Rocket Powered Descent in R^3

Bereket Abraham
Carnegie Mellon University
Pittsburgh, PA, 15213
bereketa@andrew.cmu.edu

Laura Fleury
Carnegie Mellon University
Pittsburgh, PA, 15213
lfleury@andrew.cmu.edu

Abstract – *This paper provides new results for the tracking control of an unmanned, rocket powered descent vehicle. The craft has eight input thrusters that are used to control the six translational and rotational degrees of freedom, and to achieve asymptotic tracking of four outputs, namely, three position variables for the vehicle center of mass and the angle of the axial body-fixed direction. Three cascade controllers are investigated: linear PD control, nonlinear geometric control, and novel weighted geometric control. Using numerical simulation, we examine the performance and robustness to noise of the different controllers under a canonical circular trajectory. We find that the nonlinear controllers are better suited at vertical stationkeeping at low noise levels, in particular the weighted geometric controller.*

Keywords: Cascade control, quadrotor, geometric control, nonlinear control, autonomous, underactuated system, rocket powered descent.

1 Introduction

On August 5, 2012 the Mars Science Laboratory (MSL) team from the Jet Propulsion Laboratory successfully landed the rover Curiosity in Gale crater. Their accomplishment validated a series of technical innovations in entry, descent, and landing (EDL) performance relative to previous missions. These included the first use at Mars of entry guidance to reduce the size of the landing ellipse and the first use of the Sky Crane architecture to land a 1 ton class rover on the surface. Our analysis focuses on the powered descent phase of the EDL sequence. In this phase the descent vehicle performs controlled trajectory following, unloads the rover, and then executes a ballistic escape maneuver [6].

The control problem is to precisely track smooth trajectories in R^3 with an unconstrained rigid body. The problem is challenging because the system is severely under-actuated. There are only four inputs $(F\mathbf{b}_3, M_x, M_y, M_z)$ to control 12 state variables $(x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi, \dot{\phi}, \dot{\theta}, \dot{\psi})$, including both position and attitude. In addition, the inputs are themselves idealized. For the rocket-powered lander that we are modeling, the four inputs are approximated by an array of eight fixed, throttle-able engines (see Figure 1). Finally, the rigid body dynamics is characterized by strong nonlinearities, which lead to tight inter-axis coupling effects. In order to combat this complexity we utilize the cascade hierarchy, a common architecture that is featured on planetary landers, quadcopters, VTOL aircraft, and other applications with similar dynamics.

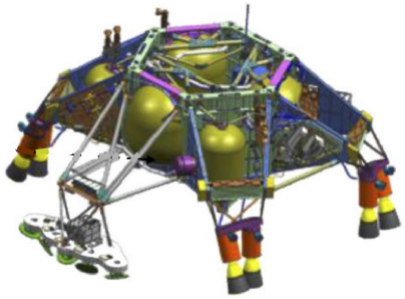

Figure 1: A diagram of the MSL descent vehicle in the Sky Crane configuration [5].

Our objectives include tracking accuracy and robust stability with respect to the rigid body dynamics. Under the assumption that the time-varying parameters can be frozen over a short period of time, the flight controllers are designed to stabilize in the presence of parametric uncertainty and noise. Both control gains in the inner attitude control loop and guidance gains in the outer position control loop are designed to maximize the vehicle performance while ensuring robustness with respect to noise. Finally, in this model we do not take into consideration propellant slosh and bending modes.

m = total mass
I = inertia matrix with respect to the body frame
$R \epsilon SO(3)$ = rotation matrix from the body frame to the inertial frame
ω = angular velocity in the body frame
x = position of the center of mass in the inertial frame
f_t = vector of thruster forces
F_T = total thrust
M = total moment in the body frame

2 Model

We control the entire system using two nested feedback loops, which form the basic building blocks of Figure 2. Together, the inner and outer loops create a hierarchical structure to control the underactuated system. The inner loop controls the vehicle's attitude while the outer loop controls the position along the given trajectory, each regulated by a PD controller. The main idea is to use the roll (ϕ) and pitch (θ) of the vehicle to affect lateral motion. We choose an inertial reference frame $\{\mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3\}$ and a body-fixed frame $\{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\}$. The origin of the body-fixed frame is located at the center of mass of this vehicle.

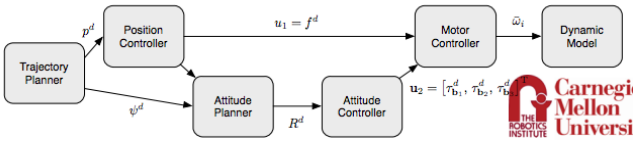

Figure 2: The Cascade PD control hierarchy.

2.1 Dynamics

First, we must define the rigid body dynamics that affect our vehicle.

$$m\ddot{\boldsymbol{x}} = -mg\boldsymbol{a}_3 + R\boldsymbol{F_T} \tag{1}$$

$$\dot{R} = R\hat{\boldsymbol{\omega}} \tag{2}$$

$$I\dot{\boldsymbol{\omega}} = \boldsymbol{M} - \boldsymbol{\omega} \times I\boldsymbol{\omega} \tag{3}$$

As you can see from Equations 1 and 3, our forces are defined in the inertial frame while our torques are defined in the body frame. We use the following rotation matrix R to convert from inertial to body frame. We model the rotation using ZXY Euler angles $(\phi, \theta, \psi)^T$, as the roll, pitch, and yaw angles respectively.

$$R = \begin{pmatrix} c\psi c\theta - s\phi s\psi s\theta & -c\phi s\psi & c\psi s\theta + c\theta s\phi s\psi \\ s\psi s\theta + s\phi c\psi s\theta & c\phi c\psi & s\psi s\theta - c\theta s\phi c\psi \\ -c\phi s\theta & s\phi & c\theta c\phi \end{pmatrix} \tag{4}$$

Finally, we use Eq. 5 to convert between the time derivative of the Euler angles and angular velocity.

$$\boldsymbol{\omega} = \begin{pmatrix} c\theta & 0 & -c\phi s\theta \\ 0 & 1 & s\phi \\ s\theta & 0 & c\theta c\phi \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \tag{5}$$

2.2 Attitude Control

We use Eq. 6 to measure the error between our desired and actual attitude. The equation can be derived using geometric reasoning, as shown by Lee, Leok and McClamroch [3]. Using the same reasoning and Eq. 2, we can also derive Eq. 7, the error in angular velocity.

$$e_R = \frac{1}{2}(R_d^T R - R^T R_d) \tag{6}$$

$$e_\omega = \omega - R^T R_d \omega_d \tag{7}$$

In order to derive a linear attitude controller, we must linearize our error metrics. Mahony, Kumar and Corke [4] do just that by assuming that the roll, pitch and angular velocities are all small. Their analysis was originally conducted for quadcopter dynamics and control, but extends well to other types of hovering craft. By linearizing about the hover position, they are able to separate the rotation matrix into separate rotations about $R(\psi_0 + \Delta\psi)$ and $R(\Delta\phi, \Delta\theta)$. From there, we can simplify the rotation error in Equation 8. If the desired angular velocity is zero (true for trajectory following), then we can use the following PD control law, Eq. 9, to determine what our desired body torques should be.

$$e_R = (\Delta\phi, \Delta\theta, \Delta\psi)^T \tag{8}$$

$$\boldsymbol{u}_2 = -K_R e_R - K_\omega e_\omega \tag{9}$$

The geometric control scheme outlined by Lee et al [3] is able to directly handle the nonlinear error functions 6 and 7. Please refer to the equation below:

$$\begin{aligned} \boldsymbol{u}_2 = I(-K_R e_R - K_\omega e_\omega) + \omega \times I\omega \\ -I(\hat{\omega} R^T R_d \omega_d - R^T R_d \dot{\omega}_d) \end{aligned} \tag{10}$$

2.3 Attitude Planner

According to the cascade control hierarchy, we use the tilt of the vehicle to affect lateral motion. In this section, we determine the necessary pitch and roll angles from the desired x and y accelerations. Because Eq. 4 is highly nonlinear, in order to invert it we have to assume small roll and pitch angles, small angular velocities, and small lateral velocities. Using Eq. 1, we can create a PD controller for the magnitude of the thrust in the axial direction and calculate the desired roll and pitch angles.

$$u_1 = m(g + \ddot{z}_d + K_{d,z}(\dot{z}_d - \dot{z}) + K_{p,z}(z_d - z)) \tag{11}$$

$$\phi_d = \frac{1}{g}(\ddot{x}_{com}\sin\psi_d - \ddot{y}_{com}\cos\psi_d) \tag{12}$$

$$\theta_d = \frac{1}{g}(\ddot{x}_{com}\cos\psi_d + \ddot{y}_{com}\sin\psi_d) \tag{13}$$

2.4 Position Control

The last step is to determine what our commanded accelerations $\ddot{x}_{com}$ and $\ddot{y}_{com}$ should be. These can be estimated using a simple PD controller, shown below.

$$\begin{aligned}\ddot{x}_{com} &= \ddot{x}_d K_{d,x}(\dot{x}_d - \dot{x}) + K_{p,x}(x_d - x)) \\ \ddot{y}_{com} &= \ddot{y}_d K_{d,y}(\dot{y}_d - \dot{y}) + K_{p,y}(y_d - y))\end{aligned} \tag{14}$$

Finally, we can obtain a nonlinear PD position controller using geometric control [3]. For our weighted geometric control $\boldsymbol{b}_3$ is calculated using a weighted average of 7:1 between the current Euler angles and desired Euler angles.

$$u_1 = m\boldsymbol{b}_3^T(\ddot{\boldsymbol{x}}_d + K_d(\dot{\boldsymbol{x}}_d - \dot{\boldsymbol{x}}) + K_p(\boldsymbol{x}_d - \boldsymbol{x}) + g\boldsymbol{a}_3) \tag{15}$$

2.5 Control Allocation

The last step is to map our four generalized control inputs to the eight fixed thrusters, while minimizing the average magnitude of each thruster in order to avoid saturation. First, we calculate the forward map from thruster magnitude to total force and torque. We then apply the Moore-Penrose pseudo inverse [7] to calculate a robust least squares solution.

$$\begin{pmatrix} F \\ M \end{pmatrix} = H_{6\times 8}\boldsymbol{f}_T \tag{16}$$

$$H^\dagger = H^T(HH^T)^{-1} \tag{17}$$

3 Results

In order to get numerical stable results, we ran our dynamics simulator at 5000 Hz for 100 sec. We ran the position controller at 2 Hz and the inner attitude controller at 10 Hz in order to get 5x frequency separation.

3.1 Gain Selection

Tuning the PD controller gains was quite challenging, given the large number of parameters. We first tuned the inner attitude controller based on estimated feed forward input, and then repeated that process for the outer position controller. The gains were chosen to minimize the L2 norm of the attitude and position mean squared error (MSE) [2], respectively. Below we show the process for the $K_{px,y}$ and the $K_{dx,y}$ gains (x and y gains were the same due to axial symmetry). Refer to Equation 14.

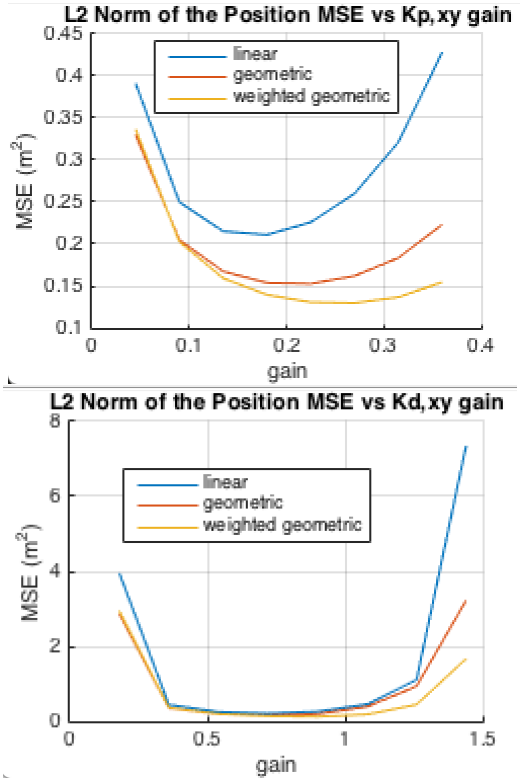

Figure 3: Mean squared position error under a range of Kp,x (top) and Kd,x (bottom) gains, per controller.

3.2 Performance

Linear control, simple geometric control, and angle weighted geometric control were all tested on a circular trajectory in order to determine which is most robust. For all three controllers, the point of weakness was discontinuity in velocity where the large change in desired angle resulted in thruster saturation and insufficient lift to maintain the lander's z-position. For the remainder of the test the thrusters remained well below saturation limits. Lander trajectories are shown in Figure 4. Roll and pitch angles remained within $\pm 20°$ from vertical.

All three controllers have low trajectory error in the horizontal plane. In the vertical direction, general geometric control has low trajectory error. Weighting the angles results in extremely low trajectory error; results are summarized in Table 1.

3.3 Robustness

We also test the performance of the controllers when subjected to Gaussian white noise in both the thrusters and the measured state. All controllers are shown to handle up to an 80 dB Signal to Noise Ratio (SNR) with varying levels of accuracy (see Table 1). Weighted geometric control is shown in Figure 5.

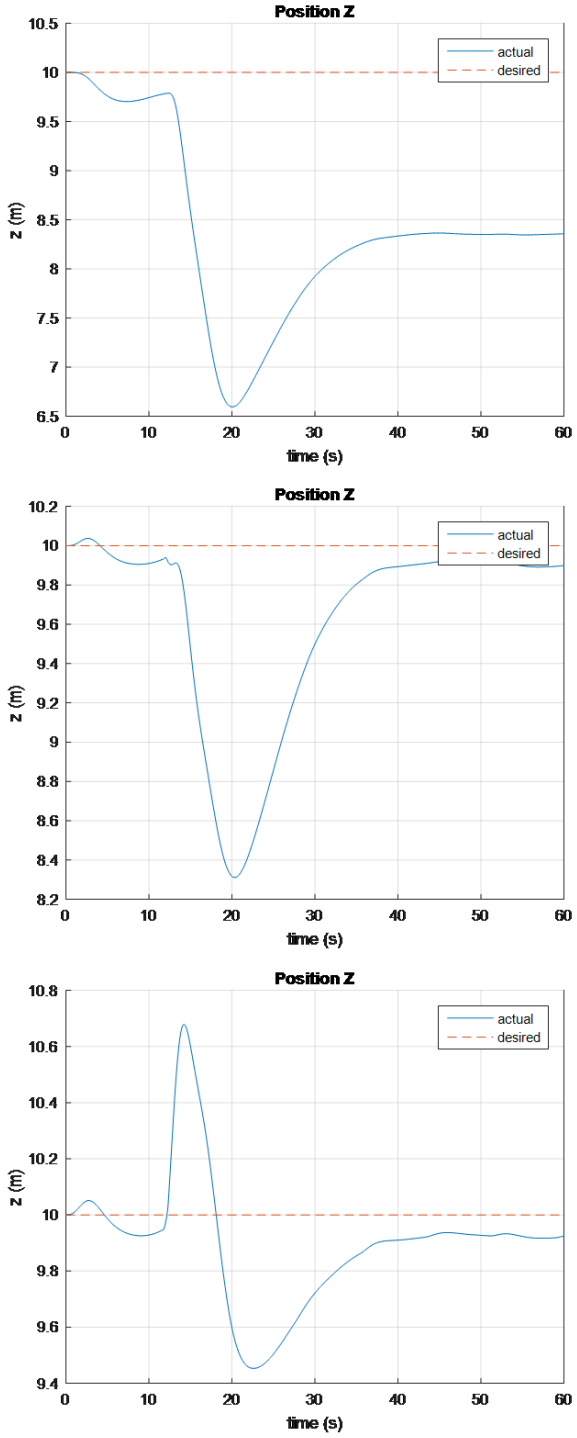

Figure 4: Z position of lander using linear control (top) simple geometric control (middle) and weighted geometric control (bottom).

Table 1: Summary of Trajectory Error

Controller	$\text{MSE}_x(\text{m}^2)$	$\text{MSE}_y(\text{m}^2)$	$\text{MSE}_z(\text{m}^2)$
Linear	0.34	0.30	3.40
SNR of 90dB	0.40	0.43	4.00
SNR of 80dB	3.30	1.80	15.20
Simple Geometric	0.38	0.29	0.39
SNR of 90dB	0.84	0.87	2.00
SNR of 80dB	1.44	3.86	66.56
Weighted Geometric	0.39	0.29	0.06
SNR of 90dB	1.00	0.45	0.38
SNR of 80dB	2.97	2.73	17.77

4 Discussion

4.1 Performance

Linear control is severely limited since attitude is not taken into account when calculating the required thrusts. This is especially apparent in our circular trajectory where constant lateral acceleration results in the lander experiencing a constant tilt from vertical. The tilted attitude results in a significant steady state error due to insufficient thrust. Geometric control addresses this issue by taking the attitude of the lander into account. It does not eliminate steady state error as it is still a PD controller, but it significantly decreases the error.

Simple geometric control is imperfect in that thrust calculations are discrete with a limited frequency. Changes in attitude that occur more rapidly than thrust calculations result in a similar problem to linear control where the thrust calculation does not take into account the true continuous attitude. Weighted geometric control includes desired angle in the calculation so it can handle rapid changes in attitude. Lee et al [3] showed similar limitations in the geometric control algorithm they applied to quadrotors.

Weighted geometric control is currently only a naive weighted average of angles. Further improvement can be made by including angular velocity and calculated torques rather than desired angles. Goodarzi et al [1] showed that discontinuities in flight trajectory can be best handled by having two different flight controllers, one to be used before the discontinuity and one after.

4.2 Robustness

While all three controllers can handle down to a SNR of 80dB before the thrusters saturate, simple geometric control is the most susceptible to noise. Even at larger SNRs simple geometric control shows a significant increase in error. In the weighted controller, the averaging of angles acts as a filter reducing the effect of noise. The linear controller shows little impact under small amounts of noise, and under larger amounts its

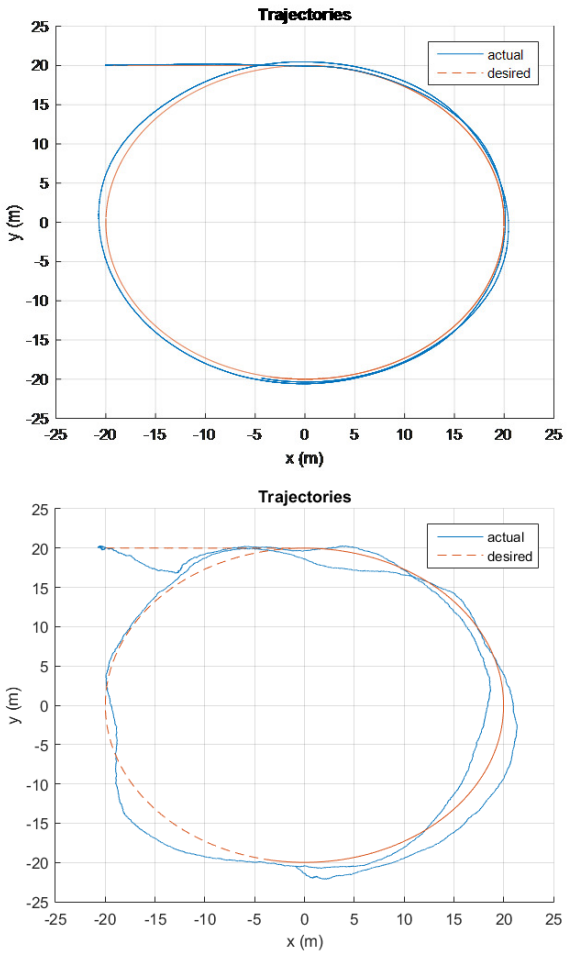

Figure 5: Trajectory under a disturbance with a SNR of 100 dB (top) and 80dB (bottom).

response is similar to the weighted geometric controller. However, if any controller is required for a system with a smaller SNR than 80dB, a filter or smoother will be required.

References

[1] F. A. Goodarzi, D. Lee, and T. Lee. Geometric adaptive tracking control of a quadrotor uav on se (3) for agile maneuvers. *ASME Journal of Dynamic Systems, Measurement and Control*, 137(9):20–32, 2015.

[2] Y. Karnavas and K. Dedousis. performance evaluation of evolutionary designed conventional agc controllers for interconnected electric power system studies in a deregulated market environment. *International Journal of Engineering, Science and Technology*, 2(3):150–166, 2010.

[3] T. Lee, M. Leoky, and N. H. McClamroch. Geometric tracking control of a quadrotor uav on se (3). In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 5420–5425. IEEE, 2010.

[4] R. Mahony, V. Kumar, and P. Corke. Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor. *Robotics & Automation Magazine, IEEE*, 19(3):20–32, 2012.

[5] M. San Martin, G. F. Mendeck, P. B. Brugarolas, G. Singh, F. Serricchio, S. W. Lee, E. C. Wong, and J. C. Essmiller. In-flight experience of the mars science laboratory guidance, navigation, and control system for entry, descent, and landing. *CEAS Space Journal*, 7(2):119–142, 2015.

[6] G. Singh, A. M. SanMartin, and E. C. Wong. Guidance and control design for powered descent and landing on mars. In *Aerospace Conference, 2007 IEEE*, pages 1–8. IEEE, 2007.

[7] S.-z. Yang, L. Wang, and P. Sun. Optimal thrust allocation logic design of dynamic positioning with pseudo-inverse method. *Journal of Shanghai Jiaotong University (Science)*, 16:118–123, 2011.