
MMS in the presence of quasi-dynamic obstacles

Oren Salzman, Tel-Aviv University



Setting

- **Workspace** –
 - Static polygonal obstacles
 - Quasi-dynamic polygonal obstacle
- **Robot** - translating and rotating polygon
 - Translational speed
 - Angular speed
- **Time** – **blue** and **red** time frames
 - **Blue time frame** – Your robot is allowed to move, quasi-dynamic obstacle is static
 - **Red time frame** – Your robot is allowed static, quasi-dynamic obstacle may move

Objective

- A set of points of interest is supplied in advance and updated as time progresses
- Plan a collision-free path hitting all points of interest as fast as possible

League

- You will compete against each other, each team serving as the other's quasi-dynamic obstacle
- In every match both teams will have identical robots and will be able to move in alternating intervals
- First team to reach all points of interest wins

Details

- Team sizes: 3
- Programming language: C++
(recommended)
- Supplied material: MMS infrastructure,
CGAL, BOOST
- Final project must compile and run under
Windows OS

Milestones (tentative)

- 28.3 Form teams & submit a general description of your planner
- 18.4 Planned project presentation
- 24.4 Final plan submission
- 30.5 Proof of Concept
- 25.7 Prototype & integration
- 2.9 Submission
- 5.9 League