Dynamic Obstacle Avoidance Strategies using Limit Cycle for the Navigation of Multi-Robot System

Ahmed Benzerrouk, Lounis Adouane, Martinet Philippe

Pascal Institute, UMR 6602 du CNRS
Clermont-Ferrand, France

Philippe.Martinet@irccyn.ec-nantes.fr
http://www.irccyn.ec-nantes.fr/~martinet
Outline of the presentation

Introduction
Motivation
On going work

Control architecture
Proposed control architecture

Modeling and control
Modeling
Attraction controller
Obstacle avoidance controller

Results
First experiment
Second experiment

Conclusion
**Motivation**

- **Control Architecture for a multi-robots cooperative system**
  - open
  - robust
  - satisfying many simultaneous objectives (velocity, formation, obstacle avoidance, ...)

- **Mobile robot control in encumbered environment**
  - Need to develop obstacle avoidance controller

- **Hybrid systems (continuous – discrete)**
**Motivation**  
*Multi robot control: state of the art*

**Multi mobile robots control architecture**

- **Centralised**
  - [Jones, 01], [Noreils, 93]

- **Distributed**
  - [Kube, 00], [Brooks, 86]

- **Cognitives**
  - [Latombe, 91], [Rimon, 92]

- **Reactives**
  - [Toibero, 07], [Egerstedt, 00]
Motivation  

**Multi robot control: state of the art**

- **Centralised**
  - [Jones, 01], [Noreils, 93]

- **Distributed**
  - [Kube, 00], [Brooks, 86]

- **Cognitives**
  - [Latombe, 91],
  - [Rimon, 92]

- **Reactives**
  - [Toibero, 07],
  - [Egerstedt, 00]
Motivation \hspace{1cm} Multi robot control : state of the art

Multi-robots systems

Robot Coordination

Leader-Follower

Emergent behavior

Virtual structure

[Tanner, 04], [Wang, 91]

[Balch, 98], [Antonelli, 07]

[Do, 07], [Li, 05]
On going work: Hybrid control architecture

Hybrid control Architectures

Elementary stable controllers

All the controllers (obstacles avoidance, objective attraction, etc.) must be designed in close relationship with the overall structure of control.

✓ Rigorous automatics control approach

✓ GPS,
✓ Cameras,
✓ Communication, ...

Coordination mechanism (CM)
**On going work:** *Multi-objectives controller design*

- Multi-mode control architectures (embedded in each robot),
- Fully distributed architecture,
- Satisfaction of local and global “robot-task” objectives,
- Hybrid systems theory as formal framework for controller design and coordination,
- Learning and adaptation parameters algorithm for controller design and coordination.

---

![Diagram](image.png)
Proposed architecture

Parameters of the formation to achieve

Obstacle Avoidance

Attraction to Dynamical Target

Hierarchical set-point selection

Control Law

Robot_i

Intended dimensions: Width: 368 Height: 368

Philippine Martinet

4th Workshop on Planning, Perception and Navigation for Intelligent Vehicles
IROS12, Vilamoura, Portugal, 7th October, 2012
Modeling: keeping a virtual formation structure (i.e. triangular)

\[
u_i = f(u_T, D_i, \Phi_i, \frac{\partial D_i}{\partial t}, \frac{\partial \Phi_i}{\partial t})
\]

\[
u_i = \begin{pmatrix} v_i \\ \omega_i \end{pmatrix} \quad u_T = \begin{pmatrix} v_T \\ \omega_T \end{pmatrix}
\]
Modeling

Kinematic model (unicycle)

Secondary dynamical target $T_i(x_{T_i}, y_{T_i}, \theta_{T_i})$

\[
\begin{align*}
\dot{x}_{T_i} &= v_T \cos(\theta_T) \\
\dot{y}_{T_i} &= v_T \sin(\theta_T)
\end{align*}
\]

Robot $i$ with $(x_i, y_i, \theta_i)$ pose

Kinematic model

\[
\begin{align*}
\dot{x}_i &= v_i \cos(\theta_i) \\
\dot{y}_i &= v_i \sin(\theta_i) \\
\dot{\theta}_i &= \omega_i
\end{align*}
\]

$\theta_i$, $v_i$ and $\omega_i$ are respectively the robot orientation, linear and angular velocities
Attraction controller

Position error can be defined by

\[ e_{x_i} = (x_{T_i} - x_i) = d_{S_i} \cos(\gamma_i) \]
\[ e_{y_i} = (y_{T_i} - y_i) = d_{S_i} \sin(\gamma_i) \]

\( d_{S_i} \) current distance between the robot \( i \) and its target \( T_i \)

\[ d_{S_i} = \sqrt{e_{x_i}^2 + e_{y_i}^2} \]

\( \gamma_i \) current angle of the robot according to its dynamical target

\[ \gamma_i = \arctan \left( \frac{e_{y_i}}{e_{x_i}} \right) \]

\[ \gamma_i = \frac{(e_{y_i}/e_{x_i})}{1 + (e_{y_i}/e_{x_i})^2} \]

Results

\[ \dot{d}_{S_i} = v_T \cos(\gamma_i - \theta_T) - v_i \cos(\gamma_i - \theta_i) \]

\[ \dot{\gamma}_i = \frac{v_T \sin(\theta_T - \gamma_i)}{d_{S_i}} - \frac{v_i \sin(\theta_i - \gamma_i)}{d_{S_i}} \]

Conclusion
Attraction controller

The set-point angle $\theta_{S_{at}}$ applied to the robot in order to reach its dynamical target keeps $\gamma_i$ constant, i.e., $\gamma_i = 0$. This can be achieved by solving the following equation:

$$\frac{v_T \cdot \sin(\theta_T - \gamma_i)}{d_{S_i}} - \frac{v_i \cdot \sin(\theta_i - \gamma_i)}{d_{S_i}} = 0$$

which leads to:

$$\theta_{S_{at}} = \arcsin\left(\frac{v_T}{v_i} \sin(\theta_T - \gamma_i)\right) + \gamma_i$$

$\text{Parameter } b = \frac{v_T}{v_i}$, with $b = \frac{v_T}{v_i} \leq 1$, $v_i \geq v_T$. 

---

Philippe Martinet

4th Workshop on Planning, Perception and Navigation for Intelligent Vehicles
IROS12, Vilamoura, Portugal, 7th October, 2012
**Obstacle avoidance controller** [KIM03, JIE06, ADOUANE09]

Principle: limit cycle approach

\((x_s, y_s)\) corresponds to the position of the robot according to the center of the convergence circle

**clockwise trajectory motion**

\[
\begin{align*}
\dot{x}_s &= y_s + x_s(R_c^2 - x_s^2 - y_s^2) \\
\dot{y}_s &= -x_s + y_s(R_c^2 - x_s^2 - y_s^2)
\end{align*}
\]

**counter-clockwise trajectory motion**

\[
\begin{align*}
\dot{x}_s &= -y_s + x_s(R_c^2 - x_s^2 - y_s^2) \\
\dot{y}_s &= x_s + y_s(R_c^2 - x_s^2 - y_s^2)
\end{align*}
\]
**Obstacle Avoidance**

**Static obstacles**

\[
\begin{align*}
\dot{x}_s &= ay_s + x_s(R_c^2 - x_s^2 - y_s^2) \\
\dot{y}_s &= -ax_s + y_s(R_c^2 - x_s^2 - y_s^2)
\end{align*}
\]

\[\theta_{Seo} = \arctan\left(\frac{y_s}{x_s}\right)\]

\[\alpha = \text{sign}(y_r)\]

---

**Results**

---

**Conclusion**

---

Philippe Martinet

4th Workshop on Planning, Perception and Navigation for Intelligent Vehicles

IROS12, Vilamoura, Portugal, 7th October, 2012
Dynamic obstacles

\[
\begin{align*}
\dot{x}_s &= \alpha y_s + x_s (R_c^2 - x_s^2 - y_s^2) \\
\dot{y}_s &= -\alpha x_s + y_s (R_c^2 - x_s^2 - y_s^2)
\end{align*}
\]

Set-point

\[
\theta_{Seo} = \arctan\left(\frac{y_s}{x_s}\right)
\]

\[
\alpha = \text{sign}(v_{Oi_y})
\]
Dynamic obstacles

\[
\begin{align*}
\dot{x}_s &= \alpha y_s + x_s \left( R_c^2 - x_s^2 - y_s^2 \right) \\
\dot{y}_s &= -\alpha x_s + y_s \left( R_c^2 - x_s^2 - y_s^2 \right)
\end{align*}
\]

Set Point

\[
\theta_{S_{eo}} = \arctan \left( \frac{\dot{y}_s}{\dot{x}_s} \right)
\]

\[
\alpha = \text{sign}(v_{Oi_y})
\]

\[
\text{sign} = \begin{cases} 
1 & \text{if } v_{Oi_y} \leq 0 \text{ (clockwise avoidance)} \\
-1 & \text{if } v_{Oi_y} > 0 \text{ (counterclockwise avoidance)}
\end{cases}
\]
Obstacle Avoidance

Dynamic obstacles

Problem: conflict risks to result in a divergence regarding the target to reach!!
When robots belong to the same group, then we assign the same direction of avoidance
Introduction

Modeling and Control

Obstacle Avoidance

Dynamic obstacles

\( \psi_i(d_{ij}) = \begin{cases} 
1 & d_{ij} > R_{\text{ext}} \\
\frac{(d_{ij} - R_{\text{int}_j})}{(R_{\text{ext}} - R_{\text{int}_j})} & R_{\text{int}_j} < d_{ij} \leq R_{\text{ext}} \\
0 & d_{ij} \leq R_{\text{int}_j}
\end{cases} \)

Velocity adjustment
**Obstacle Avoidance**

Toward a null risk of collision!

\[ \psi_j(d_{ij}) = \begin{cases} 
1 & \frac{d_{ij} - R_{\text{int}_j}}{R_{\text{ext}} - R_{\text{int}_j}} \\
0 & 
\end{cases} \]

\[ v_j = v_j \prod_{i=1, i \neq j}^{M} \psi_i(d_{ij}) \]

Avoiding local minima among same group

\[ |R_{\text{int}_k} - R_{\text{int}_l}| \geq \xi \]
**Introduction**

**Modeling and Control**

**Control architecture**

**Results**

**Conclusion**

---

**Attraction to Dynamical Target controller:**

\[ P_{Sat} = (xT_i, yT_i) \]

\[ \theta_{Si} = \theta_{Sat} \]

**Obstacle Avoidance controller:**

\[ P_{Soa} \text{ is always set to } (0, 0) \]

\[ \theta_{Si} = \theta_{Soa} \]
Control law

\[ v_i = v_{max} - (v_{max} - v_T) e^{-\frac{(d_{S_i}^2)}{\sigma^2}} \]

\[ \omega_i = \omega_{S_i} + k_1 \tilde{\theta}_i \]

\[ \theta_{S_i} \] is the set-point angle

\[ \tilde{\theta}_i = \theta_{S_i} - \theta_i \]

\[ \dot{\theta}_i = w_{S_i} - \omega_i \]

Consider the Lyapunov function

\[ V = \frac{1}{2} \tilde{\theta}_i^2 \]

\[ \dot{V} = k_1 \tilde{\theta}_i \dot{\theta}_i \]

\[ \dot{\theta}_i = -k_1 \tilde{\theta}_i \]

\[ \dot{V} = -k_1 \tilde{\theta}_i^2 < 0 \]

The control law is asymptotically stable

[IROS10]
Experimental results: Experimental site and robots
**Experimental results**  Simulating a possible conflict (without velocity vector projection)

![Robot trajectory in the [O, X, Y] reference](image)

**Divergence of the robots**
Experimental results  Using velocity vector projection (velocity reduction)

A) Obstacle avoidance

B) Self avoidance  (Reconfigutation)
Experimental results

Using velocity vector projection (velocity reduction)

ROBOT1

ROBOT3

Video 1

Video 2
Conclusion

Multi robot architecture: modeling, control, selection

Controller design (attraction, obstacle avoidance)
Control architecture
Automatic selection of avoidance direction
Velocity reduction

Validation
Real experimentation with 3 robots

Perspectives
Develop more sophisticated controllers
Increase the number of robot (up to 10)
Thanks for your attention

Any questions


Philippe.Martinet@irccyn.ec-nantes.fr