Quad-Rotor Flight Path Energy Optimization

Edward Kemper

Northern Arizona University

edwardkemper@gmail.com

April 29, 2014

NORTHERN ARIZONA UNIVERSITY

Advisor : Niranjan Venkatraman

Overview

Introduction

- Tremendous Development
- Private sector, not just military
- Autonomy
- FAA policy for commercial applications (2015)
- Rapid growth of a multi-billion dollar industry

Motivation

- Energy management is a pervasive engineering problem
- Quad-rotors have very high energy demand
- Multi-rotor systems are entirely thrust driven
- Instability = Maneuverability = High energy

Prior Work

- energy optimization and trajectory planning of fixed wing UAVs
- quad-rotors
  - basic stability
  - attitude and position control
  - dynamical model
- Classical Optimal Control is a long story

Problem Statement

We wish to find a set of control expressions for a quad-rotor UAV which minimizes the energy expended in flying between two known points by optimizing the path.

Problem Statement

Assumptions:

- the flight path that will be optimized is free of obstacles
- only modeled environmental variables
- model of the system derived from a Euler-Lagrange formulation

Path-Energy Optimization

Edward Kemper

Introduction

Dynamic Model

Classical Optimal Control

PID Control

PID Gain Optimization

Conclusions

Problem Statement

Classical Optimal Control Approach:

- control of the system and the optimization are represented in a single mathematical formulation
- Solving the optimal control problem is achieved by solving a boundary value problem
- Most literature deals with *linear* systems

Problem Statement

Heuristic approach:

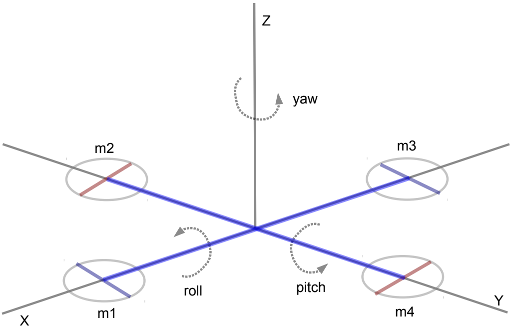
- PD gives attitude control
- PID gives position control
- this provides a platform for simulation
- the optimization procedure evaluates the results of these simulations for optimality as a function of the PID gains used in the position control expressions

Dynamic Model

- We must understand the mathematical relationships between the control input and the resulting dynamics of the system
- Euler-Lagrange formulation

Dynamic Model

- ψ is the yaw angle around the z-axis
- θ is the pitch angle around the y-axis
- ϕ is the roll angle around the x-axis



Dynamic Model

The rotor angular velocities are related to the forces they produce by:

$$f_i = k\omega_i^2 \tag{1}$$

Path-Energy Optimization
Edward Kemper
Introduction
Dynamic Model
Classical Optimal Control
PID Control
PID Gain Optimization
Conclusions

Dynamic Model

In the quad-rotor frame of reference, the motors produce torques on the system.

$$\boldsymbol{\tau}_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} lk(-\omega_2^2 + \omega_4^2) \\ lk(-\omega_1^2 + \omega_3^2) \\ \sum_{i=1}^{4} b\omega^2 \end{bmatrix} \tag{2}$$

Dynamic Model

The combined thrust of the rotors in the direction of the quad-rotor frame z axis is $T_B = [0, 0, T]^T$ where,

$$T_B = \sum_{i=1}^{4} f_i \tag{3}$$

.

Dynamic Model

In the inertial frame, the kinetic and potential energy of the system are given by

$$T_{\text{trans}} = \frac{1}{2} m \dot{\xi}^T \dot{\xi} \tag{4}$$

$$T_{\text{rot}} = \frac{1}{2} \dot{\eta}^T J \dot{\eta} \tag{5}$$

$$U = mgz. \tag{6}$$

The Lagrangian is formed as the difference between kinetic and potential energy.

Dynamic Model

The dynamics of the system are represented by the Euler - Lagrange differential equations of motion.

$$\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q}}\right) - \frac{\delta L}{\delta q} = F \tag{7}$$

$$q = [x, y, z, \psi, \theta, \phi]^T = [\xi, \eta]^T. \tag{8}$$

$$\xi = [x, y, z]^T \ , \ \eta = [\psi, \theta, \phi]^T \tag{9}$$

Dynamic Model

The linear components of the generalized forces produce the following equations.

$$f = RT_B = m\ddot{\xi} - G \tag{10}$$

The angular components are expressed as

$$\ddot{\eta} = J^{-1}\big(\tau_b - C(\eta, \dot{\eta})\dot{\eta}\big) \tag{11}$$

Dynamic Model

A complete mathematical representation of the quad-rotor is as follows.

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} + \frac{T}{m} \begin{pmatrix} c_\psi s_\theta c_\phi + s_\psi s_\phi \\ s_\psi s_\theta c_\phi - c_\psi s_\phi \\ c_\theta c_\phi \end{pmatrix} \tag{12}$$

$$\begin{pmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = J^{-1} \left[\begin{pmatrix} lk(-\omega_2^2 + \omega_4^2) \\ lk(-\omega_1^2 + \omega_3^2) \\ \sum_{i=1}^{4} b\omega_i^2 \end{pmatrix} - C \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \right] \tag{13}$$

Classical Optimal Control

We Define:

- Performance index (Lagrangian) $L[q(t), u(t), t] = u^T I u$
- Constraint equations are the system equations of motion
- The Hamiltonian $H = L(q(t), u(t), t) + \lambda(t)^T \left(F(q(t), u(t), t)\right)$
- The full objective function can be written as

$$\mathcal{J} = \nu^T \Psi(q(t_f), t_f) + \int_{t_0}^{t_f} H(q(t), u(t), t) - \lambda^T \ddot{q} dt \qquad (14)$$

Classical Optimal Control

The first variation in $\mathcal{J}$ is given by

$$\delta\mathcal{J} = \frac{\partial\mathcal{J}}{\partial q}\delta q + \frac{\partial\mathcal{J}}{\partial\dot{q}}\delta\dot{q} + \frac{\partial\mathcal{J}}{\partial u}\delta u = 0 \tag{15}$$

Classical Optimal Control

By setting the variation of J equal to zero we obtain the Co-state equations:

$$\frac{\partial H}{\partial q} = \ddot{\lambda} \tag{16}$$

$$\ddot{\lambda} = (\frac{\partial L}{\partial q})^T + (\frac{\partial F}{\partial q})^T \lambda \tag{17}$$

Note : this is a second order system...

Classical Optimal Control

and the Stationarity Conditions:

$$\frac{\partial H}{\partial u} = 0 \tag{18}$$

$$\frac{\partial L}{\partial u} + \left(\frac{\partial F}{\partial u}\right)^T \lambda = 0 \tag{19}$$

Classical Optimal Control

and Secondary algebraic Co-state conditions

$$\frac{\partial H}{\partial \dot{q}} = 0 \tag{20}$$

$$(\frac{\partial F}{\partial \dot{q}})^T \lambda = 0 \tag{21}$$

Classical Optimal Control

and Terminal Boundary conditions:

$$\nu^T \frac{\partial \Psi}{\partial q}|_{t_f} + \dot{\lambda}(t_f)^T = 0 \tag{22}$$

$$\nu^T \frac{\partial \Psi}{\partial \dot{q}}|_{t_f} - \lambda(t_f)^T = 0 \tag{23}$$

and Initial Co-state conditions

$$(\lambda^T \delta \dot{q} - \dot{\lambda}^T \delta q)|_{t_0} = 0 \tag{24}$$

$$\lambda(t_0)^T \delta \dot{q} = \dot{\lambda}(t_0)^T \delta q) \tag{25}$$

Classical Optimal Control

The optimality conditions form a two-point boundary value problem in 18 coupled, nonlinear, partial differential equations!!
to obtain a solution:

- the shooting method
- finite difference method

Since there are six state variables

The Shooting Method

the algorithm:

- solving the set of differential equations as an initial value problem
- measuring the error in the final state of the system compared to the desired final state
- Advantages
  - straightforward iterative quadrature method and error minimization
- Disadvantages
  - does not always converge, subject to the stability of the differential equations in question

The Finite Difference Method

- create a system of algebraic equations at each instance in time where the solution is desired
- derivatives in the differential equations are expressed as finite differences
- The values of each state and co-state variable are defined as unknowns at each time step
- thousands of equations and unknowns

- Advantages
  - turns the BVP into a system of algebraic equations
  - easy to solve for linear system
- Disadvantages
  - hard to solve for nonlinear system
  - does not always converge

Decision Time

The BVP takes too long to solve so we must find another way!

A Heuristic Method

- Control of the system achieved with PID expressions
- Optimization achieved by appropriately manipulating PID gains in order to change the system behavior
- Quantify system behavior with performance metrics

PID Control

- PID controllers for the x, y, and z directions
- PD controllers for each of the Euler angles (ϕ, θ, ψ)
- Assume process noise and measurement noise are zero

PID Control

PID Control Algorithm

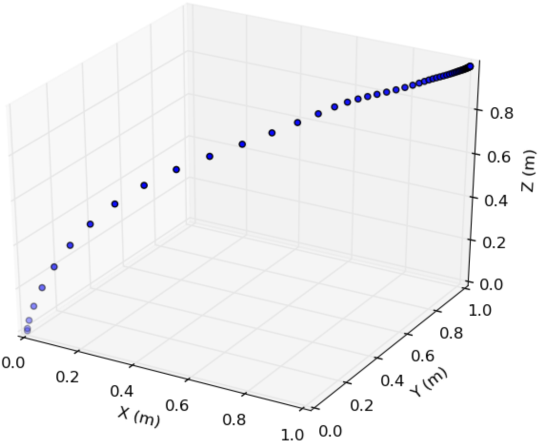
1. The position control expressions give 'commanded' linear accelerations
2. The necessary total thrust, pitch, and roll are determined.
3. The commanded torques are given by PD controllers using the commanded yaw, pitch, and roll as angular set points.
4. The motor speeds can then be determined.
5. The system model can be used to obtain the updated state of the system.
6. Repeat.

Simulation Parameters

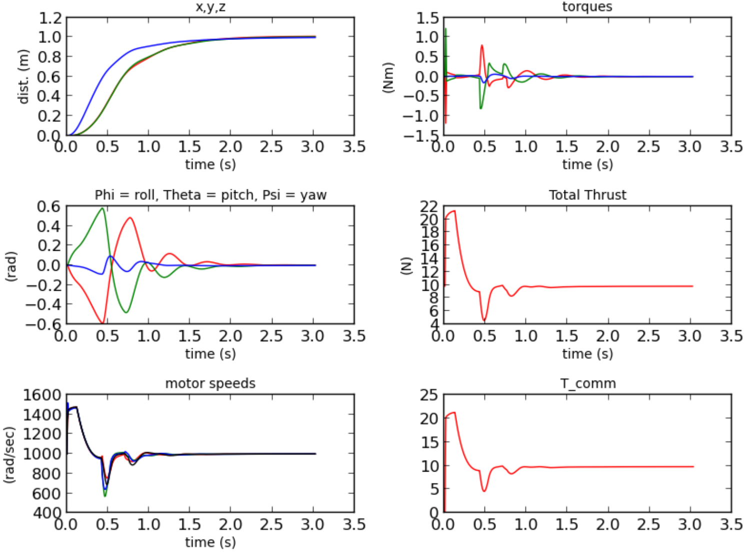
$g = -9.81$	m/s^2	acceleration due to gravity
$m = 1$	kg	mass
$L = 1$	m	length of quad-rotor arm
$b = 10^{-6}$	Nms^2/Rad^2	aerodynamic torque coef
$k = 2.45 * 10^{-6}$	Ns^2/Rad^2	aerodynamic thrust coef
$Ixx = 5.0 * 10^{-3}$	Nms^2/Rad	moments of inertia
$Iyy = 5.0 * 10^{-3}$	""	
$Izz = 10.0 * 10^{-3}$	""	

Table: Simulation Parameters

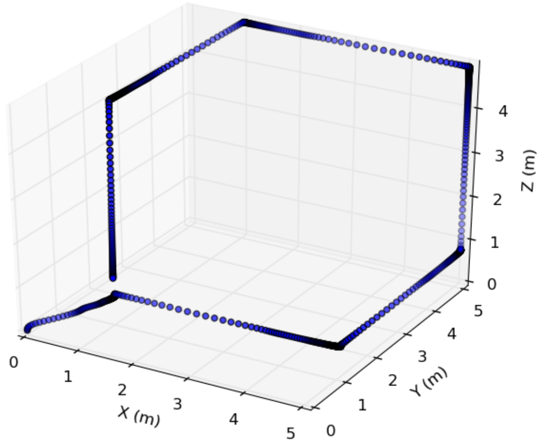
Arbitrary Sub-optimal Paths



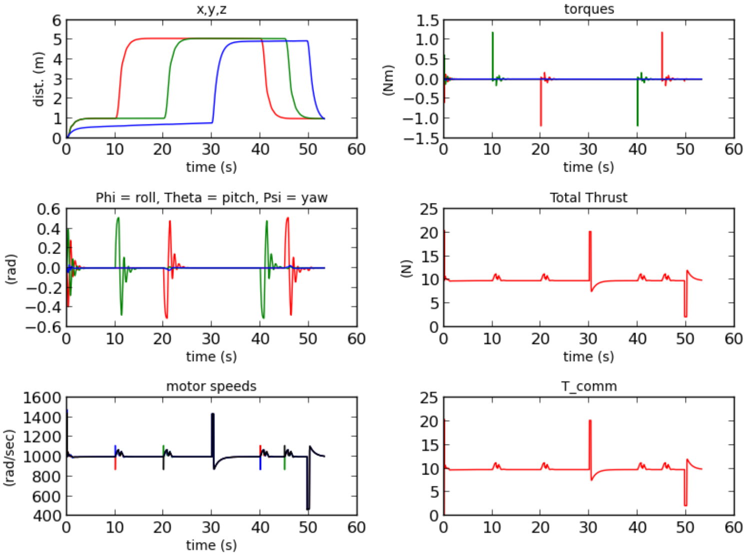
Arbitrary Sub-optimal Paths



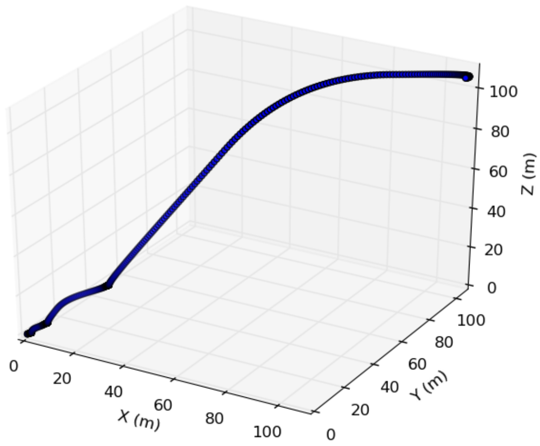
Arbitrary Sub-optimal Paths



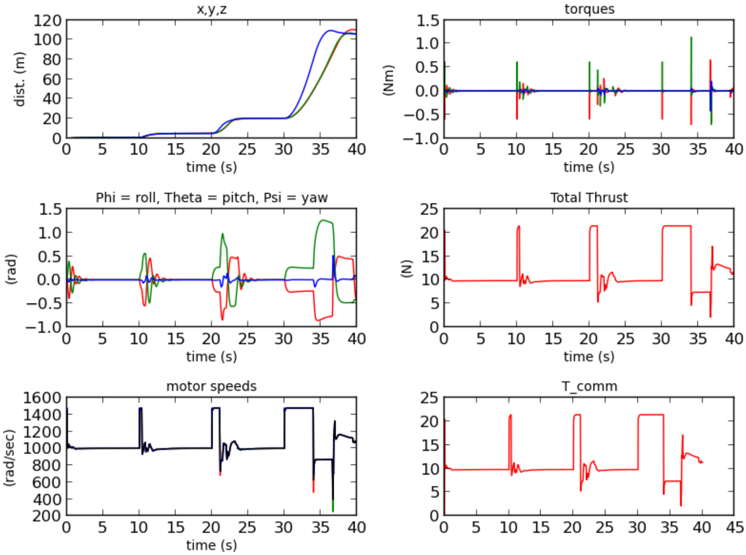
Arbitrary Sub-optimal Paths



Arbitrary Sub-optimal Paths



Arbitrary Sub-optimal Paths



Our Heuristic Optimization Method

- We aim to find: $argmin\left[\sum_{k,i}\omega_i[k] \mid K_p, K_i, K_d\right]$
- $\omega_i[k]$ is the *ith* rotor speed at the *kth* time step
- The variables K_p , K_i and K_d are the vectors of proportional, integral, and derivative gains respectively

Heuristic Method

In reality, there are other important performance criteria

- over-shoot of the desired location
- the time of flight
- marginal instabilities

Heuristic Method

An Algorithm for Optimization

1. Choose a set of proportional and derivative gains for each vector direction (x,y, and,z),
2. Perform a simulation that controls the quad-rotor from an initial vector position to a desired vector position
3. Calculate the sum of the four motor speeds over the duration of the simulation
4. Appropriately change the PID gains such that the sum of the motor speeds decreases
5. Go to step 1. Repeat until the sum of the motor speeds is found to be a minimum.

Heuristic Method

The relationship between the measured total thrust and PID gains is not well behaved!!

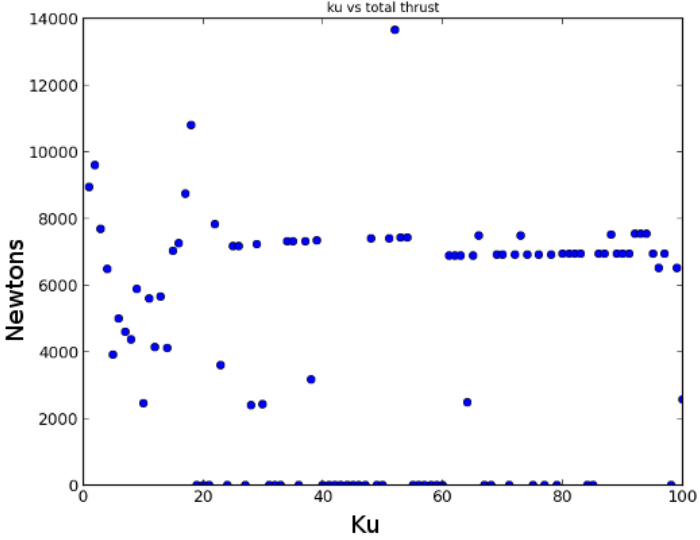
- NO steepest descent
- Needed a deeper understanding of the relationship between PID gains and performance metrics
- Try brute force approach : manually create a look-up table of PID gains v.s. measured metrics

Heuristic Method

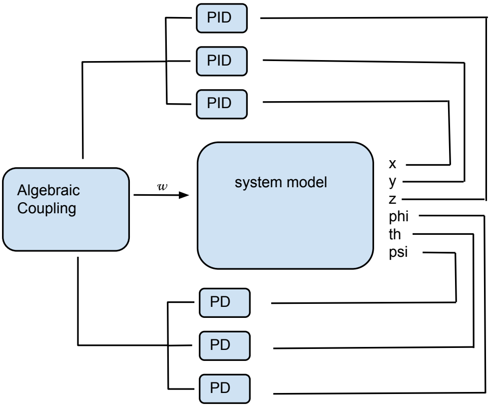
first attempt at creating a lookup table:

- Need to reduce the number of gain variables in the procedure
- Use the set point $[0, 0, 1]$ so that x and y controllers are irrelevant
- Ziegler-Nichols PID tuning method allows for PID gains to be expressed as a function of k_u

First look-up table results



Control Block Diagram



Brute-Force Implementation

- perform roughly 6000 simulations
- limit range and granularity of gain variation
- parse results for optimal run according to performance metrics

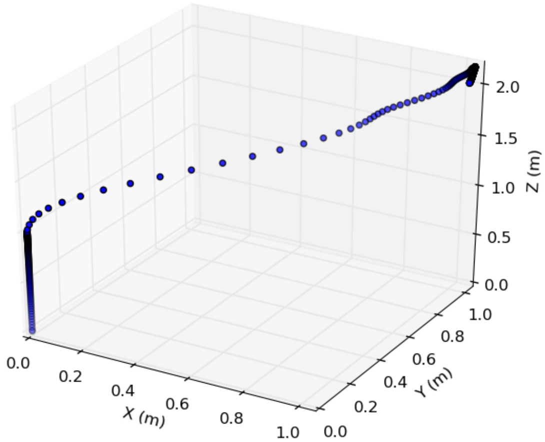
Path-Energy Optimization
Edward Kemper
Introduction
Dynamic Model
Classical Optimal Control
PID Control
PID Gain Optimization
Conclusions

Brute-Force Results

The Optimal Run	
kpx	15
kpy	15
kpz	40
kix	0.8
kiy	0.8
kiz	15
kdx	10
kdy	10
kdz	50
ending iteration	987
flight time	9.87 (s)
return value	1 (great success)
initial position	[0, 0, 1] (m)
set point	[1, 1, 2] (m)
total thrust	4969.8 (Newton seconds)
x crossings	3
x overshoot	0.0249 (m)
y crossings	1
y overshoot	0.0185 (m)
z crossings	1
z overshoot	0.0992 (m)

Brute-Force Results

The Optimal Run



Brute-Force Results

A general look-up table

- Only need to account for a small number of set points , (0,0,1),(0,1,0),(0,1,1)...
- Arbitrary optimal paths can be composed of granular optimal paths (further work)
- This approach comes closer to a real time optimization

Conclusions

- Classical Optimal Control requires too much computation
- The Heuristic Method is not viable
- Brute Force is acceptable

Further Work

- nonlinear control
- control / optimization of swarms
- sensor fusion and state estimation

Questions?