Acta Universitatis Sapientiae
Electrical and Mechanical Engineering, 9 (2017) 57-77

DOI: 10.1515/auseme-2017-0010

# Solving of the Modified Filter Algebraic Riccati Equation for H-infinity fault detection filtering

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 29, 2017; revised December 18, 2017.

**Abstract:** The objective of this paper is solving of the Modified Filter Algebraic Riccati Equation (MFARE) for calculating of the filter gain. The results are used for model-based fault detection filtering of faults in the air path of diesel engines. The H-infinity optimization approach requires the solution of a linear-quadratic optimization problem that leads to the solution of MFARE. In our paper two basic concepts for solving MFARE are examined, namely the analytically implemented gamma-iteration and casting the problem as a convex optimization problem based on Linear Matrix Inequalities (LMIs).

The algorithms are implemented in MATLAB. Each algorithm has to ensure the condition for a global convergence and also has to deliver an optimal solution. Not at least, the computational cost has to be as small as possible.

**Keywords:** modified Filter Algebraic Riccati Equation, linear-quadratic optimization problem, H-infinity optimization, gamma-iteration, LMI

## 1. Introduction

With the increasing complexity of combustion engines in current automotive vehicles, the early detection of failures for engine diagnostics plays an increasingly important role. Possible faults are due to actuator, sensor and component failures, which can lead to engine malfunctions or even damages in the worst case. The subject of our investigation is a robust model-based fault detection filtering of faults in the air path of diesel engines. The filter robustness is ensured by the application of a design trade-off that is made between the worst-case disturbance and the $L_2$ norm of the filter error. This method requires the solution of a linear-quadratic optimization problem that leads to the solution of the Modified Filter Algebraic Riccati Equation (MFARE), see e.g. in [1], [2], [3] and [4].

Combustion engines can typically be characterized by highly nonlinear processes that may have very fast dynamics. This property poses additional requirements for the fault detection filter implementation. On the one hand, the filter should be capable of running recursively, in real-time, in few millisecond cycles, by taking the constrained computational capability of on-board microcontrollers into account. On the other hand, the computational complexity of the model might need processing power usually not available for the specific application. For this reason, finding an efficient algorithm to an optimal solution of the MFARE, which is definitely the core of the fault detection filter, is of great importance.

Several investigations have been carried out in the past two decades for using LMI to issues of robust control see e.g. [5], [6], [7]. So, it has been already proven, that LMI-s are effective and powerful tools for handling complex, but standard problems, such as fast computing of global optimum with some pre-specified accuracy. This has to be done by solving of the H-infinity optimization problem. While the analytically computed gamma-iteration represents the first step to solving MFARE, we have been first off, all interested in the efficiency and robustness of the solution based on LMI, which should, in our assumption, produce a better performance.

This paper is organized as follows: after the introduction, in Section II we shortly revisit the problem of H-infinity optimization and describe briefly the derivation of MFARE. In Section III MFARE is converted to an optimization problem based on LMI. In Section IV an algorithm called gamma-iteration is implemented to solve MFARE analytically. Then it is formulated as a linear objective minimization problem using LMI. Finally, each algorithm is evaluated to measure convergence, computation cost and at last but not at least, practicability.

## 2. Deriving the Modified Filter Algebraic Riccati Equation for robust H-infinity detection filtering

### *2.1 The optimal H-infinity detection filtering problem*

The goal of H-infinity filtering is minimizing the magnitude of the effects of perturbations on the filter output and maximizing the magnitude of the transfer function from failure modes to the filter error, through the appropriate choice of filter gain. This estimation problem can be represented as a mixed $H_2$ / $H_\infty$ filtering problem (Edelmayer, 2012) [8].

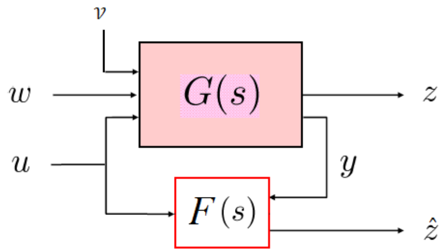


*Figure 1*: A standard setup for a robust $H_\infty$ filtering synthesis problem
(G: Generalized Plant, F: Filter)

According to the study in [7], the linear time-invariant system (LTI-system) subjected to disturbance and unknown faults can be represented in state space form as follows:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) + B_\omega \omega(t) + \sum_{i=1}^{k} L_i \nu_i(t), \\ y(t) &= Cx(t). \end{aligned} \tag{1}$$

In (1) $x\ \ \mathbb{R}^n$, $y\ \epsilon\,\mathbb{R}^p$, $u\ \epsilon\,\mathbb{R}^m$, and $\omega\ \epsilon\,\mathbb{R}^p$ denotes the process disturbance in $L_2\ [0,T]$. *A, B, C* and $B_\omega$ are appropriate constant matrices. It is assumed, that *(A, C)* is an observable pair. $B_\kappa = [B_{w,} L_\Delta]$ is the worst-case input direction and $\kappa(t) \in L_2\ [0,T]$ is the input function for all $t \in \mathbb{R}_+$ representing the worst–case effects of modelling uncertainties and external disturbances. It is to be noted, that the equation does not include parametric uncertainty [8]. The cumulative effect of a number of $k$ faults appearing in known directions $L_i$ of the state space is modelled by an additive linear term, $\sum L_i \nu_i(t)$ . $L_i\ \epsilon\,\mathbb{R}^{nxs}$ and $\nu_i(t)$ 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 [8].

For the purpose of explanation of the concept of the H-infinity filter, consider the system representation given in *Fig.1.*, where $z\ \epsilon\,\mathbb{R}^p$ denotes the output signal. Based on the LTI-system model (1), the state estimate can be obtained as

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + K(C(x(t) - \hat{x}(t))) + Bu(t), \\ \hat{y}(t) &= C\hat{x}(t), \\ \hat{z}(t) &= C_z\, \hat{x}(t). \end{aligned} \tag{2}$$

In (2), $\hat{x} \in \mathbb{R}^n$ represents the observer state, $\hat{y} \in \mathbb{R}^p$represents the output estimate, and $\hat{z} \in \mathbb{R}^p$represents the weighted output estimate, $K$ is the observer gain matrix and $C_z$ is the constant estimation weight (see in [8]).

The filter error system can be derived as

$$\begin{aligned} \dot{\tilde{\mathrm{x}}}(t) &= (A-KC)\,\tilde{\mathrm{x}}(t) + B_w w(t) + \sum_{i=1}^{k} L_i \nu_i(t), \\ \varepsilon(t) &= \mathrm{C}_z \tilde{\mathrm{x}}(t). \end{aligned} \tag{3}$$

In (3), $\tilde{x}$(t) and (t) are the state error and weighted output error, respectively, defined as

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

In the presence of faults, the estimation error does not converge asymptotically to zero, but converges asymptotically to a subspace which is different from zero [8].

In the following we have to choose the filter gain, by minimizing the magnitude of the effects of perturbations on the output of the filter, which has to maximize the magnitude of the transfer function from failure modes to the filter error.

### *2.2 Solution to a H-infinity filtering*

Based on the representation in *Fig.1,* the performance measure considered as a quadratic cost function of the minimax method is defined as

$$J(w,\nu,\widehat{z}) = \frac{1}{2}\left[\left\|z-\widehat{z}\right\|_2^2 - \gamma^2\left(\left\|w\right\|_2^2 + \left\|\nu\right\|_2^2\right)\right], \tag{5}$$

where $\gamma > 0$ is a positive rational constant.

According to the H-infinity filtering problem the quadratic cost function to be minimized is defined as

$$\sup_{w,\nu,a_i} J(w,\nu,\widehat{z}). \tag{6}$$

The performance can be formulated as a min-max problem. That is, minimizing the H-infinity norm of the transfer function, denoted by $H_{\varepsilon\kappa}$, of the worst-case disturbance to the filter output. The worst-case performance is given by

$$J(\mathrm{K},\kappa) = \sup \frac{\left\|z-\widehat{z}\ \right\|_2}{\left\|\kappa\ \right\|_2} = \left\|H_{\varepsilon\kappa}(s)\right\|_\infty . \tag{7}$$

The filter gain $K$ can be obtained by solving a linear-quadratic optimization problem, using the procedure presented below (see also in [8]).

With substitution of the decision variable $Q \in R^{nxn}$ which is a positive definite matrix, the observer equation can be described as

$$\dot{\hat{x}}(t) = (A - QC^T C)\,\hat{x}(t) + Bu(t) + QC^T y(t),$$
$$\hat{z}(t) = C_z\,\hat{x}(t). \tag{8}$$

The goal of the linear-quadratic optimization is to obtain the smallest $L_2$ -gain of the disturbance input of the system that is guaranteed to be less than a specified positive constant $\gamma_{min}$, and in the same time to increase filtering sensitivity as much as possible (Edelmayer, 2012). The algorithm, which is used to find an optimal solution for $Q$, iteratively reduces $\gamma$ until $Q$ has no longer a positive definite solution. Note that the $\gamma_{min}$ obtained this way is within a given arbitrarily small tolerance $\varepsilon > 0$.

The procedure is based on the solution of the Modified Filter Algebraic Riccati Equation (MFARE). From the bounded-real lemma, we have $\|H_{\varepsilon\kappa}\|_\infty < \gamma$ if and only if there exists $Q \geq 0$ such that

$$AQ + QA^T - Q(C^T C - \frac{1}{\gamma^2} C_z^T C_z)Q + B_K B_K^T = 0. \tag{9}$$

After solving equation (9) and getting a solution for $Q$, the filter gain matrix can be obtained as

$$K = Q\,C^T. \tag{10}$$

With the use of $\gamma_{min}$ the detection threshold of the filter can be given as

$$\tau(C_z) = \gamma_{\min} \|\kappa\|_2 . \tag{11}$$

It is important to note, that the failure modes, which have the magnitude smaller than that of the detection threshold, cannot be detected by the filter.

## 3. Solving MFARE by LMI

Originally the problem was introduced in about 1890 by the Russian mathematician Aleksandr Mikhailovich Lyapunov. Linear Matrix Inequalities (LMIs) have become nowadays effective and powerful tools for solving complex optimization problems. The applicability of LMI is really wide, starting e.g. from classical Lyapunov stability analysis of linear time variant and invariant systems, going through traditional Linear Quadratic Gaussian (LQG) control, up to the synthesis of modern robust H-infinity state feedback. The reason for it is that many problems can be cast as convex optimization problems. What is more, most of them can be converted to a standard LMI problem such as computing of global optimum with some pre-specified accuracy, even if it is to be done in our case by solving of H-infinity optimization problem. 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 convex optimization.

A detailed survey about the theory of LMI can also be found in the mathematical literature, see e.g. in [9], [10] and also in textbooks for control engineering e.g. in [11], [12], [13] and [14].

### *3.1 Standard problems involving LMIs*

A linear matrix inequality is a matrix inequality of the form

$$F(x) \overset{\Delta}{=} F_0 + \sum_{i=1}^{m} x_i F_i > 0, \tag{12}$$

where $x \in \boldsymbol{R}^m$ is the vector of decision variables, and
$F_i = F_i^T \in \boldsymbol{R}^{n\,x\,n},\ i = 0, \cdots, m$ are symmetric matrices.

Let $A(x)$, $B(x)$ and $C(x)$ be symmetric matrices that depend affinely on $x \epsilon \mathbb{R}^m$. Then, in addition to the canonical from in (12) standard LMI problems can be formulated in three different ways (see e.g. in [13]):

1. Feasibility problem with the task of finding a solution for decision variable $x$ so that the constraint
$$A(x) < 0 \tag{13}$$
is sufficient.

2. Linear objective minimization i.e. searching for $x$ which minimizes the linear function subject to an LMI.

   That is, minimize $c^T x$ subject to $A(x) < 0$ . (14)

3. Generalized eigenvalue minimization problem i.e. minimizing the maximum generalized eigenvalue of a pair of matrices, that depend affinely on a variable, subject to an LMI constraint.
   The task is to minimize $\lambda$ subject to an LMI constraint:

$$\begin{gathered} A(x) < \lambda B(x) \\ B(x) > 0 \\ C(x) < 0\,. \end{gathered} \tag{15}$$

Unfortunately, most of the control synthesis problems are not formulated as an LMI, but the nonlinear (convex) inequalities can be converted to an LMI form using the Schur complements' lemma (Boyd et. al. in 1994) [13].

According to this lemma the expressions (16) and (17) are equivalent.

$$\begin{bmatrix} Q(x) & S(x)^T \\ S(x) & R(x) \end{bmatrix} < 0, \tag{16}$$

$$R(x) < 0, \quad Q(x) - S(x)^T R(x)^{-1} S(x) < 0. \tag{17}$$

$Q(x) = Q(x)^T,\ R(x) < 0$, and $S(x)$ depend affinely on $x$.

In this manner the set of nonlinear inequalities in (17) can be represented as the LMI in (16).

Back to our problem of quadratic optimization we have to solve the MFARE as

$$AQ + QA^T - Q(C^T C - \frac{1}{\gamma^2} C_z^T C_z)Q + B_K B_K^T = 0. \tag{18}$$

To transform (18) into an LMI, at first, we rewrite it in form of inequalities. For this let $R = Q^{-1}$, so we get

$$A^T R + RA - C^T C + \frac{1}{\gamma^2} C_z^T C_z + RB_K B_K^T R < 0\ ,\ \ R > 0. \tag{19}$$

Applying the Schur complement lemma (17) for (19) yields to

$$\underbrace{A^T R + RA - C^T C}_{Q(x)} - \underbrace{\begin{bmatrix} C_z^T & RB_K \end{bmatrix}}_{S^T(x)} \underbrace{\begin{bmatrix} -\gamma^2 I & 0 \\ 0 & -I \end{bmatrix}^{-1}}_{R^{-1}(x)} \underbrace{\begin{bmatrix} C_z \\ B_K^T R \end{bmatrix}}_{S(x)} < 0. \tag{20}$$

Finally, by using the Schur complement lemma in (16) we obtain the LMI for the MFARE as

$$\begin{bmatrix} RA + A^T R - C^T C & C_z^T & RB_\kappa \\ C_z & -\gamma^2 I & 0 \\ B_\kappa^T R & 0 & -I \end{bmatrix} < 0, \tag{21}$$

which has a solution $R = R^T \in \mathbb{R}^{n\,x\,m}$ and $\gamma > 0$.

Consequently, we can solve the MFARE by minimizing $\gamma$ with respect to $R \succ 0$ subject to (21).

The corresponding Hamiltonian matrix

$$H_\gamma = \begin{bmatrix} A^T & -(C^T C - \frac{1}{\gamma^2} C_z^T C_z) \\ -B_\kappa B_\kappa^T & -A \end{bmatrix}, \tag{22}$$

has no eigenvalue on the imaginary axis.

In most cases it is possible to solve the Algebraic Riccati Equation also through similarity transformation of the Hamiltonian matrix see e.g. in [15]. Although this method is not for solving MFARE as an optimization problem, so it won't lead to an expected result, it may be useful to check a result obtained via optimization.

The method is described as follows, see in [15]. First the $(2n, n)$ matrix $V$ is built which contains the eigenvectors corresponding to the eigenvalues with negative real parts (stable invariant subspace) of the Hamiltonian matrix:

$$V = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}. \tag{23}$$

We can get the solution for matrix $Q_H$ as

$$Q_H = V_2 V_1^{-1}. \tag{24}$$

## 4. Calculation of the filter gain based on the LTI -model of the air path of the diesel engines

For the investigation of fault detection filtering problem, we are interested in the efficiency and robustness of the optimal solution. Thus, two different methods for solving MFARE are compared. First, an algorithm called gamma-iteration is implemented to solve MFARE analytically, then it is formulated as a linear objective minimization problem, solved via LMI.

### *4.1 LTI-model for the air path of diesel engines*

As mentioned in the introduction of the robust fault detection filter design methodologies that we apply in our investigation, it is required to use the LTI-model. Here we refer to a simplified nonlinear model of the air path which was first suggested by Jankovic and Kolmanovsky in 1998 [16] and later by Jung [17] for the purpose of robust control of the diesel engines. In our earlier investigation [18] we have already linearized this model around a specified operating point (Herceg, 2006) [19]. For the sake of simplification, we have considered the fuelling of diesel oil as a constant input, and not as a disturbance, furthermore the disturbance was modelled as the fluctuating change of the engine speed.

As a result, we derive the following LTI-model in the chosen operating point [18]

$$
\begin{gathered}
A=\begin{bmatrix} -5.2643 & 4.7316 & 28.5021 \\ 50.7697 & -156.9827 & 0 \\ 0 & 0.4287 & -9.0909 \end{bmatrix}, \\
B=\begin{bmatrix} 1.6111\cdot 10^{9} & 0 & 0 \\ -1.5720\cdot 10^{10} & 8.3514\cdot 10^{4} & 1.46083\cdot 10^{8} \\ 0 & -141.6484 & 0 \end{bmatrix}, \\
B_{\omega}=\begin{bmatrix} -47.7946 \\ 466.3408 \\ 0 \end{bmatrix}, \quad C=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3.924\cdot 10^{-5} \end{bmatrix},
\end{gathered}
\tag{25}
$$

where *A, B, C* and $B_\omega$ are appropriate constant matrices, $B_\omega$ is the matrix for the disturbance acting on the system.

### *4.2 Solution of the MFARE by a gamma-iteration algorithm*

This section discusses a conventional numerical method called gamma-iteration to get an optimal solution of MFARE. It has to be noted, that this method is often referred to, see e.g. in [1], [2] and [20], [21], but we have not found any algorithm about it. This has been the motivation for its description.

For the start of the explanation, the estimation weight of the filter is chosen arbitrarily, according to the methodology described in [8]:

$$C_Z = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 25 \end{bmatrix}. \tag{26}$$

The MFARE is written again as

$$AQ + QA^T - Q(C^T C - \frac{1}{\gamma^2} C_z^T C_z)Q + B_K B_K^T = 0. \tag{27}$$

Arranged for the use of the MATLAB function ***care*** [22], the equation becomes:

$$AQ + QA^T - Q\begin{bmatrix} C_z^T & C^T \end{bmatrix} \underbrace{\begin{bmatrix} -\gamma^2 I & 0 \\ 0 & I \end{bmatrix}}_{R_{care}}^{-1} \begin{bmatrix} C_z \\ C \end{bmatrix} Q + B_\kappa B_\kappa^T = 0. \tag{28}$$

It is important to note, that the function ***care*** is typically used for solving the H-infinity Riccati Equation for control problems. However, according to the principle of duality between controllers and observers the ***care*** function can be parameterized to be used for a filter in the form:

***[Q L Gr report] = care (A', CC, $B_\kappa$* $B_\kappa$', $R_{care}$, 'report')***,

where $CC = [C_Z^T \quad C^T]$.

The function ***care*** returns the optimal value for the decision variable, denoted by $Q$.

Of course the $R_{care}$ - matrix contains $\gamma$, but this has a constant value for a specified level of the disturbance attenuation. It results that the function ***care*** cannot be directly used for a quadratic minimization problem, that is, the value of $\gamma$ is to be iteratively reduced and the decision variable minimized. In this manner, in order to get the $\gamma_{min}$ value, and so the corresponding optimal solution for $Q$, we implemented an algorithm called gamma-iteration in which an interval halving method is used iteratively. The algorithm reduces the value of $\gamma$ until $Q$ has no longer positive definite solution. The $\gamma_{min}$, which is reached, is within the limits given by an arbitrarily small tolerance $\varepsilon > 0$.

The gamma-iteration algorithm can be formulated as follows.

The inputs for the method are the $A$, $B_d$, $C$, $C_z$ matrices, which define the LTI-system, *eps* as the relative accuracy of the solution, *maxgamma* as the right limit of the interval (the left limit is zero).

*a*, *b* and *i* are secondary variables, they stand for assignation of interval and counting cycle respectively.

The outputs are: matrix *Q* as a positive definite decision variable, the *gamma* as step size (midpoint), the *mingamma* variable, which contains the value of gamma at the end of an iteration, the *minigamma* contains the *gamma* value when the iteration is finished.

Each iteration performs the following steps:

1. Calculate *gamma*, the midpoint of the interval, which is assigned by *a* and *b*, that is *gamma = a+(b-a)/2*;
2. Call the MATLAB function *care* which returns the matrix *Q* and the "*report*";
3. Calculate the eigenvalues of *Q* , called *Lambda*;
4. If the convergence criteria of the iteration are not satisfied, namely: *Q* is NOT positive definite, i.e. *prod(Lambda)<=0 or t*he associated Hamiltonian matrix (22) that contains $\gamma$ has eigenvalues on or very near the imaginary axis, then the upper and lower bounds of interval are changed;
   Otherwise the value of *gamma* is saved, that is *mingamma = gamma* and the iteration is continued;
5. Examine whether the new interval defined 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 filter gain is calculated based on the previous value of *gamma* (*mingamma*).

The algorithm is implemented in MATLAB and the script is given below (the example is based on the LTI-system defined by (25)).

```
% matrices of the proposed LTI-system
A=[ -5.2643, 4.7316, 28.5021; 50.7697, -156.9827, 0; 0 , 0.4287, -9.0909 ];
B=[1.6111e+009, 0, 0; -1.5720e+010, 8.3514e+004, 146083000; 0, -141.6784, 0];
C=[1, 0 ,0 ; 0, 1, 0 ; 0, 0, 3.924e-005]; Cz=[5, 0, 0; 0 ,5, 0; 0, 0, 25];
Bd=[-47.7946 0 0  ; 466.3408 0 0  ; 0 0 0 ];
```

```
eps =1e-2;                                  % the relative accuracy of the solution
CC =[Cz', C'];                              % building the output matrix
m1 = size(Cz',2);                           % building submatrices for the Rcare
m2 = size(C',2);                                diagonally matrix
maxgamma =1100;                             % the upper limit of the interval
gamma = maxgamma;                           % the step size (midpoint)
b= maxgamma;                                % 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
          gamma = a+(b-a)/2;                % interval-halving
          i = i+1;                          % step counting

         % calculation of the Rcare diagonal matrix containing the gamma value
         Rcare = [-(gamma )^2*eye(m1) zeros(m1, m2) ; zeros(m2, m1) eye(m2)];

         % solving of the MFARE using the function care
          [Q L Gr report] = care(A', CC, Bd*Bd', Rcare, 'report')
         Lambda = eig(Q);                   % calculation of the eigenvalues of Q

         % reports:
         % if it is < 0, then the associated Hamiltonian matrix has its
         eigenvalues on or very near the imaginary axis, which results in failure
         % if prod(Lambda)<=0, then Q is not positive definite
         if (report==-1 || report==-2 || prod(Lambda)<=0)
                  a = gamma;                % the lower bound is changed to gamma
         else
                  b = gamma;                % the upper bound is changed to gamma
                  mingamma = gamma;         % saving gamma value
         end                                % the iteration is continued
end                                         % the iteration is finished
gammamin = mingamma                         % the obtained γmin
K=Q*C'                                      % the obtained filter gain
```

Repeating the $\gamma$-iteration 21 times, the optimal value of $\gamma_{min} = 4.9698$ is obtained. Using (10), the corresponding filter gain results as:

$$K = \begin{bmatrix} 257.2236 & -39.2216 & -0.0000 \\ -39.2216 & 699.2298 & 0.0000 \\ -0.7934 & 1.6744 & 0.0000 \end{bmatrix}. \tag{29}$$

It has to be noted, that in steps 8,11,13 and 20 we did not get solution because ***care*** returned with a report = -1. This means that the associated Hamiltonian matrix (22) had its eigenvalues on or very close to the imaginary axis which results in failure, see in [22]. According to the interval halving algorithm, in these steps the upper and lower bounds of the interval are changed in order to keep the solution away from the imaginary axis.

In order to prove the filter performance for disturbance attenuation, the transfer function of the disturbance to a filter residual for the obtained filter gain *K* is

$$H_{\varepsilon\omega}(s) = C_Z(sI - A + KC)^{-1} B_{\omega}. \tag{30}$$

The evolution of the disturbance attenuation during the iteration steps can be observed on the value of $\|H_{\varepsilon\omega}(s)\|_{\infty}$ , calculated in MATLAB and plotted in *Fig. 2*.

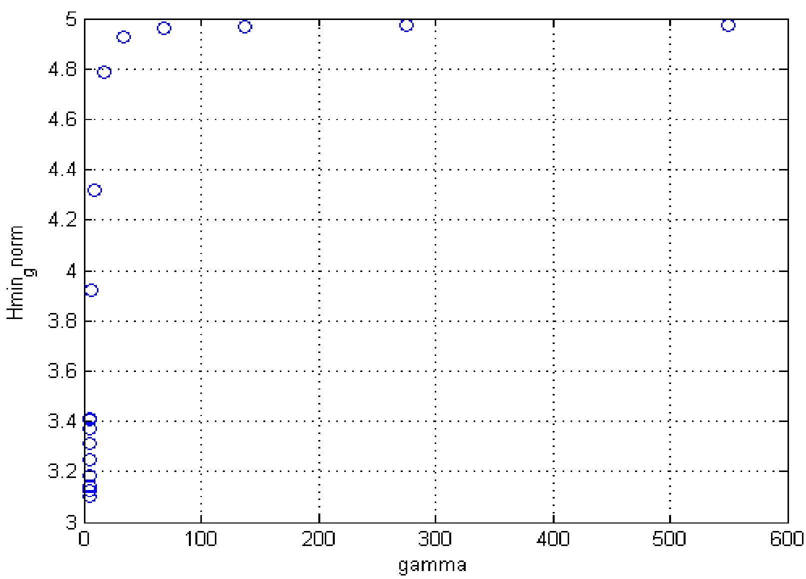


*Figure 2*: The variation of the $\|H_{\varepsilon\omega}(s)\|_{\infty}$ value as a function of gamma values during the iteration

The optimal value obtained at the end of the iteration is for $\|H_{\varepsilon\omega}(s)\|_{\infty} = 3.3737$.

### *4.3 The impact of increasing the value of* $\gamma_{min}$

As known, $\gamma$ is a measure for the filter sensitivity [8]. In the following it is examined the impact of increasing the value of $\gamma_{\min}$ .

In case if $\gamma_{min}$ reached its upper limit (here $\gamma_{min} = \infty$) the term depending on $\gamma$ dropped out and (9) was reduced to the form

$$AQ + QA^T - QC^T CQ + B_K B_K^T = 0. \tag{31}$$

*Table 1*: The impact of increasing the value of $\gamma_{min}$

| gamma | matrix K | Eigenvalues of Q | $\lVert H_{\varepsilon\kappa}(s) \rVert_\infty$ |
|---|---|---|---|
| $\gamma_{min}$ | 257.2236 -39.2216 -0.0000<br>-39.2216 699.2298 0.0000<br>-0.7934 1.6744 0.0000 | 0.0875<br>253.7718<br>702.6875 | 3.4047 |
| 10 $\gamma_{min}$ | 13.5339 -39.8412 0.0000<br>-39.8412 330.4657 0.0000<br>-0.2148 0.2480 0.0000 | 0.0017<br>8.6061<br>335.3976 | 4.9492 |
| 100 $\gamma_{min}$ | 13.4326 -39.6856 0.0000<br>-39.6856 329.3445 0.0000<br>-0.2129 0.2461 0.0000 | 0.0017<br>8.5275<br>334.2637 | 4.9717 |
| determinist. Kalman Filter | 13.4328 -39.6857 0.0000<br>-39.6856 329.3445 0.0000<br>-0.2129 0.2461 0.0000 | 0.0017<br>8.5275<br>334.2637 | 4.9719 |

The magnitude of the transfer functions of the disturbance to a filter residual for increased $\gamma_{min}$ values is shown in *Fig. 3*.

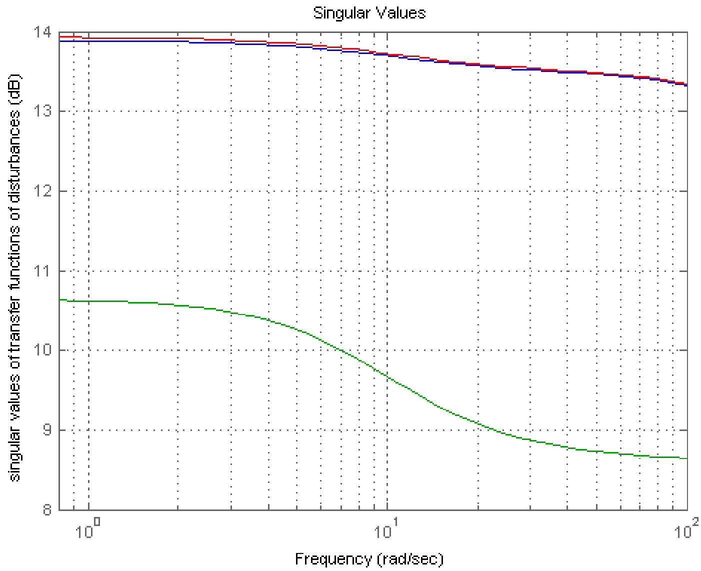


*Figure 3*: The magnitude (maximal singular values) of transfer functions:
$H_{\varepsilon\varpi}$ $(\gamma_{min})$: green line , $H_{\varepsilon\varpi}$ $(10\ \gamma_{min})$: blue line , $H_{\varepsilon\varpi}$ $(100\ \gamma_{min})$: red line

As it can be seen in Table 1, the smallest value for $\|H_{\varepsilon\varpi}\ (s)\|_\infty$ is 3.4047 (10.6415 dB) so the best filter sensitivity against worst case disturbance can be achieved in case of $\gamma_{min}$ as it is shown in *Fig.3.*

In case of $100\,\gamma_{min}$ and the deterministic Kalman-filter there exists no significant difference between the magnitudes of the transfer functions, as it can be seen in *Fig.3* (blue and red lines). For $100\,\gamma_{min}$ we get a $\|H_{\varepsilon\varpi}\ (s)\|_\infty$ = 4.9717 (13.93 dB), which results in lower disturbance attenuation.

It can be concluded, that the more the value of $\gamma_{min}$ is increased, the less filter sensitivity can be achieved. In this sense, getting an optimal $\gamma_{min}$ value is of great importance.

It is conceivable, that the H-infinity filter becomes a deterministic Kalman-filter by reaching its upper limit at $\gamma_{min} = \infty$. This can be also proven easily based on (31).

Of course, the H-infinity filter ensures that the energy gain from the disturbances to the estimation error is always less than a pre-specified level $\gamma^2$. Thus it is less conservative than the deterministic Kalman-filter. This is its main advantage from designer's point of view.

*4.3 Verification of the solution obtained for the MFARE via the Hamiltonian-matrix*

It is possible to verify the solution for the decision variable also via calculating the eigenvectors of the Hamiltonian-matrix of MFARE as it was explained in Subsection 3.1.

The resulting Hamiltonian-matrix for MFARE in case of $\gamma_{\min} = 4.9698$ is

$$H_\gamma = 10^5 \begin{bmatrix} -0.0001 & 0.0005 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & -0.0016 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0003 & 0.0000 & -0.0001 & 0.0000 & 0.0000 & 0.0003 \\ -0.0228 & 0.2229 & 0.0000 & 0.0001 & -0.0000 & -0.0003 \\ 0.2229 & -2.1747 & 0.0000 & -0.0005 & 0.0016 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & -0.0000 & 0.0001 \end{bmatrix}.$$

At first we calculate the eigenvalues and the corresponding eigenvectors of the Hamiltonian-matrix via a similarity transformation. The resulting matrix, containing the eigenvalues is

$$diag\,\lambda_i(H_\gamma) = diag\left[150.0393,\ \text{-}150.0393,\ 6.8273,\ 0.4622,\ \text{-}0.4622,\ \text{-}6.8273\right].$$

Secondly, we have to build a $(2n, n)$ matrix $V$, which contains the eigenvectors of the Hamiltonian matrix corresponding to the eigenvalues with negative real parts (23).

The submatrices of $V$, which contain the eigenvectors, are:

$$V_1 = \begin{bmatrix} 0.0005 & 0.0039 & 0.0029 \\ -0.0014 & 0.0001 & -0.0003 \\ 0.0004 & 0.0062 & -0.1194 \end{bmatrix},\ V_2 = \begin{bmatrix} 0.1749 & 0.9986 & 0.8475 \\ -0.9846 & -0.0516 & -0.5171 \\ -0.0027 & -0.0023 & -0.0139 \end{bmatrix}.$$

Let $Q_H$ denote a solution calculated using the Hamiltonian-matrix, which has a solution

$$Q_H = V_2 V_1^{-1} = \begin{bmatrix} 258.0838 & -32.9691 & -0.7468 \\ -33.7848 & 691.7261 & 1.7722 \\ -0.7809 & 1.6763 & 0.0932 \end{bmatrix}.$$

From the gamma-iteration in Subsection 4.2 we got an optimal solution as

$$Q = \begin{bmatrix} 257.2236 & -39.2216 & -0.7934 \\ -39.2216 & 699.2298 & 1.6744 \\ -0.7934 & 1.6744 & 0.0934 \end{bmatrix}.$$

It can be stated , that matrices $Q_H$ and $Q$ are slightly different. This leads to the conclusion of plausibility of an optimal solution $Q$ obtained by the gamma-iteration.

### *4.4 Solution for the MFARE by LMI*

In Section 3 we introduced the method for finding the optimal solution for MFARE implemented analytically as an interval halving algorithm. However, the task of minimization results in the task of computing a system of matrix equations which is not always convex [8].

Thus, let us now consider the problem of finding the optimal solution for the filter gain by solving of MFARE formulated as a LMI.

To handle it, several commercial software tools can be chosen. In this study the LMI Control Toolbox of MATLAB has been used, which provides a set of convenient functions to solve problems involving LMIs [23].

Generally, the solution of LMIs is carried out in two stages in MATLAB. At first, the decision variables of the LMI are defined, then it is defined the system of LMIs based on these decision variables. These are mostly represented in matrix form. In the second stage, the optimization problem is solved numerically using the chosen solvers as it is explained in Section 2.

In our case study the LMI in (21) is formulated as a linear objective minimization problem. That is, the task is to minimize a linear function of $x$ subject to an LMI constraint:

$$\min_{x} \left\{ c^T x : F(x) \succ 0 \right\}. \tag{32}$$

The LMI for the MFARE derived in Section 2 is described in (21).

In the following it is presented the MATLAB script for the linear objective minimization problem of the MFARE

```
% matrices of the proposed LTI-system
A=[ -5.2643, 4.7316, 28.5021; 50.7697, -156.9827, 0; 0 , 0.4287, -9.0909 ];
B=[1.6111e+009, 0, 0; -1.5720e+010, 8.3514e+004, 146083000; 0, -141.6784, 0];
C=[1, 0 ,0 ; 0, 1, 0 ; 0, 0, 3.924e-005]; Cz=[5, 0, 0; 0 ,5, 0; 0, 0, 25];
Bd=[-47.7946 0 0  ; 466.3408 0 0  ; 0 0 0 ];
I=eye(3);
    % specifying the matrix variables of the LMI
    setlmis([]);
    R = lmivar(1, [size(A, 1) 1]);
    % constructing the system of the LMI
    gamma2 = lmivar(1, [1, 1]);
```

```
lmiterm([1, 1, 1, R], 1, A, 's');         % R'A+AR
lmiterm([1, 1, 1, 0], -C'*C);             % -C'C
lmiterm([1, 2, 1, 0], Cz);                % Cz
lmiterm([1, 2, 2, gamma2], -1, I);        % -gamma^2I
lmiterm([1, 2, 3, 0], 0);                 % 0
lmiterm([1, 3, 1,R], Bd', 1);             % Bd'R
lmiterm([1, 3, 2, 0], 0);                 % 0
lmiterm([1, 3, 3, 0], -1);                % -I
lmiterm([-2, 1, 1,R], 1, 1);
lmiterm([-3, 1, 1, gamma2], 1, 1);
% obtaining the system of the LMI
lmimingfilt5 = getlmis;
c = mat2dec(lmimingfilt5, zeros(size(A, 1), size(A, 1)), 1);
% the relative accuracy of the solution
 options = [1e-3 , 0, 0, 0, 0];
% solving LMI
[alpha, popt] = mincx(lmimingfilt5, c,  options);
% the optimal value for the decision variable "R"'
Ropt = dec2mat(lmimingfilt5, popt, R);

% the optimal value of the gamma
gopt = dec2mat(lmimingfilt5, popt, gamma2);
% the obtained γmin
gammaopt=sqrt(gopt)
% the optimal solution of the LMI
Qopt = inv(Ropt);
% the calculated filter gain
K = Qopt*C'
```

### *4.5 Comparison of the performance of the LMI with the performance of the gamma-iteration*

The efficiency and robustness of the optimal solution are interesting aspects of the fault detection filtering problem. Thus, two different methods for solving MFARE are compared, namely the LMI formulated as a linear objective minimization problem and the numerically implemented gamma-iteration.

The results of the MATLAB simulations are shown in Table 2.

*Table 2*: Comparison of the different solutions for the MFARE

| Performance | LMI as an linear objective minimization problem | gamma-iteration |
|---|---|---|
| $\gamma_{min}$ | 4.9704 | 4.9698 |
| K | 278.80 -52.70 0.0000<br>-52.70 1308.4 0.0000<br>-0.500 0.600 0.0000 | 257.2236 -39.2216 0.0000<br>-39.2216 699.2298 0.0000<br>-0.7934 1.6744 0.0000 |
| Eigenvalues of Q | 0.3<br>276.1<br>1311 | 0.0017<br>8.5275<br>334.2637 |
| $\|H_{\varepsilon\omega}(s)\|_\infty$ | 4.4345 | 3.4047 |
| number of iterations | 9 | 21 |
| computation cost (sec) | 0.1 | 1 |

From the simulation and results of the comparison of the two different methods it can be concluded that each one gives an optimal solution. To be more precise, the minimization algorithm has been applied until the satisfaction of the positive definiteness. As it can be seen in Table 2, the smallest $\gamma_{min}$ value could be reached using the simple gamma-iteration, but the result obtained this way is just slightly different from the result obtained using LMI. However, the higher filter gain obtained in case of LMI suggests that the filter may be faster but less effective against disturbance. On other hand the burden of successive numerical computation of the quadratic matrix equality resulted in a significant computation cost. It has disadvantages despite its simplicity. From the results it is visible that modern computation methods as LMI are more capable to handle such complex mathematical problems as the solution of the MFARE. From the results mentioned above, it is conceivable that LMI-s are effective and powerful tools for handling complex but standard problems such as rapidly computing of a global optimum with some specified accuracy.

The technique of gamma-iteration, despite its slowness, is easy to be handled. Concretely it gives more flexibility to examine the solution for MFARE. For example, it is easy to analyze the impact of the *gamma* value on the number of iteration steps or the impact of changing of the disturbance on the optimal 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? How are they distributed? How does the filter gain change by reduction of the value of gamma? All these issues can be easily examined, step by step during the iterations.

## 4. Conclusion

In our paper we performed a benchmark based on collected concepts for solutions of MFARE by conventional gamma-iteration and LMI. From the simulation results of LMI, it can be concluded that it is well capable for computing the global optimum of the quadratic cost function rapidly with some specified accuracy even if this is to be done in the case of MFARE. Both methods, i.e. the gamma-iteration and the LMI formulation as a linear objective minimization problem, are capable for solution of MFARE. Moreover, they deliver only slightly different results. However, the LMI leads to an optimal solution faster, in about 100ms.

The analytically implemented gamma-iteration, despite its slowness, gives much more flexibility to examine the minimization process. For example, it is easy to examine the impact of the iteration steps or the impact of changing of the disturbance on the optimal solution. For this reason we propose the use of both approaches, that is, using the gamma-iteration in the preliminary stage in order to perform an analysis and using LMI in the stage of the synthesis to perform the implementation. Our further work will include an extension of our LMI approach to a switched linear system.

## References

[1] Edelmayer, A., Bokor, J., Keviczky, L., “An $H_\infty$ Filtering Approach to Robust Detection of Failures in Dynamical Systems”, *in Proc. 33th Annual Decision and Control, Conf.*, pp. 3037-3039, Buena Vista, USA, 1994.

[2] Edelmayer, A., Bokor, J., Keviczky, L., “An $H_\infty$ Filter Design for Linear Systems: Comparison of two Approaches” *IFAC 13$^{th}$ Triennial World Congress, San Francisco, USA*, 1996.

[3] Yaesh, I. , Shaked, U., “ Game Theory Approach to Optimal Linear State Estimation and Its Relation to the Minimum $H_\infty$ -norm Estimation”, *IEEE Trans. Aut. Control, AC-37(6)*, pp. 828-831, 1992.

[4] Chen, J., Patton, R. J., “Robust Model-Based Fault Diagnosis for Dynamic Systems”, First Edition, Springer Science & Business Media, New York, 1999.

[5] Matusu, R., “Linear Matrix Inequalities and Semidefinite Programming: Applications in Control”, Internal Journal of Mathematical Models and Methods in Applied Sciences, Vol. 8, 2014.

[6] Gahinet, P, Apkarian, P., “A linear matrix inequality approach to $H_\infty$ control”, International Journal of Robust and Nonlinear Control, Vol. 4, pp. 421–448, 1994.

[7] Iwasaki, T., Skelton, R. E., “All controllers for the general $H_\infty$ control problem: LMI existence conditions and state space formulas”, *Automatica*, Vol. 30, pp. 1307–1317, 1994.

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

[9] Chong, E., K., P., Zak, S., H., “An Introduction to Optimization”, 4th Edition, Wiley, New Jersey, 2013.

[10] Ostertag, E., “Mono- and Multivariable Control and Estimation: Linear, Quadratic and LMI Methods” , Matematical Engineering, Vol.2, Springer, Berlin, Heidelberg, 2011.

[11] Ankelhed, D., “On the design of low order $H_\infty$ controllers”, Ph.D. These, Linköping University, Linköping, 2011.

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

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

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

[15] Lunze, J., “Regelungstechnik 2 – Mehrgrößensysteme, Digitale Regelung“, Springer, 7. Auflage, 2013.

[16] Jankovic, M., Kolmanovsky, I., “Robust Nonlinear Controller for Turbocharged Diesel Engines”, *Procedings of the American Control Conference, Philadelphia*, 1998.

[17] Jung, M., “Mean-value modelling and robust control of the airpath of a turbocharged diesel engine”, Ph.D. These, University of Cambridge , 2003.

[18] Horvath, Zs., Edelmayer, A., “LTI-modelling of the Air Path of Turbocharged Diesel Engine for Fault Detection and Isolation”, *Mechanical Engineering Letters*, Vol. 14, pp. 172-188, Gödöllő, 2016.

[19] Herceg, M., “Nonlinear Model Predictive Control of a Diesel Engine with Exhaust Gas Recirculation and Variable Geometry Turbocharger”, Diploma Thesis, Slovak University of Technology in Bratislava, Bratislava, 2006.

[20] Yung, C. F., “Reduced-order $H_\infty$ controller design: An algebraic Riccati euqation approach”, Automatica, Vol. 36, pp. 923–926, 2000.

[21] Lanzon, A., Feng, Y., Anderson, B.,D.,O. and Rotkowitz, M., “ Computing the Positive Stabilizing Solution to Algebraic Riccati Equations With an Indefinite Quadratic Term via a Recursive Method”, *IEEE Trans. Aut. Control*, Vol. 53, pp. 2280–2291, NO. 10. November, 2008.

[22] https://de.mathworks.com/help/control/ref/care.html?requestedDomain=www.mathworks.com

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