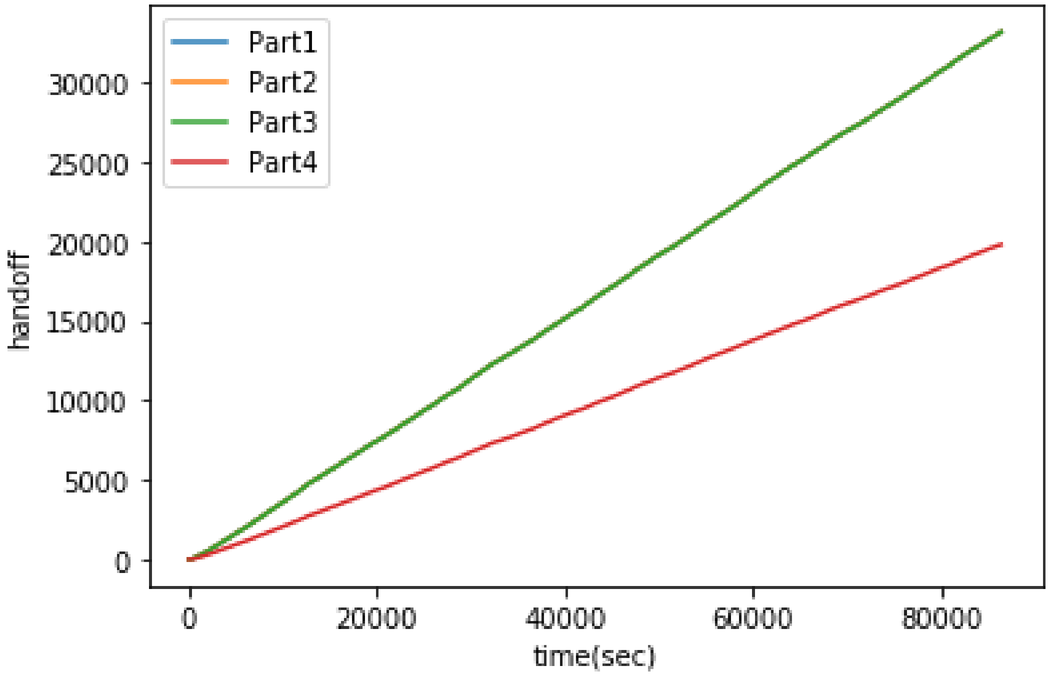
無線 project 書面報告

F74064012 黃冠淳

1. 圖表:



Handoff 數量:

(1) Best: 33427
(2) Threshold : 33427
(3) Entropy : 33413
(4) My policy: 19951

由於 Best 與 Threshold 的 handoff 數量一樣，Entropy 與兩者差一點，

三條線看起來像同一條

Policy 的總平均 Power:

(1)Best: -114.88284163152136
(2)Threshold : -114.88284163152136
(3)Entropy : -115.10790660143336
(4)My policy: -117.32804959387406

2. Source code:

```
import math
import random
import matplotlib.pyplot as plt

handoffNumPart1 = 0
handoffNumPart2 = 0
handoffNumPart3 = 0
handoffNumPart4 = 0
time = []
totalHandoffPart1 = []
totalHandoffPart2 = []
totalHandoffPart3 = []
totalHandoffPart4 = []
power1=0
power2=0
power3=0
power4=0

temp=0

Pmin=-125

#print("time: ", time)
#print("totalHandoffPart1: ", totalHandoffPart1)
#print("totalHandoffPart2: ", totalHandoffPart2)
#print("totalHandoffPart3: ", totalHandoffPart3)
#print("totalHandoffPart4: ", totalHandoffPart4)
index = 0
posX = 0
posY = 0
direction = 0
tower = 0
tower2=0
tower3=0
tower4=0
car_list = []
```

```
car_info = [index, posX, posY, direction, tower,tower2,tower3,tower4]

#print("car list:", car_list)
#print("car info: ", car_info)

#Part 1
time = []
totalHandoffPart1 = []
for i in range(0, 86400):
    # check car list
    if(len(car_list)!=0):
        # car move
        for car in car_list:
            # turn or not
            if car[1]%75 == 0 and car[2]%75 == 0:
                turnDir = random.randint(1, 6)
                if car[3] == 1:
                    if car[1]== 0 and car[2] == 0:
                        car[3] = 2
                    elif car[1]== 300 and car[2] == 0:
                        car[3] = 4
                    else:
                        if turnDir==1 or turnDir==2 or turnDir==3:
                            car[3] = car[3]
                        elif turnDir==4 or turnDir==5:
                            car[3] = 2
                        elif turnDir==6:
                            car[3] = 4
                elif car[3] == 2:
                    if car[1]== 300 and car[2] == 0:
                        car[3] = 3
                    elif car[1]== 300 and car[2] == 300:
                        car[3] = 1
                    else:
                        if turnDir==1 or turnDir==2 or turnDir==3:
                            car[3] = car[3]
                        elif turnDir==4 or turnDir==5:
                            car[3] = 3
                        elif turnDir==6:
                            car[3] = 1
                elif car[3] == 3:
                    if car[1]== 0 and car[2] == 300:
                        car[3] = 2
                    elif car[1]== 300 and car[2] == 300:
                        car[3] = 4
                    else:
                        if turnDir==1 or turnDir==2 or turnDir==3:
                            car[3] = car[3]
                        elif turnDir==4 or turnDir==5:
                            car[3] = 4
                        elif turnDir==6:
                            car[3] = 2
                elif car[3] == 4:
                    if car[1]== 0 and car[2] == 0:
                        car[3] = 3
                    elif car[1]== 0 and car[2] == 300:
                        car[3] = 1
                    else:
                        if turnDir==1 or turnDir==2 or turnDir==3:
                            car[3] = car[3]
                        elif turnDir==4 or turnDir==5:
                            car[3] = 1
                        elif turnDir==6:
                            car[3] = 3

            # move
            if car[3] == 1:
                car[2] -= 1
            elif car[3] == 2:
                car[1] += 1
            elif car[3] == 3:
                car[2] += 1
            elif car[3] == 4:
                car[1] -= 1
            if car[1]<0 or car[1]>300 or car[2]<0 or car[2]>300:
                car_list.remove(car)
```

```
            else:
                # check handoff
                ############# P A R T 1 ####################
                dis = math.sqrt((car[1]-75)*(car[1]-75) +
(car[2]-75)*(car[2]-75))*10
                if dis == 0:
                    P1=-50
                else:
                    P1 = -60-20*math.log(dis, 10)

                # tower 2
                dis = math.sqrt((car[1]-225)*(car[1]-225) +
(car[2]-75)*(car[2]-75))*10
                if dis == 0:
                    P2=-50
                else:
                    P2 = -60-20*math.log(dis, 10)

                # tower 3
                dis = math.sqrt((car[1]-225)*(car[1]-225) +
(car[2]-225)*(car[2]-225))*10
                if dis == 0:
                    P3=-50
                else:
                    P3 = -60-20*math.log(dis, 10)

                # tower 4
                dis = math.sqrt((car[1]-75)*(car[1]-75) +
(car[2]-225)*(car[2]-225))*10
                if dis == 0:
                    P4=-50
                else:
                    P4 = -60-20*math.log(dis, 10)

                towerP = [P1, P2, P3, P4]
                Pold = towerP[car[4]-1]
                Pnew = max(towerP)
                temp+=1
                if Pnew>Pold:
                    car[4] = towerP.index(Pnew) + 1
                    handoffNumPart1+=1
                power1=power1+towerP[car[4]-1]

                ############P A R T2##################

                Pold = towerP[car[5]-1]
                T = -110
                if Pnew>Pold and Pold<T:
                    car[5] = towerP.index(Pnew) + 1
                    handoffNumPart2+=1
                power2=power2+towerP[car[5]-1]



############P A RT3##################

                Pold = towerP[car[6]-1]
                if Pold<T and Pnew>Pold+5:
                    car[6] = towerP.index(Pnew) + 1
                    handoffNumPart3+=1
                power3=power3+towerP[car[6]-1]

        ############P A R T 4####################

                Pold = towerP[car[7]-1]
                if Pold<Pmin and Pnew>Pold+5:
                    car[7] = towerP.index(Pnew) + 1
                    handoffNumPart4+=1
                power4=power4+towerP[car[7]-1]
    # create new car or not
    for pos in range(0,12):
        PnewCar = random.randint(1, 31)
        if PnewCar == 1:

            if pos == 1:
```

```
        posX = 75
        posY = 0
        turnDir = random.randint(1, 6)
        if turnDir==1 or turnDir==2 or turnDir==3:
            direction = 3
        elif turnDir==4 or turnDir==5:
            direction= 4
        elif turnDir==6:
            direction = 2

        tower = 1
        tower2 = 1
        tower3 = 1
        tower4 = 1
    elif pos == 2:
        posX = 150
        posY = 0
        turnDir = random.randint(1, 6)
        if turnDir==1 or turnDir==2 or turnDir==3:
            direction = 3
        elif turnDir==4 or turnDir==5:
            direction= 4
        elif turnDir==6:
            direction = 2
        tower = 1
        tower2 = 1
        tower3 = 1
        tower4 = 1
    elif pos == 3:
        posX = 225
        posY = 0
        turnDir = random.randint(1, 6)
        if turnDir==1 or turnDir==2 or turnDir==3:
            direction = 3
        elif turnDir==4 or turnDir==5:
            direction= 4
        elif turnDir==6:
            direction = 2
    tower = 2
    tower2 = 2
    tower3 = 2
    tower4 = 2
elif pos == 4:
    posX = 300
    posY = 75
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 4
    elif turnDir==4 or turnDir==5:
        direction =1
    elif turnDir==6:
        direction = 3
    tower = 2
    tower2 = 2
    tower3 = 2
    tower4 = 2
elif pos == 5:
    posX = 300
    posY = 150
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 4
    elif turnDir==4 or turnDir==5:
        direction= 1
    elif turnDir==6:
        direction = 3
    tower = 2
    tower2 = 2
    tower3 = 2
    tower4 = 2
elif pos == 6:
    posX = 300
    posY = 225
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 4
    elif turnDir==4 or turnDir==5:
        direction = 1
    elif turnDir==6:
        direction = 3
    tower = 3
    tower2 = 3
    tower3 = 3
    tower4 = 3
elif pos == 7:
    posX = 225
    posY = 300
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 1
    elif turnDir==4 or turnDir==5:
        direction = 2
    elif turnDir==6:
        direction = 4
    tower = 3
    tower2 = 3
    tower3 = 3
    tower4 = 3
elif pos == 8:
    posX = 150
    posY = 300
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 1
    elif turnDir==4 or turnDir==5:
        direction = 2
    elif turnDir==6:
        direction = 4
    tower = 3
    tower2 = 3
    tower3 = 3
    tower4 = 3
elif pos == 9:
    posX = 75
    posY = 300
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 1
    elif turnDir==4 or turnDir==5:
        direction = 2
    elif turnDir==6:
        direction = 4
    tower = 4
    tower2 = 4
    tower3 = 4
    tower4 = 4
elif pos == 10:
    posX = 0
    posY = 225
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 2
    elif turnDir==4 or turnDir==5:
        direction = 3
    elif turnDir==6:
        direction = 1
    tower = 4
    tower2 = 4
    tower3 = 4
    tower4 = 4
elif pos == 11:
    posX = 0
    posY = 150
    turnDir = random.randint(1, 6)
    if turnDir==1 or turnDir==2 or turnDir==3:
        direction = 2
    elif turnDir==4 or turnDir==5:
        direction = 3
    elif turnDir==6:
        direction = 1
    tower = 4
    tower2 = 4
```

```
                tower3 = 4
                tower4 = 4
            elif pos == 12:
                posX = 0
                posY = 75
                turnDir = random.randint(1, 6)
                if turnDir==1 or turnDir==2 or turnDir==3:
                    direction = 1
                elif turnDir==4 or turnDir==5:
                    direction = 2
                elif turnDir==6:
                    direction = 4
                tower = 1
                tower2 = 1
                tower3 = 1
                tower4 = 1
            car_info = [index, posX, posY, direction,
tower,tower2,tower3,tower4]
            car_list.append(car_info)
            index += 1
    # end for loop: new car

    time.append(i+1)
    totalHandoffPart1.append(handoffNumPart1)


    totalHandoffPart2.append(handoffNumPart2)


    totalHandoffPart3.append(handoffNumPart3)


    totalHandoffPart4.append(handoffNumPart4)

print(power1/temp)
print(power2/temp)
print(power3/temp)
print(power4/temp)
```

```
plt.plot(time, totalHandoffPart1)
plt.plot(time, totalHandoffPart2)
plt.plot(time, totalHandoffPart3)
plt.plot(time, totalHandoffPart4)
plt.legend(["Part1", "Part2", "Part3","Part4"])
plt.xlabel("time(sec)")
plt.ylabel("handoff")
print(handoffNumPart1)
print(handoffNumPart2)
print(handoffNumPart3)
print(handoffNumPart4)
```

3. Introduction to your policy

(1)Best policy: Pnew > Pold

選訊號強度高的基地台

(2) Threshold : Pnew > Pold & Pold < T

定一個 T 是通話最低需求訊號，低於 T 才換基地台

(3) Entropy : Pnew > Pold + E

定一個訊號差值，Pnew – Pold >E 才換基地台

(4)my policy:

結合 Threshold 跟 Entropy，定一個 Pmin 與訊號差值是通話最低需求訊號，低於 Pmin 且 Pnew – Pold >E 才換基地台

特性:結合 Threshold 跟 Entropy 的優點，使 Handoff 數量達到最低，但缺點是，平均 power 也最低