Trapezoidal Decomposition: A Roadmap Algorithm for Robotic Motion Planning

(Submitted by Pavan Poudel)

A basic robotic motion planning problem is to produce a continuous motion that connects a start position of a robot S and a goal position G, while avoiding collision with known obstacles. The robot and obstacle geometry is described in a 2D or 3D workspace, while the motion is represented as a path in the configuration space. Among various motion planning algorithms, roadmap algorithms are the one which breaks down the movement task into discrete motions that satisfy movement constraints and possibly optimize some aspect of the movement by creating some roadmap.

A roadmap, RM, is a union of curves such that for all start and goal points in Q_{free} that can be connected by a path. It has following properties:

- **Accessibility:** There is a path from $q_{start} \in Q_{free}$ to some q' ∈ RM
- **Departability:** There is a path from some q''∈ RM to $q_{goal} \in Q_{free}$
- **Connectivity:** there exists a path in RM between q' and q''
- **One dimensional**

RoadMap Path Planning steps:

1. Build the roadmap
    a) nodes are points in Q_{free} (or its boundary)
    b) two nodes are connected by an edge if there is a free path between them
2. Connect start and end goal points to the road map at point q' and q'', respectively
3. Find and connect a path on the roadmap between q' and q''

The result is a path in Q_{free} from start to goal

There are different deterministic and probabilistic roadmap algorithms for the motion planning. Some of them are:

- **Visibility Graph Method**
- **Generalized Voronoi Diagram**
- **Trapezoidal Decomposition**
- **Boustrophedon Decompositio**n
- **Sensor-based coverage**

I have implemented Trapezoidal Decomposition among them.

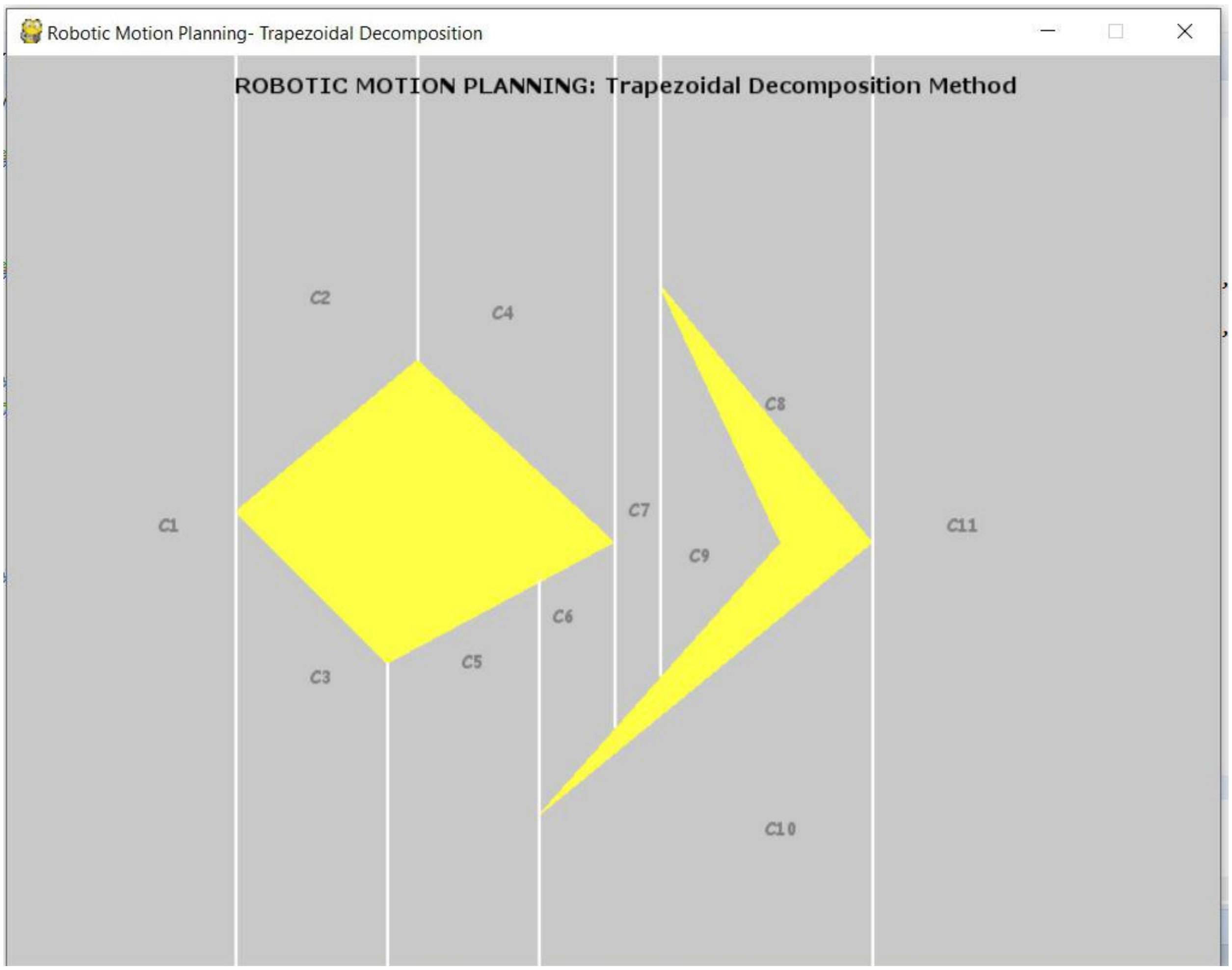


Figure 1: Free space configuration of obstacles and Trapezoidal Decomposition

Trapezoidal Decomposition:

1. The free space is decomposed into a number of cells by creating vertical sweeplines passing through a vertex of a polygonal obstacle; each cell consists of a node of an obstacle.
2. If two cells share a common boundary, they are neighbors and are connected.
3. Find the contained cells of robot position and the goal position
4. Find a path to connect these two cells

Fig.1 shows the configuration of the robot and obstacles and the cells after decomposition and the Fig.2 shows the respective graph connecting the cells.

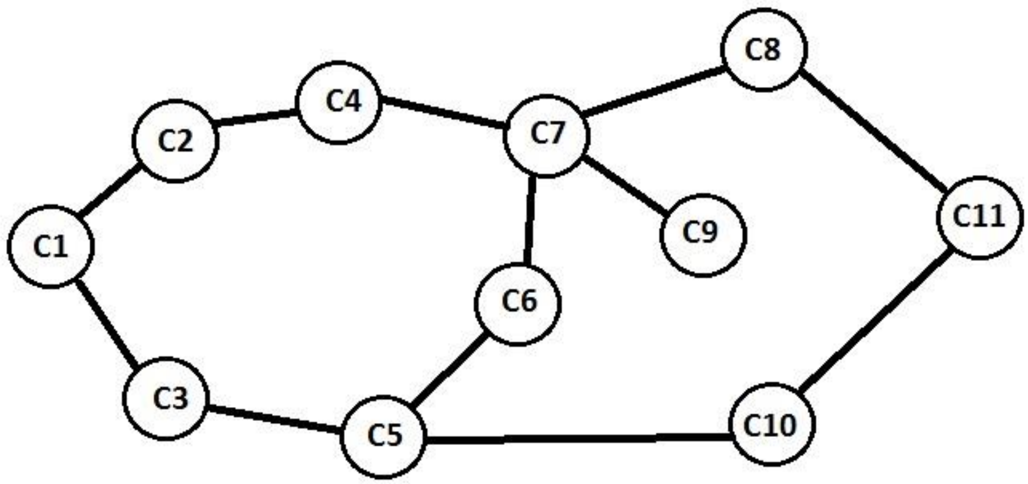
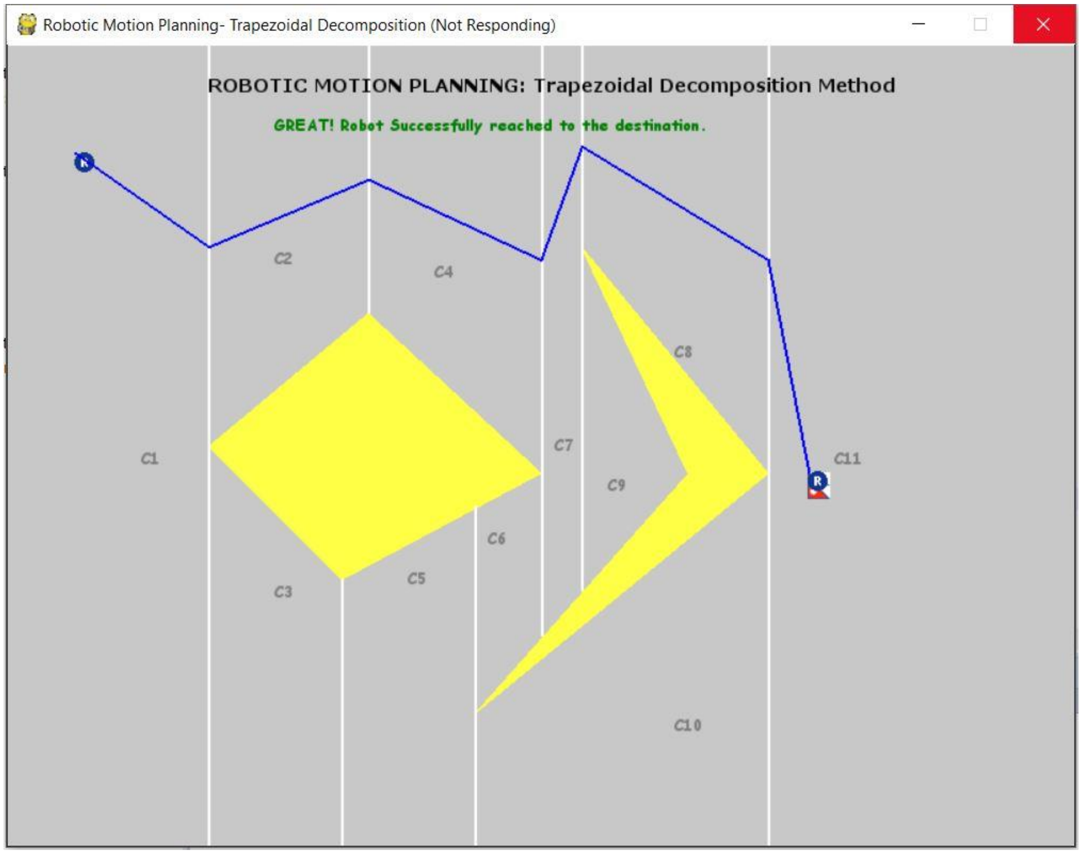
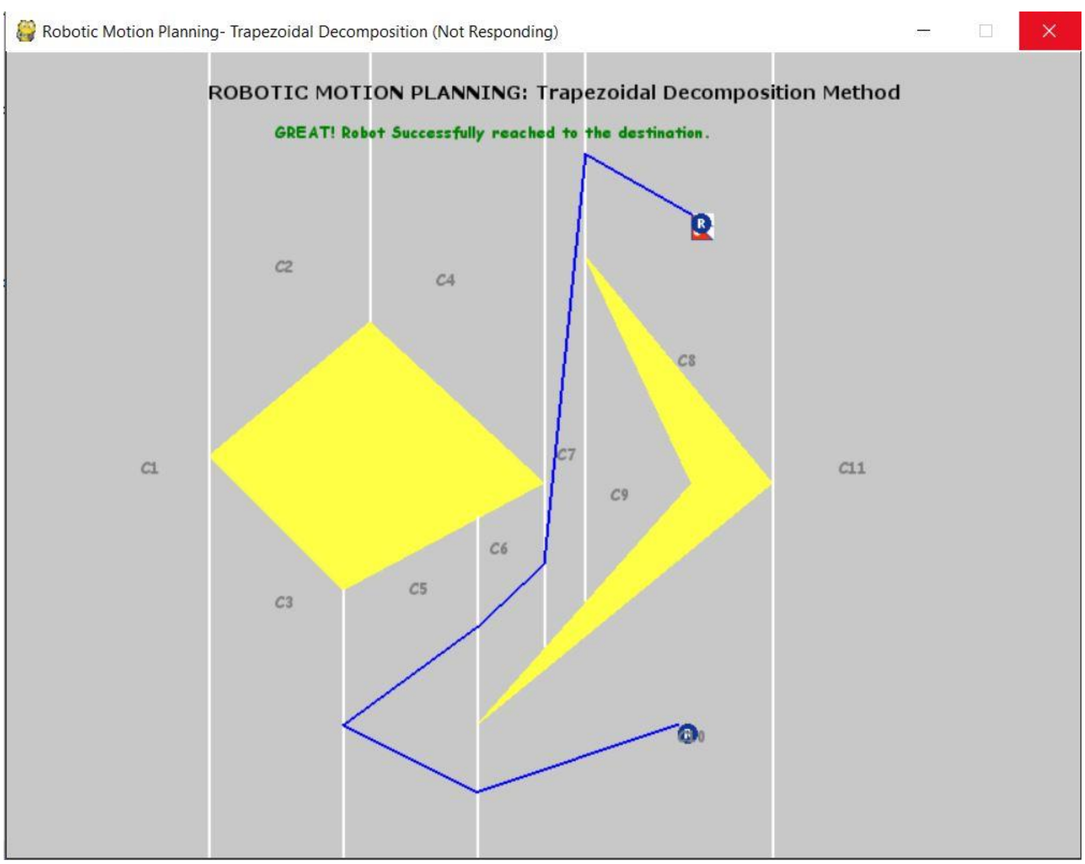


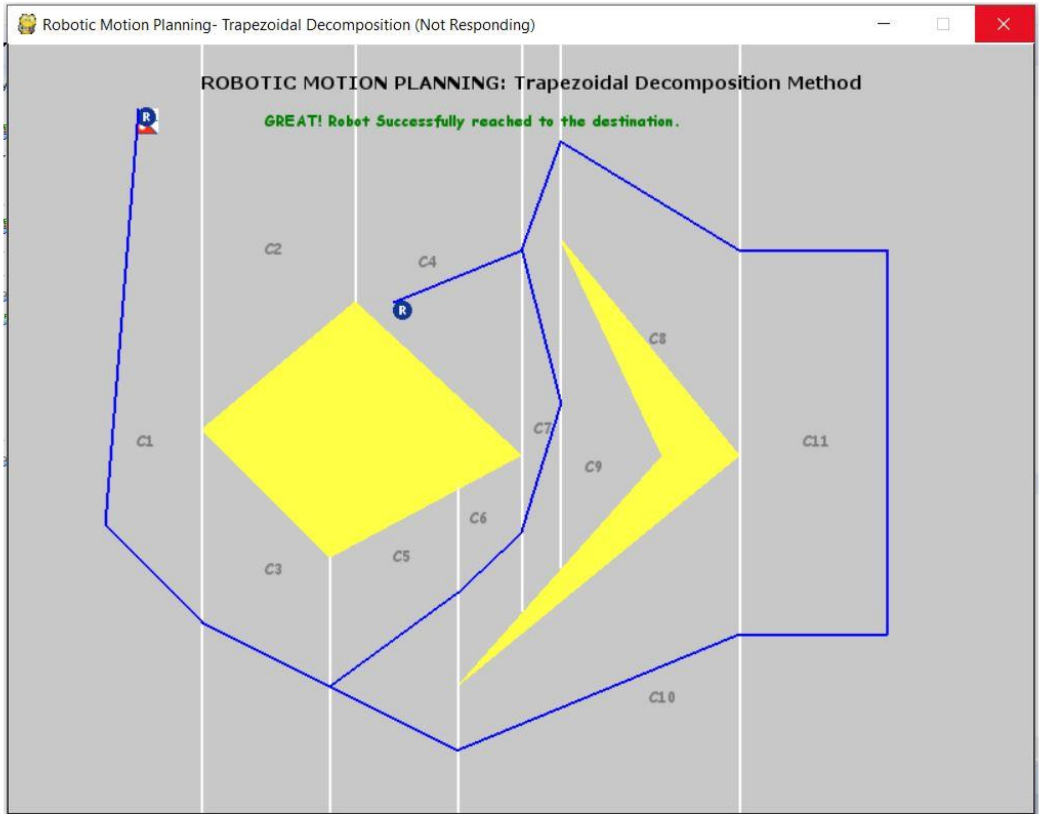
Figure 2: Connectivity graph of the decomposed cells

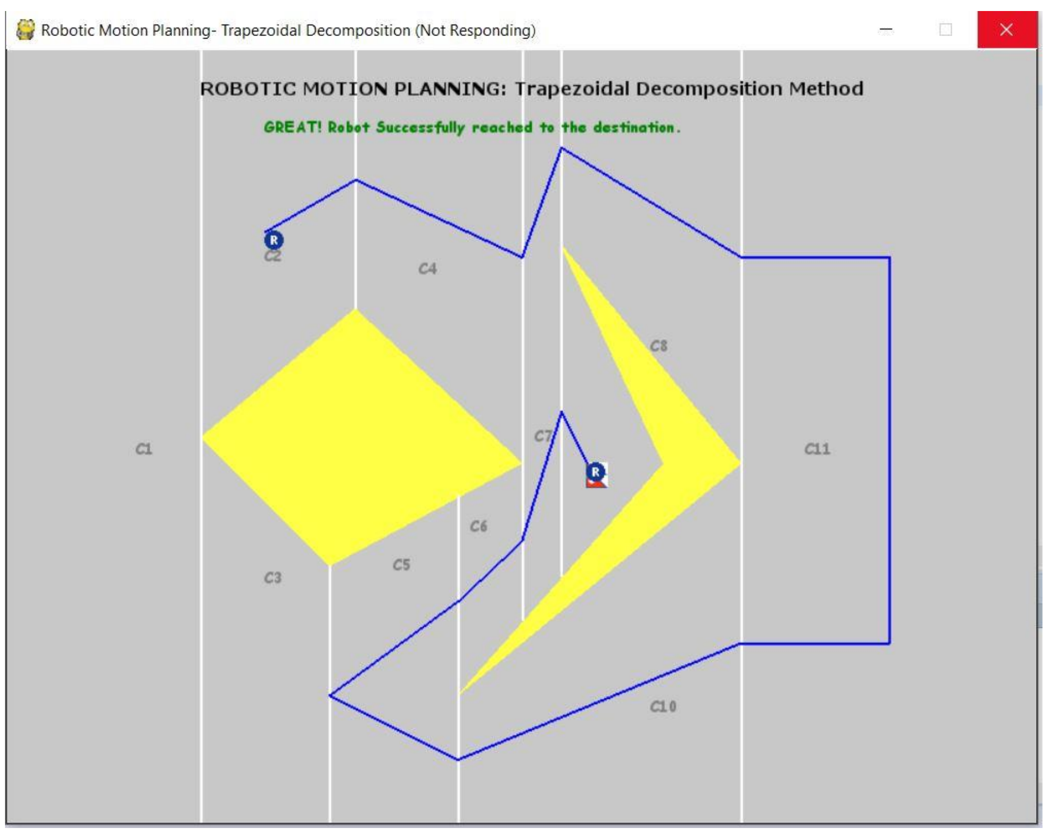
Trapezoidal Decomposition Method creates a graph of “roads” that will move the robot through the space. Robot just get on to one of the node and eventually get off again from some node to the position of the goal.

Below are some snapshots of the implementation:


Robotic Motion Planning- Trapezoidal Decomposition (Not Responding)
ROBOTIC MOTION PLANNING: Trapezoidal Decomposition Method
GREAT! Robot Successfully reached to the destination.
C1
C2
C3
C4
C5
C6
C7
C8
C9
C10
C11


Robotic Motion Planning- Trapezoidal Decomposition (Not Responding)
ROBOTIC MOTION PLANNING: Trapezoidal Decomposition Method
GREAT! Robot Successfully reached to the destination.
C1
C2
C3
C4
C5
C6
C7
C8
C9
C11


Robotic Motion Planning- Trapezoidal Decomposition (Not Responding)
ROBOTIC MOTION PLANNING: Trapezoidal Decomposition Method
GREAT! Robot Successfully reached to the destination.
C1
C2
C3
C4
C5
C6
C7
C8
C9
C10
C11


Robotic Motion Planning- Trapezoidal Decomposition (Not Responding)
ROBOTIC MOTION PLANNING: Trapezoidal Decomposition Method
GREAT! Robot Successfully reached to the destination.
C1
C2
C3
C4
C5
C6
C7
C8
C10
C11

*** Thank you ***