Unfalsified Adaptive Control for the Infrequently Switching Plant Based on Fading Memory Data

Huiyu Jin[1], Michael G. Safonov[2]
1. School of Aerospace Engineering, Xiamen University, Xiamen, 361005
E-mail: jinhy@xmu.edu.cn
2. Department of Electrical Engineering (Systems), University of Southern California, Los Angeles, CA 90007 USA
E-mail: msfaonov@usc.edu

Abstract: The problem of how to stabilize infrequently switching plant is investigated with unfalsified adaptive control approach. A fading memory data cost function is constructed to weaken the effects of timeworn data and its cost-detectablity is proved. A novel controller switching algorithm based on the ideas of self-falsification, stability overlay and resetting is designed. After each plant switching, the algorithm can detect the instability and switch to a stabilizing controller in finite switches and keeps the controller online until next plant switching. For the plant switching infrequently enough, theoretical analysis and simulations illustrate the algorithm can guarantee the cost function of the active controller uniformly bounded.

Key Words: Unfalsified adaptive control, Switching plant, Fading memory data, Cost-detectability, Algorithm

1 INTRODUCTION

Logic-based controller switching is used in adaptive control to overcome the limitations of continuous tuning [1]. Recently, Unfalsified Adaptive Control(UAC), a real-time data-driven switching control approach has gained special interests in theory and applicaitons. In UAC, we begin with a set of candidate controllers and an unknown plant. Typically, the candidate controllers have been selected so that each stabilizes at least one of several possible models of the plant. The problem of UAC control is to design a high-level unit called supervisor that switches among the candidate controllers to stabilize the unknown plant.This approach is called data-driven because all of supervisor's decisions are based on data measured on-line. When done correctly, UAC guarantees robust stability, irrespective of any parameter uncertainty and the model mismatch, provided only that the problem is feasible in the sense that at least one of the candidate controllers stabilizes the unknown plant. This is possible because closed-loop stability of the UAC adaptive system depends only on the existence of a feasible stabilizing controller in the candidate controller set, and not on the models and their parameters that may have been used for designing the candidate controllers.

There have been numerous papers investigating the features, applications and limitations of UAC [2–17]. In most of these, it is assumed that one candidate controller can stabilize the plant for all time. This is somewhat problematic because a basic motivation of adaptive control is the plant may be slowly or infrequently time-varying to such an extent such that no single candidate controller can stabilize it for all time. In such cases, even long after the UAC supervisor has converged to a feasible stabilizing controller, it will be again necessary to switch controllers when the plant changes are sufficiently large that the current stabilizing controller no longer stabilizing. To deal with such large but slow plant changes, the fading memory approach is introduced in UAC [18–20] and corresponding controller switching algorithms are studied [20–22].

In this paper, the problem of how to design the UAC supervisor to control an infrequently switching plant that cannot be stabilizing by any single one of the candidate controllers is investigated using a fading memory cost function. Its cost-detectability is proved, which is necessary in order for the UAC adaptive supervisor to be able to correctly identify and remove any destabilizing candidate controllers. To work with the non-monotone fading memory cost function, a novel controller switching algorithm is designed. It based on the idea of self-falsification and is a special case of Increasing Cost Level Algorithm(ICLA) in [13]. We will also adopt some of the ideas of the SO algorithm [23] and the resetting [20, 21]. The basic idea of the new algorithm is, if the current active controller has been unfalsified by the current cost level for a enough long time, the supervisor thinks it stabilizing the current plant and let it be on-line until it is falsified. Then, after it is indeed falsified, the supervisor thinks the plant has switched and gives every other candidate controller one opportunity to stabilize the plant at the current falsification cost level. If they all fail, the supervisor tries the controllers one by one again and increase the cost level to falsify after each controller switching. After finite switchings, the supervisor converges and the system remains stable while the cost level to falsify the current active controller increases to a greater value. After monitoring the active controller for sometime, the supervi-

This work is supported by National Nature Science Foundation (Grant No. 61374035) and Project of the Education Department of Fujian Province(JA15011).

sor thinks the controller stabilizing the plant, re-initialize the set of current active candidate controllers and waits for next time the current active controller is falsified because of plant changes. For the plant switching infrequently enough, the algorithm guarantees the cost function for the active controller is uniformly bounded for all time.

This paper is organized as follows. Section II formulates the problem. In Section III, the fading memory cost function is presented and the cost-detectability is proved. Section IV designs the controller switching algorithm and discusses on it. In Section V, simulations are carried to illustrate the performance of the algorithm.

2 PROBLEM FORMULATION

2.1 Notations

In this paper, $\mathbb{R}$ and $\mathbb{R}_+$ denote the set of real numbers and the set of non-negative real numbers respectively. We only consider the signals defined on $\mathbb{R}_+$ and whose Laplace transform are proper rational fraction. For a signal $y(\cdot)$, its truncation is defined as

$$y_\tau(t) = \begin{cases} y(t), & 0 \le t \le \tau \\ 0, & \text{otherwise} \end{cases}$$

while its $\mathcal{L}_2$-norm and $\mathcal{L}_\infty$-norm are denoted as $||y||$ and $||y||_\infty$ respectively. For a given $\lambda > 0$ and $t > 0$, we define the λ-exponential weighted norm of the truncated signal as

$$||y_t||_\lambda = \sqrt{\int_0^t (y(\tau))^2 e^{-2\lambda(t-\tau)} d\tau}.$$

2.2 The switching adaptive control problem

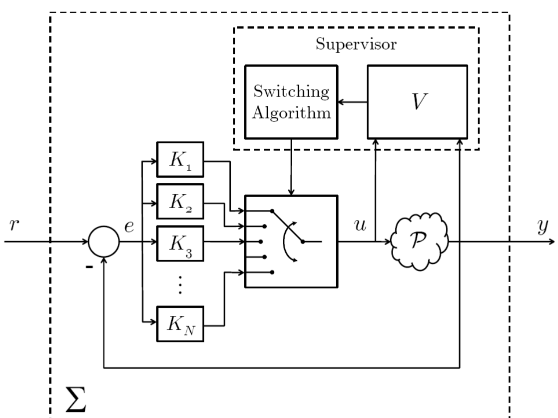

Figure 1: The switching adaptive control system.

In this paper, we consider the switching adaptive control system Σ shown in Fig. 1. In Fig. 1, there is a continuous time SISO plant $\mathcal{P}$ we know nothing about its parameters. A finite candidate controller set $\mathbb{K} = \{K_1, K_2, \cdots, K_N\}$ whose elements are all LTI are used to control plant $\mathcal{P}$. At each instant, one and only one of these candidate controllers is active. In time t, the active controller is denoted as $\hat{K}(t)$, while the system in Fig.1 is denoted as $\Sigma(\hat{K}, \mathcal{P})$. $\hat{K}(t)$ generates control signal $u(t)$ based on the reference signal $r(t)$ and output signal $y(t)$. A high-level unit called supervisor monitors the performance of $\hat{K}(t)$ and switches it if necessary. In the supervisor, there is a real-valued cost function $V(K, d_t, t)$ evaluates the performance of all candidate controllers based on the collected data $d(t) = \begin{bmatrix} u(t) & y(t) \end{bmatrix}'$. A controller switching algorithm decides whether to switch the active controller and which is the next active controller after switching. If the cost function and the controller switching algorithm do not use the parameters value of the plant $\mathcal{P}$, the supervisor is said *data-driven*. The problem on hand is to design a data-driven supervisor to guarantee the system stable and the concept stability will defined in next subsection.

2.3 stability and unfalsification

Definition 1(Stability) Consider a SISO system with input signal u and output signal y. System H is said to be *stable* if there exist constants α, β such that, for each input u and each $\tau \ge 0$, we have

$$||y_\tau||_\infty \le \beta ||u_\tau||_\infty + \alpha.$$

Otherwise, H is said to be *unstable*.$\diamond$

Definition 2(Unfalsification) Consider the system H in Definition 1. Suppose in an experiment we input the data u_1 and collect the output data y_1. We say the stability of H is *unfalsified* if there exist constants α, β such that, for each each $\tau \ge 0$, we have

$$||(y_1)_\tau||_\infty \le \beta ||(u_1)_\tau||_\infty + \alpha.$$

Otherwise, we say the stability of H is *falsified*.$\diamond$

Remark 1 Definition 1 defines stability totally based on input-output data. Its benefit is that the mathematical model of H is not used. Its disadvantage is, for a given instant t, we can not say if H is stable at t. On the other hand, it is well known that if system H in Definition 1 is lumped, causal, continuous time and LTI, then it has a rational proper fraction transfer function $H(s)$ and H is stable if and only if all poles of $H(s)$ have negative real parts. So, we can say if H is stable at time t based on its poles locations. $\diamond$

2.4 Assumptions

Here we give the assumptions in this paper.

A1 The reference signal r is bounded.

A2 The plant $\mathcal{P}$ in Fig. 1 may switch infrequently in a finite LTI set $\mathbb{P} = \{P_1, P_2, \ldots, P_M\}$ whose elements are all finite dimensional LTI plants. For each $P \in \mathbb{P}$, it contain no hidden unstable modes.

A3 For each $P_i \in \mathbb{P}$, there exists at least a candidate controller $K_j \in \mathbb{K}$ such that $\Sigma(P_i, K_j)$ is stable.

A4 All controllers $K \in \mathbb{K}$ are stably invertible; i.e., K^{-1} exists and is stable.

Remark 2 With the term of UAC, the assumption A3 is called *feasibility*. Similar to many papers on UAC, this paper does not discuss how to design a candidate controller set $\mathbb{K}$ to satisfy A3. It is a open problem and some results can be found in [24].$\diamond$

Remark 3 A4 can be relaxed by defining the fictitious reference signal based on the stable left Matrix Fraction Description of each candidate controller [7, 9]. In this paper, we use it for brief.$\diamond$

3 THE FADING MEMORY DATA COST FUNCTION AND ITS COST-DETECTABILTY

In this section, we designs the fictitious reference and fading memory cost function. Then the cost-detectability is proved.

3.1 fictitious reference signal

For a candidate controller, its fictitious reference is a hypothetical signal that would have reproduce exactly the measured data had the controller K_i been in the loop for the time period over which the data was collected. Same as [10], here the fictitious reference signal is defined for the case that all controllers in $\mathbb{K}$ assumed to be stably invertible.

Definition 3(Fictitious referencs) For a candidate controller K_i, its fictitious reference is generated by

$$\tilde{r}_i = y + K_i^{-1} u. \diamond$$

The meaning of fictitious reference can be shown with Fig.2. Suppose in Fig.2 the plant $\mathcal{P}$ is a duplicate with the plant $\mathcal{P}$ in Fig.1 and their initial value are same. Then, driven by the fictitious reference signal $\tilde{r}_k$, the system $\Sigma(K_k, \mathcal{P})$ will generate the same u and y with the system $\Sigma(\hat{K}, \mathcal{P})$.

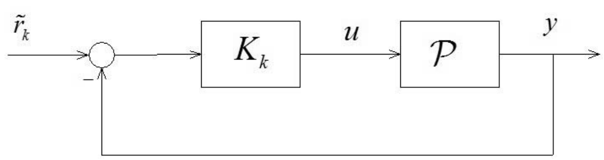

Figure 2: The meaning of fictitious reference signal.

3.2 the fading memory cost function and its cost-detectability

With fictitious reference, we construct the following fading memory cost function.

$$V(K_i, d_t, t) = \frac{\|(\tilde{r}_i - y)_t\|_\lambda^2 + \|u_t\|_\lambda^2}{\|(\tilde{r}_i)_t\|_\lambda^2 + C}, \qquad (1)$$

where C is a positive constant and λ is a positive small constant.

In UAC theory, cost-detectability is key concept which connects the unfalsification of stability to the boundedness of cost function.

Definition 4(Cost-detectability) Suppose in the switching adaptive control system Σ shown in Fig.1 we have $\hat{K}(t) \in \mathbb{K}$, $\forall t$. The cost function and controller set pair $(V, \mathbb{K})$ is *cost-detectable* if for every $\hat{K}(t)$ with at most finitely many switches, the following two statements are equivalent.

1) $V(K_f, d_t, t)$ is bounded as $t \to \infty$, where K_f is the final controller;

2) The stability of the system $\Sigma(\hat{K}, \mathcal{P})$ is unfalsified by the data pair (r, d_t).

The following is the the cost-detectablity theorem of this paper.

Theorem 1 Suppose assumption A1 holds, the plant $\mathcal{P}$ is LTI and the cost function is defined by (1). Then, the pair $(V, \mathbb{K})$ is cost-detectable.

Proof. Suppose for each $t \geq 0$, $|r(t)| \leq R_1$, where R_1 is a positive real number. Let $\tilde{r}^f$ be the fictitious reference of the final controller K^f. With the Lemma 1 in [10], there exists $a > 0$ such that $\|r - \tilde{r}^f\| \leq a < \infty$. That is, $\tilde{r}^f$ is also bounded. Suppose for each $t \geq 0$, $|\tilde{r}(t)| \leq R_2$.

If 2) holds, then there exist $\alpha_1, \alpha_2, \beta_1, \beta_2 > 0$ such that for each $t > 0$ we have

$$\|y_t\|_\infty \leq \beta_1 \|r_t\|_\infty + \alpha_1 \leq \beta_1 R_1 + \alpha_1,$$
$$\|u_t\|_\infty \leq \beta_2 \|r_t\|_\infty + \alpha_2 \leq \beta_2 R_1 + \alpha_2.$$

So, the cost function

$$\begin{aligned} V(K^f, d_t, t) &= \frac{\|(\tilde{r}^f - y)_t\|_\lambda^2 + \|u_t\|_\lambda^2}{\|\tilde{r}^f\|_\lambda^2 + C} \\ &\leq \frac{[(R_2 + \beta_1 R_1 + \alpha_1)^2 + (\beta_2 R_1 + \alpha_2)^2] \int_0^t e^{-2\lambda(t-\tau)} d\tau}{C} \\ &\leq \frac{(R_2 + \beta_1 R_1 + \alpha_1)^2 + (\beta_2 R_1 + \alpha_2)^2}{2\lambda C}, \end{aligned}$$

2) $\Rightarrow$ 1) is proved.

To prove 1) $\Rightarrow$ 2), we suppose u or y is unbounded. Because of the boundedness of $\tilde{r}^f$, there must be RHP pole or multiple imaginary axis pole in the Laplace Transform of u or y. Then, it is easy to verify $V(K_f, d_t, t)$ is also unbounded. Its converse negative proposition is, if $V(K_f, d_t, t)$ is bounded, both y and u are bounded. Suppose the bounds are α_1 and α_2 respectively. That is, for each $t > 0$, we have $|y(t)| \leq \alpha_1$ and $|u(t)| \leq \alpha_2$, which imply for each $t > 0$ that $\|y_t\|_\infty \leq \alpha_1$ and $\|u_t\|_\infty \leq \alpha_2$, and 1) $\Rightarrow$ 2) is proved. $\square$

4 THE CONTROLLER SWITCHING ALGORITHM

In this section, we present the controller switching algorithm, which is called Linear Increasing Cost Level Algorithm with Resetting(LICLA-R).

4.1 controller switching algorithm

Algorithm I LICIA-R

Constants

$\eta_0 > 0$: the initial value of the cost level to falsify the current active controller;

$\Delta\eta > 0$: the increment of the cost level to falsify the current active controller;

$dt > 0$: time increment;

$\emptyset$: empty set

$T > 0$: the time left to re-initialize the candidate controller set.

variables

t: time;

t_r: time left to reset;

$\mathbb{Q}$: the current candidate controller set,

$\hat{K}(t)$: active controller at time t.

γ: the current value of cost level to falsify the current active controller.

$KEEP$: Boolean variable to decide if γ increases after the next controller switching. If $KEEP = 1$, do not increase, otherwise increase.

Algorithm

1. Initialization: $\gamma \leftarrow \eta_0, t \leftarrow 0, \hat{K}(t) \leftarrow K_1, t_r = T, \mathbb{Q} \leftarrow \{K_1, K_2, \cdots, K_N\}, KEEP \leftarrow 1$;

2. $t \leftarrow t + dt$, collect data r, u, y, update $\tilde{v}_i$ and calculate $V(K_i, d_t, t)$ for all i;

3. IF
$$V(\hat{K}(t - dt), d_t, t) > \gamma \tag{2}$$
THEN

    $\mathbb{Q} \leftarrow \mathbb{Q} \setminus \hat{K}(t - dt)$;
    IF
      $\mathbb{Q} = \varnothing$
    THEN
      $\mathbb{Q} \leftarrow \{K_1, K_2, \cdots, K_N\}$;
      $KEEP \leftarrow 0$;
    ENDIF ;
    $\hat{K}(t) \leftarrow \arg\min_{K_i \in \mathbb{Q}} V(K_i(t), d_t, t)$;
    $t_r \leftarrow T$;
    IF
      $KEEP = 0$
    THEN
      $\gamma \leftarrow \gamma + \Delta\eta$;
    ENDIF ;

ELSE

    $\hat{K}(t) \leftarrow \hat{K}(t - dt)$;
    $t_r \leftarrow t_r - dt$;
    IF
      $t_r \leq 0$
    THEN
      $\mathbb{Q} \leftarrow \{K_1, K_2, \cdots, K_N\}$;
      $KEEP \leftarrow 1$;
    ENDIF ;

ENDIF

4. Go to 2. $\diamond$

Observation Suppose Assumption A1-A4 satisfied. $\lambda > 0$ satisfying, for each $K_i \in \mathbb{K}, P_j \in \mathbb{P}$, if $\Sigma(K_i, P_j)$ is stable, then all poles of $\Sigma(K_i, P_j)$ have real part less than $-\lambda$. The parameter dt is small enough and $T > 0$ is well chosen. Then, with cost function and Algorithm I, the system $\Sigma(\hat{K}, \mathcal{P})$ is stable and the active controller has a uniform bound for all time.

4.2 Discussion

Algorithm I is a special case of ICLA in [13] and its basic idea is self-falsification, which means to compare the current active controller's cost function with a given positive value.

Definition 5 (Self-falsification at a cost level) [10,13] Given the pair $(V, \mathbb{K})$ and a scalar $\gamma \in \mathbb{R}_+$, a controller $K_i \in \mathbb{K}$ is said to be falsified at time t *at cost level* γ by d_t if

$$V(K_i, d_t, t) > \gamma.$$

Otherwise, we say that controller K_i is unfalsified with respect to cost level γ by d_τ.$\diamond$

In Algorithm I, the supervisor collects data and verifies if (2) satisfied at each instant. If it is satisfied, then the current controller is self-falsified by γ and is switched, otherwise it will be active for next period. If the controller is switched, the supervisor moves it out of the current candidate controller set $\mathbb{Q}$ and select a new active controller in $\mathbb{Q}$ with

$$\hat{K}(t) \leftarrow \arg\min_{K_i \in \mathbb{Q}} V(K_i(t), d_t, t). \tag{3}$$

Then supervisor checks variable $KEEP$. If $KEEP = 1$, the supervisor keeps the value of γ , otherwise the supervisor increase the value of γ. If a controller has been active for T, it is looked upon stabilizing the current plant and the supervisor let the active controller be and re-initialize the candidate controller set.

The mechanism of Algorithm I to stabilize the plant can be merely sketched as follows, while the rigorous convergence theorem is not yet available. The supervisor looks the problem as a series of UAC problem to stabilizing LTI plant. Every problem begins with a plant switching. Because the supervisor does not know if the plant switches, it judges based on controller falsification, the candidate controller set $\mathbb{Q}$ and the boolean variable $KEEP$. If the active controller is falsified while all candidate controllers are in $\mathbb{Q}$ and $KEEP = 1$, the supervisor thinks the plant has just switched and destabilized the closed-loop. It gives every controller a chance to be unfalsified at the current cost level. If one of them is unfalsified, the switching of controller will stop without increment of cost level. With this approach, Algorithm I guarantees the cost level γ uniformly bounded while the plant switches endlessly. The reason is, once γ is great enough, it is impossible that all the other $N - 1$ candidate controllers are falsified because the controller which stabilizing the current plant can not be falsified. On the other hand, if all the other $N - 1$ candidates are falsified, the supervisor knows the current γ is not enough and let $KEEP = 0$ so that LICLA-R becomes LICLA, which increases γ after every controller switching. After finite controller switchings, the supervisor will find an unfalsified controller, keep it on-line and stabilizing the closed-loop, which is guaranteed by the convergence theorem of LICLA [13] and cost-detectability. Then, after monitoring the active controller for time T, the supervisor thinks it will not be falsified until next plant switching. It re-initialize $\mathbb{Q}$ and let $KEEP = 1$, which means this LTI plant UAC problem has been solved and the supervisor begins to wait for the next plant switching. It can be expected that after finite plant switching, the cost level γ will increase to and stops at such a value that after each plant switching, at most $N - 1$ controller switching will happen. So, the cost function of the active controller is uniformly bounded for all $t > 0$ and the system is stable.

The key points of Algorithm I are the following three. First, self-falsification is used. The self-falsification condition (2) is irrelative to other candidate controller's cost function value. So, if a stabilizing controller is switched on-line, it will not be switched off-line because another controller has a smaller cost function value. Self-falsification brings Algorithm I the benefit of less inserting the destabilizing controllers in the loop because simulations shows a off-line destabilizing may have smaller cost function value than the stabilizing on-line controller.

Second, the boolean variable $KEEP$, which helps the supervisor to judge if increases the cost level γ.
Third, to predict if the active controller stabilizing the plant until next plant switching based on how long it is unfalsified. If it is unfalsified for time T, the supervisor thinks it stabilizing the current plant. The rationale is, in adaptive control there are several time scales. The time scale of plant switching usually much greater than the time scale of adaptation process, while the time scale of adaptation process usually much greater than the time scale "associated with the signals in the system in the absence of and adaptation process" [2]. So, with the knowledge and experience we can choose a T, which is:
1) Long enough to guarantee if a destabilizing controller is switched on-line, it will be falsified definitely in T;
2) Short enough to guarantee the current candidate controller set re-initialization takes place much earlier than next plant switching.
Remark 4 In Algorithm I, the idea of giving falsification tests to every candidate controller with a same γ motivated by SO algorithm [23] and to re-initialize the current candidate controller adopts from resetting [20, 21].$\diamond$
Remark 5 In Algorithm I, the time T is selected based on knowledge and experience so it is not data-driven. How to find T from data will be studied next stage.$\diamond$

5 Simulations

In this section, numerical simulations are used to demonstrate the performance of Algorithm I.
Example 1 Consider the switching adaptive control system Σ shown in Fig. 1. Suppose that the plant $\mathcal{P}$ switches in

$$\mathbb{P} = \{P_1(s) = \frac{1}{s-1}, P_2(s) = \frac{-1}{s-1}\}$$

and the candidate controller set is

$$\mathbb{K} = \{K_1(s) = 2, K_2(s) = -1.5\}$$

Clearly, K_1 stabilizes P_1 while K_2 stabilizes P_2, but none of them can stabilize both P_1 and P_2. Suppose at $\mathcal{P}(0) = P_1$ and it switches every 200 s. The reference r is the unit step. We use cost function (1) to detect the stability changing of system $\Sigma(\hat{K}, \mathcal{P})$. Let $\lambda = 0.1$ and $C = 10$. Algorithm I is used with parameter $dt = 0.05, \eta_0 = 5, \Delta\eta = 2, T = 20$. Simulations are carried out with MATLAB 6.5 and the results are shown in Fig.3, Fig.4 and Fig.5.
Fig.3 illustrates the precess the switching plant is stabilized. For $t \in [0, 200)$ the plant stays in P_1 and is stabilized by K_1 and the cost level $\gamma = 5$. At $t = 200$, the plant switches to P_2, then soon K_1 is falsified at cost level $\gamma = 5$. After K_1 is switched off-line, K_2 is switched on-line, but it is also falsified at cost level $\gamma = 5$ later. Then the supervisor increase the γ to 7, and K_1 is falsified again. After that, the supervisor gives falsification test to K_2 with $\gamma = 9$. K_2 is unfalsified and on-line until the plant switches again. Then K_1 is switched on-line and faced the falsification test with $\gamma = 9$. It is also unfalsified. Then, after each switching of plant $\mathcal{P}$, there is only one controller switching will take place because the current destabilizing controller will be falsified with $\gamma = 9$ and the other controller will be unfalsified. The reason is, $\gamma = 9$ is a cost level both $\Sigma(K_1, P_1)$ and $\Sigma(K_2, P_2)$ can not reach.
To show the details of controller switching, the index of active controller nearby the plant switching are illustrated in Fig.4.
Fig.5 gives the the cost function values of the two controller. It is noticeable that, in most time, the destabilizing controller has the less cost function values. So, if use hysteresis algorithm with resetting, the destabilizing controller may be insert to the loop again and again.

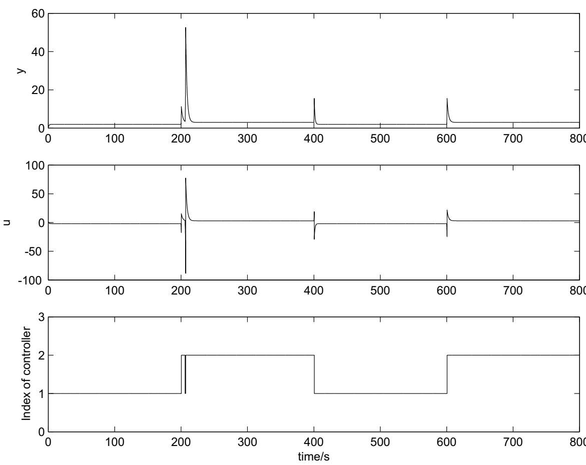

Figure 3: Stabilizing infrequent plant with Algorithm I.From top to bottom: output y, control u and the index of the active controller.

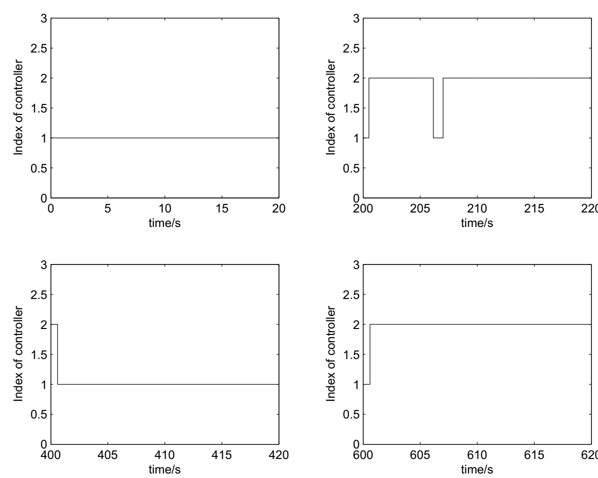

Figure 4: The controller switching after each plant switching.

6 CONCLUSIONS

In this paper, we study Unfalsified Adaptive Control for infrequently switching plants. A fading memory data cost function is constructed and a novel controller switching algorithm is designed to work with the cost function. Theoretical analysis and simulations show the algorithm can be

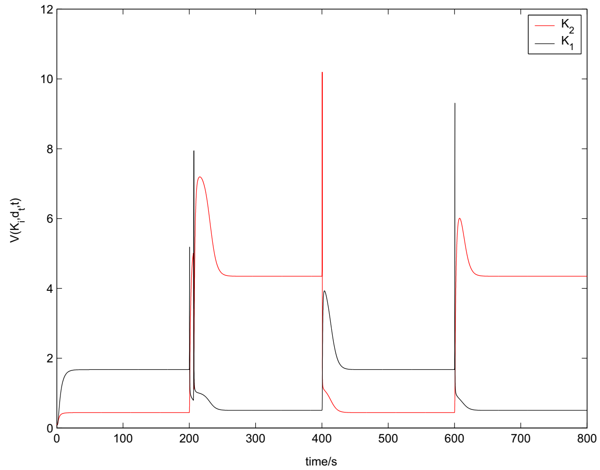

Figure 5: The cost function of the two controllers.

used to stabilize endlessly switching plants by guaranteeing the cost function of active controllers uniformly bounded. The algorithm is also helpful to prevent inserting destabilizing controller in the closed loop. To establish the convergence theorem of Algorithm I and extend the results to MIMO systems are the works of next stage research.

REFERENCES

[1] J. P. Hespanha, D. Liberzon, and A. S. Morse, "Overcoming the limitations of adaptive control by means of logic-based switching," *Systems & Control Letters*, vol. 49, no. 1, pp. 49–65, Jan. 2003.

[2] B. D. O. Anderson and A. Dehghani, "Challenges of adaptive control—past, permanent and future," *Annual Reviews in Control*, vol. 32, no. 2, pp. 123–135, Dec. 2008.

[3] S. Baldi, G. Battistelli, E. Mosca, and P. Tesi, "Multi-model unfalsified adaptive switching supervisory control," *Automatica*, vol. 46, no. 2, pp. 249–259, Feb. 2010.

[4] ——, "Multi-model unfalsified switching control of uncertain multivariable systems," *Int. J. Adaptive Control and Signal Processing*, vol. 26, no. 8, pp. 705–722, 2012.

[5] G. Battistelli, E. Mosca, M. G. Safonov, and P. Tesi, "Stability of unfalsified adaptive switching control in noisy environments," *IEEE Trans. on Automatic Control*, vol. 55, no. 10, pp. 2424–2429, Oct. 2010.

[6] M. W. Chang and M. G. Safonov, "Unfalsified adaptive control: The benefit of bandpass filters," in *Proceedings of AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, Hawaii, USA, Aug. 2008.

[7] A. Dehghani, B. D. O. Anderson, and A. Lanzon, "Unfalsified adaptive control: A new controller implementation and some remarks," in *Proceedings of the European Control Conference*, Kos, Grace, Jul. 2007.

[8] J. van Helvoort, B. de Jager, and M. Steinbuch, "Data-driven multivariable controller design using ellipsoidal unfalsified control," *Systems & Control Letters*, vol. 57, no. 9, pp. 759–762, Sep. 2008.

[9] C. Manuelli, S. G. Cheong, E. Mosca, and M. G. Safonov, "Stability of unfalsified adaptive control with non SCLI controllers and related performance under different prior knowledge," in *Proceedings of the European Control Conference*, Kos, Greece, Jul. 2007, pp. 702–708.

[10] R. Wang, A. Paul, M. Stefanovic, and M. G. Safonov, "Cost detectability and stability of adaptive control systems," *International Journal of Robust and Nonlinear Control*, vol. 17, no. 5-6, pp. 549–561, Mar-Apr 2007.

[11] M. Stefanovic and M. G. Safonov, "Safe adaptive switching control: Stability and convergence," *IEEE Transactions on Automatic Control*, vol. 53, no. 9, pp. 2012–2021, Oct. 2008.

[12] ——, *Safe Adaptive Control: Data-driven Stability Analysis and Robust Synthesis*. Berlin: Springer-Verlag, 2011.

[13] H. Jin and M. G. Safonov, "Unfalsified adaptive control: Controller switching algorithms for non-monotone cost functions," *Int. J. Adaptive Control and Signal Processing*, vol. 26, no. 8, p. 692C704, Augest 2012.

[14] A. Paul, M. Akar, M. G. Safonov, and U. Mitra, "Adaptive power control for wireless networks using multiple controllers and switching," *IEEE Transactions on Neural Networks*, vol. 16, no. 5, pp. 1212–1218, Sep. 2005.

[15] A. Ingimundarson and R. S. Sanchez-Pena, "Using the unfalsified control concept to achieve fault tolerance," in *Proceedings of the IFAC World Congress*, vol. 4, Seoul, Korea, Jul. 2008, pp. 1236–1242.

[16] H. B. Siahaan, H. Jin, and M. G. Safonov, "An adaptive pid switching controller for pressure regulation in drilling," in *Proceedings of IFAC Workshop on Automatic Control in Offshore Oil and Gas Production (ACOOG 2012)*, vol. 1, Trondheim, Norway, May-June 2012, pp. 90–94.

[17] T. Wonghong and S. Engell, "Automatic controller tuning via unfalsified control," *Journal of Process Control*, vol. 22, no. 10, pp. 2008–2025, December 2012.

[18] G. Battistelli, J. Hespanha, E. Mosca, and P. Tesi, "Unfalsified adaptive switching supervisory control of time varying systems," in *Proceedings of the IEEE Conference on Decision and Control and the Chinese Control Conference*, Shanghai, P.R. China, Dec. 2009, pp. 805–810.

[19] H. Jin, M. W. Chang, and M. G. Safonov, "Unfalsifying dominant pole locations using a fading memory cost function," in *Proceedings of the IFAC World Congress*, Milan, Italy, August 28 –September 2 2011, pp. 1291–1295.

[20] G. Battistelli, J. Hespanha, E. Mosca, and P. Tesi, "Model-free adaptive switching control of uncertain time-varying plants," in *Proceedings of the IFAC World Congress*, Milan, Italy, August 28 –September 2 2011, pp. 1273–1276.

[21] ——, "Dealing with plant variations in multi-model unfalsified switching control via adaptive memory selection," in *Proceedings of the IEEE Conference on Decision and Control and the European Control Conference*, Orland FL, USA, December 12-15 2011, pp. 366–371.

[22] H. Jin, M. W. Chang, and M. G. Safonov, "A fading memory data-driven algorithm for controller switching," in *Proceedings of the IEEE Conference on Decision and Control and the European Control Conference*, Orland FL, USA, December 12-15 2011, pp. 6097–6103.

[23] P. Rosa, J. S. Shamma, C. Silvestre, and M. Athans, "Stability overlay for adaptive control laws," *Automatica*, vol. 47, no. 5, pp. 1007–1014, 2011.

[24] B. D. O. Anderson, T. Brinsmead, F. De Brunye, J. Hespanha, D. Liberzon, and A. Morse, "Multiple model adaptive control. Part I: Finite controller covering," *Int. J. Adapt. Control Signal Process*, vol. 10, no. 11-12, pp. 909–929, Sep-Oct 2000.