arXiv:2202.01503v1 [cs.CE] 3 Feb 2022

Global sensitivity analysis based on Gaussian-process metamodelling for complex biomechanical problems

Barbara Wirthl[1,*], Sebastian Brandstaeter[1,2], Jonas Nitzler[1,3], Bernhard A. Schrefler[4,5], Wolfgang A. Wall[1]

1 Institute for Computational Mechanics, Technical University of Munich, Garching b. München, Germany
2 Institute of Continuum and Materials Mechanics, Hamburg University of Technology, Hamburg, Germany
3 Professorship for Data-Driven Materials Modeling, Technical University of Munich, Garching b. München, Germany
4 Department of Civil, Environmental and Architectural Engineering, University of Padua, Padua, Italy
5 Institute of Advanced Study, Technical University of Munich, Garching b. München, Germany

* B. Wirthl, Email: barbara.wirthl@tum.de

Abstract

Biomechanical models often need to describe very complex systems, organs or diseases, and hence also include a large number of parameters. One of the attractive features of physics-based models is that in those models (most) parameters have a clear physical meaning. Nevertheless, the determination of these parameters is often very elaborate and costly and shows a large scatter within the population. Hence, it is essential to identify the most important parameter (worth the effort) for a particular problem at hand. In order to distinguish parameters which have a significant influence on a specific model output from non-influential parameters, we use sensitivity analysis, in particular the Sobol method as a global variance-based method. However, the Sobol method requires a large number of model evaluations, which is prohibitive for computationally expensive models. We therefore employ Gaussian processes as a metamodel for the underlying full model. Metamodelling introduces further uncertainty, which we also quantify. We demonstrate the approach by applying it to two different problems: nanoparticle-mediated drug delivery in a complex, multiphase tumour-growth model, and arterial growth and remodelling. Even relatively small numbers of evaluations of the full model suffice to identify the influential parameters in both cases and to separate them from non-influential parameters. The approach also allows the quantification of higher-order interaction effects. We thus show that a variance-based global sensitivity analysis is feasible for complex, computationally expensive biomechanical models. Different aspects of sensitivity analysis are covered including a transparent declaration of the uncertainties involved in the estimation process. Such a global sensitivity analysis not only helps to massively reduce costs for experimental determination of parameters but is also highly beneficial for inverse analysis of such complex models.

Key nomenclature

Dimensions and corresponding indices

N_{GP}	Number of metamodel realisations, index k
B	Number of bootstrap samples, index b
D	Number of uncertain input parameters, index i
M	Number of Monte–Carlo samples, index m
N	Number of training samples

Random quantities

$\mathcal{X}$	Training samples
$\boldsymbol{x} \in \mathbb{R}^D$	Deterministic vector of input parameters
$\boldsymbol{X}$	Vector of D uncertain input parameters
X_i	Uncertain input parameter (RV)
$y \in \mathbb{R}$	Deterministic scalar output of interest
Y	Uncertain output of interest (RV)

Sobol method related symbols

$\hat{S}^{\diamond}$	Estimate of Sobol index of any order $\diamond$
S^i	First-order Sobol index for the i-th input parameter
S^{ij}	Second-order Sobol index for input parameters i and j
S^{Ti}	Total-order Sobol index for the i-th input parameter

Metamodel related symbols

$m_{\mathsf{GP},N}(\boldsymbol{X})$	Predictive posterior mean of the Gaussian process with optimised hyperparameters trained on N training samples
$f_{\mathrm{GP},N}(\boldsymbol{X})$	Trained Gaussian process metamodel
$k_N(\boldsymbol{X}, \boldsymbol{X}')$	Predictive posterior covariance kernel of the Gaussian process with optimised hyperparameters trained on N training samples

1 Introduction

Over the past few decades, computational biomechanical models have become an essential tool in research. The goal of these models is to allow predictions so as to better understand the underlying biological system or to support decision-making in a medical context, e.g., to choose the most efficient therapy for a specific patient. Nevertheless, the output of such models is inherently subject to uncertainty for various reasons: first, the underlying biological process is stochastic—which is particularly true for oncophysics and cancer treatment models (for example, branching process models for cancer [1, 2], or stochastic models for immunotherapy of cancer [3]). Second, the experimental data used to calibrate models is uncertain [4]. Third, the computational model itself includes sources of uncertainty, including the assumptions made to set up the model, other simplifications, or the input parameters [5].

When analysing the uncertainty of the model output, we distinguish uncertainty analysis from sensitivity analysis [5]: uncertainty analysis quantifies the uncertainty in the model output by propagating input uncertainties, via the model, onto the output [6, p. 262]. Sensitivity analysis, on the other hand, apportions the uncertainty in the model output to different sources of uncertainty in the model input [7, p. 45]. Inputs of interest can generally include not only model parameters but also boundary and initial conditions, assumptions, and constraints [8]. In the context of sensitivity analysis, those inputs of interest are commonly referred to as *factors*. Here, we only consider model parameters as sources of uncertainty and refer to those as *input parameters*.

In this study, the goal is threefold:

- Identify the most influential parameters on which further experimental estimation should focus (called factor prioritisation or ranking).
- Identify parameters with little or no effect, which can thus be set to fixed values within their range (called factor fixing or screening).
- Identify and quantify the interaction between parameters.

This knowledge expedites the efficient design of future computational and experimental studies, while avoiding wasting resources on determining non-influential parameters.

We propose to apply a special type of sensitivity analysis to achieve the goals just described. One way of quantifying the sensitivity of the model output Y on the input parameter X_i is to calculate the partial derivative $\partial Y/\partial X_i$. In practice, this involves choosing a base point $\boldsymbol{X}^*$ and then perturbing one factor at a time while keeping all remaining factors fixed. This results in a local sensitivity measure at the base point $\boldsymbol{X}^*$ which only explores one point of the input space and thus results in a deficient sensitivity analysis [5]. In contrast, the Elementary Effects method (also called Morris method [9]) is not limited to one single point but explores the whole input space. It thereby overcomes the major limitation of local methods, while only requiring a relatively small number of model evaluations. While the Elementary Effects method is a global sensitivity analysis method, it only provides semi-quantitative information and is typically used for factor fixing [7, 6]. However, it cannot detect and quantify interactions between parameters and nonlinearities [10]. In this work, we focus on complex biomechanical problems in which interactions between the different parameters can be expected. We therefore need a global method that can provide more detailed information.

Our method of choice is the Sobol[1]method [11, 12], which is a variance-based global sensitivity analysis method that decomposes the output variance into portions attributed to the input parameters (see Fig. 1). The downside is that it requires many model evaluations, which quickly becomes computationally prohibitive in the case of complex models. We propose to introduce Gaussian processes [13] as a metamodel for the full model to mitigate the problem of computationally expensive model evaluations. Since the use of a metamodel introduces a further source of uncertainty in the sensitivity analysis, we estimate the uncertainty following the approach presented by Le Gratiet *et*

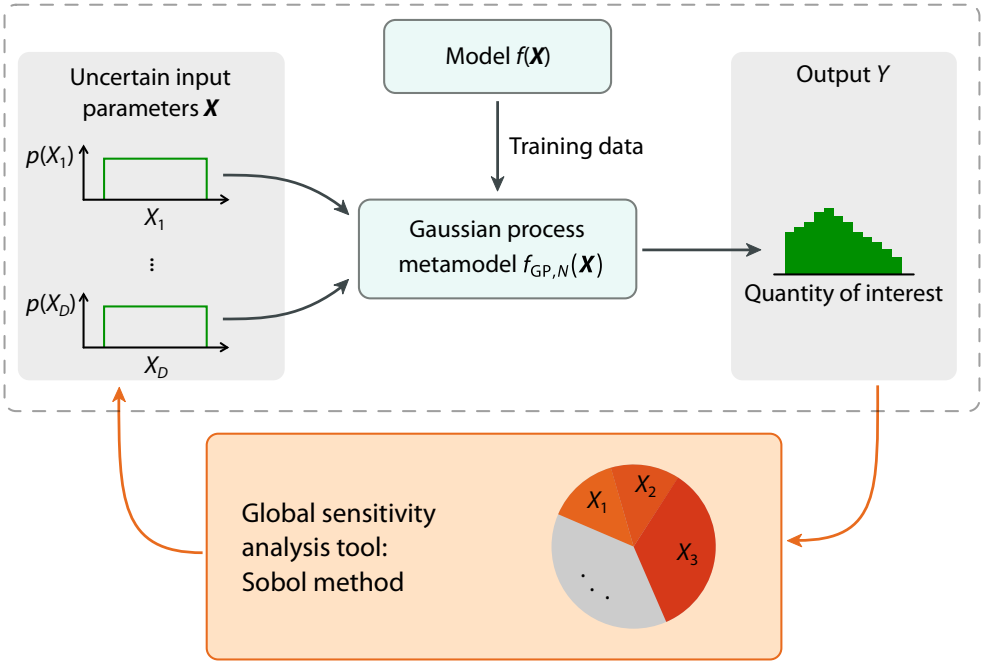


Figure 1. Overview of sensitivity analysis with a Gaussian process metamodel. The Gaussian-process metamodel $f_{\mathrm{GP},N}(\boldsymbol{X})$ is trained on a data set generated by the full model $f(\boldsymbol{X})$. Uncertain input parameters X_i are propagated via the metamodel and result in an uncertain output Y. The Sobol method, as an example of a global sensitivity analysis tool, decomposes the output variance into portions attributed to the input parameters.

al. [14]. After the full biomechanical model is substituted by the metamodel, we can calculate the Sobol indices based on Monte–Carlo integration. The uncertainty related to metamodelling and the uncertainty related to Monte–Carlo integration are analysed both separately and in total.

So far, the approach suggested by [14] has been applied to different computational models: an individual-based model simulation of microbial communities [15], a mathematical model of renal fibrosis [16], climate change simulations [17], and numerical wind-turbine models [18]. Moreover, [19] applied the same idea to the calculation of Elementary Effects of a heart model. In [15, 16, 19, 17], the authors state that they used the method, but no analysis of the associated uncertainties was presented. Only [18] quantified the uncertainty related to the metamodel and the uncertainty related to Monte–Carlo integration separately.

Our goal is to present the complete workflow of estimating Sobol indices based on Gaussian processes as a metamodel including the uncertainties: we demonstrate how to apply the approach suggested by [14] to two different biomechanical models, and we also assess the performance of the method when applied to such complex examples.

The article is structured as follows: we first introduce the reader to the Sobol method and to Gaussian processes in general. The Gaussian-process metamodel is then used to estimate the Sobol indices, including separate estimates for uncertainty related to

[1]The Sobol method was proposed by and is named after Ilya Meyerovich Sobol' (Russian: Илья Меерович Соболь) to whose last name an apostrophe is appended in English to transliterate the Russian letter ь. To avoid confusion with the apostrophe used in English grammar, we omit the apostrophe when referring to the Sobol method.

Monte–Carlo integration and uncertainty related to the metamodel based on [14]. As an example, we subsequently demonstrate how to apply this approach to a multiphase model of nanoparticle-mediated drug delivery in a solid tumour [20, 21, 22] and assess its performance in detail. Finally, we conclude with an outlook on a different complex biomechanical model, i.e., a homogenised constrained mixture model of arterial growth and remodelling [23, 24, 25].

2 Global sensitivity analysis based on Gaussian-process metamodelling

2.1 The Sobol method

In its most general form, a model f is a functional representation of the relevant physical process. The model calculates an output $y = f(\boldsymbol{x}) \in \mathbb{R}$ for any given realisation $\boldsymbol{x} \in \mathbb{R}^D$ of the uncertain input parameters $\boldsymbol{X}$, with D being the number of parameters (see upper left side of Fig. 1). We assume that the random vector $\boldsymbol{X}$ summarises the input parameters $X_1, \ldots, X_D$, which are independent random variables. The probability distribution of $\boldsymbol{X}$ is described by the probability density function $p(\boldsymbol{x})$. In the following, we assume that the input parameters are uniformly distributed, with no loss of generality. Common alternatives are the normal distribution or the log-normal distribution, among many others.

Our goal is to investigate the sensitivity of the model output to the uncertain input parameters $\boldsymbol{X}$. Because of the randomness in the input parameters, the model output Y is also a random variable defined as

$$Y = f(\boldsymbol{X}) = f(X_1, X_2, \ldots, X_D) \tag{1}$$

(see upper right side of Fig. 1). The output distribution can (partially) be described by its first two moments: the expected value $\mathbb{E}[Y]$, and the variance $\sigma_Y^2 = \mathbb{V}[Y]$ (with $\mathbb{E}[\cdot]$ denoting the expectation operator, $\mathbb{V}[\cdot]$ the variance operator, and σ_Y the standard deviation as given in Supplement A.1).

One way of characterising sensitivity is to decompose the variance of the output $\mathbb{V}[Y]$ into portions ascribed to the individual input parameters. A common sensitivity analysis method based on the decomposition of variance is the Sobol method [26]. The core idea is to decompose the output variance $\mathbb{V}[Y]$ as

$$\mathbb{V}[Y] = \sum_{i=1}^{D} V_i + \sum_{i=1}^{D-1} \sum_{j>i}^{D} V_{ij} + \cdots + V_{12\ldots D} \tag{2}$$

with the conditional variances given by

$$\begin{aligned}
V_i &= \mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}[Y|X_i]\right] \\
V_{ij} &= \mathbb{V}_{X_i,X_j}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i,j}}[Y|X_i,X_j]\right] - V_i - V_j \\
V_{ijk} &= \mathbb{V}_{X_i,X_j,X_k}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i,j,k}}[Y|X_i,X_j,X_k]\right] - V_{ij} - V_{ik} - V_{jk} - V_i - V_j - V_k \\
&\ldots
\end{aligned}$$

where $\boldsymbol{X}_{\sim i}$ denotes the vector of all input parameters except X_i. Thus, the output variance $\mathbb{V}[Y]$ is the sum of variances contributed by input parameter X_i, including interactions with other parameters. The idea now is to attribute the total variance to the individual input parameters according to their variance contribution. Note that V_{ij} is the variance contributed by the input parameters X_i and X_j but not expressed

in V_i nor V_j. This is called the interaction of parameters X_i and X_j. Note that this decomposition assumes statistical independence of the input parameters X_i.

Because $\mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[Y|X_i\right]\right]$ is the portion of the output variance ascribed to input parameter X_i, we define the first-order Sobol index S^i as

$$S^i = \frac{\mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[Y|X_i\right]\right]}{\mathbb{V}\left[Y\right]}. \tag{3}$$

The numerator $\mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[Y|X_i\right]\right]$ describes the extent to which the output variance $\mathbb{V}\left[Y\right]$ would be reduced if the parameter X_i was fixed. A parameter X_i with a high first-order index S^i should, hence, have priority when determining parameters based on experiments so as to efficiently reduce the overall uncertainty of the model. The first-order Sobol index is typically used to identify the most influential parameters, which is our first goal [10]. Moreover, a parameter with a high first-order index S^i is more likely to be identifiable from experiments, but can still be non-identifiable [8]: to decide whether a parameter is identifiable or not, an identifiability analysis is required, which complements the sensitivity analysis (see [27] for an overview of identifiability analysis).

The question now arises as to whether $S^i = 0$ is also sufficient to conclude that a parameter has no influence. In fact, this is not the case because the parameter might be involved in interactions with other parameters. A parameter may have no effect if it is varied alone; however, this may be different when it is varied in combination with another parameter, or even with several other parameters. An additional sensitivity measure that includes higher-order interaction effects is needed to identify non-influential parameters. The total-order Sobol index is, therefore, defined as

$$S^{Ti} = \frac{\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[\mathbb{V}_{X_i}\left[Y|\boldsymbol{X}_{\sim i}\right]\right]}{\mathbb{V}\left[Y\right]} = 1 - \frac{\mathbb{V}_{\boldsymbol{X}_{\sim i}}\left[\mathbb{E}_{X_i}\left[Y|\boldsymbol{X}_{\sim i}\right]\right]}{\mathbb{V}\left[Y\right]}. \tag{4}$$

In this case, the numerator $\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[\mathbb{V}_{X_i}\left[Y|\boldsymbol{X}_{\sim i}\right]\right]$ describes the expected output variance that would be left if all parameters but X_i were to be determined [10]. If—and only if—this expected output variance is close to zero, is the parameter X_i non-influential. The total-order index describes the total contribution of the parameter X_i to the output Y: this includes the first-order effect plus any higher-order effects that arise from interactions. The difference $S^{Ti} - S^i$, then, indicates interaction effects between factor X_i and any other factor [6, p. 167]. As mentioned above, the total-order index is particularly helpful in the context of factor fixing: if $S^{Ti} = 0$ (or is in practice sufficiently small), the parameter X_i is non-influential and can be fixed anywhere in its input range without affecting the output variance.

So, the first-order and the total-order Sobol indices serve our first two goals: identify the most influential and the non-influential parameters. To additionally identify interactions between two specific parameters X_i and X_j—which is our third goal—we define the second-order Sobol index as

$$S^{ij} = \frac{\mathbb{V}_{X_i,X_j}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i,j}}\left[Y|X_i,X_j\right]\right]}{\mathbb{V}\left[Y\right]} - S^i - S^j. \tag{5}$$

Finally, dividing Eq. (2) by $\mathbb{V}\left[Y\right]$ and inserting Eqs. (3) and (5) leads to

$$\sum_{i=1}^{D} S^i + \sum_{i=1}^{D-1}\sum_{j>i}^{D} S^{ij} + \ldots + S^{12\ldots D} = 1. \tag{6}$$

All sensitivity indices, thus, sum up to 1; furthermore, they are non-negative. This leads to an interesting implication which is worth noting: even when we have a large

number of parameters, we cannot have a large number of influential parameters. If all D parameters are equally influential, each can only contribute $1/D$ of the variance. If, however, a few parameters have a strong influence on the output Y, the remaining parameters can contribute even less (see [6, Sec. 2.4.3]). As [28] stated: only a small subset of parameters significantly influences one specific system output ("sparsity of factors" principle). It should also be noted that the total-order indices S^{Ti} do not, in general, sum up to 1.

2.2 Numerical approximation of Sobol indices

To estimate the Sobol indices according to Eqs. (3), (4) and (5), we need to compute conditional variances, e.g., $\mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[Y|X_i\right]\right]$, which involves evaluating multidimensional integrals in the space of the input parameters $\mathbb{R}^D$. Numerical integration based on quadrature rules becomes prohibitively expensive as the number of input space dimensions increases. This is why Monte–Carlo integration is employed, the accuracy of which is independent of the number of input space dimensions [29]. For each single integral, Monte–Carlo integration involves evaluating M Monte–Carlo samples: to compute, for example, $\mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[Y|X_i\right]\right]$ one would need M samples to calculate the inner expectation and then repeat this M times to calculate the outer variance, resulting in a computational cost of $\mathcal{O}(M^2)$ [6, p. 164]. Since M usually has to be large[2], this is impractical, especially considering that we would need to evaluate the full model f for each Monte–Carlo sample. To make the estimation of Sobol indices more efficient, [30] rewrote the multidimensional integral so that it can be computed using a single Monte–Carlo loop (summarised in Supplement A.2).

To make the best use of the model evaluations, we employ the efficient algorithms suggested by [32]: to estimate the first and the total-order indices, we generate M samples row-wise concatenated as a matrix $\boldsymbol{A}$ and further M samples concatenated as a matrix $\boldsymbol{B}$. This results in two independent $M \times D$ matrices. We introduce a third matrix $\boldsymbol{A}_{\boldsymbol{B}}^{(i)}$ for each input space dimension $i = 1, \ldots, D$, where all columns are taken from $\boldsymbol{A}$ except the i-th column, which is taken from $\boldsymbol{B}$. One sample $(\boldsymbol{B})_m$ and the corresponding sample $(\boldsymbol{A}_{\boldsymbol{B}}^{(i)})_m$ have X_i in common but differ in all other parameters $\boldsymbol{X}_{\sim i}$.

To calculate the first-order index, we then use the estimator proposed by Saltelli *et al.* [33]

$$\mathbb{V}_{X_i}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[Y|X_i\right]\right] \approx \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{B})_m \left(f(\boldsymbol{A}_{\boldsymbol{B}}^{(i)})_m - f(\boldsymbol{A})_m\right), \tag{7}$$

and for the total-order index, we use the estimator proposed by Jansen [34]:

$$\mathbb{E}_{\boldsymbol{X}_{\sim i}}\left[\mathbb{V}_{X_i}\left[Y|\boldsymbol{X}_{\sim i}\right]\right] \approx \frac{1}{2M}\sum_{m=1}^{M} \left(f(\boldsymbol{A})_m - f(\boldsymbol{A}_{\boldsymbol{B}}^{(i)})_m\right)^2. \tag{8}$$

Alternative forms were presented in [11, 34, 35, 36], among others. The denominator is estimated as

$$\mathbb{V}\left[Y\right] \approx \mathcal{V}[f([\boldsymbol{A}\ \boldsymbol{B}])], \tag{9}$$

where we estimate the variance of the output as the sample variance $\mathcal{V}$ of evaluations of all samples $\boldsymbol{A}$ and $\boldsymbol{B}$. This yields better results, i.e., an estimator with lower variance, compared to $\mathbb{V}\left[Y\right] \approx \mathcal{V}[f(\boldsymbol{A})]$ alone [32].

[2] The error of the Monte–Carlo estimate for the expectation is proportional to $\frac{\sqrt{\mathbb{V}[g]}}{\sqrt{M}}$, with g denoting the integrand. If we assume $\mathbb{V}\left[g\right]$ to be fixed, we have to increase the number of Monte–Carlo samples M, and the error of the estimate thus decreases by $\frac{1}{\sqrt{M}}$ [31].

In addition to the first and total-order indices, we estimate the second-order indices as proposed by Saltelli [32]:

$$\mathbb{V}_{X_i,X_j}\left[\mathbb{E}_{\boldsymbol{X}_{\sim i,j}}\left[Y|X_i,X_j\right]\right] \approx \frac{1}{M}\sum_{m=1}^{M} f(\boldsymbol{B}_{\boldsymbol{A}}^{(i)})_m\, f(\boldsymbol{A}_{\boldsymbol{B}}^{(j)})_m - f(\boldsymbol{A})_m\, f(\boldsymbol{B})_m \tag{10}$$

where the matrix $\boldsymbol{B}_{\boldsymbol{A}}^{(i)}$ is built similar to $\boldsymbol{A}_{\boldsymbol{B}}^{(i)}$. More details on different sensitivity-index estimators can be found in [32, 33], among others.

We hence have to evaluate our model f at all samples of the triplet $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{A}_{\boldsymbol{B}}^{(i)}$ (and additionally $\boldsymbol{B}_{\boldsymbol{A}}^{(i)}$ if second-order indices are included). This means $2M$ simulations are needed for computing $f(\boldsymbol{A})$ and $f(\boldsymbol{B})$ plus $D \cdot M$ simulations needed for computing $f(\boldsymbol{A}_{\boldsymbol{B}}^{(i)})$ for $i = 1, \ldots, D$. The cost of first and total-order indices is, hence, $M(D+2)$ simulations. If second-order indices are included, we need an additional $D \cdot M$ simulations for $f(\boldsymbol{B}_{\boldsymbol{A}}^{(i)})$, resulting in $M(2D+2)$ simulations in total (for more details see Supplement A.3 and the original publication by [32]). In practice, Quasi–Monte–Carlo (QMC) integration is often used to generate the samples because of its superior rate of convergence compared to Monte–Carlo integration [37].

2.3 Gaussian process metamodels

As just described, Monte–Carlo integration to estimate the Sobol indices requires a large number of sample evaluations and is thus computationally prohibitive if the evaluation of the underlying model is expensive. We therefore use a metamodel (also known as surrogate model or emulator) as an approximation of the full model. Classically used metamodels include polynomials, splines, neural networks, polynomial chaos expansion, support vector regression, and Gaussian processes (GPs), among others [38, 39]. Before a metamodel can be used for a sensitivity analysis, for example, it has to be trained to later ensure that it is a good approximation of the full model.

This process consists of three steps, which we first summarise (see Fig. 2) and, then, explain in more detail below:

1. Generate N training samples summarised in $\boldsymbol{\mathcal{X}}$ (resulting in an $N \times D$ matrix).
2. Evaluate the full model at the training samples to obtain the corresponding response: $\boldsymbol{\mathcal{Y}} = f(\boldsymbol{\mathcal{X}})$ (resulting in an $N \times 1$ vector).
3. Form and train the metamodel.

First, we generate N training samples that are summarised in the matrix $\boldsymbol{\mathcal{X}}$ ($N \times D$ matrix). The choice of training samples has to provide a good coverage of the input space to later ensure a good predictive quality of the metamodel. To this end, we use a QMC approach based on Sobol sequences [40] to generate the training samples.

Remark (Sequential design)**.** *A commonly used alternative to a QMC approach is Latin Hypercube Sampling (LHS) [41]. [38] state that optimised LHS is particularly well-suited for metamodel fitting. However, sequential design is also important in the context of metamodelling: if the original number of training samples is not sufficient to achieve a good predictive quality of the metamodel, additional samples can be added while still making use of the original training samples. Since only advanced LHS methods [42, 43] enable sequentially adding new points, while this is straightforward with QMC schemes [44], we use a QMC approach.*

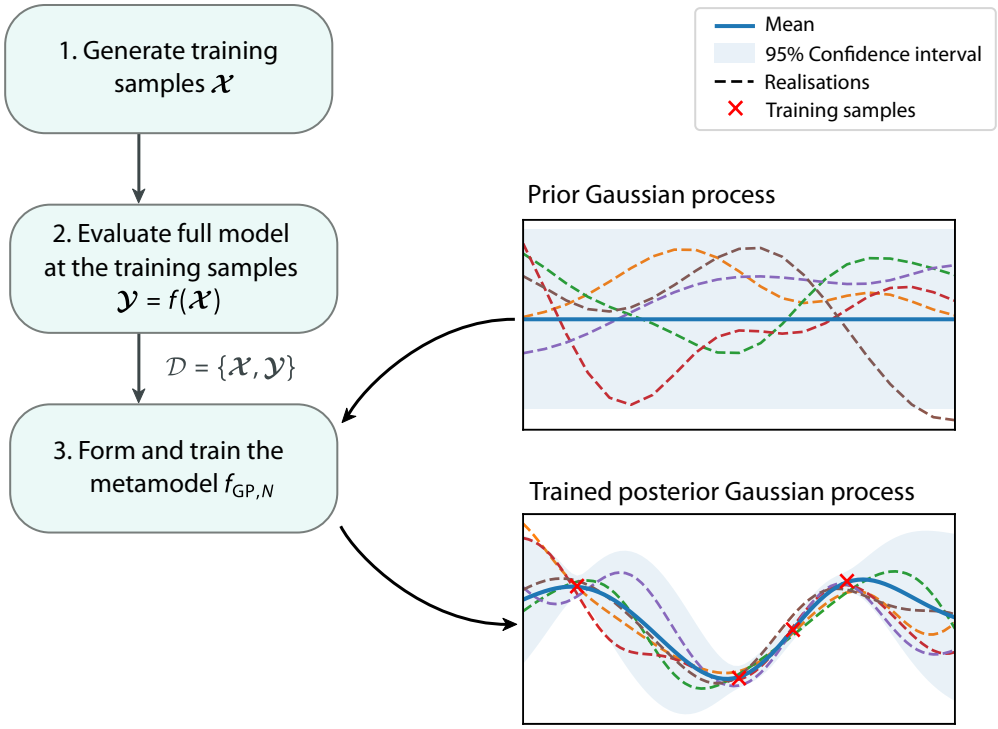


Figure 2. Schematic overview of the steps involved in training the Gaussian process metamodel. Conditioning the prior Gaussian process on the training samples results in the posterior Gaussian process.

Subsequently, we evaluate the full model at each training sample: $\boldsymbol{\mathcal{Y}} = f(\boldsymbol{\mathcal{X}})$ with $\boldsymbol{\mathcal{Y}}$, hence, being an $N \times 1$ vector. This results in the training data set

$$\mathcal{D} = \{\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}\} \tag{11}$$

where each row corresponds to one training point.

Finally, we have to *train* the metamodel. We use a GP f_{GP} as metamodel similar to [45, 46, 14] and summarise what training means for GPs below. Note that we only include a compact overview of the most relevant concepts used in this paper. For more details, the reader is referred e.g. to [13].

A GP defines a distribution over functions such that any finite set of function values $f_{\mathrm{GP}}(\boldsymbol{x}^{(1)})$, $f_{\mathrm{GP}}(\boldsymbol{x}^{(2)})$, $\ldots$, $f_{\mathrm{GP}}(\boldsymbol{x}^{(n)})$ has a joint Gaussian distribution [13, 47]. From a Bayesian point of view, we distinguish between prior and posterior: the prior GP reflects our beliefs about the metamodel before seeing any (training) data, and the posterior GP is then conditioned on the (training) data, i.e. includes the knowledge from the data (see Fig. 2). This conditioning on the data is what we refer to as training.

The prior GP $f_{\mathrm{GP}}(\boldsymbol{X})$ is given by

$$f_{\mathrm{GP}}(\boldsymbol{X}) \sim \mathrm{GP}(m_{\mathrm{GP}}(\boldsymbol{X}), k(\boldsymbol{X}, \boldsymbol{X}')) \tag{12}$$

and is completely specified by its mean function $m_{\mathrm{GP}}(\boldsymbol{X})$ and its covariance function $k(\boldsymbol{X}, \boldsymbol{X}')$ between all possible pairs $(\boldsymbol{X}, \boldsymbol{X}')$. The covariance function is a positive definite kernel, e.g., the squared exponential covariance function (also called radial basis function)

$$k(\boldsymbol{X}, \boldsymbol{X}') = \sigma_f^2 \exp\left(-\frac{1}{2\ell^2}||\boldsymbol{X} - \boldsymbol{X}'||^2\right) \tag{13}$$

with the characteristic length scale ℓ, variance parameter σ_f and $||\cdot||$ denoting the Euclidean L2-norm. We assume that the prior mean function is zero: $m_{\mathrm{GP}}(\boldsymbol{X}) = \boldsymbol{0}$, which is common practice and does not limit the GP model, as any uncertainty about the mean function can be included in the choice of a covariance function [47]. Different covariance functions exist and can be combined, for example, through multiplication or addition. [47, Chap. 2] presents a concise overview of different covariance functions for GPs and how to use them to express the structure of the data. The choice of a suitable covariance function is essential since the more a-priori knowledge goes into choosing the covariance function, the fewer data we need to train the metamodel [48].

As described above, we observe the output at N training points. The goal, then, is to predict the output at N_* new points summarised in $\boldsymbol{\mathcal{X}}_*$; in our case, those new points (where we predict the output) will be the Monte–Carlo samples for the estimation of the Sobol indices.

Remember that Eq. (12) is only the prior distribution and does not yet incorporate our knowledge from the training data. To obtain the posterior, we now condition the prior GP on our set of N training data points. This conditioning results in the key predictive equations

$$f_{\mathrm{GP},N}(\boldsymbol{\mathcal{X}_*}) \sim \mathrm{GP}(m_{\mathrm{GP},N}(\boldsymbol{\mathcal{X}}_*), k_N(\boldsymbol{\mathcal{X}}_*, \boldsymbol{\mathcal{X}}_*)) \text{ with} \tag{14}$$

$$m_{\mathrm{GP},N}(\boldsymbol{\mathcal{X}}_*) = \boldsymbol{K}_*^{\mathrm{T}} \boldsymbol{K}_\varepsilon^{-1} \boldsymbol{\mathcal{Y}}, \tag{15}$$

$$k_N(\boldsymbol{\mathcal{X}}_*, \boldsymbol{\mathcal{X}}_*) = \boldsymbol{K}_{**} - \boldsymbol{K}_*^{\mathrm{T}} \boldsymbol{K}_\varepsilon^{-1} \boldsymbol{K}_*. \tag{16}$$

In Eqs. (15) and (16), $\boldsymbol{K}_\varepsilon = \boldsymbol{K} + \sigma_y^2 \boldsymbol{I}$, where $\boldsymbol{K} = k(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}})$ denotes the $N \times N$ matrix we obtain when evaluating the covariance function (given in Eq. (13)) for all pairs of training points, similarly for $\boldsymbol{K}_* = k(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{X}}_*)$ and $\boldsymbol{K}_{**} = k(\boldsymbol{\mathcal{X}}_*, \boldsymbol{\mathcal{X}}_*)$. We use σ_y^2 as an artificially introduced variable nugget term to alleviate numerical problems [49, 50]. The

hyperparameters $\boldsymbol{\theta} = (\sigma_f, \ell)$ are optimised by maximising the log marginal likelihood using a gradient-based optimiser. The log marginal likelihood is given by

$$\log p(\boldsymbol{\mathcal{Y}}|\boldsymbol{\mathcal{X}}) = -\frac{1}{2}\boldsymbol{\mathcal{Y}}^{\mathrm{T}}\boldsymbol{K}_{\varepsilon}^{-1}\boldsymbol{\mathcal{Y}} - \frac{1}{2}\log|\boldsymbol{K}_{\varepsilon}| - \frac{N}{2}\log 2\pi \tag{17}$$

with $|\cdot|$ denoting the determinant. Maximising the log marginal likelihood given by Eq. (17) with respect to the hyperparameters $\boldsymbol{\theta}$ automatically incorporates a trade-off between model fit and model complexity: the first term in Eq. (17) penalises the model's failure to describe the data while the second term penalises high model complexity. Thus, this favours the least complex model that is able to explain the data [13].

One advantage of employing GPs as a metamodel is that predictions can be computed exactly in a closed form [47] and that GPs inherently provide uncertainty measures over the predictions. Moreover, one can incorporate a wide range of modelling assumptions into the choice of the covariance function. However, note that computing the inverse in the first term in Eq. (17) (and the determinant in the second term) is computationally expensive, i.e., on the order $\mathcal{O}(N^3)$. This cubic complexity results in slow inference as the number of training samples increases. One further challenge of using GPs as a metamodel is that they are susceptible to the curse of dimensionality: as the dimensionality of the input space increases, the number of training samples required to train the metamodel grows exponentially [51, 52] and the optimisation of hyperparameters $\boldsymbol{\theta}$ becomes impractical.

Remark (Advanced GPs)**.** *In case of large numbers of training samples and/or input space dimensions, various advanced GP metamodels are available: [53] review approaches to improve the scalability of GPs to large data sets, e.g., by using stochastic variational inference [54]; [52] present an approach with built-in dimensionality reduction.*

2.4 Estimation of Sobol indices and their uncertainty

To estimate the Sobol indices, we now use the estimators given by Eqs. (7), (8) and (10) and substitute the realisations of the full model f with those of the trained GP metamodel $f_{\mathrm{GP},N}$ as suggested by [14]:

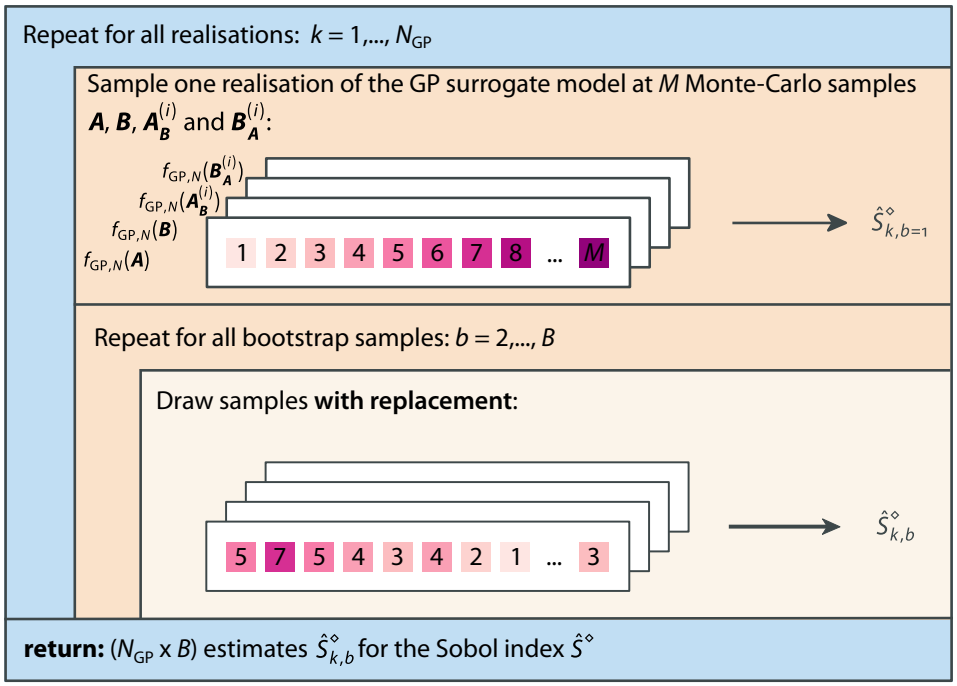
$$\hat{S}^i = \frac{\frac{1}{M}\sum_{m=1}^{M} f_{\mathrm{GP},N}(\boldsymbol{B})_m \left(f_{\mathrm{GP},N}(\boldsymbol{A}_{\boldsymbol{B}}^{(i)})_m - f_{\mathrm{GP},N}(\boldsymbol{A})_m\right)}{\mathcal{V}[f_{\mathrm{GP},N}([\boldsymbol{A}\ \boldsymbol{B}])]}, \tag{18}$$

$$\hat{S}^{Ti} = \frac{\frac{1}{2M}\sum_{m=1}^{M}(f_{\mathrm{GP},N}(\boldsymbol{A})_m - f_{\mathrm{GP},N}(\boldsymbol{A}_{\boldsymbol{B}}^{(i)})_m)^2}{\mathcal{V}[f_{\mathrm{GP},N}([\boldsymbol{A}\ \boldsymbol{B}])]}, \tag{19}$$

$$\hat{S}^{ij} = \frac{\frac{1}{M}\sum_{m=1}^{M} f_{\mathrm{GP},N}(\boldsymbol{B}_{\boldsymbol{A}}^{(i)})_m\, f_{\mathrm{GP},N}(\boldsymbol{A}_{\boldsymbol{B}}^{(j)})_m - f_{\mathrm{GP},N}(\boldsymbol{A})_m\, f_{\mathrm{GP},N}(\boldsymbol{B})_m}{\mathcal{V}[f_{\mathrm{GP},N}([\boldsymbol{A}\ \boldsymbol{B}])]} - \hat{S}^i - \hat{S}^j \tag{20}$$

with M again being the number of Monte–Carlo samples. We summarise the estimates as $\hat{S}^{\diamond}$ with $\diamond \in \{i, Ti, ij\}$ for the first, total, or second-order index estimates, respectively. Remember that we now evaluate the Monte–Carlo samples with the metamodel instead of the full model. We can, therefore, afford considerably larger numbers of Monte–Carlo samples. Since we sample realisations of the GP metamodel $f_{\mathrm{GP},N}$, the resulting estimates $\hat{S}^{\diamond}$ are again random variables. These include two sources of uncertainty: one related to the metamodel approximation and one related to the Monte–Carlo integration. To estimate those uncertainties, and additionally the total uncertainty, we employ the algorithm suggested by [14]. The steps described in the following can equally be applied to all indices of different order.

Algorithm 1. Calculation of $(N_{\mathrm{GP}} \times B)$ estimates $\hat{S}^{\diamond}_{k,b}$ of the Sobol index. We sample N_{GP} realisations of the GP metamodel and subsequently resample each realisation B times using the bootstrap technique as suggested by Le Gratiet *et al.* [14].



We visually summarise the approach in Algorithm 1; a more detailed version is included in the Supplement A.4. The core idea is to sample N_{GP} realisations of the GP metamodel and, subsequently, resample each realisation B times using the bootstrap technique [55]. This results in $N_{\mathrm{GP}} \times B$ estimates $\hat{S}^{\diamond}_{k,b}$ of the respective Sobol index. We then calculate the mean as

$$\bar{S}^{\diamond} = \frac{1}{N_{\mathrm{GP}} B} \sum_{k=1}^{N_{\mathrm{GP}}} \sum_{b=1}^{B} \hat{S}^{\diamond}_{k,b} \tag{21}$$

and the total variance as

$$\hat{\sigma}^2(S^{\diamond}) = \frac{1}{N_{\mathrm{GP}} B - 1} \sum_{k=1}^{N_{\mathrm{GP}}} \sum_{b=1}^{B} \left(\hat{S}^{\diamond}_{k,b} - \bar{S}^{\diamond} \right)^2 . \tag{22}$$

Since this estimator includes two sources of uncertainty (one related to the metamodel approximation and one related to the Monte–Carlo integration), we decompose the variance of $S^{\diamond}$ as

$$\hat{\sigma}^2(S^{\diamond}) = \underbrace{\hat{\sigma}^2_{\mathrm{GP}}(S^{\diamond})}_{\text{metamodel}} + \underbrace{\hat{\sigma}^2_{\mathrm{MC}}(S^{\diamond})}_{\text{Monte–Carlo}} . \tag{23}$$

Sampling realisations of the metamodel $f_{\mathrm{GP},N}(\boldsymbol{X})$ as opposed to using only the predictive mean $m_{\mathrm{GP},N}(\boldsymbol{X})$ allows us to take into account the covariance structure of the metamodel. The part of the variance related to the metamodel approximation can

be estimated as

$$\hat{\sigma}^2_{\mathrm{GP}}(S^\diamond) = \frac{1}{B}\sum_{b=1}^{B}\frac{1}{N_{\mathrm{GP}}-1}\sum_{k=1}^{N_{\mathrm{GP}}}\left(\hat{S}^\diamond_{k,b} - \bar{\hat{S}}^\diamond_b\right)^2 \quad \text{with} \quad \bar{\hat{S}}^\diamond_b = \frac{1}{N_{\mathrm{GP}}}\sum_{k=1}^{N_{\mathrm{GP}}}\hat{S}^\diamond_{k,b}. \tag{24}$$

Alternatively, [56] present an approach to estimate an upper bound for the metamodel error based directly on the covariance function of the GP, but their approach only provides a rough upper bound [14]. In addition, [57] present an approach to investigate the accuracy of Sobol indices based on a general relation between the accuracy of an arbitrary metamodel and the error of the estimated indices.

To estimate the uncertainty related to Monte–Carlo integration, we use the bootstrap technique [55]. The Monte–Carlo samples $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{A}^{(i)}_{\boldsymbol{B}}$ and $\boldsymbol{B}^{(i)}_{\boldsymbol{A}}$ are resampled (i.e. sampled with replacement) B times as depicted in Algorithm 1. We then calculate $\hat{S}^\diamond$ according to Eqs. (18), (19) and (20) for each bootstrap sample, resulting in B estimates for the Sobol index for each realisation k of the metamodel. The part of the variance related to the Monte–Carlo integration is given by

$$\hat{\sigma}^2_{\mathrm{MC}}(S^\diamond) = \frac{1}{N_{\mathrm{GP}}}\sum_{k=1}^{N_{\mathrm{GP}}}\frac{1}{B-1}\sum_{b=1}^{B}\left(\hat{S}^\diamond_{k,b} - \bar{\hat{S}}^\diamond_k\right)^2 \quad \text{with} \quad \bar{\hat{S}}^\diamond_k = \frac{1}{B}\sum_{b=1}^{B}\hat{S}^\diamond_{k,b}. \tag{25}$$

The bootstrap technique is based on the fact that sampling with replacement from a set of independent, identically distributed data equals sampling from the empirical distribution function of the data [58]. It is important to note that bootstrapping does not require further model evaluations. For a general introduction to the bootstrap technique, the reader is referred to [59] or [60, Sec. 5.2].

3 Application to nanoparticle-mediated drug delivery in a multiphase tumour-growth model

3.1 Model definition

An excellent example for the proposed overall approach is nanoparticle-mediated drug delivery in a multiphase tumour-growth model as all challenging motivating arguments for our approach are present in this problem class, like complex costly models and a large number of parameters. The tumour-growth model in its original form is based on the works [61, 62, 63]. [20, 21] extended the model to a five-phase model including the vasculature. Finally, [22] included and studied nanoparticle delivery. We use the tumour-growth model as a precursor to generate physically plausible results of the tumour and its microenvironment. Those results then serve as initial condition for the sensitivity analysis of nanoparticle-mediated drug delivery.

3.1.1 Underlying tumour-growth model

The model considers the tumour as a porous structure: the extracellular matrix (ECM) is the solid phase (denoted by superscript s) with several fluid phases filling its pore space. We include three fluid phases: tumour cells, host cells, and interstitial fluid (IF), denoted by superscripts t, h and l, respectively. In addition, the vasculature is modelled as an independent porous network and denoted by superscript v. The governing equations of the model are formulated on the macroscale by employing the Thermodynamically Constrained Averaging Theory (TCAT) [64, 65]. Each phase is modelled in an averaged sense based on volume fractions ε^α with α denoting an arbitrary phase. The sum of all

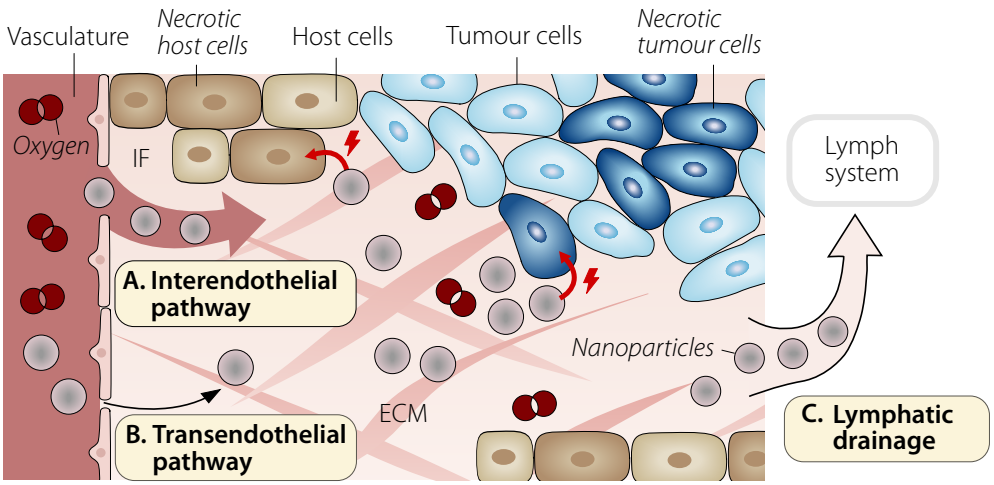


Figure 3. Schematic summary of the components of the multiphase tumour-growth model. The model comprises the ECM, as the solid phase, with three fluid phases filling its pore space: host cells, tumour cells and IF. The vasculature is included as an independent porous network. Species transported by the different phases are: oxygen, nanoparticles, necrotic host cells and necrotic tumour cells (marked in italics). We consider three mass transfer terms that transport nanoparticles into and out of the IF: A. the interendothelial pathway; B. the transendothelial pathway; C. lymphatic drainage (all transfer terms are marked in bold). Drugs mediated by the nanoparticles kill tumour and host cells (marked by red lightning). The lymph system is not explicitly included in the model.

phases must satisfy the equation

$$\varepsilon^s + \varepsilon^t + \varepsilon^h + \varepsilon^l + \varepsilon^v = 1.$$

The different phases additionally transport species. The vasculature and the IF transport oxygen with mass fractions denoted by $\omega^{n\bar{v}}$ and $\omega^{n\bar{l}}$, respectively. Similarly, $\omega^{\mathsf{NP}\bar{v}}$ and $\omega^{\mathsf{NP}\bar{l}}$ denote the mass fractions of nanoparticles in the vasculature and the IF, respectively. Finally, tumour cells and host cells are divided up into living and necrotic cells. The mass fraction of necrotic tumour cells is denoted by $\omega^{N\bar{t}}$ and the mass fraction of necrotic host cells by $\omega^{N\bar{h}}$. Fig. 3 schematically summarises all of the components of our multiphase tumour-growth model that are considered here. Note that the lymph system is not explicitly modelled.

Many commonly used tumour-growth models are data driven and thus adhere to observed data. Our model, by contrast, is based on physical laws. To give just one example: we use Fick's laws to describe the motion of oxygen in IF and Darcy's law to describe the flow of IF through the pores of the extracellular matrix. Such a physics-based approach allows us to describe the system, even under unobserved circumstances. Nevertheless, [6, Sec. 1.1.4] state that physics-based models are customarily over-parametrised: they include more laws and parameters than available data would support. This becomes particularly critical when model parameters are to be determined, e.g. by inverse analysis, and thus sensitivity analysis becomes a crucial part of model development [66].

3.1.2 Nanoparticle-mediated drug delivery

The transport of nanoparticles in our multiphase tumour-growth model is included as described in [22]. We only present a short summary in the following. Further details can be found in the original publication [22].

Nanoparticles are intravenously injected and transported in the vasculature. They then extravasate into the IF and are transported towards tumour cells and host cells. The focus here lies on the extravasation into the IF and the transport therein. Therefore, we do not explicitly include transport in the vasculature, but rather assume a constant mass fraction of nanoparticles in the vasculature $\omega^{\mathsf{NP}\bar{v}}$. The governing equation for the transport of nanoparticles in IF is the mass balance equation of nanoparticles with mass fraction $\omega^{\mathsf{NP}\bar{l}}$

$$\varepsilon^l \frac{\partial \omega^{\mathsf{NP}\bar{l}}}{\partial t} - \underbrace{\frac{\boldsymbol{k}^l}{\mu^l} \boldsymbol{\nabla} p^l \cdot \boldsymbol{\nabla} \omega^{\mathsf{NP}\bar{l}}}_{\substack{\text{convective}\\ \text{transport}}} - \underbrace{\boldsymbol{\nabla} \cdot \left(\varepsilon^l D^{\mathsf{NP}l} \boldsymbol{\nabla} \omega^{\mathsf{NP}\bar{l}} \right)}_{\substack{\text{diffusive}\\ \text{transport}}} = \frac{1}{\rho^l} \left(\underbrace{\sum_{\kappa} \overset{\mathsf{NP}\kappa \to \mathsf{NP}l}{M}}_{\substack{\text{mass transfer}\\ \text{of NP to and from IF}}} - \omega^{\mathsf{NP}\bar{l}} \sum_{\kappa} \overset{\kappa \to l}{M} \right) \tag{26}$$

where the effective diffusivity of nanoparticles in IF is given by $D^{\mathsf{NP}l}$. The superscript l denotes the IF as one of the fluid phases filling the pore space of the extracellular matrix. The IF is characterised by its viscosity μ^l, density ρ^l, permeability tensor $\boldsymbol{k}^l$, and finally the IF pressure p^l resulting from the mass balance of the fluid equation as part of the tumour-growth model. The last term results from employing the product rule, see [61, 20]. The mass transfer of nanoparticles to and from the IF includes three terms

$$\sum_{\kappa} \overset{\mathsf{NP}\kappa \to \mathsf{NP}l}{M} = \underbrace{\overset{\mathsf{NP}v \to \mathsf{NP}l}{M_{\mathsf{inter}}}}_{\text{interendothelial}} + \underbrace{\overset{\mathsf{NP}v \to \mathsf{NP}l}{M_{\mathsf{trans}}}}_{\text{transendothelial}} - \underbrace{\overset{\mathsf{NP}l \to \mathsf{NP}\,\mathsf{ly}}{M_{\mathsf{drain}}}}_{\text{drainage}} \tag{27}$$

where the physical interpretation of the different transport mechanisms is included in Fig. 3. Nanoparticles extravasate from the vasculature into the IF through two different pathways: the interendothelial and the transendothelial pathway [67, 68]. Those pathways are described by

$$\begin{aligned} \overset{\mathsf{NP}v \to \mathsf{NP}l}{M} &= \overset{\mathsf{NP}v \to \mathsf{NP}l}{M_{\mathsf{inter}}} + \overset{\mathsf{NP}v \to \mathsf{NP}l}{M_{\mathsf{trans}}} \\ &= \underbrace{\rho^v \varepsilon^v L_p^v \frac{S}{V} \left[p^v - p^l - \sigma \left(\pi^v - \pi^l \right) \right] \frac{\omega^{\mathsf{NP}\bar{v}} + \omega^{\mathsf{NP}\bar{l}}}{2}}_{\substack{\text{interendothelial}\\ \text{pathway}}} + \underbrace{\rho^v \varepsilon^v P^v \frac{S}{V} \left\langle \omega^{\mathsf{NP}\bar{v}} - \omega^{\mathsf{NP}\bar{l}} \right\rangle_+}_{\substack{\text{transendothelial}\\ \text{pathway}}} \end{aligned} \tag{28}$$

with the oncotic pressure difference between blood vessels and IF $\sigma \left(\pi^v - \pi^l \right)$, the surface-to-volume ration S/V, and Macaulay brackets $\langle \cdot \rangle_+$. This equation is based on the Staverman–Kedem–Katchalsky equation similar to [69].

The first term describes the interendothelial pathway, which is a convective process: nanoparticles are dragged by the transvascular fluid flow through gaps in the blood-vessel wall [69]. This process is governed by the hydraulic conductivity L_p^v of the blood-vessel wall which is defined as

$$L_p^v = \frac{\gamma_p \, r_0^2}{8 \mu^v t} \tag{29}$$

with the pore radius r_0, the vessel-wall thickness t, and the fraction of pores γ_p [70].

The second term in Eq. (28) describes the transendothelial pathway, which is a diffusive process: nanoparticles diffuse through the vessel-wall. This diffusive flux is

governed by the vascular permeability P^v.

Finally, the lymph system absorbs nanoparticles from the IF

$$\overset{\mathsf{NP}l\to\mathsf{NP}\,\mathsf{ly}}{M_{\mathsf{drain}}} = \rho^l \omega^{\mathsf{NP}\bar{l}} \left(L_p \frac{S}{V} \right)^{\mathsf{ly}} \langle p^l \rangle_+ \left\langle 1 - \frac{p^t}{p^{\mathsf{ly}}_{\mathsf{coll}}} \right\rangle_+ \tag{30}$$

governed by the lymphatic filtration coefficient $\left(L_p \frac{S}{V}\right)^{\mathsf{ly}}$. Lymphatic drainage is impaired above the collapsing pressure $p^{\mathsf{ly}}_{\mathsf{coll}}$, and thus no particles are removed.

We further assume that the nanoparticles transport and release anti-cancer drugs. Those kill tumour cells and thus increase the mass fraction of necrotic tumour cells $\omega^{N\bar{t}}$. At the same time, those drugs have adverse side effects and kill host cells. This additionally increases the mass fraction of necrotic host cells $\omega^{N\bar{h}}$. For the sake of simplicity, we assume that the mass fraction of killed cells is directly proportional to the mass fraction of nanoparticles present in the IF at a certain position. We introduce intraphase reaction terms that increase the mass fraction of necrotic tumour cells and host cells according to

$$r^{Nt}_{\mathsf{kill}} = \gamma^t_{\mathsf{kill}} \omega^{\mathsf{NP}\bar{l}} (1 - \omega^{N\bar{t}}) \tag{31}$$

$$r^{Nh}_{\mathsf{kill}} = \gamma^h_{\mathsf{kill}} \omega^{\mathsf{NP}\bar{l}} (1 - \omega^{N\bar{h}}) \tag{32}$$

where γ^t_{kill} and γ^h_{kill} characterise the strength of the drug.

As quantity of interest for the sensitivity analysis, we consider the mean of the necrotic fraction of tumour cells given by

$$\bar{\omega}^{N\bar{t}} = \frac{\int \omega^{N\bar{t}} \, \mathrm{d}\Omega}{A^t} \tag{33}$$

where the tumour size is defined as $A^t = \int \mathcal{H}(S^t - 0.1) \, \mathrm{d}\Omega$ with the Heaviside function $\mathcal{H}(\cdot)$, and S^t denotes the saturation of tumour cells [20]. We define the tumour as the part of the domain where $S^t > 0.1$. In the context of sensitivity analysis, it is important to choose the quantity of interest carefully and to bear in mind that a parameter that is non-influential under one particular investigated condition, e.g., one particular quantity of interest, might be highly influential under a new condition [8].

3.2 Set-up of the numerical example

The set-up presented in the following is similar to the example in [22]. We therefore only present a summary, and again refer the interested reader to the original publication for further details. The major addition to the original example is the nanoparticle-mediated killing of tumour and host cells as described by Eqs. (31) and (32).

We investigate nanoparticle transport and subsequent killing of cells by nanoparticle-mediated drugs. The transport of nanoparticles depends on the hydraulic conductivity of blood-vessel walls L^v_p and the blood-vessel wall permeability P^v (both influence the transport of nanoparticles into the IF), the diffusivity $D^{\mathsf{NP}l}$ of nanoparticles in the IF, and the lymphatic filtration coefficient $\left(L_p \frac{S}{V}\right)^{\mathsf{ly}}$ (influencing the transport of nanoparticles out of the IF). Subsequently, drugs mediated by the nanoparticles kill tumour and host cells depending on the strength of the drug, characterised by γ^t_{kill} and γ^h_{kill}. Note that the amount of killed cells largely depends on the amount of nanoparticles reaching a particular region of the domain, and hence depends on the transport parameters.

The transport of nanoparticles, and thus the question of which regions nanoparticles reach and where drugs can kill cells, essentially depends on the microenvironment of the tumour. Solid tumours exhibit typical features relevant in this context: the majority of

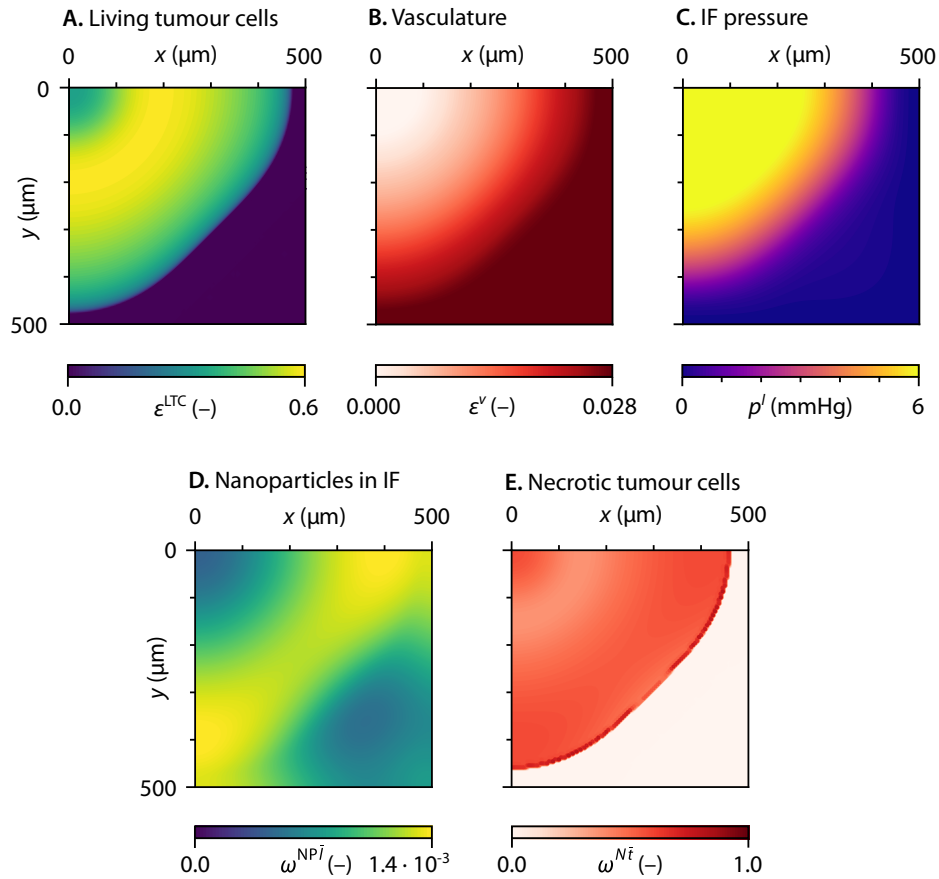


Figure 4. Grown tumour in its microenvironment (A, B, C) with distribution of nanoparticles (D) and necrotic tumour cells (E). A. Volume fraction of living tumour cells $\varepsilon^{\mathrm{LTC}}$; B. Volume fraction of vasculature ε^{v}; C. Pressure p^{l} in the IF; D. Mass fraction of nanoparticles in the IF $\omega^{\mathrm{NP}\bar{l}}$; E. Mass fraction of necrotic tumour cells $\omega^{N\bar{t}}$; Subfigures D and E as an example present the result for the mean values of the uncertain input parameters.

living tumour cells is located in the tumour periphery, whereas the tumour core mainly consists of necrotic cells (see Fig. 4A). In addition, Fig. 4B shows that the volume fraction of the vasculature is considerably lower in the tumour area because the growing tumour collapses blood-vessels. The inner core of the tumour even contains no vessels at all. Finally, the interstitial pressure in the tumour is increased and can reach 6 mmHg (see Fig. 4C). To sum up, the tumour has a necrotic core with collapsed blood vessels as well as an increased interstitial pressure, which is a structure typical for solid tumours and which has also been observed in experiments [71, 72, 73, 74].

We analyse a domain of 1 mm × 1 mm, but due to the symmetry of the problem, we only simulate one quarter of the domain (500 µm × 500 µm). The grown tumour has a radius of 440 µm as presented in Fig. 4. We analyse a time interval of 20 min of nanoparticle transport and killing of cells based on examples by [75, 76]. Assuming the intravenous infusion of nanoparticles, we prescribe a constant value of $\omega^{\mathrm{NP}\bar{v}} = 2.0 \times 10^{-3}$ for the mass fraction of nanoparticles in the vasculature.

Table 1 summarises the six uncertain input parameters included in the sensitivity analysis. We assume that all input parameters are distributed uniformly within the

Table 1. Probability distributions of uncertain input parameters. We assume that all parameters are distributed uniformly within the given range.

Symbol	Parameter	Range	Units	Source
L_p^v	Hydraulic conductivity of blood-vessel wall*	$[7.8;\ 125] \cdot 10^{-8}$	$\mathrm{mm/(Pa\,s)}$	[70]
P^v	Blood-vessel wall permeability	$[3.2;\ 128] \cdot 10^{-5}$	$\mathrm{mm/s}$	[79, 80, 81]
$D^{\mathrm{NP}l}$	Diffusivity of nanoparticles	$[0.26;\ 30.83]$	$\mathrm{\mu m^3/s}$	[81]
$\left(L_p \frac{S}{V}\right)^{\mathrm{ly}}$	Lymphatic filtration coefficient	$[0;\ 5.2] \cdot 10^{-4}$	$\mathrm{Pa^{-1}\,s^{-1}}$	[22]
γ_{kill}^t	Killing coefficient of tumour cells	$[5;\ 10] \cdot 10^{-4}$	$\mathrm{g/(mm^3\,s)}$	–
γ_{kill}^h	Killing coefficient of host cells	$[2;\ 7] \cdot 10^{-4}$	$\mathrm{g/(mm^3\,s)}$	–

$^{(\star)}$ The given values for the hydraulic conductivity of the blood-vessel wall correspond to a pore radius of $r_0 = [50;\ 200]\ \mathrm{nm}$ as used in [22].

given ranges, which are based on experimental data (see references in Table 1). The uniform distribution is chosen because we lack more specific information about the input parameters: given only the range of the input parameters (and no further information such as mean or variance), uniform distributions maximise the information entropy and hence minimise the introduced bias [77, 78]. Note that the killing coefficient of host cells γ_{kill}^h has no influence on our quantity of interest, the mean of the necrotic fraction of tumour cells—neither directly nor indirectly through coupling terms. We nevertheless include the killing coefficient of host cells γ_{kill}^h in the sensitivity analysis to investigate how reliably we can identify a non-influential input parameter as such.

Fig. 4D and E present a result for the distribution of nanoparticles in the IF and for the mass fraction of necrotic tumour cells: for this example, we used the mean values of the six uncertain input parameters given in Table 1. Most nanoparticles accumulate at the edge of the tumour, while lymphatic drainage removes most particles outside the tumour area, and roughly 50% of the tumour cells are necrotic.

The nanoparticle-mediated transport included in the multiphase tumour-growth model is implemented in our in-house research code BACI [82]. The sensitivity analysis methods, as presented above, are implemented in QUEENS [83]. QUEENS is a general purpose framework for uncertainty quantification, physics-informed machine learning, Bayesian optimisation, inverse problems and simulation analytics on distributed computer systems. We use GPy [84] as a GP framework and PyTorch [85] to generate Sobol sequences.

3.3 Predictive quality of the metamodel

Since we use a GP metamodel to estimate the Sobol indices, we first assess its predictive quality. To this end, we investigate the quality of the metamodel predictions for two different covariance functions $k(\boldsymbol{X}, \boldsymbol{X}')$ used for the GP: we compare a tensorised, squared, exponential covariance function to a tensorised 5/2-Matérn covariance function (with $\nu = 5/2$) [86, 87, 13]. A tensorised covariance function has the form $k(\boldsymbol{X}, \boldsymbol{X}') = k_1(X_1, X_1') \cdot k_2(X_2, X_2') \cdot \ldots \cdot k_D(X_D, X_D')$ and as such includes a set of hyperparameters $\boldsymbol{\theta} = (\sigma_{fi}, \ell_i)_{i=1,\ldots,D}$ for all input space dimensions, which we optimise by maximising the log marginal likelihood.

For this comparison, we consider different sizes of training sample sets $N = [10, 15, 20, 25, 30, 40, 60, 80, 100, 150, 200]$, which we generate based on Sobol sequences. Additionally, we generate a set $\mathcal{T}$ of $N_T = 1000$ testing samples disjoint of the training samples. Based on the training samples and the testing samples, we calculate the Nash–Sutcliffe efficiency

Q^2 [88] given by

$$Q^2 = 1 - \frac{\sum_{\boldsymbol{X}\in\mathcal{T}} \left(m_{\mathsf{GP},N}(\boldsymbol{X}) - f(\boldsymbol{X})\right)^2}{\sum_{\boldsymbol{X}\in\mathcal{T}} \left(m_{\mathsf{GP},N}(\boldsymbol{X}) - \bar{f}(\boldsymbol{X})\right)^2} \quad \text{with} \quad \bar{f}(\boldsymbol{X}) = \frac{1}{T}\sum_{\boldsymbol{X}\in\mathcal{T}} f(\boldsymbol{X}) \tag{34}$$

similar to [14]. This is based on the predictive posterior mean $m_{\mathsf{GP},N}(\boldsymbol{X})$ of the GP with optimised hyperparameters, and thus compares the mean prediction of the posterior GP to the actual output of the full model f. A Nash–Sutcliffe efficiency close to 1 indicates good agreement and, hence, reliable predictions. Fig. 5 shows good convergence of the Nash–Sutcliffe efficiency for both covariance functions with values close to 1, even for smaller training sample set sizes. If the number of training samples is restricted due to the computational cost, [89] suggest an algorithm to improve the metamodelling accuracy and efficiency based on sequential sampling.

Note that we use a set of testing samples here that is disjoint of our set of training samples; this means that we also evaluate our full model N_T times, which might be infeasible if the model is computationally more expensive. In those cases, one can use cross-validation methods, such as those explained in [13, Sec. 5.3], where the training set itself is split into two disjoint sets: one is actually used for training and the other for validation.

Both covariance functions yield a very similar predictive quality. In the following, we only use the tensorised, squared, exponential covariance function because it is the default choice in most applications of GPs [47] and, moreover, because it is a universal covariance function [90].

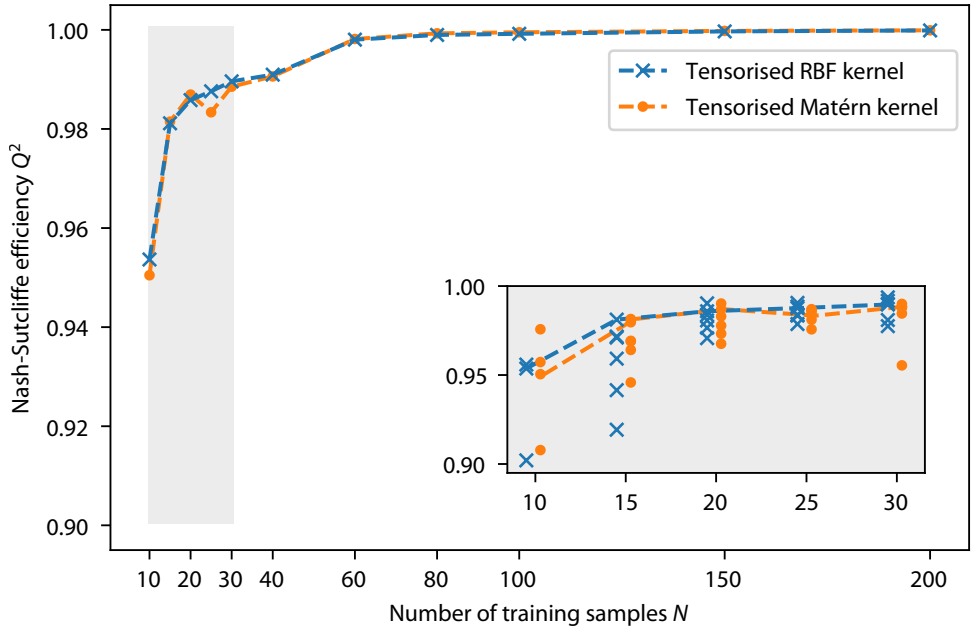


Figure 5. Nash–Sutcliffe efficiency Q^2 for different training sample set sizes with a tensorised, squared, exponential covariance function and a tensorised Matérn covariance function. N training samples were randomly generated for the main plot. For the detail plot, we repeated the process five times with different training sample sets of sizes $N = [10, 15, 20, 25, 30]$. For reference, the dashed lines in the detail plot are identical to the dashed lines in the main plot.

Remark (Randomness of metamodel training)**.** *In Fig. 5 we notice a small kink in the Nash–Sutcliffe efficiency for $N = 25$ in the case of the Matérn covariance function. Therefore, we repeat the training of the GP with other randomly chosen training sample sets for $N = [10, 15, 20, 25, 30]$ to check whether the original training sets happen to perform exceptionally well. The grey detail plot in Fig. 5 presents the results: while the Nash–Sutcliffe efficiency is still above* 0.90 *in all cases, we notice that some training sample sets result in slightly worse efficiencies than others for those smaller sample sizes. This is precisely the case for the Matérn covariance function with $N = 25$ in the main plot. Moreover, the two different covariance functions lead to slightly different efficiencies.*

One possible reason for such behaviour may be the convergence of the optimiser used to optimise the hyperparameters according to Eq. (17). As with all gradient-based optimisers, the optimisation may get stuck in a local minimum. To avoid ending the optimisation in a local minimum, one can repeat the optimisation multiple times from random initial points [52], as provided by the package GPy [84], for example, or use stochastic optimisation, such as Adam optimisation [91].

Further, we take a closer look at the underlying GP, which is depicted in Fig. 6 for 20 training samples. The projection of the GP into the input-space dimensions reveals a linear relation in most dimensions. We only see considerable nonlinearity for the blood-vessel wall permeability P^v. Those characteristics make it much easier to train the GP based on a small number of training samples.

Remark (Projection of the D-dimensional Gaussian process)**.** *The projection $m_{\mathsf{GP}i}(X_i)$ of the D-dimensional posterior mean $m_{\mathsf{GP},N}(\boldsymbol{X})$ into the input space dimension X_i (as presented in Fig. 6) is calculated as follows. First, we uniformly sample discrete values of the posterior mean $m_{\mathsf{GP},N}(\boldsymbol{X})$ in the D-dimensional input space. Second, we project those values over the input space dimension X_i. Third, the results are binned in the X_i-direction, and we calculate the mean and confidence interval for each bin. Note that we only project the posterior mean $m_{\mathsf{GP},N}(\boldsymbol{X})$ and neglect the covariance function k_N here.*

Plotting the model output over a specific input in the form of scatterplots—as done with the training samples in Fig. 6—helps us gain a general understanding of the magnitude of the underlying sensitivity [10]. [6, Sec. 1.2.7] offers a compelling interpretation of scatterplots in relation to the first-order Sobol index: if the conditional expectation $\mathbb{E}_{X_{\sim i}}[Y|X_i]$—here represented by the projection of the mean $m_{\mathsf{GP}i}$ of the GP—has a large variation across X_i, the corresponding input parameter has a high first-order Sobol index. Fig. 6 reveals that the projection of the mean $m_{\mathsf{GP}i}$ is almost constant for the diffusivity of nanoparticles $D^{\mathsf{NP}l}$, the lymphatic filtration coefficient $\left(L_p \frac{S}{V}\right)^{\mathsf{ly}}$, and the killing coefficient of host cells γ^h_{kill}. In contrast, the variation of the projection of the mean $m_{\mathsf{GP}i}$ is larger for the blood-vessel wall L^v_p, the vascular permeability P^v, and the killing coefficient γ^t_{kill}, and we therefore expect the output to be highly sensitive to those parameters.

3.4 First-order Sobol index estimates

We now assess the convergence of the first-order Sobol index estimates for increasing numbers of training samples. To calculate the mean $\bar{S}^i$ based on Eq. (21), we use $N_{\mathrm{GP}} = 500$ metamodel realisations, and the number of Monte–Carlo samples is set to $M = 10\,000$. We do not include bootstrapping here. The uncertainties in the estimates will be studied in the next section.

Fig. 7 presents the results for training sample set sizes $N = [10, \ldots, 200]$. The result confirms what we expected based on the scatterplots in the previous section: three

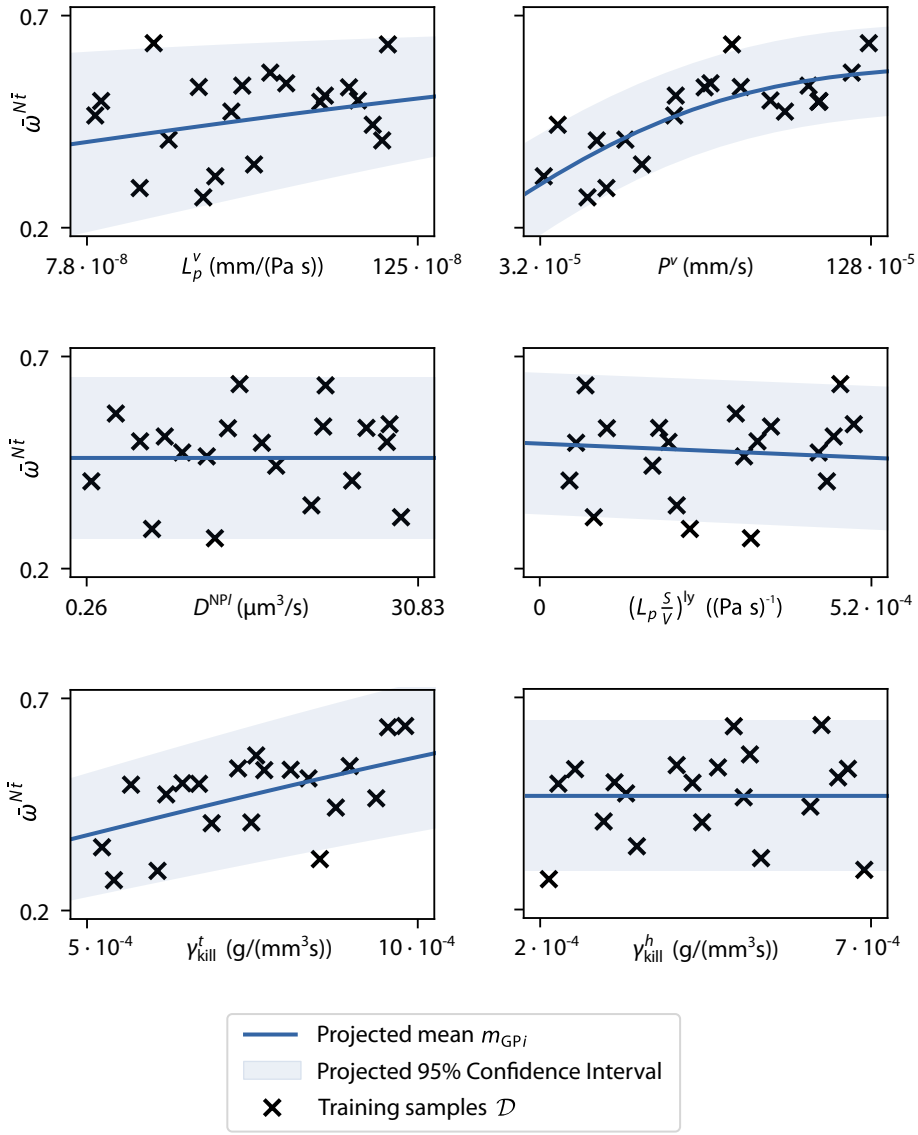


Figure 6. Gaussian process for $N = 20$ with a tensorised, squared, exponential covariance function. Projected mean $m_{\text{GP}i}(X_i)$, projected 95% confidence interval (CI) and training samples $\mathcal{D}$ for the mean of the necrotic fraction of tumour cells $\bar{\omega}^{N\bar{t}}$ (the y-axis labels apply to both figures).

parameters—namely the vascular permeability P^v, the killing coefficient γ^t_{kill}, and the hydraulic conductivity of the blood-vessel wall L^v_p—have considerably higher first-order Sobol indices than the remaining three parameters. Fig. 7 also allows to assess the convergence of the Sobol indices for an increasing number of training samples: even small sizes of training sample sets yield values close to the value based on $N = 200$. This is due to the high values of the respective Nash–Sutcliffe efficiency, as discussed in the previous section. Those results are promising, in particular for models that are computationally very expensive, and thus do not allow a large number of evaluations of the full model: the computational cost of the demonstrated approach is considerably reduced compared to an analysis based directly on evaluations of the full model, as for example presented in [92, 93] for tumour-growth models.

We assessed convergence visually based on Fig. 7. In addition, [94] present a thorough definition of convergence criteria for global sensitivity analysis results. Nevertheless, the computationally limiting factor is usually the number of Monte–Carlo samples. Since we evaluate the Monte–Carlo samples on the GP metamodel, this limitation is less critical in the presented workflow. If sampling the realisations of the metamodel for very large numbers of Monte–Carlo samples becomes an issue, [14] include an efficient approach based on conditional GPs.

3.5 Uncertainties of Sobol index estimation

We now go on to not only estimate the mean $\bar{S}^i$ but also include the uncertainty related to the metamodel and to the Monte–Carlo integration given by Eqs. (24) and (25). In addition to the first-order index S^i, we also include the total-order Sobol index S^{Ti}. Again, we use different training sample set sizes $N = [10, \ldots, 200]$ and a tensorised, squared, exponential covariance function with hyperparameters optimised based on maximising the log marginal likelihood of the GP. We draw $N_{\mathrm{GP}} = 500$ realisations of the GP, $B = 300$ bootstrap samples, and $M = 10\,000$ Monte–Carlo samples. We calculate 95% confidence intervals on the basis of the variance related to the metamodel $\hat{\sigma}^2_{\mathrm{GP}}$ and the variance related to Monte–Carlo integration $\hat{\sigma}^2_{\mathrm{MC}}$.

Fig. 8A presents the results for all six input parameters. Note the different scaling on the vertical axes. We first take a look at the results for the indices themselves. Fig. 8A again confirms that even for small numbers of training samples, the estimates for first and the total-order Sobol indices rapidly converge. As mentioned above, we do not expect any influence of the parameter γ^h_{kill} on the quantity of interest. Fig. 8A shows that we can identify this non-influential parameter as such even for small numbers of training samples. The parameter $D^{\mathsf{NP}l}$ also leads to Sobol indices close to zero for $N < 60$. For larger training sample set sizes however we get a slightly higher total-order Sobol index, which is nevertheless small. Hence, we can clearly separate the three most influential parameters from the three non-influential parameters.

Moreover, the total-order index is higher than the first-order index, in particular for the hydraulic conductivity of the blood-vessel wall L^v_p and the vessel wall permeability P^v. This leads to the conclusion that higher-order effects are indeed present. We will therefore analyse the second-order Sobol indices in the next section. In addition, Table 2 summarises the values for the first and the total-order indices for 200 training samples: the sum of all first-order Sobol indices is 0.967. Since this is close to one, we conclude that higher-order effects are present, but only play a minor role. The largest part of the output variance is covered by the first-order indices.

We now focus on the uncertainties: we assess the uncertainty related to the GP metamodel and the total uncertainty, where the latter includes both sources of uncertainty (related to Monte–Carlo integration and related to the metamodel). For small training sample set sizes, we see considerable uncertainty related to the metamodel (depicted in light blue/orange in Fig. 8A). However, the uncertainty related to Monte–Carlo

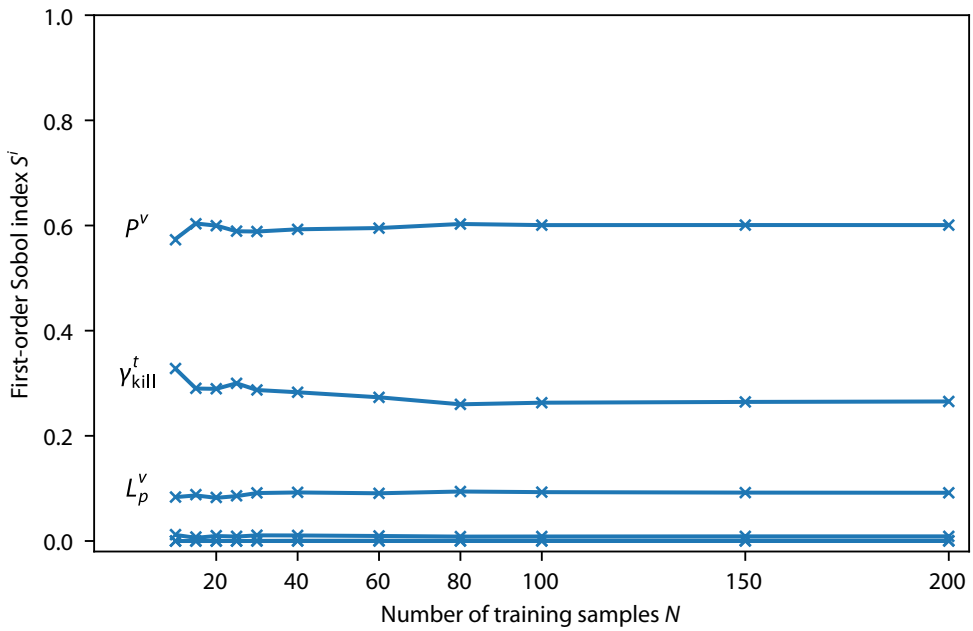


Figure 7. Convergence of first-order Sobol index estimate for increasing training sample set size. We use $M = 10\,000$ Monte–Carlo samples and $N_{\text{GP}} = 500$ realisations of the Gaussian process.

Table 2. First and total-order Sobol indices based on 200 training samples. We use $M = 10\,000$ Monte–Carlo samples, $N_{\text{GP}} = 500$ realisations of the Gaussian process, and $B = 300$ bootstrap samples.

Parameter	S^i	S^{Ti}
L^v_p	0.092	0.121
P^v	0.600	0.632
$D^{\text{NP}l}$	0.001	0.004
$\left(L_p \frac{S}{V}\right)^{\text{ly}}$	0.009	0.011
γ^t_{kill}	0.265	0.270
γ^h_{kill}	0.000	0.000
sum	0.967	

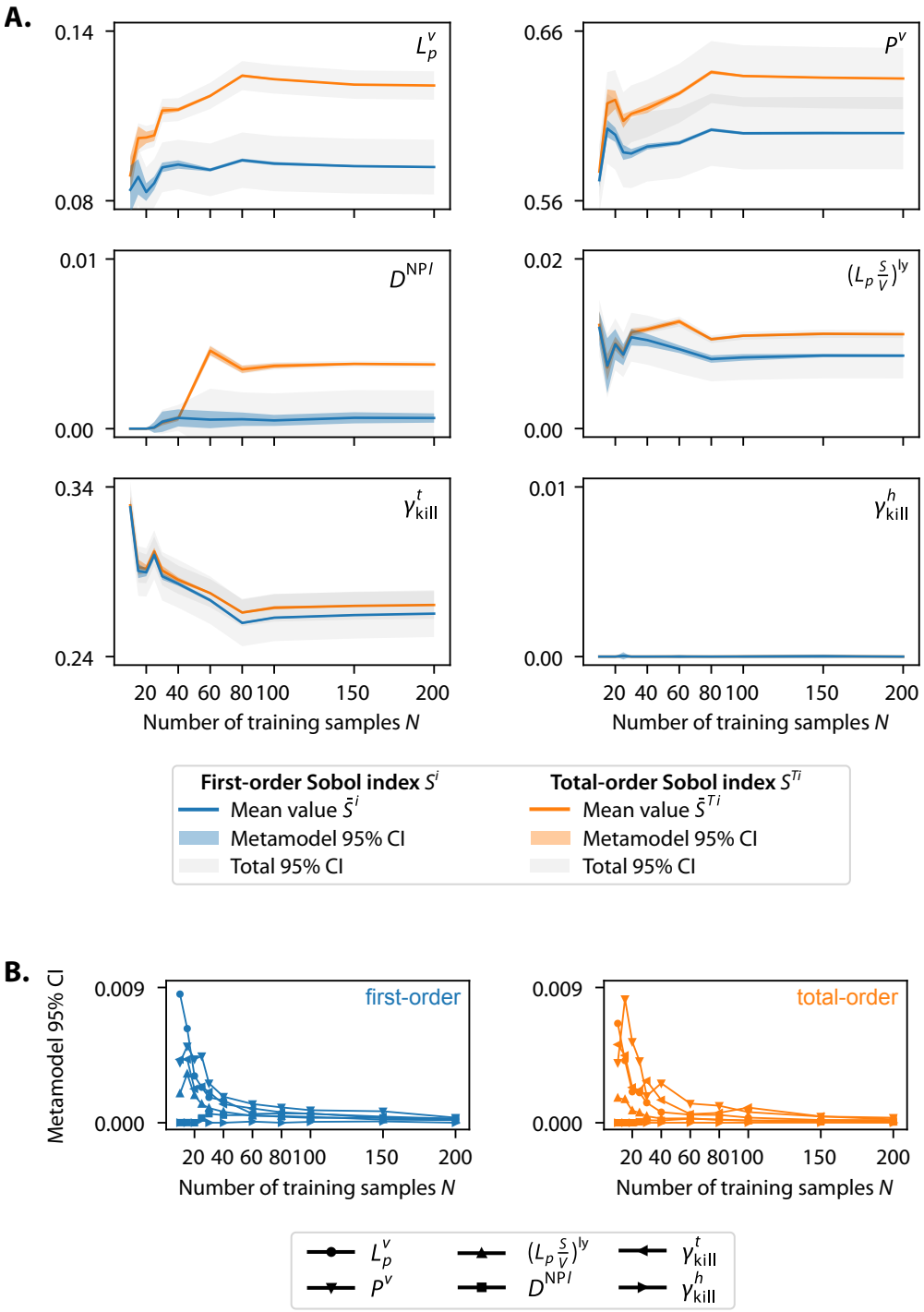


Figure 8. First-order and total-order Sobol indices and 95% confidence intervals (CI) for an increasing number of training samples. We use $M = 10\,000$ Monte–Carlo samples, $N_{GP} = 500$ realisations of the Gaussian process, and $B = 300$ bootstrap samples. Subfigure A shows Sobol indices with metamodel CI and the sum of metamodel and Monte–Carlo CI for the six input parameters separately. Monte–Carlo abbreviates as MC. Subfigure B details the metamodel CI.

integration dominates for $N > 40$. We therefore present the uncertainty related to the metamodel in detail in Fig. 8B: the uncertainty rapidly decreases as the number of training samples increases for both the first-order and the total-order index and becomes one order of magnitude smaller than the uncertainty related to Monte–Carlo integration. [18] also found the uncertainty related to the GP metamodel to be much smaller than the uncertainty related to Monte–Carlo integration in their example. The total uncertainty (depicted in grey) could be reduced even further by increasing the number of Monte–Carlo samples.

Based on these results, we conclude that including the uncertainty related to the metamodel is not absolutely necessary in our example. However, the example presented in the outlook and the example presented by [14] illustrate that this is not always the case: only taking into account the Monte–Carlo uncertainty might then underestimate the confidence interval. In such cases, it is essential to consider the uncertainty related to the metamodel. Hence, this largely depends on the model, the input parameters, and the quantity of interest, and no *one-size-fits-all* rule can be given.

Nevertheless, even taking into account the metamodel and the Monte–Carlo uncertainty may incorrectly estimate the confidence intervals: poor optimisation of the hyperparameters may result in underestimated or overestimated confidence intervals. In such cases, one could additionally consider the uncertainty related to the estimation of the hyperparameters of the GP covariance function by using a full-Bayesian approach with hyperpriors [14].

To sum up, the demonstrated workflow not only identifies parameters with a high first-order Sobol index (necessary for factor prioritisation) but also parameters with a small total-order Sobol index (necessary for factor fixing). In both cases, small numbers of training samples suffice in our example.

3.6 Second-order Sobol index estimation

Since we concluded from the results in the previous sections that higher-order effects are indeed present in our example, the goal now is to estimate the second-order Sobol indices, and thereby identify interaction effects between the input parameters. The results in the previous section show that the uncertainty related to Monte–Carlo integration is dominant, and the uncertainty related to the GP metamodel is much smaller. We therefore estimate the second-order indices based on the predictive mean $m_{\mathsf{GP},N}(\boldsymbol{X})$ of the GP and do not take into account the uncertainty related to the metamodel. Thus, we estimate the second-order Sobol indices as

$$\hat{S}^{ij} = \frac{\frac{1}{M}\sum_{m=1}^{M} m_{\mathsf{GP},N}(\boldsymbol{B}_{\boldsymbol{A}}^{(i)})_m \, m_{\mathsf{GP},N}(\boldsymbol{A}_{\boldsymbol{B}}^{(j)})_m - m_{\mathsf{GP},N}(\boldsymbol{A})_m \, m_{\mathsf{GP},N}(\boldsymbol{B})_m}{\mathcal{V}[m_{\mathsf{GP},N}([\boldsymbol{A}\ \boldsymbol{B}])]} - \hat{S}^i - \hat{S}^j \tag{35}$$

which includes estimating the first-order effects $\hat{S}^i$ and $\hat{S}^j$ also based on the mean of the GP only.

Using the same number of Monte–Carlo samples as before ($M = 10\,000$), however, results in a 95% confidence interval with the same order of magnitude as the indices themselves, and even leads to negative values for the Sobol indices (as given in Table 3). Therefore, we increase the number of Monte–Carlo samples to $M = 1\,000\,000$ to obtain reasonably small confidence intervals.

Table 3 and Fig. 9 summarise the results for the second-order Sobol indices: the highest interaction is present between P^v and L_p^v, as we already expected based on the results presented in Fig. 8. Summing up all first and second-order Sobol indices results in 0.999. We thus (almost) completely apportioned the variance in the output to the input parameters, including interaction effects.

Table 3. Second-order Sobol indices for $M = 1\,000\,000$ Monte–Carlo samples and $B = 300$ bootstrap samples. All other second-order indices are $S^{ij} < 0.001$.

Parameter i	Parameter j	$M = 10\,000$		$M = 1\,000\,000$	
		S^{ij}	95% CI	S^{ij}	95% CI
L_p^v	P^v	0.0276	0.0198	0.0283	0.0021
L_p^v	$D^{\mathrm{NP}l}$	−0.0009	0.0136	0.0003	0.0016
L_p^v	γ_{kill}^t	−0.0004	0.0142	0.0007	0.0017
P^v	γ_{kill}^t	0.0036	0.0312	0.0042	0.0030
$(L_p \frac{S}{V})^{\mathrm{ly}}$	$D^{\mathrm{NP}l}$	0.0021	0.0037	0.0023	0.0004

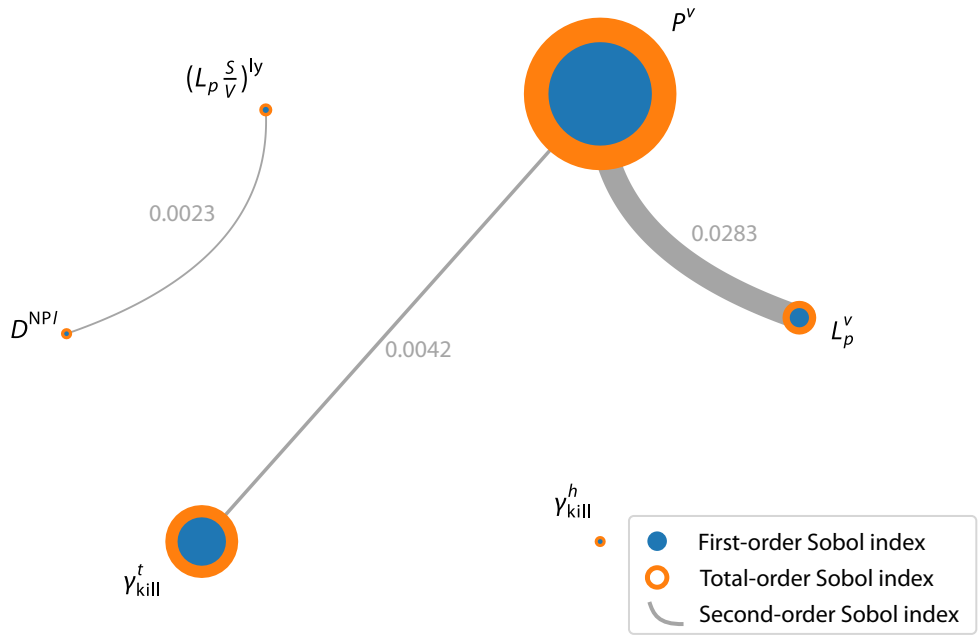


Figure 9. Second-order Sobol indices $\hat{S}^{ij}$ for $M = 1\,000\,000$ Monte–Carlo samples and $B = 300$ bootstrap samples. The blue circles represent the first-order Sobol indices. The orange surrounds represent the total-order Sobol indices. The grey areas connecting the nodes represent the second-order Sobol indices. All other second-order indices are $\hat{S}^{ij} < 0.001$.

The large number of Monte–Carlo samples necessary to estimate the second-order Sobol indices highlights the relevance of metamodel-based estimation approaches. Evaluating the full model f several million times is computationally prohibitive for most models. Without using a metamodel, estimating higher-order Sobol indices is thus impossible in most cases.

4 Application to a model of arterial growth and remodelling

In the previous sections, we assessed the performance of the current workflow as applied to a model of nanoparticle-mediated drug delivery in more detail: even small numbers of training samples result in reliable estimates of the Sobol indices and a small uncertainty related to the GP metamodel. To give an outlook, we now apply the workflow to another complex biomechanical example, namely a homogenised, constrained mixture model of arterial growth and remodelling [23, 24, 25].

[95] performed an exhaustive global sensitivity analysis, where they estimated the first and total-order Sobol indices by evaluating the full model for all Monte–Carlo samples. This however entails a large computational burden ($> 70\,000$ model evaluations). Therefore, the question arises as to whether we can reduce this computational cost by using a GP metamodel and still get reliable Sobol index estimates, including reliable uncertainty estimates.

For this comparison, we investigate *Case 2* of the original publication [95], where the maximum diameter of an idealised cylindrical abdominal aorta was studied 15 years after spontaneous damage to elastin. In this case, the majority of samples lead to minor dilatation of the vessel $d_{\mathsf{max}} < 3\,\mathrm{cm}$. In contrast, a considerable number of samples do not stabilise and keep enlarging, leading to aneurysms with a much larger diameter $d_{\mathsf{max}} >> 3\,\mathrm{cm}$ (see Fig 4b in [95]). We use the original results from [95] as a reference and compare them to our results based on the metamodel approach.

First, we take a look at the predictive quality of the GP metamodel. Once again, we use a tensorised, squared, exponential covariance function and compare the results for different numbers of training samples, $N = [40, 60, 80, 100, 150, 200, 300, 500]$ in this case. As an example, Fig. 10A presents the training samples for $N = 300$ for two parameters. The results for the remaining parameters are included in the Supplement (see Fig A.1 in Supplement). We see that the majority of samples result in a small dilatation in contrast to the fewer aneurysmatic samples with a very large diameter of up to $8\,\mathrm{cm}$. This bimodal structure of the data makes training the GP metamodel more difficult compared to our previous example. Accordingly, the Nash–Sutcliffe presented in Fig. 10B is lower, particularly for small numbers of training samples, i.e. $N < 80$.

Second, we calculate the first and total-order Sobol indices and respective uncertainties for different numbers of training samples. To this end, we use $M = 10\,000$ Monte–Carlo samples, $N_{\mathrm{GP}} = 500$ realisations of the GP metamodel, and $B = 300$ bootstrap samples. By way of example, we present the results for two parameters, the gain parameter $\bar{k}_\sigma$ and the initial volume fraction of elastin $\phi^{\mathsf{el}}_{t_0}$, in Fig. 10C. Similar plots for the remaining eight parameters are included in the Supplement (see Fig. A.2 in Supplement). For the gain parameter $\bar{k}_\sigma$, the estimates based on the metamodel converge to the reference values from [95] for both the first and the total-order Sobol index. For the initial volume fraction of elastin $\phi^{\mathsf{el}}_{t_0}$, the reference values for the Sobol indices are very small (0.01 or smaller). In this case, exact estimates based on the metamodel approach are much harder to achieve: for $N \leq 300$ training samples the estimates only stabilise. Nevertheless, we can still reliably separate the three most influential parameters from the non-influential parameters, even for small numbers of training samples (see Fig. A.2

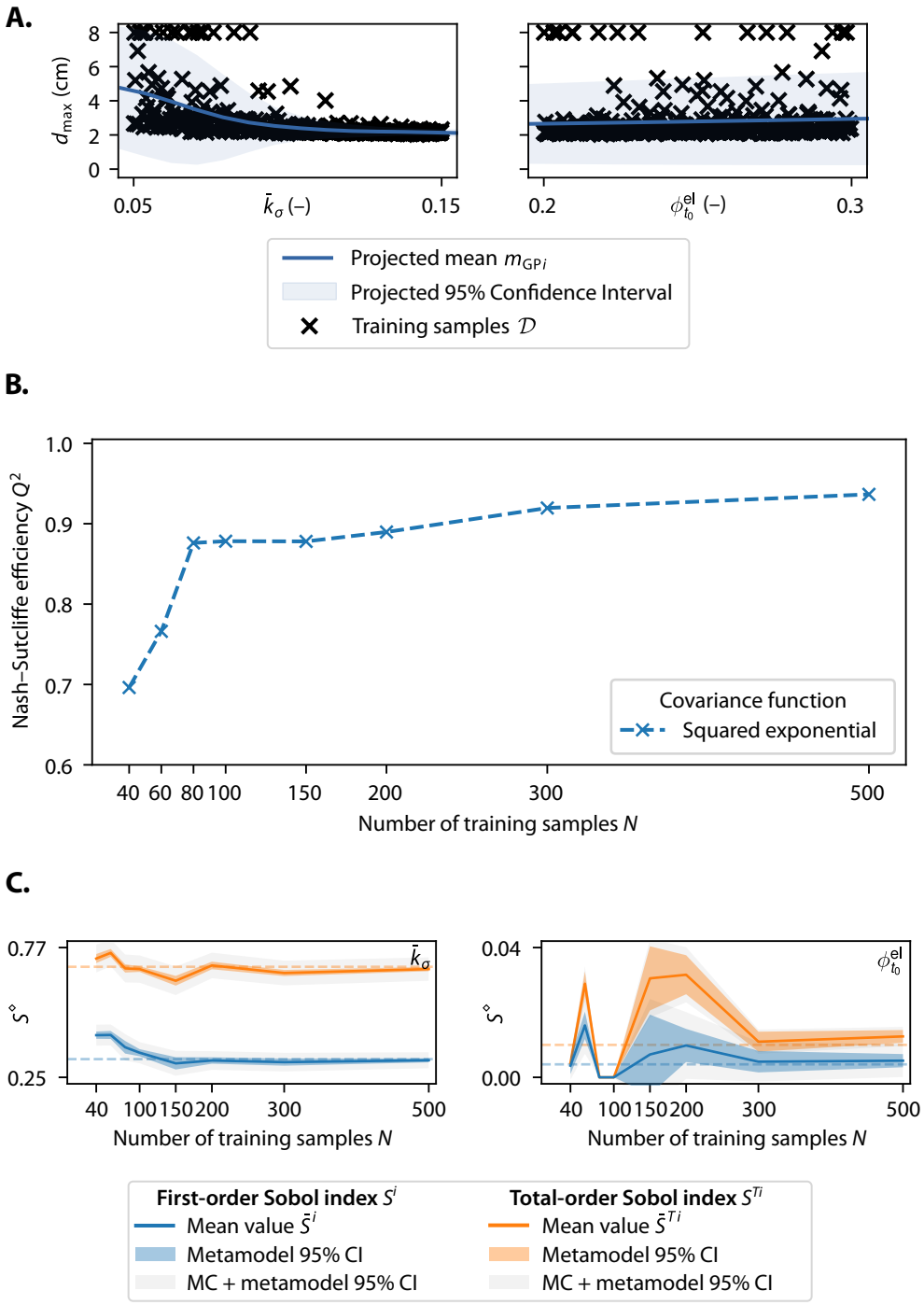


Figure 10. Summary of results for arterial growth and remodelling. A. Projected mean $m_{\mathrm{GP}i}(X_i)$, projected 95% confidence interval (CI) and training samples $\mathcal{D}$ for the diameter d_{max} (the y-axis labels apply to both figures). B. Nash–Sutcliffe efficiency Q^2 for different training sample set sizes. C. First-order and total-order Sobol indices and 95% confidence intervals (CI) for an increasing number of training samples. Monte–Carlo abbreviates as MC.

in Supplement). One further detail should be mentioned as an example: the plot for $\phi_{t_0}^{\mathsf{el}}$ reveals problems in estimating the first and total-order Sobol indices for $N = 80$ or 100. The Sobol indices and the uncertainties are all close to zero. Similar behaviour can be observed for other parameters (see Fig. A.2 in Supplement). For $N < 300$, the GP has not yet converged, and hence does not capture all features of the quantity of interest. Furthermore, we note that the uncertainty related to the metamodel is much higher and in some cases even dominates the total uncertainty in this example, while the uncertainty related to Monte–Carlo integration dominated the previous example. It is therefore important to include the uncertainty related to the metamodel because considering only the uncertainty related to Monte–Carlo integration would underestimate the total uncertainty in the Sobol index estimate.

Finally, the computation of higher-order indices was not feasible with the approach chosen in the original contribution [95]. In contrast, the following will show that the metamodel-based approach enables their computation. As an example, we again consider the gain parameter $\bar{k}_\sigma$: a closer look at Fig. 10C reveals that the total-index is considerably higher than the first-order index: $S^{T\bar{k}_\sigma} - S^{\bar{k}_\sigma} = 0.34$. This delta indicates interactions with other parameters, and thus estimating the second-order indices is of particular interest for this example. Since we see in Fig. 10C that the metamodel contributes significantly to the total uncertainty, we include uncertainty estimates for the metamodel (as opposed to relying solely on the predictive mean, as in the previous second-order estimates). The estimates indeed show interaction with two parameters: the turnover time τ[3]and the constitutive parameter k_2 ($S^{\bar{k}_\sigma \tau} = 0.19$ and $S^{\bar{k}_\sigma k_2} = 0.06$). However, the sum of all second-order indices ($S^{\bar{k}_\sigma j} = 0.25$) still does not cover the delta between the first and total-order index. Hence, we specifically estimate the third-order Sobol index for the three most influential parameters, $\bar{k}_\sigma$, τ, and k_2, resulting in considerable third-order interaction: $S^{\bar{k}_\sigma \tau k_2} = 0.06$. Detailed results for the second and third-order indices, including confidence intervals, are included in the Supplement (see Fig. A.3 in Supplement).

Thus, we are able to identify influential parameters for factor prioritisation based on the metamodel approach with small numbers of training samples, and we can also separate the influential parameters from the non-influential ones. Hence, the metamodel-based approach provides the same results as the approach based directly on the full model in the original publication [95]. The computational cost, i.e., the number of evaluations of the full model, however, is much lower when using a metamodel. Additionally, we can quantify higher-order indices which is infeasible based on evaluations of the full model.

5 Conclusion

Since a global sensitivity analysis is computationally expensive, modellers often rely on local methods alone, which may be inadequate [5]. The use of a metamodel-based approach, however, allows a global variance-based sensitivity analysis to be performed, even for computationally expensive biomechanical models with a moderate number of input space dimensions at a manageable computational cost. The number of training samples required to obtain reliable estimates for the Sobol indices depends largely on the problem set-up itself: our results demonstrate that we can identify the most influential input parameters and separate them from non-influential parameters with small numbers of training samples. However, quantifying the exact value of the Sobol indices requires more training samples. Moreover, the approach is able to quantify the uncertainty related to the metamodel: including this uncertainty is important, because considering only the uncertainty related to Monte–Carlo integration could underestimate the total uncertainty

[3]The turnover time was denoted by T in the original publication [95].

in the Sobol index estimates. The metamodel-based approach also allows an estimation of higher-order Sobol indices, and thus a quantification of interaction effects, which is not feasible without a metamodel due to the computational costs involved. While there is no *one-size-fits-all* rule, the approach is general and efficient enough to allow a study of different aspects of sensitivity analysis, including a transparent declaration of the uncertainties involved in the estimation process.

We demonstrated how a rigorous global sensitivity analysis can be applied to complex, computationally expensive problems. A carefully performed sensitivity analysis is generally an integral part to ensure the high quality of any model development [5]. By demonstrating the workflow and its application for biomechanical problems, we contribute to closing the gap between proposals of new sensitivity analysis methods and application papers [8]. We hereby encourage sensitivity analysis in general and the metamodel-based approach in particular in the biomechanics community. In the big picture of model development, the presented workflow can be a building block towards inverse analysis, or it can be a valuable tool to better understand the model itself.

Acknowledgements

SB wishes to acknowledge funding of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via project 257981274 and 386349077, JN and WAW wish to acknowledge funding of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) via project WA 1521/23, and BAS gratefully acknowledges the support of the Institute for Advanced Study – Technical University of Munich. The original version of the software QUEENS was provided by the courtesy of AdCo Engineering^{GW} GmbH, which is gratefully acknowledged.

References

[1] R. Durrett. "Cancer Modelling: A Personal Perspective". In: *Notices of the AMS* 60.3 (2013), pp. 304–309. DOI: 10.1090/noti953.

[2] R. Durrett. "Branching Process Models of Cancer". In: *Branching Process Models of Cancer*. Ed. by R. Durrett. Mathematical Biosciences Institute Lecture Series. Cham: Springer International Publishing, 2015, pp. 1–63. DOI: 10.1007/978-3-319-16065-8_1.

[3] M. Baar, L. Coquille, H. Mayer, M. Hölzel, M. Rogava, T. Tüting, and A. Bovier. "A Stochastic Model for Immunotherapy of Cancer". In: *Scientific Reports* 6.1 (2016), p. 24169. DOI: 10.1038/srep24169.

[4] G. White and I. Farrance. "Uncertainty of Measurement in Quantitative Medical Testing". In: *The Clinical Biochemist Reviews* 25.4 (2004), S1–S24.

[5] A. Saltelli, K. Aleksankina, W. Becker, P. Fennell, F. Ferretti, N. Holst, S. Li, and Q. Wu. "Why so Many Published Sensitivity Analyses Are False: A Systematic Review of Sensitivity Analysis Practices". In: *Environmental Modelling & Software* 114 (2019), pp. 29–39. DOI: 10.1016/j.envsoft.2019.01.012.

[6] A. Saltelli, ed. *Global Sensitivity Analysis: The Primer*. Chichester, England ; Hoboken, NJ: John Wiley, 2008.

[7] A. Saltelli, ed. *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*. Hoboken, NJ: Wiley, 2004.

[8] S. Razavi et al. "The Future of Sensitivity Analysis: An Essential Discipline for Systems Modeling and Policy Support". In: *Environmental Modelling & Software* 137 (2021), p. 104954. DOI: 10.1016/j.envsoft.2020.104954.

[9] M. D. Morris. "Factorial Sampling Plans for Preliminary Computational Experiments". In: *Technometrics* 33.2 (1991), pp. 161–174. DOI: 10.1080/00401706.1991.10484804.

[10] G. Qian and A. Mahdi. "Sensitivity Analysis Methods in the Biomedical Sciences". In: *Mathematical Biosciences* 323 (2020), p. 108306. DOI: 10.1016/j.mbs.2020.108306.

[11] I. M. Sobol. "Sensitivity Estimates for Nonlinear Mathematical Models". In: *Mathematical modelling and computational experiments* 1.4 (1993), pp. 407–414.

[12] I. M. Sobol. "Global Sensitivity Indices for Nonlinear Mathematical Models and Their Monte Carlo Estimates". In: *Mathematics and Computers in Simulation*. The Second IMACS Seminar on Monte Carlo Methods 55.1 (2001), pp. 271–280. DOI: 10.1016/S0378-4754(00)00270-6.

[13] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts, London, England: The MIT Press, 2006.

[14] L. Le Gratiet, C. Cannamela, and B. Iooss. "A Bayesian Approach for Global Sensitivity Analysis of (Multifidelity) Computer Codes". In: *SIAM/ASA Journal on Uncertainty Quantification* 2.1 (2014), pp. 336–363. DOI: 10.1137/130926869.

[15] O. K. Oyebamiji, D. J. Wilkinson, P. G. Jayathilake, T. P. Curtis, S. P. Rushton, B. Li, and P. Gupta. "Gaussian Process Emulation of an Individual-Based Model Simulation of Microbial Communities". In: *Journal of Computational Science* 22 (2017), pp. 69–84. DOI: 10.1016/j.jocs.2017.08.006.

[16] G. Karagiannis, W. Hao, and G. Lin. "Calibrations and Validations of Biological Models with an Application on the Renal Fibrosis". In: *International Journal for Numerical Methods in Biomedical Engineering* 36.5 (2020), e3329. DOI: 10.1002/cnm.3329.

[17] A. C. Ross, R. G. Najjar, and M. Li. "A Metamodel-Based Analysis of the Sensitivity and Uncertainty of the Response of Chesapeake Bay Salinity and Circulation to Projected Climate Change". In: *Estuaries and Coasts* (2020). DOI: 10.1007/s12237-020-00761-w.

[18] A. Hirvoas, C. Prieur, E. Arnaud, F. Caleyron, and M. M. Zuniga. "Quantification and Reduction of Uncertainties in a Wind Turbine Numerical Model Based on a Global Sensitivity Analysis and a Recursive Bayesian Inference Approach". In: *International Journal for Numerical Methods in Engineering* 122.10 (2021), pp. 2528–2544. DOI: 10.1002/nme.6630.

[19] F. Sahli Costabal, K. Matsuno, J. Yao, P. Perdikaris, and E. Kuhl. "Machine Learning in Drug Development: Characterizing the Effect of 30 Drugs on the QT Interval Using Gaussian Process Regression, Sensitivity Analysis, and Uncertainty Quantification". In: *Computer Methods in Applied Mechanics and Engineering* 348 (2019), pp. 313–333. DOI: 10.1016/j.cma.2019.01.033.

[20] J. Kremheller, A.-T. Vuong, L. Yoshihara, W. A. Wall, and B. A. Schrefler. "A Monolithic Multiphase Porous Medium Framework for (a-)Vascular Tumor Growth". In: *Computer Methods in Applied Mechanics and Engineering* 340 (2018), pp. 657–683. DOI: 10.1016/j.cma.2018.06.009.

[21] J. Kremheller, A.-T. Vuong, B. A. Schrefler, and W. A. Wall. "An Approach for Vascular Tumor Growth Based on a Hybrid Embedded/Homogenized Treatment of the Vasculature within a Multiphase Porous Medium Model". In: *International Journal for Numerical Methods in Biomedical Engineering* 35.11 (2019), e3253. DOI: 10.1002/cnm.3253.

[22] B. Wirthl, J. Kremheller, B. A. Schrefler, and W. A. Wall. "Extension of a Multiphase Tumour Growth Model to Study Nanoparticle Delivery to Solid Tumours". In: *PLOS ONE* 15.2 (2020), e0228443. DOI: 10.1371/journal.pone.0228443.

[23] C. J. Cyron, R. C. Aydin, and J. D. Humphrey. "A Homogenized Constrained Mixture (and Mechanical Analog) Model for Growth and Remodeling of Soft Tissue". In: *Biomechanics and Modeling in Mechanobiology* 15.6 (2016), pp. 1389–1403. DOI: 10.1007/s10237-016-0770-9.

[24] C. J. Cyron and J. D. Humphrey. "Growth and Remodeling of Load-Bearing Biological Soft Tissues". In: *Meccanica* 52.3 (2017), pp. 645–664. DOI: 10.1007/s11012-016-0472-5.

[25] F. A. Braeu. "Three-Dimensional Homogenized Constrained Mixture Model of Anisotropic Vascular Growth and Remodeling". PhD thesis. Munich: Technische Universität München, 2019.

[26] I. M. Sobol'. "On Sensitivity Estimation for Nonlinear Mathematical Models". In: *Matematicheskoe modelirovanie* 2.1 (1990), pp. 112–118.

[27] J. H. A. Guillaume et al. "Introductory Overview of Identifiability Analysis: A Guide to Evaluating Whether You Have the Right Type of Data for Your Modeling Purpose". In: *Environmental Modelling & Software* 119 (2019), pp. 418–432. DOI: 10.1016/j.envsoft.2019.07.007.

[28] G. E. P. Box and R. D. Meyer. "An Analysis for Unreplicated Fractional Factorials". In: *Technometrics* 28.1 (1986), pp. 11–18. DOI: 10.1080/00401706.1986.10488093.

[29] S. Asmussen and P. W. Glynn. *Stochastic Simulation: Algorithms and Analysis*. New York: Springer, 2007.

[30] T. Ishigami and T. Homma. "An Importance Quantification Technique in Uncertainty Analysis for Computer Models". In: *Proceedings. First International Symposium on Uncertainty Modeling and Analysis*. 1990, pp. 398–403. DOI: 10.1109/ISUMA.1990.151285.

[31] W. Jarosz. "Efficient Monte Carlo Methods for Light Transport in Scattering Media". PhD thesis. San Diego: University of California, 2008.

[32] A. Saltelli. "Making Best Use of Model Evaluations to Compute Sensitivity Indices". In: *Computer Physics Communications* 145.2 (2002), pp. 280–297. DOI: 10.1016/S0010-4655(02)00280-1.

[33] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola. "Variance Based Sensitivity Analysis of Model Output. Design and Estimator for the Total Sensitivity Index". In: *Computer Physics Communications* 181.2 (2010), pp. 259–270. DOI: 10.1016/j.cpc.2009.09.018.

[34] M. J. W. Jansen. "Analysis of Variance Designs for Model Output". In: *Computer Physics Communications* 117.1-2 (1999), pp. 35–43. DOI: 10.1016/S0010-4655(98)00154-4.

[35] I. M. Sobol', S. Tarantola, D. Gatelli, S. S. Kucherenko, and W. Mauntz. "Estimating the Approximation Error When Fixing Unessential Factors in Global Sensitivity Analysis". In: *Reliability Engineering & System Safety* 92.7 (2007), pp. 957–960. DOI: 10.1016/j.ress.2006.07.001.

[36] A. Janon, T. Klein, A. Lagnoux, M. Nodet, and C. Prieur. "Asymptotic Normality and Efficiency of Two Sobol Index Estimators". In: *ESAIM: Probability and Statistics* 18 (2014), pp. 342–364. DOI: 10.1051/ps/2013040.

[37] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, 1992. DOI: 10.1137/1.9781611970081.

[38] B. Iooss, L. Boussouf, V. Feuillard, and A. Marrel. "Numerical Studies of the Metamodel Fitting and Validation Processes". In: *preprint arXiv:1001.1049* (2010).

[39] K. Cheng, Z. Lu, C. Ling, and S. Zhou. "Surrogate-Assisted Global Sensitivity Analysis: An Overview". In: *Structural and Multidisciplinary Optimization* 61.3 (2020), pp. 1187–1213. DOI: 10.1007/s00158-019-02413-5.

[40] I. Sobol'. "On the Distribution of Points in a Cube and the Approximate Evaluation of Integrals". In: *USSR Computational Mathematics and Mathematical Physics* 7.4 (1967), pp. 86–112. DOI: 10.1016/0041-5553(67)90144-9.

[41] M. McKay, R. Beckman, and W. Conover. "A Comparison of Three Methods for Selecting Vales of Input Variables in the Analysis of Output From a Computer Code". In: *Technometrics* 21.2 (1979), pp. 239–245. DOI: 10.1080/00401706.1979.10489755.

[42] F. Xiong, Y. Xiong, W. Chen, and S. Yang. "Optimizing Latin Hypercube Design for Sequential Sampling of Computer Experiments". In: *Engineering Optimization* 41.8 (2009), pp. 793–810. DOI: 10.1080/03052150902852999.

[43] X. Zhou, D. K. J. Lin, X. Hu, and L. Ouyang. "Sequential Latin Hypercube Design with Both Space-Filling and Projective Properties". In: *Quality and Reliability Engineering International* 35.6 (2019), pp. 1941–1951. DOI: 10.1002/qre.2485.

[44] S. Kucherenko, D. Albrecht, and A. Saltelli. "Exploring Multi-Dimensional Spaces: A Comparison of Latin Hypercube and Quasi Monte Carlo Sampling Techniques". In: *preprint arXiv:1505.02350* (2015).

[45] J. E. Oakley and A. O'Hagan. "Probabilistic Sensitivity Analysis of Complex Models: A Bayesian Approach". In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 66.3 (2004), pp. 751–769. DOI: 10.1111/j.1467-9868.2004.05304.x.

[46] A. Marrel, B. Iooss, B. Laurent, and O. Roustant. "Calculations of Sobol Indices for the Gaussian Process Metamodel". In: *Reliability Engineering & System Safety* 94.3 (2009), pp. 742–751. DOI: 10.1016/j.ress.2008.07.008.

[47] D. K. Duvenaud. "Automatic Model Construction with Gaussian Processes". Doctor Thesis. Cambridge, England: University of Cambridge, 2014.

[48] N. D. Goodman, T. D. Ullman, and J. B. Tenenbaum. "Learning a Theory of Causality". In: *Psychological Review* 118.1 (2011), pp. 110–119. DOI: 10.1037/a0021336.

[49] A. Pepelyshev. "The Role of the Nugget Term in the Gaussian Process Method". In: *mODa 9 – Advances in Model-Oriented Design and Analysis*. Ed. by A. Giovagnoli, A. C. Atkinson, B. Torsney, and C. May. Contributions to Statistics. Heidelberg: Physica-Verlag HD, 2010, pp. 149–156. DOI: 10.1007/978-3-7908-2410-0_20.

[50] I. Andrianakis and P. G. Challenor. "The Effect of the Nugget on Gaussian Process Emulators of Computer Models". In: *Computational Statistics & Data Analysis* 56.12 (2012), pp. 4215–4228. DOI: 10.1016/j.csda.2012.04.020.

[51] Y. Bengio, O. Delalleau, and N. Roux. "The Curse of Highly Variable Functions for Local Kernel Machines". In: *Advances in Neural Information Processing Systems*. Vol. 18. MIT Press, 2006.

[52] R. Tripathy, I. Bilionis, and M. Gonzalez. "Gaussian Processes with Built-in Dimensionality Reduction: Applications to High-Dimensional Uncertainty Propagation". In: *Journal of Computational Physics* 321 (2016), pp. 191–223. DOI: 10.1016/j.jcp.2016.05.039.

[53] H. Liu, Y.-S. Ong, X. Shen, and J. Cai. "When Gaussian Process Meets Big Data: A Review of Scalable GPs". In: *IEEE Transactions on Neural Networks and Learning Systems* 31.11 (2020), pp. 4405–4423. DOI: 10.1109/TNNLS.2019.2957109.

[54] J. Hensman, N. Fusi, and N. D. Lawrence. "Gaussian Processes for Big Data". In: *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*. UAI'13. Arlington, Virginia, USA: AUAI Press, 2013, pp. 282–290.

[55] B. Efron. "Bootstrap Methods: Another Look at the Jackknife". In: *The Annals of Statistics* 7.1 (1979), pp. 1–26. DOI: 10.1214/aos/1176344552.

[56] A. Janon, M. Nodet, and C. Prieur. "Uncertainties Assessment in Global Sensitivity Indices Estimation from Metamodels". In: *International Journal for Uncertainty Quantification* 4.1 (2014). DOI: 10.1615/Int.J.UncertaintyQuantification.2012004291.

[57] I. Panin. "Risk of Estimators for Sobol' Sensitivity Indices Based on Metamodels". In: *Electronic Journal of Statistics* 15.1 (2021), pp. 235–281. DOI: 10.1214/20-EJS1793.

[58] G. E. B. Archer, A. Saltelli, and I. M. Sobol. "Sensitivity Measures, Anova-like Techniques and the Use of Bootstrap". In: *Journal of Statistical Computation and Simulation* 58.2 (1997), pp. 99–120. DOI: 10.1080/00949659708811825.

[59] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap*. 1. Edition. New York: Chapman and Hall/CRC, 1993.

[60] G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: With Applications in R*. Springer Texts in Statistics. New York: Springer-Verlag, 2013. DOI: 10.1007/978-1-4614-7138-7.

[61] G. Sciumè, S. Shelton, W. G. Gray, C. T. Miller, F. Hussain, M. Ferrari, P. Decuzzi, and B. A. Schrefler. "A Multiphase Model for Three-Dimensional Tumor Growth". In: *New Journal of Physics* 15 (2013). DOI: 10.1088/1367-2630/15/1/015005.

[62] G. Sciumè, W. G. Gray, F. Hussain, M. Ferrari, P. Decuzzi, and B. A. Schrefler. "Three Phase Flow Dynamics in Tumor Growth". In: *Computational Mechanics* 53.3 (2014), pp. 465–484. DOI: 10.1007/s00466-013-0956-2.

[63] G. Sciumè, R. Santagiuliana, M. Ferrari, P. Decuzzi, and B. A. Schrefler. "A Tumor Growth Model with Deformable ECM". In: *Physical Biology* 11.6 (2014). DOI: 10.1088/1478-3975/11/6/065004.

[64] W. G. Gray and C. T. Miller. *Introduction to the Thermodynamically Constrained Averaging Theory for Porous Medium Systems*. Springer, 2014.

[65] C. T. Miller, W. G. Gray, and B. A. Schrefler. "A Continuum Mechanical Framework for Modeling Tumor Growth and Treatment in Two- and Three-Phase Systems". In: *Archive of Applied Mechanics* (2021). DOI: 10.1007/s00419-021-01891-8.

[66] A. Saltelli et al. "Five Ways to Ensure That Models Serve Society: A Manifesto". In: *Nature* 582.7813 (2020), pp. 482–484. DOI: 10.1038/d41586-020-01812-9.

[67] S. Wilhelm, A. J. Tavares, Q. Dai, S. Ohta, J. Audet, H. F. Dvorak, and W. C. W. Chan. "Analysis of Nanoparticle Delivery to Tumours". In: *Nature Reviews Materials* 1.5 (2016), p. 16014. DOI: 10.1038/natrevmats.2016.14.

[68] S. M. Moghimi and D. Simberg. "Nanoparticle Transport Pathways into Tumors". In: *Journal of Nanoparticle Research* 20.6 (2018), p. 169. DOI: 10.1007/s11051-018-4273-8.

[69] R. K. Jain. "Transport of Molecules across Tumor Vasculature". In: *Cancer and metastasis review* 6.4 (1987), pp. 559–593. DOI: 10.1007/BF00047468.

[70] T. Stylianopoulos and R. K. Jain. "Combining Two Strategies to Improve Perfusion and Drug Delivery in Solid Tumors." In: *Proceedings of the National Academy of Sciences of the United States of America* 110.46 (2013), pp. 18632–7. DOI: 10.1073/pnas.1318415110.

[71] S. Cui. "Formation of Necrotic Cores in the Growth of Tumors: Analytic Results". In: *Acta Mathematica Scientia* 26.4 (2006), pp. 781–796. DOI: 10.1016/S0252-9602(06)60104-5.

[72] P. Macklin, S. McDougall, A. R. A. Anderson, M. A. J. Chaplain, V. Cristini, and J. Lowengrub. "Multiscale Modelling and Nonlinear Simulation of Vascular Tumour Growth". In: *Journal of Mathematical Biology* 58.4 (2009), pp. 765–798. DOI: 10.1007/s00285-008-0216-9.

[73] M. W. Dewhirst and T. W. Secomb. "Transport of Drugs from Blood Vessels to Tumour Tissue". In: *Nature Reviews Cancer* 17 (2017), p. 738. DOI: 10.1038/nrc.2017.93.

[74] T. Stylianopoulos et al. "Causes, Consequences, and Remedies for Growth-Induced Solid Stress in Murine and Human Tumors". In: *Proceedings of the National Academy of Sciences* 109.38 (2012), pp. 15101–15108. DOI: 10.1073/pnas.1213353109.

[75] G. S. Terentyuk, G. N. Maslyakova, L. V. Suleymanova, N. G. Khlebtsov, B. N. Khlebtsov, G. G. Akchurin, I. L. Maksimova, and V. V. Tuchin. "Laser-Induced Tissue Hyperthermia Mediated by Gold Nanoparticles: Toward Cancer Phototherapy". In: *Journal of Biomedical Optics* 14.2 (2009), p. 021016. DOI: 10.1117/1.3122371.

[76] M. Nabil, P. Decuzzi, and P. Zunino. "Modelling Mass and Heat Transfer in Nano-Based Cancer Hyperthermia." In: *Royal Society open science* 2.10 (2015), p. 150447. DOI: 10.1098/rsos.150447.

[77] C. E. Shannon. "A Mathematical Theory of Communication". In: *The Bell system technical journal* 27.3 (1948), pp. 379–423.

[78] S. Kullback. *Information Theory and Statistics*. Courier Corporation, 1997.

[79] Y. T. Ho, G. Adriani, S. Beyer, P.-T. Nhan, R. D. Kamm, and J. C. Y. Kah. "A Facile Method to Probe the Vascular Permeability of Nanoparticles in Nanomedicine Applications". In: *Scientific Reports* 7.1 (2017), p. 707. DOI: 10.1038/s41598-017-00750-3.

[80] M. R. Dreher, W. Liu, C. R. Michelich, M. W. Dewhirst, F. Yuan, and A. Chilkoti. "Tumor Vascular Permeability, Accumulation, and Penetration of Macromolecular Drug Carriers". In: *JNCI: Journal of the National Cancer Institute* 98.5 (2006), pp. 335–344. DOI: 10.1093/jnci/djj070.

[81] C.-Y. Chou, C.-K. Huang, K.-W. Lu, T.-L. Horng, and W.-L. Lin. "Investigation of the Spatiotemporal Responses of Nanoparticles in Tumor Tissues with a Small-Scale Mathematical Model". In: *PLoS ONE* 8.4 (2013). Ed. by J. S. Burns, e59135. DOI: 10.1371/journal.pone.0059135.

[82] BACI. *A Comprehensive Multi-Physics Simulation Framework*. https://baci.pages.gitlab.lrz.de/website/. 2021.

[83] J. Biehler, J. Nitzler, S. Brandstaeter, W. A. Wall, and V. Gravemeier. *QUEENS – a Software Platform for Uncertainty Quantification, Physics-Informed Machine Learning, Bayesian Optimization, Inverse Problems and Simulation Analytics: User Guide*. AdCo Engineering GW GmbH. 2020.

[84] GPy. *GPy: A Gaussian Process Framework in Python*. since 2012.

[85] A. Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035.

[86] B. Matern. *Spatial Variation*. Second. Lecture Notes in Statistics. New York: Springer-Verlag, 1986. DOI: 10.1007/978-1-4615-7892-5.

[87] M. L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer Series in Statistics. New York: Springer-Verlag, 1999. DOI: 10.1007/978-1-4612-1494-6.

[88] J. E. Nash and J. V. Sutcliffe. "River Flow Forecasting through Conceptual Models Part I — A Discussion of Principles". In: *Journal of Hydrology* 10.3 (1970), pp. 282–290. DOI: 10.1016/0022-1694(70)90255-6.

[89] T. Van Steenkiste, J. van der Herten, I. Couckuyt, and T. Dhaene. "Sequential Sensitivity Analysis of Expensive Black-Box Simulators with Metamodelling". In: *Applied Mathematical Modelling* 61 (2018), pp. 668–681. DOI: 10.1016/j.apm.2018.05.023.

[90] C. A. Micchelli, Y. Xu, and H. Zhang. "Universal Kernels". In: *The Journal of Machine Learning Research* 7 (2006), pp. 2651–2667.

[91] D. P. Kingma and J. Ba. "Adam: A Method for Stochastic Optimization". In: *preprint arXiv:1412.6980* (2017).

[92] M. Fritz, E. A. B. F. Lima, J. T. Oden, and B. Wohlmuth. "On the Unsteady Darcy–Forchheimer–Brinkman Equation in Local and Nonlocal Tumor Growth Models". In: *Mathematical Models and Methods in Applied Sciences* (2019). DOI: 10.1142/S0218202519500325.

[93] C. M. Phillips, E. A. B. F. Lima, R. T. Woodall, A. Brock, and T. E. Yankeelov. "A Hybrid Model of Tumor Growth and Angiogenesis: In Silico Experiments". In: *PLOS ONE* 15.4 (2020), e0231137. DOI: 10.1371/journal.pone.0231137.

[94] F. Sarrazin, F. Pianosi, and T. Wagener. "Global Sensitivity Analysis of Environmental Models: Convergence and Validation". In: *Environmental Modelling & Software* 79 (2016), pp. 135–152. DOI: 10.1016/j.envsoft.2016.02.005.

[95] S. Brandstaeter, S. L. Fuchs, J. Biehler, R. C. Aydin, W. A. Wall, and C. J. Cyron. "Global Sensitivity Analysis of a Homogenized Constrained Mixture Model of Arterial Growth and Remodeling". In: *Journal of Elasticity* (2021). DOI: 10.1007/s10659-021-09833-9.