Elliptic cross sections in blood flow regulation

Robert M. Corless

joint work with Chris Brimacombe and Mair Zamir

January 16 2023

Western University, Canada

This talk dedicated to the memory of R. Bruce Simpson (1940–2020)

Maple Transactions

Announcing Maple Transactions

a new open access journal with no page charges

mapletransactions.org

What this talk is about

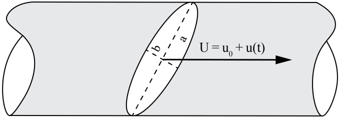


Figure 1: Pulsatile flow of blood in a vessel deformed to elliptic cross section

[Well, it's actually about my Maple code to evaluate Mathieu functions and its support infrastructure, but that hemodynamic problem is the motivating example.]

All models are wrong, but some are useful—George Box

$$\frac{\partial u_{\phi,e}}{\partial t} + \frac{1}{\rho}\frac{\partial p}{\partial z} = \frac{\mu}{\rho}\left(\frac{\partial^2 u_{\phi,e}}{\partial x^2} + \frac{\partial^2 u_{\phi,e}}{\partial y^2}\right) \tag{1}$$

We take $\rho = 1$ for blood, and $\mu = 0.04$. MZ tells me that people argue and argue about those things (especially μ) but for us it's just fine.

We will impose an oscillating pressure gradient

$$\frac{\partial p}{\partial z} = \frac{k_0}{\mu} e^{i\omega t}\ .$$

The pulsatile flow response will be $u_{\phi,e}(x,y) = w(x,y)e^{i\omega t}$.

The equation for $w(x,y)$

$$i\omega w(x,y) + \frac{k_0}{\mu\rho} = \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} \tag{2}$$

which is already looking nicer, and is even nicer yet with $w(x,y) = (v(x,y) - 1)k_0/(i\omega\mu\rho)$ but now the *elliptic* geometry comes into play. We use Lamé's "thermometric coordinates" to change to a confocal elliptic coordinate system, with foci of all ellipses at $(\pm d, 0)$:

$$\begin{aligned} x &= d\cosh(\xi)\cos(\eta) \\ y &= d\sinh(\xi)\sin(\eta)\,. \end{aligned} \tag{3}$$

Confocal elliptic coordinates

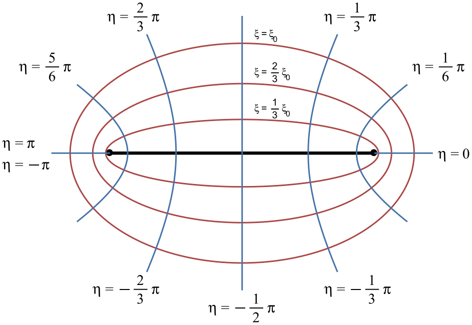


Figure 2: Thermometric coordinates: better than polar coordinates?

Separation of variables

"In order to maintain continuity with previous analyses" we change to elliptic coordinates and separate variables. The Laplacian in these confocal elliptic coordinates is

$$\frac{2}{d^2(\cosh 2\xi - \cos 2\eta)} \left(\frac{\partial^2}{\partial \xi^2} + \frac{\partial^2}{\partial \eta^2} \right) . \tag{4}$$

"The proof is left as an exercise for the student."

(Hoo boy.)

We have separation

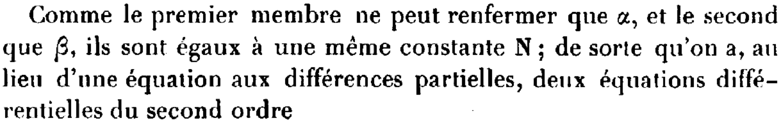
Frankly it's astonishing that $v(\xi,\eta) = f(\xi)g(\eta)$ causes the PDE to separate:

$$\frac{i\omega d^2}{2}\cosh(2\xi)f(\xi)g(\eta) - \frac{i\omega d^2}{2}\cos(2\eta)f(\xi)g(\eta) = f''(\xi)g(\eta) + f(\xi)g''(\eta) \tag{5}$$

or, putting $q = i\omega d^2/4$,

$$-\frac{f''(\xi)}{f(\xi)} + 2q\cosh 2\xi = \frac{g''(\eta)}{g(\eta)} + 2q\cos 2\eta \tag{6}$$

which, quoting Mathieu from his 1868 paper,



Comme le premier membre ne peut renfermer que α, et le second que β, ils sont égaux à une même constante N ; de sorte qu'on a, au lieu d'une équation aux différences partielles, deux équations différentielles du second ordre

Figure 3: Separation of variables: same as it ever was

Arriving at the Mathieu equation

By convention we call the separation constant $-a$. We get two[1] equations, then:

$$f''(\xi) - (a - 2q\cosh 2\xi)f(\xi) = 0 \tag{7}$$

$$g''(\eta) + (a - 2q\cos 2\eta)g(\eta) = 0\,. \tag{8}$$

Only certain values of a will make $g(\eta)$ periodic with period 2π. Some of them will make $g(\eta)$ be *even*, and of period π. This will be a classical eigenfunction expansion, then. [Or will it?] At least, they tend to be *spectrally fast* as methods of approximation.

[1] They're really the same: $\eta = i\xi$.

The Mathieu equation

$$\frac{d^2y}{dz^2} + (a - 2q\cos 2z)y = 0 \tag{9}$$

The initial conditions we need are $y(0) = 1$ with $y'(0) = 0$, which gives $w_I(z)$ in Abramowitz and Stegun, called `MathieuC` in Maple; and $y(0) = 0$ with $y'(0) = 1$, which gives $w_{II}(z)$, called `MathieuS` in Maple.

If $0 \leq z \leq 2\pi$ and a is such that $y(z)$ is periodic then $y(z)$ is a "Mathieu function". In this case a is an eigenvalue for the parameter q.

Pure imaginary z gives a so-called "modified" Mathieu function (which we also need).

The parameter q depends on the situation being modelled.

The desired solution can be highly oscillatory or of *doubly* exponential growth. Or both!

Gertrude Blanch and the Eigenvalues

Gertrude Blanch (1897–1996) wrote the chapter in Abramowitz and Stegun on Mathieu functions.

She was an extremely interesting person

Her method for computing Mathieu eigenvalues used continued fractions, and she was the first to provide a good method to compute the double eigenvalues.

Spectral expansion

Once we have Mathieu functions $\mathrm{ce}_{2m}(\eta)$ and modified Mathieu functions $\mathrm{Ce}_{2m}(\xi)$ we can find our spectral expansion:

$$v(\xi,\eta) = \sum_{m=0}^{N} b_{2m} \mathrm{Ce}_{2m}(\xi) \mathrm{ce}_{2m}(\eta) \, . \tag{10}$$

The coefficients b_{2m} are found by orthogonality:

$$b_{2m} = \frac{\int_{\eta=0}^{\pi} 1 \cdot \mathrm{ce}_{2m}(\eta)\, d\eta}{\int_{\eta=0}^{\pi} \mathrm{ce}_{2m}^2(\eta)\, d\eta} \tag{11}$$

In all but one or two cases we need only 6 terms.

Complex quantities

Those integrals are *not* conjugated: it's $\mathrm{ce}_{2m}^2(\eta)$ and not $|\mathrm{ce}_{2m}(\eta)|^2$. And they are complex.

So, sometimes that integral in the denominator can be *zero*. This happens at a double eigenvalue. And *near* to a double eigenvalue, it's also a problem.

I wrote new code in Maple for this

I wrote an Hermite–Obreschkov ODE solver in Maple in order to approximate both $f(\xi)$ and $g(\eta)$ given the eigenvalues a that are determined by q. From these, we can expand

$$v(\xi,\eta) = \sum_{m\geq 0} b_{2m}\mathrm{ce}_{2m}(\eta)\mathrm{Ce}_{2m}(\xi) \tag{12}$$

and thus solve the original equation. At least, we can do this for most values of q. For some isolated values of q, which *must occur*, we have a "double eigenvalue" problem and we lose completeness.

But why did I write a new solver?

- All codes I knew of could not handle the double eigenvalue problem explicitly.
- I wanted an independent method whose solutions could be verified *a posteriori*
- Purely imaginary $q = i\rho\omega d^2/4$ is needed [1]
- The case of “near circularity” is actually *very hard* and one needs high precision.

Could there be other uses for this?

- Efficient high-accuracy solution of IVP (or BVP) for “D-finite” ODE
- Solving *delay* differential equations by the method of steps! You can use Chebfun for this, so you could use blendstrings, instead! Quadrature is so easy with blendstrings[2].
- Ned Nedialkov and John Pryce have already implemented a similar method in a quite general way, for solving DAE. Their code `DAETS` works well. Perhaps some experimental features of this Maple code could influence future development of `DAETS`. `http://www.cas.mcmaster.ca/~nedialk/daets/`

[2] I haven’t tried this yet but I will soon.

Ancillary Code

To implement this code, I first wrote (with Erik Postma) an **efficient** and **numerically stable** evaluator for what we call “blends” (arbitrary degree two-point Hermite interpolants), together with routines for manipulating them: integration, differentiation, rootfinding, addition, multiplication, etc. [2]

A “string of blends” is a particular kind of piecewise polynomial interpolant, that has some interesting properties, especially in the context of ODE solving.

Blends

Suppose that we know some Taylor coefficients of a function at two distinct points, say $z = a$ and $z = b$. Then put $z = a + s(b - a)$ and the interval $0 \leq s \leq 1$ determines a line segment in the z-plane.

Then (Hermite, Cours d'Analyse 1873)

$$\begin{aligned} H(s) = &\sum_{j=0}^{m} p_j \sum_{k=0}^{m-j} \binom{n+k}{k} s^{k+j} (1-s)^{n+1} \\ &+ \sum_{j=0}^{n} (-1)^j q_j \sum_{k=0}^{n-j} \binom{m+k}{k} s^{m+1} (1-s)^{k+j} \end{aligned} \tag{13}$$

has $H^{(j)}(0)/j! = p_j$ for $0 \leq j \leq m$ and $H^{j}(1)/j! = q_j$ for $0 \leq j \leq n$. Here differentiation is wrt s so one has to be careful about bookkeeping.

Numerical stability

Blends are *ridiculously* stable numerically. The presence of those potentially large binomial coefficients suggests the opposite, but the doubly-recursive Horner implementation has been used successfully for degrees up to about 1000, even for very hard-to-approximate functions.

There's a couple of reasons for that success.

Backward stability result

- double Horner gives the *exact* value of a blend with Taylor coefficients $p_j(1+\gamma_{O(m+n)})$ and $q_j(1+\gamma_{O(m+n)})$.
- Independent of interval scaling!
- proof uses the positivity on $0 \leq s \leq 1$

Here $\gamma_n = nu/(1-nu)$ where u is unit roundoff. See e.g. Higham's *Accuracy and Stability of Numerical Algorithms*.

Lebesgue constants

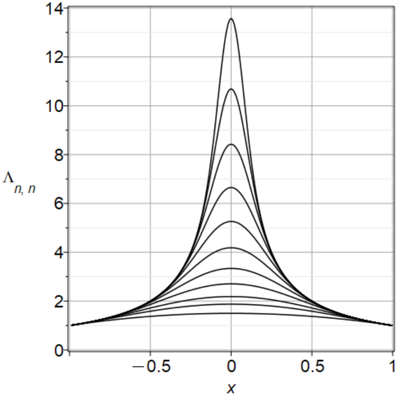


Figure 4: The Lebesgue function of balanced blends on $[-1{,}1]$ grows like $2\sqrt{n/\pi}$ where the grade is $2n+1$.

Lebesgue constants *must* be unbounded; optimal is $O(\ln n)$, like Lagrange interpolation on Chebyshev nodes. But $\sqrt{n}$ is pretty decent.

Unbalanced blends are bad, though

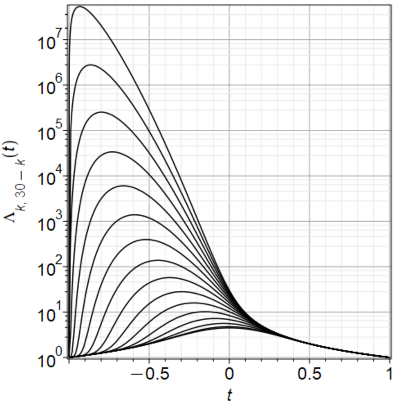


Figure 5: The Lebesgue function of *unbalanced* blends on [−1,1] grows exponentially. Just as bad as equally-spaced interpolation nodes, really.

Making it jump

If we put the series at the left to be 1,0,0, . . . and the series at the right to be -1,0,0, . . ., no analytic function can do this. Truncating, say with $m = 368$ and $n = 631$ (so the grade $m + n + 1 = 1000$) we get a polynomial, which plots below.

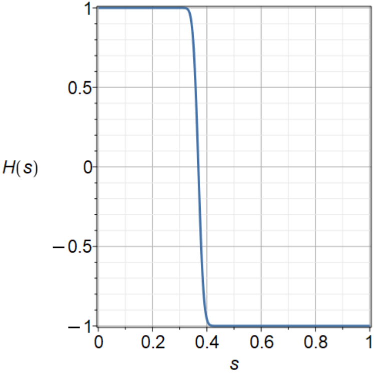


Figure 6: Look how smooth the plot is. No “Gibbs phenomenon” at all!

Blendstrings

A string of blends, or `blendstring`, is a set of the form

$$\mathcal{B} := \{L_k\}_{k=0}^{M} \tag{14}$$

where each L_k is a list of the form

$$L_k := [\alpha_k, C_{k,0}, C_{k,1}, \ldots, C_{k,m_k}] \tag{15}$$

intended to represent the known Taylor coefficients $C_{k,j}$ at the point $z = \alpha_k$.

Two blendstrings are *compatible* if they have the same knots in the same order and with the same degrees m_k at each knot. Then they can be added together, etc.

A possible blendstring

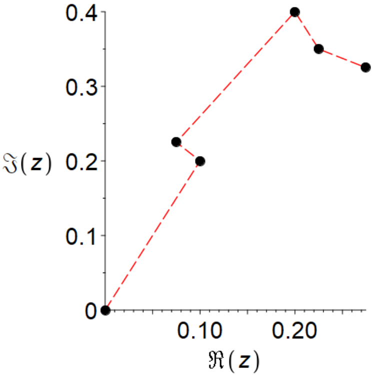


Figure 7: The knots α_k are plotted with solid circles. Taylor coefficients are known at those knots. On the segment between any two knots, Hermite's formula gives a "blend" approximating the underlying function. Paths are always straight lines in the implemented code, though

The solver

Key features (remember it's just for the Mathieu DE):

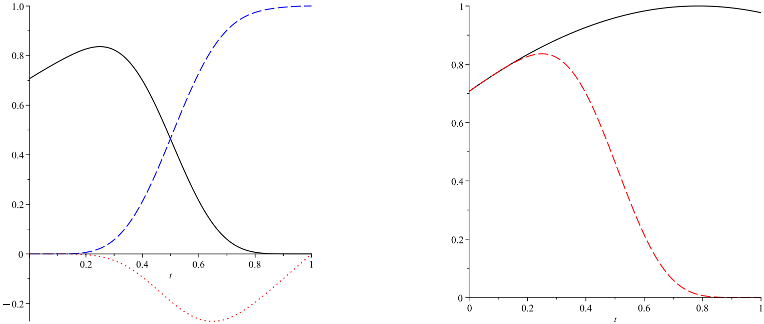
- Fixed order $2m$ that the user can choose by choosing degree m (same at each end); default is 16 by Taylor series of degree 8 at each knot. Hand-coded Taylor series loops. **[Order is not the whole story.]**
- Variable stepsize (uses Gustafsson Lund & Soderlind 1988 PID control)
- Implicit (the Mathieu equation is linear) **Not Stiff; Oscillatory**
- Collocation at $s = 1/4$ and $s = 3/4$ (Mathieu eq is 2nd order)
- Residual measured at $s = 1/2$; "defect control", asymptotically maximum
- The solver returns a blendstring (actually *two*) representing the solution (and the *other* solution, which is sometimes needed and basically free).

A few details

Taking a step from z_n to $z_{n+1} = z_n + \Delta z$ entails:

- Knowing $m+1$ Taylor coefficients at $z = z_n$. Call the solution, blended with 0 at $z = z_{n+1}$, $L(z)$
- Computing $m+1$ Taylor coefficients of $y(z)$ where $y'' + (a - 2q\cos 2z)y = 0$ with $y(z_{n+1}) = 1$ and $y'(z_{n+1}) = 0$. Call the solution, blended with 0 at $z = z_n$, $C(z)$
- Computing $m+1$ coefficients where $y(z_{n+1}) = 0$ and $y'(z_{n+1}) = 1$; call the solution blended with 0 at z_n, $S(z)$
- Computing the residuals (defects) of C and S and L at $z = z_n + \Delta z/4$ and at $z = z_n + 3\Delta z/4$ (requires 2nd derivatives of the blends)
- Choosing constants A and B so that the residual of the blend of $AC(z) + BS(z) + L(z)$ is **zero** at both those points
- Testing the residual at $z_n + \Delta z/2$ to see if the step should be accepted; if so, the $m+1$ coeffs of $AC + BS + L$ at z_{n+1} are recorded.

A picture of the collocation



(a) L, C, and S

(b) L and $y = A \cdot C + B \cdot S + L$

Figure 8: Simple collocation

An example

For $q = 1.46876861378514\ldots i$ and $a = 2.0886989027\ldots$ (the Mulholland–Goldstein double eigenvalue) we compute the eigenfunction associated to this double eigenvalue at 15 digit precision, using $m = 10$. Below is the residual using 30 digit precision, verifying *a posteriori*.

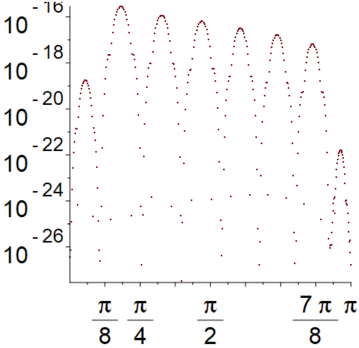


Figure 9: Residual computed at preposterously many points

Quadrature

There is a very interesting (exact!) quadrature formula for a blend, which allows *indefinite* integration of blendstrings. I don't know if this is a reinvention.

$$\begin{aligned}\int_{s=0}^{1} H(s)\,ds = &\frac{(m+1)!}{(m+n+2)!}\sum_{j=0}^{m}\frac{(n+m-j+1)!}{(m-j)!(j+1)}p_j \\ &+\frac{(n+1)!}{(m+n+2)!}\sum_{j=0}^{n}\frac{(-1)^j(n+m-j+1)!}{(n-j)!(j+1)}q_j \qquad (16)\end{aligned}$$

Differentiation

The numerical routine Erik Postma and I wrote to evaluate blends also has the ability to evaluate arbitrary derivatives, by what I call "semi-automatic differentiation".

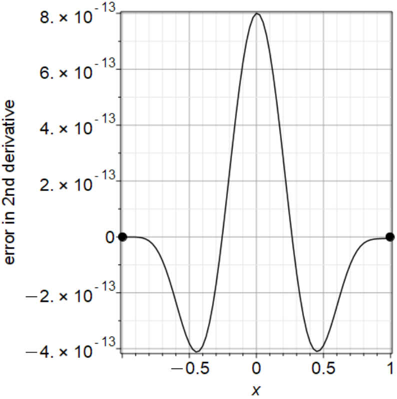


Figure 10: Error in 2nd derivative with $m = 7$ at either end

Some Hemodynamics results!

We need to compute the coefficients b_{2m} in

$$u_{\phi,e}(\xi,\eta,t) = \frac{4\hat{u}_{0,e}}{i\lambda_e}\left(1 - \sum_{m\geq 0} b_{2m}\mathrm{Ce}_{2m}(\xi,q)\mathrm{ce}_{2m}(\eta,q)\right)e^{i\omega t} \tag{17}$$

The integrals of $\mathrm{ce}_{2m}(\eta,q)$ against 1 and against itself are immediate, using the blendstring quadrature. When the eigenvalues are simple, this is enough and gives the solution.

Double eigenvalues

At the Mulholland–Goldstein point q^*, $\mathrm{ce}_0(\eta,q)$ and $\mathrm{ce}_2(\eta,q)$ coalesce. So we need a generalized eigenfunction. As with the eigenfunctions, we need to be able to integrate them and differentiate them and to combine them.

Time permitting I will give the detailed derivation using series that we have put in the paper.

The generalized eigenfunctions

We will also need the following two new functions:

$$u(\eta) := D_2(w_I)(\eta, a^*, q^*) \tag{18}$$
$$U(\xi) := D_2(w_I)(i\xi, a^*, q^*)\,. \tag{19}$$

Here $D_2(f)(x,y,z)$ means the partial derivative with respect to the second variable, and then evaluated at the point (x,y,z).

So we are *differentiating with respect to the eigenvalue.*

How to compute $u(\eta)$ and $U(\xi)$

Fréchet derivatives of the Mathieu differential equation:

$$\frac{d^2u}{d\eta^2} + (a^* - 2q^* \cos 2\eta)u + y = 0 \qquad (20)$$

$$\frac{d^2U}{d\xi^2} - (a^* - 2q^* \cosh 2\xi)U - y = 0\,. \qquad (21)$$

I use the Green's function for this (because blendstrings can be integrated basically instantly).

The Green's function

$$G(\eta,\tau) = w_I(\eta)w_{II}(\tau) - w_I(\tau)w_{II}(\eta)\,. \tag{22}$$

Then the solution of the forced equations is immediate.

$$u(\eta) = -\int_{\tau=0}^{\eta} G(\eta,\tau)\mathrm{ce}_0(\tau)\,d\tau \tag{23}$$

and analogously for $U(\xi)$. We need $U(0) = U'(0) = 0$ to preserve symmetry, and $u(0) = u(\pi) = 0$ to preserve periodicity.

At q near q^*

Now suppose that the solution at the point $q = q^* + x^2$ has the expansion

$$v(\xi,\eta) = b_0 \mathrm{Ce}_0(\xi,q)\mathrm{ce}_0(\eta,q) + b_2 \mathrm{Ce}_2(\xi,q)\mathrm{ce}_2(\eta,q) + \cdots \tag{24}$$

where the terms not included have eigenvalues that will not coalesce, and therefore the previous treatment using orthogonality will suffice to identify their coefficients b_{2m} for $m > 1$.

Series expansion

$$
\begin{aligned}
a_0 &= a^* - \alpha_1 x + O(x^2)\\
a_2 &= a^* + \alpha_1 x + O(x^2)\\
b_0 &= \frac{A_0}{x} + B_0 + O(x)\\
b_2 &= -\frac{A_0}{x} + B_2 + O(x)\\
\mathrm{Ce}_0(\xi,q) &= \mathrm{Ce}_0(\xi,q^*) - \alpha_1 U(\xi)x + O(x^2)\\
\mathrm{ce}_0(\eta,q) &= \mathrm{ce}_0(\eta,q^*) - \alpha_1 u(\eta)x + O(x^2)\\
\mathrm{Ce}_2(\xi,q) &= \mathrm{Ce}_0(\xi,q^*) + \alpha_1 U(\xi)x + O(x^2)\\
\mathrm{ce}_2(\eta,q) &= \mathrm{ce}_0(\eta,q^*) + \alpha_1 u(\eta)x + O(x^2)\,.
\end{aligned}
\tag{25}
$$

Coefficients

$$v(\xi,\eta) = b_0 \mathrm{Ce}_0(\xi,q^*)\mathrm{ce}_0(\eta,q^*) + \hat{b}_2\left(U(\xi)\mathrm{ce}_0(\eta,q^*) + \mathrm{Ce}_0(\xi,q^*)u(\eta)\right) + \cdots \tag{26}$$

where

$$b_0 = \frac{\int_{\eta=0}^{2\pi} u(\eta)\,d\eta}{\mathrm{Ce}_0(\xi_0,q^*)\int_{\eta=0}^{2\pi}\mathrm{ce}_0(\eta,q^*)u(\eta)\,d\eta} + \frac{\left(\int_{\eta=0}^{2\pi}\mathrm{ce}_0(\eta,q^*)\,d\eta\right)\left(\mathrm{Ce}_0(\xi_0,q^*)\int_{\eta=0}^{2\pi}u^2(\eta)\,d\eta + U(\xi_0)\int_{\eta=0}^{2\pi}\mathrm{ce}_0(\eta,q^*)u(\right.}{\left(\mathrm{Ce}_0(\xi_0,q^*)\int_{\eta=0}^{2\pi}\mathrm{ce}_0(\eta,q^*)u(\eta)\,d\eta\right)^2} \tag{27}$$

$$\hat{b}_2 = \frac{\int_{\eta=0}^{2\pi}\mathrm{ce}_0(\eta,q^*)\,d\eta}{\mathrm{Ce}_0(\xi_0,q^*)\int_{\eta=0}^{2\pi}\mathrm{ce}_0(\eta,q^*)u(\eta)\,d\eta}\,, \tag{28}$$

That was known, but ignored

The classic book [4] has, hidden away in a difficult chapter, a single page explaining how to do what I just showed. Nobody's code (but mine) actually does it. [That I know of.]

It's mostly ok. Mostly the double eigenvalue problem never hurts very much. The coefficients $A_0/\sqrt{q - q^*}$ and its negative are only bad when you are *really* near the double eigenvalue.

Still. Hmmph.

Where next?

The approximation theoretic properties of blends are “understood” in the sense that there are results scattered throughout the literature. By and large they are in many ways inferior to Chebyshev expansion, and to the new techniques such as AAA. Nonetheless they seem interesting to me and I shall spend some time writing things down to my satisfaction.

I would also like add Laurent series and Puiseux series, in order to explore the singularity detection and location facilities afforded by these interpolants.

Fine-tuning all the safety factors and timeouts in the code should make it more efficient (10% ? more?) (Translating it to a fast language, e.g. Julia, would do a *lot* more)

Extending this to variable order would be fun, but first, generalize it to other IVP.

Thank you for listening.

This work supported by NSERC, by the Spanish MICINN, and by the Isaac Newton Institute in Cambridge.

I thank Erik Postma and my other co-authors, and especially I thank John C. Butcher for teaching me the contour integral technique for interpolation, by which I (re)derived all these formulae. Finally, I thank **R. Bruce Simpson (1940–2020)**, who was instrumental in teaching me what Applied Mathematics was, and how to do Scientific Computing; and with whom I published my first paper, forty years ago [3].

References

[1] Chris Brimacombe, Robert M. Corless, and Mair Zamir. “Computation and Applications of Mathieu Functions: A Historical Perspective”. In: *SIAM Review* 63.4 (Nov. 2021), pp. 653–720. URL: `https://doi.org/10.1137/20m135786x`.

[2] Robert M Corless and Erik Postma. “Blends in Maple”. In: *arXiv preprint arXiv:2007.05041* (2020).

[3] Robert M. Corless and R. Bruce Simpson. “An “off-the-shelf” dynamic mesh capability”. In: *ACM SIGNUM Newsletter* 18.2 (1983), pp. 23–24.

[4] Josef Meixner, Friedrich W Schäfke, and Gerhard Wolf. *Mathieu functions*. Springer, 1980, pp. 63–101.