GAUSSIAN PROCESS EMULATORS FOR COMPUTER EXPERIMENTS WITH INEQUALITY CONSTRAINTS

Hassan Maatouk, Xavier Bay

GAUSSIAN PROCESS EMULATORS FOR COMPUTER EXPERIMENTS WITH INEQUALITY CONSTRAINTS

Hassan Maatouk[†] and Xavier Bay[†]

Abstract. Physical phenomena are observed in many fields (sciences and engineering) and are often studied by time-consuming computer codes. These codes are analyzed with statistical models, often called emulators. In many situations, the physical system (computer model output) may be known to satisfy inequality constraints with respect to some or all input variables. Our aim is to build a model capable of incorporating both data interpolation and inequality constraints into a Gaussian process emulator. By using a functional decomposition, we propose to approximate the original Gaussian process by a finite-dimensional Gaussian process such that all conditional simulations satisfy the inequality constraints in the whole domain. The mean, mode (maximum a posteriori) and prediction intervals (uncertainty quantification) of the conditional Gaussian process are calculated. To investigate the performance of the proposed model, some conditional simulations with inequality constraints such as boundary, monotonicity or convexity conditions are given.

Key words. Gaussian process emulator, inequality constraints, finite-dimensional Gaussian process, splines, uncertainty quantification, design and modeling of computer experiments

AMS subject classifications. 60G15, 60G25

1. Introduction.

In the engineering activity, runs of a computer code can be expensive and time-consuming. One solution is to use a statistical surrogate for conditioning computer model outputs at some input locations (design points). Gaussian process (GP) emulator is one of the most popular choices [23]. The reason comes from the property of the GP that uncertainty quantification can be calculated. Furthermore, it has several nice properties. For example, the conditional GP at observation data (linear equality constraints) is still a GP [5]. Additionally, some inequality constraints (such as monotonicity and convexity) of output computer responses are related to partial derivatives. In such cases, the partial derivatives of the GP are also GPs. Incorporating an infinite number of linear inequality constraints into a GP emulator, the problem becomes more difficult. The reason is that the resulting conditional process is not a GP.

In the literature of interpolation with inequality constraints, we find two types of methods. The first one is deterministic and based on splines, which have the advantage that the inequality constraints are satisfied in the whole input domain (see e.g. [16], [24] and [25]). The second one is based on the simulation of the conditional GP by using the subdivision of the input set (see e.g. [1], [6] and [11]). In that case, the inequality constraints are satisfied in a finite number of input locations. Notice that the advantage of such method is that uncertainty quantification can be calculated. In previous work, some methodologies have been based on the knowledge of the derivatives of the GP at some input locations ([11], [21] and [26]). For monotonicity constraints with noisy data, a Bayesian approach was developed in [21]. In [11] the problem is to build a GP emulator by using the prior monotonicity infor-

[†]Ecole Nationale Supérieure des Mines de St-Etienne, 158 Cours Fauriel, Saint-Etienne, France. (maatouk,bay@emse.fr).

mation of the computer model response with respect to some inputs. Their idea is based on an approach similar to [21] placing the derivatives information at specified input locations, by forcing the derivative process to be positive at these points. In such methodology, the monotonicity constraint is not guaranteed in the whole domain. Recently, a methodology based on a discrete-location approximation for incorporating inequality constraints into a GP emulator was developed in [6]. Again, the inequality constraints are not guaranteed in the whole domain.

On the other hand, Villalobos and Wahba [24] used splines to estimate an interpolation smooth function satisfying a finite family of linear inequality constraints. In term of estimation of monotone smoothing functions, using B-splines was firstly introduced by Ramsay [19]. The idea is based on the integration of B-splines defined on a properly set of knots with positive coefficients to ensure the monotonicity constraint. A similar approach is applied to econometrics in [7]. Xuming [27] takes the same approach and suggests the calculation of the coefficients by solving a finite linear minimization problem. A comparison to monotone kernel regression and an application to decreasing constraints are included.

Our aim in this paper is to build a GP emulator incorporating the advantage of splines approach in order to ensure that the inequality constraints are satisfied in the whole input domain. We propose to approximate the original GP by a finite-dimensional GP that converges uniformly to the original one. It is constructed by incorporating Gaussian random coefficients and deterministic basis functions. We show that the basis functions can be chosen such that the inequality constraints of the GP are equivalent to constraints on the coefficients. Therefore the inequality constraints are reduced to a finite number of constraints. Furthermore, any posterior sample of coefficients leads to an interpolating function satisfying the inequality constraints in the whole domain. Finally, the problem is reduced to simulate a Gaussian vector (random coefficients) restricted to convex sets which is a well-known problem with existing algorithms (see e.g. [2], [4], [8], [9],[10], [15], [18] and [22]).

The article is structured as follows: in §2, we briefly recall Gaussian process modeling for computer experiments and the choice of covariance functions. In §3, we propose a finite-dimensional GP capable of interpolating computer model outputs and incorporating inequality constraints in the whole domain, and we investigate its properties. In §4, we show some simulated examples of the conditional GP with inequality constraints (such as boundary, monotonicity or convexity conditions) in one and two dimensions. We end up this paper by some concluding remarks and future work.

2. Gaussian process emulators for computer experiments. We consider the model $y = f(\boldsymbol{x})$, where the simulator response y is assumed to be a deterministic real-valued function of the d-dimensional variable $\boldsymbol{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d$. We suppose that the real function is continuous and evaluated at n design points given by the rows of the $n \times d$ matrix $\boldsymbol{X} = (\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)})^\top$, where $\boldsymbol{x}^{(i)} \in \mathbb{R}^d$, $1 \leq i \leq n$. In practice, the evaluation of the function is expensive and must be considered highly time-consuming. The solution is to estimate the unknown function f by using a GP emulator also known as "Kriging". In this framework, y is viewed as a realization of a continuous GP,

$$Y(\boldsymbol{x}) := \eta(\boldsymbol{x}) + Z(\boldsymbol{x}), \tag{2.1}$$

where the deterministic continuous function $\eta : \boldsymbol{x} \in \mathbb{R}^d \longrightarrow \eta(\boldsymbol{x}) \in R$ is the mean and Z is a centered GP with continuous covariance function

$$K : (\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}^d \times \mathbb{R}^d \longrightarrow K(\boldsymbol{u}, \boldsymbol{v}) \in \mathbb{R}. \tag{2.2}$$

Conditionally to the observation $\boldsymbol{y} = (y(x^{(1)}), \ldots, y(x^{(n)}))^\top$ the process is still a GP:

$$Y(\boldsymbol{x}) \big| Y(\boldsymbol{X}) = \boldsymbol{y} \sim \mathcal{N}\left(\zeta(\boldsymbol{x}), \tau^2(\boldsymbol{x})\right), \tag{2.3}$$

where

$$\begin{aligned} \zeta(\boldsymbol{x}) &= \eta(\boldsymbol{x}) + \boldsymbol{k}(\boldsymbol{x})^\top \mathbb{K}^{-1}(\boldsymbol{y} - \boldsymbol{\mu}) \\ \tau^2(\boldsymbol{x}) &= K(\boldsymbol{x}, \boldsymbol{x}) - \boldsymbol{k}(\boldsymbol{x})^\top \mathbb{K}^{-1} \boldsymbol{k}(\boldsymbol{x}) \end{aligned}$$

and $\boldsymbol{\mu} = \eta(\boldsymbol{X})$ is the vector of trend values at the experimental design points, $\mathbb{K}_{i,j} = K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right)$, $i, j = 1, \ldots, n$ is the covariance matrix of $Y(\boldsymbol{X})$, and $\boldsymbol{k}(\boldsymbol{x}) = \left(K\left(\boldsymbol{x}, \boldsymbol{x}^{(i)}\right)\right)$ is the vector of covariance between $Y(\boldsymbol{x})$ and $Y(\boldsymbol{X})$. The mean $\zeta(\boldsymbol{x})$ is called Simple Kriging (SK) mean prediction of $Y(\boldsymbol{x})$ based on the computer model outputs $Y(\boldsymbol{X}) = \boldsymbol{y}$, [13].

2.1. The choice of covariance function. The choice of K has crucial consequences specially in controlling the smoothness of the Kriging metamodel. It must be chosen in the set of definite and positive kernels. Some popular kernels are the Gaussian kernel, Matérn kernel (with parameter $\lambda = 3/2, 5/2, \ldots$) and exponential kernel (Matérn kernel with parameter $\lambda = 1/2$). Notice that these kernels are placed in order of smoothness, the Gaussian kernel corresponding to C^∞ function and the exponential kernel to continuous one (see [20]). In the running examples of this paper, we will consider the Gaussian kernel defined by

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \Pi_{k=1}^d e^{-\frac{(x_k - x'_k)^2}{2\theta_k}}, \tag{2.4}$$

for all x, $x' \in \mathbb{R}^d$, where σ^2 and $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)$ are parameters.

2.2. Derivatives of Gaussian processes. In this paragraph we assume that the paths of $Y(x)$ are of class C^p. This can be guaranteed if K is smooth enough, and in particular if K is of class C^∞ (see [5]). The linearity of the differentiation operation ensures that the order partial derivatives of a GP are also GPs [5], with (see e.g. [17]):

$$E\left(\partial^p_{x_k} Y(\boldsymbol{x})\right) = \frac{\partial^p}{\partial x_k^p} E\left(Y(\boldsymbol{x})\right), \tag{2.5}$$

$$Cov\left(\partial^p_{x_k} Y(\boldsymbol{x}^{(i)}), \partial^q_{x_l} Y(\boldsymbol{x}^{(j)})\right) = \frac{\partial^{p+q}}{\partial x_k^p \partial (x'_l)^q} K\left(\boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right). \tag{2.6}$$

3. Gaussian process emulators with inequality constraints. In this section we assume that the real function (physical system) may be known to satisfy inequality constraints (such as boundary, monotonicity or convexity conditions) in the whole domain. Our aim is to incorporate both interpolation conditions and inequality constraints into a Gaussian process emulator.

3.1. Formulation of the problem. Without loss of generality the input $\boldsymbol{x}$ is in $[0,1]^d \subset \mathbb{R}^d$. We assume that the real function f is evaluated at n distinct locations in the input set,

$$f(\boldsymbol{x}^{(i)}) = y_i, \qquad i = 1, \dots, n. \tag{3.1}$$

Let $(Y(\boldsymbol{x}))_{\boldsymbol{x}\in[0,1]^d}$ be a centered GP with covariance function K and $C([0,1]^d)$ the space of continuous function on $[0,1]^d$. We denote by $\mathcal{C}$ the subset of $C([0,1]^d)$ corresponding to a given set of inequality constraints. We aim to get the conditional distribution of Y given interpolation conditions and linear inequality constraints respectively as

$$\begin{aligned} &Y\left(\boldsymbol{x}^{(i)}\right) = y_i, \qquad i = 1, \dots, n, \\ &Y \in \mathcal{C}. \end{aligned} \tag{3.2}$$

3.2. Gaussian process approximation. To handle the conditional distribution incorporating both interpolation conditions and inequality constraints, we propose to approximate the original GP Y by a finite-dimensional process of the form

$$Y^N(\boldsymbol{x}) := \sum_{j=0}^{N} \xi_j \phi_j(\boldsymbol{x}), \tag{3.3}$$

where $\xi = (\xi_0, \dots, \xi_N)^\top$ is a centered Gaussian vector with covariance matrix Γ^N and $\phi = (\phi_0, \dots, \phi_N)^\top$ is a vector of basis functions. The choice of these basis functions and Γ^N depend on the type of the inequality constraints. Notice that Y^N is a centered GP with covariance function

$$K_N(\boldsymbol{x}, \boldsymbol{x}') = \phi(\boldsymbol{x})^\top \Gamma^N \phi(\boldsymbol{x}'). \tag{3.4}$$

The advantage of the proposed model (3.3) is that the original problem is reduced to simulate the Gaussian vector ξ given that

$$\sum_{j=0}^{N} \xi_j \phi_j(\boldsymbol{x}^{(i)}) = y_i, \qquad i = 1, \dots, n, \tag{3.5}$$

$$\xi \in \mathcal{C}_\xi, \tag{3.6}$$

where $\mathcal{C}_\xi = \left\{ \boldsymbol{c} \in \mathbb{R}^{N+1} \ : \ \sum_{j=0}^{N} c_j \phi_j \in \mathcal{C}, \ \boldsymbol{c} = (c_0, \dots, c_N)^\top \right\}$. Hence the problem is equivalent to simulate a truncated Gaussian vector restricted to (3.5) and (3.6).

3.3. One dimensional cases.

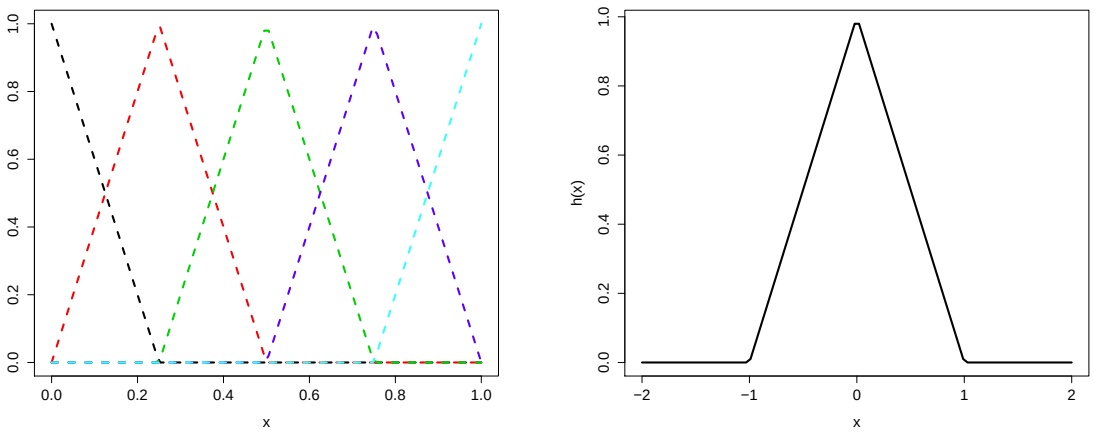
3.3.1. Boundary constraint. We assume that the real function defined in the unit interval is continuous and bounded ($a \leq f(x) \leq b,\ x \in [0,1]$), where $-\infty \leq a < b \leq +\infty$. In that case, the convex set $\mathcal{C}$ is the space of bounded functions and is defined as

$$\mathcal{C} := \{f \in C([0,1]),\ a \leq f(x) \leq b,\ x \in [0,1]\}\,. \tag{3.7}$$

Let us begin by constructing the functions $h_j,\ j = 0, \ldots, N$ that will be used in the proposed model. We first descretize the input set as $0 = u_0 < u_1 < \ldots < u_N = 1$, and on each knot we build a function. For the sake of simplicity, we use a uniform subdivision of the input set but the methodology can be adapted for any subdivision. For example at the j^{th} knot $u_j = j\delta = j/N$ the associated function is

$$h_j(x) = h\left(\frac{x - u_j}{\delta}\right), \qquad j = 0, \ldots, N, \tag{3.8}$$

where $h(x) := (1 - |x|)\,\mathbb{1}_{(|x| \leq 1)},\ x \in \mathbb{R}$, see Figures 1a and 1b below for $N = 4$. Notice that the h_j's are bounded between 0 and 1 and $\sum_{j=0}^{N} h_j(x) = 1$ for all x in $[0,1]$. Additionally, the value of these functions at any knot $u_i,\ i = 0, \ldots, N$ is equal to the delta function ($h_j(u_i) = \delta_{ij}$), where δ_{ij} is equal to one if $i = j$ and zero otherwise.



(a) (b)

Figure 1: The basis functions $h_j, (0 \leq j \leq 4)$ (Figure 1a) and the function h (Figure 1b).

The philosophy of the proposed method is presented in the following proposition:

Proposition 3.1. *With the notations introduced before, we define the finite-dimensional pro-*

cess $(Y^N(x))_{x\in[0,1]}$ *as*

$$Y^N(x) = \sum_{j=0}^{N} Y(u_j)h_j(x) = \sum_{j=0}^{N} \xi_j h_j(x), \tag{3.9}$$

where we denote $\xi_j = Y(u_j),\ j = 0, \cdots, N$. *Then we have the following properties:*

- Y^N *is a finite-dimensional GP with covariance function* $K_N(x, x') = h(x)^\top \Gamma^N h(x')$, *where* $h(x) = (h_0(x), \ldots, h_N(x))^\top$, $\Gamma^N_{i,j} = K(u_i, u_j),\ i, j = 0, \ldots, N$ *and* K *the covariance function of the original GP* Y.
- Y^N *converges uniformly to* Y *when* N *tends to infinity (with probability 1).*
- K_N *converges uniformly to* K *when* N *tends to infinity.*
- Y^N *is in* $\mathcal{C}$ *if and only if the* $(N + 1)$ *coefficients* $Y(u_j)$ *are contained in* $[a, b]$.

The advantage of this model is that the infinite number of inequality constraints of Y^N are reduced to a finite number of constraints on the coefficients $(Y(u_j))_{0\le j\le N}$. Therefore the problem is reduced to simulate the Gaussian vector $\xi = (Y(u_0), \ldots, Y(u_N))^\top$ restricted to the convex subset of $\mathbb{R}^{N+1}$, $\mathcal{C}_\xi = [a, b]^{N+1}$.

Proof. [Proof of Proposition 3.1] Since $Y(u_j),\ j = 0, \ldots, N$ are Gaussian variables, then Y^N is a GP with dimension equal to $N + 1$ and covariance function

$$cov\left(Y^N(x), Y^N(x')\right) = \sum_{i,j=0}^{N} cov\left(Y(u_i), Y(u_j)\right) h_i(x)h_j(x') = \sum_{i,j=0}^{N} K(u_i, u_j)h_i(x)h_j(x'). \tag{3.10}$$

Denote by $E^N := \text{span}\{h_0, \ldots, h_N\}$ the approximation space spanned by the functions h_j, $0 \le j \le N$. Let f_N be a function defined as

$$f_N(x) = \sum_{j=0}^{N} f(u_j)h_j(x) \in E^N, \tag{3.11}$$

where f is any continuous function. Hence the piecewise linear function f_N converges uniformly to f when N tends to infinity. This is a consequence of the fact that a continuous function is uniformly continuous on a compact set. Consequently, the second property is done as the GP Y is assumed to be continuous. From (3.10) and by a similar argument to show the uniform convergence of f_N to f, we get the third property. Now observe that $\sum_{j=0}^{N} h_j(x) = 1,\ x \in [0, 1]$. If the $(N + 1)$ coefficients $Y(u_j)_{0\le j\le N}$ are in the interval $[a, b]$ then Y^N is in $\mathcal{C}$. Conversely, suppose that Y_N is in $\mathcal{C}$ then

$$Y^N(u_i) = \sum_{j=0}^{N} Y(u_j)h_j(u_i) = \sum_{j=0}^{N} Y(u_j)\delta_{ij} = Y(u_i) \in [a, b], \tag{3.12}$$

$i = 0, \ldots, N$, which completes the proof of the last property, and hence concludes the proof of the proposition. ■

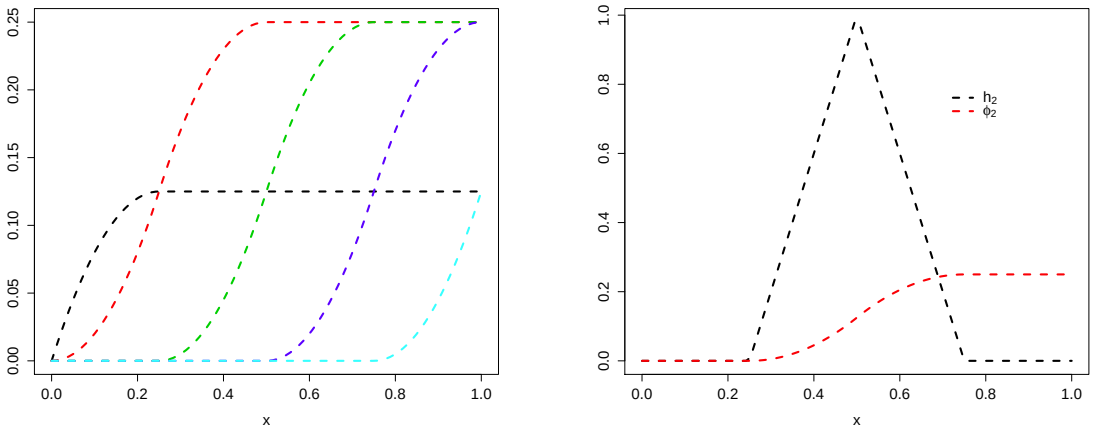
3.3.2. Monotonicity constraint. In this section the real function f is assumed to be of class C^1. The convex set $\mathcal{C}$ is the space of non-decreasing functions and is defined as

$$\mathcal{C} := \left\{ f \in C^1([0,1]) \ : \ f'(x) \geq 0, \ x \in [0,1] \right\}. \tag{3.13}$$

Since the monotonicity is related to the sign of the derivative then the proposed model is adapted from model (3.9). The basis functions are defined as the primitive functions of h_j,

$$\phi_j(x) := \int_0^x h_j(t)dt, \qquad x \in [0,1]. \tag{3.14}$$

Remark that the derivative of the basis functions ϕ_j at any knot u_i, $i = 0, \ldots, N$ is equal to the delta function ($\phi_j'(u_i) = \delta_{ij}$). In Figure 2a we illustrate the basis functions ϕ_j, $(0 \leq j \leq 4)$. Notice that all these functions are non-decreasing and starting from 0. In Figure 2b we plot the basis function ϕ_2 and the associate function h_2 for $N = 4$.



(a) (b)

Figure 2: The basis functions ϕ_j, $(0 \leq j \leq 4)$ (Figure 2a) and the function h_2 with the corresponding function ϕ_2 (Figure 2b).

Similarly to Proposition 3.1, we have the following results.

Proposition 3.2. *Using the notations introduced before, we define the finite-dimensional process* $(Y^N(x))_{x\in[0,1]}$ *as*

$$Y^N(x) = Y(0) + \sum_{j=0}^{N} Y'(u_j)\phi_j(x) = \zeta + \sum_{j=0}^{N} \xi_j \phi_j(x), \tag{3.15}$$

where we denote $\zeta = Y(0)$ *and* $\xi_j = Y'(u_j),\ j = 0, \cdots, N$. *Then we have the following properties:*

- Y^N *is a finite-dimensional GP with covariance function* $K_N(x, x') = \phi(x)^\top \Gamma^N \phi(x')$, *where* $\phi(x) = (\phi_0(x), \ldots, \phi_N(x))^\top$, $\Gamma^N_{i,j} = \frac{\partial^2 K}{\partial x \partial x'}(u_i, u_j),\ i, j = 0, \ldots, N$ *and* K *the covariance function of the original GP* Y.
- Y^N *converges uniformly to* Y *when* N *tends to infinity (with probability 1).*
- K_N *converges uniformly to* K *when* N *tends to infinity.*
- Y^N *is non-decreasing if and only if the coefficients* $(Y'(u_j))_{0 \leq j \leq N}$ *are all non-negative.*

Consequently, the $(N+2) \times (N+2)$ *covariance matrix* Γ^N_{new} *of the Gaussian vector* $(\zeta, \xi)^\top = (Y(0), Y'(u_0), \ldots, Y'(u_N))$ *is equal to*

$$\Gamma^N_{new} = \begin{bmatrix} K(0,0) & \frac{\partial K}{\partial x'}(0, u_i) \\ \frac{\partial K}{\partial x}(u_j, 0) & \Gamma^N_{i,j} \end{bmatrix}_{0 \leq i,j \leq N}.$$

From the last property, the problem is reduced to simulate the Gaussian vector $(\zeta, \xi)^\top$ restricted to the non-negative quadrant, $\left(\mathcal{C}_\xi = \left\{\xi \in \mathbb{R}^{N+1} \ :\ \xi_j \geq 0,\ j = 0, \ldots, N\right\}\right)$.

Proof. The first property is a consequence of the fact that the derivative of a GP is also a GP. We have for all $x, x' \in [0, 1]$,

$$K_N(x, x') = cov\left(Y^N(x), Y^N(x')\right) = \sum_{i,j=0}^{N} \frac{\partial^2 K}{\partial x \partial x'}(u_i, u_j)\phi_i(x)\phi_j(x'). \tag{3.16}$$

Let f and f_N be two functions of class C^1 given respectively as

$$f(x) = f(0) + \int_0^x f'(t)dt, \tag{3.17}$$

$$f_N(x) = f(0) + \int_0^x f'_N(t)dt. \tag{3.18}$$

The convergence of f_N to f is done by applying Proposition 3.1 to the following,

$$|f_N(x) - f(x)| = \left|\int_0^x f'_N(t)dt - \int_0^x f'(t)dt\right| = \left|\int_0^x (f'_N(t) - f'(t))dt\right| \leq \|f'_N - f'\|_\infty, \tag{3.19}$$

and $\|f_N - f\|_\infty \leq \|f'_N - f'\|_\infty$. The convergence of K_N to K is a consequence of (3.16). Now if $Y'(u_j),\ j = 0, \ldots, N$ are all non-negative then Y^N is non-decreasing since the basis functions $(\phi_j)_{0 \leq j \leq N}$ are non-decreasing. Conversely, if Y^N is non-decreasing, we have

$$0 \leq (Y^N)'(u_i) = \sum_{j=0}^{N} Y'(u_j)h_j(u_i) = Y'(u_j), \tag{3.20}$$

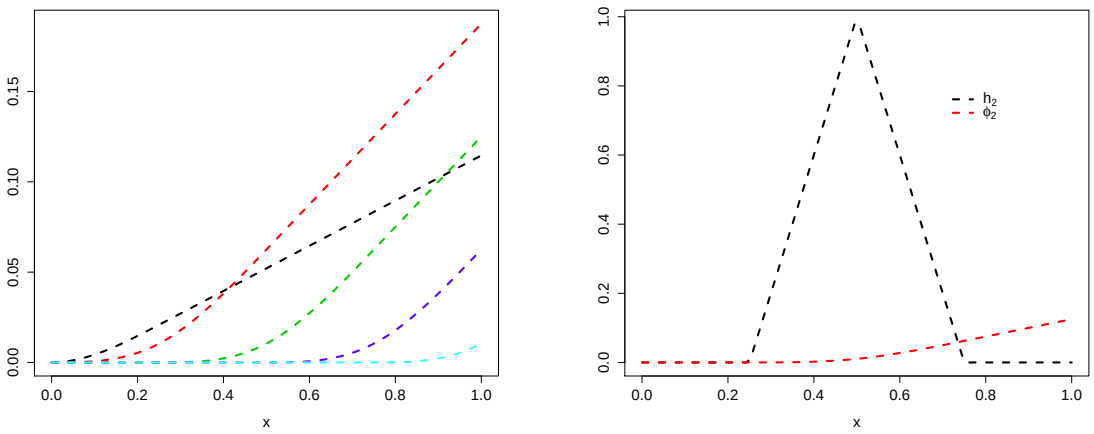
$i = 0, \ldots, N$, which completes the proof of the last property and the proposition. ■

Remark 3.3 (Monotonicity of continuous but non-derivable functions). If the real function is of class C^0 only (but possibly not derivable) and non-decreasing in the whole domain, then the proposed model defined in (3.9) is non-decreasing *if and only if* the sequence of coefficients $(Y(u_j))_j,\ j = 0, \ldots, N$ is non-decreasing (i.e. $Y(u_{j-1}) \leq Y(u_j),\ j = 1, \ldots, N$).

3.3.3. Convexity constraint. In this section the real function is supposed to be two times differentiable. Since the functions $h_j,\ j = 0, \ldots, N$ defined in (3.8) are all non-negative, then the basis functions ϕ_j are taken as the two times primitive functions of h_j,

$$\phi_j(x) := \int_0^x \left(\int_0^t h_j(u)du \right) dt. \tag{3.21}$$

In Figure 3a we illustrate the basis functions $\phi_j,\ (0 \leq j \leq 4)$. Notice that all these functions are convex and pass through the origin. Moreover, the derivatives at the origin are equal to zero. In Figure 3b we illustrate the basis function ϕ_2 and the associate function h_2.



(a) (b)

Figure 3: The basis functions $\phi_j,\ (0 \leq j \leq 4)$ (Figure 3a) and the function h_2 with the corresponding function ϕ_2 (Figure 3b).

Similarly to the monotonicity case, the second derivative of the basis functions ϕ_j'' at any knot $u_i,\ (0 \leq i \leq N)$ is equal to delta function $(\phi_j''(u_i) = \delta_{ij})$. Consequently, the proposed GP defined as

$$Y^N(x) = Y(0) + xY'(0) + \sum_{j=0}^{N} Y''(u_j)\phi_j(x) = \zeta + \kappa x + \sum_{j=0}^{N} \xi_j \phi_j(x), \tag{3.22}$$

is convex *if and only if* the $(N+1)$ random coefficients $\xi_j = Y''(u_j)$ are all non-negative, where we denote $\zeta = Y(0)$ and $\kappa = Y'(0)$. Thus the problem is simplified to generate the Gaussian vector $(\zeta, \kappa, \xi)^\top$ restricted to the convex set $\mathcal{C}_\xi$ (non-negative quadrant), $\big(\mathcal{C}_\xi = \big\{\xi \in \mathbb{R}^{N+1} \ : \ \xi_j \geq 0, \ j = 0, \dots, N\big\}\big)$. Its $(N+3) \times (N+3)$ covariance matrix Γ^N_{new} is given by

$$\Gamma^N_{\text{new}} = \begin{bmatrix} var(\zeta) & cov(\zeta,\kappa) & cov(\zeta,\xi) \\ cov(\kappa,\zeta) & var(\kappa) & cov(\kappa,\xi) \\ cov(\xi,\zeta) & cov(\xi,\kappa) & cov(\xi,\xi) \end{bmatrix} = \begin{bmatrix} K(0,0) & \frac{\partial K}{\partial x'}(0,0) & \frac{\partial^2 K}{\partial (x')^2}(0,u_i) \\ \frac{\partial K}{\partial x}(0,0) & \frac{\partial^2 K}{\partial x \partial x'}(0,0) & \frac{\partial^3 K}{\partial x \partial (x')^2}(0,u_i) \\ \frac{\partial^2 K}{\partial x^2}(u_j,0) & \frac{\partial^3 K}{\partial x^2 \partial x'}(u_j,0) & \Gamma^N_{i,j} \end{bmatrix}_{0 \leq i,j \leq N},$$

where

$$\Gamma^N_{i,j} = cov(\xi_i, \xi_j) = cov(Y''(u_i), Y''(u_j)) = \frac{\partial^4 K}{\partial x^2 \partial (x')^2}(u_i, u_j), \qquad i,j = 0, \dots, N. \tag{3.23}$$

Now we consider the problem dimension $d \geq 2$. For boundary constraints, our model can be easily extended to multidimensional cases. In the following we are interested in studying the monotonicity constraint.

3.4. Monotonicity in two dimensions. We now assume that the input is $\boldsymbol{x} = (x_1, x_2) \in \mathbb{R}^2$ and without loss of generality is in the unit square. The real function f is supposed to be monotone (non-decreasing for example) with respect to the two input variables:

$$x_1 \leq x_1' \quad \text{and} \quad x_2 \leq x_2' \quad \Rightarrow \quad f(x_1, x_2) \leq f(x_1', x_2'). \tag{3.24}$$

The idea is the same as the one-dimensional case. We construct the basis functions such that the monotonicity constraint is equivalent to constraints on the coefficients. Firstly, we discretize the unit square (e.g. uniformly to $(N+1)^2$ knots, see below Figure 9 for $N = 7$). Secondly, on each knot we build a basis function. For instance, the basis function at the knot (u_i, u_j) is defined as

$$\Phi_{i,j}(\boldsymbol{x}) = h_i(x_1) h_j(x_2), \tag{3.25}$$

where $h_j, \ j = 0, \dots, N$ are defined in (3.8). We have

$$\Phi_{i,j}(u_k, u_l) = \delta_{i,k}\delta_{j,l}, \qquad k,l = 0, \dots, N. \tag{3.26}$$

Proposition 3.4. *Using the notations introduced before, we define the finite-dimensional process* $(Y^N(\boldsymbol{x}))_{\boldsymbol{x} \in [0,1]^2}$ *as*

$$Y^N(x_1, x_2) = \sum_{i,j=0}^{N} Y(u_i, u_j) h_i(x_1) h_j(x_2) = \sum_{i,j=0}^{N} \xi_{i,j} h_i(x_1) h_j(x_2), \tag{3.27}$$

where we denote $\xi_{i,j} = Y(u_i, u_j)$ *and the functions* $h_j,\ j = 0, \dots, N$ *are defined in (3.8). Then we have the following properties:*

- Y^N *is a finite-dimensional GP with covariance function* $K_N(\boldsymbol{x}, \boldsymbol{x}') = h(\boldsymbol{x})^\top \Gamma^N h(\boldsymbol{x}')$, *where* $h(\boldsymbol{x})^\top = (h_{i,j}(\boldsymbol{x}))_{i,j} = (h_i(x_1)h_j(x_2))_{i,j}$, $\Gamma^N_{(i,j),(i',j')} = K((u_i, u_j), (u_{i'}, u_{j'}))$ *and* K *the covariance function of the original GP* Y.
- Y^N *converges uniformly to* Y *when* N *tends to infinity (with probability 1).*
- K_N *converges uniformly to* K *when* N *tends to infinity.*
- Y^N *is non-decreasing with respect to the two input variables if and only if the* $(N+1)^2$ *random coefficients* $\xi_{i,j},\ i, j = 0, \dots, N$ *verify the following linear constraints:*
  1. $\xi_{i,j} \geq \xi_{i-1,j}$ *and* $\xi_{i,j} \geq \xi_{i,j-1},\ i, j = 1, \dots, N$.
  2. $\xi_{0,j} \geq \xi_{0,j-1},\ j = 1, \dots, N$.
  3. $\xi_{0,i} \geq \xi_{0,i-1},\ i = 1, \dots, N$.

From the last property, the problem is reduced to simulate the Gaussian vector $\xi = (\xi_{i,j})_{i,j}$ restricted to the convex set $\mathcal{C}_\xi = \left\{ \xi^\top \in \mathbb{R}^{(N+1)^2} \text{such that } \xi_{i,j} \text{ verify the constraints 1. 2. and 3.} \right\}$.

Proof. The proof of the first three properties is similar to the one-dimensional case. Now if the $(N+1)^2$ coefficients $\xi_{i,j},\ i, j = 0, \cdots, N$ verify the above linear constraints 1. 2. and 3. then Y^N is non-decreasing since Y^N is a piecewise linear function for x_1 and x_2 directions. Conversely, if Y^N is non-decreasing then $Y^N(u_i, u_j) = \xi_{i,j},\ i, j = 0, \dots, N$ satisfy the constraints 1. 2. and 3.. ∎

Remark 3.5 (Monotonicity in two dimensions with respect to one variable). If the function is non-decreasing with respect to the first variable only, then the proposed GP defined as

$$Y^N(x_1, x_2) = \sum_{i,j=0}^{N} Y(u_i, u_j) h_i(x_1) h_j(x_2) = \sum_{i,j=0}^{N} \xi_{i,j} h_i(x_1) h_j(x_2), \tag{3.28}$$

is non-decreasing with respect to x_1 *if and only if* the random coefficients $\xi_{i,j} \geq \xi_{i-1,j},\ i = 1, \dots, N$ and $j = 0, \dots, N$.

3.5. Monotonicity in multidimensional cases. The d-dimensional case is a simple extension of the two-dimensional case. The finite-dimensional GP is written as

$$Y^N(\boldsymbol{x}) = \sum_{i_1,\dots,i_d=0}^{N} Y(u_{i_1}, \dots, u_{i_d}) \Pi_{\sigma \in \{1,\dots,d\}} h_\sigma(x_\sigma) = \sum_{i_1,\dots,i_d=0}^{N} \xi_{i_1,\dots,i_d} \Pi_{\sigma \in \{1,\dots,d\}} h_\sigma(x_\sigma), \tag{3.29}$$

where we denote $\xi_{i_1,\dots,i_d} = Y(u_{i_1}, \dots, u_{i_d})$. Remark 3.5 can be extended as well, for the case of a monotonicity with respect to a subset of variables. For instance, the monotonicity of Y^N with respect to the l^{th} dimension input x_l is equivalent to the fact that $\xi_{i_1,\dots,i_l,\dots,i_d} \geq \xi_{i_1,\dots,i_l-1,\dots,i_d},\ i_l = 1, \dots, N$ and $i_1, \dots, i_{l-1}, i_{l+1}, \dots, i_d = 0, \dots, N$.

3.6. Simulation of GP conditionally to equality and inequality constraints. Let us remark that in any cases the approximate GP Y^N is of the form

$$Y^N(\boldsymbol{x}) = \sum_{j=0}^{N} \xi_j \phi_j(\boldsymbol{x}), \qquad \boldsymbol{x} \in \mathbb{R}^d, \tag{3.30}$$

where $\xi = (\xi_0, \dots, \xi_N)^\top$ is a Gaussian vector with covariance matrix Γ^N and $\phi = (\phi_0, \dots, \phi_N)$ are deterministic basis functions. For instance, the constant term $Y(0)$ in model (3.15) can be written as $\xi_0\phi_0(x)$, where $\phi_0(x) = 1$. The space of interpolation conditions is $\mathcal{I} = \left\{ \boldsymbol{c} \in \mathbb{R}^{N+1} \text{ such that } \sum_{j=0}^{N} c_j \phi_j(\boldsymbol{x}^{(i)}) = y_i,\ i = 1, \dots, n \right\}$ and the set of inequality constraints $\mathcal{C}_\xi$ is a convex set (for instance, the non-negative quadrant $\xi_j \geq 0,\ j = 0, \dots, N$ for the non-decreasing constraint in one dimension). We are interested in the calculation of the mean, mode (maximum a posteriori) and prediction intervals (uncertainty quantification) of Y^N conditionally to $\xi \in \mathcal{I} \cap \mathcal{C}_\xi$. Note that their analytical forms except for the mode are not easy to find, hence we need simulation. As explained in §3.3.1, the problem is reduced to simulate the Gaussian vector $\xi = (\xi_0, \dots, \xi_N)^\top$ restricted to convex sets. In that case, several algorithms can be used (see e.g. [2], [4], [10], [14], [15], [18] and [22]).

In this section we introduce some notations that will be used in §4, and emphasize the two cases of truncated simulations. We note ξ_c the mean of ξ conditionally to $\xi \in \mathcal{I}$ without inequality constraints. Then by linearity of the conditional expectation, the so-called usual (unconstrained) Kriging mean is equal to

$$m_K(\boldsymbol{x}) := E\left(Y^N(\boldsymbol{x}) \middle| Y^N\left(\boldsymbol{x}^{(i)}\right) = y_i,\ i = 1, \dots, n\right) = \sum_{j=0}^{N} (\xi_c)_j \phi_j(\boldsymbol{x}), \tag{3.31}$$

where $\xi_c = E\left(\xi | \xi \in I\right)$.

Definition 3.6. *Denote ξ_m as the mean of the Gaussian vector ξ restricted to $\mathcal{I} \cap \mathcal{C}_\xi$. Then the inequality Kriging mean is defined as*

$$m_{KI}(\boldsymbol{x}) := E\left(Y^N(\boldsymbol{x}) \middle| Y^N\left(\boldsymbol{x}^{(i)}\right) = y_i,\ \xi \in \mathcal{C}_\xi\right) = \sum_{j=0}^{N} (\xi_m)_j \phi_j(\boldsymbol{x}), \tag{3.32}$$

where $\xi_m = E\left(\xi | \xi \in I \cap \mathcal{C}_\xi\right)$.

Finally, let μ be the maximum of the pdf of ξ restricted to $\mathcal{I} \cap \mathcal{C}_\xi$. It is the solution of the following convex optimization problem

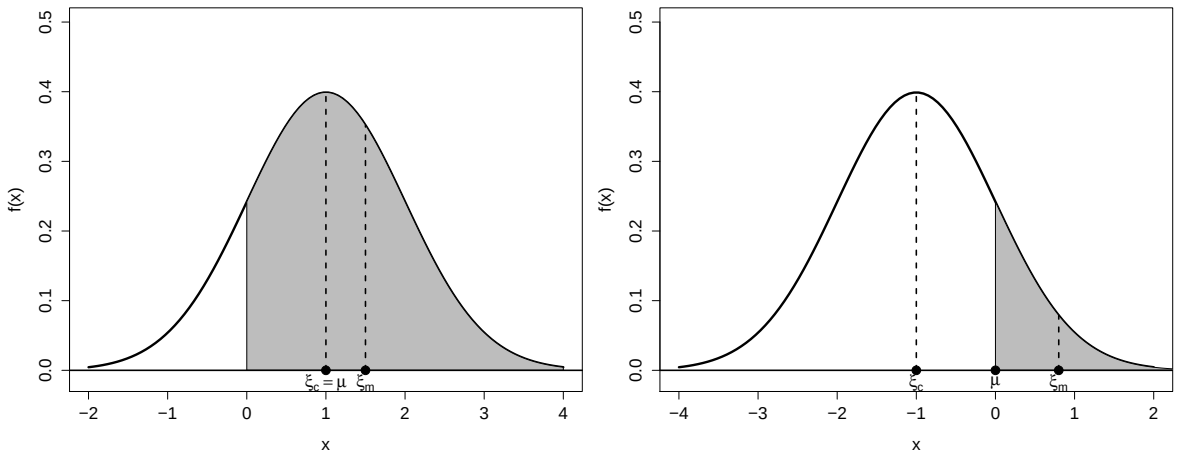
$$\mu = \arg \min_{\boldsymbol{c} \in \mathcal{I} \cap \mathcal{C}_\xi} \left(\frac{1}{2} \boldsymbol{c}^\top \left(\Gamma^N\right)^{-1} \boldsymbol{c} \right), \tag{3.33}$$

where Γ^N is the covariance matrix of the Gaussian vector ξ. In fact, μ corresponds to the mode of the truncated Gaussian vector ξ restricted to $\mathcal{I} \cap \mathcal{C}_\xi$ and its numerical calculation is a standard problem in the minimization of positive quadratic forms subject to linear constraints, see e.g. [3] and [12].

Definition 3.7. *The so-called mode (maximum a posteriori) is equal to*

$$M_{KI}(\boldsymbol{x}) = \sum_{j=0}^{N} \mu_j \phi_j(\boldsymbol{x}), \tag{3.34}$$

where $\mu = (\mu_0, \ldots, \mu_N)^\top$ *is defined in (3.33).*

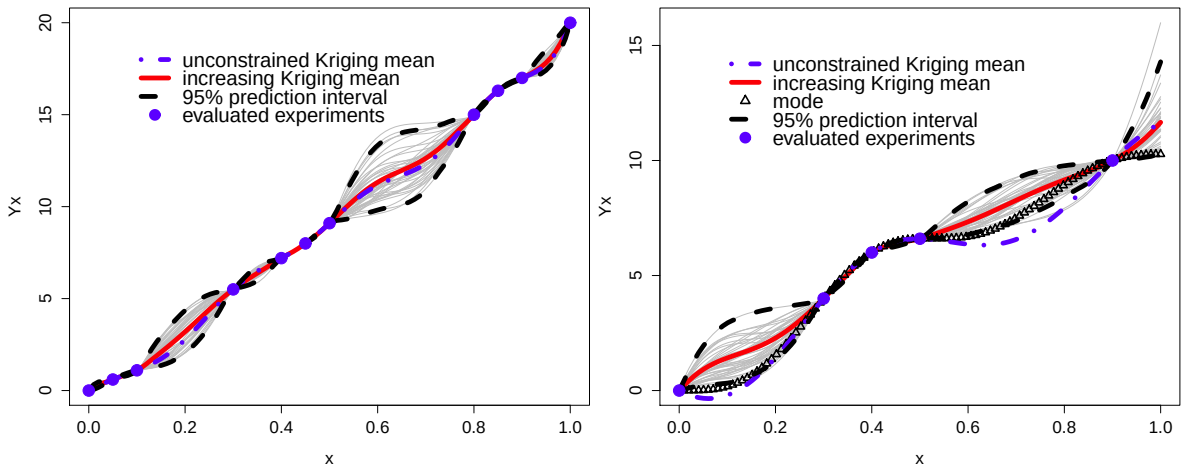


(a) (b)

Figure 4: Two cases of truncated normal variables. The mean is inside (respectively outside) the acceptance region Figure 4a (respectively Figure 4b).

In practice, we have two cases in the simulation of truncated multivariate normal distributions (see Figures 4a and 4b for example in one dimension). In the first case (Figure 4a), we have $\xi_c = \mu$ and so $m_K = M_{KI}$ and they are different from m_{KI}. In this case, ξ_c is inside $\mathcal{C}_\xi$ (non-negative quadrant) and the usual (unconstrained) Kriging mean respects the inequality constraints. The second one, where the three are different (Figure 4b). In this case, ξ_c is outside $\mathcal{C}_\xi$ and the usual (unconstrained) Kriging mean does not respect the inequality constraints.

4. Illustrative examples. In this section the simulation results are obtained by using the Gaussian covariance function. We consider first one-dimension monotonicity, convexity and boundary constraints examples. In two dimensions, we consider the monotonicity (non-decreasing) case with respect to the two input variables and to only one variable.



(a) (b)

Figure 5: Simulated paths (gray lines) drawn from the conditional Gaussian process respecting the non-decreasing constraint in the whole domain $[0, 1]$. The usual Kriging mean (blue dash-dotted line) coincides with the mode (maximum a posteriori) and respects the monotonicity in Figure 5a, but not in Figure 5b. The red line defined as the mean of the simulated paths respects the monotonicity constraint in the whole domain.

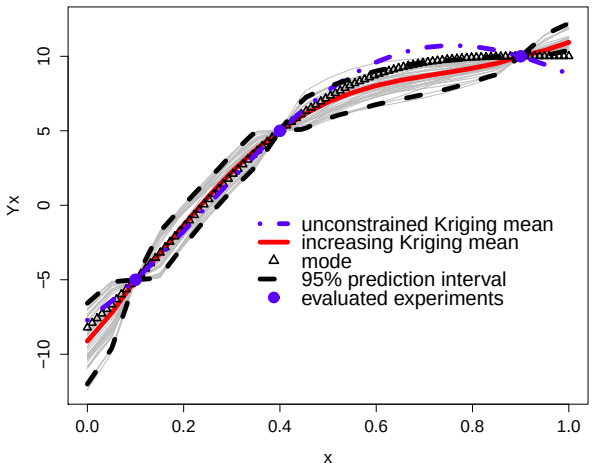


Figure 6: Simulated paths (gray lines) drawn from the model (3.9) using Remark 3.3. Notice that the simulated paths are continuous (non derivable) and non-decreasing in the whole domain.

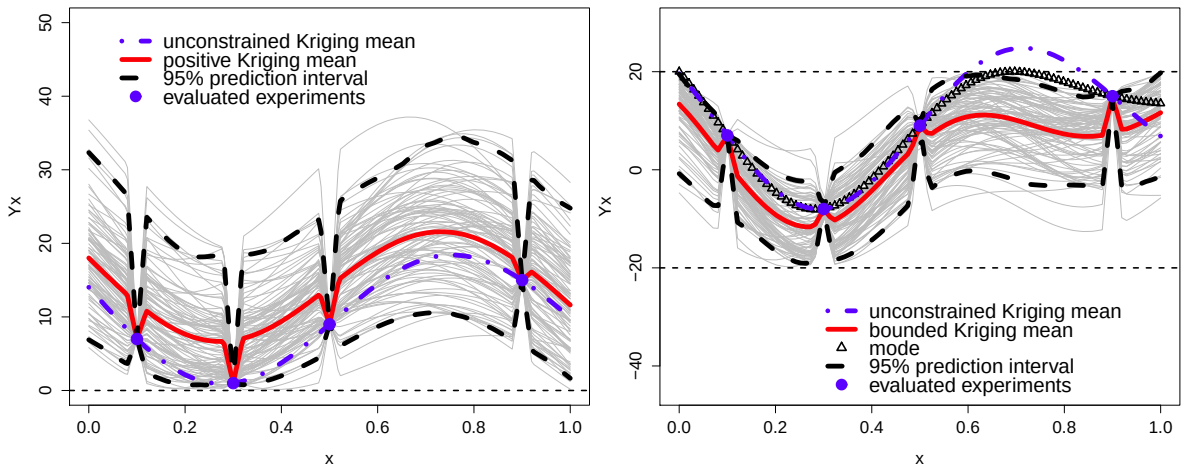
We begin with two monotonicity examples in one-dimension (Figure 5). In Figure 5a the real function is assumed to be non-decreasing and evaluated at 11 design points given by $X = (0, 0.05, 0.1, 0.3, 0.4, 0.45, 0.5, 0.8, 0.85, 0.9, 1)$ and the corresponding output $\boldsymbol{y} = (0, 0.6, 1.1, 5.5, 7.2, 8, 9.1, 15, 16.3, 17, 20)$. We choose $N = 50$ and generate 40 sample paths taken from model (3.15) conditionally to given data and monotonicity (non-decreasing) constraints ($Y'(u_j) \geq 0,\ j = 0, \ldots, N$). Notice that the simulated paths (gray lines) are non-decreasing in the whole domain $[0, 1]$, as well as the increasing Kriging mean m_{KI} (red line). The usual (unconstrained) Kriging mean m_K and the mode M_{KI} (blue dash-dotted line) coincide and are also non-decreasing. This is because ξ_c is inside the acceptance region. In Figure 5b the input is $X = (0, 0.3, 0.4, 0.5, 0.9)$ and the corresponding output is $\boldsymbol{y} = (0, 4, 6, 6.6, 10)$. The increasing Kriging mean (red line) and the mode satisfy the monotonicity constraint, contrarily to the usual (unconstrained) Kriging mean (blue dash-dotted line): it corresponds to the situation where ξ_c lies outside the acceptance region.

Assume that the real function is continuous (but non-derivable), non-decreasing and evaluated at 4 design points given by $X = (0.1, 0.4, 0.9)$ and the corresponding output $\boldsymbol{y} = (-5, 5, 10)$ (Figure 6). The simulated paths (gray lines) drawn from the finite dimensional GP defined in (3.9) conditionally to data interpolation and monotonicity constraints given in Remark 3.3. They are continuous (non-derivable) and non-decreasing in the whole domain, contrarily to the usual (unconstrained) Kriging mean. The mode (maximum a posteriori) and the increasing kriging mean (mean of gray lines) verify the monotonicity (non-decreasing) constraint in the whole domain.

Now we consider the positive and boundary constraints (Figure 7). In Figure 7a ξ_c is inside the acceptance region and the usual (unconstrained) Kriging mean coincides with the mode and respects the inequality constraint, contrarily to Figure 7b, where ξ_c lies outside the acceptance region. In both figures, the simulated paths (gray lines) satisfy the inequality constraints in the whole domain.

Next we suppose that the real function is convex on $[0, 1]$ and evaluated at three design points (Figure 8). The input values are given by $X = (0.2, 0.5, 0.9)$ and the corresponding output is $\boldsymbol{y} = (3, -5, 8)$. We choose $N = 50$ and generate 25 sample paths taken from model (3.22) conditionally to given data and convexity constraints ($\xi_j \geq 0,\ j = 0, \ldots, N$). These paths (gray lines), the mode (maximum a posteriori) and the convex Kriging mean (mean of gray lines) are convex in the whole domain, contrarily to the usual (unconstrained) Kriging mean (blue dash-dotted line).

In two dimensions, the aim is to interpolate a 2D-function defined on $[0, 1]^2$ and non-decreasing with respect to the two inputs. In that case, and by the uniform subdivision of the input set the number of knots and basis functions is $(N+1)^2$. In Figures 9, 10 and 11, we choose $N = 7$, then we have 64 knots and basis functions. Suppose that the real function is evaluated at four design points given by the rows of the 4×2 matrix $\boldsymbol{X} = \begin{bmatrix} 0.1 & 0.9 & 0.5 & 0.8 \\ 0.4 & 0.3 & 0.6 & 0.9 \end{bmatrix}^\top$ and the corresponding output $\boldsymbol{y} = (5, 12, 13, 25)$. The output values respect the monotonicity constraint in two dimensions. We generate 5 simulation surfaces taken from model (3.27) conditionally to given data and monotonicity constraints (Figure 10a). The two red surfaces are the 95% prediction intervals. To check visually the monotonicity in two dimensions, we plot in Figure 10b the contour levels of one simulation surface. The blue points represent the



(a) (b)

Figure 7: Simulated paths (gray lines) drawn from the conditional GP respecting positivity constraints Figure 7a and boundary constraints Figure 7b. The usual (unconstrained) Kriging mean and the mode coincide in Figure 7a, but not in Figure 7b.

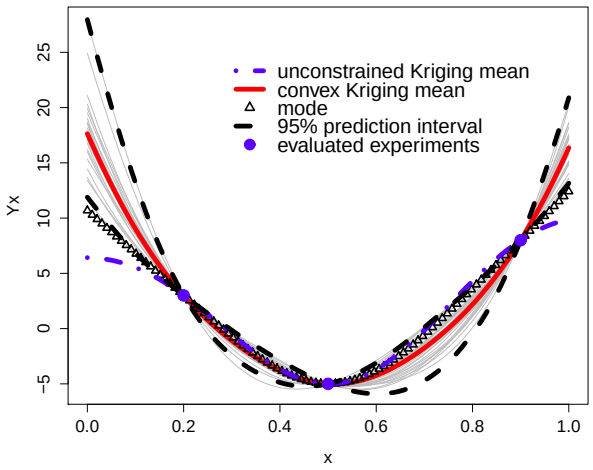


Figure 8: Simulated paths drawn from the conditional GP respecting the convexity constraint in the whole domain $[0, 1]$. The usual (unconstrained) Kriging mean (blue dash-dotted line) does not respect the convexity constraint, contrarily to the mode and the convex kriging mean.

interpolation input locations (design points). If we fix one of the variables and we draw the vertical or horizontal line, it must not intersect a contour level two times.

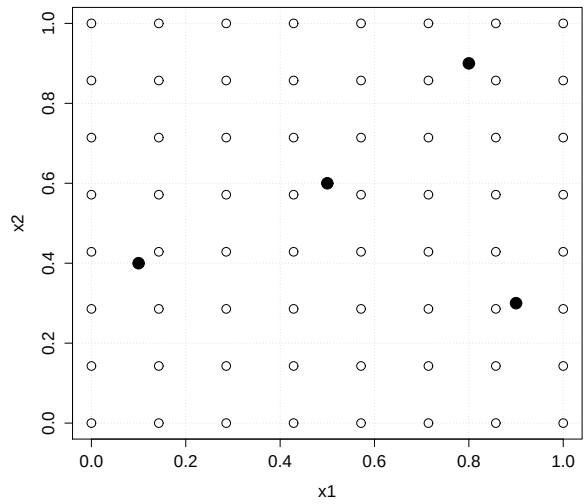


Figure 9: Design points for the monotone 2D interpolation problem (black points) and knots $(u_i, u_j)_{0\leq i,j\leq 7}$ used to define the basis functions.

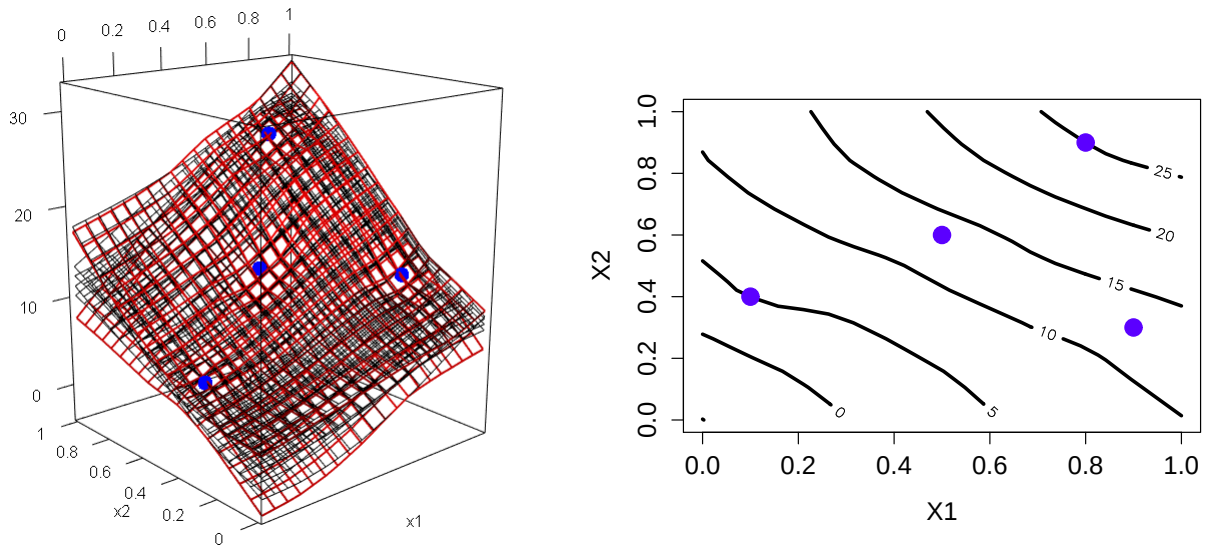
In Figure 11 we draw some simulated surfaces taken from the example used in Figure 10a. All the simulated surfaces are non-decreasing with respect to the two input variables.

In Figure 12 a simulation surface of the conditional GP at four design points (blue points) including monotonicity constraints with respect to the first input variable only is shown.

In order to investigate the convergence rate of the proposed model when N tends to infinity, we plot in Figure 13 the mode (maximum a posteriori) of the conditional GP with both interpolation conditions and boundary constraints when $N = 500$ (blue line). We remark that the red dashed line corresponding to the mode when N is small coincides quickly with the blue line generated from $N = 500$.

5. Conclusion. In this article we propose a new model for incorporating both interpolation conditions and inequality constraints into a Gaussian process emulator. Our method ensures that the inequality constraints are respected not only in a discrete subset of the input set but also in the whole domain. We suggest the approximation of the original Gaussian process by a finite-dimensional Gaussian process Y^N that converges uniformly to the original one. It is constructed by incorporating deterministic basis functions and Gaussian random coefficients. We show that the basis functions can be chosen such that the inequality constraints of Y^N are equivalent to constraints on the coefficients. Therefore the inequality constraints are reduced to a finite number of constraints, and the initial problem is equivalent to simulate a truncated Gaussian vector restricted to convex sets.

Now the problem is open to substantial future work. For practical application the pa-



(a) (b)

Figure 10: Simulated surfaces drawn from the conditional GP respecting the monotonicity constraint for the two input variables Figure 10a. Contour levels for one simulated surface Figure 10b.

rameter estimation should be investigated and Cross Validation techniques can be used. Furthermore, the choice of knots (subdivision of the input set) should be improved to reduce the simulations, as well as the number of basis functions. This problem is related to a prior information on the real function such as variability.

6. Acknowledgements. Part of this work has been conducted within the frame of the ReDice Consortium, gathering industrial (CEA, EDF, IFPEN, IRSN, Renault) and academic (Ecole des Mines de Saint-Etienne, INRIA, and the University of Bern) partners around advanced methods for Computer Experiments. The authors also thank Olivier Roustant, Laurence Grammont and Yann Richet for helpful discussions as well as the participants of the UCM2014 conference and GRF-Sim2014 workshop.

REFERENCES

[1] P. Abrahamsen and F. E. Benth, *Kriging with inequality constraints*, Mathematical Geology, 33 (2001), pp. 719–744.
[2] Carsten Botts, *An Accept-Reject Algorithm For the Positive Multivariate Normal Distribution*, Computational Statistics, 28 (2013), pp. 1749–1773.
[3] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, NY, USA, 2004.
[4] N. Chopin, *Fast Simulation of Truncated Gaussian Distributions* , Statistics and Computing, 21 (2011), pp. 275–288.
[5] H. Cramér and R. Leadbetter, *Stationary and related stochastic processes: sample function properties and their applications*, Wiley series in probability and mathematical statistics. Tracts on probability

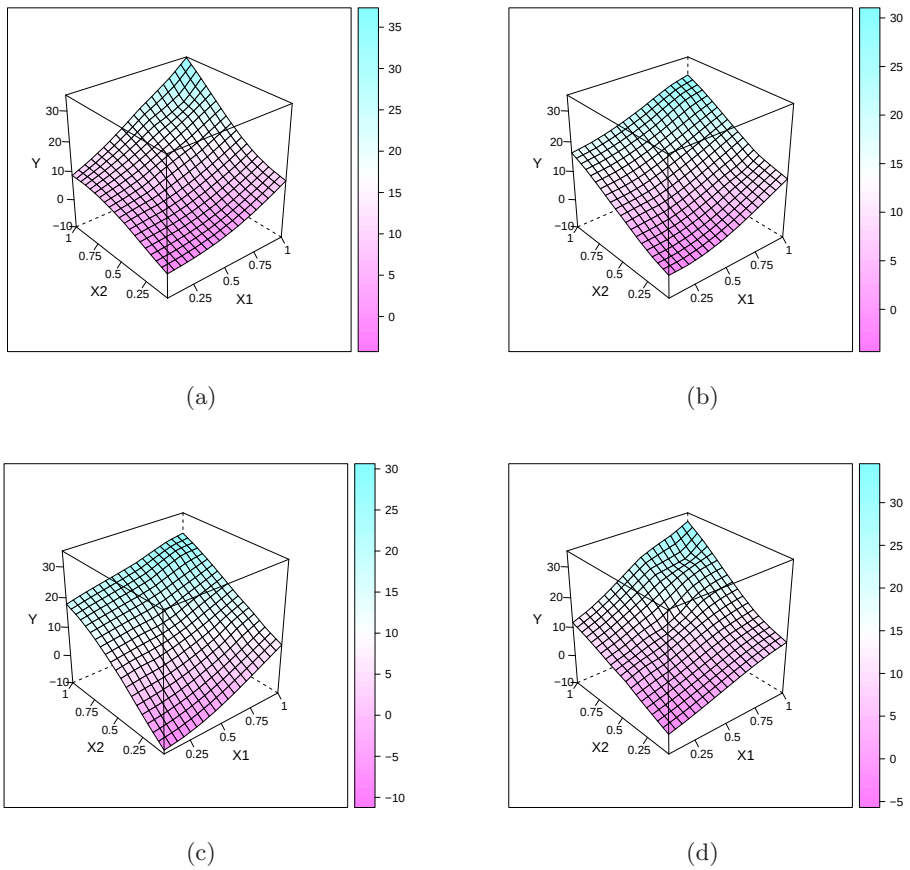


Figure 11: Simulated surfaces drawn from the example used in Figure 10a respecting the monotonicity constraint for the two input variables.

and statistics, Wiley, 1967.

[6] S. Da Veiga and A. Marrel, *Gaussian process modeling with inequality constraints*, Annales de la faculté des sciences de Toulouse, 21 (2012), pp. 529–555.

[7] D. Dole, *Cosmo: A constrained scatterplot smoother for estimating convex, monotonic transformations*, Journal of Business & Economic Statistics, 17 (1999), pp. 444–455.

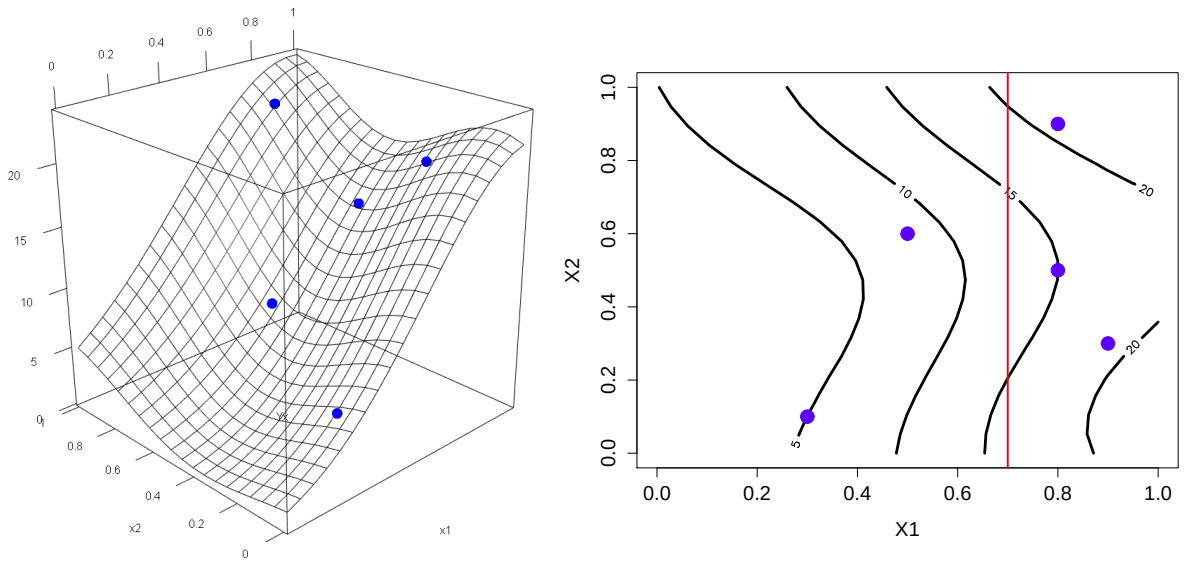
[8] X. Emery, D. Arroyo, and M. Peláez, *Simulating Large Gaussian Random Vectors Subject to Inequality Constraints by Gibbs Sampling*, Mathematical Geosciences, (2013), pp. 1–19.

[9] X. Freulon and C. Fouquet, *Conditioning a Gaussian model with inequalities*, in Geostatistics Tróia '92, Amilcar Soares, ed., vol. 5 of Quantitative Geology and Geostatistics, Springer Netherlands, 1993, pp. 201–212.

[10] J. Geweke, *Efficient Simulation from the Multivariate Normal and Student-t Distributions Subject to Linear Constraints and the Evaluation of Constraint Probabilities*, in Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface, 1991, pp. 571–578.

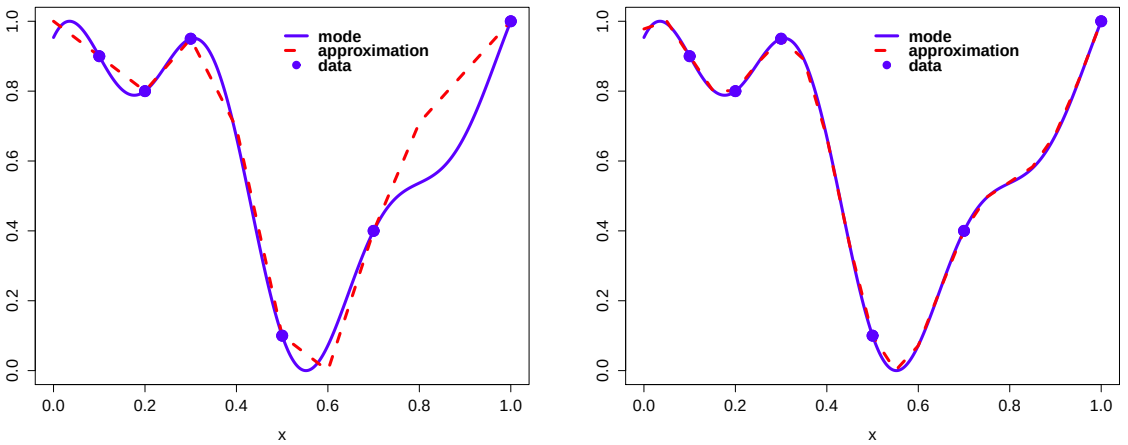
[11] S. Golchi, D. R. Bingham, H. Chipman, and D. A. Campbell, *Monotone Emulation of Computer Experiments*, ArXiv e-prints, (2013).

[12] D. Goldfarb and A. Idnani, *A numerically stable dual method for solving strictly convex quadratic programs*, Mathematical Programming, 27 (1983), pp. 1–33.



(a) (b)

Figure 12: Simulated surface drawn from the conditional GP respecting the monotonicity (non-decreasing) constraint for the first variable only, and the associated contour levels.



(a) (b)

Figure 13: The blue line represents the mode of the GP conditionally to both interpolation conditions and boundary constraints when $N = 500$. The red dashed line is the mode for $N = 10$ (Figure 13a) and $N = 20$ (Figure 13b). Notice that the red dashed line converges quickly to the blue line.

[13] D. R. Jones, M. Schonlau, and W.J. Welch, *Efficient Global Optimization of Expensive Black-Box Functions*, Journal of Global Optimization, 13 (1998), pp. 455–492.
[14] J. H. Kotecha and P.M. Djuric, *Gibbs sampling approach for generation of truncated multivariate Gaussian random variables*, in IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 3, 1999, pp. 1757–1760.
[15] H. Maatouk and X. Bay, *A New Rejection Sampling Method for Truncated Multivariate Gaussian Random Variables Restricted to Convex Sets*, (2014). hal-01063978.
[16] C. Micchelli and F. Utreras, *Smoothing and Interpolation in a Convex Subset of a Hilbert Space*, SIAM Journal on Scientific and Statistical Computing, 9 (1988), pp. 728–746.
[17] E. Parzen, *Stochastic processes*, Holden-Day series in probability and statistics, Holden-Day, San Francisco, London, Amsterdam, 1962.
[18] A. Philippe and C. P. Robert, *Perfect simulation of positive Gaussian distributions.*, Statistics and Computing, 13 (2003), pp. 179–186.
[19] J. O. Ramsay, *Monotone regression splines in action*, Statistical Science, 3 (1988), pp. 425–441.
[20] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*, The MIT Press, 2005.
[21] J. Riihimaki and A. Vehtari, *Gaussian processes with monotonicity information.*, in AISTATS, vol. 9 of JMLR Proceedings, JMLR.org, 2010, pp. 645–652.
[22] C. P. Robert, *Simulation of truncated normal variables*, Statistics and Computing, 5 (1995).
[23] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn, *Design and Analysis of Computer Experiments*, Statistical Science, 4 (1989), pp. 409–423.
[24] M. Villalobos and G. Wahba, *Inequality-Constrained Multivariate Smoothing Splines with Application to the Estimation of Posterior Probabilities*, Journal of the American Statistical Association, 82 (1987), pp. 239–248.
[25] Ian W. Wright and Edward J. Wegman, *Isotonic, Convex and Related Splines*, The Annals of Statistics, 8 (1980), pp. 1023–1035.
[26] W. Xiaojing, *Bayesian Modeling Using Latent Structures*, PhD thesis, Duke University, Department of Statistical Science, 2012.
[27] H. Xuming and S. Peide, *Monotone B-spline Smoothing*, Journal of the American Statistical Association, 93 (1996), pp. 643–650.