Journal of Computational Physics 385 (2019) 135–162

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A gPC-intrusive Monte-Carlo scheme for the resolution of the uncertain linear Boltzmann equation

Gaël Poëtte

CEA CESTA DAM, F-33114 Le Barp, France

ARTICLE INFO

Article history:
Received 26 May 2018
Received in revised form 27 October 2018
Accepted 31 January 2019
Available online 25 February 2019

Keywords:
Transport
Monte-Carlo
Uncertainty quantification
Intrusive
Sensitivity Analysis
Generalized Polynomial Chaos

ABSTRACT

In this paper, we are interested in the resolution of the time-dependent problem of particle transport in a media whose composition is uncertain. The most common resolution strategy consists of running, at prescribed points in uncertain space (at experimental designs points), a simulation device as a black-box. This ensures performing the uncertainty propagation resolution (i.e. resolution of SPDE). This kind of strategy is commonly called *non-intrusive*. The non-intrusive resolution can be carried with Monte-Carlo, quasi Monte-Carlo, generalized Polynomial Chaos etc. The latter is of interest in this paper for its fast convergence rate. First, we go over and illustrate the main drawbacks of the non-intrusive (gPC or otherwise) uncertainty propagation resolution for the linear Boltzmann equation in a simplified configuration. We then build a new gPC based MC scheme solving *intrusively* the uncertain counterpart of the problem. The paper ends with some numerical examples.

1. Introduction

Polynomial Chaos [18,33,97] and its generalizations (gPC[1]) [30,46,48,67,69,73,91,93,95] have been successfully applied to take into account uncertainties in many physical domains (stochastic elastic materials [33], finite deformations [1], heat conduction [92], incompressible flows [47,63,102], reacting flows and detonation [53], computational fluid dynamics [46,71,95] etc.). It is commonly accepted it stands for an efficient alternative to Monte-Carlo (MC) methods in relatively low stochastic dimensions (small number of uncertain parameters). Two observations can be done considering the existing literature on the subject:

(i) first, if gPC is very efficient in physical applications involving regular/smooth solutions (such as thermal heat [38,59,92,101], structural mechanics and reliability [10,11,83,84], etc.), it needs to be wisely adapted/modified for problems involving strong nonlinearities and solutions bearing steep gradient to achieve a comparable performance, see [30,46,48,70,95] amongst many others. In other words, the direct efficiency of gPC is closely related to the structure, dictating the regularity of the solution, of the set of partial differential equations (PDEs) modeling the physical phenomenon of interest. In this paper, we tackle uncertainty quantification (UQ) with gPC applied to the linear Boltzmann equation. The relevance of the latter integro-differential PDE is not to demonstrate. Amongst the applications, one can quote biology [68] with population dynamics, plasma physics (transport of ions and electrons) [23], photonics [19,37,52,62,65,74] or neutronics [24,25,39,40,51,80].

E-mail address: gael.poette@cea.fr.

[1] We will use the abbreviation gPC to denote all of them indifferently.

https://doi.org/10.1016/j.jcp.2019.01.052
0021-9991/

(ii) Second, gPC can be applied *intrusively* (see for examples [21,22,31–34,38,42–44,46,58,77,81,89,100]) or *non-intrusively* (most common application) and whether one strategy performs better than the other is far from obvious. An intrusive resolution implies modifying or even rewriting the simulation code. One needs to investigate on how the uncertainties play a role in the structure of the set of PDEs of interest. Considerations about its parallel resolution are case dependent and complex.
A non-intrusive resolution on the other hand uses an already existing code as a black-box just as an MC method. A description of non-intrusive applications is detailed in section 2 together with some references. It is commonly accepted it is the most convenient and used strategy. The parallel considerations are immediate: one can launch as many independent runs as available computational devices without the need for communications between processors (commonly called *embarrassingly parallel*). The uncertainty analyst does not really need to know the content of the black-box device, neither the underlying PDE structure it solves, nor the numerical solver embedded.

In the following, we build a gPC-intrusive MC (gPC-i-MC) scheme to solve the uncertain linear Boltzmann equation. This may seem surprising as MC methods seem intrinsically non-intrusive. However, care will be taken in the following sections to highlight why such an unconventional and original resolution strategy is relevant regarding this particular set of PDEs. To sum up, we will see that with the new approach,

- the uncertainty analysis can be done with only a minor additional cost with respect to *one* deterministic MC simulation solving the linear Boltzmann equation,
- and can be achieved with minor modifications of an existing MC code solving the deterministic linear Boltzmann equation.

In other words, in this context, i.e. the MC resolution of the deterministic linear Boltzmann equation, opening the black-box resolution code can be very efficient.

We are interested in the resolution of the uncertain linear Boltzmann equation. It models the time-dependent problem of particle transport in a media whose collisional characteristics (reaction rates or compositions), initial conditions or boundary conditions are uncertain. The resolution of the transport equation in random media is consequently a particular configuration (uncertain collision term) of the problem we aim to address. We suppose transport to be driven by the linear Boltzmann equation (1) for particles having position $x \in \mathcal{D} \subset \mathbb{R}^3$, velocity $\mathbf{v} \in \mathbb{R}^3$, at time $t \in [0,T] \subset \mathbb{R}^+$ and where the quantity $u(x,t,\mathbf{v})$ is the density of presence of the particles at $(x,t,\mathbf{v})$. The Cauchy problem satisfied by u is:

$$\begin{cases} \partial_t u(x,t,\mathbf{v}) + \mathbf{v}\cdot\nabla_x u(x,t,\mathbf{v}) = -v\sigma_t(\eta(x,t),\mathbf{v})u(x,t,\mathbf{v}) + \displaystyle\int v\sigma_s(\eta(x,t),\mathbf{v},\mathbf{v}')u(x,t,\mathbf{v}')\,\mathrm{d}\mathbf{v}', \\ u(x,0,\mathbf{v}) = u_0(x,\mathbf{v}). \end{cases} \tag{1}$$

In (1), we introduced the notation $|\mathbf{v}| = v$ to denote the norm of the velocity $\mathbf{v}$. Later on, we will also use $\omega = \frac{\mathbf{v}}{v}$, the unitary vector for the direction of the particles. Equation (1) must come with proper boundary conditions for wellposedness [35,45] but we omit them for the sake of conciseness. In other words, the Cauchy problem (1) is valid in an infinite medium [16,35,51,80]. The left hand side of (1) will be hinted at as the *streaming* counterpart of (1) whereas its right hand side will be called the *collisional* one. The above equation is linear and can be used to model the behavior of particles interacting with a *background* media defined *via* both the vector of compositions $\eta(x,t) = (\eta_1(x,t),...,\eta_M(x,t))^t$ and the microscopic properties of its components denoted by $(\sigma_{\alpha,m})_{\alpha\in\{t,s\},m\in\{1,...,M\}}$. The interaction of particles with matter is described through the macroscopic total interaction probability of particles with media $\sigma_t(x,t,\mathbf{v})$ and a scattering one $\sigma_s(x,t,\mathbf{v},\mathbf{v}')$, both characterized by

$$\sigma_t(\eta(x,t),\mathbf{v}) = \sum_{m=1}^{M}\eta_m(x,t)\sigma_{t,m}(\mathbf{v}), \text{ and } \sigma_s(\eta(x,t),\mathbf{v},\mathbf{v}') = \sum_{m=0}^{M}\eta_m(x,t)\sigma_{s,m}(\mathbf{v},\mathbf{v}'). \tag{2}$$

The microscopic collision term $(\sigma_{s,m})_{m\in\{1,...,M\}}$ can be decomposed into reaction cross-sections and can account for absorption, scattering, multiplication or even more particular types of reactions (all the $n-n'$ ones for example in neutronic applications). In the following, we will gather every reaction under a unique collision term σ_s for the sake of conciseness but the methodology described in this document takes into account uncertainties in the different reactions.

Consider the case of an uncertain collision term/a random medium then $(\sigma_\alpha)_{\alpha\in\{s,t\}}$ are characterized stochastic processes, where we want to quantify its impact on the particle flow. It is common, in uncertainty quantification works [1,33,47,53,63,92,102], to emphasize the fact a quantity is uncertain by introducing explicitly an additional dependence to the unknown of interest with respect to an uncertain parameter (a stochastic process) here denoted by $X(x,t,\mathbf{v})$: we have $\sigma_t(x,t,\mathbf{v},X(x,t,\mathbf{v}))$ and $\sigma_s(x,t,\mathbf{v},\mathbf{v}',X(x,t,\mathbf{v}))$. Note that without loss of generality in the following sections, we will consider that X is a vector $X = (X_1,...,X_Q)^t$ of Q independent random variables of probability measure $\mathrm{d}\mathcal{P}_X = \prod_{i=1}^{Q}\mathrm{d}\mathcal{P}_{X_i}$ rather than a stochastic

process: in theory, it is always possible to come back to such framework.[2] The stochasticity can here affect the media compositions or the microscopic cross-sections (reaction by reaction if wanted), the formulation is general enough. For the sake of conciseness in the following sections, we may drop the dependences and write

$$\sigma_t(x,t,\mathbf{v},X(x,t,\mathbf{v}))=\sigma_t(x,t,v,X) \text{ and } \sigma_s(x,t,\mathbf{v},\mathbf{v}',X(x,t,\mathbf{v}))=\sigma_s(x,t,\mathbf{v},\mathbf{v}',X).$$

As a result, the uncertain problem (1) is linear but bears a stochastic process parametered by $x,t,\mathbf{v}$, i.e. $u(x,t,\mathbf{v},X)$, as a solution: solving the uncertain counterpart of (1) consequently resumes to solving the SPDE given by

$$\begin{cases} \partial_t u(x,t,\mathbf{v},X)+\mathbf{v}\nabla_x u(x,t,\mathbf{v},X)= & -v\sigma_t(\eta(x,t),\mathbf{v},X)u(x,t,\mathbf{v},X) \\ & \quad +\displaystyle\int v\sigma_s(\eta(x,t),\mathbf{v},\mathbf{v}',X)u(x,t,\mathbf{v}',X)\,\mathrm{d}\mathbf{v}', \\ u(x,0,\mathbf{v},X)=u_0(x,\mathbf{v},X). \end{cases} \tag{3}$$

The above equation is valid for an infinite uncertain medium with uncertain initial conditions. The next discussions can take into account uncertainties in the boundary conditions but for the sake of conciseness we will focus on the above Cauchy problem.

The paper is organized as follows: in section 2, we will recall the most common gPC-based numerical strategy to solve (3), the non-intrusive one. It consists of running a deterministic simulation device to solve (1) at some prescribed (experimental design, see [3,27]) points[3] denoted by $(X_i)_{i\in\{1,\dots,N\}}$ of weights $(w_i)_{i\in\{1,\dots,N\}}$ to project the solution u on a gPC basis before performing some post-treatments to estimate the statistical observables of interest (mean, variance, probability of failure, moments etc.). We briefly present some numerical results in a very simple configuration. We even consider couples of resolution strategies for the black-box code, deterministic counterpart, solving (1) and for the choice of the design points, uncertain counterpart, $(X_i,w_i)_{i\in\{1,\dots,N\}}$. Care will be taken in this section to illustrate the implications of such *decoupled* resolution strategies. In section 3, we focus on one of the resolution methods for the *deterministic* system (1) which deserves particular attention with respect to uncertainties, the Monte-Carlo method. We recall briefly the construction of an MC scheme to solve the *deterministic* linear Boltzmann equation (1) together with its convergence properties. These can be found in many books but briefly recalling them with UQ-friendly notations eases the description of the gPC-i-MC scheme we present in section 4 to solve (3). In particular, it eases the description of the minimal modifications to an existing MC code to do so. It consists in building a new Monte-Carlo scheme allowing the *on-the-fly* resolution of the uncertain counterpart of the linear Boltzmann equation made possible by

- gPC,
- the structure of the transport equation,
- and the mathematical properties of MC methods.

The benefit of the new Monte-Carlo scheme are then illustrated on various examples in section 5 while section 6 is devoted to concluding remarks.

2. Non-intrusive resolution of the uncertain linear Boltzmann equation

In this section, we describe non-intrusive uncertainty propagation methods [9,20,56] in general terms. The description may, at first glance, look like a recipe but it is representative of its practical use. Recall random variable or vector X has probability measure $\mathrm{d}\mathcal{P}_X$. We aim at estimating a given statistical quantity of interest

$$I(u)=\int F(u(x,t,\mathbf{v},\xi))\,\mathrm{d}\mathcal{P}_X(\xi)=\mathbb{E}\left[F(u(x,t,\mathbf{v},X))\right]. \tag{4}$$

The latter depends on the solution u of (3) and on a post-treatment of it, introduced in (4) *via* F. Post-treatment F can either be a (vectorial) functional or an operator. For example,

- If $u\to F(u)=u$ then the statistical quantity I corresponds to the mean of $u(x,t,\mathbf{v},X)$.
- If $u\to F(u)=u^2$ it corresponds to its second order moment.
- If $u\to F(u)=\mathbf{1}_{[U,\infty[}(u)$ then $I(u)$ becomes the probability of having $u(x,t,\mathbf{v},X)$ beyond threshold U, commonly called a failure probability, see [82].
- If $u\to F(u)=u\phi_k^X$ where $(\phi_k^X(X))_{k\in\mathbb{N}}$ denotes the generalized Polynomial Chaos basis (gPC), see [1,33,47,53,63,92–95,102], associated[4] to measure $\mathrm{d}\mathcal{P}_X$ then I is the kth gPC coefficient of u defined by:

[2] At the cost of more or less tedious pretreatments leading to a controlled approximation [60,61,72,81,87] and decorrelation [49,50].
[3] We insist the notation is general enough: if X is a stochastic process, each X_i denotes a realisation of this stochastic process.
[4] I.e. such that $\int \phi_k^X\phi_l^X\,\mathrm{d}\mathcal{P}_X=\delta_{kl}$.

$$I(u)(x,t,\mathbf{v}) = \int u(x,t,\mathbf{v},X)\phi_k^X(X)\,\mathrm{d}\mathcal{P}_X = u_k^X(x,t,\mathbf{v}), \forall k \in \mathbb{N}. \tag{5}$$

In this case, the P-truncated gPC expansion bears some interesting convergence properties [18,97] as

$$u^P(x,t,\mathbf{v},X) = \sum_{k=0}^{P} u_k^X(x,t,\mathbf{v})\phi_k^X(X) \xrightarrow[P\to\infty]{L^2} u(x,t,\mathbf{v},X). \tag{6}$$

We will particularly focus on such developments in the following sections.

– We insist that F can also denote an operator: one may be interested not directly in $u(x,t,\mathbf{v},X)$ but in some post-treatments with respect to the physical variables $x,t,\mathbf{v}$. For example, we can have

$$F(u(\cdot,t,\cdot,X)) = \frac{1}{|\mathcal{D}|\times|\mathcal{F}|}\iint \mathbf{1}_{\mathcal{D}}(x)\mathbf{1}_{\mathcal{F}}(\mathbf{v})u(x,t,\mathbf{v},X)\,\mathrm{d}x\,\mathrm{d}\mathbf{v}, \tag{7}$$

where $|\mathcal{D}|$, $|\mathcal{F}|$ denote the volumes of spatial and kinetic spaces $\mathcal{D}\subset\mathbb{R}^3$ and $\mathcal{F}\subset\mathbb{R}^3$. In this case, we want to especially emphasize that it is possible to apply the material of the previous point to

$$F^P(t,X) = \sum_{k=0}^{P} F_k^X(t)\phi_k^X(X) \longrightarrow F(u(\cdot,t,\cdot,X)), \tag{8}$$

in order to approximate $F(u(\cdot,t,\cdot,X))$ on a gPC basis. In other words, the next discussion remains very general with respect to the observable of interest, even if we basically focus on u in the following numerical examples.

The non-intrusive methodology then consists of several steps:

(i) The first one corresponds to the discretisation of the random variable and its probability measure $(X,\mathrm{d}\mathcal{P}_X)$ by a numerical integration method with N points:

$$(X,\mathrm{d}\mathcal{P}_X) \approx (X_i,w_i)_{i\in\{1,\dots,N\}}. \tag{9}$$

The notation (9) for the punctual discretisation of $(X,\mathrm{d}\mathcal{P}_X)$ is very general and convenient as it can take into account many integration methods: suppose the points $(X_i)_{i\in\{1,\dots,N\}}$ are chosen sampled from the probability law of X and $(w_i=\frac{1}{N})_{i\in\{1,\dots,N\}}$, then it corresponds to a Monte-Carlo integration [76] for the estimation of I. With the same writing, we can conveniently consider Gauss quadrature points, Latin Hypercube Samples, Sparse Grids etc. [17,29,36,64,76,78,88,90]. The latter sets of points in dimension Q differ only by their asymptotic error analysis $\mathcal{O}(N^{\beta(Q)})$, i.e. the weak or strong dependence of their convergence rates $\beta(Q)$ with respect to the number of uncertain parameters Q. For example $\beta(Q)=\beta=-\frac{1}{2}$ for MC methods: the convergence rate is slow but independent of Q. We have $\beta(Q)=-1$ for N^Q equidistant points in $[-1,1]^Q$ for example. It is well-known in $1D$ stochastic dimension, equidistant points ($\beta=-1$) outperform an MC resolution ($\beta=-\frac{1}{2}$). But to obtain a given accuracy, the number N^Q of equidistant points grows exponentially fast with the dimension whereas it does not for MC methods. There exists some intermediary solution and alternatives, we refer to [17,29,36,64,76,78,88,90] and the references therein for the interested reader.

(ii) The next step consists in running N independent runs of a black-box code (in this paper it refers to the resolution of (1)) at the *a priori* chosen points $(X_i,w_i)_{i\in\{1,\dots,N\}}$ and gathering a new collection of output points: $(u(x,t,\mathbf{v},X_i),w_i)_{i\in\{1,\dots,N\}}$. It is supposed to support the main computational effort as (1) must be solved N times. The N runs are independent and their resolutions can consequently be carried on in parallel (it is often called an *embarrassingly parallel* strategy as there are no communication costs between processes, except during the post-treatment), simultaneously if one has access to as many processors as runs (N).

Equation (1) is solved up to a certain accuracy depending on its numerical solver thanks to a black-box code.[5] Let us denote by Δ the discretisation parameter of the latter, then for a first order resolution method with respect to Δ we have

$$u(x,t,\mathbf{v},X_i) = u^{BB}(x,t,\mathbf{v},X_i) + K^{BB}(x,t,\mathbf{v},X_i)\Delta + \mathcal{O}(\Delta^2). \tag{10}$$

Once again, the notation in (10) is very general. For example, if (1) is solved by an MC method, $\Delta=\frac{1}{\sqrt{N_{MC}}}$ where N_{MC} is the number of MC particles[6] and K^{BB} is the standard deviation of the process (error estimator) see [45,51]. If (1) is solved with a deterministic scheme, $\Delta=\max(\Delta x,\Delta t,\Delta v)$. High-order schemes (see [28,41] for examples) aim at canceling K^{BB} and the higher coefficients of powers of Δ. Rigorously speaking, (10) comes from some numerical analysis and implies choosing a norm, defining a space for the solution etc. but assuming a general form such as (10) will ease the later discussions.

[5] Exponant ^{BB} appears every time we are considering a numerical approximation obtained from the black-box (BB) code.

[6] Those MC particles will be defined in section 3.

(iii) Once the N runs are obtained, the rest is merely postprocessing at the observation points of interest. The estimation of $I(u)$ is mainly made by numerical integration and we have

$$\begin{aligned} I(u) &= \int_N F(u(x,t,\mathbf{v},X))\,\mathrm{d}\mathcal{P}_X, \\ &= \sum_{i=1}^{N} F(u(x,t,\mathbf{v},X_i))w_i + \mathcal{O}(N^{\beta}), \\ &= \sum_{i=1}^{N} F(u^{BB}(x,t,\mathbf{v},X_i))w_i + \mathcal{O}(\Delta) + \mathcal{O}(N^{\beta}), \\ &= I_N^{\Delta} + \mathcal{O}(\Delta) + \mathcal{O}(N^{\beta}). \end{aligned} \tag{11}$$

At the end of the process, one has access to an approximation I_N^{Δ}. The error between I and I_N^{Δ} can then be decomposed in two main parts:

$$||I - I_N^{\Delta}|| = \underbrace{\mathcal{O}(N^{\beta})}_{\text{integration error (UQ)}} + \underbrace{\mathcal{O}(\Delta)}_{\text{numerical error (BB)}}. \tag{12}$$

In (12), the error of the non-intrusive approximation have (explicitly) two parameters, N for the integration error which is relative to the resolution of the uncertainty propagation counterpart and Δ relative to the resolution of each deterministic run. Basically, if the notation $||\cdot||$ in (12) denotes the L^2 one, quantity (12) expresses an error on the variance of the observable I: if $\Delta \gg N^{\beta}$, the estimated variance of I is closer to a numerical error than a variability due to the uncertain parameters. In other words, to perform an uncertainty analysis, one must make sure $\Delta \ll N^{\beta}$. The next figure illustrates this fact. Fig. 1 presents some convergence curves in a very simple configuration for two different couples of numerical methods:

- The two graphics on top of Fig. 1 presents convergence studies obtained with a deterministic scheme of parameter[7] $\Delta = \Delta t$ for the resolution of the black-box code and N Gauss-Legendre (denoted by N_{GL}) points for the uncertainty propagation. The top left picture presents a convergence study with respect to $\Delta = \Delta t$ for fixed values of $N_{GL} = 2, 4, 8$. The right one displays a convergence study with respect to $N = N_{GL}$ for fixed values of $\Delta = \Delta t$.
- The second line shows the same convergence studies except the deterministic black-box code is solved by a stochastic (MC) scheme, described section 3, of discretisation parameter $\Delta = \frac{1}{\sqrt{N_{MC}}}$, the uncertain counterpart being also solved with an MC sampling of $(X, \mathrm{d}\mathcal{P}_X)$ with $N = N_{UQ}^{MC}$ points. The bottom left picture presents a convergence study with respect to $\Delta = N_{MC}$ for fixed values of $N = N_{UQ}^{MC} = 100, 1000$. The right one displays a convergence study with respect to $N = N_{UQ}^{MC}$ for fixed values of $\Delta = N_{MC}$.

The quantity of interest here is the variance (see (A.3) in appendix A) of the total amount of particles in a closed box $x \in \mathcal{D}, v \in \mathbb{R}^3$. Monitoring the error on any different statistical quantity would give similar results at the price of more tedious calculations in order to compute the reference solution: the configuration of interest, described in appendix A, ensures we have access to an analytical solution for $\mathbb{V}[I](t)$.

Let us comment on Fig. 1. Independently of the couple of numerical methods to solve the uncertain and the deterministic counterparts, the behaviors are very similar: every curve presents, first, a converging behavior with a slope characteristic of the numerical method applied $\mathcal{O}(\Delta t^1)$ Fig. 1 (top left), $\mathcal{O}(N^{\beta_{GL}})$ Fig. 1 (top right), $\mathcal{O}(N_{MC}^{-\frac{1}{2}})$ Fig. 1 (bottom left) and $\mathcal{O}(N^{-\frac{1}{2}})$ Fig. 1 (bottom right). Then, the curves present a, more or less pronounced, kink: a change of slope, often followed by a plateau, a stagnation of the accuracy. It corresponds to the point where the general accuracy becomes driven by the *coarser* numerical method. Refining the numerical solver relative to the $x-$axis does not allow any significant gain. In a sense, the locations of the kinks corresponds to optimal parameter choices (Δ, N): increasing the accuracy in one direction without the other induces a loss of computational time. Looking for this optimal set of parameters for efficiency can be complex and is not the purpose of this paper.

Of course, the application of gPC implies an additional parameter P which is the truncation order of the polynomial approximation

$$u(x,t,\mathbf{v},X) = \sum_{k=0}^{\infty} u_k^X(x,t,\mathbf{v})\phi_k^X(X) \approx u^P(x,t,\mathbf{v},X) = \sum_{k=0}^{P} u_k^X(x,t,\mathbf{v})\phi_k^X(X). \tag{13}$$

This parameter has been intensionally omitted in the previous studies because it will remain common to the new gPC-i-MC scheme we will describe in the forthcoming sections. In (13), every gPC coefficient is approximated up to an $\mathcal{O}(\Delta) +$

[7] Here, the configuration is such that $\Delta = \max(\Delta x, \Delta t, \Delta v) = \Delta t$, see appendix A.

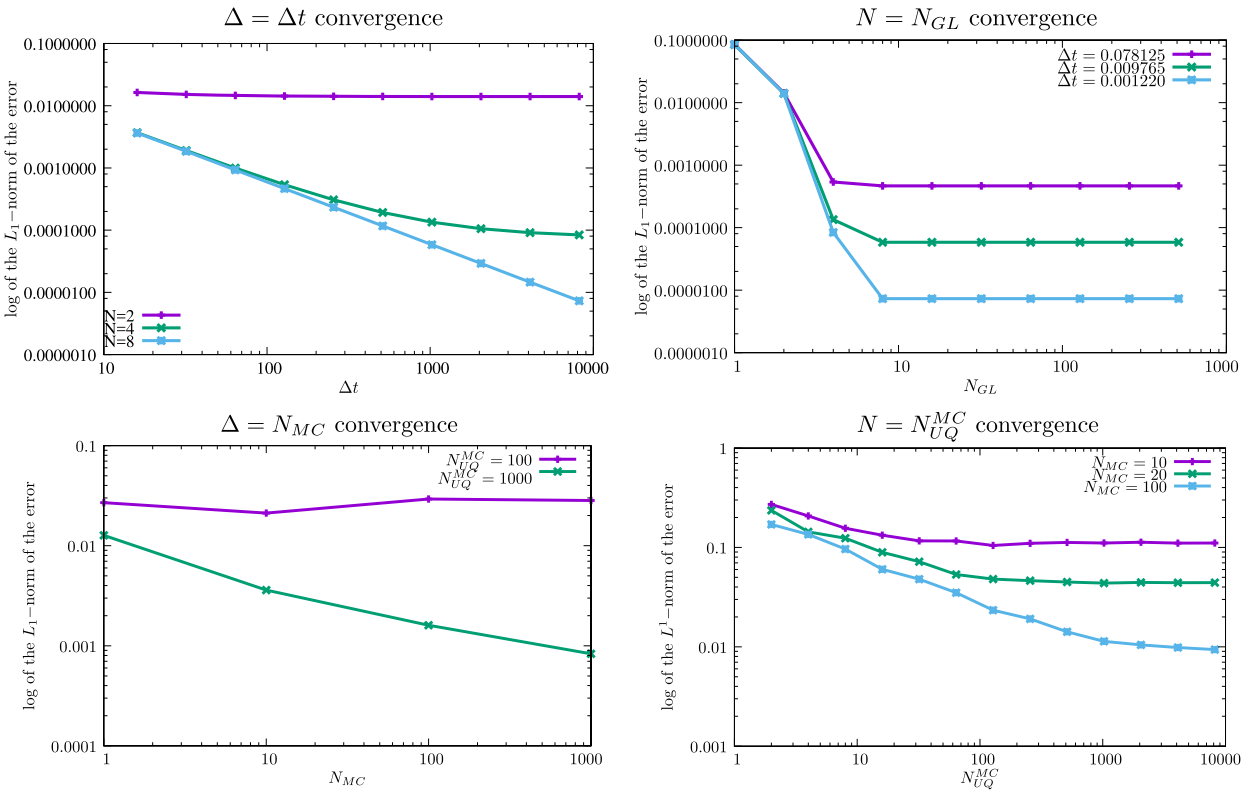

Fig. 1. Convergence studies with respect to N and Δ for two couples of numerical methods for the resolution of the uncertain linear Boltzmann equation in a homogeneous configuration.

$\mathcal{O}(N^{\beta(Q)})$ accuracy as in (11). In this paper, we intend to show that if the black-box code solves (1) with an MC scheme (i.e. $\Delta = \frac{1}{\sqrt{N_{MC}}}$) then it can be enriched (intrusively) at a relatively low cost to compute *on-the-fly* the gPC coefficients (of u as in (13) or of any general F as in (8)) during the MC resolution. Intuitively, it is easy to notice that when using a non-intrusive gPC (ni-gPC) method on an MC black-box code, basically, $N \times N_{MC}$ MC particles are treated for an overall $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ accuracy (as in the bottom left picture of Fig. 1). Such *tensorisation* of the MC particles with the experimental design related to the uncertain parameters can be avoided at the cost of minimal modifications to an existing MC solver. In order to identify accurately those modifications, we suggest briefly and formally recalling how an MC solver for (1) is built in the next section 3 and enrich the MC scheme for uncertainties in section 4.

3. The Monte-Carlo resolution of the linear Boltzmann equation (1)

In this section, we recall the construction of the semi-analog[8] MC scheme to solve (1) and the general structure of an MC code. This MC strategy is commonly used in neutronic applications and is called *implicit capture* [45]. The aim is to very concisely present the different steps of the construction as well as the algorithmic implications. This way, it will be easier, in the next section, to highlight the similarities and differences of the resolution schemes once uncertainties/gPC are introduced. Here, we describe a *backward* resolution of the transport equation [45,66] because it saves some calculations and remains very general. We refer to [66] for the reader interested in its forward resolution (it consists in applying the very same steps as above but on the adjoint version of (14)).

The transport equation with deterministic collisional counterpart in infinite medium is given by

$$\partial_t u(x,t,\mathbf{v}) + \mathbf{v}\cdot\nabla u(x,t,\mathbf{v}) + v\sigma_t(x,t,\mathbf{v})u(x,t,\mathbf{v}) = v\sigma_s(x,t,\mathbf{v})\int P(x,t,\mathbf{v},\mathbf{v}')u(x,t,\mathbf{v}')\,d\mathbf{v}'. \tag{14}$$

In (14), we introduced

[8] The next description can easily be generalized and applied to the analog or the non-analog MC scheme.

$$\sigma_s(x,t,\mathbf{v}) = \int \sigma_s(x,t,\mathbf{v},\mathbf{v}')\,\mathrm{d}\mathbf{v}',$$
$$P(x,t,\mathbf{v},\mathbf{v}') = \frac{\sigma_s(x,t,\mathbf{v},\mathbf{v}')}{\sigma_s(x,t,\mathbf{v})}.$$

Let us now rewrite (14) in recursive integral form: to do so, we apply the characteristic method, multiply (14) by $\exp(\int_0^s v\sigma_t(x+\mathbf{v}\alpha,\alpha,v)\,\mathrm{d}\alpha)$ and integrate the expression between 0 and t to obtain

$$\begin{aligned} u(x,t,\mathbf{v}) &= u_0(x-\mathbf{v}t,\mathbf{v})\exp\left(-\int_0^t v\sigma_t(x-\mathbf{v}(t-\alpha),\alpha,\mathbf{v})\,\mathrm{d}\alpha\right) \\ &+ \iint_0^t v\sigma_s(x-\mathbf{v}(t-s),s,\mathbf{v})u(x-\mathbf{v}(t-s),s,\mathbf{v}')e^{-\int_s^t v\sigma_t(x-v\mathbf{v}(t-\alpha),\mathbf{v})\,\mathrm{d}\alpha}P(x-\mathbf{v}(t-s),s,\mathbf{v},\mathbf{v}')\,\mathrm{d}\mathbf{v}'\,\mathrm{d}s. \end{aligned} \tag{15}$$

With (15), the transport equation is written in a recursive integral form for an infinite medium [16,35,51,80]. In order to alleviate the notations in (15), let us introduce

$$z_s = \{x-\mathbf{v}s, t-s, \mathbf{v}\} \text{ and } z_s' = \{x-\mathbf{v}s, t-s, \mathbf{v}'\}. \tag{16}$$

Equivalently, equation (15) becomes

$$u(z_0) = u_0(z_t)\exp\left(-\int_0^t v\sigma_t(z_{t-\alpha})\,\mathrm{d}\alpha\right) + \iint_0^t v\sigma_s(z_{t-s})u(z'_{t-s})e^{-\int_s^t v\sigma_t(z_{t-\alpha})\,\mathrm{d}\alpha}P(z_{t-s},\mathbf{v}')\,\mathrm{d}\mathbf{v}'\,\mathrm{d}s. \tag{17}$$

Few calculations on the term in factor of $u_0(z_t)$ leads to

$$\begin{aligned} \exp\left(-\int_0^t v\sigma_t(z_{t-s})\,\mathrm{d}s\right) &= \exp\left(-\int_0^t v\sigma_t(z_\alpha)\,\mathrm{d}\alpha\right), \\ &= \int_t^\infty \mathbf{1}_{[0,\infty[}(s)v\sigma_t(z_s)\exp\left(-\int_0^s v\sigma_t(z_\alpha)\,\mathrm{d}\alpha\right)\mathrm{d}s, \\ &= \int_t^\infty f(z_s)\,\mathrm{d}s. \end{aligned} \tag{18}$$

In the above expression, we introduced the probability measure, see [66],

$$f(z_s)\,\mathrm{d}s = \mathbf{1}_{[0,\infty[}(s)v\sigma_t(z_s)\exp\left(-\int_0^s v\sigma_t(z_\alpha)\,\mathrm{d}\alpha\right)\mathrm{d}s. \tag{19}$$

Equation (17) becomes

$$u(z_0) = \int\left(\mathbf{1}_{[t,\infty[}(s)u_0(z_t) + \mathbf{1}_{[0,t[}(s)\int u(z'_s)P(z_s,\mathbf{v}')\frac{\sigma_s(z_s)}{\sigma_t(z_s)}\,\mathrm{d}\mathbf{v}'\right)f(z_s)ds. \tag{20}$$

The above equation is nothing more than the backward form of the integral transport equation for an infinite media, see [51,80]. Let's now introduce $\tau,\mathcal{V}$ sampled from the probability measures $\tau\sim f(z_s)\,\mathrm{d}s$ and $\mathcal{V}\sim P(z_s,\mathbf{v}')\,\mathrm{d}\mathbf{v}'$. We can rewrite the integral equation (we go back to the native variables $x,t,\mathbf{v}$) in term of a recursive expectation over these two random variables

$$u(x,t,\mathbf{v}) = \mathbb{E}\left[\mathbf{1}_{[t,\infty[}(\tau)u_0(x-\mathbf{v}t,\mathbf{v}) + \mathbf{1}_{[0,t]}(\tau)u(x-\mathbf{v}\tau,t-\tau,\mathcal{V})\frac{\sigma_s(x-\mathbf{v}\tau,t-\tau,\mathbf{v})}{\sigma_t(x-\mathbf{v}\tau,t-\tau,\mathbf{v})}\right]. \tag{21}$$

The next step consists in introducing an MC discretization. Formally, the construction of an MC scheme relies on looking for solutions of (21) having the particular forms

$$u_p(x,t,\mathbf{v}) = w_p(t)\delta_x(x_p(t))\delta_{\mathbf{v}}(\mathbf{v}_p(t)). \tag{22}$$

Such solution u_p will be commonly called the 'MC particle' p. The MC scheme intensively uses the linearity of equation (14): if $(u_p)_{p\in\{1,\dots,N_{MC}\}}$ are independent solutions of (14) then $\sum_{p=1}^{N_{MC}}u_p$ is also solution of (14). Plugging (22) into (21) finally leads to the recursive (backward, see [66]) treatment

$$\begin{cases} w_p(t) = \mathbf{1}_{[t,\infty[}(s) w_p(0) & + \mathbf{1}_{[0,t]}(s) \dfrac{\sigma_s}{\sigma_t}(x_p(t-s), t-s, \mathbf{v}_p(t-s)) w_p(t-s), \\ x_p(t) = \mathbf{1}_{[t,\infty[}(s)(x - \mathbf{v}t) & + \mathbf{1}_{[0,t]}(s)(x - \mathbf{v}s), \\ \mathbf{v}_p(t) = \mathbf{1}_{[t,\infty[}(s)\mathbf{v} & + \mathbf{1}_{[0,t]}(s)\mathcal{V}. \end{cases} \tag{23}$$

In expression (23), we recognize the classical operations one must apply to an MC particle to solve (14):

- the 'census event' corresponds to the condition $\tau \in [t, \infty[$: the MC particle is transported along the straight line between $x(0)$ and $x(t) = x(0) - \mathbf{v}t$ with no change in its attributes (same weight $w_p(t) = w_p(0)$, same velocity $\mathbf{v}_p(t) = \mathbf{v}_p(0) = \mathbf{v}$).
- the 'scattering event' corresponds to the condition $\tau \in [0, t[$: the MC particle is transported along a straight line between $x(t-\tau)$ and $x(t-\tau) + \mathbf{v}\tau$ together with a modification of its weight according to

$$w_p(t) = \frac{\sigma_s}{\sigma_t}(x_p(t-\tau), t-\tau, \mathbf{v}_p(t-\tau)) w_p(t-\tau). \tag{24}$$

  It also implies a change of velocity from $\mathcal{V}$ to $\mathbf{v}$ at the interaction time $t - \tau$ and position $x_{t-\tau}$.
- Of course, if a grid is introduced, a 'cell exit' event is usually introduced. It is not detailed here because it is easy to deal with (based on the memorylessness of the exponential distribution for the interaction time, see [45,66]), even with the new MC scheme we present later on.
- Concerning eventual boundary conditions (see [16,35,51,80]), they have to be applied to any MC particle reaching the boundary $\partial\mathcal{D}$ of the resolution domain $\mathcal{D}$.

By construction of the MC resolution scheme, theorem 3.2.1 of [45] ensures the narrow convergence of the MC solver toward the solution of (14) in the limit $N_{MC} \to \infty$ for the considered time step $[0, t]$. The central limit theorem ensures its $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ convergence rate. It does not need a mesh or any tessellation of the space if the cross-sections are analytically known. In practice, only the material interfaces are projected on a grid and cross-sections are constant per cell and time step. In this case, the operations to perform on an MC particle u_p have much friendlier expressions:

- the interaction time is sampled from an exponential law of parameter $v_p \sigma_t(v_p)$, i.e. we have

$$\tau = -\frac{\ln(\mathcal{U}_\tau)}{v_p \sigma_t(\mathbf{v}_p)} \text{ where } \mathcal{U}_\tau \sim \mathcal{U}([0,1]) \text{ and } \mathbf{v}_p \text{ is an embedded particle field.} \tag{25}$$

  In the above expression and throughout the document, notations $\mathcal{U}_\tau \sim \mathcal{U}([0,1])$ means the random value $\mathcal{U}_\tau$ follows ($\sim$) an uniform distribution on $[0,1]$ ($\mathcal{U}([0,1])$). The latter expression has been obtained inversing the cumulative density function of the aforementioned exponential law. This is classical in MC simulations, see [45,51,80].
- The outer velocity $\mathcal{V}$ is also obtained locally inversing the cumulative density function of $P(x, t, \mathbf{v}, \mathbf{v}')\,\mathrm{d}\mathbf{v}'$. This means we have

$$\mathcal{U}_\mathcal{V} = \int_{-\infty}^{\mathcal{V}} P(x_p(t-\tau), t-\tau, \mathbf{v}_p(t-\tau), \mathbf{v}')\,\mathrm{d}\mathbf{v}' \text{ where } \mathcal{U}_\mathcal{V} \sim \mathcal{U}([0,1]), \tag{26}$$

  where $x_p, \mathbf{v}_p$ are embedded particle fields and τ is the current sampled interaction time (the change of velocity occurs only at a collision point/time). Equation (26) may appear difficult to comprehend (for example $\mathcal{V}$ is a vector). Here, we intentionally keep the expression implicit as an expression of $\mathcal{V}$ resulting from the inversion of (26) strongly depends on the format of the cross-sections (multigroup, continuous [45,51,80]). Some comments are provided in appendix B. The general expression (26) mainly allows us to insist on the fact that the material of this paper remains independent of such considerations.

Algorithm 1 in appendix C present the general canvas of an MC resolution code to solve (1) (boundary treatments are also highlighted). The latter will be useful to highlight the few modifications needed to take into account uncertainties *on-the-fly* during the MC computations.

4. The Monte-Carlo resolution of the uncertain linear Boltzmann equation (3)

As explained earlier, applying ni-gPC to an MC code solving the uncertain linear Boltzmann equation leads to a tensorisation of the N experimental design points with the N_{MC} MC particles. Computational resources are lost in the sense the overall accuracy remains, see Fig. 1, $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ with a cost being $\mathcal{O}(N \times N_{MC})$. On another hand, MC schemes are computationally insensitive to an increase in dimension. The idea of this section is to explain how one can make the most of the MC resolution of the deterministic linear Boltzmann equation to take into account $3(x) + 1(t) + 3(\mathbf{v}) + Q\,(X) = 7 + Q$ dimensions. The idea is to treat, *on-the-fly* during the MC resolution, the effect of uncertainties on the particle flow. Basically, we would like to keep the accuracy $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ with a cost being $\mathcal{O}(N_{MC})$ as for any MC method.

For this, we must build a new MC scheme. We suggest going through the same steps as in the next section and emphasize its subtleties progressively.[9] The transport equation with uncertain collisional counterpart for an infinite medium is given by

$$\begin{aligned}\partial_t u(x,t,\mathbf{v},X)+\mathbf{v}\cdot\nabla u(x,t,\mathbf{v},X) = & -v\sigma_t(x,t,\mathbf{v},X)u(x,t,\mathbf{v},X)\\ & +v\sigma_s(x,t,\mathbf{v},X)\int P(x,t,\mathbf{v},\mathbf{v}',X)u(x,t,\mathbf{v}',X)\,\mathrm{d}\mathbf{v}',\end{aligned} \tag{27}$$

with

$$\begin{aligned}\sigma_s(x,t,\mathbf{v},X) &= \int \sigma_s(x,t,\mathbf{v},\mathbf{v}',X)\,\mathrm{d}\mathbf{v}',\\ P(x,t,\mathbf{v},\mathbf{v}',X) &= \frac{\sigma_s(x,t,\mathbf{v},\mathbf{v}',X)}{\sigma_s(x,t,\mathbf{v},X)}.\end{aligned} \tag{28}$$

The idea is to go through the same steps as before but having in mind that the quantities depend also on X. We aim at identifying the changes one must perform to the samplings of section 3 to take X into account. Note that to alleviate the computation, we rely once again on the more concise notations (16). Let us first introduce

$$f(z_s,X)\,\mathrm{d}s = \mathbf{1}_{[0,\infty[}(s)v\sigma_t(z_s,X)\exp\left(-\int_0^s v\sigma_t(z_\alpha,X)\,\mathrm{d}\alpha\right)\mathrm{d}s. \tag{29}$$

Under some boundedness conditions $\forall X\in Supp(X)$, where $Supp(X)$ denotes the support of the random variable, (29) remains an exponential probability measure [66]. Here, the hypothesis $\forall X\in SuppX$ is certainly not optimal but sufficient for the property to hold. The uncertain counterpart of (20) is then given by

$$u(z_0,X) = \int\left(\mathbf{1}_{[t,\infty[}(s)u_0(z_t,X)+\mathbf{1}_{[0,t[}(s)\int u(z_s',X)P(z_s,\mathbf{v}',X)\frac{\sigma_s(z_\alpha,X)}{\sigma_t(z_\alpha,X)}\,\mathrm{d}\mathbf{v}'\right)f(z_s,X)ds. \tag{30}$$

Introduce the set of random variables $\tau_X,\mathcal{V}_X$ sampled from the probability measures $\tau_X\sim f(z_s,X)\,\mathrm{d}s$ and $\mathcal{V}_X\sim P(z_s,\mathbf{v}',X)\,\mathrm{d}\mathbf{v}'$, then the above integral equation rewritten as a recursive expectation over the native set of variables becomes

$$\begin{aligned}&u(x,t,\mathbf{v},X) =\\ &\quad\mathbb{E}\left[\mathbf{1}_{[t,\infty[}(\tau_X)u_0(x-\mathbf{v}t,\mathbf{v},X)+\mathbf{1}_{[0,t]}(\tau_X)u(x-\mathbf{v}\tau_X,t-\tau_X,\mathcal{V}_X,X)\frac{\sigma_s(x-\mathbf{v}\tau_X,t-\tau_X,\mathbf{v},X)}{\sigma_t(x-\mathbf{v}\tau_X,t-\tau_X,\mathbf{v},X)}\right].\end{aligned} \tag{31}$$

The next step consists in introducing an MC discretization allowing to take into account the uncertain variables. Let us introduce an 'uncertain MC particle' u_p defined as

$$u_p(x,t,\mathbf{v},X) = u_p(x,t,\mathbf{v})\delta_X(X_p(t)) = w_p(t)\delta_x(x_p(t))\delta_{\mathbf{v}}(\mathbf{v}_p(t))\delta_X(X_p(t)). \tag{32}$$

We are now going to identify the operations we must perform to ensure (32) is solution of (27). For this, we plug (32) into (31) and make sure $u_p(x,t,\mathbf{v},X)$ is a particular solution of (27). Plugging u_p into (31) leads to the construction of a (compatible) system of equations of unknowns $w_p(t),x_p(t),\mathbf{v}_p(t),X_p(t)$ given by

$$\begin{cases} w_p(t) = \mathbf{1}_{[t,\infty[}(\tau_X)\ w_p(0) & +\mathbf{1}_{[0,t]}(\tau_X)\dfrac{\sigma_s(x_p(t-\tau_X),t-\tau_X,\mathbf{v}_p(t-\tau_X),X_p(t-\tau_X))}{\sigma_t(x_p(t-\tau_X),t-\tau_X,\mathbf{v}_p(t-\tau_X),X_p(t-\tau_X))}w_p(t-\tau_X),\\ x_p(t) = \mathbf{1}_{[t,\infty[}(\tau_X)\ (x(0)-\mathbf{v}t) & +\mathbf{1}_{[0,t]}(\tau_X)(x_p(t-\tau_X)-\mathbf{v}\tau_X),\\ \mathbf{v}_p(t) = \mathbf{1}_{[t,\infty[}(\tau_X)\ \mathbf{v} & +\mathbf{1}_{[0,t]}(\tau_X)\mathcal{V}_X,\\ X_p(t) = \mathbf{1}_{[t,\infty[}(\tau_X)\ X_p(0) & +\mathbf{1}_{[0,t]}(\tau_X)X_p(t-\tau_X).\end{cases} \tag{33}$$

Let us focus on the last equation: inconditionally with respect to time t, $X_p(t)$ is not modified. Indeed, if $\tau_X<t$ we have $X_p(t)=X_p(t-\tau_X)$ until, events after events, the initial condition is reached leading to $X_p(t)=X_p(0)=X_p$.

Remark 4.1. The latter result tells that the uncertain variable must be sampled initially for every MC particle and remain unchanged. It also implies an MC particle must transport amongst its attributes the realisation of a random vector of size Q. This has some impact on the memory consumption of the algorithm.

[9] Note that we describe an MC resolution scheme for uncertainty based on the semi-analog (implicit capture) one. The next description can easily be generalized and applied to the analog or the non-analog MC scheme.

Now that we know that $X_p(t) = X_p$, equation (33) reduces to

$$\begin{cases} w_p(t) = \mathbf{1}_{[t,\infty[}(s)\ w_p(0) & +\mathbf{1}_{[0,t]}(s)\dfrac{\sigma_s(x_p(t-s),t-s,\mathbf{v}_p(t-s),X_p)}{\sigma_t(x_p(t-s),t-s,\mathbf{v}_p(t-s),X_p)}w_p(t-s), \\ x_p(t) = \mathbf{1}_{[t,\infty[}(s)\ (x(0)-\mathbf{v}t) & +\mathbf{1}_{[0,t]}(s)(x_p(t-s)-\mathbf{v}s), \\ \mathbf{v}_p(t) = \mathbf{1}_{[t,\infty[}(s)\ \mathbf{v} & +\mathbf{1}_{[0,t]}(s)\mathcal{V}_{X_p}, \end{cases} \tag{34}$$

where, we recall $\tau_{X_p}, \mathcal{V}_{X_p}$ are sampled from the probability measures $\tau_{X_p} \sim f(z_s, X_p)\,\mathrm{d}s$ and $\mathcal{V}_{X_p} \sim P(z_s, \mathbf{v}', X_p)\,\mathrm{d}\mathbf{v}'$. System (34) is very similar to system (23) but the samplings depending on X_p may need few more details: assume, for the sake of simplicity, the cross-sections do not depend on x, t locally (i.e. within a cell or an element of geometry). Then the probability measure (29) for the sampling of the interaction time resumes to

$$f(s,\mathbf{v}_p,X_p)\,\mathrm{d}s = \mathbf{1}_{[0,\infty[}(s)\nu_p\sigma_t(\mathbf{v}_p,X_p)e^{-\nu_p\sigma_t(\mathbf{v}_p,X_p)s}\,\mathrm{d}s. \tag{35}$$

In practice, this implies sampling τ_{X_p} according to

$$\tau_{X_p} = -\frac{\ln(\mathcal{U})}{\nu_p\sigma_t(\mathbf{v}_p,X_p)} \text{ where } \mathcal{U}\sim\mathcal{U}([0,1]) \text{ and } X_p \text{ is an embedded particle field just as } \mathbf{v}_p. \tag{36}$$

Expression (36) echoes (25). The same apply to the sampling of the outer[10] velocity $\mathcal{V}_{X_p}$ and to the weight modification of the uncertain MC particles: the cross-sections at play in (28)–(34) must be used at both the physical $(x_p(t), \mathbf{v}_p(t))$ and uncertain (X_p) fields of the uncertain MC particle.

Now the gPC coefficients can easily be estimated thanks to the uncertain MC particles: the scheme once again intensively uses the linearity of equation (27) together with the linearity of the $P-$truncated gPC approximation defined *via* (5)–(6). Indeed, $(u_p(x,t,\mathbf{v},X)\phi_k^X(X))_{p\in\{1,\dots,N_{MC}\}}, \forall k\in\{0,..,P\}$ are independent solutions of the projection of the solution of (27) onto a $P-$ truncated gPC basis. This implies the sum over the number of uncertain MC particles verifies $\forall k\in\{0,..,P\}$

$$\sum_{p=1}^{N_{MC}} u_p(x,t,\mathbf{v},X)\phi_k^X(X) \approx u_k^X(x,t,\mathbf{v}).$$

Applying the operations related to (33) to any given uncertain MC particles ensures, by construction (see theorem 3.2.1 of [45]), the convergence of the MC solver toward the projection of the solution of (27) onto the truncated gPC basis in the limit $N_{MC}\to\infty$. The overall cost remains $\mathcal{O}(N_{MC})$ together with an $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ accuracy on the gPC coefficients to compute. Note that the computation of the gPC coefficients explicitly appears in the algorithmic presentation 2 (uncertain boundary treatments are also hinted at) of the new MC scheme in appendix D together with an exhaustive description and a discussion on parallel strategies.

From the previous description, one must understand that the basic idea is to try to avoid a tensorisation between the N experimental design points concerning the random variable X and the N_{MC} samplings related to the physical variables $(x,t,\mathbf{v})$ for the MC particles. We want to build a full MC approximation of the gPC coefficient with N_{MC} sampling in the whole space of variables $(x,t,\mathbf{v},X)$. We intensively make use of the insensitiveness of an MC integration with respect to dimension to compute the gPC coefficient of any given output of interest. The new MC scheme is intrusive in the sense one must modify[11]

- the attributes of the MC particles to take into account a discretisation $(X_p)_{p\in\{1,\dots,N_{MC}\}}$ of $(X, \mathrm{d}\mathcal{P}_X)$,
- the call to the cross-sections at those points $(X_p)_{p\in\{1,\dots,N_{MC}\}}$ to sample the interaction time, the outer velocity and modify the weight of any uncertain MC particle,
- the tallies (to embed the computations of the gPC coefficients and other outputs of interest).

The last point may deserve few more details: to approximate any output of interest of the form (4), the post-treatment must also be embedded in the MC resolution. Any other quantity of interest will not directly be available unless every field of the uncertain MC particles is *stored in some files to be post-treated*. We want to avoid such solution because tracking down information with such frequency (many tallies[12] of MC particles per seconds leading to a very important volume of Input/Output) slows drastically down the computations. It can even easily make a filesystem collapse. More details will be given in the numerical examples of the next section and in the description of Algorithm 2.

At this stage of the discussion, one may also wonder why relying on gPC and consequently remaining sensitive to the dimension Q *via* the increasing number of coefficient $(u_k^X)_{k\in\{0,..,P\}}$ to be evaluated. In other words: why relying on an MC

[10] Inner would be more appropriate as we are here identifying the backward samplings.
[11] This is easier to understand thanks to the algorithmic representations of appendices C and D.
[12] See Algorithm 1 for the definition of tallying.

scheme to approximate the gPC coefficients rather than relying on an MC scheme to approximate directly the statistical observable (variance etc.) of interest? To give an element of answer, let us build the PDE satisfied by the moment of order 2 of u, solution of (3). It is defined by

$$\begin{aligned} M_2(x,t,\mathbf{v}) &= \int u^2(x,t,\mathbf{v},X)\,\mathrm{d}\mathcal{P}_X, \\ &= \int m_2(x,t,\mathbf{v},X)\,\mathrm{d}\mathcal{P}_X. \end{aligned}$$

It certainly corresponds to one of the simplest statistical observable (the variance is then $\mathbb{V}[U] = M_2 - M_1^2$). In this case, quantity m_2 is solution of

$$\begin{aligned} \partial_t m_2(x,t,\mathbf{v},X) + \mathbf{v}\cdot\nabla m_2(x,t,\mathbf{v},X) = &-2v\sigma_t(x,t,\mathbf{v},X)m_2(x,t,\mathbf{v},X) \\ &+2u(x,t,\mathbf{v},X)\int v\sigma_s(x,t,\mathbf{v},\mathbf{v}',X)u(x,t,\mathbf{v}',X)\,\mathrm{d}\mathbf{v}'. \end{aligned} \tag{37}$$

To build the above equation it is enough multiplying (27) by $u(x,t,\mathbf{v},X)$. Equation (37) is nonlinear (see the scattering term), as a consequence, the difficulty to solve (37) with an MC method can be compared to the one to solve the quadratic Boltzmann equation [6,12] for example. In other words, to be solved numerically, it may

- either need an additional linearisation hypothesis. For example, for a Nanbu-like [12] resolution, this implies relying on a time step discretisation and an MC resolution of the explicited equation

$$\begin{aligned} \partial_t m_2(x,t,\mathbf{v},X) + \mathbf{v}\cdot\nabla m_2(x,t,\mathbf{v},X) = &-2v\sigma_t(x,t,\mathbf{v},X)m_2(x,t,\mathbf{v},X) \\ &+2u(x,t^n,\mathbf{v},X)\int v\sigma_s(x,t,\mathbf{v},\mathbf{v}',X)u(x,t,\mathbf{v}',X)\,\mathrm{d}\mathbf{v}', \end{aligned} \tag{38}$$

  where $u(x,t^n,\mathbf{v},X)$ is the approximated solution at the beginning of the time step Δt. In other words the convergence depends on N_{MC} but also on the time step Δt.
- or perform a splitting of operator between the streaming part and the collisional one with an adaptation of Bird's algorithm [6]. This splitting, even if having very good mathematical properties (conservations), also introduces a dependence with respect to a time step Δt.
- or apply an analog MC scheme and keep track of the count rate to take correlations and higher moments into account [16]. This is usually done in a file which must be post-treated (binning and linear fit etc. see [16]). But analog schemes are known to have a slower convergence rate, to be computationally intensive and inadapted to very multiplicative media (in this case the size of the written file is known to explode).

Of course, the above list of alternatives may not be exhaustive. Anyway, the application of gPC does introduce a new parameter (P rather than a time step as in [12] or [6]). But the modifications to an existing solver are minor (compare Algorithms 1 and 2). Furthermore, the approximation with respect to P can even be expected to yield spectral convergence for smooth solutions [18]. In the example above, a gPC-i-MC approximation of M_2 is simply given by

$$M_2(x,t,\mathbf{v}) = \sum_{k=0}^{\infty}(u_k^X(x,t,\mathbf{v}))^2 \approx \sum_{k=0}^{P}(u_{k,N_{MC}}^X(x,t,\mathbf{v}))^2,$$

where $\forall k \in \{0,..,P\}$ we have

$$u_k^X(x,t,\mathbf{v}) \approx u_{k,N_{MC}}^X(x,t,\mathbf{v}) = \sum_{p=1}^{N_{MC}} w_p(t)\delta_x(x_p(t))\delta_{\mathbf{v}}(\mathbf{v}_p(t))\phi_k^X(X_p).$$

In the following section, we numerically verify and illustrate the previous points and even consider more elaborated statistical outputs of interest (in particular Sobol indices for sensitivity analysis, see [13,79]).

5. Numerical examples and discussions

In this section, we go through several test-problems emphasizing the strengths and weaknesses of the new MC scheme we suggested in the previous section. We first briefly go back to the simple problem of section 2 which motivated the introduction of the new approach. We then consider four spatially dependent problems for which analytical solutions are not anymore available (to our knowledge). The test-cases are progressive in difficulty and relevance: first we consider an uncertain scattering problem in one stochastic dimension (i.e. $Q = 1$) with simple statistical outputs of interest. Second, we perform a sensitivity analysis with respect to three parameters ($Q = 3$), implying the approximations of spatial Sobol indices [13,79]. For these two first test-problems (in sections 5.2–5.3), we rely on a ni-gPC application, as described in section 2, to

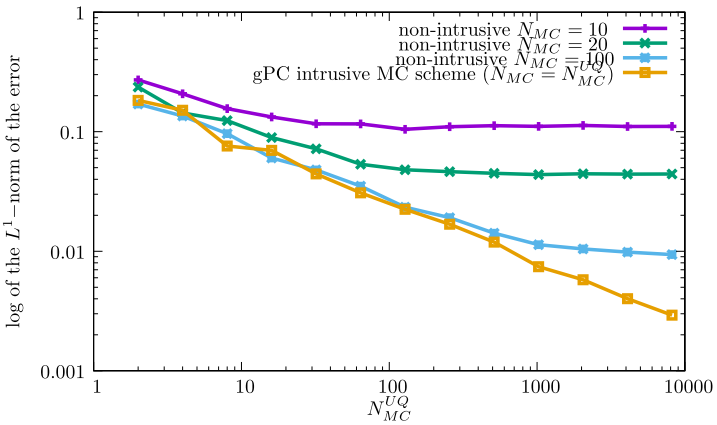

Fig. 2. Convergence studies with respect to $N = N_{MC}^{UQ}$ and $\Delta = N_{MC}$ as in Fig. 1 (bottom right) together with a curve obtained with the new gPC-i-MC scheme. For the latter, the uncertain parameters are sampled within the N_{MC} MC particles.

produce reference solutions: Gauss-Legendre points are used ($N = N_{GL}$) for their accuracy[13] and efficiency in relatively low stochastic dimensions (here, up to $Q = 3$). Finally, we tackle some two-layer medium problems: a $2D$ stochastic test-case with both uncertain initial condition and uncertain geometry, and a sensitivity analysis with respect to six parameters ($Q = 6$) for spatial particle density profiles in uncertain medium. For these two last configurations, the classical ni-gPC method is either too complex to use or too costly (this will be justified in sections 5.4 and 5.5). The test-cases are all monokinetic[14] for the sake of simplicity and ease of the reproducibility of the results.

Remark 5.1. The reader may notice the variability of the uncertain input parameters in the benchmarks is always relatively important: accordingly, we make sure the numerical discretisation of the MC simulation code (i.e. the black-box) is not too constraining (i.e. $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}}) \ll \mathcal{O}(N^{\beta(Q)})$, recall the example of section 2) and that the reference solutions obtained with ni-gPC can be produced in reasonable time. We insist this does not affect the relevance of the discussion.

5.1. Back to the simple configuration of Fig. 1

In this section, we briefly go back to the simple problem tackled in section 2. In Fig. 2, we display the same curves as in Fig. 1, obtained with a non-intrusive strategy ($N = N_{MC}^{UQ}$ and $\Delta = N_{MC}$), together with the one obtained with the gPC-i-MC scheme we described in the previous section. First, note that the new scheme is such that $N_{MC}^{UQ} = N_{MC}$: the experimental design is not anymore tensorized with the MC particles and the methodology consequently has *one less numerical parameter*. Besides, as displayed in Fig. 2, the approximation obtained with the new MC scheme does not stagnate with the increasing number of samplings. The uncertainty is solved *on-the-fly* during the MC resolution and the convergence rate for the whole problem remains $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ avoiding the kinks in the curves obtained non-intrusively.

5.2. Transport in an uncertain diffusive material

Let us now tackle a new test-problem for which an analytical solution is not available despite the relative simplicity of the configuration. Let us consider $x \in \mathcal{D} = [0, 2]$, $v = 1$. Furthermore, we assume the scattering is isotropic (notation $\int \mathrm{d}\omega = \int \mathbf{1}_{\mathcal{S}^2}(\omega)\frac{1}{4\pi}\,\mathrm{d}\omega = 1$) and that the medium is only diffusive (no absorption, i.e. $\sigma_t(X) = \sigma_s(X)$) even if uncertain. The initial condition is a Dirac mass at $x = 1$. In this particular case, (3) resumes to

$$\begin{cases} \partial_t u(x,t,\omega,X) + v\omega\nabla_x u(x,t,\omega,X) = -v\sigma_s(X)u(x,t,\omega,X) + \displaystyle\int v\sigma_s(X)u(x,t,\omega',X)\,\mathrm{d}\omega', \\ u(x,0,\mathbf{v}) = u_0(x) = \delta_1(x). \end{cases} \tag{39}$$

[13] For such integration method, we have, for any continuous functions g

$$\left| \int g(x)\,\mathrm{d}\mathcal{P}_X - \sum_{i=1}^{N} g(X_i)w_i \right| = \frac{K}{2N!} g^{(2N)}(\xi) = \kappa N^{\beta},$$

in which $g^{(n)}$ denotes the nth derivative of g, see [2,29].

[14] Taking into account multikinetic cross-sections is straightforward for the gPC-i-MC scheme. It has been designed to be able to take them into account by construction, see previous section, independently the choice of their discretization (multigroup, continuous). Some comments are provided in appendix B.

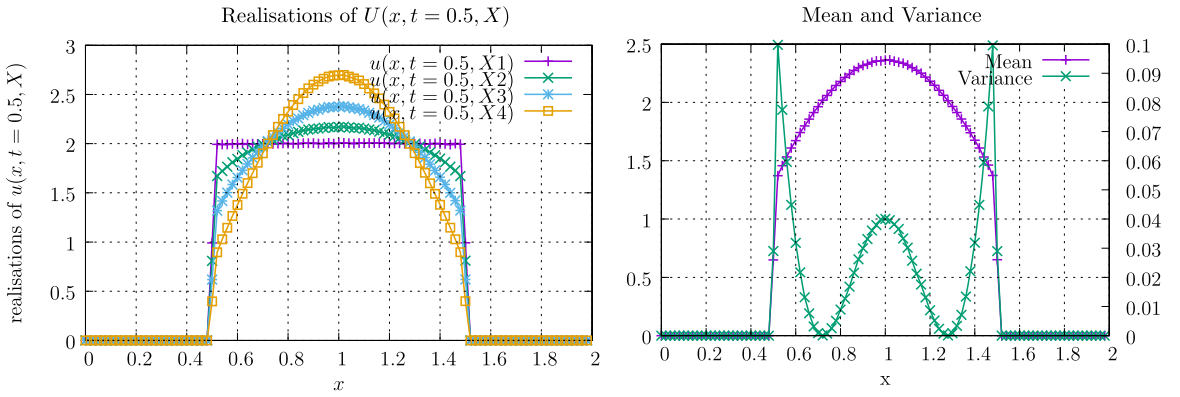

Fig. 3. Left: four realisations (taken at four non-intrusively obtained Gauss-Legendre points) of $U(x,t=0.5,X)$. Right: corresponding mean and variance of $U(x,t=0.5,X)$. The scale for the mean is on the left, for the variance on the right.

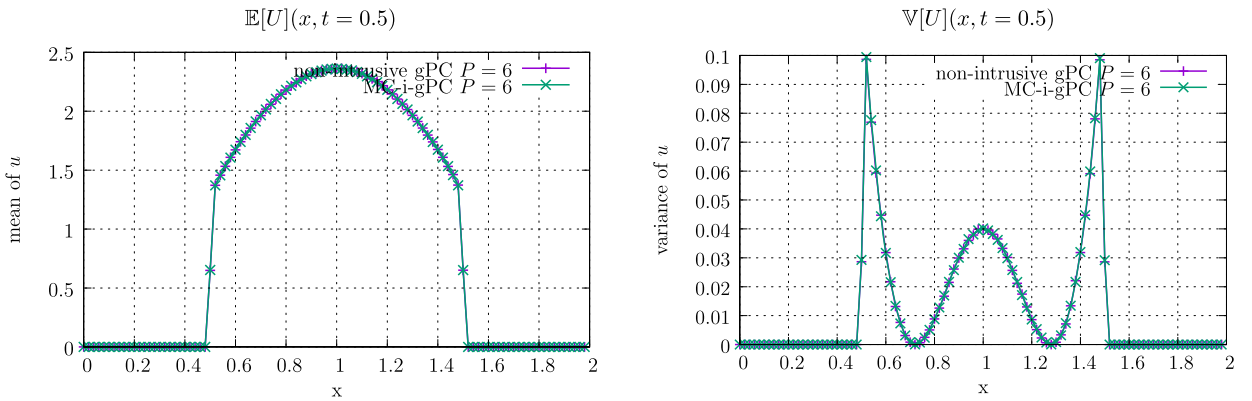

Fig. 4. Comparison between two discretisation of ni-gPC ($N_{GL}=50$, $P=6$ and $N_{GL}=4$, $P=2$) and gPC-i-MC ($P=2$). Every computation have $N_{MC}=3.2\times10^8$ particles. The results are in perfect agreement. Left: mean of $U(x,t=0.5,X)$. Right: variance of $U(x,t=0.5,X)$.

Let us consider a monodimensional uncertain parameter (i.e. $Q=1$) and assume $X\sim\mathcal{U}([-1,1])$ with $\sigma_s(X)=\overline{\sigma}_s+\hat{\sigma}_s X$ with $\overline{\sigma}_s=1$ and $\hat{\sigma}_s=0.99$. The variability is important in this example, see Remark 5.1. The fact the medium is only diffusive typically allows focusing on the difficulty tackled in the previous section relative to moment m_2 of u, see (37).

Let us now comment on the results of Figs. 3–4: Fig. 3 (left) presents four non-intrusively obtained realisations of the spatial profile $U(x,t=0.5,X)=\int u(x,t=0.5,\omega,X)\,\mathrm{d}\omega$. The particles propagate toward the left and right boundaries of the domain, hence the more or less steep fronts of the density U depending on the value of the uncertain parameter X. For $X=X_1$, the medium behaves almost as a vacuum whereas for $X=X_4$ it is very diffusive. From now on we focus on statistical observables such as the mean $\mathbb{E}[U]$ and variance $\mathbb{V}[U]$ of $U(x,t=0.5,X)$ with respect to x at time $t=0.5$. Note that every spatial statistical observable of this section is computed *via* the approximations of the gPC coefficients

$$U_k^X(x,t)=\int u_k^X(x,t,\omega)\,\mathrm{d}\omega=\iint u(x,t,\omega,X)\phi_k^X(X)\,\mathrm{d}\mathcal{P}_X\,\mathrm{d}\omega=\int U(x,t,X)\,\mathrm{d}\mathcal{P}_X, \tag{40}$$

and some of their post-treatments. For example, ($P-$truncated) approximations of the mean and variance are given by

$$\begin{aligned}\mathbb{E}[U](x,t)&=U_0^X(x,t),\\ \mathbb{V}[U](x,t)&=\sum_{k=1}^{P}(U_k^X(x,t))^2.\end{aligned} \tag{41}$$

Many other classical statistical quantities can be obtained from post-treatments of the gPC coefficients, see [11]. Some examples will be given in the following sections.

Fig. 3 presents the mean and variance profiles of U obtained with ni-gPC. The uncertainty transmitted from the medium to the particles is especially strong on the steep propagation fronts and at $x=1$ (i.e. vicinity of the initial condition $\delta_1(x)$).

Now, Fig. 4 compares the results in term of mean and variance profiles from

- (reference) ni-gPC with $N_{MC} = 3.2 \times 10^8$ particles, $N_{GL} = 50$ and $P = 6$,
- (best compromise) ni-gPC with $N_{MC} = 3.2 \times 10^8$ particles, $N_{GL} = 4$ and $P = 2$,
- gPC-i-MC with $N_{MC} = 3.2 \times 10^8$ particles and $P = 2$.

The results obtained with the above three options are in very good agreement on both observables. The fact the 'reference' solution ($P = 6$) and the 'best compromise[15]' ($P = 2$) have equivalent accuracies testifies of the fast convergence rate of gPC with respect to P. The fact the gPC-i-MC scheme also does have the same accuracy with $P = 2$ testifies that the new method can take advantage of it. Now regarding the cost of the three above options, we have

- reference: cost $= N_{GL} \times$ the averaged CPU time of one run $\approx 50 \times 85.0$ s,
- best compromise: cost $= N_{GL} \times$ the averaged CPU time of one run $\approx 4 \times 85.0$ s,
- gPC-i-MC: cost $= 1 \times$ the effective CPU time of one run $= 1 \times 86.2$ s.

The new gPC-i-MC method recovers the same results as the best compromise solution with only *one run of the MC simulation device* and very similar computational times (at least for this problem, this will not exactly be the case for the next benchmarks): it testifies of the insensitiveness to the number of dimensions of the *tracking*[16] of the MC particles. The runs were performed on $N_{\text{replication}} = 32$ replicated[17] domains for the parallel strategy, in very similar conditions as in [26]. The latter remark allows insisting on the fact that the parallel strategies which apply to a classical MC simulation also apply with the new gPC-i-MC one. The minor code modifications described in section 4 and appendix D to implement the gPC-i-MC solver do not imply a porting of the parallel counterpart, it is straightforward if already developed. A short discussion on parallel possibilities is provided in appendix D with the description of the new algorithm. The gain in computational time, for this test-case, is approximatively of $N_{GL} \approx 4$. In the above example, it remains relatively low: if one has access to $4(N_{GL}) \times 32(N_{\text{replications}}) = 128$ processors, which is common nowadays, the computational times are equivalent and the gain is only in term of computational resources.[18] We here recall that MC simulation codes are known to be computationally intensive and even such low factor (gain of only 4) may be welcome. In the following test-cases, we tackle some multidimensional uncertain problems.

5.3. Sensitivity analysis in 3D stochastic dimension

This example is a 3-dimensional stochastic (i.e. $Q = 3$) test-problem for which a reference solution with ni-gPC can still be obtained in reasonable times. The set-up is as follows:

- $v = 1$, $x \in \mathcal{D} = [0, 1]$, subdivided into $N_x = 100$ cells $\cup_{i=1}^{N_x} \mathcal{D}_i = \mathcal{D}$.
- Specular boundary condition on left (at $x = 0$) and vacuum one on the right hand side (at $x = 1$).
- Initially, the density of particles is homogeneous and deterministic, equal to 1, i.e. $u(x, t = 0, \omega, X) = u^0(x, \omega, X) = 1$ $\forall x \in \mathcal{D}, \forall \omega \in \mathcal{S}^2$.
- The medium is pure (i.e. $M = 1$ and $\eta = \eta_1$ see (2)), homogeneous and considered uncertain. It depends on three parameters $X = (X_1, X_2, X_3)$ affecting the total and scattering cross-sections and the material density as

$$\begin{array}{lll} \sigma_t(x,t,X) & = \sigma_t(X_1) & = \overline{\sigma}_t + \widehat{\sigma}_t X_1, \quad \forall x \in \mathcal{D}, t \in \mathbb{R}^+, \\ \sigma_s(x,t,\omega,\omega',X) & = \sigma_s(X_2) & = \overline{\sigma}_s + \widehat{\sigma}_s X_2, \quad \forall x \in \mathcal{D}, t \in \mathbb{R}^+, \forall (\omega,\omega') \in \mathcal{S}^2, \\ \eta(x,t,X) & = \eta(X_3) & = \overline{\eta} + \widehat{\eta} X_3, \quad \forall x \in \mathcal{D}, t \in \mathbb{R}^+, \end{array} \tag{42}$$

  in which (X_1, X_2, X_3) are independent uniformly distributed random variables on $[-1, 1]$, i.e. $\forall i \in \{1, 2, 3\}, X_i \sim \mathcal{U}([-1, 1])$.
- For the next computations, the mean quantities are set to $\overline{\sigma}_t = 1.0$, $\overline{\sigma}_s = 0.9$, $\overline{\eta} = 1.0$ and the ones controlling the variability to $\widehat{\sigma}_t = 0.4$, $\widehat{\sigma}_s = 0.4$, $\widehat{\eta} = 0.4$. Note that Remark 5.1 also applies here.
- We are interested in the mean $\mathbb{E}[U]$, variance $\mathbb{V}[U]$ and Sobol indices $\mathbb{S}^{\text{tot}}[U]$ profiles of $U(x, t, X) = \int u(x, t, \omega, X)\, \mathrm{d}\omega$ at time $t = 1.0$. The total and first order Sobol indices [13,78] relative to output U are denoted by $\mathbb{S}^{\text{tot}}[U] = (\mathbb{S}_1^{\text{tot}}[U], ..., \mathbb{S}_Q^{\text{tot}}[U])^t$ and $\mathbb{S}^1[U] = (\mathbb{S}_1^1[U], ..., \mathbb{S}_Q^1[U])^t$. They represent powerful but costly (see [13]) statistical tools designed to identify, for a given output of interest, which of the uncertain parameters explain the most the variability of the response. They can also be accurately approximated *via* post-treatments of the gPC coefficients, see [10,11].

[15] Best compromise in term of relative accuracy and restitution times. To find it, we simply ran many tests for different N_{GL}, P, N_{MC}.

[16] See Algorithm 1 for the definition of the tracking of one MC particle.

[17] Domain replication [15,26,54,55,75] corresponds to the most common parallel strategy for MC simulation codes. It takes advantage of the independence between MC particles, hence populations of MC particles: $N_{\text{replication}}$ processors each have a batch of particles and the processors only communicate at the end of the time step to average over the $N_{\text{replication}}$ populations.

[18] Note that rigorously speaking, the cost of ni-gPC remains given by the more costly run amongst the N_{GL} but we will keep considering the average CPU time over the N_{GL} calculations as a reference in the following comparisons.

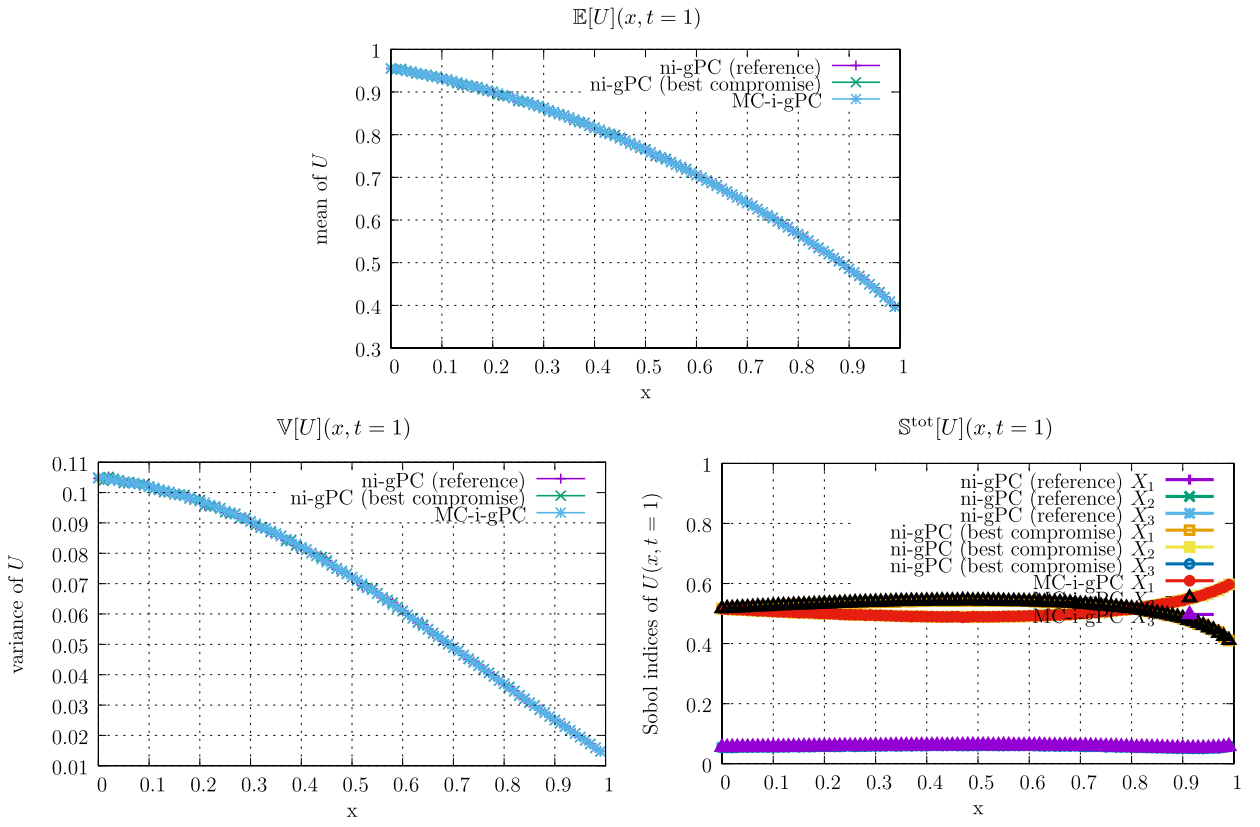

Fig. 5. Comparison between ni-gPC and gPC-i-MC. Top picture: mean of $U(x,t=1,X)$ with respect to x. Bottom left: variance of $U(x,t=1,X)$ with respect to x. Bottom right: total Sobol indices of $U(x,t=1,X)$ with respect to x.

Fig. 5 compares results obtained with ni-gPC and the gPC-i-MC scheme for the aforementioned statistical outputs of interest. Before going through resolution strategy comparisons, let us present briefly the results: Fig. 5 (top) shows the mean of $U(x,t=1,X)$. Particles are globally absorbed in the vicinity of $x=0$: indeed, we initially have $U(x,t=0,X)=1, \forall x \in \mathcal{D}$ and on Fig. 5 (top), we have $\mathbb{E}[U](x\sim 0, t=1) < \mathbb{E}[U](x\sim 0, t=0)=1$. This averaged particle absorption occurs despite the probable multiplicative effect ($\sigma_s(X) > \sigma_t(X)$ for some realisations of X) of the medium. Particles are globally lost in the vicinity of $x=1$, as $\mathbb{E}[U](x\sim 1, t=1) < \mathbb{E}[U](x\sim 1, t=0)=1$, mainly due to the vacuum boundary condition. Fig. 5 (bottom-left) shows the variance of U. The uncertainty is more important in the vicinity of $x=0$ and drops of a decade between $x=0$ and $x=1$. Now we are interested in identifying which of the uncertain parameters explain the most the previous variability: the total Sobol indices for $X=(X_1,X_2,X_3)$ are displayed Fig. 5 (bottom-right). Globally, X_3, related to an uncertainty on η, is the lesser important parameter: its total Sobol indice is the lowest at every spatial location $x\in\mathcal{D}$ of the simulation domain. Parameters X_1 and X_2, impacting respectively the scattering and total cross-sections, have a globally equivalent influence. The uncertainty on the scattering cross-section (X_2) is in particular more important than the one on the total cross-section (X_1) in the vicinity of the vacuum boundary condition $x=1$. Fig. 6 presents a comparison of total ($\mathbb{S}^{\text{tot}}[U]$) and first order ($\mathbb{S}^1[U]$) Sobol indices: we recall (see [79]) that the total indice $\mathbb{S}^{\text{tot}}_i[U]$ relative to X_i takes into account the first order indice $\mathbb{S}^1_i[U]$ of X_i together with its interaction with every other variable. In particular, in Fig. 6 (bottom-right), we see that X_3 is important mainly through its interactions (see the important difference between $\mathbb{S}^1_3[U]$ and $\mathbb{S}^{\text{tot}}_3[U]$) with X_1 and X_2. Hence, even if every of the uncertain parameter is not negligible (relatively important total Sobol indices on Fig. 6 bottom-right), Fig. 6 bottom-right attests reducing the uncertainty on X_1, X_2 will also lead to a reduction of the uncertainty due to X_3.

Going further in interpretation would need the careful study of second order Sobol indices but this is not really the scope of this section. Section 5.5 will provide a more pedagogical example. We here wanted to put forward that the new gPC-i-MC scheme is able to accurately recover statistical quantities known to be very efficient but costly [13].

A quick look to Fig. 5 shows that the three resolutions are in agreement, of equivalent accuracies, whatever the statistical observable of interest (mean, variance or Sobol indices). Each solution has been obtained with $N_{MC}=3.2\times 10^7$ particles. Let us now focus on their differences:

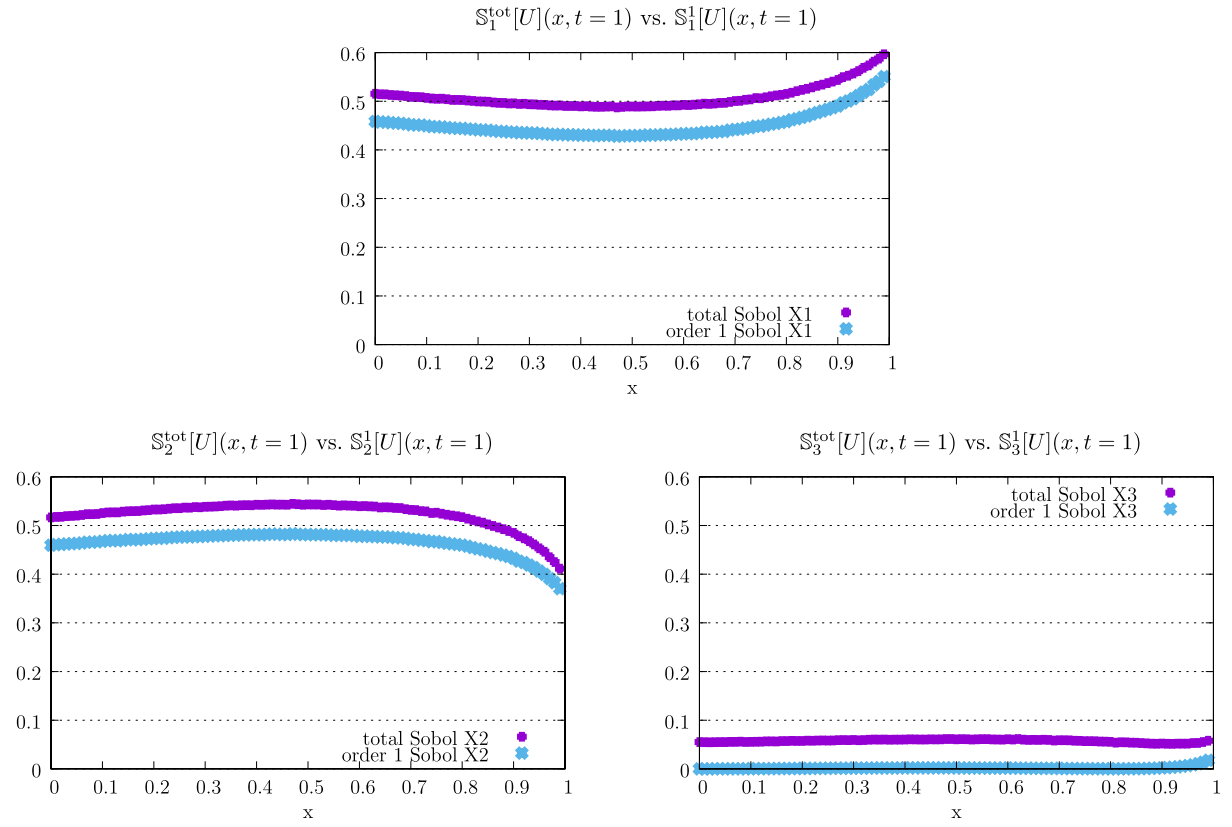

Fig. 6. All the results of this figure have been obtained applying the gPC-i-MC scheme. Comparison between total and first order Sobol indices for the three uncertain inputs X_1 (top), X_2 (bottom-left) and X_3 (bottom-right).

- (reference) ni-gPC with $N_{GL}^Q = 5^3 = 125$ and $(P+1)^Q = (3+1)^3 = 64$ coefficients,
- (best compromise) ni-gPC with $N_{GL}^Q = 4^3 = 64$ and $(P+1)^Q = (2+1)^3 = 27$ coefficients,
- gPC-i-MC with and $(P+1)^Q = (2+1)^3 = 27$ coefficients.

First, once again two different levels of discretisation for ni-gPC give equivalent results. The good agreement between the latters testifies that the uncertainty quantification counterpart of the decoupled approach is converged. Once again, the fast convergence rate of gPC is put forward: we obtain accurate solutions for low ($P = 2$) polynomial orders in every direction. The gPC-i-MC scheme gives equivalent results with one run and $(P+1)^Q = (2+1)^3 = 27$ gPC coefficients: it consequently takes advantage of the fast convergence rate of gPC.

Second, let us discuss the average CPU times and costs of the two methods for equivalent accuracies:

- reference: cost = N_{GL} × the averaged CPU time of one run $\approx 125 \times$ 3min 50 s,
- best compromise: cost = N_{GL} × the averaged CPU time of one run $\approx 64 \times$ 3min 52 s,
- gPC-i-MC: cost = 1 × the effective CPU time of one run $= 1 \times$ 4min 50 s.

Each computation was performed on $N_{\text{replication}} = 32$ replicated domains (i.e. processors).

For the previous test-case (section 5.2), the overall cost of one gPC-i-MC run was very similar to the average CPU time of the N_{GL} ni-gPC ones. For the one of this section, the CPU times present a signicative difference deserving a careful study: one gPC-i-MC run costs about ≈ 1.26 times the average CPU time of the ni-gPC ones. The main difference with the case of section 5.2 is the number of coefficients to be computed: $(P+1)^Q = (2+1)^3 = 27$ here instead of $(P+1)^Q = (2+1)^1 = 3$ for the example of section 5.2. This increase affects two phases of the presented computations:

- the size of the parallel reduction/communication between the $N_{\text{replication}}$ replicated domains,
- the number of tallies[19] one MC particle must perform.

[19] See Algorithm 1 for the definition of the *tallying* phase.

Table 1
Comparison of CPU times for sequential ($N_{\text{replication}} = 1$) and parallel (domain replication with $N_{\text{replication}} = 8$) runs for gPC-i-MC and ni-gPC.

gPC-i-MC	$N_{\text{replication}} = 1$	$N_{\text{replication}} = 8$
t_{CPU} for $(P+1)^Q = (0+1)^3 = 01$	1 min 44 s	2 min 02 s
t_{CPU} for $(P+1)^Q = (1+1)^3 = 08$	2 min 11 s	2 min 33 s
t_{CPU} for $(P+1)^Q = (2+1)^3 = 27$	4 min 16 s	4 min 33 s
t_{CPU} for $(P+1)^Q = (3+1)^3 = 64$	8 min 29 s	8 min 52 s
ni-gPC	$N_{\text{replication}} = 1$	$N_{\text{replication}} = 8$
$\bar{t}_{\text{CPU}}$	3 min 28 s	3 min 45 s

Table 1 compares the CPU times of sequential, in this case there are no communication cost, and parallel runs in comparable conditions (same N_{MC}). A horizontal reading of Table 1 gives an idea of the cost of the reduction phase between the $N_{\text{replication}} = 8$ replicated domains whereas a vertical one gives information on the cost of increasing the number of tallies the uncertain MC particles must perform to apply gPC-i-MC. The main increase in computational time comes from the *tallying* and not the parallel reduction. The *tallying* phase is more sensitive to an increase of dimension Q. Nonetheless, using $N_{\text{replication}} = 40$ replicated domains with less ($N_{MC} = 0.8 \times 10^7$) uncertain MC particles instead of $N_{\text{replication}} = 32$ and $N_{MC} = 3.2 \times 10^7$ ensures recovering the same results as in Fig. 5 with similar restitution times: in this case, for equivalent accuracies and CPU times, the gain in computational resources is of a factor $\frac{N_{GL} \times 32}{40} = \frac{64 \times 32}{40} = 51.2$.

As a concluding remark of this section, we would like to emphasize that the numerical accuracy of the gPC-i-MC scheme is still $\mathcal{O}(\frac{1}{\sqrt{N_{MC}}})$ but the cost of the treatment of one uncertain MC particle is more important than for a classical one, especially as $(P+1)^Q$ increases. We only briefly tackled domain replication as a (distributed [15,26,54,55,75]) parallel strategy but shared memory parallel ones (threads [15,26,54,55,75], GP-GPU, vectorisation) could be applied, for example, to accelerate the tallying phase. The idea is in the same vein as the one depicted in [14,96] for the *on-the-fly* Doppler broadening with the SIGMA1 algorithm to differently make the most of new computer architectures. Of course, we do not expect the efficiency for the tallying phase to reach the one of the SIGMA1 algorithm as in [14] but it opens new possibilities. From now on we will let this aspect (finer parallel strategies) of the discussion for future papers.

5.4. Transport in an uncertain geometry with uncertain initial condition

This example is a 2−dimensional stochastic test-problem (i.e. $Q = 2$). It consists in a two layered media with an initial uncertain density of particle in some part of the left medium together with an uncertain position of the interface between the two layers. Let us begin with a general presentation of the configuration:

- $v = 1$, $x \in \mathcal{D} = [0, 1]$, subdivided into N_x cells $\cup_{i=1}^{N_x} \mathcal{D}_i = \mathcal{D}$. In practice, we take $N_x = 50$.
- Specular boundary condition on the left (at $x = 0$) and vacuum one on the right hand side (at $x = 1$).
- The material is composed of two layers of different media, A and B with $\mathcal{D}_A = [0, x_{\text{int}}(X_1)]$ and $\mathcal{D}_B = [x_{\text{int}}(X_1), 1]$ such that $\mathcal{D}_A \cup \mathcal{D}_B = \mathcal{D} = [0, 1]$. Both media are pure (i.e. $M = 1$ and $\eta = \eta_1$ see (2)), homogeneous and deterministic. The interface x_{int} between the two media is considered uncertain, equal to $x_{\text{int}}(X_1) = \frac{1}{2} + \sigma_{\text{int}} X_1$ with $\sigma_{\text{int}} = 0.3$ and $X_1 \sim \mathcal{U}([-1, 1])$. We furthermore have $\forall i \in \{A, B\}$

$$\begin{aligned}
\sigma_t(x,t) &= \sum_{i\in\{A,B\}} \sigma_t(x)\mathbf{1}_{\mathcal{D}_i}(x) = \sum_{i\in\{A,B\}} \overline{\sigma}_t^i \mathbf{1}_{\mathcal{D}_i}(x), \ \forall x \in \mathcal{D}, t \in \mathbb{R}^+, \\
\sigma_s(x,t,\omega,\omega') &= \sum_{i\in\{A,B\}} \sigma_s(x)\mathbf{1}_{\mathcal{D}_i}(x) = \sum_{i\in\{A,B\}} \overline{\sigma}_s^i \mathbf{1}_{\mathcal{D}_i}(x), \ \forall x \in \mathcal{D}, t \in \mathbb{R}^+, \\
\eta(x,t) &= \sum_{i\in\{A,B\}} \eta(x)\mathbf{1}_{\mathcal{D}_i}(x) = \sum_{i\in\{A,B\}} \overline{\eta}^i \mathbf{1}_{\mathcal{D}_i}(x), \ \forall x \in \mathcal{D}, t \in \mathbb{R}^+.
\end{aligned} \tag{43}$$

- For the next computations, the quantities are set to
  $\overline{\sigma}_t^A = 1.0$, $\overline{\sigma}_s^A = 1.3$, $\overline{\eta}^A = 1.0$,
  $\overline{\sigma}_t^B = 1.0$, $\overline{\sigma}_s^B = 0.9$, $\overline{\eta}^B = 1.0$.
- We have a homogeneous and uncertain particle density $u(x, t = 0, \omega, X) = u^0(x, \omega, X) = 1 + \sigma_0 X_2 \ \forall x \in \mathcal{D}_A, \forall \omega \in \mathcal{S}^2$. In practice we take $\sigma_0 = 0.01\mathbf{1}_{x \le \frac{1}{8}}$ and $X_2 \sim \mathcal{U}([-1, 1])$.
- There are no particle initially in the right hand side of the domain, i.e. $u(x, t = 0, \omega, X) = u^0(x, \omega, X) = 0 \ \forall x \in \mathcal{D}_B, \forall \omega \in \mathcal{S}^2$.

Fig. 7 presents the results obtained with the gPC-i-MC solver with $(P+1)^Q = (6+1)^2 = 49$ coefficients and $N_{\text{replication}} \times N_{MC} = 128 \times 5 \times 10^8$ uncertain MC particles. On the left hand side, it displays spatial profiles of the mean $\mathbb{E}[U](x, t)$, standard deviation $\sqrt{\mathbb{V}[U]}(x, t)$ and N_s approximated realisations $(U^P(x, t, X_i))_{i\in\{1,\dots,N_s\}}$ at several times. The latter realisations

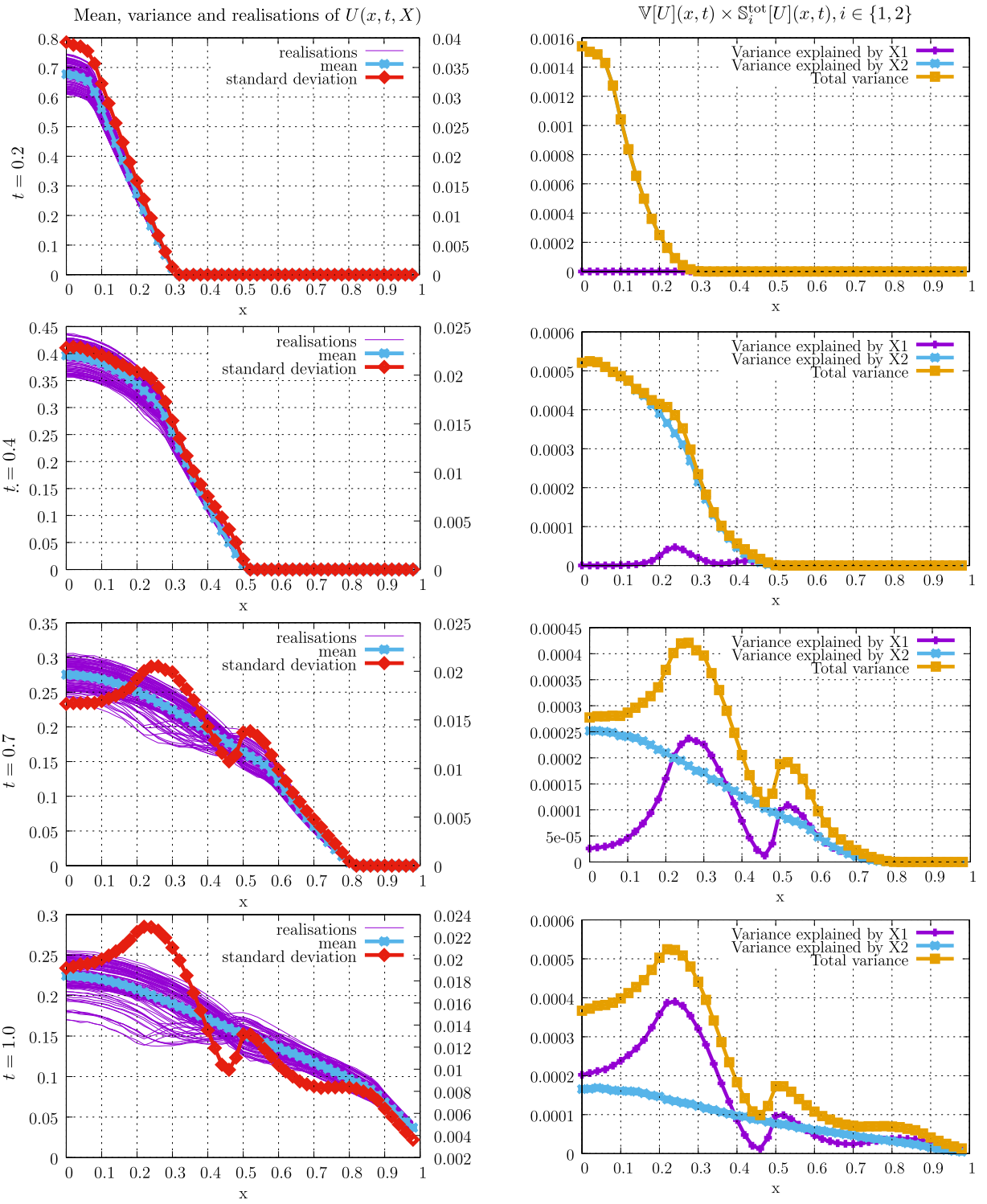

Fig. 7. All the results of this figure have been obtained applying the gPC-i-MC scheme. Left: time evolution of the mean and variance profiles together with 200 realisations recovered thanks to the use of gPC as a reduced model. Right: time evolution of the profiles of the total variance and of the amounts of variance explained by X_1 and X_2.

are post-treatments, obtained by $N_s = 200$ (post computation) samplings of $X = (X_1, X_2)^t$ and rebuilding $\forall i \in \{1, ..., N_s\}$, $U(x,t,X_i) \approx U^P(x,t,X_i) = \sum_{k=0}^{P} U_k^X(x,t)\phi_k^X(X_i)$. On the right hand side, the figure presents the total variance and the amounts of variance explained by X_1 and X_2, i.e. $\mathbb{V}[U](x,t), \mathbb{V}[U](x,t) \times \mathbb{S}_1^{\text{tot}}[U](x,t)$ and , $\mathbb{V}[U](x,t) \times \mathbb{S}_2^{\text{tot}}[U](x,t)$. Note

that for this test-case, interactions between X_1 and X_2 are negligible. The different lines of Fig. 7 correspond to different times: they give an idea of the dynamics of the particle flow and of the propagation of the uncertainties to the flow of particles. For the left column, the left scale is for the mean and for the realisations whereas the right one is for the standard deviation. Initially, the uncertainty is located in the left part of the domain. Top right picture of Fig. 7 even shows that initially, the variance is only explained by variable X_2, i.e. the uncertainty in the initial condition. As time increases, particles flow toward the uncertain interface between the two medium and X_1 begins to affect the flow of particles: we observe a kind of peak of standard deviation developing with time in the vicinity of the uncertain interface position. It is visible on the second line right picture of Fig. 7: variable X_1, responsible for the uncertainty in the interface position, begins to have a greater influence on the flow. On the left hand side, the realisations illustrates how the uncertainty affects the fluctuations around the mean of the particle flow. The fluctuations are propagating toward the right side of the domain with time. Note that the realisations are not equally distributed above and under the mean, especially on the left hand side of the resolution domain: the distribution is skewed in this area. The left column of Fig. 7 highlights how the uncertainty due to the interface position takes more and more importance with time: at the beginning ($t = 0.2$) the total variance is equal to the variance explained by X_2, i.e. the initial condition. For $t = 0.4$, the variance in the vicinity of $x \in [0.2, 0.3]$ is explained also by X_1. For the last times ($t = 0.7$ and $t = 1.0$), locally, the influence of X_1 becomes preponderant.

With this test-case, we can emphasize three important aspects of the gPC-i-MC solver:

- first, it can deal with uncertain initial conditions without more difficulties. The same applies for uncertain boundary conditions, even if not explicitly tackled in this document.
- Second, it is possible to post-treat the coefficients $(U_k^X(x,t))_{k\in\{0,\dots,P\}}$ to build some approximated realisations (often called metamodels in the literature). They can be used, as in Fig. 7 to give a better idea of the range of variability around the mean of the solution or to solve a stochastic inverse problem for example [7,57,86].
- Third, the new intrusive gPC-i-MC is very convenient to study geometric uncertainties. Suppose one wants to solve the same problem as above with a non-intrusive method: one would need to
  - either use as many meshes as realisations of the uncertain location of the interface to make sure the cells correspond with the interface realisation.
  - or use a common and very fine mesh for every non-intrusive computations.

  In the first case, one needs to build those meshes, this is often tiresome and tedious to accomplish (even in 1D spatial dimension). On some complex geometries, this can become discouraging. At the end, the results must be remapped on a common mesh for visualisation. The latter task is also far from being simple. In the second case, if the variability of the interface location is small, this may imply a very fine mesh and an untractable problem. One remaining possibility would be to develop some models of mixed cells but once again, this is intrusive.

Let us describe how the gPC-i-MC solver deals efficiently (without relying on multiple meshes nor an extremely refined one) with the above problem. We only had to specify (intrusively but this can be done *via* some user model module of the simulation code) a parametrization of the geometry with respect to x and X. Basically, we coded the function $x, X \longrightarrow \mathbf{1}_{[0,x_{\text{int}(X_1)}]}(x)\sigma_\alpha^A + \mathbf{1}_{[x_{\text{int}(X_1)},1]}(x)\sigma_\alpha^B$, which only had to be 'called' at the location x_p and the realisation X_p of every uncertain MC particle (see Algorithm 2). This way, the uncertain interface location is quite decorrelated from an eventual grid which serves only for visualisation purposes. This is demonstrated by the computations of Fig. 7 with a mesh using $N_x = 50$ cells: the uncertainty induces a variability range of the interface between $x_{\text{int}}(X_1) \in [0.2, 0.8]$ whereas the size of the cells is only of $\frac{1}{N_x} = \frac{1}{50} = 0.02$. In interval $[0.2, 0.8]$ there are consequently only 30 cells: for a non-intrusive application, there would only consequently be a maximum of $N = 30$ independent runs for the uncertainty integration. Depending on the choice of the experimental design (see section 2), this may impose a coarse accuracy on the statistical observables.

5.5. A two layered uncertain material

In this section we tackle a last test-problem in 6−stochastic dimensions. The test-case is a mix between the problem of section 5.3 and the one of section 5.4: a two-layer random media. For this test-case, we consider applying ni-gPC is not possible in reasonable times: assuming the same convergence properties as for problem of section 5.3 applies (i.e. need for at least $N_{GL} = 4$ Gauss-Legendre points and $P = 2$ per directions), we would need $N_{GL}^Q = 4^6 = 4096$ runs of an MC black-box code (i.e. $N_{GL} \times N_{\text{replication}} = 4096 \times 32 = 131072$ processors) to compute $(P+1)^Q = 3^6 = 729$ gPC coefficients and solve the problem. On another hand, gPC-i-MC can handle it efficiently as presented below.

The general set-up is as follows:

- $v = 1$, $x \in \mathcal{D} = [0, 1]$, subdivided into $N_x = 100$ cells $\cup_{i=1}^{N_x} \mathcal{D}_i = \mathcal{D}$.
- Specular boundary condition on the left (at $x = 0$) and vacuum one on the right hand side (at $x = 1$).
- Homogeneous and deterministic particle density $u(x, t = 0, \omega, X) = u^0(x, \omega, X) = 1 \ \forall x \in \mathcal{D}, \forall \omega \in \mathcal{S}^2$.
- The material is composed of two layers of different media, A and B with $\mathcal{D}_A = [0, \frac{1}{2}]$ and $\mathcal{D}_B = [\frac{1}{2}, 1]$ such that $\mathcal{D}_A \cup \mathcal{D}_B = \mathcal{D} = [0, 1]$. Both media are pure (i.e. $M = 1$ and $\eta = \eta_1$ see (2)), homogeneous and considered uncertain. Each depends on three parameters $(X^i)_{i\in\{A,B\}} = (X_1^i, X_2^i, X_3^i)_{i\in\{A,B\}}$ affecting the total and scattering cross-sections and the material density as in the test-problem of section 5.3. We have $\forall i \in \{A, B\}$

$$\begin{aligned}
\sigma_t(x,t,X) &= \sum_{i\in\{A,B\}} \sigma_t(X_1^i)\mathbf{1}_{\mathcal{D}_i}(x) = \sum_{i\in\{A,B\}} \left[\overline{\sigma}_t^i + \widehat{\sigma}_t^i X_1^i\right]\mathbf{1}_{\mathcal{D}_i}(x),\ \forall x\in\mathcal{D}, t\in\mathbb{R}^+,\\
\sigma_s(x,t,\omega,\omega',X) &= \sum_{i\in\{A,B\}} \sigma_s(X_2^i)\mathbf{1}_{\mathcal{D}_i}(x) = \sum_{i\in\{A,B\}} \left[\overline{\sigma}_s^i + \widehat{\sigma}_s^i X_2^i\right]\mathbf{1}_{\mathcal{D}_i}(x),\ \forall x\in\mathcal{D}, t\in\mathbb{R}^+,\\
\eta(x,t,X) &= \sum_{i\in\{A,B\}} \eta(X_3^i)\mathbf{1}_{\mathcal{D}_i}(x) = \sum_{i\in\{A,B\}} \left[\overline{\eta}^i + \widehat{\eta}^i X_3^i\right]\mathbf{1}_{\mathcal{D}_i}(x),\ \forall x\in\mathcal{D}, t\in\mathbb{R}^+,
\end{aligned} \tag{44}$$

in which $(X_1^i, X_2^i, X_3^i)_{i\in\{A,B\}}$ are independent uniformly distributed random variables on $[-1,1]$, i.e. $\forall i\in\{A,B\}$, $j\in\{1,2,3\}$, $X_j^i\sim\mathcal{U}([-1,1])$.

- For the next computations, the mean quantities are set to

$$\overline{\sigma}_t^A = 1.0,\ \overline{\sigma}_s^A = 1.3,\ \overline{\eta}^A = 1.0,$$
$$\overline{\sigma}_t^B = 1.0,\ \overline{\sigma}_s^B = 0.9,\ \overline{\eta}^B = 1.0,$$

and the ones controlling the variability to

$$\widehat{\sigma}_t^A = 0.4,\ \widehat{\sigma}_s^A = 0.4,\ \widehat{\eta}^A = 0.4,$$
$$\widehat{\sigma}_t^B = 0.4,\ \widehat{\sigma}_s^B = 0.4,\ \widehat{\eta}^B = 0.4,$$

Note that Remark 5.1 also applies here.

In the following, we aim at answering several questions: which of the 6 parameters explain the most the variability of the particle density at some prescribed locations? Is it possible to reduce the dimensionality of the problem? By which factor the uncertainty on the main parameters should be reduced to make the remaining ones equally influent?

Fig. 8 presents the results obtained with the gPC-i-MC scheme with $(P+1)^Q = (2+1)^6 = 729$ gPC coefficients estimated thanks to $N_{MC} = 1.024\times10^9$ uncertain MC particles. The computation has been performed on $N_{\text{replication}} = 1024$ processors and took about 750 s. Fig. 8 top-left displays the mean and variance profiles of $U(x,t=1,X) = \int u(x,t=1,\omega,X)\,\mathrm{d}\omega$. The scale for the mean is on the left hand side, for the variance on the right hand side. The profiles present two regimes, on each side of $x=0.5$. Material A, on the left hand side, is the most uncertain: the variability drastically drops between $x=0$ and $x=\frac{1}{2}$. On another hand, even if smaller in B, the variability remains almost constant on the whole range $x\in[\frac{1}{2},1]$. Fig. 8 top-right shows the total Sobol indices' profiles at $t=1$. The contribution of X_6 to the global variability of the particle density is very small $\forall x\in\mathcal{D}$: the dimension of the stochastic problem can consequently be reduced by assuming $X_6\approx\mathbb{E}[X_6]$. The six other pictures of Fig. 8 compare the total and first order Sobol indices. They confirm that the variability of X_6 can be neglected without impact on the problem together with the fact that every other parameter are influent: except for X_3, they do not depend too much on interactions as the total and first order indices are almost equal.

For the next studies, we can consequently keep $Q=5$ and choose $X_6 = \mathbb{E}[X_6] = 0$. The question arising now is: of how much should we reduce the uncertainty on the main parameters X_1, X_2, X_3 (which according to the previous study are the most influent parameters) to have every remaining input contributing equally to the global variability? Parameter X_3 being in interaction with other parameters, we can expect its variability to drop by only reducing the one on X_1, X_2. We consequently ran several similar calculations with decreasing values of $\hat{\sigma}_t^A, \hat{\sigma}_s^A$ (controlling the variabilities of X_1, X_2) until all total Sobol indice is of the same magnitude (as the ones of X_4, X_5). We performed dozen calculations and realized that having $\hat{\sigma}_t^A = 0.15, \hat{\sigma}_s^A = 0.15$ leads to satisfying results. They are displayed in Fig. 9: it compares the mean, variance and total Sobol indices in the nominal configuration ($\hat{\sigma}_t^A = 0.40, \hat{\sigma}_s^A = 0.40$) and in the newly studied one ($\hat{\sigma}_t^A = 0.15, \hat{\sigma}_s^A = 0.15$). First, Fig. 9 (left) compares the means and variances in the nominal and new configurations. The mean is slightly impacted but if we focus on the variances per parameter (Sobol indices multiplied by the total variance), the study shows that a small decrease (from 0.40 to 0.15) of the variabilities of the identified most influent parameters (X_1, X_2) leads to an important one on the outputs of interest. Besides, on Fig. 9 (right) as expected,

- the magnitude of the total Sobol indices of X_4, X_5 has not changed (remember their interactions with other variables are negligible).
- reducing the uncertainty of X_1 and X_2 has an impact on X_3: remember we have an important interaction counterpart for this variable, see the large difference between $\mathbb{S}_3^1$ and $\mathbb{S}_3^{\text{tot}}$ in Fig. 8 (bottom left). As a result, the new total Sobol indice of X_3 is now even below the ones of X_4, X_5.
- The total Sobol indices for X_1, X_2 are now of the same order as the one for X_3, X_4.

With this example, we wanted to emphasize the strength of Sobol indices to conduct a sensitivity analysis study. They allow focusing on the most relevant parameters and defining minimal objectives on which the efforts must be carried on to reduce the overall uncertainty. In the previous example, by focusing on X_1, X_2 and reducing their uncertainty of *only* a factor $\frac{0.15}{0.40} = 0.375$, it allowed reducing the effect of X_3 of a factor $\approx \frac{0.038}{0.01} = 3.80$ together with the overall uncertainty on the output of a factor $\approx \frac{0.245}{0.025} = 9.80$. Working more on X_1, X_2 would not be efficient as the overall accuracy would remain governed by the uncertainty on X_4, X_5. The Sobol indices allowed us identifying the most influent variables, the gPC-i-MC scheme allowed performing the study in reasonable times with reasonable resources.

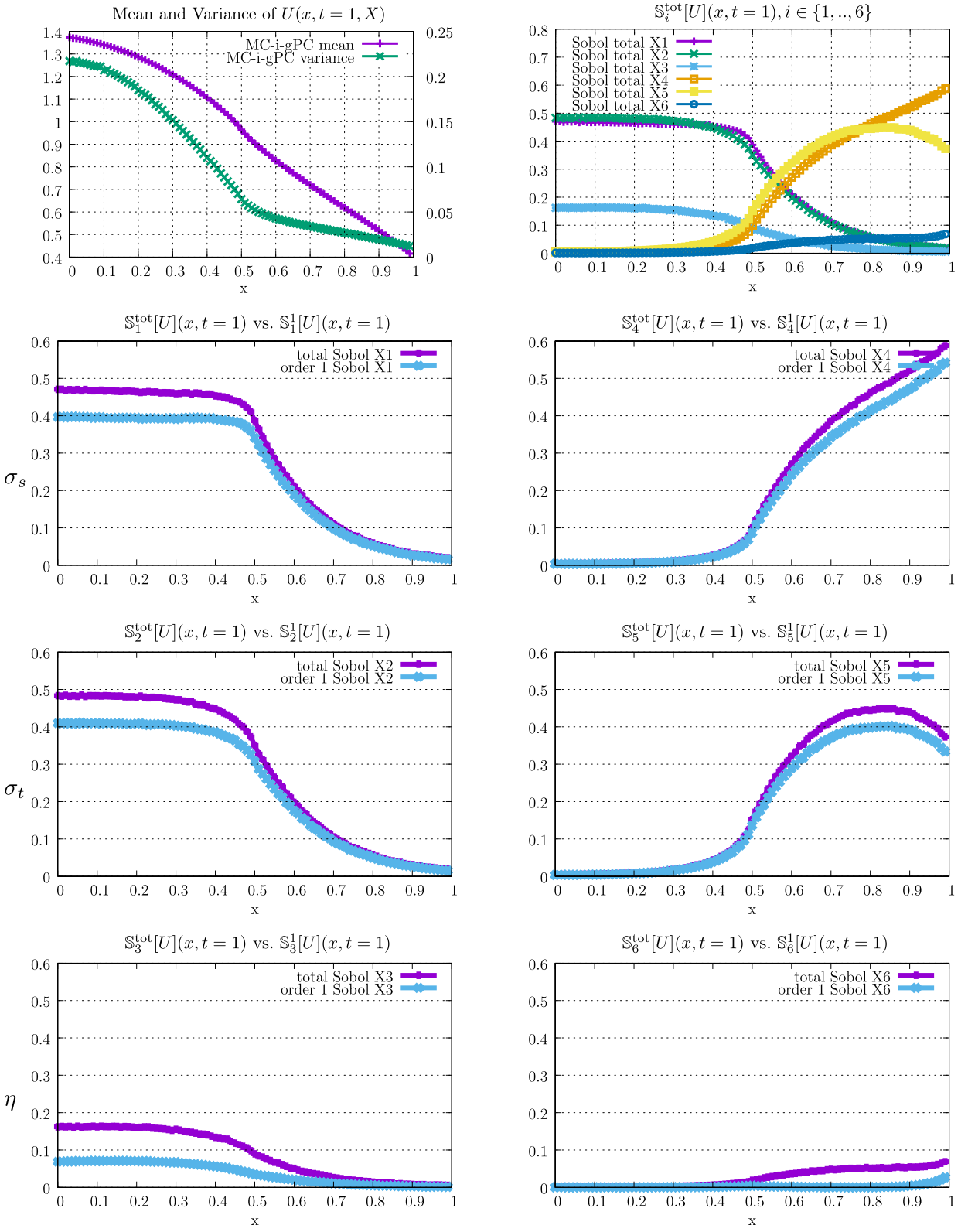

Fig. 8. All the results of this figure have been obtained applying the gPC-i-MC scheme. Top left: mean and variance profiles. Top right: total Sobol indices. Others: comparisons between total and first order Sobol indices for the six uncertain inputs $X_1, X_2, X_3; X_4, X_5, X_6$.

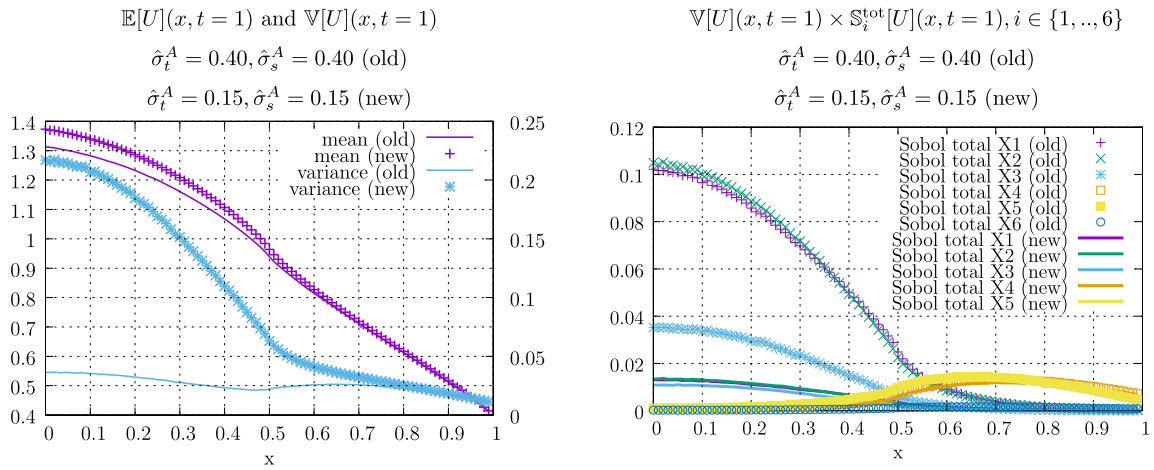

Fig. 9. All the results of this figure have been obtained applying the gPC-i-MC scheme. Left: mean and variance profiles. Right: total Sobol indices. Comparisons between the nominal ($\hat{\sigma}_t^A = 0.40, \hat{\sigma}_s^A = 0.40$) and newly ($\hat{\sigma}_t^A = 0.15, \hat{\sigma}_s^A = 0.15$) studied configurations.

6. Conclusion

To close this paper, we would like to come back on its main aspects and on future work for the gPC-i-MC solver. This paper introduces a new gPC based *intrusive* Monte-Carlo method for uncertainty quantification. Combining both intrusiveness for UQ and MC methods is singular and original. The manuscript explains how one can easily modify (hence the intrusiveness) an existing MC simulation code to take into account uncertainties in the linear Boltzmann equation *on-the-fly* during the MC resolution.

The efficiency of the new gPC based MC scheme is closely related to the insensitiveness of an increase of dimension of the tracking of the MC particles together with the fast convergence rate of gPC. The new method remains sensitive to the curse of dimensionality (as it depends on gPC) mainly during the tallying phase but shows important gains in comparison to non-intrusive gPC. We detailed the minor modifications to apply to the semi-analog/implicit capture MC scheme to take uncertainties into account *on-the-fly* during the MC resolution. The non-analog, analog and any other MC schemes can be modified similarly without further difficulties applying the material of this paper. In other words, the methodology is not scheme dependent: it also means it can be combined to variance reduction technique without major difficulties (see [8] for example).

The next steps consist in considering uncertain nonlinear couplings involving the linear Boltzmann equation: Bateman equations for neutronics [4,24,25,39,40], Stefan's law for photonics [52] or even hydrodynamics with the resolution of the BGK [5] model, solved with the new gPC-i-MC scheme. The efficiency of the new method is closely related to the unconditional stability of the MC scheme for the linear Boltzmann equation (less frequent parallel reductions). The stakes, for the aforementioned couplings/nonlinear models, will reside in the capability for the solvers to take large times steps during the MC phase, see [4] for example. Such applications will certainly be the subject of future papers.

Acknowledgements

The author would like to thank Hervé Jourdren: this work has been initiated within his team. Many thanks also to David Dureau, Adrien Bernede, Xavier Valentin, Boukhmes Mechitoua and Philippe Humbert for valuable discussions about High Performance Computing, Numerical Analysis and physical considerations. Thanks also to Karl Ryan for helping me with the English.

Appendix A. A simple analytical uncertain solution

In this section we build an analytical solution in a simple uncertain configuration. It is used as a reference solution for the convergence studies of section 2. The test-problem is also revisited in section 5.1 with the new gPC-i-MC scheme. In particular, this analytical solution is used in the quantitative results of Figs. 1 and 2.

The configuration is monokinetic (i.e. $v = 1$) and homogeneous (i.e. $u(x,t,\mathbf{v},X) = u(t,\omega,X)$). We assume the uncertainty, one-dimensional here for the sake of simplicity, affects the scattering cross-sections $\sigma_s = \overline{\sigma}_s + \hat{\sigma}_s X$, where $X \sim \mathcal{U}[-1,1]$ and $\hat{\sigma}_s$ is closely related to the variance of the uncertain scattering cross-section. Let us introduce $U(t,X) = \iint u(x,t,\omega,X)\,\mathrm{d}x\,\mathrm{d}\omega$. In the previously described configuration, the uncertain linear Boltzmann equation resumes to the following stochastic ordinary differential equation

$$\begin{cases} \partial_t U(t,X) + v\sigma_t U(t,X) = v\sigma_s(X)U(t,X), \\ U(0) = U_0, \end{cases} \tag{A.1}$$

satisfied by U. Introduce $\sigma_a = \sigma_t - \sigma_s$, then the solution is given by

$$U(t,X) = U_0 e^{-v\sigma_a(X)t} = U_0 e^{-v(\sigma_t - \overline{\sigma}_s - \hat{\sigma}_s X)t} = U_0 e^{-v(\overline{\sigma}_a - \hat{\sigma}_s X)t}. \tag{A.2}$$

The quantity $U(t,X)$ is a random variable indexed by time t, i.e. it is a stochastic process. In this case, mean and variance of the stochastic process (A.2) can be computed analytically and are given by

$$\begin{aligned} M_1^U(t) = \mathbb{E}[U(t,X)] &= \tfrac{1}{2} U_0 e^{-v\overline{\sigma}_a t} \frac{e^{v\hat{\sigma}_s t} - e^{-v\hat{\sigma}_s t}}{\hat{\sigma}_s t v}, \\ M_2^U(t) = \mathbb{E}[U^2(t,X)] &= \tfrac{1}{4} U_0^2 e^{-2v\overline{\sigma}_a t} \frac{e^{2v\hat{\sigma}_s t} - e^{-2v\hat{\sigma}_s t}}{\hat{\sigma}_s t v}, \\ \mathbb{V}[U](t) &= M_2^U(t) - (M_1^U(t))^2. \end{aligned} \tag{A.3}$$

Of course, higher order moments, probability of failure, complete characterisation of the probability density function of the stochastic process can be calculated but in Figs. 1–2 we focus on the variance $\mathbb{V}[U](t)$ to perform the convergence studies. Note that in practice, we take $v = 1, U_0 = 1, \sigma_t = \overline{\sigma}_s = 0.1, \hat{\sigma}_s = 0.1$. The L^1-norm of the error is computed at time $t = 10$. The curves of Fig. 1 implying an MC scheme are averaged over 128 computations with different seeds.

Appendix B. Comments on the notations of equation (26) for the scattering velocity distribution and on multikinetic cross-sections

In this appendix we comment on equation (26). We insist on

- the use of general notations,
- the possibility for the gPC-i-MC solver to take into account, by construction, multikinetic cross-sections. After all, the scattering distribution P is the part of the transport equation causing changes of velocities (and energies).

Let us begin by a clarification of the notations before briefly coming back to the multikinetic aspect of the solver. Expression (26) may appear difficult to comprehend. It has mainly been introduced to attest of the fact one must always inverse a cumulative density function to obtain the sampling of the inner velocity from a uniformly distributed sampling. In many applications, it is more common considering P is known in the center-of-mass frame [16,51,80]. Two samplings are then usually performed for each MC particle encountering a scattering event:

- the sampling of the norm of the inner (backward) velocity $v = |\mathbf{v}|$,
- the sampling of the angular deflection in the center-of-mass frame θ *conditionally* to having sampled v as the inner velocity.

Let us introduce $w \in \mathbb{R}^+ \longrightarrow f_v(w)$ and $\alpha, w \in [0, 2\pi] \times \mathbb{R}^+ \longrightarrow f_\theta(\alpha, w)$ the two probability density functions (pdfs) relative to the two above samplings. This means

$$\begin{aligned} f_v(w) > 0 \quad &\text{and} \quad \int_0^\infty f_v(w)\,\mathrm{d}w = 1, \\ \forall w \in \mathbb{R}^+ \;\; f_\theta(\alpha, w) > 0 \quad &\text{and} \quad \int_0^{2\pi} f_\theta(\alpha, w)\,\mathrm{d}\alpha = 1. \end{aligned}$$

Then v is sampled in practice by inversing the cumulative density function (cdf) of f_v and θ is sampled afterward by inversing the cdf of f_θ taken at v. In other words, we have

$$\mathcal{U}_v = \int_0^v f_v(w)\,\mathrm{d}w = F_v(v) \text{ with } \mathcal{U}_v \sim \mathcal{U}([0,1]), \tag{B.1}$$

and

$$\mathcal{U}_\theta = \int_0^\theta f_\theta(\alpha, v)\,\mathrm{d}\alpha = F_{\theta,v}(\theta) \text{ with } \mathcal{U}_\theta \sim \mathcal{U}([0,1]). \tag{B.2}$$

By inversing the cdf, we mean recovering v and θ from $\mathcal{U}_v$ and $\mathcal{U}_\theta$, i.e. solving $v = F_v^{-1}(\mathcal{U}_v)$ and $\theta = F_{\theta,v}^{-1}(\mathcal{U}_\theta)$. This obviously depends on the shapes of the functions f_v, f_θ, hence the reference to the *format* of the pdfs f_v and f_θ. Amongst the classical formats for the norm of the velocity field (closely related to the energy of the particles), there are multigroup, continuous cross-sections, etc. see [51,80]. In each case, inversion (B.1) applies. The difference only comes from the fact the different formats discretize differently the integral in (B.1). Amongst the classical formats for the angular deflection in the center-of-mass frame there are equiprobable tables, Legendre polynomial based ones, equi-cosinus ones, etc. see [51,80]. Once again, inversion (B.2) always applies, the difference coming from the different discretisation of the integral in (B.2).

Now, (26) is supposed to sum up the operations presented in the previous lines without assuming the samplings are done in the center-of-mass frame (this may be important for some applications, see [98,99] for example) or even assuming that one sampling is done conditionally to another (even if this is in general very convenient). The new solver only replaces (26) by

$$\mathcal{U}_{\mathcal{V}_{X_p}} = \int_{-\infty}^{\mathcal{V}_{X_p}} P(x_p(t-\tau), t-\tau, \mathbf{v}_p(t-\tau), \mathbf{v}', X_p)\,\mathrm{d}\mathbf{v}' \text{ where } \mathcal{U}_{\mathcal{V}_{X_p}} \sim \mathcal{U}([0,1]). \tag{B.3}$$

The notation implicitly tells the samplings of the inner norm of the velocity and of the angular deflection in the center-of-mass frame would only need to be parametered by X (if uncertain). In this sense, we insist on the fact the new gPC-i-MC allows, by construction, taking into account multikinetic problems.

Appendix C. Algorithmic treatments to solve (1) with an MC scheme

Algorithm 1: The MC semi-analog scheme described in term of algorithmic operations in order to compute (backward) $u(x,t,\mathbf{v})$. It corresponds to the general canvas of MC simulation codes. The while loop is commonly called the *tracking* of an MC particle. The $+=$ operation is commonly called the *tallying*.

set $u(x,t,\mathbf{v}) = 0$
for $p \in \{1, ..., N_{MC}\}$ **do**
  set $s_p = t$ *#this will be the remaining life time of particle p, it must go down to zero (backward)*
  set $x_p = x$
  set $\mathbf{v}_p = \mathbf{v}$
  set $w_p = \dfrac{1}{N_{MC}}$
  while $s_p > 0$ *and* $w_p > 0$ **do**
    if $x_p \notin \mathcal{D}$ **then**
      #here a very general function for the application of arbitrary boundary conditions
      apply_boundary_conditions$(x_p, s_p, \mathbf{v}_p)$
    end
    Sample τ from the distribution having probability measure $f(z_s^p)\,\mathrm{d}s$.
    if $\tau > s_p$ **then**
      #see the treatment in factor of $\mathbf{1}_{[t,\infty[}(\tau)$ *in (23)*
      #s_p *is the time remaining to reach census*
      #move the particle p
      $x_p\ -= \mathbf{v}_p s_p$,
      #set the life time of particle p to zero:
      $s_p = 0$
      #do not change the angle or velocity of particle p
      #do not change the weight of particle p
      #tally the contribution of particle p
      $u(x,t,\mathbf{v})\ += w_p \times u_0(x_p, \mathbf{v}_p)$
    end
    else
      #see the recursive treatment in factor of $\mathbf{1}_{[0,t]}(\tau)$ *in (23)*
      #move the particle p
      $x_p\ -= \mathbf{v}_p \tau$,
      Sample the velocity of particle p from $P(x_p, s_p, \mathbf{v}_p, \mathbf{v}')\,\mathrm{d}\mathbf{v}'$
      $\mathbf{v}_p = \mathcal{V}$
      #set the life time of particle p to:
      $s_p\ -= \tau$
      #change its weight
      $w_p\ \times = \dfrac{\sigma_s(x_p, s_p, \mathbf{v}_p)}{\sigma_t(x_p, s_p, \mathbf{v}_p)}$
    end
  end
end

Appendix D. Algorithmic treatments to solve (3) with a gPC-i-MC scheme

Algorithm 2: The gPC-intrusive MC semi-analog scheme described in term of algorithmic operations in order to compute (backward) the gPC coefficients of $u(x,t,\mathbf{v},X)$. The differences with the classical canvas are highlighted in blue: they concern the initial sampling (X_p from $\mathrm{d}\mathcal{P}_X$), the tallying (estimated the gPC coefficients $(u_k^X(x,t,\mathbf{v}))_{k\in\{0,..,P\}}$ intrusively) and the different calls to cross-sections.

for $k \in \{0,...,P\}$ **do**
  set $u_k^X(x,t,\mathbf{v}) = 0$
end
for $p \in \{1,...,N_{MC}\}$ **do**
  set $s_p = t$ *#this will be the remaining life time of particle p, it must go down to zero (backward)*
  set $x_p = x$
  set $\mathbf{v}_p = \mathbf{v}$
  set $w_p = \dfrac{1}{N_{MC}}$
  Sample X from the distribution having probability measure $\mathrm{d}\mathcal{P}_X$.
  set $X_p = X$
  while $s_p > 0$ *and* $w_p > 0$ **do**
    if $x_p \notin \mathcal{D}$ **then**
      #here a very general function for the application of arbitrary uncertain boundary conditions
      apply_boundary_conditions($x_p, s_p, \mathbf{v}_p, X_p$)
    end
    Sample τ from the distribution having probability measure $f(z_s^p, X_p)\,\mathrm{d}s$.
    if $\tau > s_p$ **then**
      #see the treatment in factor of $\mathbf{1}_{[t,\infty[}(\tau)$ *in* (23)
      #s_p *is the time remaining to reach census*
      #move the particle p
      $x_p \ -= \mathbf{v}_p s_p$,
      #set the life time of particle p to zero:
      $s_p = 0$
      #do not change the angle or velocity of particle p
      #do not change the weight of particle p
      #tally the contribution of particle p
      for $k \in \{0,...,P\}$ **do**
        $u_k^X(x,t,\mathbf{v}) += w_p \times u_0(x_p, \mathbf{v}_p, X_p)\phi_k^X(X_p)$
      end
    end
    else
      #see the recursive treatment in factor of $\mathbf{1}_{[0,t]}(\tau)$ *in* (23)
      #move the particle p
      $x_p \ -= \mathbf{v}_p \tau$,
      Sample the velocity of particle p from $P(x_p, s_p, \mathbf{v}_p, \mathbf{v}', X_p)\,\mathrm{d}\mathbf{v}'$
      $\mathbf{v}_p = \mathcal{V}$
      #set the life time of particle p to:
      $s_p \ -= \tau$
      #change its weight
      $w_p \ \times = \dfrac{\sigma_s(x_p, s_p, \mathbf{v}_p, X_p)}{\sigma_t(x_p, s_p, \mathbf{v}_p, X_p)}$
    end
  end
end

Let us briefly tackle parallelism considerations. We insist that the operations to compute one uncertain MC particle remain independent of the others. This would not be the case for a numerical strategy based on the resolution of (37) for example. Consequently, in term of parallel modification, the canvas between Algorithm 1 and Algorithm 2 does not change. The same parallel strategies apply and the parallel reduction/communications/synchronizations are at the same places within the code. There are still few differences (some are illustrated in the benchmarks of section 5):

- Distributed parallel strategies (such as *domain replication* or *domain decomposition*, see [15,26,54,55,75]): domain replication has been discussed through the examples of section 5: Algorithm 2 does imply a reduction on more arrays ($P+1$ quantities to average over the number of replications $N_{\text{replication}}$). The size of the communication is consequently more important (see Table 1). Fortunately, the MC scheme being unconditionally stable, those reduction can occur only at the observation locations/times of interest and are consequently scarse or of reasonable volume during the computation. Domain decomposition has not been tackled in this document. It is not the most common parallel strategy for MC simulation codes. It consists in decomposing the simulation domain $\mathcal{D}$ into $N_{\text{decomposition}}$ ones and applying the numerical scheme on every one of them. Domain decomposition has not been implemented with gPC-i-MC but should not

induce more difficulties than for the classical treatment: particles would still be buffered and communicated between subdomains, see [26]. Now, each uncertain particle carries an additional field of size Q, hence a bigger volume to be communicated.

– Shared memory strategies have only been hinted at in section 5.3 but surely present an interest, especially vectorisation strategies for the tallying phase which now bears the main computational load (especially as P or Q grow, see section 5.3). Once again, the idea is in the same vein as the one in [14] for the vectorisation of *on-the-fly* Doppler broadening: the gPC basis being orthonormal, there exists a three-term recurrence formulae [29,85] allowing to express ϕ_k^X recursively as a function of $\phi_{k-1}^X, \phi_{k-2}^X$. Denote by $\Phi_P^X = (\phi_0^X, ..., \phi_P^X)^t$ the vector of $P+1$ components of the sequence of orthonormal polynomials associated to X, this ensures there exists a matrix J_P^X (*Jacobi matrix*, see [29]) such that

$$\sqrt{\beta_P}\phi_{P+1}^X(X)e_P = J_P^X\Phi_P^X(X) - X\Phi_P^X(X), \tag{D.1}$$

where $e_P = (0, ..., 0, 1)^t$ of size $P+1$ and

$$J_P^X = \begin{pmatrix} \alpha_1 & \sqrt{\beta_1} & 0 & 0 & ... & 0 \\ \sqrt{\beta_1} & \alpha_2 & \sqrt{\beta_2} & 0 & ... & 0 \\ 0 & \sqrt{\beta_2} & \alpha_3 & \sqrt{\beta_3} & ... & 0 \\ 0 & ... & ... & ... & ... & 0 \\ 0 & ... & ... & ... & ... & \sqrt{\beta_P} \end{pmatrix}. \tag{D.2}$$

The latter matrix only depends on the uncertain input distributions and would not change during the whole computation. We here claim no originality, such matrix-vector products are at the basis of Chebyshev (or modified Chebyshev) algorithms [29] which can take advantage of vectorisation strategies. The gPC basis could then also be computed more efficiently *on-the-fly* for every uncertain MC particle and the cost of the gPC-i-MC scheme could be reduced. This subject will be the purpose of future studies.

References

[1] S. Acharjee, N. Zabaras, Uncertainty propagation in finite deformations - a spectral stochastic Lagrangian approach, Comput. Methods Appl. Mech. Eng. 195 (2006) 2289–2312.
[2] N.I. Akhiezer, The Classical Moment Problem, Oliver and Boyd, 1965.
[3] A. Atkinson, A. Donev, R. Tobias, Optimum Experimental Designs, with SAS, Oxford Statistical Science Series, OUP Oxford, 2007.
[4] Adrien Bernede, Gaël Poëtte, An unsplit Monte-Carlo scheme for the resolution of the linear Boltzmann equation coupled to (stiff) Bateman equations, J. Comput. Phys. (2018).
[5] P.L. Bhatnagar, E.P. Gross, M. Krook, A model for collision processes in gases. I. Small amplitude processes in charged and neutral one-component systems, Phys. Rev. 94 (1954) 511–525.
[6] G.A. Bird, Molecular Gas Dynamics and the Direct Simulation of Gas Flows, vol. 1, Clarendon Press, 1994.
[7] Alexandre Birolleau, Gaël Poëtte, Didier Lucor, Adaptive Bayesian inference for discontinuous inverse problems, application to hyperbolic conservation laws, Commun. Comput. Phys. 16 (1) (2014) 1–34.
[8] X. Blanc, C. Bordin, G. Kluth, G. Samba, Variance reduction method for particle transport equation in spherical geometry, J. Comput. Phys. 364 (2018) 274–297.
[9] G. Blatman, Adaptive Sparse Polynomial Chaos Expansions for Uncertainty Propagation and Sensitivity Analysis, Thèse de doctorat, Université Blaise Pascal - Clermont II, 2009.
[10] G. Blatman, B. Sudret, Sparse polynomial chaos expansions and adaptive stochastic finite elements using a regression approach, C. R., Méc. 336 (2008) 518–523.
[11] G. Blatman, B. Sudret, Efficient computation of global sensitivity indices using sparse polynomial chaos expansions, Reliab. Eng. Syst. Saf. 95 (2010) 1216–1229.
[12] A.V. Bobylev, K. Nanbu, Theory of collision algorithms for gases and plasmas based on the Boltzmann equation and the Landau-Fokker-Planck equation, Phys. Rev. E 61 (Apr. 2000) 4576–4586.
[13] B. Iooss, P. Lemaître, A review on global sensitivity analysis methods, in: Uncertainty Management in Simulation-Optimization of Complex Systems: Algorithms and Applications, Springer, 2015.
[14] Emeric Brun, Stéphane Chauveau, Fausto Malvagi, Patmos: a prototype Monte Carlo transport code to test high performance architectures, in: Proceedings of International Conference on Mathematics & Computational Methods Applied to Nuclear Science & Engineering, Jeju, Korea, 2017.
[15] T.A. Brunner, P.S. Brantley, An efficient, robust, domain-decomposition algorithm for particle Monte Carlo, J. Comput. Phys. 10 (2009) 3882.
[16] Dan G. Cacucci, Handbook of Nuclear Engineering, vol. 1, Springer Science & Business Media, 2010.
[17] Russel E. Caflisch, Monte Carlo and quasi-Monte Carlo methods, Acta Numer. 7 (1998) 1–49.
[18] R.H. Cameron, W.T. Martin, The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals, Ann. Math. 48 (1947) 385–392.
[19] J. Castor, Radiation Hydrodynamics, Cambridge University Press, 2004.
[20] T. Crestaux, Polynômes de Chaos pour la Propagation et la Quantification d'Incertitudes, Technical report, CEA, 2006.
[21] M.K. Deb, I.M. Babuska, J.T. Oden, Solution of stochastic partial differential equations using Galerkin finite element techniques, Comput. Methods Appl. Mech. Eng. 190 (2001) 6359–6372.
[22] Bruno Després, Gaël Poëtte, Didier Lucor, Robust uncertainty propagation in systems of conservation laws with the entropy closure method, in: Uncertainty Quantification in Computational Fluid Dynamics, in: Lecture Notes in Computational Science and Engineering, vol. 92, 2013.
[23] J.J. Duderstadt, W.R. Martin, Transport Theory, Wiley, 1979.
[24] J. Dufek, W. Gudowski, Stochastic approximation for Monte-Carlo calculation of steady-state conditions in thermal reactors, Nucl. Sci. Eng. 152 (2006) 274–283.
[25] Jan Dufek, Dan Kotlyar, Eugene Shwageraus, Jaakko Leppänen, The stochastic implicit Euler method: a stable coupling scheme for Monte Carlo burnup calculations, Ann. Nucl. Energy 60 (2013) 295–300.

[26] D. Dureau, G. Poëtte, Hybrid parallel programming models for AMR neutron Monte-Carlo transport, in: Joint International Conference on Supercomputing in Nuclear Applications + Monte-Carlo, Number 04202 in Parallelism and HPC, Monte-Carlo, 2013.
[27] V.V. Fedorov, Theory of optimal experiments, in: Probability and Mathematical Statistics, Elsevier Science, 1972.
[28] Gautier Dakin, Bruno Desprès, Stéphane Jaouen, Hervé Jourdren, High-order accurate Lagrange-remap hydrodynamic schemes on staggered Cartesian grids, CEA GAMNI, January 2016. Poster.
[29] Walter Gautschi, Orthogonal Polynomials: Applications and Computation, vol. 5, Oxford University Press, 1996.
[30] M.I. Gerritsma, J.-B. van der Steen, P. Vos, G.E. Karniadakis, Time-dependent generalized polynomial chaos, J. Comput. Phys. (2010) 8333–8363.
[31] R.G. Ghanem, Ingredients for a general purpose stochastic finite element formulation, Comput. Methods Appl. Mech. Eng. 168 (1999) 19–34.
[32] R.G. Ghanem, J. Red-Horse, Propagation of uncertainty in complex physical systems using a stochastic finite elements approach, Physica D 133 (1999) 137–144.
[33] R.G. Ghanem, P. Spanos, Stochastic Finite Elements: A Spectral Approach, Springer-Verlag, 1991.
[34] R.G. Ghanem, P.D. Spanos, Stochastic Finite Elements: A Spectral Approach, Dover, 1991.
[35] F. Golse, G. Allaire, Transport et Diffusion, Polycopié de cours, 2015.
[36] J.H. Halton, Algorithm 247: radical-inverse quasi-random point sequence, Commun. ACM 7 (12) (December 1964) 701–702.
[37] F. Hermeline, A discretization of the multigroup pn radiative transfer equation on general meshes, J. Comput. Phys. 313 (2016) 549–582.
[38] T.D. Hien, M. Kleiber, Stochastic finite element modeling in linear transient heat transfer, Comput. Methods Appl. Mech. Eng. 144 (1997) 111–124.
[39] Aarno Isotalo, Computational Methods for Burnup Calculations with Monte Carlo Neutronics, PhD. Thesis, Aalto University School of Science Department of Applied Physics, 2013.
[40] A.E. Isotalo, J. Leppänen, J. Dufek, Preventing xenon oscillations in Monte Carlo burnup calculations by enforcing equilibrium xenon distribution, Ann. Nucl. Energy 60 (2013) 78–85.
[41] H. Jourdren, S. Del Pino, Arbitrary high-order schemes for the linear advection and wave equations: application to hydrodynamics and aeroacoustics, C. R. Math. 342 (2006) 441–446.
[42] A. Keese, A General Purpose Framework for Stochastic Finite Elements, Ph.D. thesis, Mathematik und Informatik der Technischen Universitat Braunschweig, 2004.
[43] J. Kusch, M. Frank, Intrusive methods in uncertainty quantification and their connection to kinetic theory, Int. J. Adv. Eng. Sci. Appl. Math. (2018) 1–16.
[44] J. Kusch, G.W. Alldredge, M. Frank, Maximum-principle-satisfying second-order intrusive polynomial moment scheme, arXiv preprint, arXiv:1712.06966, 2017.
[45] B. Lapeyre, E. Pardoux, R. Sentis, Méthodes de Monte Carlo pour les équations de transport et de diffusion, Mathématiques & Applications, vol. 29, Springer-Verlag, 1998.
[46] O.P. Le Maître, O.M. Knio, Uncertainty propagation using Wiener-Haar expansions, J. Comput. Phys. 197 (2004) 28–57.
[47] O.P. Le Maître, O.M. Knio, A stochastic particle-mesh scheme for uncertainty propagation in vortical flows, J. Comput. Phys. 226 (2007) 645–671.
[48] O. Le Maitre, M. Reagan, H. Najm, R. Ghanem, O. Knio, Multi-resolution analysis of Wiener-type uncertainty propagation schemes, J. Comput. Phys. 197 (2004) 502–531.
[49] R. Lebrun, A. Dutfoy, A generalization of the Nataf transformation to distributions with elliptical copula, Probab. Eng. Mech. 24 (2) (2009) 172–178.
[50] R. Lebrun, A. Dutfoy, An innovating analysis of the Nataf transformation from the copula viewpoint, Probab. Eng. Mech. 24 (3) (2009) 312–320.
[51] E.E. Lewis, W.F. Miller Jr., Computational Methods of Neutron Transport, John Wiley and Son, New York, 1984.
[52] R.B. Lowrie, J.E. Morel, J.A. Hittinger, The coupling of radiation and hydrodynamics, Astrophys. J. 521 (August 1999) 432–450.
[53] D. Lucor, C. Enaux, H. Jourdren, P. Sagaut, Multi-physics stochastic design optimization: application to reacting flows and detonation, Comput. Methods Appl. Mech. Eng. 196 (2007) 5047–5062.
[54] A. Majumdar, Parallel performance study of Monte Carlo photon transport code on shared-, distributed-, and distributed-shared-memory architectures, in: Proceedings 14th International Parallel and Distributed Processing Symposium, 2000.
[55] William R. Martin, Tzu-Chiang Wan, Tarek S. Abdel-Rahman, Trevor N. Mudge, Kenichi Miura, Monte Carlo photon transport on shared memory and distributed memory parallel processors, Int. J. Supercomput. Appl. High Perform. Comput. 1 (3) (1987) 57–74.
[56] J.-M. Martinez, J. Cahen, A. Millard, D. Lucor, F. Huvelin, J. Ko, N. Poussineau, Modélisation des Incertitudes par Polynômes de Chaos – Étude d'un Écoulement en Milieux Poreux, Technical Report Rapport DM2S/DIR/RT/06-006/A, CEA-CEMRACS, 2006.
[57] Youssef Marzouk, Dongbin Xiu, A stochastic collocation approach to Bayesian inference in inverse problems, Commun. Comput. Phys. 6 (4) (2009) 826–847.
[58] L. Mathelin, O.P. Le Maître, A posteriori error analysis for stochastic finite element solutions of fluid flows with parametric uncertainties, in: ECCOMAS CFD, 2006.
[59] H.G. Matthies, A. Keese, Galerkin methods for linear and nonlinear elliptic stochastic PDEs, Comput. Methods Appl. Mech. Eng. 31 (1–2) (2003) 179–191.
[60] J. Mercer, Functions of positive and negative type and their connection with the theory of integral equations, Philos. Trans. R. Soc. 209 (1909).
[61] M. Meyer, H.G. Matthies, Efficient model reduction in non-linear dynamics using the Karhunen-Loève expansion and dual-weighted-residual methods, in: Informatikbericht 2003-08, in: Comp. Meth. Appl. Mech. Eng., TU Braunschweig, Germany, 2004.
[62] D. Mihalas, B.W. Mihalas, Foundations of Radiation Hydrodynamics, Dover Books on Physics, Dover Publications, 1999.
[63] H.N. Najm, O.P. Le Maître, O.M. Knio, R.G. Ghanem, A stochastic projection method for fluid flow I: basic formulation, J. Comput. Phys. 173 (2001) 481–511.
[64] Harald Niederreiter, Random Number Generation and Quasi-Monte Carlo Methods, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1992.
[65] T. N'Kaoua, Solution of the nonlinear radiative transfer equations by a fully implicit matrix Monte Carlo method coupled with the Rosseland diffusion equation via domain decomposition, SIAM J. Sci. Stat. Comput. 12 (3) (March 1991) 505–520.
[66] G.C. Papanicolaou, Asymptotic analysis of transport processes, Bull. Am. Math. Soc. 81 (2) (1975).
[67] Joel A. Paulson, Edward A. Buehler, Ali Mesbah, Arbitrary polynomial chaos for uncertainty propagation of correlated random variables in dynamic systems, IFAC-PapersOnLine 50 (1) (2017) 3548–3553, 20th IFAC World Congress.
[68] B. Perthame, Transport Equations in Biology, Birkhauser Verlag, Basel, Boston, Berlin, 2000.
[69] G. Poëtte, D. Lucor, Non intrusive iterative stochastic spectral representation with application to compressible gas dynamics, J. Comput. Phys. (2011), https://doi.org/10.1016/j.jcp.2011.12.038.
[70] G. Poëtte, B. Després, D. Lucor, Uncertainty quantification for systems of conservation laws, J. Comput. Phys. 228 (7) (2009) 2443–2467.
[71] G. Poëtte, B. Després, D. Lucor, Treatment of uncertain interfaces in compressible flows, Comput. Methods Appl. Mech. Eng. 200 (2010) 284–308.
[72] Gaël Poëtte, Didier Lucor, Hervé Jourdren, A stochastic surrogate model approach applied to calibration of unstable fluid flow experiments, C. R. Math. 350 (5) (2012) 319–324.
[73] Gaël Poëtte, Alexandre Birolleau, Didier Lucor, Iterative polynomial approximation adapting to arbitrary probability distribution, SIAM J. Numer. Anal. 53 (3) (2015) 1559–1584.

[74] G.C. Pomraning, The Equations of Radiation Hydrodynamics, Dover Books on Physics, Dover Publications, 1973.
[75] Richard Procassini, Matthew O'Brien, Janine Taylor, Load Balancing of Parallel Monte Carlo Transport Calculations, 2005.
[76] G. Saporta, Probabilités, Analyse de Données et Statistique, 2e édition, Technip, 2006.
[77] L. Schlachter, F. Schneider, A hyperbolicity-preserving stochastic Galerkin approximation for uncertain hyperbolic systems of equations, arXiv preprint, arXiv:1710.03587, 2017.
[78] I.M. Sobol, Uniformly distributed sequences with an additional uniform property, USSR Comput. Math. Math. Phys. 16 (5) (1976) 236–242.
[79] I. Sobol, Sensitivity estimates for nonlinear mathematical models, Mat. Model. 2 (1990) 112–118 in Russian;
Translated in English: I. Sobol, Sensitivity analysis for non-linear mathematical models, Math. Model. Comput. Exp. 1 (1993) 407–414.
[80] J. Spanier, E.M. Gelbard, Monte Carlo Principles and Neutron Transport Problems, Addison-Wesley, 1969.
[81] P. Spanos, R.G. Ghanem, Stochastic finite element expansion for random media, J. Eng. Mech. 115 (5) (1989) 1035–1053.
[82] B. Sudret, Uncertainty Propagation and Sensitivity Analysis in Mechanical Models, Contribution to Structural Reliability and Stochastic Spectral Methods, Habilitation à Diriger des Recherches, Université Blaise Pascal - Clermont II, 2007.
[83] B. Sudret, A. Der Kiureghian, Stochastic Finite Element Methods and Reliability - A State of the Art Report, Technical Report UCB/SEMM-2000/08, Department of Civil and Environmental Engineering, University of California, Berkeley, 2000.
[84] B. Sudret, Z. Guédé, M. Lemaire, Life-time reliability based assessment of structures submitted to thermal fatigue, Int. J. Fatigue 29 (7) (2006) 1359–1373.
[85] G. Szego, Orthogonal Polynomials, vol. 23, Colloquium Publications, American Mathematical Society, 1939.
[86] A. Tarantola, Inverse Problem Theory and Methods for Model Parameter Estimation, SIAM, Philadelphia, 2005.
[87] R.A. Todor, C. Schwab, Karhunen-Loève approximation of random fields by generalized fast multipole methods, J. Comput. Phys. 217 (1) (2006) 100–122.
[88] Lloyd N. Trefethen, Is Gauss quadrature better than Clenshaw-Curtis?, SIAM Rev. 50 (1) (2008) 67–87.
[89] J. Tryoen, O. Le Maître, A. Ern, Adaptive anisotropic stochastic discretization schemes for uncertain conservation laws, in: Proc. of ASME 2010, Third Joint US-European Fluids Engineering Summer Meeting, 2010.
[90] J.G. van der Corput, Verteilungsfunktionen I. Mitt, Proc. Akad. Wet. Amsterdam 38 (1935) 813–821.
[91] P. Vos, Time Dependent Polynomial Chaos, Master of science thesis, Delft University of Technology, Faculty of Aerospace Engineering, 2006.
[92] X. Wan, G.E. Karniadakis, Stochastic heat transfer enhancement in a grooved channel, J. Fluid Mech. 565 (2006) 255–278.
[93] X. Wan, G.E. Karniadakis, Beyond Wiener-Askey expansions: handling arbitrary PDFs, SIAM J. Sci. Comput. 27 (1–3) (2006).
[94] X. Wan, G.E. Karniadakis, Long-term behavior of polynomial chaos in stochastic flow simulations, Comput. Methods Appl. Mech. Eng. 195 (2006) 5582–5596.
[95] X. Wan, G.E. Karniadakis, Multi-element generalized polynomial chaos for arbitrary probability measures, SIAM J. Sci. Comput. 28 (3) (2006) 901–928.
[96] Yunsong Wang, Optimization of Monte Carlo Neutron Transport Simulations with Emerging Architectures, PhD thesis, Université Paris-Saclay, 2017, Thèse de doctorat dirigée par Calvin, Christophe Informatique Paris Saclay 2017.
[97] N. Wiener, The homogeneous chaos, Am. J. Math. 60 (1938) 897–936.
[98] B.R. Wienke, Transport equations in moving material part I: neutrons and photons, Prog. Nucl. Energy 46 (1) (2005) 13–55.
[99] B.R. Wienke, T.R. Hill, P.P. Whalen, Multigroup particle transport in a moving material, J. Comput. Phys. 72 (1) (1987) 177–201.
[100] J.A.S. Witteveen, H. Bijl, An unsteady adaptive stochastic finite elements formulation for rigid-body fluid-structure interaction, Compos. Struct. (2008).
[101] D. Xiu, G.E. Karniadakis, A new stochastic approach to transient heat conduction modeling with uncertainty, Int. J. Heat Mass Transf. 46 (2003) 4681–4693.
[102] D. Xiu, D. Lucor, G.E. Karniadakis, Stochastic modeling of flow-structure interactions, in: K.J. Bathe (Ed.), Computational Fluid and Solid Mechanics, Elsevier, Proceedings of the 1st MIT Conference, Cambridge, Massachusetts, vol. 2, 2001, pp. 1420–1423.