目　录

This page intentionally left blank.

1 箭型分而治之法

分而治之法 (Cuppen's Divide and Conquer, CDC) 首先由 Cuppen 于 1981 年提出 [1]. Gu 和 Eisenstat 于 1995 年给出了一种快速稳定的分而治之法实现方式, 称为 **箭型分而治之法** (Arrowhead Divide-and-Conquer, ADC). 他们做了大量的数值试验, 在试验中, 他们采用修正的有理逼近法求解特征方程. 数值结果表明, ADC 算法的计算精度可以与其他算法媲美, 而计算速度通常比对称 QR 和 CDC 要快, 详见 [2, 3].

1.1 算法基本框架

给定对称三对角矩阵 $T \in \mathbb{R}^{n\times n}$.

► **矩阵划分方法**:

$$
T = \left[\begin{array}{cccc|c|cccc}
a_1 & b_1 & & & & & & & \\
b_1 & \ddots & \ddots & & & & & & \\
 & \ddots & \ddots & b_{m-1} & & & & & \\
 & & b_{m-1} & a_m & b_m & & & & \\
\hline
 & & & b_m & a_{m+1} & b_{m+1} & & & \\
\hline
 & & & & b_{m+1} & a_{m+2} & b_{m+2} & & \\
 & & & & & b_{m+2} & \ddots & \ddots & \\
 & & & & & & \ddots & \ddots & b_{n-1} \\
 & & & & & & & b_{n-1} & a_n
\end{array}\right]_{n\times n}
$$

$$
= \boxed{\left[\begin{array}{c|c|c}
T_1 & b_m e_m & 0 \\
\hline
b_m e_m^\intercal & a_{m+1} & b_{m+1} e_1^\intercal \\
\hline
0 & b_{m+1} e_1^\intercal & T_2
\end{array}\right]},
$$

其中 $m = [n/2]$, $e_m = [0, \ldots, 0, 1]^\intercal \in \mathbb{R}^m$, $e_1 = [1, 0, \ldots, 0] \in \mathbb{R}^{n-m-1}$.

► **递推计算方法**: 假设小矩阵 T_1 和 T_2 的特征值和特征向量已知, 即

$$
T_1 = Q_1 \Lambda_1 Q_1^\intercal, \quad T_2 = Q_2 \Lambda_2 Q_2^\intercal,
$$

其中 Λ_1 和 Λ_2 为对角矩阵, Q_1, Q_2 为正交矩阵, 则有

$$
T = \begin{bmatrix}
Q_1 \Lambda_1 Q_1^\intercal & b_m e_m & 0 \\
b_m e_m^\intercal & a_{m+1} & b_{m+1} e_1^\intercal \\
0 & b_{m+1} e_1^\intercal & Q_2 \Lambda_2 Q_2^\intercal
\end{bmatrix}
$$

$$
\begin{aligned}
&= \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & Q_2 \end{bmatrix} \begin{bmatrix} \Lambda_1 & b_m Q_1^\intercal e_m & 0 \\ b_m e_m^\intercal Q_1 & a_{m+1} & b_{m+1} e_1^\intercal Q_2 \\ 0 & b_{m+1} Q_2^\intercal e_1^\intercal & \Lambda_2 \end{bmatrix} \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & Q_2 \end{bmatrix}^\intercal \\
&= \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & Q_2 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ I_m & 0 & 0 \\ 0 & 0 & I \end{bmatrix}^\intercal H \begin{bmatrix} 0 & 1 & 0 \\ I_m & 0 & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} Q_1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & Q_2 \end{bmatrix}^\intercal \\
&= \begin{bmatrix} 0 & Q_1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & Q_2 \end{bmatrix} H \begin{bmatrix} 0 & Q_1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & Q_2 \end{bmatrix}^\intercal ,
\end{aligned}
$$

其中

$$
\begin{aligned}
H &= \begin{bmatrix} 0 & 1 & 0 \\ I_m & 0 & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} \Lambda_1 & b_m Q_1^\intercal e_m & 0 \\ b_m e_m^\intercal Q_1 & a_{m+1} & b_{m+1} e_1^\intercal Q_2 \\ 0 & b_{m+1} Q_2^\intercal e_1^\intercal & \Lambda_2 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ I_m & 0 & 0 \\ 0 & 0 & I \end{bmatrix}^\intercal \\
&= \boxed{\begin{bmatrix} a_{m+1} & b_m e_m^\intercal Q_1 & b_{m+1} e_1^\intercal Q_2 \\ b_m Q_1^\intercal e_m & \Lambda_1 & 0 \\ b_{m+1} Q_2^\intercal e_1^\intercal & 0 & \Lambda_2 \end{bmatrix}} .
\end{aligned}
$$

因此只要计算 H 的特征值和特征向量即可.

> 由于 H 的形状像一个箭头 (比如四阶矩阵形如 $\begin{bmatrix} * & * & * & * \\ * & * & & \\ * & & * & \\ * & & & * \end{bmatrix}$), 因此称之为 **箭型矩阵**.

算法 1.1. 计算对称三对角矩阵的特征值和特征向量的箭型分而治之法 (ADC)

```
function [Q, Λ] = eig_adc(T)
if T is small enough then
    compute eigenvalues and eiganvectors of T with QR method
    return
end if
form T_1 and T_2
[Q_1, Λ_1] = eig_adc(T_1)
[Q_2, Λ_2] = eig_adc(T_2)
form H = [a_{m+1}  b_m e_m^T Q_1  b_{m+1} e_1^T Q_2; b_m Q_1^T e_m  Λ_1  0; b_{m+1} Q_2^T e_1^T  0  Λ_2]
compute the eigenvalues Λ and eigenvectors Q̂ of H
```

11: compute the eigenvectors of T with $Q = \begin{bmatrix} 0 & Q_1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & Q_2 \end{bmatrix} \hat{Q}$

12: end

1.2 箭型矩阵特征值与特征向量

为了讨论方便, 将矩阵 H 改写为如下形式

$$\boxed{H = \begin{bmatrix} \alpha & u^\intercal \\ u & D \end{bmatrix}},$$

其中 $\alpha = a_{m+1}$,

$$u = \begin{bmatrix} b_m Q_1^\intercal e_m \\ b_{m+1} Q_2^\intercal e_1^\intercal \end{bmatrix} \triangleq [u_1, u_2, \ldots, u_{n-1}]^\intercal, \quad D = \mathrm{diag}(d_1, d_2, \ldots, d_{n-1}),$$

并假定 $d_1 \geq d_2 \geq \cdots \geq d_{n-1}$.

定理 1.1: 箭型矩阵的特征值与特征向量

设 H 的特征值为 $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, 并假定 $d_1 > d_2 > \cdots > d_{n-1}$, 且所有 u_i 都非零, 则

(1) $\lambda_1 > d_1 > \lambda_2 > d_2 > \cdots > \lambda_{n-1} > d_{n-1} > \lambda_n$.

(2) λ_i 满足特征方程 $f(\lambda) = 0$, 其中

$$\boxed{f(\lambda) \triangleq \lambda - \alpha + \sum_{k=1}^{n-1} \frac{u_k^2}{d_k - \lambda}}$$

且 H 的特征值正好是 $f(\lambda)$ 的 n 个零点.

(3) 与 λ_i 对应的特征向量为

$$\left[-1, \frac{u_1}{d_1 - \lambda_i}, \ldots, \frac{u_{n-1}}{d_{n-1} - \lambda_i}\right].$$

证明. (1) 根据对称矩阵特征值分隔定理可知

$$\lambda_1 \geq d_1 \geq \lambda_2 \geq d_2 \geq \cdots \lambda_{n-1} \geq d_{n-1} \geq \lambda_n.$$

因此只需证明 λ_i 与 D 的对角线都不相等. 反证法, 假定存在某个 d_j 使得 $\lambda_i = d_j$, 并设对应的特征向量为 $x = [x_1, x_2, \ldots, x_n]^\intercal$, 则由 $Hx = \lambda_i x$ 可知

$$\alpha x_1 + u_1 x_2 + \cdots + u_{n-1} x_n = \lambda_i x_1, \tag{1.1}$$

$$u_k x_1 + d_k x_{k+1} = \lambda_i x_{k+1}, \quad k = 1, 2, \ldots, n-1. \tag{1.2}$$

又 $\lambda_i = d_j$, 因此在 (1.2) 中取 $k=j$, 可推出 $u_j x_1 = 0$, 故 $x_1 = 0$. 代入 (1.2), 根据 d_k 互不相等可推出 $x_{k+1} = 0$, $k = 1, \ldots, k-1, k+1, \ldots, n-1$. 代入 (1.1), 可得 $x_k = 0$. 所以 $x = 0$, 矛盾. 所以结论 (1) 成立.

(2) 根据等式 (1.2), 我们有

$$x_{k+1} = -\frac{u_k x_1}{d_k - \lambda_i}, \quad k = 1, 2, \ldots, n-1. \tag{1.3}$$

代入 (1.1) 式后可得

$$\left(\alpha - \sum_{k=1}^{n-1} \frac{u_k^2}{d_k - \lambda_i}\right) x_1 = \lambda_i x_1. \tag{1.4}$$

又 $x_1 \neq 0$ (否则由 (1.3) 可推出所有 x_k 都为 0, 矛盾), 所以有

$$\lambda_i - \alpha + \sum_{k=1}^{n-1} \frac{u_k^2}{d_k - \lambda_i} = 0,$$

即 λ_i 满足特征方程 $f(\lambda) = 0$.

下面证明 $f(\lambda)$ 的 n 个零点恰好是 H 的 n 个特征值. 只需证明 $f(\lambda)$ 恰好有 n 个零点. 直接计算可知

$$f'(\lambda) = 1 + \sum_{i=1}^{n-1} \frac{u_i^2}{(d_i - \lambda)^2} > 0.$$

所以 $f(\lambda)$ 在下面的 n 个区间

$$(-\infty, d_{n-1}),\ (d_{n-1}, d_{n-2}),\ \ldots,\ (d_2, d_1),\ (d_1, +\infty) \tag{1.5}$$

内都严格单调递增, 且

$$\lim_{\lambda \to -\infty} f(\lambda) = -\infty, \quad \lim_{\lambda \to +\infty} f(\lambda) = +\infty, \quad \lim_{\lambda \to d_k - 0} f(\lambda) = +\infty, \quad \lim_{\lambda \to d_k + 0} f(\lambda) = -\infty,$$

因此 $f(\lambda)$ 在每个区间内恰好有一个零点, 总共有 n 个零点. 所以结论 (2) 成立.

(3) 由前面的讨论可知, $x_1 \neq 0$. 取 $x_1 = -1$, 代入 (1.3) 即得结论 (3) 成立. □

- 如果我们记 $d_n = -\infty$, $d_0 = +\infty$, 则 H 的特征值满足: $\boxed{\lambda_i \in (d_i, d_{i-1})}$, $i = 1, 2, \ldots, n$.
- 实际计算时, 特征向量需要单位化.
- 如果特征值精度不够, 会影响特征向量的正交性.

定理 1.2: 收缩 Deflation

考虑箭型矩阵 H,

(1) 若 u 的某个分量 $u_i = 0$, 则 $\lambda = d_i$ 是 H 的特征值, 对应的特征向量为 e_{i+1}.

(2) 若存在 i 使得 $d_{i+1} = d_i$, 则 $\lambda = d_i$ 是 H 的特征值.

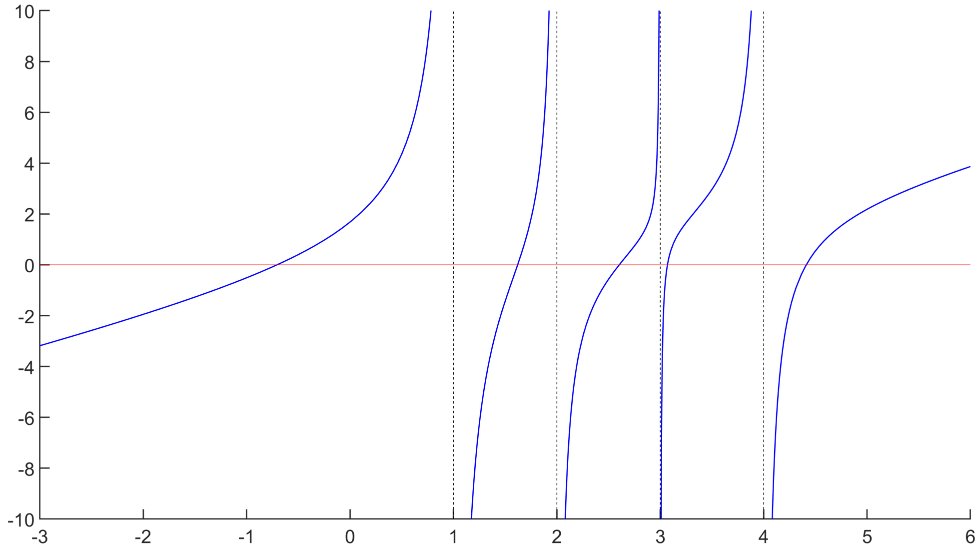

图 1.1. $f(\lambda)$ 的示意图: $f(\lambda)=\lambda-1+\dfrac{2}{1-\lambda}+\dfrac{0.8}{2-\lambda}+\dfrac{0.1}{3-\lambda}+\dfrac{1}{4-\lambda}$ 的图像

1.2.1 特征方程的预处理

基于计算稳定性和精度方面的考虑, 我们在求解特征方程时要先做预处理, 即**平移**. 我们需要在 n 个不同区间 (见 (1.5)) 内计算 $f(\lambda)$ 的零点, 在不同的区间需要采用不同的平移变换. 我们将这 n 个区间分成两种情形: 中间 $n-2$ 个有限区间和两端的 2 个无限区间.

- 首先考虑中间 $n-2$ 个有限区间, 即计算 $f(\lambda)$ 在 (d_{i+1},d_i) 内的零点 λ_{i+1}, $i=1,2,\dots,n-2$. 分两种情形:

  (1) 特征值 λ_{i+1} 位于左半区间, 即 $\lambda_i\in\left(d_{i+1},\frac{1}{2}(d_{i+1}+d_i)\right)$. 此时做平移 $\boxed{\mu=\lambda-d_{i+1}}$, 将特征函数 $f(\lambda)$ 转化为

$$
\begin{aligned}
g(\mu)&\triangleq f(\mu+d_{i+1})=\mu+d_{i+1}-\alpha+\sum_{k=1}^{i}\frac{u_k^2}{(d_k-d_{i+1})-\mu}+\sum_{k=i+1}^{n-1}\frac{u_k^2}{(d_k-d_{i+1})-\mu}\\
&\triangleq\boxed{\mu+\alpha_{i+1}+\Psi_1(\mu)+\Psi_2(\mu)},
\end{aligned}
$$

  其中 $\alpha_{i+1}=d_{i+1}-\alpha$,

$$
\boxed{\Psi_1(\mu)=\sum_{k=1}^{i}\frac{u_k^2}{\delta_k-\mu}},\quad \boxed{\Psi_2(\mu)=\sum_{k=i+1}^{n-1}\frac{u_k^2}{\delta_k-\mu}},\quad \boxed{\delta_k=d_k-d_{i+1}},\quad k=1,2,\dots,n-1.
$$

  于是原问题就转化为 $\boxed{\text{求解 } g(\mu) \text{ 在 } (0,\delta_i/2) \text{ 中的零点}}$, 其中 $\delta_i=d_i-d_{i+1}>0$.

这样做平移的好处是，对任意 $\mu \in (0, \delta_i/2)$，计算 $g(\mu)$ 的值都能达到相对高的精度．事实上，计算 $f(\lambda)$ 可能出现较大误差的因素主要集中在 $\dfrac{u_{i+1}^2}{d_{i+1}-\lambda}$：当 λ 十分接近 d_{i+1} 时，计算 $d_{i+1}-\lambda$ 就会出现相近的数相减的问题，从而导致有效数字的损失．但经过平移后，这一项就变为 $\dfrac{u_{i+1}^2}{-\mu}$，避免了相近的数相减的问题．可以证明 [3]，

$$|\tilde{g}(\mu) - g(\mu)| \le c\varepsilon_u \Big(|\mu| + |\alpha_{i+1}| + |\Psi_1(\mu)| + \Psi_2(\mu)\Big), \tag{1.6}$$

其中 $g(\mu)$ 表示精确值，$\tilde{g}(\mu)$ 表示计算值，ε_u 表示机器精度，c 是与 μ 和 n 无关的小正数．

(2) 特征值 λ_{i+1} 位于右半区间，即 $\lambda_{i+1} \in \left(\frac{1}{2}(d_{i+1}+d_i), d_i\right)$．相应地，做平移 $\boxed{\mu = \lambda - d_i}$，将特征函数 $f(\lambda)$ 转化为

$$\boxed{g(\mu) \triangleq \mu + \alpha_i + \Psi_1(\mu) + \Psi_2(\mu)},$$

其中 $\alpha_i = d_i - \alpha$，

$$\boxed{\Psi_1(\mu) = \sum_{k=1}^{i} \frac{u_k^2}{\delta_k - \mu}}, \quad \boxed{\Psi_2(\mu) = \sum_{k=i+1}^{n-1} \frac{u_k^2}{\delta_k - \mu}}, \quad \boxed{\delta_k = d_k - d_i}, \quad k = 1, 2, \dots, n-1.$$

于是原问题就转化为 求解 $g(\mu)$ 在 $(\delta_{i+1}/2, 0)$ 中的零点，其中 $\delta_{i+1} = d_{i+1} - d_i < 0$．

可以通过计算 $f(\lambda)$ 在区间中点的函数值来判断 λ_{i+1} 的位置：若 $f\left(\frac{1}{2}(d_{i+1}+d_i)\right) > 0$ 则位于左半区间，否则位于右半区间．

- 下面考虑计算 $f(\lambda)$ 在 $(-\infty, d_{n-1})$ 和 $(d_1, +\infty)$ 内的零点 λ_n 和 λ_1．

(1) 计算 $\lambda_n \in (-\infty, d_{n-1})$．先做平移 $\boxed{\mu = \lambda - d_{n-1}}$，将特征函数 $f(\lambda)$ 转化为

$$\boxed{g(\mu) \triangleq \mu + \alpha_{n-1} + \Psi_1(\mu)},$$

其中 $\alpha_{n-1} = d_{n-1} - \alpha$，

$$\boxed{\Psi_1(\mu) = \sum_{k=1}^{n-1} \frac{u_k^2}{\delta_k - \mu}}, \quad \boxed{\delta_k = d_k - d_{n-1}}, \quad k = 1, 2, \dots, n-1.$$

由于 $(-\infty, d_{n-1})$ 是个无限区间，不利于数值计算，因此我们考虑对求解区间进行更加精细的估计．易知

$$H - d_{n-1}I = \mathrm{diag}(0, \delta_1, \dots, \delta_{n-1}) + \begin{bmatrix} -\alpha_{n-1} & u^{\mathsf{T}} \\ u & 0 \end{bmatrix} \triangleq \mathrm{diag}(0, \delta_1, \dots, \delta_{n-1}) + E.$$

根据对称矩阵特征值扰动分析理论 (Weyl 定理 [?]) 可知

$$|\lambda_n - d_{n-1}| \le \|E\|_2 \le \boxed{|\alpha_{n-1}| + \|u\|_2 \triangleq \eta}.$$

因此 $\lambda_n \in (d_{n-1} - \eta, d_{n-1})$，于是原问题就转化为 求解 $g(\mu)$ 在 $(-\eta, 0)$ 中的零点．

(2) 计算 $\lambda_1 \in (d_1, +\infty)$. 做平移 $\boxed{\mu = \lambda - d_1}$, 则特征函数 $f(\lambda)$ 转化为

$$\boxed{g(\mu) \triangleq \mu + \alpha_1 + \Psi_2(\mu)},$$

其中 $\alpha_1 = d_1 - \alpha$,

$$\boxed{\Psi_2(\mu) = \sum_{k=1}^{n-1} \frac{u_k^2}{\delta_k - \mu}}, \quad \boxed{\delta_k = d_k - d_1}, \ k = 1, 2, \dots, n-1.$$

类似地, 我们可以证明 $\lambda_1 \in (d_1, d_1 + \eta)$, 其中 $\eta = |\alpha_1| + \|u\|_2$. 于是原问题就转化为 $\boxed{\text{求解 } g(\mu) \text{ 在 } (0, \eta) \text{ 中的零点}}$.

1.2.2 特征方程的数值求解

由于 $g(\mu)$ 的特点, 我们可以采用对分法或者有理函数插值法来求解, 与 CDC 中的特征方程数值求解类似. 根据 [3] 中的说法, 采用什么方法求解特征方程并非关键, 重要的是采用何种停机准则, 以及平移预处理.

论文 [3] 中给出了一个比较合理的停机准则

$$|g(\tilde{\mu})| \leq c\varepsilon_u n\Big(|\mu| + |\alpha_{i+1}| + |\Psi_1(\mu)| + \Psi_2(\mu)\Big), \tag{1.7}$$

即误差估计式 (1.6) 的上界. 此时可以将 $\tilde{\mu}$ 作为近似解.

1.2.3 特征向量的计算

我们可以直接采用定理 1.1 中的结论 (3) 来计算特征向量. 但考虑到数值稳定性, 特别是当 λ_i 与 d_{i+1} (或 d_i) 非常接近时, 虽然数值解 $\tilde{\lambda}_i$ 与精确解 λ_i 也非常接近, 但 $\dfrac{u_{i+1}^2}{d_{i+1} - \lambda_{i+1}}$ 与 $\dfrac{u_{i+1}^2}{d_{i+1} - \tilde{\lambda}_{i+1}}$ (或 $\dfrac{u_i^2}{d_i - \lambda_i}$ 与 $\dfrac{u_i^2}{d_i - \tilde{\lambda}_i}$) 可能会存在很大的误差.

我们注意到, 如果 $\tilde{\lambda}_i$ 与 λ_i 严格相等, 那就不存在这样的问题. 因此我们可以构造一个新的箭型矩阵 $\tilde{H}$, 使得 $\tilde{\lambda}_1, \tilde{\lambda}_2, \dots, \tilde{\lambda}_n$ 恰好是 $\tilde{H}$ 的特征值, 然后用 $\tilde{H}$ 的特征向量作为 H 的近似特征向量. 这样做的好处是计算 $\tilde{H}$ 特征向量能达到相对高的精度, 而且特征向量之间的正交性能达到机器精度.

矩阵 $\tilde{H}$ 可以依据下面的定理来构造.

定理 1.3

设给定的 n 个实数 $\tilde\lambda_1, \tilde\lambda_2, \dots, \tilde\lambda_n$ 和对角矩阵 $D = \mathrm{diag}(d_1, d_2, \dots, d_{n-1})$ 满足下面的分隔条件

$$\tilde\lambda_n < d_{n-1} < \tilde\lambda_{n-1} < d_{n-2} < \cdots < \tilde\lambda_2 < d_1 < \tilde\lambda_1,$$

则存在箭型矩阵

$$\tilde H = \begin{bmatrix} \tilde\alpha & \tilde u^\intercal \\ \tilde u & D \end{bmatrix},$$

使得 $\tilde\lambda_1, \tilde\lambda_2, \dots, \tilde\lambda_n$ 恰好是 $\tilde H$ 的特征值, 其中

$$\tilde\alpha = \tilde\lambda_n + \sum_{k=1}^{n-1} (\tilde\lambda_k - d_k), \tag{1.8}$$

$$\tilde u_i^2 = (d_i - \tilde\lambda_n)(\tilde\lambda_1 - d_i) \prod_{k=1}^{i-1} \frac{\tilde\lambda_{k+1} - d_i}{d_k - d_i} \prod_{k=i+1}^{n-1} \frac{\tilde\lambda_k - d_i}{d_k - d_i}, \quad i = 1, 2, \dots, n-1. \tag{1.9}$$

这里 $\tilde u_i$ 的符号可以任取.

证明. 构造法. 假设 $\tilde H$ 存在, 下面通过待定系数法确定 $\tilde\alpha$ 和 $\tilde u$ 的取值.

由于 $\tilde H$ 的特征值是 $\tilde\lambda_1, \tilde\lambda_2, \dots, \tilde\lambda_n$, 所以 (矩阵的迹等于特征值之和)

$$\tilde\alpha + d_1 + \cdots + d_{n-1} = \tilde\lambda_1 + \tilde\lambda_2 + \cdots + \tilde\lambda_n.$$

因此等式 (1.8) 成立.

下面计算 $\tilde u$. 考虑 $\tilde H$ 的特征多项式, 我们有

$$\det(\lambda I - \tilde H) = \prod_{k=1}^{n} (\lambda - \tilde\lambda_k). \tag{1.10}$$

通过直接计算可得

$$\begin{aligned}
\det(\lambda I - \tilde H) &= \det\left(\begin{bmatrix} \lambda - \tilde\alpha & -\tilde u^\intercal \\ -\tilde u & \lambda I_{n-1} - D \end{bmatrix} \right) \\
&= \det\left(\begin{bmatrix} 1 & 0 \\ \dfrac{-\tilde u}{\lambda - \tilde\alpha} & I_{n-1} \end{bmatrix} \begin{bmatrix} \lambda - \tilde\alpha & -\tilde u^\intercal \\ 0 & \lambda I_{n-1} - D - \dfrac{\tilde u \tilde u^\intercal}{\lambda - \tilde\alpha} \end{bmatrix} \right) \\
&= \det\left(\begin{bmatrix} \lambda - \tilde\alpha & -\tilde u^\intercal \\ 0 & \lambda I_{n-1} - D - \dfrac{\tilde u \tilde u^\intercal}{\lambda - \tilde\alpha} \end{bmatrix} \right) \\
&= (\lambda - \tilde\alpha) \det\left(\lambda I_{n-1} - D - \frac{\tilde u \tilde u^\intercal}{\lambda - \tilde\alpha} \right) \\
&= (\lambda - \tilde\alpha) \det(\lambda I_{n-1} - D) \det\left(I_{n-1} - (\lambda - \tilde\alpha)^{-1} (\lambda I_{n-1} - D)^{-1} \tilde u \tilde u^\intercal \right) \\
&= (\lambda - \tilde\alpha) \prod_{k=1}^{n-1} (\lambda - d_k) \left(1 - (\lambda - \tilde\alpha)^{-1} \tilde u^\intercal (\lambda I_{n-1} - D)^{-1} \tilde u \right) \\
&= (\lambda - \tilde\alpha) \prod_{k=1}^{n-1} (\lambda - d_k) - \prod_{k=1}^{n-1} (\lambda - d_k) \sum_{j=1}^{n-1} \frac{\tilde u_j^2}{\lambda - d_j}
\end{aligned}$$

$$= (\lambda - \tilde{\alpha}) \prod_{k=1}^{n-1} (\lambda - d_k) - \sum_{j=1}^{n-1} \left(\tilde{u}_j^2 \prod_{k=1}^{j-1} (\lambda - d_k) \prod_{k=j+1}^{n-1} (\lambda - d_k) \right).$$

所以等式

$$(\lambda - \tilde{\alpha}) \prod_{k=1}^{n-1} (\lambda - d_k) - \sum_{j=1}^{n-1} \left(\tilde{u}_j^2 \prod_{k=1}^{j-1} (\lambda - d_k) \prod_{k=j+1}^{n-1} (\lambda - d_k) \right) = \prod_{k=1}^{n} (\lambda - \tilde{\lambda}_k)$$

对任意 λ 都成立. 分别取 $\lambda = d_i$, $i = 1, 2, \dots, n-1$, 代入上式后可得

$$-\tilde{u}_i^2 \prod_{k=1}^{i-1} (d_i - d_k) \prod_{k=i+1}^{n-1} (d_i - d_k) = \prod_{k=1}^{n} (d_i - \tilde{\lambda}_k),$$

故

$$\tilde{u}_i^2 = (d_i - \tilde{\lambda}_n)(\tilde{\lambda}_1 - d_i) \prod_{k=1}^{i-1} \frac{\tilde{\lambda}_{k+1} - d_i}{d_k - d_i} \prod_{k=i+1}^{n-1} \frac{\tilde{\lambda}_k - d_i}{d_k - d_i},$$

即等式 (1.9) 成立.

下面证明 $\tilde{H}$ 的特征值恰好是 $\tilde{\lambda}_1, \tilde{\lambda}_2, \dots, \tilde{\lambda}_n$, 即当 $\tilde{\alpha}$ 和 $\tilde{u}_i$ 由等式 (1.8) 和 (1.9) 确定时, 等式 (1.10) 对所有 λ 都成立.

将等式 (1.10) 两边的多项式展开, 可得

$$\begin{aligned} &\det(\lambda I - \tilde{H}) = \lambda^n + \operatorname{tr}(\tilde{H}) \lambda^{n-1} + \cdots, \\ &\prod_{k=1}^{n} (\lambda - \tilde{\lambda}_k) = \lambda^n + \left(\sum_{k=1}^{n} \tilde{\lambda}_k \right) \lambda^{n-1} + \cdots. \end{aligned}$$

由 $\tilde{\alpha}$ 的表达式 (1.8) 可知, 这两个多项式中 λ^{n-1} 的系数相同. 所以 $\det(\lambda I - \tilde{H}) - \prod_{k=1}^{n} (\lambda - \tilde{\lambda}_k)$ 是一个 $n-2$ 次多项式. 又根据前面的分析可知, $\det(\lambda I - \tilde{H})$ 和 $\prod_{k=1}^{n} (\lambda - \tilde{\lambda}_k)$ 在 $n-1$ 个互不相等的数 $d_1, d_2, \dots, d_{n-1}$ 上取值相同, 即 $\det(\lambda I - \tilde{H}) - \prod_{k=1}^{n} (\lambda - \tilde{\lambda}_k)$ 存在 $n-1$ 个互异的零点, 所以只能恒为 0. 定理结论成立. □

如果计算 $\tilde{\lambda}_i$ 时是按停机准则 (1.7) 得到的, 则可以证明 [3]

$$|\tilde{\lambda}_i - \lambda_i| \le \frac{4c\varepsilon_u n \|H\|_2}{1 - c\varepsilon_u n}.$$

这表明 $\tilde{\lambda}_i$ 与 λ_i 很接近. 由此可推出

$$|\tilde{\alpha} - \alpha| = \left| \sum_{k=1}^{n} (\tilde{\lambda}_i - \lambda_i) \right| \le \frac{4c\varepsilon_u n^2 \|H\|_2}{1 - c\varepsilon_u n}. \tag{1.11}$$

另外, 也可以证明 [3]

$$|\tilde{u}_i - u_i| \le 6c\varepsilon_u n^2 \|H\|_2. \tag{1.12}$$

再根据

$$\|\tilde{H} - H\|_2 = \left\| \begin{bmatrix} \tilde{\alpha} - \alpha & (\tilde{u} - u)^{\mathsf{T}} \\ \tilde{u} - u & 0 \end{bmatrix} \right\| \le |\tilde{\alpha} - \alpha| + \|\tilde{u} - u\|_2$$

可知, $\tilde{H}$ 与 H 也很接近, 因此用 $\tilde{H}$ 的特征向量来近似 H 的特征向量是合适的.

> 不等式 (1.11) 和 (1.12) 的右端项含有 n^2, 这两个 n 都分别是由于对 n 个数进行求和, 以及对 n 个数求乘积所产生的. 如果采用一些特殊的方法, 比如用二分法对 n 个数进行求和, 则浮点计算误差中的 n 可降为 $\log n$. 因此估计式 (1.11) 和 (1.12) 中的 n^2 是比较保守的, 在实际计算中有望降为 $\mathcal{O}(n)$.

1.2.4 收缩现象

在前面的讨论中, 我们假定 D 的对角线元素互不相等且 u 的所有分量全不为零 (定理 1.1). 根据定理 1.2, 如果 D 的对角线元素有相等的, 或者 u 存在零分量, 则特征值可以直接写出来. 下面就这两种情形做简单介绍.

1.2.5 u 存在分量为 0 的情形

假定 u 存在 ℓ 个零分量 $(1 \le \ell \le n-1)$. 构造置换矩阵 $P \in \mathbb{R}^{(n-1)\times(n-1)}$, 将这 ℓ 个零分量全部移到最后面, 即

$$Pu = \begin{bmatrix} \hat{u} \\ 0_\ell \end{bmatrix},$$

其中 $\hat{u} \in \mathbb{R}^{n-\ell-1}$ 的所有分量均不为 0. 于是有

$$\begin{bmatrix} 1 & 0 \\ 0 & P \end{bmatrix} H \begin{bmatrix} 1 & 0 \\ 0 & P \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} 1 & 0 \\ 0 & P \end{bmatrix} \begin{bmatrix} \alpha & u^{\mathsf{T}} \\ u & D \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & P \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} \alpha & \hat{u}^{\mathsf{T}} & 0 \\ \hat{u} & \hat{D}_1 & 0 \\ 0 & 0 & \hat{D}_2 \end{bmatrix},$$

其中 $\mathrm{diag}(\hat{D}_1, \hat{D}_2) = PDP^{\mathsf{T}}$. 所以我们只要计算一个更小规模的箭型矩阵 $\begin{bmatrix} \alpha & \hat{u}^{\mathsf{T}} \\ \hat{u} & \hat{D}_1 \end{bmatrix}$ 的特征值和特征向量.

> 实际计算中, 如果 u 的某个分量小于给定的阈值, 比如 $|u_i| < \tau \|H\|_2$ (其中 τ 的取值跟计算精度要求有关, 比如根据前面的分析 (1.12), 我们可以取 $\tau = 6c\varepsilon_u n^2$), 则我们就直接将 u_i 设为 0.

1.2.6 D 存在对角线元素相等的情形

假定存在 i 使得 $d_{i+1} = d_i$. 将 H 的第 $1, i, i+1$ 行和第 $1, i, i+1$ 列提取出来组成一个 3 阶子矩阵

$$\hat{H} = \begin{bmatrix} \alpha & u_i & u_{i+1} \\ u_i & d_i & 0 \\ u_{i+1} & 0 & d_{i+1} \end{bmatrix}.$$

构造 Givens 变换, 将 u_{i+1} 化为 0, 即

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} u_i \\ u_{i+1} \end{bmatrix} = \begin{bmatrix} r \\ 0 \end{bmatrix}, \quad c = u_i/r,\ s = u_{i+1}/r,\ r = \sqrt{u_i^2 + u_{i+1}^2}.$$

于是有

$$\begin{bmatrix} 1 & & \\ & c & s \\ & -s & c \end{bmatrix} \begin{bmatrix} \alpha & u_i & u_{i+1} \\ u_i & d_i & 0 \\ u_{i+1} & 0 & d_{i+1} \end{bmatrix} \begin{bmatrix} 1 & & \\ & c & s \\ & -s & c \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} \alpha & r & 0 \\ r & c^2 d_i + s^2 d_{i+1} & cs(d_{i+1} - d_i) \\ 0 & cs(d_{i+1} - d_i) & s^2 d_i + c^2 d_{i+1} \end{bmatrix}$$

若 $d_{i+1}=d_i$, 则上式右端是一个箭型矩阵, 且 $u_{i+1}=0$, 因此就转化为前面一种情形.

> 实际计算中, 只要 $|d_{i+1}-d_i|<\tau\|H\|_2$, 我们就可以构造 Givens 变换, 然后将 $cs(d_{i+1}-d_i)$ 设为 0.

1.3 计算奇异值分解

我们假定 A 是不可约的 $(n+1)\times n$ 二对角矩阵, 即

$$B=\begin{bmatrix} a_1 & & & \\ b_1 & a_2 & & \\ & \ddots & \ddots & \\ & & b_{n-1} & a_n \\ & & & b_n \end{bmatrix}_{(n+1)\times n}.$$

几点说明

- 如果 $A\in\mathbb{R}^{m\times n}$, 其中 $m>n$, 则可以通过 Householder 变换转化为上述形式.
- 如果 $A\in\mathbb{R}^{m\times n}$, 其中 $m<n$, 则考虑 $A^{\intercal}$ 的奇异值分解.
- 如果 $A\in\mathbb{R}^{m\times n}$, 其中 $m=n$, 则通过 Householder 变换进行二对角化, 转化为

$$B=\begin{bmatrix} a_1 & & & \\ b_1 & a_2 & & \\ & \ddots & \ddots & \\ & & b_{n-1} & a_n \end{bmatrix}_{n\times n}.$$

构造

$$\tilde{B}=\begin{bmatrix} B \\ 0 \end{bmatrix}=\begin{bmatrix} a_1 & & & \\ b_1 & a_2 & & \\ & \ddots & \ddots & \\ & & b_{n-1} & a_n \\ & & & 0 \end{bmatrix}_{(n+1)\times n}.$$

假定 B 的值为 ℓ, 设 $\tilde{B}$ 的 SVD 为

$$\tilde{B}=U\begin{bmatrix} \Sigma \\ 0 \end{bmatrix}V, \quad \Sigma=\operatorname{diag}(\sigma_1,\sigma_2,\dots,\sigma_n),$$

其中 $\sigma_1\geq\cdots\geq\sigma_\ell>\sigma_{\ell+1}=\cdots=\sigma_n=0$, $U\in\mathbb{R}^{(n+1)\times(n+1)}$ 和 $V\in\mathbb{R}^{n\times n}$ 是正交矩阵.

具体算法可参见 [2–4]

参考文献

[1] J. J. M. Cuppen, A Divide and Conquer Method for the Symmetric Tridiagonal Eigenproblem, Numerische Mathematatik, 36 (1981), 177–195. 1

[2] M. Gu and S. C. Eisenstat, A Divide-and-Conquer algorithm for the bidiagonal SVD, SIAM Journal on Matrix Analysis and Applications, 16 (1995), 79–92. 1, 11

[3] M. Gu and S. C. Eisenstat, A Divide-and-Conquer algorithm for the symmetric tridiagonal eigenproblem, SIAM Journal on Matrix Analysis and Applications, 16 (1995), 172–191. 1, 6, 7, 9

[4] 徐树方, 钱江, 矩阵计算六讲, 高等教育出版社, 北京, 2011. 11