Computing Eigensystem

Zhengbo Zhou*

January 16, 2023

1 Divide and Conquer

1.1 Introduction

This is currently the fastest method to find all the eigenvalues and eigenvectors of symmetric tridiagonal matrices larger than $n = 25$. In the worst cases, divide-and-conquer requires $O(n^3)$ flops, but in practice, the constant is quite small. Over a large of set of random test cases, it appears to take only $O(n^{2.3})$ flops on average, and as low as $O(n^2)$ for some eigenvalue distribution.

In theory, divide-and-conquer could be implemented to run on $O(n \log^p n)$ flops, where p is a small integer. This super-fast implementation uses the fast multi-pole method (FMM), originally invented for the completely different problem of computing the mutual forces on n electrically charged particles. But the complexity of this super-fast implementation means that QR iteration is currently the algorithm of choice for finding all eigenvalues, and divide-and-conquer without the FMM is the method of choice for finding all eigenvalues and all eigenvectors.

1.2 Overview

It is quite subtle to implement in a numerically stable way. Indeed, although this method was first introduced in 1981, the "right" implementation was not discovered until 1992. This routine is available as LAPACK routines `ssyevd` for dense matrices and `sstevd` for tridiagonal matrices. This routine uses divide-and-conquer for matrices of dimension larger than 25 and automatically switches to QR iteration for smaller matrices.

*Department of Mathematics, University of Manchester, Manchester, M13 9PL, England (zhengbo.zhou@postgrad.manchester.ac.uk).

$$
T = \left[\begin{array}{cccc|cccc}
a_1 & b_1 & & & & & & \\
b_1 & \ddots & \ddots & & & & & \\
 & \ddots & a_{m-1} & b_{m-1} & & & & \\
 & & b_{m-1} & a_m & b_m & & & \\
\hline
 & & & b_m & a_{m+1} & b_{m+1} & & \\
 & & & & b_{m+1} & \ddots & & \\
 & & & & & & \ddots & b_{n-1} \\
 & & & & & & b_{n-1} & a_n
\end{array}\right]
$$

$$
= \left[\begin{array}{cccc|cccc}
a_1 & b_1 & & & & & & \\
b_1 & \ddots & \ddots & & & & & \\
 & \ddots & a_{m-1} & b_{m-1} & & & & \\
 & & b_{m-1} & a_m - b_m & & & & \\
\hline
 & & & & a_{m+1} - b_m & b_{m+1} & & \\
 & & & & b_{m+1} & \ddots & & \\
 & & & & & \ddots & b_{n-1} & \\
 & & & & & b_{n-1} & a_n &
\end{array}\right] + \left[\begin{array}{cc|cc}
 & b_m & b_m & \\
\hline
 & b_m & b_m &
\end{array}\right] \tag{1.1}
$$

$$
= \left[\begin{array}{c|c} T_1 & 0 \\ \hline 0 & T_2 \end{array}\right] + b_m \cdot \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} [0, \ldots, 0, 1, 1, 0, \ldots, 0] \equiv \left[\begin{array}{c|c} T_1 & 0 \\ \hline 0 & T_2 \end{array}\right] + b_m v v^T.
$$

Suppose that we have the eigendecomposition of T_1 and T_2: $T_i = Q_i \Lambda_i Q_i^T$. These will be computed recursively by this same algorithm. We related the eigenvalues of T to those of T_1 and T_2 as follows:

$$
\begin{aligned}
T &= \begin{bmatrix} T_1 & 0 \\ 0 & T_2 \end{bmatrix} + b_m v v^T \\
&= \begin{bmatrix} Q_1 \Lambda_1 Q_1^T & 0 \\ 0 & Q_2 \Lambda_2 Q_2^T \end{bmatrix} + b_m v v^T \\
&= \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix} \left(\begin{bmatrix} \Lambda_1 & \\ & \Lambda_2 \end{bmatrix} + b_m u u^T \right) \begin{bmatrix} Q_1^T & 0 \\ 0 & Q_2^T \end{bmatrix}
\end{aligned}
$$

where

$$
u = \begin{bmatrix} Q_1^T & 0 \\ 0 & Q_2 \end{bmatrix}, \qquad v = \begin{bmatrix} \text{last column of } Q_1^T \\ \text{first column of } Q_2^T \end{bmatrix}
$$

since $v = [0, \ldots, 0, 1, 1, 0, \ldots, 0]^T$. Therefore, the eigenvalues of T are the same as those of the similar matrix $D + \rho u u^T$ where $D = \begin{bmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{bmatrix}$ is diagonal, $\rho = b_m$ is a scalar, and

u is a vector. Henceforth, we will assume without loss of generality that the diagonal $d_1, \cdots, d_n$ of D is sorted: $d_n \leq \cdots \leq d_n$.

To find the eigenvalues of $D + \rho u u^T$, assume first that $D - \lambda I$ is nonsingular, and compute the characteristic polynomial as follows:

$$\det(D + \rho u u^T - \lambda I) = \det((D - \lambda I)(I + \rho(D - \lambda I)^{-1} u u^T)). \tag{1.2}$$

Since $D - \lambda I$ is nonsingular, $\det(I + \rho(D - \lambda I)^{-1} u u^T = 0$ whenever λ is an eigenvalue. Notice that $I + \rho(D - \lambda I)^{-1} u u^T$ is the identity plus a rank-1 matrix; the determinant of such a matrix is easy to compute:

Lemma 1.1. *If x and y are vectors,* $\det(I + x y^T) = 1 + y^T x$.

Therefore

$$\begin{aligned} \det(I + \rho(D - \lambda I)^{-1} u u^T) &= 1 + \rho u^T (D - \lambda I)^{-1} u \\ &= 1 + \rho \sum_{i=1}^{n} \frac{u_i^2}{d_i - \lambda} \equiv f(\lambda). \end{aligned}$$

and the eigenvalues of T are the roots of the so-called secular equation $f(\lambda) = 0$. If all d_i are distinct and all $u_i \neq 0$ (the generic case), the function $f(\lambda)$ has the graph shown in.

[**?**, 1221]

References