi_{r>s} (x_r - x_s)$.

Example:

$$\begin{vmatrix} 1 & x_0 & x_0^2 \\ 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \end{vmatrix} = (x_2 - x_1)(x_2 - x_0)(x_1 - x_0)$$

Proof:

Let's do the proof for n+1 = 5 (n = 4), and it will be obvious what the general case looks like. Here is our matrix in question:

$$\begin{matrix} x_0^0, x_0^1, x_0^2, x_0^3, x_0^4 \\ x_1^0, x_1^1, x_1^2, x_1^3, x_1^4 \end{matrix}$$

$x_2^0, x_2^1, x_2^2, x_2^3, x_2^4$
$x_3^0, x_3^1, x_3^2, x_3^3, x_3^4$
$x_4^0, x_4^1, x_4^2, x_4^3, x_4^4$

Our variables are $x_0, x_1.....x_4$. Let's think first of x_4 as our "variable". We know that detX is a polynomial in x_4 of degree 4 (just do the Laplace expansion along the bottom row). We know that such a poly can be written in the $\pi(x)$ form as a product of four factors $(x_4 - a_i)$, so we have:

$$detX = k(x_0, x_1, ..x_3)\ (x_4- a_1)(x_4- a_2)(x_4- a_3)(x_4- a_4)$$

where the roots are some a_1 through a_4, and where the leading constant is a function of the "other" variables. Now we know that detX = 0 if x_4 = any of the other x_i because then two rows are the same in the matrix. Thus, the roots a_i must be "the other x_i". So we now have

$$detX = k(x_0, x_1, ..x_3)\ (x_4- x_3)(x_4- x_2)(x_4- x_1)(x_4- x_0)$$

Now consider this object $k(x_0, x_1, ..x_3)$ and let us ponder x_3 as our variable now. We know that overall detX is a poly of degree 4 in x_3. We have exactly one factor showing already which is $(x_4- x_3)$, so it must be that $k(x_0, x_1, ..x_3)$ is a poly of degree 3 in x_3. In other words, we already have one of the four factors, so it must be that

$$k(x_0, x_1, ..x_3) = k'(x_0, x_1, ..x_2)(x_3- x_2)(x_3- x_1)(x_3- x_0)$$

where we have now anticipated that the three roots for the x_3 factors are the "other x_i", just as above. Next, we recur again to say that

$$k'(x_0, x_1, ..x_2) = k''(x_0, x_1)\ (x_2- x_1)(x_2- x_0)$$

by the same logic. In other words, for variable x_2 we already extracted 2 of the 4 roots in our previous 2 steps [namely, $(x_4- x_2)$ in the first step, and $(x_3- x_2)$ in the second step] so we then write down the remaining two roots as above, $(x_2- x_1)(x_2- x_0)$. Now we are close to the end with the next step:

$$k''(x_0, x_1) = k'''(x_0)\ (x_1- x_0)$$

And we now assemble our total answer:

$$detX = k'''(x_0)\ [\ (x_4- x_3)(x_4- x_2)(x_4- x_1)(x_4- x_0)\ ]\ [\ (x_3- x_2)(x_3- x_1)(x_3- x_0)\ ]\ [\ (x_2- x_1)(x_2- x_0)\ ]\ [(x_1- x_0)\ ]$$

Now if we write det(X) in the ε form we get this

$$det(X) = \varepsilon_{abcde}\ (x_a)^0(x_b)^1(x_c)^2(x_d)^3(x_e)^4 = (x_0)^0(x_1)^1(x_2)^2(x_3)^3(x_4)^4 + \text{other similar terms.}$$

The one term we have exposed is the product of the diagonal elements. We can identify this term in the above product of factors! The first four factors [$(x_4- x_3)(x_4- x_2)(x_4- x_1)(x_4- x_0)$] give us our $(x_4)^4$ and the next three [$(x_3- x_2)(x_3- x_1)(x_3- x_0)$] give us $(x_3)^3$, and so on. We know the coefficient of this one term must be 1, so we conclude that $k'''(x_0) = 1$. So our final answer is:

$$\det X = [\ (x_4 - x_3)(x_4 - x_2)(x_4 - x_1)(x_4 - x_0)\ ]\ [\ (x_3 - x_2)(x_3 - x_1)(x_3 - x_0)\ ]\ [\ (x_2 - x_1)(x_2 - x_0)\ ]\ [(x_1 - x_0)\ ]$$

$$= \Pi_{r>s}\,(x_r - x_s) \qquad \text{where } r = 4,3,2,1 \qquad \text{QED}$$

Comments: I don't think I have seen this theorem anywhere before. It is certainly not in any book I have (that I know of) except Erdelyi vol 2 page 158 top. It probably has a **name** associated with it, but I don't know how to find it because I don't know how to look this up on the web.

Corollary: As long as all the x_i are different, we have $\det X = 0$ so X is invertible.

Comment: There is a similar looking thing on page 53 of Scheid.

Theorem A2: (Angular Momentum Product Rule for Ortho Polys) If we expand $p_{j1}(x)p_{j2}(x)$as a lincom of ortho polys, we get a restricted range exactly as in the angular momentum rule of SO(3).

$$p_{j1}(x)p_{j2}(x) = \sum_{\mathbf{j=|j1-j2|}}^{\mathbf{j1+j2}} \mathbf{a_j}\ p_j(x)$$

Proof: We know that we can expand like so:

$$p_{j1}(x)p_{j2}(x) = \sum_{j=0}^{j1+j2} a_j\ p_j(x)$$

just because we know that the product is a poly of degree j1 + j2. Now close with p_k on both sides to get

$$(p_k,\ p_{j1}p_{j2}) = \sum_{j=0}^{j1+j2} a_j\ (p_k,p_j) = a_k h_k \qquad \text{// due to orthogonality}$$

Now assume that $j1 \geq j2$ and write

$$(p_k,\ p_{j1}p_{j2}) = (p_k p_{j2},\ p_{j1})$$

We know that this will be zero if $k+j2 < j1$ because if we expand $p_k p_{j2}$ in p_n, the highest will be $n = k + j2$. And if $k+j2 < j1$, then all of these are orthogonal to p_{j1}. Therefore, all $a_k = 0$ for $k < j1 - j2$. Thus, the actual sum of non-vanishing terms begins at $j = j1 - j2$.

We can then use this symbolic notation for a product.

$$n \otimes m = (n+m) + (n+m-1) + \ldots\ldots + (\ |\ n - m\ |\ )$$

where n is a shorthand for $p_n(x)$ and where the addition on the right ignores the coefficients a_i .

---

1. About the scaling freedom of the ortho polys.

Consider the Erdelyi recursion formula

$$p_{n+1} = (A_n x + B_n)p_n - C_n p_{n-1}$$

where we show three parameters (A_n,B_n,C_n). These are based on (k_n, k_n', h_n) as shown page 159. Now, suppose we have this set $\{ k_n, k_n', h_n, p_n(x) \}$ and we define $P_n(x) = s_n p_n(x)$ where s_n is an arbitrary scale factor distinct for each n. Then we have a new set $\{ K_n, K_n', H_n, P_n(x) \}$. But we know that

$$K_n = s_n k_n, \qquad K_n' = s_n k_n' \qquad H_n = s_n^2 h_n \qquad P_n(x) = s_n p_n(x)$$

just from the definitions of these quantities.

Now suppose we decide to scale things such that $H_n = 1$, so we are orthonormalized. We then have $H_n = 1 = s_n^2 h_n$ so that $s_n = 1/\sqrt{h_n}$. We then get

$$K_n = k_n/\sqrt{h_n}, \qquad K_n' = k_n'/\sqrt{h_n} \qquad H_n = 1 \qquad P_n(x) = p_n(x)/\sqrt{h_n}$$

At this point, we have to take what we get for K_n, $K_{n'}$ and P_n. We cannot "force" things to be monic, for example.

OK, suppose instead we scale to get monics. then $K_n = s_n k_n = 1$, so $s_n = 1/k_n$ and then we have

$$K_n = 1, \qquad K_n' = k_n'/k_n \qquad H_n = h_n/k_n^2 \qquad P_n(x) = p_n(x)/k_n$$

In this situation, we get what we get for H_n and we cannot control it. We cannot force these to be both monic and orthonormal, obviously.

---

2. Show how to get the ortho poly "A,B,C" recursion relation into "symmetric form"

The "genuine" recursion relation is this and it goes with $(p_n, p_n) = h_n$, (Erdelyi page 158)

$$p_{n+1} = (A_n x + B_n)p_n - C_n p_{n-1}$$

As we will show here, if you change the scale by doing $\gamma_n \phi_n(x) = p_n(x)$ such that the $\phi_n(x)$ are orthonormal, you arrive at what I have been calling "the symmetric form" of the recursion relation,

$$x \phi_n = \alpha_n \phi_{n+1} + \beta_n \phi_n + \alpha_{n-1} \phi_{n-1}$$

Rewrite this as

$$x p_n = (1/A_n)p_{n+1} - (B_n/A_n) p_n + (C_n/A_n) p_{n-1}$$

Define $\gamma_n \phi_n(x) = p_n(x)$. Then we have

$$\gamma_n x \phi_n = \gamma_{n+1} (1/A_n)\phi_{n+1} - \gamma_n (B_n/A_n) \phi_n + \gamma_{n-1} (C_n/A_n) \phi_{n-1}$$

Divide by γ_n to get

$$x\ \phi_n = (\gamma_{n+1}/\gamma_n)\,(1/A_n)\phi_{n+1} \ - \ (B_n/A_n)\,\phi_n \ + (\gamma_{n-1}/\gamma_n)\,(C_n/A_n)\ \phi_{n-1}$$

Now define $\alpha_n = (\gamma_{n+1}/\gamma_n)\,(1/A_n)$. Then we want to have

$$\alpha_{n-1} = \ (\gamma_{n-1}/\gamma_n)\,(C_n/A_n)$$

which says

$$\alpha_n = (\gamma_n/\gamma_{n+1})\ (C_{n+1}/A_{n+1})$$

So we then require that

$$(\gamma_{n+1}/\gamma_n)\,(1/A_n) \ = \ (\gamma_n/\gamma_{n+1})\ (C_{n+1}/A_{n+1})$$

which says that

$$(\gamma_{n+1}/\gamma_n)^2 = \ C_{n+1}\ (A_n/A_{n+1}) = (h_{n+1}/h_n)$$

$$\gamma_{n+1} \ = \gamma_n \sqrt{(h_{n+1}/h_n)}$$

We can now iterate to get

$$\begin{aligned}
\gamma_1 &= \gamma_0 \sqrt{(h_1/h_0)} && = \gamma_0 \sqrt{(h_1/h_0)} \\
\gamma_2 &= \gamma_0 \sqrt{(h_1/h_0)}\,\sqrt{(h_2/h_1)} && = \gamma_0 \sqrt{(h_2/h_0)} \\
\gamma_3 &= \gamma_0 \sqrt{(h_1/h_0)}\,\sqrt{(h_2/h_1)}\,\sqrt{(h_3/h_2)} && = \gamma_0 \sqrt{(h_3/h_0)} \\
&\ldots\ldots \\
\gamma_n &= \gamma_0 \sqrt{(h_n/h_0)}
\end{aligned}$$

$$(\phi_n, \phi_m) = (1/\gamma_n)^2(p_n, p_m) = (1/\gamma_n)^2\ h_n\delta_{n,m} = h_0\ (1/\gamma_0)^2\ \delta_{n,m}$$

Now, if we are handed the A_n, B_n, C_n we can think of that as determining k_n, k_n' and h_n, so we cannot willy nilly go adjusting h_0 . However, there is no reason not to choose γ_0 such that $\gamma_0 = \sqrt{h_0}$ and then we end up with a nice orthonormal set of ortho polys.

Meanwhile, we can look back at

$$\alpha_n = (\gamma_{n+1}/\gamma_n)\,(1/A_n) \ = \sqrt{(h_{n+1}/h_n)}\ (1/A_n) \ = \sqrt{(h_{n+1}/h_n)}\ \ (k_n/k_{n+1})$$

$$= \sqrt{C_{n+1}\ (A_n/A_{n+1})}\ (1/A_n) \ = \sqrt{C_{n+1}/[A_nA_{n+1}]}$$

So we can now summarize all the results:

Start:	$p_{n+1} = (A_n x + B_n)p_n - C_n p_{n-1}$ // Erdelyi notation	
Define:	$\phi_n(x) = p_n(x)/\gamma_n$	
Have:	$x\,\phi_n = (\gamma_{n+1}/\gamma_n)\,(1/A_n)\phi_{n+1} - (B_n/A_n)\,\phi_n + (\gamma_{n-1}/\gamma_n)\,(C_n/A_n)\,\phi_{n-1}$	
Require:	$x\,\phi_n = \alpha_n\,\phi_{n+1} - (B_n/A_n)\,\phi_n + \alpha_{n-1}\,\phi_{n-1}$	
Solve:	$\gamma_{n+1} = \gamma_n \sqrt{C_{n+1}\,(A_n/A_{n+1})} = \gamma_n \sqrt{(h_{n+1}/h_n)}$	
Iterate:	$\gamma_n{}^2 = (A_0/A_n)C_1C_2....C_n \Rightarrow \gamma_n = \gamma_0 \sqrt{(h_n/h_0)}$	
Find:	$(\phi_n, \phi_m) = h_0\,(1/\gamma_0)^2\,\delta_{n,m}$	
Find:	$\alpha_n = \sqrt{C_{n+1}/[A_nA_{n+1}]} = \sqrt{(h_{n+1}/h_n)}\;(k_n/k_{n+1})$	
Define:	$\beta_n = -(B_n/A_n) = (r_n - r_{n+1}) = (k_n'/k_n - k_{n+1}'/k_{n+1})$	$A_n = k_{n+1}/k_n$
End up:	$x\,\phi_n = \alpha_n\phi_{n+1} + \beta_n\,\phi_n + \alpha_{n-1}\,\phi_{n-1}$	$B_n = A_n(r_{n+1} - r_n)$
Select:	$\gamma_0{}^2 = h_0 \Rightarrow (\phi_n, \phi_m) = \delta_{n,m}$ orthonormal	$C_n = (A_n/A_{n-1})(h_n/h_{n-1})$

The conclusion is that there is a simple way to define the recursion coefficients to get the symmetric Jacobi matrix *and* orthonormal polys. We have shown in the box above how you do this if you are given either (A_n,B_n,C_n) or (h_n, k_n, k_n') as your independent variables.

Wilf states this in another way. Suppose we selected $p_n(x)$ with $h_n = 1$ from the get-go, which means that our p_n are then orthonormal. Then in the above we would get $\gamma_n = 1$ and $\phi_n = p_n$ and we would at once have the symmetric form with $\alpha_n = (k_n/k_{n+1}) = 1/A_n$ and $\beta_n = (r_n - r_{n+1})$. So he regards the symmetric form just as a characteristic of having orthonormal p_n .

---

3. Show how to get the A,B,C recursion relation into Gautschi α',β' form (via monics)

Our starting point as before is this

$$x\,p_n = (1/A_n)p_{n+1} - (B_n/A_n)\,p_n + (C_n/A_n)\,p_{n-1}$$

where $A_n = k_{n+1}/k_n$ and $B_n = A_n\,(r_{n+1} - r_n)$ and $C_n = (A_n/A_{n-1})(h_n/h_{n-1})$ so we regard (A_n,B_n,C_n) as functions of (h_n, k_n, k_n'). Since $p_{-1} = 0$, we don't care about the number C_0. We can then just set C_0 = any value we like, perhaps 0 or perhaps 1. So we only use the formula $C_n = (A_n/A_{n-1})(h_n/h_{n-1})$ for n = 1,2,3... whereas we use the other two formulas for A_n and B_n we use for n = 0,1,2...

Now, with *monics* we have $k_n = 1$ so that $A_n = 1$ and then $C_n = (h_n/h_{n-1})$ and $B_n = (k'_{n+1} - k'_n)$. Note that by requiring all monics, we have to accept h_n and k_n' being whatever they are, we cannot "set them". For monics, then, we get a simplified recursion formula

$x\,p_n = p_{n+1} - B_n\,p_n + C_n\,p_{n-1}$	p_n = monic
$x\,p_n = p_{n+1} + \alpha_n'\,p_n + \beta_n'\,p_{n-1}$	p_n = monic

where we select the names α_n' and β_n' in order to match the expression (1.7) of Gautschi (except we have primes and he does not), so we have

$$\alpha_n' = -B_n = (r_n - r_{n+1}) = k_n' - k_{n+1}'$$
$$\beta_n' = C_n = (h_n/h_{n-1})$$

Now suppose we apply the following scaling transformation

$$p_n(x) = q_n f_n(x)$$

then the above becomes

$$x\, q_n f_n = q_{n+1} f_{n+1} + \alpha_n' q_n f_n + \beta_n' q_{n-1} f_{n-1}$$

$$x\, f_n = (q_{n+1}/q_n) f_{n+1} + \alpha_n' f_n + \beta_n' (q_{n-1}/q_n) f_{n-1}$$

Now select

$$(q_{n+1}/q_n) = \sqrt{\beta_{n+1}'} \quad => \quad (q_n/q_{n-1}) = \sqrt{\beta_n'}$$

to get

$$x\, f_n = \sqrt{\beta_{n+1}'}\, f_{n+1} + \alpha_n' f_n + \sqrt{\beta_n'}\, f_{n-1}$$

The scaling factor q_n can be determined by iterating this expression,

$$(q_n/q_{n-1}) = \sqrt{\beta_n'} = C_n = \sqrt{h_n/h_{n-1}} \qquad n = 1,2,3... \text{ as noted earlier}$$

Only the q ratio shows up in our work above, so we can have q_0 be an arbitrary constant. We then have

$$q_1^2 = q_0^2 (h_1/h_0)$$
$$q_2^2 = q_1^2 (h_2/h_1) = q_0^2 (h_1/h_0)(h_2/h_1) = q_0^2 (h_2/h_0)$$
$$q_3^2 = q_2^2 (h_3/h_2) = q_0^2 (h_2/h_0) (h_3/h_2) = q_0^2 (h_3/h_0)$$
$$.....$$
$$q_n^2 = q_0^2 (h_n/h_0) \quad => \quad (h_n/q_n^2) = (h_0/q_0^2)$$

The normalization for our monic p_n is $(p_n, p_m) = h_n \delta_{n.m}$. The scaled functions f_n have this normalization

$$(f_n, f_m) = (1/[q_n q_m]) (p_n, p_m) = \delta_{n,m} (h_n/q_n^2) = (h_0/q_0^2) \delta_{n,m}$$

How just for reference, we have $p_0(x) = 1$ since monic, so

$h_0 = (p_0, p_0) = (1,1) =$ a number determined by w(x) and (a,b).

However, as noted above, q_0 was an arbitrary number, so we can set $q_0^2 = h_0$ to get

$$(f_n, f_m) = \delta_{n,m}$$

But the orthonormalized ortho poly set is unique, so it seems to me that we must have $f_n = \phi_n$ of our previous section, up to a possible sign. That is, we could have $f_3 = -\phi_3$ and $f_2 = +\phi_2$ and things still work out OK (assuming everything real here). Let's assume for the moment no sign issues. Then we have

$$x\,\phi_n = \sqrt{\beta_{n+1}'}\,\phi_{n+1} + \alpha_n'\,\phi_n + \sqrt{\beta_n'}\,\phi_{n-1}$$

$$x\,\phi_n = \alpha_n\,\phi_{n+1} + \beta_n\,\phi_n + \alpha_{n-1}\,\phi_{n-1}$$

We can then identify

$$\sqrt{\beta_{n+1}'} = \alpha_n \qquad \Rightarrow \qquad \sqrt{\beta_n'} = \alpha_{n-1}$$

$$\alpha_n' = \beta_n$$

$$\sqrt{\beta_n'} = \alpha_{n-1}$$

So everything is consistent. All we have here is a stupid "name change". There is nothing special about the square roots appearing. The interesting fact is that when you deal with monics, it is convenient to use the α_n', β_n' type coefficients, and when the α_n β_n are expressed in terms of these, we get square roots.

Comment: The monic form appears in Gautschi as $p_{n+1} = (x-\alpha_n')p_n - \beta_n' p_{n-1}$. We can compare this to Ball/Beebe (27) to see that they use $b_n = \alpha_n'$ and $a_n = \beta_n'$. Thus they would get $x\,\phi_n = \sqrt{a_{n+1}}\,\phi_{n+1} + b_n\,\phi_n + \sqrt{a_n}\,\phi_{n-1}$ with ϕ_n = their M_n. This then explains the comment below eq (31) where I was wondering why the square roots appeared.

---

4. Show how to compute the ortho polys from the recursion relation coefficients.

Let's start with this simple form

$$x\,\phi_n = \alpha_n\phi_{n+1} + \beta_n\,\phi_n + \alpha_{n-1}\,\phi_{n-1}$$

or

$$n = n \qquad \phi_{n+1} = (x - \beta_n)(1/\alpha_n)\phi_n - (\alpha_{n-1}/\alpha_n)\,\phi_{n-1}$$

We know that ϕ_0 = constant, and we know $\phi_{-1} = 0$. So away we go:

$$n = 0 \qquad \phi_1 = (x - \beta_0)(1/\alpha_0)\phi_0$$

$$n = 1 \qquad \phi_2 = (x - \beta_1)(1/\alpha_1)\phi_1 - (\alpha_0/\alpha_1)\,\phi_0 = [(x - \beta_1)(1/\alpha_1)(x - \beta_0)(1/\alpha_0)\phi_0 - (\alpha_0/\alpha_1)\,\phi_0]$$

$$n = 2 \qquad \phi_3 = (x - \beta_2)(1/\alpha_2)\phi_2 - (\alpha_1/\alpha_2)\,\phi_1$$

$$= (x - \beta_2)(1/\alpha_2)[(x - \beta_1)(1/\alpha_1)(x - \beta_0)(1/\alpha_0)\phi_0 - (\alpha_0/\alpha_1)\,\phi_0] - (\alpha_1/\alpha_2)\,[(x - \beta_0)(1/\alpha_0)\phi_0]$$

$$= (x - \beta_2)(1/\alpha_2)(x - \beta_1)(1/\alpha_1)(x - \beta_0)(1/\alpha_0)\phi_0 - (x - \beta_2)(1/\alpha_2)(\alpha_0/\alpha_1)\,\phi_0 - (\alpha_1/\alpha_2)\,[(x - \beta_0)(1/\alpha_0)\phi_0]$$

We can see that ϕ_3 is cubic and so on, so we can see for sure how this works. Getting a closed form for the result may not be so easy. In practice, you would just compute them one at a time using the recursion and be happy with that. You can make a program compute the first 100 no problem.

---

5. Show how to compute the recursion relation coefficients and the ortho polys *at the same time*, given knowledge of the w(x) and (a,b). [This is the 1884 Stieltjes method]

We know from above that we can get things into this form:

$$x\,\phi_n = \alpha_n\phi_{n+1} + \beta_n\,\phi_n + \alpha_{n-1}\,\phi_{n-1}$$
$$(\phi_n, \phi_m) = \delta_{n,m}$$

where we can rewrite the first as

$$\phi_{n+1} = (x - \beta_n)(1/\alpha_n)\phi_n - (\alpha_{n-1}/\alpha_n)\,\phi_{n-1}$$

First, close with ($x\phi_n$ to get

$$(x\phi_n, x\phi_n) = \alpha_n^2 + \beta_n^2 + \alpha_{n-1}^2 \qquad \text{// This is Jim (3.13)}$$

which we can write as

$$\alpha_n^2 = (\phi_n, x^2\phi_n) - \beta_n^2 - \alpha_{n-1}^2$$

Next, close with (ϕ_n to get

$$\beta_n = (\phi_n, x\phi_n) \qquad \text{// This is Jim (3.14)}$$

Here then is our process for computing everything at once. To get started, we always say $\phi_{-1} = 0$ for an ortho poly system of any kind. In this case, the α_{-1} in the recursion formula is a "don't care" so the simplest thing is to set $\alpha_{-1} = 0$. It has no effect on anything. So

$$\phi_{-1} = 0$$
$$\alpha_{-1} = 0$$

Next, ϕ_0 = constant, and we will just refer to this constant as ϕ_0 rather than make up a new symbol. We then have

$$\phi_0 = \phi_0$$
$$\beta_0 = (\phi_0, x\phi_0) = \phi_0^2\,(1,x)$$
$$\alpha_0 = (\phi_0, x^2\phi_0) - \beta_0^2 - \alpha_{-1}^2 = \phi_0^2\,(1,x^2) - \phi_0^2\,(1,x^2)$$

Now we move up to the 1 level:

$\phi_1(x) = (x - \beta_0)(1/\alpha_0)\phi_0$
$\beta_1 = (\phi_1, x\phi_1)$
$\alpha_1^2 = (\phi_1, x^2\phi_1) - \beta_1^2 - \alpha_0^2$

And on up to the 2 level:

$\phi_2(x) = (x - \beta_1)(1/\alpha_1)\phi_1 - (\alpha_0/\alpha_1)\phi_0$
$\beta_2 = (\phi_2, x\phi_2)$
$\alpha_2^2 = (\phi_2, x^2\phi_2) - \beta_2^2 - \alpha_1^2$

We can then continue this procedure up to any N. This method is called "The Stieltjes Procedure" and was first developed in 1884.

6a. Compute the ortho poly recursion coefficients A,B,C as integrals of the polys.

Here is our recursion formula

$p_{n+1} = (A_n x + B_n)\, p_n - C_n p_{n-1}$

Consider:

$(p_n, p_{n+1}) = A_n(p_n, xp_n) + B_n\,(p_n, p_n) - C_n(p_n, p_{n-1})$

From Lemma S3 of my Erdelyi notes we know that

$(p_n, xp_n) = k_n[\, r_n - r_{n+1}](p_n, x^n) = (\, r_n - r_{n+1})\, h_n = -\,(B_n/A_n)h_n.$

Thus we have

$0 = A_n * (B_n/A_n)h_n + B_n h_n = -\,B_n h_n + B_n h_n$

and we have learned nothing at all! Hmmm. Well I guess you regard (xp_n, p_n) as an integral you can do knowing p_n and w(x), then you can say

$-\,(B_n/A_n)h_n = (xp_n, p_n) = \int w(x)dx\; x\,[p_n(x)]^2$ // compare to Jim's (3.13)

Now let's try another p_k :

$(p_{n+1}, p_{n+1}) = A_n(p_{n+1}, xp_n) + B_n\,(p_{n+1}, p_n) - C_n(p_{n+1}, p_{n-1})$

$h_{n+1} = A_n\,(p_{n+1}, xp_n)$

So this tells us that

$$h_{n+1}(1/A_n) = (p_{n+1}, xp_n)$$

And another p_k:

$$(p_{n-1}, p_{n+1}) = A_n(p_{n-1}, xp_n) + B_n (p_{n-1}, p_n) - C_n(p_{n-1},p_{n-1})$$

$$0 = A_n(p_{n-1}, xp_n) - C_n h_{n-1}$$

Then we get

$$h_{n-1} (C_n/A_n) = (p_{n-1}, xp_n)$$

There is one other thing we can do of interest which is this:

$$p_{n+1} = (A_n x+B_n) p_n - C_n p_{n-1}$$

$$A_n x p_n = p_{n+1} - B_n p_n + C_n p_{n+1}$$

Then square both sides and integrate to get

$$A_n^2 (xp_n,xp_n) = h_{n+1} + B_n^2 h_n + C_n^2 h_{n+1}$$

so that

$$h_{n+1} (1/A_n)^2 + h_n (B_n/A_n)^2 + h_{n+1}(C_n/A_n)^2 = \int w(x)dx\, x^2 [p_n(x)]^2 \qquad \text{// compare to Jim's (3.14)}$$

So we end up with the following interesting set of relations:

$$\begin{aligned}
h_{n+1}(1/A_n) &= (p_{n+1}, xp_n) \\
h_{n-1} (C_n/A_n) &= (p_{n-1}, xp_n) \\
h_n (B_n/A_n) &= -(p_n, xp_n) \\
h_{n+1} (1/A_n)^2 + h_n (B_n/A_n)^2 + h_{n+1}(C_n/A_n)^2 &= (xp_n,xp_n)
\end{aligned}$$

6b. Repeat previous section for the "early Jim symmetrized recursion relation" polys.

We know from notes elsewhere that we can get things into this form:

$$x \phi_n = \alpha_n \phi_{n+1} + \beta_n \phi_n + \alpha_{n-1} \phi_{n-1}$$
$$(\phi_n, \phi_m) = \delta_{n,m}$$

First, close with $(x\phi_n$ to get

$(x\phi_n, x\phi_n) = \alpha_n^2 + \beta_n^2 + \alpha_{n-1}^2$ // This is Jim (3.13)

Next, close with (ϕ_n to get

$(\phi_n, x\phi_n) = \beta_n$ // This is Jim (3.14)

The second integral gives β_n, then we iterate the previous result to find the α_n . Therefore, this shows that we can "compute the recursion coefficients" if we know the orthonormalized polynomials ϕ_n and the weight function w(x) and the interval (a,b)

$\beta_n = \int dx\ w(x)x\ [\phi_n(x)]^2$

$\alpha_n^2 = \int dx\ w(x)x^2\ [\phi_n(x)]^2 - \beta_n^2 - \alpha_{n-1}^2$

6c. Repeat previous section for the monic normalized polys.

We know from notes elsewhere that we can get things into this form for monics,

$x\ p_n = p_{n+1} + \alpha_n'\ p_n + \beta_n'\ p_{n-1}$
$(p_n, p_n) = h_n$

First, close with (p_n to get

$(p_n, x\ p_n) = \alpha_n'\ h_n$

Then close with (p_{n-1} to get

$(p_{n-1}, x\ p_n) = \beta_n'\ h_{n-1}$

Now we can manipulate the LHS as follows, due to monic-ness

$(p_{n-1}, x\ p_n) = (x\ p_{n-1}, p_n) = (x^n + \text{lower}, p_n) = (x^n, p_n) = (p_n + \text{lower}, p_n) = (p_n, p_n) = h_n$

causing this result:

$\beta_n' = (h_n/h_{n-1})$ // this agrees with what we find below in section ****
$\alpha_n' = (p_n, x\ p_n)/h_n$ // this is the new result here.

So we can write this as follows:

$h_n \equiv (p_n, p_n)$ $\beta_n' = (h_n/h_{n-1})$
$s_n \equiv (p_n, x\ p_n)$ $\alpha_n' = (s_n/h_n)$

In Ball/Beebe this appears as

$t_n \equiv (M_n, M_n)$ $\quad\quad$ $a_n = (t_n/t_{n-1})$

$s_n \equiv (M_n, x\ M_n)$ $\quad\quad$ $b_n = (s_n/t_n)$

where they use $a_n = \beta_n'$ and $b_n = \alpha_n'$ as indicated in their equation (27). Also, we know that $p_n = \sqrt{h_n}\ \phi_n$ since the ϕ_n are orthonormal. Thus, B&B say $M_n = \sqrt{t_n}\ \Lambda_n$ top of page 9.