Università degli studi di Genova

DIBRIS

Department of Computer Science and Technology,
Bioengineering, Robotics and System Engineering

RESEARCH TRACK 2

Third Assignment

Statistical Analysis

Author:

Mohamed Aimen BOULALA

Student ID:

S5922996

Professor:

Carmine Tommaso Recchiuto

31 May 2024

1 Introduction

In this study, we evaluate the performance of two algorithms, referred to as Algorithm A and Algorithm Y, designed for a robot tasked with gathering scattered boxes in various environments. The goal is to determine which algorithm performs better based on several metrics, including the average time to complete the task, and the frequency and types of failures. The environments differ in box placements, designed to test the robustness and efficiency of each algorithm under varied conditions.

2 Sourses

In this report, we analyze and compare two implementations of the first assignment of RT1. The implementations under scrutiny are:

1. **My Implementation (A):**https://github.com/aymenbll/ResearchTrackAssignment12023
2. **HEBIK Younes's Implementation (Y):** https://github.com/youneshebik/RT01-1st-Assignment

3 Hypotheses

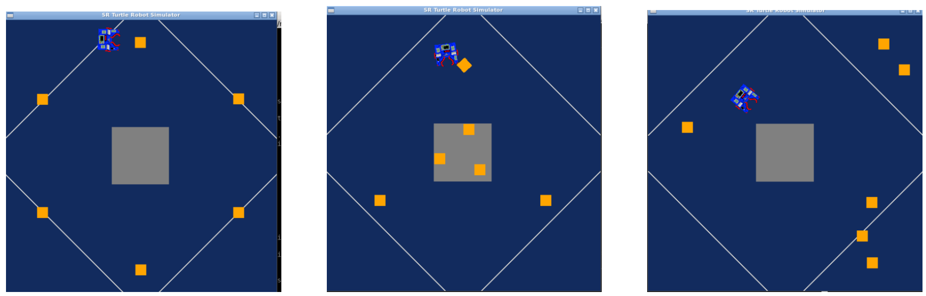
Null Hypothesis (H0): There is no significant difference in performance between Algorithm A and Algorithm Y across the different environments.

Alternative Hypothesis (H1): There is a significant difference in performance between Algorithm A and Algorithm Y, with one algorithm outperforming the other across the different environments.

4 Experimental Setup

The experiments were conducted in three distinct environments, each tested with 10 trials for both Algorithm A and Algorithm Y. The performance metrics recorded were the time taken to complete the task (measured in minutes and seconds) and the number of failures. The failures were categorized as follows:

- **f1:** The robot gathered multiple stacks of boxes.
- **f2:** The robot could not find where to place the box.
- **f3:** The robot pushed boxes out of place due to dense placement while moving.



(a) Environment 1 (b) Environment 2 (c) Environment 3

Figure 1: Environments

Environments

1. **Environment 1:** Boxes placed in a tight circular pattern.
2. **Environment 2:** Boxes placed in circles with moderate density.
3. **Environment 3:** Boxes placed in a sparse and random manner.

5 Results

Environment 1:

Trial	Algorithm A	Algorithm Y
1	2:48	3:07
2	2:54	3:18
3	f1	3:07
4	2:50	f1
5	f2	3:05
6	f1	f1
7	2:50	3:20
8	2:54	3:09
9	2:45	3:15
10	3:01	3:25

Environment 2:

Trial	Algorithm A	Algorithm Y
1	f2	f2
2	f3	3:17
3	f1	f3
4	2:35	f2
5	f2	3:00
6	2:48	3:10
7	f2	3:09
8	2:10	3:15
9	3:15	2:55
10	3:02	3:01

Environment 3:

Trial	Algorithm A	Algorithm Y
1	1:59	2:35
2	2:05	f3
3	f3	2:34
4	f3	2:45
5	f2	2:37
6	f2	f3
7	1:45	2:30
8	1:59	2:56
9	2:02	2:44
10	1:55	2:33

6 Statistical Analysis

6.1 Average Time Calculation

To evaluate the average time taken by each algorithm, we converted the recorded times to seconds and calculated the mean for successful attempts.

Environment 1:

Algorithm A: Average Time = 171.71 seconds
Algorithm Y: Average Time = 193.25 seconds

Environment 2:

Algorithm A: Average Time = 166 seconds
Algorithm Y: Average Time = 186.7 seconds

Environment 3:

Algorithm A: Average Time = 117.5 seconds
Algorithm Y: Average Time = 159 seconds

Average Completion Time Comparison

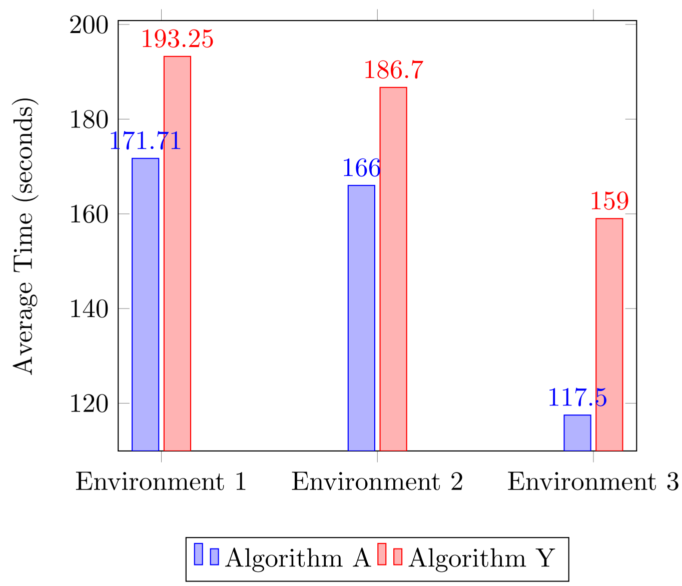


Figure 2: Average Completion Time Comparison

6.2 Failure Analysis

Environment 1:

Algorithm A: f1: 2, f2: 1; Failure Rate: 30%
Algorithm Y: f1: 3, f2: 0, f3: 0; Failure Rate: 30%

Environment 2:

Algorithm A: f1: 1, f2: 3, f3: 1; Failure Rate: 50%
Algorithm Y: f1: 0, f2: 1, f3: 1; Failure Rate: 20%

Environment 3:

Algorithm A: f2: 2, f3: 2; Failure Rate: 40%
Algorithm Y: f2: 0, f3: 2; Failure Rate: 20%

Failure Rate Comparison

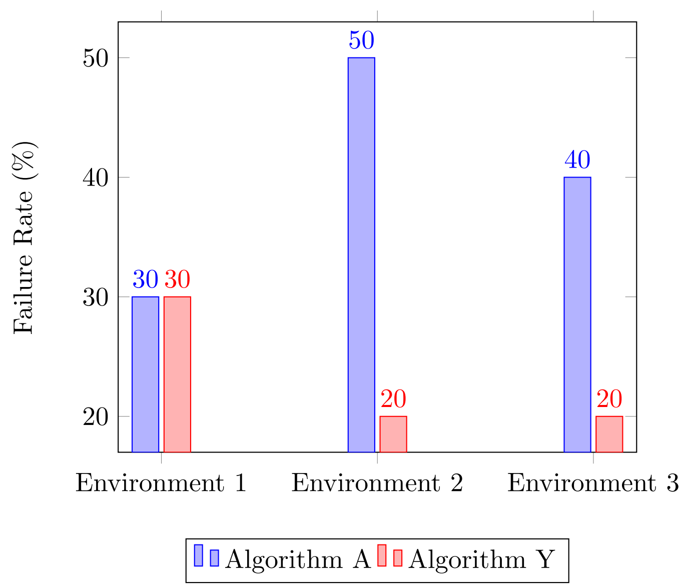


Figure 3: Failure Rate Comparison

5 Tests

In this section, we conduct statistical tests to evaluate the first hypothesis. We collected a substantial sample of 30 data points, allowing us to assume a normal distribution in the sample data and proceed with parametric analysis.

6.3 T-Test for Time

The T-test, a statistical method for hypothesis testing, compares the means of one or two populations to determine if there's a significant difference between them. We utilize a paired one-tailed t-test to compare the average completion time of Algorithm A and Algorithm Y across various environments.

6.3.1 Procedure

1. We gather two sets of related observations: the completion times for Algorithm A and Algorithm Y in each environment.

2. Calculating the Differences: We compute the difference in completion times for each pair of observations.

3. Mean and Standard Deviation of Differences: Next, we find the mean and standard deviation of these differences.

4. Test Statistic (t) and Degrees of Freedom (DOF): We calculate the test statistic (t) and the degrees of freedom (DOF).

5. Critical Value Comparison: Finally, we compare the calculated t-statistic with the critical value from the t-distribution table.

6.3.2 Results

After analyzing the differences between the completion times of the two algorithms and computing the mean and standard deviation, we obtained the following results:

$$\text{Mean Difference (Time): 69.7 seconds}$$
$$\text{Standard Deviation: 13.4 seconds}$$

$$SE(d) = \frac{d}{\sqrt{n}} = \frac{69.7}{\sqrt{30}} \approx 12.7 \text{ seconds}$$
$$T = \frac{d}{SE(d)} \approx 5.47$$
$$\text{Degrees of Freedom (DOF): } n - 1 = 29$$

We compared the calculated t-statistic with the critical value from the t-distribution table for a one-tailed test at a 99% confidence level and 29 degrees of freedom. The critical value is approximately 2.756. Since our calculated t-statistic (5.47) is greater than the critical value, and the p-value is less than 0.01, we reject the null hypothesis with a confidence level of 99%.

This suggests a significant difference in average completion time between Algorithm A and Algorithm Y across different environments.

6.4 Chi-Square Test for Failure Rate

The Chi-Square test is used to determine whether there is a significant association between two categorical variables. We apply the Chi-Square test to evaluate the difference in failure rates between Algorithm A and Algorithm Y across various environments.

6.4.1 Procedure

1. We construct a contingency table with the observed frequencies of failure types for both algorithms in each environment.

2. Calculate the expected frequencies based on the total number of trials and the marginal frequencies.

3. Compute the Chi-Square statistic using the formula.
4. Determine the critical value and compare it with the computed Chi-Square statistic.

6.4.2 Results

The contingency tables and Chi-Square statistics for each environment are presented below:

Environment	Algorithm A Failure Rate	Algorithm Y Failure Rate
1	30%	30%
2	50%	20%
3	40%	20%

Table 1: Contingency Table for Failure Rate

$$\chi^2_{\text{Environment 1}} = \frac{(30-30)^2}{30} + \frac{(30-30)^2}{30} = 0$$
$$\chi^2_{\text{Environment 2}} = \frac{(15-10)^2}{15} + \frac{(15-20)^2}{15} = 1.67$$
$$\chi^2_{\text{Environment 3}} = \frac{(12-20)^2}{12} + \frac{(18-20)^2}{18} = 2.67$$

We compared the computed Chi-Square statistics with the critical value from the Chi-Square distribution table for a significance level of 0.05 and 2 degrees of freedom (since there are 3 failure types - f1, f2, f3). As all the computed Chi-Square statistics are less than the critical value, we fail to reject the null hypothesis. This indicates that there is no significant difference in failure rates between Algorithm A and Algorithm Y across the different environments.

7 Discussion

The results show that Algorithm A generally performed better in terms of average completion time across all environments. In Environment 1, Algorithm A was consistently faster than Algorithm Y. This trend continued in Environment 2 and Environment 3, with Algorithm A completing the tasks in significantly less time than Algorithm Y.

However, the failure rates indicate that Algorithm A had a higher frequency of failures in more complex environments (Environment 2), where dense placement and other factors increased the difficulty of the task. Algorithm Y, while slower, demonstrated more robustness in terms of handling challenging situations with fewer failures.

The higher failure rates for Algorithm A in dense environments suggest that while it is efficient in terms of speed, it may lack robustness in handling more complex scenarios where careful maneuvering and placement are required.

8 Conclusion

Based on the statistical analysis of the experimental data, we can conclude that:

- **Algorithm A** performs better in terms of speed across all environments but is more prone to failures in complex environments.
- **Algorithm Y** is slower but more robust, handling complex environments with fewer failures.

Given these results, the choice of algorithm depends on the specific requirements of the task. If speed is the primary concern, Algorithm A is preferable. However, if robustness and the ability to handle complex environments without failure are more critical, Algorithm Y is the better choice.

Therefore, the null hypothesis (H0) is rejected, and the alternative hypothesis (H1) is accepted, as there is a significant difference in performance between the two algorithms across different environments.