ROBUST NONLINEAR PREDICTIVE FLIGHT CONTROL

Ahmed Youssef, Mike Grimble, Andrzej Ordys,
Arkadiusz Dutka, David Anderson*

Industrial Control Centre, Department of Electronics and Electrical Engineering, University of Strathclyde, UK
* Industrial Systems & Control Ltd, UK
ahmed.youssef@strath.ac.uk, fax: (0044) 0141 548 4203

Keywords: Model Based Predictive Control, Nonlinear System, Nonlinear Generalized Predictive Control, Flight simulation.

Abstract

The requirements, design and simulation of a control system for a precision tracking task**,** called a Pitch-rate **C**ontrol **A**ugmentation **S**ystem (CAS), for the F-16 fighter aircraft model, is considered. This control system is designed over the entire flight envelope using one of the nonlinear **M**odel **B**ased **P**redictive **C**ontrol techniques (MBPC), which is called **N**on-**l**inear **G**eneralized **P**redictive **C**ontrol (NLGPC). The non-linear flight simulations reveal that the desired performance objectives are achieved and that the controller provides acceptable performance in spite of modelling errors and plant parameter variations.

1 Introduction

It is well known that the aerodynamic characteristics and operational requirements of modern high performance aircraft have required greater consideration of increasingly intensive and complicated nonlinearities. In spite of this knowledge, the process of designing flight control systems has been carried out traditionally by using linear systems analysis and design tools. A major reason that the use of linear systems theory has been so pervasive is that there is an analytical solution available. Hence, there are generally more rigorous stability and performance proofs that may be invoked. The computational demands and implementation problems for linear system simulation are usually quite small, in comparison to a nonlinear design. Obviously, the use of linear system techniques is quite limiting due to the large envelope of operation of the aircraft and thereby the high process uncertainties and nonlinearities. It is believed that progress in nonlinear control theory, combined with computer hardware advances, now allows advanced nonlinear control strategies to be implemented successfully in flight control systems.

MBPC is a form of control in which, at each instant of time, a performance index is minimised obtaining an optimal control sequence. Only the first element of this sequence is applied to the plant. The major advantage of this type of control is its ability to account for hard constraints. In earlier studies on the theory of MBPC, the technique of quadratic programming was employed to solve the open-loop optimal control problem with constraints. This results in a rapidly growing computational burden with the number of decision variables. That is why the MBPC originated in process industry where it was applied to very slow processes. Such solutions have been unsuitable for aerospace applications, since they are too slow to deal with the relatively fast dynamics of aircraft. In this paper a nonlinear MBPC approach is proposed, which utilises an explicit optimal control law that can be solved analytically removing the need for quadratic programming, this reduces the computations and allows the controller to deal with models of systems with fast dynamics.

This paper is organised as follows: In Section 2, a mathematical description of the aircraft model movement is introduced. Section 3 describes the underlying aircraft control augmentation system and poses the performance requirements imposed on it. Section 4 gives a description of the NLGPC used for the control system design. Non-linear simulations of the closed-loop system with the NLGPC are presented in Section 5 and this paper concludes with a brief summary in Section 6.

2 Nonlinear F-16 Aircraft Modelling

A model of high-performance F-16 aircraft was used to generate the simulation results in this paper. The simulation uses the standard longitudinal equations of motion and kinematic relations found in a variety of standard references on flight dynamics (see for example [1, 2]).

$$\dot{U} = -Q\,W + \frac{F_x}{m} - g\,\sin\theta \qquad (1)$$

$$\dot{W} = Q\ U + \frac{F_z}{m} + g\ \cos\theta \quad (2)$$

$$\dot{\theta} = Q \quad (3)$$

$$\dot{Q} = \bar{q}\ S\ \bar{c}\frac{C_m}{I_y} \quad (4)$$

$$\dot{H} = U\ \sin\theta - W\ \cos\theta \quad (5)$$

$$\dot{POW} = f(\delta_t) \quad (6)$$

where: U and W are the forward and downward components of the aircraft velocity V_T respectively; g is the gravitational acceleration vector; θ is the pitch angle; m is the aircraft mass; Q is the pitch-rate; S is wing area; POW is the engine power level; H is the altitude; I_y is the moment of inertia about OY axis; $\bar{q}$ is the dynamic pressure; $\bar{c}$ is the mean dynamic chord; F_x and F_z are the total forces acting along X and Z axes respectively and are equal to:

$$F_x = \bar{q}SC_x + T \quad (7)$$

$$F_z = \bar{q}SC_z \quad (8)$$

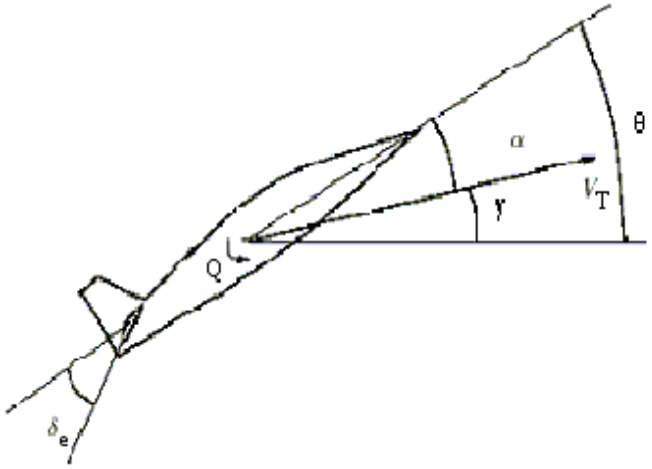

Figure 1: Illustration of longitudinal aircraft entities.

T is engine thrust vector (non-linear function depends on throttle setting δ_t); the non-dimensional aerodynamic force coefficients C_x, C_z and moment coefficient C_m depend on angle of attack α and elevator deflection δ_e. The data of these coefficients are contained in lookup tables [3]. The two actuators have magnitude saturation limits according to Table 1.

Table 1. Actuator magnitude saturation

	Upper position	Lower position
Throttle δ_t	0	1
Elevator δ_e	-25°	+25°

Since the aerodynamic force and moment components depend on the angle of attack and the aircraft velocity, we replace the state variables U and W in the above equations by V_T and α according to the following relations:

$$\alpha = \tan^{-1}\left(\frac{W}{U}\right) \quad (9)$$

$$V_T = \sqrt{U^2 + W^2} \quad (10)$$

By taking the derivatives of the set of (9) and (10), the state vector becomes as follows: $x = [V_T\ \alpha\ \theta\ Q\ H\ POW\ ]^T$.

It will be mentioned later that the model is linearised, at each simulation iteration, with a small sampling time (50 ms). Note that the set of linearised models is only used for controller design, while all results in this paper are generated from the full nonlinear model. The differences between the design model and simulation model introduce model uncertainties, which can be referred to as the model/plant mismatch. The controller employed in flight control systems should therefore be insensitive to model uncertainties over the whole envelope of operation. That is, the design must be robust.

3 Longitudinal CAS

The aircraft modes can be divided into different categories. One category includes modes that involve the rotational degrees of freedom; these are the short-period, roll, and dutch-roll modes. The second category includes the phugoid mode that involves the translational degrees of freedom. The third category includes the spiral mode that depends on aerodynamic moments. The responsiveness of an aircraft to manoeuvring commands is determined by the speed of the rotational modes. The frequencies of these modes tend to be high that a pilot would find it difficult or impossible to control the aircraft if the modes were lightly damped or unstable. Therefore, it is necessary to design augmentation systems to control these modes, and to provide the pilot with a particular type of response to the control inputs. These systems are known as Control Augmentation Systems (CASs). Normally CASs are split into two control systems, to handle longitudinal and lateral problems, assuming negligible interaction. They are implemented by feedback controllers using accelerometers and rate gyros as sensors; and elevators, ailerons, or rudder as control surfaces.

In high-performance military aircraft, the pilot may have to perform tasks such as precision tracking of targets. In this situation, a suitable controlled variable is the pitch-rate (Q), which is required to follow a pilot's stick command. It has been found that a deadbeat response to pitch-rate commands is well suited to the task. Therefore, a specialised control augmentation system is needed, which is known as a "Pitch-rate Control Augmentation System". This system is conventionally designed for the longitudinal dynamics.

Since the performance specifications of aircraft are often given in terms of time-domain criteria such as the C* criterion and D* criterion [4, 5] and these criteria are close to the step response, we shall assume henceforth that the reference input

is a step command. Designing for such a command will yield suitable time-response characteristics. For the command tracking, the following design specifications must be satisfied:

1. A step response is required to reach 90% of the final value in less than 0.5 sec.
2. The overshoot must be less than 5%.

4 Nonlinear Generalized Predictive Control

An approach to nonlinear predictive control, utilising the optimal control trajectory, calculated in the previous time instant of the control algorithm, was employed by Kouvaritakis et al. [6]. An extension of the previous optimal trajectory to the current time instant was referred to as the "tail". The system is linearised around this trajectory and this linearised time-varying system was employed to obtain the optimal control, which is calculated as a perturbation from the "tail" trajectory. In Lee et al [7] a similar methodology employing linearisation at points of the seed trajectory was introduced, using a discrete time model representation of the system. In this paper we use this method with small changes. We will use finite time cost function and consequently we will not be able to guarantee stability. This is consequence of using the Generalized Predictive Control algorithm in its base version [8] to solve the minimisation problem.

4.1 The model

The aircraft model is given by the following differential equation:

$$\begin{aligned} \dot{x}(t) &= f(x(t),u(t)) \\ y(t) &= h(x(t)) = Cx(t) \end{aligned} \tag{11}$$

where
$x(t)$ is a vector of size n_x, $u(t)$ is a vector of size n_u, $y(t)$ is a vector of size n_y.

It is assumed that, within the working range of values for x, u, both $f(x(t),u(t))$, $h(x(t))$ are smooth functions and posses continuous first derivatives with respect to all their arguments. This assumption is fulfilled by the model inherently because this was obtained by interpolating the data of the lookup tables of the aerodynamic coefficients.

Next the model (11) is linearised around a particular operating point $(\bar{x}_n, \bar{u}_n)$ on the predicted trajectory:

$$\begin{aligned} \dot{z}(t) &= \bar{A}(\bar{x}_n,\bar{u}_n)z(t) + \bar{B}(\bar{x}_n,\bar{u}_n)v(t) \\ w(t) &= Cz(t) \end{aligned} \tag{12}$$

where

$$\bar{A}(\bar{x}_n,\bar{u}_n) = \left.\frac{\partial f(x(t),u(t))}{\partial x(t)}\right|_{x(t)=\bar{x}_n,u(t)=\bar{u}_n}$$

$$\bar{B}(\bar{x}_n,\bar{u}_n) = \left.\frac{\partial f(x(t),u(t))}{\partial u(t)}\right|_{x(t)=\bar{x}_n,u(t)=\bar{u}_n}$$

Between the original system (11) and the linearised system (12) the following relationships for states, outputs and controls can be established:

$$\begin{aligned} x(t) &= \bar{x}(t) + z(t) \\ y(t) &= \bar{y}(t) + w(t) \\ u(t) &= \bar{u}(t) + v(t) \end{aligned} \tag{13}$$

The system (12) is then discretised with sampling period T_s. Index "n" in $\bar{x}_n, \bar{u}_n$ denotes discrete time and $t = n \cdot T_s$. The following discrete-time state space model is obtained:

$$\begin{aligned} z_{n+1} &= \bar{A}_n^D z_n + \bar{B}_n^D v_n \\ w_n &= Cz_n \end{aligned} \tag{14}$$

Next integral action is introduced for the discrete time model (14):

$$\begin{aligned} \xi_{n+1} &= \bar{A}_n^I \xi_n + \bar{B}_n^I \Delta v_n \\ w_n &= C^I \xi_n \end{aligned} \tag{15}$$

where

$$\bar{A}_n^I = \begin{bmatrix} \bar{A}_n^D & \bar{B}_n^D \\ 0 & I \end{bmatrix}, \bar{B}_n^I = \begin{bmatrix} \bar{B}_n^D \\ I \end{bmatrix}, \bar{C}_n^I = \begin{bmatrix} \bar{C}_n^D & 0 \end{bmatrix}. \; \xi_n = \begin{bmatrix} z_n \\ v_{n-1} \end{bmatrix},$$
$$\Delta v_n = v_n - v_{n-1}$$

With the knowledge of predicted future trajectory, it is possible to approximate nonlinear system (11) by the linear time varying model. At each iteration of control algorithm, a set of future linear models is established and is used to calculate an optimal control sequence using a time-varying linear control law.

4.2 The controller

The cost function is given by the following equation:

$$\begin{aligned} J_n = &\sum_{j=1}^{N} \left\{ (r_{n+j}^{lin} - w_{n+j})^T \Lambda_E^j (r_{n+j}^{lin} - w_{n+j}) \right\} + \\ &\sum_{j=0}^{N-1} \left\{ \Delta v_{n+j}{}^T \Lambda_V^j \Delta v_{n+j} \right\} \end{aligned} \tag{16}$$

where r_n^{lin} is the setpoint vector of size n_y for linearised system at time n, $\Lambda_E^i \geq 0, i = 1...N$ and $\Lambda_V^j > 0, j = 0...N-1$ are weighting matrices and N is a positive integer number greater or equal to one.

It should be noticed that the cost function (16) is written in terms of linearised model variables and especially setpoint values for such model could be rather difficult to supply for the algorithm without a special consideration. This problem will be addressed later. For a time being a control law minimising the cost function (16) will be established and later relationship with the cost function written for the original (nonlinear) model variables will be shown.

Now the following vectors containing current and future values of control Δv_n, future values of state ξ_n, output w_n and setpoint r_n are introduced:

$$\Delta V_{n,N-1} = \begin{bmatrix} \Delta v_n \\ \Delta v_{n+1} \\ \vdots \\ \Delta v_{n+N-1} \end{bmatrix}, \qquad \xi_{n+1,N} = \begin{bmatrix} \xi_{n+1} \\ \xi_{n+2} \\ \vdots \\ \xi_{n+N} \end{bmatrix},$$
$$W_{n+1,N} = \begin{bmatrix} w_{n+1} \\ w_{n+2} \\ \vdots \\ w_{n+N} \end{bmatrix}, \qquad R_{n+1,N} = \begin{bmatrix} r_{n+1}^{lin} \\ r_{n+2}^{lin} \\ \vdots \\ r_{n+N}^{lin} \end{bmatrix} \tag{17}$$

The cost function (16) employing notation (17) can be written in a vector form as follows:

$$J_n = (R_{n+1,N} - W_{n+1,N})^T \Lambda_E (R_{n+1,N} - W_{n+1,N}) + \Delta V_{n,N-1}^T \Lambda_V \Delta V_{n,N-1} \tag{18}$$

with

$$\Lambda_E = diag(\Lambda_E^1, \Lambda_E^2, \cdots, \Lambda_E^N)$$
$$\Lambda_V = diag(\Lambda_V^0, \Lambda_V^1, \cdots, \Lambda_V^{N-1})$$

Now it is easy to determine a future state prediction in the following form:

$$\begin{aligned} \xi_{n+j} &= [\overline{A}_{n+j-1}^I \overline{A}_{n+j-2}^I ... \overline{A}_n^I]\xi_n + \\ &[\overline{A}_{n+j-1}^I \overline{A}_{n+j-2}^I ... \overline{A}_{n+1}^I]\overline{B}_n^I \Delta v_n + \\ &[\overline{A}_{n+j-1}^I \overline{A}_{n+j-2}^I ... \overline{A}_{n+2}^I]\overline{B}_{n+1}^I \Delta v_{n+1} + ... \\ &+ \overline{A}_{n+j-1}^I \overline{B}_{n+j-2}^I \Delta v_{n+j-2} + \overline{B}_{n+j-1}^I \Delta v_{n+j-1} \end{aligned} \tag{19}$$

Note that to obtain state prediction at time instance $n+j$ knowledge of matrices predictions $\overline{A}_n^I ... \overline{A}_{n+j-1}^I$ and $\overline{B}_n^I ... \overline{B}_{n+j-1}^I$ is required.

Also the output prediction can be obtained easily from the output equation:

$$w_{n+j} = C^I \xi_{n+j} \tag{20}$$

Finally taking into consideration notation (17) and equations (19), (20), the following equation may be obtained:

$$W_{n+1,N} = \Phi_{n,N} A_n \xi_n + S_{n,N} \Delta V_{n,N-1} \tag{21}$$

where

$$\Phi_{n,N} = \begin{bmatrix} C^I \left[\prod_{k=1}^{0} \overline{A}_{n+k}^I \right] \\ C^I \left[\prod_{k=1}^{1} \overline{A}_{n+k}^I \right] \\ \vdots \\ C^I \left[\prod_{k=1}^{N-1} \overline{A}_{n+k}^I \right] \end{bmatrix}, \tag{22}$$

$$S1_{n,N} = \begin{bmatrix} C^I \left[\prod_{k=1}^{0} \overline{A}_{n+k}^I \right] \overline{B}_n^I & 0 & \cdots \\ C^I \left[\prod_{k=1}^{1} \overline{A}_{n+k}^I \right] \overline{B}_n^I & C^I \left[\prod_{k=2}^{1} \overline{A}_{n+k}^I \right] \overline{B}_{n+1}^I & \ddots \\ \vdots & \vdots & \ddots \\ C^I \left[\prod_{k=1}^{N-1} \overline{A}_{n+k}^I \right] \overline{B}_n^I & C^I \left[\prod_{k=2}^{N-1} \overline{A}_{n+k}^I \right] \overline{B}_{n+1}^I & \cdots \end{bmatrix},$$
$$S2_{n,N} = \begin{bmatrix} \cdots & 0 \\ \ddots & \vdots \\ \ddots & 0 \\ \cdots & C^I \left[\prod_{k=N}^{N-1} \overline{A}_{n+k}^I \right] \overline{B}_{n+N-1}^I \end{bmatrix} \tag{23}$$
$$S_{n,N} = \begin{bmatrix} S1_{n,N} & S2_{n,N} \end{bmatrix}$$

and

$$\left[\prod_{k=l}^{m} \overline{A}_k^I \right] \equiv \begin{cases} \overline{A}_m^I \overline{A}_{m-1}^I ... \overline{A}_l^I & if \quad l \leq m \\ I & if \quad l > m \end{cases} \tag{24}$$

Substituting $W_{n+1,N}$ in (18) by equation (21) and performing static optimisation the following equation is obtained:

$$\Delta V_{n,N-1} = (\Lambda_V + S_{n,N}{}^T \Lambda_E S_{n,N})^{-1} S_{n,N} \Lambda_E \times \times \left(R_{n+1,N} - \Theta_{n,N} A_n \xi_n \right) \tag{25}$$

More detailed derivation of the above may be found in [9]. Equation (25) represents the explicit optimal control law of the NLGPC, which can be solved analytically without computational complexity.

Now consider the difference between outputs of the linearised and of the nonlinear model of the system. Using notation similar to (17), for the model (11):

$$U_{n,N-1}=\begin{bmatrix} u_n \\ u_{n+1} \\ \vdots \\ u_{n+N-1} \end{bmatrix}, \qquad X_{n+1,N}=\begin{bmatrix} x_{n+1} \\ x_{n+2} \\ \vdots \\ x_{n+N} \end{bmatrix},$$

$$Y_{n+1,N}=\begin{bmatrix} y_{n+1} \\ y_{n+2} \\ \vdots \\ y_{n+N} \end{bmatrix}, \qquad R^{PLANT}_{n+1,N}=\begin{bmatrix} r_{n+1} \\ r_{n+2} \\ \vdots \\ r_{n+N} \end{bmatrix}$$

Taking into account that predicted trajectory (for linearised model) is different from the actual trajectory (for the nonlinear system), the updated future output prediction of nonlinear system which results from (13) is given by:

$$Y_{n+1,N}=\bar{Y}_{n+1,N}+W_{n+1,N} \tag{26}$$

where

$\bar{Y}_{n+1,N}=[\bar{y}_{n+1} \quad \bar{y}_{n+2} \quad \cdots \quad \bar{y}_{n+N}]^T$ is the prediction of future output trajectory from previous iteration. Note that the last prediction is extended to obtain the required size of the vector.

Now consider that the original nonlinear system (11) is to be controlled. It is desired that the output of the system (11) follows trajectory $R^{Plant}_{n+1,N}$. This would result from the minimisation of the following error:

$$E_n=(R^{Plant}_{n+1,N}-Y_{n+1,N})^T\,\Lambda_E\,(R^{Plant}_{n+1,N}-Y_{n+1,N}) \tag{27}$$

Taking into consideration (26), the error equation (27) after rearrangement can be rewritten as follows:

$$E_n=(R_{n+1,N}-W_{n+1,N})^T\,\Lambda_E\,(R_{n+1,N}-W_{n+1,N}) \tag{28}$$

where

$$R_{n+1,N}=R^{Plant}_{n+1,N}-\bar{Y}_{n+1,N} \tag{29}$$

$R_{n+1,N}$ is the future setpoint vector for the model linearised around the trajectory and is used within the control algorithm. Tracking error (28) is kept low by minimising of the cost function (16) provided that reference signal for the linearised model $R_{n+1,N}$ is calculated using (29).

5 Simulation Results

The problem of dealing with input constraints has been solved by approximating these constraints by means of smooth limiting function and including them into the dynamics of the plant [10]. The error function *erf* and the sigmoid function *S* have been used to approximate the actuators magnitude saturation constraints. The input constraints are included into the plant dynamics (11) by replacing δ_t and δ_e with $S(\delta_t)$ and $25erf(\frac{\delta_e}{25})$ respectively. (see [10] for more details)

The NLGPC tuning parameters were set to N = 19 (control horizon = prediction horizon), Λ_E = 20, the sampling time T_s = 50 ms, and $\Lambda_V=\begin{bmatrix}\Lambda_{\delta_t} & 0\\ 0 & \Lambda_{\delta_e}\end{bmatrix}=\begin{bmatrix}1e-5 & 0\\ 0 & 2.5e-5\end{bmatrix}$. The initial conditions are chosen to be the forward *c.g.* steady-state flight trimmed conditions according to [3]; *i.e.* x_t^T = [502 0.03936 0.03936 0 0 9.64359]^T and u_{t-1}^T = $[0.1485 \quad -1.931]^T$. As well known, the throttle control input δ_t has a negligible effect on the pitch-rate Q and is used only to update the engine-power-level and consequently the thrust. Therefore, we concern in the elevator control δ_e. As mentioned before, our design is based on step-response shaping, where the setpoint trajectory is given as:

$$Q_{com}=\begin{cases} 0\ [rad/\sec], & t<0.5\sec \\ 0.3[rad/\sec], & t\geq 0.5\sec \end{cases} \tag{30}$$

Figure (2) shows a deadbeat output trajectory with the rise-time (to reach 90% of the final value) is 0.143 sec, while Figure (3) shows the input trajectory.

Although our design is based on step-response shaping, we prove that the resulting control system, if properly designed, will give good time responses for any arbitrary reference command signal by using another dynamic trajectory as given by:

$$Q_{com}=\begin{cases} 0 & t<0.25 \\ 1.14(t-0.25) & t\geq 0.25 \\ 0.4 & t\geq 0.6 \\ 0.4\cos(\pi\dfrac{t-0.85}{0.35}) & t\geq 0.85 \\ 0.1 & t\geq 1 \end{cases} \tag{31}$$

The simulation results for this particular output trajectory and for the associated input one are shown in Figures (4-5). Figures (2,4) show that the nonlinear controller works properly as the output time responses for the closed-loop system follow the tracking signals.

6 Conclusions

A receding horizon control algorithm, which optimises a tracking performance cost for nonlinear systems with input constraints was proposed. The optimisation procedure is based on performing a dynamic linearisation around trajectory, by taking into account the changes in the system model that are likely to occur due to the nonlinearities. The aims of this paper

were to introduce the concepts of MBPC and its application to multivariable flight control systems by designing of a controller that can cope with modelling errors and, plant parameters variations, as well as provides good performance.

The simulation results indicate that issues like handling qualities and tracking capabilities may be addressed directly for all flight conditions, using this type of flight control system. This should improve levels of performance over conventional flight controller designs.

Acknowledgements

We are grateful for the support of the Engineering and Physical Science Research Council (EPSRC) grant Non-linear Predictive Control and Industrial Applications GR/N05482E468 and Industrial Non-linear Control and Applications GR/R04683/01. We also wish to thank Prof. Basil Kouvaritakis and Dr. Mark Cannon for their comments and advice.

REFERENCES

[1] A.W. Babister, "Aircraft Dynamic Stability and Response," Pergamon Press, 1980.

[2] J. H. Blakelock, "Automatic Control of Aircraft and Missiles," John Wiley & Sons, 1965.

[3] B. Stevens and F. Lewis, "Aircraft Control and Simulation," John Wiley & Sons, 1992.

[4] E. G. Rynaski, "Flying Qualities in the Time Domain," AIAA Guid., Navig. Control Conf., Snowmass, Colo., pp. 85-1849, Aug. 1985.

[5] H. N. Tobie, E. M. Elliot and L. G. Malcom, "A New Longitudinal Handling Qualities Criterion," in *Proc. National Aerospace Electronics Conf.*, Dayton, Ohio, 1966, pp. 93-99.

[6] B. Kouvaritakis, M. Cannon, and J. A. Rossiter, "Non-linear model based predictive control," Int. J. Control, vol. 72, no. 10, pp. 919-928, 1999.

[7] Y.I. Lee, B. Kouvaritakis, and M. Cannon, "Constrained receding horizon predictive control for nonlinear systems," *Automatica*, vol.38, no.12, pp. 2093-2102, 2002.

[8] A. W. Ordys, and D. W. Clarke, "A state-space description for GPC controllers," *Int. J. Systems SCI.*, vol.24, NO.9, pp. 1727-1744, 1993.

[9] M. J. Grimble, A. W. Ordys, "Non-linear Predictive Control for Manufacturing and Robotic Applications," in *Proc. IEEE Conference on Methods and Models in Automation and Robotics*, Poland, Aug. 2001, pp. 579-593.

[10] A. M. Youssef, A. W. Ordys, and M. J. Grimble, "Nonlinear Predictive Control for Fast Constrained Systems", submitted for publication in *IEE Proceedings: Control Theory & Applications* (Special Issue on: Nonlinear Model Predictive Control).

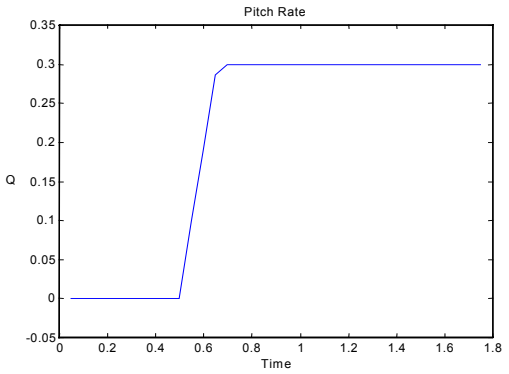

Figure (2): Setpoint and output trajectories

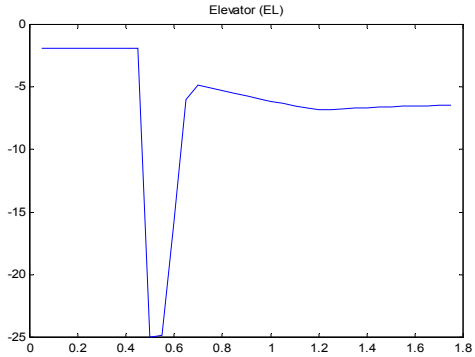

Figure (3): Control input

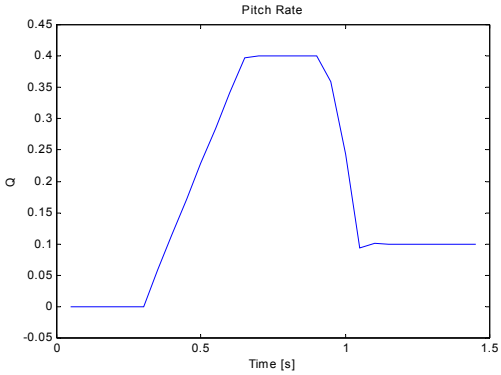

Figure (4): Setpoint and output trajectories

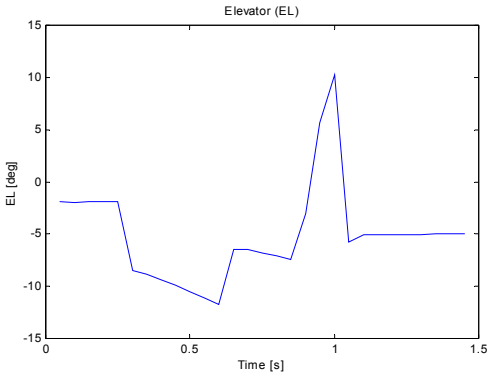

Figure (5): Control input