import matplotlib.pyplot as plt

import numpy as np

setup\_costs = np.arange(0, 5.5, 0.5)

avg\_cost\_reallocate = [

7.605778, 7.657072, 7.708366, 7.759660, 7.810954,

7.862248, 7.913542, 7.964836, 8.016330, 8.067551, 8.118773

]

avg\_cost\_combined = [

5.090796, 5.161244, 5.231693, 5.302142, 5.372590,

5.443039, 5.513488, 5.583937, 5.654385, 5.724834, 5.795283

]

avg\_cost\_maintain = [

7.983187, 8.074353, 8.165518, 8.256684, 8.347850,

8.439015, 8.530181, 8.621347, 8.712512, 8.803678, 8.894843

]

# درصد افزایش نسبت به مقدار پایه

base\_reallocate = avg\_cost\_reallocate[0]

base\_combined = avg\_cost\_combined[0]

base\_maintain = avg\_cost\_maintain[0]

increase\_reallocate = (np.array(avg\_cost\_reallocate) - base\_reallocate) / base\_reallocate \* 100

increase\_maintain = (np.array(avg\_cost\_maintain) - base\_maintain) / base\_maintain \* 100

plt.figure(figsize=(11,5))

# نمودار (a)

plt.subplot(1,2,1)

plt.plot(setup\_costs, avg\_cost\_combined, '-s',color='blue', label='reallocate-and-maintain policy', linewidth=1.5, markersize=5)

plt.plot(setup\_costs, avg\_cost\_maintain, '-o', color='red',label='maintain-only policy', linewidth=1.5, markersize=5)

plt.plot(setup\_costs, avg\_cost\_reallocate, '-^',color='orange', label='reallocate-only policy', linewidth=1.5, markersize=5)

plt.xlabel('Setup cost for maintenance')

plt.ylabel('Minimal expected cost per time unit')

plt.title('(a) Minimal average cost per unit time', fontsize=11)

plt.legend(frameon=False)

plt.tick\_params(axis='both', direction='in')

plt.box(True)

plt.grid(False)

# نمودار (b)

plt.subplot(1,2,2)

plt.plot(setup\_costs, increase\_maintain, '-', color='black', label='maintain-only policy', linewidth=1.5)

plt.plot(setup\_costs, increase\_reallocate, '--', color='black', label='reallocate-only policy', linewidth=1.5)

plt.xlabel('Setup cost for maintenance')

plt.ylabel('Percentage increase in cost (%)')

plt.title('(b) Cost increase (%)', fontsize=11)

plt.legend(frameon=False)

plt.tick\_params(axis='both', direction='in')

plt.box(True)

plt.grid(False)

plt.tight\_layout()

plt.show()