3.1 One Qubit

Unknown Author

March 13, 2014

In [1]:
```
from qutip import *
from scipy import *
```

In [2]:
```
h_p = (1./2.) + (1./2.) * sigmaz()
h_p
```

Out [2]:
Quantum object: dims = [[2], [2]], shape = [2, 2], type = oper, isHerm = True $\begin{pmatrix} 1.0 & 0.0 \\ 0.0 & 0.0 \end{pmatrix}$ (1)

In [3]:
```
h_b = (1./2.) - (1./2.) * sigmax()
h_b
```

Out [3]:
Quantum object: dims = [[2], [2]], shape = [2, 2], type = oper, isHerm = True $\begin{pmatrix} 0.5 & -0.5 \\ -0.5 & 0.5 \end{pmatrix}$ (2)

In [4]:
```
# increase taumax to get make the sweep more adiabatic
taumax = 5.0
taulist = linspace(0, taumax, 100)
```

In [5]:
```
h_t = [[h_b, lambda t, t_max : (t_max-t)/t_max],
       [h_p, lambda t, t_max : t/t_max]]
h_t
```

Out [5]:
```
[[Quantum object: dims = [[2], [2]], shape = [2, 2], type = oper,
isherm = True
Qobj data =
[[ 0.5 -0.5]
 [-0.5  0.5]],
  <function __main__.<lambda>>],
 [Quantum object: dims = [[2], [2]], shape = [2, 2], type = oper,
isherm = True
Qobj data =
[[ 1.  0.]
 [ 0.  0.]],
  <function __main__.<lambda>>]]
```

In [6]:
```
psi0 = basis(2,0)
psi0
```

Out [6]:
Quantum object: dims = [[2], [1]], shape = [2, 1], type = ket $\begin{pmatrix} 1.0 \\ 0.0 \end{pmatrix}$ (3)

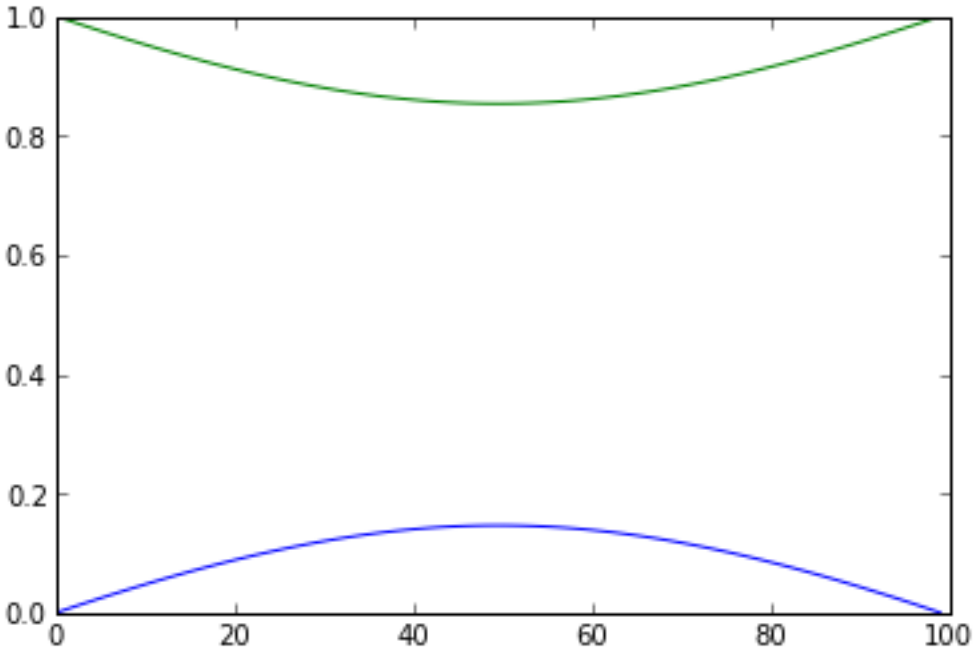
In [7]:
```
evals_mat = zeros((len(taulist), 2))
P_mat = zeros((len(taulist), 2))
```

```
idx = [0]
def process_rho(tau, psi):
    H = qobj_list_evaluate(h_t, tau, taumax)
    evals, ekets = H.eigenstates()

    evals_mat[idx[0],:] = real(evals)

    # find the overlap between the eigenstates and psi
    for n, eket in enumerate(ekets):
        P_mat[idx[0],n] = abs((eket.dag().data * psi.data)[0,0])**2

    idx[0] += 1

mesolve(h_t, psi0, taulist, [], process_rho, taumax)
```

Out [7]:
```
Odedata object with mesolve data.
---------------------------------
states = True, expect = True
num_expect = 0, num_collapse = 0
```

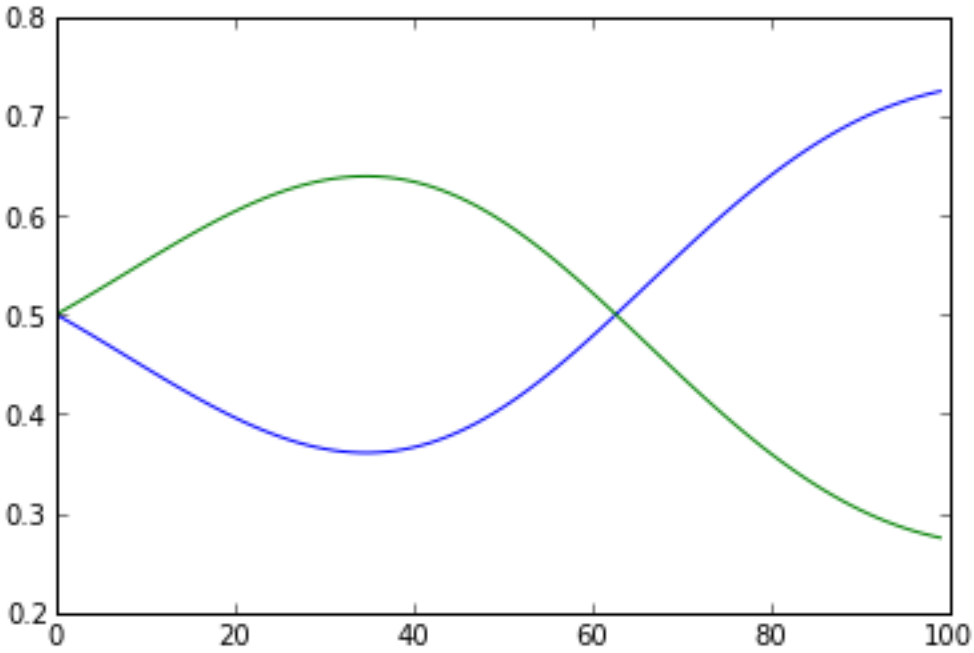
```
plot(evals_mat)
```

In [8]:
```
[<matplotlib.lines.Line2D at 0x43d5b50>,
 <matplotlib.lines.Line2D at 0x43d5dd0>]
```




```
plot(P_mat)
```

In [9]:
```
[<matplotlib.lines.Line2D at 0x45d0890>,
 <matplotlib.lines.Line2D at 0x45d0b10>]
```


0.8
0.7
0.6
0.5
0.4
0.3
0.2
0
20
40
60
80
100