oqcg

February 16, 2019

1 Imports

Need to have jate.py in your folder

```
In [1]: %run jate.py #will import everything
```

2 Next chapter

2.1 memory clear (uses regex, so be careful)

```
In [2]: %reset_selective -f var1, var2  # replace var1, var2 with your defined ones
```

2.2 Building parts

2.2.1 Building the things to be calculated only once

```
In [3]: def maker(omega_1, H_0, H_1, T_s, Lin, d=2, gamma=0.1):
            r"""maker
            Makes all the things that remain constant throught the program, but are
            repeatedly used.


            Parameters
            ----------
            omega_1 : float
                      frequency corresponding to half of the difference between
                      energy levels of the qubit

            H_0     : Qobj
                      Bare Hamiltonian

            H_1     : Qobj
                      Interaction Hamiltonian

            T_s     : Qobj
                      Unitary to be implemented in the Hilbert space
```

```
Lin     : Qobj
          Linbladian operators

d       : int
          Dimension of the matrix. Defaults to 2

gamma   : float
          Damping constant of the Linbladian


Returns
-------


ih0     : Qobj
          $I\otimes H_{0}$

ih1     : Qobj
          $I\otimes H_{1}$

h0ci    : Qobj
          $H_{0}^{*}\otimes I $

h1ci    : Qobj
          $H_{1}^{*}\otimes I $

T       : Qobj
          Target unitary transformed to the Liouville space

linbladian : Qobj
             The full lindbladian term as it appears on transformation to
             the Liouville space.

"""
I = identity(d)
L_I = tensor(I, I)
ih0 = tensor(I, H_0)
ih1 = tensor(I, H_1)
h0ci = tensor(H_0.conj(), I)
h1ci = tensor(H_1.conj(), I)
x_k = ih1 - h1ci
term1 = tensor(Lin.trans(), Lin)
term2 = tensor(I, ((Lin.dag())*(Lin)))
term3 = tensor(((Lin.trans())*(Lin.conj())), I)
lindbladian = 1j*(gamma)*(term1 - 0.5*(term2 + term3))
T = tensor(T_s.trans(), T_s) # Transforming $T_{s}$ to liouville space


return ih0, ih1, h0ci, h1ci, x_k, lindbladian, T, L_I
```

```
In [4]: omega_1 = 0.5
        H_0 = omega_1*sigmaz()
        H_1 = sigmay()
        T_s = sigmax()
        Lin = sigmaz()
        gamma = 0.1 # check for default value
        ih0, ih1, h0ci, h1ci, x_k, lindbladian, T, L_I  = maker(omega_1,
                                                                 H_0, H_1, T_s,
                                                                 Lin, d=2, gamma=gamma)
```

```
In [5]: gamma
```

```
Out[5]: 0.1
```

```
In [6]: L_I
```

Out[6]:

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \end{pmatrix}$$

2.2.2 Building $A(t)$

```
In [7]: def A(xi):
            r"""making $A(t)$"""
            A = ih0 - h0ci + xi*(ih1 - h1ci) + lindbladian
            return A
```

```
In [8]: A(0.5)
```

Out[8]:

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.0 & -0.500j & -0.500j & 0.0 \\ 0.500j & (-1.0-0.200j) & 0.0 & -0.500j \\ 0.500j & 0.0 & (1.0-0.200j) & -0.500j \\ 0.0 & 0.500j & 0.500j & 0.0 \end{pmatrix}$$

2.2.3 Building $L(t)$ and the Identity in the Liouville space

```
In [9]: def L(xi, dt):
            r"""Making $L(t) from $A(t)$"""
            L = (-1j*A(xi)*dt).expm()
            return L
```

```
In [10]: L(0.5, 0.001)
```

Out[10]:

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\left(\begin{array}{cccc}1.000 & (-4.999\times10^{-04} - 2.500\times10^{-07}j) & (-4.999\times10^{-04} + 2.500\times10^{-07}j) & \\ (4.999\times10^{-04} + 2.500\times10^{-07}j) & (1.000 + 9.998\times10^{-04}j) & -2.500\times10^{-07} & (- \\ (4.999\times10^{-04} - 2.500\times10^{-07}j) & -2.500\times10^{-07} & (1.000 - 9.998\times10^{-04}j) & (- \\ 2.500\times10^{-07} & (4.999\times10^{-04} + 2.500\times10^{-07}j) & (4.999\times10^{-04} - 2.500\times10^{-07}j) & \end{array}\right.$$

2.3 Major functions

2.3.1 Major functions 1

```
In [11]: # building the function to optimize (optimizee)
         def L_vec(xi_vec, dt):
             r"""Building the vector of differential $L(t)$"""
             L_vec = [L(xi, dt) for xi in xi_vec]
             return L_vec
```

```
In [12]: def fidelity_calc(A, B):
             r"""Making a generalised fidelity function"""
             first_part = (A - B).dag()
             second_part = (A - B)
             f_int = (first_part* second_part)
             f = f_int.tr()
             return f
```

```
In [13]: def L_full_maker(xi_vec, dt):
             r"""Building the $L(t)$ for the total time $t$"""
             xi_vec_size = xi_vec.size # finding the size of xi
             L_full = L_I # Identity for the for loop of L
             L_v = L_vec(xi_vec, dt) # calling L_vec
             for i in range(xi_vec_size): # generating L_full
                 L_full = L_full*L_v[xi_vec_size - 1 - i]
             return L_full
```

```
In [14]: def F(xi_vec, dt):
             r"""Using the fidelity metric to find out the closeness between $T$
             and $L(t)$"""
             L_full = L_full_maker(xi_vec, dt)
             F = real(-fidelity_calc(T, L_full))
             return F
```

2.3.2 Testing major functions 1

```
In [15]: fidelity_calc(sigmax(), sigmay())
```

```
Out[15]: 4.0
```

```
In [16]: fidelity_calc(sigmay(), sigmay())
```

```
Out[16]: 0.0

In [17]: xi_vec_test = array([1.0, 2.0])
         xi_vec_test

Out[17]: array([1., 2.])

In [18]: xi_vec_test.size

Out[18]: 2

In [19]: w_vec = [xi**2 for xi in xi_vec_test]
         w_vec

Out[19]: [1.0, 4.0]

In [20]: # F(xi_vec, dt)
         F(xi_vec_test, 0.001)

Out[20]: -7.998400634493138

In [21]: L_v = L_vec(xi_vec_test, 0.001)

In [22]: L_v

Out[22]: [Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False
          Qobj data =
          [[ 9.99999000e-01+0.00000000e+00j -9.99899173e-04-4.99933130e-07j
            -9.99899173e-04+4.99933130e-07j  9.99932920e-07+0.00000000e+00j]
           [ 9.99899173e-04+4.99933130e-07j  9.99798520e-01+9.99799187e-04j
            -9.99866260e-07+0.00000000e+00j -9.99899173e-04-4.99933130e-07j]
           [ 9.99899173e-04-4.99933130e-07j -9.99866260e-07+0.00000000e+00j
             9.99798520e-01-9.99799187e-04j -9.99899173e-04+4.99933130e-07j]
           [ 9.99932920e-07+0.00000000e+00j  9.99899173e-04+4.99933130e-07j
             9.99899173e-04-4.99933130e-07j  9.99999000e-01+0.00000000e+00j]],
          Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False
          Qobj data =
          [[ 9.99996000e-01+0.00000000e+00j -1.99979435e-03-9.99865260e-07j
            -1.99979435e-03+9.99865260e-07j  3.99972768e-06+0.00000000e+00j]
           [ 1.99979435e-03+9.99865260e-07j  9.99795521e-01+9.99797187e-04j
            -3.99946104e-06+0.00000000e+00j -1.99979435e-03-9.99865260e-07j]
           [ 1.99979435e-03-9.99865260e-07j -3.99946104e-06+0.00000000e+00j
             9.99795521e-01-9.99797187e-04j -1.99979435e-03+9.99865260e-07j]
           [ 3.99972768e-06+0.00000000e+00j  1.99979435e-03+9.99865260e-07j
             1.99979435e-03-9.99865260e-07j  9.99996000e-01+0.00000000e+00j]]]
```

2.3.3 Major Functions 2

```
In [23]: def L_comma_k_maker(xi_vec, k, dt):
             r"""Making of the derivative of full $L(t)$ at time $t_{k}$"""
             N = xi_vec.size
             # Determining the size of xi, and thus the time_steps indirectly.
             L_v = L_vec(xi_vec, dt)# Making of the full $L(t)$
             inner_part = L_I # Beginner for the for loop
             for i in range(N):
                 if i == ( N - 1 - k ):
                     # The step at which $X_{k}(t)$ has to be inserted
                     inner_part = inner_part*x_k*L_v[k - 1]
                 else:
                     # Usual multiplications of $L_{k}$
                     inner_part = inner_part*L_v[N - 1 - i]
             l_comma_k = inner_part
             return l_comma_k
```

```
In [24]: # L_comma_k_maker(xi_vec, k, dt)
         L_comma_k_maker(xi_vec_test, 2, 0.001)
```

Out[24]:

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\left(\begin{array}{cccc}1.000 & (-0.003-3.499\times10^{-06}j) & (-0.003+3.499\times10^{-06}j) & 8.999\\ (0.003+2.499\times10^{-06}j) & (1.000+0.002j) & (-8.998\times10^{-06}+2.999\times10^{-09}j) & (-0.003-2\\ (0.003-2.499\times10^{-06}j) & (-8.998\times10^{-06}-2.999\times10^{-09}j) & (1.000-0.002j) & (-0.003+2\\ 8.999\times10^{-06} & (0.003+3.499\times10^{-06}j) & (0.003-3.499\times10^{-06}j) & 1\end{array}\right.$$

```
In [25]: def updater(xi_vec, dt, epsilon):
             r"""Implementing the GRAPE update step"""
             xi_vec_size = xi_vec.size # finding the size of xi
             L_full = L_full_maker(xi_vec, dt)
             di = []
             for k in range(xi_vec_size):
                 # Building the thing to be added to the old function
                 L_comma_k = L_comma_k_maker(xi_vec, k, dt)
                 differentiated = T - L_comma_k
                 plain = T - L_full
                 c = -differentiated.dag()*plain
                 d = -plain.dag()*differentiated
                 inside = c.tr() + d.tr()
                 di.append(epsilon*inside)

             diff = array(di)
             xi_new_vec = xi_vec + diff
             return diff, xi_new_vec
```

```
In [26]: #  updater(xi_vec, dt, epsilon)
         updater(xi_vec_test, 0.001, 0.001)

Out[26]: (array([-0.008+0.j, -0.008+0.j]), array([0.992+0.j, 1.992+0.j]))
```

2.4 Qutip grape for closed system

```
In [27]: import time

In [28]: total_time_evo = 2*pi # total time allowed for evolution

In [29]: times = linspace(0, total_time_evo, 500)

In [30]: # vector of times at which discretization
         # is carried out

In [31]: U = T_s
         U
```

Out[31]:
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 0.0 & 1.0 \\ 1.0 & 0.0 \end{pmatrix}$$

```
In [32]: R = 500

In [33]: H_ops = [H_1]
         H_ops

Out[33]: [Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True
          Qobj data =
          [[0.+0.j 0.-1.j]
           [0.+1.j 0.+0.j]]]

In [34]: H_labels = [r'$g_{no diss}$']
         H_labels

Out[34]: ['$g_{no diss}$']

In [35]: H0 = H_0
         H0
```

Out[35]:
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 0.500 & 0.0 \\ 0.0 & -0.500 \end{pmatrix}$$

```
In [36]: c_ops = []
In [37]: from qutip.control.grape import plot_grape_control_fields, _overlap
         from qutip.control.grape import grape_unitary_adaptive, cy_grape_unitary
```

```
In [38]: from scipy.interpolate import interp1d
         from qutip.ui.progressbar import TextProgressBar, EnhancedTextProgressBar
```

```
In [39]: u0 = array([rand(len(times)) * 2 * pi * 0.05 for _ in range(len(H_ops))])
```

```
In [40]: from numpy import convolve
         u0 = [convolve(ones(10)/10, u0[idx,:], mode='same') for idx in range(len(H_ops))]
```

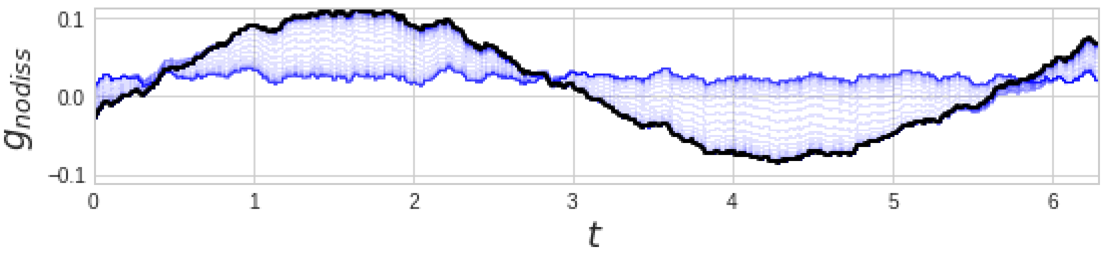
```
In [41]: u_limits = None #[0, 1 * 2 * pi]
         alpha = None
```

```
In [42]: result = cy_grape_unitary(U, H0, H_ops, R, times, u_start=u0, u_limits=u_limits,
                                   eps=2*np.pi*1, alpha=alpha, phase_sensitive=False,
                                   progress_bar=TextProgressBar())
```

```
10.0%. Run time:  35.21s. Est. time left: 00:00:05:16
20.0%. Run time:  68.84s. Est. time left: 00:00:04:35
30.0%. Run time: 102.50s. Est. time left: 00:00:03:59
40.0%. Run time: 135.89s. Est. time left: 00:00:03:23
50.0%. Run time: 169.45s. Est. time left: 00:00:02:49
60.0%. Run time: 202.93s. Est. time left: 00:00:02:15
70.0%. Run time: 236.24s. Est. time left: 00:00:01:41
80.0%. Run time: 269.57s. Est. time left: 00:00:01:07
90.0%. Run time: 302.86s. Est. time left: 00:00:00:33
Total run time: 335.83s
```

Plot of optimized control field without dissipation

```
In [43]: plot_grape_control_fields(times,
                                   result.u / (2 * np.pi), H_labels, uniform_axes=True);
```



```
In [44]: U
```

Out[44]:

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 0.0 & 1.0 \\ 1.0 & 0.0 \end{pmatrix}$$

```
In [45]: result.U_f
```

Out[45]:

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = False

$$\begin{pmatrix} -9.937\times10^{-17} & 1.000j \\ 1.0j & 9.631\times10^{-17} \end{pmatrix}$$

2.5 Analysis of result of qutip grape for closed system

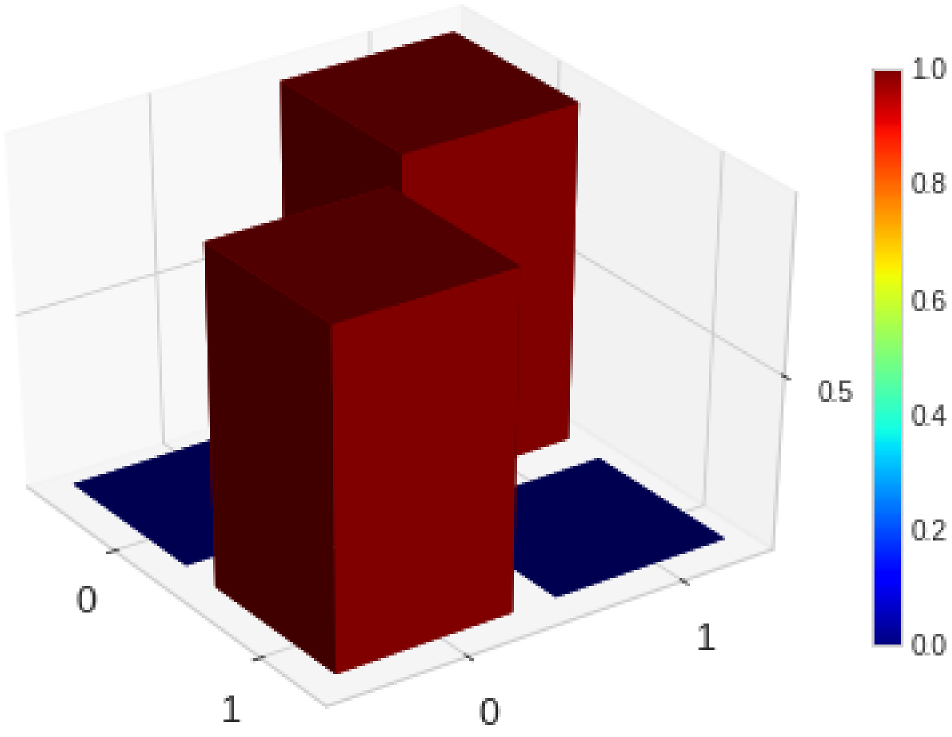
```
In [46]: result.U_f/result.U_f[0,0]
```

Out[46]:

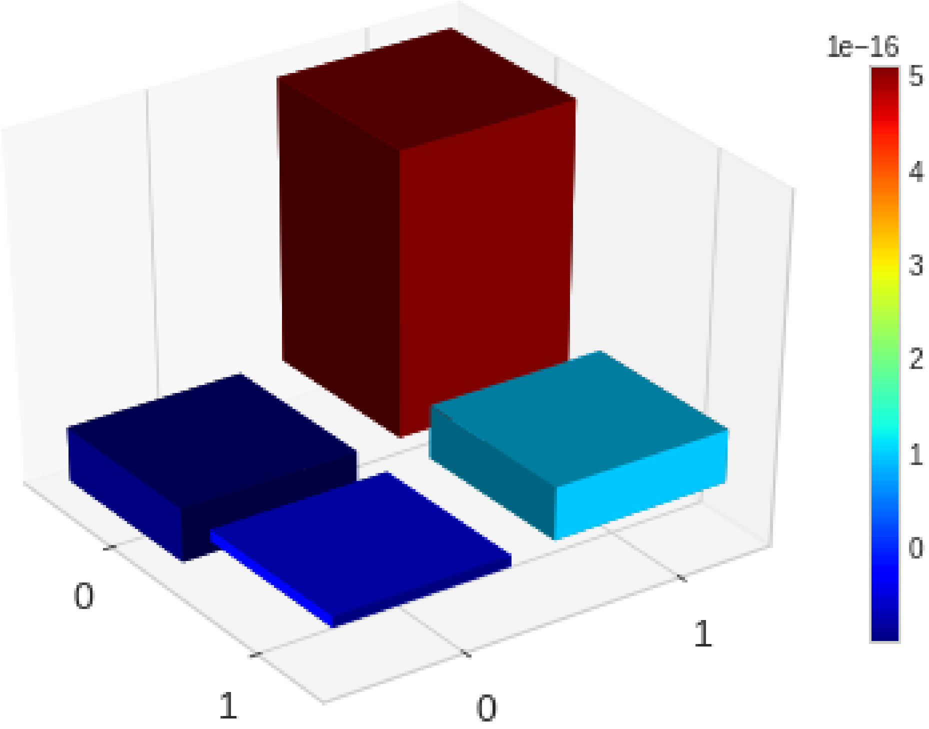
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = False

$$\begin{pmatrix} 1.0 & (-1.828\times10^{+15} - 3.440\times10^{+14}j) \\ (-1.828\times10^{+15} - 3.440\times10^{+14}j) & (0.880 + 0.348j) \end{pmatrix}$$

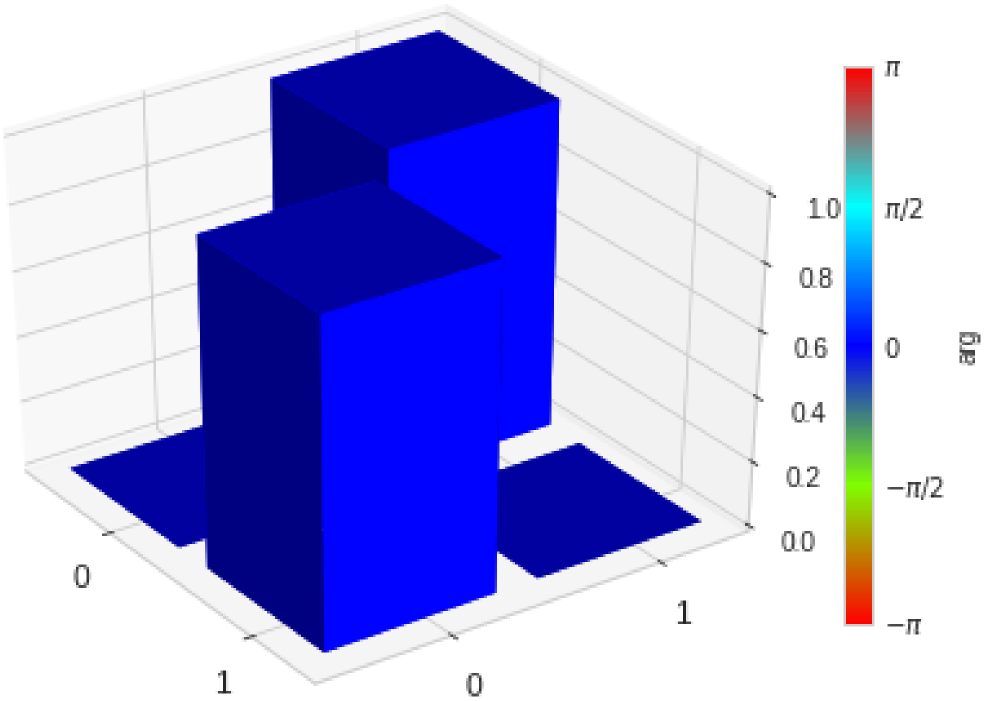
```
In [47]: matrix_histogram(U)

Out[47]: (<Figure size 432x288 with 2 Axes>,
          <mpl_toolkits.mplot3d.axes3d.Axes3D at 0x7fefd050f3c8>)
```



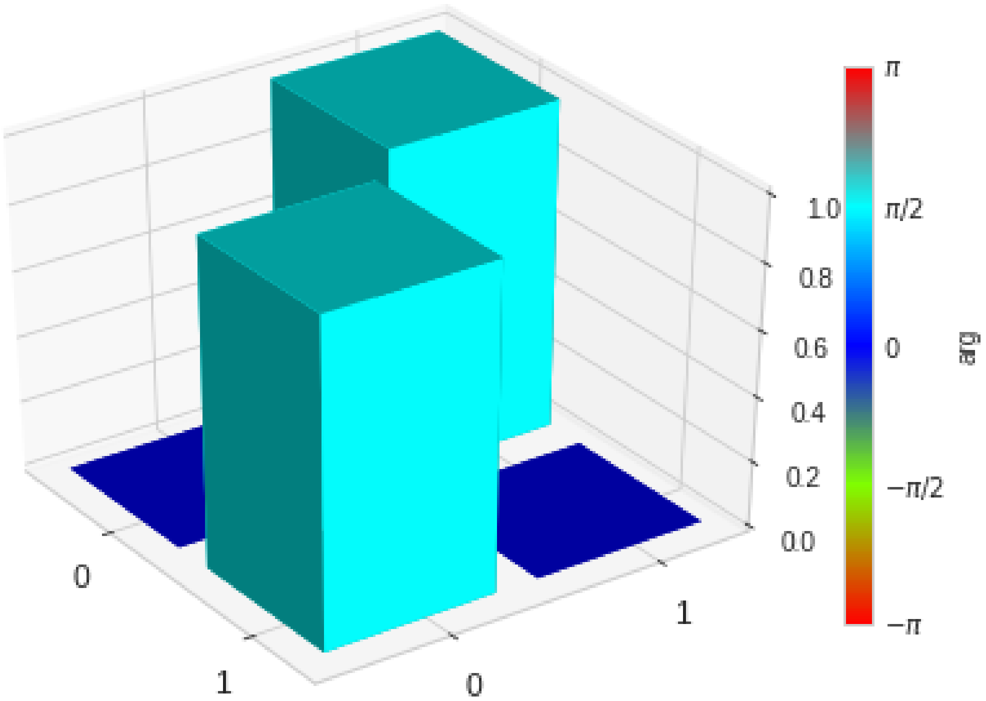
```
In [48]: matrix_histogram(result.U_f)

Out[48]: (<Figure size 432x288 with 2 Axes>,
          <mpl_toolkits.mplot3d.axes3d.Axes3D at 0x7fefd043c240>)
```



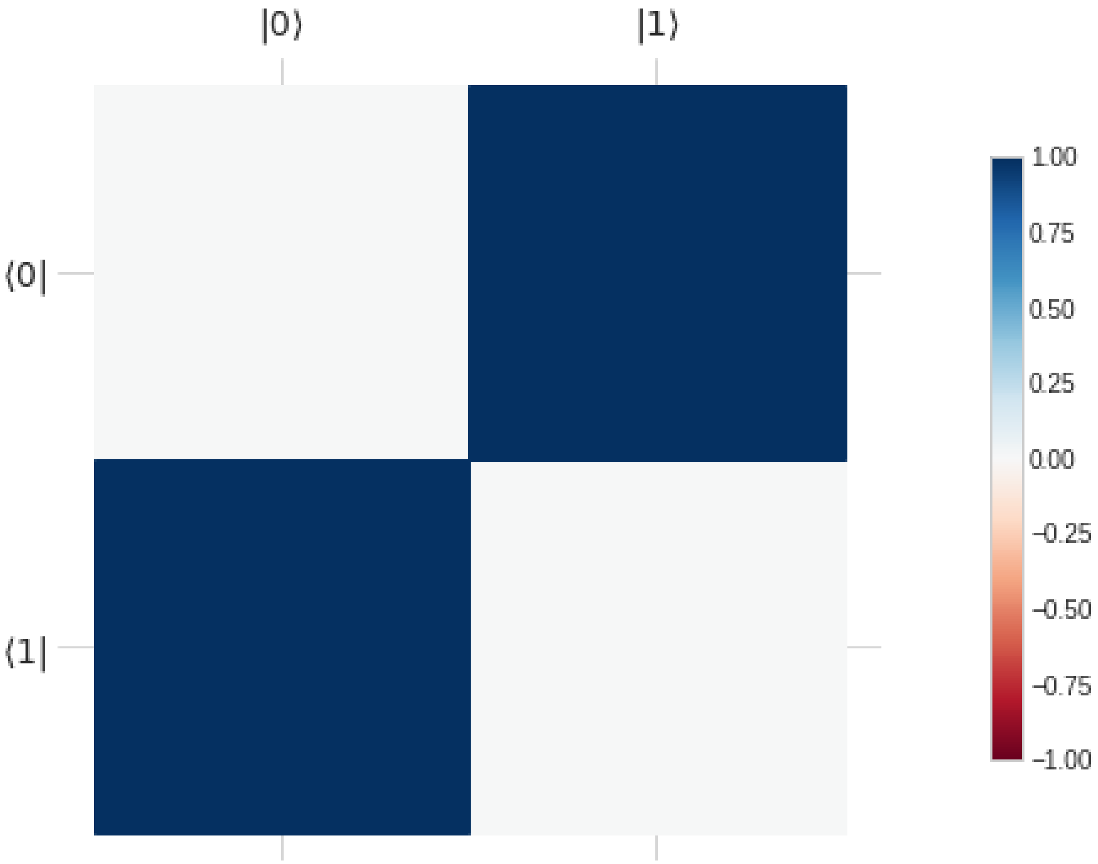
```
In [49]: matrix_histogram_complex(U)

Out[49]: (<Figure size 432x288 with 2 Axes>,
          <mpl_toolkits.mplot3d.axes3d.Axes3D at 0x7fefd03a70b8>)
```



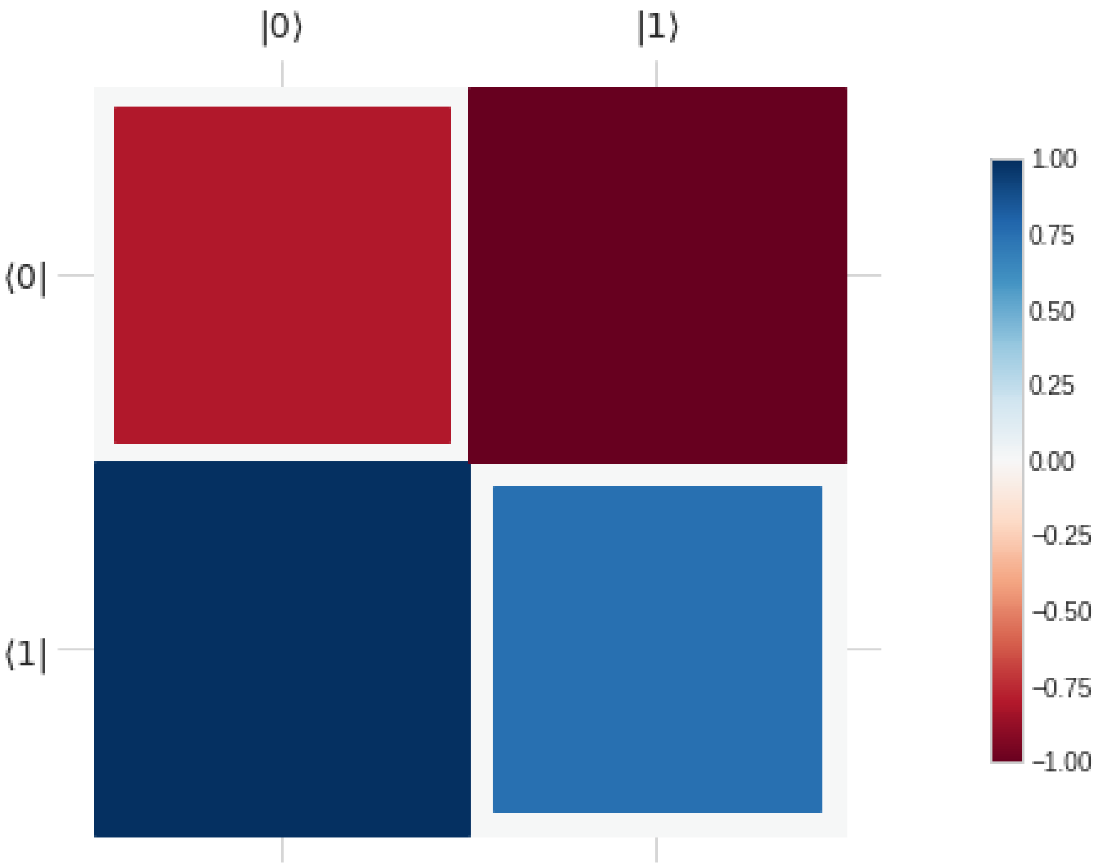
```
In [50]: matrix_histogram_complex(result.U_f)

Out[50]: (<Figure size 432x288 with 2 Axes>,
          <mpl_toolkits.mplot3d.axes3d.Axes3D at 0x7fefd0351278>)
```



```
In [51]: hinton(U)

Out[51]: (<Figure size 576x432 with 2 Axes>,
          <matplotlib.axes._subplots.AxesSubplot at 0x7fefd0276a20>)
```



```
In [52]: hinton(result.U_f)

Out[52]: (<Figure size 576x432 with 2 Axes>,
          <matplotlib.axes._subplots.AxesSubplot at 0x7fefd01c6c50>)
```



```
In [53]: updater(result.u[-1, 0, : ] , (2*pi)/500, epsilon=((0.1*2*pi)/(10**3)))

Out[53]: (array([-0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                 -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j,
                        -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j, -0.00207526+0.j]),
                 array([-0.18981489+0.j, -0.17532648+0.j, -0.14439776+0.j, -0.10824093+0.j,
                        -0.09567925+0.j, -0.06659201+0.j, -0.05155044+0.j, -0.04496856+0.j,
                        -0.05368121+0.j, -0.07263244+0.j, -0.05431784+0.j, -0.05235868+0.j,
                        -0.04252277+0.j, -0.03262759+0.j, -0.00729903+0.j,  0.00634903+0.j,
                         0.00468536+0.j,  0.00513758+0.j,  0.00464709+0.j,  0.0222782 +0.j,
                         0.03112502+0.j,  0.04024616+0.j,  0.05110875+0.j,  0.03338055+0.j,
                         0.02501782+0.j,  0.00757774+0.j,  0.01910582+0.j,  0.0345821 +0.j,
                         0.04931579+0.j,  0.06845434+0.j,  0.07356085+0.j,  0.09481535+0.j,
                         0.09752187+0.j,  0.12735617+0.j,  0.1594223 +0.j,  0.18635566+0.j,
                         0.20741238+0.j,  0.2127336 +0.j,  0.2182719 +0.j,  0.22991645+0.j,
                         0.22171453+0.j,  0.22386559+0.j,  0.21391604+0.j,  0.22508712+0.j,
                         0.23516238+0.j,  0.22820706+0.j,  0.24660838+0.j,  0.25041709+0.j,
                         0.25915332+0.j,  0.25550177+0.j,  0.26695301+0.j,  0.28777818+0.j,
                         0.31561954+0.j,  0.31105956+0.j,  0.30460653+0.j,  0.32838411+0.j,
                         0.33689041+0.j,  0.35886909+0.j,  0.37071472+0.j,  0.37578945+0.j,
                         0.38858389+0.j,  0.39025233+0.j,  0.39307612+0.j,  0.41307595+0.j,
                         0.40643024+0.j,  0.41421651+0.j,  0.40767244+0.j,  0.41419967+0.j,
                         0.43756068+0.j,  0.45977087+0.j,  0.47657047+0.j,  0.4955884 +0.j,
                         0.49683263+0.j,  0.50374738+0.j,  0.52770218+0.j,  0.53742089+0.j,
                         0.53896688+0.j,  0.55419342+0.j,  0.56511304+0.j,  0.56207053+0.j,
                         0.55901499+0.j,  0.56670222+0.j,  0.56804928+0.j,  0.55437803+0.j,
                         0.54600026+0.j,  0.52936125+0.j,  0.5401968 +0.j,  0.52835857+0.j,
                         0.51328568+0.j,  0.51052897+0.j,  0.52592567+0.j,  0.50906907+0.j,
                         0.53135472+0.j,  0.53863585+0.j,  0.54224582+0.j,  0.57518988+0.j,
                         0.58978452+0.j,  0.61021496+0.j,  0.60556076+0.j,  0.61901571+0.j,
                         0.62115889+0.j,  0.64235455+0.j,  0.62434626+0.j,  0.62492177+0.j,
                         0.63940489+0.j,  0.62798486+0.j,  0.63644454+0.j,  0.63582665+0.j,
                         0.6580747 +0.j,  0.66799219+0.j,  0.64966433+0.j,  0.63671267+0.j,
                         0.65773049+0.j,  0.6757297 +0.j,  0.66510398+0.j,  0.67480878+0.j,
                         0.67360209+0.j,  0.67040668+0.j,  0.6655067 +0.j,  0.64475393+0.j,
                         0.65016572+0.j,  0.67338778+0.j,  0.67794346+0.j,  0.66383772+0.j,
                         0.6735167 +0.j,  0.66003833+0.j,  0.636998  +0.j,  0.64153596+0.j,
                         0.65352443+0.j,  0.67916867+0.j,  0.70194022+0.j,  0.69556726+0.j,
                         0.68653746+0.j,  0.68391691+0.j,  0.67962941+0.j,  0.67423764+0.j,
                         0.69368394+0.j,  0.67258774+0.j,  0.67273545+0.j,  0.67315896+0.j,
                         0.6517371 +0.j,  0.65721385+0.j,  0.6663704 +0.j,  0.65943668+0.j,
                         0.6591273 +0.j,  0.6820261 +0.j,  0.68187594+0.j,  0.6767627 +0.j,
                  0.67666127+0.j,  0.65271957+0.j,  0.65719275+0.j,  0.64269157+0.j,
                  0.61799018+0.j,  0.64585746+0.j,  0.64096547+0.j,  0.61059959+0.j,
                  0.58608396+0.j,  0.5828865 +0.j,  0.56566646+0.j,  0.57097945+0.j,
                  0.57005031+0.j,  0.55414373+0.j,  0.55113743+0.j,  0.52995004+0.j,
                  0.53701348+0.j,  0.56019091+0.j,  0.55632425+0.j,  0.55700318+0.j,
                  0.55923798+0.j,  0.56549333+0.j,  0.57183764+0.j,  0.58336759+0.j,
                  0.59904106+0.j,  0.60235593+0.j,  0.59984626+0.j,  0.58725173+0.j,
                  0.60824822+0.j,  0.61737596+0.j,  0.59629448+0.j,  0.57428821+0.j,
                  0.55415343+0.j,  0.54625146+0.j,  0.53344648+0.j,  0.53231955+0.j,
                  0.50845455+0.j,  0.48195091+0.j,  0.44540157+0.j,  0.431681  +0.j,
                  0.43315001+0.j,  0.42375361+0.j,  0.41621515+0.j,  0.3955753 +0.j,
                  0.39640428+0.j,  0.37644841+0.j,  0.39082553+0.j,  0.39521496+0.j,
                  0.41723454+0.j,  0.40546437+0.j,  0.40224657+0.j,  0.40918534+0.j,
                  0.39232748+0.j,  0.38234751+0.j,  0.35156531+0.j,  0.35604805+0.j,
                  0.32827676+0.j,  0.33507236+0.j,  0.30427105+0.j,  0.30450557+0.j,
                  0.30689309+0.j,  0.27978009+0.j,  0.27105834+0.j,  0.25863889+0.j,
                  0.24762204+0.j,  0.21309759+0.j,  0.21591281+0.j,  0.18342081+0.j,
                  0.19755019+0.j,  0.20039123+0.j,  0.18176412+0.j,  0.17447886+0.j,
                  0.17833621+0.j,  0.18445272+0.j,  0.17677694+0.j,  0.17748326+0.j,
                  0.17436188+0.j,  0.16547039+0.j,  0.13095384+0.j,  0.09986995+0.j,
                  0.08378487+0.j,  0.09739656+0.j,  0.09928475+0.j,  0.08361583+0.j,
                  0.09399917+0.j,  0.0821147 +0.j,  0.07285404+0.j,  0.0901184 +0.j,
                  0.0966029 +0.j,  0.10258161+0.j,  0.09233394+0.j,  0.07494531+0.j,
                  0.05082207+0.j,  0.05515701+0.j,  0.04219505+0.j,  0.05339267+0.j,
                  0.04012904+0.j,  0.01383232+0.j, -0.00712591+0.j, -0.01745221+0.j,
                 -0.02624092+0.j, -0.02260088+0.j, -0.02365738+0.j, -0.03455945+0.j,
                 -0.03533082+0.j, -0.0416341 +0.j, -0.06203955+0.j, -0.06997482+0.j,
                 -0.07400132+0.j, -0.09460447+0.j, -0.09705045+0.j, -0.12786286+0.j,
                 -0.12596214+0.j, -0.14412826+0.j, -0.15418246+0.j, -0.1641098 +0.j,
                 -0.1724538 +0.j, -0.16240899+0.j, -0.17430401+0.j, -0.18499652+0.j,
                 -0.18248914+0.j, -0.18834567+0.j, -0.21361626+0.j, -0.21345717+0.j,
                 -0.23175122+0.j, -0.25453091+0.j, -0.23481785+0.j, -0.24346882+0.j,
                 -0.24454802+0.j, -0.2273565 +0.j, -0.23238386+0.j, -0.21868758+0.j,
                 -0.22534731+0.j, -0.23055593+0.j, -0.22992686+0.j, -0.21628585+0.j,
                 -0.22227603+0.j, -0.22823219+0.j, -0.24028802+0.j, -0.26882924+0.j,
                 -0.30038103+0.j, -0.31030821+0.j, -0.31250408+0.j, -0.32376877+0.j,
                 -0.3360826 +0.j, -0.35195222+0.j, -0.35783441+0.j, -0.36177639+0.j,
                 -0.36322666+0.j, -0.36044068+0.j, -0.35642047+0.j, -0.37626743+0.j,
                 -0.38360475+0.j, -0.39863258+0.j, -0.40260139+0.j, -0.40757727+0.j,
                 -0.43508512+0.j, -0.46062144+0.j, -0.46158914+0.j, -0.45860549+0.j,
                 -0.45739458+0.j, -0.45805327+0.j, -0.44543141+0.j, -0.44611099+0.j,
                 -0.4597519 +0.j, -0.46556551+0.j, -0.46845635+0.j, -0.45402347+0.j,
                 -0.44909344+0.j, -0.45986071+0.j, -0.44755705+0.j, -0.44952101+0.j,
                 -0.47978275+0.j, -0.4746718 +0.j, -0.46359934+0.j, -0.47138964+0.j,
                 -0.453233  +0.j, -0.47930815+0.j, -0.47957041+0.j, -0.47160536+0.j,
                 -0.4910898 +0.j, -0.49666723+0.j, -0.48895116+0.j, -0.49631566+0.j,
                 -0.50992145+0.j, -0.50034438+0.j, -0.50063023+0.j, -0.49669364+0.j,
                 -0.5191154 +0.j, -0.52364944+0.j, -0.52884114+0.j, -0.53148841+0.j,
                 -0.54058291+0.j, -0.5175859 +0.j, -0.51908031+0.j, -0.51596762+0.j,
                 -0.53122122+0.j, -0.51429092+0.j, -0.50679759+0.j, -0.50784827+0.j,
                 -0.51053065+0.j, -0.50386027+0.j, -0.48679365+0.j, -0.51366852+0.j,
                 -0.48936605+0.j, -0.4847466 +0.j, -0.4717867 +0.j, -0.47308547+0.j,
                 -0.448743  +0.j, -0.46236233+0.j, -0.44830721+0.j, -0.45245764+0.j,
                 -0.46613791+0.j, -0.44112902+0.j, -0.46318575+0.j, -0.46816873+0.j,
                 -0.47431248+0.j, -0.47075347+0.j, -0.47090423+0.j, -0.46445162+0.j,
                 -0.45531539+0.j, -0.46135392+0.j, -0.45954568+0.j, -0.47870619+0.j,
                 -0.45460867+0.j, -0.46672081+0.j, -0.46381593+0.j, -0.45643221+0.j,
                 -0.4763782 +0.j, -0.46129372+0.j, -0.46007989+0.j, -0.43579596+0.j,
                 -0.40148667+0.j, -0.39174005+0.j, -0.39807442+0.j, -0.39233033+0.j,
                 -0.3877545 +0.j, -0.39226954+0.j, -0.37223244+0.j, -0.35965203+0.j,
                 -0.36091033+0.j, -0.3578557 +0.j, -0.36437633+0.j, -0.35108893+0.j,
                 -0.35607355+0.j, -0.33364665+0.j, -0.33722397+0.j, -0.33484492+0.j,
                 -0.31905242+0.j, -0.31295146+0.j, -0.30349112+0.j, -0.29919257+0.j,
                 -0.29023422+0.j, -0.27636718+0.j, -0.25045532+0.j, -0.24351126+0.j,
                 -0.23495823+0.j, -0.22911451+0.j, -0.22553002+0.j, -0.23473038+0.j,
                 -0.22629084+0.j, -0.21230721+0.j, -0.20516253+0.j, -0.20141265+0.j,
                 -0.20986307+0.j, -0.21305115+0.j, -0.19524946+0.j, -0.18412027+0.j,
                 -0.20009322+0.j, -0.1847023 +0.j, -0.18612701+0.j, -0.19856394+0.j,
                 -0.20870019+0.j, -0.20057487+0.j, -0.17211355+0.j, -0.15266146+0.j,
                 -0.14846749+0.j, -0.15543135+0.j, -0.12555856+0.j, -0.11770019+0.j,
                 -0.10729537+0.j, -0.08631548+0.j, -0.07557817+0.j, -0.0905244 +0.j,
                 -0.08694717+0.j, -0.07980948+0.j, -0.08187849+0.j, -0.07066793+0.j,
                 -0.08680067+0.j, -0.09605075+0.j, -0.09707327+0.j, -0.09537512+0.j,
                 -0.09074836+0.j, -0.07462175+0.j, -0.06276156+0.j, -0.06299608+0.j,
                 -0.04911118+0.j, -0.03409736+0.j, -0.00615456+0.j,  0.02209517+0.j,
                  0.05108223+0.j,  0.06537526+0.j,  0.07100315+0.j,  0.07573659+0.j,
                  0.07504343+0.j,  0.08132914+0.j,  0.09689284+0.j,  0.11331409+0.j,
                  0.10366834+0.j,  0.11598682+0.j,  0.10581205+0.j,  0.09926045+0.j,
                  0.12010182+0.j,  0.13565544+0.j,  0.13851948+0.j,  0.13185644+0.j,
                  0.13570809+0.j,  0.14309554+0.j,  0.16754299+0.j,  0.15015905+0.j,
                  0.16899458+0.j,  0.1742532 +0.j,  0.18322914+0.j,  0.19877095+0.j,
                  0.21635625+0.j,  0.24828109+0.j,  0.24469296+0.j,  0.24112582+0.j,
                  0.25057636+0.j,  0.26814865+0.j,  0.27635856+0.j,  0.30581701+0.j,
                  0.306278  +0.j,  0.30771324+0.j,  0.29597105+0.j,  0.28247674+0.j,
                  0.30050984+0.j,  0.30248706+0.j,  0.31071373+0.j,  0.3346877 +0.j,
                  0.35408918+0.j,  0.35429758+0.j,  0.38406155+0.j,  0.40172216+0.j,
                  0.42902731+0.j,  0.45260232+0.j,  0.44896578+0.j,  0.46993703+0.j,
                  0.45508956+0.j,  0.43671676+0.j,  0.41342787+0.j,  0.41342787+0.j]))
```

```
In [54]: times[-1]
```

```
Out[54]: 6.283185307179586
```

```
In [55]: total_time_evo
```

```
Out[55]: 6.283185307179586
```

2.6 joining qutip to my code

2.7 ##### total_time

NameError Traceback (most recent call last) in () ----> 1 total_time
NameError: name 'total_time' is not defined

```
In [56]: len(times)
```

```
Out[56]: 500
```

def terminator(max_iter, time_steps=len(times), total_time= total_time_evo, epsilon= 2*pi*1): r"""Brief description of the function"""

```
xi_initial = result.u[-1, 0, : ]
#1000*random_sample((time_steps,))
dt = (2*pi)/500  #total_time/time_steps
xi_diff, xi_new_vec = updater(xi_initial, dt, epsilon)

for i in range(max_iter):
    if amax(xi_diff) < epsilon**2 :
        xi_final = xi_new_vec
        break
    else :
        xi_diff, xi_new_vec = updater(xi_new_vec, dt, epsilon)
        print(i)
        print(amax(xi_diff))


xi_final = xi_new_vec
return xi_final
```

```
In [59]: def terminator(max_iter, time_steps=len(times),
                        total_time=total_time_evo,
                        epsilon=2*pi*1):
             r"""Brief description of the function"""
             xi_initial = result.u[-1, 0, :]
             # 1000*random_sample((time_steps,))
             dt = (2*pi)/500 #total_time/time_steps
             xi_diff, xi_new_vec = updater(xi_initial, dt, epsilon)

             for i in range(max_iter):
                 if amax(xi_diff) < epsilon**2 :
                     xi_final = xi_new_vec
                     print("Tejas is unlucky")
                     break
                 else :
                     xi_diff, xi_new_vec = updater(xi_new_vec, dt, epsilon)
                     print("Tejas is a good boy")
```

```
            print(i)
            print(amax(xi_diff))

    return xi_final
```

2.8 sub topic 3

```
In [60]: new_label = [r'$g_{with diss}$']
```

2.8.1 try

```
In [61]: xi_opt = terminator(10)
```

```
Tejas is unlucky
```

```
In [62]: time_steps=len(times)
         total_time= total_time_evo
         epsilon= 2*pi*1
```

```
In [63]: dt = (2*pi)/500
         F(xi_opt, dt)
```

```
Out[63]: -4.223952468753483
```

```
In [64]: L_full_maker(xi_opt, dt)
```

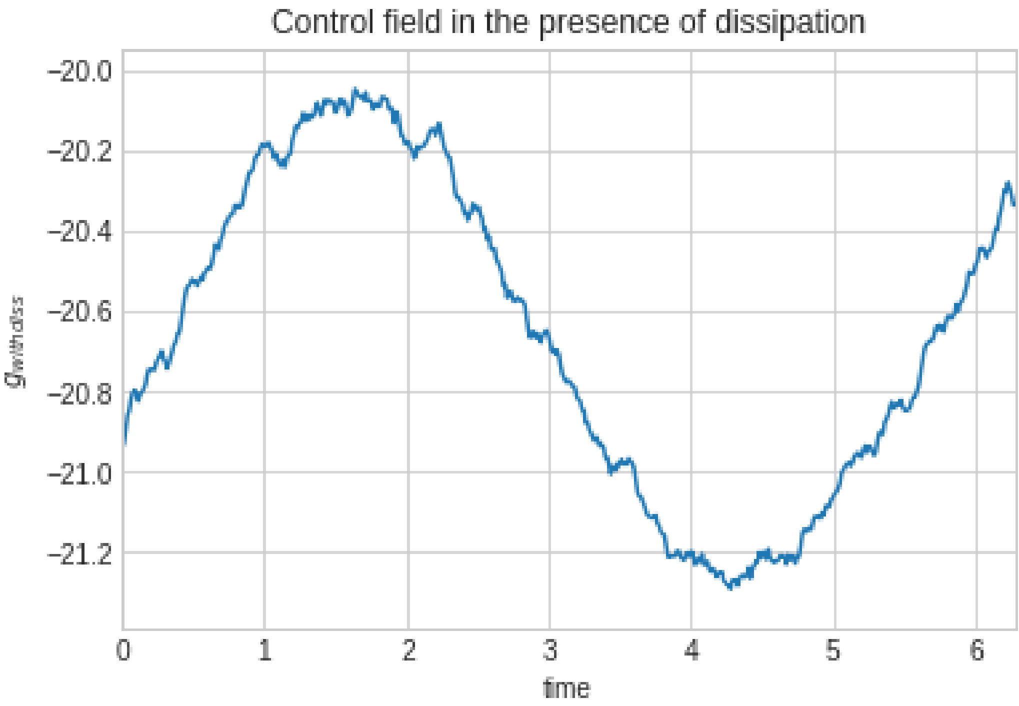
```
Out[64]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.363 & (0.228+0.007j) & (0.228-0.007j) & 0.637 \\ (-0.228-0.007j) & (0.004+0.011j) & (-0.280+1.747\times10^{-04}j) & (0.228+0.007j) \\ (-0.228+0.007j) & (-0.280-1.747\times10^{-04}j) & (0.004-0.011j) & (0.228-0.007j) \\ 0.637 & (-0.228-0.007j) & (-0.228+0.007j) & 0.363 \end{pmatrix}$$

```
In [65]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.2 try

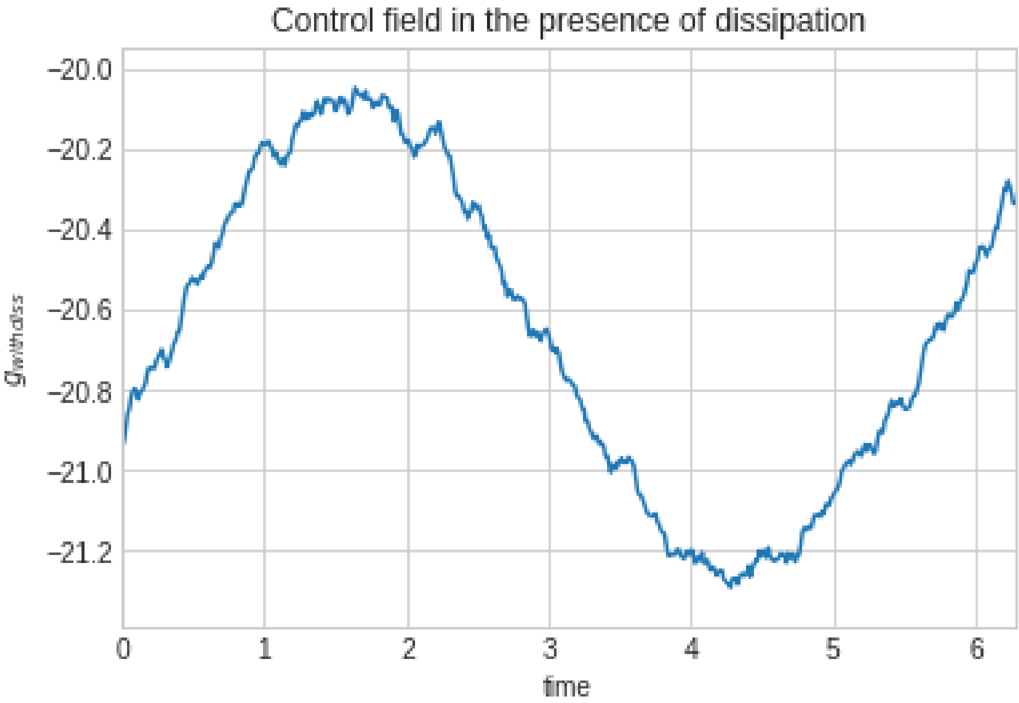
```
In [66]: xi_opt = terminator(1000)

Tejas is unlucky


In [67]: time_steps=len(times)
         total_time= total_time_evo
         epsilon= 2*pi*1

In [68]: dt = (2*pi)/500
         F(xi_opt, dt)

Out[68]: -4.223952468753483

In [69]: L_full_maker(xi_opt, dt)

Out[69]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.363 & (0.228+0.007j) & (0.228-0.007j) & 0.637 \\ (-0.228-0.007j) & (0.004+0.011j) & (-0.280+1.747\times10^{-04}j) & (0.228+0.007j) \\ (-0.228+0.007j) & (-0.280-1.747\times10^{-04}j) & (0.004-0.011j) & (0.228-0.007j) \\ 0.637 & (-0.228-0.007j) & (-0.228+0.007j) & 0.363 \end{pmatrix}$$

```
In [70]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.3 try

```
In [71]: xi_opt = terminator(1000,time_steps=len(times), total_time= total_time_evo,
                             epsilon= ((0.1*2*pi)/(times[-1])))
```

```
Tejas is unlucky
```

```
In [72]: time_steps=len(times)
         total_time= total_time_evo
         epsilon = ((0.1*2*pi)/(times[-1]))
```

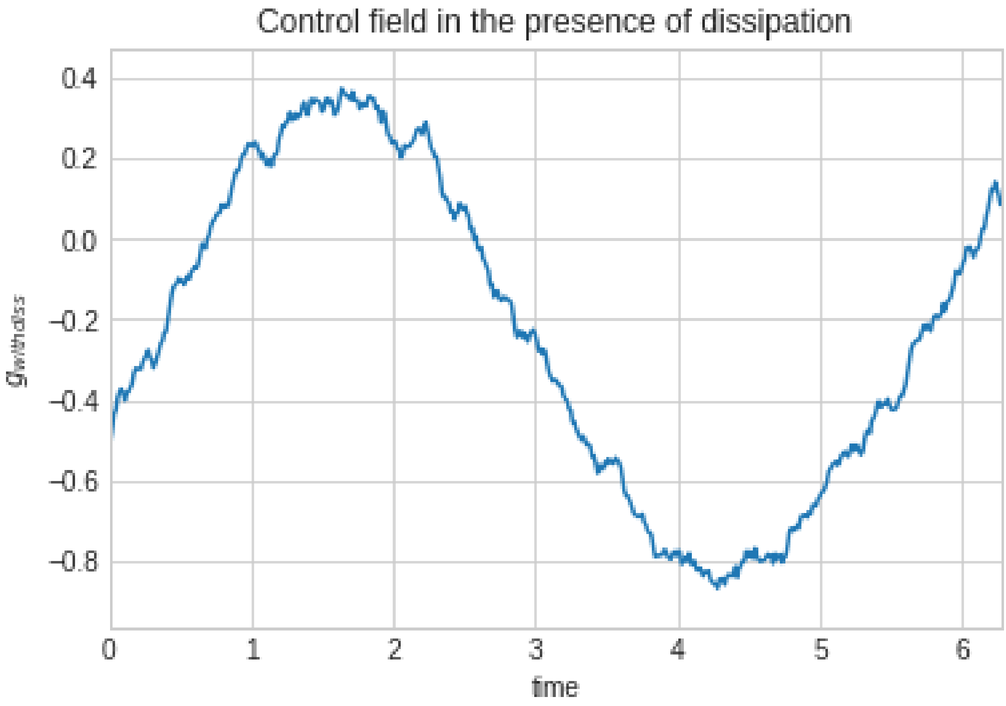
```
In [73]: dt = (2*pi)/500
         F(xi_opt, dt)

Out[73]: -2.2265924567489033

In [74]: L_full_maker(xi_opt, dt)
```

```
Out[74]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.320 & (0.088-0.088j) & (0.088+0.088j) & 0.680 \\ (0.243+0.060j) & (-0.051+0.006j) & (0.175+0.265j) & (-0.243-0.060j) \\ (0.243-0.060j) & (0.175-0.265j) & (-0.051-0.006j) & (-0.243+0.060j) \\ 0.680 & (-0.088+0.088j) & (-0.088-0.088j) & 0.320 \end{pmatrix}$$

```
In [75]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');

/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Con
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.4 try

```
In [76]: xi_opt = terminator(10,time_steps=len(times), total_time= total_time_evo,
                             epsilon= ((0.1*2*pi)/(times[-1])))
```

```
Tejas is unlucky
```

```
In [77]: time_steps=len(times)
         total_time= total_time_evo
         epsilon = ((0.1*2*pi)/(times[-1]))
```

```
In [78]: dt = (2*pi)/500
         F(xi_opt, dt)
```

```
Out[78]: -2.2265924567489033
```

```
In [79]: L_full_maker(xi_opt, dt)
```

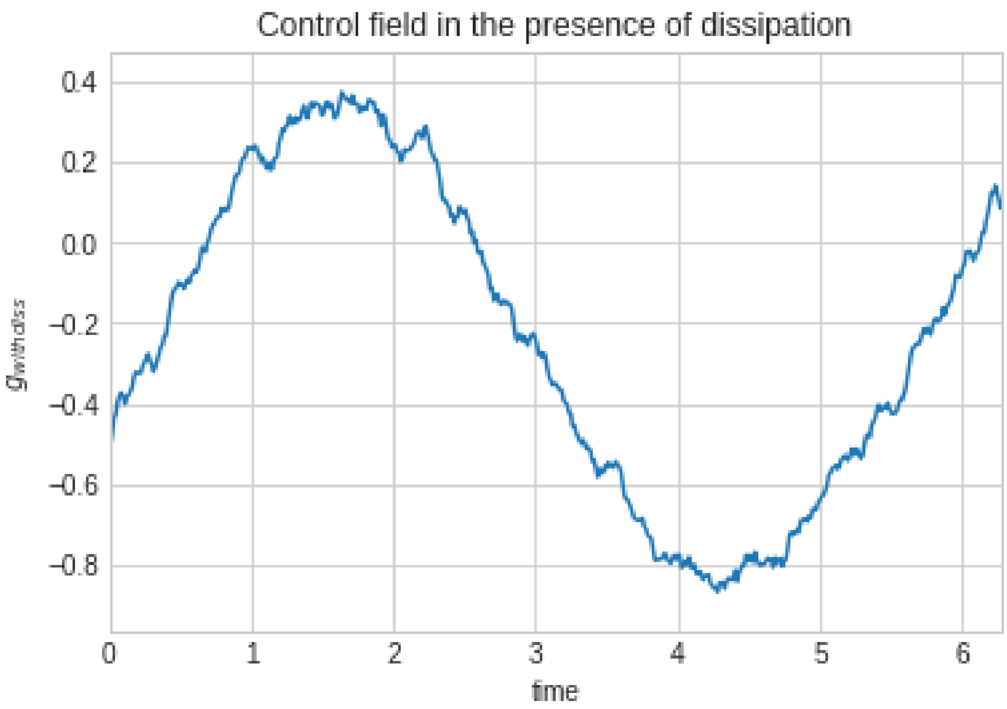
```
Out[79]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.320 & (0.088-0.088j) & (0.088+0.088j) & 0.680 \\ (0.243+0.060j) & (-0.051+0.006j) & (0.175+0.265j) & (-0.243-0.060j) \\ (0.243-0.060j) & (0.175-0.265j) & (-0.051-0.006j) & (-0.243+0.060j) \\ 0.680 & (-0.088+0.088j) & (-0.088-0.088j) & 0.320 \end{pmatrix}$$

```
In [80]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.5 try

```
In [81]: xi_opt = terminator(1000,time_steps=len(times), total_time= total_time_evo,
                            epsilon= ((0.1*2*pi)/(10**3)))
```

```
Tejas is unlucky
```

```
In [82]: time_steps=len(times)
         total_time= total_time_evo
         epsilon = ((0.1*2*pi)/(times[-1]))
```

```
In [83]: dt = (2*pi)/500
         F(xi_opt, dt)
```

```
Out[83]: -0.9549811435604786
```

```
In [84]: F(result.u[-1, 0, : ], dt)
```

```
Out[84]: -0.954828683341337
```

```
In [85]: L_full_maker(xi_opt, dt)
```

```
Out[85]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.219 & (0.033-0.027j) & (0.033+0.027j) & 0.781 \\ (-0.038+0.006j) & (-0.065-0.049j) & (0.394+0.015j) & (0.038-0.006j) \\ (-0.038-0.006j) & (0.394-0.015j) & (-0.065+0.049j) & (0.038+0.006j) \\ 0.781 & (-0.033+0.027j) & (-0.033-0.027j) & 0.219 \end{pmatrix}$$

```
In [86]: len(times)
```

```
Out[86]: 500
```

```
In [87]: new_label = [r'$g_{with diss}$']
```

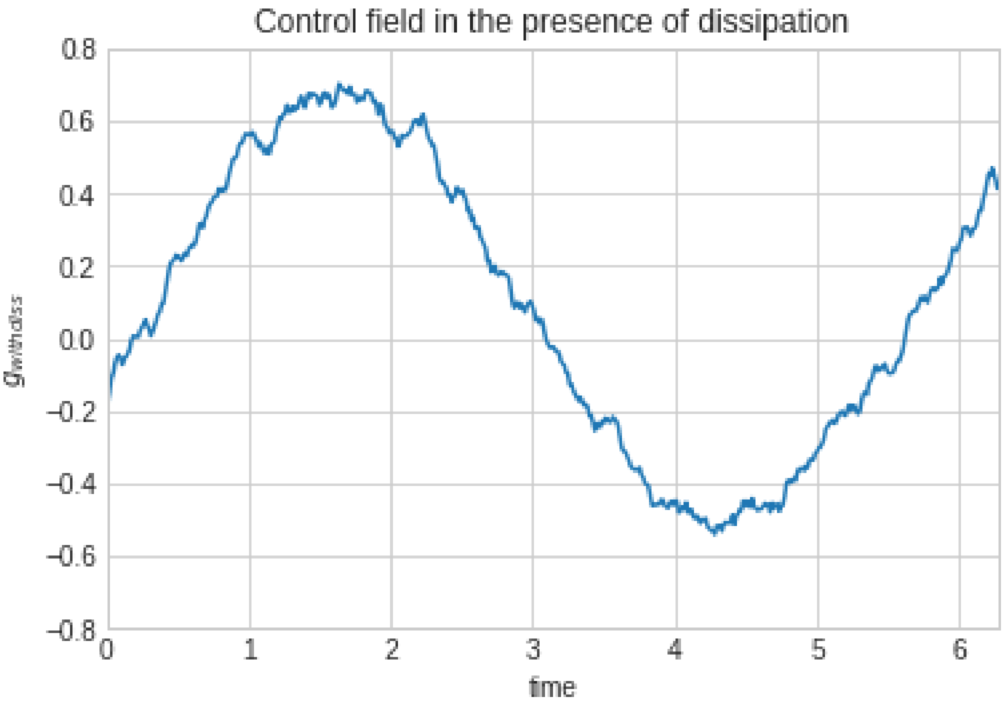
”’plot_grape_control_fields(times, xi_opt / (2 * np.pi), new_label, uniform_axes=True); ValueError Traceback (most recent call last) in () 1 plot_grape_control_fields(times, ----> 2 xi_opt / (2 * np.pi), new_label, uniform_axes=True);

/anaconda3/envs/qutip-env/lib/python3.6/site-packages/qutip/control/grape.py in plot_grape_control_fields(times, u, labels, uniform_axes) 101 import matplotlib.pyplot as plt 102 --> 103 R, J, M = u.shape 104 105 fig, axes = plt.subplots(J, 1, figsize=(8, 2 * J), squeeze=False)

ValueError: not enough values to unpack (expected 3, got 1) ”’

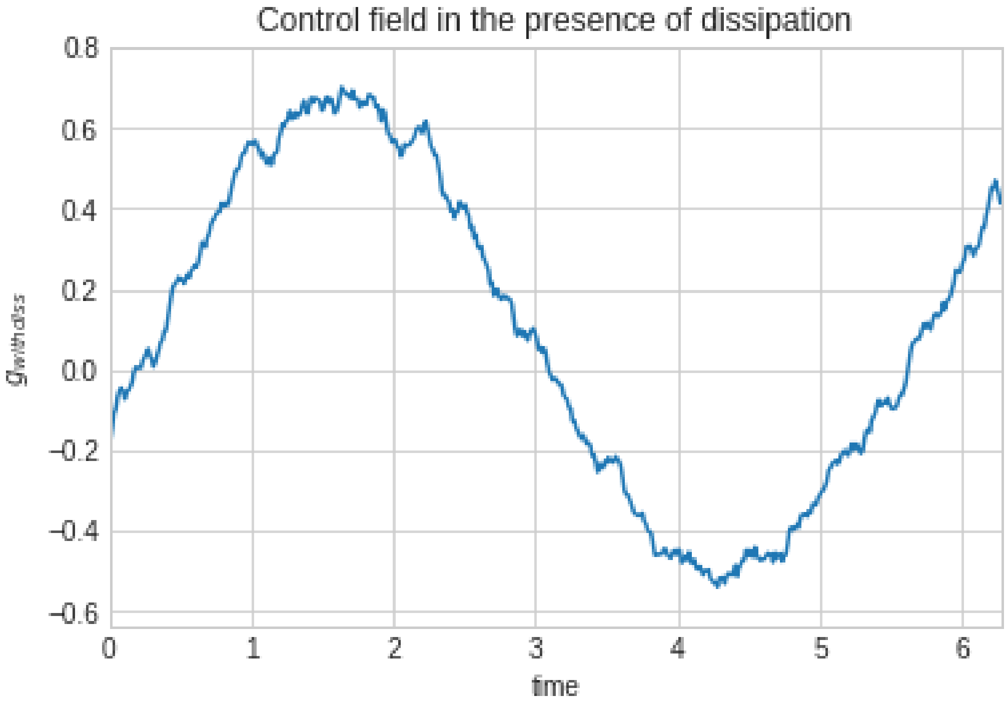
```
In [88]: ax = axes()
         ax.plot(times, xi_opt)
         ax.set(xlim=(0, total_time_evo), ylim=(-0.8,0.8 ),
               xlabel='time', ylabel= r'$g_{with diss}$',
               title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
```

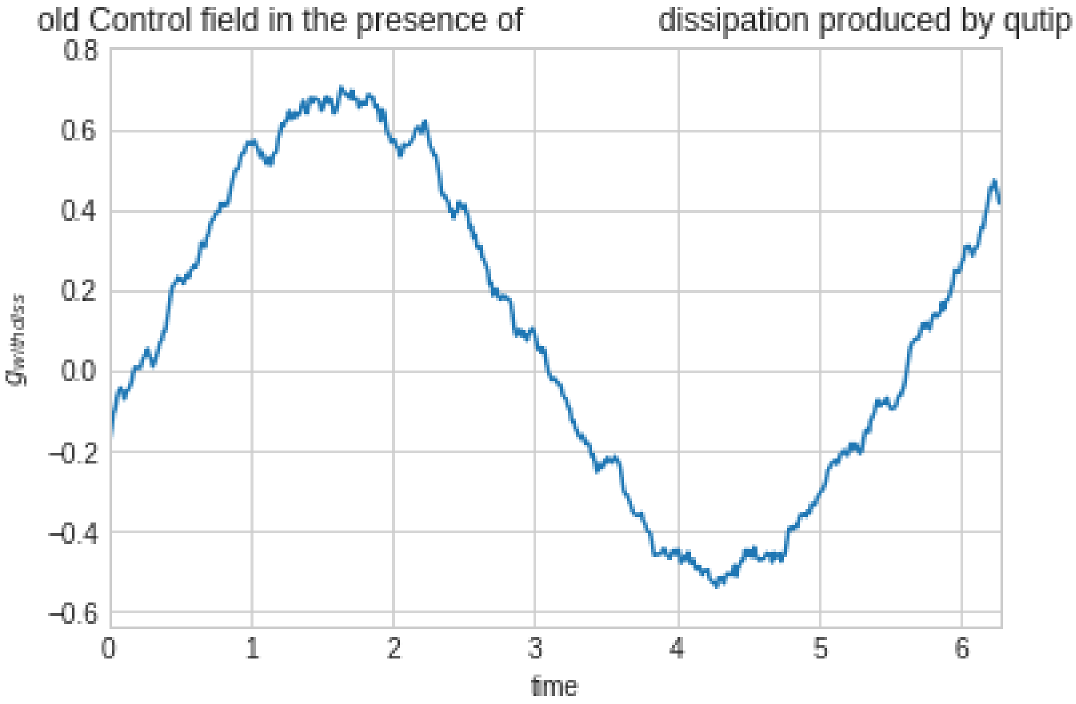


```
In [89]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



```
In [90]: ax = axes()
         ax.plot(times, result.u[-1, 0, : ])
         xi_max = amax(result.u[-1, 0, : ]) + 0.1
         xi_min = amin(result.u[-1, 0, : ]) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='old Control field in the presence of \
                       dissipation produced by qutip');
```



2.8.6 try

```
In [91]: xi_opt = terminator(1000,time_steps=len(times), total_time= total_time_evo,
                             epsilon= ((0.1*2*pi)/(10**4)))
```

```
Tejas is unlucky
```

```
In [92]: dt = (2*pi)/500
         F(xi_opt, dt)
```

```
Out[92]: -0.9548389887847603
```

```
In [93]: L_full_maker(xi_opt, dt)
```

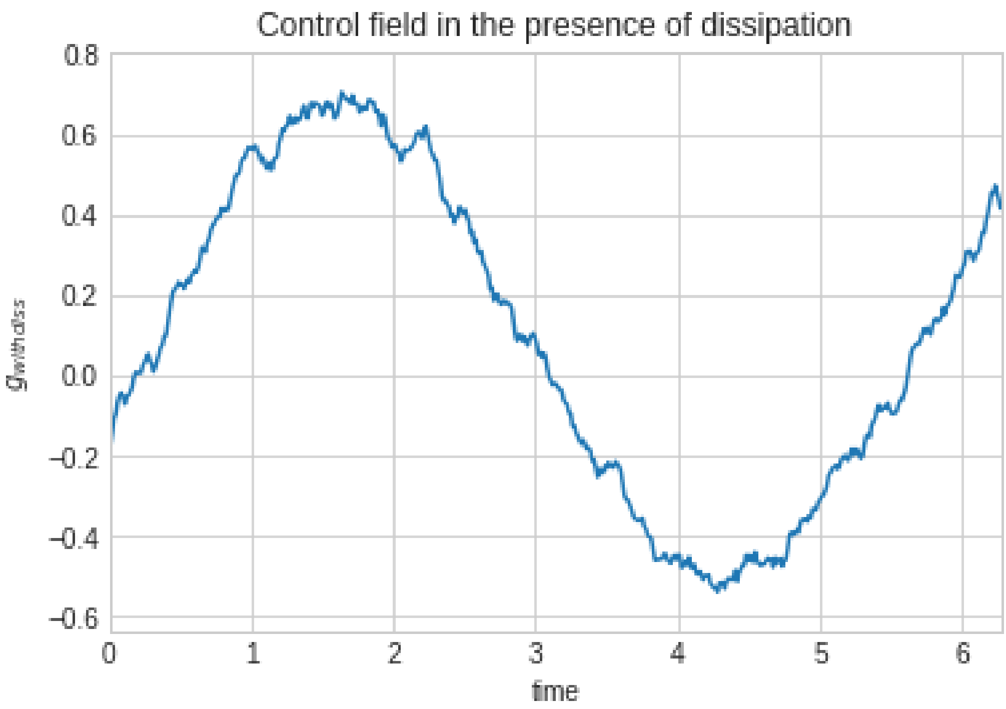
Out[93]:

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.219 & (0.035-0.028j) & (0.035+0.028j) & 0.781 \\ (-0.038+0.006j) & (-0.064-0.049j) & (0.394+0.012j) & (0.038-0.006j) \\ (-0.038-0.006j) & (0.394-0.012j) & (-0.064+0.049j) & (0.038+0.006j) \\ 0.781 & (-0.035+0.028j) & (-0.035-0.028j) & 0.219 \end{pmatrix}$$

```
In [94]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.7 try

```
In [95]: xi_opt = terminator(10**4,time_steps=len(times), total_time= total_time_evo,
                             epsilon= ((0.1*2*pi)/(10**3)))
```

```
Tejas is unlucky
```

```
In [96]: dt = (2*pi)/500
         F(xi_opt, dt)
```

```
Out[96]: -0.9549811435604786
```

```
In [97]: L_full_maker(xi_opt, dt)
```

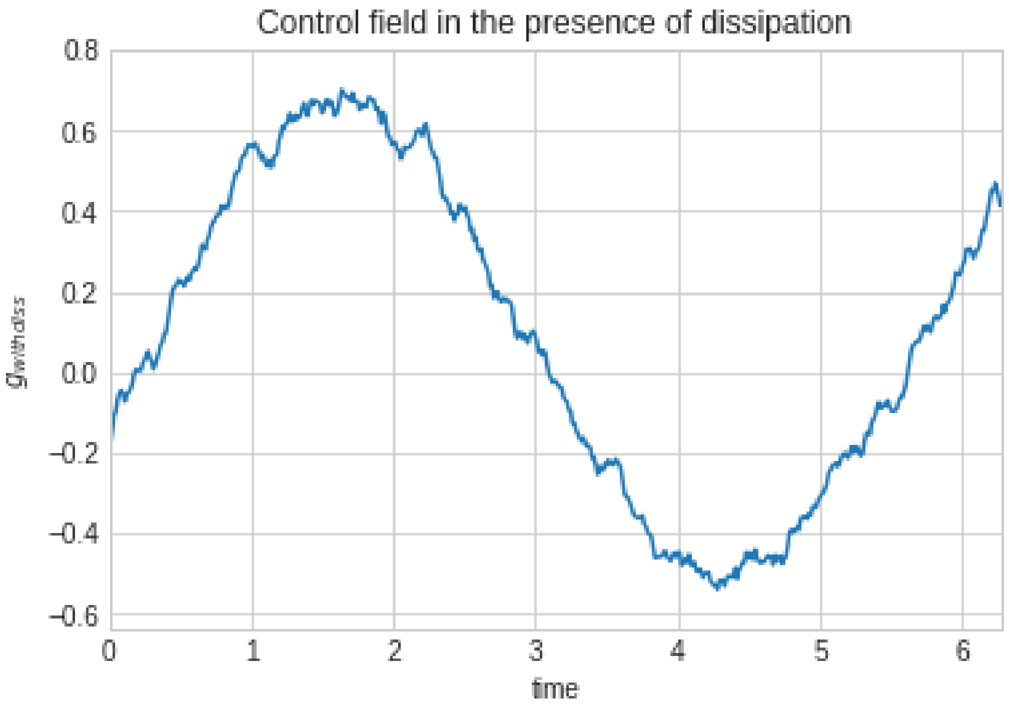
```
Out[97]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.219 & (0.033-0.027j) & (0.033+0.027j) & 0.781 \\ (-0.038+0.006j) & (-0.065-0.049j) & (0.394+0.015j) & (0.038-0.006j) \\ (-0.038-0.006j) & (0.394-0.015j) & (-0.065+0.049j) & (0.038+0.006j) \\ 0.781 & (-0.033+0.027j) & (-0.033-0.027j) & 0.219 \end{pmatrix}$$

```
In [98]: ax = axes()
         ax.plot(times, xi_opt)
         xi_max = amax(xi_opt) + 0.1
         xi_min = amin(xi_opt) - 0.1

         ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                xlabel='time', ylabel= r'$g_{with diss}$',
                title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.8 try

```
In [99]: xi_opt = terminator(10**4,time_steps=len(times), total_time= total_time_evo,
                             epsilon= ((0.1*2*pi)/(10**4)))
```

```
Tejas is unlucky
```

```
In [100]: dt = (2*pi)/500
          F(xi_opt, dt)
```

```
Out[100]: -0.9548389887847603
```

```
In [101]: L_full_maker(xi_opt, dt)
```

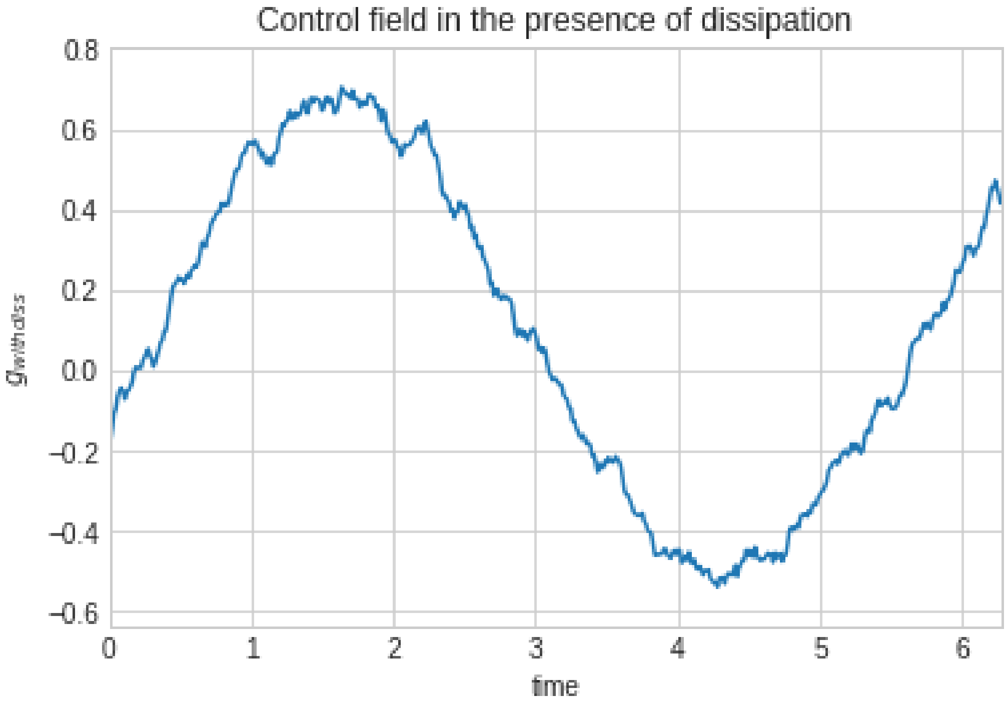
Out[101]:

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.219 & (0.035-0.028j) & (0.035+0.028j) & 0.781 \\ (-0.038+0.006j) & (-0.064-0.049j) & (0.394+0.012j) & (0.038-0.006j) \\ (-0.038-0.006j) & (0.394-0.012j) & (-0.064+0.049j) & (0.038+0.006j) \\ 0.781 & (-0.035+0.028j) & (-0.035-0.028j) & 0.219 \end{pmatrix}$$

```
In [102]: ax = axes()
          ax.plot(times, xi_opt)
          xi_max = amax(xi_opt) + 0.1
          xi_min = amin(xi_opt) - 0.1

          ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                 xlabel='time', ylabel= r'$g_{with diss}$',
                 title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.9 try

```
In [103]: xi_opt = terminator(10**4,time_steps=len(times), total_time= total_time_evo,
                               epsilon= ((0.1*2*pi)/(10**10)))
```

```
Tejas is unlucky
```

```
In [104]: dt = (2*pi)/500
          F(xi_opt, dt)

Out[104]: -0.9548286833510965

In [105]: L_full_maker(xi_opt, dt)
```

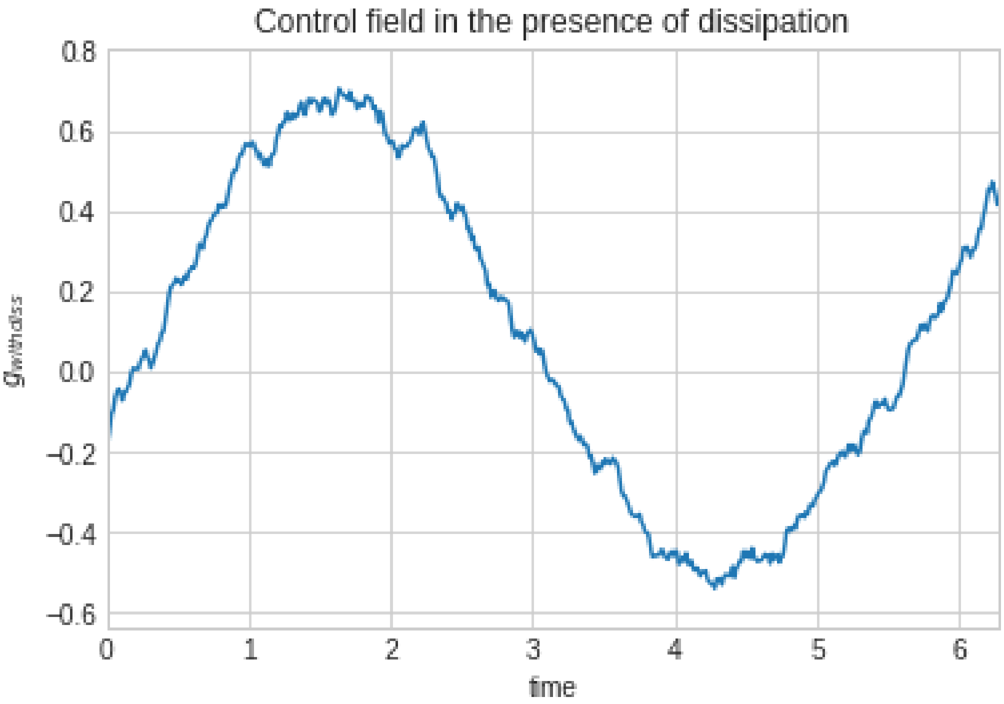
```
Out[105]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\begin{pmatrix} 0.219 & (0.035-0.028j) & (0.035+0.028j) & 0.781 \\ (-0.038+0.006j) & (-0.064-0.049j) & (0.394+0.012j) & (0.038-0.006j) \\ (-0.038-0.006j) & (0.394-0.012j) & (-0.064+0.049j) & (0.038+0.006j) \\ 0.781 & (-0.035+0.028j) & (-0.035-0.028j) & 0.219 \end{pmatrix}$$

```
In [106]: ax = axes()
          ax.plot(times, xi_opt)
          xi_max = amax(xi_opt) + 0.1
          xi_min = amin(xi_opt) - 0.1

          ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                 xlabel='time', ylabel= r'$g_{with diss}$',
                 title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.8.10 try

```
In [107]: xi_opt = terminator(10**4,time_steps=10**3, total_time= total_time_evo,
                              epsilon= ((0.1*2*pi)/(10**3)))
```

```
Tejas is unlucky
```

```
In [108]: dt = (2*pi)/(10**3)#(2*pi)/500
          # probaly happened because of hard coding of dt inside terminator
          # must try it again after changing that
          F(xi_opt, dt)
```

```
Out[108]: -4.870058606858147
```

```
In [109]: L_full_maker(xi_opt, dt)
```

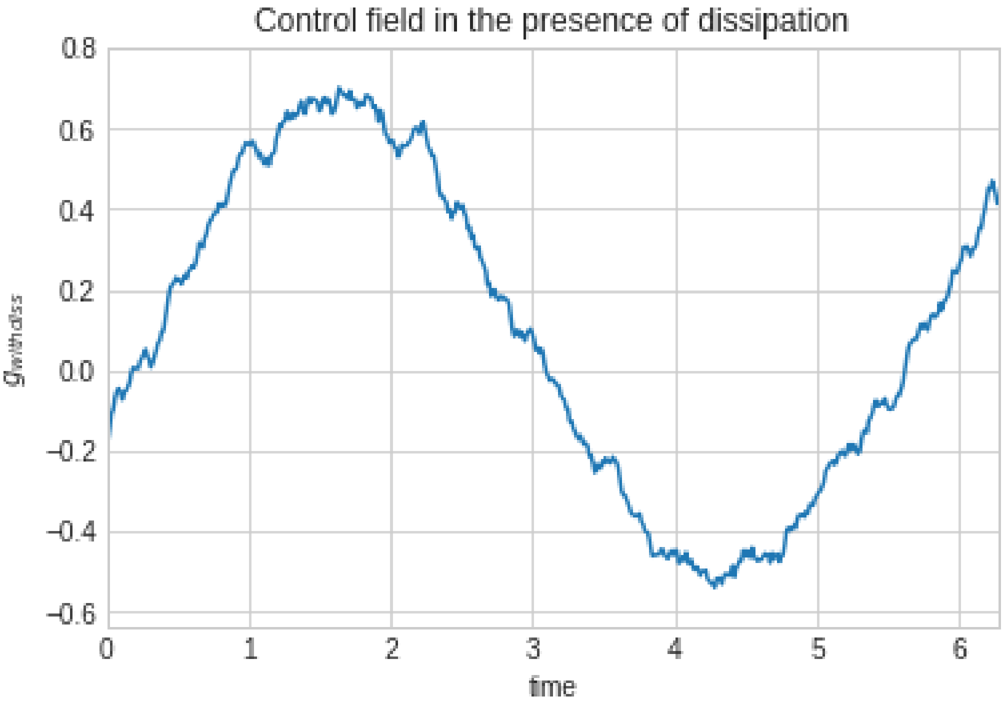
```
Out[109]:
```

Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = False

$$\left(\begin{array}{cccc}0.755 & (-0.333-0.092j) & (-0.333+0.092j) & 0.245\\(-0.256+0.074j) & (-0.402+0.137j) & (0.146-0.056j) & (0.256-0.074j)\\(-0.256-0.074j) & (0.146+0.056j) & (-0.402-0.137j) & (0.256+0.074j)\\0.245 & (0.333+0.092j) & (0.333-0.092j) & 0.755\\\end{array}\right)$$

```
In [110]: ax = axes()
          ax.plot(times, xi_opt)
          xi_max = amax(xi_opt) + 0.1
          xi_min = amin(xi_opt) - 0.1

          ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                 xlabel='time', ylabel= r'$g_{with diss}$',
                 title='Control field in the presence of dissipation');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.9 APS stuff

2.9.1 infidelity function tryouts

```
In [111]: qone = basis(2, 0)
          qone
```

```
Out[111]:
```

Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\begin{pmatrix} 1.0 \\ 0.0 \end{pmatrix}$$

```
In [112]: qzero = basis(2, 1)
          qzero
```

```
Out[112]:
```

Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\begin{pmatrix} 0.0 \\ 1.0 \end{pmatrix}$$

2.10 ###### ket2bra

NameError Traceback (most recent call last) in () ----> 1 ket2bra
NameError: name 'ket2bra' is not defined

```
In [113]: (sigmax()).matrix_element(qone, qzero)
```

```
Out[113]: (1+0j)
```

```
In [114]: abs((sigmax()).matrix_element(qone, qzero))
```

```
Out[114]: 1.0
```

```
In [115]: abs(2 + 3j)
```

```
Out[115]: 3.605551275463989
```

```
In [116]: abs(3 + 4j)
```

```
Out[116]: 5.0
```

```
In [117]: def infidelity_to_one(U):
              r"""infidelity to state one"""
              fidelity = (abs(U.matrix_element(qone, qzero)))**2
              infidelity = 1 - fidelity
              return infidelity
```

```
In [118]: infidelity_to_one(sigmax())
          # 1 - abs((sigmax()).matrix_element(qone, qzero))
          # 1 - 1 = 0
```

```
Out[118]: 0.0
```

result.u, result.u but lindbladian evolution, xi_opt

```
In [119]: qone.overlap(qzero)
```

```
Out[119]: 0j
```

```
In [120]: qzero.overlap(qzero)
```

```
Out[120]: (1+0j)
```

```
In [121]: qone.overlap(qone)
```

```
Out[121]: (1+0j)
```

```
In [122]: def infidelity_to_state(psi):
              r"""infidelity to state one"""
              fidelity = (abs(qone.overlap(psi)))**2
              infidelity = 1 - fidelity
              return infidelity
```

```
In [123]: infidelity_to_state(qzero)
```

```
Out[123]: 1.0
```

```
In [124]: infidelity_to_state(qone)
```

```
Out[124]: 0.0
```

```
In [125]: #mesolve
```

2.10.1 state building

```
In [126]: H_no_diss = [H_0, [H_1, result.u[-1, 0, :] ]]
          grape_no_diss = mesolve(H_no_diss, qzero, times, c_ops=[], e_ops=[],
                                  args={}, options=None,
                                  progress_bar=EnhancedTextProgressBar() )
```

```
 Total run time:   0.60s*] Elapsed 0.60s / Remaining 00:00:00:00[*********57%*           ] Elaps
```

```
In [127]: len(grape_no_diss.states)
```

```
Out[127]: 500
```

```
In [128]: Lin
```

```
Out[128]:
```

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & -1.0 \end{pmatrix}$$

```
In [129]: sqrt(gamma)
```

```
Out[129]: 0.31622776601683794
```

```
In [130]: c_ops_tejas = sqrt(gamma)*Lin
          c_ops_tejas
```

Out[130]:

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 0.316 & 0.0 \\ 0.0 & -0.316 \end{pmatrix}$$

```
In [131]: H_diss = [H_0, [H_1, result.u[-1, 0, :] ]]
          grape_diss = mesolve(H_no_diss, qzero, times, c_ops=[c_ops_tejas], e_ops=[],
                               args={}, options=None,
                               progress_bar=EnhancedTextProgressBar() )
```

```
 Total run time:   0.39s*] Elapsed 0.39s / Remaining 00:00:00:00
```

```
In [132]: len(grape_diss.states)
```

```
Out[132]: 500
```

```
In [133]: H_diss_optimized = [H_0, [H_1, xi_opt ]]
          grape_diss_optimized = mesolve(H_no_diss, qzero, times, c_ops=[c_ops_tejas],
                                         e_ops=[], args={}, options=None,
                                         progress_bar=TextProgressBar() )
```

```
10.0%. Run time:   0.01s. Est. time left: 00:00:00:00
20.0%. Run time:   0.02s. Est. time left: 00:00:00:00
30.0%. Run time:   0.03s. Est. time left: 00:00:00:00
40.0%. Run time:   0.04s. Est. time left: 00:00:00:00
50.0%. Run time:   0.05s. Est. time left: 00:00:00:00
60.0%. Run time:   0.06s. Est. time left: 00:00:00:00
70.0%. Run time:   0.06s. Est. time left: 00:00:00:00
80.0%. Run time:   0.07s. Est. time left: 00:00:00:00
90.0%. Run time:   0.08s. Est. time left: 00:00:00:00
Total run time:   0.10s
```

```
In [134]: len(grape_diss_optimized.states)
```

```
Out[134]: 500
```

2.10.2 states list to fidelity list

```
In [135]: infidelity_to_state_vec = vectorize(infidelity_to_state)
```

```
In [136]: infid_grape_no_diss = infidelity_to_state_vec(grape_no_diss.states)
```

```
In [137]: '''fid_grape_diss = infidelity_to_state_vec(grape_diss.states)
          TypeError                                 Traceback (most recent call last)
          <ipython-input-101-fbabc06eff84> in <module>()
          ----> 1 fid_grape_diss = infidelity_to_state_vec(grape_diss.states)

          /anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/lib/function_base.py in _
             2753                 vargs.extend([kwargs[_n] for _n in names])
             2754
          -> 2755             return self._vectorize_call(func=func, args=vargs)
             2756
             2757         def _get_ufunc_and_otypes(self, func, args):

          /anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/lib/function_base.py in _
             2823                 res = func()
             2824             else:
          -> 2825                 ufunc, otypes = self._get_ufunc_and_otypes(func=func, args=args)
             2826
             2827                 # Convert args to object arrays first

          /anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/lib/function_base.py in _
             2783
             2784                 inputs = [arg.flat[0] for arg in args]
          -> 2785                 outputs = func(*inputs)
             2786
             2787                 # Performance note: profiling indicates that -- for simple

          <ipython-input-96-32321f8d8a33> in infidelity_to_state(psi)
                1 def infidelity_to_state(psi):
                2     r"""infidelity to state one"""
          ----> 3     fidelity = (abs(qone.overlap(psi)))**2
                4     infidelity = 1 - fidelity
                5     return infidelity

          /anaconda3/envs/qutip-env/lib/python3.6/site-packages/qutip/qobj.py in overlap(self,
             1486                         return (self.data.H * state.data)[0, 0]
             1487
          -> 1488             raise TypeError("Can only calculate overlap for state vector Qobjs")
             1489
             1490         def eigenstates(self, sparse=False, sort='low',

          TypeError: Can only calculate overlap for state vector Qobjs
```

```
            '''
```

```
Out[137]: 'fid_grape_diss = infidelity_to_state_vec(grape_diss.states)\nTypeError
```

fid_grape_no_diss = infidelity_to_state_vec(grape_no_diss.states)

```
In [138]: fidelity(sigmax(), sigmax())
```

```
Out[138]: 1.9999999999999996
```

```
In [139]: zero_dm = ket2dm(qzero)
          zero_dm
```

```
Out[139]:
```

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\left(\begin{array}{cc}0.0 & 0.0\\0.0 & 1.0\end{array}\right)$$

```
In [140]: one_dm = ket2dm(qone)
          one_dm
```

```
Out[140]:
```

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\left(\begin{array}{cc}1.0 & 0.0\\0.0 & 0.0\end{array}\right)$$

```
In [141]: fidelity(zero_dm, zero_dm)
```

```
Out[141]: 1.0
```

```
In [142]: fidelity(zero_dm, one_dm)
```

```
Out[142]: 0.0
```

```
In [143]: fidelity(one_dm, zero_dm)
```

```
Out[143]: 0.0
```

```
In [144]: grape_no_diss.states[5]
```

```
Out[144]:
```

Quantum object: dims = [[2], [1]], shape = (2, 1), type = ket

$$\left(\begin{array}{c}(0.008-4.378\times10^{-05}j)\\(0.999+0.031j)\end{array}\right)$$

```
In [145]: grape_diss.states[5]
```

```
Out[145]:
```

Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 6.496 \times 10^{-05} & (0.008 - 2.952 \times 10^{-04}j) \\ (0.008 + 2.952 \times 10^{-04}j) & 1.000 \end{pmatrix}$$

```
In [146]: grape_diss.states[7]
```

```
Out[146]:
```

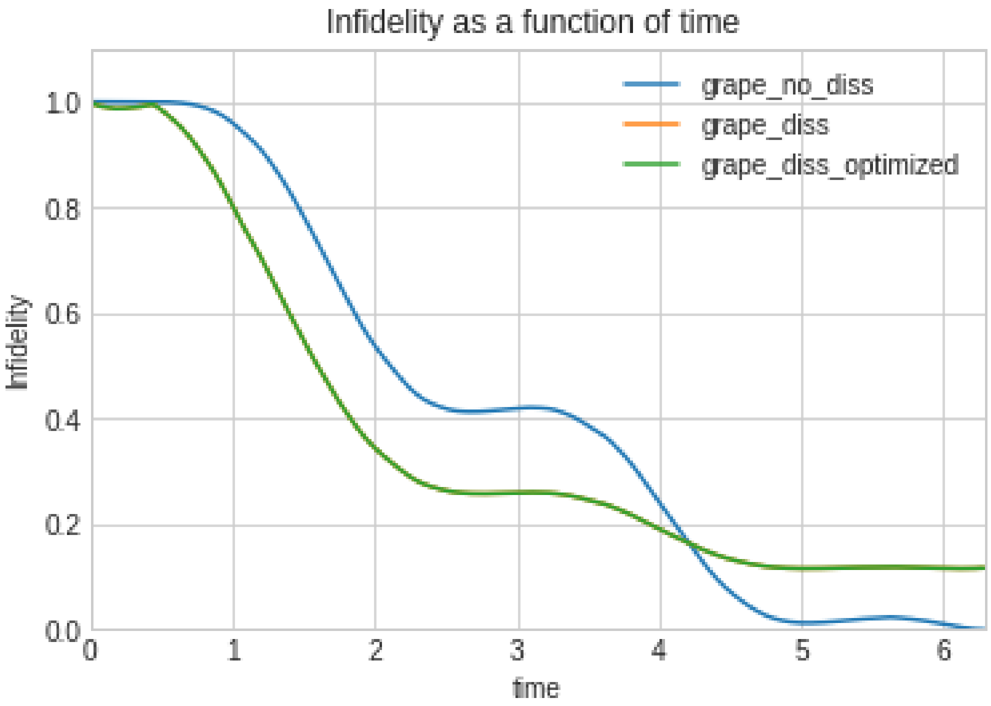
Quantum object: dims = [[2], [2]], shape = (2, 2), type = oper, isherm = True

$$\begin{pmatrix} 8.740 \times 10^{-05} & (0.009 - 5.120 \times 10^{-04}j) \\ (0.009 + 5.120 \times 10^{-04}j) & 1.000 \end{pmatrix}$$

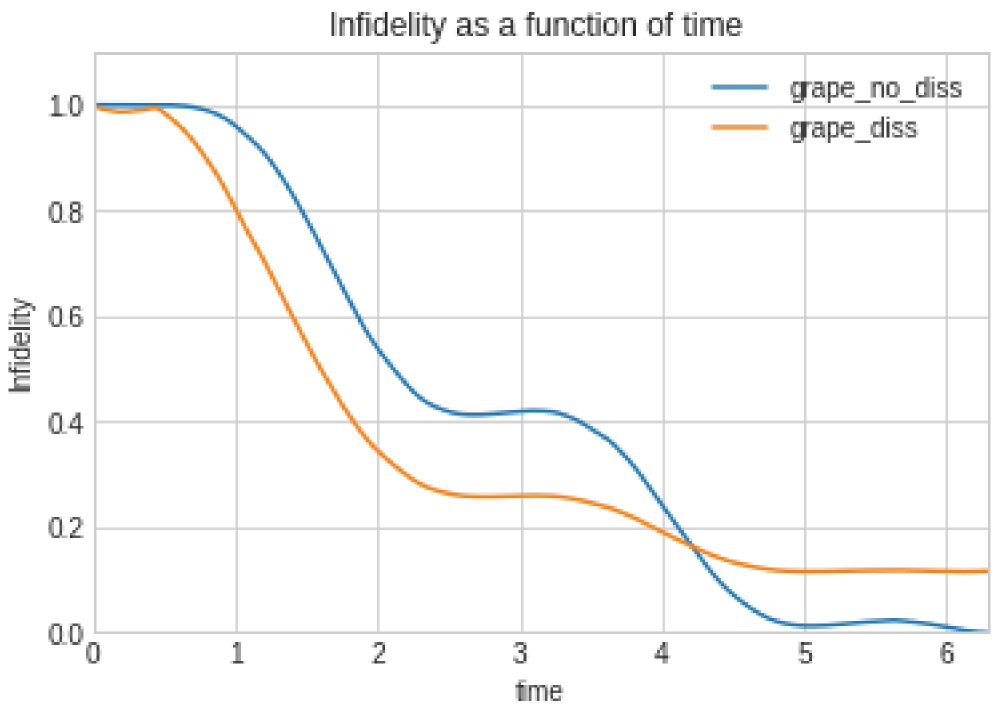
```
In [147]: fidelity(grape_diss.states[7], one_dm)

Out[147]: 0.00934875656096058

In [148]: #0.0122.347E10**05j)

In [149]: def infidelity_tejas(dm, d=2):
              r"""Brief description of the function"""
              infid = 1 - fidelity(one_dm, dm)
              return infid

In [150]: infid_dm_vec = vectorize(infidelity_tejas)

In [151]: infid_grape_diss = infid_dm_vec(grape_diss.states)

In [152]: len(infid_grape_diss)

Out[152]: 500

In [153]: infid_grape_diss_optimized = infid_dm_vec(grape_diss_optimized.states)

In [154]: len(infid_grape_diss_optimized)

Out[154]: 500
```

2.10.3 plots

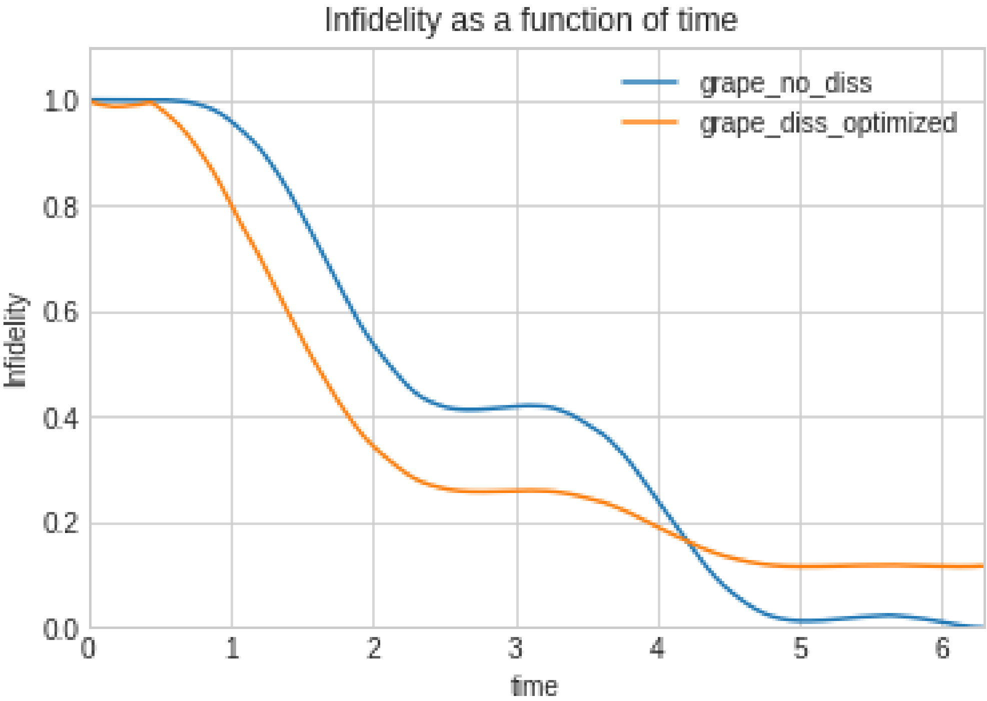
```
In [155]: ax = axes()
          ax.plot(times, infid_grape_no_diss, label='grape_no_diss')
          ax.plot(times, infid_grape_diss, label='grape_diss')
          ax.plot(times, infid_grape_diss_optimized, label='grape_diss_optimized')
          #ax.plot(x, cos(x), ':b', label='cos(x)')'-g',
          #ax.axis('equal')
          ax.legend()
          ax.set(xlim=(times[0],times[-1]), ylim=(0, 1.1),
                 xlabel='time', ylabel='Infidelity',
                 title='Infidelity as a function of time ');
```



```
In [156]: ax = axes()
          ax.plot(times, infid_grape_no_diss, label='grape_no_diss')
          ax.plot(times, infid_grape_diss, label='grape_diss')
          #ax.plot(times, infid_grape_diss_optimized, label='grape_diss_optimized')
          #ax.plot(x, cos(x), ':b', label='cos(x)')'-g',
          #ax.axis('equal')
          ax.legend()

          ax.set(xlim=(times[0],times[-1]), ylim=(0, 1.1),
                 xlabel='time', ylabel='Infidelity',
                 title='Infidelity as a function of time ');
```

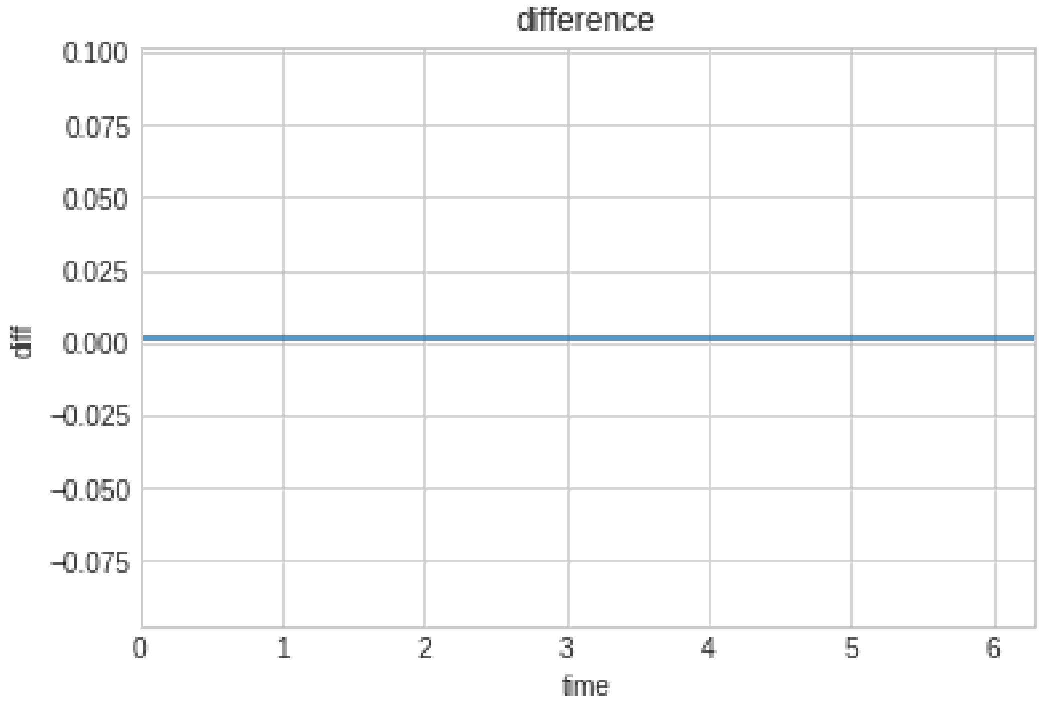


```
In [157]: ax = axes()
          ax.plot(times, infid_grape_no_diss, label='grape_no_diss')
          #ax.plot(times, infid_grape_diss, label='grape_diss')
          ax.plot(times, infid_grape_diss_optimized, label='grape_diss_optimized')
          #ax.plot(x, cos(x), ':b', label='cos(x)')'-g',
          #ax.axis('equal')
          ax.legend()

          ax.set(xlim=(times[0],times[-1]), ylim=(0, 1.1),
                 xlabel='time', ylabel='Infidelity',
                 title='Infidelity as a function of time ');
```

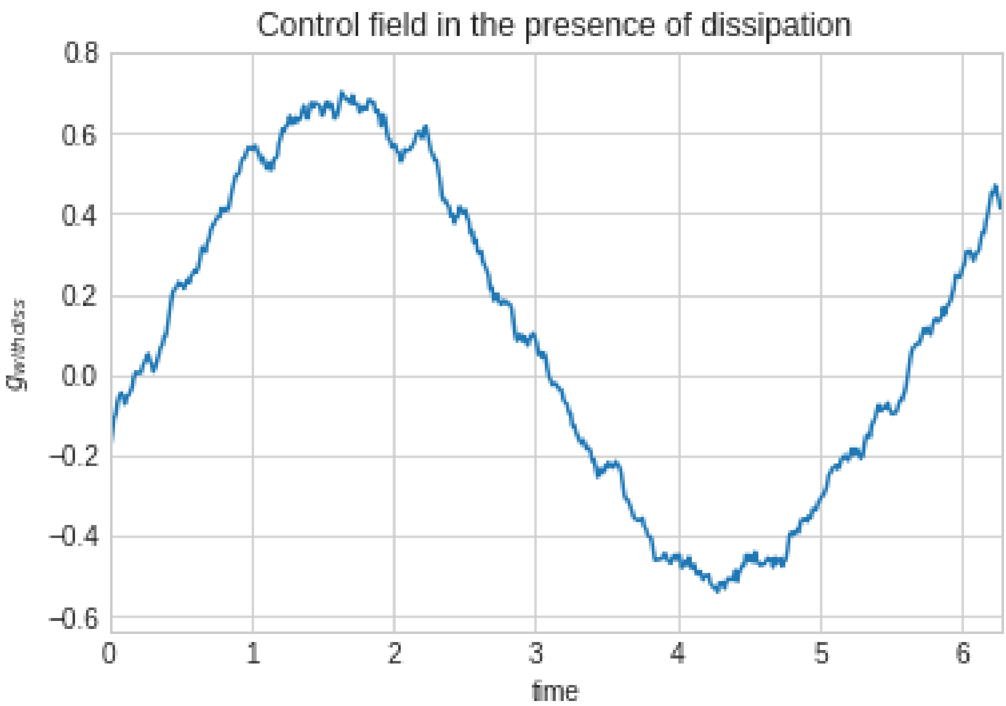


```
In [158]: ax = axes()
          c_diff = result.u[-1, 0, : ] - xi_opt
          ax.plot(times, c_diff)
          c_max = amax(c_diff) + 0.1
          c_min = amin(c_diff) - 0.1

          ax.set(xlim=(0, total_time_evo), ylim=(c_min, c_max),
                 xlabel='time', ylabel= 'diff',
                 title='difference');
```

```
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Com
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



```
In [159]: ax = axes()
          ax.plot(times, xi_opt)
          xi_max = amax(xi_opt) + 0.1
          xi_min = amin(xi_opt) - 0.1

          ax.set(xlim=(0, total_time_evo), ylim=(xi_min, xi_max),
                 xlabel='time', ylabel= r'$g_{with diss}$',
                 title='Control field in the presence of dissipation');

/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/numpy/core/numeric.py:501: Co
  return array(a, dtype, copy=False, order=order)
/home/tejas/anaconda3/envs/qutip-env/lib/python3.6/site-packages/matplotlib/transforms.py:968:
  self._points[:, 1] = interval
```



2.11 Versions

```
In [160]: from qutip.ipynbtools import version_table

          version_table()
```

```
Out[160]: <IPython.core.display.HTML object>
```

```
In [161]: cnot()
```

Out[161]:
Quantum object: dims = [[2, 2], [2, 2]], shape = (4, 4), type = oper, isherm = True

$$\begin{pmatrix} 1.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 1.0 & 0.0 \end{pmatrix}$$