Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 10 (2018) 5-19

DOI: 10.2478/auseme-2018-0001

An Algorithm for the Calculation of the Dwell Time Constraint for Switched $\mathcal{H}_\infty$ Filters

Zsolt HORVÁTH[1], András EDELMAYER[2,3]

[1] School of Postgraduate Studies of Multidisciplinary Technical Sciences, Faculty of Technical Sciences, Széchenyi István University, Győr, e-mail: zsolt2.horvath@audi.hu
[2] Department of Informatics Engineering, Faculty of Technical Sciences, Széchenyi István University, Győr, e-mail: edelmayer@sze.hu
[3] Systems and Control Laboratory, Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest, e-mail: edelmayer@sztaki.mta.hu

Manuscript received September 28, 2018; revised November 30, 2018.

Abstract: This paper presents a numerical algorithm for determining the minimum dwell time constraint for switched linear $\mathcal{H}_\infty$ fault detection filters. When applying switched systems, ensuring the stability is a crucial target, which can be guaranteed, when we switch slowly enough between the subsystems, more precisely when the intervals between two consecutive switching instants, called dwell time, are large enough. The problem formulation is based on multiple Lyapunov functions and is expressed through a special form of linear matrix inequalities (LMIs), which include a nonlinear term of the dwell time. This represents a multivariable, time dependent optimization problem. As a result, the task cannot be treated as a simple feasibility problem involving a LMI solver as it is widely used in applications of the linear control. To solve these special LMIs, we propose a numerical algorithm, called $\boldsymbol{T_d}$-iteration, which combines the procedure of interval halving with an LMI solver. The algorithm implemented in MATLAB shows its applicability as well as suggest further benefits for the switched linear control and filter synthesis.

Keywords: Switched linear system, dwell time, switched $\mathcal{H}_\infty$ fault detection filter, MFARE

1. Introduction

Switched systems for purpose of nonlinear control have been studied extensively in the two past decades and useful results are now available, see e.g. [1], [2], [3], [4] and [5]. As it was stated by several authors e.g. (Liberzon and Morse in 1999, Hespana in 2004, Chen and Saif in 2004, Colaneri in 2008), the asymptotic stability can be ensured when we switch slowly enough between the subsystems, more precisely when the intervals between two consecutive switching instants -called dwell time-, are large enough. This problem has been specially addressed in the synthesis of switched state estimator of Luenberger type, e.g. (Prandini in 2003, Chen and Saif in 2004) and it is also a crucial part in our objective of designing a switched linear $\mathcal{H}_\infty$ fault detection filter. In earlier researches different methods have been proposed for determining the minimum dwell time, see [4], [6], [7], [8], [9] and [10]. The most commonly used algorithms, such as e.g. the representation based on Kronecker products (Geromel and Colaneri, 2006), or Logic-Based Switching Algorithms (Hespana, 1998) are constructed using multiple Lyapunov functions and expressed in form of linear matrix inequalities (LMIs), see in [6], [7], [9], [10] and [11].

Since we deal with $\mathcal{H}_\infty$ filtering, the basic Lyapunov theorem needs to be extended to cope with performance requirements such as the root mean square (RMS) property of a switched system, which corresponds to finding an upper bound of the minimum dwell time. To this aim, in our research we consider a method used by (Geromel and Colaneri, 2008) for $\mathcal{H}_\infty$ nonlinear control and we have adopted it to the classical $\mathcal{H}_\infty$ detection filtering problem, see in [12], [13], [14], [15] and [16]. More exactly, the concept of the switched $\mathcal{H}_\infty$ control in [7] can be associated to the switched $\mathcal{H}_\infty$ filtering problem by duality and sufficient stability conditions can be derived.

LMIs are nowadays widely used powerful tools for solving complex optimization problem in the field of control engineering, see e.g. [17], [18], [19] and [20]. The commonly used advanced methods, however, refer to a LMI solver only accept formulation where the decision variables are included in linear terms. On the contrary, our problem formulated as LMIs, which include the term of matrix-exponential function with the dwell time, is consequently nonlinear. As a result, the task cannot be treated as a simple feasibility problem, see e.g. [17], [21] and [22]. Despite of the widespread referring to this special LMI formulation, however, there can't be found any solution algorithm about it in the control literature. In this paper we present an algorithm to calculate the common minimum dwell time, assuring each specified $\mathcal{H}_\infty$ level calculated separately for each single filter.

The contents of this paper are as follows. After the introduction, in Section II the dwell time condition for assuring stability of the switched linear $\mathcal{H}_\infty$ filter is presented. The main outcome is a special form of LMIs including the nonlinear term with the dwell time which represents a multivariable time dependent optimization problem. Section III presents the proposed numerical algorithm for the calculation of the common minimum dwell time assuring each specified $\mathcal{H}_\infty$ level. In Section IV the T_d-iteration algorithm is applied on an illustrative example in MATLAB. In Section V the main results are summarized and the paper is concluded with some final remarks.

2. Stability of the state estimation error dynamics involving the dwell time constraint

The synthesis technique proposed below is originated from results (Geromel and Colaneri, 2008) with focus on the application to robust nonlinear control, see in [7] and [6]. We have adopted this concept to a $\mathcal{H}_\infty$ detection filtering problem, which will be introduced in this chapter. However, in order to improve the detection's performance, we formulate our concept slightly different from theirs. That means, instead of calculation of the minimum dwell time assuring a common specified $\mathcal{H}_\infty$ level for each controller, we determine the common minimum dwell time to each specified $\mathcal{H}_\infty$ level calculated separately for each single filter. In the following we are referring to the concept in [12], which's system-description has been extended to a switched linear system.

Extending the switched linear system representation in [6] to the concept of perturbed system, see in [12], the extended switched linear system subjected to disturbance and faults, can be represented in state space form as follows:

$$\begin{aligned} \dot{x}(t) &= A_{\sigma(t)}x(t) + B_{\sigma(t)}u(t) + B_{\kappa\sigma(t)}\kappa(t) + \sum_{i=1}^{k} L_{i\,\sigma(t)}\nu_i(t), \quad x(0)=\xi, \\ y(\mathrm{t}) &= C_{\sigma(t)}x(t), \end{aligned} \tag{1}$$

where for all $t \geq 0$, $x(t) \in \mathbb{R}^n$ is the state vector, $\xi \in \mathbb{R}^n$ is the arbitrarily fixed initial condition, $u(t) \in \mathbb{R}^m$ is the input vector, $y(t) \in \mathbb{R}^p$ is the output vector, $\sigma(t): [0, \infty) \rightarrow \Theta$ is the piecewise constant switching function. $A_{\sigma(t)} \in \mathbb{R}^{nxn}$, $B_{\sigma(t)} \in \mathbb{R}^{nxm}$ and $C_{\sigma(t)} \in \mathbb{R}^{pxn}$ are an appropriate matrices. Assume, that the pairs $(A_{\sigma(t)}, C_{\sigma(t)})$ are observable for all $t \geq 0$. For further consideration denote n_e the number of subsystems, $\Theta = \{1, \dots, n_e\}$ an index set and $q = 1, \dots, n_e$ the sequence number of the switchings. $B_{\kappa\,\sigma(t)} = [B_{w,} L_\Delta]$ denotes the worst-case input direction and $\kappa(\mathrm{t}) \in \mathrm{L}_2\,[0, \mathrm{T}]$ is the input function for all $t \in \mathbb{R}_+$ representing the worst–case effects of modelling uncertainties and external disturbances. The cumulative effect of a number of k faults appearing in known directions $\mathrm{L_i}$ of the state space is modelled by an additive linear term $\sum L_{i\,\sigma(t)}\,\nu_i(t)$. $L_i \in \mathbb{R}^{nxs}$ and $\nu_\mathrm{i}(\mathrm{t})$ are the fault signatures and failure modes respectively. $\nu_i(t)$ are arbitrary unknown time functions for $t \geq t_{ji}$, $0 \leq t \leq T$, where t_{ji} is the time instant when the i-th fault appears and $\nu_i = 0$, if $t < t_{ji}$. If $\nu_i(t) = 0$, for every i, then the plant is assumed to be fault free. Assume, however, that only one fault appears in the system at a time.

Denote t_ℓ and $\mathrm{t}_{\ell+1}$ successive switching times satisfying $t_{\ell+1} - t_\ell \geq \tau_D$. Then the piecewise constant switching function between two consecutive switching instants as $\sigma(t): [0, \infty) \rightarrow \Theta$ for all $t(t_\ell, t_{\ell+1}]$ ensures, that the equilibrium point $x = 0$ of the system in (1) is globally asymptotically stable.

The referred constant $\tau_D > 0$ is called the dwell time. Consequently, when designing a switched system one also has to make sure, that the time difference between two consecutive switching instants not smaller than τ_D, then the asymptotical stability of the switched linear system is preserved, see e.g. in [1], [3], [4] and [7].

Generally interpreted, the fault detection filtering is done by estimating the states of the subjected system. Of course, we consider now a switched linear system approach, where the q-th sub-filter is selected whenever the q-th subsystem is active. The stability of the state estimation error dynamics may be a crucial part of such a design, which can be ensured when we switch slowly enough between the subsystems, to allow the transient effects to dissipate (Chen and Saif, 2004), (Prandini, 2015).

The state estimator for the system description (1) can be represented by the switched system as follows. Let $z \epsilon \mathbb{R}^p$ denote the output signal, then the state estimate can be obtained as

$$\begin{aligned} \dot{\hat{x}}(t) &= \left(A_{\sigma(t)} - Y_{\sigma(t)} C^T{}_{\sigma(t)} C_{\sigma(t)}\right) \hat{x}(t) + B_{\sigma(t)} u(t) + Y_{\sigma(t)} C^T{}_{\sigma(t)}\, y(t), \\ \hat{y}(t) &= C_{\sigma(t)} \hat{x}(t), \\ \hat{z}(t) &= C_{z\sigma(t)}\, \hat{x}(t), \end{aligned} \tag{2}$$

where $\hat{x} \in \mathbb{R}^n$ represents the observer state, $\hat{y} \in \mathbb{R}^p$ represents the output estimate, and $\hat{z} \in \mathbb{R}^p$ is the weighted output estimate, $Y_{\sigma(t)}$ is a positive definite matrix as a solution of the optimization problem in (5) and $C_{z\sigma(t)}$ is the estimation weighting.

The equation of the state estimation error for (2) is expressed as

$$\begin{aligned} \dot{\tilde{x}}(t) &= (A_{\sigma(t)} - Y_{\sigma(t)} C^T_{\sigma(t)} C_{\sigma(t)})\, \tilde{x}(t) + B_{\kappa\sigma(t)} \kappa(t) + \sum_{i=1}^{k} L_{i\sigma(t)}\, \nu_i(t), \\ \varepsilon(t) &= C_{z\sigma(t)} \tilde{x}(t), \end{aligned} \tag{3}$$

where $\tilde{x}$(t) and ε(t) are defined as

$$\begin{aligned} \tilde{x}(t) &= x(t) - \hat{x}(t), \\ \varepsilon(t) &= z(t) - \hat{z}(t), \end{aligned} \tag{4}$$

As the switching occurs within the finite set of $q \in \Theta = \{1, \ldots, n_e\}$ subsystems, the system description in (1) and consequently in (2) and (3) can be simply represented by the matrices $(A_q,\ B_q,\ B_{\kappa q},\ C_q,\ C_{zq},\ L_{iq},\ Y_q)$, $q \in \Theta$. Assume that all matrices A_q, $q \in \Theta$ are Hurwitz.

By duality we can associate the $\mathcal{H}_\infty$ control problem of the switched linear system described in [12] to our switched $\mathcal{H}_\infty$ filtering task, the synthesis of which is based on the Modified Riccati Equation (MFARE), that can be formulated for switched linear system as

$$A_q Y_q + Y_q A_q^T + Y_q \left(\frac{1}{\gamma_q^2} C_{z\,q}^T C_{zq} - C_q^T C_q \right) Y_q + B_{\kappa q} B_{\kappa\,q}^T = 0, \tag{5}$$

for all $q \in \Theta$. In (5) $\gamma_q > 0$ are positive rational constants and $Y_q \in R^{nxn}$ denote the decision variables which are positive definite matrices.

Following the steps of the synthesis procedure in [7], the MFARE can be factorized in form of Riccati Equation as

$$H_q Y_q + Y_q H_q^T + Q_q = 0, \ \forall q \in \Theta, \tag{6}$$

where the associated matrices are

$$W_q = Y_q \left[C_{z\,q}^T C_q^T \right] \begin{bmatrix} -\frac{1}{\gamma_q^2} I & 0 \\ 0 & I \end{bmatrix}, \tag{7}$$

$$H_q = \left(A_q - W_q \begin{bmatrix} C_{zq} \\ C_q \end{bmatrix} \right), \tag{8}$$

$$Q_q = \left(W_q \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} W_q^T + B_{\kappa q} B_{\kappa\,q}^T \right) - \gamma_q^2 W_q \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} W_q^T. \tag{9}$$

We have to note, that the optimal gain W_q is determined from the unique stabilizing solution to MAFARE and the matrix H_q is Hurwitz for each $q \in \Theta$. Since Q_q depends on the γ_{minq} value, $Q_q \geq 0$ is not guaranteed for any $q \in \Theta$. However, (6) admits a positive definite solution, since that was created by factorizing the MFARE. It is to note, that for solving the LMIs in (11) the condition $Q_q \geq 0$ is necessary, hence, if $Q_q \geq 0$ does not hold, $\gamma_q > \gamma_{minq}$ should be chosen such that $Q_q \geq 0$ holds.

For any $\sigma(t): [0, \infty) \to \Theta$ and for all $t \in (t_\ell, t_{\ell+1}]$, where $t_{\ell+1} = t_\ell + T_\ell$ with $T_\ell \geq T_d > 0$ and at $t = t_{\ell+1}$ the switching jumps to $\sigma(t) = j \in \Theta$, where the corresponding solution of the Lyapunov function along a trajectory of the switched filter state estimation error (2) is expressed by

$$\mathrm{V}\left(\tilde{\mathrm{x}}\left(t_\ell\right)\right) = \tilde{\mathrm{x}}\left(t_{\ell+1}\right)^{\mathrm{T}} Z_j \, \tilde{\mathrm{x}}\left(t_{\ell+1}\right) = \tilde{\mathrm{x}}\left(t_\ell\right)^{\mathrm{T}} e^{H_q{}^T T_\ell} Z_j e^{H_q T_\ell} \, \tilde{\mathrm{x}}\left(t_\ell\right), \tag{10}$$

where $Z_j \in R^{nxn}$ is a positive definite matrix.

The $\mathcal{H}_\infty$ control problem described in [7] can be associated to the $\mathcal{H}_\infty$ filtering problem by duality. Based on (6) and the Lyapunov function formulated along a trajectory of the state estimation error (10), one can derive time varying LMIs which can be used to obtain the common minimum dwell time constraint satisfying each $\mathcal{H}_\infty$ filter's specification.

Assume that for a given T_d there exists a collection of positive definite matrices $\{Z_1, \dots, Z_{ne}\}$ of compatible dimensions such that the LMIs

$$\begin{aligned} & H_q Z_q + Z_q H_q^T + Q_q < 0, \\ & \forall q \in \Theta, \\ & e^{H_q T_d} Z_j e^{H_q^T T_d} - Z_q + Y_q < 0, \\ & \forall q \neq j \in \Theta, \end{aligned} \tag{11}$$

hold under the worst-case input assumption in (1) for any switching signal $\sigma(t): [0, \infty) \to \Theta$ satisfying the condition $T_d = t_{\ell+1} - t_\ell \geq T_{dmin}$. Then, the equilibrium solution of the state estimation error (2) is globally asymptotically stable.

3. A numerical algorithm for determining the common minimum dwell time assuring each specified $\mathcal{H}_\infty$ level

As we have shown in the previous chapter, the problem of determining the minimum dwell time can be obtained by solving the set of LMIs (11). According to our idea, by means of combining an algorithm of interval halving for a fixed scalar T_d the LMIs can be treated as well as solved as a feasibility problem and the common minimum dwell time can be calculated. Before doing that, the MFARE in (5) was factorized and Y_q, H_q and Q_q matrices were obtained for each $q \in \Theta$. In [16] it is explained how the MFARE can be formulated and solved as a LMI. Then (11) can be represented via the following optimization problem:

$$\begin{cases} \min T_d \\ s.t.\ \ Z_q > 0 \\ \quad\ \ Z_j > 0 \\ \quad\ \ H_q Z_q + Z_q H_q^{\ T} + Q_q < 0 \\ \quad\ \ e^{H_q T_d} Z_j e^{H_q^{\ T} T_d} - Z_q + Y_q < 0. \\ \forall q \neq j \in \Theta \end{cases} \tag{12}$$

The main benefit of the LMI formulation is, that it defines a convex constraint with respect to the variable vector. For that reason, it has a convex feasible set which can be found guaranteed by means of convex optimization procedure. When using an LMI solver, however, it usually only accepts formulation where the decision variables are included in linear terms. Unfortunately the LMIs in (12), which include the term of matrix-exponential with the design scalar variable T_d, are nonlinear, consequently the task cannot be treated simply as a feasibility problem, see in [18], [19], [20] and [22]. To overcome these difficulties we implemented an algorithm called T_d-iteration, in which an interval halving method is used iteratively. The algorithm reduces gradually the value of the T_d scalar variable until the constraints of the LMIs in (12) are no longer feasible, consequently any of the Z_q matrices, has no longer positive definite solutions. The T_{dmin} which is so reached, is within the limits given by an arbitrarily small tolerance $\varepsilon > 0$ and is the minimum dwell time, such it holds that $T_{dmin} \leq \tau_D$.

The algorithm for the feasibility problem of determing the common minimum dwell time can be formulated as follows:

The inputs for the method are: Y_q, H_q and Q_q matrices for each $q \in \Theta$ which can be obtained from (5), and from (8), (9), respectively.

eps is the relative accuracy of the solution, T_{dmax} is the right limit of the interval (the left limit is zero).

The inner variables are: a, b and i. They stand for assignation of interval and counting cycle respectively. The T_{dm} variable contains the value of T_d at the end of the iteration.

The outputs are: Z_q matrices $q \in \Theta$ are positive definite decision variables, the T_d is the step size (midpoint). T_{dmin} contains the T_d value when the iteration is finished.

Each iteration performs the following steps:

1. Calculate T_d, the midpoint of the interval, which is assigned by a and b. That is $T_d = a + (b - a)/2$;

2. Calculate the matrix exponential function $e^{H_q T_d}$ for the fixed T_d value and substitute its values in (11);
3. Solve the LMIs in (11) as a feasibility problem by the MATLAB function *feasp* [22], which returns both the scalar value of t_{min} as a measure of the feasibility and the feasibility decision vector *xfeas;*
4. Call the MATLAB function *dec2mat* which returns the solutions for Z_q;
5. If the feasibility criteria with fixed T_d are not satisfied, that is $t_{min} \geq 0$, then the upper and lower bounds of interval are changed;
   Otherwise the value of T_d is saved, that is $T_{dm} = T_d$ and the iteration is continued;
6. Examine whether the new interval assigned by *b-a* reached the relative accuracy of the solution - called *epsilon*:
   - If not, the iteration is repeated;
   - If yes, the iteration is finished and the Z_q matrices are calculated based on the previous value of T_d. Additionally, $T_{dmin} = T_{dm}$.

In the following the MATLAB script for the T_d-iteration an illustrative example is presented for solving the multivariable time dependent optimization problem in (12). It was implemented for synthesis of a switched linear $\mathcal{H}_\infty$ filter, which consits of three subsystems for purpose of demonstation.

An approximate calculation of the minimum dwell time based on the L_2-norm of the state estimator system

Another, and very conservative approach is calculating the dwell time based on the L_2-norm of the H_q, see in [24]. This theorem says, that for each subsystem $q \in \{1, 2, \dots, n_e\}$ because H_q is Hurwitz, there exist $a_q \geq 0$ and $\lambda_q > 0$ such that for all $q \geq 0$, it can be written

$$\left\| e^{H_q t} \right\| \leq e^{a_q - \lambda_q t}, \tag{13}$$

$$\text{where} \quad \|H\| = \sqrt{\lambda_{max}(H^T H)}\,. \tag{14}$$

Based on this the dwell time is given as

$$\tau_D > \max_{q=1,2,\dots,\text{ne}} \left\{ \frac{a_q}{\lambda_q} \right\}. \tag{15}$$

Using the similarity transformation for matrix H, that is

$$H = TDT^{-1}\,. \tag{16}$$

The condition for the asymptotic stability expressed using T can be written as:

$$\|e^{Ht}\| \leq ||T||e^{\lambda t}||T^{-1}||. \tag{17}$$

Then using the T and D matrices, the parameter for calculating (14) can be obtained as

$$a = ln(\|T\|\|T^{-1}\|) \text{ and } \lambda = max_{1\leq i\leq n}\{\lambda_q\}\,. \tag{18}$$

4. An illustrative example

T_d – iteration algorithm

In the following the MATLAB script, for the T_d-iteration an illustrative example is presented for solving the multivariable time dependent optimization problem in (12). It was implemented for the synthesis of switched linear $\mathcal{H}_\infty$ filter and consists of three subsystems for purpose of demonstation.

Consider that the matrices Y_q, H_q and Q_q have been formerly calculated from (5), (8) and (9). Note, that these calculations are not presented in this paper.

```
% LMIFeaspDWT3m.m - 2018.06.25
% Calculating the minimum dwell time assuring the specified H-inf level
% Matrices derived from MFARE
% Subsystem 1
H1 = [-107.4991 15.4019 27.8936; 80.9958 -564.3523 1.9649; 0.1119 0.3748 -
8.6469];
Q1 = 1.0e+005 * [0.1810 -0.2182 -0.0002; -0.2182 2.7837 0.0006; -0.0002 0.0006
0.0000];
Y1 = [81.6978 -17.0191 -0.1127; -17.0191 244.1844 0.2370; -0.1127 0.2370
0.0103];
% Subsystem 2
H2 = [-91.9324 20.9809 28.0690; 106.1269 -611.2837 1.5668; 0.1088 0.4258 -
8.6914];
Q2 = 1.0e+005 * [0.1209 -0.2449 -0.0002; -0.2449 3.0927 0.0006; -0.0002 0.0006
0.0000];
Y2 = [61.5844 -18.0647 -0.1088; -18.0647 249.8336 0.2452; -0.1088 0.2452
0.0120];
% Subsystem 3
H3 = [-78.3760 25.7621 27.8720; 132.6617 -626.8514 1.9901; 0.1106 0.4382 -
8.5647];
Q3 = 1.0e+005 * [0.0854 -0.2658 -0.0001; -0.2658 3.1959 0.0008; -0.0001 0.0008
0.0000];
Y3 = [48.0245 -19.4905 -0.1127; -19.4905 250.7915 0.2733; -0.1127 0.2733
0.0142];

% Interval-halving
I=eye(3);
eps =1e-3;% the relative accuracy of the solution
Tdmax=3;       % the upper limit of the interval
Td=Tdmax; % the step size (midpoint)
b=Tdmax;       % the initial upper limit of the interval
a=0;      % the initial lower limit of the interval
i=0;      % initialization of the step counter
```

```
while (b-a)>eps      % examine whether the new interval reached the relative
accuracy
     Td = a+(b-a)/2;       % interval-halving
     i = i+1               % number of the iterations

% calulation of the matrix exponential functions each subsytems
     expHT1 = expm(Td*H1)
     expHT2 = expm(Td*H2)
     expHT3 = expm(Td*H3)

     setlmis([]);                  % define the system of LMI-s
% specifying the matrix variable Zq of the LMI ss.1
     Z1 = lmivar(1, [size(Y2, 1) 1]);
% specifying the matrix variable Zq of the LMI ss.2
     Z2 = lmivar(1, [size(Y2, 1) 1]);
% specifying the matrix variable Zq of the LMI ss.3
     Z3 = lmivar(1, [size(Y2, 1) 1]);
        % constructing the system of the LMI-s
     % for subsystem 1
     lmiterm([1, 1, 1, Z1], H1, 1, 's');  % LMI #1: Hq*Zq + Zq*Hq'
     lmiterm([1, 1, 1, 0], Q1);           % LMI #1: Qq
        lmiterm([2, 1, 1, Z2], expHT1, expHT1');  % LMI #2: expHTq*Zj*expHTq'
     lmiterm([2, 1, 1, Z3], expHT1, expHT1');  % LMI #2: expHTq*Zj*expHTq'
        lmiterm([2, 1, 1, Z1], -1, 1);        % LMI #2: -Zj
     lmiterm([2, 1, 1, 0], Y1);           % LMI #2: Yq
        % for subsystem 2
     lmiterm([3, 1, 1, Z2], H2, 1, 's');  % LMI #3: Hq*Zq + Zq*Hq'
     lmiterm([3, 1, 1, 0], Q2);           % LMI #3: Qq
     lmiterm([4, 1, 1, Z1], expHT2, expHT2');  % LMI #4: expHTq*Zj*expHTq'
     lmiterm([4, 1, 1, Z3], expHT2, expHT2');  % LMI #4: expHTq*Zj*expHTq'
     lmiterm([4, 1, 1, Z2], -1, 1);       % LMI #4: -Zj
     lmiterm([4, 1, 1, 0], Y2);           % LMI #4: Yq
        % for subsystem 2
     lmiterm([5, 1, 1, Z3], H3, 1, 's');  % LMI #5: Hq*Zq + Zq*Hq'
     lmiterm([5, 1, 1, 0], Q3);           % LMI #5: Qq
     lmiterm([6, 1, 1, Z1], expHT3, expHT3');  % LMI #6: expHTq*Zj*expHTq'
     lmiterm([6, 1, 1, Z2], expHT3, expHT3');  % LMI #6: expHTq*Zj*expHTq'
     lmiterm([6, 1, 1, Z3], -1, 1);       % LMI #6: -Zj
     lmiterm([6, 1, 1, 0], Y3);           % LMI #6: Yq

     % positivness of Zq
     lmiterm([-7, 1, 1, Z1], 1, 1);       % LMI #7: Z1>0
     lmiterm([-8, 1, 1, Z2], 1, 1);       % LMI #8: Z2>0
     lmiterm([-9, 1, 1, Z3], 1, 1);       % LMI #9: Z3>0

     lmis = getlmis;            % obtaining the system of LMI

        [tmin,xfeas] = feasp(lmis)  % calling function of feasibiliy.

% the solution Zq of ss. 1 corresponding to the
     Zs1 = dec2mat(lmis,xfeas,Z1)
% the solution Zq ss. 2 corresponding to the feasible
     Zs2 = dec2mat(lmis,xfeas,Z2)
% the solution Zq ss. 3 corresponding to the feasible
     Zs3 = dec2mat(lmis,xfeas,Z3)
```

```
% decision vector xfeas since tmin < 0
% checking constraints of feasibility. That is that if % tmin < 0.
        if tmin >= 0

          a = Td;              % the minimum is changed to the Td
          else
          b = Td;              % iteration is continued the minimum
                                        % is changed to the Td
          Tdm = b;             % saving value of Td
          end             % the iteration is continued

end                       % the iteration is finished

Tdmin = Tdm                    % the minimum dwell-time
```

The obtained values of the feasible solutions for Z_q denoted by Z_{s1}, Z_{s2}, Z_{s3} are shown below. The corresponding eigenvalues $\text{eig}(Z_q)$ denoted by $\text{eig}(Z_{s1})$, $\text{eig}(Z_{s2}), \text{eig}(Z_{s3})$ are shown in the row 10 of the Table 1. The positive eigenvalues prove the positive definiteness of Z_q and the feasibility as well. Note that in the rows 6, 8 and 11 we did not get a feasible solution, because the scalar t_{min} returned with a positive value, which means that the associated Z_q pencil contains eigenvalues on or very near to the imaginary axis. Of course, this resulted in infeasibility. In such cases, according to the algorithm of interval halving, in these steps the upper - and lower bounds of an interval changed, to ensure a proper distance between the eigenvalues and the imaginary axis.

$$Zs_1 = 10^4 * \begin{bmatrix} 0.4073 & 0.0542 & 0.5203 \\ 0.0542 & 0.0518 & 0.0806 \\ 0.5203 & 0.0806 & 1.8642 \end{bmatrix}, Zs_2 = 10^4 * \begin{bmatrix} 0.5122 & 0.0832 & 0.6322 \\ 0.0832 & 0.0573 & 0.1137 \\ 0.6322 & 0.1137 & 1.8605 \end{bmatrix},$$

$$Zs_3 = 10^4 * \begin{bmatrix} 0.6249 & 0.1250 & 0.7722 \\ 0.1250 & 0.0691 & 0.1679 \\ 0.7722 & 0.1679 & 1.8750 \end{bmatrix}.$$

Table 1: Values obtained for T_d , t_m and Z_q during the iteration

i	T_d	t_{min}	eig. (Zs_1)	eig.(Zs_2)	eig. (Zs_3)
1	1	-0.0854	25.9263 83.4787 246.1332	25.2094 65.1757 251.7411	24.3039 54.9130 252.8941
2	0.5000	-0.0849	25.9233 83.4782 246.1332	25.2067 65.1749 251.7411	24.3024 54.9119 252.8940
3	0.2500	-0.0529	25.7108 83.4469 246.1329	25.0205 65.1218 251.7407	24.1738 54.8301 252.8934
4	0.1250	-0.4185	63.3241 100.7714 246.4629	52.9620 95.4908 252.1020	44.0826 99.2161 253.4098
5	0.0625	-0.5410	68.0075 111.0561 246.6869	54.9368 107.9080 252.3338	45.0798 111.5681 253.6986
6	0.0313	0.0146	$10^4 *$ 0.0000 0.0219 3.5277	$10^4 *$ 0.0000 0.0115 3.0118	$10^4 *$ 0.0000 0.0073 2.1199
7	0.0469	-0.7438	65.5712 108.0716 246.8438	53.7206 104.2321 252.4862	44.1185 107.6819 253.8765
8	0.0391	0.0128	$10^4 *$ 0.0000 0.0177 3.8775	$10^4 *$ 0.0000 0.0098 3.0578	$10^4 *$ 0.0000 0.0114 3.2479
9	0.0430	-0.6534	63.1824 105.1323 246.8925	52.4740 100.2198 252.5290	43.2152 103.1864 253.9147
10	0.0410	-9.7372	$10^4 *$ 0.0437 0.2442 2.0353	$10^4 *$ 0.0426 0.2679 2.1196	$10^4 *$ 0.0423 0.2640 2.2627
11	0.0400	0.0126	$10^4 *$ 0.0000 0.0243 6.2884	$10^4 *$ 0.0000 0.0128 5.7631	$10^4 *$ 0.0000 0.0040 1.0035

The iteration ran till the new interval assigned by *b-a* reached the pre-specified relative accuracy of the solution $eps = 0.001$. Performed the T_d – iteration and repeated it 10-times the $T_{dmin} = 0.0410$s is obtained. The computational cost is primarily dependant on solving the q independent LMIs plus the iteration. We have seen, that despite of the multivariable time dependent optimization problem, combination of an algorithm interval halving with an LMI solver to determine the common minimum dwell time could be efficiently applied. A variation of the measures for the feasibility t_{min} during the iteration is shown in *Fig. 1*.

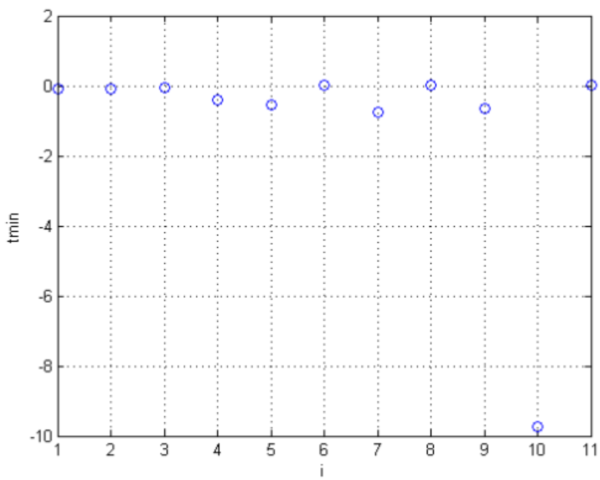

Figure 1: The variation of t_{min} during the iteration

An approximative calculation of the minimum dwell time based on the L_2-norm of the state estimator system

For purpose of comparison we applied the calculation of the minimum dwell time based on the L_2-norm of the state estimator system on the example. It is to note, that as it was introduced in the Chapter 3, this approach may lead to a conservative result.
The similarity transformation (16) has been computed by the MATLAB function $[T, D] = eig\ (H)$, see in [25], which returned the matrices T and D.
From the calculation in (15), the worst-case condition for the minimum dwell time is given by $\tau_{D*} > 0.055$ sec. The corresponding parameters in (18) were $a = 0.467$ and $\lambda = -8.474$.

5. Conclusion

This paper was concerned with a numerical algorithm for determining the minimum dwell time constraint for switched linear $\mathcal{H}_\infty$ fault detection filters.

Despite the multivariable time dependent optimization problem, by means of the T_d-iteration the common minimum dwell time assuring each specified $\mathcal{H}_\infty$ level of each single filter could be determined. The case study implemented in MATLAB resulted in positive definite solutions for Z_q and also in the corresponding minimum dwell time $T_{dmin} = 0.0410$ s. The results indicate that the frequency of changing the switching signal sequence should be lower than 24.39 Hz to ensure the robust stability of the state estimation error between the switching instants. Additionally, the dwell time was approximately calculated based on the L_2-norm of H_q. As it was shown, smaller value could be reached using the T_d-iteration, than from the approach based on the L_2-norm of H_q. Of course, the latter is only a very conservative approach. On the other hand the $\boldsymbol{T_d}$-iteration has to face with successive numerical computation of the quadratic matrix inequalities resulted in a proportional computation cost. We have found the solution after running the code in MATLAB after 0.5 second CPU time on a PC with Intel® Celeron® CPU B815 (1.60 GHz).

We think, that the technique of the T_d-iteration offers further benefits from the point of view the designer. Apart from the advantage that a variety of design specifications and constraints can be expressed through LMI-s, we assume, due to the combination with the interval halving algorithm, it gives more flexibility to examine the solution during the entire design process. For example, it is easy to analyse the impact of the T_d value on the number of iteration steps or to analyse the impact of the variation of the relative accuracy of the solution. One can easily perform experiments and get answers e.g. to the following questions: How does the iteration converge? How do the eigenvalues of the decision variable change? How close are they to the imaginary axis? Issues with such explicit conditions can be easily examined, step by step during the iterations, which can also be useful for better understanding the nature of switched systems.

References

[1] Paxman, J., “Switching Controllers: Realization, Initialization and Stability”, Ph.D Dissertation, University of Cambridge, 2003.

[2] DeCarlo, R. A., Branicky, M.S., Pettersson, S. and Lennartson, B., “Perspectives and Results on the Stability and Stabilizability of Hybrid Systems”, in *Proc. IEEE,* Vol. 88, pp. 1069–1082., 2000.

[3] Liberzon, D., Morse, A. S., “Basic Problems in Stability and Design of Switched Systems”, IEEE Contr. Syst., pp. 59–70., 1999.

[4] Hespanha, J. P., Morse, A.S., “Stability of switched systems with average dwell-time”, *Proc. 38th Conf. Decision and Control*, 1999, pp. 2655–2660.

[5] Alessandri, A., Coletta, P., “Switching observers for continuous-time and discrete-time linear systems”, in *Proceedings of the American Control Conference Arlington*, VA June 2001.

[6] Colaneri, P., "Analysis and control of linear switched systems", Lecture notes, Politecnico Di Milano, 2009.

[7] Geromel, J. C., Colaneri, P., "H∞ and Dwell Time Specifications of Switched Linear Systems", *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Dec. 9–11.2008.

[8] Chen, W., Saif, M., "Observer design for linear switched control systems", in *American Control Conference Proceedings of the 2004*, 0-7803–8335-4, Boston, 2004.

[9] Colaneri, P., Geromel, J.C., and Astolfi, A., "Stabilization of continuous-time switched nonlinear systems", Systems Control Letter, 57-1, pp. 95–103, 2008.

[10] Geromel, J. C., Colaneri, P., "Stability and stabilization of continuous-time switched linear systems", SIAM J. on Contr. Optim., 45-5, pp.1915–1930, 2006.

[11] Karabacak, O., and Sengör, N. S., "A dwell time approach to the stability of switched linear systems based on the distance between eigenvector sets", Int. J. System Science, 40(8), pp. 845–853, 2009.

[12] Edelmayer, A., "Fault detection in dynamic systems: From state estimation to direct input reconstruction", Universitas-Győr Nonprofit Kft., Győr, 2012.

[13] Edelmayer, A., Bokor, J., Keviczky, L., "An H∞ Filtering Approach to Robust Detection of Failures in Dynamical Systems", in *Proc. 33th Annual Decision and Control*, Conf., Buena Vista, USA, 1994, pp. 3037–3039.

[14] Edelmayer, A., Bokor, J., Keviczky, L., "An H∞ Filter Design for Linear Systems: Comparison of two Approaches", IFAC 13th Triennial World Congress, San Francisco, USA, 1996.

[15] Horváth, Zs., Edelmayer, A., "Robust Model-Based Detection of Faults in the Air Path of Diesel Engines", Acta Universitatis Sapientiae Electrical and Mechanical Engineering, Vol. 7 pp. 5–22, 2015.

[16] Horváth, Zs., Edelmayer, A., "Solving of the Modified Filter Algebraic Riccati Equation for H-infinity fault detection filtering", Acta Universitatis Sapientiae Electrical and Mechanical Engineering, Vol. 9, pp. 57–77, 2017.

[17] Bokor, J., Gáspár, P., Szabó, Z., "Robust Control Theory", Typotex, Budapest, 2013.

[18] Boyd, S., Ghaoui, L. E., Feron, E. and Balakrishnan, V., "Linear Matrix Inequalities in System and Control Theory", SIAM, Philadelphia, 1994.

[19] Nesterow, Y., Nemirovski, A., "Interior Point Polynomial Methods in Convex Programming: Theory and Applications", SIAM, Philadelphia, 1994.

[20] Gahinet, P, Apkarian, P., "A linear matrix inequality approach to H∞ control", International Journal of Robust and Nonlinear Control, Vol.4, pp.421–448, 1994.

[21] Duan, G., Yu, R., H., H., "LMIs in Control Systems: Analysis, Design and Applications", CRC Press, Boca Raton, 2013.

[22] https://www.mathworks.com/help/robust/ref/feasp.html

[23] Gahinet, P., Nemirovski, A., Laub, A. J. and Chilali, M. "LMI Control Toolbox for Use with Matlab", The MathWorks Inc., Natick, 1995.

[24] http://home.deib.polimi.it/prandini/file/2015_06_16%20hybrid%20systems_2.pdf

[25] https://www.mathworks.com/help/robust/ref/eig.html