MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Towards Stability in Learning-based Control: A Bayesian Optimization-based Adaptive Controller

Farahmand, A.-M.; Benosman, M.

TR2017-075 June 2017

Abstract

We propose to merge together techniques from control theory and machine learning to design a stable learning-based controller for a class of nonlinear systems. We adopt a modular adaptive control design approach that has two components. The first is a model-based robust nonlinear state feedback, which guarantees stability during learning, by rendering the closed-loop system input-to-state stable (ISS). The input is considered to be the error in the estimation of the uncertain parameters of the dynamics, and the state is considered to be the closed-loop output tracking error. The second component is a data-driven Bayesian optimization method for estimating the uncertain parameters of the dynamics, and improving the overall performance of the closed-loop system. In particular, we suggest using Gaussian Process Upper Confidence Bound (GP-UCB) algorithm, which is a method for trading-off exploration-exploitation in continuous-armed bandits. GP-UCB searches the space of uncertain parameters and gradually finds the parameters that maximize the performance of the closed-loop system. These two systems together ensure that we have a stable learning-based control algorithm.

The Multi-disciplinary Conference on Reinforcement Learning and Decision Making

Towards Stability in Learning-based Control: A Bayesian Optimization-based Adaptive Controller

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

Amir-massoud Farahmand
Mitsubishi Electric Research Laboratories (MERL)
Cambridge, MA 02139, USA
farahmand@merl.com

Abstract

We propose to merge together techniques from control theory and machine learning to design a stable learning-based controller for a class of nonlinear systems. We adopt a modular adaptive control design approach that has two components. The first is a model-based robust nonlinear state feedback, which guarantees stability during learning, by rendering the closed-loop system input-to-state stable (ISS). The input is considered to be the error in the estimation of the uncertain parameters of the dynamics, and the state is considered to be the closed-loop output tracking error. The second component is a data-driven Bayesian optimization method for estimating the uncertain parameters of the dynamics, and improving the overall performance of the closed-loop system. In particular, we suggest using Gaussian Process Upper Confidence Bound (GP-UCB) algorithm, which is a method for trading-off exploration-exploitation in continuous-armed bandits. GP-UCB searches the space of uncertain parameters and gradually finds the parameters that maximize the performance of the closed-loop system. These two systems together ensure that we have a stable learning-based control algorithm.[1]

Keywords: Adaptive Control, Modular Control, Stability, Machine Learning, Bayesian Optimization, GP-UCB

[1]This paper is a summary of Benosman and Farahmand [2016]; Benosman et al. [2016].

1 Introduction

There are much common problems between the fields of control engineering and reinforcement learning, as is mainly practiced in computer science. A central problem in both is to make a sequence of decisions in order to ensure that a dynamical system behaves in a desired manner. This high-level similarity notwithstanding, there are noticeable differences between the goals and methodologies of these two fields. Without attempting to characterize the differences in this short paper, we focus only on one crucial difference between them, and that is the prominent importance of *stability* in control engineering. This is understandable as many applications of control engineering are safety-critical. Proving that a controller (i.e., policy) makes the dynamical system stable, especially when there are uncertainties about the dynamics, requires making certain assumptions about the system. The challenge is to make as few assumptions as possible while still being able to guarantee the stability. On the other hand, methods in machine learning, in general, and reinforcement learning, in particular, are often capable of providing good solutions with minimal assumptions about the underlying system. Their guarantees, if there is any, is often in the form of suboptimality of the solution, which does not necessarily translate into the stability of the resulting policy.

We would like to have a controller design methodology that brings the best of these worlds together. Ideally, such a methodology can work with a large class of problems and can incorporate the prior knowledge about the system (which is often available in many engineering problems). The designed controller must be guaranteed to be stable, so it can be used in safety-critical problems. Its performance should also gradually improve while interacting with the environment. In this paper, we introduce such a method, which merges together a model-based controller and a data-driven machine learning algorithm. The model-based controller is designed to ensure stability, while the data-driven module improves the performance of the closed-loop system over time. More specifically, we use tools from indirect modular adaptive nonlinear control design and the Bayesian optimization technique of Gaussian Process-Upper Confidence Bound (GP-UCB) [Srinivas et al., 2010].

We take the indirect modular approach to adaptive nonlinear control design, e.g., Krstic et al. [1995]; Wang et al. [2006]; Benosman and Atinc [2013]; Atinc and Benosman [2013]; Benosman [2014]. In the indirect approach, a controller is designed by assuming that all the parameters of the dynamical system are known (certainly equivalence principle) while an estimator is used to estimate the unknown parameters online. When the design of the estimator is independent of the designed controller, the approach is called "modular". The proposed modular adaptive control uses a robust model-based nonlinear controller to ensure the stability of the system—in the input-to-state stable (ISS) sense. The learning-based algorithm, GP-UCB in particular, estimates the uncertain parameters of the dynamical system in order to maximize the performance.

Our approach can be seen as solving a continuous-armed bandit problem in which each action corresponds to a set of parameters passed to a controller design mechanism. By defining the reward signal as the closed-loop performance of the controller for the selected action, we may use GP-UCB, a method for trading-off exploration-exploitation in the bandit setting, to orchestrate the selection of actions, and thus improving the closed-loop performance.

We would like to note that a modular design that combines model-based control and a model-free algorithm, in particular an extreme seeking-based (ES) algorithm, has been proposed before [Haghi and Ariyur, 2013; Benosman and Atinc, 2013; Atinc and Benosman, 2013; Benosman, 2014; Xia and Benosman, 2015]. The limitation of the ES-based algorithm is that it converges to a local minimum. GP-UCB, on the other hand, has a guarantee, in the form of a no-regret compared to the global optimum.

2 Problem Formulation

We consider a large class of affine uncertain nonlinear systems of the form

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

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. The vector field $\Delta f(t,x)$ represents the 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.
- 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., $\sum_{i=1}^m r_i = n$.
- 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 $t = t_f$.

The goal is to design a state feedback adaptive controller such that the output tracking error is uniformly bounded, with an upper bound that is a function of the estimation error of the uncertain parameters. Meanwhile, the estimation error is gradually being decreased by the model-free learning algorithm. Note that the goal of the 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 that guarantees input-to-state stability (ISS) of the tracking error dynamics w.r.t. the estimation errors input (Section 3). Then, we combine this controller with a model-free learning algorithm to iteratively estimate the uncertain parameters, by online optimizing a user-selected cost function (Section 4).

3 Controller Design

The controller design has two steps. First, we design a stabilizing controller under nominal conditions, i.e., when $\Delta f(t, x) = 0$. Then we design a robust controller that ensures that the tracking error is guaranteed to be bounded even when we have estimate error in the parameters of the system.

Under the nominal condition $\Delta f(t, x) = 0$, it is well know (cf. Khalil [2002]) that system (1) can be written as

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

where $y^{(r)}(t) = [y_1^{(r_1)}(t),\ y_2^{(r_2)}(t),\ \cdots,\ y_m^{(r_m)}(t)]^\top$, $\xi(t) = [\xi^1(t),\ \cdots,\ \xi^m(t)]^\top$, and $\xi^i(t) = [y_i(t),\ \cdots,\ y_i^{(r_i-1)}(t)]$ for $1 \le i \le m$. The functions $b(\xi)$ and $A(\xi)$ can be written as functions of f, g and h. $A(\xi)$ is non-singular in $\tilde{X}$, where $\tilde{X}$ is the image of the set of X by the diffeomorphism $x \mapsto \xi$ between the states of system (1) and the linearized model (2). To deal with the uncertain model, we need to introduce one more assumption on system (1).

Assumption A2 The additive uncertainties $\Delta f(t, x)$ in (1) appear as additive uncertainties in the input-output linearized model (2) as follows:

$$y^{(r)}(t) \;=\; b(\xi(t)) + A(\xi(t))u(t) + \Delta b(t, \xi(t)), \tag{3}$$

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. We can first define a virtual input vector $v(t)$ as

$$v(t) \;=\; b(\xi(t)) + A(\xi(t))u(t). \tag{4}$$

Combining (2) and (4), we obtain the following input-output mapping:

$$y^{(r)}(t) \;=\; v(t). \tag{5}$$

Based on the linear system (5), it is straightforward to design a stabilizing controller for the nominal system (2) as

$$u_n = A^{-1}(\xi)\left[v_s(t, \xi) - b(\xi)\right], \tag{6}$$

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

$$v_{si} = y_{id}^{(r_i)} - K_{r_i}^i(y_i^{(r_i-1)} - y_{id}^{(r_i-1)}) - \cdots - K_1^i(y_i - y_{id}). \tag{7}$$

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

$$e_i^{(r_i)}(t) + K_{r_i}^i e^{(r_i-1)}(t) + \cdots + K_1^i e_i(t) = 0, \tag{8}$$

where $i \in \{1, 2, \ldots, m\}$. By properly selecting the gains K_j^i ($i \in \{1, 2, \ldots, m\}$ and $j \in \{1, 2, \ldots, r_i\}$), which exist under certain assumptions, we can obtain global asymptotic stability of the tracking errors $e_i(t)$. Refer to [Benosman and Farahmand, 2016; Benosman et al., 2016] for the detail. It can be shown that there exists a positive definite matrix $P > 0$ such that (see e.g. Khalil [2002])

$$\tilde{A}^T P + P\tilde{A} \;=\; -\mathbf{I} \tag{9}$$

where $\tilde{A} \in \mathbb{R}^{n \times n}$ is a diagonal block matrix given by $\tilde{A} = \mathrm{diag}\{\tilde{A}_1, \tilde{A}_2, \ldots, \tilde{A}_m\}$, and $\tilde{A}_i$ ($1 \le i \le m$) is a $r_i \times r_i$ matrix given by

$$\tilde{A}_i \;=\; \begin{bmatrix} 0 & 1 & & & \\ 0 & 0 & 1 & & \\ \vdots & \vdots & \vdots & & 1 \\ -K_1^i & -K_2^i & \cdots & \cdots & -K_{r_i}^i \end{bmatrix}.$$

We now build a robust ISS controller for the uncertain model with $\Delta f(t,x) \neq 0$. The corresponding exact linearized model is given by (3) where $\Delta b(t,\xi(t)) \neq 0$. The global asymptotic stability of the error dynamics (8) cannot be guaranteed anymore due to the additive uncertainty $\Delta b(t,\xi(t))$. We use Lyapunov reconstruction techniques to design a new controller so that the tracking error is guaranteed to be bounded given that the estimate error of $\Delta b(t,\xi(t))$ is bounded.

The new controller for the uncertain model (3) is defined as $u_f = u_n + u_r$, where the nominal controller u_n is given by (6) and the robust controller u_r will be defined shortly. By plugging-in this controller in the dynamics (3), we obtain

$$\begin{aligned} y^{(r)}(t) &= b(\xi(t)) + A(\xi(t))u_f + \Delta b(t,\xi(t)) = b(\xi(t)) + A(\xi(t))u_n + A(\xi(t))u_r + \Delta b(t,\xi(t)) \\ &= v_s(t,\xi) + A(\xi(t))u_r + \Delta b(t,\xi(t)). \end{aligned} \tag{10}$$

This leads to the following error dynamics

$$\dot{z} = \tilde{A}z + \tilde{B}\delta, \tag{11}$$

where $\tilde{A}$ is defined as above (9), δ is a $m \times 1$ vector given by $\delta = A(\xi(t))u_r + \Delta b(t,\xi(t))$ and the matrix $\tilde{B} \in \mathbb{R}^{n\times m}$ is given by $\tilde{B}^\top = \left[\tilde{B}_1^\top \dots \tilde{B}_m^\top\right]^\top$ where each $\tilde{B}_i$ $(1 \leq i \leq m)$ is given by a $r_i \times m$ matrix such that $\tilde{B}_i(l,q) = 1$ for $l = r_1$ and $q = i$, and 0 otherwise. If we choose $V(z) = z^\top P z$ as a Lyapunov function for the dynamics (11), where P is the solution of the Lyapunov equation (9), we obtain

$$\dot{V}(t) = \frac{\partial V}{\partial z}\dot{z} = \; = z^T(\tilde{A}^T P + P\tilde{A})z + 2z^T P\tilde{B}\delta = -\|z\|^2 + 2z^T P\tilde{B}\delta. \tag{12}$$

Next, we design the controller u_r based on the form of the uncertainties $\Delta b(t,\xi(t))$. More specifically, we consider here the case when $\Delta b(t,\xi(t))$ is of the following form

$$\Delta b(t,\xi(t)) = E\,Q(\xi,t), \tag{13}$$

where $E \in \mathbb{R}^{m\times m}$ is a matrix of unknown constant parameters, and $Q(\xi,t):\ \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^m$ is a known bounded function of state and time variables. We denote the estimate of E by $\hat{E}(t)$ and the estimate error by $e_E = E - \hat{E}$. We define the unknown parameter vector $\Delta = [E(1,1),...,E(m,m)]^T \in \mathbb{R}^{m^2}$, i.e., concatenation of all elements of E, its estimate is denoted by $\widehat{\Delta}(t) = [\hat{E}(1,1),...,\hat{E}(m,m)]^T$, and the estimation error vector is given by $e_\Delta(t) = \Delta - \widehat{\Delta}(t)$. We now define the following robust controller:

$$u_r = -A^{-1}(\xi)[\tilde{B}^T P z\|Q(\xi,t)\|^2 + \widehat{E}(t)Q(\xi,t)]. \tag{14}$$

The closed-loop error dynamics can be written as

$$\dot{z} = \tilde{f}(t,z,e_\Delta), \tag{15}$$

where $e_\Delta(t)$ is considered to be an input to system (15).

Theorem 1. *Consider the system* (1) *and assume that Assumptions A1, A2 hold,* $\Delta b(t,\xi(t))$ *satisfies* (13)*, and the gains* K_j^i *stabilize* (8)*. If we apply the feedback controller* $u_f = u_n + u_r$ *(cf.* (6) *and* (14)*) to the uncertain nonlinear dynamical system* (1)*, the closed-loop system* (15) *is ISS from the estimation errors input* $e_\Delta(t) \in \mathbb{R}^{m^2}$ *to the tracking errors state* $z(t) \in \mathbb{R}^n$*.*

4 GP-UCB-based Uncertainties Estimation

The robust controller in the previous section ensures that the closed-loop system is ISS. The performance of the system, however, depends on the estimate $\widehat{\Delta}(t)$ used in the robust controller design (14). There are various ways to define the performance measure. For example, for a finite horizon $0 < T_f < \infty$, we may define it as

$$J(\widehat{\Delta}) = \int_0^{T_f} \left\|e(t;\widehat{\Delta})\right\|^2 + \left\|u_f(t;\widehat{\Delta})\right\|^2 \mathrm{d}t, \tag{16}$$

in which $e(t;\widehat{\Delta})$ and $u_f(t;\widehat{\Delta})$ are the tracking error and the control signal when $\widehat{\Delta}$ is used to design the controller.

To guide the search for $\widehat{\Delta}$ leading to a high-performance controller, we propose to use GP-UCB [Srinivas et al., 2010]. GP-UCB is a Bayesian approach for stochastic optimization, i.e., the task of finding the global optimum of an unknown function when the evaluations are potentially contaminated with noise. GP-UCB balances the exploration-exploitation in the continuous-armed bandit setting and has a guarantee in the form of cumulative regret bound.

We briefly describe GP-UCB following the original paper [Srinivas et al., 2010]. Consider the cost function $J : D \to \mathbb{R}$ to be minimized, e.g., (16). This function depends on the dynamics of the closed-loop system, which itself depends on the parameters $\widehat{\Delta}$ used in the controller design (14). A Gaussian Process (GP) [Rasmussen and Williams, 2006] is defined by its mean function $\mu(\widehat{\Delta}) = \mathbb{E}\left[J(\widehat{\Delta})\right]$ and covariance function (or kernel) $\kappa(\widehat{\Delta}_1, \widehat{\Delta}_2) = \mathrm{Cov}(J(\widehat{\Delta}_1), J(\widehat{\Delta}_2))$. An example of the kernel function is $\kappa(\widehat{\Delta}_1, \widehat{\Delta}_2) = \exp(-\frac{\|\widehat{\Delta}_1 - \widehat{\Delta}_2\|^2}{2l^2})$, the squared exponential kernel with length scale $l > 0$.

The combination of GP-UCB and the controller design method of Section 3 works iteratively. Suppose that we are at iteration $\tau = 1, 2, \ldots$ and we have already selected the set of parameters $\underline{\widehat{\Delta}}_{\tau-1} \triangleq \{\widehat{\Delta}_1, \widehat{\Delta}_2, \ldots, \widehat{\Delta}_{\tau-1}\} \subset D$. For each $i = 1, \ldots, \tau - 1$, we have also observed the noisy evaluation $y_i = J(\widehat{\Delta}_i) + \eta_i$ with $\eta_i \sim N(0, \sigma^2)$ being i.i.d. Gaussian noise. The value of y_i is obtained after using $\widehat{\Delta}_i$ to design the robust controller (cf. (14)) and running the closed-loop dynamical system with the control signal $u_f = u_n + u_r$ (cf. (6) and (14)).

GP-UCB requires the computation of the posterior mean and variance of the GP given data. Suppose that initially we started with $\mathrm{GP}(0, \kappa)$, a GP with zero prior mean. We find the posterior mean and variance for a new point $\widehat{\Delta}^* \in D$ as follows: Denote the vector of observed values by $\mathbf{y}_{\tau-1} = [y_1, \ldots, y_{\tau-1}]^\top \in \mathbb{R}^{\tau-1}$, and define the Grammian matrix $K \in \mathbb{R}^{\tau-1 \times \tau-1}$ with $[K]_{i,j} = \kappa(\widehat{\Delta}_i, \widehat{\Delta}_j)$, and the vector $\kappa_* = [\kappa(\underline{\widehat{\Delta}}_1, \widehat{\Delta}^*), \ldots, \kappa(\underline{\widehat{\Delta}}_{\tau-1}, \widehat{\Delta}^*)]$. The expected mean $\mu_{\tau-1}(\widehat{\Delta}^*)$ and the variance $\sigma_{\tau-1}(\widehat{\Delta}^*)$ of the posterior of the GP evaluated at $\widehat{\Delta}^*$ are (cf. Section 2.2 of Rasmussen and Williams [2006]) $\mu_{\tau-1}(\widehat{\Delta}^*) = \kappa_* \left[K + \sigma^2 \mathbf{I}\right]^{-1} \mathbf{y}_{\tau-1}$ and $\sigma^2_{\tau-1}(\widehat{\Delta}^*) = \kappa(\widehat{\Delta}^*, \widehat{\Delta}^*) - \kappa_*^\top \left[K + \sigma^2 \mathbf{I}\right]^{-1} \kappa_*$. At iteration τ, the GP-UCB algorithm selects the next query point $\widehat{\Delta}_\tau$ by solving the following optimization problem:

$$\widehat{\Delta}_\tau \leftarrow \operatorname*{argmin}_{\widehat{\Delta} \in D} \mu_{\tau-1}(\widehat{\Delta}) - \beta_\tau^{1/2} \sigma_{\tau-1}(\widehat{\Delta}). \tag{17}$$

Here β_τ depends on the choice of kernel among other parameters of the problem [Srinivas et al., 2010]. This process repeats. The optimization problem (17) is often nonlinear and non-convex. Nonetheless, solving it only requires querying the GP, which in general is much faster than querying the original dynamical system. This is important when the dynamical system is a physical system and having as few number of interactions as possible is crucial.

5 Conclusion

We used tools from machine learning and modern control engineering to design an adaptive robust controller for a class of nonlinear systems. The designed controller is guaranteed to be input-state stable, while the Bayesian optimization technique allows its performance to improve. We have empirically studied this approach for the control of a two-link manipulator with favourable results [Benosman and Farahmand, 2016]. One of the main advantages of the proposed controller, compared to the existing model-based adaptive controllers, is that we can estimate multiple uncertainties at the same time even if they appear in the model equation in a challenging structure, e.g., linearly dependent uncertainties affecting only one output, or uncertainties appearing in a nonlinear term of the model, which are well-known limitations of the model-based estimation approaches.

References

G. Atinc and M. Benosman. Nonlinear learning-based adaptive control for electromagnetic actuators with proof of stability. In *IEEE, Conference on Decision and Control*, pages 1277–1282, 2013.

M. Benosman. Extremum-seeking based adaptive control for nonlinear systems. In *IFAC World Congress*, pages 401–406, 2014.

M. Benosman and G. Atinc. Multi-parametric extremum seeking-based learning control for electromagnetic actuators. In *American Control Conference*, pages 1914–1919, 2013.

M. Benosman and A.-m Farahmand. Bayesian optimization-based modular indirect adaptive control for a class of nonlinear systems. In *IFAC International Workshop on Adaptation and Learning in Control and Signal Processing*, 2016.

M. Benosman, A.-m Farahmand, and M. Xia. Learning-based modular indirect adaptive control for a class of nonlinear systems. In *American Control Conference (ACC)*, 2016.

P. Haghi and K. Ariyur. Adpative feedback linearization of nonlinear MIMO systems using ES-MRAC. In *American Control Conference*, pages 1828–1833, 2013.

H. Khalil. *Nonlinear Systems*. Prentice Hall, 3rd edition, 2002.

M. Krstic, I. Kanellakopoulos, and P. Kokotovic. *Nonlinear and Adpative Control Design*. Wiley, New York, 1995.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

N. Srinivas, A. Krause, S. M Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning (ICML)*, pages 1015–1022, 2010.

C. Wang, D. J Hill, S. S. Ge, and G. Chen. An ISS-modular approach for adaptive neural control of pure-feedback systems. *Automatica*, 42(5):723–731, 2006.

M. Xia and M. Benosman. Extremum seeking-based indirect adaptive control for nonlinear systems with time-varying uncertainties. In *European Control Conference*, pages 2780–2785, 2015.