Bayesian Nonparametric Adaptive Control Using Gaussian Processes

Bayesian Nonparametric Adaptive Control using Gaussian Processes

Girish Chowdhary, Hassan A. Kingravi, Jonathan P. How, and Patricio A. Vela

***Abstract*—Most current Model Reference Adaptive Control (MRAC) methods rely on parametric adaptive elements, in which the number of parameters of the adaptive element are fixed a priori, often through expert judgment. An example of such an adaptive element are Radial Basis Function Networks (RBFNs), with RBF centers pre-allocated based on the expected operating domain. If the system operates outside of the expected operating domain, this adaptive element can become non-effective in capturing and canceling the uncertainty, thus rendering the adaptive controller only semi-global in nature. This paper investigates a Gaussian Process (GP) based Bayesian MRAC architecture (GP-MRAC), which leverages the power and flexibility of GP Bayesian nonparametric models of uncertainty. GP-MRAC does not require the centers to be preallocated, can inherently handle measurement noise, and enables MRAC to handle a broader set of uncertainties, including those that are defined as distributions over functions. We use stochastic stability arguments to show that GP-MRAC guarantees good closed loop performance with no prior domain knowledge of the uncertainty. Online implementable GP inference methods are compared in numerical simulations against RBFN-MRAC with preallocated centers and are shown to provide better tracking and improved long-term learning.**

***Index Terms*—Kernel, Adaptive control, Gaussian Processes, Nonlinear control systems**

I. INTRODUCTION

For many physical applications, obtaining an exact model of the system dynamics is prohibitive. Control synthesis using an an approximate model can lead to poor or undesirable control response due to the modeling error. Adaptive control theory seeks strategies to mitigate performance loss and to ensure stable, closed-loop performance in the face of uncertainty. One direct adaptive control methodology with good performance guarantees in the presence of significant modeling uncertainty is Model Reference Adaptive Control (MRAC) [3], [19], [43], [57], [60]. Within the class of MRAC algorithms, Radial Basis Function Neural Networks (RBFNs) have emerged as a widely used universal-approximator adaptive model [52], especially when little is known about the uncertainty [26], [41], [59]. One reason for the popularity of RBFNs is that they are linear-in-the-parameters, as opposed to multi-layer perceptron Neural Networks [26], [33]. The accuracy of an RBFN representation, however, greatly depends on the choice of RBF centers [45]. Typically, authors have assumed that the operating domain of the system is known, and pre-allocated a fixed quantity of Gaussian RBF centers over the presumed domain [12], [26], [41], [46], [61].

Since the output of a Gaussian RBF decays exponentially when evaluated away from its center, the system states must remain close to the location of some RBF centers for the RBFN to effectively capture the uncertainty. Therefore, an implicit assumption in RBFN-MRAC is that the domain over which the uncertainty is defined is compact, so that it can be well approximated by a finite set of RBFs. Consequently, RBFN stability results cannot be extended globally for if the system states were to evolve outside of the assumed compact domain of operation, the RBFN is effectively unable to learn and approximate the uncertainty. One way to overcome the limitation of fixed centers is to move and add/remove RBF centers to better capture the modeling uncertainty. Conforming to the traditional approach in MRAC, authors in [40], [55], [58] have proposed RBF center tuning rules that attempt to minimize the instantaneous tracking error. However, while reducing the modeling error guarantees that the tracking error is reduced [6], [12], the opposite need not always be true. Therefore, techniques such as those in [40], [55], [58] do not guarantee that the center updates reduce the modeling error. Another limitation of traditional RBFN based MRAC methods is that they model the uncertainty as smooth deterministic function. However, in real-world situations uncertainties may have several stochastic effects, such as noise, servo chattering, turbulence etc. Authors have relied on σ-modification [19] like damping terms in parameter update equations for guaranteeing the boundedness of parameters of deterministic models of uncertainty in presence of noise [10], [31]. This approach can guarantee overall boundedness of the closed-loop system, but the added damping limits learning of the uncertainty. Therefore there is a need for better models of stochastic uncertainties using probabilistic modeling notions such as mean and variance.

This paper employs Gaussian Processes (GP) as Bayesian nonparametric adaptive elements in MRAC to address the aforementioned limitations of RBFN-MRAC with preallocated centers. As an alternative to methods with fixed parametric structure, nonparametric models are designed to overcome the local approximation properties of universal approximators. The number of parameters and their properties are not fixed, rather, they grow and adjust with the data. Within the class of nonparametric modeling methods, Bayesian modeling approaches lead to data-driven, generative models, where the model is optimized to fit the data assuming uncertain measurements. GP based models are an example of a Bayesian nonparametric regression model [50]. GPs are known to have a deep connection with kernel filtering methods through a reproducing kernel Hilbert space interpretation of GP regression [2]. The benefits of using GP Bayesian nonparametric adaptive elements include: no prior knowledge of the operating domain of the uncertainty is required, measurement noise is inherently handled, and the centers need not be pre-allocated. GP uncertainties are defined via *distributions over functions*,

G. Chowdhary and J.P. How are with the Department of Aeronautics and Astronautics at the Massachusetts Institute of Technology, Cambridge, MA, and H. A. Kingravi and P. A. Vela are with the School of Electrical and Computer Engineering at the Georgia Institute of Technology, Atlanta, GA. girishc,jhow@mit.edu, kingravi,pvela@gatech.edu

which differs from the traditional deterministic weight-space based approaches [26], [27], [41], [61]. Furthermore, Bayesian inference overcomes the shortcomings of the standard gradient based MRAC parameter update laws, such as the lack of convergence guarantees and the possibility of bursting (parameters growing unboundedly) in presence of noise [3], [6], [42].

When used in an online setting such as for adaptive control, the number of parameters in GP adaptive elements grows linearly with the measurements [50]. Furthermore, most sparsification techniques used to limit the size of the GP model require access to the entire set of training data [34], [50], [54], which cannot be provided in real-time control settings. Therefore, in order to ensure real-time feasibility, we enforce an additional restriction that the number of maximum allowable parameters at any instant of time be limited (this number is referred to as the "budget"). Once the budget is reached, any new centers are added by removing older (possibly irrelevant) centers. Each time a new center is added, the structure of the GP changes discontinuously. Thus, the stability properties of the presented GP-MRAC controllers must be established through switched stochastic stability theory. Our GP-MRAC approach removes long-standing assumptions on bounded domain of operation, a priori known number and location of RBF centers, and deterministic input-output models of the uncertainty, while being on-line implementable.

A. Related Work

Nonparametric models in adaptive control have been previously considered. Cannon and Slotine presented a heuristic algorithm to add and remove nodes of a nonparametric wavelet network over a bounded domain for adaptive control [9]. Bernard and Slotine established stable adaptive controllers using finite approximations of infinite series of wavelet function regressors [4]. In contrast to the nonparameteric models in those aforementioned works, our approach leverages information theoretic concepts such as Bayesian inference and Kullback-Leibler divergence for updating the weights and selecting the regressors [17]. Furthermore, our approach does not assume a predefined bounded domain of operation. Alpcan investigated active optimal control with Gaussian processes in the context of dual control [1]. Murray-Smith *et al* have explored Gaussian processes in the context of dual adaptive control [38], [39]. Nguyen-Tuong and Peters combined a physics based model with a Gaussian Process model for learning inverse dynamics of stable systems such as robotic arms [44]. Ko and Fox explored GP based dynamic observation and measurement models in Kalman filter frameworks [28]. Diesenroth, Ko, Rasmussen, and others have used GPs for solving control problems using model based reinforcement learning [15], [29], [49]. The above approaches require access to off-line training data for generating control models. The key difference between these methods and our approach is that GPs are used here in the MRAC framework, which is a direct adaptive control framework (output of the online trained GP is directly used for control). Similar to traditional MRAC, the focus of our architecture is to guarantee stability and good tracking of fast, unstable dynamical systems online the *first-time-around*, without using any previously recorded data. This objective is achieved by leveraging recent advances in sparse online GP learning [14] and providing stability guarantees using a new analysis approach that utilizes stochastic control theory for switched uncertain systems.

In [27] we took the first steps to unite recent advances in both kernel methods and adaptive control. The adaptive control system used non-Bayesian nonparametric kernel models with gradient based update laws. This paper incorporates Bayesian nonparametric regression techniques in adaptive control. Preliminary ideas regarding the work presented here first appeared in the conference paper [11]. The main contributions of the current work are improved techniques for online sparsification and inference, detailed stability analysis, and more complete analysis of numerical simulations.

B. Notation

Let $\mathbb{E}$ denote the expectation operator, $\mathbb{E}_x$ the expectation operator conditioned on the measurement x, and $\mathbb{V}(x)$ the variance of x. The trace operator is denoted by tr. The class of n^{th} continuously differentiable functions is denoted by $\mathcal{C}^n$. The operators $\lambda_{\min}(.)$ and $\lambda_{\max}(.)$ return the minimum and the maximum eigenvalue of a matrix. For a matrix Z, $|Z|$ denotes the number of columns. Lastly, $t \geq 0$ denotes time, where the argument t is sometimes dropped from an equation for ease of exposition.

II. Approximate Model Inversion based Model Reference Adaptive Control (AMI-MRAC)

AMI-MRAC is an approximate feedback-linearization based MRAC method that allows the design of adaptive controllers for a general class of nonlinear plants (see e.g. [7], [20]). The GP-MRAC approach is introduced in the framework of AMI-MRAC, although it should be noted that it is applicable to other MRAC architectures (see e.g. [3], [19], [43], [60]). Let $x(t) = [x_1^T(t), x_2^T(t)]^T \in D_x \subset \mathbb{R}^n$, such that $x_1(t) \in \mathbb{R}^{n_s}$, $x_2(t) \in \mathbb{R}^{n_s}$, and $n = 2n_s$. Let $\delta \in D_\delta \subset \mathbb{R}^l$, and consider the following multiple-input controllable control-affine nonlinear uncertain dynamical system

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= f(x(t)) + b(x(t))\delta(t). \end{aligned} \tag{1}$$

The functions $f(0)$, $f(0) = 0$ and b are partially unknown functions assumed to be Lipschitz over a domain $\mathcal{D}$ and the control input δ is assumed to be bounded and piecewise continuous, so as to ensure the existence and uniqueness of the solution to (1) over $\mathcal{D}$. Also assume that $l \leq n_s$ (while restrictive for overactuated systems, this assumption can be relaxed through the design of appropriate control assignment [16]). Further note that while the development here is restricted to control-affine systems, sufficient conditions exist to convert a class of non-affine in control nonlinear systems to the control-affine form in (1) (see Chapter 13 in [22]), and the AMI-MRAC framework can also be extended to a class of non-affine in control systems [21], [25].

The AMI-MRAC approach used here feedback linearizes the system by finding a pseudo-control input $\nu(t) \in \mathbb{R}^{n_s}$ that achieves a desired acceleration. If the exact plant model in (1) is known and invertible, the required control input to achieve the desired acceleration is computable by inverting the plant dynamics. However, since this usually is not the case, an approximate inversion model $\hat{f}(x)+\hat{b}(x)\delta$, where $\hat{b}$ chosen to be nonsingular for all $x(t) \in D_x$, is employed.

Given a desired pseudo-control input $\nu \in \mathbb{R}^{n_s}$ a control command δ can be found by approximate dynamic inversion:

$$\delta = \hat{b}^{-1}(x)(\nu - \hat{f}(x)). \tag{2}$$

Let $z = (x^T, \delta^T)^T \in \mathbb{R}^{n+l}$ for brevity. The use of an approximate model leads to modeling error Δ for the system,

$$\dot{x}_2 = \nu(z) + \Delta(z), \tag{3}$$

with

$$\Delta(z) = f(x) - \hat{f}(x) + (b(x) - \hat{b}(x))\delta. \tag{4}$$

Were b known and invertible with respect to δ, then an inversion model exists such that the modeling error is not dependent on the control input δ. A designer chosen reference model is used to characterize the desired response of the system

$$\begin{aligned} \dot{x}_{1_{rm}} &= x_{2_{rm}}, \\ \dot{x}_{2_{rm}} &= f_{rm}(x_{rm}, r), \end{aligned} \tag{5}$$

where $f_{rm}(x_{rm}, r)$ denotes the reference model dynamics, assumed to be continuously differentiable in x_{rm} for all $x_{rm} \in D_x \subset \mathbb{R}^n$. The command $r(t)$ is assumed to be bounded and piecewise continuous. Furthermore, f_{rm} is assumed to be such that x_{rm} is bounded for a bounded command.

Define the tracking error to be $e(t) = x_{rm}(t) - x(t)$, and the pseudo-control input ν to be

$$\nu = \nu_{rm} + \nu_{pd} - \nu_{ad}, \tag{6}$$

consisting of a linear feedforward term $\nu_{rm} = \dot{x}_{2_{rm}}$, a linear feedback term $\nu_{pd} = [K_1, K_2]e$ with $K_1 \in \mathbb{R}^{n_s \times n_s}$ and $K_2 \in \mathbb{R}^{n_s \times n_s}$, and an adaptive term $\nu_{ad}(z)$. For ν_{ad} to be able to cancel Δ, the following assumption needs to be satisfied:

Assumption 1 There exists a unique fixed-point solution to $\nu_{ad} = \Delta(x, \nu_{ad})$, $\forall x \in D_x$.

Assumption 1 implicitly requires the sign of control effectiveness to be known [25]. Sufficient conditions for satisfying this assumption are available in [25], [63].

Using (3) the tracking error dynamics can be written as

$$\dot{e} = \dot{x}_{rm} - \begin{bmatrix} x_2 \\ \nu + \Delta \end{bmatrix}. \tag{7}$$

Let $A = \begin{bmatrix} 0 & I \\ -K_1 & -K_2 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ I \end{bmatrix}$, where $0 \in \mathbb{R}^{n_s \times n_s}$ and $I \in \mathbb{R}^{n_s \times n_s}$ are the zero and identity matrices, respectively. From (6), the tracking error dynamics are then,

$$\dot{e} = Ae + B[\nu_{ad}(z) - \Delta(z)]. \tag{8}$$

The baseline full state feedback controller ν_{pd} is chosen to make A Hurwitz. Hence, for any positive definite matrix $Q \in \mathbb{R}^{n \times n}$, a positive definite solution $P \in \mathbb{R}^{n \times n}$ exists for the Lyapunov equation

$$0 = A^T P + PA + Q. \tag{9}$$

When Gaussian RBFN are used as adaptive elements, the adaptive part of the control law (6) is represented by a linear combination of Gaussian RBFs $\nu_{ad}(z) = W^T \Phi(z)$ where $W \in \mathbb{R}^{n_2 \times q}$ and $\Phi(z) = [1, \phi_2(z), \phi_3(z), ..., \phi_q(z)]^T$ is a q dimensional vector of radial basis functions. For $i = 2, 3..., q$ let c_i denote the RBF centroid and μ_i denote the RBF widths, then the Gaussian RBFs are given as $\phi_i(x) = \exp(\frac{-\|x - c_i\|^2}{2\mu_i^2})$. Gaussian RBFs are universal approximators; they can model a continuous function over a compact domain to arbitrary accuracy given q sufficiently large [45].

A. Limitations of fixed-parameter RBFN-MRAC

Typical implementations of adaptive controllers using RBFN in the literature rely on estimating the operating domain D of the system and pre-allocating the centers of a predetermined number of RBFs over that domain. In this case, it can be shown that the following weight update law,

$$\dot{W} = \text{Proj}\left(-\Gamma_W e^T PB\Phi(z)\right), \tag{10}$$

with the projection operator used to bound the weights [47], guarantees uniform ultimate boundedness[1] of the tracking error and adaptive parameters if the system operates within the domain over which the centers are distributed [52], [60], Here, Γ_W denotes a positive definite learning rate matrix. This adaptive law does not in general guarantee that the ideal parameter vector W^* is attractive unless a condition on Persistency of Excitation (PE) of the system states is satisfied [6], [26], [60]. Kingravi *et al.* showed that location of centers also affects the amount of excitation that is "visible" to the adaptive law (10) [27]. Even if $x(t)$ is exciting, when the system evolves away from where the centers are distributed, persistent excitation for the adaptive system may fail to hold. In summary, two major limitations of RBFN with preallocated centers are:

1 The RBFN approximation holds within the estimated operating domain D over which the centers have been preallocated; RBFN-MRAC is only locally effective.

2 It is difficult to guarantee the convergence of the parameters to their ideal values when adaptive laws such as (10) are used, indicating that the optimal approximation guarantee by the universal approximation theorem may not be achieved when using RBFN-MRAC.The problem is further exacerbated by the fact that the amount of excitation "visible" to the adaptive element may diminish if the system evolves outside of the estimated operating domain, over which the centers have been preallocated.

III. ADAPTIVE CONTROL USING GAUSSIAN PROCESS REGRESSION

Traditionally MRAC assumes that the uncertainty, $\Delta(z)$ in (4), is a (smooth) deterministic function for which an input-output map in the form of an NN or RBFN is learned. This paper offers an alternate view by assuming that the uncertainty is described by a time varying (prior) mean and covariance

[1] When RBFN, or other approximation based adaptive elements are used, one cannot guarantee asymptotic convergence of the tracking error to zero if there is a nonzero approximation error, even when the states are PE.

function, and using Gaussian Processes (GP) to learn the continuous function of time and state [48]. As the system evolves and measurements are taken, Bayesian posterior updates build a generative model of the uncertainty. Learning via probabilistic generative models offers a promising new approach. In contrast to learning a NN input-output representation [37], [48], the approach does not easily succumb to overlearning and also handles noise. Furthermore, it allows incorporation of stochastic effects such as servo chattering, external disturbances, and other non-deterministic effects.

A GP is defined as a collection of random variables such that every finite subset is jointly Gaussian. The joint Gaussian condition means that GPs are completely characterized by their second order statistics [50]. A GP is a distribution over functions, that is, a draw from a GP is a function. For the sake of clarity of exposition, we will assume that $\Delta(z) \in \mathbb{R}$; the extension to the multidimensional case is straightforward. When Δ follows a Gaussian process model, then

$$\Delta(\cdot) \sim \mathcal{GP}(m(\cdot), k(\cdot,\cdot)), \tag{11}$$

where $m(\cdot)$ is the mean function, and $k(\cdot,\cdot)$ is a real-valued, positive definite covariance kernel function. Under GP regression, the mean is assumed to lie in the class of functions

$$\mathcal{G} = \left\{ g(\cdot) \in \mathbb{R}^{\mathcal{X}} \;\middle|\; g(\cdot) = \sum_{i=1}^{\infty} \alpha_i k(z_i, \cdot) \right\}, \tag{12}$$

where $\mathcal{X} = \mathbb{R}^n$, $\alpha_i \in \mathbb{R}$, $z_i \in \mathcal{X}$. The space $\mathcal{G}$ is a subspace of $\mathcal{H}$, a reproducing kernel Hilbert space (RKHS), and $\|g\|_{\mathcal{H}} < \infty$ where $\|g(\cdot)\|^2_{\mathcal{H}} = \sum_{i=1}^{\infty}\sum_{j=1}^{\infty} \alpha_i \alpha_j k(z_i, z_j)$. This assumption imposes a smoothness prior on the mean and renders the problem more amenable to analysis though the representer theorem [53]. Figure 1 demonstrates the ability of GP regression to mitigate noise in the data. An accurate model of the underlying function is learned although the data is corrupted by Gaussian white noise of variance ω^2.

A. GP Regression and Reproducing Kernel Hilbert Spaces

Let $Z_\tau = \{z_1, \ldots, z_\tau\}$ be a set of state measurements, discretely sampled where $\{1 \ldots \tau\}$ are indices for the discrete sample times $\{t_1, \ldots, t_\tau\}$. The set defines a covariance matrix $K_{ij} := k(z_i, z_j)$. Given indexed sets A and B, $K(A, B)$ denotes the kernel matrix generated by the evaluations $K_{ij} = k(a_i, b_j)$ between the two sets, where $a_i \in A, b_j \in B$. For each measurement z_i, there is an observed output $y(z_i) = m(z_i) + \epsilon_i$, where $\epsilon_i \sim \mathcal{N}(0, \omega^2)$. The stacked outputs give $y = [y_1, \ldots, y_t]^T$. The most common choice of covariance kernel, and the one used here, is the Gaussian RBF kernel:

$$k(z, z') = \exp\left(-\frac{\|z - z'\|^2}{2\mu^2} \right), \tag{13}$$

The positive definite function k generates a mapping ψ to a RKHS $\mathcal{H}$ such that $k(z_i, z_j) = \langle \psi(z_i), \psi(z_j) \rangle_{\mathcal{H}}$. A finite set of points Z generates a finite-dimensional, linearly parameterized vector space of functions $\mathcal{F}_Z$ via an associated set of basis functions [5]. Figure 2 shows an example of the mapping ψ and the linear sub-space generated via data points.

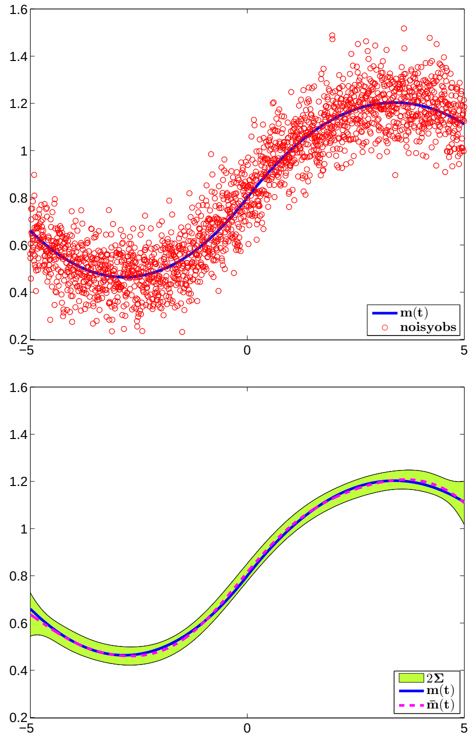


Fig. 1. An example of GP inference for a set of 2,000 measurements drawn from $\Delta(z) \sim \mathcal{GP}(m(\cdot), k(\cdot,\cdot))$, corrupted by i.i.d observation noise drawn from $\mathcal{N}(0, \omega^2)$. $m(\cdot)$ is the estimated mean, and $\Sigma(\cdot)$ is the estimated posterior variance.

GP regression fuses RKHS theory with Bayesian linear regression by utilizing a regression model of the form

$$m(z) = \beta^T \Psi(z) = \sum_{i \in \mathcal{I}} \beta_i \langle \psi(z_i), \psi(z) \rangle_{\mathcal{H}}, \tag{14}$$

where $\beta \in \mathcal{F}_Z$ is a coordinate vector (of weights), $\psi(z_i) \in \mathcal{H}$ are basis vectors, and $\mathcal{I}$ is the index set. Since GP adaptive elements admit infinite basis functions and associated parameters, they are referred to as nonparametric [50].

GP regression assumes that the uncertainty in the data and the model follow Gaussian distributions, while modeling the function estimate using a mean function $\hat{m}$ and a covariance function $\hat{\Sigma}$. Since the observations are Gaussian, the likelihood function $p(y_\tau | Z_\tau, \beta)$ is Gaussian. The initial prior is set to $p(\beta) \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma_w})$, and Bayes' rule is used to infer the posterior distribution $p(\beta | Z_\tau, y_\tau)$ with each new observation. Since the posterior is Gaussian, the update generates a revised mean $\hat{m}_\tau$ and covariance $\hat{\Sigma}_\tau$. If $|Z_\tau|$ is finite, the solution for the posterior mean and covariance is also finite [53]. In particular, given a new input $z_{\tau+1}$, the joint distribution of the data available up to τ and z_τ under the prior distribution is

$$\begin{bmatrix} y_\tau \\ y_{\tau+1} \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(Z_\tau, Z_\tau) + \omega^2 I & k_{z_{\tau+1}} \\ k^T_{z_{\tau+1}} & k^*_{\tau+1} \end{bmatrix} \right), \tag{15}$$

where $k_{z_{\tau+1}} = K(z_{\tau+1}, Z_\tau)$ and $k^*_{\tau+1} = k(z_{\tau+1}, z_{\tau+1})$. The posterior (sometimes called the predictive) distribution, obtained by conditioning the joint Gaussian prior distribution over the observation z_{t+1}, is computed by

$$p(y_{\tau+1} | Z_\tau, y_\tau, z_{\tau+1}) \sim \mathcal{N}(\hat{m}_{\tau+1}, \hat{\Sigma}_{\tau+1}), \tag{16}$$

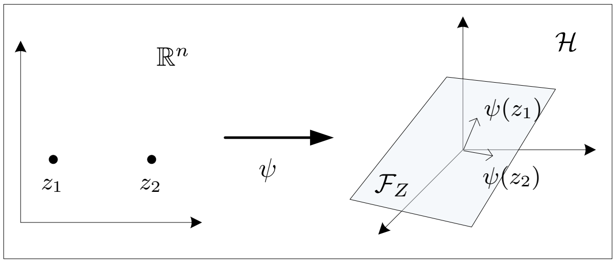


Fig. 2. An example Hilbert space mapping. If z_1 and z_2 are two points in $\mathbb{R}^n$, they generate the linear subspace $\mathcal{F}_Z \subset \mathcal{H}$, which is a family of linearly parameterized functions, via the mappings $\psi(z_1)$ and $\psi(z_2)$.

where

$$\hat{m}_{\tau+1} = \beta_{\tau+1}^T k_{z_{\tau+1}} \tag{17}$$

$$\hat{\Sigma}_{\tau+1} = k_{\tau+1}^* - k_{z_{\tau+1}}^T C_\tau k_{z_{\tau+1}} \tag{18}$$

are the updated mean and covariance estimates, respectively, and where $C_\tau := (K(Z_\tau, Z_\tau) + \omega^2 I)^{-1}$ and $\beta_{\tau+1} := C_\tau y_\tau$. Due to the Representer Theorem, the values in (17) are the best possible that could be obtained given the available data [50], [53].

Since both Z_τ and y_τ grow with data, computing the inverse becomes computationally intractable over time. This is less of a problem for traditional GP regression applications, which often involve finite learning samples and offline learning. However, in an online setting, the linear growth in the sample set cardinality degrades computational performance. Therefore, the extension of GP regression to MRAC, requires an online method to restrict the number of data points stored for inference. In the following section we outline two simple schemes for this purpose and incorporate them with MRAC to form GP-MRAC.

B. GP Bayesian nonparametric model based MRAC

We model the uncertainty using a Gaussian process based adaptive element ν_{ad}

$$\nu_{ad}(z) \sim \mathcal{GP}(\hat{m}(z), k(z, z')), \tag{19}$$

where $\hat{m}(z)$ is the estimate of the mean of (11) updated using (17) with the coordinate vector denoted by α instead of β. The adaptive signal ν_{ad} is set to the estimate of the mean $\hat{m}(z)$. Since GPs are completely characterized by their first two moments, $\|\hat{m}(z) - m(z)\| < \epsilon_1$ implies $\|\nu_{ad} - \Delta\| < \epsilon_2$ with high probability, for some $\epsilon_1, \epsilon_2 > 0$. While the posterior calculated in Equation (16) converges to the true posterior [50], it becomes untenable in an online setting due to the growth of $|Z|$. There needs to be a limit on the number of datapoints stored for posterior inference.

Many schemes exist for the approximation of a GP with a smaller set of bases, but these typically assume all the data is available. The schemes, not designed for an online setting, can be computationally costly [8], [18]. Since the set Z generates a family of functions $\mathcal{F}_Z \subset \mathcal{H}$ whose richness characterizes the quality of the posterior inference, a natural and simple way to determine whether to add a new point to the subspace is to check how well it is approximated by the elements in Z. This

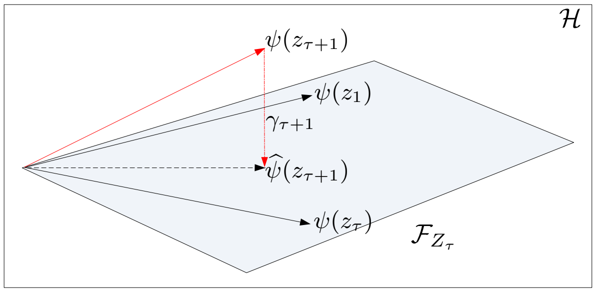


Fig. 3. An example of the projection of $\psi(z_{\tau+1})$ onto the subspace $\mathcal{F}_{Z_\tau}$ spanned by $\{\psi(z_1), \ldots, \psi(z_\tau)\}$. The scalar $\gamma_{\tau+1}$ is the length of the residual, and is a measure of the independence of $\psi(z_{\tau+1})$ w.r.t. $\mathcal{F}_{Z_\tau}$.

Algorithm 1 The Generic Gaussian Process - Model Reference Adaptive Control (GP-MRAC) algorithm

```
while new measurements are available do
    Given $z_{\tau+1}$, compute $\gamma_{\tau+1}$ using (21)
    Compute $y_{\tau+1} = \dot{\hat{x}}_{2_{\tau+1}} - \nu_{\tau+1}$
    if $\gamma_{\tau+1} > \epsilon_{tol}$ then
        if $|\mathcal{BV}(\sigma)| > p_{\max}$ then
            Delete element in $\mathcal{BV}(\sigma)$ based on methods in Section III-B
        end if
        Increase the switching index $\sigma$.
    end if
    Calculate $\hat{m}_{\tau+1}$ and $\hat{\Sigma}_{\tau+1}$.
    Set $\nu_{ad} = \hat{m}_{\tau+1}$.
    Calculate pseudo control $\nu$ using (6).
    Calculate control input using (2).
end while
```

is known as the kernel linear independence test [14], and is computed by

$$\gamma_{\tau+1} = \left\| \sum_{i=1}^{\tau} \alpha_i \psi(z_i) - \psi(z_{\tau+1}) \right\|_{\mathcal{H}}^2. \tag{20}$$

The scalar $\gamma_{\tau+1}$ is the length of the residual of $\psi(z_{\tau+1})$ projected onto the subspace $\mathcal{F}_{Z_\tau}$, as illustrated in Figure 3. When $\gamma_{\tau+1}$ is larger than a specified threshold, then a new data point should be added to the data set. The coefficient vector α minimizing (20) is given by $\alpha_\tau = K_{Z_\tau}^{-1} k_{z_{\tau+1}}$, meaning that

$$\gamma_{\tau+1} = k_{\tau+1}^* - k_{z_{\tau+1}}^T \alpha_\tau. \tag{21}$$

This restricted set of selected elements, called the *basis vector set*, is denoted by $\mathcal{BV}$. When incorporating a new data point into the GP model, the inverse kernel matrix can be recomputed with a rank-1 update.

When the budget is exceeded, a basis vector element must be removed prior to adding another element [8]. Here we examine two such schemes. The first, denoted **OP**, simply deletes the oldest vector: this scheme prioritizes temporal locality of the data for the approximation. The second, denoted **KL**, employs the *sparse online Gaussian process* algorithm [14]. The latter algorithm efficiently approximates the KL divergence between the current GP and the $(t + 1)$ alternative

GPs missing one data point each, then deletes removes the data point with the largest KL divergence. The equations for implementing the sparse online GP algorithm are detailed in the appendix as Algorithm 2.

IV. Analysis of Stability

A. Stochastic Stability Theory for Switched Systems

We begin by introducing the necessary tools in stochastic stability analysis. Consider switched stochastic differential equations of the Itô type whose solutions are a class of continuous time Markov processes [24], [32]. The system equations are

$$dx(t) = F(t,x(t))dt + G_\sigma(t,x(t))d\xi(t), \quad x(0) = x_0, \tag{22}$$

where $x \in \mathbb{R}^{n_s}$, $\xi(t) \in \mathbb{R}^{n_2}$ is a Wiener process, $\sigma(t) \in \mathbb{N}$ is the switching index which switches finitely many times in any finite time interval, $F(t,x)$ is an n_s-vector function, and $G_\sigma(t,x)$ is an $n_s \times n_2$ matrix. Assume that $F(t,0) = 0$ and $G_\sigma(t,0) = 0$. The functions $F(t,x(t))$ and $G_\sigma(t,x(t))$ are assumed to satisfy the Lipschitz condition for each switching index σ

$$\|F(t,x) - F(t,y)\| + \|G_\sigma(t,x) - G_\sigma(t,y)\| \le B\|x-y\|$$

for all x over a compact domain $\mathcal{D}$. Under these conditions the solution of (22) is a continuous Markov process. Note that the assumption on Lipschitz continuity of G_σ is reasonable for the GP formulation considered here because components of G_σ turn out to be continuously differentiable kernel functions.

The following definitions concerning the (exponential) ultimate boundedness of the solution of (22) are introduced.

Definition 1 The process $x(t)$ is said to be *mean square ultimately bounded uniformly in* σ if there exists a positive constant K such that for all t, $x_0 \in \mathbb{R}^{n_s}$, and σ,

$$\lim_{t\to\infty} \mathbb{E}_{x_0}\|x(t)\|^2 \le K. \tag{23}$$

Definition 2 The process $x(t)$ is said to be *exponentially mean square ultimately bounded uniformly in* σ if there exist positive constants K, c, and α such that for all t, $x_0 \in \mathbb{R}^{n_s}$, and for all σ

$$\mathbb{E}_{x_0}\|x(t)\|^2 \le K + c\|x_0\|^2 e^{-\alpha(t)}. \tag{24}$$

The Itô differential generator $\mathcal{L}$ for the smooth function $V(t,x)$ is given by

$$\mathcal{L}V(t,x) = \frac{\partial V(t,x)}{\partial t} + \sum_j F_j(t,x)\frac{\partial V(t,x)}{\partial x_j} + \frac{1}{2}\sum_{i,j}\left[G_\sigma G_\sigma^T\right]_{ij}(t,x)\frac{\partial^2 V(t,x)}{\partial x_j x_i}, \tag{25}$$

where $\left[G_\sigma G_\sigma^T\right]_{ij}$ is the i^{th} row and j^{th} column element in the $n_s \times n_s$ matrix $G_\sigma G_\sigma^T$. The following lemma is a special case of that proved by Miyahara [35].

Lemma 1 Let $x(t)$ be the process defined by the solution to (22). Let $V(t,x)$ be of class $\mathcal{C}^2$ with respect to x, of class $\mathcal{C}^1$ with respect to t, and bounded from below. If for some nonzero constants k_1, k_2, $\mathcal{L}V(t,x) \le k_1 - k_2V(t,x)$ then $\mathbb{E}_{x_0}V(t,x(t)) \le V(0,x_0)e^{-k_2t} + \frac{|k_1|}{k_2}(1-e^{-k_2t})$ for all $t \ge 0$.

The following theorem extends the ultimate boundedness results of Miyahara to switched Itô differential equation (22).

Theorem 1 Let $x(t)$ be the process defined by the solution to (22), and let $V(t,x)$ be a function of class $\mathcal{C}^2$ with respect to x, and class $\mathcal{C}^1$ with respect to t. If,

1) $-\alpha_1 + c_1\|x\|^2 \le V(t,x)$ for real α and $c_1 > 0$; and
2) $\mathcal{L}V(t,x) \le \beta_\sigma - c_2V(t,x)$ where $\mathcal{L}$ is the differential generator of the Itô process as defined in (25), for real β_σ and $c_2 > 0$, and all switch states σ;

then the process $x(t)$ is mean square ultimately bounded uniformly in σ. Suppose in addition $V(t,x) \le c_3\|x\|^2 + \alpha_2$, then the process $x(t)$ is exponentially mean square ultimately bounded uniformly in σ.

Proof: From Lemma 1,

$$\mathbb{E}_{x_0}V(t,x(t)) \le V(0,x_0)e^{-c_2t} + \frac{|\beta_\sigma|}{c_2}(1-e^{-c_2t}). \tag{26}$$

Therefore, $\lim_{t\to\infty}(\mathbb{E}_{x_0}V(t,x(t))) \to \frac{|\beta_\sigma|}{c_2}$. Since $-\alpha_1 + c_1\|x\|^2 \le V(t,x)$, we have $\|x\|^2 \le \frac{V(t,x)}{c_1} + \frac{\alpha_1}{c_1}$. Therefore it follows that

$$\mathbb{E}_{x_0}\|x(t)\|^2 \le \frac{1}{c_1}\mathbb{E}_{x_0}V(t,x(t)) + \frac{\alpha_1}{c_1} \to \frac{|\beta_\sigma|}{c_1c_2} + \frac{\alpha_1}{c_1} \tag{27}$$

as $t \to \infty$. From which it follows that $\lim_{t\to\infty}\mathbb{E}_{x_0}\|x(t)\|^2 \le K$ with $K = \max_\sigma(\frac{|\beta_\sigma|}{c_1c_2} + \frac{\alpha_1}{c_1})$. Therefore the process $x(t)$ is mean square ultimately bounded uniformly in σ. If in addition $V(0,x_0) \le c_3\|x_0\|^2 + \alpha_2$, then from (26) and (27), $\mathbb{E}_{x_0}\|x(t)\|^2 \le \frac{c_3}{c_1}\|x_0\|^2e^{-c_2t} + \frac{\alpha_2}{c_1} + K$. Hence, the process $x(t)$ is exponentially mean square ultimately bounded uniformly in σ. ■

B. Analysis of Stability

This section establishes the stability of Gaussian process based MRAC using Algorithm 1. Let $\sigma(t) \in \mathbb{N}$ denote a switching index that increments every time the set $\mathcal{BV}$ is modified. When the σ^{th} system is active, the mean function estimate evaluated using the current set $\mathcal{BV}(\sigma)$, is denoted $\hat{m}^\sigma$. The results presented assume that the uncertainty can be expressed as

$$\Delta(z) \sim m(z(t)) + G_\sigma(t,z(t))d\xi(t), \tag{28}$$

where $\xi(t)$ is a (zero-mean) Wiener process, and G_σ is a linear operator associated to the covariance kernel $k(z,z')$ (see Appendix 2 of [50]). Then, for a given σ, the tracking error dynamics of (8) are

$$de = Ae\,dt + B\left(\epsilon_m^\sigma(z)\,dt - G_\sigma(t,z(t))\,d\xi\right), \tag{29}$$

where $\nu_{ad}(z) \sim \mathcal{GP}(\hat{m}^\sigma(z), k(z,z'))$, and $\epsilon_m^\sigma(z) = \hat{m}^\sigma(z) - m(z)$. For the sake of brevity, we drop the time and state dependency of $G_\sigma(t,z(t))$ in the remaining section.

To prove stability, we need to bound $\|\Delta(z) - \hat{m}^\sigma(z)\|$. Consider the zero-mean Gaussian process $\{G_\sigma\,d\xi(t)\}_{t\in T}$, and define a pseudometric $d_G(t,s) = \sqrt{\mathbb{E}[\|G_\sigma\,d\xi(t) - G_\sigma\,d\xi(s)\|]^2}$

on T. Let $N(T,d,\upsilon)$ be the υ-covering number of the space; then the υ-entropy of the space (T,d) is given by $H(T,d,\upsilon) = \log N(T,d,\upsilon)$. Let $D(T)$ be the diameter of the space T with respect to the metric d_G. Then the following bound holds:

$$\mathbb{E}\sup_{t\in T} G_\sigma\, d\xi(t) \leq C\int_0^{D(T)} H^{1/2}(T,d,\upsilon)d\upsilon. \qquad (30)$$

The RHS is the *Dudley entropy integral*; if it is finite, then the process $G_\sigma\, d\xi(t)$ is bounded a.s [30]. In the case that the covariance kernel is Gaussian, and the evolution of the system is over a compact domain, the integral can be shown to be finite, and therefore $\|G_\sigma\, d\xi(t)\| \leq c_3$ a.s., for some $c_3 \in \mathbb{R}_+$.

Lemma 2 Let $\Delta(z)$ be represented by a Gaussian process as in (11), $\hat{m}^\sigma(z)$ be defined as in (19), and let $\|z(t)\|$ be bounded for all time. Then $\|\Delta(z) - \hat{m}^\sigma(z)\|$ is almost surely (a.s.) bounded for each σ.

Proof: Let $\mathcal{BV}$ represent the set of basis vectors selected by either the OP or the KL variants of Algorithm 1 for the switch state σ. Dropping the σ dependence for brevity and using (14) and (28), we have

$$\begin{aligned}&\|\Delta(z) - \hat{m}^\sigma(z)\|\\ &= \left\|\sum_{i\in\mathcal{I}}\beta_i k(z_i,z) - \sum_{j\in\mathcal{BV}}\hat{\alpha}_j k(z_j,z) + Gd\xi(z)\right\|\\ &\leq \left\|\sum_{j\in\mathcal{BV}}(\beta_i - \hat{\alpha}_i)k(z_i,z) + \sum_{i\in\mathcal{I}\backslash\mathcal{BV}}\beta_i k(z_i,z) + Gd\xi(z)\right\|\end{aligned}$$

where $\mathcal{I} = \{1,\dots,\tau\}$. Convert the norm of the three terms into the sum of three norms using the triangle inequality. Since $\sup_x |k(x,x_i)| = 1$, and because $\hat{\alpha}$ is a coordinate vector minimizing $\|\Delta(\bar{z}_i) - \hat{\alpha}k(\bar{z}_i,\bar{z})\|_{\mathcal{H}}$ via the Representer theorem [53], the first term is bounded by $c_1 := |\mathcal{BV}(\sigma)|\max_j\|\beta_j - \hat{\alpha}_j\|$. Due to the exponential decay properties of the Gaussian covariance kernel and the class of functions the mean arises from (12), the second term $c_2 := \left\|\sum_{i\in\mathcal{I}\backslash\mathcal{BV}}\beta_i k(z_i,z)\right\|$ is bounded. Since $\|Gd\xi\|$ is a.s. bounded above by a constant c_3 due to (30), the lemma follows. ■

The next Lemma shows that because Algorithm 1 adds or removes kernels from the basis vector set to keep a metric of the representation error bounded, $\epsilon_m^\sigma(z)$ is bounded.

Lemma 3 Let $\Delta(z)$ be representable by a GP as in (11), $\hat{m}^\sigma(z)$ be an online learned GP representation of the uncertainty as in (19) learned based on a sparse subset $\mathcal{BV}$ of the data selected by either the OP or the KL variants of Algorithm 1 for the switch state σ, and $m(z)$ be the mean of the GP with the full dataset. Then $\epsilon_m^\sigma(z) = \hat{m}^\sigma(z) - m(z)$ is bounded.

Proof: From Equation (22) in [14] and from the nonparametric Representer theorem (see Theorem 4 of [53])

$$\|\epsilon_m^\sigma(z)\| = \frac{\|\Delta(z) - \hat{m}^\sigma(z)\|}{\omega^2}\|k_{\tau+1}^* - k_{z_{\tau+1}}^T K_\sigma^{-1}k_{z_{\tau+1}}\|, \qquad (31)$$

where $K_\sigma := K(\mathcal{BV}(\sigma),\mathcal{BV}(\sigma))$. The term $\|k_{\tau+1}^* - k_{z_{\tau+1}}^T K_\tau^{-1}k_{z_{\tau+1}}\|$ is bounded above by ϵ_{tol} due to Algorithm 1 (see [14]). Using Lemma 2 it follows that

$$\|\epsilon_m^\sigma(z)\| \leq \frac{\bar{c}}{\omega^2}\epsilon_{tol}, \qquad (32)$$

where $\bar{c} := c_1 + c_2 + c_3$, and c_1, c_2, c_3 are defined in the proof of Lemma 2. ■

The boundedness of the tracking error can now be proven.

Theorem 2 Consider the system in (1), the control law of (2) and (6), and assume that the uncertainty $\Delta(z)$ follows a Gaussian process (11). Let $\mathcal{B}_\alpha$ be the largest compact ball in the domain $\mathcal{D}$ over which solutions to (1), and let $\beta > 0$. If there exists a set $\Theta_\beta = \{e|V(e) \leq \beta\}$ a.s., a constant $\delta = \left(\frac{2\beta}{\lambda_{\min}(P)}\right)^{\frac{1}{2}}$, and if $r(t)$ is such that the state x_{rm} of the bounded input bounded output reference model in (5) remains bounded in the compact ball $\mathcal{B}_m = \{x_{rm} : \|x_{rm}\| \leq m_{rm}\}$ such that $m_{rm} \leq \alpha - \delta$ holds for all $t \geq 0$, then Algorithm 1 and the adaptive signal $\nu_{ad}(z) = \hat{m}^\sigma(z)$ guarantee that the system is mean square uniformly ultimately bounded in probability a.s.

Proof: Let $V(e(t)) = \frac{1}{2}e^T(t)Pe(t)$ be the stochastic Lyapunov candidate, where $P > 0$ satisfies the Lyapunov equation (9). Note that the Lyapunov candidate is bounded above and below by a quadratic term since $\frac{1}{2}\lambda_{\min}(P)\|e\|^2 \leq V(e) \leq \frac{1}{2}\lambda_{\max}(P)\|e\|^2$. The Itô differential of the Lyapunov candidate along the solution of (29) for the σ^{th} system is

$$\begin{aligned}\mathcal{L}V(e) &= \sum_i \frac{\partial V(e)}{\partial e_i}Ae_i + \frac{1}{2}\sum_{i,j}\left[BG_\sigma(BG_\sigma)^T\right]_{ij}\frac{\partial^2 V(e)}{\partial e_j \partial e_i}\\ &= -\frac{1}{2}e^TQe + e^TPB\epsilon_m^\sigma(z) + \frac{1}{2}\text{tr}\left(BG_\sigma(BG_\sigma)^TP\right).\end{aligned}$$

Let $c_1 := \frac{1}{2}\|P\|\|BG_\sigma\|^2$ and $c_2 := \|PB\|$, then

$$\mathcal{L}V(e) \leq -\frac{1}{2}\lambda_{\min}(Q)\|e\|^2 + c_2\|e\|\|\epsilon_m^\sigma(z)\| + c_1. \qquad (33)$$

From Lemma 3 $\|\epsilon_m^\sigma(z)\| \leq \|\Delta(z) - \hat{m}^\sigma(z)\|\epsilon_{tol}$, and from Lemma 2 $\|\epsilon_m^\sigma(z)\| \leq c_3\epsilon_{tol}$ a.s. Therefore outside of the set $\Theta_\gamma = \{\|e\| \geq \frac{c_2c_3\epsilon_{tol} + \sqrt{c_2^2c_3^2\epsilon_{tol}^2 + 2\lambda_{\min}(Q)c_1}}{\lambda_{\min}(Q)}\}$, $\mathcal{L}V(e) \leq 0$ a.s. Let $\beta = \max_{e\in\theta_\gamma}V(e)$ and define the compact set $\Theta_\beta = \{e : \mathcal{L}V(e) \leq \beta\}$, Note that $\Theta_\gamma \subseteq \Theta_\beta$ with $\mathcal{L}V(e) < 0$ a.s. outside of Θ_β. Furthermore, note that $\frac{1}{2}\lambda_{\min}(P)\|e\|^2 \leq V(e)$ and define $\delta = \left(\frac{2\beta}{\lambda_{\min}(P)}\right)^{\frac{1}{2}}$. Then, with $r(t)$ such that x_{rm} remains bounded within $\mathcal{B}_m$, $x(t) \in \mathcal{D}$ a.s. and the solution to (1) holds. Since this is true for all σ, and because Algorithm 2 guarantees that σ does not switch arbitrarily fast, by Theorem 1, (29) is mean square uniformly ultimately bounded inside of this set a.s. [24]. ■

Remark 1 In contrast to traditional parametric adaptive control, the Lyapunov candidate does not need to be an explicit function of the adaptive element's parameters. This is because the parameters of the budgeted nonparametric mean function are discretely updated and guaranteed to be bounded over every switching interval because they either stay constant or are "renormalized" by Algorithm 2. Furthermore, note that the size of the set within which e is bounded can be reduced by reducing the representation error $\epsilon_m^\sigma(z)$ by choosing smaller values of ϵ_{tol}, or increasing $\lambda_{\min}(Q)$ by appropriately selecting K_1, K_2 in (6). Finally, in [27], it was shown that the linear independence of $\mathcal{BV}$ ensures that persistency of excitation (PE) in the state space is visible in $\mathcal{H}$. Since the KL variant of the algorithm aims to enforce this independence subject to the tolerance ϵ_{tol}, PE is never lost (ensuring $K(Z_\tau, Z_\tau)$ is invertible).

Remark 2 The assertions in Theorem 2 are stronger than those typically proven for fixed-center RBFN-MRAC [26], [52] because a domain over which the centers have been allocated is not required to be assumed, rather the operating domain $\mathcal{D}$ over which conditions to guarantee a continuous time Markov process solution to (22) are satisfied is simply required to exist, and could be as large as $\mathcal{R}^n$. Furthermore, mean square boundedness is shown in a stochastic setting [24], which is stronger than uniform ultimate boundedness in a deterministic setting traditionally shown for RBFN-MRAC.

The following corollary extends the above result to the case when an exact representation of the mean function is available (i.e. $\epsilon_m(z) = 0$), such as might be possible in cases where the uncertainty can be represented accurately using a tabular representation, and is equivalent to assuming that the allowable function class in (12) contains a finite number of kernels. This corollary is also applicable to robust control of linear systems where the uncertainty is characterized by a GP whose mean has been identified using recorded data.

Corollary 3 Consider the system in (1), the reference model in (5), and the control law of (2) and (6). Let the uncertainty $\Delta(z)$ follows a Gaussian process as in (11) with a fixed number of kernels, then Algorithm 1 and the adaptive signal $\nu_{ad}(z) = m(z)$ guarantees that $e(t)$ is exponentially mean squared ultimately bounded.

Proof: With $\epsilon_m(z) = 0$, (33) reduces to

$$\mathcal{L}V(e) \leq -\frac{\lambda_{\min}(Q)}{2\lambda_{\max}(P)}V(e) + c_4. \tag{34}$$

The result follows from Theorem 1 with constant σ (no switching). ■

The following corollary shows that if $\epsilon_m^\sigma(z) = \hat{m}^\sigma(z) - m(z)$ tends to zero as $t \to \infty$, then the closed loop system is uniformly exponentially ultimately bounded a.s. This result is useful in the case where the mean $m(z)$ is exactly a finite linear combination of l kernels and an online GP learning algorithm (such as Algorithm 2 with $p_{\max} \geq l$) eventually learns an exact representation of the mean function given sufficient samples.

Corollary 4 Consider the system in (1), the reference model in (5), the control law of (2) and (6). Let the uncertainty $\Delta(z)$ follow a Gaussian process as in (11). Then Algorithm 1 and the adaptive signal $\nu_{ad}(z) = \hat{m}^\sigma(z)$ with $\lim_{t\to\infty} \hat{m}^\sigma(z) = m(z)$ guarantees that $e(t)$ is mean squared ultimately bounded.

Proof: Since $\epsilon_m(z) \to 0$ as $t \to \infty$ we have that (33) approaches (34). The result now follows from Theorem 1. ■

V. Trajectory Tracking in Presence of Wing Rock Dynamics in an unknown operating domain

Modern highly swept-back or delta wing fighter aircraft are susceptible to lightly damped oscillations in roll known as "wing rock". Wing rock often occurs at conditions encountered during landing [51]. Wing rock is a highly nonlinear phenomena, and when it is present, a baseline linear controller designed assuming known linear dynamics may not be able to guarantee acceptable trajectory tracking performance. Therefore, an adaptive controller capable of mitigating effects of wing rock is critical for safe landing. In this section the GP-MRAC approach is compared with fixed-center RBFN-MRAC in numerical simulations. Note that the goal is not to present a highly tuned adaptive controller for wing rock, several other authors have already done that assuming known operating domain or known bases of $\Delta(x)$ [13], [23], [36], [56], [62]. Rather, the goal is to test the performance when the assumption of known operating domain (needed by fixed-center RBFN-MRAC to pre-allocate centers) or known bases of uncertainty are violated, forcing the controller to adapt to unknown operating conditions. In that sense, one would be tempted to compare the results to [27]. However, the results here are not directly comparable to any of the aforementioned papers because those papers did not consider stochasticity in the dynamics and measurement noise, as is done here.

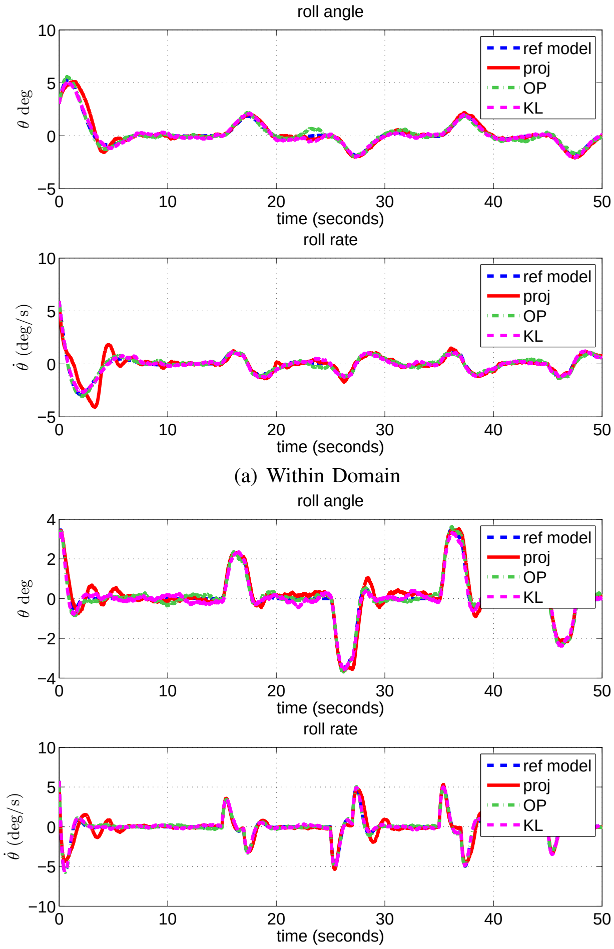
Let θ denote the roll attitude of an aircraft, p denote the roll rate and δ_a denote the aileron control input. Then a model for wing rock dynamics is [36]

$$\begin{aligned}\dot{\theta} &= p \\ \dot{p} &= L_{\delta_a}\delta_a + \Delta(x),\end{aligned} \tag{35}$$

where $L_{\delta_a} = 3$ and $\Delta(x)$ is a model of the wing rock dynamics and is assumed to be unknown to the controller. Unlike [51] and [36] who assume a deterministic model with a known basis for $\Delta(x)$, we assume that the uncertainty arises from a distribution over functions approximated by a GP with variance ω_n^2 and mean

$$\bar{\Delta}(x) = W_0^* + W_1^*\theta + W_2^*p + W_3^*|\theta|p + W_4^*|p|p + W_5^*\theta^3. \tag{36}$$

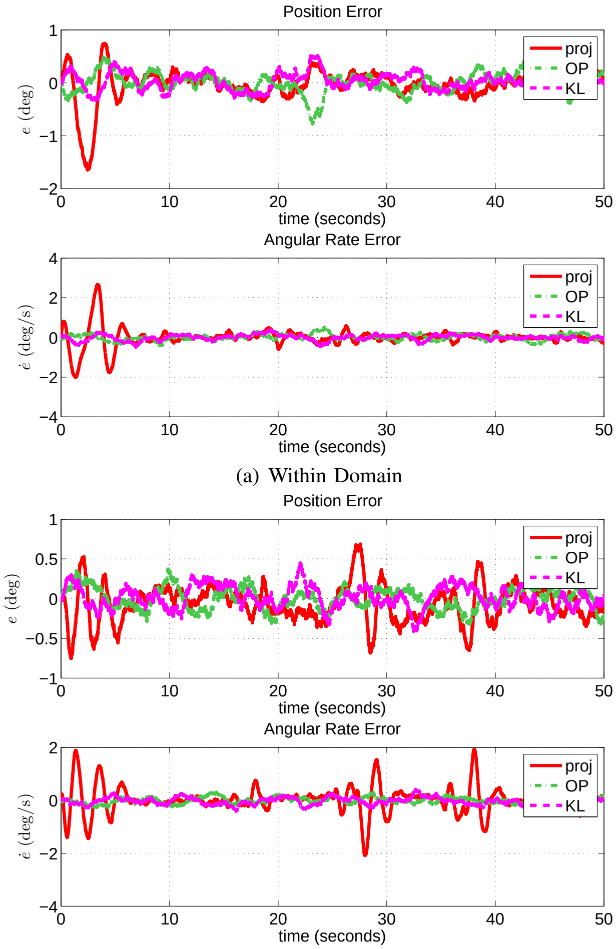
The parameters for the mean function are motivated from [36] $W_1^* = 0.2314, W_2^* = 0.6918, W_3^* = -0.6245, W_4^* = 0.0095, W_5^* = 0.0214$. In addition, a trim error is introduced by setting $W_0^* = 0.8$. The mean function is unknown to the controller, and the chosen inversion model has the form $\nu = \frac{1}{L_{\delta_a}}\delta_a$. This choice results in the mean of the modeling uncertainty of (4) being given by $\Delta(x)$. Stochasticity is introduced by adding Gaussian white noise of variance $\omega_n = 0.01$ to the states. The adaptive controller uses the control law of (6). The gain for the linear part of control law (ν_{pd}) was set to relatively low values $[1.2, 1.2]$ to better characterize the effect of adaptation/learning. A second order reference model with a natural frequency of 1 rad/sec and damping ratio of 0.5 is used [27]. The simulation uses a time-step of 0.05 sec. The maximum number of points to be stored ($p_{\max}$) was arbitrarily set to 100, and points were selected for storage based on both the oldest point (OP) and KL divergence (KL) (see Section III-B). The projection operator places an upper bound of 10 on individual RBF weights when using RBFN-MRAC with fixed centers, see (10). Both the case when the chosen RBFN centers cover the expected domain of operation and the case



(a) Within Domain

(b) Outside Domain

Fig. 4. Comparison of system states and reference model when using GP regression based MRAC and RBFN-MRAC with the projection operator and uniformly distributed centers over their respective domains. The state measurements are corrupted with Gaussian white noise.



(a) Within Domain

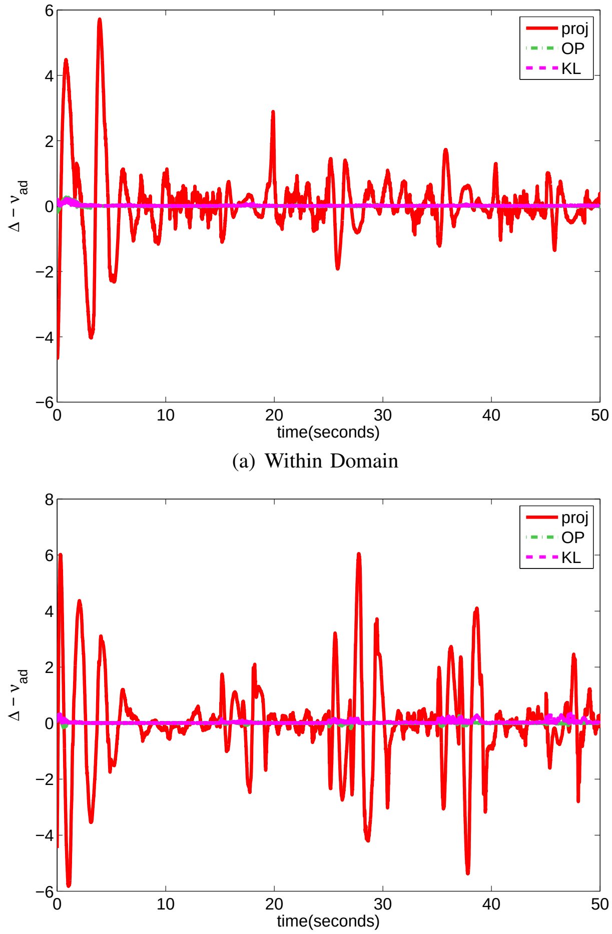
(b) Outside Domain

Fig. 5. Comparison of tracking error when using GP regression based MRAC and RBFN-MRAC with the projection operator and uniformly distributed centers over their respective domains. Compared to Fig. 4(a), RBFN-MRAC with uniformly distributed centers has higher transient tracking error than GP-MRAC because the commands drive the system out of the range over which the centers were distributed.

when the system is driven outside of the a priori expected domain of operation are examined.

A. *System Within Domain of Operation*

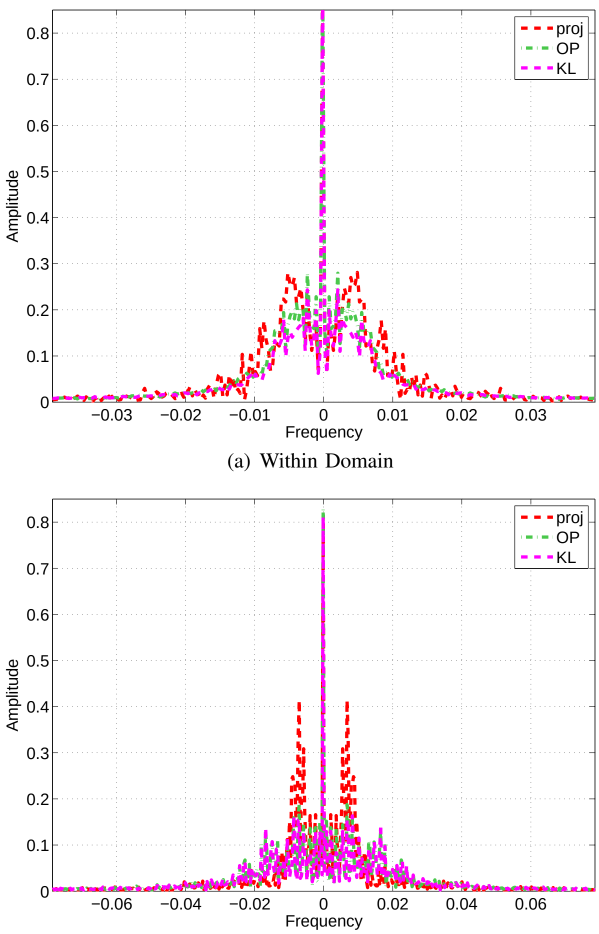
In the first case, a set of reference commands lead the system, after a short transient, to a relatively compact set in the state space, which allows for good online approximation of the uncertainty with 100 centers. The goal of this scenario is to compare GP-MRAC with fixed-center RBFN-MRAC when the domain of operation is known. Figure 4(a) compares the system states and the reference model when using GP-MRAC versus RBFN-MRAC with the projection operator and uniformly distributed centers over $[-2,2]\times[-2,2]$. While GP-MRAC performs significantly better in the transient, especially when tracking the command in $\dot{\theta}$, the long term behavior of all the controllers is similar. This is to be expected as the reference model drives the system to operate over the known domain of operation, hence RBFN-MRAC can be tuned to yield very good tracking [23], [62]. However, the poorer performance of RBFN-MRAC in the initial transient stresses the fact that transient performance guarantees of RBFN-MRAC can be lost if the system states leave the expected domain of operation, this is not the case for the nonparametric GP-MRAC approach as it selects centers online. Figure 5(a) compares the tracking error for both the controllers. The GP-MRAC system has less oscillations. Figure 6(a) compares the learned models of GP-MRAC and RBFN-MRAC. While the GP adaptive element output is almost indistinguishable from the uncertainty in presence of measurement noise, the RBFN does not accurately learn the uncertainty. The poor learning performance of RBFN with the learning law of 10 is to be expected [6], [26], [60], and Figure 6(a) clearly shows that GP-MRAC yields much better learning. 7(a) plots the energy of the spectrum of the signal $\nu_{ad}-\Delta$ for all the controllers considered. It can be seen that for the RBFN-MRAC controller the spectra for this signal has more energy at nonzero frequencies. This indicates that there are greater number of oscillations in RBFN-MRAC. Figure 8(a) shows online uncertainty tracking for the KL divergence method by plotting the posterior mean and variance at each timestep, showing good approximation error, and confidence in the estimates. Figure 9(a) shows the trajectory of the system controlled by the KL divergence method in state space.



(a) Within Domain

(b) Outside Domain

Fig. 6. Error between the adaptive element output and the actual uncertainty (the signal $\nu_{ad} - \Delta$). RBF MRAC approximation with uniformly distributed centers is significantly worse than the GP approximations.



(a) Within Domain

(b) Outside Domain

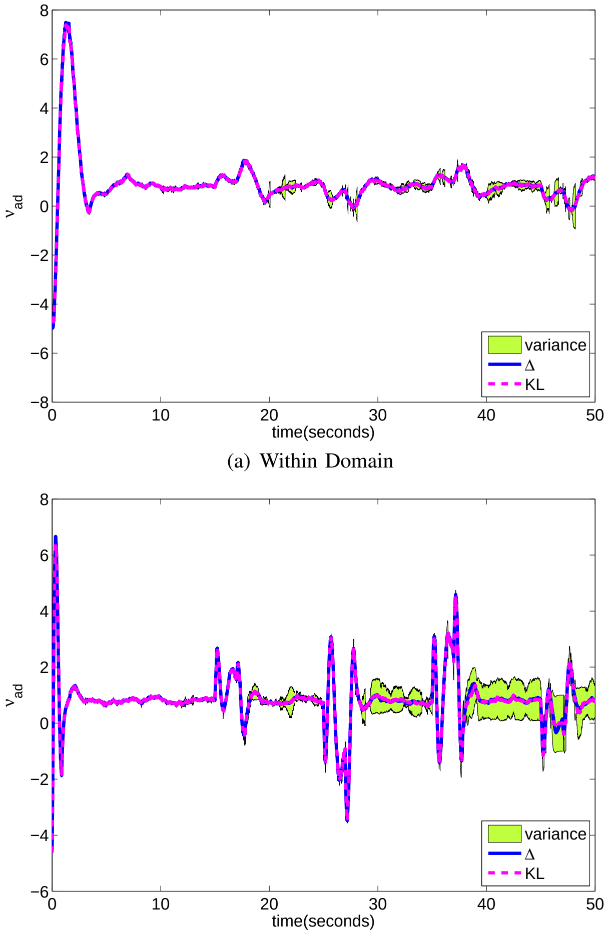
Fig. 7. Energy of the spectra of the error between the adaptive element output and the actual uncertainty. This figure quantifies the greater number of oscillations while tracking in RBF MRAC.

B. System Driven Outside Domain of Operation

The above scenario demonstrated that the baseline trajectory tracking performance of GP-MRAC is somewhat comparable to RBFN-MRAC with update law of 10 when the system stays within operating domain. However, GP-MRAC's true usefulness becomes apparent when its nonparametric form is more fully exploited. Therefore, in this second case, the system is given reference commands that drive it outside of the expected domain of operation, i.e. where RBF centers are not allocated, and RBFN-MRAC has not been tuned to yield good performance. Significant performance differences occur between the two methods. Figures 4(b) and 5(b) show that the tracking performance of GP-MRAC is superior to RBFN-MRAC, especially when tracking the $\dot{\theta}$ commands. Figure 6(b) shows that RBFN-MRAC does an even poorer job of learning the uncertainty than before, while GP-MRAC learning performance is excellent. In Figure 7(b), the harmonics associated to the oscillations become much more apparent. It can be clearly seen that there is a significant peak at nonzero frequencies for RBFN-MRAC that is not present in GP-MRAC. This indicates that the GP-MRAC controller is better able to mitigate the oscillations associated with wing rock dynamics. Figure 9(b) shows the distribution of centers chosen by the controllers. Although the KL divergence method more widely distributes the centers over the trajectory of the system in comparison to the OP method, there is little difference in the online performance between the OP and KL divergence GP-MRAC controllers. The negligible difference indicates that the most important factor to performance is the existence of centers near the current state. However, their performance differs significantly in terms of long-term learning.

C. Illustration of Long-Term Learning Effect

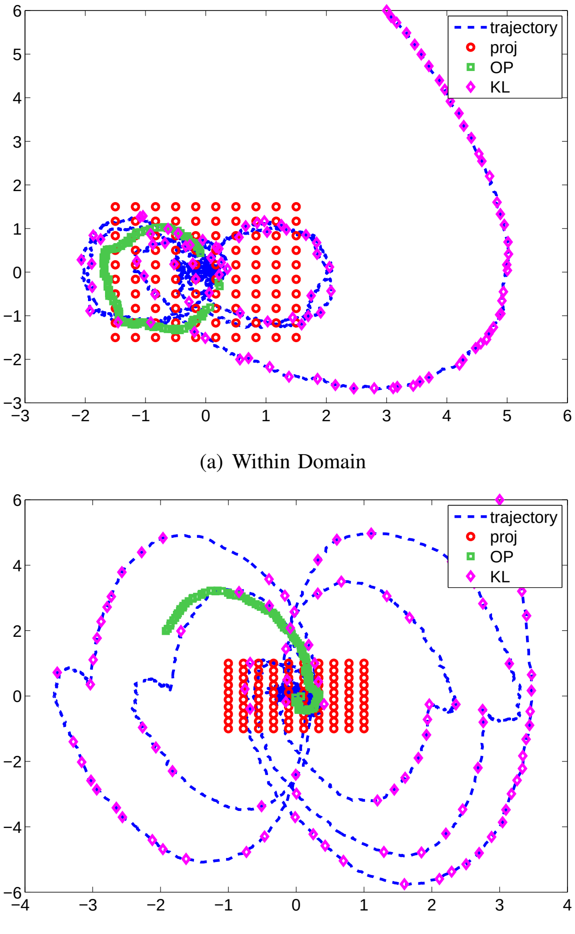
One of the most important contributions of GP-MRAC, particularly when using the KL divergence scheme, is the the introduction of long-term learning in the adaptive controller. Long-term learning here is characterized by better approximation of the uncertainty over the entire domain, a quality that is missing from RBFN-MRAC with the update law of 10. In order to illustrate this, the learned parameters of the adaptive elements of the systems are recorded at the final time T of the simulation. The systems are then reinitialized with these parameters, which are *frozen*, and the simulations are run again. Figures 10(a) and 10(b) show the uncertainty tracking results for the two scenarios mentioned above. As can be seen, the locality that drives the RBFN-MRAC updates results in a controller that doesn't capture the uncertainty



(a) Within Domain

(b) Outside Domain

Fig. 8. Online uncertainty tracking for the KL method.



(a) Within Domain

(b) Outside Domain

Fig. 9. Trajectories in the state space for both command sets.

globally. While GP-MRAC with the OP scheme does a better job in this regard, its choice of center placement means that approximation accuracy is local also. Since the KL divergence scheme chooses points to keep based on the distance between the GP models, it is able to retain global information about the uncertainty and thus achieve the best performance.

Overall, these results show that the GP-MRAC as described by Algorithm 1 is successful in capturing the uncertainty without the need for prior domain domain knowledge.

VI. Conclusion

This paper modeled the uncertainty in Model Reference Adaptive Control (MRAC) as a distribution over functions rather than via a deterministic function. This approach to uncertainty modeling is relevant to many real-world scenarios involving the presence of noise, servo chattering, or other non-smooth effects. To accurately learn the uncertain function, we used Gaussian Process adaptive elements, which leverage a powerful and robust nonparametric framework to perform inference directly in the space of functions. We extended a GP inference method to work within a preset budget, allowing its use in an online setting. These two modifications define the GP-MRAC algorithm, which was proven to be stable through the use of stochastic stability theory for switched systems. Simulations employing GP-MRAC for the control of wing rock dynamics demonstrate the efficacy of GP-MRAC as a budgeted, nonparametric adaptive control method which requires no domain knowledge of the uncertainty.

VII. Acknowledgments

This research was supported in part by ONR MURI Grant N000141110688 and NSF grant ECS #0846750.

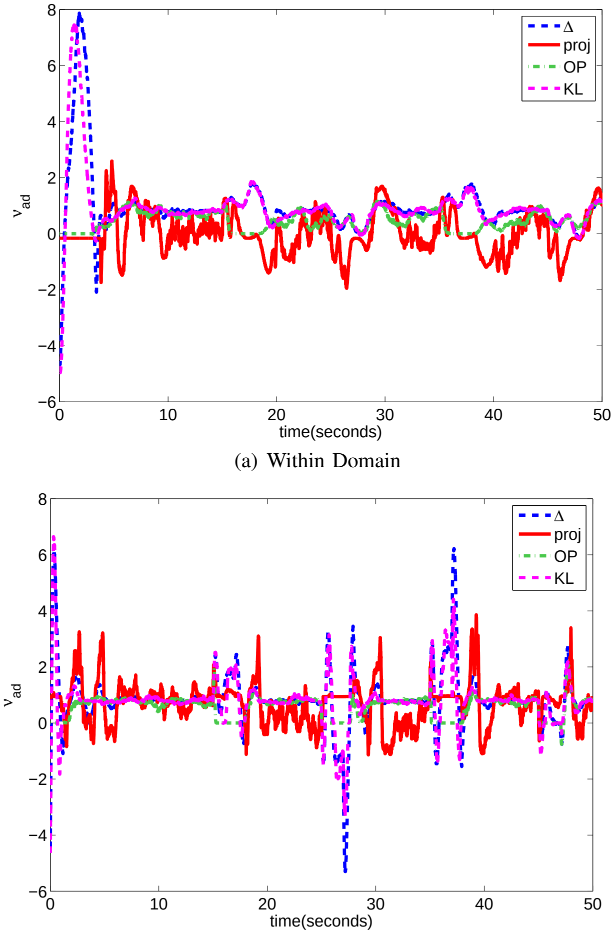
VIII. Appendix

This section outlines the implementation details of the sparse online Gaussian process algorithm [14]. At time $\tau+1$, given a new datapoint $z_{\tau+1}$, the algorithm minimizes the KL divergence between the model with the datapoint included, and the $\tau+1$ models with one datapoint deleted. To compute the updates in an online fashion, define the scalar quantities

$$q^{(\tau+1)} = \frac{y - \alpha_\tau^T k_{x_\tau}}{\omega_n^2 + k_{x_\tau}^T C_\tau k_{x_\tau} + k_\tau^*}, \tag{37}$$

$$r^{(\tau+1)} = -\frac{1}{\omega_n^2 + k_{x_\tau}^T C_\tau k_{x_\tau} + k_t^*}, \tag{38}$$

where α_τ, k_{x_τ}, and C_τ are defined in (17) and (18). Let $e_{\tau+1}$ be the $(\tau+1)$ coordinate vector, and let $T_{\tau+1}(\cdot)$ and $U_{\tau+1}(\cdot)$ denote operators that extend a τ-dimensional vector and matrix to a $(\tau+1)$ vector and $(\tau+1)\times(\tau+1)$ matrix by appending



(a) Within Domain

(b) Outside Domain

Fig. 10. Comparison of uncertainty tracking after the models are learned and the weights are frozen. As can be seen, the locality of the proj operator and OP controllers precludes true learning upon the domain.

zeros to them. The GP parameters can be solved for recursively using the equations

$$\begin{aligned} \alpha_{\tau+1} &= T_{\tau+1}(\alpha_\tau) + q^{(\tau+1)} s_{\tau+1}, \\ C_{\tau+1} &= U_{\tau+1}(C_\tau) + r^{(\tau+1)} s_{\tau+1} s_{\tau+1}^T, \\ s_{\tau+1} &= T_{\tau+1}(C_\tau k_{x_{\tau+1}}) + e_{\tau+1}. \end{aligned} \tag{39}$$

The inverse of the Gram matrix, dentoed by Q, needed to solve for $\gamma_{\tau+1}$ is updated online through the equation

$$\begin{aligned} Q_{\tau+1} &= U_{\tau+1}(Q_\tau) \\ &\quad + \gamma_{\tau+1}^{-1} \left(T_{\tau+1}(\hat{e}_{\tau+1}) - e_{\tau+1}\right) \left(T_{\tau+1}(\hat{e}_{\tau+1}) - e_{\tau+1}\right)^T, \end{aligned} \tag{40}$$

where $\hat{e}_{\tau+1} := Q_\tau k_{z_{\tau+1}}$. Finally, in order to delete an element, one computes the model parameters with the $(\tau+1)$-th point, and chooses the basis vector with the smallest score measure, given by

$$\epsilon_i = \frac{|\alpha_{\tau+1}(i)|}{Q_{\tau+1}(i,i)}. \tag{41}$$

Algorithm 2 The budgeted sparse Gaussian process algorithm

```
while new measurements $(z_{\tau+1}, y_{\tau+1})$ are available do
  Compute $q^{(\tau+1)}$, $r^{(\tau+1)}$, $k^*_{\tau+1}$, $k_{z_{\tau+1}}$, $\hat{e}_{\tau+1}$ and $\gamma_{\tau+1}$.
  if $\gamma_{\tau+1} < \epsilon_{tol}$ then
    Perform a reduced update, using $\hat{e}_{\tau+1}$ in (39) without extending the length of the parameters $\alpha$ and $C$.
  else
    Perform the update in (39) using $e_{\tau+1}$. Add the current input to the $\mathcal{BV}$ set, and compute the Gram matrix inverse using (40).
    if $|\mathcal{BV}| > p_{\max}$ then
      Compute scores for the candidate $\mathcal{BV}$'s using (41), find the vector corresponding to the lowest score, and delete it using (42).
    end if
  end if
end while
```

Let ι be the basis vector chosen to be discarded by the score (41). Then the deletion equations are given by

$$\begin{aligned} \hat{\alpha} &= \hat{\alpha}^{\neg\iota} - \alpha^* \frac{Q^*}{q^*}, \\ \hat{C} &= C^{\neg\iota} + c^* \frac{Q^* Q^{*T}}{q^{*2}} - \frac{1}{q^*}\left[Q^* C^{*T} + C^* Q^{*T}\right], \\ \hat{Q} &= Q^{\neg\iota} - \frac{Q^* Q^{*T}}{q^*}, \end{aligned} \tag{42}$$

where α^* is the ι^{th} component in the vector $\alpha_{\tau+1}$, and $\alpha^{\neg\iota}$ represents the remaining vector. Similarly, $C^{\neg\iota}$ ($Q^{\neg\iota}$) represents the $\tau \times \tau$ submatrix in the $(\tau+1) \times (\tau+1)$ matrix $C_{\tau+1}$ ($Q_{\tau+1}$) associated to the basis vectors being kept, c^* (q^*) represents the (ι, ι) index into the matrix chosen by the score measure, and C^* (Q^*) is the remaining τ-dimensional column vector. Using the above equations, the budgeted sparse GP algorithm is summarized by Algorithm 2.

References

[1] T. Alpcan. Dual control with active learning using Gaussian process regression. Arxiv 1105.2211v1, Technical University Berlin, Berlin, 2011.

[2] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, May 1950.

[3] K. Aström and B. Wittenmark. *Adaptive Control*. Addison-Weseley, Readings, 2nd edition, 1995.

[4] C. Bernard and J. Slotine. Adaptive control with multiresolution bases. In *Proceedings of the 36th IEEE Conference on Decision and Control, 1997*, volume 4, pages 3884 –3889, Dec 1997.

[5] C. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[6] S. Boyd and S. Sastry. Necessary and sufficient conditions for parameter convergence in adaptive control. *Automatica*, 22(6):629–639, 1986.

[7] A. Calise, N. Hovakimyan, and M. Idan. Adaptive output feedback control of nonlinear systems using neural networks. *Automatica*, 37(8):1201–1211, 2001. Special Issue on Neural Networks for Feedback Control.

[8] J. Candela and C. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

[9] M. Cannon and J. Slotine. Space-frequency localized basis function networks for nonlinear system estimation and control. *Neurocomputing*, 9(3):293 – 342, 1995. Control and Robotics, Part III.

[10] Y. H. Chen. Adaptive robust control of uncertain systems with measurement noise. *Automatica*, 28(4):715–728, 1992.

[11] G. Chowdhary, J. How, and H. Kingravi. Model reference adaptive control using nonparametric adaptive elements. In *Conference on Guidance Navigation and Control*, Minneapolis, MN, August 2012. AIAA. Invited.

[12] G. Chowdhary and E. Johnson. Concurrent learning for convergence in adaptive control without persistency of excitation. In *Proceedings of the 49th IEEE Conference on Decision and Control*, pages 3674–3679, 2010.

[13] G. Chowdhary and E. N. Johnson. A singular value maximizing data recording algorithm for concurrent learning. In *American Control Conference*, San Francisco, CA, June 2011.

[14] L. Csató and M. Opper. Sparse on-line Gaussian processes. *Neural Computation*, 14(3):641–668, 2002.

[15] M. P. Deisenroth. *Efficient Reinforcement Learning using Gaussian Processes*. KIT Scientific Publishing, Karlsruhe, Germany, 1 edition, 2010.

[16] W. Durham. Constrained control allocation. *AIAA Journal of Guidance, Control, and Dynamics*, 16:717–772, 1993.

[17] A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian Data Analysis*. Chapman and Hall, 2nd edition, 2004.

[18] M. Gredilla, J. Candela, C. Rasmussen, and A. Figueiras-Vidal. Sparse spectrum Gaussian process regression. *Journal of Machine Learning Research*, 11:1865–1881, 2010.

[19] P. A. Ioannou and J. Sun. *Robust Adaptive Control*. Prentice-Hall, Upper Saddle River, 1996.

[20] E. Johnson and S. Kannan. Adaptive trajectory control for autonomous helicopters. *Journal of Guidance Control and Dynamics*, 28(3):524–538, May 2005.

[21] S. Kannan. *Adaptive Control of Systems in Cascade with Saturation*. PhD thesis, Georgia Institute of Technology, Atlanta Ga, 2005.

[22] H. K. Khalil. *Nonlinear Systems*. Macmillan, New York, 2002.

[23] E. Kharisov and N. Hovakimyan. Application of l1 adaptive controller to wing-rock. In *AIAA Infotech@Aerospace*, april 2010.

[24] R. Khasminksii. *Stochastic Stability of Differential Equations*. Springer, Berlin, Germany, 2 edition, 2012.

[25] N. Kim. *Improved Methods in Neural Network Based Adaptive Output Feedback Control, with Applications to Flight Control*. PhD thesis, Georgia Institute of Technology, Atlanta GA, 2003.

[26] Y. H. Kim and F. Lewis. *High-Level Feedback Control with Neural Networks*, volume 21 of *Robotics and Intelligent Systems*. World Scientific, Singapore, 1998.

[27] H. A. Kingravi, G. Chowdhary, P. A. Vela, and E. N. Johnson. Reproducing kernel Hilbert space approach for the online update of radial bases in neuro-adaptive control. *IEEE Transactions on Neural Networks and Learning Systems*, 23(7):1130 –1141, July 2012.

[28] J. Ko and D. Fox. GP-BayesFilters: Bayesian filtering using Gaussian process prediction and observation models. *Autonomous Robots*, 27:75–90, 2009.

[29] J. Ko, D. Klein, D. Fox, and D. Haehnel. Gaussian processes and reinforcement learning for identification and control of an autonomous blimp. In *IEEE International Conference on Robotics and Automation*, pages 742–747. IEEE, 2007.

[30] V. Koltchinskii. *Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems: École dÉté de Probabilités de Saint-Flour XXXVIII-2008*, volume 2033. Springer, 2011.

[31] G. Kreisselmeier and K. S. Narendra. Stable model reference adaptive control in the presence of bounded disturbances. *IEEE Transactions on Automatic Control*, AC-27(6):1169–1175, 1982.

[32] H. J. Kushner. *Stochastic Stability and Control*. Academic Press, New York, NY, 1967.

[33] F. L. Lewis. Nonlinear network structures for feedback control. *Asian Journal of Control*, 1:205–228, 1999. Special Issue on Neural Networks for Feedback Control.

[34] W. Liu, J. C. Principe, and S. Haykin. *Kernel Adaptive Filtering: A Comprehensive Introduction*. Wiley, Hoboken, New Jersey, 2010.

[35] Y. Miyahara. Ultimate boundedness of the systems governed by stochastic differential equations. *Nagoya Math Journal*, 47:111–144, 1972.

[36] M. M. Monahemi and M. Krstic. Control of wingrock motion using adaptive feedback linearization. *Journal of Guidance Control and Dynamics*, 19(4):905–912, August 1996.

[37] K.-R. Müller, S. Mika, G. Rätsch, S. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–202, 2001.

[38] R. Murray-Smith and D. Sbarbaro. Nonlinear adaptive control using non-parametric Gaussian process prior models. In *15th Triennial World Congress of the International Federation of Automatic Control*. International Federation of Automatic Control (IFAC), 2002.

[39] R. Murray-Smith, D. Sbarbaro, C. Rasmussen, and A. Girard. Adaptive, cautious, predictive control with Gaussian process priors. In P. V. de Hof, B. Wahlberg, and S. Weiland, editors, *13th IFAC Symposium on System Identification*, pages 1195–1200. Elsevier Science, 2003.

[40] F. Nardi. *Neural Network based Adaptive Algorithms for Nonlinear Control*. PhD thesis, Georgia Institute of Technology, School of Aerospace Engineering, Atlanta, GA, 2000.

[41] K. Narendra. Neural networks for control theory and practice. *Proceedings of the IEEE*, 84(10):1385 –1406, oct 1996.

[42] K. S. Narendra and A. M. Annaswamy. Robust adaptive control in the presence of bounded disturbances. *IEEE Transactions on Automatic Control*, AC-31(4):306–315, 1986.

[43] K. S. Narendra and A. M. Annaswamy. *Stable Adaptive Systems*. Prentice-Hall, Englewood Cliffs, 1989.

[44] D. Nguyen-Tuong and J. Peters. Using model knowledge for learning inverse dynamics. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 2677 –2682, may 2010.

[45] J. Park and I. Sandberg. Universal approximation using radial-basis-function networks. *Neural Computatations*, 3:246–257, 1991.

[46] D. Patino and D. Liu. Neural network based model reference adaptive control system. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 30(1):198–204, 2000.

[47] J.-B. Pomet and L. Praly. Adaptive nonlinear regulation: estimation from the lyapunov equation. *IEEE Transactions on Automatic Control*, 37(6):729 –740, jun 1992.

[48] C. Rasmussen and Z. Ghahramani. Occam's razor. *Advances in Neural Information Processing Systems*, 2000.

[49] C. Rasmussen and M. Kuss. Gaussian processes in reinforcement learning. *Advances in Neural Information Processing Systems*, 16:751–759, 2004.

[50] C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[51] A. A. Saad. *Simulation and analysis of wing rock physics for a generic fighter model with three degrees of freedom*. PhD thesis, Air Force Institute of Technology, Air University, Wright-Patterson Air Force Base, Dayton, Ohio, 2000.

[52] R. Sanner and J.-J. Slotine. Gaussian networks for direct adaptive control. *IEEE Transactions on Neural Networks*, 3(6):837 –863, nov 1992.

[53] B. Scholkopf, R. Herbrich, and A. Smola. A generalized representer theorem. In D. Helmbold and B. Williamson, editors, *Computational Learning Theory*, volume 2111 of *Lecture Notes in Computer Science*, pages 416–426. Springer Berlin / Heidelberg, 2001.

[54] B. Scholköpf and A. Smola. *Support Vector Machines, Regularization, Optimization, and Beyond*. MIT press, Cambridge, MA, USA, 2002.

[55] P. Shankar. *Self-Organizing radial basis function networks for adaptive flight control and aircraft engine state estimation*. Ph.D., The Ohio State University, Ohio, 2007.

[56] S. N. Singh, W. Yim, and W. R. Wells. Direct adaptive control of wing rock motion of slender delta wings. *Journal of Guidance Control and Dynamics*, 18(1):25–30, Feb. 1995.

[57] M. Steinberg. Historical overview of research in reconfigurable flight control. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 219(4):263–275, 2005.

[58] N. Sundararajan, P. Saratchandran, and L. Yan. *Fully Tuned Radial Basis Function Neural Networks for Flight Control*. Springer, 2002.

[59] J. A. K. Suykens, J. P. Vandewalle, and B. D. Moor. *Artificial Neural Networks for Modelling and Control of Non-Linear Systems*. Kluwer, Norwell, 1996.

[60] G. Tao. *Adaptive Control Design and Analysis*. Wiley, New York, 2003.

[61] K. Y. Volyanskyy, W. M. Haddad, and A. J. Calise. A new neuroadaptive control architecture for nonlinear uncertain dynamical systems: Beyond σ and e-modifications. *IEEE Transactions on Neural Networks*, 20(11):1707–1723, Nov 2009.

[62] T. Yucelen and A. Calise. A derivative-free model reference adaptive controller for the generic transport model. In *AIAA Guidance, Control and Navigation Conference*, Toronto, Canada, August 2010. invited.

[63] T. Zhang, S. Ge, and C. Hang. Direct adaptive control of non-affine nonlinear systems using multilayer neural networks. In *American Control Conference, 1998. Proceedings of the 1998*, volume 1, pages 515 –519 vol.1, jun 1998.