HW1

September 12, 2022

1 Finding The Transfer Matrix

[ ]:
```
import matplotlib.pyplot as plt
import numpy as np

plt.style.use("seaborn")
plt.rcParams['figure.dpi'] = 150
plt.rcParams['figure.figsize'] = (8, 3)
```

1.1 Stating of The Problem

We have two non-trivial potentials, and we are asked to find the the transmission coefficients for both of them. The two reflected potentials are given by

$$V_{\pm}(x) = \pm\frac{\hbar^2}{2ma^2}\frac{n(n+1)}{\cosh^2(x/a)}$$

1.2 The Strategy

We can use the power of transfer matrix; it's more like a black box. We input the coefficients of the incident wave, then we get the coefficients of the outgoing wave function. But the issue is that we need to break the potential down into smaller pieces.

1.3 The Schrödinger Equation

The Schrödinger equation, then, reduces to

$$\frac{d^2\psi}{dx^2} + \left[k^2 + \frac{n(n+1)}{a^2\cosh^2(x/a)}\right]\psi = 0 \qquad E = \frac{\hbar^2 k^2}{2m} > 0$$

We can assume $\hbar = \omega = 1$, and $ka = 1$

The case that we are working on here is the scattering state wave function. We are asked to consider an incident energy $E < |V_{max}|$.

1.4 The Scatter Solution $E > 0$

Starting from the TISE

$$-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2} + V\psi = E\psi$$

We substitute the potential (let's try V_+ first)

$$-\frac{\hbar^2}{2m}\frac{d^2\psi}{dx^2} + \left[\frac{\hbar^2}{2ma^2}\frac{n(n+1)}{\cosh^2(x/a)}\right]\psi = E\psi$$

But we divide the potential $V(x)$ into n constant pieces, as

$$V \approx V(x_1) + \cdots + V(x_i) + \cdots + V(x_n)$$

Then we get

$$\frac{d^2\psi}{dx^2} + k^2\psi = 0 \qquad k \equiv \frac{\sqrt{2m(E-V_i)}}{\hbar}$$

Note that k can be real in some regions and not in the others. The general solution takes the form

$$\psi(x) = c_1 e^{ikx} + c_2 e^{-ikx}$$

or by indicating their traveling direction and the x positions

$$\psi_i(x) = \psi_{+i}(x) + \psi_{-i}(x)$$

- $\psi_i(x)$ is the wave function with $V = V(x_1)$. - $\psi_{+i}(x)$ is the wave function traveling to the right. - $\psi_{-i}(x)$ is the wave function traveling to the left.

1.4.1 Transfer Matrix

We are willing to find a matrix M that applies to the incident wave function ψ_1 to give us the outgoing wave function ψ_n, as

$$\psi_n = M\psi_1$$

The definition is given as

$$M(x_n, x_1) = M_s(k_0, k_n)\, M_0(k_n, L_n)\, M_s(k_n, k_{n-1}) \ldots \ldots M_0(k_2, L_2)\, M_s(k_2, k_1)\, M_0(k_1, L_1)\, M_s(k_1, K_0)$$

where

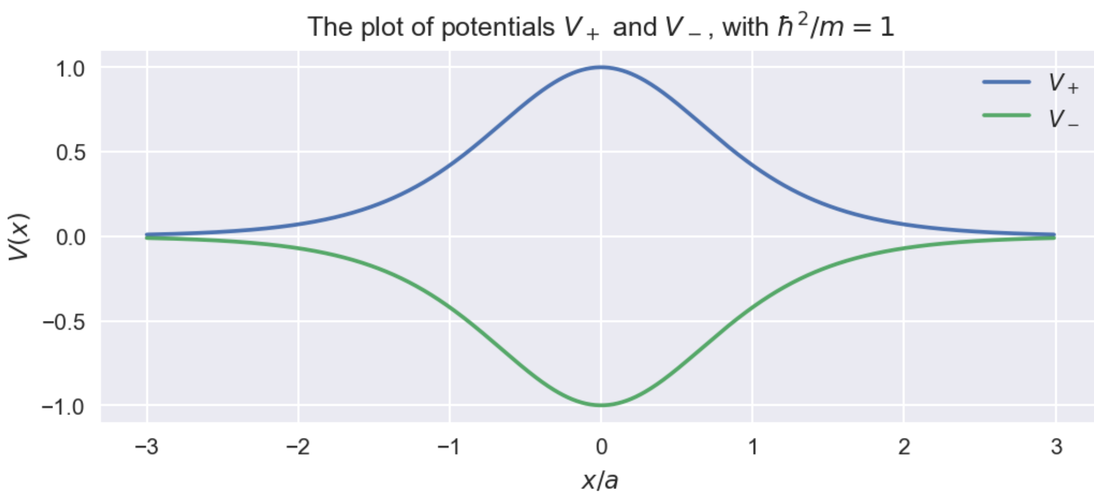
$$M_0(k, L) = \begin{bmatrix} e^{ikL} & 0 \\ 0 & e^{-ikL} \end{bmatrix} \qquad M_s(k_+, k_-) = \frac{1}{2}\begin{bmatrix} 1+\frac{k_-}{k_+} & 1-\frac{k_-}{k_+} \\ 1-\frac{k_-}{k_+} & 1+\frac{k_-}{k_+} \end{bmatrix} \qquad k_i = \frac{\sqrt{2m(E-V_i)}}{\hbar}$$

and L is width of each segment; in our problem it's constant for all segments.

1.5 A Glance at The Potentials

[ ]:
```
n = 1
a = 1
x = np.arange(-3, 3, 0.01)

def potential(x, a, n, sign):
    return sign * n*(n+1) / (np.cosh(x/a) / a)**2 / 2
```

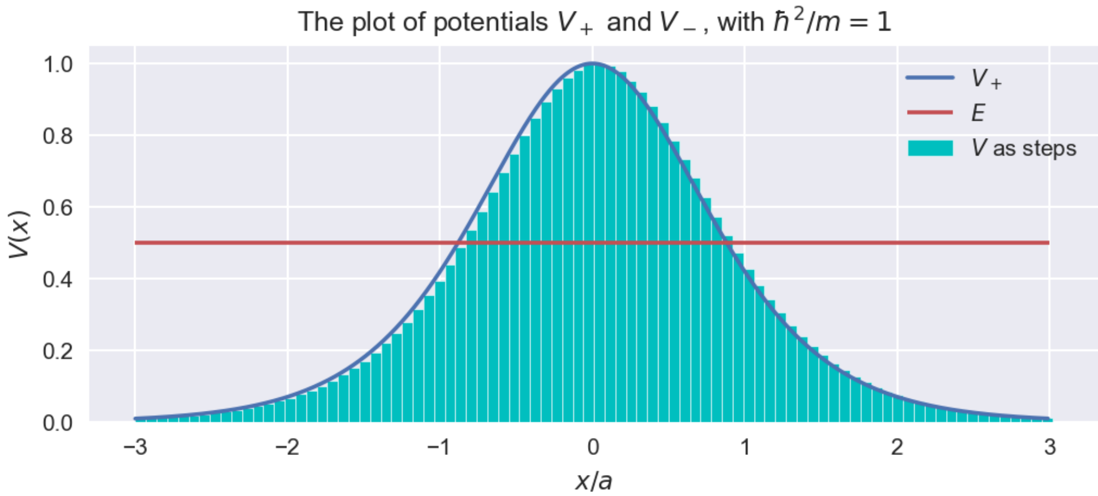
```
plt.plot(x/a, potential(x, a, n, sign=1), label='$V_+$')
plt.plot(x/a, potential(x, a, n, sign=-1), label='$V_-$')
plt.title('The plot of potentials $V_+$ and $V_-$, with $\hbar^2/m = 1$')
plt.legend()
plt.ylabel('$V(x)$')
plt.xlabel('$x/a$')
plt.show()
```



1.6 1. $V_+(x)$

[ ]:
```
E = 0.5
width = 0.07
x_limit = 3
intervals = np.arange(-x_limit, x_limit, width)
x = np.arange(-x_limit, x_limit, 0.01)

plt.plot(x / a, potential(x, a, n, sign=1), label="$V_+$",)
plt.bar(
    intervals,
    potential(intervals, a, n, sign=1),
    width=width,
    color="c",
    align='edge',
    edgecolor="w",
    label='$V$ as steps',
)
plt.hlines(E, -x_limit, x_limit, colors=['C2'], label='$E$')
plt.title("The plot of potentials $V_+$ and $V_-$, with $\hbar^2/m = 1$")
plt.legend()
```

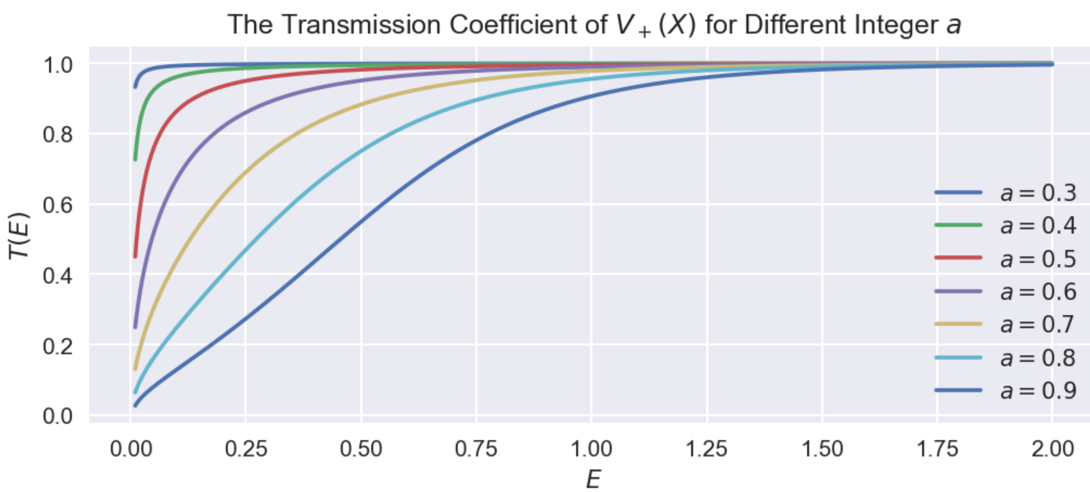
```
plt.ylabel("$V(x)$")
plt.xlabel("$x/a$")
plt.show()
```




```
[ ]: def k(i, E, a, sign):
         hbar = m = 1
         V = potential(intervals[i], a, n, sign=sign)
         return np.sqrt(complex(
             2*m*(E-V)
         )) / hbar

     def transfer_matrix(E, a, sign):
         result = np.identity(2)
         L = width
         for i in range(1, len(intervals)):
             k_neg = k(i-1, E, a, sign)
             k_pos = k(i, E, a, sign)
             M0 = np.matrix([
                 [np.exp(1j*k(i, E, a, sign)*L), 0],
                 [0                  , np.exp(-1j*k(i, E, a, sign)*L)]
             ])
             Ms = 0.5 * np.matrix([
                 [1+k_neg/k_pos, 1-k_neg/k_pos],
                 [1-k_neg/k_pos, 1+k_neg/k_pos]
             ])
             result = M0 @ Ms @ result

         return result
```

```
def transmission(E, a, sign):
    M = transfer_matrix(E, a, sign)
    return abs(np.linalg.det(M) / M[1,1])
```

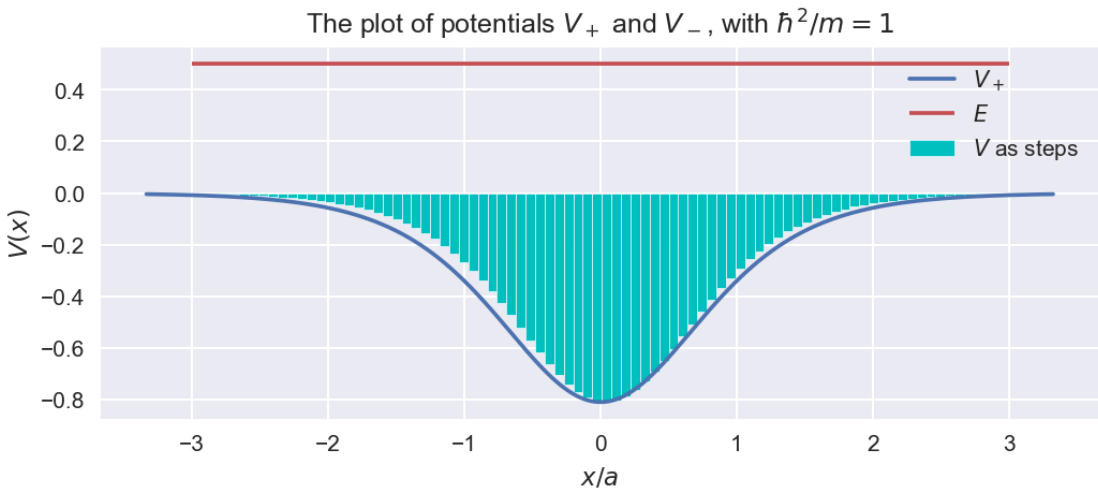
[ ]:
```
E_range = np.arange(0.01, 2, 0.001)

for a_ in np.arange(0.3, 1, 0.1):
    plt.plot(E_range, [transmission(E, a_, sign=+1) for E in E_range],␣
 ↪label=f'$a={round(a_, 1)}$')
plt.title('The Transmission Coefficient of $V_+(X)$ for Different Integer $a$')
plt.xlabel('$E$')
plt.ylabel('$T(E)$')
plt.legend()
plt.show()
```



1.7 2. $V\_(x)$

[ ]:
```
plt.plot(x / a_, potential(x, a_, n, sign=-1), label="$V_+$",)
plt.bar(
    intervals,
    potential(intervals, a_, n, sign=-1),
    width=width,
    color="c",
    align='edge',
    edgecolor="w",
    label='$V$ as steps',
)
plt.hlines(E, -x_limit, x_limit, colors=['C2'], label='$E$')
plt.title("The plot of potentials $V_+$ and $V_-$, with $\hbar^2/m = 1$")
```

```
plt.legend()
plt.ylabel("$V(x)$")
plt.xlabel("$x/a$")
plt.show()
```




```
[ ]: E_range = np.arange(0.01, 2, 0.001)

     for a_ in np.arange(0.3, 1, 0.1):
         plt.plot(E_range, [transmission(E, a_, sign=-1) for E in E_range],␣
      ↪label=f'$a={round(a_, 1)}$')
     plt.title('The Transmission Coefficient of $V_-(X)$ for Different Integer $a$')
     plt.xlabel('$E$')
     plt.ylabel('$T(E)$')
     plt.legend()
     plt.show()
```