Trajectory Optimization for an Inertial Cube

Ethan Weber
6.832 (Underactuated Robotics) Final Project
Massachusetts Institute of Technology
Cambridge, MA
ejweber@mit.edu

***Abstract*—Inertially-actuated robots pose interesting problems for underactuated control. Here we present a trajectory opimization method to control a 2d cube with a torque-limited flywheel mounted inside. Inspired by earlier work done with the Cubli [2] and M-Blocks [3], we use contact-implicit trajectory optimization to generate trajectories for the inerial cube. We use one actuator (the flywheel) to control the 8 states of the robot over time. Using Sparse Nonlinear OPTimizer (SNOPT) in Drake, the interial cube can be descriped as a floating body with contact implicit movements via linear complementarity contraints. This leads to elegant optimal trajectory control for the interial cube.**

***Index Terms*—trajectory optimization, underactuated, inertia, contact-implicit**

I. Introduction

The inertial cube is particularly interesting both due to its extreme underactuation and its simplicity. After seeing the potential of M-Blocks [3], it's clear that inertially-actuated cubes have potential to be used in a variety of applications. The posibilities of movements are great, as seen in this video https://www.youtube.com/watch?v=Nns0qzd8Noo. The cubli also presents very nice control in stabilization to resist movement https://www.youtube.com/watch?v=n_6p-1J551Y. By combining prior work with ideas from optimal control and trajectory optimization, the inertial cube has potential for much more complex behavior. With elegant algorithms, we may be able to help robots explore extraterrestrial space [6] or create configurable robots like the nanobots in the movie *Big Hero 6*. We are interested in applying our best underactuated control algorithms to the system.

II. Defining the Model

In this section, we describe the interial cube in floating-body coordinates with a few simplifying assumptions. The dynamics work out nicely when described in this way. We can then proceed to optimize over the state trajectory, input torque, and external contact forces over time.

A. State Space Explanation

Fig. 1 is a diagram depicting the states of the cube. The cube is defined in floating-coordinates, meaning there is no notion of the ground in the state description. However, there are a 4 contact vectors, 1 for each corner. These are described in the figure.

Following from the cube diagram, the full states can be described in the following way.

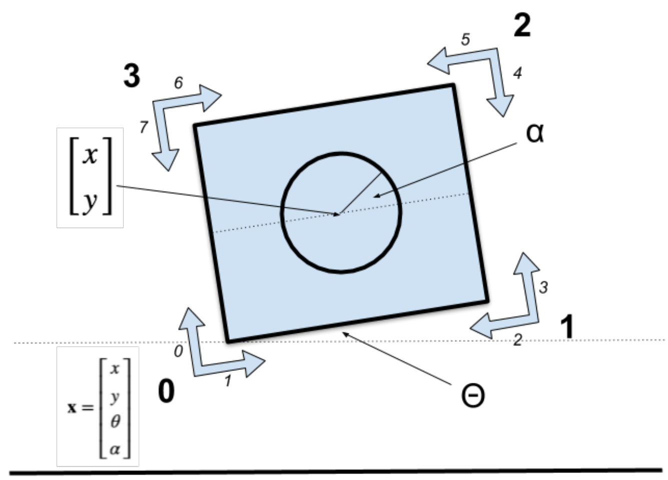

Fig. 1. The states of the floating inertial cube. The derivates of these states are also included in the total state vector $\mathbf{x}$ but are not shown here for convenience.

$$\mathbf{x} = \begin{bmatrix} x & y & \theta & \alpha & \dot{x} & \dot{y} & \dot{\theta} & \dot{\alpha} \end{bmatrix}^T \tag{1}$$

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{\theta} & \dot{\alpha} & \ddot{x} & \ddot{y} & \ddot{\theta} & \ddot{\alpha} \end{bmatrix}^T \tag{2}$$

These state vector $\mathbf{x}$ includes both the positions, angles, and their velocities. This is standard state-space notation. Following from Fig. 1, the dynamics are the following. These dynamics are informed by prior work done at Chalmers University of Technology [4] but modified for this state-space description and free-body coordinates.

$$\ddot{x} = (f_1 - f_2 + f_6 - f_5)\cos\theta - (f_0 + f_3 - f_4 - f_7)\sin\theta \tag{3}$$

$$\ddot{y} = (f_1 - f_2 + f_6 - f_5)\sin\theta + (f_0 + f_3 - f_4 - f_7)\cos\theta - g \tag{4}$$

$$\ddot{\theta} = \frac{-u + b_w\dot{\alpha} - b_c\dot{\theta}}{I_c} + \frac{1}{2}\Big(\sum_{n\in 1,3,5,7} f_n - \sum_{n\in 0,2,4,6} f_n\Big) \tag{5}$$

$$\ddot{\alpha} = \frac{u(I_c + I_w) + b_c I_w \dot{\theta} - b_w \frac{I_c + I_w}{2}\dot{\alpha}}{I_w I_c} \tag{6}$$

Here we explain the notation used in the dynamics and the few simplifying assumptions made.

- f_n is the force acting on the corner based on Fig. 1.

- u is the torque on the interial wheel.
- The cube is unit sized, meaning its dimensions are 1 for each edge length.
- b_w and b_c are the friction coeffients for the wheel and cube respectively.
- I_w and I_c are the moments of inertia for the wheel and cube respectively.
- The mass of the cube and wheel are each 0.5 kg.

In this paper, the static values are the following. We chose not to spend much time on choosing accurate values. The purpose of this project is to use contact-implicit trajectory optimization, which is described in the next section.

- $b_w = b_c = 0.5$
- $I_w = I_c = 0.5$
- $g = 9.81$

III. Optimization Formulation

In this section, we explain the necessary nonlinear program formulation to solve for a cube trajectory through space. We'll start by explaining this in the context of a swing up of the cube. This motion involves contact with the ground, which is why we are choosing do use contact-implicit trajectory optimization to avoid dealing with many modes of the system [1].

A. The Swing-Up Formulation

The swing-up is performed by getting the cube to stand on exactly one corner in a stable position. By using the contact-implicit trajectory optimization, this is possible. We formulate the optimization problem with direct transcription and a complimentarity contraint to keep the cube above the ground.

$$\begin{aligned}
&\underset{x[0:N],u[0:N],f[0:N]}{\text{find}} && \\
&\text{subject to} && x[n+1] = x[n] + f(x[n], u[n], f[n])dt, \\
& && n \in [0, N-1], \\
& && -u_{max} < u < u_{max}, \\
& && 0 < f[n][:] < f_{max}, \\
& && c[n][:] \cdot \phi[:] = 0.0,
\end{aligned}$$

Here is a description of the values used in the optimization.

- $f(x[n], u[n], f[n])$ is $\dot{\boldsymbol{x}}$.
- dt is the timestep.
- N is the number of knot points, which will implicitly define the timestep.
- $0 < f[n][:] < f_{max}$ to make sure contact forces only push away from the ground
- $c[n][i]$ is an intermediate value describing if there is a force on the ith corner. $c[n][0] = f[n][0] * f[n][1]$
  $c[n][1] = f[n][2] * f[n][3]$
  $c[n][2] = f[n][4] * f[n][5]$
  $c[n][3] = f[n][6] * f[n][7]$
- $\phi[i]$ is the distance from the ith corner to the ground

.

B. Swing-Up Results

The computed trajectory values are shown in Fig. 2, 3, and 4. This computed result is formated without a cost function. Rather, we are looking for a solution the satisfied all of our constraints.

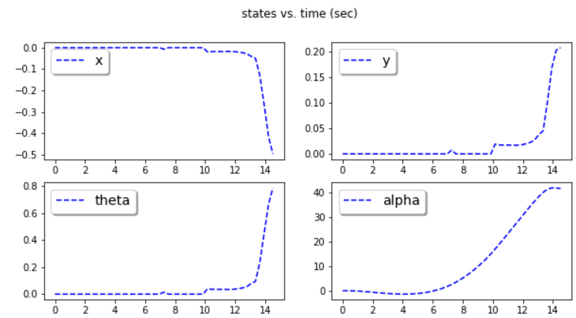

Fig. 2. Swing up trajectory states.

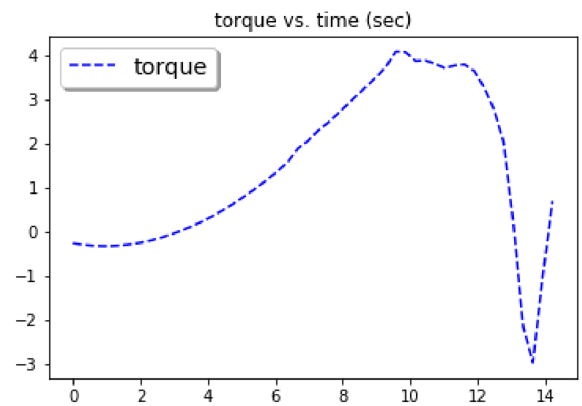

Fig. 3. Swing up trajectory torque input.

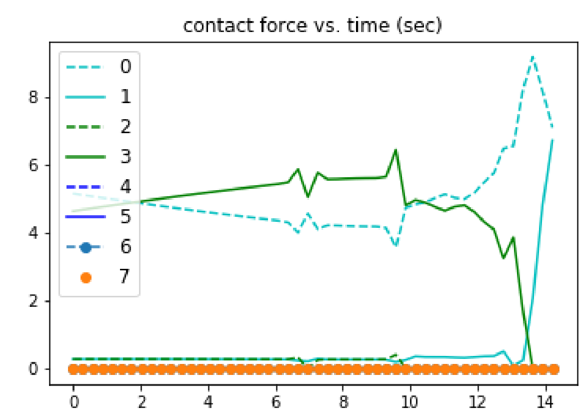

Fig. 4. Swing up trajectory ground forces.

Explain that LQR could be used to control after achieving swing up.

C. Experiments in Higher Dimensional State Space

Explain how 3D seemed to help in some cases even though some states don't enter dynamics.

D. Stable Walking Motion

Explain how to achieve stable walking motion.

Results

Explanation of results section. This should include more diagrams.

Future Work

Future work section.

References

[1] M. Posa, C. Cantu, and R. Tedrake. "A Direct Method for Trajectory Optimization of Rigid Bodies Through Contact," in The International Journal of Robotics Research, 2014 http://groups.csail.mit.edu/robotics-center/public_papers/Posa13.pdf

[2] M. Gajamohan, M. Merz, I. Thommen, and R. D'Andrea. "The Cubli: A Cube that can Jump Up and Balance," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012 https://www.ethz.ch/content/dam/ethz/special-interest/mavt/dynamic-systems-n-control/idsc-dam/Research_DAndrea/Cubli/Cubli_IROS2012.pdf

[3] J. Romaniskin, K. Gilpin, and D. Rus. "M-Blocks: Momentum-driven, Magnetic Modular Robots," in Proc. IROS, IEEE/RSJ, 2013 http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.824.7275&rep=rep1&type=pdf

[4] E. Bjerke, B. Pehrsson. "Development of a Nonlinear Mechatronic Cube," Chalmers University of Technology, 2016 http://publications.lib.chalmers.se/records/fulltext/233543/233543.pdf

[5] Y. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli, and Y. Bengio. "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization," in Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2, 2014 https://ganguli-gang.stanford.edu/pdf/14.SaddlePoint.NIPS.pdf

[6] B. Hockman and M. Pavone. "Stochastic Motion Planning for Hopping Rovers on Small Solar System Bodies," in Int. Symp. on Robotics Research, 2017 https://asl.stanford.edu/wp-content/papercite-data/pdf/Hockman.Pavone.ISRR17.pdf