Contents

SpaceRyde

Khang Nguyen

October 2022

1 Problem Summarize

This section will summarize the problem that was given to us. This will help us understand the problem more clearly.

1.1 Air Traffic Control

- Track the plane position
- Control Zone radius 10 km
- Queues landing on arrival (based on time)
- Tell the plane to land when it's their time ***else*** fly in a circle with a radius of 1 km around a point.
- Plane must not come within 100m of each other.

1.2 Plane

- Assume an airplane is a point mass
- Spawn Randomly
- Plane Travel at 140 m/s
- Stop and disappear at the end of the landing strip.
- Plane transmits the position at 10Hz.
- Speed is still 140m/s at the end of the runway

1.3 Extra Assumption

1.3.1 Assumption for plane

- The airplane can land in both directions on the runway
- The communication between the plane and ATC is instantaneous. I am assuming that we will use radio communication which travels at the speed of light.

1.3.2 Assumption for ATC

- The ATC will know if an airplane enters the control zone immediately

2 Problem Set Up.

In this section, we analyze all the constrain that were given. Since the problem is very complicated, we also simplify by making some restrictions and assumptions. NOTE: This may not be the most efficient way to solve this problem, but it would be functional.

2.1 Worst Case

To set up the problem, we must think of the worst-case scenario. The worst case is when the maximum amount of planes enter the control area at the same time.

Since every plane is 100 m apart, we can **estimate** the maximum number of planes that can enter the area at a time is the circumference divided by the arc with a 100 m chord. We can use the following equation to calculate the maximum number of planes entering the control area.

$$N_{max} = \frac{2\pi r}{2r \cdot sin^{-1}(d/2r)}$$

I propose adding two more constraints to the problem. The area where all airplanes can circle while waiting for landing is the green area shown in the diagram below.

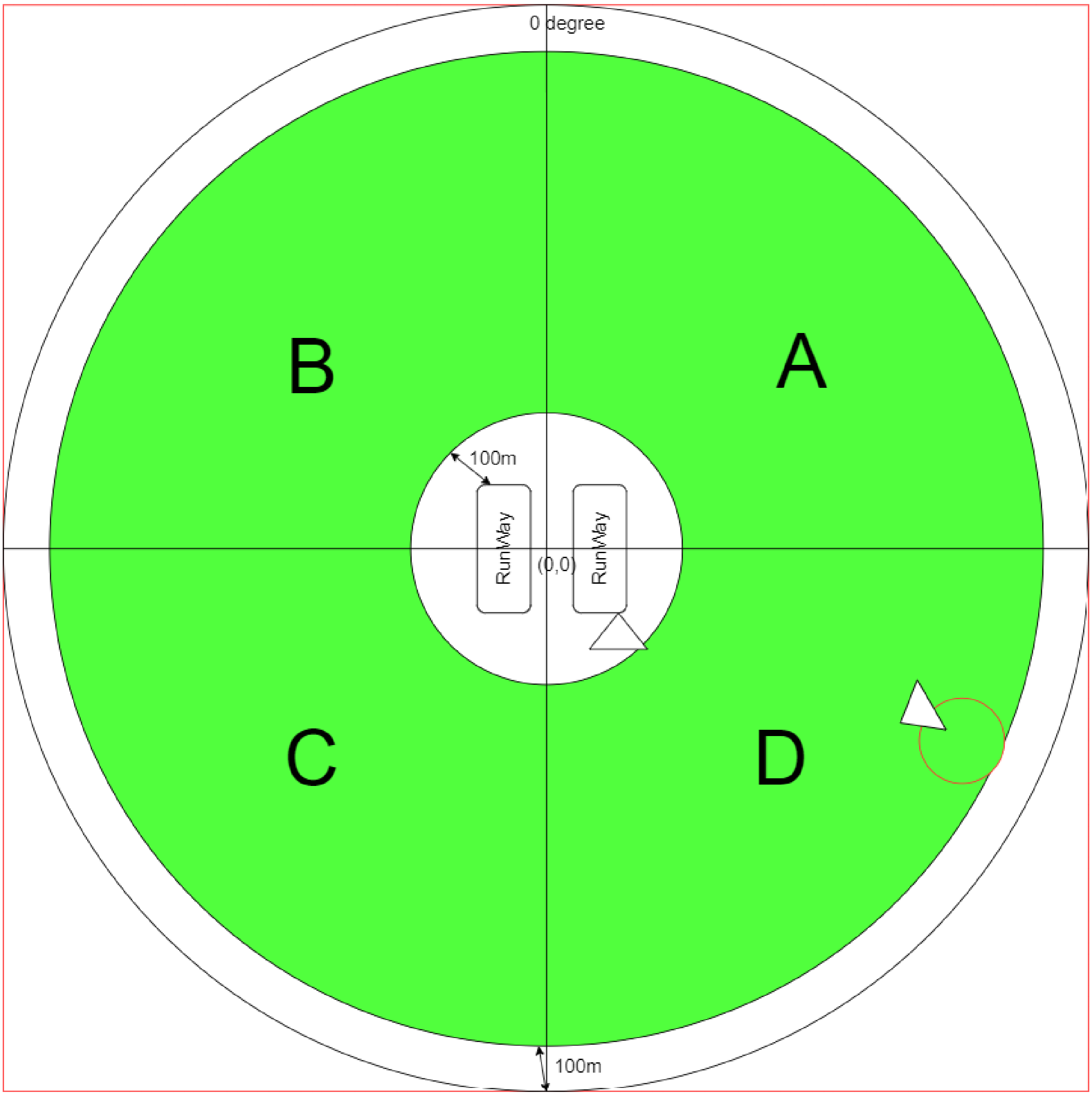

Figure 1-1: The green area from this diagram is the area where we allow the plane to make a circle while waiting for landing

We create these two areas for safety reasons. We do not want an airplane to enter the circle and be within 100m of another airplane. By doing this, planes that enter the area will be at least 100 m from the closest plane. The inner circle is the safety area for landing, this makes sure that when a plane approaches landing the other plane that waiting is not within 100 m. We will call these regions outer and inner safety zone.

We also divided the circle into four different regions as shown in the diagram labelled A,B, C, D. We divide this into 4 regions because the airplane can land from four positions. The plane will land on one of the ends of one of the two runways.

We would use both polar coordinates and Cartesian coordinates to solve this problem. The Cartesian Coordinates is used to determine the longitude and latitude of the plane. From this, it's easy to convert polar coordinates It is also easy to determine which region the plane is in. We can convert between the two coordinate systems using the following equations.

Cartesian Coordinate to Polar Coordinate

$$x = r \cdot cos(\theta)$$

$$y = r \cdot sin(\theta)$$

Polar Coordinate to Cartesian Coordinate

$$r = \sqrt{x^2 + y^2}$$

$$\theta = tan^{-1}(\frac{x}{y})$$

2.2 Waiting Plane

We know that the plane will make a 1 km radius circle in the green area while waiting for landing. We could have up to 62 airplanes circling at the same time without being within 100m of each other. Therefore, the distance between the center of one circle to the other is 1100m.

We will set up the location of the waiting circle to be another circle with a radius starting at the center of the control area. We will call these the set-up circle. The outermost of it will have a radius of the outer safe zone radius ($R_{outersafe}$) subtracted from the radius of the waiting circle (1 km)

$$R_{outersafe} = R_{outersafe} - 1000$$

The most inter radius of these set-up circles with being radius us the inner safety zone ($R_{Inersafe}$) plus the radius of the waiting circle (1 km).

$$R_{outersafe} = R_{Inersafe} - 1000$$

We can calculate the number of the set-up circle by the following equation

$$N_{SetUpCircle} = (\frac{R_{outersafe} - R_{outersafe}}{1100}) + 1$$

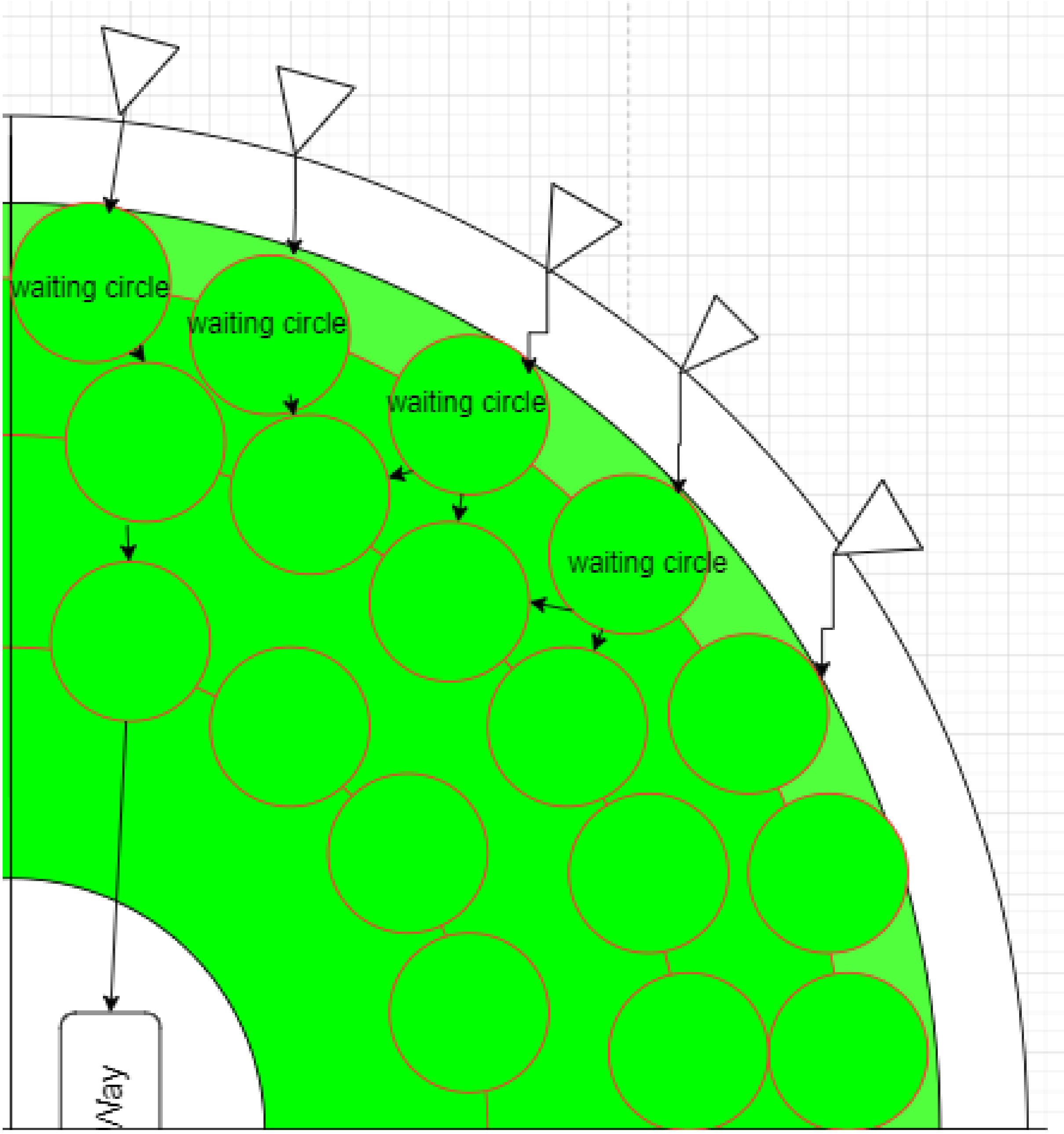

Figure 1-2: Waiting circle is set up to be on another circle with a radius starting at the center of the runway

Due to the factor of safety, we will make each circle to be 1100 m apart from the centre of each circle. This ensures that while circling the plane would not be within 100 m of each other. We can develop the following pattern for a plane to switch circles when it's full.

2.2.1 Iner Safety Zone

We need to set up the inner circle radius. We can calculate the radius of the inner safety zone.

$$R_{InerSafetyZone} = \sqrt{(250+100)^2 + 250^2} + 100$$

2.3 Time

The Air Plane travel relative to time. Time is the fundamental dimension for these physical questions. We will need a continuous time system to calculate the time. I will attach a code that I wrote before about timekeeping if we need a high-precision application. But for now, a continuous time starting when we run the code is sufficient.

We know that each plane can transmit its position at the rate 1Hz. This means that the plane will send its location every second to the ATC. Since each airplane travels 140m/s, the plane will be 140 m off from the location it was before

2.4 Communication

We will need to use an interrupt signal to communicate with each plane. Since this is a time-sensitive mission, we would need the fastest way to communicate with the plane.

3 Solve Problem

With the setup above, we can now use it to design our system. There are a few things we would need the code to do. In this problem, we will use a priority queue. The value that we prioritize is time.

Here is the list of things that our software needs to do.

1. Check to see if any place planning to land. If there are, add the plane that enters the nearest waiting circle.
2. If there is one plane, send it to the closes runway. A zone and B zone will have priority. C and D land will have less priority.
3. If the airplane can not land, append it to the nearest number waiting for the circle.
4. When landing, remove the plane from the priority queue.
5. Move the airplane from the outer circle into the inner circle

Due to the complexity of the problem, we will deal with each zone separately.

We will also model this as follow, we can also solve it like a binary tree with each plane is a bit of data.