DEVELOPMENT AND TUNING OF A NONLINEAR SIX DOF MODEL AND CONTROLLERS FOR A LARGE UAV

P. Jagadeesh Balaji*, K. Senthil Kumar** and J. Shanmugam+

Abstract

This paper presents development of non-linear six-degree-of-freedom model for a large UAV and simulation of its dynamics during its landing phase. A non-linear six-degree-of-freedom aircraft model was developed using the block available in the Aerosim Aeronautical Simulation Blockset in MATLAB / SIMULINK environment. The aircraft parameter data for a large UAV was taken. The developed model could be used for simulating the dynamics of the UAV with single piston engine and fixed pitch propeller. The following controllers were tuned using the above model to simulate its landing phase. 'Bank to Aileron', 'Airspeed error to Pitch command', 'Pitch error to Elevator deflection', 'Engine rpm error to Throttle'. The first approach was by conventional techniques using PI and PID controllers. In the next stage fuzzy logic controllers were designed for the above cases. All fuzzy controllers were designed using Mamdani inference system.

Keywords: *UAV, 6DOF model, controller tuning*

Nomenclature

b = wing span
c = mean aerodynamic chord
e = wing planform efficiency factor
ht = high throttle
k_p, k_i, k_d = proportional integral and derivative gains
lt = low throttle
nl = negative large
ns = negative small
ode = ordinary differential equation.
pl = positive large
p, q, r = angular velocities along X, Y and Z-axes
ps = positive small
pSL = sea-level pressure
r_t = radius of turn
AR = aspect ratio
BSFC = brake specific fuel consumption
DOF = degrees of freedom
EOM = equations of motion
FLC = fuzzy logic controller
Ieng = engine shaft moment of inertia
Iprop = propeller moment of inertia
M = mach number
MAP = manifold pressure vector
Meng = engine moment
Mprop = propeller moment
NGC = navigation guidance and control
NL = negative large
NS = negative small
PID = proportional integral derivative
PL = positive large
PS = positive small
R = propeller radius
RPM = rotations per minute
SISO = single input single output
TSL = sea level temperature
UAV = unmanned aerial vehicle
WMM = world magnetic model
2-D = two dimensional
1-D = one dimensional
C_D = drag coefficient
C_{D0} = zero-lift drag coefficient
$C_D^{\delta f}$ = drag coefficient-lift control (flap) derivative
$C_D^{\delta e}$ = drag coefficient pitch control(elevator)derivative

* Student, M.E. Avionics ** Lecturer + Professor
Division of Avionics, Department of Electronics Engineering, Madras Institute of Technology, Chennai-600 044, India
Email : jagadeesh_balaji@yahoo.com
Manuscript received on 27 Jun 2006; Paper reviewed, revised and accepted on 07 May 2007

$C_D^{\delta a}$ = drag coefficient roll control aileron derivative

$C_D^{\delta r}$ = drag coefficient yaw control (rudder) derivative

C_D^M = drag coefficient mach number derivative

C_L = lift coefficient

C_{L0} = lift coefficient zero-alpha lift

C_L^α = lift coefficient alpha derivative

$C_L^{\delta f}$ = lift coefficient lift control (flap) derivative

$C_L^{\delta e}$ = lift coefficient pitch control (elevator) derivative

$C_L^{\dot{\alpha}}$ = lift coefficient alpha dot derivative

C_L^q = lift coefficient pitch rate derivative

C_L^M = lift coefficient mach number derivative

C_P = coefficient of power

C_T = coefficient of thrust

C_Y = side-force coefficient

C_Y^β = side-force coefficient sideslip derivative

$C_Y^{\delta a}$ = side-force coefficient roll control derivative

$C_Y^{\delta r}$ = side-force coefficient yaw control derivative

C_Y^p = side-force coefficient roll rate derivative

C_Y^r = side-force coefficient yaw rate derivative

C_l = roll moment coefficient

C_l^β = roll moment coefficient sideslip derivative

$C_l^{\delta a}$ = roll moment coefficient roll control derivative

$C_l^{\delta r}$ = roll moment coefficient yaw control derivative

C_l^p = roll moment coefficient roll rate derivative

C_l^r = roll moment coefficient yaw rate derivative

C_m = pitch moment coefficient

C_{m0} = pitch moment coefficient at zero-alpha lift

C_m^α = pitch moment coefficient alpha derivative

$C_m^{\delta f}$ = pitch moment coefficient lift control (flap) derivative

$C_m^{\delta e}$ = pitch moment coefficient pitch control derivative

$C_m^{\dot{\alpha}}$ = pitch moment coefficient alpha-dot derivative

C_m^q = pitch moment coefficient pitch rate derivative

C_m^M = pitch moment coefficient mach number derivative

C_n = yaw moment coefficient

C_n^β = yaw moment coefficient sideslip derivative

$C_n^{\delta a}$ = yaw moment coefficient roll control derivative

$C_n^{\delta r}$ = yaw moment coefficient yaw control derivative

C_n^p = yaw moment coefficient roll rate derivative

C_n^r = yaw moment coefficient yaw rate derivative

F_P = propeller force

Va = airspeed

Φ = bank angle

Ω = engine shaft angular velocity

$\dot{\Omega}$ = engine shaft angular acceleration

Introduction

In this paper, development of nonlinear six-degree of freedom aircraft model is described. The aircraft model for the UAV was developed using the **6-DOF Aircraft Model (with Body-frame EOM)** block from the **AeroSim** blockset, with the propulsion subsystem modified. Aerosim has got two aircraft model examples, namely Navion (a general aviation airplane) and Aerosonde (a weather-reconnaissance and remote sensing mission UAV). Using these two aircraft models NGC simulations were carried out [1] and [2]. The complete aircraft model blocks available can be cutomized using aircraft configuration script. Thus, any UAV can be modelled using the aircraft configuration script template available in samples directory under the Aerosim tree. In this project aircraft data for a large UAV was taken. The aircraft configuration script consists of five sections. In section 1, the aerodynamic parameters of the UAV are specified. In section 2 the geometry and aerodynamic performance of the propeller are specified. In section 3, the engine characteristics are specified. All engine data is

given at sea level. In section 4, the inertia parameters of the UAV are specified, namely mass, centre of gravity location and moments of inertia. In section 5, the calendar date used by the world magnetic model is specified. When all the parameters of the aircraft configuration script are filled up then it can be saved with a unique name. The internal structure of the complete UAV block is shown in Fig.1. The internal structure of aircraft model has the following subsystems. They are aerodynamic model with linear aerodynamics, propulsion model with piston-engine propulsion and fixed pitch propeller block, inertia model which has weight variation due to fuel consumption, atmosphere model with standard atmosphere, wind gusts and turbulence blocks, Earth model (which provides Earth radius, gravity and magnetic field components at current aircraft location) and equations of motion subsystem with nonlinear equations of motion. In the aerodynamic model the aerodynamic force and moment coefficients are computed using linear combinations of aerodynamic derivatives. The aerodynamic coefficients are computed using the following formulae [3].

$$C_L = C_{L0} + C_L^{\alpha} . \alpha + C_L^{\delta f} . \delta_f + C_L^{\delta e} . \delta e$$

$$+ (c / 2V_a) . (C_L^{\dot{\alpha}} . \dot{\alpha} + C_L^{q} . q) + C_L^{M} . M \qquad (1)$$

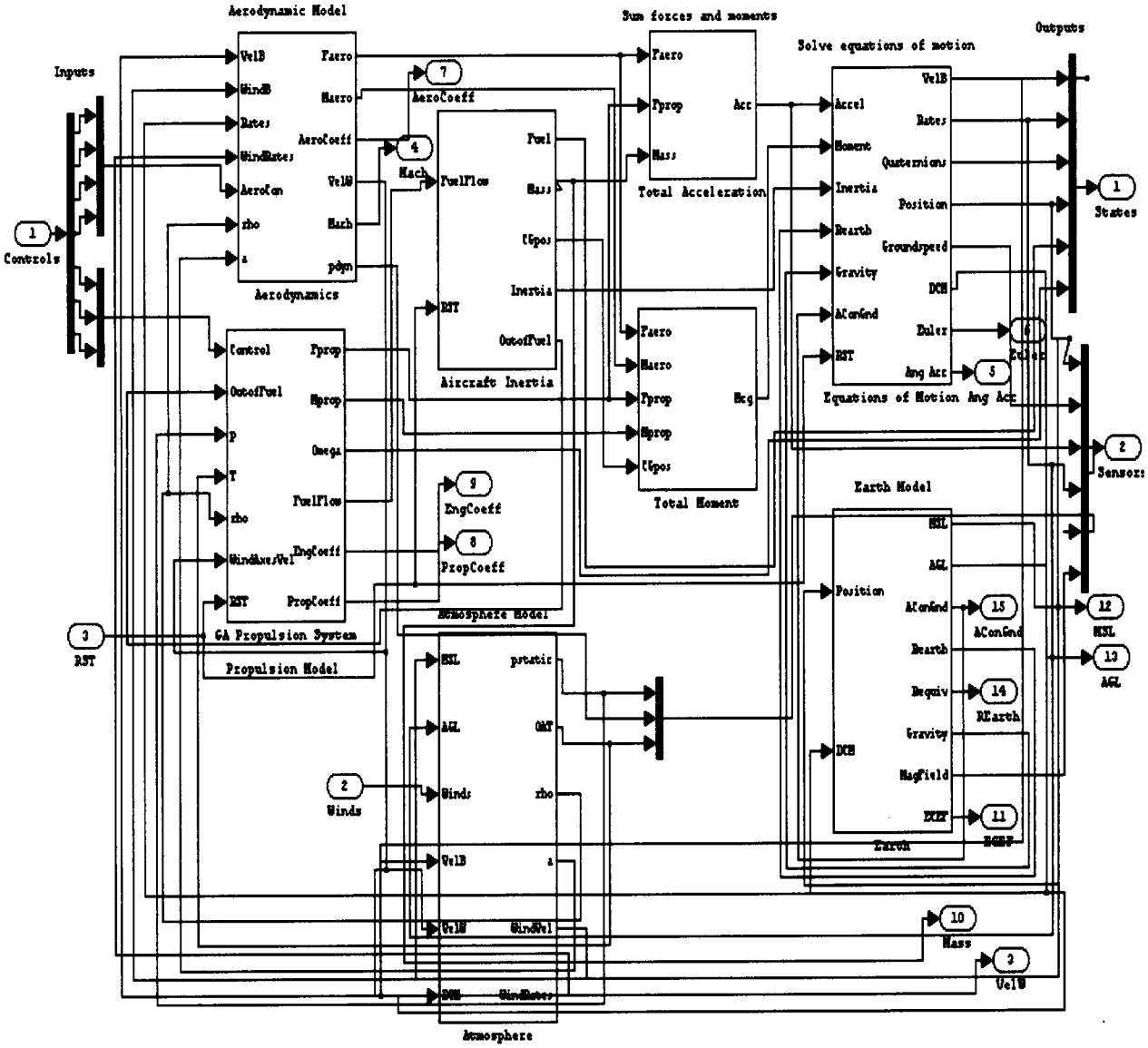

Fig.1 Internal structure of the complete UAV model block

$$C_D = C_{D0} + (C_L - C_{L0})^2/(\pi\, eAR) + C_D^{\delta f} . \delta f + C_D^{\delta e} . \delta e + C_D^{\delta a} . \delta a + C_D^{\delta r} . \delta r + C_D^{M} . M \quad (2)$$

$$C_Y = C_Y^{\beta} . \beta + C_Y^{\delta a} . \delta a + C_Y^{\delta r} . \delta r + (b/2V_a) . (C_Y^{p} . p + C_Y^{r} . r) \quad (3)$$

$$C_m = C_{m0} + C_m^{\alpha} . \alpha + C_m^{\delta f} . \delta_f + C_m^{\delta e} . \delta e + (c/2V_a) . (C_m^{\dot{\alpha}} . \dot{\alpha} + C_m^{q} . q) + C_m^{M} . M \quad (4)$$

$$C_l = C_l^{\beta} . \beta + C_l^{\delta a} . \delta a + C_l^{\delta r} . \delta r + (b/2V_a) . (C_l^{p} . p + C_l^{r} . r) \quad (5)$$

$$C_n = Cn^{\beta} . \beta + C_n^{\delta a}\ \delta a + Cn^{\delta r} . \delta r + (b/2V_a) . (C\,n^{p} . p + Cn^{r} . r) \quad (6)$$

The differential equation that describes the dynamics of the propulsion system is

$$(Ieng + Iprop) . \dot{\Omega} = Meng + Mprop \quad (7)$$

This equation is integrated forward in time to compute the engine speed Ω at the next time step. The propeller force and moment are computed using the formulae,

$$F_P = (4/\pi^2)\, \rho\, R^4\, \Omega^2\, C_T \quad (8)$$

$$M_P = -(4/\pi^3)\, \rho\, R^5\, \Omega^2\, C_P \quad (9)$$

The **forces** block in EOM block implemented the rigid-body 6 degree-of-freedom force equations that describe the time variation of the aircraft velocities. The **moments** block in EOM block integrated the rigid-body 6 degree-of-freedom moment equations to obtain the instantaneous body angular rates. The detailed explanation of all the blocks utilised in the development of the UAV model is available in Aerosim user's guide [3].

In the following sections how to setup and run the newly built UAV model is explained. The landing control algorithm adopted to simulate the dynamics of the UAV during its landing phase is explained. The control algorithm adopted during base leg and final turn is explained. The fuzzy controllers designed are explained. Thus the present work stands as first hand experience in developing a nonlinear six DOF model for a large UAV and do the simulation analysis for the same.

Setting up and Running the UAV Model

When the aircraft configuration script developed was run, a new aircraft configuration file with .mat extension was created. This file name was then used in the complete aircraft block **(6 DOF body frame EOM)** available in the Aerosim library. In the general-aviation propulsion system block, the power developed by the engine is computed as a function of RPM and MAP. The data available to interpolate power was not adequate. So, the 2-D power lookup table f(RPM, MAP) in the piston engine block was replaced by 1-D power lookup table f(throttle) as shown in Fig.2. The power output was quite good with the above tuning. The block parameters for the complete aircraft (EOM) block consisted of aircraft configuration file, initial conditions for velocities, angular rates, attitude, position, fuel mass, engine speed, ground altitude, WMM coefficient file, simulation date and sample time.

All the aircraft block parameters were filled up and the UAV model was made ready for simulation. The UAV was given an initial velocity of 35 m/s, initial altitude of 1000 m and all other initial conditions in the block parameters were filled. The model was run in open loop. The simulink model used was as shown in Fig. 3. Before the model was run, the simulation pull-down menu in the open loop simulink model was selected. Then the configuration parameters option was selected. In the configuration parameters the solver type was set to fixed-step. The integration scheme was set to ode4 or ode5 [3]. The fixed step size was set to match the aircraft model sample time, which was 0.005 seconds. The bank response indicated that the spiral mode was unstable. When the model was run with engine OFF the bank angle was zero. Thus, spiral mode instability was due to engine torque. The open loop responses with engine ON and engine OFF were as shown in Fig. 4.

Controller Tuning By Conventional Method

The 'airspeed to elevator', 'bank to aileron' and 'engine rpm to throttle' controllers were tuned using the above model. With these controllers the landing phase of the UAV was simulated. Trim and linearization routine was developed for the developed model based on the already available trim and linearization routine for Aerosonde UAV available in **trim** directory. The control-

Fig.2 Inclusion of 1-D power lookup table in the piston engine block

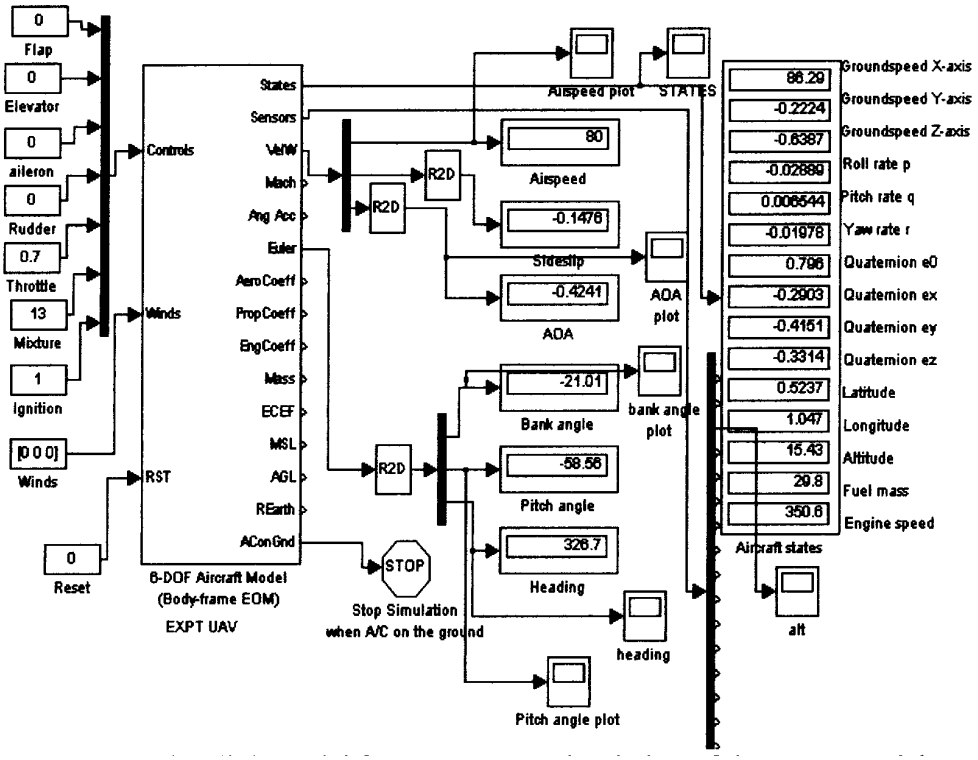

Fig. 3 Simulink model for open loop simulation of the UAV model

Fig.4 Open loop responses for the UAV model with engine ON and OFF

ler design using the linear models based on this routine did not give correct results. So, the controllers were tuned by trial and error process. A PI controller was found to be sufficient for bank control. The values of k_p and k_i were found to be 3.5 and 1 respectively. PID controller was implemented for airspeed error to pitch command. The values of k_p , k_i and k_d were found to be -4.5, -1 and -0.1 respectively. The pitch error to elevator deflection controller gains were found to be $k_p = 4$, $k_i = 0.87$ and $k_d = 0.4$. A PI controller was found sufficient for 'engine rpm error to throttle command'. The gains for the controller were found to be $k_p = 0.01$ and $k_i = 0.1$.

Base Leg and Final Turn Algorithm

The base leg and final turn was assumed to commence from an altitude of 330 m. The airspeed command was maintained at 35 m/s. The bank command was given as -15 degrees. The engine rpm command was given as 1700 rpm in order to maintain a rate of descent of around 5m/s. The initial conditions were filled in the block parameters and the aircraft configuration file was 'exptuav'. This phase lasted till 58m, for which the default ground was taken to be 15.58m above mean sea level. The UAV took a radius of turn as 466.5m and descended to an altitude of 58m, where the landing phase started. The responses for the UAV model during base leg and final turn were as shown in Fig. 5.

Landing Phase Algorithm

The approach was assumed to start at an altitude of 58m. The airspeed command was kept at 32.5m/s. The bank command was kept at 0 degrees. The engine rpm command was maintained at 2080 rpm. The engine rpm was maintained constant in order to maintain a rate of descent of around 1.8 m/s. The glide slope was assumed to be 3 degrees. The range was assumed to be 800m. The

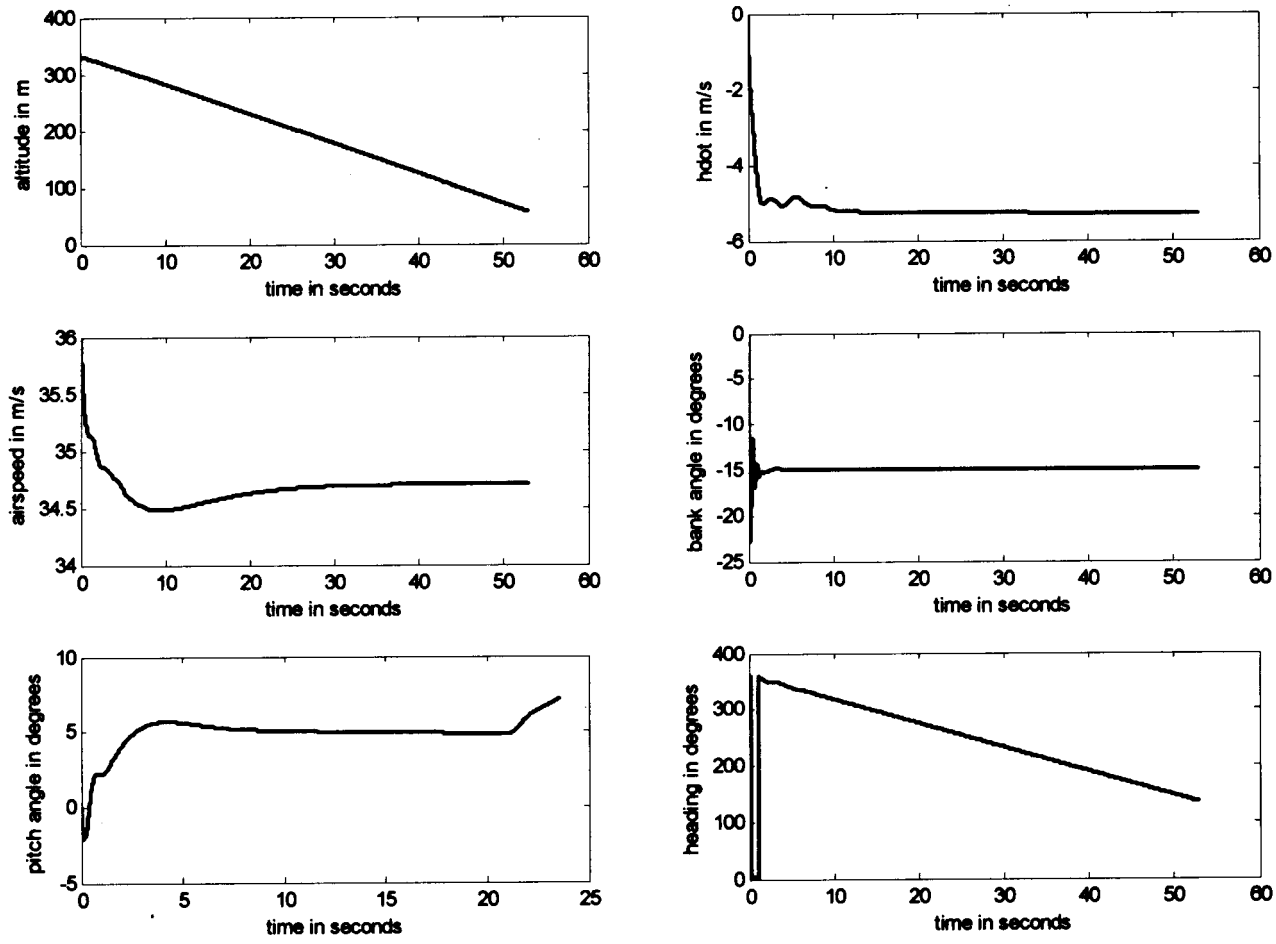

Fig.5 Base leg and final turn responses for the UAV model

default ground altitude was 15.58 m. The initial conditions were filled in the block parameters.

The flare was made to commence at an altitude of 19m above mean sea level, by cutting off the elevator control input from the controller and giving a constant elevator deflection of 20 degrees. This was implemented in the simulink model by providing a conditional switch, which monitored the altitude and disconnected the elevator control input when the altitude reached 19m and a constant elevator deflection was given to the elevator. The simulink model for the landing phase of UAV is shown in Fig. 6. The responses for landing phase were as shown in Fig. 7.

Fuzzy Logic Controller Design

During the landing phase, the system dynamics is nonlinear and human expertise for landing was available, so FLC for landing was attempted in the place of conventional controllers. All fuzzy logic controllers were SISO systems and were designed using Mamdani inference system. Center of area defuzzification scheme was used for all the controllers. Some of the rules used in the bank to aileron FLC were as follows:

- If bank error is PS then aileron deflection is NS
- If bank error is NL then aileron deflection is PL

Some of the rules used by the airspeed to elevator FLC were as follows:

- If airspeed error is pl then elevator deflection is nl.
- If airspeed error is ps then elevator deflection is ns.

Some of the rules used by the engine rpm to throttle FLC were as follows:

- If rpm error is pl then throttle is ht.
- If rpm error is ps then throttle is lt.

Fig. 6 Simulink model for the landing phase of the UAV

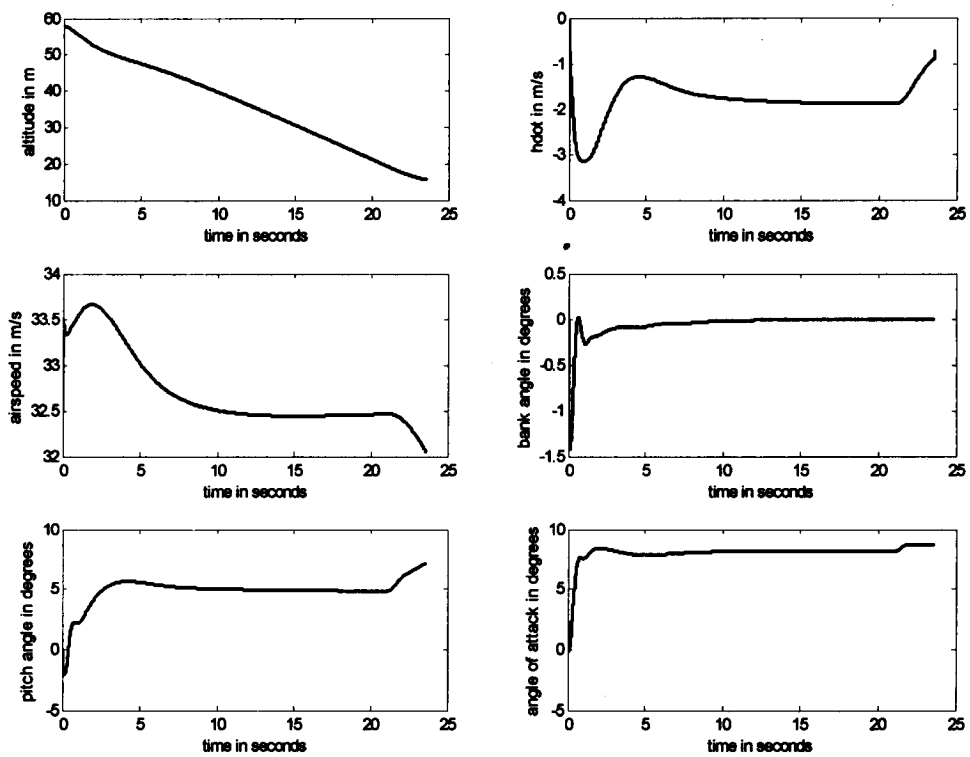

Fig.7 Landing phase responses for the UAV model

The landing phase responses for the UAV model using FLCs were as shown in Fig. 8.The rule viewers for bank to aileron, airspeed to elevator and engine rpm to throttle were as shown in Figs. 9, 10 and 11 respectively.

Discussion of Results

During base leg and final turn the UAV took a semi-circular descend to come to the landing phase. The heading was linearly decreasing. The radius of turn was calculated using the formula

$$Va^2/(r_t \cdot g) = \tan \phi \qquad (10)$$

During glide slope the airspeed was settling to the commanded value of 32.5m/s. The sink rate was also maintained constant to around 1.8m/s. At the time of flare the constant elevator deflection had resulted in a gradual decrease in the rate of descent and made the UAV land safely. The airspeed also decreases during flare and its value was below the stall speed of the UAV.

Conclusion

A non-linear 6-DOF model was developed for a large UAV. This will serve as an experience for developing a 6-DOF model for any large UAV with single piston engine and fixed pitch propeller. The developed model was simulated to know its dynamics during landing phase, which can be helpful for pilot assisted landing. The future scope of the project can be the following:

- Manual pilot commands can be provided through the joystick interface block.

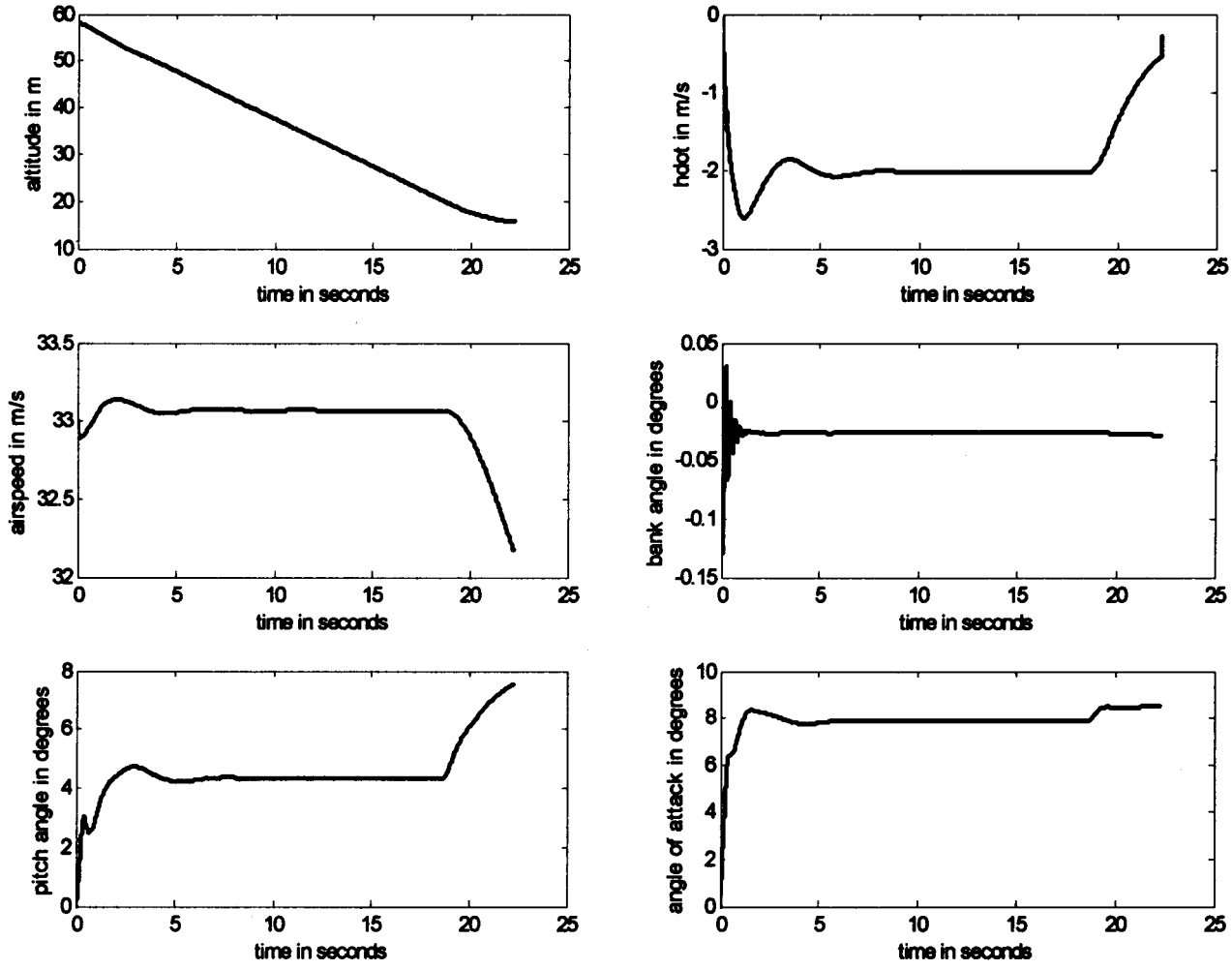

Fig.8 Landing phase responses for the UAV model using FLCs

Fig.9 Rule viewer for bank to aileron fuzzy logic controller

Fig.10 Rule viewer for airspeed to elevator fuzzy logic controller

Fig.11 Rule viewer for engine RPM to throttle fuzzy logic controller

- Flight parameters from simulation can be sent for visual output by making use of the flight simulator interface blocks.
- Actuator models can be included in the simulation.
- The sensor models can be included to increase the realism of the simulation.
- Trim and linerisation routine can be tuned to give correct results.
- The effect of wind on the dynamics of the UAV can be studied and
- Other UAV missions like waypoint navigation, control and guidance can be simulated.

References

1. Doitsidis, L., Valavanis, K.P., Tsourveloudis, N.C. and Kontitsis, M., "A Framework for Fuzzy Logic Based UAV Navigation and Control", Proceedings of the IEEE, International Conference on Robotics and Automation, New Orleans, LA, USA, Apr. 2004, pp.4041-4046.

2. Randy, C. Hoover., "Fusion of Hard and Soft Computing for Guidance Navigation and Control of Uninhabited Vehicles", MS Thesis, Idaho State University, Dec. 2004.

3. Aerosim, Aeronautical Simulation Blockset. v1.2, Users Guide, www.u-dynamics.com.

4. Riseborough, P., "Automatic Take-off and Landing Control for Small UAV's", BAE Systems Australia, Melbourne, ASCC-2004.

5. Robert, C. Nelson., "Flight Stability and Automatic Control", McGraw-Hill, Inc., 1989.

6. Blakelock, J.H., "Automatic Control of Aircraft and Missiles", John Wiley Sons, New York, 1990.