Advanced Computational Quantum Mechanics

– Computational Lectures –

Liam Scarlett

Curtin University

Last updated: April 21, 2026

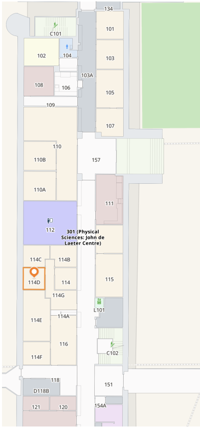
Welcome

Contact:

Slack: Group discussion/assignment help.
`Click here to join.`

email: `liam.scarlett@curtin.edu.au`

office: 301.114D



Unit plan:

Lecture 1: Atomic structure: application to H

Lecture 1(b): Molecular structure: application to H_2^+ (optional)

Lecture 2: Numerov's method: quantum harmonic oscillator and continuum waves

Lecture 3: Momentum-space potential scattering

Lecture 4: Electron-hydrogen V-matrix elements

Assessments: One assignment per lecture, equally weighted.
Due before the start of the next lecture.

1. Atomic structure

Lecture outcomes

- Code the hydrogen electronic Hamiltonian
- Solve the eigenvalue problem
- Demonstrate convergence in the obtained energies
- Code the Laguerre basis functions
- Recover the radial wave functions and compare with analytical solutions

Hydrogen wave functions

- Wave functions can be written as:

$$\Phi = \Phi_{n\ell m}(\boldsymbol{r}) = \frac{1}{r}\bar{\Phi}_n^{\ell}(r)Y_\ell^m(\hat{\boldsymbol{r}}) \tag{1}$$

  - Characterised by quantum numbers $(n,\ \ell,\ m)$
  - Spherical symmetry allows separation into radial and angular components
  - The $\frac{1}{r}$ factor is not necessary, but it is useful to cancel the r^2 factor from the volume element when computing matrix elements between two states

- Schrödinger equation:

$$H\Phi_{n\ell m}(\boldsymbol{r}) = E_n\Phi_{n\ell m}(\boldsymbol{r}) \tag{2}$$

  - Hamiltonian operator: $H = K + V(\boldsymbol{r})$
  - Kinetic-energy operator: $K = -\frac{1}{2}\nabla^2 = -\dfrac{1}{2r}\dfrac{\partial^2}{\partial r^2}(r\cdot\ ) + \dfrac{L^2}{2r^2}$
  - Squared angular momentum operator:

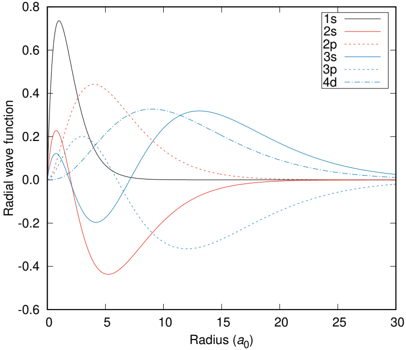
$$L^2 = -\frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial}{\partial\theta} - \frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\phi^2}\right) \tag{3}$$

  - Spherical harmonics Y_ℓ^m are eigenfunctions of L^2 and L_z

$$L^2Y_\ell^m(\hat{\boldsymbol{r}}) = \ell(\ell+1)Y_\ell^m(\hat{\boldsymbol{r}}) \tag{4}$$

$$L_zY_\ell^m(\hat{\boldsymbol{r}}) = mY_\ell^m(\hat{\boldsymbol{r}}) \tag{5}$$

Analytical solutions



n	energy (a.u.)
1	-0.5
2	-0.125
3	$-0.055\bar{5}$
4	-0.03125
n	$-0.5/n^2$

n	ℓ	$\bar{\Phi}_n^\ell$
1	0	$2re^{-r}$
2	0	$\frac{r}{\sqrt{2}}\left(1-\frac{r}{2}\right)e^{-r/2}$
2	1	$\frac{r^2}{\sqrt{24}}e^{-r/2}$
3	0	$\frac{2r}{\sqrt{27}}\left(1-\frac{2r}{3}+\frac{2r^2}{27}\right)e^{-r/3}$
3	1	$\frac{8r^2}{27\sqrt{6}}\left(1-\frac{r}{6}\right)e^{-r/3}$
3	2	$\frac{4r^3}{81\sqrt{30}}e^{-r/3}$

One-electron diagonalisation

- Spherical harmonics Y_ℓ^m form a complete basis on the sphere
- Similarly we choose a set of radial functions $\{\varphi_{k\ell}\}_{k=1}^\infty$ which form a complete basis for the radial L^2 space, then

$$\{|\phi_j\rangle\}_{j=1}^\infty\,, \qquad \langle \boldsymbol{r}|\phi_j\rangle = \frac{1}{r}\varphi_{k_j\ell_j}(r)Y_{\ell_j}^{m_j}(\hat{\boldsymbol{r}}) \tag{6}$$

  forms a complete basis for the 3D L^2 space.
  - We choose an ℓ-dependent radial basis to account for different radial behaviour of different ℓ states.
- We expand the electronic state as

$$|\Phi\rangle = \sum_j c_j|\phi_j\rangle \tag{7}$$

  - Sum over j is implicitely a sum over k, ℓ, m: $\sum_{k=1}^\infty \sum_{\ell=0}^\infty \sum_{m=-\ell}^\ell$
- Substitute into Schrödinger equation:

$$\sum_j c_j H|\phi_j\rangle = E\sum_j c_j|\phi_j\rangle \tag{8}$$

$$\sum_j c_j\langle\phi_i|H|\phi_j\rangle = E\sum_j c_j\langle\phi_i|\phi_j\rangle \tag{9}$$

One-electron diagonalisation

$$\sum_j c_j \langle \phi_i | H | \phi_j \rangle = E \sum_j c_j \langle \phi_i | \phi_j \rangle \tag{10}$$

- If we allow i to vary over same range as j this becomes a matrix equation:

$$\sum_j H_{ij} c_{ji} = E_i \sum_j B_{ij} c_{ji} \tag{11}$$

  - Hamiltonian matrix $H_{ij} \equiv \langle \phi_i | H | \phi_j \rangle$
  - Overlap matrix $B_{ij} \equiv \langle \phi_i | \phi_j \rangle$
  - Each value of i corresponds to a different solution (state)
- For an orthonormal basis, $B_{ij} = \delta_{ij}$ and we solve the standard eigenvalue problem

$$H\boldsymbol{c}^i = E_i \boldsymbol{c}^i \qquad (\boldsymbol{c}^i = i\text{th column of } c \text{ matrix}) \tag{12}$$

- For a general non-orthogonal basis we solve the *generalised* eigenvalue problem

$$H\boldsymbol{c}^i = E_i B \boldsymbol{c}^i \tag{13}$$

One-electron diagonalisation

- To perform the calculation we must truncate the basis to some finite number N of functions, and solve

$$\sum_{j=1}^{N} H_{ij} c_{ji}^{N} = E_i^N \sum_{j=1}^{N} B_{ij} c_{ji}^{N}, \qquad |\Phi_i^N\rangle = \sum_{j=1}^{N} c_{ji}^{N} |\phi_j\rangle \tag{14}$$

- The solutions $|\Phi_i^N\rangle$ are *not* eigenstates of the Hamiltonian operator, so we call them *pseudostates*
- Pseudostates satisfy $\langle \Phi_i^N | H | \Phi_j^N \rangle = \epsilon_i \delta_{ij}$ (they diagonalise the Hamiltonian), but $H|\Phi_i^N\rangle \neq \epsilon_i |\Phi_i^N\rangle$.
- The completeness of the basis guarantees that for bound states:

$$\lim_{N\to\infty} |\Phi_i^N\rangle = |\Phi_i\rangle \tag{15}$$

$$\lim_{N\to\infty} E_i^N = E_i \tag{16}$$

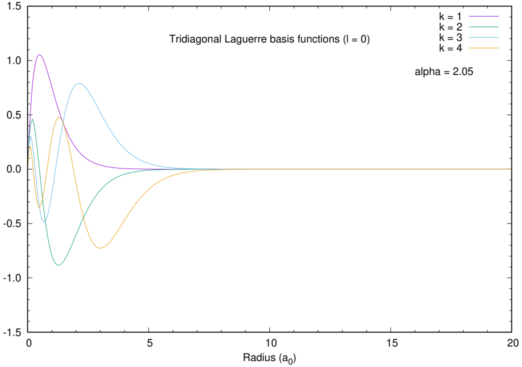
and hence we can obtain convergence to arbitrary accuracy by increasing N

Laguerre basis functions

- We will utilise the following non-orthogonal radial basis:

$$\varphi_{k\ell}(r) = \sqrt{\frac{\alpha_\ell (k-1)!}{(k+\ell)(k+2\ell)!}} (2\alpha_\ell r)^{\ell+1} e^{-\alpha_\ell r} L_{k-1}^{2\ell+1}(2\alpha_\ell r) \tag{17}$$

  - α_ℓ is a scaling factor
  - $L_{k-1}^{2\ell+1}$ are the generalised Laguerre polynomials
  - Note that Igor's lecture notes use $L_{k-1}^{2\ell+2}$ which creates an orthogonal basis



Laguerre basis functions

- This basis is tridiagonal (i.e. the overlap matrix is tridiagonal) and has the following useful analytical matrix elements:

$$\langle \varphi_{k'\ell'} | \varphi_{k\ell} \rangle = \delta_{\ell'\ell} \cdot \begin{cases} 1, & k = k' \\ 0, & k > k' + 1 \\ -\frac{1}{2}\sqrt{1 - \dfrac{\ell(\ell+1)}{(k'+\ell)(k'+\ell+1)}}, & k = k' + 1 \end{cases} \tag{18}$$

$$\langle \varphi_{k\ell} | \varphi_{k'\ell'} \rangle = \langle \varphi_{k'\ell'} | \varphi_{k\ell} \rangle \tag{19}$$

$$\langle \varphi_{k'\ell'} | \frac{1}{r} | \varphi_{k\ell} \rangle = \frac{\alpha_\ell}{(k+\ell)} \delta_{k'k} \delta_{\ell'\ell} \tag{20}$$

- Eq. (18) is the statement of tridiagonality, Eq. (19) says that the overlap matrix is symmetric (superdiagonal equal to subdiagonal), and Eq. (20) is the matrix element of the Coulomb potential (will be useful later).
- If you take the 1s basis function ($k = 1$, $\ell = 0$), with $\alpha_0 = 1$, you get the Hydrogen 1s radial wave function:

$$\phi_{10}(r; \alpha_0 = 1) = 2re^{-r} \tag{21}$$

Laguerre basis functions

- The 3D basis then has:

$$\langle \phi_i | \phi_j \rangle = \langle \varphi_{k_i \ell_i} | \varphi_{k_j \ell_i} \rangle \delta_{\ell_i \ell_j} \delta_{m_i m_j} \tag{22}$$

$$\langle \phi_i | K | \phi_j \rangle = \alpha_{\ell_i}^2 \delta_{ij} - \frac{\alpha_{\ell_i}^2}{2} \langle \varphi_{k_i \ell_i} | \varphi_{k_j \ell_i} \rangle \delta_{\ell_i \ell_j} \delta_{m_i m_j} \tag{23}$$

$$\langle \phi_i | \frac{1}{r} | \phi_j \rangle = \frac{\alpha_{\ell_i}}{(k + \ell_i)} \delta_{ij} \tag{24}$$

- These can all be derived using the following properties of the Laguerre polynomials:

$$L_n^t(x) = L_n^{t+1}(x) - L_{n-1}^{t+1}(x) \tag{25}$$

$$\int\limits_0^\infty x^t e^{-x} L_n^t(x) L_m^t(x) \, \mathrm{d}x = \frac{\Gamma(n+t+1)}{\Gamma(n+1)} \delta_{nm} \tag{26}$$

$$x \frac{\mathrm{d}^2}{\mathrm{d}x^2} L_n^t(x) + (t + 1 - x) \frac{\mathrm{d}}{\mathrm{d}x} L_n^t(x) = -n L_n^t(x) \tag{27}$$

H one-electron V-matrix elements

- In general, if $V(\boldsymbol{r})$ is spherically symmetric:

$$V(\boldsymbol{r}) = V(r) \tag{28}$$

$$\langle \phi_i | V | \phi_j \rangle = \langle \varphi_{k_i \ell_i} | V | \varphi_{k_j \ell_i} \rangle \delta_{\ell_i \ell_j} \delta_{m_i m_j} \tag{29}$$

- For the case of one-electron atoms we have

$$V(r) = -\frac{Z}{r} \tag{30}$$

$$\langle \phi_i | V | \phi_j \rangle = -\frac{Z \alpha_{\ell_i}}{(k + \ell_i)} \delta_{ij} \tag{31}$$

- The Hamiltonian matrix is then

$$\langle \phi_i | H | \phi_j \rangle = \alpha_{\ell_i}^2 \delta_{ij} - \frac{Z \alpha_{\ell_i}}{(k_i + \ell_i)} \delta_{ij} - \frac{\alpha_{\ell_i}^2}{2} \langle \varphi_{k_i \ell_i} | \varphi_{k_j \ell_i} \rangle \delta_{\ell_i \ell_j} \delta_{m_i m_j} \tag{32}$$

  - Note how the Hamiltonian matrix is zero whenever $(\ell_i, m_i) \neq (\ell_j, m_j)$

H one-electron V-matrix elements

- General rule of quantum mechanics: if an operator commutes with the Hamiltonian the corresponding observable is a constant of motion
- It is easy to show that $[K, L^2] = [K, L_z] = 0$
- For the spherically-symmetric case we trivially have $[V, L^2] = [V, L_z] = 0$
- Putting these together we have:

$$[H, L^2] = [H, L_z] = 0 \tag{33}$$

  i.e. ℓ and m are both constants of motion

- This allows the structure calculation to be performed separately for each fixed value of ℓ and m:

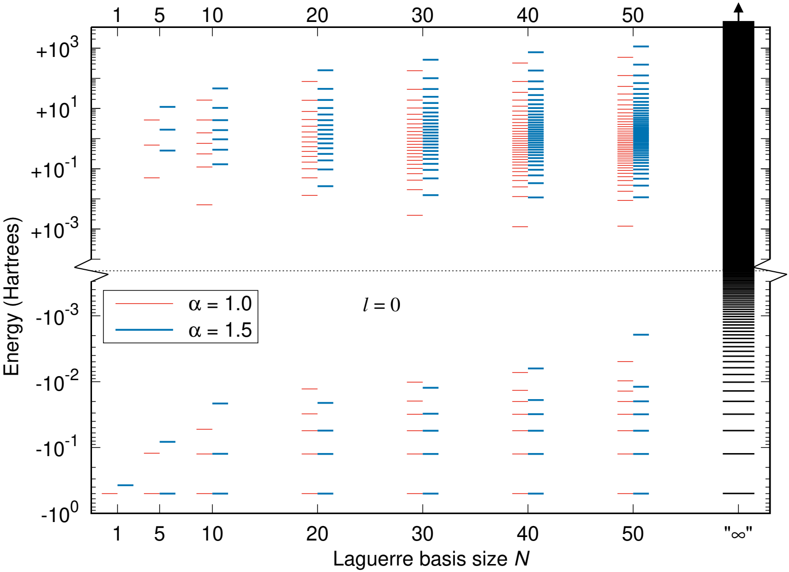
$$|\Phi_{i\ell m}\rangle = \sum_j c^{\ell}_{ji} |\phi^{\ell m}_j\rangle \tag{34}$$

$$\phi^{\ell m}_j(\boldsymbol{r}) = \frac{1}{r}\varphi_{j\ell}(r) Y^m_\ell(\hat{\boldsymbol{r}}) \tag{35}$$

  or for the radial wave function:

$$|\bar{\Phi}^{\ell}_i\rangle = \sum_j c^{\ell}_{ji} |\varphi_{j\ell}\rangle \tag{36}$$

Hydrogen energies in a Laguerre basis



Outline for calculating energies

- Read N, α, and ℓ from user
- Define $N \times N$ arrays `H` and `B` to store the Hamiltonian and overlap matrices
- Define an $N \times 1$ array `w` to store obtained energies
- Define an $N \times N$ array `z` to store expansion coefficients
- Calculate the overlap matrix. Since it is tridiagonal it is easiest to use a single `do` loop:

```
B = 0.0d0
do i=1, N-1
  B(i,i) = 1.0d0
  B(i,i+1) = ???
  B(i+1,i) = B(i,i+1)
enddo
B(N,N) = 1.0d0
```

- Calculate the Hamiltonian matrix in a similar way, making use of the overlap matrix elements you already have
- Call the `rsg` subroutine:

```
call rsg(N,N,H,B,w,1,z,ier)
```

- Save the energies from the `w` array to file in a way which will be convenient for plotting (e.g. first column = n, second column = $E(n)$)

Use the **rsg** Fortran subroutine (file **rsg.f**) to solve the matrix equations.

```
      subroutine rsg(nm,n,a,b,w,matz,z,ierr)
      integer n,nm,ierr,matz
      real*8 a(nm,n),b(nm,n),w(n),z(nm,n),fv1(n),fv2(n)
!
!     this subroutine calls the recommended sequence of
!     subroutines from the eigensystem subroutine package (eispack)
!     to find the eigenvalues and eigenvectors (if desired)
!     for the real symmetric generalized eigenproblem ax = (lambda)bx.
!
!     on input
!
!        nm  must be set to the row dimension of the two-dimensional
!        array parameters as declared in the calling program dimension statement.
!
!        n  is the order of the matrices a and b.
!
!        a  contains a real symmetric matrix.
!
!        b  contains a positive definite real symmetric matrix.
!
!        matz  is an integer variable set equal to zero if
!        only eigenvalues are desired. otherwise it is set to
!        any non-zero integer for both eigenvalues and eigenvectors.
!
!     on output
!
!        w  contains the eigenvalues in ascending order.
!
!        z  contains the eigenvectors arranged in columns if matz is not zero.
!
!        ierr  is an integer output variable set equal to an error
!           completion code. the normal completion code is zero.
```

Generating the basis functions

- The Laguerre polynomials can be obtained using the recurrence relation

$$L_{n+1}^{t}(x) = \frac{(2n+1+t-x)L_{n}^{t}(x) - (n+t)L_{n-1}^{t}(x)}{n+1} \tag{37}$$

with

$$L_{0}^{t}(x) = 1 \tag{38}$$

$$L_{1}^{t}(x) = 1 + t - x. \tag{39}$$

- If we write the radial basis functions as

$$\varphi_{k\ell}(r) = \sqrt{\frac{\alpha_{\ell}(k-1)!}{(k+\ell)(k+2\ell)!}}\,\tilde{\varphi}_{k\ell}(r) \tag{40}$$

then $\tilde{\varphi}_{k\ell}(r)$ satisfes the recurrence relation:

$$\tilde{\varphi}_{k\ell}(r) = \frac{2(k-1+\ell-\alpha_{\ell}r)\tilde{\varphi}_{k-1,\ell}(r) - (k+2\ell-1)\tilde{\varphi}_{k-2}(r)}{k-1} \tag{41}$$

with

$$\tilde{\varphi}_{1\ell}(r) = (2\alpha_{\ell}r)^{\ell+1}e^{-\alpha_{\ell}r} \tag{42}$$

$$\tilde{\varphi}_{2\ell}(r) = 2(\ell+1-\alpha_{\ell}r)(2\alpha_{\ell}r)^{\ell+1}e^{-\alpha_{\ell}r}. \tag{43}$$

Outline for generating basis functions

- Define a real variable **`dr`** and an integer **`nr`** to store the grid spacing and number of grid points, respectively
- Define a real variable **`rmax`** to store the maximum r in the grid
- Read **`rmax`** and **`dr`** from the user, and calculate **`nr`**
- Define an **`nr`** $\times$ 1 array **`rgrid`** and populate it with the radial grid values from 0 to **`rmax`**
- Define an **`nr`** $\times$ `N` array **`basis`** to store the basis functions (each column is a function on the radial grid)
- Hard code the first two $\tilde{\varphi}$ functions and store in the first two columns of **`basis`**. Make use of Fortran's array-slice notation to do this easily without looping over the radial grid:

```
    basis(:,1) = (2.0d0*alpha*rgrid(:))**(l+1)*exp(-alpha*rgrid(:))
```

- Loop over columns 3 to N of the **`basis`** array and use the recurrence relation for $\tilde{\varphi}$ on the previous slide to calculate the remaining functions. Array-slice notation will make your life easier here too.
- Finally, loop over all columns of the **`basis`** array and multiply by the appropriate normalisation constant to obtain φ.
- Print the basis functions to file for plotting – first column = **`rgrid`**, subsequent columns = each basis function

Outline for recovering wave functions

- The hard work is all done now and the radial wave functions can be easily recovered using

$$\bar{\Phi}_i(r) = \sum_{j=1}^{N} c_{ji}\varphi_j(r) \tag{44}$$

- Define an `nr` × N array `wf` to store the wave functions (same layout as `basis` array)
- Set the entire `wf` array to zero as we will be building it up as a sum
- Loop over the states and evalaute each one using array-slice notation and an additional loop over each basis function

1(b). Molecular structure (optional)

Lecture outcomes

- Calculate H_2^+ Hamiltonian matrix
- Obtain H_2^+ potential-energy curves
- Calculate nuclear vibration Hamiltonian matrix
- Generate vibrational wave functions

Molecular wave functions

- Molecular wavefunctions must describe electronic, vibrational, rotational motion:

$$\Phi = \Phi_{nvJ}(\boldsymbol{r}_1, \boldsymbol{r}_2, \ldots, \boldsymbol{R}_1, \boldsymbol{R}_2, \ldots) \tag{45}$$

  - n: electronic state
  - v: vibrational state
  - J rotational state
  - $\boldsymbol{r}$: electronic coordinates
  - $\boldsymbol{R}$: nuclear coordinates
- Molecular Hamiltonian contains KE operators for every particle, and PE terms for interaction between all particles:

$$H = -\frac{1}{2}\sum_i \nabla^2_{\boldsymbol{r}_i} - \frac{1}{2}\sum_\alpha \frac{\nabla^2_{\boldsymbol{R}_\alpha}}{M_\alpha} + \sum_{\substack{i,j \\ i>j}} V_{ij}(\boldsymbol{r}_i, \boldsymbol{r}_j) + \sum_{\substack{\alpha,\beta \\ \alpha>\beta}} V_{\alpha\beta}(\boldsymbol{R}_\alpha, \boldsymbol{R}_\beta) + \sum_{i,\alpha} V_{i\alpha}(\boldsymbol{r}_i, \boldsymbol{R}_\alpha) \tag{46}$$

  - $\sum_i$: sum over all electrons
  - $\sum_\alpha$: sum over all nuclei
  - V_{ij}: electron-electron potential
  - $V_{\alpha\beta}$: nuclei-nuclei potential
  - $V_{i\alpha}$: electron-nuclei potential
  - M_α: mass of nuclei α

Born-Oppenheimer approximation

- Molecular Hamiltonian can be written as sum of electronic and nuclear Hamiltonians:

$$H = H_{\text{elec}} + H_{\text{nucl}} \tag{47}$$

$$H_{\text{elec}} = -\frac{1}{2}\sum_{i} \nabla^2_{\boldsymbol{r}_i} + \sum_{\substack{i,j\\ i\neq j}} V_{ij}(\boldsymbol{r}_i, \boldsymbol{r}_j) + \sum_{i,\alpha} V_{i\alpha}(\boldsymbol{r}_i, \boldsymbol{R}_\alpha) + \sum_{\substack{\alpha,\beta\\ \alpha\neq\beta}} V_{\alpha\beta}(\boldsymbol{R}_\alpha, \boldsymbol{R}_\beta) \tag{48}$$

$$H_{\text{nucl}} = -\frac{1}{2}\sum_{\alpha} \frac{1}{M_\alpha}\nabla^2_{\boldsymbol{R}_\alpha} \tag{49}$$

- Born-Oppenheimer approximation:

$$\Phi_{nvJ}(\boldsymbol{r}_1, \ldots, \boldsymbol{R}_1, \ldots) = \Phi_n(\boldsymbol{r}_1, \ldots; \boldsymbol{R}_1, \ldots)\Phi^n_{vJ}(\boldsymbol{R}_1, \ldots) \tag{50}$$

where

$$H_{\text{elec}}\Phi_n = \epsilon_n \Phi_n \tag{51}$$

Born-Oppenheimer approximation

- So what is the effect of the total molecular Hamiltonian on the Born-Oppenheimer wave function?

$$\begin{aligned} H\Phi_{nvJ} &= (H_{\text{elec}} + H_{\text{nucl}})\Phi_n \Phi^n_{vJ} && (52) \\ &= (H_{\text{nucl}} + \epsilon_n(\boldsymbol{R}_1, \ldots))\Phi_n \Phi^n_{vJ} && (53) \end{aligned}$$

- Since Φ_{nvJ} must satisfy the molecular Schrödinger equation:

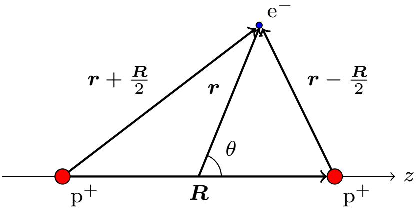
$$H\Phi_{nvJ} = \varepsilon_{nvJ}\Phi_{nvJ}, \qquad \varepsilon_{nvJ} \text{ total molecular energy} \tag{54}$$

  we get an equation for the nuclear wave function:

$$(H_{\text{nucl}} + \epsilon_n(\boldsymbol{R}_1, \ldots))\Phi^n_{vJ} = \varepsilon_{nvJ}\Phi^n_{vJ} \tag{55}$$

- So we obtain Φ_{nvJ} in two steps:
  1. Solve electronic SE for many different nuclear geometries to obtain potential-energy surface $\epsilon_n(\boldsymbol{R}_1, \ldots)$
  2. Solve nuclear SE

The case for H_2^+



- One electron and two nuclei
- We choose a spherical coordinate system with the z axis aligned with the internuclear axis and the origin at the geometric centre of the nuclei

- In this coordinate system the electronic wavefunction is only a function of the electron coordinate and the *distance* R between the two nuclei:

$$\Phi_n = \Phi_n(\boldsymbol{r}; R) \tag{56}$$

- We can also simplfy the nuclear Hamiltonian:

$$H_{\text{nucl}} = -\frac{1}{2M_1}\nabla^2_{\boldsymbol{R}_1} - \frac{1}{2M_2}\nabla^2_{\boldsymbol{R}_2} = -\frac{1}{2\mu}\nabla^2_{\boldsymbol{R}}, \tag{57}$$

  where $\mu = M_1 M_2/(M_1+M_2)$ is the nuclear reduced mass and $\boldsymbol{R}$ is the vector pointing from one nuclei to the other

H_2^+ electronic structure

- For H_2^+ there are no electron-electron potentials so the we can write the electronic Hamiltonian as:

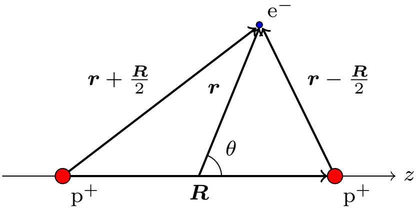
$$H_{\mathrm{elec}} = -\frac{1}{2}\nabla_{\boldsymbol{r}}^2 + V(\boldsymbol{r}) + \frac{1}{R}, \tag{58}$$

where

$$V(\boldsymbol{r}) = -\frac{1}{\left|\boldsymbol{r} + \frac{\boldsymbol{R}}{2}\right|} - \frac{1}{\left|\boldsymbol{r} - \frac{\boldsymbol{R}}{2}\right|} \tag{59}$$

is the electron-nuclei potential and $1/R$ is the nuclear repulsion potential

- Note that since $1/R$ is a scalar, it has no effect on the electronic wave functions and only acts to shift the electronic energies, so we can ignore it for now and add $1/R$ to the electronic energies at the end

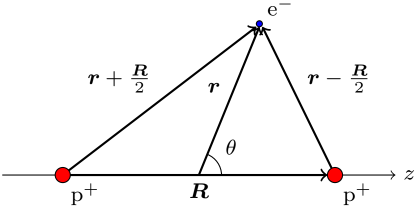


H_2^+ electronic V-matrix elements

- For molecules $V(\boldsymbol{r})$ is no longer spherically symmetric
- The H_2^+ potential

$$V(\boldsymbol{r}) = -\frac{1}{\left|\boldsymbol{r} + \frac{\boldsymbol{R}}{2}\right|} - \frac{1}{\left|\boldsymbol{r} - \frac{\boldsymbol{R}}{2}\right|}$$

can be expanded in spherical harmonics as



$$V(\boldsymbol{r}) = -2 \sum_{\lambda \text{ even}} \sqrt{\frac{4\pi}{2\lambda+1}} \frac{r_<^\lambda}{r_>^{\lambda+1}} Y_\lambda^0(\hat{\boldsymbol{r}})$$

$$r_< \equiv \min(r, R/2)$$

$$r_> \equiv \max(r, R/2)$$

- In practice we choose some maximum $\lambda_{\max}$:

$$V(\boldsymbol{r}) = -2 \sum_{\lambda \text{ even}}^{\lambda_{\max}} \sqrt{\frac{4\pi}{2\lambda+1}} \frac{r_<^\lambda}{r_>^{\lambda+1}} Y_\lambda^0(\hat{\boldsymbol{r}})$$

H_2^+ electronic V-matrix elements

- To calculate the matrix elements of $V(\boldsymbol{r})$ we must evalute

$$\langle\phi_i|\frac{r_<^\lambda}{r_>^{\lambda+1}}Y_\lambda^0(\hat{\boldsymbol{r}})|\phi_j\rangle = \iiint\limits_{\mathbb{R}^3}\frac{1}{r}\varphi_{k_j\ell_j}(r)Y_{\ell_i}^{m_i*}(\hat{\boldsymbol{r}})\frac{r_<^\lambda}{r_>^{\lambda+1}}Y_\lambda^0(\hat{\boldsymbol{r}})\frac{1}{r}\varphi_{k_i\ell_i}(r)Y_{\ell_j}^{m_j}(\hat{\boldsymbol{r}})\,\mathrm{d}\boldsymbol{r}$$

$$= \int\limits_0^\infty \frac{1}{r^2}\varphi_{k_j\ell_j}(r)\frac{r_<^\lambda}{r_>^{\lambda+1}}\varphi_{k_i\ell_i}(r)r^2\,\mathrm{d}r \iint\limits_\Omega Y_{\ell_i}^{m_i*}(\hat{\boldsymbol{r}})Y_\lambda^0(\hat{\boldsymbol{r}})Y_{\ell_j}^{m_j}(\hat{\boldsymbol{r}})\,\mathrm{d}\Omega$$

$$= \int\limits_0^\infty \varphi_{k_j\ell_j}(r)\frac{r_<^\lambda}{r_>^{\lambda+1}}\varphi_{k_i\ell_i}(r)\,\mathrm{d}r\ \langle Y_{\ell_i}^{m_i}|Y_\lambda^0|Y_{\ell_j}^{m_j}\rangle$$

- So the V-matrix element is

$$V_{ij} = -2\delta_{m_im_j}\delta_{\pi_i\pi_j}\sum_{\lambda\text{ even}}^{\lambda_{\max}}\int\limits_0^\infty \varphi_{k_j\ell_j}(r)\frac{r_<^\lambda}{r_>^{\lambda+1}}\varphi_{k_i\ell_i}(r)\,\mathrm{d}r\sqrt{\frac{4\pi}{2\lambda+1}}\langle Y_{\ell_i}^{m_i}|Y_\lambda^0|Y_{\ell_j}^{m_j}\rangle \tag{60}$$

- The $\delta_{\pi_i\pi_j}$ means that the electronic parity $\pi = (-1)^\ell$ is converved, and is a result of selection rules for the angular integral

H_2^+ electronic V-matrix elements

- The `YINT` subroutine in the `wigner.f` file will evalaute

$$\sqrt{\frac{4\pi}{2\lambda+1}}\langle Y_{\ell_i}^{m_i}|Y_\lambda^0|Y_{\ell_j}^{m_j}\rangle$$

```fortran
      function Yint(WJ1,WM1,WLAM,WMU,WJ2,WM2)
      implicit none

      real*8 Yint
      real*8 WJ1, WLAM, WJ2, WM1, WM2, WMU
      real*8 CGC

c     Yint = <J1 M1| sqrt((4.0*PI)/(2d0*lam + 1d0)) Y_{lam mu} | J2 M2>

      Yint = sqrt( (2.*WJ2+1.)/(2.*WJ1+1.) ) *
     >    CGC(WJ2,0d0,WLAM,0d0,WJ1,0d0) *
     >    CGC(WJ2,WM2,WLAM,WMU,WJ1,WM1)

      return
      end
```

Numerical integration

return to lecture 2

- Need an array of numerical integration weights `weights(1:nr)`
- The integral of a function $f(r)$ over the radial grid is then

$$\int_0^{r_{\max}} f(r)\,\mathrm{d}r = \texttt{sum(f(:) * weights(:))} \tag{61}$$

- We can use Simpson's rule:

$$\int f(r)\,\mathrm{d}r \approx [f(r_1)+4f(r_2)+2f(r_3)+4f(r_4)+\cdots+4f(r_{\texttt{nr}-1})+f(r_{\texttt{nr}})]\cdot\frac{\texttt{dr}}{3} \tag{62}$$

- Number of grid points must be odd - `if (mod(nr,2) == 0) nr = nr + 1`
- Simpson weights can be easily obtained:

```
weights(1) = 1.0d0
do i=2, nr-1
  weights(i) = 2.0d0 + 2.0d0*mod(i+1,2)
enddo
weights(nr) = 1.0d0
weights(:) = weights(:) * dr / 3.0d0
```

- The integrand can be expressed as (using the variable `Rn` for R):

$$\varphi_{k_j\ell_j}(r)\frac{r_<^{\lambda}}{r_>^{\lambda+1}}\varphi_{k_i\ell_i}(r) \tag{63}$$

$$= \texttt{basis(:,j) * min(rgrid(:),Rn/2.0d0)**lambda/max(rgrid(:),Rn/2.0d0)**(lambda+1) * basis(:,i)} \tag{64}$$

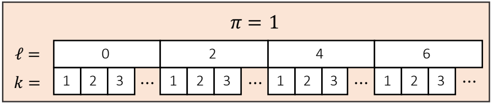
Setting up the basis

- We use the same 3D basis as before:

$$\{|\phi_j\rangle\}_{j=1}^{\infty}\,, \qquad \langle \boldsymbol{r}|\phi_j\rangle = \frac{1}{r}\varphi_{k_j\ell_j}(r)Y_{\ell_j}^{m_j}(\hat{\boldsymbol{r}}) \tag{65}$$

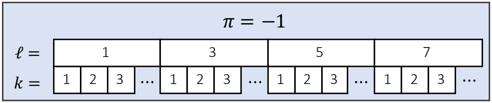
$$\varphi_{k\ell}(r) = \sqrt{\frac{\alpha(k-1)!}{(k+\ell)(k+2\ell)!}}(2\alpha r)^{\ell+1}e^{-\alpha r}L_{k-1}^{2\ell+1}(2\alpha r) \tag{66}$$

- But now rather than setting up the basis per ℓ, we need to include all ℓ (subject to selection rules) per conserved (m, π)
- For now assume $m = 0$, we need to organise a list of basis functions to include for each parity $\pi = \pm 1$:



$\pi = 1$

$\ell =$	0				2				4				6			
$k =$	1	2	3	⋯	1	2	3	⋯	1	2	3	⋯	1	2	3	⋯

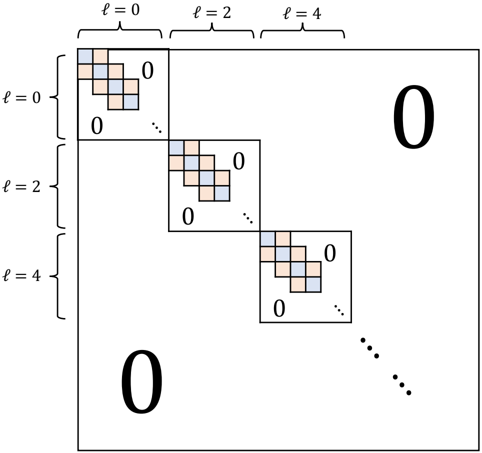


$\pi = -1$

$\ell =$	1				3				5				7			
$k =$	1	2	3	⋯	1	2	3	⋯	1	2	3	⋯	1	2	3	⋯

Setting up the basis

- The overlap matrix is block-tridiagonal:



Setting up the basis

- The value of m in a given symmetry (m, π) also affects set of basis functions because $m \leq \ell$
- To determine the number of basis function for a given parity **`par`** and **`m`**

```
num_func = 0
do l=0, lmax
  if((-1)**l /= par .or. l < m) cycle
  num_func = num_func + N
enddo
```

- To keep track of which functions are in each index of the basis:

```
allocate(k_list(num_func), l_list(num_func))
i=0
do l=0, lmax
  if((-1)**l /= par .or. l < m) cycle
  do k=1, N
    i = i + 1
    k_list(i) = k
    l_list(i) = l
  enddo
enddo
```

Calculating matrix elements

```
allocate(H(num_func,num_func), B(num_func,num_func), V(num_func,num_func))
```

- Evalaluation of the K and B matrices will be similar to the atomic case:

```
do i=1, num_func-1
  B(i,i) = 1.0d0
  l = l_list(i)
  k = k_list(i)
  if(l_list(i+1) /= l) cycle !off-diagonal elements with different l are zero
  B(i,i+1) = ???
  B(i+1,i) = B(i,i+1)
enddo
```

- The V matrix will have non-zero elements everywhere:

```
do i=1, num_func
  do j=1, num_func
    li = l_list(i)
    lj = l_list(j)

    !calculate V-matrix element

  enddo
enddo

H = H + V
```

Obtaining potential-energy curves

- The **rsg** subroutine is called same as last time:

```
      call rsg(num_func,num_func,H,B,w,1,z,ier)
```

- We will obtain **num_func** number of energies in each symmetry, much more than we care about
- Let's save just the lowest energy from each symmetry:

```
     nstates = 2 * (m_max+1) !Number of symmetries

     allocate(energies(nstates)) !allocate space for energies in all symmetries

     nstate = 0 !Index for the energies array

     do m=0, m_max
       do par=-1, 1, 2 !just two parities: -1 and 1

         !Obtain energies for this symmetry here

         nstate = nstate + 1
         energies(nstate) = w(1)

       enddo
     enddo
```

Obtaining potential-energy curves

- The energies will not be in order, so we need to sort them. We don't care about efficiency here so use any method you like but a simple implementation is:

```
logical :: sorted

sorted = .false.
do while(.not.sorted)
  sorted = .true.
  do i=1, nstates-1
    E1 = energies(i)
    E2 = energies(i+1)

    if(E2 < E1) then
      sorted = .false.
      temp = energies(i)
      energies(i) = energies(i+1)
      energies(i+1) = temp
    endif
  enddo
enddo
```

- Then you can print the energies to file

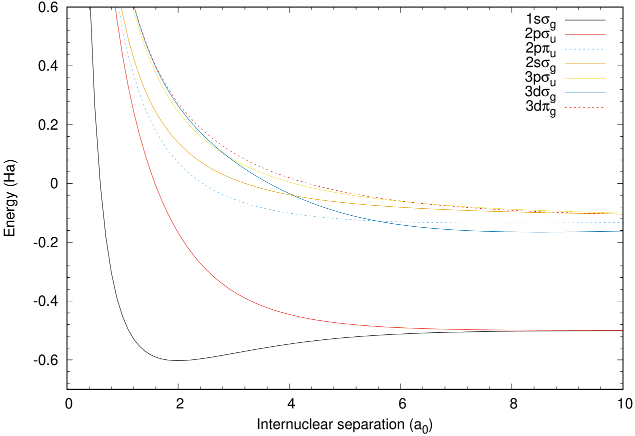
Obtaining potential-energy curves

- To run at a number of different R values you can use a **`bash`** script.
- Your input file will need to read the value of R, so to automate the process put a place holder like `RRRR` in the file, then:

```
#!/bin/bash
ls PEC.* &>/dev/null && rm PEC.* #remove any files from last run
for R in 0.1 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.5 7.0 7.5 8.0
do
  mkdir R=${R}
  sed s/RRRR/${R}/ data.in > R=${R}/data.in
  cd R=${R}
  ../main
  for i in $(seq 1 4)
  do
    echo ${R} $(awk '$1=='${i} E.out) >> ../PEC.${i}
    #E.out is whatever you call the energies output file in your program
    #This will create files PEC.1, PEC.2 etc with PEC(R) for each state
  done
  cd ..
done
```

Obtaining potential-energy curves

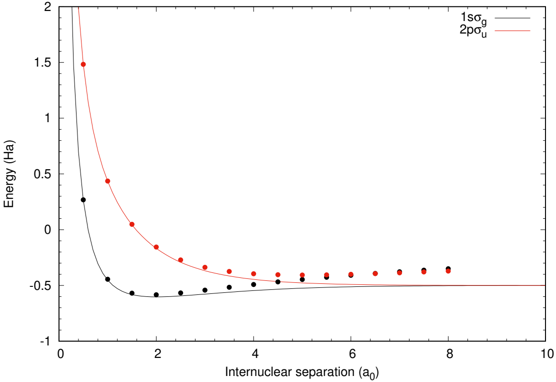
- Accurate potential-energy curves for the first 7 electronic states of H_2^+:



Obtaining potential-energy curves

- Getting accurate energies is more difficult at larger R
- Here is a comparison between the accurate potential-energy curves for the first two H_2^+ states ($1s\sigma_g$ and $2p\sigma_u$) with the results from a calculation with

$$\ell_{\max} = 3, \qquad N = 10, \qquad \alpha = 2.0 \tag{67}$$



Vibrational wave functions

- Following from Eq. (55) the Schrödinger equation for the nuclear wave function is

$$\left[-\frac{1}{2\mu}\nabla^2_{\boldsymbol{R}} + \epsilon_n(R)\right]\Phi^n_{vJ}(\boldsymbol{R}) = \varepsilon_{nvJ}\Phi^n_{vJ}(\boldsymbol{R}) \tag{68}$$

- This equation is essentially the same as the Hydrogen SE, and here we can also perform a separation of variables:

$$\Phi^n_{vJ}(\boldsymbol{R}) = \Phi^n_v(R)\Phi^n_J(\hat{\boldsymbol{R}}) \tag{69}$$

- For our purposes we will neglect rotational motion, and solve the following equation for the vibrational wave function:

$$\left[-\frac{1}{2\mu}\frac{\mathrm{d}^2}{\mathrm{d}R^2} + \epsilon_n(R)\right]\Phi^n_v(R) = \varepsilon_{nv}\Phi^n_v(R) \tag{70}$$

- Solving this is the same as solving the Hydrogen radial equation but with the inclusion of μ, and with $\ell = 0$

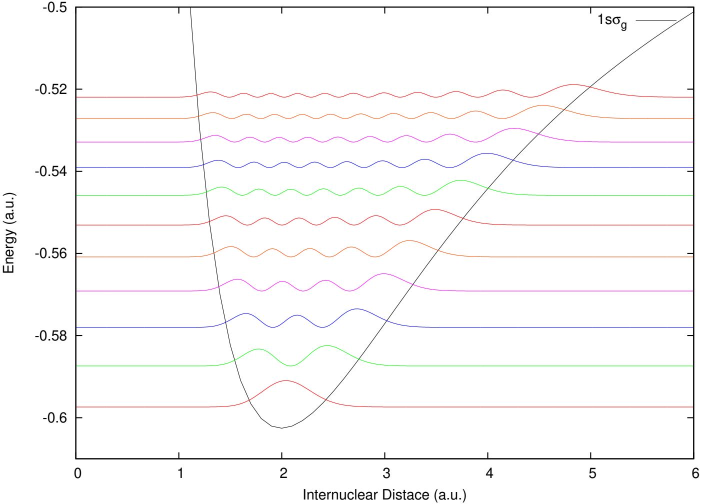
Vibrational wave functions

- The potential-energy curve $\epsilon(R)$ needs to be interpolated onto the grid defined in your code:

```
      SUBROUTINE INTRPL(L,X,Y,N,U,V)
      IMPLICIT REAL*8(A-H,O-Z)
      IMPLICIT INTEGER*4(I-N)

!     DOUBLE PRECISION INTERPOLATION OF A SINGLE VALUED FUNCTION
!     THIS SUBROUTINE INTERPOLATES, FROM VALUES OF THE FUNCTION
!     GIVEN  AS ORDINATES OF INPUT DATA POINTS IN AN X-Y PLANE
!     AND FOR A GIVEN SET OF X VALUES(ABSCISSAE),THE VALUES OF
!     A SINGLE VALUED FUNCTION Y=Y(X).

!     THE INPUT PARAMETERS ARE;

!     L=NUMBER OF DATA POINTS
!     (MUST BE TWO OR GREATER)
!     X=ARRAY OF DIMENSION L STORING THE X VALUES
!     OF INPUT DATA POINTS (IN ASCENDING ORDER)
!     Y=ARRAY OF DIMENSION L STORING THE Y VALUES OF INPUT DATA POINTS
!     N=NUMBER OF POINTS AT WHICH INTERPOLATION OF THE Y-VALUES
!     IS REQUIRED (MUST BE 1 OR GREATER)
!     U=ARRAY OF DIMENSION N STORING THE X VALUES
!     OF THE DESIRED POINTS

!     THE OUTPUT PARAMETER IS V=ARRAY OF DIMENSION N WHERE THE
!     INTERPOLATED Y VALUES ARE TO BE DISPLAYED
```

Vibrational wave functions

- Some vibrational wave functions in the H_2^+ $1s\sigma_g$ state:



2. Numerov's method

Lecture outcomes

- Apply finite-difference methods to solving the one-dimensional Schrödinger equation
- Calculate true continuum waves
- Investigate the dissociation of H_2^+

Finite difference methods

You are probably familiar with Euler's method for solving first-order ODE's:

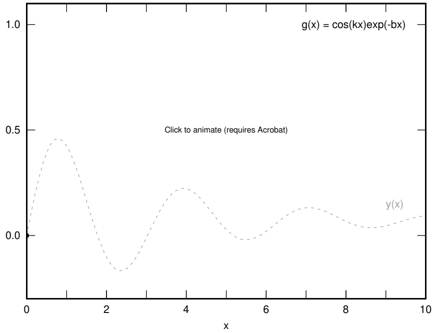
$$\frac{\mathrm{d}y}{\mathrm{d}x} = g(x)$$

by writing

$$\frac{\mathrm{d}y}{\mathrm{d}x} \approx \frac{y(x+\delta x) - y(x)}{\delta x}$$

and deriving an iteration scheme

$$y_{i+1} = y_i + g_i \delta x.$$



Similar expressions can be formed for higher-order derivatives:

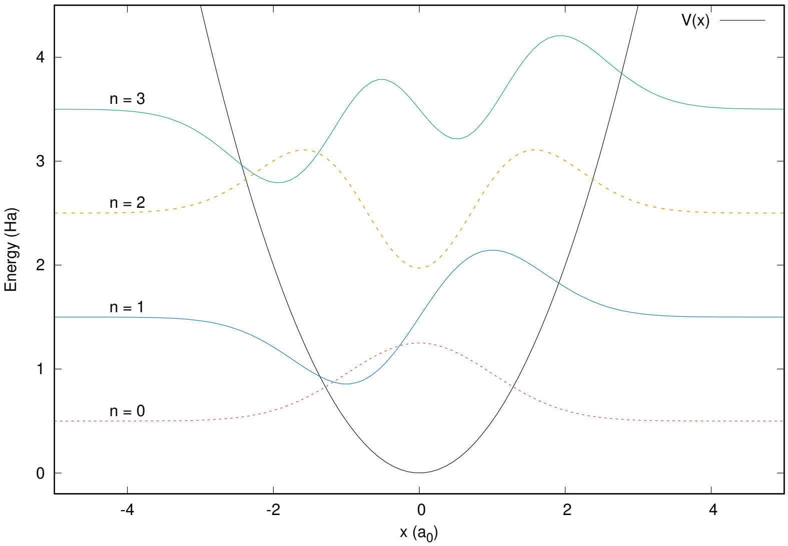
$$y(x+\delta x) = y(x) + \frac{\mathrm{d}y}{\mathrm{d}x}\delta x + \tfrac{1}{2}\frac{\mathrm{d}^2y}{\mathrm{d}x^2}\delta x^2 + \mathcal{O}(\delta x^3)$$

$$y(x-\delta x) = y(x) - \frac{\mathrm{d}y}{\mathrm{d}x}\delta x + \tfrac{1}{2}\frac{\mathrm{d}^2y}{\mathrm{d}x^2}\delta x^2 - \mathcal{O}(\delta x^3)$$

$$\frac{\mathrm{d}^2y}{\mathrm{d}x^2} \approx \frac{y(x+\delta x) - 2y(x) + y(x-\delta x)}{\delta x^2}.$$

This method neglects third-order and higher terms.

Quantum Harmonic Oscillator



We wish to solve

$$-\frac{1}{2}\frac{\mathrm{d}^2\psi(x)}{\mathrm{d}x^2} + V(x)\psi(x) = E\psi(x)$$

with

$$V(x) = \frac{1}{2}\omega^2 x^2.$$

Here, ω is the classical oscillator angular frequency. Discretising the x domain, we have

$$\begin{aligned}
\psi(x_i) &= \psi_i \\
V(x_i) &= V_i \\
\left.\frac{\mathrm{d}^2\psi}{\mathrm{d}x^2}\right|_{x=x_i} &= \frac{\psi_{i+1} - 2\psi_i + \psi_{i-1}}{\delta x^2}.
\end{aligned}$$

Substitution into the above Schrödinger equation gives

$$\begin{aligned}
&-\frac{1}{2}\frac{\psi_{i+1} - 2\psi_i + \psi_{i-1}}{\delta x^2} + (V_i - E)\psi_i = 0 \\
\Rightarrow &\psi_{i+1} = 2\left[\delta x^2(V_i - E) + 1\right]\psi_i - \psi_{i-1}.
\end{aligned}$$

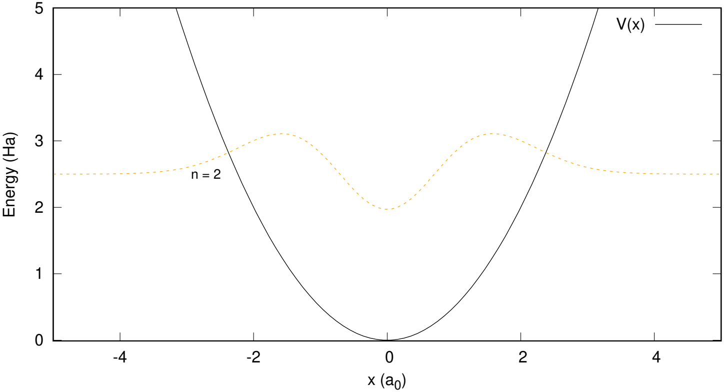
To obtain the solution $\{\psi_i\}_{i=1}^{N}$ iteratively we require two starting points ψ_1 and ψ_2 and an energy E.

Boundary conditions:

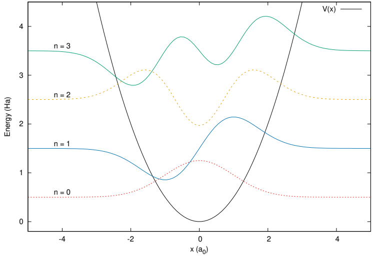
$V(x)$ is symmetric and $V \to \infty$ as $x \to \pm\infty$, so $\psi(x = \pm\infty) = 0$. If we define our x grid by $\{x_i\}_{i=1}^{N} = \{-a, -a + \delta x, \ldots, a - \delta x, a\}$ then provided $V(a) = V_N$ is sufficiently large we have

$$\psi_1 = \psi(-a) = 0, \qquad \psi_N = \psi(a) = 0.$$

We are free to choose ψ_2 arbitrarily and then normalise the solution afterwards. Let $\psi_2 = s$ for some small number s so the solution doesn't grow too large.



Shooting with bisection

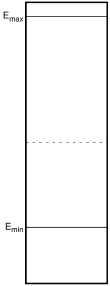


We need to guess an energy E to begin the iteration procedure. If E is not a true eigenenergy then the boundary condition $\psi_N = 0$ will not be met.
For the canonical solutions (right), ψ should always approach 0 from the positive direction for large x. If E_n is a true eigenenergy, then we have $(*)$

- $E > E_n \Rightarrow \psi \to -\infty$
- $E < E_n \Rightarrow \psi \to +\infty$

To locate the correct E_n we can use the bisection method:

- choose some suitable upper and lower bounds on the energy $E_{\min}$ and $E_{\max}$
- shoot (find ψ by iterating) with $E = (E_{\min} + E_{\max})/2$
- if $\psi_N > 0$ then set $E_{\min} = E$
- if $\psi_N < 0$ then set $E_{\max} = E$
- repeat until $|\psi_N| \leq \epsilon$ for some tolerance ϵ



To make use of the assumptions above $(*)$ we must set $\psi_2 = (-1)^n s$ for some small s (try $s = 10^{-4}$) to ensure the solutions match the canonical forms.

Counting nodes

There are an infinite number of solutions $\{\psi_n\}_{n=0}^{\infty}$, how do we find the one we want?
Count the number of nodes - if there are too many (too few) then decrease (increase) the energy.
The full procedure to find a specific eigenstate n is then:

0) Estimate E_{min} and E_{max}
1) Using $E = (E_{\text{min}} + E_{\text{max}})/2$ shoot from the left boundary to the right boundary to generate the wave function
2) Count the number of nodes n' in the wave function:
    - if $n' < n$ then set $E_{\text{min}} = E$ and go back to step (1)
    - if $n' > n$ then set $E_{\text{max}} = E$ and go back to step (1)
    - if $n' = n$ then continue
3) Check the boundary condition $\psi_N = 0$:
    - if $\psi_N > 0$ then set $E_{\text{min}} = E$
    - if $\psi_N < 0$ then set $E_{\text{max}} = E$
4) Check for convergence:
    - if $|E_{\text{max}} - E_{\text{min}}| \leq \epsilon$ or $|\psi_N| \leq \epsilon$ for some tolerance $\epsilon \ll 1$ then continue
    - If convergence is not achieved then go back to step (1)
5) Normalise the wave function

Normalising the wave function

The normalisation condition is

$$\langle\psi|\psi\rangle = \int_{-\infty}^{\infty} |\psi(x)|^2 \, \mathrm{d}x = 1.$$

Perform the integration numerically (e.g. Simpson integration from Lecture 1b) and then the normalised wave function is given by

$$\psi^{\text{norm.}} = \frac{1}{\sqrt{\langle\psi|\psi\rangle}}\psi.$$

Numerical solutions to the QHO can be compared with the analytical solutions. Below are some solutions for $\omega = 1$, in atomic units:

$$\begin{aligned}
\psi_0(x) &= \pi^{-1/4} e^{-x^2/2}, & E_0 &= \frac{1}{2} \\
\psi_1(x) &= \pi^{-1/4}\sqrt{2}x e^{-x^2/2}, & E_1 &= \frac{3}{2} \\
\psi_2(x) &= \pi^{-1/4}\frac{1}{\sqrt{2}}(2x^2-1) e^{-x^2/2}, & E_2 &= \frac{5}{2} \\
\psi_3(x) &= \pi^{-1/4}\frac{1}{\sqrt{3}}(2x^3-3x) e^{-x^2/2}, & E_3 &= \frac{7}{2}
\end{aligned}$$

Numerov's method

Numerov's method is a more accurate finite difference scheme to solve differential equations of the form

$$\frac{\mathrm{d}^2 f(x)}{\mathrm{d}x^2} = g(x)f(x).$$

Again begin with the Taylor expansions

$$f(x+\delta x) = f(x) + f^{(1)}(x)\delta x + \tfrac{1}{2!}f^{(2)}(x)\delta x^2 + \tfrac{1}{3!}f^{(3)}(x)\delta x^3 + \tfrac{1}{4!}f^{(4)}(x)\delta x^4 + \mathcal{O}(\delta x^5)$$

$$f(x-\delta x) = f(x) - f^{(1)}(x)\delta x + \tfrac{1}{2!}f^{(2)}(x)\delta x^2 - \tfrac{1}{3!}f^{(3)}(x)\delta x^3 + \tfrac{1}{4!}f^{(4)}(x)\delta x^4 - \mathcal{O}(\delta x^5)$$

and sum them (ignoring fifth order and higher terms):

$$f(x+\delta x) + f(x-\delta x) = 2f(x) + f^{(2)}(x)\delta x^2 + \frac{1}{12}f^{(4)}(x)\delta x^4.$$

We can write $f^{(4)}(x)$ as the second derivative of $f^{(2)}(x) = g(x)f(x)$:

$$f^{(4)}(x) = \frac{\mathrm{d}^2}{\mathrm{d}x^2}g(x)f(x) \approx \frac{g(x+\delta x)f(x+\delta x) - 2g(x)f(x) + g(x-\delta x)f(x-\delta x)}{\delta x^2}.$$

After some algebra, we get the Numerov iteration scheme

$$f_{i+1} = \frac{2\left(1+\dfrac{5\delta x^2}{12}g_i\right)f_i - \left(1-\frac{\delta x^2}{12}g_{i-1}\right)f_{i-1}}{1-\frac{\delta x^2}{12}g_{i+1}}.$$

To solve the Schrödinger equation

$$-\frac{1}{2}\frac{\mathrm{d}^2\psi(x)}{\mathrm{d}x^2} + V(x)\psi(x) = E\psi(x)$$

using Numerov's method, we set $g(x) = 2(V(x) - E)$, giving

$$\begin{aligned} \psi_{i+1} &= \frac{2\left(1 + \dfrac{5\delta x^2}{12} g_i\right)\psi_i - \left(1 - \frac{\delta x^2}{12} g_{i-1}\right)\psi_{i-1}}{1 - \frac{\delta x^2}{12} g_{i+1}} \\ g_i &= g(x_i) = 2(V_i - E). \end{aligned}$$

Like the previous shooting-with-bisection method, Numerov's method requires two starting points ψ_1 and ψ_2 and the energy E to find a solution. However this method is accurate to fourth order rather then second order.
To enforce correct boundary conditions for the QHO, we can set

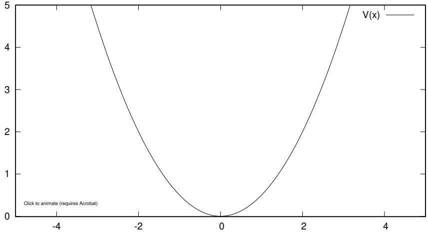
$$\psi_1 = 0 \qquad \psi_2 = (-1)^n s \qquad \psi_{N-1} = s \qquad \psi_N = 0,$$

iterate forwards from the left and backwards from the right, and match the solutions ψ^{L} and ψ^{R} at some point x_{m}, giving the approximate solution

$$\psi(x) = \begin{cases} \psi^{\mathrm{L}}(x)/\psi^{\mathrm{L}}(x_{\mathrm{m}}) & x \le x_{\mathrm{m}} \\ \psi^{\mathrm{R}}(x)/\psi^{\mathrm{R}}(x_{\mathrm{m}}) & x > x_{\mathrm{m}} \end{cases}$$

Cooley's energy correction

A faster way of locating the correct eigenenergy than the bisection method. The energy correction is given by



$$\Delta E \approx \frac{\langle\psi|H-E|\psi\rangle}{\langle\psi|\psi\rangle}.$$

In our discrete representation:

$$\langle\psi|\psi\rangle \approx \delta x \sum_{i=1}^{N} |\psi_i|^2$$

$$\langle\psi|H-E|\psi\rangle \approx \delta x \sum_{i=1}^{N} \psi_i \left[-\frac{1}{2}\frac{Y_{i+1}-2Y_i+Y_{i-1}}{\delta x^2} + (V_i - E)\psi_i\right],$$

where for compactness $Y_i = \left(1 - \frac{\delta x^2}{12} g_i\right)\psi_i$, and we have used the Numerov iteration scheme to replace $(H-E)|\psi\rangle$.
Since $(H-E)|\psi\rangle = 0$ everywhere except the matching point x_{m}, Cooley's energy correction is given by

$$\Delta E \approx \frac{\psi_m}{\sum_{i=1}^{N} |\psi_i|^2} \left[-\frac{1}{2}\frac{Y_{m+1}-2Y_m+Y_{m-1}}{\delta x^2} + (V_m - E)\psi_m\right].$$

Numerov-Cooley method

The Numerov-Cooley method begins with a guess of the energy E, then solving for the approximate wave function ψ repeatedly with the energy modified each time by the ΔE given by Cooley's correction.
Note that this method will always converge to the eigenenergy closest to your initial guess, so it is useful to first use node counting and the bisection method to find an appropriate initial guess for the Numerov-Cooley algorithm.
The full procedure to find a specific eigenstate n is then:

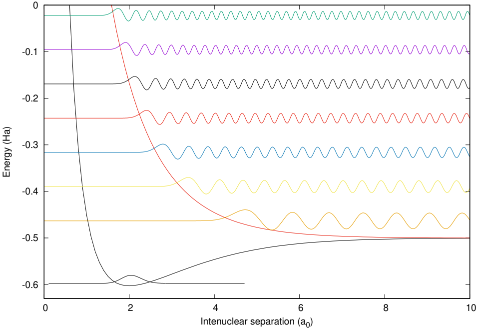
0) Estimate E_{min} and E_{max}
1) Using $E = (E_{\text{min}} + E_{\text{max}})/2$ shoot from the left and right boundaries and generate the approximate wave function
2) count the number of nodes n' in the wave function:
   - if $n' < n$ then set $E_{\text{min}} = E$ and go back to step (1)
   - if $n' > n$ then set $E_{\text{max}} = E$ and go back to step (1)
   - if $n' = n$ then continue
3) shoot from the left and right boundaries and generate the approximate wave function
4) compute Cooley's energy correction ΔE:
   - if $|\Delta E| > \epsilon$ for some tolerance $\epsilon \ll 1$ then replace E with $E + \Delta E$ and go back to step (3)
   - if $|\Delta E| \leq \epsilon$ then continue
5) normalise the wave function

Dissociative wave functions

Last week: bound vibrational wave functions (energy less than $R = \infty$ limit of PEC) represent oscillation of nuclei.
Solutions to vibrational wave equation

$$\left[-\frac{1}{2\mu}\frac{\mathrm{d}^2}{\mathrm{d}R^2} + \epsilon(R) - E\right]\nu(R) = 0 \tag{71}$$

with energies E above the PEC limit represent dissociation (nuclei fly apart)



- The energy E of the wave is partly kinetic energy of nuclei and partly the residual energy of the hydrogen atom (-0.5 Ha):

$$E = E_\mathrm{k} + D \tag{72}$$

  where E_k is usually called the “kinetic energy release” since it is the energy released when the molecule dissociates, and D is the $R = \infty$ limit of the PEC
- So it is convenient to rewrite the SE as

$$\left[-\frac{1}{2\mu} \frac{\mathrm{d}^2}{\mathrm{d}R^2} + \epsilon(R) - E_\mathrm{k} - D \right] \nu(R) = 0 \tag{73}$$

- Normalisation of continuum waves has some arbitrariness ($\langle \nu | \nu \rangle$ is undefined)
- We can choose the normalisation to satisfy the useful condition

$$\int \nu(R; E_\mathrm{k}) \nu(R'; E_\mathrm{k}) \, \mathrm{d}E_\mathrm{k} = \delta(R - R') \tag{74}$$

  by ensuring that as $R \to \infty$ the amplitude of $\nu(R)$ tends to

$$\sqrt{\frac{2\mu}{k\pi}}, \tag{75}$$

  where

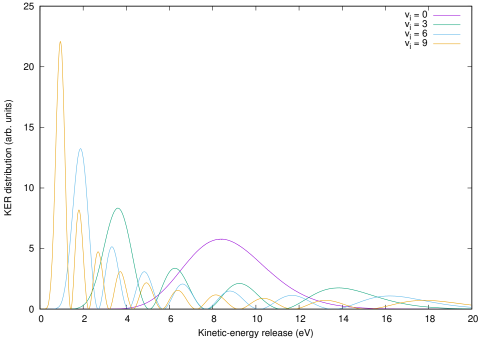
$$k = \sqrt{2\mu E_\mathrm{k}} \tag{76}$$

Franck-Condon approximation

- Kinetic-energy-release distribution: cross section as a function of E_k for excitation of the dissociative levels:

$$\frac{\mathrm{d}\sigma_{f,iv_i}}{\mathrm{d}E_\mathrm{k}} = \left| \langle \nu_{f E_\mathrm{k}} | \sqrt{\sigma_{f,i}(R)} | \nu_{iv_i} \rangle \right|^2 , \qquad (77)$$

  where $\sigma_{f,i}(R)$ is the “fixed-nuclei” cross section for the electronic excitation $i \to f$ as a function of R.
- Franck-Condon approximation: cross section for electronic+vibrational excitation is proportional to the Franck-Condon factor $\left| \langle \nu_{f E_\mathrm{k}} | \nu_{iv_i} \rangle \right|^2$



3. Potential scattering

Lecture outcomes

- Become familiar with partial-wave analysis
- Learn the methods for solving the Lippmann-Schwinger in momentum space
- Understand how differential and integral cross sections are calculated

In the theory lectures you have seen a derivation of the Lippmann-Schwinger equations in the s-wave model of electron-atom scattering:

$$\langle k_f \phi_f^{(N)} | T^{(SN)} | \phi_i^{(N)} k_i \rangle = \langle k_f \phi_f^{(N)} | V_S | \phi_i^{(N)} k_i \rangle + \sum_{n=1}^{N} \int_0^\infty \frac{\langle k_f \phi_f^{(N)} | V^{(SN)} | \phi_n^{(N)} k \rangle \langle k \phi_n^{(N)} | T^{(SN)} | \phi_i^{(N)} k_i \rangle}{E + i0 - \epsilon_n^{(N)} - k^2/2} \, \mathrm{d}k \tag{78}$$

To illustrate the method of solving these equations, we will consider the simpler problem with only the $1s$ state kept in the close-coupling expansion, and exchange neglected, but relax the assumption of $\ell = 0$ for the projectile.
The $1s$-$1s$ direct V-matrix elements are

$$\langle \boldsymbol{k}_f \phi_{1s} | \left[\frac{z}{r_1} - \frac{z}{|\boldsymbol{r}_1 - \boldsymbol{r}_2|} \right] | \phi_{1s} \boldsymbol{k}_i \rangle, \tag{79}$$

which we can integrate over the target space using the analytical $1s$ wave function to obtain

$$\langle \boldsymbol{k}_f | z \left(1 + \frac{1}{r_1} \right) e^{-2r_1} | \boldsymbol{k}_i \rangle. \tag{80}$$

Hence (dropping the subscript on r_1 from now on), we can forget about the target and treat this is a problem of a charged particle scattering on a static (structureless) potential given by

$$V(r) = z \left(1 + \frac{1}{r} \right) e^{-2r}. \tag{81}$$

Schrödinger equation and wavefunctions

The incident projectile is asymptotically (large r) described by plane waves $|\boldsymbol{k}_i\rangle$ and $|\boldsymbol{k}_f\rangle$, where

$$\langle \boldsymbol{r}|\boldsymbol{k}\rangle = \frac{1}{(2\pi)^{3/2}} e^{i\boldsymbol{r}\cdot\boldsymbol{k}}. \tag{82}$$

Note that in the present case only $k_f = k_i$ is physical.

The scattering wavefunction $\psi(\boldsymbol{r}, \boldsymbol{k}_i)$ describes the projectile at all r.

The Schrödinger equation for $|\psi\rangle$ is

$$(E - H_0)|\psi\rangle = V|\psi\rangle, \tag{83}$$

where H_0 is the free Hamiltonian (kinetic energy operator):

$$H_0|\boldsymbol{k}\rangle = \frac{k^2}{2}|\boldsymbol{k}\rangle. \tag{84}$$

Scattering equations and matrix elements

The Lippmann-Schwinger equation for the potential scattering system is:

$$\langle \boldsymbol{k}_f|T|\boldsymbol{k}_i\rangle = \langle \boldsymbol{k}_f|V|\boldsymbol{k}_i\rangle + \int \frac{\langle \boldsymbol{k}_f|V|\boldsymbol{k}\rangle\langle \boldsymbol{k}|T|\boldsymbol{k}_i\rangle}{E + i0 - k^2/2}\,\mathrm{d}\boldsymbol{k}, \tag{85}$$

where $\mathrm{d}\boldsymbol{k}$ indicates the integral is over all space.
To simplify this equation the plane waves are expanded in partial waves:

$$\langle \boldsymbol{r}|\boldsymbol{k}\rangle = \sqrt{\frac{2}{\pi}}\frac{1}{kr}\sum_{\ell m} i^\ell u_\ell(r;k)Y_\ell^m(\hat{\boldsymbol{r}})Y_\ell^{m*}(\hat{\boldsymbol{k}}), \tag{86}$$

where u_ℓ are the free-particle continuum waves which satisfy

$$\left[-\frac{1}{2}\frac{\mathrm{d}^2}{\mathrm{d}r^2} + \frac{\ell(\ell+1)}{2r^2} - \epsilon_k\right]u_\ell(r;k) = 0, \qquad \epsilon_k \equiv \frac{k^2}{2} \tag{87}$$

Substituting the partial-wave expansion of the plane waves into $\langle \boldsymbol{k}_f|V|\boldsymbol{k}_i\rangle$ gives

$$\langle \boldsymbol{k}_f|V|\boldsymbol{k}_i\rangle \tag{88}$$

$$= \frac{2}{\pi}\frac{1}{k_f k_i}\sum_{\substack{\ell_f m_f\\ \ell_i m_i}} i^{\ell_i-\ell_f}\int u_{\ell_f}(k_f r)Y_{\ell_f}^{m_f *}(\hat{\boldsymbol{r}})Y_{\ell_f}^{m_f}(\hat{\boldsymbol{k}}_f)V(\hat{\boldsymbol{r}})u_{\ell_i}(k_i r)Y_{\ell_i}^{m_i}(\hat{\boldsymbol{r}})Y_{\ell_i}^{m_i *}(\hat{\boldsymbol{k}}_i)\frac{\mathrm{d}\boldsymbol{r}}{r^2}$$

$$= \frac{1}{k_f k_i}\sum_{\substack{\ell_f m_f\\ \ell_i m_i}} i^{\ell_i-\ell_f}V_{\ell_f m_f,\ell_i m_i}(k_f,k_i)Y_{\ell_f}^{m_f}(\hat{\boldsymbol{k}}_f)Y_{\ell_i}^{m_i *}(\hat{\boldsymbol{k}}_i) \tag{89}$$

The partial-wave V-matrix elements are defined by

$$V_{\ell_f m_f,\ell_i m_i}(k_f,k_i) = \frac{2}{\pi}\int \frac{1}{r^2} u_{\ell_f}(k_f r) V(\hat{\boldsymbol{r}}) u_{\ell_i}(k_i r) Y^*_{\ell_f m_f}(\hat{\boldsymbol{r}}) Y_{\ell_i m_i}(\hat{\boldsymbol{r}})\,\mathrm{d}\boldsymbol{r}. \tag{90}$$

For a spherically symmetric potential, $V(\boldsymbol{r}) = V(r)$ and this simplifies to

$$V_{\ell_f m_f,\ell_i m_i}(k_f,k_i) = \frac{2}{\pi}\int_0^\infty \frac{1}{r^2} u_{\ell_f}(k_f r) V(r) u_{\ell_i}(k_i r) r^2\,\mathrm{d}r \int Y^*_{\ell_f m_f}(\hat{\boldsymbol{r}}) Y_{\ell_i m_i}(\hat{\boldsymbol{r}})\,\mathrm{d}\hat{\boldsymbol{r}} \tag{91}$$

$$= \frac{2}{\pi}\int_0^\infty \frac{1}{r^2} u_{\ell_f}(k_f r) V(r) u_{\ell_i}(k_i r) r^2\,\mathrm{d}r\ \delta_{\ell_f \ell_i}\delta_{m_f m_i}. \tag{92}$$

Since the partial-wave V matrix elements have no dependence on m_f,m_i and are non-zero only for $\ell_f = \ell_i$, we can denote them by a single ℓ:

$$V_\ell(k_f,k_i) = \frac{2}{\pi}\int_0^\infty u_\ell(r;k_f) V(r) u_\ell(r;k_i)\,\mathrm{d}r \tag{93}$$

and we can write the partial-wave expansion of $\langle \boldsymbol{k}_f|V|\boldsymbol{k}_i\rangle$ as

$$\langle \boldsymbol{k}_f|V|\boldsymbol{k}_i\rangle = \frac{1}{k_f k_i}\sum_{\ell m} V_\ell(k_f,k_i) Y_{\ell m}(\hat{\boldsymbol{k}}_f) Y^*_{\ell m}(\hat{\boldsymbol{k}}_i). \tag{94}$$

The T-matrix is expanded in a similar fashion:

$$\langle \boldsymbol{k}_f | T | \boldsymbol{k}_i \rangle = \frac{1}{k_f k_i} \sum_{\ell m} T_\ell(k_f, k_i) Y_{\ell m}(\hat{\boldsymbol{k}}_f) Y^*_{\ell m}(\hat{\boldsymbol{k}}_i). \tag{95}$$

Hence, by obtaining the partial-wave T-matrix elements $T_\ell(k_f, k_i)$ the scattering problem is solved. Substituting the partial-wave expansions of the 3D T and V matrices into the Lippmann-Schwinger equation gives the partial-wave Lippmann-Schwinger equation:

$$T_\ell(k_f, k_i) = V_\ell(k_f, k_i) + \int_0^\infty \frac{V_\ell(k_f, k) T_\ell(k, k_i)}{E - k^2/2 + i0} \, \mathrm{d}k. \tag{96}$$

Integrating over the singularity at $k = k_i$, we have

$$T_\ell(k_f, k_i) = V_\ell(k_f, k_i) + \mathcal{P} \int_0^\infty \frac{V_\ell(k_f, k) T_\ell(k, k_i)}{E - k^2/2} \, \mathrm{d}k - \frac{i\pi}{k_f} V_\ell(k_f, k_i) T_\ell(k_i, k_i). \tag{97}$$

Now defining the K matrix:

$$K_\ell(k_f, k_i) = T_\ell(k_f, k_i) \left[1 + \frac{i\pi}{k_f} K_\ell(k_f, k_i) \right] \tag{98}$$

we have

$$K_\ell(k_f, k_i) = V_\ell(k_f, k_i) + \mathcal{P} \int_0^\infty \frac{V_\ell(k_f, k) K_\ell(k, k_i)}{E - k^2/2} \, \mathrm{d}k \tag{99}$$

which can be solved using real arithmetic.

Solving the K-matrix equation

To solve

$$K_\ell(k_f, k_i) = V_\ell(k_f, k_i) + \mathcal{P}\int_0^\infty \frac{V_\ell(k_f, k)K_\ell(k, k_i)}{E - k^2/2}\,\mathrm{d}k \tag{100}$$

we evaluate the integral numerically by discretising the k domain we are integrating over:

$$K_\ell(k_f, k_i) = V_\ell(k_f, k_i) + \sum_n w_n V_\ell(k_f, k_n) K_\ell(k_n, k_i), \tag{101}$$

where n covers some finite number of grid points of our choosing, k_n is the value of k at the nth point in the grid, and w_n contains both the integration weight and the value of the Green's function at the point n:

$$G_n = \frac{1}{E - k_n^2/2}. \tag{102}$$

Now by writing

$$K_{fi} \equiv K(k_f, k_i), \qquad V_{fn} \equiv V(k_f, k_n) \qquad \text{etc.} \tag{103}$$

we have

$$K_{fi} = V_{fi} + \sum_n w_n V_{fn} K_{ni}. \tag{104}$$

Note we have neglected ℓ as we can solve the equation independently for each ℓ. Remember that the dependence on ℓ enters the equation via the V matrix.

Solving the K-matrix equation

We have now transformed the integral equation

$$K_\ell(k_f, k_i) = V_\ell(k_f, k_i) + \mathcal{P}\int_0^\infty \frac{V_\ell(k_f, k)K_\ell(k, k_i)}{E - k^2/2}\,\mathrm{d}k \tag{105}$$

into a matrix equation

$$K_{fi} = V_{fi} + \sum_n w_n V_{fn} K_{ni}. \tag{106}$$

To solve this equation we must allow f to range over the same values of k as n, and rearrange:

$$\sum_n \left[\delta_{fn} - w_n V_{fn}\right] K_{ni} = V_{fi}. \tag{107}$$

Because our k grid does not contain the on-shell momentum, f and n only range over the off-shell momenta, this means that:

- The solution K_{ni} we obtain is *half on shell*
- We must then obtain the on-shell K_{fi} by substituting the half-on-shell K_{ni} back into

$$K_{fi} = V_{fi} + \sum_n w_n V_{fn} K_{ni}. \tag{108}$$

Obtaining the on-shell T-matrix and scattering amplitude

We rearrange the definition of the K matrix:

$$K_{fi} = T_{fi}\left[1 + \frac{i\pi}{k_f} K_{fi}\right] \tag{109}$$

to give

$$T_{fi} = \frac{K_{fi}}{1 + \frac{i\pi}{k_f} K_{fi}} \tag{110}$$

for the on-shell T matrix element.
From the theory lectures remember the definition of the scattering amplitude

$$f(\boldsymbol{k}_f, \boldsymbol{k}_i) = -4\pi^2 \langle \boldsymbol{k}_f | T | \boldsymbol{k}_i \rangle, \tag{111}$$

and in terms of the partial-wave T-matrix elements we have obtained:

$$f(\boldsymbol{k}_f, \boldsymbol{k}_i) = -\frac{4\pi^2}{k_f k_i} \sum_{\ell m} T_\ell(k_f, k_i) Y_{\ell m}(\hat{\boldsymbol{k}}_f) Y_{\ell m}(\hat{\boldsymbol{k}}_i). \tag{112}$$

Noting that for a static potential only elastic ($\left|\boldsymbol{k}_f\right| = |\boldsymbol{k}_i|$) scattering can occur we can write

$$f(\boldsymbol{k}_f, \boldsymbol{k}_i) = -\frac{4\pi^2}{k_i^2} \sum_{\ell m} T_\ell(k_i, k_i) Y_{\ell m}(\hat{\boldsymbol{k}}_f) Y_{\ell m}(\hat{\boldsymbol{k}}_i). \tag{113}$$

Differential cross sections

Using one of the many useful identities of the spherical harmonics:

$$\sum_{m=-\ell}^{\ell} Y_{\ell m}^{*}(\hat{\boldsymbol{x}})Y_{\ell m}(\hat{\boldsymbol{y}}) = \frac{2\ell+1}{4\pi}P_\ell(\hat{\boldsymbol{x}}\cdot\hat{\boldsymbol{y}}), \tag{114}$$

where P_ℓ is the ℓth Legendre polynomial, we have

$$f(\boldsymbol{k}_f,\boldsymbol{k}_i) = -\frac{\pi}{k_i^2}\sum_\ell (2\ell+1)T_\ell(k_i,k_i)P_\ell(\cos\theta), \tag{115}$$

where

$$\cos\theta = \hat{\boldsymbol{k}}_f\cdot\hat{\boldsymbol{k}}_i \tag{116}$$

and hence θ is the angle of deflection of the scattering particle.
The differential cross section for scattering with an angle θ is given by

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega} = \frac{k_f}{k_i}\left|f(\boldsymbol{k}_f,\boldsymbol{k}_i)\right|^2 = \left|f(\boldsymbol{k}_f,\boldsymbol{k}_i)\right|^2. \tag{117}$$

Remember that the scattering amplitude is complex, so

$$|f|^2 = f^* f \neq f^2. \tag{118}$$

In Fortran, the `abs` function accepts complex arguments too.

Integrated cross sections

The integrated cross section can be obtain by numerically integrating the differential cross section:

$$\sigma = \int \left| f(\boldsymbol{k}_f, \boldsymbol{k}_i) \right|^2 \, \mathrm{d}\Omega, \tag{119}$$

but we can also derive an analytical expression for σ. We start by writing

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega} = \left| f(\boldsymbol{k}_f, \boldsymbol{k}_i) \right|^2 = \frac{\pi^2}{k_i^4} \sum_{\ell} \sum_{\ell'} (2\ell+1)(2\ell'+1) T_\ell(k_i, k_i) T^*_{\ell'}(k_i, k_i) P_\ell(\cos\theta) P_{\ell'}(\cos\theta). \tag{120}$$

Since the dependence on θ is entirely contained in the two Legendre polynomials:

$$\sigma = \frac{\pi^2}{k_i^4} \sum_{\ell} \sum_{\ell'} (2\ell+1)(2\ell'+1) T_\ell(k_i, k_i) T^*_{\ell'}(k_i, k_i) \int P_\ell(\cos\theta) P_{\ell'}(\cos\theta) \, \mathrm{d}\Omega \tag{121}$$

$$= \frac{\pi^2}{k_i^4} \sum_{\ell} \sum_{\ell'} (2\ell+1)(2\ell'+1) T_\ell(k_i, k_i) T^*_{\ell'}(k_i, k_i) \int_0^{2\pi} \mathrm{d}\phi \int_0^{\pi} P_\ell(\cos\theta) P_{\ell'}(\cos\theta) \sin\theta \, \mathrm{d}\theta \tag{122}$$

$$= \frac{2\pi^3}{k_i^4} \sum_{\ell} \sum_{\ell'} (2\ell+1)(2\ell'+1) T_\ell(k_i, k_i) T^*_{\ell'}(k_i, k_i) \int_0^{\pi} P_\ell(\cos\theta) P_{\ell'}(\cos\theta) \sin\theta \, \mathrm{d}\theta \tag{123}$$

Integrated cross sections

The Legendre polynomials form an orthogonal set:

$$\int_0^\pi P_\ell(\cos\theta) P_{\ell'}(\cos\theta) \sin\theta \, \mathrm{d}\theta = \frac{2}{2\ell+1}\delta_{\ell\ell'}, \tag{124}$$

which reduces σ to

$$\sigma = \frac{2\pi^3}{k_i^4} \sum_\ell \sum_{\ell'} (2\ell+1)(2\ell'+1) T_\ell(k_i,k_i) T^*_{\ell'}(k_i,k_i) \frac{2}{2\ell+1}\delta_{\ell\ell'} \tag{125}$$

$$= \frac{4\pi^3}{k_i^4} \sum_\ell (2\ell+1) T_\ell(k_i,k_i) T^*_\ell(k_i,k_i) \tag{126}$$

$$= \frac{4\pi^3}{k_i^4} \sum_\ell (2\ell+1) \left| T_\ell(k_i,k_i) \right|^2 . \tag{127}$$

We can define the partial-wave cross section

$$\sigma_\ell \equiv \frac{4\pi^3}{k_i^4} (2\ell+1) \left| T_\ell(k_i,k_i) \right|^2 \tag{128}$$

and then write

$$\sigma = \sum_\ell \sigma_\ell . \tag{129}$$

Computational implementation - outline

Given a central potential $V(r)$, we solve the scattering problem for a number of incident energies E. Before going further, here is a basic outline:

1. read in input data: incident energy, grid size/spacing, $\ell_{\max}$, projectile charge
2. set up the radial grid and integration weights
3. set up the k grid, Green's function, and integration weights
4. set up the interaction potential $V(r)$ on a grid
5. for each $\ell = 0, \ldots, \ell_{\max}$:
   a) Calculate the continuum waves for each k in the k grid
   b) Calculate the V-matrix elements for each k_n,k_m in the k grid

$$V_{nm} = \frac{2}{\pi} \int_0^\infty u_\ell(r;k_n) V(r) u_\ell(r;k_m)\, \mathrm{d}r \tag{130}$$

   c) Solve the system of linear equations for the half-on-shell K matrix K_{ni}

$$\sum_n \left[\delta_{fn} - w_n V_{fn}\right] K_{ni} = V_{fi} \tag{131}$$

   d) Obtain the on-shell K-matrix element using

$$K_{fi} = V_{fi} + \sum_n w_n V_{fn} K_{ni}. \tag{132}$$

   e) Use the on-shell K-matrix element K_{fi} to obtain the on-shell T-matrix element

$$T_{fi} = \frac{K_{fi}}{1 + \frac{i\pi}{k_f} K_{fi}} \tag{133}$$

Radial and momentum-space grids

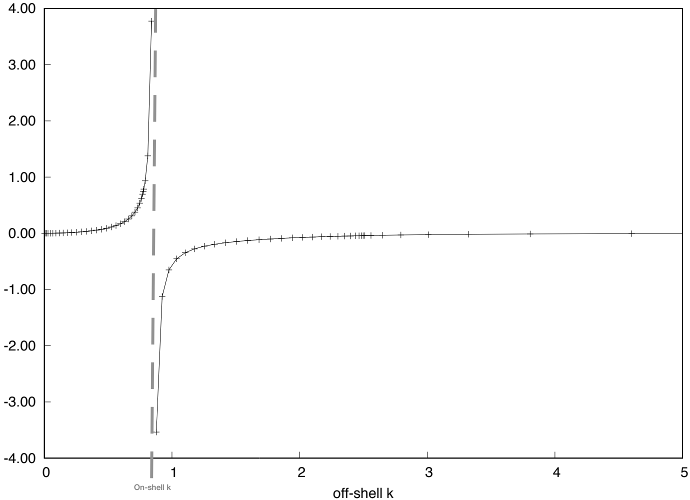
- We may be dealing with potentials which are undefined at $r = 0$, but any wave functions we calculate must be zero at $r = 0$. To avoid numerical issues, the first radial grid point should be $\mathrm{d}r$.
- When setting up Simpson integration weights, skip the first point so that instead of having n odd and

$$\int_0^\infty f(r)\,\mathrm{d}r \approx (f(r_1)+4f(r_2)+2f(r_3)+4f(r_4)+\cdots+4f(r_{n-1})+f(r_n))\cdot\frac{\delta r}{3} \quad (134)$$

  we have n even and

$$\int_0^\infty f(r)\,\mathrm{d}r \approx (4f(r_1)+2f(r_2)+4f(r_3)+\cdots+4f(r_{n-1})+2f(r_n))\cdot\frac{\delta r}{3}. \quad (135)$$

- You will be provided with a subroutine for setting up the k grid. It populates two arrays, `kgrid(1:nkmax)` and `kweights(1:nkmax)`. Since the integration in the K-matrix equation only covers the off-shell k, the on-shell k is stored in `kgrid(1)` so that the principle-value integration can be performed over the points `kgrid(2)` onwards.
- The `kweights` array contains the corresponding integration weights and values of the Green's function (denominator in the K-matrix equation). The on-shell point is not included in the integration so `kweights(1)` is zero.



Above: half-shell integrand in the K-matrix equation. k-grid points near the on-shell point must be chosen symmetrically about the singularity.

Continuum waves

- The continuum waves should be stored in a matrix `contwaves(:,:)`, with each column containing the radial function $u_\ell(r;k)$ for a given k in the k grid.
- You should modify your existing forwards-shooting numerov subroutine to calculate the solutions, using $\dfrac{\ell(\ell+1)}{2r^2}$ as the potential term and $k^2/2$ as the energy.
- The solutions u_ℓ are related to the spherical bessel functions j_ℓ:

  $$u_\ell(r;k) = krj_\ell(kr) \tag{136}$$

  and at small values of r are given by

  $$\lim_{r\to 0} u_\ell(r;k) = \frac{(rk)^{\ell+1}}{(2\ell+1)!!}, \tag{137}$$

  where !! denotes the double factorial:

  $$n!! \equiv \begin{cases} n(n-2)(n-4)\cdots 2, & n \text{ even} \\ n(n-2)(n-4)\cdots 1, & n \text{ odd} \end{cases} . \tag{138}$$

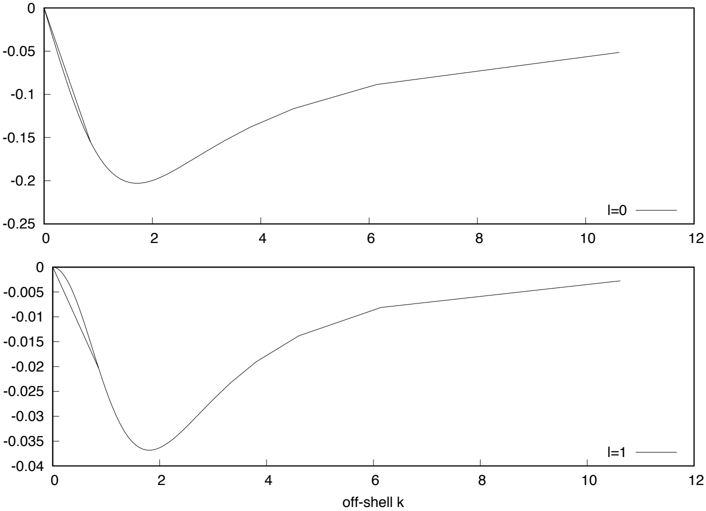
- Eq. (137) should be used to set the first two values of u_ℓ before the Numerov iteration takes over.
- To check that your continuum waves are calculated properly compare them with $\sin(kr)$ for $\ell = 0$.

V-matrix elements

- The V-matrix elements are straight-forward numerical integrations

$$V_{nm} = \frac{2}{\pi} \int_0^\infty u_\ell(r;k_n) V(r) u_\ell(r;k_m)\,\mathrm{d}r. \tag{139}$$

- Store the V-matrix elements in a matrix `Vmat(:,:)` where both indices range over the entire k grid. The `Vmat(1,1)` element is the fully-on-shell V-matrix element. All elements in the first row or column are half on-shell. All remaining elements are fully off-shell.



Above: half-onshell V-matrix element $V_\ell(k, k_i)$ for $\ell = 0$ and $\ell = 1$, using the radial potential

$$V(r) = -\left(1 + \frac{1}{r}\right) e^{-2r}. \tag{140}$$

Solving the K-matrix equation

$$\sum_n \left[\delta_{fn} - w_n V_{fn}\right] K_{ni} = V_{fi} \tag{141}$$

- In the above equation f and n range over all off-shell indices, while i is the on-shell index
- We will use the `LAPACK` subroutine `DGESV` to solve the linear system $Ax = b$. This subroutine inputs the coefficient matrix A and the RHS vector b, and overwrites b with the solution vector x.
- Define a matrix `A(:,:)` of dimension `(nkmax-1)*(nkgmax-1)`
- Populate `A` with values

$$\texttt{A(f,n)} = \delta_{\texttt{fn}} - \texttt{kweights(n+1)*Vmat(f+1,n+1)}, \qquad \texttt{f,n} = 1, \ldots, \texttt{nkmax-1} \tag{142}$$

- Define a vector `Koff` to contain the solution vector (half-off-shell K-matrix elements). Prepare it for the `DGESV` subroutine by populating it with the RHS vector elements:

$$\texttt{Koff(n) = Vmat(n+1,1).} \tag{143}$$

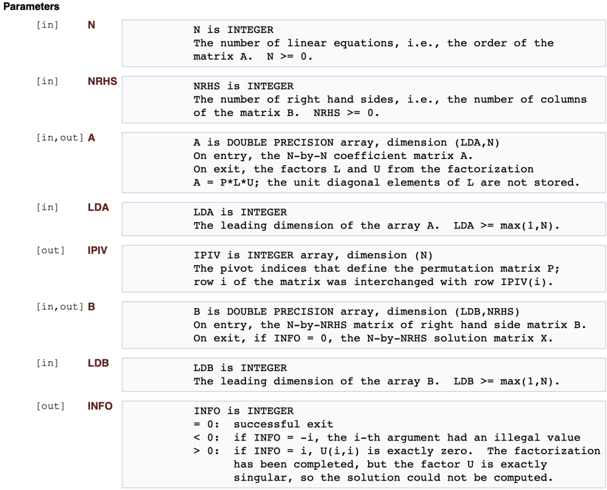
- After calling the `DGESV` subroutine, obtain the on-shell K-matrix elements using

$$K_{fi} = V_{fi} + \sum_n w_n V_{fn} K_{ni}. \tag{144}$$

DGESV Documentation

From netlib.org

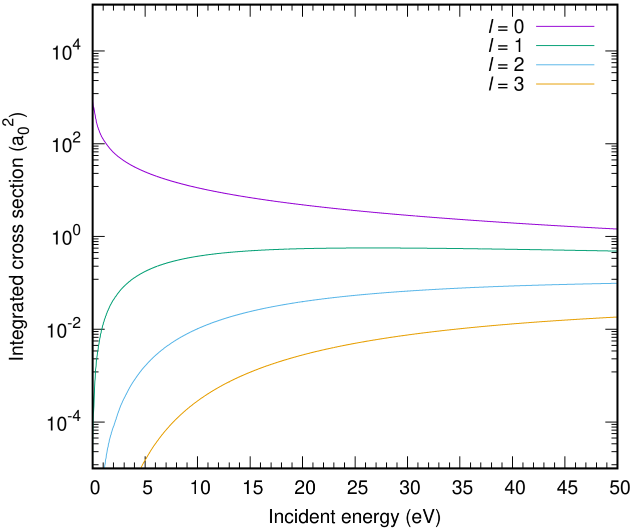
DGESV computes the solution to system of linear equations `A * X = B`



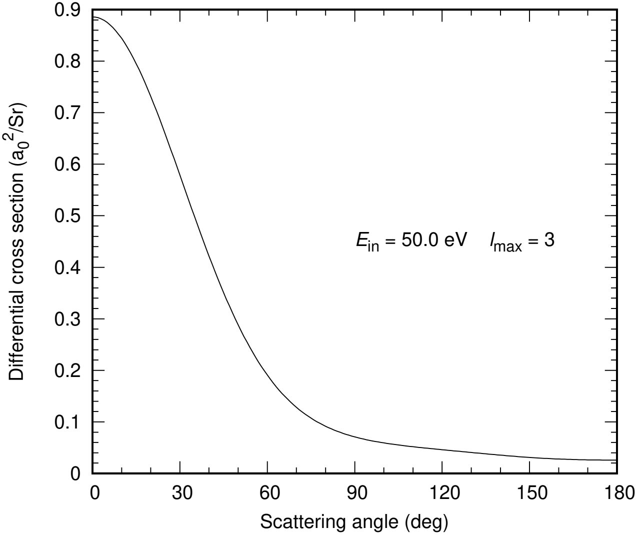
Parameters

[in]	**N**	N is INTEGER The number of linear equations, i.e., the order of the matrix A. N >= 0.
[in]	**NRHS**	NRHS is INTEGER The number of right hand sides, i.e., the number of columns of the matrix B. NRHS >= 0.
[in,out]	**A**	A is DOUBLE PRECISION array, dimension (LDA,N) On entry, the N-by-N coefficient matrix A. On exit, the factors L and U from the factorization A = P*L*U; the unit diagonal elements of L are not stored.
[in]	**LDA**	LDA is INTEGER The leading dimension of the array A. LDA >= max(1,N).
[out]	**IPIV**	IPIV is INTEGER array, dimension (N) The pivot indices that define the permutation matrix P; row i of the matrix was interchanged with row IPIV(i).
[in,out]	**B**	B is DOUBLE PRECISION array, dimension (LDB,NRHS) On entry, the N-by-NRHS matrix of right hand side matrix B. On exit, if INFO = 0, the N-by-NRHS solution matrix X.
[in]	**LDB**	LDB is INTEGER The leading dimension of the array B. LDB >= max(1,N).
[out]	**INFO**	INFO is INTEGER = 0: successful exit < 0: if INFO = -i, the i-th argument had an illegal value > 0: if INFO = i, U(i,i) is exactly zero. The factorization has been completed, but the factor U is exactly singular, so the solution could not be computed.

Example integrated cross sections (electron)



Example differential cross section (electron)



Example K- and T-matrix elements (50 eV electron)

ℓ	on-shell K	on-shell T
0	`-5.22854E-01`	`(-3.01497E-01, -2.58338E-01)`
1	`-1.35882E-01`	`(-1.29462E-01, -2.88289E-02)`
2	`-4.63427E-02`	`(-4.60770E-02, -3.49937E-03)`
3	`-1.69079E-02`	`(-1.68949E-02, -4.68133E-04)`

4. Electron-hydrogen V-matrix elements

Lecture outcomes

- Learn how V-matrix elements are calculated for the real problem of s-wave e-H scattering
- See how non-uniqueness affects real scattering calculations

To calculate electron-hydrogen scattering we must evaluate the V-matrix elements

$$\begin{aligned} V^S_{f,i}(\boldsymbol{k}',\boldsymbol{k}) &= \langle \boldsymbol{k}'\Phi_f|V^S|\Phi_i\boldsymbol{k}\rangle && (145)\\ &= \langle \boldsymbol{k}'\Phi_f|V-(-1)^S(E-H)P_r|\Phi_i\boldsymbol{k}\rangle && (146)\\ &= \underbrace{\langle \boldsymbol{k}'\Phi_f|V_1+V_{12}|\Phi_i\boldsymbol{k}\rangle}_{\text{direct term}} -(-1)^S\underbrace{\langle \boldsymbol{k}'\Phi_f|E-H|\boldsymbol{k}\Phi_i\rangle}_{\text{exchange term}} && (147)\end{aligned}$$

- $V_1 = -\dfrac{1}{r_1}$ electron-nuclear potential
- $V_{12} = |\boldsymbol{r}_1-\boldsymbol{r}_2|^{-1} = \displaystyle\sum_{\lambda=0}^{\infty}\frac{r_<^\lambda}{r_>^{\lambda+1}}P_\lambda(\cos\theta)$ electron-electron potential

We'll work within the s-wave model so the electron-electron potential can be treated as

$$V_{12} = \frac{1}{r_>}, \tag{148}$$

and the partial-wave radial functions are

$$\langle r|k\rangle = \sqrt{\frac{2}{\pi}}\frac{1}{r}u_0(r;k) = \sqrt{\frac{2}{\pi}}\frac{1}{r}\sin(kr). \tag{149}$$

The target wave functions are just the $\ell = 0$ radial Hydrogen wave functions from Lecture 1:

$$\langle r|\Phi_i\rangle = \sum_j c_{ji}\varphi_j(r), \tag{150}$$

where φ are the radial Laguerre basis functions.

Direct term

Recalling the partial-wave expansion of the V-matrix elements from Eq. (89) in Lecture 3, we then have in the s-wave model:

$$\langle \boldsymbol{k}'\Phi_f|V^S|\Phi_i\boldsymbol{k}\rangle = \frac{1}{k_f k_i}\langle k'\Phi_f|V^S|\Phi_i k\rangle. \tag{151}$$

The direct partial-wave V-matrix element is:

$$\langle k'\Phi_f|V_1 + V_{12}|\Phi_i k\rangle = \frac{2}{\pi}\int_0^\infty\int_0^\infty \sin(k'r_1)\Phi_f(r_2)\left(\frac{1}{r_>} - \frac{1}{r_1}\right)\Phi_i(r_2)\sin(kr_1)\,\mathrm{d}r_1\,\mathrm{d}r_2 \tag{152}$$

$$= \frac{2}{\pi}\int_0^\infty \sin(k'r_1)\sin(kr_1)\left[\frac{1}{r_1}\int_0^{r_1}\Phi_f(r_2)\Phi_i(r_2)\mathrm{d}r_2 + \int_{r_1}^\infty \Phi_f(r_2)\frac{1}{r_2}\Phi_i(r_2)\mathrm{d}r_2 - \frac{\delta_{fi}}{r_1}\right]\mathrm{d}r_1. \tag{153}$$

We need to efficiently evaluate an integral of the form

$$\int_0^\infty f(r_1)\left(\frac{1}{r_1}\int_0^{r_1} g(r_2)\,\mathrm{d}r_2 + \int_{r_1}^\infty \frac{1}{r_2}g(r_2)\,\mathrm{d}r_2\right)\,\mathrm{d}r_1 \tag{154}$$

$$\approx \sum_{i=1}^n w_i f(r_i)\left(\frac{1}{r_i}\sum_{j=1}^i w_j g(r_j) + \sum_{j=i}^n w_j\frac{1}{r_j}g(r_j)\right) \tag{155}$$

where w_i are integration weights and n is the number of discrete radial grid points.

To avoid recalculating the inner sums for each value of r_1 in the outer sum, we follow these steps:

1. Iterating forwards over the radial grid from $i = 1$ to n, populate an array

$$\mathtt{A1}(i) = \sum_{j=1}^{i} w_j g(r_j) = \mathtt{A1}(i-1) + w_i g(r_i) \tag{156}$$

2. Iterating backwards over the radial grid from $i = n$ to 1, populate an array

$$\mathtt{A2}(i) = \sum_{j=i}^{n} w_j \frac{1}{r_j} g(r_j) = \mathtt{A2}(i+1) + w_i \frac{1}{r_i} g(r_i) \tag{157}$$

3. Evaluate the final sum using

$$\sum_{i=1}^{n} w_i f(r_i) \left(\frac{1}{r_i} \sum_{j=1}^{i} w_j g(r_j) + \sum_{j=i}^{n} w_j \frac{1}{r_j} g(r_j) \right) = \sum_{i}^{n} w_i f(r_i) \left(\frac{1}{r_i} \mathtt{A1}(i) + \mathtt{A2}(i) \right). \tag{158}$$

Exchange term

The total Hamiltonian is

$$H = K_1 + K_2 + V_1 + V_2 + V_{12} \tag{159}$$

So the exchange matrix element can be split up into

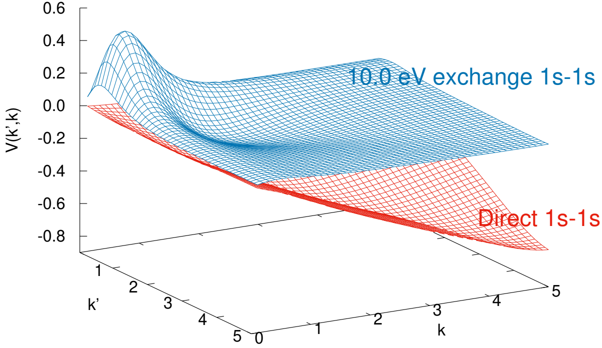
$$\begin{aligned}\langle k'\Phi_f|E-H|k\Phi_i\rangle = (E - k'^2/2 - k^2/2)\langle k'|\Phi_i\rangle\langle\Phi_f|k\rangle \\ - \langle k'|V_1|\Phi_i\rangle\langle\Phi_f|k\rangle - \langle k'|\Phi_i\rangle\langle\Phi_f|V_2|k\rangle \\ - \langle k'\Phi_f|V_{12}|k\Phi_i\rangle.\end{aligned} \tag{160}$$

All but the last term have been separated into products of one-dimensional integrals. The *non-separable* term

$$\begin{aligned}&\langle k'\Phi_f|V_{12}|k\Phi_i\rangle \\ &= \frac{2}{\pi}\int_0^\infty \sin(k'r_1)\Phi_i(r_1)\left[\frac{1}{r_1}\int_0^{r_1}\Phi_f(r_2)\sin(kr_2)\,\mathrm{d}r_2 + \int_{r_1}^\infty \Phi_f(r_2)\frac{1}{r_2}\sin(kr_2)\,\mathrm{d}r_2\right]\mathrm{d}r_1\end{aligned} \tag{161}$$

has the same form as the V_{12} term of the direct matrix element, and can be evaluated in exactly the same way.

Example fully off-shell V-matrix elements



Note that the exchange matrix elements are dependent on the incident energy, while direct matrix elements are not. In the above example the incident energy is 10 eV, and the momenta k' and k take on arbitrary off-shell values.

Some analytical on-shell matrix elements for hydrogen eigenstates

	Direct
$1s$-$1s$	$-\frac{1}{4}\frac{k^2}{k^2+1}-\frac{1}{4}\ln(1+k^2)$
$1s$-$2s$	$\frac{16}{81}\frac{k(4k^2+3)\sqrt{8k^2-6}}{(4k^2+1)^2}$
$1s$-$3s$	$\frac{9\sqrt{3}k(135k^4+87k^2-4)\sqrt{9k^2-8}}{128(9k^2+1)^3}$

	Exchange
$1s$-$1s$	$-\frac{k^2(k^2-3)}{(k^2+1)^3}$
$1s$-$2s$	$-\frac{16k(16k^4-72k^2+13)\sqrt{8k^2-6}}{9(4k^2+1)^4}$
$1s$-$3s$	$-\frac{9\sqrt{3}k(1701k^6-8208k^4+2325k^2-70)\sqrt{9k^2-8}}{8(9k^2+1)^5}$

In these formulas, k' has been expressed in terms of the incident k using the analytical energies of the Hydrogen states. For the exchange matrix elements the total energy E is evaluated using the incident energy $k^2/2$ and energy of the hydrogen $1s$ state.

The factor of $\frac{2}{\pi}$ is not included in these formulas, so remember to include it yourself.

Non-uniqueness

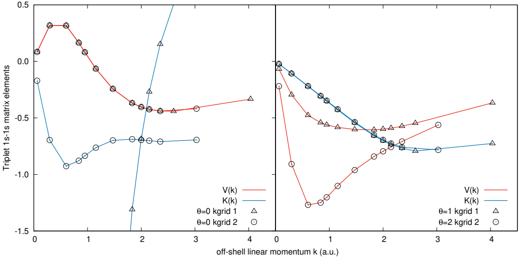
Refer to Igor's lecture notes for details on non-uniqueness and how it is resolved with the "theta trick". For the computational implementation the only change required is to replace the total energy E with

$$E(1-\theta+(-1)^S\theta) \tag{162}$$

in the exchange matrix element. The total partial-wave V-matrix element is then

$$\langle k'\Phi_f|V^S(\theta)|\Phi_i k\rangle = \langle k'\Phi_f|V_1+V_{12}|\Phi_i k\rangle-(-1)^S\langle k'\Phi_f|E(1-\theta+(-1)^S\theta)-H|k\Phi_i\rangle. \tag{163}$$

For singlet scattering ($S=0$) the V-matrix is independent of θ are there is no non-uniqueness issue. For triplet scattering ($S=1$) there *is* non-uniqueness and the V-matrix element *is* dependent on θ.



Left: Half-offshell triplet V and K matrix elements for two k grids (up and down triangles) differing only in the last two points, and $\theta = 0$, 1, and 2.