Journal Pre-proof

UncertainSCI: Uncertainty quantification for computational models in biomedicine and bioengineering

Akil Narayan, Zexin Liu, Jake A. Bergquist, Chantel Charlebois, Sumientra Rampersad, Lindsay Rupp, Dana Brooks, Dan White, Jess Tate, Rob S. MacLeod

PII: S0010-4825(22)01115-5
DOI: https://doi.org/10.1016/j.compbiomed.2022.106407
Reference: CBM 106407

To appear in: *Computers in Biology and Medicine*

Received date : 16 August 2022
Revised date : 7 November 2022
Accepted date : 3 December 2022

Please cite this article as: A. Narayan, Z. Liu, J.A. Bergquist et al., UncertainSCI: Uncertainty quantification for computational models in biomedicine and bioengineering, *Computers in Biology and Medicine* (2022), doi: https://doi.org/10.1016/j.compbiomed.2022.106407.

UncertainSCI: Uncertainty quantification for computational models in biomedicine and bioengineering

Akil Narayan[a,c,*,1], Zexin Liu[a,c], Jake A. Bergquist[a,b], Chantel Charlebois[a,b], Sumientra Rampersad[d,e], Lindsay Rupp[a,b], Dana Brooks[e], Dan White[a,b], Jess Tate[a,b] and Rob S. MacLeod[a,b]

[a]*Scientific Computing and Imaging Institute, University of Utah, 72 Central Campus Dr, Salt Lake City, Utah, 84112, United States*
[b]*Department of Biomedical Engineering, University of Utah, 72 Central Campus Dr, Salt Lake City, Utah, 84112, United States*
[c]*Department of Mathematics, University of Utah, 72 Central Campus Dr, Salt Lake City, Utah, 84112, United States*
[d]*Department of Physics, University of Massachusetts Boston, Boston, Massachusetts, USA*
[e]*Department of Electrical and Computer Engineering, Northeastern University, Boston, Massachusetts, USA*

ARTICLE INFO

Keywords:
Biomedical simulations,
uncertainty quantification
open-source software

ABSTRACT

Background Computational biomedical simulations frequently contain parameters that model physical features, material coefficients, and physiological effects, whose values are typically assumed known *a priori*. Understanding the effect of variability in those assumed values is currently a topic of great interest. A general-purpose software tool that quantifies how variation in these parameters affects model outputs is not broadly available in biomedicine. For this reason, we developed the 'UncertainSCI' uncertainty quantification software suite to facilitate analysis of uncertainty due to parametric variability.

Methods We developed and distributed a new open-source Python-based software tool, UncertainSCI, which employs advanced parameter sampling techniques to build polynomial chaos (PC) emulators that can be used to predict model outputs for general parameter values. Uncertainty of model outputs is studied by modeling parameters as random variables, and model output statistics and sensitivities are then easily computed from the emulator. Our approaches utilize modern, near-optimal techniques for sampling and PC construction based on weighted Fekete points constructed by subsampling from a suitably randomized candidate set.

Results Concentrating on two test cases—modeling bioelectric potentials in the heart and electric stimulation in the brain—we illustrate the use of UncertainSCI to estimate variability, statistics, and sensitivities associated with multiple parameters in these models.

Conclusions UncertainSCI is a powerful yet lightweight tool enabling sophisticated probing of parametric variability and uncertainty in biomedical simulations. Its non-intrusive pipeline allows users to leverage existing software libraries and suites to accurately ascertain parametric uncertainty in a variety of applications.

Keywords: Biomedical simulations, uncertainty quantification, open-source software

1. Introduction

Computer simulations have become an invaluable tool for investigating and predicting biological function in biomedicine, and are useful in designing and assessing treatments. Typical computer simulations or *forward models* in biomedicine contain numerous parameters, possibly hand-tuned or optimized by comparing to experimental data, that influence the outcome of the simulation. For example, the width and conductivity of the cerebrospinal fluid layer surrounding the brain have direct impact on the predicted electric fields in simulations of neuromodulation [20, 21]. It is rare that these parameters have well-established, fixed values, especially because such parameters typically vary among individuals in a patient cohort, and within individuals over time. The possible parameter variations, therefore, limit the predictive

Email address: akil@sci.utah.edu (A. Narayan)
URL: http://www.sci.utah.edu/~akil (A. Narayan)
ORCID(s): 0000-0002-5914-4207 (A. Narayan); 0000-0003-3409-5709 (Z. Liu); 0000-0002-4586-6911 (J.A. Bergquist); 0000-0002-4139-3539 (C. Charlebois); 0000-0001-9860-4459 (S. Rampersad); 0000-0002-2688-7688 (L. Rupp); 0000-0003-3231-6715 (D. Brooks); 0000-0002-9270-4819 (D. White); 0000-0002-2934-1453 (J. Tate); 0000-0002-0000-0356 (R.S. MacLeod)

power and efficacy of forward models only tuned with nominal values, resulting in the need for quantification of the effect on model outputs of parameter variability for increased model understanding and impact.

In the field of uncertainty quantification (UQ), techniques for "forward" propagation of uncertainty provide quantitative evaluations of the effects of parametric variability on forward models. (The field of UQ is quite broad, including stochastic inverse and inference problems, decision theory, design and optimization, and model selection, but the focus in this article is on forward UQ.) In our particular setting of *parametric* UQ, input parameters to a simulation are modeled as random variables endowed with a probability distribution that reflects previous experience about the variation of the parameter. The randomness due to these parameters is pushed forward to a model output, which itself becomes random. The resulting probability distribution of the output becomes the target of interest, and computational techniques in UQ can be used to approximate statistics of this output distribution. In this way, means and standard deviations can be computed, in addition to more sophisticated descriptors such as quantiles, sensitivities, and higher-order statistics such as skewness and kurtosis. In particular, see Table 2 for more precise descriptions of some of these computable statistics.

Non-intrusive UQ strategies are preferred in practice since they require little or no changes to existing simulation pipelines, and instead utilize several evaluations of a forward simulation to characterize the distribution of the model output. The most direct approach is a simple Monte Carlo (MC) method [27, 6, 49]. However, MC approaches have limitations that more sophisticated UQ can address. In particular, when dependence on parameters is smooth (which is frequently the case in biomedical simulations), non-intrusive polynomial chaos (PC) techniques can provide very accurate estimates for model output statistics with orders-of-magnitude fewer forward model evaluations compared to an MC approach. For this reason, PC approaches are promising techniques for the biomedical community. Of particular importance is the development of open-source software, which makes UQ capabilities accessible to a diverse range of scientists, and enables extension and modification by technically savvy users.

However, general-purpose and open-source tools using modern UQ techniques are lacking in some respects. Existing UQ toolboxes [30, 44, 14, 36, 26, 1, 33, 12, 11, 32, 40, 39, 41] do not leverage recent advances in data science for optimally building UQ emulators, and are not closely integrated with existing biomedical software.

One particular existing opportunity for community benefit is through the exercise of near-optimal least squares approximation methods that employ randomized subsampling techniques. Such procedures are typically called *leverage score sketching* for large least squares problems in the numerical linear algebra and theoretical computer science communities [24, 48], and *optimal* or *induced* sampling in the approximation theory community [10, 25]. The implementation of such procedures in an open-source software package is one achievement of work in this manuscript.

We have developed a Python-based software suite, UncertainSCI [30], which utilizes modern PC techniques to accomplish common forward-model UQ tasks in biomedical settings. UncertainSCI non-intrusively interfaces with existing software that evaluates a forward model, repeatedly queries this forward model over a parameter ensemble, and processes this gathered data to produce a parameter-to-model-output emulator. This computationally efficient emulator can subsequently be used to replace the (typically expensive) forward model, or can be leveraged within UncertainSCI to compute statistics (means, moments, sensitivities, quantiles) of the probability distribution of the model output. UncertainSCI allows the parameter random variables to be endowed with various types of distributions. While the previous qualities are shared by many existing UQ toolboxes, the unique technology employed by UncertainSCI involves recent advances in high-dimensional approximation that ensures the construction of near-optimal emulators for general polynomial spaces in evaluating uncertainty. We demonstrate UncertainSCI's ability to closely integrate with existing biomedical software and produce meaningful forward UQ analysis results through two cardiac applications and two neural applications. We show that statistics and sensitivities can be easily computed and provide informative estimations of uncertainty. By releasing UncertainSCI as an open-source package we are leveraging both the software engineering advantages of open-source code (e.g., the need for proper documentation and interpretable code), as well as engineering application advantages of customizable prototypes (e.g., the ability to easily add new features or modify existing ones to meet a user's specific needs.)

1.1. Comparison to alternative UQ software

UncertainSCI is not the only toolbox for non-intrusively investigating uncertainty in forward models. Numerous other toolboxes accomplish similar goals. Table 1 summarizes capabilities of several other software toolboxes for forward UQ using PC methods, and showcases some advantages of UncertainSCI in the context of those other toolboxes. Table 1 is *not* meant as a comprehensive comparison, and should be used mainly to highlight the strengths of UncertainSCI in terms of weighted-maximum, volume-based sampling and mean best-approximation guarantees, which are

Table 1
Software capabilities for PC-based forward uncertainty quantification. The features below are chosen to highlight the strengths of UncertainSCI. Many software packages have significant capabilities that UncertainSCI does not, e.g., methods for dependent random variables, inverse and inference problems, Gaussian process models, sparse grids, etc. For a more precise meaning of "Flexible polynomial spaces", see section 2.3. For "Weighted max volume sampling", see section 2.4.2. For "Mean best-approximation guarantees", see section 2.4.1.

	UncertainSCI	UncertainPy	ChaosPy	SimNIBS	UQLab	PyApprox	DAKOTA	Sparse Grids Matlab	UQTk	MUQ	Tasmanian
	[30]	[44]	[14]	[36]	[26]	[18]	[1]	[33]	[12, 11]	[32]	[40, 39, 41]
Open-source	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
First-/second-order statistics	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗
Sensitivity analysis	✗	✗	✗		✗	✗	✗	✗	✗	✗	✗
Medians, quantiles	✗		✗	✗	✗	✗	✗	✗	✗	✗	
General scalar distributions	✗		✗		✗	✗	✗	✗		✗	✗
Flexible polynomial spaces	✗		✗		✗	✗	✗	✗		✗	✗
Tensor-product sampling	✗		✗		✗	✗	✗	✗	✗	✗	✗
Weighted max-volume sampling	✗										
Mean best-approximation guarantees	✗										

capabilities that do not exist in alternative toolboxes.

Like all software packages, UncertainSCI is not comprehensive. Some forward UQ capabilities *not* present in UncertainSCI that are available in some other software packages are sparse grid approximations and low discrepancy (quasi-Monte Carlo) sequences. UncertainSCI does not currently address inverse problems in UQ, which are problems of substantial modern interest. In contrast, some of the software packages in Table 1 do have capabilities for inverse problems. However, UncertainSCI provides the feature of best approximation guarantees that is unique among competitors.

2. Methods

Forward models in biomedical applications often involve *aleatoric parametric* uncertainty, *i.e.,* that which is characterized typically by d parameters $\boldsymbol{p} \in \mathbb{R}^d$ with an *a priori* known probability distribution μ. The forward model is a parameter-to-output map u that takes as input a parameter value $\boldsymbol{p}$, and returns a model output $u(\boldsymbol{p})$, which typically consists of field values in a potentially high-dimensional space $\mathbb{R}^q$. Realistic examples of both the parametric encoding $\boldsymbol{p}$ and the map u are as follows:

- Shape variability appears in electrocardiographic imaging (ECGI), which entails estimating cardiac potentials u from measured body surface potentials [43]. With an appropriate parametric model for shape variability, which is due to the imaging and segmentation pipeline, values of $\boldsymbol{p}$ specify cardiac geometry. The output size q is the number of discrete points on the cardiac surface.
- Uncertainty in parameter values that govern physiological processes is present whenever patient-specific measurements of such processes are effectively impossible. Forward simulations of transcranial current stimulation seek to predict the electric field u in a target area in the brain due to a transcranially applied electric current [34]. The prediction of the field is parameterized by $\boldsymbol{p}$, and embeds uncertainty due to conductivities of tissues such as cerebrospinal fluid. Here, q is the number of discrete points in the target area.

In summary, deterministic simulations of u at fixed nominal values of $\boldsymbol{p}$ fail to capture variations due to uncertainty in $\boldsymbol{p}$, and, even if the nominal value accurately captures the average values of the parameters, they may not represent mean output behavior (the mean of the output need not be the output at the mean).

We describe in what follows the methodology that UncertainSCI uses to quantify forward uncertainty, *non-intrusive polynomial chaos*. Many characteristics of this approach are standard in forward UQ pipelines, and we refer to more

comprehensive references for a detailed discussion of the approach [49, 37, 42]. We focus below on technical capabilities in UncertainSCI that are notable and novel.

2.1. Input distribution for $\boldsymbol{p}$

The d-dimensional random vector $\boldsymbol{p}$ is assumed to have independent components, *i.e.,* the joint probability density w has the product form,

$$w(\boldsymbol{p}) = \prod_{j=1}^{d} w_j(p_j), \qquad \boldsymbol{p} = (p_1, \dots, p_d), \tag{1}$$

with each w_j a one-dimensional probability density. UncertainSCI currently assumes this componentwise independence amongst the parameters, and for each w_j allows various families of parametric distributions (*e.g.,* Normal, Beta, Exponential, *etc.*) and nonparametric distributions (*e.g.,* empirical distributions). One exception to the independence assumption is that correlated Normal distributions are allowed via a whitening transform.

The more general case when $\boldsymbol{p}$ has dependent components is becoming of practical interest and some of the alternative toolboxes discussed in Table 1 allow numerical treatment of dependent variables. However, a numerically tractable and principled approach for general dependent parameter cases is currently lacking, and is the subject of active research [19].

2.2. PC approaches

To capture and quantitatively compute metrics characterizing this uncertainty, UncertainSCI utilizes polynomial chaos (PC) methods. PC methods build an emulator u_N to u of the form,

$$u(\boldsymbol{p}) \approx u_N(\boldsymbol{p}) = \sum_{n=1}^{N} \widehat{u}_n \phi_n(\boldsymbol{p}), \qquad u_N \in P_N := \mathrm{span}_{n=1,\dots,N}\{\phi_n\}, \tag{2}$$

where $\widehat{u}_n$ are u-valued coefficients that must be computed from data, and $\{\phi_n(\boldsymbol{p})\}$ are d-variate polynomials in $\boldsymbol{p}$. The polynomial functions ϕ_n are fixed, and are typically chosen as orthogonal polynomials with respect to the density w. we give more details in section 2.3. Once u_N is constructed, then evaluation and manipulation of u_N in (2) is very computationally efficient, and can easily yield output statistics or quantities of interest; see section 2.5 for a more detailed discussion of the outputs of UncertainSCI.

UncertainSCI requires as input model responses $u(\boldsymbol{p}_m)$ over a particular ensemble $\{\boldsymbol{p}_m\}_{m=1}^{M}$ and uses this to compute the coefficients $\widehat{u}_n$. This step typically forms the bulk of the computational time for UQ analysis, as collecting the model data can require $M \gg 1$ solutions from a computationally intensive model u. We describe in the coming sections how the ϕ_n functions are chosen, how the model response data is collected, and how the resulting coefficients $\widehat{u}_n$ are computed.

2.3. Setting the polynomials ϕ_n

UncertainSCI uses orthogonal polynomials as the basis functions ϕ_n, which is a common choice in PC approaches. In particular, we fix variable $j = 1, \dots, d$, and let $\psi_k^{(j)}(p_j)$ be the degree-k polynomial in p_j that is orthonormal under the marginal density w_j of p_j in (1):

$$\deg \psi_k^{(j)} = k, \qquad \mathbb{E}\left[\psi_k^{(j)}(p_j)\psi_\ell^{(j)}(p_j)\right] = \int_{\mathbb{R}} \psi_k^{(j)}(q_j)\psi_\ell^{(j)}(q_j) w_j(q_j) dq_j = \delta_{k,\ell}, \tag{3}$$

for $k, \ell = 0, 1, \dots$ and where $\delta_{k,j}$ is the Kronecker delta. When w_j is a "standard" distribution, then the $\psi_k^{(j)}$ are (normalized) versions of classical orthogonal polynomials: Legendre polynomials if w_j is uniform, Hermite polynomials if w_j is a Gaussian density, *etc.* [50].

The basis functions used in UncertainSCI are products of these single-variable orthogonal polynomials. Let $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d$ ($\mathbb{N}_0 = \{0, 1, \dots, \}$) be a d-dimensional multi-index, *i.e.,* a point on the non-negative integer lattice in d dimensions. Then a "degree"-$\boldsymbol{\alpha}$ polynomial can be defined as,

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{p}) := \prod_{j=1}^{d} \psi_{\alpha_j}^{(j)}(p_j), \tag{4}$$

and we have that the polynomials $\{\psi_{\boldsymbol{\alpha}}\}_{\boldsymbol{\alpha}}$ are orthonormal with respect to the joint density w of $\boldsymbol{p}$.

The polynomial functions ϕ_n in (2) are an(y) enumeration of the polynomials $\psi_{\boldsymbol{\alpha}}$ for $\boldsymbol{\alpha}$ belonging to a size-N set of multi-indices $\boldsymbol{J}$. Thus,

$$\{\phi_n(\boldsymbol{p})\}_{n=1}^N = \{\psi_{\boldsymbol{\alpha}}(\boldsymbol{p})\}_{\boldsymbol{\alpha}\in\boldsymbol{J}}. \qquad\qquad P_N = \operatorname{span}\left\{\psi_{\boldsymbol{\alpha}} \;\middle|\; \boldsymbol{\alpha}\in\boldsymbol{J}\right\} \tag{5}$$

In more detail, a multi-index set $\boldsymbol{J}$ is introduced, which is a size-N subset of the non-negative integer lattice $\mathbb{N}_0^d$. In UncertainSCI, this set $\boldsymbol{J}$ in general can be defined as a $\boldsymbol{v}$-anisotropic ball of ℓ^p-radius k, that is,

$$\boldsymbol{J} = \boldsymbol{J}(\boldsymbol{v},p,k) = \left\{\boldsymbol{\alpha}\in\mathbb{N}_0^d \;\middle|\; \|\boldsymbol{\alpha}\|_{\boldsymbol{v},p}\le k\right\}, \qquad\qquad \|\boldsymbol{\alpha}\|_{\boldsymbol{v},p}^p := \sum_{j=1}^d \left|\frac{\alpha_j}{v_j}\right|^p, \tag{6}$$

where $\boldsymbol{v}$ is a given d-dimensional vector (with $v_j\neq 0$ for all j), $0<p<\infty$, and $k\ge 0$. The vector $\boldsymbol{v}$ controls anistropy, promoting importance of certain variables over others. The parameter p controls the number of interaction terms between parameters: a large p yields many interaction terms. The parameter k is the "order" of the approximation, and corresponds to polynomial degree with $p=1$ (and for this reason k is typically chosen as an integer). Larger k yields higher degree polynomial terms, which more accurately capture higher-order moments of the model response. UncertainSCI allows slightly more general values of these parameters, for example $p=0,\infty$ with appropriate definitions of $\|\cdot\|_0$ and $\|\cdot\|_\infty$ in these cases.

As common examples, the values $p=1$ and $\boldsymbol{v}=(1,1,\dots,1)$ corresponds to a set of multi-indices corresponding to polynomials up to degree-k in d variables. With the same $\boldsymbol{v}$ but $p=\infty$, then $\boldsymbol{J}$ corresponds to a tensor product space of degree-k polynomials, containing polynomials up to degree k in any single variable.

UncertainSCI also allows more general definitions of $\boldsymbol{J}$ than in (6). In particular, anisotropic hyperbolic cross sets are allowed, and adaptively-built monotone lower index sets are supported as well.

2.4. Weighted least squares

UncertainSCI uses least-squares-based training to compute the coefficients $\widehat{u}_n$. With $u(\boldsymbol{p}_m)$, $m=1,\dots,M$, data collected from the forward model over the parameter ensemble $\{\boldsymbol{p}_m\}_{m=1}^M$, we compute the coefficients via a standard weighted least-squares formulation,

$$\min_{\widehat{u}_1,\dots,\widehat{u}_N} \sum_{m=1}^M w_m^2 \left\|u_N(\boldsymbol{p}_m)-u(\boldsymbol{p}_m)\right\|_2^2, \tag{7}$$

where the weights w_m are frequently used to promote unbiasedness when the parameter ensemble $\{\boldsymbol{p}_m\}_{m=1}^M$ is generated randomly. UncertainSCI has two options for computing the candidate set and the weights, which set it apart from competing toolboxes in Table 1:

- Random sampling from a distribution biased with respect to the density w of $\boldsymbol{p}$.
- Weighted D-optimal design computed via optimization, where an initial condition for the optimization is the previous random sampling.

The first approach is conceptually straightforward: one computes the ensemble $(w_m,\boldsymbol{p}_m)_{m=1}^M$ with which to solve (7), constructing the $\boldsymbol{p}_m$ as independent and identically distributed (iid) samples. The feature that makes UncertainSCI unique is that a mean quasi-optimal error is achievable by the emulator u_N by sampling $\boldsymbol{p}_m$ via a special density (distinct from w). The second approach identifies the $\boldsymbol{p}_m$ as the solution to a D-optimal optimization problem that promotes stability of the least-squares problem (7) that determines the emulator u_N. One loses rigorous mathematical convergence statements with this approach, but our experiments (and existing investigations in the literature) suggest that this procedure is in practice more effective than random sampling.

However, one strength common to both of these approaches is their flexibility with respect to the polynomial space P_N defined by the multi-index choice $\boldsymbol{J}$. For example, the guarantees in section 2.4.1 hold for *any* choice of $\boldsymbol{J}$. Such guarantees do not hold for the sampling alternatives in table 1, *e.g.,* Monte Carlo, quasi-Monte Carlo, or sparse grids.

2.4.1. Random sampling

Here we discuss the random sampling method described in the previous paragraph, and codify the "mean best-approximation guarantees" claimed in Table 1. Our goal is to quantify the error of the computed emulator u_N relative to the best mean-square approximation u_N^*,

$$u_N^* = \operatorname*{argmin}_{v \in P_N} \|u - v\|_{L^2_w} \tag{8}$$

Of particular interest is the availability of a near-optimal approximation certificate when constructing the emulator u_N, *i.e.,* the identification of an $\epsilon > 0$ that connects the error committed by the optimization (7) to that of the optimal (best possible) emulator u_N^*:

$$\left\|u - u_N\right\|^2_{L^2_w} \leq (1+\epsilon)\left\|u - u_N^*\right\|^2_{L^2_w}. \tag{9}$$

Recent advances in leverage-score sampling methods from statistics and in optimal approximations from mathematics dictate a random sampling strategy that is parsimonious and simultaneously results in such a near-optimality certificate. Precisely, we define the distribution μ_N that is *induced* by the subspace P_N as,

$$\mathrm{d}\mu_N(\boldsymbol{p}) = \sup_{v \in P_N \setminus \{0\}} \frac{v^2(\boldsymbol{p})}{\|v\|^2_{L^2_w}} \mathrm{d}\mu(\boldsymbol{p}). \tag{10}$$

This induced distribution yields an optimality certificate ϵ in the following sense: Given some $0 < \delta < 1$ and $0 < \epsilon < 1$, let $\widetilde{M}$ samples $\widetilde{\boldsymbol{p}}_m$ be randomly drawn with weights w_m, all chosen to obey

$$\widetilde{\boldsymbol{p}}_m \sim \mu_N, \qquad w_m = \frac{\mathrm{d}\mu}{\mathrm{d}\mu_N}(\widetilde{\boldsymbol{p}}_m), \qquad \widetilde{M} \geq \frac{3\log(4N/\delta)}{\epsilon^2} N, \tag{11}$$

We can construct u_N from these $\widetilde{M}$ samples via a weighted least squares procedure (7). It turns out this is sufficient to prove rather strong guarantees on the emulator u_N. To state this result, first note that the emulator can be truncated pointwise as follows: Define $\tau > 0$ as the maximum value of u, and define a corresponding truncated emulator:

$$\tau := \max_{\boldsymbol{p}} |u(\boldsymbol{p})|, \qquad u_{N,\tau}(\boldsymbol{p}) := \begin{cases} u_N(\boldsymbol{p}), & \text{if } |u_N(\boldsymbol{p})| \leq \tau \\ \tau \operatorname{sign}(u_N(\boldsymbol{p})), & \text{if } |u_N(\boldsymbol{p})| > \tau \end{cases} \tag{12}$$

Then, on average, $u_{N,\tau}$ achieves an error bound similar to (9):

$$\mathbb{E}\|u - u_{N,\tau}\|^2_{L^2_w} \leq (1+\epsilon)\|u - u_N^*\|^2_{L^2_w} + 2\tau^2\delta \tag{13}$$

Above, the expectation is with respect to the randomness in the sampling of $\widetilde{\boldsymbol{p}}_m$. The above result is an adaptation of the seminal estimates in [10], but similar results appear in related forms elsewhere [13, 24, 29]. We emphasize the salient points of this bound: first, (13) holds in expectation, and not for any particular construction of u_N; hence it is a *mean* best-approximation guarantee. Second, (11) requires only *log-linear* sampling for $\widetilde{M}$ with respect to N, which is close to the optimal $\widetilde{M} = N$. Finally, the truncation parameter τ should be thought of as a very large value, whose impact in (13) is offset by taking smaller δ.

Note that UncertainSCI does not require τ as input, instead returning the (untruncated) emulator u_N from (7). The untruncated emulator u_N satisfies a similar bound as in (13) without the τ term, but only with high probability (more precisely, with probability at least $1 - \delta$).

Of course, to achieve the bound above, one must sample $\widetilde{\boldsymbol{p}}_m$ from μ_N defined in (10). For general P_N, this measure can be fairly complicated [17], and so using somewhat classical sampling approaches such as rejection sampling can incur a computational burden that scales exponentially with the number of parameters d. However, if the components of the random parameter $\boldsymbol{p}$ are independent, as is assumed in UncertainSCI, then sampling $\widetilde{\boldsymbol{p}_m}$ from the distribution μ_N can be efficiently accomplished with complexity linear in d [28, 25].

The strength of this strictly random approach is the guarantee (13). However, note that the number of samples $\widetilde{M}$ required in (11) can still be quite large relative to N[1] and so while UncertainSCI supports the procedure above, the default method used is a more parsimonious optimization-based sampling. This optimization procedure empirically works quite well with many fewer samples, but loses rigorous mathematical guarantees.

[1] For example, taking $\epsilon = 1/2$, $\delta = 10^{-2}$, and $N = 100$, then (11) requires $M \gtrsim 127N$.

2.4.2. Weighted D-optimal sampling

To simultaneously reduce the number of required samples M and stabilize variability due to random sampling, we employ a particular type of weighted D-optimal design, which we formulate as an optimization problem. Precisely, we utilize the method of weighted approximate Fekete points [4, 16], which approximately solves a weighted maximum volume optimization problem,

$$\underset{\boldsymbol{p}_1,\dots,\boldsymbol{p}_M \subset \{\widetilde{\boldsymbol{p}}_m\}_{m=1}^{\widetilde{M}}}{\operatorname{argmax}} (\det(A^T A)) \prod_{m=1}^{M} \frac{\mathrm{d}\mu}{\mathrm{d}\mu_N}(\boldsymbol{p}_m), \tag{14}$$

where A is the standard $M \times N$ design matrix, $(A)_{m,n} = \phi_n(\boldsymbol{p}_m)$. The optimal solution to such a weighted maxvol problem is known to have attractive stability properties. For example, one can show that the maximum possible value of the objective in (14) is unity, and if a design $\boldsymbol{p}_1, \dots, \boldsymbol{p}_M$ is computed that achieves this maximum, then the least squares problem (7) has unity condition number, and hence is optimally stable [16]. However, in practice, achieving unity objective in (14) is rare. Nevertheless, it has been observed even in this case that sampling generated according to (14) is rather effective. [16, 5]

Computing an exact solution to (14) is a rather difficult (*e.g.,* non-convex) optimization problem. However, an approximate solution is easily computed through a greedy procedure, which starts from a large discrete candidate set and prunes down this set via linear algebraic routines. This approximate solution procedure comes with provable asymptotic guarantees [4], and is the implementation in UncertainSCI. The candidate set we use is a set $\widetilde{M} > M$ of samples that are iid generated as in (11).

Once the $\boldsymbol{p}_m$ are computed as (approximate) solutions to (14), the final emulator u_N that UncertainSCI builds is the least-squares solution (7), with w_m set as in (11). Using this approach, M samples are generated that are quasi-random: The candidate set of $\widetilde{M}$ points generated through (11) is random, but the computational solution to (14) is deterministically generated from the candidate set.

The strength of this second approach is that empirically one achieves very good results even with a modest $M \approx N + 10$ samples [16, 5]. However, rigorous guarantees similar to (13) are not available.

2.4.3. Investigating accuracy of u_N

Because the weighted maximum volume sampling method from the previous section loses strict mathematical rigor, it is important to be able to ascertain accuracy of the constructed emulator u_N. UncertainSCI computes and makes available two empirical metrics for inspection that can be used to evaluate accuracy. The first is simply the residual from the least squares problem (7)

$$R^2 = \sum_{m=1}^{M} w_m^2 \|u(\boldsymbol{p}_m) - u_N(\boldsymbol{p}_m)\|_2^2. \tag{15}$$

This quantity can be used to evaluate how well the choice of polynomial space P_N accurately captures variation in the model response u. Larger values of R suggest that P_N should include more polynomial terms.

The second empirical metric is a leave-one-out cross validation metric, computed as an m-averaged discrepancy between u_N and $u_{N,m}$, where the latter is the least squares solution (7) formed by excluding sample $u(\boldsymbol{p}_m)$:

$$CV = \frac{1}{M} \sum_{m=1}^{M} \left\| u_N - u_{N,m} \right\|_{L^2_w}, \qquad u_{N,m} := \underset{v \in P_N}{\operatorname{argmin}} \sum_{\substack{j=1 \\ j \neq m}}^{M} w_j^2 \left\| u_N(\boldsymbol{p}_j) - u(\boldsymbol{p}_j) \right\|_2^2. \tag{16}$$

The CV quantity measures sensitivity of the emulator u_N to the data. Larger values of CV suggest that more samples should be collected to promote accuracy of the emulator.

2.5. Postprocessing the UQ emulator

Once the emulator u_N is built, computing statistics of the model output is accomplished with relatively simple manipulation of the emulator coefficients $\widehat{u}_n$. For example, if $\mathbb{E}$ is the mathematical expectation under the randomness due to p, then the mean of u can be approximated by,

$$\mathbb{E}\, u(\boldsymbol{p}) \approx \mathbb{E}\, u_N(\boldsymbol{p}) = \sum_{n=1}^{N} \widehat{u}_n (\mathbb{E}\, \phi_n(\boldsymbol{p})), \tag{17}$$

Table 2
Statistics computable by UncertainSCI via manipulation of a scalar-valued UQ emulator u_N. For the global sensitivity, the summation is taken over all nonempty *strict* subsets $\mathcal{J}$ of $\mathcal{I}$. For vector-valued models, UncertainSCI computes componentwise statistics.

Mean	$\mathbb{E}u_N(\boldsymbol{p})$
Variance $\mathrm{Var}(u_N)$	$\mathbb{E}(u_N(\boldsymbol{p}) - \mathbb{E}u_N(\boldsymbol{p}))^2$
Median	Value m such that $P(u_N \leq m) \geq \frac{1}{2}$ and $P(u_N \geq m) \geq \frac{1}{2}$
Quantiles	Given $\delta \in (0,1)$, value q such that $P(u_N \geq q) \geq 1-\delta$ and $P(u_N \leq q) \geq \delta$
Total sensitivity $S_{T,\mathcal{I}}$	Given $\mathcal{I} \subset \{1,\ldots,d\}$, the ratio $V(\mathcal{I})/\mathrm{Var}(u_N)$
Global sensitivity $S_{G,\mathcal{I}}$	Given $\mathcal{I} \subset \{1,\ldots,d\}$, the ratio $\frac{V(\mathcal{I})-\sum_{\emptyset\neq\mathcal{J}\subset\mathcal{I}} V(\mathcal{J})}{\mathrm{Var}(u_N)}$
Local sensitivity	Given a fixed parameter value $\widetilde{\boldsymbol{p}}$, $\nabla u_N(\widetilde{\boldsymbol{p}})$.

where $\mathbb{E}\,\phi_n(\boldsymbol{p})$ are constants that are explicitly computable (*e.g.,* in the common case where $\phi_1(\boldsymbol{p}) \equiv 1$, then $\mathbb{E}\,\phi_1 = 1$, and the remaining expectations are 0). Such simple manipulations allow for quick computations of output statistics (including higher-order moments). Of notable utility are sensitivity indices [38], which, when $\boldsymbol{p}$ has independent components, measure the relative contribution of various parameters or parameter combinations to the overall variability of the emulator. To describe these indices, we define the variance associated to parameter index set $\mathcal{I}$ as,

$$V(\mathcal{I}) := \mathrm{Var}\left(\mathbb{E}\left(u|\boldsymbol{p}_{\mathcal{I}}\right)\right), \qquad \mathcal{I} \subset \{1,\ldots,d\}, \tag{18}$$

where the conditional expectation $\mathbb{E}\left(u|\boldsymbol{p}_{\mathcal{I}}\right)$ is a random variable, and where $\boldsymbol{p}_{\mathcal{I}}$ denotes the size-$|\mathcal{I}|$ vector of random variables associated with index set $\mathcal{I}$. Sensitivity indices are computed as ratios of $V(\mathcal{I})$ to the overall variance, cf. Table 2. Several other types of statistics are computable in UncertainSCI, also included in Table 2.

3. Results

We demonstrate the utility of UncertainSCI in characterizing and exploring uncertainty for biomedical applications through two cardiac and two neural bioelectric simulation applications. UncertainSCI is open-source and publicly hosted [30]. The size of the raw data for the following experiments makes dissemination of reproducible scripts challenging, but upon request we can provide data and code associated with the figures in this manuscript.

In all our simulation, we use the weighted maximum volume sampling as described in section 2.4.2, with $M = N + 10$. The polynomial space is chosen as the isotropic total degree space $J(\boldsymbol{\nu}, 1, k)$ with $\boldsymbol{\nu} = (1, 1, \ldots, 1)$ and k as described in each section.

3.1. Cardiac application 1: Passive bidomain model

The passive bidomain describes a conceptual and numerical approach to a forward problem from cardiac cellular transmembrane potentials to extracellular potentials during the plateau phase of the cardiac action potential [7, 47]. We assume that during the plateau there exists a difference in the transmembrane potential between healthy and ischemic tissue of -40 mV, spread over a narrow transition region. We compute the extracellular potentials Φ_e by solving the bidomain equation (19) in a passive, quasi-static sense:

$$\nabla \cdot (\sigma_i \nabla \Phi_m) = -\nabla \cdot (\sigma_h \nabla \Phi_e), \tag{19}$$

where the transmembrane potentials Φ_m, intracellular (σ_i), and tissue (σ_h) conductivity tensors are defined throughout the myocardium. The conductivity tensor (σ) is described in terms of longitudinal σ_L and transverse σ_T components for both intracellular and interstitial domains.

Table 3 contains typical conductivity values used in contemporary models; however, these values are unknown in realistic settings. We will therefore model these conductivity values as random parameters and use UncertainSCI to estimate the resulting uncertainty in the extracellular potentials. Equation (19) is then discretized via finite elements, resulting in a simulation that maps transmembrane potentials, via random conductivity values, to extracellular potentials on the heart surface.

Table 3
Conductivity ratios relative to the base conductivity value (0.16 S/cm^2) for each tissue label. Subscripts on σ indicate the tissue domain (i/e = intracellular/extracellular) and the direction of conductivity (L/T = longitudinal/transverse). The subscript 'B' represents the blood pool, where conductivity does not have a direction. For example, σ_{eL} is conductivity ratio in the extracellular domain in the longitudinal direction with respect to the base value, and σ_{iB} is the intracellular blood pool conductivity ratio with respect to the same base value.

Conductivity parameter		Healthy Tissue	Ischemic Tissue
Intracellular longitudinal	σ_{iL}	1.0	0.1
Intracellular transverse	σ_{iT}	0.1	0.001
Intracellular blood pool	σ_{iB}	0.0	0.0
Extracellular longitudinal	σ_{eL}	1.0	0.5
Extracellular transverse	σ_{eT}	0.333	0.25
Extracellular blood pool	σ_{eB}	3.0	3.0

We quantified forward parametric uncertainty in this passive bidomain model using UncertainSCI. We examined the effects of varying the conductivity parameters σ_{iL}, σ_{iT}, σ_{eL}, and σ_{eT} within a known myocardial region in which there was acute myocardial ischemia. We varied these values uniformly with a range of $\pm 20\%$ of the values listed in Table 3, assuming independence of the parameters. We used a space of polynomials up to degree $k = 5$ for the subspace P_N. Figure 1 shows the results of this analysis, depicting various statistics and sensitivities. The UQ analysis indicates a clear dependence on longitudinal conductivities σ_{iL} and σ_{eL} compared to profound insensitivity to transverse values σ_{iT}, σ_{eT}. We also noted that the standard deviation of estimated extracellular potentials was highest in the region of highest disagreement between the measurements and forward solutions.

3.2. Cardiac application 2: Rule-based myocardial fiber orientations

We implemented a finite element simulation of the spread of electrical activation in the ventricles of the heart in which we varied the directions of the fibers that make up heart tissue. The mean directions of these fibers were assigned within each region in the heart by an angle, α, with respect to a 'short axis', *i.e.,* a direction that crosses the width of the heart. Baseline values of α represented the local average direction and could be assigned based on published rules [2, 15] to smoothly transition between -60° on the outer surface (epicardium), α_{epi}, to +60° on the inner surface (endocardium), α_{endo} of the ventricles [23].

We used UncertainSCI to predict uncertainty in the spread of electrical activation through the heart associated with variability in the fiber angles around the baseline values. Specifically, we modeled α_{epi} and α_{endo} as independent and uniformly distributed from -35 to -85° and 35 to 85°, respectively [23]. We used degree $k = 5$ for the polynomial space P_N and analyzed the spread of activation following a single stimulus from the epicardium of the left ventricular free wall. The *CARPentry* simulator was applied to calculate the spread of activation via a bi-Eikonal normalization method [45, 31, 15].

Figure 2A shows the epicardial projections of the activation sequence for baseline fiber orientations along with the mean and standard deviation from UncertainSCI. The baseline and mean activation sequences matched closely while the overall standard deviation was only 10% as large and showed elevations near the anterior base and apex. UncertainSCI also provides (Figure 2B) separate standard deviations associated with the epicardial and endocardial fiber orientations (*i.e.,* the variances V_1 and V_2 from (18)) as well as the combined global sensitivity map, *i.e.,* $S_{1,2}$. Variations in the epicardial fiber orientation produced higher standard deviations (≤ 9) than did variations in the endocardial fiber orientation (≤ 2.7) with distinctly different spatial patterns. The combined global sensitivity was another order of magnitude smaller, suggesting that the impacts of varying α_{endo} and α_{epi} are largely uncorrelated, showing only very small regions of significant variability (≤ 0.6) on the right ventricle.

3.3. Neural application 1: Transcranial direct current stimulation

Transcranial direct current stimulation (tDCS) sends weak currents through the brain via electrodes on the scalp to modulate brain function, known as neuromodulation. Effects range from improved memory function to faster rehabilitation after stroke [22]. Finite element models can simulate the electric fields induced by tDCS to understand and improve experimental and clinical results. Such models represent the geometry of tissues in the head based on

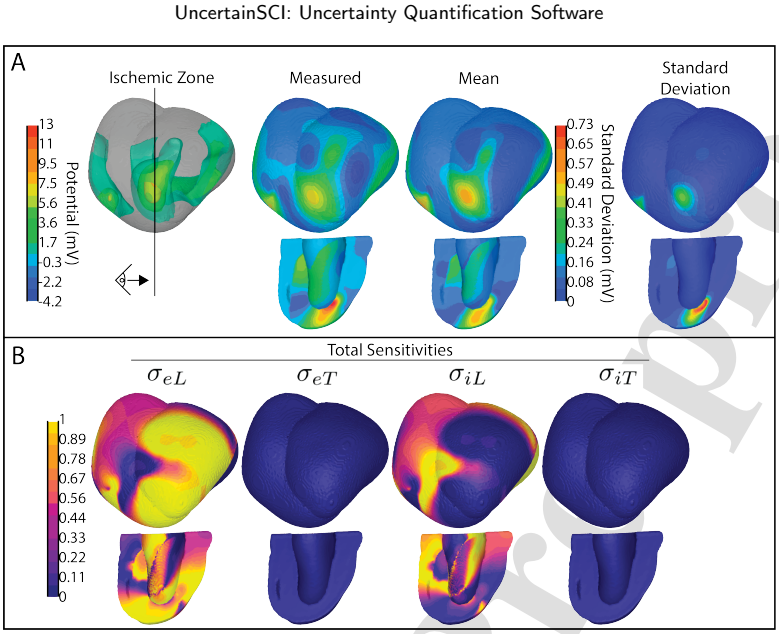

Figure 1: Cardiac bioelectric simulations with the passive bidomain model: UQ of simulated zones of myocardial ischemia relative to measured potentials. **A:** Isosurfaces of 1) the measured ischemic regions (black line shows cut plane for subsequent visualization, with eye glyph pointing in direction of visualization), 2) measured extracellular potentials, 3) the mean forward solution, and 4) the standard deviation due to variation in the four ischemic conductivity values. **B:** Sensitivity due to variation in 1) the extracellular longitudinal conductivity (σ_{eL}), 2) extracellular transverse conductivity (σ_{eT}), 3) intracellular longitudinal conductivity (σ_{iL}), and 4) intracellular transverse conductivity (σ_{iT}). Results adapted with permission from [3].

magnetic resonance images (MRI) and use tissue conductivities from the literature, but there is significant variation even in the literature for these conductivity values. We used UncertainSCI to quantify how this variability affects tDCS-induced electric fields.

We employed a finite element head model of a healthy volunteer (Fig. 3A), commonly used stimulation parameters (electrodes over the motor cortices, 1 mA current), and standard numerical methods for solving the resulting equations [34]. We connected the SCIRun simulation software [8] to UncertainSCI to model variability due to all tissue conductivities, which were endowed with beta distributions with parameters $\alpha = \beta = 3$ over predetermined intervals for each parameter. The conductivities we considered were scalp (0.28–0.87 S/m), skull (0.0016–0.33 S/m), cerebrospinal fluid (CSF, 1.7696–1.8104 S/m), gray matter (0.22–0.67 S/m) and white matter (0.09–0.29 S/m) [46]. A polynomial order of 10 was used for the space P_N, and UncertainSCI utilized a parameter ensemble of size 1297, thus requiring 1297 separate SCIRun simulations.

Fig. 3 presents results shown on coronal slices of the head model taken through the centers of the electrodes. The mean electric field strength (Fig. 3B) of all simulations is almost identical to a baseline simulation using the mean of all conductivity parameters (not shown). Standard deviations were high in areas where mean field strengths were high (Fig. 3B). Total sensitivity values in Fig. 3C indicate the relative contributions of each tissue conductivity to the standard deviation in Fig. 3B. In the targeted brain area, the motor cortex, the electric field strength was primarily affected by uncertainty in scalp and CSF conductivities. Overall, uncertainty in scalp conductivity had the largest effect.

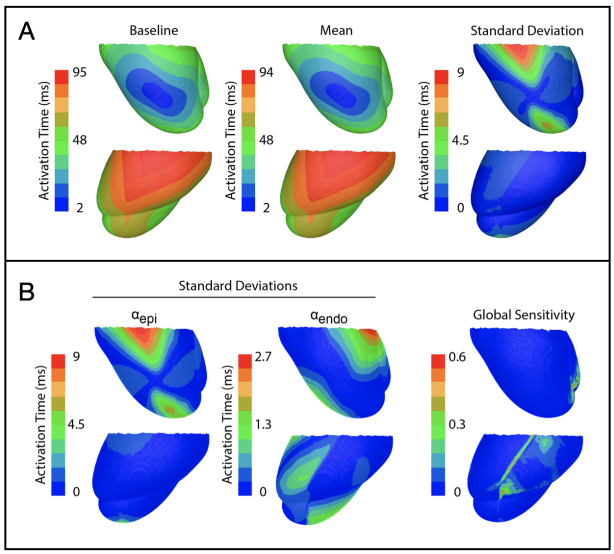

Figure 2: A: Activation sequence for an epicardial stimulation location, showing 1) the activation sequence for default fiber orientations, 2) the mean activation sequence, and 3) the total standard deviation of the activation sequence. The top row of each pair contains left-ventricular views and the bottom row contains right-ventricular views. **B:** Parameter sensitivities for an epicardial stimulation location, showing the standard deviation contributions of the 1) epicardial and 2) endocardial fiber orientation, and 3) the global sensitivities of the activation sequence due to both parameters. Views are the same as in panel A. Results adapted with permission from [35].

3.4. Neural application 2: Direct electrocortical stimulation (DECS)

Neuromodulation can also occur by means of currents applied directly to the cortex through electrocorticography (ECoG) grids implanted under the skull. ECoG grids can both stimulate and measure electrical activity. Clinical applications include brain-computer interfaces and mapping brain regions necessary for normal function before resection surgery. Applying simulation models of DECS to these scenarios can help investigators to understand experimental and clinical results and optimize stimulation to specific regions of interest. The major sources of uncertainty in these models include tissue conductivities, electrode locations, and postoperative brain shift, all of which contribute to uncertainty in model predictions. Here we quantified the uncertainty of the simulated voltages due to conductivity and electrode location variability.

We created a detailed finite element model of the brain and CSF of an epilepsy patient undergoing preoperative monitoring. We simulated bipolar stimulation with 0.75 mA of current between a pair of neighboring ECoG electrodes located on motor cortex (see blue and red electrodes in Fig. 4A) with SCIRun using previously described methods [9], and calculated the resulting potential at each node in the brain. We used UncertainSCI to quantify the uncertainty resulting from variable tissue conductivities, where each conductivity was modeled with a beta distribution using $\alpha = \beta = 3$ over the following intervals: CSF (1.7696–1.8104 S/m), gray matter (0.22–0.67 S/m) and white matter (0.09–0.29 S/m). We additionally modeled the variable locations of the stimulating electrodes as point sources by randomly selecting a node within a four-millimeter radius of the electrode centroid from a discrete uniform distribution.

Fig. 4 shows a sagittal slice through the head model that includes the center of the bipolar stimulating electrodes. Fig. 4A shows the mean and standard deviation of the voltage resulting from bipolar stimulation. Voltages were greatest directly surrounding the stimulating electrodes and dropped off with distance; areas around the electrodes also exhibited the largest standard deviations due to uncertainty. The total sensitivity values for each parameter (Fig. 4B) quantify the relative contributions of uncertainty in tissue conductivities and electrode locations. From these results, we conclude

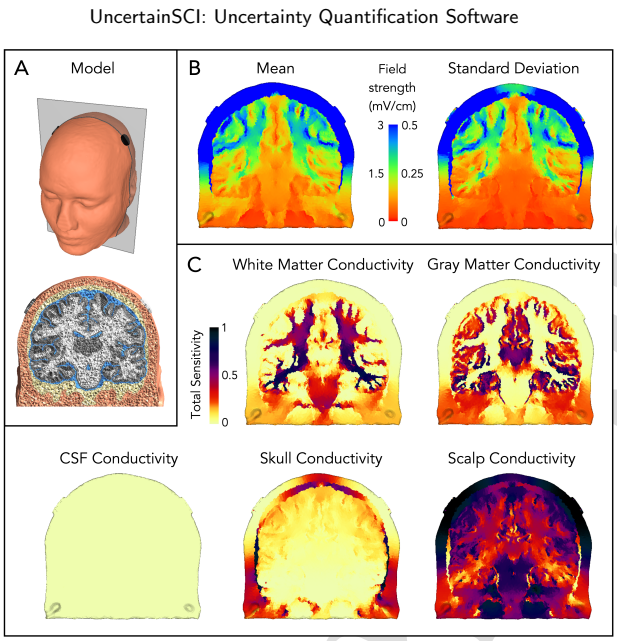

Figure 3: Quantification of uncertainty in simulations of transcranial direct current stimulation (tDCS). **A:** Head model with two electrodes through which current flow was simulated. The vertical plane indicates where the model was cut for all subsequent figures. The model includes skin (pink), skull (yellow), CSF (blue), gray matter (gray) and white matter (white). **B:** Mean and standard deviation of electric field strength in 1297 simulations of tDCS with all tissue conductivities modeled with uncertainty. **C:** Total sensitivity values for each tissue conductivity indicate their relative contributions to the standard deviation.

that variability in electrode locations contributed more uncertainty to the resulting voltages than did variability in CSF conductivity.

4. Conclusion

We have shown the utility of the Python-based software suite UncertainSCI in characterizing and exploring forward uncertainty in biomedical simulation pipelines. In particular we emphasize the following advantages of UncertainSCI:

- The software package UncertainSCI is an open-source Python package that implements non-intrusive forward UQ methods by building emulators.
- UncertainSCI builds emulators through recent efficient implementations of randomized techniques that boast near-optimal convergence guarantees [24, 48, 10, 25].
- Emulators can be easily manipulated to yield statistics and sensitivities of model outputs, providing an informative summary of forward UQ in simulations.

These capabilities of UncertainSCI allow users to evaluate and compare the impacts of parameter variations in complex biological models. In particular we have demonstrated UncertainSCI's capabilities on four target applications:

- uncertainty in cardiac extracellular potentials in a passive bidomain model due to variation in conductivity parameters;
- impact of uncertainty in cardiac fiber orientation on electrical activation in ventricles of the heart;

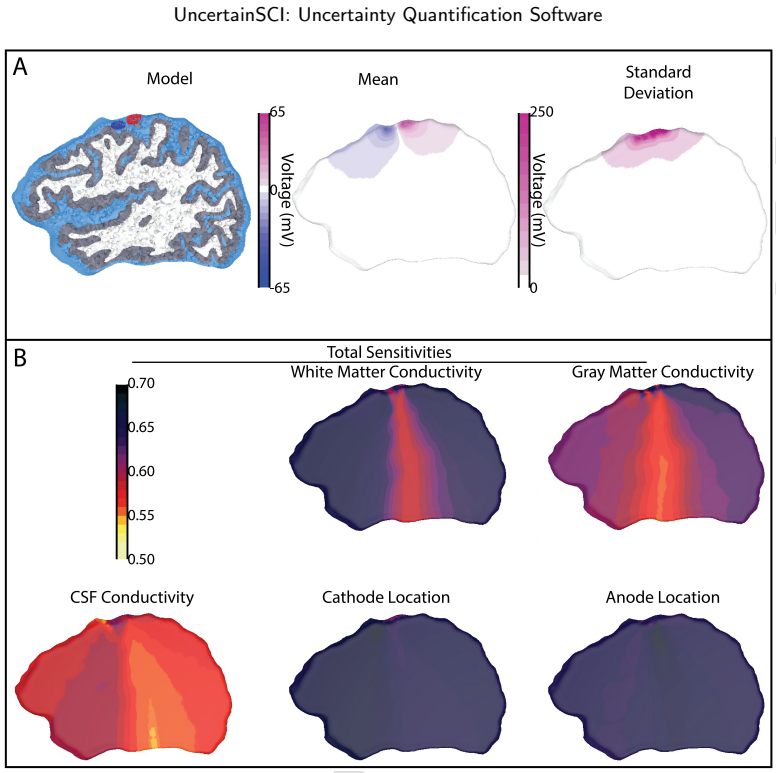

Figure 4: Quantification of uncertainty in tissue conductivities and electrode locations on the voltage resulting from direct electrocortical stimulation. **A:** Sagittal slice through the finite element model, which includes CSF (blue), gray matter (gray) and white matter (white). The nodes used as possible locations for the cathode (blue) and anode (red) are shown on the CSF surface. The mean voltage and standard deviation due to uncertainty in tissue conductivities and electrode locations are shown on the same sagittal slice. **B:** Total sensitivity values for each uncertain parameter indicate their relative contributions to the standard deviation.

- impact of tissue conductivity uncertainty in transcranical direct current stimulation;
- uncertainty quantification for direct electrocortical stimulation varying tissue conductivity.

Future development of UncertainSCI will build in several new capabilities and methods to expand the types of UQ problems that can be addressed, including features to handle dependent random variables, adaptive emulator construction, inverse problems, and optimization-based design.

5. Acknowledgements

This project was supported by grants from the National Institute of Biomedical Imaging and Bioengineering (U24EB029012) and National Institute of General Medical Sciences (P41GM103545 and R24GM136986) from the National Institutes of Health.

References

[1] B.M. Adams, W.J. Bohnhoff, K.R. Dalbey, M.S. Ebeida, J.P. Eddy, M.S. Eldred, R.W. Hooper, P.D. Hough, K.T. Hu, J.D. Jakeman, M. Khalil, K.A. Maupin, J.A. Monschke, E.M. Ridgeway, A.A. Rushdi, D.T. Seidl, J.A. Stephens, L.P. Swiler, and J.G. Winokur, *DAKOTA, a multilevel*

parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 6.15 User's Manual, Tech. report, Sandia National Laboratories Albuquerque, NM, 2021, Sandia Technical Report SAND2020-12495.

[2] Jason D Bayer, Robert C Blake, Gernot Plank, and Natalia A Trayanova, *A novel rule-based algorithm for assigning myocardial fiber orientation to computational heart models*, Annals of Biomedical Engineering **40** (2012), no. 10, 2243–2254.

[3] Jake A. Bergquist, B. Zenger, L. C. Rupp, A. Narayan, and R. S. MacLeod, *Uncertainty quantification in simulations of myocardial ischemia*, 2021 Computing in Cardiology, in Press, 2021, pp. 1–4.

[4] L. Bos, S. De Marchi, A. Sommariva, and M. Vianello, *Computing Multivariate Fekete and Leja Points by Numerical Linear Algebra*, SIAM Journal on Numerical Analysis **48** (2010), no. 5, 1984.

[5] Kyle M. Burk, Akil Narayan, and Joseph A. Orr, *Efficient sampling for polynomial chaos-based uncertainty quantification and sensitivity analysis using weighted approximate Fekete points*, International Journal for Numerical Methods in Biomedical Engineering **36** (2020), no. 11, e3395, arXiv: 2008.04854.

[6] John Burkardt and Michael Eldred, *Comparison of Non-Intrusive Polynomial Chaos and Stochastic Collocation Methods for Uncertainty Quantification*, 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, American Institute of Aeronautics and Astronautics, 2009.

[7] B.M. Burton, K.K. Aras, W.W. Good, J.D. Tate, B. Zenger, and R.S. MacLeod, *A framework for image-based modeling of acute myocardial ischemia using intramurally recorded extracellular potential*, Annal. Biomed. Eng. **46** (2018), no. 9, 1325–1336.

[8] B.M. Burton, J.D. Tate, B. Erem, D.J. Swenson, D.F. Wang, D.H. Brooks, P.M. van Dam, and R.S. MacLeod, *A toolkit for forward/inverse problems in electrocardiography within the SCIRun problem solving environment.*, Proceedings of the IEEE Engineering in Medicine and Biology Society 33rd Annual International Conference, IEEE Eng. in Med. and Biol. Soc., 2011, pp. 1–4.

[9] C.M. Charlebois, D.J. Caldwell, S.M. Rampersad, A.P. Janson, J.G. Ojemann, D.H. Brooks, R.S. MacLeod, C.R. Butson, and A.D. Dorval, *Validating patient-specific finite element models of direct electrocortical stimulation.*, Front Neurosci **15** (2021), 691701.

[10] A. Cohen and G. Migliorati, *Optimal weighted least-squares methods*, SMAI Journal of Computational Mathematics **3** (2017), 181–203, arxiv:1608.00512 [math.NA].

[11] Bert Debusschere, Khachik Sargsyan, Cosmin Safta, and Kenny Chowdhary, *Uncertainty Quantification Toolkit (UQTk)*, Handbook of Uncertainty Quantification (Roger Ghanem, David Higdon, and Houman Owhadi, eds.), Springer International Publishing, 2017, pp. 1807–1827.

[12] Bert J. Debusschere, Habib N. Najm, Philippe P. Pebay, Omar M. Knio, Roger G. Ghanem, and Olivier P. Le Maitre, *Numerical Challenges in the Use of Polynomial Chaos Representations for Stochastic Processes*, SIAM Journal on Scientific Computing **26** (2004), no. 2, 698–719.

[13] Petros Drineas, Michael W. Mahoney, S. Muthukrishnan, and Tamás Sarlós, *Faster least squares approximation*, Numerische Mathematik **117** (2011), no. 2, 219–249.

[14] J. Feinberg and H.P. Langtangen, *Chaospy: An open source tool for designing methods of uncertainty quantification*, Journal of Computational Science **11** (2015), 46–57.

[15] Karli Gillette, Matthias A.F. Gsell, Anton J. Prassl, Elias Karabelas, Ursula Reiter, Gert Reiter, Thomas Grandits, Štern Darko, Martin Urschler, Jason Bayer, Christoph A. Augustin, Aurel V. Neic, Thomas Pock, Edward J. Vigmond, and G. Plank, *A framework for the generation of digital twins of cardiac electrophysiology from clinical 12-leads ECGs*, Accepted: Medical Image Analysis (2021).

[16] L. Guo, A. Narayan, L. Yan, and T. Zhou, *Weighted Approximate Fekete Points: Sampling for Least-Squares Polynomial Approximation*, SIAM Journal on Scientific Computing **40** (2018), no. 1, A366–A387, arXiv:1708.01296 [math.NA].

[17] Ling Guo, Akil Narayan, and Tao Zhou, *Constructing Least-Squares Polynomial Approximations*, SIAM Review **62** (2020), no. 2, 483–508.

[18] John Jakeman, *PyApprox: Enabling efficient model analysis.*, Tech. Report SAND2022-10458, Sandia National Lab. (SNL-NM), Albuquerque, NM (United States), 2022.

[19] John D. Jakeman, Fabian Franzelin, Akil Narayan, Michael Eldred, and Dirk Plfüger, *Polynomial chaos expansions for dependent random variables*, Computer Methods in Applied Mechanics and Engineering **351** (2019), 643–666.

[20] AM Janssen, SM Rampersad, F Lucka, B Lanfer, S Lew, Ü Aydin, CH Wolters, DF Stegeman, and TF Oostendorp, *The influence of sulcus width on simulated electric fields induced by transcranial magnetic stimulation.*, Phys Med Biol **58** (2013), no. 14, 4881–96.

[21] J Jiang, DQ Truong, Z Esmaeilpour, Y Huang, BW Badran, and Bikson M, *Enhanced tES and tDCS computational models by meninges emulation*, J Neural Eng **17** (2020), no. 1, 016027.

[22] JP Lefaucheur, A Antal, SS Ayache, DH Benninger, J Brunelin, F Cogiamanian, M Cotelli, D De Ridder, R Ferrucci, B Langguth, P Marangolo, V Mylius, MA Nitsche, F Padberg, U Palm, E Poulet, A Priori, S Rossi, M Schecklmann, S Vanneste, U Ziemann, L Garcia-Larrea, and W Paulus, *Evidence-based guidelines on the therapeutic use of transcranial direct current stimulation (tdcs)*, Clin Neurophysiol **128** (2017), no. 1, 56–92.

[23] Herve Lombaert, Jean-Marc Peyrat, Pierre Croisille, Stanislas Rapacchi, Laurent Fanton, Farida Cheriet, Patrick Clarysse, Isabelle Magnin, Hervé Delingette, and Nicholas Ayache, *Human atlas of the cardiac fiber architecture: study on a healthy population*, IEEE Transactions on Medical Imaging **31** (2012), no. 7, 1436–1447.

[24] Michael W. Mahoney, *Randomized Algorithms for Matrices and Data*, Foundations and Trends® in Machine Learning **3** (2011), no. 2, 123–224, arXiv: 1104.5557.

[25] Osman Asif Malik, Yiming Xu, Nuojin Cheng, Stephen Becker, Alireza Doostan, and Akil Narayan, *Fast Algorithms for Monotone Lower Subsets of Kronecker Least Squares Problems*, 2022, arXiv:2209.05662 [cs, math].

[26] Stefano Marelli and Bruno Sudret, *UQLab: A Framework for Uncertainty Quantification in Matlab*, (2014), 2554–2563.

[27] Habib N. Najm, *Uncertainty Quantification and Polynomial Chaos Techniques in Computational Fluid Dynamics*, Annual Review of Fluid Mechanics **41** (2009), no. 1, 35–52.

[28] A. Narayan, *Computation of Induced Orthogonal Polynomial Distributions*, Electronic Transacations in Numerical Analysis **50** (2017), 71–97, arXiv:1704.08465 [math].

[29] A. Narayan, J. Jakeman, and T. Zhou, *A Christoffel function weighted least squares algorithm for collocation approximations*, Mathematics of Computation **86** (2017), no. 306, 1913–1947, arXiv: 1412.4305 [math.NA].

[30] Akil Narayan, Zexin Liu, Jake Bergquist, Chantel Charlebois, Sumientra Rampersad, Lindsay Rupp, Dana Brooks, Dan White, JEss Tate, and Rob S. MacLeod, *UncertainSCI*, `https://github.com/SCIInstitute/UncertainSCI/`, 2022.

[31] Aurel Neic, Matthias AF Gsell, Elias Karabelas, Anton J Prassl, and Gernot Plank, *Automating image-based mesh generation and manipulation tasks in cardiac modeling workflows using meshtool*, SoftwareX **11** (2020), 100454.

[32] Matthew Parno, Andrew Davis, Linus Seelinger, and Youssef Marzouk, *Mit uncertainty quantification (muq) library*, 2014.

[33] Chiara Piazzola and Lorenzo Tamellini, *The Sparse Grids Matlab kit – a Matlab implementation of sparse grids for high-dimensional function approximation and uncertainty quantification*, arXiv:2203.09314 [cs, math] (2022), arXiv: 2203.09314.

[34] SM Rampersad, AM Janssen, F Lucka, Ü Aydin, B Lanfer, S Lew, CH Wolters, DF Stegeman, and TF Oostendorp, *Simulating transcranial direct current stimulation with a detailed anisotropic human head model*, IEEE Trans Neural Syst Rehabil Eng **22** (2014), no. 3, 441–52.

[35] L. C. Rupp, Jake A. Bergquist, B. Zenger, K. Gillette, A. Narayan, G. Plank, and R. S. MacLeod, *The role of myocardial fiber direction in epicardial activation patterns via uncertainty quantification*, 2021 Computing in Cardiology, in Press, 2021, pp. 1–4.

[36] Guilherme B. Saturnino, Axel Thielscher, Kristoffer H. Madsen, Thomas R. Knösche, and Konstantin Weise, *A principled approach to conductivity uncertainty analysis in electric field calculations*, NeuroImage **188** (2019), 821–834.

[37] R. C. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications*, SIAM-Society for Industrial and Applied Mathematics, Philadelphia, December 2013 (English).

[38] I.M. Sobol, *Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates*, Mathematics and Computers in Simulation **55** (2001), no. 1-3, 271–280.

[39] M Stoyanov, *User manual: Tasmanian sparse grids*, Tech. Report ORNL/TM-2015/596, Oak Ridge National Laboratory, One Bethel Valley Road, Oak Ridge, TN, 2015.

[40] Miroslav Stoyanov, Damien Lebrun-Grandie, John Burkardt, and Drayton Munster, *Tasmanian*, 9 2013.

[41] Miroslav K Stoyanov and Clayton G Webster, *A dynamically adaptive sparse grids method for quasi-optimal interpolation of multidimensional functions*, Computers & Mathematics with Applications **71** (2016), no. 11, 2449–2465.

[42] T. J. Sullivan, *Introduction to Uncertainty Quantification*, 1st ed. 2015 edition ed., Springer, New York, NY, December 2015 (English).

[43] Jess D. Tate, Wilson W. Good, Nejib Zemzemi, Machteld Boonstra, Peter van Dam, Dana H. Brooks, Akil Narayan, and Rob S. MacLeod, *Uncertainty quantification of the effects of segmentation variability in ECGI*, Functional Imaging and Modeling of the Heart, Springer-Cham, Palo Alto, USA, 2021, pp. 515–522.

[44] S. Tennøoe, G. Halnes, and G.T. Einevoll, *Uncertainpy: A Python Toolbox for Uncertainty Quantification and Sensitivity Analysis in Computational Neuroscience*, Frontiers in Neuroinformatics **12** (2018).

[45] EJ Vigmond, R Weber Dos Santos, AJ Prassl, M Deo, and G Plank, *Solvers for the cardiac bidomain equations*, Progress in Biophysics and Molecular Biology **96** (2008), no. 1, 3–18.

[46] J Vorwerk, Ü Aydin, CH Wolters, and CR Butson, *Influence of head tissue conductivity uncertainties on eeg dipole reconstruction*, Front Neurosci **13** (2019), no. 531.

[47] D. Wang, R.M. Kirby, R.S. MacLeod, and C.R. Johnson, *Inverse electrocardiographic source localization of ischemia: An optimization framework and finite element solution*, J. Comp. Phy. **250** (2013), 403–424.

[48] David P. Woodruff, *Sketching as a Tool for Numerical Linear Algebra*, Foundations and Trends® in Theoretical Computer Science **10** (2014), no. 1–2, 1–157.

[49] D. Xiu, *Numerical Methods for Stochastic Computations: A Spectral Method Approach*, Princeton University Press, July 2010.

[50] D. Xiu and G.E. Karniadakis, *The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations*, SIAM Journal on Scientific Computing **24** (2002), no. 2, 619–644.

Highlights

- UncertainSCI is novel software that implements support for uncertainty quantification
- UncertainSCI is open-source python-based software that connects non-intrusively to any simulation software package
- With four biomedical examples, we show usability, accuracy and efficiency of UncertainSCI

Conflict of Interest Statement

None Declared