# Listing 6.20: Coherent states
Petridis Petros

June 19, 2022

[1]:

```
from numpy import *
import numpy as np , matplotlib.pyplot as plt, math
from PIL import Image

sqpi=np.sqrt(np.pi)#
E=3.0 #
alpha=np.sqrt(E-0.5) #||
factr=np.exp(0.5*alpha*alpha) # ||^2/2


def Hermite(x,n):
    if(n==0): #
        p=1.0
    elif(n==1): #
        p=2*x
    else: # >=2
        p0=1
        p1=2*x
        for i in range(1,n):
            p2=2*x*p1-2*i*p0
            p0=p1
            p1=p2
            p=p2
    return p

def glauber(x,t,nmax):
    Reterm=0.0
    Imterm=0.0

    factr=np.exp(-0.5*alpha*alpha) #exp(-||^2/2)

    for n in range(0,nmax+1):
        fact=np.sqrt(1.0/(math.factorial(n)*sqpi*(2**n)))
        psin=fact*Hermite(x,n)*np.exp(-0.5*x*x)

        den=np.sqrt(math.factorial(n))
```

```
        num=factr*(alpha**n)*psin

        # exp[-i(n+1/2)]=cos(n+1/2) -isin(n+1/2)
        #Sum real + imaginary  nmax
        Reterm+=num*(np.cos((n+0.5)*t))/den
        Imterm+=num*(np.sin((n+0.5)*t))/den

    #  coherent states ||^2 = * = Real^2 + Imaginary^2 -> P=(*)
    phi=Reterm*Reterm+Imterm*Imterm
    return phi

#   plot    (x,t),
def animate(t):
    y=glauber(xx,t,nmax)
    s=str(t)
    plt.plot(xx,y,label=s)
    leg=plt.legend(loc="best",ncol=4,mode="expand",shadow=True)
    #legend ,
```

[2]:
```
#Question 1
im = Image.open("C:/Users/petro/Desktop/as.png")
im
```

[2]:

**Ερώτηση 1:** coherent states ιδιοκαταστάσεις $\hat{\alpha}$ και $\hat{H}$.

Coherent states είναι οι ιδιοκαταστάσεις του τελεστή καταστροφής, δηλαδή ισχύει: $\hat{a}|\alpha>=\alpha|\alpha>$.

Απόδηξη:

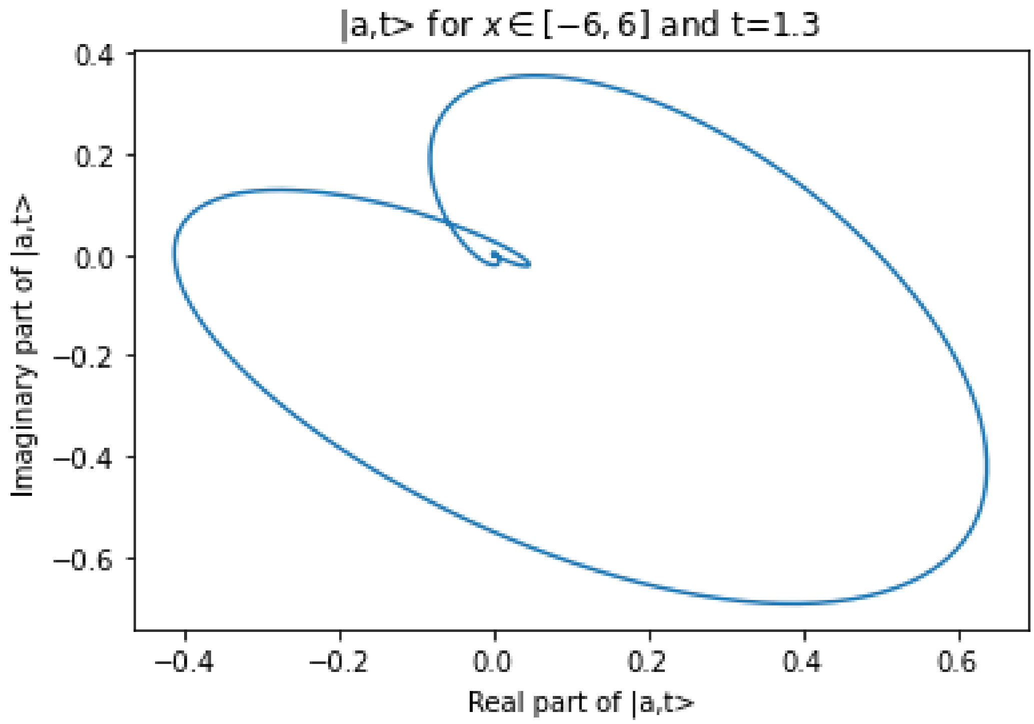
$$
\begin{aligned}
\hat{a}|\alpha> &= \hat{a}e^{-|a|^2/2}\sum_{n=0}^{\infty}\frac{a^n}{\sqrt{n!}}|n> \\
&= e^{-|a|^2/2}\sum_{n=0}^{\infty}\frac{a^n}{\sqrt{n!}}\hat{a}|n> \\
&= e^{-|a|^2/2}\sum_{n=1}^{\infty}\frac{a^n}{\sqrt{n!}}\sqrt{n}|n-1> \\
&= e^{-|a|^2/2}\sum_{n'=0}^{\infty}\frac{a^{n'+1}}{\sqrt{(n'+1)!}}\sqrt{n'}|n'> \\
&= \alpha e^{-|a|^2/2}\sum_{n'=0}^{\infty}\frac{a^{n'}}{\sqrt{(n')!}}|n'> \\
&= \alpha|\alpha>
\end{aligned}
$$

κι αφού υπάρχει ο όρος $|n>$ μέσω στην κατάσταση των coherent states θα είναι και ιδιοκαταστάσεις της Χάμιλον.

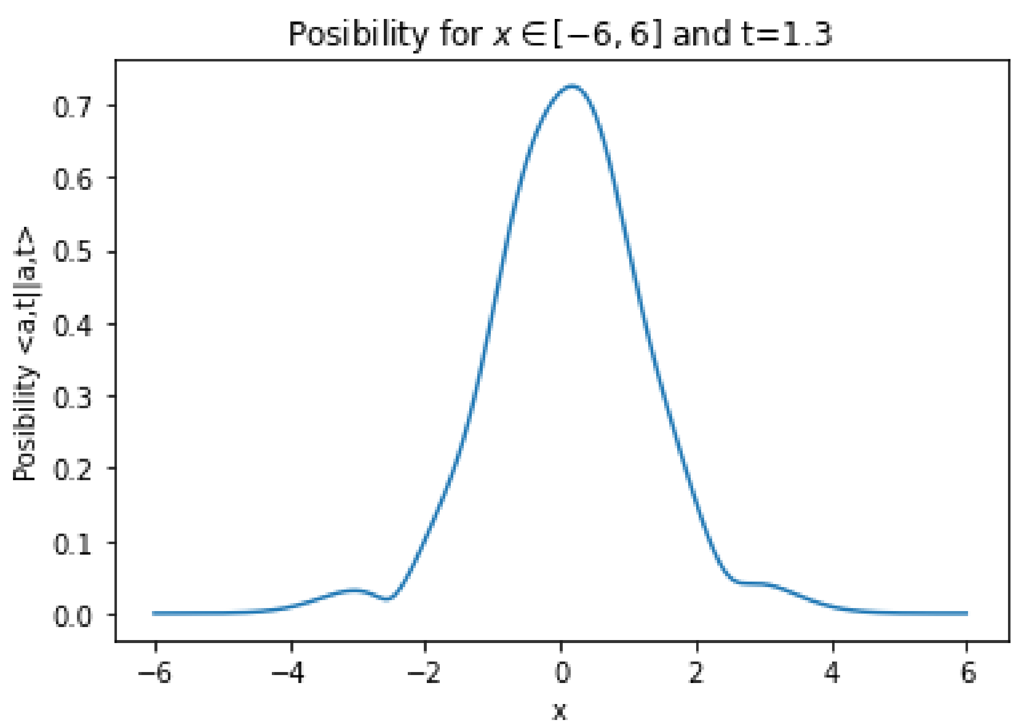
[3]:
```
#Question 2
#|a,t> nmax=5
nmax=5
```

```python
E=2
a=np.sqrt(E-0.5)
def coherent(x,t,nmax,alpha):
    Reterm=0.0
    Imterm=0.0

    factr=np.exp(-0.5*abs(alpha**2)) #exp(-||^2/2)

    for n in range(0,nmax+1):
        fact=np.sqrt(1.0/(math.factorial(n)*sqpi*(2**n)))
        psin=fact*Hermite(x,n)*np.exp(-0.5*x*x)

        den=np.sqrt(math.factorial(n))

        num=factr*(alpha**n)*psin

        # exp[-i(n+1/2)]=cos(n+1/2) -isin(n+1/2)
        Reterm+=num*(np.cos((n+0.5)*t))/den
        Imterm-=num*(np.sin((n+0.5)*t))/den

    return [Reterm,Imterm]
x=np.linspace(-6,6,1000)
plt.figure()
plt.plot(coherent(x,1.3,nmax,a)[0],coherent(x,1.5,5,a)[1])
plt.xlabel("Real part of |a,t>")
plt.ylabel("Imaginary part of |a,t>")
plt.title("|a,t> for ""$x\in[-6,6]$"" and t=1.3")

plt.figure()
plt.plot(x,np.sqrt(coherent(x,1.5,nmax,a)[0]**2+coherent(x,1.5,5,a)[1]**2))
plt.xlabel("x")
plt.ylabel("Posibility <a,t||a,t>")
plt.title("Posibility for ""$x\in[-6,6]$"" and t=1.3")
```

```
[3]: Text(0.5, 1.0, 'Posibility for $x\\in[-6,6]$ and t=1.3')
```



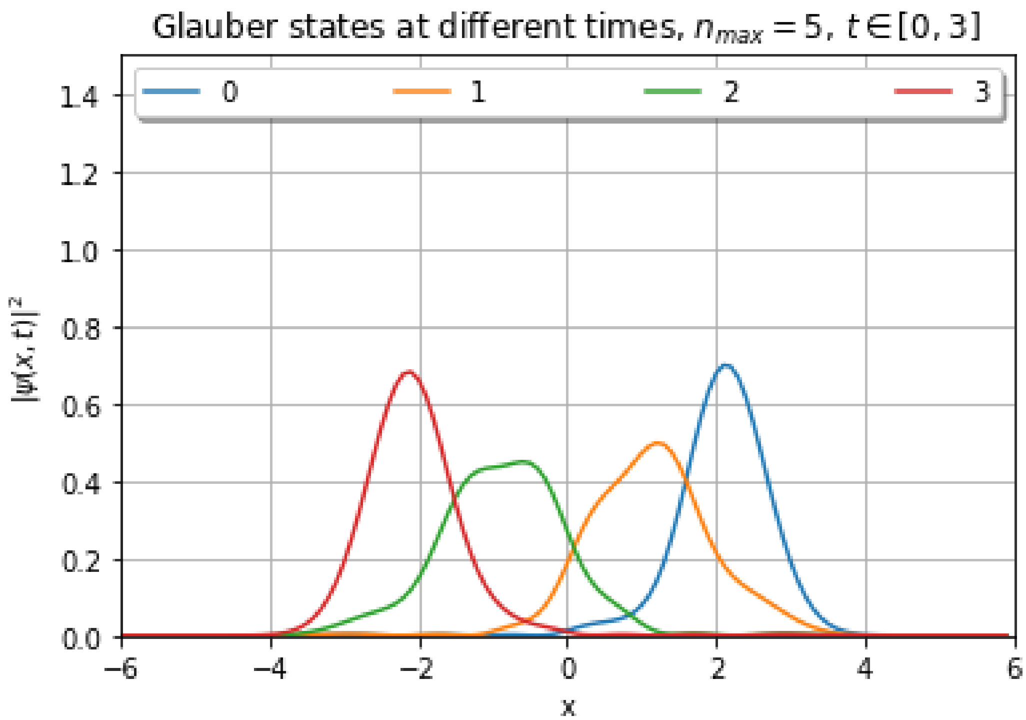
|a,t> for $x \in [-6, 6]$ and t=1.3
Imaginary part of |a,t>
Real part of |a,t>
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.4
−0.2
0.0
0.2
0.4
0.6

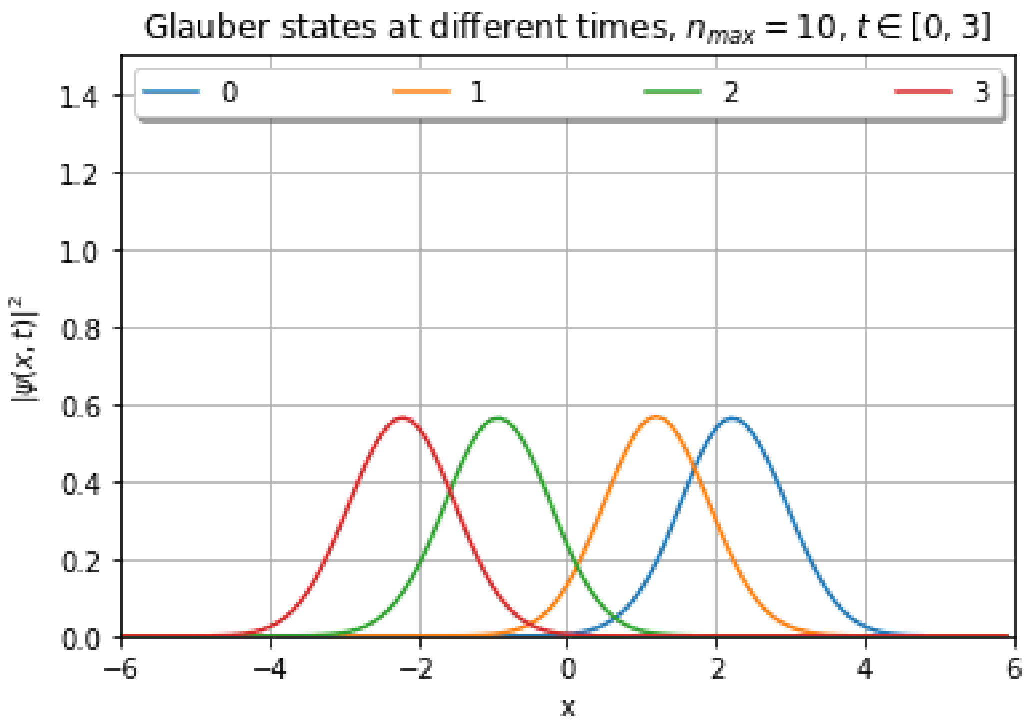



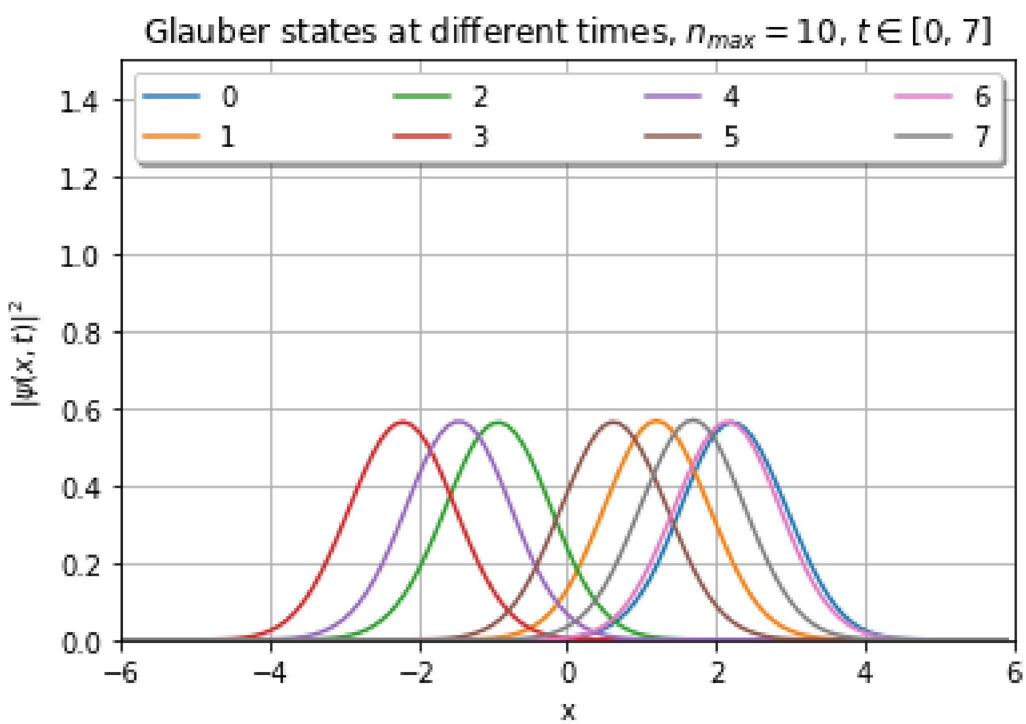
Posibility for $x \in [-6, 6]$ and t=1.3
Posibility <a,t||a,t>
x
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−6
−4
−2
0
2
4
6

```
[4]: #Question 4
#nmax=5
#Energy=3
E=3
nmax=5
a=np.sqrt(E-1/2)

fig=plt.figure()
ax=fig.add_subplot(111,autoscale_on=True,xlim=(-6,6),ylim=(0,1.5))
ax.grid()

plt.title("Glauber states at different times, ""$n_{max}=$"f"%d, " "$t\in␣
 ↪[0,3]$"%nmax)
plt.xlabel("x")
plt.ylabel("$|\psi(x,t)|^2$")

xx=np.arange(-6.0,6.0,0.1)

for t in np.arange(0,4,1):
    animate(t)


nmax=10
fig=plt.figure()
ax=fig.add_subplot(111,autoscale_on=True,xlim=(-6,6),ylim=(0,1.5))
ax.grid()

plt.title("Glauber states at different times, ""$n_{max}=$"f"%d, " "$t\in␣
 ↪[0,3]$"%nmax)
plt.xlabel("x")
plt.ylabel("$|\psi(x,t)|^2$")

xx=np.arange(-6.0,6.0,0.1)

for t in np.arange(0,4,1):
    animate(t)


nmax=10
fig=plt.figure()
ax=fig.add_subplot(111,autoscale_on=True,xlim=(-6,6),ylim=(0,1.5))
ax.grid()

plt.title("Glauber states at different times, ""$n_{max}=$"f"%d, " "$t\in␣
 ↪[0,7]$"%nmax)
plt.xlabel("x")
plt.ylabel("$|\psi(x,t)|^2$")
```

```
xx=np.arange(-6.0,6.0,0.1)

for t in np.arange(0,8,1):
    animate(t)
```




Glauber states at different times, $n_{max} = 10$, $t \in [0, 3]$
0
1
2
3
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
$|\psi(x,t)|^2$
−6
−4
−2
0
2
4
6
x


Glauber states at different times, $n_{max} = 10$, $t \in [0, 7]$
0
1
2
3
4
5
6
7
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
$|\psi(x,t)|^2$
−6
−4
−2
0
2
4
6
x

```
[ ]:
```