Reading report
Robot motion planning in dynamic environment[1]

Dharmin Bakaraniya

April 26, 2018

Abstract

This paper presents a method for computing the motions of a robot in dynamic environments, subject to the robot dynamics and its actuator con- straints. This method is based on the concept of Velocity Obstacle, which defines the set of feasible robot velocities that would result in a collision between the robot and an obstacle moving at a given velocity. The avoidance maneuver at a specific time is thus computed by selecting robot's velocities out of that set. A trajectory consisting of a sequence of avoidance maneuvers at discrete time intervals is generated by a search of a tree of avoidance maneuvers. An exhaustive search computes near minimum-time trajectories, whereas a heuristic search generates feasible trajectories for on-line applications. These trajectories are compared to the optimal trajectory computed by a dynamic optimization that minimizes motion time, subject to robot dynamics, its actuator limits and the state inequality constraints due to the moving obstacles. This approach is demonstrated for planning the trajectory of an automated vehicle in an Intelligent Vehicle Highway System sce- nario.

1 Summary

- Motion planning in dynamic environment is considerably more difficult that static motion planning because it needs to solve path planning and velocity planning at the same time.
- Motion planning in staric environment is guaranteed to come up with a solution if one exists but in dynamic environment the solution is 'intractable'.
- The advantages of this approach are
  1. 'It permits efficient geometric representation of potential avoidance maneuvers of the moving obstacle'
  2. No limit on number of obstacle to be avoided.
  3. No seperate representation for stationary and moving obstacle.

4. Considers robot dynamics and actuator constrints.

- Velocity Obstacle (VO) is an extention of Configuration Space Obstacle for a dynamic environment.
- For any given obstacle B_1 and moving object A, there will be a collision if $\mathbf{v}_{A,B_1}$ lies between the two tangents of B_1 (λ_f and λ_r).
- The circle around which the tangents are to be drawn depend on the velocity of A. This makes a lot more sense because if A was moving very slowly then the circle around B_1 very small which means very specific movement will actually result into collision.
- The area between these tangents is called collision cone.
- Velocity obstacle is addition (Minkowski vector sum) of collision cone with the velocity $\mathbf{v}_{B_1}$

$$VO = CC_{A,B_1} + \mathbf{v}_{B_1}$$

- For stationary obstacle, $\mathbf{v}_{B_1} = 0$ and thus VO will not be translated.
- VO for multiple obstacle can be combined

$$VO = \bigcup_{i=1}^{m} VO_i$$

- VO can be calculated periodically to fight against variable velocity.
- To avoid obstacle, the robot calculates it reachable velocities by considering the acceleration that it can achieve and the time till next interval.
- From this reachable velocities, the robot can then calculate reachable avoidance velocities by substracting VO from reachable velocities.
- To calculate avoidance trajectories, authors suggest two approaches.
    1. Global search over all feasible meaneuvers at regular interval.
    2. Heuristic search for online application when the trajectories of obstacles are not known before hand. There are 2 hueristic proposed by the authors
        (a) TG (to goal): Selects highest avoidance velocity in direction of goal
        (b) MV (maximum velocity): Selects maximum velocity towards the goal within α variance.
- The heuristic approaches is a function which prioritises survival of robot, reaching the target and minimizing travel time.
- TG takes 6.07 s to complete the same task (avoiding two moving obstacles with different velocities and get to other side of the road) while MV takes 4.85 s. The optimal solution takes 4.6 s for the robot to complete this task.

2 Doubts and notes

- I could not understand most of the part in section 3.
- There were no experiments performed on a real robot to test the theory.
- There was no data given on how much actual processing time each iteration takes.
- The approach seems quite simple but very costly (computationally).

References

[1] Fiorini, Paolo, and Zvi Shiller. "Robot motion planning in dynamic environments." International Symposium of Robotic Research. Springer Munich, Germany, 1995.