4.4 Evaluation Metrics

This subsection explores the evaluation metrics used to assess the performance of the local planning algorithms discussed in Chapter 2.2.3. These metrics are crucial for understanding how each algorithm contributes to safe, efficient, and smooth navigation for robots.

The evaluation criteria are categorized into safety, efficiency, and smoothness. Each metric provides valuable insights into different aspects of the algorithms capabilities and limitations:

4.4.1 Safety Metrics

The safety metrics are used to evaluate and determine if the local planners ensure the robot has safely navigated to its destination. The security level of local planners is measured using two metrics: the minimal distance to the nearest obstacle d_{o} and whether the robot completed the navigation task without collisions. If a collision occurs the navigation is aborted and the trial is considered a failure. From [36] the following equation was used to obtain the minimum distance:

$$d_o = \min\{d_i\}, \quad 1 \leq i \leq N \tag{10}$$

Where N represents the total number of lethal obstacle cells identified in the local costmap. Each cell in the local costmap can have one of 100 different cost values. Those values can be obtained by subscribing to the costmap topic:

```
$ /j100_0001/local_costmap/costmap
```

The topic publishes a 2-dimensional array representing the rows and columns of the costmap in a variable named data, which contains the values of each grid cell. The first value in the data array corresponds to the grid cell located in the lower right corner. By looping through the array, the grid cells with the highest cost values equal to 100 can be identified as lethal obstacles. These cells are represented in purple in Figure 27 and typically indicate areas where the robot cannot navigate, such as permanent and solid obstacles.

Once these cells are identified, their index positions both column and row are used to determine their current location within the local costmap relative to the *odom* frame. Subsequently, the center position of the robot is obtained by subscribing to the ground truth odometry topic relative to the *odom* frame, as described in

4.1.1. The Euclidean distance between each lethal obstacle cell and the current position of the robot is then calculated. All distances d_{i} are appended to an array, which is accessed at the end of the navigation process to determine the lowest value, representing d_{o}.

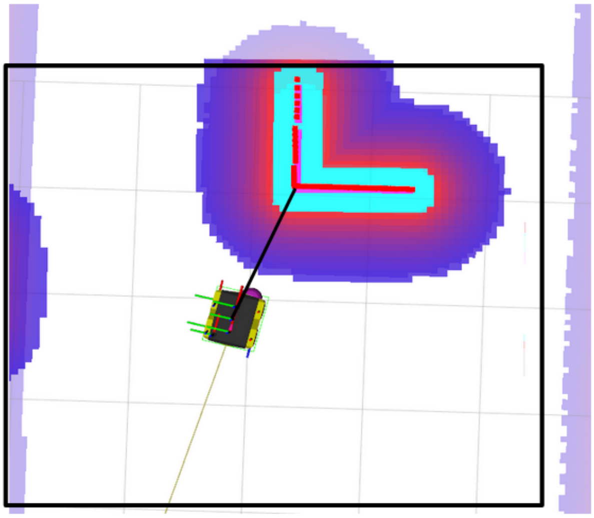

Figure 27: *Calculating the euclidean distance from the center of the robot to the occupied cells*

4.4.2 Efficiency Metrics

The efficiency metrics are used to evaluate the distance and time efficiency of the local planners. The total travel time T of the robot from the start to the goal is used to evaluate the motion efficiency of the algorithm using equation 11:

$$T = t_{\mathrm{end}} - t_{\mathrm{start}} \tag{11}$$

The navigation process begins at t_{start} and concludes when the robot reaches its final destination at t_{end}. To calculate the total distance traveled, the Euclidean distance between each consecutive pair of robot positions is computed for a trajectory consisting of N poses, starting from the initial position i. This is done using Equation 12, where p_{i} and $p_{\mathrm{i+1}}$ represent the positions of the robot at the current and subsequent time steps, respectively:

$$\text{Total distance} = \sum_{i=1}^{N-1} \|\vec{p}_{i+1} - \vec{p}_i\|, \tag{12}$$

Finally, if the robot cannot find an optimal path and gets stuck in a local minimum, a recovery behavior is triggered to escape the situation. Thus, counting the number of recoveries is a useful measure of the local planner's efficiency in addressing issues. This parameter is obtained from an action client that provides a feedback message indicating the number of recoveries the robot performed throughout the entire navigation process.

4.4.3 Smoothness Metrics

Smoothness metrics are utilized to evaluate the quality of motion commands generated by local planners. In this thesis, the smoothness performance of local planners is evaluated extensively through path smoothness which is calculated by obtaining the angle between two succeeding positions. Equation 13 was used from [34]:

$$\epsilon = 1 - \frac{\sum \alpha_i}{N_{\text{angles}}} \tag{13}$$

Where the N_{angles} represents the total number of angles computed along the path. The angle α_{i} indicates the angle between vectors $x_{\text{i}+1}$ and x_{i}. A smoother journey has smaller angles between consecutive vectors, which should possibly approach zero. As a result, a high value of this measure suggests a smoother path, with 1 representing the ideal value.

This chapter focuses on evaluating local planning algorithms for robotic navigation in varied environments, including simulation and real-world scenarios. The goal is to assess how well these algorithms enable the Clearpath Robotics Jackal robot to navigate through environments with static and dynamic obstacles. Key metrics such as safety, efficiency, and smoothness are used to measure algorithm performance in both simulated environments using Gazebo and real world tests. These metrics provide essential insights into the local planners effectiveness and are crucial for the subsequent analysis in the results and discussion chapter.