Recurrence Relations for Orthogonal Polynomials and Efficient Algorithms PhL 10.17.04
Jim Ball

I had an earlier version of this paper dated 2/98, and I don't see it listed on-line anywhere. It may never have been published. It is not in the SIAM search.

Section 2. The weighted ortho of polys is stated in 2.1, the recurrence relations (normalized as in previous paper) in 2.2, where g(x) = x for ALL A&S ortho polys, and we end up with 2.3 which is the same equation as in the previous paper. But this time the matrix is called "the Jacobi matrix" and the main point that the zeros of a ϕ are the eigenvalues is credited at least to Wilf 1962. [In the previous paper, you might get the idea that Jim discovered this concept himself.] So I am fine with page 2.

In 2.4 Jim defines V_n^i as shown with I think the idea that we are going to replace the continuous spatial variable x with the discrete label i which labels x_i . He notes two new facts here. First, since the zeros are in fact nicely spaced, that TQL2 type algorithm works well to do the matrix work. Second, he comments that the work is N^2 in time, which I can certainly believe. In passing, he briefly mentions the $P_L(x)$ as an example and gives the α and β. In the other paper, he did the associated Legendres, so this is a special case of that.

Section 3. Jim's language here leaves me confused, but I have my own pretty good language for this kind of stuff.

Consider the **X** operator and write: $\mathbf{X}\ |x\rangle = x\ |x\rangle$

where I am using QM notation, and |x> is a state localized at position x, and x is a scalar. Note that we would say that |x> is an eigenstate of the operator **X**. Now imagine a (different) complete set of basis states |m> and define $\langle x|m\rangle = \phi_m(x)$ as the basis functions in the x representation. Then we know that $1 = |m\rangle\langle m|$ (with implied summation) and we then have

$$\langle n|\mathbf{X}|m\rangle\langle m|x\rangle = x\langle n|x\rangle \qquad \text{or} \qquad X_{nm}\phi_m(x) = x\ \phi_n(x)$$

This last equation says that x is an eigenvalue of the matrix X with eigenvector $\phi_m(x)$. In the case that x is one of the zeros of $\phi_{N+!}(x)$, we have an equation $J_{nm}\phi_m(x_i) = x_i\ \phi_m(x_i)$ from (2.3), where we have N+1 as the dimension of our space, and where J is the Jacobi matrix. Doing a comparison suggests that

$$J_{nm} = \text{Jacobi matrix} = X_{nm} = \langle n|\mathbf{X}|m\rangle = \langle n|x\rangle\langle x|\mathbf{X}|x'\rangle\langle x'|m\rangle$$

where $\langle x|\mathbf{X}|x'\rangle = x\delta(x-x')$, so we of course have $J_{nm} = \int dx\ \phi_n(x)\ x\ \phi_m(x)$.

The point here is that I agree with Jim's claim that the Jacobi matrix is the same as the A_{nm} matrix he shows in (3.1) [which is my X_{nm} matrix above]. This is all true because g(x) = x in those recursion relations! The Jacobi matrix can thus be interpreted as the matrix elements of the X operator in a basis that is the ortho polynomials for interval (a,b) with weight w(x). If we are talking about some N, then the space is only complete of course for functions that are up to an Nth degree polynomial, but we are going to use this idea to approximate for a general function f(x).

As I say, Jim's text on pages 3 and 4 is very hazy IMHO. I think my method has the potential to make it all crystal clear, as opposed to hazy, but we shall carry on. The conclusion above 3.2 is that the 3.1

integral shown is the same as the Jacobi matrix. Yes, we have a space of size N+1 and it is complete only for certain functions, and all that stuff, so you can use the word "truncation" to finite N.

Well, maybe more talk is needed here. The space |x> is infinite, but suppose we consider a finite number of x values x_i and then we talk about a basic state $|x_i>$. Completeness is now a sum over these states. But when we go discrete, we might have normalization problems.

I think with w(x) floating around, you need to say this:

$$\mathbf{1} = \int dx\, w(x)\, |x><x| \qquad \text{// completeness}$$

Then if you close this with <m| and |n>, you will get

$$<m|n> = \delta_{m,n} = \int dx\, w(x) <m|x><x|n> = \int dx\, w(x)\phi_m(x)\phi_n(x)$$

which is then the correct orthonormality condition.

What is the corresponding stuff in the discrete-x space? First, consider this fact:

$$<x_i \mid x_i> = \Sigma_n <x_i|n><n|x_i> = \Sigma_n \phi_n(x_i)^2 = S_i$$

If S_i is not 1, then these space states are not properly normalized! We then need to define a rescaled state like so:

$$|i> = (1/\sqrt{S_i})|x_i> \qquad => \quad <i|i> = 1$$

In this properly normalized world, completeness looks like this:

$$\mathbf{1} = \Sigma_i\, |i><i| = \Sigma_i\, (1/S_i)\, |x_i><x_i|$$

which we compare to the form shown earlier

$$\mathbf{1} = \int dx\, w(x)\, |x><x| \qquad \text{// completeness}$$

and we conclude that in the discrete space basis x_i the weight function is $1/S_i$.

Jim's notation is then this:

$$<i|n> = v_n^i = (1/\sqrt{S_i})<x_i|n> = (1/\sqrt{S_i})\phi_n(x_i) = (1/\sqrt{S_i})V_n^i$$

OK, I am now happy with all equations through (3.6), this last showing that you can expand a power on the complete set of polys. I think the fact Jim really wants is this:

$$f(x) = \mathbf{\Sigma}_{m,n} F_{mn}\, \phi_n(x)\phi_m(x) \qquad \text{// for any poly f(x) up to order 2N}$$

Now if you insert this into the left side of (3.7), the integral creates $\delta_{n,m}$ and f(x) = tr(F). Now suppose you write

$$f(x_i) = \Sigma_{m,n} F_{mn} \phi_n(x_i)\phi_m(x_i)$$

If you insert this into the RHS of (3.7), you get

$$\Sigma_i \ (1/S_i) \ \Sigma_{m,n} F_{mn} \phi_n(x_i)\phi_m(x_i) = \Sigma_{m,n} F_{mn} \ \Sigma_i \ (1/S_i) \ \phi_n(x_i)\phi_m(x_i)$$

$$= \Sigma_{m,n} F_{mn} \ \Sigma_i \ (1/S_i) \ \langle n|x_i\rangle\langle x_i|m\rangle \ = \Sigma_{m,n} F_{mn} \ \langle n|m\rangle = tr(F)$$

So this shows that (3.7) is exactly correct for f(x) = any poly of order up to 2N. **BUT**, (3.7) is not just any old boring equation, it is exactly the form we talk about when doing Gaussian quadrature. We already know that x_i are the zeros of $\phi_{N+1}(x)$. Now we can identify the "weights" with $1/S_i$, and these are just the normalization factors for the ortho polys (scaled to give the desired recursion relations.

In 3.8 Jim expands a poly of degree N onto the basis functions $\phi_n(x)$ and reverse solves for the coefficients in 3.9. I agree with all these equations (after I have made fixes), but don't see why I should care about these last three equations.

Well let's go back to 3.7 again. This is Gaussian Quadrature and we need the nodes x_i and we need to evaluate the function f at the notes, and we need the weights $1/S_i$. We have already seen how we get the nodes from the Jacobi matrix based on the recursion relations. We have already seen how we get the weights, they are 1 over the norm factors of the ortho polys. Finally, we wonder how we compute the functions f(x) at the nodes? One way is shown in 3.10 in the unusual case that we happen to be given the coefficients a_n for f_n expanded as shown in 3.8.

Section 4. Here I skip the intro and go down at once to the example. I agree with (4.2) from memory. The confusion is now that Jim has redefined his A coefficients in going from 4.2 to 4.3, so I will just put primed on the ones in 4.2. In 4.3 we don't have the B stuff because we are expanding on the inside of a sphere, all Jackson stuff. So OK to 4.3 and with 4.4. I derive (4.5) from otho of <L|L'> and completeness in the i-space. The point here is I think this: you have a complete solution to your problem if you can find the set of A_L coefficients. In 4.5, you can compute these from the boundary value on the sphere of Φ (that is to say, r = 1 for first argument), but you only need look at some discrete values of θ_i corresponding tht the x_i. ($x = \cos\theta$). So your boundary values are a set of N+1 numbers, and then you can compute your A_L. Probably more generally (4.5) requires an integral over x, so you can just think of this as doing GQ for that integral and replacing it with a sum. I have no idea why Jim is so interested in this kind of problem, maybe it relates to lattice gauge theory somehow.

Section 5. Here he repeats the trick of the earlier paper to cause computation to be more efficient.

Conclusions: not too exciting. This entire paper seems a bit odd to me. Is Jim's method of getting the weights a new discovery, or is it old news? In the usual GQ theory, we do have formulas for the weights A_n that relate to the polynomials. You have to evaluate the polys at the nodes in either method. I wonder if this paper was rejected because it does not really seem to go anywhere, and the motivation is weak. I hope to learn more about this motivation in the other papers.]

Note added: let's try again on deriving (3.7).

Let,

$$f(x) = \mathbf{\Sigma}_{m,n} F_{mn} \phi_n(x)\phi_m(x)$$

If you evaluate at the x_i and insert this into the RHS of (3.7), you get

$$\Sigma_i \ (1/S_i) \ \mathbf{\Sigma}_{m,n} F_{mn} \phi_n(x_i)\phi_m(x_i) = \mathbf{\Sigma}_{m,n} F_{mn} \ \Sigma_i \ (1/S_i) \phi_n(x_i)\phi_m(x_i)$$

$$= \mathbf{\Sigma}_{m,n} F_{mn} \ \Sigma_i \ (1/S_i) \langle n|x_i\rangle\langle x_i|m\rangle \ = \mathbf{\Sigma}_{m,n} F_{mn} \ \langle n|m\rangle = tr(F)$$

If you insert this into the LHS of (3.7), you get

$$\int dx\ w(x) \mathbf{\Sigma}_{m,n} F_{mn} \phi_n(x)\phi_m(x) \ = \mathbf{\Sigma}_{m,n} F_{mn} \left[\int dx\ w(x) \ \phi_n(x)\phi_m(x) \right]$$

$$= \mathbf{\Sigma}_{m,n} F_{mn} \delta_{m,n} = tr(F)$$

So the GQ formula is exact for polys of degree 2N.