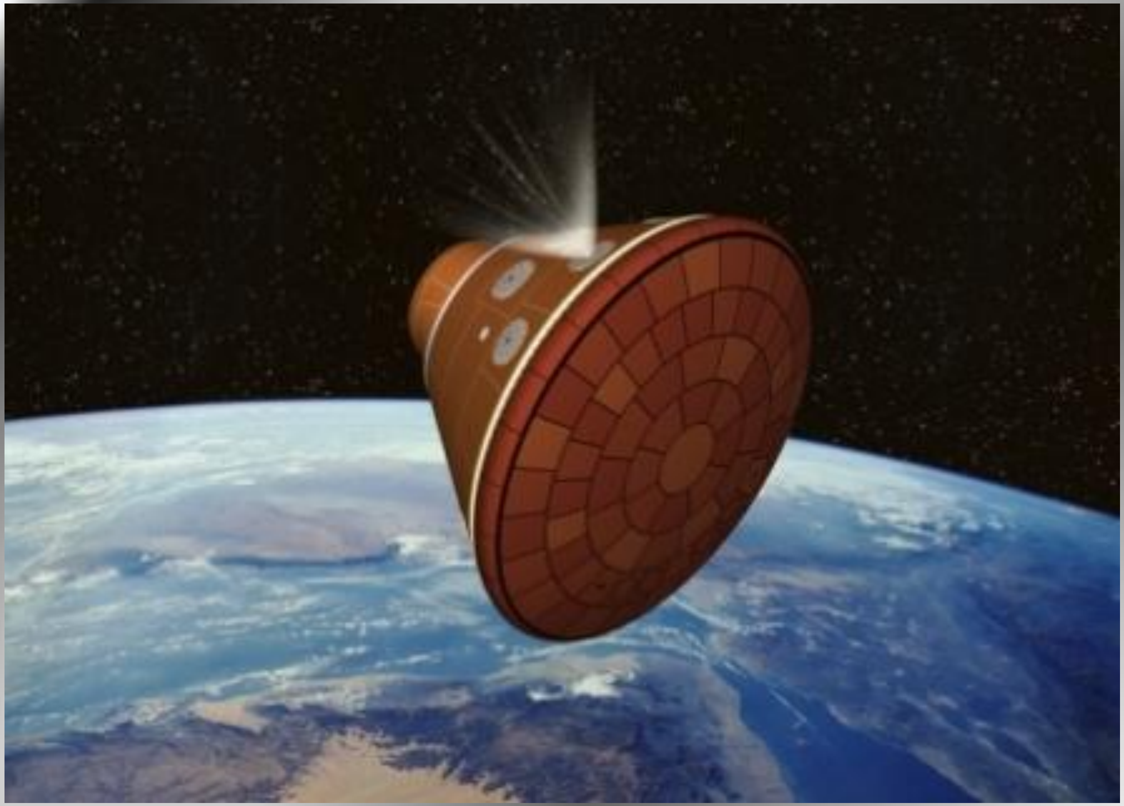
Optimal Control of Apollo Space Capsule During Atmospheric Reentry



Evan Kontras

Department of Mechanical and Aerospace Engineering

University of Missouri

Introduction

- In order to bring the Apollo space capsule safely back to Earth, an optimal control system has been designed to ensure that the final states of velocity, flight-path angle, altitude, and range to go are within a specified criteria. The single control is the capsule bank angle, effectively modulating the capsule lift and drag.
- An open loop Simulink simulation was run with nominal initial conditions to obtain a reference trajectory which defines the ideal reentry situation.
- The optimal control system has been designed to control the capsule for extreme overshoot and undershoot reentry conditions.

Reentry Specifics

- The capsule is assumed to be entering Earth's atmosphere after an initial skip out and Kepler lob, resulting in the following nominal entry conditions for velocity, flight-path angle, altitude, and range angle to go, respectively.

$V = 7162.8$ m/s $\gamma = -2.0$ deg $h = 80000$ m $\theta = 10.556$ deg

- With the nominal initial states and ideal atmospheric conditions, a constant bank angle $\sigma = 53^\circ$ would deliver the capsule to the desired final states.

- However, the optimal control system has been designed to deliver the capsule to the desired final states while having the following worst case undershoot and overshoot initial states. Entry velocity is assumed nominal, as the velocity value of 7162.8 m/s activates the control system.

Undershoot reentry initial states: $\gamma = -2.2$ deg, $h = 72{,}000$ m, and $\theta = 11.556$ deg

Overshoot reentry initial states: $\gamma = -1.8$ deg, $h = 88{,}000$ m, and $\theta = 9.556$ deg

Governing Equations

- The governing flight dynamics for the Apollo capsule during reentry are given by the following equations:

$$\dot{V} = \frac{-D}{m} - g\sin\gamma \qquad L = \bar{q}SC_L$$

$$\dot{\gamma} = \frac{L}{mV} - \frac{g}{V}\cos\gamma \qquad D = \bar{q}SC_D$$

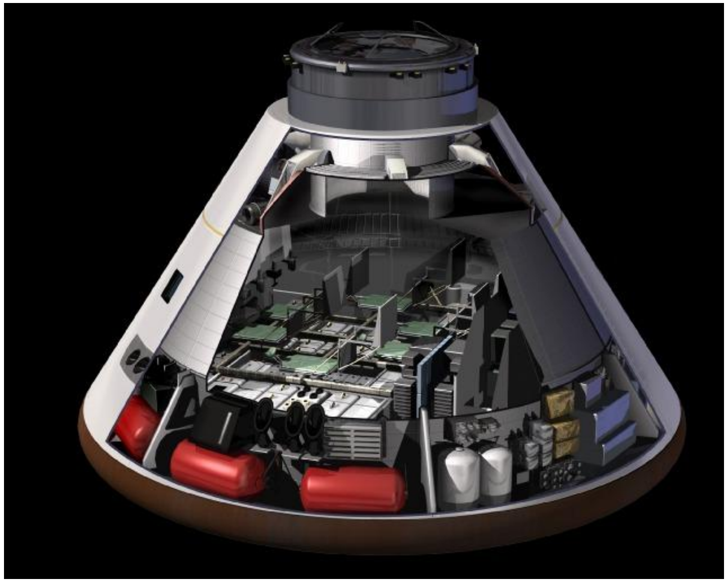
$$\dot{h} = V\sin\gamma \qquad g = g_{SL}\left(\frac{R_e}{R_e + h}\right)^2$$

$$\dot{\theta} = \frac{-V\cos\gamma}{R_e + h} \qquad \rho = \rho_0 \exp\left[\left(h_0 - h\right)/H\right]$$

Variable	*Description*
L	Lift force
D	Drag force
rho	Density of air
m	Vehicle mass
g	Gravitational acceleration
*g*_SL	*g* at sea level
h_0	Reference altitude
H	Scale height
S	Capsule surface area
*C*_l	Coefficient of lift
*C*_d	Coefficient of drag
Re	Earth radius

Capsule and Reentry Details

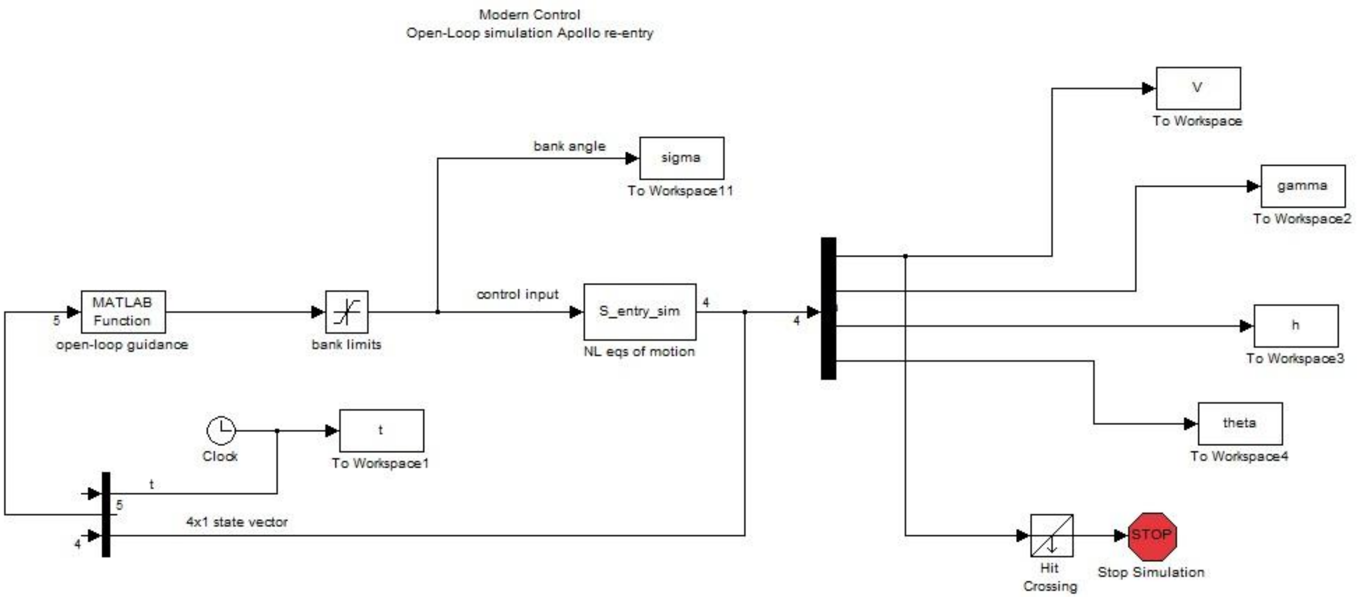
- Capsule area is $S = 12.0687\ \text{m}^2$
- Lift coefficient $C_L = 0.38773$
- Drag coefficient $C_D = 1.292433$
- Capsule mass $m = 5500$ kg



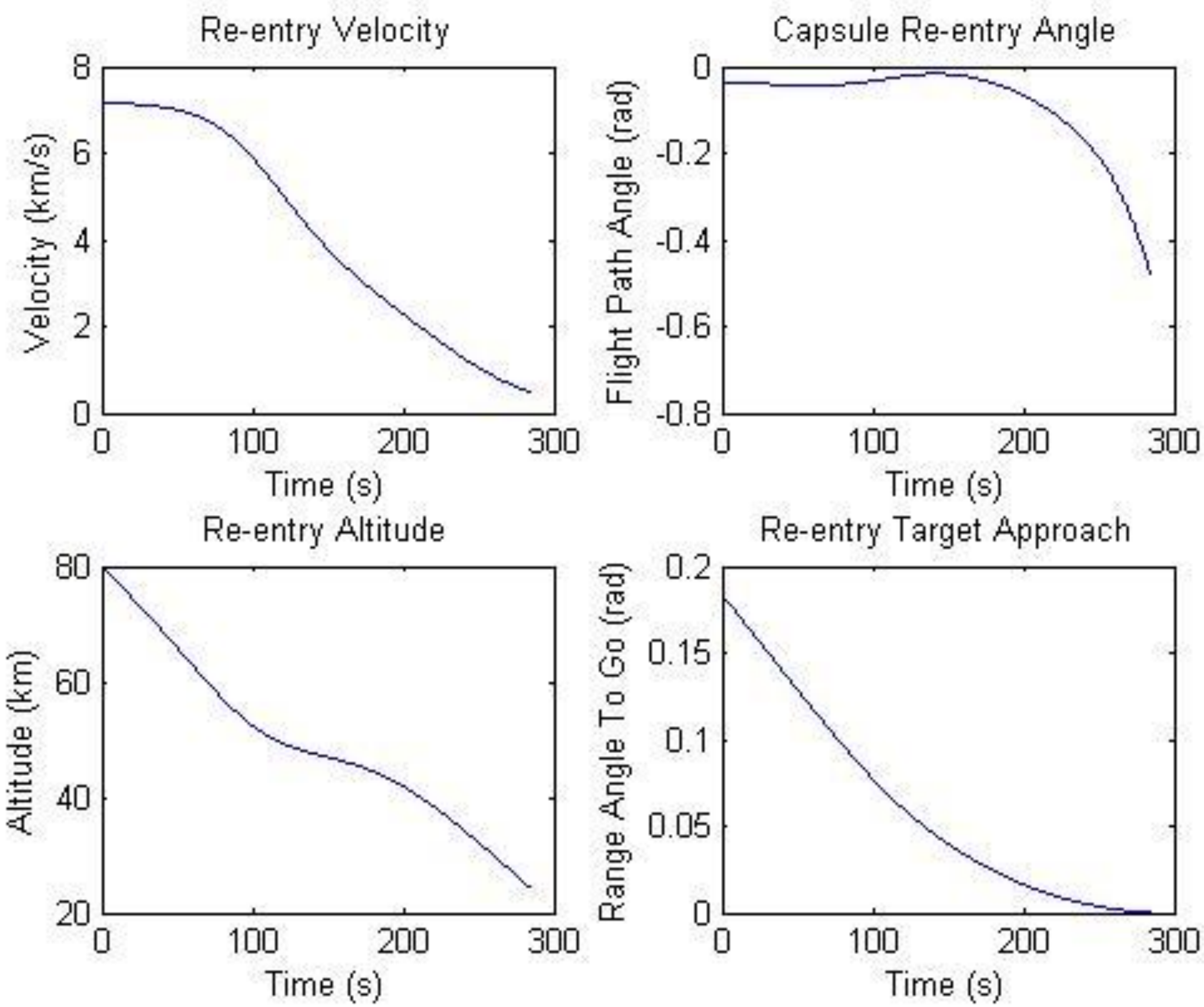
- Reference density at reference altitude $\rho_0 = 2.7649(10^{-4})\ \text{kg/m}^3$
- Reference altitude $h_0 = 60{,}000$ m
- Scale height $H = 7{,}000$ m
- Gravity at sea level $g_{SL} = 9.80665\ \text{m/s}^2$

Simulink Model (Open Loop)

- The open loop Simulink model used to obtain the reference trajectory is shown below, and plots of each reference state follow:



Reference States vs. Time



Simulation

- When the capsule velocity reaches the nominal velocity from the open loop simulation, the optimal control system is initiated.
- Control stops when velocity reaches 500 m/s.
- The closed-loop control system uses the optimal linear quadratic regulator method. MATLAB and Simulink were used for analysis.
- The actual terminal states were specified to have the following errors from the terminal reference states :
    - Final flight-path angle error $\leq$ 0.5 deg
    - Final altitude error $\leq$ 100 m
    - Final range-to-go $\leq$ 6 km

Linear Quadratic Regulator Methods

- LQR methods are well known and widely used in optimal control problems.
- LQR applies to linear systems with a quadratic performance index.
- Capable of balancing desired system response and control effort.
- A “State regulator” LQR has been implemented for the Apollo control system. The LQR obtains a control **u**(t) which attempts to keeps the system states near zero.
- The system must first be linearized to apply LQR methods. For a state space representation (SSR) to be used, this requires recalculation of the linear system matrices at regular intervals. The SSR is written in terms of state and input deviations.

$$\delta\dot{\mathrm{x}} = A\delta\mathrm{x} + B\delta\mathrm{u}$$

$$\delta\mathrm{y} = C\delta\mathrm{x} + D\delta\mathrm{u}$$

Linear Quadratic Regulator Methods

- For a non-linear system that has been linearized in this way, it is the state and input deviations that are driven to zero by the LQR. In this way, the actual states are driven to the reference states.

$\delta \mathrm{x} = \mathrm{x} - \mathrm{x}^*$

$\delta \mathrm{u} = \mathrm{u} - \mathrm{u}^*$ It should be noted that for multiple input multiple output (MIMO) systems x and u are vectors.

- Control is obtained by multiplying a gain matrix by the system states as shown below. However, LQR utilizes a unique gain matrix **K** that allows the designer to weight the importance of state, control, and final state deviations through the use of the Riccati matrix ***M***.

Control: $\mathrm{u(t)} = \mathrm{K(t)x(t)}$

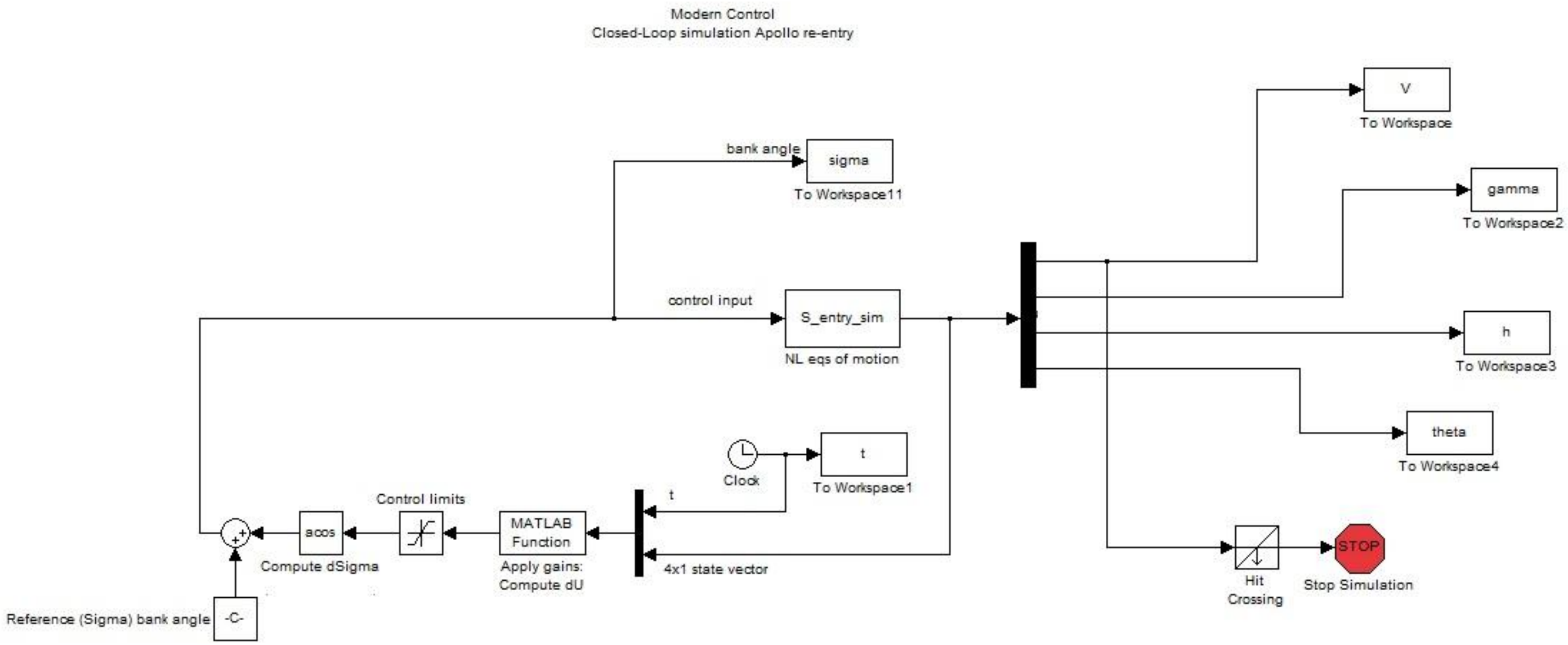
Gain matrix: $\mathrm{K(t)} = \mathrm{R}^{-1} B' M(t)$

Riccati matrix: $-\dot{M} = MA + A'M - MB\mathrm{R}^{-1}B'M + Q$

- **R** and ***Q*** are the control and state weighting matrices respectively.
- The final states are weighted based on the terminal BC, *M*(T) = *M*_final, where *M*_final is the user defined weighting matrix.

Simulink Model (Closed Loop)

The closed loop Simulink model used to implement LQR methods for optimal control of the Apollo reentry trajectory is shown below.



Summary of Simulation Process

The LQR method has been implemented using Simulink and Matlab as follows

- First the reference trajectory was obtained from the open loop simulation and all states stored as a .mat file. The .mat file was then used to create polynomial fits to give all reference states as functions of time.
- An m-file was provided which integrated the Riccati equation offline and computed the gain matrix for user specified state, control, and final state weighting values.
- The state gains were stored in a similar manner and polynomials of state gains vs. time were created.
- Using the open loop Simulink simulation as the basis, the state outputs from the non-linear eq. of motion and simulation time were input to a “closed loop” m-file which computed state deviations from reference, computed state gains, and calculated and output the control deviation (change in bank angle).
- The bank angle deviation was added to the reference bank angle of 53 deg.
- The new bank angle was input to the non-linear eq. of motion, closing the system loop.
- This process was repeated over time until simulation velocity reached 500 m/s.

Simulation Results

- With the use of the “State regulator” LQR method, it is possible to control the Apollo capsule during reentry for both worst case scenarios. However, many different Q, R, and M matrix values were tried but the optimal values were not completely found. No set of values was tested by the user to yield all three terminal state errors as specified, though the set of values does exist. Presented here are the closest of many trials. A monte carlo simulation would be capable of finding optimal Q, R, and M values.

The magnitude of final errors were found to be:

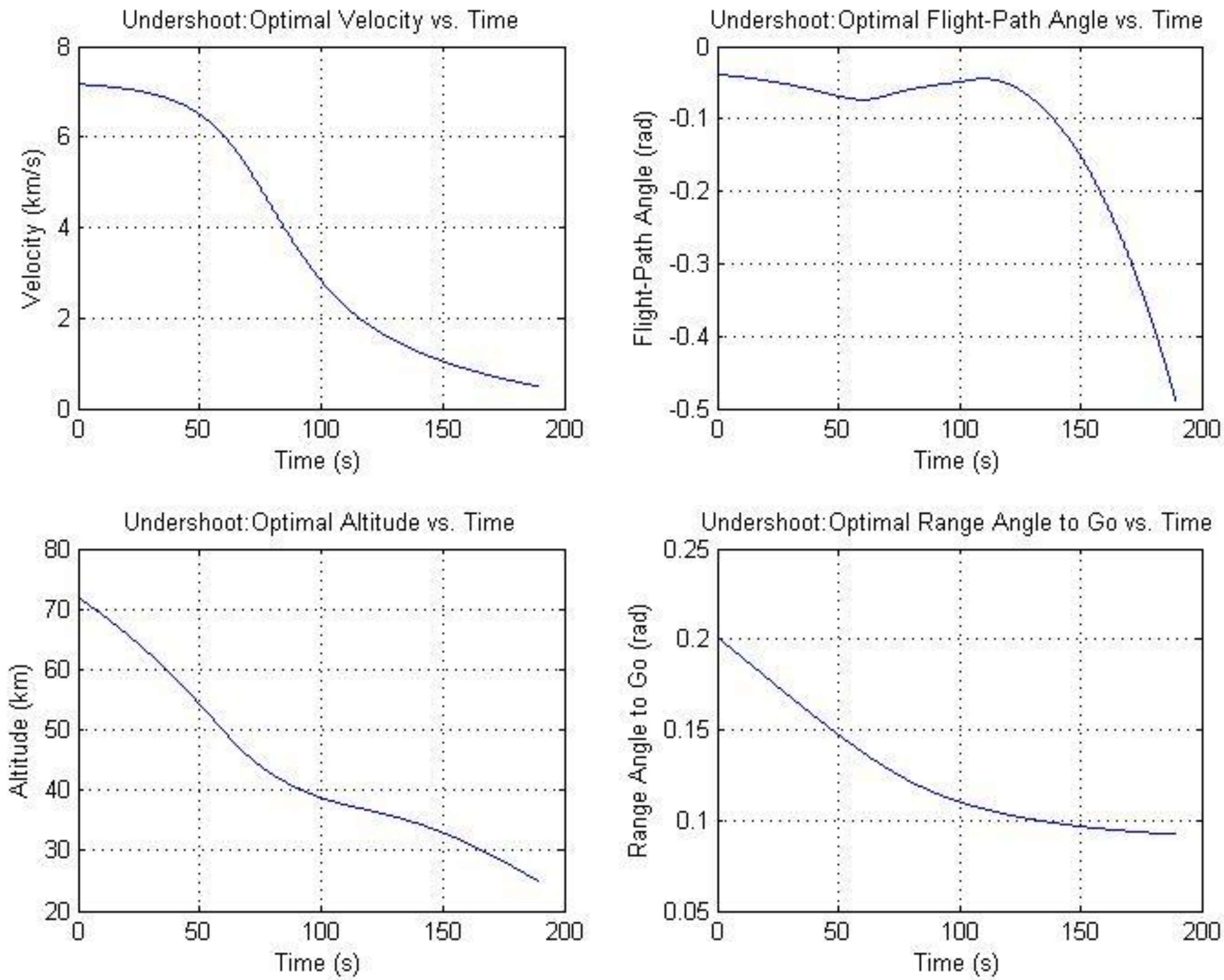
<u>For undershoot initial conditions</u>

- Final flight-path angle error ≤ 1.15 deg
- Final altitude error ≤ 31.6 m
- Final range-to-go ≤ 58.9 km

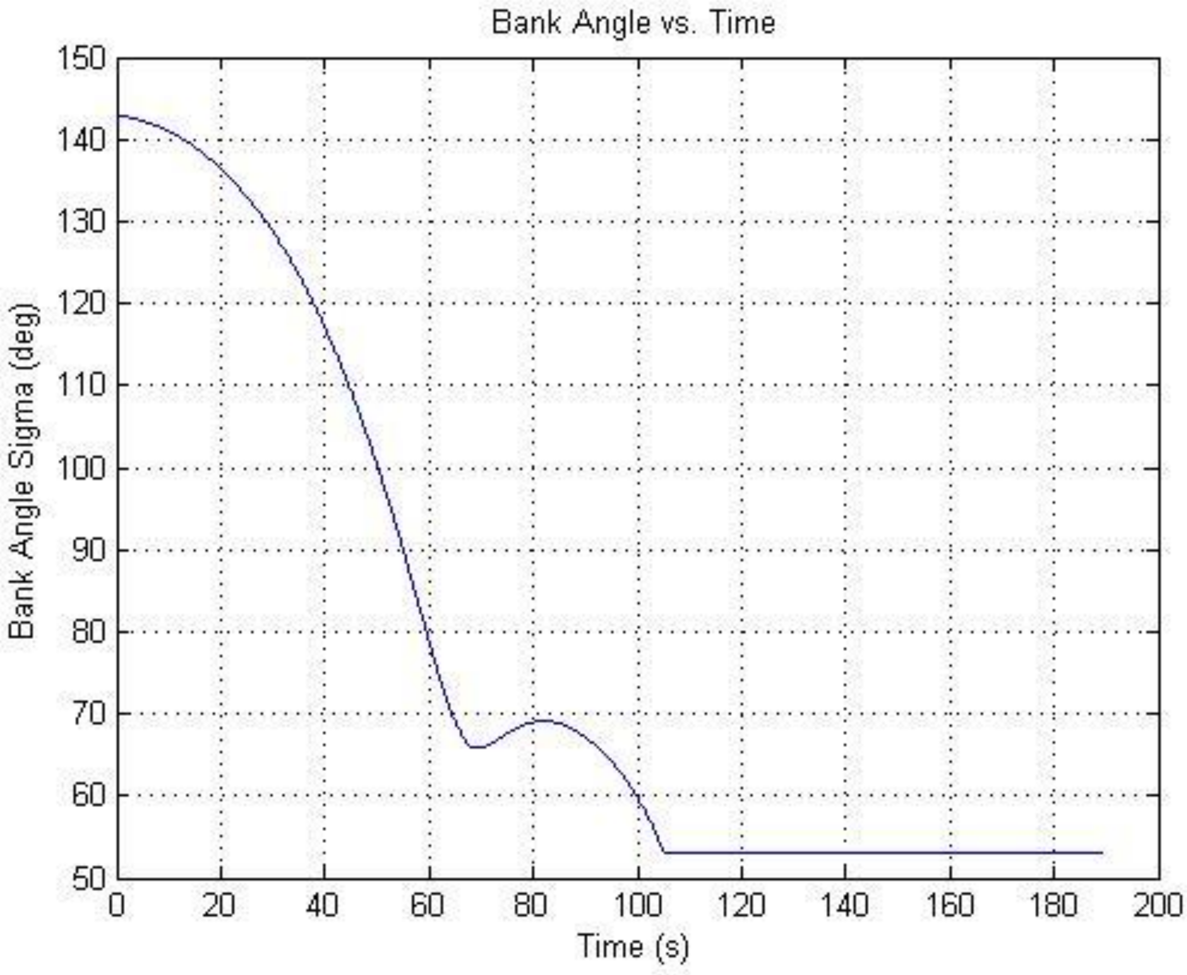
<u>For overshoot initial conditions</u>

- Final flight-path angle error ≤ 0.344 deg
- Final altitude error ≤ 88.7 m
- Final range-to-go ≤ 167 km

Results (Undershoot)

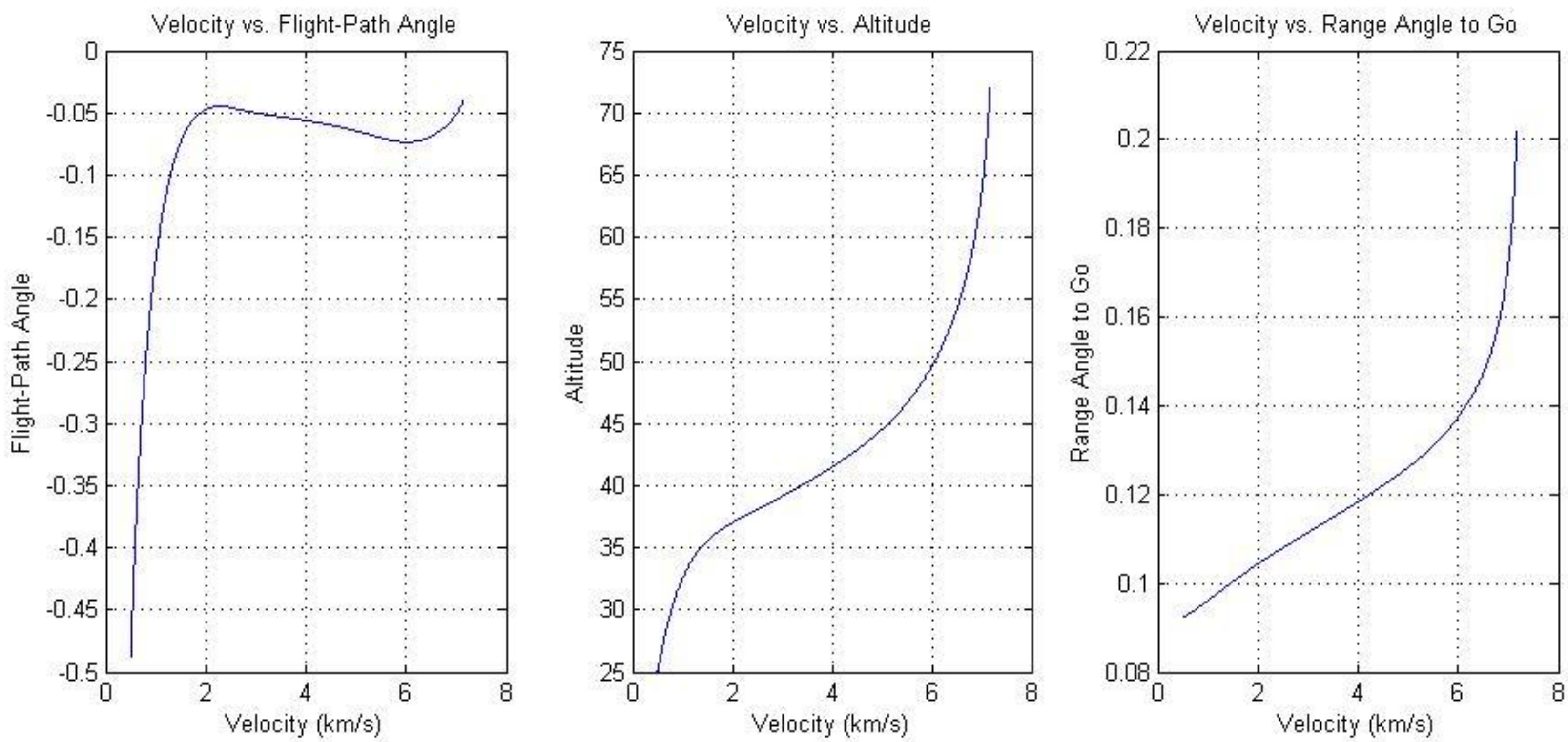


Bank Angle

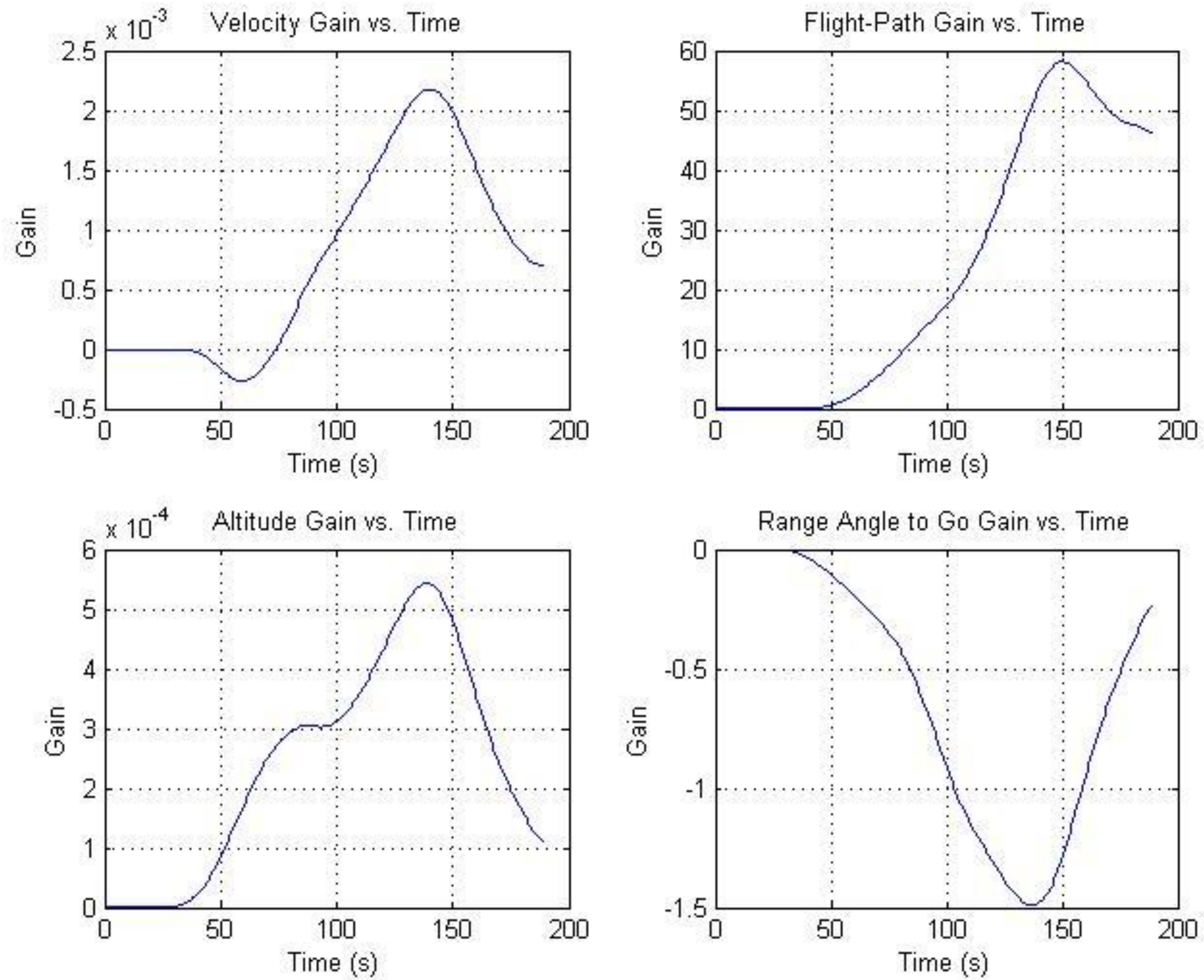


Bank Angle vs. Time
Bank Angle Sigma (deg)
150
140
130
120
110
100
90
80
70
60
50
0
20
40
60
80
100
120
140
160
180
200
Time (s)

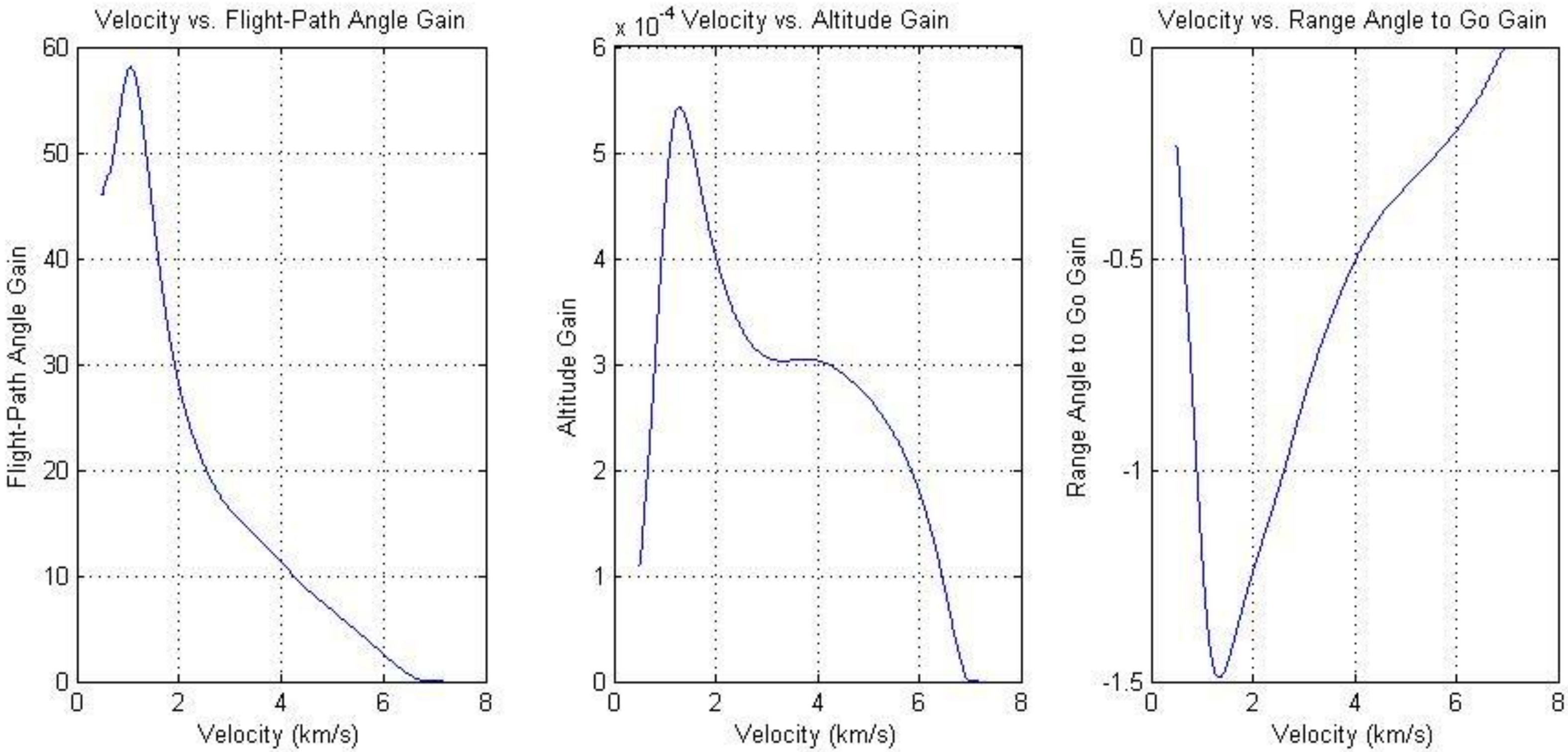
Velocity vs. States



States Gains vs. Time

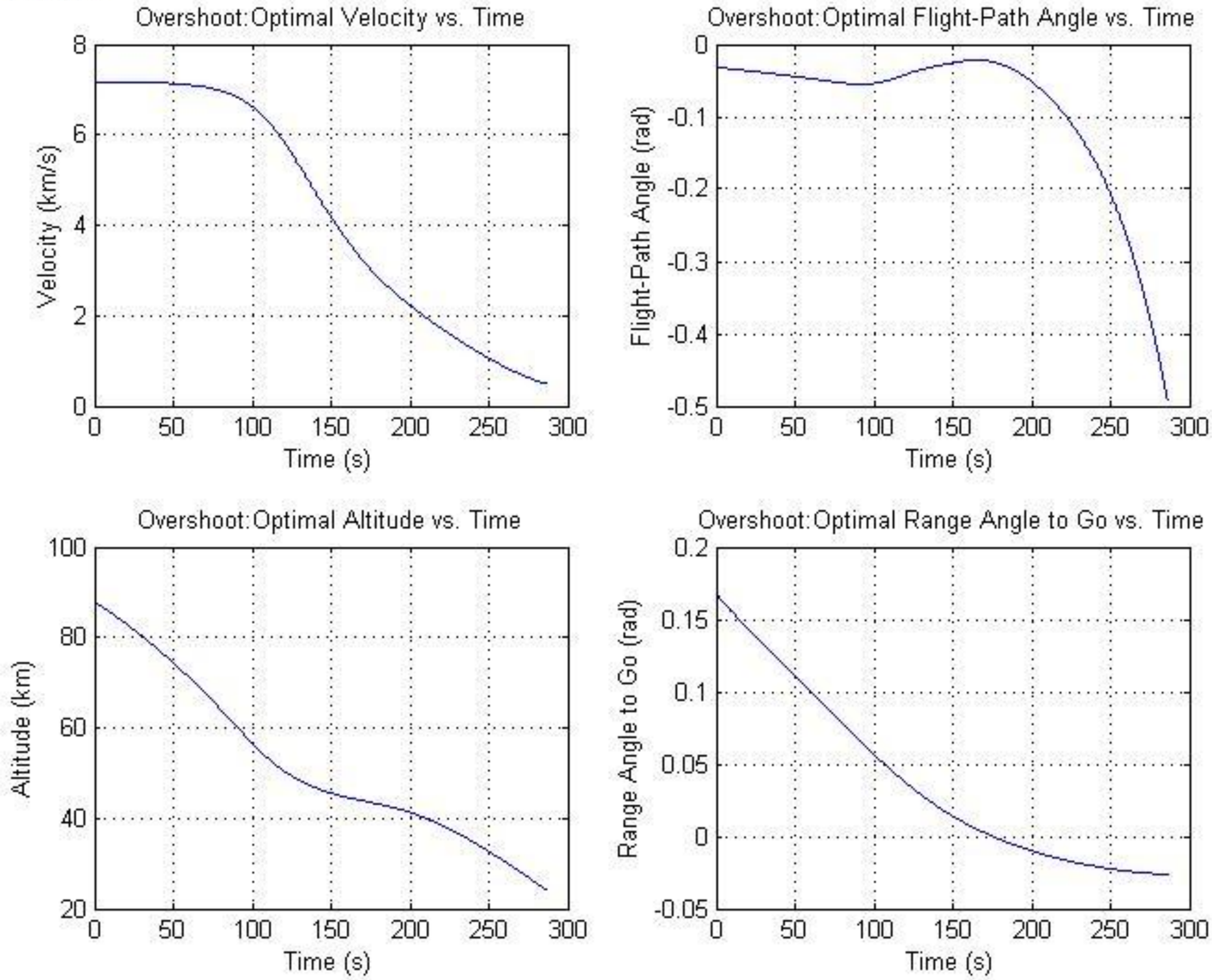


Velocity vs. State Gains

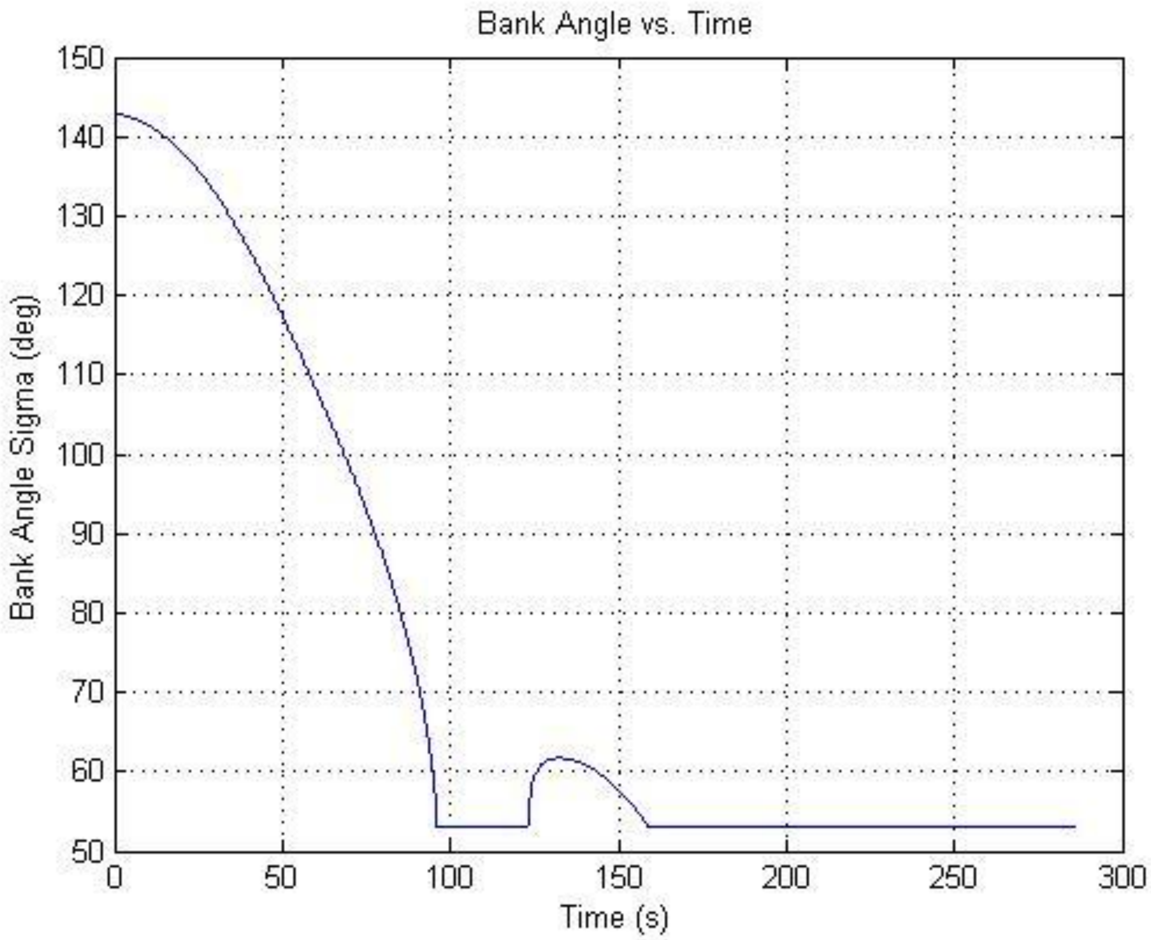


MIZZOU ENGINEERING™

Results (Overshoot Scenario)

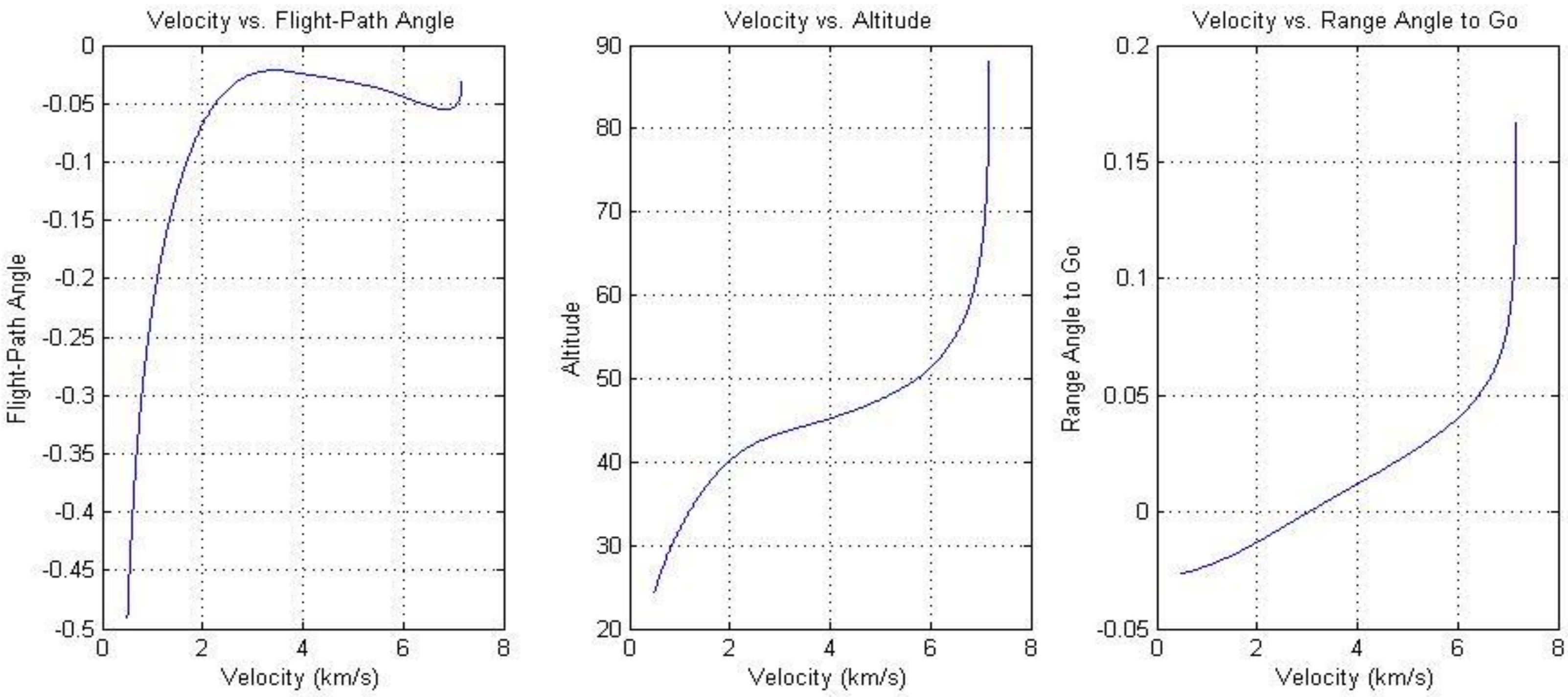


Bank Angle

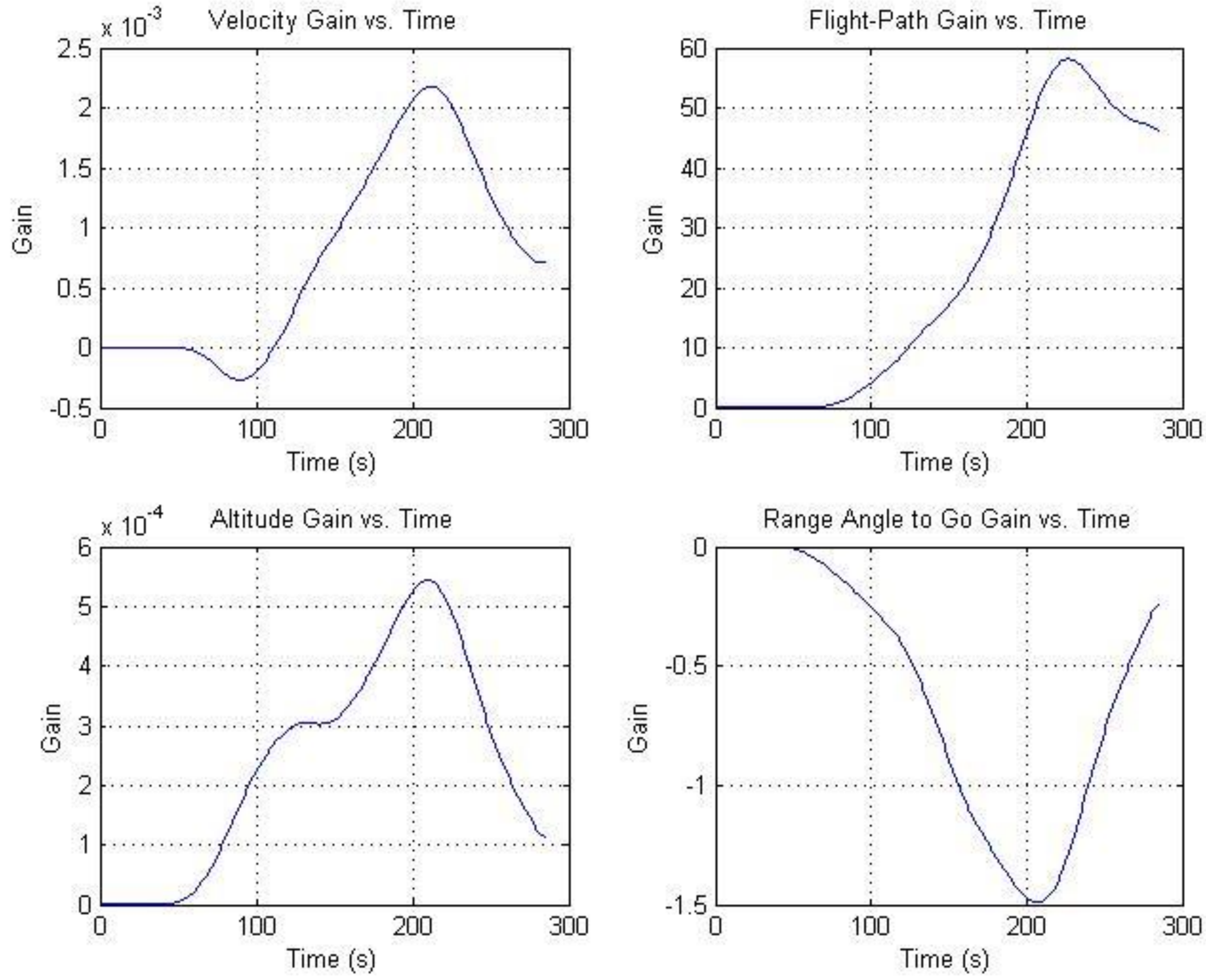


Bank Angle vs. Time
150
140
130
120
110
100
90
80
70
60
50
Bank Angle Sigma (deg)
0
50
100
150
200
250
300
Time (s)

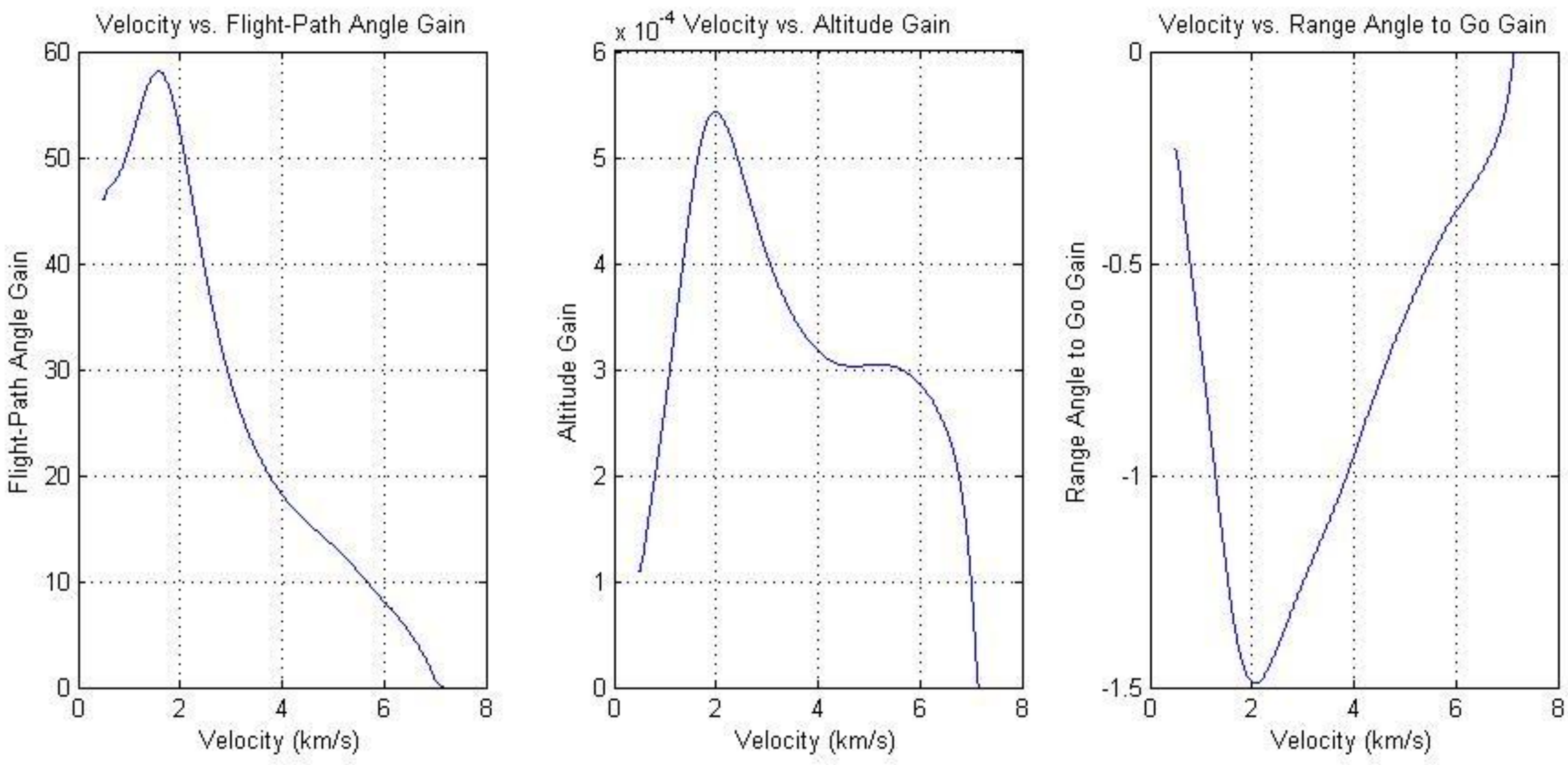
Velocity vs. States



State Gains vs. Time



Velocity vs. State Gains



Conclusions

- The analysis done has proven that the LQR method is an effective way of obtaining a desired system response, while ensuring that control effort is not out of physical limits.
- Predicting/guessing the Q, R, and M matrix values proved to be very difficult. Many trials were attempted with a large variation in final state error results between the three states of interest.
- The computational complexity of the simulation lead to long simulation running times (~2-6 min.), making multiple trials difficult.
- However, even without obtaining all final states within specified error, it is actually quite easy to ensure one, or two of the three state errors are satisfied, demonstrating the control one can impose on a physical system using LQR and the effectiveness of the simulation created.

REFERENCES

[1] C. A. Kluever, “MAE 7720 Modern Control class notes”, University of Missouri, Columbia, MO, 2010.

[2] D.S. Naidu, “Optimal Control Systems”, CRC press, Boca Raton, FL, 2000.

[3] P.N. Paraskevopoulos, “Modern Control Engineering”, Marcel Dekker Inc., New York, NY, 2002.

[4] W.L. Brogan, “Modern Control Theory”, Prentice-Hall, Engelwood Cliffs, NJ, 1985.

[5] C.K. Chui, G. Chen, “Linear Systems and Optimal Control”, Springer Publishers, New York, NY, 1989.