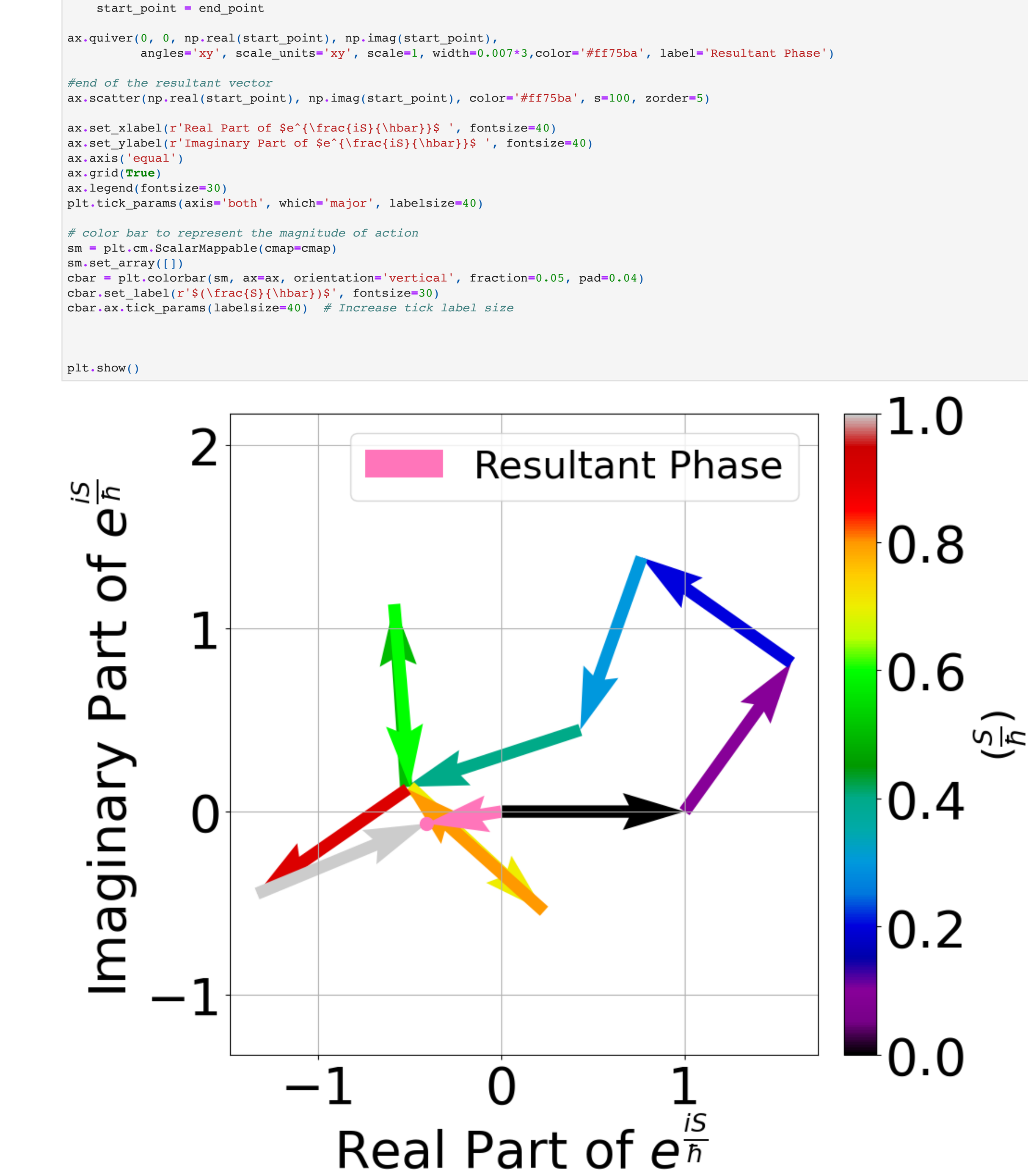
In [3]:
```
import matplotlib.pyplot as plt
import numpy as np

# Define actions and phases
actions_extreme = np.array([1e-40, 1e-34, 1e-20, 1e-10, 1e-5, 1e5, 1e10, 1e20, 1e25, 1e30, 1e35])
h_bar_si = 1.0545718e-34  # Planck's constant in m^2 kg / s
phases_extreme = np.exp(1j * actions_extreme / h_bar_si)
log_actions = np.log10(actions_extreme/h_bar_si)


fig, ax = plt.subplots(figsize=(9, 9))
start_point = 0 + 0j  # Start at the origin

#'nipy_spectral' colormap
cmap = plt.get_cmap('nipy_spectral')
colors = cmap(np.linspace(0, 1, len(log_actions)))

# vectors using head-to-tail
for i, phase in enumerate(phases_extreme):
    end_point = start_point + phase
    color = colors[i]
    ax.quiver(np.real(start_point), np.imag(start_point), np.real(phase), np.imag(phase),
              angles='xy', scale_units='xy', scale=1,  width=0.007*3, color=color)

    # start point for the next vector
    start_point = end_point

ax.quiver(0, 0, np.real(start_point), np.imag(start_point),
          angles='xy', scale_units='xy', scale=1, width=0.007*3,color='#ff75ba', label='Resultant Phase')

#end of the resultant vector
ax.scatter(np.real(start_point), np.imag(start_point), color='#ff75ba', s=100, zorder=5)

ax.set_xlabel(r'Real Part of $e^{\frac{iS}{\hbar}}$ ', fontsize=40)
ax.set_ylabel(r'Imaginary Part of $e^{\frac{iS}{\hbar}}$ ', fontsize=40)
ax.axis('equal')
ax.grid(True)
ax.legend(fontsize=30)
plt.tick_params(axis='both', which='major', labelsize=40)

# color bar to represent the magnitude of action
sm = plt.cm.ScalarMappable(cmap=cmap)
sm.set_array([])
cbar = plt.colorbar(sm, ax=ax, orientation='vertical', fraction=0.05, pad=0.04)
cbar.set_label(r'$(\frac{S}{\hbar})$', fontsize=30)
cbar.ax.tick_params(labelsize=40)  # Increase tick label size



plt.show()
```



In [ ]: