

Rescue Line - Technical Challenge

RoboCupJunior Rescue Committee

Rescue Line - Technical Challenge 1	1
Field	1
Play	2
Start of Game	2
Lack of Progress	2
Scoring	2
End of Game	3

Rescue Line - Technical Challenge

Field

Technical Challenge field must contain, at least:

- 1 obstacle
- 1 intersection

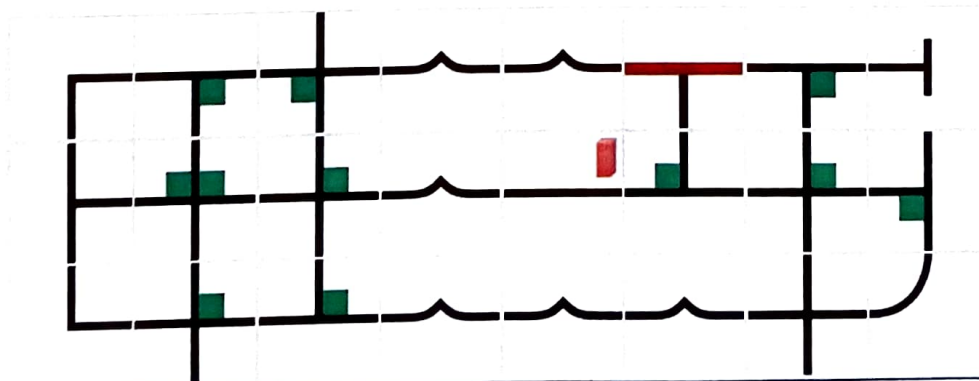


Fig1 - Example of a Technical Challenge field with 14 intersections and 2 obstacles along the desired path

Different fields should contain different numbers of intersections or dead ends before each obstacle, however the total number of intersections/dead ends and obstacles must keep the same in all fields.

Teams should be called to the competition area without previously knowing their assigned competition field.

Play

The aim of the technical challenge is to provide opportunities to the teams to show their ability to quickly modify the behaviour of their robots. General rules apply unless otherwise stated.

Start of Game

- Each team has a maximum of **4 minutes** for a game. The game includes the time for calibration and the scoring run.
- Robots will start behind the joint of the start tile.

Lack of Progress

A lack of progress occurs when:

- Team captain declares a lack of progress.
- The robot loses the black line without regaining it by the next tile in the sequence.
- The robot reaches a line that is not in the intended sequence.
- The robot fails at navigating hazards in the field.
- The robot does not behave as expected by this Technical Challenge when identifying the presence of an obstacle.

If a lack of progress occurs, the robot must be positioned on the start tile facing the path towards the goal tile and checked by the referee.

Scoring

A robot is awarded points for successfully navigating each hazard (intersections/dead ends and obstacles). Points are awarded per hazard when the robot has reached the next tile in sequence.

In addition to navigating the path as usual,

- The robot must count the number of intersections/dead ends faced along the path. Intersections without green markers are also counted.
- When the robot identifies the presence of an obstacle on the path, its behaviour must consider the number of intersections/dead ends counted up to that point:
 - If the total number of intersections/dead ends is **even**, the robot must move a little away from the obstacle and rotate 360° around its own axis clockwise. After that, the robot should navigate around the obstacle as usual by the left side of the obstacle.
 - If the total number of intersections/dead ends is **odd**, the robot must move a little away from the obstacle and rotate 360° around its own axis

- counterclockwise. After that, the robot should navigate around the obstacle as usual by the right side of the obstacle.
- c. If there are still intersections/dead ends on the path after the obstacle, the robot must continue counting, as there may be other obstacles further ahead.
 - d. During the run, the intersection/dead ends count must be continuous and must not be restarted after each obstacle.
 - e. The intersection/dead end count must be restarted after a LoP.
 - f. Each intersection/dead end successfully navigated is scored **10 points**.
 - g. Each obstacle successfully identified by the expected behaviour is scored **100 points**.
 - h. Exit bonus: The robot stops for 10 seconds at the goal tile, it is scored **50 points**.
 - i. Each LoP adds a penalty of **-5 points** to the Exit bonus as **(#Exit bonus - 5*#LoP)** (min. **0 points**)

End of Game

The game ends when:

- a. The robot reaches the goal tile and completely stops for 10 seconds.
- b. The **4 minutes** of allowed game time expires.
- c. Team captain calls the end of the game.