ECE 462 - Homework #5

Collin Heist

February 19, 2020

1 Problem 4.2.1

The Hamiltonian matrix can be used to directly find the eigenvalues and eigenfunctions, these are time-domain functions and therefore can be simulated accordingly.

2 Problem 4.3.2

2.1 Part (a)

$$|\sigma_y - \lambda I| = det(\begin{bmatrix} -\lambda & \frac{-i}{2} \\ \frac{i}{2} & -\lambda \end{bmatrix})$$

$$\lambda_{1,2} = \pm\frac{1}{2}$$

With these corresponding eigenvectors:

$$= \sigma_y \cdot \overrightarrow{\lambda_{1,2}} = \lambda_{1,2} \cdot \overrightarrow{\lambda_{1,2}}$$

$$\overrightarrow{\lambda_1} = \begin{bmatrix} i \\ 1 \end{bmatrix}$$

$$\overrightarrow{\lambda_2} = \begin{bmatrix} -i \\ 1 \end{bmatrix}$$

2.2 Part (b)

To show that the eigenfunctions are orthonormal, I'll verify the following is true:

$$\overrightarrow{\lambda_1^{+}} \cdot \overrightarrow{\lambda_2} = 0$$

$$\begin{bmatrix} -i & 1 \end{bmatrix} \cdot \begin{bmatrix} -i \\ 1 \end{bmatrix} = -1 + 1 = 0$$

Since the inner product of the eigen vectors is equal to zero, the functions are orthonormal.

2.3 Part (c)

$$V \cdot \sigma_y = [\overrightarrow{\lambda_1}, \overrightarrow{\lambda_2}]$$

$$V = \begin{bmatrix} 2 & -2 \\ 2i & -2i \end{bmatrix}$$

3 Problem 4.3.3

3.1 Part (a)

It can clearly be seen that the matrix's complex transposition is identical to the original matrix, therefore the matrix *is* Hermitian.

3.2 Part (b)

```
In [26]: import numpy as np
         import pandas as pd
         A = [[2, 1, 0-1j], [1, 3, 0+1j], [0+1j, 0-1j, 2]]

         eigen_values, eigen_vectors = np.linalg.eig(A)
         print ("Eigenvalues: \n{}\nEigenvectors:\n{}".format(eigen_values, eigen_vectors))
```

```
Eigenvalues:
[0.27+0.j 3.  +0.j 3.73+0.j]
Eigenvectors:
[[-0.00e+00+0.63j  7.07e-01+0.j    3.25e-01+0.j   ]
 [-0.00e+00-0.46j -9.83e-17+0.j    8.88e-01+0.j   ]
 [ 6.28e-01+0.j    0.00e+00+0.71j  0.00e+00-0.33j]]
```

```
Out[26]: array([[-7.07e-01+6.13e-01j,  7.07e-01-3.03e-01j, -2.96e-01-7.07e-01j],
                [ 9.83e-17-1.95e+00j, -9.83e-17+1.35e+00j,  2.09e+00+9.83e-17j],
                [ 6.13e-01-7.07e-01j, -3.03e-01+7.07e-01j,  7.07e-01+2.96e-01j]])
```

3.3 Part (c)

$$VA = [\overrightarrow{\lambda}] \rightarrow V = [\overrightarrow{\lambda}]A^{-1}$$

Thus the matrix that transforms A to the eigenfunction representation is:

```
In [27]: eigen_vectors @ np.linalg.inv(A)
```

```
Out[27]: array([[-7.07e-01+6.13e-01j,  7.07e-01-3.03e-01j, -2.96e-01-7.07e-01j],
                [ 9.83e-17-1.95e+00j, -9.83e-17+1.35e+00j,  2.09e+00+9.83e-17j],
                [ 6.13e-01-7.07e-01j, -3.03e-01+7.07e-01j,  7.07e-01+2.96e-01j]])
```

4 Problem 4.3.4

$$M = \begin{bmatrix} a_1 b_1 & a_1 b_2 \\ a_2 b_1 & a_2 b_2 \end{bmatrix}$$