Task A

May 11, 2025

```python
[18]: import dynamiqs as dq
      import jax.numpy as jnp
      import jax
      from dynamiqs import *
      import matplotlib.pyplot as plt

      import imageio
```

```python
[19]: #parameters
      pi=jnp.pi
      N=50 #Hilbert space dimension
      n=2 #number Fock state
      alpha=2 #coherent state amplitude
      g2=1.0
      k_b=10
      e_d=-4

      T       = 4.0       # total time
      n_steps = 80        # number frame
      tlist  = jnp.linspace(0, T, n_steps)
```

```python
[20]: #operators and states
      a=destroy(N)
      I  = eye(N)
      a_sys  = tensor(a,  I)
      adag_sys = a_sys.dag()
      b_sys  = tensor(I,  a)
      bdag_sys = b_sys.dag()
      H=g2.conjugate()*adag_sys@adag_sys@b_sys+g2*a_sys@a_sys@bdag_sys+e_d.
        ↪conjugate()*b_sys+e_d*bdag_sys

      c_ops = [jnp.sqrt(k_b) * b_sys]

      vacuum_state=tensor(basis(N,0), basis(N,0))

      fock_state=fock(N, n)
      coherent_state=coherent(N, alpha)
      cat_state=1/jnp.sqrt(2)*(coherent(N,alpha)+coherent(N,-alpha))
```

```
cat_state_3=1/jnp.sqrt(3)*(coherent(N,alpha)+coherent(N,jnp.exp(1j*2*pi/
↪3)*alpha)+coherent(N,jnp.exp(-1j*2*pi/3)*alpha))
```

[21]:
```
#Wigner functions
"""gridspace=jnp.linspace(-5,5, 100)
W_fock = wigner(fock_state, xvec=gridspace, yvec=gridspace)
W_coherent = wigner(coherent_state, xvec=gridspace, yvec=gridspace)"""
W_coherent=wigner(coherent_state, 5,5)
W_fock2 = wigner(fock_state,5,5)
```

[22]:
```
xvec, yvec, W_fock=W_fock2
```

[23]:
```
plt.contourf(xvec, yvec,W_fock, levels=100,cmap="seismic", vmax=1, vmin=-1)
plt.xlabel("x")
plt.ylabel("p")
plt.colorbar()
```
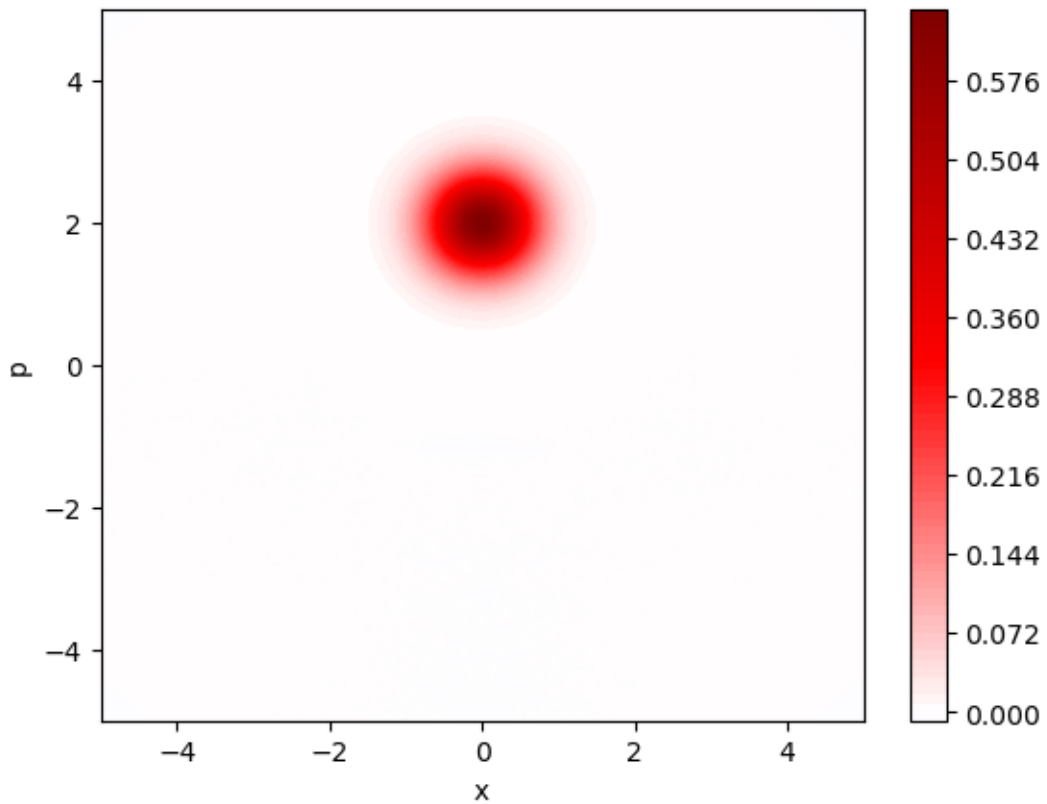
[23]: <matplotlib.colorbar.Colorbar at 0x310d9b410>
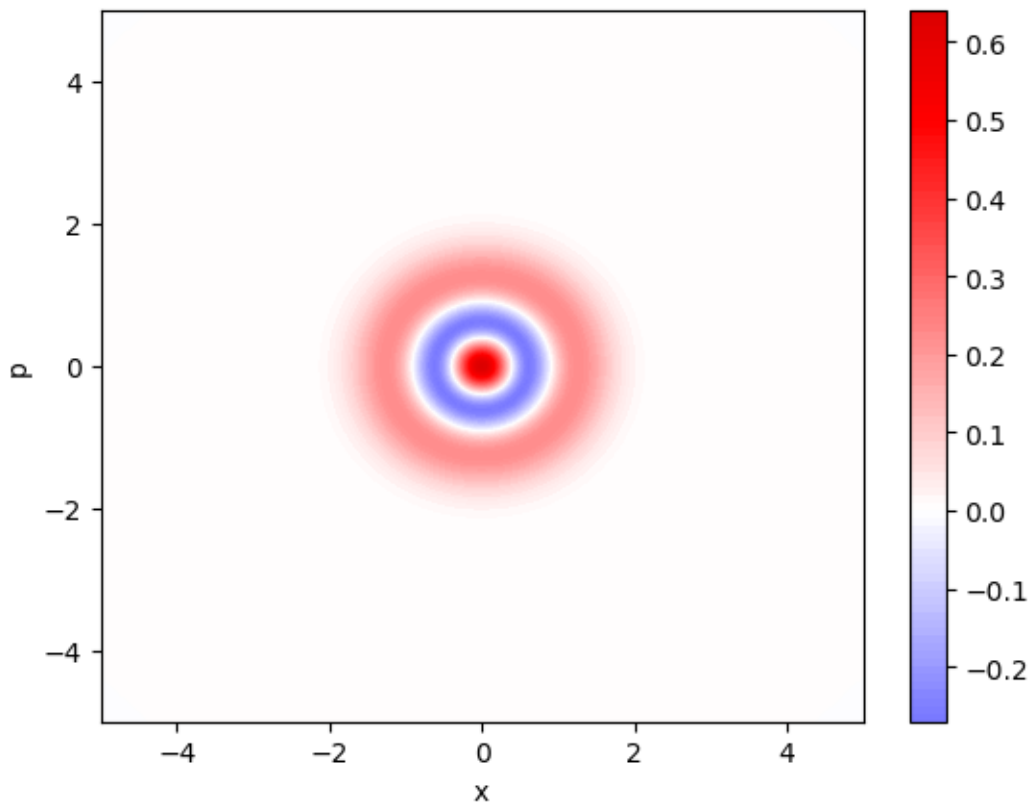
[24]:
```
xvec,yvec, W_coherent=W_coherent
```
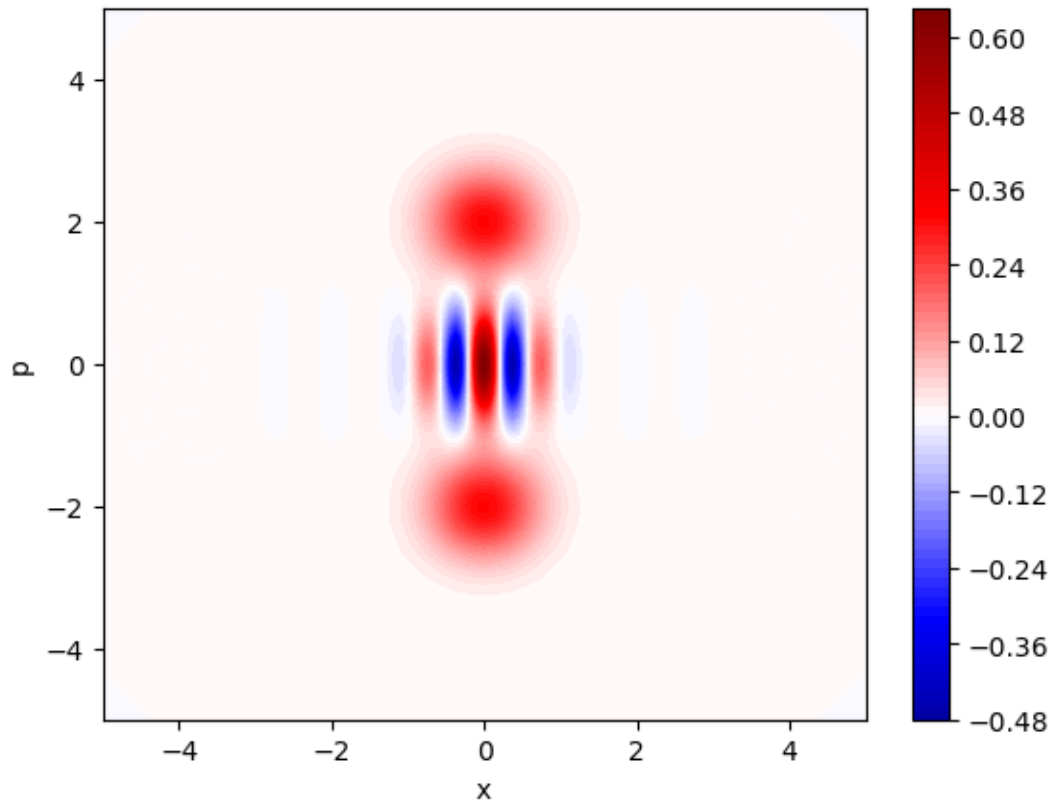
```
plt.contourf(xvec, yvec,W_coherent, levels=100,cmap="seismic", vmax=2/pi,
   ↪vmin=-2/pi)
plt.xlabel("x")
plt.ylabel("p")
plt.colorbar()
```
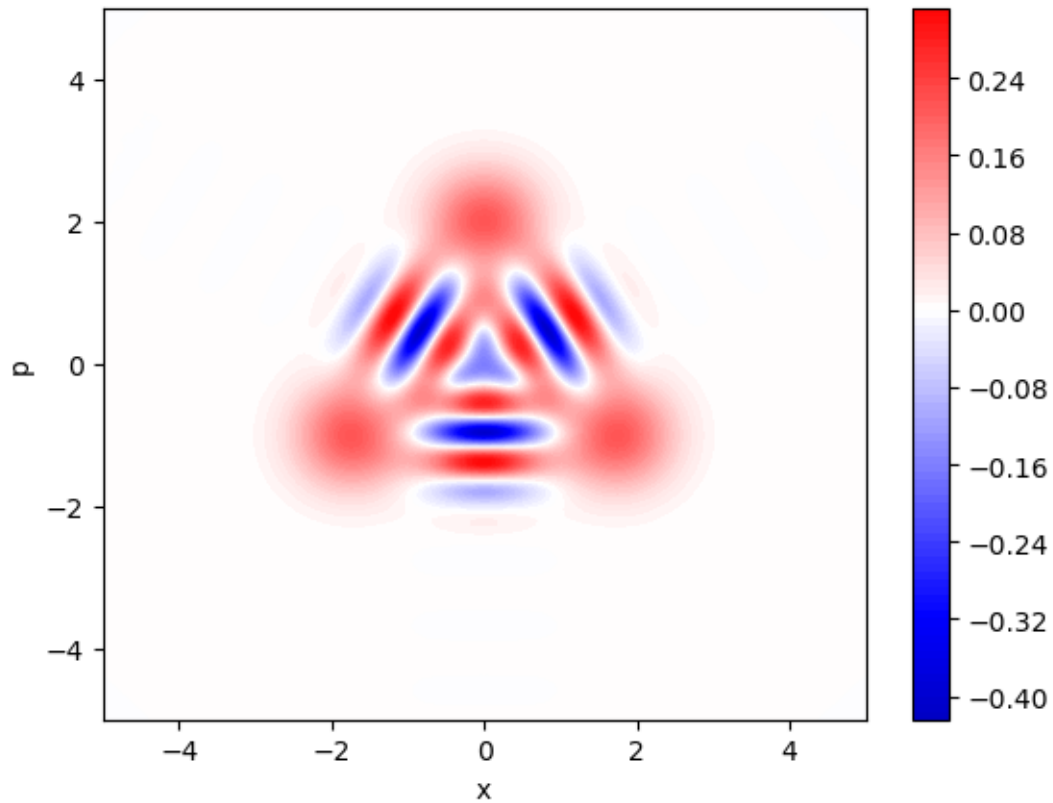
[24]: <matplotlib.colorbar.Colorbar at 0x32b0fbfe0>

[25]:
```
xvec,yvec, W_coherent=wigner(cat_state, 5,5)
plt.contourf(xvec, yvec,W_coherent.T, levels=100,cmap="seismic", vmax=2/pi,
   ↪vmin=-2/pi)
plt.xlabel("x")
plt.ylabel("p")
plt.colorbar()
```

[25]: <matplotlib.colorbar.Colorbar at 0x310dc60f0>

```
[26]: xvec,yvec, W_coherent=wigner(cat_state_3, 5,5)
      plt.contourf(xvec, yvec,W_coherent.T, levels=100,cmap="seismic", vmax=2/pi,␣
        ↪vmin=-2/pi)
      plt.xlabel("x")
      plt.ylabel("p")
      plt.colorbar()
```

```
[26]: <matplotlib.colorbar.Colorbar at 0x121075670>
```

```
result = mesolve(H, c_ops, vacuum_state, tlist)

filenames = []
for idx, rho_ab in enumerate(result.states):

    rho_a = rho_ab.ptrace(0)

    xvec, yvec, W = wigner(rho_a, 5, 5)

    # plot
    fig, ax = plt.subplots(figsize=(4,4))
    c = ax.contourf(xvec, yvec,W, levels=100,cmap="seismic", vmax=2/pi, vmin=-2/
 ↪pi)
    ax.set_title(f"t = {tlist[idx]:.2f}")
    ax.set_xlabel("x")
    ax.set_ylabel("p")

    #saving frame
```

| | 0.2%   elapsed 9.81s   remaining 01h05m11s