



HAL
open science

Uncertainty Characterization of ablative materials for atmospheric reentry of spacecrafts using PuMA

Florian Girault, Pietro Marco Congedo

► **To cite this version:**

Florian Girault, Pietro Marco Congedo. Uncertainty Characterization of ablative materials for atmospheric reentry of spacecrafts using PuMA. [Research Report] RR-9417, Inria Saclay -Île de France. 2021. hal-03384925

HAL Id: hal-03384925

<https://inria.hal.science/hal-03384925v1>

Submitted on 19 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Uncertainty Characterization of ablative materials for atmospheric reentry of spacecrafts using PuMA

Florian Girault, Pietro Marco Congedo

**RESEARCH
REPORT**

N° 9417

August 2021

Project-Team PLATON

ISRN INRIA/RR--9417--FR+ENG

ISSN 0249-6399



Uncertainty Characterization of ablative materials for atmospheric reentry of spacecrafts using PuMA

Florian Girault, Pietro Marco Congedo

Project-Team PLATON

Research Report n° 9417 — August 2021 — 52 pages

Abstract: This work aims to characterize material properties at a microscopic level using the PuMA Software. Specifically, the first objective is to compute statistics of the material properties taking into account several sources of uncertainties. An intrinsic source of uncertainty is given by the stochastic feature of PuMA which yields a stochastic output. An additional source of uncertainties is the choice of some physical parameters, such as for example the thermal conductivity of the fiber. A second objective of this work is to provide some practical guidelines for performing measurements, in particular to perform tomography on some material.

Key-words: uncertainty quantification, kriging, carbon/phenolic ablators, thermal protection systems, porous materials, microstructure, thermal conductivity

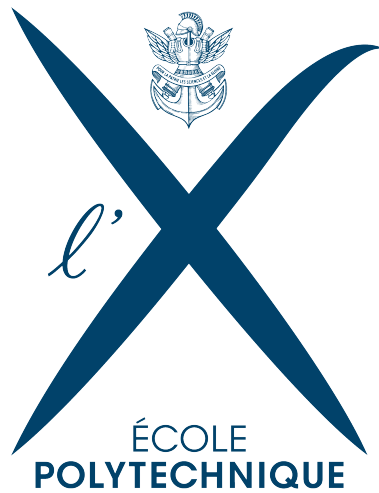
**RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE**

1 rue Honoré d'Estienne d'Orves
Bâtiment Alan Turing
Campus de l'École Polytechnique
91120 Palaiseau

Caractérisation d'incertitude de matériaux ablateurs dans le contexte de véhicules spatiaux en rentrée atmosphérique grâce à PuMA

Résumé : Ce travail vise à caractériser les propriétés des matériaux à un niveau microscopique en utilisant le logiciel PuMA. Plus précisément, le premier objectif est de calculer les statistiques des propriétés des matériaux en tenant compte de plusieurs sources d'incertitudes. Une source intrinsèque d'incertitude est donnée par la simulation de PuMA qui produit une sortie stochastique. Une source supplémentaire d'incertitudes est le choix de certains paramètres physiques, comme par exemple la conductivité thermique de la fibre. Un deuxième objectif de ce travail est de fournir quelques directives pratiques pour effectuer des mesures, en particulier pour réaliser la tomographie de certains matériaux.

Mots-clés : quantification d'incertitude, kriging, ablateurs phénoliques/carbonés, système de protection thermique, matériaux poreux, microstructure, conductivité thermique



Contents

Acknowledgment	6
1 Introduction	7
2 PuMA and Tomography	10
2.1 PuMA basics	10
2.2 Quantity of interest: Thermal conductivity	11
2.2.1 Thermal conductivity	11
2.3 Geometrical resolution in PuMA	13
3 Formulation of the stochastic problem	15
3.1 PuMA: Stochastic solver and parametric dependencies	15
3.2 Assumptions and hypothesis	15
3.3 Mathematical framework	17
3.3.1 Building a predictive model	17
3.3.2 Sensitivity analysis	17
4 Numerical methods	19
4.1 State of the art	19
4.2 Kriging basics	20
4.2.1 Covariance functions and Trend	21
4.2.2 Optimization Functions	24
4.3 Stochastic Kriging	26
4.4 Joint metamodeling	27
4.4.1 Variance metamodel	27
4.4.2 Mean metamodel	28
4.4.3 Allocation methodology	28
4.4.4 Joint Metamodel Algorithm	30
4.5 Illustration	30
4.6 Sensitivity analysis	33
5 Results	34
5.1 Parameters convergence in PuMA	34
5.1.1 Choice of the numerical resolution and tomography	34
5.1.2 Results	35
5.2 UQ numerical campaign	37
5.3 UQ-based prediction	38
5.3.1 Convergence of metamodel	38
5.3.2 Mean prediction	39
5.3.3 Sensitivity analysis	40
6 Conclusions and perspectives	45
6.1 Main achievements	45
6.2 Perspectives	45
Appendices	46
A Convergence of unbiased sample variance	46

B Allocation distribution

47

List of Figures

1	Galileo Probe Heatshield Ablation. Credits to [1].	7
2	IXV before and after atmospheric re-entry : mechanical and thermal stresses can be noticed on the side of the spacecraft on the TPS. Note that the nose has been restored for exposure. Credits to ESA ^{1,2}	8
3	Typical microstructures dealt with by PuMA, at several scale. Credits to [2]. . .	10
4	Example microstructures computationally generated within PuMA V2.1: (a) Random transverse isotropic fiber preform, (b) Packed sphere bed with varying sphere diameters, and (c) Periodic foam structure with uniform void diameter. Credits to [2]	11
5	Fiber representation for different r_{vx} . Visualization is output of marching cube algorithm. Credits to [2].	13
6	PuMA generation module parameters.	14
7	First is a tomographic image of a sample. Second is the 3D representation PuMA is capable of extracting from this image. Third is a typical domain PuMA can artificially generate, with approximately the same parameters of radius, length, orientation and porosity	16
8	Gaussian kernel properties.	22
9	Matérn kernel properties.	23
10	Rational Quadratic kernel distribution from eq (24) for $\theta = 1$ for several α	23
11	Two 2D covariance distribution based on Matérn $\frac{5}{2}$ kernel. First one uses an ellipsoidal kernel, and second one uses a separable kernel. For both figures $\theta = (0.5, 3)^T$	24
12	Figure (a) shows the 2 parameters likelihood surface plot as a function of θ and τ^2 . Two local optima are noticed, indicated by '+'. Figure (b) shows the GP response (mean and 95% interval confidence) for the most to the left optima. Figure (c) shows the GP response for the second optima. Credits to [3]	26
13	Convergence of QoI as functions of r_{vx} and D_{phy} . (a) shows mean of repetitions and (b) precises the error on the mean. Statistical quantities are computed over 100 repetitions, each at nominal parameters $p = 0.85$, $\lambda_f = 12.0W/m.K$, $\lambda_g = 0.04W/m.K$. There is no uncertainty on r_{vx} but $l_{vx} = 80r_{vx} \pm 50\%$	36
14	Convergence of ERMSEs when increasing C and N . The reference is empirical QoI at all training points.	38
15	Influence of temperature on total sample conductivity in case of helium. Plotted are means and total variability errorbars.	39
16	Influence of gas on total sample conductivity for $T = 1380K$. Plotted are means and total variability errorbars.	39
17	Convergence of Sobol indices when increasing C and N	40
18	(a) (resp. (c)) show variance decomposition of the total variability In Plane (resp. Through Thickness) for three temperatures. (b) and (d) show associated Sobol decomposition. Gas is helium.	42
19	(a) (resp. (c)) show variance decomposition of the total variability In Plane (resp. Through Thickness) for three gases. (b) and (d) show associated Sobol decomposition. Temperature is $T = 1380K$	43

Acknowledgment

I would like to express my gratitude to my internship director Pietro Congedo. Thank you for your support all along this period, for having guided my research and for your patience despite all mishaps. Your dedication and enthusiasm were valued help in particular in this special period of remote working.

Thank you to von Karman Institute for Fluid Dynamics members Alessandro Turchi, Thierry Magin and Bernd Helber for your advice, your external opinions were always relevant. Special thanks to Francisco Torres Herrador who has always been quick to response and very helpful.

Moreover, my thanks also go to co-developers of PuMA, and especially to Joseph Ferguson who has modified the software so that we could use it efficiently in our simulation campaign.

Some simulations presented in this paper were carried out using the PlaFRIM experimental testbed, supported by Inria, CNRS (LABRI and IMB), Université de Bordeaux, Bordeaux INP and Conseil Régional d'Aquitaine (see <https://www.plafrim.fr/>). Therefore I would like to thank the support team who has helped me into implementing PuMA on their hardware.

Finally, thanks to Inria and École polytechnique for having made possible this experience. I also want to thank Platon team for welcoming me, and hope we will get in touch in person next time.

1 Introduction

The atmospheric entry is the passage of an object from outer space into the gases of an atmosphere of a planet. While traveling in the atmosphere, these bodies usually exceed sound speed at $7000 - 8000 m.s^{-1}$ which is equivalent to Mach 25. At these speeds, detached shock waves form around the object due to the strong compression of the particles over a very short distance. During this compression, a large fraction of the kinetic energy of the vehicle is transferred to the particles in the flow. This energy transfer results in a temperature rise of several thousand Kelvin and in high chemical activity, which causes loss of mass (ablation) and even complete disintegration of the object (see recent impressive video¹ made by European Space Agency and Deutsches Zentrum für Luft- und Raumfahrt).

When dealing with man-made spacecraft, it is therefore essential to protect its integrity by means of a Thermal Protection System (TPS). The new generation of TPS ablative materials are usually made up of an organic precursor such as carbon fiber (CF) felts and a polymeric isotropic matrix. These materials are able to dissipate the impinging hyperthermal fluxes through a disintegration process, by transforming the received energy into chemical processes and crumbling gradually, while the remaining virgin material is able to insulate the spacecraft.

The design of the TPS still requires several improvements in terms of physical models, experimental validation and numerical predictions. Figure 1 illustrates the inaccurate ablation prediction observed for the Galileo mission probe due to the complexity of the atmosphere of Titan.

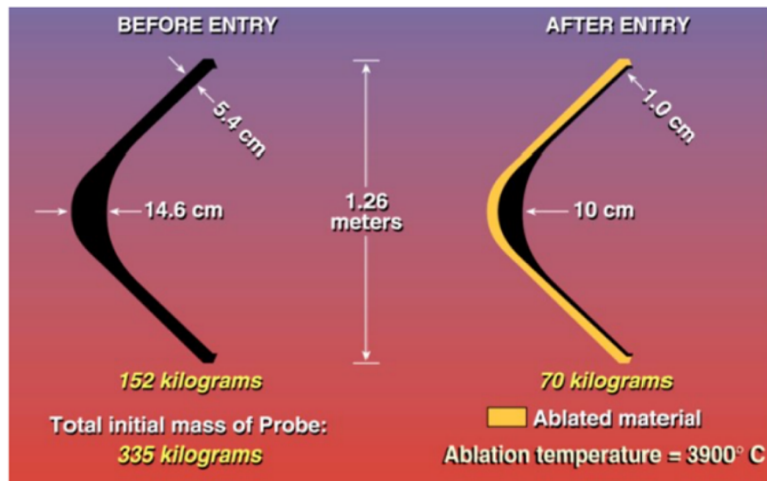


Figure 1: Galileo Probe Heatshield Ablation. Credits to [1].

Of course, the safety of astronauts is at the centre of these concerns, especially after the destruction of the Columbia shuttle due to the loss of a TPS tile during launch. The emergence of new public and private players in space transportation also contributes to the interest in TPS, as it participates to flight independence from other stakeholders. Atmospheric entries of human objects are not limited to Earth, as many probes and rovers are seeking to get closer to atmospheric bodies, either by doing as much analyses as possible before eventually disintegrating (Cassini-Huygens probe on Saturn), or by direct landing on the ground (6 Martian rovers not to

¹https://www.youtube.com/watch?v=1YXdv4Ry2XY&ab_channel=EuropeanSpaceAgency%2CESA. Credits to ESA/DLR, license.

mention all landers, Huygens on Titan); soon perhaps we will see a crewed spacecraft operating an atmospheric re-entry on Mars.

Therefore there is still a lot to be understood in order to master atmospheric re-entry. Recent ESA Intermediate eXperimental Vehicle mission developed by ESA (figure 2) to increase the control of gliding atmospheric re-entry is an example, leading the way to future vehicle Space Rider.



Figure 2: IXV before and after atmospheric re-entry : mechanical and thermal stresses can be noticed on the side of the spacecraft on the TPS. Note that the nose has been restored for exposure. Credits to ESA^{1,2}

Characterization of material physical properties through numerical modeling is a wide area of research. Thousands of contributions have been produced, for plenty of mechanical topics : stress resistance, viscoelasticity, thermo-mechanical coupling, fracture etc. Among types of materials, composite materials have been particularly studied. Their usage is extremely diverse due to the multitude of properties they can feature. To mention just a few: automobile [4], aviation [5, 6], space research [7], medicine [8], sports [9].

In this paper, we will focus on porous materials such as those employed in TPS. Optimizing their properties needs a deep understanding of how they conduct, so that the right quantity of material is used, and correctly applied on the spacecraft. Several papers already contribute to studying carbon porous materials. [10] propose a stochastic modeling method of carbon fiber reinforced composites with polynomial chaos. [11, 12] study properties of such material and the influence of design parameters thanks to Finite Element Analysis and tomography. [13] focuses on the influence of the very orientation of the fibers in high-fidelity simulations. Finally, [14] studies heat capacity of carbon composites at high temperature in the context of TPS design in case of pyrolysis.

More specifically, our study takes into account uncertainties of parameters and modeling for material characterization. All materials are manufactured with uncertainty on parameters, and controlling the impact of uncertainty on the variability of the properties of interest is necessary for

¹https://www.esa.int/ESA_Multimedia/Images/2015/01/IXV_is_being_prepared_for_launch

²https://www.esa.int/ESA_Multimedia/Images/2015/06/IXV_being_displayed_at_the_Paris_Air_and_Space_Show

reliability. Uncertainty Quantification (UQ) seeks to characterize and potentially reduce these uncertainties of models or physical systems. It is particularly adapted to numerical methods because it usually requires random yet controlled data, which are more easily generated by numerical models [15]. Uncertainty propagation can be forward (for predicting variability of output given variability of inputs) or backward (predicting for a certain output distribution the input parameters and their uncertainty) [16]. It does not only focus on parameters uncertainty but also on inherent stochasticity of the model, and is capable of assessing influence of each type of variability. Finally, optimization under uncertainty is a fundamental axis of research nowadays, due to both needs of cost optimality and reliability [17].

UQ has been applied to mechanical models, and some procedures have been described in [18]. Concerning TPS, some papers have already applied UQ methodology. [19] performs sensitivity analysis on pyrolysis gas composition taking Stardust spacecraft as example. [20] studies the respective impact of several chemical reactions for laminar and turbulent aeroheating predictions for Mars entry. Finally [21] applies uncertainty analysis for thermochemical ablation modeling for estimating effects of the uncertain estimates of a plasma wind-tunnel test conditions on the final results of the model.

In our study we will take advantage of PuMA Software developed by NASA [2]. This numerical tool can perform mechanical properties computations on digitized carbon preform domain. Its particularity is that it does not make average computation on large domains like other existing software, but rather on small domains of typical size of a few hundreds of micrometers. For now, few contributions have been made thanks to PuMA. [22] describes some weave generation capabilities, [23] performs a preliminary analysis of the influence of microstructure on macroscopic properties. But there is nothing done with UQ methods.

Taking into account uncertainties within PuMA has a twofold interest:

- Assessment of the prediction of some quantities of interest when designing a given material in terms of domain size and voxel resolution. This also leads to the capability of preparing tomography campaign, so that the experimenter has an idea of the minimal size a sample should have in order to be representative, and what variabilities should be expected when observing a material.
- Providing a methodology for building a surrogate of the thermal conductivity of porous materials as a function of several geometrical and physical parameters. We aim at building a surrogate both for prediction purposes and for performing uncertainty quantification and sensitivity analysis. In particular, we consider uncertain parameters and the intrinsic stochasticity of the PuMA solver.

The first part of this paper is dedicated to a description of PuMA software, and what it is possible to do thanks to it. Then we formalize the simulation problem, introducing the parameters and the stochastic aspect of the code. In the third section, we introduce the numerical tools we need for building the metamodel and proceed to a sensitivity analysis. Finally we apply our methodology to the real mechanical problem estimating the influence of each parameter over the output variability, and give advice for performing tomography.

2 PuMA and Tomography

2.1 PuMA basics

Porous Microstructure Analysis (PuMA) is a software developed by NASA to calculate material properties at the microscopic scale. From a 3D material sample (whose source will be detailed below), PuMA can compute geometric properties such as tortuosity or porosity and physical properties such as thermal and electrical conductivities. Finally, it implements a particle-based oxidation module: it simulates the decomposition of carbon materials due to oxidation under the collision of high-temperature particles. We specify that we use version 2.2 of PuMA.

Let us make a quick review of the properties PuMA can compute. Beforehand, it divides the sample into core materials. Most basics for a fiber made porous material will be on the one hand fibers themselves and the gas surrounding fibers on the other hand: it is the case figure 3. Of course, you might need to decompose the material more precisely, adding several types of fibers, sticky resin, gas concentrations etc.: PuMA allows to define internal subdomains to a certain extent.

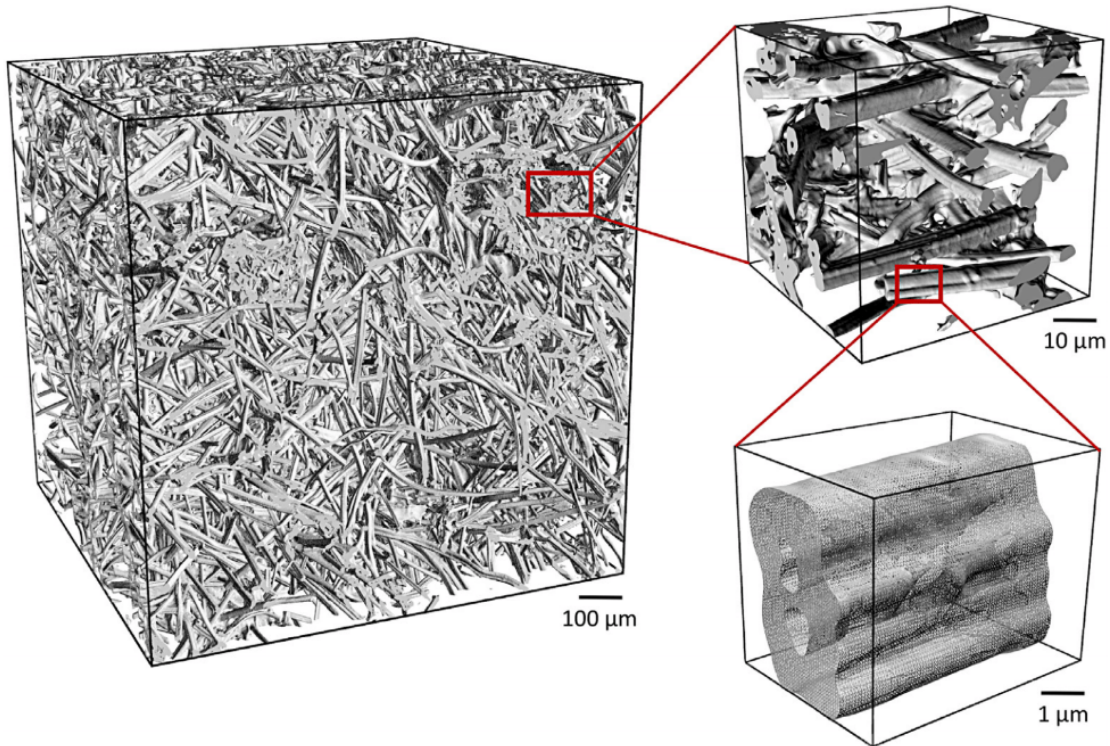


Figure 3: Typical microstructures dealt with by PuMA, at several scale. Credits to [2].

The domain may come from two different sources. The first one is importing a microtomography image into PuMA, such as a TIFF image: internal subdomains then corresponds to an interval of greyscale. The second one is artificial generation. Given certain parameters and shapes, PuMA is capable of randomly generate domains that converge toward the parameters. See figure 4 for a few examples. In these domains, the basic length is the voxel, tantamount to pixel but in 3D.

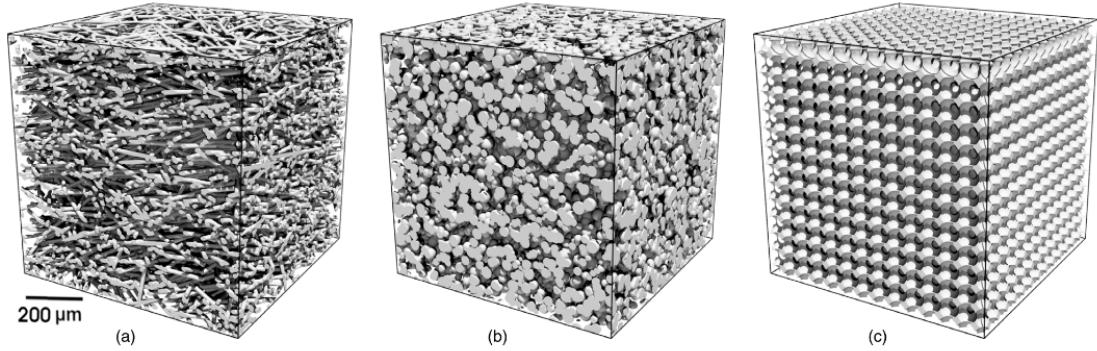


Figure 4: Example microstructures computationally generated within PuMA V2.1: (a) Random transverse isotropic fiber preform, (b) Packed sphere bed with varying sphere diameters, and (c) Periodic foam structure with uniform void diameter. Credits to [2]

Once the domain is defined, PuMA can compute :

- Volume fraction of the subdomain i : $\Phi_i = \frac{V_i}{V_{tot}}$, using sums of voxels for computation of volumes
- **Porosity** p , volume fraction of the void or ambient gas
- Specific surface area SSA , ratio between the total surface and the total volume. The computation is implemented with a Marching Cubes Algorithm
- Tortuosity η quantifies the effects of the material microstructure on diffusion : $\eta = p \frac{D_{ref}}{D_{eff}}$ where D_{ref} (resp. D_{eff}) accounts for the effective diffusion coefficient in the porous material (resp. the reference diffusion coefficient)
- Effective electrical σ_{elec} and **thermal conductivity** λ which depend on the material microstructure and its materials' conductivities.

2.2 Quantity of interest: Thermal conductivity

We are interested in computing some thermal properties of the material. One of these properties is the thermal conductivity of the samples. Keeping in mind the two objectives of this paper, we need to define the simulation parameters clearly. In this section, first, we remind the heat equation verified by the temperature and thermal conductivity in a physical medium. Then, we focus on the parameters of PuMA and define the problem for designing further simulation campaigns and tomography.

2.2.1 Thermal conductivity

As introduced, our study mainly focuses on the mean thermal conductivity of a sample, accounting for all the material constituents. Making uncertainty quantification with PuMA enables us to study the mean conductivity at the macroscopic scale and the variation at smaller scales (0.1mm). At this scale, variations are more significant, and the effects of geometry are much more visible. This section reminds how to compute the mean thermal conductivity of a domain

theoretically but doesn't expand on how PuMA implements it. Before all, note that in this section 2.2.1 and only in this one, $\boldsymbol{\lambda}$ stands for the local thermal conductivity tensor (voxel scale), and $\bar{\boldsymbol{\lambda}}$ stands for the thermal conductivity tensor at the sample scale.

The heat equation writes :

$$\rho C_P \frac{\partial T}{\partial t} + \nabla \cdot (\boldsymbol{\lambda} \cdot \nabla T) = S \quad (1)$$

where ρ is the local density, C_P is the local heat capacity, $\boldsymbol{\lambda}$ the local thermal conductivity (tensor in the most general case), T the local temperature and S a volume source. We are interested in the very steady case, with no external source. Furthermore, we assume the material to be locally isotropic, so that (1) becomes :

$$\nabla \cdot (\boldsymbol{\lambda} \cdot \nabla T) = 0 \quad (2)$$

If \mathcal{X} is the physical domain and $\partial\mathcal{X}$ its boundary, the usual associated boundary condition is

$$\forall \mathbf{x} \in \partial\mathcal{X}, \quad T(\mathbf{x}) = t(\mathbf{x}) \quad (3)$$

The method used is the so-called EJ-Heat method [24], based on a finite-difference method to calculate the effective thermal conductivity of a material based on its constituent parts. For a rectangular cuboid sample, computations are made direction by direction. For a given Cartesian direction, a temperature gradient is imposed on the two opposite corresponding faces. Let's do it for any direction \mathbf{x}_α :

$$T^\alpha(x_\alpha = 0, :, :) = T_1, \quad T^\alpha(x_\alpha = d_\alpha, :, :) = T_2$$

We assume temperature to be periodic in the other directions. Then the heat flux is computed on the three directions, from which is computed the mean thermal conductivity for each direction : it gives column α of tensor $\bar{\boldsymbol{\lambda}}$.

We begin by transforming temperature and coordinates variables :

$$\begin{aligned} T^\alpha &\longrightarrow \frac{T^\alpha - T_1}{T_2 - T_1} \\ x_\beta &\longrightarrow \frac{x_\beta}{d_\beta} \quad \text{for } \beta = 1, 2, 3 \end{aligned}$$

In particular

$$T^\alpha(x_\alpha = 0, :, :) = 0, \quad T^\alpha(x_\alpha = 1, :, :) = 0$$

Then we write T as the sum of a mean linear field and a local field U^α : $T^\alpha = U^\alpha + x_\alpha$. Under Einstein convention, equation 2 becomes :

$$\frac{\partial}{\partial x_k} \left(\boldsymbol{\lambda}_{ki} \frac{\partial U^\alpha}{\partial x_i} + \boldsymbol{\lambda}_{ki} \delta_{i\alpha} \right) = 0 \quad (4)$$

Once this equation has been solved in U^α , we recall $\mathbf{q}^\alpha = (\boldsymbol{\lambda}_{\beta i} \frac{\partial U^\alpha}{\partial x_i} + \boldsymbol{\lambda}_{\beta i} \delta_{i\alpha}) \mathbf{x}_\beta$ is the local heat flux (for direction computation α).

Finally the sample thermal conductivity comes from :

$$\bar{\mathbf{q}}^\alpha = \bar{\boldsymbol{\lambda}} \nabla T \quad \Rightarrow \quad \bar{\boldsymbol{\lambda}}_{j\alpha} = \bar{\mathbf{q}}_j^\alpha = \int_0^1 \int_0^1 \int_0^1 \boldsymbol{\lambda}_{ji} \frac{\partial U^\alpha}{\partial x_i} + \boldsymbol{\lambda}_{ji} \delta_{i\alpha} dx_1 dx_2 dx_3 \quad (5)$$

2.3 Geometrical resolution in PuMA

In this part, we focus on configuring PuMA for simulations and tomography. First, let us note that we try to predict general trends, and we need plenty of simulations. Because we can not afford many inputs from tomography and need to control the parameters, we will widely use PuMA feature artificial generation.

Let us use the subscript phy for physical quantities and vx for PuMA quantities. For example, R_{phy} is the average radius of fiber (a few micrometers) but r_{vx} is the number of voxels of the radius as used in PuMA, noted “Radius (vox)”. Concerning the number of voxels, note that in PuMA, the accuracy level is tantamount to the number of voxels by the radius of fiber (R_{vx}), while in tomography, the physical size of a voxel defines it. See figure 5 as illustration. Knowing the R_{phy} (a few micrometers), it is straightforward to convert PuMA parameters to physical space. The conversion to the physical space is made thanks to the PuMA parameter “Voxel Length” l_{vx}^{phy} defined by $\frac{R_{phy}}{r_{vx}}$.

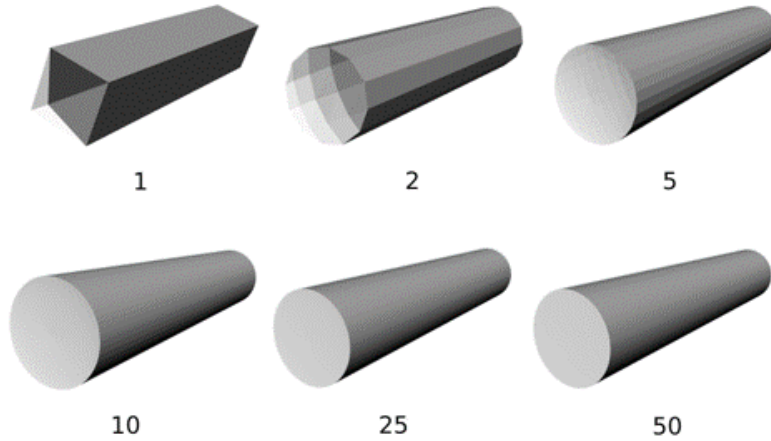


Figure 5: Fiber representation for different r_{vx} . Visualization is output of marching cube algorithm. Credits to [2].

The two tunable PuMA parameters are then r_{vx} and d_{vx} . Once r_{vx} is chosen, then we fix l_{vx}^{phy} in order to match R_{phy} . If one multiplies r_{vx} by a factor γ , then l_{vx}^{phy} should be divided by the same factor in order to preserve R_{phy} . The numerical domain is a rectangular mesh of cubic voxels, tuned by the 3×1 vector d_{vx} . Figure 6 shows PuMA generation module window.

Regarding the domain size, the point is to consider a domain large enough to be representative, i.e. permitting to compute converged average properties of the material. Precisely, this means that convergence is attained when the variation of the output of the simulation with respect to the domain size becomes lower than a certain threshold. Since the generation is stochastic, convergence should be computed by looking at the conditional mean of the underlying distribution for each domain size. Note that the variance of the same distribution should converge to zero when the domain size tends to infinity. The additional problem here is that some physical parameters are known with a certain level of uncertainty (ex: thermal conductivity of the fiber). Considering this uncertainty is then also required to have a reliable estimation of the representative domain size. At least, one should assess the sensitivity of the representative domain size to the input uncertainty.

Nonetheless, when decreasing D_{phy} , porosity of the generated domain is still the value given as input. Therefore, we do not zoom in a given sample, thus changing geometrical parameters such

Input Parameters			
Domain	X	Y	Z
	400	400	400
Radius (vox):	5	±	0
Length (vox):	400	±	0
Angle Variation	X	Y	Z
	90	90	25
Voxel Length (um):	1		
Porosity:	0.85		
<input checked="" type="checkbox"/> Allow Intersections	<input type="checkbox"/> Save Info		
Generate	100%		
Generation Output			
SSA (1/m):	56078		
SA (m^2):	3.56214e-6		
Triangles	1.07586e+7		
Porosity	0.849581		

Figure 6: PuMA generation module parameters.

as porosity. We are generating domains increasingly smaller with given geometrical parameters and study the mechanical properties.

Remember the objective of the study : study the variability of certain mechanical properties at small scales. So we do not try to generate the greatest possible domain, because we know stochasticity is a critical aspect at industrial scale. But if we consider too small domains, no manufacturer can achieve this level of accuracy when trying to respect geometrical parameters.

Let's summarize constraints about r_{vx} and D_{phy} .

- r_{vx} should be large enough to define correctly the fiber.
- D_{phy} should be large enough to meet realistic manufacturing accuracy and tomography size, and thus introduce stochasticity in fibers distribution.
- D_{phy} should not be too large in order to keep accounting for small domains.
- d_{vx} should not be too large for computational cost reasons : for simulation with $d_{vx} = 100vx$ it lasts approximately three minutes when it lasts several hours for $d_{vx} \sim 1000vx$, without guarantee of success. Despite PlaFRIM multiple big memory nodes and parallelisation of simulations, running one simulation campaign therefore takes a lot of time. Parallelisation was performed thanks to Parallel Python module runned on Inria PlaFRIM cluster.

Note that last constraint also concerns both r_{vx} and D_{phy} , because $d_{vx} = \frac{r_{vx}D_{phy}}{R_{phy}}$. Considering cubic samples, the maximum d_{vx} we have computed with reasonable computation time is $d_{vx} \sim 1000vx$.

Next in this work, specifically a part of Section 5, illustrates some results concerning a numerical procedure to use PuMA for setting up tomography.

3 Formulation of the stochastic problem

In this section, we address the issue of stochasticity more precisely and formalize the mathematical framework of our study. First, we clarify the assumptions for building a predictive model for thermal conductivity. Then, we introduce notations that will be useful for modeling the problem.

3.1 PuMA: Stochastic solver and parametric dependencies

Designing thermal protection systems requires a certified tool to predict material performances. However, for some kinds of materials it is particularly challenging to provide an accurate description of the properties of interest.

For instance, properties can change when decreasing the size of the sample domain by introducing effects that were averaged at bigger scales. It is especially the case for highly porous material we are interested in. For this kind of material, made of carbon fibers of typical diameter of a few μm , porosity is of the order of 0.9, which means only 10 % of the entire volume is indeed conductive solid; what remains is schematically either gas or cohesive resin. Ultimately, for normal-size protection of total volume about $0.1 - 1m^3$, averaging porosity and other parameters/properties may lead to underestimating the variations at smaller scales. The designing process should then identify the design parameters, their associated uncertainties, and the causes of non-parametric stochasticity.

Once specified the input parameters, PuMA generates a geometry and provide an estimation of the material properties, which is intrinsically stochastic, due mainly to fibers orientation, radius and length following some random distributions. Repetitions of the PuMA simulations are then required to get averaged properties taking into account this intrinsic stochasticity. One can easily persuade itself that for a given porosity, theoretically an infinity of geometries are possible and two different geometries do not lead to the same quantity of interest (QoI) \mathbf{G} . An additional source of uncertainty is given by some other parameters, which are not well-known, and are essentially:

- Porosity p
- Inherent properties of the fibers G_f
- Inherent properties the gas G_g

Overall, PuMA is treated here as a stochastic solver which has parametric dependencies.

3.2 Assumptions and hypothesis

What follows apply for any QoI computed by PuMA. From what has been described Section 2, we will use the following model.

Material hypothesis

- The material is made of two phases : carbon fiber and gas. Both phases are homogenous (their properties do not depend on where they are measured).
- Conditions are uniform. For all external parameters to the material (*eg.* medium temperature), they are constant in all the domain.
- Properties are uniform. Each phases has properties that does not vary with the location.

- Generally, thermal protection systems feature a main geometrical plane in which fibers are approximately distributed. Later, what is related to the plane is called "In Plane" (IP) and what is related to the normal direction is called "Through Thickness" (TT).

We divide the quantity \mathbf{G} into two components : one in plane and one through thickness. We write the sample quantity $\mathbf{G} = (G_{IP}, G_{TT})^T$. Note that in Section 2, \mathbf{G} was a 3×3 tensor. Yet in practice diagonal terms are dominant for typical QoI, approximately 2 orders of magnitude higher than crossed terms (see [23] for details). In next paragraph, G_{IP} and G_{TT} are more precisely defined.

Numerical hypothesis

- Fibers are cylinders. This is a strong approximation : in real materials, they have a cylinder-like shape, but not straight, and sometimes gather in clusters without specific shape. However we need it for using PuMA preform generation tool. Note that we do not impose cylinders not to intersect.
- Fibers radius and length follow statistic distributions. We note R_f (resp. L_f) and σ_{R_f} (resp. σ_{L_f}) the mean and standard deviation of fibers' radius (resp. length).
- Fibers orientation in space follows a transverse isotropic statistical distribution. That is to say, if we locate a fiber orientation on a 3D sphere of radius 1 in spherical coordinates (θ, Φ) (with \mathbf{z} the normal vector), then Φ is identically distributed in $[0, 2\pi]$ and θ follows a law of expected value equal to $\frac{\pi}{2}$. Because samples are cubes, in plane \mathbf{G} is the mean of the transverse \mathbf{G} : $G_{IP} = \frac{G_x + G_y}{2}$, $G_{TT} = G_z$

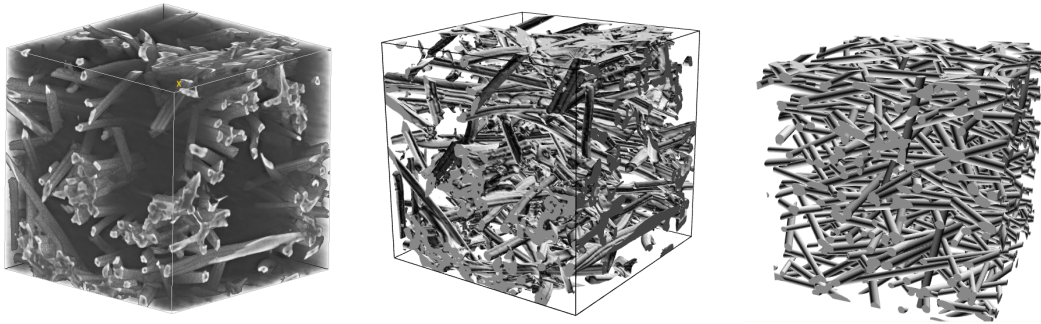


Figure 7: First is a tomographic image of a sample. Second is the 3D representation PuMA is capable of extracting from this image. Third is a typical domain PuMA can artificially generate, with approximately the same parameters of radius, length, orientation and porosity

Figure 7 compares 3 images. They illustrates the 3 degrees of approximation from raw observation of material to PuMA simulation, along with the assumptions we have made. At the end of this paper we will discuss the effects of such approximations.

Mathematical hypothesis

We need some information about the stochasticity aspect of the model. We work under the following assumption :

Assumption 1. $\forall \mathbf{x} \in \mathcal{X}$, $\mathbf{G}(\mathbf{x}) \sim \mathcal{N}(\mu(\mathbf{x}), \mathbf{V}(\mathbf{x}))$, where $\mathbf{V}(\mathbf{x})$ is the 2×2 covariance matrix.

Note that *a priori* $\mathbf{V}(\mathbf{x})$ is not diagonal. However, in following sections we will build two independent surrogate models, one for each component : therefore we focus on the diagonal terms of the matrix.

Even knowing the statistical laws related to the last numerical assumptions, there is still an infinite number of possibilities for choosing the positions of fiber; this is why we describe our model as non completely parametric.

Geometry parameters are not explicitly function variables : they are fixed to be as representative as a real material as possible. Their values are taken from [25], who studies Calcarb[®] CBCF 18-2000. However they play a role in stochasticity. Still note that even with deterministic radius, length and orientation, the model remains stochastic.

3.3 Mathematical framework

3.3.1 Building a predictive model

Let's cast our model into a mathematical framework. Suppose we know the distribution of fibers geometric parameters : orientation, radius and length. We fix the temperature and gas type which are external parameters.

Let \mathcal{X} be the set of parameters of interest. Then the quantity of interest is a stochastic process $\{G(F(\mathbf{x}, \boldsymbol{\xi}))\}_{\mathbf{x} \in \mathcal{X}}$ defined on the probability space $(\Xi, \mathcal{F}, \mathbb{P})$ and measurable space $(\mathbb{R}^n, \mathcal{X})$, where Ξ is the set of stochastic parameters. In a few paragraphs, we will discuss further what is $\boldsymbol{\xi}$. F is the function associating \mathbf{x} and $\boldsymbol{\xi}$ to the stochastic state. For convenience, we will independently write $G(F(\mathbf{x}, \boldsymbol{\xi}))$ and $G(\mathbf{x}, \boldsymbol{\xi})$.

This means for a given $\mathbf{x} \in \mathcal{X}$, $G(\mathbf{x})$ is a random variable with values in \mathbb{R}^n . Furthermore, the probability law of $G(\mathbf{x})$ depends on the choice of geometric parameters laws as well as external parameter. We assume $G(\mathbf{x})$ is a second-order random variable. In this study, we propose a numerical strategy to build a surrogate model of $G(\mathbf{x})$ using the values estimated of G on a few training points. Then, we use this surrogate model to compute the first and second moment of $G(\mathbf{x})$ with respect to \mathbf{x} . An additional question we address in this work is how to perform a sensitivity analysis of G with respect to the stochasticity and the parametric dependencies, which will be detailed in the following section.

3.3.2 Sensitivity analysis

Second step is assessing the respective impacts of parametric and non parametric uncertainty on the variability of G . Sensitivity analysis has been employed in several contexts. When collecting data, it can help identifying main variables and uncertainties for optimizing additional collections [26]. Furthermore, sensitivity analysis can play an important role in model verification and validation when developing and refining models (*e.g.* [27, 28]). It can also provide insight into the robustness of model results when making decisions (*e.g.* [29, 30]). Overall, sensitivity analysis methods have been applied in diverse fields, including complex engineering systems, economics, social sciences, medical decision making, and others (*e.g.* [31, 32, 33]).

[34] distinguishes two categories of sensitivity analysis : local and global. The former focuses in the characterization of the variability in the statistical moments based on their derivatives with respect to the parameters ([35]). Global analysis aims at assessing the uncertainty effects when parameters vary over their whole range following their own distribution. They are often based on the so-called Sobol or ANOVA decomposition accounting for inter-dependencies between parameters [36], before computing the partial variances and sensitivity indices [37].

[38] has proposed a new framework dealing with stochasticity as an independent source of variability and add it in the Sobol decomposition. This allows the decomposition of the stochas-

tic model output, and not the decomposition of its statistical moments. [34] suggests a general methods for global sensitivity analysis in the case of stochastic simulation with uncertain parameters

We introduce the Sobol-Hoeffding decomposition [39] from which we derive an expression for the sensitivity indices, which provide a measure for the impact of all different sources on the global variability. Assuming \mathbf{x} and $\boldsymbol{\xi}$ are independent variables, the SH decomposition separates G into a sum of independent and orthogonal elements :

$$G(\mathbf{x}, \boldsymbol{\xi}) = \bar{G} + G_{par}(\mathbf{x}) + G_{ch}(\boldsymbol{\xi}) + G_{mix}(\mathbf{x}, \boldsymbol{\xi}) \quad (6)$$

where \bar{G} is a constant, $G_{par}(\mathbf{x})$ is a random functional that depends only on the parameters, $G_{ch}(\boldsymbol{\xi})$ depends only on the Poisson processes, and $G_{mix}(\mathbf{x}, \boldsymbol{\xi})$ depends on both random inputs. Orthogonality gives uniqueness of the decomposition :

$$\begin{aligned} G_{par}(\mathbf{x}) &= \mathbb{E}[G|\mathbf{x}] - \mathbb{E}[G] \\ G_{ch}(\boldsymbol{\xi}) &= \mathbb{E}[G|\boldsymbol{\xi}] - \mathbb{E}[G] \\ G_{mix}(\mathbf{x}, \boldsymbol{\xi}) &= G(\mathbf{x}, \boldsymbol{\xi}) + \mathbb{E}[G] - \mathbb{E}[G|\mathbf{x}] - \mathbb{E}[G|\boldsymbol{\xi}] \end{aligned} \quad (7)$$

Orthogonality also gives the following decomposition :

$$\mathbb{V}[G] = V_{par} + V_{ch} + V_{mix} \quad (8)$$

Finally, the Sobol indices are :

$$S_{par} = \frac{V_{par}}{\mathbb{V}[G]}, \quad S_{ch} = \frac{V_{ch}}{\mathbb{V}[G]}, \quad S_{mix} = \frac{V_{mix}}{\mathbb{V}[G]} \quad (9)$$

We have exposed here the decomposition for only two parameters (one uncertain and one stochastic), but the method can apply to an undefined number of parameters, as long as they are mutually independent. If \mathbf{U} is the generalized random vector $\mathbf{U} = (\mathbf{x}_1, \dots, \mathbf{x}_s, \boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_r)$ with mutually independent components, the generalized decomposition of G is :

$$G(\mathbf{U}) = \sum_{u \in \mathbb{P}_{\mathcal{U}}} G_u(\mathbf{U}_u) \quad (10)$$

where G_u are mutually orthogonal random functions depending of subvectors \mathbf{U}_u of \mathbf{U} . These functionals are recursively given by :

$$G_u(\mathbf{U}_u) = \mathbb{E}[G|\mathbf{U}_u] - \left(\sum_{v \subset u} G_v(\mathbf{U}_v) \right) \quad (11)$$

4 Numerical methods

The main objective of this work is to evaluate statistics of some quantities of interest, *i.e.* material properties estimated with Puma, which is dependent on some input variables, treated as input uncertainties. However, computational costs are too high for using Monte Carlo-based sampling. This section aims to develop a surrogate model of the material properties. Because the single output of Puma is intrinsically stochastic, and we are interested in estimating the mean as well as the variability (two first statistical moments), a specific methodology is required. Here, we work on the popular stochastic kriging methodology used for estimating an unknown response surface with uncertainty. It is particularly well suited for Gaussian distributions of the expected outputs, a hypothesis that has been discussed above. However, the model we use is not standard kriging : we do not assume the variability to be uniform and the noise has to be estimated under cost restriction.

First of all, we make a quick literature review for surrogate types for stochastic solvers. Subsequently we introduce the basics of heteroscedastic kriging. Then, the method presented in [40] is described, known as stochastic kriging. In the third part, we introduce a novel surrogate method and provide an algorithm tested on some numerical examples. Finally, we detail how to use the model for performing a sensitivity analysis with a Monte Carlo algorithm.

4.1 State of the art

For estimating statistics of a model, several methods have been used. One direct estimation is method of moments [41] but because it demands a lot of simulations, it is common practice for performing Uncertainty Quantification (UQ) to build a surrogate model of the QoI. An overview of the different surrogate modeling strategies for Uncertainty Propagation is proposed in [42] and [43].

Polynomial chaos expansions (PCE) [44] model the output as a polynomial approximation made of a sum multivariate orthogonal polynomials, where the variable is a random variable. Because we can not generate an infinite basis of polynomials, the mean model is supposed to be a truncated linear combination of these basis polynomials. Typically Legendre polynomials are used for uniform distributions and Hermite polynomials for standard Gaussian distributions. Coefficients are estimated from the existing experimental design. If PCE is easy to evaluate once built, it become less practical when the number of dimensions increases because the number of coefficients increases exponentially. Low-rank tensor approximations (LRA) consider in contrast the metamodel as a sum of rank-one functions, the latter being products of univariate functions. Then, the estimation cost is only linear with the number of dimensions. Note that PCE and LRA are not by themselves predictors, but are used in other methodes such as Generalized additive models (GAM) [45, 46], and can perform sensitivity analysis [42].

Other methods usually work under the same objectives as [47, 40], usually focusing the prediction efforts on same statistics (mean, variance and quantiles) but the surrogating techniques may differ; in [48] GAM models were used to predict the mean and the variance of the conditional distribution. In [49] the mean function was predicted by assuming that the output is a mixture of normal distributions.

Several artificial neural network (ANN) approaches have been also developed to approximate time consuming simulations in structural mechanics [50]. In [51] different network architectures are presented. Other popular surrogate model techniques include neural networks and support vector machines [52].

Kriging methods have proved to be able of performing forward uncertainty propagation [53], notably in UQ and sensitivity analysis context. It predicts the value of a function (which can be

non parametric), by computing a weighed sum over the values of the function at design points, where the function value is known or estimated. Kriging is based on Bayesian inference : the algorithm assumes that the model output $Y(\mathbf{x})$ is a realization of a Gaussian process conditioned on some training values ([54, 55]). Combining a *prior* distribution and a likelihood optimization, a Gaussian process can then predict function value at new locations.

In this paper, we will focus on Kriging method. The reasons for this choice is that a first analysis campaign has shown that our solver’s output is gaussian-like, which is well treated by Kriging. Furthermore, this type of surrogate has proven to be general enough for predicting complex statistical models with low dimensionality. We also found Kriging framework as quite natural for UQ and adapted to what we intend to do. Finally, it already has a wide open-source contribution of usable codes that can be adjusted to our needs.

4.2 Kriging basics

We aim at modeling an unknown function $y(\mathbf{x})$ where $\mathbf{x} = (x_1, \dots, x_d)^T$ is a set of design parameters. In this study’s scope, \mathbf{x} might account for material porosity, gas thermal conductivity, fibers’ orientation etc. and $y(\mathbf{x})$ the associated global thermal conductivity.

Classical Feature Regression approach assumes the observed response obtained from the j^{th} replication at \mathbf{x} is given by the model :

$$\mathcal{Y}_j(\mathbf{x}) = \boldsymbol{\beta}^T \mathbf{f}(\mathbf{x}) + \varepsilon_j(\mathbf{x}) \quad (12)$$

where $\{f_j; j = 1, \dots, P\}$ are known base functions called features and $\boldsymbol{\beta}$ is a vector of unknown parameters. Because we are interested in stochastic simulations, one adds a variability term $\varepsilon_j(\mathbf{x})$, sometimes called *intrinsic* noise because it is inherent to the sampling simulation variability. $\varepsilon_j(\mathbf{x})$ is the realization of a gaussian random variable, which variance may depends on \mathbf{x} : $\varepsilon(\mathbf{x}) \sim \mathcal{N}(0, V(\mathbf{x}))$. If V is uniform then this noise is referred to as *homoscedastic*, and *heteroscedastic* if it varies. Because we address heteroscedastic noise problems, we prefer considering replication based models rather than single-run design.

Feature Regression is a natural extension of a simple linear regression that can be applied to non-linear cases, often more complex to handle. However, features are provided by hand after looking at available training points. Therefore, these functions must either be guessed or explored before regression, which strictly reduces possibilities, especially when the number of dimensions increases [3].

Kriging (or Gaussian Process modeling) is a stochastic method used for regression. The *posterior* GP $g(x)$ is completely specified by its mean function $\mathbf{m}(\mathbf{x})$ and covariance function $\Sigma(\mathbf{x}, \mathbf{x}')$:

$$\mathbf{m}(\mathbf{x}) = \mathbb{E}[g(\mathbf{x})] \quad (13)$$

$$\Sigma(\mathbf{x}, \mathbf{x}') = \mathbb{E}[(g(\mathbf{x}) - \mathbf{m}(\mathbf{x}))(g(\mathbf{x}') - \mathbf{m}(\mathbf{x}'))] \quad (14)$$

The remainder of this subsection describes how to build these functions.

From Feature Regression, one casts the deterministic problem into the GP statistical framework. The stochastic simulation output surface on replication j at design point \mathbf{x} is now given by :

$$\mathcal{Y}_j(\mathbf{x}) = \boldsymbol{\beta}^T \mathbf{f}(\mathbf{x}) + \mathcal{M}(\mathbf{x}) + \varepsilon_j(\mathbf{x}) \quad (15)$$

where $\mathcal{M}(\mathbf{x})$ is a realization of a mean zero random field with $\Sigma_M(\mathbf{x}, \mathbf{x}')$ as covariance function (*ie.* $\Sigma_M(\mathbf{x}, \mathbf{x}') = Cov[(\mathcal{M}(x)), (\mathcal{M}(x'))]$). Introducing \mathcal{M} permits to take *spatial correlation*

into account : if \mathbf{x} and \mathbf{x}' are near in design space, then $\mathcal{M}(\mathbf{x})$ and $\mathcal{M}(\mathbf{x}')$ should be near in output space.

Let $\mathbf{X} = (\mathbf{x}_i, n_i)_{i=1, \dots, N}$ be the *design of experiment (DoE)* of our model, where n_i is the number of simulation replications taken at design setting \mathbf{x}_i . Let's then note :

$$\bar{\mathcal{Y}}(\mathbf{x}_i) = \frac{1}{n_i} \sum_{j=1}^{n_i} \mathcal{Y}_j(\mathbf{x}_i), \quad (16)$$

where $\mathcal{Y}_j(\mathbf{x}_i)$ is the output of the j^{th} replication simulation at design point \mathbf{x}_i so that $\mathbf{S}_i = \{\mathcal{Y}_j(\mathbf{x}_i), i = 1, \dots, n_i\}$ is the set of simulation outputs at design point \mathbf{x}_i , and let $\bar{\mathcal{Y}} = (\bar{\mathcal{Y}}(\mathbf{x}_1) \dots \bar{\mathcal{Y}}(\mathbf{x}_N))^T$.

From the *DoE*, we aim at predicting $Y(\mathbf{x}_0) = \beta^T \mathbf{f}(\mathbf{x}_0) + \mathcal{M}(\mathbf{x}_0)$ at any \mathbf{x}_0 of interest. Let's remark that

$$\bar{\mathcal{Y}}(\mathbf{x}_i) = Y(\mathbf{x}_i) + \bar{\varepsilon}(\mathbf{x}_i)$$

That is, $\bar{\mathcal{Y}}(\mathbf{x}_i)$ is the mean estimate of QoI obtained at design point \mathbf{x}_i plus the simulation sampling error associated $\bar{\varepsilon}(\mathbf{x}_i)$.

Let's note :

- Σ_M the $N \times N$ correlation covariance matrix implied between design points, with (p, q) element equal to $Cov[\mathcal{M}(\mathbf{x}_p), \mathcal{M}(\mathbf{x}_q)]$
- $\Sigma_M(\mathbf{x}_0, \cdot) = (Cov[\mathcal{M}(\mathbf{x}_0), \mathcal{M}(\mathbf{x}_1)], \dots, Cov[\mathcal{M}(\mathbf{x}_0, \cdot), \mathcal{M}(\mathbf{x}_N)])^T$ the $N \times 1$ vector of correlation covariance between the point of interest \mathbf{x}_0 and the design points
- Σ_ε the $N \times N$ covariance matrix implied by the intrinsic noise over design points, with (p, q) element equal to $Cov[\bar{\varepsilon}(\mathbf{x}_p), (\bar{\varepsilon}(\mathbf{x}_q)]$.
- F the $N \times P$ matrix with elements $F_{pq} = f_q(\mathbf{x}_p)$

For fixed unknown parameters (eg. β , $\Sigma_M(\cdot, \cdot)$), [3] shows that for linear predictor, the mean squared error optimal predictor and the associated mean squared error are given by

$$\hat{Y}(\mathbf{x}_0) = \beta^T \mathbf{f}(\mathbf{x}_0) + \Sigma_M(\mathbf{x}_0, \cdot)^T (\Sigma_M + \Sigma_\varepsilon)^{-1} (\bar{\mathcal{Y}} - F\beta) \quad (17)$$

$$MSE(\mathbf{x}_0) = \Sigma_M(\mathbf{x}_0, \mathbf{x}_0) - \Sigma_M(\mathbf{x}_0, \cdot)^T (\Sigma_M + \Sigma_\varepsilon)^{-1} \Sigma_M(\mathbf{x}_0, \cdot) \quad (18)$$

In actual application, the parameters have to be estimated. The estimation relies on an optimization problem, generally trying to maximize a function standing for the chance to obtain the training values given how the model is built. Two main objective functions are usually used: the likelihood function and the cross-validation error. But before describing these methods, the model has have more structure, which comes with $\Sigma_M(\cdot, \cdot)$ (which is for now completely unknown).

4.2.1 Covariance functions and Trend

A common way to structure the covariance matrix is making the assumption it relies on a few hyperparameters, so that we can write $\Sigma_M(\mathbf{x}, \mathbf{x}') = \tau^2 k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a vector of matrix dimension. For k to be a valid correlation function, it needs to be symmetric positive semi-definite. Furthermore, we may assume k to be stationary, which means it depends only on the difference between variables :

$$\Sigma_M(\mathbf{x}, \mathbf{x}') = \tau^2 k(\mathbf{x} - \mathbf{x}'; \boldsymbol{\theta}). \quad (19)$$

Finally, we assume $k(\mathbf{0}, \boldsymbol{\theta}) = 1$ and $k(\mathbf{w}; \boldsymbol{\theta}) \xrightarrow{|\mathbf{w}| \rightarrow \infty} 0$. Then, τ^2 can be interpreted as the variance of $\mathcal{M}(\mathbf{x})$ for all \mathbf{x} .

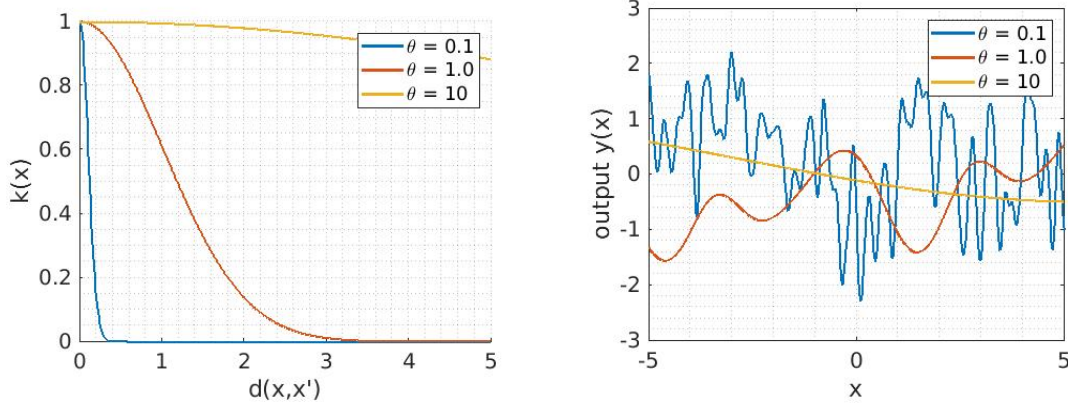
Below we describe several families of such functions, sometimes called *kernels*. They can of course be combined by basic algebraic operations. Furthermore, the kernels described below employ $\boldsymbol{\theta}$ as *characteristic length vector* [3].

- **Exponential kernel :**

$$k(x, x'; \theta) = \exp \left[-\frac{|x - x'|}{\theta} \right] \quad (20)$$

- **Gaussian kernel or squared exponential :**

$$k(x, x'; \theta) = \exp \left[-\frac{1}{2} \left(\frac{|x - x'|}{\theta} \right)^2 \right] \quad (21)$$



(a) Kernel distribution from eq (21) for several θ . (b) Random sampling from a GP with gaussian covariance function.

Figure 8: Gaussian kernel properties.

- **Matérn family kernel :** the general form of Matérn kernels is

$$k(x, x'; \theta, \nu) = \frac{1}{2^{\nu-1} \Gamma(\nu)} \left(\sqrt{2\nu} \frac{|x - x'|}{\theta} \right)^{\nu} K_{\nu} \left(\sqrt{2\nu} \frac{|x - x'|}{\theta} \right) \quad (22)$$

where K_{ν} is the modified Bessel function of the second kind. Most used values of ν are $\nu = \frac{3}{2}$ and $\nu = \frac{5}{2}$, for which

$$\begin{aligned} k(x, x'; \theta, \frac{3}{2}) &= \left(1 + \frac{\sqrt{3}|x - x'|}{\theta} \right) \exp \left[-\frac{\sqrt{3}|x - x'|}{\theta} \right] \\ k(x, x'; \theta, \frac{5}{2}) &= \left(1 + \frac{\sqrt{5}|x - x'|}{\theta} + \frac{5|x - x'|^2}{3\theta^2} \right) \exp \left[-\frac{\sqrt{5}|x - x'|}{\theta} \right] \end{aligned} \quad (23)$$

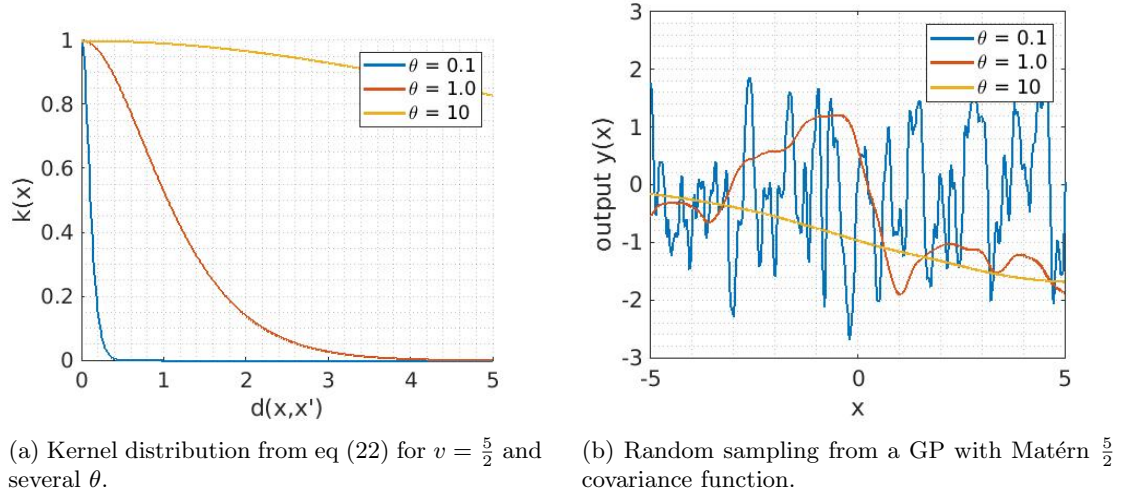
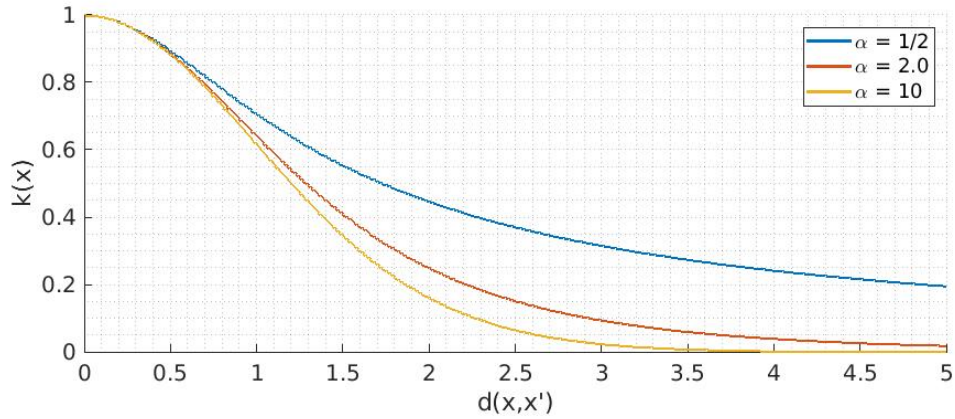


Figure 9: Matérn kernel properties.

- Rational Quadratic kernel :

$$k(x, x'; \alpha, \theta) = \left(1 + \frac{|x - x'|^2}{2\alpha\theta^2}\right)^{-\alpha} \quad (24)$$


 Figure 10: Rational Quadratic kernel distribution from eq (24) for $\theta = 1$ for several α .

We have made the description on a one-dimension input space for readability, but they can be easily extended to multi-dimensional models from one-dimensional correlation families using one of the following constructions [56] :

- Ellipsoidal kernel [3] :

$$k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}) = k\left(\sqrt{\sum_{i=1}^d \left(\frac{x_i - x'_i}{\theta_i}\right)^2}\right)$$

- **Separable kernel** [54] :

$$k(\mathbf{x}, \mathbf{x}'; \boldsymbol{\theta}) = \prod_{i=1}^d k(x_i, x'_i; \theta_i)$$

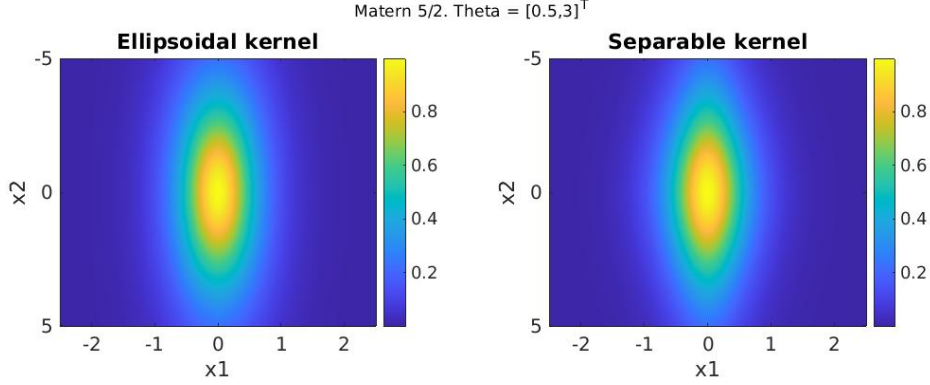


Figure 11: Two 2D covariance distribution based on Matérn $\frac{5}{2}$ kernel. First one uses an ellipsoidal kernel, and second one uses a separable kernel. For both figures $\boldsymbol{\theta} = (0.5, 3)^T$

Finally, the optimization depends on the choice of the trend $\boldsymbol{\beta}^T \mathbf{f}(x_0)$ in (15). To cite shortly what are the most used trends [57],

- in *simple kriging* the trend is known as a specified function of \mathbf{x} (no estimation of $\boldsymbol{\beta}$)
- in *ordinary kriging*, the trend is an unknown constant $\boldsymbol{\beta}^T \mathbf{f}(x_0) = \beta_0$
- in *quadratic kriging*, the trend is a polynomial function of \mathbf{x} of given order with unknown coefficients
- in *universal kriging*, the trend is a more general linear combination of prescribed arbitrary functions.

Now we have a framework for optimization. As we have said before, the aim is trying to find the set of parameters and hyperparameters that fits at best the observations, *ie.* the training set $(\mathbf{X}, \bar{\mathcal{Y}})$

From now, we will work under the following assumption (remind assumption 1):

Assumption 2. *The random field \mathcal{M} is a stationary Gaussian random field and $\{\varepsilon_s(\mathbf{x}_i)\}_{s=1, \dots, n_i}$ are *i.i.d* $\mathcal{N}(0, V(\mathbf{x}_i))$, independent of $\varepsilon_t(\mathbf{x}_j)$ for all t and $j \neq i$, and independent of \mathcal{M} .*

The independence of $\varepsilon_s(\mathbf{x}_i)$ and $\varepsilon_s(\mathbf{x}_j)$ (for $i \neq j$) means we don't use Common Random Numbers for the samples generations. This also means Σ_ε is diagonal.

4.2.2 Optimization Functions

Marginal Likelihood

For a given training set, the log marginal likelihood is defined as followed :

$$\mathcal{L}_{ML}(\boldsymbol{\theta}) = p(\bar{\mathcal{Y}}|\mathbf{X}, \boldsymbol{\theta}) \quad (25)$$

i.e. for a hyperparameter the probability to observe $\bar{\mathcal{Y}}$ at \mathbf{X} . [40] shows that for a gaussian process, it can be written as:

$$\begin{aligned} \mathcal{L}_{ML}(\boldsymbol{\beta}, \tau^2, \boldsymbol{\theta}) &= -\frac{N}{2} \ln(2\pi) \\ &\quad - \frac{1}{2} \ln [|\tau^2 \mathbf{K}_M(\boldsymbol{\theta}) + \Sigma_\varepsilon|] \\ &\quad - \frac{1}{2} (\bar{\mathcal{Y}} - F\boldsymbol{\beta})^T [\tau^2 \mathbf{K}_M(\boldsymbol{\theta}) + \Sigma_\varepsilon]^{-1} (\bar{\mathcal{Y}} - F\boldsymbol{\beta}) \end{aligned} \quad (26)$$

where $\mathbf{K}_M(\boldsymbol{\theta})$ is the correlation matrix of M over design points. [3] gives methods and references for maximizing log marginal likelihood, and figure 12 gives a one-dimensional kriging example with two local optima along with the GP prediction associated. It is computed for a homoscedastic problem with simple kriging (zero mean).

Figure 12(b) and 12(c) show the role of the parameters. In 12(b), the lower length scale θ takes variation as trend variation, whereas in 12(c), the high noise coefficient corresponds to a much simpler model but with more intrinsic variance. This trend illustrates the importance of choosing hyperparameters and how critical it is to choose the design points.

Cross Validation error

Cross-validation method rests on using the DoE for evaluating the choice of parameters which is optimal [3]. Let's consider :

$$\log p(y_i | X, \bar{\mathcal{Y}}_{-i}, \tau^2, \boldsymbol{\theta}) = -\frac{1}{2} \log \sigma_i^2 - \frac{(y_i - \mu_i)^2}{2\sigma_i^2} - \frac{1}{2} \log 2\pi$$

where $\bar{\mathcal{Y}}_{-i}$ accounts for all model outputs excepts number i , μ_i and σ_i^2 respectively the predictive mean and predictive variance given by (15) trained on $(\mathbf{X}_{-i}, \bar{\mathcal{Y}}_{-i})$. The so-called Leave-One-Out predictive probability is :

$$\mathcal{L}_{LOO}(\mathbf{X}, \bar{\mathcal{Y}}, \tau^2, \boldsymbol{\theta}) = \sum_{i=1}^N \log p(y_i | X, \mathbf{y}_{-i}, \boldsymbol{\theta}) \quad (27)$$

If it seems more computationally expensive than Marginal Likelihood because it demands inverting $\tau^2 \mathbf{K}_M(\boldsymbol{\theta}) + \Sigma_\varepsilon$ for each design point left out, methods have been developed to reduce the computation cost (because only a row and column change when moving from i to j) using inversion partitioning [40].

Finally, the kriging optimization problem writes :

Problem 1.

$$\begin{aligned} &\text{minimize} \quad \mathcal{L}(\dots, \tau^2, \boldsymbol{\theta}) \\ &\text{s.t} \quad \Sigma_\varepsilon > \Sigma_{\varepsilon, \min}, \quad \tau > 0, \quad \boldsymbol{\theta} > 0 \\ &\text{where} \quad \mathcal{L} \text{ is a valid function of hyperparameters and other information} \\ &\text{with} \quad \Sigma_\varepsilon \text{ the training covariance matrix} \end{aligned}$$

Note that we have added $\Sigma_\varepsilon > \Sigma_{\varepsilon, \min}$ for stability reasons matrices inversion (nugget effect). If you have *a priori* knowledge of the function you want to model, you might want to add boundaries to τ and $\boldsymbol{\theta}$ for imposing a noise level and a characteristic distance.

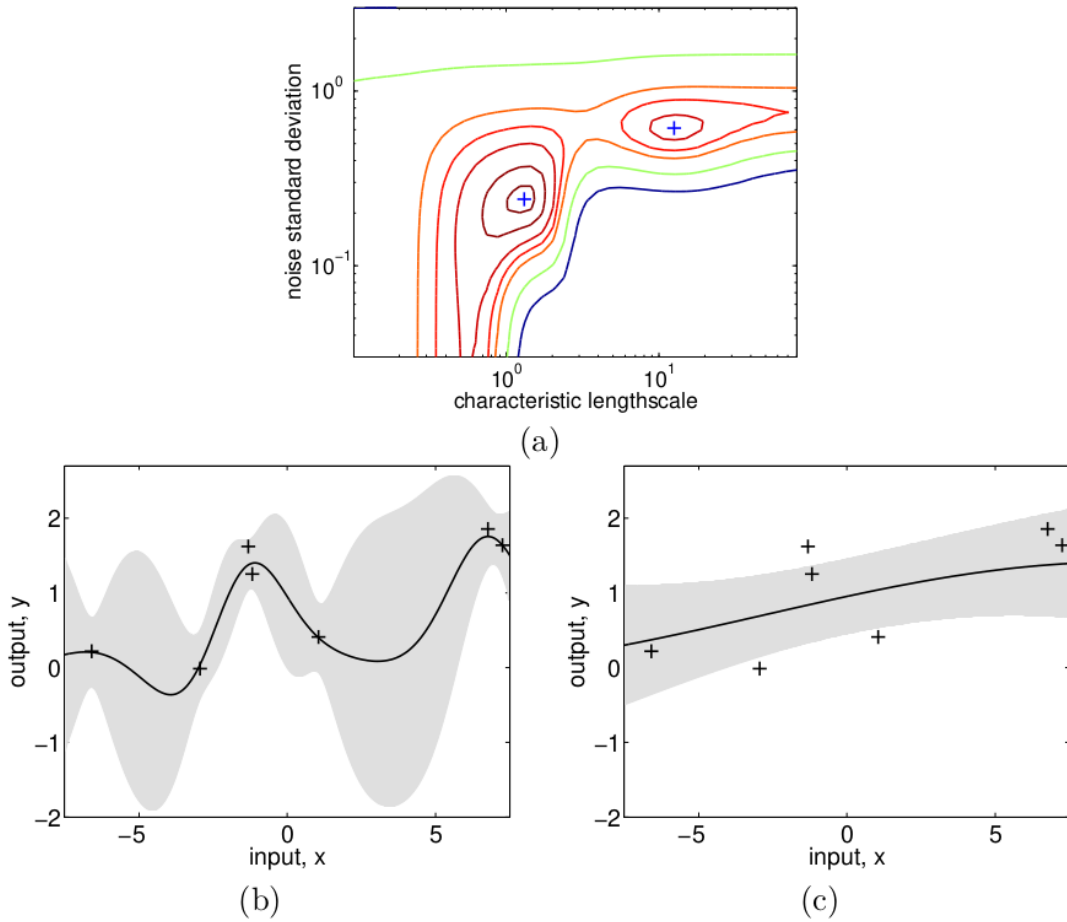


Figure 12: Figure (a) shows the 2 parameters likelihood surface plot as a function of θ and τ^2 . Two local optima are noticed, indicated by '+'. Figure (b) shows the GP response (mean and 95% interval confidence) for the most to the left optima. Figure (c) shows the GP response for the second optima. Credits to [3]

4.3 Stochastic Kriging

Given above was one of the standard methods for applying kriging. However, you may notice one assumption has been made: Σ_ε is known. However, in some situations, it may not be accurate. *Stochastic Kriging* (SK) methodology proposed by [40] is a metamodeling tool for approximating a mean response surface implied by a stochastic simulation. It is an extension of the kriging methodology capable of accounting for sampling uncertainty (Σ_ε , also called *intrinsic uncertainty*) inherent in a stochastic simulation in addition to the response-surface uncertainty (Σ_M , also called *extrinsic uncertainty*).

Let's study the estimation of Σ_ε and the effects of such a method on the predictions.

One basic approach is given as follows :

- Because V is unknown, it has to be estimated. We then build a kriging metamodel for

variance, with the associated predictor \hat{V}_{SK} .

- V is not observable, even at design points. The model is then trained with :

$$\mathcal{S}^2(\mathbf{x}_i) = \frac{1}{n_i - 1} \sum_{j=1}^{n_i} (\mathcal{Y}_j(\mathbf{x}_i) - \bar{\mathcal{Y}}(\mathbf{x}_i))^2 \quad (28)$$

- Because \mathcal{S}^2 is observed and not V , there are extrinsic and intrinsic uncertainties. However, because [40] is not interested in V except as it impacts the mean prediction, intrinsic uncertainties are ignored. Therefore \hat{V}_{SK} is an interpolator, $\hat{V}_{SK}(\mathbf{x}_i) = \mathcal{S}^2(\mathbf{x}_i)$ at design points.

With these notations, [40] shows the following theorem.

Theorem 1. Let $\hat{\Sigma}_\varepsilon = \text{Diag}\{\frac{\hat{V}_{SK}(\mathbf{x}_1)}{n_1}, \dots, \frac{\hat{V}_{SK}(\mathbf{x}_N)}{n_N}\}$ and

$$\hat{Y}_{SK}(\mathbf{x}_0) = \boldsymbol{\beta}^T \mathbf{f}(\mathbf{x}_0) + \Sigma_M(\mathbf{x}_0, \cdot)^T (\Sigma_M + \hat{\Sigma}_\varepsilon)^{-1} (\bar{\mathcal{Y}} - F\boldsymbol{\beta}) \quad (29)$$

Then under Assumption 2, $\mathbb{E} [\hat{Y}_{SK} - Y(\mathbf{x}_0)] = 0$

That is to say, estimating the intrinsic variance doesn't introduce prediction bias. The confidence of the prediction can be questionable, but [40] argues that for n_i not too small (typically superior to 10), the MSE only slightly increases.

4.4 Joint metamodeling

In this section, we address the issue of surrogating V as well as Y . Some problems demands not only to estimate the mean of a simulation but also the variance of the stochastic simulation. This is especially the case where the objective is to identify the influence of stochasticity source on the variation of the quantity of interest. For example, concerning PuMA, we want to separate the contribution of the stochasticity and the uncertain parameters on the output variance with sensitivity analysis method. Thus stochasticity of the solver must be modeled too.

A first approach consists in building polynomial approximation of the mean and variance separately [58], based on repeated calculations with the same set of controllable input variables. This dual modeling approach has been successfully applied for robust conception problems. However, our purpose here is to fit accurately both mean and variance. Within this context, it has been shown that the dual model performs less than the joint model [59]. For example, [60] applies a specific joint metamodel developed in [61] in the context of accidental scenario in a pressurized water reactor.

Below is described the method we will use, that fits both mean and variance extending Stochastic Kriging. We choose to continue developing a repetition based approach rather than one-sampling strategy.

4.4.1 Variance metamodel

Because we are interested both in mean and variance prediction, one needs to be cautious about the variance metamodel. Based on an idea in [62], the variance metamodel is no longer built as an interpolator : we add some extrinsic noise. Among the noise estimators provided in the article, we choose estimation via *Bootstrap Sample Variances*, because the comparison with others methods

proves it to be adapted to variable replication allocation schemes (which have not been discussed yet), with performance almost every time comparable to the best estimator. Furthermore, it is easy to understand and can be applied even when the number of replications is low. We draw B bootstrap samples each of size n_i from \mathbf{S}_i , resulting in B bootstrap samples $\{\mathbf{S}_i^b\}_{b=1,\dots,B}$ where $\mathbf{S}_i^b = \{\mathcal{Y}_j^b(\mathbf{x}_i)\}_{j=1,\dots,n_i}$. Then the b th sample variance $s_{i,b}^2$ at design point \mathbf{x}_i is computed with \mathbf{S}_i^b . Finally, the variance estimate of the sampling variance is :

$$\hat{V}_{s^2}(\mathbf{x}_i) = \frac{1}{B-1} \sum_{b=1}^B (s_{i,b}^2 - \overline{s_{i,\cdot}^2})^2 \quad \text{with} \quad \overline{s_{i,\cdot}^2} = \frac{1}{B} \sum_{b=1}^B s_{i,b}^2 \quad (30)$$

From this, we can build a heteroscedastic predictor \hat{V}^* for the variance as we have described before, with \mathcal{S}^2 (resp. \hat{V}_{s^2}) as an estimator of the variance (resp. noise) at the design points.

4.4.2 Mean metamodel

Even if we deliberately choose to develop the two metamodels rather independently, we still introduce some bonds between them. In particular, in (29), $\hat{\Sigma}_\varepsilon^*$ becomes

$$\hat{\Sigma}_\varepsilon^* = \text{Diag}\left\{\frac{\hat{V}^*(\mathbf{x}_1)}{n_1}, \dots, \frac{\hat{V}^*(\mathbf{x}_N)}{n_N}\right\} \quad (31)$$

where now \hat{V}^* is not anymore an interpolator. Finally, the GP predictor for the mean is given by :

$$\hat{Y}^*(\mathbf{x}_0) = \beta^T \mathbf{f}(\mathbf{x}_0) + \Sigma_M(\mathbf{x}_0, \cdot)^T (\Sigma_M + \hat{\Sigma}_\varepsilon^*)^{-1} (\bar{\mathcal{Y}} - F\beta) \quad (32)$$

Note that the variance and mean metamodels hyperparameters optimization processes are separated but not uncorrelated. Doing this we differ from other methods, usually iterative (e.g. [63]).

4.4.3 Allocation methodology

Until now, we have not discussed replications. In particular, real applications are limited by the total computation cost, in money or time. Because we focus on computational applications, budget will be measured with computation time. Furthermore, we will work under the following assumption.

Assumption 3. *Every simulation costs the same budget, no matter the set of input parameter.*

This helps developing a strategy for cost management. From now, a budget of C distributed among several simulations ensues running C simulations, no matter the allocation scheme. We choose to try two strategies. We remind that the DoE is $\mathbf{X} = (\mathbf{x}_i, n_i)_{i=1,\dots,N}$. In particular under assumption 3, every allocation scheme should verify

$$\sum_{i=1}^N n_i \leq C \quad (33)$$

Finally, note that increasing the number of replications at a design point benefits the two metamodels we try to build.

- it decreases the noise level on the estimation of the variance. Indeed recall expression of $\mathcal{S}^2(\mathbf{x}_i)$ equation (28) : $\mathcal{Y}_j(\mathbf{x}_i)$ are n_i *iid.* random variables of law $\mathcal{N}(Y(\mathbf{x}_i), V(\mathbf{x}_i))$, and $\bar{\mathcal{Y}}(\mathbf{x}_i)$ as their mean follows the law $\mathcal{N}(Y(\mathbf{x}_i), \frac{V(\mathbf{x}_i)}{n_i})$. Then (find the details in appendix A) :

$$\mathbb{E} [\mathcal{S}^2(\mathbf{x}_i)] = V(\mathbf{x}_i) \quad \text{Var}(\mathcal{S}^2(\mathbf{x}_i)) = \frac{2}{n-1} V(\mathbf{x}_i)^2 \quad (34)$$

- it decreases the noise on the mean metamodel because of the expression of $\hat{\Sigma}_\varepsilon^*$ in equation (31).

Equal allocation scheme The most basic strategy is distributing budget equally among all design points. Because this strategy does not take advantage of any prior knowledge of the objective function we want to model, it will not necessarily lead to the best results. Nonetheless it does not need any extra information, and is decided before running any simulation which is non negligible.

Unequal IMSE minimizing allocation scheme This approach stands as one of the possible methods consisting in minimising one of the convergence criteria of the metamodel. The following IMSE (integral min square root error) strategy has been proposed in [40], and we will discuss how to simplify it in order to use it practically.

Let \mathcal{X} be the d -dimensional space of interest from which have been chosen the design points \mathbf{x}_i . Let $\mathbf{n} = (n_1, \dots, n_N)$. The problem can be written as follows :

Problem 2.

$$\begin{aligned} \text{minimize} \quad & \text{IMSE}(\mathbf{n}) = \int_{\mathcal{X}_0} \text{MSE}(\mathbf{x}_0; \mathbf{n}) d\mathbf{x}_0 \\ \text{s.t} \quad & \mathbf{n} \in \mathbb{N}^N, \mathbf{n}^T \mathbf{1}_N \leq C \\ \text{where} \quad & \text{MSE}(\mathbf{x}_0; \mathbf{n}) = \Sigma_M(\mathbf{x}_0, \mathbf{x}_0) - \Sigma_M(\mathbf{x}_0, \cdot)^T (\Sigma_M + \Sigma_\varepsilon(\mathbf{n}))^{-1} \Sigma_M(\mathbf{x}_0, \cdot) \\ \text{with} \quad & \Sigma_M \text{ and } \Sigma_\varepsilon \text{ the optimized covariance} \\ & \text{associated to the joint metamodel} \end{aligned}$$

Let $\Sigma(\mathbf{n}) = \Sigma_M + \Sigma_\varepsilon(\mathbf{n})$. [40] shows the solution with the integrity constraint relaxed satisfies $n_i^* \propto \sqrt{V(\mathbf{x}_i) C_i(\mathbf{n}^*)}$ where $C_i(\mathbf{n}) = [\Sigma(\mathbf{n})^{-1} \mathbf{W} \Sigma(\mathbf{n})^{-1}]_{ii}$ and \mathbf{W} is a $N \times N$ matrix $W_{ij} = \int_{\mathcal{X}_0} k(\mathbf{x}_i, \mathbf{x}_0) k(\mathbf{x}_j, \mathbf{x}_0) d\mathbf{x}_0$

Under the assumption that N is large enough and training design are equally spatially distributed, we show in appendix B :

$$n_i = C \frac{\sqrt{V_i}}{\sum_{j=1}^N \sqrt{V_j}} \quad (35)$$

Note that Eq. 35 is derived only from a problem on the mean metamodel. However, you can easily persuade yourself it is a relevant allocation choice as well for the variance metamodel (see equation 34).

In practice, V is unknown. Therefore, we develop a strategy based on several steps for estimating V before building the joint metamodel, each step bringing some more insight into the variability of the problem. Since it does not need too much explanation, you will find it in the general algorithm 1 (step 3-4).

4.4.4 Joint Metamodel Algorithm

Finally, we need to discuss how to choose the design points. Because kriging is based on inferring thanks to "close" training points, we naturally choose a space filling design of experiment, thanks to a Latin Hypercube Sampling (LHS) for example. Without any prior information on the function we model, there is not much more to say. Yet note you can only make prediction on a finite domain : $MSE(\mathbf{x}_0) \xrightarrow{d(\mathbf{x}_0, \mathbf{X}) \rightarrow \infty} \infty$ where $d(\mathbf{x}_0, \mathbf{X})$ is the distance between \mathbf{x}_0 and the training set, so that it is not relevant to predict out of the convex hull of the training set. It is even recommended not to predict near the boundary of the training set, because the prediction will mainly rely on interior points (if a variation occurs next to the boundary, one has less luck to predict it).

In case we have some information on the function (prior to any simulation or after a few simulations), heuristically it is preferable to choose points where the function is not linear over a hyperparameter scale, so that variations are well predicted

Algorithm 1: Joint metamodel algorithm

- Step 1 :** For a given number N , generate the training set;
Step 2 : Allocate a minimum number of repetitions for each training point. 2 repetitions is a minimum (so that variance can be estimated), but higher numbers are preferable (eg. 5);
Step 3 : Choose the total budget C , and a number of allocation steps r ;
Step 4 : **for** i *between 1 and* r **do**
 Define $C_{temp} = \frac{iC}{r}$ as the temporary total budget. Then allocate C_{temp} repetitions under the Unequal IMSE minimizing allocation scheme, using the already completed simulations to make variance estimation;
end
Step 5 : With the design of experiment completed, build the joint metamodel (\hat{Y}^*, \hat{V}^*)
-

In our algorithm, we have made only one step for defining the training set. Other strategies may rely on an iterative process : between Step 2 and Step 3 or between iterations of Step 4, one might compute an intermediary model and add training designs at specific locations (typically where the intermediary MSE is high). Because our physical model proves to have variations that vary smoothly in the design space, we didn't look deeper in this direction, since LHS designs are satisfying.

4.5 Illustration

To illustrate the methodology described above, we consider the 3D-Hartmann function, defined as follows :

$$m(\mathbf{x}) = - \sum_{i=1}^4 \exp \left(- \sum_{j=1}^3 A_{ij} (x_j - P_{ij}^2) \right) \quad (36)$$

$$\text{where } \alpha = \begin{pmatrix} 1.0 \\ 1.2 \\ 3.0 \\ 3.2 \end{pmatrix}, \quad A = \begin{pmatrix} 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \\ 3.0 & 10 & 30 \\ 0.1 & 10 & 35 \end{pmatrix} \quad \text{and} \quad P = 10^{-4} \begin{pmatrix} 3689 & 1170 & 2673 \\ 4699 & 4387 & 7470 \\ 1091 & 8732 & 5547 \\ 381.0 & 5743 & 8828 \end{pmatrix}$$

We aim at proving that our joint metamodel strategy works as intended and gives prediction as accurate as other methods, both on the mean function and on the variance function. Furthermore,

we try to illustrate the role of budget and number of design points in the accuracy of the prediction.

The j^{th} simulation output on design point \mathbf{x}_i is generated as $\mathcal{Y}_j(\mathbf{x}_i) = m(\mathbf{x}_i) + \varepsilon_j(\mathbf{x}_i)$, where $\varepsilon_j(\mathbf{x}_i)$ are defined as in Assumption 2, with $V(\mathbf{x}) = \delta|m(\mathbf{x})|$ where δ is a magnitude parameter.

In order to compare models, we define the empirical root mean squared error (ERMSE) over R check points noted $(\mathbf{p}_i)_{i=1,\dots,R}$. For a given experiment subscripted e (*ie.* a surrogate built on a training set), the accuracy of one of the estimation surrogate \hat{G}_e compared to the associated theoretical function g is :

$$ERMSE_e = \sqrt{\frac{1}{R} \sum_{i=1}^R \left(\hat{G}_e(\mathbf{p}_i) - g(\mathbf{p}_i) \right)^2} \quad (37)$$

$(\mathbf{p}_i)_{i=1,\dots,R}$ is fixed for all experiments and generated by a LHS in $[0, 1]^3$. Then we can make statistics over several experiments and compare ERMSEs.

k	interpolator	homoscedastic	SK	Joint metamodel
250	m : 1.184	m : 0.256	m : 0.193 V : 0.497	m : 0.185 V : 0.534
200	m : 0.731	m : 0.262	m : 0.186 V : 0.561	m : 0.175 V : 0.508
125	m : 0.399	m : 0.264	m : 0.186 V : 0.576	m : 0.175 V : 0.398
100	m : 0.395	m : 0.278	m : 0.193 V : 0.518	m : 0.183 V : 0.379
80	m : 0.390	m : 0.308	m : 0.252 V : 0.701	m : 0.236 V : 0.395
40	m : 0.337	m : 0.330	m : 0.294 V : 0.615	m : 0.297 V : 0.411
20	m : 0.444	m : 0.443	m : 0.431 V : 1.278	m : 0.448 V : 0.531

Table 1: ERMSE obtained with an equal allocation scheme

Tab 1 and 2 report the obtained results and are generated with the same budget $C = 2e3$ and $\delta = 1$. The training designs are always the same for a given number of training points (among kriging types and experiments). Interpolator (with small nugget effect for stability) is a kriging model that assumes there is no noise (both on simulation output and training set): it is, of course, not the most appropriate model for highly stochastic simulations, but it is still relevant for comparison. The homoscedastic model assumes a uniform unknown noise on data: it is the basics kriging model for stochastic simulation. Interpolator and homoscedastic models unequal allocation scheme directly use the theoretical variance while stochastic kriging and our joint metamodel estimate it as described in algorithm 1. Finally, values are written under the format: mean over 100 experiments of ERMSE on mean function - mean over 100 experiments of ERMSE on variance function (if applicable).

As expected, one may observe trends. First, using the unequal allocation scheme generally improves the quality of mean predictions. However, note that it is not the case at a high number of training points for our joint metamodel. Indeed, because the local variance is estimated as the model progresses, significant errors on estimations of variance with a few samples can lead

k	interpolator	homoscedastic	SK	Joint metamodel
250	m : 1.063	m : 0.238	m : 0.160 V : 0.431	m : 0.197 V : 0.468
200	m : 0.843	m : 0.242	m : 0.159 V : 0.459	m : 0.175 V : 0.426
125	m : 0.372	m : 0.247	m : 0.158 V : 0.455	m : 0.174 V : 0.351
100	m : 0.358	m : 0.257	m : 0.167 V : 0.438	m : 0.167 V : 0.324
80	m : 0.354	m : 0.276	m : 0.210 V : 0.571	m : 0.204 V : 0.358
40	m : 0.302	m : 0.303	m : 0.277 V : 0.579	m : 0.284 V : 0.374
20	m : 0.439	m : 0.438	m : 0.428 V : 1.139	m : 0.426 V : 0.493

Table 2: ERMSE obtained with an unequal allocation scheme

to poor allocation schemes. Still, the final estimation is unsatisfactory because the maximum number of allocations is reduced (budget is fixed).

Second, for all metamodel types, the quality of prediction first decreases and latter increases when the number of training points increases. The first part of this variation is attributable to the lack of information available with a few training points and missing variations. The second part comes from the lack of replications at training points, leading to under-fitting because of large noise. It is not necessarily the case for the homoscedastic model because it tries to fit a large homoscedastic noise to a heteroscedastic model. However, it impacts the joint metamodel all the more it has to estimate the variance in unequal allocation scheme.

Finally, the order of quality between metamodels is respected: for nominal allocations, our joint metamodel performs better than SK, then homoscedastic model than interpolator. What is satisfying is that even learning the variance as new samples come in, the joint metamodel shows similar results for mean prediction than SK (which does not make an estimation of variance). When it does not performs better for mean prediction, a slight quality loss on mean leads to a great gain for variance prediction (for $k = 40$ a mean loss of 3% leads to an improved variance by 35%).

Now we can study the impact of the total budget on estimation accuracy. We fix the number of training points at $N = 80$ and increase the budget. Results are detailed in Table 3.

Budget	5e2	7e2	1e3	2e3	3e3	5e3	7e3	1e4
ERMSE (m)	0.355	0.270	0.250	0.206	0.156	0.127	0.123	0.123
ERMSE (V)	0.621	0.537	0.466	0.349	0.256	0.201	0.176	0.171

Table 3: Impact of total budget on ERMSE of mean and variance metamodel of the joint metamodel for an unequal allocation scheme. 80 training points are used, and each value is the mean of 100 experiments.

As expected ERMSE first quickly decreases when increasing budget, before reaching a convergence value : there will always be error between training points. Yet it shows that you do not necessarily need more training points for better prediction, but increasing budget is more

profitable (but more costly) when comparing to table 2. Note that for this test case, taking $C = 2e3$ was quite reasonable, since large fraction of ERMSE on mean and variance predictions was avoided. Finally, a deal between repetitions and exploration has to be made, so that the allocation is Pareto-optimal.

4.6 Sensitivity analysis

Finally, our analysis of the PuMA simulations takes advantage of a sensitivity analysis. Reminding Section 3, gaussian processes enables to predict output of the model in a given domain. Notably, if one takes the domain as a multi-dimensional interval centered on a nominal vector of parameters, the model can predict not only the nominal output but also outputs if there is uncertainty on parameters. Furthermore, our joint metamodel is also capable of predicting the inherent variance of the output on the domain : under Assumption 1, we can model the stochastic aspect of the QoI. Then we can envisage to make (cheap) stochastic prediction on the uncertainty domain, and estimate Sobol indices.

For this, we will use the approach of [34], that implements a variance-based characterization of the uncertainties induced by the parameters and the simulation. It prescribes a Monte Carlo sampling algorithm for estimating the sensitivity indices.

Let's recall from Section 3 the SH decomposition equation 6. Let (\hat{G}^*, \hat{V}^*) be the mean and variance metamodels associated to G . We assume each component of \mathbf{x} is defined as a nominal value and an uncertainty law (typically $x_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$). $\boldsymbol{\xi}$ also has a its own known law.

Let \mathbb{X} and \mathcal{E} be two sample sets of S independent realizations of the parameters and stochastic parameter. As discussed before, \mathbf{X}_i is a sample of mutually independent parameters, whereas $\boldsymbol{\xi}_i$ is an initialized seed. From these samples, Monte Carlo estimators of expectation and variance are :

$$\widehat{\mathbb{E}}_G = \frac{1}{S} \sum_{i=1}^S \hat{G}(\mathbf{X}_i, \boldsymbol{\xi}_i) \quad (38)$$

$$\widehat{\mathbb{V}}_G = \frac{1}{S} \sum_{i=1}^S \hat{G}(\mathbf{X}_i, \boldsymbol{\xi}_i)^2 - \widehat{\mathbb{E}}_G^2 \quad (39)$$

Now let $\tilde{\mathbb{X}}$ and $\tilde{\mathcal{E}}$ independent replicas of \mathbb{X} and \mathcal{E} . Monte Carlo method gives :

$$\widehat{V}_{par} = \frac{1}{S-1} \sum_{i=1}^S \hat{G}(\mathbf{X}_i, \boldsymbol{\xi}_i) \hat{G}(\tilde{\mathbf{X}}_i, \boldsymbol{\xi}_i) - \widehat{\mathbb{E}}_G^2 \quad (40)$$

$$\widehat{V}_{ch} = \frac{1}{S-1} \sum_{i=1}^S \hat{G}(\mathbf{X}_i, \boldsymbol{\xi}_i) \hat{G}(\mathbf{X}_i, \tilde{\boldsymbol{\xi}}_i) - \widehat{\mathbb{E}}_G^2 \quad (41)$$

Associated sensitivity indices are finally estimated :

$$\widehat{S}_{par} = \frac{\widehat{V}_{par}}{\widehat{\mathbb{V}}_G}, \quad \widehat{S}_{ch} = \frac{\widehat{V}_{ch}}{\widehat{\mathbb{V}}_G}, \quad \widehat{S}_{mix} = 1 - \widehat{S}_{par} - \widehat{S}_{ch} \quad (42)$$

You can find illustration of this method in [34] with Birth-Death and Schlögl examples.

5 Results

This section is devoted to illustrating the main results obtained by applying the UQ-based numerical method to PuMA. Specifically, we discuss the notion of convergence in the case of a stochastic solver such as PuMA. Moreover, we show how PuMA can be used to set up a priori best parameters to perform a tomography efficiently. We will then reuse this study for preparing the metamodel simulation campaign. The remaining section illustrates several detailed analyses, which allows a thorough understanding of the influence of several physical parameters of interest, putting into practice the mathematical background we have introduced before.

Note that in our study's framework, for a given fiber material (resp. gas), λ_f (resp. λ_g) is only a function of temperature T . In particular, pressure is not a parameter if we do not consider the rarefied medium.

This work focuses on the impact of p , λ_f and stochasticity for several gases and temperatures. The application of the surrogate-based strategy is straightforward. For each gas and temperature (discretized arbitrarily), we perform an independent UQ study by building a DOE with the two uncertain parameters, *i.e.* p and λ_f .

5.1 Parameters convergence in PuMA

PuMA can import images from tomography or can yield artificial material based on some geometrical and mechanical constraints. The present work deals mainly with the second option. Here, we focus specifically on the influence of the mesh resolution and other numerical parameters to attain converged prediction of the quantities of interest. This section completes Section 2 for defining convergence suited to PuMA and illustrates some results driving the choice of the best resolution in terms of computational cost and accuracy.

5.1.1 Choice of the numerical resolution and tomography

Question: *How can we use PuMA to get relevant indications to perform a tomography ? This point raises questions about the choice of the domain size, the number of voxels to use for representing the fibers and the domain, and finally about a procedure to assess the resolution needed to measure average properties of the material.*

Let us introduce the function $f(x, \xi)$, where x are the geometrical parameters, ξ is the set of uncertain parameters to take into account, f is stochastic. We consider that R_{phy} is known since it comes from the physical fiber under consideration. The problem reads as follows:

Minimize r_{vx} and Dom_{phy}
 with the constraint that $Err(f(r_{vx}, D_{phy}, \cdot)) < Criterion1$
 with the constraint that $|\bar{f}(r_{vx}, D_{phy}, \cdot) - \bar{f}(r_{vx}^{max}, D_{phy}^{max}, \cdot)| < Criterion2$
 with the constraint that $d_{vx} = Dom_{phy}/l_{vx}^{phy}$, where $l_{vx}^{phy} = R_{phy}/r_{vx}$

Once this problem is solved, one could say that the tomography should have a resolution lower than "Voxel Length" for a physical domain of Dom_{phy} .

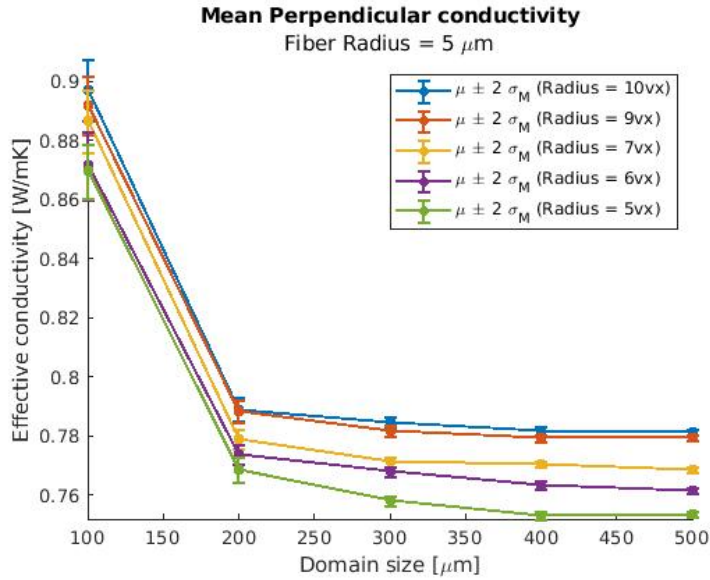
Below we describe our methodology for choosing r_{vx} and D_{phy} . One can follow a resampling procedure: for a certain number of repetitions of a set of parameters, the error of the mean of these simulations must be lower than certain a threshold. One uses two iterative loops in order to find out the optimal parameters for PuMA, which are R_{vx} and D_{phy} . The procedure is described below :

1. Define the maximum calculable d_{vx} . In our case it is $d_{vx} = 1000vx$.

2. Choose a range of r_{vx} that will be tested. A preliminary study not shown here and a visual evaluation of figure 5 have led us to study radius from $5vx$ to $10vx$.
3. Deduct possible D_{phy} and associated d_{vx} that will be computed. We also require a minimal D_{phy} of $100\mu m$.
4. Choose a nominal set of parameters.
5. Simulate one hundred times the stochastic QoI at the nominal point for all couples of numerical parameters (r_{vx}, D_{phy}) . It gives an empirical mean and error of this estimation (the error is given by bootstrap sampling).

5.1.2 Results

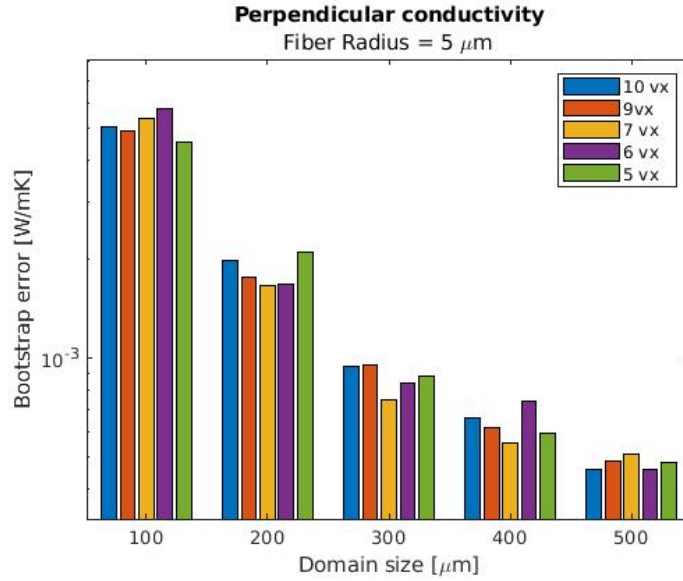
Results for sample thermal conductivity are shown figure 13(a) and 13(b). Only IP are plotted, because TT are quite identical.



13(a)

As expected it features two trends. First, when increasing the domain size, the error decreases (figure 13(b)) : samples are big enough to all be representative and therefore the error on the mean become small. On the contrary, for small domains, the error on mean is large and even the mean value seems very far from convergence value which are supposed to be the most accurate values figure 13(a). It may be because PuMA struggle to converge, or because 100 repetitions are not enough, but from $300\mu m$ is converged (final value is within the 95% confidence interval). Second you may notice the influence of r_{vx} . When it increases, the converged value also increases, until a value around $0.782W/m.K$. Not shown here are also computation with $r_{vx} = 15$, where the converged value is almost indiscernible from one at $r_{vx} = 10$.

Before making our final choice on (r_{vx}, D_{phy}) , note that what we intend to do is not to make a perfect predictor of the QoI : a small error because the two numerical parameters are not as high as possible is not critical. This is all the more right that we are going to make estimation of



13(b)

Figure 13: Convergence of QoI as functions of r_{vx} and D_{phy} . (a) shows mean of repetitions and (b) precises the error on the mean. Statistical quantities are computed over 100 repetitions, each at nominal parameters $p = 0.85$, $\lambda_f = 12.0\text{W/m.K}$, $\lambda_g = 0.04\text{W/m.K}$. There is no uncertainty on r_{vx} but $l_{vx} = 80r_{vx} \pm 50\%$

QoI with like 100-200 repetitions. Therefore we will be satisfied if for 100 repetitions the mean value confidence interval and confidence interval for $(r_{vx}, D_{phy}) = (10, 500\mu\text{m})$ (our reference value) mostly intersect.

Everything considered, these are the numerical parameters we will use for now :

- $r_{vx} = 8vx$
- $D_{phy} = 300\mu\text{m}$

Furthermore, note that we choose 30° as angle variation parameter for sampling fiber's orientation, a value advised by VKI.

Finally, here are the physical values for radius and length :

- $R_{phy} = 5 \pm 0.625\mu\text{m}$
- $L_{phy} = 800 \pm 500\mu\text{m}$

Here we account for both great stochasticity of fiber length as well as possibility of fibers clusters (but obviously this is not equivalent).

Tomography

Before heading towards metamodeling, we can conclude on tomography usage of PuMA. Indeed, figure 13 gives some insight : if you are looking for a certain accuracy, then you should find r_{vx} which fits this accuracy and turn it into tomography accuracy (which is equivalent to

(p_{vx}^{phy}). If you are looking for a certain precision on the mean of your QoI, then you should choose D_{phy} so that the variability of the mean is small enough.

Besides, it can be used for designing a material. Let's say you need a certain value of mean for your QoI, and you want it to have a maximum percentage of variation. Then you can know until which sample size you need to be accurate in geometrical parameters (such as porosity) in order to meet this criterion : if you respect geometrical parameters for every subdomain of size d then the variation is guaranteed to be lower than a certain threshold.

5.2 UQ numerical campaign

The previous study on PuMA behavior regarding stochasticity has brought some information over how to perform tomography, but also how to configure PuMA for producing more quantitative studies on thermal conductivities. This is what follows is dedicated to.

We remind the objective : build a surrogate model of thermal conductivity next to parameters' nominal value, and evaluate the impact of uncertainty and stochasticity on the variability of the output.

The results are concentrated on two axis. First, we will make the analysis for helium as medium gas, for several temperatures and see the impact of the temperature on the Sobol indices (fibers and gas conductivities should vary). Then we will fix the temperature and switch to nitrogen and argon, thus changing the gas conductivity, and see the effects of this choice.

Values come from [64] and are related to TC2 type of fibers pre-treated at $2500K$. In Section 2, we said we were interested in Calcarb[®] CBCF 18-2000 pre-treated at $2000K$, but TC2 is the same type (rayon), and $2000K$ is relatively close enough to consider their respective performances to be equivalent, at least for what we intend to demonstrate. We summarize the values below :

- Temperatures are $T = 855, 1380, 1875K$
- Gases are helium, argon and nitrogen
- Once temperature is chosen, the gas conductivity is supposed to be known without uncertainty. Its value is computed at ambient pressure with Mutation⁺⁺ [65]. At writing time, the software does not consider pressure as a parameter anyway, as long as medium is not rarefied.
- Nominal values and uncertainty of fiber conductivity come from [64]. We use the simple formula :

$$\lambda_f = \rho c_p \kappa$$

where ρ is the local density, c_p is the local heat capacity and κ is the local thermal diffusivity.

- Nominal porosity p is the mean porosity of Calcarb[®] CBCF 18-2000 : $p = 0.89$. Uncertainty is $2\sigma_p = 0.01$.

Overall, for a given temperature and gas we consider conductivity as a stochastic function of p and λ_f , which are known with uncertainty. Therefore we can apply the numerical methods that have been introduced Sections 3 and 4.

For each experiment, boundaries of the domain of interest (from which the DoE is generated and on which prediction will be made) is defined based on nominal values and uncertainties of p and λ_f . $N_{max} = 30$ training points are then generated with LHS. As discussed above, making an exact number of simulation (hundreds) for each training point is quite tricky. Therefore we have

run thousands of simulations and then apply the unequal allocation scheme with 4 allocation steps. It has resulted in a total budget of $C_{max} = 3800$, distributed among the 30 training points. Note that we use UQLab project [66] on MATLAB for coding and solving.

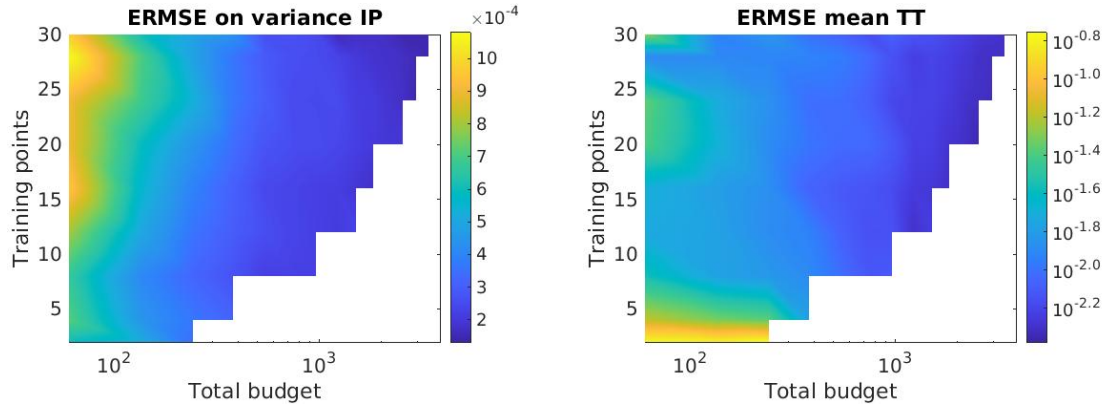
Before talking about predictions, note that Assumption 1 has been checked : when making simulations for a given training points, distributions of λ (TT and IP) converge to Gaussian distributions.

5.3 UQ-based prediction

5.3.1 Convergence of metamodel

Before analysing results, we need to check if our methods have indeed reached convergence. For this, we have used ERMSE on mean and variance metamodels in order to evaluate the errors of the method when increasing the number of training points taken into account (randomly chosen among the 30) and increasing the allocated budget. However, ERMSE demands reference values to compare predictions to. Since not knowing the exact function is precisely the interest of metamodeling, we have chosen to take the empirical mean and empirical variance at all 30 training points as reference, and compare it to the prediction given by the submodels.

Figure 14 shows the result of convergence. As expected, increasing C and N globally decreases the error. However note that increasing N with fixed C does not necessarily lead to better performance : because the same budget is allocated on more points, *a priori* estimation of QoI are less accurate. This is especially true for variance figure 14(a) for low budgets. This effect disappears at higher budget because it is compensated by the hyperparameters optimization process. Furthermore one observes the convergence is quite fast when increasing N when C is larger than $10e2$: the function varies quite slowly and stochasticity is more responsible for inaccuracy than lack of training points.



(a) Convergence of In Plane variance ERMSE in $[W/m.K]^2$

(b) Convergence of Through Thickness mean ERMSE in $[W/m.K]$

Figure 14: Convergence of ERMSEs when increasing C and N . The reference is empirical QoI at all training points.

Finally, there is one thing one may want to check is that our model is not under-fitting or over-fitting. This has been verified : hyperparameters does not show irregular behaviors and MSEs of both mean and variance metamodels look has expected.

5.3.2 Mean prediction

We can now be quite confident in our results. First step if computing for these sets of parameters with uncertainty the mean conductivity of a sample. Let's begin by the impact of temperature in the case of helium. Both IP and TT conductivities are plotted figure 15.

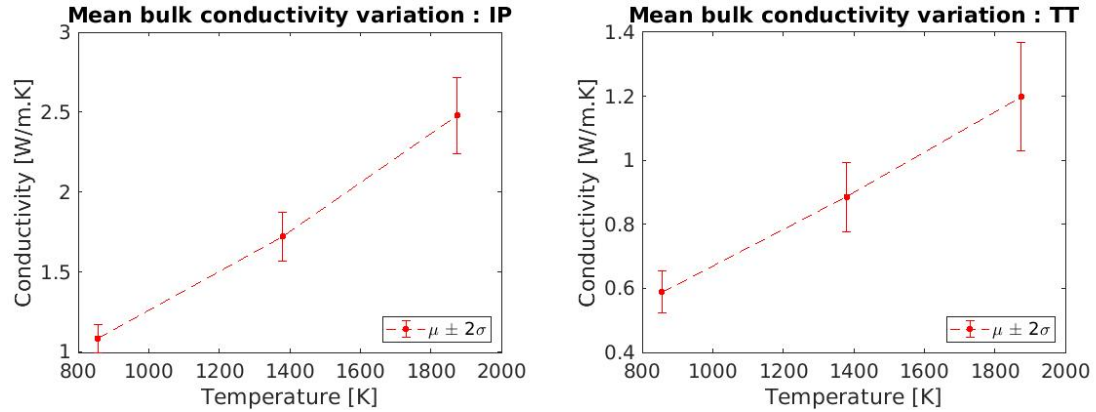


Figure 15: Influence of temperature on total sample conductivity in case of helium. Plotted are means and total variability errorbars.

For helium, increasing temperature leads to a rise of conductivity. This is in fact the case for most gases and fiber types. [64] shows that fiber conductivity is itself an increasing function of temperature for this interval of temperature. This is also the conclusion of Mutation⁺⁺ for gas conductivity.

Finally, IP conductivity is larger than TT conductivity : it is due to the geometry and fibers orientation preferentially in the plane.

Now figure 16 is plotted the influence of the choice of the gas on the total conductivity, for a fixed intermediate temperature of $T = 1380K$. Here again no surprise : for this temperature, $\lambda_{helium} > \lambda_{nitrogen} > \lambda_{argon}$. If interactions between gas and fiber are neglected, this explains the results.

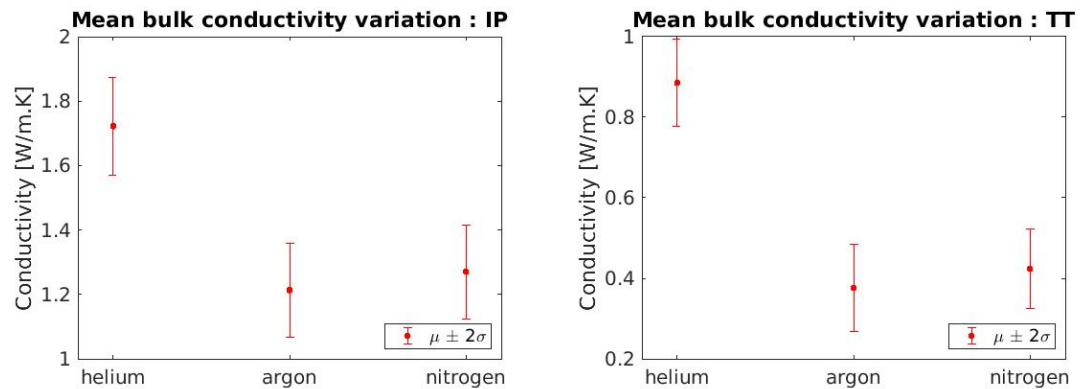


Figure 16: Influence of gas on total sample conductivity for $T = 1380K$. Plotted are means and total variability errorbars.

5.3.3 Sensitivity analysis

We now have some insight into mean response of the material given parameters. This section focuses on the variability of the output, and its decomposition after variable components. We recall we uncertainty on porosity and fiber conductivity as well as model intrinsic stochasticity as sources of variability. As for mean analysis, we will first study variability for several temperature and then for several gases.

Preliminary verification

Before all, we need to meet the requirement for performing variance decomposition as introduced in Section 4. Uncertainties over parameters of interest are taken from literature, independently from each other. Therefore we need to check that model stochasticity is independent from parameters.

We use PuMA artificial domain generation tool. Stochasticity is then decided by how PuMA chooses positions, length and radius of fibers (we choose deterministic QoI solvers among those available). In the very code of PuMA, those are generated one by one independently, with a seed chosen at the beginning of the generation. Therefore we can control stochasticity with the value given to the seed : unless we deliberately choose a link between parameters choice and seed choice when generating random samples (it has been described as a sensitivity analysis method in [43]), there is no dependency between stochasticity and parametric uncertainty and we can use SH decomposition. ξ is then a seed, with initialization based on the current time for example. Therefore, $F(\mathbf{x}, \xi)$ might be a 3D matrix, where each element is the type of material of the associated voxel in the domain or the inherent property of the voxel. We can then use SH decomposition.

We also suggest to verify the convergence of the QoI in addition to the convergence of ERMSEs. Figure 17 shows the sensitivity analysis results from the sub-models of the ERMSE convergence analysis. For now please focus on the convergence of surfaces rather than absolute values that will be studied latter. As well as for ERMSEs, convergence is fast (in both budget

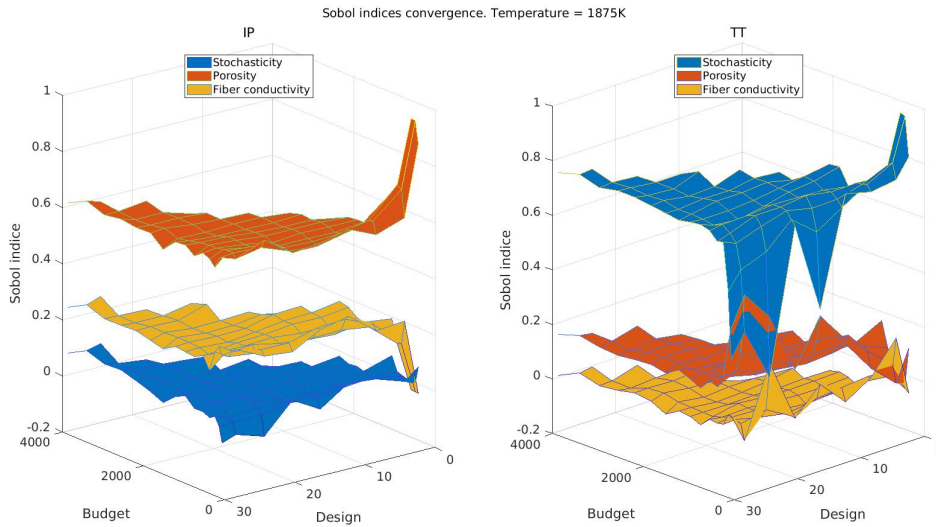


Figure 17: Convergence of Sobol indices when increasing C and N .

and number of training points terms). This mean our metamodeling method is quite strong, and does not need as much simulations for reaching convergence. But because too much accuracy is better than not enough, we use all data available for our analyses.

Temperature influence

Figure 18 shows the results from the sensitivity analysis for helium for the three temperatures.

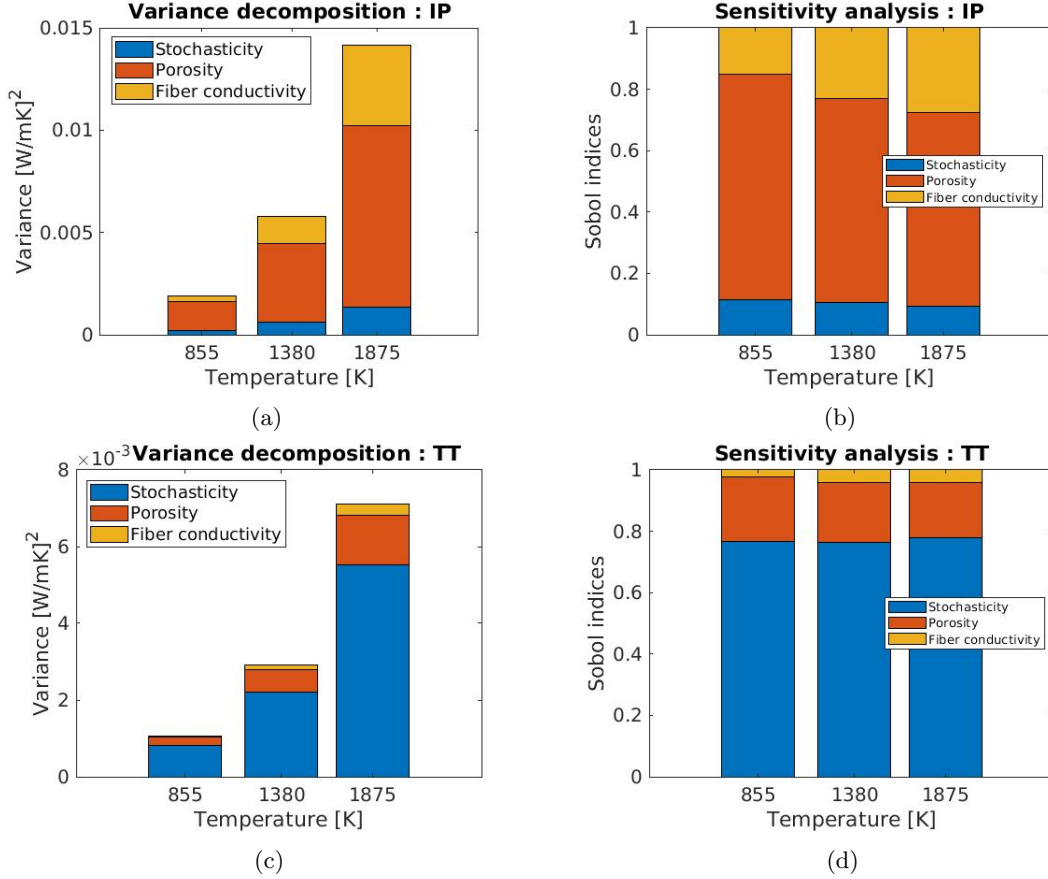


Figure 18: (a) (resp. (c)) show variance decomposition of the total variability In Plane (resp. Through Thickness) for three temperatures. (b) and (d) show associated Sobol decomposition. Gas is helium.

Note that stacked variance for a given a temperature is directly equal to the square root of the associated total errorbar divided by four of figure 15. Finally, note that Sobol indices have been normalized to one. Indeed the sum of the three order one indices were very close to one, and we assumed the difference was because of the Monte Carlo estimation method. Overall it was largely inferior to components indices, this is why we normalized them *a posteriori* for readability.

Several things can be noticed. Here are some general remarks :

- Regarding Sobol indices (figures 18(b) and 18(d)) there is a clear hierarchy between components, different in IP and TT cases.
- Regarding variance magnitude (figures 18(a) and 18(c)). Total variability increases with temperature, as well as all its components. At least two reasons can be given : first, fiber conductivity uncertainty increases with temperature, so that it seems normal its associated component increases. You can notably note fiber conductivity influence increases with

temperature. Second, mean fiber conductivity also increases with the temperature : a change in geometry therefore have more impact one the output. What remains, it is due of more complex mutual relations between parameters and stochasticity, as well as relative increase of λ_f with respect to λ_g

Note that stochasticity component is much higher in through plane case. This is not only due to the fact that they are two differents properties (remember λ_{TT} is smaller than λ_{IP} from section 5.3.2), but also because λ_{TT} is the mean of two quantities that are supposed to follow approximately the same type of law as λ_{IP} : therefore variance is lower in plane.

Gas influence

Figure 19 shows the results from the sensitivity analysis for the three gases for the intermediate temperature.

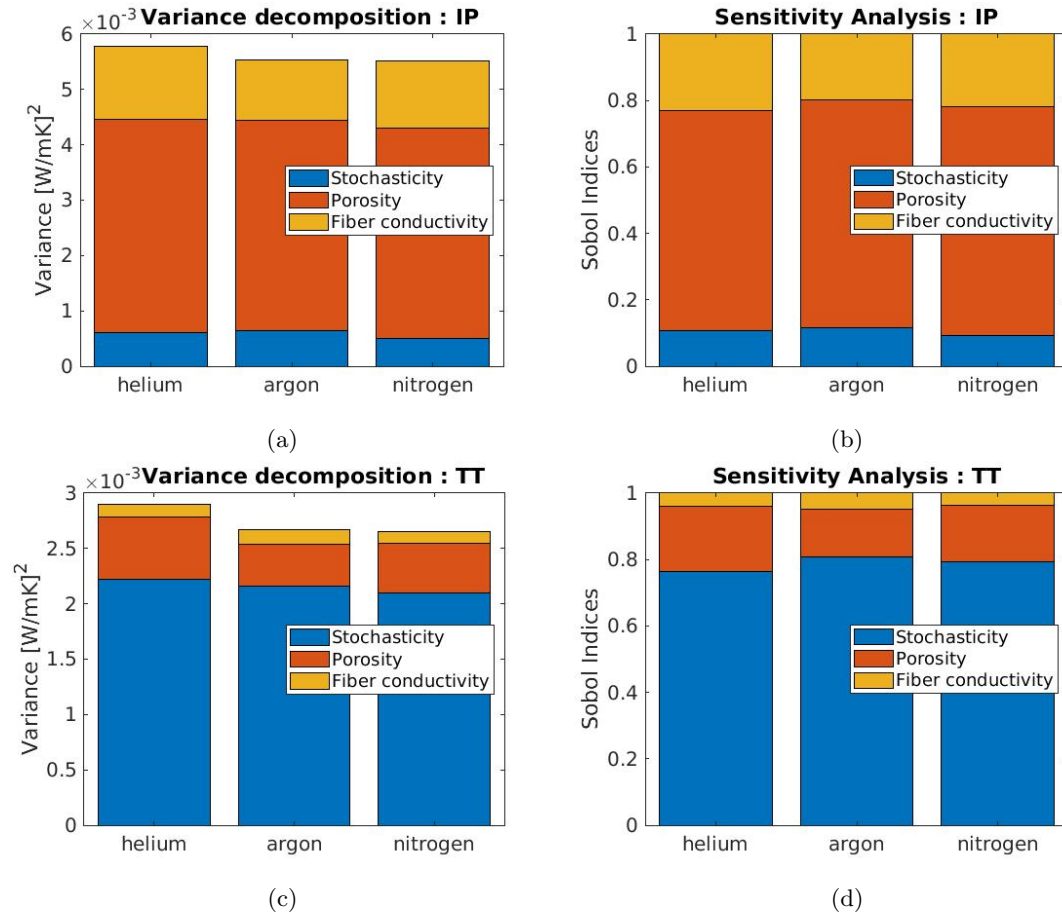


Figure 19: (a) (resp. (c)) show variance decomposition of the total variability In Plane (resp. Through Thickness) for three gases. (b) and (d) show associated Sobol decomposition. Temperature is $T = 1380K$.

Between each gas, main difference is thus the nominal gas conductivity. We have seen section 5.3.2 it does change the mean conductivity of the sample. However, the impact on variance

is much slighter . Figures 19 (a) and (c) seems to show that it decreases total variability but figures 19 (b) and (d) does not show great difference between gases regarding repartition of indices. Furthermore it is difficult to estimate the error made by the Monte Carlo estimation method of Sobol indices and draw a particular trend.

6 Conclusions and perspectives

6.1 Main achievements

The contributions of this paper are summarized in this section :

- From the methodological point of view, we have introduced a novel methodology for surrogate which shows to be effective at least for our mechanical model. On the test case, it has performed as good as other metamodels tested, and has proven to be capable of predicting both mean and variance of stochastic simulations.
- From an informatic point of view, we have automatized PuMA to allow parallel computations within a UQ framework carrying out the first massive simulation campaign with PuMA as far as we know.
- From the physical point of view, we have performed analyses of several behaviors :
 - Tomography analysis campaign can be prepared thanks to PuMA, taking into account the information one has on the material,
 - Building a surrogate allows to predict thermal conductivities for new parameters, but also to predict the influence of all sources of variability. On the contrary, a set of sensitivity analysis could lead to backward propagation and help for designing new material,
 - Given the current knowledge on fiber and gas properties, thermal conductivities variability of studied porous material main component is porosity for in plane conductivity and inherent stochasticity for through thickness conductivity,
 - Changing gas does change mean conductivity as expected but very little the total variance. This means when external conditions other than temperature change, gas influence should be precised only for variability issues,
 - As expected, when increasing temperature, the thermal conductivity increases. But it is also the case for its variance. Furthermore, the higher temperature is, the higher Sobol index of fiber conductivity is,
 - Sobol indices of order 2 or higher appear to be negligible compared to Sobol indices of order 1.

6.2 Perspectives

- From an algorithmic point of view, we could enhance the computational cost. Simulations campaign are quite costly, but because simulations outcomes are quite unpredictable we had to run massively.
- From a physical point of view, applying systematically the proposed framework to analyse PuMA results seems to have proven to be adapted. This could be extended to other studies on PuMA, either for new kinds of parameters or properties.
- The UQ problem could be adapted to tomography images, even if less data are available. For instance, rather than decreasing the domain with fixed geometrical parameters, one could zoom in tomography images and isolate sub domains with variable properties.
- A more recent version of PuMA has been released [67]. It can notably generated curved fiber with variable cross sections, which would make simulations closer to reality.

Appendices

A Convergence of unbiased sample variance

Let Z_1, \dots, Z_n *iid.* be n random variables of law $\mathcal{N}(\mu, \sigma^2)$. Let's note

$$\mathcal{S}^2 = \frac{1}{n-1} \sum_{i=1}^n (Z_i - \bar{Z})^2$$

the empirical variance of the sampling, where \bar{Z} is the empirical mean of the sampling $\bar{Z} = \frac{\sum_{i=1}^n Z_i}{n}$. Before all one remembers $\bar{Z} \sim \mathcal{N}(\mu, \frac{\sigma^2}{n})$ and μ can be taken equal to zero (turning $Z_i - \bar{Z}$ into $(Z_i - \mu) - (\bar{Z} - \mu)$).

First, let's compute

$$\begin{aligned} \mathbb{E}[Z_i \bar{Z}] &= \mathbb{E} \left[\frac{Z_i^2}{n} + \frac{1}{n} \sum_{j \neq i} Z_i Z_j \right] = \frac{\sigma^2}{n} \\ \mathbb{E}[(Z_i - \bar{Z})^2] &= \sigma^2 + \frac{\sigma^2}{n} - 2\mathbb{E}[Z_i \bar{Z}] = \frac{n-1}{n} \sigma^2 \end{aligned}$$

Therefore one obtains

$$\begin{aligned} \mathbb{E}[\mathcal{S}^2] &= \frac{n}{n-1} \mathbb{E}[(Z_1 - \bar{Z})^2] = \sigma^2 \\ &= \sigma^2 \end{aligned}$$

Then let's notice : $\mathcal{S}^2 = \frac{\sum_{i=1}^n Z_i^2}{n-1} - \frac{n}{n-1} \bar{Z}^2$

One needs the following equalities, that can be computed for example from integration of the density of probability :

$$\mathbb{E}[Z_i^4] = 3\sigma^4 \quad \text{Var}(Z_i^2) = 2\sigma^4 \quad \text{Var}(\bar{Z}^2) = 2\frac{\sigma^4}{n^2}$$

Then :

$$\begin{aligned} \text{Cov} \left(\frac{\sum_{i=1}^n Z_i^2}{n-1}, \frac{n}{n-1} \bar{Z}^2 \right) &= \frac{n}{(n-1)^2} \mathbb{E} \left[\left(\sum_{i=1}^n Z_i^2 - n\sigma^2 \right) \left(\bar{Z}^2 - \frac{\sigma^2}{n} \right) \right] \\ &= \frac{n^2}{(n-1)^2} \mathbb{E}[Z_1^2 \bar{Z}^2] - \frac{n}{(n-1)^2} \sigma^4 \\ \mathbb{E}[Z_1^2 \bar{Z}^2] &= \mathbb{E} \left[\frac{Z_1^4 + 2Z_1^3 \sum_{i=2}^n Z_i + Z_1^2 (\sum_{i=2}^n Z_i)^2}{n^2} \right] \\ &= \frac{3\sigma^4}{n^2} + \frac{\sigma^2}{n^2} \mathbb{E} \left[\sum_{i=2}^n Z_i^2 + 2 \sum_{1 < i \neq j} Z_i Z_j \right] \\ &= \frac{3}{n^2} \sigma^4 + \frac{n-1}{n^2} \sigma^4 \end{aligned}$$

Finally :

$$\begin{aligned}\text{Var}(\mathcal{S}^2) &= \frac{n}{(n-1)^2} 2\sigma^4 + \frac{n^2}{(n-1)^2} \frac{2}{n^2} \sigma^4 \\ &\quad - 2 \frac{n^2}{(n-1)^2} \left(\frac{3}{n^2} \sigma^4 + \frac{n-1}{n^2} \sigma^4 \right) + \frac{2n}{(n-1)^2} \sigma^4 \\ &= \frac{2}{n-1} \sigma^4\end{aligned}$$

B Allocation distribution

Let $\Sigma(\mathbf{n}) = \Sigma_M + \Sigma_\varepsilon(\mathbf{n})$ where $\Sigma_\varepsilon(\mathbf{n}) = (\frac{V_1}{n_1}, \dots, \frac{V_1}{n_1})$. If we assume $n_i \gg 1$, then $\Sigma(\mathbf{n}) \approx \Sigma_M$, with $\Sigma_{M,ij} = k(\mathbf{x}_i, \mathbf{x}_j)$.

We assume the model has converged : hyperparameter is a characteristic lengthscale. Because training points are space filling and k is isotropic, \mathbf{W}_{ij} and $\Sigma_{M,st}$ do not depend on (i, j, s, t) but on the relative positions of $\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_s$ and \mathbf{x}_t (in reality, it does for \mathbf{W} when looking at training point at the very edge of the domain, but we are only looking for an approximation of \mathbf{n}).

Finally when looking at diagonal terms $[\Sigma_M^{-1} \mathbf{W} \Sigma_M^{-1}]_{ii}$, the subscript ii is not important. They are linear combinations of matrices elements, so you can switch from ii to jj by an appropriate permutation of all indices.

References

- [1] K. Raed and U. Gross. Modeling of influence of gas atmosphere and pore-size distribution on the effective thermal conductivity of knudsen and non-knudsen porous materials. *International Journal of Thermophysics*, 30:1343–1356, 08 2009.
- [2] J.C. Ferguson and al. Puma: the porous microstructure analysis software. *SoftwareX*, 7:81–87, 2018.
- [3] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. the MIT Press, 2006.
- [4] Maeder, G. Le choix des matériaux dans l’industrie automobile. *Matériaux & Techniques*, 86:39–47, 1998.
- [5] A. Timmis and al. Environmental impact assessment of aviation emission reduction through the implementation of composite materials. *The International Journal of Life Cycle Assessment*, 20:233–243, 02 2014.
- [6] E. Kablov and al. Glass and ceramics based high-temperature composite materials for use in aviation technology. *Glass and Ceramics*, 69, 07 2012.
- [7] I. Kondyurina and al. Polymerisation of composite materials in space environment for development of a moon base. *Advances in Space Research*, 37(1):109–115, 2006. The Moon and Near-Earth Objects.
- [8] M.S. Scholz and al. The use of composite materials in modern orthopaedic medicine and prosthetic devices: A review. *Composites Science and Technology*, 71(16):1791–1803, 2011.
- [9] J.L. Wang. Application of composite materials on sports equipments. In *Mechanical Engineering and Green Manufacturing II*, volume 155 of *Applied Mechanics and Materials*, pages 903–906. Trans Tech Publications Ltd, 3 2012.
- [10] M. Thapa, S. Mulani, and R. Walters. Stochastic multi-scale modeling of carbon fiber reinforced composites with polynomial chaos. *Composite Structures*, 04 2019.
- [11] L. Collins and S. Roberts. Mesoscale simulation of woven composite design decisions, 2021.
- [12] F. Panerai, J. Ferguson, J. Lachaud, A. Martin, M. Gasch, and N. Mansour. Microtomography based analysis of thermal conductivity, diffusivity and oxidation behavior of rigid and flexible fibrous insulators. *International Journal of Heat and Mass Transfer*, 108:801–811, 2017.
- [13] F. Semeraro, J. Ferguson, F. Panerai, R. King, and N. Mansour. Anisotropic analysis of fibrous and woven materials part 1: Estimation of local orientation. *Computational Materials Science*, 178:109631, 2020.
- [14] F.-Herrador, A. Turchi, K.M. Van Geem, J. Blondeau, and T Magin. Determination of heat capacity of carbon composites with application to carbon/phenolic ablators up to high temperatures. *Aerospace Science and Technology*, 108:106375, 2021.
- [15] E. Laporte and Le P. Tallec. *Numerical Methods in Sensitivity Analysis and Shape Optimization*. 01 2003.

-
- [16] R. Smith. Uncertainty quantification - theory, implementation, and applications. In *Computational science and engineering*, 2014.
- [17] Mickael Rivier. *Low-cost methods for constrained multi-objective optimization under uncertainty*. Theses, Institut Polytechnique de Paris, September 2020.
- [18] S. Mahadevan and B. Liang. Error and uncertainty quantification and sensitivity analysis in mechanics computational models. *International Journal for Uncertainty Quantification*, 1:147–161, 01 2011.
- [19] M. Rivier, J. Lachaud, and P.M. Congedo. Ablative thermal protection system under uncertainties including pyrolysis gas composition. *Aerospace Science and Technology*, 84:1059–1069, 2019.
- [20] W. Xiaoyong and al. Uncertainty analysis of laminar and turbulent aeroheating predictions for mars entry. *International Journal of Heat and Mass Transfer*, 112:533–543, 2017.
- [21] A. Turchi, P.M. Congedo, B. Helber, and T.E. Magin. Thermochemical ablation modeling forward uncertainty analysis—part ii: Application to plasma wind-tunnel testing. *International Journal of Thermal Sciences*, 118:510–517, 2017.
- [22] J. Ferguson and al. Microscale artificial weave generation capabilities for thermal protection system material modelling, 2019.
- [23] J.B. Rico Orero. Computational study on the effect of the microstructure on macroscopic properties for carbon fiber felts. Master’s thesis, 2020.
- [24] A. Wiegmann and A. Zemitis. Ej-heat: A fast explicit jump harmonic averaging solver for the effective heat conductivity of composite materials. Technical Report 94, Fraunhofer (ITWM), 2006.
- [25] G.L. Vignoles and al. Ablative and catalytic behavior of carbon-based porous thermal protection materials in nitrogen plasmas. *Carbon*, 134:376–390, 2018.
- [26] A. Cullen and H. Frey. *Probabilistic Techniques in Exposure Assessment. A Handbook for Dealing with Variability and Uncertainty in Models and Inputs*. Plenum Press, New York and London, 1999.
- [27] J.P.C Kleijnen. Statistical validation of simulation models. *European Journal of Operational Research*, 87(1):21–34, 1995.
- [28] J.P.C Kleijnen and R.C. Sargent. A methodology for fitting and validating metamodels in simulation. *European Journal of Operational Research*, 120(1):14–29, 2000.
- [29] A. Phillips, D. Janies, and W.C. Wheeler. Multiple sequence alignment in phylogenetic analysis. *Molecular phylogenetics and evolution*, 16:317–30, 10 2000.
- [30] A. Saltelli, K. Chan, and E. M. Scott. *Sensitivity Analysis*. John Wiley & Sons, Ltd.: West Sussex, England., 2000.
- [31] Roger J. Breeding Jon C. Helton. Calculation of reactor accident safety goals. *Reliability Engineering & System Safety*, 39:129–158, 1993.
- [32] J. Arora and E. Haug. Methods of design sensitivity analysis in structural optimization. *AIAA Journal*, 17(9):970–974, 1979.

- [33] E.J. Haug, K.K. Choi, and V. Komkov. *Design sensitivity analysis of structural systems*. Academic Press, 1986.
- [34] M. Navarro Jimenez, O.P. Le Maitre, and O.M. Knio. Global Sensitivity Analysis in Stochastic Simulators of Uncertain Reaction Networks. *The Journal of Chemical Physics*, 145, 2017.
- [35] M. Rathinama, P.W. Sheppard, and M. Khammash. Efficient computation of parameter sensitivities of discrete stochastic chemical reaction networks. *The Journal of Chemical Physics*, 132, 2010.
- [36] A. Saltelli and al. *Sensitivity analysis in practice: a guide to assessing scientific models*. John Wiley & Sons, 2004.
- [37] A. Alexanderian and al. Preconditioned bayesian regression for stochastic chemical kinetics. *Journal of Scientific Computing*, 58:592–626, 2013.
- [38] O.P. Le Maitre and O.M. Knio. Pc analysis of stochastic differential equations driven by wiener noise. *Reliability Engineering & System Safety*, 135:107–124, 2015.
- [39] I.M. Sobol. On sensitivity estimation for nonlinear mathematical models. *Matematicheskoe Modelirovanie*, 2(1):112–118, 1990.
- [40] B. Ankenman, B.L. Nelson, and J. Staum. Stochastic Kriging for Simulation Metamodeling. *Operations Research*, 58(2):371–382, 2010.
- [41] X. Wang and Z. Peng. Method of moments for estimating uncertainty distribution. *Journal of Uncertainty Analysis and Applications*, 2, 01 2010.
- [42] B. Sudret, S. Marelli, and J. Wiart. Surrogate models for uncertainty quantification: An overview. In *2017 11th European Conference on Antennas and Propagation (EUCAP)*, pages 793–797, 2017.
- [43] Soumaya Azzi. *Surrogate modeling of stochastic simulators*. PhD thesis, 2020.
- [44] N. Fajraoui, S. Marelli, and B. Sudret. Sequential design of experiment for sparse polynomial chaos expansions. *SIAM/ASA Journal on Uncertainty Quantification*, 5:1061–1085, 01 2017.
- [45] R. Higdon. *Generalized Additive Models*. Springer New York, New York, NY, 2013.
- [46] X. Zhu and B. Sudret. Global sensitivity analysis for stochastic simulators based on generalized lambda surrogate models. *Reliability Engineering & System Safety*, 214:107815, 2021.
- [47] M. Plumlee and R. Tuo. Building accurate emulators for stochastic simulations via quantile kriging. *Technometrics*, 56(4):466–473, 2014.
- [48] R. Rigby and D. Stasinopoulos. Construction of reference centiles using mean and dispersion additive models. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 49:41–50, 04 2000.
- [49] Brian J. Reich, Eric Kalendra, Curtis B. Storlie, Howard D. Bondell, and Montserrat Fuentes. Variable selection for high dimensional bayesian density estimation: application to human exposure simulation. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 61(1):47–66, 2012.

- [50] S. Freitag. *Artificial Neural Networks in Structural Mechanics*, volume 12, pages 1–26. 09 2015.
- [51] R. Rojas. *Neural Networks: A Systematic Introduction*. Springer-Verlag, Berlin, Heidelberg, 1996.
- [52] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, 14:199–222, 2004.
- [53] N. Razaaly and P.M Congedo. Novel algorithm using active metamodel learning and importance sampling: Application to multiple failure regions of low probability. *Journal of Computational Physics*, 368, 04 2018.
- [54] J. Sacks, W.J. Welch, T.J. Mitchell, and H.P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423, 1989.
- [55] J. Oakley and A. O’Hagan. Probabilistic sensitivity analysis of complex models: a bayesian approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(3):751–769, 2004.
- [56] C. Lataniotis, D. Wicaksono, S. Marelli, and B. Sudret. UQLab user manual – Kriging (Gaussian process modeling), 2021. Report # UQLab-V1.4-105.
- [57] V. Dubourg. *Adaptive surrogate models for reliability analysis and reliability-based design optimization*. PhD thesis, Université Blaise Pascal, Clermont-Ferrand, France, 2011.
- [58] G. Vining and R. Myers. Combining taguchi and response surface philosophies: A dual response approach. *Journal of Quality Technology*, 22(1):38–45, 1990.
- [59] I. Zabalza, J. P. Dejean, and D. Collombier. Prediction and density estimation of a horizontal well productivity index using generalized linear. European Association of Geoscientists & Engineers, 1998.
- [60] A. Marrel and B. Iooss. Advanced methodology for uncertainty propagation in computer experiments with large number of inputs : application to accidental scenario in a pressurized water reactor. In *European Safety and Reliability Conference (ESREL - 2018)*, Trondheim, Norway, June 2018.
- [61] B. Iooss and M. Ribatet and A. Marrel. Global Sensitivity Analysis of Stochastic Computer Models with joint metamodels, January 2009. Unpublished.
- [62] W. Wang and X. Chen. The effects of estimation of heteroscedasticity on stochastic kriging. 2016.
- [63] M. Binois, R. Gramacy, and . Ludkovski. Practical heteroscedastic gaussian process modeling for large simulation experiments. *Journal of Computational and Graphical Statistics*, 27(4):808–821, 2018.
- [64] C. Pradère, J.C Batsale, J.M Goyhénèche, R. Pailler, and S. Dilhaire. Thermal properties of carbon fibers at very high temperature. *Carbon*, 47(3):737–743, March 2009.
- [65] J. Scoggins, V. Leroy, G. Bellas-Chatzigeorgis, B. Dias, and T. Magin. Mutation++: Multicomponent thermodynamic and transport properties for ionized gases in c++. *SoftwareX*, 12:100575, 2020.

- [66] S. Marelli and B.Sudret. Uqlab: A framework for uncertainty quantification in matlab. *The 2nd International Conference on Vulnerability and Risk Analysis and Management*, pages 2554–2563, 2014.
- [67] JFerguson, F. Semeraro, J. Thornton, F. Panerai, A. Borner, and N. Mansour. Update 3.0 to “puma: The porous microstructure analysis software”, (pii:s2352711018300281). *SoftwareX*, 15:100775, 2021.



**RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE**

1 rue Honoré d'Estienne d'Orves
Bâtiment Alan Turing
Campus de l'École Polytechnique
91120 Palaiseau

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399