INDUSTRIAL AND SYSTEMS ENGINEERING

Blending Physics with Data Using An Efficient Gaussian Process Regression with Soft Inequality and Monotonicity Constraints

DIDEM KOCHAN[1] AND XIU YANG[1]

[1]Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, 18015 USA

ISE Technical Report 24T-012

LEHIGH UNIVERSITY

Blending Physics with Data Using An Efficient Gaussian Process Regression with Soft Inequality and Monotonicity Constraints

Didem Kochan [1], **Xiu Yang** [1,*]

[1] *Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, United States of America*

Correspondence*:
Xiu Yang
xiy518@lehigh.edu

NOMENCLATURE

Abbreviations	
additiveGP	Additive Gaussian Process method
GP	Gaussian Process
HMC	Hamiltonian Monte Carlo
HMCad	Hard-constrained Hamiltonian Monte Carlo with adaptivity
HMCboth	Hard-constrained Hamiltonian Monte Carlo with both adaptivity and variance
HMCsoftad	Soft-constrained Hamiltonian Monte Carlo with adaptivity
HMCsoftboth	Soft-constrained Hamiltonian Monte Carlo with both adaptivity and variance
HMCsoftvar	Soft-constrained Hamiltonian Monte Carlo with variance
HMCvar	Hard-constrained Hamiltonian Monte Carlo with variance
MCMC	Markov chain Monte Carlo
MH	Metropolis-Hastings
PDE	Partial differential equations
QHMC	Quantum-inspired Hamiltonian Monte Carlo
QHMCad	Hard-constrained Quantum-inspired Hamiltonian Monte Carlo with adaptivity
QHMCboth	Hard-constrained Quantum-inspired Hamiltonian Monte Carlo with adaptivity and variance
QHMCsoftad	Soft-constrained Quantum-inspired Hamiltonian Monte Carlo with adaptivity
QHMCsoftboth	Soft-constrained Quantum-inspired Hamiltonian Monte Carlo with both adaptivity and variance
QHMCsoftvar	Soft-constrained Quantum-inspired Hamiltonian Monte Carlo with variance
QHMCvar	Hard-constrained Quantum-inspired Hamiltonian Monte Carlo with variance
SNR	Signal-to-noise ratio
tnHMC	Truncated Gaussian method with Hamiltonian Monte Carlo sampling
tnQHMC	Truncated Gaussian method with Quantum-inspired Hamiltonian Monte Carlo sampling
Symbols	
$\delta_{x,x'}$	Kronecker Delta
σ^2	Signal variance
θ	Hyperparameters of Gaussian model
l	Length-scale

ABSTRACT

In this work, we propose a new Gaussian process (GP) regression framework that enforces the physical constraints in a probabilistic manner. Specifically, we focus on inequality and monotonicity constraints. This GP model is trained by the quantum-inspired Hamiltonian Monte Carlo (QHMC) algorithm, which is an efficient way to sample from a broad class of distributions by allowing a particle to have a random mass matrix with a probability distribution. Integrating the QHMC into the inequality and monotonicity constrained GP regression in the probabilistic sense, our approach enhances the accuracy and reduces the variance in the resulting GP model. Additionally, the probabilistic aspect of the method leads to reduced computational expenses and execution time. Further, we present an adaptive learning algorithm that guides the selection of constraint locations. The accuracy and efficiency of the method are demonstrated in estimating the hyperparameter of high-dimensional GP models under noisy conditions, reconstructing the sparsely observed state of a steady state heat transport problem, and learning a conservative tracer distribution from sparse tracer concentration measurements.

1 INTRODUCTION

In many real-world applications, measuring complex systems or evaluating computational models can be time-consuming, costly or computationally intensive. Gaussian process (GP) regression is one of the Bayesian techniques that addresses this problem by building a surrogate model. It is a supervised machine learning framework that has been widely used in regression and classification tasks. A GP can be interpreted as a suitable probability distribution on a set of functions, which can be conditioned on observations using Bayes' rule (Lange-Hegermann, 2021). GP regression has found applications in various challenging practical problems including multi-target regression problems Nabati et al. (2022), biomedical applications Dürichen et al. (2014); Pimentel et al. (2013), robotics Williams et al. (2008) and mechanical engineering applications Song et al. (2021); Li et al. (2023), etc. The recent research demonstrate that a GP regression model can make predictions incorporating prior information (kernels) and generate uncertainty measures over predictions (Rasmussen et al., 2006). However, prior knowledge often includes physical laws, and using the standard GP regression framework may lead to an unbounded model in which some points can take infeasible values that violate physical laws (Lange-Hegermann, 2021). For example, non-negativity is a requirement for various physical properties such as temperature, density and viscosity (Pensoneault et al., 2020). Incorporating physical information in GP framework can regularize the behaviour of the model and provide more realistic uncertainties, since the approach concurrently evaluates problem data and physical models (Swiler et al., 2020; Ezati et al., 2024).

A significant amount of research has been conducted to incorporate physical information in GP framework, resulting in various techniques and methodologies (Swiler et al., 2020). For example, a probit model for the likelihood of derivative information can be employed to enforce monotonicity constraints (Riihimäki and Vehtari, 2010). Although this approach can also be used to enforce convexity in one dimension, an additional requirement on Hessian is incorporated for higher dimensions (Da Veiga and Marrel, 2012). In (López-Lopera et al., 2022) an additive GP approach is introduced to account for monotonicity constraints. Although posterior sampling step can be challenging, the additive GP framework enables to satisfy the constraints everywhere in the input space, and it is scalable to higher dimensions. The work presented in Gulian et al. (2022) presents a framework in which spectral decomposition covariance kernels and differential equation constraints are used in a co-kriging setup to perform GP regression constrained by boundary value problems. With their inherent advantages, physics-informed GP models that incorporate physical constraints has applications in diverse areas, such as manufacturing Qiang et al. (2023), forecasting in power grids Mao et al. (2020) or urban flooding models Kohanpur et al. (2023), mimicing drivers' behavior Wang et al. (2021), monitoring intelligent tire systems Barbosa et al. (2021), predicting fuel flow rate Chati and Balakrishnan (2017), designing wind turbines Wilkie and Galasso (2021), etc. Due to their flexibility, physics-informed GP models can be combined with several approaches to enhance the accuracy of model predictions. These works show that integrating physical knowledge into the prediction process provides accurate results.

Enforcing inequality constraints into a GP is typically challenging as the conditional process, subject to these constraints, does not retain the properties of a GP (Maatouk and Bay, 2017). One of the approaches to handle this problem is a data augmentation approach in which the inequality constraints are enforced at various locations and approximate samples are drawn from the predictive distribution (Abrahamsen and Benth, 2001), or using a block covariance kernel (Raissi et al., 2017). Implicitly constrained GP regression method proposed in (Salzmann and Urtasun, 2010) shows that the mean prediction of a GP implicitly satisfies linear constraints, if the constraints are satisfied by the training data. A similar approach shows that when we impose linear inequality constraints on a finite set of points in the domain, the resulting process is a compound Gaussian Process with a truncated Gaussian mean (Agrell, 2019). Most of the approaches assume that the inequalities are satisfied on a finite set of input locations. Based on that assumption, the methods approximate the posterior distribution given those constraint input points. The approach introduced in (Da Veiga and Marrel, 2012) is an example of these methods, where maximum likelihood estimation of GP hyperparameters are investigated under the constraint assumptions. In practice, this should also limit the number of constraint points needed for an effective discrete-location approximation. In addition, the method is not efficient on high-dimensional datasets as it takes a large amount of time to train the GP model.

To the best of our knowlege, the first Gaussian method that satisfies certain inequalities at all the input space is proposed by Maatouk and Bay (Maatouk and Bay, 2017). The GP approximation of the samples are performed in the finite-dimensional space functions, and a rejection sampling method is used for approximating the posterior. The convergence properties of the method is investigated in (Maatouk et al., 2015). Although using the rejection sampling to obtain posterior helps convergence, it might be computationally expensive. Similar to the previous approaches in which a set of inputs satisfy the constraints, this method also suffers from the curse of dimensionality. Later, the truncated Gaussian approach (López-Lopera et al., 2018) extends the framework in (Maatouk and Bay, 2017) to general sets of linear inequalities. Building upon the approaches in (Maatouk and Bay, 2017) and (Maatouk et al., 2015), the work presented in (López-Lopera et al., 2018) introduces a finite-dimensional approach that incorporates inequalities for both data interpolation and covariance parameter estimation. In this work, the posterior distribution is expressed as a truncated multinormal distribution. The method uses different Markov Chain Monte Carlo (MCMC) methods and exact sampling methods to obtain the posterior distribution. Among the various MCMC sampling techniques including Gibbs, Metropolis-Hastings (MH) and Hamiltonian Monte Carlo (HMC), the results indicate that HMC sampling is the most efficient one. The truncated Gaussian approaches offer several advantages, including the ability to achieve high accuracy and the flexibility in satisfying multiple inequality conditions. However, although those types of methods address the limitations in (Maatouk and Bay, 2017), they might be time consuming particularly in applications with large datasets or high-dimensional spaces.

In this work, we use QHMC algorithm to train the GP model, and enforce the inequality and monotonicity constraints in a probabilistic manner. Our work addresses the computational limitations caused by high dimensions or large datasets. Unlike truncated Gaussian methods in (López-Lopera et al., 2018) for inequality constraints, or additive GP (López-Lopera et al., 2022) with monotonicity constraints, the proposed method can maintain its efficiency on higher dimensions. Further, we adopt an adaptive learning algorithm that selects the constraint locations. The efficiency and accuracy of the QHMC algorithms are demonstrated on inequality and monotonicity constrained problems. Inequality constrained examples include lower and higher dimensional synthetic problems, a conservative tracer distribution from sparse tracer concentration measurements and a three-dimensional heat transfer problem, while monotonicity constrained examples provide lower and higher dimensional synthetic problems. Our contributions can be summarized in three key points: (i) QHMC reduces difference between posterior mean and the ground truth, (ii) utilizing QHMC in a probabilistic sense decreases variance and uncertainty, and (iii) the proposed algorithm is a robust, efficient and flexible method applicable to a wide range of problems. We implemented QHMC sampling in the truncated Gaussian approach to enhance accuracy and efficiency while working with the QHMC algorithm.

2 GAUSSIAN PROCESS UNDER INEQUALITY CONSTRAINTS

2.1 Standard GP regression framework

Suppose we have a target function represented by values $\mathbf{y} = (y^{(1)}, y^{(2)}, ..., y^{(T)})^N$, where $y^{(i)} \in \mathbb{R}$ are observations at locations $\mathbf{X} = \{x^{(i)}\}_{i=1}^N$. Here, $x^{(i)}$ represents d-dimensional vectors in the domain $\mathcal{D} \in \mathbb{R}^d$. Using the framework provided in Kuss and Rasmussen (2003), we approximate the target function by a GP, denoted as $Y(.,.) : D \times \Omega \to \mathrm{R}$. We can express Y as

$$Y(x) := GP[\mu(x), K(x, x')], \tag{1}$$

where $\mu(.)$ is the mean function and $K(x, x')$ is the covariance function defined as

$$\mu(x) = \mathbb{E}[Y(x)], \quad \text{and} \quad K(x, x') = \mathbb{E}[Y(x) - \mu(x)][Y(x') - \mu(x')] \tag{2}$$

Typically, the standard squared exponential covariance kernel can be used as a kernel function:

$$K(x, x') = \sigma^2 \exp\left(-\frac{||x - x'||_2^2}{2l^2}\right) + \sigma_n^2 \delta_{x,x'}, \tag{3}$$

where σ^2 is the signal variance, $\delta_{x,x'}$ is the Kronecker delta function and l is the length-scale. We then assume that the observation includes an additive independent identically distributed (i.i.d.) Gaussian noise term ϵ and having zero mean and variance σ_n^2. We denote the hyperparameters by $\theta = (\sigma, l, \sigma_n)$, and estimate them using the training data. The parameters can be estimated by minimizing the negative marginal log-likelihood Kuss and Rasmussen (2003); Stein (1988); Zhang (2004):

$$-\log[p(\mathbf{Y}|\mathbf{X}, \theta)] = \frac{1}{2}[(\mathbf{y} - \mu)^{\mathrm{T}} K^{-1}(\mathbf{y} - \mu) + \log|K| + N \log(2\pi)]. \tag{4}$$

The following section shows how the parameter updates are performed using the QHMC method.

2.2 Quantum-inspired Hamiltonian Monte Carlo

QHMC is an enhanced version of the HMC algorithm that incorporates a random mass matrix for the particles, following a probability distribution. In conventional HMC, the position is represented by the original variables (x), while Gaussian momentum is represented by auxiliary variables (q). Utilizing the energy-time uncertainty relation of quantum mechanics, QHMC allows a particle to have a random mass matrix with a probability distribution. Consequently, in addition to the position and momentum variables, a mass variable (m) is introduced within the QHMC framework. Having a third variable offers the advantage of exploring various landscapes in the state-space. As a result, unlike standard HMC or conventional sampling methods such as MH and Gibbs, QHMC can perform well on discontinuous, non-smooth and spiky distributions Barbu and Zhu (2020); Liu and Zhang (2019). In particular, while the performance of HMC and MH sampling degrade when the distribution is ill-conditioned or multi-modal, the performance of QHMC does not have these limitations. Moreover, QHMC maintains its performance with almost zero additional cost of resampling the mass variable. Due to its efficiency and adaptibility, QHMC can easily integrate with other techniques, or be modified to enhance its performance based on specific objectives and applications. For example, stochastic versions of QHMC can yield accurate solutions with increased efficiency, and the approach is applicable to various scenarios involving missing data Liu and Zhang (2019); Kochan et al. (2022).

The quantum nature of QHMC can be understood by considering a one-dimensional harmonic oscillator example provided in Liu and Zhang (2019). Let us consider a ball with a fixed mass m attached to a spring at the origin. Assuming x is the displacement, the magnitude of the restoring force that pulls back the ball to the origin is $F = -kx$, and the ball oscillates around the origin with period $T = 2\pi\sqrt{\frac{m}{k}}$. In contrast to standard HMC where the mass m is fixed at 1, QHMC incorporates a time-varying mass, allowing the ball to experience acceleration and explore various distribution landscapes. That is, QHMC has the capability to employ a short time period T, corresponding to a small mass m, to efficiently explore broad but flat regions. Conversely, in spiky regions, it can switch to a larger time period T, *i.e.* larger m, to ensure thorough exploration of all corners of the landscape Liu and Zhang (2019).

The implementation of QHMC is straightforward: (i) construct a stochastic process $M(t)$ for the mass, and at each time t, (ii) sample $M(t)$ from a distribution $P_M(M)$. Resampling the positive-definite mass matrix is the only additional step to the standard HMC procedure. In practice, assuming that $P_M(M)$ is independent of x and q, a mass density function $P_M(M)$ with mean μ_m and variance σ_m^2 can be where I is the identity matrix. QHMC framework simulates the following dynamical system:

$$d\begin{pmatrix} x \\ q \end{pmatrix} = dt\begin{pmatrix} M(t)^{-1}q \\ -\nabla U(x) \end{pmatrix}. \tag{5}$$

In this setting, the potential energy function of the QHMC system is $U(x) = -\log[p(\mathbf{Y}|\mathbf{X}, \theta)]$, i.e., the negative of marginal log-likelihood. Algorithm 1 summarizes the steps of QHMC sampling, and, here, we consider the location variables $\{x^{(i)}\}_{i=1}^N$ in GP model as the position variables x in Algorithm 1. The method evolves the QHMC dynamics to update the locations x. In this work, we implement the QHMC method for inequality constrained GP regression in a probabilistic manner.

2.3 Proposed method

Instead of enforcing all constraints strictly, the approach introduced in Pensoneault et al. (2020) minimizes the negative marginal log-likelihood function in Equation 4 while allowing constraint violations with a small probability. For example, for non-negativity constraints, the following requirement is imposed to the problem:

$$P[(\mathbf{Y}(x)|x,\theta) < 0] \leq \eta, \quad \text{for all} \quad x \in \mathcal{D}, \tag{6}$$

where $0 < \eta << 1$.

In contrast to enforcing the constraint via truncated Gaussian assumption Maatouk and Bay (2017) or performing inference based on the Laplace approximation and expectation propagation Jensen et al. (2013), the proposed method preserves the Gaussian posterior of the standard GP regression. The method uses a slight modification of the existing cost function. Given a model that follows a Gaussian distribution, the constraint can be re-expressed by the posterior mean and posterior standard deviation:

$$y^*(x) + \phi^{-1}(\eta)s(x) \geq 0, \quad \text{for all} \quad x \in \mathcal{D}, \tag{7}$$

where $y^*(x)$ stands for the posterior mean, s is the standard deviation and ϕ is the cumulative distribution function of a Gaussian random variable. Following the work in Pensoneault et al. (2020), in this study η was set to 2.2% for demonstration purposes. As a result, $\phi^{-1}(\eta) = -2$, indicating that two standard deviations below the mean is still nonnegative. Then, the formulation of the optimization problem is given as

$$\begin{gathered}\text{argmin}_\theta \quad -\log[p(\mathbf{Y}|\mathbf{X},\theta)] \quad \text{such that} \\ y^*(x) - 2s(x) \geq 0.\end{gathered} \tag{8}$$

In this particular form of the optimization problem, a functional constraint described by Equation 8 is existent. It can be prohibitive or impossible to satisfy this constraint at all points across the entire domain. Therefore, we adopt a strategy where Equation 8 is enforced only on a selected set of m constraint points denoted as $\mathbf{X}_c = {x_c^{(i)}}_{i=1}^m$. The optimization problem can be reformulated as

$$\begin{gathered}\text{argmin}_\theta \quad -\log[p(\mathbf{Y}|\mathbf{X},\theta)] \quad \text{such that} \\ y^*(x_c^{(i)}) - 2s(x_c^{(i)}) \geq 0 \quad \text{for all} \quad i = 1, 2, ..., m,\end{gathered} \tag{9}$$

where hyperparameters are estimated to enforce bounds. Solving this optimization problem can be very challenging, and hence, in Pensoneault et al. (2020) additional regularization terms are added. Rather than directly solving the optimization problem, this work adopts the soft-QHMC method, which introduces inequality constraints with a high probability (e.g., 95%) by selecting a specific set of m constraint points in the domain. Then non-negativity on the posterior GP is enforced at these selected points. The log-likelihood in Equation 4 is minimized using the QHMC algorithm. Leveraging the Bayesian estimation Gelman et al. (2014), we can approximate the posterior distribution by log-likelihood function and prior probability distribution as shown in the following:

$$p(\mathbf{X},\theta|\mathbf{Y}) \propto p(\mathbf{X},\theta,\mathbf{Y}) = p(\theta)p(\mathbf{X}|\theta)p(\mathbf{Y}|\mathbf{X},\theta). \tag{10}$$

The QHMC training flow starts with this Bayesian learning and proceeds with an MCMC procedure for drawing samples generated by the Bayesian framework. A general sampling procedure at step t is given as

$$\begin{aligned}X^{(t+1)} &\sim \pi(X|\theta) = p(X|\theta^{(t)}, Y), \\ \theta^{(t+1)} &\sim \pi(\theta|X) = p(\theta|X^{(t+1)}, Y).\end{aligned} \tag{11}$$

The workflow of soft inequality-constrained GP regression is summarized in Algorithm 2, where QHMC sampling (provided in Algorithm 1) is used as a GP training method. In this version of non-negativity enforced GP regression, the constraint points are located where the posterior variance is highest.

Algorithm 1 QHMC Training for GP with Inequality Constraints

Input: Initial point x_0, step size ϵ, number of simulation steps L, mass distribution parameters μ_m and σ_m.

for $t = 1, 2, ...$ **do**
  Resample $M_t \sim P_M(M)$
  Resample $q_t \sim N(0, M_t)$
  $(x_0, q_0) = (x^{(t)}, q^{(t)})$
  $q_0 \leftarrow q_0 - \frac{\epsilon}{2}\nabla U(x_0)$
  for $i = 1, 2, ..., L-1$ **do**
    $x_i \leftarrow x_{i-1} + \epsilon M_t^{-1} q_{i-1}$
    $q_i \leftarrow q_{i-1} - \frac{\epsilon}{2}\nabla U(x_i)$
  end for
  $x_L \leftarrow x_{L-1} + \epsilon M_t^{-1} q_{L-1}$
  $q_L \leftarrow q_{L-1} - \frac{\epsilon}{2}\nabla U(x_L)$
  $(\hat{x}, \hat{q}) = (x_L, q_L)$
  MH step: $u \sim$ Uniform$[0, 1]$;
  $\rho = e^{-H(\hat{x},\hat{q}) + H(x^{(t)}, q^{(t)})}$;
  if $u < \min(1, \rho)$ **then**
    $(x^{(t+1)}, q^{(t+1)}) = (\hat{x}, \hat{q})$
  else
    $(x^{(t+1)}, q^{(t+1)} = (x^{(t)}, q^{(t)})$
  end if
end for
Output: $\{x^{(1)}, x^{(2)}, ...\}$

Algorithm 2 Soft Inequality-constrained GP Regression

Specify m constraint points denoted by $\mathbf{X}_c = x_c^{(i)}{}_{i=1}^{m}$, where corresponding observation $y^*(x_c)^{(i)}$.
for $i = 1, 2, ..., m$ **do**
  Compute the MSE of $s^2(x_c^{(i)})$ of MLE prediction $y^*(x_c)$ for $x_c \in \mathcal{D}$.
  Obtain observation $y^*(x_c)^{(i)}$ at $x_c^{(i)}$
  Locate $x_c^{(i+1)}$ for the maximum of $s^2(x_c^{(i)})$ for $x_c \in \mathcal{D}$.
end for
Construct the MLE prediction of $y^*(x)$ using QHMC training.

2.3.1 Enforcing Monotonicity Constraints

Monotonicity constraints on a GP can be enforced using the likelihood of derivative observations. After the selection of active constraints, non-negativity constraints are incorporated in the partial derivative, *i.e.*

$$\frac{\partial f}{\partial x_i}(\mathbf{x_i}) \geq 0, \tag{12}$$

where f is a vector of N latent values. In the soft-constrained GP method, we introduce the non-negativity information in Equation 12 on a set of selected points, and apply the same procedure as in Equation 9.

Since the derivative is also a GP with with mean and covariance matrix Riihimäki and Vehtari (2010):

$$\mu(x') = \mathbb{E}\left[\frac{\partial Y(x)}{\partial x_i}\right], \quad \text{and} \quad K(x,x') = \frac{\partial}{\partial x_i}\frac{\partial}{\partial x'_i}K(x,x'), \tag{13}$$

the new posterior distribution is given as

$$\begin{aligned} p(\mathbf{y}^*, \theta|\mathbf{y}, \mathbf{x}, \mathbf{x}^*) &= \int p(\mathbf{y}^*, \theta|f^*)p(f^*|\mathbf{y}, \mathbf{x}, \mathbf{x}^*)df, \\ p(f^*|\mathbf{y}, \mathbf{x}, \mathbf{x}^*) &= \int\int p(f^*|\mathbf{x}^*, f, f')p(f, f'|\mathbf{x}, \mathbf{y})df df', \end{aligned} \tag{14}$$

where $\mathbf{y}^*$ and $\mathbf{f}^*$ denote the predictions at location $\mathbf{x}^*$.

3 THEORETICAL ANALYSIS OF THE METHOD

In this section, employing Bayes' Theorem, we demonstrate how QHMC is capable of producing a steady-state distribution that approximates the actual posterior distribution. Then, we examine the convergence characteristics of the probabilistic approach on the optimization problem outlined in Equation 9.

3.1 Convergence of QHMC training

The study presented in Liu and Zhang (2019) demonstrates that the QHMC framework can effectively capture a correct steady-state distribution that describes the desired posterior distribution $p(x) \propto \exp(-U(x))$ via Bayes' rule. The joint probability density of (x, q, M) can be calculated by Bayesian theorem:

$$p(x, q, M) = p(x, q|M)P_M(M), \tag{15}$$

where the conditional distribution is approximated as follows:

$$p(x, q|M) \propto \exp\left(-U(x) - K(q)\right) = \exp\left(-U(x)\right)\exp\left(-\frac{1}{2}q^T M^{-1} q\right), \tag{16}$$

Then, $p(x)$ can be written as

$$p(x) = \int_q \int_M dqdMp(x, q, M) \propto \exp(-U(x)), \tag{17}$$

which shows that the marginal steady distribution approaches the true posterior distribution Liu and Zhang (2019).

3.2 Convergence properties of probabilistic approach

In this section, we show that satisfying the constraints on a set of locations x in the domain $\mathcal{D}$ preserves convergence. Recall the following inequality-constrained optimization problem:

$$\begin{aligned} &\text{argmin}_\theta \quad -\log[p(\mathbf{Y}|\mathbf{X}, \theta)] \quad \text{such that} \\ &\qquad y^*(x_c^{(i)}) - 2s(x_c^{(i)}) \geq 0 \quad \text{for all} \quad i = 1, 2, ..., m. \end{aligned} \tag{18}$$

Now, it is necessary to demonstrate that the result obtained by using the selected set of input locations converge to the value of the regression model's output. This convergence ensures that probabilistic approach will eventually result in a model that satisfy the desired conditions.

Note that throughout the proof, it is assumed that $\mathcal{D}$ is finite. The proof can be constructed for the cases whether the domain is countable or uncountable.

(i) Assume that the domain $\mathcal{D}$ is a countable set containing N elements. Then, select a subset $\mathcal{D}_m \in \mathcal{D}$ with m points, where $x_c^{(1)}, x_c^{(2)}, ..., x_c^{(m)} \in \mathcal{D}_m$. For each point $x \in \mathcal{D}$, there exists a Gaussian probability distribution. The set of distributions associated with $x \in \mathcal{D}$ is denoted as $\mathcal{P}$. For the constraint points $x \in \mathcal{D}_m$, there are m constraints and their corresponding probability distributions, which can be defined as $\mathcal{P}_m$. Additionally, we introduce a set $H(x)$ such that

$$H(x) := \{\theta | p(\mathbf{Y}|\mathbf{X}, \theta) < 0\}, \tag{19}$$

which covers the locations where the non-negativity constraint is violated. For each fixed $x \in \mathcal{D}$, define

$$\begin{aligned} v(x) &:= \inf_{P \in \mathcal{P}} P(\mathbf{Y}|\mathbf{X}, \theta) < 0 \equiv \inf_{P \in \mathcal{P}} P(H(x)), \quad \text{and} \\ v_m(x) &:= \inf_{P \in \mathcal{P}_m} P(\mathbf{Y}|\mathbf{X}, \theta) < 0 \equiv \inf_{P \in \mathcal{P}_m} P(H(x)). \end{aligned} \tag{20}$$

(ii) Assume that the domain $\mathcal{D}$ is a finite but uncountable set. In this case, a countable subset $\tilde{\mathcal{D}}$ with $x \in \tilde{\mathcal{D}}$ can be constructed. The set of probability distributions are defined as in case (i). Since $\mathcal{D}$ is finite, the set $\mathcal{D} \cup \{x\}$ is also finite. Consequently, the sets $H(x), v(x)$ and $v_m(x)$ can be constructed as in the first case. Next steps establish a convergence of v_m over v as $\mathcal{P}_m$ converges to $\mathcal{P}$.

First, let us provide distance metrics used throughout the proof. Following the definitions in Guo et al. (2015), let

$$d(x, A) := \inf_{x' \in A} ||x - x'|| \tag{21}$$

denote the distance from a point x to a set A. Then, the distance of two compact sets A and B can be defined as

$$\mathbb{D}(A, B) := \sup_{x \in A} d(x, B). \tag{22}$$

Then, the Hausdorff distance between A and B is defined as $\mathbb{H}(A, B) := \max\{\mathbb{D}(A, B), \mathbb{D}(B, A)\}$. Finally, we define a pseudo-metric $\mathbf{d}$ to describe the distance between two probability distributions P and $\tilde{P}$ as

$$\mathbf{d}(P, \tilde{P}) := \sup_{x \in \mathcal{D}} |P(H(x)) - \tilde{P}(H(x))|, \tag{23}$$

where $\mathcal{D}$ is the domain specified in Section 3.2.

ASSUMPTION 1. *Suppose that the probability distributions $\mathcal{P}$ and $\mathcal{P}_m$ satisfy the following conditions:*

1. *There exists a weakly compact set $\tilde{\mathcal{P}}$ such that $\mathcal{P} \subset \tilde{\mathcal{P}}$ and $\mathcal{P}_m \subset \tilde{\mathcal{P}}$.*
2. $\lim_{m \to N} \mathbf{d}(\mathcal{P}, \mathcal{P}_m) = 0$, *with probability 1.*
3. $\lim_{m \to N} \mathbf{d}(\mathcal{P}_m, \mathcal{P}) = 0$, *with probability 1.*

Now, we show that Theorem 1 holds under the assumptions in Assumption 1. Recall that we have

$$\mathbb{H}(\text{conv}V, \text{conv}V_m) = \max\left\{\left|\sup_{P\in\mathcal{P}_m} P(H(x)) - \sup_{P\in\mathcal{P}} P(H(x))\right|, \left|\inf_{P\in\mathcal{P}_m} P(H(x)) - \inf_{P\in\mathcal{P}} P(H(x))\right|\right\}.$$

Based on the definition and property of Hausdorff distance Hess (1999) we also have

$$\mathbb{H}(\text{conv}V, \text{conv}V_m) \leq \mathbb{H}(V, V_m) \leq \max\{\mathbb{D}(V, V_m), \mathbb{D}(V_m, V)\}. \tag{24}$$

Consider the distance of two sets:

$$\begin{aligned}
\mathbb{D}(V, V_m) &= \sup_{v\in V} \inf_{v'\in V_m} ||v - v'|| \\
&= \sup_{P\in\mathcal{P}} \inf_{\tilde{P}\in\mathcal{P}_m} ||P(H(x)) - \tilde{P}(H(x))|| \\
&\leq \sup_{P\in\mathcal{P}} \inf_{\tilde{P}\in\mathcal{P}_m} \sup_{x\in\mathcal{D}} ||P(H(x)) - \tilde{P}(H(x))|| \\
&= \mathbf{d}(\mathcal{P}, \mathcal{P}_m),
\end{aligned} \tag{25}$$

and apply the same procedure to obtain $\mathbb{D}(V_m, V) \leq \mathbf{d}(\mathcal{P}_m, \mathcal{P})$. Hence,

$$\mathbb{H}(\text{conv}V, \text{conv}V_m) \leq \mathbb{H}(V, V_m) \leq \mathbb{H}(\mathcal{P}_m, \mathcal{P}). \tag{26}$$

Consequently, we obtain

$$\begin{aligned}
|v_m(x) - v(x)| &\leq \left|\inf_{P\in\mathcal{P}_m} P(H(x)) - \inf_{P\in\mathcal{P}} P(H(x))\right| \\
&\leq \mathbb{H}(\text{conv}V, \text{conv}V_m) \\
&\leq \mathbb{H}(\mathcal{P}_m, \mathcal{P}).
\end{aligned} \tag{27}$$

THEOREM 1. *v_m converges to v as $\mathcal{P}_m$ converges to $\mathcal{P}$, that is*

$$\lim_{m\to N} \sup_{x\in\mathcal{D}} |v_m(x) - v(x)| = 0.$$

PROOF. Let us assume that $x \in \mathcal{D}$ is fixed, and define

$$V := \{P(H(x)) : P \in \text{cl}\mathcal{P}\}, \quad \text{and,} \quad V_m := \{P(H(x)) : P \in \text{cl}\mathcal{P}_m\}, \tag{28}$$

where cl represents the closure. Note that both V and V_m are bounded subsets in $\mathbb{R}^d$. Let us define a, b, a_m and b_m such that

$$a := \inf_{v\in V} v, \quad b := \sup_{v\in V} v, \quad a_m := \inf_{v\in V_m} v, \quad b_m := \sup_{v\in V_m} v, \tag{29}$$

The Hausdorff distance between convex hulls (conv) of the sets V and V_m are calculated as Hess (1999)

$$\mathbb{H}(\text{conv}V, \text{conv}V_m) = \max\{|b_m - b|, |a - a_m|\}. \tag{30}$$

Since we know that

$$b_m - b = \sup_{v \in V_m} v - \sup_{v \in V} v, \quad \text{and} \quad a_m - a = \inf_{v \in V_m} v - \inf_{v \in V} v, \tag{31}$$

we have

$$\mathbb{H}(\text{conv}V, \text{conv}V_m) = \max\left\{\left|\sup_{P \in \mathcal{P}_m} P(H(x)) - \sup_{P \in \mathcal{P}} P(H(x))\right|, \left|\inf_{P \in \mathcal{P}_m} P(H(x)) - \inf_{P \in \mathcal{P}} P(H(x))\right|\right\} \tag{32}$$

Based on the definition and property of Hausdorff distance Hess (1999) we have

$$\mathbb{H}(\text{conv}V, \text{conv}V_m) \leq \mathbb{H}(V, V_m), \tag{33}$$

resulting in Guo et al. (2015)

$$|v_m(x) - v(x)| \leq \mathbb{H}(V, V_m) \leq \mathbb{H}(\mathcal{P}, \mathcal{P}_m). \tag{34}$$

In this setting, x can be any point in $\mathcal{D}$, and the right hand side of the inequality is independent of x. The proof can be completed by taking the supremum of each side with respect to x Guo et al. (2015).

4 NUMERICAL EXAMPLES

In this section, we evaluate the performance of the proposed algorithms on various examples including synthetic and real data. The evaluations consider the size and dimension of the datasets. Several versions of QHMC algorithms are introduced and compared depending on the selection of constraint point locations and probabilistic approach.

Rather than randomly locating m constraint points, the algorithm starts with an empty constraint set and determine the locations of the constraint points one by one adaptively. Throughout this process, various strategies are employed for adding the constraints. The specific approaches are outlined as follows:

1. Constraint-adaptive approach: This approach examines whether the constraint is satisfied at a location. The function value is calculated, and if the constraint is violated at that location, then a constraint point is added.
2. Variance-adaptive approach: This approach calculates the prediction variance in the test set. Constraint points are identified at the positions where the variance values are highest. The goal here is basically to reduce the variance in predictions and increase the stability.
3. Combination of constraint and variance adaption: In this approach, a threshold value (e.g. 0.20) is determined for the variance, and the algorithm locates constraint points to the locations where the highest prediction variance is observed. Once the variance reduces to the threshold value, the algorithm switches to the first strategy, in which it locates constraint points where the violation occurs.

We represent the constraint-adaptive, hard-constrained approach as QHMCad and its soft-constrained counterpart as QHMCsoftad. Similarly, QHMCvar refers to the method focusing on variance, while QHMCsoftvar corresponds to its soft-constrained version. The combination of the two approaches with hard and soft constraints are denoted by QHMCboth and QHMCsoftboth, respectively. For the sake of comparison, truncated Gaussian algorithms using an HMC sampler (tnHMC) and a QHMC sampler (tnQHMC) for inequality-constrained examples are implemented, while additive GP (additiveGP) algorithm is adapted for monotonicity-constrained examples.

For the synthetic examples, the time and accuracy performances of the algorithms are evaluated while simultaneously changing the dataset size and noise level in the data. Following Pensoneault et al. (2020), as our metric, we calculate the relative l_2 error between the posterior mean y^* and the true value of the target function $f(x)$ on a set of test points $\mathbf{X}_t = \{x_T^{(i)}\}_{i=1}^{N_t}$:

$$E = \sqrt{\frac{\sum_{i=1}^{N_t}[y^*(x_T^{(i)}) - f(x_T^{(i)})]^2}{\sum_{i=1}^{N_t} f(x_T^{(i)})^2}}. \tag{35}$$

We solve the constrained optimization problems in MATLAB. Additionally, in order to highlight the advantage of QHMC over HMC, the proposed approach is implemented with using the standard HMC procedure. The relative error, posterior variance and execution time of each version of QHMC and HMC algorithms are presented.

4.1 Inequaltiy Constraints

This section provides two synthetic examples and two real-life application examples to demonstrate the effectiveness of QHMC algorithms on inequality constraints. Synthetic examples compare the performance QHMC approach with truncated Gaussian methods for a 2-dimensional and a 10-dimensional problems. For the 2-dimensional example, the primary focus is on enforcing the non-negativity constraints within the GP model. In the case of the 10-dimensional example, we generalize our analysis to satisfy a different inequality constraint, and evaluate the performances of truncated Gaussian, QHMC and soft-QHMC methods. Third example considers conservative transport in a steady-state velocity field in heterogeneous porous media. Despite being a two-dimensional problem, the non-homogeneous structure of the solute concentration introduces complexity and increases the level of difficulty. The last example is a 3-dimensional heat transfer problem in a hallow sphere.

4.1.1 Example 1

Consider the following 2D function under non-negativity constraints:

$$f(x) = \arctan 5x_1 + \arctan x_2, \tag{36}$$

where $\{x_1, x_2\} \in [0, 1]^2$. In this example, the GP model is trained via QHMC over 20 randomly selected locations.

Figure 1 presents the relative error values of the algorithms with respect to two parameters: the size of the dataset and signal-to-noise ratio (SNR). It can be seen that the most accurate results without adding any noise are provided by QHMCboth and tnQHMC algorithms with around 10% relative error. However, upon introducing the noise to the data and increasing its magnitude, a distinct pattern is observed. The QHMC methods exhibit relative error values of approximately 15% within the SNR range of 15% to 20%. In contrast, the relative error of the truncated Gaussian methods increases to 25% within the same noise range. This pattern demonstrates that QHMC methods can tolerate noise and maintain higher accuracy under these conditions.

In Table 1, the comparison between QHMC and HMC algorithms with a dataset size of 200 is presented. The relative error values indicate that QHMC yields approximately 20% more accurate results than HMC,

Table 1. Comparison of QHMC and HMC on 2D, inequality.

Method	Error	Posterior Var	Time	Method	Error	Posterior Var	Time
QHMC-ad	0.10	0.14	46s	HMC-ad	0.12	0.17	52s
QHMC-soft-ad	0.11	0.16	39s	HMC-soft-ad	0.13	0.19	48s
QHMC-var	0.11	0.12	40s	HMC-var	0.13	0.14	46s
QHMC-soft-var	0.12	0.15	34s	HMC-soft-var	0.15	0.14	42s
QHMC-both	**0.08**	0.13	48s	HMC-both	0.10	0.14	53s
QHMC-soft-both	0.09	0.13	39s	HMC-soft-both	0.12	0.15	44s

and it achieves this with a shorter processing time. Consequently, QHMC demonstrates both higher accuracy and efficiency compared to HMC.

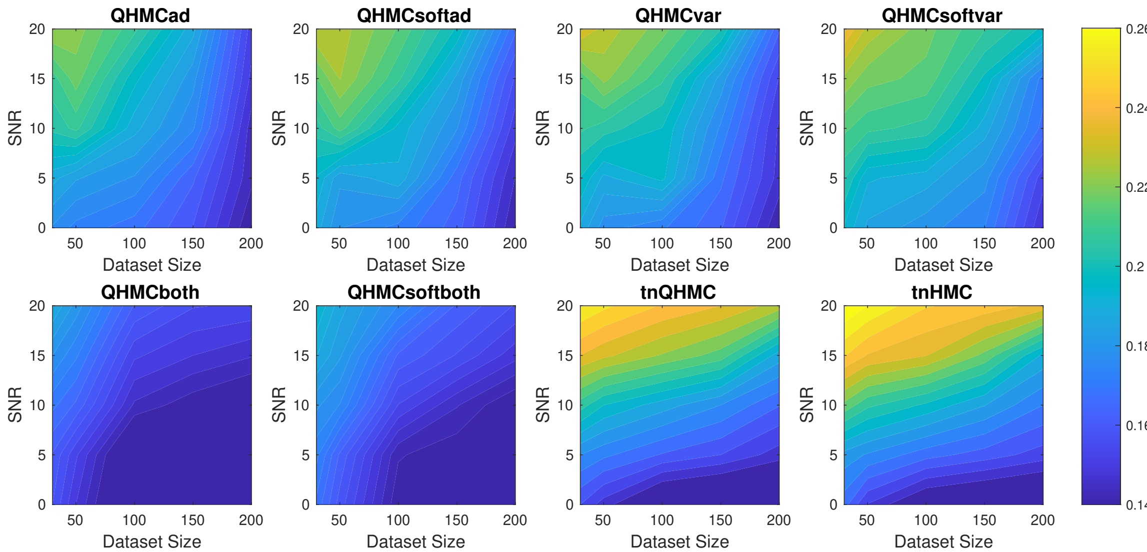

Figure 1. Relative error of the algorithms with different SNR and data sizes for Example 1 (2D), inequality.

Further, we compare the time performances of the algorithms in Figure 2 which demonstrates that QHMC methods, especially the probabilistic QHMC approaches can perform much faster than the truncated Gaussian methods. In this simple 2D example, the presence of noise does not significantly impact the running times of the QHMC algorithms. In contrast, truncated Gaussian algorithms are slower under noisy environment even when the dataset size is small. Additionally, it can be observed in Figure 3 that the QHMC algorithms, especially QHMCvar and QHMCboth are the most robust ones, as their small relative error comes with a small posterior variance. In contrast, the posterior variance values of the truncated Gaussian methods are higher than QHMC posterior variances even when there is no noise, and gets higher along with the relative error (see Figure 1) when the SNR levels increase. Combining all of these experiments, the inference is that QHMC methods achieve higher accuracy within a shorter time frame. Consequently, these methods prove to be more efficient and robust as they can effectively tolerate changes in parameters. Additionally, it is worth noting that a slight improvement is achieved in the performance of truncated Gaussian algorithms by implementing tnQHMC. Based on the numerical results obtained by tnQHMC, it can be concluded that employing tnQHMC not only yields higher accuracy but also saves some computational time compared to tnHMC.

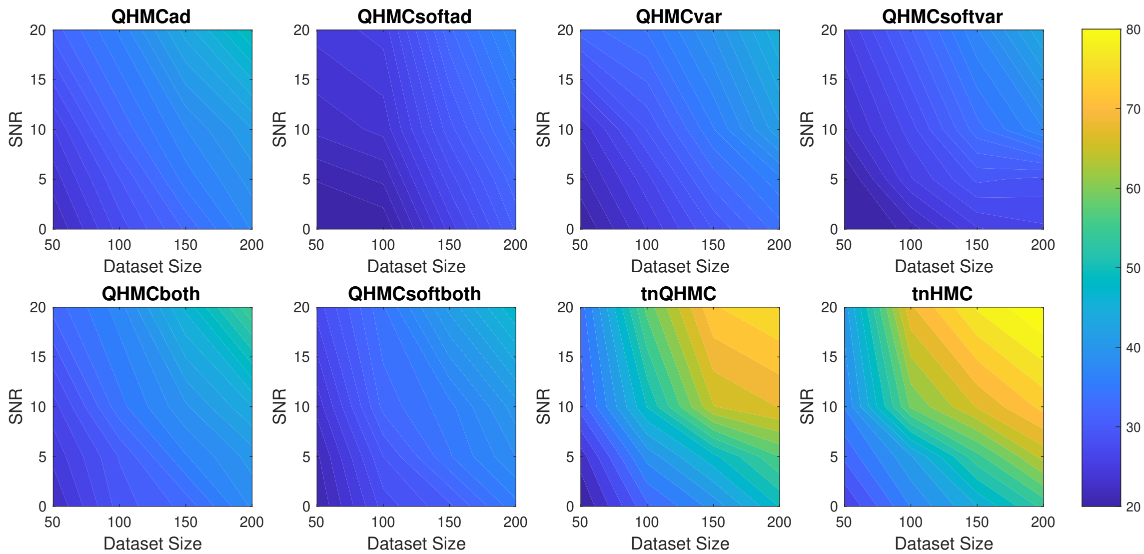

Figure 2. Execution times (in seconds) of the algorithms with different SNR and datasizes for Example 1 (2D), inequality.

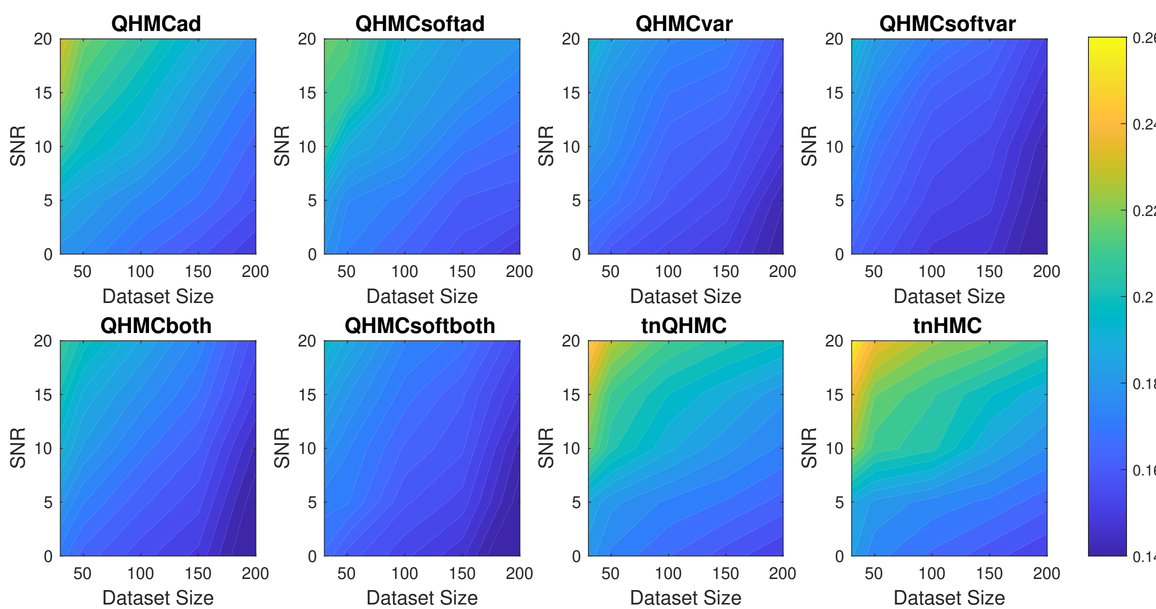

Figure 3. Posterior variances of the algorithms with different SNR and datasizes for Example 1 (2D), inequality.

Table 2. Comparison of QHMC and HMC on 10D, inequality.

Method	Error	Posterior Var	Time	Method	Error	Posterior Var	Time
QHMC-ad	0.10	0.13	39m 17s	HMC-ad	0.12	0.15	43m 33s
QHMC-soft-ad	0.11	0.14	36m 21s	HMC-soft-ad	0.13	0.15	41m 10s
QHMC-var	0.11	0.11	37m 4s	HMC-var	0.13	0.12	41m 31s
QHMC-soft-var	0.12	0.11	34m 23s	HMC-soft-var	0.14	0.12	37m 42s
QHMC-both	**0.09**	0.12	40m 8s	HMC-both	0.10	0.14	44m 23s
QHMC-soft-both	0.10	0.12	37m 53s	HMC-soft-both	0.12	0.14	42m 5s

352 ### 4.1.2 Example 2

353 Consider the 10D Ackley function Eriksson and Poloczek (2021) defined as follows:

$$f(x) = -a \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} \cos cx_i}\right) + a + \exp 1, \tag{37}$$

354 where $d = 10$, $a = 20$, $b = 0.2$ and $c = 2\pi$. We study the performance of the algorithms on the domain 355 $[-10, 10]^{10}$ while enforcing the function to be greater than 5.

356 Figure 4 illustrates that QHMCboth, QHMCsoftboth and truncated Gaussian algorithms yield the lowest 357 error when there is no noise in the data. However, as the noise level increases, truncated Gaussian methods 358 fall behind all QHMC approaches. Specifically, both the QHMCboth and QHMCsofthboth algorithms 359 demonstrate the ability to tolerate noise levels up to 15% with an associated relative error of approximately 360 15%. However, other variants of QHMC methods display greater noise tolerance when dealing with larger 361 datasets. With fewer than 100 data points, the error rate reaches around 25%, but it decreases to $15 - 20\%$ 362 when the number of data points exceeds 100.

363 Figure 5 illustrates the time comparison of the algorithms, where QHMC methods provide around 364 $30 - 35\%$ time efficiency for the datasets larger than a size of 150. Combining this time advantage with the 365 higher accuracy of QHMC indicates that both soft and hard constrained QHMC algorithms outperform 366 truncated Gaussian methods across various criteria. QHMC methods offer the flexibility to employ one 367 of the algorithms depending on the priority of the experiments. For example, if speed is the primary 368 consideration, QHMCsoftvar is the fastest method while maintaining a good level of accuracy. If accuracy 369 is the most important metric, employing QHMCboth would be a wiser choice, as it still offers significant 370 time savings compared to other methods.

371 Figure 6 presents that the posterior variance values of truncated Gaussian methods are significantly higher 372 than that of the QHMC algorithms, especially when the noise levels are higher than 5%. As expected, 373 QHMCvar and QHMCsoftvar algorithms offer the lowest variance, while QHMCboth and QHMCsoftboth 374 follow them. A clear pattern is shown in the figure, in which QHMC approaches can tolerate higher noise 375 levels especially when the dataset is large. It is notable that our method demonstrates a significant increase 376 in efficiency as the dimension increases. When comparing this 10D example to the 2D case, the execution 377 times of the truncated Gaussian methods are notably impacted by the dimension, even in the absence of 378 noise in the datasets. Although their relative error levels can remain low without noise, it takes 1.5 times 379 longer than the QHMC methods to offer those accuracy. Additionally, this observation holds only for cases 380 where the data is noise-free. As soon as noise is present, the accuracy of truncated Gaussian methods

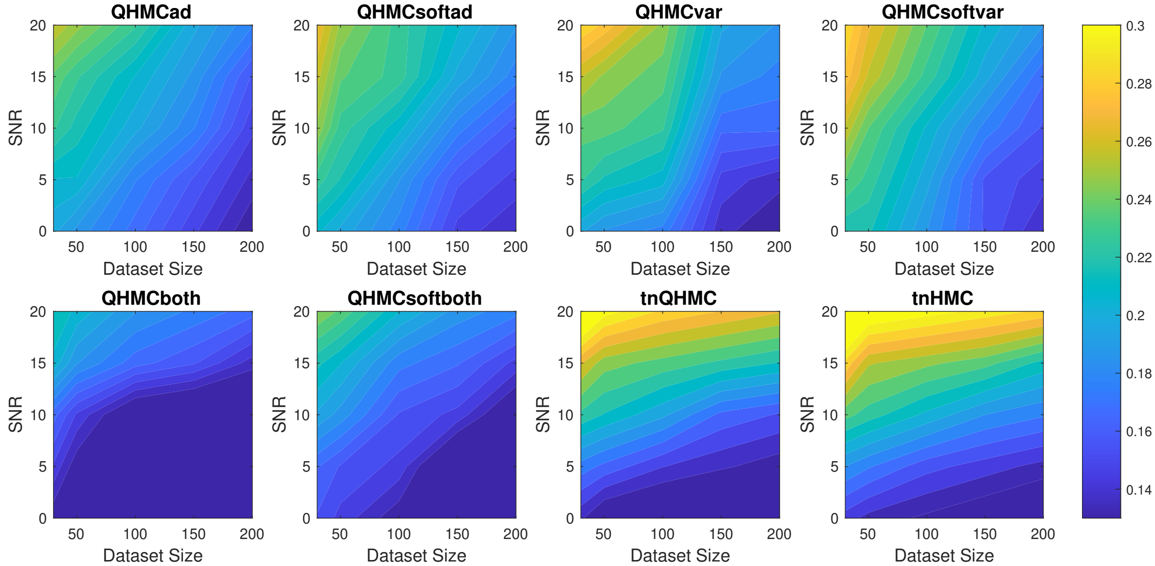

Figure 4. Relative error of the algorithms with different SNR and data sizes for Example 2 (10D), inequality.

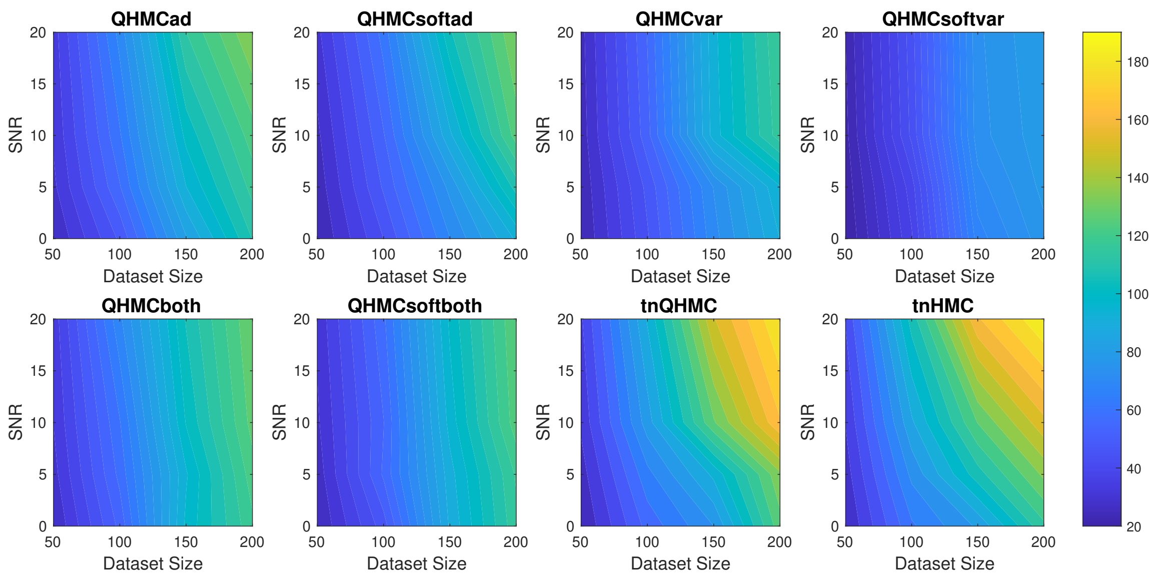

Figure 5. Execution times (in minutes) of the algorithms with different SNR and datasizes for Example 2 (10D), inequality.

deteriorates, whereas QHMC methods can withstand the noise and yield good results in a shorter time span.

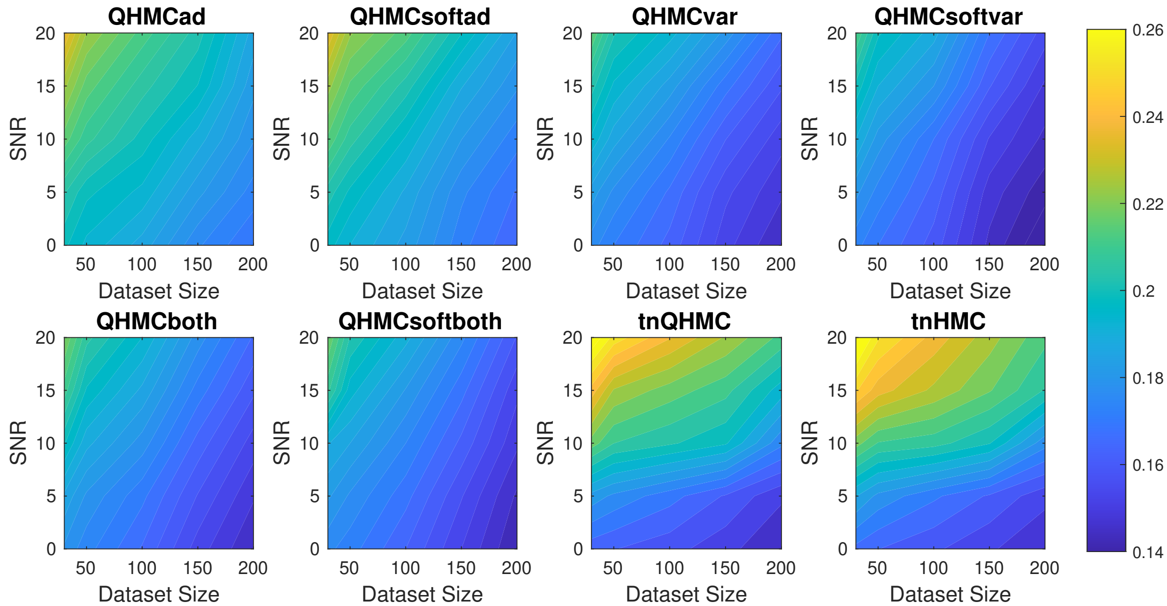

Figure 6. Posterior variances of the algorithms with different SNR and datasizes for Example 2 (10D), inequality.

4.1.3 Example 3: Solute transport in heterogeneous porous media

Following the example in Yang et al. (2019), we examine conservative transport within a constant velocity field in heterogeneous porous media. Let us denote the solute concentration by $C(\mathbf{x}, t)(\mathbf{x} = (x, y)^{\mathrm{T}})$, and suppose that the measurements of $C(\mathbf{x}, t)$ are available at various locations at different times. Conservation laws can be used to describe the processes of flow and transport. Specifically, Darcy flow equation describes the flow by Yang et al. (2019)

$$\begin{cases} \nabla \cdot (K \nabla h) = 0, & \mathbf{x} \in \mathbb{D}, \\ \frac{\partial h}{\partial \mathbf{n}} = 0, & y = 0 \text{ or } y = L_2, \\ h = H_1, & x = 0, \\ h = H_2, & x = L_1, \end{cases} \tag{38}$$

where $h(\mathbf{x}, w)$ is the hydraulic head, $\mathbb{D} = [0, L_1] \times [0, L_2]$ is the simulation domain with $L_1 = 256$ and $L_2 = 128$, H_1 and H_2 are known boundary head values and $K(\mathbf{x}, w)$ is the unknown hydraulic conductivity field. The field is represented as a stochastic process, with the distribution of values described by a log-normal distribution. Specifically, it is expressed as $K(\mathbf{x}, w) = \exp Z(\mathbf{x}, w)$, where is a second-order stationary GP with a known exponential covariance function, $\mathrm{Cov}\{Z(\mathbf{x}), Z(\mathbf{x}')\} = \sigma_Z^2 \exp\left(-|\mathbf{x} - \mathbf{x}'|/l_z\right)$ where variance $\sigma_Z^2 = 2$ and correlation length $l_z = 5$. The solute transport by the advection-dispersion equation Emmanuel and Berkowitz (2005); Lin and Tartakovsky (2009); Yang et al. (2019) can be described

Table 3. Comparison of QHMC and HMC on solute transport with nonnegativity.

Method	Error	Posterior Var	Time	Method	Error	Posterior Var	Time
QHMC-ad	0.18	0.13	83s	HMC-ad	0.20	0.14	89s
QHMC-soft-ad	0.19	0.13	75s	HMC-soft-ad	0.22	0.15	83s
QHMC-var	0.20	0.12	80s	HMC-var	0.23	0.13	91s
QHMC-soft-var	0.21	0.13	71s	HMC-soft-var	0.24	0.14	79s
QHMC-both	**0.13**	0.12	86s	HMC-both	0.15	0.14	97s
QHMC-soft-both	0.14	0.13	74s	HMC-soft-both	0.15	0.15	82s
tnQHMC	0.15	0.13	96s	tnHMC	0.16	0.16	103s

by

$$\begin{cases} \frac{\partial C}{\partial t} + \nabla \cdot (v\mathbf{C}) = \nabla \cdot \left(\frac{D_w}{\tau} + \alpha ||\mathbf{v}||_2 \right) \nabla C, & \mathbf{x} \text{ in } \mathbb{D}, \\ C = Q\delta(\mathbf{x} - \mathbf{x}^*), & t = 0, \\ \frac{\partial C}{\partial \mathbf{n}} = 0, & y = 0 \text{ or } y = L_2 \text{ or } x = L_1, \\ C = 0, & x = 0. \end{cases} \tag{39}$$

In this context, $C(\mathbf{x}, t; w)$ represents the solute concentration defined over $\mathbb{D} \times [0, T] \times \Omega$, $\mathbf{v}$ denotes the fluid velocity given by $\mathbf{v}(\mathbf{x}; w) = -K(\mathbf{x}; \omega)\nabla h(\mathbf{x}, \omega)/\phi$ with ϕ being porosity; D_w is the diffusion coefficient, τ stands for the tortuosity, and α is the dispersivity tensor, with diagonal components α_L and α_T. In this study, the transport parameters are defined as follows: $\phi = 0.317, \tau = \phi^{1/3}, D_w = 2.5 \times 10^{-5}, \alpha_L = 5$ and $\alpha_T = 0.5$. Lastly, the solute is instantaneously injected at $\mathbf{x}^* = (50, 64)$ at $t = 0$ with the intensity $Q = 1$ Yang et al. (2019). In Figure 7, the ground truth with observation locations and constraint locations are presented to provide an insight into the structure of solute concentration.

Table 3 presents a comparison of all versions of QHMC and HMC methods, along with the truncated Gaussian algorithms. Similar to the results observed with synthetic examples, the QHMCboth, QHMCsoftboth, and tnQHMC algorithms demonstrate the most accurate predictions with a relative error of $13 - 15\%$. Notably, QHMCsoftboth emerges as the fastest among the methods while achieving higher accuracy. For instance, the error value obtained by QHMCsoftboth is 0.14, whereas tnQHMC's error is 0.15. However, QHMCsoftboth delivers a 20% time efficiency gain with slightly superior accuracy. In Figure 8, a comprehensive comparison of the algorithms is presented. The decrease in relative error values is noticeable as constraints are gradually added, following the adopted adaptive approach. Initially, the error is 0.5 and gradually decreases to approximately 0.13. Furthermore, it is evident that the QHMCboth and QHMCsoftboth methods consistently deliver the most accurate results at each step, whereas the performance of the QHMCsoftvar method is outperformed by other approaches.

4.1.4 Example 4: Heat Transfer in a Hallow Sphere

This 3-dimensional example considers a heat transfer problem in a hallow sphere. Let $B_r(0)$ represent a ball centered at 0 with radius r. Defining the hallow sphere as $D = B_4(0) - B_2(0)$, the equations are

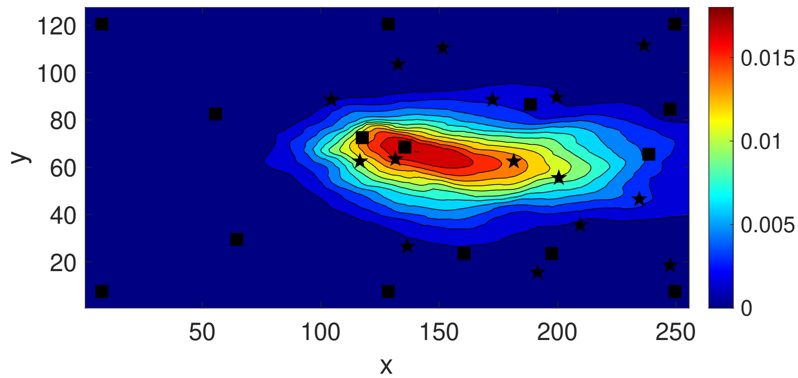

Figure 7. Observation locations (black squares) and constraint locations (black stars).

given as Yang et al. (2021)

$$
\begin{cases}
\frac{\partial u(\mathbf{x},t)}{\partial t} - \nabla \cdot (\kappa \nabla u(\mathbf{x},t)) = 0, & \mathbf{x} \in D, \\
\kappa \frac{\partial u(\mathbf{x},t)}{\partial \mathbf{n}} = \theta^2(\pi-\theta)^2\phi^2(\pi-\phi)^2, & \text{if } \|\mathbf{x}|_2 = 4 \text{ and } \phi \geq 0, \\
\kappa \frac{\partial u(\mathbf{x},t)}{\partial \mathbf{n}} = 0, & \text{if } \|\mathbf{x}\|_2 = 4 \text{ and } \phi < 0, \\
u(\mathbf{x},t) = 0, & \text{if } \|\mathbf{x}\|_2 = 2.
\end{cases}
\tag{40}
$$

In this context, $\mathbf{n}$ denotes the normal vector pointing outward, while θ and ϕ represent the azimuthal and elevation angles, respectively, of points within the sphere. We determine the precise heat conductivity using $\kappa = 1.0 + \exp(0.05u)$. The quadratic elements with 12,854 degrees of freedom are employed, and we set $y(\mathbf{x}) = u(\mathbf{x}, 10)$ to solve the partial differential equations (PDE). Starting with 6 initial locations at 0 on the surface, 6 new constraint locations are introduced based on the active learning approach of the QHMC version. In Figure 9, the decrease is evident in relative error while the constraints are added step by step. In addition, Figure 10 shows the ground truth and the GP result obtained by QHMCsoftboth algorithm, where QHMCsoftboth $y^*(x)$ matches the reference model. The constraint locations of the result are shown in Figure 11. Moreover, its posterior variance is small based on the results shown in Table 4. The table also provides the error, posterior variance and time performances of QHMC and HMC algorithms, where the advantages of QHMC over HMC in all categories, even with the truncated Gaussian algorithm are observed. Although all of the algorithms complete the GP regression in less than 1 minute, comparing the truncated Gaussian method with QHMC-based algorithms, $40-60\%$ time efficiency along with compatible accuracy of QHMC algorithms is achieved. In addition to the time and accuracy performances, it is shown that the posterior variance values are smallest with QHMCvar and QHMCboth approaches, followed by tnQHMC and QHMCad approaches. Using HMC sampling in all methods generates larger posterior variances.

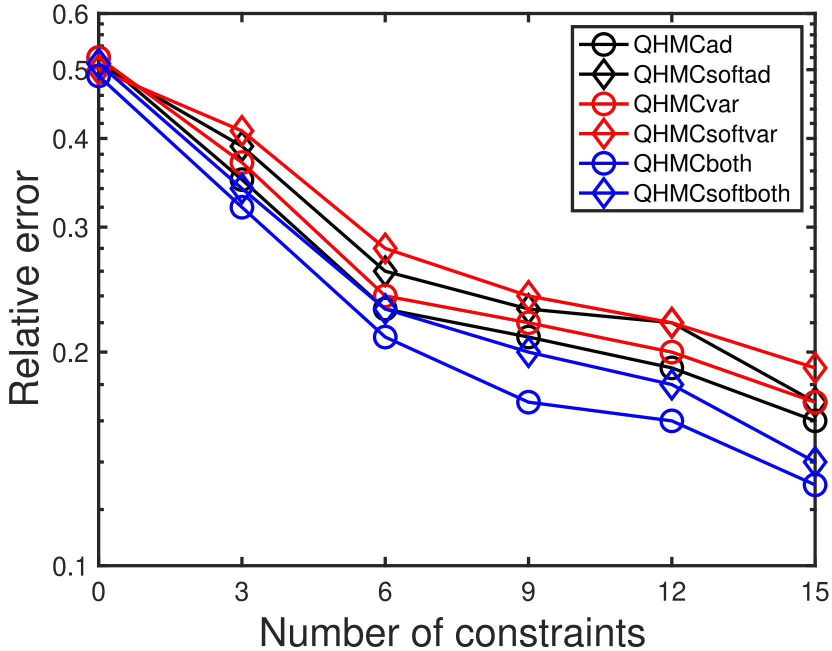

Figure 8. The change in relative error while adding constraints, solute transport.

Table 4. Comparison of QHMC and HMC on heat transfer with nonnegativity.

Method	Error	Posterior Var	Time	Method	Error	Posterior Var	Time
QHMC-ad	0.04	0.04	34s	HMC-ad	0.06	0.07	40s
QHMC-soft-ad	0.05	0.04	30s	HMC-soft-ad	0.07	0.07	32s
QHMC-var	0.05	0.02	30s	HMC-var	0.09	0.05	27s
QHMC-soft-var	0.06	0.03	26s	HMC-soft-var	0.10	0.05	29s
QHMC-both	**0.02**	0.03	32s	HMC-both	0.04	0.05	37s
QHMC-soft-both	0.03	0.03	27s	HMC-soft-both	0.05	0.06	35s
tnQHMC	0.04	0.05	51s	tnHMC	0.06	0.07	56s

4.2 Monotonicity Constraints

This section provides two numerical examples to investigate the effectiveness of our algorithms on monotonicity constraints. As stated in Section 2.3.1, the monotonicity constraints are enforced in the direction of active variables. Similar to the comparisons in previous section, we illustrate the advantages of QHMC over HMC, and then compare the performance of QHMC algorithms with additive GP approach introduced in López-Lopera et al. (2022) with respect to the same criteria as in the previous section.

4.2.1 Example 1

Consider the following 5D function with monotonicity constraints López-Lopera et al. (2022):

$$f(x) = \arctan(5x_1) + \arctan(2x_2) + x_3 + 2x_4^2 + \frac{2}{1 + \exp -10(x_5 - \frac{1}{2})}. \quad (41)$$

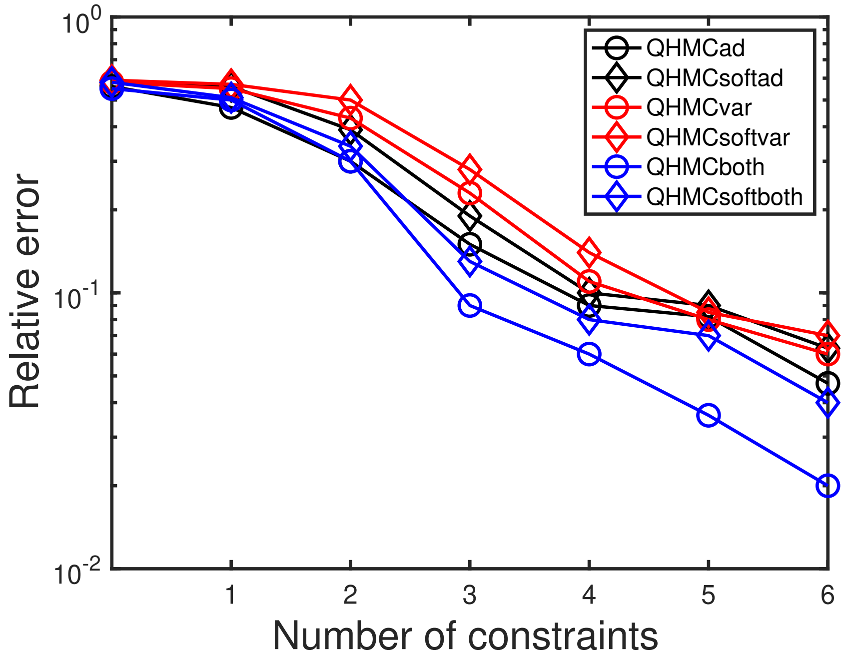

Figure 9. The change in relative error while adding constraints, heat equation.

Table 5. Comparison of QHMC and HMC on 5D, monotonicity.

Method	Error	Posterior Var	Time	Method	Error	Posterior Var	Time
QHMC-ad	0.11	0.16	2m 23s	HMC-ad	0.13	0.17	3m 14s
QHMC-soft-ad	0.14	0.18	1m 57s	HMC-soft-ad	0.17	0.20	2m 48s
QHMC-var	0.12	0.15	2m 13s	HMC-var	0.15	0.17	2m 58s
QHMC-soft-var	0.15	0.17	1m 42s	HMC-soft-var	0.18	0.19	2m 16s
QHMC-both	**0.10**	0.13	2m 25s	HMC-both	0.12	0.15	2m 58s
QHMC-soft-both	0.12	0.14	**1m 55s**	HMC-soft-both	0.14	0.15	2m 39s

Table 5 shows the performances of HMC and QHMC algorithms, where we observe that QHMC achieves higher accuracy with lower variance in a shorter amount of time. The comparison proves that each version of QHMC is more efficient than HMC In addition, Figure 12 shows the relative error values of QHMC and additive GP algorithms with respect to the change in SNR and dataset size. Based on the results, it is clear that QHMCboth and QHMCsoftboth provide the most accurate results under every different condition, while the difference is more remarkable for the cases in which noise is higher. Although QHMCboth and QHMCsoftboth provides the most accurate results, other QHMC versions also generate more accurate results then additive GP method. Moreover, Figure 13 shows that the soft-constrained QHMC approaches are faster than the hard-constrained QHMC, while hard-constrained QHMC versions are still faster than additive GP algorithm.

4.2.2 Example 2

We provide a 20-dimensional example to indicate the applicability and effectiveness of QHMC algorithms on higher dimensions with monotonicity constraint. We consider the target function used in López-Lopera

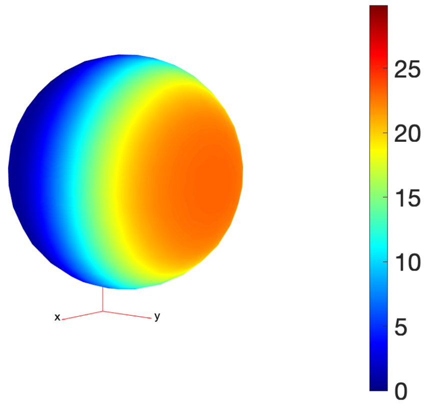

Figure 10a. Heat equation data, ground truth $y(x)$.

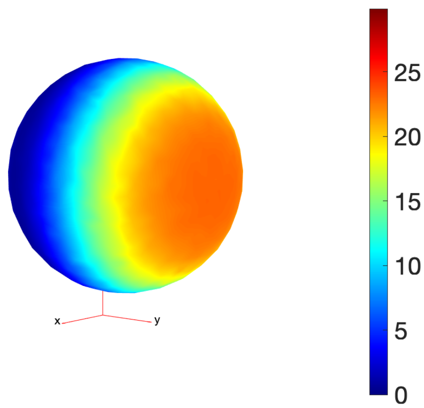

Figure 10b. QHMCsoftboth prediction $y^*(x)$.

Figure 10. Comparison of the ground truth and QHMCsoftboth result.

et al. (2022); Bachoc et al. (2022)

$$f(x_1, x_2, ..., x_d) = \sum_{i=1}^{d} \arctan 5 \left[1 - \frac{i}{d+1}\right] x_i \tag{42}$$

with $d = 20$.

Table 6 illustrates accuracy and time advantages of QHMC over HMC. For each version of QHMC and HMC, using QHMC sampling in a specific version accelerates the process while increasing the accuracy. Overall comparison shows that among all versions with QHMC and HMC sampling, QHMCboth is the most accurate approach, while QHMCsoftboth is the fastest and ranked second in accuracy. Figure 15 and Figure 16 show the relative error and time performances of QHMC-based algorithms, HMCsoftboth and additive GP algorithm, respectively. In this final example with the highest dimension, the same phenomenon is observed as in previous results: soft-constrained versions demonstrate greater efficiency,

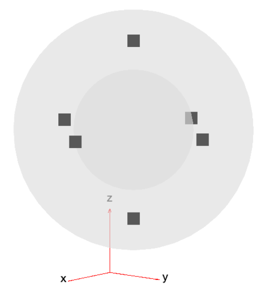

Figure 11a. Initial locations.

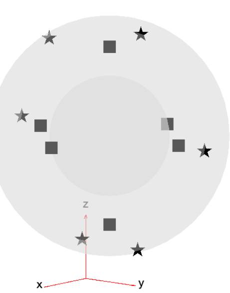

Figure 11b. Constraint locations added by QHMC.

Figure 11. Initial locations (squares) and adaptively added constraint locations (stars).

while hard-constrained QHMC approaches remain faster than additive GP across different conditions, including high noise levels. Based on Figure 15, QHMCboth can tolerate noise levels up to 10% with the smallest error, and it can still provide good accuracy (error is around 0.15) even when the SNR is higher than 10%. It is also worth to mention that although the error values generated by HMCsoftboth and additiveGP are pretty close, HMCsoftboth performs faster than additiveGP, especially when the dataset is larger and noise level is higher.

4.3 Discussion

In the scope of the proposed QHMC-based method, this work investigates the advantages and disadvantages of using soft-constrained approach on physics-informed GP regression. The comparison of modified versions of proposed algorithm along with a recent method is further performed to validate the

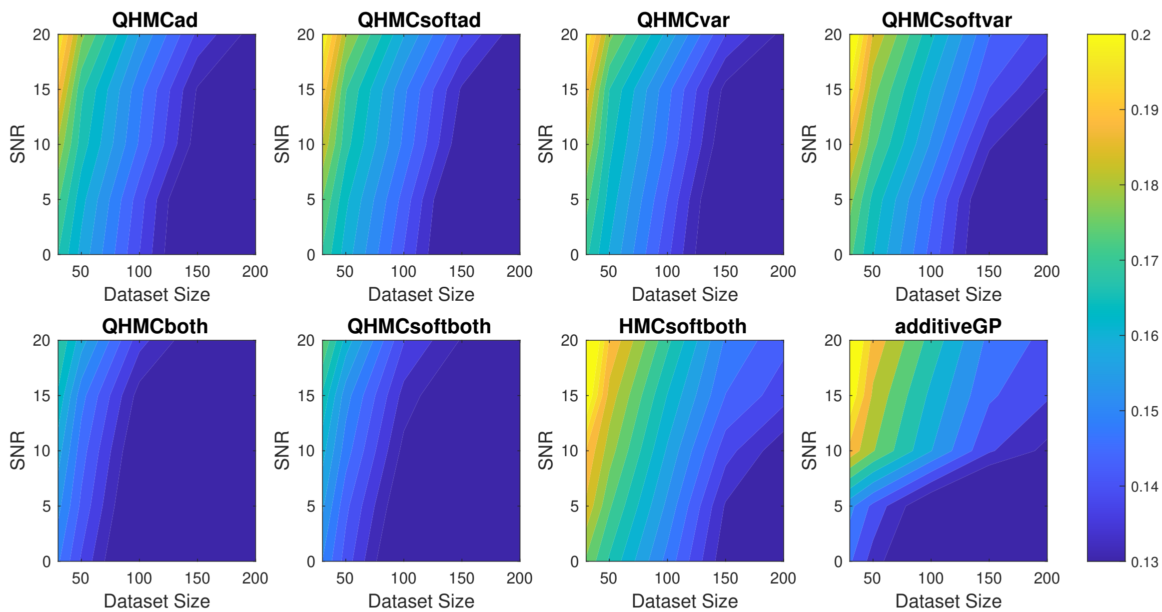

Figure 12. Relative error of the algorithms with different SNR and data sizes for Example 1 (5D), monotonicity.

Table 6. Comparison of QHMC and HMC on 20D, monotonicity.

Method	Error	Posterior Var	Time	Method	Error	Posterior Var	Time
QHMC-ad	0.13	0.18	33m 1s	HMC-ad	0.15	0.21	35m 38s
QHMC-soft-ad	0.15	0.19	31m 21s	HMC-soft-ad	0.18	0.22	33m 41s
QHMC-var	0.14	0.16	32m 53s	HMC-var	0.17	0.17	34m 21s
QHMC-soft-var	0.16	0.17	29m 42s	HMC-soft-var	0.19	0.18	31m 17s
QHMC-both	**0.11**	0.14	33m 45s	HMC-both	0.14	0.16	36m 21s
QHMC-soft-both	0.12	0.15	**29m 48s**	HMC-soft-both	0.15	0.17	33m 11s

superiority of the approach. The significant findings and the corresponding possible reasons are summarized as follows:

1. Synthetic examples are designed to highlight the robustness and efficiency of proposed method. In one example, considering two criteria: dataset size and SNR. The QHMC-based algorithms are evaluated in an environment with a range of $0 - 20\%$ SNR, and results provided in Figure 1, Figure 4 , Figure 12, and Figure 15 have shown that both soft and hard-constrained versions of proposed method tolerate the noise in the data, especially if it is less then 10%. In addition, the methods are more tolerant when the dataset size increases. This part of the experiments for each synthetic example proved the robustness of the proposed method.
2. Additionally, the numerical results of synthetic examples include the execution times for when the SNR and dataset size increase in each example. The goal is to underscore the effectiveness of the proposed algorithm. Figure 2, Figure 5, Figure 13, Figure 16 show the time advantages of the algorithms, especially for the soft-constrained versions.

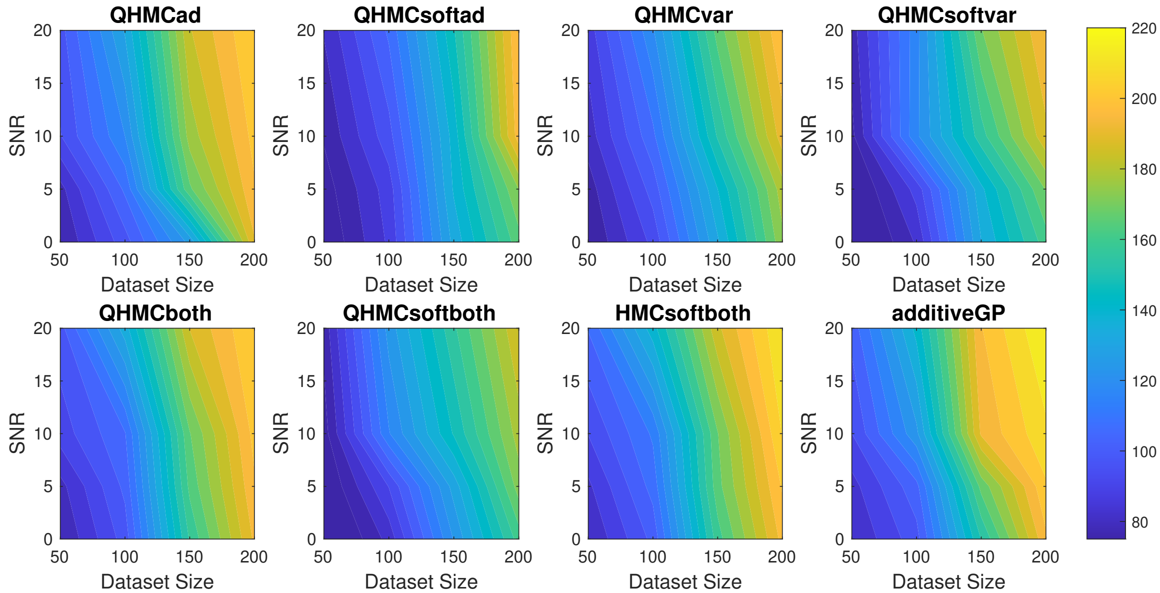

Figure 13. Execution times (in seconds) of the algorithms with different SNR and data sizes for Example 1 (5D), monotonicity.

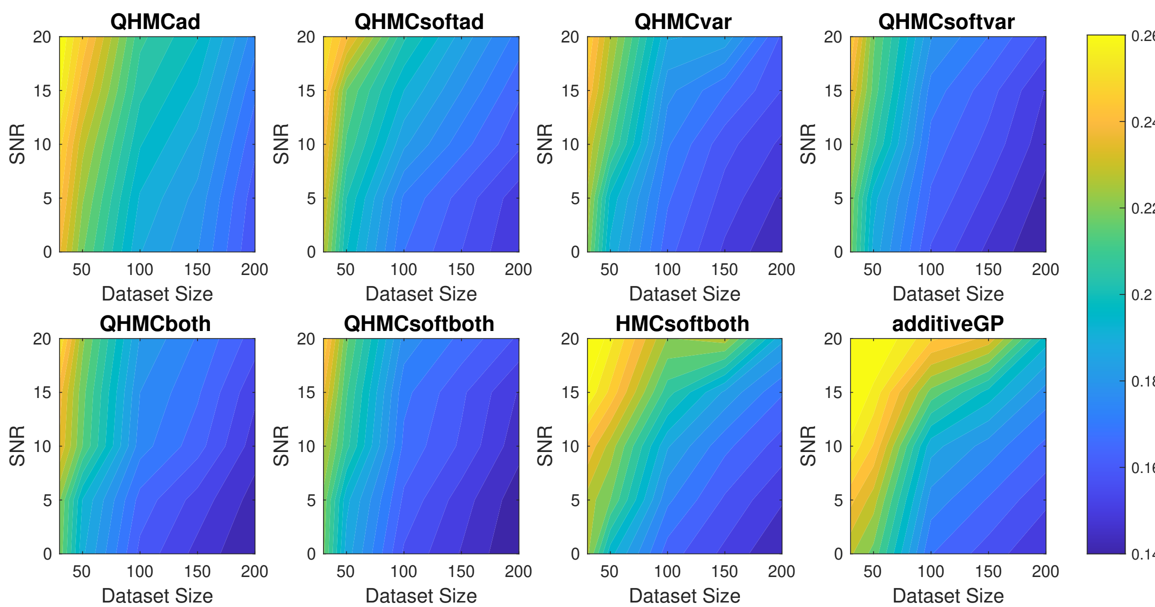

Figure 14. Posterior variances of the algorithms with different SNR and data sizes for Example 1 (5D), monotonicity.

3. The dimensions of synthetic examples are selected to verify that the robustness and efficiency of the algorithms remain for higher dimensions. For inequality-constrained scenarios, evaluations are performed on 2D and 10D problems, while for monotonicity-constrained algorithms evaluations are

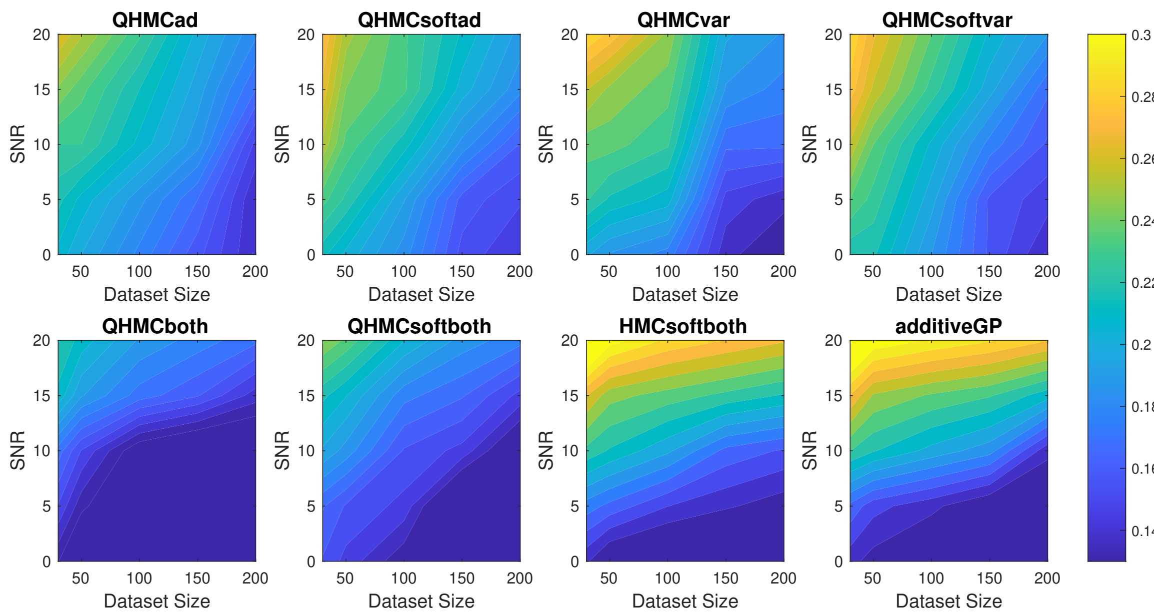

Figure 15. Relative error of the algorithms with different SNR and data sizes for Example 2 (20D), monotonicity.

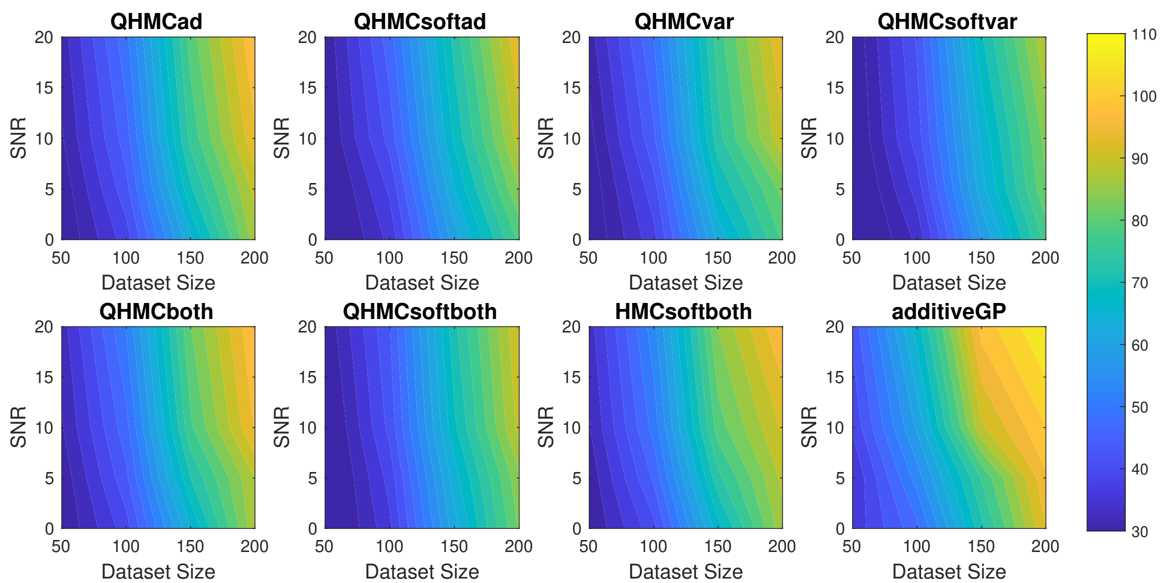

Figure 16. Execution times (in minutes) of the algorithms with different SNR and data sizes for Example 2 (20D), monotonicity.

performed on 5D and 20D problems. The results have verified that the performance of proposed methods can maintain the accuracy for higher-dimensional cases in a relatively short amount of times.

4. The real-life applications are chosen to verify that the proposed method is promising to generalize different type of problems. The solute concentration example is a 2D problem with non-homogeneous

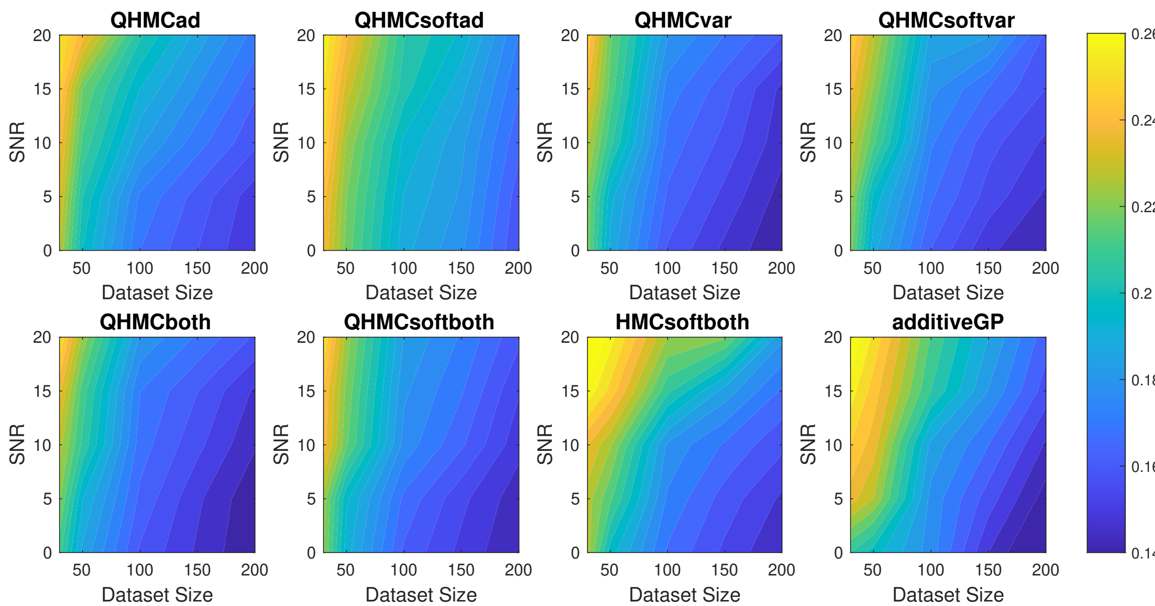

Figure 17. Posterior variances of the algorithms with different SNR and data sizes for Example 2 (20D), monotonicity.

structure, while heat transfer problem is a 3D problem that requires PDE solving. On the contrary of synthetic examples, in this set of experiments, the dataset size is fixed and there is no injected Gaussian noise in the data. We present a comprehensive comparison of all methods along with the truncated Gaussian algorithm. Step by step decrease in the error is presented in Figure 8 and Figure 9, where the success of all versions are verified.

5. The proposed method is a combination of QHMC algorithm and a probabilistic approach for phiysics-informed GP. QHMC training provides accuracy due to its broad state space exploration, while probabilistic approach lowers the variance. In each case, we start with the experiments conducted with fixed dataset size and zero SNR to demonstrate the superiority of QHMC over HMC. The HMC versions of the proposed methods are implemented and compared to the corresponding QHMC algorithms in Table 1, Table 2, Table 5, Table 6, Table 3. The findings for every single case confirm that QHMC enhances the accuracy, robustness and efficiency. After demonstrating the superiority of QHMC method, a comprehensive evaluation is performed for QHMC-based methods in different scenarios. Again, for the sake of verification of efficiency of soft-constrained QHMC, we implemented the hard constrained versions by choosing the violation probability as 0.005. The findings indicate that the soft-constrained approaches reduce computational expenses while maintaining accuracy comparable to that of the hard-constrained counterparts. Releasing the constraints by a probabilistic sense has brought efficiency, while decreasing the posterior variance.

6. We should also note that while the numerical results indicate that the current approach is a robust and efficient QHMC algorithm, the impact of the probability of constraint violation should be further investigated. The experiments were conducted with a relatively low probability of releasing the constraints (around 5%) and the accuracy was maintained under these conditions. However, allowing for more violations may pose limitations. In addition, the performance of the proposed approach on different type of constrained optimization problems, including those involving equality constraints,

can be more challenging. Addressing these challenges can be both a limitation and a potential future work for QHMC-based, physics-informed GP regression.

5 CONCLUSION

Leveraging the accuracy of QHMC training and the efficiency of probabilistic approach, this work introduced a soft-constrained QHMC algorithm to enforce inequality and monotonicity constraints on the GP. The proposed algorithm reduces the difference between ground truth and the posterior mean in the resulting GP model, while increasing the efficiency by attaining the accurate results in a short amount of time. To further enhance the performance of the QHMC algorithms across various scenarios, modified versions of QHMC are implemented adopting adaptive learning. These versions provide flexibility in selecting the most suitable algorithm based on the specific priorities of a given problem.

We provided the convergence of QHMC by showing that its steady-state distribution approach the true posterior density, and theoretically justified that the probabilistic approach preserves convergence. Finally, we have implemented our methods to solve several types of optimization problems. Each experiment initially outlined the benefits of QHMC sampling in comparison to HMC sampling. These advantages remained consistent across all cases, resulting in approximately a 20% time-saving and 15% higher accuracy. Having demonstrated the advantages of QHMC sampling, further evaluation on the performances of the algorithms across various scenarios was performed. The examples cover higher-dimensional problems featuring both inequality and monotonicity constraints. Furthermore, the evaluations include real-world applications where injecting physical properties is essential, particularly in cases involving inequality constraints.

In the context of inequality-constrained Gaussian processes (GPs), we explored 2-dimensional and 10-dimensional synthetic problems, along with two real applications involving 2-dimensional and 3-dimensional data. For synthetic examples, the relative error, posterior variance and execution time of the algorithms were compared while gradually increasing the noise level and dataset size. Overall, QHMC-based algorithms outperformed the truncated Gaussian methods. Although the truncated Gaussian methods provide high accuracy in the absence of noise and are compatible with QHMC approaches, their relative error and posterior variances increase as the noise appeared and increased. Moreover, the advantages of soft-constrained QHMC became more evident, particularly in higher-dimensional cases, when compared to truncated Gaussian and even hard-constrained QHMC. The time comparison of the algorithms underscores that the truncated Gaussian methods are significantly impacted by the curse of dimensionality and large datasets, exhibiting slower performance under these conditions. In real-world application scenarios featuring 2-dimensional and 3-dimensional data, the findings were consistent with those observed in the synthetic examples. Although the accuracy level may not reach the highest levels observed in the synthetic examples and 3-dimensional heat equation problem, the observed trend remains consistent. The lower accuracy observed in the latter problem can be attributed to the non-homogeneous structure of solute concentration.

In the case of monotonicity-constrained GP, we addressed 5-dimensional and 20-dimensional examples, utilizing the same configuration as employed for inequality-constrained GP. A comprehensive comparison was conducted between all versions of QHMC algorithms and the additive GP method. The results indicate that QHMC-based approaches hold a notable advantage, particularly in scenarios involving noise and large datasets. While additive GP proves to be a strong method suitable for high-dimensional cases, QHMC algorithms performed faster and yielded lower variances.

In conclusion, the work has demonstrated that soft-constrained QHMC is a robust, efficient and flexible method that can be applicable to higher dimensional cases and large datasets. Numerical results have shown that soft-constrained QHMC is promising to be generalized to various applications with different physical properties.

REFERENCES

Abrahamsen, P. and Benth, F. E. (2001). Kriging with inequality constraints. *Mathematical Geology* 33, 719–744

Agrell, C. (2019). Gaussian processes with linear operator inequality constraints. *arXiv preprint arXiv:1901.03134*

Bachoc, F., López-Lopera, A. F., and Roustant, O. (2022). Sequential construction and dimension reduction of gaussian processes under inequality constraints. *SIAM Journal on Mathematics of Data Science* 4, 772–800

Barbosa, B. H. G., Xu, N., Askari, H., and Khajepour, A. (2021). Lateral force prediction using Gaussian process regression for intelligent tire systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 52, 5332–5343

Barbu, A. and Zhu, S.-C. (2020). *Monte Carlo Methods*, vol. 35 (Springer)

Chati, Y. S. and Balakrishnan, H. (2017). A Gaussian process regression approach to model aircraft engine fuel flow rate. In *Proceedings of the 8th International Conference on Cyber-Physical Systems*. 131–140

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

Dürichen, R., Pimentel, M. A., Clifton, L., Schweikard, A., and Clifton, D. A. (2014). Multi-task Gaussian process models for biomedical applications. In *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)* (IEEE), 492–495

Emmanuel, S. and Berkowitz, B. (2005). Mixing-induced precipitation and porosity evolution in porous media. *Advances in water resources* 28, 337–344

Eriksson, D. and Poloczek, M. (2021). Scalable constrained Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics* (PMLR), 730–738

Ezati, M., Esmaeilbeigi, M., and Kamandi, A. (2024). Novel approaches for hyper-parameter tuning of physics-informed gaussian processes: application to parametric pdes. *Engineering with Computers* , 1–20

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2014). *Bayesian Data Analysis* (Tyler & Francis Group, Inc.)

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

Guo, S., Xu, H., and Zhang, L. (2015). Stability analysis for mathematical programs with distributionally robust chance constraint. *SIAM J. Optim (to appear)*

Hess, C. (1999). Conditional expectation and martingales of random sets. *Pattern Recognition* 32, 1543–1567

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

Kochan, D., Zhang, Z., and Yang, X. (2022). A quantum-inspired Hamiltonian Monte Carlo method for missing data imputation. In *Mathematical and Scientific Machine Learning* (PMLR), 17–32

Kohanpur, A. H., Saksena, S., Dey, S., Johnson, J. M., Riasi, M. S., Yeghiazarian, L., et al. (2023). Urban flood modeling: Uncertainty quantification and physics-informed Gaussian processes regression forecasting. *Water Resources Research* 59, e2022WR033939

Kuss, M. and Rasmussen, C. (2003). Gaussian processes in reinforcement learning. *Advances in neural information processing systems* 16

Lange-Hegermann, M. (2021). Linearly constrained Gaussian processes with boundary conditions. In *International Conference on Artificial Intelligence and Statistics* (PMLR), 1090–1098

Li, X.-Q., Song, L.-K., Choy, Y.-S., and Bai, G.-C. (2023). Fatigue reliability analysis of aeroengine blade-disc systems using physics-informed ensemble learning. *Philosophical Transactions of the Royal Society A* 381, 20220384

Lin, G. and Tartakovsky, A. M. (2009). An efficient, high-order probabilistic collocation method on sparse grids for three-dimensional flow and solute transport in randomly heterogeneous porous media. *Advances in Water Resources* 32, 712–722

Liu, Z. and Zhang, Z. (2019). Quantum-inspired Hamiltonian Monte Carlo for Bayesian sampling. *arXiv preprint arXiv:1912.01937*

López-Lopera, A., Bachoc, F., and Roustant, O. (2022). High-dimensional additive Gaussian processes under monotonicity constraints. *Advances in Neural Information Processing Systems* 35, 8041–8053

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

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

Maatouk, H., Roustant, O., and Richet, Y. (2015). Cross-validation estimations of hyper-parameters of Gaussian processes with inequality constraints. *Procedia Environmental Sciences* 27, 38–44

Mao, Z., Jagtap, A. D., and Karniadakis, G. E. (2020). Physics-informed neural networks for high-speed flows. *Computer Methods in Applied Mechanics and Engineering* 360, 112789

Nabati, M., Ghorashi, S. A., and Shahbazian, R. (2022). Jgpr: a computationally efficient multi-target Gaussian process regression algorithm. *Machine Learning* 111, 1987–2010

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

Pimentel, M. A., Clifton, D. A., Clifton, L., and Tarassenko, L. (2013). Probabilistic estimation of respiratory rate using gaussian processes. In *2013 35th annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (IEEE), 2902–2905

Qiang, B., Shi, K., Liu, N., Ren, J., and Shi, Y. (2023). Integrating physics-informed recurrent Gaussian process regression into instance transfer for predicting tool wear in milling process. *Journal of Manufacturing Systems* 68, 42–55

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

Rasmussen, C. E., Williams, C. K., et al. (2006). *Gaussian processes for machine learning*, vol. 1 (Springer)

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

Salzmann, M. and Urtasun, R. (2010). Implicitly constrained Gaussian process regression for monocular non-rigid pose estimation. *Advances in neural information processing systems* 23

Song, F., Shi, K., and Li, K. (2021). A nonlinear finite element-based supervised machine learning approach for efficiently predicting collapse resistance of wireline tool housings subjected to combined loads. In *ASME International Mechanical Engineering Congress and Exposition* (American Society of Mechanical Engineers), vol. 85680, V012T12A057

Stein, M. L. (1988). Asymptotically efficient prediction of a random field with a misspecified covariance function. *The Annals of Statistics* 16, 55–63

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

Wang, Y., Wang, Z., Han, K., Tiwari, P., and Work, D. B. (2021). Personalized adaptive cruise control via Gaussian process regression. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)* (IEEE), 1496–1502

Wilkie, D. and Galasso, C. (2021). Gaussian process regression for fatigue reliability analysis of offshore wind turbines. *Structural Safety* 88, 102020

Williams, C., Klanke, S., Vijayakumar, S., and Chai, K. (2008). Multi-task Gaussian process learning of robot inverse dynamics. *Advances in neural information processing systems* 21

Yang, X., Barajas-Solano, D., Tartakovsky, G., and Tartakovsky, A. M. (2019). Physics-informed cokriging: A Gaussian-process-regression-based multifidelity method for data-model convergence. *Journal of Computational Physics* 395, 410–431

Yang, X., Tartakovsky, G., and Tartakovsky, A. M. (2021). Physics information aided kriging using stochastic simulation models. *SIAM Journal on Scientific Computing* 43, A3862–A3891

Zhang, H. (2004). Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics. *Journal of the American Statistical Association* 99, 250–261