Eigenvalue problems solved with Jacobi's method

Gustav Bredal Erikstad, Eirik Elias Prydz Fredborg
& Marte Cecilie Wegger

September 29, 2020

Abstract

In this report the Jacobi method for calculating eigenvalues is applied to the buckling beam problem. The algorithm is verified by comparing the calculated eigenvalues with eigenvalues from Armadillos *eigsym* function, and by verifying orthogonality of the eigenvectors. The algorithm is then adapted to the quantum mechanical case of one electron in a three dimensional potential well. From this optimal values for the parameters $\rho_{\max}$ and N are determined to be $3.00, 3.67, 4.00, 4.67$ for the first four eigenvalues and $N \approx 550$. Next, the algorithm is applied to the two electron case in order to investigate the effect of the strength of the potential (represented by ω_r) on the accuracy of the determined eigenvalues. The optimal ω_r is found to be between 0 and 1. Finally, possible improvements to be implemented in later studies are discussed.

I Introduction

In this report we want to study three different cases, where in all we want to solve a differential equation in order to find the eigenvalues of a tridiagonal Toeplitz matrix A by use of Jacobi's method. This method is not the most efficient and there are some factors we need to take into account which can lead to loss of numerical precision. First we want to look at a case where we have a buckling beam. We want to solve a wave function by first discretizing it, then find the eigenvalues by solving the eigenvalue problem $A\mathbf{u} = \lambda\mathbf{u}$, where λ is the eigenvalues and $\mathbf{u}$ is the eigenvectors, through transformations SAS^T where S is a rotation matrix. This we want to do in order to later compare the eigenvalues of matrix A found with Armadillo before the transformations, with the calculated eigenvalues after use of Jacobi's method. The next two cases we want to look at can be drawn to the quantum world, more specific two cases where we are looking at the harmonic oscillator problem in three dimensions, Schroedinger's equation, first with one electron and then with two. This we want to do for the same reasons as in the first case. We are going to look at these two cases as a mathematical problem. We also need to implement some unit tests in order to ensure that our code is working and returning the correct values.

II Method

A The buckling beam problem

We want to solve a wave function problem - the buckling beam problem,

$$\gamma \frac{d^2u(\rho)}{d\rho^2} = -Fu(x), \tag{1}$$

where u is the vertical displacement of the beam in the y direction, γ is a specific physical constant, and F is an applied force at (L,0), where L is the length of the beam, and $x \in [0, L]$. We set $u(0) = u(L) = 0$ by applying Dirichlet boundary conditions. By introducing a dimensionless length $\rho = x/L$, and saying that $\lambda = FL^2/\gamma$ we can rewrite Eq.(1) to

$$\frac{d^2u(\rho)}{d\rho^2} = -\lambda u(\rho), \tag{2}$$

which is an equation that becomes an eigenvalue problem when discretized. We end up with

$$-\frac{u_{i+1} - 2u_i + u_{i-1}}{h^2} = \lambda u_i, \tag{3}$$

where h is the step size defined as $(\rho_N - \rho_0)/N$, $u_i = u_{\rho_i}$ and $\rho_i = \rho_0 + ih$.

Eq.(3) can be rewritten as an eigenvalue problem and get

$$\begin{bmatrix} d & a & 0 & ... & ... & 0 & 0 \\ a & d & a & ... & ... & 0 & 0 \\ 0 & a & d & a & 0 & ... & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & ... & ... & 0 & a & d & a \\ 0 & ... & ... & 0 & 0 & a & d \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-2} \\ u_{N-1} \end{bmatrix} = \lambda \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_{N-2} \\ u_{N-1} \end{bmatrix},$$

$$\rightarrow \quad A\mathbf{u} = \lambda \mathbf{u}, \tag{4}$$

without including the endpoint u_i and u_0, and where λ is the eigenvalues, $\mathbf{u}$ is the eigenvectors, and the diagonal and non-diagonal elements are defined as $d = 2/h^2$ and $a = -1/h^2$ respectively. We define the matrix to run from $i = 1$ to $i = N - 1$. The eigenvalues and eigenvectors are given as

$$\lambda_j = d + 2a\cos\left(\frac{j\pi}{N}\right), \tag{5}$$

$$\mathbf{u}_j = \left[\sin\left(\frac{j\pi}{N}\right), \sin\left(\frac{2j\pi}{N}\right), ..., \sin\left((N-1)\frac{j\pi}{N}\right)\right]^T, \tag{6}$$

where $j = 1, 2, ..., N-1$. We later want to plot the eigenvector for the lowest eigenvalue and compare this eigenvector with the analytical solution.

Next, we want to check if a unitary transformation preserves the orthogonality of the obtained eigenvectors. This we want to find out so that we later can use a unitary transformation in order to find the eigenvalues. We have that $\mathbf{v}_i$ is

$$\mathbf{v}_i = \begin{bmatrix} v_{i1} \\ \vdots \\ \vdots \\ v_{in} \end{bmatrix},$$

and we assume that the basis is orthogonal, so

$$\mathbf{v}_j^T \mathbf{v}_i = \delta_{ij}.$$

We want to use an orthogonal transformation

$$\mathbf{w}_i = \mathbf{U}\mathbf{v}_i$$

in order to preserve the dot product and orthogonality between the eigenvectors. We have that

$$\begin{aligned} \mathbf{w}_i &= \mathbf{U}\mathbf{v}_i, \\ \mathbf{w}_j &= \mathbf{U}\mathbf{v}_j, \end{aligned}$$

so that

$$\begin{aligned} \mathbf{w}_i^T \mathbf{w}_j &= (\mathbf{U}\mathbf{v}_i)^T \mathbf{U}\mathbf{v}_j, \\ &= \mathbf{v}_i^T \mathbf{U}^T \mathbf{U}\mathbf{v}_j, \\ &= \mathbf{v}_i^T \mathbf{v}_j, \\ &= \delta_{ij}, \end{aligned}$$

which shows that the dot product and orthogonality between the eigenvectors are preserved.

B Jacobi's method

We now want to write a function which implements Jacobi's rotation algorithm. This we want to do in order to solve Eq.(4). Jacobi's method is a rotation along an axis, with the goal of finding the eigenvalues of a symmetric matrix A trough transformations, $SAS^T = D = [\lambda_1, \lambda_2, ..., \lambda_n]$, where

D is a diagonal matrix with the eigenvalues on the diagonal. The rotation matrix S is defined as

$$S = \begin{bmatrix} 1 & 0 & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & 1 & 0 & \dots & \dots & \dots & \dots & \vdots \\ \vdots & 0 & \ddots & \dots & \dots & \dots & \dots & \vdots \\ \vdots & \dots & \dots & \cos\theta & ... & \sin\theta & ... & \vdots \\ \vdots & \dots & \dots & \vdots & \ddots & \vdots & \dots & \vdots \\ \vdots & \dots & \dots & \sin\theta & ... & \cos\theta & \dots & \vdots \\ \vdots & \dots & \dots & \dots & \dots & \dots & \ddots & 0 \\ 0 & \dots & \dots & \dots & \dots & \dots & 0 & 1 \end{bmatrix}.$$

We want the non-diagonal matrix elements of the transformed matrix to be non-zero, so we need to define θ in such a way that the non-diagonal matrix elements a_{kl} become non-zero. By using the transformation SAS^T, and defining

$$\begin{aligned} s &= \sin\theta, \\ c &= \cos\theta, \\ \tan\theta = t &= s/c, \\ \cot 2\theta = \tau &= \frac{a_{ll} - a_{kk}}{2a_{kl}}, \end{aligned}$$

and by using $\cot 2\theta = 1/2(\cot\theta - tan\theta)$, we obtain

$$\begin{aligned} t^2 + 2\tau t - 1 &= 0, \\ \rightarrow \;\; t &= -\tau \pm \sqrt{1+\tau^2}. \end{aligned} \tag{7}$$

The derivation of this can be found **here**. Eq.(7) is an expression we need to worry about, when it can lead to loss of numerical precision as τ increases. When $\tau \gg 0$ we get $t \approx -\tau + \tau$, but these two τ's are not identical, which means trouble. We therefore need to rewrite Eq.(7), by multiplying with the conjugate in the numerator and the denominator, which gives us

$$\begin{aligned} t &= \frac{1}{\tau \mp \sqrt{1+\tau^2}}, \\ \rightarrow \;\; t &= \frac{1}{\tau + \sqrt{1+\tau^2}}, \quad \text{for } \tau > 0, \\ \rightarrow \;\; t &= \frac{1}{\tau + \sqrt{1-\tau^2}}, \quad \text{for } \tau < 0, \end{aligned}$$

Next, we define

$$s = tc \quad \text{and} \quad c = \frac{1}{\sqrt{1+t^2}}.$$

We need to find the largest matrix element because, the most efficient way to find a_{kl} is to rotate along the largest non-diagonal matrix element for each rotation. Jacobi's method in code is presented in the appendix.

In order to make sure our code is working, it is wise to implement some unit tests, in order to test the mathematical properties of our algorithm, like verifying the eigenvalues of the matrix before and after Jacobi's method, by use of Armadillo. We require the relative error to be less than 10^{-10}. Another test is to test if the orthogonality of the eigenvectors are preserved. In this instance we require the dot product of the eigenvectors to be less than 10^{-10}.

C Quantum dots in three dimensions, one electron

Next, we want to look at a situation where we assume we have electrons moving in a three-dimensional harmonic oscillator potential. We also assume they repell each other via the static Coulomb interaction and that we have spherical symmetry. We want to find the solution of the radial part of Schroedinger's equation for one electron by solving it numerically,

$$-\frac{\hbar^2}{2m}\left(\frac{1}{r^2}\frac{d}{dr}r^2\frac{d}{dr} - \frac{l(l+1)}{r^2}\right)R(r) + V(r)R(r) = ER(r), \tag{8}$$

where l is the orbital momentum of the electron, which we will set to 0 meaning the electron has no orbital angular momentum, $V(r)$ is the harmonic oscillator potential, which can be expressed as $V(r) = (1/2)kr^2$ where $k = m\omega^2$. ω is the oscillator frequency. E is the energy of the harmonic oscillator in three dimensions, and can be expressed as

$$E_{nl} = \hbar\omega\left(2n + l + \frac{3}{2}\right), \quad n = 0, 1, 2, ... \text{ and } l = 0, 1, 2, .. \tag{9}$$

By setting the boundary conditions to $u(0) = 0$ and $u(\infty) = 0$, introducing a dimensionless variable $\rho = (1/\alpha)r$ where α is a constant with dimension length, we can rewrite Eq.(8) to

$$-\frac{d^2}{d\rho^2}u(\rho) + \rho^2 u(\rho) = \lambda u(\rho). \tag{10}$$

The analytical results for the first four one-electron energies is the eigenvalues $\lambda_1 = 3$, $\lambda_2 = 7$, $\lambda_3 = 11$, $\lambda_4 = 17$. When looking at Eq.(10), we see that

it is mathematically similar to Eq.(2) for the buckling beam problem except that we have an extra potential, which needs to be added to the diagonal element, so that we get $d = 2/h^2 + V(\rho) = 2/h^2 + \rho^2$.

Another factor that needs to be taken into account is the approximation of $\rho_{\max}$. We want to see which $\rho_{\max}$ values gives results closest to the first four analytical eigenvalues. $\rho_{\max}$ is defined as ρ_N, and we still have the step size $h = (\rho_N - \rho_0)/N$, which means that the step length depends on both number of integration points and $\rho_{\max}$. We therefore also want to see how many integration points we need in order to produce eigenvalues with a relative error less than 10^{-4}. We set $N = 350$ and test for different values of $\rho_{\max}$ between 3 and 6, and then look at the relative error between the calculations and the analytical answers.

D Quantum dots in three dimensions, two electrons

We now want to study a case where we have two electrons in a harmonic oscillator well. They also interact via a repulsive Coulomb interaction. The single-electron equation is defined as

$$-\frac{\hbar^2}{2m}\frac{d^2}{dr^2}u(r) + \frac{1}{2}kr^2u(r) = E^{(1)}u(r), \tag{11}$$

where $E^{(1)}$ is the energy where we only have one electron. We then define the Schroedinger's equation for a case where we have two electrons, which has a two-electron wave function $u(r_1, r_2)$ and energy $E^{(2)}$,

$$\left(-\frac{\hbar^2}{2m}\frac{d^2}{dr_1^2} - \frac{\hbar^2}{2m}\frac{d^2}{dr_2^2} + \frac{1}{2}kr_1^2 + \frac{1}{2}kr_2^2\right)u(r_1, r_2) = E^{(2)}u(r_1, r_2). \tag{12}$$

This equation can also be rewritten and simplified, by introducing the new coordinates $r = r_1 - r_2$, the center-of-mass coordinate $R = 1/2(r_1 + r2_)$, and by using the ansatz $u(r, R) = \psi(r)\phi(R)$ we can separate the equations for r and R, and the energy is given as the sum of the E_r and E_R. We obtain

$$-\frac{d^2}{d\rho^2}\psi(\rho) + \omega_r^2\rho^2\psi(\rho) + \frac{1}{\rho} = \lambda\psi(\rho), \tag{13}$$

where ω_r reflects the strength of the oscillator potential. Again, we see that this expression is similar to both Eq.(2) and Eq.(10), but we now need to add another potential to the diagonal element, so we get that $d = 2/h^2 + \omega_r^2\rho^2 + 1/\rho$. We still say that $l = 0$, and we will look at the cases where $\omega_r = 0.01, 0.5, 1, 5$, for the ground state only.

III Results

When we are using $\rho \in [0, 1]$ and a tolerance $\epsilon = 10^{-10}$, the number of similarity transformations needed before we reach a result where all non-diagonal matrix elements are essentially zero, for the buckling beam problem, are presented in Table 1 and visualized in Figure 1.

Table 1: The number of iterations, N, needed before we reach a result where all non-diagonal matrix elements are essentially zero.

N	10	50	100	150
Number of iterations	152	4574	18568	42306

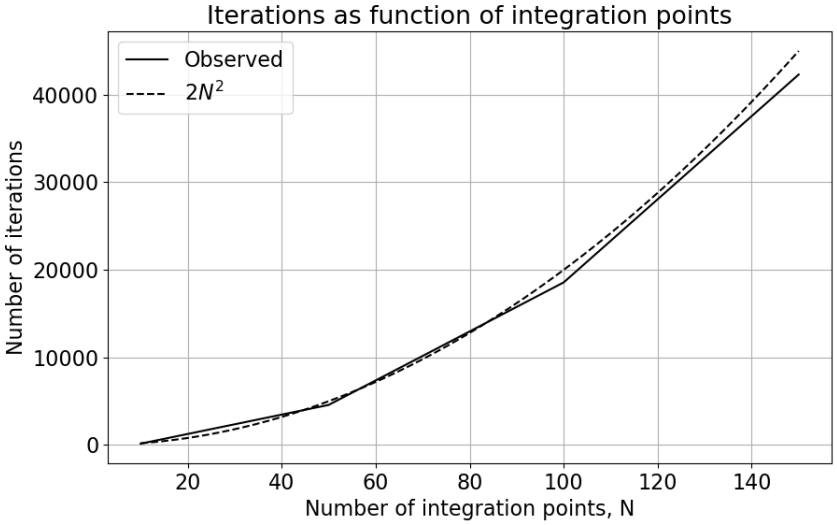

Figure 1: Figure visualizing the number of iterations, N, needed before we reach a result where all non-diagonal matrix elements are essentially zero. The dashed line represents $2N^2$.

All numerical calculations has passed the unit test implemented in the code. The orthogonality of the eigenvectors are preserved, and the relative error of the eigenvalues is within a tolerance of $\epsilon = 10^{-10}$ when comparing them from before and after the transformations, in other words the eigenvalues are conserved trough the transformations.

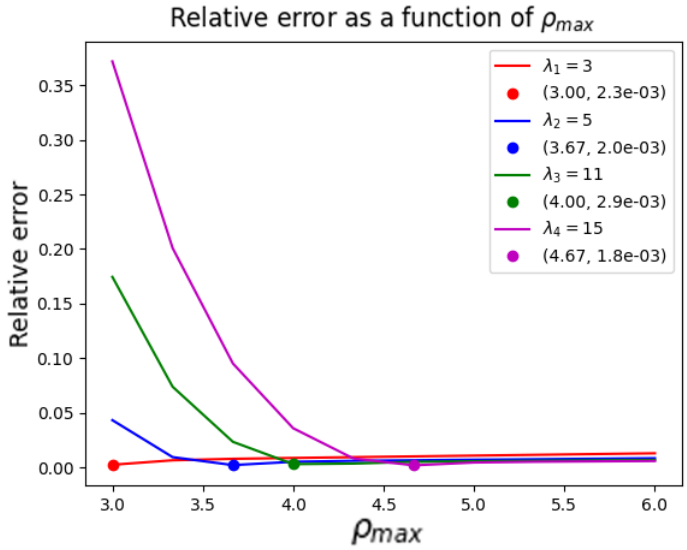

Figure 2: Relative error as a function of $\rho_{\max}$, in the case with one electron. The red line represents $\rho_{\max}$ for $\lambda_1 = 3$, the blue line correspond to $\lambda_2 = 5$, the green line to $\lambda_3 = 11$ and the purple line correspond to $\lambda_4 = 15$.

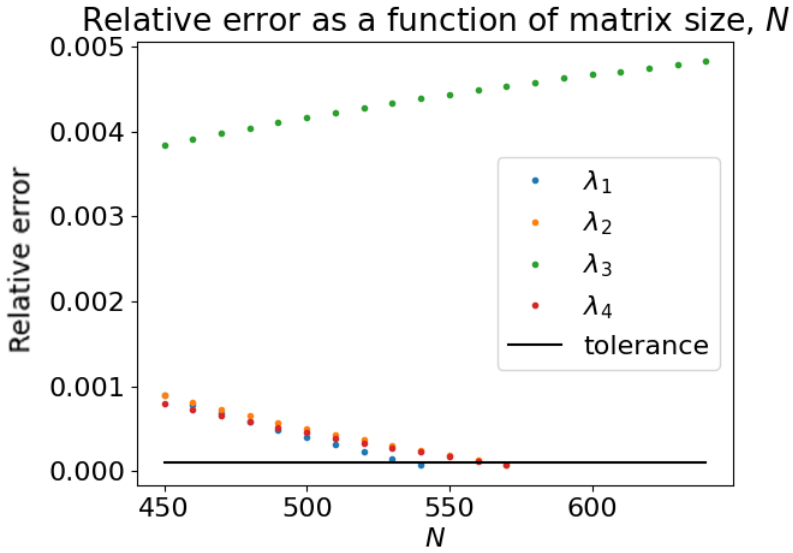

Figure 3: Relative error as a function of matrix size, N, in the case with one electron. The blue dots represents the relative error for $\lambda_1 = 3$, the orange dots correspond to $\lambda_2 = 5$, the green dots to $\lambda_3 = 11$ and the red dots correspond to $\lambda_4 = 15$.

When the harmonic oscillator potential is added to the tridiagonal ma-

trix A, and we know that the actual eigenvalues are $\lambda = 3, 7, 11$ and 15, we observe the optimal endpoint $\rho_{\max}$ for the four different eigenvalues λ, in the case with one electron, presented in Figure 2. We find that the optimal $\rho_{\max}$ values for eigenvalue 1 trough 4 are respectively 3.00, 3.67, 4.00, 4.67.

In Figure 3, we observe the number of integration points we need in order to produce eigenvalues with relative error less than 10^{-4} compared to the analytic solutions of $\lambda = 3, 7, 11$ and 15. From the plot we read that the relative error corresponding to λ_1, λ_2, λ_4 passes beneath the tolerance value around $N \approx 550$.

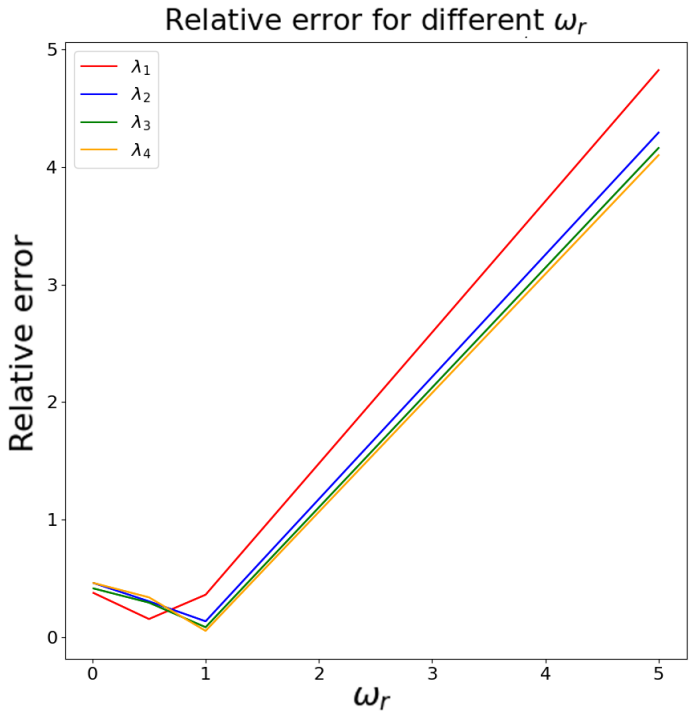

Figure 4: Relative error of the eigenvalues $\lambda_1 = 3$, $\lambda_2 = 5$, $\lambda_3 = 11$ and $\lambda_4 = 15$, as a function of the strength of the oscillator potential represented by ω_r, in the case with two electrons. $\omega_r = 0.01, 0.5, 1$ and 5.

In Figure 4, the relative error of the eigenvalues as a function of the strength of the oscillator potential represented by ω_r is visualized, in the case with two electrons, where $\omega_r = 0.01, 0.5, 1$ and 5.

IV Discussion and conclusion

When looking at Table 1 and Figure 1, where the number of iterations needed before we reach a result where all non-diagonal matrix elements are essentially zero, for the buckling beam problem, are presented, we observe that the number of iterations increases rapidly with the size of the matrix N. We observe that the number of iterations goes as approximately $2N^2$, which is consistent with what we would expect as the size of the matrix goes as N^2.

The unit test passed for all the choices of parameters presented in this report. This indicates that the algorithms are stable in the parameter space evaluated. An alternative unit test for the eigenvalues could be to compare the computed values to the analytic ones, as the buckling beam has analytic solutions. This could increase the accuracy of the expected values.

In the case where we look at one electron, we find values for $\rho_{\max}$, presented in Figure 2, however the resolution is quite poor. This is a significant source of error in the following analyses, as they make use of these values. This might be the reason for the observed behaviour of λ_3 in Figure 3. We would expect this to also decrease with N. A possible improvement of the analysis would be to increase the solution of $\rho_{\max}$.

Moving on to the case where we look at two electrons, we observe in Figure 4 that Jacobi's method is best suited for evaluating potentials in the case where $\omega_r \leq 1$. An area of improvement is to increase the resolution of ω_r between 1 and 5.

In conclusion, the iterations of similarity transforms in Jacobi's method for the buckling beam problem goes as $2N^2$. We find that Jacobi's method is most accurate for $\rho_{\max} = 3, 3, 67, 4, 4, 67$, which in turn yields an optimal $N \approx 550$. To improve the results in this report we could increase resolution of the different parameters. In addition we suggest evaluating for a mesh grid with the different parameters to account for mutual dependencies.

V Appendix

Algorithm 1 Jacobi's method

while max element $> \epsilon$ **do**
  for $i = 1, 2, 3, ..., n$ **do**
    for $j = i + 1, i + 2, ..., n$ **do**
      $a_{ij} = |A_{ij}|$
      if $a_{ij} >$ max element **then**
        max element $= a_{ij}$, $k = i$, $l = j$
      end if
    end for
  end for

  if $A_{kl} \neq 0$ **then**
    $\tau = (A_{ll} - A_{kk})/(2A_{kl})$
    if $\tau \geq 0$ **then**
      $t = 1/(\tau + \sqrt{1 + \tau^2})$
    else
      $t = -1/(-\tau + \sqrt{1 + \tau^2})$
    end if
    $c = 1/\sqrt{1 + t^2}$
    $s = ct$
  else
    $c = 1$
    $s = 0$
  end if

  $a_{kk} = A_{kk}; \quad a_{ll} = A_{ll}$
  $A_{kk} = c^2 a_{kk} - 2csA_{kl} + s^2 a_{ll}$
  $A_{ll} = s^2 a_{kk} + 2csA_{kl} + c^2 a_{ll}$
  $A_{kl} = A_{lk} = 0$

  for $i = 1, 2, 3, ..., n$ **do**
    if $i \neq k \wedge i \neq l$ **then**
      $a_{ik} = A_{ik}; \quad a_{il} = A_{il}$
      $A_{ik} = ca_{ik} - sa_{il}$
      $A_{il} = ca_{il} + sa_{ik}$
      $A_{li} = A_{il}; \quad A_{ki} = A_{ik}$
    end if
  end for
end while

$\lambda_i = A_{ii}$