Revista Brasileira de Ensino de Física
https://doi.org/10.1590/1806-9126-RBEF-2025-0043

Material Suplementar para "Moléculas formadas por átomos hidrogenóides como poços de potencial finito: solução numérica da equação de Schröndiger e proposta de extensão didática via modelos mentais"

Apêndice A - Programa em FORTRAN usado para gerar as funções de onda das moléculas diatômica relacionadas nas tabelas (1) e (2).

```
      program schroed1d
!     .. Scalar Arguments ..
!     Author David L. Azevedo
!     Date 21/07/2024
!     Please, do not remove the comments above.
!     If you use this code, please cite this paper.
      CHARACTER*1 JOBZ, ajobz
      Parameter (N=5000)
      real*8 D(N), E(N-1), Work(2*N-1), Z(N,N)
      real*8 V(N)
      real*8 x1,x2,dx,x,y
      real*8 la, Va, a, dela, lt
      integer i,j,nla,nva
      Jobz='N'
      LDZ=N
!     box-well, finite well
!     eigenvalues, all eigenvalues
!     eigenvector, first eigenvector
!     this code can be easily modified to write all the eigenvectors

      open(10,file='box-well.dat')
      open(20,file='eigenvalues-well.dat')
      open(40,file='eigenvector-well.dat')

! reset all matrix to zero
         do i=1, N
            D(i)=0.0d0
         end do
            do i=1, N-1
               E(i)=0.0d0
            end do

         do i=1, 2*N-1
            Work(i)=0.0d0
         end do

         do i=1, N
         do j=1, N
            Z(i,j)=0.0d0
         end do
         end do
! reset all matrix to zero
! integration interval [-20,20]
         x1=-20.0d0
         x2= 20.0d0
         lt=1.0d0
         a=0.0d0
! input test
         Va=-0.84797
         la=2.80057

         write(6,*)'Input V (potential depth):'
!        read(5,*)Va
```

```
          write(6,*)'Input L(width of well)'
!         read(5,*)la
          write(6,*)'The eigenvalues output should be at: eigenvalues-well.dat '
          write(6,*)'The eigenvector output should be at: eigenvector-well.dat '
          write(6,*)'write the first eigenvector(Y/N)? '
          Jobz='V'
          read(5,*)ajobz
          if (ajobz.eq.'Y') then
             jobz='V'
          write(6,*)'all vectors will be calculated'
          write(6,*)'obtaining all vectors could be very time consuming!'
          else
            jobz='N'
          end if

! input test
!         Va=-0.84797
!         la=2.80057
!         Jobz='V'

          dx=(x2-x1)/N
          write(6,*)'Parameters run in a.u.'
          write(6,25)'Depth. V','L(width)  ','x1   ','x2   ', 'N   ', 'Int. step'
          write(6,10)Va,la,x1,x2,N, dx
          write(20,*)'Parameters run in a.u.'
          write(20,25)'Depth. V','L(width)  ','x1   ','x2   ', 'N  ', 'Int. step'
          write(20,10)Va,la,x1,x2,N, dx
          write(20,25)'1st value','2nd value',' 2nd - 1st ','INFO'
! write finite-well points
          x=0.0d0
          do i=1,N
             x=x1+dx*(i-1)
             if((x.ge.-la/2.0d0).and.(x.le.la/2.0d0)) then
               y=Va
               else
               y=0.0d0
             end if
             write(10,40)x,y
          end do
          x=0.0d0
          do i=1, N
             x=x1+dx*(i-1)
!             if((x.lt.0.0d0).or.(x.gt.la)) then
             if((x.ge.-la/2.0d0).and.(x.le.la/2.0d0)) then
                   D(i)=Va+1.0d0/(dx**2.0d0)
                else
                  D(i)=0.0d0+1.0d0/(dx**2.0d0)
                end if
          end do
          a = dble(-1.0d0/(2.0d0*(dx**2.0d0)))
          do i=1, N-1
             E(i)=a
          end do
!         write(6,10)D
!         write(6,10)E

          call DSTEV( JOBZ, N, D, E, Z, LDZ, WORK, INFO )
          write(6,*)'Solutions in a.u. '
          write(6,25)'1st value','2nd value','2nd - 1st','INFO'
          write(6,11)D(1),D(2),D(2)-D(1),INFO
          write (6,*)'First eigenvalues'
          write (6,10)(D(k1),k1=1,4)
          write(20,11)D(1),D(2),D(2)-D(1),INFO
```

```
          if(jobz.eq.'V')then
          write (6,*)'Writing first eigenvector to: eigenvector-well.dat'
! choosing a different i, you could obtain the i-th eigenvector
!          do i=1,1
             write(40,40)(x1+dx*(j-1),Z(j,1),j=1,N)
!          end do
! if you run i from 1 to N, you could write all eigenvectors
          end if

 10   format(4F14.6,I5,1F14.6 )
 11   format(3F14.6,I5,1F14.6 )
 40   format(2F14.6)
 25   format(4A14,A6,1A14)

          stop
          end
```