AAE2004 INTRODUCTION TO AVIATION SYSTEMS
AAE
DESIGN OF PATH PLANNING ALGORITHM FOR AIRCRAFT OPERATION

Group 9

GROUP MEMBERS:

TURALY YERLAN

TAM CHUNG HONG FREDERICK

THAPA NAMNET

TSANG MAN TIM

TSE PUI YI

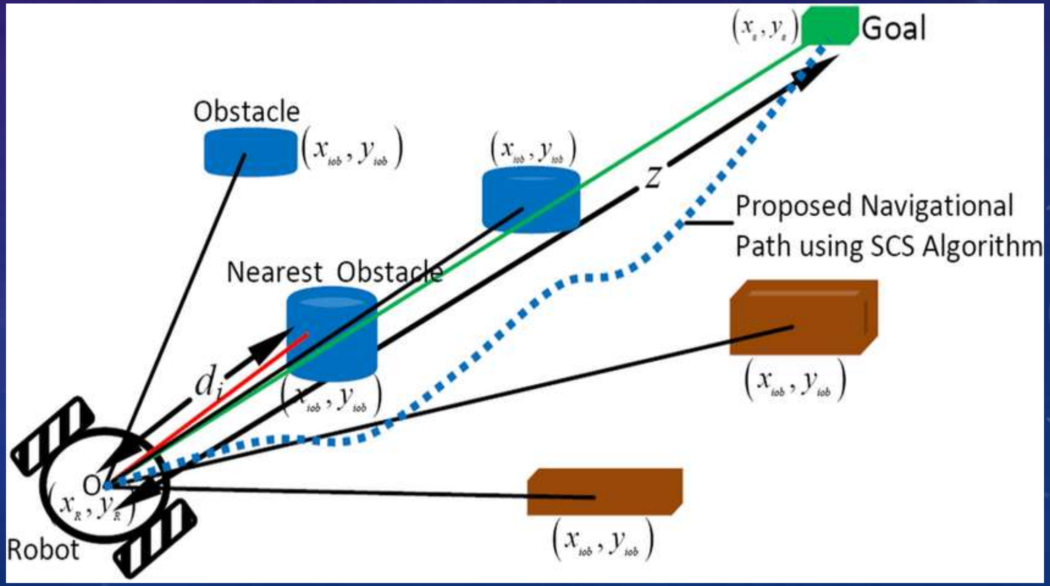
TSE YAN CHAU

TUMBAPO DINA HANG

TABLE OF CONTENTS

WHAT IS PATH PLANNING?


universalweather.com/blog


Goal
Obstacle
Nearest Obstacle
Proposed Navigational
Path using SCS Algorithm
Robot

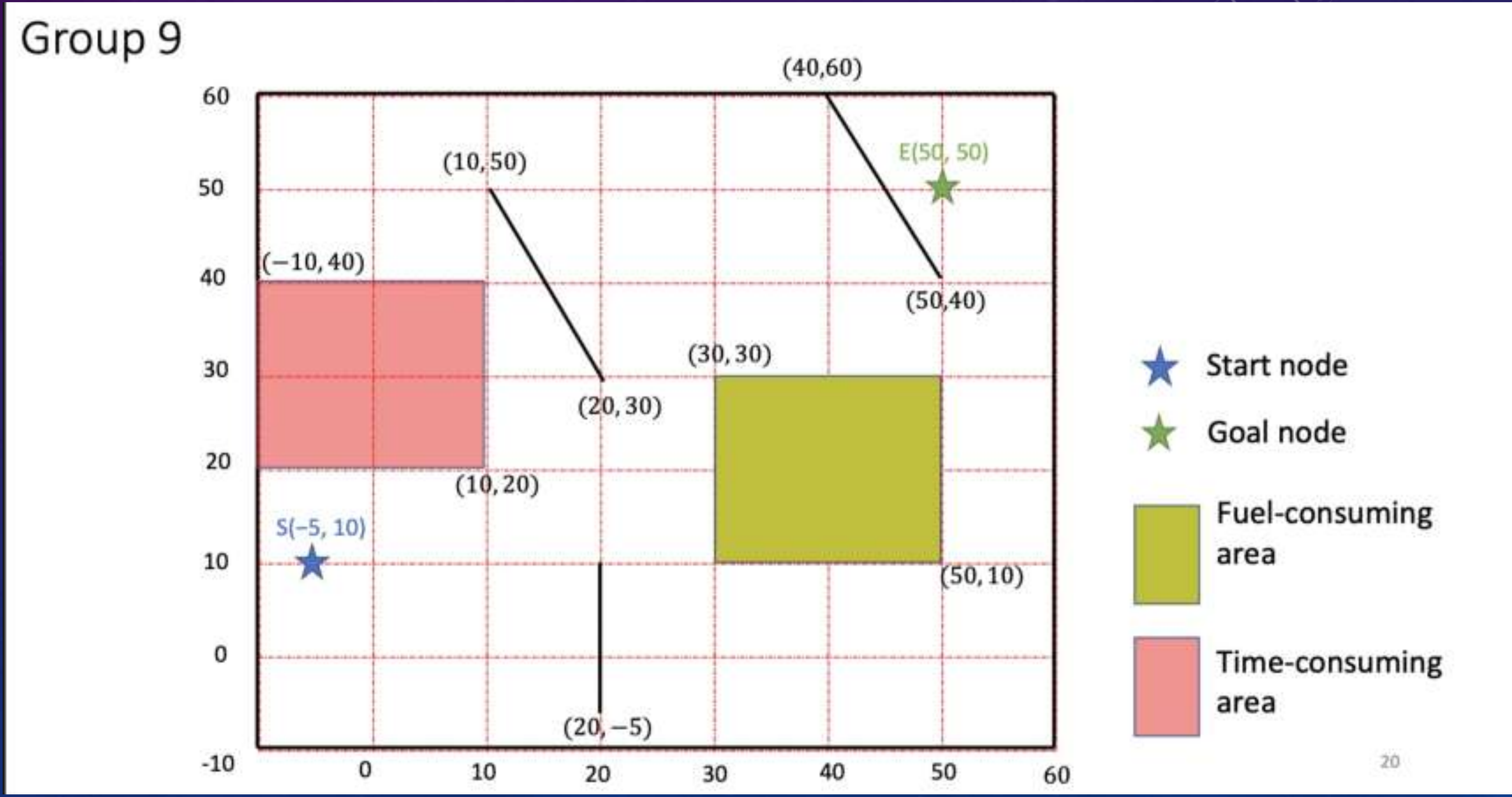
IMPORTANCE TO AVIATION ENGINEERING

OUR GOAL !

- Select the best aircraft models with minimum costs
- Design safe and cheapest path considering different scenarios

TASK 1

Finding an appropriate aircraft model that achieve minimum cost for **3** scenario for the challenge



[CODING 1]

```
self.Delta_C1 = 0.2 # cost intensive area 1 modifier time
self.Delta_C2 = 0.4 # cost intensive area 2 modifier fuel
```

→ Accountable for the percentage charge of going through these two cost intensive areas, cost and time zones

```
# reaching goal
if current.x == goal_node.x and current.y == goal_node.y:
    print("Total Trip time required -> ",current.cost )
    goal_node.parent_index = current.parent_index
    goal_node.cost = current.cost
    T = current.cost
    #A321neo
    F321 = 54 #Fuel Consumption
    P321 = 200 #Passenger capacity
    TL321 = 10 #Cost time Low
    TM321 = 15 #Cost time Medium
    TH321 = 20 #Cost time High
    CA321 = 1800 #Fixed Cost
```

→ Add constants

written each constant for each aircraft separately

```
N320 = math.ceil(TP / P321) #number of flights capable, maximum 12
Total_cost_A321 = (T * F321 * CF + T * TM321 + CA321) * N320
if N320<=12:
    print("Total Cost of A321neo -> ",Total_cost_A321)
else:
    print("A321neo is not viable")
```

→ Calculates the total number of flights necessary for A321neo to carry all 3000 passengers in a week

```python
# start and goal position
sx = -5.0  # [m]
sy = 10.0  # [m]
gx = 50.0  # [m]
gy = 50.0  # [m]
grid_size = 1  # [m]
robot_radius = 1.0  # [m]
```

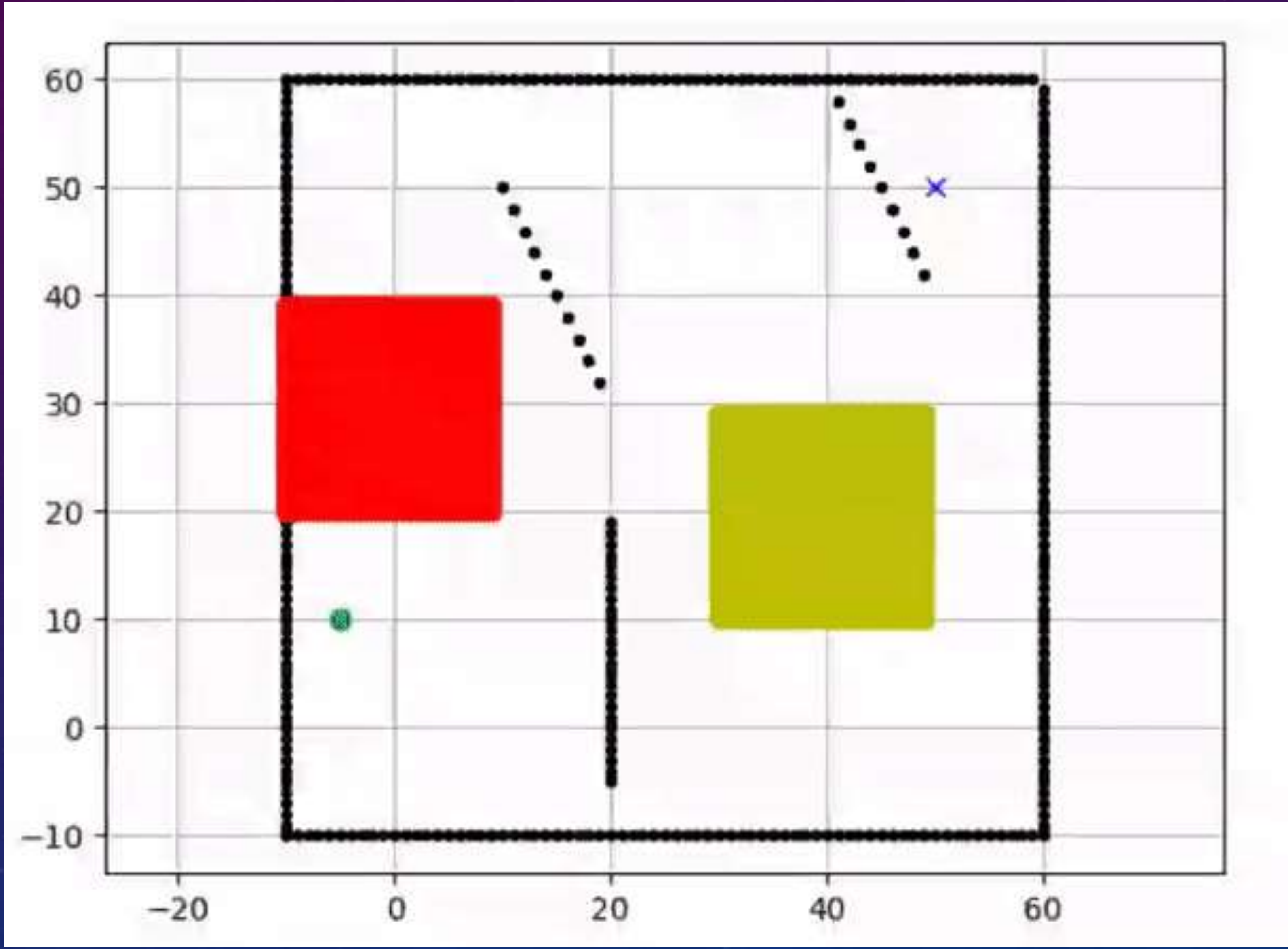
```python
for i in range(-5, 20): # draw the free border
    ox.append(20.0)
    oy.append(i)
for i in range(40, 50):
    ox.append(i)
    oy.append(-2 * i + 140)
for i in range(10, 20):
    ox.append(i)
    oy.append(-2 * i + 70)
```

→ Represents the obstacles that do not let aircraft to fly through them.

```python
# set cost intesive area 1
tc_x, tc_y = [], []
for i in range(-10, 10):
    for j in range(20, 40):
        tc_x.append(i)
        tc_y.append(j)

# set cost intesive area 2
fc_x, fc_y = [], []
for i in range(30, 50):
    for j in range(10, 30):
        fc_x.append(i)
        fc_y.append(j)
```

→ Setting of intensive areas



60
50
40
30
20
10
0
−10
−20
0
20
40
60

$$C = C_F \cdot \Delta F \cdot T_{best} + C_T \cdot T_{best} + C_c$$

Scenario 1:

- 3000 passengers (week)
- Max. of 12 flights
- Time cost = MEDIUM
- Fuel cost = $0.76/kg

Scenario 2:

- 1250 passengers (month)
- Max. of 5 flights
- Time cost = HIGH
- Fuel cost = $0.88/kg

Scenario 3:

- 2500 passengers (week)
- Max. of 25 flights
- Time cost = LOW
- Fuel cost = $0.95/kg

→ Total Trip time required -> 77.7mins

A330neo -> $85894.42/trip

A350-900 -> $89186.64/trip

A330neo -> $49191.80/trip

A350-900 -> $45168.54/trip

A330neo -> $84267.23/trip

A350-900 -> $85552.68/trip

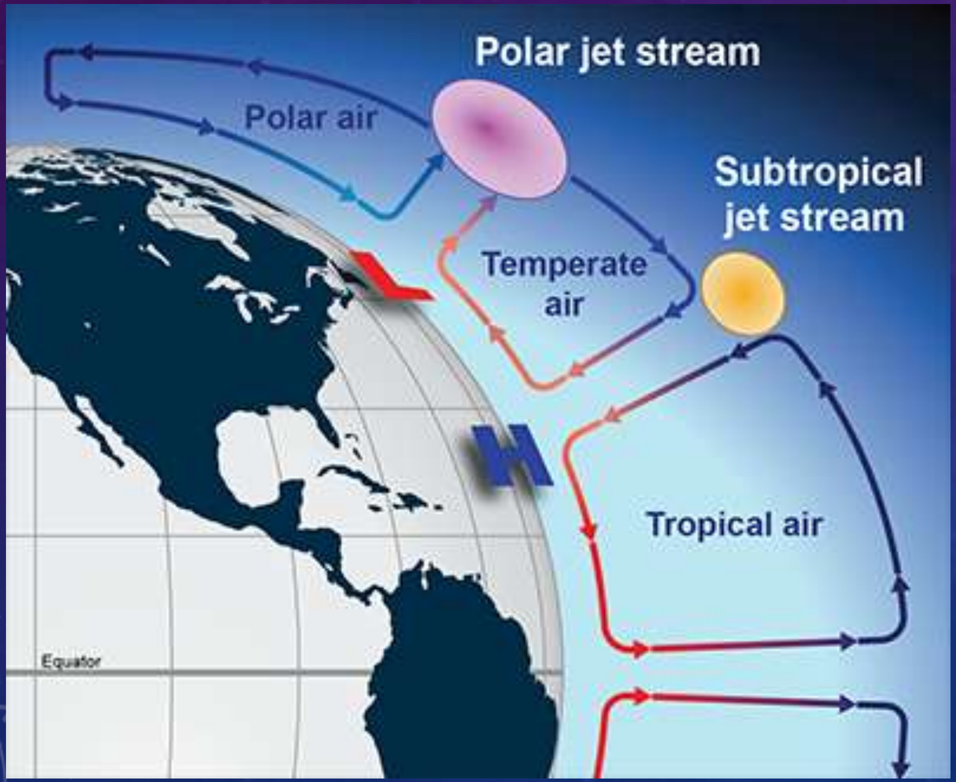
→ Result

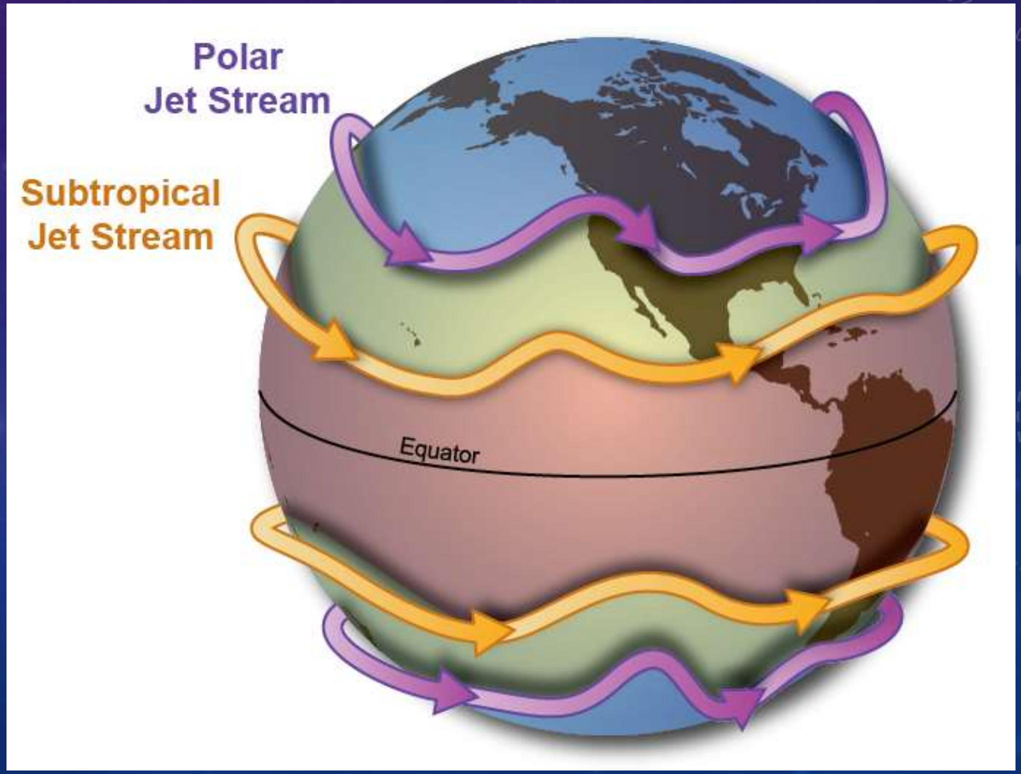
TASK 2

Design a new <u>minus cost area</u> that can reduce the cost of the route.

II

Jet Stream





[CODING 2]

Create a new area

```
# set jet stream area
for x in range(10, 49):   # 40 values
    js_x, js_y = [], []
    for i in range(-10, 60):
        for j in range(x, x+5):
            js_x.append(i)
            js_y.append(j)
    a_star = AStarPlanner(ox, oy, grid_size, robot_radius, fc_x, fc_y, tc_x, tc_y, js_x, js_y)
    rx, ry, costpath = a_star.planning(sx, sy, gx, gy, x)

    if x==10:
        lowest = costpath
        bestx = x

    if costpath < lowest:
        lowest = costpath
        bestx = x
x=3
```

```python
#set jet stream best x
js_x, js_y = [], []
for i in range(-10, 60):
    for j in range(bestx, bestx+5):
        js_x.append(i)
        js_y.append(j)
```

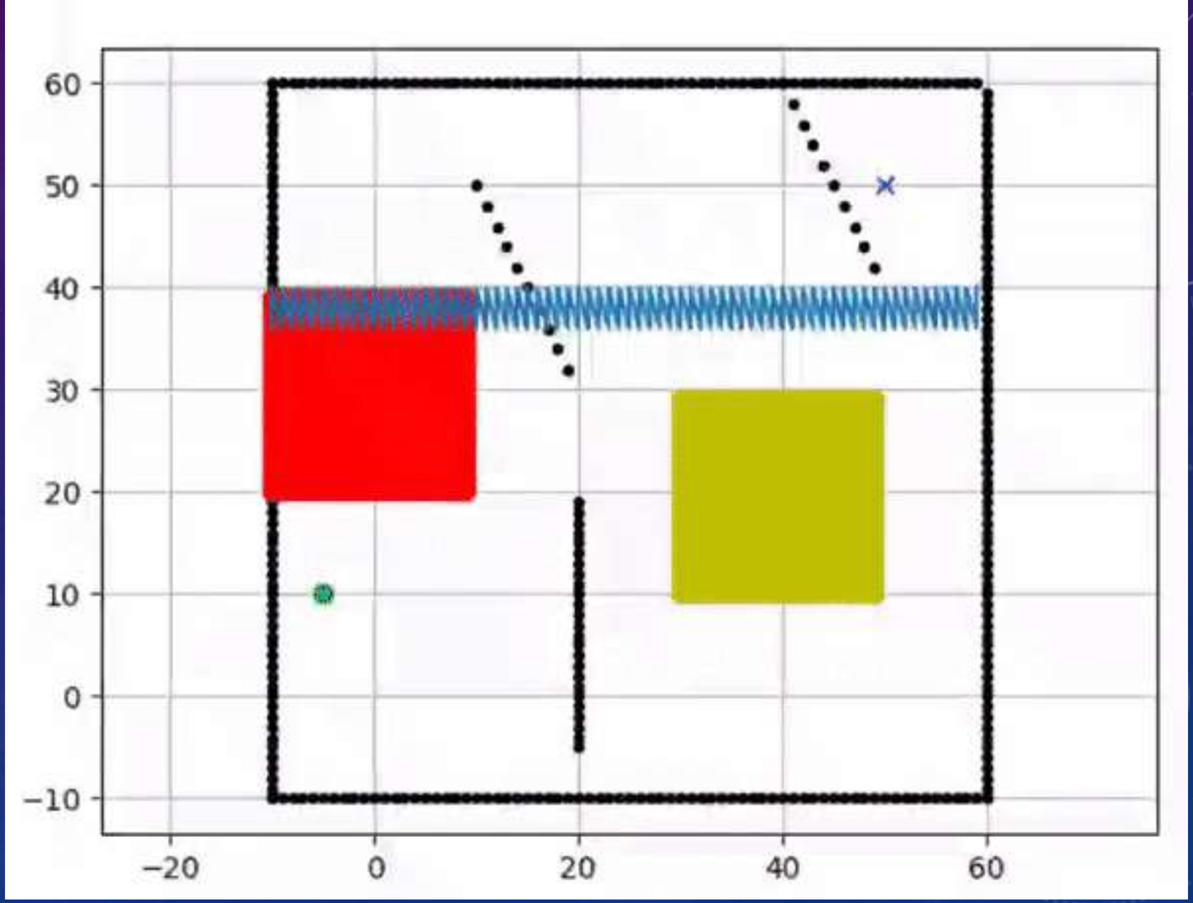
→ Set a range, choose the best area for jeet stream base on the best x value

```python
a_star = AStarPlanner(ox, oy, grid_size, robot_radius, fc_x, fc_y, tc_x, tc_y, js_x, js_y)
rx, ry, costpath = a_star.planning(sx, sy, gx, gy, x)
```

→ In order to make the programme work

Result:

- Using Scenario 1 in Task 1 as the background
- Cost along the jet stream is reduced by 5%



TASK 3

Design a new Aircraft Model that achieve minimum cost for the challenge mentioned in Task 1

$$\boxed{C = C_F \cdot \Delta F \cdot T_{best} + C_T \cdot T_{best} + C_c}$$

C_F=cost of fuel per kg

C_T=time related cost per minute of flight

C_c=fixed cost independent of time

ΔF=trip fuel

ΔT=trip time

Potential design of aircraft: B797

- Twin win RR Trent 1000 engine aircraft
- The targeted no of flights is 11 with 273 passengers per flight

```
TP = 3000 #Total passengers per week
MF = 12 # Maximum number of flights per week
CT = 2000 # Fixed cost for twin-engine
CF = 2500 # Fixed cost for 4-engine
Ctime = 12 # Cost for time
F = 20 # Kg per minute
usdbbl = 105.68 # usd/bbl for Asia n Oceania Region
```

- List of constants

$$C = C_F \cdot \Delta F \cdot T_{best} + C_T \cdot T_{best} + C_c$$

C_F : $ 105.68/bbl

ΔF : 2*20 = 40 kg/min = 0.251592 bbl/min

T_{best} : 77.7mins

C_T : 273 passengers/ 50*$2/min+12(base)

=22.92 =$23/min

C_c : $2000

RESULT:

Twin-engine B797 -> 69735.13

Passenger capacity B797 -> 273

Max. cost = $69375

[CODING 3]

```python
for add_pass in range(0, 200):
    passengers = 250 + add_pass # number of passengers
    Cfuel = usdbbl/ 119.24047 # cost for fuel
    m = math.ceil(add_pass/50) # by what value it is increased
    N = math.ceil(TP/passengers) # number of flights

    if passengers >= 300:
        Total_cost = (T * 4 * F * Cfuel + T * (18 + 2 * m ) + CF)*N
    else:
        Total_cost = (T * 2 * F * Cfuel + T * (18 + 2 * m ) + CT)*N


    if passengers==250:
        lowest = Total_cost
        bestpassengers = passengers
    if Total_cost < lowest:
        lowest = Total_cost
        bestpassengers = passengers
```

→

MOST ENSSENTIAL PART !

- Write a formula for converting from USD per barrel to cost for fuel in terms of kilograms
- Formulas of "m" and "N" are responsible for adding additional costs for every 50 passengers and revealing the total number of flights

```python
Nbest = math.ceil(TP/bestpassengers) # number of flights with best number of passengers
if bestpassengers >= 300:
    print("Total cost of 4-engine B797-> ",lowest)
    print("Passenger capacity B797-> ",bestpassengers)
else:
    print("Total cost of twin-engine B797 -> ",lowest)
    print("Passenger capacity B797->",bestpassengers)
break
```

←

The coding will give either one of the solution which probably describes all information that is needed.


THANKS
FOR YOUR ATTENTION
memegen.es