readme

Post-Disaster Operations Improvement: Route Planning and Optimization for Swarm UAVs in Flight-Restricted Areas to Enhance Rescue Efforts

Muhammed Mirac Ozer

**Algorithm 1** Detection of intersection of line segment.

{The line segments are given with coordinates.}

{Line segments given on the plane.}

{Coordinates of line segments.}

{The coordinates of the given line segment

{Vector consisting of measured intersection points.}

**Input:** , The number of line segments formed by the boundaries of the obstacles

**for** to **do**

**Input:** ,

**end for**

**for** to **do**

**if**  **then**

Since the i'th line segment is parallel to the given line segment, it is removed from the list.

values can be adjusted again.

is accepted.

**else**

**end if**

**end for** {Intersection conditions are checked.}

**for** to **do**

{ their lengths can be calculated by considering the coordinates of the endpoints in accordance with and .}

**end for**

**if**  **then**

{the argument value}

{case where }

{The Lth line segment intersects with the given line segment.}

**else**

The line segment does not intersect with the given segment.

**end if**

**end for** (step 2)

{A list of intersecting straight lines and their intersection points are written.}

**for**  to  **do**

Print {The coordinates of the intersection points are printed.}

Print

**end for**

This algorithm is suitable for use in cases where the obstacles are polygonal. When obstacles are presented in the form of a circle, the algorithm is easily implemented and the object's perimeter is considered an obstacle. For this, the center of the obstacles and the radius () must be known. When the obstacles are given in the form of a circle, the condition for determining the intersection with the straight-line segment arises by solving the equation of the straight line and the equation of the circle together. If the equality condition in Equation 2 is met, the line is tangent to the circle. When it is small, an intersection occurs at two points as shown in Equation 3. The intersection conditions in this case () are also shown in Equation 1.

|  |  |
| --- | --- |
|  | (1) |
|  | (2) |
|  | (3) |

Thus, when the obstacles are given in the form of a circle, the conditions of Equation 1 and Equation 3 are checked and the intersection coordinates are calculated with the formula Equation 2. In both cases, when the UAV encounters an obstacle on its straight-line route (polygonal or circular), it moves in a straight line again by taking the shortest path to the second intersection line.

A computational experiment has been conducted on the proposed algorithms. To do this, a written application and the Google Maps application were used to launch a number of UAV flights from Piazza AVM on Azerbaijan Boulevard in Kahramanmaraş city to Dedezade Street, where damaged buildings are concentrated. The starting and ending points of the UAVs are connected by a straight line. To apply the above algorithms, obstacles along the flight path were determined and the route equation was determined with this algorithm.