Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-13 (2016) 253–258

Bayesian Optimization-based Modular Indirect Adaptive Control for a Class of Nonlinear Systems

Mouhacine Benosman *

* *Mouhacine Benosman (m_benosman@ieee.org) is with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA.*

Amir-massoud Farahmand **

** *Amir-massoud Farahmand (farahmand@merl.com) is with MERL.*

Abstract: We study in this paper the problem of adaptive trajectory tracking control for a class of nonlinear systems with parametric uncertainties. We propose to use *a modular adaptive approach*, where we first design a robust nonlinear state feedback which renders the closed loop input-to-state stable (ISS). The input is considered to be the estimation error of the uncertain parameters, and the state is considered to be the closed-loop output tracking error. We augment this robust ISS controller with a model-free learning algorithm to estimate the model uncertainties. We implement this method with a Bayesian optimization-based method called Gaussian Process Upper Confidence Bound (GP-UCB). The combination of the ISS feedback and the learning algorithms gives a *learning-based modular indirect adaptive controller*. We test the efficiency of this approach on a two-link robot manipulator example, under noisy measurements conditions.

1. INTRODUCTION

Many adaptive methods have been proposed over the years for linear and nonlinear systems, e.g., Krstic et al. [1995]. In this work we focus on a specific type of adaptive control, namely, the indirect modular approach to adaptive nonlinear control, e.g., Krstic et al. [1995], Wang et al. [2006], Benosman and Atinc [2013], Atinc and Benosman [2013], Benosman [2014b,a], Xia and Benosman [2015], Lavretsky [2009], Haghi and Ariyur [2011]. In the direct approach, first a controller is designed by assuming that all the parameters are known (certainly equivalence principle), and then an identifier is used to estimate the unknown parameters online. The identifier might be independent of the designed controller, in which case the approach is called 'modular'. A modular approach has been proposed in Wang et al. [2006] for adaptive neural control of pure-feedback nonlinear systems, where the input-to-state stability (ISS) modularity of the controller-estimator is achieved and the closed-loop stability is guaranteed by the small-gain theorem, e.g., Sontag [1989].

In this work, we present a modular adaptive design which combines model-free learning methods and robust model-based nonlinear control to propose a learning-based modular indirect adaptive controller. Here, a model-free learning algorithm is used to estimate in closed-loop the uncertain parameters of the model. The main difference with the existing model-based indirect adaptive control methods, is the fact that we do not use the model to design the parameters estimation filters. Indeed, model-based indirect adaptive controllers are based on parameters' estimators designed using the model of the system, e.g., the X-swapping methods presented in Krstic et al. [1995]. Here, because we do not use the system dynamics to design the estimation filters we can deal with a more general class of uncertainties, e.g., nonlinear uncertainties can be estimated with the proposed approach, see Atinc and Benosman [2013] for some preliminary results. Furthermore, with the proposed approach we can estimate a vector of linearly dependent uncertainties, a case which cannot be solved using model-based filters, e.g., in Benosman and Atinc [2015] it is shown that the X-swapping model-based method fails to estimate a vector of linearly dependent parameters.

In this work, we implement the proposed approach with a Bayesian optimization-based method called GP-UCB. The latter solves the exploration-exploitation problem in the continuous armed bandit problem, which is a non-associative reinforcement learning (RL) setting.

We want to underline here that compared to 'pure' model-free controllers, e.g., pure RL algorithms, the proposed control has a different goal. The available model-free controllers are meant for output or state regulation. In the contrary, here we propose to use model-free learning to complement a model-based nonlinear control to estimate the unknown parameters of the model. Here the control goal, i.e., state or output trajectory tracking, is handled by the model-based controller. The learning algorithm is used to improve the tracking performance of the model-based controller. Once the learning algorithm has converged, one can carry on using the nonlinear model-based feedback controller alone, without the need of the learning algorithm. Moreover, we believe that this type of controller can converge faster to an optimal performance, comparatively to the pure model-free controller. The reason is that the model-free algorithms assume no knowledge about the system, and thus start the search for an optimal control signal from scratch. On the other hand, by 'partly' using a model-based controller we are taking advantage of the partial information given by the physics of the system.

A modular design merging model-based control and an extremum seeker has been proposed in Haghi and Ariyur [2011, 2013], Benosman and Atinc [2012, 2013], Atinc and Benosman [2013], Benosman [2014b,a], Xia and Benosman

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2016.07.960

[2015]. In Haghi and Ariyur [2011, 2013], extremum seeking is used to complement a model-based controller, under linearity of the model assumption in Haghi and Ariyur [2011], or under the assumption of linear parametrization of the control in terms of the uncertainties in Haghi and Ariyur [2013]. The modular design idea of using a model-based controller with ISS guarantee, complemented with an ES-based module can be found in Atinc and Benosman [2013], Benosman [2014b,a], Xia and Benosman [2015], where the ES was used to estimate the model parameters, and in Benosman and Atinc [2013], Benosman [2015] where feedback gains were tuned using ES algorithms. The work of this paper falls in this class of ISS-based modular indirect adaptive controllers. The difference with other ES-based adaptive controllers is that, due to the ISS modular design we can use any model-free learning algorithm to estimate the model uncertainties, not necessarily extremum seeking-based. To emphasize this we show here the performance of the controller when using a type of RL-based learning algorithm.

The rest of the paper is organized as follows. In Section 2, we formulate the problem. The nominal controller design are presented in Section 3. In Section 3.2, a robust controller is designed which guarantees ISS from the estimation error input to the tracking error state. In section 3.3, we introduce the RL GP-UCB algorithm as a model-free learning to complement the ISS controller. Section 4 is dedicated to an application example and the paper conclusion is given in Section 5.

Throughout the paper, we use $\|\cdot\|$ to denote the Euclidean norm; i.e., for a vector $x \in \mathbb{R}^n$, we have $\|x\| \triangleq \|x\|_2 = \sqrt{x^T x}$, where x^T denotes the transpose of the vector x. We denote by $\mathrm{Card}(S)$ the size of a finite set S. The Frobenius norm of a matrix $A \in \mathbb{R}^{m\times n}$, with elements a_{ij}, is defined as $\|A\|_F \triangleq \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}|a_{ij}|^2}$. Given $x \in \mathbb{R}^m$, the signum function is defined as $\mathrm{sign}(x) \triangleq [\mathrm{sign}(x_1),\ \mathrm{sign}(x_2),\ \cdots,\ \mathrm{sign}(x_m)]^T$, where sign(.) denotes the classical signum function.

2. PROBLEM FORMULATION

We consider here affine uncertain nonlinear systems of the form

$$\begin{array}{rcl}\dot{x} &=& f(x) + \Delta f(t,x) + g(x)u,\\ y &=& h(x),\end{array} \tag{1}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^p$, $y \in \mathbb{R}^m$ $(p \geq m)$, represent the state, the input and the controlled output vectors, respectively. $\Delta f(t,x)$ is a vector field representing additive model uncertainties. The vector fields f, Δf, columns of g and function h satisfy the following assumptions.

Assumption A1 The function $f : \mathbb{R}^n \to \mathbb{R}^n$ and the columns of $g : \mathbb{R}^n \to \mathbb{R}^p$ are $\mathbb{C}^\infty$ vector fields on a bounded set X of $\mathbb{R}^n$ and $h : \mathbb{R}^n \to \mathbb{R}^m$ is a $\mathbb{C}^\infty$ vector on X. The vector field $\Delta f(x)$ is $\mathbb{C}^1$ on X.

Assumption A2 System (1) has a well-defined (vector) relative degree $\{r_1,\ r_2,\ \cdots,\ r_m\}$ at each point $x^0 \in X$, and the system is linearizable, i.e., $\sum_{i=1}^m r_i = n$.

Assumption A3 The desired output trajectories y_{id} $(1 \leq i \leq m)$ are C^∞ functions of time, relating desired initial points $y_{id}(0)$ at $t = 0$ to desired final points $y_{id}(t_f)$ at $t = t_f$.

Our objective is to design a state feedback adaptive controller such that the output tracking error is uniformly bounded, whereas the tracking error upper-bound is function of the uncertain parameters estimation error, which can be decreased by the model-free learning. We stress here that the goal of learning algorithm is not stabilization but rather performance optimization, i.e., the learning improves the parameters' estimation error, which in turn improves the output tracking error. To achieve this control objective, we proceed as follows: First, we design a robust controller which can guarantee input-to-state stability (ISS) of the tracking error dynamics w.r.t the estimation errors input. Then, we combine this controller with a model-free learning algorithm to iteratively estimate the uncertain parameters, by optimizing online a desired learning cost function.

3. ADAPTIVE CONTROLLER DESIGN

3.1 Nominal Controller

Let us first consider the system under nominal conditions, i.e., when $\Delta f(t,x) = 0$. In this case, it is well know, e.g., Khalil [2002], that system (1) can be written as

$$y^{(r)}(t) \ = \ b(\xi(t)) + A(\xi(t))u(t), \tag{2}$$

where

$$\begin{array}{rcl} y^{(r)}(t) &=& [y_1^{(r_1)}(t),\ y_2^{(r_2)}(t),\ \cdots,\ y_m^{(r_m)}(t)]^T,\\ \xi(t) &=& [\xi^1(t),\ \cdots,\ \xi^m(t)]^T,\\ \xi^i(t) &=& [y_i(t),\ \cdots,\ y_i^{(r_i-1)}(t)], \quad 1 \leq i \leq m \end{array} \tag{3}$$

The functions $b(\xi)$, $A(\xi)$ can be written as functions of f, g and h, and $A(\xi)$ is non-singular in $\tilde{X}$, where $\tilde{X}$ is the image of the set of X by the diffeomorphism $x \to \xi$ between the states of system (1) and the linearized model (2). Now, to deal with the uncertain model, we first need to introduce one more assumption on system (1).

Assumption A4 The additive uncertainties $\Delta f(t,x)$ in (1) appear as additive uncertainties in the input-output linearized model (2)-(3) as follows:

$$y^{(r)}(t) \ = \ b(\xi(t)) + A(\xi(t))u(t) + \Delta b(t, \xi(t)), \tag{4}$$

where $\Delta b(t,\xi)$ is $\mathbb{C}^1$ w.r.t. the state vector $\xi \in \tilde{X}$.

It is well known that the nominal model (2) can be easily transformed into a linear input-output mapping. Indeed, we can first define a virtual input vector $v(t)$ as

$$v(t) \ = \ b(\xi(t)) + A(\xi(t))u(t). \tag{5}$$

Combining (2) and (5), we can obtain the following input-output mapping

$$y^{(r)}(t) \ = \ v(t). \tag{6}$$

Based on the linear system (6), it is straightforward to design a stabilizing controller for the nominal system (2) as

$$u_n = A^{-1}(\xi)\left[v_s(t,\xi) - b(\xi)\right], \tag{7}$$

where v_s is a $m\times 1$ vector and the i-th $(1 \leq i \leq m)$ element v_{si} is given by

$$v_{si} = y_{id}^{(r_i)} - K_{r_i}^i(y_i^{(r_i-1)} - y_{id}^{(r_i-1)}) - \cdots - K_1^i(y_i - y_{id}). \tag{8}$$

If we denote the tracking error as $e_i(t) \triangleq y_i(t) - y_{id}(t)$, we obtain the following tracking error dynamics

$$e_i^{(r_i)}(t) + K_{r_i}^i e^{(r_i-1)}(t) + \cdots + K_1^i e_i(t) = 0, \tag{9}$$

where $i \in \{1,\ 2,\ \cdots,\ m\}$. By properly selecting the gains K_j^i where $i \in \{1,\ 2,\ \cdots,\ m\}$ and $j \in \{1,\ 2,\ \cdots,\ r_i\}$, we can obtain global asymptotic stability of the tracking errors $e_i(t)$. To formalize this condition, we add the following assumption.