Network Calculus Based Fault Diagnosis Decision-Making for Networked Control System

Christophe Aubrun, Jean-Philippe Georges, Dominique Sauter, Eric Rondeau

Network Calculus Based Fault Diagnosis Decision-Making for Networked Control Systems

Christophe Aubrun, Jean-Philippe Georges, Dominique Sauter, Eric Rondeau
Centre de Recherche en Automatique de Nancy
Nancy-Université, CNRS
Faculté des Sciences, BP 239
F-54506 Vandœuvre les Nancy cedex, France
christophe.aubrun@cran.uhp-nancy.fr

Abstract

This paper deals with the problem of Fault Detection and Isolation (FDI) in Networked Control Systems (NCS). The effects of unknown networked induced delays on conventional observed based residual generator are studied. It is shown that the detection performances may be reduced due to the sensitivity of the residuals signals to the delays. With the assumption that the network delays belong to a given bounded set, a threshold is defined on the basis of the network characteristic, in order to enhance the robustness of the decision making process. The detection thresholds, which depend on the bounded values of the network delays, are determined from the network calculus theory.
Keywords: fault detection, residual generation, network calculus, network induced delays, robustness.

1. Introduction

A new trend in the field of fault diagnosis for technical systems is the implementation of supervision functionalities such as performance evaluation, Fault Diagnosis and Isolation (FDI) procedure and reconfiguration mechanisms on networked architecture systems. The study and the design of such application, called Networked Control Systems (NCS) [11], has been becoming an important emerging research field. Usually NCS are subject to unknown network induced delays, and data dropouts [10, 11]. The control issues of NCS have attracted most attention of many researchers with taking into account network characteristics [6, 9, 18], but very few treat the FDI for NCS problem. The procedures of model-based FDI problem are:

- generations of residuals which are ideally close to zero under fault-free conditions, minimally sensitive to noises and disturbances and maximally sensitive to faults,
- residual evaluation, namely design of decision rules based on these residuals.

When involving the model-based FDI of NCS, the aforementioned problems will be more complex than those of traditional point-to-point systems because of the network induced effects and conventional theories with ideal assumptions such as non-delayed sensing and actuation must be re-evaluated. In this paper, we are interested in evaluating first the effects of the communication induced delays in the performance of the FDI system. It will be shown that the performances of FDI are altered by the delay (see figure 1). Depending on the input signal applied to the system, delays can induce false alarms. In order to overcome this problem, the threshold which is used in the decision making is adjusted according to modifications of the network characteristics.

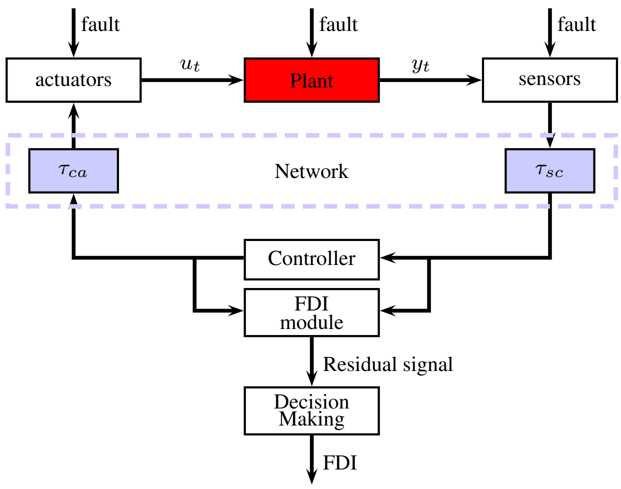

Figure 1. Conventional FDI scheme in the NCS context

The proposed approach for threshold adaptation is based on the value of the upper bound value of the induced delay. The delay information is determined by means of the network calculus theory. The paper is organized as follows. Section II introduces the general principle of conventional FDI. Section III presents FDI approach in the NCS context and the effects of the network on the residual signal. Section IV introduces the network calculus tools and conclusion and further work are discussed in the last section.

2. FDI principle

Fault diagnosis implies to design residuals close to zero in the fault-free case and deviated from zero in the presence of faults. Residuals should possess the ability to discriminate between all possible modes of faults. Residuals generation having directional properties in response to a particular fault is an attractive way for enhancing the fault isolability. The fault isolation filter, proposed in this paper, is a special full-order state observer which generates output residuals with directional properties in response to each fault. First developed by Beard [1], fault detection filter has been revisited by Massoumnia [13], from the geometric state-space control theory and by White and Speyer [17] in the context of eigenstructure assignment. Further improvements were suggested by Liu and Si [12] and Keller and Sauter [7].

In this section, we introduce the conventional structure of the FDI filter. Let us consider a classical sensor faults state space representation:

$$\begin{cases} x\left(t\right) = Ax\left(t\right) + \sum_{i=1}^{m} B_i u_i\left(t\right) + Ef\left(t\right) \\ y\left(t\right) = Cx\left(t\right) + Ff\left(t\right) \end{cases} \tag{1}$$

where $x \in \Re^n$, $y \in \Re^n$, $u \in \Re^p$, F and E are the faults distribution matrices and $f \in \Re^p$ is the fault magnitude vector. All matrices are assumed to be known with appropriate dimensions, also $rank\, C = m$ and $rank\, F = p \leq m$.

It is assumed that the residual generation and evaluation steps are executed instantaneously at each sampling period k. Based on this assumption, if the control signal is kept constant over each sampling interval T, and if it is considered that fault signal is constant, the discrete time system can be described by:

$$\begin{cases} x\left(\left(k+1\right)T\right) = \Phi x\left(kT\right) + \sum_{i=1}^{m} \Gamma_i u_i\left(kT\right) + \Xi f\left(kT\right) \\ y\left(kT\right) = Cx\left(kT\right) + Ff\left(kT\right) \end{cases} \tag{2}$$

where

$$\Phi = e^{AT};\ \Gamma_i = \int_0^T e^{As} B_i ds;\ \Xi = \int_0^T e^{As} E ds \tag{3}$$

The fault diagnosis procedure consists in three stages:

- Residuals generation: it consists in associating a model-observation pair to evaluate difference with respect to the normal operating conditions;
- Residuals evaluation: the residuals are compared to some predefined threshold according to a test and, at this stage, symptoms are produced;
- Decision making: the role of the decision making is to decide according to the symptoms, which elements are faulty, that is isolation.

This implies the design of residuals that are close to zero in the fault-free situations while clearly deviating from zero in the presence of faults and that possess the ability to discriminate between all possible modes of faults, which explain the use of the term decision-making. Classically, decision-making is realized according to an elementary logic based on a comparison between a predefined threshold and the residual signal. Then, the problem is to design an observer whose inputs are the measurable outputs and control inputs which generates the residual signals, in such a way that the influence of the faults is decoupled from each other and from the disturbances effects.

Let us consider an observer for system (1) of the form:

$$\begin{cases} \hat{x}\left(\left(k+1\right)T\right) = \Phi\hat{x}\left(kT\right) + \sum_{i=1}^{m} \Gamma_i u_i\left(kT\right) \\ \qquad + L\left(y\left(kT\right) - \hat{y}\left(kT\right)\right) \\ \hat{y}\left(kT\right) = C\hat{x}\left(kT\right) \end{cases} \tag{4}$$

The residual generator is given by:

$$r\left(kT\right) = G\left(y\left(kT\right) - \hat{y}\left(kT\right)\right) \tag{5}$$

where L and $G \in \Re^{n\times m}$ are design matrices determined such that detection and isolation requirements are met. The estimation error, ϵ, and the residual equation, r, are determined from equations (4) and (5):

$$\begin{aligned} \epsilon\left(\left(k+1\right)T\right) &= \left(\Phi - LC\right)\epsilon\left(kT\right) + \left(\Xi - LF\right)f\left(kT\right) \\ r\left(kT\right) &= G\left(y\left(kT\right) - \hat{y}\left(kT\right)\right) \end{aligned} \tag{6}$$

There are various approaches [2, 4, 15] for determining the gain matrices L and G. This problem will not be treated in this paper. The residual generator is given by equations (4) and (5) will be considered in the context of NCS.

3. FDI for networked control systems

It is now supposed that the system is controlled over a switched network. In this case, the sensor to controller delays and controller to actuator delays, have to be taken into account as it illustrated at figure 2.

Note that the delays, in general, cannot be considered constant and known. Network induced delays may vary,

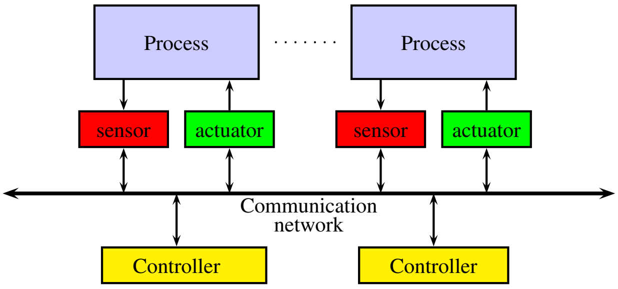

Figure 2. Networked system architecture

depending on the network traffic, medium access protocol and the hardware. From the controller viewpoint the system of equation (1) can be described by the model:

$$\begin{cases} x(t) = Ax(t) + \sum_{i=1}^{m} B_i u_i(t-\tau_i) + Ef(t) \\ y(t) = Cx(t) \end{cases} \quad (7)$$

where τ_i represents the global induced delays observed from the controller side which include the delay from controller to actuators τ_{ca} and the delay from sensors to controller, τ_{sc} ($\tau_i = \tau_{sc} + \tau_{ca}$). When the controller receives the measurement from the sensor, it immediately computes the new control input value u_k and a 'ZOH' device holds this value constant until the next measurement reading.

The discrete time model in equation (1) is then given by:

$$\begin{aligned} x((k+1)T) =& \Phi x(kT) + \sum_{i=1}^{m} \Gamma_{0,i} u_i(kT) \\ &+ \sum_{i=1}^{m} \Gamma_{1,i} u_i((k-1)T) + \Xi f(kT) \end{aligned} \quad (8)$$

where

$$\Gamma_{0,i} = \int_0^{T-\tau_i} e^{As} B_i ds; \; \Gamma_{1,i} = \int_{T-\tau_i}^{T} e^{As} B_i ds \quad (9)$$

Equation (8) is rewritten under the following form:

$$\begin{cases} x((k+1)T) = \Phi x(kT) + \sum_{i=1}^{m} \Gamma_{0,i} u_i(kT) \\ \qquad - \sum_{i=1}^{m} \Gamma_{1,i} \Delta u_i(kT) + \Xi f(kT) \\ y(kT) = Cx(kT) + Ff(kT) \end{cases} \quad (10)$$

with

$$\Delta u_i(kT) = u_i(kT) - u_i((k-1)T) \quad (11)$$

In order to achieve the FDI step within the networked architecture, the residual signal has to be defined from equation (10) and (5). Firstly, the estimation error is expressed by:

$$\epsilon(kT) = x(kT) - \hat{x}(kT)$$

and the residual $r(kT)$ propagates as

$$\begin{cases} \epsilon((k+1)T) = (\Phi - LC)\epsilon(kT) + \sum_{i=1}^{m} \Gamma_{1,i} \Delta u_i(kT) \\ \qquad + (\Xi - LG) f(kT) \\ r(kT) = GC\epsilon(kT) \end{cases} \quad (12)$$

It appears from equation (12), that the residual signal depends on the amplitude of the term:

$$\sum_{i=1}^{m} \Gamma_{1,i} \Delta u_i(kT)$$

which is a function of τ_i. In the conventional FDI scheme the residual is compared to a threshold which can be fixed or time varying. In this case, the threshold is independent from the network induced delay. The threshold is defined as follows:

$$\begin{matrix} \Psi(r_i(kT)/f_i) \geq Th(kT) & \text{for } f_i \neq 0 \\ \Psi(r_i(kT)/f_i) < Th(kT) & \text{for } f_i = 0 \end{matrix} \quad (13)$$

where $\Psi(t) = |r(t)|$ is an evaluation function. In order to improve the robustness of the decision making against τ_i, the threshold is designed according to the network characteristics as follows:

$$Th(r_i(kT)/f_i) = \sup|r_i(kT)| = \sup|G\epsilon(kT)|$$

Let us consider $\overline{\tau}$ and $\underline{\tau}$ respectively the upper bound value and the lower bound value of the induced delay. Then with:

$$|\overline{r}(kT)| = \sup_{0 \leq \tau_i \leq \overline{\tau}} |r(kT)|$$

The following inequality holds

$$|\underline{r}(kT)| \leq |r(kT)| \leq |\overline{r}(kT)|$$

Several approaches have been proposed in the literature for adaptive thresholding [15], and interval analysis [16]. The interval analysis is based on the network characteristic provided by mean of the network calculus theory.

4. Network Calculus

4.1. Introduction

Delays in a network mainly depend on the protocols involved (especially the medium access method), the network topology and the traffic load. Resulting from the load of traffic variation or even the considerable changes in the network topology, network induced delays might be characterized by huge time variance. This is especially true when the network is shared with applications other than process control.

A first approach relies on delays measurement. Usually, the delay measurement relies on the round trip time (RTT) measurement due to its easy implementation: no clock synchronisation is required since the computations are running on the same device. However, for the NCS, RTT may not be adequate and instrumentation for measuring the end-to-end delay should be developed. The main difficulty in measuring an end-to-end transmission delay is due to timing issues like non-synchronized clocks and scheduling policies of the operating system stack. Consequently, unidirectional delays measurement rely on the synchronisation of the clocks of the two end-systems as presented in [14]. Since the obtained delay measurement corresponds to the delay experienced by the last message for a given flow, this methodology only enables an estimate to be made for the latest delay. As this is based on past observations, it gives information about the trends and time variance of the delay. As a consequence of the procedure, the measurement of the delay will only be available at the receiver side. In NCSs, this implies that the transmission delay of the last control information is stored on the actuator, even though this knowledge is more relevant to the controller.

Consequently, a second approach consists to look for robust methods that will enable to take into account uncertainties introduced by the unknown delays time variance. In that way, the paper presents in the following a method based on the Network Calculus theory in order to upper-bound $\overline{\tau_{ca}}$ and $\overline{\tau_{sc}}$, i.e. the control and measurement delays. Those upper-bounds will be used to adapt the FDI residual to the delays.

The upper bound delay estimation algorithm applies ideas from network calculus theory, see [3, 8]. In this paper, the network architecture considered corresponds to a switched Ethernet architecture (linked to IEEE 802.1D standard). The approach consists of modelling switches as a combination of basic components: multiplexers, demultiplexers and FIFO queues, as shown in figure 3.

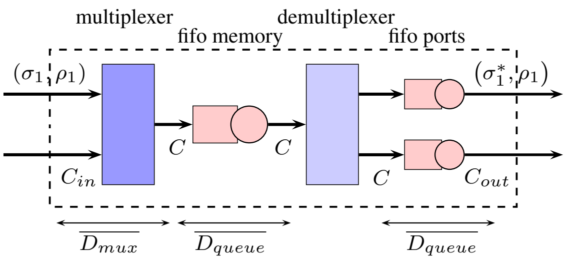

Figure 3. Model of a 2-port switch in a full duplex mode based on shared memory and a cut-through management

4.2. Maximum delay for crossing one Ethernet switch

In the mathematical analysis, the traffic arriving at the switch, both periodic and aperiodic is modelled as a leaky bucket controller. Data will arrive at the leaky rate only if the level of the bucket is less than the maximum bucket size. In network calculus theory the traffic models are represented as arrival curves and, with the assumption that the traffic follows the leaky bucket model and that the incoming rate is limited by the port capacity, these curves are affine and have the form:

$$b(t) = \min(C_{in}t, \sigma + \rho t)$$

where σ is the maximum amount of data that can arrive in a burst, ρ is an upper bound of the average rate of the traffic flow, and C_{in} is the capacity of the input port. In the same way, service curves are used to represent the minimal data processing activity of the components. Typical arrival and service curves are shown in figure 4.

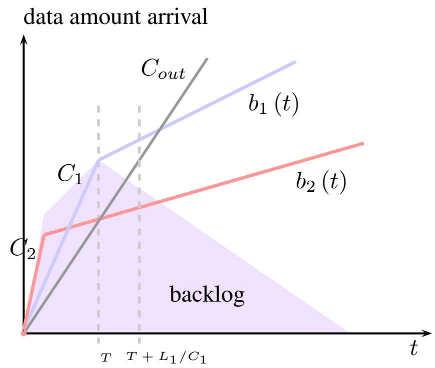

Figure 4. Arrival ($b_1(t)$, $b_2(t)$) and service ($C_{out}t$) curves and backlog evolution inside the two-input FIFO multiplexer

The approach used in analysing the upper bound delay for crossing a two-inputs multiplexer is shown next as introduced in the work presented in [5]. The approach is based on the evolution of a specific parameter, the backlog. The backlog is the number of bits waiting in the component, and it is a measure of congestion over the component. For the arrival curves in figure 4, the upper bound backlog occurs at time t where the following line is a maximum:

$$b_1(t) + b_2\left(t + \frac{L}{C_2}\right) - C_{out}t \tag{14}$$

where b_1 and b_2 are the arrival curves of stream 1 and 2 at time t, L is the maximum length of the frames, C_2 is the capacity of the input port 2, and C_{out} is the capacity of the output link.

When the upper bound backlog over the component is known, the upper bound delay over the component is then obtained by dividing the maximum backlog value by the capacity of the output link of the multiplexer.

In a FIFO m-inputs multiplexer, the delay for any incoming bit from the stream i is upper-bounded by:

$$\overline{D_{mux,i}} = \frac{1}{C_{out}} \min_k \overline{B_{mux,k}} \tag{15}$$

where $\overline{B_{mux,k}}$ is an upper-bound of the backlog in the bursty periods u_k, such that $1 \leq k \leq m$.

For $k = i$, the bursty period is defined by $u_i = \frac{\sigma_i}{C_i - \rho_i}$ and the backlog is upper-bounded by :

$$\overline{B_{mux,i}} = \sum_{z=1, z\neq i}^{m} \left(\sigma_z + \rho_z \left(u_i + \frac{L_z}{C_z}\right)\right) + u_i\left(C_i - C_{out}\right)$$

where σ_i is the burstiness of the stream i, ρ_i is the average rate of arrival of the data of stream i, L_i is the maximum length of the frames of stream i, and C_i is the capacity of the import port i. For $k \neq i$ such that $1 \leq k \leq m$, we have $u_k = \frac{\sigma_k}{(C_k - \rho_k) - L_k/C_k}$ and

$$\overline{B_{mux,i}} = \sum_{z=1, z\neq i}^{m} \left(\sigma_z + \rho_z \left(u_k + \frac{L_z}{C_z}\right)\right) + u_k\left(C_k - C_{out}\right) - \rho_i \frac{L_i}{C_i} + L_k$$

For the FIFO queue the delay of any byte is upper-bounded by:

$$\overline{D_{queue}} = \frac{1}{C_{out}} \frac{C_{in} - C_{out}}{C_{in} - \rho_{in}} \sigma_{in} \quad (16)$$

For the demultiplexer it is assumed that the time required to route the output port is relatively negligible compared to the other delays, i.e. the demultiplexer does not generate delays.

4.3. Maximum end-to-end delays for crossing switched Ethernet network

Computation of the upper bound end-to-end delays requires that special attention is paid to the input parameters of previous equations. The maximum delay value $\overline{D}$ depends on the leaky bucket parameters: the maximum amount of traffic σ that can arrive in a burst, and the upper bound of the average rate of the traffic flow ρ. In order to calculate the maximum delay over the network, it is hence necessary that the envelope (σ, ρ) is known at every point in the network. However, as shown in Figure 5, only the initial arrival curve values (σ^0, ρ^0) are usually known, and the values for other arrival curves have to be determined.

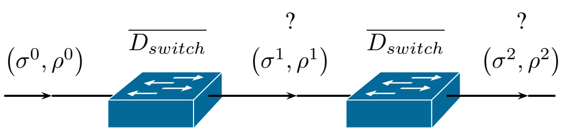

Figure 5. Burstiness along a switched Ethernet network

To calculate all the arrival curve values the following equations can be used:

$$\sigma_{out} = \sigma_{in} + \rho_{in}\overline{D}, \quad \rho_{out} = \rho_{in} \quad (17)$$

For example, for the arrival curve (σ^1, ρ^1) in figure 5 the envelope after the first switch is:

$$(\sigma^1, \rho^1) = (\sigma^0 + \rho^0 \overline{D_{switch}}, \rho^0)$$

The last part of the method used to obtain the upper-bounded delay estimate is the resolution of the burstiness characteristic of each flow at each point in the network. First, the burstiness values are determined by solving the equation system:

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & b_{n2} & \dots & a_{nn} \end{bmatrix} * \begin{bmatrix} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix} \quad (18)$$

where a_{ij} and b_i are the coefficients determined at each iteration of the equation (17). After solving the above equation, the upper bound end-to-end delays are obtained from

$$\overline{D_i} = \frac{\sigma_i^h - \sigma_i^0}{\rho_i} \quad (19)$$

where h is the number of crossed switches. For a complete discussion about the algorithm, interested readers may refer to [5].

4.4. Illustration

In order to illustrate the method presented in the previous paragraph, we considered the architecture presented in the Figure 6.

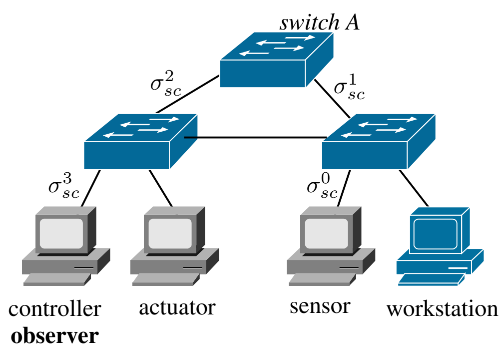

Figure 6. A redundant switched architecture

The network shown in the Figure 6 interconnects the controller, actuator and the sensor together by using a redundant switched Ethernet architecture, such that if a link between two switches breakdowns, the network will be able to carry on the communications. The architecture is here also shared with other applications than the control of the system, such that a workstation is also linked to the network.

Traffic arrivals are modelled as follows. Periodical exchanges from the sensor to the controller/observer are constrained by the arrival curve $b_{sc}(t) = \sigma_{sc} + \rho_{sc}t$ and exchanges from the controller to the actuator by $b_{ca}(t) =$

$\sigma_{ca}+\rho_{ca}t$. In the same time, the workstation sends frames to the controller. Traffic is here constrained by the arrival curve $b_w(t) = \sigma_w + \rho_w t$ with $\sigma_w \gg \sigma_{sc}$.

Consider now the delay supported by the frames sent by the sensor. Delays depend on the network topology, and consequently the communication path. In switched Ethernet network, the Spanning Tree Protocol is used to define an active topology in which the loop are eliminated. Firstly, it is assumed that a hierarchical active topology is defined, such that the measures will pass through the switch A.

The determination of an upper-bound $\overline{\tau_{sc}}$ consist of writing the equation system presented in equation (18). For that, it is necessary to write for each flows, the expression of the output burstiness for each switch and for each switch basic components defined in the figure 3. Formula are obtained according to the equations (17), (15) and (16). Hence the upper bound is given by the following formula:

$$\overline{\tau_{sc}} = \frac{\sigma_{sc}^3 - \sigma_{sc}^0}{\rho_{sc}}$$

The principle will be the same for the delay supported by the control frames.

To note also that the proposed method enables to take into account *network faults*. Indeed, if the case of a link failure between two switches, the Spanning Tree Protocol will defined a new active topology, and new communications paths. In the STP algorithm, the root switch is automatically determined by the protocol according switches parameters. Actually, the link failure detection follows an active probing process, such that the detection time is a compromise with the network load. By applying once again the previous analysis, a new upper-bound $\overline{\tau_{sc}}$ could be determined. This will be interesting in order to control the adaptation of the FDI algorithms to the network evolution. The global FDI scheme for the NCS is hence presented at figure 7.

As shown at figure 7, a new block entitled *Network Calculus* is added in order to pick up the network performance and provide to the FDI decision making process network information required to take into account the network influence.

5. Conclusion

This paper presents a residual evaluation strategy for fault diagnosis of Networked Control Systems. The interval for the admissible delays is computed by using network calculus theory and is used to determine the threshold. Based on this approach, the decision making is adjusted according to some network characteristics such as the traffic load or the topology. In case of unexpected changes in the network architecture (component breakdown), the network behavior is modified and transmission delays may vary. In this case, the proposed method can maintain a good level of performance of the FDI procedure.

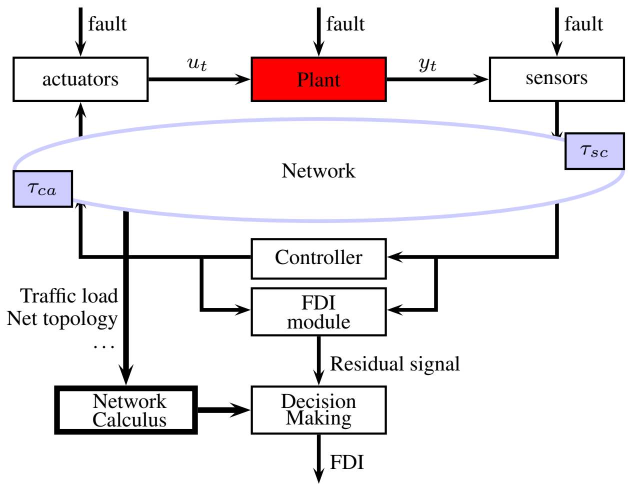

Figure 7. Global FDI scheme with adaptive threshold

Acknowledgment

This work is supported by the Safe NeCS project funded by the ANR (Agence Nationale de la Recherche, France) under grant ANR-05-SSIA-0015-03.

References

[1] R. Beard. *Failure accommodation in linear systems through self-re-organisation*. PhD thesis, MA, 1971.

[2] J. Chen and R. Patton. *Robust model-based fault diagnosis for dynamic systems*. Kluwer academic publishers, 1999.

[3] R. Cruz. A calculus for network delay, part i: network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–141, Jan. 1991.

[4] P. Frank and S. Ding. Current development in the theory of fdi. In *Proc. IFAC Safeprocess*, pages 16–27, 2000.

[5] J.-P. Georges, T. Divoux, and E. Rondeau. Confronting the performances of a switched ethernet network with industrial constraints by using the Network Calculus. *International Journal of Communication Systems (IJCS)*, 18(9):877–903, Nov. 2005.

[6] S.-S. Hu and Q.-X. Zhu. Stochastic optimal control and analysis of stability of networked control systems with long delay. *Automatica*, 39:1877–1884, 2003.

[7] J.-Y. Keller and D. Sauter. Some robust fault isolation filter for discrete time systems. In *4th IFAC SafeProcess Symposium*, pages 385–394, 2000.

[8] J.-Y. Le Boudec and P. Thiran. *Network calculus, a theory of deterministic queueing systems for the Internet*. Lecture Notes in Computer Science. Springer Verlag, 2001.

[9] S. Li, L. Yu, Z. Wang, and Y. Sun. Lmi approach to guaranteed cost control for networked control systems. *Developments in Chemical Engineering and Mineral Processing: Special Issue on Advanced Control and Real-Time Systems*, 13:351–361, 2005.

[10] Q. Ling and M. Lemmon. Robust performance of soft real-time networked control systems with data dropouts. In

Proc. Of the IEEE Conference on Decision and Control, 2002.

[11] Q. Ling and M. Lemmon. Soft real-time scheduling of networked systems with dropouts governed by a markov chain. In *Proc. Of the American Conference on Control*, 2003.

[12] B. Liu and S. J. Fault isolation filter design for linear time-invariant systems. *IEEE Transactions on Automatic Control,*, 42:704–707, 1997.

[13] M. Massoumnia. A geometric approach to the synthesis of failure detection filters. *IEEE Transactions on Automatic Control*, 31:839–846, 1986.

[14] A. Pasztor and D. Veitch. A precision infrastructure for active probing. In *Workshop on Passive and Active Measurements (PAM'01)*, Amsterdam, Apr. 2001.

[15] R. Patton, P. Frank, and R. Clark, editors. *Fault diagnosis in dynamic system*, International series in systems and control engineering. Prentice Hall, 1989.

[16] V. Puig, J. Quevedo, T. Escobet, and S. De las Heras. Passive robust fault detection approaches using interval models. In *15th Triennial World Congress of the International Federation of Automatic Control*, 2002.

[17] J. White and J. Speyer. Detection filter design: spectral theory and algorithms. *IEEE Trans. on Automatic Control*, 32:595–603, 1987.

[18] W. Zhang, S. Branicky, and S. Phillips. Stability of networked control systems. *IEEE Control Systems magazine*, 21:84–89, Feb. 2001.