On Dynamic Reconfiguration of Multi-Robot Formations

Rafael Fierro

MARHES Laboratory
Oklahoma State University
School of Electrical and Computer
r fierro@okstate.edu

http://r fierro.okstate.edu

University of Oklahoma
April 21, 2003

People

MARHES Lab – OSU
• Faculty
  Rafael Fierro
• Ph.D., student
  Jose Sanchez
• Master students
  EzzAldeen Edwan
  Kirk Wesselowski
• Undergrads
  Colin Anderson
  Amanda Filbeck
  Chris Williams

http://r fierro.okstate.edu/marhes

GRASP Lab – U Penn
• Faculty
  Vijay Kumar
• Ph.D., students
  Aveek K. Das
  Peng Song
  John Spletzer

http://www.cis.upenn.edu/mars
Motivation

- Small, low cost expendable vehicles are becoming more common. They need to operate cooperatively in tactical situations;
- Many biological systems exhibit formation, coordination and group behaviors.

- Develop the theory and software tools to coordinate networks of semi-autonomous vehicles.

Applications

- Collaborative mapping and exploration
- Cooperative transport
- Force multiplication
- Relay communications
- Networks of smart mobile sensors
- Satellite clustering
- Swarming
- Formation Flight

Research Initiatives

RoboCup (Robots Playing Soccer)
USAR (Urban Search and Rescue)
Hierarchical Architecture

- **Group Control**
- **Formation Control**
- **Local Control**

Hybrid Systems and Abstractions

- **Consistent abstractions**
- **Parallel and sequential composition**
- **Refinement**

Pappas *et al.*, IEEE TRA, 2001
Example

Team trajectory

Final Formation

Initial formation

Reconfiguration

Group Control: Abstractions

Develop suitable descriptions of the group of vehicles

- Smaller dimensional system that describes team behavior,
- Preserve some properties of interest,

\[ x = \{x_i, \ldots, x_N\} \in \prod_{i=1}^N X_i \equiv X \subseteq \mathbb{R}^m \]

\[ \dot{x} = f(x, u) \]

\[ \dot{y} = g(y, \nu) \quad \text{with} \quad y \in \Gamma \subseteq \mathbb{R}^p \quad \text{and} \quad p \ll m \]

- Smooth map \( \Phi : X \rightarrow \Gamma \) and \( y = \Phi(x) \)
Optimization Based Formation Control

Control Graphs

- Graph assignment algorithm
- Stability of the switched system becomes an issue
The control graph should be assigned such that
\[ \dot{V} \leq 0 \]

Fierro et al., ICRA2001

**Graph Assignment Algorithm**

**Algorithm 1** Control graph assignment algorithm (CGA)

- Initialize adjacency matrix \( H(i, j) := 0; \)
- For all robot \( k \in \{1, 2, \ldots, n\} \), \( k \neq \text{leader} \) do
  - \( H(i, k):=1 \) for \( SB_{ik} C \), edges \((i, k) \in \) spanning tree of \( G_{\text{vis}} \);
  - \( d_k := \) depth of node \( k \) in communication graph \( G_{\text{comm}} \);
  - Find set \( P_k \) of robots visible to \( k \) with depths \( d_k, d_k - 1 \);
  - If \( P_k = \emptyset \) (disconnected) then
    - Report failure at \( k \), break;
  - \( S_k := P_k \) sorted by ascending \( \delta t_{ik} K_{ik} S_{ik}^2 \) \( (i \in P_k) \);
  - If \( \text{numOfElements}(S_k) \geq 2 \) then
    - Pick last two elements \( i, j \in P_k \);
    - If \( e_{ijk} = (l_{ik} + l_{jk} - l_{ij}) \neq 0 \) then
      - \( H(i, k) := 1, H(j, k) := 1 \) for \( SS_{ijk} C \);
    - Else
      - Repeat above check for remaining \( j \in S_k \) in order;
  - Generate set-points \( r^d \) for desired shape \( S^d \)
Control Graph Assignment Algorithm

Local (kinematic) Control

Basic Leader-Following

Obstacle Leader-Following

2 Leader-Following

Dilation Control
Theorem Assume:

- The leader’s trajectory is well-behaved

Then, the formation is stable and the system error of the linearized system converges exponentially to zero.

Remarks Are the internal dynamics stable?

\[
\dot{e}_x = \frac{v}{d} \sin e_x + \eta_1 e_x, \\
\dot{e}_y = \frac{v}{d} \cos e_x + \eta_1 e_y
\]

MPC methods have some potential advantages over Input/Output feedback linearization approaches.

- Ability of incorporate constraints,
- Optimization-based methods may be more robust and more flexible in meeting performance requirements, but may be computationally expensive,
- We develop a dual-mode MPC algorithm with a terminal constraint instead of an MPC algorithm with a terminal cost,
- The optimization-based controller (MPC) drives the system to the terminal constraint set \( X_f \),
- Within \( X_f \) the system utilizes an I/O feedback linearization control law. Thus, stability can be proven.
**Objective Function**

\[ V(k) = V_{pos}(k) + V_{in}(k) + V_{col}(k) \]

\[ V_{pos}(k) = \sum_{m=1}^{H_p} x^T(k + m)Q(k + m) \]

\[ V_{in}(k) = \sum_{m=1}^{H_p} \Delta U^T(k - 1 + m)R\Delta U(k - 1 + m) \]

\[ V_{col}(k) = \sum_{m=1}^{H_p} e^{-c_{i}(k-1+m)/\tau} \]

\[ c_{ij} = \|x_i - x_j\|_2 - r_{\text{min}} \]

---

**Feedback linearization fails**

**Dual MPC successfully controls the formation.**
Bow, Port and Starboard

- We use nautical terms to emphasize the lack of leaders and followers

![MPC of three nonholonomic robots](image)

Applications

**Coordinated manipulation:**
- Approach
- Organize
- Transport
Multimedia Version of

Formation Flight Geometry

The state vector is arranged into four sets

- \( x_1 = (V, \gamma, \chi) \)
- \( x_2 = (\mu, \alpha, \beta) \)
- \( x_3 = (p, q, r) \)
- \( x_4 = (e_x, e_y, e_z) \)

- The output vector is
  \[ z_{ij} = \begin{bmatrix} i_x & i_y & i_z & \mu_j \end{bmatrix}^T \]

where

\[
\begin{pmatrix}
i_x \\
i_y \\
i_z \\
1
\end{pmatrix} = ^i A_E \begin{pmatrix} E_x \\
E_y \\
E_z \\
1
\end{pmatrix}
\]

\[
\hat{\mu}_{ij} = \mu_i - \mu_j
\]

We adopted A. Isidori’s formulation
• Applying I/O feedback linearization via dynamic extension

\[ i_x^{(4)} = z_{1_4} = \overline{w}_1 \quad i_y^{(4)} = z_{2_4} = \overline{w}_2 \quad i_z^{(4)} = z_{3_4} = \overline{w}_3 \quad \mu_{ij} \equiv \overline{z}_{4_4} = \overline{w}_4 \]

Basic Leader-Following

\[ z_{4_{i(i+1)k}}^{(4)} = \overline{w}_{1k} \quad z_{2_{i(i+1)k}}^{(4)} = \overline{w}_{2k} \quad z_{3_{i(i+1)k}}^{(4)} = \overline{w}_{3k} \quad \mu_{i(k)} \equiv \overline{z}_{4k} = \overline{w}_{4k} \]

2- Leader-Following
Simulation Results

3. UAVs Flying in Closer Formation

Multi-Robot Experimental Testbed at OSU

- Off-the-shelf components
- Tamiya™ TXT-1 Platform

- Laptops (< 2.7 lb.)
- Integrated wireless,
- Firewire port,
- USB, etc.
- Serial servo controller
Sensors

- Stereo vision for cooperative sensing, exploration and mapping,
- IR and sonar for obstacle avoidance,
- GPS, odometer, and gyros for localization.

Multi-Robot Experimental Testbed at OSU
Conclusions and Future Work

- We describe a modular architecture for coordinating distributed multi-vehicle systems;
- We present two optimization-based control approaches;
- Work has to be done at the interfaces of sensing, communication and control;
- Learning should be integrated at all levels.
- Need for a research agenda:
  - Define metrics for cooperative control;
  - Implement and evaluate different communication architectures, and networking technologies;
  - Vision based-control of UAVs.

Relevant Publications


