Bayesian autoregression to optimize temporal Matérn kernel Gaussian process hyperparameters

Wouter M. Kouw   *Eindhoven University of Technology, Eindhoven, Netherlands*

Abstract

Gaussian processes are important models in the field of probabilistic numerics. We present a procedure for optimizing Matérn kernel temporal Gaussian processes with respect to the kernel covariance function's hyperparameters. It is based on casting the optimization problem as a recursive Bayesian estimation procedure for the parameters of an autoregressive model. We demonstrate that the proposed procedure outperforms maximizing the marginal likelihood as well as Hamiltonian Monte Carlo sampling, both in terms of runtime and ultimate root mean square error in Gaussian process regression.

1 Introduction

Temporal Gaussian processes are powerful probabilistic models to inter- and extrapolate time-series (Särkkä et al., 2013). They are used to forecast weather patterns such as rainfall and solar irradiation, to analyze brain recordings, and to monitor the structural health of windmills (Särkkä et al., 2013; Salcedo-Sanz et al., 2014; Rogers et al., 2020). Optimizing the hyperparameters that govern the shape of the kernel covariance function (e.g., length scales) is notoriously challenging: the objective function may have strong local optima and regions of divergence (Rasmussen and Williams, 2006; Svensson et al., 2015). These can cause gradient-based techniques, such as maximizing the marginal likelihood, to under-perform or even fail. Here we present a procedure that estimates the optimal hyperparameters quickly and robustly.

Probabilistic numerics quantifies and tracks uncertainty as it propagates through a mathematical model or computational procedure (Hennig et al., 2022). Traditional numerical methods are often limited to point estimates and are sensitive to noise or unexpected variations in measured signals. By incorporating sources of uncertainty explicitly, probabilistic numerical methods can adapt to perturbations and search spaces more effectively. The best example of this is probably Bayesian optimization, where instead of pursuing a gradient myopically, the optimizer infers the best possible next trial based on a balance between minimizing uncertainty and reaching the extremum (Garnett, 2023; Hennig et al., 2022). Viewing optimization as an inference problem is rapidly leading to advances in speed and robustness (Zhilinskas, 1975; Hennig and Kiefel, 2013; Mahsereci and Hennig, 2017; McLeod et al., 2018). We focus on optimizing hyperparameters of Gaussian processes. The most well-studied procedure for this is maximization of the marginal likelihood (Ying, 1991, 1993; Karvonen et al., 2019). This is computationally expensive as fitting a Gaussian process is already expensive in the number of data points, but now the fit procedure must be repeated for every trial of the hyperparameters (Sec. 2, Figure 1). Pre-conditioning techniques can

accelerate this procedure (Wenger et al., 2022), but we believe further progress can be made by inferring the hyperparameters instead of optimizing them. Markov Chain Monte Carlo methods have produced robust procedures (Murray et al., 2010; Svensson et al., 2015). We explore Bayesian filtering techniques with analytic solutions in order to improve in terms of speed. This is achieved for Matérn kernel temporal Gaussian process by casting them to linear stochastic differential equations, discretizing them to autoregressive processes and performing recursive Bayesian estimation on their parameters. Maximum a posteriori estimates of the autoregressive coefficients and noise precision parameter are reverted to kernel hyperparameters. To be specific, our contributions consists of:

- We formulate the temporal Gaussian process as an autoregressive difference equation using higher-order finite difference techniques.

- We propose a recursive Bayesian estimation procedure for kernel hyperparameters that relies on reverting a variable substitution.

The proposed procedure is validated on both simulated and real data, and compared against state-of-the-art methods.
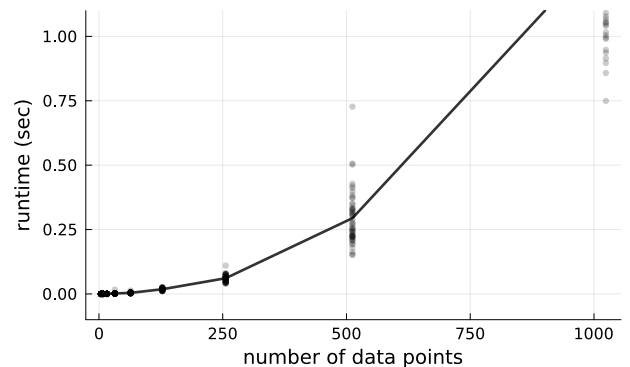

Figure 1: Simulation demonstrating the exponential increase in runtime for Gaussian process regression as the number of data points increases. The dots represent runtimes for simulated experiments and the solid line represents their average.

2 Problem Statement

Let $f : \mathbb{R}_+ \to \mathbb{R}$ be a latent scalar function over time and $y_k \in \mathbb{R}$ be an observation of the function at time $t_k \in \mathbb{R}_+$ for positive real numbers $\mathbb{R}_+$. Observations are assumed to be noise-free, which implies our likelihood has the form of a Dirac delta function:

$$p(y_k \mid f, t_k) = \delta(y_k - f(t_k)). \qquad (1)$$

Our prior distribution for the unknown function f is a zero-mean Gaussian process:

$$p\left(f \mid t; \psi\right) = \mathcal{GP}\left(f(t) \mid 0, \kappa_\psi(t, t')\right), \qquad (2)$$

with kernel covariance function $\kappa(\cdot)$ and hyperparameters ψ. We focus on the Matérn class of stationary[1] covariance functions

$$\kappa_\psi^\nu(t, t') = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{l}|t-t'|\right)^\nu B_\nu\left(\frac{\sqrt{2\nu}}{l}|t-t'|\right), \quad (3)$$

in which ν is a smoothness, σ a magnitude, and l a length scale hyperparameter (Rasmussen and Williams, 2006). $\Gamma(\cdot)$ is a gamma function and $B_\nu(\cdot)$ is a modified Bessel function of the second kind. We restrict the smoothness parameter to half-integers, i.e., $\nu = m - 1/2$ for positive integers $m = 1, 2, \ldots$. We will consider specific choices of m (mainly 1 and 2) and let $\psi = (\sigma, l)$.

Suppose we have a data set of N observations. Let $\mathbf{t} = \begin{bmatrix} t_1 & \ldots & t_N \end{bmatrix}^\mathsf{T}$ and $\mathbf{y} = \begin{bmatrix} y_1 & \ldots & y_N \end{bmatrix}^\mathsf{T}$. Most commonly, the kernel hyperparameters are obtained by maximizing the marginal likelihood:

$$\psi^* = \underset{\psi \in \Psi}{\arg\max} \ p(\mathbf{y} \mid \mathbf{t}; \psi), \qquad (4)$$

The marginal likelihood is formed by integrating the likelihood (Eq. 1) over the Gaussian process prior distribution (Eq. 2) (Rasmussen and Williams, 2006):

$$p(\mathbf{y} \mid \mathbf{t}; \psi) = \int p(\mathbf{y} \mid f, \mathbf{t}) \, p(f \mid \mathbf{t}; \psi) \, \mathrm{d}f \qquad (5)$$

$$= (2\pi)^{-1/2} |K_\psi|^{-1/2} \exp\left(-\frac{1}{2}\mathbf{y}^\mathsf{T} K_\psi^{-1} \mathbf{y}\right). \quad (6)$$

where K_ψ is the Gram matrix of the kernel covariance function applied to all pairwise combinations of t.

The problem is that the maximization procedure scales poorly with the amount of data: in gradient-based optimization, a new $N \times N$ Gram matrix K_ψ must be inverted for each iteration. It is thus $O(LN^3)$ where L is the maximum number of gradient-based update steps. Figure 1 visualizes the runtime of this hyperparameter optimization procedure[2] as a function of the number of data points. Note that the exponential nature.

---

[1]Stationary covariance functions are only a function of a difference in time $t - t'$ (Rasmussen and Williams, 2006).

[2]Experimental details: we sampled 50 realizations of a Matérn-1/2 kernel Gaussian process over time steps of size $\Delta = 0.1$ with random sinusoidal outputs (frequency sampled from a uniform distribution over the interval $(0, 2)$ and phase sampled uniformly from $(0, \pi)$) for $N = 2\hat{\ }\{2, \ldots, 10\}$. We timed the marginal likelihood maximization procedure for each realization.

3 Solution Procedure

Our hyperparameter optimization procedure consists of casting the Gaussian process to a linear stochastic differential equation, discretizing it in time using a higher-order finite difference method, inferring a posterior distribution over autoregressive model parameters and converting the maximum a posteriori estimates back to kernel hyperparameters.

3.1 Stochastic Differential Equations

Temporal Gaussian processes with stationary Matérn kernel covariance functions have an equivalent representation as scalar linear stochastic differential equations (Hartikainen and Särkkä, 2010; Särkkä et al., 2013). For the sake of completeness, we shall restate this result briefly here.

Consider an m-th order differential equation with coefficients a_n for $n = 0, \ldots, m - 1$ driven by a white noise process $w(t)$ with spectral density ς^2:

$$\frac{d^m f(t)}{dt^m} + \sum_{n=0}^{m-1} a_n \frac{d^n f(t)}{dt^n} = w(t). \qquad (7)$$

Taking the Fourier transform on both sides gives

$$(i\omega)^m F(\omega) + \sum_{n=0}^{m-1} a_n (i\omega)^n F(\omega) = W(\omega), \qquad (8)$$

for frequency ω. Note that the process in the frequency domain can be generated by a system with input $W(\omega)$ and a transfer function $H(\omega) = F(\omega)/W(\omega)$ that contains all the coefficients a_n (Glad and Ljung, 2018). The equivalent representation is based on the fact that the power spectral density of the process depends on the transfer function (Oppenheim et al., 1997):

$$S_F(\omega) = |H(\omega)|^2 S_W(\omega) = H(i\omega)\varsigma^2 H(-i\omega). \qquad (9)$$

Through the Wiener–Khinchin theorem, we know that the power spectral density of the Matérn kernel covariance function (Eq. 3) is

$$S(\omega) = \sigma^2 \frac{2\pi^{1/2}\Gamma(\nu+1/2)}{\Gamma(\nu)} \lambda^{2\nu}(\lambda^2 + \omega^2)^{-(\nu+1/2)}, \quad (10)$$

where $\lambda = \sqrt{2\nu}/l$ (Rasmussen and Williams, 2006). So the coefficients a_n and spectral density ς^2 can be identified by matching Eq. 10 and Eq. 9. The rational polynomial in ω^2 can be factorized into two rational polynomials in ω,

$$(\lambda^2+\omega^2)^{-(\nu+\frac{1}{2})} = \underbrace{(\lambda+i\omega)^{-(\nu+\frac{1}{2})}}_{H(i\omega)} \underbrace{(\lambda - i\omega)^{-(\nu+\frac{1}{2})}}_{H(-i\omega)}, \quad (11)$$

which are recognized as the two transfer functions in Eq. 9. The first function $H(i\omega) = (\lambda + i\omega)^{-(\nu+1/2)}$ has its poles in the upper plane and will generate a stable stochastic process (Hartikainen and Särkkä, 2010). The binomial theorem generates the coefficients of its characteristic polynomial, which form the coefficients a_n of

Eq. 8:

$$a_n = \binom{m}{n} \lambda^{m-n}. \tag{12}$$

The remaining terms become the spectral density of the white noise process, i.e.,

$$\varsigma^2 = \sigma^2 \lambda^{2\nu} 2\pi^{1/2} \Gamma(\nu + 1/2)/\Gamma(\nu). \tag{13}$$

Examples We will use two running examples throughout the paper to illustrate steps: the $\nu = 1/2$ (i.e., $m = 1$) and $\nu = 3/2$ (i.e., $m = 2$) cases. Firstly, for $\nu = 1/2$, we have:

$$H(i\omega) = (\lambda + i\omega)^{-1}. \tag{14}$$

This gives a first-order stochastic process,

$$(i\omega)F(\omega) + \lambda F(\omega) = W(\omega), \tag{15}$$

whose time domain form is

$$\frac{df(t)}{dt} + \lambda f(t) = w(t). \tag{16}$$

So we have only one coefficient, that is $a_0 = \lambda$. For the white noise, we have:

$$\varsigma^2 = \sigma^2 \lambda 2\pi^{1/2} \Gamma(1)/\Gamma(1/2) = 2\sigma^2 \lambda. \tag{17}$$

For the $\nu = 3/2$ case, the transfer function is

$$H(i\omega) = (\lambda + i\omega)^{-2} = (\lambda^2 + 2\lambda i\omega + (i\omega)^2)^{-1}. \tag{18}$$

This gives an order $m = 2$ stochastic process,

$$(i\omega)^2 F(\omega) + 2\lambda(i\omega)F(\omega) + \lambda^2 F(\omega) = W(\omega), \tag{19}$$

whose time domain form is

$$\frac{d^2 f(t)}{dt^2} + 2\lambda \frac{df(t)}{dt} + \lambda^2 f(t) = w(t). \tag{20}$$

So, in this case $a_1 = 2\lambda$ and $a_0 = \lambda^2$. For the white noise, we have:

$$\varsigma^2 = \sigma^2 \lambda^3 2\pi^{1/2} \Gamma(2)/\Gamma(3/2) = 4\sigma^2 \lambda^3. \tag{21}$$

3.2 Time Discretization

In seminal earlier work (Hartikainen and Särkkä, 2010; Särkkä and Hartikainen, 2012; Särkkä et al., 2013), Eq. 7 is represented as a Markov process with state vector $x(t) = \begin{bmatrix} f(t) & df(t)/dt & \dots & dt^{m-1}f(t)/dt^{m-1} \end{bmatrix}$,

$$\frac{dx(t)}{dt} = Fx(t) + Lw(t). \tag{22}$$

The transition matrix and noise matrices are:

$$F = \begin{bmatrix} 0 & 1 & \dots & 0 \\ \vdots & 0 & 1 & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ -a_0 & -a_1 & \dots & -a_{m-1} \end{bmatrix}, \; L = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ 1 \end{bmatrix}. \tag{23}$$

This is treated as a state-space model, and Bayesian filtering techniques are used to estimate the value of $f(t)$ at specific times. We will explore a different direction, using higher-order finite difference techniques to cast Eq. 7 as a discrete-time autoregressive process.

Consider an explicit forward finite difference approximation to an order-m derivative (Hamming, 2012):

$$\frac{d^m f(t)}{dt^m} \approx \frac{1}{\Delta^m} \sum_{n=0}^{m} (-1)^{m-n} \binom{m}{n} f(t + n\Delta), \tag{24}$$

where Δ is a fixed step size, i.e., $\Delta = t_k - t_{k\text{-}1}$ for all k. Let $f_k = f(t_k)$, $f_{k+1} = f(t_k + \Delta)$, $f_{k+2} = f(t_k + 2\Delta)$ and so on. Applying the forward finite difference approximation to each of the derivatives in Eq. 7 produces

$$\sum_{n=0}^{m} a_n \frac{d^n f(t)}{dt^n} \approx \sum_{n=0}^{m} \frac{a_n}{\Delta^n} \sum_{j=0}^{n} (-1)^{n-j} \binom{n}{j} f_{k+j}. \tag{25}$$

In a time step of size Δ, $w(t)$ contributes to the process with an increment drawn from a zero-mean Gaussian w_k with variance $\varsigma^2 \Delta$ (Särkkä and Solin, 2019);

$$\int_0^{\Delta} w(s)ds = w_k \sim \mathcal{N}\left(0, \varsigma^2 \Delta\right). \tag{26}$$

We have thus discretized the stochastic differential equation in Eq. 7 to:

$$\sum_{n=0}^{m} \frac{a_n}{\Delta^n} \sum_{j=0}^{n} (-1)^{n-j} \binom{n}{j} f_{k+j} = w_k. \tag{27}$$

We would like to isolate the most forward function step on one side of the equation;

$$\sum_{n=0}^{m} \frac{a_n}{\Delta^n} \sum_{j=0}^{n} (-1)^{n-j} \binom{n}{j} f_{k+j} = \frac{a_m}{\Delta^m} f_{k+m} + \tag{28}$$

$$\sum_{n=0}^{m-1} \frac{a_m}{\Delta^m} (-1)^{m-n} \binom{m}{n} f_{k+n} + \sum_{j=0}^{n} \frac{a_j}{\Delta^j} (-1)^{n-j} \binom{n}{j} f_{k+j}.$$

Note that, due to the nature of the transfer function, the coefficient belonging to the highest-order derivative, a_m, will always be 1 (Eq. 7). That means the leading coefficient is $1/\Delta^m$. Multiplying by Δ^m and using Eq. 28, Eq. 27 becomes:

$$f_{k+m} = -\sum_{n=0}^{m-1} (-1)^{m-n} \binom{m}{n} f_{k+n} \tag{29}$$

$$- \sum_{j=0}^{n} a_j \Delta^{m-j} (-1)^{n-j} \binom{n}{j} f_{k+j} + \Delta^m w_k.$$

Under this time discretization scheme, the stochastic differential equation representation of the temporal Gaussian process becomes an autoregressive process (Sec. 3.3).

Examples For the $m = 1$ case, the forward finite difference is familiar

$$\frac{df(t)}{dt} \approx \frac{f_{k+1} - f_k}{\Delta}. \tag{30}$$

3

Under this scheme, the weighted sum of derivatives is approximately:

$$a_1 \frac{df(t)}{dt} + a_0 f(t) \approx a_1 \frac{f_{k+1} - f_k}{\Delta} + a_0 f_k \tag{31}$$

$$= \frac{a_1}{\Delta} f_{k+1} + (a_0 - \frac{a_1}{\Delta}) f_k . \tag{32}$$

Equating this to the discretized noise and moving terms around yields:

$$\frac{a_1}{\Delta} f_{k+1} + (a_0 - \frac{a_1}{\Delta}) f_k = w_k \tag{33}$$

$$f_{k+1} = (1 - a_0 \Delta) f_k + \Delta w_k . \tag{34}$$

For the $m = 2$ case, the forward finite difference is:

$$\frac{d^2 f(t)}{dt^2} \approx \frac{f_{k+2} - 2f_{k+1} + f_k}{\Delta^2} . \tag{35}$$

Incorporating this into the weighted sum of derivatives yields:

$$
a_2 \frac{d^2 f(t)}{dt^2} + a_1 \frac{df(t)}{dt} + a_0 f(t)
$$
$$
\approx \frac{a_2}{\Delta^2} \left(\binom{2}{0} f_k - \binom{2}{1} f_{k+1} + \binom{2}{2} f_{k+2} \right)
$$
$$
+ \frac{a_1}{\Delta^1} \left(\binom{1}{0} f_k - \binom{1}{1} f_{k+1} + \binom{1}{2} f_{k+2} \right)
$$
$$
+ \frac{a_0}{\Delta^0} \left(\binom{0}{0} f_k - \binom{0}{1} f_{k+1} + \binom{0}{2} f_{k+2} \right) \tag{36}
$$
$$
= \frac{a_2}{\Delta^2} (f_k - 2f_{k+1} + f_{k+2}) + \frac{a_1}{\Delta} (f_k - f_{k+1}) + a_0 f_k \tag{37}
$$
$$
= \frac{a_2}{\Delta^2} f_{k+2} - (\frac{2a_2}{\Delta^2} + \frac{a_1}{\Delta}) f_{k+1} + (\frac{a_2}{\Delta^2} + \frac{a_1}{\Delta} + a_0) f_k . \tag{38}
$$

Equating this to the discretized noise, setting $a_2 = 1$, and moving terms around yields:

$$\frac{a_2}{\Delta^2} f_{k+2} - (\frac{2a_2}{\Delta^2} + \frac{a_1}{\Delta}) f_{k+1} + (\frac{a_2}{\Delta^2} + \frac{a_1}{\Delta} + a_0) f_k = w_k \tag{39}$$

$$f_{k+2} = (2 + a_1 \Delta) f_{k+1} - (1 + a_1 \Delta + a_0 \Delta^2) f_k + \Delta^2 w_k . \tag{40}$$

3.3 Probabilistic Model Specification

In order to specify a probabilistic model and infer the kernel hyperparameters λ, σ, we perform a number of variable substitutions. First, we combine the terms involving f on the right-hand side of Eq. 29 into:

$$
-\sum_{n=0}^{m-1} (-1)^{m-n} \binom{m}{n} f_{k+n} - \sum_{j=0}^{n} a_j \Delta^{m-j} (-1)^{n-j} \binom{n}{j} f_{k+j}
$$
$$
= \sum_{n=0}^{m-1} \theta_n f_{k+n} , \tag{41}
$$

where the substitutions are

$$\theta_n = (-1)^{m-n+1} \binom{m}{n} - \sum_{j=0}^{m-1} a_n \Delta^{m-n} (-1)^{j-n} \binom{j}{n} . \tag{42}$$

Plugging Eq. 41 into Eq. 29 reveals the autoregressive nature of the higher-order finite difference approximation of the stochastic differential equation:

$$f_{k+m} = \sum_{n=0}^{m-1} \theta_n f_{k+n} + \Delta^m w_k . \tag{43}$$

We shall therefore refer to θ_n as the autoregressive coefficients.

To substitute for the scale of the Gaussian noise contribution, we first incorporate the Δ^m scaling factor into the variance of w_k. Let $\bar{w}_k = \Delta^m w_k$ such that $\bar{w}_k \sim \mathcal{N}(0, \Delta^{2m+1} \varsigma^2)$. Then we define a noise precision parameter τ as:

$$\tau = 1/(\Delta^{2m+1} \varsigma^2) \tag{44}$$

$$= 1/(\Delta^{2m+1} \sigma^2 \lambda^{2\nu} 2\pi^{1/2} \Gamma(\nu + 1/2)/\Gamma(\nu)) . \tag{45}$$

The kernel hyperparameters are part of the autoregressive coefficients and noise precision parameters. We can now specify a standard probabilistic autoregressive model, pose suitable prior distributions and infer posterior distributions. Reverting the variable substitutions using the maximum a posteriori estimates will give us estimates of the optimal kernel hyperparameters ψ^*.

Examples For $\nu = 1/2$ (i.e., $m = 1$), we combine Eqs. 34 and 16 to find:

$$\theta_0 = 1 - a_0 \Delta = 1 - \lambda \Delta \tag{46}$$

$$\tau = (\varsigma^2 \Delta^3)^{-1} = 1/(2\sigma^2 \lambda \Delta^3) . \tag{47}$$

In the $\nu = 3/2$ ($m = 2$) case, the substitutions are:

$$\theta_0 = 2 + a_1 \Delta = 2 + 2\lambda \Delta \tag{48}$$

$$\theta_1 = -(1 + a_1 \Delta + a_0 \Delta^2) = -1 - 2\lambda \Delta - \lambda^2 \Delta^2 \tag{49}$$

$$\tau = (\varsigma^2 \Delta^5)^{-1} = 1/(4\sigma^2 \lambda^3 \Delta^5) . \tag{50}$$

Likelihood Function Given the variable substitution above, we can define a likelihood function. Consider a buffer of previous observations[3]

$$\bar{y}_k = \begin{bmatrix} y_k & y_{k-1} & \cdots & y_{k-m-1} \end{bmatrix} . \tag{51}$$

Then we may write

$$p(y_{k+1} \mid \bar{y}_k, \theta, \tau) = \mathcal{N}(y_{k+1} \mid \theta^\mathsf{T} \bar{y}_k, \tau^{-1}) , \tag{52}$$

for unknown coefficients $\theta = \begin{bmatrix} \theta_0 & \cdots & \theta_{m-1} \end{bmatrix}^\mathsf{T}$ and noise precision parameter τ.

Prior Distribution We specify a joint prior distribution on the autoregression coefficient θ and the noise precision τ. Specifically, a compound multivariate Gaussian and univariate Gamma distribution:

$$p(\theta, \tau) = \mathcal{N}\mathcal{G}(\theta, \tau \mid \mu, \Lambda, \alpha, \beta) \tag{53}$$

$$= \mathcal{N}(\theta \mid \mu, (\tau\Lambda)^{-1}) \mathcal{G}(\tau \mid \alpha, \beta) , \tag{54}$$

---

[3]For $k < m - 1$, note that the buffer is initialized with zeros and is filled with observations as they arrive.

with mean μ, precision matrix Λ, shape parameter α and rate parameter β. Their density functions are:

$$\mathcal{N}(\theta|\mu, (\tau\Lambda)^{-1}) = \frac{|\tau\Lambda|^{1/2}}{(2\pi)^{m/2}} \exp(-\frac{\tau}{2}(\theta-\mu)^{\mathsf{T}}\Lambda(\theta-\mu)) \quad (55)$$

$$\mathcal{G}(\tau \mid \alpha, \beta) = \frac{\beta^\alpha}{\Gamma(\alpha)}\tau^{\alpha-1}\exp(-\beta\tau). \quad (56)$$

This choice of priors is conjugate to the autoregressive likelihood (Shaarawy and Ali, 2008; Kouw, 2023).

3.4 Bayesian Inference Procedure

We will adopt a Bayesian filtering procedure to obtain a posterior distribution over parameters (Särkkä and Svensson, 2023):

$$p(\theta, \tau \mid y_{1:k+1}) = \frac{p(y_{k+1} \mid \bar{y}_k, \theta, \tau)}{p(y_{k+1} \mid y_{1:k})}p(\theta, \tau \mid y_{1:k}). \quad (57)$$

As the prior distribution is conjugate to the autoregressive likelihood, it yields an exact posterior distribution

$$p(\theta, \tau|y_{1:k+1}) = \mathcal{N}\mathcal{G}\big(\theta, \tau|\mu_{k+1}, \Lambda_{k+1}, \alpha_{k+1}, \beta_{k+1}\big), \quad (58)$$

with parameters:

$$\mu_{k+1} = \big(\bar{y}_k\bar{y}_k^{\mathsf{T}} + \Lambda_k\big)^{-1}\big(\bar{y}_k y_{k+1} + \Lambda_k\mu_k\big) \quad (59)$$

$$\Lambda_{k+1} = \bar{y}_k\bar{y}_k^{\mathsf{T}} + \Lambda_k \quad (60)$$

$$\alpha_{k+1} = \alpha_k + \frac{1}{2} \quad (61)$$

$$\beta_{k+1} = \beta_k + \frac{1}{2}\big(y_{k+1}^2 - \mu_{k+1}^{\mathsf{T}}\Lambda_{k+1}\mu_{k+1} + \mu_k^{\mathsf{T}}\Lambda_k\mu_k\big). \quad (62)$$

This is a recursive solution, which means that we start with a set of initial points $(\mu_0, \Lambda_0, \alpha_0, \beta_0)$ and that each update aims to produce a better estimate. Whether there is a strict improvement is an open question. Note that these parameters can be reverted at any time to produce an estimate of the kernel hyperparameters. In that sense, we have a running solution, a valuable property when the computational budget is limited.

3.5 Reverting the Substitution

The variable substitutions in Equations 42 and 44 are a system of polynomial equations. Reverting the substitution means finding values for λ and σ for which every polynomial equation is equal to the substituted variables θ, τ. In other words, we must solve the system of polynomial equations (Sturmfels, 2002). For $m \geq 2$, the system is overdetermined, i.e., there are more equations than unknowns, and it is non-homogeneous (i.e., it has non-zero constants), which means it will probably have no real solutions (Chen and Li, 2015). If there are none, the substituted variables cannot be reverted exactly. However, we can still look for approximate reversions that may prove to perform well. Here we opt for a nonlinear least-squares approach. We construct an objective function that consists of the sum of squared deviations of every polynomial from the MAP estimates

of the autoregressive parameters, μ_n and $(\alpha - 1)/\beta$. These are:

$$g_n(\psi) = \big(\mu_n - \theta_n\big)^2, \quad g_m(\psi) = \Big(\frac{\alpha-1}{\beta} - \tau\Big)^2, \quad (63)$$

for $n = 0, \ldots, m-1$, θ_n as defined in Eq. 42 and τ in Eq. 44. The objective is minimized with respect to $\psi = (\lambda, \sigma)$ with the constraint that both should be positive real-valued numbers $\Psi \in \mathbb{R}_+^2$;

$$\psi^* = \arg\min_{\psi \in \Psi} \sum_{i=0}^{m} g_i(\psi). \quad (64)$$

Note that this optimization problem scales with the degree of the Matérn kernel m, and not the number of training data points N (as is the case in the original marginal likelihood maximization problem of Eq. 4).

Examples For the $\nu = 1/2$ ($m = 1$) case, the system is linear and can be reverted exactly,

$$\mu = 1 - \lambda\Delta \quad \implies \quad \lambda = \frac{1-\mu}{\Delta}. \quad (65)$$

This result is plugged into the noise substitution,

$$\frac{\alpha-1}{\beta} = 1/\big(2\sigma^2\lambda\Delta^3\big) = 1/\big(2\sigma^2(1-\mu)\Delta^2\big), \quad (66)$$

yielding the reversion

$$\sigma^2 = \frac{\beta}{2(\alpha-1)(1-\mu)\Delta^2}. \quad (67)$$

Note that this reversion can be done at any time, i.e., for any $k = 1, \ldots, N$.

The $\nu = 3/2$ ($m = 2$) case is a system of polynomial equations. The application of homotopy continuation techniques reveals that it has no real solutions (Chen and Li, 2015). Plugging in the specific polynomials from Equations 48-50, produces the following squared error terms for the nonlinear least-squares objective:

$$g_0(\psi) = \Big(\mu_1 - (2 + 2\lambda\Delta)\Big)^2 \quad (68)$$

$$g_1(\psi) = \Big(\mu_2 - (-1 - 2\lambda\Delta - \lambda^2\Delta^2)\Big)^2 \quad (69)$$

$$g_2(\psi) = \Big(\frac{\alpha-1}{\beta} - 1/(4\sigma^2\lambda^3\Delta^5)\Big)^2. \quad (70)$$

Minimizing the sum of these functions with respect to ψ produces approximate reversions of the autoregressive parameters to the kernel hyperparameters.

4 Experiments

We perform simulation experiments with randomly generated temporal Gaussian processes[4]. The proposed probabilistic solution (referred to as BAR) is compared to the two most commonly used procedures: maximizing the marginal likelihood with respect to the kernel hyperparameters (Eq. 4; referred to as MML) and

---

[4]For code and additional experimental details, see github.com/biaslab/ProbNum2025-BARGPhparams.

Hamiltonian Monte Carlo sampling (referred to as HMC). For this, we employ the state-of-the-art toolbox GaussianProcesses.jl (Fairbrother et al., 2022). The hyperparameters are expressed on a log-scale to allow for unconstrained optimization. The mean of the prior distribution is fixed to 0's, the initial points for MML are 1.0, the priors for HMC are Exponential distributions, and the maximum number of iterations is limited to 1000. To solve the optimization problem in Eq. 64, we use Optim.jl with a log-barrier function to enact the positivity constraints and L-BFGS as optimizer (Mogensen and Riseth, 2018).

Evaluation To evaluate the found hyperparameters, we plug them into the standard predictive distribution formulation of a Matérn-kernel Gaussian process regression model. The degree of the Matérn kernel for testing will match the one for training. We express performance as the root mean square error (RMSE) between the mean vector of the predictive distribution and the observed data points.

4.1 Simulations

Matérn-1/2 Kernel We generate 50 realizations of an isotropic Matérn-1/2 kernel Gaussian process as training set, and another 50 as test set. The system λ was sampled from a Beta distribution with a shape parameter of 10 and a rate parameter of 4 and the system noise precision parameter τ was sampled from a Gamma distribution with shape parameter 10 and rate parameter 1. The time step size was $\Delta = 0.1$ and the outputs were generated from sinusoid with frequency sampled from a uniform distribution over the interval $(0, 2)$ and phase sampled uniformly from $(0, \pi)$. For the prior parameters of the autoregressive solution, we used $\mu_0 = 0.0$, $\Lambda_0 = 10^{-3}$, $\alpha_0 = 2$ and $\beta_0 = 0.1$. These should be considered as weakly informative, i.e., not flat but not concentrated on the system parameters either.

In the first experiment, we time both procedures as a function of the number of data points N. Figure 2 (top) demonstrates that BAR is much faster and scales better than both MML and HMC. In the second experiment, the training and test data set sizes are fixed to $N = 100$ and we compare the ultimate root mean square error of the estimated hyperparameters as a function of the procedure's runtime. Figure 2 (bottom) shows that BAR dominates not just in runtime, but also in terms of RMSE.

Matérn-3/2 Kernel We generate 10 realizations of an isotropic Mat'ern-3/2 kernel Gaussian process as training set, and another 10 as test set. As before, the system λ was sampled from a Beta distribution with a shape parameter of 10 and a rate parameter of 4 and the system noise precision parameter τ was sampled from a Gamma distribution with shape parameter 10 and rate parameter 1. The time step size was still $\Delta = 0.1$ and the outputs were also generated from sinusoid with frequency sampled from a uniform distribution over the interval
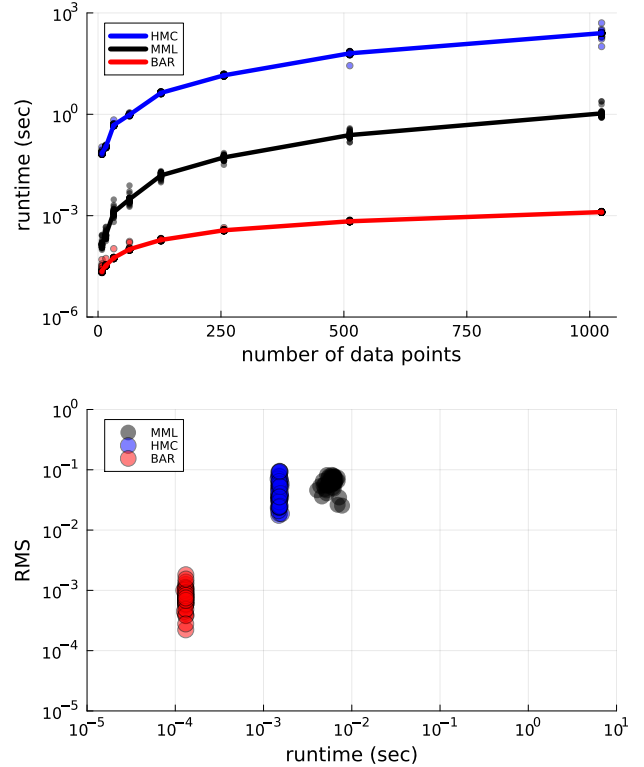

Figure 2: Matérn-1/2. Comparison of maximizing the marginal likelihood (MML), Hamiltonian Monte Carlo (HMC) versus Bayesian autoregression (BAR) for runtime (in seconds) as a function of the number of training data points (top) and root mean square error as a function of runtime (bottom; $N = 100$).

$(0, 2)$ and phase sampled uniformly from $(0, \pi)$. Figure 3 (top) again shows runtime comparisons, demonstrating that, although BAR is slower than MML for $N \leq 16$, it scales much better than MML and HMC (factor 10^3 improvement over MML and 10^6 improvement over HMC).

Figure 2 (bottom) shows that BAR still outperforms MML and HMC in terms of RMSE by a wide margin.

4.2 Real data

Room Occupancy Various measurements were taken to estimate the number of occupants in a room, using non-intrusive devices that sense temperature, (infrared) light, CO2 and sound (Singh et al., 2018). We subsampled the data to a measurement every 2 minutes, leaving 2531 data points. We performed 100 experiments where we sampled uniformly at random one of the 17 features and a starting point between 1 and 2531. We then created the training set from the first $N = 100$ points after the start, and the test set from the 100 after that. Figure 4 (top) shows the RMSE versus runtime comparison of the three methods. BAR dominates the other two in terms of runtime. On average, it performs better than MML and HMC but there are data sets for which it performs worse than MML or HMC.

Monitoring Hydraulics In this data set, a hydraulic test rig's condition was monitored over time using a range of sensors (Helwig et al., 2015). We focused on
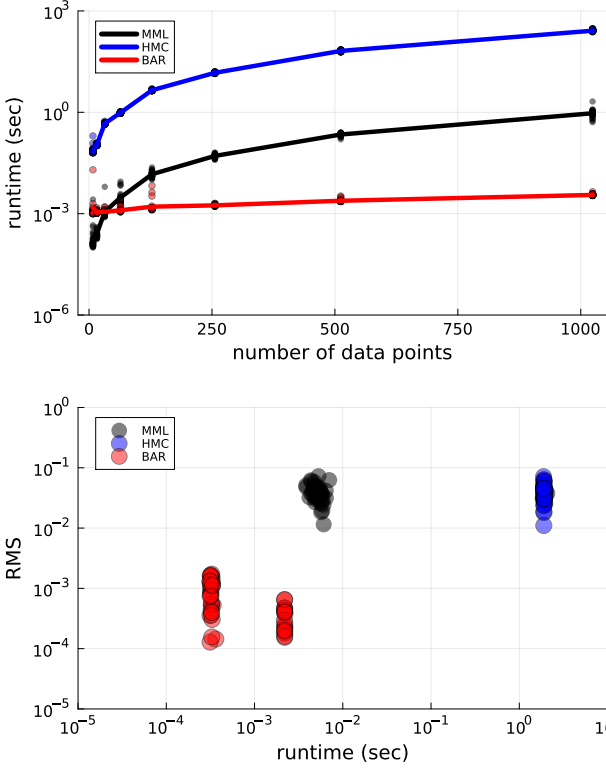
Figure 3: Matérn-3/2. Comparison of maximizing the marginal likelihood (MML), Hamiltonian Monte Carlo (HMC) and Bayesian autoregression (BAR) for runtime (in seconds) as a function of the number of training data points (top) and root mean square error as a function of runtime (bottom; $N = 100$).

Figure 4: Comparison of BAR, HMC and MML for a Matérn-1/2 kernel Gaussian process in terms RMSE versus runtime. (Top) Room occupancy data set. (Bottom) Condition monitoring of hydraulic system data.

temperature, vibration and cooling power, all read out once per second. The measurements are taken over multiple cycles that last 60 seconds ($N = 60$ for a single cycle). We sampled uniformly at random 1 of the 3 sensors and 2 of the 2205 cycles (1 for training and 1 for testing). Figure 4 plots the RMSE versus runtime for all three methods. BAR dominates in terms of runtime and outperforms the other methods in terms of RMSE in nearly all cases.

5 Discussion

For the Matérn-1/2 kernel, we effectively turned an optimization problem into a Bayesian filtering one. For higher order Matérn kernels, we have turned an optimization problem that scales on the order of the number of data points into one that scales on the order of the kernel. Although this will produce solutions faster, it remains to be analyzed how the approximate reversion relates to the optimal hyperparameters. The proposed method is specific to Matérn kernel Gaussian processes and will not easily generalize to other kernel covariance functions. It may be possible to utilize it for the squared exponential kernel; it can be expressed as a stochastic differential equation but it is of infinite-order and would require a finite-order approximation (Hartikainen and Särkkä, 2010; Särkkä et al., 2013). In terms of experimental validation, it should be noted that there are many more methods for obtaining kernel hyperparameters for Gaussian processes (e.g., Bayesian optimization, sequential Monte Carlo), and more work is needed to
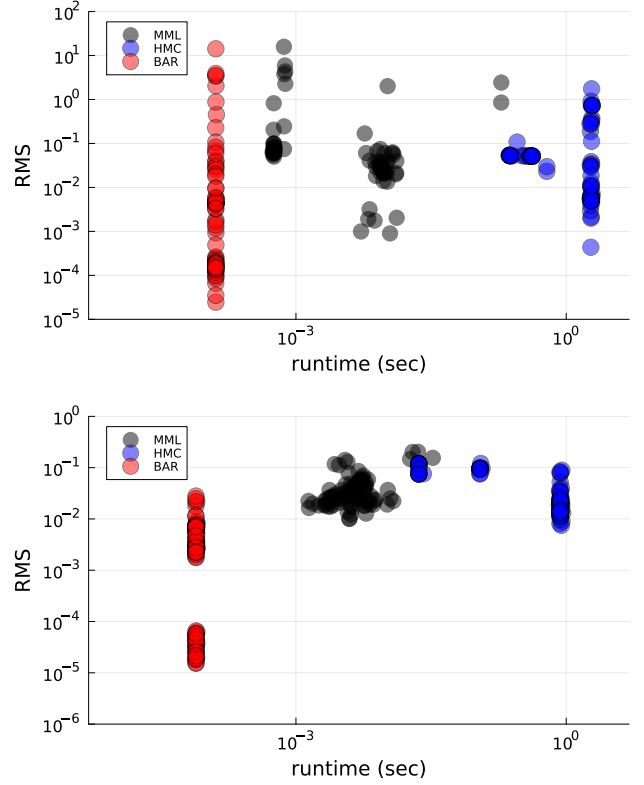
fully characterize the position of the proposed approach in that landscape (Rasmussen and Williams, 2006; Murray et al., 2010; Svensson et al., 2015).

There are several points of improvement for the proposed solution. Firstly, we have yet to incorporate noisy observations. This could be achieved by incorporating the observation noise into the variable substitution in Section 3.3. However, if the observation noise precision parameter is unknown, then this would complicate the variable substitution reversion. Secondly, it is unclear what the effect of the approximation errors (high-order finite differences, numerical solution to system of polynomial equations) is on the Bayesian parameter estimates and on the subsequent kernel hyperparameter estimates; does it lead to bias? Thirdly, it remains to be studied whether a Bayesian autoregressive inference procedure produces consistent estimates of the kernel hyperparameters. The maximal marginal likelihood estimator is strongly consistent and asymptotically normal (Ying, 1993), but the approximation errors inherent to our proposed procedure prevent a similar analysis.

6 Conclusion

We presented a Bayesian inference procedure for obtaining the hyperparameters of Matérn kernel temporal Gaussian processes, models important for time series analysis and forecasting. The solution was based on casting the stochastic differential equation representation of the temporal Gaussian process as an autoregres-

sive difference equation and then performing recursive Bayesian estimation of the autoregressive coefficients and noise precision parameter. The proposed procedure was faster than state-of-the-art methods and produced estimates that led to improved predictive performance in subsequent Gaussian process regression.

References

T. Chen and T.-Y. Li. Homotopy continuation method for solving systems of nonlinear and polynomial equations. *Communications in Information and Systems*, 15(2):119–307, 2015.

J. Fairbrother, C. Nemeth, M. Rischard, J. Brea, and T. Pinder. GaussianProcesses.jl: A nonparametric Bayes package for the Julia language. *Journal of Statistical Software*, 102:1–36, 2022.

R. Garnett. *Bayesian optimization*. Cambridge University Press, 2023.

T. Glad and L. Ljung. *Control Theory: Multivariable and Nonlinear Methods*. CRC press, 2018.

R. W. Hamming. *Introduction to applied numerical analysis*. Courier Corporation, 2012.

J. Hartikainen and S. Särkkä. Kalman filtering and smoothing solutions to temporal Gaussian process regression models. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 379–384, 2010.

N. Helwig, E. Pignanelli, and A. Schütze. Condition monitoring of a complex hydraulic system using multivariate statistics. In *IEEE International Instrumentation and Measurement Technology Conference*, pages 210–215, 2015.

P. Hennig and M. Kiefel. Quasi-newton methods: A new direction. *The Journal of Machine Learning Research*, 14(1):843–865, 2013.

P. Hennig, M. A. Osborne, and H. P. Kersting. *Probabilistic Numerics: Computation as Machine Learning*. Cambridge University Press, 2022.

T. Karvonen, F. Tronarp, and S. Särkkä. Asymptotics of maximum likelihood parameter estimates for Gaussian processes: The Ornstein–Uhlenbeck prior. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6, 2019.

W. M. Kouw. Information-seeking polynomial NARX model-predictive control through expected free energy minimization. *IEEE Control Systems Letters*, 2023.

M. Mahsereci and P. Hennig. Probabilistic line searches for stochastic optimization. *Journal of Machine Learning Research*, 18(119):1–59, 2017.

M. McLeod, S. Roberts, and M. A. Osborne. Optimization, fast and slow: optimally switching between local and Bayesian optimization. In *International Conference on Machine Learning*, pages 3443–3452. PMLR, 2018.

P. K. Mogensen and A. N. Riseth. Optim: A mathematical optimization package for Julia. *Journal of Open Source Software*, 3(24):615, 2018.

I. Murray, R. Adams, and D. MacKay. Elliptical slice sampling. In *International Conference on Artificial Intelligence and Statistics*, pages 541–548. PMLR, 2010.

A. V. Oppenheim, A. S. Willsky, and S. H. Nawab. *Signals & systems*. Pearson Educación, 1997.

C. E. Rasmussen and C. K. Williams. *Gaussian processes for machine learning*, volume 1. Springer, 2006.

T. Rogers, K. Worden, and E. Cross. On the application of Gaussian process latent force models for joint input-state-parameter estimation: With a view to Bayesian operational identification. *Mechanical Systems and Signal Processing*, 140:106580, 2020.

S. Salcedo-Sanz, C. Casanova-Mateo, J. Muñoz-Marí, and G. Camps-Valls. Prediction of daily global solar irradiation using temporal Gaussian processes. *IEEE Geoscience and Remote Sensing Letters*, 11 (11):1936–1940, 2014.

S. Särkkä and J. Hartikainen. Infinite-dimensional Kalman filtering approach to spatio-temporal Gaussian process regression. In *Artificial Intelligence and Statistics*, pages 993–1001. PMLR, 2012.

S. Särkkä and A. Solin. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

S. Särkkä and L. Svensson. *Bayesian filtering and smoothing*, volume 17. Cambridge University Press, 2023.

S. Särkkä, A. Solin, and J. Hartikainen. Spatiotemporal learning via infinite-dimensional Bayesian filtering and smoothing: A look at Gaussian process regression through Kalman filtering. *IEEE Signal Processing Magazine*, 30(4):51–61, 2013.

S. M. Shaarawy and S. S. Ali. Bayesian identification of multivariate autoregressive processes. *Communications in Statistics—Theory and Methods*, 37(5):791–802, 2008.

A. P. Singh, V. Jain, S. Chaudhari, F. A. Kraemer, S. Werner, and V. Garg. Machine learning-based occupancy estimation using multivariate sensor nodes. In *IEEE Globecom Workshops*, pages 1–6, 2018.

B. Sturmfels. *Solving systems of polynomial equations*. Number 97. American Mathematical Society, 2002.

A. Svensson, J. Dahlin, and T. B. Schön. Marginalizing Gaussian process hyperparameters using sequential Monte Carlo. In *IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pages 477–480, 2015.

J. Wenger, G. Pleiss, P. Hennig, J. Cunningham, and J. Gardner. Preconditioning for scalable Gaussian process hyperparameter optimization. In *International Conference on Machine Learning*, pages 23751–23780. PMLR, 2022.

Z. Ying. Asymptotic properties of a maximum likelihood estimator with data from a Gaussian process. *Journal of Multivariate Analysis*, 36(2):280–296, 1991.

Z. Ying. Maximum likelihood estimation of parameters under a spatial sampling scheme. *The Annals of Statistics*, pages 1567–1590, 1993.

A. Zhilinskas. Single-step Bayesian search method for an extremum of functions of a single variable. *Cybernetics*, 11(1):160–166, 1975.