Random Matrices
MSRI Publications
Volume **65**, 2014

How long does it take to compute the eigenvalues of a random symmetric matrix?

CHRISTIAN W. PFRANG, PERCY DEIFT AND GOVIND MENON

We present the results of an empirical study of the performance of the QR algorithm (with and without shifts) and the Toda algorithm on random symmetric matrices. The random matrices are chosen from six ensembles, four of which lie in the Wigner class. For all three algorithms, we observe a form of universality for the deflation time statistics for random matrices within the Wigner class. For these ensembles, the empirical distribution of a normalized deflation time is found to collapse onto a curve that depends only on the algorithm, but not on the matrix size or deflation tolerance provided the matrix size is large enough. For the QR algorithm with the Wilkinson shift, the observed universality is even stronger and includes certain non-Wigner ensembles. Our experiments also provide a quantitative statistical picture of the accelerated convergence with shifts.

1. Introduction

We present the results of a statistical study of the performance of the QR and Toda eigenvalue algorithms on random symmetric matrices. Our work is mainly inspired by progress in quantifying the "probability of difficulty" and "typical behavior" for several numerical algorithms [Demmel 1988; Goldstine and von Neumann 1951]. This approach has led to a deeper understanding of the efficacy of fundamental numerical algorithms such as Gaussian elimination and the simplex method [Rudelson and Vershynin 2008; Sankar et al. 2006; Smale 1983; Tao and Vu 2010]. It has also stimulated new ideas in random matrix theory [Dumitriu and Edelman 2002; Edelman 1988; Edelman and Sutton 2007]. Testing eigenvalue algorithms with random input continues this effort. In related

The work of Pfrang and Menon was supported in part by NSF grant DMS 07-48482. The work of Deift was supported in part by NSF grant DMS 10-01886. Deift also acknowledges support by a grant from The Simonyi Fund at the Institute for Advanced Study in Princeton.

MSC2010: 65F15, 65Y20, 82B44, 60B20.

Keywords: symmetric eigenvalue problem, QR algorithm, Toda algorithm, matrix sign algorithm, random matrix theory.

work [Pfrang 2011], we have also studied the performance of a version of the matrix sign algorithm. However, these results are of a different character, and apart from some theoretical observations, we do not present any experimental results for this algorithm (see [Pfrang 2011] for more information). Our study is empirical — a study of the eigenvalue problem from the viewpoint of complexity theory is presented in [Armentano 2014].

1.1. ***Algorithms and ensembles.*** It is natural to study the QR algorithm because of its elegance and fundamental practical importance. But in fact all the algorithms we study are linked by a common framework. In each case, an initial matrix L_0 is diagonalized via a sequence of isospectral iterates L_m. The gist of the framework is that the L_m correspond exactly to the flow of a completely integrable Hamiltonian system evaluated at integer times. The Hamiltonian for these flows is of the form tr $G(L)$ where G is a real-valued function defined on an interval. Different choices of G generate different algorithms: $G(x) = x(\log x - 1)$ yields unshifted QR, $G(x) = x^2/2$ yields Toda, and $G(x) = |x|$ yields the matrix sign algorithm. As noted above, we will not present any numerical experiments on the matrix sign algorithm (but see Section 2). We note that the practical implementation of the QR algorithm requires an efficient shifting strategy. Our work includes a study of the QR algorithm with the Wilkinson shift as discussed below.

Initial matrices are drawn from six ensembles that arise in random matrix theory. These are listed below in Section 2.4. For many random matrix ensembles, as the size of the matrix grows, the density of eigenvalues and suitably rescaled fluctuations have limiting distributions that may be computed explicitly. Four of the ensembles we study consist of random matrices with independent entries subject to the constraint of symmetry. The law of these entries is chosen so that these ensembles have the Wigner semicircle law as limiting spectral density. We say that these ensembles are in the *Wigner class*. Numerical experiments with these ensembles are contrasted with two ensembles that do not belong to the Wigner class.

1.2. ***Deflation and QR with the Wilkinson shift.*** In evaluating these algorithms we focus on the statistics of deflation. Given a real, symmetric, $n \times n$ matrix L and an integer k between 1 and n, we write

$$L = \begin{pmatrix} L_{11} & L_{12} \\ L_{12}^T & L_{22} \end{pmatrix}, \quad \tilde{L} = \begin{pmatrix} L_{11} & 0 \\ 0 & L_{22} \end{pmatrix}, \tag{1}$$

where L_{11} is a $k \times k$ block. Let λ_j and $\tilde{\lambda}_j$, $j = 1, \ldots, n$, denote the eigenvalues of L and $\tilde{L}$. For a fixed tolerance $\epsilon > 0$ we say that L is deflated to $\tilde{L}$ when the off-diagonal block L_{12} is so small that $\max_j |\lambda_j - \tilde{\lambda}_j| < \epsilon$. The *deflation*

time is the number of iterations m before L_m can be deflated by a tolerance $\epsilon > 0$ at some index k. The *deflation index* is this value of k. Since the iterative eigenvalue algorithms correspond to Hamiltonian flows, there is also a natural notion of deflation time for the Hamiltonian flows (see equations (17) and (18) below).

Let us now explain why deflation serves as a useful measure of the time required to compute the eigenvalues of a matrix. The cost of practical computation requires an analysis of algorithms, hardware and software. In our study, we only focus on the algorithm, and "time" is taken to mean the number of iterations required for convergence. In our experiments we have observed that the QR and Toda algorithms deflate a matrix at the upper-left or lower-right corner with high probability. The deflation index for the shifted QR algorithm is $n-1$ with overwhelming probability. The deflation index for unshifted QR is also typically $n-1$ (see Figures 19 and 20). As a consequence, the deflation time is typically the same as the time taken to compute an eigenvalue. We then expect that the time taken to compute all eigenvalues with these algorithms is determined by n deflations. By contrast, we find that the matrix sign algorithm typically deflates a matrix in the middle and does not immediately yield any eigenvalues. Instead, these are obtained after a divide-and-conquer procedure that consists of approximately $\log_2 n$ deflations. Thus, for all these algorithms a finite sequence of deflation times determines the number of iterations necessary to compute eigenvalues. We must note however, that we do not track all deflations in our experiments, only the first. This restriction is necessary to keep the datasets manageable as n increases. A more extensive study that tracks all deflation times for these algorithms will certainly yield further interesting information. Finally, as we show in Section 2.6 below, the notion of deflation time is also of theoretical value since it is the starting point for an analysis of the expected number of iterations for eigenvalue algorithms that is similar in spirit to [Smale 1983].

The convergence of the QR algorithm is greatly accelerated by shifts. We will only consider the Wilkinson shift, i.e., the shift is the eigenvalue of the 2×2 lower diagonal corner of the matrix that is closer to L_{nn}. The QR algorithm on tridiagonal matrices is cubically convergent with this choice of shift (this is generically true [Wilkinson 1968]; see also [Leite et al. 2010] for a more careful analysis). As noted above, the unshifted QR algorithm deflates at index $n-1$ with very high probability. Since the Wilkinson shift utilizes the lower 2×2 block of the matrix, the number of the iterations required for shifted QR, as opposed to unshifted QR, to deflate is far smaller. While such acceleration of convergence is well-known, some features of our experiments still come as a surprise. For example, a striking feature of Figures 1 and 2 is that the number of iterations required to deflate a random matrix with the QR

algorithm (shifted and unshifted) is almost independent of n for matrices as large as 190×190.

1.3. ***Universality.*** Our main empirical findings concern universal fluctuations in the deflation time distribution for the QR algorithm (shifted and unshifted) and the Toda algorithm for ensembles in the Wigner class. We sample the deflation time for a range of matrix size and deflation tolerance combinations and normalize these empirical distributions to mean zero and variance one. The resulting histograms have the same general shape and in particular, the same tails on the right side (see in particular Figures 4, 7 and 10). In other words, *the fluctuations in deflation time are universal*. For the Toda and unshifted QR algorithm, the observed limiting fluctuations for Wigner and non-Wigner ensembles are distinct (see Figures 6 and 9). In addition, we find that the universal distributions for Wigner ensembles have exponential tails for unshifted QR and Gaussian tails for Toda (Figures 6 and 12). Universality of the tails is quantified with a statistical methodology developed in [Clauset et al. 2009]. Quite remarkably, for the (Wilkinson) shifted QR algorithm, the observed universality is stronger: to a good approximation *all* tested ensembles show the same limiting distribution (see Figure 9).

The origin of such universality is not clear. We do not understand fully if our results are connected with the now familiar universality theorems of random matrix theory such as those that describe fluctuations in the bulk and at the edge of the spectrum for the Wigner ensembles [Erdős and Yau 2012; Mehta 2004; Tracy and Widom 1994]. Unlike these universality theorems, where the mean and variance are known theoretically, in our work the mean and variance of the deflation time are computed empirically and we have not yet been able to determine analytically how these depend on n. It does appear however that the mean deflation time is linearly proportional to $\log \epsilon$ (see Figures 13 and 15).

More broadly, our experiments are suggestive of a wider class of questions concerning universality of fluctuations for computations in numerical linear algebra. For example, in similar experiments to be reported elsewhere, one of the authors (P.D.) and Sheehan Olver have studied the solution x to the linear equation $Ax = b$ empirically, when A is a random positive symmetric matrix and b is a random vector. They compute the solution using the conjugate gradient method and observe universal fluctuations in the number of iterations required for convergence, independent of the choice of ensemble for A and b.

We now discuss the algorithms and ensembles in greater detail. This is followed by a description of the results in Section 3. The implementation of the algorithms is discussed briefly in Section 4.

2. Algorithms, ensembles and deflation statistics

2.1. ***Notation.*** We denote the space of real, symmetric $n \times n$ matrices by $\mathrm{Symm}(n)$ and the space of real $n \times m$ matrices by $\mathbb{R}^{n\times m}$. Matrices in $\mathrm{Symm}(n)$ are denoted L or M and the iterates of an eigenvalue algorithm are denoted L_m or M_m, $m = 0, 1, 2, \ldots$. We use Q to denote an orthogonal matrix and R an upper triangular matrix with positive diagonal entries, typically with reference to a QR factorization. We use $\sigma(L)$ to denote the spectrum of L. The spectral decomposition of $L \in \mathrm{Symm}(n)$ is written

$$L = U \Lambda U^T,$$

where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ the matrix of eigenvalues and U denotes the orthogonal matrix of eigenvectors of L. We also use the following standard notation. The $n \times n$ identity matrix is I; the standard basis in $\mathbb{R}^n$ is $(e_1, \ldots, e_n)$; the unit sphere in $\mathbb{R}^{n+1}$ and its positive orthant are S^n and S^n_+, respectively; the symmetric group of order n is S_n.

When $L \in \mathrm{Symm}(n)$ is tridiagonal, we let $(a_1, \ldots, a_n)$ be its diagonal entries and $(b_1, \ldots, b_{n-1})$ its off-diagonal entries. A Jacobi matrix is a tridiagonal matrix with $b_i > 0$ for all i. The space of Jacobi matrices is denoted $\mathrm{Jac}(n)$. We use $u = U^T e_1$ to denote the first row of the matrix of eigenvectors. When $L \in \mathrm{Jac}(n)$, $\sigma(L)$ is simple and we may assume that $\lambda_1 > \lambda_2 > \cdots > \lambda_n$. Moreover, all the components u_i are nonzero and we may assume that $u_i > 0$ (this is also generically true for $L \in \mathrm{Symm}(n)$). Let $\mathcal{M}$ denote the manifold $\{(\Lambda, u) \in \mathbb{R}^n \times S^n_+ | \lambda_1 > \lambda_2 > \cdots > \lambda_n\}$. It is a basic result in the spectral and inverse spectral theory of Jacobi matrices that the matrix L can be reconstructed if Λ and u are given. More precisely, the spectral map $\mathcal{S} : \mathrm{Jac}(n) \to \mathcal{M}$ defined by $L \mapsto (\Lambda, u)$ is a diffeomorphism [Deift et al. 1983, Theorem 2].

We use the following standard notation for probabilistic notions. The phrase independent and identically distributed is abbreviated to iid. A normal random variable with mean μ and variance σ^2 is denoted $\mathcal{N}(\mu, \sigma^2)$; a Bernoulli random variable that is ± 1 with probability $1/2$ is denoted $\mathcal{B}$; a random variable with the χ-distribution with parameter k is denoted χ_k. The notation $X \sim Y$ means that X has the same law as Y.

2.2. ***The QR algorithm and Hamiltonian eigenvalue algorithms.*** We assume the reader is familiar with the QR algorithm (excellent textbook presentations are [Demmel 1997; Golub and Van Loan 1996; Trefethen and Bau 1997]). In the unshifted QR algorithm the iterates M_m are generated through QR factorizations and matrix multiplication in the reverse order:

$$Q_m R_m = M_m, \quad M_{m+1} = R_m Q_m, \quad m = 0, 1, 2, \ldots. \tag{2}$$

The shifted QR algorithm relies on a shift μ_m at each step, and the modified steps

$$Q_m R_m = M_m - \mu_m I, \quad M_{m+1} = R_m Q_m + \mu_m I, \quad m = 0, 1, 2, \ldots. \tag{3}$$

Typical shifts, such as the Wilkinson shift, are constructed from the lower 2×2 block of M_m [Golub and Van Loan 1996, p. 418].

In the early 80s it was discovered that the QR algorithm is intimately connected with integrable Hamiltonian systems [Symes 1980; 1981/82; Deift et al. 1986; 1983; Nanda 1985]. We summarize these results below. An expanded presentation of these connections may be found in [Deift et al. 1996; 1993; Pfrang 2011]. A different exposition that explains these ideas in a fashion "intrinsic" to numerical linear algebra is [Watkins 1984].

Assume G is a piecewise smooth real-valued function defined on an interval, and set $g = G'$. If g is defined on $\sigma(L)$, we define $g(L) := U g(\Lambda) U^T$. Let M_- denote the strictly lower triangular part of the square matrix M, and $\mathrm{pr}_{\mathfrak{k}} M := M_-^T - M_-$, the projection of M onto skew-symmetric matrices. We then consider the ordinary differential equation

$$\dot{L} = [\mathrm{pr}_{\mathfrak{k}} g(L), L]. \tag{4}$$

Equation (4) defines a completely integrable Hamiltonian flow on the space of (generic) symmetric matrices with Hamiltonian $H(L) = \mathrm{tr}\ G(L)$ and symplectic structure detailed in [Deift et al. 1986]. This flow is connected to the unshifted QR algorithm as follows.

Theorem 1. *Let g be a real-valued function defined on $\sigma(L_0)$. Then*

(a) *The solution to Equation* (4) *with initial condition L_0 is an isospectral deformation*

$$L(t) = Q(t)^T L_0 Q(t), \tag{5}$$

where the orthogonal matrix $Q(t)$ is given by the unique QR factorization

$$e^{tg(L_0)} = Q(t) R(t), \quad t \geq 0, \tag{6}$$

that has $Q(0) = I$ and depends smoothly on t.

(b) *At integer times $m = 0, 1, 2 \ldots$, the solution $L(m)$ satisfies*

$$e^{g(L(m))} = M_m, \tag{7}$$

where M_m is the m-th step of the QR algorithm applied to the initial matrix $M_0 = e^{g(L_0)}$.

(c) *Assume that the spectrum $\sigma(L_0)$ is simple and that g is injective on $\sigma(L_0)$. Then $L_\infty = \lim_{t\to\infty} L(t)$ is a diagonal matrix consisting of the eigenvalues of L_0.*

The case of tridiagonal matrices is of practical and theoretical importance. When L_0 is tridiagonal, so is $L(t)$, and the flow can be linearized using the spectral map $\mathcal{S}$ for Jacobi matrices.

Theorem 2. *Assume $L_0 \in \mathrm{Jac}(n)$. Then the solution $L(t)$ to* (4) *is an isospectral deformation $L(t) = U(t)^T \Lambda U(t)$ and the evolution of $u(t) = U(t)^T e_1$ and $L(t)$ is given explicitly by*

$$u(t) = \frac{e^{tg(\Lambda)}u_0}{\|e^{tg(\Lambda)}u_0\|}, \quad L(t) = \mathcal{S}^{-1}(\lambda, u(t)). \tag{8}$$

Assume g is injective on $\sigma(L_0)$. Then

$$\lim_{t\to\infty} L(t) = \mathrm{diag}(\lambda_{\sigma_1}, \ldots, \lambda_{\sigma_n}), \tag{9}$$

where $\sigma \in S_n$ is the permutation such that $g(\lambda_{\sigma_1}) > \cdots > g(\lambda_{\sigma_n})$.

Theorem 1 and Theorem 2 may be used to develop numerical schemes. The main observation is that each choice of a Hamiltonian $H(L) = \mathrm{tr}\ G(L)$ corresponds to a choice of an algorithm. In particular, we have:

(1) the *unshifted QR algorithm*: $g(x) = \log x$, $G(x) = x(\log x - 1)$ and $H_{\mathrm{QR}}(L) = \mathrm{tr}\,[L \log L - L]$ [Nanda 1985];

(2) the *Toda algorithm*: $g(x) = x$, $G(x) = x^2/2$ and $H_{\mathrm{Toda}}(L) = \frac{1}{2}\,\mathrm{tr}\ L^2$ (in this case, (4) describes the evolution of the Toda lattice [Moser and Zehnder 2005]);

(3) the *matrix sign algorithm*: $g(x) = \mathrm{sign}\, x$, $G(x) = |x|$ and $H_{\mathrm{sign}}(L) = \mathrm{tr}\ |L|$.

Of course, each step of the shifted QR algorithm, $L \mapsto L - \mu I$, is Hamiltonian, with Hamiltonian $H_{\mathrm{QR,shift}}(L) = H_{\mathrm{QR}}(L - \mu I)$. While every function G defines a Hamiltonian not all choices are equally relevant. Since our goal is to find the spectral decomposition of L_0, we must assume that U and Λ are unknown. But then how are we to compute the matrix-valued functions $g(L)$ or $e^{g(L)}$ efficiently? The choices $g(x) = \log x$ and $g(x) = x$ are special since these give $e^{g(L)} = L$ and $g(L) = L$, respectively. The first choice gives the QR algorithm (strictly speaking a branch of the logarithm must be chosen so that (4) is well-defined, but this does not affect the QR algorithm because of (7)). For the second choice $g(x) = x$, the vector field (4) is faster to compute than the matrix exponential $e^{L(m)}$ and it is natural to use an ordinary differential equation solver for (4) to diagonalize L. This is the essence of the Toda algorithm.

Our final choice $g(x) = \operatorname{sign}(x)$ requires further comment since the observation that the matrix sign algorithm is Hamiltonian seems to us to be new. Assume zero is not an eigenvalue of L_0 and let $\Sigma_\pm$ denote the eigenspaces of L_0 corresponding to positive and negative eigenvalues, respectively. Consider matrices $Q_\pm$ whose columns form an orthonormal basis for $\Sigma_\pm$, respectively. Then the matrices $P_+ = Q_+ Q_+^T$ and $P_- = Q_- Q_-^T$ are orthogonal projections onto $\Sigma_\pm$, respectively, and we find $\operatorname{sign}(L_0) = P_+ - P_-$ and $(I \pm \operatorname{sign}(L_0))/2 = P_\pm$. It is immediate that

$$e^{t \operatorname{sign}(L_0)} = e^t P_+ - e^{-t} P_-, \quad \text{and} \quad \lim_{t\to\infty} e^{-t} e^{t \operatorname{sign}(L_0)} = P_+. \tag{10}$$

The projection P_+ has a rank-revealing QR factorization $P_+ = U_\infty R_\infty \Pi$ [Higham 2008, Chapter 2.5]. The matrix sign algorithm rests on the fact that with U_∞ as above, $U_\infty^T U_\infty = I$, and the matrix

$$\tilde{L} = U_\infty^T L_0 U_\infty \tag{11}$$

is block-diagonal as in (1), where L_{11} is $k \times k$ with $k = \dim(\Sigma_+)$. Clearly, $\sigma(\tilde{L}) = \sigma(L_0)$.

Thus, the procedure to deflate a matrix using the matrix sign algorithm is:

(1) Given L_0, compute $\operatorname{sign}(L_0)$ and hence $P_+ = (I + \operatorname{sign}(L_0))/2$.

(2) Compute U_∞ using a rank-revealing QR decomposition of P_+.

(3) Compute $\tilde{L} = U_\infty^T L_0 U_\infty$.

We note that $\operatorname{sign}(L_0)$ can be computed efficiently using a scaled Newton iteration and inverse-free modifications of this procedure [Bai et al. 1997; Higham 2008; Malyshev 1993]. The complete spectral decomposition of L_0 may be determined in a sequence of deflation steps. At each stage, the number of iterations required to deflate the matrix depends on the number of iterations required to compute $\operatorname{sign}(L_0)$.

From the dynamical point of view, let $L(t)$ denote the solution to (4) with $g(L) = \operatorname{sign}(L)$. Then it may be shown that for generic initial data $\Pi = I$ and $\lim_{t\to\infty} L(t) = \tilde{L}$ where $\tilde{L} = U_\infty^T L_0 U_\infty$ is the block-diagonal matrix obtained above by the matrix sign algorithm. While this dynamical interpretation of the matrix sign algorithm is of theoretical interest, it is not clear how to implement the algorithm numerically in an effective manner.

We have not tested the performance of the matrix sign algorithm with random input in full generality. Instead, we have tested the deflation behavior of this algorithm in a more restricted setting by first precomputing $\operatorname{sign}(L_0)$ and then using Theorem 2. These results are not presented in this paper: the interested reader is referred to [Pfrang 2011].

2.3. ***Deflation criterion.*** Consider a symmetric matrix A with eigenvalues

$$\lambda_1 \geq \cdots \geq \lambda_n,$$

a symmetric matrix B, a positive number ϵ and the perturbed matrix $A + \epsilon B$ with eigenvalues

$$\lambda_1(\epsilon) \geq \cdots \geq \lambda_n(\epsilon).$$

Standard perturbation theory [Demmel 1997, Theorem 5.1] implies

$$|\lambda_i - \lambda_i(\epsilon)| \leq \epsilon \|B\|_2. \tag{12}$$

When deflating Jacobi matrices the perturbation matrix is of the form

$$B = \begin{pmatrix} 0 & E_{1k}^T \\ E_{1k} & 0 \end{pmatrix}, \tag{13}$$

where the only nonzero entry in $E_{1k} \in \mathbb{R}^{(n-k)\times k}$ is a one in the upper right corner. Clearly, $\|B\|_2 = 1$ in this case. For the deflation of full symmetric matrices, the perturbation matrix has the structure

$$B = \begin{pmatrix} 0 & B_{21}^T \\ B_{21} & 0 \end{pmatrix}, \tag{14}$$

where again $B_{21} \in \mathbb{R}^{(n-k)\times k}$, but now all entries of B satisfy $|b_{ij}| \leq 1$. In this case, one may show that $\|B\|_2 \leq \sqrt{k(n-k)}$.

We now define the deflation criterion. If L is a Jacobi matrix define

$$\hat{\epsilon}_k = b_k. \tag{15}$$

If $L = (l_{ij})$ is a full symmetric matrix, set

$$\hat{\epsilon}_k = \sqrt{k(n-k)} \max_{\substack{k<i\leq n \\ 1\leq j\leq k}} |l_{ij}|. \tag{16}$$

Assume L_m is a sequence of iterates (Jacobi or full symmetric) obtained through an iterative eigenvalue algorithm. For a given tolerance $\epsilon > 0$ and initial matrix L_0 we define the *deflation time*

$$\tau_{n,\epsilon}(L_0) = \min\bigl\{m \mid \hat{\epsilon}_k(L_m) < \epsilon \text{ for some } 1 \leq k \leq n-1\bigr\}. \tag{17}$$

For calculations based on the Hamiltonian flow (4) it is more natural to consider the real valued deflation time

$$\tau_{n,\epsilon}(L_0) = \inf\bigl\{t > 0 \mid \hat{\epsilon}_k(L(t)) < \epsilon \text{ for some } 1 \leq k \leq n-1\bigr\}. \tag{18}$$

The location where the matrix deflates is called the *deflation index*:

$$\iota_{n,\epsilon}(L_0) = \arg \min_{1 \le k \le n-1} \hat{\epsilon}_k(L_{\tau_{n,\epsilon}(L_0)}). \tag{19}$$

There is an important difference between deflation and the asymptotic convergence guaranteed by Theorem 1. While Theorem 1 may be used to compute asymptotic rates of convergence as $t \to \infty$ [Deift et al. 1983, Theorem 3], in practice the rate of convergence is determined by deflation and transients play an important role. We illustrate this with a simple example.

Fix $\lambda_1 > \lambda_2 > 0$, let $\Lambda = \operatorname{diag}(\lambda_1, \lambda_2)$ and consider the QR flow on Symm(2) with the initial matrix

$$L_0 = Q_0 \Lambda Q_0^T, \quad Q_0 = \begin{pmatrix} \cos\theta_0 & \sin\theta_0 \\ \sin\theta_0 & -\cos\theta_0 \end{pmatrix}. \tag{20}$$

According to Theorem 2, $\lim_{t\to\infty} L(t) = \Lambda$ for every θ_0. However, if $\theta_0 \approx \pi/2$, L_0 is a small perturbation of $\operatorname{diag}(\lambda_2, \lambda_1)$, and in practice, the algorithm would immediately deflate and return L_0. But according to Theorem 2, $L(t)$ must evolve so that the initial diagonal terms "turn around" and are presented in the correct order $\operatorname{diag}(\lambda_1, \lambda_2)$ as $t \to \infty$ (see (9)). More generally, consider $\Lambda = (\lambda_1, \ldots, \lambda_n)$ with $\lambda_1 > \lambda_2 > \cdots > \lambda_n > 0$. Each permutation $\sigma \in S_n$ yields a distinct fixed point $\Lambda_\sigma = (\lambda_{\sigma_1}, \ldots, \lambda_{\sigma_n})$ for the QR and Toda algorithms. In a numerical calculation, an initial condition close to Λ_σ is immediately deflated. Alternatively, iterates may pass close to one of the permutations Λ_σ and again deflation occurs at finite times. However, only the equilibrium $(\lambda_1, \ldots, \lambda_n)$ attracts generic initial conditions [Deift et al. 1983]. Thus the notion of convergence as $t \to \infty$ and deflation are completely distinct.

2.4. ***Ensembles.*** We now introduce the six ensembles of random matrices that we will analyze. For general introductions on random matrices see [Deift 1999; Edelman and Rao 2005; Mehta 2004]. The simplest way to construct an ensemble of random matrices is to choose entries independently subject only to the constraint of symmetry. Such ensembles are called *Wigner ensembles.* We also say that an ensemble lies in the *Wigner class* if the limiting spectral distribution for this ensemble is the Wigner semicircle law (described below). We consider four Wigner ensembles in the Wigner class:

1. the Gaussian orthogonal ensemble (GOE) (independent entries where we have $M_{ii} \sim \sqrt{2}\mathcal{N}(0,1),\ M_{ij} \sim \mathcal{N}(0,1),\ i > j$);
2. the Gaussian Wigner ensemble (GWE) (iid $M_{ij} \sim \mathcal{N}(0,1),\ i \ge j$);
3. the Bernoulli ensemble (iid $M_{ij} \sim \mathcal{B}$, $i \ge j$);

4. the Hermite-1 ensemble on Jacobi matrices (iid $a_k \sim \mathcal{N}(0,1),\ k = 1,\dots,n$ and independent $b_k \sim \chi_k,\ k = 0,\dots,n-1$).

Items 1–3 are ensembles of full symmetric matrices. The distinction between 1 and 2 is that the variance of the diagonal and off-diagonal entries of matrices in GOE is different to ensure orthogonal invariance (see [Mehta 2004]). Hermite-1 is an ensemble of Jacobi matrices obtained by applying the Householder tridiagonalization procedure to the GOE ensemble. It is a remarkable fact that the entries remain independent under tridiagonalization (this is not true when matrices from ensembles (2) and (3) are tridiagonalized).

A choice of an ensemble of random, symmetric matrices is a choice of a probability measure on the space of symmetric matrices. When the matrix entries are independent this measure is a product measure. For example, the measure corresponding to GOE has density

$$P_{\mathrm{GOE}}(M) = 2^{2n/2}(2\pi)^{-n(n+1)/4} e^{-\frac{1}{4}\operatorname{tr}(M^2)}. \tag{21}$$

For all these ensembles, while the matrix entries are independent, the eigenvalues are not. The joint density of eigenvalues for GOE and Hermite-1 may be computed explicitly and is given by the determinantal formula [Mehta 2004, Chapter 3]

$$f_1(\Lambda) = \frac{1}{Z_n} |\triangle_n(\lambda)| e^{-\frac{|\lambda|^2}{2}}, \quad \triangle_n(\lambda) = \prod_{i<j} (\lambda_i - \lambda_j). \tag{22}$$

The normalization constant Z_n may be computed explicitly. By contrast, while the analogues of (21) for ensembles 2 and 3 are clear, there is no explicit analogue for (22).

The ensembles 1–4 are in the Wigner class, i.e., for each of these ensembles

$$\lim_{n\to\infty} \frac{1}{n} \#\{\lambda_i \in \sqrt{n}(a,b)\} = \int_a^b \nu(x)\,dx, \tag{23}$$

where $\nu(x)$ denotes the density of the *Wigner semicircle law*

$$\nu(x) = \frac{1}{2\pi}\sqrt{4-x^2}\ \mathbb{1}_{|x|\le 2}. \tag{24}$$

We will contrast our results on these ensembles with two ensembles of Jacobi matrices that are not in the Wigner class. These are:

5. the uniform doubly stochastic Jacobi ensemble (UDSJ);
6. the Jacobi uniform ensemble (JUE).

Doubly stochastic Jacobi matrices of dimension $n \times n$ form a compact polytope in $\mathbb{R}^{n-1}$ which can be equipped with its uniform measure [Diaconis and Wood

2010]. This is the UDSJ ensemble. We can approximately sample from this ensemble using a Gibbs sampler.

JUE is defined using the spectral map $\mathcal{S}$ for Jac(n). Since we may describe Jacobi matrices by their spectral data (Λ, u), a probability measure on the spectral data pulls back under $\mathcal{S}^{-1}$ to a probability measure on Jac(n). For JUE, we replace (22) with eigenvalues chosen independently and uniformly on an interval and u distributed uniformly on the orthant S_+^{n-1}. In our numerical simulations we assume the eigenvalues are uniformly distributed on $[-2\sqrt{n}, 2\sqrt{n}]$ because this interval corresponds to the support of the semicircle law and allows a comparison between JUE and ensembles in the Wigner class. A particularly important aspect of JUE is that the eigenvalues do not repel one another. This strongly affects the statistics of $\tau_{n,\epsilon}$ as shown below (for unshifted QR and Toda, but not for shifted QR!).

2.5. ***The normalized deflation time.*** We have now defined the algorithms, ensembles and deflation criterion. For a given algorithm and ensemble, $\tau_{n,\epsilon}(L)$ and $\iota_{n,\epsilon}(L)$ are random variables that depends on the random initial matrix L and $\epsilon > 0$. We explore the empirical distributions of $\tau_{n,\epsilon}$ and $\iota_{n,\epsilon}$ in simulations. Our main empirical finding is that for each algorithm these empirical distributions collapse into a universal distribution for the Wigner ensembles 1–4. Let $\mu_{n,\epsilon}$ and $\sigma_{n,\epsilon}^2$ denote the empirically determined mean and variance of $\tau_{n,\epsilon}(L)$ for a particular algorithm and ensemble.

Our simulations suggest that the normalized deflation time

$$T_{n,\epsilon} = \frac{\tau_{n,\epsilon} - \mu_{n,\epsilon}}{\sigma_{n,\epsilon}} \tag{25}$$

converges in distribution as $n \to \infty$ and $\epsilon \to 0$ and that the limit is the same for ensembles in the Wigner class (see Figures 4 and 10). Both $\mu_{n,\epsilon}$ and $\sigma_{n,\epsilon}$ are computed empirically. Our numerical calculations also suggest that $\mu_{n,\epsilon} \sim C\,|\log\epsilon|$ for all ensembles in the Wigner class (see Figures 13 and 15). As already noted above, a surprising outcome of our simulations is that universality for shifted QR is more encompassing, and actually holds for all six ensembles 1–6.

In order to prove convergence in distribution of $T_{n,\epsilon}$ it is first necessary to estimate the mean and variance of τ. We present below a calculation of $\mu_{2,\epsilon}$ that illustrates the subtle role of eigenvalue repulsion.

2.6. ***The scaling of the expected deflation time.*** In this section we estimate the expected deflation time of the Toda flow on Symm(2). We show that

$$\mu_{2,\epsilon,\mathrm{GOE}} \sim C\,|\log\epsilon|, \quad \text{but } \mu_{2,\epsilon,\mathrm{JUE}} \sim C\,|\log\epsilon|^2, \quad \epsilon \to 0. \tag{26}$$

The interval of support for the JUE density is chosen here to be $[-1, 1]$. This choice only affects the prefactor C, not the term $|\log \epsilon|^2$.

In order to establish these asymptotics, we first determine the deflation time τ_ϵ as a function of the initial condition (for brevity we write τ_ϵ for $\tau_{2,\epsilon}$ since $n = 2$ is fixed). Since $M(t) \in \mathrm{Symm}(2)$ we may write $M = U(t)\Lambda U(t)^T$, where $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2)$, $\lambda_1 > \lambda_2$, and

$$U(t) = \begin{pmatrix} \cos\theta(t) & \sin\theta(t) \\ \sin\theta(t) & -\cos\theta(t) \end{pmatrix}. \tag{27}$$

Note that $m_{12} > 0$ corresponds to $\theta \in (0, \pi/2)$. We use Theorem 2 to obtain

$$m_{12}(t) = (\lambda_1 - \lambda_2)\cos\theta(t)\sin\theta(t) = (\lambda_1 - \lambda_2) \cdot \frac{e^{t(\lambda_2 - \lambda_1)} \cdot \tan\theta_0}{1 + e^{2t(\lambda_2 - \lambda_1)} \tan^2\theta_0}. \tag{28}$$

Here $\theta_0 = \theta(0)$. Now we set $m_{12}(\tau_\epsilon) = \epsilon$ and solve to find

$$(\lambda_1 - \lambda_2)\tau_\epsilon = \begin{cases} 0 & m_{12}(0) \le \epsilon, \\ \log\tan\theta_0 - \log\left[\dfrac{\lambda_1 - \lambda_2}{2\epsilon} - \sqrt{\dfrac{(\lambda_1 - \lambda_2)^2}{4\epsilon^2} - 1}\right] & m_{12}(0) > \epsilon. \end{cases} \tag{29}$$

The asymptotics of τ_ϵ are easily determined. We have

$$(\lambda_1 - \lambda_2)\tau_\epsilon \sim -\log\epsilon + \log\tan\theta_0 + \log(\lambda_1 - \lambda_2), \quad \epsilon \to 0. \tag{30}$$

In order to compute the mean deflation time for GOE and JUE we first change to spectral variables. As noted above, the spectral map $\mathcal{S}$ is a diffeomorphism between the set of 2×2 symmetric matrices with $m_{12} > 0$ and the set $\{\lambda_1 > \lambda_2\} \times (0, \pi/2)$. The Jacobian of this transformation is $\lambda_1 - \lambda_2$, so that

$$dm_{11}\, dm_{22}\, dm_{12} = (\lambda_1 - \lambda_2)\, d\lambda_1\, d\lambda_2\, d\theta. \tag{31}$$

The mean deflation time for GOE is then given by

$$\mu_{2,\epsilon,\mathrm{GOE}} = \frac{1}{Z_1} \int_{-\infty}^{\infty} \int_{-\infty}^{\lambda_1} \int_0^{\pi/2} \tau_\epsilon(\lambda_1, \lambda_2, \theta) e^{-(\lambda_1^2 + \lambda_2^2)/4} (\lambda_1 - \lambda_2)\, d\lambda_1\, d\lambda_2\, d\theta. \tag{32}$$

For JUE, the eigenvalues are chosen uniformly from $[-1, 1]$ and we find

$$\mu_{2,\epsilon,\mathrm{JUE}} = \frac{1}{Z_2} \int_{-1}^{1} \int_{-1}^{\lambda_1} \int_0^{\pi/2} \tau_\epsilon(\lambda_1, \lambda_2, \theta)\, d\lambda_2\, d\lambda_2\, d\theta. \tag{33}$$

Here Z_1 and Z_2 are normalizing constants for these probability densities.

The asymptotic behavior of (30), combined with (32) and (33), suggests the following leading order behavior as $\epsilon \to 0$:

$$\mu_{2,\epsilon,\mathrm{GOE}} \sim \frac{|\log\epsilon|}{Z_1} \int_{-\infty}^{\infty}\int_{-\infty}^{\lambda_1}\int_{0}^{\pi/2} e^{-(\lambda_1^2+\lambda_2^2)/4}\, d\lambda_1\, d\lambda_2\, d\theta \sim C_1\,|\log\epsilon|,$$

$$\mu_{2,\epsilon,\mathrm{JUE}} \sim \frac{|\log\epsilon|}{Z_2} \int_{-1}^{1}\int_{-1}^{\lambda_1}\int_{0}^{\pi/2} \frac{1}{\lambda_1-\lambda_2}\mathbb{1}_{m_{12}>\epsilon}\, d\lambda_1\, d\lambda_2\, d\theta \sim C_2\,|\log\epsilon|^2.$$

Here C_i denote constants that may be computed explicitly. The second integral is divergent without the cut-off $\mathbb{1}_{m_{12}>\epsilon}$: the cut-off gives rise to an additional factor of $|\log\epsilon|$. With more effort, these formal estimates may be made rigorous.

The analogous calculations for $M(t) \in \mathrm{Jac}(n)$, $n > 2$ are quite subtle. For Jacobi matrices deflation occurs when $M(t)$ approaches the boundary $\partial\,\mathrm{Jac}(n)$ of $\mathrm{Jac}(n)$ (see for example [Deift et al. 1983, Figs. 6 and 7]). A theoretical analysis of such deflations, which we have not carried out yet, is a significant challenge as it requires a detailed understanding of the geometry of both the flow and the initial probability distribution in the vicinity of $\partial\,\mathrm{Jac}(n)$ in high dimensions. For this reason, we are reduced to using the empirical mean $\mu_{n,\epsilon}$ and variance $\sigma^2_{n,\epsilon}$ to define the normalized deflation time in (25).

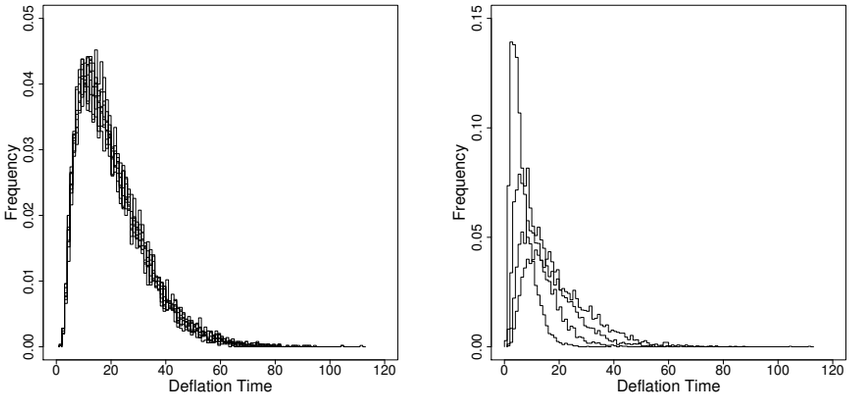
3. Results

We generated a large number (typically 5000–10,000) of samples of the deflation time and the deflation index for each choice of the following parameters:

1. an eigenvalue algorithm (QR without shift, QR with shift, Toda);
2. a random matrix ensemble;
3. matrix size n (typically ranging from $10, 30, \ldots, 190$);
4. tolerance ϵ (typically 10^{-k}, $k = 2, 4, 6, 8$).

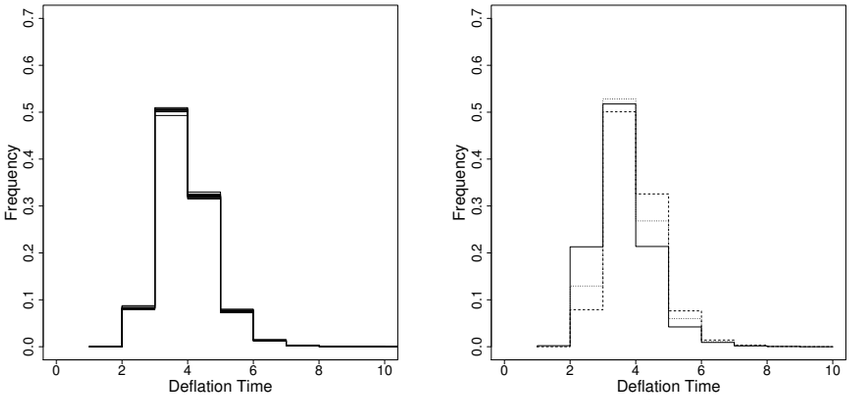
We present a representative sample of our main results. Further statistical tests, figures and tables that amplify our conclusions may be found in [Pfrang 2011].

3.1. ***Unscaled deflation time statistics for GOE.*** We first present deflation time statistics for $\tau_{n,\epsilon}$ for a fixed ensemble (GOE) for both the QR (shifted and unshifted) and Toda algorithms. The statistics of $\tau_{n,\epsilon}$ for the unshifted QR algorithm are shown in Figure 1. Similar statistics for the QR algorithm with Wilkinson shift and the Toda algorithm are shown in Figures 2 and 3, respectively. These figures reflect the typical dependence of these algorithms on n and ϵ for ensembles 1–6. Similar statistics for other ensembles may be found in [Pfrang 2011, Chapter 7]. In all cases, we observe that the histograms for the QR algorithm are relatively insensitive to n and shift to the right as ϵ decreases. The



(a) (b)

Figure 1. QR algorithm applied to GOE. (a) Histogram for empirical frequency $\tau_{n,\epsilon}$ as n ranges from $10, 30, \ldots, 190$ for a fixed deflation tolerance $\epsilon = 10^{-8}$. The curves (10 of them, plotted one on top of another) do not depend significantly on n. (b) Histogram for empirical frequency of $\tau_{n,\epsilon}$ when $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ for fixed matrix size $n = 190$. Curves move to the right as ϵ decreases.



(a) (b)

Figure 2. Shifted QR algorithm applied to GOE. (a) Histogram for empirical frequency $\tau_{n,\epsilon}$ as n ranges from $10, 30, \ldots, 190$ for a fixed deflation tolerance $\epsilon = 10^{-12}$. In the case of the unshifted QR algorithm, curves are insensitive to n, though the tail becomes more pronounced for larger n. (b) Histogram for empirical frequency $\tau_{n,\epsilon}$ when $\epsilon = 10^{-k}$, $k = 8, 10, 12$ for fixed matrix size $n = 190$. Curves move to the right as ϵ decreases.

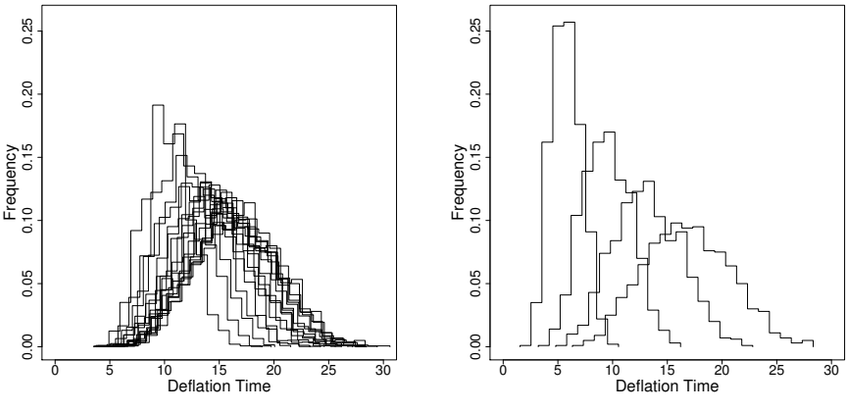


Figure 3. Toda algorithm applied to GOE. (a) Histogram for empirical frequency of $\tau_{n,\epsilon}$ as n ranges from $10, 30, \ldots, 190$ for a fixed deflation tolerance $\epsilon = 10^{-8}$. Curves drift to the right as n increases. (b) Histogram for empirical frequency $\tau_{n,\epsilon}$ when $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ for fixed matrix size $n = 190$. Curves move to the right as ϵ decreases.

effect of the Wilkinson shift is to sharply reduce the number of iterations required (note the different scale of the abscissa in Figures 1 and 2). The values of ϵ for shifted QR are much smaller than those chosen for QR without shifts. This choice is necessary to generate a viable data set for the shifted QR algorithm with sufficient variation in the deflation time. The histograms for the Toda algorithm shift to the right as n increases and ϵ decreases, as discussed below.

3.2. ***Normalized deflation time and universality for the Wigner class.*** We now present results that show the collapse of all data onto universal curves depending only on the algorithm under the rescaling (25). The statistics of the empirical mean $\mu_{n,\epsilon}$ and standard deviation $\sigma_{n,\epsilon}$ are discussed a little later. The empirical distribution of the normalized deflation time $T_{n,\epsilon}$ for the QR algorithm with initial data from the Wigner ensembles is shown in Figure 4. All the data contained in Figure 1 collapse onto the single curve seen in Figure 4(a). Analogous data for the other Wigner class ensembles 2–4 collapse onto the *same universal curve*. The normalized deflation time distributions for UDSJ and JUE are shown in Figure 5. While we again observe a collapse of the data, it is not onto the curve of Figure 4(a). This contrast is amplified in the comparison of the tails of the normalized deflation time (see Figure 6). QQ plots that directly compare the histograms of these distributions may be found in [Pfrang 2011].

The most obvious difference between the behavior of the unshifted and shifted QR algorithm is that the spread in the deflation time for the shifted QR algorithm is

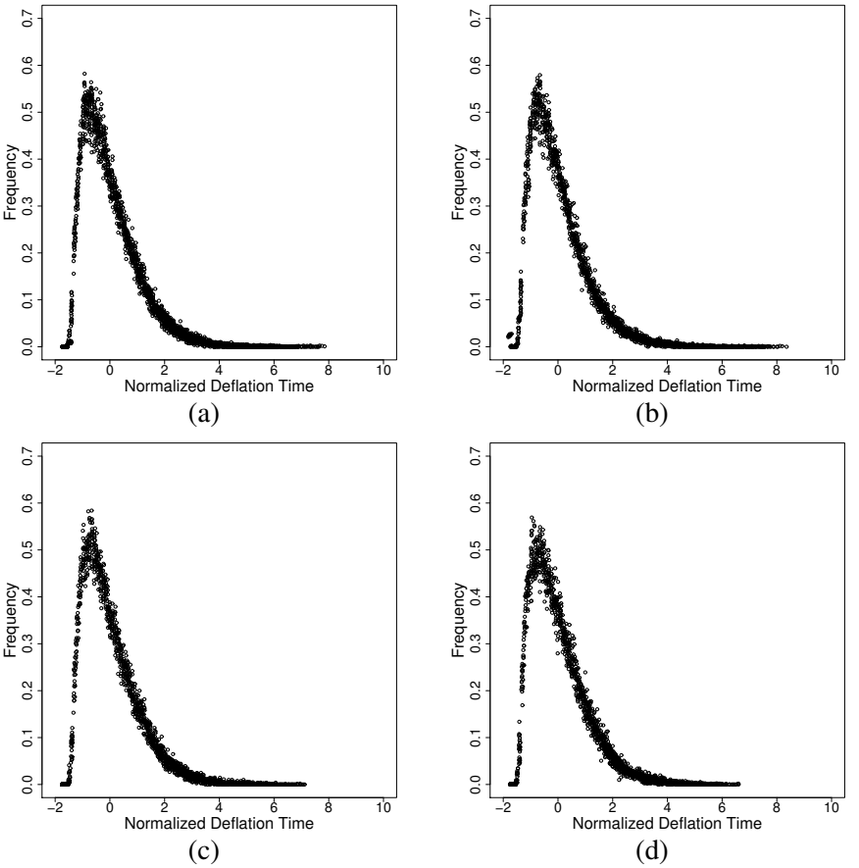
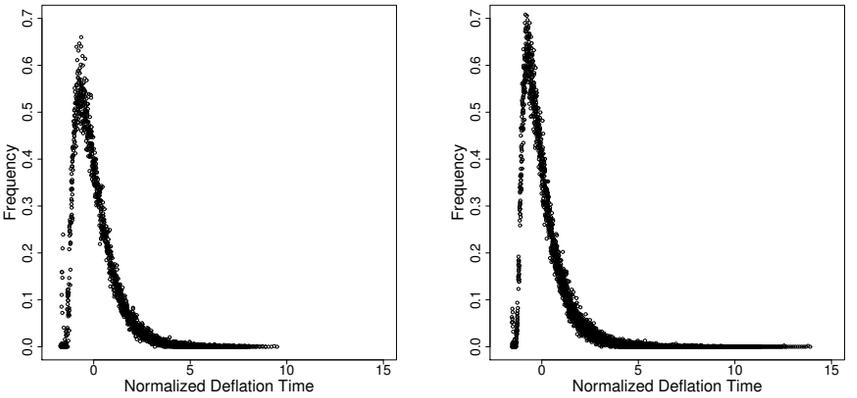


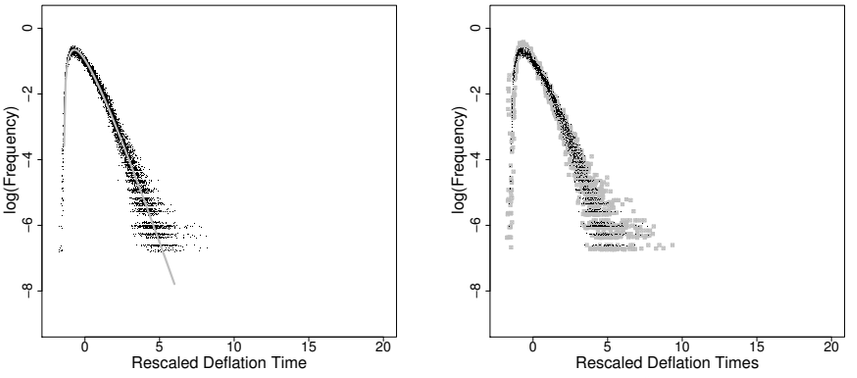
Figure 4. Universal deflation time statistics for QR algorithm applied to Wigner class. Empirical deflation time normalized as in (25) for $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ and n ranging from $10, 30, \ldots, 190$. Random matrix ensembles are (a) GOE; (b) Hermite-1; (c) GWE; and (d) Bernoulli, with (a)–(d) obtained by rescaling data of 10×4 fixed-n and fixed-ϵ histograms and plotting them together. All these data collapse onto one universal curve. Plotting all 160 histograms together in Figure 6 further demonstrates universality of the deflation algorithm.

much narrower. However, this does not seem to affect our general conclusion that there is universality for each Hamiltonian eigenvalue algorithm. The normalized deflation time distribution for shifted QR is shown in Figures 7 and 8. Moreover, for shifted QR, the deflation times vary far less with the choice of underlying ensemble than the unshifted QR algorithm. In particular, we see a strong similarity for all ensembles in Figure 9. This behavior is in contrast with that of unshifted QR, shown in Figure 6.



(a) (b)

Figure 5. QR algorithm applied to non-Wigner ensembles. Normalized empirical deflation time distributions for QR algorithm with $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ and n ranging from $10, 30, \ldots, 190$. Random matrix ensembles are (a) UDSJ and (b) JUE. Each figure contains normalized empirical data of 40 fixed-n and fixed-ϵ histograms. All data are observed to collapse onto a single curve. However, these curves are not the same for UDSJ and JUE, and neither of these coincides with curve for Wigner data shown in Figure 4.



(a) (b)

Figure 6. Exponential tail for QR algorithm. Histograms of normalized deflation time for QR algorithms on a logarithmic scale. (a) Wigner data: Empirical normalized deflation time distributions from all 160 histograms of Wigner class initial data (black dots) are compared with a gamma distribution with parameters $k = 2$ and $\theta = 1$ shifted to mean zero (gray line). (b) non-Wigner data: Empirical normalized deflation time distributions from 40 GOE histograms (black dots) contrasted with data from 40 UDSJ histograms (gray squares).

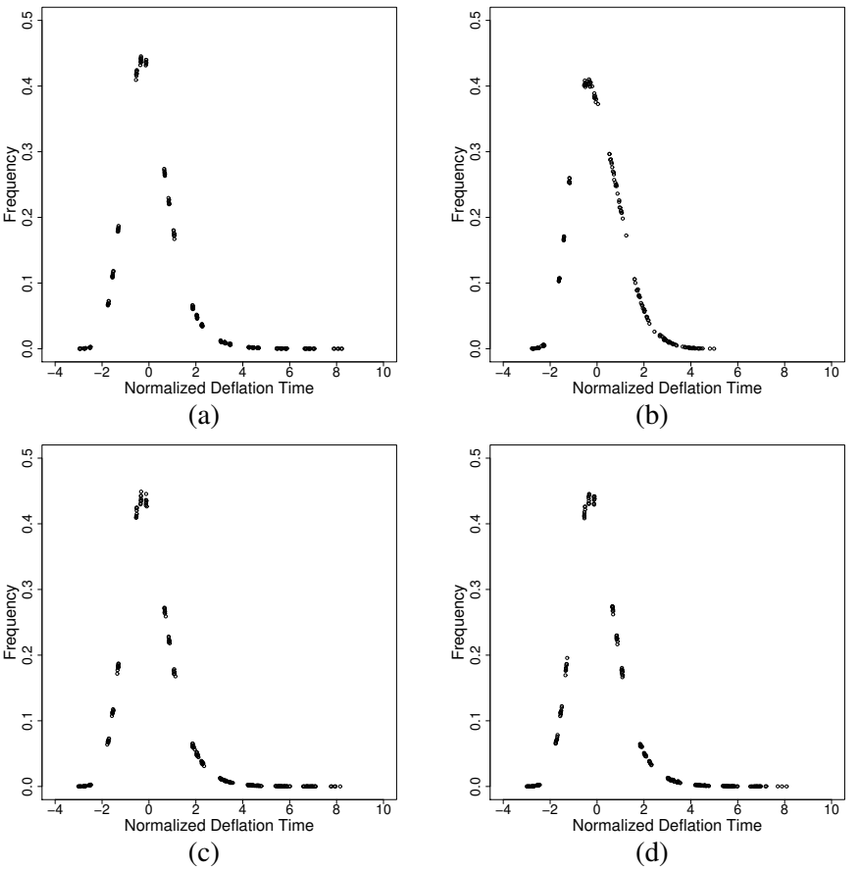
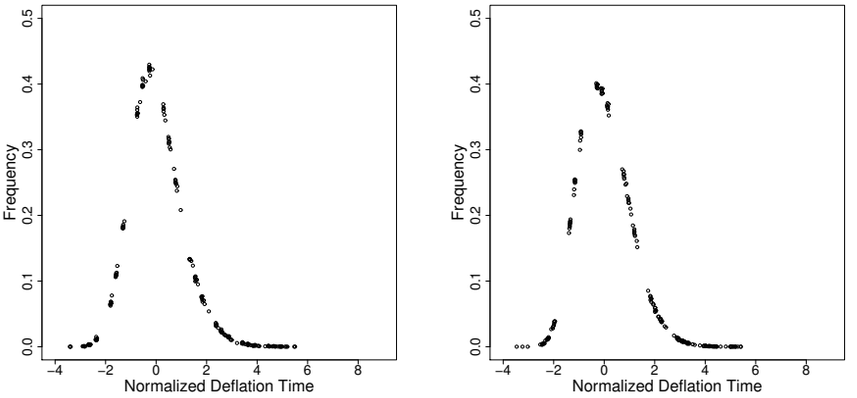


Figure 7. Universality for shifted QR algorithm on Wigner class. Empirical deflation time for QR algorithm with Wilkinson shift normalized as in (25) with $\epsilon = 10^{-k}$, $k = 8, 10, 12$ and n ranging from $10, 30, \ldots, 190$. Note that ϵ is significantly smaller than for unshifted QR algorithm. Ensembles are (a) GOE; (b) Hermite-1; (c) GWE; and (d) Bernoulli. Figures (a)–(d) are obtained by collapsing data as in Figure 4. Peak of the TE1 ensemble is lower, and tail shorter, than those for other three ensembles.

Finally, we have also observed universality for the Toda algorithm. The empirical distribution of the normalized deflation time for the Wigner ensembles is shown in Figure 10. Again, all the data contained in Figure 3 collapse onto the single curve seen in Figure 10(a). Further, analogous data for the other Wigner ensembles 2–4 collapse onto the same curve. The data for UDSJ and JUE collapse under normalization, but not onto the same distribution (see Figures 5 and 12).



(a) (b)

Figure 8. Shifted QR algorithm applied to non-Wigner ensembles. Normalized empirical deflation time distributions for QR algorithm with Wilkinson shift for $\epsilon = 10^{-k}$, $k = 8, 10, 12$ and n ranging from $10, 30, \ldots, 190$. Random matrix ensembles are (a) UDSJ and (b) JUE. Note that results for these ensembles seem very similar to those for Wigner class data shown in Figure 7. UDSJ is similar to full matrix ensembles, while JUE is similar to TE1, also a tridiagonal ensemble.

Remark 1. We note that for both the QR and Toda algorithms the limiting distribution of the normalized deflation time $T_{n,\epsilon}$ for UDSJ and JUE is distinct from that of ensembles in the Wigner class. This raises the interesting issue in random matrix theory whether UDSJ and JUE are in the same universality class as Wigner ensembles and invariant ensembles. As JUE does not have eigenvalue repulsion built in, this is unlikely to be the case.

3.3. ***Universal tails for deflation times.*** We used a hypothesis testing approach to quantify the statement that the rescaled deflation time has an exponential tail for QR and a Gaussian tail for Toda. Our approach is modeled on the methodology of [Clauset et al. 2009]. Given deflation time data D we perform maximum likelihood estimation of parameters for distribution families conditioned on observing only values above a cutoff value $x_{\min}(D)$ and use a semiparametric approach to compute p-values for these parameters. Based on D and our parameter estimate, we compute resampled data sets and a modified Kolmogorov–Smirnov statistic measuring the distance between the empirical distribution function and the ones resulting from our maximum likelihood estimates. The semiparametric p-value is given as the proportion of instances that the resampled data sets yield larger modified KS statistics than the original. If this p-value is large we accept the hypothesis that the original data set has in fact the proposed decay in the right tail.

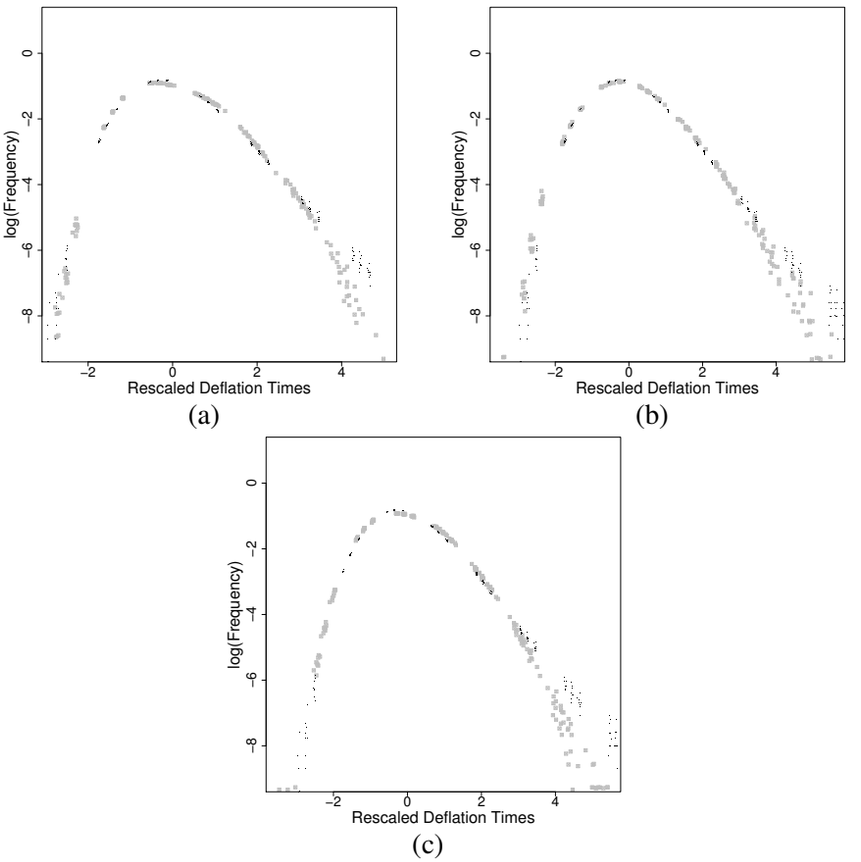


Figure 9. Comparison of ensembles for shifted QR. Histograms of normalized deflation time for shifted QR algorithms on a logarithmic scale. In (a)–(c), GOE (black dots) is contrasted with data from a second ensemble (gray dots). (a) GOE and TE1: Empirical normalized deflation time distributions from 40 GOE histograms (black dots) contrasted with data from 40 TE1 histograms (gray squares); (b) GOE and UDSJ; (c) GOE and JUE.

We applied this approach with the Gaussian, Exponential, Weibull and Gamma families. We found that the exponential tails fit the QR runtime data especially well for small values of the deflation tolerance. The fit of the Toda runtime data to Gaussian tails is very compelling across most experimental regimes. Direct pictorial comparisons of the normalized Toda runtimes with the standard normal as well as normalized QR runtimes with normalized Gamma distributions are shown in Figure 6. Further details of the statistical tests may be found in [Pfrang 2011].

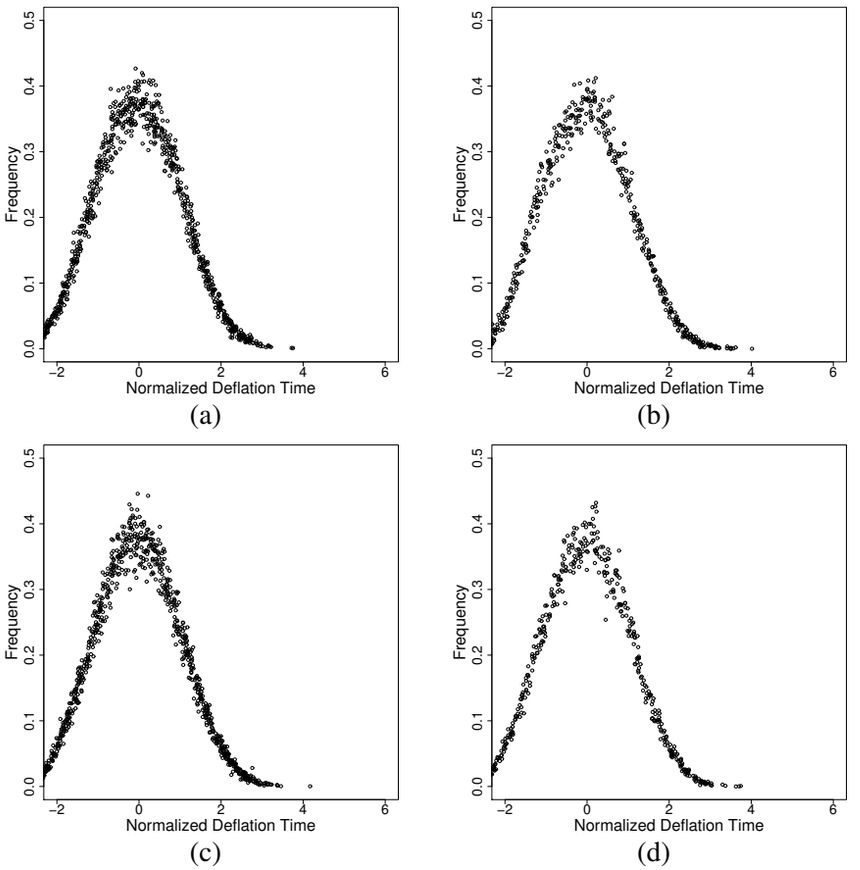


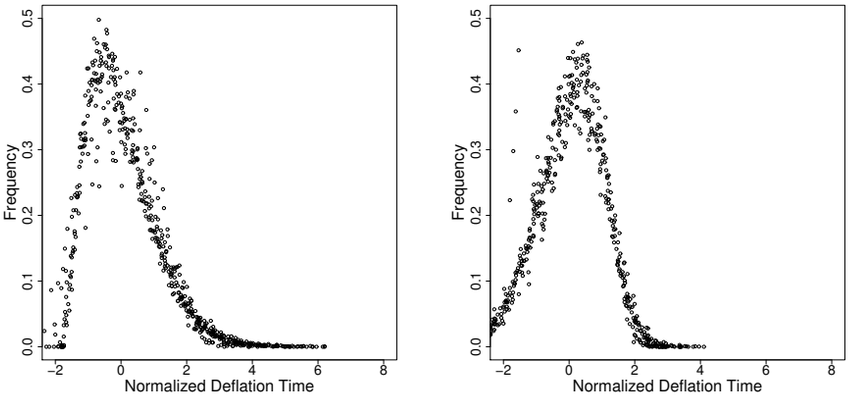
Figure 10. Universal deflation time statistics for Toda algorithm applied to Wigner class. Empirical deflation time normalized as in (25) for $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ and n ranging from $10, 30, \ldots, 190$. Random matrix ensembles are (a) GOE; (b) Hermite-1; (c) GWE; and (d) Bernoulli, with (a)–(d) obtained by rescaling data of 40 fixed-n and fixed-ϵ histograms and plotting them together. All these data collapse onto one universal curve. Universality is amplified in Figure 12.

3.4. ***The dependence of $\mu_{n,\epsilon}$ and $\sigma_{n,\epsilon}$ on n and ϵ.*** We used linear regression to express $\mu_{n,\epsilon}$ and $\sigma_{n,\epsilon}$ as functions of $\log \epsilon$ and n. Only the best fits are reported here. The data for the QR algorithm was matched very well by

$$\mu_{n,\epsilon} \approx a_0 + a_1 n + a_2 \log \epsilon, \tag{34}$$

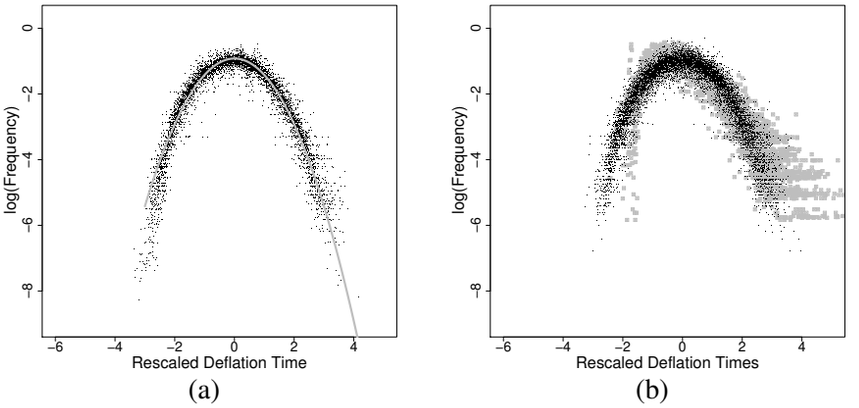
$$\sigma_{n,\epsilon} \approx b_0 + b_1 n + b_2 \log \epsilon. \tag{35}$$

This regression is compared visually with the numerical data in Figures 13 and 14.



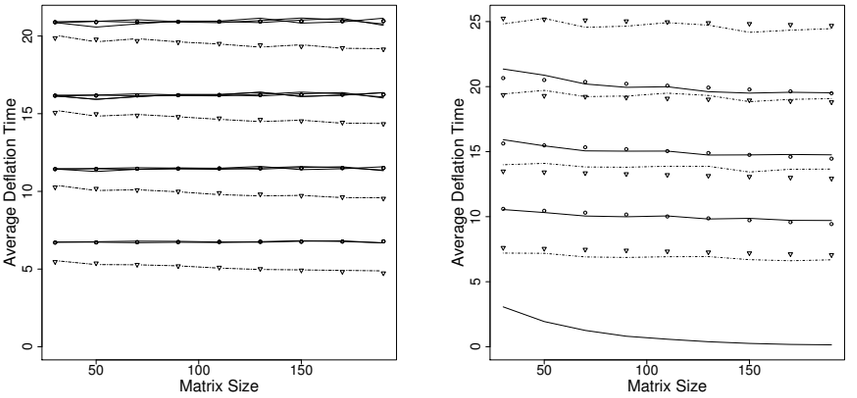
(a) (b)

Figure 11. Toda algorithm applied to non-Wigner ensembles. Normalized empirical deflation time distributions for Toda algorithm with $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ and n ranging from 10, 30, ..., 190. Random matrix ensembles are (a) UDSJ and (b) JUE; each contains normalized empirical data of 40 fixed-n and fixed-ϵ histograms. All data are observed to collapse onto a single curve. However, curves are not the same for UDSJ and JUE and neither of these coincides with Wigner data curve shown in Figure 10.



(a) (b)

Figure 12. Gaussian tail for Toda algorithm. Histograms of normalized deflation time for QR algorithms on a logarithmic scale. (a) Wigner data: Empirical normalized deflation time distributions from all 160 histograms of Wigner class initial data (black dots) compared with standard normal distribution (gray line). (b) non-Wigner data: Empirical normalized deflation time distributions from 40 GOE histograms (black dots) contrasted with data from 40 UDSJ histograms (gray squares).



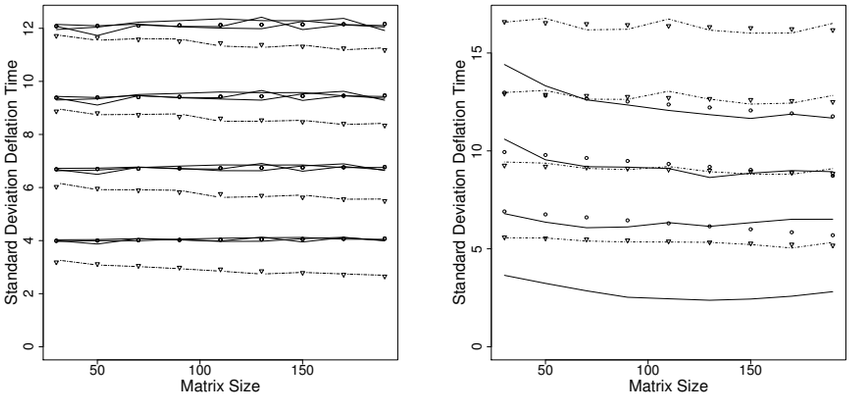
(a) (b)

Figure 13. Mean deflation time $\mu_{n,\epsilon}$ for QR algorithm. Empirical average of deflation time for $\epsilon = 10^{-k}$, $k = 2, 4, 6, 8$ and n in the range $10, 30, \ldots, 190$. (a) Wigner class initial data: Full lines are empirical mean $\mu_{n,\epsilon}$ for GOE, GWE and Bernoulli ensembles. Note that they seem to align well with one another. Circles are values obtained from the regression estimate (34) with parameters listed in Table 1. Dashed line and triangles represent empirical data and regression, respectively, for Hermite-1 ensemble. (b) JUE and UDSJ initial data: Full line and dashed line are empirical mean $\mu_{n,\epsilon}$ for UDSJ and JUE data, respectively. Circles and triangles are regression estimates for UDSJ and JUE, respectively. As ϵ decreases, curves move up monotonically. Regression is not applied to lowest curve in (b) since $\epsilon = 0.01$ is sufficiently large that several matrices deflate instantaneously.

The regression parameters are tabulated in Tables 1 and 2. Since the means and variances do not visually appear to depend on n for ensembles 1–3 we have also included the p-values for the t-test of the hypothesis that the coefficient corresponding to the dimension is zero. Note that $\mu_{n,\epsilon}$ and $\sigma_{n,\epsilon}$ are almost identical for the ensembles 1–3 in the Wigner class, while for the Hermite-1 initial data both statistics have a slightly larger value.

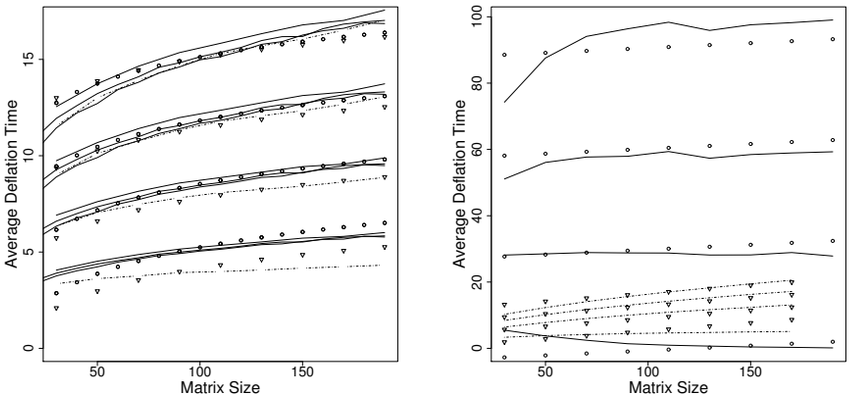
Ensemble	a_0	a_1	a_2	p-value a_1
GOE, GWE, Bernoulli	1.96824	0.0004690	-1.0263649	0.0095
Hermite-1	.802338	$-.004554$	-1.042907	$< 2 \cdot 10^{-16}$
UDSJ	0.7648330	-0.0072921	-1.0916354	$4.16 \cdot 10^{-8}$
JUE	1.844126	-0.003467	-1.276037	0.0256

Table 1. Regression parameters for $\mu_{n,\epsilon}$ for the unshifted QR algorithm.



(a) (b)

Figure 14. Standard deviation $\sigma_{n,\epsilon}$ of deflation time for QR algorithm. (a) Ensembles in Wigner class; (b) JUE and UDSJ. Legend as in Figure 13 with regression parameters from Table 2.



(a) (b)

Figure 15. Mean deflation time $\mu_{n,\epsilon}$ for Toda algorithm. Mean deflation time distributions of Toda algorithm for initial data described in Figure 13. (a) Wigner class initial data; (b) JUE and UDSJ initial data. Legend as in Figure 13 and regression parameters as in Table 3.

Ensemble	b_0	b_1	b_2	p-value b_1
GOE, GWE, Bernoulli	1.2799509	0.0005311	−0.5854859	0.0118
Hermite-1	0.442622	−.003329	−.617517	$8 \cdot 10^{-15}$
UDSJ	1.066713	−0.007584	−0.658920	0.000353
JUE	2.0044243	−0.0026034	−0.7961700	0.000185

Table 2. Regression parameters for $\sigma_{n,\epsilon}$ for the unshifted QR algorithm.

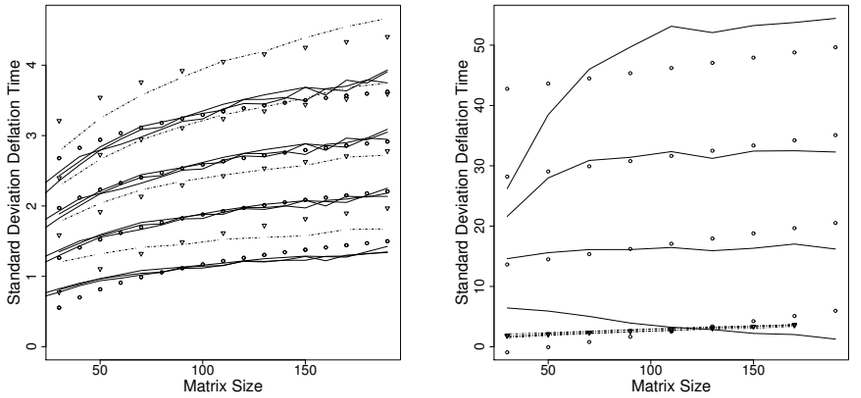


Figure 16. Standard deviation $\sigma_{n,\epsilon}$ of deflation time for Toda algorithm. (a) Ensembles in the Wigner class; (b) JUE and UDSJ. Legend as in Figure 13 and regression parameters as in Table 4.

Ensemble	a_0	a_1	a_2
GOE, GWE, Bernoulli	-6.0669	1.2888	-0.7302
Hermite-1	-7.0273	1.6795	-0.7708
UDSJ	-34.01514	0.02984	-6.60133
JUE	-2.78614	0.05318	-0.74315

Table 3. Regression parameters for $\mu_{n,\epsilon}$ for the Toda algorithm. UDSJ and JUE are fit to (34)–(35) and the Wigner class ensembles are fit to (36)–(37).

The deflation time depends more strongly on n for the Toda algorithm. We explored several regressions but our results for Toda are more ambiguous than for QR. We found that the non-Wigner ensembles (UDSJ and JUE) could be fit with an expression of the form (34)–(35). However, the Wigner class ensembles were better suited to the regression

$$\mu_{n,\epsilon} \approx a_0 + a_1 \log n + a_2 \log \epsilon \tag{36}$$

$$\sigma_{n,\epsilon} \approx b_0 + b_1 \log n + b_2 \log \epsilon \tag{37}$$

The results of this regression are presented in Figures 15–16 and Tables 3–4.

3.5. ***Deflation index statistics and the effect of the Wilkinson shift.*** The remarkable acceleration of QR by shifting is of course well known. Our experiments provide a quantitative statistical picture for the efficacy of the shift. Figure 17

Ensemble	b_0	b_1	b_2
GOE, Gaussian Wigner, Bernoulli	−1.6532	0.3347	−0.1569
Hermite-1	−2.1233	0.6324	−0.1727
UDSJ	−16.46367	0.04845	−3.10561
JUE	0.97525	0.04068	0.01451

Table 4. Regression parameters for $\sigma_{n,\epsilon}$ for the Toda algorithm. UDSJ and JUE are fit to (34)–(35) and the Wigner class ensembles are fit to (36)–(37).

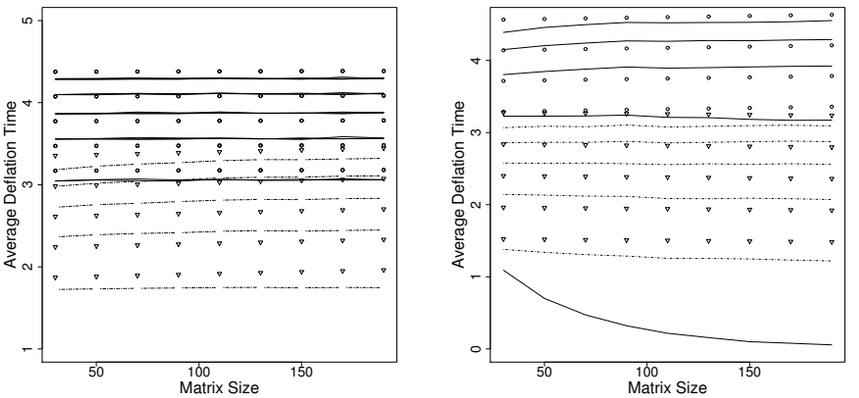


Figure 17. Effect of the Wilkinson shift. Mean deflation time $\mu_{n,\epsilon}$ for QR algorithm with Wilkinson shift. (a) Wigner class ensembles; (b) JUE and UDSJ. Empirical data are generated for $\epsilon = 10^{-2}, \ldots, 10^{-8}$ and $n = 20, \ldots, 190$. Empirical data and a regression of the form (34) are presented in same line-styles as in Figure 13. Observe that $\mu_{n,\epsilon}$ is almost independent of n and that curves move upwards as ϵ decreases, as in Figure 13, but the scale of the ordinate is different. Regression is not applied to lowest curve in (b) since $\epsilon = 0.01$ is sufficiently large that several matrices deflate instantaneously.

shows that the deflation time is sharply reduced by the Wilkinson shift. Figure 18 shows that the standard deviation of the deflation time is also sharply reduced by the shift. Deflation takes only a few iterations independent of the size of the matrix. This is in sharp contrast with the unshifted QR algorithm.

An explanation for the speed-up lies in the statistics of the deflation index shown in Figure 19. We find that the unshifted QR algorithm deflates at the bottom right corner of the matrix with high probability. Since the Wilkinson shift uses only the 2×2 lower-right block of the matrix, small off-diagonal terms

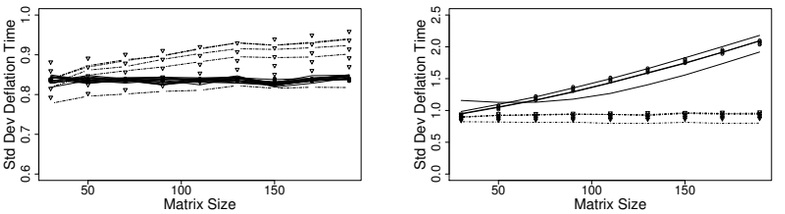


Figure 18. Standard deviation of deflation time with Wilkinson shift. (a) Wigner class ensembles; (b) JUE and UDSJ. Line styles are as in Figure 17 with a regression of the form (35).

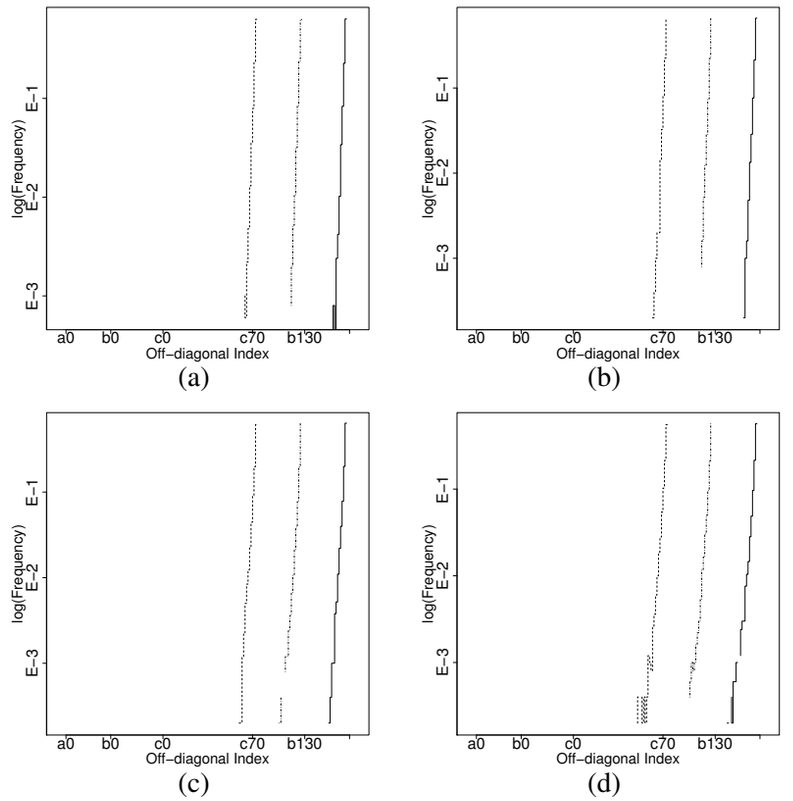


Figure 19. Empirical distributions of deflation index $\iota_{n,\epsilon}$ for unshifted QR algorithm. Figures show histograms of frequency with which deflation occurs at a given off-diagonal index. To aid visibility, distribution is centered so that peaks do not overlap. Off-diagonal index takes values between 0 and $n-2$. Here "a", "b" and "c" refer to ensembles with $n = 190$, 130 and 70, respectively. Ensembles shown are (a) Hermite-1; (b) GOE; (c) UDSJ; and (d) JUE.

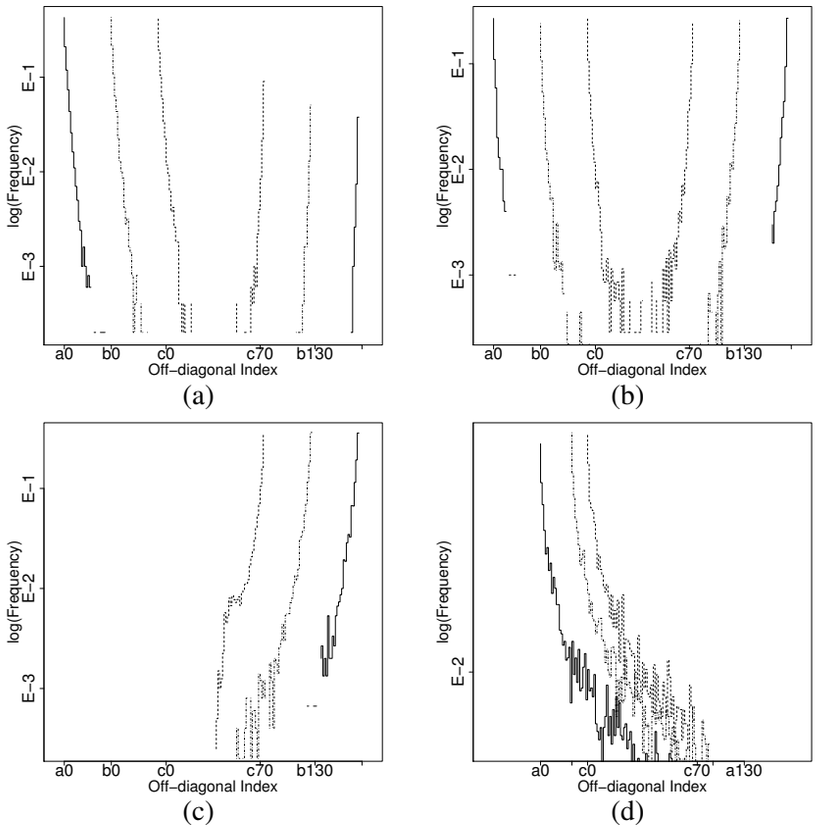


Figure 20. Empirical distributions of the deflation index $\iota_{n,\epsilon}$ for the Toda algorithm. Figures show histograms of frequency with which deflation occurs at a given off-diagonal index. Ensembles are as in Figure 19.

in this block accelerate the unshifted algorithm greatly. In contrast with the QR algorithm, the Toda algorithm deflates at both the upper-left and lower-right corner of the matrix (Figure 20). Note though that deflation is still predominantly at the corners of the matrix. Similar statistics for other ensembles may be found in [Pfrang 2011].

4. Methods and implementation

The algorithms were implemented in Python and run on a computing cluster using the module `mpi4py`. For numerical computations we relied on the `scipy` module except in the case of the RKPW spectral reconstruction procedure [Gragg and Harrod 1984] which was implemented in `C`. Our simulation strategy was

to generate a number N of samples for each (ϵ, n)-pair of tolerances given by $\epsilon \in \{10^{-k} : k = 2, 4, 6, 8\}$ and matrix dimensions $n \in \{10, 30, \ldots, 190\}$. One initial matrix sample of size $n_i \times n_i$ is used to generate deflation time and deflation index samples for all pairs $(\tilde{\epsilon}, n_i)$, where $\tilde{\epsilon}$ is in our list of tolerances. To do this we advance the matrix using the algorithm under consideration until we undercut each of the tolerances in the list and save the corresponding statistics along the way. Typically for each (ϵ, n)-combination we generate between 1000 and 5000 samples. In the following we present a short summary of the implementation strategies chosen for the individual algorithms.

4.1. ***QR algorithm.*** Our simulation code uses the QR decomposition and matrix multiplication methods provided by `scipy` for the case of full symmetric matrices. For Jacobi matrices we implemented the efficient (unshifted) QR step presented for example in [Golub and Van Loan 1996]. We augment these implementations to include the Wilkinson shift by subtracting (adding) the shift value before (after) the QR step, respectively.

4.2. ***Toda algorithm.*** Both Jacobi and full symmetric matrices are treated similarly for this algorithm. The implementation uses the QR representation (8) to generate Toda steps T_n as follows:

$$M_k = \exp(T_k) = Q_k R_k, \tag{38}$$

$$M_{k+1} = R_k Q_k, \tag{39}$$

$$T_{k+1} = \log(M_{k+1}). \tag{40}$$

Our implementation uses `scipy` routines for the matrix exponentials and matrix logarithms. Note that in general the matrix exponential of a Jacobi matrix is full symmetric. `scipy` is also used for the QR decomposition and standard matrix multiplication routines for the reverse order multiplication.

Note that we do not use an ordinary differential equation solver to solve (4) and diagonalize the matrix as proposed in [Deift et al. 1983]. This is because our goal here is not to develop a competitive numerical scheme, but to compute reliable statistics of the deflation time for different algorithms. The above numerical scheme based on QR factorization was validated against both an ordinary differential equation solver based method and the use of the explicit solution (8) with the RKPW implementation of the inverse spectral map.

5. Acknowledgments

The numerical results presented here are part of the first author's Ph.D dissertation at Brown University [Pfrang 2011]. The help of the support staff at the Center for Computation and Visualization at Brown University is gratefully acknowledged.

We also thank Jim Demmel, Luen-Chau Li, Irina Nenciu and Nick Trefethen for their interest in this study. Finally, we thank the anonymous referee for several valuable comments and suggestions regarding our work.

References

[Armentano 2014] D. Armentano, "Complexity of path-following methods for the eigenvalue problem", *Found. Comput. Math.* **14**:2 (2014), 185–236.

[Bai et al. 1997] Z. Bai, J. Demmel, and M. Gu, "An inverse free parallel spectral divide and conquer algorithm for nonsymmetric eigenproblems", *Numer. Math.* **76**:3 (1997), 279–308.

[Clauset et al. 2009] A. Clauset, C. R. Shalizi, and M. E. J. Newman, "Power-law distributions in empirical data", *SIAM Rev.* **51**:4 (2009), 661–703.

[Deift 1999] P. A. Deift, *Orthogonal polynomials and random matrices: a Riemann–Hilbert approach*, Courant Lecture Notes in Mathematics **3**, Courant Institute of Mathematical Sciences, New York, 1999.

[Deift et al. 1983] P. Deift, T. Nanda, and C. Tomei, "Ordinary differential equations and the symmetric eigenvalue problem", *SIAM J. Numer. Anal.* **20**:1 (1983), 1–22.

[Deift et al. 1986] P. Deift, L.-C. Li, T. Nanda, and C. Tomei, "The Toda flow on a generic orbit is integrable", *Comm. Pure Appl. Math.* **39**:2 (1986), 183–232.

[Deift et al. 1993] P. Deift, L.-C. Li, and C. Tomei, "Symplectic aspects of some eigenvalue algorithms", pp. 511–536 in *Important developments in soliton theory*, edited by A. S. Fokas and V. E. Zakharov, Springer, Berlin, 1993.

[Deift et al. 1996] P. Deift, C. D. Levermore, and C. E. Wayne (editors), *Dynamical systems and probabilistic methods in partial differential equations: Proceedings of the 1994 AMS-SIAM Summer Seminar* (Berkeley, California, June 20–July 1, 1994), Lectures in Applied Mathematics **31**, American Mathematical Society, Providence, RI, 1996.

[Demmel 1988] J. W. Demmel, "The probability that a numerical analysis problem is difficult", *Math. Comp.* **50**:182 (1988), 449–480.

[Demmel 1997] J. W. Demmel, *Applied numerical linear algebra*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[Diaconis and Wood 2010] P. Diaconis and P. Wood, "Random doubly stochastic Jacobi matrices", 2010. Unpublished manuscript.

[Dumitriu and Edelman 2002] I. Dumitriu and A. Edelman, "Matrix models for beta ensembles", *J. Math. Phys.* **43**:11 (2002), 5830–5847.

[Edelman 1988] A. Edelman, "Eigenvalues and condition numbers of random matrices", *SIAM J. Matrix Anal. Appl.* **9**:4 (1988), 543–560.

[Edelman and Rao 2005] A. Edelman and N. R. Rao, "Random matrix theory", *Acta Numer.* **14** (2005), 233–297.

[Edelman and Sutton 2007] A. Edelman and B. D. Sutton, "From random matrices to stochastic operators", *J. Stat. Phys.* **127**:6 (2007), 1121–1165.

[Erdős and Yau 2012] L. Erdős and H.-T. Yau, "Universality of local spectral statistics of random matrices", *Bull. Amer. Math. Soc.* (*N.S.*) **49**:3 (2012), 377–414.

[Goldstine and von Neumann 1951] H. H. Goldstine and J. von Neumann, "Numerical inverting of matrices of high order, II", *Proc. Amer. Math. Soc.* **2** (1951), 188–202.

[Golub and Van Loan 1996] G. H. Golub and C. F. Van Loan, *Matrix computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[Gragg and Harrod 1984] W. B. Gragg and W. J. Harrod, "The numerically stable reconstruction of Jacobi matrices from spectral data", *Numer. Math.* **44**:3 (1984), 317–335.

[Higham 2008] N. J. Higham, *Functions of matrices: theory and computation*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2008.

[Leite et al. 2010] R. S. Leite, N. C. Saldanha, and C. Tomei, "The asymptotics of Wilkinson's shift: loss of cubic convergence", *Found. Comput. Math.* **10**:1 (2010), 15–36.

[Malyshev 1993] A. N. Malyshev, "Parallel algorithm for solving some spectral problems of linear algebra", *Linear Algebra Appl.* **188/189** (1993), 489–520.

[Mehta 2004] M. L. Mehta, *Random matrices*, 3rd ed., Pure and Applied Mathematics (Amsterdam) **142**, Elsevier/Academic Press, Amsterdam, 2004.

[Moser and Zehnder 2005] J. Moser and E. J. Zehnder, *Notes on dynamical systems*, Courant Lecture Notes in Mathematics **12**, Courant Institute of Mathematical Sciences, New York, 2005.

[Nanda 1985] T. Nanda, "Differential equations and the *QR* algorithm", *SIAM J. Numer. Anal.* **22**:2 (1985), 310–321.

[Pfrang 2011] C. W. Pfrang, *Diagonalizing random matrices with integrable systems*, Ph.D. thesis, Brown University, 2011, https://repository.library.brown.edu/studio/item/bdr:11289/.

[Rudelson and Vershynin 2008] M. Rudelson and R. Vershynin, "The Littlewood–Offord problem and invertibility of random matrices", *Adv. Math.* **218**:2 (2008), 600–633.

[Sankar et al. 2006] A. Sankar, D. A. Spielman, and S.-H. Teng, "Smoothed analysis of the condition numbers and growth factors of matrices", *SIAM J. Matrix Anal. Appl.* **28**:2 (2006), 446–476.

[Smale 1983] S. Smale, "On the average number of steps of the simplex method of linear programming", *Math. Programming* **27**:3 (1983), 241–262.

[Symes 1980] W. W. Symes, "Hamiltonian group actions and integrable systems", *Phys. D* **1**:4 (1980), 339–374.

[Symes 1981/82] W. W. Symes, "The *QR* algorithm and scattering for the finite nonperiodic Toda lattice", *Phys. D* **4**:2 (1981/82), 275–280.

[Tao and Vu 2010] T. Tao and V. Vu, "Random matrices: the distribution of the smallest singular values", *Geom. Funct. Anal.* **20**:1 (2010), 260–297.

[Tracy and Widom 1994] C. A. Tracy and H. Widom, "Level-spacing distributions and the Airy kernel", *Comm. Math. Phys.* **159**:1 (1994), 151–174.

[Trefethen and Bau 1997] L. N. Trefethen and D. Bau, III, *Numerical linear algebra*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.

[Watkins 1984] D. S. Watkins, "Isospectral flows", *SIAM Rev.* **26**:3 (1984), 379–391.

[Wilkinson 1968] J. H. Wilkinson, "Global convergence of tridiagonal QR algorithm with origin shifts", *Linear Algebra and Appl.* **1** (1968), 409–420.

christian.w.pfrang@jpmorgan.com
J. P. Morgan Securities, 383 Madison Avenue, New York, New York 10033, United States

deift@cims.nyu.edu
Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, New York 10012, United States

menon@dam.brown.edu
Division of Applied Mathematics, Brown University, 182 George Street, Providence, 02912, United States