Fuzzy sliding mode autopilot design for nonminimum phase and nonlinear UAV

A.R. Babaei*, M. Mortazavi and M.H. Moradi

Aerospace Engineering Department, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran
Center of Excellence in Computational Aerospace, Aerospace Engineering Department, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran
Medical Engineering Department, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran

Abstract. The fuzzy sliding mode control based on the multi-objective genetic algorithm is proposed to design the altitude autopilot of a UAV. This case presents an interesting challenge due to non-minimum phase characteristic, nonlinearities and uncertainties of the altitude to elevator relation. The response of this autopilot is investigated through various criteria such as time response characteristics, robustness with respect to parametric uncertainties, and robustness with respect to unmodeled dynamics. The parametric robustness is investigated with reduction in significant longitudinal stability coefficients. Also, a nonlinear model in presence of the coupling terms is used to investigate the robustness with respect to unmodeled dynamics. In spite of a designed classic autopilot, it is shown by simulation that combining of the sliding mode control robustness and the fuzzy logic control independence of system model can guarantee the acceptable robust performance and stability with respect to unmodeled dynamics and parametric uncertainty, while the number of FSMC rules is smaller than that for the conventional fuzzy logic control.

Keywords: Autopilot, UAV, sliding mode control, fuzzy logic control, uncertainty, nonminimum Phase

Nomenclature

FSMC = Fuzzy Sliding Mode Control
FLC = Fuzzy Logic Control
FLS = Fuzzy Logic System
SMC = Sliding Mode Control
I/O = input/output
GA = genetic algorithm
UAV = unmanned aerial vehicle
δE = elevator control variable
δEtrim = elevator trim angle
δR = rudder control variable
δA = aileron control variable
h = UAV altitude
hc = Command altitude
φ = roll angle
θ = pitch angle
ψ = heading angle
ρ = roll angle rate
Q = pitch angle rate
R = heading angle rate
U = forward velocity
W = vertical velocity
V = lateral velocity
Vt = total velocity
α = angle of attack
β = side slip angle
Npop = the number of chromosome population
Nvar = the number of a chromosome variables
Nkeep = the number of kept chromosome for mating
β0 = a random number on the interval [0, 1]
1. Introduction

The main difficulties in aerospace vehicles autopilot design are known to result from the aerodynamic parameter uncertainties. Because the knowledge of aerodynamic coefficients as well as their dependencies on some parameters (e.g. angle of attack, side slip angle... etc.) is very imprecise, the controller design must not be sensitive to the variations of these coefficients. In general, the dynamics of aerospace vehicles are nonlinear, time varying, uncertain and they are controlled by assuming that inertial cross couplings among roll, pitch, and yaw dynamics are negligible. In practice, there is coupling among the dynamics, and this can result in performance degradation. To design a high performance autopilot, it is desirable to use a more general model containing the coupling terms. Therefore, one of major problems in designing automatic flight control system is modeling uncertainties such as parameters variation in characterizing an aerospace vehicle model and unknown nonlinear dynamics.

In order to get ride of the exact model restrictions, several adaptive schemes have been introduced to solve the problem of linearly parameterized uncertainties [3, 13], which are referred to as structured uncertainties. Unfortunately, in the some of applications, there are some controlled systems which are characterized by an unmodeled or/and unknown dynamics which are referred the unstructured uncertainties.

Feedback linearization [22] is a popular method used in nonlinear control applications, and there have been several flight control demonstrations [1, 6, 18, and 23]. This technique has received much of attention and shows great promise. The main drawback to the nonlinear control approach such as feedback linearization is that, as model-based control method, they require accurate knowledge of the plant dynamics. This is significant in flight control since the aerodynamic of parameters always contain some degree of uncertainty. In this paper, proposed technique is applied to the UAV altitude control which is aerodynamically controlled by tail control surfaces using single-loop scheme. The uncertainties that are considered are due to both parametric uncertainty and unknown nonlinear effects specially, high coupling among altitude and yaw variables. In fact, the I/O feedback linearization technique which has been regarded as the powerful design method for nonlinear systems, can not directly applied to the altitude control of UAVs because pitch channel of tail-controlled UAVs is inherently nonminimum phase. This fact makes it difficult to design the high performance autopilot.

In this paper, it is tried to achieve many properties: desirable tracking, suitable time response characteristics, robustness with respect to unmodeled dynamics such as coupling effects, and robustness with respect to parametric uncertainty. Two strategies are utilized to design an autopilot. In first strategy, the classic compensator that utilizes the root locus method capabilities is designed based on the linear model. In second strategy, the sliding mode control [22] (the system uncertainties and external disturbances can be handled by it), the fuzzy logic control [25] (it is independent of system model), and the GA optimum search mechanism [9] have been used in designing the autopilot called FSMC. The FSMC method has recently been used for various applications [12, 14, and 27]. Also, optimal design of fuzzy systems has been widely applied in different applications by the evolutionary algorithms [2, 5, 16, 21, and 26]. In [14], the UAV altitude based on linear model is controlled by the expert knowledge-based FLC so that robustness with respect to parametric uncertainty is achieved. In ref. [5] and [24] the missile acceleration that is nonminimum phase is indirectly controlled (lateral velocities are controlled) by FLC and GA. In ref. [3], the UAV altitude with linear model is controlled by combination of a linear controller and a simple adaptive controller to modify robustness with respect to parametric uncertainties only. In ref. [16], the fuzzy PID controller based on GA for nonminimum phase and linear systems is presented to eliminate undershoot that it is due to unstable zeros. Besides, ref. [2] satisfies a nonminimum phase and nonlinear problem using conventional FLC optimization by GA. Large rule base then high computational burden is the main drawback of conventional FLC. In this paper, it is attempted to overcome this problem...
by using fuzzy sliding mode control for a critical case study.

The rest of the paper is organized as follows: Section 2 presents the linear and nonlinear model of a UAV. The linear autopilot design is presented in Section 3. The FSMC based on GA autopilot is designed in Section 4. The simulation results and their analytic discussions are presented in Section 5 to investigate several aspects. Finally, conclusions are drawn in Section 6.

2. UAV model

The nonlinear motion equations of UAV are derived as follows [2 and 4]:

\[ U = -9.8 \sin \theta - QW + RV - 0.0125U - 16.63x + 16.64 \Delta E + 4.5 \]  
\[ V = 9.8 \sin \phi \cos \theta + PW - RU - 263.77\beta - 0.0053P + 1.64R - 0.00323\Delta \lambda - 58.24q \]  
\[ W = 9.8 \cos \phi \cos \theta + QU - PV - 0.065U - 259\alpha - 1.3Q + 57.5\Delta \theta + 21 \]  
\[ P = -1.51QR + 0.045PQ + 76.77\beta - 1.9P + 0.68R + 149\Delta \alpha + 1056 \]  
\[ Q = 1.03PR - 0.017(P^2 - 4^2) - 988\alpha - 0.9Q + 1362\Delta \phi - 0.284 \]  
\[ R = -0.038QR - 0.05PQ + 306\beta - 0.044P + 2.8R + 2.273\Delta \theta + 4346R \]  
\[ \alpha = \frac{W \cos \alpha - U \sin \alpha}{V \cos \beta} \]  
\[ \beta = \frac{1}{V} \left( -U \cos \alpha \sin \beta + V \cos \beta - W \sin \alpha \sin \beta \right) \]  
\[ \phi = P + Q \sin \phi \tan \theta + R \cos \phi \tan \theta \]  
\[ \theta = Q \cos \phi - R \sin \phi \]  
\[ \psi = (Q \sin \phi + R \cos \phi) \sec \theta \]  
\[ h = U \sin \theta - V \cos \theta \sin \phi - W \cos \theta \cos \phi \]  
\[ V_c = \sqrt{U^2 + V^2 + W^2} \]  
\[ \delta_2 = \delta_{E\alpha} + \delta_\phi \]

Some of the UAV variables are defined in Fig. 1. We are aimed to design the altitude hold mode autopilot for this UAV. An altitude hold mode is an autopilot mode which tries to maintain altitude by the elevator input. The nominal linear model for altitude is written as following transfer function [2]:

\[ \frac{h(s)}{\delta (s)} = \frac{-57.3 (s - 24.6) (s + 21) (s + 0.008)}{s (s + 0.011s + 0.0022) (s^2 + 2.12s + 98.4)} \]  

(2)

It can be received that the altitude output and elevator input relation is non-minimum phase, and the longitudinal flight modes aren’t desirable. If 40% reduction is applied to \( C_{uu} \), \( C_{uv} \), \( C_{Du} \), and \( C_{Ju} \), a degraded linear model is presented as following transfer function to investigate the parametric robustness [2]:

\[ \frac{h(s)}{\delta (s)} = \frac{-57.3 (s - 25.5) (s + 21.6) (s + 0.0017)}{s (s + 0.0055s + 0.0021) (s^2 + 1.82s + 64)} \]  

(3)

These variations in plant dynamics are caused moving an unstable zero further to the right, and also caused degradation of the phugoid and the short period flight modes.

3. Linear autopilot design

In this section, classic autopilot is designed for \( \psi \) and \( h \) variables using the nominal linear model. However, the \( \psi \)-autopilot isn’t presented and assumed that there

Fig. 1. Definition of UAV variables.
aren’t uncertainties in directional-lateral channel. The block diagram of the altitude hold mode to design the compensator is shown in Fig. 2.

Two compensators are designed by the root locus techniques as follows [2]:

\[
G_{Ca} = \frac{0.00681(S + 0.1)(S^2 + 2.12S + 98.4)}{(S + 20)(S^2 + 6.94S + 13.1)} \quad (4-a)
\]

\[
G_{Cb} = \frac{0.012(S + 0.05)(S^2 + 2.12S + 98.4)}{(S + 20)(S^2 + 6S + 15.25)} \quad (4-b)
\]

For closed loop system, the dominant characteristics are attained by applying these autopilots to the nominal linear model (2) as following values.

a: \(\xi = 0.79, \omega_n = 1.36\), b: \(\xi = 0.6, \omega_n = 2.31\)

To evaluate the designed autopilots, those are applied to the nominal linear model, the degraded linear model and the nonlinear model. The commands are \(\psi_c = 0\), 10 degree and \(h_c = 10\) m. The simulation results are illustrated in Figs. 3 and 4. According to Fig. 3, the desirable response is seen for the nominal linear model. Concerning degraded linear model, relative parametric robustness of first controller and low parametric robustness of second controller is known. By inspection of Fig. 4, the time response of both autopilots has been degraded for the UAV nonlinear model because the nonlinear terms have been appeared. Clearly, increasing the \(\psi\)–command value leads to increasing the degraded coupling nonlinear effects. Therefore, the compensators can’t meet the wanted requirements in term of the parametric uncertainties and the unmodeled dynamics. Therefore, the compensators can’t meet the wanted requirements in term of the parametric uncertainties and the unmodeled dynamics. Now, eliminating the nonlinear terms effects is required to achieve the desirable tracking. In addition, the autopilot should not be very sensitive to the variation of system parameters. Due to these reasons, it is tried to exploit the knowledge-based FLC that is independent of system model.

It should be noted that the elevator trim angle (here, \(\delta_{E\text{trim}} = 1.92^\circ\)) must be added to the control input in nonlinear simulation procedure because the controllers have been designed based on the nominal linear model.
4. FSMC autopilot design

The SMC [22] is a robust design methodology using a systematic scheme based on a sliding mode surface and Lyapunov stability theorem. The main advantage of the SMC is that the system uncertainties and external disturbances can be handled under the invariance characteristics of system’s sliding mode state with guaranteed system stability.

Fuzzy systems are knowledge-based or rule-based systems that were initiated by Lotfi A. Zadeh in 1965 [28]. The heart of a fuzzy system is a knowledge consisting of the so-called fuzzy IF-THEN rules. A fuzzy IF-THEN rule is an IF-THEN statement in which some words are characterized by continuous membership functions. A block diagram of a fuzzy logic system is shown in Fig. 5.

The SMC theory uses discontinuous control action to drive state trajectories toward a specific surface until stable equilibrium state is reached. This principle provides guidance to design a fuzzy logic controller.

Consider the following single-input nth order nonlinear system:

\[
\begin{align*}
    x_1(t) &= x_2(t) \\
    x_2(t) &= x_3(t) \\
    &\vdots \\
    x_n(t) &= f(X) + h(X)u
\end{align*}
\]

where \( X = [x_1 \ x_2 \ \ldots \ x_n]^T \) is the state vector, and \( u \) is the control input. If the desired state vector is defined as \( X_c = [x_{c1} \ x_{c2} \ \ldots \ x_{cn}]^T \), then the error vector \( E = [e_1 \ e_2 \ \ldots \ e_n]^T \) could be written as

\[
    E = X - X_c = [x_1 - x_{c1} \ x_2 - x_{c2} \ \ldots \ x_n - x_{cn}]^T \tag{6}
\]

and a linear function \( s : \mathbb{R}^n \rightarrow \mathbb{R} \) is defined as

\[
    s = CE \tag{7}
\]

where \( C = [c_1 \ c_2 \ \ldots \ c_n] \) is the coefficient row vector, and \( c_i \)'s are all real numbers. Then, a sliding surface can be represented as

\[
    S = c_1x_1 + c_2x_2 + \ldots + c_nx_n = 0 \tag{8}
\]

To design the control input \( u(t) \) so that the state trajectories are driven and attracted toward the sliding surface and then remain sliding on it, the following inequality must be satisfied

\[
    \dot{s} < -\eta \ |s| \tag{9}
\]

The idea behind equation (9) is in the sense of Lyapunov function. If we treat \( s \) as a scalar time function and the Lyapunov function \( v \) as

\[
    v = \frac{1}{2} s^2 > 0 \tag{10}
\]

Together with (9), we can guarantee that asymptotic stability of the system is satisfied.

Here, the altitude control is concerned. For this reason, we can use the equations (1-a)-(1-o) as follow:

\[
    X = F(X) + B(X)u
\]

\[
    s_1 = x_2 \tag{11}
\]

\[
    s_2 = f(X_1) + h(X_1)k_1 + \Delta (X_2, \Delta_k)
\]

where, \( X = [X_1^T \ X_2^T]^T \), \( X_1 = [U \ W \ Q \ \phi \ \theta]^T \), \( X_2 = [V \ \Phi \ \phi \ \phi \ \phi]^T \), \( \delta = [\delta_1 \ \delta_2 \ \delta_3 \ \delta_4 \ \delta_5]^T \), \( \delta_1 = \delta_k \), \( \delta_2 = [\delta_3 \ \delta_4 \ \delta_5]^T \), \( \phi = h \), and \( \Delta (X_2, \Delta_k) \) is contribution of lateral-directional variables that we call it coupling term. Specially, the \( \Delta \) term is nearly zero when \( \phi \) is zero. The \( F(X_1), f(X_1), h(X_1) \) are the continuous linear or the nonlinear functions that they are derived by equations (1-a)-(1-o).

By concerning altitude control significance in this paper, the sliding function is defined as follows (output-based sliding function):

\[
    s = (s_1 - x_{c1}) + \lambda (s_1 - x_{c1}) \tag{12}
\]

and the sliding surface is presented as follow:

\[
    S = (s_1 - x_{c1}) + \lambda (s_1 - x_{c1}) = 0 \tag{13}
\]

By concerning equations (9) and (11) we have following relation.
\[ x_s = f[(\dot{x}_1 - x_{1*}) + \lambda(\dot{x}_1 - \dot{x}_{1*})] \]
\[ = f(X_1) + \Delta X_2, \delta_j) + b(X_1)\dot{h}_G - \dot{x}_{1*} + \lambda \dot{x}_C \]
\[ \leq |s| F_0 + b(X_1)\dot{h}_G - \dot{x}_{1*} + \lambda \dot{x}_C \quad (14) \]

where \( e = x_1 - x_{1*} \), and \( f(X_1) + \Delta X_2, \delta_j) \leq F_0 \).
Therefore, we can drive following results:
- The control input on the two sides of the sliding surface are opposite in sign and its magnitude is proportional to the sliding function (12).
- If the term \( b \) be negative, then \( \text{sign}(\dot{h}_G) = \text{sign}(s) \).

We can conclude that \( \dot{h}_G \propto s \) for negative \( b \). This can be used to derive a fuzzy sliding mode controller. Therefore, the following fuzzy sliding mode control is designed to obtain the control input:

\[ R_j : \quad \text{IF } s \text{ IS } s_l \text{ THEN } \dot{h}_G \text{ IS } U_l \quad (15) \]

where \( s_l \) is the linguistic value of \( s \) in the \( j^{th} \)-fuzzy rule, and \( U_l \) is the linguistic value of \( \dot{h}_G \) in the \( j^{th} \)-fuzzy rule. Here, term \( b \) is derived as follows:

\[ b = 16.6 \sin \theta - 57.5 \cos \theta \quad (16) \]

For \( \theta < 74^\circ \), term \( b \) is negative. For our case, the condition \( \theta < 74^\circ \) is always satisfied. Therefore, we can tell that term \( b \) is always negative. Based on these discussions, the rules base, for FSMC design, is presented in Table 1, so that the gaussian membership functions for the sliding mode function are used (see Fig. 6).

Suppose that the fuzzy set \( U_l \) and \( s_l \) in (15) are normal with center \( \delta_l \) and \( \lambda_l \). Then the fuzzy systems with rule base (15), product inference engine, singleton fuzzifier, and center average defuzzifier are of the following form (25):

\[ \delta_G = \frac{\sum_{i=1}^{M} \mu_i(x)}{\sum_{i=1}^{M} \mu_i(x)}, \quad \mu_i(x) = \exp \left[ -\frac{(x - \delta_i)^2}{2\lambda_i^2} \right] \quad (17) \]

where, \( M = 7 \).

<table>
<thead>
<tr>
<th>Rule no.</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF ( s ) NB THEN ( u ) IS NB</td>
</tr>
<tr>
<td>2</td>
<td>IF ( s ) NM THEN ( u ) IS NM</td>
</tr>
<tr>
<td>3</td>
<td>IF ( s ) ZE THEN ( u ) IS ZE</td>
</tr>
<tr>
<td>4</td>
<td>IF ( s ) PS THEN ( u ) IS PS</td>
</tr>
<tr>
<td>5</td>
<td>IF ( s ) PM THEN ( u ) IS PM</td>
</tr>
<tr>
<td>7</td>
<td>IF ( s ) PB THEN ( u ) IS PB</td>
</tr>
</tbody>
</table>

Conventionally, the FLC is designed by expert knowledge and experience. It is difficult to decide the control rules as system is complex, for instance, a nonminimum phase. Increasingly, the GA is used to facilitate the FLS design so that it can be used to design the FSMC. The GA is a general-purpose search algorithm that uses principles inspired by natural population genetics to evolve solutions. It was first proposed in ref. [10]. The GA has been theoretically and empirically proven to provide a robust search in complex spaces, thereby offering a valid approach to problems requiring efficient and effective searches [8].

Here, the continuous or real-valued GA is used [9]. A path through the components of the GA is shown as a flowchart in Fig. 7. First, a chromosome population \( (N_{pop}) \) is randomly generated. Each chromosome specifies a candidate solution of the optimization problem. The fitness of all individuals with respect to the optimization task is then evaluated by a scalar cost function (fitness function).

The cost function is a surface with peaks and valleys when displayed in variable space, much like a topographic map. To find a valley, an optimization algorithm searches for the minimum cost. To find a peak, an optimization algorithm searches for the maximum cost. A cost function generates an output from a set of input variables (a chromosome). The object is to modify the output in some desirable fashion by finding the appropriate values for the input variables.
If the chromosome has $N_{var}$ variables given by $p_1, p_2, \ldots, p_{N_{var}}$, then the chromosome is written as an $N_{var}$ element row vector. Then, it is the time to decide which chromosomes in the initial population are fit enough to survive and possibly reproduce offspring in the next generation. The $N_{pop}$ costs and associated chromosomes are ranked from lowest cost to highest cost. From the $N_{pop}$ chromosomes in a given generation, only the top $N_{keep}$ are kept for mating (process of natural selection) and the rest are discarded to make room for the new offspring. One mother and one father in some random fashion are selected. Each pair produces two offspring that contain traits from each parent. In addition the parents survive to be part of the next generation.

One crossover point is randomly selected, and then the blending methods are applied by finding ways to combine variable values from the two parents into new variable values in the offspring. A single offspring variable value, $p_{new}$, comes from a combination of the two corresponding offspring variable values [19]:

$$p_{new} = \beta_0 p_{m} + (1 - \beta_0) p_{d}$$  \hfill (18)

If care is not taken, the GA can converge too quickly into one region of the cost surface. If this area is in the region of the global minimum, that is good. However, some functions have many local minima. If the tendency to converge quickly is not solved, a local rather than a global minimum is attained. To avoid this problem of overly fast convergence, it is forced the routine to explore other areas of the cost surface by randomly introducing changes, or mutations, in some of the variables. Most users of the continuous GA add a normally distributed random number to the variable selected for mutation:

$$p_n' = p_n + \sigma N_n(0, 1)$$  \hfill (19)

The centers of output membership functions ($\bar{\mu}$), $\lambda$, and variance ($\sigma_i$) for input membership functions are considered as a chromosome. The input variable boundaries are $u_{up} = 12^\circ$, $u_{low} = -12^\circ$, and for the sliding mode function, these are $s_{up} = 100$, $s_{low} = -100$.

Here, to determine the FSMC parameters, the GA with following properties is used:

- Chromosome population, $N_{pop}$ = 50
- Number of generation = 50
- Mutation rate = 2%
- $N_{keep}$ = 50%
- Cost function, $J = W_1 S_1 + W_2 S_2 + W_3 S_3$, so that $W_i$ is corresponding weight (see Fig. 8) [2]:

$$S_1 = \int_{t_1}^{t_2} |h - h_c| dt$$, this parameter covers rise time and undershoot that is caused by nonminimum phase effect.  

$$S_2 = \int_{t_1}^{t_2} |h - h_c| dt$$, this parameter includes overshoot value.
\( S = \int_0^T (\text{\textbf{h}} - h_c) \text{d}t \), this parameter comprises setting time, steady state error, and unstable effects.

The parameters in Fig. 8 are defined as follow:

- \( t_1 \) is time of first intersection between altitude time response and altitude command,
- \( t_2 \) is time of second intersection between altitude time response and altitude command, and
- \( T \) is final time of simulation. If the intersections don’t create, the \( t_i \) parameter is considered zero value.

In this alternative form of the cost function, \textit{overshoot}, \textit{undershoot}, \textit{rise time}, \textit{settling time}, \textit{steady state error}, and \textit{stability} objectives are easily covered to determine the optimal FSMC. Therefore, computation of all objectives isn’t individually required.

By consideration a trade-off procedure between time response characteristics and robustness, the weights are chosen as \( W_1 = 1 \), \( W_2 = 1.4 \), and \( W_3 = 1.3 \).

The closed loop scheme for this strategy is illustrated in Fig. 9. Due to the computational requirements, the controller is generally evolved off-line (using GA and a model of the controlled process). For this reason, the nominal linear model, as an available mathematical model of UAV, is utilized to achieve the controller parameters. Then, the robustness of controller with respect to the unmodeled dynamics that are not considered in the GA is investigated by using the nonlinear model in simulation procedure. Here, the nonlinear model is assumed as a real model of UAV. The GA results based on the nominal linear model are presented in Table 2. Simulation result for the nominal linear model is illustrated in Fig. 10. This figure displays the suitable time response characteristics. Also, a mean fitness value in each generation is shown in Fig. 11.

![Fig. 9. FSMC Scheme.](image)

### Table 2

The optimal parameters of the FSMC

<table>
<thead>
<tr>
<th>( \sigma_{\text{NB}} )</th>
<th>( \sigma_{\text{NM}} )</th>
<th>( \sigma_{\text{NS}} )</th>
<th>( \sigma_{\text{ZE}} )</th>
<th>( \sigma_{\text{PS}} )</th>
<th>( \sigma_{\text{PM}} )</th>
<th>( \lambda )</th>
</tr>
</thead>
<tbody>
<tr>
<td>23.6</td>
<td>28.4</td>
<td>19.8</td>
<td>17.5</td>
<td>26.9</td>
<td>20.8</td>
<td></td>
</tr>
<tr>
<td>23.8</td>
<td>21.6</td>
<td>19.8</td>
<td>17.5</td>
<td>26.9</td>
<td>20.8</td>
<td></td>
</tr>
<tr>
<td>-7.6</td>
<td>-4.7</td>
<td>0</td>
<td>4.34</td>
<td>9.5</td>
<td>0.84</td>
<td></td>
</tr>
</tbody>
</table>

![Fig. 10. Nominal linear model time response with FSMC-GA autopilot.](image)

![Fig. 11. Mean fitness value for FSMC.](image)

## 5. Results

In this section, the designed autopilots will be investigated in several aspects. First, robustness of FSMC is investigated. The simulation results are illustrated in Fig. 12 for degraded linear model and nonlinear model. The desirable performance of FSMC in presence of parametric uncertainties and unmodeled dynamics is clear. Therefore, FSMC with low rules and parameters is a robust method to design autopilots such as altitude hold mode. Therefore, in spite of the
comparable time response characteristics of classic controllers and FSMC for nominal linear model. FSMC strategy is robust with respect to parametric uncertainty and unmodeled dynamics. In Fig. 13, input behavior is illustrated for three autopilots. FSMC input is smaller than that for classic autopilots. Also, FSMC input has a desirable behavior.

Second, the performance of classic control and FSMC is investigated by reducing and increasing controllers gain. For instance, if we multiply signal of controllers by 0.5 and 2, then the results is shown in Fig. 14. The time response of system in presence of classic autopilot is largely degraded. For multiplier 2, overshoot is nearly 50% and for multiplier 0.5 rise time is nearly 5 second. But, the performance degradation of FSMC is less than that for classic autopilots, while these parameters are 30% and 2.7 second.

Thirdly, it is required that we investigate other flight variables such as velocity, angle of attack, and pitch rate. Time response of these variables is illustrated in Fig. 15. According to this figure, the velocity time response has desirable behavior for FSMC, but Angle of attack and pitch rate variables time response hasn’t desirable behavior. The altitude control leads to modifying phugoid mode (in this mode, velocity and altitude variables are significant), and we cannot well modify short period mode of aircraft (in this mode, angle of attack and pitch rate variables are significant). This goal is feasible by using the inner loops, or applying other states in sliding mode function and redefinition of cost function in term of the new sliding mode function. In other work, we are going to use recent procedure for other applications. In this application, altitude hold mode is
6. Conclusions

In this paper, the altitude hold mode autopilot has been designed for an unmanned aerial vehicle which is nonminimum phase, and its model includes nonlinearities and uncertainties. There are nonlinear terms such as coupling among the dynamics that may lead to performance degradation. The considered uncertainties are due to both parametric uncertainty and unmodeled nonlinear terms. Therefore, it has been tried to achieve some properties: desirable time response characteristics, robustness with respect to parametric uncertainty, and robustness with respect to unmodeled dynamics. The fuzzy sliding mode controller has been proposed to achieve these requirements. The simulation results show that the compensators can’t meet the wanted requirements in term of the parametric uncertainties and the unmodeled dynamics. Increasing the \( \psi \) command leads to increasing the degraded coupling nonlinear effects. In the fuzzy sliding mode autopilot design procedure, the multi-objective genetic algorithm has been used to mechanize the optimal determination of controller parameters based on an efficient cost function. In this cost function, the computation of all objectives isn’t individually required so that undershoot, overshoot, rise time, settling time, steady state error, and stability objectives have been easily covered by it. The results show the fuzzy sliding mode autopilot outperforms the compensator in terms of robustness with respect to parametric uncertainties and robustness with respect to unmodeled dynamics while both have comparable time response characteristics for nominal linear model. Also, after the control input signal gain widely changed, the performance margin of FSMC is better than that for classic autopilots. Finally, fuzzy sliding mode control with low rules then low computational burden has high potential to design the autopilot modes in presence of uncertainties.

**References**


