

Research Notes

Ryan Caulfield

Contents

1	Potentials in QM	3
1.1	Local vs Non-Local Potentials	3
1.2	A Numerical Example of a Non-Local Potential	3
2	Scattering	3
2.1	Scattering Amplitude	3
2.2	Partial Waves	3
2.3	Scattering Matrices	4
3	Nuclear Reactions	4
3.1	Nuclear Optical Potential	4
3.2	A Phenomenological Optical Potential	5
3.3	A Numerical Example of Scattering With an Optical Potential	5
3.4	A Simple Model of a Two Level Nuclei	5
A	Interpolation	7
A.1	Global Interpolation by Polynomials	7
A.2	Piecewise Interpolation and Splines	8
B	Integration	8
B.1	Newton-Cotes Formula	8
B.2	Romberg Integration	8
B.3	Quadrature	8
C	Discrete Fourier Transform	9
D	Derivatives	9
D.1	Finite Difference Schemes	9
D.2	Spectral Derivatives	9

1 Potentials in QM

1.1 Local vs Non-Local Potentials

Often, our first examples in quantum mechanics are systems with local potentials in position space. To be specific, we represent our wave function $|\Psi\rangle$ as a complex valued function of \vec{r} and t , $\Psi(\vec{r}, t)$, and our potential is a function of just the spacial coordinates, $V(\vec{r})$. Then the Schrödinger equation becomes a separable partial differential equation for the unknown function $\Psi(\vec{r}, t)$. The Schrödinger equation becomes the following.

$$i\hbar \frac{\partial}{\partial t} \Psi(\vec{r}, t) = -\frac{\hbar^2}{2m} \nabla^2 \Psi(\vec{r}, t) + V(x) \Psi(\vec{r}, t) \quad (1.1)$$

In this case, the operator \hat{V} applied to $\Psi(\vec{r}, t)$ just yields $V(\vec{r})\Psi(\vec{r})$. The result only depends on the wave function $\Psi(\vec{r}, t)$ at the point \vec{r} . We call this a local potential, as it only depends on the wave function locally at the point \vec{r} .

A more abstract way to define this is to say that a potential \hat{V} is local with respect to a basis if it is diagonal in that basis, i.e. $\langle x | \hat{V} | x' \rangle = 0$ if $x \neq x'$. This need not always be the case. We could have a potential such that $\hat{V}\Psi(\vec{r}, t) = \int V(\vec{r}, \vec{r}') \Psi(\vec{r}', t) d^3\vec{r}'$. In this case, the Schrödinger equation becomes the following.

$$i\hbar \frac{\partial}{\partial t} \Psi(\vec{r}, t) = -\frac{\hbar^2}{2m} \nabla^2 \Psi(\vec{r}, t) + \int V(\vec{r}, \vec{r}') \Psi(\vec{r}', t) d^3\vec{r}' \quad (1.2)$$

This is now an integro-differential equation, which is more complicated to solve. One method of solving this is to discretize the system and solve it numerically. This is practical in one spacial dimension or when the potential only depends on the magnitude of the spacial coordinate. When this is true, it can be recast as a matrix equation. Unfortunately, the matrix will be dense, making these methods slow.

1.2 A Numerical Example of a Non-Local Potential

Find bound states of a non-local nuclear potential. Include SO-coupling and coulomb repulsion. Use wood-saxon potentials like in Bertulani chapter 10 and the paper by Cho et al.

2 Scattering

Brief intro

2.1 Scattering Amplitude

Define scattering amplitude

2.2 Partial Waves

Explain partial waves and phase shifts. Show how to reconstruct scattering amplitude from phase shifts.

2.3 Scattering Matrices

Define S, T and R/K matrices. Show how to solve for them and get phase shifts.

3 Nuclear Reactions

Basics about nuclear reactions

3.1 Nuclear Optical Potential

Suppose we have a single nucleon scattering off of a nucleus with N nucleons. We can describe this by a Hamiltonian \mathbf{H} .

$$\mathbf{H} = \mathbf{T}_0 + \mathbf{V}(r_0, r_1, \dots, r_N) + \mathbf{H}_{nuc}(r_1, r_2, \dots, r_N)$$

\mathbf{T}_0 is the kinetic operator for the projectile, \mathbf{H}_{nuc} is the Hamiltonian which describes the bound particles in the nucleus, and \mathbf{V} is the potential between the projectile and all of the particles in the nucleus. Let $\{\phi_i\}_{i=1}^N$ be the eigenstates of \mathbf{H}_{nuc} with energies $\{\epsilon_i\}$. We want to solve for the eigenstates of the full Hamiltonian, so let's expand the eigenfunction Ψ_E in the complete basis $\{\phi_i\}$.

$$\Psi_E(r_0, r_1, \dots) = \sum_{j=0}^N u_j(r_0) \phi_j(r_1, r_2, \dots)$$

Let's substitute this expansion into the Schrödinger equation, $\mathbf{H}\Psi_E = E\Psi_E$.

$$\sum_{j=0}^N \mathbf{T}_0 u_j \phi_j + \mathbf{V} u_j \phi_j + \mathbf{H}_{nuc} u_j \phi_j = E \sum_{j=0}^N u_j \phi_j$$

Now, we use the fact that $\mathbf{H}_{nuc} u_j \phi_j = u_j \mathbf{H}_{nuc} \phi_j = \epsilon_j u_j \phi_j$

$$\sum_{j=0}^N (\mathbf{T}_0 + \mathbf{V} + \epsilon_j) u_j \phi_j = E \sum_{j=0}^N u_j \phi_j$$

Since the ϕ_j 's are orthogonal, we can multiply both sides by ϕ_i on both sides and integrate over the variables (r_1, r_2, \dots) to obtain the following.

Note: $\langle \phi_i | \mathbf{V} u_j | \phi_j \rangle = \langle \phi_i | \mathbf{V} | \phi_j \rangle u_j = V_{ij} u_j$

$$\sum_{j=0}^N (\mathbf{T}_0 \delta_{ij} + V_{ij} + \epsilon_j \delta_{ij}) u_i = E \sum_{j=0}^N u_j \delta_{ij}$$

$$(\mathbf{T}_0 + \epsilon_i) u_i + \sum_{j=0}^N V_{ij} u_j = E u_i$$

For simplification, we introduce the following notation.

$$\Phi = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \end{pmatrix}$$

$$\mathbf{V}_0 = (V_{01} \quad V_{02} \quad \dots)$$

$$H_{ij} = \mathbf{T}_0 \delta_{ij} + V_{ij} + \epsilon_j \delta_{ij}$$

Looking at the $i = 0$ case and $i \neq 0$ case, we have the following two equations.

$$(\mathbf{T}_0 + V_{00} + \epsilon_0 - E)u_0 = -\mathbf{V}_0 \Phi \quad (3.1)$$

$$(\mathbf{H} - E)\Phi = -\mathbf{V}_0^\dagger u_0 \quad (3.2)$$

We can eliminate Φ in this expression and get the following equation for u_0 .

$$(\mathbf{T}_0 + V_{00}(r_0) + \mathbf{V}_0 \frac{1}{E + i\eta - \mathbf{H}} \mathbf{V}_0^\dagger)u_0(r_0) = (E - \epsilon_0)u_0(r_0) \quad (3.3)$$

We now have a single variable time-independent Schrödinger equation for the unknown $u_0(r_0)$, with a modified potential which we will call the optical potential, V_{opt} .

$$V_{opt}(r_0) = V_{00}(r_0) + \mathbf{V}_0 \frac{1}{E + i\eta - \mathbf{H}} \mathbf{V}_0^\dagger \quad (3.4)$$

This is, in general, an energy dependent, complex and non-local potential when working with outgoing spherical wave boundary conditions. This is not a simple quantity to calculate for a given \mathbf{V} , and the non-locality of V_{opt} makes it difficult to solve equation (3.3).

3.2 A Phenomenological Optical Potential

This should give an explanation like whats in bertulani.

3.3 A Numerical Example of Scattering With an Optical Potential

This should involve code and plots.

3.4 A Simple Model of a Two Level Nuclei

Here we will explore a simple system in which we can find \mathbf{V}_{opt} and $\{u_i\}$ exactly. Suppose our nucleus was just a simple two level system. We can describe \mathbf{H}_{nuc} and $\{\phi_i\}$ as follows.

$$\mathbf{H}_{nuc} = \begin{pmatrix} \epsilon_0 & 0 \\ 0 & \epsilon_1 \end{pmatrix}$$

$$\phi_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

$$\phi_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

We will work in the basis of ϕ_0, ϕ_1 and the momentum basis for the scatterer p . Let's imagine the simplest potential we can which couples the two states ϕ_0 and ϕ_1 .

$$\mathbf{V} = \begin{pmatrix} 0 & V_0 \\ V_0^* & 0 \end{pmatrix}$$

From scratch

Let's expand Ψ_E in the basis $\{\phi_i\}$.

$$\Psi_E(p) = u_0(p) \begin{pmatrix} 1 \\ 0 \end{pmatrix} + u_1(p) \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} u_0(p) \\ u_1(p) \end{pmatrix}$$

Substituting this expansion into the Schrödinger equation, we get the following.

$$\begin{aligned} \mathbf{T}_0 \begin{pmatrix} u_0(p) \\ u_1(p) \end{pmatrix} + \begin{pmatrix} 0 & V_0 \\ V_0^* & 0 \end{pmatrix} \begin{pmatrix} u_0(p) \\ u_1(p) \end{pmatrix} + \begin{pmatrix} \epsilon_0 & 0 \\ 0 & \epsilon_1 \end{pmatrix} \begin{pmatrix} u_0(p) \\ u_1(p) \end{pmatrix} &= E \begin{pmatrix} u_0(p) \\ u_1(p) \end{pmatrix} \\ \begin{pmatrix} \mathbf{T}_0 u_0(p) + V_0 u_1(p) + \epsilon_0 u_0(p) & \\ \mathbf{T}_0 u_1(p) + V_0^* u_0(p) + \epsilon_1 u_1(p) & \end{pmatrix} &= E \begin{pmatrix} u_0(p) \\ u_1(p) \end{pmatrix} \end{aligned}$$

Solving the second equation for u_1 and substituting into the second equation we get the following.

$$\begin{aligned} (\mathbf{T}_0 + V_0 \frac{1}{E + i\eta - \epsilon_1 - \mathbf{T}_0} V_0^* + \epsilon_0) u_0(p) &= E u_0(p) \\ (p^2 + \frac{|V_0|^2}{E + i\eta - \epsilon_1 - p^2}) u_0(p) &= (E - \epsilon_0) u_0(p) \\ V_{opt}(p) &= \frac{|V_0|^2}{E + i\eta - \epsilon_1 - p^2} \end{aligned}$$

Just evaluate 3.4

In the notation of equation 3.4 we have the following.

$$\begin{aligned} \mathbf{V}_0 &= \langle 0 | \mathbf{V} | 1 \rangle = V_0 \\ \mathbf{V}_0^\dagger &= \langle 0 | \mathbf{V} | 1 \rangle^* = V_0^* \\ \mathbf{H} = H_{11} &= \mathbf{T}_0 + \mathbf{V}_{11} + \epsilon_1 = p^2 + \epsilon_1 \\ V_{00} &= \langle 0 | \mathbf{V} | 0 \rangle = 0 \end{aligned}$$

We can evaluate equation 3.4.

$$V_{opt}(p) = \frac{|V_0|^2}{E + i\eta - \epsilon_1 - p^2}$$

A Interpolation

To be able to do Calculus, we must be able to represent a function numerically. We can of course define a function by a rule, but this is not useful when we are trying to represent a function that we do not know explicitly. Often, we only know that a function satisfies some differential equation or recurrence relation, and evaluating it at a point may be difficult. To solve this, we represent a function by sampling the function at a finite number of points. We can then evaluate the function at other points using interpolation.

A.1 Global Interpolation by Polynomials

First, we can sample the function at a sequence of points, and approximate it by the unique polynomial which agrees with the function at each points. Let's consider a function $f(x) : [a, b] \rightarrow \mathbb{R}$. Let x_i be a sequence of points in $[a, b]$ such that $x_0 = a$ and $x_N = b$. Now, we can approximate $f(x)$ by the unique polynomial which agrees with $f(x)$ at each point x_i . To find such a polynomial, we need to find coefficients a_j such that for each i ,

$$a_0 + a_1x_i + a_2x_i^2 + \dots + a_Nx_i^N = f(x_i)$$

We can convert this to a matrix equation $\mathbf{X}\vec{\mathbf{a}} = \vec{\mathbf{f}}$, where $\vec{\mathbf{a}}$ is the vector of coefficients, $\vec{\mathbf{f}}$ is the vector of function values such that $f_i = f(x_i)$, and $X_{ij} = x_i^j$. Now, we can find the coefficients easily by solving the matrix equation. The solution is given by

$$\vec{\mathbf{a}} = \mathbf{X}^{-1}\vec{\mathbf{f}}$$

With the coefficients, we can evaluate $f(x)$ at any $x \in [a, b]$ by

$$f(x) = \sum_{i=0}^N a_i x^i$$

As constructed, this method of interpolation should approximate an Nth degree polynomial exactly. But, how accurately does it approximate non-polynomial functions? The Stone-Weierstrass theorem tells us that every continuous function can be uniformly approximated as a polynomial. One would assume that as the number of sample points increases, the method should converge uniformly to the function. It turns out that this is not true in general. In particular, choosing x_i to be evenly spaced and a function whose Taylor series has a finite radius of convergence this method fails spectacularly.

Add a plot here showing runge phenomena and a short description

We can mitigate this problem by using points which have an asymptotic distribution in $[0, 1]$ given by

$$\rho(x) = \frac{N}{\pi\sqrt{1-x^2}}$$

Often, we satisfy this condition using roots or extrema of certain sets of orthogonal polynomials such as Legendre or Chebyshev polynomials. These special polynomials turn out to be very useful for quadrature and derivatives, and they will be explained further in later sections.

Add a plot here showing the same function used previously but interpolated with Chebyshev nodes and a short description

A.2 Piecewise Interpolation and Splines

Add stuff about stuff.

B Integration

We now want to calculate definite integrals of functions represented numerically. One way of doing this, is to decide on an interpolation scheme, and integrate the interpolant exactly. This can often be done easily when using a polynomial interpolant. What we would like is a rule as follows.

$$\int_a^b f(x) = \sum_{i=0}^{N-1} f(x_i) w_i$$

The challenge is finding the right values of w_i .

B.1 Newton-Cotes Formula

The simplest method for integration is to use piecewise linear interpolation, since each region $[x_i, x_{i+1}]$ can be integrated exactly using the geometric formula for a trapezoid. The method is appropriately named the trapezoid rule.

$$\int_a^b f(x) = \frac{f(x_0)+f(x_1)}{2}h + \frac{f(x_1)+f(x_2)}{2}h + \dots$$

This simplifies to the following.

$$\int_a^b f(x) = \frac{h}{2}(f(x_0) + f(x_N) + 2 \sum_{i=1}^{N-1} f(x_i))$$

It is easy to see using the Taylor expansion of $f(x)$ that the error in this method is $\mathcal{O}(N^{-2})$.

add a plot for this

Similar rules can be derived by interpolating the function using higher order polynomials. For example, interpolating by a quadratic yields Simpson's rule. (note: N should be even)

$$\int_a^b f(x) = \frac{h}{3}(f(x_0) + f(x_N) + 4 \sum_{i=1}^{\frac{N}{2}} f(x_{2i-1}) + 2 \sum_{i=1}^{\frac{N-1}{2}} f(x_{2i}))$$

B.2 Romberg Integration

Add stuff about stuff.

B.3 Quadrature

Add stuff about stuff.

C Discrete Fourier Transform

Add stuff about stuff

D Derivatives

Now that we are able to represent functions, we want to differentiate them. This is a problem if we are representing a function as an array of sample points as the standard definition of a derivative requires taking a limit. Instead, we can decide on an interpolation scheme, and then take the derivative of the interpolant to be the derivative of our function. We can then evaluate this derivative at the same sample points to create a new array which represents the derivative.

D.1 Finite Difference Schemes

Consider as an example, interpolating by piecewise linear functions. Between each point x_i and it's adjacent point x_{i+1} , we have

$$f(x) = f(x_i) + \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} x$$

We can then set the derivative to

$$f'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$$

This can be expressed compactly as

$$\frac{d\vec{f}}{dx} = \mathbf{D}\vec{f}$$

With \mathbf{D} given by

$$\mathbf{D} = \begin{bmatrix} \frac{1}{x_0 - x_1} & -\frac{1}{x_0 - x_1} & \cdots & \cdots & \cdots \\ 0 & \frac{1}{x_1 - x_2} & -\frac{1}{x_1 - x_2} & \cdots & \cdots \\ \vdots & 0 & \frac{1}{x_2 - x_3} & -\frac{1}{x_2 - x_3} & \cdots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}$$

The accuracy of a method such as this depends on the choice of x_i 's and the particular interpolation technique. The above is an example of a finite difference scheme. In the case of equal spacing, \mathbf{D} takes on a simple form.

$$\mathbf{D} = \frac{1}{h} \begin{bmatrix} 1 & -1 & \cdots & \cdots & \cdots \\ 0 & 1 & -1 & \cdots & \cdots \\ \vdots & 0 & 1 & -1 & \cdots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{bmatrix}$$

D.2 Spectral Derivatives

Add stuff about stuff.