HAL
open science

A real time router fault accommodation

Tushar Jain, Joseph Julien Yamé, Dominique Sauter

To cite this version:

Tushar Jain, Joseph Julien Yamé, Dominique Sauter. A real time router fault accommodation. Conference on Control and Fault-Tolerant Systems, SysTol'10, Oct 2010, Nice, France. pp.899-903. hal-00547937

HAL Id: hal-00547937

https://hal.science/hal-00547937v1

Submitted on 17 Dec 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

2010 Conference on Control and Fault Tolerant Systems
Nice, France, October 6-8, 2010

A Real Time Router Fault Accommodation

Tushar Jain, Joseph J. Yamé, Dominique Sauter

Abstract— **In this paper, a fault tolerance theory for queue based congestion control is proposed in the behavioral system theoretical framework. The router dynamics changes instantaneously when the fault occurs causing the undesirable instability in the queue length. To improve the dynamics of TCP/AQM networks, suitable controllers (stable and optimal) are synthesized using the concept of canonical controller in real time. This makes the network to maintain its stability. The main feature of this fault accommodation technique is that it solely relies on the input-output trajectories generated by the actual plant model (without having its exact knowledge) and on the performance specifications given by the desired behavior. The *least restrictive controller* satisfying the desired behavior is chosen based upon the generated trajectories.**

I. INTRODUCTION

RECENT research focuses the attention to achieve fault tolerance control (FTC) without employing the fault diagnosis module [1], [2]. In basic methodology, the complete FTC is composed of two parts: Fault Detection- Identification (FDI) and Fault Accommodation (FA). These two modules help to achieve a successful fault tolerant system. In fact, a proper exchange of information is required between the two which make the integrated system difficult to realize. When a fault occurs, FDI becomes active and FA waits for the information about fault behavior which induces delay referred to as *fault detection delay*. After receiving this knowledge of fault, FA module rectifies it and FDI is adapted to new scenario to get rid of false detection. Though these modules work parallel in independent path or in different time axis but require subtle information among each other. By information, we mean the fault occurrence time, fault behavior, its severity, etc [3]. Researches on robustness of both modules are in practice since last many years. Various techniques are evolved to achieve FTC [4] but in some cases the knowledge of fault is pre-assumed [5] considering only the issues regarding the impact of fault detection and correction delays on the system behavior. An important requirement for FDI system used in FTC is that the faults should be estimated with accuracy in real time. Second, the knowledge of plant must be available to carry out FDI.

Looking out these shortcomings, the concept of behavioral system theory in perspective of FTC is proposed in [6] without determining the fault behavior and in [7], which requires the information about fault. This work is the extension in terms of synthesizing the controllers using the concept of canonical controllers [8] with respect to the desired behavior. In [6], the behavioral approach for FTC is proposed in conjunction with the unfalsified control theory [9]. The advantage in employing behavioral concept is that it works on the trajectories generated by the system in real time. This helps in eliminating the need of exact model of plant and hence the FDI. The main point which catches the attention in [6] is the controller synthesis under the unfalsified framework when the effect of occurring fault is not taken into account beforehand. It is obvious that a synthesis of controller which achieves the given specification utilizing the dynamics of plant is one of the central issues of control system theory. This problem is handled in this work inspired from the approach of achievable behavior followed in [8] and the references there in. In [10], the central role of synthesizing the controller is played by the set of trajectories along which the dynamics of a system evolves. Moreover, a controller yielding the given specification is regarded as a canonical controller.

A natural way of coping with the fault accommodation problem is to synthesize the control parameters based upon the system trajectories when the fault occurs. This approach has been applied to an internet router for the congestion control. On the occurrence of fault the network characteristics changes instantaneously provoking congestion at different level of communication. The problem of internet congestion is tackled in [11] from the linear control theoretic point of view and from non-linear control in [12]. Recently, few work is done on internet congestion as a FTC problem [13], [14]. From these backgrounds, this paper provides a synthesis of linear canonical controller for a given desired behavior without having the knowledge of exact mathematical model of the system.

[Notations]: $\mathbb{R}$ is the set of real numbers. $\mathbb{R}^{p\times q}[\xi]$ is the set of polynomial matrices with real coefficients of size $p \times q$, where ξ is an operator. $\mathfrak{L}^q$ denotes the class of LTI differential systems with q variables. Let c denotes the control variables and w denotes to-be-controlled variables. At any given time, these variables take their value in signal space $\mathbb{C}$ and $\mathbb{W}$ respectively. Let $\mathbf{W}$ and $\mathbf{C}$ denote the set of all trajectories of the variable w and c.

II. ACHIEVABLE BEHAVIOR AND CANONICAL CONTROLLER

Here the brief overview of achievable (or implementable) behavior using partial interconnection and canonical controllers is discussed [8],[15]. For a linear time invariant differential system, the full plant behavior, $\mathcal{P}_f \in \mathfrak{L}^{q+k}$ be the full behavior comprising a set of solutions (w, c) such that

Manuscript received May 28, 2010.

T. Jain, D. Sauter and J.J. Yamé are with Centre de Recherche en Automatique de Nancy-CRAN-UMR 7039, Nancy Universite, CNRS-INPL-UHP, France tushar.jain@cran.uhp-nancy.fr

$$R\left(\frac{d}{dt}\right)w(t) = M\left(\frac{d}{dt}\right)c(t) \qquad w \in \mathbb{R}^q, c \in \mathbb{R}^k \quad (1)$$

where $R(\xi) \in \mathbb{R}^{l\times q}[\xi]$ and $M(\xi) \in \mathbb{R}^{l\times k}[\xi]$. Here c denotes the variables which are accessible to the controller, and w denotes the variables whose behavior we intend to restrict as shown in Fig.1. This restriction is imposed by the interconnection of (1) with another subsystem having behavior, $\mathcal{C} \in \mathfrak{L}^k$

$$H\left(\frac{d}{dt}\right)c(t) = 0 \quad (2)$$

where $H(\xi) \in \mathbb{R}^{h\times k}[\xi]$ is the controller. The interconnection of subsystems (1) and (2) gives the controlled subsystem which are shared by c variables as

$$\mathcal{P}_f \parallel_c \mathcal{C} \quad = \quad \{(w,c) \mid (w,c) \in \mathcal{P}_f \text{ and } c \in \mathcal{C}\} \quad (3)$$

It is assumed that the manifest variables (to-be-controlled) are observable from the shared auxiliary variables (control variables). Eliminating c from the full controlled behavior using the elimination theorem [10], the restriction $(\mathcal{P}_f \| _c \mathcal{C})_w$ on the behavior of the manifest variable w, defined by,

$$(\mathcal{P}_f \parallel_c \mathcal{C})_w = \{w \in \mathbf{W} \mid \exists c \in \mathcal{C} \text{ such that } (w,c) \in \mathcal{P}_f\} \quad (4)$$

Therefore, for given $\mathcal{P}_f \in \mathfrak{L}^{q+k}$ it is required to find $\mathcal{C} \in \mathfrak{L}^k$ which implements $\mathcal{K} \in \mathfrak{L}^q$ through c, where $\mathcal{K} = (\mathcal{P}_f \parallel_c \mathcal{C})_w$. This is known as *implementability theorem*. It depends only on the projected full plant behavior $(\mathcal{P}_f)_w$ and on the behavior consisting of the plant trajectories with the interconnection variables put equal to zero. This behavior is denoted by $\mathcal{N}_w(\mathcal{P}_f)$, and is called the hidden behavior. It is defined as

$$\mathcal{N}_w(\mathcal{P}_f) = \{w \mid (w,0) \in \mathcal{P}_f\} \quad (5)$$

Theorem 1 [*cf. [10]*] Let $\mathcal{P}_f \in \mathfrak{L}^{q+k}$ be the full plant behavior. Then $\mathcal{K} \in \mathfrak{L}^q$ is implementable by a controller $\mathcal{C} \in \mathfrak{L}^k$ acting on the interconnection variable c if and only if

$$\mathcal{N}_w(\mathcal{P}_f) \subseteq \mathcal{K} \subseteq (\mathcal{P}_f)_w$$

Theorem 1. shows that $\mathcal{K}$ can be *any* behavior that is wedged in between the given behaviors $\mathcal{N}_w(\mathcal{P}_f)$ and $(\mathcal{P}_f)_w$. Here we define the stabilizability of the manifest variable as the desired behavior which states, there exists a $\mathcal{C} \in \mathfrak{L}^k$ such that the monic characteristic polynomial $r \in (\mathcal{P}_f \parallel_c \mathcal{C})_w$ is Hurwitz, assuming $(\mathcal{P}_f)_w$ is stabilizable. Hence the controller $\mathcal{C}$ should yield the controlled behavior $\mathcal{K} = \mathcal{S}$. We call $\mathcal{S}$ the *desired controlled behavior*. The controllers to be implemented can be named as *stabilizing controllers*.

A controller imposed the restrictions only on the control variables and it is propagated through the plant to the to-be-controlled variables. The designing of controllers which satisfies the desired behavior requires the set of differential equations in $\mathbb{R}^{q\times k}$. So here we will use the idea of canonical controller which incorporates the concept of 'internal model principle (IMP)'. The construction of canonical controller requires the use of a plant model that has the same behavior as the plant. The restriction on the control variables can be made by interconnecting the plant model to the desired behavior $\mathcal{S}$ using the to-be-controlled variables. This is how the canonical controller is constructed. Here we denote this plant model by $\mathcal{P}_{imp}$. The behavior of the canonical controller obtained using this construction is denoted as $\mathcal{C}_{can}^{imp}$.

$$\mathcal{C}_{can}^{imp} = \{c \in \mathbf{C} \mid \exists v \in \mathbf{W} \text{ such that } (v,c) \in \mathcal{P}_{imp} \text{ and } v \in \mathcal{S}\} \quad (6)$$

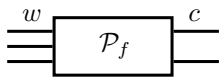

Fig. 1. Plant

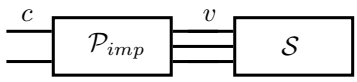

Fig. 2. Canonical Controller

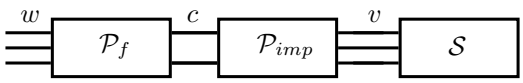

Fig. 3. Interconnected System (Plant and Controller)

Fig.2 and Fig.3 shows the construction of canonical controller and the full interconnected system respectively. Here we recommend to use the remarks 2.5 and 2.6 made in [8]. In brief, it states that if the plant model $\mathcal{P}_{imp}$ in the canonical controller $\mathcal{C}_{can}^{imp}$ differs from the actual model $\mathcal{P}_f$ then the desired behavior is still achievable i.e. $\mathcal{S} \subset \left(\mathcal{P}_f \parallel_c \mathcal{C}_{can}^{imp}\right)_w$ if $\mathcal{P}_{imp} \subset \mathcal{P}_f$, while $\left(\mathcal{P}_f \parallel_c \mathcal{C}_{can}^{imp}\right)_w \subset \mathcal{S}$ if $\mathcal{P}_f \subset \mathcal{P}_{imp}$. In [8], no distinction is made between $\mathcal{P}_f$ and $\mathcal{P}_{imp}$ and are considered the same. We use remark 2.5 to analyze the behavior of fault and remark 2.6 to *shrink* or *enlarge* $\mathcal{S}$ such that the interconnection $\left(\mathcal{P}_f \parallel_c \mathcal{C}_{can}^{imp}\right)_w$ approximates the desired behavior 'as good as possible'. Note that canonical controller is the class of all controllers which follows the desired behavior and here the desired behavior is the stability of the interconnected system as mentioned in previous paragraph.

III. AQM ROUTER AND FAULT ANALYSIS

A. Router Modeling and Desired Behavior

A window based nonlinear fluid-flow dynamic model for TCP networks is considered in this study. A detailed explanation of this model is presented in [11,17]. The coupled non-linear differential equations which reflects the dynamics of TCP accurately with the average TCP window size and the average queue length are given as:

$$\dot{w(t)} = \frac{1}{R(t)} - \frac{w(t)}{2}.\frac{w(t-R(t))}{R(t-R(t))}.p(t-R(t)) \quad (7a)$$

$$\dot{q(t)} = -C + \frac{N(t)}{R(t)}.w(t) \quad (7b)$$

$$R(t) = \frac{q\,(t)}{C} + T_p \quad (7c)$$

where w is the average TCP window size (in packets); q is the instantaneous queue length (in packets); T_p is the propagation delay (in seconds); R is the transmission round-trip time (RTT); C is the link capacity (in packets/sec); N is the number of TCP connections; and p is the packet dropping probability, which is the control input to decrease the sending rate and maintain the bottleneck queue length. All the above variables are supposed to be non-negative. Eq. 7a uses the additive increase and multiplicative decrease (AIMD) congestion control algorithm to evaluate the average window size during the TCP flow while (7b) is the dynamics of the queue length length accumulated as the transmission rate surpasses the link capacity.

Given the vector of network parameters $\eta = (N, C, d)$, a set of feasible operating points Ω_η is defined by

$$\Omega_\eta = \{(w_0, q_0, p_0) : w_0 \in (0, w_{max}), q_0 \in (0, q_{max}), \\ p_0 \in (0,1) \text{ and } \dot{w} = 0; \dot{q} = 0\} \quad (8)$$

The network parameters η are feasible if Ω_η is nonempty. A linearized model is for TCP congestion control, delays, and queues is expressed by

$$\begin{aligned} P(s) &= \frac{C^2/2N}{(s+2N/d^2C)(s+1/d)}.e^{-sd} \quad (9) \\ orP(s) &= P_0(s).e^{-Ls} \end{aligned}$$

where d is the RTT. Here $P(s)$ is the characterization of internal model discussed in previous section which only depends on η. The proportional-integral (PI) controller is suggested in [16] for the stability of queue length denoted by $G(s; k_p, k_i)$. So we fixed the structure of canonical controller and now its parameters are to be explored using the internal model and the desired behavior. We use algorithm [17, sec.11.5] to define the stability region because RouthHurwitz stability criterion is not applicable to (11). The region $\mathcal{S}_R$ contains the complete set of points (k_p, k_i) for which the desired behavior is implemented for all delays between 0 and d. As it is stated in [8] that if one controller can achieve the desired behavior then the same behavior can also be implemented by different controllers of the same class. The stability region $\mathcal{S}_R$ is expressed as $\mathcal{S}_R = \mathcal{S}_1 \setminus \mathcal{S}_L$, where

- $\mathcal{S}_1 = \mathcal{S}_0 \setminus \mathcal{S}_N$.
- $\mathcal{S}_0$ is the set of k_p and k_i values that stabilize the delay-free system $P_0(s)$.
- $\mathcal{S}_N$ is the set of k_p and k_i values such that $G(s; k_p, k_i)P_0(s)$ is an improper transfer function.
- $\mathcal{S}_L$ is the set of (k_p, k_i) values such that $G(s; k_p, k_i)P_0(s)$ has a minimal destabilizing delay that is less than or equal to d. Formally, $\mathcal{S}_L$ is

$$\mathcal{S}_L = \{(k_p, k_i) \notin \mathcal{S}_N : \exists L \in [0, d], \omega \in \mathbb{R} \\ \text{such that } G\,(j\omega; k_p, k_i)\, P_0\,(s)\,.e^{-jL\omega} = -1\}.$$

To compute $\mathcal{S}_R$, first define the projection of the stability region $\mathcal{S}_R$ on the line $k_p = \hat{k}_p$ as:

$$\mathcal{S}_{R,\hat{k}_p} = \{(k_p, k_i) \in \mathcal{S}_R : k_p = \hat{k}_p\},$$

so that the stability region can be calculated for each value of the proportional gain $\hat{k}_p$:

$$\mathcal{S}_R = \bigcup_{\hat{k}_p} \mathcal{S}_{R,\hat{k}_p}.$$

To compute $\mathcal{S}_{R,\hat{k}_p}$, define the projections

$$\begin{aligned} \mathcal{S}_{1,\hat{k}_p} &= \{(k_p, k_i) \in \mathcal{S}_1 : k_p = \hat{k}_p\}, \\ \mathcal{S}_{N,\hat{k}_p} &= \{(k_p, k_i) \in \mathcal{S}_N : k_p = \hat{k}_p\}, \\ \mathcal{S}_{L,\hat{k}_p} &= \{(k_p, k_i) \in \mathcal{S}_L : k_p = \hat{k}_p\}, \end{aligned}$$

Then, $\mathcal{S}_{R,\hat{k}_p} = \mathcal{S}_{1,\hat{k}_p} \setminus \mathcal{S}_{L,\hat{k}_p}$. The set $\mathcal{S}_{L,\hat{k}_p}$ can be further decomposed and computed as:

$$\mathcal{S}_{L,\hat{k}_p} = \mathcal{S}^+_{L,\hat{k}_p} \cup \mathcal{S}^-_{L,\hat{k}_p}.$$

where

$$\begin{aligned} \mathcal{S}^+_{L,\hat{k}_p} &= \left\{\left(\hat{k}_p, k_i\right) \notin \mathcal{S}_{N,\hat{k}_p} : \exists \omega \in \Omega^+, k_i = \sqrt{M\,(\omega)}\right\}, \\ \mathcal{S}^-_{L,\hat{k}_p} &= \left\{\left(\hat{k}_p, k_i\right) \notin \mathcal{S}_{N,\hat{k}_p} : \exists \omega \in \Omega^-, k_i = -\sqrt{M\,(\omega)}\right\}, \\ \Omega^+ &= \left\{\omega : \omega > 0, M\,(\omega) \geq 0, \right. \\ &\quad \left. \frac{\pi + \angle\left[\left(\sqrt{M\,(\omega)} + j\hat{k}_p\omega\right) R_0\,(j\omega)\right]}{\omega} \leq d\right\}, \\ \Omega^- &= \left\{\omega : \omega > 0, M\,(\omega) \geq 0, \right. \\ &\quad \left. \frac{\pi + \angle\left[\left(-\sqrt{M\,(\omega)} + j\hat{k}_p\omega\right) R_0\,(j\omega)\right]}{\omega} \leq d\right\}, \\ M\,(\omega) &= \frac{1}{\mid R_0\,(j\omega)\mid^2} - \hat{k}_p^2\omega^2, \\ R_0\,(s) &= \frac{P_0\,(s)}{s}. \end{aligned}$$

Note 2. Using this algorithm, the sensitivity of $\mathcal{S}_R$ to the set η is studied. On varying one parameter while keeping the other two constant, it is found that the region $\mathcal{S}_R$ increases on increasing the values of N and on decreasing the values of d, C.

B. Description of Fault Behavior

In basic control theory, faults are categorized into three types: (i) actuator faults, (ii) process parameter faults and (iii) sensor faults. Here we only study the process parameter faults where the parameters (given by set η) of the plant changes instantaneously resulting the queue length instability. Therefore to maintain its stability in the TCP/AQM network, this kind of fault should be accommodated in real time. Internet congestion control as FTC problem is discussed in [13] using Multi-RED (random early detection) controller configuration but the work lacks in determining the parameters of controller in real time. The fault scenarios and the parameters of controllers required to maintain the stability of queue length is known a priori. In [14], only the diagnosis module is studied for the possible occurrence of actuator faults and sensor faults. Here the fault tolerant control is achieved without using the explicit diagnosis module and the parameters of controllers are determined without the priori knowledge of faults.

Before going further, some necessary points needs to be highlighted. In an AQM router, when the number of connections or users get increased it leads to the proportional increase in RTT for fixed capacity [18]. Interestingly, N and RTT opposes each other from the point of defining the stability region [cf. Note 2.]. Now we define the internal model of the plat for which the set of controllers be designed. Here one minor assumption is made that the router with fixed capacity is always connected with some user and at any point of time the number of user can only increase to a maximum level that router can support. Referring (9), we see that RTT depends on T_p which is constant and $(q(t), C)$ is always positive.

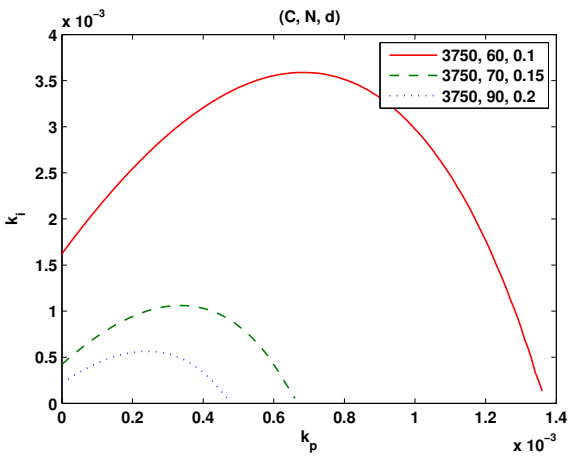

Fig. 4. Stability region (under the curve) for different sets of η

For example, we define $\eta_{imp} = (60, 3750, 0.1)$ for internal model as shown in Fig.4 and η_{cur} for the current set of parameters of plant. This gives the set of all controllers $\mathcal{C}_{can}^{imp}$. If we see the interconnection $\left(\mathcal{P}_f \parallel_c \mathcal{C}_{can}^{imp}\right)_w$, considering $\mathcal{P}_f = \mathcal{P}_{imp}$ or $\eta_{cur} = \eta_{imp}$ all set of controllers implement the desired behavior. Now suppose the parametric fault occurs and η_{cur} changes to $(70, 3750, 0.15)$. The stability region in Fig.4 for η_{cur} shows that the interconnected system $\left(\mathcal{P}_f \parallel_c \mathcal{C}_{can}^{imp}\right)_w \subset \mathcal{S}$. Therefore, the composed system satisfies atleast the 'specifications' given by $\mathcal{S}$.

IV. THE LEAST RESTRICTIVE CONTROLLER

In the previous section, the set of all possible controllers are determined by the interconnection of internal plant model and the desired behavior. For the optimal performance of TCP/AQM network there must exist a set of controllers that achieves it. Here by optimal performance we mean, the cost performance index J which is to be minimized. Now the problem is formulated as to find a set of controllers $\mathcal{C}_{res} \subset \mathcal{C}_{can}^{imp}$ such that $\mathcal{D} \subset \left(\mathcal{P}_f \parallel_c \mathcal{C}_{can}^{imp}\right)_w \subset \mathcal{S}$, where $\mathcal{D}$ is the optimal behavior and it denotes $\mathcal{D} = \left(\mathcal{P}_f \parallel_c \mathcal{C}_{res}\right)_w$. The set $\mathcal{D}$ is given by $\mathcal{D} = \{w \in \mathbf{W} : J(w) < \gamma\}$ where γ is a real bound. The cost function J is assumed to capture the control objective and examples of such functions are the integral square error (ISE), plant output variance etc.

Now, consider the fault tolerant control problem. Here we introduce the theory of switching control which uses the concept of *fictitious reference*. This reference signal is generated by the input-output data collected in the real time and by the controller inversion. This method is successfully studied in [6] for the application to hydraulic process. One of the objectionable question in [6] is that the possible faults are defined beforehand and the controller configuration with respect to each mode is designed. Now, in real time if any of the faulty mode occurs the correct controller is selected from the controller bank. This switching control can also be seen in [13] but no supervisor or reconfiguration mechanism is installed in the whole process. Here using the concept of canonical controller every unknown faulty mode is characterized in the region S_R. In real time operation only this region is to be explored to select the set of restricted controllers. Therefore, we can say S_R as the *region of explore*. This means that for any faulty behavior, there exists some set (k_p, k_i) which can accommodate the fault.

Assume the system is working satisfactorily with the interconnection of LRC and plant. After the occurrence of fault, the current controller is not able to achieve the optimal behavior. The set of canonical controllers might achieve the desired behavior but now the problem is to select the LRC that can accommodate the fault successfully and guarantees the optimal performance. To achieve this task, input-output data (u_{data}, y_{data}) is collected upto the current time. This data is used to generate the fictitious reference, $\tilde{r}$ for each controller without putting them into the feedback loop. Now based upon the cost function $J(w)$ where $w = (\tilde{r}, u_{data}, y_{data})$, only those controllers are considered to be the least restrictive which satisfies the optimal desired behavior.

V. CONCLUSION

In this theoretical study, a fault tolerant control is achieved based upon the to-be-controlled and control variables trajectories in the behavioral framework. The problem of internet congestion control is discussed from the viewpoint of fault

tolerance theory. Moreover, the other types of faults as mentioned in section III.B needs to be explored to achieve the complete FTC. As the TCP/AQM is SISO system after linearization, the modeling of loss in actuator efficiency would be the point of interest. The synthesis of controller in real time is the major issue in model free FTC approach which is handled successfully for the application to internet router.

REFERENCES

[1] J.J.Yamé, M. Kinnaert,"Performance based Switching for Fault tolerant control", 5th *IFAC SAFEPREOCESS*, 2003, pp. 555-560.

[2] A. Ingimundarson, R.S.S. Pena, "Using the Unfalsified Control Concept to achieve Fault Tolerance", *World Congress IFAC*, 2008, pp. 1236-1242.

[3] J. Jiang, Fault-Tolerant Control Systems- An Introductory Overview, *ACTA Automatica SINCA*, 31(1), 2005, pp. 161-174.

[4] H. Noura, D. Sauter, F. Hamelin, D. Theilliol, Fault-Tolerant Control in Dynamic Systems: Application to a Winding Machine, *IEEE Control System magazine*, February 2000, pp. 33-49.

[5] D. Theilliol, C. Join, Y. Zhang, Actuator Fault Tolerant Control Design Based on a Reconfigurable Reference Input, *Int. J. Appl. Math. Comput. Sci.*, 18(4), 2008, pp. 553-560

[6] J.J. Yamé, D. Sauter, "A Real-Time Model-Free Reconfiguration Mechanism for Fault-Tolerance: Application to a Hydraulic Process", 10th *Intl. Conf. on Control, Automation, Robotics and Vision*, 2008, pp. 91-96

[7] T. Jain. J.J. Yamé, D. Sauter, "A model based 2-DOF fault tolerant control strategy: Behavioral Theoretic Approach", 18th *Med. Conf*, 2010 (to appear).

[8] A.J. van der Schaft, Achievable behavior of general systems, *Systems and Control letters*,49(2), 2003, pp. 141-149.

[9] M.G. Safonov, T.C. Tsao, The Unfalsfied Control Concept and Learning, *IEEE Trans. Automatic Control*, 42(6), 1997, pp. 843-847.

[10] J.C.Willems, On interconnection, control and feedback, *IEEE Trans. Automatic Control*, 42(3), 1997, pp. 326-339.

[11] C.V. Hollot, V. Misra, D. Towsley, W. Gong, A Control Theoretic Analysis of RED, *Tech. Report UM-CS*-2000 − 041, 2000.

[12] F. Ren, C. Lin, B. Wei, A nonlinear control theoretic analysis to TCP-RED system, *Comp. Networks.*, 49, 2005, pp. 580-592.

[13] C. Aubrun, C. Join, K. Menighed, "Multi-RED Controller for Router Fault Accommodation", *IFAC SAFEPROCESS*, 2006, pp. 319-324.

[14] M. Fliess, C. Join, H. Mounier, An Introduction to Nonlinear Fault Diagnosis with an application to a Congested Internet Router, *Advances in Comm. Control Networks-LNCIS*, 2004, pp. 327-343.

[15] A.A. Julius, J.C. Willems, M.N. Bellur, H.L. Trentelman, "The canonical controllers and regular interconnection", *Systems and Control letters*, 54(8), 2005, pp. 787-797.

[16] C.V. Hollot, V. Misra, D. Towsley, W. Gong, Analysis and Design of Controllers for AQM Routers Supporting TCP Flows, *IEEE Trans. Automatic Control*, 47(6), 2002, pp. 945-959.

[17] G.J. Silva, A. Datta, S.P. Bhattacharyya, PID Controller for Time-Delay Systems, *Birkhuser*, 2005

[18] C.M. Kellet, R.N. Shorten, D.J. Leith, "Sizing Internet Router Buffers, Active Queue Management, and the Lure Problem", *Conf. in Decision and Control* , 2006, pp. 650-654