Computers and Chemical Engineering 178 (2023) 108320

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Physics-based Penalization for Hyperparameter Estimation in Gaussian Process Regression

Jinhyeun Kim [a], Christopher Luettgen [a,c], Kamran Paynabar [b,*], Fani Boukouvala [a,*]

[a] School of Chemical & Biomolecular Engineering, Georgia Institute of Technology
[b] H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology
[c] Renewable Bioproducts Institute, Georgia Institute of Technology

ARTICLE INFO

Keywords:
Gaussian Process Regression
Maximum Likelihood Estimation
Physics-informed Machine Learning

ABSTRACT

In Gaussian Process Regression (GPR), hyperparameters are often estimated by maximizing the marginal likelihood function. However, this data-dominant hyperparameter estimation process can lead to poor extrapolation performance and often violates known physics, especially in sparse data scenarios. In this paper, we embed physics-based knowledge through penalization of the marginal likelihood objective function and study the effect of this new objective on consistency of optimal hyperparameters and quality of GPR fit. Three case studies are presented, where physics-based knowledge is available in the form of linear Partial Differential Equations (PDEs), while initial or boundary conditions are not known so direct forward simulation of the model is challenging. The results reveal that the new hyperparameter set obtained from the augmented marginal likelihood function can improve the prediction performance of GPR, reduce the violation of the underlying physics, and mitigate overfitting problems.

1. Introduction

Gaussian Process Regression (GPR) is a powerful interpolation technique to construct a predictive model with a finite set of observation points available in a system (Rasmussen, 2003). Due to its flexibility to approximate arbitrary continuous functions, and to provide an accompanying measure of the uncertainty of prediction, GPR is extensively used in various areas such as time-series analysis (Roberts et al., 2013), Bayesian optimization (Gustafsson et al., 2020; J. Kim & Choi, 2019; M. Kim et al., 2022; Pahari et al., 2021; Paulson & Lu, 2022), model calibration (Bradley et al., 2022; Dai et al., 2022; Eugene et al., 2020; Kennedy & O'Hagan, 2001), experimental design (W. Chen et al., 2008; Olofsson et al., 2018; Olofsson et al., 2021; Petsagkourakis & Galvanin, 2021), feasibility analysis (Boukouvala & Ierapetritou, 2012), prediction (Grbić et al., 2013; Kong et al., 2018), process modeling (Ahmad & Karimi, 2021; Alves et al., 2022), optimization (Davis & Ierapetritou, 2007; Quirante et al., 2015; Schweidtmann et al., 2021; Wiebe et al., 2022) and control (Berkenkamp & Schoellig, 2015; Bonzanini et al., 2021; Jain et al., 2018; Kocijan et al., 2003). Its Bayesian interpretation based on simple parameterization makes GPR a competing candidate among many other Machine Learning (ML) models such as Support Vector Regression (SVR) and Neural Networks (NN). In recent years, GPR has been actively improved and extended under different contexts, including efficient optimization (Cao et al., 2013), improved learning of the data (Damianou & Lawrence, 2013; Mattos & Barreto, 2019), generalizable kernels (Damianou & Lawrence, 2013; Duvenaud et al., 2011; Wilson & Adams, 2013), and integration with physics-based laws (Constantinescu & Anitescu, 2013; Jidling et al., 2017b; Nevin et al., 2021; Paulson & Lu, 2022; Raissi & Karniadakis, 2018; Raissi et al., 2017; X. Yang et al., 2018).

GPR is a nonparametric kernel-based method fully characterized by the mean and kernel functions (Rasmussen, 2003). Therefore, the choice of functional forms of the mean and kernel functions and their hyperparameters play a critical role in GPR performance (Fischer et al., 2016; Rasmussen, 2003). In practice, modelers often use a zero-prior mean and choose a kernel depending on the belief (e.g., smoothness, periodicity) of the system. In this case, hyperparameters of the kernel become a primary interest to modelers since model performance is highly determined by its hyperparameters.

Hyperparameters in GPR are estimated via two popular methods: (a) Maximum (marginal) Likelihood Estimation (MLE) (Blum & Riedmiller, 2013) and (b) Markov Chain Monte Carlo (MCMC) sampling (Titsias et al., 2008). MLE is a point estimation method that produces a single set of hyperparameters by maximizing the (marginal) likelihood function.

[*] Corresponding authors.
E-mail addresses: kamran.paynabar@isye.gatech.edu (K. Paynabar), fani.boukouvala@chbe.gatech.edu (F. Boukouvala).

https://doi.org/10.1016/j.compchemeng.2023.108320
Received 9 March 2023; Received in revised form 29 May 2023; Accepted 29 May 2023
Available online 19 June 2023

On the other hand, MCMC utilizes the full prior probability distribution and generates the posterior distribution of hyperparameters by the number of MCMC draws. While MCMC is powerful in that it incorporates the prior distribution of hyperparameters and has computational tractability for the non-Gaussian likelihood function (Titsias et al., 2008), it requires expert knowledge for the choice of prior and careful tuning of MCMC parameters to avoid a highly biased model (Bayarri et al., 2007; Liu et al., 2009).

Therefore, MLE is commonly chosen by many practitioners since it is less computationally burdensome, and it is more straightforward to implement and use. While MLE is asymptotically unbiased and considered a good estimator for large samples (Firth, 1993), it may produce biased estimates (i.e., the expected value of MLE estimates are far from the true parameter values), ill-posed (i.e., the prediction with obtained MLE estimates are sensitive to small data perturbations) (Karvonen & Oates, 2022), or cause overfitting problems under sparse data scenarios (Greenland et al., 2000) as it solely depends on training data. Moreover, the MLE objective is nonconvex (Z. Chen & Wang, 2018; Manzhos & Ihara, 2021; Mohammed & Cawley, 2017) and thus it is highly dependent on initialization and prior assumptions, which lead to convergence to various locally optimal solutions. Fitted GPR models that are either overfitted or locally optimal may fit the observed data well, but can severely violate the underlying physics of the system, especially in extrapolatory regions.

Fortunately, valuable physics-based information is available in many scientific and engineering applications in addition to the data collected. This physics-based information commonly exists as a form of equality or inequality constraints, where the constraints can either be represented as a set of algebraic equations, or as a set of ordinary differential (ODE) or partial differential equations (PDE). If this first-principle knowledge is available, then embedding it in various forms during training has been shown to improve the generalizability of the fitted surrogate models. Recent studies demonstrate the potential of GPR under this hybrid modeling framework (e.g., a combination of physics-based equalities and GPR) in forward (Albert & Rath, 2020; Gulian et al., 2022; Jidling et al., 2017b; Lange-Hegermann, 2018, 2021; Lorenzi & Filippone, 2018; Raissi & Karniadakis, 2018; Särkkä, 2011) and inverse (Rai & Tripathi, 2019; Raissi & Karniadakis, 2018; Raissi et al., 2017; S. Yang et al., 2021) problems of ODE/PDE systems. GPR with different physical constraints in the form of inequalities (e.g., bounded, monotonicity, convexity constraints) has also been widely studied (Swiler et al., 2020) via a truncated Gaussian assumption (Da Veiga & Marrel, 2012; López-Lopera et al., 2018; Maatouk & Bay, 2017; X. Wang & Berger, 2016), bounded likelihood function (Bachoc et al., 2019; Jensen et al., 2013; Riihimäki & Vehtari, 2010), constrained hyperparameter optimization (Pensoneault et al., 2020), or deep probabilistic models (Lorenzi & Filippone, 2018). While these approaches can improve the model's performance and reduce the violation of the physical constraints known to the system, there exists little systematic study on how physics-based knowledge affects the optimization of hyperparameters of GPR models. Since the posterior hyperparameters of GPR determine the success or failure of the trained model, it is crucial to obtain posterior hyperparameters that correctly capture system dynamics and avoid biased models.

A special case of hybridization that is even more challenging is when PDEs are available but directly non-solvable because initial and/or boundary conditions are not exact or unattainable (Christov, 2013; Vessella, 2015; Y. B. Wang et al., 2010; Z. Wang et al., 2021; Xiong et al., 2006). In many diverse fields including mechanical engineering (Y. B. Wang et al., 2010; Xiong et al., 2006), electromagnetic engineering (Vessella, 2015), material science engineering (Z. Wang et al., 2021), and chemical engineering (Christov, 2013; Kevrekidis et al., 2017), governing PDEs are available but non-solvable because the information at hand is incomplete. It has been studied that non-rigorous settings or imperfect knowledge of initial and boundary conditions can result in discovering wrong dynamics of the system, non-consonant between initial and boundary conditions (Kevrekidis et al., 2017) and raise different issues in numerical discretization techniques such as singularities (Flyer & Fornberg, 2003; Fornberg & Flyer, 2004) and non-convergence (Liang et al., 2021).

In order to embed physics-based information into the hyperparameter estimation process, we study the incorporation of physics-based penalty terms into the MLE function. Penalization of the MLE function has been studied before in different contexts, where it has been shown that it can provide solutions for challenges such as unbounded likelihood problems (Ciuperca et al., 2003; Ng, 2022), biased estimator problems (Firth, 1993), parameter estimation instability and overfitting problems (Cole et al., 2014; Coles & Dixon, 1999; Papukdee et al., 2022; Tamuri et al., 2014). We utilize this MLE penalization approach under the hybridization context (i.e., physics-informed ML) by incorporating a physical violation amount of the model as a penalization term into the MLE objective. It is important to note that physics-based penalization has proven to have a successful tuning effect on the prediction performance of neural networks (Raissi et al., 2019), where physics-based information is incorporated as a loss term during training via automatic differentiation (Baydin et al., 2018). A major advantage of GPR models is the analytical property that any linear transformation of a Gaussian process is also Gaussian (Rasmussen, 2003). This allows us to analytically express the physical violations as a function of hyperparameters of GPR, which is critical when embedding physics in the case where initial and/or boundary conditions are not available.

In this work, we address two important questions. First, does physics-based penalization have a significant and meaningful tuning effect on the hyperparameter estimation process under sparse data scenarios? Second, can GPR with these physics-embedded hyperparameters improve the model's generalizability and reduce the violation of the physics for systems where only partial or imperfect physics-based knowledge is available? We present three case studies where physics-based knowledge is available in the form of PDE and compare the performance of GPRs where hyperparameters are obtained via penalized MLE, standard MLE and MCMC, respectively. Two-way ANOVA analysis (Scheffe, 1999) is performed to test the significance of the physics-based penalization and the computational complexity is presented to discuss the efficiency of the method. We have observed that by penalizing the MLE objective, we can find the hyperparameter set that improves the prediction performance of GPR, while reducing the violation of physics and overfitting problems more consistently than conventional initialization approaches under sparse data scenarios.

The remainder of the paper is structured as follows. Section 2 introduces the basic terms and formulation of GPR. Section 3 explains the reformulation of the physics-based knowledge (partial differential equations (PDE)) as a function of hyperparameters of GPR. Section 4 describes the physics-based penalization approach in the context of maximum likelihood estimation. Section 5 presents three case studies and compares the prediction performance of the proposed approach and standard GPR. Section 6 provides a possible explanation for the reduction in uncertainty observed with the penalization approach, analyzes the effect of the degree of extrapolation, and presents the computational complexity of the proposed approach. Section 7 summarizes the method and describes future directions.

2. Gaussian process regression

Gaussian Process Regression (GPR) is a non-parametric model which describes the probability distribution over functions, with the assumption that every finite collection of $f(\boldsymbol{x})$ and $f(\boldsymbol{x}^{'})$ follows a multivariate Gaussian distribution (where, $\boldsymbol{x}$ and $\boldsymbol{x}^{'}$ refer to two different input locations). The logic of the Gaussian Process lies in updating the prior belief over the function $p(f|\boldsymbol{X})$ with the observation dataset $(\boldsymbol{X}, \boldsymbol{y})$ to infer the model structure via the Bayesian rule (Rasmussen, 2003). With the zero-mean prior $p(f|\boldsymbol{X}, \ \theta) \sim N(0, \ \mathrm{K})$, the posterior predictive distribution of f at single test point X^* follows

$$p(f|X^*, \boldsymbol{X},\ \boldsymbol{y},\ \theta^*)\ \sim\ N\left(k^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}\boldsymbol{y},\ k^{**}-k^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}k^*\right) \tag{1}$$

Where k^* describes the covariance between the training inputs $X^{(1)}$, $X^{(2)}, \ldots, X^{(N)}$ and test input X^*, $k^* = (k(X^{(1)}, X^*), k(X^{(2)}, X^*), \ldots, k(X^{(N)}, X^*))^T$, k^{**} is the kernel function between the test input, $k^{**} = k(X^*, X^*)$, K is the kernel matrix populated by the kernel function between training inputs, i.e., $\mathrm{K} = \begin{bmatrix} k(X^{(1)}, X^{(1)}) & \cdots & k(X^{(1)}, X^{(N)}) \\ \vdots & \ddots & \vdots \\ k(X^{(N)}, X^{(1)}) & \cdots & k(X^{(N)}, X^{(N)}) \end{bmatrix}$, θ^* is the posterior hyperparameters of GPR and σ_n^2 is the variance of Gaussian observation noise in $\boldsymbol{y}$.

The kernel function $k(\cdot,\cdot)$ describes the covariance between the outputs, and it is commonly selected based on our belief in the system. The popular Gaussian kernels assume smoothness over function:

$$k\left(X^{(i)}, X^{(j)}\right) = cov\left(f\left(X^{(i)}\right), f\left(X^{(j)}\right)\right) = \tau^2\exp\left(-\frac{1}{2}\sum_{q=1}^{p} w_q\left(x_q^{(i)} - x_q^{(j)}\right)^2\right) \tag{2}$$

Where $X^{(i)}$ is the i^{th} observation of a p-dimensional input space, $X^{(i)} = (x_1^{(i)}, x_2^{(i)} \ldots, x_p^{(i)})$, and τ^2 and w_q are the hyperparameters of the kernel. Hyperparameters of GPR (i.e., $\theta = [\tau^2, w_q, \sigma_n^2]$) are often estimated by maximizing the marginal likelihood function $p(\boldsymbol{y}|\boldsymbol{X})$:

$$\begin{aligned} \theta^* &= \underset{\theta}{\operatorname{argmax}}[logp(\boldsymbol{y}|\boldsymbol{X}, \theta)] \\ &= \underset{\theta}{\operatorname{argmin}}\left[\frac{1}{2}log\left|\mathrm{K}+\sigma_n^2\mathrm{I}\right| + \frac{1}{2}\boldsymbol{y}^T\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}\boldsymbol{y} + \frac{N}{2}log2\pi\right] \end{aligned} \tag{3}$$

The aforementioned procedure is outlined in Algorithm 1 in the Appendix F.

3. Reformulation of the partial differential equation (PDE) with Gaussian process models

A useful analytical property of a Gaussian Process (GP) is that when a GP model is linearly transformed, the transformation also follows a GP. This property has been previously used to embed derivative/integral information into the GP modeling (Albert & Rath, 2020; Graepel, 2003; Gulian et al., 2022; Jidling et al., 2017a; Lange-Hegermann, 2018; Morris et al., 1993; Rai & Tripathi, 2019; Raissi et al., 2017; Solak et al., 2002; H. Wang & Zhou, 2021). Here, we utilize this property for the system where physics-based information is given as a form of linear PDE (i.e., linear transformation of the response variable f). We denote the linear operator $\mathscr{L}$ from PDE in a way that it satisfies $\mathscr{L}(f) = 0$. As an illustrative example, consider the heat equation, described by the following PDE:

$$\frac{\partial f}{\partial t} = \frac{\partial^2 f}{\partial x^2} \tag{4}$$

where f is the heat (response variable) and t and x is the time and the position (input variables). Then the linear operator $\mathscr{L}$ is defined as

$$\mathscr{L}(\cdot) = \frac{\partial(\cdot)}{\partial t} - \frac{\partial^2(\cdot)}{\partial x^2} \tag{5}$$

Note that the linear operator $\mathscr{L}$ is defined to meet $\mathscr{L}(f) = 0$. For simplicity, let us denote input x as x_1 and t as x_2. Let $X^{(i)}$ be the i^{th} input space observation, then $X^{(i)} = (x^{(i)}, t^{(i)}) = (x_1^{(i)},\ x_2^{(i)})$ where $i = 1, 2, \ldots, N$, and let the single test point X^* as $X^* = (x^*, t^*) = (x_1^*,\ x_2^*)$. Here, $\boldsymbol{X}$ is the N × 2 observation input matrix, $\boldsymbol{X} = ((X^{(1)})^T, (X^{(2)})^T, \ldots, (X^{(N)})^T)^T$.

By utilizing GP's analytical property that the linear transformation of GP follows a Gaussian Process (Rasmussen, 2003), we can derive the explicit GP posterior predictive distribution for each derivative term:

$$f|X^*, \boldsymbol{X},\ \boldsymbol{y}\ \sim\ N\left(k^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}\boldsymbol{y},\ k^{**}-k^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}k^*\right) \tag{6}$$

$$\frac{\partial f}{\partial x_2^*}|X^*, \boldsymbol{X}, \boldsymbol{y}\ \sim\ N\left(k_{x_2^*}^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}\boldsymbol{y},\ k_{x_2^*}^{**}-k_{x_2^*}^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}k_{x_2^*}^*\right) \tag{7}$$

where $k_{x_2^*}^* = \frac{\partial}{\partial x_2^*}(k(X^{(1)}, X^*), k(X^{(2)}, X^*), \ldots, k(X^{(N)}, X^*))^T$, and $k_{x_2^*}^{**} = \frac{\partial^2}{\partial x_2^* \partial x_2^*}k(X^*, X^*)$. Note that we have direct access to the functional form of GP posterior predictive distribution before estimating the hyperparameters θ^* from the MLE function. Each component in vector $k_{x_2^*}^*$ is calculated as

$$\begin{aligned} \frac{\partial}{\partial x_2^*}k\left(X^{(i)}, X^*\right) &= \frac{\partial}{\partial x_2^*}\left[\tau^2\exp\left(-\frac{1}{2}\sum_{q=1}^{2} w_q\left(x_q^{(i)} - x_q^*\right)^2\right)\right] \\ &= w_2\left(x_2^{(i)} - x_2^*\right)k\left(X^{(i)}, X^*\right) \end{aligned} \tag{8}$$

Similarly, the second derivative term f_{xx} follows the GP posterior predictive distribution:

$$\frac{\partial^2 f}{\partial x_1^{*2}}|X^*, \boldsymbol{X},\ \boldsymbol{y} \sim\ N\left(k_{x_1^{*2}}^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}\boldsymbol{y},\ k_{x_1^{*2}}^{**}-k_{x_1^{*2}}^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}k_{x_1^{*2}}^*\right) \tag{9}$$

where $k_{x_1^{*2}}^{*T} = \frac{\partial^2}{\partial x_1^{*2}}(k(X^{(1)}, X^*), k(X^{(2)}, X^*), \ldots, k(X^{(N)}, X^*))^T$ and $k_{x_1^{*2}}^{**} = \frac{\partial^4}{\partial x_1^{*2} \partial x_1^{*2}}k(X^*, X^*)$. Each component in vector $k_{x_1^{*2}}^{*T}$ is calculated as

$$\frac{\partial^2}{\partial x_1^{*2}}k\left(X^{(i)}, X^*\right) = \frac{\partial}{\partial x_1^*}\left[w_1\left(x_1^{(i)} - x_1^*\right)k\left(X^{(i)}, X^*\right)\right]$$

$$= w_1\left(w_1\left(x_1^{(i)} - x_1^*\right)^2 - 1\right)k\left(X^{(i)}, X^*\right) \tag{10}$$

Likewise, the PDE can be reformulated with the GP posterior predictive distribution at different testing locations $X^{*,(j)} = (x_1^{*,(j)},\ x_2^{*,(j)})$, where $j = 1, 2, \ldots, N_{test}$. We denote the mean of the GP posterior predictive distribution at test point X^* after a linear transformation as $\mathscr{L}(f)_{mean}$:

$$\mathscr{L}(f)_{mean} = \frac{\partial f}{\partial x_2^*} - \frac{\partial^2 f}{\partial x_1^{*2}}$$

$$= k_{x_2^*}^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\ \right)^{-1}\boldsymbol{y} - k_{x_1^{*2}}^{*T}\left(\mathrm{K}+\sigma_n^2\mathrm{I}\ \right)^{-1}\boldsymbol{y}$$

$$= \left(k_{x_2^*}^{*T} - k_{x_1^{*2}}^{*T}\right)\left(\mathrm{K}+\sigma_n^2\mathrm{I}\ \right)^{-1}\boldsymbol{y} = \mathscr{K}^*\left(\mathrm{K}+\sigma_n^2\mathrm{I}\ \right)^{-1}\boldsymbol{y} \tag{11}$$

Here, $\mathscr{L}(f)_{mean}$ is a function of test input X^* and the hyperparameters θ given observation dataset $(\boldsymbol{X}, \boldsymbol{y})$. The above steps show how physics-based knowledge can be reformulated as a function of hyperparameters in GPR. These steps in GPR have an analogous role as an automatic differentiation (Baydin et al., 2018) employed in physics-informed Neural Networks (Raissi et al., 2019).

4. Penalization of physics-based knowledge in maximum likelihood estimation

Here, we introduce physics-based knowledge as a form of $\mathscr{L}(f)_{mean}$ into the marginal likelihood function as shown in Eq. (3). We use the L_2-norm squared of the linearly-transformed mean predictive distribution $\|\mathscr{L}(f)_{mean}\|_2^2$ as a Physics Violation (PV) function denoted by $PV(\theta, X^*|\boldsymbol{X}, \boldsymbol{y})$. Note that PV is a function of test input $X^* = (x_1^*,\ x_2^*)$, and we define the collection of test inputs for calculating PV as collocation points X_{col}^*. These collocation points are also referred to as *virtual points* in constrained GPR fields, where the physics-based constraints are imposed (Da Veiga & Marrel, 2012; Golchi et al., 2015; Riihimäki & Vehtari, 2010; Swiler et al., 2020; X. Wang & Berger, 2016).

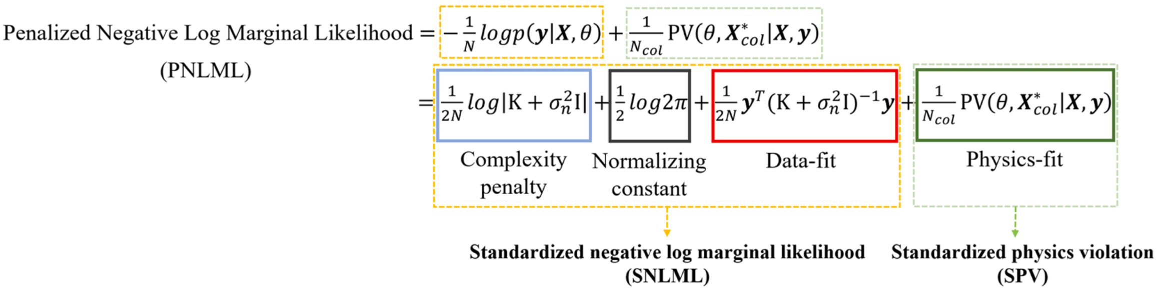
Penalized Negative Log Marginal Likelihood (PNLML)
$= -\frac{1}{N} logp(\boldsymbol{y}|\boldsymbol{X},\theta) + \frac{1}{N_{col}} \mathrm{PV}(\theta, \boldsymbol{X}^*_{col}|\boldsymbol{X},\boldsymbol{y})$
$= \frac{1}{2N} log|\mathrm{K}+\sigma_n^2\mathrm{I}| + \frac{1}{2} log2\pi + \frac{1}{2N}\boldsymbol{y}^T(\mathrm{K}+\sigma_n^2\mathrm{I})^{-1}\boldsymbol{y} + \frac{1}{N_{col}} \mathrm{PV}(\theta, \boldsymbol{X}^*_{col}|\boldsymbol{X},\boldsymbol{y})$
Complexity penalty
Normalizing constant
Data-fit
Physics-fit
Standardized negative log marginal likelihood (SNLML)
Standardized physics violation (SPV)

Fig. 1. Interpretation of Penalized Negative Log Marginal Likelihood (PNLML).

Table 1
Priors used in the performed experiments.

	Prior Distribution
Prior 1	$\theta_i \sim Uniform(0,1)$
Prior 2	$\log(\theta_i) \sim Uniform(-5,5)$
Prior 3	$\theta_i \sim inv\ \chi^2(2)$
Prior 4	$\theta_i \sim Gamma(7.5,\ 1)$

Since physics-based knowledge indicates that the linearly-transformed mean predictive distribution $\mathscr{L}(f)_{mean}$ should be zero for everywhere in the system domain $X^* \in \Omega_{system}$, any non-zero amount of the PV function suggests some degree of violation of underlying physics. From this observation, we introduce $\| \mathscr{L}(f)_{mean}\|_2^2$ into the negative marginal likelihood function while standardizing each component by the number of observation data points N and the number of collocation points N_{col}, respectively.

$$\theta^*_{physics} = \underset{\theta}{\mathrm{argmin}}\left[-\frac{1}{N}logp(\boldsymbol{y}|\boldsymbol{X},\theta) + \frac{1}{N_{col}}\mathrm{PV}\left(\theta, \boldsymbol{X}^*_{col}|\boldsymbol{X},\boldsymbol{y}\right)\right] \tag{12}$$

Fig. 1 describes the interpretation of each term in the augmented marginal likelihood function (i.e., penalized negative log marginal likelihood). The first term indicates the complexity penalty of a model, the second term is a normalizing constant, and the third term determines the data-fit (Rasmussen, 2003). The newly augmented fourth term is constructed from the physics-based knowledge and determines the physics-fit of the model. This augmented objective function considers the model's physics-fit to the given physics-based knowledge and the data-fit to the training data. We minimize this augmented objective function using the L-BFGS-B optimization algorithm (Zhu et al., 1997) to get a new set of physics-informed hyperparameters. The aforementioned procedure is summarized in Algorithm 2 in the Appendix F.

It is important to note that the standard marginal likelihood function is non-convex and the convergence to a global optimum is not guaranteed. Locally optimal hyperparameters can lead to poor extrapolation and interpretability, and cause overfitting problems. A common approach to tackle this issue is to use multiple starting points from a specific prior distribution, perform multi-start local optimization, and choose the hyperparameters with the largest marginal likelihood (Z. Chen & Wang, 2018). Since the PNLML is a non-convex function as well, we also perform multiple initializations for the penalization approach as well as the conventional approach. This analysis will allow us to analyze the difference in consistency of convergence and statistical significance in the results between the physics-informed approach and the black-box GPR approach.

5. Results

Three case studies are introduced to discuss the effect of the physics-based penalization in GPR. Three different hyperparameter estimation processes are presented - MLE, pMLE (MLE with physics-based penalization), and MCMC, where MLE and pMLE are the point estimation methods and MCMC is the full Bayesian treatment of the GPR in terms of hyperparameter estimation and prediction. Since neither informative priors nor careful tuning of MCMC parameters from domain knowledge is involved here, we consider MLE and MCMC methods as purely data-driven methods.

The hyperparameter estimation process in GPR and pGPR is affected by various factors such as choice of kernels, optimization algorithm and parameter settings, priors of hyperparameters, distribution/number of observation data, and location and amount of collocation points. In this paper, we investigate the effect of three important factors: (a) Priors of hyperparameters, (b) distribution/number of observation data, and (c) collocation points. We use a squared exponential kernel with a prior mean of 0, $f|X,\theta \sim N(0,\mathrm{K})$ and the L-BFGS-B optimization algorithm (Zhu et al., 1997) for all case studies.

In point estimation methods (MLE and pMLE), four different non-informative priors (Chen & Wang, 2018; Wilson & Adams, 2013) of hyperparameters in Table 1 are used throughout the experiment to see how the penalized objective guides the MLE process using different non-informed starting points. Here, multistart optimization is also performed for each prior because objective functions are non-convex and L-BFGS-B is a local solver. A full Bayesian treatment with Metropolis-Hastings MCMC algorithm (Chib & Greenberg, 1995) is performed with the standard likelihood function and with Prior 4 in Table 1, using the R package *mcmc* (Geyer & Johnson, 2013). 1000 MCMC draw (samples) is used with 100 Burn-in (i.e., discard 100 initial samples) for all case studies.

The distribution of training data (X, y) for a given system domain is another important factor that determines the interpolation and extrapolation prediction performance of the method. This is because GPR prediction converges to a (zero) prior mean if the test location is far from the training region (Appendix A). To take this factor into account, we introduce the space-filling degree (SFD) of the training data.

SFD is measured by computing the ratio of the convex hull area A_{convex} (constructed from the training data) to the entire system domain area A_{system}. An additional condition on the distance between any two training data points is added to impose uniformity in the sampling and avoid extreme cases where most data is clustered. The sampling criteria $\Omega_{\mathscr{D}}$ for the case studies presented in Sections 5.1. (Laplace equation) and 5.2. (Heat equation) are shown in Eq. (13).

$$\Omega_{\mathscr{D}} = \left[\mathscr{D}:\ \ \mathrm{SFD}\ \geq 0.75, \min\left(\mathscr{D}_i, \mathscr{D}_j\right) \geq 0.2\right] \tag{13}$$

where $\mathrm{SFD} = \frac{\mathrm{A}_{convex}}{\mathrm{A}_{system}}$ and $\mathscr{D}_i$, $\mathscr{D}_j$ are samples drawn from the training dataset ($i, j = 1, 2, \ldots, N$ where N is the number of training data).

Prediction performance is evaluated using the standardized root mean squared error (SRMSE) and the mean standardized log loss (MSLL) on the test dataset. Note that MSLL incorporates predictive variance (uncertainty) of the trained model in addition to the prediction error. Usually, the smaller MSLL indicates a better model has been identified (Rasmussen, 2003).

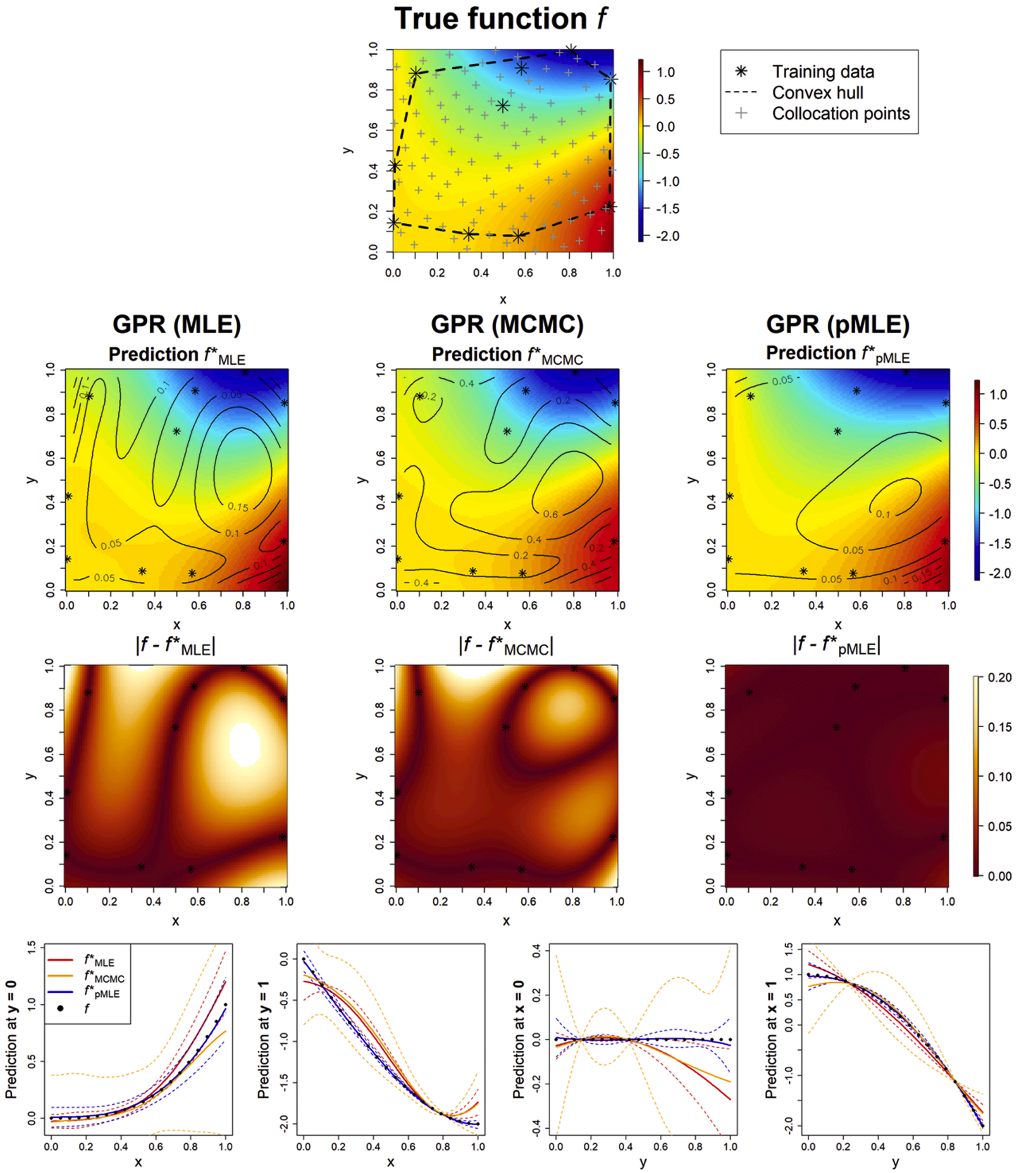

Fig. 2. *Laplace's equation: Top*: True function f and the distribution of training data (Set 9 in Fig. 3) and collocation points. *Second row*: Prediction performance of GPR for three different hyperparameter estimation methods (MLE, MCMC, and pMLE). *Third row*: Error plot between the true function f and each prediction method. *Bottom*: Prediction at four system boundaries in the system. The 95% confidence intervals ($1.96 \times \sigma$) are plotted as a dotted line.

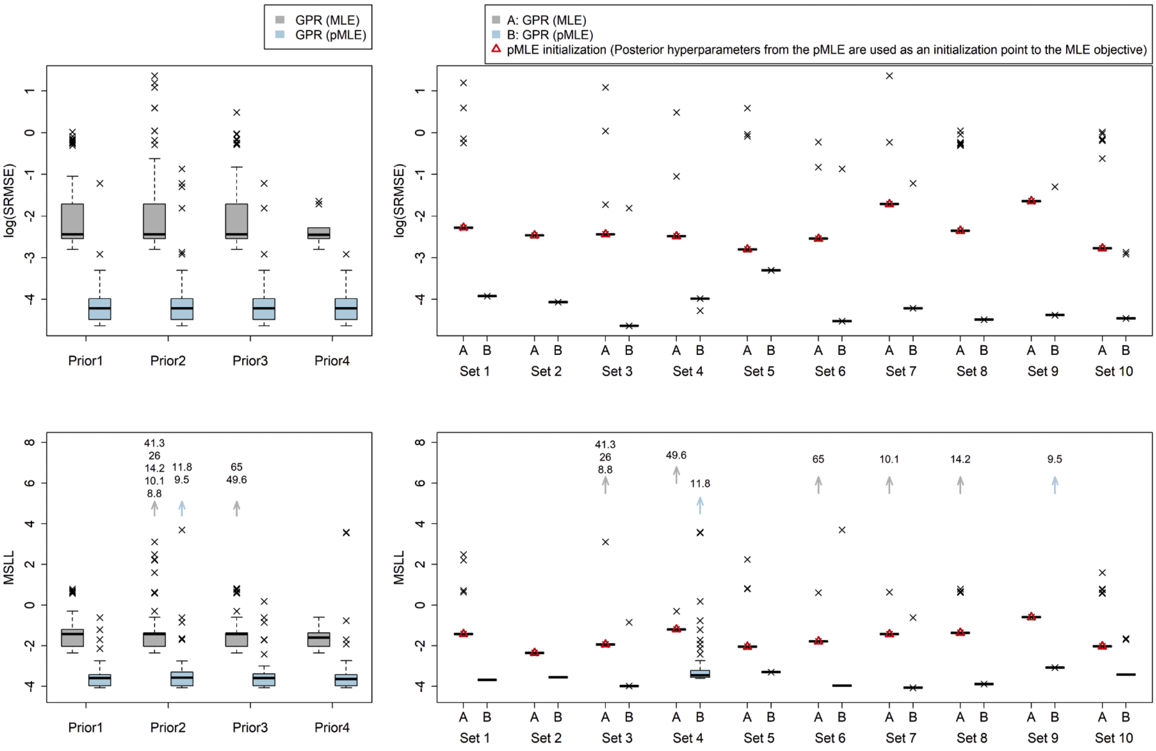

Fig. 3. *Laplace's equation: Left*: Boxplot of log(SRMSE) and MSLL for four different priors in Table 1. 25 initializations are performed for each prior in the optimization. Each boxplot shows the combined results of 10 different training set scenarios and contains 250 data points (25 × 10). *Right*: Boxplot of log(SRMSE) and MSLL for 10 different training set scenarios (i.e., Set 1, Set 2, …, Set 10). Each boxplot shows the combined results of 4 different priors and contains 100 data points (25 × 4). The red triangle points show prediction error when the **best** posterior hyperparameters (i.e., posterior hyperparameters that produce the lowest SRMSE) from the pMLE are used as an initialization point for the optimization of the standard MLE objective function. *Left* and *Right Plots* show the boxplot of the same dataset but with a different axis, and the outliers outside of the y-range are presented as the numbers for the corresponding cases. The box plot results for different training datasets are provided in Table C in Appendix C with the MCMC prediction results.

$$\mathrm{SRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \tilde{y}_i\right)^2}}{\sigma_y} \tag{14}$$

$$\mathrm{MSLL} = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{1}{2}\log\left(2\pi\tilde{\sigma}_i^2\right) + \frac{\left(y_i - \tilde{y}_i\right)^2}{2\tilde{\sigma}_i^2}\right] \tag{15}$$

Here, y_i is the test output, $\tilde{y}_i$ is the predicted output, and $\tilde{\sigma}_i^2$ is the predicted variance where $i = 1, 2, \ldots, N$. σ_y is the standardized deviation of y_i, and it is taken into account to consider the scale of the output values. If the MSLL does not converge at the test point ($\tilde{\sigma}_i^2 \rightarrow 0$), we do not include this test point in the calculation.

5.1. Laplace's equation

Laplace's equation is a partial differential equation (PDE) that has different cartesian solutions $f = x^3 - 3xy^2, y^3 - 3x^2y, \cos(kx)\cosh(ky), \cos(kx)e^{ky}, e^{-kx}\sin(ky), \ldots$ depending on the boundary conditions specified for the system:

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = 0 \tag{16}$$

We consider the case where the boundary condition is unknown to the modeler so that the true solution cannot be estimated from numerical discretization techniques.

In order to train and evaluate the model performance, we first assume that the true solution of the system is $f(x, y) = x^3 - 3xy^2$ and sample 200 data points from f using *Maxpro* (Joseph et al., 2020) space-filling design. Out of 200 data points, we sample 10 training data points under criteria $\Omega_{\mathscr{D}}$ (equation (13)). The rest of the points are used for testing model accuracy. This is repeated 10 times to quantify the generalizability of the methods for different training/test data scenarios. In the penalized marginal likelihood function (pMLE), the physics violation function is calculated on 100 collocation points (X^*_{col}), uniformly sampled with *Maxpro* (Joseph et al., 2020). Note that the only available knowledge at hand is the 10 sparse observation points and the physics-based knowledge (PDE *without* the boundary conditions). We observe how physics-based penalization can help tune the hyperparameters so that the trained model can better represent the true system.

Fig. 2 shows the true solution f and the prediction performance of trained GPR models via three different hyperparameter estimation methods (f^*_{MLE}: MLE, f^*_{MCMC}: MCMC, f^*_{pMLE}: pMLE) for one of the training/test data scenarios. From 25 multiple initializations for each prior in Table 1, the hyperparameter set which produces the lowest SRMSE test error is selected for model prediction in Fig. 2.

As expected, the prediction performance of data-driven methods (f^*_{MLE}, f^*_{MCMC}) is reasonable at training points, however, they become poor in the region where data is not observed. Since standard GPR (f^*_{MLE}, f^*_{MCMC}) only relies on training data, the trained model is not able to determine how to behave in the region where dynamics are not observed. Moreover, it is clearly shown from the error plot in Fig. 2 that the data-driven approach suffers from overfitting. On the other hand, the physics-based penalization approach (f^*_{pMLE}) can reduce overfitting problems in these unexplored regions (the region where no training data is observed) and train models to reflect the underlying physics of the system.

Fig. 3 shows the box plot of the prediction performance of different methods for four different priors and 10 different training/test data scenarios. Here, 25 initializations are performed for each prior for 10

Table 2
Laplace's equation: The p values from two-way ANOVA analysis (Scheffe, 1999) for different priors and the methods

	SRMSEs	MSLL
Priors	0.001	0.060
Methods (MLE vs pMLE)	$< 2.2\text{e-}16$	$< 2.0\text{e-}16$

different training datasets, respectively. These results indicate that the hyperparameters estimated with the physics-based penalization can reduce the prediction error as well as uncertainty (lower SRMSE and MSLL) for different priors and for different training/test data scenarios.

Note that multiple initializations are required for the point estimation methods (MLE and pMLE) because of the non-convexity of the objective functions. Non-informative prior settings (initializations) of the hyperparameters can result in outliers that cause poor prediction performance in both cases. However, the penalization approach can reduce the number of trials of initialization, produces a smaller number of outliers, and give us more consistent results.

Table 2 shows the p values calculated from the two-way ANOVA analysis (Scheffe, 1999) for different priors and point estimation methods. The p value in Table 2 determines whether each factor (prior type, methods type) has a significant impact on SRMSEs and MSLLs, and the small p value indicates the significant difference between groups. The results reveal that the penalization has a statistically significant effect on both SRMSEs and MSLLs. Here, it is interesting to see that physics embedding as a form of a *mean* of physical violation function has a significant effect on MSLL (i.e., uncertainty information). This may be due to the benefit of reducing overfitting problems, which will be discussed in Section 6. While physics-based penalization has a significant impact on both prediction accuracy and the uncertainty information, the selection of a prior distribution does not show a significant effect on the MSLL at a 5% significance level, which agrees with a previous study (Z. Chen & Wang, 2018) showing that priors for hyperparameter tuning have no notable impacts on the model performance.

Fig. 4 shows a schematic illustration of two different objective functions for MLE and pMLE. This visualization shows the effect of physics-based penalization on the objective function, which identifies significantly different local optima influenced by highly physics-violated regions.

Table 3 shows the average values of SRMSE, MSLL, SNLML, and SPV (Fig. 1) for three different approaches. It is observed that the SPV is estimated to be very large in data-driven methods (MLE, MCMC), while the physics-based penalization approach (pMLE) favors the region where the physics violation is small and reduces the violation of underlying physics by sacrificing the data-fit (Low SPV and large SNLML value). Note that there is a big difference in the scale of values between the SNLML and SPV (e.g., 0.436 (SNLML) and 722 (SPV) for GPR (MLE) in Table 3). This is because SPV involves the second derivative information and the L2-norm square calculation. However, it is interesting that sacrificing the data-fit in pMLE does not lead to *violating* the observation data-fit of the model, and a significant reduction of physical violation can still be achieved. In other words, including the physics violation term in the marginal likelihood function can mitigate overfitting problems by balancing the data-fit and the physics-fit. This observation agrees with the previous study that the Maximum penalized likelihood estimation (MPLE) approach can reduce the overfitting problems when samples are limited (Tamuri et al., 2014).

5.2. *Heat equation*

The distribution of heat f is described by the following PDE:

$$\frac{\partial f}{\partial t} = \frac{\partial^2 f}{\partial x^2} \tag{17}$$

where the output heat f is a function of position x and time t. We assume $f = e^{-6.25t}\cos(2.5x) - e^{-t}\cos(x-1)$ is the true dynamics of the system, and sample 200 total data points from the true solution in domain $x \in [0, 1]$, $t \in [0, 1]$ using *Maxpro* space-filling design (Joseph et al., 2020). Out of 200 total data points, 10 training data points are sampled under criteria $\Omega_{\mathscr{D}}$ (Eq. (13)) and the rest are saved as a test set. Using the above scheme, 10 different training/test datasets are generated to evaluate the model performance.

As in the previous case study, we assume that the true dynamics of the system is unknown, but the modeler has the domain knowledge that the system should follow the underlying physics $f_t = f_{xx}$. Note that there are infinite solutions f that meet the equation $f_t = f_{xx}$ without initial and boundary conditions, but we constrain the solution space by utilizing the sparse observation data. Here, 200 collocation points are sampled over the domain using *Maxpro* (Joseph et al., 2020) and 25 initializations are performed for each prior in Table 1 for each training dataset.

Fig. 5 shows that the penalization with physics-based knowledge (f^*_{pMLE}) helps improve the prediction performance within the system

Table 3
Laplace's equation: Average log(SRMSE), MSLL, SNLML, and SPV for three different hyperparameter estimation methods. (Brackets show the standard errors)

	Log(SRMSE)	MSLL	SNLML	SPV
GPR (MLE)	-2.23 (0.613)	-1.29 (3.250)	0.436 (0.317)	7.22e2 (8.39e3)
GPR (MCMC)	-1.99 (0.248)	-1.15 (0.131)	0.706 (0.097)	3.72e1 (7.17)
GPR (pMLE)	-4.15 (0.486)	-3.53 (0.898)	0.879 (0.128)	1.27 (2.89e1)

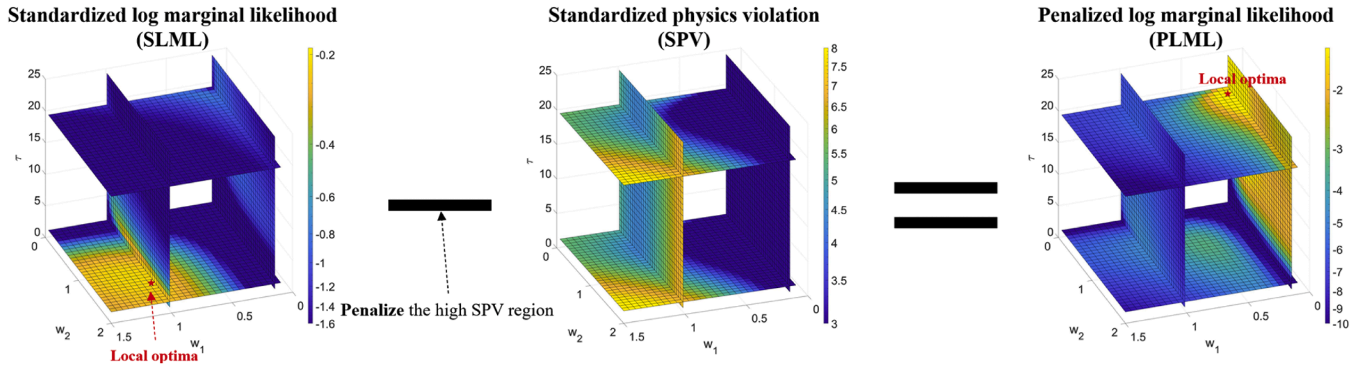

Fig. 4. *Laplace's equation*: Standardized log marginal likelihood (SLML), Standardized physics violation (SPV) and the penalized log marginal likelihood (PLML) values for three different hyperparameters (w_1, w_2, τ) locations when Set 1 in Fig. 3 is used. One of the local optima is presented for each objective function, respectively. Log scale is used for better visualization of local optima in each objective function.

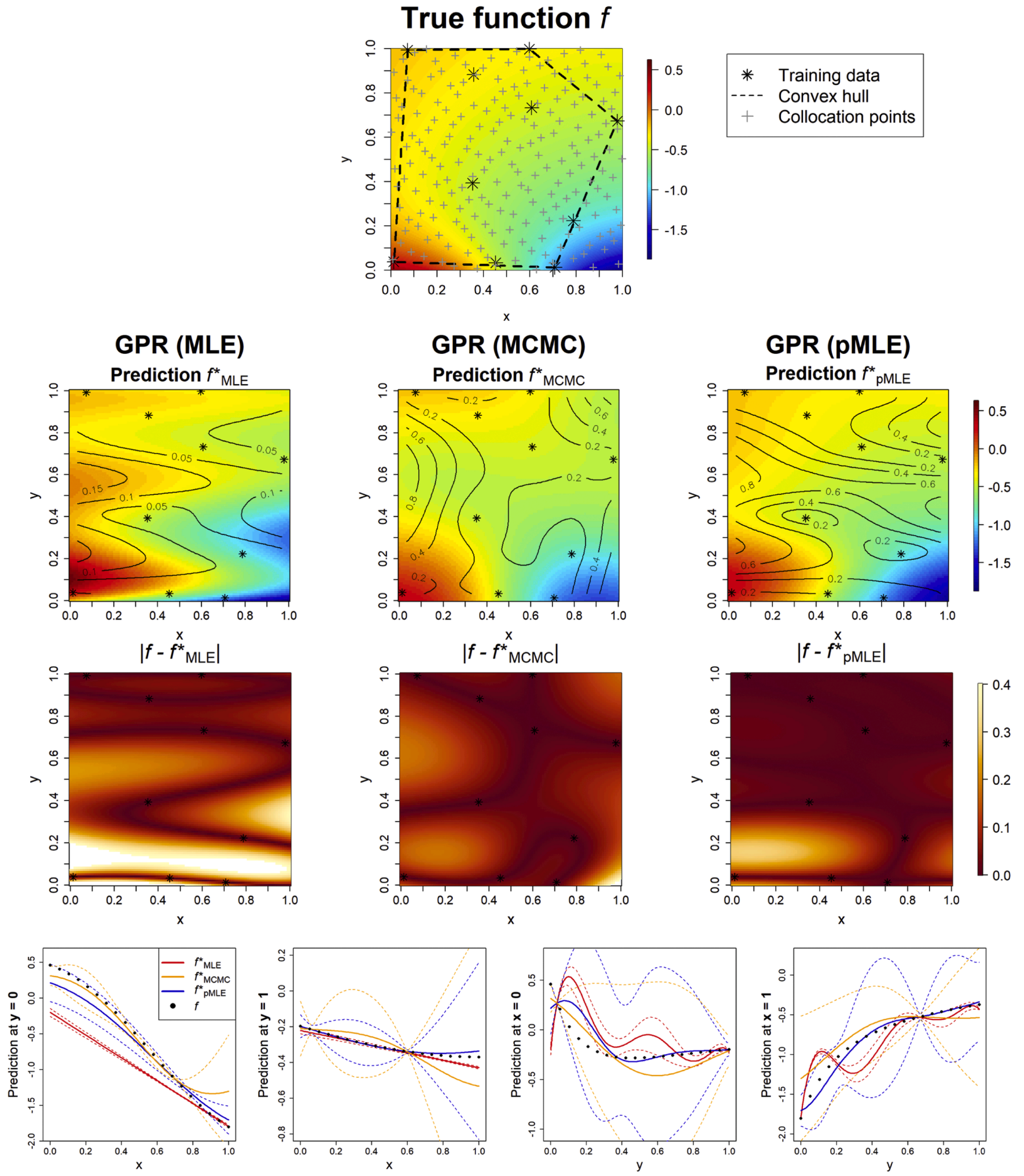

Fig. 5. *Heat equation: Top*: True function f and the distribution of training data (Set 8 in Fig. 6) and collocation points. *Second row*: Prediction performance of GPR for three different hyperparameter estimation methods (MLE, MCMC, and pMLE). *Third row*: Error plot between the true function f and each prediction method. *Bottom*: Prediction at four system boundaries in the system. The 95% confidence intervals ($1.96 \times \sigma$) are plotted as a dotted line.

domain and at the four different boundaries of the system, when compared to the standard data-driven GPR (f^*_{MLE}, f^*_{MCMC}). Since physics-based penalization considers the physics-fit on the system domain through the collocation points, it mitigates overfitting problems by reducing the violation of the underlying physics of the system.

Fig. 6 shows the boxplots of the prediction performance for different hyperparameter estimation methods for different priors in Table 1 and for 10 different training/test data scenarios. The general trend shows that the prediction error (SRMSEs and MSLL) is decreased when the hyperparameters are estimated from the penalized marginal likelihood

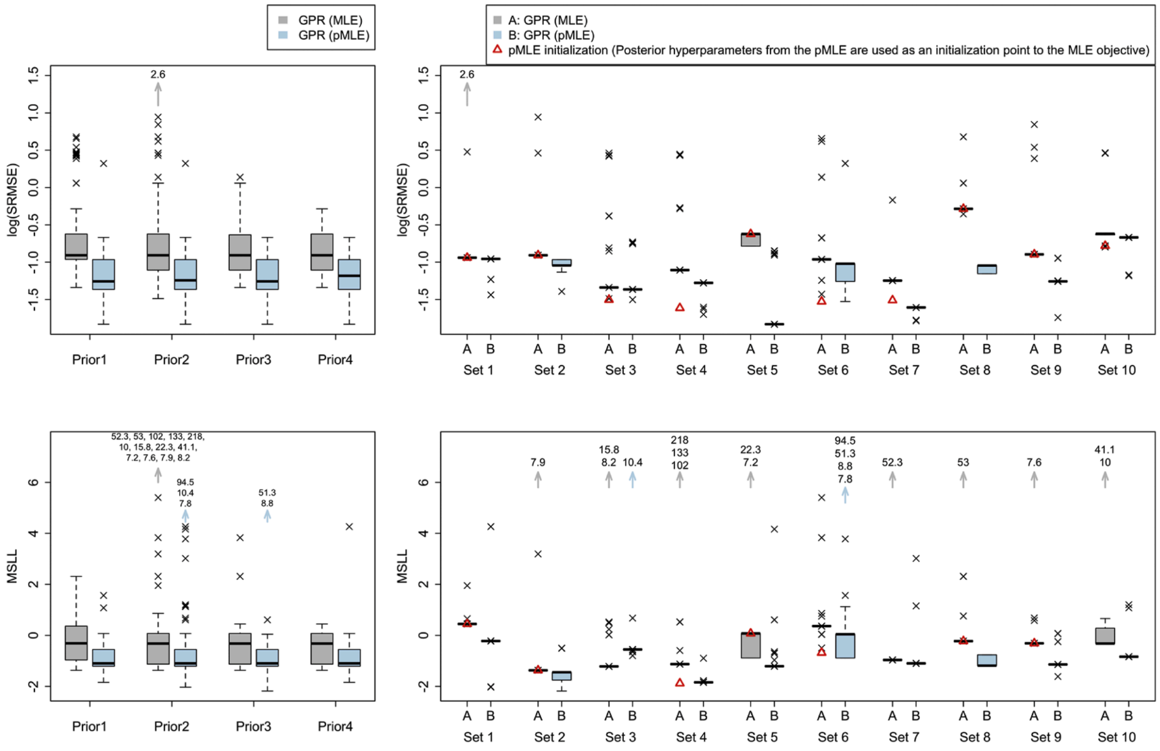

Fig. 6. *Heat equation: Left*: Boxplot of log(SRMSE) and MSLL for four different priors in Table 1. 25 initializations are performed for each prior in the optimization. Each boxplot shows the combined results of 10 different training set scenarios and contains 250 data points (25 × 10). *Right*: Boxplot of log(SRMSE) and MSLL for 10 different training set scenarios (i.e., Set 1, Set 2, …, Set 10). Each boxplot shows the combined results of 4 different priors and contains 100 data points (25 × 4). The red triangle points show prediction error when the **best** posterior hyperparameters (i.e., posterior hyperparameters that produce the lowest SRMSE) from the pMLE are used as an initialization point for the optimization of the standard MLE objective function. *Left* and *Right Plots* show the boxplot of the same dataset but with a different axis, and the outliers outside of the y-range are presented as the numbers for the corresponding cases. The MSLL for Set 7 and Set 10 for pMLE initialization (red triangle) are 7.75 and 40.30, respectively, and omitted due to the range of the plot. The test points that have zero variance are excluded from MSLL calculation, which results in higher MSLL in pMLE for set 3 (12 test points have zero variance with hyperparameters obtained from pMLE). The box plot results for different training datasets are provided in Table C in Appendix C with the MCMC prediction results.

Table 4
Heat equation: The p values (Scheffe, 1999) for different priors and the methods

	SRMSEs	MSLL
Priors	0.0187	0.0002
Methods (MLE vs pMLE)	< 2e-16	0.0012

function.

In this case study however, it is important to note that embedding physics-based information during non-convex optimization does not always outperform standard GPR (e.g., Set 3 in Fig. 6), even when the penalization has a statistically significant effect on the prediction of the model (Table 4). One possible reason is that the 100 multiple initializations from four different non-informative priors are not sufficient to find good local optima. For example, Standard GPR (e.g., Set 4 and 7 in Fig. 6) could find a new hyperparameter set (red triangle) when the posterior hyperparameters from the penalized marginal likelihood function are used as an initialization point in the optimization of the standard marginal likelihood function. Note that this newly found hyperparameter set is not discoverable in the previous simulation with 100 initializations from different priors. This indicates the possibilities of an unexplored region of local optima. The other reason may include

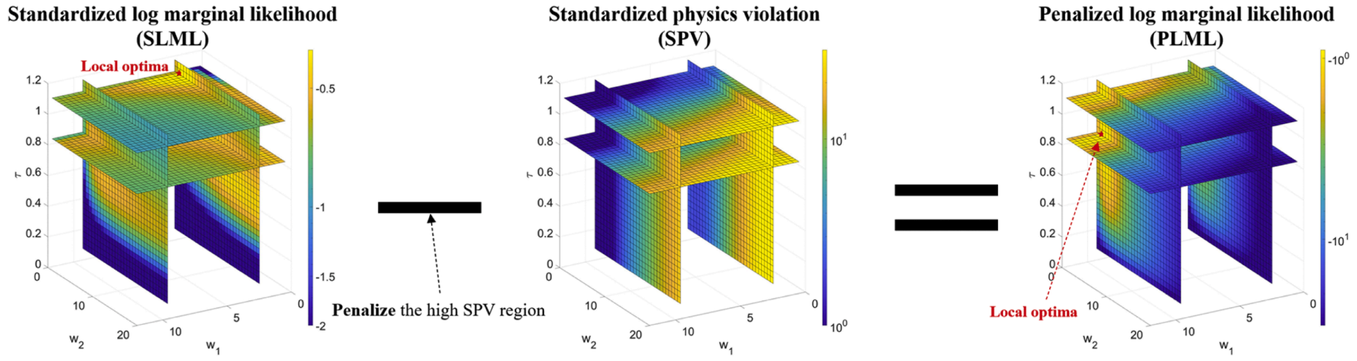

Fig. 7. *Heat equation: Left*: Standardized log marginal likelihood (SLML). *Middle*: Standardized physics violation (SPV). *Right*: penalized log marginal likelihood (PLML) values for three different hyperparameters (w_1, w_2, τ) locations when Set 5 in Fig. 6 is used. One of the local optima is presented for each objective function, respectively. Log scale is used for better visualization of local optima in each objective function.

Table 5
Heat equation: Average log(SRMSE), MSLL, SNLML, and SPV for three different hyperparameter estimation methods. (Brackets show the standard errors)

	Log(SRMSE)	MSLL	SNLML	SPV
GPR (MLE)	-0.841 (0.380)	0.263 (9.20)	0.282 (0.298)	1.30e2 (1.10e3)
GPR (MCMC)	-1.03 (0.226)	-1.18 (0.184)	0.520 (0.106)	5.61 (2.19)
GPR (pMLE)	-1.21 (0.326)	-0.748 (3.54)	0.403 (0.335)	0.766 (2.12)

the use of a stationary Gaussian kernel under sparse data for explaining the non-stationary dynamics described by the heat equation. To accurately capture the complex dynamics of a system, a modeler may use a more generalizable kernel (e.g., Matern kernel (Rasmussen, 2003), additive Gaussian Kernel (Duvenaud et al., 2011)) or advanced GPR modeling framework (e.g., deep Gaussian process (Damianou & Lawrence, 2013)) and combine it with the physics-based penalization approach.

Fig. 6 (red triangle) indicates that the posterior hyperparameter set obtained from physics-based penalization often acts as an *informative initialization* that can help find better local optima with the standard marginal likelihood. Similarly, a modeler can use the posterior hyperparameters obtained from the marginal likelihood estimation as an initialization/starting point for optimizing the penalized marginal likelihood function, instead of starting from a non-informative prior.

Fig. 7 shows that the physics-based penalization transforms the marginal likelihood function into a new objective that embeds the physics-based knowledge and changes the landscape entirely. It can also be observed based on this mapping that the purely data-driven landscape is more “flat”, with multiple equivalent locally optimal solutions across the search space, while the PLML objective landscape creates a surface that has a narrower basin of local optima, mostly clustered in a smaller region of the space. This may explain why when using this hybrid objective approach, we could more consistently find similar local optima with improved performance.

Table 5 shows that the standard data-driven GPR (MLE, MCMC) produces a high SPV value and lower SNLML, which indicates the high violation of physics-based information and thus overfitting problems. This is expected, because it only relies on the available dataset without considering any physics-based knowledge. GPR with physics-based penalization (pMLE) balances the data-fit and the physics-fit and successfully finds a new hyperparameter set that produces a lower prediction error while mitigating overfitting problems.

5.3. Fiber orientation probability distribution (FOPD) model

The aim of this case study is to highlight the potential of the proposed method to capture the underlying nature of a system’s response, when limited and noisy experimental data are available. We consider the Fiber Orientation Probability Distribution (FOPD) model (Olson et al., 2004) which is derived from the Fokker-Planck equation (Risken, 1996). The FOPD model describes the time evolution of the probability density function of particles within a fluid. It is analogous to the standard convection-diffusion equation used for molecular diffusion and heat transfer, but in the FOPD model, the convection and diffusion are determined by fiber orientation angle and the turbulent dispersion, respectively (Olson et al., 2004). This model is used to describe fiber orientation of pulp suspensions in paper manufacturing machines, which is a property linked to final paper quality and tensile strength.

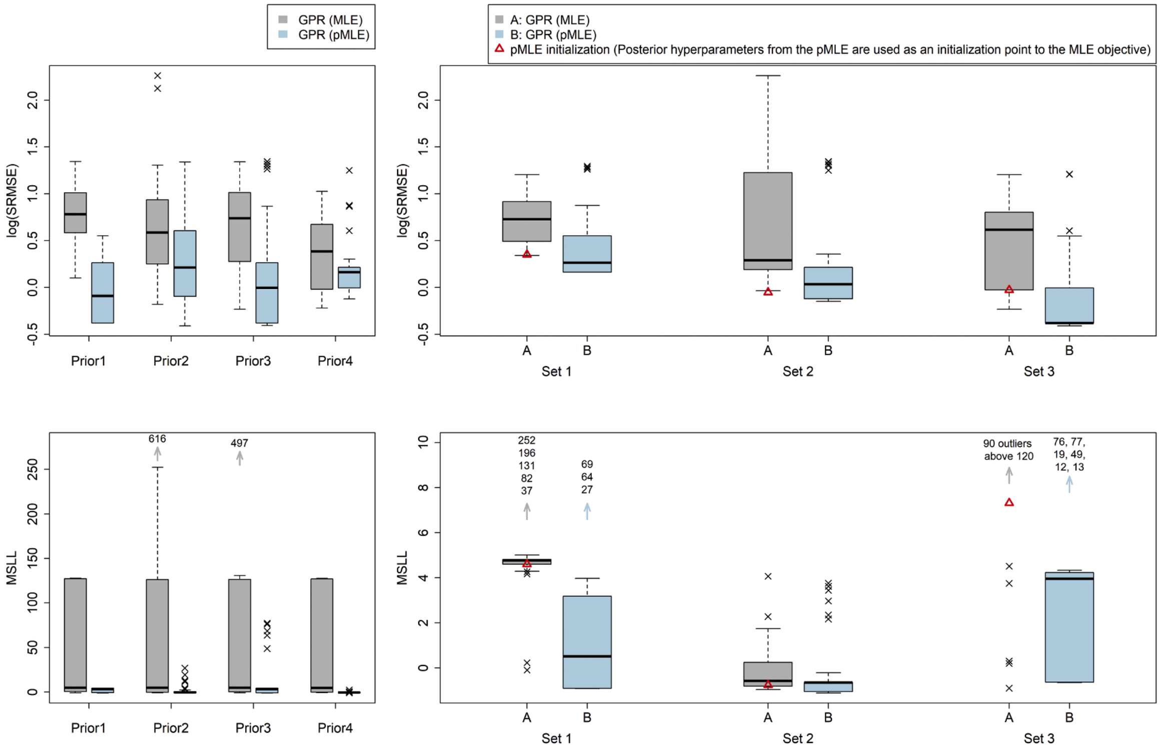

Fig. 8. *FOPD model: Left*: Boxplot of log(SRMSE) and MSLL for four different priors in Table 1. 25 initializations are performed for each prior in the optimization. Each boxplot shows the combined results of 3 different training set scenarios and contains 75 data points (25 × 3). *Right*: Boxplot of log(SRMSE) and MSLL for 3 different training set scenarios. Each boxplot shows the combined results of 4 different priors and contains 100 data points (25 × 4). The red triangle points show prediction error when the **best** posterior hyperparameters (i.e., posterior hyperparameters that produce the lowest SRMSE) from the pMLE are used as an initialization point for the optimization of the standard MLE objective function. *Left* and *Right Plots* show the boxplot of the same dataset but with a different axis, and the outliers outside of the y-range are presented as the numbers for the corresponding cases. The box plot results for different training datasets are provided in Table C in Appendix C with the MCMC prediction results.

Table 6
FOPD model: Average log(SRMSE), MSLL, SNLML, and SPV for three different hyperparameter estimation methods. (Brackets show the standard errors)

	log(SRMSE)	MSLL	SNLML	SPV
GPR (MLE)	0.601 (0.440)	4.47e1 (7.27e1)	-1.23 (0.999)	1.58e3 (1.78e4)
GPR (MCMC)	0.399 (0.662)	6.95 (1.13e1)	-1.13 (0.955)	2.39e1 (3.69e1)
GPR (pMLE)	0.141 (0.402)	2.12 (0.903e1)	0.083 (1.46)	2.89 (0.959)

Under steady-state conditions with only one spatial direction, the equation becomes:

$$u\frac{\partial\psi}{\partial x} = D_p\frac{\partial^2\psi}{\partial\phi^2} - \frac{\partial(\dot{\phi}\psi)}{\partial\phi} \tag{18}$$

where $\psi(x,\phi)$ is fiber orientation probability distribution, x is the position of the particle, ϕ is the projected angle of the fiber, $\dot{\phi}$ is the rotational angular velocity of fiber, D_p is the dispersion coefficient, and u is the fluid velocity. If the fiber suspension flow is considered in the headbox section (contracting channel) of the paper machine, $\dot{\phi}$ and u have the following relationships, which are derived from the continuity equation:

$$u(x) = \frac{u_0}{1-\left(1-\frac{1}{R}\right)\left(\frac{x}{L}\right)},\ \dot{\phi} = -\frac{L}{u_0}\frac{\partial u}{\partial x}sin(2\phi) \tag{19}$$

where L is the headbox length, R is the contraction ratio, u_0 is the inlet velocity. The dimensionless form with above relationships becomes

$$\overline{u}\frac{\partial\psi}{\partial\overline{x}} = \overline{D}_p\frac{\partial^2\psi}{\partial\phi^2} + R\frac{\partial(\sin(2\phi)\psi)}{\partial\phi}$$

$$= \overline{D}_p\frac{\partial^2\psi}{\partial\phi^2} + 2Rcos(2\phi)\left[\frac{\partial\psi}{\partial\phi} + \psi\right] \tag{20}$$

where $\overline{D}_p = \frac{LD_p}{u_0}$, $\overline{x} = x/L$ and $\overline{u} = \frac{u}{u_0}$. Eq. (20) describes the fiber orientation probability distribution of pulp suspensions in the headbox of a paper machine.

It is common that the fiber orientation is only and partially observed near the exit ($x/L = 0.97$) of the headbox and the fiber orientation is randomly distributed at the entrance (i.e., $\psi(x = 0) = 1/\pi$). We use 9 equidistant data at ($x = 0$) and a few experimental data (Zhang, 2001) near the exit ($x/L = 0.97$) of the headbox for training the model. Note that the additional homogeneous Neumann boundary conditions at two boundaries $\psi(\phi = -\pi/2)$ and $\psi(\phi = \pi/2)$ are also required (Olson et al., 2004) to solve Eq. (20) with the numerical discretization technique such as Streamline Upwind/Petrov-Galerkin method (SUPG) (Brooks & Hughes, 1982). However, if the observation data is noisy and the boundary conditions are not exact or unobtainable, the numerical discretization approach may result in an extremely ill-posed problem (Alifanov, 2012) where very small perturbation of boundary condition observation results in significant large errors in the numerical solution of PDEs. If this is the case, our approach can be an alternative direction to avoid large numerical errors while still capturing the underlying nature of the true solution.

A noise hyperparameter σ_n^2 (Eq. (1)) is introduced to capture the noise in the experimental data, which is estimated with the kernel hyperparameters during the (penalized) marginal likelihood estimation. Note that the starting point of the noise hyperparameter and the bounds during the optimization are important. If the noise hyperparameter is

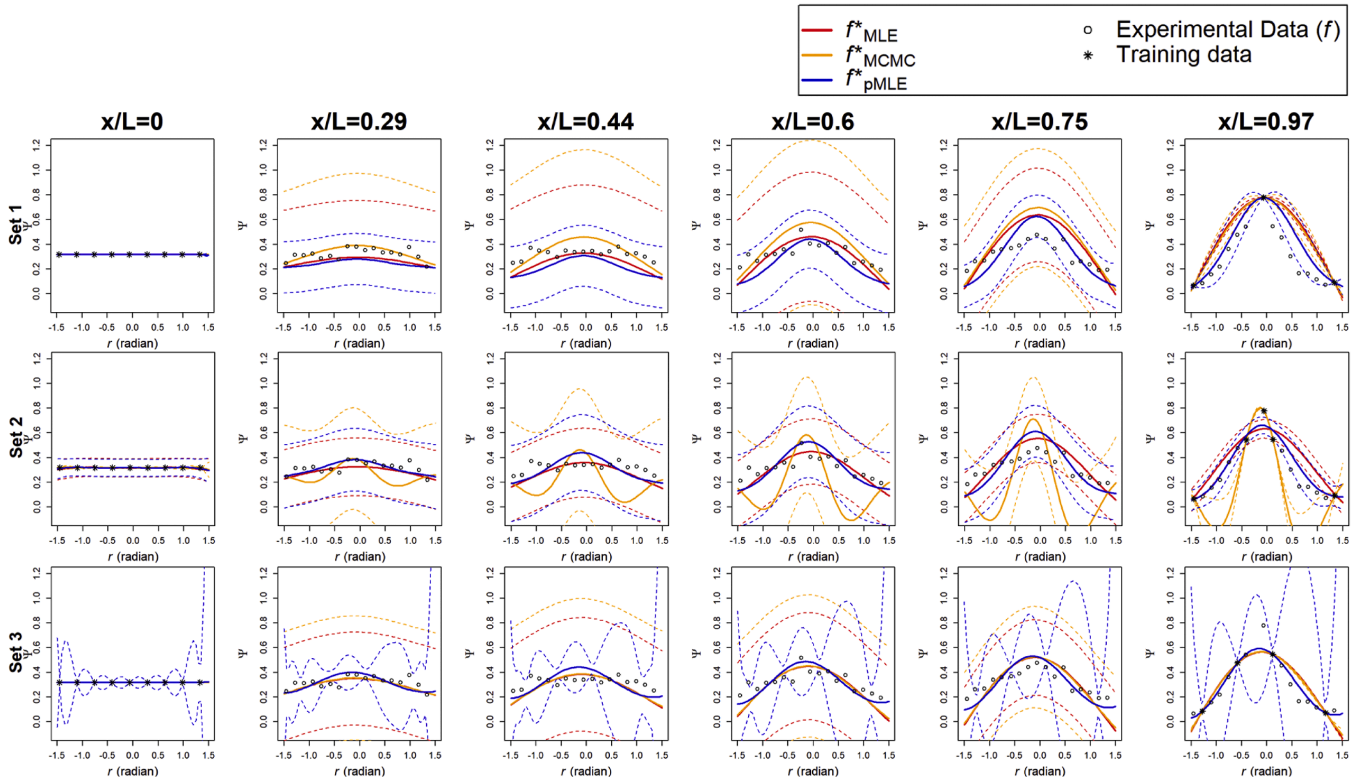

Fig. 9. *FOPD model*: Prediction performance of GPR with three different hyperparameter estimation methods. *Top, Middle*, and *Bottom Row* use three different training datasets. The fiber orientation probability distribution ψ at six different locations (x/L = 0, 0.29, 0.44, 0.6, 0.75, and 0.97) when R=10 and $\overline{D}_p$=2 are shown, where γ is the fiber orientation distribution in the plane of the paper and is calculated by substituting γ for ϕ in Eq. (20) and with the relationship $\dot{\gamma} = -\frac{1}{2}Rsin(2\gamma)$. The 68.2% confidence intervals (1 × σ) are plotted as a dotted line.

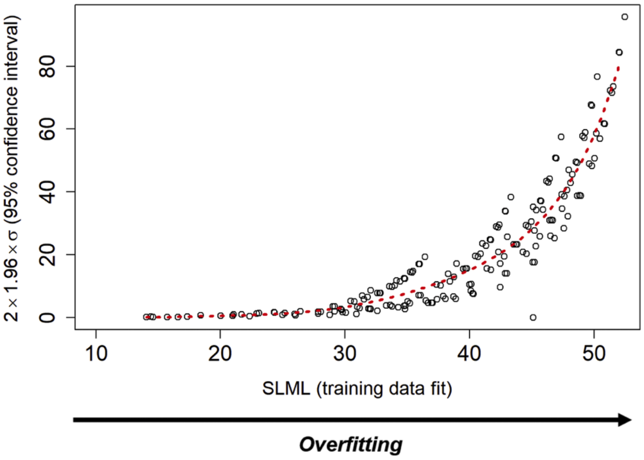

Fig. 10. *Heat equation*: Uncertainty and the SLML are measured at 216 different hyperparameter locations: $w_1 = w_2 = [0.10, 2.08, 4.06, 6.04, 8.02, 10.0]$, and $\tau = [0.10, 20.08, 40.06, 60.04, 80.02, 100.0]$. The average value of the uncertainty and the SLML at 200 test points are used for each hyperparameter set. Training set 5 in Fig. 6 is used for calculation.

Table 7
Heat equation: The Pearson correlation coefficient (Benesty et al., 2009) between SLML, SPV, and uncertainty (95% confidence interval) are measured at 216 different hyperparameter locations: $w_1 = w_2 = [0.10, 2.08, 4.06, 6.04, 8.02, 10.0]$, and $\tau = [0.10, 20.08, 40.06, 60.04, 80.02, 100.0]$. 10 different training datasets are sampled for SFD ~ [0.175,0.225], [0.275,0.325], [0.475,0.525], [0.775,0.825], respectively, and the average values are presented. Outliers are detected using the interquartile range (IQR) criterion and excluded from the calculation.

Pearson correlation coefficient	SLML	SPV	Uncertainty
SLML	-		
SPV	-0.086	-	
Uncertainty	**0.748**	0.273	-

not regulated properly, we may reach the local optima where data variability is mostly captured by noise. To avoid this issue, we set the upper bound of the noise hyperparameter as 1e^{-2} and set the prior (initialization) of it as 1e^{-5} for point estimation methods (MLE and pMLE). For MCMC analysis, we sample the noise hyperparameter from the Gamma distribution: $\sigma_n^2 \sim \mathit{Gamma}(0.05, 1)$ in each iteration. 1000 collocation points are used for the penalization, which are uniformly sampled on the system domain with *Maxpro* (Joseph et al., 2020).

Fig. 8 shows the boxplots of SRMSE and the MSLL for different priors and three training data scenarios. Results show that the physics-based penalization can help find local optima that produce a better prediction performance while mitigating the violation of physics (SPV in Table 6). It is interesting to see that a large number of outliers are present in MSLL when the model is trained with the observation data set 3 (90 outliers that produce MSLL larger than 120). This indicates that optimization with the MLE objective is prone to get stuck in a locally optimal region that produces large uncertainty for the four different priors experimented (Table 1). By penalizing the MLE objective with physics-based information, this region is no longer found and thus significantly improves the model performance.

According to all case studies (Table 3, 5, and 6) we investigated, the MLE hyperparameter estimation method has the lowest SNLML values, implying that the overfitting problems may be the most prominent. By embedding the physics-based knowledge into the optimization process, GPR with pMLE significantly mitigates the overfitting issue.

Fig. 9 shows the prediction performance of different hyperparameter estimation methods for three different training datasets. Note that the *bell-shaped nature* of the true system is captured with the physics-based penalization method (f^*_{pMLE}), which is not achievable with the standard data-driven approach (f^*_{MLE}, f^*_{MCMC}). This gives a promising outlook that the proposed approach (f^*_{pMLE}) can not only improve the performance over the data-driven methods (f^*_{MLE}, f^*_{MCMC}), but also discover the underlying nature of the true system, which was not discoverable before with the limited samples.

6. Discussion

6.1. Uncertainty reduction

It is interesting to see that a physics-based penalization approach can also reduce the uncertainty of the prediction in many cases. This implies that the GPR model captures the behavior of the system under a tighter uncertainty interval. While the assessment of the quality of the uncertainty is yet another open problem (Li et al., 2021), it is worth discussing why physics-based penalization can reduce the uncertainty. In GPR, it is known that variance (uncertainty) of the posterior predictive distribution has a closed-form representation (Eq. (1)) if a Gaussian likelihood is assumed. If a zero prior mean is used and other variables (e.g., type of kernels, training data used) are kept constant, the posterior predictive distribution and the reduction of the uncertainty in penalization are influenced by the posterior hyperparameters.

As shown in Table 3, 5, and 6, the physics-based penalization helps reduce overfitting problems in sparse data scenarios, as demonstrated by the higher SNLML (or lower SLML). If uncertainty increases in overfitting scenarios, mitigating overfitting problems with the physics-based penalization may help reduce the uncertainty as well. Under sparse data scenarios (Fig. 10), we observed a trend that the uncertainty increases as the SLML (training data-fit) is increased, implying that the overfitting can cause an increase in uncertainty.

Table 7 shows the Pearson correlation coefficient (Benesty et al., 2009) between SLML, SPV, and uncertainty. 10 different training set scenarios are considered for 4 different space-filling degrees (SFD = [0.175,0.225], [0.275,0.325], [0.475,0.525], [0.775,0.825]), respectively. The general trend shows that the uncertainty and the SLML have a high correlation (uncertainty increases as SLML is increased) under sparse data scenarios. While the data points in Fig. 10 do not necessarily represent local optima, it is expected that the local optima that have high SLML will likely have high uncertainty as well, which can be the reason why we observe the reduced uncertainty in the penalization approach.

6.2. Degree of extrapolation

While the physics-based penalization demonstrates a tuning effect for better prediction, it still suffers from extrapolation challenges (Appendix A). If training data is sampled in a limited space, the tuning effect may not be significant. Fig. 11 shows the interpolation and extrapolation prediction performance for the Laplace **equation** case study. The interpolation prediction error is calculated on the test points, which are located inside of the convex hull constructed from training data, and the remaining points outside of the convex hull are considered for calculating the extrapolation error.

It is observed that both the interpolation and extrapolation prediction error is lower in the penalization approach, while the performance improvement from the physics-based tuning increases as the data is more regularly sampled over the system domain (i.e., SFD is increased). In addition, physics-based penalization can help reduce the outliers for different initializations.

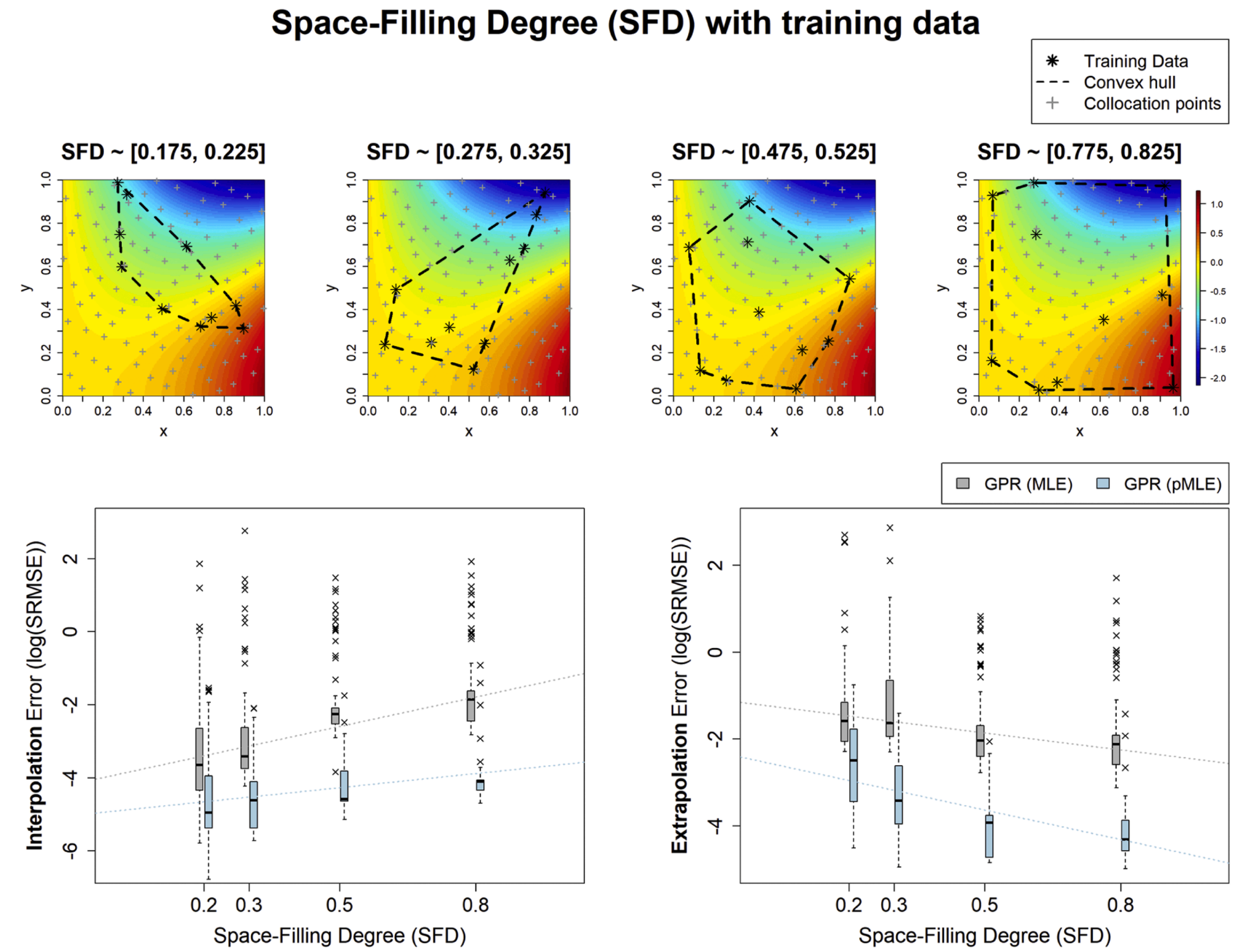

Fig. 11. *Laplace's equation: Top*: One example of sampled training data for different SFD values considered. *Bottom*: Interpolation and Extrapolation prediction error for GPRs with two different hyperparameter estimation methods (MLE and pMLE). 25 initializations are used for each prior in Table 1 and 10 different training dataset is tested (Each boxplot contains 1000 data points (25 × 4 × 10)). Here, the minimum distance condition $\min(\mathscr{D}_i, \mathscr{D}_j)$ in Eq. (13) is not considered.

There can be the case that the penalization still works effectively when SFD is low. Consider the case where the system dynamics are only abruptly changed in the centerline and dynamics are constant in the remaining region. If the modeler sets the constant GP prior mean that is close to the real dynamics (in the remaining region) and focuses on collecting the training data in the centerline, the physics-based penalization may still work better since the closed-form distribution of the GPR converges to the prior mean for the extrapolation region. In order for the physics-based tuning effect to be effective in different real applications, the training data should be sampled evenly on the system domain and capture the rough dynamics of the system.

6.3. *Computational complexity*

The computational cost for maximizing the marginal likelihood function is $O(N^3)$ (Rasmussen, 2003) where N is the number of training data and is dominated by inverting the kernel matrix $\left(\mathrm{K}+\sigma_n^2\mathrm{I}\right)^{-1}$. Traditionally, Cholesky decomposition is used for numerical stability and faster calculation (Rasmussen, 2003). Algorithm 3 and Algorithm 4 show the optimization procedure with the marginal likelihood and the penalized marginal likelihood using Cholesky Decomposition, respectively.

Penalized marginal likelihood function includes the additional estimation of $\mathrm{PV}(\theta, X^*_{col}|X, y)$ for each iteration of optimization, which adds additional complexity $O(N_{col}N^2/2)$. In the scarce dataset $N_{col} \gg N$, optimization of penalized marginal likelihood is dominated by $O(N_{col}N^2/2)$.

Fig. 12 shows the computational cost (for the hyperparameter estimation) and the prediction error (SRMSE) for the Laplace **equation** case study for a different number of training data and collocation points (Table is provided in Appendix D, E). Since the penalized approach requires additional estimation of $\mathrm{PV}(\theta, X^*_{col}|X, y)$ for each iteration of optimization, it yields additional computational time compared to the standard GPR (GPR.MLE) and the MCMC approach. However, this extra cost has the benefit of leading to improved prediction quality. Note that CPU results can be affected by algorithmic settings. For example, results are reported for fixed MCMC settings (i.e., 1000 MCMC draws), and the computational time will increase when the number of draws increases. Here, GPR with a full Bayesian treatment (GPR.MCMC) shows a larger error than the point estimation method with the standard marginal likelihood (GPR.MLE). This indicates the difficulty in tuning MCMC parameters and selecting appropriate priors.

Fig. 12 and Appendix D show that the small number of collocation points is still effective for embedding physics into GPR. In other words, a small number of collocation points may be sufficient to find a tuned local optimum. This may lead to an interesting discussion on an efficient sampling of collocation points to embed physics.

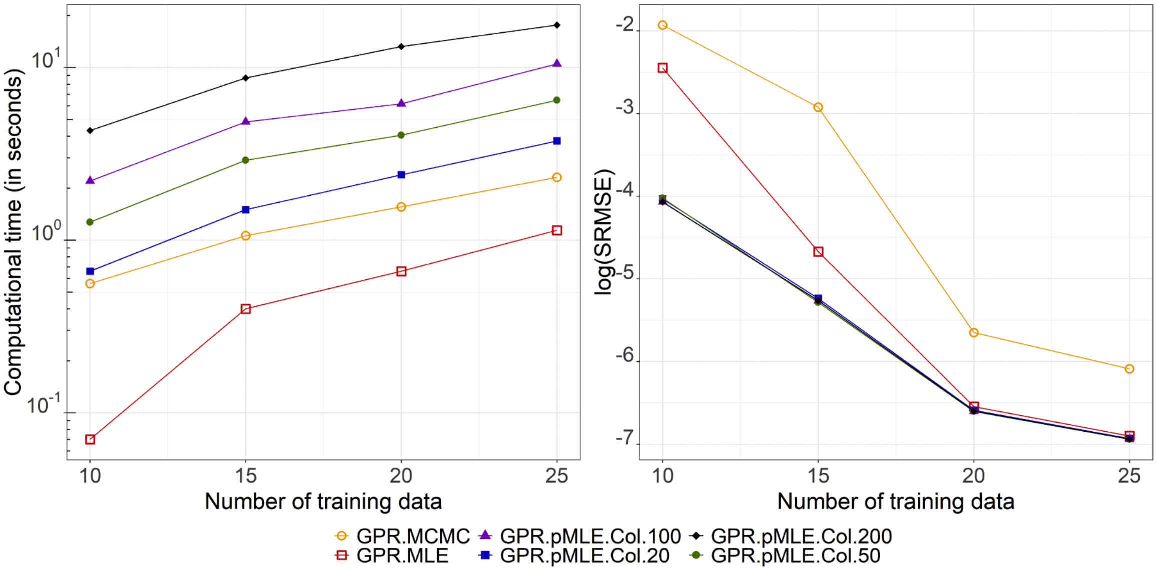

Fig. 12. *Laplace's equation: Left*: Average computational time for the **hyperparameter estimation step** (in seconds) of three different methods. GPR.MLE refers to the standard GPR where hyperparameters are estimated by optimizing the marginal likelihood function, GPR.MCMC is the standard GPR with full Bayesian analysis using the Metropolis-Hastings MCMC algorithm (Chib & Greenberg, 1995), and the pMLE.Col.20, pMLE.Col.50, pMLE.Col.100, and pMLE.Col.200 refers to the penalization method where hyperparameters are estimated by optimizing penalized marginal likelihood function (pMLE) with different collocation points $N_{col} = 20$, 50,100 and 200, respectively. 100 initializations are performed from Prior 4 for GPR.MLE and pMLE, and the average computational time is plotted. GPR.MCMC generates 1000 MCMC draws with Prior 4. Note that the computational time for GPR.MCMC depends on the number of MCMC draws. *Right*: Prediction error on the test points for three different methods. Predictions are estimated at 100 uniformly distributed points in each dimension (a total of 10000 prediction points). The **best** point estimation value (posterior hyperparameters that produce the lowest error) from 100 initializations are chosen for prediction. GPR.MCMC prediction is made with 100 Burn-in. Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz Processor is used.

7. Conclusions

We introduced physics-based penalization terms into the marginal likelihood function during hyperparameter estimation in Gaussian Process Regression (GPR). A series of results showed that physics-based penalization can improve prediction performance, reduce uncertainty, mitigate overfitting problems, and capture underlying physics that is not discoverable with the standard purely data-driven approach. As verified by p-values from two-way ANOVA analysis (Scheffe, 1999), physics-based penalization shows a meaningful tuning effect on the predictability of the GPR model.

The key observation is that the estimated posterior hyperparameters obtained from the penalized marginal likelihood function produce smaller physics violations (SPV) while sacrificing the data-fit (SLML). This is because the penalized marginal likelihood objective function balances the data-fit and the physics-fit during optimization. Based on the case studies of this paper, we have observed that this modification leads to more robust GPR hyperparameter tuning, which is more robust to overfitting and has improved predictability in slighlty extrapolated regions.

While physics-based penalization showed promising potential in sparse data scenarios in different case studies, it is not a universal tool that always produces a better result than the standard data-driven approach. Rather, it should be interpreted as a physics-based tool that a modeler can consider if a non-informative prior does not help, or multiple initializations in MLE fails to find helpful local optima.

We believe that this work will be helpful to systems where data acquisition is expensive and only a small set of samples is available, as the physics-based penalization can help reduce the bias and overfitting problems. In a broader context, this work can give modelers promising options to understand complex real-world problems and help extend our knowledge of systems, which is not achievable with only the data itself.

In this paper, we used the squared-exponential kernel to model the system's dynamics. However, dynamics from different stochastic partial differential equations may have non-stationary behavior, so GP with a stationary kernel may have a limitation in accurate approximation even under physics-based penalization. Moreover, if the modeler decides to use a non-Gaussian likelihood function, the exact inference of the marginal likelihood function may be intractable so that the approximation may be needed (Titsias et al., 2008). In this case, a transformation of the observation space can be applied (Snelson et al., 2003).

We also used the L2-norm squared of the physics violation function over the system domain to incorporate physics-based knowledge into GPR. If outliers are present, however, this loss function can overemphasize the effect of the outliers and bias the hyperparameter optimization process. This adds additional difficulty in finding the proper hyperparameters of the kernel when a noise hyperparameter is introduced. Therefore, it is important to set a good starting point and reasonable noise level bounds during optimization to prevent the optimization process from converging to the undesirable local optima (e.g., data variability is mostly captured by noise).

A modeler can adjust the important factors that affect the hyperparameter estimation process to improve the prediction performance of the proposed model. For example, one can test different kernels and optimization algorithms, adjust optimization parameters, increase the number of initializations (i.e., initial starting points) for optimization, and incorporate more collocation points. A modeler can also introduce an additional regularization parameter λ to the penalized MLE function (Eq. (12)) and perform cross-validation (CV) to fine-tune the balance between physics-fit and the data-fit. In order to improve the penalization effect, various physics-based penalty functions can be tested in the future. For example, the probabilistic formulation of the physics-based penalization (Lorenzi & Filippone, 2018; Tamuri et al., 2014) for hyperparameter tuning will be an interesting topic. Also, one can think of incorporating uncertainty information into the physics violation function or investigation of the effective sampling of collocation points (e.g., sampling the region where uncertainty is large).

Finally, a physics-based penalization approach can also give insight into physics-embedded experimental design or sequential sampling (e.g., Bayesian optimization) (Paulson & Lu, 2022). For example, an effective sampling strategy, such as sampling the next data point where the prediction improvement or the uncertainty reduction over standard GPR is the highest, would be an interesting topic for future research.

CRediT authorship contribution statement

Jinhyeun Kim: Conceptualization, Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing. **Christopher Luettgen:** Resources, Writing – review & editing. **Kamran Paynabar:** Supervision, Writing – review & editing, Funding acquisition, Methodology. **Fani Boukouvala:** Writing – review & editing, Supervision, Project administration, Funding acquisition, Methodology.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

The authors greatly acknowledge the Renewable Bioproducts Institute (RBI) at Georgia Tech, that awarded a graduate student fellowship to Jinhyeun Kim to conduct this research. Fani Boukouvala and Jinhyeun Kim were also supported by DOE RAPID/AICHE (Synopsis project) and NSF Career (CAREER – 1944678) and Kamran Paynabar by NSF CMMI-1839591.

Appendix A. Extrapolation challenges in Gaussian process regression

In standard GPR, Extrapolation performance in GPR is largely influenced by the kernel. This can be explained by Eq. (21) which represents GP posterior mean at test point X^* as a linear combination of N kernel functions (Rasmussen, 2003):

$$\widehat{m}_{posterior}\ (X^*) = m_{prior}(X^*) + \sum_{i=1}^{N} \alpha_i k\left(X^{(i)}, X^*\right) \tag{21}$$

where $\alpha_i = (\mathrm{K} + \sigma_n^2 \mathrm{I}\ )^{-1}\boldsymbol{y}$ and $k(X^{(i)},\ X^*) = cov(f(X^{(i)}),\ f(X^*))$. In standard GPR with squared exponential kernel $k(X^{(i)},\ X^*) = \tau^2 \exp\left(- \frac{1}{2}\sum_{q=1}^{p} w_q (x_q^{(i)} - x_q^*)^2 \right)$, the posterior mean $\widehat{m}_{posterior}\ (X^*)$ converges to the prior mean $m_{prior}(X^*)$ when the testing location is far from the training data range $|x_q^{(i)} - x_q^*|\uparrow$.

In penalized GPR, the physics violation function PV can also be represented as a linear transformation of kernel functions. With zero prior mean

$$\mathrm{PV}(\theta, X^*|\boldsymbol{X}, \boldsymbol{y}) = \| \mathscr{L}(f)_{mean} \|_2^2 = \| \mathscr{K}\left(\mathrm{K} + \sigma_n^2 \mathrm{I}\ \right)^{-1}\boldsymbol{y}\ \|_2^2\ = \| \sum_{i=1}^{N} \alpha_i k_{physics}\left(X^{(i)}, X^*\right) \|_2^2 \tag{22}$$

where $\mathscr{K} = (k_{x_2^*}^{*T} - k_{x_1^{*2}}^{*T}), \alpha_i = (\mathrm{K} + \sigma_n^2 \mathrm{I}\ )^{-1}\boldsymbol{y}$ and $k_{physics}(X^{(i)}, X^*) = [w_2(x_2^{(i)} - x_2^*) - w_1(w_1(x_1^{(i)} - x_1^*)^2 - 1)]k(X^{(i)}, X^*) = g(X^{(i)}, X^*)k(X^{(i)}, X^*)$. The performance improvement of physics-based penalized GPR over standard GPR will be reduced when extrapolation region is large since the effect of physics-based knowledge embedded in $g(X^{(i)}, X^*)$ is dissipated out by the effect of squared exponential kernel converging to the (zero) prior mean (e.g., $k(X^{(i)}, X^*) \rightarrow 0$).

Appendix B. Interpretation and Reformulation of Penalization in MLE Estimation

Marginal likelihood is a non-convex function, and different prior settings on the standard GPR can lead to different local optima. Therefore, proper prior settings are very important and can determine the performance of GPR. Introducing penalization term into marginal likelihood function can be interpreted as utilizing the physics-embedded prior. If we assume that likelihood follows Gaussian, the penalized negative log marginal likelihood (PNLML) under the Gaussian prior $f|\boldsymbol{X}, \theta\ \sim\ N(0, \mathrm{K})$ becomes

$$\begin{aligned} \mathrm{PNLML} &= logp(\boldsymbol{y}|\boldsymbol{X}, \theta) + \mathrm{PV}\left(\theta, \boldsymbol{X}_{col}^*|\boldsymbol{X}, \boldsymbol{y}\right) \\ &= -log\left(p(\boldsymbol{y}|\boldsymbol{X}, \theta)e^{-\mathrm{PV}\left(\theta, \boldsymbol{X}_{col}^*|\boldsymbol{X}, \boldsymbol{y}\right)}\right) \\ &= -log\left(\frac{1}{\left|\mathrm{K} + \sigma_n^2 \mathrm{I}\right|^{1/2}(2\pi)^{N/2}} e^{-\frac{1}{2}\boldsymbol{y}^T\left(\mathrm{K} + \sigma_n^2 \mathrm{I}\ \right)^{-1}\boldsymbol{y} - \mathrm{PV}\left(\theta, \boldsymbol{X}_{col}^*|\boldsymbol{X}, \boldsymbol{y}\right)}\right) \end{aligned} \tag{23}$$

Note that the marginal likelihood function is estimated by integrating out f, $log[p(\boldsymbol{y}|\boldsymbol{X}, \theta)] = log[\int p(\boldsymbol{y}|f, \boldsymbol{X}, \theta)p(f|\boldsymbol{X}, \theta)df] = -\frac{1}{2}log\left|\mathrm{K} + \sigma_n^2 \mathrm{I}\right| - \frac{1}{2}\boldsymbol{y}^T(\mathrm{K} + \sigma_n^2 \mathrm{I})^{-1}\boldsymbol{y} - \frac{N}{2}log2\pi$. The physics-violation function for the 5.2. Heat case study can be reformulated as

$$\mathrm{PV}(\theta, \boldsymbol{X}^*_{col}|\boldsymbol{X},\boldsymbol{y}) = \| (\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1})(\mathrm{K} + \sigma^2_n \mathrm{I}\ )^{-1}\boldsymbol{y} \|^2_2$$

$$= \left[\left(\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1}\right)\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\boldsymbol{y}\right]^T \left[\left(\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1}\right)\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\boldsymbol{y}\right]$$

$$= \boldsymbol{y}^{\boldsymbol{T}}\left(\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\right)^T \left(\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1}\right)^T \left(\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1}\right)\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\boldsymbol{y}$$

$$= \boldsymbol{y}^{\boldsymbol{T}}\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\mathscr{K}^*\left(\Sigma + \sigma^2_n \mathrm{I}\ \right)^{-1}\boldsymbol{y} \tag{24}$$

Where $\mathscr{K}^* = (\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1})^T_{N\times\ 1}(\boldsymbol{k}^{*T}_{x^*_2} - \boldsymbol{k}^{*T}_{x^{*2}_1})_{1\ \times N}$ denotes the $N\ \times N$ matrix, $\mathscr{K}^* = [\mathscr{K}^*_1, \mathscr{K}^*_2, \ldots, \mathscr{K}^*_{N_{col}}]$, $\mathscr{K}^*_i = [\mathscr{K}^*(X^{(1)}, X^{(i)}_{col}), \mathscr{K}^*(X^{(2)}, X^{(i)}_{col}), \ldots, \mathscr{K}^*(X^{(N)}, X^{(i)}_{col})]^T$, and $\mathscr{K}^*\ = k^{*T}_{x^*_2} - k^{*T}_{x^{*2}_1}$. Then penalized marginal likelihood becomes

$$\mathrm{PNLML} = -log\left(\frac{1}{\left|\mathrm{K} + \sigma^2_n \mathrm{I}\right|^{1/2}(2\pi)^{N/2}} e^{-\frac{1}{2}y^T\left[\left(\mathrm{K}+\sigma^2_n\mathrm{I}\ \right)^{-1}\left(I+2\mathscr{K}^*\left(\mathrm{K}+\sigma^2_n\mathrm{I}\ \right)^{-1}\right)\right]y}\right) \tag{25}$$

If we add $\psi = -log\left(\frac{|\mathrm{K}+\sigma^2_n\mathrm{I}|^{1/2}}{\left|[(\mathrm{K}+\sigma^2_n\mathrm{I}\ )^{-1}(I+2\mathscr{K}^*(\mathrm{K}+\sigma^2_n\mathrm{I}\ )^{-1})]^{-1}\right|^{1/2}}\right)$ to Eq. (25):

$$\mathrm{PNLML} = -log\left(\frac{1}{\left|\left[\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\left(I + 2\mathscr{K}^*\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\right)\right]^{-1}\right|^{1/2}(2\pi)^{N/2}} e^{-\frac{1}{2}y^T\left[\left(\mathrm{K}+\sigma^2_n\mathrm{I}\ \right)^{-1}\left(I+2\mathscr{K}^*\left(\mathrm{K}+\sigma^2_n\mathrm{I}\ \right)^{-1}\right)\right]y}\right) \tag{26}$$

If $[(\mathrm{K} + \sigma^2_n \mathrm{I}\ )^{-1}(I + 2\mathscr{K}^*(\mathrm{K} + \sigma^2_n \mathrm{I}\ )^{-1})]^{-1}$ is positive semi-definite, we can see that penalized marginal likelihood function can be interpreted as using the updated prior of $p(f|0, [(\mathrm{K} + \sigma^2_n \mathrm{I}\ )^{-1}(I + 2\mathscr{K}^*(\mathrm{K} + \sigma^2_n \mathrm{I}\ )^{-1})]^{-1})$ in standard GPR

$$p(\boldsymbol{y}|\boldsymbol{X}) = \int p(\boldsymbol{y}|f, \boldsymbol{X}, \theta)p(f|\boldsymbol{X}, \theta)df$$

$$= \int N(\boldsymbol{y}|f, 0_{N\times N})p\left(f|0, \left[\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\left(I + 2\mathscr{K}^*\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\right)\right]^{-1}\right)df$$

$$= N\left(\boldsymbol{y}|0, \left[\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\left(I + 2\mathscr{K}^*\left(\mathrm{K} + \sigma^2_n \mathrm{I}\ \right)^{-1}\right)\right]^{-1}\right) \tag{27}$$

The updated prior is constructed with matrix $\mathscr{K}^*$ which is populated from physics-based knowledge, so it can be interpreted as a physics-embedded prior.

Appendix C. Prediction performance for different hyperparameter estimation processes

Table C
The log(SRMSE) and the MSLL for different training dataset scenarios for three case studies. The median values are reported.

Case Study	Different training dataset scenarios	log(SRMSE)			MSLL		
		MLE	pMLE	MCMC	MLE	pMLE	MCMC
Laplace	Set 1	-2.28	-3.92	-1.54	-1.42	-3.68	-1.12
	Set 2	-2.47	-4.07	-1.99	-2.35	-3.55	-1.18
	Set 3	-2.44	-4.63	-2.19	-1.94	-3.98	-1.11
	Set 4	-2.49	-3.98	-1.82	-1.19	-3.45	-1.26
	Set 5	-2.80	-3.30	-2.10	-2.05	-3.30	-1.10
	Set 6	-2.54	-4.52	-1.99	-1.78	-3.96	-1.31
	Set 7	-1.72	-4.21	-1.87	-1.43	-4.07	-1.27
	Set 8	-2.35	-4.49	-1.84	-1.36	-3.88	-1.18
	Set 9	-1.65	-4.38	-2.08	-0.60	-3.08	-0.84
	Set 10	-2.78	-4.46	-2.47	-2.03	-3.42	-1.13
Heat	Set 1	-0.94	-0.96	-0.75	0.45	-0.22	-1.11
	Set 2	-0.91	-1.04	-0.83	-1.37	-1.45	-1.18
	Set 3	-1.34	-1.36	-1.43	-1.22	-0.55	-1.44
	Set 4	-1.11	-1.28	-1.08	-1.13	-1.84	-1.46
	Set 5	-0.62	-1.83	-0.80	0.08	-1.21	-1.03
	Set 6	-0.96	-1.02	-1.18	0.36	0.05	-1.29
	Set 7	-1.25	-1.61	-1.29	-0.96	-1.10	-1.19
	Set 8	-0.29	-1.05	-1.12	-0.23	-1.18	-1.18
	Set 9	-0.90	-1.26	-0.91	-0.31	-1.14	-0.93
	Set 10	-0.62	-0.67	-0.91	-0.32	-0.84	-0.94
FOPD	Set 1	0.73	0.27	0.39	4.77	0.52	0.60
	Set 2	0.29	0.04	1.06	-0.57	-0.64	0.21
	Set 3	0.62	-0.38	-0.26	127.21	3.96	20.03

Appendix D. Effect of collocation points

Table D
The lowest SRMSEs ($\times 10^3$) achieved by three methods for the different number of training data (N_{data}) and collocation points (N_{col}). Training data is sampled under criteria $\Omega_{\mathscr{D},\ N_{data}=10} = [\mathscr{D} : \text{SFD} \geq 0.75,\ \min(\mathscr{D}_i, \mathscr{D}_j) \geq 0.2]$, $\Omega_{\mathscr{D},\ N_{data}=15} = [\mathscr{D} : \text{SFD} \geq 0.8,\ \min(\mathscr{D}_i, \mathscr{D}_j) \geq 0.17]$, $\Omega_{\mathscr{D},\ N_{data}=20} = [\mathscr{D} : \text{SFD} \geq 0.8,\ \min(\mathscr{D}_i, \mathscr{D}_j) \geq 0.13]$, and $\Omega_{\mathscr{D},\ N_{data}=25} = [\mathscr{D} : \text{SFD} \geq 0.8,\ \min(\mathscr{D}_i, \mathscr{D}_j) \geq 0.1]$. Collocation points are independently sampled for each case using *Maxpro* design (Joseph et al., 2020). Prior 4 in Table 1 is used for all methods. 100 initializations are performed from Prior 4 for the point estimation methods (GPR (MLE) and GPR (pMLE)) while 1000 MCMC draws with Prior 4 (and 100 Burn-in) is used for GPR (MCMC).

Case Study	N_{data}	GPR (MLE)	GPR (MCMC)	GPR (pMLE)			
				$N_{col} = 20$	$N_{col} = 50$	$N_{col} = 100$	$N_{col} = 200$
Laplace	10	86.5	145	17.7	16.9	17.1	17.1
	15	9.37	53.7	5.31	5.07	5.17	5.19
	20	1.44	3.52	1.37	1.35	1.36	1.36
	25	1.01	2.27	0.977	0.966	0.970	0.969
Heat	10	234	250	212	209	212	210
	15	153	160	107	123	114	119
	20	89.2	80.5	77.0	77.6	76.9	77.4
	25	32.1	30.3	25.6	25.6	25.4	25.3

Appendix E. Computational cost

Computational time is recorded for different hyperparameter estimation processes. In the high-dimensional space and a large number of training data, it is expected that the point estimation method (i.e., optimization of (penalized) marginal likelihood) will become more efficient if collocation points are effectively selected, as the computational time rapidly goes up for full Bayesian analysis (GPR-MCMC).

Table E
Computational time (in seconds) for maximizing the marginal likelihood (GPR-MLE), penalized marginal likelihood with a different number of collocation points (GPR (pMLE)), and using full Bayesian treatment with Metropolis-Hastings MCMC algorithm (Chib & Greenberg, 1995) (GPR (MCMC)). Note that the computational cost is presented for an average computational time of 100 initializations from Prior 4 for GPR (MLE) and GPR (pMLE). The computational cost for GPR (MCMC) is presented for generating 1000 MCMC draws with Prior 4. Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz Processor is used.

Case Study	N_{data}	GPR (MLE)	GPR (MCMC)	GPR (pMLE)			
				$N_{col} = 20$	$N_{col} = 50$	$N_{col} = 100$	$N_{col} = 200$
Laplace	10	0.07	0.56	0.66	1.27	2.20	4.31
	15	0.40	1.06	1.50	2.90	4.84	8.69
	20	0.66	1.56	2.39	4.05	6.17	13.2
	25	1.14	2.31	3.75	6.47	10.5	17.6
Heat	10	0.09	0.55	0.49	0.94	1.71	3.34
	15	0.13	0.78	0.67	1.27	2.30	4.25
	20	0.25	1.11	0.87	1.64	2.90	5.37
	25	0.27	1.51	1.27	2.28	3.58	6.94

Appendix F. Algorithms

Algorithm 1
Standard Gaussian Process Regression

```
Input: Observation set $(X, y)$, GP hyperparameters $\theta$
  1: Gaussian Prior: $f|X,\theta \sim N(0, K)$
  2: Maximum (marginal) Likelihood Estimation: $\theta^* \leftarrow \underset{\theta}{\text{argmin}}[-\log p(y|X,\theta)]$
Output: Posterior predictive distribution: $f|X,X^*,\ y,\theta^* \sim N(k^{*T}(K+\sigma_n^2 I)^{-1}y,\ k^{**} - k^{*T}(K+\sigma_n^2 I)^{-1}k^*)$
```

Algorithm 2
Gaussian Process Regression with PNLML

```
Input: Observation dataset $(X,y)$, GP hyperparameters $\theta$, number of observation dataset $N$, number of collocation points $N_{col}$
  1: Gaussian Prior: $f|X,\theta \sim N(0,\mathrm{K})$
  2: Formulation of Physics Violation (PV) function: $PV(\theta,X^*_{col}|X,y) = \| \mathscr{K}^*(\mathrm{K}+\sigma_n^2\mathrm{I})^{-1}y \|_2^2$
  3: Maximum (marginal) Likelihood Estimation with PNLML:
$\theta^*_{physics} \leftarrow \underset{\theta}{\operatorname{argmin}}\left[-\frac{1}{N}logp(y|X,\theta)+\frac{1}{N_{col}}PV(\theta,X^*_{col}|X,y)\right]$
Output: Posterior predictive distribution: $f|X,X^*,\ y,\theta^*_{physics} \sim N(k^{*T}(\mathrm{K}+\sigma_n^2\mathrm{I})^{-1}y,\ k^{**}-k^{*T}(\mathrm{K}+\sigma_n^2\mathrm{I})^{-1}k^*)$
```

Algorithm 3
Optimization of Marginal Likelihood using Cholesky Decomposition

```
Input: Observation dataset $(X,y)$, kernel function $k$, kernel matrix $\mathrm{K}$, collocation points $X_{col}$, number of observation dataset $N$, GP hyperparameters $\theta$
  1: $\mathrm{L} = \mathrm{cholesky}(\mathrm{K}+\sigma_n^2 I)$
  2: $\mathfrak{v} = L^T\backslash(L\backslash y)$
  3: $logp(y|X,\theta) = -\frac{1}{2}y^T\mathfrak{v} - \sum_{i=1}^N logL_{ii} - \frac{N}{2}log2\pi$
Output: $\theta^* = \underset{\theta}{\operatorname{argmin}} \log[-p(y|X,\theta)]$
```

* Computational Cost is $O(N^3/6)$ for line 1 and $O(N^2/2)$ for line 2

Algorithm 4
Optimization of Penalized Marginal Likelihood using Cholesky Decomposition

```
Input: Observation dataset $(X,y)$, kernel function $k$, kernel matrix $\mathrm{K}$, noise $\sigma_n^2$, collocation points $X_{col}$, physics-based linearly transformed Kernel $\mathscr{K}$, number of observation dataset $N$,
  number of collocation points $N_{col}$
  1: $\mathrm{L} = \mathrm{cholesky}(\mathrm{K}+\sigma_n^2 I)$
  2: $\mathfrak{v} = L^T\backslash(L\backslash y)$
  3: $logp(y|X,\theta) = -\frac{1}{2}y^T\mathfrak{v} - \sum_{i=1}^N logL_{ii} - \frac{N}{2}log2\pi$
  4: $\mathrm{PV}(\theta,X^*_{col}|X,y) = \| \mathscr{K}^{*T}\mathfrak{v} \|_2^2$ where $\mathscr{K}^* = [\mathscr{K}^*_1,\mathscr{K}^*_2,\ldots,\mathscr{K}^*_{N_{col}}]$, $\mathscr{K}^*_i = [\mathscr{K}^*(X^{(1)},X^{(i)}_{col}),\mathscr{K}^*(X^{(2)},X^{(i)}_{col}),\ldots,\mathscr{K}^*(X^{(N)},X^{(i)}_{col})]^T$
Output: $\theta^*_{physics} = \underset{\theta}{\operatorname{argmin}}\left[-\frac{1}{N}logp(y|X,\theta)+\frac{1}{N_{col}}\mathrm{PV}(\theta,X^*_{col}|X,y)\right]$
```

* Computational Cost is $O(N^3/6)$ for line 1, $O(N^2/2)$ for line 2, and $O(N_{col}N^2/2)$ for line 4

References

Ahmad, M., Karimi, I.A., 2021. Revised learning based evolutionary assistive paradigm for surrogate selection (LEAPS2v2). Computers & Chemical Engineering 152, 107385.

Albert, C.G., Rath, K., 2020. Gaussian Process Regression for Data Fulfilling Linear Differential Equations with Localized Sources. Entropy 22, 152.

Alifanov, O.M., 2012. Inverse heat transfer problems. Springer Science & Business Media.

Alves, V., Gazzaneo, V., Lima, F.V., 2022. A machine learning-based process operability framework using Gaussian processes. Computers & Chemical Engineering 163, 107835.

Bachoc, F., Lagnoux, A., López-Lopera, A.F., 2019. Maximum likelihood estimation for Gaussian processes under inequality constraints. Electronic Journal of Statistics 13, 2921–2969.

Bayarri, M.J., Berger, J.O., Paulo, R., Sacks, J., Cafeo, J.A., Cavendish, J., Lin, C.-H., Tu, J., 2007. A Framework for Validation of Computer Models. Technometrics 49, 138–154.

Baydin, A.G., Pearlmutter, B.A., Radul, A.A., Siskind, J.M., 2018. Automatic differentiation in machine learning: a survey. Journal of Marchine Learning Research 18, 1–43.

Benesty, J., Chen, J., Huang, Y., Cohen, I., 2009. Pearson correlation coefficient. Noise reduction in speech processing. Springer, pp. 1–4.

Berkenkamp, F., Schoellig, A.P., 2015. Safe and robust learning control with Gaussian processes. In: 2015 European Control Conference (ECC). IEEE, pp. 2496–2501.

Blum, M., Riedmiller, M.A., 2013. Optimization of Gaussian process hyperparameters using Rprop. In: ESANN. Citeseer, pp. 339–344.

Bonzanini, A.D., Paulson, J.A., Makrygiorgos, G., Mesbah, A., 2021. Fast approximate learning-based multistage nonlinear model predictive control using Gaussian processes and deep neural networks. Computers & Chemical Engineering 145, 107174.

Boukouvala, F., Ierapetritou, M.G., 2012. Feasibility analysis of black-box processes using an adaptive sampling Kriging-based method. Computers & Chemical Engineering 36, 358–368.

Bradley, W., Kim, J., Kilwein, Z., Blakely, L., Eydenberg, M., Jalvin, J., Laird, C., Boukouvala, F., 2022. Perspectives on the integration between first-principles and data-driven modeling. Computers & Chemical Engineering, 107898.

Brooks, A.N., Hughes, T.J., 1982. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. Computer Methods in Applied Mechanics and Engineering 32, 199–259.

Cao, Y., Brubaker, M.A., Fleet, D.J., Hertzmann, A., 2013. Efficient optimization for sparse Gaussian process regression. Advances in Neural Information Processing Systems 26.

Chen, W., Xiong, Y., Tsui, K.-L., & Wang, S. (2008). A design-driven validation approach using Bayesian prediction models.

Chen, Z., Wang, B., 2018. How priors of initial hyperparameters affect Gaussian process regression models. Neurocomputing 275, 1702–1710.

Chib, S., Greenberg, E., 1995. Understanding the metropolis-hastings algorithm. The american statistician 49, 327–335.

Christov, I.C., 2013. On a difficulty in the formulation of initial and boundary conditions for eigenfunction expansion solutions for the start-up of fluid flow. Mechanics Research Communications 51, 86–92.

Ciuperca, G., Ridolfi, A., Idier, J., 2003. Penalized maximum likelihood estimator for normal mixtures. Scandinavian Journal of Statistics 30, 45–59.

Cole, S.R., Chu, H., Greenland, S., 2014. Maximum likelihood, profile likelihood, and penalized likelihood: a primer. American journal of epidemiology 179, 252–260.

Coles, S.G., Dixon, M.J., 1999. Likelihood-based inference for extreme value models. Extremes 2, 5–23.

Constantinescu, E.M., Anitescu, M., 2013. Physics-based covariance models for Gaussian processes with multiple outputs. International Journal for Uncertainty Quantification 3.

Da Veiga, S., Marrel, A., 2012. Gaussian process modeling with inequality constraints. In: Annales de la Faculté des sciences de Toulouse: Mathématiques, 21, pp. 529–555.

Dai, W., Mohammadi, S., Cremaschi, S., 2022. A hybrid modeling framework using dimensional analysis for erosion predictions. Computers & Chemical Engineering 156, 107577.

Damianou, A., Lawrence, N.D., 2013. Deep gaussian processes. Artificial intelligence and statistics. PMLR, pp. 207–215.

Davis, E., Ierapetritou, M., 2007. A kriging method for the solution of nonlinear programs with black-box functions. AIChE Journal 53, 2001–2012.

Duvenaud, D.K., Nickisch, H., Rasmussen, C., 2011. Additive gaussian processes. Advances in Neural Information Processing Systems 24.

Eugene, E.A., Gao, X., Dowling, A.W., 2020. Learning and optimization with Bayesian hybrid models. In: 2020 American Control Conference (ACC). IEEE, pp. 3997–4002.

Firth, D., 1993. Bias reduction of maximum likelihood estimates. Biometrika 80, 27–38.

Fischer, B., Gorbach, N., Bauer, S., Bian, Y. A., & Buhmann, J. (2016). Model Selection for Gaussian Process Regression by Approximation Set Coding.

Flyer, N., Fornberg, B., 2003. Accurate numerical resolution of transients in initial-boundary value problems for the heat equation. Journal of Computational Physics 184, 526–539.

Fornberg, B., Flyer, N., 2004. On the nature of initial-boundary value solutions for dispersive equations. SIAM Journal on Applied Mathematics 64, 546–564.

Geyer, C. J., & Johnson, L. T. (2013). Mcmc: Markov chain monte carlo. In: R package version 0.9-2, URL http://CRAN.R-project.org/package=mcmc.

Golchi, S., Bingham, D.R., Chipman, H., Campbell, D.A., 2015. Monotone emulation of computer experiments. SIAM/ASA Journal on Uncertainty Quantification 3, 370–392.

Graepel, T., 2003. Solving noisy linear operator equations by Gaussian processes: Application to ordinary and partial differential equations. In: ICML, 3, pp. 234–241.

Grbić, R., Kurtagić, D., Slišković, D., 2013. Stream water temperature prediction based on Gaussian process regression. Expert systems with applications 40, 7407–7414.

Greenland, S., Schwartzbaum, J.A., Finkle, W.D., 2000. Problems due to small samples and sparse data in conditional logistic regression analysis. American journal of epidemiology 151, 531–539.

Gulian, M., Frankel, A., Swiler, L., 2022. Gaussian process regression constrained by boundary value problems. Computer Methods in Applied Mechanics and Engineering 388, 114117.

Gustafsson, O., Villani, M., & Stockhammar, P. (2020). Bayesian Optimization of Hyperparameters when the Marginal Likelihood is Estimated by MCMC. arXiv preprint arXiv:2004.10092.

Jain, A., Nghiem, T., Morari, M., Mangharam, R., 2018. Learning and control using Gaussian processes. In: 2018 ACM/IEEE 9th international conference on cyber-physical systems (ICCPS). IEEE, pp. 140–149.

Jensen, B.S., Nielsen, J.B., Larsen, J., 2013. Bounded gaussian process regression. In: 2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP). IEEE, pp. 1–6.

Jidling, C., Wahlström, N., Wills, A., & Schön, T. (2017a). Linearly constrained Gaussian processes.

Jidling, C., Wahlström, N., Wills, A., Schön, T.B., 2017b. Linearly constrained Gaussian processes. Advances in Neural Information Processing Systems 30.

Joseph, V.R., Gul, E., Ba, S., 2020. Designing computer experiments with multiple types of factors: The MaxPro approach. Journal of Quality Technology 52, 343–354.

Karvonen, T., & Oates, C. J. (2022). Maximum likelihood estimation in Gaussian process regression is ill-posed. arXiv preprint arXiv:2203.09179.

Kennedy, M.C., O'Hagan, A, 2001. Bayesian calibration of computer models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63, 425–464.

Kevrekidis, P., Williams, M., Mantzavinos, D., Charalampidis, E., Choi, M., Kevrekidis, I., 2017. REVISITING DIFFUSION. Quarterly of Applied Mathematics 75, 581–598.

Kim, J., & Choi, S. (2019). Practical Bayesian Optimization with Threshold-Guided Marginal Likelihood Maximization. arXiv preprint arXiv:1905.07540.

Kim, M., Cho, S., Han, A., Han, Y., Kwon, J.S.-I., Na, J., Moon, I., 2022. Multi-Objective Bayesian Optimization for Design and Operating of Fluidized Bed Reactor. In: Computer Aided Chemical Engineering, 49. Elsevier, pp. 1297–1302.

Kocijan, J., Murray-Smith, R., Rasmussen, C.E., Likar, B., 2003. Predictive control with Gaussian process models. In: The IEEE Region 8 EUROCON 2003. Computer as a Tool, 1. IEEE, pp. 352–356.

Kong, D., Chen, Y., Li, N., 2018. Gaussian process regression for tool wear prediction. Mechanical systems and signal processing 104, 556–574.

Lange-Hegermann, M., 2018. Algorithmic linearly constrained Gaussian processes. Advances in Neural Information Processing Systems 31.

Lange-Hegermann, M., 2021. Linearly Constrained Gaussian Processes with Boundary Conditions. In: Arindam, B., Kenji, F. (Eds.), Proceedings of The 24th International Conference on Artificial Intelligence and Statistics, 130. PMLR, pp. 1090–1098. Proceedings of Machine Learning Research.

Li, Y., Rao, S., Hassaine, A., Ramakrishnan, R., Canoy, D., Salimi-Khorshidi, G., Mamouei, M., Lukasiewicz, T., Rahimi, K., 2021. Deep Bayesian Gaussian processes for uncertainty estimation in electronic health records. Scientific reports 11, 1–13.

Liang, S., Jiang, S. W., Harlim, J., & Yang, H. (2021). Solving pdes on unknown manifolds with machine learning. arXiv preprint arXiv:2106.06682.

Liu, F., Bayarri, M., Berger, J., 2009. Modularization in Bayesian analysis, with emphasis on analysis of computer models. Bayesian Analysis 4.

López-Lopera, A.F., Bachoc, F., Durrande, N., Roustant, O., 2018. Finite-dimensional Gaussian approximation with linear inequality constraints. SIAM/ASA Journal on Uncertainty Quantification 6, 1224–1255.

Lorenzi, M., Filippone, M., 2018. Constraining the dynamics of deep probabilistic models. In: International Conference on Machine Learning. PMLR, pp. 3227–3236.

Maatouk, H., Bay, X., 2017. Gaussian process emulators for computer experiments with inequality constraints. Mathematical Geosciences 49, 557–582.

Manzhos, S., & Ihara, M. (2021). Rectangularization of Gaussian process regression for optimization of hyperparameters. arXiv preprint arXiv:2112.02467.

Mattos, C.L.C., Barreto, G.A., 2019. A stochastic variational framework for recurrent gaussian processes models. Neural Networks 112, 54–72.

Mohammed, R.O., Cawley, G.C., 2017. Over-fitting in model selection with Gaussian process regression. In: International Conference on Machine Learning and Data Mining in Pattern Recognition. Springer, pp. 192–205.

Morris, M.D., Mitchell, T.J., Ylvisaker, D., 1993. Bayesian design and analysis of computer experiments: use of derivatives in surface prediction. Technometrics 35, 243–255.

Nevin, J.W., Vaquero-Caballero, F., Ives, D.J., Savory, S.J., 2021. Physics-informed Gaussian process regression for optical fiber communication systems. Journal of Lightwave Technology 39, 6833–6844.

Ng, T.L.J., 2022. Penalized maximum likelihood estimator for mixture of von Mises–Fisher distributions. Metrika 1–23.

Olofsson, S., Deisenroth, M.P., Misener, R., 2018. Design of experiments for model discrimination using Gaussian process surrogate models. In: Computer Aided Chemical Engineering, 44. Elsevier, pp. 847–852.

Olofsson, S., Schultz, E. S., Mhamdi, A., Mitsos, A., Deisenroth, M. P., & Misener, R. (2021). Using Gaussian Processes to Design Dynamic Experiments for Black-Box Model Discrimination under Uncertainty. arXiv preprint arXiv:2102.03782.

Olson, J.A., Frigaard, I., Chan, C., Hämäläinen, J.P., 2004. Modeling a turbulent fibre suspension flowing in a planar contraction: The one-dimensional headbox. International Journal of Multiphase Flow 30, 51–66.

Pahari, S., Moon, J., Akbulut, M., Hwang, S., Kwon, J.S.-I., 2021. Estimation of microstructural properties of wormlike micelles via a multi-scale multi-recommendation batch bayesian optimization. Industrial & Engineering Chemistry Research 60, 15669–15678.

Papukdee, N., Park, J.-S., Busababodhin, P., 2022. Penalized likelihood approach for the four-parameter kappa distribution. Journal of Applied Statistics 49, 1559–1573.

Paulson, J.A., Lu, C., 2022. COBALT: COnstrained Bayesian optimizAtion of computationaLly expensive grey-box models exploiting derivaTive information. Computers & Chemical Engineering 160, 107700.

Pensoneault, A., Yang, X., Zhu, X., 2020. Nonnegativity-enforced Gaussian process regression. Theoretical and Applied Mechanics Letters 10, 182–187.

Petsagkourakis, P., Galvanin, F., 2021. Safe model-based design of experiments using Gaussian processes. Computers & Chemical Engineering 151, 107339.

Quirante, N., Javaloyes, J., Ruiz-Femenia, R., Caballero, J.A., 2015. Optimization of chemical processes using surrogate models based on a Kriging interpolation. In: Computer Aided Chemical Engineering, 37. Elsevier, pp. 179–184.

Rai, P.K., Tripathi, S., 2019. Gaussian process for estimating parameters of partial differential equations and its application to the Richards equation. Stochastic Environmental Research and Risk Assessment 33, 1629–1649.

Raissi, M., Karniadakis, G.E., 2018. Hidden physics models: Machine learning of nonlinear partial differential equations. Journal of Computational Physics 357, 125–141.

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2017. Machine learning of linear differential equations using Gaussian processes. Journal of Computational Physics 348, 683–693.

Raissi, M., Perdikaris, P., Karniadakis, G.E., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics 378, 686–707.

Rasmussen, C.E., 2003. Gaussian processes in machine learning. Summer school on machine learning. Springer, pp. 63–71.

Riihimäki, J., Vehtari, A., 2010. Gaussian processes with monotonicity information. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics, pp. 645–652. JMLR Workshop and Conference Proceedings.

Risken, H., 1996. Fokker-planck equation. The Fokker-Planck Equation. Springer, pp. 63–95.

Roberts, S., Osborne, M., Ebden, M., Reece, S., Gibson, N., Aigrain, S., 2013. Gaussian processes for time-series modelling. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 371, 20110550.

Särkkä, S., 2011. Linear Operators and Stochastic Partial Differential Equations in Gaussian Process Regression. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 151–158.

Scheffe, H., 1999. The analysis of variance, 72. John Wiley & Sons.

Schweidtmann, A.M., Bongartz, D., Grothe, D., Kerkenhoff, T., Lin, X., Najman, J., Mitsos, A., 2021. Deterministic global optimization with Gaussian processes embedded. Mathematical Programming Computation 13, 553–581.

Snelson, E., Ghahramani, Z., Rasmussen, C., 2003. Warped gaussian processes. Advances in Neural Information Processing Systems 16.

Solak, E., Murray-Smith, R., Leithead, W., Leith, D., Rasmussen, C., 2002. Derivative observations in Gaussian process models of dynamic systems. Advances in Neural Information Processing Systems 15.

Swiler, L.P., Gulian, M., Frankel, A.L., Safta, C., Jakeman, J.D., 2020. A survey of constrained Gaussian process regression: Approaches and implementation challenges. Journal of Machine Learning for Modeling and Computing 1.

Tamuri, A.U., Goldman, N., dos Reis, M., 2014. A penalized-likelihood method to estimate the distribution of selection coefficients from phylogenetic data. Genetics 197, 257–271.

Titsias, M.K., Lawrence, N., Rattray, M., 2008. Markov chain Monte Carlo algorithms for Gaussian processes. Inference and Estimation in Probabilistic. Time-Series Models 9, 298.

Vessella, S., 2015. Stability Estimates for an Inverse Hyperbolic Initial Boundary Value Problem with Unknown Boundaries. SIAM J. Math. Anal. 47, 1419–1457.

Wang, H., Zhou, X., 2021. Explicit estimation of derivatives from data and differential equations by gaussian process regression. International Journal for Uncertainty Quantification 11.

Wang, X., Berger, J.O., 2016. Estimating shape constrained functions using Gaussian processes. SIAM/ASA Journal on Uncertainty Quantification 4, 1–25.

Wang, Y.B., Cheng, J., Nakagawa, J., Yamamoto, M., 2010. A numerical method for solving the inverse heat conduction problem without initial value. Inverse Problems in Science and Engineering 18, 655–671.

Wang, Z., Huan, X., Garikipati, K., 2021. Variational system identification of the partial differential equations governing microstructure evolution in materials: Inference over sparse and spatially unrelated data. Computer Methods in Applied Mechanics and Engineering 377, 113706.

Wiebe, J., Cecílio, I., Dunlop, J., Misener, R., 2022. A robust approach to warped Gaussian process-constrained optimization. Mathematical Programming 196, 805–839.

Wilson, A., Adams, R., 2013. Gaussian process kernels for pattern discovery and extrapolation. In: International conference on machine learning. PMLR, pp. 1067–1075.

Xiong, X., Fu, C., Li, H.-F., 2006. Fourier regularization method of a sideways heat equation for determining surface heat flux. Journal of Mathematical Analysis and Applications 317, 331–348.

Yang, S., Wong, S.W., Kou, S., 2021. Inference of dynamic systems from noisy and sparse data via manifold-constrained Gaussian processes. Proceedings of the National Academy of Sciences 118, e2020397118.

Yang, X., Tartakovsky, G., & Tartakovsky, A. (2018). Physics-informed kriging: A physics-informed Gaussian process regression method for data-model convergence. arXiv preprint arXiv:1809.03461.

Zhang, X., 2001. Fiber orientation in a headbox. University of British Columbia.

Zhu, C., Byrd, R.H., Lu, P., Nocedal, J., 1997. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. ACM Transactions on mathematical software (TOMS) 23, 550–560.