Implementation of distributed consensus on an outdoor testbed

Apurva Joshi\textsuperscript{1}    Narendra Limbu\textsuperscript{3}    Indrajit Ahuja\textsuperscript{4}
Ameer K. Mulla\textsuperscript{1}    Hoam Chung\textsuperscript{2}    Debraj Chakraborty\textsuperscript{1}

\textsuperscript{1}Indian Institute of Technology Bombay, Mumbai, India
\textsuperscript{2}Monash University, Melbourne, Australia
\textsuperscript{3}IFM Engineering Pvt. Ltd., Pune, India
\textsuperscript{4}Larsen & Toubro India Ltd., India

European Control Conference, 2016
Overview

**Consensus:** all agents reach an autonomously decided common state

**Distributed:** all agents take control decisions themselves

**Contribution:** Distributed multiagent consensus algorithm

**Convergence:** theoretically provable

**Practical implementation:** outdoor, on three quadrotors
Motivation

- Quadrotors: good benchmark for testing advanced control algorithms
  - Ease of assembly, availability of components
  - Nonlinear underactuated dynamics
  - Six degrees of freedom

- State of the art:
  - Sophisticated theory for multiagent consensus (single and double integrator agents) already developed
  - Practical implementation: Indoors with motion capture cameras or with centralized control algorithms
  - COLLMOT group, Eötvös University, Hungary: distributed, empirical consensus law without proof of convergence
Table of contents

Quadrotor dynamics
   Frames of reference
   Thrust and torque generation

Control laws
   Inner loops
   Waypoint navigation
   Consensus law

Implementation
   Simulation
   Hardware
   Experiment design

Results
   Consensus video
   Position plots
   Angle correction plots

Conclusion
Quadroto r dynamics

Frames of reference:

Earth frame, \{E\}

- $x_E$— axis points in the north direction
- $y_E$— axis points in the east direction
- $z_E$— axis points in the up direction
Quadrotor dynamics

Frames of reference:

Body frame, \{B\}
- \(x_B\) — axis points towards front end
- \(y_B\) — axis points towards right end
- \(z_B\) — axis points downwards
Quadrotor dynamics

Frames of reference:

An auxiliary frame, \{V\},

- same origin as \{B\}
- \(z_V\) axis is parallel to \(z_E\) axis
- \(x_V, y_V\) axes are projections of \(x_B, y_B\) onto a plane parallel to the \(x_Ey_E\) plane in \{E\} and passing through the origin of \{V\}.
Quadrotor dynamics

- Motion along six degrees of freedom achieved by varying rotor speeds, $\bar{\omega}_i$
- Generating pairwise difference in rotor thrusts leads to rotational motion

\[
\begin{bmatrix}
T \\
\tau_x \\
\tau_y \\
\tau_z
\end{bmatrix} =
\begin{bmatrix}
-b & -b & -b & -b \\
0 & -db & 0 & db \\
db & 0 & -db & 0 \\
k & -k & k & -k
\end{bmatrix}
\begin{bmatrix}
\bar{\omega}_1^2 \\
\bar{\omega}_2^2 \\
\bar{\omega}_3^2 \\
\bar{\omega}_4^2
\end{bmatrix} =
A
\begin{bmatrix}
\bar{\omega}_1^2 \\
\bar{\omega}_2^2 \\
\bar{\omega}_3^2 \\
\bar{\omega}_4^2
\end{bmatrix}
\]

where,

- $T$ is thrust generated, $\begin{bmatrix} \tau_x & \tau_y & \tau_z \end{bmatrix}^T$ are the torques generated
- $b,k$: constants
- $d$: distance of the motor from the CoG of the quadrotor
Control loops

Figure: A block diagram of the quadrotor control loops
Waypoint navigation

- Process by which the quadrotor navigates to different positions $\mathbf{p}^E = \begin{bmatrix} p_x^E & p_y^E \end{bmatrix}^T \in \mathbb{R}^2$ in $\{E\}$
- Generate $\tau_x$ and $\tau_y$ to vary $\theta_p$ and $\theta_r$ and thus maneuver the quadrotor
- Keep $\theta_y$ constant using heading control loop.
Waypoint navigation

- Consider the frame \{V\}. To accelerate along \(x_V\) and \(y_V\) axes, we need to generate forces

\[
\begin{align*}
  f_x^V &= T \sin \theta_p \approx T \theta_p \\
  f_y^V &= T \sin \theta_r \cos \theta_p \approx T \theta_r.
\end{align*}
\]  

for small \(\theta_x\) and \(\theta_y\)

- We control the motion using a PD law

\[
f^V = m K_f [K_p (p^V - p^V) - v^V]
\]

where \(f^V = \begin{bmatrix} f_x^V & f_y^V \end{bmatrix}^T\)

- Then, desired values of angles, \(\Theta^* = \begin{bmatrix} \theta^*_p & \theta^*_r \end{bmatrix} \in \mathbb{R}^2\) are

\[
\Theta^* = \frac{m K_f}{T} [K_p (p^E - p^E) - v^E]
\]
Waypoint navigation

- To attain $\Theta^* = \left[ \begin{array}{c} \theta_p^* \\ \theta_r^* \end{array} \right] \in \mathbb{R}^2$, generate torques

$$\mathbf{\Gamma} = \left[ \begin{array}{c} \tau_x \\ \tau_y \end{array} \right]^T \in \mathbb{R}^2$$

using a PD controller

$$\mathbf{\Gamma} = K_{pr,p} (\Theta^* - \Theta) + K_{dr,p} (\dot{\Theta}^* - \dot{\Theta}) \quad (6)$$

where $K_{pr,p} = \left[ \begin{array}{cc} K_{pr} & K_{pp} \end{array} \right]^T \in \mathbb{R}^2$ and

$K_{dr,p} = \left[ \begin{array}{cc} K_{dr} & K_{dp} \end{array} \right]^T \in \mathbb{R}^2$ are the control gains.

- Controller designed such that $\theta_p \to \theta_p^*$ and $\theta_r \to \theta_r^*$ almost immediately
Waypoint navigation

- In the \( \{E\} \) frame,

\[
f^E = R_V^E f^V
\]  

(7)

and

\[
f_x^V = T \sin \theta_p \approx T \theta_p \tag{8}
\]

\[
f_y^V = T \sin \theta_r \cos \theta_p \approx T \theta_r. \tag{9}
\]

for small \( \theta_x \) and \( \theta_y \)

- If we can vary \( \theta_p \) and \( \theta_r \) independently and instantaneously, then motion in the \( x_E y_E \) plane can be modelled as a double integrator.
Waypoint navigation

- We vary $\theta_p$ and $\theta_r$ independently and quickly such that change in angle is much faster than translational motion
- As $\theta_p$ and $\theta_r$ change, the vertical component of $T$ reduces by a factor of the cosine of $\theta_p$ and $\theta_r$
- But $\theta_p$ and $\theta_r$ are small and altitude control loop is fast

Hence the quadrotor can be modelled as a double integrator

$$\dot{p}^E = v^E, \quad \dot{v}^E = f^E$$

(10)

where

- $p^E = \begin{bmatrix} p_x^E & p_y^E \end{bmatrix}^T \in \mathbb{R}^2$ is the position in $\{E\}$
- $v^E = \begin{bmatrix} v_x^E & v_y^E \end{bmatrix}^T \in \mathbb{R}^2$ is the velocity in $\{E\}$
- $f^E = \begin{bmatrix} f_x^E & f_y^E \end{bmatrix}^T \in \mathbb{R}^2$ is the acceleration input in $\{E\}$
Consensus law

- If, for all $p^E_i(0)$ and $v^E_i(0)$ and all $i, j = 1, ..., n$, $\|p^E_i(t) - p^E_j(t)\| \to 0$ and $v^E_i \to 0$ as $t \to \infty$ then consensus achieved

- Information exchange modelled as undirected graph $G_n := (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{1, ..., n\}$ is the set of nodes and $\mathcal{E} \subseteq (\mathcal{V} \times \mathcal{V})$ is the set of edges

- Node $\equiv$ quadrotor, edge $\equiv$ available communication channel

- Set of neighbours, $\mathcal{N}_i := \{j \in \mathcal{V} : (i, j) \in \mathcal{E}\}$.

- Laplacian matrix $\mathcal{L}_n$ of a graph $G_n$ is given by
  $\mathcal{L}_n = [l_{ij}] \in \mathbb{R}^{n \times n}; l_{ij} = -a_{ij}, i \neq j, l_{ii} = \sum_{j=1}^{n}a_{ij}$. 

Consensus law

We propose the following consensus law for the $n$ quadrotors, where each quadrotor is modelled as a double integrator

$$ f_i^E = \sum_{j \in \mathcal{N}_i} a_{ij} (p_j^E - p_i^E) - \beta v_i^E, \quad i = 1, \ldots, n $$  \hspace{1cm} (11)

where

- $a_{ij}$ is the $(i,j)$ entry of the adjacency matrix $A_n \in \mathbb{R}^{n \times n}$ corresponding to the communication graph, $\mathcal{G}_n$
- $\beta$ is a positive constant

Implementation

Simulation
Hardware
Experiment design
Results
Consensus video
Position plots
Angle correction plots
Conclusion
Consensus law

Theorem
Given a system

\[ \dot{p}^E = v^E, \quad \dot{v}^E = f^E \] (12)

The control law \(^1\),

\[ f^E_i = \sum_{j \in \mathcal{N}_i} a_{ij} (p^E_j - p^E_i) - \beta v^E_i, \quad i = 1, \ldots, n \]

achieves consensus asymptotically iff \( \mathcal{G}_n \) is connected

As a result

\( \triangleright \) \( p(t) \to (\beta 1_n 1_n^T \otimes l_2) p(0) + (1_n 1_n^T \otimes l_2) v(0) \)

\( \triangleright \) \( v(t) \to 0 \) as \( t \to \infty \)

Hence \( \|p_i^E(t) - p_j^E(t)\| \to 0 \) and \( v_i^E \to 0 \) as \( t \to \infty \) for all \( i, j = 1, \ldots, n \)

\(^1\) Proof similar to W. Ren, R. Beard, Distributed consensus in multi-vehicle cooperative control, Springer, 2008
Simulation

- Quadrotor dynamics modeled as double-integrator dynamics decoupled along two principle axes
- Proposed consensus law first tested on a multi-agent system simulation platform with double integrator agents to predict the behaviour of the actual quadrotor system
## Hardware

<table>
<thead>
<tr>
<th>Component</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frame</td>
<td>Hobbyking X550</td>
</tr>
<tr>
<td>Motors</td>
<td>Turnigy Park 480 Outrunner 1320 kv BLDC</td>
</tr>
<tr>
<td>Motor driver</td>
<td>AE-30A Brushless ESC</td>
</tr>
<tr>
<td>Processor</td>
<td>APM 2.6 with ATmega 2560 microcontroller</td>
</tr>
<tr>
<td>Barometer</td>
<td>MS5611</td>
</tr>
<tr>
<td>IMU</td>
<td>MPU 6000 Gyroscope + Accelerometer</td>
</tr>
<tr>
<td>GPS</td>
<td>3DR uBlox GPS with Compass</td>
</tr>
<tr>
<td>Communication</td>
<td>XBee Pro S1, XBee S8</td>
</tr>
<tr>
<td>Power</td>
<td>LiPo 5000 mAh, 3S 50C, 11.1V battery</td>
</tr>
</tbody>
</table>

**Table:** List of hardware components used
Experiment design

- Each quadrotor broadcasts its position information,
  \[ p^E = \begin{bmatrix} p_x^E \\ p_y^E \end{bmatrix}^T \]
  which is measured by the on-board GPS receiver.
- Position accuracy of the GPS receiver is 2.5m CEP.
- Thus, for practical reasons, we say that the quadrotors reach consensus if
  \[ \|p_i^E(t) - p_j^E(t)\| \leq 2.5m \]
  for all \( i, j = 1, 2, 3 \).
Implementation of distributed consensus on an outdoor testbed
Joshi, Limbu, Ahuja, Mulla, Chung, Chakraborty

Quadrotor dynamics
Frames of reference
Thrust and torque generation

Control laws
Inner loops
Waypoint navigation
Consensus law

Implementation
Simulation
Hardware
Experiment design

Results
Consensus video
Position plots
Angle correction plots

Conclusion
Position plots

Latitude and Longitude tracking

[Graphs showing position plots for Latitude and Longitude tracking]
Angle correction plots

Roll and pitch angle correction

Frames updated at 10 Hz

Roll Angles (in degrees)

Frames updated at 10 Hz

Pitch angles (in degrees)
Conclusion

Proposed and implemented a decentralized consensus law wherein

- On-board controllers take navigation decisions by communication with its neighbours $\implies$ decentralized!
- Justification for approximating the quadrotor as two independent double integrators acting along the $x$ and $y-$ axes of motion
- Outdoor environment $\implies$ inherent GPS errors. However, the quadrotors still successfully managed to reach consensus.
Thank you :) 

Questions?