ELSEVIER

Signal Processing 55 (1996) 313–320

SIGNAL PROCESSING

Decomposition solution of H_∞ filter gain in singularly perturbed systems

Xuemin Shen*, Li Deng

Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1

Received 16 June 1995; revised 19 April 1996 and 5 August 1996

Abstract

In this paper, we propose a decomposition numerical method for the solution of the H_∞ filter gain in singularly perturbed systems. The decomposition removes the ill-conditioning (stiffness) problems of singularly perturbed systems so that only low-order, well-defined subsystems are involved in algebraic computation. We have achieved the decomposition via the use of a nonsingular transformation, which is applied to the *Hamiltonian* form of the singularly perturbed H_∞ filtering system. An efficient Newton-type algorithm is used for the related computation. An F-8 aircraft application example is given to demonstrate the efficiency of the proposed method.

Zusammenfassung

In diesem Beitrag schlagen wir eine numerische Zerlegungsmethode zur Bestimmung der Verstärkung eines H_∞-Filters in singulär gestörten Systemen vor. Die Zerlegung beseitigt Probleme der schlechten Konditionierung (Steife) singulär gestörter Systeme, so daß nur wohldefinierte Untersysteme niedriger Ordnung an algebraischen Berechnungen beteiligt sind. Wir haben die Zerlegung durch die Verwendung einer nicht-singulären Transformation erreicht, welche auf die Hamilton-Form des singulär gestörten H_∞-Filtersystems angewandt wird. Ein effizienter Algorithms newtonscher Art wird für die betroffene Berechnung gebraucht. Als Beispiel wird eine Anwendung in einem F8-Flugzeug herangezogen, um die Wirksamkeit des vorgeschlagenen Verfahrens zu demonstrieren.

Résumé

Dans cet article, nous présentons une méthode de décomposition numérique pour trouver le gain de filtre H_∞ dans des systèmes perturbés singulièrement. La décomposition élimine les problèmes mal posés (rigidité) des systèmes perturbés singulièrement de sorte que seuls des sous-systèmes d'ordre peu élevé et bien définis sont nécessaires au calcul algébrique. Nous avons obtenu la décomposition en utilisant une décomposition non singulière, qui est appliquée à la forme hamiltonienne du système de filtrage perturbé singulièrement. Un algorithme efficient de type Newton est utilisé pour le calcul y relatif. Un exemple d'application à l'avion F-8 est donné, afin de montrer l'efficacité de la méthode proposée.

Keywords: H_∞ filter; Singular perturbation; Decomposition; Riccati equation

* Corresponding author. Tel: 519-885-1211, ext. 2691; fax: 519-746-3077; e-mail: xshen@crg4.uwaterloo.ca.

PII S0165-1684(96)00140-5

1. Introduction

The Kalman–Bucy filter works extremely well in cases where the dynamical system is accurately modeled. However, most real systems include model errors, arising from variation of parameters, noise, etc. In order to ensure a more robust filter design, recently, a new class of optimal filter has been developed using H_∞ minimum estimation error spectrum criterion [7, 15, 1, 2, 8, 16]. Unlike the celebrated Wiener and/or Kalman filtering design which minimizes the variance of the estimation error, the H_∞ estimator is designed to guarantee the smallest possible estimation error energy over all possible disturbances of fixed energy, i.e. the operator relating the noise signals and parameter uncertainties to the resulting estimation error should possess an H_∞ norm less than a prescribed positive value. Therefore, H_∞ filter is over-conservative, which results more robust in terms of model uncertainties and lack of statistical information on the exogenous signals.

H_∞ filtering and control problems for the singularly perturbed system have been studied in different set-up by many researchers [14, 17, 10, 4, 13]. Both the solutions of filter and regulator gains for the singularly perturbed systems are related to the solutions of the differential or algebraic Riccati-type equations. Due to the presence of small parasitic parameters, the Riccati-type equations are ill-defined and may be very difficult to be solved directly. In this paper we will present an efficient recursive numerical method for the solution of the H_∞ filter gain in a singularly perturbed system. Two important reasons for this study are: (a) to avoid an ill-defined numerical problem of singularly perturbed systems; (b) to reduce the size of required computations, generate solutions with any desired accuracy from decoupled slow and fast subsystems, and speed up the estimation process. The presented method is as follows. We first decompose the *Hamiltonian* form of the singularly perturbed system into two well-defined subsystems (slow and fast) so that the H_∞ filter gain is obtained by solving two reduced-order linear algebraic equations instead of the nonlinear ill-defined differential Riccati equation. We then apply an efficient Newton-type algorithm for the related computation. The proposed method produces a near-optimal solution of the H_∞ filter gain with an any desired order of accuracy, i.e. $O(\varepsilon^i)$, where ε is a small positive parasitic parameter, and i is the number of iteration. An F-8 aircraft application example is given to demonstrate the efficiency of the method.

2. Problem formulation

Consider a continuous-time linear system as

$$\dot{x}_1(t) = A_1 x_1(t) + A_2 x_2(t) + B_1 w(t), \tag{1}$$

$$\varepsilon \dot{x}_2(t) = A_3 x_1(t) + A_4 x_2(t) + B_2 w(t), \tag{2}$$

with a linear measurement

$$y(t) = C_1 x_1(t) + C_2 x_2(t) + v(t), \tag{3}$$

where state vector $x_1(t) \in \mathscr{R}^{n_1}$, state vector $x_2(t) \in \mathscr{R}^{n_2}$ and measurements $y(t) \in \mathscr{R}^p$, $w(t) \in \mathscr{R}^m$ and $v(t) \in \mathscr{R}^p$ are the system and measurement noise vectors. $A_i, i = 1, 2, 3, 4$, B_j and C_j, $j = 1, 2$, are system matrices of the appropriate dimensions. ε is a small scalar positive parameter. The system (1)–(3) is called singularly perturbed system in the sense that when ε is neglected (made zero) the order of the system is reduced from $n = n_1 + n_2$ to n_1. The system has 'slow' and 'fast' phenomena because state vector x_2 changes much faster than state vector x_1 near a boundary layer.

For the system (1)–(3), we want to design a filter to estimate system states. We are interested not necessarily in the estimation of $x(t) = [x_1^{\mathrm{T}}(t)\ x_2^{\mathrm{T}}(t)]$ but in the estimation of a linear combination of $x(t)$. Let a vector $z(t) \in \mathscr{R}^q$ be the linear combination of $x(t)$,

$$z(t) = G_1 x_1(t) + G_2 x_2(t). \tag{4}$$

The estimation system is illustrated in Fig. 1.

The measure of the filtering performance over a finite time interval $[0, T]$ is defined as a disturbance attenuation function

$$J_{af} = \frac{\int_0^{\mathrm{T}} \|z(t) - \hat{z}(t)\|_R^2 \,\mathrm{d}t}{\|x_0 - \hat{x}_0\|_{P_0^{-1}}^2 + \int_0^{\mathrm{T}} \{\|w(t)\|_{W^{-1}}^2 + \|v(t)\|_{V^{-1}}^2\} \,\mathrm{d}t}, \tag{5}$$

where $e(t) = z(t) - \hat{z}(t)$ is the estimation error, $(x_0 - \hat{x}_0)$ is the error in the state-estimate at the initial time $t = 0$, $\hat{x}_0$ is the initial state-estimate which

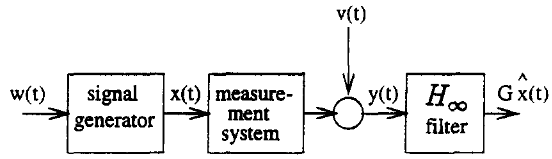

Fig. 1. The signal generating system.

is known, $p_0 > 0$ is a positive-definite matrix that reflects a priori knowledge as to how close the initial guess $\hat{x}_0$ is to x_0, $R \geqslant 0$, $W > 0$ and $V > 0$ are the weighting matrices which are chosen by the designer according to the performance requirements. The notation $\int_0^T \|s\|_R^2 \,\mathrm{d}t = \int_0^T (s^T R s)\,\mathrm{d}t$ is defined as the square of the weighted (by R) L_2 norm of s. The H_∞ filter is required to guarantee that the optimal estimate $z(t)$ among all possible $\hat{z}(t)$ (i.e. the worst-case performance measure) should satisfy

$$\sup J < \frac{1}{\gamma}, \tag{6}$$

where "sup" stands for supremum and $\gamma > 0$ is a prescribed level of noise attenuation.

It has been shown [15, 1, 12] that the H_∞ filter for the system (1)–(4) with performance measurement criterion (5) can be obtained for the maximum value of γ for which the Hamiltonian matrix

$$H_F = \begin{bmatrix} A & BWB^T \\ C^T V^{-1} C - \gamma G^T R G & -A^T \end{bmatrix} \tag{7}$$

has no eigenvalues on the imaginary axis, and if there exists a positive-definite solution to the following differential Riccati equation:

$$\dot{P}(t) = AP(t) + P(t)A^T + BWB^T - P(t)[C^T V^{-1} C - \gamma G^T R G]P(t), \quad P(0) = p_0. \tag{8}$$

Then an H_∞ filter is given by

$$\dot{\hat{x}}(t) = A\hat{x}(t) + P(t)C^T V^{-1}[y(t) - C\hat{x}(t)], \tag{9}$$

where

$$K(t) = P(t)C^T V^{-1} \tag{10}$$

is the gain of the H_∞ filter and

$$\hat{x}(t) = \begin{bmatrix} \hat{x}_1(t) \\ \hat{x}_2(t) \end{bmatrix}, \quad A = \begin{bmatrix} A_1 & A_2 \\ A_3/\varepsilon & A_4/\varepsilon \end{bmatrix},$$

$$B = \begin{bmatrix} B_1 \\ B_2/\varepsilon \end{bmatrix}, \quad C = [C_1\ C_2], \quad G = [G_1\ G_2].$$

The fact that H_F has no eigenvalues on the imaginary axis guarantees that $\sup J < 1/\gamma$ [3, p. 27], which suggests a way to find the largest γ: select a positive number γ; test if $\sup J < 1/\gamma$ by calculating the eigenvalues of H_F; increase or decrease γ accordingly; repeat. The existence of $P(t)\ \forall t \in [0, T]$ is essential for optimality.

It is important to note that the dependence on the linear combination of the states that we intend to estimate (i.e., the G_i) distinguishes the H_∞ filter from the Kalman filter. In Kalman filtering, the optimal estimator produces the best estimate of all the states, independent of G. In the H_∞ filtering, the optimal estimator produces the best estimate of that particular combination of states whose estimate is sought, i.e., the H_∞ filter is specifically tuned toward the linear combination $G_i x_i$.

The presence of the small parasitic parameter ε makes this problem numerically ill-defined, producing a so-called numerical stiff problem. In order to overcome this difficulty and obtain an efficient numerical method for solving Eq. (8), we will utilize the *Hamiltonian* form (7) for the solution of the differential Riccati-type equation and a nonsingular Chang transformation [2]. The *Hamitonian* form can 'linearize' the differential Riccati equation and the Chang transformation is used to block diagonalize the *Hamiltonian*, so that the required solution of the Riccati equation is obtained in terms of reduced-order problems. In addition, an efficient Newton-type algorithm [6] (with quadratic rate of convergence, i.e. $O(\varepsilon^{2^i})$, where i is the number of iterations) is used to solve the algebraic equations, which results in forming the Chang transformation.

3. Decomposition solution of the H_∞ filter gain

Consider the pair of linear matrix differential equations

$$\dot{M}(t) = AM(t) + BWB^{\mathrm{T}}N(t), \quad M(t_0) = P_0, \tag{11}$$

$$\dot{N}(t) = [C^{\mathrm{T}}V^{-1}C - \gamma G^{\mathrm{T}}RG]M(t) - A^{\mathrm{T}}N(t),$$
$$N(t_0) = I. \tag{12}$$

This pair of equations can also be written into the *Hamiltonian* system

$$\begin{bmatrix} \dot{M}(t) \\ \dot{N}(t) \end{bmatrix} = \begin{bmatrix} A & BWB^{\mathrm{T}} \\ C^{\mathrm{T}}V^{-1}C - \gamma G^{\mathrm{T}}RG & -A^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} M(t) \\ N(t) \end{bmatrix}. \tag{13}$$

Since the initial value problem is linear, the solution of the pair equations is assumed to be of the form

$$M(t) = P(t)N(t). \tag{14}$$

Differentiating this form we obtain

$$\dot{M}(t) = \dot{P}(t)N(t) + P(t)\dot{N}(t). \tag{15}$$

Using Eqs. (11) and (12), this becomes

$$AM(t) + BWB^{\mathrm{T}}N(t) = \dot{P}(t)N(t) + P(t)[C^{\mathrm{T}}V^{-1}C - \gamma G^{\mathrm{T}}RG]M(t) - P(t)A^{\mathrm{T}}N(t). \tag{16}$$

Substituting (14) and collecting terms produces the equation

$$\{\dot{P}(t) - P(t)A^{\mathrm{T}} - AP(t) + P(t)[C^{\mathrm{T}}V^{-1}C - \gamma G^{\mathrm{T}}RG]P(t) - BWB^{\mathrm{T}}\}N(t) = 0. \tag{17}$$

If $N(t)$ is assumed to be nonsingular for all $T \geqslant t \geqslant t_0$, this equation is equivalent to (8), and $P(t)$ can be obtained by

$$P(t) = M(t)N^{-1}(t). \tag{18}$$

Note that

$$P(t_0) = M(t_0)N^{-1}(t_0) = P_0 I = P_0, \tag{19}$$

thus the initial condition is also satisfied.

However, $N(t)$ may be close to singular for $t_0 \leqslant t \leqslant T$, which can cause the numerical instabilities associated with (18). A reinitialization technique (such as the Modified Davison–Maki Algorithm) [9] can be applied to solve the invertibility problem of $N(t)$ whenever necessary.

For singularly perturbed systems, we know the nature of the solution of (8), which is properly scaled as [6]

$$P(t) = \begin{bmatrix} P_1(t) & P_2(t) \\ P_2^{\mathrm{T}}(t) & \dfrac{P_3}{\varepsilon}(t) \end{bmatrix}, \quad P(t_0) = \begin{bmatrix} P_1(t_0) & P_2(t_0) \\ P_2^{\mathrm{T}}(t_0) & \dfrac{P_3}{\varepsilon}(t_0) \end{bmatrix}, \tag{20}$$

where $\dim P_1 = n_1 \times n_1$, $\dim P_3 = n_2 \times n_2$, $n_1 + n_2 = n$ (n_1 – slow variables, n_2 – fast variables).

We introduce compatible partitions of $M(t)$ and $N(t)$ matrices:

$$M(t) = \begin{bmatrix} M_1(t) & M_2(t) \\ M_3(t) & M_4(t) \end{bmatrix}, \quad N(t) = \begin{bmatrix} N_1(t) & N_2(t) \\ \varepsilon N_3(t) & \varepsilon N_4(t) \end{bmatrix}. \tag{21}$$

Partitioning (11) and (12), according to (21), will reveal a decoupled structure, that is, equations for $M_1(t), M_3(t), N_1(t)$ and $N_3(t)$ are independent of equations for $M_2(t), M_4(t), N_2(t)$ and $N_4(t)$ and vice versa:

$$\begin{bmatrix} \dot{M}_1(t) \\ \dot{M}_3(t) \\ \dot{N}_1(t) \\ \varepsilon\dot{N}_3(t) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 & Q_1 & Q_2/\varepsilon \\ A_3/\varepsilon & A_4/\varepsilon & Q_2^{\mathrm{T}}/\varepsilon & Q_3/\varepsilon^2 \\ S_1 & S_2 & -A_1^{\mathrm{T}} & -A_3^{\mathrm{T}}/\varepsilon \\ S_2^{\mathrm{T}} & S_3 & -A_2^{\mathrm{T}} & -A_4^{\mathrm{T}}/\varepsilon \end{bmatrix} \times \begin{bmatrix} M_1(t) \\ M_3(t) \\ N_1(t) \\ \varepsilon N_3(t) \end{bmatrix} = \mathscr{H} \begin{bmatrix} M_1(t) \\ M_3(t) \\ N_1(t) \\ \varepsilon N_3(t) \end{bmatrix}, \tag{22}$$

$$\begin{bmatrix} \dot{M}_2(t) \\ \dot{M}_4(t) \\ \dot{N}_2(t) \\ \varepsilon\dot{N}_4(t) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 & Q_1 & Q_2/\varepsilon \\ A_3/\varepsilon & A_4/\varepsilon & Q_2^{\mathrm{T}}/\varepsilon & Q_3/\varepsilon^2 \\ S_1 & S_2 & -A_1^{\mathrm{T}} & -A_3^{\mathrm{T}}/\varepsilon \\ S_2^{\mathrm{T}} & S_3 & -A_2^{\mathrm{T}} & -A_4^{\mathrm{T}}/\varepsilon \end{bmatrix} \times \begin{bmatrix} M_2(t) \\ M_4(t) \\ N_2(t) \\ \varepsilon N_4(t) \end{bmatrix} = \mathscr{H} \begin{bmatrix} M_2(t) \\ M_4(t) \\ N_2(t) \\ \varepsilon N_4(t) \end{bmatrix}, \tag{23}$$

where

$$Q = BWB^{\mathrm{T}} = \begin{bmatrix} B_1WB_1^{\mathrm{T}} & B_1WB_2^{\mathrm{T}}/\varepsilon \\ B_2WB_1^{\mathrm{T}}/\varepsilon & B_2WB_2^{\mathrm{T}}/\varepsilon^2 \end{bmatrix} = \begin{bmatrix} Q_1 & Q_2/\varepsilon \\ Q_2^{\mathrm{T}}/\varepsilon & Q_3/\varepsilon^2 \end{bmatrix},$$

$$S = C^{\mathrm{T}}V^{-1}C - \gamma G^{\mathrm{T}}RG = \begin{bmatrix} C_1^{\mathrm{T}}V^{-1}C_1 - \gamma G_1^{\mathrm{T}}RG_1 & C_1^{\mathrm{T}}V^{-1}C_2 - \gamma G_1^{\mathrm{T}}RG_2 \\ C_2^{\mathrm{T}}V^{-1}C_1 - \gamma G_2^{\mathrm{T}}RG_1 & C_2^{\mathrm{T}}V^{-1}C_2 - \gamma G_2^{\mathrm{T}}RG_2 \end{bmatrix} = \begin{bmatrix} S_1 & S_2 \\ S_2^{\mathrm{T}} & S_3 \end{bmatrix}.$$

Interchanging second and third rows in (22) and (23), respectively, produces

$$\begin{bmatrix} \dot{M}_1(t) \\ \dot{N}_1(t) \\ \varepsilon\dot{M}_3(t) \\ \varepsilon\dot{N}_3(t) \end{bmatrix} = \begin{bmatrix} A_1 & Q_1 & A_2 & Q_2 \\ S_1 & -A_1^{\mathrm{T}} & S_2 & -A_3^{\mathrm{T}} \\ A_3 & Q_2^{\mathrm{T}} & A_4 & Q_3 \\ S_2^{\mathrm{T}} & -A_2^{\mathrm{T}} & S_3 & -A_4^{\mathrm{T}} \end{bmatrix} \times \begin{bmatrix} M_1(t) \\ N_1(t) \\ M_3(t) \\ N_3(t) \end{bmatrix} = \begin{bmatrix} T_1 & T_2 \\ T_3 & T_4 \end{bmatrix} \begin{bmatrix} M_1(t) \\ N_1(t) \\ M_3(t) \\ N_3(t) \end{bmatrix} \tag{24}$$

and

$$\begin{bmatrix} \dot{M}_2(t) \\ \dot{N}_2(t) \\ \varepsilon\dot{M}_4(t) \\ \varepsilon\dot{N}_4(t) \end{bmatrix} = \begin{bmatrix} A_1 & Q_1 & A_2 & Q_2 \\ S_1 & -A_1^{\mathrm{T}} & S_2 & -A_3^{\mathrm{T}} \\ A_3 & Q_2^{\mathrm{T}} & A_4 & Q_3 \\ S_2^{\mathrm{T}} & -A_2^{\mathrm{T}} & S_3 & -A_4^{\mathrm{T}} \end{bmatrix} \times \begin{bmatrix} M_2(t) \\ N_2(t) \\ M_4(t) \\ N_4(t) \end{bmatrix} = \begin{bmatrix} T_1 & T_2 \\ T_3 & T_4 \end{bmatrix} \begin{bmatrix} M_2(t) \\ N_2(t) \\ M_4(t) \\ N_4(t) \end{bmatrix}, \tag{25}$$

where

$$T_1 = \begin{bmatrix} A_1 & Q_1 \\ S_1 & -A_1^{\mathrm{T}} \end{bmatrix}, \qquad T_2 = \begin{bmatrix} A_2 & Q_2 \\ S_2 & -A_3^{\mathrm{T}} \end{bmatrix},$$

$$T_3 = \begin{bmatrix} A_3 & Q_2^{\mathrm{T}} \\ S_2^{\mathrm{T}} & -A_2^{\mathrm{T}} \end{bmatrix}, \qquad T_4 = \begin{bmatrix} A_4 & Q_3 \\ S_3 & -A_4^{\mathrm{T}} \end{bmatrix}.$$

It is important to note that both (24) and (25) retain the singular perturbation form. Introducing notations

$$U(t) = \begin{bmatrix} M_1(t) \\ N_1(t) \end{bmatrix} = \begin{bmatrix} U_1(t) \\ U_2(t) \end{bmatrix}, \tag{26a}$$

$$Z(t) = \begin{bmatrix} M_3(t) \\ N_3(t) \end{bmatrix} = \begin{bmatrix} Z_1(t) \\ Z_2(t) \end{bmatrix}, \tag{26b}$$

$$X(t) = \begin{bmatrix} M_2(t) \\ N_2(t) \end{bmatrix} = \begin{bmatrix} X_1(t) \\ X_2(t) \end{bmatrix}, \quad Y(t) = \begin{bmatrix} M_4(t) \\ N_4(t) \end{bmatrix} = \begin{bmatrix} Y_1(t) \\ Y_2(t) \end{bmatrix}, \tag{27}$$

we obtain two singularly perturbed system matrix equations

$$\dot{U}(t) = T_1U(t) + T_2Z(t), \quad \varepsilon\dot{Z}(t) = T_3U(t) + T_4Z(t), \tag{28}$$

$$\dot{X}(t) = T_1X(t) + T_2Y(t), \quad \varepsilon\dot{Y}(t) = T_3X(t) + T_4Y(t), \tag{29}$$

with new initial conditions

$$U(t_0) = \begin{bmatrix} P_1(0) \\ I_{n_1} \end{bmatrix}, \qquad Z(t_0) = \begin{bmatrix} P_2^{\mathrm{T}}(0) \\ 0 \end{bmatrix},$$

$$X(t_0) = \begin{bmatrix} P_2(0) \\ 0 \end{bmatrix}, \qquad Y(t_0) = \begin{bmatrix} P_3(0)/\varepsilon \\ I_{n_2}/\varepsilon \end{bmatrix}.$$

Note that systems (28) and (29) have exactly the same form and the only difference is the initial conditions.

In the sequel, we introduce the following transformation [2] defined by

$$J = \begin{bmatrix} I_{2n_1} - \varepsilon HL & -\varepsilon H \\ L & I_{2n_2} \end{bmatrix}, \quad J^{-1} = \begin{bmatrix} I_{2n_1} & \varepsilon H \\ -L & I_{2n_2} - \varepsilon LH \end{bmatrix}, \tag{30}$$

where matrices L and H satisfy

$$T_4L - T_3 - \varepsilon L(T_1 - T_2L) = 0, \tag{31}$$

$$-H(T_4 + \varepsilon LT_2) + T_2 + \varepsilon(T_1 - T_2L)H = 0. \tag{32}$$

The matrices L and H can be obtained by using the recursive Newton-type algorithm [6] with quadratic rate of convergence, under the condition that T_4 is nonsingular when $\varepsilon = 0$. The algorithm is briefly summarized in the following.

Assumption. The pair (A_4, B_2) is controllable and (A_4, C_2) is observable.

Under this assumption, matrix T_4 is nonsingular [5]. For the algebraic equation (31) the initial guess is easily obtained to $O(\varepsilon)$ accuracy, by setting $\varepsilon = 0$ in the equation, that is

$$L^{(0)} = T_4^{-1}T_3 = L + O(\varepsilon). \tag{33}$$

Thus the Newton sequence will be $O(\varepsilon^2)$, $O(\varepsilon^4)$, $O(\varepsilon^8), \ldots, O(\varepsilon^{2^i})$ close to the exact solution, respectively, in each iteration.

The Newton-type algorithm for solving (31) can be constructed by setting $L^{(i+1)} = L^{(i)} + \Delta L^{(i)}$ and neglecting $O(\Delta L)^2$ terms. This will produce a Lyapunov-type equation of the form

$$D_1^{(i)}L^{(i+1)} + L^{(i+1)}D_2^{(i)} = E^{(i)}, \tag{34}$$

where

$$D_1^{(i)} = T_4 + \varepsilon L^{(i)}T_2, \qquad D_2^{(i)} = -\varepsilon(T_1 - T_2L^{(i)}),$$

$$E^{(i)} = T_3 + \varepsilon L^{(i)}T_2L^{(i)}, \quad i = 0, 1, 2, \ldots,$$

with the initial condition given by (33).

Having found the solution of (31), up to the required degree of accuracy, one can get the solution of (32) by solving directly a Lyapunov equation of the form

$$H^{(i)}D_1^{(i)} + D_2^{(i)}H^{(i)} = T_2, \tag{35}$$

which implies $H^{(i)} = H + O(\varepsilon^{2^i})$.

The transformation (30) is then applied to (28) and (29):

$$\begin{aligned} \begin{bmatrix} \underline{U}(t) \\ \underline{Z}(t) \end{bmatrix} &= J \begin{bmatrix} U(t) \\ Z(t) \end{bmatrix}, \\ \begin{bmatrix} \underline{X}(t) \\ \underline{Y}(t) \end{bmatrix} &= J \begin{bmatrix} X(t) \\ Y(t) \end{bmatrix}, \end{aligned} \tag{36}$$

with

$$\begin{bmatrix} \underline{U}(t_0) \\ \underline{Z}(t_0) \end{bmatrix} = J \begin{bmatrix} U(t_0) \\ Z(t_0) \end{bmatrix}, \quad \begin{bmatrix} \underline{X}(t_0) \\ \underline{Y}(t_0) \end{bmatrix} = J \begin{bmatrix} X(t_0) \\ Y(t_0) \end{bmatrix}.$$

This will produce two completely decoupled subsystems

$$\begin{aligned} &\dot{\underline{U}}(t) = (T_1 - T_2L)\underline{U}(t), \\ &\underline{U}(t_0) = (I_{2n_1} - \varepsilon HL)U(t_0) - \varepsilon HZ(t_0), \end{aligned} \tag{37}$$

$$\begin{aligned} &\varepsilon\dot{\underline{Z}}(t) = (T_4 + \varepsilon LT_2)\underline{Z}(t), \\ &\underline{Z}(t_0) = LU(t_0) + Z(t_0), \end{aligned} \tag{38}$$

$$\begin{aligned} &\dot{\underline{X}}(t) = (T_1 - T_2L)\underline{X}(t), \\ &\underline{X}(t_0) = (I_{2n_1} - \varepsilon HL)X(t_0) - \varepsilon HY(t_0), \end{aligned} \tag{39}$$

$$\begin{aligned} &\varepsilon\dot{\underline{Y}}(t) = (T_4 + \varepsilon LT_2)\underline{Y}(t), \\ &\underline{Y}(t_0) = LX(t_0) + Y(t_0). \end{aligned} \tag{40}$$

Solutions of (37)–(40) are given by

$$\begin{aligned} &\underline{U}(t) = e^{(T_1 - T_2L)t}\underline{U}(t_0), \\ &\underline{Z}(t) = e^{1/\varepsilon(T_4 + \varepsilon LT_2)t}\underline{Z}(t_0), \end{aligned} \tag{41}$$

$$\begin{aligned} &\underline{X}(t) = e^{(T_1 - T_2L)t}\underline{X}(t_0), \\ &\underline{Y}(t) = e^{1/\varepsilon(T_4 + \varepsilon LT_2)t}\underline{Y}(t_0). \end{aligned} \tag{42}$$

From Eq. (36) we have

$$\begin{aligned} \begin{bmatrix} U(t) \\ Z(t) \end{bmatrix} &= J^{-1} \begin{bmatrix} \underline{U}(t) \\ \underline{Z}(t) \end{bmatrix}, \\ \begin{bmatrix} X(t) \\ Y(t) \end{bmatrix} &= J^{-1} \begin{bmatrix} \underline{X}(t) \\ \underline{Y}(t) \end{bmatrix}. \end{aligned} \tag{43}$$

Then the solutions in the original coordinates are

$$U(t) = e^{(T_1 - T_2L)t}\underline{U}(t_0) + \varepsilon He^{1/\varepsilon(T_4 + \varepsilon LT_2)t}\underline{Z}(t_0), \tag{44}$$

$$\begin{aligned} Z(t) = &-Le^{(T_1 - T_2L)t}\underline{U}(t_0) \\ &+(I - \varepsilon LH)e^{1/\varepsilon(T_4 + \varepsilon LT_2)t}\underline{Z}(t_0), \end{aligned} \tag{45}$$

$$X(t) = e^{(T_1 - T_2L)t}\underline{X}(t_0) + \varepsilon He^{1/\varepsilon(T_4 + \varepsilon LT_2)t}\underline{Y}(t_0), \tag{46}$$

$$\begin{aligned} Y(t) = &-Le^{(T_1 - T_2L)t}\underline{X}(t_0) \\ &+(I - \varepsilon LH)e^{1/\varepsilon(T_4 + \varepsilon LT_2)t}\underline{Y}(t_0). \end{aligned} \tag{47}$$

Partitioning (44)–(47) according to (26)–(27) will produce all components of matrices $M(t)$ and $N(t)$, so that the required solution of (8) is given by

$$\begin{aligned} P(t) &= \begin{bmatrix} M_1(t) & M_2(t) \\ M_3(t) & M_4(t) \end{bmatrix} \begin{bmatrix} N_1(t) & N_2(t) \\ \varepsilon N_3(t) & \varepsilon N_4(t) \end{bmatrix}^{-1} \\ &= \begin{bmatrix} U_1(t) & X_1(t) \\ Z_1(t) & Y_1(t) \end{bmatrix} \begin{bmatrix} U_2(t) & X_2(t) \\ \varepsilon Z_2(t) & \varepsilon Y_2(t) \end{bmatrix}^{-1}. \end{aligned} \tag{48}$$

Thus, in order to obtain the solution of ill-defined differential Riccati equation (8), i.e. $P(t)$, one would have to take direct integration of stiff linear differential system (13) of size $2n$, which takes a total of $8n^3$ multiplications per time step [9, method 4]. In our approach, the total number of multiplication per time step is $16(n_1^3+n_2^3)$. When n_1 is close to n_2, the saving is about $32n_1^3$ multiplications per time step. It should be mentioned that although well-defined decoupled linear subsystems (37), (39) of size $2n_1$ are in slow time scale, and (38), (40) of size $2n_2$ are in fast time scale, after time scaling, we can use same integration step size to integrate (37)–(40) in parallel. Furthermore the integration step size for (13) is smaller than that for (37)–(40) due to the stiffness, the ratio of the step sizes depends on ε. In other words, in order to obtain the solution $P(t)$ over a time duration T, the number of step for computing (13) is much larger than that for (37)–(40). The smaller the ε is, the more compuatation is reduced by the decomposition approach.

The accuracy of $P(t)$ depends on the accurate solutions of L in (31) and H in (32). Using the Newton-type algorithm, we can obtain solutions of L and H with any desired accuracy. The computations for L and H are negligible as compared with that of solving (13) and (37)–(40), since they are independent from the time duration T. After having the solution of $P(t)$, the H_∞ filter gain can then be obtained from (10).

The proposed algorithm which presents a complete solution to our problem is as follows.

Step 1. Using (34) with (33) to calculate $L^{(i+1)}$ recursively, then solve (35) to obtain $H^{(i)}$;

Step 2. Calculate $\underline{U}(t_0), \underline{Z}(t_0), \underline{X}(t_0)$ and $\underline{Y}(t_0)$ from Eqs. (37)–(40);

Step 3. Calculate $U(t)$, $Z(t)$, $X(t)$ and $Y(t)$ from Eqs. (44)–(47);

Step 4: Calculate $P(t)$ from Eq. (48).

4. Application example

In order to demonstrate the proposed method for the solution of the H_∞ filter gains, we present an F-8 aircraft application example [6]. The linearized system model of the motion of the aircraft has four system states: horizontal-velocity deviation; the flight-path angle; the angle of attack and the pitch rate. The system matrices are

$$A_1 = \begin{bmatrix} 0.278386 & -0.965256 \\ 0.089833 & -0.290700 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} -0.074210 & 0.016017 \\ 0.012815 & -0.001398 \end{bmatrix},$$

$$A_3 = \begin{bmatrix} -0.001815 & 0.005873 \\ 0.002850 & -0.009223 \end{bmatrix},$$

$$A_4 = \begin{bmatrix} -0.030344 & 0.075024 \\ -0.075092 & -0.016777 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} -46.626960 \\ 7.858776 \end{bmatrix}, \qquad B_2 = \begin{bmatrix} -18.210002 \\ -45.049998 \end{bmatrix},$$

$$C_1 = \begin{bmatrix} 0 & 0 \\ 1 & -3.236 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0 & 0.005 \\ -0.003152 & 0.01302 \end{bmatrix}.$$

The matrices G_1 and G_2 represent the specific choice of the linear combination of states to be estimated,

$$G_1 = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}, \qquad G_2 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

The small parasitic parameter ε which is roughly the ratio of the magnitude of the slow eigenvalues to that of the fast eigenvalues, weighting matrices W, V and the noise attenuation constant γ are: $\varepsilon = 0.025$, $W = 0.000315$, $V = \text{diag}[0.000686, 40]$, $R = \text{diag}[1, 1]$ and γ=1.44. The initial condition $P(t_0) = \text{diag}[0.1, 0.1, 1.0, 1.0]$. With the proposed method, simulation result for the singularly perturbed matrix differential equation (8) is obtained by using the package PC-MATLAB for the computer-aided control system design [11].

Using (34) and (35), after 5 iterations, we obtain matrices $L^{(5)}$ and $H^{(5)}$:

$$L^{(5)} = \begin{bmatrix} -1.7309 & -3.5437 & -0.8503 & -0.1150 \\ 0.1298 & 1.4618 & 0.0372 & 0.0131 \\ 0.2899 & 0.6983 & 0.5196 & -0.0134 \\ -0.3089 & -0.6989 & 0.7262 & -0.1822 \end{bmatrix},$$

$$H^{(5)} = \begin{bmatrix} 0.5258 & 0.7365 & 0.8576 & -0.0358 \\ -0.0135 & -0.1839 & 0.1170 & -0.0133 \\ -0.2932 & 0.3133 & -1.7521 & 0.1303 \\ -0.6994 & 0.7057 & -3.5583 & 1.4669 \end{bmatrix}.$$

The initial time is selected as $t_0 = 0$. When $t = 0.5$, using (48) we obtain

$$P_{\mathrm{app}}(t = 0.5)$$

$$= \begin{bmatrix} 0.2795 & -0.0431 & 3.4242 & 3.3020 \\ -0.0431 & 0.0777 & -0.2456 & -0.4599 \\ 3.4242 & -0.2456 & 116.72 & 76.6439 \\ 3.3020 & -0.4599 & 76.6439 & 106.7942 \end{bmatrix}. \tag{49}$$

The obtained solution P_{app}, given by (49), is identical to the solution of the global Riccati differential equation (8) obtained by using any standard method [9]. However, in our method we have been using the reduced-order algorithm and the problem of ill-conditioning due to the singularly perturbed structure is eliminated. After getting the solution of (8), the H_∞ filter gain can be obtained by (10):

$$K(t = 0.5) = P_{\mathrm{app}}(t = 0.5)C^{\mathrm{T}}V^{-1}$$

$$= \begin{bmatrix} 24.0672 & 0.0113 \\ -3.3522 & -0.0075 \\ 558.6291 & 0.1212 \\ 778.3835 & 0.1485 \end{bmatrix}. \tag{50}$$

5. Conclusions

The H_∞ filter gain of singularly perturbed systems is obtained. Instead of solving nonlinear ill-defined differential Riccati equation, we determine the H_∞ filter gain directly from the decomposed *Hamiltonian* form of the system. The proposed method overcomes the stiffness problem and considerably reduces the amount of required computation.

Acknowledgements

The authors would like to thank Professor Zoran Gajic for his many valuable suggestions, the NSERC of Canada for the financial support of this work.

References

[1] R.N. Banavar and J.L. Speyer, “A linear quadratic game theory approach to estimation and smoothing”, *Proc. IEEE ACC*, Boston, MA, 26–28 June 1991, pp. 2818–2822.

[2] K. Chang, “Singular perturbations of a general boundary value problem”, *SIAM J. Math. Anal.*, Vol. 3, No. 3, August 1972, pp. 520–526.

[3] J.C. Doyle, B.A. Francis and A.R. Tannenbaum, *Feedback Control Theory*, Macmillan New York, 1992.

[4] V. Dragan, “Asymptotic expansions for game theoretic Riccati equations and stabilization with disturbance attenuation for singularly perturbed system”, *Systems Control Lett.*, Vol. 20, No. 6, June 1993, pp. 455–463.

[5] E. Fridman, Exact decomposition of singularly perturbed H_∞-optimal control problem, Technical Report, EE-S-93-16, Tel-Aviv University, Israel, 1993.

[6] Z. Gajic and X. Shen, *Parallel Algorithms for Optimal Control of Large Scale Linear Systems*, Springer, London, 1993.

[7] M.J. Grimble and A. Elsayed, “Solution of the H_∞ optimal linear filtering problem for discrete-time systems”, *IEEE Trans. Acoust. Speech Signal Process*, Vol. ASSP-38, No.7, July 1990, pp. 1092–1104.

[8] B. Hassibi and T. Kailath, “H_∞ adaptive filtering”, *Proc. IEEE Internat. Conf. Acoust. Speech Signal Process.-95*, Detroit, MI, 9–12 May 1995, 949–952.

[9] C. Kenney and R. Leipnik, “Numerical integration of the differential matrix riccati equation”, *IEEE Trans. Automat. Control*, Vol. AC-30, No. 10, October 1985, pp. 962–970.

[10] H.K. Khalil and F. Chen, “H_∞ control of two-time-scale systems”, *Systems Control Lett.*, Vol. 19, No. 1, July 1992, pp. 35–42.

[11] C. Moler, J. Little and S. Bamgert, *PC-MATLAB*, The Math works, Sherborn, MD, 1987.

[12] K.M. Nagpal and P.P. Khargonekar, “Filtering and smoothing in an H_∞ setting”, *IEEE Trans. Automat. Control*, Vol. AC-36, No. 2, February 1991, pp. 152–166.

[13] Z. Pan and T. Basar, “H_∞-optimal control for singularly perturbed systems: Part I: Perfect state measurements”, *Automatica*, Vol. 29, No. 2, March 1993, pp. 401–423.

[14] S.M. Shahruz, “Design of H_∞-optimal compensators for singularly perturbed system”, *Proc. 28th IEEE CDC*, Tampa, FL, 13–15 December 1989, pp. 2397–2398.

[15] U. Shaked, “H_∞-minimum error state estimation of linear stationary processes”, *IEEE Trans. Automat. Control*, Vol. AC-35, No. 5, May 1990, pp. 554–558.

[16] X. Shen and L. Deng, “Discrete H_∞ filter design with application to speech enhancement”, *Proc. IEEE Internat. Conf. Acoust. Speech Signal Process.-95*, Detroit, MI, 9–12 May 1995, pp. 1504–1507.

[17] J.L. Vian and M.E. Sawan, “H_∞ control for singularly perturbed system”, *Proc. 30th IEEE CDC*, Brighton, UK, 11–13 December 1991, pp. 1072–1074.