Aerospace Science and Technology 77 (2018) 677–687

Contents lists available at ScienceDirect

Aerospace Science and Technology

www.elsevier.com/locate/aescte

Mode decoupling robust eigenstructure assignment applied to the lateral-directional dynamics of the F-16 aircraft

Onur Albostan *, Metin Gökaşan

Control and Automation Engineering Department, Istanbul Technical University, 34469, Istanbul, Turkey

ARTICLE INFO

Article history:
Received 17 November 2017
Received in revised form 1 April 2018
Accepted 5 April 2018
Available online 9 April 2018

Keywords:
Robustness
Sensitivity
Mode decoupling
Nelder–Mead optimization
Flight control

ABSTRACT

In this paper, a robust controller is designed which is based on the eigenstructure assignment technique. With this technique, the physical modes of the system are decoupled by the appropriate shaping of the eigenvectors. While keeping the desired eigenvector structure as much as possible, Nelder–Mead optimization algorithm is implemented for the search of the optimal robust solution. Different optimization cost functions are investigated in terms of eigenvalue sensitivity and stability robustness. Design solutions are compared with each other and with a fundamental solution in terms of robustness measures. Furthermore, a small amount of flexibility is included in the selection of the eigenvalue locations and the improvement in the robustness is evaluated by singular value analysis. The effectiveness of the control system is demonstrated with nonlinear simulations on the F-16 aircraft mathematical model.

1. Introduction

Eigenstructure assignment is a well-studied MIMO control system design technique. Several methods have been proposed over four decades. While some researchers [1] were more focused on robustness problem, others showed more interest in mode decoupling problem [2,3]. More recent papers [4–6] handled the control problem both in terms of mode decoupling and robustness. Sobel et al. stated that the problem should be defined as robustness improvement instead of robustness optimization [3]. This proposition follows the strict adaptation of eigenvector structure, such that the possibility of robustness optimization is not available.

In most of the studies, the stability robustness at the input (plant actuators) had been investigated in addition to the eigenvalue sensitivity [4]. Doyle suggested that evaluating the stability robustness in the input node is not sufficient and the designer should also put emphasis on the stability margins at the output node [7]. While evaluating the stability robustness, singular value theory had been employed and the use of return difference matrix was introduced [8]. However, the stability margins computed from the return difference matrix found to be conservative and the inverse return difference matrix had been included in order to reduce the conservatism [9,10].

This paper takes advantage of the previous techniques and searches for the optimal stability robustness of the control loops both in input and output nodes of the system. Several design alternatives are evaluated in order to demonstrate the improvement in stability robustness. The eigenvalue locations are also included in the optimization problem and the solution is acquired without violating the one step procedure. The measures of stability robustness are based on the stability margins of each control loop which are determined by the minimum singular value of the return difference matrix and the inverse return difference matrix. Robustness analysis are conducted with simplified second order sensor models while the simulations include the full sensor models. The resultant feedback matrices are computed with the linearized lateral-directional dynamic equations of the F-16 aircraft [11,12]. Finally, the effectiveness of the control laws is tested by roll rate tracking, bank-to-bank and snap roll maneuvers with nonlinear simulations.

2. F-16 aircraft model description

Aerodynamic model is based on wind tunnel test results of F-16 at NASA Langley and Ames Research center. This database is valid for angle of attack values of $-20°$ to $90°$, angle of sideslip values of $\pm 30°$ and the true airspeed values smaller than 0.6 Mach flight conditions [11].

Engine model has been created according to the studies in [12]. Engine model data is valid for true airspeed values smaller than 1 Mach and pressure altitude values smaller than 15240 m.

* Corresponding author.
E-mail addresses: onuralbostan@itu.edu.tr (O. Albostan), metingokasan@itu.edu.tr (M. Gökaşan).

https://doi.org/10.1016/j.ast.2018.04.011
1270-9638/

Nomenclature

A	system matrix	LQR	linear quadratic regulator
B	control matrix	NM	Nelder–Mead
K	state feedback matrix	α	angle of attack
V	right eigenvector matrix	$\emptyset$	roll angle
Y	left eigenvector matrix	θ	pitch angle
p_s	stability axis roll rate	ψ	yaw angle
r_s	stability axis yaw rate	μ	bank angle
β	sideslip angle	γ	flight path angle
$\underline{\sigma}$	minimum singular value	χ	heading angle
c_i	eigenvalue sensitivity	δ_e	elevator deflection
$\kappa_2(V)$	condition number of the matrix V	δ_a	aileron deflection
$\mathcal{N}$	null space	δ_r	rudder deflection
MDREA	mode decoupling robust eigenstructure assignment	δ_{LEF}	leading edge flap deflection
CSAS	control and stability augmentation system		

Table 1
Actuator model parameters.

Control	Position limit	Rate limit	Time constant
δ_e	±25°	60°/s	0.0495 s
δ_a	±21.5°	80°/s	0.0495 s
δ_r	±30°	120°/s	0.0495 s
δ_{LEF}	0.25°	25°/s	0.136 s

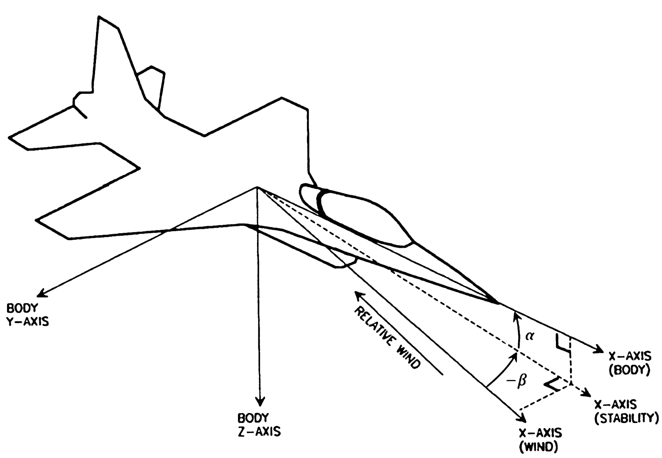

Fig. 1. Reference frames of F-16 aircraft.

Control surface actuators are modeled according to the parameters given in Table 1.

Leading edge flap actuator is modeled in an open loop structure. The leading edge flap position is scheduled with angle of attack and true airspeed. Reference frames are illustrated in Fig. 1.

Transformation among the reference frames can be stated as following.

$$F_O \xrightarrow{\chi,\gamma,\mu} F_W \xrightarrow{-\beta} F_S \xrightarrow{\alpha} F_B \quad \leftrightarrow \quad F_O \xrightarrow{\psi,\theta,\phi} F_B$$

The full sensor dynamics for output measurement in a typical flight control application are defined in [13]. Computation delay is taken as 10 ms and transformed into Laplace domain with 2nd order Pade approximation.

$$G_{air\_data\_sensor}(s) = 1/(1+0.02s)$$

$$G_{att\_sensor}(s) = 1/(1+0.00323s+0.00104s^2)$$

$$G_{rate\_sensor\_notch}(s) = \left(1-0.005346s+0.0001903s^2\right) / \left(1+0.03082s+0.0004942s^2\right)$$

$$G_{anti\_aliasing}(s) = 1/\left(1+0.00398s+0.0000158s^2\right)$$

$$G_{AD\_converter}(s) = (1-0.00208)/(1+0.00417s)$$

$$G_{delay}(s) = \left(1-0.0017s+8.33\cdot 10^{-6}s^2\right) / \left(1+0.0017s+8.33\cdot 10^{-6}s^2\right)$$

$$G_{averaging}(s) = (1-0.00208)/(1+0.00417s)$$

$$G_{noise}(s) = 0.05/\left(1+0.0089+0.00042s^2\right) \times \left(0.053s/(1+0.053s)\right)$$

$$G_{air\_data}(s) = G_{air\_data\_sensor}(s)G_{anti\_aliasing}(s)G_{AD\_converter}(s) \times G_{delay}(s)$$

$$G_{att}(s) = G_{att\_sensor}(s)G_{anti\_aliasing}(s)G_{AD\_converter}(s)G_{delay}(s)$$

$$G_{rate\_1}(s) = G_{rate\_sensor\_notch}(s)G_{averaging}(s)G_{AD\_converter}(s) \times G_{delay}(s)$$

$$G_{rate}(s) = G_{rate\_1}(s)/\left(1-G_{rate\_1}(s)G_{noise}(s)\right)$$

$$G_{def}(s) = 5.17\cdot 10^{-6}/(1+0.0032s+5.17\cdot 10^{-6}s^2)$$

3. Eigenstructure assignment

Eigenstructure assignment is a linear control technique, which places the closed loop poles into desired locations; in addition, for multi input systems it also gives the designer a flexibility to manipulate the corresponding eigenvectors. For a linear time independent system with n states and m control inputs ($m < n$), the designer can arbitrarily assign the m elements of the each eigenvector [2].

3.1. Mode decoupling eigenstructure assignment

Physical modes of the dynamic systems can be decoupled from each other by canceling out the cross-coupled terms in each eigenvector [3,5].

Two symbolic examples are demonstrated below where x_1, $\ldots, x_4$ are the states and $v_1, \ldots, v_4$ are the corresponding eigenvectors of the closed loop system ($m = 2$).

$$\text{(a)} \begin{matrix} x_1 \rightarrow \\ x_2 \rightarrow \\ x_3 \rightarrow \\ x_4 \rightarrow \end{matrix} \underbrace{\begin{bmatrix} x \\ x \\ 0 \\ 0 \end{bmatrix}}_{v_1} \underbrace{\begin{bmatrix} x \\ x \\ 0 \\ 0 \end{bmatrix}}_{v_2} \underbrace{\begin{bmatrix} 0 \\ 0 \\ x \\ x \end{bmatrix}}_{v_3} \underbrace{\begin{bmatrix} 0 \\ 0 \\ x \\ x \end{bmatrix}}_{v_4}$$