2005 American Control Conference
June 8-10, 2005. Portland, OR, USA

ThA11.1

Feedback Linearization Guidance for Approach and Landing of Reusable Launch Vehicles

Bradley T. Burchett

Abstract— **A feedback linearization guidance system is developed for Approach and Landing of a reusable launch vehicle. The three-input three-output system is derived through a state transformation using Lie algebra, resulting in a five state linear system. The linearization is shown to be valid for headings within ninety degrees of runway heading and flight path angles within ninety degrees of horizontal. A pseudo control is designed based on a Linear Quadratic tracker with integrators. Lie algebra then gives the transformation from pseudo control to physical control. An approximate aerodynamic model gives a further transformation from lift, drag, and bank angle, to negative z axis acceleration, bank angle, and speed brake position. The controller is simulated in a Matlab / Simulink environment, and certain controller parameters are optimized using a genetic algorithm.**

I. INTRODUCTION

Feedback linearization is a powerful method of transforming a non-linear multivariable plant into a model with linear error behavior. Any system that is affine in the control can be controlled using feedback linearization control. The point mass dynamics of a Reusable Launch Vehicle (RLV) in gliding return flight can be written so that they are affine in the terms $L\sin\phi$, $L\cos\phi$ and D. Treating these quantities as the physical controls, we can arrive at a feedback linearization scheme for the otherwise complicated non-linear dynamics of a RLV. This method is applied with the goal of guiding the RLV to within 150ft of the ground on a specified glidepath within 30ft of runway centerline. Final flare guidance is beyond the scope of this work.

Many authors have recently investigated advanced methods of guidance and control for reusable launch vehicles. Barton [1] , [2] , and Girerd [3] have proposed new methods for onboard trajectory generation, relying on linearization through a change in independent variable, and optimizing the trajectory through solution of a two-point boundary value problem. Grantham [4] has proposed the use of an adaptive critic neural network for trajectory generation and guidance during the final approach phase. Schmitt and Burchett [5] have most recently proposed a genetic algorithm (GA) tuned fuzzy logic approach for final approach guidance. Schiermann et al have authored a number of papers recently on this topic, the most recent of which [6] includes backstepping guidance laws with onboard trajectory reshaping.

This work investigates the possibility of employing feedback linearization control to generate the guidance commands on final approach so that the error signals are asymptotically stable. Section II reviews the requirements for feedback linearization, shows the system dynamic equations, and proposes a format that meets the requirements. Section III reviews input/output feedback linearization using Lie algebra. Section IV presents the resulting mathematics for the RLV dynamics. Section V shows performance of the proposed guidance scheme.

II. FEEDBACK LINEARIZATION OF MIMO SYSTEMS

Feedback linearization of multi-input multi-output (MIMO) systems can be applied whenever the equations of motion can be written in the following form. [7]

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g_1}(\mathbf{x})\,\mathbf{u_1} + \ldots + \mathbf{g_m}(\mathbf{x})\,\mathbf{u_m} \tag{1}$$

$$\begin{aligned} y_1 &= h_1(\mathbf{x}) \\ &\vdots \\ y_m &= h_m(\mathbf{x}) \end{aligned}$$

or, more compactly,

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\,\mathbf{u} \\ \mathbf{y} &= \mathbf{h}(\mathbf{x})\,. \end{aligned} \tag{2}$$

That is, the system must be *affine* in the control. The equations of motion for an aircraft flying a gliding return trajectory are shown in Equation 3[8].

$$\begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{h} \\ \dot{\psi} \\ \dot{\gamma} \\ \dot{V} \end{Bmatrix} = \begin{Bmatrix} V\cos\gamma\cos\psi \\ V\cos\gamma\sin\psi \\ V\sin\gamma \\ 0 \\ -\frac{g\cos\gamma}{V} \\ -g\sin\gamma \end{Bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{mV\cos\gamma} & 0 & 0 \\ 0 & \frac{1}{mV} & 0 \\ 0 & 0 & -\frac{1}{m} \end{bmatrix} \begin{Bmatrix} L\sin\phi \\ L\cos\phi \\ D \end{Bmatrix} \tag{3}$$

The aircraft is treated as a point mass flying over a flat, non-rotating earth. The states are as follows: x = runway downrange coordinate (ft), y = runway crossrange coordinate (ft), h = height above runway (ft), ψ = aircraft heading relative to runway heading (rad), γ = flight path angle (rad), V = aircraft forward speed (ft/s). These equations are affine in the quantities $L\sin\phi$, $L\cos\phi$, and D.Thus, if we consider these quantities to be the controls, we can apply feedback linearization. First, define the outputs as y, h,

This work was supported in part by NASA contract number NAG8-1915

Bradley T. Burchett is an Assistant Professor of Mechanical Engineering at Rose-Hulman Institute of Technology, 5500 Wabash Ave., Terre Haute, IN 47803 burchett@rose-hulman.edu

0-7803-9098-9/05/$25.00 ©2005 AACC

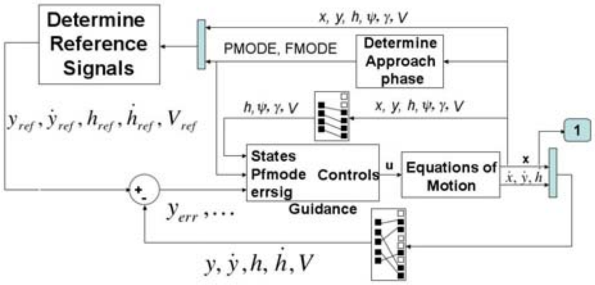

Fig. 1. RLV Simulation Block Diagram.

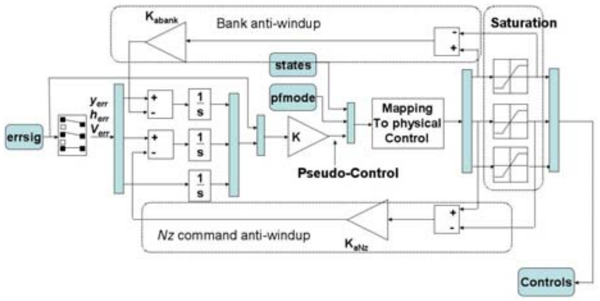

Fig. 2. RLV Guidance Subsystem Block Diagram.

and V, making the system 'square', that is equal number of inputs and outputs. Then

$$\mathbf{h}(\mathbf{x}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x} = \left\{ \begin{array}{c} y \\ h \\ V \end{array} \right\} \qquad (4)$$

III. Input Output Feedback Linearization

The Lie derivative is defined as

$$L_{\mathbf{f}} h = \nabla h \bullet f \qquad (5)$$

Where h is a scalar function of n variables, such as the first element of $\mathbf{h}$, and $\mathbf{h}$ is a vector function with n elements. The Lie derivative is thus a scalar product. Likewise

$$L_g L_f h = \nabla\left(\nabla h \bullet f\right) g \qquad (6)$$

and

$$L_g L_f^2 h = \nabla\left(\nabla\left(\nabla h \bullet f\right) f\right) g$$

and the ∇ operator can be expanded as

$$\nabla h = \begin{bmatrix} \frac{\partial h}{\partial y_1} & \frac{\partial h}{\partial y_2} & \cdots & \frac{\partial h}{\partial y_n} \end{bmatrix}. \qquad (7)$$

Then the input-output linearization is accomplished by differentiating the outputs until the inputs appear. For instance, using Lie Algebra, the first derivative of the jth state in the linear system is

$$\dot{y_j} = L_{\mathbf{f}} h_j + \sum_{i=1}^{m} \left(L_{g_i} h_j\right) u_i \qquad (8)$$

If the inputs do not appear, then we have to differentiate again. For each output, we then find the highest derivative of the corresponding linear state by

$$y_j^{(r_j)} = L_{\mathbf{f}}^{r_j} h_j + \sum_{i=1}^{m} L_{g_i} L_{\mathbf{f}}^{r_j - 1} h_j u_i \qquad (9)$$

Where r_j is the smallest integer such that at least one of the inputs appears in $y_j^{(r_j)}$. After performing the procedure for each output, we are left with the m equations corresponding to the m outputs.

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \begin{bmatrix} L_{\mathbf{f}}^{r_1} h_1(\mathbf{x}) \\ \vdots \\ L_{\mathbf{f}}^{r_m} h_m(\mathbf{x}) \end{bmatrix} + \mathbf{E}(\mathbf{x}) \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix}$$

Where $\mathbf{E}$, the decoupling matrix is defined as:

$$\mathbf{E}(\mathbf{x}) = \begin{bmatrix} L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_1 - 1} h_1 & \cdots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_1 - 1} h_1 \\ \vdots & \ddots & \vdots \\ L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_m - 1} h_m & \cdots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_m - 1} h_m \end{bmatrix}$$

As long as $\mathbf{E}$ is non-singular, the following transformation from pseudo-control to physical control is valid.

$$\mathbf{u} = -\mathbf{E}^{-1} \begin{bmatrix} L_{\mathbf{f}}^{r_1} h_1(\mathbf{x}) \\ \vdots \\ L_{\mathbf{f}}^{r_m} h_m(\mathbf{x}) \end{bmatrix} + \mathbf{E}^{-1} \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix} \qquad (10)$$

And the corresponding linear system is

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \begin{bmatrix} v_1 \\ \vdots \\ v_m \end{bmatrix} \qquad (11)$$

IV. Feedback Linearization for RLV Gliding Dynamics

Applying Eq. (8) to Eq. (3), for j = 1 and 2, we get zero. Thus, we need to take a second derivative. The quantity $L_{\mathbf{g}_i} L_{\mathbf{f}} h_1$ expands to

$$L_{\mathbf{g}_i} L_{\mathbf{f}} h_1 = \begin{bmatrix} \frac{\partial L_{\mathbf{f}} h_1}{\partial x} & \frac{\partial L_{\mathbf{f}} h_1}{\partial y} & \frac{\partial L_{\mathbf{f}} h_1}{\partial h} & \frac{\partial L_{\mathbf{f}} h_1}{\partial \psi} & \frac{\partial L_{\mathbf{f}} h_1}{\partial \gamma} & \frac{\partial L_{\mathbf{f}} h_1}{\partial V} \end{bmatrix} \mathbf{g}_i$$

which, for j = 1, yields

$$\ddot{y}_1 = \frac{\cos\psi}{m} L \sin\phi - \frac{\sin\gamma \sin\psi}{m} L \cos\phi - \frac{\cos\gamma \sin\psi}{m} D$$

and for j - 2, we get

$$\ddot{y}_2 = -g + \frac{L}{m} \cos\gamma \cos\phi - \frac{D}{m} \sin\gamma \qquad (12)$$

For the third ouput, applying Eq (8) gives

$$\dot{y}_3 = -gsin\gamma - \frac{D}{m}. \qquad (13)$$

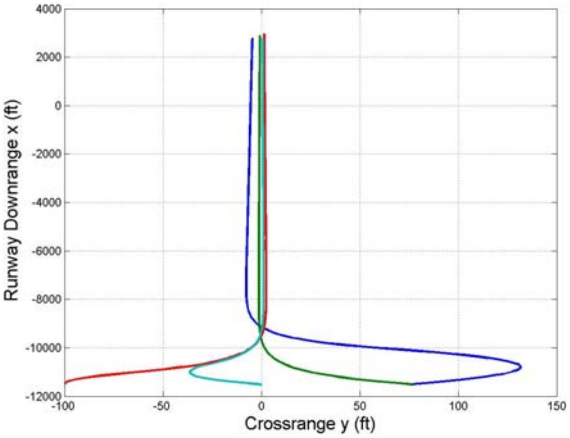

Fig. 3. Horizontal Guidance Performance for Various Initial Conditions.

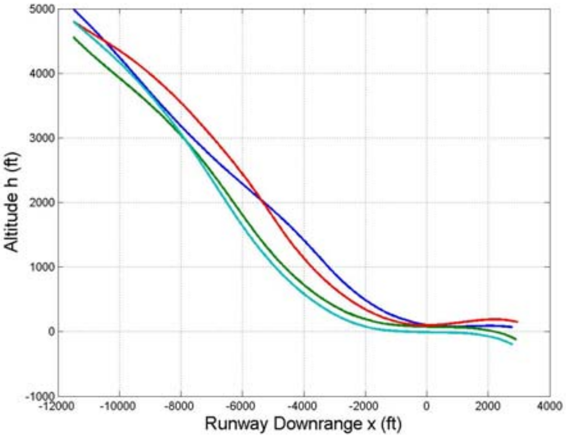

Fig. 4. Vertical Guidance Performance for Various Initial Conditions.

The decoupling matrix is

$$\mathbf{E} = \frac{1}{m}\left[\begin{array}{ccc} \cos\psi & -\sin\gamma\sin\psi & -\cos\gamma\sin\psi \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & 0 & -1 \end{array}\right] \tag{14}$$

Which is clearly invertible as long as $\psi \neq \pi/2$ and $\gamma \neq \pi/2$. The resulting linear system is:

$$\begin{aligned} \dot{\mathbf{z}} &= \mathbf{Az} + \mathbf{Bv} \\ \mathbf{y} &= \mathbf{Cz} \end{aligned} \tag{15}$$

Where $\mathbf{z} = \left[\begin{array}{ccccc} y_1 & \dot{y_1} & y_2 & \dot{y_2} & y_3 \end{array}\right]$, $\mathbf{v} = \left[\begin{array}{ccc} v_1 & v_2 & v_3 \end{array}\right]$ and

$$\mathbf{A} = \left[\begin{array}{ccccc} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

$$\mathbf{B} = \left[\begin{array}{ccc} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array}\right]$$

$$\mathbf{C} = \left[\begin{array}{ccccc} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

We can design a control for this linear system, assuming full-state feedback since the linear state is related to the physical state as follows:

$$\left\{\begin{array}{c} y_1 \\ \dot{y_1} \\ y_2 \\ \dot{y_2} \\ y_3 \end{array}\right\} = \left\{\begin{array}{c} y \\ V\cos\gamma\sin\psi \\ h \\ V\sin\gamma \\ V \end{array}\right\}$$

An initial design for the pseudo control was found as $\mathbf{v} = -\mathbf{Kz}$, where the gain $\mathbf{K}$ was selected as the LQR gain with $\mathbf{Q} = \mathbf{I}$, and $\mathbf{R} = 50\mathbf{I}$, of the appropriate dimensions. This design was not capable of tracking the flare maneuver to a satisfactory level of steady state error. Thus, the system was then augmented with integrators after a technique given by Kuo. [9] This can be done for MIMO systerms by augmenting the system state space description as follows.

$$\bar{\mathbf{A}} = \left[\begin{array}{cc} \mathbf{A} & \mathbf{0} \\ -\mathbf{C} & \mathbf{0} \end{array}\right]$$

$$\bar{\mathbf{B}} = \left[\begin{array}{c} \mathbf{B} \\ \mathbf{0} \end{array}\right]$$

$$\bar{\mathbf{C}} = \left[\begin{array}{cc} \mathbf{C} & \mathbf{0} \end{array}\right]$$

$$\bar{\mathbf{K}} = \left[\begin{array}{cc} \mathbf{K} & \mathbf{K}_I \end{array}\right]$$

$$\mathbf{K} = \left[\begin{array}{ccccc} K_{p1} & K_{d1} & 0 & 0 & 0 \\ 0 & 0 & K_{p2} & K_{d2} & 0 \\ 0 & 0 & 0 & 0 & K_{p3} \end{array}\right]$$

And $\mathbf{K}_I$ is the diagonal matrix of integral gains. The final system block diagram is given in Figure 1. Then applying Eq (10), the physical control is:

$$\mathbf{u} = \left\{\begin{array}{c} Ls_\phi \\ Lc_\phi \\ D \end{array}\right\} = \left\{\begin{array}{c} -\frac{s_\gamma s_\psi mg}{c_\psi c_\gamma} \\ -\frac{mg}{c_\gamma} \\ 0 \end{array}\right\} + \left\{\begin{array}{c} \frac{mv_1}{c_\psi} + \frac{s_\gamma s_\psi mv_2}{c_\psi c_\gamma} - \frac{s_\psi mv_3}{c_\psi c_\gamma} \\ \frac{mv_2}{c_\gamma} - \frac{ms_\gamma v_3}{c_\gamma} \\ -mv_3 \end{array}\right\} \tag{16}$$

Where we have used the notation $\sin(\theta) = s_\theta$, $\cos(\theta) = c_\theta$ for brevity. Historically, the guidance commands provided to the RLV inner-loop controller have been, bank angle, ϕ, negative z axis acceleration, N_z, and speed brake position δ_{SB}. We can convert our physical control to the desired guidance commands through interpretation of Eq (16) followed by inversion of an approximate aerodynamic model. First note that if we take the 2-norm of the first two elements of $\mathbf{u}$, we will get the desired lift.

$$L = \sqrt{(L\sin\phi)^2 + (L\cos\phi)^2}$$

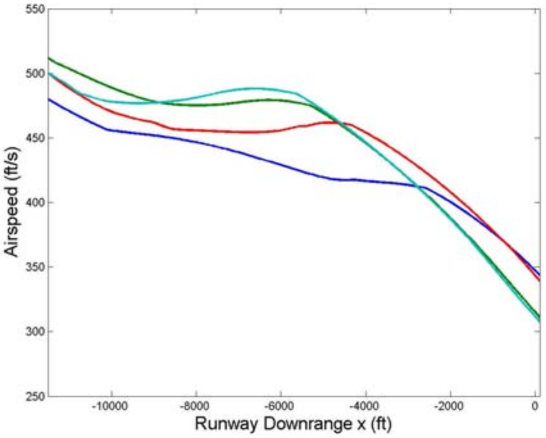

Fig. 5. Forward Speed Guidance Performance for Various Initial Conditions.

Then the desired bank angle can be determined from the first element since $\sin^{-1}$ gives a unique solution between $\pm\pi/2$.

$$\phi = \sin^{-1}\left(L \sin\left(\phi/L\right)\right)$$

The drag 'command' is then limited to be positive and at least as large as one fifth the magnitude of lift, since L/D max for an RLV on final approach is typically no more than five. Next, the lift command is converted to negative z axis acceleration command by the formula

$$N_z = L/W - 1/\cos\phi$$

The negative z axis acceleration command is then limited to 0.6g. Finally, commanded drag is converted to a speedbrake position command using a rough aerodynamic model that was obtained from a multiple regression of X-33 simulation data for the final approach phase immediately before and after speed brake opening. The drag coefficient was determined to be approximately

$$\begin{aligned} C_D &\approx -0.494C_L^2 + 0.308C_L - 0.142M \\ &+ 0.00140\delta_{SB} + 0.1368 \end{aligned} \qquad (17)$$

This equation is valid for mach numbers $0.4 \leq M \leq 0.6$, lift coefficients $0.12 \leq C_L \leq 0.4$, and speed brake commands in the range $12 \leq \delta_{SB} \leq 29$. Inverting Eq (17), and limiting the speed brake command to $0 \leq \delta_{SB} \leq 20$ results in a valid speedbrake command.

TABLE I

FINAL GAIN VALUES

Channel	K_{pi}	K_{Ii}	K_{di}	Anti-Windup
Bank	7.592	0.140	8.596	0.328
Altitude	0.0	0.001	0.063	0.0421
Speed	2.259	0.134	0.0	0.0

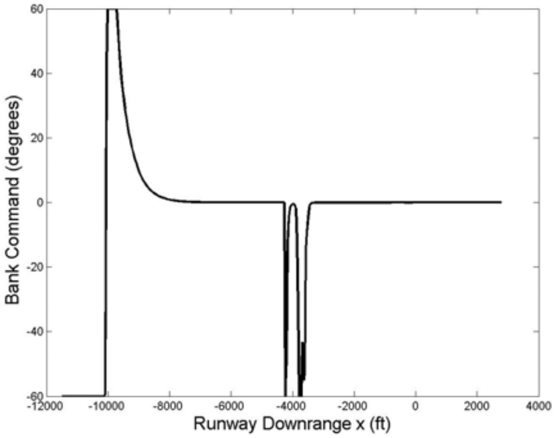

Fig. 6. Example Bank Commands for RLV Approach and Landing.

V. RESULTS

The control scheme described in Section IV and the RLV Dynamics described in Section II were implemented in a Simulink simulation. The block diagram is shown in Figures 1 and 2. Figure 2 shows the internal workings of the guidance block. The results shown are based on considering the bank command saturated at $|\phi| = 60°$. Negative Z axis acceleration is saturated when $|N_z| = 0.6$. The Speedbrake command is limited to $0 \leq \delta_{SB} \leq 20°$. The feedback paths immediately prior to and after the saturation blocks are used to implement integrator anti-windup after a scheme presented by Franklin [10]. The MATLAB function block labeled guidance implements the mathematics shown from Eq. 16 through Eq. 17 including inversion of the simplified aero model.

The simulation options were set to use a 4th order Runge-Kutta integration scheme with a fixed time step of 0.1 sec. This was done to simulate the current 10Hz guidance update rate implemented in the MAVERiC X-33 simulation, and to keep computation time and precision within reason.

Because of the large non-linearity between pseudo-control and saturation blocks, the pseudo control gains K_{pi}, K_{di}, integral gains $\mathbf{K}_I$, and anti-windup gains K_{ab}, K_{aNz} were determined by genetic algorithm tuning of the simulation itself. The genetic algorithm was implemented in Matlab, by running the trajectory simulation repeatedly with various gains. The relative performance of each set of gains was determined by the sum of 2-norms of error signals of y, h and V, plus a scaling factor q times the 2-norm of the bank command. That is

$$J = \|\mathbf{y}_{err}\|_2 + \|\mathbf{h}_{err}\|_2 + \|\mathbf{V}_{err}\|_2 + q\,\|\phi\|_2$$

where $\mathbf{y}_{err}$, $\mathbf{h}_{err}$ and $\mathbf{V}_{err}$ are vectors containing each error signal for the duration of the simulation. The bank command is included in the cost function to reduce oscillations across extended runway centerline.

The genetic algorithm used a population size of 30, mutation probability of 3%, and tournament selection of the mating pool. The best gains after 50 generations of tuning are shown in Table 1.

TABLE II

INITIAL CONDITIONS FOR GUIDANCE TESTING

Variable	IC 1	IC 2	IC 3	IC 4
x (ft)	-11500	-11500	-11500	-11500
y (ft)	76.7	76.7	-100	0.0
h (ft AGL)	5000	4560	4800	4800
ψ (rad)	0.153	-0.153	0.0	-0.153
γ (rad)	-0.44	-0.44	-0.3	-0.4
V (ft/s)	480	512	500	500

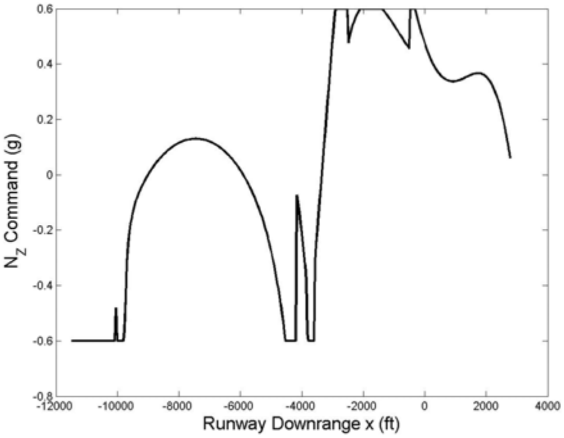

Fig. 7. Example N_z Commands for RLV Approach and Landing.

Figures 3 through 8 show typical performance of the feedback linearization guidance. The initial conditions tested are shown in Table 2. As shown in Figure 3, the proposed system guides the RLV from a wide dispersion in initial crossrange positions and headings back to the runway centerline with minimal oscillation. Figure 4 shows that for a small dispersion in initial altitude, the system guides the RLV to a successful flare. Only case four results in an unacceptably low runway threshold crossing height. This is also the case with the shortest groundtrack, and hence the highest initial energy. It is clear in Figure 4 that the guidance dumps this extra energy, but is unable to pull up early enough for a satisfactory landing. In fact, the GA optimization balances tracking error between crossrange, altitude and airspeed as shown in above. Figure 5 shows the corresponding forward velocity profiles. The bank commands for a typical approach are shown in Figure 6. Here we show a single approach for clarity. The command is initially saturated, but the amount of bank commanded shrinks substantially as the RLV gets close to the runway threshold. Figure 7 shows the negative z axis acceleration command profile for a typical approach. The command is initially saturated in order to correct to glidepath. It is also saturated during the flare pull up. Finally, Figure 8 shows the speedbrake commands for a typical trajectory. Although the command saturates several times, there is no apparent 'fanning' of the speedbrake.

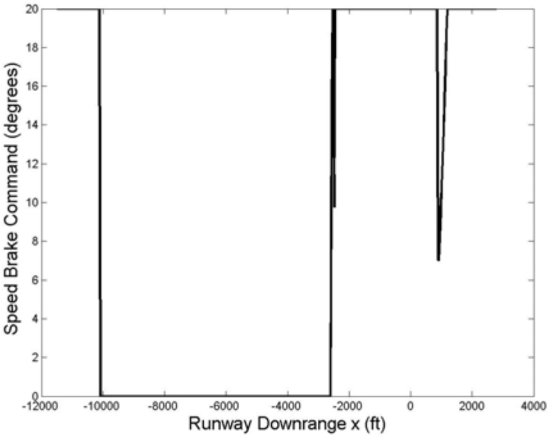

Fig. 8. Example Speedbrake Commands for RLV Approach and Landing.

VI. CONCLUSIONS

This paper has shown a technique of automatically generating the guidance commands for approach and landing of an RLV. Feedback Linearization was applied to the vehicle point mass dynamics, resulting in a linear system with inputs that are combinations of lift, drag and bank angle. By applying a simple aerodynamic model, lift and drag are mapped to negative z axis acceleration and speedbrake commands. A genetic algorithm was applied to determine the best possible pseudo-control gains in the presence of actuator saturation and integrator anti-windup. The system provides satisfactory performance for a wide dispersion of initial headings, crossrange positions, and airspeeds. The results presented here are based on a low-fidelity model and thus should be considered preliminary.

REFERENCES

[1] Barton, G. H., "Autolanding Trajectory Design for the X-34," AIAA Paper No 99-4161, 1999.

[2] Barton, G. H., "New Methodologies for Assessing the Robustness of the X-34 Autolanding Trajectories," *Advances in the Astronautical Sciences*, Volume 107 Guidance and Control 2001, pp. 193-214.

[3] Girerd, A., Barton, G., "Next Generation Enty Guidance–Onboard Trajectory Generation for Unpowered Drop Tests," AIAA Paper No 2000-3960, 2000.

[4] Grantham, K. "Adaptive Critic Neural Network Based Terminal Area Energy Management / Entry Guidance," AIAA Paper 2003-305.

[5] Schmitt, C. A., and Burchett, B. T., "Fuzzy IPPD Guidance for Approach and Landing of a Reusable Launch Vehicle," AIAA Paper 2004-6254, 2004.

[6] Schierman, J. D., Ward, D. G., Hull, J. R., and Gandhi, N., "Intelligent Guidance and Trajectory Command Systems for Autonomous Space Vehicles," AIAA Paper 2004-6253, 2004.

[7] Slotine, J. E., and Li, W., *Applied Nonlinear Control*, Prentice Hall, Englewood Cliffs, N.J., 1991.

[8] Vinh, N. X., *Optimal Trajectories in Atmospheric Flight*, Elsevier, New York, 1981, ch 3.

[9] Kuo, B. C., and Golnaraghi, F., *Automatic Control Systems*, 8th Ed., Wiley, New York, 2003.

[10] Franklin, G. F., Powell, J. D., and Emami-Naeini, A., *Feedback Control of Dynamic Systems*, 4th Ed., Prentice Hall, Upper Saddle River, New Jersey, 2002.