Symbolic-Numeric Computing for Bohemian Matrices

Robert M. Corless
Joint work with many others

December 9, 2021

Western University, Canada

Announcing Maple Transactions

a new open access journal with no page charges

mapletransactions.org

For example, see

Peter J. Baddoo and Lloyd N. Trefethen. *[illegible]*. Maple Transactions Volume 1, Issue 1, Article 14124 (July 2021). https://doi.org/10.5206/mt.v1i1.14124

Richard P. Brent. *[illegible]*. Maple Transactions Volume 1, Issue 1, Article 14069 (July 2021). https://doi.org/10.5206/mt.v1i1.14069

There is also a transcript of an interview with these authors, conducted by Annie Cuyt.

If you’re getting worried about this talk (so SLOW!)

You can find an earlier (longer) version at

a video on my YouTube channel entitled “Skew Symmetric Tridiagonal Bohemians”

The papers the talk today is based on are

What can we learn from Bohemian Matrices?
https://doi.org/10.5206/mt.v1i1.14039

and

Skew-symmetric tridiagonal Bohemian matrices
https://doi.org/10.5206/mt.v1i2.14360

What is Hybrid Symbolic-Numeric Computation, anyway?

Hybrid Symbolic-Numeric Computation: A computation that uses *’‾ẅ˻ !‾˻ẅā* and either *!ωђ‾ð̇ђ˻ð̇‾* or *gāƕ’˻ẅā*.

That is, it uses notions both from algebra (or geometry) and from analysis.

I prefer this definition over others. Excellent examples of its use in general scientific computing include modern research in Differential-Algebraic Equations, Nonlinear Eigenvalue Problems, and spectral methods.

Bohemian Matrices

A family of matrices is called "Bohemian" if all entries are all from a single finite population $\breve{e}$. The name comes from BOunded HEight Matrix of Integers. See bohemianmatrices.com for instances.

See also 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 2016.

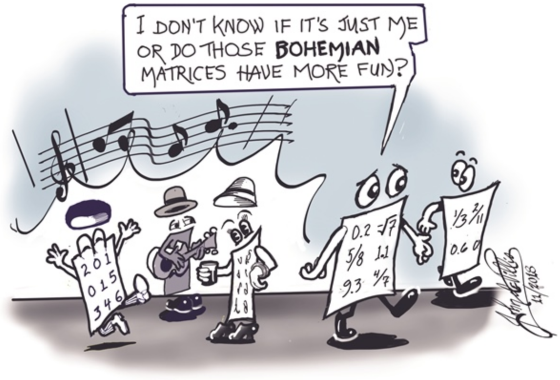


Figure 1: A cartoon by mathematician John de Pillis (UC Riverside), which appeared in Nick Higham's column in SIAM News

Why we're interested

Framing the problem this way allows one to study common properties of discrete random matrices, to investigate extreme possibilities by exhaustive computation, and to analyze several interesting combinatorial problems.

Our original motivation was simply the construction of test problems for eigenvalue solvers; Steven Thornton has solved several w0yy0ωɧ eigenvalue problems, and uncovered small instances (10 by 10 matrices with complex entries, 20 by 20 matrices with real entries) for which Matlab's `dhf` routine failed to converge. [Reported to the Mathworks.]

Other uses

Nick Higham has used Bohemian matrices as a class to optimize over to look for improved lower bounds on such things as the growth factor in matrix factoring; Laureano Gonzalez-Vega has looked at *!ωʷʷāɏƕ‾0ωɧ gƕ‾ʷ0!ā’*.

We have used this idea to understand some things about simple matrix structures, such as upper Hessenberg and upper Hessenberg Toeplitz matrices.

Skew-symmetric tridiagonal Bohemian matrices

Here is a seven-by-seven example of SSTB[$\breve{e}$]. If $\breve{e}$ has $\#\breve{e}$ elements, then the number of such matrices is $\#\breve{e}^6$.

$$\begin{bmatrix} 0 & \text{ῤ}_1 & 0 & 0 & 0 & 0 & 0 \\ \text{ῤ}_1 & 0 & \text{ῤ}_2 & 0 & 0 & 0 & 0 \\ 0 & \text{ῤ}_2 & 0 & \text{ῤ}_3 & 0 & 0 & 0 \\ 0 & 0 & \text{ῤ}_3 & 0 & \text{ῤ}_4 & 0 & 0 \\ 0 & 0 & 0 & \text{ῤ}_4 & 0 & \text{ῤ}_5 & 0 \\ 0 & 0 & 0 & 0 & \text{ῤ}_5 & 0 & \text{ῤ}_6 \\ 0 & 0 & 0 & 0 & 0 & \text{ῤ}_6 & 0 \end{bmatrix} \tag{1}$$

A picture

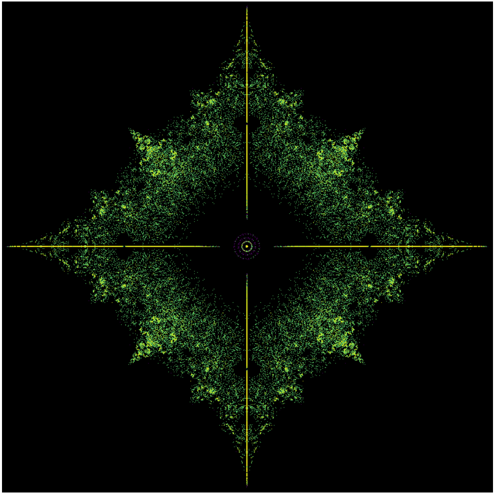


Figure 2: Density of eigenvalues of all $4^{14} = 268;435;456$ fifteen by fifteen skew-symmetric tridiagonal matrices with population $\breve{e} = [\pm 1; \pm \mathring{0}]$. Note the "rose" in the middle and its symmetries. Computed in Maple (15 seconds).

Eigenvalue Computation

The computation of eigenvalues is well-known to be *ɧ˞ ɡ ā‍ꟺ!ƕƴƴ˴ ’‾ƕƐƴā* and is efficiently carried out by QR iteration (Francis, Kublanovskaya independently early 60s). It would be hard to think of a more important algorithm; certainly it's one of the top 50 of the previous century.

But sometimes, even so, there can be difficulty.

Remarkably (to a numerical analyst) sometimes[1] computation of the roots of the characteristic polynomial can be better.

For *!ωɡ Ꞩƴā’ ’ɟā˝ ˥’˴ ɡ ɡ ā‾ꟺ! ‾ꟺăƕÔωɧƕƴ* matrices there ought to be specialized algorithms that are faster by a factor of about 15 here, but they were not necessary (even if they exist).

[1] Admittedly, hardly ever

For this class, polynomials are pretty good

Figure 2 was computed using eigenvalues of only $2^{14} = 16{,}384$ matrices (thus explaining the 15 seconds taken), with $\beta = [1, i]$ not the $4^{14} > 2.68 \times 10^{8}$ matrices with $\beta \in [\pm 1, \pm i]$. I could have done even better by using just the 8146 unique characteristic polynomials of this family. The characteristic polynomials satisfy the recurrence relation

$$p_{k+1} = z p_k + \beta_k^2 p_{k-1} \tag{2}$$

and $p_0 = 1$, $p_1 = z$. **So there is no need for** -1 **or** $-i$.

Even so, I used ***Dhfdmu`ktdr***, because degree fifteen polynomials can still be ill-conditioned. [I could have used MPSolve by Bini and Robol, though.]

The polynomials are not perfect, though

Theorem

İḓā maximum characteristic height $^{a}{}_{g}$ *ωı̀ ¯ḓā’ā ăŏ g āɧ’ŏωɧ g g ƕ¯ẅŏ!ā’ ŏ’ ƹω˞ ɧăāă ƹ˓*

$$\frac{1}{g+1}\{_{g+1} < {}^{a}{}_{g} < \{_{g+1} \tag{3}$$

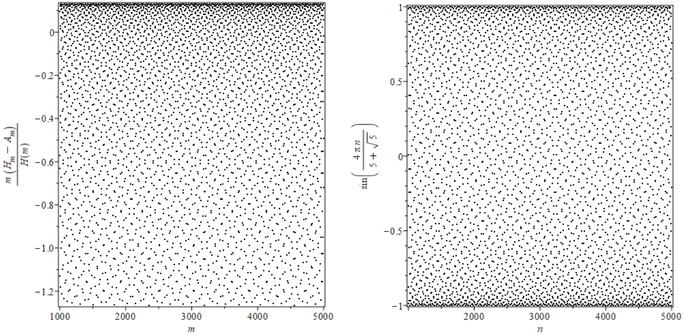
˝ ḓāẅā $\{_0 = 0$, $\{_1 = 1$ *ƕɧă* $\{_{j+1} = \{_j + \{_{j-1}$ *ƕẅā ¯ḓā {ŏƹωɧƕ!!ŏ ɧ˞ g ƹāẅ’¡*

Ùωẅā ˝ ā ḓƕ˙ā

$$^{a}{}_{g} \qquad {}_{g} := \frac{5^{1/4}}{\sqrt{2\pi(g+1)}}^{\,g+1} \tag{4}$$

˝ ḓāẅā $= (1+\sqrt{5})/2$ *ŏ’ ¯ḓā Ôωɏăāɧ ẅƕ¯ŏω¡ This means that the condition number for the worst of these polynomials grows exponentially with the dimension¡*

A Thousand Points of Light



(a) Relative Error in Asymptotic Formula

(b) Sampling sine at discrete points

Figure 3: See also Doug Hardin and Gil Strang's lovely paper "A Thousand Points of Light".

But some of the polynomials are needed

The real issue is g [illegible]. It turns out that at dimensions $g = 2^j$ 1 for some j, there are 2^j 1 nilpotent matrices in this family.

g	#matrices 4^{g} 1	#matrices 2^{g} 1	#nilpotents
1	1	1	1
3	16	4	2
7	4096	64	4
15	2:6844 10^8	16384	8
31	1:1529 10^{18}	1073741824	16

Table 1: Number of nilpotents (by exhaustive computation, the last one in Julia on a 4-core 1.3GHz machine which took just over 2 minutes).

Theorems (proofs in the paper)

Theorem
[illegible] = 2^j − 1 [illegible]

Theorem
[illegible] = 1 [illegible] = 1 [illegible] [0] [illegible] = 2 [illegible] = 3 [illegible] [1; [illegible]] [illegible] [[illegible]; 1] [illegible] = 2^j − 1 [illegible] [[illegible]; 1; [illegible]; rev([illegible])] [illegible] [[illegible]; [illegible]; 1; rev([illegible])] [illegible] = 2^{j+1} − 1

I thought I had a proof that this construction gave [illegible] the nilpotents, but I now think I don't.