### On the Unitary Block-Diagonalisation of General Matrices and Applications

#### Frank Uhlig

Abstract

### Abstract

We study new matrix based computations for a recent cluster of extraordinary results in six distinct branches of mathematics that are inter-connected in multiple multidimensional ways. The first quoted paper deals with fractional ordinary differential equations and the proportional secting method for accelerating terminal value problems therein by a factor of around 8. Then we study how a century old and previously unsolved quantum physics problem can be solved by using the hermitean Johnson  $\mathcal{F}(t)$  function of field of values computations. In quantum physics terms, this solution makes an assessment of our Chemical Element Tables finally possible after 100 + years of not knowing. Then we study accurate and fast computations of field of values boundary curves, even for decomposable matrices. This was impossible before and has been abandoned for several years now. To solve the unitary block decomposition problem for general square matrices we use a discretized predictive Zhang Neural Network method for the resulting Johnson block  $\mathcal{F}_{j}(t)$  field of values functions. Overall the computational methods in this cluster are all conditionally stable and none is just backward stable. They all give us highly accurate results such as adapted ZNN methods for matrix flow A(t) problems that find nonsingular static matrix A symmetrizers with small condition numbers for the first time. A detailed survey of Zhang Neural Networks details their seven step set-up process for the first time, giving ten matrix flow example derivations of this new process. BOX

**Keywords:** math history, numerical analysis, fractional ODEs, shooting methods, proportional secting, Linear algebra, unitary block-decomposition, invariant subspace theory, quantum physics, time-varying matrix problem, conditionally stable algorithm, Johnson matrix flow, neural network, zeroing neural network, discretized ZNN algorithm, field of values, matrix flows, time-varying numerical algorithm, ZNN set-up, predictive numerical method, matrix symmetrizer

**AMS** : 65F10, 65F15, 65F30, 65J10, 65L10, 34A08, 35A08, 15A18, 15A21, 15A23, 15A60, 15B57, 81Q10

# An Introduction – of sorts

For general complex or real 1-parameter matrix flows  $A(t)_{n,n}$  or for static matrices Athis paper considers ways to decompose matrix flows A(t) or single matrices  $A_{n,n}$  globally via one constant hermitean matrix similarity  $C_{n,n}$  as

$$A(t) = C^{-1} \cdot diag(A_1(t), ..., A_{\ell}(t)) \cdot C$$

or

$$A = C^{-1} \cdot diag(A_1, \dots, A_\ell) \cdot C.$$

Here each diagonal block  $A_k(t)$  or  $A_k$  is square and their number  $\ell$  exceeds 1 – if this is possible.

The theory behind our proposed algorithm is elementary and uses the concept of invariant subspaces for the MATLAB **eig** computed 'eigenvectors' of another associated flow matrix  $B(t_a)$  to find the coarsest simultaneous block structure for all flow matrices  $B(t_b)$  and consequently block-diagonalizes the given matrix flow A(t) or the given static matrix A itself.

The method works in  $O(n^2)$  time for all matrices  $A_{n,n}$  and all matrix flows A(t), be they real or complex, normal, with Jordan structures or repeated eigenvalues, and differentiable, continuous, or discontinuous.

We aim to discover unitarily diagonal-block decomposable matrices and flows from sensor given data. For unitarily block-diagonalisable A or A(t), the complexity of their numerical treatment decreases swiftly for  $O(n^3)$ matrix processes when working on each of their diagonal blocks separately.

### The proof : The Unitary Block-Decompositions and Fast and Accurate Field of Values Boundary Computations of all Square Matrices

The unitary block decomposition of matrix flows or single square matrices has never been resolved by algebraic means in 100 + years. Our new way of testing and establishing unitary decomposability of a matrix flow or a static matrix hinges on a numerical algorithm that decides the lay of the near zero entries and of the sizable ones in the hermitean Johnson matrix flow

$$\mathbf{F}(\mathbf{t}) = \mathbf{cos}(\mathbf{t})\mathbf{H} + \mathbf{sin}(\mathbf{t})\mathbf{K}$$

for  $0 \le t \le 2\pi$ .

[ We assume today that all matrices and matrix flows are real in this talk. ]

The algorithm starts from two linear independent Johnson flow matrices  $\mathbf{F}(t_1)$ and  $\mathbf{F}(t_2)$ . We form the 0-1 logic matrices  $Flogic(t_1)$  and  $Flogic(t_2)$  in Matlab's **spy** function and assemble the normalized eigenvectors of  $\mathbf{F}(t_1)$  in  $V'(t_1)$ so that adjacent rows in  $Flogic(t_1)$  have equal 0 and 1 patterned blocks.

We reorder the rows of  $Flogic(t_2)$  so that they have the same 0-1 pattern as  $Flogic(t_1)$  by using a combinatorial algorithm that establishes a joint proper unitary block decomposition of both logic 0-1 spy pattern matrices.

Going down from the top block leads to a joint unitary block diagonal structure for all F(t) with  $t \in [0, 2\pi]$ . Since F(0) = H and  $F(\pi/2) = K$ , the matrix A or the flow A(t) have the same unitary block structure as the hermitean matrices  $F(t_1)$  and  $F(t_2)$  do, and our algorithm is complete. This algorithm depends on the magnitude relations between the non-zero entries and the near zero entries in  $F(t_i)$ . To distinguish between computed entries as being 'zero' or definitely 'nonzero' is a an uncharted question. We have set the 'zero' threshold heuristically to  $||A||_{fro} \cdot 10^{-13} \approx ||A||_{fro} \cdot eps$ when working in double precision in Matlab.

To prove the main **Unitary Diagonability Theorem**, we assume that the hermitean Johnson flow F(t) contains two linearly independent matrices  $F(t_a) \neq F(t_b)$  where  $F(t) = \cos(t)H + \sin(t)K$  for the hermitean and skew parts  $H = (A + A^*)/2 = H^*$  and  $K = (A - A^*)/(2i) = K^*$  of A.

We know of no way to prove the unitary block-diagonalization theorem algebraically; an algebraic proof of our result was never attempted. An algorithmic proof was nearly impossible before the advent of computers and the understanding of matrix flows such as the Johnson flow  $\mathcal{F}(t)$  and of the predictive qualities and accuracy of Zhang Neural Networks.

The ability to construct the field of values F(A) boundary curve of some unitarily decomposable matrices quickly and accurately via shooting methods led to recent further studies of matrix eigencurve behavior by many who tried to locate their crossings or hyperbolic avoidances that had first appeared in original Quantum Physics studies in Copenhagen, Berlin and Göttingen in the 1920s. Albert Einstein shunned this fundamental Quantum Theory problem and rather worked on relativity. Bohr's group wanted to understand eigencurve crossings and their relation to the unitary matrix block-decomposability of matrices, but neither succeeded.

Eigencurve crossing studies for the unsolved unitary block-diagonalization problem began anew with Charlie Johnson's work on the field of values in the 1970s and advanced in the 1990s in Luca Dieci et al's, Loisel and Maxwell's, and my eigencurve papers. By 2020 several small sub-results of the unitary block-decomposition problem had been solved, but no complete classification had been found and the century old problem was still open in 2023.

Our final classification of unitary block-decompositions for both, general matrix flows and static matrices, was inspired by Yunong Zhang's parametervarying Neural Networks that date back to his doctoral thesis of 2001.

These recent developments are inter-woven and all have contributed to each other's solution and to the author's understanding of this cluster's new fundamental computational results. Each of the new results has advanced its area tremendously where no math methods had been found for decades, by researchers that relied on standard mathematical or computational approaches.

The mathematical process and its results were never predictable – until everything fell into place in 2023/24, was mathematically sound, proved, and their papers quickly accepted.

The **conditionally stable** Matrix Computational algorithms of this research cluster deliver **highly accurate results** at **high speed** and they do not suffer from the inaccuracies of our the traditional 'backward stable' methods

This is the end of my math talk with selections from the new extraordinary mathematical research cluster.

# What is our most urgent task today, in Linear Algebra and Matrix Theory, teaching and research wise ?

# What needs to be done immediately ?

We <u>must modernize our early College teachings in Linear Algebra and</u> enliven this area of Mathematics that helps us all so extraordinarily in our daily cell-phone and internet based lives and in our AI and engineering advances.

How do we, how do you mostly teach beginners' linear algebra classes today?

Which subjects do we, you, and I teach?

Using modern Matrix Theory based ideas **or** from a classical and algebraic standpoint ?

Top-down taught or taught interactively ? Th

That is the question

## Math Education : the Necessity of Modernizing the First Linear Algebra Course via coherent Matrix Theory Based Lesson Plans

This is the subject of a *serious Math Education session*, some time late at night, in Ithaka. All are welcome