



Monte Carlo advances and concentrated solar applications

J. Delatorre^a, G. Baud^a, J.J. Bézian^{a,*}, S. Blanco^b, C. Caliot^c, J.F. Cornet^d, C. Coustet^g,
J. Dauchet^d, M. El Hafi^a, V. Eymet^a, R. Fournier^b, J. Gautrais^e, O. Gourmel^f, D. Joseph^g,
N. Meilhac^b, A. Pajot^f, M. Paulin^f, P. Perez^h, B. Piaud^g, M. Rogerⁱ, J. Rolland^a,
F. Veynandt^a, S. Weitz^b

^a Laboratoire RAPSODEE - UMR 5302 - ENSTIMAC - Campus Jarlard - 81013 Albi CT Cedex 09, France

^b LAPLACE, UMR 5213 - Université Paul Sabatier, 118, Route de Narbonne - 31062 Toulouse Cedex, France

^c PROMES - UPR CNRS 8521 - 7, rue du Four Solaire, 66120 Font Romeu Odeillo, France

^d GEPEA - UMR CNRS 6144, boulevard de l'Université, CRTT-BP 406, 44602 Saint-Nazaire Cedex, France

^e Centre de Recherche sur la Cognition Animale - UMR 5169 - Université Paul Sabatier, 118, Route de Narbonne - 31062 Toulouse Cedex, France

^f IRIT - CNRS, UMR 5505 - Université Paul Sabatier, 118, Route de Narbonne - 31062 Toulouse Cedex, France

^g HPC - SA - 3, chemin du pigeonier de la cépière, Bâtiment C - 31100 Toulouse, France

^h ONERA/DOta Chemin de la Humière, 91761 Palaiseau, France

ⁱ Mechanical Engineering Department, Instituto Superior Técnico/IDMEC - Technical University of Lisbon, Av. Rovisco Pais, 1049-001 Lisboa, Portugal

Communicated by: Associate Editor L. Vant-Hull

Abstract

The Monte Carlo method is partially reviewed with the objective of illustrating how some of the most recent methodological advances can benefit to concentrated solar research. This review puts forward the practical consequences of writing down and handling the integral formulation associated to each Monte Carlo algorithm. Starting with simple examples and up to the most complex multiple reflection, multiple scattering configurations, we try to argue that these formulations are very much accessible to the non specialist and that they allow a straightforward entry to sensitivity computations (for assistance in design optimization processes) and to convergence enhancement techniques involving subtle concepts such as control variate and zero variance. All illustration examples make PROMES - UPR CNRS 8521 - 7, rue du Four Solaire, 66120 Font Romeu Odeillo, France use of the public domain development environment EDStar (including advanced parallelized computer graphics libraries) and are meant to serve as start basis either for the upgrading of existing Monte Carlo codes, or for fast implementation of ad hoc codes when specific needs cannot be answered with standard concentrated solar codes (in particular as far as the new generation of solar receivers is concerned).

© 2013 Elsevier Ltd. All rights reserved.

Keywords: Monte Carlo algorithm; Concentrated solar energy; Solar energy flux density distribution; Solar concentrators design optimization; Sensitivity computation

1. Introduction

Concentrated solar processes such as solar thermal power plants essentially convert solar energy into high density heat fluxes to be collected by working fluids. As far as radiation is

concerned, such processes involve both a concentrating system (an heliostat or an heliostat field) and a receiver. Concentrating systems and first generation receivers were initially designed using geometrical optics, but higher accuracy requirements or advanced receiver technologies imply a deeper entry into physical optics and radiative transfer. This is commonly translated into the development of Monte Carlo codes. They are preferred to other numerical

* Corresponding author. Tel.: +33 563 493 292.

E-mail address: bezian@mines-albi.fr (J.J. Bézian).

simulation techniques because of their flexibility (in terms of inclusion of new physical phenomena) and their ability to deal with geometrically complex realistic systems. But this trend introduces new methodological questions:

- Can we fill the gap between the statement of flexibility and the practical implementation, for instance, of a Monte Carlo code simulating such a complex system as a full heliostat field and a hyperbolic mirror focused on a fluidized bed receiver including multiple reflections at metallic boundaries and multiple scattering by silicate particles?
- Can the computation requirements of Monte Carlo codes be reduced to become compatible with design optimization procedures that engineers have set up thinking of fast geometrical optics convolution methods?
- More positively, can we benefit of the specific features of Monte Carlo methods as far as physical interpretation of simulation results is concerned, that is to say of sensitivity evaluation, adjoint computation, zero-variance formulation, integral reformulation, etc., as ways to understand the impacts of adjustable design parameters?

These three questions (practical implementation, convergence enhancement and identification of leading physical mechanisms) extend way beyond concentrated solar applications and are at the origin of intensive research efforts of Monte Carlo specialists dealing with all kinds of particle transport physics. The objective of the present text is precisely to review those among the corresponding methodological advances that we evaluate as mature enough to directly contribute to today's concentrated solar research.

Among them, a particular attention is devoted to the zero-variance concept and the sensitivity evaluation techniques. Both represent recent significant advances and we try to argue that they are very much accessible to the non-specialist. For didactic reasons, only parts of these two advances are fully exposed: we only consider the part of the zero-variance concept that can be exposed without the use of adjoint formulations and the most subtle part of sensitivity evaluation techniques, corresponding to domain deformation sensitivities, is introduced but is not illustrated. The missing parts would require deeper developments and we hope that the present text will serve as an introduction for the reader wishing to enter the corresponding specialized literature. Another point is also strongly highlighted: today's easy access to the main concepts and computation tools issued of the computer graphics research community during the last twenty years (in particular under the solicitation of the film and game industries). This constitutes a very significant help for radiation physicists in the process of geometrically defining (modeling) complex scenes and accelerating photon tracking in such scenes. The details of these techniques are avoided here as they strictly belong to the field of computer sciences and are exposed in excellent textbooks, but we try to be as accurate as possible concerning the way they can be translated for practical use in the concentrated

solar context. This objective is addressed using the public domain development environment EDStar, a simple implementation strategy example that may be helpful to code designers. EDStar also allows us to provide, on an associated website, the documented codes corresponding to each of the following concentrated solar illustrative examples. The presentation is organized so that these successive examples are representative of the wide diversity of concentrated solar systems and are of increasing complexity, up to quite realistic configurations meant to serve as start basis for researchers facing the need for rapid implementation or upgrade of dedicated Monte Carlo codes, when their questions extend beyond the scope of available simulation tools.

A strong presentation choice was made that consists in putting forward, in a systematic manner, the integral formulation associated to each Monte Carlo algorithm. This led us to a set of coherent conventions and notations that may require an adaptation effort, but the expected reader's benefit is that all methodological questions are then addressed on a common basis, from the most theoretical integral reformulation exercises, to the most practical aspects of code implementation.

The article is organized the following way: the methodological review itself is exposed in Section 2 and the more practical consideration related to implementation are the object of a short separate section (Section 3); four concentrated solar examples are then detailed in Section 4 and the last section (Section 5) is devoted to a brief classification of what we perceive as the main remaining open questions.

2. Integral formulation for variance reduction and sensitivity analysis

Since Metropolis original work in 1949 (Metropolis and Ulam, 1949), numerous monographs and review articles have been devoted to the Monte Carlo method. In the present context, we are essentially concerned by the simulation of a linear transport phenomenon (photons do not interact, neither directly, nor indirectly in most of the above listed radiative transfer problems). But even concentrating on this subclass of Monte Carlo algorithms, numerous excellent specialized monographs and reviews are available. We choose, very arbitrarily, to point out here Hammersley and Handscomb's monograph (Hammersley and Handscomb, 1964) because of the everlasting influence of this short synthesis in the community and Howell's review (Howell, 1998) because of its proximity with our more specific engineering application concerns. These texts provide a sufficient theoretical framework for most of the algorithms commonly encountered in concentrated solar research and may serve as a meaningful ground basis for any further bibliographic research. We also invite the reader wishing to enter more deeply in the field of linear transport Monte Carlo to explore the particle transport literature (Anderson, 1986; Allison, 2006; Agostinelli, 2003), with an emphasis on the neutron transport and plasma physics literature

(Spanier and Gelbard, 2008; Azmy and Sartor, 2010; Doolen and Hendricks, 1987; Kalos and Whitlock, 1986; Lux and Koblinger, 1991; Glouckov et al., 2003), and the computer graphics literature for specific questions associated to complex geometries (Veatch, 1998; Dutre et al., 2002; Pharr and Humphreys, 2004).

Among more recent methodological advances, the zero-variance concept and the sensitivity estimation theoretical framework are detailed hereafter, as they can be at least partially translated into simple systematic procedures in the concentrated solar context. Both rely on a full explicitation of the strict relationship between a linear transport Monte Carlo algorithm and an integral transport formulation. This relationship is therefore first discussed in Section 2.1, which is also the opportunity to briefly recall the essential steps of designing and optimizing Monte Carlo algorithms. Section 2.1 will also help clarifying the description of the four examples of Section 4, where advanced Monte Carlo algorithms will only be discussed in terms of their corresponding integral formulation. Sections 2.2 and 2.3 deal specifically with zero-variance and sensitivities, with the emphasis on the aspects that are mature enough for immediate use in the present context.

2.1. Integral formulation

Just as linear transport equations are specific (among general transport equations) in the sense that integral solutions can be derived using systematic procedures, linear transport Monte Carlo algorithms are specific in the sense that the sampled *events* are strictly independent, whereas in non-linear transport Monte Carlo algorithms the random history of each particle is statistically dependent on that of the other particles. This means that when considering a physical quantity A , any approximation \tilde{a}_N of A corresponding to a linear Monte Carlo algorithm using N sampled events is constructed as

$$\tilde{a}_N = \frac{1}{N} \sum_{i=1}^N w_i \quad (1)$$

where $w_1, w_2 \dots w_N$ are N independent realizations of a given random variable W . Events can be very simple, as in the standard algorithm for computation of the shape factor F_{01} between two black surfaces S_0 and S_1 (the fraction of the radiation emitted by S_0 that impacts S_1), where N straight rays are sampled starting from S_0 and where the *weight* w_i takes the value 1 if the i th ray intersects S_1 and 0 otherwise. On the contrary, as illustrated in Section 4, events become rapidly quite complex as soon as multiple reflection and multiple scattering phenomena are simulated. But in all cases events are independent, which has two essential consequences: a meaningful statistical uncertainty (an evaluation of the standard deviation of the estimator) is systematically available as

$$\tilde{\sigma}_N = \frac{1}{\sqrt{N-1}} \sqrt{\left(\frac{1}{N} \sum_{i=1}^N w_i^2 \right) - \tilde{a}_N^2} \quad (2)$$

and parallel versions of the algorithm are easy to implement (see Section 3.2).

Let us first consider the above mentioned shape factor example: the geometrical notations, the Monte Carlo algorithm and its corresponding integral formulation are given in Fig. 1. The meaning of the integral formulation is simply that when $N \rightarrow +\infty$, the estimator \tilde{a}_N evaluates F_{01} as the expectation of the random variable $W = \hat{w}(\mathbf{X}_0, \mathbf{\Omega}_0(\mathbf{X}_0))$, that is to say $E(W) = \int_{S_0} p_{\mathbf{X}_0}(\mathbf{x}_0) d\mathbf{x}_0 \int_{h_0(\mathbf{x}_0)} p_{\mathbf{\Omega}_0}(\mathbf{\omega}_0 | \mathbf{x}_0) d\mathbf{\omega}_0 \hat{w}(\mathbf{x}_0, \mathbf{\omega}_0)$, where \mathbf{X}_0 is a random location uniformly distributed on S_0 and $\mathbf{\Omega}_0(\mathbf{X}_0)$ is a random unit vector distributed according to Lambert law on the hemisphere exiting S_0 at \mathbf{X}_0 . On such a simple example, it is obvious that the integral formulation is a strict translation of the initial Monte Carlo algorithm. We will see later that even with very complex algorithms, the translation remains quite straightforward.

The reverse translation is also very useful. Let us still consider the computation of F_{01} but let us now start from a standard shape factor integral expression:

$$F_{01} = \int_{S_0} d\mathbf{x}_0 \int_{S_1} d\mathbf{x}_1 \frac{[\mathbf{\omega}_0(\mathbf{x}_0, \mathbf{x}_1) \cdot \mathbf{n}_0(\mathbf{x}_0)][-\mathbf{\omega}_0(\mathbf{x}_0, \mathbf{x}_1) \cdot \mathbf{n}_1(\mathbf{x}_1)]}{\pi S_0(\mathbf{x}_1 - \mathbf{x}_0)^2} H(\mathbf{y}_1 = \mathbf{x}_1) \quad (3)$$

where $\mathbf{n}_1(\mathbf{x}_1)$ is the normal to S_1 at location \mathbf{x}_1 and $\mathbf{\omega}_0(\mathbf{x}_0, \mathbf{x}_1) = \frac{\mathbf{x}_1 - \mathbf{x}_0}{\sqrt{(\mathbf{x}_1 - \mathbf{x}_0)^2}}$ is the unit vector from \mathbf{x}_0 to \mathbf{x}_1 . The Heaviside notation $H(\mathbf{y}_1 = \mathbf{x}_1)$ represents the fact that there can be no direct exchange from \mathbf{x}_0 to \mathbf{x}_1 if the first surface impact \mathbf{y}_1 from \mathbf{x}_0 in the direction $\mathbf{\omega}_0$ occurs before \mathbf{x}_1 . In such an integral formulation, by comparison with the integral formulation of Fig. 1, the integration domain is different and no statistical view point is introduced. But one can arbitrarily choose any probability density function $p_{\mathbf{X}_0}$ on S_0 , for instance $p_{\mathbf{X}_0}(\mathbf{x}_0) = \frac{1}{S_0}$, and any probability density function $p_{\mathbf{X}_1}$ on S_1 , for instance $p_{\mathbf{X}_1}(\mathbf{x}_1) = \frac{1}{S_1}$, and report them into Eq. (3) in order to get the expression of the corresponding weight function $\hat{w}(\mathbf{x}_0, \mathbf{x}_1)$:

$$F_{01} = E(W) = \int_{S_0} p_{\mathbf{X}_0}(\mathbf{x}_0) d\mathbf{x}_0 \int_{S_1} p_{\mathbf{X}_1}(\mathbf{x}_1) d\mathbf{x}_1 \hat{w}(\mathbf{x}_0, \mathbf{x}_1) \quad (4)$$

with

$$\hat{w}(\mathbf{x}_0, \mathbf{x}_1) = \frac{1}{p_{\mathbf{X}_0}(\mathbf{x}_0) p_{\mathbf{X}_1}(\mathbf{x}_1)} \frac{[\mathbf{\omega}_0(\mathbf{x}_0, \mathbf{x}_1) \cdot \mathbf{n}_0(\mathbf{x}_0)][-\mathbf{\omega}_0(\mathbf{x}_0, \mathbf{x}_1) \cdot \mathbf{n}_1(\mathbf{x}_1)]}{\pi S_0(\mathbf{x}_1 - \mathbf{x}_0)^2} H(\mathbf{y}_1 = \mathbf{x}_1)$$

And now this integral formulation can be easily translated into a new Monte Carlo algorithm as illustrated in Fig. 2.

With these elementary pictures, we can rapidly illustrate the three techniques that are most commonly used to optimize a given Monte Carlo algorithm: importance sampling, control variate and integral reformulation. For the purpose

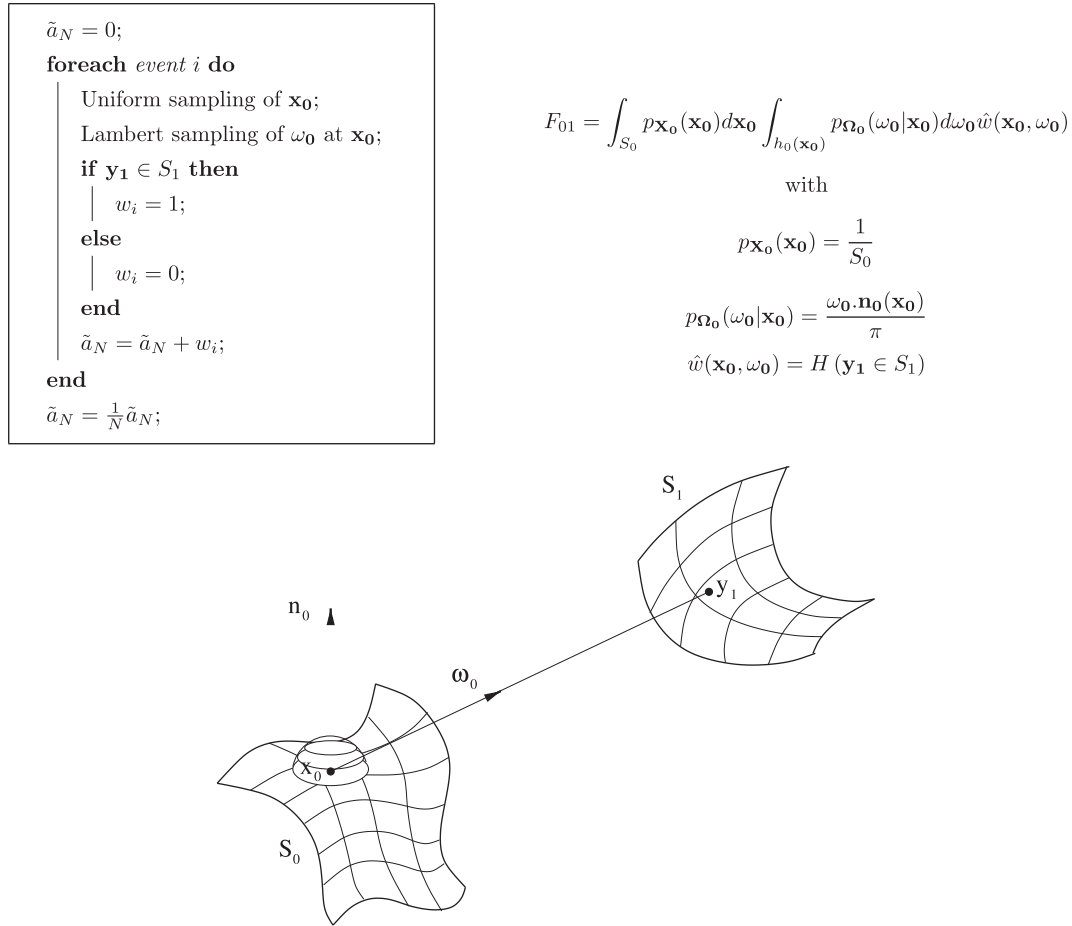


Fig. 1. The standard Monte Carlo algorithm for computation of the shape factor F_{01} from surface S_0 to surface S_1 , and its corresponding integral formulation. \mathbf{x}_0 is a location sampled on S_0 , $\mathbf{n}_0(\mathbf{x}_0)$ is the unit normal exiting S_0 at \mathbf{x}_0 , and ω_0 is a unit vector sampled in the hemisphere $h_0(\mathbf{x}_0)$ exiting S_0 at \mathbf{x}_0 . The location \mathbf{y}_1 is the first intersection of the half-line (\mathbf{x}_0, ω_0) with either S_0 , S_1 , or with the limit of the simulation domain. The Heaviside notation $H(\mathbf{y}_1 \in S_1)$ stands for the value 1 if $\mathbf{y}_1 \in S_1$ and 0 otherwise.

of this illustration, let us consider the very general case of a Monte Carlo algorithm in which:

- a random vector \mathbf{X} of any finite dimension is sampled on its definition domain \mathcal{D}_X according to its probability density function p_X ,
- a random vector \mathbf{Y} of any finite dimension is then sampled on its definition domain \mathcal{D}_Y according to its probability density function p_Y , where \mathcal{D}_Y and p_Y both depend on the sampled value of \mathbf{X} ,
- the Monte Carlo weight is computed as a function \hat{w} of both the sampled value of \mathbf{X} and the sampled value of \mathbf{Y} .

The corresponding integral formulation is

$$A = E(W) = \int_{\mathcal{D}_X} p_X(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_Y(\mathbf{x})} p_Y(\mathbf{y} | \mathbf{x}) d\mathbf{y} \hat{w}(\mathbf{x}, \mathbf{y}) \quad (6)$$

The only lack of generality is that only two random vectors \mathbf{X} and \mathbf{Y} are considered. But all the following statements remain strictly valid for a Monte Carlo algorithm involving a third random vector \mathbf{Z} depending on the realizations of \mathbf{X} and \mathbf{Y} , or even involving an infinite number of random vec-

tors, as we will see with the multiple reflection, multiple scattering receiver example. Note that our first shape factor example belongs to the class of Monte Carlo algorithms covered by Eq. (6): \mathbf{X} is to be related to the location \mathbf{X}_0 and \mathbf{Y} to the direction Ω_0 that is indeed sampled on a definition domain and according to a probability density function that both depend on the sampled value of \mathbf{X}_0 .

Importance sampling. In Eq. (4) and Fig. 2, the choice of $p_{\mathbf{x}_0}$ and $p_{\mathbf{x}_1}$ was arbitrary. Of course this choice has consequences in terms of numerical convergence: the standard deviation $\tilde{\sigma}_N$ is modified when modifying $p_{\mathbf{x}_0}$ and $p_{\mathbf{x}_1}$ and *importance sampling* is the task of optimizing these probability density functions to get small standard deviations with a limited number of sampled events. Obviously, this technique is very easy to implement when the algorithm has been obtained on the basis of a given integral formulation: the \mathbf{X}_0 and \mathbf{X}_1 sampling procedures are replaced by those corresponding to the new probability density functions and the probability density function expressions are modified in the expression of $\hat{w}(\mathbf{x}_0, \mathbf{x}_1)$ in Eq. (5). In the general case, even when the algorithm has been designed on the basis of statistical physics pictures, without writing

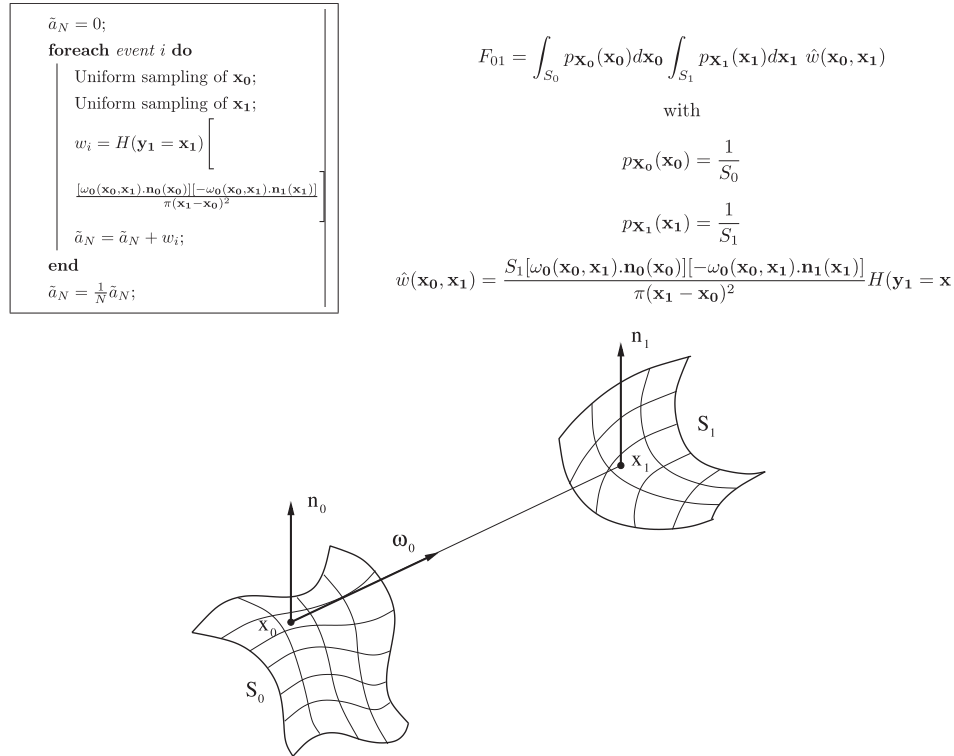


Fig. 2. Monte Carlo algorithm for computation of the shape factor F_{01} from surface S_0 to surface S_1 . This algorithm is constructed on the basis of the standard shape factor integral expression of Eq. (3). ω_0 is the unit vector pointing from the location \mathbf{x}_0 , sampled on S_0 , to the location \mathbf{x}_1 , sampled on S_1 . The notation $H(\mathbf{y}_1 = \mathbf{x}_1)$ stands for the value 1 if \mathbf{x}_1 is the first intersection of the half-line (\mathbf{x}_0, ω_0) with S_1 (no blocking), and 0 otherwise.

down the corresponding integrals, it suffices to know that for linear transport Monte Carlo algorithms an integral translation exists and that its form is comparable to that of Eq. (6). Then, replacing the probability density functions with optimized probability density functions $p_{\mathbf{x},opt}$ and $p_{\mathbf{y},opt}$ leads to some integral of the form

$$A = E(W_{opt}) = \int_{\mathcal{D}_{\mathbf{x}}} p_{\mathbf{x},opt}(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_{\mathbf{y}}(\mathbf{x})} p_{\mathbf{y},opt}(\mathbf{y}|\mathbf{x}) d\mathbf{y} \hat{w}_{opt}(\mathbf{x}, \mathbf{y}) \quad (7)$$

with

$$\hat{w}_{opt}(\mathbf{x}, \mathbf{y}) = \hat{w}(\mathbf{x}, \mathbf{y}) \frac{p_{\mathbf{x}}(\mathbf{x})}{p_{\mathbf{x},opt}(\mathbf{x})} \frac{p_{\mathbf{y}}(\mathbf{y}|\mathbf{x})}{p_{\mathbf{y},opt}(\mathbf{y}|\mathbf{x})} \quad (8)$$

This means that the explicit integral formulation is not required: the algorithm is unchanged except that the optimized probability density functions are now used for sampling of \mathbf{X} and \mathbf{Y} (instead of the original probability density functions) and that the old weight is now simply multiplied by the ratios of the original probability density functions to the optimized probability density functions. However the probability density function choice can be very subtle, with some similarities with the question of choosing an optimized non-uniform grid in deterministic numerical methods.

Control variate. If a function $\hat{v}(\mathbf{x}, \mathbf{y})$ can be found such that

$$\int_{\mathcal{D}_{\mathbf{x}}} p_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_{\mathbf{y}}(\mathbf{x})} p_{\mathbf{y}}(\mathbf{y}|\mathbf{x}) d\mathbf{y} \hat{v}(\mathbf{x}, \mathbf{y}) = 0 \quad (9)$$

then the weight function $\hat{w}(\mathbf{x}, \mathbf{y})$ in Eq. (6) can be replaced by an hopefully optimized weight function $\hat{w}_{opt}(\mathbf{x}, \mathbf{y}) = \hat{w}(\mathbf{x}, \mathbf{y}) + \hat{v}(\mathbf{x}, \mathbf{y})$. Indeed

$$A = \int_{\mathcal{D}_{\mathbf{x}}} p_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_{\mathbf{y}}(\mathbf{x})} p_{\mathbf{y}}(\mathbf{y}|\mathbf{x}) d\mathbf{y} \hat{w}_{opt}(\mathbf{x}, \mathbf{y}) = \int_{\mathcal{D}_{\mathbf{x}}} p_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_{\mathbf{y}}(\mathbf{x})} p_{\mathbf{y}}(\mathbf{y}|\mathbf{x}) d\mathbf{y} [\hat{w}(\mathbf{x}, \mathbf{y}) + \hat{v}(\mathbf{x}, \mathbf{y})] \quad (10)$$

As for importance sampling, the explicit integral formulation is not required: it suffices to know that the pondered integral of \hat{v} is null. A practical example will be mentioned below in which \hat{v} could be chosen using simple physical pictures without any manipulation of the full integral expressions.

Integral reformulation. We have seen with the shape factor example that two distinct integral formulations could be used, leading to two different algorithms. Obviously this is also a way – in practice the most efficient – to reduce the variance of Monte Carlo algorithms. If the extensions of the surfaces S_0 and S_1 are small compared to the distance between the two surfaces, then the standard algorithm (where a location \mathbf{x}_0 is sampled on S_0 and a direction ω_0 is sampled in the exit hemisphere) will show a very slow convergence rate. Indeed most rays will not intersect S_1 , the corresponding weights will be null, and very seldom

an intersection will occur leading to a weight value of 1: consequently the relative standard deviation of the estimator tends to infinity when the distance between the two surfaces tends to infinity. On the contrary, the second algorithm (where a location \mathbf{x}_0 is sampled on S_0 and a location \mathbf{x}_1 is sampled on S_1) leads to a relative standard deviation that tends to zero in the same limit. Depending on the characteristics of the problem, a formulation can therefore be ideally suited or fully unadapted, and a regular practice of Monte Carlo methods implies undeniably that one has grown enough confidence to enter deep into the integral formulation and transform it (on the basis of available physical pictures) when serious convergence difficulties are encountered.

Illustrations. Let us further illustrate these three concepts (importance sampling, control variate and integral reformulation) using a simplified version of the third example of Section 4. A perfectly collimated uniform beam of monochromatic surfacic power density φ_v is incident perpendicularly to the horizontal plane glass window of a fluidized bed receiver. The window, noted \mathcal{G} , is perfectly transparent and all refraction effects are neglected. The other surfaces, noted \mathcal{R} , are diffuse perfectly reflective. The gas–particle mixture filling the volume is purely absorbing (no scattering) with an uniform monochromatic absorption coefficient $k_{a,v}$. One among the various Monte Carlo algorithms allowing the evaluation of the radiative power A absorbed by the volume in a given frequency interval $[v_{min}, v_{max}]$ is detailed in Fig. 3: it starts with the uniform sampling of a frequency v on $[v_{min}, v_{max}]$ and a uniform sampling of an entrance location \mathbf{x}_0 on the window, and then a multiple reflection path is followed where reflection directions are sampled at each surface encounter, until exit through the window.

When translating such an algorithm into an integral formulation, the only additional difficulties (by comparison with the preceding examples) are associated to the integration domain, that is now of infinite dimension, and to the branching tests $\mathbf{x}_j \in \mathcal{G}$ and $\mathbf{x}_j \in \mathcal{R}$. For the branching tests, a generic Heaviside notation can be used where, for instance, $H(\mathbf{x}_j \in \mathcal{G})$ is to be interpreted as $H(\mathbf{x}_j \in \mathcal{G}) = 1$ if $\mathbf{x}_j \in \mathcal{G}$ and $H(\mathbf{x}_j \in \mathcal{G}) = 0$ otherwise. Infinite dimension is a direct consequence of multiple reflection (multiple reflection and multiple scattering in the general case) and the formal difficulty of writing expressions including an infinite number of integral signs can be bypassed, in this specific context, thanks to the Fredholm formalism. For didactic reasons, in the integral formulation of Fig. 3, each of the successive integrals is written down as a translation of each of the successive sampling procedures. But it is worth pointing out that Fredholm formalism is very useful in the process of classifying and analyzing common features of Monte Carlo algorithms. As in the previous shape factor examples, the fact that there is a one-to-one relationship between this integral expression and the algorithm is obvious, and although the integral expressions get rapidly very long, they remain quite easy to interpret. A simple first

consequence is that independently of any attempt to transform these integrals for convergence enhancement, they are at least an alternative to the often very long textual descriptions of algorithms, and they facilitate the task of rigorously controlling that the algorithm indeed converges to the addressed quantity.

Various convergence difficulties can be encountered when using integral formulations such as that of Fig. 3. Let us consider for instance the particular case where the receiver is filled with a gas (no particle) which absorption spectrum is composed of n lorentzian absorption lines of center frequencies $v_{c,1}, v_{c,2} \dots v_{c,n}$, intensities $s_1, s_2 \dots s_n$ and half width at half height δ (see Fig. 4). If δ is small compared to the average distance between successive line centers, it is possible that absorption is only significant at frequencies in the immediate vicinity of line centers. Therefore, as frequency is sampled uniformly on $[v_{min}, v_{max}]$, most sampled events will lead to weight values close to zero and the standard deviation of the estimator will be high. Importance sampling will then be immediately useful if the optimized frequency probability density function favors the frequency regions around the lines. We will come back to this example in Section 2.2. Another example of convergence difficulties is the case where the considered quantity is not the radiative power absorbed by the whole volume \mathcal{V} , but that absorbed by a small part Ω of this volume (for instance one of the sub-volumes corresponding to a spatial discretization of the receiver when radiative transfer is computed to be coupled with other local heat transfer or chemical processes). In such a case, the only modification to the preceding algorithm is that the absorption rate $[1 - \exp(-k_{a,v}d_j)]$ is replaced by $\sum_{q=1}^{m_j} \exp(-k_{a,v}b_q)[1 - \exp(-k_{a,v}c_q)]$ where b_q is the length of the multireflection optical path until the q th intersection with Ω 's boundary, c_q is the length of the path within Ω at this q th encounter and m_j is the number of times the path crosses Ω before exit through the window at \mathbf{x}_j (see Fig. 3). As soon as Ω is very small compared to the whole volume, most of the optical paths never encounter Ω (exit through the window or surface absorption occur before the first encounter) and most events lead therefore to a strictly null value of the weight. Again, the consequence is a high value of the standard deviation of the estimator. The common way to bypass this convergence difficulty is to make use of a simple integral reformulation in which the optical path is not followed from the entrance through the window until exit or absorption, but is followed backward from the considered sub volume to the entrance through the window.

Using this receiver example, at solar frequencies, it is difficult to convincingly illustrate the control variate technique, but if we consider the question of simulating the same system at infrared frequencies, then a new convergence difficulty occurs that is easily bypassed with the introduction of a control variate. At infrared frequencies we are commonly interested by radiative power balances: the radiative power absorbed by the volume minus the emitted one. If we consider the limit case where the system is isotherm, this balance

```

 $\hat{a}_N = 0;$ 
foreach event  $i$  do
    Uniform sampling of  $\nu$ ;
    Uniform sampling of  $\mathbf{x}_0$ ;
    Setting  $\omega_0$  to the vertical downward
    direction;
    Computation of  $\mathbf{x}_1$ ;
     $j = 0$ ;
    while  $\mathbf{x}_{j+1} \notin \mathcal{G}$  do
        Lambert sampling of  $\omega_{j+1}$ ;
        Computation of  $\mathbf{x}_{j+2}$ ;
         $j = j + 1$ ;
    end
     $w_i = \hat{w}_j$ ;
     $\hat{a}_N = \hat{a}_N + w_i$ ;
end
 $\hat{a}_N = \frac{1}{N} \hat{a}_N$ ;
    
```

$$A = \int_{\nu_{min}}^{\nu_{max}} p_{\nu}(\nu) d\nu \int_{\mathcal{G}} p_{\mathbf{x}_0}(\mathbf{x}_0) d\mathbf{x}_0 \{ H(\mathbf{x}_1 \in \mathcal{G}) \hat{w}_1 + H(\mathbf{x}_1 \in \mathcal{R}) \int_{2\pi} p_{\Omega_1}^{\mathcal{R}}(\omega_1) d\omega_1 \{ H(\mathbf{x}_2 \in \mathcal{G}) \hat{w}_2 + H(\mathbf{x}_2 \in \mathcal{R}) \int_{2\pi} p_{\Omega_2}^{\mathcal{R}}(\omega_2) d\omega_2 \dots \} \}$$

$$\hat{w}_j \equiv \hat{w}_j(\nu, \mathbf{x}_0, \omega_1, \omega_2 \dots \omega_{j-1}) = \varphi_{\nu} S_{\mathcal{G}}(\nu_{max} - \nu_{min}) [1 - \exp(-k_{a,\nu} d_j)]$$

with

$$p_{\nu}(\nu) = \frac{1}{\nu_{max} - \nu_{min}}$$

$$p_{\mathbf{x}_0}(\mathbf{x}_0) = \frac{1}{S_{\mathcal{G}}}$$

$$p_{\Omega_q}^{\mathcal{R}}(\omega_q) = \frac{\omega_q \cdot \mathbf{n}_q}{\pi}$$

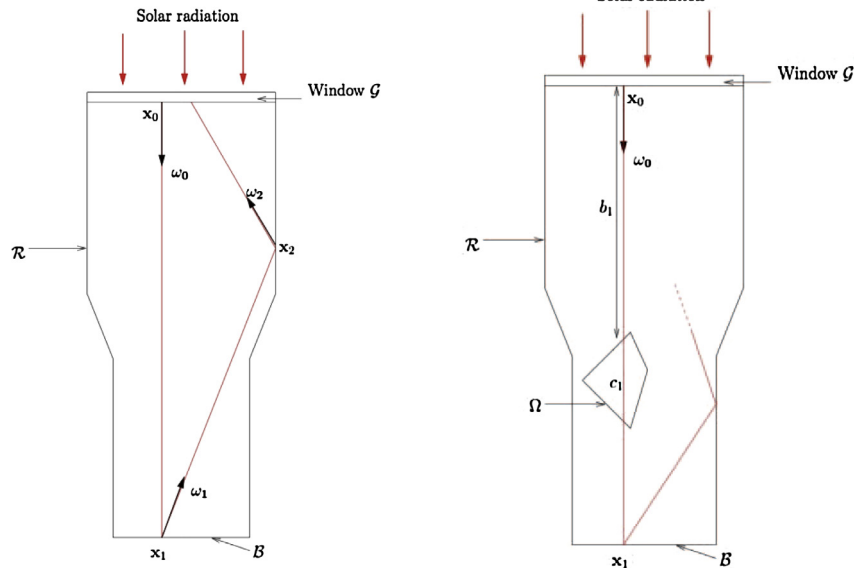


Fig. 3. Simplified version of the fluidized bed receiver example of Section 4. A perfectly collimated uniform beam of monochromatic surfacic power density φ_{ν} is incident in the direction ω_0 , perpendicularly to an horizontal plane glass window \mathcal{G} . The gas–particle mixture filling the volume is purely absorbing (no scattering) with a uniform monochromatic absorption coefficient $k_{a,\nu}$. Refraction by the window is neglected and the surface of the receiver \mathcal{R} is assumed diffuse perfectly reflective. \mathbf{n}_q is the inward surface normal at the location \mathbf{x}_q of the q th intersection with $\mathcal{R} \cup \mathcal{G}$. The reflected direction at \mathbf{x}_q is noted ω_q . $d_j = \sum_{q=0}^{j-1} \|\mathbf{x}_{q+1} - \mathbf{x}_q\|$ is the length of the multireflection optical path $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2 \dots \mathbf{x}_j$ and $[1 - \exp(-k_{a,\nu} d_j)]$ is the absorption rate along this path. When a sub-volume Ω is introduced, b_q is the length of the multireflection optical path until the q th intersection with Ω 's boundary and c_q is the length of the path within Ω at this q th encounter.

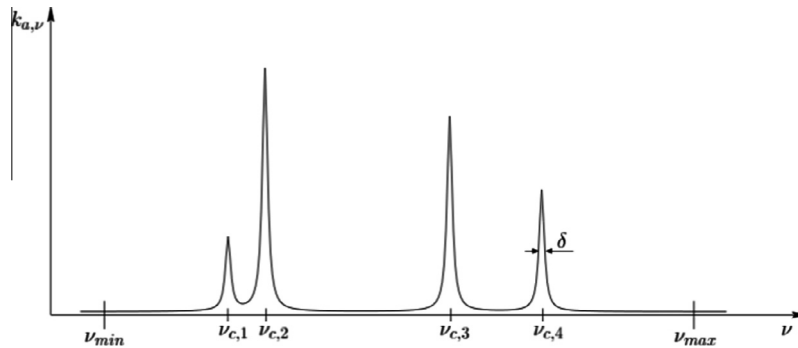


Fig. 4. A gas absorption spectrum example including four Lorentzian lines.

is null. But as most algorithms follow optical paths from an emission location to an absorption location, they commonly evaluate separately the absorption and emission parts and they don't produce a strictly null evaluation of the balance. If the system is not isotherm but is now close to a given temperature T_{ref} , then the radiative power balance corresponds to a small difference between the emission and the absorption contributions. But evaluating accurately a quantity as the difference of two close quantities requires low relative uncertainties, which means that a very large number of events will be needed to ensure that both emission and absorption are evaluated with enough accuracy for their difference to be meaningful. This implies that the relative standard deviation of the balance estimator tends to infinity when the relative maximum temperature difference tends to zero. For concentrated solar application we are seldom dealing with quasi-isotherm systems, but when intense convection is at work the system can still be isotherm enough for such convergence difficulties to be met, as is indeed the case in the bottom part of the fluidized bed receiver example of Section 4. Many advanced solutions are available (Cherkaoui et al., 1996; Howell, 1998) but a very simple one can be rapidly exposed here: we know that the weights corresponding to the isotherm case lead to a null solution at the limit of an infinite number of events, which represents therefore a possible choice for a control variate in the sense of Eq. (9). In practice, using this control variate, the algorithm structure is identical, but each time the Planck function $I_v^{eq}(T)$ is used to represent the thermal emission at a location where the temperature is T , this Planck function value is simply replaced by the difference $I_v^{eq}(T) - I_v^{eq}(T_{ref})$ and the problem of evaluating the difference between two large quantities vanishes as we are now evaluating the deviation to the isotherm case.

2.2. Zero-variance

Let us now use the type of integral formulations illustrated above to comment on the benefits that can be made of the zero-variance concept in application fields such as concentrated solar research. As illustrated in Assaraf and Caffarel (1999), Hoogenboom (2008), zero-variance algorithms are as old as the Monte Carlo method itself. They are algorithms such that for each event the weight value is strictly identical to the addressed quantity itself (with the preceding notations, W is distributed according to a Dirac distribution centered on A). This means that only one single event is required to get a perfect estimator. In the process of designing a zero-variance algorithm for the estimation of A , an a priori perfect knowledge of A is required, which means that such an ideal algorithm can only be practically implemented when it is useless (when A is known). However, even outside pure theoretical considerations, we will argue that working on integral formulations along the line of Assaraf and Caffarel (1999), Hoogenboom (2008), as if all exact solutions were available, is an efficient way to orient, in a quasi-system-

atic manner, the above-listed optimization procedures (mainly importance sampling and integral reformulation). The heart of zero-variance theory, as exposed in the recent literature, is based on adjoint models, but hereafter the objective will be to briefly expose the underlying principles and translate them into application examples using the above presented material. For advanced considerations, all the required concepts are exposed in a clear and detailed manner in Hoogenboom (2008). Note also that there is here a possible confusion with the variance reduction literature: research efforts concerning the zero-variance concept have very little in common with automatic importance sampling techniques, where the sampled weights are statistically analyzed along the simulation process (and a tunable sampling procedure is gradually adjusted on the basis of the gathered statistical information). Here the approach is different: the addressed question is the physical understanding of the information required to ideally optimize a given algorithm (when possible). If this information is easily accessible (under any approximate form) an optimized version of the algorithm can be designed in a straightforward manner (and can even be combined with automatic importance sampling techniques). If it is not easily accessible, then the reasons can be physically analyzed in order to suggest meaningful integral reformulations.

Ideal optimization can be easily illustrated using a one dimension version of Eq. (6):

$$A = E(W) = \int_{\mathcal{D}_X} p_X(x) dx \hat{w}(x) = \int_{x_{min}}^{x_{max}} p_X(x) dx \hat{w}(x) \quad (11)$$

where the definition domain of the random variable X is the segment $\mathcal{D}_X = [x_{min}, x_{max}]$, and where the weight function \hat{w} is assumed strictly positive on this segment. Applying the procedure of Eq. (7) with the ideally optimized probability density function $p'_{X,opt}(x) = \frac{p_X(x) \hat{w}(x)}{A}$ leads to

$$A = E(W_{opt}) = \int_{x_{min}}^{x_{max}} p'_{X,opt}(x) dx \hat{w}_{opt}(x) \quad (12)$$

with $\hat{w}_{opt}(x) = \hat{w} \frac{p_X(x)}{p'_{X,opt}(x)} = A$ for all x in $[x_{min}, x_{max}]$. The random variable $W_{opt} = \hat{w}_{opt}(X)$ is therefore distributed according to a Dirac distribution centered on A and the variance of the estimator \tilde{a}_N is strictly null: the algorithm evaluates A exactly whatever the number of events, even with one single event.

We can translate this into a practical example considering the preceding solar receiver in the particular case where all opaque surfaces are now black and where the geometry is a cylinder of height h . Then all reflections vanish, the weight is independent of the entrance location \mathbf{x}_0 and the integral formulation of Fig. 3 becomes

$$A = \int_{v_{min}}^{v_{max}} p_v(v) dv \hat{w}(v) \quad (13)$$

with $\hat{w}(v) = \varphi_v S_G(v_{max} - v_{min})[1 - \exp(-k_{a,v}h)]$. The frequency v plays here the role of X in Eq. (11) and the ideal optimization is achieved with

$$p_{v,opt}^I(v) = \frac{p_v(v) \hat{w}(v)}{A} = \frac{\varphi_v[1 - \exp(-k_{a,v}h)]}{\int_{v_{min}}^{v_{max}} \varphi_v[1 - \exp(-k_{a,v}h)]dv} \quad (14)$$

This ideally optimized probability density function is therefore a normalized form of the product $p_v(v) \hat{w}(v)$ where the normalization coefficient is A itself. Of course, if a Monte Carlo algorithm is being designed, it is because the φ_v and $k_{a,v}$ spectra are complex enough for A to be difficult to evaluate and therefore $p_{v,opt}^I$ is essentially impractical. Its only interest is in the process of choosing an approximate model leading to a non-ideally but satisfactorily optimized probability density function. For such spectral considerations, a common practice is to consider the optically thin and optically thick limits. At the optically thick limit, the exponentials tend to zero at all frequencies and

$$p_{v,opt}^I(v) \approx \frac{\varphi_v}{\int_{v_{min}}^{v_{max}} \varphi_v dv} \quad (15)$$

At the optically thin limit, the exponentials can be linearized and

$$p_{v,opt}^I(v) \approx \frac{\varphi_v k_{a,v}}{\int_{v_{min}}^{v_{max}} \varphi_v k_{a,v} dv} \quad (16)$$

The code designer then addresses the question of choosing a criterion on h to retain one of the two limit cases, or a linear combination of them (remembering that this only affects the convergence behavior but not the limit value of the estimate at $N \rightarrow +\infty$), and finally designs efficient ways of sampling v according to these probability density functions, for instance using pre-tabulated discrete probability density functions covering all the encountered φ_v and $k_{a,v}$ spectra.

Thinking in terms of zero-variance algorithms gets slightly more complex as soon as each event requires the sampling of more than one random variable. Let us consider formally the question of reducing the variance to zero using importance sampling on the basis of Eq. (6) where each event is defined via the sampling of two random variables \mathbf{X} and \mathbf{Y} . Introducing two ideally optimized probability density functions $p_{\mathbf{X},opt}^I$ and $p_{\mathbf{Y},opt}^I$, we write (see Eqs. (7) and (8))

$$A = \int_{\mathcal{D}_{\mathbf{X}}} p_{\mathbf{X},opt}^I(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_{\mathbf{Y}(\mathbf{x})}} p_{\mathbf{Y},opt}^I(\mathbf{y}|\mathbf{x}) d\mathbf{y} \hat{w}_{opt}^I(\mathbf{x}, \mathbf{y}) \quad (17)$$

with

$$\hat{w}_{opt}^I(\mathbf{x}, \mathbf{y}) = \frac{p_{\mathbf{X}}(\mathbf{x})}{p_{\mathbf{X},opt}^I(\mathbf{x})} \frac{p_{\mathbf{Y}}(\mathbf{y}|\mathbf{x})}{p_{\mathbf{Y},opt}^I(\mathbf{y}|\mathbf{x})} \hat{w}(\mathbf{x}, \mathbf{y}) \quad (18)$$

where $p_{\mathbf{X},opt}^I$ and $p_{\mathbf{Y},opt}^I$ are chosen such that $\hat{w}_{opt}^I(\mathbf{x}, \mathbf{y}) = A$. This imposes in particular that \hat{w}_{opt}^I is independent of \mathbf{y} , which means that the ratio $\frac{p_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) \hat{w}(\mathbf{x}, \mathbf{y})}{p_{\mathbf{Y},opt}^I(\mathbf{y}|\mathbf{x})}$ must be independent of \mathbf{y} whatever \mathbf{x} . This requires that

$$p_{\mathbf{Y},opt}^I(\mathbf{y}|\mathbf{x}) = \frac{p_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) \hat{w}(\mathbf{x}, \mathbf{y})}{\int_{\mathcal{D}_{\mathbf{Y}(\mathbf{x})}} p_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) \hat{w}(\mathbf{x}, \mathbf{y}) d\mathbf{y}} \quad (19)$$

and reporting this expression into Eq. (18) leads to

$$p_{\mathbf{X},opt}^I(\mathbf{x}) = \frac{p_{\mathbf{X}}(\mathbf{x}) A_{\mathbf{X}}(\mathbf{x})}{\int_{\mathcal{D}_{\mathbf{X}}} p_{\mathbf{X}}(\mathbf{x}) A_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}} \quad (20)$$

with $A_{\mathbf{X}}(\mathbf{x}) = \int_{\mathcal{D}_{\mathbf{Y}(\mathbf{x})}} p_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) \hat{w}(\mathbf{x}, \mathbf{y}) d\mathbf{y}$. These few lines demonstrate that the ideally optimized probability density functions are unique and that they can be constructed successively following the same procedure as the one illustrated in the first example with a single integral. For the integral over \mathbf{x} , we introduced $A_{\mathbf{X}}(\mathbf{x})$ that is the contribution to A of all events with the same sampled \mathbf{x} value,

$$A = \int_{\mathcal{D}_{\mathbf{X}}} p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} A_{\mathbf{X}}(\mathbf{x}) \quad (21)$$

and assuming that $A_{\mathbf{X}}(\mathbf{x})$ is known, Eq. (20) simply means that the ideally optimized probability density function $p_{\mathbf{X},opt}^I$ accounts simultaneously for both the probability of \mathbf{x} to occur and the contribution of \mathbf{x} to the final result. For the second integral and for any sampled value of \mathbf{x} , Eq. (19) corresponds to the ideal optimization process applied to

$$A_{\mathbf{X}}(\mathbf{x}) = \int_{\mathcal{D}_{\mathbf{Y}(\mathbf{x})}} p_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) d\mathbf{y} A_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) \quad (22)$$

with $A_{\mathbf{Y}}(\mathbf{y}|\mathbf{x}) = \hat{w}(\mathbf{x}, \mathbf{y})$. The process would be the same if Eq. (17) would include more than two integrals: $A_{\mathbf{Y}}(\mathbf{y}|\mathbf{x})$ would have an integral form and each new integral would lead to the definition of a new ideally optimized probability density function.

As already mentioned, $p_{\mathbf{X},opt}^I$ and $p_{\mathbf{Y},opt}^I$ (and the eventually other successive probability density functions) cannot be used as such and their expressions are only useful under approximate forms corresponding to the introduction of simplified physical models allowing analytic integrations. Let us illustrate this point coming back to the example of Fig. 3 in its initial form, with any geometry and perfectly reflective surfaces. With such an algorithm, convergence difficulties are essentially encountered when complex spectral properties are introduced, such as in the case of gaseous absorption (as already mentioned in Section 2.1 and Fig. 4). If we apply the same methodology as in Eqs. (19) and (20) to the successive integrals of the integral formulation of Fig. 3, we get

$$p_{v,opt}^I(v) = \frac{p_v(v)A_v(v)}{\int_{v_{min}}^{v_{max}} p_v(v)A_v(v)dv} \quad (23)$$

$$p_{\mathbf{x}_0,opt}^I(\mathbf{x}_0|v) = \frac{p_{\mathbf{x}_0}(\mathbf{x}_0)A_{\mathbf{x}_0}(v, \mathbf{x}_0)}{\int_{\mathcal{G}} p_{\mathbf{x}_0}(\mathbf{x}_0)A_{\mathbf{x}_0}(v, \mathbf{x}_0)d\mathbf{x}_0} \quad (24)$$

$$p_{\Omega_1,opt}^{\mathcal{R},I}(\omega_1|v, \mathbf{x}_0) = \frac{p_{\Omega_1}^{\mathcal{R}}(\omega_1)A_{\Omega_1}(v, \mathbf{x}_0, \omega_1)}{\int_{2\pi} p_{\Omega_1}^{\mathcal{R}}(\omega_1)A_{\Omega_1}(v, \mathbf{x}_0, \omega_1)d\omega_1} \quad (25)$$

$$p_{\Omega_2,opt}^{\mathcal{R},I}(\omega_2|v, \mathbf{x}_0, \omega_1) = \frac{p_{\Omega_2}^{\mathcal{R}}(\omega_2)A_{\Omega_2}(v, \mathbf{x}_0, \omega_1, \omega_2)}{\int_{2\pi} p_{\Omega_2}^{\mathcal{R}}(\omega_2)A_{\Omega_2}(v, \mathbf{x}_0, \omega_1, \omega_2)d\omega_2} \quad (26)$$

$$\dots \quad (27)$$

Then the question is: how can we model $A_v, A_{\mathbf{x}_0}, A_{\Omega_1}$, etc. using simple enough physical approximations so that all expressions can be handled analytically and so that efficient sampling procedures are available for each of the resulting optimized probability density functions? This question is seldom trivial, and the common practice is to concentrate on the main source of variance. In the present example, we should then concentrate on the sampling of frequency. But this requires that we are able to derive a simple approximation of A_v , that is to say of the monochromatic total absorption by the gas along all multi-reflection optical paths. This may sound extremely difficult, but we must keep in mind that we are only seeking information in order to efficiently orient the sampling procedure (here to concentrate on the sampling of frequency in the vicinity of line centers when required) and that approximating the ideally optimized probability density function using very simple approximations can reduce the standard deviation of the estimator by orders of magnitude (for instance in the optically thin limit with thin absorption lines). We can therefore use any estimate of the average length $\langle L \rangle$ of multi-reflection optical paths within the receiver (Blanco and Fournier, 2003; Blanco and Fournier, 2006), as function of the receiver's geometry and of the surface emissivities, and write (retaining only the frequency dependence)

$$A_v \approx \varphi_v[1 - \exp(-k_{a,v}\langle L \rangle)] \quad (28)$$

We are then nearly back to the first example of the present subsection and we can make use of the optically thin and optically thick limits as indicated above.

But even lower variances can be reached thinking in terms of zero-variance algorithms if we allow us to consider not only importance sampling but also simple integral reformulations. Indeed, when optimizing the frequency probability density function using $\langle L \rangle$ as indicated above, frequency sampling may remain the main source of variance. The reason is that even if $\langle L \rangle$ is accurately evaluated, this average value can be the result of small paths (exit after one reflection at a surface location close to the entrance) for which the optimized probability density function derived at the optically thin limit could be satisfactory, and long multiple reflection paths for which the optically thick limit would be better adapted. And we cannot distinguish these two types of optical paths because frequency is the first sampled variable, before the optical path sampling.

But here the surfaces are perfectly reflective and the multiple reflection optical path sampling procedure is independent of frequency. We can therefore invert integrals in the integral formulation of Fig. 3 to get

$$A = \int_{\mathcal{G}} p_{\mathbf{x}_0}(\mathbf{x}_0)d\mathbf{x}_0 \left\{ H(\mathbf{x}_1 \in \mathcal{G}) \int_{v_{min}}^{v_{max}} p_{v_1}(v_1)dv_1 \hat{w}_1 + H(\mathbf{x}_1 \in \mathcal{R}) \int_{2\pi} p_{\Omega_1}^{\mathcal{R}}(\omega_1)d\omega_1 \right. \\ \left\{ H(\mathbf{x}_2 \in \mathcal{G}) \int_{v_{min}}^{v_{max}} p_{v_2}(v_2)dv_2 \hat{w}_2 + H(\mathbf{x}_2 \in \mathcal{R}) \int_{2\pi} p_{\Omega_2}^{\mathcal{R}}(\omega_2)d\omega_2 \right. \\ \dots \} \} \quad (29)$$

with $p_{v_j}(v_j) = \frac{1}{v_{max} - v_{min}}$. This corresponds to an algorithm in which a multiple-reflection photon trajectory is sampled first, and frequency is sampled afterward, at a stage where the trajectory length d_j is known. In terms of ideal optimization, this integral inversion induces a significant simplification in the sense that

$$p_{v_j,opt}^I(v_j) = \frac{p_{v_j}(v_j) \hat{w}_j}{\int_{v_{min}}^{v_{max}} p_{v_j}(v_j) \hat{w}_j dv_j} = \frac{\varphi_{v_j}[1 - \exp(-k_{a,v_j}d_j)]}{\int_{v_{min}}^{v_{max}} \varphi_{v_j}[1 - \exp(-k_{a,v_j}d_j)]dv_j} \quad (30)$$

which means that we are now strictly back to the first example of Eq. (14) where the optimization procedures based on the optically thin and thick limits are efficient, using for instance a choice criterion based on the comparison of $k_{a,v_j}d_j$ with unity. Extension to partially reflective non-gray surfaces is quite straightforward: it only requires that a spectral-average bidirectional reflectivity function is used when constructing the optical path at a stage where frequency is not yet sampled, and once frequency is sampled, the weight is pondered by the product of the monochromatic reflectivity to spectral-average reflectivity ratios.

In the preceding derivations, it is important to note that an assumption was implicitly made: \hat{w} was assumed to be strictly positive, otherwise $p_{\mathbf{x}_0,opt}^I$ and $p_{\mathbf{x}_1,opt}^I$ (Eqs. (19) and (20)) could not be interpreted as optimized probability density functions. Of course, if \hat{w} is strictly negative, the difficulty can be easily bypassed using a simple sign change, but the above presented zero-variance methodology does not apply when the weight function changes sign across the integration domain. More subtle approaches can be used in such cases, based on integral reformulations, for instance adding a constant value that shifts the weight function and turns it into a strictly positive or strictly negative function, but nothing fully systematic could be proposed. Section 2.3 will illustrate how sign changes may lead to effective difficulties when evaluating the sensitivities of A to parameters appearing in the sampling probability density functions.

We will end this subsection with multiple scattering, that was devoted a very specific attention in the zero-variance literature and that is directly relevant to the question of simulating some of the most recently designed solar receivers. As will be illustrated in Sections 4.3 and 4.4, solar

receivers involving multiple scattering of radiation by particle suspensions may rapidly be associated with strong convergence difficulties when using standard Monte Carlo algorithms. This question was first encountered in the early stage of nuclear technologies with the attempts to design nuclear protection devices using the overall reflection features of layers composed of highly scattering material (Hammersley and Handscomb, 1964). This required that the very weak amount of radiation transmitted through the layer be predicted with a few percents accuracy, which represented an extremely high accuracy level in terms of percentage of the incident radiation, and such an accuracy could not be reached with the most common Monte Carlo algorithms. This led to an extensive research literature that cannot be fully summarized here but is worth a close attention considering today's convergence open questions. The work of Berger (1955, 1956), closely revisited in Hammersley and Handscomb (1964), is maybe one of the best examples of how freely the integral formulation can be transformed when one is driven by the ideal objective of a zero-variance algorithm. This led to a failure for subtle numerical reasons, but the corresponding attempt remains pedagogically the more convincing illustration of such methodologies. The work of Hoogenboom (2008), although very academic, was much more successful. Some of its implications are far reaching and are briefly illustrated hereafter, thinking of concentrated solar applications. The reader should however keep in mind that the question of optimizing multiple scattering algorithms remains a fully open question as far as realistic devices are concerned.

Let us again consider the fluidized bed receiver example of Section 2.1 with particles that are now both absorbing and scattering. For simplification the absorption coefficient $k_{a,v}$, the scattering coefficient $k_{s,v}$ and the phase function $p_{s,v}(\omega|\omega_{inc})$ (that is to say the probability density function of the outgoing direction ω after scattering for an incoming direction ω_{inc}) are assumed uniform. An example of non-uniform scattering is provided in Sections 2.3 and 4.3. The side surface \mathcal{S} is still diffuse perfectly reflective, but the bottom surface \mathcal{B} is now black and the addressed quantity is now the radiative flux at the bottom. The algorithm of Fig. 3 can be easily modified to address this new quantity including scattering. Once \mathbf{x}_j and ω_j are determined, an optical thickness κ_j is sampled on $]0, +\infty[$ according to Beer extinction law $p_{Beer}(\kappa) = \exp(-\kappa)$ and the next location $\mathbf{x}_{j+1} \equiv \mathbf{x}_{j+1}(\mathbf{x}_j, \omega_j, \kappa_j)$ is at the surface, that is to say $\mathbf{x}_{j+1} = \mathbf{y}_{j+1} \in \mathcal{S} \cup \mathcal{G} \cup \mathcal{B}$, if $\kappa_j > k_{s,v}|\mathbf{y}_{j+1} - \mathbf{x}_j|$ (where \mathbf{y}_{j+1} is the intersection with the total surface of the receiver of the straight ray starting at \mathbf{x}_j in the direction ω_j , see Fig. 5), and is in the volume otherwise, at the scattering location, that is to say $\mathbf{x}_{j+1} = \mathbf{x}_j + \frac{\kappa_j}{k_{s,v}}\omega_j$. When \mathbf{x}_{j+1} is at the surface, ω_{j+1} is sampled according to the reflection law as in Section 2.1, and when \mathbf{x}_{j+1} is in the volume, ω_{j+1} is sampled according to the phase function. The optical path ends when the bottom surface (absorption) or the glass window (exit) are encountered. This means that the corresponding integral formulation includes sequences

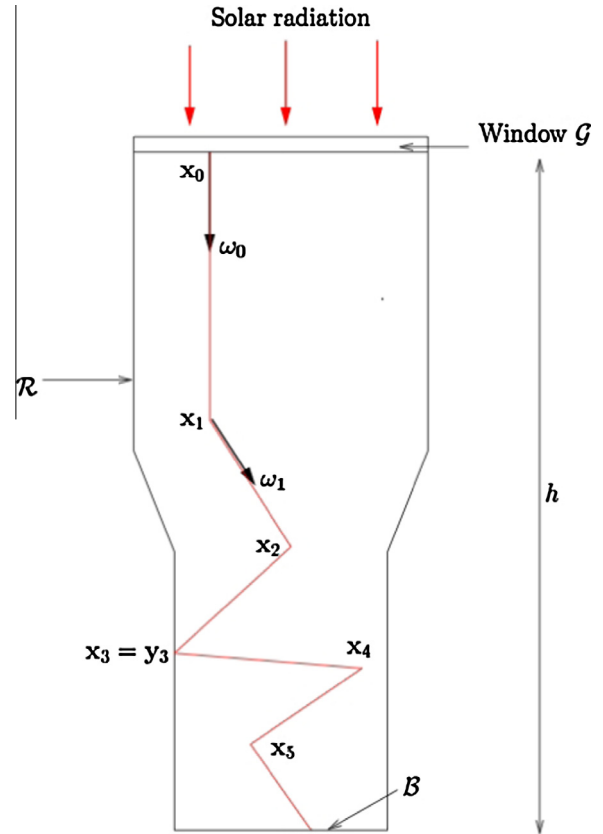


Fig. 5. An example of a multiple scattering, multiple reflection optical path within the receiver, leading to an absorption at the bottom.

such as (see Sections 4.3 and 4.4 for complete formulation examples)

$$\begin{aligned} & \dots \int_0^\infty p_{\kappa_j}(\kappa_j) d\kappa_j \left\{ H(\mathbf{x}_{j+1} \in \mathcal{B} \cup \mathcal{G}) \hat{w}_{j+1} + H(\mathbf{x}_{j+1} \in \mathcal{S}) \right. \\ & \int_{2\pi} p_{\Omega_{j+1}}^{\mathcal{S}}(\omega_{j+1}) d\omega_{j+1} A_{\Omega_{j+1}}^{\mathcal{S}}(\dots, \kappa_j, \omega_{j+1}) + H(\mathbf{x}_{j+1} \in \mathcal{V}) \\ & \left. \int_{4\pi} p_{\Omega_{j+1}}^{\mathcal{V}}(\omega_{j+1}) d\omega_{j+1} A_{\Omega_{j+1}}^{\mathcal{V}}(\dots, \kappa_j, \omega_{j+1}) \right\} \quad (31) \end{aligned}$$

with $p_{\kappa_j}(\kappa_j) = \exp(-\kappa_j)$ and $p_{\Omega_{j+1}}^{\mathcal{V}}(\omega_{j+1}) = p_{s,v}(\omega_{j+1}|\omega_j)$, and where $A_{\Omega_{j+1}}^{\mathcal{V}}(\dots, \kappa_j, \omega_{j+1})$ represents formally all the successive next integrals in the scattering case, and is therefore the contribution to the final result of all optical paths having the same initial history, up to a scattering event at $\mathbf{x}_{j+1} \in \mathcal{V}$ with a scattering direction ω_{j+1} , and differing only afterward¹. When encountering convergence difficulties one will try, for instance, to optimize the sampling of the scattering direction ω_{j+1} in order to get a lower variance of the estimated radiative flux at the bottom of the receiver. Let us first comment on the kind of attempts that could be

¹ $A_{\Omega_{j+1}}^{\mathcal{S}}$ is similarly the contribution to the final result of all optical paths having the same initial history, up to a reflection at $\mathbf{x}_{j+1} \in \mathcal{S}$ with a reflection direction ω_{j+1} . Eq. (31) provides an explicit representation of the four algorithmic branches: \mathbf{x}_{j+1} is the location of either an absorption at the bottom \mathcal{B} , or an exit through the window \mathcal{G} , or a reflection on the side surface \mathcal{S} , or a scattering event within the volume \mathcal{V} .

made without the use of any specific theoretical support. This will help us to illustrate how different (and not immediately intuitive) can be the physical pictures resulting of the zero-variance approach.

The convergence difficulties are associated with the fact that when the volume of the receiver is highly scattering, most sampled optical paths never reach the bottom of the receiver: the probability is high that after a few scattering events photons are back scattered through the window and only the unlikely trajectories penetrating deep into the volume can contribute to the bottom flux. The variance of the bottom flux estimator can therefore be extremely high for high scattering optical thicknesses ($k_{s,v}h \gg 1$ where h is the total height of the receiver, see Fig. 5). Thinking in terms of importance sampling, one of the first attempts could therefore be to favor the scattering directions toward the bottom. This leads indeed to significant variance reductions if the probability density function of ω_{j+1} is subtly adjusted, but this is only true for small single scattering albedo $\frac{k_{s,v}}{k_{a,v}+k_{s,v}}$ (that is to say for relatively small values of the scattering coefficient $k_{s,v}$ compared to the absorption coefficient $k_{a,v}$). For high single scattering albedo such an attempt to favor the directions toward the bottom leads systematically to an increase of the variance. This means that the physical picture that we associated to the convergence difficulties is at least partially incorrect.

Applying the zero-variance methodology to analyze this contradiction, one first defines rigorously the ideally optimized probability density function

$$p_{\Omega_{j+1},opt}^{\nu,I}(\omega_{j+1}) = \frac{p_{\Omega_{j+1}}^{\nu}(\omega_{j+1})A_{\Omega_{j+1}}^{\nu}(\dots, \kappa_j, \omega_{j+1})}{\int_{4\pi} p_{\Omega_{j+1}}^{\nu}(\omega'_{j+1})A_{\Omega_{j+1}}^{\nu}(\dots, \kappa_j, \omega'_{j+1})d\omega'_{j+1}} \quad (32)$$

and then uses approximate models to understand its main features. In particular, does $p_{\Omega_{j+1},opt}^{\nu,I}$ favor or not the sampling of directions toward the bottom? The simplest approximate model of radiation propagation in highly scattering materials is the diffusion approximation. Under this approximation the angular distribution of photons at each location is quasi-isotropic and all radiative quantities can be evaluated using only the spatial distribution of the photon density: there is no need to distinguish the photons by their propagation direction. As far as our problem example is concerned, this implies that $A_{\Omega_{j+1}}^{\nu}(\dots, \kappa_j, \omega_{j+1})$ is quasi-independent of ω_{j+1} : photons propagating toward the bottom do not contribute significantly more to the flux at the bottom of the receiver than those propagating in any other direction. The ideally optimized probability density function is therefore approximately

$$\begin{aligned} p_{\Omega_{j+1},opt}^{\nu,I}(\omega_{j+1}) &\approx \frac{p_{\Omega_{j+1}}^{\nu}(\omega_{j+1})A_{\Omega_{j+1}}^{\nu}(\dots, \kappa_j, \omega_{j+1})}{A_{\Omega_{j+1}}^{\nu}(\dots, \kappa_j, \omega_{j+1}) \int_{4\pi} p_{\Omega_{j+1}}^{\nu}(\omega'_{j+1})d\omega'_{j+1}} \\ &= p_{\Omega_{j+1}}^{\nu}(\omega_{j+1}) = p_{s,v}(\omega_{j+1}|\omega_j) \end{aligned} \quad (33)$$

According to the diffusion approximation, sampling directions according to the phase function (as in any standard algorithm) is therefore very close to the ideal sampling, which provides a simple explanation of why all attempts to favor scattering directions toward the bottom lead to worse convergence properties. Such attempts are in fact only meaningful at locations where the diffusion approximation is not valid, that is to say close to the boundaries (as far as we are concerned, close to the light entrance window and close to the bottom of the receiver). Ideal optimization therefore implies that for the first scattering events, immediately after the photon entrance through the window, angular sampling should be biased toward the bottom, then the phase function should be used, without any bias, for all other scattering events, until the immediate vicinity of the bottom is reached where again the directions toward the bottom should be favored. Obviously, such a procedure is very difficult to implement as each scattering event must be treated differently, depending on its location, on the basis of any fast evaluation of $A_{\Omega_{j+1}}$ and its dependence on ω_{j+1} . This could be achieved successfully for pure one dimension scattering, but dealing with realistic configurations such as those of Sections 4.3 and 4.4 remains an open question: convergence difficulties are so commonly encountered, in so many application fields, when dealing with highly scattering media with the Monte Carlo method, that the above summarized ideas, initiated in Hoogenboom (2008), are undeniably worth a very close further attention.

2.3. Sensitivity analysis

The benefits of integral formulation efforts extend beyond the question of bypassing convergence difficulties: they are very significant in the process of analyzing (physically interpreting) simulation results. Among such analysis, the one that was devoted the deeper theoretical attention is sensitivity analysis. It was shown, using a general and systematic approach, that when a Monte Carlo algorithm is used for the estimation of any physical quantity A , a simple and fast additional procedure can be implemented that simultaneously estimates the sensitivity of A to any parameter. This practically means that when a Monte Carlo code is available that computes A , very few additional code lines are required to transform it so that not only A is computed but also the partial derivatives of A with respect to all the parameters of interest, this interest being either in terms of physical analysis (how does A evolve when modifying the parameter?), or in terms of optimum design (what is the optimum value of the parameter considering a target value for A ?). Such sensitivities of A to a parameter π are noted $\partial_{\pi}A$ hereafter.

Prior to any formal consideration, we want to emphasize that the keywords “Monte Carlo” and “Sensitivity Analysis” are often associated in the literature for reasons that have nothing to do with the simultaneous evaluation of A and $\partial_{\pi}A$. The commonly accepted meaning of “Sensitivity Analysis” is indeed much broader than analyzing the behavior of a model

by measuring local derivatives with respect to successive parameter (Saltelli et al., 2004; Cacuci, 2003). It refers to the analysis of the ways A depends on the variations of all types of model parameters, boundary conditions and initial conditions (for unstationary studies). This includes non-linear responses to finite-size perturbations and combined perturbations of multiple inputs viewed as random variables (for instance because of measurements uncertainties in data assimilation contexts). Evaluation of local derivatives $\partial_\pi A$ is therefore only a restricted class of sensitivity-analysis tools. However, to the best of our Knowledge, it is the only class for which the use of advanced integral formulation techniques was reported. Considering the scope of the present article, we concentrate exclusively on how $\partial_\pi A$ can be evaluated together with A . The main point is that not only the code is little modified, but the computation time is also little increased because the evaluation of $\partial_\pi A$ uses the same sampling events as those initially required for the evaluation of A . However two distinct difficulties may be encountered when implementing sensitivity computations: first, even if the resulting implementation is easy the required integral formulation can be very tedious; second, nothing ensures that if the sampling distributions have been carefully adjusted so that A is accurately estimated, then $\partial_\pi A$ is accurately estimated using the same number of sampled events (the zero variance algorithms for A and $\partial_\pi A$ may have nothing in common).

In practice, prior to any development, three questions must be examined, each positive answer adding one complexity level in terms of both formal derivation and statistical convergence:

1. Do the sampling probability density functions depend on π ?
2. Does the integration domain depend on π ?
3. Does π appear in any algorithmic test?

Case a: Let us first consider a simple case in which all answers are negative. If p_X, p_Y, \mathcal{D}_X and \mathcal{D}_Y in Eq. (6) are independent of π , and if $\hat{w}(\mathbf{x}, \mathbf{y}) \equiv \hat{w}(\mathbf{x}, \mathbf{y}; \pi)$ is a continuous function of π (no algorithmic test involving π), then $\partial_\pi A$ can be written as

$$\begin{aligned} \partial_\pi A &= E(W_\pi) \\ &= \int_{\mathcal{D}_X} p_X(\mathbf{x}) d\mathbf{x} \int_{\mathcal{D}_Y(\mathbf{x})} p_Y(\mathbf{y}|\mathbf{x}) d\mathbf{y} \hat{w}_\pi(\mathbf{x}, \mathbf{y}; \pi) \end{aligned} \quad (34)$$

with $\hat{w}_\pi = \partial_\pi \hat{w}$. The only difference between the integral formulations of Eqs. (6) and (34) is that the weight function \hat{w} is replaced by its partial derivative. This simply means that A and $\partial_\pi A$ can be evaluated using the very same sampling algorithm: for each $i \in \{1, N\}$, \mathbf{x}_i is sampled according to p_X , \mathbf{y}_i is sampled according to p_Y under the condition $\mathbf{x} = \mathbf{x}_i$, w_i is computed as $\hat{w}(\mathbf{x}_i, \mathbf{y}_i)$, $w_{\pi,i}$ is computed as $\partial_\pi \hat{w}(\mathbf{x}_i, \mathbf{y}_i)$, A is evaluated using the estimate $\tilde{a}_N = \frac{1}{N} \sum_{i=1}^N w_i$ and $\partial_\pi A$ using the estimate $\tilde{a}_{\pi,N} = \frac{1}{N} \sum_{i=1}^N w_{\pi,i}$.

We will see that in all cases (whatever the answers to the above three questions), the same algorithmic structure can be kept, modifying only the way $w_{\pi,i}$ is computed. But sticking to the present case, a first possible convergence difficulty can be pointed out: as suggested in de Lataillade et al. (2002), let us consider an example where \hat{w} has the functional form $\hat{w}(\mathbf{x}, \mathbf{y}; \pi) = f(\mathbf{x}, \mathbf{y}) + \pi g(\mathbf{x}, \mathbf{y})$ in the particular case of $\pi = 0$. This leads to $\hat{w}(\mathbf{x}, \mathbf{y}; 0) = f(\mathbf{x}, \mathbf{y})$ and $\hat{w}_\pi(\mathbf{x}, \mathbf{y}; 0) = g(\mathbf{x}, \mathbf{y})$. In such a case, the weight functions corresponding to A and $\partial_\pi A$ can obviously be so distinct (we imposed no constraint relating f – g except that they are both independent of π) that if the sampling algorithm was optimized so that it evaluates A accurately, then the variance of the sensitivity estimator is impractically large. Hopefully such extreme cases have only been identified theoretically and our practice of implementing sensitivity computations has always been successful when the integration domain, the sampling probability density functions and the algorithmic tests did not involve the considered parameter.

Case b: Let us now assume that the above first question is answered positively and the two others negatively. The sampling probability density functions p_X and p_Y depend on π and Eq. (6) leads again to Eq. (34) with

$$\hat{w}_\pi = \partial_\pi \hat{w} + \hat{w} \left[\frac{\partial_\pi p_X}{p_X} + \frac{\partial_\pi p_Y}{p_Y} \right] \quad (35)$$

The algorithmic conclusions are therefore identical to those of case a, but the convergence question is complexified as soon as the second and third terms of the right side of Eq. (35) are not negligible compared to the first one. The main reason is that the partial derivatives of the probability density functions change sign across their definition domain. Indeed the normalization constraint $\int_{\mathcal{D}_X} p_X(\mathbf{x}) d\mathbf{x} = 1$ imposes that $\int_{\mathcal{D}_X} \partial_\pi p_X(\mathbf{x}) d\mathbf{x} = 0$ which tells us that $\partial_\pi p_X$ takes positive and negative values across \mathcal{D}_X that compensate each other. As mentioned in Section 2.2, weight function sign changes can be sources of convergence difficulties (no zero-variance algorithm can be easily designed) and it can be shown, in the present example, that such difficulties can for instance be practically encountered as soon as the variations with \mathbf{x} of \hat{w} and $\frac{\partial_\pi p_X}{p_X}$ are weakly correlated (the integral contribution of $\frac{\partial_\pi p_X}{p_X} \hat{w}$ will tend to zero but its contribution to the variance of the estimator of $\partial_\pi A$ can be large).

Case c: A positive answer to the second question leads to *domain deformation sensitivities* as defined in Roger et al. (2005). We will assume that the first question is still answered positively and, for simplification, that \mathcal{D}_X and \mathcal{D}_Y are finite segments of one dimension domains. Let us write $\mathcal{D}_X = [a_1(\pi), b_1(\pi)]$ and $\mathcal{D}_Y = [a_2(x; \pi), b_2(x; \pi)]$. It was shown in Roger et al. (2005) that a domain deformation velocity field $\mathbf{V}_\pi(x, y; \pi)$ could be introduced, leading to a sensitivity integral formulation identical to that of Eq. (34) with

$$\begin{aligned} \hat{w}_\pi = \partial_\pi \hat{w} + \hat{w} \left[\frac{\partial_\pi p_X}{p_X} + \frac{\partial_\pi p_Y}{p_Y} \right] \\ + \frac{\partial_x(p_X p_Y \hat{w} V_{\pi,1}) + \partial_y(p_X p_Y \hat{w} V_{\pi,2})}{p_X p_Y} \end{aligned} \quad (36)$$

where $V_{\pi,1}$ and $V_{\pi,2}$ are respectively the x -component and the y -component of \mathbf{V}_π . A wide degree of freedom is associated to the choice of the \mathbf{V}_π field, in particular as far as convergence enhancement is concerned. But this point was very little explored and the authors suggested the use of one particular field constructed via a systematic procedure. In the present case, this procedure leads to

$$V_{\pi,1}(x; \pi) = \partial_\pi a_1(\pi) + \frac{\partial_\pi b_1(\pi) - \partial_x a_1(\pi)}{b_1(\pi) - a_1(\pi)} (x - a_1(\pi)) \quad (37)$$

$$\begin{aligned} V_{\pi,2}(x, y; \pi) = [\partial_\pi a_2(x; \pi) + V_{\pi,1}(x; \pi) \partial_x a_2(x; \pi)] \\ + \frac{[\partial_\pi b_2(x; \pi) + V_{\pi,1}(x; \pi) \partial_x b_2(x; \pi)] - [\partial_\pi a_2(x; \pi) + V_{\pi,1}(x; \pi) \partial_x a_2(x; \pi)]}{b_2(x; \pi) - a_2(x; \pi)} (y - a_2(x; \pi)) \end{aligned} \quad (38)$$

and \hat{w}_π from Eq. (36) becomes

$$\begin{aligned} \hat{w}_\pi = \partial_\pi \hat{w} + \hat{w} \left[\frac{\partial_\pi p_X}{p_X} + \frac{\partial_\pi p_Y}{p_Y} + \left(\frac{\partial_x p_X}{p_X} + \frac{\partial_x p_Y}{p_Y} + \partial_x \hat{w} \right) V_{\pi,1} \right. \\ \left. + \left(\frac{\partial_y p_Y}{p_Y} + \partial_y \hat{w} \right) V_{\pi,2} + \partial_x V_{\pi,1} + \partial_y V_{\pi,2} \right] \end{aligned} \quad (39)$$

Note that the partial derivatives of p_X and p_Y with respect to x and y do not induce a systematic source of convergence difficulty (unlike what we observed for the partial derivatives with respect to π in case b): they are indeed not submitted, in the general case, to any integral constraint. However, it can be easily imagined that formal difficulties may rapidly appear when extending Eqs. (37)–(39) to integration domains of higher dimensions, or of infinite dimension as in the case of multiple reflection/scattering radiative transfer.

Case d: Let us now assume that the second question is answered negatively and the third positively. The preceding derivations do not hold as \hat{w} is non derivable with respect to π . But we will see that this difficulty can be easily bypassed, leading to the same conclusions as those of case c: no specific convergence difficulty, but a potentially very tedious formal derivation of the sensitivity weight. Let us take the same example as in case c assuming now that the computation of the weight involves a test on the variable y : if $y < y^*(x; \pi)$ it is computed using a function \hat{w}_π^{inf} , otherwise using another function \hat{w}_π^{sup} , both functions being derivable with respect to π . Using Heaviside functions, this translates into

$$\begin{aligned} \hat{w}(x, y; \pi) = H(y < y^*(x; \pi)) \hat{w}_\pi^{inf}(x, y; \pi) + H(y \\ \geq y^*(x; \pi)) \hat{w}_\pi^{sup}(x, y; \pi) \end{aligned} \quad (40)$$

Under such conditions, a systematic practical approach can be to temporarily transform the integral formulation to make it compatible with the preceding case, derive the sensitivity weight in this frame, and finally transform back the integral formulation of the sensitivity so that it is strictly compatible with the initial algorithm. In our example, the integral over \mathcal{D}_Y of the Heaviside functions is first transformed into the sum of two integrals and Eq. (6) becomes:

$$\begin{aligned} A = E(W^{inf}) + E(W^{sup}) \\ = \int_{a_1(\pi)}^{b_1(\pi)} p_X(x; \pi) dx \int_{a_2(x; \pi)}^{y^*(x; \pi)} p_Y(y|x; \pi) dy \hat{w}_\pi^{inf}(x, y; \pi) \\ + \int_{a_1(\pi)}^{b_1(\pi)} p_X(x; \pi) dx \\ \times \int_{y^*(x; \pi)}^{b_2(x; \pi)} p_Y(y|x; \pi) dy \hat{w}_\pi^{sup}(x, y; \pi) \end{aligned} \quad (41)$$

The sensitivity can then be seen as the sum of two sensitivities corresponding to case c and

$$\begin{aligned} \partial_\pi A = E(W_\pi^{inf}) + E(W_\pi^{sup}) \\ = \int_{a_1(\pi)}^{b_1(\pi)} p_X(x; \pi) dx \\ \times \int_{a_2(x; \pi)}^{y^*(x; \pi)} p_Y(y|x; \pi) dy \hat{w}_\pi^{inf}(x, y; \pi) \\ + \int_{a_1(\pi)}^{b_1(\pi)} p_X(x; \pi) dx \\ \times \int_{y^*(x; \pi)}^{b_2(x; \pi)} p_Y(y|x; \pi) dy \hat{w}_\pi^{sup}(x, y; \pi) \end{aligned} \quad (42)$$

with

$$\begin{aligned} \hat{w}_\pi^{inf} = \partial_\pi \hat{w}_\pi^{inf} + \hat{w}_\pi^{inf} \left[\frac{\partial_\pi p_X}{p_X} + \frac{\partial_\pi p_Y}{p_Y} + \left(\frac{\partial_x p_X}{p_X} + \frac{\partial_x p_Y}{p_Y} + \partial_x \hat{w}_\pi^{inf} \right) V_{\pi,1} \right. \\ \left. + \left(\frac{\partial_y p_Y}{p_Y} + \partial_y \hat{w}_\pi^{inf} \right) V_{\pi,2} + \partial_x V_{\pi,1} + \partial_y V_{\pi,2} \right] \end{aligned} \quad (43)$$

and

$$\hat{w}_{\pi}^{sup} = \partial_{\pi} \hat{w}_{\pi}^{sup} + \hat{w}_{\pi}^{sup} \left[\frac{\partial_{\pi} p_X}{p_X} + \frac{\partial_{\pi} p_Y}{p_Y} + \left(\frac{\partial_x p_X}{p_X} + \frac{\partial_x p_Y}{p_Y} + \partial_x \hat{w} \right) V_{\pi,1} + \left(\frac{\partial_y p_Y}{p_Y} + \partial_y \hat{w} \right) V_{\pi,2}^{sup} + \partial_x V_{\pi,1} + \partial_y V_{\pi,2} \right] \quad (44)$$

where $V_{\pi,1}$ is defined as in case c (see Eq. (37)) and where $V_{\pi,2}^{inf}$ and $V_{\pi,2}^{sup}$ are defined as in Eq. (38), replacing b_2 with y^* for $V_{\pi,2}^{inf}$ and replacing a_2 with y^* for $V_{\pi,2}^{sup}$.

$$V_{\pi,2}^{inf}(x, y; \pi) = [\partial_{\pi} a_2(x; \pi) + V_{\pi,1}(x; \pi) \partial_x a_2(x; \pi)] + \frac{[\partial_{\pi} y^*(x; \pi) + V_{\pi,1}(x; \pi) \partial_x y^*(x; \pi)] - [\partial_{\pi} a_2(x; \pi) + V_{\pi,1}(x; \pi) \partial_x a_2(x; \pi)]}{y^*(x; \pi) - a_2(x; \pi)} (y - a_2(x; \pi)) \quad (45)$$

$$V_{\pi,2}^{sup}(x, y; \pi) = [\partial_{\pi} y^*(x; \pi) + V_{\pi,1}(x; \pi) \partial_x y^*(x; \pi)] + \frac{[\partial_{\pi} b_2(x; \pi) + V_{\pi,1}(x; \pi) \partial_x b_2(x; \pi)] - [\partial_{\pi} y^*(x; \pi) + V_{\pi,1}(x; \pi) \partial_x y^*(x; \pi)]}{b_2(x; \pi) - y^*(x; \pi)} (y - y^*(x; \pi)) \quad (46)$$

Finally Eq. (42) is transformed back to a form identical to that of Eq. (6):

$$\begin{aligned} \partial_{\pi} A &= E(W_{\pi}) \\ &= \int_{a_1(\pi)}^{b_1(\pi)} p_X(x; \pi) dx \\ &\quad \times \int_{a_2(x; \pi)}^{b_2(x; \pi)} p_Y(y|x; \pi) dy \hat{w}_{\pi}(x, y; \pi) \end{aligned} \quad (47)$$

with

$$\begin{aligned} \hat{w}_{\pi}(x, y; \pi) &= H(y < y^*(x; \pi)) \hat{w}_{\pi}^{inf}(x, y; \pi) + H(y \\ &\geq y^*(x; \pi)) \hat{w}_{\pi}^{sup}(x, y; \pi) \end{aligned} \quad (48)$$

which means that the same sampling events can be used and $\partial_{\pi} A$ can again be evaluated simultaneously with A : the branching between \hat{w}_{π}^{inf} and \hat{w}_{π}^{sup} is simply translated into one between \hat{w}_{π}^{inf} and \hat{w}_{π}^{sup} .

Multiple scattering, multiple reflection: Practical non-academic examples of sensitivity computations for analysis of concentrated solar devices are detailed in Section 4. But we will first discuss a multiple scattering/reflection example in order to point out two final remaining techniques: the handling of integration domains of infinite dimension, and the use of variable changes (when possible) to transform a domain deformation sensitivity problem into a simpler one in which π appears neither in the integration domain, nor in any test. Let us come back to the fluidized bed receiver example of Fig. 5 in Section 2.2, assuming now that scattering is non-uniform and considering the question of evaluating the sensitivity of the absorbed radiative power to a parameter π of the scattering coefficient field (for instance any parameter of the particle density field). The scattering coefficients $k_{s,v}$ depends therefore on location, frequency and π and we write $k_{s,v} \equiv k_{s,v}(\mathbf{x}; \pi)$. Eq.

(31) is unchanged, but the test determining whether the next scattering location belongs to the volume (or no scattering event occurs before surface encounter) becomes

$$\begin{aligned} H(\mathbf{x}_{j+1} \in \mathcal{V}) \\ = H\left(\kappa_j < \int_0^{\|\mathbf{y}_{j+1} - \mathbf{x}_j\|} k_{s,v}(\mathbf{x}_j + s\boldsymbol{\omega}_j; \pi) ds\right) \end{aligned} \quad (49)$$

which means that the addressed sensitivity is a domain

deformation sensitivity (π appears within the test). But it can be quite easily observed that the sampling of an optical thickness κ_j according to Beer extinction law is identical to the sampling of a scattering free path λ_j according to

$$\begin{aligned} p_{\lambda_j}(\lambda_j; \pi) &= k_{s,v}(\mathbf{x}_j + \lambda_j \boldsymbol{\omega}_j; \pi) \\ &\quad \times \exp\left(-\int_0^{\lambda_j} k_{s,v}(\mathbf{x}_j + s\boldsymbol{\omega}_j; \pi) ds\right) \end{aligned} \quad (50)$$

and that in terms of scattering free paths the test becomes

$$H(\mathbf{x}_{j+1} \in \mathcal{V}) = H(\lambda_j < \|\mathbf{y}_{j+1} - \mathbf{x}_j\|) \quad (51)$$

Changing variable from κ_j to λ_j changes therefore nothing to the algorithm, but the parameter does not appear within the test anymore and the sensitivity problem can be considered as belonging to case b. The sensitivity weight can therefore be computed in a straightforward manner, only noticing that the successive scattering and reflection events (potentially up to infinity) are translated into sums of logarithmic derivatives:

$$\hat{w}_{\pi, j+1} = \hat{w}_{j+1} \sum_{q=0}^j \frac{\partial_{\pi} p_{\lambda_q}}{p_{\lambda_q}} \quad (52)$$

We will make use of such sensitivity weight expressions in the infinite dimension case for the multiple scattering/reflection applicative examples of Sections 4.3 and 4.4. This will illustrate that most sensitivity evaluation requirements lead to quite simple formal derivations, even in complex non academic cases, and are very much accessible to the non-specialist (practical implementation will be discussed in Sections 3.2 and 3.3). However, one must keep in mind that, as it was already pointed out, nothing ensures that when a quantity is accurately evaluated its sensitivities are evaluated with the same relative accuracy using the same sampled events. When convergence difficulties are

encountered in terms of sensitivity evaluation, the only way toward convergence enhancement is the use of control variate. Indeed the other convergence enhancement techniques listed in Section 2.1 lead to a modification of the sampling procedure which means that:

- either one keeps the initial idea of using the same sample events to evaluate both the addressed quantity and its sensitivities, and then a better convergence for a given sensitivity is very much likely to imply a worse convergence for the main quantity
- or additional samples are specifically drawn to meet the needs of a given sensitivity, but this obviously increases the computation costs.

Up to now, the second choice was never explored for practical applications, but it may be meaningful in numerous cases where evaluating sensitivities is essential, and where the attempts to make such an evaluation using the difference of two successive Monte Carlo computations (perturbing the parameter) lead to highly inaccurate predictions because of the statistical nature of Monte Carlo algorithms (see Table 1 in [Roger et al. \(2005\)](#)).

3. Practical implementation

We advocate in this section that the practical use of such Monte Carlo techniques as those listed in Section 2 is not restricted to radiative transfer specialists. We illustrate how researchers with a minimum background in radiation physics can rapidly learn how to design codes for their specific applications, including the most recent optical data and algorithmic tools. This illustration is made using a development environment designed by the authors as the result of a long term methodological research project. We believe that this environment meets most of the needs of today's concentrated solar research (as far as radiative transfer simulation is concerned), but our point is essentially that many available programming libraries and data banks are now mature enough to allow non-specialists to design radiative transfer codes *themselves* when their research needs are not yet covered by the existing solar system simulation codes. Any other development environment than the one described here may play a similar role and we hope that the reader clearly understands the present section as one reported experience of what can be performed gathering the tools provided by the radiation physics, numerical mathematics and computer sciences communities.

3.1. EDStar

The development environment EDStar ([EDStar, xxxx](#)) is maintained by a group of physicists specialized in radiative transfer and out-of-equilibrium statistical thermodynamics ([STARWest, xxxx](#)). Its purpose is essentially to facilitate the task of analyzing complex systems using the family of statistical corpuscular transport models intro-

duced by Boltzmann in 1872 ([Boltzmann, 1872](#)). EDSTAR was set up and is maintained by the present authors. Its essential part is devoted to photon transport (within the validity range of the radiative transfer equation) but we are also involved in fields such as micro scale gaseous thermal flows and liquid–gas transitions, or self-organization in biology. Concerning photon transport, our application fields outside solar energy are mainly atmospheric sciences, astrophysics and combustion sciences. For each application EDStar aims at:

- gathering available collision models and data banks, essentially cross sections and transition statistics (as far as radiation is concerned this translates into absorption coefficients, scattering coefficients and single scattering phase functions);
- providing a programming environment allowing to easily design numerical codes addressing the solution of Boltzmann-like kinetic equations (in particular of the radiative transfer equation).

We also try to gather available reference solutions of transport equations (analytic solutions and benchmark solutions) in order to help the code designers during the validation phases. Our website is still under progress but at least all the elements required to produce codes similar to the code examples discussed in the present article are already accessible. The code examples themselves are presented under a fully commented and open format and are meant to serve as useful start basis for developers.

The parts of EDStar that are directly related to concentrated solar research are essentially those concerning

- particle absorption and scattering properties;
- gaseous absorption line properties;
- libraries for GPU-parallel Monte Carlo programming;
- libraries for modeling of complex geometries and fast photon tracking in these geometries.

We concentrate hereafter on the programming libraries that we designed more specifically to simplify the practical implementation of advanced Monte Carlo algorithms for photon transport in complex geometries.

3.2. Mcm

The mcm C++ object library addresses the question of programming Monte Carlo algorithms, including sensitivity estimations, for parallel computing hardware. It hides all the technical aspects of this question (independence of parallel sampled random numbers, memory allocation, input/outputs, further sampling procedures) behind an interface that remains independent of any specific application. The mcm3D development environment described in the next paragraph will be dedicated to photon transport, but as far as mcm is concerned, all kinds of Monte Carlo algorithms can be considered.

Monte Carlo algorithms evaluate statistically one or several quantities $I_1, I_2 \dots I_q$ together with their sensitivities to a given set of parameters $\pi_1, \pi_2 \dots \pi_p$. As we have seen in Section 2, whatever the algorithmic complexity each addressed quantity A (that is to say each A_j and each sensitivity $\frac{\partial A_j}{\partial \pi_k}$ to π_k) is viewed as the expectation of a random variable W and is evaluated as \tilde{a}_N (see Eq. (1)) using a finite number N of independent samples w_k of W . The computation of \tilde{a}_N and its statistical uncertainty $\tilde{\sigma}_N$ in Eqs. (1) and (2) only require the weight and square weight averages, which means that a common feature of Monte Carlo algorithms is the random sampling of \hat{w}_i and the incrementation of $\sum_{i=1}^N \hat{w}_i$ and $\sum_{i=1}^N \hat{w}_i^2$ for each of the q addressed quantities and their p sensitivities. This algorithmic structure can be easily parallelized provided that a procedure is set up that ensures that random samples performed by distinct processes are truly independent. If this is ensured, then the computation task can be splitted into any number of processes, each of them taking in charge the generation of a sub-part of the required \hat{w}_i samples. The total sample sum and sample square sum are then obtained by simply adding the final results of all processes. The efficiency of such a parallelization methodology is high as the communication needs are restricted to the final task of adding the process results: no intermediate communication is required. This statement is not completely true if a strict independence of the random samples is required. Some communication is needed to ensure this independence, however the corresponding increase of the total computation time remains very short.

The following practical choices are made in the Mcm object library:

- The GNU Scientific Library (GSL) (GNU Scientific Library, xxxx) is used for uniform pseudorandom number sampling in the unit interval. Many distinct pseudorandom number generators are implemented in the GSL, which allows to easily check that a given scientific result is indeed the same whatever the choice of one pseudorandom number generator among those insuring the same level of uniformity and independence.² In the present text, all simulations were performed with the RANLXD2 generator, i.e. a double precision implementation of the RANLUX generator (period of about 10^{171}) (James, 1994).
- The Message Passing Interface (MPI) (The Message Passing Interface, xxxx) is used for communication between processes.

- A simple master–slave parallel algorithm is used in which one process is dedicated to the distribution of the tasks of generating W samples and incrementing the sample sum and the sample square sum.
- One process is also dedicated to the control of random number independence. The principle is that a very large number of samples are generated prior to the Monte Carlo run. The corresponding series is splitted into N_{seq} successive sequences of N_{rnd} samples and the state of the generator at the beginning of each sequence is stored. Each process involved in the generation of the W_A samples starts with one of the generator states and if it requires more than N_{rnd} random number generations, a communication is established with the random number independence controller that sends a new generator state and keeps track of all the already used sequences.

We do not detail here the further technical choices concerning how memory is allocated as function of the number of evaluated quantities and the number of considered sensitivities, how the random number independence controller deals with processes asking for more sequences than available, and how input and output files are formatted in a way that allows the code to be restarted by simply generating further independent W samples when the initial evaluation does not meet the accuracy requirements. Further information and practical examples are available on the EDStar website (EDStar, xxxx).

3.3. Mcm3D

The mcm3D development environment is meant to combine the above described mcm library with a geometrical object library developed by the computer graphics research community. Photo-realistic rendering requires the modelling of geometrically highly complex scenes, including the accurate representation of the optical properties of interfaces (in particular of opaque surfaces), and efficient tracking procedures to follow photons from one interface encounter to the next. The tools developed by computer graphics researchers to achieve these tasks are now very mature, with a high abstraction level that makes them immediately applicable to radiative transfer research. Photo-realism sometimes requires also the representation of semi-transparent media, with volume absorption, volume emission and multiple scattering, but we evaluated that the corresponding tools do not yet meet the accuracy requirements of radiation physicists. We therefore focus here on the question of tracking photons in non-participating media and we give illustrations in Section 4 of how volume absorption, emission and scattering algorithms can be easily implemented through simple additions to the non-participating media algorithms.

Among the tools developed by the computer graphics research community, we chose to retain the C++ object library designed in the frame of the Physically Based Rendering Techniques (pbrt) project. Pbrt is an up-to-date

² If such a test is negative, this means that the algorithm requires a higher level generator. This is commonly related to a high variance of W , implying that very small sub-domains play essential roles and must be accurately covered statistically. This may rapidly lead to the conclusion that, whatever the available computation power, the considered algorithm is unpractical prior to the application of some kind of importance sampling or integral reformulation procedure.

physically based renderer that is identified as well written, well commented and striking a good balance between performance and extensibility. But its major quality, in our context, is that it is accompanied by an excellent textbook (Pharr and Humphreys, 2004), ideally adapted to physicists in the sense that the leading choice is to establish each rendering technique on a physically grounded basis, using a style and a vocabulary both very close to those of the radiation physics literature. Choosing *pbirt*, we therefore benefit not only of a set of available computation tools, but also of all the means to fully understand them and learn how to design extensions meeting the needs of concentrated solar research. Among the tasks achieved by the *pbirt* objects, those related to radiative transfer are essentially:

- Geometric representation of surfaces, rays definition (as half-lines specified by their origin and direction, continuous refraction effects being excluded), and computation of the intersection of a ray and a single surface.
- Representation of material optical properties via the definition of (measured or phenomenologically modeled) bidirectional reflectance distribution functions at the interface between two different materials: absorption and reflection at an interface between a semi-transparent material and an opaque material; reflection and transmission at an interface between two semi-transparent materials.
- Definition of primitives that combine a geometrically defined surface and material optical properties at each of its points. The potentially large number of such primitives that are required to model a complex scene are then aggregated into an acceleration structure that allows a fast computation of the first primitive encountered by a ray starting in any direction from any location in the scene.

Pbirt objects also allow the representation of volume optical properties and implement various integrators to solve the radiative transfer equation. But, as above mentioned, we decided to define semi-transparent properties in an independent manner and to perform integration using the *mcm* library. A difficulty that we encountered was that *pbirt* is not initially under the form of a ready for use C++ object class library: it is presented as a code performing image rendering. We therefore needed to make further choices concerning the way we should extract the required objects in a way compatible with both *mcm*'s structure and *pbirt*'s parser: we wanted that the Monte Carlo algorithm be entered by the programmer exactly as defined in *mcmMonteCarlo* (the higher level *mcm* object class), and that scene description be performed using the exact same input file as that of *pbirt*. This was achieved by modifying *Scene*, the higher level *pbirt* object class, so that it inherits of *mcmMonteCarlo*. Without entering into further technical details, this allows that we keep all the flexibility required to implement the algorithmic diversity discussed in Section 2 (and illustrated in Section 4), and that we still

benefit of all the experience of modelling complex scenes using the *pbirt*'s input file format. As a trivial by-product, *mcm3d* still allows the use of *pbirt*'s original code to make images of concentrated solar systems, which, outside communication purposes, is useful during the scene description debug phases (See the image examples provided in the figures of Section 4).

Altogether, the practice of implementing a Monte Carlo algorithm with *mcm3D* implies the edition of two files: one in which the *W* sampling algorithm is to be coded using the C++ programming language (together with the *mcm* and *pbirt* object classes, in particular the *mcm*'s parallel random generator), and one in which the scene is to be described using the *pbirt* input conventions. Both the *W*-sampling algorithm file and the scene description file are provided on the EDStar website (EDStar, xxxx) for the very simple shape factor of Fig. 1, as well as for each of the four concentrated solar applications of Section 4.

4. Concentrated solar application examples

Each of the simulation examples reported hereafter is extracted from an independent concentrated solar research project, which conclusions will be exposed in dedicated publications. Here, only the Monte Carlo algorithms and their convergence features are presented: physical interpretations and potential conclusions in terms of process design are omitted. For each application a picture of the real process is provided together with a scheme of the corresponding simulated radiative configuration. Computation times are given in the captions of the figures displaying the simulation results. They correspond to a single process on a Quad-Core Intel Xeon 2.66 GHz hardware and linearity was systematically checked both as far as sample size and parallelization are concerned. The reader should find all means to reproduce each of the following simulation results on the associated website (EDStar, xxxx).

The first application (Section 4.1, Fig. 6) deals with a system of 132 heliostats pointing toward a central tower receiver. The geometry and the shadowing and blocking effects are quite complex, but the algorithm is very simple in the sense that always exactly one reflection occurs. The second application (Section 4.2, Fig. 9, a linear Fresnel reflector) is very similar except that multiple reflection occurs within the receiver. In these two first applications sensitivities to mirror imperfections and pointing errors are simultaneously computed. The third application is the fluidized bed receiver that we used for illustration all along the theoretical presentation (Section 4.3, Fig. 11). The corresponding geometry is quite simple, but an additional algorithmic complexity is illustrated with the occurrence of both multiple reflection and multiple scattering in an heterogeneous medium. The sensitivity to the particle density spatial distribution is computed. The last example (Section 4.4, Fig. 13, an enclosed solar photobioreactor) is similar as far as algorithmic complexity is concerned, but the geometrical complexity is high (979 parallel cylindrical

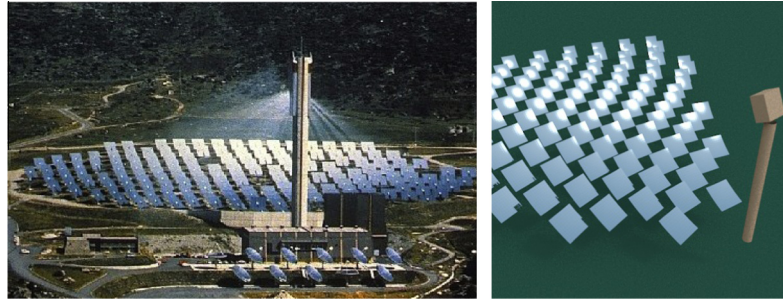


Fig. 6. (a) French themis solar thermal power plant (1986). (b) The simulated central receiver system involves 132 heliostats, each based on a square spherical structure of 10 m × 10 m and constituted with adjacent 1 m × 1 m square mirrors. The receiver (a 10 m × 10 m square) is at the top of a tower of height h (typically between 50 m and 100 m).

optical fibers) and spectral dependencies of optical properties are accounted for. The computed sensitivity concerns the parameters of the size distribution of the absorbing and scattering photosynthetic microorganisms, that involves a coupling with electromagnetic codes allowing the cross section characterization. Note that all evaluated sensitivities belong to the a and b cases defined in Section 2.3. We exclude the more complex c and d cases for which, at the present stage, engineers may still find it easier to evaluate sensitivities by repeating simulations with finite-size perturbed parameter values, which is practically feasible in the here-addressed examples considering our reported computation times.

4.1. Central receiver system

Central receiver systems for solar energy concentration and collection (see Fig. 6a) are based on fields of sun-tracking mirrors (heliostats) that reflect the incident solar flux toward a receiver at the top of a tower. From 80% to 95% of the reflected energy is absorbed into a working fluid. The main advantage of central receiver systems is

the high temperature levels (up to 1000 K) that allow to reach high power cycle efficiency.

From a radiative point of view, the simulated system is sketched in Fig. 6b. The evaluated quantity is the solar power A impacting the target T^+ (the receiver) after one reflection by the heliostat field. Both pointing imperfections and shadowing/blocking effects are accounted for. The algorithm starts with the sampling of a location x_1 on the reflective surface of the whole heliostat field \mathcal{H}^+ of area $S_{\mathcal{H}^+}$ (the total surface of the heliostat field, including the non-reflective back faces, is noted \mathcal{H}). Then a direction ω_S is sampled within the solar cone \mathcal{C} of angular radius θ_S , defining the incidence direction $\omega_0 = -\omega_S$. In order to check that no shading occurs, x_0 is defined as the first intersection with a solid surface of the ray starting at x_1 in the direction ω_S and it is tested that x_0 does not belong to \mathcal{H} or \mathcal{T} (the total wall surface of the tower, including the target). When no shading occurs, a reflected direction ω_1 is sampled in order to represent reflection and pointing imperfections, and it is checked that the first intersection x_2 of the ray starting at x_1 in the direction ω_1 belongs to the target (no blocking occurs). Reflection imperfections (due to surface quality and shape uncertainties) and pointing imperfections lead to an overall angular spreading that is modeled all together as a distribution of effective normal vectors \mathbf{n}_h at x_1 around the ideal normal vector \mathbf{n}_1 , and ω_1 corresponds to the specular reflection of ω_1 by a surface of normal \mathbf{n}_h (see Fig. 7).

The associated integral formulation is the following:

$$A = \int_{\mathcal{H}^+} p_{X_1}(\mathbf{x}_1) d\mathbf{x}_1 \int_{\mathcal{C}} p_{\Omega_S}(\omega_S) d\omega_S \{ H(\mathbf{x}_0 \in (\mathcal{H} \cup \mathcal{T})) \hat{w}_{out} + H(\mathbf{x}_0 \notin (\mathcal{H} \cup \mathcal{T})) \int_{\mathcal{D}_{N_h}(\omega_0)} p_{N_h}(\mathbf{n}_h | \omega_0; \beta) d\mathbf{n}_h \{ H(\mathbf{x}_2 \notin T^+) \hat{w}_{out} + H(\mathbf{x}_2 \in T^+) \hat{w}_{in} \} \} \quad (53)$$

with

$$p_{X_1}(\mathbf{x}_1) = \frac{1}{S_{\mathcal{H}^+}} \quad (54)$$

$$p_{\Omega_S}(\omega_S) = \frac{1}{\int_{\mathcal{C}} d\omega_S} = \frac{1}{2\pi(1 - \cos \theta_S)} \quad (55)$$

$$\hat{w}_{out} = 0 \quad (56)$$

$$\hat{w}_{in} = DNI (\omega_S \cdot \mathbf{n}_h) \cdot S_{\mathcal{H}^+} \quad (57)$$

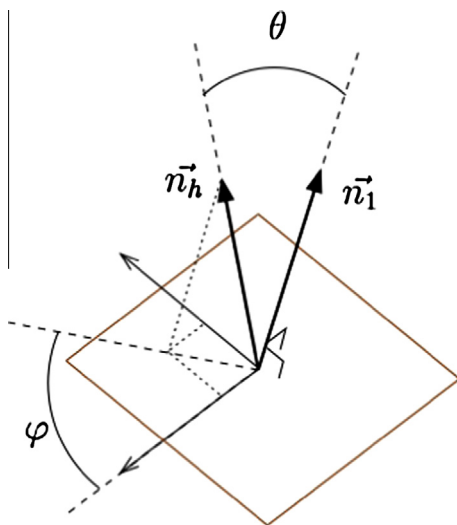


Fig. 7. In the algorithm, the integral $\int_{\mathcal{D}_{N_h}(\omega_0)} p_{N_h}(\mathbf{n}_h | \omega_0; \beta) d\mathbf{n}_h$ is translated into the sampling of the spherical angles $(\theta; \varphi)$ and the reflected direction ω_1 is then directly computed as $\omega_1 = 2(\omega_S \cdot \mathbf{n}_h) \mathbf{n}_h - \omega_S$.

where DNI is the direct normal insolation. For probability density function p_{N_h} , the Blinn's model is used³, of parameter β , with a truncation of the distribution of $\mathbf{n}_h \cdot \mathbf{n}_1$ avoiding the occurrence of reflected directions toward the surface for quasi-tangent incidences. This leads to

$$p_{N_h}(\mathbf{n}_h|\omega_0; \beta) = \frac{2 + \frac{1}{\beta}}{2\pi(1 - \mu(\omega_0)^{2+\frac{1}{\beta}})} (\mathbf{n}_h \cdot \mathbf{n}_1)^{1+\frac{1}{\beta}} \quad (58)$$

The definition domain $\mathcal{D}_{N_h}(\omega_0)$ corresponds to the part of the unit hemisphere such that $\mathbf{n}_h \cdot \mathbf{n}_1 < \mu(\omega_0)$, where $\mu(\omega_0) = \cos(\frac{\pi}{4} - \frac{1}{2} \arccos(\omega_0 \cdot \mathbf{n}_1))$.

Because of truncation, the angular spreading resulting of this Blinn's like reflection is dependent on the incident direction (even if this dependence tends to zero when β tends to zero, that is to say when the spreading is weak as for most solar applications). A measure of the angular spreading is the quadratic average deviation ϵ in the particular case $\omega_0 = -\mathbf{n}_1$, i.e.

$$\begin{aligned} \epsilon &= \sqrt{\langle \theta^2 \rangle} \approx \sqrt{\langle \sin(\theta)^2 \rangle} \\ &= \sqrt{1 - \left(\frac{2 + \frac{1}{\beta}}{4 + \frac{1}{\beta}} \right) \left[\frac{1 - \left(\frac{\sqrt{2}}{2} \right)^{4+\frac{1}{\beta}}}{1 - \left(\frac{\sqrt{2}}{2} \right)^{2+\frac{1}{\beta}}} \right]} \end{aligned} \quad (59)$$

In order to evaluate the sensitivity $\partial_{\epsilon} A$ of A to the angular spreading ϵ , we write

$$\partial_{\epsilon} A = \frac{\partial A}{\partial \epsilon} = \frac{\partial A}{\partial \beta} \frac{\partial \beta}{\partial \epsilon} = \frac{\partial A}{\partial \beta} \cdot \left(\frac{d\epsilon}{d\beta} \right)^{-1} \quad (60)$$

and the sensitivity weights become

$$\hat{w}_{\epsilon, out} = 0 \quad (61)$$

$$\hat{w}_{\epsilon, in} = \hat{w}_{in} \frac{\partial_{\beta} p_{N_h}(\mathbf{n}_h|\omega_0; \beta)}{p_{N_h}(\mathbf{n}_h|\omega_0; \beta)} \cdot \left(\frac{d\epsilon}{d\beta} \right)^{-1} \quad (62)$$

with

$$\begin{aligned} \frac{\partial_{\beta} p_{N_h}(\mathbf{n}_h|\omega_0; \beta)}{p_{N_h}(\mathbf{n}_h|\omega_0; \beta)} &= \frac{-1}{\beta^2} \\ &\cdot \left[\frac{1}{2 + \frac{1}{\beta}} + \frac{\ln(\mu(\omega_0))\mu(\omega_0)^{2+\frac{1}{\beta}}}{1 - \mu(\omega_0)^{2+\frac{1}{\beta}}} + \ln(\mathbf{n}_h \cdot \mathbf{n}_1) \right] \end{aligned} \quad (63)$$

Fig. 8 displays the simulation results corresponding to A and $\partial_{\epsilon} A$ as function of the tower height h , for $\beta = 10^{-5}$ corresponding to $\epsilon = 4.5$ mrad. The very same algorithm was implemented on a faster purely dedicated platform (by opposition to EDSTAR that is a general purpose development environment) and the resulting

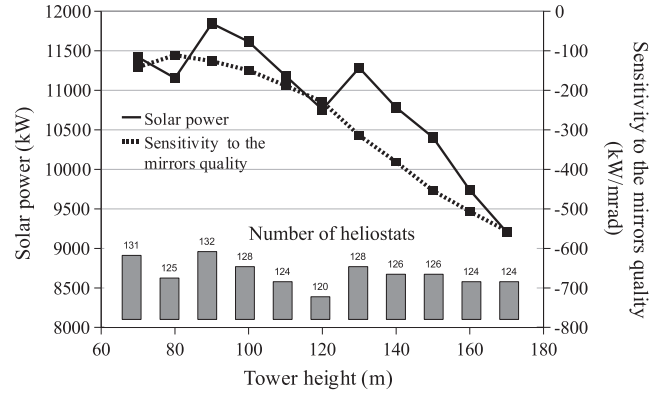


Fig. 8. Simulation results corresponding to A and $\partial_{\epsilon} A$ as defined in Eq. (53) with the appropriate weights, as function of the tower height h . Each point is evaluated using 10^6 samples and requires a computational time of approximately 30 s.

computation efficiencies were compared to the most standard available concentrated solar codes in Rocchia (2012).

4.2. Linear Fresnel reflector

Linear Fresnel reflectors are used for solar concentration as a promising alternative to parabolic troughs. They are composed of mirror bands tilted to all reflect the light on a single linear receiver over the mirror field (see Fig. 9a and b). The receiver ensures the heating of flowing air, acting as the heat source of a thermodynamic cycle. The simplicity of the design saves costs that compensate for lower efficiency (Dersch et al., 2009).

From a radiative point of view, the simulated system is sketched in Fig. 9c. The evaluated quantity is the lineic solar power A impacting the absorbing pipe \mathcal{P} within the receiver. The tested Monte Carlo algorithm is identical to that of the preceding example up to the stage where the ray impacts the glass window \mathcal{T}^+ . The symbol \mathcal{H} corresponds now to the union of all mirror bands, \mathcal{H}^+ is their reflective side, and \mathcal{T} is the union of all the external surfaces of the receiver system: $\mathcal{T} = \mathcal{T}^+ \cup \mathcal{R}^-$ where \mathcal{R}^- is the external surface of the secondary reflector. The difference starts once the location \mathbf{x}_2 is found on the glass window. The fraction of the solar power transmitted through the window impacts either the external surface \mathcal{R}^+ of the secondary reflector or \mathcal{P} at a location \mathbf{x}_3 . If \mathbf{x}_3 belongs to \mathcal{R}^+ a reflection direction is sampled according to a truncated Blinn's model and the reflected power is further tracked within the receiver. If \mathbf{x}_3 belongs to \mathcal{P} a reflection test is performed: either absorption occurs and the tracking ends, otherwise a reflection direction is sampled according to Lambert distribution and the ray is further tracked within the receiver. In all cases leading to a further tracking of the ray, a new surface intersection \mathbf{x}_4 is defined within $\mathcal{R}^+ \cup \mathcal{P} \cup \mathcal{T}^+$. If \mathbf{x}_4 belongs to $\mathcal{R}^+ \cup \mathcal{P}$ the algorithm continues as for \mathbf{x}_3 and if \mathbf{x}_4 belongs to the glass window \mathcal{T}^+ the specular

³ BLINN proposed a model where the distribution of microfacet normals is approximated by an exponential falloff (Pharr and Humphreys, 2004).

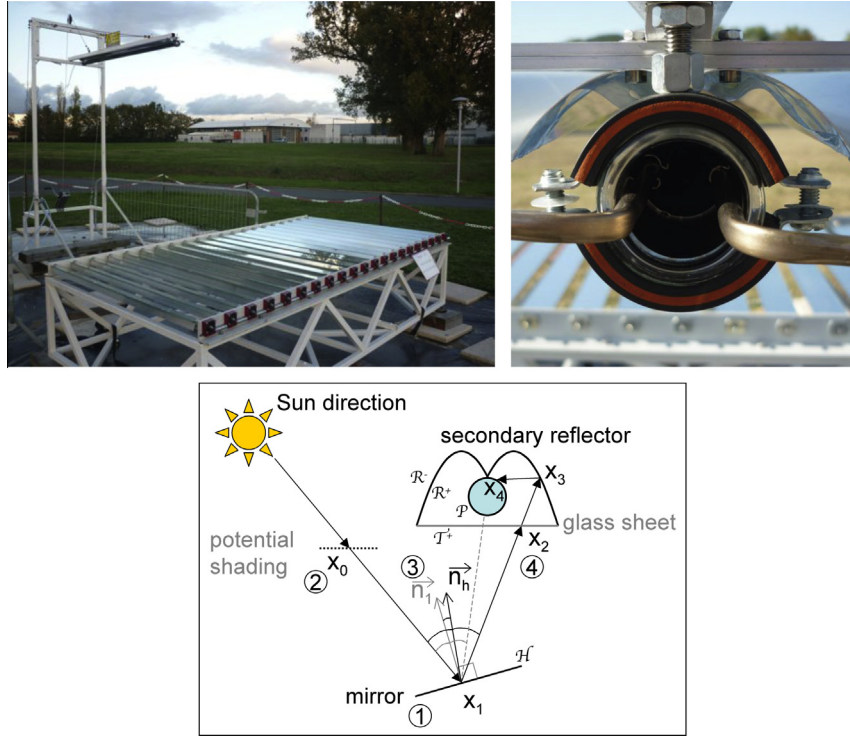


Fig. 9. (a) The simulated linear Fresnel reflector prototype; (b) details of the corresponding linear receiver; (c) receiver scheme and geometrical definitions. The prototype involves 21 flat 0.1 m width and 1.5 m long mirrors of reflectivity $\rho_{\mathcal{H}^+}=0.91$. The total width of the concentrator is 3 m. The linear receiver is 1.8 m above the mirror field. This receiver involves a 2 m long and 3.8 cm diameter diffuse gray absorbing pipe of reflectivity $\rho_P = 0.1$ (we usually mention an receiver tube absorptivity of 0.9). Around this pipe is a 12 cm wide gray reflective secondary reflector of internal reflectivity $\rho_{\mathcal{R}^+}=0.95$, closed with a flat glass window of reflectivity $\rho_{T^+} = 0.04$ and transmissivity $\tau_{T^+} = 0.92$. The glass window reflection is perfectly specular. The imperfections of the primary reflection are modeled according to the same truncated Blinn model as in the preceding example, with a Blinn parameter value $\beta_{\mathcal{H}^+} = 5.10^{-5}$ corresponding to a quadratic average deviation $\epsilon_{\mathcal{H}^+} = 10$ mrad. The truncated Blinn model is used with the same parameter to model the imperfections of reflexion on the secondary reflector, as well as on the glass sheet and on the receiver tube.

reflected power is further tracked within the receiver.⁴ All optical properties are given in the caption of Fig. 9.

The associated integral formulation is the following:

$$A = \int_{\mathcal{H}^+} p_{X_1}(\mathbf{x}_1) d\mathbf{x}_1 \int_{\mathcal{C}} p_{\Omega_S}(\omega_S) d\omega_S \{ H(\mathbf{x}_0 \in (\mathcal{H} \cup \mathcal{T})) \hat{w}_{out} + H(\mathbf{x}_0 \notin (\mathcal{H} \cup \mathcal{T})) \int_{\mathcal{D}_{N_{h,1}}(\omega_0)} p_{N_{h,1}}(\mathbf{n}_{h,1} | \omega_0; \beta_{\mathcal{H}^+}) d\mathbf{n}_{h,1} \times \{ H(\mathbf{x}_2 \notin \mathcal{T}^+) \hat{w}_{out} + H(\mathbf{x}_2 \in \mathcal{T}^+) \{ H(\mathbf{x}_3 \in \mathcal{P}) \int_0^1 p_{R_3}(r_3) dr_3 \times \left\{ H(r_3 > \rho_P) \hat{w}_3 + H(r_3 < \rho_P) \int_{2\pi} p_{\Omega_3}(\omega_3) d\omega_3 A_3 \right\} + H(\mathbf{x}_3 \in \mathcal{R}^+) \int_{\mathcal{D}_{N_{h,3}}(\omega_2)} p_{N_{h,3}}(\mathbf{n}_{h,3} | \omega_2; \beta_{\mathcal{R}^+}) d\mathbf{n}_{h,3} A_3 \} \} \} \quad (64)$$

with

⁴ Note that the power transmitted outside the receiver is not taken into account in the integral formulation in Eq. (64) because its fraction reflected back inside the receiver from the mirror bands is negligible (lower than 0.5%).

$$A_j = H(\mathbf{x}_{j+1} \in \mathcal{P}) \int_0^1 p_{R_{j+1}}(r_{j+1}) dr_{j+1} \{ H(r_{j+1} > \rho_P) \hat{w}_{j+1} + H(r_{j+1} < \rho_P) \int_{2\pi} p_{\Omega_{j+1}}(\omega_{j+1}) d\omega_{j+1} A_{j+1} \} + H(\mathbf{x}_{j+1} \in \mathcal{R}^+) \int_{\mathcal{D}_{N_{h,j+1}}(\omega_j)} p_{N_{h,j+1}}(\mathbf{n}_{h,j+1} | \omega_j; \beta_{\mathcal{R}^+}) d\mathbf{n}_{h,j+1} A_{j+1} + H(\mathbf{x}_{j+1} \in \mathcal{T}^+) A_{j+1} \quad (65)$$

and

$$\omega_2 = \omega_1 \quad (66)$$

$$p_{R_q}(r_q) = 1 \quad (67)$$

$$p_{\Omega_q}(\omega_q) = \frac{\omega_q \cdot \mathbf{n}_q}{\pi} \quad (68)$$

$$\hat{w}_{out} = 0 \quad (69)$$

$$\hat{w}_3 = \tau_{T^+} \rho_{\mathcal{H}^+} DNI (\omega_S \cdot \mathbf{n}_{h,1}) \cdot S_{\mathcal{H}^+} \quad (70)$$

$$\hat{w}_{j+1} = \left[\prod_{q=3}^j (H(\mathbf{x}_q \in \mathcal{P}) + H(\mathbf{x}_q \in \mathcal{R}^+) \rho_{\mathcal{R}^+} + H(\mathbf{x}_q \in \mathcal{T}^+) \rho_{T^+}) \right] \tau_{T^+} \rho_{\mathcal{H}^+} DNI (\omega_S \cdot \mathbf{n}_{h,1}) \cdot S_{\mathcal{H}^+} \quad (71)$$

For the evaluation of the sensitivity $\partial_{\epsilon_{H^+}} A$ of A to the angular spreading ϵ_{H^+} , the same approach is used as in the preceding example, with

$$\hat{w}_{\epsilon_{H^+},out} = 0 \quad (72)$$

$$\hat{w}_{\epsilon_{H^+},k} = \hat{w}_k \frac{\partial_{\beta} p_{N_{h,1}}(\mathbf{n}_{h,1}|\omega_0; \beta_{H^+})}{p_{N_{h,1}}(\mathbf{n}_{h,1}|\omega_0; \beta_{H^+})} \cdot \left(\frac{d\epsilon_{H^+}}{d\beta_{H^+}} \right)^{-1} \quad (73)$$

The evaluation of the sensitivity $\partial_{\epsilon_{R^+}} A$ of A to the angular spreading ϵ_{R^+} involves multiple reflection and is handled according to the procedure introduced in Section 2.3, Eq. (52), which leads to

$$\hat{w}_{\epsilon_{R^+},out} = 0 \quad (74)$$

$$\hat{w}_{\epsilon_{R^+},3} = 0 \quad (75)$$

$$\hat{w}_{\epsilon_{R^+},j+1} = \hat{w}_{j+1} \sum_{q=3}^j H(\mathbf{x}_q \in \mathcal{R}^+) \frac{\partial_{\beta} p_{N_{h,q}}(\mathbf{n}_{h,q}|\omega_{q-1}; \beta_{R^+})}{p_{N_{h,q}}(\mathbf{n}_{h,q}|\omega_{q-1}; \beta_{R^+})} \cdot \left(\frac{d\epsilon_{R^+}}{d\beta_{R^+}} \right)^{-1} \quad (76)$$

Fig. 10 displays the simulation results corresponding to A and $\partial_{\epsilon_{H^+}} A$ as function of time (along the day).

4.3. Fluidized bed receiver

Fluidized beds are commonly used as thermal exchangers for high temperature processes. A gap toward the improvement of thermodynamic cycles for solar electricity production will be the finding of a technological solution for direct gas heating within solar receivers (with the objective of reaching temperature levels higher than 1000 K at the turbine entrance) and the use of solid fluidized particles directly submitted to the concentrated solar flux is an extensively explored foreseen solution (Flamant and

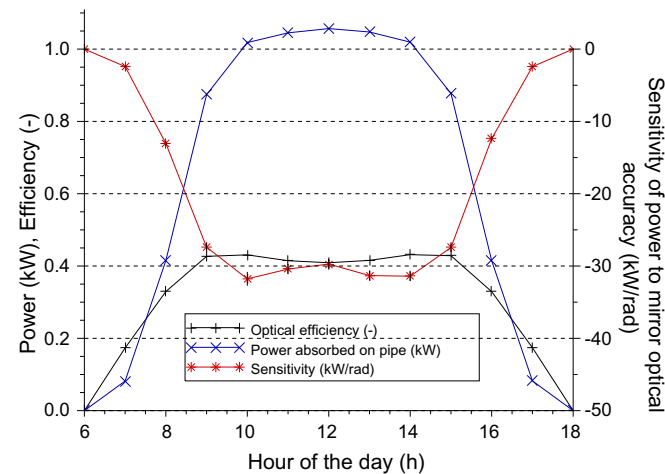


Fig. 10. Simulation results corresponding to A and $\partial_{\epsilon_{H^+}} A$ as defined in Eq. (64) with the appropriate weights, as function of time (along the day). Each point is evaluated using 10^6 samples and requires a computational time of approximately 12 min (without parallelization, nor any algorithmic optimization).

Menigault, 1986; Haddad and Elsayed, 1988; Muller et al., 2003; Trommer, 2005).

The system considered hereafter is described technically in Fig. 11. It corresponds to a completed version of the fluidized bed receiver example that we used for illustration all along the theoretical presentation of Section 2. The particle density η is heterogeneous: above a given altitude, it decreases exponentially according to a scale height h that is highly dependant on the speed of the incoming air flow. The absorption and scattering coefficients $k_{a,v}$ and $k_{s,v}$ are assumed independent of frequency and are related to the Planck average absorption and scattering cross sections σ_a and σ_s according to $k_{a,v} = \eta \sigma_a$ and $k_{s,v} = \eta \sigma_s$. The evaluated quantity A is the fraction of the incoming solar radiative power that exits the receiver through the quartz window \mathcal{G} after reflection on the stainless steel surface \mathcal{R} and/or scattering by particles within the core \mathcal{V} . As described in Section 2.2 and Fig. 5, the algorithm starts with the sampling of an incidence location \mathbf{x}_0 on \mathcal{G} and the incidence direction ω_0 is perpendicular to \mathcal{G} . Then a scattering optical thickness κ_0 is sampled exponentially

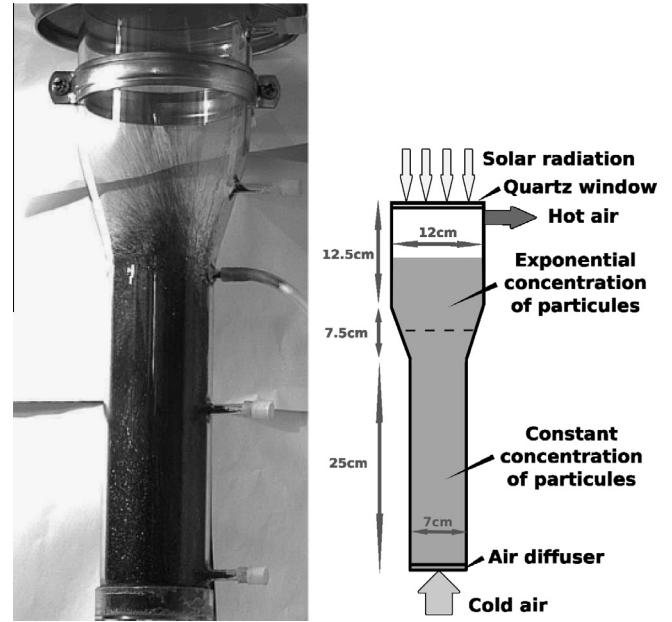


Fig. 11. (a) The simulated fluidized bed receiver; (b) scheme and dimensions. The receiver includes a diffuser at the bottom (cold air entrance), a lower cylindrical column filled with dense homogeneously distributed silicon carbide particles (particle density around $\eta_0 = 26.10^3 \text{ kg m}^{-3}$), and a higher cylinder-conical column surmounted by a quartz window. The particle density decreases exponentially with altitude above $z_0 = 27 \text{ cm}$ according to $\eta(\mathbf{x}; h) = \eta_0 \exp(-\frac{z-z_0}{h})$ where h is the decrease scale height and z is the vertical coordinate of the position vector \mathbf{x} (the altitude $z = 0$ corresponds to the top of the air diffuser). The stainless steel surface of the receiver is gray diffuse reflective with a reflectivity $\rho^R = 0.4$. The quartz window is gray specular reflective with a reflectivity $\rho^G = 0.1$. The particles are spherical with a monodisperse distribution radius $r_p = 280 \mu\text{m}$. The Planck average absorption and scattering cross sections σ_a and σ_s , as well as the asymmetry parameter of the phase function g , are obtained using Mie theory (Mie, 1908; van de Hulst, 1957) between $0.24 \mu\text{m}$ and $4 \mu\text{m}$: $\sigma_a = 0.50$, $\sigma_s = 1.50$, $g = 0.70$. The Henyey–Greenstein phase function is used.

and defines an interaction location \mathbf{x}_1 that is at the surface ($\mathbf{x}_1 \in \mathcal{R}$) if κ_0 is greater than the optical thickness of the air-particle mixture from \mathbf{x}_0 to the surface, and is within the volume otherwise ($\mathbf{x}_1 \in \mathcal{V}$). If \mathbf{x}_1 is at the surface, a direction ω_1 is sampled according to the reflection angular distribution $p_{\Omega_1}^{\mathcal{R}}$. If \mathbf{x}_1 is within the volume, ω_1 is sampled according to single scattering phase function $p_{\Omega_1}^{\mathcal{V}}$. In both cases a new scattering optical thickness κ_1 is sampled and the algorithm continues recursively with κ_1 defining a new interaction location \mathbf{x}_2 , etc., until one of the interaction locations, say \mathbf{x}_{j+1} belongs to the quartz window. Then a random number $r_{j+1}^{\mathcal{R}}$ is sampled uniformly on the unit interval and the optical path exits the window if $r_{j+1}^{\mathcal{R}}$ is greater than the window's reflectivity $\rho^{\mathcal{R}}$ and is specularly reflected otherwise ($\omega_{j+1} = \omega_j - 2(\omega_j \cdot \mathbf{n}_{j+1})\mathbf{n}_{j+1}$ where \mathbf{n}_{j+1} is the exit unit normal at \mathbf{x}_{j+1}). If the extinction along the multiple reflection, multiple scattering optical path reaches a given threshold (the weight \hat{w}_j is lower than a fixed value ϵ), the algorithm switches to a russian roulette for representation of reflection at each \mathcal{R} encounter, which interrupts long paths contributing only very little to the evaluation of A , without introducing any statistical bias.

The associated integral formulation is the following:

$$A = \int_{\mathcal{G}} p_{\mathbf{x}_0}(\mathbf{x}_0) d\mathbf{x}_0 \int_0^\infty p_{\kappa_0}(\kappa_0) d\kappa_0 \left\{ \begin{aligned} &H(\mathbf{x}_1 \in \mathcal{R}) \int_{2\pi} p_{\Omega_1}^{\mathcal{R}}(\omega_1) d\omega_1 \int_0^\infty p_{\kappa_1}(\kappa_1) d\kappa_1 \mathcal{A}_1 \\ &+ H(\mathbf{x}_1 \in \mathcal{V}) \int_{4\pi} p_{\Omega_1}^{\mathcal{V}}(\omega_1|\omega_0) d\omega_1 \int_0^\infty p_{\kappa_1}(\kappa_1) d\kappa_1 \mathcal{A}_1 \end{aligned} \right\} \quad (77)$$

where \mathcal{A}_j is recursively defined as

$$\begin{aligned} \mathcal{A}_j = &H(\mathbf{x}_{j+1} \in \mathcal{G}) \int_0^1 p_{R_{j+1}^{\mathcal{G}}}(r_{j+1}^{\mathcal{G}}) dr_{j+1}^{\mathcal{G}} \left\{ \begin{aligned} &H(r_{j+1}^{\mathcal{G}} > \rho^{\mathcal{G}}) \hat{w}_{j+1} \\ &+ H(r_{j+1}^{\mathcal{G}} \leq \rho^{\mathcal{G}}) \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} \end{aligned} \right\} \\ &+ H(\mathbf{x}_{j+1} \in \mathcal{R}) \left\{ \begin{aligned} &H(\hat{w}_j \leq \epsilon) \int_0^1 p_{R_{j+1}^{\mathcal{R}}}(r_{j+1}^{\mathcal{R}}) dr_{j+1}^{\mathcal{R}} \times \dots \\ &\left\{ \begin{aligned} &H(r_{j+1}^{\mathcal{R}} > \rho^{\mathcal{R}}) \times \hat{w}_{out} \\ &+ H(r_{j+1}^{\mathcal{R}} \leq \rho^{\mathcal{R}}) \int_{2\pi} p_{\Omega_{j+1}}^{\mathcal{R}}(\omega_{j+1}) d\omega_{j+1} \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} \end{aligned} \right\} \\ &+ H(\hat{w}_j > \epsilon) \int_{2\pi} p_{\Omega_{j+1}}^{\mathcal{R}}(\omega_{j+1}) d\omega_{j+1} \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} \end{aligned} \right\} \\ &+ H(\mathbf{x}_{j+1} \in \mathcal{V}) \int_{4\pi} p_{\Omega_{j+1}}^{\mathcal{V}}(\omega_{j+1}|\omega_j) d\omega_{j+1} \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} \end{aligned} \quad (78)$$

with

$$p_{\mathbf{x}_0}(\mathbf{x}_0) = \frac{1}{S_{\mathcal{G}}} \quad (79)$$

$$p_{\kappa_j}(\kappa_j) = \exp(-\kappa_j) \quad (80)$$

$$p_{\Omega_j}^{\mathcal{R}}(\omega_j) = \frac{\omega_j \cdot \mathbf{n}_j}{\pi} \quad (81)$$

$$p_{\Omega_{j+1}}^{\mathcal{V}}(\omega_{j+1}|\omega_j) = \frac{1 - g^2}{[1 + g^2 - 2g(\omega_{j+1} \cdot \omega_j)]^{3/2}} \quad (82)$$

$$p_{R_j^{\mathcal{G}}}(r_j^{\mathcal{G}}) = 1 \quad (83)$$

$$\hat{w}_{out} = 0 \quad (84)$$

$$\hat{w}_0 = S_{\mathcal{G}} \quad (85)$$

$$\hat{w}_{j+1} = \hat{w}_j \exp(-\sigma_a \alpha_j) \left\{ \begin{aligned} &H(\mathbf{x}_{j+1} \in \mathcal{G}) \times 1 \\ &+ H(\mathbf{x}_1 \in \mathcal{R}) \left\{ \begin{aligned} &H(\hat{w}_j > \epsilon) \rho^{\mathcal{R}} \\ &+ H(\hat{w}_j \leq \epsilon) \times 1 \end{aligned} \right\} \\ &+ H(\mathbf{x}_1 \in \mathcal{V}) \times 1 \end{aligned} \right\} \quad (86)$$

$$\alpha_j = \int_0^{\|\mathbf{x}_{j+1} - \mathbf{x}_j\|} \eta(\mathbf{x}_j + s\omega_j; h) ds \quad (87)$$

The evaluation of the sensitivity $\partial_h A$ of A to the scale height h involves inhomogeneous multiple reflection, multiple scattering and is handled according to the procedure introduced in Section 2.3, Eqs. (49)–(52). The only difference is that the weight is a function of the parameter. Adding the weight derivative to the right side of Eq. (52) leads to

$$\hat{w}_{h,j+1} = \partial_h \hat{w}_{h,j+1} + \hat{w}_{j+1} \sum_{q=0}^j \frac{\partial_h p_{\lambda_q}}{p_{\lambda_q}} \quad (88)$$

which translates into

$$\hat{w}_{h,j+1} = \hat{w}_{j+1} \sum_{q=0}^j \left[-(\sigma_a + \sigma_s) \partial_h \alpha_q + H(x_{q+1} \in \mathcal{V}) \frac{\partial_h \eta(\mathbf{x}_{q+1}; h)}{\eta(\mathbf{x}_{q+1}; h)} \right] \quad (89)$$

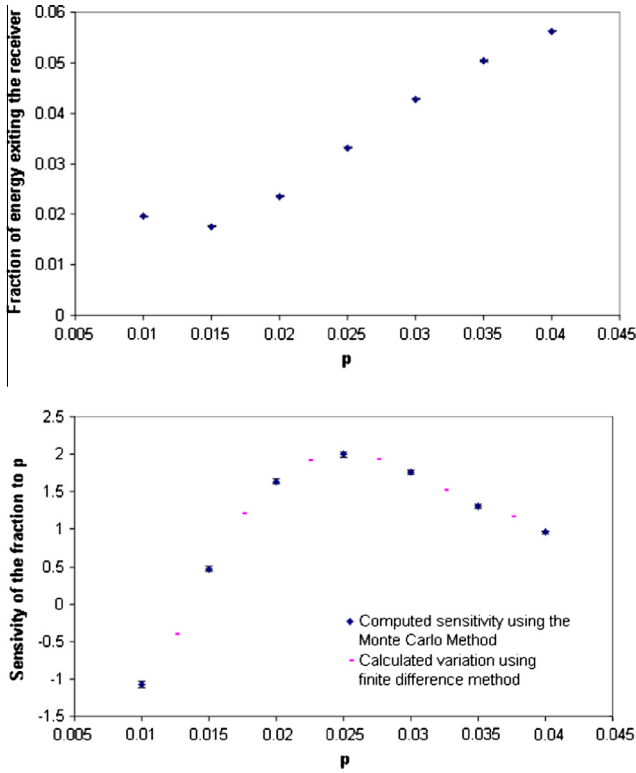


Fig. 12. Simulation results corresponding to A and $\partial_p A$ as defined in Eqs. (77) and (78) with the appropriate weights, as function of η_0 , with $\epsilon = 10^{-12}$. $\eta(\mathbf{x}; h) = \eta_0 \exp(-\frac{z-z_0}{h})$ where $h = p$ is the decrease scale height. Each point is evaluated using 10^6 samples and requires a computational time of a few minutes.

when the choice is arbitrarily made that $\partial_h \eta(\mathbf{x}; h) = 0$ for the sampled scattering locations \mathbf{x} that lay outside the receiver (and are algorithmically translated into surface impacts and reflections).

Fig. 12 displays the simulation results corresponding to A and $\partial_p A$ as function of the bottom particle density η_0 .

4.4. Enclosed solar photobioreactor

Enclosed solar photobioreactors are processes of biomass production in which photosynthesis is catalyzed by photosynthetic microorganisms. Future developments of the photobioreactors technology depend on the ability to design large scale prototypes allowing an efficient light utilization. In order to reach this requirement it is necessary to study the field of absorbed energy inside the culture, which is a scattering and absorbing medium where the microorganisms are the multi-diffusers (Cornet and Dussap, 2009; Dauchet et al., 2011).

We focus here on a concept of volumetrically lightened photobioreactor named DiCoFluV (see Fig. 13a) in which the incident solar light flux density is diluted in the volume thanks to light-diffusing optical fibers emitting a quasi homogeneous density flux on the totality

of their surface (Cornet, 2010). The corresponding radiative configuration is shown on Fig. 13b. The presented simulations correspond to cyanobacteria *Arthrospira platensis* PCC 8005 (generic name: *Spirulina*) which radiative properties ($\sigma_{a,v}$, $\sigma_{s,v}$ and g_v) are given in figure Fig. 14. Thus, fixing a uniform microorganism density η (under the perfect mixing assumption), radiative properties of the culture medium \mathcal{V} are all known ($p_{\Omega,v}^v, k_{a,v} = \eta \sigma_{a,v}, k_{s,v} = \eta \sigma_{s,v}$), and we will design a Monte Carlo algorithm to evaluate the photo-synthetically active specific radiant power absorbed A^5 (W kg^{-1}) (see Dauchet et al., 2011) at any location \mathbf{x}_0 inside the volume of culture as well as its sensitivities to η and to the parameters of the *Spirulina* size distribution \bar{a} and s (see Fig. 14 caption).

$$A(\mathbf{x}_0) = \int_{v_{\min}}^{v_{\max}} A_v(\mathbf{x}_0) dv = \int_{v_{\min}}^{v_{\max}} \int_{4\pi} \sigma_{a,v} L_v(\mathbf{x}_0, \omega_0) d\omega_0 dv \quad (90)$$

where $L_v(\mathbf{x}_0, \omega)$ is the radiance at position \mathbf{x}_0 and in the direction ω_0 , and where $[v_{\min}, v_{\max}]$ is the photo-synthetically active radiation domain (see Fig. 14). In the following integral formulation, by comparison with the fluidized bed receiver example Section 4.3, the main differences are essentially the spectral integration and the backward formulation (see Section 2.1). First, a frequency is sampled uniformly according to $p_v(v)$. This frequency determines the radiative properties and the surface flux density φ_v emitted by the fibers \mathcal{F} for the current optical path. Then, starting from $\mathbf{x}_0 \in \mathcal{V}$, a first direction ω_0 is uniformly sampled according to $p_{\Omega_0}^v$ and the successive interaction locations \mathbf{x}_j ($j > 0$) are sampled as in Section 4.3. If \mathbf{x}_j belongs to the stainless steel surface of the reactor \mathcal{R} , ω_j is sampled according to the reflection angular distribution $p_{\Omega_j}^{\mathcal{R}}$ (and the weight is multiplied by $\rho^{\mathcal{R}}$). If $\mathbf{x}_j \in \mathcal{F}$, a Russian roulette is performed: if the random number r_j uniformly sampled over the unit interval is lower than the fibers reflectivity $\rho^{\mathcal{F}}$, then the optical path is reflected and ω_j is sampled according to $p_{\Omega_j}^{\mathcal{F}}$, otherwise optical path sampling ends and the associated weight is the radiance emitted by the fiber and attenuated along the optical path. Hence, the specific power absorbed $A(\mathbf{x}_0)$ is calculated from:

⁵ Specific radiant power is defined as the radiative power per mass unit.

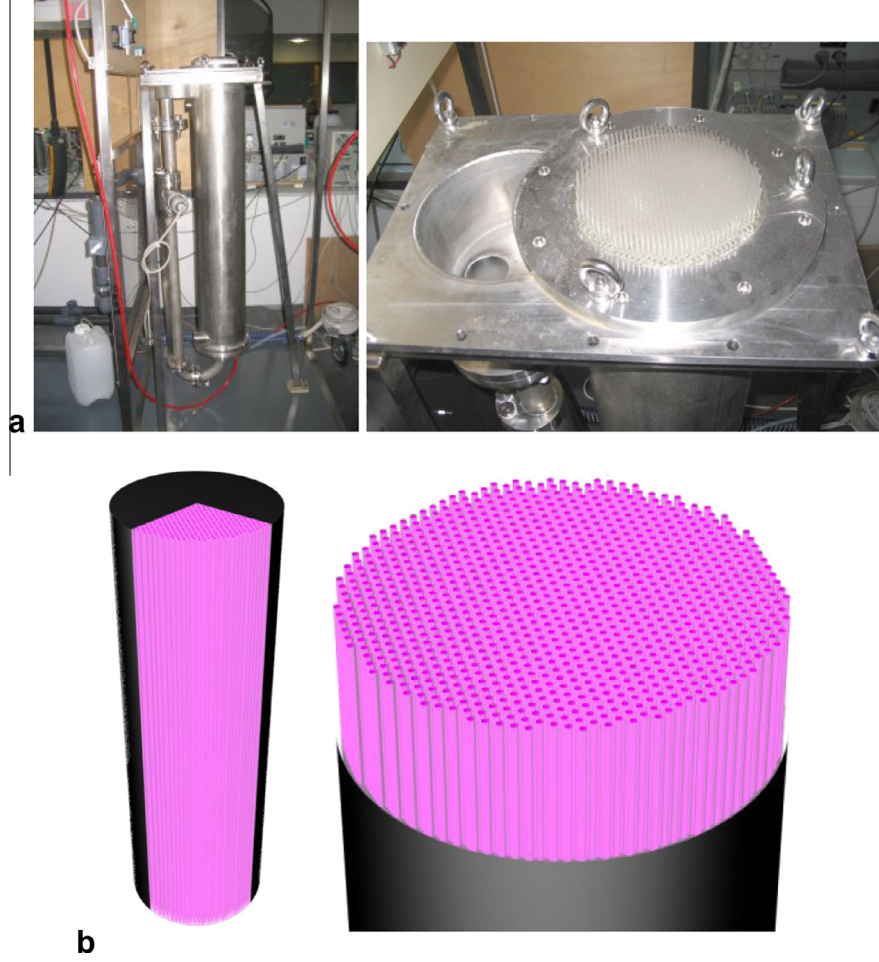


Fig. 13. (a) Prototype of volumetrically lightened photobioreactor DiCoFluV (Dilution Contrôlée du Flux en Volume) (Cornet, 2010). (b) EDStar geometry: both reactor (\mathcal{R}) and fibers (\mathcal{F}) are cylinders of height 1 m; reactor diameter 16.5 cm; fibers axis are organized on a 2D hexagonal lattice of angle $\frac{\pi}{3}$, distance between two fiber axis is 4.8 mm; fiber diameter 2.4 mm; minimal distance between a fiber axis and the reactor 3.6 mm; 979 fibers in total. The stainless steel surface of the reactor \mathcal{R} is modeled as a diffuse-reflective surface with a uniform reflectivity $\rho^{\mathcal{R}}$, and the light-diffusing optical fibers surfaces \mathcal{F} are modeled as diffuse-reflective with a uniform reflectivity $\rho^{\mathcal{F}}$ and Lambertian emitting with a uniform surface flux density ϕ_v .

$$A(\mathbf{x}_0) = \int_{v_{\min}}^{v_{\max}} p_v(v) dv \int_{4\pi} p_{\Omega_0}^{\mathcal{V}}(\omega_0) d\omega_0 \int_0^\infty p_{\kappa_0}(\kappa_0) d\kappa_0 \times \left\{ \begin{aligned} &H(\mathbf{x}_1 \in \mathcal{R}) \int_{2\pi} p_{\Omega_1}^{\mathcal{R}}(\omega_1) d\omega_1 \int_0^\infty p_{\kappa_1}(\kappa_1) d\kappa_1 \mathcal{A}_1 + \\ &H(\mathbf{x}_1 \in \mathcal{F}) \int_0^1 p_{R_1}(r_1) dr_1 \left\{ \begin{aligned} &H(r_1 \leq \rho^{\mathcal{F}}) \int_{2\pi} p_{\Omega_1}^{\mathcal{F}}(\omega_1) d\omega_1 \int_0^\infty p_{\kappa_1}(\kappa_1) d\kappa_1 \mathcal{A}_1 + \\ &H(r_1 > \rho^{\mathcal{F}}) \hat{w}_1 \end{aligned} \right\} + \\ &H(\mathbf{x}_1 \in \mathcal{V}) \int_{4\pi} p_{\Omega_1, v}^{\mathcal{V}}(\omega_1 | \omega_0) d\omega_1 \int_0^\infty p_{\kappa_1}(\kappa_1) d\kappa_1 \mathcal{A}_1 \end{aligned} \right\} \quad (91)$$

where \mathcal{A}_j is recursively defined as

$$\mathcal{A}_j = \left\{ \begin{aligned} &H(\mathbf{x}_{j+1} \in \mathcal{R}) \int_{2\pi} p_{\Omega_{j+1}}^{\mathcal{R}}(\omega_{j+1}) d\omega_{j+1} \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} + \\ &H(\mathbf{x}_{j+1} \in \mathcal{F}) \int_0^1 p_{R_{j+1}}(r_{j+1}) dr_{j+1} \left\{ \begin{aligned} &H(r_{j+1} \leq \rho^{\mathcal{F}}) \int_{2\pi} p_{\Omega_{j+1}}^{\mathcal{F}}(\omega_{j+1}) d\omega_{j+1} \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} + \\ &H(r_{j+1} > \rho^{\mathcal{F}}) \hat{w}_{j+1} \end{aligned} \right\} + \\ &H(\mathbf{x}_{j+1} \in \mathcal{V}) \int_{4\pi} p_{\Omega_{j+1}, v}^{\mathcal{V}}(\omega_{j+1} | \omega_j) d\omega_{j+1} \int_0^\infty p_{\kappa_{j+1}}(\kappa_{j+1}) d\kappa_{j+1} \mathcal{A}_{j+1} \end{aligned} \right\} \quad (92)$$

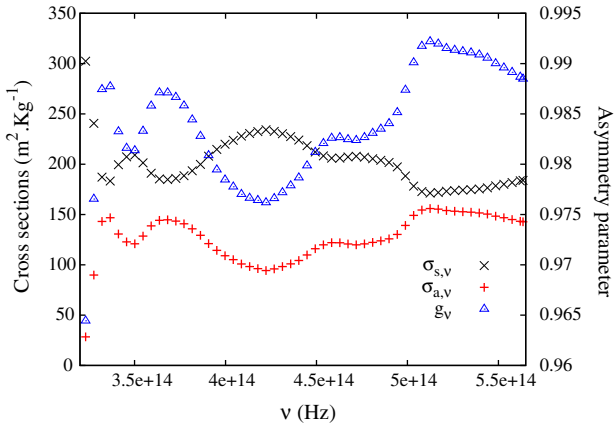


Fig. 14. Spectral radiative properties of *Spirulina* over the photo-synthetically active radiation domain. Results obtained running a Mie code (Mie, 1908; van de Hulst, 1957) for available data on the *Spirulina* spectral complex refractive index and considering a volume equivalent spheres model with a radius log-normal distribution $p_A(a) = \frac{1}{\sqrt{2\pi} \ln s} \exp\left[-\frac{(\ln a - \ln \bar{a})^2}{2 \ln^2 s}\right]$ of mean radius $\bar{a} = 15.52 \mu\text{m}$ and deviation $s = 1.11$. Cross sections $\sigma_{a,v}$ and $\sigma_{s,v}$ are expressed per unit mass of dry biomass by dividing the results of the Mie code ($\text{m}^2 \cdot \text{particle}^{-1}$) by the effective mass of *Spirulina*: $\mathcal{M} = \bar{\rho} \frac{4}{3} \pi \{\bar{a} \exp[\frac{5}{2} \ln^2 s]\}^3$, where $\bar{\rho} = 280 \text{ Kg m}^{-3}$ is the dry biomass density inside a microorganism. For data management convenience, information on the phase function is reduced to its asymmetry parameter $g_v = \int_{4\pi} p_{\Omega,v}(\omega|\omega') \omega \cdot \omega' d\omega$.

where $p_v(v) = \frac{1}{v_{\max} - v_{\min}}$, $p_{\Omega_0}^v = \frac{1}{4\pi}$, $p_{\Omega_j,v}^v$ for $j > 0$ is an Henyey–Greenstein function of asymmetry parameter g_v : $p_{\Omega_{j+1},v}^v(\omega_{j+1}|\omega_j) = \frac{1}{4\pi} \frac{1 - g_v^2}{(1 - 2g_v\omega_{j+1}\omega_j + g_v^2)^{3/2}}$, $p_{\Omega_j}^{\mathcal{R}}(\omega_j) = p_{\Omega_j}^{\mathcal{F}}(\omega_j) = \frac{\omega_j \cdot \mathbf{n}_j}{\pi}$, $p_{R_j} = 1$, $p_{\kappa_j}(\kappa_j) = \exp(-\kappa_j)$, $\mathbf{x}_{j+1} \equiv \mathbf{x}_{j+1}(\mathbf{x}_j, \omega_j, \kappa_j)$ as defined in Sections 2.3 and 4.3, and finally:

$$\hat{w}_{j+1} = \hat{w}_j (H(\mathbf{x}_{j+1} \in \mathcal{R}) \times \rho^{\mathcal{R}} + H(\mathbf{x}_{j+1} \notin \mathcal{R}) \times 1) \exp(-k_{a,v} \|\mathbf{x}_{j+1} - \mathbf{x}_j\|) \quad (93)$$

with

$$\hat{w}_0 = \frac{\varphi_v \sigma_{a,v}}{\pi (1 - \rho^{\mathcal{F}})} \frac{1}{p_v(v) p_{\Omega_0}^v}$$

On the basis of Sections 2.3 and 4.3 (Eq. (89) with uniform density η even outside the reactor and $h = \eta$), the integral formulation corresponding to the sensitivity $\partial_{\eta} A(\mathbf{x}_0)$ is identical to that of $A(\mathbf{x}_0)$ in Eqs. (91) and (92), replacing only \hat{w}_j with $\hat{w}_{\eta,j}$:

$$\hat{w}_{\eta,j} = \hat{w}_j \left[-\sigma_{a,v} d_j + \sum_{q=0}^{j-1} \frac{1 - \kappa_q}{\eta} \right] \quad (94)$$

The Monte Carlo algorithm depends on the *Spirulina* size distribution through radiative properties $\sigma_{a,v}$, $\sigma_{s,v}$ and g_v , that are computed using a Mie code with a given size distribution as an input. Thus, sensitivities to a parameter π of the *Spirulina* size distribution (π being either \bar{a} or s) is written as:

$$\partial_{\pi} A = \int_{v_{\min}}^{v_{\max}} \left[\frac{\partial A_v}{\partial \sigma_{a,v}} \frac{\partial \sigma_{a,v}}{\partial \pi} + \frac{\partial A_v}{\partial \sigma_{s,v}} \frac{\partial \sigma_{s,v}}{\partial \pi} + \frac{\partial A_v}{\partial g_v} \frac{\partial g_v}{\partial \pi} \right] dv \quad (95)$$

On the basis of Section 2.3 (cases a and b), the integral formulation corresponding to the terms $\int_{v_{\min}}^{v_{\max}} \frac{\partial A_v}{\partial \sigma_{a,v}} dv$, $\int_{v_{\min}}^{v_{\max}} \frac{\partial A_v}{\partial \sigma_{s,v}} dv$ and $\int_{v_{\min}}^{v_{\max}} \frac{\partial A_v}{\partial g_v} dv$ are identical to that of $A(\mathbf{x}_0)$ in Eqs. (91) and (92), replacing only \hat{w}_j respectively with $\hat{w}_{\sigma_{a,v},j}$, $\hat{w}_{\sigma_{s,v},j}$ and $\hat{w}_{g_v,j}$:

$$\hat{w}_{\sigma_{a,v},j} = \hat{w}_j \left(\frac{1}{\sigma_{a,v}} - \eta d_j \right) \quad (96)$$

$$\hat{w}_{\sigma_{s,v},j} = \hat{w}_j \sum_{q=0}^{j-1} \frac{1 - \kappa_q}{\sigma_{s,v}} \quad (97)$$

$$\hat{w}_{g_v,j} = \hat{w}_j \sum_{q=0}^{j-1} \left[H(\mathbf{x}_j \in \mathcal{V}) \times \left(-\frac{2g_v}{1 - g_v^2} - 3 \frac{g_v - \omega_q \cdot \omega_{q-1}}{1 - 2g_v \omega_q \cdot \omega_{q-1} + g_v^2} \right) \right] \quad (98)$$

with d_j the total length of the multi-scattering-and-reflection optical path, as defined in Section 2.1. Thus the integral formulation corresponding to the sensitivities $\partial_{\pi} A(\mathbf{x}_0)$ are identical to that of $A(\mathbf{x}_0)$ in Eqs. (91) and (92), replacing only \hat{w}_j respectively with $\hat{w}_{\pi,j}$:

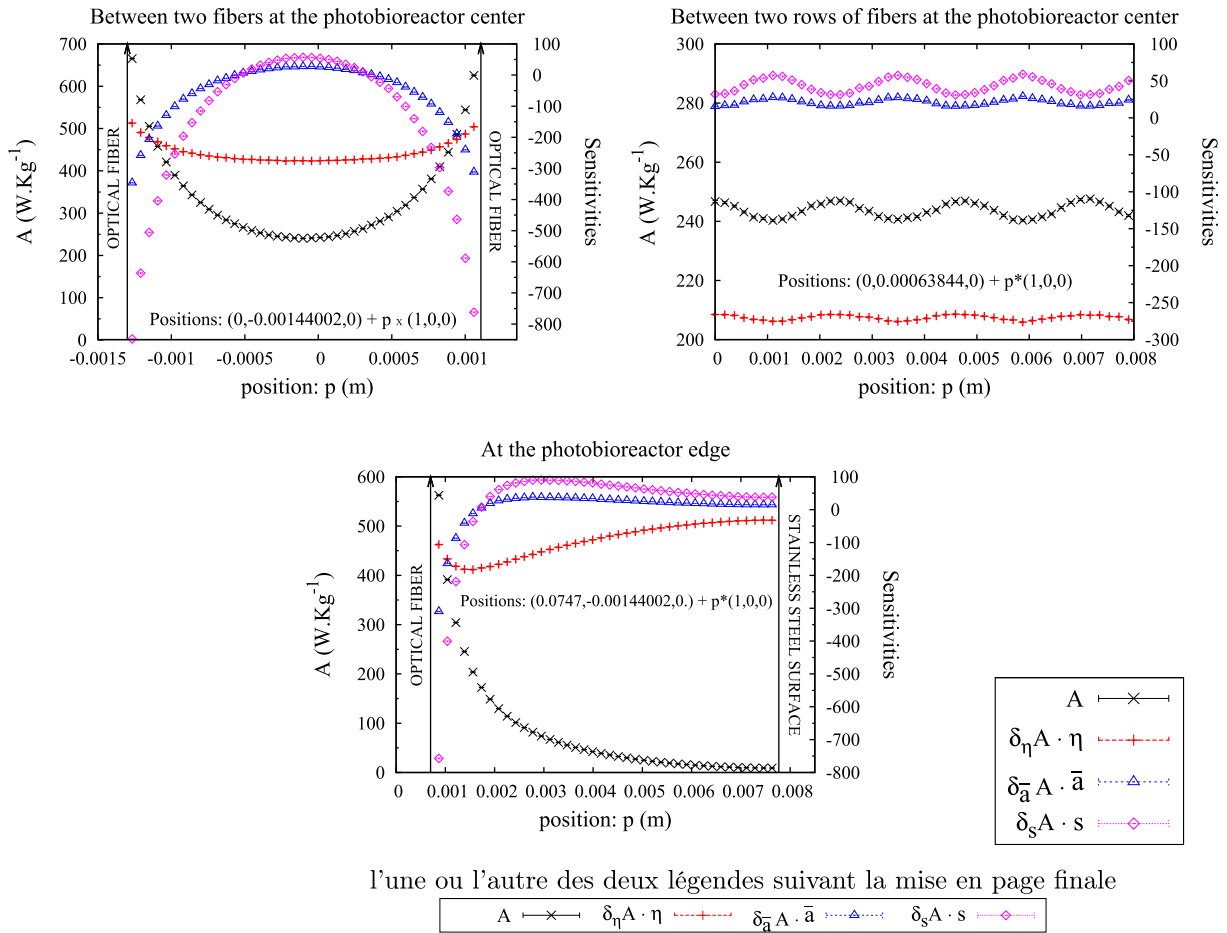
$$\hat{w}_{\pi,j} = \hat{w}_{\sigma_{a,v},j} \frac{\partial \sigma_{a,v}}{\partial \pi} + \hat{w}_{\sigma_{s,v},j} \frac{\partial \sigma_{s,v}}{\partial \pi} + \hat{w}_{g_v,j} \frac{\partial g_v}{\partial \pi} \quad (99)$$

where the spectra $\frac{\partial \sigma_{a,v}}{\partial \pi}$, $\frac{\partial \sigma_{s,v}}{\partial \pi}$ and $\frac{\partial g_v}{\partial \pi}$ are numerically evaluated by running the Mie codes for a 1% variation of π around its value and taking into account that the effective mass of *Spirulina* also depends on π , whether π is \bar{a} or s (see Fig. 14 caption).

Fig. 15 displays the simulation results corresponding to A , $\delta_{\eta} A$, $\delta_{\bar{a}} A$ and $\delta_s A$ in function of the position inside the reactor.

5. Open questions

The above presented methodological advances, implementation techniques and application examples were selected to support the argument that concentrated solar engineers and physicists can easily enter the most recent Monte Carlo literature and efficiently translate the exposed material for application to their research objects. We also tried to illustrate that numerous programming libraries are now available that reduce very considerably the time required for the implementation of any fully up to date Monte Carlo code including the type of physical and geometrical complexity that is typical of concentrated solar research. Doing so we did not highlight the standard convergence difficulties encountered by developers entering the field of Monte Carlo methods for the first time, in particular those associated with geometrical difficulties. As was illustrated with the shape factor examples, choosing one integral formulation or the other may mean that convergence is fast or unreachable. In the Monte Carlo literature, these types of geometrical alternatives are named forward (or analog) algorithms, backward algorithms,



l'une ou l'autre des deux légendes suivant la mise en page finale

Fig. 15. Simulation results corresponding to the field of radiant power absorbed $A(\mathbf{x}_0)$ and its sensitivities $\delta_{\eta}A$, $\delta_{\bar{a}}A$ and δ_sA as defined in Eqs. (91) and (92). Sensitivities are expressed in terms of a relative variation of the *Spirulina* density η and the parameters \bar{a} and s of the *Spirulina* size distribution (see plot key). Results obtained for the *Spirulina* spectral properties Fig. 14, $\varphi_v = 10^{-7} \frac{2\pi h \nu^3}{c^2} \frac{1}{\exp\left(\frac{h\nu}{k_B 6000} - 1\right)}$ (proportional to the Planck's law at 6000 K),

$\rho^{\mathcal{F}} = 0.1$, $\rho^{\mathcal{R}} = 0.5$, $\eta = 4 \text{ kg m}^{-3}$ of dry biomass and 10^6 realizations. For each position, the calculation (including the sensitivities) takes roughly 1 min on one processor and gives a relative standard deviation around 0.1%.

symmetrical (or reciprocal) algorithms, etc. They are clearly described in the above mentioned textbooks and should be quite easily and rapidly acquired. This means that the only strong convergence difficulties that concentrated solar specialists are expected to encounter are those involving more than pure geometrical considerations, in particular those involving spectral properties or optically thick semi-transparent material (as for the bottom part of the above described fluidized bed receiver).

But the fact that very few convergence difficulties are encountered does not mean that convergence enhancement is not required. This only means that even the most complex concentrated solar systems will require less than a few hours of computation time (more commonly a few minutes) for a 1% accuracy to be reached using modern parallel computers. But the question of the very much shorter computation times required for iterative optimized procedures remains widely open. Today, the zero-variance concept is the best reported theoretical approach to this question. This statement is made thinking of researchers

also willing to gain a deeper physical understanding of the leading mechanisms at work in the considered system. Without this complementary objective, brute force automatic variance reduction techniques could be appreciated as playing a similar role, but the literature still points out implementation difficulties when increasing the dimension of the integration domain, and the reported strongest variance reductions still rely on subtle physical pictures of the source of variance, i.e. on such pictures the zero-variance concept helps clarifying. However, we have shown that starting from a given integral formulation, the zero-variance algorithm is associated to an ideally optimized probability density function for each of the sampled variables, and that this is only useful, in practice, if an approximate model can be used to derive a probability density function that is both close enough to the ideal one and simple enough for efficient sampling procedures to be set up. Again, choosing one integral formulation or another may define entirely different modeling questions, some easy to address (as in the spectral example of Section 2.2) and some

others untractable. In summary, when the objective is to accelerate Monte Carlo algorithms to make them compatible with iterative design procedures, the only available systematic approach is to write down the ideal probability density function for each of the most critical sampled variables and to think of simple approximate models. If such models are missing, attempts can be made using simple integral reformulations, or the difficulty can be translated into a theoretical question to radiative transfer specialists. From this point of view the practice of simulating last generation concentrated solar systems (in particular solar receivers) is likely to open specific fields of investigation related to complex multiple reflection, multiple scattering optical path statistics. It is also worth reminding at this stage that an essential part of the zero-variance methodology has been left aside: the use of adjoint transport formulations. A full understanding of these formulations requires the help of quite advanced mathematics, but it may very well be that the most significant forthcoming advances are to be searched along this line.

Sensitivity evaluation is the second theoretical framework that was highlighted in the above presentation. As far as concentrated solar engineering is concerned, the main idea is the following: Monte Carlo methods are sometimes computationally expensive, but they can provide additional information concerning the influence of free design parameters. This is obviously a helpful source of orientation for engineers attempting to make argued choices for these parameters. But this can also be useful at the heart of automatic design optimization procedures that include the evaluation of Jacobian matrices (first order sensitivity matrices). Very much like the expression of ideally optimized probability density functions, the expression of sensitivity weights is commonly quite straightforward and can easily be repeated for each of the sensitivities to be evaluated simultaneously (i.e. for each of the Jacobian matrix elements). However, this is only practically meaningful if these weights can be computed without too strong a modification of the initial algorithm, and if the relative variances of the sensitivity weights are similar to that of the main quantity. There are essentially two reasons why the developer may observe that the initial algorithm needs to be strongly modified to include sensitivity evaluations. The first reason is that sensitivity weights may involve geometrical quantities that are not commonly included in the geometric libraries (such as the one used in EDStar). The second reason is that domain deformation sensitivities require the evaluation of vector quantities potentially of the size of the integration domain, that is to say of infinite dimension for multiple reflection, multiple scattering systems. This may therefore strongly impact the computation sequence corresponding to optical path tracking and may require subtle memory managements. These two difficulties are quite technical and may be easily bypassed by computer science specialists, but we want to mention them as open questions in the

present context as they moderate our statement that today's programming libraries solve most of our practical requirements. But the stronger remaining open question related to sensitivity evaluation clearly belongs to radiative transfer research: how can sensitivity variances be reduced when sensitivity convergence is observed to be slower (when increasing the sample size) than that of the main quantity? In many cases this difficulty is not encountered. But when encountering it, then the developer will observe that no systematic procedure is available to address it. Furthermore, it is extremely difficult to guess whether a sensitivity will easily converge or not. Consequently, as for the zero-variance methodology, the only solution is to implement the sensitivity weight computation (which will require little investment in a wide majority of cases) and if convergence difficulties are encountered, they should be first addressed using simple integral reformulations, and then forwarded to radiative transfer specialists that will try to physically understand the underlying reasons why the variance is high and think of adapted control variate (as this is the only mentioned practical solution at the present stage).

References

- Agostinelli, S. et al., 2003. Geant4 – a simulation toolkit. *Nuclear Instruments and Methods in Physics Research A* 506, 250–303.
- Allison, J. et al., 2006. Geant4 developments and applications. *IEEE Transactions on Nuclear Science* 53 (1), 270–278.
- Anderson, H.L., 1986. Metropolis, Monte Carlo and the MANIAC. *Los Alamos Science* 14, 96–108.
- Assaraf, R., Caffarel, M., 1999. Zero-variance principle for Monte Carlo algorithms. *Physical Review Letters* 83 (23), 4682–4685.
- Azmy, Y., Sartor, E., 2010. *Nuclear Computational Science*. Springer (ISBN: 9048134102).
- Berger, J.M., 1955. Reflection and transmission of gamma radiation by barriers: Monte-Carlo calculation by a collision-density method. *Journal of Research of the National Bureau of Standards* 55, 343–350.
- Berger, J.M., Doggett, J., 1956. Reflection and transmission of gamma radiation by barriers: semianalytic Monte-Carlo calculation. *Journal of Research of the National Bureau of Standards* 56, 89–98.
- Blanco, S., Fournier, R., 2003. An invariance property of diffusive random walks. *Europhysics Letters* 61 (2), 168.
- Blanco, S., Fournier, R., 2006. Short-Path Statistics and the Diffusion Approximation. *Physical Review Letters* 97 (23), 230604.
- Boltzmann, L., 1872. Weitere Studien über das Wärmegleichgewicht unter Gasmolekülen. *Sitzungsberichte der Akademie der Wissenschaften. Wien II* 66, 275–370.
- Cacuci, Dan G., 2003. In: *Sensitivity and Uncertainty Analysis: Theory*, vol. I. Chapman and Hall.
- Cherkaoui, M., Dufresne, J.L., Fournier, R., Grandpeix, J.Y., Lahellec, A., 1996. Monte Carlo simulation of radiation in gases with a narrow-band model and a net-exchange formulation. *Journal of Heat Transfer* 118 (2), 401–407.
- Cornet, J.F., 2010. Calculation of optimal design and ideal productivities of volumetrically lightened photobioreactors using the constructal approach. *Chemical Engineering Science* 65 (2), 985–998.
- Cornet, J.F., Dussap, C.-G., 2009. A simple and reliable formula for assessment of maximum volumetric productivities in photobioreactors. *Biotechnology Progress* 25 (2), 424–435.
- Dauchet, J., Blanco, S., Cornet, J.-F., El Hafi, M., Fournier, R., 2011. Multiple-scattering radiative transfer study for solar photobioreactors

- optimization. In: Proceedings of the SolarPACES International Symposium on Solar Thermal Concentrating Technologies.
- de Lataillade, A., Blanco, S., Clergent, Y., Dufresne, J.L., El Hafi, M., Fournier, R., 2002. Monte Carlo method and sensitivity estimations. *JQSRT* 75, 529–538.
- Dersch, J., Morin, G., Eck, M. and Häberle, A., 2009. Comparison of Linear Fresnel and Parabolic Through Collector Systems – System Analysis to Determines Break Even Costs of Linear Fresnel Collectors, Berlin, pp 8.
- Doolen, G.D., Hendricks, J., 1987. Monte-Carlo at work Los Alamos Science No 15.
- Dutre, P., Bala, K., Bekaert, P., 2002. Advanced Global Illumination. A.K. Peters, Ltd., Natick, MA, USA (ISBN: 1568811772).
- EDStar (STARWest development environment) <http://www.starwest.ups-tlse.fr/edstar/edstar.html>, including the codes of the four application examples at <http://www.starwest.ups-tlse.fr/monte-carlo-concentrated-solar-examples/>.
- Flamant, G., Menigault, T., 1986. Combined wall-to-fluidized bed heat transfer. Bubbles and emulsion contributions at high temperature. *International Journal of the Heat Mass Transfer* 30 (9), 1803–1812.
- Glouchkov, D., Koshelev, K., Schulz, A., 2003. Monte Carlo simulation of photon transport for optically thick, differentially moving plasmas: II. Escape factors for differentially moving spheres. *JQSRT* 81, 191–197.
- GSL – GNU Scientific Library. <<http://www.gnu.org/software/gsl>>.
- Haddad, I.M., Elsayed, M.M., 1988. Transient performance fluidized bed solar receiver at various parametric conditions. *Solar and Wind Technology Journal* 5 (6), 653–659.
- Hammersley, J.M., Handscomb, D.C., 1964. Monte Carlo Methods. Chapman and Hall, London.
- Hoogenboom, J.E., 2008. Zero-variance Monte Carlo schemes revisited. *Nuclear Science and Engineering* 160 (1), 1–22.
- Howell, J.R., 1998. The Monte Carlo method in radiative heat transfer. *Journal of Heat Transfer-Transactions of the Asme* 120 (3), 547–560.
- James, F., 1994. RANLUX: a Fortran implementation of the high-quality pseudo-random number generator of Lüscher. *Computer Physics Communications* 79, 111–114.
- Kalos, M.H., Whitlock, P.A., 1986. Monte Carlo Methods. John Wiley & Sons.
- Lux, I., Koblinger, L., 1991. Monte Carlo particle transport methods: neutron and photon calculations. CRC Press, Boca Raton.
- Metropolis, N., Ulam, S., 1949. The monte carlo method. *Journal of the American Statistical Association* 44 (247), 335–341.
- Mie, G., 1908. Beitrage zur optik truber medien speziell kolloidallen metallosungen. *Annalen Der Physik* 25, 377–445.
- Muller, R., Zedtwitz, P.V., Wokaun, A., Steinfeld, A., 2003. Kinetic investigation on steam gasification of charcoal under direct high-flux irradiation. *Chemical Engineering Science* 58, 5111–5119.
- Pharr, M., Humphreys, G., 2004. Physically Based Rendering: From Theory to Implementation. Elsevier.
- Roccia, J.P. et al., 2012. SOLFAST a ray-tracing Monte carlo software for solar concentrating facilities. *Journal of Physics conference series* 369, 012029.
- Roger, M., Blanco, S., El Hafi, M., Fournier, R., 2005. Monte Carlo estimates of domain-deformation sensitivities. *Physical Review Letter* 95 (18).
- Saltelli, A., Tarantola, S., Campolongo, F., Ratto, M., 2004. Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models. John Wiley and Sons.
- Spanier, J., Gelbard, E.M., 2008. Monte Carlo Principles and Neutron Transport Problems. Dover (ISBN: 0486462935).
- STARWest (STATistiques Radiatives du Sud-Ouest). <http://www.starwest.ups-tlse.fr/>.
- The Message Passing Interface (MPI) standard. <<http://www.mcs.anl.gov/research/projects/mpi>>.
- Trommer, D. et al., 2005. Hydrogen production by steam-gasification of petroleum coke using concentrated solar power: thermodynamic and kinetic analyzes. *International Journal of Hydrogen Energy* 30, 605–618.
- van de Hulst, H.C., 1957. Light Scattering by Small Particles. John Wiley and Sons, New York.
- Veach, E., 1998. Robust Monte Carlo Methods for Light Transport Simulation. Phd Thesis, Stanford University, Stanford, CA, USA.