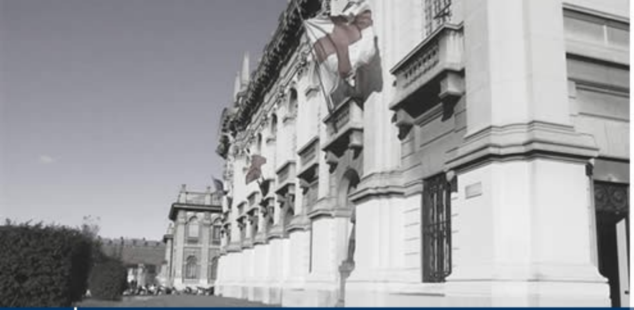


POLITECNICO DI MILANO

Object-oriented modelling of multi-rotor vehicle dynamics

Metodologie ICT per i Sistemi di Controllo – A.A. 2013/2014

March 26th, 2014

Marcello Pogliani – Davide Tateo

Objectives

This project

Design and implementation of basic components
for the particular case of fixed pitch (variable rpm) quadrotor

Tool: OpenModelica [1]

In particular:

- Modelica Multibody library (basic rigid body dynamics, gears, etc)
- Modelica Electrical library for motors
- Simple models for rotor force/torque generation mechanisms [2]
- Existing implementations for control algorithms (PI control)

Approach

1. Familiarisation with the system

2. Hierarchical breakdown into subsystems and definition of interfaces

3. Development of models for individual components

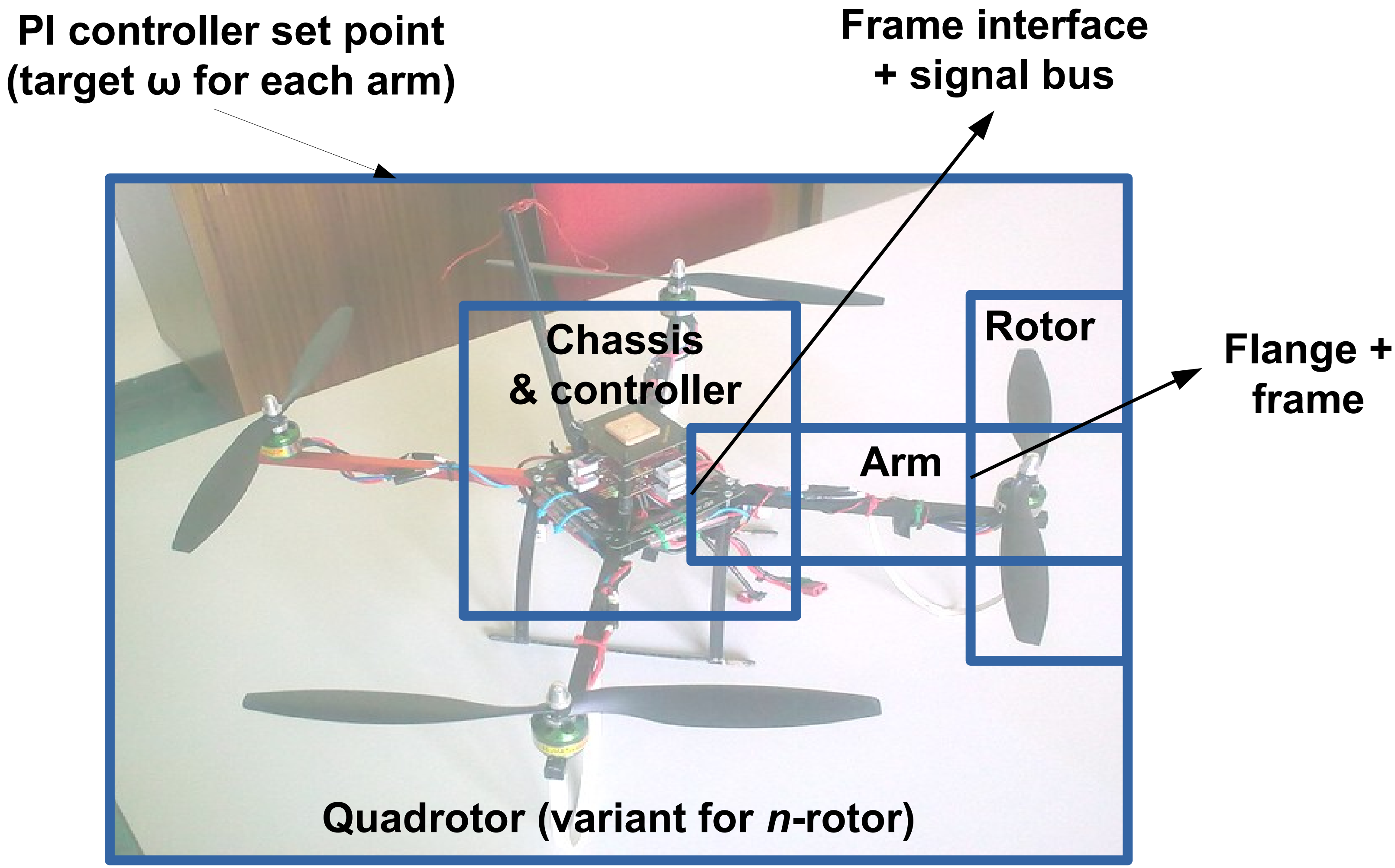
4. Development and testing of simulator

Familiarisation with the system

Main components

- Structure (metallic tubes)
- Power system (battery, power electronics)
- On-board computer (including sensors)
- Motor drivers
- Motors
- Propellers

Hierarchical breakdown into subsystems and definition of interfaces



The `Multirotor` package

- Multirotor
  - Examples
    - TestQuadrotor
    - TestNRotor
    - TestYaw
    - TestRollPitch
    - Test
  - Basics
    - Rotor
    - Quadrotor
    - Controller
    - ChassisNRotor
    - Chassis
    - Bus
    - Arm
    - PartialArm
    - NRotor

Models for individual components

Rotor

Leverages a simple model from [2] to compute lift and drag

$$L = \alpha \cdot bl \cdot \omega^2$$

$$Tl = sign(\omega) \cdot \left(bd_1 \omega^2 + bd_2 \omega^2 \alpha^2 + bd_3 \cdot |\omega| \cdot \alpha \right)$$

Incorporates the model of an inertia (from the Modelica standard library)

Interfaces:

- **Frame** to the rigid body of the arm (transmit the lift)
- **Flange** to the motor (transmit the torque)

Parameters:
α = 10°
bd[] = {9.96e-09,2.46e-10,4.33e-07}
bl = 3.88e-07
Jr = 3.5e-05 kg · m^2 (rotor inertia)

Models for individual components

`PartialArm`

- Rigid body (from the Multibody library)
- Gear (ideal)
- **`Bus`** connection: speed sensor signal [rad/s] + control signal [V]
- Signal-controlled voltage generator

`Arm` extends **`PartialArm`** with:

- Connection to a **`DC_PermanentMagnet`** motor
- Connection to a Rotor model (see previous slide)

Arm (rigid body):
length = 0.2 m
diameter = 0.01 m
mass = 0.18 kg

Gear:
gearRatio = 1

Motor:
motorVoltage = 12 V
motorResistance = 0.3 ohm
motorInductance = 0.0015 H
motorVelocity = 1100 rad/s
motorInertia = 3.5e-05 kg · m^2
rotorInertia = 3.5e-05 kg · m^2

Models for individual components

`Chassis` / **`ChassisNRotor`**

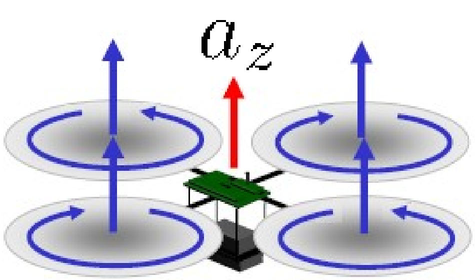
- Rigid body to model the weight of the central support + microcontroller + …
- Connect all the arms through **`Frame`**s, taking into account the correct orientation (using a **`FixedRotation`** component)

`Controller`

- A set of PI controllers, one per arm
- Directly connected to the arms through **Bus** connections
- Set point: the desired angular speed

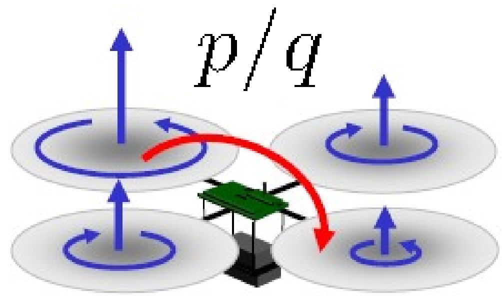
Development and testing of the simulator

Developed simulations (package **`Examples`**):



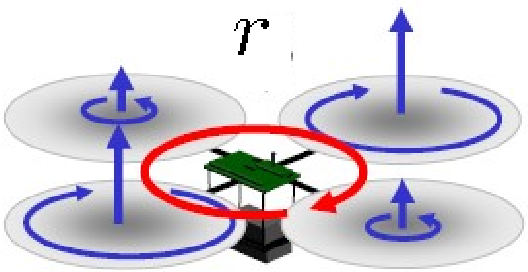
TestQuadrotor

(U_{col})



TestRollPitch

(U_{lon}/U_{lat})



TestYaw

(U_{ped})

Notice*: the set point is the angular velocity of each arm; it is easy to attach a more complex control system to the simulator in order to provide a more significant input (set point) in terms of roll\pitch\yaw*

References

[1] *OpenModelica*, https://www.openmodelica.org/

[2] M. J. Cutler, *Design and Control of an Autonomous Variable-Pitch Quadrotor Helicopter*, Master's Thesis, Massachusets Institute of Technology, September 2012

[3] P. Fritzson et al., *OpenModelica Users Guide*, March 2014

[4] P. Fritzson and O. Rogovchenko, *Introduction to Object-Oriented Modeling, Simulation and Control with Modelica*, Tutorial at MODPROD, February 2012

[5] *Modelica® - A Unified Object-Oriented Language for Physical Systems Modeling*, Language Specification, Version 3.2, March 2010