Homework of Motion Planning for Mobile Robots

mengchaoheng[*]

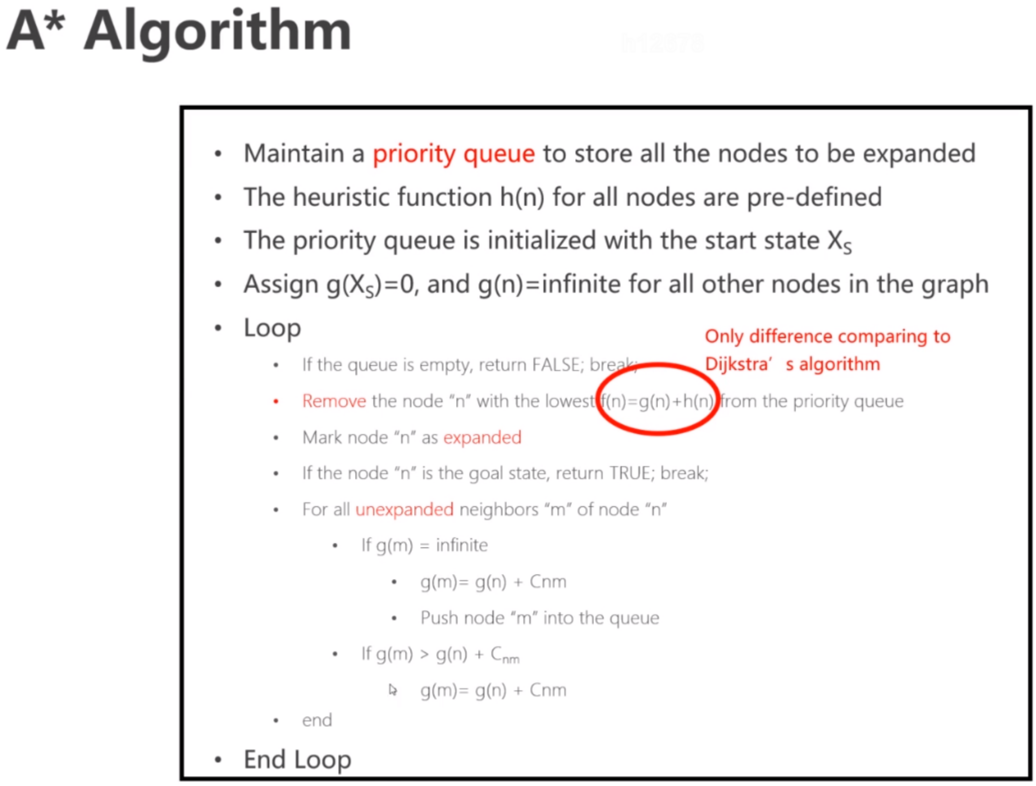
Institute of Intelligent Manufacturing, Guangdong Academy of Sciences, Guangzhou, China

Abstract

ROS implementation of A* algorithm and JPS algorithm。

Keywords: A*, JPS, MATLAB, ROS

1. Result of path searcheer



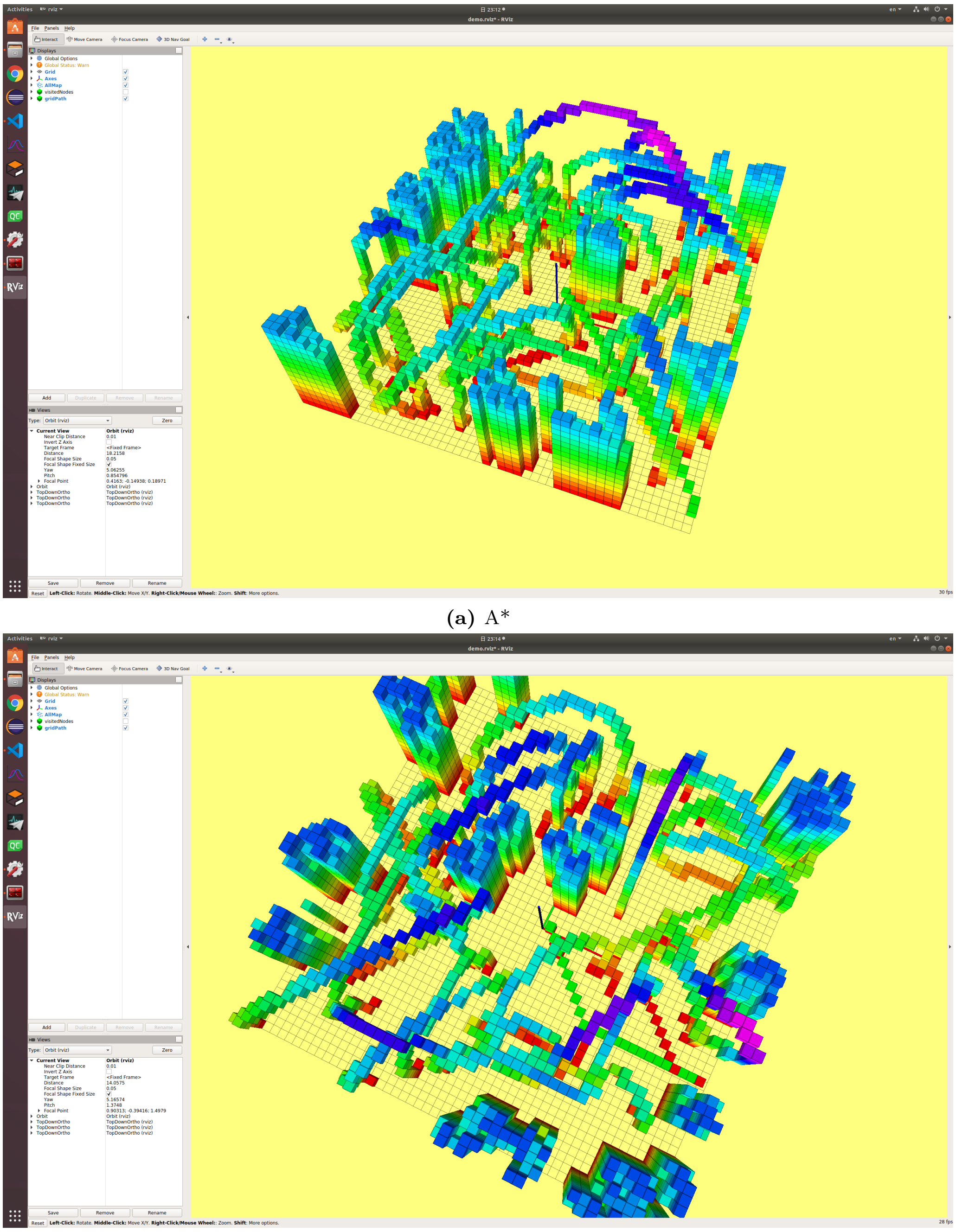
A* Algorithm

```
• Maintain a priority queue to store all the nodes to be expanded
• The heuristic function h(n) for all nodes are pre-defined
• The priority queue is initialized with the start state $X_S$
• Assign $g(X_S)=0$, and g(n)=infinite for all other nodes in the graph
• Loop
    • If the queue is empty, return FALSE; break;
    • Remove the node "n" with the lowest f(n)=g(n)+h(n) from the priority queue
    • Mark node "n" as expanded
    • If the node "n" is the goal state, return TRUE; break;
    • For all unexpanded neighbors "m" of node "n"
        • If g(m) = infinite
            • g(m)= g(n) + Cnm
            • Push node "m" into the queue
        • If $g(m) > g(n) + C_{nm}$
            • g(m)= g(n) + Cnm
    • end
• End Loop
```

Only difference comparing to Dijkstra' s algorithm

Figure 1: pseudocode of A* algorithm

[*]Corresponding author
Email address: ch.meng@giim.ac.cn (mengchaoheng)



Type: Orbit (rviz)

Current View	Orbit (rviz)
Near Clip Distance	0.01
Invert Z Axis	□
Target Frame	<Fixed Frame>
Distance	18.2158
Focal Shape Size	0.05
Focal Shape Fixed Size	☑
Yaw	5.06255
Pitch	0.854796
Focal Point	0.4163; -0.14938; 0.18971
Orbit	Orbit (rviz)
TopDownOrtho	TopDownOrtho (rviz)
TopDownOrtho	TopDownOrtho (rviz)
TopDownOrtho	TopDownOrtho (rviz)

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click/Mouse Wheel:** Zoom. **Shift:** More options.

(a) A*

Type: Orbit (rviz)

Current View	Orbit (rviz)
Near Clip Distance	0.01
Invert Z Axis	□
Target Frame	<Fixed Frame>
Distance	14.0575
Focal Shape Size	0.05
Focal Shape Fixed Size	☑
Yaw	5.16574
Pitch	1.3748
Focal Point	0.90313; -0.39416; 1.4979
Orbit	Orbit (rviz)
TopDownOrtho	TopDownOrtho (rviz)
TopDownOrtho	TopDownOrtho (rviz)
TopDownOrtho	TopDownOrtho (rviz)

Left-Click: Rotate. **Middle-Click:** Move X/Y. **Right-Click/Mouse Wheel:** Zoom. **Shift:** More options.

(b) JPS and A*

Figure 2: path

2. Comparison of different heuristic functions

Randomly select 3 maps, set the end point to the lower right corner of the map and the height of two grids. Compare the average operating efficiency of different heuristic functions as shown in the following table:

Table 1: Comparison of different heuristic functions

heuristic function	run time (ms)	visited nodes
Diagonal	0.541511	450
Euclidean	0.781643	679
Manhattan	0.135416	28
Diagonal with Tie Breaker	0.123847	32

3. A* and JPS

JPS works better in complex maps in small spaces with many obstacles. But A* works better on maps with a lot of open space.