Automatica 50 (2014) 874–882

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

# Adaptive memory in multi-model switching control of uncertain plants☆

Giorgio Battistelli[a], Edoardo Mosca[a], Pietro Tesi[b,1]

[a] Dipartimento di Ingegneria dell'Informazione (DINFO) - Università di Firenze, 50139 Firenze, Italy

[b] ITM, Faculty of Mathematics and Natural Sciences, University of Groningen, 9747 AG Groningen, The Netherlands

ARTICLE INFO

*Article history:*
Received 7 December 2011
Received in revised form
3 May 2013
Accepted 12 November 2013
Available online 30 December 2013

*Keywords:*
Control of uncertain plants
Adaptive control
Switching supervisory control

ABSTRACT

This paper describes some recent results in multi-model switching control. The scheme here considered embeds a finite family of pre-designed controllers and a high-level unit which selects, at each instant of time, the candidate controller to be placed in feedback to the uncertain plant. The study considers a switching strategy where controller selection is based on windowed cost functions. The key feature of the proposed strategy is that the window (the memory) is not kept constant, but, on the contrary, is adjusted on-line, on the grounds of measured data. The potential benefits of using an adaptive memory switching strategy are discussed and illustrated through a benchmark example.



## 1. Introduction

Adaptive switching control (ASC) has recently gained special attention as a viable approach for controlling plants subject to possibly time-varying uncertainties. Switching control algorithms usually embed a family of pre-designed candidate controllers and a supervisory unit. The task of the latter is to infer, on the grounds of plant input/output data, the potential behavior achievable by the use of each candidate controller, and select the one providing the most favorable potential behavior. This task is accomplished by associating to each candidate controller a cost function, which, at every time, quantifies the quality of the potential behavior yielded by the related candidate controller. Based on the values taken on by the cost functions, the supervisor decides whether the current controller is adequate, and, in the negative, which among the remaining controllers will replace the previous one. For an early overview of the topic, the reader is referred to Morse (1995).

Various ASC schemes have been proposed in the literature. They can be classified on the basis of the switching logic they adopt (possible alternatives being pre-routing, dwell-time and hysteresis switching), and on the basis of the adopted cost function. As for the latter, the main current approaches to ASC can be subdivided into two different groups, multi-model vs. model-free, depending on whether or not plant models are employed. Among the most significant multi-model approaches there are those investigated in Anderson et al. (2001), Fekri, Athans, and Pascoal (2007), Morse (1995), Narendra and Balakrishnan (1997) and Vu and Liberzon (2010). On the other hand, the most relevant approaches in the second group are those developed within the so-called *unfalsified control* framework (Battistelli, Mosca, Safonov, & Tesi, 2010; Safonov & Tsao, 1997; Stefanovic & Safonov, 2008; Stefanovic, Wang, Paul, & Safonov, 2007); see also Rosa, Shamma, Silvestre, and Athans (2011) for another relevant model-free switching control scheme.

An alternative approach was recently considered in Baldi, Battistelli, Mosca, and Tesi (2010, 2011), with the idea of combining the positive features of both multi-model architectures and unfalsified control. This approach, called MMUASC (Multi-Model Unfalsified Adaptive Switching Control), was shown to provide a reduction of learning transients when prior information on the plant uncertainty set is available (which is the distinguishing feature of model-based approaches) as well as stability guarantees under the only assumption that a stabilizing controller exists in the candidate controller set (which is the distinguishing feature of unfalsified control).

The limitation of the MMUASC scheme considered in Baldi et al. (2010, 2011) lies in the *persistent memory* feature of the cost function. Specifically, in the original MMUASC approach, the switching decision is achieved by comparing the maximum value taken on by each cost function over the whole observation interval

☆ The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Raul Ordóñez under the direction of Editor Miroslav Krstic.

*E-mail addresses:* giorgio.battistelli@unifi.it (G. Battistelli), mosca@dsi.unifi.it (E. Mosca), p.tesi@rug.nl (P. Tesi).

[1] Tel.: +31 050 3637649; fax: +39 055 4796363.

0005-1098/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.automatica.2013.12.002

up to the current time. As elaborated next in more detail, such a strategy is common to various ASC schemes (Angeli & Mosca, 2004; Pait & Kassab, 2001; Stefanovic et al., 2007) and is adopted to avoid instability caused by fast controller switching, a phenomenon difficult to prevent in the presence of large plant uncertainties if the controller selection relies solely on the current values of the cost functions.

The aim of this paper is then to develop a MMUASC scheme satisfying the *fading memory* condition.[2] It will be shown that the proposed scheme successfully addresses the critical question of how to safely forget old data, while retaining the desirable robustness features of the original MMUASC scheme. The result is achieved by means of a novel switching decision strategy. Instead of basing controller selection solely on the current values of the cost functions, we base controller selection on *windowed cost functions*. The idea is that the length of the window (the *memory*) is determined on-line, in an *adaptive* fashion: based on measured data, past records of the cost functions can be stored so as to facilitate 'learning', or discarded if they contain information irrelevant for stability. The findings of this paper build on recent results reported in Battistelli, Hespanha, Mosca, and Tesi (2013), where an adaptive memory strategy was proposed for model-free switching algorithms. Therefore, the present paper not only widens the theoretical ground of MMUASC but also indicates that the use of strategies based on adaptive memory can be of practical relevance for both model-free and model-based approaches to adaptive switching control.

The paper is organized as follows. Section 2 describes the problem of interest. Section 3 and 4 develop the proposed control scheme, and analyze the stability properties of the switched system. Section 5 discusses a simulation example by which the effectiveness of the proposed approach is exhibited. Section 6 concludes the paper by summing up results and open problems. All the proofs are given in the Appendix.

## 2. Background

Consider the problem of controlling a dynamical system whose input–output behavior can be described by the difference equation

$$y(t) = (1 - A(d))\, y(t) + B(d)\, u(t) + \nu(t) \tag{1}$$

where $A(d) = 1 + \sum_{k=1}^{n} a_k d^k$ and $B(d) = \sum_{k=1}^{n} b_k d^k$ are unknown polynomials in the unit backward shift operator $d$. Without loss of generality, it is supposed that (1) holds for any $t \in \mathbb{Z}_+ := \{0, 1, \ldots\}$ with the initialization $y(t) = 0, u(t) = 0$ for $t < 0$. Accordingly, the equation error $\nu$ accounts for exogenous disturbances, measurement noises, and possible non-zero plant initial conditions.

Denote by $\mathcal{P}$ the plant in (1) having transfer function $\mathcal{P}(d) = B(d)/A(d)$. We consider the problem of controlling $\mathcal{P}$ via a family of pre-designed controllers supervised by a high-level switching logic. For simplicity of exposition, the analysis will be carried out assuming the plant time-invariant. The possibility of successfully applying the proposed switching control scheme when the plant is subject to time variations will be discussed in Section 6 also by means of simulation results.

Consider a family $\mathscr{C} = \{\mathcal{C}_i,\ i \in \overleftarrow{N}\}$, $\overleftarrow{N} := \{1, 2, \ldots, N\}$ of $N$ linear time-invariant controllers having transfer functions $\mathcal{C}_i(d) := S_i(d)/R_i(d)$, $R_i(d) = 1 + \sum_{k=1}^{m_i} r_{ki} d^k$ and $S_i(d) = \sum_{k=0}^{m_i} s_{ki} d^k$, $m_i := \max\{\deg R_i, \deg S_i\}$. We denote by $\sigma(\cdot) : \mathbb{Z}_+ \mapsto \overleftarrow{N}$ the controller switching signal, *i.e.* the signal that identifies the candidate controller connected in feedback to the plant at each time. The control input $u$ is then computed via the difference equation

$$u(t) = (1 - R_{\sigma(t)}(d))\, u(t) + S_{\sigma(t)}(d)\, (r(t) - y(t)) \tag{2}$$

where $r$ is the reference signal.

**Remark 1** (*Computation of* (2)). Let $m := \max_{i \in \overleftarrow{N}} m_i$. The control action is initialized at time 0 from zero initial conditions, *i.e.* $u(k) = y(k) := 0$ for $k = -1, \ldots, -m$. At each time step $t$, we then solve (2) for the index corresponding to the switching signal $\sigma(t)$, by using $r(t) - y(t)$ and the most recent $m - 1$ samples of $u(k)$ and $r(k) - y(k)$. This (multi)-controller implements the so-called state-sharing in the sense of Morse (1995) since the samples $u(k)$ and $y(k)$ in (2) are not reinitialized upon switching. For alternative strategies employing controller-state resetting see, *e.g.*, Cheong and Safonov (2012) and Battistelli, Mari, Mosca, and Tesi (2013). □

### 2.1. Preliminaries

Let $(\mathcal{P}/\mathcal{C}_i)$ denote the feedback loop made up by $\mathcal{P}$ fed-back by $\mathcal{C}_i$. Also, let

$$w(t) := [r(t)\ \ \nu(t)]', \qquad z(t) := [u(t)\ \ y(t)]'. \tag{3}$$

Further, let the $l_\infty$-norm of a vector-valued sequence $s$ be defined as $\|s\|_\infty := \sup_{\tau \in \mathbb{Z}_+} \max_j |s^{(j)}(\tau)|$, where $s^{(j)}(\tau)$ denotes the $j$-th component of $s(\tau)$.

**Definition 1.** A sequence is said to be *bounded* if its $l_\infty$-norm is finite. The switched system (1) and (2) is said to be *stable* if, for all initial conditions, any bounded exogenous input sequence $w$ produces a bounded output sequence $z$. □

Let now

$$\alpha_i(d) := A(d)\, R_i(d) + B(d)\, S_i(d) \tag{4}$$

denote the characteristic polynomial of $(\mathcal{P}/\mathcal{C}_i)$. Also, let $\lambda_{(\mathcal{P}/\mathcal{C}_i)}$ denote the spectral radius of $(\mathcal{P}/\mathcal{C}_i)$, i.e., the inverse of the smallest among the absolute values of the roots of $\alpha_i(d)$, and define

$$\lambda_{\mathcal{P}} := \min_{i \in \overleftarrow{N}} \lambda_{(\mathcal{P}/\mathcal{C}_i)}. \tag{5}$$

**Assumption 1.** $\lambda_{\mathcal{P}} < 1$.

**Assumption 2.** The input sequence $w$ is bounded.

**Remark 2.** Assumption 1 is a natural requirement in order for the stabilization problem to be well-posed. It is indeed usually referred to as the *feasibility* requirement (Baldi et al., 2010; Stefanovic et al., 2007). □

## 3. Cost functions

In order to take switching decisions, the supervisor embodies a family $\mathscr{L} := \{\ell_i,\ i \in \overleftarrow{N}\}$ of cost functions, by which it infers the closed-loop behavior achievable by the use of each candidate controller. We briefly recall the main idea underlying the MMUASC approach as introduced in Baldi et al. (2010), and establish some results that will be used in the next section. All the equations which follow are understood to hold under zero initial conditions.

Consider a family $\mathscr{M} := \{\mathcal{M}_i,\ i \in \overleftarrow{N}\}$ of $N$ strictly causal linear time-invariant models $\mathcal{M}_i$ having transfer functions $\mathcal{M}_i(d) := B_i(d)/A_i(d)$, where $A_i(d)$ and $B_i(d)$ are polynomials in $d$. The set $\mathscr{M}$ forms, along with $\mathscr{C}$, a family $\mathscr{R} := \{(\mathcal{M}_i/\mathcal{C}_i),\ i \in \overleftarrow{N}\}$ of (internally stable) *nominal-loops*.

[2] Roughly (a precise definition will be given in Section 4), fading memory expresses the idea that the effect of past data vanishes with time. Fading memory is then essential for any control scheme to maintain the ability to respond to changes in the environment; see Jin, Chang, and Safonov (2011), Jin and Safonov (2012) for recent discussions on this point.