Motion Planning | Collision Avoidance

Autonomous Mobile Robots

Martin Rufli – IBM Research GmbH
Margarita Chli, Paul Furgale, Marco Hutter, Davide Scaramuzza, Roland Siegwart
Classic collision avoidance | overview

- Methods compute actuator commands based on local environment
- They are characterized by
  - Being light on computational resources
  - Being purely local and thus prone to local optima
  - Incorporation of system models
Dynamic Window Approach (DWA) | working principle

- Robot is assumed to instantaneously move on circular arcs \((v, \omega)\)
- 2D evidence grid is transformed into \((v, \omega)\) input-space based on robot deceleration capabilities / kino-dynamics, leading to \(V_o\)
- Static window \(V_s\) constrains velocities
- Dynamic window \(V_d\) accounts for vehicle dynamics
- Selection of \((v, \omega)\)-pair within \(V_r = V_o \cap V_s \cap V_d\) maximizing objective containing heading, distance to goal and velocity terms
Dynamic Window Approach (DWA) | working principle

![Diagram of DWA working principle]
Dynamic Window Approach (DWA) | working principle
Dynamic Window Approach (DWA) | working principle
Dynamic Window Approach (DWA) | working principle
Dynamic Window Approach (DWA) | properties

- DWA accounts for robot kino-dynamics
- Cost function is prone to local optima
- The method assumes that objects are static
Velocity Obstacles (VO) | working principle

- The robot is assumed to move on piece-wise linear curves
- The Velocity Obstacle is composed of all robot velocities leading to a collision with an obstacle before a horizon time $\tau$
Velocity Obstacles (VO) | working principle

- The robot is assumed to move on piece-wise linear curves
- The Velocity Obstacle is composed of all robot velocities leading to a collision with an obstacle before a horizon time $\tau$
Velocity Obstacles (VO) | working principle

- The robot is assumed to move on piece-wise linear curves
- The Velocity Obstacle is composed of all robot velocities leading to a collision with an obstacle before a horizon time $\tau$
Velocity Obstacles (VO) | working principle

- The robot is assumed to move on piece-wise linear curves
- The Velocity Obstacle is composed of all robot velocities leading to a collision with an obstacle before a horizon time $\tau$

$$\|p_{RO} + v_R t\| < r_R + r_O$$

$$VO_{RO}^\tau = \bigcup_{0 \leq t \leq \tau} D\left(\frac{p_{RO}}{t}, \frac{r_{RO}}{t}\right)$$
The robot is assumed to move on piece-wise linear curves.

The Velocity Obstacle is composed of all robot velocities leading to a collision with an obstacle before a horizon time $\tau$. 

Velocity Obstacles (VO) | working principle
Velocity Obstacles (VO) | working principle

- The robot is assumed to move on piece-wise linear curves
- The Velocity Obstacle is composed of all robot velocities leading to a collision with an obstacle before a horizon time $\tau$
Velocity Obstacles (VO) | properties

- VO considers the velocity of other objects
- It is prone to local optima
- It does not model interaction effects
Interactive collision avoidance | overview

- Methods compute actuator commands based on local environment
- They are characterized by
  - Being light on computational resources
  - Being purely local and thus prone to local optima
  - Incorporation of system models and higher-order reflection
Reciprocal Velocity Obstacles | working principle

- The robot is assumed to move on piece-wise linear curves
- Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property
Reciprocal Velocity Obstacles | working principle

- The robot is assumed to move on piece-wise linear curves
- Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property
The robot is assumed to move on piece-wise linear curves.

Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property.

\[
VO_{RO}^t = \bigcup_{0 \leq t \leq \tau} D \left( -\frac{p_{RO}}{t}, \frac{r_{RO}}{t} \right)
\]
The robot is assumed to move on piece-wise linear curves.

Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property.

\[
VO_{RO}^{t} = \bigcup_{0\leq t \leq \tau} D\left(-\frac{p_{RO}}{t}, \frac{r_{RO}}{t}\right)
\]
The robot is assumed to move on piece-wise linear curves

Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property

\[ VO_{RO}^\tau = \bigcup_{0 \leq t \leq \tau} D \left( -\frac{p_{RO}}{t}, \frac{r_{RO}}{t} \right) \]
The robot is assumed to move on piece-wise linear curves

Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property
Reciprocal Velocity Obstacles | working principle

- The robot is assumed to move on piece-wise linear curves
- Identical to the Velocity Obstacles method, except that collision avoidance is shared between interacting agents – fairness property
Reciprocal Velocity Obstacles | properties

- Cost function is prone to local optima
- Interaction is handled via a fairness property
- The method is restricted to agents with omni-directional actuation
Collision Avoidance | further reading

- Integration of more complex motion models into reciprocal collision avoidance
- Integration with global search methods