Fast eigendecomposition of unitary upper Hessenberg matrices

18.338 final project

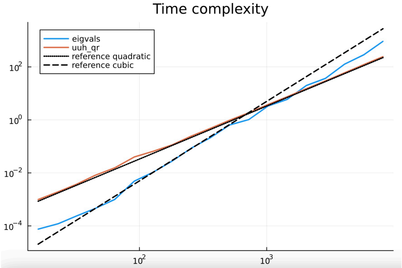
Nicholas West

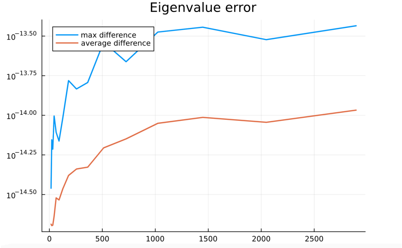
December 9, 2024

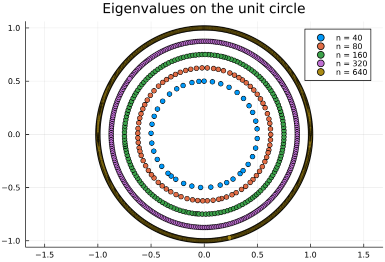
Aim of the report

- Implement a fast QR algorithm for computing eigenvalues of unitary upper Hessenberg matrices ($O(n^2)$ time, $O(n)$ space)
- RMT application: sampling eigenvalues of unitary matrices in $O(n^2)$ time

Summary of results: 3 plots







Main idea of algorithm [1]

- A unitary matrix is similar to a unitary upper hessenberg matrix which can be represented in $O(n)$ parameters
- A shifted QR algorithm can be implemented such that each iterate costs $O(n)$ flops

Visual algorithm

(Blackboard)

Algorithm details

Ingredients:

- Factored form: $U = C_1 C_2 \cdots C_{n-1}$
- Shift computation: $\mu \approx \mathrm{eig}(U[n-1:n,\ n-1:n])$
- Bulge computation: Compute $x = (U - \mu I)e_1$, solve $B_1^* x = \gamma e_1$
- Bulge chase: $U \to B_1^* U B_1$. Fuse, turnover (t), similarity (s), t, s, t, s, $\cdots$, t, s, fuse. $B_1^* U B_1 = \tilde{C}_1 \tilde{C}_2 \cdots \tilde{C}_{n-1}$
- Deflation: If $C_{n-1} \approx$ diagonal, save $\lambda_n = U[n,n]$, decouple system. Repeat on $C_1 \cdots C_{n-2}$

Random matrix connection

- Main idea: can sample factored form with same eigenvalues as random unitary

Random matrix connection

- Main idea: can sample factored form with same eigenvalues as random unitary

Sampling algorithm [2]

For $i = 1, 2, \cdots, n-1$

- Sample $\alpha_j \sim N_{\mathbb{C}}(0,1)$, $\beta_j \sim \sqrt{\chi^2_{\mathbb{C}}(n-j)}$
- Form $P_i = I - 2\dfrac{v_j v_j^*}{v_j^* v_j}$ where

$$v_j = \begin{bmatrix} 0_{j-1} \\ \alpha_j + \exp(i\,\mathrm{Arg}(\alpha_j))\sqrt{|\alpha_j|^2 + |\beta_j|^2} \\ \beta_j \\ 0_{n-j-1} \end{bmatrix}$$

- Form $D_i = -\exp(i\,\mathrm{Arg}(\alpha_j))$

Sample $D_n \sim \mathrm{Uniform}$ on unit circle.
Compute $C_1 C_2 \cdots C_{n-1} = P_1 P_2 \cdots P_{n-1} D$

Eigs of $C_1 C_2 \cdots C_{n-1} \sim$ eigs of U a random unitary matrix

Why does this work? (Blackboard)

- Recall naive random unitary
- Improvement: sample Householders directly
- Improvement: similar to core transformations that can be sampled directly

Next steps

- Implement fixed determinants
- Make into a package
- Optimize code
- Real arithmetic, double shift for orthogonal matrices

Bibliography I

[1] Jared L. Aurentz et al. "Fast and stable unitary QR algorithm". In: *Electron. Trans. Numer. Anal* 44 (2015), pp. 327–341.

[2] Massimiliano Fasi and Leonardo Robol. "Sampling the eigenvalues of random orthogonal and unitary matrices". In: *Linear Algebra and its Applications* 620 (July 2021), pp. 297–321. ISSN: 0024-3795. DOI: 10.1016/j.laa.2021.02.031.