Distributed Learning of Gaussian Process Augmented State-Space Models

Sebastian Karlsson*, Isaac Skog†*, and Gustaf Hendeby*
* Dept. of Electrical Engineering, Linköping University, Linköping, Sweden
e-mail: firstname.lastname@liu.se
† School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden
e-mail: skog@kth.se

I. Introduction

This work deals with methods of joint filtering and system identification, by means of Gaussian process-augmented state space models (GPASSM). These models are a class of grey-box models which combine a prior physics-based model of a dynamical system with a learned model of unknown dynamics or inputs. The unknown part of the system is modeled by a Gaussian process which is trained on estimation data, becoming more informative with repeated exposure of system dynamics. This allows the user to pre-define an approximate model of a system which iteratively becomes more accurate, enabling enhanced filtering performance compared to purely model-based methods, while providing a guaranteed performance level even in regions with sparse or non-existent data. Using Gaussian processes (GP:s) for the learned part of the dynamics, instead of black-box models such as deep neural networks which in recent years have been massively successful in many domains, is motivated by the explicit treatment of model uncertainty in GP:s. Such a measure of uncertainty is highly relevant in applications of this method for autonomous or safety-critical systems.

GPASSM models have been used successfully in applications such as learning driver behaviours [1] and anomaly detection in marine vessels [2]. The models have been shown to significantly reduce estimation error compared to non-augmented models, and methods for reducing the computational complexity using basis function expansions have been explored and shown to be effective. Furthermore, adaptive selection of basis functions is possible which can further improve the computational complexity.

While showing promising results, these models have not yet seen much use in practice, partly due to the still non-favourable complexity scaling of Gaussian processes. The aim of this work is to take the next step to increase the understanding of different methods for combining Gaussian processes with state space models, and ultimately to facilitate proper scalability of the methods rendering them more practically useful.

The first step to achieve this goal is investigating different ways of training the GP and using its predictions within the framework of state filtering, to identify strengths and weaknesses. In future developments, distribution of the GPASSM model to several computational nodes is to be investigated which may further improve tractability.

This work was partially supported by the Wallenberg Artificial Intelligence, Autonomous Systems and Software Program (WASP), funded by Knut and Alice Wallenberg Foundation.

II. Problem description

The GPASSM model structure used in the sequel is

$$x_{k+1} = \underbrace{Fx_k}_{\text{Physical model}} + \underbrace{u_{GP}(x_k)}_{\text{Data driven model}} \tag{1a}$$

$$y_k = Hx_k + e_k \tag{1b}$$

$$u_{GP}(x_k) \sim \mathcal{GP}\big(\mu(x_k), K(x, x')\big) \tag{1c}$$

where F describes the pre-determined system dynamics as a linear state space model, and $u_{GP}(x_k)$ is a nonlinear state-dependent system input which is modeled as a Gaussian process with mean $\mu(x_k)$ and covariance $K(x, x')$.

Three different methods of combining the state-space and the GP models are envisioned:

A. *Offline learning*

In the first method, filtering is performed using only the state-space model, ignoring $u_{GP}(x)$. The estimated states are stored and used afterwards to train $u_{GP}(x)$ by treating part of the state as a GP model input, and another part as output. For instance, in the example presented in section III, position states are used as input and acceleration as output. This batch-training method is incapable of continuous improvement, but is simple to analyze and compare to other methods. The state-space and the GP model parts are clearly separated with this method, which makes a combined treatment of their individual uncertainties less obvious.

B. *Online learning with input prediction*

In the second method, the state-space and the GP model are separate, as in the first method. However, the GP is updated and used online during the tracking by alternating between updating x_k and $u_{GP}(x)$. Due to this, the estimation will become more informed of the unmodeled parts of the system during the tracking process, resulting in higher accuracy. However, erroneous state estimates due to noise or model error will immediately be incorporated in the model, which in turn may aggravate the model error affecting future estimates negatively. Appropriately dealing with this error feedback effect is of interest to investigate in this work.

C. State augmentation by GP parameters

In the third method, the GP model is parameterized and its parameters are concatenated to the state vector by letting $\tilde{x} = [x, \theta]^\top$ and $u_{GP}(x) = f(\theta, x)$, where f is a function of the state x parameterized by θ. Here, the state-space model and the GP model are fully joined into a single framework, and the GP parameters will be updated by the same filtering algorithm that is used to estimate the state. This method has been investigated in previous works [2] [3] and has several advantages. Due to the joined nature of the models, uncertainty in the estimated state and in the learned GP parameters will be connected by the joint covariance matrix. Consequently, training data based on uncertain state estimates will propagate to uncertainty in the model output, which may reduce the impact of the aforementioned error feedback. A disadvantage of this method is that the GP model must be parameterized, for instance using methods based on inducing points or basis function expansion. This induces a need to define the model order a priori, reducing the flexibility of the GP model.

III. Illustrative Example

The investigation is carried out using a simulated toy problem where an electrically charged particle with mass is subjected to a uniform gravitational field and a non-uniform electrical field, seen in Fig. 1. The location of the particle is tracked using a Kalman filter, where the time update step uses a GPASSM. The prior state-space model is a standard motion model, such as constant-velocity or constant-acceleration, while the Gaussian process is set up to learn the acceleration field acting on the particle. By using the joint model, significantly improved tracking should be possible while the tracking is still adequate even before a single data point has been collected.

This experiment setup enables a comparison between different methods of combining state-space and GP models. The multi-modal aspect of the acceleration field can be used to compare tracking accuracy in different regions — in areas where the gravitational force dominates, a low-complexity acceleration model should suffice. On the other hand, while in areas where the acceleration is rapidly changing due to the electrical forces dominating, a more complex model would be required. Due to the shape of the acceleration field, the region directly below the electrical charge will never be explored, raising the question of how to treat lack of information. The experiment setup additionally has interesting real-world counterparts, such as navigation under unknown wind disturbances or analysis of system behaviours by modeling the system input.

To illustrate the proposed method, 25 particles are simulated for 100 time steps using methods A and B respectively. During each particle's duration, 10 equally-spaced estimates are used to train the GP model. The resulting GP models are visualized in Fig. 2 by predicting the acceleration $u_{GP}(x)$ on a grid of positions. In regions where more particles have passed, more data is available and the model has less uncertainty.

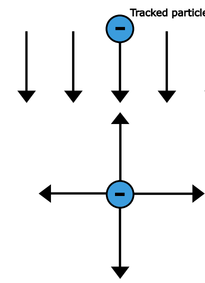
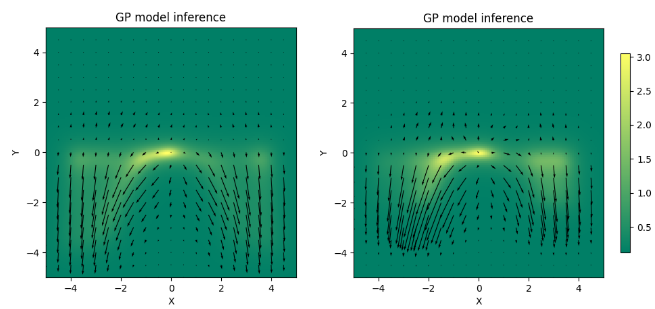


Fig. 1: Illustration of experiment setup, where a tracked particle is affected by gravity and an electrical field.



(a) Method A (b) Method B

Fig. 2: Quiver plot of predicted acceleration $u_{GP}(x)$ on a grid of x values, with associated GP precision.

IV. Summary and Outlook

The GPASSM class of models have been shown to be effective in joint filtering and system identification. The described experiment setup has validated these results in preliminary tests, and will be used in future work for evaluating properties of the GPASSM as well as extending the models, primarily with the goal of enabling real-time use by distributing the computations. Distributed learning and estimation is possible to investigate by simulating a multitude of particles simultaneously or in sequence. Modifying the GPASSM from a centralized model to a near-edge model is a possible method for enabling GPASSM models to be used in large-scale applications. Some methods in consideration are consensus and diffusion Kalman filters, as well as Bayesian committee machines. Strategies for active learning and properly dealing with uncertain inputs are other avenues of investigation.

References

[1] A. Kullberg, I. Skog, and G. Hendeby. Learning driver behaviors using a gaussian process augmented state-space model. In *IEEE International Conference on Information Fusion (FUSION)*.

[2] A. Kullberg, I. Skog, and G. Hendeby. Learning motion patterns in ais data and detecting anomalous vessel behavior. In *IEEE International Conference on Information Fusion (FUSION)*, 2021.

[3] A. Kullberg, I. Skog, and G. Hendeby. Online joint state inference and learning of partially unknown state-space models. *IEEE Transactions on Signal Processing*, 69:4149–4161, 2021.