Abstract—In this work we study the problem of non-parametric estimation for non-linear time-space dynamic stochastic processes and in particular for Gaussian processes (GP). GP methods have been mainly applied to spatial regression and represent the state of the art for machine learning thanks to their universal representing properties. However, their extension to dynamic processes has been elusive so far since standard machine learning tools give rise unscalable algorithms. In this work we propose a systematic and explicit procedure to address this problem by pairing GP regression with Kalman Filtering. In particular, under the specific time-space separability assumption of the kernel which models process and periodic sampling on a (possibly non-uniform) space-grid, we show how to build a finite dimensional discrete-time state-space exact representation for the modeled process. The major finding is that the state at instant k of the associated Kalman Filter represents a sufficient statistic to compute the minimum variance prediction of the process at instant k over any arbitrary finite subset of the space. The proposed strategy is then compared with the standard non-parametric estimation and truncated non-parametric estimation strategies both in terms of estimation performance and computation complexity.

Index Terms—Gaussian regression, machine learning, Kalman filtering, spatio-temporal Gaussian processes.

I. INTRODUCTION

Gaussian process-based (GP) regression [1] is a Bayesian learning framework where GP are used as nonparametric priors for regressors functions. Nowadays, GP based methods have heavily increased their popularity [2], [3] in disciplines such as statistical inference and machine learning [3]. In the classical machine learning setup the modeled process is considered static. Consequently, classical GP based regression, i.e., Kriging [4], often assumes as input variables just spatial locations. Nevertheless, the method can be extended to learn spatio-temporal processes by treating the time variable as an additional input feature [3]. In dynamical scenarios however, the classical GP based regression paradigm presents practical limitations. These are mainly due to the heavy memory and computational requirements which grow cubical as the number of input data. As second drawback, the classical GP based regression methods have heavily increased their popularity [2], [3] in disciplines such as statistical inference and machine learning [3].

In this section we briefly recall the required preliminaries on nonparametric estimation, Kalman filtering and spectral factorization.
A. Nonparametric Estimation

In this section we review some fundamental aspects regarding the nonparametric Gaussian regression.

Let \( f : \mathcal{A} \rightarrow \mathbb{R} \) be a zero-mean Gaussian field with covariance, also called kernel, \( K : \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R} \), where \( \mathcal{A} \) is a compact set. Assume to have a set of \( N \in \mathbb{N}_{>0} \) noisy measurements of the form

\[
y_i = f(a_i) + v_i,
\]

where \( v_i \) is a zero-mean Gaussian noise with variance \( \sigma^2 \), i.e. \( v_i \sim \mathcal{N}(0, \sigma^2) \), independent from the unknown function. Given the data set of input locations \( \{a_i, y_i\}_{i=1}^N \), it is known [12], [2] the estimate \( \hat{f}(\cdot) \) of \( f \) is a linear combination of the kernel sections \( K(a_i, \cdot) \), i.e., the kernel sampled in the values corresponding to the available input locations. In particular, for any \( a \in \mathcal{A} \), it holds that

\[
\hat{f}(a) := \mathbb{E}[f(a)|\{a_i, y_i\}_{i=1}^N] = \sum_{i=1}^N c_i K(a_i, a).
\]

The expansion coefficients \( c_i \) are obtained as

\[
\begin{bmatrix}
  c_1 \\
  \vdots \\
  c_N
\end{bmatrix} = (\bar{K} + \sigma^2 I)^{-1}
\begin{bmatrix}
  y_1 \\
  \vdots \\
  y_N
\end{bmatrix}, \quad \bar{K} \in \mathbb{R}^{N \times N}, \quad [\bar{K}]_{ij} = K(a_i, a_j),
\]

where \( I \) denotes the identity matrix of suitable dimension and \( [\bar{K}]_{ij} \) denotes the \( i \)-th entry of the matrix \( \bar{K} \). Finally, the posterior variance of \( \hat{f}(a) \) evaluated at the generic location \( a \in \mathcal{A} \) is given by

\[
V(a) = \text{Var}[f(a)|\{a_i, y_i\}_{i=1}^N] = K(a, a) - 
\begin{bmatrix}
  K(a_1, a) & \cdots & K(a_N, a)
\end{bmatrix} \begin{bmatrix}
  \bar{K} + \sigma^2 I
\end{bmatrix}^{-1}
\begin{bmatrix}
  K(a_1, a) \\
  \vdots \\
  K(a_N, a)
\end{bmatrix}.
\]

Clearly, because of the matrix inversion in both (3) and (4), the method scales as \( O(N^3) \). Moreover, in real-time applications, where a certain number of measurements are collected at each iteration, all the past measurements must be kept in memory. Thus, the method is more suitable for a batch, almost static implementation rather than an iterative time-varying one.

Remark 1 (Spatio-temporal processes): In the following we consider spatio-temporal processes. Conversely to classical Gaussian processes, where \( a \) usually denotes a spatial variable, in spatio-temporal processes \( a \) represents both time and space. Hence, without loss of generality, we can write \( f(a) = f(x, t) \). Accordingly, the domain \( \mathcal{A} \) can be decomposed as \( \mathcal{A} := \mathcal{X} \times \mathbb{R}_+ \), with \( \mathcal{X} \) and \( \mathbb{R}_+ \) denoting the spatial and temporal domain, respectively.

B. Kalman Filtering

In this section we briefly recall some basic notions on Kalman filtering for finite-dimensional discrete-time linear state-space dynamical systems [13].

Consider the following system

\[
\begin{align}
 s_{k+1} &= A s_k + w_k, \\
 y_k &= C_k s_k + v_k,
\end{align}
\]

where, at each iteration \( k \), \( s_k \in \mathbb{R}^n \) is the state vector, \( y_k \in \mathbb{R}^m \) is the output vector, \( w_k \in \mathbb{R}^n \) and \( v_k \in \mathbb{R}^m \) are i.i.d. zero-mean Gaussian random vectors with covariance matrices \( Q \geq 0 \) and \( R > 0 \), respectively. \( A \in \mathbb{R}^{m \times n} \) is the state matrix and \( C_k \in \mathbb{R}^{m \times n} \) is the time-varying output matrix. As commonly done, we assume both the process and measurement noise to be uncorrelated with respect to each other, i.e. \( \mathbb{E}[w_k v_k] = 0 \forall k \). We also assume the initial condition \( s_0 \) is drawn from a Gaussian distribution with zero mean and covariance \( \Sigma_0 \), i.e., \( s_0 \sim \mathcal{N}(0, \Sigma_0) \).

The Kalman Filter applied to the system (5) is described by the following recursive equations

\[
\begin{align}
\hat{s}_{k+1|k} &= A \hat{s}_{k|k} \quad (6a) \\
\Sigma_{k+1|k} &= \Sigma_{k|k} A^T + Q \quad (6b) \\
\hat{s}_{k+1|k+1} &= \hat{s}_{k+1|k} + L_{k+1} (y_{k+1} - C_k \hat{s}_{k+1|k}) \\
\Sigma_{k+1|k+1} &= \Sigma_{k+1|k} - L_{k+1} C_k \Sigma_{k+1|k} \\
L_{k+1} &= \Sigma_{k+1|k} C_k^T (C_k \Sigma_{k+1|k} C_k^T + R)^{-1} \quad (6c)
\end{align}
\]

where \( \hat{s}_{k|k} \) and \( \Sigma_{k|k} \) represent the filtered estimate of the state and the posterior error covariance, respectively; \( \hat{s}_{k+1|k} \) and \( \Sigma_{k+1|k} \) represent the (one step) predicted state estimate and error covariance, respectively; \( L_{k+1} \) is the Kalman gain; finally, the filter is initialized assuming \( s_{0|0} = \mathbb{E}[s_0] = 0 \) and \( \Sigma_{0|0} = \text{Cov}(s_0) = \Sigma_0 \).

We recall that, under the assumptions of normal distributed noises and perfect model knowledge, the Kalman filter is optimal, in mean square sense. Then, equations (6) return the minimum mean square error estimate of the state, which corresponds to

\[
\hat{s}_k = \mathbb{E}[s_k|y_0, \ldots, y_k],
\]

that is, the estimate of the state given all the measurements up to the \( k \)-th one. Moreover, under the Markovianity (memory-less of the system) property of the state, it holds that

\[
\mathbb{E}[s_k|y_0, \ldots, y_k] = \mathbb{E}[s_k|s_{k-1}, y_k],
\]

that is, the previous state and the last measurement represent the sufficient statistic to compute the optimal estimate of the state at the current time instant.

Finally, it is well known, [14], that if we assume the output matrix constant, i.e. \( C_k = C \), under the hypothesis of stabilizability of the pair \( (A, Q) \) and detectability of the pair \( (A, C) \) the estimation error covariance of the Kalman filter converges to a unique value from any initial condition.

C. Spectral factorization of random processes

Here, we recall some notions about spectral factorization of random processes and realization theory. In particular we want to show how, specific class of processes admits an equivalent exact state-space representation.

Consider a stationary random process \( f(t) \) with covariance \( h(\tau) \). Thanks to the Wiener-Khinchin theorem, it is known that the power spectral density (PSD) of the process is equal to the Fourier transform of its covariance \( h \), i.e.,

\[
S(\omega) := \mathbb{E}[h(\tau)](\omega).
\]
Moreover, in the particular case when \( S = S_t \) is rational of order \( 2r \), thanks to spectral factorization \([15]\), its PSDs can be rewritten as \( S_t(\omega) = W(\omega)W(-\omega) \) with
\[
W(\omega) = \frac{b_{r-1}(i\omega)^{-1} + b_{r-2}(i\omega)^{-2} + \cdots + b_0}{(i\omega)^{r} + a_{r-1}(i\omega)^{-1} + \cdots + a_0}.
\] (7)

If necessary, to obtain the form (7) numerator and denominator coefficients of \( W \) are expanded and scaled. Finally, from realization theory, we have that rational functions of the form (7) are in correspondence to the equivalent continuous time state space representation \([16]\) (companion form) given by
\[
\begin{bmatrix}
    s_i = F s_i + G w_t \\
    z_t = H s_i
\end{bmatrix}
\]
where \( w_t \sim \mathcal{N}(0, I) \), the model matrices are equal to
\[
F = \begin{bmatrix}
    0 & 1 & 0 & \ldots & 0 \\
    0 & 0 & 1 & \ldots & 0 \\
    \vdots & \ddots & \ddots & \ddots & 0 \\
    -a_0 & -a_1 & -a_2 & \ldots & -a_{r-1} \\
\end{bmatrix}, \quad G = \begin{bmatrix}
    0 \\
    0 \\
    \vdots \\
    1
\end{bmatrix},
\]
\[
H = [b_0 \ b_1 \ b_2 \ \ldots \ \ b_{r-1}],
\]
and the initial state is \( s_0 \sim \mathcal{N}(0, \Sigma_0) \), with \( \Sigma_0 \) computed as solution of the Lyapunov equation \( FX + XF^T + GG^T = 0 \).  

\[\Box\]

III. MAIN CONTRIBUTION

Here, we present the main contributions of the paper. First, we state the problem at hand. Second, we formally show how to build an exact state space representation for a certain class of GPs. Then, we bridge GP regression and Kalman filtering, providing a clear and systematic methodology to implement the filter. As it will be clear, we first focus on estimating the process of interest over an “observable” finite collection of points. Finally, we show how to extend the estimation over an arbitrary “unobservable” finite collection of locations.

A. Problem Formulation

Consider a function \( f: \mathcal{X} \times \mathbb{R} \rightarrow \mathbb{R} \) modeled as a zero-mean Gaussian Process with covariance \( K \). Hereafter, for the sake of notation ease, we use \( f_t(x) \) instead of \( f(x,t) \). We assume \( \mathcal{X} \) to be a finite collection of points, i.e.,
\[
\mathcal{X} := \{ x_1, \ldots, x_n \ | \ x_i \in \mathbb{R}^d \}.
\]
We assume noisy measurements of the form (1) come from a subset \( \mathcal{X}_{\text{meas}} \subseteq \mathcal{X} \) of given locations. We formally define \( \mathcal{X}_{\text{meas}} \) as follows.

Definition 2 (Measurements Space): Consider the finite set \( \mathcal{X} \). We denote with \( \mathcal{X}_{\text{meas}} \subseteq \mathcal{X} \) a finite collection of points containing \( M \leq N \) locations from \( \mathcal{X} \), i.e.
\[
\mathcal{X}_{\text{meas}} := \{ x_1, \ldots, x_M \ | \ x_i \in \mathcal{X} \}.
\]

\(^1\)Observe that, by following the notation used in Definition 2, we have that the first \( M \) points of \( \mathcal{X} \) represents the measurements locations. This holds without loss of generality assuming \( \mathcal{X} \) has been a priori ordered.

Precisely, to consider the most general case, we assume to be able to collect the measurements at discrete time instants \( t = kT \), where \( T \) denotes the sampling time, only from a time-varying subset of locations, namely \( \mathcal{M}(k) \subseteq \mathcal{X}_{\text{meas}} \). The problem we want to solve is that of estimating \( f \) over the entire “partially observable” domain \( \mathcal{X} \), exploiting measurements coming from the “observable” set \( \mathcal{X}_{\text{meas}} \). The problem could arise in diverse applications, e.g., in weather forecasting where, given a small finite set of weather stations which are able to collect measurements at certain discrete time instants, the goal is to estimate the weather conditions on a larger area.

To state our solution, we restrict the analysis on a specific yet sufficiently rich class of kernel functions.

Assumption 3 (Generating Kernel properties): The kernel function \( K \), covariance of the Gaussian process \( f_t(x) \), is separable in time and space and it is stationary in time, namely,
\[
K(x,x',t,t') = K_s(x,x')h(\tau), \quad \tau = t' - t.
\]
In addition, the power spectral density \( S_t(\omega) \) of \( h(\tau) \) is a rational function of order \( 2r \).  

Differently from \([10], [11]\), in Assumption 3 we do not require space stationarity of \( K_s \) but only time stationarity. This let us consider a wider class of generating space kernels \( K_s \), e.g., kernel spline.

Our solution consists of two steps: first we show how to estimate the process \( f_t \) over \( \mathcal{X}_{\text{meas}} \) (Section III-B). Then,
we extend our result to obtain a prediction of the process outside \( \mathcal{X}_{\text{meas}} \) (Section III-C). Precisely, we show how our first solution can be exploited to reconstruct \( f_t \) on the set \( \mathcal{X}_{\text{pred}} \) where

\[
\mathcal{X}_{\text{pred}} := \mathcal{X} / \mathcal{X}_{\text{meas}} , \quad (P := |\mathcal{X}_{\text{pred}}| = N - M).
\]

Figure 1 shows an example of spatio-temporal sampling, as well as the measurements collection process over time.

B. Kalman Regression on \( \mathcal{X}_{\text{meas}} \)

To implement the Kalman equations (6), the first step is to build a state space representation for the Gaussian process \( f_t \). In particular, we are interested in reconstructing \( f_t \) over the “observable” \( \mathcal{X}_{\text{meas}} \). To compactly represent the process over \( \mathcal{X}_{\text{meas}} \), it is convenient to define the vector

\[
\mathbf{f}_t := [f_t(x_1), ..., f_t(x_M)]^T ,
\]

The next proposition exploits Assumption 3 and the state-space realization for rational PSD given in (8) to show that the process \( \mathbf{f}_t \) admits an equivalent exact continuous-time state-space representation.

**Proposition 4 (Equivalent CT-SS representation for \( \mathbf{f}_t \)):** Consider the process \( \mathbf{f}_t : \mathcal{X}_{\text{meas}} \times \mathbb{R}_+ \rightarrow \mathbb{R}^M \). Assume the generating kernel \( K \) satisfies Assumption 3. Let the triplet \((F,G,H)\) be a state-space representation for \( S_t(\alpha) \) as described in II-C. Then, \( \mathbf{f}_t \) admits the following strictly proper state-space representation

\[
\begin{aligned}
S_j&: \begin{cases} 
\dot{s}_j^t &= F s_j^t + G w_t^j, \\
z_j^t &= H s_j^t 
\end{cases}, & j \in \{1, ..., M\}, \\
\mathbf{f}_t &= K_s^{1/2} \mathbf{z}_t 
\end{aligned}
\]

where \( j \) is an index cycling through all the input locations of \( \mathcal{X}_{\text{meas}} \); where \( \mathbf{z}_t := [z_1^t, ..., z_M^t]^T \) and \( K_s \in \mathbb{R}^{M \times M} \) is obtained sampling \( K_s \) over \( \mathcal{X}_{\text{meas}} \); and \( w_t^j \) and \( s_0^j \) are defined as for system (8).

**Proof:** First of all, notice that the process \( \mathbf{f}_t \) is a Gaussian process since it is the solution of a linear differential equation driven by Gaussian noise \( w_t \). To conclude the proof we need to show that the covariance of \( \mathbf{f}_t \) is indeed \( \bar{K} = K_h^{(t)} \). As previously shown, the first two equations of model (9) are the state space representation of the rational power spectral density \( S(\omega) \) thus \( \mathbb{E} [\mathbf{f}_t \mathbf{f}_s^T] = K_h^{(t)} \) it follows that

\[
\mathbb{E} [\mathbf{f}_t \mathbf{f}_s^T] = K_h^{(t)} \mathbb{E} [K_h^{(t)}]^T = K_h^{(t)}.
\]

Observe that the subsystems \( S_j \) in (9) are independent one from each other in the sense that one can easily verify that \( \mathbb{E} [\mathbf{f}_t \mathbf{f}_s^T] = 0 \ \forall \ t, s \neq j \). Basically, Proposition 4 states that, for each location \( x_j \in \mathcal{X}_{\text{meas}} \), the time evolution of \( f_t \) admits a state space representation given by the system \( S_j \) in equation (9). Then, these state space representations are “combined” through the sampled spatial kernel \( K_s \) to build a representation for the overall process \( \mathbf{f}_t \). Observe that Proposition 4 gives a continuous-time state-space representation for the process. However, the Kalman equations (6) are defined in discrete time. Thus, in the following we show how to reconstruct an estimate \( \hat{\mathbf{f}}_k \) of \( \mathbf{f}_t \) at discrete time instants, say \( t = kT \), defined as

\[
\hat{\mathbf{f}}_k := \mathbb{E} \left[ \mathbf{f}_t \mathbf{1}_{\{j \mid x_j \in \mathcal{M}(t), \ell = 0, ..., K\}} \right].
\]

Finally, we recall from Section III-A that we assume to collect measurements coming from \( \mathcal{M}(k) \) at \( t = kT \), i.e.,

\[
y_k^j = f_k(x_j) + \nu_k^j, \quad x_j \in \mathcal{M}(k) , \nu_k^j \sim \mathcal{N}(0, \sigma^2).
\]

Then, given the state-space model of Proposition 4 and a measurement model (11), we have all the necessary elements to bridge GP regression and Kalman filtering on \( \mathcal{X}_{\text{meas}} \).

**Proposition 5 (Kalman regression on \( \mathcal{X}_{\text{meas}} \)):** Assume Assumption 3 holds. Moreover, assume to collect periodic measurements of the form (11) at every \( t = kT \). Then, the estimate \( \hat{\mathbf{f}}_k \) of \( \mathbf{f}_t \) is given by

\[
\hat{\mathbf{f}}_k = K_h^{1/2} \mathbf{H} \bar{S}_k , \quad \mathbf{H} := \text{blkdiag}(H, ..., H) , \bar{S}_k = \text{output of the time-varying Kalman filter (6) applied to the discrete-time system (5) with matrices } (A,C, Q, R) \text{ where } A := \text{blkdiag}(\bar{F}, ..., \bar{F}) , \bar{Q} := \text{blkdiag}(\bar{Q}, ..., \bar{Q}) , \text{being } \bar{F} \text{ and } \bar{Q} \text{ defined as}
\]

\[
\bar{F} = e^{\bar{F} T} , \quad \bar{Q} = \int_0^T (e^{ar{F} t}) \mathbf{GG}^T (e^{ar{F} t})^T dt.
\]

\(^2\text{In the following, for brevity, we might drop the sampling time } T \text{ from } kT \text{ and use just } k \text{ to denote the corresponding discrete time instant.}
and where $R := \sigma I$ and $C_k := l_k \bar{K}_s^{1/2} \mathbf{H}$, being $l_k \in \{0, 1\}^{M_k \times M}$ the matrix $K$ is initialized as $\bar{s}_{0|-1} = 0$, $\Sigma_{0|-1} = \text{blkdiag}(\Sigma_0, \ldots, \Sigma_0)$, where $\Sigma_0$ is solution of the Lyapunov equation $FX + X^TF^T + GG^T = 0$.

Proof: The proof directly follows from the discretization of the CT-SS models $S_t$ of Proposition 4. Once the overall system discrete system, with space vector $s_k := \left[(s_k^1)^T, \ldots, (s_k^M)^T\right]^T$, is rewritten in compact matrix form, the Kalman equations (6) straightforward apply.

Figure 2 shows a representation of the process and of the measurements formations. All the bold symbols refer to vector notation and are obtained stacking in vector form the corresponding non-bold symbols. Additionally, the upper part of Figure 3 shows a block diagram of the estimation scheme. Observe that the block “Time Varying KF”, which implements the Kalman equations (6), is the only time-varying block. This is due to the time-varying measurements. Consequently, if the $C_k$ matrix is constant, the Kalman gain converges to a constant which can be computed offline. In this case the filtering correspond to a static matrix multiplication hence, the computational burden (see Section IV) is alleviated.

To conclude this section, we present an exhaustive example to help the reader’s intuition in the building process of the presented estimation procedure.

**Example 6:** Consider the exponential time kernel $k(t)$:

$$h(t) = \lambda e^{-\sigma \tau}$$

satisfying Assumption 3 since its PSD $S_r$ is equal to

$$S_r(\omega) = \frac{\sqrt{2\lambda} \sigma}{(\sigma + i\omega) (\sigma - i\omega)}$$

which is rational of order 2. Now, consider a zero-mean Gaussian process $f(t)$ with covariance

$$K(x, x', \tau) = K_s(x, x') h(\tau) = e^{-\sigma (x_1 - x_2)^2/2} \lambda e^{-\sigma \tau}$$

that is, a Gaussian spatial kernel and a exponential time kernel. Thanks to Proposition 4, since $K$ satisfies Assumption 3, $f(t)$ admits a state space representation. In particular, given $S_r$ as in (13) with

$$W(i\omega) = \frac{\sqrt{2\lambda} \sigma}{(\sigma + i\omega)},$$

it is easy to see the state-space model matrices are equal to

$$F = -\sigma, \quad H = \sqrt{2\lambda} \sigma, \quad G = 1,$$

while the matrix $\bar{K}_s^{1/2}$ is computed as the Cholesky factorization of the sampled kernel $\bar{K}_s$.

Finally, as stated in Proposition 5, the discrete time state-space representation for $f_k$ is given by

$$F = e^{-\sigma T}, \quad H = \sqrt{2\lambda} \sigma, \quad \bar{Q} = \int_0^T e^{-2\sigma \tau} d\tau.$$ 

To conclude the example we show one case when $h(\cdot)$ does not satisfy Assumption 3. Indeed, considering the squared exponential (Gaussian) kernel defined as

$$h(\tau) = \lambda e^{-\sigma \tau^2},$$

it can be seen its power spectral density is not rational,

$$S(\omega) = \sqrt{\frac{\lambda}{\sigma}} e^{-\left(\frac{\omega}{2\sigma}\right)^2}.$$ (15)

It is worth noticing that, in cases when the PSD $S$ of $h$ is not rational it is always possible to build a rational PSD $\hat{S}$ which approximate the true one. Different approximating methods can be used, e.g., Taylor series expansion or Pade approximation. This leads to an approximate state-space model for $f$. In Section V we present some simulations and, as it will be explained, to approximate $S$ we use a different approach. Indeed, the $\hat{S}$ is computed as the solution of a suitable non-linear weighted least-squares problem.

**C. Kalman Regression on $\mathcal{X}^\text{pred}$**

Here we extend the result of Proposition 5 to build an estimate of the process $f_kT$ over the “prediction” space $\mathcal{X}^\text{pred}$ as defined at the end of Section III-A.

To this end, let

$$\eta_k := f_k(\mathcal{X}^\text{pred}),$$

be the vector representing the process $f_kT$ sampled over $\mathcal{X}^\text{pred}$. Finally, we introduce the following symbols

$$\hat{\eta}_k := \mathbb{E}\left[\eta_k | \{x_j, y_j\}, x_j \in \mathcal{M}(\ell), \ell = 0, \ldots, k\right],$$

$$\Gamma = \text{Cov}(\eta_k, f_k) = \bar{K}_s(\mathcal{X}^\text{pred}, \mathcal{X}^\text{meas}),$$

$$\bar{V}_\eta = \text{Var}(\eta_k) = \bar{K}_s(\mathcal{X}^\text{pred}, \mathcal{X}^\text{pred}),$$

where $\bar{K}_s(\cdot, \cdot)$ denotes the kernel $K_s$ evaluated in all the locations contained in its arguments.

**Proposition 7 (Kalman Regression on $\mathcal{X}^\text{pred}$):** Consider the process $f_k : \mathcal{X} \times \mathbb{R}_+ \mapsto \mathbb{R}$ generated by the kernel $K$ satisfying Assumption 3. Then, the estimate $\hat{\eta}_k$ of $\eta_k$ is given by

$$\hat{\eta}_k = \Psi f_k,$$ (16)

where $\Psi := \Gamma \bar{K}_s^{-1}$. The posterior variance is given by

$$\text{Var}\left(\eta_k | \{x_j, y_j\}, x_j \in \mathcal{M}(\ell), \ell = 0, \ldots, k\right) =$$

$$\bar{V}_\eta - \Gamma \left(\bar{K}_s^{-1} - \bar{K}_s^{-1} (\bar{K}_s^{-1} + \bar{R}^{-1})^{-1} \bar{K}_s^{-1}\right) \Gamma^T.$$ (17)

Proof: Since $K$ satisfies Assumption 3 then, we have that $K(x, x', \tau) = K_s(x, x') h(\tau), \tau = (k-j)T$, and, without loss of generality, we can assume $h(0) = 1$. Then, it holds that $\text{Var}(f_k) = \bar{K}_s$. In the following we drop the sampling time $T$ from the notation; moreover, we assume to be at time instant $k$, while $j < k$ represents a generic previous time instant.

Now, let be $\varphi_k := [f_k^T, \ldots, f_{k-j-1}^T]^T$. Moreover, for brevity instead of $h(k-j)$ we use the simpler $h_{k-j}$. Hence, it can be seen that $\text{Cov}(f_k, \eta_k) = h_{k-j} \Gamma^T$.

We first study $p(\varphi_k, \eta_k | f_k)$. For the conditional variance, we have that

$$\text{Var}(\varphi_k, \eta_k | f_k) = \text{Var}(\varphi_k^T \eta_k^T) -$$

$$\text{Cov}(\varphi_k^T, \eta_k^T) \text{Var}(f_k)^{-1} \text{Cov}(f_k^T, \varphi_k^T \eta_k^T)^T.$$ (17)
Then, we have that

\[ \text{Lemma 1 combined with the conditional independence} \]

the third comes from the definition (10) of \( \sigma \)
on a larger

where the first equality holds because we are conditioning

so

\[ \text{of (17) has the following structure} \]

It is easy to see that the second term of the right hand side

of (17) has the following structure

\[
\begin{bmatrix}
    K_s & h_1 K_s & h_2 K_s & \cdots & h_{k-1} \Gamma^T \\
    K_s^T & h_1^T K_s & h_2^T K_s & \cdots & h_{k-2} \Gamma^T \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    \Gamma h_{k-1}^T & \Gamma h_{k-2}^T & \cdots & h_1 \Gamma^T & * \\
\end{bmatrix}
\]

Hence, by subtracting it to the first term, i.e., \( \text{Var}(\phi_k^T \eta_k^T \Gamma^T) \), the last column and the last row cancel out (except for the diagonal block). This means that \( \phi_k \) and \( \eta_k \) are conditionally independent given \( \bar{f}_k \). Thus we have that

\[ p(\phi_k, \eta_k | \bar{f}_k) = p(\phi_k | \bar{f}_k) p(\eta_k | \bar{f}_k). \] (18)

Thank to this we can write

\[
p(\eta_k | \phi_k, \bar{f}_k) \overset{\text{Bayes}}{=} p(\phi_k, \eta_k | \bar{f}_k) p(\bar{f}_k) = p(\phi_k | \bar{f}_k) p(\eta_k | \bar{f}_k) = p(\eta_k | \bar{f}_k),
\]

so \( \eta_k \) is independent from all the past \( f_j \) contained in \( \phi_k \).

Then, we have that

\[
\begin{align*}
E[\eta_k | \{x_j, y'_j\}, x_j \in \mathcal{M}(\ell), \ell = 0, \ldots, k] &= E\left[ E[\eta_k | \phi_k, \bar{f}_k] | \{x_j, y'_j\}, x_j \in \mathcal{M}(\ell), \ell = 0, \ldots, k \right] \\
&= E\left[ E[\eta_k | \phi_k] | \{x_j, y'_j\}, x_j \in \mathcal{M}(\ell), \ell = 0, \ldots, k \right] \\
&= \Gamma K_s^{-1} \bar{f}_k = \Psi \bar{f}_k,
\end{align*}
\]

where the first equality holds because we are conditioning

on a larger \( \sigma \)-algebra; the second holds thanks to (18) and

the third comes from the definition (10) of \( \bar{f}_k \). Finally, for

the posterior variance we refer the reader [17], Appendix

A - Lemma 1 combined with the conditional independence

stated in Eq. (18).

The combination of Proposition 5 and Proposition 7 state

that the output of the Kalman filter captures all the necessary

information, contained in the measurements, to estimate

the entire process. Indeed, \( \bar{f}_k \) is a sufficient statistic to reconstruct

\( f_k \) over the entire domain \( \mathcal{X} \). Figure 3 shows a block

diagram of the overall estimation scheme.

IV. COMPUTATIONAL COMPLEXITY

Before presenting compelling numerical tests, we discuss

some computational aspects.

As already mentioned, the computational burden per iteration

for the standard nonparametric approach grows cubical

with the total number of collected measurements. Interest-

ingly, thanks to its recursive implementation, the proposed

Kalman’s computational complexity scales as \( O(rM_k^3 + rM_k) \),

where \( M_k \) is the number of collected measurements

per iteration and \( r \) is the order a single state space model

\( S_j \) in (9). The first term in the cost is due to the matrix

vector multiplication to compute the state update of Eq. (6c);

while the second to the computation of the Kalman gain

as in Eq. (6e). Additionally, it is worth noticing that to

compute \( \bar{n}_k \), the matrix \( \Psi' \in \mathbb{R}^{P \times M} \) is fixed thus, in a real-time

implementation, can be pre-computed offline. Therefore, to

reconstruct the entire process only a matrix-vector multipli-

cation, which costs \( O(MP) \), is needed. Thanks to this, our

Kalman regression procedure is characterized by an overall

cost complexity per iteration which scales as

\[ O(rM_k^3 + O(rM_k^3 + MP)) \]. \] (19)

Conversely, the nonparametric approach needs to invert a

matrix of increasing size at every iteration plus a matrix-

vector multiplication to compute the estimate for the overall

process, leading to a complexity of order

\[ O\left( \sum_{l=1}^{k} M_l \right)^3 + P \sum_{l=1}^{k} M_l \). \] (20)

Thus, in a real-time implementation, the computational cost

per iteration for the Kalman scales linearly with the model

complexity \( r \). Conversely the cost for the classical nonpa-

rametric implementation grows cubically with the total number

of collected measurements.

In the next section, we compare the proposed Kalman

regression scheme with a modified version of the classical

nonparametric implementation [7] based on a finite memory

approach, which we refer to as truncated nonparametric.

That is instead of storing in memory all the collected mea-

surements up to the current iteration, only the measurements

collected during the last \( q \) time instants are stored and

processed. This is commonly done in practice in order to keep

memory and computational requirements fixed. From (20),

it is easy to see that the truncated nonparametric scales as

\[ O\left( \sum_{l=k-q}^{k} M_l \right)^3 + P \sum_{l=k-q}^{k} M_l \). \] (21)

V. SIMULATIONS

In this section we present some simulations to show the

effectiveness of the proposed Kalman regression.

The hardware test-bed consists of a 2.7 GHz Intel Core i5

processor with 16GB RAM running MATLAB® 2015. To

plot the spatio-temporal evolution of the modeled function,

we work on a 1D space. More specifically, \( \mathcal{X} \) consists of a

line of length 100 [p.u.], uniformly sampled every 1 [p.u.]
Fig. 4: 3D representation of the optimal output obtained using the nonparametric estimation procedure.

Fig. 5: 3D representation of the output of the proposed Kalman regression procedure. State-space model of order \( r = 6 \).

\(|\mathcal{X}| = N = 100\). The sampling time is fixed and equal to 0.2 [s]. \( \mathcal{X}_{\text{meas}} \) consists of \( M = 80 \) randomly selected locations. Finally, \( \sigma = 1 \) [p.u.]. To test the effectiveness of the proposed approach even on processes whose kernel does not satisfy Assumption 3, the selected process is drawn by a spatio-temporal Gaussian kernel \( K \) with

\[
K_s(x, x') = e^{-0.2||x-x'||^2}, \quad h(\tau) = e^{-||\tau||^2/2}.
\]

As mentioned at the end of Example 6, to approximate the non rational PSD \( S(\omega) \) we compute \( \hat{S}_r(\omega) \) as the solution of a parametric non-linear weighted least-squares problem. More specifically, for a given order \( r \) we have that

\[
\hat{S}_r(\omega) = \arg\min_{\{a_i\}_{i=0}^r, \{b_i\}_{i=0}^{r-1}} \int_0^\infty \| S_i(w) - S(w) \|_{S(\omega)} dw.
\]

where \( \{a_i\}_{i=0}^r \) and \( \{b_i\}_{i=0}^{r-1} \) are the coefficients of the spectral factor \( W(\hat{\omega}) \) of \( S_r(\omega) \).

A. Estimation performance

First we compare Kalman with respect to the classical nonparametric method. At each time step \( k \), we collect noisy measurements of the form (11) from \( M_k \) randomly selected locations within \( \mathcal{X}_{\text{meas}} \), where \( M_k \) is randomly drawn from the uniform distribution over the set [60, 80].

Figures 4 and 5 show the estimates and the corresponding posterior variance obtained using the nonparametric method and Kalman, respectively. The nonparametric approach, at every iteration \( k \), uses all the measurements collected up to \( k \). Observe how the output are almost exactly the same. The difference is due to the fact that Kalman is built on \( \hat{S}_r(\omega) \), with \( r = 6 \), instead of the true \( S(\omega) \). Finally, notice that since the measurements locations change at every iteration, the posterior variance oscillates.

B. Computational performance

Here, we want to compare the proposed Kalman regression with the truncated nonparametric implementation described in Section IV. In the following we put \( M = N \) (\( \mathcal{X}_{\text{meas}} \equiv \mathcal{X} \)), so \( P = 0 \) (\( \mathcal{X}_{\text{pred}} = \emptyset \)). Moreover we assume \( M_k = M \), which is equivalent to collect measurements from all the locations. Thanks to this, the computational complexities per iteration (see Section IV) reduce to \( O(rM^2) \) for Kalman and to \( O(q'M^2) \) for the truncated nonparametric, respectively. Therefore, \( r \) and \( q' \) represent a measure for the complexity of the corresponding approach.

We compare the methods in term of CPU time per iteration
and in terms of estimation fit computed as
\[
\text{Fit} \ [%] = \left(1 - \frac{\|\hat{f}_r - f_{np}\|}{\|f_{np}\|}\right) \times 100, \quad (22)
\]
where \(\hat{f}_r\) denotes the estimate obtained either using Kalman or the truncated nonparametric; while \(f_{np}\) denotes the classical nonparametric estimate using all the available measurements. For the truncated nonparametric, Figure 6 shows the fit as function of the memory steps \(q\). For Kalman, the fit is plotted as function of the model order \(r\). It can be seen that, for the same level of complexity, Kalman in general achieves a better fit. We stress the fact that the performance in terms of fit for the truncated nonparametric highly depends on the ratio between the process and the measurements noise. Indeed, for high process noise, the information contained in the measurements collected during the last few iterations already contains all the necessary information to reconstruct the process. Thus, the fit curve would increase more rapidly. Conversely, Kalman is optimal hence it does not depend on the ratio. As final comparison, Figure 7 reports the fit versus the CPU time. The main fact to be highlighted is that to achieve a higher level of estimation accuracy, Kalman requires even less CPU time than the truncated nonparametric. It is important to underline that the CPU time deeply depend on \(M\) as well as on \(r\) and \(q\). Indeed, for greater values of \(M\) Kalman could overwhelm the truncated approach even more. Conversely, if smaller value for \(M\) are chosen, the truncated nonparametric might be more efficient.

VI. CONCLUSIONS AND FUTURE WORKS

In this work we focused of the efficient estimation of time-varying spatio-temporal processes by combining GP nonparametric regression and Kalman filtering. We developed a computationally efficient and exact procedure for estimating the underlying process on a finite number of measurement and prediction locations via a finite dimensional state-space representation. The results are based on a specific separability assumption on the generating kernel for the modeled process, and we showed that the major computational bottleneck is given only by the number of distinct measurement locations, and not by the prediction locations. Future avenues of research regard the relaxation of the assumption on the generating kernel and the extension of the proposed approach to prediction over any desired point in time and space under non-periodic sampling.

REFERENCES