A FAST ITERATIVE ALGORITHM FOR NEAR-DIAGONAL EIGENVALUE PROBLEMS[*]

MASEIM KENMOE[†], RONALD KRIEMANN[‡], MATTEO SMERLAK[‡], AND ANTON S. ZADORIN[‡]

Abstract. We introduce a novel eigenvalue algorithm for near-diagonal matrices inspired from Rayleigh-Schrödinger perturbation theory and termed *iterative perturbative theory* (IPT). ~~Built upon a "perturbative" partitioning of the matrix into diagonal and off-diagonal parts, IPT can be used to compute one or all eigenpairs~~Contrary to standard eigenvalue algorithms which are either 'direct' (to compute all eigenpairs) or 'iterative' (to compute just a few), IPT computes any number of eigenpairs in parallel with the same basic iterative ~~algorithm. Thanks to the high parallelism of matrix-matrix products, full-spectrum IPT shows excellent performance on multi-core processors and GPUs, with large speed-ups over standard direct methods (up to $\sim$ 50x with respect to LAPACK and MAGMA). For matrices which are not close to being diagonal but have well-separated eigenvalues, IPT can be be used to refine low-precision eigenpairs obtained by other methods.~~ procedure, outperforming LAPACK and preconditioned iterative solvers respectively. We give sufficient conditions for linear convergence and demonstrate performance on dense and sparse test matrices~~. In a real-world application~~ , including one from quantum chemistry~~, we find that single-eigenpair IPT performs similarly to the Davidson algorithm.~~ . The code is available at http://github.com/msmerlak/IterativePerturbationTheory.jl.

Key words. eigenvalue algorithm, perturbation theory, iterative method

AMS subject classifications. 65F15

1. Introduction. ~~Computing the eigenvalues and~~

How can one compute the eigenvectors of a matrix M that is already close to being diagonal~~(or diagonalizable in a knownbasis) is a classic problem with applications throughout science. *Ab initio* studies of electronic and nuclear structures, for instance, often involve Hamiltonian operators expressed in bases that nearly diagonalize them, *e.g.* in self-consistent field or configuration interaction calculations [1]. In classical electromagnetism, perturbative problems arise when the solution to a slightly deformed problem are sought, for instance when boundary conditions are shifted or optical properties are varied [2]. In mechanical engineering, one often wants to know the effect of slight deformations of stiffness or density on spectra, as in Rayleigh's pioneering analysis of vibrating strings [3]. More generally, we might know approximate eigenpairs through some method, and the problem is to refine them to a given precision.~~ , or equivalently when approximate eigenvectors are known? In physics and chemistry, a standard approach, Rayleigh-Schrödinger perturbation theory [3, 4], attempts to compute these eigenvectors as power series in the 'perturbation' $\Delta = M - D$, where D is diagonal and Δ is small. The purpose of this paper is to show that a variant of this method, based on fixed-point iteration rather than power series expansion, can be the basis for a new kind of preconditioned eigenvalue algorithm. We stress from the outset that, like Rayleigh-Schrödinger perturbation theory, this algorithm (termed Iterative Perturbation Theory, IPT for short) has a limited scope: if M is not sufficiently close

[*] Submitted to the editors in Dec. 2020.

Funding: Funding for this work was provided by the Alexander von Humboldt Foundation in the framework of the Sofja Kovalevskaja Award endowed by the German Federal Ministry of Education and Research.

[†] University of Dschang, Cameroon, and Max Planck Institute for the Mathematical Sciences, Leipzig, Germany

[‡] Max Planck Institute for the Mathematical Sciences, Leipzig, Germany

to being diagonal, IPT will not converge. But in those cases—common in physical [3, 2] or chemical [1] applications—where M is in fact near-diagonal, IPT proves much more efficient than established methods, sometimes by an order of magnitude or more.

Eigenvalue ~~perturbation theory provides estimates and bounds for the variation of eigenpairs with respect to perturbations of the matrix [5].Here our aim is , instead, to compute these eigenpairs exactly (up to numerical error) , *i.e.* we are interested in eigenvalue algorithms for perturbative problems. In cases where only a single (or a few) eigenpairs are desired, iterative algorithms that take advantage of near-diagonality are available,~~ algorithms are usually classified in 'direct' and 'iterative' methods [6] . These correspond to very different approaches to the eigenvalue problem. Direct methods, such as the QR and divide-and-conquer algorithms, reduce the the matrix into Hessenberg (or tridiagonal) form before they can compute complete set of eigenvectors.[1] Unfortunately, this initial step is difficult to parallelize, breaks any sparsity patterns in M, and has unfavorable (cubic) complexity. For this reason, direct algorithms are only applicable to small matrices with size $n \lesssim 10^4$.

Iterative methods, on the other hand, only aim to compute a few eigenpairs of M through 'matrix-free' applications of M to vectors. Because they do not require reducing (or even explicitly forming) M, iterative methods can be applied to much larger problems than direct approaches, particularly when M is sparse. Central to most of these methods is the Rayleigh-Ritz procedure: at each iteration, a small subspace is formed (e.g. ~~Davidson-type subspace iteration methods [7, 8, 9, 10] or LOBPCG with a diagonal preconditioner [11]. To obtain the complete set of eigenvectors of near-diagonal symmetric matrices, Jacobi's algorithm may be preferred to QR-based methods: the former's local quadratic convergence and good parallelizability are such that converged eigenvectors can sometimes be obtained in a fraction of the time required for tri-diagonalization~~by expansion from a smaller subspace), and the eigenvectors of the matrix projected to that subspace are computed. This is ~~especially true on modern graphical processing units (GPUs), whose massive parallelism allow for extremely high flop rates that can sometimes offset the unfavourable complexity of Jacobi's algorithm [6].~~ the basis for classical Krylov methods (Arnoldi, Lanczos), but also for preconditioned methods which use an approximation of M^{-1} to speed up convergence. The latter include Davidson methods (Generalized Davidson [7, 10] , Jacobi-Davidson [8, 9]), Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG) [11], etc.

~~Here we introduce a novel iterative algorithm for the computation of one or all eigenpairs of a near-diagonal matrix. While more restricted in its applicability than some of the aforementioned methods, its performance tends to be higher, particularly on full-spectrum problems. This follows from our algorithm's simple linear-algebraic structure: each iteration consists of just one~~ Our algorithm has a different structure. For starters, IPT makes no distinction between direct and iterative procedures: it computes any desired number of eigenpairs in parallel, from one to all of them, using the same basic iterative procedure. Second, IPT does not involve any reduction of the matrix, relying instead on repeated products $M \times Z$, where Z is an $n \times k$ matrix when k eigenpairs are requested. If $k = 1$, this is a matrix-vector ~~multiplication (for~~

[1] One exception is Jacobi's algorithm for symmetric matrices, which annihilates off-diagonal elements with two-dimensional rotations. Because the complexity of Jacobi's algorithm is larger than that of tridiagonal reduction, this algorithm is not used in practice, except perhaps for small problems on GPUs.

~~one eigenpair) or one~~ product, as in iterative methods; when $k = N$ (the full spectrum problem), this is a matrix-matrix ~~multiplication (for all eigenpairs). These operations are highly optimized in the Basic Linear Algebra Subprograms (BLAS Level 2 and 3 respectively), resulting in low execution times. Moreover, since matrix-matrix multiplication has~~ product, with sub-cubic theoretical complexity ~~$\mathcal{O}(N^{2.376})$, so does our algorithm, in contrast with classical direct algorithms which are $\mathcal{O}(N^3)$ for the full spectrum problem.~~

~~Iterative perturbation theory (IPT) was inspired by the Rayleigh-Schrödinger (RS) perturbative expansion familiar from textbook quantum mechanics [12]. (The relationship between the two methods in the context of quantum physics will be discussed in a companion paper [13].) Its structure, however, is different: instead of seeking the perturbed eigenvectors as power series in the perturbation, we compute them iteratively, as fixed points of a quadratic equation . One consequence of this difference is that , unlike Rayleigh-Schrödinger expansions [4], the domain of convergence of our method as a scalar perturbation parameter is varied is not restricted to a disk in the complex plane bounded by exceptional points (values of the parameter such that the matrix is defective); instead, IPT shows for large perturbation the period-doubling route to chaos.~~ $\mathcal{O}(n^{2.376})$. Third, IPT does not rely on the Rayleigh-Ritz procedure or external eigensolvers to solve a projected problem—it is completely self-contained. Fourth, existing methods normally distinguish between symmetric (or Hermitian) and non-symmetric problems. IPT does not assume or use symmetry and is equally efficient with non-symmetric problems. Finally, and most importantly in our view, perhaps, IPT reduces the eigenvalue equation into an elementary fixed-point problem. This means that ~~a fixed point loses its stability and spawns a limit cycle of period two (two points alternated by the map). The limit cycle, in turn, loses stability by spawning a limit cycle of period four, which then spawns a limit cycle of period eight, and so on. At a finite value of the perturbation parameter an accumulation of infinitely many of such period doublings happens culminating in the appearance of a chaotic attractor (see, for example, [14]).~~ the vast repertoire of methods developed in numerical for solving non-linear equations, including fixed-point acceleration methods, become directly relevant to the eigenvalue problem. We feel that this connection between non-linear analysis and numerical linear algebra is the most appealing aspect of IPT.

Our presentation starts with a reformulation of the eigenvalue equation as a fixed point equation for a quadratic map in complex projective space (section 3). We then establish a sufficient condition for fixed point iteration to converge and illustrate its divergence for larger perturbations with a simple two-dimensional example (section 4). An interesting aspect of IPT is its compabibility with fixed-point acceleration methods (section 5). Next, we consider the computational efficiency of our method on multi-CPU and GPU architectures using ~~random test matrices~~test matrices, either sampled at random or drawn from chemistry applications (section 6). We ~~then discuss one possible applications of our algorithm as a refinement method for eigenvectors (). We~~ conclude with some possible directions for future work.

2. Definitions and notations.

Below we denote complex vectors and functions to vector spaces by bold face lowercase letters: $\mathbf{v} \in \mathbb{C}^n$. We denote coordinates of vector $\mathbf{v}$ in the standard basis of $\mathbb{C}^n$ either by $(\mathbf{v})_i$ or by v_i, using the same letter but in Roman. For matrices and functions to spaces of matrices we use upper case Roman letters. Superscripts in parentheses (e.g. $x^{(k)}$) denote the order of an approximation, while the superscripts in square brackets (e.g. $x^{[k]}$) enumerate terms in asymptotic series.

Everywhere in the text we understand by *genericity* of an object in some set $\mathbb{C}^n$ with the standard topology the condition that that object belongs to a fixed, open, everywhere dense subset of $\mathbb{C}^n$. A phrase "condition A is false for a generic object $O \in S$" should be read as "condition A corresponds to a closed nowhere dense subset of S". For example, genericity of an $n \times n$ matrix M is understood with M viewed as an element of $\mathbb{C}^{n\times n}$. A phrase "eigenvalues of a generic $n \times n$ matrix are pairwise different" should be understood as "the set of matrices with eigenvalues of higher multiplicity form a nowhere dense closed subset in $\mathbb{C}^{n\times n}$". Genericity of a family of partitions (defined below) $D + \varepsilon\Delta$, where $D = \operatorname{diag}(\mathbf{v})$, $\mathbf{v} \in \mathbb{C}^n$, and $\Delta \in \mathbb{C}^{n\times n}$, $\varepsilon \in \mathbb{C}$, is understood as genericity of a point $(\mathbf{v}, \Delta, \varepsilon) \in \mathbb{C}^n \times \mathbb{C}^{n\times n} \times \mathbb{C}$, and so on.

We use some elementary notions from projective and differential geometry. Their definitions are given below.

Informally, ~~an~~ the $(n-1)$-dimensional projective space $\mathbb{C}P^{n-1}$ is the space of complex lines (1-dimensional complex subspaces) of the vector space $\mathbb{C}^n$ (the notion is generalizable to vectors spaces over fields different from $\mathbb{C}$). It is constructed in the following way. Consider an equivalence relation of *nonzero* vectors $\mathbf{v}, \mathbf{w} \in \mathbb{C}^n$ given by $\mathbf{v} \sim \mathbf{w}$ if $\exists \alpha \in \mathbb{C}$ such that $\mathbf{v} = \alpha\mathbf{w}$. In other words, all collinear vectors, with exception of 0, are declared to be equivalent.

Definition 2.1. *The* projective space $\mathbb{C}P^{n-1}$ *is the factor-space* $(\mathbb{C}^n \setminus \{0\})/{\sim}$ *by the indicated equivalence relation with the associated factor-topology.*

We care about eigenvectors only up to normalization, therefore the projective space is a natural space that contains them as its points. As usual, we denote the equivalence class of a vector $\mathbf{z} \in \mathbb{C}^n$ (a point in $\mathbb{C}P^{n-1}$) as $[\mathbf{z}]$, and $\mathbf{z}$ is called a representative of that class. The coordinates of $\mathbb{C}^n$ can be used to (nonuniquely) parametrize points of $\mathbb{C}P^{n-1}$ in the following way.

Definition 2.2. *A tuple* $[z_1 : z_2 : \cdots : z_n]$ *is called* homogeneous coordinates *of a point* $p \in \mathbb{C}P^{n-1}$ *if there is a vector* $\mathbf{z} \in \mathbb{C}^n$ *such that* $p = [\mathbf{z}]$ *and* $\mathbf{z}$ *has coordinates* $(z_1, z_2, \ldots, z_n)$.

Thus each point of the projective space has a set of homogeneous coordinates, but all of them are related by homogeneous rescaling: two homogeneous coordinates of the same point $[x_1 : \cdots : x_n]$ and $[y_1 : \cdots : y_n]$ are related by $x_i = \alpha y_i$ with the same α for each i.

$\mathbb{C}P^{n-1}$ has a structure of a complex $(n-1)$-dimensional manifold.

Definition 2.3. *A* complex (coordinate) chart *on a topological space* M *is a homeomorphism* $\phi\colon V \to O \in \mathbb{C}^n$ *from some open set* $V \subset M$ *to an open set* O *of* $\mathbb{C}^n$ *with the standard topology for some dimension* n.

It is convenient to denote a chart ϕ defined on $V \subset M$ with (V, ϕ) explicitly marking its domain.

Definition 2.4. *Let there be two complex charts* (V, ϕ) *and* (W, ψ), $V, W \subset M$, *and let their ranges have the same dimensionality* n. *The homeomorphism* $\phi \circ \psi^{-1}$ *between* $\psi(V \cap W) \subset \mathbb{C}^n$ *and* $\phi(V \cap W) \subset \mathbb{C}^n$ *is called a* transition function *(between charts* ψ *and* ϕ*).*

Definition 2.5. *A topological space* M^n *is called a* complex n-dimensional manifold *if 1) there is a set of charts* $\{(U_i, \phi_i)\}_{i\in I}$ *(*I *is some index set) of* M^n *such that* $\bigcup_{i\in I} U_i = M^n$, *2) all the ranges of* ϕ_i *have the same dimensionality* n *and 3) all the transition functions* $\phi_i \circ \phi_j^{-1}$ *are holomorphic functions between open subset of* $\mathbb{C}^n$.

The system $\{(U_i, \phi_i)\}_{i\in I}$ *is called a* holomorphic atlas *on* M^n.

An atlas allows to work with a manifold using usual methods of linear algebra and analysis in $\mathbb{C}^n$. The standard atlas of charts that gives $\mathbb{C}P^{n-1}$ a structure of an $(n-1)$-dimensional complex manifold is given by n special charts. For each i, $1 \le i \le n$, define $U_i = \{[\mathbf{z}] : z_i \ne 0\}$, label coordinates of $\mathbb{C}^{n-1}$ by $\zeta_1, \ldots, \zeta_{i-1}, \zeta_{i+1}, \ldots, \zeta_n$ (i-th label is omitted) and set $\zeta_k = z_k/z_i$, which defines a map $\phi_i \colon U_i \to \mathbb{C}^{n-1}$. This defines a ~~holomorphic (indeed rational)~~ complex chart on $\mathbb{C}P^{n-1}$ with the whole of $\mathbb{C}^{n-1}$ as its image. It is easy to see that $\{U_i, \phi_i\}_{1\le i\le n}$ forms a holomorphic atlas of $\mathbb{C}P^{n-1}$. The transition functions from ϕ_i with its range coordinatized by ζ_k and ϕ_j with its range coordinatized by η_k (with the same labeling omission rule for each chart as above) is given by $\eta_k = \zeta_k/\zeta_j$ for $k \ne i, j$ and $\eta_i = 1/\zeta_j$. They are simple rational maps and therefore are holomorphic.

In fact, the action of ϕ_i on $[\mathbf{z}]$ can be understood as setting $z_i = 1$, rescaling the rest of z_k accordingly and taking the whole tuple $(z_1, \ldots, z_n)$ as coordinates of a point in the original space $\mathbb{C}^n$. The image of the chart (U_i, ϕ_i) becomes an $(n-1)$-dimensional affine subspace of the original space. In the following we will abuse notations by denoting the domain of the i-th chart, the chart itself, and the corresponding affine subspace with the same symbol U_i. We will refer to these special charts as *affine charts*. Geometric relationships between U_i and the action of the transition functions as projections between them is schematically shown in **??** (other notations on this sketch will be introduced below).

Polynomials and their systems can be used to define subsets on linear complex spaces.

Definition 2.6. *Let* $\{P_k\}_k$ *be a system of polynomials in* n *variables* $\{z_i\}_i$. *The subset* $V = \{\mathbf{z} \in \mathbb{C}^n : \forall\, k\, P_k(z_1, \ldots, z_n) = 0\}$ *is called an* algebraic variety.

A polynomial P such that $\forall \alpha \in \mathbb{C}\ P(\alpha z_1, \ldots, \alpha z_n) = \alpha^k P(z_1, \ldots, z_n)$ is called a *homogeneous polynomial*. It is obvious that if a point $\mathbf{z}$ is a root of a homogeneous polynomial, then every representative of its equivalence class $[\mathbf{z}]$ is a root, too. Thus homogeneous polynomials can be used to define subsets in projective space.

Definition 2.7. *Let* $\{P_k\}_k$ *be a system of homogeneous polynomials in* n *variables* $\{z_i\}_i$. *The subset* $V = \{[\mathbf{z}] \in \mathbb{C}P^{n-1} : \forall\, k\, P_k(z_1, \ldots, z_n) = 0\}$ *is called a* projective variety.

Note that any projective variety defines an algebraic variety in each affine chart U_i.

3. Eigenvectors as fixed points. Consider an $n \times n$ complex matrix M. Its eigenvectors are elements of $\mathbf{z} \in \mathbb{C}^n$.

Lemma 3.1. *The eigenvectors of* M *are in one-to-one correspondence with non-zero solutions of the systems* $\{\mathcal{E}_{ij}\}_{i,j}$ *of polynomial equations in coordinates of* $\mathbf{z}$ *for all* i *and* j, *where*

$$\mathcal{E}_{ij} : (M\mathbf{z})_j z_i = (M\mathbf{z})_i z_j.$$

Proof. The system $\{\mathcal{E}_{ij}\}_{i,j}$ states that the tensor $M\mathbf{z} \otimes \mathbf{z} - \mathbf{z} \otimes M\mathbf{z}$ vanishes. This statement is equivalent to the statement that $M\mathbf{z}$ and $\mathbf{z}$ are collinear. □

~~Being defined up to a multiplicative constant~~As already noted, the eigenvectors of matrix M are naturally identified with elements of the complex projective space $\mathbb{C}P^{n-1}$. From this point of view, the system $\{\mathcal{E}_{ij}\}_{i,j}$ defines a projective variety.

~~A schematic sketch of the geometrical relations between the original space $\mathbb{C}^n$, the projective space $\mathbb{C}P^{n-1}$, its affine charts, and points in these different spaces~~

Algorithm 3.1 One eigenvector of a near-diagonal matrix M

```
Choose a partition $M = D + \Delta$ with $D$ diagonal
Compute $\mathbf{g}_i \leftarrow ((D_{jj} - D_{ii})^{-1})_{j:j\neq i}$ and set $(\mathbf{g}_i)_i = 0$
Initialize $\mathbf{z} \leftarrow \mathbf{e}_i$
while $|\mathbf{z} - \mathbf{f}_i(\mathbf{z})|/|\mathbf{z}| > \eta$ for some tolerance $\eta$ do
    $\mathbf{z} \leftarrow \mathbf{f}_i(\mathbf{z})$ with $\mathbf{f}_i$ defined by (3.1)
end while
Return eigenvector $\mathbf{z}$
Return eigenvalue $\lambda_i = (M\mathbf{z})_i$
```

~~that correspond to certain eigenvectors. The projective space $\mathbb{C}P^{n-1}$ (in red) can be associated with the unit sphere in $\mathbb{C}^n$, in which each circle carved out in it by a complex line passing through the origin (a special case of a real plane) is glued to form a single point (here exemplified by two abstract points p and q, the circles are degenerated to pairs of points in real settings). The standard basis $\{\mathbf{e}_k\}$ of $\mathbb{C}^n$ (eigenvectors of D) define affine charts U_k of $\mathbb{C}P^{n-1}$ (only two are depicted: for $k = i$ in blue and $k = j$ in green). U_i and U_j are shown as affine subspaces of $\mathbb{C}^n$. Note that $\mathbf{e}_j$ is at infinity of U_i and $\mathbf{e}_i$ is at infinity of U_j. An actual eigenvector and all its nonzero rescalings correspond to a single point in $\mathbb{C}P^{n-1}$ and in each U_k. Two eigenvectors $\mathbf{z}_i$ and $\mathbf{z}_j$ are depicted with a rescaling choice such that they are equal to the i-th and j-th columns of matrix Z.~~

Fix an index i. To find the full set of eigenvectors in a single chart U_i it is enough to solve a smaller set of equations.

LEMMA 3.2. *Eigenvectors $\mathbf{z}$ of M such that $[\mathbf{z}] \in U_i$ are in one-to-one correspondence with the solutions of the system of equations $\{\mathcal{E}_{ij}\}_{j:j\neq i}$ in the same chart.*

Proof. We need to prove that for $[\mathbf{z}] \in U_i$ the two following propositions are equivalent: (i) $\mathbf{z}$ is a root of $(M\mathbf{z})_j z_i = (M\mathbf{z})_i z_j$ for $j \neq i$ and (ii) $\exists \lambda \in \mathbb{C}$ $M\mathbf{z} = \lambda\mathbf{z}$.

The $(i) \Rightarrow (ii)$ direction. For $[\mathbf{z}] \in U_i$ it is enough to consider only vectors with $z_i = 1$. Then from $\{\mathcal{E}_{ij}\}_{j:j\neq i}$ we conclude that $(M\mathbf{z})_j = (M\mathbf{z})_i z_j$ for all $j \neq i$. If we now denote the common factor in these expressions as $\lambda = (M\mathbf{z})_i$, then (ii) immediately follows.

The $(ii) \Rightarrow (i)$ direction. Consider an eigenvector $\mathbf{z}$ of M such that $z_i \neq 0$, and thus $[\mathbf{z}] \in U_i$. From the eigenvalue equation $M\mathbf{z} = \lambda\mathbf{z}$ we can express its eigenvalue as $\lambda = (M\mathbf{z})_i / z_i$; inserting this back into $M\mathbf{z} = \lambda\mathbf{z}$ shows that $\mathbf{z}$ is a root of the system $\{\mathcal{E}_{ij}\}_{j:i\neq j}$. □

Denoting V_i the projective variety defined by the system $\{\mathcal{E}_{ij}\}_{j:j\neq i}$ for a fixed i and recalling that the set $\{U_i\}_i$ forms an atlas we conclude that the set of eigenvectors of M can be identified with the set $\cup_i(V_i \cap U_i)$, which is in fact the projective variety $\cap_i V_i$. Since eigenvectors generically have non-zero coordinates in all directions, each component $V_i \cap U_i$ typically contains the complete set of eigenvectors.

It should be noted, however, that V_i taken alone may have points unrelated to the eigenvectors of M, but these additional points can only be at infinity in U_i by Lemma 3.2. Indeed, consider points of V_i such that $z_i = 0$. At the very least they include points that are given by $(M\mathbf{z})_i = 0$ in addition to $z_i = 0$. Thus, in general, there is a whole $(n-3)$-dimensional complex projective subspace of such solutions in $\mathbb{C}P^{n-1}$ (corresponding to a $(n-2)$-dimensional complex hyperplane in the affine space $\mathbb{C}^n$).

We now further assume a partitioning

$$M = D + \Delta$$

with a diagonal part D and the residual part Δ. The motivation comes from physics, where the diagonal part D often consists of unperturbed eigenvalues $\mathring{\lambda}_i = D_{ii}$ and Δ represents perturbations in the ~~so called~~ so-called perturbation theory. In practice D can be taken as the diagonal elements of M, although different partitionings can sometimes be more appropriate [15]. Provided the $\mathring{\lambda}_i$'s are all simple (non-degenerate, that is pairwise different), we can rewrite the polynomial system $\{\mathcal{E}_{ij}\}_{j:j\neq i}$ for a particular fixed i as

$$(D\mathbf{z} + \Delta\mathbf{z})_j z_i = (D\mathbf{z} + \Delta\mathbf{z})_i z_j \text{ for } j \neq i,$$

or equivalently

$$\mathring{\lambda}_j z_j z_i + (\Delta\mathbf{z})_j z_i = \mathring{\lambda}_i z_i z_j + (\Delta\mathbf{z})_i z_j \text{ for } j \neq i,$$

which yields

$$z_i z_j = (\mathring{\lambda}_j - \mathring{\lambda}_i)^{-1} \left(z_j(\Delta\mathbf{z})_i - z_i(\Delta\mathbf{z})_j\right) \text{ for } j \neq i.$$

In the affine subspace U_i this gives (by setting $z_i = 1$)

$$z_j = (\mathring{\lambda}_j - \mathring{\lambda}_i)^{-1} \left(z_j(\Delta\mathbf{z})_i - (\Delta\mathbf{z})_j\right) \text{ for } j \neq i.$$

Thus, the eigenvectors of M in U_i can be identified with the solutions of the fixed point equation $\mathbf{z} = \mathbf{f}_i(\mathbf{z})$ with $\mathbf{f}_i : \mathbb{C}^n \to \mathbb{C}^n$ the map

$$\mathbf{f}_i(\mathbf{z}) = \mathbf{e}_i + \mathbf{g}_i \circ (\mathbf{z}(\Delta\mathbf{z})_i - \Delta\mathbf{z}) \tag{3.1}$$

where $\mathbf{e}_i$ is the i-th standard basis vector of $\mathbb{C}^n$, $\mathbf{g}_i$ is the i-th column of the *inverse gaps matrix* G with components $G_{jk} = (\mathbf{g}_k)_j = (\mathring{\lambda}_j - \mathring{\lambda}_k)^{-1}$ for $j \neq k$ and $G_{jj} = (\mathbf{g}_j)_j = 0$. Here $\circ$ denotes the component-wise product of vectors in the standard basis. To obtain the eigenvector of M closest to the basis vector $\mathbf{e}_i$, we can try to solve (3.1) by fixed-point iteration; this is the basic idea of our method, presented in Algorithm 3.1. As noted, each iteration of $\mathbf{f}_i$ consists of a single multiplication of the present vector by the perturbation Δ, followed by element-wise multiplication by the vector $\mathbf{g}_i$. The resulting solution will be automatically normalized such that its i-th coordinate is equal to 1.

~~One eigenvector of a near-diagonal matrix M~~

~~Initialize $\mathbf{z} \leftarrow \mathbf{e}_i$~~

The same iterative technique can be used to compute all eigenvectors of M in parallel (Algorithm 3.2). For this it suffices to bundle all n candidate eigenvectors for each i into a matrix Z and apply the map $\mathbf{f}_i$ to the i-th column of Z. This corresponds to the matrix map

$$F(Z) \equiv I + G \circ \Big(Z\,\mathcal{D}(\Delta Z) - \Delta Z\Big), \tag{3.2}$$

where $\circ$ denotes the Hadamard (element-wise) product of matrices and $\mathcal{D}(X)$ is the diagonal matrix built with the diagonal elements of matrix X. Starting from $Z^{(0)} = I$, we obtain a sequence of matrices $Z^{(k)} = F(Z^{(k-1)})$ whose limit as $k \to \infty$, if exists, is the full set of eigenvectors. We call this approach *iterative perturbation theory* (IPT).

Algorithm 3.2 Full eigendecomposition of a near-diagonal matrix M

```
Choose a partition $M = D + \Delta$ with $D$ diagonal
Compute $G \leftarrow ((D_{ii} - D_{jj})^{-1})_{i \neq n}$ and set $g_{ii} = 0$
Initialize $Z \leftarrow I$
while $\|Z - F(Z)\|/\|Z\| > \eta$ for some tolerance $\eta$ do
    $Z \leftarrow F(Z)$ with $F$ defined by (3.2)
end while
Return eigenmatrix $Z$
Return eigenvalues $\Lambda = \mathcal{D}(MZ)$
```

4. Convergence and divergence.

In this section we look at the convergence of fixed-point iteration for the map (3.2). In a nutshell, the off-diagonal elements Δ must be small compared to the diagonal gaps $\mathring{\lambda}_i - \mathring{\lambda}_j$, a typical condition for eigenvector perturbation theory [4].

4.1. A sufficient condition for convergence.

Let $\|\cdot\|$ denote the spectral norm of a matrix, i.e. its largest singular value. Let $M \in \mathbb{C}^{n\times n}$ be a matrix. Let $M = D + \Delta$ be its partition into a diagonal matrix D and the residual matrix Δ such that the corresponding to D matrix of inverse gaps G is defined, which implies that all diagonal elements of D are pairwise different. Let $F : \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$ be the mapping defined by G and Δ as in the previous section.

Theorem 4.1. *If*

$$\|G\|\|\Delta\| < 3 - 2\sqrt{2}, \tag{4.1}$$

then the dynamical system defined by iterations of F and $A^{(0)} = I$ converges to a unique, asymptotically stable fixed point in the ball $B_{\sqrt{2}}(I)$.

Proof. The proof uses the Banach fixed-point theorem based on the $\|\cdot\|$ norm, which is sub-multiplicative with respect to both the matrix and Hadamard products [16].

First, the estimate

$$\|F(Z) - I\| \leq \|G\|\|\Delta\|(\|Z\| + \|Z\|^2)$$

implies that F maps a closed ball $B_r(I)$ of radius r centered on I onto itself whenever $\|G\|\|\Delta\| \leq r/[(1+r)(2+r)]$. Next, from (3.2) we have the estimate

$$\|F(Z) - F(Z')\| \leq \|G\|\|\Delta\|\,(1 + \|Z + Z'\|)\,\|Z - Z'\|.$$

Hence, F is contracting in $B_r(I)$ provided $\|G\|\|\Delta\| < 1/[1 + 2(1+r)]$. When both conditions on $\|G\|\|\Delta\|$ hold the Banach fixed-point theorem implies that $Z^{(k)} = F^k(I)$ converges exponentially to a unique fixed point Z^* within $B_r(I)$ as $k \to \infty$. Choosing the optimal radius

$$\operatorname*{argmax}_{r>0} \min\left(\frac{r}{(1+r)(2+r)}, \frac{1}{1+2(1+r)}\right) = \sqrt{2},$$

we see that $\|G\|\|\Delta\| < 3 - 2\sqrt{2} \approx 0.17$ guarantees convergence to the fixed point Z^*.□

The set of solutions of the equation $Z = F(Z)$ contains matrices composed of all combinations of eigenvectors of M with the appropriate normalization (the i-th coordinate of the column at the i-th position is set to 1). Some solutions may contain several repeated eigenvectors. Such solutions are rank-deficient, viz. $\operatorname{rank} Z < n$. In principle, there is a danger that the iterative algorithms converges to one such solution with a loss of information about the eigenvectors of M as a result. The following ~~lemma~~ theorem guarantees that under condition (4.1) this does not happen.

Theorem 4.2. *Let condition (4.1) hold for a partition of matrix M and Z^* be the unique fixed point of F in $B_{\sqrt{2}}(I)$. Then Z^* has full rank.*

Proof. The proof uses additional lemmas and one theorem provided in Appendix A.

A square matrix is rank-deficient if either two of its columns are collinear or more than two of its columns are linearly dependent but not pairwise collinear. For Z^* the latter is only possible if the corresponding eigenvectors belong to an eigenspace of M of dimension higher than one, which is ruled out by Lemma A.1. By Lemma A.2, Z^* does not contain any defective eigenvectors of M. Then the rank can be lost only by a repetition of an eigenvector (with renormalization) corresponding to an eigenvalue of multiplicity one.

Embed M into the family $M_\varepsilon = D + \varepsilon\Delta$ with $\varepsilon \in [0, 1]$, so that $M_1 = M$ and $M_0 = D$. Consider a partition for each member of the family $M_\varepsilon = D_\varepsilon + \Delta_\varepsilon$ such that $D_\varepsilon = D$ and $\Delta_\varepsilon = \varepsilon\Delta$. Let $G_\varepsilon = G$ be the inverse gaps matrix of D_ε and F_ε be the mapping defined by G_ε and Δ_ε according to (3.2) with a substitution of G by G_ε and Δ by Δ_ε. It follows that $\|G_\varepsilon\|\|\Delta_\varepsilon\| < 3 - 2\sqrt{2}$ holds for all ε. Thus, according to Theorem 4.1, each F_ε has a unique fixed point Z^*_ε in $B_{\sqrt{2}}(I)$. Furthermore, by Lemmas A.1 and A.2 we can be sure that none of the columns of Z^*_ε for each value of ε corresponds either to an eigenvector with algebraic multiplicity higher than one or lays in an eigenspace of M_ε with dimension higher than one. Thus they all correspond to eigenvalues that are simple roots of the characteristic polynomial of M_ε. Then all projective points that induce columns of Z^*_ε are differentiable functions of ε by Theorem A.3. It means that if some two columns of $Z^* = Z^*_1$ are collinear (and thus correspond to the same projective point), the same columns stay collinear in Z^*_ε for all ε. But none of the columns of $I = Z^*_0$ are collinear. We conclude that Z^* cannot have collinear columns. □

Corollary 4.3. *If M has eigenvalues with multiplicity higher than one, then for any partition $\tilde{M} = \tilde{D} + \tilde{\Delta}$ of any matrix $\tilde{M}$ similar to M, where $\tilde{D}$ is diagonal such that the corresponding inverse gaps matrix $\tilde{G}$ is defined, the relation $\|\tilde{G}\|\|\tilde{\Delta}\| \geq 3 - 2\sqrt{2}$ holds.*

The following subsections are dedicated to aspects of IPT related to quantum-mechanical perturbation theory (sec. 4.2) and dynamical systems theory (sec. 4.3~~,~~ ~~??~~). They may be omitted by the non-specialist reader.

4.2. Contrast with ~~RS~~ Rayleigh-Schrödinger perturbation theory. It is interesting to constrast the present iterative method with conventional ~~RS~~ Rayleigh-Schrödinger (RS) perturbation theory, where the eigenvectors of a parametric matrix $M = D + \varepsilon\Delta$ are sought as power series in ε, *viz.* $\mathbf{z} = \sum_\ell \varepsilon^\ell \mathbf{z}^{[\ell]}$, where vector-terms (so called corrections) $\mathbf{z}^{[\ell]}$ do not depend on ε. Provided that D has distinct diagonal entries, it is indeed possible to express the matrix of eigenvectors Z (with the same normalization as in Section 3 and with the order of eigenvectors such that $Z \to I$ as $\varepsilon \to 0$) as a power series $Z = \sum_\ell \varepsilon^\ell Z^{[\ell]}$, which converges in some disk around $\varepsilon = 0$ [4].

Then the order-k approximation of Z takes the form $Z_{\mathrm{RS}}^{(k)} = \sum_{\ell=0}^{k} \varepsilon^{\ell} Z^{[\ell]}$. The matrix corrections $Z^{[\ell]}$ are obtained from $Z^{[0]} = I$ via the recursion (Appendix B)

$$Z^{[\ell]} = G \circ \left(\sum_{s=0}^{\ell-1} Z^{[\ell-1-s]} \, \mathcal{D}(\Delta Z^{[s]}) - \Delta Z^{[\ell-1]} \right). \tag{4.2}$$

The iterative scheme $Z^{(k)}$ completely contains this RS series in the sense that $Z^{(k)} = Z_{\mathrm{RS}}^{(k)} + \mathcal{O}(\varepsilon^{k+1})$; this can be seen by induction (Appendix C). In other words, we can recover the usual perturbative expansion of Z to order k by iterating k times the map F and dropping all terms $\mathcal{O}(\varepsilon^{k+1})$. Moreover, the parameter whose smallness determines the convergence of the RS series is the product of the perturbation magnitude $|\varepsilon|$ with the inverse diagonal gaps $\|G\|$ [4], just as it determines the contraction property of F.

But IPT also differs from the RS series in two key ways. First, the complexity of each iteration is constant (essentially just one matrix product with Δ), whereas computing the RS corrections $Z^{[\ell]}$ involves the sum of increasingly many matrix products. Second, not being defined as a power series, the convergence of $Z^{(k)}$ when $k \to \infty$ is not *a priori* restricted to a disk in the complex ε-plane. Together, these two differences suggest that IPT has the potential to converge faster, and in a larger domain, than RS perturbation theory. This is what we now examine, starting from an elementary but explicit example.

~~Convergence of IPT in the two-dimensional example . Left: In the complex ε-plane, RS perturbation theory (RS-PT) converges inside a circle of radius $1/2$ (orange line) bounded by the exceptional points $\pm i/2$ where eigenvalues have branch-point singularities and M is not diagonalizable. Dynamical perturbation theory (IPT) converges inside the domain bounded by the blue cardioid, which is larger—especially along the real axis, where there is no singularity. Outside this domain, the map can converge to a periodic cycle, be chaotic or diverge to infinity, following flip bifurcations (along the real axis) and fold bifurcations (at the singularities). The domain where the map remains bounded (black area) is a conformal transformation of the Mandelbrot set. Right: The bifurcation diagram for the quadratic map f along the real ε-axis illustrates the period-doubling route to chaos as ε increases away from 0 (in absolute value). Orange and left vertical lines indicate the boundary of the convergence domains of RS-PT and IPT respectively.~~

4.3. An explicit 2×2 example. To build intuition, let us consider the parametric matrix

$$M = \begin{pmatrix} 0 & \varepsilon \\ \varepsilon & 1 \end{pmatrix}. \tag{4.3}$$

This matrix has eigenvalues $\lambda_{\pm} = (1 \pm \sqrt{1+4\varepsilon^2})/2$, both of which are analytic inside the disk $|\varepsilon| < 1/2$ but have branch-point singularities at $\varepsilon = \pm i/2$. (These singularities are exceptional points, *i.e.* M is not diagonalizable for these values.) Because the RS series is a power series, these imaginary points contaminate its convergence also on the real axis, where no singularity exists: Z_{RS} diverges for any value of ε outside the disk of radius $1/2$, and in particular for real $\varepsilon > 1/2$.

Considering instead our iterative scheme, one easily computes

$$Z^{(k)} = \begin{pmatrix} 1 & -f^k(0) \\ f^k(0) & 1 \end{pmatrix},$$

where $f(x) = \varepsilon(x^2 - 1)$ and the superscripts indicate k-fold iterates. This one-dimensional map has two fixed points at $x^*_\pm = \lambda_\pm/\varepsilon$. Of these two fixed points x^*_+ is always unstable, while x^*_- is stable for $\varepsilon \in (-\sqrt{3}/2, \sqrt{3}/2)$ and loses its stability at $\varepsilon = \pm\sqrt{3}/2$ in a flip bifurcation. At yet larger values of ε, the iterated map f^k—hence the fixed-point iterations $Z^{(k)}$—follows the period-doubling route to chaos familiar from the logistic map [17]. For values of ε along the imaginary axis, we find that the map is stable if $\Im\varepsilon \in (-1/2, 1/2)$ and loses stability in a fold bifurcation at the exceptional points $\varepsilon = \pm i/2$. The full domain of convergence of the system is strictly larger than the RS disk, as shown in **??**. We also observe that the disk where both schemes converge, the dynamical scheme does so with a better rate than RS perturbation theory: we check that $|f^k(0) - x^*_-| \sim |1 - \sqrt{1 + 4\varepsilon^2}|^k = \mathcal{O}(|2\varepsilon^2|^k)$, while the remainder of the RS decays as $\mathcal{O}(|2\varepsilon|^k)$. This is a third way in which the dynamical scheme outperforms the usual RS series, at least in this case: not only is each iteration computationally cheaper, but the number of iterations required to reach a given precision is lower.

~~The set where IPT loses stability (the blue curve in) can be computed as $4\varepsilon^2 + e^{it}(2 - e^{it}) = 0$ (same equation for both eigenvectors). The cusps of this curve are at $\varepsilon = \pm i/2$. In this particular case, the convergence circle of the RS perturbation theory is completely contained in~~ Although it is possible to analytically compute the convergence domain of ~~the iterative perturbation theory, and their boundaries intersect only at the cusp points. This convergence domain for IPT is directly related to the main cardioid of the classical Mandelbrot set: the set of complex values of the parameter c that lead to a bound trajectory of the classical quadratic (holomorphic) dynamical system $x^{(k+1)} = (x^{(k)})^2 + c$. The main cardioid of the Mandelbort set (the domain of stability of a steady state) is bounded by the curve $4c - e^{it}(2 - e^{it}) = 0$. The boundary of the stability domain of our 2×2 example is simply a conformal transform of this cardioid by two complementary branches of the square root function composed with the sign inversion. The origin of this relation becomes obvious after the parameter change $c \mapsto -\varepsilon^2$ followed by the variable change $x \mapsto \varepsilon x$. This brings the classical system to the dynamical system of the only nontrivial component of the first line for our~~ our iterative scheme in this simple case, in practice this task is prohibitive even for very small matrices, as it can be seen on a slightly more complicated 3×3 example. This example along with the computation of the convergence domain for the 2×2 ~~example: $x^{(k+1)} = \varepsilon\left((x^{(k)})^2 - 1\right)$. The nontrivial component of the second line follows an equivalent (up to the sign change of the variable) equation: $x^{(k+1)} = \varepsilon\left(1 - (x^{(k)})^2\right)$.~~ case can be found in Appendix D.

4.4. ~~Cusps and exceptional points~~.

5. Acceleration.

~~The link between the convergence of IPT and the singularities of the spectrum of M (as a function of the complex perturbation parameter ε) generalizes to higher dimensions. Consider a map $\mathbf{f}_i$ for some fixed i. An attracting equilibrium point of the corresponding dynamical system $\mathbf{z}^{(k)} = \mathbf{f}_i(\mathbf{z}^{(k-1)})$ loses its stability when the Jacobian matrix $\partial\mathbf{f}_i$ of $\mathbf{f}_i$, $(\partial\mathbf{f}_i)_{js} \equiv \partial(\mathbf{f}_i)_j/\partial z_s = \partial F_{ji}/\partial z_s$, has an eigenvalue (or *multiplier*) with absolute value equal to 1 at this point. The convergence domain of the dynamical system for the whole matrix Z of the eigenvectors is equal to the intersection of the convergence domains for its individual lines .~~

~~Consider the system of $n + 1$ polynomial equations of $n + 2$ complex variables (z_j for $1 \leq j \leq n$, ε, and μ)~~

$$\begin{cases}\mathbf{z} = \mathbf{f}_i(\mathbf{z}),\\ \det(\partial\mathbf{f}_i - \mu I) = 0.\end{cases}$$

~~The variable μ here plays the role of a multiplier of a steady state. Either by successively computing resultants or by constructing a Groebner basis with the correct lexicographical order, one can exclude the variables z_j from this system, which results in a single polynomial of two variables $(\varepsilon,\mu)\mapsto P(\varepsilon,\mu)$. This polynomial defines a complex 1-dimensional variety. The projection to the ε-plane of the real 1-dimensional variety defined by $\{P=0, |\mu|^2=1\}$ corresponds to some curve C. A more informative way is to represent this curve as a complex function of a real variable t implicitly defined by $P(\varepsilon, e^{it})=0$~~As a fixed-point problem, IPT is directly amenable to the broad set of fixed-point acceleration methods familiar from numerical analysis [18]. Generally speaking, acceleration techniques aim to reduce the number of iterations required for convergence by combining several iterates of the function in each update.

~~The curve C is the locus where a fixed point of $\mathbf{f}_i$ have a multiplier on the unit circle. In particular, the fixed point that at $\varepsilon=0$ corresponds to $z_i=1$ and $z_j=0$, $j\neq i$, loses its stability along a particular subset of this curve. The convergence domain of the iterative perturbation theory is the domain that is bounded by these parts of the curve and that contains 0. It is possible to show that C is a smooth curve with cusps (return points) , which correspond to the values of ε such that M has a nontrivial Jordan normal form (is non-diagonalizable) , see . In a typical case,all cusps a related to the merging of a pair of eigenvectors of M. For the dynamical system , the cusps, thus, correspond to the fold bifurcations of its steady states [19]. One of the multipliers equals to 1 at such merged point [14], so these values of ε can be found as a subset of the roots of the univariate polynomial $P(\varepsilon,1)$. Not all its roots generally correspond to cusps and fold bifurcations, though, as demonstrated in . On the other hand, all the cusps/fold bifurcation points are among the ε-roots of the discriminant polynomial $\mathrm{Disc}_x\det(M-xI)$. These roots may correspond to an actually defective matrix M (when two eigenvectors merge and M acquires a nontrivial Jordan normal form) or a simple degeneration of its eigenvalues (when two eigenvalues become equal but M retains a trivial Jordan normal form). Simple degenerations of eigenvalues are not related to cusps. Thus, the cusp points of C can be found as the intersection of the root sets of the two polynomials.~~ We experimented with the well-known Anderson acceleration [20] and found that, indeed, IPT can be made to converge faster and in a larger domain that with simple Picard iteration. Here we use the more recent (but simpler) technique called Alternating Cyclic Extrapolation (ACX) [21]. This method consists in replacing line 4,5 in Algorithm 3.1 (or similarly in Algorithm 3.2) by Algorithm 5.1. As will see below, IPT with ACX acceleration (IPT-ACX) is a very efficient approach to near-diagonal eigenvalue problems.

6. Performance.

~~Runtime vs. dimension N for dense, non-symmetric perturbative matrices of the form in single precision. Blue lines refer to IPT and orange lines to LAPACK's GEEV routine, implemented on 128 CPU cores (continuous lines) or on a Titan V GPU (dashed lines). Within its perturbative convergence domain, IPT is up to 32 and 91 times faster than GEEV, respectively.~~

~~We now compare the efficiency and accuracy of IPT to reference eigenvalues routines~~We investigated the performance of IPT with respect to classical algorithms using several test matrices. Our system consists of ~~a dual AMD EPYC~~ 2×64 CPU

Algorithm 5.1 Alternating Cyclic Extrapolation [21], with vector of orders $o = (3, 2, 3, 2, \cdots)$

```
$i = 1$
while $|\mathbf{z} - \mathbf{f}_i(\mathbf{z})|/|\mathbf{z}| > \eta$ for some tolerance $\eta$ do
    $p \leftarrow o_k$
    $\Delta^0 \leftarrow \mathbf{z}$
    $\Delta^1 \leftarrow \mathbf{f}_i(\mathbf{z}) - \mathbf{z}$
    $\Delta^2 \leftarrow \mathbf{f}_i^2(\mathbf{z}) - 2\mathbf{f}_i(\mathbf{z}) + \mathbf{z}$
    if $p = 2$ then
        $\sigma \leftarrow |\langle \Delta^2, \Delta^1 \rangle|/\|\Delta^2\|$
        $\mathbf{z} \leftarrow 2\sigma\Delta^1 + \sigma^2\Delta^2$
    else if $p = 3$ then
        $\Delta^3 \leftarrow \mathbf{f}_i^3(\mathbf{z}) - 3\mathbf{f}_i^2(\mathbf{z}) + 3\mathbf{f}_i(\mathbf{z}) - \mathbf{z}$
        $\sigma \leftarrow |\langle \Delta^3, \Delta^2 \rangle|/\|\Delta^3\|$
        $\mathbf{z} \leftarrow 3\sigma\Delta^1 + 3\sigma^2\Delta^2 + \sigma^3\Delta^3$
    end if
end while
```

cores (AMD Epyc 7702 ~~with 128 CPU cores using using LAPACK and SLEPc [22] and a NVidia Titan V GPUusing MAGMA [23] and Nvidia cuSOLVER. Test matrices are of the form~~

$$M = \operatorname{diag}(n)_{1\leq n\leq N} + \varepsilon R$$

~~where R is an array of standard normal random variables and ε a positive parameter. We consider four cases: R can either be dense or sparse (with density $50/N$, corresponding to $\sim 50N$ non-zero elements), and non-symmetric or symmetric (via $(R+R^T)/2$).For each case the time required to compute all N eigenpairs in double precision is denoted T_{eig}, and the time to compute the lowest-lying eigenpair is denote T_{eigs}. As a reference we use the time for one matrix-matrix multiplication T_{mm} or~~ at 2.0 GHz) and one ~~matrix-vector multiplication T_{mv}, respectively~~A100 NVidia GPU. ~~In all figures below we use continuous lines for experiments on the CPU and dashed lines for experiments on the GPU, and brackets indicate the wrapper we used to call the relevant eigenvalue routine. Unless specified otherwise, the error tolerance is $\eta = 100\epsilon$, where $\epsilon \simeq 2.2 \cdot 10^{-16}$ (resp. $\epsilon \simeq 1.2 \cdot 10^{-7}$) is machine epsilon in double (resp. single) precision.~~

6.1. ~~Full spectrum~~.

6.1. Few eigenvalues.

To compute a small subset of eigenvalues of a near-diagonal matrix, the reference algorithms are preconditioned iterative methods, with D^{-1} as preconditioner. (Krylov methods are much slower for these problems.) We compared IPT to the symmetric eigensolver PRIMME (PReconditioned Iterative MultiMethod Eigensolver) [24], a package containing various implementations of Rayleigh Quotient Iteration (RQI), Generalized Davidson (GD, GD+k), Jacobi-Davidson (JDQR, JDQMR), and Locally Optimal Block Preconditioned Conjugate Gradient (LOBPCG).[2] PRIMME also includes a "dynamic" mode with shifts dynamically between GD+k and JDQMR in an attempt to minimize the number of matrix-vector multiplications.

~~IPT vs. LAPACK for the complete diagonalization of perturbative matrices with increasing ε, in double precision. The largest speed-ups are obtained for the~~

[2] We also tested SLEPc [22] but, on our single-node system, PRIMME was faster.

~~non-symmetric problem.~~ We remind the reader that GD, JD and RQI all consist in expanding a small search subspace, orthonormalizing that subspace, and computing the eigendecomposition of M in that subspace. Where they differ is how the subspace is expanded. GD uses the current residual vector $\mathbf{r} = \mathbf{f}(\mathbf{z}) - \mathbf{z}$ after preconditioning with D^{-1}. JD and RQI, by contrast, solve a linear "correction" equation (approximately or exactly, respectively) and expand the subspace with the correction vector; this inner iteration involves additional matrix-vector products per step compared to GD. We refer the reader to the documentation of PRIMME for implementation details.

~~For the complete diagonalization problem we compare IPT to MATLAB's **eig** function, which calls LAPACK's routines GEEV for non-symmetric matrices and SYEV for symmetric matrices; on the GPU MAGMA is used instead.~~

~~shows the corresponding CPU timings in the dense, sparse, symmetric and non-symmetric cases ($N = 4096$). We make two observations. First, unlike LAPACK, our algorithm has a similar complexity with symmetric and non-symmetric matrices, leading to larger speed-ups in the latter case. Second, because it is based on matrix-matrix multiplication, IPT performs especially well on sparse matrices. This is another difference with standard full spectrum algorithms, which do not take advantage of sparsity.~~

~~As noted earlier, Hessenberg reduction is inefficient for matrices which are already near diagonal. In the next experiment we compare IPT with Nvidia's GPU implementation of Jacobi's algorithm, which does not rely on reduction and is known to have local quadratic convergence. In spite of thisadvantage, Jacobi (labeled SYEVJ in cuSOLVER) only proved faster than divide-and-conquer SYEVD for small matrices sizes; for $N = 8182$ SYEVJ is slower than SYEVD for all values of ε (, center and right panels). By contrast, our implementation of IPT in CUDA is faster than SYEVD for $\varepsilon \leq 0.05$. For dense, non-symmetric matrices, IPT is much faster than GEEV for all ε compatible with convergence (~~For the implementation of IPT and ACX we used the Julia programming language [25]. The code is freely available at [26] (for IPT) and [27] (for the timings and figures). As already noted, ~~top-left panel) .~~ when using PRIMME we set D^{-1} as preconditioner. Other than this, we used the default settings for all methods.

~~Timing of full spectrum routines on the GPU for dense perturbative matrices. Here IPT proves to be faster than both Divide and Conquer (SYEVD) and Jacobi (SYEVJ) when ε is sufficiently small.~~ Our first example is from a real-world chemistry application. Full-configuration interaction (FCI) consists in computing the ground state (lowest-lying eigenvector) of the Hamiltonian operator of a molecule in a finite basis set [1]. Because molecular orbitals can be computed approximately using self-consisted field approximations, the corresponding (symmetric) Hamiltonian matrix is near-diagonal, and suitable eigensolvers must make use of this information. (Indeed, Davidson's diagonally-preconditioned algorithm [7] was introduced for this purpose.)

6.2. ~~**One eigenpair.**~~ ~~Next we consider the computation of just the lowest eigenpair of the matrices above, corresponding to the column $n = 1$. Here performance comparison are based on the SLEPc library [22, 28], which includes the modern Krylov-Schur algorithm [29] as well as preconditioned eigensolvers including Generalized Davidson (GD) [7, 30], Jacobi-Davidson [9] and , for symmetric problems, LOBPCG [11]. When applying preconditioned eigensolvers , we use a diagonal ('Jacobi') preconditioner thattakes advantage of the near-diagonal structure~~ Please note that, unlike standard iterative eigensolvers, IPT does not naturally target extremal eigenvalues; instead it computes perturbations the diagonal elements of M. ~~For a discussion of Davidson-type~~

~~methods in comparison with Lanczos, see [30]. Finally, we note that shift-and-invert methods were tested on the near-diagonal problems , but proved less efficient than all methods reported here.~~In particular, to compute the lowest eigenvalue of a FCI matrix, we simply choose apply Algorithm 3.1 for i such that $M_{ii} = \min \operatorname{diag}(M)$.

~~presents our results. In all cases (dense, sparse, symmetric or non-symmetric) IPT runs much faster than ARPACK. Davidson, an algorithm designed for diagonally dominant matrices and used e. g. in quantum chemistry applications, provides a more meaningul reference point. Here we find that IPT is up to $\sim$ 5x faster at small ε. This can be explained by the fact that , altough IPT requires a similar number of iterations as Davidson, each iteration is cheaper, with just one~~ We computed the ground state of the FCI matrix for water (H_2O) in the minimal basis set “sto-3g” (with $n = 441$ in this example). (To compute its elements we used the quantum chemistry package PySCF [31].) ?? (left) shows the convergence history (residual norm $\|\mathbf{f}(\mathbf{z}) - \mathbf{z}\|$ vs. number of matrix-vector ~~multiplication per step.~~

~~IPT vs. standard iterative algorithms for the lowest-lying eigenpair of perturbative matrices of the form and increasing ε. Davidson-type methods with diagonal preconditioning outperform the Krylov-Schur algorithm on these problems, and IPT is yet faster when the perturbation is sufficiently small (or sparse).~~

6.2. ~~Parallelism.~~ ~~Underlying the performance of IPT is the high parallelism of matrix-vector and, especially, matrix-vector multiplication. Such parallelism is not shared by Householder reflections, which is why eigenvalue routines based on Hessenberg or tri-diagonal reduction do not scale as well on~~ products, matvecs) for various algorithms in PRIMME and for IPT(-ACX). In this example, both IPT and IPT-ACX converge to the desired tolerance (here $\eta = 10^{-12}$) with fewer matvecs that any of the PRIMME methods, with the latter about twice faster than the former. ?? (right) shows the corresponding timings on our multi-core ~~or GPU architectures. We illustrate this in , where the time to diagonalize a dense, non-symmetric matrix with $\varepsilon = 0.1$ is plotted as a function of the number of active CPU cores. As expected from its simple structure, IPT parallelizes just as well as matrix-matrix multiplication (almost linearly in the number of cores). On large clusters with thousands of cores, we expect IPT will outperform GEEV by several orders of magnitude.~~

~~Parallel scaling. Unlike direct methods based on Hessenberg reduction (here GEEV for a near-diagonal matrix of the form with parameters as indicated), IPT scales as well as BLAS (here matrix-matrix multiplication). As a result, the speed-up of IPT over GEEV is only limited by the number of CPU cores available.~~

6.2. ~~Accuracy.~~ ~~Finally we compared the accuracy of IPT on the full spectrum problem with LAPACK via MATLAB's **eig**. For this we used~~ machine.

We performed the same calculations with a near-diagonal ~~matrices of the form with $N = 1024$ in double precision varying $\varepsilon \in [10^{-4}, 0.2]$. For each ε we ran IPT~~ matrix considered by Morgan in his comparison of preconditioned eigensolvers [32]: M is the tri-diagonal, symmetric matrix with $M_{ii} = i$ and ~~**eig** to obtain the matrix of eigenvectors Z and corresponding eigenvalues $\Lambda = \operatorname{diag}(\lambda_n)_{1 \le n \le N}$. We then measured the accuracy of each routine through the residual error $\|MZ - Z\Lambda\|_F$, where $\|\cdot\|_F$ denotes the Frobenius norm. We find that IPT is more than an order of magnitude more accurate than LAPACK, with a median residual error of $4.4 \cdot 10^{-11}$ and $6.4 \cdot 10^{-10}$ respectively~~$M_{i,i+1} = M_{i,i-1} = 0.5$. ?? shows the results. Here IPT requires many more matvecs than Davidson methods, but IPT-ACX does not, converging equally fast as the fastest PRIMME method for this example, LOBPCG. But since LOBPCG involves additional steps per iteration (diagonalization in the subspace), IPT-ACX proves faster overall (**??**, right panel).

7. ~~Applications.~~ ~~In this final section we consider two possible applications of IPT : as a mixed-precision eigenvalue algorithm (full spectrum problem), and as a potential competitor to Davidson in quantum chemistry (lowest-lying eigenvector).~~

6.1. ~~A mixed-precision eigenvalue algorithm.~~ ~~The condition that a matrix be near-diagonal of course severely restricts the applicability of IPT, although relevant examples exist in quantum chemistry, see below. However, IPT can be used more generally as a *refinement* method for well-conditioned eigenvalue problems. Consider a generic matrix M with well separated eigenvalues, and assume given an approximation of its eigenmatrix Z_0. For instance, we could compute Z_0 using standard drivers in single precision, resulting in a single precision eigenmatrix $Z_{0,s}$. Next, we convert this matrix to double precision, denoted $Z_{0,d}$, and apply IPT to the near-diagonal matrix $M' = Z_{0,d}^{-1} M Z_{0,d}$. Once iterations have converged to a new eigenmatrix Z', we can rotate back and obtain accurate eigenvectors for M as $Z = Z_{0,d}Z'$. In practice the linear solve and matrix multiplication steps are fast, and so computing Z takes about the same time as the low-precision diagonalization $Z_{0,s}$~~But the advantage of IPT vis-a-vis standard iterative methods becomes more apparent when we request several eigenpairs rather than just one. Unlike other methods, IPT computes eigenpairs in parallel (corresponding to the columns of Z) rather than sequentially, after deflation. The benefit of this parallelism is on display in **??**, where the timing to compute k eigenvalues is compared across algorithms. For $k = 50$, IPT-ACX is already several times faster than the fastest iterative method, and this gap only increases with k.

~~Obviously, this method cannot be used for ill-conditioned eigenvalue problems, because then (*i*) diagonal gaps become large and IPT ceases to converge, and (*ii*) the condition number of $Z_{0,d}$ becomes large and accuracy is lost in the linear solve step. To measure the robustness of~~ Finally we show in **??** how IPT fails when off-diagonal elements become too large compared to diagonal gaps. For this we consider a modification of the ~~mixed-precision algorithm to eigenvalue clustering, we considered matrices~~ Morgan matrix with $M_{ii} = i$ and $M_{i,i+1} = M_{i,i-1} = \varepsilon$ and increase ε.

6.1. Full spectrum. Next we consider the problem of computing all eigenpairs of a large, dense matrix of the form

$$\underline{J_\alpha} M = \underline{Q^T D_\alpha Q}\,\cancel{\text{with}}\ \underline{D_\alpha =}\,\mathrm{diag}(\underline{10^{-\alpha n/N}}\, i)_{\underline{1\le n\le N},1\le i\le n} + \varepsilon R_n \tag{6.1}$$

where ~~α is a parameter controlling the spacing between eigenvalues and Q a random orthogonal matrix~~ R_n is a $n \times n$ matrix with uniformly distributed random entries in $[0, 1]$. For a symmetric matrix with similar properties we consider $S = (M + M^t)/2$. ~~For matrices of size $N = 1024$ we found that IPT remained applicable up to $\alpha \simeq 4$, corresponding to a minimal spectral gap of $1.5 \cdot 10^{-6}$. (For this matrix **eigs** failed to converge with the default parameters.) For all values of α below this threshold, we obtain acceptable residual errors ($\sim$ 5x larger than DGEEV) and significant speed-ups over DGEEV ($2-3$x faster), see .~~

~~Iterative refinement of eigenvectors of matrix M with well-separated eigenvalues Compute eigenvectors $Z_{0,s}$ in single precision *e.g.* using SGEEV or SSYEV Set $Z_{0,d}$ as $Z_{0,s}$ in double precision Compute $M' \leftarrow Z_{0,d}^{-1} M Z_{0,d}$ using a linear solver in double precision Compute eigenvectors Z' and Λ using IPT(M') Return eigenvectors $Z = Z_{0,d}Z'$ Return eigenvalues Λ~~

6.2. ~~Full configuration interaction.~~ ~~Finally, we tested IPT on a real-world problem from quantum chemistry, the full configuration interaction (FCI)approach to *ab initio* electronic structure calculation [1]. Given a molecule, its geometry (the position of nuclei) and a basis set, FCI provides the most accurate estimate of the energy and wavefunction of electrons available. Formally, an approximation of the eigenvectors of the Hamiltonian matrix is first computed with self-consistent field (Hartree-Fock)methods; in that basis the Hamiltonian matrix H is~~ Dense eigenproblems are normally solved with the QR or divide-or-conquer algorithm, implemented in the Linear Algebra PACKage (LAPACK) under the name (GEEV). On NVidia GPUs, the CUSOLVER [33] package provides an efficient version of divide-and-conquer for symmetric matrices (SYEVD). As already emphasized, these methods do not take advantage of any particular structure in M except symmetry; in particular they do not perform better on sparse or near-diagonal ~~(as well as symmetric), and FCI proceeds by computing iteratively the eigenvector z_0 with smallest eigenvalue λ_0, usually using the Davidson algorithm already mentioned above. Because the dimension N of the Hamiltonian matrix H for n_e electrons distributed in n_b basis sets is (n_b choose n_e) (hence grows factorially with system size), the bottleneck in iterative FCI computations is the matrix-vector product $H\mathbf{z}$. (Fig. **??** shows the sparsity pattern of a small FCI matrix.)~~matrices than on general matrices.

~~In order to reduce the number of iterations as much as possible, we applied Anderson acceleration to the fixed point iteration $\mathbf{z} \leftarrow \mathbf{f}_0(\mathbf{z})$ in Algorithm 3.1. We recall that Anderson acceleration with memory m replaces the current iterate $\mathbf{z}^{(k)}$ with a convex combination of $m_k = \min\{m, k\}$ previous iterates, with weights $\alpha^{(k-j)}$ ($0 \leq j \leq m_k$) defined by a least-squares minimization problem (see [20] and references therein) . We compared IPT with Anderson acceleration (with memory $m = 5$)with the implementation of the Davidson algorithm provided in the open-source quantum chemistry suite PySCF [31] (using the function **pyscf.lib.davidson**) ; for a python implementation of Anderson acceleration we used the code from Ref. [34]. The convergence criterion was based on the residual, namely $\|H\mathbf{z}_0 - \lambda_0\mathbf{z}_0\| \leq 10^{-8}$.~~

~~Sparsity pattern of a small FCI matrix, here for H_2O in the minimal basis set sto-3g.~~

~~Molecule Basis set N (K~~We timed multi-CPU and GPU implementations of IPT to these methods for matrices (6.1) with $\varepsilon = 10^{-2}$, double precision ($\eta = 10^{-12}$), ~~T) Davidson~~ and increasingly large dimension n. (~~K, T) IPT-AA He cc-pvqz 900~~ Both IPT and IPT-ACX diverge for $\varepsilon \gtrsim .05$) compared with these reference routines (**??**). The result is that IPT is up to 2 orders of magnitude more efficient than the non-symmetric routine GEEV, and also more efficient than the (~~9, 0.098) (8, 0.055) Be cc-pvtz 189225 (13, 1.82) (13, 1.82) H_2O sto-6g 441 (9, 0.003) (8, 0.002) H_2O 6-31G 1656369 (18, 5.36) (20, 4.75) LiH 6-31g 3025 (12, 0.028) (12, 0.027) LiH cc-pvtz 894916 (15, 39.95) (15, 41.86)~~ much faster) symmetric routine SYEVD.

~~As showed in Table 1, we find that IPT-AA converges for realistic FCI problems with a number~~ The reason for this enhanced performance on near-diagonal problems is again due to the better parallelism of IPT. For the full-spectrum problem, IPT relies on matrix-matrix products which are efficiently parallelized with BLAS 3; the Hessenberg reduction with underlies direct methods does not benefit from such parallelism. **??** shows the timing of eigendecompositions vs. that of ~~iterations and timing comparable to Davidson. One conceptual difference between the two methods, however, is that IPT is self-contained, whereas Davidson relies on other eigenvalue algorithms to obtain Ritz values~~matrix-matrix multiplications as a function of the number of CPU cores used. While this ratio increases with GEEV (indicating worse parallelism than BLAS 3), it does not with IPT (as expected).

7. Discussion. We have presented a new eigenvalue algorithm for near-diagonal matrices, be them symmetric or non-symmetric, dense or sparse. IPT can be applied to obtain a single perturbed eigenvector ~~, with performance comparable to~~ and often outperforms state-of-the-art preconditioned eigensolvers. IPT can also be applied to the full spectrum problem; in that case IPT benefits from a lower theoretical complexity and lower runtime than ~~standard methods~~ dense eigensolvers based on Hessenberg reduction. The largest ~~speed-ups are~~ speed-ups—up to two-orders of magnitude—are obtained for non-symmetric, full spectrum problems.

~~We emphasize that the near-diagonality condition is quite restrictive: off-diagonal matrix elements must be small compared to diagonal spacings. While their are important applications with this property (e.g. full configuration interaction), it is fair to say that~~ To our knowledge, IPT is ~~more akin to a refinement procedure than a full-fledged eigenvalue algorithm. Even so IPT can be useful: using a mixed-precision approach we computed the eigenvectors of a general dense matrix with random entries to double precision in a fraction of the time required by DGEEV.~~ the first full-spectrum eigensolver that is able to take initial guesses into account.

Future work should focus on stabilizing our procedure for larger perturbations. For instance, when the perturbation is too large and IPT blows up, we may shrink ε it to ε/Q for some integer Q, diagonalize the matrix with this smaller perturbation, restart the algorithm using the diagonal matrix thus obtained as initial condition, and repeat the operation Q times. This approach is similar to the homotopy continuation method for finding polynomial roots [10] and can effectively extend the domain of convergence of the present iterative scheme. Another idea is to leverage the projective-geometric structure outlined above, for instance by using charts on the complex projective space other than $z_i = 1$, which would lead to different maps with different convergence properties. A third possibility is to use the freedom in choosing the diagonal matrix D to construct maps with larger convergence domains, a trick which is known to sometimes improve the convergence of the RS series [15]. Finally, we saw that the convergence of IPT can be improved using ~~Anderson acceleration, which can be viewed as an *ad hoc* non-linear Krylov subspace method~~acceleration methods. It would be interesting to see ~~whether there exists an optimal combination of previous iterates to find the fixed points of the IPTmap~~how far such ideas can go in extending the scope of IPT.

~~IPT as a refinement algorithm for ill-conditioned problems with increasingly small eigengap. Combining single-precision GEEV with IPT gives eigenvectors with acceptable accuracy at a fraction of the computational cost of double-precision GEEV.~~

Acknowledgments. We thank Rostislav Matveev and Alexander Heaton for useful discussions.

REFERENCES

[1] A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory*. Dover Publications, 1996.

[2] D. Pozar, *Microwave Engineering (2nd edition)*. Wiley, New York, 1998.

[3] J. W. S. Rayleigh, *Theory of Sound. I (2nd ed.)*. London McMillan, 1894.

[4] T. Kato, *Perturbation Theory for Linear Operators*. Springer, 1995.

[5] G. W. Stewart and J.-G. Sun, *Matrix Perturbation Theory*. Academic Press, New York, 1990.

[6] J. W. Demmel, *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, 1997.

[7] E. R. Davidson, “The iterative calculation of a few of the lowest eigenvalues and corresponding eigenvectors of large real-symmetric matrices,” *Journal of Computational Physics*, vol. 17, pp. 87–94, jan 1975.

[8] D. R. Fokkema, G. L. Sleijpen, and H. A. Van der Vorst, “Jacobi–davidson style qr and qz algorithms for the reduction of matrix pencils,” *SIAM journal on scientific computing*, vol. 20, no. 1, pp. 94–125, 1998.

[9] G. L. Sleijpen and H. A. Van der Vorst, “A jacobi–davidson iteration method for linear eigenvalue problems,” *SIAM Review*, vol. 42, no. 2, pp. 267–293, 2000.

[10] A. Morgan, *Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems.* Society for Industrial and Applied Mathematics, jan 2009.

[11] A. Knyazev, “Recent implementations, applications, and extensions of the locally optimal block preconditioned conjugate gradient method (lobpcg),” *arXiv preprint arXiv:1708.08354*, 2017.

[12] E. M. Lifshitz and L. D. Landau, *Quantum Mechanics: Non-relativistic Theory.* Pergamon Press, 1965.

[13] M. Smerlak, “Convergence without resummation: an iterative approach to perturbative eigenvalue problems,” *arXiv preprint arXiv:2105.04972*, 2021.

[14] Y. A. Kuznetsov, *Elements of Applied Bifurcation Theory.* Springer, 2004.

[15] P. R. Surján and Á. Szabados, “Appendix to “Studies in Perturbation Theory”: The Problem of Partitioning,” in *Fundamental World of Quantum Chemistry*, pp. 129–185, Springer Netherlands, 2004.

[16] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge University Press, 2012.

[17] R. M. May, “Simple mathematical models with very complicated dynamics,” *Nature*, vol. 261, pp. 459–467, jun 1976.

[18] V. Eyert, “A comparative study on methods for convergence acceleration of iterative vector sequences,” *Journal of Computational Physics*, vol. 124, no. 2, pp. 271–285, 1996.

[19] V. Dolotin and A. Morozov, “On the shapes of elementary domains, or why mandelbrot set is made from almost ideal circles?,” *International Journal of Modern Physics A*, vol. 23, pp. 3613–3684, sep 2008.

[20] H. F. Walker and P. Ni, “Anderson Acceleration for Fixed-Point Iterations,” *SIAM Journal on Numerical Analysis*, vol. 49, pp. 1715–1735, jan 2011.

[21] N. Lepage-Saucier, “Alternating cyclic extrapolation methods for optimization algorithms,” *arXiv preprint arXiv:2104.04974*, 2021.

[22] V. Hernandez, J. E. Roman, and V. Vidal, “Slepc: A scalable and flexible toolkit for the solution of eigenvalue problems,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 31, no. 3, pp. 351–362, 2005.

[23] S. Tomov, J. Dongarra, and M. Baboulin, “Towards dense linear algebra for hybrid GPU accelerated manycore systems,” *Parallel Computing*, vol. 36, pp. 232–240, jun 2010.

[24] A. Stathopoulos and J. R. McCombs, “Primme: Preconditioned iterative multimethod eigensolver—methods and software description,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 37, no. 2, pp. 1–30, 2010.

[25] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, “Julia: A fresh approach to numerical computing,” *SIAM Review*, vol. 59, no. 1, pp. 65–98, 2017.

[26] M. Smerlak, 2022.

[27] M. Smerlak, 2022.

[28] L. D. Dalcin, R. R. Paz, P. A. Kler, and A. Cosimo, “Parallel distributed computing using python,” *Advances in Water Resources*, vol. 34, no. 9, pp. 1124–1139, 2011.

[29] G. W. Stewart, “A krylov–schur algorithm for large eigenproblems,” *SIAM Journal on Matrix Analysis and Applications*, vol. 23, no. 3, pp. 601–614, 2002.

[30] R. B. Morgan and D. S. Scott, “Generalizations of davidson’s method for computing eigenvalues of sparse symmetric matrices,” *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 817–825, 1986.

[31] Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. D. McClain, E. R. Sayfutyarova, S. Sharma, S. Wouters, and G. K. Chan, “Pyscf: the python-based simulations of chemistry framework,” 2017.

[32] R. B. Morgan, “Preconditioning eigenvalues and some comparison of solvers,” *Journal of computational and applied mathematics*, vol. 123, no. 1-2, pp. 101–115, 2000.

[33] R. B. Lehoucq, D. C. Sorensen, and C. Yang, *ARPACK Users Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods.* Society for Industrial and Applied Mathematics, 1998.

[34] J. Zhang, B. O’Donoghue, and S. Boyd, “Globally convergent type-i anderson acceleration for nonsmooth fixed-point iterations,” *SIAM Journal on Optimization*, vol. 30, no. 4, pp. 3170–

3197, 2020.

[35] P. Lax, “Linear algebra and its applications, john & sons,” *Inc., Hoboken, New Jersey*, 2007.

[36] R. Roth and J. Langhammer, “Padé-resummed high-order perturbation theory for nuclear structure calculations,” *Physics Letters B*, vol. 683, pp. 272–277, jan 2010.

Appendix A. Additional lemmas for Section 4.

Here as in the main text we consider an $n \times n$ matrix M with complex entries along with its partition $M = D + \Delta$, where D is diagonal with pairwise distinct diagonal elements. The partition defines a mapping $F\colon \mathbb{C}^{n\times n} \to \mathbb{C}^{n\times n}$ as in the main text.

Lemma A.1. *Let $M = D + \Delta$ be a matrix with a partition that obeys (4.1). Then there are no eigenvectors of M from eigenspaces of dimension higher than one as columns of the corresponding unique fixed point Z^* of F in $B_{\sqrt{2}}(I)$.*

Proof. If Z^* contains an eigenvector from an eigenspace of M with dimension higher than one, then any matrix Z obtained by a substitution in Z^* of that eigenvector with a vector from this eigenspace is a fixed point of F, too. This means that there is a whole subspace of fixed points of F of dimension higher than zero which includes Z^*. But this contradicts the uniqueness of Z^* in $B_{\sqrt{2}}(I)$. □

Lemma A.2. *If M and its partition obey (4.1) and M is defective, then the unique fixed point Z^* of F in $B_{\sqrt{2}}$ does not contain any eigenvector causing the defect.*

Proof. By Lemma (A.1), it is enough to assume that M does not have geometrically multiple eigenvalues and all its eigenvectors are isolated projective points, so all fixed points of F are isolated matrices. Let such M be defective. Let J be a Jordan normal form of M and S be the transition matrix that turns M into J. Consider a deformation $J_{\mathbf{d}}$ of J given by $J_{\mathbf{d}} = J + \operatorname{diag}(\mathbf{d})$, where $\mathbf{d} \in \mathbb{C}^n$. This induces a deformation of M given by $M_{\mathbf{d}} = M + S \operatorname{diag}(\mathbf{d}) S^{-1}$. It is clear that one can find $\mathbf{d}$ arbitrarily close to 0 (in the standard topology of $\mathbb{C}^n$) such that all diagonal entries of $J_{\mathbf{d}}$ are different. As $J_{\mathbf{d}}$ is upper triangular, its eigenvalues (and thus those of $M_{\mathbf{d}}$) are equal to its diagonal entries. But then $M_{\mathbf{d}}$ has a full eigenbasis of isolated eigenvectors. The formerly merged eigenvectors split into distinct ones. Furthermore, the new eigenvectors that split from an old defective eigenvector can be made arbitrarily close to it (as projective points in the standard topology of $\mathbb{C}P^{n-1}$) by choosing $\mathbf{d}$ close enough to 0. This is obvious from considering the eigenvectors spawned from a single block of $J_{\mathbf{d}}$ and their behavior at the limit $\mathbf{d} \to 0$.

All this implies that there is a deformation $M_{\mathbf{d}}$ of M such that in its partition $M_{\mathbf{d}} = D + \Delta_{\mathbf{d}}$, $\Delta_{\mathbf{d}} = \Delta + S \operatorname{diag}(\mathbf{d}) S^{-1}$, $\|\Delta_{\mathbf{d}} - \Delta\|$ is arbitrarily small and thus $\|G\|\|\Delta_{\mathbf{d}}\| < 3 - 2\sqrt{2}$ still holds. At the same time, Z^* splits into several arbitrarily close to it, and thus still contained in $B_{\sqrt{2}}(I)$ for small enough $\mathbf{d}$, fixed points of the mapping $F_{\mathbf{d}}$ defined by the partition, with two or more of them being full-rank and multiple associated lower rank fixed points. But this contradicts the uniqueness of the fixed point of $F_{\mathbf{d}}$ computed for the deformed matrix $M_{\mathbf{d}}$. □

Theorem A.3 ([35] Theorem 8, page 130). *Let M_ε be a differentiable matrix-valued function of real ε, a_ε an eigenvalue of M_ε of multiplicity one. Then we can choose an eigenvector $\mathbf{h}_\varepsilon$ of M_ε pertaining to the eigenvalue a_ε to depend differentiably on ε.*

The theorem guarantees that the eigenvector corresponding to an eigenvalue with multiplicity one is a differentiable function of the parameter as a projective space valued function.

Appendix B. Rayleigh-Schrödinger perturbation theory.

Here we recall the derivation of the Rayleigh-Schrödinger recursion, given *e.g.* in [36]. The idea behind the original perturbation theory is the following. Let a parametric family of matrices $M_\varepsilon = D + \varepsilon\Delta$ be given, where $\varepsilon \in \mathbb{C}$ is the parameter,

$M_\varepsilon \in \mathbb{C}^{n\times n}$, $M_0 = D$ is diagonal and is called the *unperturbed* matrix, and Δ is the *perturbation* matrix. The problem is to find eigenvectors of M_ε in the form of an asymptotic series in powers of the parameter ε. We specifically assume a generic case of the unperturbed matrix D having distinct diagonal entries. The perturbation matrix Δ can be arbitrary.

Let $\lambda_i \in \mathbb{C}$ and $\mathbf{z}_i \in \mathbb{C}^n$ respectively be the i-th eigenvalue and the i-th eigenvector of M_ε ordered such that

$$\lim_{\varepsilon\to 0} \lambda_i = \mathring{\lambda}_i, \quad \lim_{\varepsilon\to 0} \mathbf{z}_i = \mathbf{e}_i, \tag{B.1}$$

where $\mathring{\lambda}_i = D_{ii}$, $\mathbf{e}_i$ are the standard basis vectors of $\mathbb{C}^n$, and normalized according to $\langle \mathbf{e}_i, \mathbf{z}_i\rangle = 1$ for all ε (a choice known in physics as "intermediate normalization"), where by $\langle\cdot,\cdot\rangle$ we understand the standard scalar product in $\mathbb{C}^n$. Such normalization corresponds to singling out eigenvectorst from the affine charts U_i seen as affine subspaces of $\mathbb{C}^n$ as defined in Section 3 of the main text. Therefore, the matrix composed of $\mathbf{z}_i$ in the chosen order is matrix Z from the main text. The possibility to order the eigenvectors in such a way that (B.1) holds follows from the well known fact that, given the condition on D and the chosen normalization, λ_i and $\mathbf{z}_i$ are holomorphic in ε in a certain disk around 0, $\mathbf{z}_i$ are distinct, and these functions are given by series convergent in that disk [4]:

$$\lambda_i = \sum_{\ell\geq 0} \varepsilon^\ell \lambda_i^{[\ell]} \quad \text{and} \quad \mathbf{z}_i = \sum_{\ell\geq 0} \varepsilon^\ell \mathbf{z}_i^{[\ell]}. \tag{B.2}$$

As a consequence, matrix Z itself is holomorphic in the same disk and can be represented by a power series in ε

$$Z = \sum_{\ell\geq 0} \varepsilon^\ell Z^{[\ell]}. \tag{B.3}$$

Substituting expressions (B.2) into the eigenvalue equation $M\mathbf{z} = \lambda\mathbf{z}$ and making use of the Cauchy product formula, this yields $D\mathbf{z}_i^{[0]} = \lambda_i^{[0]}\mathbf{z}_i^{[0]}$ at zeroth order (hence $\lambda_i^{[0]} = \mathring{\lambda}_i$) and for $\ell \geq 1$

$$(D - \mathring{\lambda}_i I)\mathbf{z}_i^{[\ell]} = \sum_{s=1}^{\ell} \lambda_i^{[s]} \mathbf{z}_i^{[\ell-s]} - \Delta \mathbf{z}_i^{[\ell-1]},$$

where I is the unit $n \times n$ matrix.

It is convenient to expand $\mathbf{z}_i^{[\ell]}$ in the basis of the eigenvectors of D as $\mathbf{z}_i^{[\ell]} = \sum_{j=1}^n Z_{ji}^{[\ell]}\mathbf{e}_j$ with $Z_{ij}^{[0]} = \delta_{ij}$ and $Z_{ii}^{[\ell]} = 0$ for $\ell \geq 1$. By construction, $Z_{ij}^{[\ell]}$ are elements of matrices $Z^{[\ell]}$ from (B.3). This gives for each $\ell \geq 1$ and $1 \leq j \leq n$

$$(\mathring{\lambda}_j - \mathring{\lambda}_i) Z_{ji}^{[\ell]} = \sum_{s=1}^{\ell} \lambda_i^{[s]} Z_{ji}^{[\ell-s]} - (\Delta Z^{[\ell-1]})_{ji}.$$

The equation for the eigenvalues correction is extracted by swapping i with j in this equation and using $Z_{ii}^{[\ell]} = \delta_{\ell,0}$. This leads to $\lambda_i^{[\ell]} = (\Delta Z^{[\ell-1]})_{ii}$. Injecting this back into the equation above we arrive at

$$Z_{ij}^{[\ell]} = G_{ij}\Big(\sum_{s=1}^{\ell} (\Delta Z^{[s-1]})_{jj} Z_{ij}^{[\ell-s]} - (\Delta Z^{[\ell-1]})_{ij}\Big). \tag{B.4}$$

Appendix C. Iterative perturbation theory contains the RS series.

We prove $Z^{(k)} = Z^{(k)}_{\text{RS}} + \mathcal{O}(\varepsilon^{k+1})$ by induction. Obviously $Z^{(0)} = Z^{(0)}_{\text{RS}} = I$. Suppose that $Z^{(k-1)} = Z^{(k-1)}_{\text{RS}} + \mathcal{O}(\varepsilon^k)$ or, more specifically,

$$Z^{(k-1)} = \sum_{\ell=0}^{k-1} \varepsilon^\ell Z^{[\ell]} + \mathcal{O}(\varepsilon^k),$$

where the matrices $Z^{[\ell]}$ satisfy the recursion (B.4). Then from $Z^{(k)} = F(Z^{(k-1)})$ we have

$$Z^{(k)} = I + \varepsilon G \circ \left(\sum_{m=0}^{k-1} \varepsilon^m Z^{[m]} \mathcal{D} \left(\sum_{\ell=0}^{k-1} \varepsilon^\ell \Delta Z^{[\ell]} \right) - \sum_{\ell=0}^{k-1} \varepsilon^\ell \Delta Z^{[\ell]} \right) + \mathcal{O}(\varepsilon^{k+1}).$$

From this expression it is easy to see that the term of s-th order in ε in $Z^{(k)}$ is given by terms with $\ell + m = s - 1$, *viz.*

$$\varepsilon^s G \circ \left(\left(\sum_{\ell=0}^{s-1} Z^{[s-1-\ell]} \mathcal{D} \left(\Delta Z^{[\ell]} \right) \right) - \Delta Z^{[s-1]} \right).$$

This term matches exactly the RS correction term $\varepsilon^s Z^{[s]}$. This concludes the proof.

Appendix D. ~~Further explicit~~ Explicit examples.

~~The~~

D.1. A general approach to find the convergence domain.

Following the notations of the main text, consider the map $\mathbf{f}_i$ for some fixed i. An attracting equilibrium point of the corresponding dynamical system $\mathbf{z}^{(k)} = \mathbf{f}_i(\mathbf{z}^{(k-1)})$ loses its stability when the Jacobian matrix $\partial \mathbf{f}_i$ of $\mathbf{f}_i$, $(\partial \mathbf{f}_i)_{js} \equiv \partial (\mathbf{f}_i)_j / \partial z_s = \partial F_{ji} / \partial z_s$, has an eigenvalue (called *multiplier* in this case of a discrete-time dynamical system) with absolute value equal to 1 at this point. The convergence domain of the dynamical system (3.2) for the whole matrix Z of the eigenvectors is equal to the intersection of the convergence domains for its individual lines (3.1).

Consider the system of $n+1$ polynomial equations of $n+2$ complex variables (z_j for $1 \le j \le n$, ε, and μ)

$$\begin{cases} \mathbf{z} = \mathbf{f}_i(\mathbf{z}), \\ \det(\partial \mathbf{f}_i - \mu I) = 0. \end{cases}$$

The variable μ here plays the role of a multiplier of a steady state. Either by successively computing resultants or by constructing a Groebner basis with the correct lexicographical order, one can exclude the variables z_j from this system, which results in a single polynomial of two variables $(\varepsilon, \mu) \mapsto P(\varepsilon, \mu)$. This polynomial defines a complex 1-dimensional variety. The projection to the ε-plane of the real 1-dimensional variety defined by $\{P = 0, |\mu|^2 = 1\}$ corresponds to some curve C. A more informative way is to represent this curve as a complex function of a real variable t implicitly defined by $P(\varepsilon, e^{it}) = 0$.

The curve C is the locus where a fixed point of $\mathbf{f}_i$ have a multiplier on the unit circle. In particular, the fixed point that at $\varepsilon = 0$ corresponds to $z_i = 1$ and $z_j = 0$, $j \neq i$, loses its stability along a particular subset of this curve. The convergence domain of the iterative perturbation theory is the domain that is bounded by these parts of the curve and that contains 0.

D.2. The 2×2 example of the main text. According to the procedure outlined above, the set where IPT loses stability (C, the blue curve in **??**) is found to be given by the parametric equation $4\varepsilon^2 + e^{it}(2 - e^{it}) = 0$ (same equation for both eigenvectors). The cusps (return points, the points where $d\varepsilon/dt = 0$) of this curve are at $\varepsilon = \pm i/2$. In this particular case, the convergence circle of the RS perturbation theory is completely contained in the convergence domain of the iterative perturbation theory, and their boundaries intersect only at the cusp points. This convergence domain for IPT is directly related to the main cardioid of the classical Mandelbrot set: the set of complex values of the parameter c that lead to a bound trajectory of the classical quadratic (holomorphic) dynamical system $x^{(k+1)} = (x^{(k)})^2 + c$. The main cardioid of the Mandelbort set (the domain of stability of a steady state) is bounded by the curve $4c - e^{it}(2 - e^{it}) = 0$. The boundary of the stability domain of our 2×2 example is simply a conformal transform of this cardioid by two complementary branches of the square root function composed with the sign inversion. The origin of this relation becomes obvious after the parameter change $c \mapsto -\varepsilon^2$ followed by the variable change $x \mapsto \varepsilon x$. This brings the classical system to the dynamical system of the only nontrivial component of the first line for our 2×2 example: $x^{(k+1)} = \varepsilon\left((x^{(k)})^2 - 1\right)$. The nontrivial component of the second line follows an equivalent (up to the sign change of the variable) equation: $x^{(k+1)} = \varepsilon\left(1 - (x^{(k)})^2\right)$.

This particular 2×2 ~~example in the main text is~~ example is very special. In fact, any 2-dimensional case is special in the following sense. The iterative approximating sequence for $M = D + \varepsilon\Delta$ for any D and Δ takes the form

$$Z^{(k)} = \begin{pmatrix} 1 & f_1^k(0) \\ f_2^k(0) & 1 \end{pmatrix},$$

where f_j are univariate quadratic polynomial functions related by $g_1(x) = x^2 g_2(1/x)$, with $g_j(x) \equiv x - f_j(x)$. The first special feature of this recursion scheme is that it is equivalent to a 1-dimensional quadratic discrete-time dynamical system for each column of A. This implies that the only critical point of either f_j (0, when the diagonal elements of Δ are equal) is necessarily attracted by at most unique stable fixed point. The second special feature is fact that both columns have exactly the same convergence domains in the ε-plane. To see this, suppose that x_1 and x_2 are the roots of g_1. Then it follows that $1/x_1$ and $1/x_2$ are the roots of g_2. As $g_2'(1/x) = 2g_1(x)/x - g_1'(x)$, and (like for any quadratic polynomial) $g_1'(x_1) = -g_1'(x_2)$, we also see that $g_1'(x_{1,2}) = g_2'(1/x_{2,1})$, and thus $f_1'(x_{1,2}) = f_2'(1/x_{2,1})$. Therefore, the fixed point of the dynamical systems defined by $x^{(k+1)} = f_1(x^{(k)})$ corresponding to an eigenvector and the fixed point of $x^{(k+1)} = f_2(x^{(k)})$ corresponding to the different eigenvector are stable or unstable simultaneously.

These two properties are not generic when $n > 2$. Therefore, we provide another explicit example of a 3×3 matrix to foster some intuition for more general cases.

D.3. An additional 3×3 example. Consider the following parametric 3×3 matrix and its partition:

$$M = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{pmatrix} + \varepsilon \begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \end{pmatrix}.$$

The polynomial P that defines the fixed point degeneration curve C here takes the form for $\mathbf{z}_1$

$$\begin{aligned}P(\varepsilon,\mu) = {} & 63792\varepsilon^7 - 28352\varepsilon^6\mu - 68040\varepsilon^6 - 29556\varepsilon^5\mu^2 + 89352\varepsilon^5\mu \\ & - 13239\varepsilon^5 + 960\varepsilon^4\mu^3 + 14516\varepsilon^4\mu^2 - 39164\varepsilon^4\mu + 12116\varepsilon^4 \\ & + 5616\varepsilon^3\mu^4 - 26658\varepsilon^3\mu^3 + 29988\varepsilon^3\mu^2 - 546\varepsilon^3\mu - 2448\varepsilon^3 \\ & + 468\varepsilon^2\mu^5 - 3720\varepsilon^2\mu^4 + 12820\varepsilon^2\mu^3 - 17648\varepsilon^2\mu^2 + 7584\varepsilon^2\mu \\ & - 1296\varepsilon^2 - 243\varepsilon\mu^6 + 1404\varepsilon\mu^5 - 2619\varepsilon\mu^4 + 1350\varepsilon\mu^3 + 432\varepsilon\mu^2 \\ & + 108\mu^6 - 792\mu^5 + 1980\mu^4 - 1872\mu^3 + 432\mu^2,\end{aligned} \tag{D.1}$$

for $\mathbf{z}_2$

$$\begin{aligned}P(\varepsilon,\mu) = {} & 113408\varepsilon^7 - 63792\varepsilon^6\mu - 120960\varepsilon^6 + 7416\varepsilon^5\mu^2 + 53208\varepsilon^5\mu \\ & + 36424\varepsilon^5 + 6525\varepsilon^4\mu^3 - 11034\varepsilon^4\mu^2 - 13824\varepsilon^4\mu - 5664\varepsilon^4 \\ & - 3156\varepsilon^3\mu^4 - 2472\varepsilon^3\mu^3 + 10332\varepsilon^3\mu^2 + 3696\varepsilon^3\mu + 3088\varepsilon^3 \\ & - 72\varepsilon^2\mu^5 + 1800\varepsilon^2\mu^4 - 1800\varepsilon^2\mu^3 - 2088\varepsilon^2\mu^2 - 1296\varepsilon^2\mu \\ & - 576\varepsilon^2 + 128\varepsilon\mu^6 - 24\varepsilon\mu^5 - 736\varepsilon\mu^4 - 120\varepsilon\mu^3 + 1328\varepsilon\mu^2 \\ & - 72\mu^6 + 108\mu^5 + 360\mu^4 - 432\mu^3 - 288\mu^2,\end{aligned} \tag{D.2}$$

and for $\mathbf{z}_3$

$$\begin{aligned}P(\varepsilon,\mu) = {} & 35440\varepsilon^7 - 42528\varepsilon^6\mu - 37800\varepsilon^6 - 32360\varepsilon^5\mu^2 + 110080\varepsilon^5\mu \\ & - 23295\varepsilon^5 + 29112\varepsilon^4\mu^3 - 4800\varepsilon^4\mu^2 - 88614\varepsilon^4\mu + 51024\varepsilon^4 \\ & + 14640\varepsilon^3\mu^4 - 78760\varepsilon^3\mu^3 + 116760\varepsilon^3\mu^2 - 52920\varepsilon^3\mu + 5400\varepsilon^3 \\ & - 2376\varepsilon^2\mu^5 - 10152\varepsilon^2\mu^4 + 70296\varepsilon^2\mu^3 - 101496\varepsilon^2\mu^2 + 41904\varepsilon^2\mu \\ & - 864\varepsilon^2 - 1080\varepsilon\mu^6 + 7920\varepsilon\mu^5 - 19620\varepsilon\mu^4 + 19440\varepsilon\mu^3 - 6480\varepsilon\mu^2 \\ & + 1296\mu^6 - 7992\mu^5 + 17712\mu^4 - 16416\mu^3 + 5184\mu^2.\end{aligned} \tag{D.3}$$

As we can see, the dynamical systems for all three columns of Z ($\mathbf{z}_1$, $\mathbf{z}_2$, and $\mathbf{z}_3$) have different domains of convergence in the ε plane. The corresponding curves are depicted in **??**, **??**, and **??**, respectively.

There are differences also in curves for individuals columns with those for the 2×2 case. Note that a curve C does not contain enough information to find the convergence domain itself. The domains on **??**–**??** were found empirically. Of course, they are bound by some parts of C and include the point $\varepsilon = 0$. The reason for some parts of C not forming the boundary of the domain is that its different parts correspond to different eigenvectors. In other words, they belong to different branches of a multivalued eigenvector function of ε, the cusps being the branching points.

Consider as a particular example the case $\mathbf{z}_2$. The curve C intersects itself at $\varepsilon \approx -0.49$. Above the real axis about this point, one of the two intersecting branches of the curve form the convergence boundary. Below the real axis, the other one takes its place. This indicates that the two branches correspond to different multipliers of the same fixed point. When $\Im\varepsilon > 0$, one of them crosses the unitary circle at the boundary of the convergence domain; when $\Im\varepsilon < 0$, the other one does. At $\varepsilon \approx -0.49$, both of them cross the unitary circle simultaneously. This situation corresponds, thus, to a Neimark-Sacker bifurcation (the discrete time analog of the Andronov-Hopf bifurcation). Both branches are in fact parts of the same continuous curve that passes through the point $\varepsilon \approx 0.56$. Around this point, the curve poses no problem to the convergence of the dynamical system. The reason for this is that an excursion around cusps (branching points of eigenvectors) permutes some eigenvectors. As a consequence, the curve at $\varepsilon \approx -0.49$ corresponds to the loss of stability of the eigenvector that is a continuation of the unique stable eigenvector at $\varepsilon = 0$ by the path $[0, -0.49]$. At the same time, the same curve at $\varepsilon \approx 0.56$ indicates a unitary by absolute value multiplier of an eigenvector that is not a continuation of the initial one by the path $[0, 0.56]$.

Not all features of the curves depicted this 3×3 case are generic either. The particular symmetry with respect to the complex conjugation of the curve and of its cusps is not generic for general complex matrices D and Δ, but it is a generic feature of matrices with real components. In this particular case, due to this symmetry, the only possible bifurcations for real values of ε are the flip bifurcations (a multiplier equals to -1, typically followed by the cycle doubling cascade), the Neimark-Sacker bifurcation (two multipliers assume complex conjugate values $e^{\pm it}$ for some t), and, if the matrices are not symmetric, the fold bifurcation (a multiplier is equal to 1). With symmetric real matrices, the fold bifurcation is not encountered because the cusps cannot be real but instead form complex conjugated pairs. These features are the consequence of the behavior of $\det(D + \varepsilon\Delta - xI)$ with respect to complex conjugation and from the fact that symmetric real matrices cannot have nontrivial Jordan forms.

Likewise, Hermitian matrices result in complex conjugate nonreal cusp pairs, but the curve itself is not necessarily symmetric. As a result, there are many more ways for a steady state to lose its stability, from which the fold bifurcation is, however, excluded. Generic bifurcations at $\varepsilon \in \mathbb{R}$ here consist in a multiplier getting a value e^{it} for some $t \neq 0$. The situation for general complex matrices lacks any symmetry at all. Here steady states lose their stability by a multiplier crossing the unit circle with any value of t, and thus the fold bifurcation, although possible, is not generic. It is generic for one-parameter (in addition to ε) families of matrices.

As already noted, for the holomorphic dynamics of any 2×2 case the unique critical point is guaranteed to be attracted by the unique attracting periodic orbit, if the latter exists. This, in turn, guarantees that for any Δ with zero diagonal the iteration of F starting from the identity matrix converges to the needed solution provided that ε is in the convergence domain. This is not true anymore for $n > 2$, starting already from the fact that there are no discrete critical points in larger dimensions. The problem of finding a good initial condition becomes non-trivial. As can be seen on **??**, the particular 3×3 case encounters this problem for the second column ($\mathbf{z}_2$). The naive iteration with $Z^{(0)} = I$ does not converge to the existing attracting fixed point of the dynamical system near some boundaries of its convergence domain. Our current understanding of this phenomenon is the crossing of the initial point by the attraction basin boundary (in the $\mathbf{z}$-space). This boundary is generally fractal. Perhaps this explains the eroded appearance of the empirical convergence domain of the autonomous iteration.

To somewhat mitigate this complication, we applied a nonautonomous iteration scheme in the form, omitting details, $\mathbf{z}^{(k+1)} = \mathbf{f}_2(\mathbf{z}^{(k)}, \varepsilon(1-\alpha^k))$ with $\mathbf{z}^{(0)} = (0, 1, 0)^T$, where α is some positive number $\alpha < 1$, so that $\lim_{k\to\infty} \varepsilon(1 - \alpha^k) = \varepsilon$, and we explicitly indicated the dependence of $\mathbf{f}_2(\mathbf{z}, \varepsilon)$ on ε. The idea of this *ad hoc* approach is the continuation of the steady state in the extended $(\mathbf{z}, \varepsilon)$-phase space from values of ε that put $\mathbf{z}^{(0)}$ inside the convergence domain of that steady state. Doing so, we managed to empirically recover the theoretical convergence domain (see **??**).

Finally, we would like to point out an interesting generic occurrence of a unitary multiplier without the fold bifurcation. For $\mathbf{z}_1$, this situation takes place at $\varepsilon \approx 0.45$, for $\mathbf{z}_2$ at $\varepsilon \approx 0.56$, and for $\mathbf{z}_3$ at $\varepsilon \approx 1.2$. All three points are on the real axis, as is expected from the symmetry considerations above. There is no cusp at these points and no fold bifurcations (no merging of eigenvectors), as it should be for symmetric real matrices. Instead, another multiplier of the same fixed point goes to infinity at the same value of ε (the point becomes super-repelling). As a result, the theorem of the reduction to the central manifold is not applicable. ~~The convergence domain on the ε-plane for the first column of Z (the first eigenvector $\mathbf{z}_1$) for the 3×3 example. The Mandelbrot-like set (domain where orbits remain bounded) of the iterative scheme is shown in black and grey. The empirical convergence domain is shown in black. Its largest component corresponds to the stability of a steady state (the applicability domain of the iterative method). Small components correspond to stability of various periodic orbits. Various shades of grey show the values of ε that lead to divergence to infinity (the darker the slower the divergence). In red are the values of ε where the matrix is non-diagonalizable. Same as for the second eigenvector $\mathbf{z}_2$ (the second column of Z). Small components correspond to stability of various periodic orbits. Various shades of grey show the values of ε that lead to divergence to infinity (the darker the slower the divergence). Same as for the third eigenvector $\mathbf{z}_3$ (the third column of Z) .~~

Appendix E. Exceptional points and cusps of C.

In a large part of this section we drop the previously adopted notation conventions for vectors. We use instead the more appropriate notation style from differential geometry.

We will prove that if M is defective at some value of ε, then $d\varepsilon/dt = 0$ for the curve C at that point (as above, μ is a multiplier of the dynamical perturbation theory and C is defined by $\mu = e^{it}$ and is locally considered as a curve $t \mapsto \varepsilon(t)$).

Fix i and consider the dynamical system for the i-th column only. Let $\mathbf{x} = (x_1, \dots, x_{n-1})$ be a tuple of affine coordinates in the affine chart U_i. Note that the coordinates are rearranged in comparison to z_j: $x_1 = z_1$, …, $x_{i-1} = z_{i-1}$, $x_i = z_{i+1}$, …, $x_{n-1} = z_n$. We consider the representation of the corresponding dynamical system in U_i too. Let it be given by the tuple of functions $\mathbf{h} = (h_1, \dots, h_{n-1})$ that correspond to functions in $\mathbf{f}_j$ with $(\mathbf{f}_i)_i$ omitted ($(\mathbf{f}_i)_i$ is trivially 1 in U_i). The same rearrangement is implied for h_j as for x_j. Thus, the dynamical system is defined by $\mathbf{x}^{(k+1)} = \mathbf{h}(\mathbf{x}^{(k)})$ and the stationary states are defined by the system of equations $\mathbf{x} = \mathbf{h}(\mathbf{x})$.

Consider $\mathbb{C}^{n+1}$ with coordinates $(x_1, \dots, x_{n-1}, \varepsilon, \nu)$, where $\nu \equiv \mu - 1$, as a complex analytic manifold with these coordinates as global holomorphic coordinates on it. Let us define polynomial functions

$$\mathcal{F}_j \colon \mathbb{C}^{n+1} \to \mathbb{C}, (\mathbf{x}, \varepsilon, \nu) \mapsto h_j(\mathbf{x}, \varepsilon) - x_j,$$

and $\mathcal{F} = (\mathcal{F}_1, \dots, \mathcal{F}_{n-1})$. Let us denote $J \equiv \dfrac{\partial \mathcal{F}}{\partial \mathbf{x}} \equiv \dfrac{\partial(\mathcal{F}_1, \dots, \mathcal{F}_{n-1})}{\partial(x_1, \dots, x_{n-1})}$ the Jacobian matrix of $\mathcal{F}$ with respect to variables x_j. Let I be the unitary $(n-1) \times (n-1)$ matrix.

Consider the complex 1-dimensional variety $\mathcal{C} \subset \mathbb{C}^{n+1}$ defined by the polynomial system

$$\begin{cases} \mathcal{F} = 0, \\ \det(J - \nu I) = 0. \end{cases}$$

Curve C is the projection to the ε-plane of the real 1-dimensional variety $\tilde{C} = \mathcal{C} \cap \{|\nu+1|^2 = 1\}$ in $\mathbb{C}^{N+1}$ considered as $\mathbb{R}^{2(N+1)}$.

First, note that if M is defective at some value of ε, then the system of equations $\mathcal{F} = 0$ (with this values of ε fixed and considered for unknowns $x \in \mathbb{C}^{n-1}$) has a root x of multiplicity greater than 1. This means that the hyperplanes $\{\mathcal{F}_j = 0\}$ are not in general position at the intersection that corresponds to this root, which implies $\det \partial\mathcal{F}/\partial\mathbf{x} = 0$. Therefore, the point $p \in \mathbb{C}^{n+1}$ with the same x and ε and with $\nu = 0$ belongs to $\mathcal{C}$ and represents this non-diagonalizability of M.

Let d denote the exterior derivative and $\wedge$ denote the exterior product on the complex of holomorphic exterior forms $\Omega^\bullet(\mathbb{C}^{n+1})$. Alternatively, one may treat it in purely axiomatic way as the Kähler differential on the algebra of holomorphic functions on $\mathbb{C}^{n+1}$ with the appropriate factorization in the end. Let $p \in \mathcal{C}$ be a point of geometric degeneration of M with $\nu = 0$ as above.

Theorem E.1. *Assume that the following nondegeneration condition holds:* $d_p \det J \wedge \bigwedge_j d_p\mathcal{F}_j \neq 0$. *Then* $\mathcal{C}$ *can be locally parametrized by* ν *around* p *and, with this parametrization,* $d\varepsilon/d\nu = 0$ *at* p.

Proof. Let $T_p\mathbb{C}^{n+1}$ be the holomorphic tangent space to $\mathbb{C}^{n+1}$ at p, that is the tangent space spanned by the holomorphic vector fields $\partial_j \equiv \partial/\partial x_j$, $\partial_\varepsilon \equiv \partial/\partial\varepsilon$, $\partial_\nu \equiv \partial/\partial\nu$ at p. Let us denote $\iota_u\sigma_p$ the contraction of a holomorphic form σ ($\sigma_p \in \bigwedge_p^\bullet \mathbb{C}^{n+1}$) with a tangent vector $u \in T_p\mathbb{C}^{n+1}$ at point p.

Let us denote $\omega_p \equiv d_p \det(J - \nu I) \wedge \bigwedge_j d_p\mathcal{F}_j$ and $\varpi_p \equiv d_p \det J \wedge \bigwedge_j d_p\mathcal{F}_j$. By the premise, $\varpi_p \neq 0$, which also implies $\omega_p \neq 0$. Indeed, the free term (with respect to ν) of the polynomial $\det(J - \nu I)$ is equal to $\det J$, and thus

$$d_p \det(J - \nu I) = \partial_\nu \det(J - \nu I)|_p \, d_p\nu + d_p \det J, \tag{E.1}$$

where the two terms are linearly independent because $\partial_\nu \det J = 0$. Therefore, as non of $\mathcal{F}_j$ depends on ν, ω_p differs from ϖ_p by an addition of a linearly independent term.

Let $v \in T_p\mathbb{C}^{n+1}$ be a nonzero vector with coordinates $(v^i, v^\varepsilon, v^\nu)$ tangent to $\mathcal{C}$. It means that $v \det(J - \nu I) = v\mathcal{F}_j = 0$, where by vf we denote the action of a vector v on a function f. This, in turn, implies $\iota_v\omega_p = 0$.

As all $\mathcal{F}_j$ depend only on x and ε, we have $d_p\varepsilon \wedge \bigwedge_j d_p\mathcal{F}_j = \det J|_p \, d_p\varepsilon \wedge \bigwedge_j d_p x_j = 0$, and thus $d_p\varepsilon \wedge \omega_p = 0$. Therefore, we have

$$\iota_v(d_p\varepsilon \wedge \omega_p) = -d_p\varepsilon \wedge \iota_v\omega_p + v^\varepsilon\omega_p = v^\varepsilon\omega_p = 0.$$

This implies $v^\varepsilon = 0$.

On the other hand, by (E.1) we have

$$d_p\nu \wedge d_p \det(J - \nu I) = d_p\nu \wedge d_p \det J,$$

and thus $d_p\nu \wedge \omega_p = d_p\nu \wedge \varpi_p \neq 0$. But $d\nu \wedge \omega \in \Omega^{n+1}(\mathbb{C}^{n+1})$, and therefore, for any holomorphic tangent vector $u \in T_p\mathbb{C}^{n+1}$, $u \neq 0$ is equivalent to $\iota_u(d_p\nu \wedge \omega_p) \neq 0$. This results in

$$\iota_v(d_p\nu \wedge \omega_p) = -d_p\nu \wedge \iota_v\omega_p + v^\nu\omega_p = v^\nu\omega_p \neq 0,$$

and thus $v^\nu \neq 0$.

By the holomorphic implicit function theorem, $\mathcal{C}$ can be holomorphically parametrized by ν in a neighborhood of p. Together with $v^{\varepsilon} = 0$ it implies that we have $d\varepsilon/d\nu|_p = 0$ on $\mathcal{C}$. □

Now, consider a smooth real curve parametrized by a real parameter t on the complex ν-plane that without degeneracy passes through 0. This curve is lifted to $\mathcal{C}$ and the resulting smooth real curve is parametrized by t. From $d\varepsilon/d\nu = 0$ we conclude that $d\varepsilon/dt = 0$ at p too. In our case we have the curve $\mu(t) = e^{it}$ or $\nu(t) = e^{it}-1$, which passes through $\mu = 1$ at $t = 0$. $\tilde{C}$ is projected without degeneration to $\mathbb{C} \times \mathbb{R} \ni (\varepsilon, t)$ locally near p and then to $\mathbb{C} \ni \varepsilon$ with degeneration at the projection of p given by $d\varepsilon/dt = 0$.