Hermite Interpolational Bohemians: An ongoing project with Lalo

Robert M. Corless

from before 2004 until 2024 and beyond

Maple Transactions, ORCCA, Western University & University of Waterloo, Canada

Announcement: Maple Transactions

Maple Transactions

an open access journal with no page charges

mapletransactions.org

We welcome expositions on topics of interest to the Maple community, including in computer-assisted research in mathematics, education, and applications. Student papers especially welcome.

Another announcement

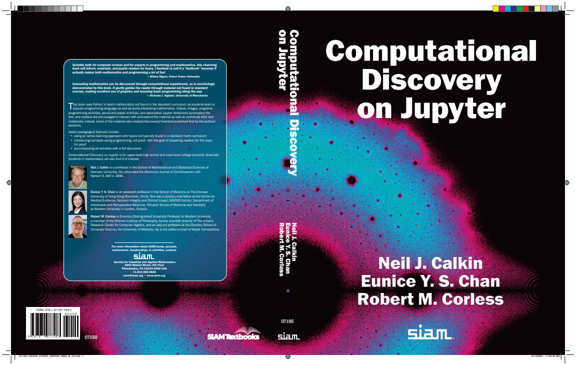

Figure 1: A new book from SIAM: Calkin, Chan, & Corless, "Computational Discovery on Jupyter", published November 2023

Has some background on Bohemian Matrices.

Bohemian Matrices

A family of matrices is called “Bohemian” if all entries are all from a single finite population P. The name comes from BOunded HEight Matrix of Integers. See bohemianmatrices.com for instances.

See also the [link] London Mathematical Society Newsletter, November 2020, page 16.

Such matrices have been studied for quite a long time (e.g. by Olga Taussky–Todd), though the name “Bohemian” only dates to 2015. See also the Wikipedia entry at `https://en.wikipedia.org/wiki/Bohemian_matrices`.

A partial list of some related work

- Chan, E. Y., Corless, R. M., **Gonzalez-Vega, L.**, Sendra, J. R., Sendra, J., & Thornton, S. E. (2020). Upper Hessenberg and Toeplitz Bohemians. LAA, 601, 72-100.
- Aruliah, D. A., Corless, R. M., Diaz-Toca, G. M., **Gonzalez-Vega, L.**, & Shakoori, A. (2015). The Bézout matrix for Hermite interpolants. LAA, 474, 12-29.
- Corless, R. M., Diaz-Toca, G. M., Fioravanti, M., **Gonzalez-Vega, L.**, Rua, I. F., & Shakoori, A. (2013). Computing the topology of a real algebraic plane curve whose defining equations are available only "by values". CAGD, 30(7), 675-706.
- Diaz-Toca, G. M., Fioravanti, M., **Gonzalez-Vega, L.**, & Shakoori, A. (2013). Using implicit equations of parametric curves and surfaces without computing them: Polynomial algebra by values. CAGD, 30(1), 116-139.

Continued

- Butcher, J. C., Corless, R. M., **Gonzalez-Vega, L.**, & Shakoori, A. (2011). Polynomial algebra for Birkhoff interpolants. Numerical Algorithms, 56(3), 319-347.
- Corless, R. M., Shakoori, A., Aruliah, D. A., & **Gonzalez-Vega, L.** (2008). Barycentric Hermite interpolants for event location in initial-value problems. JNAIAM 3, 1-16.
- Aruliah, D. A., Corless, R. M., **Gonzalez-Vega, L**., & Shakoori, A. (2007, July). Companion matrix pencils for Hermite interpolants. Proc SNC (pp. 197-198).
- Amiraslani, Corless, **Gonzalez-Vega**, Shakoori: **Polynomial Algebra by Values** (January 2004). RMC Invited talk EACA 2004 Santander

Evolving promises

European Society of Computational Methods in Sciences and Engineering (ESCMSE)

Journal of Numerical Analysis, Industrial and Applied Mathematics (JNAIAM)
vol. 3, no. 1-2, 2008, pp. 1-16
ISSN 1790–8140

Barycentric Hermite Interpolants for Event Location in Initial-Value Problems[1]

Robert M. Corless[2]**, Azar Shakoori, D.A. Aruliah, Laureano Gonzalez-Vega**

Department of Applied Mathematics,
Faculty of Sciences,
University of Western Ontario,
Ontario, Canada

Received 2 December, 2007; accepted in revised form 20 December, 2007

[24] A. Shakoori, D. A. Aruliah, Robert M. Corless, and L. Gonzalez-Vega. Bézoutians and companion matrix pencils for barycentric Hermite interpolants. To be submitted.

Figure 2: A reference from our 2007 paper, that was never finished (the 2015 LAA paper replaced part of it)

Figure 3: RMC, LGV, and the late Agnes Szanto (1966–2022) at the EACA 2004 conference banquet in Somo

Today's problem

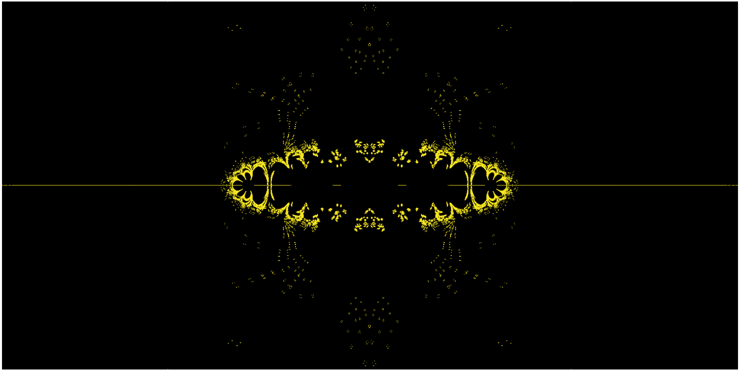

Figure 4: A density plot of all 9,961,472 zeros of all confluency-4 Hermite interpolants taking values, slopes, and derivatives ± 1 on the nodes $[-1, -1/2, 0, 1/2, 1]$. Polynomial zeros were computed by solving generalized eigenvalue problems. Why flower petals?

Zoomed

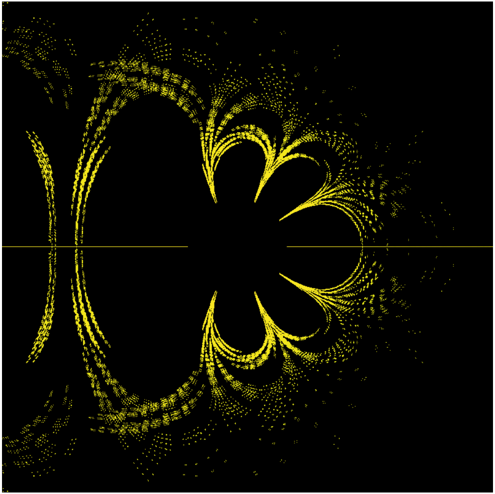

Figure 5: Confluency 4, zoomed in to $0.75 \leq \Re(\lambda) \leq 1.25$, $-0.25 \leq \Im(\lambda) \leq 0.25$. The flower petals are clearer, and weirder. Looks like 8 spikes around a dark circle.

Confluency 5

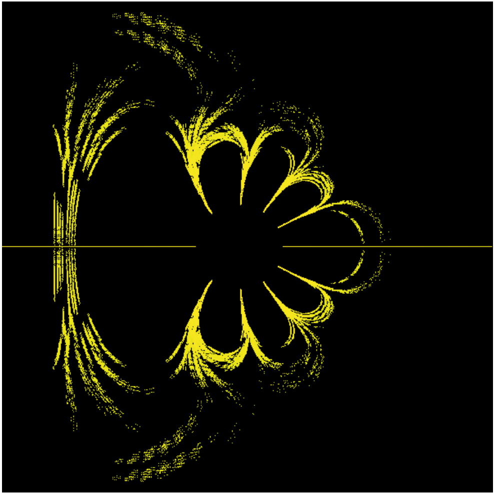

Figure 6: Confluency 5 at each node (so, grade 24) zoomed in to $0.75 \leq \Re(\lambda) \leq 1.25$, $-0.25 \leq \Im(\lambda) \leq 0.25$. We count 10 spikes around a dark circle. Sample of 500,000 matrices, so 12M eigenvalues.

Confluency 6

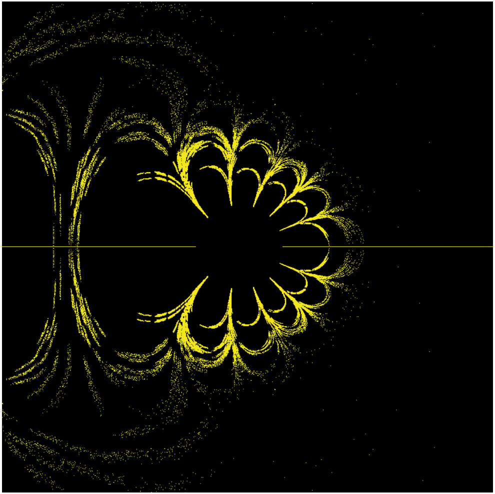

Figure 7: Confluency 6 at each node (so, grade 29) zoomed in to $0.75 \leq \Re(\lambda) \leq 1.25$, $-0.25 \leq \Im(\lambda) \leq 0.25$. Count: 12 spikes. Sample of 500,000 matrices, so 14.5M eigenvalues.

Confluency 7

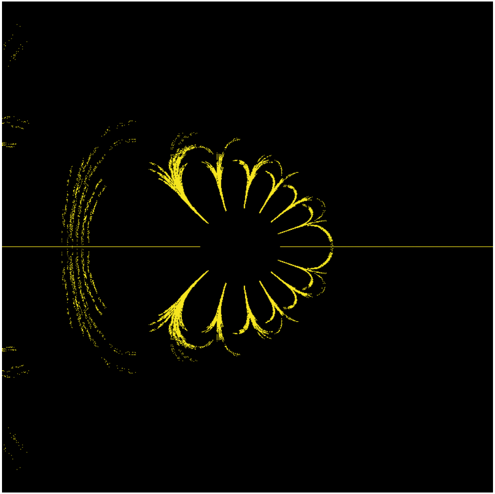

Figure 8: Confluency 7 at each node (so, grade 34) zoomed in to $0.75 \le \Re(\lambda) \le 1.25$, $-0.25 \le \Im(\lambda) \le 0.25$. Count: 14 spikes. Sample of 500,000 matrices, so 17M eigenvalues.

Confluency 13

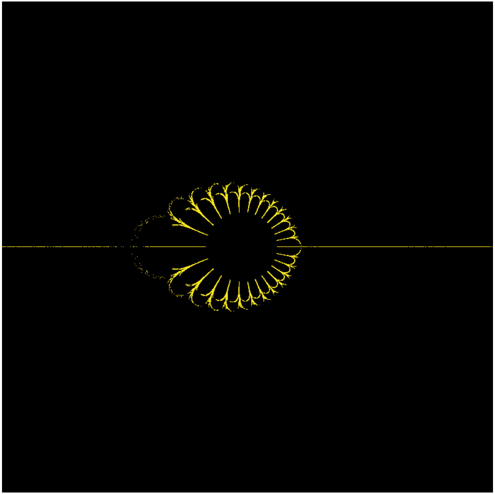

Figure 9: Confluency 13 at each node (so, grade 64) zoomed in to $0.75 \leq \Re(\lambda) \leq 1.25$, $-0.25 \leq \Im(\lambda) \leq 0.25$. Count: 26 spikes. Sample of 250,000 matrices, so 16M eigenvalues.

Confluency 12 with 13 in the middle

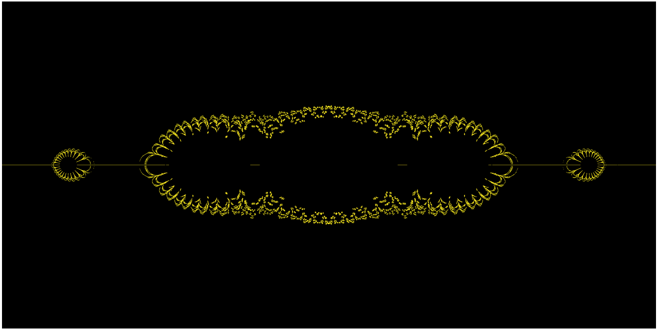

Figure 10: Confluency 12 at each nonzero node and 13 at zero, so, grade 60—Happy Birthday, Lalo! Plotted on $-1.25 \leq \Re(\lambda) \leq 1.25$, $-0.625 \leq \Im(\lambda) \leq 0.625$. Most features of this figure are *unexplained*. [But I have ideas.]

The Hermite Interpolational Companion Pencil

Put

$$C_0 = \begin{bmatrix} 0 & -f/2 & -e & -d & -c & -b & -a \\ 1/16 & 7 & 0 & 0 & 0 & 0 & 0 \\ -5/64 & 1 & 7 & 0 & 0 & 0 & 0 \\ 17/256 & 0 & 1 & 7 & 0 & 0 & 0 \\ -1/16 & 0 & 0 & 0 & 5 & 0 & 0 \\ -1/16 & 0 & 0 & 0 & 1 & 5 & 0 \\ -1/256 & 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix} \qquad (1)$$

and C_1 the identity matrix with its upper-left entry zeroed out. Then $p(z) = \det(zC_1 - C_0)$ is a polynomial of grade 5 satisfying $p(3) = a$, $p(5) = b$, $p'(5) = c$, $p(7) = d$, $p'(7) = e$, and $p''(7) = f$. The generalized eigenvalues of the matrix pencil (C_0, C_1) therefore give the roots of $p(z) = 0$.

What are those (unnormalized) numbers in the first column?

Those numbers are from a partial fraction

$$\frac{1}{(z-3)(z-5)^2(z-7)^3} = -\frac{1/256}{(z-3)} - \frac{1/16}{(z-5)} - \frac{1/16}{(z-5)^2} + \frac{17/256}{(z-7)} - \frac{5/64}{(z-7)^2} + \frac{1/16}{(z-7)^3}\,. \tag{2}$$

For floating-point nodes we want to use a numerically stable algorithm (Maple's "convert to parfrac" is not, but just bump up Digits if you only have one Hermite interpolation problem to do).

You just read them off in order and fill the column.

Notice the block-diagonal structure, as well: a transposed 3 by 3 Jordan block for the triply-confluent node $z = 7$, a transposed 2 by 2 Jordan block for the doubly-confluent node $z = 5$, and a diagonal entry for $z = 3$ which has "confluency" just 1.

The general case

If we write the node polynomial as

$$w(x) = \prod_{i=0}^{m} (x - x_i)^{s_i} \tag{3}$$

then we may compute the *generalized* barycentric weights by computing the partial fraction decomposition of

$$\frac{1}{w(x)} = \sum_{i=0}^{m} \sum_{j=0}^{s_i - 1} \frac{\beta_{i,j}}{(x - x_i)^{j+1}} \tag{4}$$

and then the barycentric forms can be written down immediately [Corless & Fillion 2013, chapter 8]:

$$p(z) = w(z) \sum_{i=0}^{m} \sum_{j=0}^{s_i-1} \sum_{k=0}^{j} \beta_{i,j} \rho_{i,k} (z - x_i)^{k-j-1} \tag{5}$$

and

$$p(z) = \frac{\displaystyle\sum_{i=0}^{m} \sum_{j=0}^{s_i-1} \sum_{k=0}^{j} \beta_{i,j} \rho_{i,k} (z - x_i)^{k-j-1}}{\displaystyle\sum_{i=0}^{n} \sum_{j=0}^{s_i-1} \beta_{i,j} (z - x_i)^{-j-1}} . \tag{6}$$

Code

Matlab code to compute the generalized barycentric weights can be found at `http://www.nfillion.com/coderepository/Graduate_Introduction_to_Numerical_Methods/genbarywts.m` and Matlab code to evaluate the Hermite interpolational polynomial can be found at `http://www.nfillion.com/coderepository/Graduate_Introduction_to_Numerical_Methods/hermiteval.m`. The Maple code to do the same, called `BHIP`, is in the workbook accompanying my paper "Barycentric Hermite Interpolation," Maple Transactions Volume 4 Issue 2 (July 2024).

Scaling

We can multiply that first row, or the first column, by any scalar we like and the eigenvalues will not change. The matrices will be more balanced if we scale the first column so it has norm 1. Dividing by the largest element does that. This helps the accuracy of the eigenvalues for larger dimensions.

The eigenvalue problem *does* get difficult to solve as the confluency increases, because *some* of the generalized barycentric weights grow combinatorially. Having weights of greatly different sizes makes the eigenvalue problem difficult.

But [I claim] it's better conditioned that expanding it all out into the monomial basis is. And that poor conditioning slows down fsolve.

Linear in the first row

Since "determinant" is linear in the first row, we can find explicit determinantal expressions for the Hermite interpolational basis on those nodes with that set of confluencies: e.g. put

$$H_{2,0} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1/16 & z-7 & 0 & 0 & 0 & 0 & 0 \\ 5/64 & -1 & z-7 & 0 & 0 & 0 & 0 \\ -17/256 & 0 & -1 & z-7 & 0 & 0 & 0 \\ 1/16 & 0 & 0 & 0 & z-5 & 0 & 0 \\ 1/16 & 0 & 0 & 0 & -1 & z-5 & 0 \\ 1/256 & 0 & 0 & 0 & 0 & 0 & z-3 \end{bmatrix} \tag{7}$$

and then $\det H_{2,0} = [c](p(z))$ is the Hermite interpolational basis polynomial that picks out the term that is 0 when $z = 5$, has derivative 1 when $z = 5$, and is 0 at all other spots where data is given. (We already had an explicit expression for this determinant, I just thought this was neat.)

Back to the Bohemian problem

The five nodes are $[-1, -1/2, 0, 1/2, 1]$ and for the first computation we know four pieces of information $p(\tau_k)$, $p'(\tau_k)$, $p''(\tau_k)$, and $p'''(\tau_k)$ at each one, so twenty pieces of information. Therefore the grade of the polynomial is 19, one less than 20. The companion pencil for the problem is 21 by 21, so two of its eigenvalues will spuriously be ∞. We throw those away.

More details

For our problem, each of those twenty pieces of information can be 1 or -1 only—this is a "Bernoulli" problem, analogous to Littlewood polynomials $\pm 1 \pm x \pm x^2 \pm \cdots \pm x^n$. So there are 2^{20} such sets of numbers. We take advantage of one symmetry, though: if $p(z)$ is in this collection, so is $-p(z)$, and so we can fix one of those values: say $p(-1) = 1$. This means there are "only" $2^{19} = 524{,}288$ such matrices, so at most $2^{19} \times 19 = 9{,}961{,}472$ eigenvalues (roots).

It took Maple about 14 minutes to compute all these eigenvalues (which was more than three times as fast as computing the zeros of the degree 19 polynomials, even though Maple's fsolve has been greatly improved in Maple 2024).

Some inefficiencies

- Each matrix has two spurious infinite eigenvalues. For 20 by 20 matrices, that's about 10% inefficient
- Solving a generalized eigenvalue problem is about 5 times as expensive as solving a simple eigenvalue problem of the same dimension
- Solving eigenvalue problems costs $O(m^3)$ flops while solving polynomial problems "ought" to cost only $O(m^2)$. Specialized algorithms exist for many matrices of analogous structure that are $O(m^2)$ and reasonably stable [Leonardo Robol, Jared Aurentz, David Watkins, others]. But nothing that I know of for this exact structure.

That last one particularly bites: these experiments could potentially go 20 times faster, so one minute each instead of twenty minutes, for the dimensions used for this talk.

Some efficiencies

- Still *way* faster than fsolve (at all confluencies I tried)
- Not changing basis means these computations are all as numerically stable as possible
- The problems are in fact well-conditioned* in this basis
- **Neither I nor anyone I know** had to write any specialized solvers. I just used `LinearAlgebra:-Eigenvalues` right off the shelf (and threw away the first two eigenvalues, which were always the infinite ones).
- The problem is “trivially parallel.” All I need to do is make the Task model work in Maple (or switch to Julia). **Maple can now automatically do some things in parallel.**

* if the confluency is not “too large,” that is

Look at the time

$> rtvecE := CodeTools\text{:-}Usage(\ getSomeEigenvals(500000, m, evalf(P),$
$fam\ )\ ) :$

```
memory used=226.41GiB, alloc change=0.50GiB, cpu time=
4.48h, real time=66.02m, gc time=12.81m
```

Figure 11: On this 4-core, 8-thread Surface Pro, sometimes Maple takes substantial advantage. I did *nothing* to direct the parallelism.

Concluding Remarks

This particular Bohemian problem (why flower petals?) is not yet understood—though I have an idea! But I hope that you can see that

- One can solve Hermite interpolational problems easily
- One can find roots of polynomials expressed in Hermite interpolational bases *without* changing the polynomial basis
- Experimental mathematics can use these tools
- Bohemian matrices make good test problems.

Thank you

Thank you for listening! And,

Happy Birthday, Lalo!

This work was partially supported by NSERC grant RGPIN-2020-06438, and partially supported by the grant PID2020-113192GB-I00 (Mathematical Visualization: Foundations, Algorithms and Applications) from the Spanish MICINN.

The *conference* LALO60 has been generously supported by The Fields Institute for Mathematical Research, CUNEF University, ORCCA, CARGO, Maplesoft, and the Computer Science Department at Western University.