An Introduction to Gaussian Processes for the Kalman Filter Expert

Steven Reece and **Stephen Roberts**
Robotics Research Group
Dept. Engineering Science
Oxford University, UK.
{reece, sjrob}@robots.ox.ac.uk

Abstract – *We examine the close relationship between Gaussian processes and the Kalman filter and show how Gaussian processes can be interpreted using familiar Kalman filter mathematical concepts. We use this insight to develop a novel hybrid filter, which we call the KFGP, for spatial-temporal modelling. The KFGP uses Gaussian process kernels to model the spatial field while exploiting efficient Kalman filter state-based approaches to model the temporal component. We also develop a Gaussian process kernel for the familiar Kalman filter near constant acceleration model.*

Keywords: Gaussian processes, Kalman filter, Kriged Kalman filter, Bayesian methods, Near constant acceleration model.

1   Introduction

Gaussian processes (GP) are experiencing a resurgence of interest. Current applications are in diverse fields such as geophysics, medical imaging, multi-sensor fusion [12] and sensor placement [4]. GPs have been used to solve classification and regression problems, although we shall focus on regression problems in this paper. GPs are often the preferred approach to function estimation[1] as they offer several useful properties [3]: GPs do not require a discretised representation of an environment and they are able to predict function values at arbitrary locations; GPs are non-parametric regression models and thereby are able to approximate an extremely wide range of problem domains. In contrast to other regression models, GPs provide a measure of estimate uncertainty and this takes into account the noise of the data points.

Kernels (also called covariance functions) encode information about the spatial correlation between measurements. These kernels are widely applicable models of the underlying process and can be tailored to specific applications by the refinement of their hyperparameters. Efficient techniques exist to learn these hyperparameters from training data [13].

The relationship between Gaussian processes and other kernel based models such as splines, Kriging, ARMA, and support vector machines are well known [2, 13]. Despite the fact that Kalman filters (KF) can be seen as a special case of Gaussian processes (GPs) [9] they differ in the way the models need to be thought about (i.e. physical state-based versus covariance function) that describe the underlying process. The KF is not normally thought of as a non-parametric model and it is commonly, although mistakenly, believed that the Kalman filter is constrained to operate over discrete time intervals. The KF has been widely used for time-series forecasting and, when the problem can be represented by linear, Markovian process models and linear observation models, the KF can be the most efficient approach.

In this paper we argue that the GP can be seen as a special case of the KF. Thus, our intuitions about the KF, often obtained after many years of experience with KF implementations, lend themselves to GPs. Further, since the iterative state-based operations associated with Kalman filters have not yet been introduced to GPs we argue that KFs can be more efficient than GPs in some cases. Consequently, we will develop efficient hybrid KF and GP methods for function estimation.

Computational efficiency isn't the only reason for developing the relationship between GPs and KFs. We may also determine how to exploit novel techniques used in Kalman filtering, such as Covariance Intersection [7], in Gaussian processes. Encouragingly, KF approaches have seen some introduction into GP methods recently. For example, [8] use the Unscented transform [6] to accommodate training data uncertainty in GPs.

This paper is structured as follows: Section 2 describes the familiar Kalman filter and then shows how it can be used for function estimation; Section 3 introduces Gaussian processes and then Section 4 establishes an interpretation of GPs from a Kalman filter perspective. Consequently, we show how non-parametric GP covariance functions can incorporated into Kalman filter process models in Section 5. We call the combined filter the *KFGP*. Section 6 applies the KFGP to adaptive field estimation and we also develop a GP

---

[1] A stochastic *process* is a generalisation of a finite-dimensional distribution to an infinite-dimensional function [13].

kernel for the familiar Kalman filter near constant acceleration model. Finally, we conclude in Section 7.

2 The Kalman Filter

The Kalman filter is used to calculate the marginal posterior mean and covariance of a multivariate Gaussian distribution over a time sequence t_k indexed $k \in \mathbb{N}$. The observation vector, y_k, is drawn from a noisy multivariate Gaussian process, f_k, at time t_k:

$$y_k = H_k f_k + \epsilon_k .$$

H_k is a linear *observation model* and ϵ_k is a zero-mean multivariate Gaussian random vector with covariance R. The KF determines the mean, $\bar{f}_{k|k}$, and covariance $P_{k|k}$ of the multivariate Gaussian over f_k conditioned on the observations y_j ($j \le k$). Let $\bar{f}_{k|k-1}$ and $P_{k|k-1}$ be the multivariate Gaussian mean and covariance of f_k at time t_k given observations y_j ($j < k$) then the KF fusion equations are:

$$\bar{f}_{k|k} = \bar{f}_{k|k-1} + \Gamma_k [y_k - H_k \bar{f}_{k|k-1}] ,$$
$$P_{k|k} = [I - \Gamma_k H_k] P_{k|k-1}$$

where:

$$\Gamma_k = P_{k|k-1} H_k^T [H_k P_{k|k-1} H_k^T + R]^{-1}$$

is the Kalman gain.

We assume that we are given, or can learn, the prior Gaussian mean and covariance, $\bar{f}_{0|0}$ and $P_{0|0}$, respectively. Of course, the Kalman filter also has the capability of performing efficient predictions when the temporal process is Markovian. However, we shall ignore this capability for now and focus on the KF fusion operation alone.
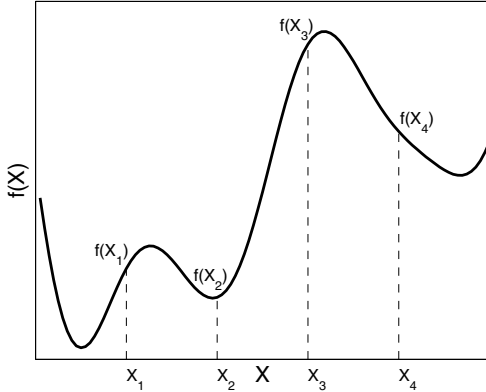

Figure 1: A function sampled at four inputs.

We now turn our attention to the inference of functions. Let $f(x)$ be a nonlinear function over a vector space x. We shall refer to x as the *input* vector and $f(x)$ as the scalar *output*. Although the function f may be initially unknown, we may occasionally glimpse different parts of the function

and subsequently infer the whole function. At iteration k we may obtain observations Y_k of this function at n input values $X = \{x_1, \ldots, x_n\}$ (see Figure 1). The observations may be noisy but we can assume:

$$Y_k = f(X) + \epsilon_k \tag{1}$$

where ϵ_k is drawn from a zero-mean multivariate Gaussian distribution with covariance $R = \sigma^2 I$. By choosing a diagonal covariance R we are asserting that the observations of the function at different input values are uncorrelated. Suppose we wish to infer (or predict) the function values at m inputs $X_* = \{x_1^*, \ldots, x_m^*\}$:

$$f(X_*) = (f(x_1^*), \ldots, f(x_m^*)) .$$

For clarity of exposition we will define $X' \triangleq X_* \cup X$ and the indicator matrices H^* and H_k such that $X_* = H^* X'$ and $X = H_k X'$. (1) then becomes:

$$Y_k = H_k f(X') + \epsilon_k .$$

Now, assume we have a prior Gaussian distribution over $f(X')$ with mean $\bar{f}_{|0}(X')$ and covariance $P_{|0}(X', X')$. [2] Then we could apply the KF fusion equations to the state vector $f(X')$ and obtain an estimate for the function at all inputs X' using all previous observations:

$$\bar{f}_{|k}(X') = \bar{f}_{|k-1}(X') + \Gamma_k [Y_k - H_k \bar{f}_{|k-1}(X')] ,$$
$$P_{|k}(X', X') = [I - \Gamma_k H_k] P_{|k-1}(X', X')$$

where:

$$\Gamma_k = P_{|k-1}(X', X') H_k^T [H_k P_{|k-1}(X', X') H_k^T + \sigma^2 I]^{-1} .$$

and $\bar{f}_{|k}(\cdot)$ and $P_{|k}(\cdot)$ are the posterior Gaussian mean and covariance of the function f. Pre and post multiplying by the indicator function H^* and noting that $\bar{f}_{|k-1}(X) = H_k \bar{f}_{|k-1}(X')$ and $P_{|k-1}(X, X) = H_k P_{|k-1}(X', X') H_k^T$, we recover the KF fusion equations for X_* only:

$$\bar{f}_{|k}(X_*) = \bar{f}_{|k-1}(X_*) + [P_{|k-1}(X, X) + \sigma^2 I]^{-1} \times$$
$$[Y_k - \bar{f}_{|k-1}(X)] , \tag{2}$$
$$P_{|k}(X_*, X_*) = P_{|k-1}(X_*, X_*) - P_{|k-1}(X_*, X) \times$$
$$[P_{|k-1}(X, X) + \sigma^2 I]^{-1} P_{|k-1}(X, X_*) . \tag{3}$$

where $P_{|k-1}(X_*, X)$ is the function covariance between the training points, X, and the prediction points, X_*.

Thus, if we are given a priori distributions over $f(X_*)$ and $f(X)$ and observations for f at inputs X then we can estimate the function at inputs X_* using the KF fusion equations. In general, the input vectors can be either spatial or temporal or they can be combinations of the spatial and temporal dimensions.

---

[2]The term $\bar{f}_{|j}$ is the mean of f given measurements up to time t_j when f is stationary (i.e. f is not changing over time).

It remains to determine the form of the mean and covariance for the prior distribution over the function f. We will show that the prior covariance can be created using covariance functions borrowed from Gaussian processes. In the following sections we will also show how these kernels can be incorporated into Kalman filter process models and consequently develop a novel approach to non-stationary field estimation. But first, we shall introduce Gaussian processes (and their covariance functions) and then establish a link between them and Kalman filters.

3 Gaussian Processes

A GP is often thought of as a "Gaussian distribution over functions" [13]. It can be thought of as the generalisation of a Gaussian distribution over a finite vector space to a function space of infinite dimension. Just as a Gaussian is fully specified by its mean and covariance matrix, a Gaussian process is fully described by its mean and *covariance function* K. Although our functions are infinitely dimensional, GPs are used to infer, or predict, function values at a finite set of *prediction points* from the observed data. The training data $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ is drawn from a noisy process:

$$y_i = f(x_i) + \epsilon_i \tag{4}$$

where ϵ_i is independent identically distributed Gaussian noise with variance σ^2. For convenience both inputs and outputs are aggregated into $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$ respectively. The GP estimates the value of the function f at arbitrary prediction points $X_* = \{x_{*1}, \ldots, x_{*m}\}$. The basic GP regression equations are given in [13]:

$$\bar{f}_* = m(X_*) + K(X_*, X)[K(X, X) + \sigma^2 I]^{-1} \times$$
$$(Y - m(X)), \tag{5}$$
$$\mathrm{Cov}(f_*) = K(X_*, X_*) - K(X_*, X) \times$$
$$[K(X, X) + \sigma^2 I]^{-1} K(X_*, X)^T \tag{6}$$

where I is the identity matrix, $m(\cdot)$ is the prior function mean[3], $\bar{f}_*$ is the posterior function mean and $\mathrm{Cov}(f_*)$ is the posterior covariance. The matrix $K(X, X)$ denotes the joint prior distribution covariance of the function at inputs X. This covariance matrix has elements:

$$K(x_i, x_j) = \mathrm{Cov}(f(x_i), f(x_j)) .$$

The matrix $K(X_*, X)$, obtained from the kernel K, is the covariance between the function at the prediction points, X_*, and the training inputs X. Many kernels have been designed to capture various properties of the modelled phenomenon including smoothness, dimensionality and stationarity. They are constructed to guarantee that any matrix obtained from them is a covariance matrix (i.e. positive semi-definite) irrespective of the choice of inputs X. Kernels can

be combined to form new kernels. Each kernel has a set of hyperparameters which control the magnitude of the covariance eigenvalues and also the degree of correlation between the outputs.[4]

4 Understanding GPs from a KF Perspective

The purpose of this section is to demonstrate that the Kalman filter expert is also, perhaps without knowing it, automatically a Gaussian process expert. The only deficiency may be a lack of familiarity with the kernels (i.e. models) that plug into GPs.

Comparing the Kalman filter fusion equations, (2) and (3), with the Gaussian process equations, (5) and (6), it is clear that the GP covariance matrix $K(X, X)$ corresponds to the KF prior covariance $P_{|0}(X, X)$ and the GP prior mean $m(X)$ corresponds to the KF estimate $\bar{f}_{|0}(X)$. Consequently, GP kernels can be used to derive prior distributions for the KF by asserting:

$$\bar{f}_{|0}(X) = m(X) ,$$
$$P_{|0}(X, X) = K(X, X) .$$

The matrix $K(X_*, X)[K(X, X) + \sigma^2 I]^{-1}$ in (5) and (6) is equivalent to the Kalman gain where the function values at X are observed and values at X_* are inferred. The GP requires that the observed inputs X are declared explicitly. However, the output function can be inferred at any arbitrary input value x_*. Similarly, for the KF.

If observations are made at new input values, X^N, then the GP pools all the data, new and old, before inferring the posterior distribution from scratch. The KF, on the other hand, fuses the new observations with the posterior distribution inferred from the previous observations. It may seem that, for the KF, all possible inputs X must be anticipated a priori in order to be able to calculate $\bar{f}_{|k-1}(X^N)$ using (2). This is not the case, however, as we can incorporate new inputs on the fly and interpolate the outputs $f(X^N \cup X)$ from the data. This is possible using a novel extension to the Kalman Smoother [5], which we present for the first time here and call the *Augmented Kalman Smoother* (AKS).

The AKS, just like the familiar Kalman Smoother, allows us to interpolate function values at new inputs without the need to refer to the observation history. The AKS can be used to interpolate over spatial or temporal dimensions. The AKS is derived in the Appendix and is applied to spatial dimensions here. Defining, $X^a \triangleq X^N \cup X$:

$$\bar{f}_{|k-1}(X^a) = \bar{f}_{|0}(X^a) + K(X^a, X)K(X, X)^{-1}$$
$$\times [\bar{f}_{|k-1}(X) - \bar{f}_{|0}(X)] ,$$
$$P_{|k-1}(X^a, X^a) = K(X^a, X^a) + K(X^a, X)K(X, X)^{-1}$$
$$\times (P_{|k-1}(X, X) - K(X, X))$$
$$\times K(X, X)^{-1} K(X^a, X)^T .$$

The AKS can be used to augment the state vector with function values inferred at a finite set of new input points.

---

[3]Often a zero prior function is chosen: $m(x) = 0$ for all x.

[4]The reader is invited to read [13] for an overview of GP kernels.

However, both the posterior mean and variance for any arbitrary individual prediction point can be determined using the AKS. Thus, the AKS can be seen as an infinite dimensional model over the input space.

5 Tracking Non-Stationary Functions

So far we have looked at stationary functions. We will now relax this constraint and let the function vary with time. We will assume that the function dynamics are Markovian. That is, the state of the function at time t_k is dependent on its state at time t_{k-1} only. Normally, a GP would model both spatial and time dimensions together via the product of spatial and temporal kernels. However, when the temporal dynamics are Markovian, the Kalman filter process model can be used to incorporate the temporal dimension more efficiently. This section describes how GP covariance functions can be incorporated into KF process models. We call the new filter the *KFGP*.

Let the spatial kernel at times t_{k-1} and t_k be K_{k-1} and K_k respectively. The spatial kernel which models the function at both times t_{k-1} and t_k is K:

$$K = \begin{pmatrix} K_k & C_{k-1} \\ C_{k-1}^T & K_{k-1} \end{pmatrix} \quad (7)$$

where C_{k-1} is chosen so that K is positive semi-definite. Now, given the posterior mean, $\bar{f}_{k-1}$, and covariance, P_{k-1}, at time t_{k-1}, we can determine the posterior mean, $\bar{f}_k$, and covariance, P_k, at time t_k using the AKS applied to the temporal dimension:

$$\bar{f}_{k|k-1} = \bar{f}_{k|0} + C_{k-1}K_{k-1}^{-1}(\bar{f}_{k-1|k-1} - \bar{f}_{k-1|0}) ,$$
$$P_{k|k-1} = K_k + C_{k-1}K_{k-1}^{-1}[P_{k-1|k-1} - K_{k-1}]K_{k-1}^{-1}C_{k-1}^T .$$

By defining the KF process model:

$$G_{k-1} \triangleq C_{k-1}K_{k-1}^{-1} \quad (8)$$

and the KF process noise covariance:

$$Q_k \triangleq K_k - C_{k-1}K_{k-1}^{-1}C_{k-1}^T \quad (9)$$

we recover the familiar KF equations. The first equation we recover is the prediction of the posterior state:

$$\bar{f}_{k|k-1} = G_{k-1}\bar{f}_{k-1|k-1} . \quad (10)$$

Secondly, we recover the prediction of the prior state:

$$\bar{f}_{k|0} = G_{k-1}\bar{f}_{k-1|0}$$

and finally, we recover the posterior covariance:

$$P_{k|k-1} = G_{k-1}P_{k-1|k-1}G_{k-1}^T + Q_k .$$

As with the standard Kalman filter, the process noise covariance introduces uncertainty into the state prediction when the process model is weak. That is, the value of Q_k is large when the correlation between f_k and f_{k-1} is weak (i.e.

when C_{k-1} is small). Also, Q_k is large when the difference between the spatial GP prior covariances, K_{k-1} and K_k, is large. Thus, significant changes in the spatial kernel can lead to significant uncertainty in the predicted state.

In general, the process model, G_{k-1}, is a function of the time difference, $t_k - t_{k-1}$, so that $G_{k-1} = G(t_k - t_{k-1})$. The posterior mean at time t_k should not depend on whether it is calculated using (10) directly or, alternatively via some intermediate time $t_{k'}$, where $t_{k-1} < t_{k'} < t_k$. Mathematically, we require:

$$G(t_k - t_{k-1}) = G(t_k - t_{k'})G(t_{k'} - t_{k-1}) . \quad (11)$$

Kernels K, in (7), which satisfy (11) are called *transitive* models. The exponential covariance function satisfies transitivity [13]. We introduce a transitive kernel for the near constant acceleration model (NCAM) in Section 6.1. We are free to choose any KF prediction model, G_{k-1}, provided that the covariance K is positive semi-definite [5] and transitive.

We note that, although the state vector represents the function at a small number of inputs, the AKS can be used to infer the function mean and variance at any input and at any time t_k since the kernel K_k is known. This is despite the fact that the spatial kernel changes over time.

The spatial kernel K_k is governed by a set of hyperparameters whose values change over time t_k. The computational burden of the tracker resides almost entirely in the integration over these hyperparameters which is necessary to calculate the posterior distribution of the process. Consider a one step GP prediction using a sequence of observations obtained at T time instances. Suppose we have N hyperparameter samples at each of the T time instances (drawn using MCMC, for example). This would produce N^T hyperparameter sequences over the full T time period. Using a non-stationary kernel, which is the full T dimensional extension of (7) above, each hyperparameter sequence leads to a $ST \times ST$ prior covariance matrix for a process with a spatial field sampled at S inputs. Since the computational complexity for the inversion of this matrix is of order $\bigcirc((ST)^3)$ then the overall computational complexity for a GP prediction is of order $\bigcirc(N^T(ST)^3)$.

When the process is Markovian a KF implementation can be significantly faster than the equivalent standard GP implementation. The KF makes a prediction from the previous estimate so, over a period of T time steps, the number of KF evaluations over N hyperparameters, is TN. If the KF state vector is of size SV (where V is typically much smaller than T) then the computational complexity for inverting the state covariance matrix is $(SV)^3$ and therefore, the computational complexity of the KF over a T time period is of order $\bigcirc(TN(SV)^3)$. Consequently, the KF can be considerably faster than the equivalent GP for Markovian temporal processes.

---

[5]Note, Q_k is positive semi-definite (a requirement for Q_k to be a true KF process noise covariance) whenever K is positive semi-definite.

6 Applications

We demonstrate the synergy between Gaussian processes and the Kalman filter via two applications.

6.1 Tracking Periodic Dynamics

Time series prediction performs GP regression over the temporal dimension [13, 12]. In this section, we show that KF process models can be incorporated into GP time-series prediction.

The near constant acceleration model (NCAM) was developed in the 1970's [5], chiefly as a model for target tracking within the Kalman filter [1]. The NCAM is a general expression for all second order differentiable functions g which are subject to a random diffusion ϵ:

$$g(x_k) = g(x_{k-1}) + (x_k - x_{k-1})g'(x_{k-1})$$
$$+ \frac{1}{2}(x_k - x_{k-1})^2 g''(x_{k-1}) + \epsilon_k.$$

where x now denotes time and x_k increases in value with increasing index i (i.e. $x_k > x_j$ if and only if $i > j$).

Defining the *state vector* ϕ_k as:

$$\phi_k \triangleq \begin{pmatrix} g(x_k) \\ g'(x_k) \\ g''(x_k) \end{pmatrix}$$

and the temporal difference $\delta_k = x_k - x_{k-1} > 0$ then:

$$\phi_k = G_k\phi_{k-1} + q\epsilon_k \,, \tag{12}$$
$$\bar{\phi}_k = G_k\bar{\phi}_{k-1} \tag{13}$$

where:

$$G_k = G(\delta_k) = \begin{pmatrix} 1 & \delta_k & \frac{\delta_k^2}{2} \\ 0 & 1 & \delta_k \\ 0 & 0 & 1 \end{pmatrix}, \tag{14}$$

$q \geq 0$ is the acceleration scaling coefficient and ϵ_k is assumed to be a zero-mean Gaussian random variable with covariance:

$$Q_k = Q(\delta_k) = \begin{pmatrix} \delta_k^5/20 & \delta_k^4/8 & \delta_k^3/6 \\ \delta_k^4/8 & \delta_k^3/3 & \delta_k^2/2 \\ \delta_k^3/6 & \delta_k^2/2 & \delta_k \end{pmatrix}.$$

Note that, by choosing δ_k sufficiently large, the NCAM can determine the mean and covariance for the process g for *any* future time. Also note that the NCAM is transitive.

Defining:

$$M(\delta) \triangleq \begin{pmatrix} 1 & \delta & \frac{\delta^2}{2} \end{pmatrix},$$
$$N(\delta) \triangleq \begin{pmatrix} \delta^5/20 & \delta^4/8 & \delta^3/6 \end{pmatrix}$$

the near constant acceleration GP kernel $K_{ij}^{NCAM} = \mathrm{Cov}(g(x_i), g(x_j))$ is [14]:

$$K_{ij}^{NCAM} = M(x_i - x_0)\,\mathrm{Cov}(\phi_0, \phi_0)M(x_j - x_0)^T$$
$$+ qM(x_i - x_j)N(x_j - x_0)^T \qquad (x_i \geq x_j)\,.$$

The NCAM GP kernel makes clear the relationship between the Kalman filter parameters, especially q, in (12) and the identical Gaussian process hyperparameters (including q). As a matter of fact, the non-parametric GP model is equivalent to the Kalman filter parametric model in this case. We note that, although ϕ, which is modelled by the KF, is a Markov process, the sub-process g by itself, which is modelled by the GP, is not Markovian. The augmented state vector, ϕ, allows the KF to fit a smooth curve to the data points. In [14], we compare the NCAM model with alternative approximate models obtained using combinations of off-the-shelf Gaussian process kernels. Our results in [14] demonstrate the care required when choosing kernels for Gaussian process trackers.

We demonstrate the efficacy of the NCAM GP on a one dimensional target tracking problem that would be difficult to implement using a Kalman filter [15]. The target moves according to NCAM and is tracked by a range sensor. This sensor is mounted on a platform which is itself subject to a cyclic perturbation in its elevation (see Figure 3). For example, the sensor could be mounted on a small boat which rises and falls with the waves. We assume that the perturbation function is not known a priori and that it comprises multiple modes so that it is not simply sinusoidal.
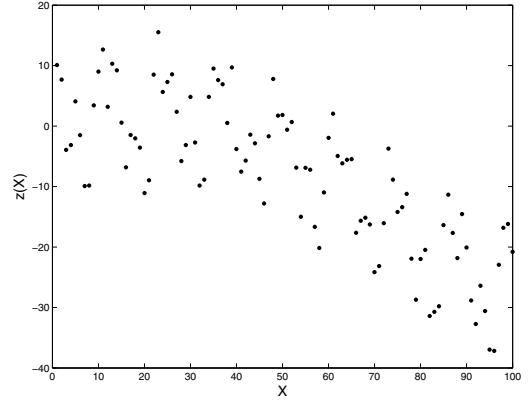

Figure 2: Sensor data for target following NCAM trajectory obtained from platform undergoing periodic motion.

Figure 2 shows a typical data stream for a tracked target which moves according to NCAM and is sensed using an elevation sensor which rises and falls periodically. This data is generated using a target trajectory sampled from the NCAM GP prior and platform dynamics sampled from the following squared-exponential periodic kernel [10, 13]:

$$K^{\mathrm{PER}}(x, x') = \mu_{\mathrm{SEP}} \exp\left(-\frac{2\sin^2\left(\frac{x-x'}{2}\right)}{l_{\mathrm{SEP}}^2}\right) \tag{15}$$

where the input and output scale hyperparameters are chosen to be $l_{\mathrm{SEP}}^2 = 10$ and $\mu_{\mathrm{SEP}} = 50$. The target is sensed at each time step.

When $f(x)$ and $p(x)$ are the target's location and the platform's elevation at time x, respectively, then the sensor's measurement $z(x)$ is:

$$z(x) = f(x) - p(x) + \epsilon_x \qquad (16)$$

where ϵ_x is a zero-mean Gaussian random variable with variance σ^2. In our experiments $\sigma^2 = 5$. We note that the measurement model in (16) differs from the basic GP measurement model in (4) and the functions of interest, namely the target's trajectory and the platform dynamics, are hidden variables. Consequently, the GP equations, (5) and (6), are modified in order to determined the means, $\bar{f}$ and $\bar{p}$, and covariances, $\mathrm{Cov}(f)$ and $\mathrm{Cov}(p)$, of the posterior distributions of the target trajectory and platform dynamics.

In keeping with the thesis of this paper, we will show how the new GP equations can be derived from the corresponding Kalman filter fusion equations for this problem. At time x_k, define the $2k$ state vector as the stacked vector $V = \langle f(1), \ldots, f(k), p(1), \ldots, p(k) \rangle^T$ and the k observation vector z. The $k \times 2k$ observation model is $H = [I \quad -I]$ where I is the $k \times k$ identity matrix. The expected prior elevation is chosen to be zero for both the target and the platform. Further, the prior covariance matrices for times $X = \{x_1, \ldots, x_k\}$ for the target and platform dynamics are $K^{NCAM}(X, X)$ and $K^{PER}(X, X)$, respectively. Since the target and platform dynamics are a priori independent then the prior covariance P over V is:

$$P = \begin{pmatrix} K^{NCAM}(X, X) & 0 \\ 0 & K^{PER}(X, X) \end{pmatrix} .$$

The Kalman filter fusion equations are then:

$$\bar{V}(X) = \Gamma z,$$
$$\mathrm{Cov}(V(X)) = (I - \Gamma H)P$$

where the Kalman gain is:

$$\Gamma = PH^T[HPH^T + \sigma^2 I]^{-1}$$
$$= \begin{pmatrix} K^{NCAM}(X, X) \\ -K^{PER}(X, X) \end{pmatrix} \Sigma^{-1}$$

and:

$$\Sigma = K^{NCAM}(X, X) + K^{PER}(X, X) + \sigma^2 I .$$

Substituting expressions Γ, P and H into the KF fusion equations and then extracting the appropriate rows and columns from $\bar{V}(X)$ and $\mathrm{Cov}(V(X))$ gives the following GP equations, one set to infer the target dynamics and another for the platform dynamics:

$$\bar{f}(X) = K^{NCAM}(X, X)\Sigma^{-1} z ,$$
$$\mathrm{Cov}(f) = K^{NCAM}(X, X) - K^{NCAM}(X, X)\Sigma^{-1}$$
$$\times K^{NCAM}(X, X)^T ,$$
$$\bar{p}(X) = -K^{PER}(X, X)\Sigma^{-1} z ,$$
$$\mathrm{Cov}(p) = K^{PER}(X, X) - K^{PER}(X, X)\Sigma^{-1}K^{PER}(X, X)^T .$$

K^{PER} is the squared exponential periodic kernel in (15).

Figure 3 shows the first standard deviation for the target's location and the platform's elevation posteriors inferred using the NCAM GP and the data in Figure 2. The figure also shows the smoothed trajectory over the entire run obtained using all 100 measurements. Also shown is the ground truth for comparison. The tracker successfully infers both the target's location and the platform's elevation everywhere.
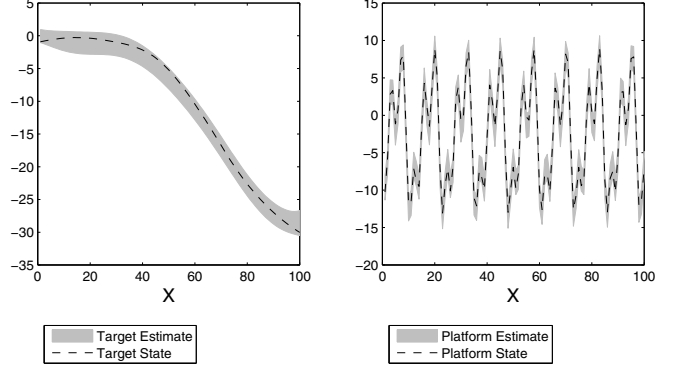

Figure 3: Target location and platform elevation inferred using NCAM GP from data in Figure 2. Also shown is the ground truth.

For further details of the NCAM GP, including proofs, the reader is referred to [14].

6.2 Spatial-Temporal Tracking

We present a novel approach to spatial-temporal field estimation using the KFGP developed in Section 5. The approach tracks dynamically changing fields using a Kalman filter and, at each time instant, the field is defined by a spatial GP kernel. Our approach is similar to the Kriged Kalman filter [11]. However, in [11] the spatial field is modelled by a non-stationary mixture model with a stationary GP residual component. In our approach the spatial field is modelled at each instant entirely by a GP kernel. Our theory supports a drastic change of kernel at any time instant, if required, or a more passive change through gradual hyperparameter drift within a single kernel.

In our experiments we use the square-exponential kernel to model the spatial field. However, in order to adapt to changing function curvature we allow the scale length L to change over time:

$$[K_{k-1}]_{ij} = \exp\left(-\frac{(x_i - x_j)^2}{L_{k-1}^2}\right) ,$$
$$[K_k]_{ij} = \exp\left(-\frac{(x_i - x_j)^2}{L_k^2}\right) .$$

Figures 4 to 6 show instances of our function as it evolves over 16 epochs. Its length scale grows initially before shrinking again after epoch 7.

We may choose any KF prediction model provided that K in (7) is positive semi-definite and transitive. A valid choice for equidistant time points is:

$$C_{k-1} = \frac{1}{1 + \exp \beta_k} \text{Chol}(K_k)\text{Chol}(K_{k-1})^T \qquad (17)$$

where $Chol$ is the Cholesky decomposition and $\beta_k \in \mathbb{R}$. The correlation coefficient $(1 + \exp \beta_k)^{-1}$ is guaranteed to be between 0 and 1. Using (8) and (9) we obtain the process model and process noise covariance:

$$G_{k-1} = \frac{\text{Chol}(K_k)\text{Chol}(K_{k-1})^{-1}}{1 + \exp \beta_k} \ ,$$
$$Q_k = \left(1 - \frac{1}{(1 + \exp \beta_k)^2}\right) K_k \ .$$

The correlation length scale, β, and spatial length scale, L, evolve accordingly:

$$\beta_k = \beta_{k-1} + \omega_k \ ,$$
$$L_k = L_{k-1} + \lambda_k$$

where ω_k and λ_k are zero-mean Gaussian random variables with stationary variances which are parameters of the model. All parameters and hyperparameters are marginalised using MCMC.

At each time-step observations are taken from 7 randomly chosen training points. The figures show the function estimates obtained using three different filters:

1. the left panel shows the combined GP and KF approach (KFGP) developed above.

2. the middle panel shows the estimate calculated using just the current observations with no prediction from previous observations.

3. the right panel shows the effect of neglecting the process model. The function is assumed to be stationary.

Clearly, the excessive uncertainty shown in the middle panels in Figures 4 to 6 compared to the left panels demonstrates that there is significant information carry over by the KFGP over time. Also, neglecting the dynamic nature of the function leads to significant estimate divergence as is evident in the right panels in the figures. The estimate at iteration 16 has diverged completely and is not shown in Figure 6.

These experiments illustrate the efficacy of our spatial-temporal tracker using a very simple process model. The tracker, as implemented, does not model the rate of change of the spatial phenomenon and consequently can lack precision in some problems. A more advanced approach incorporates the NCAM, developed in the previous section, to model both the rate of change and accelerations of the spatial phenomenon. This approach will be presented in a future publication.
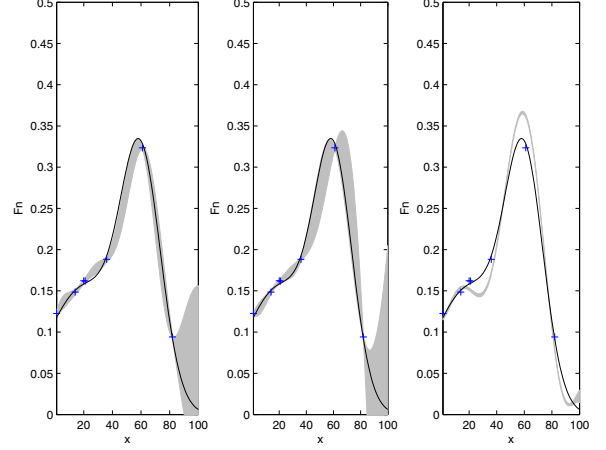

Figure 4: After 4 iterations. The solid line is the true function and crosses '+' are observations at the current epoch. The left graph shows the KFGP estimate. The middle graph is the estimate calculated using just the current observations with no prediction from previous observations (but with parameter learning). The right graph shows the effect of neglecting the process model and noise. The filter for this model assumes that the function is stationary.

7 Conclusions

We have presented an introduction to Gaussian processes for those readers more familiar with the Kalman filter and have explored the close links between GPs and KFs. Consequently, we have shown that our intuitions about the Kalman filter readily lend themselves to Gaussian processes. We have demonstrated how GP kernels can be incorporated into the Kalman filter and have also determined the form for KF process models in terms of these kernels. Consequently, we have developed a novel spatial-temporal field estimator. We have also demonstrated how the near constant acceleration model, commonly used within the Kalman filter, can be incorporated into the GP kernel repertoire.

Acknowledgments

References

[1] Y. Bar-Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic Press, 1988.

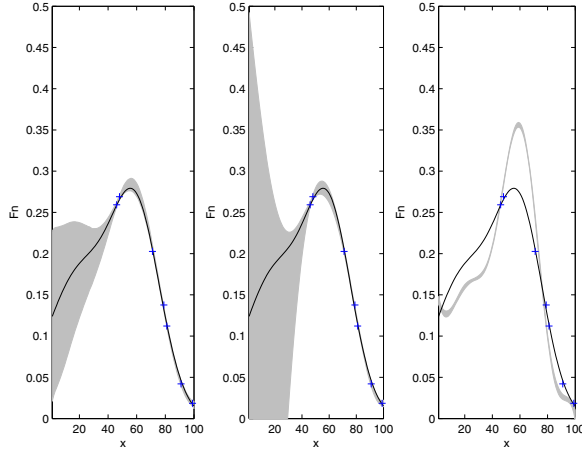[2] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

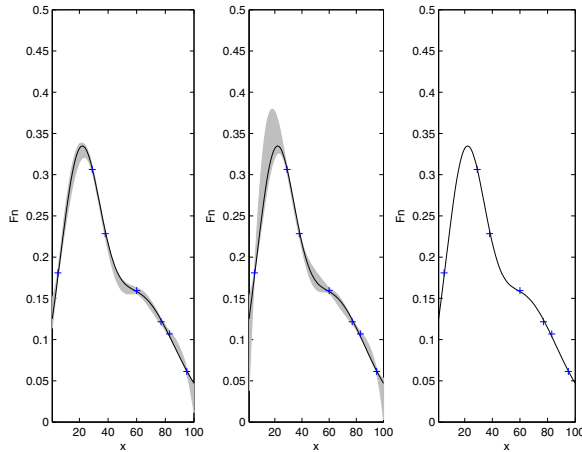Figure 5: After 7 iterations. Explanation as in Figure 4.

Figure 6: After 16 iterations. Explanation as in Figure 4.

[3] B. Ferris, D. Haehnel, and D. Fox. Gaussian processes for signal strength-based location estimation. *Robotics: Science and Systems*, 2006.

[4] C. Guestrin, A. Krause, and A. Singh. Near-optimal sensor placements in gaussian processes. In *22nd International Conference on Machine Learning (ICML)*, pages 265–272. ICML, 2005.

[5] A. H. Jaswinski. *Stochastic Processes and Filtering Theory*. New York: Academic Press, 1970.

[6] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *In Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, pages 182–193, 1997.

[7] S. J. Julier and J. K. Uhlmann. A non-divergent estimation algorithm in the presence of unknown correlations. In *Proceedings of the 1997 American Control Conference (ACC'97), Albuquerque, New Mexico*, pages 2369–2373, 1997.

[8] J. Ko, D. J. Klein, D. Fox, and D. Haehnel. Gp-ukf: Unscented kalman filters with gaussian process prediction and observation models. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2007*, pages 1901–1907, 2007.

[9] D. J. Leith, M. Heidl, and J. V. Ringwood. Gaussian process prior models for electrical load forecasting. In *Eighth International Conference on Probabilistic Methods Applied to Power Systems, Iowa State University, Ames, Sept. 12–16*, 2004.

[10] D. J. C. Mackay. Introduction to gaussian processes. In C. M. Bishop, editor, *Neural Networks and Machine Learning*. Springer-Verlag, 1998.

[11] K. Mardia, C. Goodall, E. Redfern, and F. Alonso. The kriged kalman filter (with discussion). *TEST: An Official Journal of the Spanish Society of Statistics and Operations Research*, 7(2):217–282, December 1998.

[12] M. Osborne, A. Rogers, A. Ramchurn, S. Roberts, and N. R. Jennings. Towards real-time information processing of sensor network data using computationally efficient multi-output gaussian processes. In *IPSN 2008: International Conference on Information Processing in Sensor Networks, St. Louis, Missouri*, 2008.

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

[14] S. Reece and S. Roberts. The near constant acceleration model gaussian process kernel. *IEEE Transactions on Signal Processing Letters*, 2010. In press.

[15] S. Särkkä, A Vehtari, and J Lampinen. Prediction of estsp competition time series by unscented kalman filter and rts smoother. In *Proceedings of the European Symposium on Time Series Prediction (ESTSP), Espoo, Finland*, 2007.

A   The Augmented Kalman Smoother (AKS)

Partition the multivariate Gaussian random variable f thus:

$$f = \begin{pmatrix} f_A \\ f_B \end{pmatrix} \ .$$

Define the prior mean:

$$\bar{f} = \begin{pmatrix} \bar{f}_A \\ \bar{f}_B \end{pmatrix}$$

and the prior covariance:

$$K = \begin{pmatrix} K_{A,A} & K_{A,B} \\ K_{A,B}^T & K_{B,B} \end{pmatrix} \ .$$

Also, let $\bar{f}_{B|z_B}$ and $P_{B|z_B}$ be the posterior mean and covariance of f_B, respectively. If z_B is conditionally independent

of f_A given f_B then the posterior mean, $\bar{f}_{|z_B}$ and covariance, $P_{|z_B}$, of f is:

$$\bar{f}_{|z_B} = \begin{pmatrix} \bar{f}_A + K_{A,B}K_{B,B}^{-1}(\bar{f}_{B|z_B} - \bar{f}_B) \\ \bar{f}_{B|z_B} \end{pmatrix}$$

and:

$$P_{|z_B} = K + \begin{pmatrix} K_{A,B}K_{B,B}^{-1} \\ I \end{pmatrix} [P_{B|z_B} - K_{B,B}] \begin{pmatrix} K_{B,B}^{-1}K_{B,A} \\ I \end{pmatrix}^T .$$

Proof: Using Bayes' rule, when z_B is conditionally independent of f_A given f_B:

$$p(f \mid z_B) = p(f \mid f_B)p(f_B \mid z_B) .$$

The Gaussian mean and covariance of f conditioned on f_B are:

$$\bar{f}_{|B} = \bar{f} + K_{:,B}K_{B,B}^{-1}(f_B - \bar{f}_B) , \qquad (18)$$
$$P_{|B} = K - K_{:,B}K_{B,B}^{-1}K_{:,B}^T \qquad (19)$$

where:

$$K_{:,B} \triangleq \begin{pmatrix} K_{A,B} \\ K_{B,B} \end{pmatrix} .$$

Thus:

$$\begin{aligned}
\bar{f}_{|z_B} &= \int df_B \, \bar{f}_{|B} p(f_B \mid z_B) \\
&= \bar{f} + K_{:,B}K_{B,B}^{-1}(\bar{f}_{B|z_B} - \bar{f}_B) \qquad (20) \\
&= \begin{pmatrix} \bar{f}_A + K_{A,B}K_{B,B}^{-1}(\bar{f}_{B|z_B} - \bar{f}_B) \\ \bar{f}_{B|z_B} \end{pmatrix} .
\end{aligned}$$

Combining (18) and (20):

$$\bar{f}_{|B} - \bar{f}_{|z_B} = K_{:,B}K_{B,B}^{-1}(f_B - \bar{f}_{B|z_B})$$

and therefore:

$$\begin{aligned}
\int df_B \, p(f_B \mid z_B) \, &(\bar{f}_{|B} - \bar{f}_{|z_B})(\bar{f}_{|B} - \bar{f}_{|z_B})^T \\
&= K_{:,B}K_{B,B}^{-1}P_{B|z_B}K_{B,B}^{-1}K_{:,B}^T .
\end{aligned}$$

Thus, with reference to (19):

$$\begin{aligned}
P_{|z_B} &= \int df_B \, p(f_B \mid z_B) \, [P_{|B} + (\bar{f}_{|B} - \bar{f}_{|z_B})(\bar{f}_{|B} - \bar{f}_{|z_B})^T] \\
&= K + K_{:,B}K_{B,B}^{-1}[P_{B|z_B} - K_{B,B}]K_{B,B}^{-1}K_{:,B}^T \\
&= K + \begin{pmatrix} K_{A,B}K_{B,B}^{-1} \\ I \end{pmatrix} [P_{B|z_B} - K_{B,B}] \begin{pmatrix} K_{B,B}^{-1}K_{A,B}^T \\ I \end{pmatrix}^T .
\end{aligned}$$

□