Proceedings of the 2022 30th Mediterranean Conference on Control and Automation (MED),
28 June-1 July 2022, Vouliagmeni, Greece
DOI:10.1109/MED54222.2022.9837175

Incremental Nonlinear Dynamic Inversion with Sparse Online Gaussian Processes Adaptation for Partially Unknown Systems

Dmitry Ignatyev
School of Aerospace, Transport and Manufacturing
Cranfield University
Cranfield, UK
https://orcid.org/0000-0003-3627-3740

Antonios Tsourdos
School of Aerospace, Transport and Manufacturing
Cranfield University
Cranfield, UK
https://orcid.org/0000-0002-3966-7633

***Abstract*— Sensor-based Incremental control is a recently developed family of techniques with a reduced dependency on a plant model. This approach uses measurements or estimates of current state derivatives and actuator states to linearize the dynamics with respect to the previous time moment. However, in such a formulation, the control system is sensitive to the quality of measurements or estimations. The presence of uncertainties caused by unforeseen malfunctions in measurement and/or actuation systems could provoke drastic performance degradation. The paper proposes a sensor-based Incremental Nonlinear Dynamic Inversion (INDI) control algorithm augmented with Budgeted Sparse Online Gaussian Processes Adaptation for the compensation of unknown system behaviour. INDI performs quite efficiently under design conditions. Meanwhile, GP-based direct adaptation provides not only long-term dependency learning but also noise signal filtering. The efficiency of the proposed approach is demonstrated with a longitudinal motion of a missile.**

Keywords*— Gaussian processes, Data-driven, Sensor based, Nonlinear control systems, uncertainty.

I. Introduction

Nonlinear Dynamics Inversion (NDI) and Backstepping (BS) techniques have become popular control strategies for adaptation since they can be used for global linearization of the system dynamics and control decoupling [1], [2]. Later, to make the NDI and BS controls more robust and fault-tolerant an incremental-type sensor-based form has been proposed [3], [4].

Regardless of the fact that incremental-type controllers demonstrates robustness to some failures [5], [6], these types are sensible to a quality of sensing information. Noisy measurements can cause degradation of the control system and require filtering techniques to be applied. However, filtering of the very noisy signal might cause significant delays in measurement circuit and could harm control performance as a result. Furthermore, possible failures of the sensors or actuation system might cause devastating results. Augmenting adaptation loops could tackle these issues.

Neural networks, including Radial Basis Functions (RBF), are quite popular for on-line identification and adaptive control since they are universal approximators and can match any uncertainty (for example, see. [7], [8]). RBFs have advantage, namely, they are linear-in-the-parameters, as opposed to multilayer perceptron neural networks. However, performance of former approach is significantly determined by selection of the RBF centres. Normally, researches preallocate a fixed quantity of Gaussian RBF centres over the presumed domain [7], [9]. The system states must stay close to the location of the preallocated RBF centres because a Gaussian RBF output decays exponentially away from its centre; otherwise, the system would not be able to capture the uncertainty.

To tackle the issues mentioned above we propose to use a Gaussian Process (GP) online identification framework to estimate aircraft control derivatives and perform in-direct adaptive control loop augmenting baseline IBKS controller. GP brings promising Bayesian paradigm to adaptive control by considering the estimation as a statistical problem [10]. The efficiency of applying GP as an augmenting loop to the backstepping strategy is demonstrated in [11]. However, the authors in [11] collected data prior to the control design. Furthermore, the data should be collected on a compact set. All these peculiarities limit the controller performance. Within the proposed approach, GPs utilize a Bayesian framework to represent uncertainties as a distribution over functions. It is assumed that the uncertainty and the model follow Gaussian distributions, with the uncertainty being estimated using its mean and covariance function. One of the advantages of the proposed method is that it does not require prior assumptions about operating domain. From the provided flight data, GP is able to dynamically choose new kernel locations to guarantee domain coverage. Furthermore, measurement noise is explicitly handled, and parameters such as the centres of RBFs does not require pre-allocation. GP approach allows Bayesian inference to overcome shortcomings of the standard gradient based parameter update laws, e.g. lack of convergence guarantees and possible instabilities under noise presence [12], [13]. This method was

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

applied for design of direct Model Reference Adaptive Control GP-MRAC [14], where GP was used to match uncertainty to produce compensating control commands. In [15], a GP-based indirect adaptive augmentation to sensor based Incremental Backstepping control was proposed to tackle issues with failures in actuation system of an aircraft.

In the current work, we develop a GP based adaptation to incremental controller to match a wide range of uncertainties.

II. Problem Statement

We are considering nonlinear strict-feedback systems [2], which can be represented in the following form

$$\begin{aligned}\dot{\boldsymbol{x}} &= \boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\xi_1,\\ \dot{\xi}_i &= f_i(\boldsymbol{w_i}) + g_i(\boldsymbol{w}_i)\xi_{i+1}, i = 2, \dots, h-1,.\\ \dot{\xi}_h &= f_h(\boldsymbol{w}_h) + g_h(\boldsymbol{w}_h)\boldsymbol{u},\end{aligned} \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^{\boldsymbol{n}}$ and $\xi_i \in \mathbb{R}, i = 1, \dots, h$ are the system states, $u \in \mathbb{R}$ is the control input, $\boldsymbol{w}_i = [\boldsymbol{x}^T, \xi_1, \dots, \xi_i], i = 1, \dots, h$. For brevity, unless stated otherwise, whenever the subscript i is utilized, the full set, i.e., $i = 1, \dots, h$, is referred to. The nonlinear functions $\boldsymbol{F}$ and f_i are assumed to be unknown system dynamics, whereas the nonlinear functions $\boldsymbol{G}$ and g_i are assumed to be known. In this case, ξ_1 is a control input for $\boldsymbol{x}$ sub-system, and ξ_{i+1} is a control input for ξ_i sub-system.

III. Incremental Form of System Dynamics

The concept of the Incremental Nonlinear Dynamics Inversion (INDI) utilizes an idea that the system dynamics (1) can be represented in an incremental form. This can be achieved via using sensor and actuator measurements for feedback and can significantly reduce dependency on the accurate knowledge of the plant dynamics. Expanding Eq. (1) into the Taylor series around $(\boldsymbol{x}_0, \xi_{1_0}, \dots, \xi_{h_0}, \boldsymbol{u}_0)$ corresponding to the previous time moment t_0 the dynamics (1) can be expressed in the following form

$$\begin{aligned}\dot{\boldsymbol{x}} &\cong \dot{\boldsymbol{x}}_{\boldsymbol{0}} + A_x\Delta\boldsymbol{x} + B_{\xi_1}\Delta\xi_1 + \Delta_{\boldsymbol{x}},\\ \dot{\xi}_i &\cong \dot{\xi}_{i_0} + A_{\xi_i}\Delta\xi_i + B_{\xi_{i+1}}\Delta\xi_{i+1} + \Delta_{\boldsymbol{\xi_i}}, i = 2, \dots, h-1,.\\ \dot{\xi}_h &\cong \dot{\xi}_{h_0} + A_{\xi_h}\Delta\xi_h + B_u\Delta\boldsymbol{u} + \Delta_{\xi_h},\end{aligned} \tag{2}$$

where $A_x = \frac{\partial(\boldsymbol{F}(\boldsymbol{x})+\boldsymbol{G}(\boldsymbol{x})\xi_1)}{\partial \boldsymbol{x}}\Big|_{\boldsymbol{w}_1=\boldsymbol{w}_{1_0}}$, $B_{\xi_1} = \frac{\partial(\boldsymbol{F}(\boldsymbol{x})+\boldsymbol{G}(\boldsymbol{x})\xi_1)}{\partial \xi_1}\Big|_{\boldsymbol{w}_1=\boldsymbol{w}_{1_0}}$, $A_{\xi_i} = \frac{\partial(f_i(\boldsymbol{w}_i)+g_i(\boldsymbol{w}_i)\xi_{i+1})}{\partial \xi_i}\Big|_{\boldsymbol{w}_i=\boldsymbol{w}_{i_0}}$ $B_{\xi_{i+1}} = \frac{\partial(f_i(\boldsymbol{w}_i)+g_i(\boldsymbol{w}_i)\xi_{i+1})}{\partial \xi_{i+1}}\Big|_{\boldsymbol{w}_i=\boldsymbol{w}_{i_0}}$, $B_u = \frac{f_h(\boldsymbol{w}_h)+g_h(\boldsymbol{w}_h)u}{\partial u}\Big|_{\boldsymbol{w}_h=\boldsymbol{w}_{h_0},\boldsymbol{u}=\boldsymbol{u_0}}$ are partial derivatives with respect to the state vector parameters and the control input, $\Delta\boldsymbol{x} = \boldsymbol{x} - \boldsymbol{x}_{\boldsymbol{0}}$, $\Delta\xi_i = \xi_i - \xi_{i_0}$, $\Delta\boldsymbol{u} = \boldsymbol{u} - \boldsymbol{u}_{\boldsymbol{0}}$ are increments in the state derivatives and the control input. This means an approximate linearization about the previously measured signals is performed incrementally. We also included in the dynamics equations terms $\Delta_{\boldsymbol{x}}$, $\Delta_{\boldsymbol{\xi_i}}$, which are uncertainties that might be due to failed estimation/measurement of state vector derivatives and/or faulty actuation. For a sufficiently high frequency rate such that $\boldsymbol{F}(\boldsymbol{x}) + \boldsymbol{G}(\boldsymbol{x})\xi_1$, $f_i(\boldsymbol{w_i}) + g_i(\boldsymbol{w}_i)\xi_{i+1}$ and $f_h(\boldsymbol{w}_h)$ do not change significantly during sampling time, the following approximations are assumed $\Delta\boldsymbol{x} \ll \Delta\dot{\boldsymbol{x}}$, $\Delta\boldsymbol{x} \ll \Delta\xi_i$; $\Delta\xi_i \ll \Delta\dot{\xi}_i, \Delta\xi_i \ll \Delta\xi_{i+1}$. Such assumptions become possible for a real plant because the control inputs ξ_i, $\boldsymbol{u}$ directly affect the state derivatives $\dot{\boldsymbol{w}}_i$, whereas the increments in the state vector are only changed by integrating these state derivatives. In this case, the dynamics can be simplified as follows

$$\begin{aligned}\dot{\boldsymbol{x}} &\cong \dot{\boldsymbol{x}}_{\boldsymbol{0}} + B_{\xi_1}\Delta\xi_1 + \Delta_{\boldsymbol{x}},\\ \dot{\xi}_i &\cong \dot{\xi}_{i_0} + B_{\xi_{i+1}}\Delta\xi_{i+1} + \Delta_{\boldsymbol{\xi_i}}, i = 2, \dots, h-1,.\\ \dot{\xi}_h &\cong \dot{\xi}_{h_0} + B_u\Delta\boldsymbol{u} + \Delta_{\xi_h}.\end{aligned} \tag{3}$$

One of the main weak points of the sensor-based controller is that it requires precise measurements or estimations of the state vector derivatives and actuator position. Thus, the stability and performance of the system can be significantly harmed by some faulty conditions. To simulate effect of possible failure of state derivative measurement system or actuation system, we introduced the uncertainty terms $\Delta_{\boldsymbol{x}}$, $\Delta_{\boldsymbol{\xi_i}}$. GP estimation of control efficiency augmenting the Incremental Backstepping controller was shown to be efficient in tackling unknown nonlinear dynamics in actuation system [15]. In the current research, we assumed a more general uncertainty that might result not only from the failed actuator, but also from other sources, for example from failed on-board sensors. Even though a multivariate GP formulation exists [16] and can be used to approximate each $\Delta_{\boldsymbol{x}}$, $\Delta_{\boldsymbol{\xi_i}}$ by a single GP, such a formulation is very cumbersome and corresponds to a high computational cost, as opposed to utilizing multiple scalar GPs for individual Δ.

IV. Sparse On-Line Gaussian Processes

GP utilizes Bayesian paradigm for the adaptive control by considering the identification as a statistical problem [10]. GP is non-parametric because the "parameters" to be identified are functions f_x of an input variable $x \in R^d$. Function f is characterised by its statistics, namely, by the mean $\langle f \rangle$ and the covariance, which is also called the kernel $K_0(x, x') = Cov(\varsigma, \varsigma')$ [17]. The a priori assumption is that f is a Gaussian process. Indeed, according to the Central Limit Theorem any sufficiently large set of random samples f_i is considered to have normal distribution. Within the Bayesian framework, given a set of input-output observations (x_n, ς_n) $(n = 1, \dots, N)$ the posterior distribution of the process f_x is computed via prior and the likelihood.

A representation of posterior means $\langle f_x \rangle_t = \langle f(x_t) \rangle_t$ and the posterior covariance $K_t(x, x')$, where t denotes the number of data points, with a finite linear combinations of kernels $K_0(x, x_i)$ evaluated at the training inputs x_i is proposed in [10]. Using sequential projections of the posterior process on the manifold of Gaussian processes, approximate recursions for the effective parameters of these representations can be obtained. To avoid enormous growth of the size of representations an elegant algorithm for extraction of a smaller subset of input data is proposed [10]. Such a subset allows an on-line sparse representation of the posterior process, which is used to predict the GP model.

The posterior expectations within the Bayesian approach are conventionally expressed by high-dimensional integrals. Obviously, this is not applicable for on-line identification. However, it was shown in [10] that the posterior mean and the posterior covariance of the process arbitrary inputs can be expressed as a combination of a finite set of parameters which depend on the training data only. To make Bayesian interference trackable on-line, the posterior is projected to the closest Gaussian process by a single sequential sweep through the examples.

The posterior GP approximation with its posterior means and the posterior covariance can be estimated using the initial kernel $K_0(x, x')$ and the likelihoods

$$\begin{aligned} \langle f_x \rangle_t &= \boldsymbol{\alpha}_t^T \boldsymbol{k}_x \\ K_t(x, x') &= K_0(x, x') + \boldsymbol{k}_x^T \boldsymbol{C}_t \boldsymbol{k}_{x'} \end{aligned} \tag{4}$$

where $\boldsymbol{k}_x = [K_0(x_1, x), \dots, K_0(x_t, x)]^T$ is the kernel functions, $\boldsymbol{\alpha}_t = [\alpha_t(1), \dots, \alpha_t(t)]^T$ is the coefficient vector, $\boldsymbol{C}_t = \{C_t(ij)\}_{i,j=1,t}$. is the coefficient matrix. It should be noted that coefficients $\alpha_t(i)$ and $C_t(ij)$ do not depend on x and x' [10]. For the regression problems, Radial Basis Functions (RBF) are quite popular choice for kernel functions (see, for example, [10], [14], [18], [19])

$$K(x, x') = exp(\tfrac{\|x - x'\|^2}{2\sigma_x^2}). \tag{5}$$

A. *Online Learning*

The recursive update of the GP parameters in Eq. (4) can be performed via the following equations:

$$\begin{aligned} \boldsymbol{\alpha}_{t+1} &= T_{t+1}(\boldsymbol{\alpha}_t) + q^{(t+1)} \boldsymbol{s}_{t+1}, \\ \boldsymbol{C}_t &= U_{t+1}(\boldsymbol{C}_t) + r^{(t+1)} \boldsymbol{s}_{t+1} \boldsymbol{s}_{t+1}^T, \\ \boldsymbol{s}_{t+1} &= T_{t+1}(\boldsymbol{C}_t \boldsymbol{k}_{t+1}) + \boldsymbol{e}_{t+1}, \end{aligned} \tag{6}$$

where $\boldsymbol{k}_{t+1} = \boldsymbol{k}_{x_{t+1}}$ and $\boldsymbol{e}_{t+1}$ the $t+1$-th unit vector and $\boldsymbol{s}_{t+1}$ is introduced for clarity. Operators T_{t+1} and U_{t+1} extend a t-dimensional vector and matrix to a $t+1$ – dimensional one by appending zeros at the end of the vector and to the last row and column of the matrix respectively.

For the RBF kernel functions the $q^{(t+1)}$ and $r^{(t+1)}$ are as follows

$$\begin{aligned} q^{(t+1)} &= (\varsigma - \boldsymbol{\alpha}_t^T \boldsymbol{k}_x)/\sigma_x^2, \\ r^{(t+1)} &= -1/\sigma_x^2, \end{aligned} \tag{7}$$

where $\sigma_x^2 = \sigma_0^2 + \boldsymbol{k}_x^T \boldsymbol{C}_t \boldsymbol{k}_x + k_x^*$, $k_x^* = K_0(x, x)$. One can conclude that the dimension of the vector $\boldsymbol{\alpha}$ and the size of matrix $\boldsymbol{C}$ increases with each data point added since $\boldsymbol{e}_{t+1}$ is the $t+1$-th unit vector.

The updates in the form of Eqs. (6) has a drawback since the number of parameters increases quadratically with the number of training examples. An effective way of controlling the number of parameters was proposed in Ref. [10], namely, sparseness within the GP framework was introduced. According to this approach, the update of the GP parameters is implemented without increase in the number of parameters $\boldsymbol{\alpha}$ and $\boldsymbol{C}$ when, according a certain criterion, the error due to the approximation is not too large.

If the new input x_{t+1} is such that

$$\boldsymbol{K_0}(x, x_{t+1}) = \sum_{i=1}^{t} \hat{e}_{t+1}(i) K_0(x, x_i) \tag{8}$$

is true for all x, then the update can be achieved exactly. In this case, the updated process in the form Eq. (4) is represented by only the first *t* inputs, but with "renormalised" parameters $\widehat{\boldsymbol{\alpha}}$ and $\widehat{\boldsymbol{C}}$ and the update (6) is implemented without extending the size of the parameters $\boldsymbol{\alpha}$ and $\boldsymbol{C}$ and $\boldsymbol{s}_{t+1}$ as follows:

$$\begin{aligned} \widehat{\boldsymbol{\alpha}}_{t+1} &= \boldsymbol{\alpha}_t + q^{(t+1)} \hat{\boldsymbol{s}}_{t+1}, \\ \widehat{\boldsymbol{C}}_t &= \boldsymbol{C}_t + r^{(t+1)} \hat{\boldsymbol{s}}_{t+1} \hat{\boldsymbol{s}}_{t+1}^T, \\ \hat{\boldsymbol{s}}_{t+1} &= \boldsymbol{C}_t \boldsymbol{k}_{t+1} + \hat{\boldsymbol{e}}_{t+1}, \end{aligned} \tag{9}$$

where $\widehat{\boldsymbol{\alpha}}_{t+1}, \widehat{\boldsymbol{C}}_t$ and $\hat{\boldsymbol{s}}_{t+1}$ are *t*-th unit vectors.

Obviously, for most kernels and inputs x_{t+1} relationship (8) does not hold for all input x. However, the updates in the form of (9) might be used for approximations if $\hat{\boldsymbol{e}}_{t+1}$ is determined by minimising the error measure

$$\|K_0(\cdot, x_{t+1}) - \textstyle\sum_{i=1}^{t} \hat{e}_{t+1}(i) K_0(\cdot, x_i)\|^2, \tag{10}$$

where $\|\cdot\|$ is a norm in a space of functions of inputs x. If the norm is defined via the inner product of the Reproducing Kernel Hilbert Space (RKHS) generated by the kernel K_0, then minimising (10), one can obtain the following expression

$$\hat{\boldsymbol{e}}_{t+1} = \boldsymbol{K}_t^{-1} \boldsymbol{k}_{t+1}, \tag{11}$$

where $\boldsymbol{K}_t = \{K_0(x_i, x_j)\}_{i,j=1,t}$ is the Gram matrix. In this case, the equation

$$\widehat{K}_0(x, x_{t+1}) = \textstyle\sum_{i=1}^{t} \hat{e}_{t+1}(i) K_0(\cdot, x_i), \tag{12}$$

gives the orthogonal projection of the function $K_0(x, x_{t+1})$ on the linear span of the functions $K_0(x, x_i)$.

The update rule (9) is performed when a measure of the approximation error

$$\gamma_{t+1} = k_{t+1}^* - \boldsymbol{k}_{t+1}^T \boldsymbol{K}_t^{-1} \boldsymbol{k}_{t+1} \tag{13}$$

does not exceed some tolerance level $\epsilon_{tol} > 0$. Here, $k_{t+1}^* = K_0(x_{t+1}, x_{t+1})$. The Eq. (13) has a geometrical interpretation, namely, it is a square norm of the "residual vector" from the projection in the RKHS. Alternatively, it measures the "novelty" of the current input. If γ_{t+1} is higher than a threshold value then the current input holds additional information as compared to the existing set of inputs, which is called "basis vector set" or *BV* set, and thus it should be added to this set. Proceeding sequentially, some of the inputs are left out and others are included in the *BV* set. However, because

of the projection (12) the inputs left out from the *BV* set will still contribute to the final GP configuration – the one used for prediction and to measure the posterior uncertainties. But the latter inputs will not be stored and do not lead to an increase of the size of the parameter set [10].

B. Deleting a Basis Vector

Recursive update of the GP parameters (6) is implemented while the *BV* set does not exceed the budget, namely, the maximum number of elements in the *BV*. Thus, a pruning procedure should be introduced. When a new example is estimated as novel, this procedure should get rid of one of the basis vectors and replace it by the new input vector. Two different strategies can be applied for selection of the vector from the *BV* set. The first strategy is supposed to add a novel input vector instead of the oldest basis vector [18]. The second strategy [10] proposes to replace the basis vector with the smallest error. The former might be preferred for a fast-varying process. However, here we will follow the later approach since it provides enhanced richness of the *BV* set.

The removal procedure assumes that the respective *BV* was added and the previous update step (9) was implemented. In this case $\boldsymbol{\alpha}_{t+1}$ has $t+1$ elements, and C_{t+1} and Q_{t+1} are the $(t+1)\times(t+1)$ matrices. If we assume that the last added element should be deleted the decomposition of the $\boldsymbol{\alpha}_{t+1}$, C_{t+1} and Q_{t+1} could be represented as follows:

$$\boldsymbol{\alpha}_{t+1}=\begin{bmatrix}\boldsymbol{\alpha}_t^l\\ \alpha^r\end{bmatrix}, C_{t+1}=\begin{bmatrix}C_t^l & \boldsymbol{c}^r\\ \boldsymbol{c}^{rT} & c^r\end{bmatrix}, Q_{t+1}=\begin{bmatrix}Q_t^l & \boldsymbol{q}^r\\ \boldsymbol{q}^{rT} & q^r\end{bmatrix}, \tag{14}$$

where C_t^l and Q_t^l are $t\times t$ sub-matrices extracted from the $(t+1)\times(t+1)$ matrices C_{t+1} and Q_{t+1}. For simplicity, this representation is shown for the case when the last element should be removed, however, similar partitioning could be done for a general case. Updating equations for the element deleting case are the following:

$$\begin{aligned}\hat{\boldsymbol{\alpha}} &= \boldsymbol{\alpha}_t^l-\alpha^r\frac{\boldsymbol{q}^r}{q^r},\\ \hat{C} &= C_t^l+c^r\frac{\boldsymbol{q}^r\boldsymbol{q}^{rT}}{q^{r2}}-\frac{1}{q^r}\left[\boldsymbol{q}^r\boldsymbol{c}^{rT}+\boldsymbol{c}^r\boldsymbol{q}^{rT}\right],\\ \hat{Q} &= Q_t^l-\frac{\boldsymbol{q}^r\boldsymbol{q}^{rT}}{q^r},\end{aligned} \tag{15}$$

where $\hat{\boldsymbol{\alpha}}$, $\hat{C}$ and $\hat{Q}$ are the parameters after the deletion of the last basis vector and $\boldsymbol{\alpha}_t^l$, C_t^l, Q_t^l, α^r, $\boldsymbol{c}^r$, $\boldsymbol{q}^r$, c^r and q^r are taken from GP parameters before deletion.

To decide the element of the *BV* set to be deleted a score measure for each element *i* is calculated

$$\varepsilon_i=\frac{|\alpha_{t+1}(i)|}{Q_{t+1}(i,i)}. \tag{16}$$

The basis vector with minimal score (16) is deleted. This method provides deleting of a basis vector from the *BV* set with minimal loss of information. Finally, the budgeted sparse GP algorithm is summarized by Algorithm 1.

Algorithm 1 Budgeted sparse GP algorithm

0: Initialize the *BV* set with an empty set, maximum number of the set elements with *d*, a tolerance with ϵ_{tol}, $\boldsymbol{\alpha}$, $\mathbf{C}$, $\mathbf{Q}$ with empty values.

For each new measurement $(x_{t+1}, \varsigma_{t+1})$ **iterate**

1. Compute $\boldsymbol{q}^{t+1}$, $\boldsymbol{r}^{t+1}$, k_{t+1}^*, $\boldsymbol{k}_{t+1}$, $\hat{\boldsymbol{e}}_{t+1}$ and γ_{t+1}.
2. **If** $\gamma_{t+1} < \epsilon_{tol}$ **then**

Perform a reduced update using (9).

3. **else**

Perform an update using (6). Add the current input to the *BV* set, and compute the inversed Gram matrix.

4. **If** $|BV| > d$ **then**

Compute scores for the *BV* elements via (16) find the vector corresponding to the lowest score, and delete it using (15).

V. Longitudinal Flight Control

A. Flight Dynamics Model

In this section, performance of the proposed approach is considered with a tracking control of longitudinal missile model from [20]. The second-order nonlinear model represents longitudinal dynamics of a vehicle traveling at an altitude of approximately 6000 meters, with aerodynamic coefficients represented as third order polynomials in angle of attack α and Mach number M. The nonlinear equations of motion in the pitch plane are given by

$$\begin{aligned}\dot{x}_1 &= x_2+f_1(x_1)+g_1u\\ \dot{x}_2 &= f_2(x_1)+g_2u\,,\end{aligned} \tag{17}$$

where $x_1=\alpha$, $x_2=q$, $g_1=C_1b_z$ and $g_2=C_2b_m$ and

$$\begin{aligned}f_1(x_1) &= C_1[\varphi_{z1}(x_1)+\varphi_{z2}(x_1)M],\ C_1=\frac{\bar{q}S}{mV},\\ f_2(x_1) &= C_2[\varphi_{m1}(x_1)+\varphi_{m2}(x_1)M],\ C_2=\frac{\bar{q}Sd}{I_{yy}}.\end{aligned} \tag{18}$$

Here

$$\begin{aligned}\varphi_{z1}(x_1) &= -288.7x_1^3+50.32x_1|x_1|-23.89x_1\\ \varphi_{z2}(x_1) &= -13.53x_1|x_1|+4.185x_1,\\ \varphi_{m1}(x_1) &= 303.1x_1^3-246.3x_1|x_1|-37.56x_1,\\ \varphi_{m2}(x_1) &= 71.51x_1|x_1|+10.01x_1.\end{aligned} \tag{19}$$

The model is valid for $-10°\le\alpha\le10°$ and $1.8\le\alpha\le2.6$.

Details of the aerodynamic coefficients can be found in [20].

B. *INDI Longitudinal Controller*

INDI longitudinal controller that tracks a smooth command reference y_r with the pitch rate x_2 was developed in [21] and used here as a baseline controller. Details of the controller design can be found in the original paper. During the design, it was assumed that the aerodynamic force and moment functions are accurately known and the Mach number M is treated as a parameter available for measurement. Moreover, for this second-order system in non-lower triangular form due to $g_1 u$ and $f_2(x_1)$, pitch rate control using INDI is possible due to the timescale separation principle [4].

The rate tracking error is introduced as follows

$$z_2 = x_2 - x_{2_{ref}}. \tag{20}$$

The z_2-dynamics satisfies the following error

$$\dot{z}_2 = \dot{x}_2 - \dot{x}_{2_{ref}}, \tag{21}$$

for which the following exponentially stable desired error dynamics is introduced

$$\dot{z}_2 + k_{P_2} z_2 = 0, \; k_{P_2}\text{=50 rad/s.} \tag{22}$$

INDI was derived from the following approximate dynamics of the pitch rate

$$\dot{x}_2 \cong \dot{x}_{2_0} + \bar{g} u, \tag{23}$$

where $\bar{g}_2$ is the estimate of g_2. The control law in this case is obtained as

$$u \cong u_0 + \bar{g}^{-1}(\nu - \dot{x}_{2_0}), \tag{24}$$

where

$$\nu = -k_{P_2} z_2 + \dot{x}_{2_{ref}}. \tag{25}$$

Should be noted here that the accurate knowledge of f_2 were replaced with measurements (or estimates) as $f_2 \cong \dot{x}_{2_0}$, which reduces dependence of the control law on the model.

C. *GP Adaptation*

As was mentioned before, incremental-type controller is dependent on the quality of the measurement (estimation) of state derivatives, furthermore, the incremental-type controller can have degraded performance and even lost the system stability in case of presence on unknown nonlinearities, e.g. transport delay or unknown actuator dynamics [15]. To improve robustness of the controller to uncertainties, we introduced GP-based adaptation and modified the controller in the following form

$$u \cong u_0 + \bar{g}^{-1}(\nu - \dot{x}_{2_0} + \hat{\Delta}_{GP}), \tag{26}$$

where $\hat{\Delta}_{GP}$ is a GP estimation of uncertainties. $\varsigma = \Delta\dot{x}_{2_0} - \bar{g}\Delta u$. The maximum number d of the BV set was 6, $\epsilon_{tol} = 1e-5$, $\sigma_0^2 = 1.5$. To obtain the appropriate overlapping between neighbouring kernels the RBF width is specified as follows

$$\sigma = \frac{10}{2(\sqrt{d}-1)}.$$

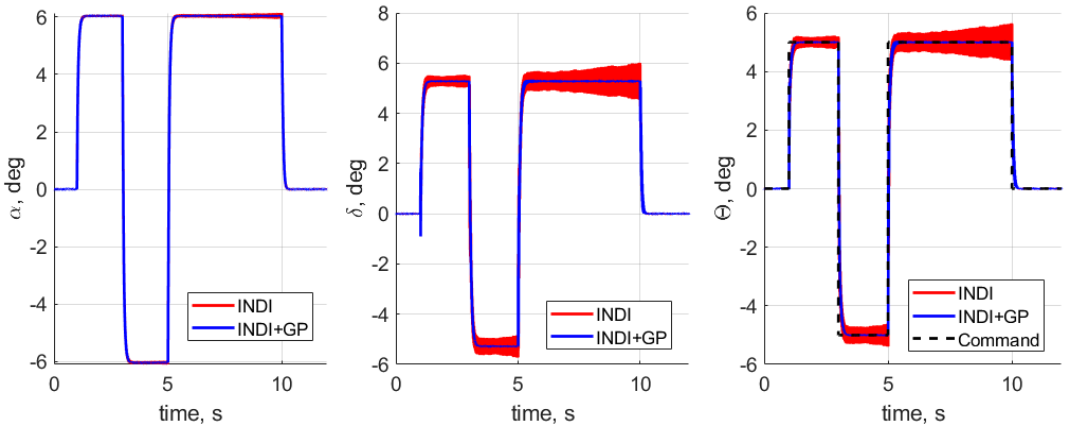

Fig.1. Two failures and loss of effectiveness

VI. Simulation Results

In the current section an ability of the proposed controller is evaluated in a case of presence of unknown nonlinear dynamics in the actuator $F(s) = (s + 0.25)^{-1}$ and a sensor noise, namely, zero-mean Gaussian white-noise with standard deviation $1e^{-6}$. The proposed INDI with GP adaptation is compared with the baseline GP controller. The results are presented in Fig.1.

From the figure one could see that the INDI has instability, however the proposed adaptation cancels out the existing uncertainty and follows reference signal. It should be noted, as opposite to [11] the GP performs identification of unknown function online. Furthermore, these approach can tackle a wide class of uncertainties, not only ones presented in the actuation as in [15].

VII. CONCLUSIONS

Sensor-based incremental control is recently developed technique with a reduced dependency on the on-board model. This approach uses estimates of the state derivatives and the current actuator states to linearize the system dynamics with respect to the previous time moment. Some previous researches suggest that incremental controllers are robust to actuator failures, when the system remains input affine. However, this type of controllers is sensitive to the quality of the measured or estimated state vector derivative. However, a failure of measurement system might cause a noise growth and thus drastically effect the stability of the system. Intensive filtering of a noisy signal also can produce undesired signal delays and thus cause the system instability.

In this work, we utilised GP framework to assist INDI in tackling uncertainties. GP provide a flexible nonparametric data-driven modelling framework that incorporates an automatic trade-off between data fitting and regularization in noisy conditions. The other advantage is that GP enables long-term learning. To make Bayesian interference trackable on-line, the posterior is projected to the closest Gaussian process by a single sequential sweep through the examples. Within the approach, the number of basis vectors is limited with predefined "budget" to make the algorithm computationally efficient. The input vectors providing the maximum information richness to the basis set, namely, having maximum scores, are selected as the basis vectors, while basis vectors with lowest scores are deleted from the set. Budgeted sparse approximation of GP allowed to train the model online with a low-computational costs, which is extremely important for real-world applications. Our simulation results revealed improved tracking performance and stability of the INDI with GP-based adaptation. Adaptive loop performs not only a capturing of uncertainty but also a noise signal filtering. We showed that a combination of multiple failures of sensors and actuators, the system dynamics might loss its input affine property. As a result, the stability of the closed-loop system with pure INDI cannot be guaranteed anymore. The proposed GP-adaptive augmentation compensates the unmodelled dynamics.

References

[1] J.-J. E. Slotine and W. Li, *Applied Nonlinear Control*, vol. 62, no. 7. 1991.

[2] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*, 1st. Ed. John Wiley & Sons, Inc., USA., 1995.

[3] L. G. Sun, C. C. de Visser, Q. P. Chu, and W. Falkena, "Hybrid Sensor-Based Backstepping Control Approach with Its Application to Fault-Tolerant Flight Control," *J. Guid. Control. Dyn.*, vol. 37, no. 1, pp. 59–71, 2013, doi: 10.2514/1.61890.

[4] S. Sieberling, Q. P. Chu, and J. A. Mulder, "Robust Flight Control Using Incremental Nonlinear Dynamic Inversion and Angular Acceleration Prediction," *J. Guid. Control. Dyn.*, vol. 33, no. 6, pp. 1732–1742, Nov. 2010, doi: 10.2514/1.49978.

[5] G. P. Falconi, V. A. Marvakov, and F. Holzapfel, "Fault tolerant control for a hexarotor system using Incremental Backstepping," *2016 IEEE Conf. Control Appl. CCA 2016*, no. 1, pp. 237–242, 2016, doi: 10.1109/CCA.2016.7587842.

[6] D. I. Ignatyev, H. S. Shin, and A. Tsourdos, "Two-layer adaptive augmentation for incremental backstepping flight control of transport aircraft in uncertain conditions," *Aerosp. Sci. Technol.*, vol. 105, p. 106051, 2020, doi: 10.1016/j.ast.2020.106051.

[7] A. Calise, Y. Shin, and M. Johnson, "A Comparison Study of Classical and Neural Network Based Adaptive Control of Wing Rock," no. August, pp. 1–17, 2004, doi: 10.2514/6.2004-5320.

[8] D. I. Ignatyev, H. S. Shin, and A. Tsourdos, "Bayesian calibration for multiple source regression model," *Neurocomputing*, vol. 318, pp. 55–64, 2018, doi: 10.1016/j.neucom.2018.08.027.

[9] K. S. Narendra, "Neural networks for control: Theory and practice," *Proc. IEEE*, vol. 84, no. 10, pp. 1385–1406, 1996, doi: 10.1109/5.537106.

[10] L. Csató and M. Opper, "Sparse On-Line Gaussian Processes," *Neural Comput.*, vol. 14, no. 3, pp. 641–668, Mar. 2002, doi: 10.1162/089976602317250933.

[11] A. Capone and S. Hirche, "Backstepping for Partially Unknown Nonlinear Systems Using Gaussian Processes," *IEEE Control Syst. Lett.*, vol. 3, no. 2, pp. 416–421, 2019, doi: 10.1109/LCSYS.2018.2890467.

[12] P. A. Ioannou and P. V Kokotovic, "Instability analysis and improvement of robustness of adaptive control," *Automatica*, vol. 20, no. 5, pp. 583–594, 1984, doi: https://doi.org/10.1016/0005-1098(84)90009-8.

[13] B. W. Karl Johan Åström, *Adaptive Control*, 2nd ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1994.

[14] G. Chowdhary, H. A. Kingravi, J. P. How, and P. A. Vela, "Bayesian Nonparametric Adaptive Control Using Gaussian Processes," *IEEE Trans. NEURAL NETWORKS Learn. Syst.*, vol. 26, no. 3, pp. 537–550, 2015, doi: 10.1109/TNNLS.2014.2319052.

[15] D. Ignatyev, H.-S. Shin, and A. Tsourdos, "Gaussian Process Adaptive Incremental Backstepping Flight Control," pp. 1–17, 2022, doi: 10.2514/6.2022-2032.

[16] C. E. Rasmussen and C. K. I. Williams, "Gaussian Processes for Machine Learning." The MIT Press, 23-Nov-2005, doi: 10.7551/mitpress/3206.001.0001.

[17] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Second. Springer, 1999.

[18] R. C. Grande, G. Chowdhary, and J. P. How, "Experimental Validation of Bayesian Nonparametric Adaptive Control Using Gaussian Processes," *J. Aerosp. Inf. Syst.*, vol. 11, no. 9, pp. 565–578, 2014, doi: 10.2514/1.i010190.

[19] D. I. Ignatyev, M. E. Sidoryuk, K. A. Kolinko, and A. N. Khrabrov, "Dynamic rig for validation of control algorithms at high angles of attack," *J. Aircr.*, vol. 54, no. 5, pp. 1760–1771, 2017, doi: 10.2514/1.C034167.

[20] S. H. Kim, Y. S. Kim, and C. Song, "A robust adaptive nonlinear control approach to missile autopilot design," *Control Eng. Pract.*, vol. 12, no. 2, pp. 149–154, 2004, doi: 10.1016/S0967-0661(03)00016-9.

[21] A. B. Paul, W. van Ekeren, and Q. P. Chu, "PI(D) tuning for Flight Control Systems via Incremental Nonlinear Dynamic Inversion," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 8175–8180, 2017, doi: 10.1016/j.ifacol.2017.08.1265.

https://dspace.org/

Incremental nonlinear dynamic inversion with sparse online Gaussian processes adaptation for partially unknown systems

Ignatyev, Dmitry; Tsourdos, Antonios

2022-08-01
IEEE
https://doi.org/10.1109/MED54222.2022.9837175
Ignatyev D, Tsourdos A. (2022) Incremental nonlinear dynamic inversion with sparse online Gaussian processes adaptation for partially unknown systems. In: 2022 30th Mediterranean Conference on Control and Automation (MED), 28 June - 1 July 2022, Vouliagmeni, Greece, pp. 233-238

Downloaded from a DSpace repository. https://dspace.org/