SIAM J. MATRIX ANAL. APPL.
Vol. 41, No. 4, pp. 1528–1545

COALESCING EIGENVALUES AND CROSSING EIGENCURVES OF 1-PARAMETER MATRIX FLOWS*

FRANK UHLIG†

Abstract. We investigate the eigenvalue curves of 1-parameter Hermitian and general complex or real matrix flows $A(t)$ in light of their geometry and the uniform decomposability of $A(t)$ for all parameters t. The results by Hund and by von Neumann and Wigner in the 1920s for eigencurve crossings of "generic" Hermitian matrix flows $A(t) = (A(t))^*$ are clarified. A conjecture on extending these results to general nonnormal or non-Hermitian 1-parameter matrix flows is formulated and investigated. An algorithm to compute the block dimensions of uniformly decomposable Hermitian matrix flows is described and tested. The algorithm uses the Zhang neural network method to compute the time-varying matrix eigenvalue curves of $A(t)$ for $t_o \leq t \leq t_f$. Similar efforts for general complex matrix flows are described. This extension leads to many new and open problems. Specifically, we point to the difficult relationship between the geometry of eigencurves for general complex matrix flows $A(t)$ and a general flow's decomposability into block-diagonal form via one fixed unitary or general matrix similarity for all parameters t.

Key words. matrix eigenvalues, time-varying matrix flows, eigenvalue curve, eigencurve crossing, Hund–von Neumann–Wigner theorem, Zhang neural network, numerical matrix algorithm, decomposable matrix, block-diagonal matrix, block-diagonal matrix reconstruction

AMS subject classifications. 15A60, 65F15, 65F30, 15A18

DOI. 10.1137/19M1286141

1. Introduction.

The eigenvalues of real or complex matrix flows $A(t)_{n,n}$ have been studied for more than 90 years; see [9], [21] for the earliest papers that came about within foundational quantum theory. Parameter-dependent eigenvalue curves of Hermitian matrix flows and their possible crossings have become important for studies on stability and bifurcation such as for molecular aspects of quantum and chemical physics, in the study of energy surfaces, in structural analyses, in antenna theory, and in other areas; see, e.g., [16], [17], [10], or [15] and specifically [3, p. 520] for a listing of recent references. In the early 1900s it was most important for quantum mechanics to understand whether parameter-varying matrix eigenvalue curves would intersect or cross, leading to different practical results and quantum state implications if they would. The two fundamental papers by Hund [9] and von Neumann and Wigner [21] "proved" that Hermitian matrix flows that depend on one (or two) parameters, such as $A(t) = F + tG$ or $A(t) = F + tG + t^2H$ for constant Hermitian matrices F, G, and/or H, would not allow eigencurve crossings in what is now called the "generic" case.

This classical "result" has been repeated (see, e.g., [3, p. 519, second paragraph in the introduction]) often in the literature without much explanation of what the historic generic case means and why. "Genericity" originated in the quantum physics problems of [9] and [21], which studied "generically" indecomposable matrix flows for individual molecules and atoms under extreme pressure or heat variations. Quantum physics and quantum chemistry models naturally deal with parameter-dependent matrix flows that are indecomposable. I was made aware of the potential "nongeneric" behavior of

*Received by the editors September 9, 2019; accepted for publication (in revised form) by W.-W. Lin July 6, 2020; published electronically October 8, 2020.
https://doi.org/10.1137/19M1286141
†Department of Mathematics and Statistics, Auburn University, Auburn, AL 36849-5310 USA (uhligfd@auburn.edu).

1-parameter matrix flows when normal non-Hermitian matrices were named as exceptions to the Hund–von Neumann–Wigner no-crossing rule in [11, Ex. 7.1, Figure 7.2, pp. 1739, 1740] in computational studies of the field of values of a constant matrix. As the simplest and worst case counterexample for the classical generic result with Hermitian 1-parameter matrix flows, note that if a $2n$-by-$2n$ 1-parameter matrix flow $A(t) = F(t) + tG(t)$ is generated from two compatibly dimensioned block-diagonal Hermitian matrix flows $F(t) = \operatorname{diag}(f, f)_{2n,2n}$ and $G(t) = \operatorname{diag}(g, g)_{2n,2n}$ with repeated Hermitian n-by-n blocks $f(t)_{n,n}$ and $g(t)_{n,n}$, respectively, then every point on every eigencurve of $A(t)_{2n,2n}$ is doubly covered, i.e., every point of every eigencurve of $A(t)_{2n,2n}$ is a "crossing point" with a repeated eigenvalue.

For mathematics this simple nongeneric example produces many noteworthy challenges: Can there be eigenvalue curve crossings in more general 1-parameter matrix flow settings? When do they occur, if ever? How can they be found? Which is the coarsest block-diagonalization for a given 1-parameter general dense complex or Hermitian matrix flow $A(t)$? Which is the finest? Can the decomposition block sizes for a decomposable matrix flow $A(t)$ be determined from its eigencurves or not? Can an actual block-diagonalization of a decomposable 1-parameter matrix flow $A(t)$ be computed for all parameter values t?

The initial paper [21] by von Neumann and Wigner studied 1- to 3-variable Hermitian matrix flows in regards to eigencurve crossings. Stone [17] first approached their noncrossing rule through manifolds and topology, and Friedland, Robbin, and Sylvester [7] then classified generic sets of matrices that must contain matrices with repeated eigenvalues, i.e., experience eigencurve crossings. They did so via algebraic geometry and topological means. Subsequently Dieci et al. [2], [6] studied multiparameter Hermitian matrix flows via Schur forms, and Dieci, Papini, and Pugliese [3, Exs. 4.1, 4.2, pp. 533–535] created an algorithm for approximating coalescing points of eigenvalues for 2- or 3-parameter Hermitian matrix flows of the form $f(x, y, z)F + g(x, y, z)G + h(x, y, z)H$ with constant Hermitian matrices F, G, H and multinomials f, g, and h. Early computational efforts to compute the eigencurves of matrix flows are due to Kalaba, Spingarn and Tesfatsion [10], who differentiated the matrix eigenvalue, eigenvector equation and studied an ODE initial value path-finding method for Hermitian parameter-varying matrix flows.

The study of matrix flows' eigenvalue behavior was extended to that of their singular values by Bunse-Gerstner et al. [1], O'Neil [14], and most recently by Dieci and Pugliese [4, 5].

Here we consider 1-parameter Hermitian and general complex or real matrix flows $A(t) = (a_{k,j}(t))_{n,n}$ with 1-parameter varying entry functions. This extends the earlier $A(t) = F + tG + t^2H$ or $f(x, y, z)F + g(x, y, z)G + h(x, y, z)H$ approaches in two directions, namely, to general nonnormal matrix flows and to those with general entry functions. Throughout we assume that the individual entry functions of all considered matrix flows are continuous in order to be able to work with continuous eigencurves.

Section 2 deals with 1-parameter Hermitian or symmetric matrix flows $A(t)$ and their eigencurve crossings in the nongeneric case and relates the latter to equivalent separable or decomposable block matrix flows, where the term "equivalent" means uniformly via one fixed matrix similarity $S^{-1}A(t)S$ for all parameters t. An algorithm for finding the block dimensions of decomposable 1-parameter Hermitian flows and for finding a block-diagonal form that can be obtained uniformly for any specific decomposable Hermitian matrix flows is given. Open questions are raised. Section 3 deals with general complex or real 1-parameter matrix flows. Throughout we use the Zhang neural network (ZNN) method merely as a tool to compute matrix flow eigencurves. Currently this is the fastest and most accurate method for eigenanalyses of time-varying matrix flows; see [19] and [22, 20] for comparisons with Francis multi-shift QR and ODE path following methods. Note that any of the mentioned methods would be equally appropriate here and all would generate similar results.

2. Hermitian 1-parameter matrix flows and eigencurve crossings. This section deals with 1-parameter matrix flows $A(t)_{n,n}$, their eigenvalue curves for a time or parameter interval $t_o \leq t \leq t_f$, and the notion of matrix separability or matrix block-decomposition. More specifically we deal with Hermitian matrix flows in this section; general matrix flows and their eigencurves are discussed in section 3.

DEFINITION. (1) *A constant square matrix $A_{n,n}$ is called* separable *or* decomposable *if A is similar to a proper block-diagonal matrix,. Here and below "proper" means that A's block-diagonal representation has at least two diagonal blocks.*

(2) *A* 1*-parameter real or complex square matrix flow $A(t)_{n,n}$ is called* separable *or* decomposable *on an interval $[t_o, t_f] \subset \mathbb{R}$ if each $A(t)$ can be reduced uniformly to the same proper block-diagonal form via the same fixed matrix similarity.*

Note that an indecomposable or a decomposable matrix flow might contain specific matrices $A(t)$ that may be reduced further for a specific value of $t \in [t_o, t_f]$ if, for example, some strategic entries in the common block-diagonal form of all $A(t)$ become zero for some values of t.

Differing from our approach, Sibuya [16] has studied and classified nonuniform but local block-diagonalizable matrix flows $A(t)$ for which there exist block-diagonalizing similarities via $T(t)$ for every t so that $T(t)^{-1}A(t)T(t)$ is in block-diagonal form.

Obviously the eigencurves of a block-diagonal Hermitian matrix flow $A(t) = \text{diag}(A_1(t), A_2(t), \ldots, A_k(t))$ are simply the superpositions of the eigencurves of each of its individual matrix blocks $A_i(t)$. If $k = 2$ and the eigencurves of the first diagonal block $A_1(t)$ hover around 100 on a given interval $[t_o, t_f]$ and those of the second block $A_2(t)$ hover around -50, for example, then there will likely be no crossings among the set of eigencurves of $A(t)$ on the given interval, and therefore the eigencurves of $A(t) = \text{diag}(A_1(t), A_2(t))$ carry little information regarding the possible decomposability of $A(t)$. In this case it might be advisable to enlarge the interval $[t_o, t_f]$ since the decomposability of parameter-varying matrix flows is a global property. We will study and learn more about this phenomenon later on.

If, on the other hand, there are observed eigencurve crossings for a 1-parameter Hermitian matrix flow $A(t)_{n,n}$, then—according to the Hund–von Neumann–Wigner no-crossing rule for generic flows—$A(t)$ is separable or decomposable, i.e., there is a nonsingular constant matrix $S_{n,n}$ so that $S^{-1} \cdot A(t) \cdot S$ is uniformly and properly block diagonal for all t. Throughout we will use unitary similarities $S = U$ with $UU^* = I$ in order to not affect the eigenvalue conditioning of the matrix flows. All our programs are designed to work with ordinary similarities and also with unitary or orthogonal ones.

Note that once we know the eigencurve data for $A(t)_{n,n}$ approximately we can interpolate the eigenvalue curves and form a diagonal matrix flow $\tilde{A}(t) = \text{diag}(a_1(t), \ldots, a_n(t))$ with the individual eigencurve functions $a_i(t)$ of $A(t)$ in successive diagonal positions of $\tilde{A}(t)$. Therefore every matrix flow might potentially come from a completely decomposable, i.e., a diagonal, matrix flow. Therefore looking for the finest possible decomposition structure of matrix flows is futile. It would be more sensible to try and find the coarsest proper decomposition structure of a given 1-parameter varying matrix flow instead. This is a new and worthwhile question that combines function geometry with matrix analysis.

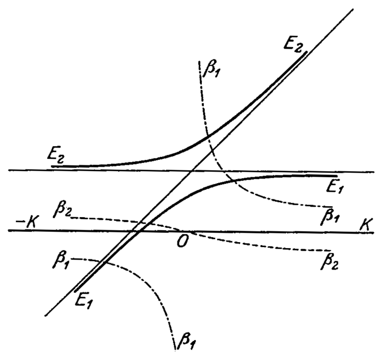

Fig. 1. *From* [21, p. 469].

The classical Hermitian matrix flow results of Hund [9] and von Neumann and Wigner [21], stated explicitly for indecomposable "generic" 1-parameter matrix flows, are as follows.

Hund–von Neumann–Wigner Theorem (HvNW; see [9, 21]). *If $A(t)$ is an indecomposable 1-parameter Hermitian matrix flow, then*

(a) *(or ¬(c)) the eigenvalue curves in $\mathbb{R}^2 \simeq \mathbb{C}$ of $A(t)$ do not intersect, and*

(b) *if two eigenvalue curves approach each other, they avoid crossing each other by veering off in a hyperbolic way where the approaching angle of either eigencurve equals the departing angle of the other eigenvalue curve after their close encounter.*

If two eigencurves of a 1-parameter Hermitian matrix flow $A(t)$ cross each other, then

(c) *(or ¬(a)) the matrix flow is uniformly decomposable by a constant matrix similarity.*

Clearly parts (a) and (c) are logically equivalent. They are stated here in their negated forms for clarity. Neither converse of (a) or (c) is true. For a proof of part (b) using Schrödinger's perturbation method and asymptotics, see [21, sect. 2, p. 469] and its Figure 1 (copied above). A similar process of the eigenvalue veering off was observed and studied for multiparameter eigencurves in [3], and coalescing eigenvalues were called "conical intersections of eigenvalues" there.

Throughout we use the fast and highly accurate Zhang neural network (ZNN) approach (see [22, 20, 19]) as our tool to plot the eigencurves of 1-parameter varying matrix flows $A(t)$ efficiently and well. From the eigencurve plots of $A(t)$ we then determine increasingly finer block-diagonal decomposition sizes in a new algorithm that computes the coarsest possible proper block-diagonalization for the given matrix flow. Our first example plots the real eigencurves for a Hermitian time-varying random entry matrix flow $A(t)_{11,11}$ as follows.

In Figure 2, the 11 real eigencurves for $A(t) = A(t)^*$ are labeled at $t_o = 0$ by their eigenvalues in descending order along the left edge. They are traced in differing colors until $t_f = 6$, where their curve labels are repeated for clarity along the right edge. A colored legend panel (color is available in the electronic version of this article) further helps with determining which eigencurve crosses which. We compute eigencurve crossings in our MATLAB code `Chermitmatrixfloweig.m`.[1] The crossings data is stored

[1] MATLAB codes for all main and auxiliary m-files are collected and referenced in http://www.auburn.edu/~uhligfd/m_files/Eigencurves/.

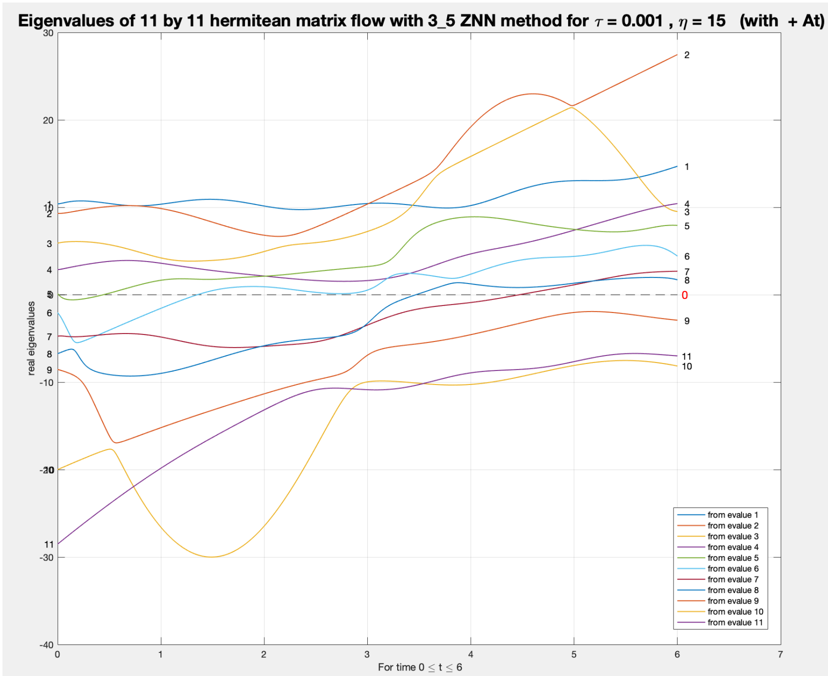

Fig. 2.

in an $n-1$-by-$n+1$ matrix R1 for n-by-n Hermitian matrix flows. For Figure 2, R1's first 5 columns are

Curve number	Crosses eigencurve with label			
1	2	3	0	0
2	0	0	0	0
3	4	0	0	0
4	5	6	0	0
5	0	0	0	0
6	7	0	0	0
7	8	0	0	0
8	0	0	0	0
9	0	0	0	0
10	11	0	0	0

Our algorithm then computes the vector $ve = (1, -1, -1, 1, -1, -1, 1, -1, 2, 3, -3) \in \mathbb{Z}^{11}$ from R1. This integer vector ve separates the eigencurves into crossing groups which fall into 5 sets here, as indicated by ve's five distinct entries 1, –1, 2, 3, and –3. The vector ve always starts with a 1 for the first (top left) starting curve, and all other entries in ve are initially set to zero. All eigencurves that the first eigencurve crosses are subsequently labeled with –1 in ve. If there are zero entries in ve afterwards, the algorithm starts from there for the indices 2 and –2 and indexes the eigencurves below in R1 as before. This process is repeated until all 11 eigencurves are labeled with positive or negative integers in succession in ve. If there are data clashes where a nonzero entry k in ve does not conform with a new crossing requirement of $-k$ in a position, this indicates the need for an additional eigencurve group. Therefore the integer label for this clashing entry in ve is upped to $k+1$ and all subsequent ve

entries in this row of R1 are reset to zero. The remaining crossing numbers in R1 for the current curve are skipped for a fresh start. Then we continue with the next row of R1. Eigencurves with opposite signed integer values in ve cross each other. Further details are listed in the % code line comments of `Chermitmatrixfloweig.m`.

We advise occasionally trying to create the decomposition data vector ve for a given Hermitian matrix flow by hand, using pencil and paper, the crossing matrix R1, and curve plot verifications to learn how the automatic code uses part (a) of the HvNW Theorem.

In our chosen 11-by-11 matrix flow example, and according to the currently best available eigencurve grouping vector

$$ve = (1, -1, -1, 1, -1, -1, 1, -1, 2, 3, -3),$$

there are altogether five diagonal blocks suggested, as three eigencurves carry a 1 and five a –1, meaning that those with label 1 cross those with label -1, and vice versa, while no others are crossed by or crossing these 8 eigencurves. Moreover there is a single eigencurve, the ninth eigencurve associated with the label 2 in ve, that is not crossing those above nor those below, as can readily be seen in R1 and in Figure 2. The ninth eigencurve is separate from any other eigencurve. Finally there are two eigencurves labeled 10 and 11 at the left edge of Figure 2 that cross one another and no others. They are given the labels 3 and –3 in positions 10 and 11 of ve. Thus far we have processed the eigencurve crossing data in R1 according to only part (a) of the HvNW Theorem.

Part (b) of the HvNW Theorem involves knowledge of "almost crossings," i.e., of hyperbolic near approaches and almost touching eigencurve pairs. These are currently identified through visual examination of the eigencurve plot such as depicted in Figure 2 here. We have not investigated how to accomplish this task computationally. Note that with $0 \leq t \leq 6$ in Figure 2, the eigencurve pairs with labels 2 and 3, 5 and 6, 6 and 8, as well as 9 and 10 avoid crossings in a hyperbolic fashion, and therefore each of these paired eigencurves must come from the same block in a diagonal block reduction of $A(t)$. If we enter their indices in rowwise increasing ordered pairs in the matrix

$$Touch = \begin{pmatrix} 2 & 3 \\ 5 & 6 \\ 6 & 8 \\ 9 & 10 \end{pmatrix}$$

and invoke the command `ve = almostTouch(ve,Touch)`, this will adjust the previously computed entries of ve accordingly. For this example `almostTouch` refines ve to $(1, -1, -1, 1, -1, -1, 1, -1, 2, 2, -2)$. Now we have reduced the possible block-decomposition for the matrix flow $A(t)$ from five blocks associated with five labels 1, –1, 2, 3, and –3 to just four, namely 1, –1, 2, and –2, and we might still be able to improve our knowledge of the coarsest block-diagonal structure of $A(t)$ under similarities by looking at larger or different time intervals and thereby learn more about additional eigencurve crossings and/or new almost touching eigencurve pairs.

The eigencurve plots in Figure 3 for the same matrix flow $A(t)_{11,11}$ over the enlarged time interval $-7 \leq t \leq 6$ now create eigencurves of $A(t)$ with six hyperbolic avoidances for the curves labeled 1 and 3, 3 and 4, 4 and 6, 6 and 7, 7 and 9, and 9 and 10. These label pairs are again stored in an almost touching matrix *Touch* for the enlarged interval. Calling `Chermitmatrixfloweig.m` for $A(t)$ and the enlarged time interval creates a new eigencrossing data matrix R1 since the left time edge curve

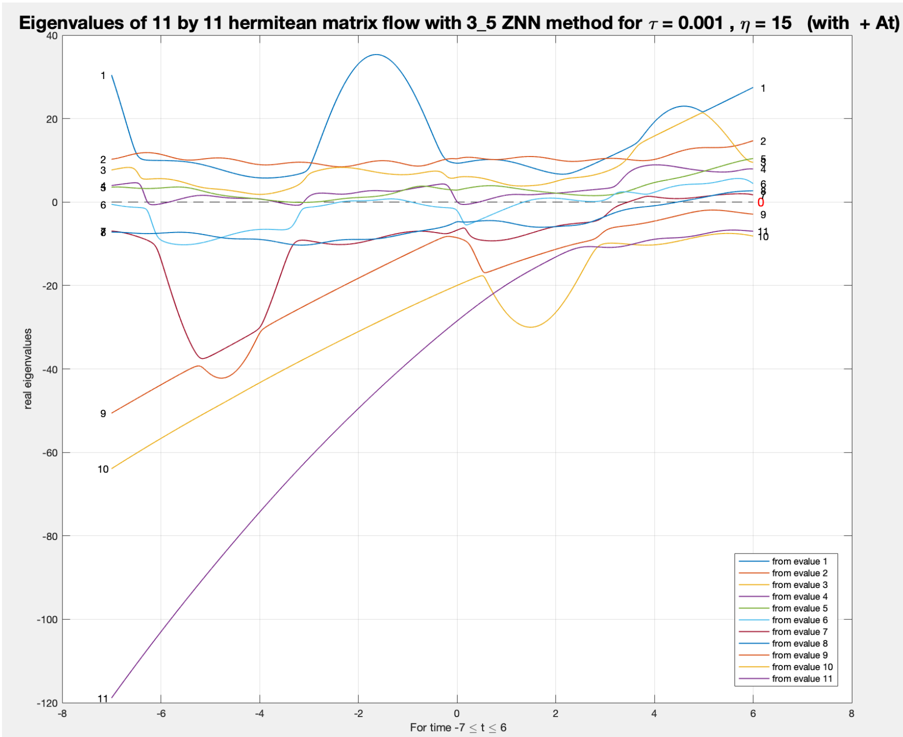

Fig. 3.

order obviously changes between $t_o = 0$ and $t_o = -7$ as new crossings occur on the leftward extended time interval:

Curve number	Crosses eigencurve with label			
1	2	0	0	0
2	3	0	0	0
3	5	0	0	0
4	5	0	0	0
5	6	0	0	0
6	8	0	0	0
7	8	0	0	0
8	0	0	0	0
9	0	0	0	0
10	11	0	0	0

From our new R1 and the respective new almost touching matrix *Touch*, a new coarsest decomposition into indecomposable diagonal blocks of $A(t)$ is computed in the vector $ve = (1, -1, 1, 1, -1, 1, 1, -1, 1, 1, -1)$ by our algorithm. Accordingly the composable matrix flow $A(t)$ can—at least in theory—be decomposed uniformly and most coarsely under one similarity into two irreducible diagonal blocks of dimensions 4 and 7, as the four $+1$ entries and the seven -1 entries in ve indicate.

Regarding speed and accuracy running `Chermitmatrixfloweig(11,-7,6,5,7, 0.0005,50,1,1)` on the interval $[t_o, t_f] = [-7, 6]$, the ZNN computed eigenvalues agree to six leading digits with the eigenvalues of $A(6)$ at the time interval end $t_f = 6$ when we use the convergent look-ahead finite difference formula `5_7`. The computations take around 8.3 seconds. Figure 2 for the eigencurves on the reduced time interval $[0, 6]$ was drawn by the slightly less accurate ZNN method `3_5` in under 2 seconds. For the sampling gap of $\tau = 1/2000$ in Figure 3, our ZNN-based method `5_7` evaluates 26,000 intermediate eigendata sets between $t_o = -7$ and $t_f = 6$. In each time step, ZNN computes one recursive update and solves one linear equation without ever using any matrix eigencomputations except to obtain the initial startup data. For an explanation of ZNN's speed, its look-ahead difference formulas, and its general workings, see [19, 18].

Can our dense complex Hermitian matrix flow example $A(t)$ be decomposed uniformly into a direct sum of a 4-dimensional and a 7-dimensional indecomposable diagonal block as predicted above? How was the matrix flow $A(t)$ constructed? Can a coarsest block-diagonalization be reconstructed from a given dense Hermitian matrix flow $A(t)$ and its eigencurves somehow, if at all?

To form the dense 11-by-11 complex Hermitian test matrix flow example $A(t)$, we started with a 7-by-7 single parameter indecomposable general matrix flow $B_1(t)$ with complex function entries and appended it to become 11-by-11 block diagonal via the MATLAB command `B2(t) = blkdiag(B1(t),2*B1(t)(2:5,2:5))`. Then we transformed the resulting general matrix flow $B_2(t) \in \mathbb{C}^{11,11}$ to become the complex Hermitian matrix flow $B(t) = B_2(t) + B_2(t)^*$. Finally, we obscured the original block-diagonal form, composed of one indecomposable 7-dimensional diagonal block and one indecomposable 4-dimensional one, by replacing $B(t)$ with $A(t) = U^* B(t) U$ for a fixed complex random entry 11-by-11 unitary matrix U. The matrix flow $A(t)$ that we used in our code thus was both complex Hermitian and dense. The matrices $A(t)$ give no sign of decomposability. Yet our method has computed the indecomposable coarsest block-diagonal structure for $A(t)$ correctly by relying on the crossing geometry of its eigencurves and our code.

Our MATLAB code `Chermitmatrixfloweig.m` together with `almostTouch.m` and a bit of eyeballing for "almost touching" hyperbolic crossings retrieved the hidden block-diagonal structure of this dense complex Hermitian matrix flow readily once the time interval was chosen large enough. If we had eyeballed wrongly, `almostTouch.m` is designed to recognize such errors, and it would indicate which row in *Touch* is erroneous. In this case we recommend removing the row of offending curve labels from *Touch* and retrying.

Our second example comes from a discussion of a 6-by-6 real symmetric test matrix flow and eigencurve crossings that appeared in 2010 on *Mathematica*'s stack-exchange.com site.[2] The poster, Jack S., suggests the symmetric matrix flow

$$B(t) = \begin{pmatrix} 21t + 1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 7t + 1/2 & 0 & 0 & 7 \cdot 2^{1/2} t & 0 \\ 0 & 0 & 1/2 - 7t & 0 & 0 & 7 \cdot 2^{1/2} t \\ 0 & 0 & 0 & 1/2 - 21t & 0 & 0 \\ 0 & 7 \cdot 2^{1/2} t & 0 & 0 & 14t - 1 & 0 \\ 0 & 0 & 7 \cdot 2^{1/2} t & 0 & 0 & -14t - 1 \end{pmatrix}_{6,6}$$

and notices that "Sometimes the eigenvalues may cross each other, but I want to make sure the right eigenvalue stays associated with its own state." The site studies eigenvalue analyses and their potential tracking failures in parameter-varying matrix

[2]See "Tracking Eigenvalues through a Crossing," https://mathematica.stackexchange.com/questions/165167/tracking-eigenvalues-through-a-crossing.

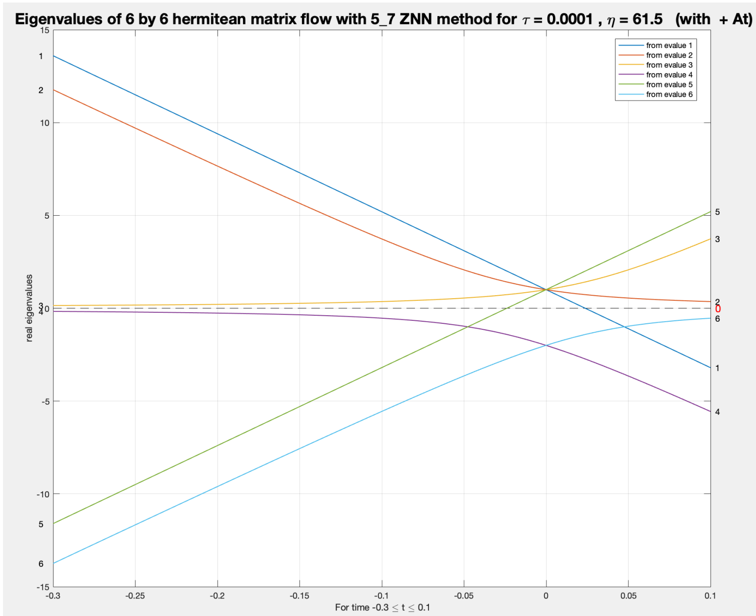

FIG. 4.

problems. ODE path-continuing methods are offered, as well as characteristic polynomial approaches and a simple block-diagonalization of the original matrix flow $B(t)$, followed by computing the parameter-varying eigenvalues for each separate diagonal block. We use ZNN and our algorithm, with $B(t)$ again camouflaged as a dense real symmetric matrix flow $A(t) = U^T B(t)U$ for a random orthogonal entry matrix U.

A call of `Chermitmatrixfloweig(6,-.3,.1,5,7,0.0001,61.5,1,1)` takes half a second and achieves 14 accurate leading digits for all computed eigenvalues of $A(t)$ at $t = t_f = 0.1$. The eigencurve crossings output is stored in R below.

Curve number	Crosses eigencurve with label				
1	2	3	5	6	0
2	3	5	0	0	0
3	5	0	0	0	0
4	5	6	0	0	0
5	0	0	0	0	0

The eigencurves are graphed in Figure 4.

The above call computes $ve = (1, -1, 2, 2, -2, -2)$. Note that the graph in Figure 4 shows no almost touching eigencurve behavior. By looking at the label multiplicities in ve, we note that $A(t)$ can be reduced by an orthogonal similarity to a block-diagonal matrix composed of four indecomposable blocks, namely, two of size 1-by-1 corresponding to the labels 1 and -1, and two of size 2-by-2 for the repeated labels 2, 2 and -2, -2. This conforms well with $A(t)$'s origin in $B(t)$, which assumes precisely this block-diagonal structure after applying a simple permutation similarity to $B(t)$.

Finally we check our algorithm with the real diagonal 5-by-5 matrix flow seed

$$B(t) = \text{diag}(\sin(1-1/2t), 1/2\cos(1/3t), \sin(t)\cos(-1-0.2t), \cos(2t-1/2), \cos(1+3t)^2)$$

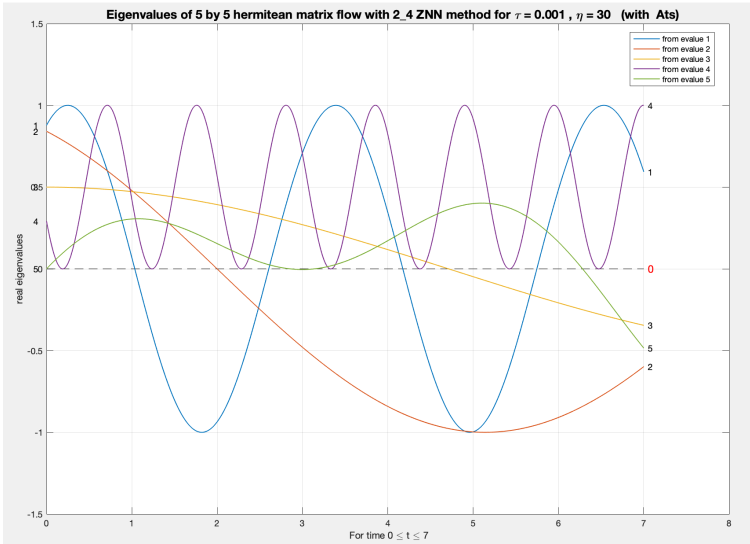

Fig. 5.

after it has been transformed by an orthogonal random entry matrix $U_{5,5}$ into the dense real symmetric matrix flow $A(t) = U^T B(t) U$. Figure 5 shows no almost touching hyperbolic evasions and dozens of eigencurve crossings, as HvNW predicts.

The computed eigencurve crossing data matrix R1 and the block-diagonalization vector ve for this example are

Curve number	Crosses eigencurve with label			
1	2	3	4	5
2	3	4	5	0
3	4	5	0	0
4	5	0	0	0

and $$ve = \begin{pmatrix} 1 \\ -1 \\ 2 \\ -2 \\ 3 \end{pmatrix}.$$

Clearly `Chermitmatrixfloweig` resolves this eigencurve and flow diagonalability problem perfectly in ve, which contains 5 distinct labels without repetition for our 5-by-5 symmetric flow problem, i.e., diagonalization is the coarsest proper indecomposable block decomposition for $A(t)_{5,5}$.

We close the Hermitian matrix flow eigencrossings section with a set of open questions.

Question 2.1. *What is the actual "coarsest block-diagonal form" of a Hermitian matrix flow $A(t)$ depending on its eigencurve crossing geometry? Can our method compute the coarsest block-diagonal form reliably?*

As we realize, any given eigencurve crossings data may in fact be generated from an underlying diagonal matrix flow $A(t)$ where every diagonal entry contains one respective eigencurve generating function. We would call such a diagonal decomposition the "finest block-diagonalization." This may or may not be achievable from the given dense Hermitian matrix flow $A(t)$. Instead, our algorithm tries to reduce the number of possible diagonal blocks in its computations by running through all eigencurve crossings in turn. A large number of distinct entries in ve signifies a relatively "fine coarsest block-decomposition." When the algorithm terminates after a potentially complete *Touch* matrix incorporation, we believe—or at least hope—that we have gained insight into the "coarsest block-diagonalization" of $A(t)$.

And thus our question really is, are we done then? Can we be sure that this algorithm is complete in the sense that it can deal correctly with all possible eigencurve crossing data matrices R1? Which matrices R1 are possible, and which are impossible, to achieve as eigencurve crossing matrices? How can these questions be answered mathematically and logically? Is knowing the eigencurve crossing matrix R1 sufficient to solve this problem? Of course not; the eigencurve crossing data matrix R1 is totally insufficient for indecomposable n-by-n Hermitian matrix flows whose eigencurves are widely separated over the reals and never cross. This situation would make our algorithm's output vector $ve = (1, 2, 3, \ldots, n)$ with n distinct integers and indicate 1-by-1 block diagonalability in error. What other data would be needed in this worst-case scenario? Do such matrix flows even exist?

We do not know the answers to these questions.

3. General 1-parameter complex and real matrix flows and their eigencurves. In this section we depict and study general complex and real 1-parameter varying matrix flows $A(t)$ that are neither Hermitian nor real symmetric. Such less restricted matrix flows generally give us "wild and woolly" 3-dimensional images in their eigencurve plots such as shown in Figure 6. This figure's nonnormal complex matrix flow $C(t)_{11,11}$ was built from our earlier complex 7-by-7 seed matrix $B_2(t)$ without the modifications to create a Hermitian flow whose real eigencurves were depicted in 2-dimensional $\mathbb{R}^2$ in Figures 2 and 3 earlier. Then $H(t) = U^*C(t)U$ is a dense general nonnormal 11-by-11 complex matrix flow for any dense "camouflaging" unitary matrix U. To plot Figure 6 we used the general complex flow version m-file `Cmatrixfloweig.m` that is again based on the ZNN method for speed and accuracy. For such general flows $H(t)$, $\mathbb{R}^3$ describes the parameter or time t on one axis and the real and imaginary eigenvalue parts of $H(t)$ in the perpendicular plane at t. When plotting time-varying complex eigencurves in $\mathbb{R}^3$, it is very unlikely that 1-dimensional eigencurves will ever meet or cross—unless, of course, the chosen complex matrix flow has a repeated block such as $A(t) = \mathrm{diag}(C(t), C(t))$, for example.

For "general" decomposable complex flows we have never observed eigencurve crossings which are standard with decomposable Hermitian flows. From the depicted eigencurves of Figure 6, it appears impossible to assert whether or not the flow $H(t)$ allows a 4-by-4 and 7-by-7 block-decomposition, despite its very creation as a decomposable flow. There is no crossing matrix to construct for generic non-Hermitian complex matrix flows, or almost never, since there literally appear to be no actual crossings in $\mathbb{R}^3$ for complex 1-parameter general matrix flows. Our general complex matrix flow code `Cmatrixfloweig.m` instead checks on the minimal distances between individual eigencurves as a function of t for decreasing distances of 1, 0.01, 0.0001, 0.000001, and even smaller. Regarding eigencurve distance minima we have never encountered random entry complex flow examples where two eigencurves got as close as 10^{-6} or 10^{-10} units.

How are the eigencurve geometry and the decomposability of general nonnormal complex matrix flows related, if at all? We could find no literature and at first had difficulties to conceive of, let alone construct, suitable complex 1-parameter matrix flows whose eigencurves in $\mathbb{R}^3$ might exhibit the two crossing conditions (a) and (b) of the HvNW Theorem for Hermitian matrix flows.

Eventually we were able to construct a nonnormal 1-parameter decomposable general 10-by-10 complex matrix flow $A(t)$ whose eigencurves exhibit HvNW-like be-

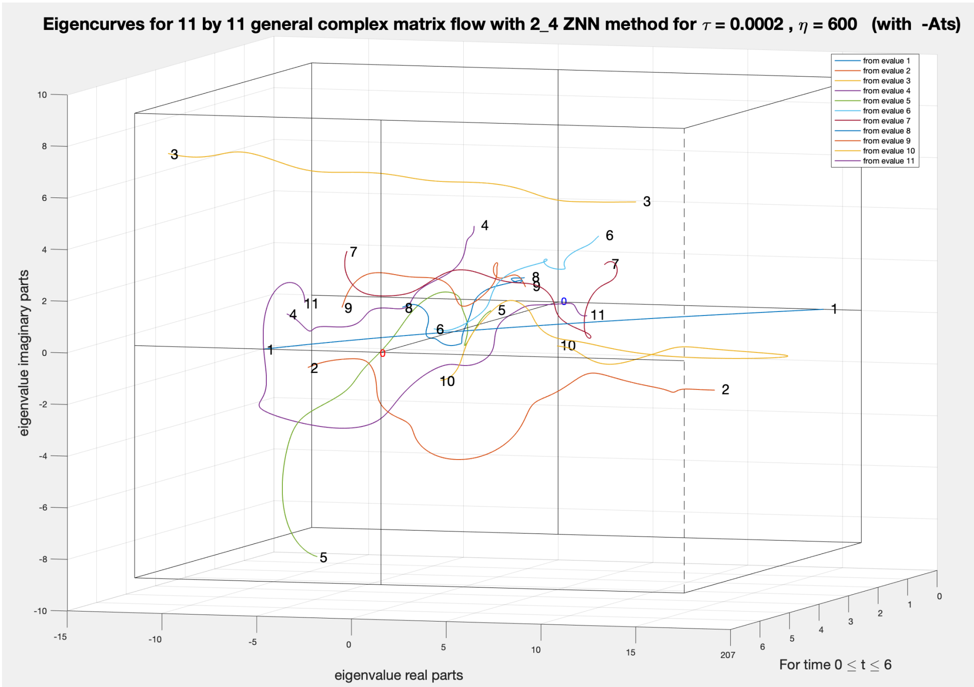

FIG. 6.

havior. The set of Figures 7, 8, and 9 and further discussions below suggest that the two results (a) and (b) of Hund [9] and of von Neumann and Wigner [21] for Hermitian matrix flows may hold for all general complex and real 1-parameter matrix flows. A proof thereof is beckoning.

This nonnormal complex matrix flow is composed of two indecomposable tridiagonal complex matrix flows and their block-diagonal join. One of these is the 4-by-4 nonnormal complex matrix flow

$$A4(t)=\begin{pmatrix} i(2-e^{t-1})+t/6 & 1 & 0 & 0 \\ 1 & -2-2i\sin(t-1) & 1 & 0 \\ 0 & 1 & 2i-2t & 1 \\ 0 & 0 & 1 & \sin(t+2)+it, \end{pmatrix},$$

the second one is 6-by-6,

$$A6(t)=\begin{pmatrix} i-2\cos(2t) & 1 & 0 & 0 & 0 & 0 \\ 1 & -2-2i\sin(t-1) & 1 & 0 & 0 & 0 \\ 0 & 1 & 2i-t & 1 & 0 & 0 \\ 0 & 0 & 1 & ie^{\sin(t)} & 1 & 0 \\ 0 & 0 & 0 & 1 & t/2+\sin(t)\cos(2it)/100 & 1 \\ 0 & 0 & 0 & 0 & 1 & t-i/8\cos(it/3-1) \end{pmatrix},$$

and $A10(t)$ is the concatenated array

$$A10(t) = \begin{pmatrix} A6(t)_{6,6} & O_{6,4} \\ O_{4,6} & A4(t)_{4,4} \end{pmatrix}_{10,10} .$$

Each of the above complex tridiagonal matrix flows is again made into a dense flow via fixed unitary similarity transformations $B_{..} = U_{..}^{*}A_{..}(t)U_{..}$ before analyzing

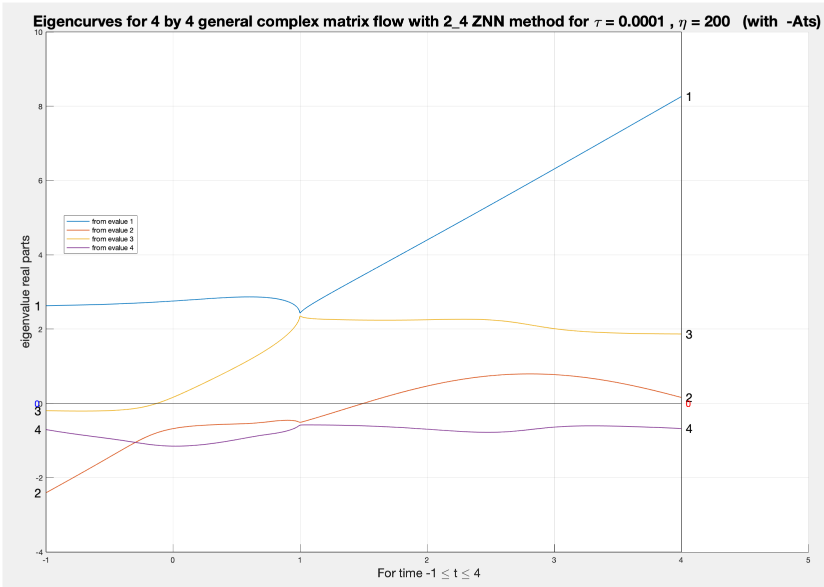

FIG. 7.

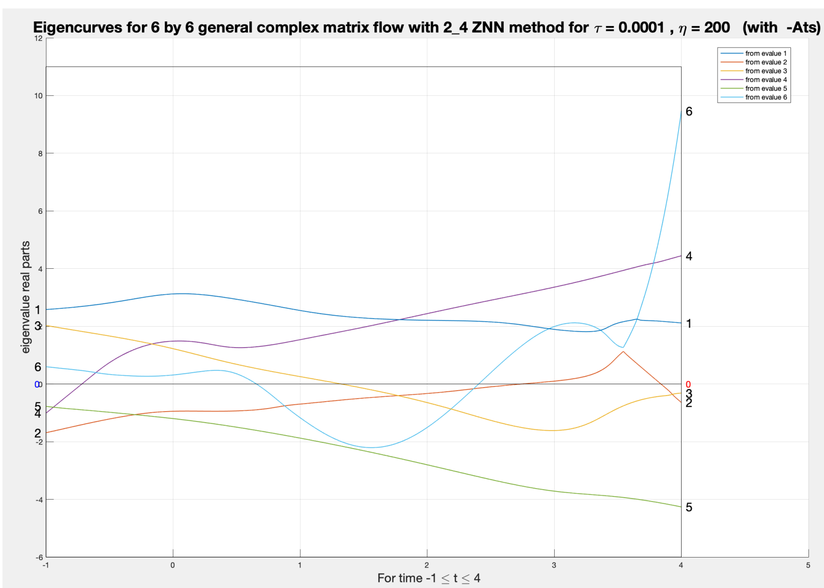

FIG. 8.

and plotting, with matrices $U_{..}$ of compatible dimensions 4-by-4, 6-by-6, or 10-by-10, respectively.

Figure 7 shows the eigencurves of $B4(t)$ when projected onto the eigenvalue real parts and time plane. It also shows "eigencurve avoidance" for the curve pair 1 and 3 and for the pair 2 and 4 for the complex nonnormal flow $B4(t)$ near $t = 1$.

Figure 8 shows a similar plot for $B6(t)$, again with "hyperbolic avoidance" now near $t = 3.5$.

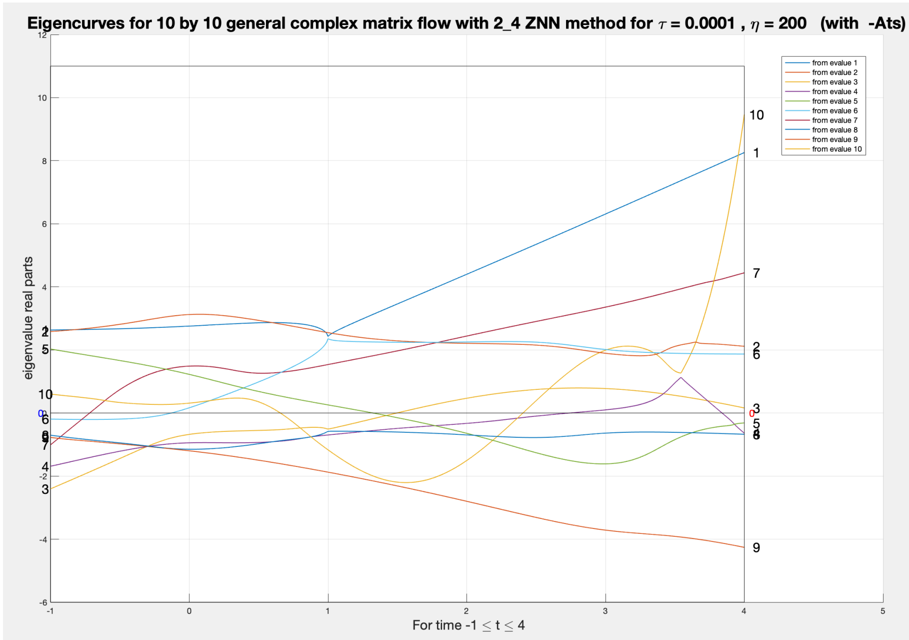

FIG. 9.

TABLE 1
Equivalent eigencurve number translations.

	In Figure 10 for $B6(t)$	In Figure 11 for $B10(t)$			In Figure 9 for $B4(t)$		
			1	$\to 4$	$1 \leftarrow 4$	alm. touch 6	–|
	$1 \to 2$	$1 \leftarrow$	2				|
			3	$\to 3$	$3 \leftarrow 3$	alm. touch 8	—|
|– alm. touch 6	$2 \to 4$	$2 \leftarrow$	4				| |
|	$3 \to 5$	$3 \leftarrow$	5				| |
|			6	$\to 6$	$6 \leftarrow 2$	alm. touch 1	–| |
|	$4 \to 7$	$4 \leftarrow$	7				|
|			8	$\to 1$	$8 \leftarrow 1$	alm. touch 3	—|
|			X				
|	$5 \to 9$	$5 \leftarrow$	9				
|– alm.touch 2	$6 \to 10$	$6 \leftarrow$	10				

The combined, superimposed eigencurve plot for the block-diagonal matrix flow $B10(t)$, again projected onto the real parts and time plane, is given in Figure 9.

The eigencurve "near touchings" for both $B4$ and $B6$ are visible in the eigencurve plot for their concatenated flow $B10(t)$. The Rc eigencurve nearness data of $A10$ was computed after generating $B10(t)$ in `formA10tricompl` via `Cmatrixfloweig(10,-1,4,2,4,0.0001,200,1,-1)`. The nearness data of general complex flows is stored in Rc. The data in Rc for our example indicates that curves 8 and 9 in Figure 9 get to within almost 10^{-2} units of each other, indicating that these two curves are close to crossing. But they show no signs of "hyperbolic avoidance." For help with reading the three plots in Figures 7 through 9, Table 1 shows an equivalence list for the respective plot number labels and the eigencurves' "almost touching" hyperbolic avoidance behavior.

Near time $t_1 = 0.8174$ the eigencurves for $B10(t_1)$ with labels 8 and 9 pass each other in the real parts, imaginary parts, and time space $\mathbb{R}^3$ at a $7.6 \cdot 10^{-3}$ distance. After subtracting $(.007.64534340607459 + .00011384079651410 86i) \cdot I_6$ from the first 6-by-6 tridiagonal block of $B10$, the eigenvalues of the resulting eigencurves with labels 8 and 9 of the modified complex matrix flow $B10eps(t_1)$ agree in their 15 leading digits, and for all practical purposes we have constructed a decomposable complex nonnormal matrix flow with an eigencurve crossing.

Looking at the imperceptibly different eigencurve plot for $B10eps(t)$ as displayed in MATLAB's plot window exhibits the three, now slightly shifted "almost crossings" of $B10$ clearly. The two eigencurves labeled 8 and 9 now cross each other for $B10eps(t)$ and continue without interference, exemplifying HvNW's conditions (b) and (c) for general complex 1-parameter matrix flows and insinuating that the modified flow $B10eps(t)$ decomposes into at least two, possibly indecomposable diagonal blocks. Note that eigencurve 8 for the modified nonnormal complex flow $B10eps(t)$ comes from $B4$, and eigencurve 9 derives from the modified flow $B6$ according to our eigencurve number translations list above.

Here, differing from the Hermitian 1-parameter matrix flow case, we have not been able to assert the actual number of coarsest blocks or their sizes.

An interesting example from [10] is the real nonnormal matrix flow

$$A(t) = U^T \begin{pmatrix} 1 & t \\ t^2 & 3 \end{pmatrix} U$$

for a fixed random entry real orthogonal 2-by-2 matrix U. $A(t)$'s two eigenvalues are complex conjugates for $t < -1$; they double up as 2 and 2 at $t = -1$ for an eigencurve crossing and are distinct real for $t > -1$, forcing its eigencurves to make right angle turns at their crossing point $t = -1$. As this general matrix flow is indecomposable, it violates part (a) of a generalized HvNW Theorem. Lui, Keller, and Kwok [12] have studied homotopy methods to plot eigencurves of general matrix flows $A(t) = (1-t)A_0 + tA_1$ and found similar eigenpath behavior at real/complex bifurcation points. See also Heiss [8] and Marcus and Parilis [13].

An accurate picture is plotted by the MATLAB call of `Cmatrixfloweig(2,-1.006,0,3,3,0.002,30,1,1)` in Figure 10 for $-1.006 \le t \le 0$. However, moving the starting time further back to $t_o = -2$ gives an incomplete picture.

Figure 11 shows only one, now doubled, eigencurve for $t > -1$ when computed via ZNN. We have observed similar behavior with eigencurve pairs for nonnormal real matrix flows that transit from being complex conjugates to being real, or vice versa. Sometimes a curve pair stays intact, coming in as a complex conjugate eigenvalue pair and leaving as a real pair. Sometimes the new eigencurve branches overlay and collapse into one, and sometimes two formerly doubled-up complex conjugate or real eigencurve pairs proceed as two separate ones after such 90° direction flips. The reason for this behavior of ZNN eigenmethods eludes us. Similar glitches may occur with ODE IVP path continuation methods for nonnormal real matrix flows, or they may not; refer to [12, 8], for example. We are also reminded of Markus and Parilis's result in [13] that slight perturbations of a static matrix A with a repeated eigenvalue λ or with repeated Jordan blocks for the same eigenvalue λ may only result in coarser, larger Jordan blocks for the perturbed eigenvalue $\tilde{\lambda}$ and never in finer Jordan structures for a perturbed $\tilde{A}$.

Based on the second 10-by-10 example of this section and—with trepidation regarding the validity of part (a)—we conjecture a generalization of the HvNW Theorem [9, 21].

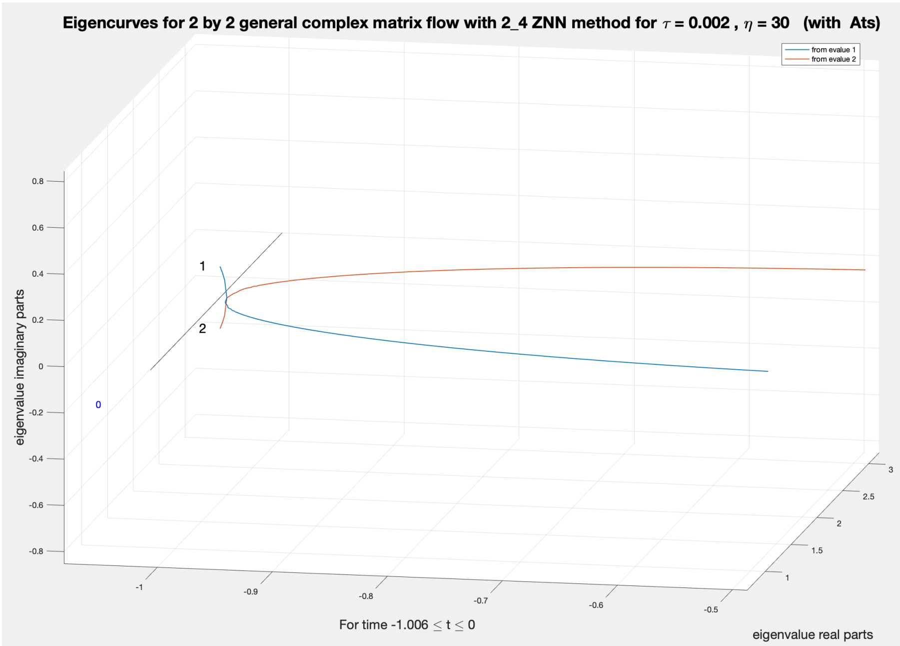

FIG. 10.

GENERALIZED HUND–VON NEUMANN–WIGNER THEOREM (a conjecture). *If $A(t)$ is an indecomposable general time-varying complex or real matrix flow, then*

(a) *the eigencurves of $A(t)$ in the 3-dimensional real part, imaginary part, and time space $\mathbb{R}^3$ do not intersect;*

(b) *if two eigencurves of $A(t)$ approach each other, they veer off in a hyperbolic way where the approaching space angle of either eigencurve equals the departing space angle of the other after their close encounter.*

If two eigencurves of a 1-parameter general complex matrix flow $A(t)$ cross each other, then

(c) *the matrix flow is uniformly decomposable by a constant matrix similarity.*

Again, statements (a) and (c) above are equivalent logically and are stated only for clarity. Their converses do not hold.

Here are several related open questions.

QUESTION 3.1. *Is a generalized HvNW conjecture true? How can it be proved for general matrix flows?*

QUESTION 3.2. *Given a 1-parameter varying complex nonnormal and dense matrix flow $A(t)$, how can we assess its uniform decomposability from its eigencurve graphs in $\mathbb{R}^3$? In 3-dimensional space it is relatively rare for two spatial curves to cross. This rarity is due to the third degree of freedom here when compared with assessing the decomposability of Hermitian matrix flows from their eigencurve behavior, which can be deduced entirely from $\mathbb{R}^2$; see section 2. How can we develop computational methods to find eigencurve crossings and details of a flow's decomposability from its 3-dimensional eigencurve data? Is it enough, for example, to extend the parameter interval of the plots as done for Hermitian flows?*

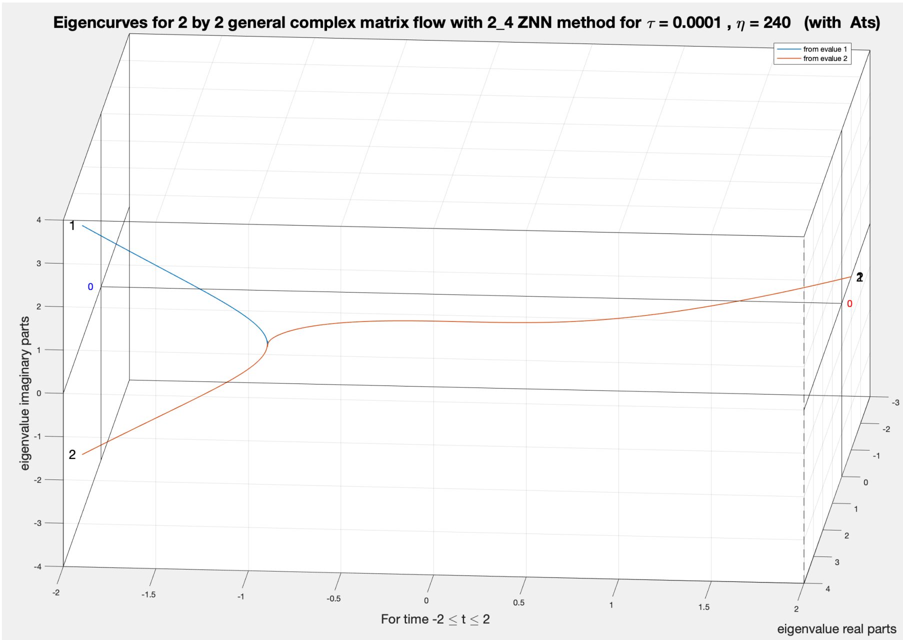

FIG. 11.

QUESTION 3.3. *Are there any other criteria or data beyond eigencurve data that can help determine the uniform block-diagonalability for a general complex matrix flow and yield its block dimensions? What could they be?*

QUESTION 3.4. *What lies behind the occasional doubling up of real matrix flow $A(t)$ eigencurve pairs that transit from being complex conjugate to real or in the reverse direction when evaluated via ZNN eigenmethods?*

Acknowledgments. I am thankful to Nick Trefethen's mention of "decomposing matrix flows" when I was visiting his group in Oxford in May 2019. His comment sent me thinking differently and more deeply, which helped me to develop the algorithms further and to formulate the conjecture. I am also grateful for the comments of the referees, who narrowed my focus to the "generic" indecomposable case studies and extensions in the literature.

REFERENCES

[1] A. BUNSE-GERSTNER, R. BYERS, V. MEHRMANN, AND N. K. NICHOLS, *Numerical computation of an analytic singular value decomposition of a matrix valued function*, Numer. Math., 60 (1991), pp. 1–39.

[2] L. DIECI AND T. EIROLA, *On smooth decompositions of matrices*, SIAM J. Matrix Anal. Appl., 20 (1999), pp. 800–819, https://doi.org/10.1137/S0895479897330182.

[3] L. DIECI, A. PAPINI, AND A. PUGLIESE, *Approximating coalescing points for eigenvalues of Hermitian matrices of three parameters*, SIAM J. Matrix Anal. Appl., 34 (2013), pp. 519–541, https://doi.org/10.1137/120898036.

[4] L. DIECI AND A. PUGLIESE, *Singular values of two-parameter matrices: An algorithm to accurately find their intersections*, Math. Comput. Simul., 79 (2008), pp. 1255–1269.

[5] L. DIECI AND A. PUGLIESE, *Two-parameter SVD: Coalescing singular values and periodicity*, SIAM J. Matrix Anal. Appl., 31 (2009), pp. 375–403, https://doi.org/10.1137/07067982X.

[6] L. Dieci and A. Pugliese, *Hermitian matrices depending on three parameters: Coalescing eigenvalues*, Linear Algebra Appl., 436 (2012), pp. 4120–4142.
[7] S. Friedland, J. W. Robbin, and J. H. Sylvester, *On the crossing rule*, Comm. Pure Appl. Math., 37 (1984), pp. 19–37.
[8] W. D. Heiss, *Exceptional points of non-Hermitian operators*, J. Phys. A, 37 (2004), pp. 2455–2464.
[9] F. H. Hund, *Zur Deutung der Molekelspektren.* I., Z. Phys., 40 (1927), pp. 742–764.
[10] R. Kalaba, K. Spingarn and L. Tesfatsion, *Individual tracking of an eigenvalue and eigenvector of a parametrized matrix*, Nonlinear Anal., 5 (1981), pp. 337–340.
[11] S. Loisel and P. Maxwell, *Path-following method to determine the field of values of a matrix at high accuracy*, SIAM J. Matrix Anal. Appl., 39 (2018), pp. 1726–1749, https://doi.org/10.1137/17M1148608.
[12] S. H. Lui, H. B. Keller, and T. W. C. Kwok, *Homotopy method for large, sparse, real nonsymmetric eigenvalue problem*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 312–333, https://doi.org/10.1137/S0895479894273900.
[13] A. S. Markus and E. E. Parilis, *The change of the Jordan structure of a matrix under small perturbations*, Linear Algebra Appl., 54 (1983), pp. 139–152.
[14] K. A. O'Neil, *Critical points of the singular value decomposition*, SIAM J. Matrix Anal. Appl., 27 (2005), pp. 459–473, https://doi.org/10.1137/040611719.
[15] K. R. Schab, J. M. Outwater, M. W. Young, and J. T. Bernhard, *Eigenvalue crossing avoidance in characteristic modes*, IEEE Trans. Antennas Propag., 64 (2016), pp. 2617–2627.
[16] Y. Sibuya, *Some global properties of matrices of functions of one variable*, Math. Ann., 161 (1965), pp. 67–77.
[17] A. J. Stone, *Spin-orbit coupling and the intersection of potential energy surfaces in polyatomic molecules*, Proc. Royal Soc. London A Math. Phys. Sci., 351 (1976), pp. 141–150.
[18] F. Uhlig, *The construction of high order convergent look-ahead finite difference formulas for Zhang neural networks*, J. Difference Equ. Appl., 25 (2019), pp. 930–941, https://doi.org/10.1080/10236198.2019.1627343.
[19] F. Uhlig, *Zhang neural networks for fast and accurate computations of the field of values*, Linear Multilinear Alg., (2019), https://doi.org/10.1080/03081087.2019.1648375.
[20] F. Uhlig and Y. Zhang, *Time-varying matrix eigenanalyses via Zhang Neural Networks and finite difference equations*, Linear Algebra Appl., 580 (2019), pp. 417–435, https://doi.org/10.1016/j.laa.2019.06.028.
[21] J. von Neumann and E. P. Wigner, *On the behavior of the eigenvalues of adiabatic processes*, Phys. Zeitschrift, 30 (1929), pp. 467–470.
[22] Y. Zhang, M. Yang, C. Li, F. Uhlig, and H. Hu, *New continuous ZD model for computation of time-varying eigenvalues and corresponding eigenvectors*, submitted for publication.