|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждениевысшего образования«МИРЭА - Российский технологический университет»РТУ МИРЭА |

**ОТЧЕТ ПО ЗАДАНИЮ**

по дисциплине

«Управление интеллектуальными роботами и робототехническими системами»

Выполнили студенты группы КРБО-01-19 Верхоланцев Д.В.

Нестерова М.В

Принял Голубов В.В.

Работы выполнена «\_\_»\_\_\_\_\_\_\_202\_\_ г.

«Зачтено»

«\_\_»\_\_\_\_\_\_\_202\_\_ г.

Москва 2023

**Цель работы:** получение пути от точки до точки в дискретном пространстве с изменяемым размером дискреты при помощи алгоритмов Дейкстры, А\* и RRT.

**Ход работы:**

**1) Получение карты и её дискретизация.**

Процесс получения карты с использованием данных одометрии и лидара подробно описан в предыдущей работе. Дискретизация карты происходит следующим образом:

1)Для каждой точки данных находим ячейку сетки, которой она принадлежит.

2)Если расстояние от этой точки до центра ячейки меньше квадратного корня из размера ячейки, полагаем, что в данной ячейке находится препятствие. В матрице, соответствующей имеющейся сетке, обозначим элементы с препятствиями единицами, а свободные – нулями.

**2) Получение взвешенного графа на основе имеющейся матрицы.**

Для формирования графа использовалась библиотека NetworkX. Процесс формирования графа происходит следующим образом:

1)Присваиваем каждому нулевому элементу матрицы идентификационный номер.

2)Проходимся по всем нулевым элемента матрицы, добавляя в граф в качестве узлов элементы матрицы достижимые при совершении одного из восьми разрешённых шагов. Вес рёбер определим как расстояние между ячейками.

**3) Алгоритм Дейкстры.**

1)Создаём очередь из неизученных вершин графа. Расстояние до стартовой вершины зададим равным нулю.

2)Пока очередь не станет пустой, выбираем вершину с минимальным расстоянием.

3)Изучаем соседей выбранного узла. Если расстояние от выбранной вершины до соседа меньше того, что было присвоено ему ранее, присваиваем ему новое значение. Возвращаемся ко второму пункту.

4)Итогом предыдущей части алгоритма станет определение кратчайшего расстояния от начальной вершины до каждой другой. Осталось методом вычитания из расстояния до конечной точки веса перехода определить путь от конечной до начальной точки.

**4)А\***

1)Создаём список посещённых вершин и список потенциальных вершин. Задаём для начальной вершины значения расстояния до неё (g), от неё до конечной (h) и их сумму (f). Добавляем эту вершину в список потенциальных.

2)Пока список потенциальных вершин не опустеет, выбираем вершину с минимальным f. Добавляем полученную вершину в список посещённых. Если вершина соответствует конечной, останавливаем цикл.

3)Для каждой непосещённой вершины, соседствующей с выбранной на предыдущем шаге, считаем значения h, g и f. Если рассчитанное расстояние g для вершины больше того, что ей было присвоено ранее, выбираем следующего соседа. В противном случае, присваиваем вершине рассчитанные значения и добавляем её в список потенциальных вершин. Переходим ко второму пункту.

4)Итогом работы предыдущей части алгоритма является список посещённых вершин, где для каждой вершины обозначена длина кратчайшего путь до неё. Воспользуемся тем же методом получения пути, который применялся в алгоритме Дейкстры.

**5)RRT**

В отличие от прошлых двух алгоритмов данный алгоритм не будет опираться на имеющийся граф. Алгоритм RRT был реализован следующим образом:

1) Присваиваем каждому нулевому элементу матрицы идентификационный номер. Создадим пустой граф и добавим в него стартовую точку, соответствующую центру выбранной ячейки. Найдём расстояние от точки, соответствующей последней добавленной вершине графа, до конечной (далее «расстояние»).

2) Пока расстояние больше . Выбираем случайную точку среди свободных.

3) Находим ближайшую к случайно выбранной вершину графа. Пошагово строим отрезок между ними. Если натыкаемся на препятствие или превышаем максимальную длину отрезка, останавливаемся и запоминаем координаты точки остановки. Добавляем эту точку к графу, соединяя её с ближайшей точкой ребром направленным от новой точки к старой. Таким образом мы получим граф направленный из всех вершин к начальной. Найдём расстояние от только что добавленной точки до конечной и вернёмся ко второму пункту.

4) Добавим к графу конечную точку. Найдём путь от неё до начальной путём перехода от вершины к вершине в соответствии с направлением рёбер.