NEW GUIDANCE LAW AND AUTOPILOT DESIGN FOR LAUNCH ROCKETS

Chia-Hao Cheng [*] **Fu-Kuang Yeh** [*]
Li-Chen Fu, *IEEE Fellow* [**]

[*] *Department of Electrical Engineering,*
[**] *College of Electrical Engineering and Computer Science,*
National Taiwan University, Taipei, Taiwan, R.O.C.

Abstract: In this paper, the decision parts of a launch rocket—guidance law (GL) and autopilot—are discussed. In the controller design, the differential geometry and sliding mode control are employed. In addition to the individual GL and autopilot system, stability of the integrated guidance/autopilot (G/A) system are discussed. Finally, various simulations are demonstrated to examine the feasibility and performance of the rocket's integrated G/A system.

Keywords: Guidance, Autopilot, Differential Geometry, Sliding Mode, Rockets

1. INTRODUCTION

The control of flight vehicles has been an important topic ever since the range of human activities was extended from the earth to the universe. It becomes even more influential to mankind as the modern technology progresses, such as invention of the rockets that transport spacecrafts and satellites beyond the sky. In this paper, the controller design of trajectory-following launch rockets are discussed. Specially, GL and autopilot systems are the two control units in which we are interested among whole rocket system. Conventionally, the autopilot system design of spacecrafts and aircrafts are based on the optimal control theory and the systemical discussion can be found in (Bryson, 1994). Yet, inventive work that utilize adaptive control and neural network control on the G/A of autonomous launch vehicles is proposed by (Johnson *et al.*, 2001). Nevertheless, the rocket autopilot systems with quaternion representation and sliding mode control design in our previous research works (Yeh, 2003) and (Cheng, 2004) have also acquired excellent performance, and are reinvented in this paper.

In this paper, new GL and autopilot systems are proposed. The utilization of differential geometry and sliding mode theory in GL is designed for the launch rocket to follow the preset trajectory. To construct a whole-phase-single-strategy rocket autopilot system that is suitable in both endo- and exo-atmosphere, we apply the sliding mode theory and use a wingless rocket airframe based on TVC/DCS (Thrust Vector Control/Divert Control System). In addition to the respective GL and autopilot subsystems, the integrated G/A systems are discussed, and the design methodologies of the system parameters are suggested. Various simulations including aerodynamic model are demonstrated at the last to verify the feasibility and test the performance of the rocket's integrated G/A systems.

2. NEW GL FOR LAUNCH ROCKETS

3-D Geometry Modelling Generally speaking, GLs and autopilot systems are both the strategic parts in an airframe; decisions and control commands are made by them. GLs guide the airframe's directions based on the current state

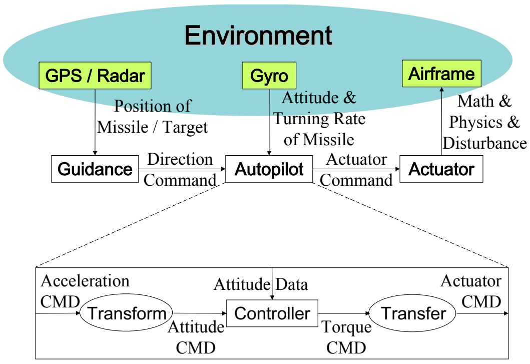

Fig. 1. Subsystems of an Airframe

of the airframe and the circumstances, while autopilot systems which depend on a well-designed controller follow the guidance commands stably. The illustrative and conceptual flowchart of the commands in an airframe is shown in Fig. 1, and the GL system is discussed in this section. The GL is designed for a transport rocket to trace a preset optimal launching trajectory from the earth to the space. The preset optimal launching trajectory is characterized by a set of polynomial in three dimensions:

$$\mathbf{r}(\tau) = \begin{bmatrix} x(\tau) & y(\tau) & z(\tau) \end{bmatrix}^T \tag{1}$$

Based on the differential geometry (Goetz, 1970), we can find the unit tangent vector $\mathbf{t}$, unit normal vector $\mathbf{n}$, unit binormal vector $\mathbf{b}$, and curvature κ of the preset trajectory as follows:

$$\begin{aligned} \mathbf{t}(\tau) &= \frac{\dot{\mathbf{r}}}{|\dot{\mathbf{r}}|} & \mathbf{n}(\tau) &= \mathbf{b}\times\mathbf{t} \\ \mathbf{b}(\tau) &= \frac{\dot{\mathbf{r}}\cdot\ddot{\mathbf{r}}}{|\dot{\mathbf{r}}\cdot\ddot{\mathbf{r}}|} & \kappa(\tau) &= \frac{|\dot{\mathbf{r}}\times\ddot{\mathbf{r}}|}{|\dot{\mathbf{r}}|^3} \end{aligned} \tag{2}$$

The proposed strategy for a rocket to follow a trajectory is firstly to find the $\mathbf{n}(\tau') - \mathbf{b}(\tau')$ plane that includes the current position of the rocket $\boldsymbol{r}_R(t)$, then minimizing the distance error $\boldsymbol{r}(t) = \mathbf{r}(\tau') - \boldsymbol{r}_R(t)$, as shown in Fig. 2. In other words, the rocket $\boldsymbol{r}_R$ is supposed to track the desired point $\boldsymbol{r}_D = \mathbf{r}(\tau')$ on the $\mathbf{n}$-$\mathbf{b}$ plane of the moving orthogonal coordinate $\mathbf{t}$-$\mathbf{n}$-$\mathbf{b}$. To do so, we begin with finding the dynamic equations of distance error, velocity error, and acceleration error:

$$\begin{aligned} \boldsymbol{r} = {} & \boldsymbol{r}_D - \boldsymbol{r}_R && = \mathbf{r}(\tau') - \boldsymbol{r}_R && (3) \\ \dot{\boldsymbol{r}} = \boldsymbol{v} = {} & \boldsymbol{v}_D - \boldsymbol{v}_R && = (\boldsymbol{v}_R\cdot\mathbf{t})\mathbf{t} - \boldsymbol{v}_R && \\ = {} & \boldsymbol{v}_{R\|} - \boldsymbol{v}_R && = -\boldsymbol{v}_{R\perp} && (4) \\ \dot{\boldsymbol{v}} = \boldsymbol{a} = {} & \boldsymbol{a}_D - \boldsymbol{a}_R && && \\ = {} & \kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} + (\boldsymbol{a}_R\cdot\mathbf{t})\mathbf{t} - \boldsymbol{a}_R && && \\ = {} & \kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} + \boldsymbol{a}_{R\|} - \boldsymbol{a}_R && && \\ = {} & \kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} - \boldsymbol{a}_{R\perp} && && (5) \end{aligned}$$

In (3), the desired point $\mathbf{r}(\tau')$ is exactly the projection of the rocket position $\boldsymbol{r}_R(x, y, z)$ on the preset trajectory $\mathbf{r}(\tau)$, and it is be determined by the parameter τ' that can be calculated by:

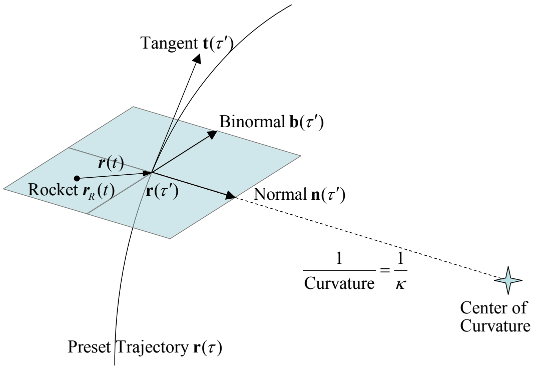
Tangent $\mathbf{t}(\tau')$
Binormal $\mathbf{b}(\tau')$
$\boldsymbol{r}(t)$
Rocket $\boldsymbol{r}_R(t)$
$\mathbf{r}(\tau')$
Normal $\mathbf{n}(\tau')$
$\frac{1}{\text{Curvature}} = \frac{1}{\kappa}$
Center of Curvature
Preset Trajectory $\mathbf{r}(\tau)$

Fig. 2. Geometry of Trajectory-Following Rocket

$$\tau' = \left\{\tau \middle| \left[\mathbf{r}(\tau) - \boldsymbol{r}_R\right]\cdot\dot{\mathbf{r}}(\tau) = 0\right\} \tag{6}$$

In (4), we avoid directly computing the time derivative of $\boldsymbol{r}_D$ due to the inevitable time delay and resultant numerical error. Instead, we use the fact that $\boldsymbol{v}_D$, the velocity of the projection point, is identical to the projection of the rocket velocity on $\mathbf{t}$, i.e. $(\boldsymbol{v}_R\cdot\mathbf{t})\mathbf{t}$. Since it is exactly the component of $\boldsymbol{v}_R$ parallel to $\mathbf{t}$, we define $\boldsymbol{v}_{R\|} = (\boldsymbol{v}_R\cdot\mathbf{t})\mathbf{t}$ and $\boldsymbol{v}_{R\perp} = \boldsymbol{v}_R - \boldsymbol{v}_{R\|}$, the component of $\boldsymbol{v}_R$ perpendicular to $\mathbf{t}$ and on the $\mathbf{n}$-$\mathbf{b}$ plane. Similarly, we also avoid computing the time derivative of $\boldsymbol{v}_D$ directly in (5). We instead use the fact that $\boldsymbol{a}_D$, the acceleration of the projection point, is equivalent to the summation of $(\boldsymbol{a}_R\cdot\mathbf{t})\mathbf{t}$, the projection of the rocket acceleration on $\mathbf{t}$, and the normal acceleration of the trajectory. The latter is determined by its radius of curvature and the component of rocket velocity in $\mathbf{n}$ direction, i.e.

$$\frac{|\boldsymbol{v}_{R\|}|^2}{\frac{1}{\kappa}}\mathbf{n} = \kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} \tag{7}$$

Likewise, we also define $\boldsymbol{a}_{R\|} = (\boldsymbol{a}_R\cdot\mathbf{t})\mathbf{t}$ and $\boldsymbol{a}_{R\perp} = \boldsymbol{a}_R - \boldsymbol{a}_{R\|}$. Divide the acceleration of the rocket into components as:

$$\boldsymbol{a}_R = \boldsymbol{g}_R + \tfrac{\boldsymbol{F}_R}{m} = \boldsymbol{g}_R + \tfrac{\dot{m}}{m}\boldsymbol{v}_R + \boldsymbol{a}_{Rp}\ , \tag{8}$$

where $\boldsymbol{g}_R$ is the gravity, $\boldsymbol{F}_R$ is the propellent force, and $\boldsymbol{a}_{Rp}$ is the acceleration due to the propellant of the rocket and variation of mass of the rocket. Accordingly, (5) becomes:

$$\boldsymbol{a} = \kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} - \boldsymbol{g}_{R\perp} - \tfrac{\dot{m}}{m}\boldsymbol{v}_{R\perp} - \boldsymbol{a}_{Rp\perp}\ , \tag{9}$$

where the subscript $\perp$ represents the components on the $\mathbf{n}$-$\mathbf{b}$ plane, i.e. the components perpendicular to $\mathbf{t}$. Notice that $\boldsymbol{a}$, $\boldsymbol{v}$, and $\boldsymbol{r}$ are all on the same $\mathbf{n}$-$\mathbf{b}$ plane.

Guidance Laws Design Applying the sliding mode control theory, we define the sliding surface variable as

$$S_g = \boldsymbol{v} + \lambda\boldsymbol{r}\ , \tag{10}$$

where λ is a 3×3 diagonal P.D. matrix, and S_g is a 3×1 vector. Since $\dot{\boldsymbol{r}} = \boldsymbol{v}$, thus $S_g = 0$ implies $\{\boldsymbol{r}\,,\,\boldsymbol{v}\} \to 0$ exponentially. To design a controller capable of leading the convergence to S_g, we choose the Lyapunov function candidate

$$V_g(S_g) = \tfrac{1}{2}\,S_g^T S_g \tag{11}$$

Taking the time derivative of V_g , we have:

$$\dot{V}_g = S_g^T\big(\kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} - \boldsymbol{g}_{R\perp} - \tfrac{\dot{m}}{m}\boldsymbol{v}_{R\perp} - \boldsymbol{a}_{Rp\perp} \\ - \lambda\boldsymbol{v}_{R\perp} + d_g\big) \ , \qquad (12)$$

where the disturbance $d_g^{(3\times1)}$ is specially taken into consideration, which comes from both parametric uncertainties and unmodelled dynamics. Parametric uncertainties might be caused by the estimated errors of position, velocity, mass, variance of mass, propellent force, etc., whereas unmodelled dynamics could be aerodynamics, parasitic dynamics, etc. Without loss of generality, we assume that there exists an upper bound d_g^{max} on the disturbance d_g , i.e. $|d_g| \leq d_g^{max}$. Design the control law as:

$$\boldsymbol{a}_{Rp\perp} = \kappa|\boldsymbol{v}_{R\|}|^2\mathbf{n} - \boldsymbol{g}_{R\perp} - \frac{\dot{m}}{m}\boldsymbol{v}_{R\perp} - \lambda\boldsymbol{v}_{R\perp} \\ +(d_g^{max} + \eta_g)sgn(S_g) \ , \qquad (13)$$

where $\eta_g^{(3\times1)} > 0$ is an adjustable vector variable, and $sgn(S_g) = [sgn(S_{g1}),\ sgn(S_{g2}),\ sgn(S_{g3})]^T$. With the designed control law, we get:

$$\dot{V}_g = S_g^T d_g - S_g^T d_g^{max} sgn(S_g) - S_g^T \eta_g sgn(S_g) \\ \leq -\eta_g^T \,|\, S_g| \qquad (14)$$

Accordingly, the system trajectories will reach the sliding surface $S_g = 0$ in finite time less than $\max_i \frac{|S_{gi}(t_0)|}{\eta_g}$, remain on it, and then are constrained by the sliding surface itself, namely $S_g = \boldsymbol{v} + \lambda\boldsymbol{r}$, which implies the exponential convergence of distance error $\boldsymbol{r}$ and velocity error $\boldsymbol{v}$. Knowing that $|\boldsymbol{a}_M| = \frac{N}{m}$, $\boldsymbol{a}_{Rp\|} = (\boldsymbol{a}_{Rp}\cdot\mathbf{t})\mathbf{t}$, and $\boldsymbol{a}_{Rp\perp} = \boldsymbol{a}_{Rp} - \boldsymbol{a}_{Rp\|}$, we have

$$\boldsymbol{a}_{Rp\|} = \mathbf{t}\sqrt{\left(\tfrac{N}{m}\right)^2 - |\boldsymbol{a}_{Rp\perp}|^2} \qquad (15)$$

Hence, the desired acceleration command is

$$\boldsymbol{a}_d = \boldsymbol{a}_{Rp\perp} + \boldsymbol{a}_{Rp\|} \ , \qquad (16)$$

and is delivered to the autopilot system described in next section.

3. AUTOPILOT DESIGN AND ANALYSIS OF INTEGRATED G/A SYSTEM

The autopilot systems and the analysis of integrated G/A system are discussed in this section. An autopilot system is a mechanism which receives the acceleration commands from GLs and controls the actuators to stabilize the airframe while performing those commands. In this paper, the autopilot system is divided into three parts for design as in Fig. 1, but the actuator dynamics is excluded from our topics. Fig. 3 shows the architecture of the rocket, where TVC is a mechanism realized by a moveable tail nozzle, and DCS is an additional auxiliary propellent system mounted on lateral of the airframe. Both TVC and DCS have the advantage of maneuvering ability in both endo- and exo-atmosphere, whereas control surfaces work only in endoatmosphere. With this reason and a motivation to design a whole-phase-single-strategy controller, we choose an airframe without control surface, which dominates the aerodynamic force. By this, the designed whole-phase-single-strategy controller will be more compatible with the circumstance where less aerodynamic influence on the airframe in both endo- and exo-atmosphere is expected. Based on physics laws, the kinematic equations of the airframe are composed of those of both TVC and DCS, that is

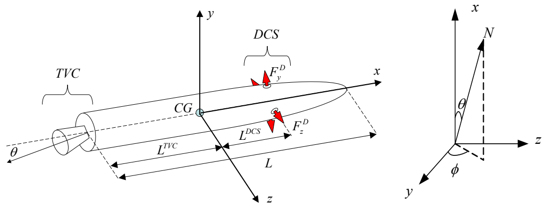

Fig. 3. (a) Rocket Configuration (b) TVC Moving Angles

$$\boldsymbol{F}_b^{prop} = \boldsymbol{F}_b^{TVC} + \boldsymbol{F}_b^{DCS} = \begin{bmatrix} N\cos\theta \\ N\sin\theta\cos\phi + F_y^D \\ N\sin\theta\sin\phi + F_z^D \end{bmatrix}$$

$$\boldsymbol{\tau}_b^{prop} = \boldsymbol{\tau}_b^{TVC} + \boldsymbol{\tau}_b^{DCS} \qquad (17)$$

$$= \begin{bmatrix} 0 \\ N\,l^{TVC}\sin\theta\sin\phi - l^{DCS}F_z^D \\ -N\,l^{TVC}\sin\theta\cos\phi + l^{DCS}F_y^D \end{bmatrix} \qquad (18)$$

Translate Acceleration CMDs into Attitude CMDs The "Transform" part in Fig. 1 transforms the acceleration commands from a GL into the attitude commands. In Fig. 4, the vector of the desired acceleration command $\boldsymbol{a}_d$ generated from (16) is produced by the constant main propellant TVC, and the directions of y_b and z_b are the "don't care" factors for this cylindrical, non-aerodynamically maneuvering airframe. To interpret the attitude commands, we firstly set a reference coordinate frame, and then find the relationship between the reference coordinate frame and the acceleration

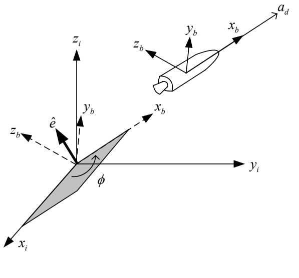

Fig. 4. Desire Quaternion

command vector. According to the Euler's rotation theory, there exists a vector $\hat{e}$ and an angle ϕ such that x_i will coincide with x_b after undergoing an rotation of the angle ϕ by the vector $\hat{e}$. The vector $\hat{e}$ and the angle ϕ can be obtained by:

$$\hat{e} = \tfrac{x_i \times x_b}{|x_i \times x_b|} = \tfrac{x_i \times a_d}{|x_i \times a_d|} \tag{19}$$

$$\phi = \cos^{-1}\left(\tfrac{x_i \cdot x_b}{|x_i||x_b|}\right) = \cos^{-1}\left(\tfrac{x_i \cdot a_d}{|x_i||a_d|}\right) \ , \tag{20}$$

which can be used to represent the desired attitude command in quaternion representation as described in the following subject.

Translate Attitude CMDs into Torque CMDs The "Controller" part in Fig. 1 generated the torque commands for the desired attitude commands. The attitude is expressed by quaternion q and angular velocity $\boldsymbol{\omega}$; $(q_e,\, \boldsymbol{\omega}_e)$ are defined as the errors between desired values $(q_d,\, \boldsymbol{\omega}_d)$ and current values $(q,\, \boldsymbol{\omega})$ as follows:

$$\begin{bmatrix} q_{e1} \\ q_{e2} \\ q_{e3} \\ q_{e4} \end{bmatrix} = \begin{bmatrix} q_{d4} & q_{d3} & -q_{d2} & -q_{d1} \\ -q_{d3} & q_{d4} & q_{d1} & -q_{d2} \\ q_{d2} & -q_{d1} & q_{d4} & -q_{d3} \\ q_{d1} & q_{d2} & q_{d3} & q_{d4} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix}$$

$$\boldsymbol{\omega}_e = \boldsymbol{\omega} - \boldsymbol{\omega}_d \ , \tag{21}$$

where the quaternion is defined as:

$$q = \begin{bmatrix} \boldsymbol{q} \\ q_4 \end{bmatrix} = [\, q_1 \ q_2 \ q_3 \ q_4\,]^T = \begin{bmatrix} \hat{e}\sin(\phi/2) \\ \cos(\phi/2) \end{bmatrix}, \tag{22}$$

where the unit vector $\hat{e} = [e_1 \ e_2 \ e_3]^T$ and angle ϕ are the Euler Axis and Angle exactly. By (22), q_d can be computed from (19) and (20), and in turn yield the desired angular velocity by the kinematic equation of quaternion:

$$\boldsymbol{\omega}_d = 2 \begin{bmatrix} q_{d4} & q_{d3} & -q_{d2} & -q_{d1} \\ -q_{d3} & q_{d4} & q_{d1} & -q_{d2} \\ q_{d2} & -q_{d1} & q_{d4} & -q_{d3} \end{bmatrix} \dot{q}_d \tag{23}$$

To generate the torque commands that eliminate the attitude errors q_e and $\boldsymbol{\omega}_e$, a controller design based on sliding mode control theory is proposed. Define the sliding surface variable as

$$S_a = P\boldsymbol{q}_e + \boldsymbol{\omega}_e \ , \tag{24}$$

where P is a 3×3 P.D. matrix, whereas S_a, $\boldsymbol{q}_e$, and $\boldsymbol{\omega}_e$ are 3×1 vectors. Choose the Lyapunov function candidate as

$$V_a(S_a) = \tfrac{1}{2}\, S_a^T J\, S_a \ , \tag{25}$$

where J is the inertia matrix of the airframe. Taking the time derivative of V_a , we have:

$$\dot{V}_a = S_a^T \left(\tfrac{1}{2}\dot{J} S_a + JP\dot{\boldsymbol{q}}_e + J\dot{\boldsymbol{\omega}} - J\dot{\boldsymbol{\omega}}_d \right) \tag{26}$$

Applying the kinematic equation of quaternion

$$\dot{\boldsymbol{q}}_e = \tfrac{1}{2}\, \boldsymbol{q}_e \times \boldsymbol{\omega}_e + \tfrac{1}{2}\, q_{e4}\boldsymbol{\omega}_e \tag{27}$$

and the Euler's equation

$$J\dot{\boldsymbol{\omega}} = \boldsymbol{\tau}_b - \dot{J}\boldsymbol{\omega} - \boldsymbol{\omega} \times (J\,\boldsymbol{\omega}) \ , \tag{28}$$

we then have:

$$\begin{aligned} \dot{V}_a = S_a^T \big(& \frac{1}{2}\dot{J} S_a + JP(\frac{1}{2}\, \boldsymbol{q}_e \times \boldsymbol{\omega}_e + \frac{1}{2}\, q_{e4}\boldsymbol{\omega}_e) \\ & + \boldsymbol{\tau}_b - \dot{J}\boldsymbol{\omega} - \boldsymbol{\omega} \times (J\,\boldsymbol{\omega}) - J\dot{\boldsymbol{\omega}}_d + d_a \big), \end{aligned} \tag{29}$$

where the torque command $\boldsymbol{\tau}_b$ is the control input of this error tracking system, and the disturbance $d_a^{(3\times1)}$ is also taken into consideration as in GL, including aerodynamics, actuator dynamics, parasitic dynamics, etc. Assume that there exists an upper bound d_a^{max} on the disturbance d_a , i.e. $|d_{ai}| \leq d_{ai}^{max}$, i=1, 2, 3, and design the control law:

$$\begin{aligned} \boldsymbol{\tau}_{Db} = & -\tfrac{1}{2}\dot{J} S_a - JP\big(\tfrac{1}{2}\, \boldsymbol{q}_e \times \boldsymbol{\omega}_e + \tfrac{1}{2}\, q_{e4}\boldsymbol{\omega}_e\big) + \dot{J}\boldsymbol{\omega} \\ & + \boldsymbol{\omega} \times (J\,\boldsymbol{\omega}) + J\dot{\boldsymbol{\omega}}_d - (d_a^{max} + \eta_a)\, sgn(S_a), \end{aligned} \tag{30}$$

where $\eta_a^{(3\times1)} > 0$ is an adjustable vector variable, and $sgn(S_a) = [sgn(S_{a1}),\, sgn(S_{a2}),\, sgn(S_{a3})]^T$. With the designed control law, we get:

$$\begin{aligned} \dot{V}_a & = S_a^T d_a - S_a^T d_a^{max} sgn(S_a) - S_a^T \eta_a sgn(S_a) \\ & \leq -\eta_a^T \,|\, S_a| \end{aligned} \tag{31}$$

Accordingly, the system trajectories will reach the sliding surface $S_a = 0$ in finite time less than $\max_i \frac{|S_{ai}(t_0)|}{\eta_a}$, remain on it, and then are constrained by the sliding surface itself, namely

$$S_a = P\boldsymbol{q}_e + \boldsymbol{\omega}_e = 0 \tag{32}$$

Now, we shall verify that $\boldsymbol{\omega}_e$ and $\boldsymbol{q}_e$ will converge to zero when the above condition is reached. Define the Lyapunov function candidate as

$$V_q(\boldsymbol{q}_e) = \boldsymbol{q}_e^T \boldsymbol{q}_e \tag{33}$$

Take the time derivative of V_q and use (32):

$$\dot{V}_q = q_{e4}\, \boldsymbol{q}_e^T \boldsymbol{\omega}_e = -q_{e4}\, \boldsymbol{q}_e^T P\boldsymbol{q}_e \tag{34}$$

Substitute (32) into the kinematic equation of quaternion:

$$\dot{q}_{e4} = -\tfrac{1}{2}\, \boldsymbol{q}_e \cdot \boldsymbol{\omega}_e = \tfrac{1}{2}\, \boldsymbol{q}_e^T P\boldsymbol{q}_e \ \geq 0 \ , \tag{35}$$

which implies that q_{e4} is a non-decreasing positive variable. Define the domain of rotation angle ϕ to be between $-\pi$ and π, which implies a non-negative q_{e4}, to avoid the sign ambiguity in the representation of quaternion. Thus $q_{e4} = \cos(\frac{\phi}{2}) \geq 0$. If we select a nonzero $q_{e4}(t_0)$, which is determined by the designable initial conditions, we have $0 < q_{e4}(t_0) \leq q_{e4}(t)$, $\forall\, t \geq 0$. Substituting it into (34), we have

$$\dot{V}_q \leq -q_{e4}(t_0)\boldsymbol{q}_e^T P\boldsymbol{q}_e < 0 \ , \tag{36}$$

i.e. $V_q(\boldsymbol{q}_e)$ is P.D., and $\dot{V}_q(\boldsymbol{q}_e)$ is N.D. In addition, $V_q(\boldsymbol{q}_e) \to \infty$ as $|\boldsymbol{q}_e| \to \infty$. According to the Lyapunov direct method, the equilibrium point $\boldsymbol{q}_e = 0$ is globally asymptotically stable. Combined with (32), the asymptotical stability of $\boldsymbol{q}_e$ also implies the asymptotical stability of $\boldsymbol{\omega}_e$.

In summary, with the control law (30) the system trajectories will reach the sliding surface (32) in finite time less than $\max_i \frac{|S_{ai}(t_0)|}{\eta_a}$, remain on it, and it follows the asymptotic convergence of the attitude error $\boldsymbol{q}_e$ and angular velocity error $\boldsymbol{\omega}_e$.

Translate Torque CMDs into Actuator CMDs The "Transfer" part in Fig. 1 is designed to calculate the actuator commands that can reach the desired torque commands. It is interesting to note that a TVC system generates an unpleasantly accompanied lateral force while it is trying to generate a rotational torque by moving the nozzle. This accompaniment, however, can be easily eliminated by the addition of the auxiliary DCS. In other words, the cooperative strategy between TVC and DCS ensures a rotational torque without lateral force accompanied, which is a proper way for an airframe to change the direction of heading. To do so, we simply let the summation of the last two terms of (17) be zero, i.e.,

$$F_y^D = -N \sin\theta \cos\phi \ ; \ F_z^D = -N \sin\theta \sin\phi \ , \qquad (37)$$

and thus get the desired relationship between TVC moving angles and the required DCS propulsion. Substituting them into (18), we have:

$$\tau_{by} = \left(N\, l^{TVC} + N\, l^{DCS}\right) \sin\theta \sin\phi \qquad (38)$$

$$\tau_{bz} = -\left(N\, l^{TVC} + N\, l^{DCS}\right) \sin\theta \cos\phi \ , \qquad (39)$$

which are the rotational torque generated by the cooperation of the actuators TVC and DCS. To attain the desired torque commands τ_{Db} from (30), the required actuator commands by reformulating (38) and (39) should be:

$$\sin\theta_D = \frac{\sqrt{\tau_{Dby}^2 + \tau_{Dbz}^2}}{N\, l^{TVC} + N\, l^{DCS}} \qquad (40)$$

$$\sin\phi_D = \frac{\tau_{Dby}}{\sqrt{\tau_{Dby}^2 + \tau_{Dbz}^2}} \quad ; \quad \cos\phi_D = \frac{-\tau_{Dbz}}{\sqrt{\tau_{Dby}^2 + \tau_{Dbz}^2}}$$

Substituting these required TVC moving angles into (37), we can obtain the correspondingly required DCS propulsion:

$$F_y^D = \frac{\tau_{Dbz}}{l^{TVC} + l^{DCS}} \quad ; \quad F_z^D = \frac{-\tau_{Dby}}{l^{TVC} + l^{DCS}} \qquad (41)$$

In (40), $0 \le \phi_D \le 2\pi$ and $0 \le \theta_D \le \theta_M = \sin^{-1} k$, the maximum limit of TVC moving angle. The following saturation function is used to modify the torque commands if the desired actuator angle θ_D exceed the its limit θ_M:

$$\tau_{Dbi} = \tau_{Dbi} \ , \text{ if} \qquad (42)$$

$$\sqrt{(\tau_{Dby})^2 + (\tau_{Dbz})^2} \le kN \left(l^{TVC} + l^{DCS}\right)$$

$$\tau_{Dbi} = \frac{\tau_{Dbi}}{\sqrt{\left(\tau_{Dby}\right)^2 + (\tau_{Dbz})^2}}\, kN\left(l^{TVC} + l^{DCS}\right) \ , \text{ o.w.}$$

where the subscript $i = y$ or z.

Integrated G/A System Analysis While deriving the GL, we assume having a perfect autopilot system to execute the guidance commands and thus fulfill the aimed goal. The assumption is unrealistic though, and thus the the stability analysis should be re-derived for an integrated G/A system with an imperfect autopilot subsystem. Recall from (16) that $\boldsymbol{a}_d = \boldsymbol{a}_{Rp\perp} + \boldsymbol{a}_{Rp\parallel}$, the desired acceleration command given by GL. For an imperfect autopilot system, the actual acceleration output is defined as $\boldsymbol{a}_c$ rather than $\boldsymbol{a}_d$, and the relationship between them is found in (Cheng, 2004), where the final result is:

$$\boldsymbol{a}_{c\perp} = (1 + E)\, \boldsymbol{a}_{Rp\perp} \qquad (43)$$

If the control $|\boldsymbol{a}_{Rp\perp}| \neq 0$, and we let P in $S_a = \boldsymbol{\omega}_e + P\boldsymbol{q}_e$ to be $P = p\, I_{3\times3}$, then the bound of E is found as:

$$|E| \le \tfrac{1}{p}\, E_A \ , \qquad (44)$$

where E_A is a bounded positive scalar. The control $|\boldsymbol{a}_{Rp\perp}| = 0$ if the situation is satisfying and no further control is needed. Define the Lyapunov function candidate of the integrated G/A system:

$$V = \tfrac{1}{2}\, S_g^T S_g + \tfrac{1}{2}\, S_a^T J\, S_a \ , \qquad (45)$$

Substitute $\boldsymbol{a}_{c\perp}$ for $\boldsymbol{a}_{Rp\perp}$ in (9), it is found in (Cheng, 2004) that

$$\dot{V} \le -\eta_g^T |S_g| + S_g^T \left(d_g + \frac{E_A E_B}{p}\right) - d_g^{max} |S_g| - \eta_a^T\, |\, S_a| \ , \qquad (46)$$

where E_B is a bounded positive scalar. If we choose a large enough p and let d_g^{max} to be

$$d_g^{max} \ge |d_g| + \frac{E_A E_B}{p} \qquad (47)$$

Then we have:

$$\dot{V} \le -\eta_g^T |S_g| - \eta_a^T\, |\, S_a| \qquad (48)$$

is N.D. In addition, $V \to \infty$ as $\left|[x^T; S_a^T]\right| \to \infty$. According to the Lyapunov direct method, we can therefore conclude that with the suggested system parameters the overall integrated G/A system is globally asymptotically stable.

4. SIMULATION AND ANALYSIS

In this section, we examine the performance of the rocket integrated G/A system. In this simulation, we referred to actual information from the launch rocket Taurus, which recently transported the Taiwanese Satellite II in June, 2004. The launch rocket Taurus consists four stages, and the Stage IV starts igniting about $8min$ after the Stage III had been burnout. In other words, the rocket made no further control but was affected only by the gravity and did free motion in its direction by its own velocity from the instance the Stage III being burnout to the instance the Stage IV starting igniting. In fact, the orbit position, entering velocity and attitude of the instance Stage III being burnout are all designed to achieve those of the instance Stage IV starting igniting.

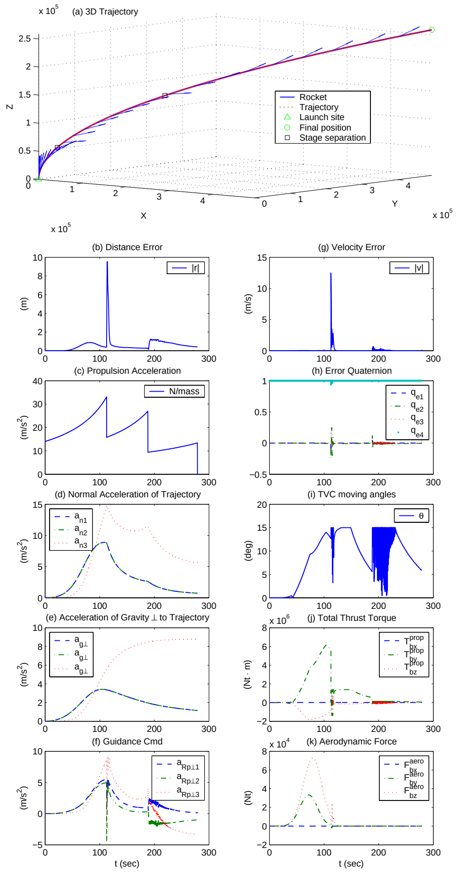

Fig. 5. Rocket's Integrated System

Therefore, the simulation is made on the flight sequence from Stage I to Stage III, and the goal is to track the position, velocity and attitude of the preset trajectory profile. The preset trajectory can be characterized by 2 order polynomial as:

$$\mathbf{r}(\tau) = \begin{bmatrix} x(\tau) \\ y(\tau) \\ z(\tau) \end{bmatrix} = \begin{bmatrix} 0.49427\tau^2 + 4.0778e^{-12}\tau \\ 0.49427\tau^2 + 4.0778e^{-12}\tau \\ 3.2886e^{-16}\tau^2 + 259\tau \end{bmatrix}$$

where $\tau = 0 \sim 1000$. Because of the different mechanism in each stage, the parameters of rocket airframe are divided into 4 sets, and can be found in (Isakowitz, 1991). The initial total mass and length of whole rocket are 75,000kg and 27.4m, respectively, and the initial velocity of rocket is set 2.59m/s upright. In the simulation, we simply omit the rotation and revolution of the earth.

In Fig. 5(a), we draw an arrow each 10sec to represent the attitude of the rocket. The total flight time is 279sec, final velocity is 6744.7m/s and the maximum distance error is less than 9.5474m throughout the whole launch phase, thus satisfy the requirements. Through the whole phase, two dramatic changes are occurred when the rocket makes stage separations in 112.6sec and 188.2sec. In (b), (g) and (h), the peak value of errors happen at the two time instant, and those errors have converged immediately as expected. In (c), the variation of propulsion acceleration is induced by the decreasing of the fuel mass and the different propellent force in each stage. In (d), the normal acceleration of trajectory increasing as the airspeed growing but decreasing as the reducing of the radius of curvature of the preset trajectory. In (i), the moving angle of TVC θ reaches the constraint 15deg. In (j), the total thrust torque is generated by the cooperation TVC and DCS. In (k), the aerodynamics produces huge effect and induces distance errors before 112.6sec, when the rocket is inside atmosphere and is with the maximum surface without any stage separations.

In summary, the performance of the designed rocket integrated G/A system is excellent, and the system can achieve the trajectory-following task in the presence of aerodynamics which is considered as disturbance over the design.

5. CONCLUSIONS

Being the decision parts of an rocket, GL and autopilot system are discussed is this paper. A wingless airframe based on TVC/DCS and the sliding mode theory is proposed, which makes the autopilot system an whole-phase-single-strategy robust controller. Besides, new rocket GL based on differential geometry and sliding mode theory is presented. Finally, the stability of the integrated G/A system is guaranteed by the suggested system parameters, and verified by various simulations.

6. ACKNOWLEDGEMENT

This paper is sponsored by National Science Council under the contract NSC93-2752-E-002-007-PAE.

REFERENCES

Bryson, Arthur E. Jr. (1994). *Control of Spacecraft and Aircraft*. Princeton. New Jersey.

Cheng, Chia-Hao (2004). *Novel Guidance Law and Autopilot Designs for Intercepting Missiles and Launch Rockets*. Master's Thesis, NTU EE. Taipei.

Goetz, Abraham (1970). *Introduction to Differential Geometry*. Assison-Wesley.

Isakowitz, Steven J. (1991). *International Reference Guide to Space Launch Systems, 2nd Edition*. AIAA, Inc.

Johnson, Eric N., Anthony J. Calise and J. Eric Corban (2001). Adaptive guidance and control for autonomous launch vehicles. *IEEE Proceedings* pp. 2669–2682.

Yeh, Fu-Kuang (2003). *Variable Structure Theory Based Integrated Guidance/Autopilot Design for Maneuverable Flying Vehicles*. Ph.D's Dissertation, NTU EE. Taipei.