COMPUTER ARCHITECTURE LAB

AS-0 REPORT
DHEERAJ REDDY
220010009
CSE,IIT DHARWAD

January 14, 2024

Contents

1 AIM

Consider the scenario where one country, called the defending country (DC), wishes to defend its border against another country, called the attacking country (AC), whose aim is to send an infiltrator to cross the border and enter DC's land. DC decides to deploy a wireless sensor network along the border. If a sensor detects an infiltration attempt, DC can then send its troops to counter the infiltration. Quite obviously, the infiltrator would like to enter DC's land without triggering any sensors.

2 COMPONENTS

1. BORDER
2. SENSORS
3. INFILTRATOR
4. CLOCK
5. PLOTTING FILE

3 WORKING OF EACH COMPONENT

3.1 SENSOR

1.The Class Sensor stores two variables probability and its state that is on or off.

2.It has a change method that is for every 10 seconds according the probability given its modeled as a biased coin toss and if there is a 1 then it implies the sensor is on and if there is a 0 that implies the sensor is off.

3.2 BORDER

1.The Class BORDER contains two variables width of the border and the probability that will be given to the sensors and a 2d array of sensors of width and length 1000.

2.The border class only have one method which takes the input as x and y co-ordinates and outputs the state of the sensor at that location.

3.3 INFILTRATOR

1.The Class infiltrator contains a variable position which stores the x,y co-ordinates of the position of the infiltrator and width of the border.

2.The infiltrator has a method move which checks the state of the sensor he

is present and the succesive sensors and if both the sensors are off then the position of the infiltrator gets updated according to it.

3.4 CLOCK

1.The Class Clock contains a variable time and basically works like a stopwatch in our simulation.

2.It has two methods increase which increments the time and get time which outputs the time.

3.5 PLOTTING

1.The plotting file generates 100 random values of probability between 0.1 and 0.9 and also 100 random values of width between 5 to 50 and then fills the output.txt file with this data.

2.Then it will make the our main method work by feeding each input values and recording all the output values.

3.Then it plots the graph of width vs crossing time vs probability for all these values.

4 MAIN METHOD

THE MAIN METHOD IS PRESENT IN THE PROJECT.JAVA FILE AND IT WORKS AS FOLLOWS:

1.It Initialises all the variables and creates the border and initialises all the sensors in the border and then keeps the initial time as zero.

2.It makes sure that for every 10 seconds the infiltrator moves and the sensors change their states.

3.The method now collects each different set of values of width and probability and then give the time by which the infiltrator crosses the border.

5 OBSERVATIONS

1.MAINLY THE TIME TAKEN TO CROSS IS MORE DEPENDENT ON PROBABILITY THAN WIDTH.
2.IF THE PROBABILITY IS NEAR TO ZERO THEN EVEN IF THE WIDTH IS SMALL THE TIME TAKEN IS LARGE.
3.WHEN THE WIDTH IS CONSTANT THE GRAPH RESEMBLES THE GRAPH OF EXPONENT.
4.WHEN THE PROBABILITY IS CONSTANT THE GIVE LOOKS LIKE A INCREAING GRAPH WITH WIDTH.

5.1 OUTPUT FILE

```
85 0.597114 1720
44 0.641378 830
58 0.779404 800
47 0.338851 1980
21 0.284851 1520
67 0.435899 2060
32 0.828101 360
94 0.881390 1100
15 0.374080 590
84 0.691058 1420
83 0.825355 1020
33 0.323848 1740
82 0.785500 1070
59 0.590372 1000
38 0.277330 2070
88 0.528892 2130
93 0.645004 1560
91 0.755151 1190
7 0.401606 200
16 0.416779 430
15 0.210868 1710
90 0.689120 1590
71 0.737548 1090
89 0.245502 6660
72 0.528354 1690
```

Figure 1: OUTPUT FILE

5.2 GRAPHS

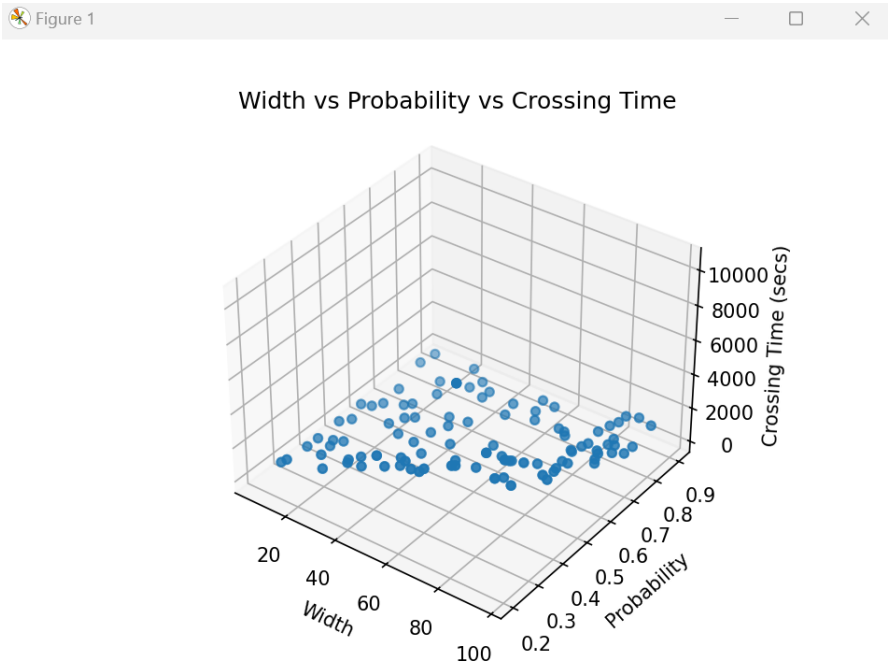

Figure 2: WHEN WIDTH AND PROBABILITY ARE VARYING

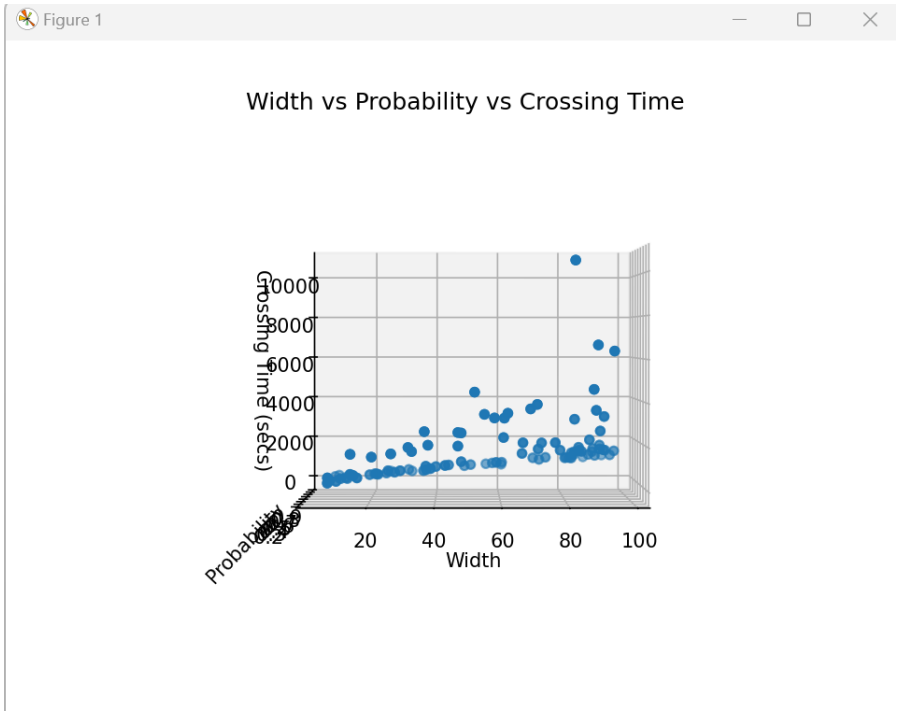

Figure 3: WHEN ONLY WIDTH IS VARYING

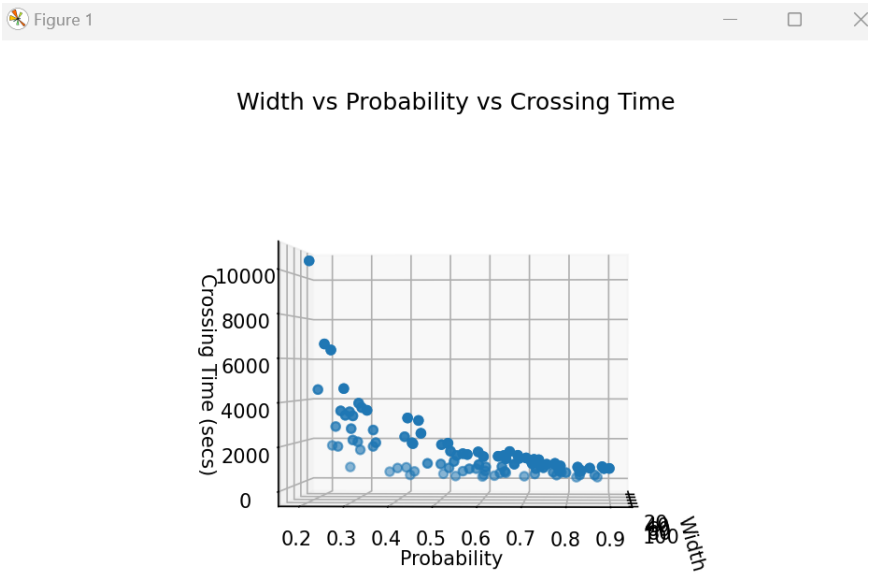

Figure 4: WHEN ONLY PROBABILITY IS VARYING