



Intelligent and Mobile Robotics

Assignment 2

*Robotic challenge solver using
the CiberRato simulation environment*

Diogo Pereira de Jesus, 97596
Pedro Miguel Tavares Rodrigues, 92338

Table of Contents



universidade de aveiro
theoria poesis praxis

1. Timeline
2. First Iteration and Results
3. Second Iteration and Results
4. Third Iteration and Results
5. To Improve
6. Conclusion

- **First iteration:** execute all defined tasks in a *noiseless environment*.
- **Second iteration:** add *motors noise* and correct the movement model **predicted position** based on both the compass sensor and line sensor.
- **Third iteration:** add *compass sensor noise* and filter the compass sensor readings.

Localization:

- Movement model only.

Mapping:

- Compute the **position** of **each element** in the line sensor.
- Find the **nearest cell** to **each element** of the line sensor.
- Compute **possible walls** that contain the **closest cell**.
- Check if the element (when at '1') **intersects any** of the walls.
- Define the cells that compose **each intersected wall** as neighbors.



Planning:

- Compute the **shortest distance** between **each checkpoint (A^*)**.
- Traveling salesman problem.
- Find **best path** by going through all **possible combinations**.



Edge cases:

- A element of the line sensor might intersect more than one wall when positioned very close to a **cell center** (point where walls unite).

In this case, just **ignore its detection** and **do not update** the perceived map.



Edge cases:

- If the robot is traversing a **diagonal wall**, the **closest cell** to a **element** of the line sensor might not be the cell the robot is **heading to**.

The **wall** that **would be added** to the perceived map **could be wrong**.

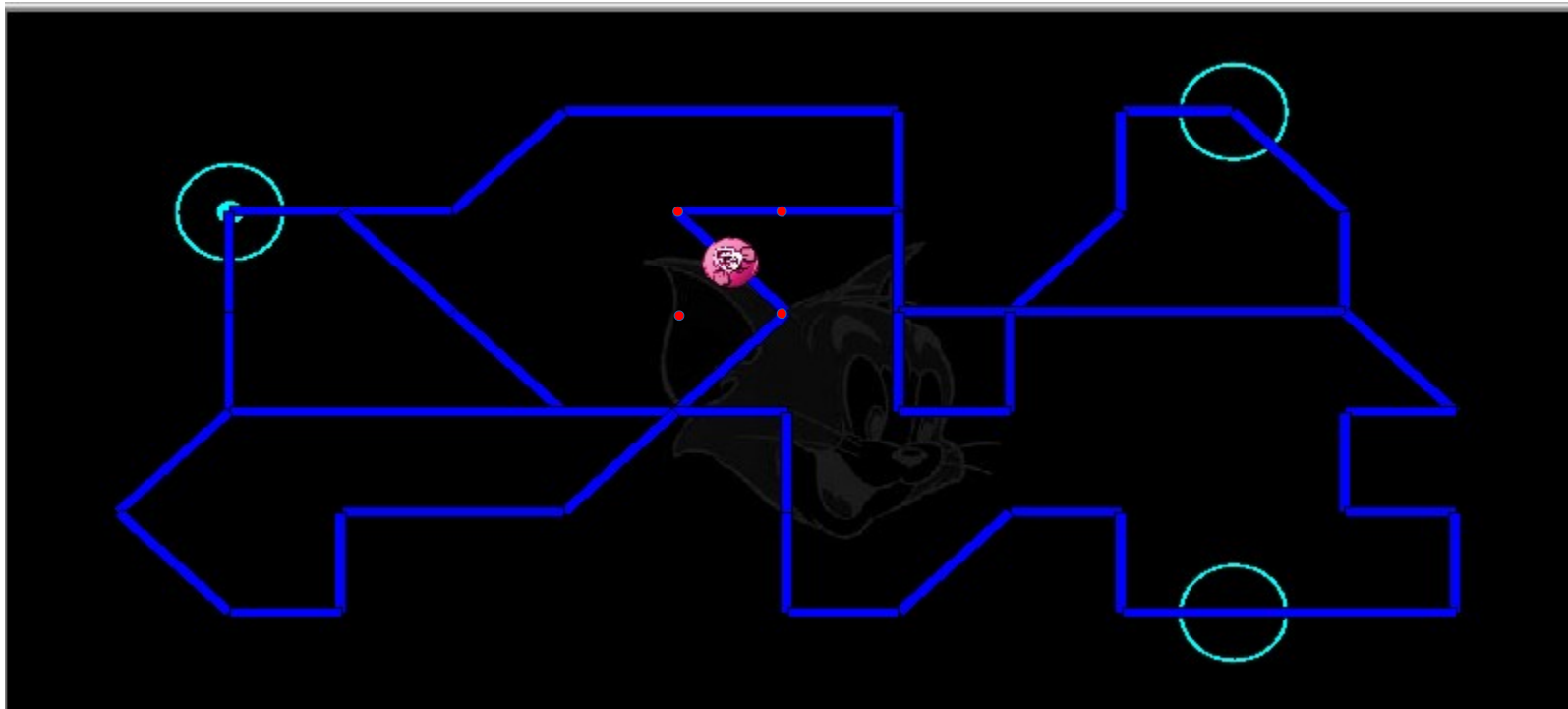
To solve this, check **both walls** that **intersect** that **exact position on the map**.

If one of the walls is **already present** on the perceived map, **do not** add the other.

First Iteration



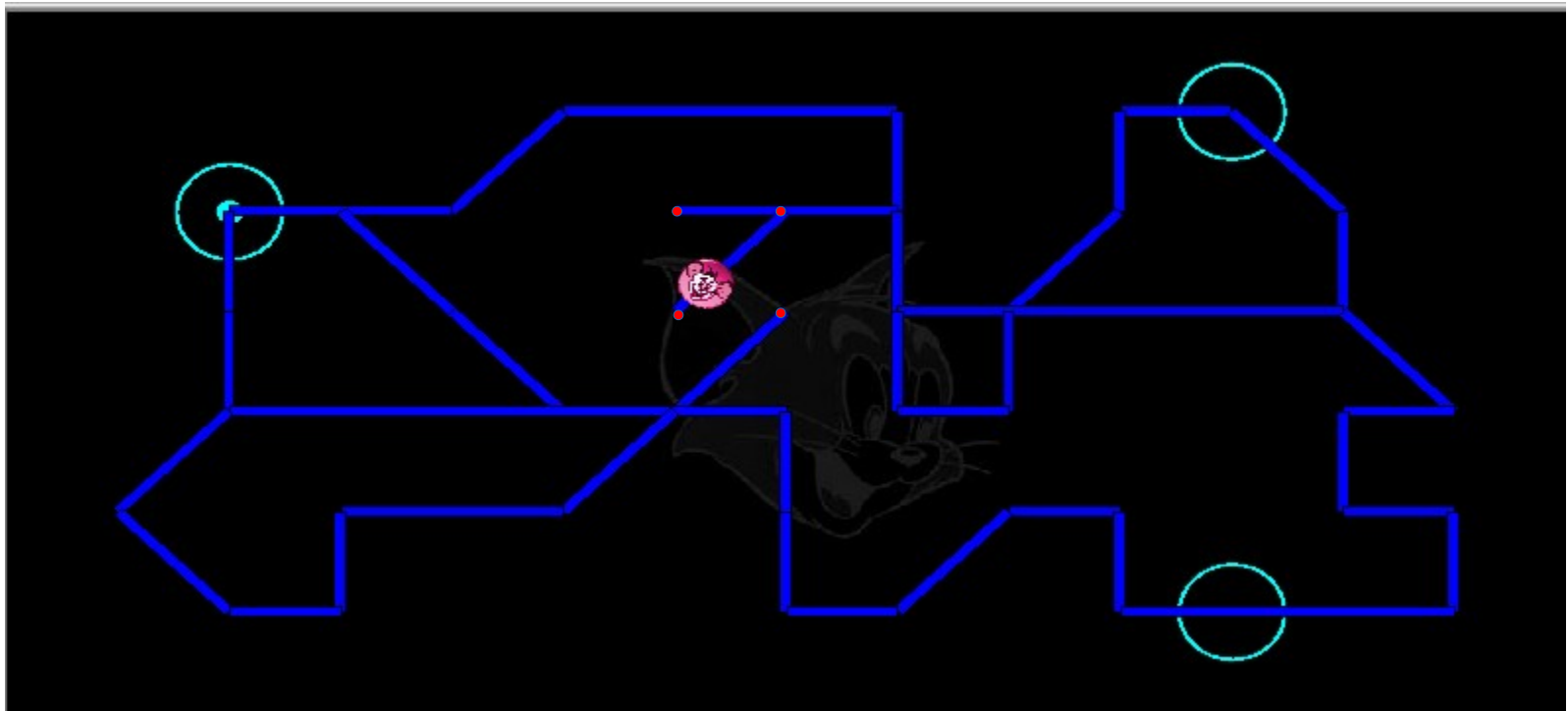
universidade de aveiro
theoria poiesis praxis



First Iteration



universidade de aveiro
theoria poiesis praxis





How to decide next position:

- The perceived map introduces the concept of **cell expansion**.
- A cell is expanded when the robot has **visited it** (the robot was < 0.1 units of distance from its center, with exception for the first cell).
- The next cell will be the **closest one** to the **current robot position** that is not expanded (**breath first**).

Controller:

- The line sensor **does not have noise**, so a **proportional (P) controller** will handle the robot **movement**.
- The only exception happens when the robot's **direction** ($\pm 22.5^\circ$) is not equal to the angle **defined** by its **current position** and the **next cell center**.

In this case, **rotate** the robot **manually** to align it.

Results



	Map 1	Map 2	Map 3	Map 4	Map 5	Map 6
Average Map score	1.00	1.00	1.00	1.00	1.00	1.00
Median Map score	1.00	1.00	1.00	1.00	1.00	1.00
Average Path score	1.00	1.00	1.00	1.00	1.00	1.00
Median Path score	1.00	1.00	1.00	1.00	1.00	1.00

Number of Runs = 100
Cycle Time = 10



Localization:

- Movement model is **corrected** using both the **compass sensor** and the **line sensor**.
- The readings from the **compass** sensor **replace** the direction computed by the movement model at **each iteration**.
- The line sensor is used to detect **mismatches** between the **real position** and the **expected position**.



Localization:

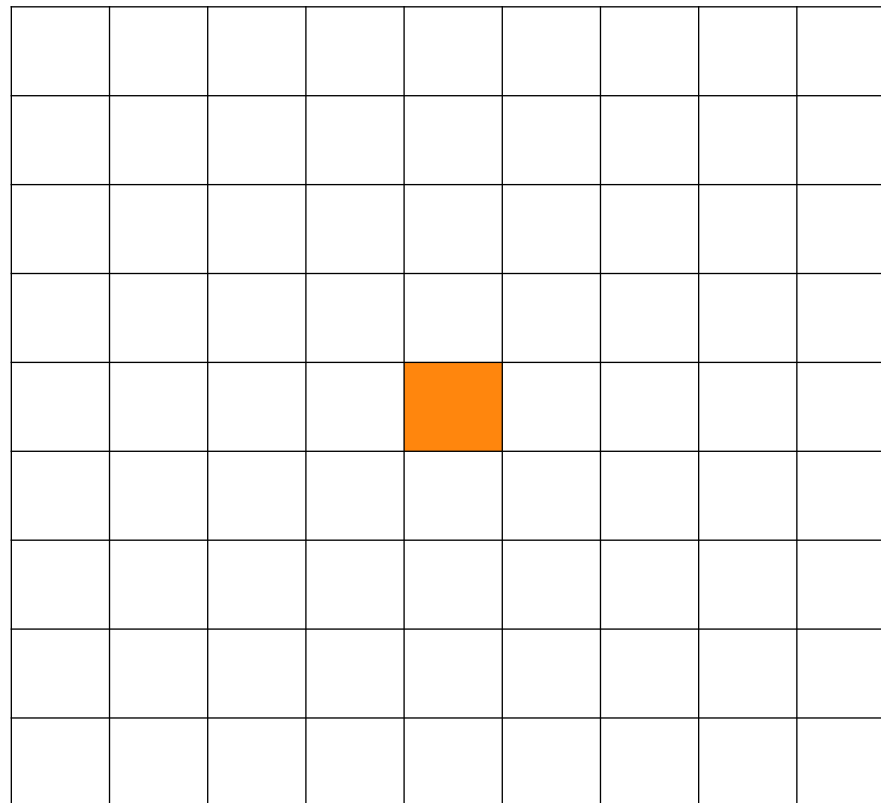
- When the line sensor **element detects a wall** and its **computed position does not intersect any possible walls**, the movement model **needs to be corrected**.
- To apply the correction, the robot will **search nearby positions** until one that **matches the line detection is found** (Chebyshev distance).

Second Iteration



universidade de aveiro
theoria poesis praxis

Localization:

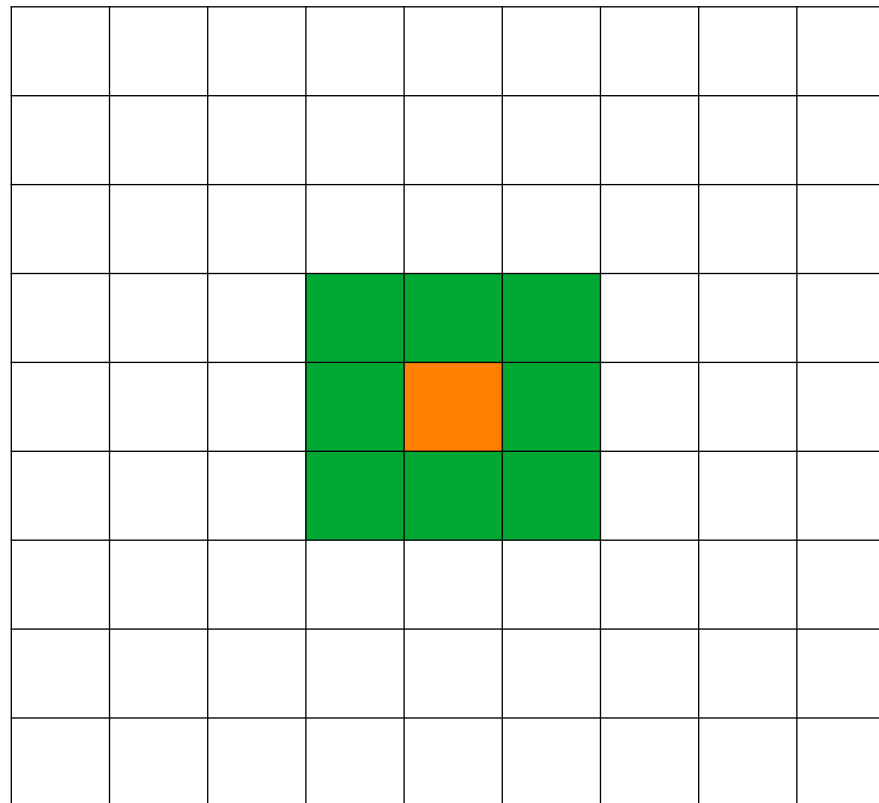


Second Iteration



universidade de aveiro
theoria poesis praxis

Localization:

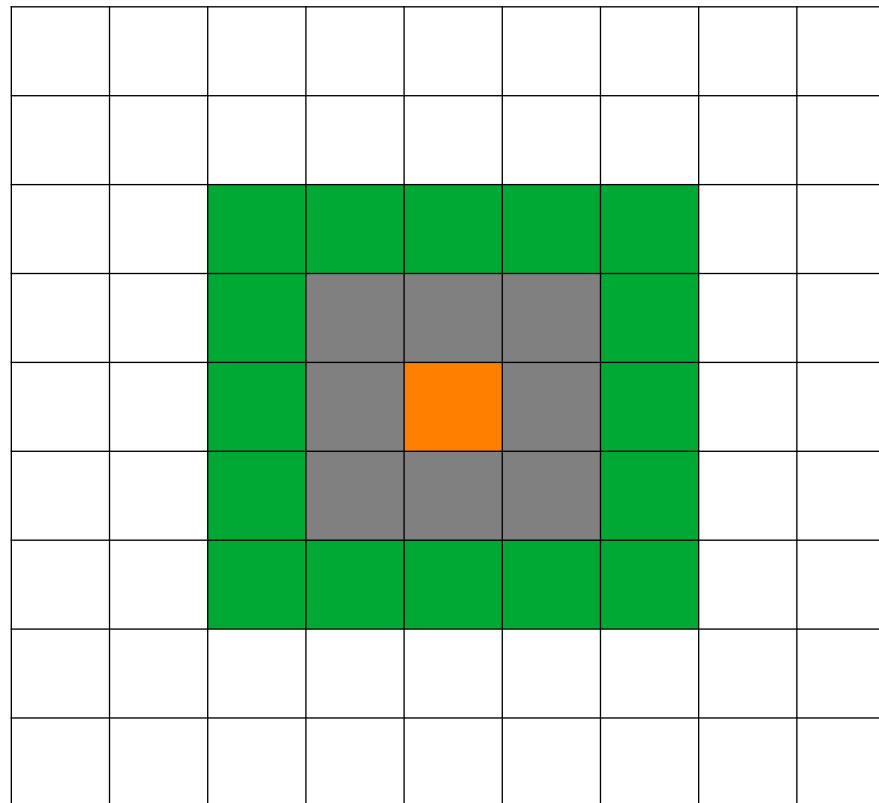


Second Iteration



universidade de aveiro
theoria poesis praxis

Localization:

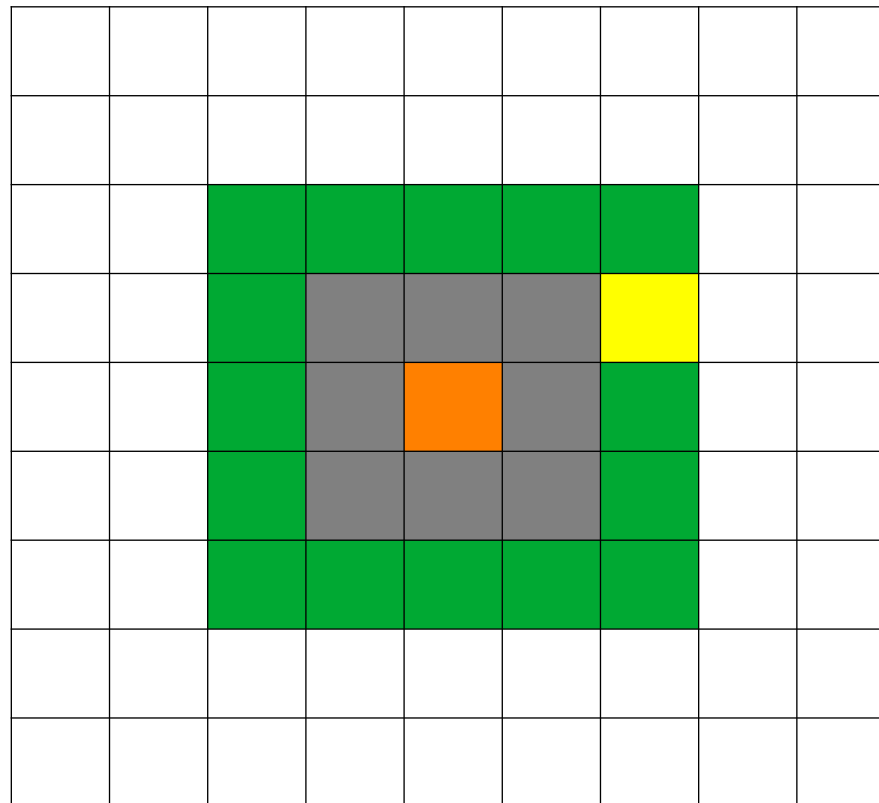


Second Iteration



universidade de aveiro
theoria poesis praxis

Localization:

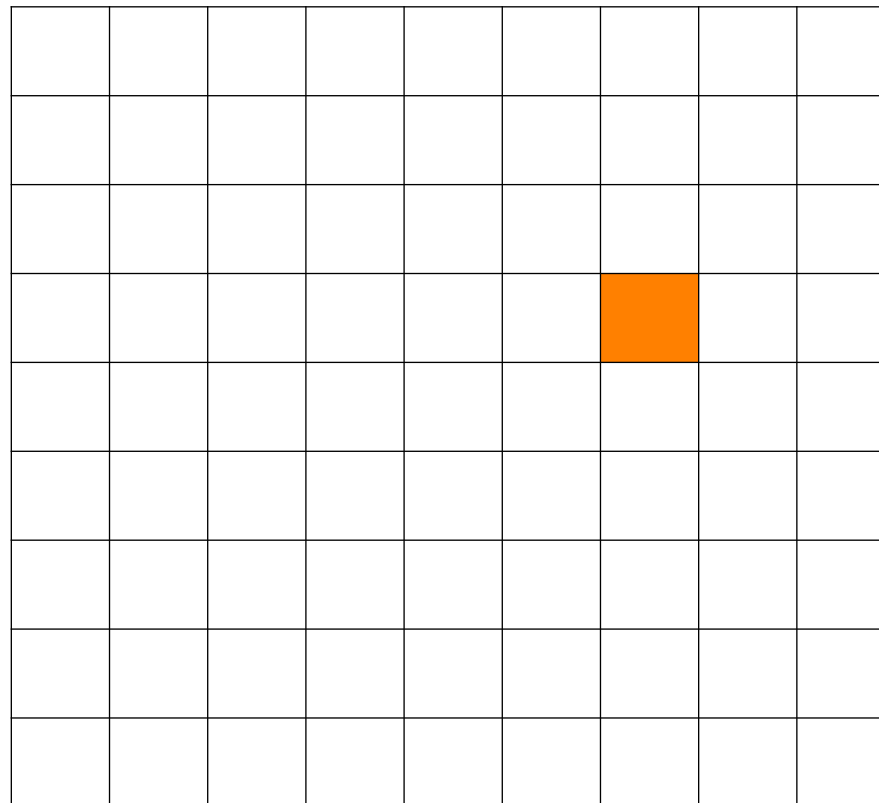


Second Iteration



universidade de aveiro
theoria poesis praxis

Localization:





Mapping:

- Due to noise, **non-existing walls can be detected.**
- If the **detected** wall **links** the **nearby cell** to a cell that **does not link with any cell** or **is not expanded**, the agent will correct the perceived map after trying to visit that cell.



Mapping:

- If the cell that belongs to the wrongly detected wall **is already expanded**, the agent will find the **mismatch in detections** and correct it by **trying to traverse that wall**.

Results



	Map 1	Map 2	Map 3	Map 4	Map 5	Map 6
Average Map score	0.98	0.99	0.99	0.99	0.99	0.97
Median Map score	1.00	1.00	1.00	1.00	1.00	1.00
Average Path score	0.97	0.99	0.76	0.87	0.90	0.89
Median Path score	1.00	1.00	1.00	1.00	1.00	1.00

Number of Runs = 100
Cycle Time = 10



Localization:

- The readings of the **compass sensor** are filtered using a **Kalman Filter**.
- The **standard deviation** of both the motors and the compass sensor are available at the start of simulation.
- Noise from both the **motors** and the **compass sensor** was applied to the **received value**, so both are used in the filter.

Final Results



universidade de aveiro
theoria poiesis praxis

Maps	Map 1	Map 2	Map 3	Map 4	Map 5	Map 6
Average Map score	0.99	0.99	0.97	0.98	0.96	0.92
Median Map score	1.00	1.00	1.00	1.00	0.97	0.99
Average Path score	0.90	0.99	0.59	0.70	0.74	0.75
Median Path score	1.00	1.00	1.00	1.00	0.76	1.00

Number of Runs = 100
Cycle Time = 10

- The correction of **wrongly detected walls** only occurs at the time of a **cell expansion** and **cell traversal**.
- This means that if **more than one wall** is wrongly detected, **only one** will be **corrected** at expansion time.
- To solve this, **before ending**, the agent should **check linkage differences for each cell** and correct it by **visiting each dubious connection**.

- The final iteration of the agent was **able to solve the challenge** proposed a **significant amount of times**.
- The **improvement** cited earlier could lead the agent to even more precise results, leaving the error to **cases where a non-existing road is wrongly detected on both cells**.



- https://wrfranklin.org/Research/Short_Notes/pnpoly.html
- <https://www.kalmanfilter.net>