Computers, Materials & Continua
DOI: 10.32604/cmc.2022.023030
Article

Tech Science Press

Smart Anti-Pinch Window Simulation Using H_/H$_\infty$ Criterion and MOPSO

Maedeh Mohammadi Azni[1], Mohammad Ali Sadrnia[1], Shahab S. Band[2,*] and Zulkefli Bin Mansor[3]

[1]Department of Electrical and Robotics Engineering, Shahrood University of Technology, Semnan, Iran
[2]Future Technology Research Center, College of Future, National Yunlin University of Science and Technology 123 University Road, Section 3, Douliou, Yunlin, 64002, Taiwan
[3]Faculty of Information Science and Technology, Universiti Kebangsan Malaysia, 43600, UKM Bangi, Selangor, Malaysia
*Corresponding Author: Shahab S. Band. Email: shamshirbands@yuntech.edu.tw
Received: 26 August 2021; Accepted: 10 November 2021

Abstract: Automobile power windows are mechanisms that can be opened and shut with the press of a button. Although these windows can comfort the effort of occupancy to move the window, failure to recognize the person's body part at the right time will result in damage and in some cases, loss of that part. An anti-pinch mechanism is an excellent choice to solve this problem, which detects the obstacle in the glass path immediately and moves it down. In this paper, an optimal solution H_/H$_\infty$ is presented for fault detection of the anti-pinch window system. The anti-pinch makes it possible to detect an obstacle and prevent damages through sampling parameters such as current consumption, the speed and the position of DC motors. In this research, a speed-based method is used to detect the obstacles. In order to secure the anti-pinch window, an optimal algorithm based on a fault detection observer is suggested. In the residual design, the proposed fault detection algorithm uses the DC motor angular velocity rate. Robustness against disturbances and sensitivity to the faults are considered as an optimization problem based on Multi-Objective Particle Swarm Optimization algorithm. Finally, an optimal filter for solving the fault problem is designed using the H_/H$_\infty$ method. The results show that the simulated anti-pinch window is pretty sensitive to the fault, in the sense that it can detect the obstacle in 50 ms after the fault occurrence.

Keywords: H_/H$_\infty$; anti-pinch; residual; fault detection; multi-objective particle swarm optimization (MOPSO); multi-objective optimization; automotive; power windows; electric windows

1 Introduction

The increasing growth of automated systems and their application in larger and more complex systems, and consequently the increased demand for safety systems, has led to a greater tendency towards fault detection techniques, particularly, model-based methods in dynamic systems. Failure to recognize in due time will result in damage and loss of a significant part of the capabilities and

information and human resources in some cases. Due to the damage caused by the fault, the industries were interested in seeking a way to minimize the incidence of faults. It is impossible to prevent a fault in control systems. However, if the fault can be detected in a timely fashion and identified dynamically, then by applying a proper control rule, the amount of damage can be reduced to an acceptable level. Systems that have such capabilities are called fault-tolerant control systems. In these systems, some performance drop is also acceptable in the event of a fault [1,2].

There are two main approaches to fault detection: model-based fault diagnosis and non-model fault diagnosis [3]. The non-model fault diagnosis does not require a dynamic model of the system to detect a fault [4]. An analytical model-based diagnostic method is one of the most important methods for fault detection [5]. The model-based methods are generally faster and more accurate than the non-model approach. These methods use the idea of producing the residual signal, which results from a mismatch between the estimated behavior and the real system [6]. There are various methods for generating the residual signal, which are based on the observer-based approach, parity equations approach, and parameter estimation approach [7].

In the automotive industry, electronic systems, as they provide the customer's comfort, encounter vehicles with new needs [8]. One of these needs is the use of an automatic power window, which requires the finger to be pressed on the command button to move the window until the end of its course.

Due to non-compliance with safety issues, there have been several incidents involving work with power windows. Therefore, failure to correctly detect the fault will damage the part of the body in the window movement path [2,9]. As a result, the anti-pinch window control system has been very much taken into consideration.

In general, common methods for detecting pinch conditions are divided into two batches. In the first method, the Pinch estimator assumes that the window velocity is significantly reduced and the motion is steady. In this type of estimation, the required value of calculations is low, but its performance decreases in the presence of noise.

Another type that detects the pinch condition is when the motor torque exceeds a predetermined value, requiring an additional current sensor to prevent false alarms [10–12].

H_-/H_∞ is a method that decreases the effect of disturbance on the residual and increases the effect of the fault on it. We use H_- to show the criterion of the fault effect [13], which should be maximized and used H_∞ to show the disturbance effect. The criterion H_- and H_∞ is defined as [14,15]:

$$H_-=||T_{rf}||_- = \inf \underline{\sigma}(T_{rf}(jw))>\beta_0 \tag{1}$$

$$H_\infty=||T_{rd}||_\infty = \sup\ \overline{\sigma}(T_{rd}(jw))<\gamma_0 \tag{2}$$

T_{rf} and T_{rd} are the transfers of the fault and the disturbance to the residual signal, respectively. The two scalars $\beta_0 > 0$, $\gamma_0 > 0$ [14]. It should be kept in mind that in condition (1), H_- is not really the norm. For fault detection, H_- is defined as follows.

Definition: For the system given as $y(s)= G(s)u(s)$ the criterion H_- is as [16]:

If the number of rows $G(s)$ is greater than the number of its columns and $\forall\omega$, $G(-j\omega)G^T(j\omega)> 0$, we have:

$$||G(s)||_- = \min_\omega \underline{\sigma}(G^T(j\omega)) \tag{3}$$

If the number of rows G(s) is less than the number of its columns and $\forall\omega.G^T(-j\omega)G(j\omega)> 0$:

$$||G(s)||_- = \min_\omega \underline{\sigma}(G(j\omega)) \tag{4}$$

To calculate this criterion, we introduce the following Lemma.

Lemma: Assume that A, B, P, S, R are matrices with suitable dimensions; while P and R are symmetric, $R> 0$ and (A, B) are stable. Suppose two assumptions are satisfied [16]:

1. A does not have an eigenvalue on the imaginary axis.
2. P is defined as a positive semi-definite or negative semi-definite, while for (A, P), there is no observable mode on the imaginary axis.

Then X is a unique real symmetric response for the following equation [17]:

$$(A - BR^{-1}S^T)^TX + X(A - BR^{-1}S^T) - XBR^{-1}B^TX + P - SR^{-1}S^T = 0 \tag{5}$$

where, $A - BR^{-1}S^T - BR^{-1}B^TX$ is stable.

In this research, an optimal algorithm is considered using the H_-/H_∞ method for the anti-pinch window system. In this algorithm, angular velocity rate information is used as a fault under pinch conditions. The fault detection observer is designed sensitive to the fault and robust to disturbances. To be more specific, in the result section, we can see that the fault detection has happened within 50 ms after the fault occurrence, resulting in having a window with negligible damage to the obstacle which is on its path.

2 Methodology

In most cars, lifting the window is done using a very precise lever that keeps it leveling up. A small electric motor is connected to a spiral gear and several circular gears to produce sufficient lifting force to lift the window.

One of the important features of the power windows is that they cannot be opened by pressing; the helical gear prevents it. Most spiral gears are locked automatically due to the angle of contact between the helical gear and the round gear teeth. The helical gear can rotate the round gear, but the rounded gear cannot rotate the helical gear. The friction between the teeth makes the gears lock. The lever has a long arm attached to the bar that holds the bottom of the window. When the window rises, the end of the arm slips into the slot on the bar. On the other hand, there is a large plate at the end of the rod, where the gear teeth fall into it. The motor rotates the gear engaged with these teeth.

New generation cars are equipped with power windows that reduce the passenger's quest for moving car windows [18]. If the object is placed in the window path, for example, the child's hand, the window must stop at the moment. One method that automakers use to control the power of the window is the design of a circuit that measures the torque of the lift motor. If the motor torque decreases, the circuit will invert the inlet to the motor, and thus the window will return to the bottom. But the problem here is how long a fault is detected [19]. In fact, less time is desirable for our problem since less damage occurs to the obstacle. The overall view of the power window is shown in Fig. 1.

We consider the linear time-invariant system as following [14]:

$$\begin{aligned} \dot{x} &= Ax + Bu + E_f f + E_v v + E_n n \\ y &= C_2x + D_2u + F_f f + F_v v + F_n n \end{aligned} \tag{6}$$

where x is the state vector, u input control, y is the measurement vector, n is white noise, v is the disturbance, and f is the fault vector input.

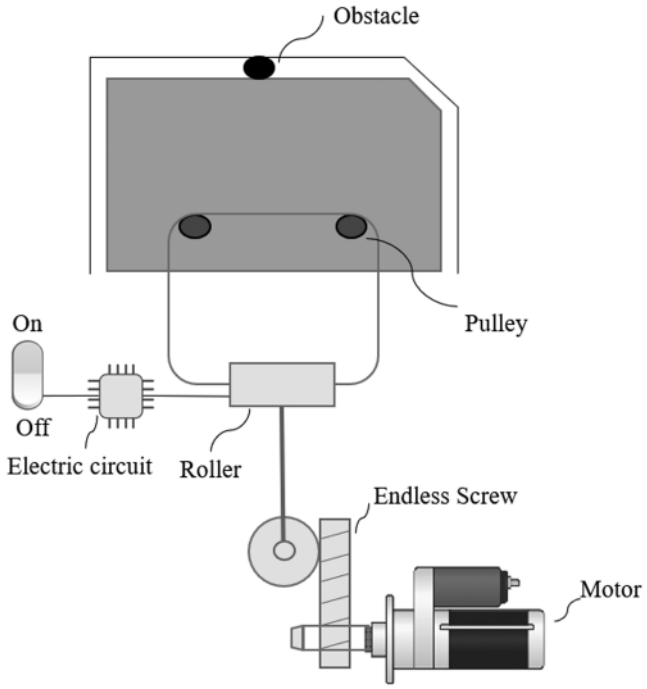

Figure 1: The overall view of the power window

The residual generator is considered as follows [14]:

$$\begin{aligned}\dot{\hat{x}} &= (A - LC_2)\hat{x} + (B - LD_2)u + Ly \\ \hat{y} &= C_2\hat{x} + D_2u \\ r &= y - \hat{y}\end{aligned} \tag{7}$$

where L, $\hat{y}$, y and r are the observer matrix gain, predicted output, observed output and residual vector, respectively. Using (6) and (7):

$$\begin{aligned}\dot{e} &= (A - LC_2)e + (E_f - LF_f)f + (E_v - LF_v)v + (E_n - LF_n)n \\ r &= C_2e + F_ff + F_vv + F_nn\end{aligned} \tag{8}$$

Which shows that the dynamics of the residual signal r depends on the state e in addition to the parameters f, v and n. Another form of residual in Eq. (8) is as follows [14]:

$$r(s) = T_{rf}f(s) + T_{rn}n(s) + T_{rv}v(s) \tag{9}$$

where T_{rf}, T_{rn} and T_{rv} are transfer functions from f, n and v to r, respectively.

2.1 Analytical Model of the Fault Detection Algorithm

The angular velocity ω is considered as a state variable and we have $\dot{\omega}$ from [14]:

$$\dot{\omega} = -\frac{b}{J}\omega + \frac{1}{J}T_c - \frac{1}{J}(T_p + T_w) + u_v \tag{10}$$

where b, J, T_c, T_p and T_w are viscous friction coefficient, moment inertia, control torque, pinch torque and load torque, respectively. In addition, $T = T_p + T_w$ is considered as the second state variable. The torque rate is also added as the third state variable:

$$\ddot{T} = u_{Td} \tag{11}$$

The matrices A, B, C_2 and D_2 of the Eq. (7) are as follows [14]:

$$A = \begin{bmatrix} -\frac{k_e k_t}{JR_m} & -\frac{1}{J} & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} \frac{k_t}{JR_m} \\ 0 \\ 0 \end{bmatrix}, \; C_2 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \; D_2 = [0]$$

The DC motor parameters are obtained by experiments. The values of these parameters are shown in the Tab. 1.

Table 1: Nominal values of motor parameters [14]

Motor parameters	Value
R_m	0.85 [Ω]
L_m	0.649 [mH]
K_e	0.1204 [V/s/rad]
K_t	0.1204 [V/s/rad]
T_n	9.3×10^{-3} [s]
J	1.586×10^{-4} [kg/m^2]
V_c	12[V]

By placing the above table data in the matrices A, B, C_2, D_2 :

$$A = \begin{bmatrix} -107.530 & -6305.170 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \; B = \begin{bmatrix} 893.109 \\ 0 \\ 0 \end{bmatrix} C_2 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, \; D_2 = [0]$$

According to Eq. (6), to detect the fault, we select the matrices E_f, E_v, E_n, F_f, F_v and F_n as [14]:

$$E_n = \begin{bmatrix} 893.109 \\ 0 \\ 0 \end{bmatrix}, \; E_v = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \; E_f = \begin{bmatrix} 893.109 \\ 0 \\ 0 \end{bmatrix}, \; F_n = [0.05], \; F_v = [0], \; F_f = [1]$$

3 Simulation of the System by Solving the MOPSO Algorithm

We want to transform the problem of controller design into an optimization problem and then solve the problem using the MOPSO algorithm [20]. We design the system in a way that can withstand any kind of uncertainty. We consider the linear state equation as [17]:

$$\dot{x} = Ax + Ew + Bu \tag{12}$$

where $w = \begin{bmatrix} f \\ v \\ n \end{bmatrix}$, $E = \begin{bmatrix} E_f & E_V & E_n \end{bmatrix}$ and the output equation is as:

$$z = C_1x + F_1w + D_1u \tag{13}$$

And, the measurement equation is:

$$y = C_2x + F_2w + D_2u \tag{14}$$

where $F_2 = \begin{bmatrix} F_f & F_V & F_n \end{bmatrix}$.

The goal is to put the system in a proper position by applying a u so that z meet zero [21,22].

According to the Eqs. (12)–(14) we have:

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} = \begin{bmatrix} A & E & B \\ C_1 & F_1 & D_1 \\ C_2 & F_2 & D_2 \end{bmatrix} \begin{bmatrix} x \\ w \\ u \end{bmatrix}$$

The transfer function model is as [23]:

$$P(s) = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} (sI - A)^{-1} \begin{bmatrix} E & B \end{bmatrix} + \begin{bmatrix} F_1 & D_1 \\ F_2 & D_2 \end{bmatrix} = \begin{bmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{bmatrix} \tag{15}$$

And, the closed-loop transfer function for input w and output z is:

$$T_{zw}(s) = P_{11} + P_{12}L(I - P_{22}L)^{-1}P_{21} \tag{16}$$

Since, $T_{zw}(s)$ is a multivariable transfer function, we should use the norm of that to solve the problem. At first, we consider the constant gain controller design problem as an optimization problem. Afterward, with the help of the MOPSO algorithm, we find L so that $\|T_{zw}(s)\|$ is minimized. In fact, z must remain independent of the w; if we can supply it, we will have a robust controller.

The design goal in our chosen system is to reduce the angular velocity rate of the motor $\dot{\omega}$ to zero. In fact, in our chosen system $Z \propto \dot{\omega}$.

Therefore, Eqs. (12)–(14) are rewritten as:

$$\begin{bmatrix} \dot{\omega} \\ \dot{T} \\ \ddot{T} \end{bmatrix} = \begin{bmatrix} -107.5 & -6305.2 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ T \\ \dot{T} \end{bmatrix} + \begin{bmatrix} 893.1 & 1 & 893.1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} f \\ v \\ n \end{bmatrix} + \begin{bmatrix} 893.1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ u_v \\ u_{Td} \end{bmatrix} \tag{17}$$

$$z = \begin{bmatrix} -107.5 & -6305.2 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ T \\ \dot{T} \end{bmatrix} + \begin{bmatrix} \alpha & 1 & 893.1 \end{bmatrix} \begin{bmatrix} f \\ v \\ n \end{bmatrix} + \begin{bmatrix} 893.1 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ u_v \\ u_{Td} \end{bmatrix} \tag{18}$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \omega \\ T \\ \dot{T} \end{bmatrix} + \begin{bmatrix} 893.1088 & 0 & 0 \end{bmatrix} \begin{bmatrix} f \\ v \\ n \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ u_v \\ u_{Td} \end{bmatrix} \tag{19}$$

In the Eq. (18), $\alpha \in [890 \quad 896]$ is uncertainty and $\alpha_0 = 893.1$ [14]. The toolbox is simulated to create a robust fault detection system, as shown in Fig. 2.

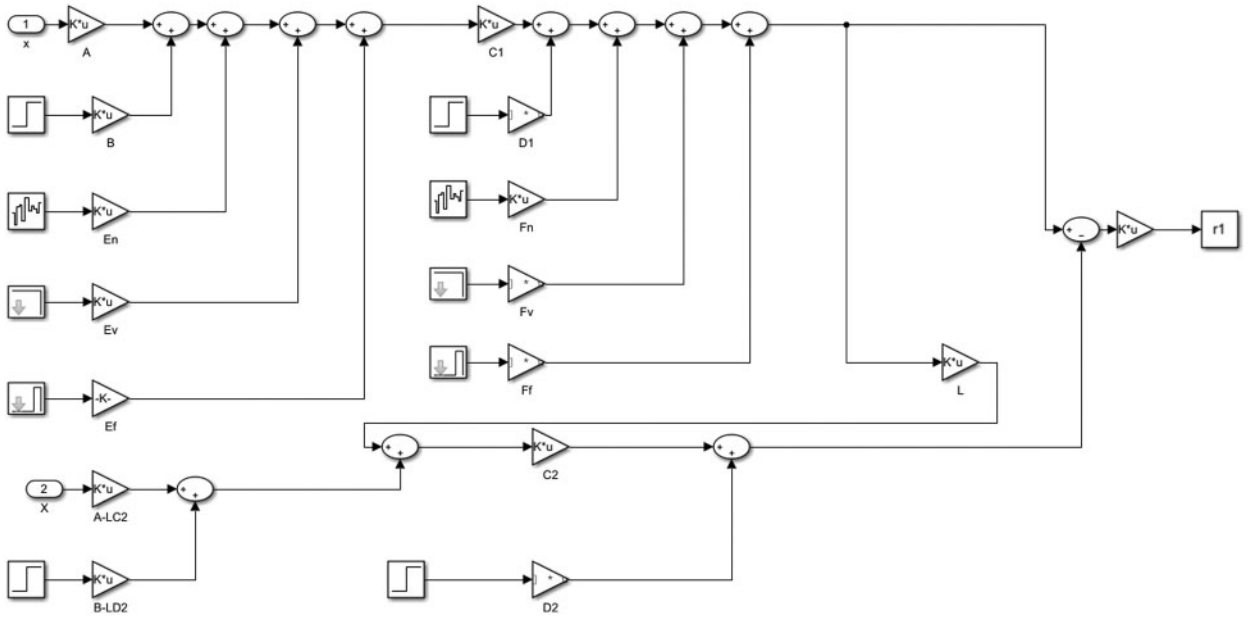

Figure 2: Anti-pinch window simulation toolbox

The impulse signal is v with a delay of 0 s, a domain of 50, and a width of 0.3 s. In addition, white noise is simulated with 0.00014, and its sampling frequency is 0.01.

The delay of impulse signal f is 2.7 s, with the amplitude of 1, and the width of 1 s. Now, we obtain a lookup using the MOPSO algorithm in MATLAB software.

4 Simulation and Results

In this section, we face a multi-objective optimization problem. The goal of problem-solving is to reduce the disturbance effect and increase the fault effect, meaning that we have to define two objective functions and simultaneously optimize their value [24,25]. Therefore, we use the MOPSO algorithm and define the cost functions according to the criterion H_-/H_∞ , as:

$$Z_1 = ||T_{zw_d}||_\infty$$

$$Z_2 = ||T_{zw_f}||_- \tag{20}$$

where $w_d = \begin{bmatrix} v \\ n \end{bmatrix}$ and $w_f = [f]$. Afterward, the observer's gain is obtained as follows:

$$L = \begin{bmatrix} 00.0459 \\ 0.08533 \\ 1 \end{bmatrix}$$

Fig. 3 shows the response of the MOPSO algorithm, which is represented by o and $*$, where $*$ are members of the archive. The horizontal and vertical graphs represent the Z_1 and Z_2 cost functions in Eq. (20) respectively. It can be seen that each particle has been selected from the leader's archive and has done its own.

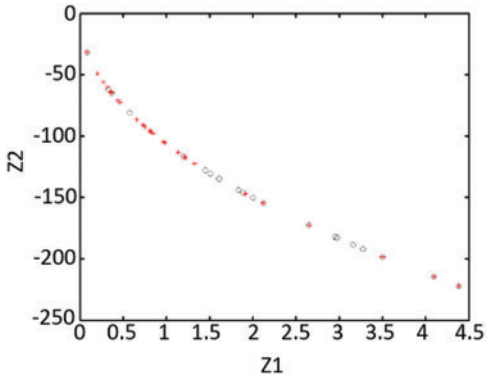

Figure 3: The cost functions of the mopso algorithm in the h_-/h_∞ method

As shown in Fig. 4, the sensitivity to the disturbance is well-faded and H_-/H_∞ is succeeded in separating the fault from disturbance. We can get the fault detection time using the threshold level which is shown with a discontinuous line in Fig. 4. This figure shows that after the pinch occurs in 2.7 s, the detection is done after 50 msec.

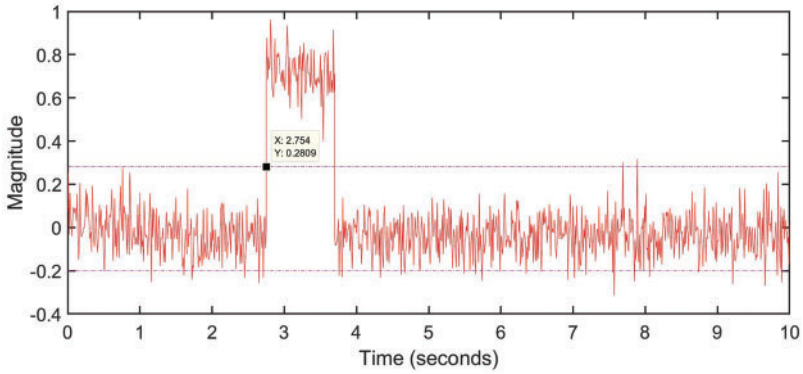

Figure 4: The residual signal in the h_-/h_∞ method in the presence of fault and disturbance

4.1 Closed-Loop System Analysis with L Controller and H_-/H_∞

In control systems, in order to formulate the controller design problem meeting the required characteristics, we use weighting functions [26]. In this section, we use the H_-/H_∞ criterion and select

20 points from the shape of the bode diagram of an additive fault which is shown in Fig. 5. Afterward, we consider w_a (weighting functions) as a second-order function, resulting in obtaining its transfer function according to the selected points as follows:

$$w_a = \frac{4845\ s^2\ +\ 3189\ s\ +\ 0/5558}{s^2 + 3206\ s\ +\ 175000}$$

To achieve the optimal level of disturbance reduction, it is necessary to establish a relation between S (the closed-loop system sensitivity function) and w_a : $||[w_a S]||_\infty < 1$ [26].

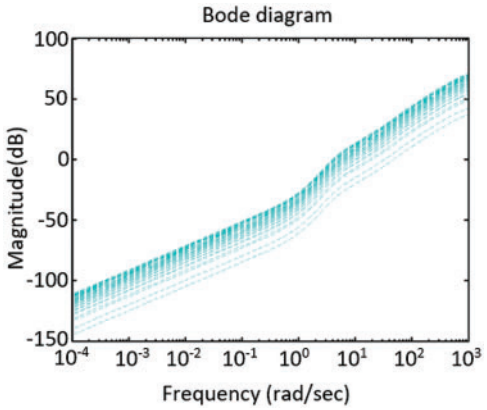

Figure 5: Bode diagram of an additive fault

We assume that for all types of frequency:

$$||(\mathrm{I} + \mathrm{GL})^{-1}||_\infty < \frac{1}{w_a} \tag{21}$$

To reach this condition, we can say if and only if $\sigma[(I + GL)^{-1}] < \left|\frac{1}{w_a}\right|$. To be more specific, if the frequency sensitivity function remains within limits imposed by the inverse weighting function, the control performances are met [27]. The exceptional values of $\frac{1}{w_a}$ are shown in Fig. 6.

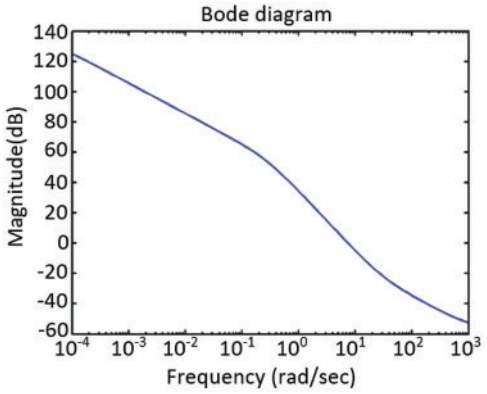

Figure 6: Exceptional values $\frac{1}{w_a}$ using the h_-/h_∞

The result of the comparison between the sensitivity function and the inverse weighting function is shown in Fig. 7. As can be seen, the sensitivity function is located below the $\frac{1}{w_a}$ graph at low frequencies as it was our required goal to meet control performances.

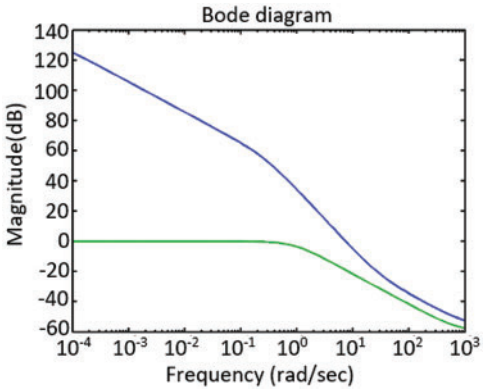

Figure 7: The inverse of the weighted function (blue line) and the sensitivity function using the h_-/h_∞ (green line)

4.2 Comparison of System Simulation Results with MOPSO and LMI Algorithms

Using the LMI algorithm, the observer's gain for this system is obtained from [14]:

$$L = \begin{bmatrix} 7.3967 \\ -0.0164 \\ 0.0201 \end{bmatrix}$$

Fig. 8 shows the residual signal in the LMI and MOPSO methods using the H_-/H_∞ criterion. The comparison of the pinch detection time in these two methods shows that MOPSO has been more successful and can detect a fault 20 milliseconds earlier than LMI, resulting in reducing the injury that is applied to the obstacle in the window path.

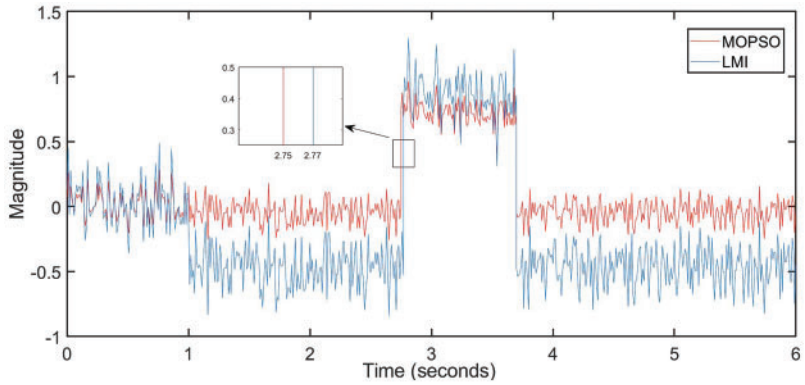

Figure 8: The residual signal comparison in the mopso and lmi method using the h_-/h_∞

5 Conclusions

In this paper, the design of the H_-/H_∞ observer and its use in detecting and isolating the sensor fault in the anti-pinch system of the car were discussed. According to the results obtained, it is pretty evident that in the H_-/H_∞ method, the effect of the fault on the residual signal increases, and at the same time, the effect of the unknown input is almost fading. In this paper, the MOPSO method was used to solve the H_-/H_∞ problem; it was shown that the proposed algorithm could well capture the optimal value for observer gain. This simulation shows that the effect of the fault on the residual has increased, and the observer can detect the fault in 50 ms after once the glass of window meets the obstacle placed in the path of the window. The results show that the H_-/H_∞ method is effective to solve our problem and achieve our requirement. However, in this method, we had to set the values of the noise, disturbance, and fault to simulate the model; if we have a method that does not need to set the values for these three parameters, we will have a simpler simulation model for this system.

Funding Statement: This research was supported by DP-FTSM-2021, Dana Lonjakan Penerbitan FTSM, UKM.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] I. Hwang, S. Kim, Y. Kim and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 3, pp. 636–653, May 2010.

[2] X. He, Z. Wang, Y. Liu, L. Qin and D. Zhou, "Fault-tolerant control for an internet-based three-tank system: Accommodation to sensor bias faults," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2266–2275, Mar. 2017.

[3] I. Samy, I. Postlethwaite and D. -W. Gu, "Survey and application of sensor fault detection and isolation schemes," *Control Engineering Practice*, vol. 19, no. 7, pp. 658–674, Jul. 2011.

[4] J. Ma and J. Jiang, "Applications of fault detection and diagnosis methods in nuclear power plants: A review," *Progress in Nuclear Energy-PROG NUCL ENERGY*, vol. 53, pp. 255–266, Apr. 2011.

[5] A. Mojallal and S. Lotfifard, "Multi-physics graphical model-based fault detection and isolation in wind turbines," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 5599–5612, Nov. 2018.

[6] E. Sobhani-Tehrani and K. Khorasani, *Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach*, vol. 383, Springer Science & Business Media, pp. 21–49, 2009.

[7] R. Isermann, "Model-based fault-detection and diagnosis–status and applications," *Annual Reviews in Control*, vol. 29, no. 1, pp. 71–85, Jan. 2005.

[8] J. Kunkel, "Anti-pinch window control system," US7605554B2, Oct. 20, 2009 Accessed: Aug. 25, 2021. [Online]. Available: https://patents.google.com/patent/US7605554/en.

[9] H. -J. Lee, W. -S. Ra, T. -S. Yoon and J. -B. Park, "Practical pinch torque detection algorithm for anti-pinch window control system application," *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 2526–2531, 2005.

[10] H. -W. Kim and S. -K. Sul, "A new motor speed estimator using kalman filter in low-speed range," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 4, pp. 498–504, Aug. 1996.

[11] G. S. Buja, R. Menis and M. I. Valla, "Disturbance torque estimation in a sensorless DC drive," *IEEE Transactions on Industrial Electronics*, vol. 42, no. 4, pp. 351–357, Aug. 1995.

[12] C. De Angelo, G. Bossio, J. Solsona, G. O. Garcia and M. I. Valla, "Mechanical sensorless speed control of permanent-magnet AC motors driving an unknown load," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 2, pp. 406–414, Apr. 2006.

[13] A. Alif, M. Darouach and M. Boutayeb, "Design of robust $_{\text{infty}}$ reduced-order unknown-input filter for a class of uncertain linear neutral systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 6–19, Jan. 2010.
[14] H. Li, X. Wang, F. Liu, H. Chen and Y. Meng, "Robust fault detection algorithm for the smart anti-pinch window of pure electric vehicles," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 5, pp. 8683–8693, May 2013.
[15] Z. Li, E. Mazars and I. M. Jaimoukha, "State–space solution to the H_{-}/H_{∞} fault-detection problem," in *IEEE Conference on Decision and Control. 45th International Conference*, San Diego, USA, pp. 2177–2182, 2007.
[16] S. X. Ding, "Residual generation with enhanced robustness against unknown inputs:" *Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*, vol. 473, Springer Science & Business Media, pp. 211–232, 2008.
[17] C. -T. Chen, "Stability:" *Linear system theory and design*, 3rd ed. New York: Oxford University Press, UK, 1999.
[18] N. A. John, M. Sherki and S. A. Patil, "Anti-pinch mechanism for power window," in *SAE International, Warrendale*, PA, SAE Technical Paper 2016-28–0197, USA, 2016.
[19] R. P. Gerbetz, "Anti-pinch window control system and method," EP1402611B1, Mar. 21, 2012 Accessed: Aug. 25, 2021. [Online]. Available: https://patents.google.com/patent/EP1402611B1/en.
[20] H. R. Madvar, M. Dehghani, R. Memarzadeh, E. Salwana, A. Mosavi *et al.,* "Derivation of optimized equations for estimation of dispersion coefficient in natural streams using hybridized ANN with PSO and CSO algorithms," *IEEE Access*, vol. 8, pp. 156582–156599, 2020.
[21] A. Arram, M. Ayob and A. Sulaiman, "Hybrid bird mating optimizer with single-based algorithms for combinatorial optimization problems," *IEEE Access*, vol. 9, pp. 115972–115989, 2021.
[22] M. S. A. Daweri, S. Abdullah and K. A. Z. Ariffin, "A Migration-based cuttlefish algorithm with short-term memory for optimization problems," *IEEE Access*, vol. 8, pp. 70270–70292, 2020.
[23] K. J. Åström and R. M. Murray, "Transfer function:" *Feedback Systems: An introduction for scientists and engineers*, Second edition. USA: Princeton University Press, 2021.
[24] R. Sihwail, O. S. Solaiman, K. Omar, K. A. Z. Ariffin, M. Alswaitti *et al.,* "A hybrid approach for solving systems of nonlinear equations using harris hawks optimization and newton's method," *IEEE Access*, vol. 9, pp. 95791–95807, 2021.
[25] T. -A. N. Abdali, R. Hassan, R. C. Muniyandi, A. H. Mohd Aman, Q. N. Nguyen *et al.,* "Optimized particle swarm optimization algorithm for the realization of an enhanced energy-eware location-aided routing protocol in MANET," *Information*, vol. 11, no. 11, pp. 529, Nov. 2020.
[26] S. A. Frank, "Regulation:" *Control theory tutorial: Basic concepts illustrated by software examples*," *Cham: Springer International Publishing*, 2018.
[27] M. Dulau and S. -E. Oltean, "The effects of weighting functions on the performances of robust control systems," *Proceedings*, vol. 63, no. 1, pp. 46, 2020.