Vectorized Monte-Carlo method for sensitivity models in radiative transfer : Application to Concentrated Solar Power

Zili HE

Université Fédérale

Toulouse Midi-Pyrénées

THÈSE

en vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

délivré par

IMT – École Nationale Supérieure des Mines d'Albi-Carmaux

---

présentée et soutenue par

Zili HE

le 7 décembre 2022

Vectorized Monte-Carlo Method for sensitivity models in radiative transfer: Application to Concentrated Solar Power

---

École doctorale et discipline ou spécialité :
MEGEP : Génie des procédés et de l'Environnement

Unité de recherche :
Centre RAPSODEE, UMR CNRS 5302, IMT Mines Albi

Directrice de thèse :
Mouna EL HAFI, Professeure, IMT Mines Albi

Autres membres du jury :
Céline CORNET, Professeure, Université de Lille *(Rapporteure)*
Yann FAVENNEC, Maître de conférences, HDR, Université de Nantes *(Rapporteur)*
José GONZALES-AGUILAR, Senior researcher, Instituto IMDEA Energía, Madrid *(Examinateur)*
Rodolphe VAILLON, Directeur de recherche, IES, Montpellier *(Président)*
Nicolas MELLADO, Chargé de recherche, IRIT, Toulouse *(Examinateur)*
Simon EIBNER, Maître assistant, IMT Mines Albi *(Co-encadrant)*
Jean-Jacques BEZIAN, Professeur retraité, IMT Mines Albi *(Invité)*
Eugene d'EON, Chercheur, NVIDIA, Wellington, Nouvelle-Zélande *(Invité)*

Summary

Acknowledgements

Words cannot express my gratitude to my supervisors, Simon Eibner, Mouna El Hafi, and Jean-Jacques Bezian (who advised me for one year and then retired), for their invaluable patience and feedback. I could not have undertaken this journey without the valuable scientific discussions with Richard Fournier, Stephane Blanco, Paule Lapeyre, and Eugene D'eon, who generally provided knowledge and expertise. Additionally, this endeavor would not have been possible without the generous support from the physicians and computer scientists of MesoStar, especially Vincent Forest, who encouraged me to explore the world of open source. Thanks should also go to my cohort members and study participants from the EdStar group, who impacted and inspired me.

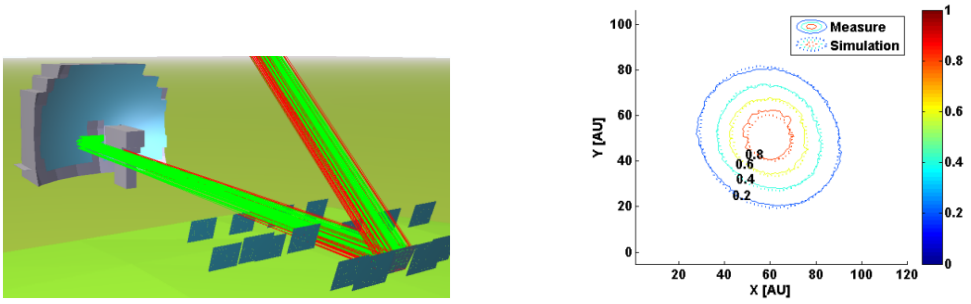
Lastly, I would be remiss in not mentioning my family, especially my parents. Their belief in me has kept my spirits and motivation high during this process. Finally, I would also like to thank Wan-Hsuan Lu, who supported me and brightened my life.

“A man without dreams is no different than salted fish.”

—Stephen Chow

Context

Concentrating Solar Power (CSP) technologies have an undeniably significant potential to produce electricity and heat. This Ph.D. thesis is motivated by issues related to optimization and inverse problems, where the sensitivities of different parameters in a CSP system are needed. These parameters could be, for example, the positions and the properties of reflection of heliostats. In a recent study [5], Caliot *et al.* inversely calculate the reflective property of a heliostat from a measured flux map on the receiver of a CSP facility. In order to achieve this, the sensitivity of the reflective property to the flux was needed. Moreover, sensitivities to design parameters (such as positions, rotations, and sizes of heliostats) are required to optimize the optical performance of a STP system. However, to the author's knowledge, no methods are available in the CSP community to estimate the sensitivities involving geometric parameters. Therefore, this thesis aims to fill this gap.



(a) Ray-tracing for one heliostat **(b)** Flux map

Figure 1 – Based on the measured flux map and the sensitivity of the rms-slope parameters, which characterizes the error of reflection on a heliostat, researchers can determine the real rms-slope parameters of the heliostat by inverse methods [5].

Before focusing on the sensitivities, the evaluation of optical performance in CSP systems is mainly based on the Monte Carlo method [62]. This specific numerical tool has the advantage of dealing with the complexity of the studied geometry, the coupling of different physical phenomena, or the large timescale [6, 15, 24, 57]. The complexity of the studied geometry means that the geometric scales for different elements in a radiative system are very different. For example, studies have been done on thermal analysis for a teapot in a city (Fig.2). The geometric scales of elements are very different, from a large modern building to a tree near the road and a small window on the wall of a garage in a city. Villefranque *et al.* studied the thermal condition of this teapot [58]. Thermal radiation among all the elements, even the thermal diffusion in a thermally isolated wall, and the transmittance of double-glazed

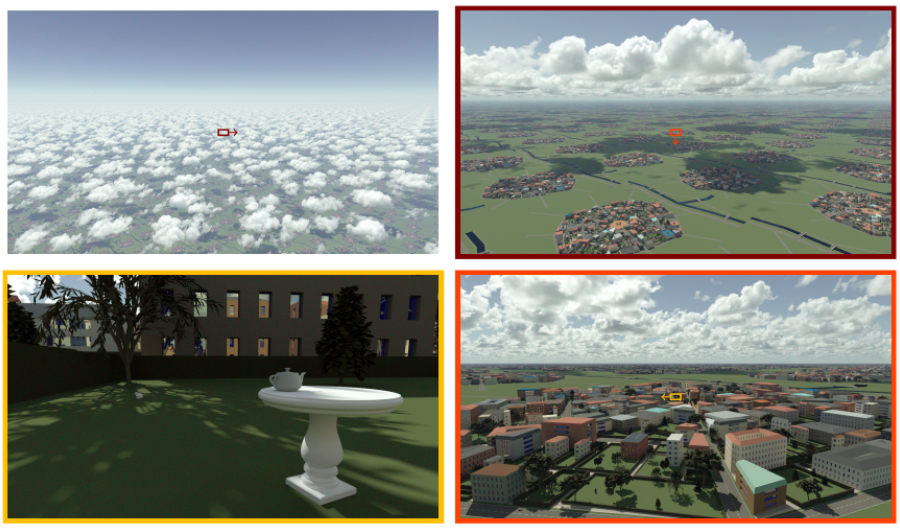


Figure 2 – The thermal condition of a teapot on a table, in a city, and under the cumulus clouds can be simulated by the developed Monte Carlo method [58]

windows must be considered. Such a Monte Carlo method makes thermal transfer modeling with geometric complexity possible. The geometry can also be very complex in a CSP system. For example, Ivanpah solar power facility is the largest solar facility in the world in 2022 and is composed of 173500 heliostats (Fig.3). Each heliostat follows the sun position during the day and can shade other heliostats. The Monte Carlo method is well adapted to deal with this geometric complexity.

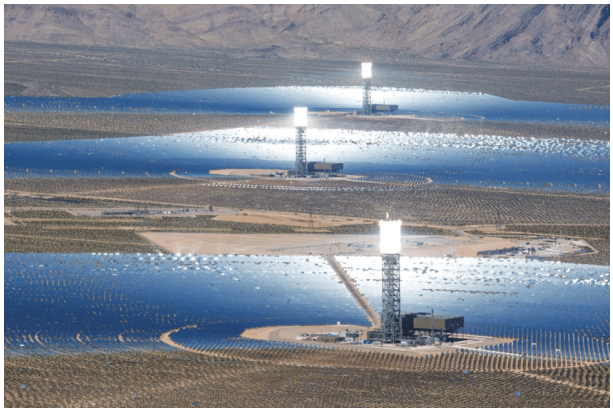


Figure 3 – The Ivanpah Solar Electric Generating System [41]

The work in this thesis is developed within a research group (RAPSODEE[1], LAPLACE[2], Mesostar[3] and EDstar group [4]) where the principal activities are about the numerical simulation of radiative transfer by Monte Carlo method. Estimating sensitivity by the Monte Carlo method in CSP systems remains an open research problem, and only very few studies have been done [9, 15, 31]. Physicians in the group have been working on estimating sensitivities by the Monte Carlo method since 2002. De Lataillade studied the heating ratio

[1]Recherche d'Albi en génie des Procédés, des SOlides Divisés, de l'Énergie et de l'Environnement
[2]Le Laboratoire Plasma et Conversion d'Energie
[3]https://www.meso-star.com/
[4]http://www.edstar.cnrs.fr/prod/fr/

sensitivity to the species concentration in a CH_4 flame [10]. However, the estimation of sensitivities involving geometric parameters still needed to be possible. In 2006, Roger *et al.* developed a Monte Carlo method to estimate the sensitivities geometric parameters [50, 51]. This method is noted as the "method of observable derivation" in this thesis. The physical meaning of this name will be discussed in Chapter.2. After that, De la Torre attempted to apply the method of observable derivation to estimate the geometric sensitivities in CSP systems in his thesis in 2011 [9]. Unfortunately, the author did not consider the blocking and shadowing effects. It is because applying the method of observable derivation for cases involving complex geometry requires a great effort of formal development. The geometry of a CSP system with heliostats is too complex to take all blocking and shadowing into account, which limits the application of this method in CSP system. In 2021, a general transport model of geometric sensitivity was proposed by Lapeyre *et al.*, which is also dedicated to estimating sensitivities in radiative systems. This method is noted "method of sensitivity model" in this thesis. The physical meaning of the name will be discussed in Chapter.2. Unlike the method of observable derivation, less formal development is required for the method of sensitivity model to be applied in cases involving complex geometries. This method has the potential to be applied to CSP applications.

However, the method of sensitivity model has a weakness compared to that of observable derivation. For the method of observable derivation, when an observable is evaluated, its sensitivity to any parameter is also evaluated at the expense of a low supplementary computation effort. This advantage is because the Monte Carlo method estimates a vector containing the observable and all its sensitivities to different parameters. Therefore, the estimation of observable and all its sensitivities are completely "vectorized". This method will be detailed in Chapter.3. For the method of sensitivity model, it loses this advantage of complete vectorization. Users must build a model for each sensitivity corresponding to each parameter. Therefore, at least one Monte Carlo estimation is needed for each model. Nevertheless, when applying the method in cases involving complex geometries, sources in a sensitivity model can be found on the surface or on the segments, which makes the model difficult to be solved by the Monte Carlo method. Unlike the method of observable derivation, the sensitivity model performs Monte Carlo estimations for scalars. At least one Monte Carlo estimation is needed for each source to solve such a sensitivity model. This inconvenience makes the coding development heavier and the required calculating time very consuming. Here comes an interesting problem to solve, which is to regain the advantage of vectorization for the method of sensitivity model. The advantage of complete vectorization is hard to be regained following the method of sensitivity model, but a strategy of partial vectorization is provided in Chapter.4. Following this strategy, a method of estimating sensitivities for a STP system is developed in Chapter.5. Finally, in Chapter.6, we attempt to achieve complete vectorization following the method of sensitivity model by transforming all line sources in the system to surface sources. Some academic cases are developed, and some limitations will be discussed.

Lastly, efforts are also made to understand and compare the research in the image synthesis community for similar radiative transfer problems about sensitivities. Rendering images of high quality is one of the main challenges in image synthesis. It is about producing a 2D image from a description of a 3D scene. There are many ways to render images, but physically based rendering is one of the most realistic. It is because the physically based rendering follows the principles of physics to model the interaction of light and matter (by ray-tracing) to simulate reality [46]. Ray tracing is a straightforward algorithm based on following the path of a ray of light through a scene as it interacts with and bounces off objects in an environment. Nowadays, almost all the images of synthesis in the cinema industry are done by the Monte Carlo method and ray-tracing algorithms. Thanks to the numerical tools and libraries [46] developed in the field of image synthesis, the implementation of the Monte Carlo method in this thesis becomes possible. Furthermore, researchers in image synthesis also work in the estimation of sensitivities. The corresponding method is developed

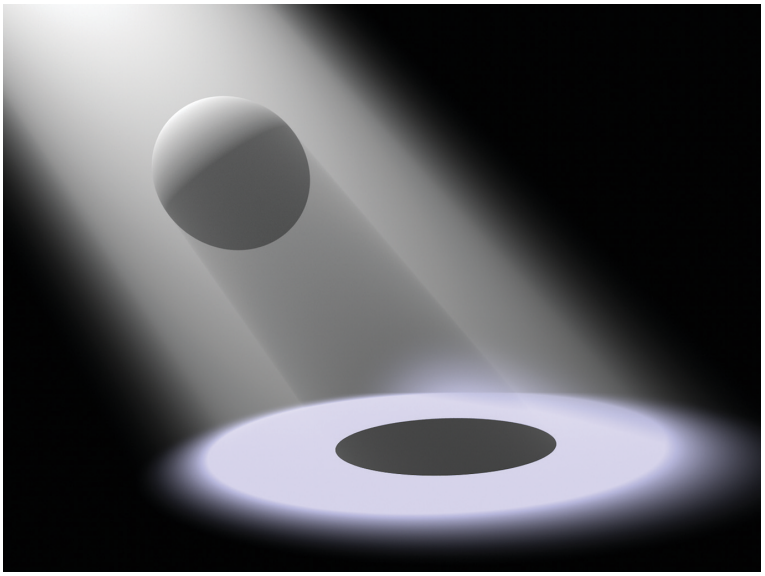


Figure 4 – A spotlight is shining on a sphere through the frog, where the spotlight's lighting distribution and the sphere's shadow are visible due to the additional scattering in the semi-transparent medium [46].

and studied in Chapter.2 and Chapter.3 and is called differentiable rendering [70]. This method has already been used in many applications involving complex geometries. Coupled with the optimization algorithms and neutral network [26], the sensitivities are used for the reconstruction of 3D objects [27, 28, 56, 65], the body shapes [4, 45], hand shapes [1, 72], face shapes [20], etc. However, although we share similar research problems in estimating sensitivities, the viewpoints and vocabularies between researchers in image synthesis and our group have significant differences. The differential rendering method mainly focuses on the mathematical solution and the algorithms. Less attention has been paid to interpreting the sensitivities physically. This method is compared with the other two methods mentioned in the last paragraph in Chapter.3, where we explain, analyze and compare the three approaches. It is found that sensitivity analysis is hard to perform following the method of differentiable rendering. However, sensitivity analysis is performed following the method of sensitivity model in Chapter.5 for a CSP application. We will see in this thesis that the two communities have many connections in the research problems of sensitivity and can benefit from the research of each other.

Finally, this thesis is organized into six chapters:

- Chapter.1 aims to prepare a theoretical basis for readers unfamiliar with the physics of radiation or the method of Monte Carlo. We will review the integral and statistical radiation approaches and introduce the Monte Carlo method.
- Chapter.2 introduces the objective and the research problem of this thesis. We firstly review the different CSP optical systems, the standard numerical tools for evaluating the optical performance, and the needs of estimating sensitivities in CSP system. This thesis aims to develop a method to estimate sensitivities in CSP system. To achieve this objective, we review the three methods in the community of radiative transfer and image synthesis: the method of observable derivation, the method of differentiable rendering, and the method of sensitivity model. The advantages and disadvantages are listed for each method. After that, we define the research problem of this thesis: regain the advantage of vectorization for the method of sensitivity model.

- Chapter.3 compares and reviews the three viewpoints for the estimations of sensitivities in a radiative system by applying them to a simple one-dimension radiative problem. The three viewpoints consist of the method of observable derivation developed in the thesis of Maxime Roger [51], the method of differentiable rendering used in image synthesis community [70], and the method of sensitivity model developed in the thesis of Paule Lapeyre [32]. The theoretical link between the three methods will be distinguished, and each method's advantages and conveniences will be discussed.

- Chapter.4 aims to apply the method of sensitivity model in cases involving complex geometry. To achieve this, we will introduce a strategy of vectorization to deal with a large number of sources in the models of sensitivities. This chapter will prepare the more complex application in a STP system for the following chapter.

- In Chapter.5, we apply the method of sensitivity model in a STP system, following the strategy of vectorization presented in the last chapter. Also, thanks to the models of sensitivities, the physical interpretation of the sensitivities can be realized. The contributions of different physical events in a STP system (blocking, shadowing, spillage, etc.) to the sensitivities can be analyzed in detail, helping engineers to understand and optimize the system.

- Following the method of sensitivity model, Chapter.6 aims to regain the advantage of the method of observable derivation: a complete vectorization. It will be done by transforming the line sources to surface sources in the sensitivity model. Some academic examples are developed, and some limitations are discussed.

1

Monte-Carlo method and radiative transfer

1.1 Introduction

This chapter introduces the physical basis of radiative transfer and the Monte Carlo method, a numerical tool to simulate the physics of radiative transfer. An example demonstrates how to solve radiative transfer problems using the Monte Carlo method. Through this example, we introduce essential physical concepts used in this thesis, such as “path spaces”, “forward and reverse ray-tracing”, “vectorization”, etc. Last but not least, Monte Carlo algorithms are attached to a physical image since they are deeply linked with the physics of radiative transfer. Therefore, at the end of this chapter, we discuss the relation between the Monte Carlo algorithm, the proposed physical image, and the corresponding mathematical formulation.

1.2 Review of statistics

It is necessary to review the basis of statistical concepts before introducing the Monte Carlo method. The following parts of this thesis, the *random variable*s will be denoted by capital letters. In contrast, the minuscule letters will be used to denote their possible values, which is also called a *realization* or a *sampling* of the *random variable*. For example, c is noted as a *realization* of *random variable* C.

The value of a *random variable* is not predictable, and it can only be determined by an experiment. Furthermore, the *random variable* is classified by two families: discrete *random variable*s and continuous *random variable*s.

Definition

Letting A be a discrete *random variable*, it can be defined by:

- a set of its all possible values $A \in \{a_1, a_2, a_3, \ldots, a_n\}$ and
- the probabilities $\mathscr{P}_A(a_i)$ that A takes the value of a_i, $\forall i \in \{1, 2, \ldots, n\}$.

The sum of the probabilities has to be normalized:

$$\sum_{i=1}^{n} \mathscr{P}_A(a_i) = 1 \tag{1.1}$$

Letting B be a continuous *random variable*, it can be defined by:

- the continuous domain of its all possible values $B \in [b_{min}, b_{max}]$ and
- the pdf (Probability density function) $\mathscr{P}_B(b)$.

The integral of $\mathscr{P}_B(b)$ in the domain of $B \in [b_{min}, b_{max}]$ has to be normalised:

$$\int_{b_{min}}^{b_{max}} \mathscr{P}_B(b) = 1 \tag{1.2}$$

Besides these two definitions, a *random variable* can be defined by applying a function to another *random variable*. For example, if C is defined as a function of a *random variable* B:

$$C = f(B), \tag{1.3}$$

C is also itself a *random variable*.

Expected Value

The expected value corresponds to the first moment of the *random variable*. For the discrete *random variable* A, it is defined mathematically by:

$$\mathbb{E}[A] = \sum_{i=1}^{n} a_i \mathscr{P}_A(a_i). \tag{1.4}$$

For the continuous *random variable* B:

$$\mathbb{E}[B] = \int_{b_{min}}^{b_{max}} b\mathscr{P}_B(b)db. \tag{1.5}$$

For the *random variable* C which is defined by Eq.1.3, its expected value can be written as:

$$\boxed{\mathbb{E}[C] = \mathbb{E}[f(B)] = \int_{b_{min}}^{b_{max}} f(b)\mathscr{P}_B(b)db} \tag{1.6}$$

It is a very important property because Eq.1.6 means that the estimation of the expected value of C (or $f(B)$) needs only the knowledge of the pdf of B, but not of $f(B)$.

Furthermore, if $B_1, B_2, \ldots, B_n$ are some independent continuous *random variable*s, with $b1 \in [b_{1min}, b_{1max}], \ldots, b_n \in [b_{nmin}, b_{nmax}]$, the value of the function $g(b_1, b_2, \ldots, b_n)$ will be also a *random variable* and its expected value can be written as:

$$\mathbb{E}\big[g(B_1, \ldots, B_n)\big] = \int_{b_{1min}}^{b_{1max}} \mathscr{P}_{B1}(b_1)db_1 \cdots \int_{b_{nmin}}^{b_{nmax}} \mathscr{P}_{Bn}(b_n)db_n g(b_1, \ldots, b_n) \tag{1.7}$$

Variance and standard deviation

The variance of a *random variable* can describe the distribution of its possible values, also called its second central moment.

The following equations show the mathematical definitions of the variance of the *random variable* A and B.

$$\begin{aligned} \mathbb{V}[A] &= \sum_{i=1}^{n}(a_i - \mathbb{E}[A])^2\mathscr{P}_A(a_i) \\ &= \mathbb{E}[(A - \mathbb{E}[A])^2] \\ &= \mathbb{E}[A^2 - 2\mathbb{E}[A]A + \mathbb{E}^2[A]] \\ &= \mathbb{E}[(A^2)] - \mathbb{E}^2[A] \end{aligned} \tag{1.8}$$

$$\begin{aligned} \mathbb{V}[B] &= \int_{b_{min}}^{b_{max}} (b - \mathbb{E}[B])^2\mathscr{P}_B(b)db \\ &= \mathbb{E}[(B - \mathbb{E}[B])^2] \\ &= \mathbb{E}[B^2 - 2\mathbb{E}[B]B + \mathbb{E}^2[B]] \\ &= \mathbb{E}[(B^2)] - \mathbb{E}^2[B] \end{aligned} \tag{1.9}$$

The positive square root of the variance of a *random variable* is its standard deviation.

$$\sigma[A] = \sqrt{\mathbb{V}[A]} = \sqrt{\mathbb{E}[(A^2)] - \mathbb{E}^2[A]} \tag{1.10}$$

$$\sigma[B] = \sqrt{\mathbb{V}[B]} = \sqrt{\mathbb{E}[(B^2)] - \mathbb{E}^2[B]} \tag{1.11}$$

While for the *random variable* C, it shares the same form of variance and standard deviation with B since it is defined as a function of B and remains continuous. Its variance is:

$$\mathbb{V}[C] = \mathbb{E}[(C^2)] - \mathbb{E}^2[C] \tag{1.12}$$

and its standard deviation is:

$$\sigma[C] = \sqrt{\mathbb{V}[C]} = \sqrt{\mathbb{E}[(C^2)] - \mathbb{E}^2[C]} \tag{1.13}$$

1.3 Monte Carlo method

The Monte Carlo method can be seen as a numerical method to estimate the values of an integral statistically [14]. The idea of the Monte Carlo method is firstly developed by *Stanislaw Ulam*, as part of the Manhattan Project during *World War II*, with *John von Neumann, Enrico Fermi, Nicholas Metropolis*, and others. *Metropolis* and *Ulam* published the first paper in the open literature on the method [42].

1.3.1 Estimation of expected values

The Monte Carlo method is based on the *law of large numbers*. If n_{MC} is the number of independent *realization* of X(X can be discrete or continuous), and x_j denoted the value of the j^{th} realization, the expected value of X can be estimated without bias when n_{MC} tends to infinity:

$$\mathbb{E}[X] = \lim_{n_{MC}\to+\infty}\left(\frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} x_j\right). \tag{1.14}$$

For the three *random variable*s defined previously: A, B and C, their expected value can be written as Eq.1.4, Eq.1.5 and Eq.1.6, which leads to the following equations:

$$\mathbb{E}[A] = \sum_{i=1}^{n} a_i \mathscr{P}_A(a_i) = \lim_{n_{MC}\to+\infty}\left(\frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} a_j\right) \tag{1.15}$$

$$\mathbb{E}[B] = \int_{b_{min}}^{b_{max}} b\mathscr{P}_B(b)db = \lim_{n_{MC}\to+\infty}\left(\frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} b_j\right) \tag{1.16}$$

$$\mathbb{E}[C] = \int_{b_{min}}^{b_{max}} f(b)\mathscr{P}_B(b)db = \lim_{n_{MC}\to+\infty}\left(\frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} f(b_j)\right) \tag{1.17}$$

It's noted that $b_j \in [b_{min}, b_{max}]$ and the index j signifies the j^{th} realization of the *random variable* B. The j^{th} realization of A follows the pdf $\mathscr{P}_A$ giving the value of a_j, and the j^{th} realization of B follows the pdf $\mathscr{P}_B$ giving the value of b_j.

However, practically n_{MC} can not be infinite. Nevertheless, following the *law of large numbers*, with the increase of n_{MC}, this estimation can be almost always convergent, which leads to the following equations when n_{MC} is large enough:

$$\sum_{i=1}^{n} a_i \mathscr{P}_A(a_i) \approx \frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} a_j \tag{1.18}$$

$$\int_{b_{min}}^{b_{max}} b\mathscr{P}_B(b)db \approx \frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} b_j \tag{1.19}$$

$$\boxed{\int_{b_{min}}^{b_{max}} f(b)\mathscr{P}_B(b)db \approx \frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} f(b_j)} \tag{1.20}$$

Eq.1.18, Eq.1.19 and Eq.1.20 are also called the Monte Carlo *estimators* [46].

Example.1.3.1 shows an application of the Monte Carlo method in playing a die.

EXAMPLE 1.3.1

Playing a die.
If our *random variable* A is now the result of playing a die, while the action of playing for once is called one *realization* or one *sampling*. The result will have six possible values, which are:

$$A \in \{1, 2, 3, 4, 5, 6\} \tag{1.21}$$

If we have a perfect die, then the probability of the fact that A takes each value in the set will be the same: $\mathscr{P}_A(a_i) = \frac{1}{6}$, $\forall i \in \{1, 2, 3, 4, 5, 6\}$.
Therefore, the expected value of A is given by:

$$\begin{aligned}\mathbb{E}[A] &= \sum_{i=1}^{6} \mathscr{P}_A(a_i) \\ &= 1 \times \frac{1}{6} + 2 \times \frac{1}{6} + 3 \times \frac{1}{6} + 4 \times \frac{1}{6} + 5 \times \frac{1}{6} + 6 \times \frac{1}{6} \\ &= \frac{7}{2}.\end{aligned} \tag{1.22}$$

Now, if we estimate the $\mathbb{E}[A]$ by the Monte Carlo method, we need to:

- play the die numerous times, noted n_{MC};
- save the value of a_j for the j^{th} time of estimation;
- calculate the mean of all saved a_j, which is:

$$\mathbb{E}[A] \approx \frac{1}{n_{MC}}\sum_{j=1}^{n_{MC}} a_j \tag{1.23}$$

Finally, with a large enough n_{MC}, the value of $\mathbb{E}[A]$ estimated by the Monte Carlo method will tend to $\frac{7}{2}$.

Based on Eq.1.20, one can estimate the value of any integrals by the Monte Carlo method. It will be sufficient to transform the integral to a form of expected value (Eq.1.20). The form of expected value consists of a **pdf** of Monte Carlo (normalised in its domain of integral), and a **weight** of Monte Carlo (which is $f(b_j)$ in this case).

It is also noted that the two boundaries of the integral domain can tend to infinity. It requires only insuring that the normalisation of the pdf (Eq.1.2) is satisfied.

The Monte Carlo method has advantages compared to other methods when the dimension of integral increases. For example the *random variable* taking the value of a function $g(B_1, B_2, \ldots, B_n)$ is defined previously and its expected value is defined in Eq.1.7. Based on the *law of large numbers*, the following equation can be established:

$$\int_{b_{1min}}^{b_{1max}} \mathscr{P}_{B_1}(b_1)db_1 \cdots \int_{b_{nmin}}^{b_{nmax}} \mathscr{P}_{B_n}(b_n)db_n g(b_1,\ldots,b_n) \approx \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} g_j(b_1,\ldots,b_n) \quad (1.24)$$

$g_j(b1,\ldots,b_n)$ denotes the j^{th} realization of the function $g(B_1,\ldots,B_n)$. Each of such realization needs n realizations to determinate $b_1, b_2, \ldots, b_n$, following the pdf $\mathscr{P}_{B1}, \mathscr{P}_{B2}, \ldots, \mathscr{P}_{Bn}$.

1.3.2 Double randomization

A form of expected value consists of a pdf and a weight of Monte Carlo, as shown in Eq.1.20. We will often see in the rest of this thesis that the weight of Monte Carlo includes an expected value of another random number. In other words, from an algorithmic point of view, the knowledge of Monte Carlo weight for each realization requires the knowledge of another expected value, which also needs to be estimated by the Monte Carlo method.

We consider here an integral with a form of expected value:

$$\int_{b_{1min}}^{b_{1max}} \mathscr{P}_{B_1}(b_1) f\left(b_1, \mathbb{E}_{B_2|B_1}[B_2|B_1]\right) db_1 \equiv \mathbb{E}_{B_1}\left[f(B_1, \mathbb{E}_{B_2|B_1}[B_2|B_1])\right] \quad (1.25)$$

reminding B_1 and $B_2|B_1$ are the random variables. The notation $B_2|B_1$ signifies that the random number B_2 depends on B_1, read as B_2 knowing B_1. $\mathscr{P}_{B_1}$ is the pdf of B_1 and $\mathscr{P}_{B_2|B_1}$ is the pdf of $B_2|B_1$. In order to distinguish the two expected values, $\mathbb{E}_{B_1}[B_1]$ is the expected value of B_1 and $\mathbb{E}_{B_1|B_2}[B_1|B_2]$ is the expected value of $B_2|B_1$. $\mathbb{E}_{B_2|B_1}[B_2|B_1]$ in Eq.1.25 can be written as:

$$\mathbb{E}[B_2|B_1] = \int_{b_{2min}}^{b_{2max}} \mathscr{P}_{B_2|B_1}(b_2) b_2 db_2 \quad (1.26)$$

In order to estimate the value of integral in Eq.1.25, the first strategy consists of (for each realization):

- sampling b_1 following the pdf $\mathscr{P}_{B_1}$;
- estimate the expected value of $\mathbb{E}_{B_2|B_1}[B_2|B_1 = b_1]$ by Monte Carlo method;
- use the result of estimation of $\mathbb{E}_{B_2|B_1}[B_2|B_1 = b_1]$ (noted e) as the second input of the function f and b_1 as the first input of the function f. $f(b_1, e)$ is then the result of one realization.

However, this strategy is not practicable. If the number of sampling for B_1 is n_1 and that for $B_2|B_1$ is n_2, the total number of sampling following this strategy will be $n_1 n_2$. The calculating time will be huge if more expected values appear in the function f in Eq.1.25.

It is then necessary to introduce the strategy of *Double randomization* here. The idea of double randomization is that: *The expected value of an expected value is also an expected value*:

$$\begin{aligned} &\mathbb{E}_{B_1}\left[f(B_1, \mathbb{E}_{B_2|B_1}[B_2|B_1])\right] \\ =&\mathbb{E}_{B_1,B_2|B_1}\left[f(B_1, B_2)\right] \\ =&\int_{b_{1min}}^{b_{1max}} \mathscr{P}_{B_1}(b_1) \int_{b_{2min}}^{b_{2max}} \mathscr{P}_{B_2|B_1=b_1}(b_2) f\left(b_1, b_2\right) db_1 db_2 \end{aligned} . \quad (1.27)$$

However, the function f needs to be linear to apply this strategy. Since the non-linear Monte Carlo estimation is not studied in this thesis, readers can be directed to the Ph.D. work of Jean-Marc Tregan for more details [55].

1.3.3 Estimation of variances and standard deviations

The Monte Carlo method can estimate an expected value and also evaluate the uncertainty of this estimation. This property is based on another important law in the statistic: the *central limit theorem*, which establishes that when independent *random variable*s are summed up, the distribution of this sum tends to a Gaussian distribution. This theorem indicates that the estimation of an expected value by the Monte Carlo method is also a *random variable*, of which the distribution tends to the Gaussian distribution when the number of realizations n_{MC} increases. Therefore, this Gaussian distribution of the estimations can be characterized by its variance and its standard deviation.

Suppose we try to estimate the expected value of a *random variable* X by the Monte Carlo method. In that case, the value of such estimation is also a random number, denoted $\check{X}$:

$$\check{X} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} x_j \tag{1.28}$$

If n_{MC} is large enough, according to the *central limit theorem*, the distribution of $\check{X}$ will follow the Gaussian distribution, characterized by its variance $\mathbb{V}[\check{X}]$ or its standard deviation $\sigma[\check{X}]$ [14]:

$$\sigma[\check{X}] = \sqrt{\mathbb{V}[\check{X}]} = \sqrt{\frac{1}{n_{MC}-1}(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} (x_j)^2 - (\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} x_j)^2} \tag{1.29}$$

Example.1.3.2 demonstrates the process of estimating an integral of such a form:

$$\int_{x_{min}}^{x_{max}} f(x)dx \tag{1.30}$$

Example 1.3.2

Considering an integral of which the analytical solution is known, a Monte Carlo method is applied to estimate this same solution:

$$\int_0^{\frac{1}{2}} e^{-x}dx = 1 - \frac{1}{\sqrt{e}} \approx 0.39346. \tag{1.31}$$

This integral can be rewritten and be transformed into a form of the expected value of a *random variable* Y:

$$\mathbb{E}(Y) = \int_0^{\frac{1}{2}} e^{-x}dx = \int_0^{\frac{1}{2}} \mathcal{P}(x)\frac{e^{-x}}{\mathcal{P}(x)}dx \tag{1.32}$$

In Eq.1.32, the only constraint is that the $\mathcal{P}(x)$ needs to be normalized in the domain of integral, which leads to infinite possible choices for building this pdf. In this example, $\mathcal{P}(x) = 2$ is chosen, which means x is uniformly sampled within the domain of integral $x \in [0, \frac{1}{2}]$.

Furthermore, we note $\check{Y}$ as the result of $\mathbb{E}(Y)$ estimated by the Monte Carlo method. It is then given by:

$$\check{Y} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(\frac{e^{-x_j}}{\mathcal{P}(x_j)}\right) = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(\hat{w}_j\right). \tag{1.33}$$

Also, its standard deviation $\sigma[\check{Y}]$ is given by:

$$\sigma[\check{Y}] = \sqrt{\frac{1}{n_{MC}-1}\left(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \hat{w}_j^2 - \left(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \hat{w}_j\right)^2\right.}. \tag{1.34}$$

x_j is the j^{th} realization of the value x following the pdf $\mathcal{P}(x)$. The term $\frac{e^{-x_j}}{\mathcal{P}(x_j)}$ is called the weight of Monte Carlo, denoted $\hat{w}_j$. This estimation is based on the following algorithm:

- uniformly sample a value of x_j for $x_j \in [0, \frac{1}{2}]$;
- compute and save the weight $\hat{w}_j$ and $\hat{w}_j^2$:

$$\hat{w}_j = \frac{e^{-x_j}}{\mathcal{P}(x_j)}, \tag{1.35}$$

$$\hat{w}_j^2 = \left(\frac{e^{-x_j}}{\mathcal{P}(x_j)}\right)^2; \tag{1.36}$$

- after n_{MC} times of realization, calculate the expected value and the standard deviation following Eq.1.33 and Eq.1.34.

In Fig.1.1 and Table.1.2 (at the end of this chapter), the estimations by Monte Carlo method are run with different n_{MC}. Obviously, with the increase of the n_{MC}, the value of estimations tends to be convergent to the analytical solution, and the variance of the estimations is reduced.

1.3.4 Variance reduction by importance sampling

In Example.1.3.2, the pdf $\mathcal{P}(x)$ is chosen arbitrarily. However, this choice enormously influences convergence performance: the standard deviation will change with different choices of pdf. The importance sampling technique allows optimizing the choice of this pdf.

In order to estimate the value of an integral such as Eq.1.30, the strategy of choosing a "good" pdf is to find a function $f^*(x)$ which is closed to $f(x)$, then a normalised pdf $\mathcal{P}^*(x)$ can be generated [14]:

$$\mathcal{P}^*(x) = \frac{f^*(x)}{\int_{x_{min}}^{x_{max}} f^*(x)dx} \tag{1.37}$$

Eq.1.30 becomes:

$$\int_{x_{min}}^{x_{max}} \mathcal{P}^*(x) \frac{f(x)}{\mathcal{P}^*(x)} dx \tag{1.38}$$

Nowadays, it is common to generate the pseudo-random number numerically by the widely used pseudo-random number generators such as Mersen Twister and RANLUX [14, 25]. It is always more challenging to generate random numbers with a non-uniform pdf.

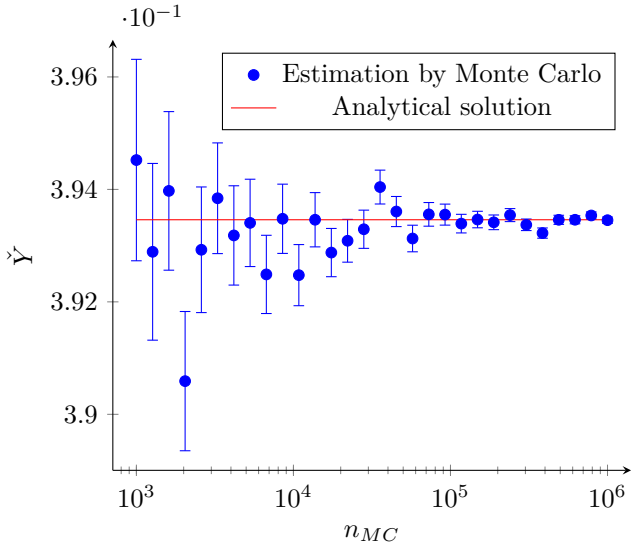


Figure 1.1 – Estimations of expected values and standard deviations. A confidence interval of $\pm\sigma$ is shown in this figure.

Practically, in order to generate a specifically distributed random number, we generate a random number r_i uniformly between 0 and 1, and we reverse the following equation:

$$r_i = cdf(x) \tag{1.39}$$

which is:

$$x = cdf^{-1}(r_i) \tag{1.40}$$

cdf (Cumulative distribution function) is the integral of the pdf:

$$cdf(x) = \int_{x_{min}}^{x} \mathcal{P}^*(x')dx' \tag{1.41}$$

However, this technique of sampling is feasible only when the inversion of *cdf* is possible (a solution exists for the reformulation from Eq.1.39 to Eq.1.40). In the Example.1.3.3, we will apply the technique of importance sampling to estimate the value of the integral in Eq.1.31.

Example 1.3.3

Considering the integral of Eq.1.31, $f^*(x)$ will be chosen as the first order of expansion e^{-x} to build $\mathcal{P}^*$:

$$f^*(x) = 1 - x \tag{1.42}$$

The integral can then be rewritten and transformed into a form of the expected value of a *random number* Y^*:

$$\mathbb{E}(Y^*) = \int_0^{\frac{1}{2}} e^{-x}dx = \int_0^{\frac{1}{2}} \mathcal{P}^*(x)\frac{e^{-x}}{\mathcal{P}^*(x)}dx \tag{1.43}$$

where

$$\mathscr{P}^*(x) = \frac{f^*(x)}{\int_0^{\frac{1}{2}} f^*(x)dx} = (1-x)\frac{8}{3}. \tag{1.44}$$

Its cdf is therefore:

$$cdf(x) = \int_0^x \mathscr{P}^*(x')dx' = -\frac{x^2}{4} + \frac{8x}{3} \tag{1.45}$$

We uniformly generate a random number r_0 between 0 and 1 so that the random number X can be sampled following $\mathscr{P}^*(x)$:

$$x = cdf^{-1}(r_0) = -\frac{\sqrt{-3r_0+4}-2}{2} \tag{1.46}$$

Furthermore, we note $\check{Y}^*$ as the result of $\mathbb{E}(Y^*)$ estimated by the Monte Carlo method. It is then given by:

$$\check{Y}^* = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(\frac{e^{-x_j}}{\mathscr{P}^*(x_j)}\right) = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(\hat{w}_j^*\right). \tag{1.47}$$

Also, its standard deviation $\sigma[\check{Y}^*]$ is given by:

$$\sigma[\check{Y}^*] = \sqrt{\frac{1}{n_{MC}-1}\left(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \hat{w}_j^{*2} - \left(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \hat{w}_j^*\right)^2\right)}. \tag{1.48}$$

x_j is the j^{th} realization of the value x following the pdf $\mathscr{P}^*(x)$. The term $\frac{e^{-x_j}}{\mathscr{P}(x_j)}$ is called the weight of Monte Carlo, denoted $\hat{w}_j$.
This estimation is based on the following algorithm:

- uniformly sample a value of r_0 for $r_0 \in [0,1]$;
- calculate x_j based on the sampled r_0;
- calculate and save the weight $\hat{w}_j^*$ and $\hat{w}_j^{*2}$:

$$\hat{w}_j^* = \frac{e^{-x_j}}{\mathscr{P}^*(x_j)}, \tag{1.49}$$

$$\hat{w}_j^{*2} = \left(\frac{e^{-x_j}}{\mathscr{P}^*(x_j)}\right)^2; \tag{1.50}$$

- after n_{MC} times of realization, calculate the expected value and the variance following Eq.1.47 and Eq.1.48.

Table.1.3 lists the results of estimations (at the end of this chapter), using the optimized pdf $\mathscr{P}$. In Fig.1.2, the estimations by Monte Carlo method with two pdfs are compared.

1.3.5 Vectorization

When the integrals to be estimated have the same integral domain, the process of estimation by the Monte Carlo method can be vectorized using a single and the same pdf. As an

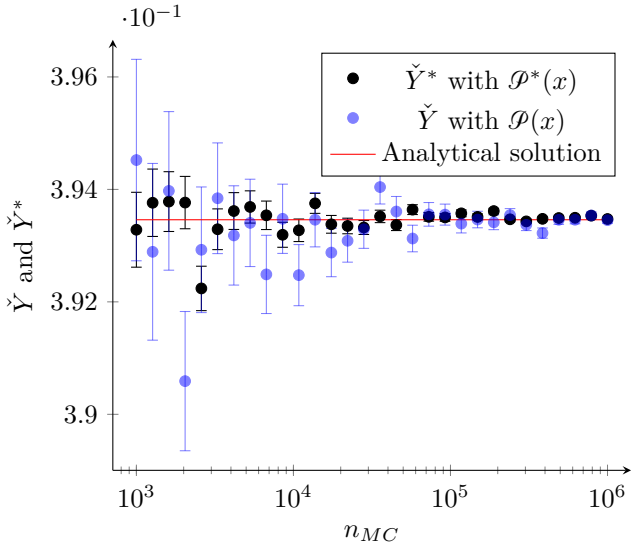


Figure 1.2 – Estimations of expected values and standard deviations. A confidence interval of $\pm\sigma$ is shown in this figure. The convergence performance is better for the estimation with $\mathscr{P}^*(x)$ using the importance sampling technique.

example, we would like to estimate the three following integrals simultaneously:

$$\int_{t_1}^{t_2} f_1(x)dx, \int_{t_1}^{t_2} f_2(x)dx, \int_{t_1}^{t_2} f_3(x)dx \tag{1.51}$$

We choose a common pdf $\mathscr{P}_c$ and rewritten Eq.1.51 into a form of a vector of expected values of three *random number* Y_1, Y_2 and Y_3:

$$\begin{bmatrix}\mathbb{E}(Y_1)\\ \mathbb{E}(Y_2)\\ \mathbb{E}(Y_3)\end{bmatrix} = \int_{t_1}^{t_2} \mathscr{P}_c(x) \begin{bmatrix}f_1(x)\\ f_2(x)\\ f_3(x)\end{bmatrix} \left(\frac{1}{\mathscr{P}_c(x)}\right) dx \tag{1.52}$$

Furthermore, we note $\check{Y}_1$, $\check{Y}_2$ and $\check{Y}_3$ as the results of $\mathbb{E}(Y_1)$, $\mathbb{E}(Y_2)$ and $\mathbb{E}(Y_3)$ estimated by Monte Carlo method. They are then given by:

$$\begin{bmatrix}\check{Y}_1\\ \check{Y}_2\\ \check{Y}_3\end{bmatrix} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \begin{bmatrix}f_1(x_j)\\ f_2(x_j)\\ f_3(x_j)\end{bmatrix} \left(\frac{1}{\mathscr{P}_c(x_j)}\right) \tag{1.53}$$

This technique can then be generalized to n integrals:

$$\begin{bmatrix}\check{Y}_1\\ \check{Y}_2\\ \cdots\\ \check{Y}_n\end{bmatrix} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \begin{bmatrix}f_1(x_j)\\ f_2(x_j)\\ \cdots\\ f_n(x_j)\end{bmatrix} \left(\frac{1}{\mathscr{P}_c(x_j)}\right) \tag{1.54}$$

The interest of this vectorization technique is that the n integrals with the same integral domain can be estimated based on the same set of random numbers, which requires n_{MC} realizations of X. However, without vectorization, n Monte Carlo estimations are needed to estimate the n integral, which requires $n \times n_{MC}$ realizations of X. Since most of the

calculating time is spent on generating random numbers, this vectorization technique can reduce the calculating time for almost a factor of n, which is the dimension of the vector composing of integrals.

The vectorization technique can also be used for integrals with different integral domains. The strategy is first to homogenize the integral domain and, secondly, apply the previously mentioned technique to vectorize the estimations. Mathematically, lots of methods can be used to homogenize the integral domains. Herein, only the technique of variable substitution is introduced. Let us consider two integrals with different integral domains:

$$\int_{x_1^{min}}^{x_1^{max}} f_1(x)dx, \int_{x_2^{min}}^{x_2^{max}} f_2(x)dx \tag{1.55}$$

The first integral in Eq.1.55 can be rewritten:

$$\int_{x_1^{min}}^{x_1^{max}} f_1(x)dx = \int_0^{x_1^{max}} f_1(x)dx - \int_0^{x_1^{min}} f_1(x)dx \tag{1.56}$$

We then choose a constant $\dot{c}$ to homogenise the integral domains into $[0, \dot{c}]$.

We impose:

$$u_1 = x\frac{\dot{c}}{x_1^{max}} \tag{1.57}$$

and

$$v_1 = x\frac{\dot{c}}{x_1^{min}} \tag{1.58}$$

Eq.1.56 becomes:

$$\int_{x_1^{min}}^{x_1^{max}} f_1(x)dx = \int_0^{\dot{c}} f(u_1\frac{x_1^{max}}{\dot{c}})\frac{x_1^{max}}{\dot{c}}du_1 - \int_0^{\dot{c}} f(v_1\frac{x_1^{min}}{\dot{c}})\frac{x_1^{min}}{\dot{c}}dv_1 \tag{1.59}$$

A similar technique can be applied to the second integral in Eq.1.55:

$$\int_{x_2^{min}}^{x_2^{max}} f_2(x)dx = \int_0^{\dot{c}} f(u_2\frac{x_2^{max}}{\dot{c}})\frac{x_2^{max}}{\dot{c}}du_2 - \int_0^{\dot{c}} f(v_2\frac{x_2^{min}}{\dot{c}})\frac{x_2^{min}}{\dot{c}}dv_2 \tag{1.60}$$

with

$$u_2 = x\frac{\dot{c}}{x_2^{max}} \tag{1.61}$$

and

$$v_2 = x\frac{\dot{c}}{x_2^{min}} \tag{1.62}$$

Finally, the 2 integrals in Eq.1.55 can be vectorized:

$$\begin{bmatrix} \int_{x_1^{min}}^{x_1^{max}} f_1(x)dx \\ \int_{x_2^{min}}^{x_2^{max}} f_2(x)dx \end{bmatrix} = \int_0^{\dot{c}} \begin{bmatrix} f(x\frac{x_1^{max}}{\dot{c}})\frac{x_1^{max}}{\dot{c}} - f(x\frac{x_1^{min}}{\dot{c}})\frac{x_1^{min}}{\dot{c}} \\ f(x\frac{x_2^{max}}{\dot{c}})\frac{x_2^{max}}{\dot{c}} - f(x\frac{x_2^{min}}{\dot{c}})\frac{x_2^{min}}{\dot{c}} \end{bmatrix} dx \tag{1.63}$$

The two integrals can then be estimated by one set of random numbers following the previously mentioned vectorization technique. It is then possible to generalize this technique to n dimension:

$$\begin{bmatrix} \int_{x_1^{min}}^{x_1^{max}} f_1(x)dx \\ \int_{x_1^{min}}^{x_2^{max}} f_2(x)dx \\ \cdots \\ \int_{x_n^{min}}^{x_n^{max}} f_n(x)dx \end{bmatrix} = \int_0^{\dot{c}} \begin{bmatrix} f(x\frac{x_1^{max}}{\dot{c}})\frac{x_1^{max}}{\dot{c}} - f(x\frac{x_1^{min}}{\dot{c}})\frac{x_1^{min}}{\dot{c}} \\ f(x\frac{x_2^{max}}{\dot{c}})\frac{x_2^{max}}{\dot{c}} - f(x\frac{x_2^{min}}{\dot{c}})\frac{x_2^{min}}{\dot{c}} \\ \cdots \\ f(x\frac{x_n^{max}}{\dot{c}})\frac{x_n^{max}}{\dot{c}} - f(x\frac{x_n^{min}}{\dot{c}})\frac{x_n^{min}}{\dot{c}} \end{bmatrix} dx \tag{1.64}$$

We will then apply this technique to calculate two integrals in the Example.1.3.4.

Example 1.3.4

Considering the following two integrals with different integral domains:

$$\int_0^{\frac{1}{2}} e^{-x}dx; \int_0^1 e^{-x}dx \tag{1.65}$$

The first integral can be then rewritten:

$$\int_0^{\frac{1}{2}} e^{-x}dx = \int_0^1 \frac{1}{2}e^{-\frac{x}{2}}dx \tag{1.66}$$

We then vectorize the estimations of the two integrals, choosing a uniform pdf and transformed it into an expected value form of *random numbers* Y_1 and Y_2:

$$\begin{bmatrix} \mathbb{E}(Y_1) \\ \mathbb{E}(Y_2) \end{bmatrix} = \int_0^1 \mathscr{P}(x) \begin{bmatrix} \frac{1}{2}e^{-\frac{x}{2}} \\ e^{-x} \end{bmatrix} \left(\frac{1}{\mathscr{P}(x)}\right) dx \tag{1.67}$$

where

$$\mathscr{P}(x) = 1. \tag{1.68}$$

Furthermore, we note $\check{Y}_1$ and $\check{Y}_2$ as the results of $\mathbb{E}(Y_1)$ and $\mathbb{E}(Y_2)$ estimated by Monte Carlo method:

$$\begin{bmatrix} \check{Y}_1 \\ \check{Y}_2 \end{bmatrix} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \begin{bmatrix} \frac{e^{-\frac{x_j}{2}}}{2\mathscr{P}(x_j)} \\ \frac{e^{-x_j}}{\mathscr{P}(x_j)} \end{bmatrix} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \begin{bmatrix} \hat{w}_{j,1} \\ \hat{w}_{j,2} \end{bmatrix} \tag{1.69}$$

The standard deviation of this estimation of the Monte Carlo method can also be established:

$$\begin{bmatrix} \sigma[\check{Y}_1] \\ \sigma[\check{Y}_2] \end{bmatrix} = \sqrt{\frac{1}{n_{MC}-1}\left(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \begin{bmatrix} \hat{w}_{j,1} \\ \hat{w}_{j,2} \end{bmatrix}^2\right) - \left(\frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \begin{bmatrix} \hat{w}_{j,1} \\ \hat{w}_{j,2} \end{bmatrix}\right)^2}. \tag{1.70}$$

This estimation is based on the following algorithm:

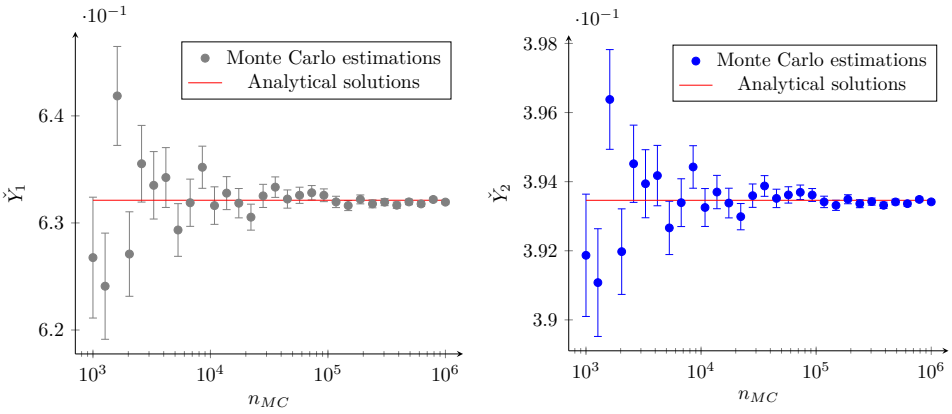
- uniformly sample a value of r_0 for $r_0 \in [0, 1]$;
- calculate x_j based on the sampled r_0;
- count and save the weight $\vec{w}_j$ and $\vec{w}_j^2$:

$$\vec{w}_j = \begin{bmatrix} \frac{e^{-\frac{x_j}{2}}}{2\mathscr{P}(x_j)} \\ \frac{e^{-x_j}}{\mathscr{P}(x_j)} \end{bmatrix} \tag{1.71}$$

$$\vec{w}_j^2 = \begin{bmatrix} \frac{e^{-\frac{x_j}{2}}}{2\mathcal{P}(x_j)} \\ \frac{e^{-x_j}}{\mathcal{P}(x_j)} \end{bmatrix}^2 ; \tag{1.72}$$

- after n_{MC} times of realization, calculate the expected value and the variance following Eq.1.69 and Eq.1.70.

In Fig.1.3, the vectorized estimations by Monte Carlo method for the two integrals in Eq.1.65 are plotted. The numerical results are in Table.1.4 and Table.1.5 (at the end of this chapter).



(a) Estimations of $\int_0^1 e^{-x}dx$.

(b) Estimations of $\int_0^{\frac{1}{2}} e^{-x}dx$.

Figure 1.3 – The estimations use the same set of random numbers, which accelerates the calculation. A confidence interval of $\pm\sigma$ is shown in this figure. However, it is found that the convergence performance for the estimations of the two integrals is different. As it is discussed in Section.1.3.4, the choice of pdf has enormous influence on the convergence performance. When we apply the vectorization technique, the same pdf is used for all integrals. Therefore, this common pdf must be chosen carefully.

1.3.6 Sensitivity

We have shown in the previous subsections how to estimate the value of an integral by applying the Monte Carlo method. In this subsection, we focus on estimating the derivative of the integral with respect to a parameter (which is called the estimation of sensitivity).

Suppose the following integral:

$$F(\ddot{\pi}) = \int_{x_{min}(\ddot{\pi})}^{x_{max}(\ddot{\pi})} f(x, \ddot{\pi})dx \tag{1.73}$$

where $\ddot{\pi}$ is a parameter of the integral. For each $\ddot{\pi}$ in its space of definition, the value of $F(\ddot{\pi})$ can be estimated by Monte Carlo method. Furthermore, its derivative with respect to $\ddot{\pi}$: $\partial_{\ddot{\pi}}F$ is also an integral which can also be estimated by Monte Carlo method [10, 51]. Before introducing the sensitivity estimation directly, we will first introduce a deterministic method: finite difference method to approximate the sensitivity [21]. However, the severe limitations and inconveniences of the latter method will also be discussed hereafter.

Approximation by finite difference method

When F is derivable around $\ddot{\pi}$:

$$\partial_{\ddot{\pi}} F = \lim_{\epsilon \to 0} \frac{F(\ddot{\pi} + \epsilon) - F(\ddot{\pi} - \epsilon)}{2\epsilon} \tag{1.74}$$

The finite difference method approximates $\partial_{\ddot{\pi}} F$ by giving a non-zero fixed value to ϵ. We note its value of approximation as $\partial_{\ddot{\pi}} \tilde{F}$:

$$\partial_{\ddot{\pi}} \tilde{F} \approx \frac{F(\ddot{\pi} + \Delta\ddot{\pi}) - F(\ddot{\pi} - \Delta\ddot{\pi})}{2\Delta\ddot{\pi}} \tag{1.75}$$

Since the integral F is estimated by the Monte Carlo method, its statistical standard deviation $\sigma(F)$ is also estimated at the same time [14]. When approximating $\partial_{\ddot{\pi}} \tilde{F}$, its standard deviation $\sigma\left[\partial_{\ddot{\pi}} \tilde{F}\right]$ can be obtained [21]:

$$\sigma\left[\partial_{\ddot{\pi}} \tilde{F}\right] \approx \frac{\sigma(F(\ddot{\pi} + \Delta\ddot{\pi})) + \sigma(F(\ddot{\pi} - \Delta\ddot{\pi}))}{2\Delta\ddot{\pi}} \tag{1.76}$$

The finite difference method has mainly two inconveniences:

1. It approximates $\partial_{\ddot{\pi}} \tilde{F}$ by the discretization of the parameter $\ddot{\pi}$. Mathematically, when $F(\ddot{\pi})$ is discontinued between $(\ddot{\pi} - \Delta\ddot{\pi})$ and $(\ddot{\pi} + \Delta\ddot{\pi})$, the approximation is unreliable.
2. The related standard deviation $\sigma\left[\partial_{\ddot{\pi}} \tilde{F}\right]$ is hard to control[21]. According to Eq.1.74 and Eq.1.75, with an $\Delta\ddot{\pi}$ tending to 0, the bias of approximation decreases. However, the $\sigma\left[\partial_{\ddot{\pi}} \tilde{F}\right]$ will become huge since $\Delta\ddot{\pi}$ tends to 0.

Direct differentiation of the integral

Instead of approximating the sensitivity $\partial_{\ddot{\pi}} F$ by finite difference method, we reformulate the sensitivity into a new integral form, and estimate it by Monte Carlo method [10, 51].

We differentiate Eq.1.73 with respect to $\ddot{\pi}$:

$$\partial_{\ddot{\pi}} F = \partial_{\ddot{\pi}} x_{max}(\ddot{\pi}) f\left(x_{max}(\ddot{\pi}); \ddot{\pi}\right) - \partial_{\ddot{\pi}} x_{min}(\ddot{\pi}) f\left(x_{min}(\ddot{\pi}); \ddot{\pi}\right) + \int_{x_{min}(\ddot{\pi})}^{x_{max}(\ddot{\pi})} \partial_{\ddot{\pi}} f(x, \ddot{\pi}) dx \tag{1.77}$$

which can be transformed into a single integral[51]:

$$\partial_{\ddot{\pi}} F = \int_{x_{min}(\ddot{\pi})}^{x_{max}(\ddot{\pi})} \left[\partial_{\ddot{\pi}} f(x, \ddot{\pi}) + \partial_x (v(x) f(x, \ddot{\pi}))\right] dx \tag{1.78}$$

where v is called the deformation velocity[51]:

$$v(x) = \partial_{\ddot{\pi}} x_{min}(\ddot{\pi}) + \frac{\partial_{\ddot{\pi}} x_{max}(\ddot{\pi}) - \partial_{\ddot{\pi}} (x_{min}(\ddot{\pi}))}{x_{max}(\ddot{\pi}) - x_{min}(\ddot{\pi})} (x - x_{min}(\ddot{\pi})) \tag{1.79}$$

The method can be generalized to an integral of n dimensions:

$$F_n = \int_{x_1^{min}(\ddot{\pi})}^{x_1^{max}(\ddot{\pi})} dx_1 \int_{x_2^{min}(\ddot{\pi})}^{x_2^{max}(\ddot{\pi})} dx_2 \cdots \int_{x_n^{min}(\ddot{\pi})}^{x_n^{max}(\ddot{\pi})} dx_n f(x_1, x_2, \ldots, x_n; \ddot{\pi}) \tag{1.80}$$

of which the derivative with respect to $\ddot{\pi}$ is:

$$\partial_{\ddot{\pi}} F_n = \int_{x_1^{min}(\ddot{\pi})}^{x_1^{max}(\ddot{\pi})} dx_1 \int_{x_2^{min}(\ddot{\pi})}^{x_2^{max}(\ddot{\pi})} dx_2 \cdots \int_{x_n^{min}(\ddot{\pi})}^{x_n^{max}(\ddot{\pi})} dx_n \left(\partial_{\ddot{\pi}} f(x_1, x_2, \ldots, x_n; \ddot{\pi}) + \vec{\nabla}_x \cdot \vec{V} f(x_1, x_2, \ldots, x_n; \ddot{\pi}) \right) \tag{1.81}$$

where the vector of deformation velocity $\vec{V} = [v_1, v_2, \ldots, v_n]$ is built component by component, for $i \in [1, 2, \ldots, n]$:

$$v_i(x_i) = \partial_{\ddot{\pi}} x_i^{min}(\ddot{\pi}) + \frac{\partial_{\ddot{\pi}} x_i^{max}(\ddot{\pi}) - \partial_{\ddot{\pi}}(x_i^{min}(\ddot{\pi}))}{x_i^{max}(\ddot{\pi}) - x_i^{min}(\ddot{\pi})} (x_i - x_i^{min}(\ddot{\pi})) \tag{1.82}$$

Concerning this method, it can be pointed out that:

1. The formulations of F and $\partial_{\ddot{\pi}} F$ have the same integral domain, meaning they can be vectorized and estimated by the same set of random numbers.
2. Since the derivative $\partial_{\ddot{\pi}} F$ is rewritten in an integral formulation (Eq.1.81), it can then be estimated by Monte Carlo method, benefiting from the variance reduction technique (by using the importance sampling technique).
3. Compared to the method of approximation (finite difference method), this method estimates the exact value of the integral without bias. Moreover, unlike the finite difference method, the standard deviation does not depend on the discretization parameter ($\Delta\ddot{\pi}$ in Eq.1.75).
4. The difficulty of applying this method is mainly about defining the deformation velocity vector (see Eq.1.81). In some exceptional cases, for example, when the integral domain is independent of the parameter $\ddot{\pi}$, the deformation velocity vector is $\vec{0}$. In this particular case, it is referred to as the problem of parametric sensitivity [10].

The following example (Example.1.3.5) compares the results of estimations by this technique and approximations by finite difference method.

Example 1.3.5

Consider the following integral which is a function of $\ddot{\pi}$, $\forall \ddot{\pi} \in \mathbb{R}$:

$$F(\ddot{\pi}) = \int_0^{2\ddot{\pi}} e^{-\ddot{\pi}x} dx \tag{1.83}$$

For each $\ddot{\pi}$ in its space of definition, we apply Monte Carlo method to estimate the value of the integral $F(\ddot{\pi})$ and the standard deviation of the estimation $\sigma[F(\ddot{\pi})]$ will be calculated respectfully:

$$F(\ddot{\pi}) = \int_0^{2\ddot{\pi}} \mathscr{P}(x, \ddot{\pi}) \frac{e^{-\ddot{\pi}x}}{\mathscr{P}(x, \ddot{\pi})} dx \tag{1.84}$$

where $\mathscr{P}(x, \ddot{\pi})$ is chosen to be uniform:

$$\mathscr{P}(x, \ddot{\pi}) = \frac{1}{2\ddot{\pi}} \tag{1.85}$$

We apply the finite difference method to approximate its derivative $\partial_{\ddot{\pi}} \tilde{F}$ and the standard deviation of the approximation $\sigma[\partial_{\ddot{\pi}} \tilde{F}]$ by Eq.1.75 and Eq.1.76.

Alternatively, applying Eq.1.78 and Eq.1.79 to this example, its derivative can be rewritten in an integral formulation. We apply the Monte Carlo method to estimate its value, and the standard deviation of its estimation will be calculated respectfully:

$$\partial_{\ddot{\pi}} F = \int_0^{2\ddot{\pi}} \mathscr{P}(x, \ddot{\pi}) \frac{\left[\partial_{\ddot{\pi}}(e^{-\ddot{\pi}x})\partial_x(v(x)e^{-\ddot{\pi}x})\right]}{\mathscr{P}(x, \ddot{\pi})} dx \tag{1.86}$$

To be noted that Eq.1.86 and Eq.1.84 have the same integral domain. Therefore, the estimations of F and $\partial_{\ddot{\pi}} F$ can be vectorized.
Finally, the analytical solution of its derivative is available for validation purposes:

$$\partial_{\ddot{\pi}} F = -\frac{-4\ddot{\pi}^2 e^{-2\ddot{\pi}^2} - e^{-2\ddot{\pi}^2} + 1}{\ddot{\pi}^2} \tag{1.87}$$

In the Fig.1.4, the analytical solution, estimations and approximations of $\partial_{\ddot{\pi}} F$ are plotted. The results are also listed in Table.1.6. Most of the time, if the value of the integral is estimated by the Monte Carlo method, the approximation of its sensitivity by the finite difference method is usually inaccurate.

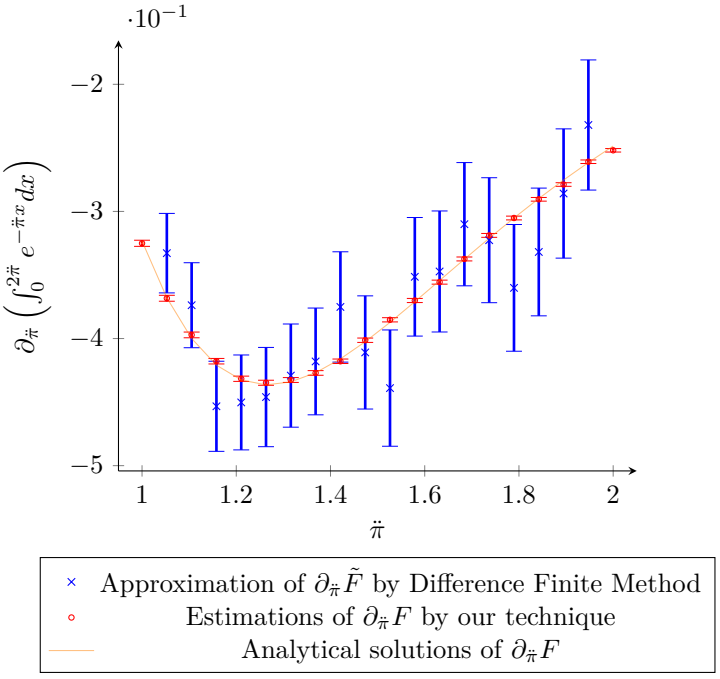


Figure 1.4 – A confidence interval of $\pm\sigma$ is shown in this figure. Each point of approximation requires two estimations of F, which includes 20000 realizations of random numbers. In contrast, each point of estimation requires only one estimation of $\partial_{\ddot{\pi}} F$, which includes 100000 realizations of random numbers. The calculating time of an approximation is, therefore, approximately two times the calculating time of estimation. However, even with a double calculation time, the standard deviation of the approximations is still huge compared to the estimations.

1.4 Modeling a radiative system

Similar to other physical systems, the modeling of a radiative system is based on several requirements: the equation that governs the physical interactions in the medium and on the

boundaries. In the following part of this chapter, we will establish the RTE, which governs the transport phenomena in the medium of a radiative system. After that, we will also show different boundary conditions in a radiative system.

1.4.1 Mesoscopic and corpuscular description of the radiation

There is more than one way to describe the physics of radiation. This thesis has a corpuscular vision of this physic [7]. More precisely, the description of the physics of radiation in this thesis is *mesoscopic* [8] (which is distinguished from the *microscopic* description [30] and *macroscopic* description [47]).

In the mesoscopic point of view, the descriptor of a system is the function of distribution: $f(\vec{x}, \vec{v}, t)$, of which the variables $\vec{x}$, $\vec{v}$, t are independent. They are defined in the domain of geometry $\vec{x} \in \mathscr{D}_{\mathbf{x}}$, the domain of velocity $\vec{v} \in \mathscr{D}_{\mathbf{v}}$ and t denotes the dimension of the time. It is considered continuous in $\mathscr{D}_{\mathbf{x}}$, $\mathscr{D}_{\mathbf{v}}$ and in time.

$f(\vec{x}, \vec{v}, t)$ represents the distribution of the particles in $\mathscr{D}_{\mathbf{x}}$ and $\mathscr{D}_{\mathbf{v}}$. Statistically, it can also be regarded as the probability of the number of photons in $\mathscr{D}_{\mathbf{x}}$ and $\mathscr{D}_{\mathbf{v}}$ at the moment of t. This statistical vision makes using statistical methods possible (such as the Monte Carlo method).

1.4.2 Radiative Transport Equation

In a system of radiation, $f_\nu(\vec{x}, \vec{v}, t)$ is used to describe the distribution of photons of which the frequency is [Hz].

Firstly, it is assumed that the photons do not interact with each other but only with the medium. The photons have a velocity of light c in the medium. The domain of velocity $\mathscr{D}_{\mathbf{v}}$ can be then referred to $\mathscr{D}_\Omega$ and the descriptor $f_\nu(\vec{x}, \vec{v}, t)$ can be referred to $f_\nu(\vec{x}, \vec{\omega}, t)$. Physically, it represents the number of photons at position $\vec{x}$, at the moment t and having a velocity c on the normalized vector of direction $\vec{\omega}$ (within an infinitesimal solid angle characterized by the vector of direction $\vec{\omega}$). Therefore, its unit is $[\mathrm{m}^{-3} \cdot \mathrm{sr}^{-1} \cdot \mathrm{s}^{-1} \cdot \mathrm{Hz}^{-1}]$. We can note that the value of $f_\nu(\vec{x}, \vec{\omega}, t)$ represents the number of photons on the position $\vec{x}$, following the direction $\vec{\omega}$ at the moment of t.

Radiative transport in transparent medium

Firstly, we consider a system of radiation with a transparent medium.

For an infinitive short period of time δt, a photon can travel $\vec{x} + c\vec{\omega}\delta t$ in $\mathscr{D}_{\mathbf{x}}$. For the transparent medium, Eq.1.88 can be established.

$$f_\nu(\vec{x}, \vec{\omega}, t) = f_\nu(\vec{x} + c\delta t\vec{\omega}, \vec{\omega}, t + \delta t) \tag{1.88}$$

We expend the right size of the Eq.1.88 around $\vec{x}$ and t:

$$f_\nu(\vec{x}, \vec{\omega}, t) = f_\nu(\vec{x}, \vec{\omega}, t) + c\delta t\vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}, t) + \partial_t f_\nu(\vec{x}, \vec{\omega}, t), \tag{1.89}$$

implying to:

$$\frac{\partial_t f_\nu(\vec{x}, \vec{\omega}, t)}{c\delta t} + \vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}, t) = 0. \tag{1.90}$$

The first term of Eq.1.90 represents the variation of the number of photons at different moments. In contrast, the second term represents the spatial derivative of the number of photons in the direction of propagation.

For most engineering questions, the radiative transfer is much faster than other physical phenomena, it is very often that we study it only in a stationary state, and its unit becomes $[\mathrm{m}^{-3} \cdot \mathrm{sr}^{-1} \cdot \mathrm{Hz}^{-1}]$. In other words, the distribution of the photons does not depend on time:

$$\vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}) = 0. \tag{1.91}$$

Radiative transport in semi-transparent medium

When the medium becomes semi-transparent, a photon can be absorbed, scattered, or emitted in the medium[18].

Absorption If the absorption is the only collision that happens in the medium, the transport of photons can be described by:

$$\vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}) = -k_{a,\nu} f_\nu(\vec{x}, \vec{\omega}) \tag{1.92}$$

$k_{a,\nu}$ $[\mathrm{m}^{-1}]$ is a property of the medium, called the *absorption coefficient*. Herein, the medium is homogeneous but can also depend on x in an inhomogeneous medium. It is defined as the inverse value of the *mean free path* of absorption (the average length traveling for a photon before being absorbed in the medium). In a purely absorbent medium, when the $k_{a,\nu}$ is large, we say that the medium is optically thick, which means that the photon will have more chance to be absorbed in a relatively short path. In the opposite case, the medium is called optically thin.

Scattering In this thesis, several assumptions are made to describe the physics of scattering [29, 54]:

- The polarization of the photon is not considered.
- The photon is conserved (not disappearing) when the scattering happens.
- The scattering of one photon does not influence the scattering of other photons.

If the scattering is the only collision that happens in the medium, the transport of photons can be described by:

$$\vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}) = -k_{d,\nu} f_\nu(\vec{x}, \vec{\omega}) + k_{d,\nu} \int_{\Omega'} \mathscr{P}(-\vec{\omega}' | -\vec{\omega}) d\vec{\omega}' f_\nu(\vec{x}, \vec{\omega}') \tag{1.93}$$

Similar to $k_{a,\nu}$, $k_{d,\nu}$ $[\mathrm{m}^{-1}]$ is called the *scattering coefficient*, defined as the inverse value of the *mean free path* of scattering. The first term of the right side of Eq.1.93 represents the extinction of photons when scattering happens in the medium, representing a loss term to the radiative balance. The second term of the right side of Eq.1.93 indicates the photons scattered to the direction $\vec{\omega}$, coming from all other directions, which is considered as a gain.

The function of $\mathscr{P}(-\vec{\omega}' | -\vec{\omega})$ is called the *Phase-function*. It characterizes the probability of the fact that a photon is scattered to the direction $-\vec{\omega}'$, coming from the direction $-\vec{\omega}$. This formulation favors a reciprocal/adjoint interpretation thanks to the micro-reversibility relation $\mathscr{P}(-\vec{\omega}' | -\vec{\omega}) = \mathscr{P}(\vec{\omega} | \vec{\omega}')$. For example, one of the common *Phase-functions* is the *isotropic Phase-function*:

$$\mathscr{P}_{isotropic}(-\vec{\omega}' | -\vec{\omega}) = \frac{1}{4\pi} \tag{1.94}$$

General case Besides the absorption and the scattering, the medium may also have the capacity to emit some photons (see Sec.1.4.3 for more details). The emission of photons is represented by a term of source S in this section. In order to characterize these three phenomena, the general case of radiative transport of photons can be described by Eq.1.95, of which S means the source of emission in the medium:

$$\vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}) = -k_{a,\nu} f_\nu(\vec{x}, \vec{\omega}) - k_{d,\nu} f_\nu(\vec{x}, \vec{\omega}) + \mathrm{S} + k_{d,\nu} \int_{\Omega'} \mathcal{P}(-\vec{\omega}' | -\vec{\omega}) d\vec{\omega}' f_\nu(\vec{x}, \vec{\omega}') \quad (1.95)$$

The right side of Eq.1.95 is also written as an operator $\mathcal{C}$:

$$\vec{\omega} \cdot \partial_1 f_\nu(\vec{x}, \vec{\omega}) = \mathcal{C}(f_\nu(\vec{x}, \vec{\omega})) + \mathrm{S} \quad (1.96)$$

$\mathcal{C}$ is called the collision operator. Physically, the left side of Eq.1.95 and Eq.1.96 represent the projection of the spatial gradient of the distribution function of photons on the direction of propagation. The right side represents the collisions and emissions of photons in the propagation direction, including absorption and scattering.

1.4.3 Transport of intensity

The corpuscular vision of radiation is proposed by *M.Planck* and *A.Einstein* at the beginning of the $20th$ century. In this vision, the photons are considered to have no mass, but they carry a quantity of energy E, which is a function of their frequency:

$$E = h\nu \quad (1.97)$$

where $h = 6.62607 \times 10^{-34} Js$ is called the *constant of Plank*.

Most of the time, we are more interested in the energy in a system of radiation than the number of photons. Therefore, a mesoscopic physical quantity I is defined, which is called the *intensity* (*Luminance* in French). It describes the energy flow crossing a perpendicular unit surface, which is proportional to the number of photons crossing this unit surface and their speed:

$$I_\nu(\vec{x}, \vec{\omega}) = h\nu c f_\nu(\vec{x}, \vec{\omega}) \quad (1.98)$$

Therefore, the unit of $I_\nu(\vec{x}, \vec{\omega})$ is $[\mathrm{W} \cdot \mathrm{m}^{-2} \cdot \mathrm{sr}^{-1} \cdot \mathrm{Hz}^{-1}]$.

We now use I_ν as the descriptor of the physical system of radiation. The transport of energy in the mesoscopic point of view in the medium can be described by:

$$\vec{\omega} \cdot \partial_1 I_\nu(\vec{x}, \vec{\omega}) = 0 \quad (1.99)$$

and

$$\vec{\omega} \cdot \partial_1 I_\nu(\vec{x}, \vec{\omega}) = \mathcal{C}(I_\nu(\vec{x}, \vec{\omega})) + S. \quad (1.100)$$

Eq.1.99 describes the transport of I_ν in transparent medium and Eq.1.100 in semi-transparent medium. In order to characterize the term of source S, the assumption of *local thermodynamic equilibrium* is needed.

Emission Under the assumption of *local thermodynamic equilibrium*, the emission of a point in the medium is based on its intrinsic properties and temperature, which is not influenced by other collisions and the incoming radiation on this point. There are some cases where it can not be applied. For example, when the medium is at low pressure.

Based on this assumption, we apply the *Planck's law* and the *Kirchhoff's law*.

Planck's law

All bodies of which the temperature is higher than $0K$ emit energy. According to the *Planck's law*, the monochromatic intensity I_ν emitted by a *blackbody* is a function of its temperature. It is called the monochromatic equilibrium intensity I_ν^{eq} (See Eq.1.101).

$$I_\nu^{eq}(T) = \frac{2h\nu^3}{c^2}\frac{1}{exp(\frac{h\nu}{k_B T}) - 1} \tag{1.101}$$

$k_B = 1.38065 \times 10^{-23} J \cdot K^{-1}$ is the *Boltzmann constant*.

Kirchhoff's law

When a system is in radiative equilibrium,

$$\alpha_{\nu,\vec{\omega}} = \epsilon_{\nu,\vec{\omega}}. \tag{1.102}$$

$\alpha_{\nu,\vec{\omega}}$ is the monochromatic and directional absorptivity and the $\epsilon_{\nu,\vec{\omega}}$ is the monochromatic and directional emissivity.

If we apply the assumption of *local thermodynamic equilibrium* to all points of the medium of which the field of temperature is inhomogeneous, following the *Planck's law* and the *Kirchhoff's law*, the term of source S in Eq.1.100 becomes:

$$\mathrm{S}(\vec{x}) = k_{a,\nu} I_\nu^{eq}\big(T(\vec{x})\big). \tag{1.103}$$

Finally, the general equation of RTE is developed:

$$\vec{\omega} \cdot \partial_1 I_\nu(\vec{x},\vec{\omega}) = -k_{a,\nu} I_\nu(\vec{x},\vec{\omega}) - k_{d,\nu} I_\nu(\vec{x},\vec{\omega}) + k_{a,\nu} I_\nu^{eq}\big(T(\vec{x})\big) + k_{d,\nu}\int_{\Omega'} \mathscr{P}(-\vec{\omega}'|-\vec{\omega}) d\vec{\omega}' I_\nu(\vec{x},\vec{\omega}') \tag{1.104}$$

However, the study of frequency ν and the heterogeneity of the medium are not the objectives of this thesis. We will use the homogeneous (properties of the medium do not depend on $\vec{x}$) version of the RTE and the integrated intensity (on wavelength) $I[\mathrm{W/m^2 sr}]$ in this thesis. Also, the medium will always be "cold" (no emission) in all the examples developed in this thesis:

$$\vec{\omega} \cdot \partial_1 I(\vec{x},\vec{\omega}) = -k_a I(\vec{x},\vec{\omega}) - k_d I(\vec{x},\vec{\omega}) + k_d \int_{\Omega'} \mathscr{P}(-\vec{\omega}'|-\vec{\omega}) d\vec{\omega}' I(\vec{x},\vec{\omega}') \tag{1.105}$$

1.4.4 Boundary conditions

The RTE demonstrated in this stage describes the interactions of the descriptor (intensity I) in the medium. In order to fully describe a radiative system, the boundary conditions are also needed.

On the boundary, noting $\mathcal{C}_b$ as the reflection operator and $\mathring{I}$ as the source of emission on the boundary, the incoming intensity equation is:

$$I(\vec{x},\vec{\omega}) = \mathcal{C}_b[I(\vec{x},\vec{\omega}')] + \mathring{I}(\vec{x},\vec{\omega}) \tag{1.106}$$

with

$$\mathcal{C}_b[I] = \rho(\vec{x},-\vec{\omega}) \int_{2\pi(\vec{n})} \mathscr{P}_b(-\vec{\omega}'|\vec{x},-\vec{\omega}) d\vec{\omega}' I(\vec{x},\vec{\omega}') \tag{1.107}$$

where $\vec{n}$ is the normal to the boundary at $\vec{x}$, $\vec{\omega}$ the direction within the hemisphere 2π characterised by $\vec{n}$, $\rho(\vec{x},-\vec{\omega})$ the surface reflectivity for a photon impacting the boundary in direction $-\vec{\omega}$, and $\mathscr{P}_b$ the probability density that the reflection direction is $-\vec{\omega}'$ for a photon reflected at $\vec{x}$ coming from direction $-\vec{\omega}$ (the product of $\rho\mathscr{P}_b$ is called BRDF). Eq.1.107 favors a reciprocal/adjoint interpretation thanks to the micro-reversibility relation $(\vec{\omega}\cdot\vec{n})\rho(\vec{x},-\vec{\omega})\mathscr{P}_b(-\vec{\omega}'|\vec{x},-\vec{\omega}) = -(\vec{\omega}'\cdot\vec{n})\rho(\vec{x},\vec{\omega})\mathscr{P}_b(\vec{\omega}'|\vec{x},\vec{\omega})$. The physical picture then becomes that of a photon initially in direction $-\vec{\omega}$ reflected in direction $-\vec{\omega}'$ (see Fig.1.5).

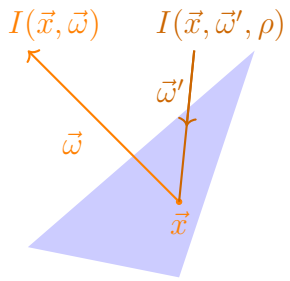


Figure 1.5 – Reflection of the intensity on the boundary.

For example, if the boundary is diffuse, the $\mathscr{P}_b$ corresponding is:

$$\mathscr{P}_b(-\vec{\omega}'|\vec{x},-\vec{\omega}) = \frac{|-\vec{\omega}'\cdot\vec{n}|}{\pi} \tag{1.108}$$

In this thesis, the boundaries are usually defined under some physical assumptions. We list some of them here:

1. The boundary is **black** means that the boundary is a black body. No intensity is reflected, but the boundary still emits intensity: $\rho(\vec{x},-\vec{\omega}) = 0$
2. The boundary is **cold** means that the boundary is not emitting intensity: $\mathring{I} = 0$.
3. The boundary is **Lambertian** means that the emitting intensity does not depend on the direction of emission: $\mathring{I}(\vec{x},\vec{\omega}) \equiv \mathring{I}(\vec{x})$.

1.4.5 Observable

In this stage, the RTE is established, and the boundary conditions are built. Therefore, the transport model of intensity for a radiative system is completed. An observable is a physical quantity that can be measured in the system. It can be simply the intensity I on a location $\vec{x}_{obs}$ towards a direction $\vec{\omega}_{obs}$: $I(\vec{x}_{obs},\vec{\omega}_{obs})$. Also, it can be some macroscopic descriptors. The transformation from the mesoscopic descriptor (intensity I) to different macroscopic descriptors is done by integrations.

Scalar flux G $[\mathbf{W \cdot m^{-2}}]$ The irradiance of a unit surface is the integral of I for all incoming solid angles:

$$G(\vec{x}_{obs}) = \int_{\Omega} I(\vec{x}_{obs}, \vec{\omega}) d\vec{\omega} \tag{1.109}$$

When we consider a 3D radiative system, $\Omega = 4\pi$.

Total incident irradiance ϕ $[\mathbf{W \cdot m^{-2}}]$ The radiance flux is the energy flux traveling perpendicularly to a unit surface:

$$\phi(\vec{x}_{obs}) = \int_{\Omega} I(\vec{x}_{obs}, \vec{\omega}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} \tag{1.110}$$

Impacting power P $[\mathbf{W}]$ The impacting power on a surface is the integral of radiance flux over all the surface:

$$P = \int_{\mathcal{S}} \int_{\Omega} I(\vec{x}, \vec{\omega}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} d\vec{x} \tag{1.111}$$

where $\mathcal{S}$ is the area of the surface being impacted.

1.5 Simulation of a radiative transfer problem

In this section, we solve a simple radiative problem by the Monte Carlo method. As an essential objective of this chapter, we discuss some algorithmic concepts (forward ray-tracing and reverse ray-tracing) and some physical concepts (ray path, path space, etc.). We consider a transparent medium with an emitter $\mathscr{E}$, which is a square, and a receiver $\mathscr{R}$, which is also a square (see Fig.1.6). The emitter $\mathscr{E}$ is "black" and the receiver $\mathscr{R}$ is "cold" and "black". We aim to determine the impacting power P[W] (the observable) on the receiver $\mathscr{R}$.

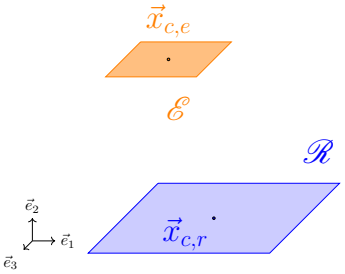


Figure 1.6 – Configuration of the radiative system. $\mathscr{E}$ is the emitter, its geometric centre is on $\vec{x}_{c,e} = (-1, 1, 0)$. $\mathscr{R}$ is the receiver, its geometric centre is on $\vec{x}_{c,r} = (0, -2.5, 0)$. The length of $\mathscr{E}$ is $l_e = l_0$ and the length of $\mathscr{R}$ is $l_r = 2l_0$.

1.5.1 Building the model

Firstly, a complete model is needed to solve the radiative problem. Referring to Eq.1.105, the radiative transport equation in a transparent medium is:

$$\vec{\omega} \cdot \partial_1 I(\vec{x}, \vec{\omega}) = 0 \tag{1.112}$$

where $\vec{x}$ is the position vector and $\vec{\omega}$ is the direction of propagation.

Referring to Eq.1.106 and the assumptions we made, the boundary conditions can be built:

$$\begin{cases} I(\vec{x}_e, \vec{\omega}_e) = I_0 \\ I(\vec{x}_r, \vec{\omega}_r) = 0 \end{cases} \tag{1.113}$$

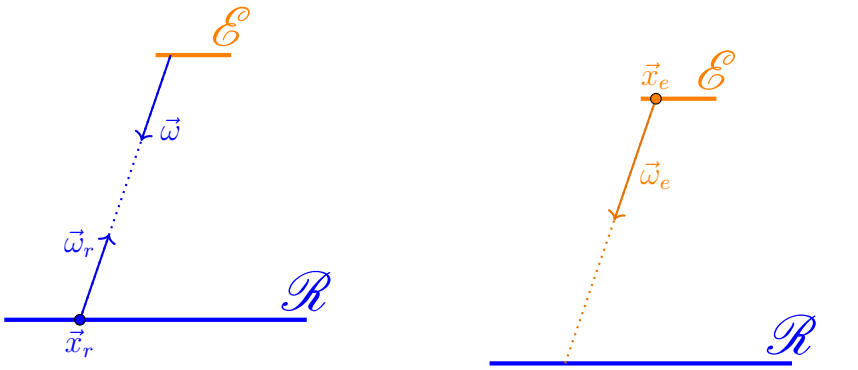
where the index e is used for the boundary condition on $\mathscr{E}$ and r is used for the boundary condition of $\mathscr{R}$.

1.5.2 Building the integral formulation

Our objective is the impacting power P[W] on the receiver $\mathscr{R}$. Referring to Eq.1.111, the impacting power can be formulated as:

$$P = \int_{\mathcal{S}_r} \int_{\Omega} I(\vec{x}_r, \vec{\omega}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} d\vec{x}_r \tag{1.114}$$

where $\mathcal{S}_r$ is the area of the surface of $\mathscr{R}$ and $\Omega = 2\pi$. The intensity in Eq.1.114 is the incoming intensity on the position $\vec{x}_r$, following the direction $-\vec{\omega} \equiv \vec{\omega}_r$, seeing Fig.1.7a. Therefore, we can launch a ray path from $\vec{x}_r$, following the direction $\vec{\omega}_r$. If the ray hits the emitter $\mathscr{E}$, the intensity $I(\vec{x}_r, \vec{\omega}) = I_0$. Otherwise, $I(\vec{x}_r, \vec{\omega}) = 0$.



(a) Reverse ray-tracing **(b)** Forward ray-tracing

Figure 1.7 – Reverse and Forward ray-tracing. The forward ray-tracing algorithm searches where the photon is going (from the emitter). The backward ray-tracing algorithm searches where the photon comes from (to the receiver).

This ray-tracing test can be expressed mathematically as a Heaviside function:

$$P = \int_{\mathcal{S}_r} \int_{\Omega} I_0 \mathcal{H}(\vec{x}_{hit} \in \mathscr{E}) |\vec{\omega}_r \cdot \vec{n}| d\vec{\omega}_r d\vec{x}_r \tag{1.115}$$

where $\vec{x}_{hit}$ is the first point of intersection of the ray staring from $\vec{x}_r$, towards $\vec{\omega}_r$ (or $-\vec{\omega}$). The Heaviside function $\mathcal{H}(\vec{x}_{hit} \in \mathscr{E})$ will be set to 1 when $\vec{x}$ is on $\mathscr{E}$. Otherwise, it will be set to 0.

We then reformulate P into an expected value form to be solved by the Monte Carlo method:

$$P = \int_{\mathcal{S}_r} \mathcal{P}_{X_r}(\vec{x}_r) d\vec{x}_r \int_{\Omega} \mathcal{P}_{\Omega_r}(\vec{\omega}_r) d\vec{\omega}_r \left\{ I_0 \mathcal{H}(\vec{x}_{hit} \in \mathscr{E}) \mathcal{S}_r \pi \right\} \tag{1.116}$$

where $\mathscr{P}_{X_r}$ and $\mathscr{P}_{\Omega_r}$ are the pdfs:

$$\mathscr{P}_{X_r}(\vec{x}_r) = \frac{1}{\mathcal{S}_r}; \mathscr{P}_{\Omega_r}(\vec{\omega}_r) = \frac{|\vec{\omega}_r \cdot \vec{n}|}{\pi} \tag{1.117}$$

Finally, if we note the estimation of P by the Monte Carlo method as $\check{P}$, it is then given by:

$$\check{P} = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(I_0 \mathscr{H}(\vec{x}_{hit,j} \in \mathscr{E}) \mathcal{S}_r \pi\right) = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(\hat{w}_j\right). \tag{1.118}$$

where $\vec{x}_{hit,j}$ and $\hat{w}_j$ refer to the j^{th} Monte Carlo realization. Algorithmically, for each Monte Carlo realization:

1. we sample a position on $\mathscr{R}$ based on $\mathscr{P}_{X_r}$.
2. we sample a direction $\vec{\omega}_r$ in the hemisphere characterised by $\vec{n}$, based on $\mathscr{P}_{\Omega_r}$.
3. we launch a ray-tracing test from the point $\vec{x}_r$ towards $\vec{\omega}_r$.
4. if the first intersection point $\vec{x}_{hit}$ is found on $\mathscr{E}$, $\hat{w}_j = I_0 \mathcal{S}_r \pi$.
5. otherwise, $\hat{w}_j = 0$.

This algorithm is also called reverse ray-tracing. There is another possibility to construct the algorithm called forward ray-tracing.

1.5.3 Forward ray-tracing and reverse ray-tracing

In the algorithm of reverse ray-tracing, the rays are launched from the receiver $\mathscr{R}$. On the contrary, the rays are launched from the emitter (the source) $\mathscr{E}$ in the forward ray-tracing algorithm. Following the definition of a solid angle, we have the following:

$$d\vec{\omega}_r = \frac{|\vec{\omega}_r \cdot \vec{n}| d\vec{x}_e}{r^2}; d\vec{\omega}_e = \frac{|\vec{\omega}_e \cdot \vec{n}| d\vec{x}_r}{r^2} \tag{1.119}$$

where r is the distance between $\vec{x}_r$ and $\vec{x}_e$.

We substitute Eq.1.119 into Eq.1.115 and we have:

$$P = \int_{\mathcal{S}_e} \int_{\Omega} I_0 \mathscr{H}(\vec{x}'_{hit} \in \mathscr{R}) |\vec{\omega}_e \cdot \vec{n}| d\vec{\omega}_e d\vec{x}_e \tag{1.120}$$

where $\vec{x}'_{hit}$ is the first point of intersection of the ray staring from $\vec{x}_e$, towards $\vec{\omega}_e$ and $\mathcal{S}_e$ is the area of the surface of emitter $\mathscr{E}$.

The corresponding expected value form of the observable P will be:

$$P = \int_{\mathcal{S}_e} \mathscr{P}_{X_e}(\vec{x}_e) d\vec{x}_e \int_{\Omega} \mathscr{P}_{\Omega_e}(\vec{\omega}_e) d\vec{\omega}_e \left\{I_0 \mathscr{H}(\vec{x}'_{hit} \in \mathscr{R}) \mathcal{S}_e \pi\right\} \tag{1.121}$$

where $\mathscr{P}_{X_r}$ and $\mathscr{P}_{\Omega}$ are the pdfs:

$$\mathscr{P}_{X_e}(\vec{x}_e) = \frac{1}{\mathcal{S}_e}; \mathscr{P}_{\Omega_e}(\vec{\omega}_e) = \frac{|\vec{\omega}_e \cdot \vec{n}|}{\pi} \tag{1.122}$$

Finally, if we note the estimation of P by the Monte Carlo method with a forward ray-tracing algorithm as $\check{P}'$ ($\check{P}$ for the reverse tray-tracing algorithm), it is then given by:

$$\check{P}' = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(I_0 \mathcal{H}(\vec{x}'_{hit,j} \in \mathscr{E}) \mathcal{S}_e \pi \right) = \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} \left(\hat{w}_j \right). \tag{1.123}$$

where $\vec{x}'_{hit,j}$ and $\hat{w}_j$ refer to the j^{th} Monte Carlo realization. Algorithmically, for each Monte Carlo realization:

1. we sample a position on $\mathscr{E}$ based on $\mathcal{P}_{X_e}$.
2. we sample a direction $\vec{\omega}_e$ in the hemisphere characterised by $\vec{n}$, based on $\mathcal{P}_{\Omega_e}$.
3. we launch a ray-tracing test from the point $\vec{x}_e$ towards $\vec{\omega}_e$.
4. if the first intersection point $\vec{x}'_{hit}$ is found on $\mathscr{R}$, $\hat{w}_j = I_0 \mathcal{S}_e \pi$.
5. otherwise, $\hat{w}_j = 0$.

The impacting power P can be estimated using the forward or reverse ray-tracing algorithm. Both of these algorithms suggest their own physical image (Fig.1.7). For the forward ray-tracing algorithm, we try to follow "all" the photons emitted from the emitter to see if they arrive at the receiver. As for the reverse ray-tracing algorithm, we try to observe "all" the incoming directions on all receivers to see if photons are coming from the emitter.

reverse ray-tracing		forward ray-tracing	
$\check{P}$	$\sigma[\check{P}]$	$\check{P}'$	$\sigma[\check{P}']$
253.69×10^{-3}	1.77×10^{-3}	253.62×10^{-3}	855.83×10^{-6}

Table 1.1 – Results of estimations of reverse ray-tracing algorithm and forward ray-tracing algorithms, $n_{MC} = 1000000$.

The results of estimations for the two algorithms are compared in Table.1.1. The two results have almost the same value estimated for the impacting power. However, the corresponding standard deviations are different. The choice of inverse or forward ray-tracing algorithm significantly impacts the convergence performance. In this specific case, the convergence performance of the forward ray-tracing algorithm is better than that of reverse ray-tracing because the receiver $\mathscr{R}$ has a higher area than the emitter $\mathscr{E}$. When we launch the rays from the receiver $\mathscr{R}$, most of them will not hit the emitter $\mathscr{E}$, and the Monte Carlo weight $\hat{w}_j$ is often null, resulting in a high standard deviation. On the contrary, when we launch the rays from the emitter $\mathscr{E}$ following the forward ray-tracing algorithm, there will be more chance that the rays can hit the receiver $\mathscr{R}$, leading to a better convergence performance.

To sum up, the size of the receiver and the sources (the emitter) directly influence the ray-tracing algorithm's convergence performance. In the following parts of this thesis, we will see some sources on a segment or point. In this case, we prefer the forward ray-tracing algorithm (sampling from the sources), or we can propose a special treatment to perform the estimation.

1.5.4 Integral domains, path spaces, samplings, and vectorization

It is noticed that when an integral formulation is formulated (especially in an expected value form), a clear physical image and a Monte Carlo algorithm are attached to it. An integral formulation has three interpretations in mathematics, physics, and algorithmic.

Here, we take the example of the forward ray-tracing algorithm (Eq.1.121). Mathematically, we formulate a product of a spatial integral (about $\vec{x}_e$) and an angular integral (about $\vec{\omega}_e$), where the integral domain is on all the surface of $\mathscr{E}$ and over 2π around the normal of the surface $\vec{n}$.

Physically, we define path spaces, where the rays are launched, starting from all points on the emitter $\mathscr{E}$, towards all directions over 2π around $\vec{n}$.

From an algorithmic point of view, we construct an algorithm where we sample a position on the surface of $\mathscr{E}$ and then a direction over 2π around $\vec{n}$.

Then we create the following correspondence:

$$\textit{integral domain} \iff \textit{path space} \iff \textit{sampling}$$

This is important as part of this thesis since we practice the idea of "vectorization". As already introduced in Section.1.3.5, the Monte Carlo estimations of integrals can be vectorized when they have a common integral domain, and the same set of sampling random numbers can be used to estimate them. Now, when we solve the radiative problems by the Monte Carlo method, we say that if we have the same path space for different observables, their estimations can be vectorized. The same sampling set (for example, positions and directions) can then be used for these estimations.

An example is herein given: the receiver $\mathscr{R}$ is separated into $\mathscr{R}_1$ and $\mathscr{R}_2$. Now, we aim to estimate two observable P_1 and P_2, corresponding to the impacting power on $\mathscr{R}_1$ and $\mathscr{R}_2$ respectively (Fig.1.8). Their estimations can be easily vectorized:

$$\begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \int_{S_e} \mathscr{P}_{X_e}(\vec{x}_e) d\vec{x}_e \int_{\Omega} \mathscr{P}_{\Omega_e}(\vec{\omega}_e) d\vec{\omega}_e \left\{ \begin{bmatrix} I_0 \mathscr{H}(\vec{x}'_{hit} \in \mathscr{R}_1) \\ I_0 \mathscr{H}(\vec{x}'_{hit} \in \mathscr{R}_2) \end{bmatrix} \mathcal{S}_e \pi \right\} \tag{1.124}$$

where the pdf $\mathscr{P}_{X_e}$ and $\mathscr{P}_{\Omega_e}$ are already defined in Eq.1.122.

From an algorithmic point of view, we sample a position on $\mathscr{E}$ and a direction on 2π around $\vec{n}$. After that, we launch a ray following the sampled $\vec{x}_e$ and $\vec{\omega}_e$, to see if the ray intersects $\mathscr{R}_1$, $\mathscr{R}_2$ or neither of them. Therefore, we say that mathematically, the estimations of the two observables (P_1 and P_2) are based on the same integral domain. Physically, the estimations are based on the same path space. From an algorithmic point of view, the estimations use only one set of sampling (for $\vec{x}_e$ and $\vec{\omega}_e$).

1.6 Conclusion of the chapter

After a quick review of basic statistics, we have introduced the Monte Carlo method, which is used to estimate an integral in an expected value form. The result is always given with a confidence interval that relies on variance. When the variance is too high, reduction variance techniques such as *importance sampling* can be applied. When several expected values have to be estimated simultaneously, the *vectorization* technique can sometimes be implemented to reuse the set of random numbers. This method can be applied when the expected values

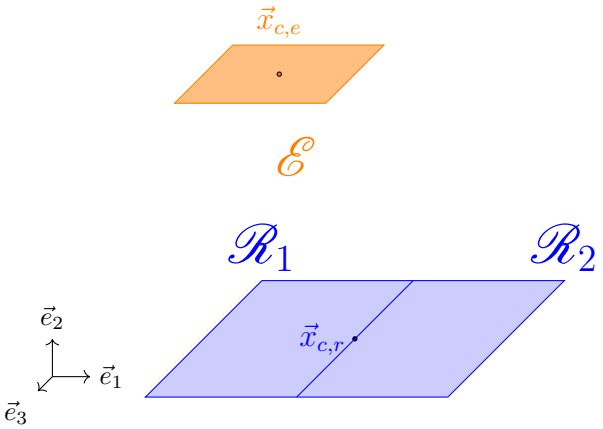


Figure 1.8 – Same configuration of Fig.1.6 but the receiver is split into two parts.

share the same pdf. By reducing random number generation, the computation time is greatly improved.

Finite differences can naively achieve approximating sensitivities or derivatives. It relies on estimating two expected values with an infinitesimal variation of a chosen parameter. This inefficient approach results in biased sensitivity estimation with a very high variance. We have shown that direct differentiation of the expected value is much more appropriate since the estimation is unbiased and gives an acceptable error bar.

In the second part of this chapter, the physics of radiative transfer is introduced with a corpuscular and mesoscopic point of view. The physics of radiative transfer in the medium is described by RTE, using the intensity as the descriptor. The boundary condition governs the transport of intensity on the boundaries. In order to build a Monte Carlo algorithm to estimate a physical observable, the observable has to be reformulated as an integral in an expected value form.

An example of solving radiative transfer problem by Monte Carlo method is given at the end. Similarly to the reciprocity of rays, the Monte Carlo algorithm can be built differently. It can be built from the emitter to the receiver or, inversely, from the receiver to the emitter. These two strategies refer to the forward and reverse ray-tracing algorithms. Last but not least, we link the vocabulary of integral formulations, physics, and algorithms to prepare further studies about *vectorization* in the radiative systems in the following chapters.

1.7 Numerical results

n_{MC}	1000000	788046	621017	489390	385662
$\check{Y}$	393.45×10^{-3}	393.54×10^{-3}	393.46×10^{-3}	393.46×10^{-3}	393.22×10^{-3}
$\sigma[\check{Y}]$	57×10^{-6}	64×10^{-6}	72×10^{-6}	81×10^{-6}	91×10^{-6}
n_{MC}	303920	239503	188739	148735	117210
$\check{Y}$	393.37×10^{-3}	393.54×10^{-3}	393.41×10^{-3}	393.46×10^{-3}	393.39×10^{-3}
$\sigma[\check{Y}]$	103×10^{-6}	116×10^{-6}	131×10^{-6}	147×10^{-6}	166×10^{-6}
n_{MC}	92367	72790	57362	45204	35622
$\check{Y}$	393.55×10^{-3}	393.56×10^{-3}	393.13×10^{-3}	393.61×10^{-3}	394.04×10^{-3}
$\sigma[\check{Y}]$	187×10^{-6}	210×10^{-6}	236×10^{-6}	268×10^{-6}	301×10^{-6}
n_{MC}	28072	22122	17433	13738	10826
$\check{Y}$	393.29×10^{-3}	393.09×10^{-3}	392.88×10^{-3}	393.46×10^{-3}	392.48×10^{-3}
$\sigma[\check{Y}]$	340×10^{-6}	381×10^{-6}	429×10^{-6}	483×10^{-6}	544×10^{-6}
n_{MC}	8532	6723	5298	4175	3290
$\check{Y}$	393.48×10^{-3}	392.49×10^{-3}	393.40×10^{-3}	393.18×10^{-3}	393.84×10^{-3}
$\sigma[\check{Y}]$	616×10^{-6}	696×10^{-6}	777×10^{-6}	883×10^{-6}	985×10^{-6}
n_{MC}	2593	2043	1610	1269	1000
$\check{Y}$	392.93×10^{-3}	390.59×10^{-3}	393.97×10^{-3}	392.89×10^{-3}	394.52×10^{-3}
$\sigma[\check{Y}]$	1.12×10^{-3}	1.24×10^{-3}	1.41×10^{-3}	1.57×10^{-3}	1.79×10^{-3}

Table 1.2 – Expected values and variances of estimations

n_{MC}	1000000	788046	621017	489390	385662
$\check{Y}^*$	393.47×10^{-3}	393.53×10^{-3}	393.50×10^{-3}	393.49×10^{-3}	393.48×10^{-3}
$\sigma[\check{Y}^*]$	21×10^{-6}	24×10^{-6}	26×10^{-6}	30×10^{-6}	34×10^{-6}
n_{MC}	303920	239503	188739	148735	117210
$\check{Y}^*$	393.43×10^{-3}	393.47×10^{-3}	393.62×10^{-3}	393.51×10^{-3}	393.57×10^{-3}
$\sigma[\check{Y}^*]$	38×10^{-6}	43×10^{-6}	48×10^{-6}	54×10^{-6}	61×10^{-6}
n_{MC}	92367	72790	57362	45204	35622
$\check{Y}^*$	393.50×10^{-3}	393.51×10^{-3}	393.64×10^{-3}	393.37×10^{-3}	393.52×10^{-3}
$\sigma[\check{Y}^*]$	69×10^{-6}	77×10^{-6}	87×10^{-6}	98×10^{-6}	111×10^{-6}
n_{MC}	28072	22122	17433	13738	10826
$\check{Y}^*$	393.32×10^{-3}	393.35×10^{-3}	393.38×10^{-3}	393.75×10^{-3}	393.27×10^{-3}
$\sigma[\check{Y}^*]$	124×10^{-6}	139×10^{-6}	158×10^{-6}	178×10^{-6}	200×10^{-6}
n_{MC}	8532	6723	5298	4175	3290
$\check{Y}^*$	393.19×10^{-3}	393.54×10^{-3}	393.69×10^{-3}	393.62×10^{-3}	393.29×10^{-3}
$\sigma[\check{Y}^*]$	222×10^{-6}	253×10^{-6}	287×10^{-6}	327×10^{-6}	362×10^{-6}
n_{MC}	2593	2043	1610	1269	1000
$\check{Y}^*$	392.24×10^{-3}	393.77×10^{-3}	393.78×10^{-3}	393.76×10^{-3}	393.28×10^{-3}
$\sigma[\check{Y}^*]$	395×10^{-6}	466×10^{-6}	532×10^{-6}	599×10^{-6}	666×10^{-6}

Table 1.3 – Expected values and standard deviation of estimations with the important sampling technique.

n_{MC}	1000000	788046	621017	489390	385662
$\check{Y}_1$	631.95×10^{-3}	632.19×10^{-3}	631.78×10^{-3}	631.97×10^{-3}	631.66×10^{-3}
$\sigma[\check{Y}_1]$	181×10^{-6}	204×10^{-6}	230×10^{-6}	259×10^{-6}	292×10^{-6}
n_{MC}	303920	239503	188739	148735	117210
$\check{Y}_1$	631.94×10^{-3}	631.78×10^{-3}	632.20×10^{-3}	631.62×10^{-3}	631.96×10^{-3}
$\sigma[\check{Y}_1]$	328×10^{-6}	370×10^{-6}	417×10^{-6}	469×10^{-6}	529×10^{-6}
n_{MC}	92367	72790	57362	45204	35622
$\check{Y}_1$	632.58×10^{-3}	632.82×10^{-3}	632.58×10^{-3}	632.23×10^{-3}	633.34×10^{-3}
$\sigma[\check{Y}_1]$	596×10^{-6}	671×10^{-6}	756×10^{-6}	850×10^{-6}	957×10^{-6}
n_{MC}	28072	22122	17433	13738	10826
$\check{Y}_1$	632.53×10^{-3}	630.54×10^{-3}	631.85×10^{-3}	632.78×10^{-3}	631.62×10^{-3}
$\sigma[\check{Y}_1]$	1.08×10^{-3}	1.21×10^{-3}	1.37×10^{-3}	1.54×10^{-3}	1.75×10^{-3}
n_{MC}	8532	6723	5298	4175	3290
$\check{Y}_1$	635.20×10^{-3}	631.89×10^{-3}	629.34×10^{-3}	634.24×10^{-3}	633.51×10^{-3}
$\sigma[\check{Y}_1]$	1.97×10^{-3}	2.21×10^{-3}	2.46×10^{-3}	2.79×10^{-3}	3.15×10^{-3}
n_{MC}	2593	2043	1610	1269	1000
$\check{Y}_1$	635.53×10^{-3}	627.10×10^{-3}	641.86×10^{-3}	624.09×10^{-3}	626.76×10^{-3}
$\sigma[\check{Y}_1]$	3.58×10^{-3}	3.95×10^{-3}	4.62×10^{-3}	4.96×10^{-3}	5.64×10^{-3}

Table 1.4 – Expected values and standard deviation of estimations with the vectorization technique for $\check{Y}_1$.

n_{MC}	1000000	788046	621017	489390	385662
$\check{Y}$	393.42×10^{-3}	393.49×10^{-3}	393.36×10^{-3}	393.41×10^{-3}	393.32×10^{-3}
$\sigma[\check{Y}]$	57×10^{-6}	64×10^{-6}	72×10^{-6}	81×10^{-6}	91×10^{-6}
n_{MC}	303920	239503	188739	148735	117210
$\check{Y}$	393.42×10^{-3}	393.36×10^{-3}	393.49×10^{-3}	393.32×10^{-3}	393.41×10^{-3}
$\sigma[\check{Y}]$	103×10^{-6}	116×10^{-6}	131×10^{-6}	147×10^{-6}	166×10^{-6}
n_{MC}	92367	72790	57362	45204	35622
$\check{Y}$	393.61×10^{-3}	393.69×10^{-3}	393.62×10^{-3}	393.52×10^{-3}	393.88×10^{-3}
$\sigma[\check{Y}]$	187×10^{-6}	210×10^{-6}	237×10^{-6}	266×10^{-6}	300×10^{-6}
n_{MC}	28072	22122	17433	13738	10826
$\check{Y}$	393.59×10^{-3}	392.99×10^{-3}	393.38×10^{-3}	393.70×10^{-3}	393.25×10^{-3}
$\sigma[\check{Y}]$	338×10^{-6}	380×10^{-6}	429×10^{-6}	482×10^{-6}	549×10^{-6}
n_{MC}	8532	6723	5298	4175	3290
$\check{Y}$	394.43×10^{-3}	393.39×10^{-3}	392.66×10^{-3}	394.18×10^{-3}	393.94×10^{-3}
$\sigma[\check{Y}]$	615×10^{-6}	692×10^{-6}	771×10^{-6}	874×10^{-6}	985×10^{-6}
n_{MC}	2593	2043	1610	1269	1000
$\check{Y}$	394.52×10^{-3}	391.98×10^{-3}	396.38×10^{-3}	391.08×10^{-3}	391.87×10^{-3}
$\sigma[\check{Y}]$	1.12×10^{-3}	1.24×10^{-3}	1.44×10^{-3}	1.56×10^{-3}	1.77×10^{-3}

Table 1.5 – Expected values and standard deviation of estimations with the vectorization technique for $\check{Y}_2$.

$\ddot{\pi}$	$\partial\tilde{F}(\ddot{\pi})$	$\sigma[\partial\tilde{F}(\ddot{\pi})]$	$\partial F(\ddot{\pi})$	$\sigma[\partial F(\ddot{\pi})]$	analytical solution
2.00			-251.91×10^{-3}	1.37×10^{-3}	-248.57×10^{-3}
1.95	-232.00×10^{-3}	-51.19×10^{-3}	-260.91×10^{-3}	1.39×10^{-3}	-261.53×10^{-3}
1.89	-285.89×10^{-3}	-50.85×10^{-3}	-278.83×10^{-3}	1.41×10^{-3}	-275.29×10^{-3}
1.84	-331.89×10^{-3}	-50.25×10^{-3}	-290.47×10^{-3}	1.44×10^{-3}	-289.85×10^{-3}
1.79	-360.14×10^{-3}	-49.85×10^{-3}	-305.18×10^{-3}	1.48×10^{-3}	-305.15×10^{-3}
1.74	-322.65×10^{-3}	-49.16×10^{-3}	-318.86×10^{-3}	1.52×10^{-3}	-321.11×10^{-3}
1.68	-310.05×10^{-3}	-48.48×10^{-3}	-337.39×10^{-3}	1.54×10^{-3}	-337.58×10^{-3}
1.63	-347.24×10^{-3}	-47.61×10^{-3}	-355.63×10^{-3}	1.58×10^{-3}	-354.33×10^{-3}
1.58	-351.43×10^{-3}	-46.64×10^{-3}	-370.07×10^{-3}	1.62×10^{-3}	-371.04×10^{-3}
1.53	-438.99×10^{-3}	-45.75×10^{-3}	-385.26×10^{-3}	1.66×10^{-3}	-387.29×10^{-3}
1.47	-410.94×10^{-3}	-44.52×10^{-3}	-401.33×10^{-3}	1.72×10^{-3}	-402.51×10^{-3}
1.42	-375.15×10^{-3}	-43.40×10^{-3}	-417.88×10^{-3}	1.76×10^{-3}	-416.00×10^{-3}
1.37	-418.01×10^{-3}	-42.00×10^{-3}	-427.06×10^{-3}	1.83×10^{-3}	-426.88×10^{-3}
1.32	-429.15×10^{-3}	-40.59×10^{-3}	-432.62×10^{-3}	1.90×10^{-3}	-434.10×10^{-3}
1.26	-446.04×10^{-3}	-39.05×10^{-3}	-434.78×10^{-3}	1.98×10^{-3}	-436.46×10^{-3}
1.21	-450.26×10^{-3}	-37.34×10^{-3}	-431.64×10^{-3}	2.07×10^{-3}	-432.58×10^{-3}
1.16	-453.30×10^{-3}	-35.44×10^{-3}	-417.82×10^{-3}	2.17×10^{-3}	-420.94×10^{-3}
1.11	-373.79×10^{-3}	-33.44×10^{-3}	-397.15×10^{-3}	2.25×10^{-3}	-399.95×10^{-3}
1.05	-332.83×10^{-3}	-31.27×10^{-3}	-368.31×10^{-3}	2.33×10^{-3}	-367.95×10^{-3}
1.00			-325.07×10^{-3}	2.41×10^{-3}	-323.32×10^{-3}

Table 1.6 – Comparison of the results obtained by finite difference and Monte Carlo method.

2

Objective and research problem

2.1 Introduction

In the last chapter, we introduced some necessary elements of radiative transfer and the Monte Carlo method. Herein, we provide a more specific context about the realized work in this thesis.

Firstly, a literature study about the need to estimate sensitivities in a radiative system will be introduced. This literature study mainly focuses on the field of CSP, which is the starting point of this thesis. Monte Carlo methods are widely used to compute optical efficiency in CSP research. Indeed, deterministic methods are not used to compute optical efficiencies of CSP system due to complex geometries. However, evaluating sensitivity with respect to the geometric parameters (translation, rotation, and heliostat size) remains an open research problem. We will see in this chapter that our objective is to develop a tool to estimate this kind of geometric sensitivities in CSP systems.

Researchers in EDstar group[1] and image synthesis have focused on sensitivity estimation for years. Therefore, we review the available tools in the physics of radiative transfer and image synthesis that deal with estimating sensitivities by the Monte Carlo method. Difficulties are found when we want to apply these tools to estimate sensitivities in CSP system or, more generally, in a system with complex (non-convex) geometry. These difficulties are thoroughly distinguished and discussed in this chapter. The research problem we address is to overcome these difficulties by using the method of sensitivity model, which is relevant to estimate sensitivities in a radiative system with complex geometry. In addition, we investigate the technique of vectorization, which becomes crucial when the number of sensitivities to estimate increases. We will also introduce and discuss why we chose to follow this strategy.

[1] http://www.edstar.cnrs.fr/prod/fr/

2.2 Needs of estimating sensitivities by Monte Carlo method in CSP systems

2.2.1 Optical system of CSP

The thermodynamic production of electricity by solar energy can be dated back to the 1980s [43]. Nowadays, the technology of CSP is also used for heat production for industrial processes [49]. An optical system is designed to concentrate the solar radiance to attain a high temperature on the solar receiver. The energy concentrated on the receiver is then transformed into electricity, stored in chemical energy, or used by industrial processes.

We distinguish here two prominent families of concentration technology:

The linear concentrators Solar radiance is concentrated on a long tube. Heat transfer fluid flows inside the tube and carries the concentrated energy to the applications. The tubes are set on the focal line of the reflectors concentrating solar radiance. The Linear Fresnel reflector and the Parabolic trough reflectors (Fig.2.1) are in this category.

The point-focus concentrators Solar radiance is concentrated on a point where a receiver is located. The concentrators follow the sun in a day by two-axis rotations: azimuth and elevation. The technology of STP and the parabolic dish (Fig.2.1) are in this category.

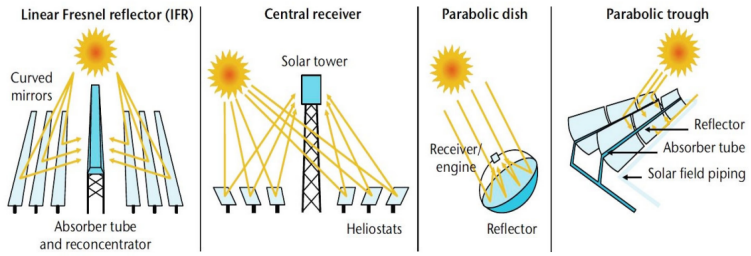


Figure 2.1 – The four main CSP technologies. Parabolic through and linear Fresnel reflector are linear concentrators. Parabolic dish and central receiver (STP) are point-focus concentrators.

2.2.2 Monte Carlo method in CSP

If we focus on the optical system of CSP facilities (Fig.2.1), it is firstly based on radiative transfer phenomena. Researchers focus on how to evaluate the optical performances of the optical system. For example, how much power (noted P in [W]) can a receiver capture from the reflected solar power at a given time during the day?

There are two main families of numerical tools in the CSP community [62], evaluating the optical performances and answering this question:

1. MCRT method [3, 5, 35, 48, 63].
2. Cone optics convolution-based method [12, 39, 53, 59, 60].

Cone optics convolution-based method

The first codes following this concept can be dated back to the 1970s in the project of Solar One in US [19]. Following this method, the mirror shape, optical errors, and sun shape are represented by analytical distribution, usually Gaussian distributions. After that,

the reflected image on the receiver is obtained by the convolution of these three analytical distributions.

Researchers use the codes following this concept mainly for faster optical modeling. Some codes remain very popular (HFLCAL [53], and SolarPILOT [60], etc). However, the accuracy of this kind of method is generally lower than the MCRT methods [62].

MCRT method

Some articles review and compare different tools of simulation [19, 36, 37, 62]. Compared to the other method, MCRT method is justified to be more versatile [19] because the real points of interactions of photons in complex geometries can be produced. Several Monte-Carlo-based tools are compared to the convolution-based methods in the study of [62], revealing that the Monte Carlo methods have better performances concerning accuracy.

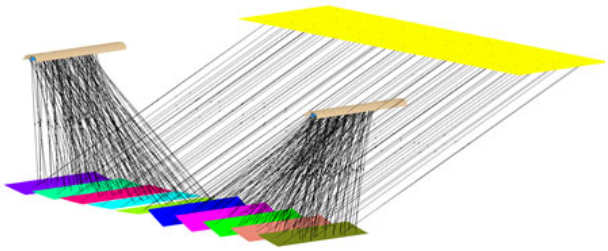


Figure 2.2 – SolTrace uses a MCRT method [63], shown here being applied to a linear Fresnel CSP configuration. Rays are randomly launched from the plane above reflectors.

Moreover, different MCRT methods can be further classified:

- Collision-based algorithms
- Integral Formulation Monte Carlo (IFMC) [11]

Collision-based algorithms The Collision-based algorithms are also called “analog Monte Carlo” in a Monte Carlo point of view [14]. They mimic the physical phenomena of photon transport. The most widely used codes in this category includes MIRVAL [35], Tonatiuh [3], SolTrace [63], Traceri [61], etc.

This kind of MCRT method has similar algorithms, which are also commonly labeled the algorithms of “Russian roulette”. Starting from a plane above all reflectors, some solar rays are launched according to the solar position. The rays are associated with a random number in the range of 0 to 1. Rays will be reflected when the random number is higher than the reflectivity; otherwise, they are absorbed by the surface. It is then tested if the rays are reflected and hit the receiver (Fig.2.2). The impacting power P[W] on the receiver is then estimated according to the portion of solar rays arriving at the receiver.

Since the rays are launched from the source of intensity, the plane above heliostats, those algorithms are regarded as forward ray-tracing algorithms (Section.1.5.3 in Chapter.1).

IFMC In the framework of IFMC, the impacting power P[W] on the receiver is written as an integral formulation based on the physics of radiative transfer, and more specifically on the RTE. It is also the aspect of working in our research group [11], corresponding to one of the spirits of this thesis, which is "formulating the integral formulation and solving it by Monte Carlo method". In our group, based on this approach, a numerical tool, SOLSTICE, has been developed by Mesostar[2]. The algorithm of SOLSTICE is very efficient compared to the Collision-based algorithms [62] because the techniques of variance reduction are used to construct the corresponding Monte Carlo algorithm [14]. Some of them are introduced in Chapter.1.

For example, instead of launching solar rays from a plane over the reflectors, SOLSTICE launches the solar rays from the surfaces of reflectors according to the solar cone (Fig.2.3). It avoids launching massive solar rays hitting the ground but not the reflectors [9]. Referring to the example shown in Section.1.4 in Chapter.1, the Monte Carlo weight is more often null for collision-based algorithms, resulting that SOLSTICE converges faster than the collision-based algorithms [62].

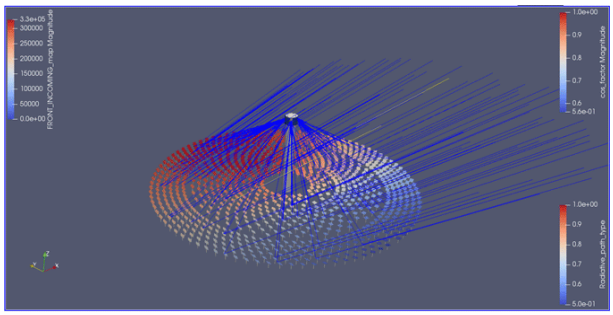


Figure 2.3 – The SOLSTICE algorithm relies on the first sampling of a position on the reflector, followed by sampling in a direction in the solar cone to calculate the reflected rays and check if it intercepts the receiver.

The highlight of IFMC methods is that the observable (usually it is the power P) is expressed as an integral formulation with an expected value form. The corresponding Monte Carlo algorithm is then built based on this integral formulation to estimate the observable. Compared to other collision-based algorithms, it has mainly the following interests:

- Techniques of variance reduction can be applied to improve the convergence performance.
- The spectral properties can be considered in SOLSTICE, for example, in modeling the Fresnel Concentrator Lens. This is because the power P[W] is expressed as the product of spatial integral, angular integral and spectral integral of monochromatic intensity $I_\nu[\mathrm{W}\cdot\mathrm{m}^{-3}\cdot\mathrm{sr}^{-1}]$.
- The annual received energy E[kWh] is easy to be calculated by IFMC method because it is just a temporal integral of the power P. A study concerning this research problem can be referred to [16].
- Other physical modelings (heat transfer modeling of the receiver) can be coupled with the radiative transfer model. The coupling model can be solved numerically by the Monte Carlo method. For example, a numerical tool, STARDIS, is developed by Mesostar to simulate heat transfer problems using the Monte Carlo method. The temperature T is the observable formulated as an integral formulation in STARDIS. It

[2]https://www.meso-star.com/

is then possible to formulate the observable in the CSP system, usually the power P, as a function of intensity I and temperature T, in an integral formulation. By doing this, the heat transfer phenomena can be considered in the evaluation of the CSP optical performance.

Therefore, we choose to work in this thesis in the framework of IFMC to maximize its potential in CSP applications.

2.2.3 Sensitivity studies in CSP systems

Simulation tools that evaluate the optical performance for CSP systems are various. However, estimating gradient (sensitivities) remains challenging in the CSP community. Only a few studies calculating the gradient have been realized because the sensitivities are hard to obtain. Sensitivities are usually required to solve the inverse problem and for optimization purposes.

Solving inverse problems

Solving a physical problem with a complete model (with modeling equations, physical properties, and boundary conditions) is called a direct problem or forward problem.

Now, for example, if the physical properties are missing, the physical model can not be solved. However, some inversion procedures may evaluate the missing physical properties by fitting the model output with real ones that can be obtained through experiments. The problem of evaluating such missing physical properties is called an inverse or backward problem. Solving an inverse problem often requires sensitivities [17].

For example, researchers aim to determinate a reflective property (noted $\ddot{\pi}$) of a heliostat [5]. They need to measure the real flux map of this heliostat on the receiver and simulate the flux map with an initial $\ddot{\pi}$ for the same heliostat. If the impact of $\ddot{\pi}$ to the flux map (the sensitivity) can be known, they can then adjust the value of $\ddot{\pi}$ so that the simulated flux map corresponds to the measured one. The adjusted value of $\ddot{\pi}$ represents the reflective property corresponding to the real heliostat.

Geometric optimizations

The geometry of a CSP system can be very complex. For example, it can be composed of a massive number of heliostats. It is challenging to optimize the geometric design of a STP system to maximize the impacting power on the receiver, to homogenize the flux map on the receiver, or to maximize gathered energy over time.

Usually, researchers couple an MCRT method (see section 2.2.2) with gradient-free optimization algorithms such as the genetic algorithms, PSO, etc [13, 16, 73]. However, the gradient-free optimization algorithm is used as a "black box tool". Nevertheless, it is hard to know which parameters are the most sensitive.

Few researchers calculate the sensitivities of geometric parameters for optimization purposes. The geometric sensitivities are approximated by the finite difference method to optimize some simple solar concentrators [40]. However, as discussed in Chapter.1, the corresponding standard deviation is very hard to control. Therefore, in the literature, this method of approximation is only used for simple geometries of the solar concentrator. Also, another attempt to approximate the gradient has been suggested by [66] by the "area-ratio" method. However, this method is based on specific assumptions and induces an empiric weight.

In Chapter.5, we introduce and develop a method to estimate geometric sensitivities based on the derivation of RTE followed by a reformulation as an integral formulation. In an STP system, the sensitivities of impacting power with respect to the translation, rotation, and the

length of edge for each heliostat in the field can be obtained. We will not only distinguish the system's most sensitive parameters but also physically understand why they are sensitive. Heliostat position, orientation, and size can be adjusted to optimize the optical performance of a STP system by knowing those sensitivities.

Last but not least, it makes using gradient-based optimization algorithms possible. For an optimization problem, when the parameter space dimension is large, the gradient-based optimization algorithm is usually used [17]. The optimization problems in CSP usually have a significant parameter space dimension because many parameters have to be optimized. Since the optimization processes are limited to gradient-free methods, researchers use empiric parameterization techniques to decrease the numbers of parameters [16, 67]. With the information on sensitivities, gradient-based optimization algorithms can be used to avoid these empiric approaches. However, regardless of the Monte Carlo method to compute the gradient (sensitivities), it is always given with a standard deviation of this gradient, which needs to be better handled by optimization algorithms.

2.3 Estimate sensitivities by Monte Carlo method in radiative system

More generally, researchers in physics and image synthesis have been working around the question of estimating sensitivities in radiative systems for years. Specially in image synthesis, coupling with the algorithm of optimizations and neutral network [26], the sensitivities are then served to the 3D object reconstruction [27, 28, 56, 65], the body shape estimation [4, 45], hand shape estimation [1, 72], face reconstruction [20], etc.

Referring to Chapter.1, researchers focus on estimating the sensitivity of observable with respect to a parameter that characterizes the radiative system (for example, $\ddot{\pi}$). It could be the sensitivity of the intensity on a location $\vec{x}_{obs}$ towards a direction $\vec{\omega}_{obs}$:

$$\partial_{\ddot{\pi}} I(\vec{x}, \vec{\omega}, \ddot{\pi}), \tag{2.1}$$

the sensitivity of Irradiance:

$$\partial_{\ddot{\pi}} G, \tag{2.2}$$

the sensitivity of radiance flux:

$$\partial_{\ddot{\pi}} \phi, \tag{2.3}$$

or the sensitivity of impacting power:

$$\partial_{\ddot{\pi}} P. \tag{2.4}$$

We classify the available methods which can be used to estimate these sensitivities of observables into three categories. They are detailed in the next paragraph:

1. Method of observable derivation
2. Method of differentiable rendering
3. Method of sensitivity model

2.3.1 Approximation by Finite Difference method

Before introducing the three methods of estimating sensitivity, we first review the finite difference method, which is a naive approach to approximate sensitivities. It is already introduced in Chapter.1.

This method suffers from several drawbacks:

- The approximation is biased because of discretization.
- The standard deviation tends to infinity when the discretization parameter tends to zero.

Despite these drawbacks, this method is used for validation purposes in this thesis, in the community of radiative transfer [34, 51] and the community of image synthesis [71].

2.3.2 Method of observable derivation

This method is based on differentiating the integral of an expected value form. An example can be found in Chapter.3. In his Ph.D. work, Roger distinguished two cases [51]. Firstly, when the integration domain is independent of the parameter to be differentiated, the required formal development is relatively simple [10]. The related sensitivity is named a "parametric sensitivity". Otherwise, if the integration domain depends on the parameter to be differentiated, the formal development is more complicated since it introduces a domain-deformation velocity [51]. In this case, it is referred to as a "geometric sensitivity".

Advantages

This method allows us to estimate the observable and its gradient “simultaneously”. In other words, the estimations of the observable and its sensitivities are entirely vectorized.

Therefore, “simultaneously” means:

1. Mathematically, the observable (usually is the intensity I) and its sensitivity with respect to a parameter (for example $\bar{\pi}$ and the sensitivity $\partial_{\bar{\pi}} I$) are formulated as two expected values to be solved by Monte Carlo method. When these integrals are defined on the same domain, they can be vectorized (seeing Example.1.3.5 in Chapter.1) and be solved by Monte Carlo method “simultaneously”.

2. Physically, when we simulate a radiative system by the Monte Carlo method, we generate the ray paths within their path space. These ray paths contain information as the intensity or its sensitivities. When the path space of intensity and the path space of its sensitivities are the same, the same ray path simulated can contain information on the intensity and its gradient. The transfer of intensity and its sensitivities can then be simulated “simultaneously”.

3. In an algorithmic point of view, the estimation of Monte Carlo requires samplings according to a pdf. If the estimation of observable and its sensitivities have the same pdf, they can then be estimated “simultaneously”. Practically, a sampled random number is used twice, but the corresponding weights are different.

Disadvantages

A complete review of the difficulties found when applying this method in a CSP context has been realized by De La Torre [9]. He detailed three examples to estimate the sensitivity of the impacting power with respect to the tower height.

1. The required formal development is huge, even if only one heliostat is considered. Therefore, it is almost impractical to consider the blocking and shadowing effects when considering more than one heliostat.

2. An issue of convergence is found when considering a heliostat that is not perfectly specular.

This method seems limited to academic examples.

2.3.3 Method of differentiable rendering

Attention has also been paid to the research field of image synthesis around the problem of estimating the sensitivities by Monte Carlo. There are different techniques to render a 2D image from a 3D scene. Among them, the PBR is one of the most realistic ways [46], which is also based on the Monte Carlo method. Researchers in image synthesis are also motivated to calculate the sensitivities for the applications in 3D object reconstruction, body shape estimation, hand shape estimation, etc.

Unlike the method of observable derivation, researchers in image synthesis construct an integral formulation for the observable and another for its sensitivity. Furthermore, according to different sensitivities, their integral formulations can be different. This process is called differentiable rendering in the literature of image synthesis [26], and in this thesis, we call it the method of differentiable rendering. The method of differentiable rendering is adapted to complex geometries. More details about this method can be found in Chapter.3.

Advantages

This method is used widely in the community of image synthesis, where complex geometry is regarded as a major challenge. Studies are realized to render images of sensitivity for complex geometries [38, 68, 71]. Therefore, this method is well adapted to be applied in radiative systems involving complex geometry.

Disadvantages

This method focuses on rendering images of sensitivities. Therefore, the corresponding algorithm is based on a reverse ray-tracing algorithm [46]. However, the reverse ray-tracing algorithm is proved to have poor convergence performance in evaluating optical performance in CSP [9]. Integrating this method to CSP applications will be inconvenient.

Furthermore, sensitivities' physical interpretations seem less important in the image synthesis community. The studies focus more on the solution of the partial differential equations, the algorithm, and the pipelines [26]. However, the physical interpretations of the sensitivities are essential for engineers and physicians.

2.3.4 Method of sensitivity model

The sensitivity model method was first developed by Lapeyre [32]. It relies on building physical models for the sensitivities by differentiating the model (which consists of differentiating the RTE and the boundary conditions). The sensitivities in the system are regarded as physical quantities transported in the radiative system. They behave similarly to the intensity: absorbed, scattered, emitted, and reflected in the system. The similitude of the transport of intensity and of sensitivity is studied in [31, 32, 34].

In a radiative system, the intensity I is the descriptor, a function of position $\vec{x}$ and a direction of propagation $\vec{\omega}$. If $\ddot{\pi}$ is one of the parameters characterizing the system, the intensity is also a function of $\ddot{\pi}$. If we note the intensity $I \equiv I(\vec{x}, \vec{\omega}, \ddot{\pi})$, the sensitivity (noted s) is then its derivative with respect to the parameter $\ddot{\pi}$:

$$s(\vec{x}, \vec{\omega}, \ddot{\pi}) \equiv \partial_{\ddot{\pi}} I(\vec{x}, \vec{\omega}, \ddot{\pi}) \tag{2.5}$$

Physically, it can be regarded as the local perturbation of intensity on $\vec{x}$, following $\vec{\omega}$, with respect to the parameter $\ddot{\pi}$. This perturbation is also transported in the radiative system, as shown in Fig.2.4.

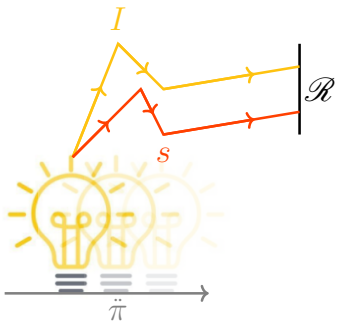


Figure 2.4 – Intensity I is emitted from the bulb and scattered in the semi-transparent medium. If $\ddot{\pi}$ characterizes the position of the bulb, a perturbation of intensity with respect to $\ddot{\pi}$, noted s, can be regarded as being emitted from the bulb and scattered in the medium too.

We have seen in the last chapter that we created a model of transport for intensity. We then formulate the integral formulation for an observable, on which a physical image of "photons" and a Monte Carlo algorithm are proposed. Following the method of sensitivity model, we create models for different sensitivities. We also formulate integral formulations for the sensitivities of observable. Physical images and Monte Carlo algorithms are also proposed for these integral formulations for sensitivities.

In order to build a model of sensitivity with respect to a parameter $\ddot{\pi}$, we need to differentiate the model of intensity (differentiate the RTE and the boundary conditions). We will see in the following chapters of this thesis that the boundary condition of the sensitivity is usually coupled with the spatial derivative and angular derivative of the intensity.

The spatial derivative is defined by

$$\partial_1 I(\vec{x}, \vec{\omega}, \ddot{\pi}) \tag{2.6}$$

and the angular derivative is defined by

$$\partial_2 I(\vec{x}, \vec{\omega}, \ddot{\pi}). \tag{2.7}$$

Similarly to the sensitivity $s(\vec{x}, \vec{\omega}, \ddot{\pi})$, the spatial derivative and the angular derivative are also regarded as physical quantities being transported in the system. The similitude of the transports for intensity, sensitivity, spatial derivative, and angular derivative have been considered for the first time in the Ph.D. work of Paule Lapeyre [32, 34].

Advantages

The main advantage of this method is the convenience of interpreting the sensitivities physically. We will see later in Chapter.5 that the sensitivities of the impacting power in a CSP system, with respect to the translation, rotation, and the size of each heliostat are estimated based on this method. Nevertheless, a detailed analysis of how the physical phenomena in STP system (blocking, shadowing, spillage, etc.) affect the sensitivities is done. Engineers and researchers can then understand which parameters are more sensitive than others and why they are more sensitive.

This advantage benefits from the existence of the sensitivity model. The sensitivity model uses the sensitivity of intensity as the descriptor. It describes how the sensitivity of intensity is transported in the system and distinguishes the sources of sensitivity. The sources of sensitivity are related to the perturbation of intensity caused by the perturbation of the corresponding parameter of the system. By analyzing the sensitivity models, the physical interpretations of sensitivities become possible. More details about analyzing the sensitivity in a CSP system can be referred to Chapter.5.

Disadvantages

When the number of sensitivities becomes large, the calculating time becomes consuming, and the coding development becomes a heavy task. Compared to the method of observable derivation, the method of sensitivity model is more adapted to complex configurations (for example, a STP system). However, a radiative system can depend on many parameters, and each parameter must have its corresponding sensitivity model.

For example, in Fig.2.5, $\ddot{\pi}_1$ and $\ddot{\pi}_2$ characterize the two directions of translation of the bulb. If we want to estimate an observable and its derivatives with respect to the two parameters, three models have to be built: the model of intensity I, the model of sensitivity for $\ddot{\pi}_1$: $s_1 \equiv \partial_{\ddot{\pi}_1} I$ and the model of sensitivity for $\ddot{\pi}_2$: $s_2 \equiv \partial_{\ddot{\pi}_2} I$. For CSP applications, the order of magnitude of sensitivities depends on the number of heliostats and the number of considered parameters for each heliostat (translation, rotation, size...). For example, Ivanpah solar power facility is the largest solar facility in the world in 2022 and is composed of 173500 heliostats. This very high number of parameters has motivated the vectorization technique to decrease the computing time, which is the research problem of this thesis. The research problem is discussed in detail in the following section.

Moreover, we will see in the following chapters that when the boundary conditions of intensity are not mathematically smooth (for example, the borders of heliostats make the boundary conditions discontinuous), surface and line sources can both appear in each sensitivity model. Therefore, for each sensitivity model, the path spaces of different sources are different because they have different locations: on the surfaces and the segments. As discussed in Chapter1, different path spaces mean that different Monte Carlo samplings are needed, making the coding development heavier and the computing time-consuming.

2.4 The research problem

The method of observable derivation estimates an observable and all its sensitivities "simultaneously" because they have the same path space (Fig.2.6). In other words, the estimations of the observable and all its sensitivities are naturally vectorized because the intensity and

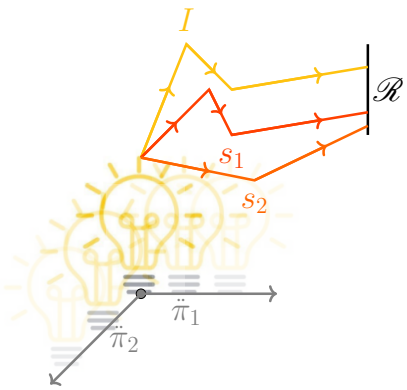


Figure 2.5 – For the two parameters characterizing the position of the bulb ($\ddot{\pi}_1$ and $\ddot{\pi}_2$), two sensitivity models are needed. In order to solve the problems of intensity and sensitivities, three models must be built.

the sensitivities are all transported as a vectorized form (Fig.2.6). However, it can not be applied to complex CSP systems, especially when shadowing or blocking between heliostats cannot be neglected. In addition, we will see in the next chapter that the formal development required following this method is cumbersome.

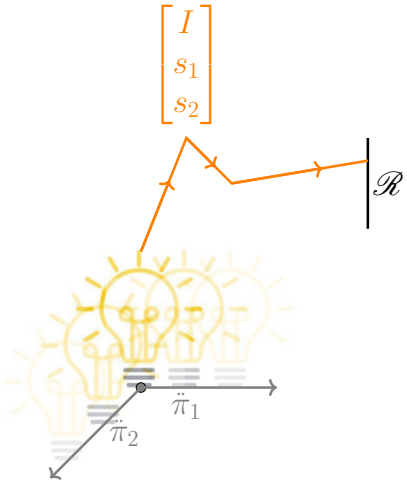


Figure 2.6 – The method of observable derivation has only one path space for the intensity and all sensitivities. The transports of the intensity and the sensitivities are vectorized.

On the other hand, the image synthesis community has developed efficient algorithms to estimate sensitivities that can be applied to complex geometries. However, it focuses on producing rendering images, but the reverse ray-tracing algorithms have poor convergence performances in CSP.

The method of sensitivity model can be applied to complex problems, but the number of path spaces increases, and the sensitivities are estimated separately. It becomes almost hardly practicable when the number of sensitivities to evaluate is high. Therefore, in this work, the research problem is to regain the advantages of "vectorization" for the method of sensitivity model.

Therefore, in this thesis, we tackle this research problem step by step. In Chapter.3, we will apply and compare the three methods mentioned in Section.2.3 into a simple 1-D

radiative transfer case. Secondly, we will apply the method of sensitivity model to radiative transfer cases involving complex geometries. We will distinguish different sources having the same path space in the sensitivity model. Sometimes it is obvious, but sometimes, mathematical reformulation efforts are needed. We will then only vectorize the sources sharing the same path space (in Chapter.4). In Chapter.5, we address the estimation of thousands of sensitivities for a CSP system composed of thousands of heliostats that can shade each other. The massive estimation of sensitivities becomes achievable only with this strategy of vectorization. Lastly, we reformulate the sensitivity models to vectorize all the sources. This last chapter will rely on the use of Stokes' theorem. By doing this, the advantages of the method of observable derivation (a perfect vectorization) can be regained, but some limitations are identified.

3

Three viewpoints on estimating sensitivities in radiative systems

3.1 Introduction

In this chapter, we will discuss and review the three viewpoints found in the literature to estimate sensitivities in a radiative system announced in Chapter.2

1. The method of observable derivation [50, 51].
2. The method of differentiable rendering [38, 68, 70, 71].
3. The method of sensitivity model [32, 33].

To better understand the three methods, we apply them to a one-dimension radiative problem, which is a typical, widely studied radiative problem for which an analytical solution is available [2]. Their performances, conveniences, and limitations will be compared and discussed in this application context.

3.2 Case study: radiation in a 1d-rod

A one-dimension rod (Fig.3.1) is considered, which extends from 0 to $\ddot{\pi}$ We denote a generic point in the rod by x. A source of radiation locates on the left side of rod where $x = 0$ and there is no source of radiation on the right side at $x = \ddot{\pi}$. These are two Dirichlet boundary conditions. The photons move to the right and left along this rod without interacting with each other, but collisions happen in the medium. The absorption and scattering in the rod are considered homogeneous and isotropic (k_a and k_s are constant), and the medium is considered "cold" (no emission).

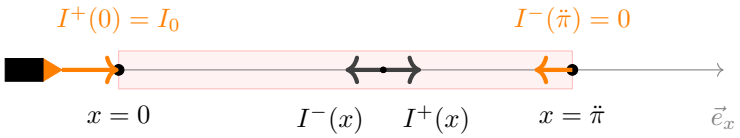


Figure 3.1 – Configuration of 1-D rod

In standard radiative transfer modeling, the intensity is a function of position $\vec{x}$ and direction of propagation $\vec{\omega}$. Since we focus now on a 1-D problem, the position is noted by a scalar x, $\forall x \in [0, \ddot{\pi}]$. The direction is noted by index $+$ and $-$. Therefore, $I^+(x)$ represents the intensity at position x, propagating toward the right side and $I^-(x)$ represents the intensity at position x, propagating toward the left side (Fig.3.1).

We address the two following questions:

1. What is the intensity at each point and of the two directions in the rod: $I^+(x)$ and $I^-(x)$? The intensity is the observable.
2. What is the sensitivity of the observable (the intensity) at each point and toward the two directions in the rod, with respect to the length of the rod $\ddot{\pi}$: $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$?

In order to answer the first question, standard radiative transfer modeling and a Monte-Carlo algorithm will be built to estimate $I^+(x)$ and $I^-(x)$. The process will be detailed in the following section. Furthermore, in order to estimate $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$, the three methods will be applied.

3.3 Estimation of intensity

The standard radiative transfer modeling process to estimate the intensity of all points $I^+(x)$ and $I^-(x)$ has three steps:

1. Build a model of intensity, including the equation of transport in the medium (RTE) and the boundary conditions.
2. Formulate the observable in an integral form (also called Fredholm equation of the second kind, seeing the following box).
3. Transform the Fredholm equation into an expected value form to evaluate the observable by Monte Carlo methods.

Fredholm equation of the second kind

A Fredholm equation of the second kind is given as:

$$\phi(t) = f(t) + \lambda \int_a^b K(t,s)\phi(s)ds \tag{3.1}$$

Given the kernel $K(t,s)$, a constant λ and the function $f(t)$, the problem is typically to find the function $\phi(t)$.

3.3.1 The model of intensity

Referring to Chapter.1, the RTE in a homogeneous cold medium is:

$$\vec{\omega} \cdot \partial_1 I(\vec{x},\vec{\omega}) = -k_a I(\vec{x},\vec{\omega}) - k_s I(\vec{x},\vec{\omega}) + k_s \int_{\Omega'} \mathscr{P}(-\vec{\omega}'|-\vec{\omega}) d\vec{\omega}' I(\vec{x},\vec{\omega}'). \tag{3.2}$$

Since this example is a one-dimensional problem, the direction of propagation $\vec{\omega}$ will be a one-dimensional vector, which has only two possible values: $\vec{\omega} \in \{-\vec{e}_x, \vec{e}_x\}$, seeing Fig.3.1. $\vec{\omega} = \vec{e}_x$ represents the right propagation of the intensity and vice-versa. The position vector $\vec{x}$ can be noted as a scalar x because it is also a vector of one dimension. Finally, we note: $I^+(x) = I(\vec{x},\vec{\omega})|_{\vec{\omega}=\vec{e}_x}$ representing the right-side propagation and $I^-(x) = I(\vec{x},\vec{\omega})|_{\vec{\omega}=-\vec{e}_x}$ the left-side propagation.

Since the scattering is isotropic, in this one-dimensional problem, the phase-function becomes: $\mathscr{P}(-\vec{\omega}'|-\vec{\omega}) = \frac{1}{2}$, because only two directions are available for the transport of intensity.

Therefore, we have:

$$\int_{\Omega'} \mathscr{P}(-\vec{\omega}'|-\vec{\omega}) d\vec{\omega}' I(\vec{x},\vec{\omega}') = \frac{I(\vec{x},\vec{\omega}')}{2}|_{\vec{\omega}'=\vec{e}_x} + \frac{I(\vec{x},\vec{\omega}')}{2}|_{\vec{\omega}'=-\vec{e}_x} = \frac{I^-(x)+I^+(x)}{2} \tag{3.3}$$

The RTE in this 1-D rod problem becomes:

$$\begin{cases} \dfrac{dI^+(x)}{dx} = -k_a I^+(x) - k_s I^+(x) + k_s(\dfrac{I^+(x)}{2} + \dfrac{I^-(x)}{2}) \\ -\dfrac{dI^-(x)}{dx} = -k_a I^-(x) - k_s I^-(x) + k_s(\dfrac{I^+(x)}{2} + \dfrac{I^-(x)}{2}). \end{cases} \tag{3.4}$$

Alternatively, let k_e be the extinction coefficient: $k_e = k_s + k_a$ and λ be the single-scattering albedo: $\lambda = \frac{k_s}{k_e}$. The RTE in this 1-D rod problem can also be written as:

$$\begin{cases} \dfrac{dI^+(x)}{dx} = -k_e I^+(x) + k_e\lambda(\dfrac{I^+(x)}{2} + \dfrac{I^-(x)}{2}) \\ -\dfrac{dI^-(x)}{dx} = -k_e I^-(x) + k_e\lambda(\dfrac{I^+(x)}{2} + \dfrac{I^-(x)}{2}). \end{cases} \tag{3.5}$$

Eq.3.4 and Eq.3.5 are then equivalent. A source of intensity locates on $x = 0$, which yields the Dirichlet boundary conditions:

$$\begin{cases} I^+(0) = I_0 \\ I^-(\ddot{\pi}) = 0 \end{cases} \tag{3.6}$$

Herein, Eq.3.5 (or Eq.3.4) and Eq.3.6 compose the complete model of intensity I. This model is already mathematically solved[2]. The analytical solution exists for the intensity at all points in the rod propagating to the left $I^-(x)$ and to the right $I^+(x)$:

$$I^+(x|\lambda, k_e, \ddot{\pi}) = \frac{2(-1+\lambda)\cosh\left((\ddot{\pi}-x)\sqrt{1-\lambda}k_e\right) + \sqrt{1-\lambda}(-2+\lambda)\sinh\left((\ddot{\pi}-x)\sqrt{1-\lambda}k_e\right)}{2(-1+\lambda)\cosh\left(\ddot{\pi}\sqrt{1-\lambda}k_e\right) + \sqrt{1-\lambda}(-2+\lambda)\sinh\left(\ddot{\pi}\sqrt{1-\lambda}k_e\right)} \tag{3.7}$$

and

$$I^-(x|\lambda, k_e, \ddot{\pi}) = \frac{-\lambda\sqrt{1-\lambda}\sinh\left((\ddot{\pi}-x)\sqrt{1-\lambda}k_e\right)}{2(-1+\lambda)\cosh\left(\ddot{\pi}\sqrt{1-\lambda}k_e\right) + \sqrt{1-\lambda}(-2+\lambda)\sinh\left(\ddot{\pi}\sqrt{1-\lambda}k_e\right)} \tag{3.8}$$

Now, we try to solve the model of intensity I numerically by the Monte Carlo method. The analytical solutions will then be used to validate our results obtained by the Monte Carlo method.

3.3.2 Fredholm Equation

The model of intensity is described by differential equations (Eq.3.5), but the Monte-Carlo method estimates the value of an integral (in an expected value form).

Based on the model of intensity, we build an integral equation for the intensity $I^+(x)$ and $I^-(x)$ (the observable) so that we transfer the problem to the estimation of an expected value, which can be estimated by Monte-Carlo method.

Mathematically, the Eq.3.5 is composed of two *First-order non-homogeneous linear differential equation*, of which the solution is an integral formulation (the Fredholm equation).

Solution of the first-order homogeneous linear differential equation

Let's consider a *First-order non-homogeneous linear differential equation* and a boundary condition at x_0:

$$\begin{cases} \dfrac{dy}{dx} + \alpha y = \beta(x) \\ y(x_0) = y_0 \end{cases} \tag{3.9}$$

where y and β are functions of x ($x \in [0, x_0]$) and α is a constant, supposing that the boundary condition is known $y(x_0) = y_0$.
Following the integrating factor method, we multiply the equation by $e^{\alpha x}$ for two sides:

$$\frac{dy}{dx}e^{\alpha x} + \alpha y e^{\alpha x} = \beta(x)e^{\alpha x} \tag{3.10}$$

which is:

$$(ye^{\alpha x})' = \beta(x)e^{\alpha x}, \tag{3.11}$$

implying that

$$\int_x^{x_0} \left(y(x')e^{\alpha x'}\right)' dx' = \int_x^{x_0} \beta(x')e^{\alpha x'} dx'. \tag{3.12}$$

Eq.3.12 yields to

$$y(x_0)e^{\alpha x_0} - ye^{\alpha x} = \int_x^{x_0} \beta(x')e^{\alpha x'} dx'. \tag{3.13}$$

Finally, we have:

$$y(x) = y(x_0)e^{\alpha(x_0-x)} - \int_x^{x_0} \beta(x')e^{\alpha(x'-x)} dx'. \tag{3.14}$$

Specially, when $x_0 = 0$:

$$y(x) = y(0)e^{-\alpha x} + \int_0^{x} \beta(x')e^{-\alpha(x-x')} dx' \tag{3.15}$$

Eq.3.15 and Eq.3.14 are both the solutions of Eq.3.10. The two solutions correspond to the two boundary conditions.

Therefore, based on Eq.3.5, the observable $I^+(x)$ and $I^-(x)$ can be formulated in an integral form:

$$\begin{cases} I^+(x) = I^+(0)e^{-k_e x} + \int_0^x e^{-k_e(x-x')} \dfrac{k_e\lambda}{2}(I^+(x') + I^-(x'))dx' \\ I^-(x) = I^-(\ddot{\pi})e^{-k_e(\ddot{\pi}-x)} + \int_x^{\ddot{\pi}} e^{-k_e(x'-x)} \dfrac{k_e\lambda}{2}(I^+(x') + I^-(x'))dx' \end{cases} \tag{3.16}$$

Eq.3.16 is called the Fredholm Equation of the second kind. The intensity appears on both sides of the equation. In this work, we will use the Monte-Carlo method to find the solution of this Fredholm Equation (find $I^+(x)$ and $I^-(x)$), which requires reformulating it into an expected value form.

3.3.3 Expected value form

Expected value reformulation requires introducing a pdf. On the first equation of Eq.3.16, we let $\xi^+ = x - x'$, so that we have $x' = x - \xi^+$ and $dx' = -d\xi^+$:

$$\begin{aligned} I^+(x) =& I^+(0)e^{-k_e x} + \int_0^x e^{-k_e\xi^+} \frac{k_e\lambda}{2}(I^+(x-\xi^+) + I^-(x-\xi^+))d\xi^+ \\ =& I^+(0)e^{-k_a x} \int_x^\infty k_s e^{-k_s\xi^+} d\xi^+ \\ &+ \int_0^x e^{-k_e\xi^+} \frac{k_e\lambda}{2}(I^+(x-\xi^+) + I^-(x-\xi^+))d\xi^+. \end{aligned} \tag{3.17}$$

With $k_e = k_a + k_s$ and $\lambda = \frac{k_s}{k_e}$, Eq.3.17 yields to:

$$I^+(x) = \int_0^\infty k_s e^{-k_s\xi} d\xi \begin{pmatrix} I^+(0)e^{-k_a x}\mathcal{H}(\xi > x) \\ +\frac{1}{2}I^+(x-\xi)e^{-k_a\xi}\mathcal{H}(\xi < x) \\ +\frac{1}{2}I^-(x-\xi)e^{-k_a\xi}\mathcal{H}(\xi < x) \end{pmatrix} \tag{3.18}$$

where $k_s e^{-k_s \xi}$ is the normalised pdf ($\int_0^\infty k_s e^{-k_s \xi} d\xi = 1$) of the random variable Ξ (ξ is a realization of Ξ). $\mathcal{H}$ is the Heaviside function that is equal to 1 if the condition is true and 0 if it is not. Physically, ξ is the sampled path length of scattering. We will also generate a probability for the direction of scattering: $\frac{1}{2}$ for the right side propagation and $\frac{1}{2}$ for the left side. Eq.3.18 is then reformulated as:

$$I^+(x) = \int_0^\infty k_s e^{-k_s \xi} d\xi \int_0^1 dr \left(\underbrace{\begin{array}{r} I^+(0) e^{-k_a x} \mathcal{H}(\xi > x) \\ + I^+(x-\xi) e^{-k_a \xi} \mathcal{H}(\xi < x) \mathcal{H}(r < \frac{1}{2}) \\ + I^-(x-\xi) e^{-k_a \xi} \mathcal{H}(\xi < x) \mathcal{H}(r > \frac{1}{2}) \end{array}}_{\hat{w}^+} \right) \tag{3.19}$$

The sum in the parentheses is called the weight of Monte-Carlo, noted $\hat{w}^+$.

Let $\xi^- = x' - x$, so that we have $x' = \xi^- + x$ and $dx' = d\xi^-$:

$$\begin{aligned} I^-(x) =& I^-(\ddot{\pi}) e^{-k_e(\ddot{\pi} - x)} + \int_0^{\ddot{\pi} - x} e^{-k_e \xi^-} \frac{k_e \lambda}{2} (I^+(x+\xi^-) + I^-(x+\xi^-)) d\xi^- \\ =& I^-(\ddot{\pi}) e^{-k_a x} \int_{\ddot{\pi} - x}^\infty k_s e^{-k_s \xi^-} d\xi^- \\ &+ \int_0^{\ddot{\pi} - x} e^{-k_e(\xi^-)} \frac{k_e \lambda}{2} (I^+(x+\xi^-) + I^-(x+\xi^-)) d\xi^- \end{aligned} \tag{3.20}$$

which yields:

$$I^-(x) = \int_0^\infty k_s e^{-k_s \xi} d\xi \begin{pmatrix} I^-(\ddot{\pi}) e^{-k_a(\ddot{\pi} - x)} \mathcal{H}(\xi > \ddot{\pi} - x) \\ + \frac{1}{2} I^+(x+\xi) e^{-k_a \xi} \mathcal{H}(\xi < \ddot{\pi} - x) \\ + \frac{1}{2} I^-(x+\xi) e^{-k_a \xi} \mathcal{H}(\xi < \ddot{\pi} - x) \end{pmatrix}. \tag{3.21}$$

In the same way as previously, Eq.3.21 is then reformulated as:

$$I^-(x) = \int_0^\infty k_s e^{-k_s \xi} d\xi \int_0^1 dr \left(\underbrace{\begin{array}{r} I^-(\ddot{\pi}) e^{-k_a(\ddot{\pi} - x)} \mathcal{H}(\xi > \ddot{\pi} - x) \\ + I^+(x+\xi) e^{-k_a \xi} \mathcal{H}(\xi < \ddot{\pi} - x) \mathcal{H}(r < \frac{1}{2}) \\ + I^-(x+\xi) e^{-k_a \xi} \mathcal{H}(\xi < \ddot{\pi} - x) \mathcal{H}(r > \frac{1}{2}) \end{array}}_{\hat{w}^-} \right). \tag{3.22}$$

The sum in the parentheses is called the weight of Monte-Carlo, noted $\hat{w}^-$.

At this stage, the integral formulation of $I^+(x)$ and $I^-(x)$ are transformed into two expected value forms (Eq.3.19 and Eq.3.22). It is noticed that the Monte-Carlo weight of Eq.3.19 and Eq.3.22 consist of I^+ and I^-. Therefore, a recursive process is used to estimate $I^+(x)$ and $I^-(x)$.

A continuous random variable Ξ corresponds to the sampling of path length.

It is defined by its domain of definition and its pdf $\mathscr{P}_\Xi$:

$$\begin{cases} \Xi \in [0, +\infty) \\ \mathscr{P}_\Xi(\xi) = k_s e^{-k_s \xi} \end{cases} \tag{3.23}$$

Another continuous random number R corresponds to the sampling of direction. It is defined by its domain of definition and its pdf $\mathscr{P}_R$:

$$\begin{cases} R \in [0, 1) \\ \mathscr{P}_R(r) = 1 \end{cases} \tag{3.24}$$

Recursively, series of the path length ξ and direction r of scattering will be sampled, until the boundary is reached (until the Heaviside function $\mathscr{H}(\xi > x)$ in 3.19 or the Heaviside function $\mathscr{H}(\xi > \ddot{\pi} - x)$ in 3.22 is true). Finally, these recursive samplings define a random variable Γ. Its domain of definition and its pdf are then the following:

$$\begin{cases} \Gamma \in [0, +\infty) \times [0, 1] \times [0, +\infty) \times [0, 1] \dots \\ \mathscr{P}_\Gamma(\gamma) = k_s e^{-k_s \xi} \times 1 \times k_s e^{-k_s \xi} \dots \end{cases} \tag{3.25}$$

Therefore, $I^+(x)$ and $I^-(x)$ can be expressed as expected values of Γ:

$$I^+(x) = \mathbb{E}[\hat{w}^+(\Gamma)] \tag{3.26}$$

and

$$I^-(x) = \mathbb{E}[\hat{w}^-(\Gamma)]. \tag{3.27}$$

Two Monte-Carlo estimators are then built to estimate the value of $I^+(x)$ and $I^-(x)$:

$$I^+(x) \approx \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} (\hat{w}_j^+(\Gamma)) \tag{3.28}$$

and

$$I^-(x) \approx \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} (\hat{w}_j^-(\Gamma)) \tag{3.29}$$

Eq.3.28 and Eq.3.29 are called the estimators of $I^+(x)$ and of $I^-(x)$, where the index j corresponds to its j^{th} realization. A corresponding Monte-Carlo algorithm can then be developed to estimate these expected values (Algo.1).

However, we will continue to write a general equation of Eq.3.18 and Eq.3.21 to interpret a clear physical image of this algorithm.

As discussed in the previous chapters, almost always, when we write the integral formulation as the solution to the transport problems in radiation, we can see the physical image of transport and the corresponding algorithm directly from the integral formulation.

3.3.4 Integral formulation, algorithm, and a physical image

The notations of $I^+(x)$ and $I^-(x)$ are used for the previous parts of this chapter to facilitate the reading. However, in this subsection, we will use another nomenclature, which leads to a clear physical image.

We will note the intensity as a function of two scalars: $I \equiv I(x,\omega)$, $\forall x \in [0, \ddot{\pi}]$, $\forall \omega \in \{-1, 1\}$. When $\omega = 1$, the propagation is toward the right side and vice-versa. Therefore, the intensity to be estimated $I^+(x)$ and $I^-(x)$ become:

$$I^+(x) \equiv I(x,\omega)|_{\omega=1}; I^-(x) \equiv I(x,\omega)|_{\omega=-1} \tag{3.30}$$

The boundary conditions (referring to Eq.3.6) are noted:

$$I^+(0) \equiv \mathring{I}(\omega)|_{\omega=1}; I^-(\ddot{\pi}) \equiv \mathring{I}(\omega)|_{\omega=-1} \tag{3.31}$$

The intensity I on the boundary is noted as $\mathring{I}$ to indicate that it is a source of intensity.

The Phase-function of ω is discontinuous, but we extend it here to a smooth function of pdf, satisfying:

$$\int_{-1}^{1} P_\Omega(\omega) d\omega = 1 \tag{3.32}$$

Finally, we have a general integral formulation for $I^+(x)$ and $I^-(x)$, with a index j, $\forall j \in \mathbb{N}_0$ for ω, ξ and d:

$$\begin{aligned} I(x,\omega) = \int_0^\infty k_s e^{-k_s \xi_0} d\xi_0 \Bigg\{ \mathcal{H}(\xi_0 - d_0) \mathring{I}(\omega_0) e^{-k_a d_0} + \\ \mathcal{H}(d_0 - \xi_0) \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_0^\infty k_s e^{-k_s \xi_1} d\xi_1 \Bigg\{ \mathcal{H}(\xi_1 - d_1) \mathring{I}(\omega_1) e^{-k_a(\xi_0 + d_1)} + \\ \mathcal{H}(d_1 - \xi_1) \int_{-1}^{1} P_\Omega(\omega_2) d\omega_2 \dots \Bigg\} \Bigg\} \end{aligned} \tag{3.33}$$

with

$$\begin{cases} d_0 = \omega_0 x - \dfrac{\ddot{\pi}(\omega_0 - 1)}{2} \\ d_j = (x - \displaystyle\sum_{k=0}^{j-1} \omega_k \xi_k) \omega_j - \dfrac{\ddot{\pi}(\omega_j - 1)}{2}, j > 0 \end{cases} \tag{3.34}$$

where ξ_0 and ξ_1 represent respectively the path length of the photon for its 1^{st} and 2^{nd} scattering; $\omega_0 = \omega$ which is the initial direction of propagation; ω_1 and ω_2 represent respectively the direction of propagation of the photon after its 1^{st} and 2^{nd} scattering; d_0 represents the distance that the path of photon needs to go through to escape the rod without scattering; d_1 represents the distance that, starting from the position where located the 1^{st} scattering, the photon need to go through to escape the rod without 2^{nd} scattering.

The Monte-Carlo algorithm can be read from Eq.3.33. A clear physical image (see Fig.3.2) is linked to this formulation (Eq.3.33).

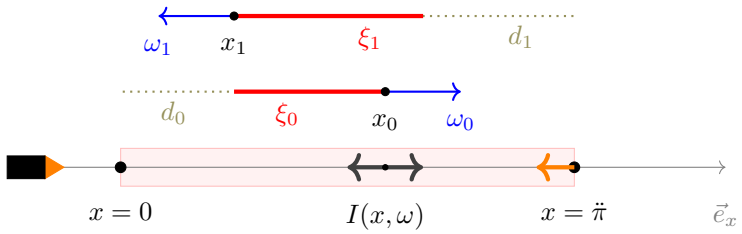


Figure 3.2 – Proposed physical image. Two scatterings are shown before the photon arrives at the point x, following the direction ω.

In order to determine the intensity $I(x, \omega)$, we will inversely trace the optical path from the position x in the direction ω, searching its origin ($x = 0$ or $x = \ddot{\pi}$ at the beginning of the path) and its optical path length that it has gone through. We will sample a path length for each realization according to $\mathcal{P}_\Xi$ on the direction $-\omega$ to determine the position where scattering happens.

- If the scattering point is out of the rod, we consider that the path arrived at position x without scattering. In this case (the Heaviside function $\mathcal{H}(\xi_0 - d_0)$ will be true), the optical path length of this photon is counted, and this photon carries the intensity from the boundary condition, attenuated by the absorption along the optical path.
- If the point of scattering is inside the rod (the Heaviside function $\mathcal{H}(d_0 - \xi_0)$ will be true), we sample a direction of scattering and then sample a path length again to determine the next scattering point.

We repeat this process again and again until the scattering point is out of the rod. Then we calculate how much intensity the photon carries to the point x in the direction ω_0 after all attenuation on its optical path length. Finally, we simulate the behaviors of millions of optical paths in this way. The estimated intensity of $I(x, \omega)$ will be the mean intensity carried by each photon.

In a particular case, when the scattering coefficient is null ($k_s = 0$), no scattering happens in the medium. The photons arrived at the position x, propagating toward the right side ($I(x, \omega)|_{\omega=1}$) will all originate from the left boundary. The arriving intensity will then be attenuated along the optical path in the medium. Therefore, when $k_s = 0$:

$$I(x, \omega) = \mathring{I}(\omega_0) e^{-k_a d_0} \tag{3.35}$$

and according to Eq.3.34, when $\omega = 1$:

$$d_0 = x \tag{3.36}$$

and when $\omega = -1$:

$$d_0 = \ddot{\pi} - x. \tag{3.37}$$

The corresponding algorithm to estimate $I(x, \omega)$ will be Algo.1.

Algorithm 1 Estimation of $I(x, \omega)$

Input: x, ω, k_a, k_s, n_{MC}, $\ddot{\pi}$

1: Initialize the sum, and the sum of square $W \leftarrow 0$, and $V \leftarrow 0$

for $i = 0 \rightarrow n_{MC}$ **do**
  Initialize the weight of Monte-Carlo $w \leftarrow 0$
  Initialize the total length of optical path $l_{tot} \leftarrow 0$
  Initialize the keeprunning flag $keeprunning \leftarrow 1$
  while keeprunning **do**
    if $\omega = 1$ **then**
      Sample a path length ξ following pdf: $p_{\Xi}(\xi) = k_s e^{-k_s \xi}$
      if $\xi >= x$ **then**
        $l_{tot} \leftarrow l_{tot} + x$
        $w \leftarrow I^+(0) e^{-k_a l_{tot}}$
        $keeprunning \leftarrow 0$
      else
        $l_{tot} \leftarrow l_{tot} + \xi$
        $x \leftarrow x - \xi$
        Sample a random number r uniformly for $r \in [0, 1]$
        if $r < \frac{1}{2}$ **then**
          $\omega \leftarrow 1$
        else
          $\omega \leftarrow -1$
        end if
      end if
    end if
    if $\omega = -1$ **then**
      Sample a path length ξ following pdf: $p_{\Xi}(\xi) = k_s e^{-k_s \xi}$
      if $\xi >= (\ddot{\pi} - x)$ **then**
        $l_{tot} \leftarrow l_{tot} + (\ddot{\pi} - x)$
        $w \leftarrow I^+(0) e^{-k_a l_{tot}}$
        $keeprunning \leftarrow 0$
      else
        $l_{tot} \leftarrow l_{tot} + \xi$
        $x \leftarrow x + \xi$
        Sample a random number r uniformly for $r \in [0, 1]$
        if $r < \frac{1}{2}$ **then**
          $\omega \leftarrow 1$
        else
          $\omega \leftarrow -1$
        end if
      end if
    end if
  end while
  $W \leftarrow W + w$
  $V \leftarrow V + w^2$
end for

Output: $I(x, \omega) \leftarrow \frac{W}{n_{MC}}$, $\sigma[I(x, \omega)] \leftarrow \sqrt{\frac{\frac{V}{n_{MC}} - (\frac{W}{n_{MC}})^2}{n_{MC} - 1}}$

3.4 Estimate sensitivities

Herein we remind that the objective is to estimate the sensitivity of intensity with respect to the length of the rod: $\partial_{\ddot{\pi}} I^+$ and $\partial_{\ddot{\pi}} I^-$. The previous section (Sec.3.3) shows the three steps to estimate the intensity by Monte Carlo methods.

In the following part of this chapter, we will apply three different methods to estimate the sensitivity:

1. The method of observable derivation in which the expected value formulation is derived (Eq.3.33)

2. The method of differentiable rendering, which consists of differentiating the Fredholm Equation(Eq.3.16)

3. The method of sensitivity model in which the sensitivity is seen as a physical value that is transported over the domain. It requires to differentiate the radiative model of intensity (Eq.3.6 and Eq.3.5).

These three methods are detailed in the following subsections.

3.4.1 Method of observable derivation

The principle of this method is briefly introduced in Chapter.1. Theories and applications are developed in [9, 50, 51] and four steps are to be followed when applying this method [50]:

1. **Formulate the integral formulation in an expected value form for Monte-Carlo algorithm** which is already done at this stage (Eq.3.33).

2. **Transform the integral formulation to adapt the calculation of derivative**. This is because in Eq.3.33, the targeting parameter $\ddot{\pi}$ is in the expression of d_j and d_0 in the Heaviside function. However, The method of observable derivation can only be applied when the targeting parameter $\ddot{\pi}$ is in the integral domain. Therefore, we will make efforts to formulate the integral domain depending on $\ddot{\pi}$ and Heaviside functions independent of $\ddot{\pi}$.

3. **Calculate the deformation velocity of the integral domain and build the integral formulation for estimating the sensitivity**. We will apply the formulations in [50, 51] to calculate the deformation velocity.

4. **Re-adapt the integral formulation to a form of expected value, as the same form of estimating the intensity**. This allows us to vectorize the estimations of intensity and sensitivity, which is the main advantage of this method compared to the other two methods.

Reformulation of the integral formulation

Based on Eq.3.33, we apply the Heaviside function to the integral domain, so that with Eq.3.38, no more Heaviside function is appearing and the integral domain is now depending on the parameter $\ddot{\pi}$. We remind here $\forall j \in \mathbb{N}_0, d_j \equiv d_j(\ddot{\pi})$ (Eq.3.34).

$$
\begin{aligned}
I(x,\omega) = \int_{d_0}^{\infty} k_s e^{-k_s \xi_0} d\xi_0 \mathring{I}(\omega_0) e^{-k_a d_0} + \\
\int_0^{d_0} k_s e^{-k_s \xi_0} d\xi_0 \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_{d_1}^{\infty} k_s e^{-k_s \xi_1} d\xi_1 \mathring{I}(\omega_1) e^{-ka(\xi_0+d_1)} + \\
\int_0^{d_1} k_s e^{-k_s \xi_1} d\xi_1 \int_{-1}^{1} P_\Omega(\omega_2) d\omega_2 \dots
\end{aligned} \tag{3.38}
$$

We can now apply the method of observable derivation.

Calculation of the deformation velocity of the integral domain and construction of the integral formulation for estimating the sensitivity

However, the formulation Eq.3.38 is still too complex to be differentiated. We will continue the development to simplify it.

The formulation of intensity $I(x,\omega)$ in Eq.3.38 is the sum of integrals: $I(x,\omega) = I_0(x,\omega) + I_1(x,\omega) + \ldots + I_i(x,\omega) + \ldots$ where $\forall i \in \mathbb{N}_0$. A physical interpretation exists for this decomposition of $I(x,\omega)$ [50]. $I(x,\omega)$ is the intensity arriving at x, following the direction ω. $I(x,\omega)$ is composed of all intensity coming without scattering, with once scattering, twice scatterings, etc. The index i indicates the number of scattering events before the path arrived on x, following the direction ω.

$I_i(x,\omega)$, which represents the intensity arriving to x, following the direction ω after i scatterings, is therefore written as:

$$\begin{aligned} I_i(x,\omega) = &\int_0^{d_0} k_s e^{-k_s\xi_0} d\xi_0 \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_0^{d_1} k_s e^{-k_s\xi_1} d\xi_1 \int_{-1}^{1} P_\Omega(\omega_2) d\omega_2 \\ &\ldots \int_{-1}^{1} P_\Omega(\omega_i) d\omega_i \int_{d_i}^{\infty} k_s e^{-k_s\xi_i} d\xi_i \left\{ \mathring{I}(\omega_i) e^{-ka(\xi_0+\xi_1\ldots+\xi_{i-1}+d_i)} \right\} \\ = &\int_0^{d_0} d\xi_0 \int_{-1}^{1} d\omega_1 \int_0^{d_1} d\xi_1 \int_{-1}^{1} d\omega_2 \ldots \int_{d_i}^{\infty} d\xi_i g_i(x,\omega_0,\xi_0,\omega_1,\ldots,\xi_i) \end{aligned} \tag{3.39}$$

specially when $i = 0$,

$$\begin{aligned} I_0(x,\omega) &= \int_{d_0}^{\infty} k_s e^{-k_s\xi_0} d\xi_0 \left\{ \mathring{I}(\omega_0) e^{-k_a d_0} \right\} \\ &= \int_{d_0}^{\infty} d\xi_0 g_0(x,\omega_0,\xi_0) \end{aligned} \tag{3.40}$$

with

$$g_i(x,\omega_0,\xi_0,\omega_1,\ldots,\xi_i) = k_s e^{-kd\xi_0} P_\Omega(\omega_1) k_s e^{-kd\xi_1} \ldots \mathring{I}(\omega_i) e^{-ka(\xi_0+\xi_1\ldots+\xi_{i-1}+d_i)} \tag{3.41}$$

and

$$g_0(x,\omega_0,\xi_0) = k_s e^{-k_s\xi_0} \mathring{I}(\omega_0) e^{-k_a d_0} \tag{3.42}$$

$I_i(x,\omega)$ has therefore, an integral domain noted D_i:

$$D_i = [0,d_0] \times [-1,1] \times [0,d_1] \times \cdots \times [d_i, +\infty] \tag{3.43}$$

As for the sensitivity $\partial_{\vec{\pi}} I(x,\omega)$, it can then be decomposed in the same way:

$$\partial_{\vec{\pi}} I(x,\omega) = \partial_{\vec{\pi}} I_0(x,\omega) + \partial_{\vec{\pi}} I_1(x,\omega) + \partial_{\vec{\pi}} I_2(x,\omega) \ldots \tag{3.44}$$

with $\forall i \in \mathbb{N}_0$:

$$\partial_{\vec{\pi}} I_i(x,\omega) = \int_0^{d_0} d\xi_0 \int_{-1}^{1} d\omega_1 \ldots \int_{d_i}^{\infty} d\xi_i \left\{ \partial_{\vec{\pi}} g_i + \vec{\nabla} \cdot \left(g_i \vec{V}_{\vec{\pi} i} \right) \right\} \tag{3.45}$$

where $\vec{V}_{\ddot{\pi}i}$ is the domain deformation velocity, which is studied in [50, 51].

Formulation of domain deformation velocity [50, 51]

We consider the following integral with an expected value form:

$$< O >= \int_{\mathcal{D}_1} \mathcal{P}_{\phi_1}(x_1)dx_1 \int_{\mathcal{D}_2} \mathcal{P}_{\phi_2}(x_2)dx_2 \cdots \int_{\mathcal{D}_n} \mathcal{P}_{\phi_n}(x_n)dx_n O(x_1, x_2, \ldots, x_n) \tag{3.46}$$

where $\mathcal{P}_{\phi_1}, \mathcal{P}_{\phi_2}, \ldots$ are the pdfs of a random variables $X_1, X_2, \ldots$ and $\mathcal{D}_1, \mathcal{D}_2 \ldots$ are the definition domains of $X_1, X_2, \ldots$.
If $< O >$ is a function of a parameter $\ddot{\pi}$:

$$< O >\equiv < O > (\ddot{\pi}), \tag{3.47}$$

the derivative of $< O >$ with respect to $\ddot{\pi}$ is formulated as follows:

$$\partial_{\ddot{\pi}} < O >= \int_{\mathcal{D}_1(\ddot{\pi})} dx_1 \int_{\mathcal{D}_2(\ddot{\pi})} dx_2 \cdots \int_{\mathcal{D}_n(\ddot{\pi})} dx_n \boldsymbol{\mathcal{P}}_\phi(\boldsymbol{x}) \left(\partial_{\ddot{\pi}} O(\boldsymbol{x}) + O(\boldsymbol{x}) \frac{\partial_{\ddot{\pi}} \boldsymbol{\mathcal{P}}_\phi(\boldsymbol{x})}{\boldsymbol{\mathcal{P}}_{\boldsymbol{\phi}}(\boldsymbol{x})} + \frac{\vec{\nabla} \cdot (\boldsymbol{\mathcal{P}}_\phi(\boldsymbol{x}) O(\boldsymbol{x}) \vec{V}_{\ddot{\pi}}(\boldsymbol{x}))}{\boldsymbol{\mathcal{P}}_\phi(\boldsymbol{x})} \right) \tag{3.48}$$

with

$$\boldsymbol{\mathcal{P}}_\phi(\boldsymbol{x}) = \prod_{k=1}^{n} \mathcal{P}_{\phi_k}(x_k), \tag{3.49}$$

$$O(\boldsymbol{x}) = O(x_1, x_2, \ldots, x_n) \tag{3.50}$$

Also, $\vec{V}_{\ddot{\pi}}(\boldsymbol{x})$ is called the deformation velocity. It is built component by component, starting with $V_{\ddot{\pi},1}$ which is a function of y_1 only. For $y_i \in \mathcal{D}_i$, where $\mathcal{D}_i \equiv [a_i, b_i]$:

$$V_{\ddot{\pi},1}(y_1; \ddot{\pi}) = \partial_{\ddot{\pi}} a_1 + \frac{\partial_{\ddot{\pi}} b_1 - \partial_{\ddot{\pi}} a_1}{b_1 - a_1}(y_1 - a_1) \tag{3.51}$$

$$V_{\ddot{\pi},i}(y_1, y_2, \ldots, y_i; \ddot{\pi}) = V_{\ddot{\pi},i}|_{y_i=a_i} + \frac{V_{\ddot{\pi},i}|_{y_i=b_i} - V_{\ddot{\pi},i}|_{y_i=a_i}}{b_i - a_i}(y_i - a_i) \tag{3.52}$$

with

$$\begin{aligned} V_{\ddot{\pi},i}|_{y_i=a_i} &\equiv V_{\ddot{\pi},i}(y_1, y_2, \ldots, y_{i-1}, y_i = a_i; \ddot{\pi}) \\ &= \partial_{\ddot{\pi}} a_i + \sum_{k=1}^{i-1} V_{\ddot{\pi},k}(y_1, y_2, \ldots, y_k; \ddot{\pi}) \partial_{y_k} a_i. \end{aligned} \tag{3.53}$$

Its dimension is the same as the dimension of the integral domain D_i:

$$\vec{V}_{\ddot{\pi}i} = \begin{bmatrix} V_{\ddot{\pi}i,\xi_0} \\ V_{\ddot{\pi}i,\omega_1} \\ V_{\ddot{\pi}i,\xi_1} \\ V_{\ddot{\pi}i,\omega_2} \\ \dots \\ V_{\ddot{\pi}i,\xi_i} \end{bmatrix} \tag{3.54}$$

where

$$\begin{cases} V_{\ddot{\pi}i,\xi_0} = -\dfrac{(\omega_0 - 1)\xi_0}{2d_0} \\ V_{\ddot{\pi}i,\xi_j} = \left[-\dfrac{\omega_j - 1}{2} + \displaystyle\sum_{k=0}^{j-1} V_{\ddot{\pi},k}(\xi_0, \xi_1, \dots, \xi_k; \ddot{\pi})(-\omega_k\omega_j) \right] \dfrac{\xi_j}{d_j}, \quad 0 < j < i \\ V_{\ddot{\pi}i,\xi_i} = -\dfrac{\omega_i - 1}{2} + \displaystyle\sum_{k=0}^{i-1} V_{\ddot{\pi},k}(\xi_0, \xi_1, \dots, \xi_k; \ddot{\pi})(-\omega_k\omega_i) \\ V_{\ddot{\pi}i,\omega_0} = V_{\ddot{\pi}i,\omega_j} = V_{\ddot{\pi}i,\omega_i} = 0 \end{cases} \tag{3.55}$$

$\partial_{\ddot{\pi}} I(x,\omega)$ will then become:

$$\begin{aligned} \partial_{\ddot{\pi}} I(x,\omega) = \int_{d_0}^{\infty} k_s e^{-k_s\xi_0} d\xi_0 \left[\frac{\partial_{\ddot{\pi}} g_0 + \vec{\nabla} \cdot \left(g_0 \vec{V}_{\ddot{\pi}0} \right)}{k_s e^{-k_s\xi_0}} \right] + \\ \int_0^{d_0} k_s e^{-k_s\xi_0} d\xi_0 \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_{d_1}^{\infty} k_s e^{-k_s\xi_1} d\xi_1 \left[\frac{\partial_{\ddot{\pi}} g_1 + \vec{\nabla} \cdot \left(g_1 \vec{V}_{\ddot{\pi}1} \right)}{{k_s}^2 e^{-k_s(\xi_0+\xi_1)}} \right] + \\ \int_0^{d_0} k_s e^{-k_s\xi_0} d\xi_0 \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_0^{d_1} k_s e^{-k_s\xi_1} d\xi_1 \int_{-1}^{1} P_\Omega(\omega_2) d\omega_2 \dots \end{aligned} \tag{3.56}$$

Construction of the same pdf for the intensity and for the sensitivity

We now transform the Eq.3.56 to an expected value form:

$$\begin{aligned} \partial_{\ddot{\pi}} I(x,\omega) = \int_0^{\infty} k_s e^{-k_s\xi_0} d\xi_0 \Bigg\{ \mathcal{H}(\xi_0 - d_0) \left[\frac{\partial_{\ddot{\pi}} g_0 + \vec{\nabla} \cdot \left(g_0 \vec{V}_{\ddot{\pi}0} \right)}{k_s e^{-k_s\xi_0}} \right] + \\ \mathcal{H}(d_0 - \xi_0) \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_0^{\infty} k_s e^{-k_s\xi_1} d\xi_1 \Bigg\{ \mathcal{H}(\xi_1 - d_1) \left[\frac{\partial_{\ddot{\pi}} g_1 + \vec{\nabla} \cdot \left(g_1 \vec{V}_{\ddot{\pi}1} \right)}{{k_s}^2 e^{-k_s(\xi_0+\xi_1)} P_\Omega(\omega_1)} \right] + \\ \mathcal{H}(d_1 - \xi_1) \int_{-1}^{1} P_\Omega(\omega_2) d\omega_2 \dots \Bigg\} \Bigg\} \end{aligned} \tag{3.57}$$

At this stage, one can easily find out that the integral formulation for $I(x,\omega)$ and $\partial_{\ddot{\pi}} I(x,\omega)$ share the same integral domain and the same pdf. Therefore, an optical path propagated in the rod can carry the intensity and the sensitivity at the same time. In other words, the

intensity and the sensitivity have the same path space. The estimations of the two values can be then vectorized:

$$\begin{bmatrix} I(x,\omega) \\ \partial_{\ddot{\pi}} I(x,\omega) \end{bmatrix} = \int_0^\infty k_s e^{-k_s \xi_0} d\xi_0 \Bigg\{ \mathcal{H}(\xi_0 - d_0) \begin{bmatrix} \mathring{I}(\omega_0) e^{-k_a d_0} \\ \frac{\partial_{\ddot{\pi}} g_0 + \vec{\nabla} \cdot \left(g_0 \vec{V}_{\ddot{\pi}0} \right)}{k_s e^{-k_s \xi_0}} \end{bmatrix} + \mathcal{H}(d_0 - \xi_0) \int_{-1}^{1} P_\Omega(\omega_1) d\omega_1 \int_0^\infty k_s e^{-k_s \xi_1} d\xi_1 \Bigg\{ \mathcal{H}(\xi_1 - d_1) \begin{bmatrix} \mathring{I}(\omega_1) e^{-k_a (\xi_0 + d_1)} \\ \frac{\partial_{\ddot{\pi}} g_1 + \vec{\nabla} \cdot \left(g_1 \vec{V}_{\ddot{\pi}1} \right)}{(k_s)^2 e^{-k_s (\xi_0 + \xi_1)} P_\Omega(\omega_1)} \end{bmatrix} + \mathcal{H}(d_1 - \xi_1) \int_{-1}^{1} P_\Omega(\omega_2) d\omega_2 \dots \Bigg\} \Bigg\} \quad (3.58)$$

This is a massive advantage because the intensity and the sensitivity share the same path space. As it is interpreted in Section.3.3.4, we simulate the scattering history of millions of optical paths. When we reach the boundary, we calculate how much intensity the path carries to the point of observation x and the direction of observation ω_0. In our case of estimating sensitivity, we can also calculate how much sensitivity the path carries to the point x in the direction ω.

Instead of simulating one path for intensity and another for sensitivity, each path carries two kinds of information simultaneously, which is the main interest of vectorizing the estimation of the intensity and the sensitivity.

However, the information on the sensitivity of the boundary is very complicated to obtain (even in this one-dimension example), which makes this method difficult to be applied to more realistic cases.

The corresponding algorithm simultaneously estimates the intensity and the sensitivity is Algo.2.

Algorithm 2 Evaluation of $I(x,\omega)$ and $\partial_{\ddot{\pi}} I(x,\omega)$

Input: x, ω, k_a, k_s, n_{MC}, $\ddot{\pi}$

Initialize the sum of w_L and the sum of ${w_L}^2$ $W_L \leftarrow 0$ and $V_L \leftarrow 0$
Initialize the sum of w_S and the sum of ${w_S}^2$ $W_S \leftarrow 0$ and $V_S \leftarrow 0$
for $i = 0 \rightarrow n_{MC}$ **do**
    Initialize the weight of Monte-Carlo for L $w_L \leftarrow 0$
    Initialize the weight of Monte-Carlo for $\partial_{\ddot{\pi}} L$ $w_S \leftarrow 0$
    Initialize the scattering counter $i \leftarrow 0$
    Initialize the total length of optical path $l_{tot} \leftarrow 0$
    Initialize the keeprunning flag $keeprunning \leftarrow 1$
    Initialize the tables ξ, ω, x
    Initialize the position $x[0] \leftarrow x$
    Initialize the position $\omega[0] \leftarrow \omega$
    while keeprunning **do**
        if $\omega[i] = 1$ **then**
            Sample a path length ξ' following pdf: $p_\Xi(\xi') = k_s e^{-k_s \xi'}$
            $\xi[i] \leftarrow \xi'$
            if $\xi[i] >= x[i]$ **then**
                $l_{tot} \leftarrow l_{tot} + x[i]$
                $w_L \leftarrow \mathring{I}(\omega[i]) e^{-k_a l_{tot}}$
                $w_S \leftarrow \frac{\partial_{\ddot{\pi}} g_i + \vec{\nabla} \cdot \left(g_i \vec{V}_{\ddot{\pi}i} \right)}{(k_s)^{i+1} e^{-k_s (\xi_0 + \xi_1 + \dots + \xi_i)} \prod_{k=1}^{i} P_\Omega(\omega_k)}$
                $keeprunning \leftarrow 0$

else
$l_{tot} \leftarrow l_{tot} + \xi[i]$
$x[i+1] \leftarrow x[i] - \xi[i]$
$i \leftarrow i + 1$
Sample a random number r uniformly for $r \in [0, 1]$
if $r < \frac{1}{2}$ **then**
$\omega[i] \leftarrow 1$
else
$\omega[i] \leftarrow -1$
end if
end if
end if
if $\omega[i] = -1$ **then**
Sample a path length ξ' following pdf: $p_\Xi(\xi') = k_s e^{-k_s \xi'}$
$\xi[i] \leftarrow \xi'$
if $\xi[i] >= (\ddot{\pi} - x[i])$ **then**
$l_{tot} \leftarrow l_{tot} + (\ddot{\pi} - x[i])$
$w_L \leftarrow \mathring{I}(\omega[i]) e^{-k_a l_{tot}}$
$w_S \leftarrow \frac{\partial_{\ddot{\pi}} g_i + \vec{\nabla} \cdot (g_i \vec{V}_{\ddot{\pi} i})}{(k_s)^{i+1} e^{-k_s(\xi_0 + \xi_1 + \ldots + \xi_i)} \prod_{k=1}^{i} P_\Omega(\omega_k)}$
$keeprunning \leftarrow 0$
else
$l_{tot} \leftarrow l_{tot} + \xi[i]$
$x[i+1] \leftarrow x[i] + \xi[i]$
$i \leftarrow i + 1$
Sample a random number r uniformly for $r \in [0, 1]$
if $r < \frac{1}{2}$ **then**
$\omega[i] \leftarrow 1$
else
$\omega[i] \leftarrow -1$
end if
end if
end if
end while
$W_L \leftarrow W_L + w_L$
$V_L \leftarrow V_L + w_L^2$
$W_S \leftarrow W_S + w_S$
$V_S \leftarrow V_S + w_S^2$
end for
Output:
$I(x, \omega) \leftarrow \frac{W_L}{n_{MC}}$, $\sigma[I(x, \omega)] \leftarrow \sqrt{\frac{\frac{V_L}{n_{MC}} - (\frac{W_L}{n_{MC}})^2}{n_{MC} - 1}}$
$\partial_{\ddot{\pi}} I(x, \omega) \leftarrow \frac{W_S}{n_{MC}}$, $\sigma[\partial_{\ddot{\pi}} I(x, \omega)] \leftarrow \sqrt{\frac{\frac{V_S}{n_{MC}} - (\frac{W_S}{n_{MC}})^2}{n_{MC} - 1}}$

3.4.2 Method of differentiable rendering

Before introducing the method, we review the concept of an "estimator" [46] in the research field of image synthesis. In our example, the intensity on a point x of two directions $I^+(x)$ and $I^-(x)$ are written in Eq.3.19 and Eq.3.22. They can therefore be written as two expected values in Eq.3.26 and Eq.3.27. Finally, two Monte-Carlo estimators are built to estimate the value of $I^+(x)$ and $I^-(x)$ (Eq.3.28 and Eq.3.29)

In this subsection, we aim to build the two corresponding estimators for sensitivities $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$.

This method of differentiable rendering consists in differentiating the Fredholm Equation (Eq.3.16). We differentiate here Eq.3.16 with respect to $\ddot{\pi}$:

$$\begin{cases} \partial_{\ddot{\pi}} I^+(x) = & \displaystyle\int_0^x e^{-k_e(x-x')} \frac{k_e \lambda}{2} (\partial_{\ddot{\pi}} I^+(x') + \partial_{\ddot{\pi}} I^-(x')) dx' \\ \partial_{\ddot{\pi}} I^-(x) = & \displaystyle\int_x^{\ddot{\pi}} e^{-k_e(x'-x)} \frac{k_e \lambda}{2} (\partial_{\ddot{\pi}} I^+(x') + \partial_{\ddot{\pi}} I^-(x')) dx' \\ & \qquad + e^{-k_e(\ddot{\pi}-x)} \frac{k_e \lambda}{2} I^+(\ddot{\pi}) \end{cases} \tag{3.59}$$

Eq.3.59 is then the Fredholm Equation of sensitivity. After that, we must transform it into an expected value form, generating the pdfs. Following the similar process in Section.3.3.3, the expected value form of $\partial_{\ddot{\pi}} I^+(x)$ is following:

$$\partial_{\ddot{\pi}} I^+(x) = \int_0^\infty k_s e^{-k_s \xi} d\xi \int_0^1 dr \underbrace{\begin{pmatrix} 0 & \mathcal{H}(\xi > x) \\ +\partial_{\ddot{\pi}} I^+(x-\xi) e^{-k_a \xi} & \mathcal{H}(\xi < x)\mathcal{H}(r < \frac{1}{2}) \\ +\partial_{\ddot{\pi}} I^-(x-\xi) e^{-k_a \xi} & \mathcal{H}(\xi < x)\mathcal{H}(r > \frac{1}{2}) \end{pmatrix}}_{\hat{w}_s^+} \tag{3.60}$$

and the expected value form of $\partial_{\ddot{\pi}} I^-(x)$ is:

$$\partial_{\ddot{\pi}} I^-(x) = \int_0^\infty k_s e^{-k_s \xi} d\xi \int_0^1 dr \underbrace{\begin{pmatrix} \frac{k_s}{2} I^+(\ddot{\pi}) e^{-k_a(\ddot{\pi}-x)} & \mathcal{H}(\xi > \ddot{\pi} - x) \\ +\partial_{\ddot{\pi}} I^+(x+\xi) e^{-k_a \xi} & \mathcal{H}(\xi < \ddot{\pi} - x)\mathcal{H}(r < \frac{1}{2}) \\ +\partial_{\ddot{\pi}} I^-(x+\xi) e^{-k_a \xi} & \mathcal{H}(\xi < \ddot{\pi} - x)\mathcal{H}(r > \frac{1}{2}) \end{pmatrix}}_{\hat{w}_s^-} \tag{3.61}$$

At this stage, the integral formulation of $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$ are transformed into two expected value forms (Eq.3.60 and Eq.3.61). Similar to the process described in Section.3.3.3, a recursive process is used to estimate these two sensitivity values. Recursively, series of the path length ξ (the random variable Ξ is defined in Eq.3.23) and the direction r (R is defined in Eq.3.24) of scattering will be sampled, until the boundary is reached. A random number Γ_s is defined by these recursive samplings. Its domain of definition and its pdf is then:

$$\begin{cases} \Gamma_s \in [0, +\infty) \times [0, 1] \times [0, +\infty) \times [0, 1] \dots \\ \mathcal{P}_{\Gamma_s}(\gamma_s) = k_s e^{-k_s \xi} \times 1 \times k_s e^{-k_s \xi} \dots \end{cases} \tag{3.62}$$

The definition of random variable Γ_s is the same as Γ (defined in Eq.3.25) used when solving the intensity model. It is still necessary to use the index s in Γ_s to distinguish the samplings when we solve the two models: the model of intensity and the model of sensitivity.

It is noticed that the value of $I^+(\ddot{\pi})$ is required in the Monte-Carlo weight $\hat{w}_s^-$ of Eq.3.61. Using the expected value form of I^+ in Eq.3.26, the expected value form of $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$ can be written as:

$$\partial_{\ddot{\pi}} I^+(x) = \mathbb{E}_{\Gamma_s}[\hat{w}_s^+(\Gamma_s, \mathbb{E}_{\Gamma|\Gamma_s}[\hat{w}^+(\Gamma|\Gamma_s)])] \tag{3.63}$$

and

$$\partial_{\ddot{\pi}} I^-(x) = \mathbb{E}_{\Gamma_s}[\hat{w}_s^-(\Gamma_s, \mathbb{E}_{\Gamma|\Gamma_s}[\hat{w}^+(\Gamma|\Gamma_s)])] \tag{3.64}$$

where $\mathbb{E}_{\Gamma_s}$ is the expected value corresponding to the sampling of Γ_s and $\mathbb{E}_{\Gamma|\Gamma_s}$ is the expected value corresponding to the sampling of Γ knowing Γ_s. The Monte-Carlo weight $\hat{w}^+$ and $\hat{w}^-$ are given by Eq.3.19 and Eq.3.22.

As discussed in Chapter.1, a Monte-Carlo estimator can be built naively. For each Monte-Carlo realization, the path lengths and scattering directions are sampled. When the value of $I^+(\ddot{\pi})$ is needed to compose the weight of this realization, we start up another Monte-Carlo estimator which estimates the value of $I^+(\ddot{\pi})$. However, the corresponding calculating time becomes huge, following this process of estimation of $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$.

Fortunately, via the statistic technique of *Double Randomization*, introduced in Chapter.1, we can avoid this difficulty. Since the Monte-Carlo weight $\hat{w}^+$, $\hat{w}^-$, $\hat{w}_s^+$ and $\hat{w}_s^-$ are linear functions, the expected value forms of $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$ can be reformulated:

$$\partial_{\ddot{\pi}} I^+(x) = \mathbb{E}_{\Gamma_s,\Gamma|\Gamma_s}[\hat{w}_s^+(\Gamma_s, \hat{w}^+(\Gamma|\Gamma_s))] \tag{3.65}$$

and

$$\partial_{\ddot{\pi}} I^-(x) = \mathbb{E}_{\Gamma_s,\Gamma|\Gamma_s}[\hat{w}_s^-(\Gamma_s, \hat{w}^+(\Gamma|\Gamma_s))] \tag{3.66}$$

The corresponding estimators are:

$$\partial_{\ddot{\pi}} I^+(x) \approx \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} (\hat{w}_s^+(\Gamma_s, \hat{w}^+(\Gamma|\Gamma_s))) \tag{3.67}$$

and

$$\partial_{\ddot{\pi}} I^-(x) \approx \frac{1}{n_{MC}} \sum_{j=1}^{n_{MC}} (\hat{w}_s^-(\Gamma_s, \hat{w}^+(\Gamma|\Gamma_s))) \tag{3.68}$$

Using the estimators shown in Eq.3.67 and Eq.3.68, the path lengths ξ and the directions of scattering r are sampled. These recursive processes sample a realization of the random variable Γ_s for the sensitivity estimator. If the Monte-Carlo weight requires the knowledge of $I^+(\ddot{\pi})$, instead of estimating it by another Monte-Carlo estimator, a recursive process of sampling ξ and r is started up following the expected value form of $I^+(x)|_{x=\ddot{\pi}}$ (Eq.3.19). The Monte-Carlo weight of intensity is retrieved to compose the Monte-Carlo weight of sensitivity.

The corresponding algorithm which estimates the sensitivity following this method is Algo.3.

Algorithm 3 Estimation of $\partial_{\ddot{\pi}} I^+(x)$ or $\partial_{\ddot{\pi}} I^-(x)$

Input: x, ω, k_a, k_s, n_{MC}, $\ddot{\pi}$ ▷ $\omega \in \{-1, 1\}$

if $\omega = 1$ **then** ▷ Initialize the flag
    $flag \leftarrow S^+$
else
    $flag \leftarrow S^-$
end if
Initialize the sum of the weight of Monte-Carlo $W_S \leftarrow 0$
Initialize the sum of the square of the weight of Monte-Carlo $V_S \leftarrow 0$
for $i = 0 \rightarrow n_{MC}$ **do**
    Initialize the weight of Monte-Carlo $w \leftarrow 0$
    Initialize the total length of optical path $l_{tot} \leftarrow 0$
    Initialize the keeprunning flag $keeprunning \leftarrow 1$
    while keeprunning **do**
        if $flag = S^+$ **then**
            Sample a path length ξ following pdf: $p_\Xi(\xi) = k_s e^{-k_s \xi}$
            if $\xi >= x$ **then**
                $w \leftarrow 0$
                $keeprunning \leftarrow 0$
            else
                $l_{tot} \leftarrow l_{tot} + \xi$
                $x \leftarrow x - \xi$
                Sample a random number r uniformly for $r \in [0, 1]$
                if $r < \frac{1}{2}$ **then**
                    $flag \leftarrow S^+$
                else
                    $flag \leftarrow S^-$
                end if
            end if
        end if
        if $flag = S^-$ **then**
            Sample a path length ξ following pdf: $p_\Xi(\xi) = k_s e^{-k_s \xi}$
            if $\xi >= (\ddot{\pi} - x)$ **then**
                $l_{tot} \leftarrow l_{tot} + (\ddot{\pi} - x)$
                $flag \leftarrow I^+$
                $x \leftarrow \ddot{\pi}$
            else
                $l_{tot} \leftarrow l_{tot} + \xi$
                $x \leftarrow x + \xi$
                Sample a random number r uniformly for $r \in [0, 1]$
                if $r < \frac{1}{2}$ **then**
                    $flag \leftarrow S^+$
                else
                    $flag \leftarrow S^-$
                end if
            end if
        end if
        if $flag = I^+$ **then**
            Sample a path length ξ following pdf: $p_\Xi(\xi) = k_s e^{-k_s \xi}$
            if $\xi >= x$ **then**
                $w \leftarrow \frac{k_s}{2} I^+(0) e^{-k_a l_{tot}}$
                $keeprunning \leftarrow 0$
            else
                $l_{tot} \leftarrow l_{tot} + \xi$
                $x \leftarrow x - \xi$

Sample a random number r uniformly for $r \in [0,1]$
if $r < \frac{1}{2}$ **then**
$flag \leftarrow I^+$
else
$flag \leftarrow I^-$
end if
end if
end if
if $flag = I^-$ **then**
Sample a path length ξ following pdf: $p_\Xi(\xi) = k_s e^{-k_s \xi}$
if $\xi >= (\ddot{\pi} - x)$ **then**
$w \leftarrow 0$
$keeprunning \leftarrow 0$
else
$l_{tot} \leftarrow l_{tot} + \xi$
$x \leftarrow x + \xi$
Sample a random number r uniformly for $r \in [0,1]$
if $r < \frac{1}{2}$ **then**
$flag \leftarrow I^+$
else
$flag \leftarrow I^-$
end if
end if
end if
end while
$W_S \leftarrow W_S + w$
$V_S \leftarrow V_S + w^2$
end for

Output: $\partial_{\ddot{\pi}} I^+(x)$ or $\partial_{\ddot{\pi}} I^-(x) \leftarrow \frac{W_S}{n_{MC}}$, $\sigma[\partial_{\ddot{\pi}} I^+(x)]$ or $\sigma[\partial_{\ddot{\pi}} I^-(x)] \leftarrow \sqrt{\frac{\frac{V_S}{n_{MC}} - (\frac{W_S}{n_{MC}})^2}{n_{MC}-1}}$

3.4.3 Method of the model of sensitivity

This method differentiates the intensity model to finally build a model of sensitivity [32, 33]. Compared to the method of observable derivation, this method is more flexible to be applied to complex problems. While compared to the method of differentiable rendering, this method has a very clear physical image. In the model of intensity, the intensity is the descriptor. We will see in the following part of this chapter that by building its own model, the sensitivity can be regarded as a descriptor of the model of intensity. The sensitivity then carries the local perturbation of intensity in the system as information. The intensity is absorbed and scattered following the model of intensity, while similarly, the sensitivity is absorbed and scattered following the model of sensitivity. We can then benefit from the well-developed tools of modeling and simulating the transport of intensity to model and simulate the sensitivities in the system.

In this section, based on the model of intensity I^+ and I^-, we will build a model of the sensitivity $\partial_{\ddot{\pi}} I^+$ and $\partial_{\ddot{\pi}} I^-$. Also, we will study and discuss the physical behaviors of $\partial_{\ddot{\pi}} I^+$ and $\partial_{\ddot{\pi}} I^-$ in the system.

RTE for the sensitivity

The RTE describes the transport of intensity in the medium, shown in Eq.3.5. We differentiate Eq.3.5 with respect to $\ddot{\pi}$ and we obtain the RTE for $\partial_{\ddot{\pi}} I^+$ and $\partial_{\ddot{\pi}} I^-$:

$$\begin{cases} \dfrac{d\partial_{\ddot{\pi}} I^+(x)}{dx} = -k_e \partial_{\ddot{\pi}} I^+(x) + k_e \lambda (\dfrac{\partial_{\ddot{\pi}} I^+(x)}{2} + \dfrac{\partial_{\ddot{\pi}} I^-(x)}{2}) \\ -\dfrac{d\partial_{\ddot{\pi}} I^-(x)}{dx} = -k_e \partial_{\ddot{\pi}} I^-(x) + k_e \lambda (\dfrac{\partial_{\ddot{\pi}} I^+(x)}{2} + \dfrac{\partial_{\ddot{\pi}} I^-(x)}{2}). \end{cases} \tag{3.69}$$

The development from Eq.3.5 to Eq.3.69 is straightforward because the properties of the medium (k_e and λ) are independent of $\ddot{\pi}$. At this stage, we consider the sensitivity $\partial_{\ddot{\pi}} I^+$ and $\partial_{\ddot{\pi}} I^-$ propagating in the system as the intensity does. Moreover, it is transported in exactly the same way as the intensity: it is also absorbed and scattered in the medium, following the properties of the medium (k_e and λ).

Boundary conditions for the sensitivity

However, the derivative of the boundary condition of intensity is less straightforward. We apply here the method developed by Lapeyre to obtain the corresponding boundary conditions for sensitivity [32].

The left boundary is not characterized by the parameter $\ddot{\pi}$. Therefore, the boundary condition on this boundary differentiated by $\ddot{\pi}$ is 0:

$$\partial_{\ddot{\pi}} I^+(0) = 0 \tag{3.70}$$

The right boundary is characterized by $\ddot{\pi}$. Therefore, its boundary condition for sensitivity is not null. It is considered a "cold black" wall because no intensity is emitted or reflected on this boundary. The boundary condition of sensitivity for a "cold black" wall is formulated as following [32]:

$$\partial_{\ddot{\pi}} I^-(\ddot{\pi}) = \partial_{\ddot{\pi}} I(x,\omega)|_{x=\ddot{\pi},\omega=-1} = -\alpha \left(-\frac{dI^-(x)}{dx} \right) \Bigg|_{x=\ddot{\pi}} \tag{3.71}$$

Figure 3.3 – Schema of v and ω. In the 3D and 2D configuration [32], $\alpha = \frac{\vec{v}\cdot\vec{n}}{\vec{\omega}\cdot\vec{n}}$ where $\vec{n}$ is the normal. This formulation can then be generalized in this 1-D example: $\alpha = \frac{v}{\omega}$.

where $\alpha = -v$ and $v = 1$, called the "velocity of spatial deformation" seeing Fig.3.3, which is distinguished from the "deformation velocity" in Sec.3.4.1. We imply the Eq.3.5 into Eq.3.71 and we obtain the boundary condition of sensitivity on $x = \ddot{\pi}$:

$$\partial_{\ddot{\pi}} I^-(\ddot{\pi}) = \frac{k_s}{2} I^+(\ddot{\pi}) \tag{3.72}$$

Finally, we obtain a boundary condition coupled with the model of intensity. As mentioned, we can apply the principle of *Double Randomization* to solve the two coupling models.

Alternatively, a full development of this boundary condition is detailed below:

Development of the boundary condition for the sensitivity [32–34]

In order to obtain the boundary condition for the sensitivity in this example, we need to define a material space $\mathcal{M}$ and a geometrical space $\mathcal{G}$ for the point of boundary (where $x = \ddot{\pi}$).
In general, the material space keeps the material's properties and is independent of the deformation (independent of $\ddot{\pi}$ in this case). While in the material space, we note this point as $\vec{y} \in \mathcal{M}$, which is independent of $\ddot{\pi}$. While in the geometric space, we note this point as $\vec{\ddot{y}} \in \mathcal{G}$ which depends on $\ddot{\pi}$: $\vec{\ddot{y}} \equiv \vec{\ddot{y}}(\ddot{\pi})$.
The two spaces are then linked by a function Z:

$$\vec{\ddot{y}} = Z(\vec{y}, \ddot{\pi}) = \ddot{\pi} \tag{3.73}$$

We keep the notation $Z(\vec{y}, \ddot{\pi})$, because in a general case, the function Z is a function of $\vec{y}$.
Also, we denote the radiative intensity in geometric space as $I^-(\vec{\ddot{y}}, \ddot{\pi})$ and $I^+(\vec{\ddot{y}}, \ddot{\pi})$ and in material space as $L^-(\vec{y}, \ddot{\pi})$ and $L^+(\vec{y}, \ddot{\pi})$.
Therefore, we have:

$$\begin{cases} I^+(\vec{\ddot{y}}, \ddot{\pi}) = L^+(\vec{y}, \ddot{\pi}) \\ I^-(\vec{\ddot{y}}, \ddot{\pi}) = L^-(\vec{y}, \ddot{\pi}) \end{cases} \tag{3.74}$$

We take the derivative of Eq 3.74 with respect to $\ddot{\pi}$:

$$\begin{cases} \partial_{\ddot{\pi}} I^+(\vec{\ddot{y}}, \ddot{\pi}) = \partial_{\ddot{\pi}} L^+(\vec{y}, \ddot{\pi}) \\ \partial_{\ddot{\pi}} I^-(\vec{\ddot{y}}, \ddot{\pi}) = \partial_{\ddot{\pi}} L^-(\vec{y}, \ddot{\pi}) \end{cases} \tag{3.75}$$

The following equations are then yielded:

$$\begin{cases} \partial_1 I^+(\vec{\ddot{y}}, \ddot{\pi}) \partial_{\ddot{\pi}} \vec{\ddot{y}} + \partial_2 I^+(\vec{\ddot{y}}, \ddot{\pi}) = \partial_{\ddot{\pi}} L^+(\vec{y}, \ddot{\pi}) \\ \partial_1 I^-(\vec{\ddot{y}}, \ddot{\pi}) \partial_{\ddot{\pi}} \vec{\ddot{y}} + \partial_2 I^-(\vec{\ddot{y}}, \ddot{\pi}) = \partial_{\ddot{\pi}} L^-(\vec{y}, \ddot{\pi}) \end{cases} \tag{3.76}$$

$\partial_1 I^-(\vec{\ddot{y}}, \ddot{\pi})$ and $\partial_1 I^+(\vec{\ddot{y}}, \ddot{\pi})$ are the spatial gradient of I^+ and I^-, which can be replaced by the collision terms in RTE (Eq.3.5). $\partial_{\ddot{\pi}} \vec{\ddot{y}}$ is the derivative of function Z: $\partial_{\ddot{\pi}} \vec{\ddot{y}} = \partial_{\ddot{\pi}} Z = 1$ and the source of intensity on the boundary is independent to $\ddot{\pi}$, therefore $\partial_{\ddot{\pi}} L^+ = \partial_{\ddot{\pi}} L^- = 0$.
Therefore, now we have the following:

$$\begin{cases} \partial_2 I^+(\vec{\ddot{y}}, \ddot{\pi}) = k_e I^+(\vec{\ddot{y}}) - \frac{k_s}{2} I^+(\vec{\ddot{y}}) - \frac{k_s}{2} I^-(\vec{\ddot{y}}) \\ \partial_2 I^-(\vec{\ddot{y}}, \ddot{\pi}) = -k_e I^-(\vec{\ddot{y}}) + \frac{k_s}{2} I^+(\vec{\ddot{y}}) + \frac{k_s}{2} I^-(\vec{\ddot{y}}) \end{cases} \tag{3.77}$$

with $\vec{\ddot{y}} = \ddot{\pi}$:

$$\partial_{\ddot{\pi}} I^-(\ddot{\pi}) = \partial_2 I^-(\vec{\ddot{y}}, \ddot{\pi})|_{\vec{\ddot{y}} = \ddot{\pi}} = \frac{k_s}{2} I^+(\ddot{\pi}) \tag{3.78}$$

The transport of sensitivity

Herein, the model of sensitivity is built. We describe the transport of sensitivity and compare it with the transport of intensity. For the intensity, Eq.3.5 describes its transport in the medium and Eq.3.6 describes its boundary conditions. A schema of its transport model is shown in Fig.3.4. A laser gun emits intensity on the left boundary, whereas the source of intensity is null on the right boundary towards the left direction. The intensity travels inside the rod, being absorbed or scattered.

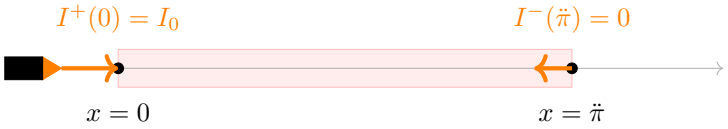


Figure 3.4 – The schema of the transport of intensity with a source of intensity (a laser gun) on the left boundary.

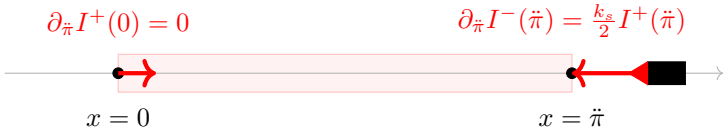


Figure 3.5 – The schema of the transport of intensity with a source of sensitivity (a laser gun of sensitivity) on the right boundary.

Similarly, the transport of sensitivity in the medium is described by the Eq.3.69 and the boundary conditions of sensitivity are described in Eq.3.70 and Eq.3.71. A schema of its model of transport is shown in Fig.3.5. Since the parameter $\ddot{\pi}$ characterizes the rod's right boundary, the sensitivity source is located on the right boundary. We can imagine that a laser of sensitivity is set on the right side of the rod. On the contrary, the left boundary is not characterized by $\ddot{\pi}$. Therefore, the source of sensitivity is null on the point $x = 0$. Similar to the intensity, the sensitivity travels in the rod, absorbed and scattered.

Solve the model of sensitivity

The model of sensitivity is similar to the model of intensity. In this example, we can use exactly the same mathematical tools to solve the model of sensitivity.

We found that the corresponding Fredholm Equation is exactly the same as Eq.3.59. Therefore, we write exactly the same integral formulation as a shape of expected value of $\partial_{\ddot{\pi}} I^+$ and $\partial_{\ddot{\pi}} I^-$ as Eq.3.60 and Eq.3.61. Consequently, the algorithm corresponding to this method is exactly the same as Algo.3.

3.5 Results and discussions

3.5.1 Intensity

The results of estimations for $I^+(x)$ and $I^-(x)$ in the rod are plotted in Fig.3.6 and Fig.3.7 for two cases of different k_e. Each point in the figure corresponds to one estimation by Algo.1. The analytical results are also plotted in the figure, which is well correlated with the results of estimations.

Furthermore, the results show the tendency of the *law of beer lambert*. Since the source of intensity exists only on the left side, the two figures show the tendency of attenuation of intensity. For the first case (Fig.3.6), the intensity is attenuated almost linearly with a reasonable small value of k_e. The medium is "thin" in this case. For the second case (Fig.3.7), the intensity is attenuated almost exponentially, with a reasonable big value of k_e. The medium is "thick" in this case.

3.5.2 Sensitivity

The results of estimations for $\partial_{\ddot{\pi}} I^+(x)$ and $\partial_{\ddot{\pi}} I^-(x)$ are plotted in Fig.3.8 and Fig.3.9 for two cases of different k_e. Each point in the figure corresponds to one estimation by Algo.2 or Algo.3. Results obtained by the three methods fit the analytical solution.

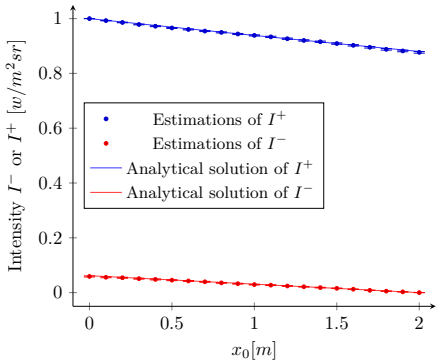


Figure 3.6 – Estimations of intensity ($\lambda = 0.7$, $k_e = 0.1m^{-1}$, $\ddot{\pi} = 2m$, $n_{MC} = 10^4$, $I_0 = 1w/m^2sr$)

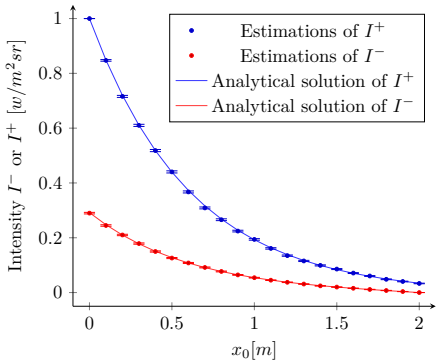
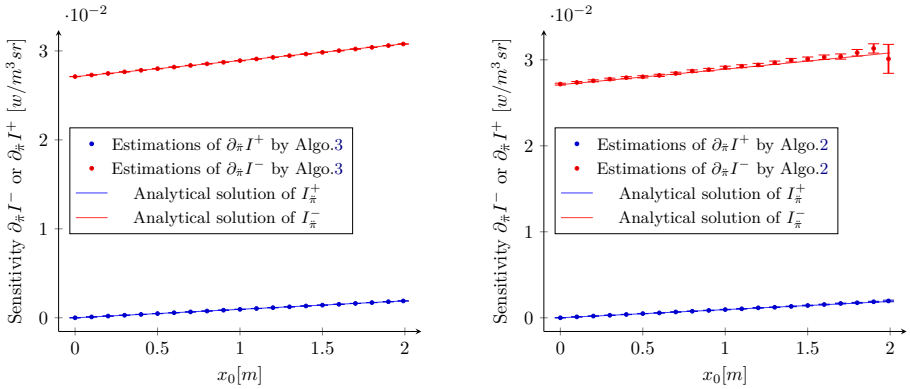


Figure 3.7 – Estimations of intensity ($\lambda = 0.7$, $k_e = 3m^{-1}$, $\ddot{\pi} = 2m$, $n_{MC} = 10^4$, $I_0 = 1w/m^2sr$)

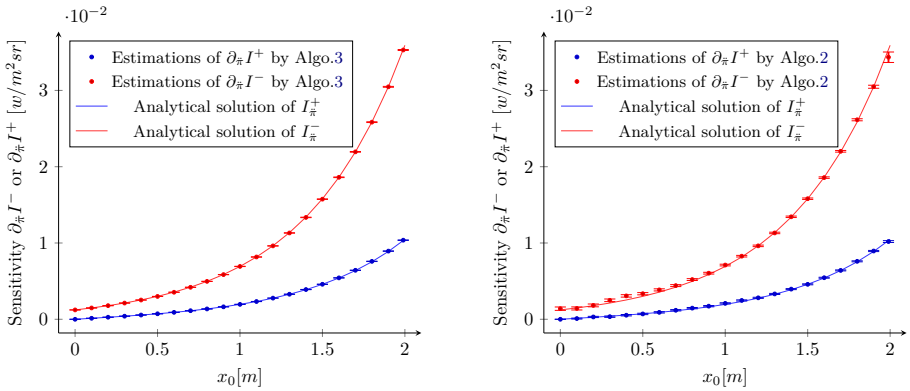
It is interesting to find the tendency of *law of beer lambert* for the propagation of sensitivity. As shown in Fig.3.5, a sensitivity source appears on the rod's right boundary. Therefore, a nearly linear attenuation of sensitivity is found in Fig.3.8 because of the "thin" medium, while for the sensitivity, it is attenuated from right to left. Similarly, an exponential attenuation of sensitivity is observed in Fig.3.9 because of the "thick" medium from the right side to the left.

Last but not least, it is observed that the sensitivity estimations for the points near the right boundary converge slower for the method of observable derivation. Further studies are needed to explain it.



(a) The method of sensitivity model and the method of differentiable rendering **(b)** The method of observable derivation

Figure 3.8 – Estimations of sensitivity ($\lambda = 0.7$, $k_e = 0.1m^{-1}$, $\ddot{\pi} = 2m$, $n_{MC} = 10^6$, $I_0 = 1w/m^2 sr$)



(a) The method of sensitivity model and the method of differentiable rendering **(b)** The method of observable derivation

Figure 3.9 – Estimations of sensitivity ($\lambda = 0.7$, $k_e = 3m^{-1}$, $\ddot{\pi} = 2m$, $n_{MC} = 10^4$, $I_0 = 1w/m^2 sr$)

3.6 Conclusion of the chapter

In this chapter, we investigate three methods to compute the sensitivity of intensity in a one-dimensional radiative case study. The analytical solution is known for comparison purposes. RTE is used to describe the transport of intensity in the medium, and the boundary

conditions are set. The medium is assumed to be cold with homogeneous absorption and scattering properties. Two Dirichlet boundary conditions are set on the boundaries. After building the transport model, the observable I is formulated into a Fredholm equation using standard mathematical tools. After that, the Fredholm Equation is transformed into an expected value. Finally, as mentioned in Chapter.1, the expected value can be estimated by the Monte Carlo method. Moreover, the Monte Carlo algorithm helps build a physical image since each path mimics the transport of photons.

After that, three methods of estimating sensitivity are introduced and compared. The method of observable derivation relies on differentiating directly the observable in an expected value form. The method of differentiable rendering is based on the derivation of the observable in a Fredholm Equation form. The resulting formulation is then reformulated into an expected value form. The last method relies on differentiating the model of intensity to build the model of sensitivity. The sensitivity of intensity becomes the descriptor of the sensitivity model, absorbed and scattered in the system. The resulting sensitivity model is then solved, having the same process as solving the model of intensity. The model of intensity and sensitivity are coupled, but this is algorithmically handled by Double Randomization.

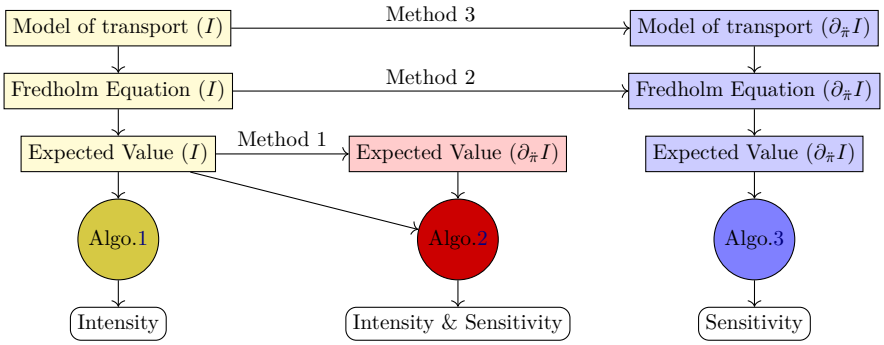


Figure 3.10 – A roadmap for the estimation of sensitivity with method 1: method of observable derivation; method 2: method of differentiable rendering; method 3: method of sensitivity model.

Fig.3.10 sums up the differences between the three methods we investigated in this chapter. Compared to the other two methods, the method of observable derivation is perfectly vectorized to simultaneously estimate the intensity and the sensitivity (as discussed in Chapter.2). However, the corresponding formal development is very demanding. Few applications are developed following this method [9].

The method of differentiable rendering is more relevant than the former method to address complex geometries. More applications are found to improve the reconstruction of 3D scenes and designing materials with specified optical properties [44]. However, it loses the advantage of vectorization.

The method of sensitivity model shares the same algorithm as the method of differentiable rendering and remains adapted for more complex applications. Furthermore, this method is of great interest to physicians and engineers because of the attached physical image. The sensitivity can be described physically as the intensity helping physicians and engineers better understand their systems.

However, this method also loses the advantage of vectorization. A radiative system can be characterized by thousands of parameters (for example, the concentrated solar power system). Sensitivities of all these parameters are of great importance for optimization purposes. In order to estimate all the sensitivities, we need to build a model of sensitivity for each of them. If we estimate them one by one, it becomes very costly for calculation resources. Herein,

we suggest to **partially vectorize the estimations of sensitivities**. We will study and analyze the models of sensitivity in the radiative system carefully, trying to regroup and vectorize the estimations of sensitivities.

In the following chapters, we will apply the method of sensitivity model in some geometrically complex systems. We will study and analyze the models of sensitivity carefully, regroup different kinds of sensitivities and vectorize their estimations. Nevertheless, We will also address the estimation of thousands of sensitivities that remains out of range if the vectorization technique is not applied.

4

The method of sensitivity model applied to radiative systems involving complex geometries: implementation via vectorization

4.1 Introduction

In the last chapter, we reviewed the three viewpoints (methods) of estimating sensitivities in a radiative system: the method of observable derivation, the method of differential rendering, and the method of sensitivity model. In the following parts of this thesis, we will focus on the method of sensitivity model, which is detailed in [32–34]. We choose the method of sensitivity model to achieve the objective of this thesis: building an algorithm to estimate the geometric sensitivities in a STP system. This chapter dedicates to preparing this application, which is introduced in the next chapter, by firstly applying the method of sensitivity method to cases involving complex geometries.

Compared to the method of observable derivation, as mentioned in Chapter.3, the method of sensitivity model is more flexible and more adapted to complex geometry. Indeed, De la Torre attempted to apply the method of observable derivation in CSP system [9]. However, as it is discussed in Chapter.2, the effects of blocking and shadowing in CSP system are not able to be taken into account [9] because of the difficulty of formal mathematical development.

Compared to the method of differential rendering, the method of sensitivity model physically interprets the contributions of sensitivity because it is based on transport models for sensitivities. Then, applying the method of sensitivity model instead of the method of differential rendering allows us to physically interpret the contributions of physical events (blocking, shadowing, and spillage) in CSP system to the sensitivities, as we will show in the next chapter.

Therefore, we focus on the method of sensitivity model instead of the other two methods in the following parts of this thesis.

In the previous study of Paule Lapeyre in her Ph.D. thesis, she has proposed this method of sensitivity model [32–34]. The implementation has been performed only for academic cases. We propose extending this method to real applications involving complex geometries with many parameters, such as CSP systems. In this chapter, we apply the method of sensitivity models to cases involving complex geometries described by the triangles. Complex geometries could be described by triangles. We make this choice here because we use computer graphics libraries, where the most commonly used shapes to describe complex geometries are the triangles [46].

4.1.1 Research problem of the chapter

The main difficulty of applying the method of sensitivity models to triangulated geometries is the high number of sources appearing in the models of sensitivities.

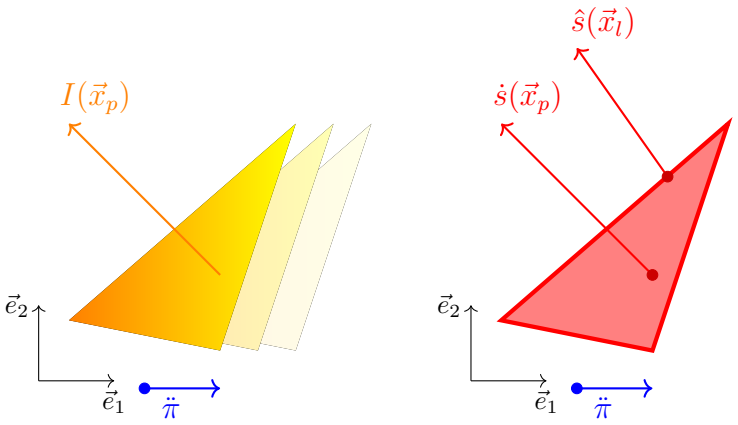
In order to illustrate this difficulty in an intuitive manner, we take a triangle as an example. en Fig.4.1, we suppose that a gradient of temperature is on the surface of a single triangle. The emitted intensity is then a function of the position on the surface $\vec{x}_p$: $I_{emit} \equiv I(\vec{x}_p)$. A geometric parameter $\ddot{\pi}$ characterizes the translation of this triangle. The perturbation of $\ddot{\pi}$ causes the perturbation of the local emitted intensity $I(\vec{x}_p)$. This local perturbation is defined as the source of sensitivity. However, two perturbations of intensity are distinguished: the perturbation on the surface of the triangle $\dot{s}(\vec{x}_p)$ and the perturbation on the segments of the triangle $\hat{s}(\vec{x}_l)$, where $\vec{x}_p$ represents the points on the surface and $\vec{x}_l$ represents the points on the segments. $\dot{s}(\vec{x}_p)$ is originated from the spatial gradient of emitted intensity (emitted intensity is a function of the position: $I(\vec{x}_p)$). $I(\vec{x}_p)$ is derivable around $\vec{x}_p$:

$$\dot{s}(\vec{x}_p) = \lim_{\Delta\ddot{\pi}\leftarrow 0} \frac{I(\vec{x}_p - \Delta\ddot{\pi}\vec{e}_1) - I(\vec{x}_p)}{\Delta\ddot{\pi}} \tag{4.1}$$

$\hat{s}(\vec{x}_l)$ originated from the discontinuous profile of emitted intensity on the borders of the triangle. However, $I(\vec{x}_l)$ is not smooth around $\vec{x}_l$, because $\vec{x}_l$ is defined on the borders. When

we differentiate the intensity on the borders, we will obtain a Dirac. $\hat{s}(\vec{x}_l)$ is studied in [34] and the detailed development of this term is shown in the following sections of this chapter.

It is just essential to note here only surface sources exist in the model of intensity, while the model of sensitivity consists of surface sources on the surface and line sources on the segments (Fig.4.1). However, complex geometry is usually described by thousands of triangles. The sources of sensitivity might locate on the surface and segments of all triangles. The question addressed in this chapter is to propose a strategy: the vectorization.



(a) Surface source of intensity.

(b) Surface source and line source of sensitivity.

Figure 4.1 – The model of intensity consists of line sources, while the model of sensitivity often consists of line sources and surface sources. $\vec{e}_1$ and $\vec{e}_2$ are the global coordinate vectors and $\ddot{\pi}$ characterises the translation of triangle following $\vec{e}_2$. $\vec{x}_p$ represents the position on the surface, where located the surface sources. $\vec{x}_l$ represents the position on the edge, where located the line sources. I is the emitted intensity, $\dot{s}$ is the surface source of sensitivity, and $\hat{s}$ is the line source of sensitivity.

4.1.2 Proposition: Vectorization

The idea of vectorization is that, instead of propagating a scalar as the descriptor of the transport model (for example, the descriptor of the model of intensity is the intensity), we propagate a vector as the descriptor of the transport. We follow the four steps to apply the method of sensitivity model to cases involving complex geometries (see also Fig.4.2):

- Firstly, we build the transport models for intensity and for all sensitivities interested in scalar forms (each sensitivity corresponds to a parameter of the system). In order to tackle the cases with triangulated and complex geometry, boundary conditions are developed for each triangle.
- Following the standard process described in Chapter.3, we formulate the observable by an integral of intensity, and we formulate the sensitivities of the observable by integrals of sensitivities of intensity (into a form of Fredolhm Equation and then a form of expected value).
- After that, we analyze the path spaces corresponding to the integral formulations (for the observable and for all the sensitivities of the observable).
- We distinguish the sources belonging to the same path space and vectorize the transport of intensity or sensitivities with the same optical paths.

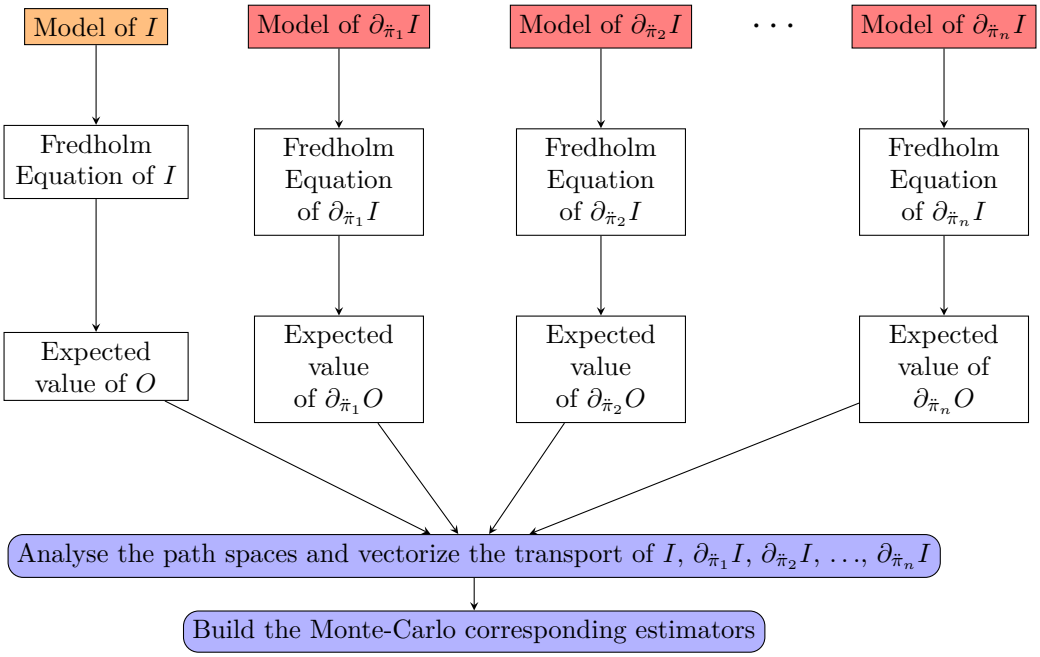


Figure 4.2 – A roadmap of vectorization. I is the intensity, the descriptor of the model of intensity. $\ddot{\pi}_1$, $\ddot{\pi}_2$, …, $\ddot{\pi}_n$ are the parameters of the system and $\partial_{\ddot{\pi}_1} I$, $\partial_{\ddot{\pi}_2} I$, …, $\partial_{\ddot{\pi}_n} I$ are the corresponding sensitivities, which are the descriptors of the corresponding models of sensitivities. This strategy consists of formulating the expected value form of the observable O, and its sensitivity $\partial_{\ddot{\pi}_1} O$, $\partial_{\ddot{\pi}_2} O$, …, $\partial_{\ddot{\pi}_n} O$ separately. After that, we analyse the path spaces to vectorize the transport of I, $\partial_{\ddot{\pi}_1} I$, $\partial_{\ddot{\pi}_2} I$, …, $\partial_{\ddot{\pi}_n} I$.

4.1.3 Structure of the chapter

This strategy will be applied to two examples of applications involving complex and triangulated geometries: To apply the process defined in Fig.4.2, we have chosen two examples that involve complex geometries that have been triangulated:

- The first example concerns a diffuse multi-reflection case. We are interested in estimating the observable $I(\vec{x}_{obs}, \vec{\omega}_{obs})$ and its sensitivity with respect to the unique reflectivity of the geometry $\partial_\rho I(\vec{x}_{obs}, \vec{\omega}_{obs})$, where $\vec{x}_{obs}$ is the point of observation, ρ the reflectivity of the diffuse reflection and $\vec{\omega}_{obs}$ is the direction of observation . The two estimations will be vectorized following the proposed strategy in Fig.4.2. Once the estimations of I and $\partial_\rho I$ are vectorized, an image of I and an image of s_ρ can be rendered simultaneously. In the computer graphics community, the image of sensitivity is used for inversion processes and machine learning process [26]

- The second example concerns an emitter in a semi-transparent medium. We are interested firstly in estimating the observable P, which is the impacting power on a receiver in the radiative system. We are also interested in estimating the sensitivities of the power $\partial_{\ddot{\boldsymbol{\pi}}} P$, with respect to the translations and rotations of the emitter, where $\ddot{\boldsymbol{\pi}}$ is the vector of parameters that characterize the translations and rotations of the emitter. After building the models of intensity and its sensitivities, the observable and sensitivities are formulated into an integral form. The path spaces of estimating the observable and the sensitivities are analyzed. Finally, vectorization is performed for the optical paths of different descriptors (intensity and sensitivities) in the same path space.

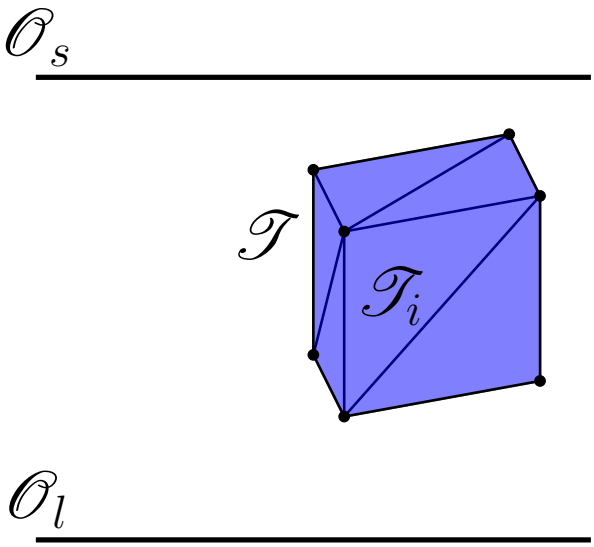


Figure 4.3 – Configuration of the example. The object is presented by a triangulated cube in the figure, but it can be replaced by any triangulated shape.

4.2 First Example

4.2.1 Description of the case study

The configuration is shown in Fig.4.3. The transport of I in the medium is governed by a partial differential equation at any locations $\vec{x}$, following a vector of direction $\vec{\omega}$ in a unit sphere $\mathcal{S}$ (RTE). The model is closed by the two boundaries: the "sky" (noted $\mathscr{O}_s$) and the "ground" (noted $\mathscr{O}_l$). An "object" described by triangles are placed between $\mathscr{O}_s$ and $\mathscr{O}_l$. A triangle is noted $\mathscr{T}_i$ with the index i for i^{th} triangle, and the set of all $\mathscr{T}_i$ composes the "object" noted $\mathscr{T} \equiv \{\mathscr{T}_i\}$, $\forall i \in \{1, 2, \ldots, n_{\mathscr{T}}\}$, where $n_{\mathscr{T}}$ is the number of triangles.

The following assumptions are imposed:

1. The medium is transparent.
2. $\mathscr{O}_l$ is a cold black body that does not emit or reflect any intensity.
3. $\mathscr{O}_s$ is a black body that does not reflect any intensity but emits intensity.
4. $\mathscr{T}$ is a cold opaque body with a surface of diffuse reflection. The reflectivity ρ is homogeneous for all the surfaces of $\mathscr{T}$.

Herein we have an objective of estimating two values on the point of observation in the system $\vec{x}$, following a direction of observation $\vec{\omega}$.

1. The observable is the intensity $I(\vec{x}, \vec{\omega})$.
2. The sensitivity of the observable with respect to ρ is $s_\rho(\vec{x}, \vec{\omega}) \equiv \partial_\rho I(\vec{x}, \vec{\omega})$.

We will build two models of transport, one for s_ρ and the other for I, and estimate them by the Monte Carlo method. After that, we analyze the path spaces of estimating I and s_ρ and vectorize the two estimations.

4.2.2 Transport model of intensity with the boundary condition for each triangle

Since the medium is considered transparent, the RTE for intensity is written as:

$$\vec{\omega} \cdot \partial_1 I(\vec{x}, \vec{\omega}, \rho) = 0 \tag{4.2}$$

The boundary conditions for $\mathscr{O}_l$ and $\mathscr{O}_s$ are noted with an index of O_s and O_l respectively:

$$\begin{cases} I(\vec{x}_{O_s}, \vec{\omega}_{O_s}) = \mathring{I}(\vec{x}_{O_s}, \vec{\omega}_{O_s}) = I_0 \\ I(\vec{x}_{O_l}, \vec{\omega}_{O_l}) = 0 \end{cases} \tag{4.3}$$

A source of intensity is located on the boundary condition of $\mathscr{O}_s$. To distinguish the source of intensity and the transport of intensity, we note $\mathring{I}$ for the source on the boundary. In this example, the source of intensity is constant I_0.

After that, we build the boundary condition of intensity for each triangle. The boundary condition of intensity on the i^{th} triangle $\mathscr{T}_i$ is noted I_i, with the index p for the position $\vec{x}_p$ and direction $\vec{\omega}_p$:

$$I_i(\vec{x}_p, \vec{\omega}_p, \rho) = \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega'_p}(\vec{\omega}'_p) d\vec{\omega}'_p \rho I(\vec{x}_p, \vec{\omega}'_p, \rho) \tag{4.4}$$

with

$$\mathscr{P}_{\Omega'_p}(\vec{\omega}'_p) = |\vec{\omega}'_p \cdot \vec{n}_i| \tag{4.5}$$

which is the normalized pdf ($\int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p = 1$). The product of $\rho\mathscr{P}_{\Omega'_p}$ is also called BRDF of a diffuse surface, $\vec{\omega}'_p$ the direction of the incoming intensity at the point $\vec{x}_p$ and $\vec{n}_i$ is the normal of the triangle $\mathscr{T}_i$.

The outgoing intensity of the i^{th} triangle at the point $\vec{x}_p$ following the direction $\vec{\omega}_p$ is the reflected intensity at the same point following the incoming direction $\vec{\omega}'_p$, multiplied by the reflectivity ρ. A physical image of the reflection on the triangle is shown in Fig.4.4. To be noted that the reflection is diffuse (Eq.4.5). Therefore the reflected direction $\vec{\omega}_p$ is independent of the incoming direction $\vec{\omega}'_p$.

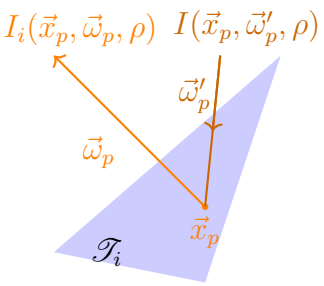


Figure 4.4 – Reflection of the intensity on the i^{th} triangle. This figure shows a physical image of reflection governed by Eq.4.4.

4.2.3 Estimation of the observable

A model of intensity is built at this stage. It is then possible to measure the observable in the system, which is the intensity I of a point $\vec{x}$, following the propagation direction $\vec{\omega}$: $I(\vec{x},\vec{\omega})$.

Following the model of I described in Sec.4.2.2, the observed intensity $I(\vec{x},\vec{\omega})$ can be formulated as:

$$I(\vec{x},\vec{\omega},\rho) = I_i(\vec{x}_p,\vec{\omega}_p,\rho)\mathcal{H}(\vec{x}_{hit}\in\mathscr{T}_i) + I_0\mathcal{H}(\vec{x}_{hit}\in\mathscr{O}_s) + 0\mathcal{H}(\vec{x}_{hit}\in\mathscr{O}_l) \tag{4.6}$$

where $\vec{x}_{hit}$ is the first intersection of the ray-tracing test starting from $\vec{x}$, following $-\vec{\omega}$ and $\mathcal{H}$ is the Heaviside function. Eq.4.6 means that the intensity $I(\vec{x},\vec{\omega},\rho)$ has three possible origins. It can be from the three boundary conditions, seeing Fig.4.5.

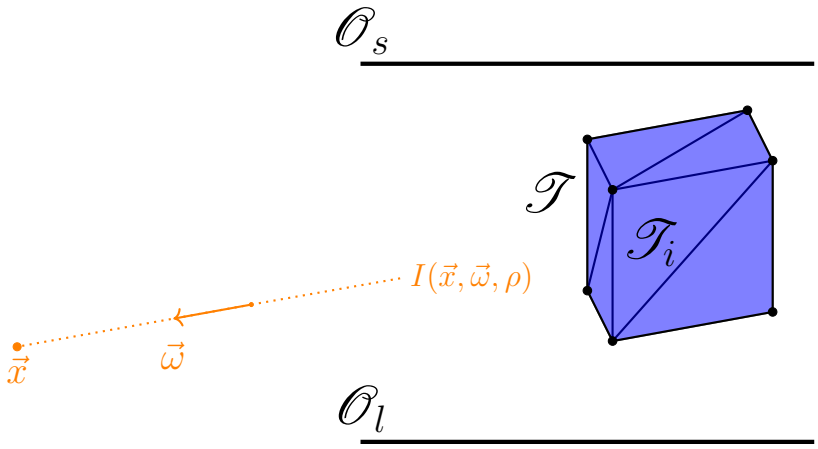


Figure 4.5 – The observable is $I(\vec{x},\vec{\omega},\rho)$, the intensity propagated from $\vec{x}$, following $\vec{\omega}$. Three possible events can happen: the ray hits the $\mathscr{O}_s$, and $I(\vec{x},\vec{\omega},\rho) = I_0$; the ray hits the $\mathscr{O}_l$ and $I(\vec{x},\vec{\omega},\rho) = 0$; the ray hits the $\mathscr{T}_i$, $I(\vec{x},\vec{\omega},\rho) = I_i(\vec{x}_p,\vec{\omega}_p,\rho)$.

We finally substitute the boundary condition Eq.4.4 into Eq.4.6 and we reformulate it iteratively:

$$\begin{aligned} I(\vec{x},\vec{\omega},\rho) &= I_0\mathcal{H}(\vec{x}_{0,hit}\in\mathscr{O}_s) \\ &+ \mathcal{H}(\vec{x}_{0,hit}\in\mathscr{T}_i)\int_{2\pi(\vec{n}_i)} \mathcal{P}_{\Omega'_p}(\vec{\omega}'_{p1})d\vec{\omega}'_{p1}\left[\rho I_0\mathcal{H}(\vec{x}_{1,hit}\in\mathscr{O}_s) + \mathcal{H}(\vec{x}_{1,hit}\in\mathscr{T}_i)\dots\right] \end{aligned} \tag{4.7}$$

where $\vec{x}_{0,hit}$ and $\vec{x}_{1,hit}$ are the points of intersection after 0 and 1 reflection; $\vec{\omega}'_{p1}$ is the direction of propagation after 1 reflection.

At this stage, the observable $I(\vec{x},\vec{\omega})$ is formulated into a form of integral, which is then possible to be estimated by a Monte-Carlo algorithm. For each realization, we proceed with a reverse ray-tracing process, starting from the point $\vec{x}$, following the direction $-\vec{\omega}$. If the first intersection is found on $\mathscr{O}_s$, we count I_0 for this realization; if the first intersection is found on $\mathscr{O}_l$, we count 0; if the first intersection is found on $\mathscr{T}_i$, we will continue to search for the incoming intensity, sampling a direction following $\mathcal{P}_{\Omega'_p}$. Systematically, for each Monte-Carlo realization, we note $\vec{x}_{k,hit}$, $\forall k\in\mathbb{N}_0$ as the point of intersection after k reflections, when $\vec{x}_{k,hit}\in\mathscr{O}_s$, we count $\rho^k I_0$ for this realization; when $\vec{x}_{k,hit}\in\mathscr{O}_l$, we count 0; when $\vec{x}_{k,hit}\in\mathscr{T}$, we continue the $(k+1)^{th}$ reflection.

Before going further, we first build the model of transport for the sensitivity and formulate the sensitivity of observable into an integral form.

4.2.4 Transport model of sensitivity with the boundary condition for each triangle

In this subsection, we build the transport model for sensitivity s_ρ. With this transport model, the sensitivity s_ρ is considered as a physical quantity that is transported in the system. The transport model of sensitivity is built based on differentiating the transport model of intensity mathematically, but we propose to give an image of the "transport of sensitivity" at each stage.

We differentiate Eq.4.2 to obtain the RTE for sensitivity:

$$\vec{\omega} \cdot \partial_1 s_\rho(\vec{x}, \vec{\omega}, \rho) = 0 \tag{4.8}$$

A physical image of transport of sensitivity can be created based on Eq.4.8. Comparing to Eq.4.2, we can imagine that the medium is also "transparent" for s_ρ:

Secondly, we differentiate then the boundary conditions for $\mathscr{O}_l$ and $\mathscr{O}_s$ (Eq.4.3).

$$\begin{cases} s_\rho(\vec{x}_{O_s}, \vec{\omega}_{O_s}) = 0 \\ s_\rho(\vec{x}_{O_l}, \vec{\omega}_{O_l}) = 0 \end{cases} \tag{4.9}$$

Since the boundary conditions of $\mathscr{O}_l$ and $\mathscr{O}_s$ are not characterised by the parameter ρ, the boundary conditions on $\mathscr{O}_s$ and $\mathscr{O}_l$ are null. Based on Eq.4.9, we propose an image of "radiative physics" for sensitivity. We consider that these boundaries are "cold black" bodies for the sensitivity s_ρ. The sensitivity s_ρ arriving on these two boundaries is then "absorbed", and no sensitivity is "emitted" on these boundaries.

After that, we build the boundary condition of sensitivity for each triangle of the object $\mathscr{T}$. We differentiate the boundary conditions in Eq.4.4 with respect to ρ to obtain the boundary conditions of s_ρ, where $s_{\rho i}$ is the boundary condition of sensitivity for the i^{th} triangle $\mathscr{T}_i$:

$$s_{\rho_i}(\vec{x}_p, \vec{\omega}_p, \rho) = \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega'_p}(\vec{\omega}'_p) \rho s_\rho(\vec{x}_p, \vec{\omega}'_p, \rho) d\vec{\omega}'_p + \mathring{s}(\vec{x}_p, \vec{\omega}'_p, \rho) \tag{4.10}$$

In Eq.4.10, two terms are identified: the first term is a term of transport, and the second term is a term of the source.

The first term is about the diffuse reflection of the sensitivity s_ρ, where $\mathscr{P}_{\Omega'_i}$ has been shown in Eq.4.5. This term is regarded as a reflection term to create a similar image of transport compared to the transport of intensity. Similar to the transport of intensity on the triangle (Eq.4.4), the "reflected" sensitivity $s_{\rho i}$ is the incoming sensitivity s_ρ on the same point following the incoming direction $\vec{\omega}'_i$ and multiplied by the reflectivity ρ.

The second term $\mathring{s}_\rho$ is the source of sensitivity:

$$\mathring{s}_{\rho i}(\vec{x}_p, \vec{\omega}_p, \rho) = \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega'_p}(\vec{\omega}'_p) I(\vec{x}_p, \vec{\omega}'_p, \rho) d\vec{\omega}'_p \tag{4.11}$$

Unlike the first term, the source of sensitivity (Eq.4.11) is not a function of sensitivity s_ρ but a function of the intensity. We remind here the physical definition of the sensitivity: the local perturbation of the intensity due to the perturbation of the corresponding parameter. Therefore, the source of sensitivity is the origin of the sensitivity of the system. In other words, it is the origin of the perturbation of intensity due to the perturbation of the corresponding parameter. In our example, the parameter is the reflectivity ρ. The perturbation of ρ perturbs

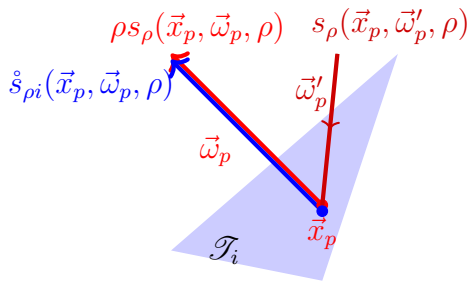


Figure 4.6 – Reflection of the sensitivity on the i^{th} triangle. This figure shows a physical image of reflection governed by Eq.4.10. The incoming sensitivity s_ρ is reflected with a supplementary source $\mathring{s}_{\rho i}$.

the reflected intensity on the triangle. Therefore, the source of sensitivity $\mathring{s}_{\rho i}$ characterizes this perturbation of reflected intensity due to the perturbation of ρ. It is then a function of the incoming intensity $I(\vec{x}_p, \vec{\omega}'_p, \rho)$ (Eq.4.11).

At this stage, we have built a transport model of sensitivity s_ρ. It is important to note that a physical image of the transport of sensitivity is proposed. In the medium, the sensitivity is transported following Eq.4.8. The medium is considered transparent for the sensitivity. On the boundaries of $\mathscr{O}_s$ and $\mathscr{O}_l$, the arriving sensitivities are absorbed, and no sensitivities are emitted. On the triangles $\mathscr{T}_i$, the sensitivity arriving is reflected, and the triangles emit also sensitivity (Fig.4.6). Finally, the sensitivity of the observable, which is the sensitivity $s_\rho(\vec{x}, \vec{\omega}, \rho)$ on a point $\vec{x}$, following the direction $\vec{\omega}$, can be estimated following the model developed.

4.2.5 Estimation of the sensitivity of the observable

Following the model of s_ρ described in Sec.4.2.4, the observed sensitivity s_ρ can be formulated as:

$$s_\rho(\vec{x}, \vec{\omega}, \rho) = s_{\rho i}(\vec{x}_p, \vec{\omega}_p, \rho)\mathcal{H}(\vec{x}_{hit} \in \mathscr{T}_i) + 0\mathcal{H}(\vec{x}_{hit} \notin \mathscr{T}) \tag{4.12}$$

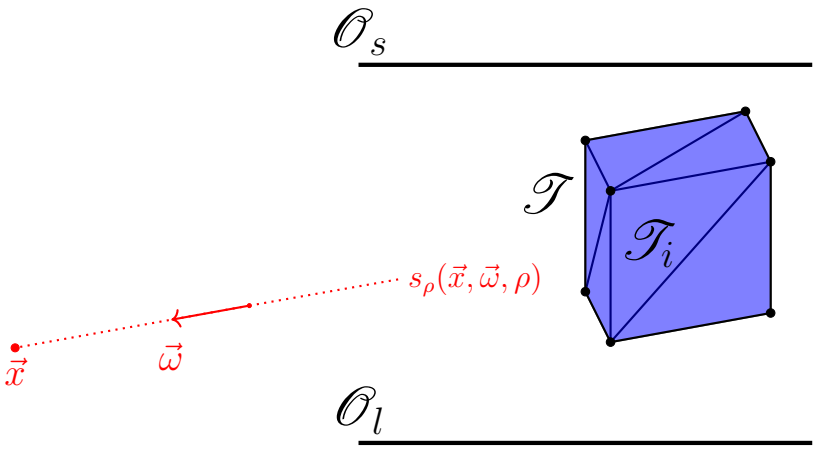


Figure 4.7 – The sensitivity of observable is $s_\rho(\vec{x}, \vec{\omega}, \rho)$, the sensitivity of intensity propagated from $\vec{x}$, following $\vec{\omega}$. Two possible events can happen: the ray hits the $\mathscr{O}_s$ or $\mathscr{O}_l$, then $s_\rho(\vec{x}, \vec{\omega}, \rho) = 0$; the ray hits the $\mathscr{T}_i$, $s_\rho(\vec{x}, \vec{\omega}, \rho) = s_{\rho i}(\vec{x}_p, \vec{\omega}_p, \rho)$.

Since the medium is transparent for the sensitivity, the observed sensitivity $s_\rho(\vec{x},\vec{\omega},\rho)$ directly comes from the boundary conditions (Fig.4.7). We perform the reverse ray-tracing test from $\vec{x}$, following $-\vec{\omega}$. Eq.4.12 signifies the fact that 2 events can happen when a ray is launched from the position $\vec{x}$ towards the direction $-\vec{\omega}$: hitting or not hitting the object $\mathscr{T}$ (Fig.4.7). When the ray hits $\mathscr{T}$, the boundary condition on the triangle $s_{\rho i}$ is referred to Eq.4.10. Herein, we rewrite the boundary condition on the triangles, using the same pdf for the transport of sensitivity and the transport of intensity:

$$s_{\rho_i}(\vec{x}_p,\vec{\omega}_p,\rho)=\int_{2\pi(\vec{n}_i)}\mathscr{P}_{\Omega'_p}(\vec{\omega}'_p)\left[\rho s_\rho(\vec{x}_p,\vec{\omega}'_p,\rho)+I(\vec{x}_p,\vec{\omega}'_p,\rho)\right]d\vec{\omega}'_p \tag{4.13}$$

Eq.4.13 is then exactly equivalent to Eq.4.10. The reason that we propose this reformulation is that Eq.4.10 separates the transport term (the reflection) and the source term to indicate a physical image of the transport of the sensitivity, as it is shown in Fig.4.6. Herein, Eq.4.13 allows to study the path space of the transport of s_ρ and I. The transport of the two quantities shares the same path space, which is indicated by Eq.4.13. The reflected sensitivity $s_{\rho i}(\vec{x}_p,\vec{\omega}_p,\rho)$ is the incoming sensitivity $s_\rho(\vec{x}_p,\vec{\omega}_p,\rho)$ multiplied by the reflectivity, plus the intensity coming from the same direction $(\vec{\omega}'_p)$, the corresponding physical image is Fig.4.8. The incoming sensitivity s_ρ and intensity I are then associated to another ray-tracing test, which are indicated in Eq.4.6 and Eq.4.12. It is important to note here that the two ray-tracing tests, one for sensitivity and the other one for sensitivity, can be carried by only one ray, as it is shown in Fig.4.8. Therefore, the transport of the sensitivity s_ρ and the intensity I in the medium and on the boundaries is exactly the same.

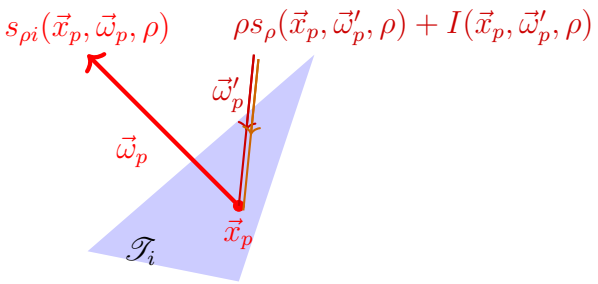


Figure 4.8 – Reflection of the sensitivity on the i^{th} triangle. This figure shows a physical image of reflection governed by Eq.4.13. The reflected sensitivity $s_{\rho i}$ is the sum of the sensitivity incoming multiplied by the reflectivity ρs_ρ and the incoming intensity following the same direction $\vec{\omega}'_p$.

Finally, based on Eq.4.12, we substitute Eq.4.13, Eq.4.6 and Eq.4.4 recursively, we get the following iterative formulation for the sensitivity at point $\vec{x}$, following $\vec{\omega}$:

$$\begin{aligned}s_\rho(\vec{x},\vec{\omega},\rho)=\,&0\mathscr{H}(\vec{x}_{0,hit}\notin\mathscr{T})\\&+\mathscr{H}(\vec{x}_{0,hit}\in\mathscr{T}_i)\int_{2\pi(\vec{n}_i)}\mathscr{P}_{\Omega_p}(\vec{\omega}_{p1})d\vec{\omega}'_{p1}\big[I_0\mathscr{H}(\vec{x}_{1,hit}\in\mathscr{O}_s)+\\&\mathscr{H}(\vec{x}_{1,hit}\in\mathscr{T}_i)\int_{2\pi(\vec{n}_i)}\mathscr{P}_{\Omega_p}(\vec{\omega}_{p2})d\vec{\omega}'_{p2}\big(2\rho I_0\mathscr{H}(\vec{x}_{2,hit}\in\mathscr{O}_s)+\\&\qquad\mathscr{H}(\vec{x}_{2,hit}\in\mathscr{T}_i)\ldots\big)\big]\end{aligned} \tag{4.14}$$

where $\vec{x}_{0,hit}$, $\vec{x}_{1,hit}$, $\vec{x}_{2,hit}$ are the points of intersection after 0, 1 and 2 reflection(s) respectively; $\vec{\omega}_{p1}$ and $\vec{\omega}_{p2}$ are the directions of propagation after 1 and 2 reflection(s).

At this stage, the sensitivity of the observable s_ρ is also formulated into an integral form, which is possible to be estimated by a Monte-Carlo algorithm. Systematically, for each Monte-Carlo realization, if we note $\vec{x}_{k,hit}$, $\forall k \in \mathbb{N}_0$ as the point of intersection after k reflection, when $\vec{x}_{k,hit} \in \mathscr{O}_s$, we count $k\rho^{(k-1)}I_0$ for this realization; when $\vec{x}_{k,hit} \in \mathscr{O}_l$, we count 0 for this realization; when $\vec{x}_{k,hit} \in \mathscr{T}$, we continue the $(k+1)^{th}$ reflection.

4.2.6 Vectorization

The observable $I(\vec{x},\vec{\omega},\rho)$ is formulated in a integral form in Eq.4.7 in Section.4.2.3. The sensitivity of the observable $s_\rho(\vec{x},\vec{\omega},\rho)$ is formulated in a integral form in Eq.4.14 in Section.4.2.5.

It is interesting to find out that the two integral formulations have the same integral domains and the same pdfs. In other words, the path space of I and s_ρ is the same. We benefit from this fact to write Eq.4.7 and Eq.4.14 in a vectorized form:

$$\begin{aligned}\begin{bmatrix} I(\vec{x},\vec{\omega},\rho) \\ s_\rho(\vec{x},\vec{\omega},\rho)\end{bmatrix} &= \begin{bmatrix}\rho I_0 \\ 0\end{bmatrix}\mathcal{H}(\vec{x}_{0,hit}\in\mathscr{O}_s) \\ &+ \mathcal{H}(\vec{x}_{0,hit}\in\mathscr{T}_i)\int_{2\pi(\vec{n}_i)}\mathcal{P}_{\Omega_p}(\vec{\omega}_{p1})d\vec{\omega}'_{p1}\Big[\begin{bmatrix}\rho I_0 \\ I_0\end{bmatrix}\mathcal{H}(\vec{x}_{1,hit}\in\mathscr{O}_s)+ \\ &\mathcal{H}(\vec{x}_{1,hit}\in\mathscr{T}_i)\int_{2\pi(\vec{n}_i)}\mathcal{P}_{\Omega_p}(\vec{\omega}_{p2})d\vec{\omega}'_{p2}\Big(\begin{bmatrix}\rho^2 I_0 \\ 2\rho I_0\end{bmatrix} I_0\mathcal{H}(\vec{x}_{2,hit}\in\mathscr{O}_s)+ \\ &\mathcal{H}(\vec{x}_{2,hit}\in\mathscr{T}_i)\dots\Big)\Big] \end{aligned} \quad (4.15)$$

The reformulation has a significant impact on the algorithms to estimate I and s_ρ: Before formulating the vectorized form (Eq.4.15), we build different optical paths for the estimation of observable I (Fig.4.5) and its sensitivity s_ρ (Fig.4.7). After formulating the vectorized form (Eq.4.15), we build the same optical paths for the estimation of observable and its sensitivity (Fig.4.9).

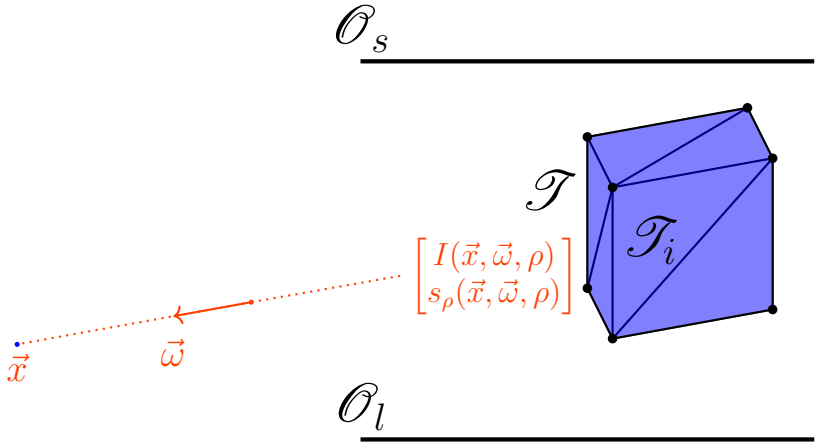


Figure 4.9 – I and s_ρ are transported in a vectorized form. Instead of launching optical paths for the estimation of I and other optical paths for the estimation of s_ρ. The same optical paths are used for the estimation of the vector of these two descriptors.

Systematically, for each Monte-Carlo realization, if we note $\vec{x}_{k,hit}$, $\forall k \in \mathbb{N}_0$ as the point of intersection after k reflections, when $\vec{x}_{k,hit} \in \mathscr{O}_s$, we count $\rho^k I_0$ for the Monte-Carlo

weight of intensity and $k\rho^{(k-1)}I_0$ for the sensitivity; when $\vec{x}_{k,hit} \in \mathscr{O}_l$, we count 0 for the Monte-Carlo weight of intensity and for the sensitivity; when $\vec{x}_{k,hit} \in \mathscr{T}$, we continue the $(k+1)^{th}$ reflection.

Finally, with this algorithm (Eq.4.15), only one ray is needed for each Monte-Carlo realization. The information on intensity and sensitivity is carried out on the same ray. It is because the path space of intensity and sensitivity is the same.

4.2.7 Results

At this stage, the estimations of observable I and its sensitivity s_ρ are vectorized. In order to show how powerful the vectorized form (Eq.4.15) is, we benefit from this vectorized form to render an image of I and another image of s_ρ simultaneously. In other words, at the same time that an image of I is rendered, the image of s_ρ can also be rendered at a price of a low supplementary computation effort. As we already indicated that the image s_ρ can serve as necessary information to process optimization and machine learning process in the community of image synthesis [26].

Since rendering an image is not the objective of this chapter, the corresponding developments and algorithms are in Appendix.B. We render the images under the Monte-Carlo ray-tracing library of Star-engine[1]. The open source codes can be found on this page[2]. Finally, the image is converted to a PGM format[3].

We remind that the method we developed in this chapter can be applied to all kinds of triangulated geometry and that the models are built generally by the boundary condition of each triangle. In this example, we use two referential triangulated geometries, standford bunny and the standford dragon[4], as the reflecting object. The corresponding image of I is shown in Fig.4.10a and Fig.4.11a. The rendered image of s_ρ is shown in Fig.4.10b and Fig.4.11b.

As mentioned in Section.4.3.1, the "sky" is a black body, the "ground" is a cold black body and the "object" has diffuse reflected surface. For the rendered image of intensity I (Fig.4.10a and Fig.4.11a), the "sky" is white, the ground is black and the object is gray. However, for the rendered image of sensitivity s_ρ (Fig.4.10b and Fig.4.11b), the "sky" and "ground" are black because they are independent of the reflectivity ρ.

The calculating time for rendering the image of intensity Fig.4.10a is 30.494s and 31.575s for Fig.4.11a. Furthermore, the calculating time for rendering the two images of Fig.4.10a and Fig.4.10b is only 34.436s. The corresponding time for rendering the two images of Fig.4.11a and Fig.4.11b is 32.385s.

To sum up, at the same time that an image of I is rendered, the image of s_ρ can also be rendered at a price of a low supplementary computation effort (around 3s more for Fig.4.10 and 2s more for Fig.4.11). All rendering algorithms that we used in this section are performed on an ordinary laptop[5]. The calculation is run parallelly on the CPUs.

[1] git@gitlab.com:meso-star/star-engine.git
[2] git@gitlab.com:hzl007x/partial_renderer.git
[3] http://netpbm.sourceforge.net/doc/pgm.html
[4] http://graphics.stanford.edu/data/3Dscanrep/
[5] 4 CPUs of i5 Intel™, 8th generation

(a) Image of intensity I **(b)** Image of sensitivity s_ρ

Figure 4.10 – $I_0 = 1\mathrm{W}/(\mathrm{m}^2\,\mathrm{sr})$ and $\rho = 0.1$. 800×600 pixels; $n_{MC} = 1024$ for each pixel. The color scale is proportional to the value of average intensity arrived on the pixel (see Appendix.B). The white color corresponds to $1\mathrm{W}/(\mathrm{m}^2\,\mathrm{sr})$ and the black color corresponds to 0.

(a) Image of intensity I. **(b)** Image of sensitivity s_ρ

Figure 4.11 – $I_0 = 1\mathrm{W}/(\mathrm{m}^2\,\mathrm{sr})$ and $\rho = 0.4$. 800×600 pixels; $n_{MC} = 1024$ for each pixel. The color scale is proportional to the value of average sensitivity arrived on the pixel (see Appendix.B). The white color corresponds to $1\mathrm{W}/(\mathrm{m}^2\,\mathrm{sr})$ and the black color corresponds to 0.

4.3 Second example

4.3.1 Description of the case study

The boundaries are closed by a "box", noted $\mathscr{B}$. A receiver $\mathscr{R}$ and a triangulated "object" are located in the box. A triangle is noted $\mathscr{T}_i$ with the index i for the i^{th} triangle, and the set of all $\mathscr{T}_i$ composes the "object" noted $\mathscr{T} \equiv \{\mathscr{T}_i\}$, $\forall i \in \{1, 2, \ldots, n_{\mathscr{T}}\}$, where $n_{\mathscr{T}}$ is the number of triangles. The configuration is shown in Fig.4.12.

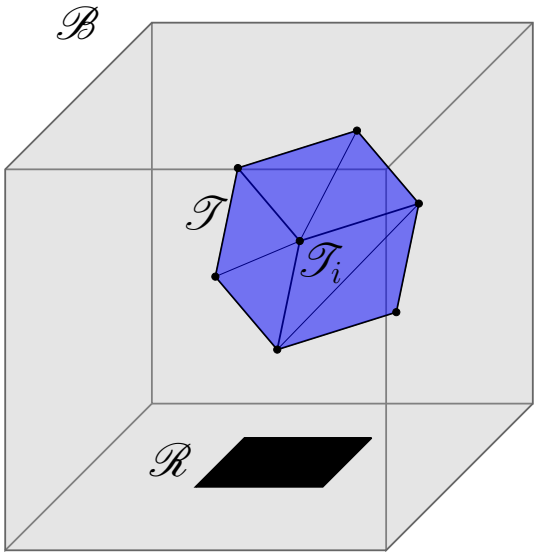


Figure 4.12 – Configuration of the example. The object is presented by a triangulated cube in the figure, but it can be replaced by any triangulated shape.

It is assumed that:

1. The medium is semi-transparent and homogeneous. The absorption is characterized by the absorption coefficient k_a, and the scattering is isotropic, characterized by the scattering coefficient k_s.
2. $\mathscr{B}$ and $\mathscr{R}$ are cold black bodies (which do not emit or reflect any intensity).
3. $\mathscr{T}$ is a black body (a Lambertian emitter) with a homogeneous temperature (which emits intensity in a spatially and angularly homogeneous way).

4.3.2 Definition of the sensitivity of intensity and the sensitivity of the observable

In this example, we aim to estimate the following:

1. The observable is the impacting power P on the receiver $\mathscr{R}$.
2. The sensitivities (gradient) of the observable with respect to a vector $\ddot{\boldsymbol{\pi}} \equiv [\ddot{\pi}_j], \forall j \in \{1, 2, 3, 4\}$, seeing Fig.4.13 for each component. We note this gradient as $\boldsymbol{S} \equiv \partial_{\ddot{\boldsymbol{\pi}}} P \equiv [S_j]$.

As it is discussed in Chapter.1, the observable is usually the integral of the descriptor (intensity). Herein, the sensitivity of the observable is also the integral of the descriptor (sensitivity of the intensity). Intensity is the descriptor of the model of intensity. The sensitivity of the intensity is the descriptor of the model of sensitivity.

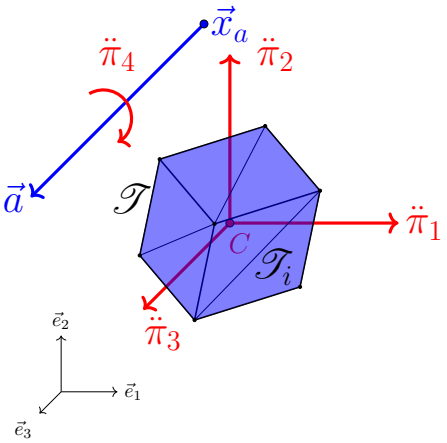


Figure 4.13 – Components of $\ddot{\boldsymbol{\pi}}$. Point C is the geometric center of $\mathscr{T}$; $\vec{e}_1$, $\vec{e}_2$ and $\vec{e}_3$ compose the global coordinate system; $\ddot{\pi}_1$, $\ddot{\pi}_2$ and $\ddot{\pi}_3$ characterise respectively the translation of $\mathscr{T}$ following $\vec{e}_1$, $\vec{e}_2$, $\vec{e}_3$; $\vec{a}$ is an axis of rotation through a fixed point $\vec{x}_a$ and $\ddot{\pi}_4$ characterizes the rotation of $\mathscr{T}$ around $\vec{a}$.

Sensitivity of the intensity

As usual, we build a model following the standard radiative transfer physics, using the intensity I as the descriptor. The transport of I in the medium is governed by a partial differential equation (RTE) at any locations $\vec{x}$ inside the volume $\mathcal{G}$, following a vector of direction $\vec{\omega}$ in a unit sphere $\mathcal{S}$. On the boundaries $\partial\mathcal{G}$, the transport of I is governed by the boundary conditions, where the locations are noted with an index. For example, in Eq.4.19, the position and direction are noted with an index r because they describe the boundary condition of the receiver $\mathscr{R}$.

The specific intensity in the system is then defined as $I \equiv I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}})$, where $\ddot{\boldsymbol{\pi}}$ is the geometric vector composed of $\ddot{\pi}_j$ (see Fig.4.13). $\vec{x}$, $\vec{\omega}$ and $\ddot{\boldsymbol{\pi}}$ are independent variables and the vector $\boldsymbol{s}$ is defined as the derivative of I with respect to $\ddot{\boldsymbol{\pi}}$, which is called also the sensitivity of the intensity:

$$\boldsymbol{s}(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) = [s_j(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}})] = \partial_{\ddot{\boldsymbol{\pi}}} I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) \tag{4.16}$$

Sensitivity of the observable

The impacting power P is the observable. It can be formulated by the product of spatial integral and angular integral of the incoming I on the receiver $\mathscr{R}$:

$$P = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r I(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \tag{4.17}$$

where $\vec{x}_r$ is a position on the receiver, $\vec{\omega}_r$ the direction of observation, $\vec{n}_r$ the normal of the receiver $\mathscr{R}$.

Vector $\boldsymbol{S}$ is defined as the derivative of P with respect to $\ddot{\boldsymbol{\pi}}$, which can be developed in a straightforward way since $\vec{x}_r$ and $\vec{\omega}_r$ are independent of $\ddot{\boldsymbol{\pi}}$ who characterises only the object $\mathscr{T}$:

$$\boldsymbol{S}(\ddot{\boldsymbol{\pi}}) \equiv [S_j(\ddot{\boldsymbol{\pi}})] = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi(\vec{n}_i)} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r \boldsymbol{s}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \tag{4.18}$$

and the components of $\boldsymbol{S}$ are defined by:

$$S_j(\ddot{\boldsymbol{\pi}}) = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi(\vec{n}_i)} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r s_j(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \tag{4.19}$$

Therefore, the sensitivity of the power with respect to a geometric parameter S_j is formulated by the product of spatial integral and angular integral of the incoming s_j on the receiver $\mathscr{R}$. Note that S_j is the sensitivity of the power, and s_j is the sensitivity of the intensity, which is also the descriptor of the system of the sensitivity. To estimate $P(\ddot{\boldsymbol{\pi}})$ and $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$, we need to build a transport model of I and a transport model of s_j in the system. Moreover, in order to apply the method to all kinds of triangulated geometries, the boundary conditions for each triangle needed to be built.

4.3.3 Transport model of intensity for triangulated geometries

As it is discussed in the introduction of this chapter, the triangle is one of the most common shapes to describe complex geometries. In this chapter, all geometries are triangulated.

The medium is considered semi-transparent with homogeneous and isotropic properties of absorption and scattering. The RTE for intensity is written as:

$$\begin{aligned} \vec{\omega} \cdot \partial_1 I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) &= \mathcal{C}[I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}})] \\ &= -k_a I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) - k_s I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) + k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}', \ddot{\boldsymbol{\pi}}) \end{aligned} \tag{4.20}$$

where $\mathcal{C}$ is the collision operator.

The boundary conditions for $\mathscr{B}$ and $\mathscr{R}$ are noted with index of B and r. As it is mentioned previously, these boundaries are cold and black:

$$\begin{cases} I(\vec{x}_B, \vec{\omega}_B) = 0 \\ I(\vec{x}_r, \vec{\omega}_r) = 0 \end{cases} \tag{4.21}$$

The boundary condition for $\mathscr{T}_i$ is noted with an index of p. The object $\mathscr{T}$ is a black body:

$$I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \mathring{I}_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = I_0 \tag{4.22}$$

Similar to the last example, we distinguish $\mathring{I}$ as the source and I as the transport of intensity in the medium or on the boundary. The triangle $\mathscr{T}_i$ has only a source term. It does not reflect the coming intensity.

At this stage, the transport model of intensity is built, with the transport equation in the medium (the RTE in Eq.4.20), the boundary conditions of $\mathscr{B}$ and $\mathscr{R}$ (Eq.4.21), as well as the boundary condition for each triangle. In the following subsection, we build the model of transport for sensitivity.

4.3.4 Transport model of sensitivity with the boundary condition of each triangle

Transport equation in the medium

We differentiate Eq.4.20 with respect to a component of $\ddot{\boldsymbol{\pi}}$ and we obtain the RTE for the sensitivity:

$$\vec{\omega} \cdot \vec{\partial}_1 s_j(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) = -k_a s_j(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) - k_s s_j(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) + k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' s_j(\vec{x}, \vec{\omega}', \ddot{\boldsymbol{\pi}}) \tag{4.23}$$

If we compare the RTE of I (Eq.4.20) and the RTE of s_j (Eq.4.23), we observe that the intensity I and the sensitivity of intensity s_j share the same model of transport in the medium. This is a new and interesting point of view because we can associate all the knowledge we have built concerning the solution of the model of intensity. We propose to build a similar image for the sensitivity s_j, which could be absorbed and scattered in the medium as the intensity I.

Boundary conditions for the box and the receiver

We differentiate then the boundary conditions of intensity (Eq.4.21) for $\mathscr{B}$ and $\mathscr{R}$:

$$\begin{cases} s_j(\vec{x}_r, \vec{\omega}_r) = 0 \\ s_j(\vec{x}_B, \vec{\omega}_B) = 0 \end{cases} \tag{4.24}$$

Since the boundary conditions of $\mathscr{B}$ and $\mathscr{R}$ do not depended on the geometric vector $\ddot{\boldsymbol{\pi}}$, the boundary conditions on $\mathscr{B}$ and $\mathscr{R}$ are null. Based on the results of derivation (Eq.4.24), we propose an image of the "physics of radiation" for sensitivity. We can consider these two boundary conditions as "cold black" bodies for sensitivity s_j. Physically, it means that the sensitivity arriving at the two boundaries is all "absorbed" and the two boundaries do not "emit" sensitivity. This similar image of sensitivity can help a lot in understanding and analyzing these sensitivities.

Boundary condition for each triangle of the object

The development of the boundary condition of each triangle is less straightforward and represents tough work. We apply the method developed in the Ph.D. work of Paule Lapeyre to build the boundary condition of sensitivity for each triangle of the object [32–34]. However, the formal development of the boundary condition is not the objective of this chapter. This chapter aims to illustrate how the strategy of vectorization is implemented. Therefore, we only present the result of the boundary condition of sensitivity for each triangle of the object. We make a choice to put one part of the formal developments to Appendix.C. It corresponds to the application of the method developed by Paule Lapeyre [32] in her Ph.D. thesis into this specific example. Herein, we focus on the physical interpretation of the boundary condition of sensitivity.

Eq.4.25 is the boundary condition of sensitivity for the i^{th} triangle and j^{th} parameter, resulted from applying method of Paule Lapeyre. Only the source of sensitivity, which is noted $\mathring{s}_{i,j}$, exists for the boundary condition of sensitivity:

$$s_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{4.25}$$

Sources of sensitivity are found on each triangle. We remind here the physical definition of the sensitivity: the local perturbation of the intensity due to the perturbation of the

corresponding parameter. Therefore, the source of sensitivity is the origin of the sensitivity of the system. In other words, it is the origin of the perturbation of intensity due to the perturbation of the corresponding parameter. In this example, the parameter $\ddot{\boldsymbol{\pi}}$ characterises the translation and the rotation of the object $\mathcal{T}$ (see Fig.4.13). With the perturbation of $\ddot{\boldsymbol{\pi}}$, the perturbation of the position and rotation of each triangle is created, which creates the perturbation of intensity on the triangles. Therefore, the sources of sensitivity are found on all the triangles, of which the position and rotation are characterized by $\ddot{\boldsymbol{\pi}}$.

However, two kinds of sources are distinguished. As it is discussed earlier in the introduction of this chapter in Section.4.1.1, the intensity on the borders of triangles is discontinuous, and its gradient induces a Dirac term. The source of sensitivity $\mathring{s}_{i,j}$ is then composed by a surface source $\dot{s}_{i,j}$ and a line source $\hat{s}_{i,j,k}$, where k is the index of the border of a given triangle:

$$\mathring{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \dot{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) + \sum_{k=1}^{3} \hat{s}_{i,j,k}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}) \tag{4.26}$$

We also propose here a physical radiative image for sensitivities. Based on Eq.4.26, the surface and each edge of the triangles "emit" sensitivities because the surface source and line sources appear by differentiating the boundary condition of intensity.

For the surface source:

$$\dot{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = -\alpha_{i,j}\mathcal{C}[I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})] \tag{4.27}$$

$\mathcal{C}$ is the collision operator, referring to Eq.4.20. $\alpha_{i,j}$ can be regarded as a geometric factor, of which the mathematical formulation is shown and developed in Appendix.C.

The surface sources represent the perturbation of intensity of all the points on the surface. This perturbation is due to the perturbation of $\ddot{\boldsymbol{\pi}}$. With the perturbation of $\ddot{\boldsymbol{\pi}}$, the triangle will be translated or rotated and interacts with the semi-transparent medium. Therefore, the absorption and scattering effect in the medium will be affected. This is why the surface sources of sensitivity on the triangles depend on the radiative properties of the medium (Eq.4.27).

For the line source on the k^{th} border of the triangle:

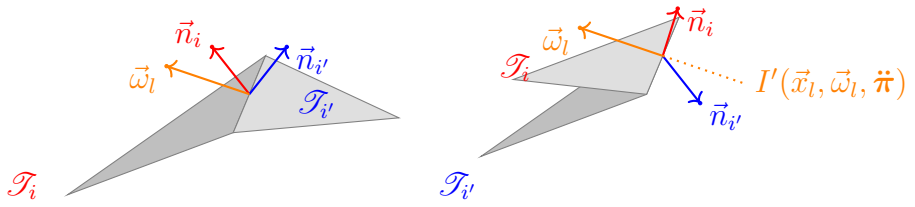
$$\hat{s}_{i,j,k}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}) = \hat{\beta}_{i,j,k}\big(I_0 - I'(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}})\big)\mathcal{H}(\vec{\omega}_l \cdot \vec{n}_{i'} < 0) \tag{4.28}$$

$\hat{\beta}_{i,j,k}$ is also a geometric factor, corresponding to the j^{th} parameter and the k^{th} segment of the i^{th} triangle. The detailed demonstrations can be found in Appendix.C.

A position on the edge of $\mathscr{T}_i$ is noted $\vec{x}_l$, $\forall \vec{x}_l \in \partial\mathscr{T}_i$. Respectively, the direction of emission on the edge is noted $\vec{\omega}_l$, $\forall \vec{\omega}_l \in \mathcal{S}$. $\vec{n}_{i'}$ is the normal of the triangle $\mathscr{T}_{i'}$ who shares the edge (where located the line source) with $\mathscr{T}_i$ and I' is the intensity coming backward, following the direction of $\vec{\omega}_l$ and passing through $\vec{x}_l$, seeing Fig.4.14. The Heaviside function in Eq.4.28 indicates that this line source depends on the scalar product of the normal of the neighboring triangle ($\vec{n}_{i'}$) and the direction of emission $\vec{\omega}_l$.

Physically, the line source of sensitivity comes from the discontinuous profile of the triangles, as it is discussed in the introduction of the chapter (Section.4.1.1). The perturbation of intensity is not smooth on the edge of triangles, and two cases exist. For the case in Fig.4.14a, the perturbation of $\ddot{\boldsymbol{\pi}}$ does not cause the perturbation of intensity on the segment sharing by the two triangles. It is because the two triangles have the same emitted intensity. The translation and rotation of the triangles have no impact on the intensity of the shared segment, following $\vec{\omega}_l$ in Fig.4.14a. However, in the other case (Fig.4.14b), the perturbation of $\ddot{\boldsymbol{\pi}}$ has

an impact on the perturbation of the emitted intensity on the segment. It is because with the translation and rotation of the triangles, the intensity on the shared segment, following $\vec{\omega}_l$ in Fig.4.14b might take the value of the coming intensity from the backward.



(a) $\vec{\omega}_l \cdot \vec{n}_{i'} > 0$ **(b)** $\vec{\omega}_l \cdot \vec{n}_{i'} < 0$

Figure 4.14 – The schema of the line source on edge shared by $\mathscr{T}_i$ and $\mathscr{T}_{i'}$. When $\vec{\omega}_l \cdot \vec{n}_{i'} > 0$, the line source is null in this case because the triangles have homogeneous emitted intensity. When $\vec{\omega}_l \cdot \vec{n}_{i'} < 0$, the line source is not null. I' is then the intensity coming backward, following $\vec{\omega}_l$, passing through $\vec{x}_l$.

To sum up, it is just important to note here that at each triangle, a surface source is found on the surface, and a line source is found on each edge. These two kinds of sources have different path spaces because of their different location. In the following sections, we have to deal with these different path spaces through the strategy of vectorization.

4.3.5 Estimation of the observable and its sensitivities

The model of I and of s_j have been built in this stage. Based on these models, we can now estimate the observable, which is the impacting power P on the receiver $\mathscr{R}$, as well as its vector of sensitivities $\boldsymbol{S}$ with respect to the geometric parameters $\ddot{\boldsymbol{\pi}}$.

Estimation of the observable

A general model for the intensity I (Eq.4.29) has been completely built at this stage.

$$\begin{cases} \vec{\omega} \cdot \partial_1 I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) = -k_a I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) - k_s I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) + k_s \displaystyle\int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}', \ddot{\boldsymbol{\pi}}) \\ I(\vec{x}_B, \vec{\omega}_B) = 0 \\ I(\vec{x}_r, \vec{\omega}_r) = 0 \\ I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \mathring{I}_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = I_0 \end{cases} \tag{4.29}$$

Based on this model, we aim to build algorithms to estimate the impacting power P by the Monte Carlo method.

Concerning the model of intensity I, the only source is the intensity from the emitter $\mathring{I}$ located on the surface of $\mathscr{T}$. P will be estimated by summing the source $\mathring{I}$ captured on the receiver $\mathscr{R}$ (see also Eq.4.17).

The corresponding statistical ray tracing process can start from the receiver looking for the sources or, inversely, from the sources located on $\mathscr{T}_i$, looking for the receiver. The strategy of proceeding statistical ray tracing process has enormous influences on the convergence rate.

Herein, unlike the previous example where the reverse ray-tracing process proceeded (the one-dimensional example in Chapter.3 and the example of rendering in this chapter), we choose to proceed with a forward ray-tracing process (start the statistical ray-tracing process by the sources, looking for the receiver $\mathscr{R}$). The interests of forward ray-tracing and reverse

ray-tracing process are already discussed in Chapter.1. We choose the forward ray-tracing here to have better convergence performance for the estimation of sensitivities. This choice will be discussed in the next subsection.

The impacting power on $\mathscr{R}$ is the sum of all sources of intensity emitted spatially and angularly, hitting the receiver after the scattering walks in the medium. The sources are only on the object $\mathscr{T}$ because $\mathscr{B}$, $\mathscr{R}$ and the medium are cold:

$$P(\ddot{\boldsymbol{\pi}}) = \int_{\mathscr{T}} \mathscr{P}_{X_p}(\vec{x}_p) d\vec{x}_p \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p \left\{ \pi \mathcal{S}_{\mathscr{T}} \mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \right\} \tag{4.30}$$

with the pdfs:

$$\mathscr{P}_{X_p}(\vec{x}_p) = \frac{1}{\mathcal{S}_{\mathscr{T}}}; \mathscr{P}_{\Omega_p}(\vec{\omega}_p) = \frac{|\vec{\omega}_p \cdot \vec{n}_i|}{\pi} \tag{4.31}$$

where $\mathcal{S}_{\mathscr{T}}$ is the area of the surface of $\mathscr{T}$ and $\mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ is the intensity emitted on the point $\vec{x}_p$ ($\vec{x}_p$ is on the surface of the triangles), following the direction $\vec{\omega}_p$, arriving at the receiver $\mathscr{R}$, after the scattering events in the medium and attenuated by the absorption event all along the ray path. Based on the model (Eq.4.29), the same process as it is shown in Chapter.3 will be applied in this example. We write the Fredholm equation, and we reformulate it into an expected value form. Finally, we get an iterative integral formulation for $\mathring{I}_{i,hit}$:

$$\begin{aligned} \mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \int_0^\infty k_s e^{-k_s \sigma_0} d\sigma_0 \Big\{ & \mathcal{H}(\sigma_0 - d_0) \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{R}) I_0 e^{-k_a d_0} + \\ \mathcal{H}(d_0 - \sigma_0) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^\infty k_s e^{-k_s \sigma_1} d\sigma_1 \Big\{ & \mathcal{H}(\sigma_1 - d_1) \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{R}) I_0 e^{-k_a(\sigma_0 + d_1)} + \\ & \mathcal{H}(d_1 - \sigma_1) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \dots \Big\} \Big\} \end{aligned} \tag{4.32}$$

where σ_0 and σ_1 represent respectively the path length of the photon for its 1^{st} and 2^{nd} scattering; $\vec{x}_{0,hit}$, $\vec{x}_{1,hit}$ are the points of intersection after 0 and 1 scattering respectively; $\vec{\omega}_1$ and $\vec{\omega}_2$ represent respectively the direction of propagation of the photon after its 1^{st} and 2^{nd} scattering; d_0 represents the distance that the photon needs to go through to hit a boundary without scattering; d_1 represents the distance that, starting from the position where located the 1^{st} scattering, the photon need to go through to hit a boundary without 2^{nd} scattering.

The corresponding Algorithm is Algo.4, referring to Eq.4.30. In Algo.4, each realization of $\mathring{I}_{i,hit}$ for sampled $\vec{x}_p$ and $\vec{\omega}_p$ is done by Algo.5, referring to Eq.4.32.

Algorithm 4 Evaluation of $P(\ddot{\boldsymbol{\pi}})$

Input: $\mathscr{B}$, $\mathscr{T}$, $\mathscr{R}$, n_{MC}, k_a, k_s

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{B}$: the geometry of the box.
▷ $\mathscr{T}$: the triangulated geometry of an object.
▷ $\mathscr{R}$: the geometry of the receiver.
▷ k_a: the coefficient of absorption.
▷ k_s: the coefficient of scattering.

1: $W_I \leftarrow 0$ and $V_I \leftarrow 0$.

▷ Initialize the sum and the sum of square

2: **for** $i_{MC} = 0 \rightarrow n_{MC}$ **do**
3: Initialize the weight of Monte-Carlo $w_I \leftarrow 0$
4: Compute the area of the surface of $\mathscr{T}$: $\mathcal{S}_{\mathscr{T}}$
5: Sample uniformly a position $\vec{x}_p$ on $\mathscr{T}$
6: Sample a direction $\vec{\omega}_p$ based on $\mathcal{P}_{\Omega_p}(\vec{\omega}_p)$
7: Get $\mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ by Algo.5
8: $w_I \leftarrow \mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \mathcal{S}_{\mathscr{T}} \pi$
9: $W_I \leftarrow W_I + w_I$
10: $V_I \leftarrow V_I + w_I^2$
11: $i_{MC} \leftarrow i_{MC} + 1$
12: **end for**
Output: $P \leftarrow \frac{W_I}{n_{MC}}$, $\sigma[P] \leftarrow \sqrt{\frac{\frac{V_I}{n_{MC}} - (\frac{W_I}{n_{MC}})^2}{n_{MC}-1}}$

Algorithm 5 Get $\mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ with the sampled $\vec{x}_p$ and $\vec{\omega}_p$

Input: $\mathscr{B}$, $\mathscr{T}$, $\mathscr{R}$, $\vec{x}_p$, $\vec{\omega}_p$, k_a, k_s
▷ $\mathscr{T}$: the triangulated geometry of an object.
▷ $\mathscr{R}$: the geometry of the receiver.
▷ $\mathscr{B}$: the geometry of the box.
▷ k_a: the coefficient of absorption.
▷ k_s: the coefficient of scattering.

Initialize the keeprunning flag $keeprunning \leftarrow 1$
Initialize the total length of optical path $\mathcal{L}_{tot} \leftarrow 0$
Initialize the ray-tracing process: $\vec{x} \leftarrow \vec{x}_p$, $\vec{\omega} \leftarrow \vec{\omega}_p$
while keeprunning **do**
Get the first intersection point of the ray $(\vec{x}, \vec{\omega})$: $\vec{x}_{hit}$
Get the distance between $\vec{x}_{hit}$ and $\vec{x}$: $d \leftarrow ||\vec{x}_{hit} - \vec{x}||$
Sample a length based on $\mathcal{P}_{\Sigma}(\sigma) = k_s e^{-k_s \sigma}$
if $\sigma < d$ **then**
Accumulate the length of optical path $\mathcal{L}_{tot} \leftarrow \mathcal{L}_{tot} + \sigma$
Sample a direction $\vec{\omega}'$ based on $\mathcal{P}_{\Omega'}(\vec{\omega}') = \frac{1}{4\pi}$
Update the position $\vec{x} \leftarrow \vec{x} + \vec{\omega}\sigma$
Update the direction of propagation $\vec{\omega} \leftarrow \vec{\omega}'$
else if $\vec{x}_{hit} \in \mathscr{R}$ **then**
$\mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \leftarrow I_0 e^{-k_a \mathcal{L}_{tot}} \mathcal{S}_{\mathscr{T}} \pi$
$keeprunning \leftarrow 0$
else
$\mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \leftarrow 0$
$keeprunning \leftarrow 0$
end if
end while
Output: $I_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$

Estimation of sensitivities

The general model for the sensitivities s_j (Eq.4.33) has also been built in this stage.

$$\begin{cases} \vec{\omega}\cdot\vec{\partial}_1 s_j(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) = -k_a s_j(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) - k_s s_j(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) + k_s\displaystyle\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}' s_j(\vec{x},\vec{\omega}',\ddot{\boldsymbol{\pi}}) \\ s(\vec{x}_B,\vec{\omega}_B) = 0 \\ s(\vec{x}_r,\vec{\omega}_r) = 0 \\ \dot{s}_{i,j}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = -\alpha_{i,j}\left(-k_a I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) - k_s I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) + k_s\displaystyle\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}' I(\vec{x},\vec{\omega}',\ddot{\boldsymbol{\pi}})\right) \\ \hat{s}_{i,j,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) = \hat{\beta}_{i,j,k}\big(I_0 - I'(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}})\big)\mathcal{H}(\vec{\omega}_l\cdot\vec{n}_{i'} < 0) \end{cases} \tag{4.33}$$

Concerning the model of s_j, there are sources of sensitivities $\dot{s}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})$ on the surface of the triangle $\mathscr{T}_i$ and sources of sensitivities $\hat{s}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}})$ on the edges of the triangle $\mathscr{T}_i$. S_j will be estimated by summing all the sources $\dot{s}_{i,j}$ and $\hat{s}_{i,j,k}$ for all the "object" $\mathscr{T}$, captured on the receiver $\mathscr{R}$(see Eq.4.18 and 4.19).

Therefore, the sensitivity of impacting power with respect to the parameter $\ddot{\pi}_j$ is the sum of all sources of sensitivities emitted spatially and angularly, on the surfaces $\mathscr{T}_i$ and on the edges $\partial\mathscr{T}_i$, hitting the receiver after the scattering walks in the medium:

$$\begin{aligned} S_j(\ddot{\pi}_j) = &\sum_{i=1}^{n_{\mathscr{T}}}\int_{\mathscr{T}_i}\mathscr{P}_{X_{p,i}}(\vec{x}_p)d\vec{x}_p\int_{2\pi(\vec{n}_i)}\mathscr{P}_{\Omega_p}(\vec{\omega}_p)d\vec{\omega}_p\left\{\dot{s}_{i,j,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})\mathcal{S}_{\mathscr{T}_i}\pi\right\} \\ &+\sum_{i=1}^{n_{\mathscr{T}}}\sum_{k=1}^{3}\int_{\partial\mathscr{T}_{i,k}}\mathscr{P}_{X_{l,i,k}}(\vec{x}_l)d\vec{x}_l\int_{2\pi(\vec{n}_i)}\mathscr{P}_{\Omega_l}(\vec{\omega}_l)d\vec{\omega}_l\left\{\hat{s}_{i,j,k,hit}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}})\frac{\mathcal{L}_{i,k}\pi}{|\vec{\omega}_p\cdot\vec{n}_i|}\right\} \end{aligned} \tag{4.34}$$

with the pdfs:

$$\mathscr{P}_{X_{p,i}}(\vec{x}_p) = \frac{1}{\mathcal{S}_i};\mathscr{P}_{X_{l,i,k}}(\vec{x}_l) = \frac{1}{\mathcal{L}_{i,k}};\mathscr{P}_{\Omega_l}(\vec{\omega}_l) = \frac{|\vec{\omega}_l\cdot\vec{n}_i|}{\pi} \tag{4.35}$$

where $\mathcal{S}_i$ is the area of the surface of $\mathscr{T}_i$, $\mathcal{L}_{i,k}$ the length of the k^{th} edge of the triangle $\mathscr{T}_i$. $\dot{s}_{i,j,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})$ and $\hat{s}_{i,j,k,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})$ are the sensitivity emitted on the point $\vec{x}_p$ and point $\vec{x}_l$, following the direction $\vec{\omega}_p$ and $\vec{\omega}_l$ respectively, arriving at the receiver $\mathscr{R}$, after the scattering events in the medium and attenuated by the absorption event all along the ray path. As usual, we write the Fredholm equation, and we reformulate it into an expected value form. Finally, we get an iterative integral formulation for $\dot{s}_{i,j}$ and $\hat{s}_{i,j,k}$:

$$\begin{aligned} \dot{s}_{i,j,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \int_0^\infty k_s e^{-k_s\sigma_0}d\sigma_0\Bigg\{&\mathcal{H}(\sigma_0-d_0)\mathcal{H}(\vec{x}_{0,hit}\in\mathscr{R}) \\ &\dot{s}_{i,j}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})e^{-k_a d_0} + \mathcal{H}(d_0-\sigma_0)\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_1\int_0^\infty k_s e^{-k_s\sigma_1}d\sigma_1 \\ &\Bigg\{\mathcal{H}(\sigma_1-d_1)\mathcal{H}(\vec{x}_{1,hit}\in\mathscr{R})\dot{s}_{i,j}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})e^{-ka(\sigma_0+d_1)} + \\ &\mathcal{H}(d_1-\sigma_1)\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_2\ldots\Bigg\}\Bigg\} \end{aligned} \tag{4.36}$$

$$\hat{s}_{i,j,k,hit}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) = \int_0^\infty k_s e^{-k_s\sigma_0} d\sigma_0 \Bigg\{ \mathcal{H}(\sigma_0 - d_0)\mathcal{H}(\vec{x}_{0,hit} \in \mathscr{R}) \\ \hat{s}_{i,j,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) e^{-k_a d_0} + \mathcal{H}(d_0 - \sigma_0) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^\infty k_s e^{-k_s\sigma_1} d\sigma_1 \\ \Big\{ \mathcal{H}(\sigma_1 - d_1)\mathcal{H}(\vec{x}_{1,hit} \in \mathscr{R}) \hat{s}_{i,j,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) e^{-ka(\sigma_0+d_1)} + \\ \mathcal{H}(d_1 - \sigma_1) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \dots \Big\} \Bigg\} \quad (4.37)$$

In the first term of Eq.4.34, the surface sources are counted triangle by triangle, which requires performing a Monte-Carlo estimation for each triangle. In the second term of Eq.4.34, the line sources are counted edge by edge for each triangle, which is, performing three Monte-Carlo estimations for each triangle (for its three edges).

Moreover, Eq.4.34 is dedicated only to estimate S_j. The same process is repeated to estimate S_1, S_2, S_3 and S_4 separately. The corresponding coding development is heavy, and the calculating time is consuming. In order to optimize the algorithm, we will use again the technique of vectorization, aiming to estimate the $\boldsymbol{S} \equiv [S_1, S_2, S_3, S_4]^T$ in vectorized form. The reason that we can perform this vectorization is that, $\forall j$, the corresponding line sources share the same path space, and the surface sources share the other same path space.

4.3.6 The vectorized Monte-Carlo method

In this subsection, we vectorize firstly the estimation of $\boldsymbol{S}$, by vectorizing the transport of line sources and surface sources separately. Secondly, efforts are made to vectorize also the estimation of the observable P and the estimation of $\boldsymbol{S}$, by vectorizing the transport of the surface sources of intensity and the surface sources of sensitivity.

Vectorization of the sensitivities

We remain that $\boldsymbol{S}(\ddot{\boldsymbol{\pi}}) = \vec{\nabla} P(\ddot{\boldsymbol{\pi}}) = [\frac{\partial P}{\partial \ddot{\pi}_1} \frac{\partial P}{\partial \ddot{\pi}_2} \frac{\partial P}{\partial \ddot{\pi}_3} \frac{\partial P}{\partial \ddot{\pi}_4}]^T$. We rewrite the Eq.4.34 into a vectorized form:

$$\boldsymbol{S}(\ddot{\boldsymbol{\pi}}) = \sum_{i=1}^{n_{\mathscr{T}}} \int_{\mathscr{T}_i} \mathscr{P}_{X_{p,i}}(\vec{x}_p) d\vec{x}_p \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p \left\{ \dot{\boldsymbol{s}}_{i,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \mathcal{S}_i \pi \right\} \\ + \sum_{i=1}^{n_{\mathscr{T}}} \sum_{k=1}^{3} \int_{\partial\mathscr{T}_{i,k}} \mathscr{P}_{X_{l,i,k}}(\vec{x}_l) d\vec{x}_l \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_l}(\vec{\omega}_l) d\vec{\omega}_l \left\{ \hat{\boldsymbol{s}}_{i,k,hit}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) \frac{\mathcal{L}_{i,k}\pi}{|\vec{\omega}_p \cdot \vec{n}_i|} \right\} \quad (4.38)$$

where

$$\dot{\boldsymbol{s}}_{i,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \begin{bmatrix} \dot{s}_{i,1,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \dot{s}_{i,2,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \dot{s}_{i,3,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \dot{s}_{i,4,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \end{bmatrix} \quad (4.39)$$

and

$$\hat{\boldsymbol{s}}_{i,k,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \begin{bmatrix} \hat{s}_{i,1,k,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \hat{s}_{i,2,k,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \hat{s}_{i,3,k,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \hat{s}_{i,4,k,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \end{bmatrix} \tag{4.40}$$

In fact, the only effort that is made is to write Eq.4.34 for the case where $j = 1,2,3,4$. Physically, this vectorized form (Eq.4.38) regroups the space path of different sources. The surface sources $\dot{s}_{i,j,hit}$, $\forall j \in \{1,2,3,4\}$ share the same path spaces. Therefore, they are rewritten in a vectorized form in Eq.4.39 and the same pdf in Eq.4.38 is used to transport the vector of surface sources $\dot{\boldsymbol{s}}_{i,hit}$. A similar treatment is done for line sources.

The propagations of the sensitivities $\dot{\boldsymbol{s}}_{i,hit}$ and $\hat{\boldsymbol{s}}_{i,k,hit}$ can be referred to Eq.4.36 and Eq.4.37:

$$\begin{aligned} \dot{\boldsymbol{s}}_{i,hit}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \int_0^\infty k_s e^{-k_s\sigma_0} d\sigma_0 \Big\{ & \mathcal{H}(\sigma_0 - d_0)\mathcal{H}(\vec{x}_{0,hit} \in \mathscr{R}) \\ & \dot{\boldsymbol{s}}_i(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) e^{-k_a d_0} + \mathcal{H}(d_0 - \sigma_0) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^\infty k_s e^{-k_s\sigma_1} d\sigma_1 \\ & \Big\{ \mathcal{H}(\sigma_1 - d_1)\mathcal{H}(\vec{x}_{1,hit} \in \mathscr{R}) \dot{\boldsymbol{s}}_i(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) e^{-ka(\sigma_0+d_1)} + \\ & \mathcal{H}(d_1 - \sigma_1) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \ldots \Big\} \Big\} \end{aligned} \tag{4.41}$$

$$\begin{aligned} \hat{\boldsymbol{s}}_{i,k,hit}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) = \int_0^\infty k_s e^{-k_s\sigma_0} d\sigma_0 \Big\{ & \mathcal{H}(\sigma_0 - d_0)\mathcal{H}(\vec{x}_{0,hit} \in \mathscr{R}) \\ & \hat{\boldsymbol{s}}_{i,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) \cdot \vec{t}_i e^{-k_a d_0} + \mathcal{H}(d_0 - \sigma_0) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^\infty k_s e^{-k_s\sigma_1} d\sigma_1 \\ & \Big\{ \mathcal{H}(\sigma_1 - d_1)\mathcal{H}(\vec{x}_{1,hit} \in \mathscr{R}) \hat{\boldsymbol{s}}_{i,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) e^{-ka(\sigma_0+d_1)} + \\ & \mathcal{H}(d_1 - \sigma_1) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \ldots \Big\} \Big\} \end{aligned} \tag{4.42}$$

where $\dot{\boldsymbol{s}}_i(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \begin{bmatrix} \dot{s}_{i,1}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \dot{s}_{i,2}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \dot{s}_{i,3}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \\ \dot{s}_{i,4}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \end{bmatrix}$ and $\hat{\boldsymbol{s}}_i(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) = \begin{bmatrix} \hat{s}_{i,1,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) \\ \hat{s}_{i,2,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) \\ \hat{s}_{i,3,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) \\ \hat{s}_{i,4,k}(\vec{x}_l,\vec{\omega}_l,\ddot{\boldsymbol{\pi}}) \end{bmatrix}$ are the boundary conditions, seeing Eq.4.33.

Adding the observable to the vectorized form

Secondly, we aim to vectorize also the estimation of the observable (P). When we observe Eq.4.30, which estimates the observable, and Eq.4.38, which estimates the sensitivities, we find that they do not have the same pdfs, because the integral domains are not the same. Herein, we make efforts to rewrite the Eq.4.30, in order to homogenize the pdfs in Eq.4.30 and in Eq.4.38.

In Eq.4.30, the sources of intensity are accumulated by the integral of all the surfaces of $\mathscr{T}$. We can rewrite Eq.4.30, so that the sources $\mathring{I}_{i,hit}$ arrived to the receiver $\mathscr{R}$ are accumulated triangle by triangle:

$$P(\ddot{\boldsymbol{\pi}}) = \sum_{i=1}^{n_{\mathscr{T}}} \int_{\mathscr{T}} \mathscr{P}_{X_{p,i}}(\vec{x}_p) d\vec{x}_p \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p \left\{ \mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \mathcal{S}_i \pi \right\} \tag{4.43}$$

By this simple reformulation, the pdfs of estimating P become the same as the pdfs of accumulating surface sources of $\boldsymbol{S}$ (the first term in Eq.4.38). Therefore, we can write Eq.4.43 and Eq.4.38 in a vectorized form:

$$\begin{aligned} \begin{bmatrix} P(\ddot{\boldsymbol{\pi}}) \\ \boldsymbol{S}(\ddot{\boldsymbol{\pi}}) \end{bmatrix} = & \sum_{i=1}^{n_{\mathscr{T}}} \int_{\mathscr{T}_i} \mathscr{P}_{X_{p,i}}(\vec{x}_p) d\vec{x}_p \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p \left\{ \begin{bmatrix} \mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \\ \dot{\boldsymbol{s}}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \end{bmatrix} \mathcal{S}_i \pi \right\} \\ & + \sum_{i=1}^{n_{\mathscr{T}}} \int_{\partial \mathscr{T}_i} \mathscr{P}_{X_{l,i,k}}(\vec{x}_l) d\vec{x}_l \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_l}(\vec{\omega}_l) d\vec{\omega}_p \left\{ \begin{bmatrix} 0 \\ \hat{\boldsymbol{s}}_{i,k}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}) \end{bmatrix} \frac{\mathcal{L}_{i,k} \pi}{|\vec{\omega}_p \cdot \vec{n}_i|} \right\} \end{aligned} \tag{4.44}$$

From Eq.4.32 and Eq.4.36, we can also observe that $\mathring{I}_{i,hit}$ and $\dot{\boldsymbol{s}}_{i,hit}$ have the same path space in the medium (they share the same domains of integral in the medium). Therefore, the propagation of $\mathring{I}_{i,hit}$ and $\dot{\boldsymbol{s}}_{i,hit}$ in the medium can be also vectorized:

$$\begin{aligned} \begin{bmatrix} \mathring{I}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \\ \dot{\boldsymbol{s}}_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \end{bmatrix} = & \int_0^\infty k_s e^{-k_s \sigma_0} d\sigma_0 \Bigg\{ \mathcal{H}(\sigma_0 - d_0) \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{R}) \\ & \begin{bmatrix} \mathring{I}_0 \\ \dot{\boldsymbol{s}}_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \end{bmatrix} e^{-k_a d_0} + \mathcal{H}(d_0 - \sigma_0) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^\infty k_s e^{-k_s \sigma_1} d\sigma_1 \\ & \Bigg\{ \mathcal{H}(\sigma_1 - d_1) \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{R}) \begin{bmatrix} \mathring{I}_0 \\ \dot{\boldsymbol{s}}_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \end{bmatrix} e^{-ka(\sigma_0 + d_1)} + \\ & \mathcal{H}(d_1 - \sigma_1) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \dots \Bigg\} \Bigg\} \end{aligned} \tag{4.45}$$

To summarize, in order to estimate $P(\ddot{\boldsymbol{\pi}})$ and $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$, we need Algo.6 to sum the surface sources and Algo.7 to sum the line sources. The vector $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$ is then separated by a surface part $\dot{\boldsymbol{S}}$ and a line part $\hat{\boldsymbol{S}}$, which are to be estimated by Algo.6 and Algo.7 respectively:

$$\boldsymbol{S}(\ddot{\boldsymbol{\pi}}) = \dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) + \hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) \tag{4.46}$$

At this stage, the estimation of P and $\boldsymbol{S}$ are vectorized. Instead of performing Monte-Carlo estimations for P, S_1, S_2, S_3 and S_4 separately, we vectorize the transport of sources sharing the same path space and estimate P and $\boldsymbol{S}$ by performing Algo.6 and Algo.7. It reduces the complexity of coding development and largely reduces the required computing time.

Algorithm 6 Evaluation of $P(\ddot{\boldsymbol{\pi}})$ and $\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})$

Input: $\mathscr{B}$, $\mathscr{T}$, $\mathscr{R}$, n_{MC}, k_a, k_s

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{B}$: the geometry of the box.
▷ $\mathscr{T}$: the triangulated geometry of an object.

▷ $\mathscr{R}$: the geometry of the receiver.
▷ k_a: the coefficient of absorption.
▷ k_s: the coefficient of scattering.

$P(\ddot{\boldsymbol{\pi}}) \leftarrow 0$, $\sigma[P(\ddot{\boldsymbol{\pi}})] \leftarrow \mathbf{0}$ and $\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) \leftarrow 0$, $\sigma[\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})] \leftarrow \mathbf{0}$
Get the number of triangles composing $\mathscr{T}$: $n_{\mathscr{T}}$
for $i_{\mathscr{T}} = 1 \rightarrow n_{\mathscr{T}}$ **do**
    Get the index $i \leftarrow i_{\mathscr{T}}$
    $W_I \leftarrow 0$, $\boldsymbol{W}_s \leftarrow \mathbf{0}$ and $V_I \leftarrow 0$, $\boldsymbol{V}_s \leftarrow \mathbf{0}$
        ▷ Initialize the sum and the sum of square
    for $i_{MC} = 0 \rightarrow n_{MC}$ **do**
        Initialize the weight of Monte-Carlo $w_I \leftarrow 0$ and $\boldsymbol{w}_s \leftarrow \mathbf{0}$
        Initialize the keeprunning flag $keeprunning \leftarrow 1$
        Initialize the total optical path length $\mathscr{L}_{tot} \leftarrow 0$
        Sample uniformly a position $\vec{x}_p$ on $\mathscr{T}_i$
        Sample a direction $\vec{\omega}_p$ based on $\mathscr{P}_{\Omega_p}(\vec{\omega}_p)$
        Initialize the ray-tracing process: $\vec{x} \leftarrow \vec{x}_p$, $\vec{\omega} \leftarrow \vec{\omega}_p$
        while keeprunning **do**
            get the first intersection point of the ray $(\vec{x}, \vec{\omega})$: $\vec{x}_{hit}$
            Get the distance between $\vec{x}_{hit}$ and $\vec{x}$: $d \leftarrow ||\vec{x}_{hit} - \vec{x}||$
            Sample a length based on $\mathscr{P}_{\Sigma}(\sigma) = k_s e^{-k_s \sigma}$
            if $\sigma < d$ **then**
                Accumulate the length of optical path $\mathscr{L}_{tot} \leftarrow \mathscr{L}_{tot} + \sigma$
                Sample a direction $\vec{\omega}'$ based on $\mathscr{P}_{\Omega'}(\vec{\omega}') = \frac{1}{4\pi}$
                Update the position $\vec{x} \leftarrow \vec{x} + \vec{\omega}\sigma$
                Update the direction of propagation $\vec{\omega} \leftarrow \vec{\omega}'$
            else if $\vec{x}_{hit} \in \mathscr{R}$ **then**
                $w_I \leftarrow I_0 e^{-k_a \mathscr{L}_{tot}} \mathcal{S}_{\mathscr{T}} \pi$
                Sample a direction $\vec{\omega}'$
                if $\vec{\omega}' \cdot \vec{n}_i > 0$ **then**
                    $\vec{\omega}_p \leftarrow \vec{\omega}'$
                    Get $I_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ by Algo.5
                    Calculate $\vec{\chi}_j, \forall j \in \{1,2,3,4\}$, following Tabel.C.1
                    Calculate $\boldsymbol{\alpha}_i \equiv [\alpha_{i,j}], \forall j \in \{1,2,3,4\}$, following Eq.C.16
                    $\boldsymbol{w}_s \leftarrow -\boldsymbol{\alpha}_i \left(-k_a I_0 - k_s I_0 + k_s I_{i,hit}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})\right) e^{-k_a \mathscr{L}_{tot}} \mathcal{S}_{\mathscr{T}} \pi$
                else
                    $\boldsymbol{w}_s \leftarrow -\boldsymbol{\alpha}_i \left(-k_a I_0 - k_s I_0\right) e^{-ka \mathscr{L}_{tot}} \mathcal{S}_{\mathscr{T}} \pi$
                end if
                $keeprunning \leftarrow 0$
            else
                $w_I \leftarrow 0$
                $keeprunning \leftarrow 0$
            end if
        end while
        $W_I \leftarrow W_I + w_I$
        $V_I \leftarrow V_I + w_I^2$
        $\boldsymbol{W}_s \leftarrow \boldsymbol{W}_s + \boldsymbol{w}_s$
        $\boldsymbol{V}_s \leftarrow \boldsymbol{V}_s + \boldsymbol{w}_s^2$
        $i_{MC} \leftarrow i_{MC} + 1$
    end for
    $P(\ddot{\boldsymbol{\pi}}) \leftarrow P(\ddot{\boldsymbol{\pi}}) + \frac{W_I}{n_{MC}}$
    $\sigma[P(\ddot{\boldsymbol{\pi}})] \leftarrow \sigma[P(\ddot{\boldsymbol{\pi}})] + \sqrt{\frac{\frac{V_I}{n_{MC}} - (\frac{W_I}{n_{MC}})^2}{n_{MC}-1}}$
    $\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) \leftarrow \dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) + \frac{\boldsymbol{W}_s}{n_{MC}}$

$\sigma[\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})] \leftarrow \sigma[\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})] + \sqrt{\frac{\frac{\boldsymbol{V}_s}{n_{MC}} - (\frac{\boldsymbol{W}_s}{n_{MC}})^2}{n_{MC}-1}}$
$i_{\mathscr{T}} \leftarrow i_{\mathscr{T}} + 1$
end for
Output: $P(\ddot{\boldsymbol{\pi}}), \sigma[P(\ddot{\boldsymbol{\pi}})], \dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}), \sigma[\dot{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})]$

Algorithm 7 Evaluation of $\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})$

Input: $\mathscr{B}$, $\mathscr{T}$, $\mathscr{R}$, n_{MC}, k_a, k_s

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{B}$: the geometry of the box.
▷ $\mathscr{T}$: the triangulated geometry of an object.
▷ $\mathscr{R}$: the geometry of the receiver.
▷ k_a: the coefficient of absorption.
▷ k_s: the coefficient of scattering.

$\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) \leftarrow 0$, $\sigma[\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})] \leftarrow \boldsymbol{0}$
Get the number of triangles composing $\mathscr{T}$: $n_{\mathscr{T}}$
for $i_{\mathscr{T}} = 1 \rightarrow n_{\mathscr{T}}$ **do**
  Get the index $i \leftarrow i_{\mathscr{T}}$
  for $k = 1 \leftarrow 3$ **do**
    $\boldsymbol{W}_s \leftarrow \boldsymbol{0}$, $\boldsymbol{V}_s \leftarrow \boldsymbol{0}$
    ▷ Initialize the sum and the sum of square
    for $i_{MC} = 0 \rightarrow n_{MC}$ **do**
      Initialize the weight of Monte-Carlo $\boldsymbol{w}_s \leftarrow \boldsymbol{0}$
      Initialize the keeprunning flag $keeprunning \leftarrow 1$
      Initialize the total optical path length $\mathscr{L}_{tot} \leftarrow 0$
      Sample uniformly a position $\vec{x}_l$ on $\partial\mathscr{T}_{i,k}$
      Sample a direction $\vec{\omega}_l$ based on $\mathscr{P}_{\Omega_l}(\vec{\omega}_l)$
      Initialize the ray-tracing process: $\vec{x} \leftarrow \vec{x}_l$, $\vec{\omega} \leftarrow \vec{\omega}_l$
      while keeprunning **do**
        Get the first intersection point of the ray $(\vec{x}, \vec{\omega})$: $\vec{x}_{hit}$
        Get the distance between $\vec{x}_{hit}$ and $\vec{x}$: $d \leftarrow ||\vec{x}_{hit} - \vec{x}||$
        Sample a length based on $\mathscr{P}_{\Sigma}(\sigma) = k_s e^{-k_s\sigma}$
        if $\sigma < d$ **then**
          Accumulate the length of optical path $\mathscr{L}_{tot} \leftarrow \mathscr{L}_{tot} + \sigma$
          Sample a direction $\vec{\omega}'$ based on $\mathscr{P}_{\Omega'}(\vec{\omega}') = \frac{1}{4\pi}$
          Update the position $\vec{x} \leftarrow \vec{x} + \vec{\omega}\sigma$
          Update the direction of propagation $\vec{\omega} \leftarrow \vec{\omega}'$
        else if $\vec{x}_{hit} \in \mathscr{R}$ **then**
          Calculate $\vec{\chi}_j, \forall j \in \{1,2,3,4\}$, following Tabel.C.1
          Get $\vec{t}_k$ of the edge
          if $\vec{\omega}_l \cdot \vec{n}_{i'} > 0$ **then**
            $I_{i,hit}(\vec{x}, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}) = I_0$
          else
            Get $I_{i,hit}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}})$ by Algo.5
          end if
          $\boldsymbol{w}_s \leftarrow \begin{bmatrix} (\vec{\omega}_l \times \vec{\chi}_1) \cdot \vec{t}_k \\ (\vec{\omega}_l \times \vec{\chi}_2) \cdot \vec{t}_k \\ (\vec{\omega}_l \times \vec{\chi}_3) \cdot \vec{t}_k \\ (\vec{\omega}_l \times \vec{\chi}_4) \cdot \vec{t}_k \end{bmatrix} \big(I_0 - I_{i,hit}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}})\big) e^{-k_a \mathscr{L}_{tot}} \mathcal{S}_{\mathscr{T}} \pi$
          $keeprunning \leftarrow 0$
        else
          $w_I \leftarrow 0$
          $keeprunning \leftarrow 0$

end if
end while
$\boldsymbol{W}_s \leftarrow \boldsymbol{W}_s + \boldsymbol{w}_s$
$\boldsymbol{V}_s \leftarrow \boldsymbol{V}_s + \boldsymbol{w}_s^2$
$i_{MC} \leftarrow i_{MC} + 1$
end for
$k \leftarrow k + 1$
end for
$\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) \leftarrow \hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) + \frac{\boldsymbol{W}_s}{n_{MC}}$
$\sigma[\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})] \leftarrow \sigma[\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})] + \sqrt{\frac{\frac{\boldsymbol{V}_s}{n_{MC}} - (\frac{\boldsymbol{W}_s}{n_{MC}})^2}{n_{MC}-1}}$
$i_{\mathscr{T}} \leftarrow i_{\mathscr{T}} + 1$
end for
Output: $\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}), \sigma[\hat{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})]$

4.3.7 Validation of the estimation

This section of results is dedicated to validation purposes. Reminding that our method can be used for all kinds of triangulated geometry, herein, we take a geometry of a dodecahedron as the "object" $\mathscr{T}$ in this example. It consists of 20 points and 36 triangles. The geometric configuration is shown in Fig.4.15. The impacting power P on the receiver $\mathscr{R}$ of this configuration will be estimated by Algo.6. The corresponding vector of sensitivity $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$ will be estimated by Algo.6 and Algo.7 for different coefficients of absorption and scattering. Also, the vector $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$ will be approximated by the finite difference method and the results of approximation will be noted $\tilde{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}}) \equiv [\tilde{S}_j], \forall j \in \{1, 2, 3, 4\}$.

As it is introduced in Chapter.1, finite difference method is a common method to approximate the sensitivities. When P is derivable around $\ddot{\pi}_j$:

$$S_j = \lim_{\epsilon \to 0} \frac{P(\ddot{\pi}_j + \epsilon) - P(\ddot{\pi}_j - \epsilon)}{2\epsilon} \tag{4.47}$$

The finite difference method approximates S_j by giving a non-zero fixed value to ϵ. We note the result of approximation as $\tilde{S}_j$:

$$S_j \approx \frac{P(\ddot{\pi}_j + \Delta\ddot{\pi}_j) - P(\ddot{\pi}_j - \Delta\ddot{\pi}_j)}{2\Delta\ddot{\pi}_j} \equiv \tilde{S}_j \tag{4.48}$$

Practically, the approximation of $\tilde{S}_j$ requires estimating P twice. Since the impacting power P is estimated by the Monte-Carlo method, its statistical standard deviation $\sigma(P)$ is also estimated at the same time [14]. When approximating $\tilde{S}_j$, its standard deviation $\sigma(\tilde{S}_j)$ can be obtained [21]:

$$\sigma(\tilde{S}_j) \approx \frac{\sigma(P(\ddot{\pi}_j + \Delta\ddot{\pi}_j)) + \sigma(P(\ddot{\pi}_j - \Delta\ddot{\pi}_j))}{2\Delta\ddot{\pi}_j} \tag{4.49}$$

The major drawback of the finite difference method is that the related standard deviation $\sigma(\tilde{S}_j)$ is hard to control[21] and tends to infinity as $\Delta\ddot{\pi}_j$ tends to zero. Furthermore, it is important to be noted that if P is not smooth between $\ddot{\pi}_j - \Delta\ddot{\pi}_j$ and $\ddot{\pi}_j + \Delta\ddot{\pi}_j$, the approximation is not reliable.

However, we will still use the results approximated by Finite Difference method $\tilde{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})$ to validate our estimations $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$.

The impacting power P and the vector $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$ are estimated by Algo.6 and Algo.7. Also, $\tilde{\boldsymbol{S}}(\ddot{\boldsymbol{\pi}})$ is approximated by the Finite Difference method, where the impacting power P in Eq.4.47 is estimated by Algo.4. The corresponding results are shown in Table.4.1.

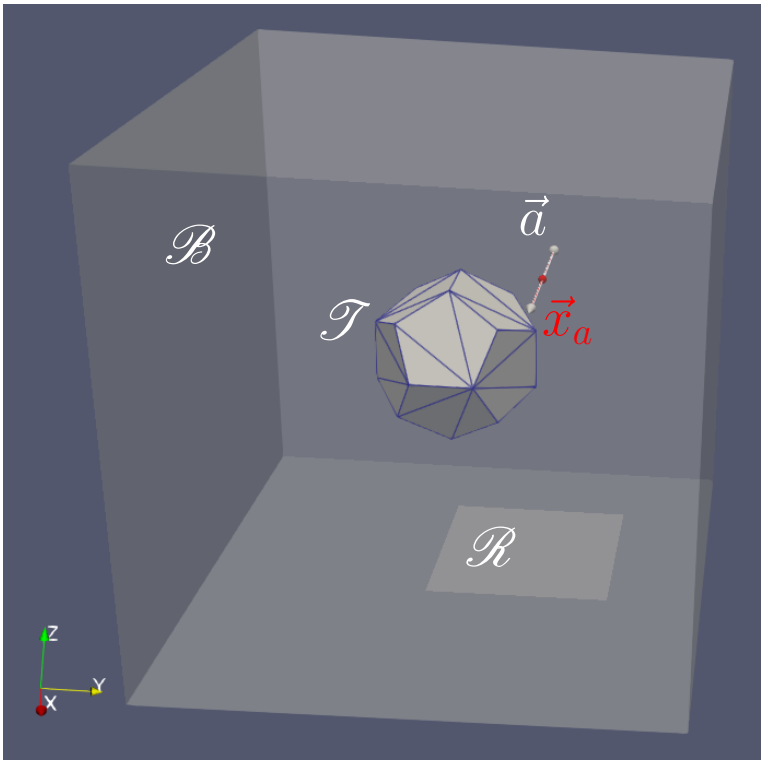


Figure 4.15 – Geometric configuration. The geometric centres of $\mathscr{R}$, $\mathscr{B}$, $\mathscr{T}$ are $\vec{x}_{c,r} \equiv (1, 0.5, -2)$, $\vec{x}_{c,b} \equiv (0, 0, 0)$ and $\vec{x}_{c,t} \equiv (0, 0, 0)$. The length of $\mathscr{B}$ is $\mathscr{L}_b = 9\,\mathrm{m}$ and the length of $\mathscr{R}$ is $\mathscr{L}_r \equiv 1\,\mathrm{m}$. The axis of rotation $\vec{a}$ is characterised by a vector $[1, 0, 0]$, passing through the point $\vec{x}_a \equiv (0, 1, 1)$.

4.4 Conclusion of the chapter

This chapter is dedicated to introducing the vectorized Monte-Carlo method, which allows preparing the application of the method of sensitivity model to a CSP system. We implement the method of sensitivity model into two examples involving complex and triangulated geometry using the technique of vectorization. This technique helps to accelerate the estimations and facilitate coding development.

Firstly, we built a model of intensity where the sources of intensity are specified. Based on this model, models of sensitivities can be built by using the tools in [32–34], where the sources of sensitivities are specified.

Secondly, following the three steps mentioned in Chapter.3 to solve a radiative model, we formulated the iterative integral formulation following the propagations of all the sources (Eq.4.7 and Eq.4.14 in the first example and Eq.4.17,4.32,4.34,4.36 and Eq.4.37 in the second example).

Finally, we distinguished the integrals having the same domains of integrations, and we wrote them into a vectorized form (Eq.4.15, Eq.4.44 and Eq.4.36). From this vectorized form, Monte Carlo algorithms can be designed to solve the radiative models.

This strategy of vectorization is about **vectorizing the transport of the sources that share the same path space.** If we wish to distinguish the sources that share the same path space, we need to verify the **locations** of the sources on the boundaries and the **transport** of the sources in the medium. When the sources are in the same locations on the boundary and transported by the same RTE in the medium, the sources can be vectorized.

		$k_a = 0, k_s = 0$			
P	1.61	W			
$\sigma[P]$	4.00×10^{-3}	W			
S_1	674.68×10^{-3}	W/m	$\tilde{S}_1$	679×10^{-3}	W/m
$\sigma[S_1]$	16.94×10^{-3}	W/m	$\sigma[\tilde{S}_1]$	71.22×10^{-3}	W/m
S_2	351.04×10^{-3}	W/m	$\tilde{S}_2$	348.50×10^{-3}	W/m
$\sigma[S_2]$	15.38×10^{-3}	W/m	$\sigma[\tilde{S}_2]$	71.22×10^{-3}	W/m
S_3	-1.03	W/m	$\tilde{S}_3$	-1.03	W/m
$\sigma[S_3]$	9.37×10^{-3}	W/m	$\sigma[\tilde{S}_3]$	71.22×10^{-3}	W/m
S_4	1.42	W/rad	$\tilde{S}_4$	1.43	W/rad
$\sigma[S_4]$	17.47×10^{-3}	W/rad	$\sigma[\tilde{S}_4]$	71.22×10^{-3}	W/rad
		$k_a = 1, k_s = 1$			
P	146.89×10^{-3}	W			
$\sigma[P]$	542.32×10^{-6}	W			
S_1	123.65×10^{-3}	W/m	$\tilde{S}_1$	113.70×10^{-3}	W/m
$\sigma[S_1]$	2.52×10^{-3}	W/m	$\sigma[\tilde{S}_1]$	9.24×10^{-3}	W/m
S_2	74.53×10^{-3}	W/m	$\tilde{S}_2$	53.60×10^{-3}	W/m
$\sigma[S_2]$	2.52×10^{-3}	W/m	$\sigma[\tilde{S}_2]$	9.24×10^{-3}	W/m
S_3	-342.57×10^{-3}	W/m	$\tilde{S}_3$	-324.40×10^{-3}	W/m
$\sigma[S_3]$	2.76×10^{-3}	W/m	$\sigma[\tilde{S}_3]$	9.24×10^{-3}	W/m
S_4	404.06×10^{-3}	W/rad	$\tilde{S}_4$	388.15×10^{-3}	W/rad
$\sigma[S_4]$	4.27×10^{-3}	W/rad	$\sigma[\tilde{S}_4]$	9.24×10^{-3}	W/rad
		$k_a = 0.1, k_s = 0.2$			
P	1.12	W			
$\sigma[P]$	3.15×10^{-3}	W			
S_1	552.40×10^{-3}	W/m	$\tilde{S}_1$	530.50×10^{-3}	W/m
$\sigma[S_1]$	15.69×10^{-3}	W/m	$\sigma[\tilde{S}_1]$	49.70×10^{-3}	W/m
S_2	285.64×10^{-3}	W/m	$\tilde{S}_2$	275×10^{-3}	W/m
$\sigma[S_2]$	15.59×10^{-3}	W/m	$\sigma[\tilde{S}_2]$	49.70×10^{-3}	W/m
S_3	-991.82×10^{-3}	W/m	$\tilde{S}_3$	-972×10^{-3}	W/m
$\sigma[S_3]$	10.10×10^{-3}	W/m	$\sigma[\tilde{S}_3]$	49.70×10^{-3}	W/m
S_4	1.30	W/rad	$\tilde{S}_4$	1.29	W/rad
$\sigma[S_4]$	20.12×10^{-3}	W/rad	$\sigma[\tilde{S}_4]$	49.71×10^{-3}	W/rad

Table 4.1 – Validations for three different groups of medium properties. k_s and k_a are the coefficients of scattering and the coefficient of absorption, respectively. In the left column, P and $\boldsymbol{S}$ are estimated by Algo.6 and Algo.7 with 10^6 Monte-Carlo realizations for each surface and for each edge of the triangles composing $\mathscr{T}$. In the right column, $\tilde{\boldsymbol{S}}$ is approximated by Finite Difference method (Eq.4.48), where $P(\ddot{\pi}_j + \Delta\ddot{\pi}_j)$ and $P(\ddot{\pi}_j - \Delta\ddot{\pi}_j)$ are estimated with 10^8 Monte-Carlo realizations.

For the first example, I and s_ρ are transported in the same way in the medium (Eq.4.2 and Eq.4.8). On the boundary, the source $\mathring{I}$ and $\mathring{s}_\rho$ are all on all the surfaces of $\mathscr{T}$. Therefore, the propagation of I and s_ρ can be vectorized.

For the second example, I and $s_i, \forall j \in \{1, 2, 3, 4\}$ are transported also in the same way in the medium (Eq.4.20 and Eq.4.23). However, sources of sensitivities $\dot{s}_{i,j}, \forall i \in \{1, 2, \ldots, n_{\mathscr{T}}\}, \forall j \in \{1, 2, 3, 4\}$ are found on each surface of $\mathscr{T}$ and $\hat{s}_{i,j,k}, \forall i \in \{1, 2, \ldots, n_{\mathscr{T}}\}, \forall j \in \{1, 2, 3, 4\}, \forall k \in \{1, 2, 3\}$ are found on each edge of each triangle $\mathscr{T}_i$. Therefore, the propagations of $\dot{s}_{i,j}$ and I (which is also a surface source) and the propagations of the line sources $\hat{s}_{i,j,k}$ are vectorized

separately. Even with this strategy of vectorization, it is still not easy to manage all the surface sources and line sources on every triangle of a complex and triangulated geometry. Therefore, later in Chapter.6, efforts will be made to transform all line sources into surface sources. Consequently, only surface sources are left, and the estimations of the observable and all its sensitivities can be completely vectorized.

5

Estimating sensitivities in a Solar Tower Power system

5.1 Introduction

In the previous chapter, we introduced a strategy of vectorization, aiming to apply the method of sensitivity model to complex and triangulated geometries. In this chapter, we will apply the method of sensitivity method to estimate the sensitivities in STP system by Monte-Carlo, following the same strategy of vectorization. We choose to study the heliostat field of Sierra SunTower CSP project which is composed of 24360 flat heliostats of $1.14m^2$. The heliostat field is located in Lancaster, USA (34.731 N, 118.139 W). Additional data can be found on https://solarpaces.nrel.gov/project/sierra-suntower. Contrarily to the majority of STP systems, heliostats are flat. We restrict our study to only one quarter of the heliostat field (6090 heliostats). In this chapter, surfaces are not described by triangles but only by flat rectangles. We choose to estimate the sensitivities of the translation of three directions (horizontal, vertical, and height), two rotations (elevation and azimuth), as well as the size of each heliostat. It represents 36540 sensitivities. Therefore, the vectorization technique becomes crucial to handle the large number of parameters.

We benefit from the models of sensitivity to interpret the sensitivities physically and to perform sensitivity analysis. Thanks to the sensitivity models, the contribution of physical events (blocking, spillage, and shadowing) to each of the 36540 sensitivities can also be estimated and analyzed.

This paper has been accepted by the journal *Solar Energy* [22]: He, Zili, Paule Lapeyre, Stephane Blanco, Simon Eibner, Mouna El Hafi, and Richard Fournier. "Monte-Carlo Estimation of Geometric Sensitivities in Solar Power Tower Systems of Flat Mirrors." Solar Energy 253 (March 15, 2023): 9–29. https://doi.org/10.1016/j.solener.2023.02.013.

5.2 Estimating the geometric sensitivities

Monte-Carlo estimation of geometric sensitivities in Solar Power Tower systems of flat mirrors

Zili He[1,*], Paule Lapeyre[2], Stéphane Blanco[3], Simon Eibner[1], Mouna El Hafi[1], and Richard Fournier[3]

[1]IMT Mines Albi, RAPSODEE - UMR CNRS 5302, Campus Jarlard, 81013 Albi, France
[2]Department of Mechanical and Mechatronics Engineering, University of Waterloo, 200 University Avenue West, Waterloo, ON N2I 3G1, Canada
[3]LAPLACE - UMR CNRS 5213, 118, Route de Narbonne, 31062 Toulouse, France
[*]Corresponding author: zili.he@mines-albi.fr

March 6, 2023

Abstract

Optical optimizations for a Concentrated Solar Power (CSP) system are currently limited largely to gradient-free methods, since the gradient is hard to obtain by using the existing numerical optics tools available in the community. This article aims to build new algorithms of the Monte-Carlo type, which numerically estimate the gradient of power impacting the receiver with respect to the geometric parameters that characterize the geometric status of the heliostats in the heliostats field, for a Solar Power Tower (SPT) system.

Physical models will be built for the specific intensity and also for its derivatives to the geometric parameters of the heliostats, also called geometric sensitivities of intensity. Similar to the intensity but with their own models, they are regarded as physical quantities emitted, absorbed and reflected in the system. They carry the perturbations of intensity as information, corresponding to the relations between the geometric parameters and the physical events in the SPT system: blocking, spillage, shadowing, etc. These relations will be distinguished and discussed.

Finally, not only the gradient but also the contributions of physical events (blocking, spillage, shadowing, etc.) to the gradient can be estimated for further sensitivity analysis and optimization processes.

Nomenclature

I	Intensity
N	Number of realizations of Monte-Carlo estimation
P	Total power impacting receiver
$S_{i,j}$	Sensitivity of Power
$\boldsymbol{S}$	Matrix of sensitivity of power
$\ddot{\boldsymbol{\pi}}$	Matrix of parameters
$\ddot{\pi}_{i,j}$	Geometric parameter of a heliostat
$\mathcal{G}$	Total set of the position in the system
$\mathcal{S}$	Unit Sphere
$\mathring{I}$	Source of intensity
$\mathring{s}$	Source of sensitivity
$\mathring{s}^{blo}$	Source of sensitivity: backward blocking
$\mathring{s}^{shad-b}$	Source of sensitivity: backward shadowing
$\mathring{s}^{shad-f}$	Source of sensitivity: forward shadowing
$\mathring{s}^{tar}$	Source of sensitivity: targeting
$\mathscr{H}$	Heliostats, $\mathscr{H} \subset \partial\mathcal{G}$
$\mathscr{H}_i$	A Heliostat, $\mathscr{H}_i \subset \mathscr{H}$
$\mathscr{H}_i^+$	Reflecting surface of a heliostat, $\mathscr{H}_i^+ \subset \mathscr{H}_i$
$\mathscr{H}_i^-$	Rearward surface of a heliostat, $\mathscr{H}_i^- \subset \mathscr{H}_i$
$\mathscr{O}_l$	Lower outline, $\mathscr{O}_l \subset \partial\mathcal{G}$
$\mathscr{O}_s$	Upper outline, $\mathscr{O}_s \subset \partial\mathcal{G}$
$\mathscr{R}$	Receiver, $\mathscr{R} \subset \partial\mathcal{G}$
$\partial\mathcal{G}$	Total set of the position on the boundaries, $\partial\mathcal{G} \subset \mathcal{G}$
$\vec{\omega}$	Direction in the system, $\vec{\omega} \in \mathcal{S}$
$\vec{\omega}_b$	Direction attached on $\vec{x}_b$, $\vec{\omega}_b \in \mathcal{S}$
$\vec{\omega}_c$	Solar position: pointing from the center of the sun to $\vec{x}_p$.
$\vec{\omega}_p$	Direction attached on $\vec{x}_p$, $\vec{\omega}_p \in \mathcal{S}$
$\vec{\omega}_r$	Direction attached on $\vec{x}_r$, $\vec{\omega}_r \in \mathcal{S}$
$\vec{a}_{i,\phi}$	Axis of a heliostat for azimuth angle
$\vec{a}_{i,\theta}$	Axis of a heliostat for elevation angle
$\vec{n}_i$	Normal of a heliostat, $\vec{n}_i \in \mathcal{S}$
$\vec{n}_r$	Normal of the receiver, $\vec{n}_r \in \mathcal{S}$
$\vec{v}$	Velocity of deformation
$\vec{x}$	Position in the system, $\vec{x} \in \mathcal{G}$
$\vec{x}_b$	Position on rearward surface of a heliostat, $\vec{x}_b \in \mathscr{H}_i^-$
$\vec{x}_p$	Position on reflecting surface of a heliostat, $\vec{x}_p \in \mathscr{H}_i^+$
$\vec{x}_r$	Position on receiver, $\vec{x}_r \in \mathscr{R}$
$n_{\ddot{\pi}}$	Number of parameters for each heliostat
$n_{\mathscr{H}}$	Number of heliostats
$s_{i,j}$	Sensitivity of Intensity

Acronyms

CBMC	Collision-Based Monte-Carlo algorithms
CSP	Concentrated Solar Power
DNI	Direct Normal Irradiation
IFMC	Integral Formulation Monte-Carlo
RTE	Radiative Transfer Equation
SPT	Solar Power Tower

Greek symbols

ρ	Reflectivity
σ	Standard deviation

Subscrips

b	Rearward surface of a heliostat
i	Index of heliostat
j	Index of parameter of a heliostat
k	Index of borders of a heliostat
l	Lower outline
p	Reflecting surface of a heliostat
r	Receiver
s	Upper outline

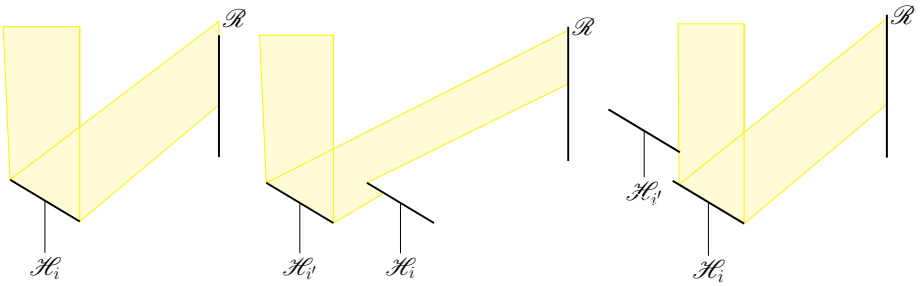
1 Introduction

Concentrating solar power (CSP) technology is one of the promising options to replace fossil fuel resources. Solar Power Tower (SPT) is a mature technology, and commercial facilities that follow this principle are currently in operation [1]. The optical system described by the heliostats field usually contributes to about 40% of the total cost of an SPT system[2]. The optical study of SPT systems has been an active research field [3–5]. Evaluations and optimizations of optical performance for SPT systems are essential.

As regards the evaluation of optical performance, there are two common categories of optical modeling methods: Monte-Carlo ray-tracing methods [6–9] and the cone optics convolution-based methods [10–12]. Compared to the latter kinds of method, the Monte-Carlo ray tracing methods are more versatile and can reproduce the real intersections of photons in complex geometries[13]. Several Monte-Carlo-based tools are compared to the convolution methods in the study[14], revealing that the Monte-Carlo methods have better performance in accuracy.

However, the optimizations of optical performance, which are usually related to geometric optimizations are limited largely to gradient-free methods [15, 16]. An important reason for choosing gradient-free methods is usually that 'no gradient information is available from Monte-Carlo ray tracing methods' [17, 18]. The gradient-based methods are usually used when the dimension of the parameter space of optimization is large [19], which is often the case in CSP optimization. Limited by the dimension of the parameter space, researchers can only perform optimization with small numbers of parameters. Diago *et al.* optimized only the height and the diameter of the tertiary concentrator in a Beam-Down system. In order to decrease the number of parameters to be optimized, Yu *et al.* optimized the aiming strategy by regrouping the heliostats, which point to some fixed points on the receiver. Furthermore, the gradient-free methods are usually treated as a "black box" tool, and the knowledge of the gradient helps to understand the parameters' influences on the optimization's objective.

Only a few studies have applied gradient-based methods to optimize the geometry of solar concentrators. Yang *et al.* [20] suggested approximating it by using area ratios. However, this method introduces an empiric weight. In studies carried out by Marston *et al.*, the gradient is approximated by a finite difference [21, 22], which is also called the resimulation method by Gobet [23]. However, beyond the fact that these studies deal only with simple geometrical situations, these authors claim that associated uncertainties are difficult to control. Regarding complex geometries such as a heliostat field in an SPT system with a large number of parameters, they cannot be treated in such a way (using finite difference approximation) because of the huge computational time and the lack of accuracy. Besides this approach of approximation, in the framework of Integral Formulation Monte-Carlo (IFMC) [24], the impacting power on the receiver in an SPT system is written as an integral formulation based on the physics of radiation. By differentiating and reformulating the corresponding integral formulation (for example, with respect to the angular spreading coefficient [24]), the gradient is also formulated as an integral formulation and can be estimated through



(a) Spillage effect. (b) Blocking effect. (c) Shadowing effect.

Figure 1: Spillage, blocking, and shadowing effects are common physical events in the SPT system. In this work, we further classify them into spillage, forward-blocking, backward-blocking, forward-shadowing, and backward-shadowing effects to estimate and analyze the sensitivity of geometric parameters. In the figure, $\mathscr{H}_i$ and $\mathscr{H}_{i'}$ are the heliostats and $\mathscr{R}$ is the receiver.

the Monte-Carlo method[25]. However, when the differentiated parameter is a geometric parameter that interacts in the integral domain (for example, the size of heliostats), great formal development efforts are needed[26]. Integrating this approach to a complex system such as the heliostats field of an SPT system seems nowadays to be impracticable. A very recent work [27] proposes an alternative to the work performed by [26]. This new approach proposes that the intensity and the geometric sensitivity of intensity (the derivative of the intensity with respect to a geometric parameter) should be regarded as two physical quantities emitted, transported, and intersected in a radiative system [27]. Based on this point of view, we can benefit from years of research in the radiative transfer field, on modeling the intensity to model the geometric sensitivity in an SPT system, making estimating the gradient in such a complex system possible. In parallel to our works, the literature shows that shape sensitivity calculations using the radiative transfer equation and its derivative has been the subject of much work in the computer graphics community [28–31]. Their objective is to reconstitute shapes to create the desired image by inversion. For many applications, such as remote sensing or deep neural networks for artificial intelligence [32] this "differentiable rendering" will become increasingly important in the coming years in this community. While we can take advantage of computer graphics tools to deal with the complex geometry of an SPT system, we still need to understand how the phenomena in SPT systems (blocking in Fig.1b, shadowing in Fig.1c, and spillage in Fig.1a) influence the sensitivities, which is also the objective of this work. Therefore, we will build a model of sensitivity and analyze the sources of the sensitivities in this model, which is specific to our approach [27, 33]. The sensitivities are propagated instead of the intensities following the Radiative Transport Equation (RTE) and the sensitivity sources are identified by this approach on the borders or on the surfaces of the studied system when we change geometric parameters. Consequently, we will see in this work that sensitivities can be presented in an intuitive manner: displacements and rotations of heliostats create sensitivity sources on the edges and surfaces of the heliostats that move according to different parameters of displacement and rotation.

As stated above, this paper aims to develop a method to estimate the

unit	W/m
$S^{tar}_{676,1}$	70.63 ± 0.94
$S^{blo}_{676,1}$	-30.69 ± 0.58
$S_{676,1}$	39.93 ± 1.51

Table 1: Sensitivity and its contributions (estimated values and their standard deviations) for the translation X of the 676^{th} heliostat. For the 676^{th} heliostat in Fig.2 (indexed as 676 in the table), the sensitivity of its translation X (indexed as 1 in the table) has two contributions. The translation will block its neighboring heliostat in its back ($S^{blo}_{676,1}$ is negative). However, the translation helps it to be blocked less by its neighboring heliostat on its front ($S^{tar}_{676,1}$ is positive). The sensitivity of translation X for this heliostat is then the sum of these two contributions ($S_{676,1}$).

geometric gradient (sensitivities) of SPT systems and to indicate how the blocking, spillage, and shadowing effects contribute to the gradient (sensitivities). The implementation of this method leads to different results that will be of interest to the SPT community to design (Fig.2a,2b,2c,2d) solar power plants, and also for their remote control (Fig.2e,2f). In Fig.2, x-y represents the ground plane, z is the height of the heliostat position, and each point is dedicated to a heliostat in the field. Furthermore, for each parameter of each heliostat, a detailed analysis of the contributions (spillage, blocking, shadowing) of sensitivity can be carried out (Table.1).

Finally, the article is organized into three parts:

- Section.2 provides the models of transport for the intensity and its derivative towards each heliostat five degrees of freedom (three translations and two rotations) and its size.

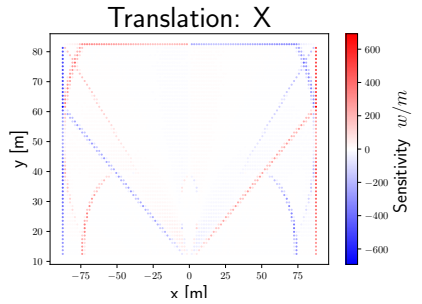
- Section.3 provides the integral formulations and algorithms to estimate the power collected by the receiver and the different sensitivities towards these six geometric parameters for each heliostat.

- Section.4 is dedicated to validation purposes: the estimation of the different sensitivities is compared to the approximation by the finite difference method. Comparisons and discussions about these two methods will be undertaken.

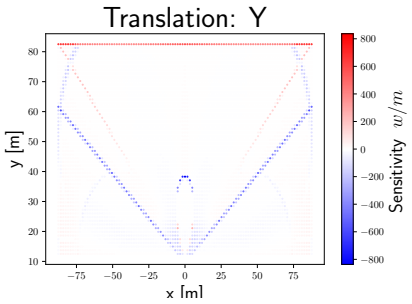
2 General models

In this work, we address the question of estimating the power P impacting the receiver of an SPT system with flat squared heliostats (Fig.3), as well as its derivatives $S_{i,j} \equiv \partial_{\ddot{\pi}_{i,j}} P$ of geometric parameters $\ddot{\pi}$ with i the index of a heliostat in the heliostats field and j the index of a geometric parameter ($j \in \{1, 2, 3, 4, 5, 6\}$, see Fig.4).

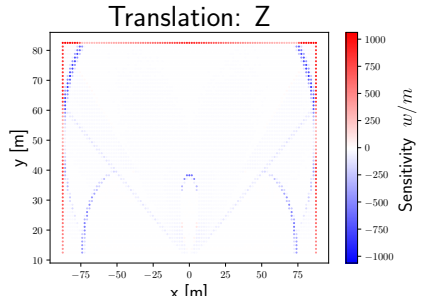
In order to numerically estimate P, we will build a model following the standard radiative transfer physics, using the intensity I as the descriptor. The transport of I in the medium is governed by a partial differential equation at any



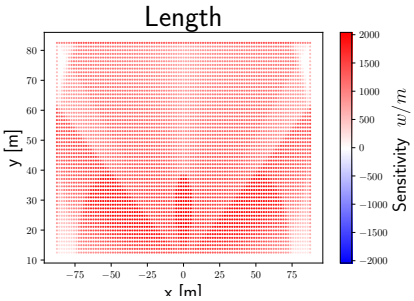
(a) Sensitivity of position-x for each heliostat.



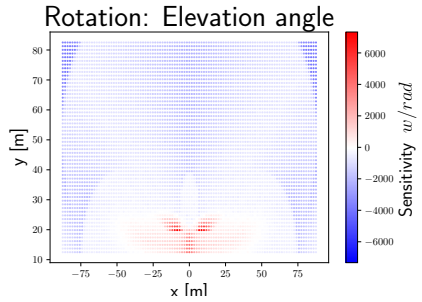
(b) Sensitivity of position-y for each heliostat.



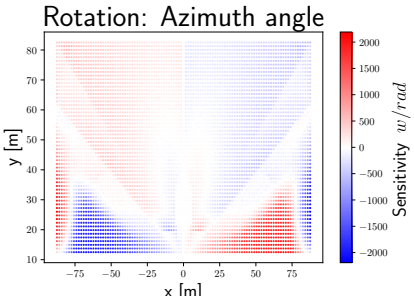
(c) Sensitivity of position-z for each heliostat.



(d) Sensitivity of the length of borders for each heliostat.



(e) Sensitivity of elevation angle for each heliostat.



(f) Sensitivity of azimuth angle for each heliostat.

Figure 2: Sensitivity of the total impacting power P $[w]$ for each heliostat in the field. Each point is dedicated to a heliostat, pointing to the center of the receiver at the moment of solar noon at the summer solstice. The layout corresponds to a real heliostat field: Sierra SunTower [34] (receiver at 50 m height at (0,0) and more details of the heliostats field are in Table.6 and Section.4.2). The corresponding color indicates its sensitivity with respect to its positions, rotations, and lengths. (10^6 Monte-Carlo realizations on each border of each heliostat and 10^8 Monte-Carlo realizations on each surface of heliostats).

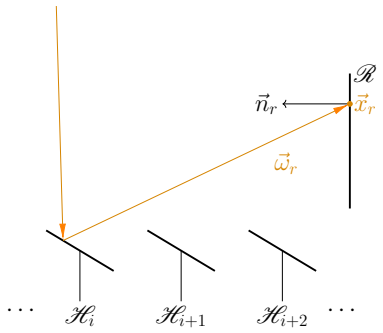


Figure 3: An SPT system consists of heliostats $\mathscr{H}_i$ (with an index of i) on the ground and a receiver $\mathscr{R}$ at the top of a tower.

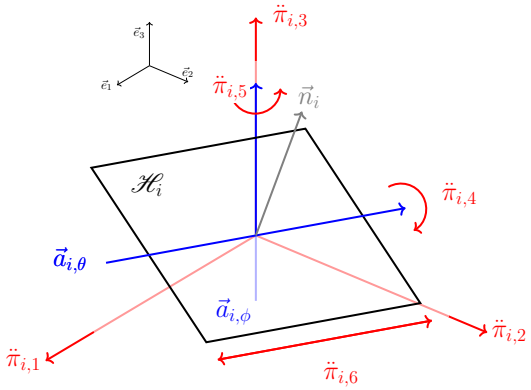


Figure 4: A heliostat ($\mathscr{H}_i$) has 5 degrees of freedom, which includes the translation ($\ddot{\pi}_{i,1}$,$\ddot{\pi}_{i,2}$ and $\ddot{\pi}_{i,3}$), following the three axes in the global coordinate system ($\vec{e}_1$, $\vec{e}_2$ and $\vec{e}_3$) and the rotation ($\ddot{\pi}_{i,4}$ and $\ddot{\pi}_{i,5}$) corresponding to an altazimuth mount according to the two vectors of rotation ($\vec{a}_{i,\theta}$ and $\vec{a}_{i,\phi}$). Also, the length of the size of the heliostat is included in this study ($\ddot{\pi}_{i,6}$). Those six parameters are indexed by j.

locations $\vec{x}$ inside the field $\mathcal{G}$, following a vector of direction $\vec{\omega}$ in a unit sphere $\mathcal{S}$ (RTE). On the boundaries $\partial\mathcal{G}$, the transport of I is governed by the boundary conditions, where the locations are noted with index, see Section.2.4 for details.

Equivalent to P, in order to numerically estimate $S_{i,j}$, we will build a radiative model in the medium and on the boundaries using the geometric sensitivity of intensity $s \equiv \partial_{\ddot{\pi}_{i,j}} I$ as the descriptor[27]. This descriptor carries the perturbation of I as information, corresponding to the relations between the geometric parameters $\ddot{\pi}_{i,j}$ and the physical events in the SPT system: blocking, spillage, shadowing, etc.

I and $s_{i,j}$ will be modeled locally in the medium and boundary conditions. After that, P and $S_{i,j}$ can be observed by formulating integral formulations of I and of $s_{i,j}$.

2.1 Definitions

2.1.1 Geometric sensitivity of intensity

The specific intensity used to model the STP system is considered independent of the wavelength [14, 35]. It is defined as $I \equiv I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}})$, where $\ddot{\boldsymbol{\pi}}$ is the matrix composed of $\ddot{\pi}_{i,j}$ (see Fig.4):

$$\ddot{\boldsymbol{\pi}} \equiv [\ddot{\pi}_{i,j}]\,. \tag{1}$$

$\vec{x}$, $\vec{\omega}$ and $\ddot{\boldsymbol{\pi}}$ are independent variables and matrix $\boldsymbol{s}$ is defined as the derivative of I with respect to $\ddot{\boldsymbol{\pi}}$ [33]:

$$\boldsymbol{s}(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) \equiv [s_{i,j}(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}})] = \partial_{\ddot{\boldsymbol{\pi}}} I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}). \tag{2}$$

The components of $\boldsymbol{s}$ are:

$$s_{i,j}(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) = \partial_{\ddot{\pi}_{i,j}} I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}). \tag{3}$$

2.1.2 Geometric sensitivity of power

We implement this work within the framework of IFMC [24]. P can be estimated by the product of spatial integral and angular integral of incoming I on the receiver:

$$P(\ddot{\boldsymbol{\pi}}) = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r I(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \tag{4}$$

where $\vec{x}_r$ is a position on the receiver, $\vec{\omega}_r$ the direction of observation, $\vec{n}_r$ the normal of the receiver, and $\mathscr{R}$ the absorbing surface of the receiver (see Fig.3).

Matrix $\boldsymbol{S}$ is defined as the derivative of P with respect to $\ddot{\boldsymbol{\pi}}$, which can be developed in a straight forward way since $\vec{x}_r$ and $\vec{\omega}_r$ are independent of $\ddot{\boldsymbol{\pi}}$ that characterizes only the heliostats:

$$\boldsymbol{S}(\ddot{\boldsymbol{\pi}}) \equiv [S_{i,j}(\ddot{\boldsymbol{\pi}})] = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r \boldsymbol{s}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}). \tag{5}$$

The components of $\boldsymbol{S}$ are:

$$S_{i,j}(\ddot{\boldsymbol{\pi}}) = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r s_{i,j}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}). \tag{6}$$

We aim to estimate $P(\ddot{\boldsymbol{\pi}})$ and $S_{i,j}(\ddot{\boldsymbol{\pi}})$ by building a model of I and a model of $s_{i,j}$ in an SPT system.

2.2 Assumptions

The following assumptions are made for modeling I and $s_{i,j}$ in an SPT system:

1. The Pill-Box model is used to describe the sun shape (Eq.15).
2. The heliostats are considered perfectly specular, and beam pointing error is neglected.
3. The reflectivity of heliostats is considered homogeneous on the surface.
4. The medium between heliostats and the receiver is considered transparent.

It is noted that even though they are typical assumptions for modeling an SPT system, more realistic modelings can be realized based on the theoretical studies[27, 33] without specific difficulties (more realistic sun shape model, spatially and angularly inhomogeneous reflectivity, imperfect reflection, semi-transparent medium, etc.). In this work, we aim to build a basic model for I and $s_{i,j}$ in an SPT system. More realistic assumptions can be taken into account based on the developed basic model in this work. For example, if an angularly inhomogeneous reflectivity is considered in future work, it will have extra influences on the sensitivity with respect to the rotations of the heliostats. These extra influences can be considered as an additional source of sensitivity based on the developed model in this work.

2.3 Transport in the medium

2.3.1 Transport of intensity

The transport of intensity in the medium is modeled by RTE [36]. In an inhomogeneous medium, the collisions of absorption and scattering are described by the collision operator $\mathcal{C}$ [29]:

$$\frac{\partial I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})}{\partial \vec{x}} = \mathcal{C}[I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})] + \kappa_a(\vec{x})I_b(\vec{x}) \tag{7}$$

with

$$\begin{aligned}
&\mathcal{C}[I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})] = \\
&-\kappa_a(\vec{x})I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) \\
&-\kappa_s(\vec{x})I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) \\
&+\frac{1}{4\pi}\int_{\omega'=4\pi}\kappa_s(\vec{x})I(\vec{x},\vec{\omega}',\ddot{\boldsymbol{\pi}})\Phi(\vec{\omega}',\vec{\omega})d\vec{\omega}'
\end{aligned} \tag{8}$$

where κ_a is the absorption coefficient, κ_s the scattering coefficient, $\vec{x}$ the position vector in the medium, $\vec{\omega}$ the vector of propagation and $\Phi(\vec{\omega}',\vec{\omega})$ is the phase-function. Under the assumption of local thermal equilibrium, the source $\kappa_a(\vec{x})I_b(\vec{x})$ (in Eq.7) is isotropic and is a function of temperature, and I_b is the specific intensity at equilibrium following the Planck's law.

In this work, we will consider a transparent and cold medium (where $\kappa_s = 0$, $\kappa_a = 0$, and $I_b(\vec{x}) = 0$). Therefore, the equation of transport for I used in this work is:

$$\frac{\partial I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})}{\partial \vec{x}} = 0 \tag{9}$$

2.3.2 Transport of geometric sensitivity of intensity

We differentiate Eq.7 with respect to the geometric parameter $\ddot{\boldsymbol{\pi}}$ on the both sides:

$$\frac{\partial^2 I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})}{\partial \vec{x}\partial \ddot{\boldsymbol{\pi}}} = \frac{\partial\big[\mathcal{C}[I(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})]\big]}{\partial \ddot{\boldsymbol{\pi}}} \tag{10}$$

Referring to the definition of the sensitivity (Eq.2), the following equation transport is yielded:

$$\frac{\partial \boldsymbol{s}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})}{\partial \vec{x}} = \mathcal{C}[\boldsymbol{s}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})] \tag{11}$$

The development from Eq.7 to Eq.11 is relatively straightforward. First, it is because the medium's properties (κ_a, κ_s, Φ and I_b) are not dependent on $\ddot{\boldsymbol{\pi}}$, and secondly, $\vec{x}$, $\vec{\omega}$ and $\ddot{\boldsymbol{\pi}}$ are three independent variables.

Referring to Eq.2 and Eq.3, the equation of transport of each of the component $s_{i,j}$ is [33]:

$$\frac{\partial s_{i,j}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})}{\partial \vec{x}} = \mathcal{C}[s_{i,j}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})] \tag{12}$$

with $\mathcal{C}$ the same collision operator as it is in Eq.8:

$$\begin{aligned}
&\mathcal{C}[s_{i,j}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})] = \\
&-\kappa_a(\vec{x}) s_{i,j}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) \\
&-\kappa_s(\vec{x}) s_{i,j}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}}) \\
&+\frac{1}{4\pi}\int_{\omega'=4\pi} \kappa_s(\vec{x}) s_{i,j}(\vec{x},\vec{\omega}',\ddot{\boldsymbol{\pi}})\Phi(\vec{\omega}',\vec{\omega})d\vec{\omega}'
\end{aligned} \tag{13}$$

Finally, the collision operator $\mathcal{C}$ is also applied to the geometric sensitivities when describing its transport in the medium. $s_{i,j}$ can be regarded as a physical quantity transported in the medium. Similar to the intensity I, the sensitivity of intensity $s_{i,j}$ is also absorbed and scattered. This new point of view allows us to study the geometric sensitivity of intensity practically, benefiting from years of research in the transport of intensity in the physics of radiative transfer.

In this work, we will consider a transparent and cold medium. The intensity and the geometric sensitivity will have the same model of transport in the medium:

$$\frac{\partial s_{i,j}(\vec{x},\vec{\omega},\ddot{\boldsymbol{\pi}})}{\partial \vec{x}} = 0 \tag{14}$$

In order to create a similar physical picture for the transport of I and $s_{i,j}$ in the medium, we can consider that the medium is also 'transparent' for the geometric sensitivities $s_{i,j}$.

2.4 Boundary conditions

I and $s_{i,j}$ are transported following Eq.9 and Eq.14 in the medium, while on the boundaries, they can be emitted, reflected, or absorbed.

Seeing Fig.5, positions and directions on the boundaries are noted with an index of s on $\mathscr{O}_s$, r on $\mathscr{R}$, p on $\mathscr{H}_i^+$, b on $\mathscr{H}_i^-$ and O_l on $\mathscr{O}_l$. We will build the boundary conditions for I and then differentiate them with respect to $\ddot{\pi}_{i,j}$ to establish the boundary conditions for $s_{i,j}$.

2.4.1 Upper outline $\mathscr{O}_s$

In an SPT system, the only source of intensity is the sun. Most of the time, we use the sun-shape models, which are usually angular functions, to describe the arriving intensity in the system[14, 37, 38]. To be noticed that this work is a stationary study, which means it is established for a fixed solar position.

The pill-box sun shape is used in this work:

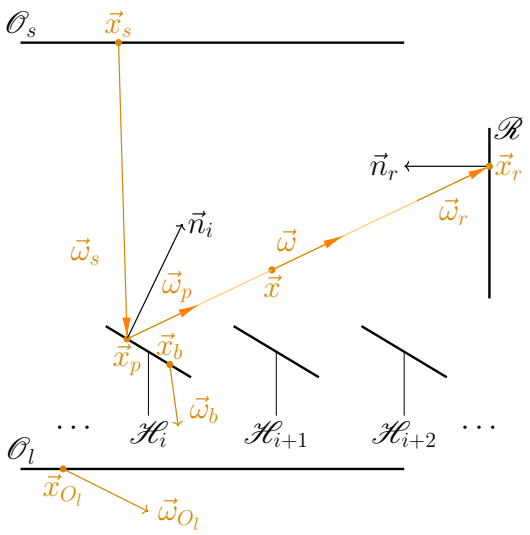


Figure 5: The system consists of heliostats $\mathscr{H}_i$ (with index of i) on the ground with $\mathscr{H}_i^+$ its reflecting surface and $\mathscr{H}_i^-$ its rearward surface, a receiver $\mathscr{R}$, an upper outline $\mathscr{O}_s$ and an lower outline $\mathscr{O}_l$. Solar intensity is reflected by $\mathscr{H}_i^+$ and collected on the $\mathscr{R}$.

$$I(\vec{x}_s, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \equiv \mathring{I}(\vec{\omega}_s) = I_0 \mathcal{H}(\vec{\omega}_s \cdot \vec{\omega}_c - \cos(\theta_{disk})) \tag{15}$$

where $\vec{\omega}_c$ is a vector that characterizes the solar position (fixed in this work), I_0 the constant intensity coming from the sun, $\mathcal{H}$ the Heaviside function and $\theta_{isk} = 0.0046\ rad$ the radial angular radius of the solar disk. Moreover, this boundary condition is noted $\mathring{I}$ because it is the source of I in the system.

The upper outline $\mathscr{O}_s$ can be considered emitting intensity towards the system, but it is not reflecting the incoming intensity. Therefore, this boundary can be considered as a 'black body' for I.

The derivative of I on $\mathscr{O}_s$ with respect to $\ddot{\pi}_{i,j}$ is therefore null, because $\ddot{\pi}_{i,j}$ characterizes the geometry of heliostats instead of $\mathscr{O}_s$:

$$s_{i,j}(\vec{x}_s, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) = 0. \tag{16}$$

Following Eq.16, no $s_{i,j}$ is emitted by the sun, and all $s_{i,j}$ transported to the upper outline $\mathscr{O}_s$ are not reflected but absorbed. Therefore, in order to create a similar physical picture for the transport of I and $s_{i,j}$ on the boundaries, $\mathscr{O}_s$ can be regarded as a 'cold black body' for $s_{i,j}$.

2.4.2 Receiver $\mathscr{R}$ and lower outline $\mathscr{O}_l$

$\mathscr{R}$ and $\mathscr{O}_l$ in our system (Fig.5) are considered 'cold black body' for I. All the intensity is not hitting the heliostats, and the receiver will be lost. The radiative emission of the receiver is not included in the model since the heat transfer of the receiver is out of the scope of this work.

The boundary conditions of receiver $\mathscr{R}$ and lower outline $\mathscr{O}_l$ are then established:

$$\begin{cases} I(\vec{x}_r, -\vec{\omega}_r, \ddot{\boldsymbol{\pi}}) = 0 \\ I(\vec{x}_{O_l}, -\vec{\omega}_{O_l}, \ddot{\boldsymbol{\pi}}) = 0 \end{cases} \tag{17}$$

where $-\vec{\omega}_r$ and $-\vec{\omega}_{O_l}$ are the outgoing direction of propagation of I from the receiver and the lower outline (Fig.5).

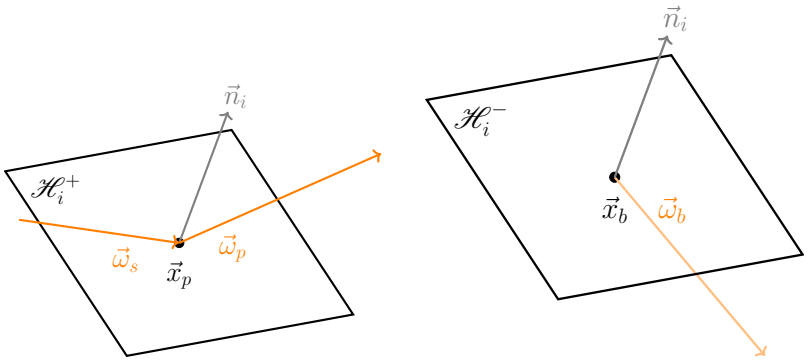
Since $\ddot{\pi}_{i,j}$ only characterizes the heliostats, there are no $s_{i,j}$ going out from these boundaries:

$$\begin{cases} s_{i,j}(\vec{x}_r, -\vec{\omega}_r, \ddot{\boldsymbol{\pi}}) = 0 \\ s_{i,j}(\vec{x}_{O_l}, -\vec{\omega}_{O_l}, \ddot{\boldsymbol{\pi}}) = 0. \end{cases} \tag{18}$$

In order to create a similar physical picture for the transport of I and $s_{i,j}$ on the boundaries, $\mathscr{R}$ and $\mathscr{O}_l$ can also be regarded as 'cold black body' for $s_{i,j}$. All $s_{i,j}$ that are not hitting the heliostat or the receiver will be lost, and $s_{i,j}$ are not emitted from these boundaries.

2.4.3 Heliostat $\mathscr{H}_i$

Intensity: We define $\vec{n}_i$ as the normal of the reflecting surface of $\mathscr{H}_i$. The boundary conditions of I on a heliostat $\mathscr{H}_i$ are modeled separately by two parts: one for the reflecting surface $\mathscr{H}_i^+$ (where $\vec{\omega}_p \cdot \vec{n}_i > 0$), the other for the rearward surface $\mathscr{H}_i^-$ (where $\vec{\omega}_b \cdot \vec{n}_i < 0$), see Fig.6. Also, the heliostats are considered 'cold', which means they do not emit intensity.



(a) Reflecting surface where $\vec{\omega}_p \cdot \vec{n}_i > 0$ (b) Rearward surface where $\vec{\omega}_b \cdot \vec{n}_i < 0$

Figure 6: Orientation convention of the heliostat $\mathscr{H}_i$

The transport of intensity on the reflecting surface of the heliostat can be described by:

$$I(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \rho I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \tag{19}$$

where ρ is the reflectivity, $I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}})$ the incoming intensity and $\vec{\omega}_p$ the direction reflected by heliostat of $\vec{\omega}_s$ (Fig.6a).

The transport of I on the rearward surface of $\mathscr{H}_i$ can be described by:

$$I(\vec{x}_b, \vec{\omega}_b, \ddot{\boldsymbol{\pi}}) = 0. \tag{20}$$

The rearward surface of $\mathscr{H}_i$ is also considered a 'cold black body'.

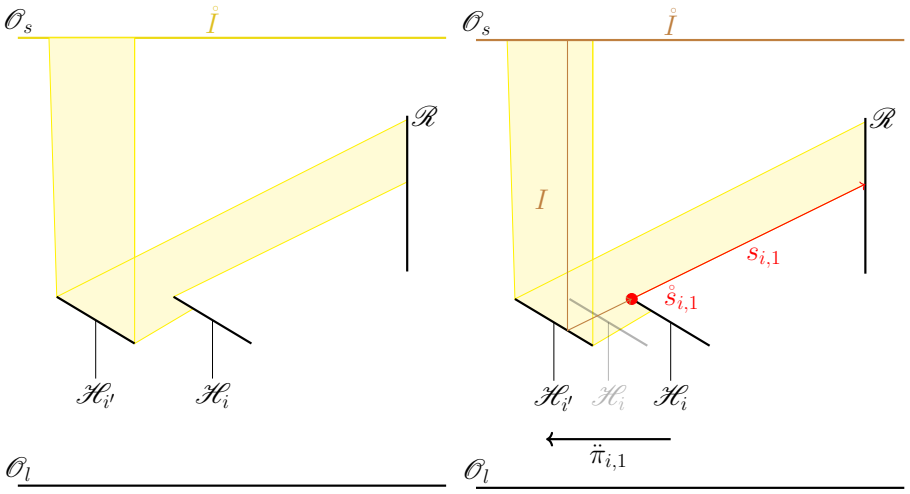
Geometric sensitivity of intensity: The development of the boundary conditions of $s_{i,j}$ for $\mathscr{H}_i$ is less straightforward. [33] proposed a model of geometric sensitivity on reflecting surfaces. Based on this model, the boundary conditions of $s_{i,j}$ is yielded for reflecting surface (Eq.21, referring to Fig.6a) and also for the rearward surface (Eq.22, referring to Fig.6b). It is found that the boundary condition of the reflecting surface contains emission terms (the sources) and a reflection term, see details in Appendix.A.

$$\begin{aligned} s_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) =& \mathring{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \\ &+ \rho s_{i,j}(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \end{aligned} \tag{21}$$

$$s_{i,j}(\vec{x}_b, \vec{\omega}_b, \ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}(\vec{x}_b, \vec{\omega}_b, \ddot{\boldsymbol{\pi}}) \tag{22}$$

We note $\mathring{s}_{i,j}$ as the sources of $s_{i,j}$ (as we noted $\mathring{I}$ as the source of I).

The sources $\mathring{s}_{i,j}$ can be physically regarded as the local perturbations of I with respect to $\ddot{\pi}_{i,j}$ on $\mathscr{H}_i$, while $s_{i,j}$ is such perturbations transported in the radiative system (see Fig.7). Usually, the sources of sensitivity $\mathring{s}_{i,j}$ are functions of I, where the two models of transport are coupled.



(a) Source of intensity $\mathring{I}$ (b) Source of sensitivity $\mathring{s}_{i,1}$: An example.

Figure 7: The source of intensity is on the boundary $\mathscr{O}_s$. In this example, The source of sensitivity is on the border of heliostat $\mathscr{H}_i$ because the perturbation of $\ddot{\pi}_{i,1}$ will create locally a perturbation of I on the border of $\mathscr{H}_i$, which causes a perturbation of P. Also, the source $\mathring{s}_{i,j}$ is coupled with the arrived intensity $\mathring{I}$. Therefore, the model of $s_{i,j}$ is coupled with the model of I. The optical path of $s_{i,j}$ is in red, and the coupled optical path of I is in brown.

Moreover, different types of perturbations (sources) are distinguished, related to blocking, shadowing, and spillage effects. Therefore, we further classify the sources by their contributions to these different physical effects.

For the reflecting surface $\mathscr{H}_i^+$:

$$\begin{aligned} \mathring{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}^{tar}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) + \mathring{s}_{i,j}^{blo}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \\ + \mathring{s}_{i,j}^{shad-b}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \end{aligned} \tag{23}$$

and for the rearward surface $\mathscr{H}_i^-$:

$$\mathring{s}_{i,j}(\vec{x}_b, \vec{\omega}_b, \ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}^{shad-f}(\vec{x}_b, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}). \tag{24}$$

$\mathring{s}_{i,j}^{tar}$ is named the source of targeting. It is the perturbation caused by the targeting or missing of the reflected intensity on $\mathscr{H}_i^+$, with respect to $\ddot{\pi}_{i,j}$. It is separated into two parts:

$$\mathring{s}_{i,j}^{tar} = \mathring{s}_{i,j}^{tar,spatial} + \mathring{s}_{i,j}^{tar,angular}. \tag{25}$$

$\mathring{s}_{i,j}^{tar,spatial}$ is originated from the change of position of $\mathscr{H}_i$ with respect to $\ddot{\pi}_{i,j}$, located on the border of $\mathscr{H}_i^+$ (see Fig.8 for an example of $\ddot{\pi}_{i,1}$). $\mathring{s}_{i,j}^{tar,angular}$ is originated from the change of normal, located on the surface of $\mathscr{H}_i^+$ (see Fig.9 for an example of $\ddot{\pi}_{i,4}$, but to be noted that perturbation of $\ddot{\pi}_{i,4}$ leads to the change of position and also the change of normal).

Concerning the change of position, the sources $\mathring{s}_{i,j}^{tar,spatial}$ are located on the border of the heliostat (see formal developments in Appendix.A and Appendix.B). If we note the sum of $\mathring{s}_{i,j}^{tar,spatial}$ captured on $\mathscr{R}$ to be $S_{i,j}^{tar,spatial}$, it will be 0 when $\mathring{s}_{i,j}^{tar,spatial}$ are all captured or all missed, see Fig.8. However, when the spillage effect (Fig.8c), forward-blocking effect (Fig.8d) or backward-shadowing effect (Fig.8e) occurs, the perturbation of $\ddot{\pi}_{i,j}$ will perturb these effects and $\mathring{s}_{i,j}^{tar,spatial}$ will be partially captured. Therefore, $S_{i,j}^{tar,spatial}$ will be non-null .

Concerning the change of normal, the sources $\mathring{s}_{i,j}^{tar,angular}$ are located on the surface of the heliostat (see details in Appendix.A and Appendix.B). If we note the sum of $\mathring{s}_{i,j}^{tar,angular}$ captured on $\mathscr{R}$ to be $S_{i,j}^{tar,angular}$, it will be non-null when $\mathring{s}_{i,j}^{tar,angular}$ is all captured on $\mathscr{R}$ since the perturbation of $\ddot{\pi}_{i,j}$ perturbs the cosine effect, and it will be 0 when all is missing, see Fig.9. Moreover, when the spillage effect (Fig.9c), forward-blocking effect (Fig.9d) or backward-shadowing effect (Fig.9e) occurs, the perturbation of $\ddot{\pi}_{i,j}$ will perturb these effects and $\mathring{s}_{i,j}^{tar,angular}$ will be partially captured.

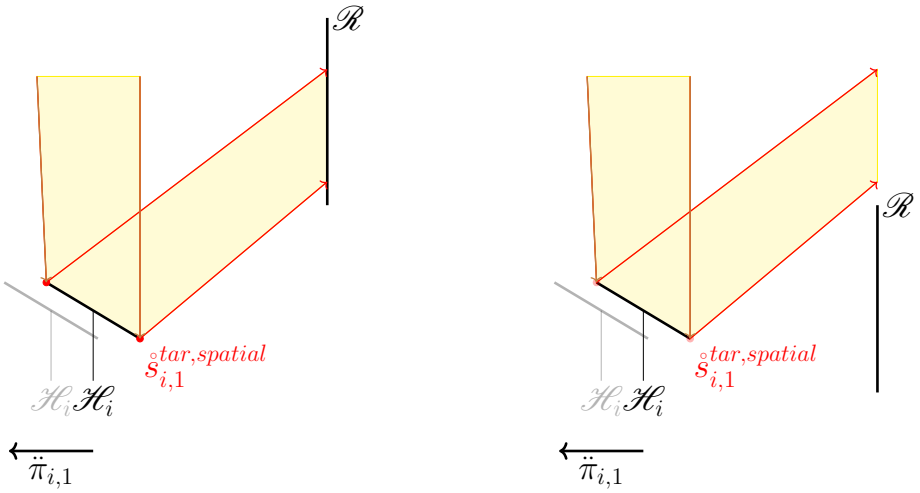
We must distinguish the contributions from backward and forward shadowing in the model of $s_{i,j}$. Backward-shadowing means that the perturbation of $\ddot{\pi}_{i,j}$ affects the shadows on the heliostat itself Fig.10a. Forward-shadowing means that the perturbation of $\ddot{\pi}_{i,j}$ affects the shadows on a neighboring heliostat Fig.11.

$\mathring{s}_{i,j}^{shad-b}$ is named the source of backward-shadowing. This perturbation is observed on the surface of $\mathscr{H}_i^+$, but originated from the border of a neighboring heliostat, see Fig.10a and 10b. It is also separated by two parts, corresponding to the change of position and direction of $\mathscr{H}_i$:

$$\mathring{s}_{i,j}^{shad-b} = \mathring{s}_{i,j}^{shad-b,spatial} + \mathring{s}_{i,j}^{shad-b,angular} \tag{26}$$

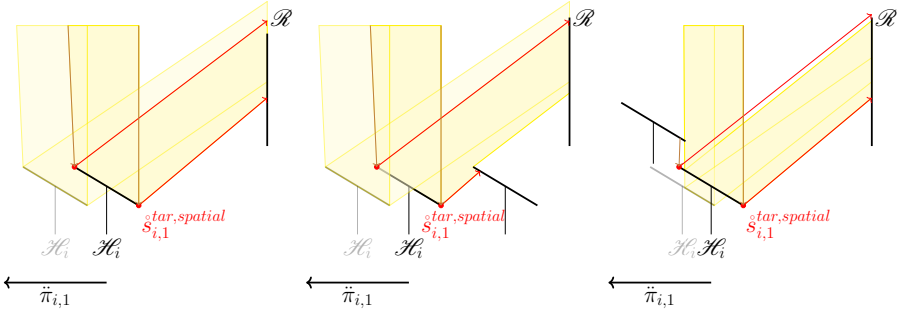
We note $S_{i,j}^{shad-b,spatial}$ as the sum of the source $\mathring{s}_{i,j}^{shad-b,spatial}$ captured on $\mathscr{R}$. When backward-shadowing occurs, the perturbation of $\ddot{\pi}_{i,j}$ will perturb the effect of backward-shadowing by the change of position (see an example of $\ddot{\pi}_{i,1}$ in Fig.10a) and $S_{i,j}^{shad-b,spatial} \neq 0$.

We note $S_{i,j}^{shad-b,angular}$ as the sum of the source $\mathring{s}_{i,j}^{shad-b,angular}$ captured on $\mathscr{R}$. When backward-shadowing occurs, the perturbation of $\ddot{\pi}_{i,j}$ will perturb



(a) All the sources are captured.

(b) All the sources are missing

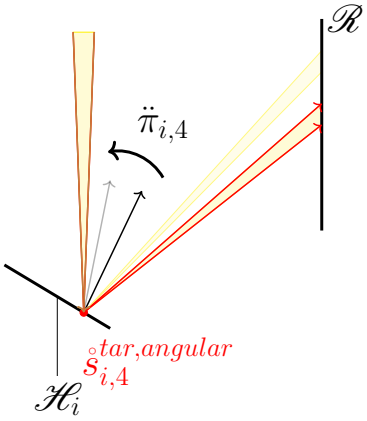


(c) Sources are partially captured: Spillage effect.

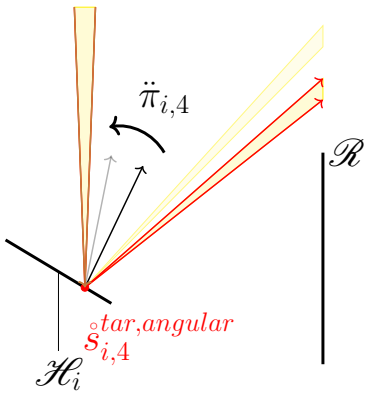
(d) Sources are partially captured: Forward-blocking effect.

(e) Sources are partially captured: Backward-shadowing effect.

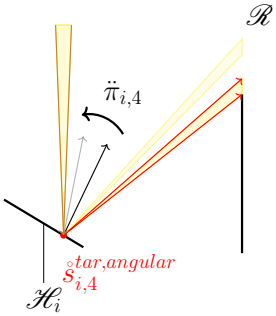
Figure 8: The optical path of I is in yellow, and that of $s_{i,j}$ is in red. When all the sources of $\mathring{s}^{tar,spatial}_{i,1}$ are all captured by $\mathscr{R}$ or they are all missing, the perturbation of $\ddot{\pi}_{i,1}$ is not causing the perturbation of P and $S^{tar,spatial}_{i,1} = 0$. When spillage effect, forward-blocking effect, or backward-shadowing effect occurs, $\mathring{s}^{tar,spatial}_{i,1}$ is partially captured by $\mathscr{R}$ and $S^{tar,spatial}_{i,1} \neq 0$.



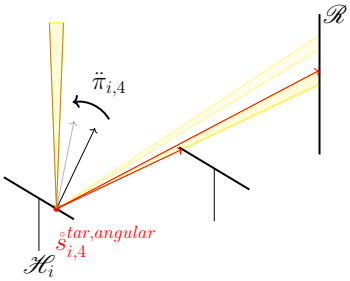
(a) All the sources are captured



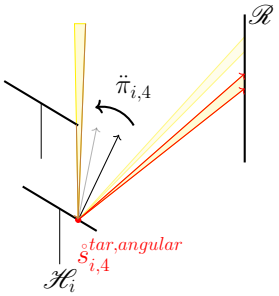
(b) All the sources is missed



(c) Sources are partially captured: Spillage effect

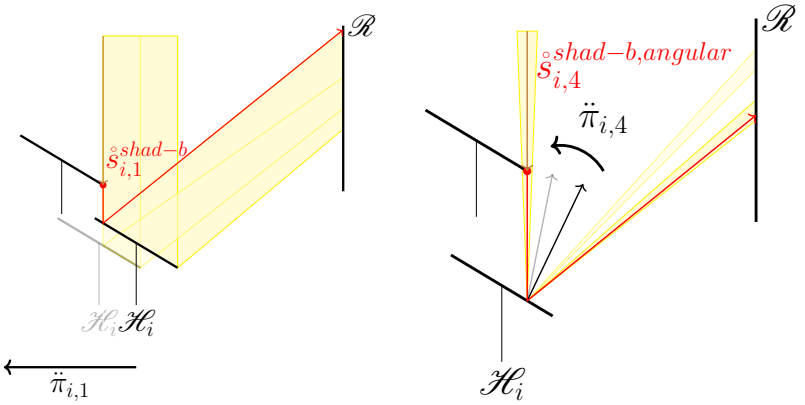


(d) Sources are partially captured: Forward-blocking effect



(e) Sources are partially captured: Backward-shadowing effect

Figure 9: With a perturbation of $\ddot{\pi}_{i,4}$, the perturbation of I is observed on all surfaces of $\mathscr{H}_i$ and borders of Ω_s (the solar cone). When all the sources of $\mathring{s}^{tar,angular}_{i,1}$ is all captured on $\mathscr{R}_{i,1}$, the perturbation of $\ddot{\pi}$ causes the perturbation of P because of the cosine effect and $S^{tar,angular}_{i,1} \neq 0$. When all the sources are missing, $S^{tar,angular}_{i,1} = 0$. Finally, when the spillage effect, forward-blocking effect or backward-shadowing effect occurs, $\mathring{s}^{tar,angular}_{i,1}$ is partially captured by $\mathscr{R}$ and $S^{tar,angular}_{i,1} \neq 0$.



(a) Perturbation due to the change of position

(b) Perturbation due to the change of normal

Figure 10: When the backward shadowing effect occurs, with a perturbation of $\ddot{\pi}_{i,4}$, P is perturbed because of its change of normal. The source of sensitivity $\mathring{s}^{shad-b}_{i,4}$ is on the border of the neighboring heliostat. P is also perturbed because of its change of position. The source of sensitivity $\mathring{s}^{shad-b}_{i,1}$ is on the border of the neighboring heliostat. The backward-shadowing is about the shadowing effect created by the neighboring heliostat.

the effect of backward-shadowing by the change of normal (see an example of $\ddot{\pi}_{i,4}$ in Fig.10b) and $S^{shad-b,spatial}_{i,j} \neq 0$.

$\mathring{s}^{shad-f}_{i,j}$ is named the source of forward-shadowing. This perturbation of I is caused by the creation of shadows. With a perturbation of $\ddot{\pi}_{i,j}$, shadows might be created on another heliostat in the field.

We note $S^{shad-f}_{i,j}$ as the sum of the source $\mathring{s}^{shad-f}_{i,j}$ captured on $\mathscr{R}$. When forward-shadowing occurs, a perturbation of $\ddot{\pi}_{i,j}$ will perturb the effect of forward-shadowing (see an example of $\ddot{\pi}_{i,1}$ in Fig.11) and $S^{shad-f}_{i,j} \neq 0$.

$\mathring{s}^{blo}_{i,j}$ is named the source of backward-blocking. This perturbation of I is caused by blocking. With a perturbation of $\ddot{\pi}_{i,j}$, $\mathscr{H}_i$ might block the I reflected on another heliostat in the field.

We note $S^{blo}_{i,j}$ as the sum of the source $\mathring{s}^{blo}_{i,j}$ captured on $\mathscr{R}$. When backward-blocking occurs, a perturbation of $\ddot{\pi}_{i,j}$ will perturb the effect of backward-blocking (see an example of $\ddot{\pi}_{i,1}$ in Fig.12) and $S^{blo}_{i,j} \neq 0$. Finally, the geometric sensitivity of power $S_{i,j}$ is the sum of all captured sources:

$$S_{i,j} = S^{tar}_{i,j} + S^{blo}_{i,j} + S^{shad-b}_{i,j} + S^{shad-f}_{i,j} \tag{27}$$

with

$$S^{tar}_{i,j} = S^{tar,spatial}_{i,j} + S^{tar,angular}_{i,j} \tag{28}$$

and

$$S^{shad-b}_{i,j} = S^{shad-b,spatial}_{i,j} + S^{shad-b,angular}_{i,j}. \tag{29}$$

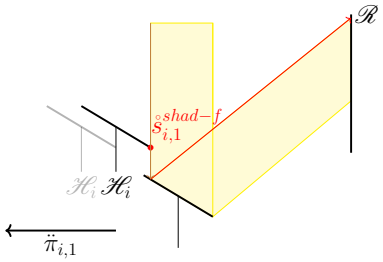


Figure 11: When the forward-shadowing effect occurs, with a perturbation of $\ddot{\pi}_{i,1}$, a perturbation of P is observed. The source of sensitivity $\mathring{s}_{i,1}^{shad-f}$ is on the border of the heliostat itself. The forward-shadowing is about the shadowing effect created by the heliostat itself.

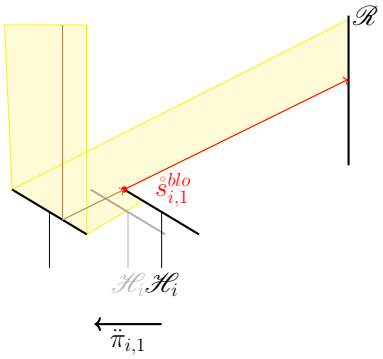


Figure 12: When the backward-blocking effect occurs, a perturbation of $\ddot{\pi}_{i,1}$ causes the perturbation of P. This is because $\ddot{\pi}$ characterizes how much intensity reflected is blocked by $\mathscr{H}_i$. The source of sensitivity $\mathring{s}_{i,j}^{blo}$ is on the border of $\mathscr{H}_i^+$.

Each contribution in Eq.27 is the sum of the sources $\mathring{s}_{i,j}$ captured by the receiver respectively:

$$\begin{bmatrix} S_{i,j}^{tar}(\ddot{\boldsymbol{\pi}}) \\ S_{i,j}^{blo}(\ddot{\boldsymbol{\pi}}) \\ S_{i,j}^{shad-b}(\ddot{\boldsymbol{\pi}}) \\ S_{i,j}^{shad-f}(\ddot{\boldsymbol{\pi}}) \end{bmatrix} = \int_{\mathscr{R}} d\vec{x}_r \int_{2\pi} |\vec{\omega}_r \cdot \vec{n}_r| d\vec{\omega}_r \begin{bmatrix} \mathring{s}_{i,j}^{tar}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \\ \mathring{s}_{i,j}^{blo}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \\ \mathring{s}_{i,j}^{shad-b}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \\ \mathring{s}_{i,j}^{shad-f}(\vec{x}_r, \vec{\omega}_r, \ddot{\boldsymbol{\pi}}) \end{bmatrix}. \quad (30)$$

3 Algorithms

In the previous section, general models for the intensity I and its derivatives $s_{i,j}$ are built for an SPT system. Based on the models, we aim to build algorithms

to estimate the impacting power P and the matrix $\boldsymbol{S}$ by the Monte-Carlo ray tracing method.

Concerning the model of intensity I, the only source is the intensity from the sun $\mathring{I}$ located on the surface that delimits the sky, $\mathscr{O}_s$. P will be estimated by summing the source $\mathring{I}$ captured on the receiver $\mathscr{R}$(see Eq.4).

Concerning the model of $s_{i,j}$, there are several sources of sensitivities on the heliostat $\mathscr{H}_i$. $S_{i,j}$ will be estimated by summing all the sources $\mathring{s}_{i,j}$ captured on the receiver $\mathscr{R}$(see Eq.5).

The corresponding statistical ray tracing process can start from the receiver looking for the sources or, inversely, from the sources located on the heliostat, looking for the receiver. The strategy of proceeding statistical ray tracing process has enormous influences on the convergence rate.

In order to estimate the impacting power P, unlike the Collision-Based Monte-Carlo algorithms (CBMC) where the ray-tracing starts from a plane above reflectors ($\mathscr{O}_s$ in our case)[7], we integrate this work in the framework of IFMC, starting the ray-tracing process from the reflecting surfaces ($\mathscr{H}^+$ in our case), which improves convergence rate[14, 24].

In order to estimate the matrix of sensitivity $\boldsymbol{S}$, we start the statistical ray-tracing process by the sources, looking for the receiver $\mathscr{R}$.

3.1 Estimation of impacting power

The algorithm to compute the impacting power by IFMC has been thoroughly described by [24, 39]. Here is a quick reminder of the integral formulation for estimating P under the assumptions made in Section.2.2:

$$P = \int_{\mathscr{H}^+} p_X(\vec{x}_{p1}) d\vec{x}_{p1} \int_{\Omega_s} p_{\Omega_s}(\vec{\omega}_s) d\vec{\omega}_s \{\mathcal{H}(\vec{x}_{p0} \in \mathscr{O}_s)\mathcal{H}(\vec{x}_{p2} \in \mathscr{R})\hat{w}\} \quad (31)$$

with

$$p_X(\vec{x}_{p1}) = \frac{1}{\mathcal{S}_{\mathscr{H}^+}}, \quad (32)$$

$$p_{\Omega_s}(\vec{\omega}_s) = \frac{1}{\int_C d\vec{\omega}_s} = \frac{1}{2\pi(1-\cos\theta_{disk})} \quad (33)$$

$$\hat{w} = DNI\rho|\vec{\omega}_s \cdot \vec{n}_i|\mathcal{S}_{\mathscr{H}^+} \quad (34)$$

where $\mathcal{S}_{\mathscr{H}^+}$ is the area of all reflected surfaces of the heliostats field.

Equation 31 is the integral formulation of the model to compute the impacting power P and can also be seen as an expected value which is the core of an IFMC algorithm. The expected value P can be estimated by the mean operator (Algo.2 in Appendix.E):

$$P \approx \frac{1}{N}\sum_{q=1}^{N}[\mathcal{H}(\vec{x}_{p0,q} \in \mathscr{O}_s)\mathcal{H}(\vec{x}_{p2,q} \in \mathscr{R})\hat{w}_q] \quad (35)$$

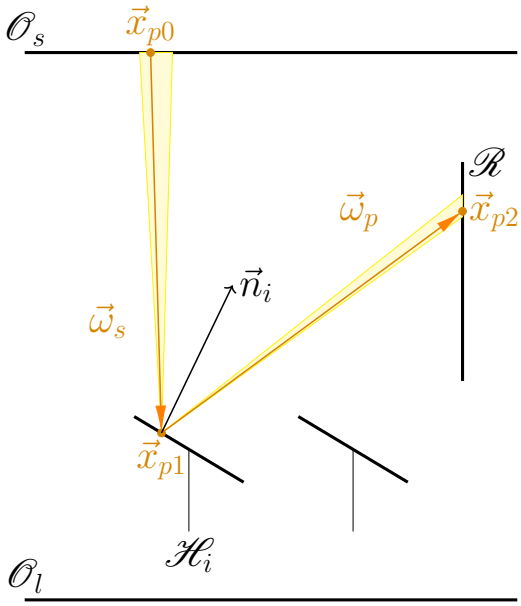


Figure 13: In order to estimate P, for each realization, the algorithm starts by uniformly sampling a random position $\vec{x}_{p1}$ (following Eq.32) on the reflected surface of the whole heliostat field $\mathscr{H}^+$. Then, a direction is sampled within the solar cone Ω_s of angular radius θ_{disk} following the density of probability p_{Ω_s} (Eq.33). In order to check if the shadowing effect occurs, $\vec{x}_{p0}$ is defined as the first intersection with a boundary of the system, starting at $\vec{x}_{p1}$ in the direction $-\vec{\omega}_s$, and it is tested if $\vec{x}_{p0}$ belongs to $\mathscr{O}_s$. If there is no shadowing, the reflected direction $\vec{\omega}_p$ is computed based on the law of reflection for specular surfaces from the sampled $\vec{\omega}_s$ and the local normal $\vec{n}_i$ at position $\vec{x}_{p1}$. In order to check if the spillage or the blocking effect occurs, $\vec{x}_{p2}$ is defined as the first intersection with a boundary of the system, starting at $\vec{x}_{p1}$ in the direction $\vec{\omega}_p$, and it is tested if $\vec{x}_{p2}$ belongs to $\mathscr{R}$. Finally, if no shadowing nor blocking occurs, the weight of Monte-Carlo for this realization $\hat{w}$ is computed.

where $\vec{x}_{p0,q}$ and $\vec{x}_{p2,q}$ are obtained through the described ray-tracing algorithm on Fig.13 for the q^{th} realization. $\hat{w}_q$ is the value of $\hat{w}$ for the q^{th} realization.

The standard deviation $\sigma(P)$ is computed simultaneously with P [40].

3.2 Estimation of the geometric sensitivities of power

The whole matrix of sensitivity of impacting power $\boldsymbol{S}$ has a dimension of $n_{\mathscr{H}} \times n_{\vec{\pi}}$, where $n_{\mathscr{H}}$ is the number of heliostats and $n_{\vec{\pi}} = 6$ is the number of geometric parameters of each heliostat, referring to Fig.4. When the number of heliostats $n_{\mathscr{H}}$ becomes large, it will be very costly in computing time if we estimate them one by one.

Herein, we will build an efficient algorithm to estimate $\boldsymbol{S}$. The matrix of $\boldsymbol{S}$ is composed of the vectors of sensitivities for each heliostat $\boldsymbol{S} \equiv \left[\vec{S}_i\right]$ (reminding the index i for i^{th} Heliostat in the field):

$$\vec{S}_i(\ddot{\boldsymbol{\pi}}) \equiv \begin{bmatrix} S_{i,1}(\ddot{\boldsymbol{\pi}}) \\ S_{i,2}(\ddot{\boldsymbol{\pi}}) \\ S_{i,3}(\ddot{\boldsymbol{\pi}}) \\ S_{i,4}(\ddot{\boldsymbol{\pi}}) \\ S_{i,5}(\ddot{\boldsymbol{\pi}}) \\ S_{i,6}(\ddot{\boldsymbol{\pi}}) \end{bmatrix}^T \tag{36}$$

Following Eq.27, each $S_{i,j}$ has 4 contributions. We defined the vectors of contributions as well as the vectors of sources similarly, and finally, we have:

$$\vec{S}_i = \vec{S}_i^{tar} + \vec{S}_i^{blo} + \vec{S}_i^{shad-b} + \vec{S}_i^{shad-f} \tag{37}$$

and

$$\vec{\mathring{s}}_i = \vec{\mathring{s}}_i^{tar} + \vec{\mathring{s}}_i^{blo} + \vec{\mathring{s}}_i^{shad-b} + \vec{\mathring{s}}_i^{shad-f}. \tag{38}$$

The contributions will then be the sum of captured sources respectively on the $\mathscr{R}$. The corresponding integral formulations are detailed in Appendix.B. The algorithm to estimate $\boldsymbol{S}$ and its standard deviation $\sigma(\boldsymbol{S})$ is Algo.1.

Algorithm 1 Estimate $\boldsymbol{S}$ and $\sigma(\mathbf{S})$

Input: Geometries of heliostats and receiver, $\vec{\omega}_c$
▷ $\vec{\omega}_c$ characterizes the solar position (Eq.15)

Initialization: $i \leftarrow 0$
while $i < n_{\mathscr{H}}$ **do** ▷ $n_{\mathscr{H}}$ number of heliostats
Estimate $\vec{S}_i^{tar}$ and $\vec{\sigma}(\vec{S}_i^{tar})$ by Algo.3
Estimate $\vec{S}_i^{blo}$ and $\vec{\sigma}(\vec{S}_i^{blo})$ by Algo.4
Estimate $\vec{S}_i^{shad-b}$ and $\vec{\sigma}(\vec{S}_i^{shad-b})$ by Algo.6
Estimate $\vec{S}_i^{shad-f}$ and $\vec{\sigma}(\vec{S}_i^{shad-f})$ by Algo.5
$\vec{S}_i \leftarrow \vec{S}_i^{tar} + \vec{S}_i^{blo} + \vec{S}_i^{shad-b} + \vec{S}_i^{shad-f}$
$\vec{\sigma}(\vec{S}_i) \leftarrow \vec{\sigma}(\vec{S}_i^{mat}) + \vec{\sigma}(\vec{S}_i^{blo}) + \vec{\sigma}(\vec{S}_i^{shad-b}) + \vec{\sigma}(\vec{S}_i^{shad-f})$
$i \leftarrow i + 1$
end while
Output: $\boldsymbol{S} \leftarrow \left[\vec{S}_i\right]$, $\boldsymbol{\sigma}(\boldsymbol{S}) \leftarrow \left[\vec{\sigma}(\vec{S}_i)\right]$

In the following part of this section, we will discuss the estimation of each vector of contribution in Eq.37.

3.2.1 Contribution of the effect of targeting $\vec{S}_i^{tar}$

$S_{i,j}^{tar}$ is a component of $\vec{S}_i^{tar}$, estimated by the sum of the sources of targeting $\mathring{s}_{i,j}^{tar}(\ddot{\boldsymbol{\pi}})$ arriving to the receiver. The corresponding integral formulation is yielded in Appendix.B and is separated into two parts, corresponding to the change of position (Fig.8) and the change of normal (Fig.9), noted $\vec{S}_i^{tar,spatial}$ and $\vec{S}_i^{tar,angular}$ respectively. The corresponding sources of sensitivities are separated by $\mathring{s}_{i,j}^{tar,spatial}$ and $\mathring{s}_{i,j}^{tar,angular}$ in Eq.25. Those sources correspond to the spatial gradient of intensity I and the angular gradient of I [33]. It is noted that the $\ddot{\pi}_{i,4}$ and the $\ddot{\pi}_{i,5}$ characterize the change of normal and also the

change of positions (because they are related to the rotations). In contrast, other parameters characterize only the change of position.

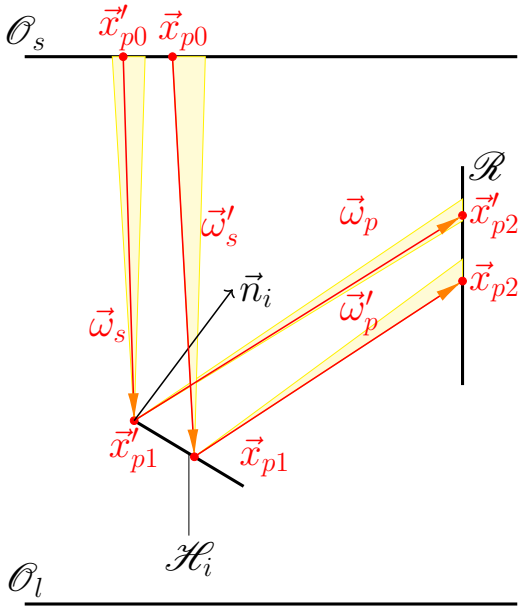


Figure 14: Algorithm to compute $\vec{S}_i^{tar}$. $\vec{x}'_{p1}$ is on the 4 borders of the heliostat $\partial\mathscr{H}_i$ having k as the index of the border, $\vec{x}'_{p0}$ the first intersection starting by $\vec{x}'_{p1}$ following -$\vec{\omega}_s$, $\vec{x}'_{p2}$ the first intersection starting by $\vec{x}'_{p1}$ following $\vec{\omega}_p$, $\vec{\omega}'_s$ the direction on the border of solar cone $\partial\Omega_s$, $\vec{\omega}'_p$ the direction of reflection of $\vec{\omega}'_s$, $\vec{x}_{p0}$ the first intersection starting by $\vec{x}_{p1}$ following -$\vec{\omega}'_s$ and $\vec{x}_{p2}$ the first intersection starting by $\vec{x}_{p1}$ following $\vec{\omega}'_p$. For each realization of estimating $\vec{S}_i^{tar,spatial}$, we sample $\vec{x}'_{p1}$ and $\vec{\omega}_s$. If $\vec{x}'_{p0}$ hits $\mathscr{O}_s$ and $\vec{x}'_{p2}$ hits $\mathscr{R}$ we count $\vec{w}_1$ (Eq.42). For each realization of estimating $\vec{S}_i^{tar,angular}$, we sample $\vec{x}_{p1}$ and $\vec{\omega}'_s$. If $\vec{x}_{p0}$ hits $\mathscr{O}_s$ and $\vec{x}'_{p2}$ hits $\mathscr{R}$ we count $\vec{w}_2$ (Eq.43).

The integral formulation for estimating $\vec{S}_i^{tar}$ is detailed in Eq. 39 and Eq. 40. Corresponding probability density functions are given in Eq. 41. In Eq. 40, k refers to the four sides of the squared heliostat.

$$\vec{S}_i^{tar}(\ddot{\boldsymbol{\pi}}) = \vec{S}_i^{tar,spatial}(\ddot{\boldsymbol{\pi}}) + \vec{S}_i^{tar,angular}(\ddot{\boldsymbol{\pi}}) \tag{39}$$

$$\begin{cases} \vec{S}_i^{tar,spatial}(\ddot{\boldsymbol{\pi}}) = \sum_{k=1}^{4} \int_{l_k} p_{X'}(\vec{x}'_{p1}) d\vec{x}'_p \int_{\Omega_s} p_{\Omega_s}(\vec{\omega}_s) d\vec{\omega}_s \\ \qquad\qquad \{\vec{w}_1 \mathcal{H}(\vec{x}'_{p0} \in \mathscr{O}_s) \mathcal{H}(\vec{x}'_{p2} \in \mathscr{R})\} \\ \vec{S}_i^{tar,angular}(\ddot{\boldsymbol{\pi}}) = \int_{\mathscr{H}_i^+} p_{X_i}(\vec{x}_{p1}) d\vec{x}_p \int_{\partial\Omega_s} p_{\Omega'_s}(\vec{\omega}'_s) d\vec{\omega}'_s \\ \qquad\qquad \{\vec{w}_2 \mathcal{H}(\vec{x}_{p0} \in \mathscr{O}_s) \mathcal{H}(\vec{x}_{p2} \in \mathscr{R})\} \end{cases} \tag{40}$$

where

$$p_{X_i}(\vec{x}_{p1}) = \frac{1}{\mathcal{S}_{\mathscr{H}_i^+}}; p_{X'}(\vec{x}'_{p1}) = \frac{1}{l_k}; p_{\Omega'_s}(\vec{\omega}'_s) = \frac{1}{2\pi} \tag{41}$$

$$\vec{w}_1 = \frac{\vec{\mathring{s}}_i^{tar,spatial}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})}{p_{X'}(\vec{x}'_{p1}) p_{\Omega_s}(\vec{\omega}_s)}; \tag{42}$$

$$\vec{w}_2 = \frac{\vec{\mathring{s}}_i^{tar,angular}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})}{p_{X_i}(\vec{x}_{p1}) p_{\Omega'_s}(\vec{\omega}'_s)}. \tag{43}$$

The notation and the Monte-Carlo ray-tracing algorithm are shown in Fig.14. See also Algo.3 for the complete algorithm and Appendix.C for the explicit expression of $\vec{w}_1$ and $\vec{w}_2$.

3.2.2 Contribution of backward-blocking effect $\vec{S}_i^{blo}$

$S_{i,j}^{blo}$ is a component of $\vec{S}_i^{blo}$, estimated by the sum of the sources of backward-blocking $\mathring{s}_{i,j}^{blo}(\ddot{\boldsymbol{\pi}})$ arriving to the receiver. The corresponding integral formulation is yielded in Appendix.B.

Unlike the source of targeting, the evolution of $\ddot{\pi}_{i,j}$ has no effects on the change of normal of the reflected intensity (Fig.15). Intensity is reflected following $\vec{n}_{i'}$ on $\mathscr{H}_{i'}^+$, while $\ddot{\pi}_{i,j}$ characterizes the heliostat $\mathscr{H}_i$. Therefore, $\mathring{s}_{i,j}^{blo}$ is only about the change of position.

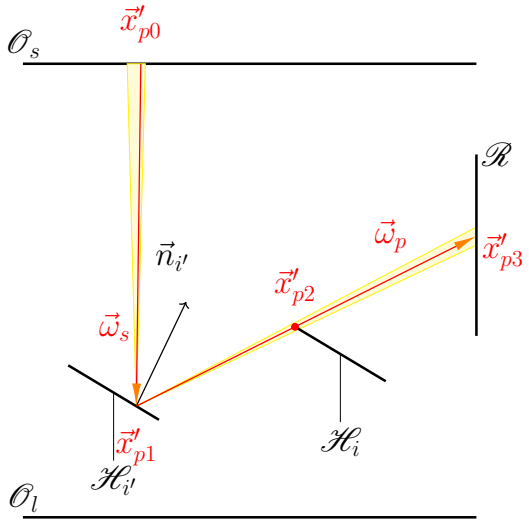


Figure 15: Algorithm to compute $\vec{S}_i^{blo}$. $\vec{x}'_{p1}$ is the first intersection starting by $\vec{x}'_{p2}$, following -$\vec{\omega}_p$, $\vec{x}'_{p0}$ the first intersection starting by $\vec{x}'_{p1}$ following -$\vec{\omega}_s$, $\vec{x}'_{p3}$ the first intersection starting by $\vec{x}'_{p1}$ following $\vec{\omega}_p$. For each realization of estimating $\vec{S}_i^{blo}$, we sample $\vec{x}'_{p2}$ and $\vec{\omega}_s$, and we calculate $\vec{\omega}_p$. If $\vec{x}'_{p0}$ hits $\mathscr{O}_s$, $\vec{x}'_{p1}$ hits $\mathscr{H}_{i'}$ and $\vec{x}'_{p2}$ hits $\mathscr{R}$ we count $\vec{w}_3$.

Seeing Fig.15 for the ray-tracing algorithm, the integral formulation for estimating $\vec{S}_i^{blo}$ is as following:

$$\vec{S}_i^{blo}(\ddot{\boldsymbol{\pi}}) = \sum_{i'=1}^{n_{\mathscr{H}}, i' \neq i} \sum_{k=1}^{4} \int_{l_k} p_{X'}(\vec{x}'_{p2}) d\vec{x}'_{p2} \int_{\Omega_s} p_{\Omega_s}(\vec{\omega}_s) d\vec{\omega}_s \left\{ \vec{w}_3 \mathcal{H}(\vec{x}'_{p0} \in \mathscr{O}_s) \mathcal{H}(\vec{x}'_{p1} \in \mathscr{H}_{i'}^{+}) \mathcal{H}(\vec{x}'_{p3} \in \mathscr{R}) \right\} \quad (44)$$

where

$$\vec{w}_3 = \frac{\vec{\mathring{s}}^{blo}(\vec{x}'_{p2}, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})}{p_{X'(\vec{x}'_{p2})} p_{\Omega_s}(\vec{\omega}_s)} \quad (45)$$

with $n_{\mathscr{H}}$ the number of heliostats in the field.

Moreover, the blocked heliostat $\mathscr{H}_{i'}$ can be all other heliostats except $\mathscr{H}_i$ in the heliostats field. In order to capture all the sources $\mathring{s}_{i,j}^{blo}(\ddot{\pi}_{i,j})$, we need to proceed ray-tracing test for all other heliostats except $\mathscr{H}_i$ in the field to detect the blocking effect. However, most of them will not be blocked by $\mathscr{H}_i$ since the distance between the two heliostats is large. In order to accelerate and optimize the algorithm, we will proceed with the ray-tracing tests on the heliostats, of which the distance to $\mathscr{H}_i$ is within a limited length. This length is defined as l_d. See Algo.4 for the complete algorithm and Appendix.C for the explicit expression of $\vec{w}_3$.

3.2.3 Contribution of forward-shadowing $\vec{S}_{i,j}^{shad-f}$

$S_{i,j}^{shad-f}$ is a component of $\vec{S}_{i,j}^{shad-f}$, estimated by the sum of the sources of forward-shadowing $\mathring{s}_{i,j}^{shad-f}(\ddot{\boldsymbol{\pi}})$ arriving to the receiver. The corresponding integral formulation is yielded in Appendix.B.

Seeing the ray-tracing algorithms on Fig.16, $\vec{S}_i^{shad-f}$ can be estimated by the following integral formulation:

$$\vec{S}_i^{shad-f}(\ddot{\boldsymbol{\pi}}) = \sum_{k=1}^{4} \int_{l_k} p_{X'}(\vec{x}'_{p1}) d\vec{x}'_{p1} \int_{\Omega_s} p_{\Omega_s}(\vec{\omega}_s) d\vec{\omega}_s \left\{ \vec{w}_4 \mathcal{H}(\vec{x}'_{p0} \in \mathscr{O}_s) \mathcal{H}(\vec{x}'_{p2} \in \mathscr{H}^{+}) \mathcal{H}(\vec{x}'_{p3} \in \mathscr{R}) \right\}, \quad (46)$$

where

$$\vec{w}_4 = \frac{\vec{\mathring{s}}_i^{shad-f}(\vec{x}'_{p1}, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})}{p_{X'}(\vec{x}'_{p1}) p_{\Omega_s}(\vec{\omega}_s)}. \quad (47)$$

See also Algo.5 for the complete algorithm and Appendix.C for the explicit expression of $\vec{w}_4$.

3.2.4 Contribution of backward-shadowing effect $\vec{S}_i^{shad-b}$

$S_{i,j}^{shad-b}$ is a component of $\vec{S}_i^{shad-b}$, estimated by the sum of the sources of backward-shadowing $\mathring{s}_{i,j}^{shad-b}(\ddot{\boldsymbol{\pi}})$ arriving to the receiver. The corresponding integral formulation is yielded in Appendix.B. Similar to the targeting effect, the evolution of $\ddot{\pi}_{i,j}$ might have two possible impacts on the heliostat $\mathscr{H}_i$: the change

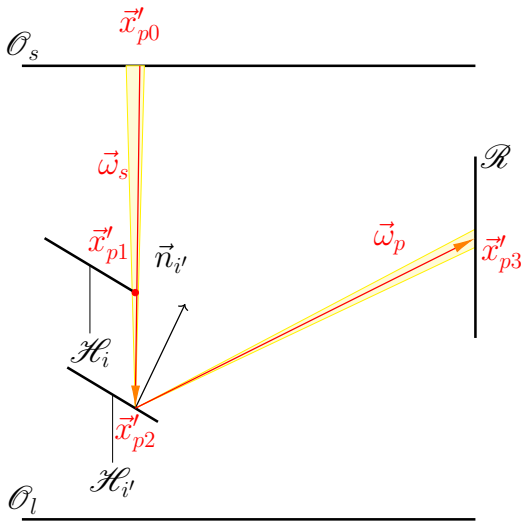


Figure 16: Algorithm to compute $\vec{S}_i^{shad-f}$. $\vec{x}'_{p0}$ is the first intersection starting by $\vec{x}'_{p1}$, following -$\vec{\omega}_s$, $\vec{x}'_{p2}$ the first intersection starting by $\vec{x}'_{p1}$ following $\vec{\omega}_s$, $\vec{x}'_{p3}$ the first intersection starting by $\vec{x}'_{p2}$ following $\vec{\omega}_p$. For each realization of estimating $\vec{S}_i^{blo}$, we sample $\vec{x}'_{p1}$ and $\vec{\omega}_s$, and we calculate $\vec{\omega}_p$. $\vec{\omega}_p$ is reflected following the normal $\vec{n}_{i'}$ which is the normal of $\mathscr{H}_{i'}$. If $\vec{x}'_{p0}$ hits $\mathscr{O}_s$, $\vec{x}'_{p2}$ hits $\mathscr{H}_{i'}^+$ and $\vec{x}'_{p3}$ hits $\mathscr{R}$ we count $\vec{w}_4$.

of position and the change of normal, impacting the backward-shadowing effect. Therefore, the contribution $\vec{S}_i^{shad-b}$ can be divided by two parts: $\vec{S}_i^{shad-b,spatial}$ for the change of position and $\vec{S}_i^{shad-b,angular}$ for the change of normal. Also, the sources $\mathring{s}_{i,j}^{shad-b,spatial}$ are divided by two parts respectively: $\mathring{s}_{i,j}^{shad-b,spatial}$ and $\mathring{s}_{i,j}^{shad-b,angular}$ in Eq.26. They correspond to the spatial and the angular gradient of intensity I on the reflecting surface of the heliostat $\mathscr{H}_i^+$[33].

Seeing Fig.10a and Fig.10b, these two gradients (spatial and angular) are both originated from the borders of another heliostat (from point $\vec{x}'_{p1}$ in Fig.17).

Seeing the algorithm of ray-tracing in Fig17, $\vec{S}_i^{shad-b}$ can be estimated through the following integral formulation:

$$\vec{S}_i^{shad-b}(\ddot{\vec{\pi}}) = \sum_{k=1}^{4\times n_{\mathscr{H}}} \int_{l_k} p_{X'}(\vec{x}'_{p1}) d\vec{x}'_{p1} \int_{\Omega_s} p_{\Omega_s}(\vec{\omega}_s) d\vec{\omega}_s \left\{ (\vec{w}_5 + \vec{w}_6) \mathcal{H}(\vec{x}'_{p0} \in \mathscr{O}_s) \mathcal{H}(\vec{x}'_{p2} \in \mathscr{H}_i^+) \mathcal{H}(\vec{x}'_{p3} \in \mathscr{R}) \right\}, \quad (48)$$

where

$$\vec{w}_5 = \frac{\vec{\mathring{s}}_i^{shad-b,spatial}(\vec{x}'_{p1}, \vec{\omega}_p, \ddot{\vec{\pi}})}{p_{X'}(\vec{x}'_{p1}) p_{\Omega_s}(\vec{\omega}_s)}; \quad (49)$$

$$\vec{w}_6 = \frac{\vec{\mathring{s}}_i^{shad-b,angular}(\vec{x}'_{p1}, \vec{\omega}_p, \ddot{\vec{\pi}})}{p_{X'}(\vec{x}'_{p1}) p_{\Omega_s}(\vec{\omega}_s)}. \quad (50)$$

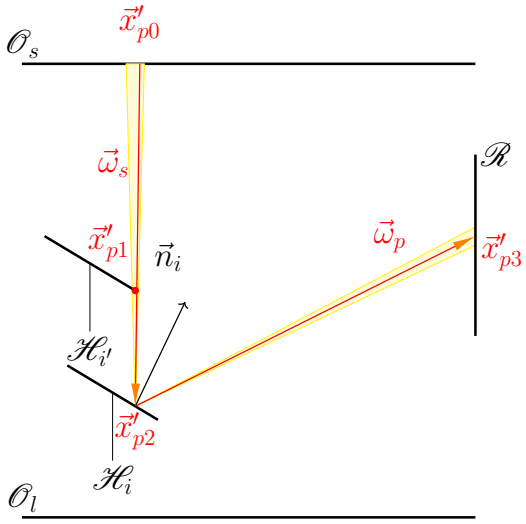


Figure 17: Algorithm to compute $\vec{S}_i^{shad-b}$. $\vec{x}'_{p0}$ is the first intersection starting by $\vec{x}'_{p1}$, following -$\vec{\omega}_s$, $\vec{x}'_{p2}$ the first intersection starting by $\vec{x}'_{p1}$ following $\vec{\omega}_s$, $\vec{x}'_{p3}$ the first intersection starting by $\vec{x}'_{p2}$ following $\vec{\omega}_p$, $\vec{t}_{i',k}$ the vector of circulation on the shadowing heliostat $\mathscr{H}_{i'}$, l_{shad-b} the length of the optical path between $\vec{x}'_{p1}$ and $\vec{x}'_{p2}$. For each realization of estimating $\vec{S}_i^{blo}$, we sample $\vec{x}'_{p1}$ and $\vec{\omega}_s$, and we calculate $\vec{\omega}_p$. $\vec{\omega}_p$ is reflected following the normal $\vec{n}_i$ which is the normal of $\mathscr{H}_i$. If $\vec{x}'_{p0}$ hits $\mathscr{O}_s$, $\vec{x}'_{p2}$ hits $\mathscr{H}_i^+$ and $\vec{x}'_{p3}$ hits $\mathscr{R}$ we count $\vec{w}_5 + \vec{w}_6$.

Similar to the backward-blocking effect, $\mathscr{H}_{i'}$ can be all other heliostats except $\mathscr{H}_i$ itself in the heliostats field. In order to detect all backward-shadowing effects, we need to proceed with ray-tracing tests for all the borders of all heliostats in the field except the borders of $\mathscr{H}_i$ ($4 \times (n_{\mathscr{H}} - 1)$ in total where $n_{\mathscr{H}}$ is the number of heliostats in the field). However, most of them will not create shadows on $\mathscr{H}_i$ since they are too far from each other. In order to accelerate and optimize the algorithm, we will only proceed with the ray-tracing tests for the heliostats, of which the distance to $\mathscr{H}_i$ is within l_d. See also Algo.6 for the complete algorithm and Appendix.C for the explicit expression of $\vec{w}_5$ and $\vec{w}_6$.

4 Results, validations, and comparisons

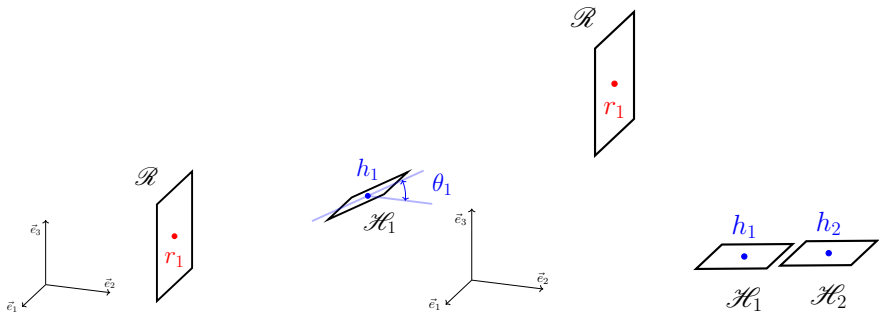
In this section, we will proceed with the following:

1. Validations of our method by finite difference method using four examples.
2. A comparison of the finite difference method with our method in calculating time.

4.1 Validations

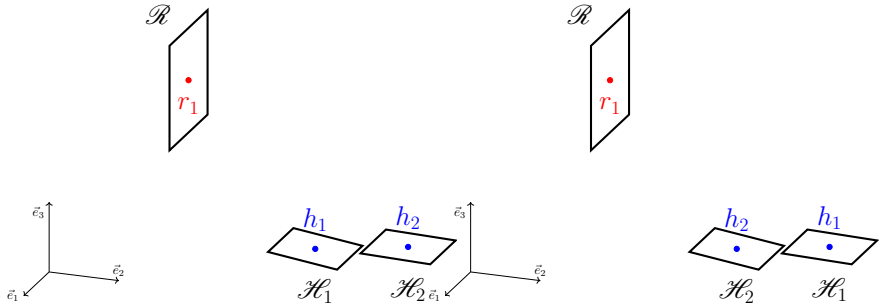
We will here proceed with four simple examples (see Fig.18) where we apply our method and finite difference method for validation purposes. The sensitivities of $\mathscr{H}_1$ are calculated and results are shwon in Table.2, 3, 4 and 5. In case 1, only

the spillage effect is detected. In case 2, the blocking effect is detected. In case 3, the backward-shadowing effect is detected. In case 4, the forward-shadowing effect is detected. The estimation results by our method are validated by the finite difference method, which will be discussed in the following subsection.



(a) Case 1: $r_1 = (0, 0, 0)$, $h_1 = (1, 5, 1)$ and $\theta_1 = 30°$. Solar elevation angle = 60°.

(b) Case 2: $r_1 = (0, 0, 2)$, $h_1 = (0.25, 2, 0)$ and $h_2 = (0, 3, 0)$. Solar elevation angle = 60°.



(c) Case 3: $r_1 = (0, 0, 2)$, $h_1 = (0.25, 2, 0)$ and $h_2 = (0, 3, 0)$. Solar elevation angle = 30°.

(d) Case 4: $r_1 = (0, 0, 2)$, $h_2 = (0.25, 2, 0)$ and $h_1 = (0, 3, 0)$. Solar elevation angle = 30°.

Figure 18: The receiver $\mathscr{R}$ is a rectangular with lengths of 1.50 m and r_1 is its geometric center. The heliostats $\mathscr{H}_1$ and $\mathscr{H}_2$ are two rectangles with lengths of 1 m and h_1, h_2 are their geometric centers respectively. In case 1, θ_1 is the horizontal angle of $\mathscr{H}_1$. In case 2, 3 and 4, the heliostat $\mathscr{H}_1$ and $\mathscr{H}_2$ point to r_1 according to the solar elevation angle.

4.1.1 Approximation by finite difference method

The finite difference method is a standard method to approximate the sensitivities. When P is derivable around $\ddot{\pi}_{i,j}$:

$$S_{i,j} = \lim_{\epsilon \to 0} \frac{P(\ddot{\pi}_{i,j} + \epsilon) - P(\ddot{\pi}_{i,j} - \epsilon)}{2\epsilon}. \tag{51}$$

The finite difference method approximates $S_{i,j}$ by giving a non-zero fixed value to ϵ. We note the result of approximation as $\tilde{S}_{i,j}$:

$$S_{i,j} \approx \frac{P(\ddot{\pi}_{i,j} + \Delta\ddot{\pi}_{i,j}) - P(\ddot{\pi}_{i,j} - \Delta\ddot{\pi}_{i,j})}{2\Delta\ddot{\pi}_{i,j}} \equiv \tilde{S}_{i,j}. \tag{52}$$

unit	W/m	W/m	W/m	W/rad	W/rad	W/m
j	1	2	3	4	5	6
$S_{1,j}^{tar,spatial}$	-4.56	244.82×10^{-3}	-125.12	54.24	-2.02	33.45
$S_{1,j}^{tar,angular}$	0.00	0.00	0.00	1.14×10^{3}	-10.37	0.00
$S_{1,j}^{blo}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-b,spatial}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-b,angular}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-f}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}$	-4.56	244.82×10^{-3}	-125.12	1.20×10^{3}	-12.39	33.45
$\sigma(S_{1,j})$	47.41×10^{-3}	876.00×10^{-6}	331.02×10^{-3}	2.19	382.67×10^{-3}	106.14×10^{-3}
$\tilde{S}_{1,j}$	-4.50	244.00×10^{-3}	-124.91	1.19×10^{3}	-12.27	33.41
$\sigma(\tilde{S}_{1,j})$	235.23×10^{-3}	235.23×10^{-3}	238.22×10^{-3}	2.38	2.35	236.04×10^{-3}

Table 2: Validation - Case 1: 10^6 realizations on each border and 10^8 realizations on the surface for an estimation of $S_{1,j}$ and 10^7 realizations for each estimation of P when approximating $\tilde{S}_{1,j}$ by finite difference method.

unit	W/m	W/m	W/m	W/rad	W/rad	W/m
j	1	2	3	4	5	6
$S_{1,j}^{tar,spatial}$	-793.35	143.71×10^{-3}	141.22×10^{-3}	456.62	356.02×10^{-3}	991.76
$S_{1,j}^{tar,angular}$	0.00	0.00	0.00	-912.74	-463.55	0.00
$S_{1,j}^{blo}$	139.39	-277.29	-415.94	-188.09	14.84	-234.30
$S_{1,j}^{shad-b,spatial}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-b,angular}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-f}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}$	-653.96	-277.15	-415.80	-644.22	-448.36	757.46
$\sigma(S_{1,j})$	270.41×10^{-3}	890.09×10^{-3}	1.03	13.81	2.08	643.34×10^{-3}
$\tilde{S}_{1,j}$	-654.25	-277.20	-415.90	-640.00	-450.50	757.50
$\sigma(\tilde{S}_{1,j})$	180.35×10^{-3}	181.87×10^{-3}	181.38×10^{-3}	18.23	1.82	1.82

Table 3: Validation - Case 2: 10^6 realizations on each border and 10^9 realizations on the surface for an estimation of $S_{1,j}$ and 10^9 realizations for each estimation of P when approximating $\tilde{S}_{1,j}$ by finite difference method.

Practically, the approximation of $\tilde{S}_{i,j}$ requires estimating P twice. Since the impacting power P is estimated by the Monte-Carlo method, its statistical standard deviation $\sigma(P)$ is also estimated at the same time [40]. When approximating $\tilde{S}_{i,j}$, its standard deviation $\sigma(\tilde{S}_{i,j})$ can be obtained [23]:

$$\sigma(\tilde{S}_{i,j}) \approx \frac{\sigma(P(\ddot{\pi}_{i,j} + \Delta\ddot{\pi}_{i,j})) + \sigma(P(\ddot{\pi}_{i,j} - \Delta\ddot{\pi}_{i,j}))}{2\Delta\ddot{\pi}_{i,j}}. \tag{53}$$

The major drawback of the finite difference method is that the related standard deviation $\sigma(\tilde{S}_{i,j})$ is hard to control[23] and tends to infinity as $\Delta\ddot{\pi}_{i,j}$ tends to zero.

4.2 Comparison with the finite difference method

In this section, we will apply and discuss the algorithms previously introduced in the context of a functioning solar tower station: Sierra SunTower [34]. This case study has been chosen since our sensitivity model was developed for flat heliostats.

This solar tower station is located in Mojave Desert, California, at a latitude of 34.7°. It consists of 2 solar towers, each with north and south heliostat subfields[34].

The heliostat field and the two towers are symmetrically distributed. Therefore, we only focus on the field on the southwest side and its related tower,

unit	W/m	W/m	W/m	W/rad	W/rad	W/m
j	1	2	3	4	5	6
$S_{1,j}^{tar,spatial}$	-236.19	-531.19	-530.35	-493.65	35.84	801.24
$S_{1,j}^{tar,angular}$	0.00	0.00	0.00	1.64×10^3	137.89	0.00
$S_{1,j}^{blo}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-b,spatial}$	0.00	155.41	1.18×10^3	-461.94	-19.41	12×10^{-6}
$S_{1,j}^{shad-b,angular}$	0.00	0.00	0.00	-200.32	-1.56	0.00
$S_{1,j}^{shad-f}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}$	-236.19	-375.79	650.09	479.69	152.76	801.24
$\sigma(S_{1,j})$	496.41×10^{-3}	427.22×10^{-3}	1.02	11.88	1.53	393.25×10^{-3}
$\tilde{S}_{1,j}$	-236.00	-376.00	649.50	490.00	150.00	800.50
$\sigma(\tilde{S}_{1,j})$	780.28×10^{-3}	771.28×10^{-3}	770.40×10^{-3}	7.72	7.72	773.85×10^{-3}

Table 4: Validation - Case 3: 10^6 realizations on each border and 10^9 realizations on the surface for an estimation of $S_{1,j}$ and 10^9 realizations for each estimation of P when approximating $\tilde{S}_{1,j}$ by finite difference method.

unit	W/m	W/m	W/m	W/rad	W/rad	W/m
j	1	2	3	4	5	6
$S_{1,j}^{tar,spatial}$	-1.29×10^{-3}	-2.10×10^{-3}	-2.31×10^{-3}	-849.47	242.46×10^{-3}	1.06×10^3
$S_{1,j}^{tar,angular}$	0.00	0.00	0.00	1.69×10^3	-88.94×10^{-3}	0.00
$S_{1,j}^{blo}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-b,spatial}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-b,angular}$	0.00	0.00	0.00	0.00	0.00	0.00
$S_{1,j}^{shad-f}$	-68.35	375.08	-649.65	-318.62	17.21	-232.07
$S_{1,j}$	-68.35	375.08	-649.66	523.12	17.37	823.17
$\sigma(S_{1,j})$	181.08×10^{-3}	220.47×10^{-3}	382.90×10^{-3}	10.45	973.89×10^{-3}	206.71×10^{-3}
$\tilde{S}_{1,j}$	-68.50	375.00	-649.50	530.00	15.00	823.00
$\sigma(\tilde{S}_{1,j})$	771.70×10^{-3}	771.28×10^{-3}	770.40×10^{-3}	7.72	7.72	771.70×10^{-3}

Table 5: Validation - Case 4: 10^6 realizations on each border and 10^9 realizations on the surface for an estimation of $S_{1,j}$ and 10^9 realizations for each estimation of P when approximating $\tilde{S}_{1,j}$ by finite difference method.

which includes 6090 flat heliostats. It is easy to extend the calculation for the whole field in the south (including 12180 heliostats). However, extra efforts are needed for the field on the north side since the towers' shadows will impact the sensitivities.

Table.6 shows the geometric parameters of the radiative system, as well as the solar positions for the summer solstice.

Characteristic	Value
Overall north-south length	95 $[m]$
Overall east-west length	175 $[m]$
Distance between the tower and the first row of heliostats	12.5 $[m]$
Number of heliostats per column	58
Number of heliostats per row	105
Size of heliostats	1.067×1.067 $[m^2]$
Height of the centre of receiver	50 $[m]$
Size of receiver	4×4 $[m^2]$
Zenith angle for the summer solstice at solar noon	79°

Table 6: Geometric configuration of the system[34]

The present work focuses on the optical performance (the impacting power P). It is assumed that:

1. The sun's position is fixed (solar noon for the summer solstice)
2. Incoming Direct Normal Intensity (DNI) is considered homogeneous for the whole heliostat field (DNI = $1000W/m^2$).

The incoming power, $P(\ddot{\boldsymbol{\pi}})$ and its derivatives $\boldsymbol{S}(\ddot{\boldsymbol{\pi}})$ towards geometric parameters $\ddot{\boldsymbol{\pi}} \equiv [\ddot{\pi}_{i,j}]$ are estimated. It is reminded that i refers to the i^{th} heliostat ($i \in \{1, 2 \ldots, 6090\}$) and j to one of the six geometric parameters ($j \in \{1, 2, \ldots, 6\}$) as shown on Fig.4.

Results have already been shown at the beginning of this article (Fig.2). It is noted that the accuracy of the simulation depends on the number of Monte Carlo realizations. Each Monte-Carlo realization consists of the corresponding sampling process and the ray-tracing tests described in Section.3.

When we focus on one parameter of one heliostat in the field (for example, the size of the 5297^{th} heliostat: $\ddot{\pi}_{5297,6}$), it has a tinny impact on the total impacting power P. It causes a poor convergence performance when applying the finite difference method, as shown in Table.7. Compared to our method, the method of approximation by finite difference takes much more time to converge because the choice of $\Delta\ddot{\pi}$ enormously impacts the convergence rate and the precision for the approximation method of finite difference. In contrast, our method of estimation avoids this choice. These calculations are run parallelly in an ordinary laptop[1], and our method converges in 0.188s. According to [40], to reduce ten times the standard deviation of a Monte-Carlo estimation, approximately 100 times the calculation time is needed. In this application case, the approximation of the

[1] 4 CPUs of i5 Intel™, 8th generation

finite difference method will then spend approximately 2×10^{11} more times of calculating time to attend the same order of standard deviation of our method.

Last but not least, it is found that in Table.2, Table.3, Table.4 and Table.5, the advantage of our method in convergence performance is not that significant compared to the finite difference method. We have the following comments:

1. The four cases are dedicated to validation purposes. Therefore, the geometries are relatively simple compared to a functioning SPT system. However, the finite difference method will become impracticable when the geometries become more complex.

2. Even with simple geometries, the standard derivation of the finite difference method ($\sigma(\tilde{S}_{i,j})$) is hard to be controlled. It is strongly dependent on the value of $\Delta\ddot{\pi}_{i,j}$ chosen in Eq.52 and Eq.53.

3. The derivation related to the discretization can not be handled for the finite difference method. It is also largely related to the choice of $\Delta\ddot{\pi}$ in Eq.52 and Eq.53. However, our method can handle the statistical standard derivation. We avoid the derivation related to the discretization and the choice of $\Delta\ddot{\pi}_{i,j}$.

Finite difference method	Value	standard deviation	calculation time
$P(\ddot{\pi}_{5297,6} - \Delta\ddot{\pi}_{5297,6})$	$3231.62[w]$	$0.0389[w]$	$272.5[s]$
$P(\ddot{\pi}_{5297,6} + \Delta\ddot{\pi}_{5297,6})$	$3231.85[w]$	$0.0389[w]$	$272.5[s]$
$\tilde{S}_{5297,6}(\ddot{\pi}_{5297,6})$	$1.15[w/m]$	$0.389[w/m]$	$545[s]$
Our method	Value	standard deviation	calculation time
$S_{5297,6}(\ddot{\pi}_{5297,6})$	$1.14974[w/m]$	$0.000041[w/m]$	$0.188[s]$

Table 7: $\ddot{\pi}_{5297,6}$ is originally equal to 1.067 (see Table.6), while herein, a reasonable value of $\Delta\ddot{\pi}_{5297,6}$ is chosen: $\Delta\ddot{\pi}_{5297,6} = 0.1[m]$. Two estimations of impacting power ($P(\ddot{\pi}_{5297,6} - \Delta\ddot{\pi}_{5297,6})$ and $P(\ddot{\pi}_{5297,6} + \Delta\ddot{\pi}_{5297,6})$) are realized following Algo.2 (10^9 Monte-Carlo realizations) . The sensitivity of power $\tilde{S}_{5297,6}$ is approximated based on these two estimations. Also, the sensitivity of power $S_{5297,6}$ is estimated following Algo.1 (10^4 Monte-Carlo realizations for each border of the heliostats).

5 Conclusion

We have presented general models for the intensity I and the sensitivity of intensity $s_{i,j}$ in a radiative system of an SPT system, as well as the complete algorithms for estimating the matrix of sensitivity of power $\boldsymbol{S}$. These algorithms are validated by the finite difference method using four typical examples. Since the sensitivity of a heliostat of a geometric parameter $S_{i,j}$ is estimated by the accumulation of four contributions: $S_{i,j}^{tar}$, $S_{i,j}^{blo}$, $S_{i,j}^{shad-b}$ and $S_{i,j}^{shad-f}$, the contributions of different physical events to the sensitivity $S_{i,j}$ in an SPT system (blocking, shadowing, spillage) can be analyzed in detail. After that, an actual

functioning SPT system was used as a test-case study. Firstly, a comparison of computing time was made for estimating sensitivity by our method and its approximation by the finite difference method. Secondly, the matrix of sensitivity of power $\boldsymbol{S}$ was calculated. Engineers can benefit from this information on sensitivities to optimize the optical design and the aiming strategy for the SPT system. Last but not least, this work makes the integration of gradient-based optimizations for the optical system in an SPT becomes possible. However, the sensitivity model developed in the present paper is only applied to a field of flat heliostats. Thus, a promising perspective would be to extend this model to curved heliostats.

This work received financial support from the from Region Occitanie and the SOLSTICE laboratory of Excellence (ANR-10-LABX-22-01).

A Boundary conditions of geometric sensitivities for a heliostat

In this work, a heliostat is modeled by two surfaces without thickness: the reflecting surface $\mathscr{H}_i^+$ and the rearward surface $\mathscr{H}_i^-$.

Based on the general model in[33], the boundary condition of geometric sensitivities for a cold, specular mirror with homogeneous reflectivity is developed, where $I(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ and $I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}})$ are spatially and angularly smooth:

$$\begin{aligned} s_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = &- \partial_{1,\vec{v}_{i,j}} I(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \\ &+ \rho \partial_{1,\vec{v}_{i,j}} I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \\ &- \rho \partial_R^{\vec{a}} I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \\ &+ \rho \partial_R^{\vec{a}'} I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \\ &+ \rho s_{i,j}(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \end{aligned} \tag{54}$$

where $\partial_{1,\vec{v}_{i,j}}$ is the operator for spatial gradient following the vector $\vec{v}_{i,j}$ which are the velocity of deformation (yielded in Appendix.D), $\partial_R^{\vec{a}}$ the operator for angular gradient following the vector of rotation $\vec{a}$, and $\vec{a}'$ the reflected vector of $-\vec{a}$ on $\mathscr{H}_i^+$[33]. However, when we model the reflecting surface $\mathscr{H}_i^+$, $I(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ is spatially discontinued because the heliostat has 4 edges and $I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}})$ can be angularly discontinued (when backward-shadowing occurs, see Fig.10b).

In [33], the discontinuity of spatial gradient and angular gradient of intensity I are studied. The boundary condition of $\mathscr{H}_i^+$ can then be yielded. Furthermore, in this work, we classify the different sources of sensitivity based on the background of CSP applications:

$$\begin{aligned} s_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}^{tar}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) + \mathring{s}_{i,j}^{blo}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \\ + \mathring{s}_{i,j}^{shad-b}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) + \rho s_{i,j}(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}). \end{aligned} \tag{55}$$

$\mathring{s}_{i,j}^{tar}$ is separated into the source $\mathring{s}_{i,j}^{tar,spatial}$ on the border of heliostat $\partial\mathscr{H}_i^+$ and the source $\mathring{s}_{i,j}^{tar,angular}$ on the surface $\mathscr{H}_i^+$: $\mathring{s}_{i,j}^{tar} = \mathring{s}_{i,j}^{tar,spatial} + \mathring{s}_{i,j}^{tar,angular}$, where:

$$\mathring{s}_{i,j}^{tar,spatial}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = -(\vec{\omega}_p \times \vec{v}_{i,j}) \cdot \vec{t}_{i,k} \rho I(\vec{x}_p, \vec{\omega}_s, \ddot{\boldsymbol{\pi}}) \tag{56}$$

and

$$\mathring{s}_{i,j}^{tar,angular}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = -\rho\partial_R^{\vec{a}} I(\vec{x}_p,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) + \rho\partial_R^{\vec{a}'} I(\vec{x}_p,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}). \quad (57)$$

$\vec{t}_{i,k}$ are the vectors of circulation of $\mathscr{H}_i^+$, depending on $\vec{x}_p$, see Fig.19.

When the backward-shadowing does not occur, the angular gradient in Eq.57 will be the angular derivative of the boundary condition of $\mathscr{O}_s$ (previously defined in Eq.15, which is the Pill-box sun shape model). The development is straightforward, and they are the sources on the border of the solar cone $\partial\Omega_s$:

$$\partial_R^{\vec{a}} I(\vec{x}_p,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) = I_0(\vec{\omega}_s\times\vec{a})\cdot\vec{\omega}_c\mathcal{H}(hit_b\in\mathscr{O}_s) \quad (58)$$

$$\partial_R^{\vec{a}'} I(\vec{x}_p,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) = I_0(\vec{\omega}_s\times\vec{a}')\cdot\vec{\omega}_c\mathcal{H}(hit_b\in\mathscr{O}_s) \quad (59)$$

where $\mathcal{H}(hit_b\in\mathscr{O}_s)$ corresponds to a backward ray-tracing test which will be true if the direction of $-\vec{\omega}_s$ starting from $\vec{x}_p$ is not shadowed by other heliostats in the field.

$\mathring{s}_{i,j}^{blo}$ is the source on the border of heliostat $\partial\mathscr{H}_i^+$:

$$\mathring{s}_{i,j}^{blo}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = (\vec{\omega}_p\times\vec{v}_{i,j})\cdot\vec{t}_{i,k}I(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \quad (60)$$

When backward-shadowing effect occurs, $\mathscr{H}_i^+$ is shadowed by another heliostat $\mathscr{H}_{i'}^+$, a spatial gradient exists on the surface of $\mathscr{H}_i^+$ (Fig.10a). An angular gradient exists within the solar cone Ω_s (Fig.10b). Furthermore, in our case, these sources on the surface of $\mathscr{H}^+$ and within the cone Ω_s all come from the border of the shadowing heliostat $\partial\mathscr{H}_{i'}^+$.

Following the models of spatial gradient and angular gradient in [33], the source $\mathring{s}_{i,j}^{shad-b}$ is on the borders of all heliostats in the field except the heliostat $\mathscr{H}_i$, which is $\partial\mathscr{H}_{i'}^+, i'\neq i$. Similar to the source of targeting, we separate the source $\mathring{s}_{i,j}^{shad-b}$ by $\mathring{s}_{i,j}^{shad-b,spatial}$ and $\mathring{s}^{shad-b,angular}$:

$$\mathring{s}_{i,j}^{shad-b}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}^{shad-b,spatial}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) + \mathring{s}_{i,j}^{shad-b,angular}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) \quad (61)$$

where

$$\mathring{s}_{i,j}^{shad-b,spatial}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \rho\frac{\vec{v}_{i,j}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k}I(\vec{x}_{p'},\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) \quad (62)$$

and

$$\mathring{s}_{i,j}^{shad-b,angular}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}}) = \\ -\rho l_{shad-b}(\vec{\omega}_s\times\vec{a}\times\vec{\omega}_s)\cdot\vec{t}_{i',k}I(\vec{x}_{p'},\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) \\ +\rho l_{shad-b}(\vec{\omega}_s\times\vec{a}'\times\vec{\omega}_s)\cdot\vec{t}_{i',k}I(\vec{x}_{p'},\vec{\omega}_s,\ddot{\boldsymbol{\pi}}). \quad (63)$$

$\vec{t}_{i',k}$ are the vectors of circulation of $\mathscr{H}_{i'}^{+}$[33], see Fig.19 and l_{shad-b} is the length of optical path between $\vec{x}'_p$ and $\vec{x}_p$ (Fig.17).

The boundary condition of the rearward surface $\mathscr{H}_i^{-}$ can be developed based on the general model of opaque-black surface in [33]:

$$s(\vec{x}_b,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}^{shad-f}(\vec{x}_b,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) \quad (64)$$

and $\mathring{s}_{i,j}^{shad-f}$ is the source on the border of the heliostat $\mathscr{H}_i^{-}$:

$$\mathring{s}_{i,j}^{shad-f}(\vec{x}_b,\vec{\omega}_b,\ddot{\boldsymbol{\pi}}) = \\ (\vec{\omega}_s\times\vec{v}_{i,j})\cdot\vec{t}_{i,k}I(\vec{x}_b,\vec{\omega}_s,\ddot{\boldsymbol{\pi}}) \quad (65)$$

B Integral formulations

The vectors of contributions of $\boldsymbol{S}$ defined in Section.3.2 are the sum of the vectors of sources captured by the receiver $\mathscr{R}$ respectively. $\vec{S}_i^{tar,spatial}(\ddot{\boldsymbol{\pi}})$ is estimated by integrating $\vec{\mathring{s}}_i^{tar,spatial}$ on $\partial\mathscr{H}_i^{+}$ and on Ω_s:

$$\vec{S}_i^{tar,spatial}(\ddot{\boldsymbol{\pi}}) = \oint_{\partial\mathscr{H}_i^{+}} d\vec{x}_p \int_{\Omega_s} d\vec{\omega}_s \\ \left\{\mathscr{H}(hit_f\in\mathscr{R})\vec{\mathring{s}}_i^{tar,spatial}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})\right\} \quad (66)$$

where the Heaviside function $\mathcal{H}(hit_f\in\mathscr{R})$ corresponds to a forward ray-tracing test which will be true if the source arrives at $\mathscr{R}$ (the ray staring at $\vec{x}_p$, following $\vec{\omega}_p$ impacts the receiver $\mathscr{R}$).

$\vec{S}_i^{tar,angular}(\ddot{\boldsymbol{\pi}})$ is estimated by integrating $\vec{\mathring{s}}_i^{tar,angular}$ on $\mathscr{H}_i^{+}$ and on $\partial\Omega_s$:

$$\vec{S}_i^{tar,angular}(\ddot{\boldsymbol{\pi}}) = \int_{\mathscr{H}_i^{+}} d\vec{x}_p \oint_{\partial\Omega_s} d\vec{\omega}_s \\ \left\{\mathscr{H}(hit_f\in\mathscr{R})\vec{\mathring{s}}_i^{tar,angular}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})\right\} \quad (67)$$

$\vec{S}_i^{blo}(\ddot{\boldsymbol{\pi}})$ is estimated by integrating $\vec{\mathring{s}}_i^{blo}$ on $\partial\mathscr{H}_i^{+}$ and on Ω_s:

$$\vec{S}_i^{blo}(\ddot{\boldsymbol{\pi}}) = \oint_{\partial\mathscr{H}_i^{+}} d\vec{x}_p \int_{\Omega_s} d\vec{\omega}_s \\ \left\{\mathscr{H}(hit_f\in\mathscr{R})\vec{\mathring{s}}_i^{blo}(\vec{x}_p,\vec{\omega}_p,\ddot{\boldsymbol{\pi}})\right\} \quad (68)$$

$\vec{S}_i^{shad-b}(\ddot{\vec{\pi}})$ is estimated by integrating $\vec{\mathring{s}}_i^{shad-b}$ on all the borders of other heliostats in the field $\partial\mathscr{H}_{i'}^{+}$ and on Ω_s:

$$\vec{S}_i^{shad-b}(\ddot{\vec{\pi}}) = \sum_{i'=1}^{n_{\mathscr{H}}, i'\neq i} \oint_{\partial\mathscr{H}_{i'}^{+}} d\vec{x}_p \int_{\Omega_s} d\vec{\omega}_s \Big\{ \mathscr{H}(hit_f \in \mathscr{R}) \vec{\mathring{s}}_i^{shad-b}(\vec{x}_p, \vec{\omega}_p, \ddot{\vec{\pi}}) \Big\} \quad (69)$$

$\vec{S}_i^{shad-f}(\ddot{\vec{\pi}})$ is estimated by integrating $\vec{\mathring{s}}_i^{shad-f}$ on $\partial\mathscr{H}_i^{-}$ and on Ω_s:

$$\vec{S}_i^{shad-f}(\ddot{\vec{\pi}}) = \oint_{\partial\mathscr{H}_i^{-}} d\vec{x}_b \int_{\Omega_s} d\vec{\omega}_s \Big\{ \mathscr{H}(hit_f \in \mathscr{R}) \vec{\mathring{s}}_i^{shad-f}(\vec{x}_b, \vec{\omega}_b, \ddot{\vec{\pi}}) \Big\} \quad (70)$$

Finally, the corresponding integral formulations of $\vec{S}^{tar}(\ddot{\vec{\pi}})$, $\vec{S}^{tar}(\ddot{\vec{\pi}})$, $\vec{S}^{blo}(\ddot{\vec{\pi}})$, $\vec{S}^{shad-b}(\ddot{\vec{\pi}})$ and $\vec{S}^{shad-f}(\ddot{\vec{\pi}})$ are shown in Section.3.

C Explicit expression of Monte-Carlo weight

C.1 Contribution of targeting

We substitute Eq.56 into Eq.42, as well as Eq.57 into Eq.43. The following explicit expressions of Monte-Carlo weight are then yielded:

$$\vec{w}_1 = -l_k DNI \begin{bmatrix} (\vec{\omega}_p \times \vec{v}_{i,1}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,2}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,3}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,4}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,5}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,6}) \cdot \vec{t}_{i,k} \end{bmatrix}; \quad (71)$$

$$\vec{w}_2 = 2\pi I_0 \mathcal{S}_{\mathscr{H}_i^{+}} |\vec{\omega}'_s \cdot \vec{n}_i| \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2(\vec{\omega}'_s \times \vec{a}_{i,\theta}) \cdot \vec{\omega}_c \\ \Big(\vec{\omega}'_s \times (\vec{a}'_{i,\phi} - \vec{a}_{i,\phi})\Big) \cdot \vec{\omega}_c \\ 0 \end{bmatrix}. \quad (72)$$

where $\mathcal{S}_{\mathscr{H}_i^{+}}$ the area of the reflecting surface of the heliostat $\mathscr{H}_i^{+}$, l_k the length of k^{th} side of the heliostat $\mathscr{H}_i$ and $\vec{t}_{i,k}$ the vectors of circulation which is clockwise around the normal $\vec{n}_i$ following the convention of[33], see Fig.19. $\vec{v}_{i,j}$ is the velocity of deformation linked to the point $\vec{x}'_{p1}$. Generally, it is a function of a point $\vec{x}$ on the heliostat $\mathscr{H}_i$:

$$\vec{v}_{i,j} \equiv \vec{v}_{i,j}(\vec{x}). \quad (73)$$

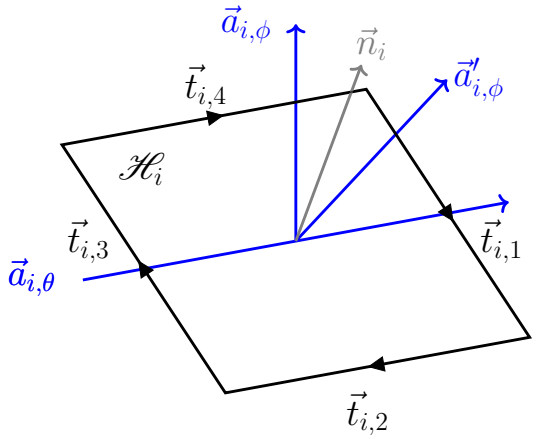


Figure 19: $\vec{t}_{i,k}$ are the clockwise vectors of circulation around $\vec{n}_i$

The vectors of $\vec{v}_{i,j}$ are given by Table.8, where $\vec{e} = \vec{x} - \vec{x}_{c,i}$ and $\vec{x}_{c,i}$ is the central position of the heliostat $\mathscr{H}_i$. The demonstrations are in Appendix.D and for example, the field of $\vec{v}_{i,4}(\vec{x}_p)$ and $\vec{v}_{i,5}(\vec{x}_p)$ are shown in Fig.20 and Fig.21.

$\vec{v}_{i,1}$	$\vec{v}_{i,2}$	$\vec{v}_{i,3}$	$\vec{v}_{i,4}$	$\vec{v}_{i,5}$	$\vec{v}_{i,6}$
$[1,0,0]$	$[0,1,0]$	$[0,0,1]$	$\vec{a}_{i,\theta} \times \vec{e}$	$\vec{a}_{i,\phi} \times \vec{e}$	$\frac{\vec{e}}{l_k}$

Table 8: Components of $\vec{v}_{i,j}$ for a point $\vec{x}$ on $\mathscr{H}_i$

In Eq.72, $\vec{a}_{i,\theta}$ is the axis of rotation related to $\ddot{\pi}_{i,4}$ and $\vec{a}_{i,\phi}$ the axis of rotation related to $\ddot{\pi}_{i,5}$. $\vec{a}'_{i,\phi}$ is the vector reflected by $\mathscr{H}_i^+$ from -$\vec{a}_{i,\phi}$ [33], see Fig.19.

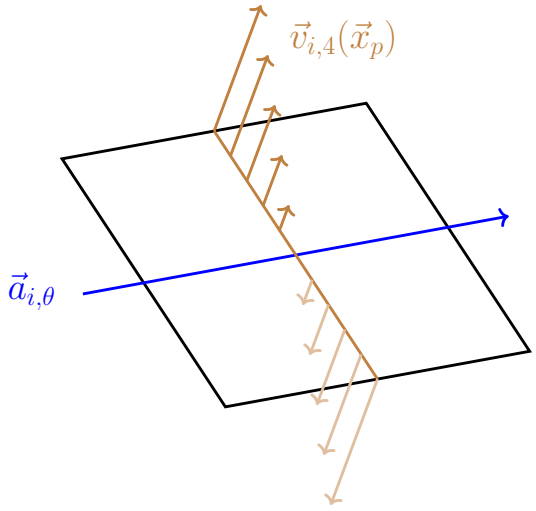


Figure 20: Schema of the field of $\vec{v}_{i,4}(\vec{x}_p)$

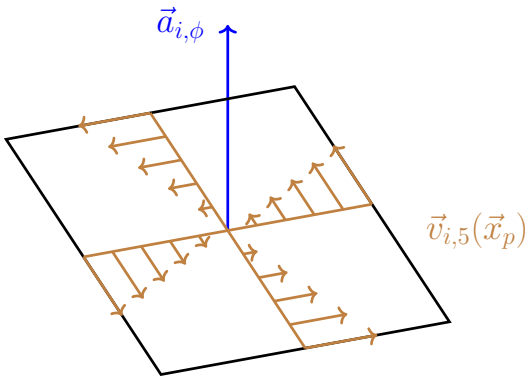


Figure 21: Schema of the field of $\vec{v}_{i,5}(\vec{x}_p)$

C.2 Contribution of backward-blocking

We substitute Eq.60 into Eq.45. The following explicit expression of Monte-Carlo weight is then yielded:

$$\vec{w}_3 = l_k DNI \begin{bmatrix} (\vec{\omega}_p \times \vec{v}_{i,1}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,2}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,3}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,4}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,5}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_p \times \vec{v}_{i,6}) \cdot \vec{t}_{i,k} \end{bmatrix}; \tag{74}$$

C.3 Contribution of forward-shadowing

We substitute Eq.65 into Eq.47. The following explicit expression of Monte-Carlo weight is then yielded:

$$\vec{w}_4 = -l_k DNI \begin{bmatrix} (\vec{\omega}_s \times \vec{v}_{i,1}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_s \times \vec{v}_{i,2}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_s \times \vec{v}_{i,3}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_s \times \vec{v}_{i,4}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_s \times \vec{v}_{i,5}) \cdot \vec{t}_{i,k} \\ (\vec{\omega}_s \times \vec{v}_{i,6}) \cdot \vec{t}_{i,k} \end{bmatrix}. \tag{75}$$

C.4 Contribution of backward-shadowing

We substitute Eq.62 into Eq.49, as well as Eq.63 into Eq.50. The following explicit expressions of Monte-Carlo weight are then yielded:

with

$$\vec{w}_5 = \rho l_k DNI \begin{bmatrix} \frac{\vec{v}_{i,1}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k} \\ \frac{\vec{v}_{i,2}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k} \\ \frac{\vec{v}_{i,3}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k} \\ \frac{\vec{v}_{i,4}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k} \\ \frac{\vec{v}_{i,5}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k} \\ \frac{\vec{v}_{i,6}\cdot\vec{n}_i}{\vec{\omega}_p\cdot\vec{n}_i}(\vec{\omega}_s\times\vec{\omega}_p)\cdot\vec{t}_{i',k} \end{bmatrix} \tag{76}$$

and

$$\vec{w}_6 = \rho l_k DNI l_{shad-b} \begin{bmatrix} 0 \\ 0 \\ 0 \\ -2(\vec{\omega}_s\times\vec{a}_{i,\theta}\times\vec{\omega}_s)\cdot\vec{t}_{i',k} \\ (\vec{\omega}_s\times(\vec{a}_{i,\phi}-\vec{a}_{i,\phi'})\times\vec{\omega}_s)\cdot\vec{t}_{i',k} \\ 0 \end{bmatrix} \tag{77}$$

D Velocity of deformation

Following [33], we need to define a geometric space, and a material space for a heliostat $\mathscr{H}_i$, and special notations are needed to calculate the velocity of deformation. In geometric space, the positions $\vec{\vec{y}}$ and directions $\vec{\vec{\omega}}$ are functions of the geometric parameter $\ddot{\pi}_{i,j}$, while in material space, the positions are noted $\vec{y}$ and directions are noted $\vec{\omega}$. They are not depended on $\ddot{\pi}_{i,j}$.

The function Z and Ω link the two spaces so that the positions and directions in one space can refer to the other space:

$$\vec{\vec{y}} = Z(\vec{y}, \ddot{\pi}_{i,j}); \vec{\vec{\omega}} = \Omega(\vec{\omega}, \ddot{\pi}) \tag{78}$$

We define the velocity of deformation $\vec{v}_{i,j}$ as the derivative of Z with respect to $\ddot{\pi}_{i,j}$:

$$\vec{v}_{i,j} \equiv \partial_{\ddot{\pi}_{i,j}} Z(\vec{y}, \ddot{\pi}_{i,j}) \tag{79}$$

For different $\ddot{\pi}_{i,j}$, the velocities of deformations $\vec{v}_{i,j}$ are summarized in Table.9, where $\vec{a}_\theta$ and $\vec{a}_\phi$ are two axes of rotation for the heliostat $\mathscr{H}_i$ (in Fig.4) and l_k the original length of size of the heliostat.

	$Z(\vec{y}, \ddot{\pi}_{i,j})$	$\vec{v} = \partial_{\ddot{\pi}} Z$
$\ddot{\pi}_{i,1}$	$\begin{bmatrix} \ddot{\pi}_{i,1} \\ 0 \\ 0 \end{bmatrix} + \vec{y}$	$\vec{v} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$
$\ddot{\pi}_{i,2}$	$\begin{bmatrix} 0 \\ \ddot{\pi}_{i,2} \\ 0 \end{bmatrix} + \vec{y}$	$\vec{v} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$
$\ddot{\pi}_{i,3}$	$\begin{bmatrix} 0 \\ 0 \\ \ddot{\pi}_{i,3} \end{bmatrix} + \vec{y}$	$\vec{v} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
$\ddot{\pi}_{i,4}$	$\vec{y} + \ddot{\pi}_{i,4}(\vec{a}_{i,\theta} \times \vec{y})$	$\vec{a}_{i,\theta} \times \vec{y}$
$\ddot{\pi}_{i,5}$	$\vec{y} + \ddot{\pi}_{i,5}(\vec{a}_{i,\phi} \times \vec{y})$	$\vec{a}_{i,\phi} \times \vec{y}$
$\ddot{\pi}_{i,6}$	$\frac{\ddot{\pi}_{i,6}}{l_k} \vec{y}$	$\frac{\vec{y}}{l_k}$

Table 9: The functions Z and $\vec{v}$ for the 6 geometric parameters of heliostat

E Algorithms

Algorithm 2 Estimate P

Input: Geometry of heliostats and receiver, $\vec{\omega}_c$, N
▷ N number of realizations

Initialization: $n \leftarrow 0$, $sum \leftarrow 0$, $sum2 \leftarrow 0$
while $n < N$ **do**
  Sample $\vec{x}_{p1}$ on $\mathcal{S}_{\mathscr{H}^+}$ based on p_X
  Sample $\vec{\omega}_s$ within Ω_s based on p_{Ω_s}.
  Compute $\vec{\omega}_p$ by the law of specular reflection
  Get $\vec{x}_{p0}$ from $\vec{x}_{p1}$ following $-\vec{\omega}_s$
  Get $\vec{x}_{p2}$ from $\vec{x}_{p1}$ following $\vec{\omega}_p$
  if $\vec{x}_{p0} \in \mathscr{O}_s$ & $\vec{x}_{p2} \in \mathscr{R}$ **then**
    Compute $\hat{w}$
    $sum \leftarrow sum + \hat{w}$
    $sum2 \leftarrow sum2 + \hat{w}^2$
  else
    $sum \leftarrow sum + 0$
    $sum2 \leftarrow sum2 + 0$
  end if
  $n \leftarrow n + 1$
end while
Output: $P \leftarrow \frac{sum}{N}$, $\sigma(P) \leftarrow \sqrt{\frac{\frac{sum2}{N} - (\frac{sum}{N})^2}{N-1}}$

Algorithm 3 Estimate $\vec{S}_i^{tar}$ and $\vec{\sigma}(\vec{S}_i^{tar})$

Input: Geometry of heliostats and receiver, $\vec{\omega}_c$, N

▷ N number of realizations

Initialization: $\vec{S}_i^{tar} \leftarrow \mathbf{0}$, $\vec{\sigma}(\vec{S}_i^{tar}) \leftarrow \mathbf{0}$

while $k < 4$ **do** ▷ $\mathscr{H}_i$ has 4 borders

Initialization: $n \leftarrow 0$, $\overrightarrow{sum} \leftarrow \mathbf{0}$, $\overrightarrow{sum2} \leftarrow \mathbf{0}$

while $n < N$ **do**

Sample $\vec{x}'_{p1}$ on l_k based on $p_{X'}$

Sample $\vec{\omega}_s$ within Ω_s based on p_{Ω_s}.

Compute $\vec{\omega}_p$ based on $\vec{\omega}_s$ and $\vec{n}_i$ by the law of specular reflection

Get $\vec{x}'_{p0}$ from $\vec{x}'_{p1}$ following $-\vec{\omega}_s$

Get $\vec{x}'_{p2}$ from $\vec{x}'_{p1}$ following $\vec{\omega}_p$

if $\vec{x}'_{p0} \in \mathscr{O}_s$ & $\vec{x}'_{p2} \in \mathscr{R}$ **then**

Compute $\vec{w}_1$

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \vec{w}_1$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \vec{w}_1^2$

else

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \mathbf{0}$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \mathbf{0}$

end if

$n \leftarrow n + 1$

end while

$\vec{S}_i^{tar} \leftarrow \vec{S}_i^{tar} + \frac{\overrightarrow{sum}}{N}$, $\vec{\sigma}(\vec{S}_i^{tar}) \leftarrow \vec{\sigma}(\vec{S}_i^{tar}) + \sigma(P) \leftarrow \sqrt{\frac{\frac{\overrightarrow{sum2}}{N} - (\frac{\overrightarrow{sum}}{N})^2}{N-1}}$

$k \leftarrow k + 1$

end while

Initialization: $n \leftarrow 0$, $\overrightarrow{sum} \leftarrow \mathbf{0}$, $\overrightarrow{sum2} \leftarrow \mathbf{0}$

while $n < N$ **do** ▷ N number of realizations

Sample $\vec{x}_{p1}$ on $\mathcal{S}_{\mathscr{H}_i^+}$ based on p_{X_i}

Sample $\vec{\omega}'_s$ on $\partial\Omega_s$ based on $p_{\Omega'_s}$.

Compute $\vec{\omega}'_p$ based on $\vec{\omega}'_s$ and $\vec{n}_i$ by the law of specular reflection

Get $\vec{x}_{p0}$ from $\vec{x}_{p1}$ following $-\vec{\omega}'_s$

Get $\vec{x}_{p2}$ from $\vec{x}_{p1}$ following $\vec{\omega}'_p$

if $\vec{x}_{p0} \in \mathscr{O}_s$ & $\vec{x}_{p2} \in \mathscr{R}$ **then**

Compute $\vec{w}_2$

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \vec{w}_2$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \vec{w}_2^2$

else

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \mathbf{0}$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \mathbf{0}$

end if

$n \leftarrow n + 1$

end while

$\vec{S}_i^{tar} \leftarrow \vec{S}_i^{tar} + \frac{\overrightarrow{sum}}{N}$, $\vec{\sigma}(\vec{S}_i^{tar}) \leftarrow \vec{\sigma}(\vec{S}_i^{tar}) + \sigma(P) \leftarrow \sqrt{\frac{\frac{\overrightarrow{sum2}}{N} - (\frac{\overrightarrow{sum}}{N})^2}{N-1}}$

Output: $\vec{S}_i^{tar}$, $\vec{\sigma}(\vec{S}_i^{tar})$

Algorithm 4 Estimate $\vec{S}_i^{blo}$ and $\vec{\sigma}(\vec{S}_i^{blo})$

Input: Geometry of heliostats and receiver, $\vec{\omega}_c$, N, i

▷ N number of realizations

Initialization: $i' \leftarrow 0$, $\vec{S}_i^{blo} \leftarrow \mathbf{0}$, $\vec{\sigma}(\vec{S}_i^{blo}) \leftarrow \mathbf{0}$ ▷ i' index of heliostat being blocked

while $i' < n_{\mathscr{H}}$ **do** ▷ $n_{\mathscr{H}}$ number of heliostats

Compute the distance $l_{i,i'}$ between $\mathscr{H}_i$ and $\mathscr{H}_{i'}$

if $l_{i,i'} < l_d$ & $i' \neq i$ **then**

▷ Ignore the too-far-away heliostats for backward-blocking effect

Initialization: $k \leftarrow 0$

while $k < 4$ **do** ▷ $\mathscr{H}_i$ has 4 borders

Initialization: $n \leftarrow 0$, $\overrightarrow{sum} \leftarrow \mathbf{0}$, $\overrightarrow{sum2} \leftarrow \mathbf{0}$

while $n < N$ **do**

Sample $\vec{x}'_{p2}$ on l_k based on $p_{X'}$

Sample $\vec{\omega}_s$ within Ω_s based on p_{Ω_s}.

Compute $\vec{\omega}_p$ based on $\vec{\omega}_s$ and $\vec{n}_{i'}$ by the law of specular reflection

Get $\vec{x}'_{p1}$ from $\vec{x}'_{p2}$ following $-\vec{\omega}_p$

Get $\vec{x}'_{p0}$ from $\vec{x}'_{p1}$ following $-\vec{\omega}_s$

Get $\vec{x}'_{p3}$ from $\vec{x}'_{p2}$ following $\vec{\omega}_p$

if $\vec{x}'_{p0} \in \mathscr{O}_s$ & $\vec{x}'_{p1} \in \mathscr{H}_{i'}^{+}$ & $\vec{x}'_{p3} \in \mathscr{R}$ **then**

Compute $\vec{w}$

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} - \vec{w}_1$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} - \vec{w}_1^2$

else

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \mathbf{0}$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \mathbf{0}$

end if

$n \leftarrow n + 1$

end while

$k \leftarrow k + 1$

$\vec{S}_i^{blo} \leftarrow \vec{S}_i^{blo} + \frac{\overrightarrow{sum}}{N}$, $\vec{\sigma}(\vec{S}_i^{blo}) \leftarrow \vec{\sigma}(\vec{S}_i^{blo}) + \sigma(P) \leftarrow \sqrt{\frac{\frac{\overrightarrow{sum2}}{N} - (\frac{\overrightarrow{sum}}{N})^2}{N-1}}$

end while

end if

$i' \leftarrow i' + 1$

end while

Output: $\vec{S}_i^{blo}$, $\vec{\sigma}(\vec{S}_i^{blo})$

Algorithm 5 Estimate $\vec{S}_i^{shad-f}$ and $\vec{\sigma}(\vec{S}_i^{shad-f})$

Input: Geometry of heliostats and receiver, $\vec{\omega}_c$, N

▷ N number of realizations

Initialization: $k \leftarrow 0$, $\vec{S}_i^{shad-f} \leftarrow \mathbf{0}$, $\vec{\sigma}(\vec{S}_i^{shad-f}) \leftarrow \mathbf{0}$

while $k < 4$ **do** ▷ $\mathscr{H}_i$ has 4 borders

Initialization: $n \leftarrow 0$, $\overrightarrow{sum} \leftarrow \mathbf{0}$, $\overrightarrow{sum2} \leftarrow \mathbf{0}$

while $n < N$ **do**

Sample $\vec{x}'_{p1}$ on l_k of $\mathscr{H}_i^+$ based on $p_{X'}$

Sample $\vec{\omega}_s$ within Ω_s based on p_{Ω_s}.

Get $\vec{x}'_{p2}$ from $\vec{x}'_{p1}$ following $\vec{\omega}_s$

Get the normal $\vec{n}_{i'}$ on $\vec{x}_{p2}$

Compute $\vec{\omega}_p$ based on $\vec{\omega}_s$ and $\vec{n}_{i'}$ by the law of specular reflection

Get $\vec{x}'_{p0}$ from $\vec{x}'_{p1}$ following $-\vec{\omega}_s$

Get $\vec{x}'_{p3}$ from $\vec{x}'_{p2}$ following $\vec{\omega}_p$

if $\vec{x}'_{p0} \in \mathscr{O}_s$ & $\vec{x}'_{p1} \in \mathscr{H}^+$ & $\vec{x}'_{p3} \in \mathscr{R}$ **then**

Compute $\vec{w}$

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \vec{w}_3$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \vec{w}_3^2$

else

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \mathbf{0}$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \mathbf{0}$

end if

$n \leftarrow n + 1$

end while

$k \leftarrow k + 1$

end while

$\vec{S}_i^{shad-f} \leftarrow \vec{S}_i^{shad-f} + \frac{\overrightarrow{sum}}{N}$, $\vec{\sigma}(\vec{S}_i^{shad-f}) \leftarrow \vec{\sigma}(\vec{S}_i^{shad-f}) + \sigma(P) \leftarrow \sqrt{\frac{\frac{\overrightarrow{sum2}}{N} - (\frac{\overrightarrow{sum}}{N})^2}{N-1}}$

Output: $\vec{S}_i^{shad-f}$, $\vec{\sigma}(\vec{S}_i^{shad-f})$

Algorithm 6 Estimate $\vec{S}_i^{shad-b}$ and $\vec{\sigma}(\vec{S}_i^{shad-b})$

Input: Geometry of heliostats and receiver, $\vec{\omega}_c$, N, i

▷ N number of realizations

Initialization: $i' \leftarrow 0$, $\vec{S}_i^{shad-b} \leftarrow \mathbf{0}$, $\vec{\sigma}(\vec{S}_i^{shad-b}) \leftarrow \mathbf{0}$ ▷ i' index of shadowing heliostat

while $i' < n_{\mathscr{H}}$ **do** ▷ $n_{\mathscr{H}}$ number of heliostats

Compute the distance $l_{i,i'}$ between $\mathscr{H}_i$ and $\mathscr{H}_{i'}$

if $l_{i,i'} < l_d$ & $i' \neq i$ **then**

▷ Ignore the too-far-away heliostats for backward-shadowing effect

Initialization: $k \leftarrow 0$

while $k < 4$ **do** ▷ loop all borders on $\mathscr{H}_{i'}$

Initialization: $n \leftarrow 0$, $\overrightarrow{sum} \leftarrow \mathbf{0}$, $\overrightarrow{sum2} \leftarrow \mathbf{0}$

while $n < N$ **do**

Sample $\vec{x}'_{p1}$ on l_k of $\mathscr{H}_{i'}$ based on $p_{X'}$

Sample $\vec{\omega}_s$ within Ω_s based on p_{Ω_s}.

Get $\vec{x}'_{p2}$ from $\vec{x}'_{p1}$ following $\vec{\omega}_s$

Compute $\vec{\omega}_p$ based on $\vec{\omega}_s$ and $\vec{n}_i$ by the law of specular reflection

Get $\vec{x}'_{p0}$ from $\vec{x}'_{p1}$ following $-\vec{\omega}_s$

Get $\vec{x}'_{p3}$ from $\vec{x}'_{p2}$ following $\vec{\omega}_p$

if $\vec{x}'_{p0} \in \mathscr{O}_s$ & $\vec{x}'_{p2} \in \mathscr{H}_i^+$ & $\vec{x}'_{p3} \in \mathscr{R}$ **then**

Compute $\vec{w}_4$ and $\vec{w}_5$

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \vec{w}_4 + \vec{w}_5$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + (\vec{w}_4 + \vec{w}_5)^2$

else

$\overrightarrow{sum} \leftarrow \overrightarrow{sum} + \mathbf{0}$

$\overrightarrow{sum2} \leftarrow \overrightarrow{sum2} + \mathbf{0}$

end if

$n \leftarrow n + 1$

end while

$k \leftarrow k + 1$

$\vec{S}_i^{shad-b} \leftarrow \vec{S}_i^{shad-b} + \frac{\overrightarrow{sum}}{N}$, $\vec{\sigma}(\vec{S}_i^{shad-b}) \leftarrow \vec{\sigma}(\vec{S}_i^{shad-b}) + \sigma(P) \leftarrow \sqrt{\frac{\frac{\overrightarrow{sum2}}{N} - (\frac{\overrightarrow{sum}}{N})^2}{N-1}}$

end while

end if

$i' \leftarrow i' + 1$

end while

Output: $\vec{S}_i^{shad-b}$, $\vec{\sigma}(\vec{S}_i^{shad-b})$

References

[1] Clifford K Ho. Advances in central receivers for concentrating solar applications. *Solar energy*, 152:38–56, 2017.

[2] Reiner Buck. Heliostat field layout improvement by nonrestricted refinement. *Journal of solar energy engineering*, 136(2), 2014.

[3] Lorin L Vant-Hull. An educated ray trace approach to solar tower optics. *Optical Engineering*, 16(5):497–504, 1977.

[4] FW Lipps and Lorin L Vant-Hull. User manual for the university of houston solar central receiver, cellwise performance model: Ns. Technical report, Houston Univ., TX (USA). Energy Lab., 1980.

[5] Lifeng Li, Joe Coventry, Roman Bader, John Pye, and Wojciech Lipiński. Optics of solar central receiver systems: a review. *Optics express*, 24(14):A985–A1007, 2016.

[6] Manuel J Blanco, Amaia Mutuberria, Pierre Garcia, Raquel Gastesi, and Victor Martin. Preliminary validation of tonatiuh. In *SolarPACES conference*, 2009.

[7] Cyril Caliot, Hadrien Benoit, Emmanuel Guillot, Jean-Louis Sans, Alain Ferriere, Gilles Flamant, Christophe Coustet, and Benjamin Piaud. Validation of a monte carlo integral formulation applied to solar facility simulations and use of sensitivities. *Journal of Solar Energy Engineering*, 137(2):021019, 2015.

[8] PL Leary and JD Hankins. User's guide for mirval: a computer code for comparing designs of heliostat-receiver optics for central receiver solar power plants. Technical report, Sandia National Lab.(SNL-CA), Livermore, CA (United States), 1979.

[9] Tim Wendelin. Soltrace: a new optical modeling tool for concentrating solar optics. In *International solar energy conference*, volume 36762, pages 253–260, 2003.

[10] Peter Schwarzbözl, Robert Pitz-Paal, and Mark Schmitz. Visual hflcal-a software tool for layout and optimisation of heliostat fields. In *Proceedings*, 2009.

[11] Charles N Vittitoe and Frank Biggs. User's guide to helios: a computer program for modeling the optical behavior of reflecting solar concentrators. part iii. appendices concerning helios-code details. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 1981.

[12] Michael J Wagner and Tim Wendelin. Solarpilot: A power tower solar field layout and characterization tool. *Solar Energy*, 171:185–196, 2018.

[13] Pierre Garcia, Alain Ferriere, and Jean-Jacques Bezian. Codes for solar flux calculation dedicated to central receiver system applications: A comparative review. *Solar Energy*, 82(3):189–197, 2008.

[14] Ye Wang, Daniel Potter, Charles-Alexis Asselineau, Clotilde Corsi, Michael Wagner, Cyril Caliot, Benjamin Piaud, Manuel Blanco, Jin-Soo Kim, and John Pye. Verification of optical modelling of sunshape and surface slope error for concentrating solar power systems. *Solar Energy*, 195:461–474, 2020.

[15] Robert Pitz-Paal, Nicolas Bayer Botero, and Aldo Steinfeld. Heliostat field layout optimization for high-temperature solar thermochemical processing. *Solar energy*, 85(2):334–343, 2011.

[16] Olivier Farges, Jean-Jacques Bézian, Hélène Bru, Mouna El Hafi, Richard Fournier, and Christophe Spiesser. Life-time integration using monte carlo methods when optimizing the design of concentrated solar power plants. *Solar Energy*, 113:57–62, 2015.

[17] Miguel Diago, Nicolas Calvet, and Peter R Armstrong. Net power maximization from a faceted beam-down solar concentrator. *Solar Energy*, 204:476–488, 2020.

[18] Yiyi Zhou and Yuhong Zhao. Heliostat field layout design for solar tower power plant based on gpu. *IFAC Proceedings Volumes*, 47(3):4953–4958, 2014.

[19] Y Favennec, F Dubot, D Le Hardy, B Rousseau, and DR Rousse. Space-dependent sobolev gradients as a regularization for inverse radiative transfer problems. *Mathematical Problems in Engineering*, 2016, 2016.

[20] Seonghyeok Yang, Kyungeun Lee, and Ikjin Lee. Pattern-free heliostat field layout optimization using physics-based gradient. *Solar Energy*, 206:722–731, August 2020.

[21] AJ Marston, KJ Daun, and MR Collins. Geometric optimization of concentrating solar collectors using monte carlo simulation. *Journal of solar energy engineering*, 132(4), 2010.

[22] Andrew James Marston. Geometric optimization of solar concentrating collectors using quasi-monte carlo simulation. Master's thesis, University of Waterloo, 2010.

[23] Emmanuel Gobet. Stochastic differential equations and Feynman-Kac formulas. In *Monte-Carlo Methods and Stochastic Processes*. Chapman and Hall/CRC, 2016. Num Pages: 46.

[24] Jérémie Delatorre, Germain Baud, Jean-Jacques Bézian, Stéphane Blanco, Cyril Caliot, Jean-François Cornet, Christophe Coustet, Jérémi Dauchet, Mouna El Hafi, Vincent Eymet, et al. Monte carlo advances and concentrated solar applications. *Solar Energy*, 103:653–681, 2014.

[25] A De Lataillade, Stéphane Blanco, Y Clergent, Jean-Louis Dufresne, Mouna El Hafi, and Richard Fournier. Monte carlo method and sensitivity estimations. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 75(5):529–538, 2002.

[26] Maxime Roger, Stéphane Blanco, Mouna El Hafi, and Richard Fournier. Monte carlo estimates of domain-deformation sensitivities. *Physical review letters*, 95(18):180601, 2005.

[27] P Lapeyre, Stéphane Blanco, Cyril Caliot, J Dauchet, Mouna El Hafi, Richard Fournier, Olivier Farges, Jacques Gautrais, and Maxime Roger. Monte-carlo and sensitivity transport models for domain deformation. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 251:107022, 2020.

[28] Cheng Zhang, Bailey Miller, Kan Yan, Ioannis Gkioulekas, and Shuang Zhao. Path-space differentiable rendering. *ACM transactions on graphics*, 39(4), 2020.

[29] Cheng Zhang, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao. A differential theory of radiative transfer. *ACM Transactions on Graphics (TOG)*, 38(6):1–16, 2019.

[30] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Transactions on Graphics (TOG)*, 37(6):1–11, 2018.

[31] Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. Monte carlo estimators for differential light transport. *ACM Transactions on Graphics (TOG)*, 40(4):1–16, 2021.

[32] Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. Differentiable rendering: A survey. *arXiv preprint arXiv:2006.12057*, 2020.

[33] Paule Lapeyre, Zili He, Stéphane Blanco, Cyril Caliot, Christophe Coustet, Jérémi Dauchet, Mouna El Hafi, Simon Eibner, Eugene d'Eon, Olivier Farges, Richard Fournier, Jacques Gautrais, Nada Chems Mourtaday, and Maxime Roger. A physical model and a monte carlo estimate for the specific intensity spatial derivative, angular derivative and geometric sensitivity, 2022.

[34] Steve Schell. Design and evaluation of esolar's heliostat fields. *Solar Energy*, 85(4):614–619, 2011.

[35] Philippe Blanc, Bella Espinar, Norbert Geuder, Christian Gueymard, Richard Meyer, Robert Pitz-Paal, Bernhard Reinhardt, David Renné, Manajit Sengupta, Lucien Wald, et al. Direct normal irradiance related definitions and applications: The circumsolar issue. *Solar Energy*, 110:561–577, 2014.

[36] John R Howell, M Pinar Mengüç, Kyle Daun, and Robert Siegel. *Thermal radiation heat transfer*. CRC press, 2020.

[37] Daniel Buie, AG Monger, and CJ Dey. Sunshape distributions for terrestrial solar simulations. *Solar energy*, 74(2):113–122, 2003.

[38] Charles N Vittitoe and Frank Biggs. Six-gaussian representation of the angular-brightness distribution for solar radiation. *Solar Energy*, 27(6):469–490, 1981.

[39] Benjamin Piaud, Christophe Coustet, Cyril Caliot, Emmanuel Guillot, and Gilles Flamant. Application of Monte-Carlo sensitivities estimation in Solfast-4D. In *SolarPaces 2012 Conference*, pages 1–9, 2012.

[40] William L Dunn and J Kenneth Shultis. *Exploring monte carlo methods*. Elsevier, 2011.

6

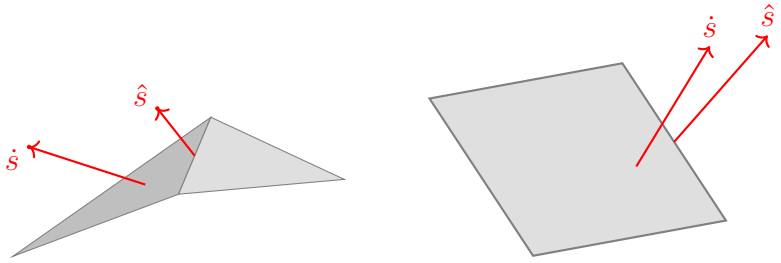
Transforming line sources into surface sources by Stokes' theorem

6.1 Introduction

In the last chapter, the method of sensitivity model has been applied to the research field of CSP, using the technique of vectorization. Now, let's first take a step back and review the previous chapters.

In Chapter.2, three methods are distinguished to tackle the problem of "estimating the sensitivities of a radiative system by Monte-Carlo method". They are then applied to a one-dimensional problem in Chapter.3. After that, Chapter.4 aims to apply the method of sensitivity model to complex, triangulated geometries. We also proposed the strategy of vectorization in Chapter.4 to manage the increase of the sources in the models of sensitivity. Also, following the same strategy of vectorization, the method of sensitivity model is applied to a CSP system in Chapter.5.

In Chapter.4 and Chapter.5, the boundary conditions are discontinuous (on the junction of two triangles in Chapter.4, seeing Fig.6.1a and the borders of heliostats in Chapter.5, seeing Fig.6.1b).



(a) Junction of two triangles. **(b)** A heliostat.

Figure 6.1 – The schema of line source $\hat{s}$ and the surface source $\dot{s}$ for the junction of two triangles and a heliostat.

In this case, the sources of sensitivity are found on the surface (noted $\dot{s}$) and on the edge (noted $\hat{s}$), these sources have then different path spaces on the boundary. Therefore, when we integrate all emitted sources to get the outgoing sensitivity of power (noted S), it is required to build an integral formulation for the surface sources and the other one for the line sources:

$$S = \int_{\mathcal{S}} \int_{2\pi(\vec{n})} |\vec{\omega} \cdot \vec{n}| \dot{s} d\vec{x}_p d\vec{\omega} + \oint_{\mathcal{L}} \int_{2\pi(\vec{n})} |\vec{\omega} \cdot \vec{n}| \hat{s} d\vec{x}_l d\vec{\omega} \tag{6.1}$$

where $\mathcal{S}$ is the area of the surface of the triangle or heliostat, $\mathcal{L}$ the length of the junction or the border of the heliostat, $\vec{x}_p$ a vector of position on the surface and $\vec{x}_l$ a vector of position on the junction or on the border of a heliostat.

More generally, it is required to estimate the observable $\mathcal{O}$ and all of its sensitivities $S_1, S_2, \ldots, S_n$ for a given radiative system. Following the strategy of vectorization proposed in Chapter.4 and Chapter.5, the line sources and surface sources are vectorized separately:

$$\begin{bmatrix} \mathcal{O} \\ S_1 \\ S_2 \\ \ldots \\ S_n \end{bmatrix} = \int_{\mathcal{S}} \int_{2\pi(\vec{n})} |\vec{\omega} \cdot \vec{n}| \begin{bmatrix} \mathring{I} \\ \dot{s}_1 \\ \dot{s}_2 \\ \ldots \\ \dot{s}_n \end{bmatrix} d\vec{x}_p d\vec{\omega} + \oint_{\mathcal{L}} \int_{2\pi(\vec{n})} |\vec{\omega} \cdot \vec{n}| \begin{bmatrix} 0 \\ \hat{s}_1 \\ \hat{s}_2 \\ \ldots \\ \hat{s}_n \end{bmatrix} d\vec{x}_l d\vec{\omega} \tag{6.2}$$

where $\mathring{I}$ is the source of intensity, which is usually a surface source. $\dot{s}_1, \dot{s}_2, \ldots, \dot{s}_n$ are the surface sources of sensitivity and $\hat{s}_1, \hat{s}_2, \ldots, \hat{s}_n$ are the line sources of sensitivity.

However, the line integral in Eq.6.2 implies two inconveniences when applying the Monte-Carlo method:

1. For a complex, triangulated geometry, it is complicated to sample over all edges. Some complex algorithms are developed to achieve this kind of sampling in the community of image synthesis [69].

2. On the boundary, a surface sampling and a line sampling are needed, corresponding to two path spaces (or two integral formulations as it is shown in Eq.6.1). However, this means that the two integral requires two Monte-Carlo estimations.

In this chapter, we explore the possibility of transforming the line sources $\hat{s}$ into surface sources. The line source $\hat{s}$ is transformed to surface source $\dot{\hat{s}}$. After that, only one space path is found on the boundary, as shown in Fig.6.2, corresponding to only one integral formulation. Eq.6.1 becomes:

$$S = \int_{\mathcal{S}} \int_{2\pi(\vec{n})} |\vec{\omega} \cdot \vec{n}| (\dot{s} + \dot{\hat{s}}) d\vec{x}_p d\vec{\omega} \tag{6.3}$$

Consequently, line integral no longer existed in Eq.6.3. The line sampling is avoided, and the path space of the sources $\dot{s}$ and $\dot{\hat{s}}$ is the same.

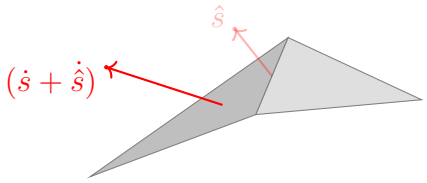


Figure 6.2 – Transforming line source $\hat{s}$ into surface source $\dot{\hat{s}}$.

More generally, we perform this technique to the line sources of all sensitivities in Eq.6.2. It becomes:

$$\begin{bmatrix} \mathcal{O} \\ S_1 \\ S_2 \\ \dots \\ S_n \end{bmatrix} = \int_{\mathcal{S}} \int_{2\pi(\vec{n})} |\vec{\omega} \cdot \vec{n}| \begin{bmatrix} \mathring{I} \\ \dot{s}_1 + \dot{\hat{s}}_1 \\ \dot{s}_2 + \dot{\hat{s}}_2 \\ \dots \\ \dot{s}_n + \dot{\hat{s}}_n \end{bmatrix} d\vec{x}_p d\vec{\omega} \tag{6.4}$$

Finally, all sources of sensitivity and intensity are surface sources, sharing only one path space in Eq.6.4. The observable and all its sensitivities are estimated in a vectorized form, with only one Monte-Carlo estimation. We remind that this is the significant advantage of the method of observable estimation that is introduced in Chapter.2 and Chapter.3. Therefore, following the method of sensitivity model, the vectorization advantage of the method of observable derivation is completely regained. This chapter dedicates to introducing this technique of transformation.

The model of sensitivity is coupled with the model of intensity, the model of spatial derivative, and the model of angular derivative [34]. As it is mentioned in Chapter.4, the origin of the line source of sensitivity always comes from the model of spatial derivative. Therefore, we focus on it, and the chapter is organized as follows:

- In Section.6.2, we first define a radiative case study. After that, we build a radiative transport model both for the intensity and for its spatial derivative. We also discuss how the line sources appear in this section.
- In Section.6.3, we reformulate the emitted line sources, transforming them into surface sources so that only surface sources exist in the system.
- The technique of transforming line sources into surface sources is tested in different case with different medium: transparent (Section.6.4), absorbent (Section.6.5) and scattering (Section.6.6).
- Section.6.7 is for validation purposes. Also, a convergence issue is distinguished and discussed.

6.2 Modeling of the study case

6.2.1 Description of the study case

We begin with an example of a convex domain bounded by plane surfaces (Fig.6.3). The geometric domain is noted $\mathcal{G} \subset \mathbb{R}^3$, bounded by the boundary $\partial\mathcal{G}$. Also, the edges between two plane surfaces are noted $\mathcal{L} \subset \mathcal{G}$. The set $\mathcal{G}$ includes then the interior of the geometric domain, its boundary and the edges: $\mathcal{G} = \mathring{\mathcal{G}} \cup \partial\mathcal{G} \cup \mathcal{L}$.

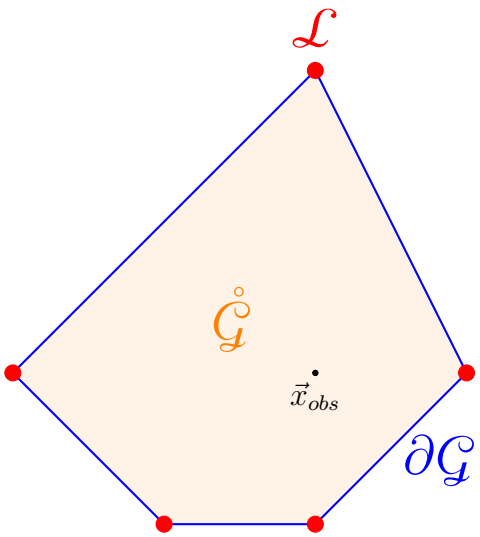


Figure 6.3 – A convex domain bounded by plane surfaces where $\mathcal{G} = \mathring{\mathcal{G}} \cup \partial\mathcal{G} \cup \mathcal{L}$. It is noted that the 3D geometry is presented as a 2D figure here. Therefore, the volume $\mathring{\mathcal{G}}$, the surface $\partial\mathcal{G}$, and the lines $\mathcal{L}$ are presented with one-dimension less.

The intensity $I \equiv I(\vec{x}, \vec{\omega})$ has 2 independent variables:

- The position $\vec{x}$ is a vector of $\mathbb{R}^3$ that belongs to the set $\mathcal{G} \subset \mathbb{R}^3$.
- The direction $\vec{\omega}$ is an element of the unit sphere $\mathcal{S}$.

We also make the following assumptions:

1. The medium is cold, homogeneous, and isotropic with transparent, absorbent, or scattering medium (the three cases are studied differently in the following sections).
2. The bounded plane surfaces are all "black bodies".

An observation point $\vec{x}_{obs} \in \mathcal{G}$ is set in the domain (Fig.6.3). The observable is the angular integral of intensity around $\vec{x}_{obs}$, which is noted G (Eq.6.5), as well as its spatial gradient projected to a unit vector $\vec{\chi}$ (Eq.6.6).

$$G(\vec{x}_{obs}) = \int_{4\pi} I(\vec{x}_{obs}, \vec{\omega}) d\vec{\omega} \tag{6.5}$$

$$\partial_{1,\vec{\chi}} G(\vec{x}_{obs}) = \int_{4\pi} \partial_{1,\vec{\chi}} I(\vec{x}_{obs}, \vec{\omega}) d\vec{\omega} \tag{6.6}$$

In order to estimate G and $\partial_{1,\vec{\chi}} G$, we will build a model of intensity and a model of spatial derivative, and we solve them by the Monte Carlo method.

6.2.2 The model of intensity

The model of intensity is already introduced in Chapter.1. Noting $\mathcal{C}$ as the collision operator, the stationary RTE with the assumptions we made for this chapter is:

$$\vec{\omega} \cdot \partial_1 I = \mathcal{C}[I] \qquad \vec{x} \in \mathcal{G} \tag{6.7}$$

with

$$\mathcal{C}[I(\vec{x},\vec{\omega})] = -k_a I(\vec{x},\vec{\omega}) - k_s I(\vec{x},\vec{\omega}) + k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x},\vec{\omega}') \tag{6.8}$$

where k_a is the absorption coefficient, k_s the scattering coefficient, $\frac{1}{4\pi}$ the single scattering phase function in an isotropic scattering medium, which can be regarded as the density of probability that the scattering direction becomes $\vec{\omega}$ for a photon coming from the direction of $\vec{\omega}'$.

The boundary $\partial\mathcal{G}$ is only composed of black bodies. Therefore, they only emit intensity, and no intensity is reflected:

$$I = \mathring{I} \qquad \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \tag{6.9}$$

with $\mathring{I}$, which is noted as the source of intensity and $\vec{n}$ the normal of the plane surface, pointing to the inside of the domain $\mathring{\mathcal{G}}$.

Intensity is the radiant flux emitted, reflected, transmitted, or received by a given surface per unit solid angle per unit projected area. Its unit is $\mathrm{W/m^2}$ sr, and it is a physical quantity defined on a surface. Therefore, we do not define the boundary condition of intensity for$\mathcal{L}$.

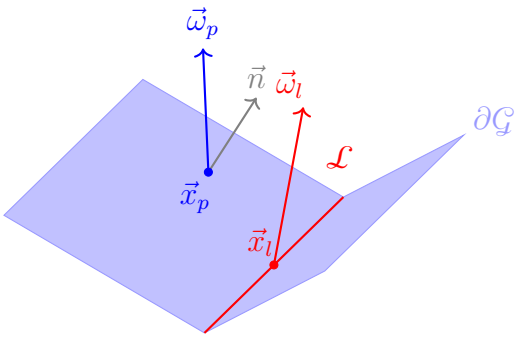


Figure 6.4 – A schema of notations on the boundaries. $\vec{x}_p$ and $\vec{\omega}_p$ are on the surface, where the normal is noted $\vec{n}$. $\vec{x}_l$ and $\vec{\omega}_l$ are on the edge.

To sum up, we use the index p for the boundary condition on $\partial\mathcal{G}$, so that $\vec{x}_p \in \partial\mathcal{G}$ and $\vec{\omega}_p \in \acute{\mathcal{S}}$, where $\acute{\mathcal{S}}$ is the hemisphere characterized by $\vec{\omega}_p \cdot \vec{n} > 0$. We use the index l for the boundary conditions on $\mathcal{L}$, so that $\vec{x}_l \in \mathcal{L}$ and $\vec{\omega}_l \in \mathcal{S}$, (seeing a schema of notations in Fig.6.4). The model of transport of intensity becomes:

$$\begin{cases} \vec{\omega} \cdot \partial_1 I(\vec{x},\vec{\omega}) = \mathcal{C}[I(\vec{x},\vec{\omega})] \\ I(\vec{x}_p,\vec{\omega}_p) = \mathring{I}(\vec{x}_p,\vec{\omega}_p) \end{cases} \tag{6.10}$$

It can be noticed that in the model of intensity (Eq.6.10), the only source in the system is on the surface boundary $\partial\mathcal{G}$.

6.2.3 The model of spatial derivative

The model of the spatial derivative is well studied in [34]. This subsection is a brief summary of the model of spatial derivative. The spatial derivative is made along a given direction, namely along a unit vector $\vec{\chi}$, which means that:

$$\partial_{1,\vec{\chi}} I = \vec{\chi} \cdot \vec{\nabla} I = \partial_{1,\vec{\chi}} I(\vec{x}, \vec{\omega}) = \lim_{\tau \to 0} \frac{I(\vec{x} + \tau\vec{\chi}) - I(\vec{x}, \vec{\omega})}{\tau} \tag{6.11}$$

Therefore, the spatial derivative $\partial_{1,\vec{\chi}} I$ has three independent variables: $(\vec{x}, \vec{\omega}, \vec{\chi})$ they are the position $\vec{x}$, the direction of propagation $\vec{\omega}$ and the direction of differentiation $\vec{\chi}$ (Fig.6.5).

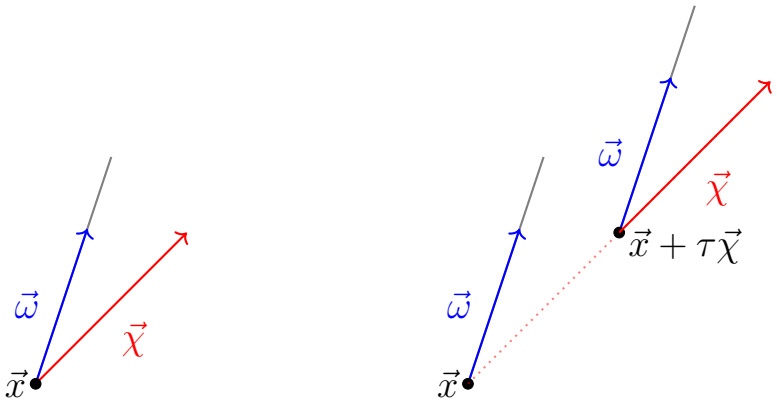
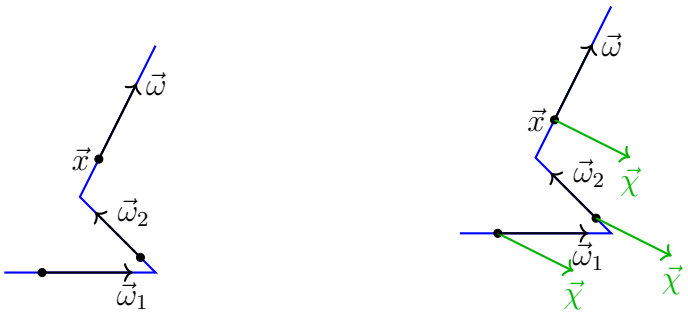


Figure 6.5 – The spatial derivative $\partial_{1,\vec{\chi}} I$ pictured as an elementary displacement following the direction of differentiation $\vec{\chi}$ (Eq.6.11). When picturing the physical model of spatial derivative of intensity, we need to draw the location $\vec{x}$, the direction of propagation $\vec{\omega}$ and also the direction of differentiation $\vec{\chi}$.

The spatial derivative $\partial_{1,\vec{\chi}} I$ is also considered as a physical quantity that could be absorbed, scattered in the medium, and also reflected, emitted on the boundaries (seeing Fig.6.6 for a multiple-scattering schema).



(a) Scattering of I in the medium. **(b)** Scattering of $\partial_{1,\vec{\chi}} I$ in the medium.

Figure 6.6 – A multiple-scattering photon trajectory leading to location $\vec{x}$ and transport direction $\vec{\omega}$. Nothing changes for the transport of $\partial_{1,\vec{\chi}} I$, compared to the transport of I. The differentiation direction $\vec{\chi}$ is conserved at each scattering event.

Following the assumptions made for this chapter (homogeneous and cold medium), the spatial derivative follows the same equation of transport as the intensity does in the system:

$$\vec{\omega} \cdot \partial_1(\partial_{1,\vec{\chi}} I) = \mathcal{C}[\partial_{1,\vec{\chi}} I] \qquad \vec{x} \in \mathcal{G} \tag{6.12}$$

On the boundary of $\partial\mathcal{G}$, it composts only the black bodies. Therefore, no spatial derivative is reflected [34]. On the surface, only the sources of spatial derivative $\dot{\mathrm{S}}_{b,\vec{\chi}}$ are found on the boundaries. The index b means the source is on the "boundary".

$$\partial_{1,\vec{\chi}} I = \dot{\mathrm{S}}_{b,\vec{\chi}}[I] \qquad \vec{x} \in \partial\mathcal{G};\ \vec{\omega}\cdot\vec{n} > 0 \tag{6.13}$$

with $\dot{\mathrm{S}}_{b,\vec{\chi}}[I] = \alpha C[I] + \beta\partial_{1,\vec{u}}\mathring{I}$, leading to

$$\begin{aligned}\dot{\mathrm{S}}_{b,\vec{\chi}} = &-\alpha k_a \mathring{I} - \alpha k_s \mathring{I} \\ &+ \alpha k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x},\vec{\omega}') \\ &+ \beta\partial_{1,\vec{u}}\mathring{I}\end{aligned} \tag{6.14}$$

In Eq.6.13 and Eq.6.14, $\vec{\chi}$ is decomposed as the sum of two vectors: $\vec{\omega}$ and $\vec{u}$, which is tangent to the boundary $\partial\mathcal{G}$ (Fig.6.7):

$$\vec{\chi} = \alpha\vec{\omega} + \beta\vec{u} \tag{6.15}$$

with [34]:

$$\alpha = \frac{\vec{\chi}\cdot\vec{n}}{\vec{\omega}\cdot\vec{n}}; \beta = ||\vec{\chi} - \alpha\vec{\omega}||; \vec{u} = \frac{\vec{\chi} - \alpha\vec{\omega}}{\beta} \quad \text{or} \quad \beta\vec{u} = \frac{(\vec{\omega}\times\vec{\chi})\times\vec{n}}{\vec{\omega}\cdot\vec{n}} \tag{6.16}$$

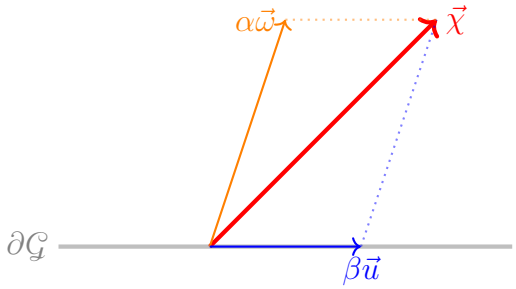


Figure 6.7 – Decomposition of the spatial velocity of deformation. The gradient of intensity projected to $\vec{\chi}$ is unknown. It is then decomposed to the direction of $\vec{\omega}$, and $\vec{u}$, where the gradients are described by RTE and the boundary condition.

In our case, the boundary is triangulated as an ensemble of plane surfaces (Fig.6.3). The intensity in a given direction becomes discontinuous at the edge because the intensity sources $\mathring{I}$ are different on the two plane surfaces. The outgoing intensity is then discontinuous when crossing the edge of two plane surfaces. The spatial derivative of outgoing intensity on the boundary requires then a Dirac formulation on the edges. When this Dirac formulation is integrated on the surface, only the integral over the edge remains. Therefore, it is considered that line emissions of the spatial gradient of intensity are on the edges $\mathcal{L}$ [34]:

$$\partial_{1,\vec{\chi}} I = \hat{\mathrm{S}}_{b,\vec{\chi}}[I] \qquad \vec{x} \in \mathcal{L} \tag{6.17}$$

If we name each plane surface with an index $i, \partial\mathcal{G}_i \in \partial\mathcal{G}, \forall i \in [1,2,\ldots,n]$, where n is the total number of the plane surfaces, each edge can be named with two indexes of

$(i, i'), \forall i \in [1, 2, \ldots, n]$, which are the two indexes of the two surfaces sharing the same edge. The line source on the edge of $\mathcal{L}_{i,i'}$ is:

$$\hat{\mathrm{S}}_{b,\vec{\chi}} = (\vec{\omega} \times \vec{\chi}) \cdot \vec{t}_i (I_i - I_{i'}) \qquad \vec{x} \in \mathcal{L}_{i,i'} \tag{6.18}$$

According to the convention made in Fig.6.8 [34]. I_i and $I_{i'}$ are the two intensity values at the edge $\mathcal{L}_{i,i'}$. $\vec{t}_i$ and $\vec{t}_{i'}$ are the two unit vectors defined locally on the two triangles and are tangent to the edge. In this chapter, we only study a convex domain (Fig.6.3). As it is discussed in Fig.4.14 in Chapter.4, when the geometry is convex ($\vec{\omega}_l \cdot \vec{n_{i'}} > 0$) the value of I_i takes the value of outgoing intensity of $\partial\mathcal{G}_i$ and the value of $I_{i'}$ takes the value of outgoing intensity of $\partial\mathcal{G}_{i'}$:

$$I_i = \lim_{\epsilon \to 0} \mathring{I}(\vec{x}_l - \epsilon \vec{m}_i, \vec{\omega}_l) \tag{6.19}$$

and

$$I_{i'} = \lim_{\epsilon \to 0} \mathring{I}(\vec{x}_l - \epsilon \vec{m}_{i'}, \vec{\omega}_l) \tag{6.20}$$

Therefore, the boundary condition on the edge $\mathcal{L}_{i,i'}$ can be separated into two parts, with the outgoing intensity I_i coming from the plane surface $\partial\mathcal{G}_i$ and $I_{i'}$ from the plane surface $\partial\mathcal{G}_{i'}$:

$$\hat{\mathrm{S}}_{b,\vec{\chi}} = (\vec{\omega} \times \vec{\chi}) \cdot \vec{t}_i I_i + (\vec{\omega} \times \vec{\chi}) \cdot \vec{t}_{i'} I_{i'} \qquad \vec{x} \in \mathcal{L}_{i,i'} \tag{6.21}$$

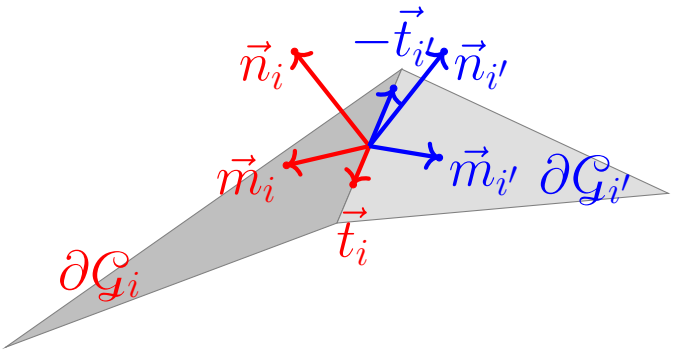


Figure 6.8 – The units vectors attached to $\partial\mathcal{G}_i$ and $\mathscr{T}_{i'}$ form two orthonormal basis: $(\vec{m}_i, \vec{t}, \vec{n}_i)$ and $(\vec{m}_{i'}, -\vec{t}, \vec{n}_{i'})$.

To sum up, we use the index p for the boundary condition on $\partial\mathcal{G}$ and the index l for the boundary conditions on $\mathcal{L}$. The model of transport of intensity becomes:

$$\begin{cases} \vec{\omega} \cdot \partial_1(\partial_{1,\vec{\chi}} I(\vec{x}_p, \vec{\omega}_p)) = C[\partial_{1,\vec{\chi}} I(\vec{x}, \vec{\omega})] \\ \partial_{1,\vec{\chi}} I(\vec{x}_p, \vec{\omega}_p) = \dot{\mathrm{S}}_{b,\vec{\chi}}(\vec{x}_p, \vec{\omega}_p) \end{cases} \tag{6.22}$$

It can be noticed that in the model of spatial derivative of intensity (Eq.6.22), the sources in the system are on the surface boundary $\partial\mathcal{G}$ and also on the line boundary $\mathcal{L}$.

If we have an overlook of the model of intensity (Eq.6.10) and the model of spatial derivative of intensity (Eq.6.22), we find that sources are located on the surface $\partial\mathcal{G}$ and on the edges $\mathcal{L}$. In the following part of this chapter, we focus on the line source $\hat{\mathrm{S}}_{b,\vec{\chi}}$, transforming it into a surface source so that only surface sources remain in the system.

6.3 Transforming line sources into surface sources with Stokes' theorem

In this stage, the model of transport for the spatial derivative is developed. Before considering $\partial_{1,\vec{\chi}} G$ in Eq.6.6, at the point $\vec{x}_{obs}$, we consider a solid angle Ω under which a subpart of the boundary $\partial\mathcal{G}$ is viewed, noted $\partial\mathcal{G}^{\Omega}$ and a subpart of the boundary $\mathcal{L}_{i,i'}$ is viewed, noted $\mathcal{L}^{\Omega}_{i,i'}$, seeing Fig.6.9. If we address the integration over Ω of the surface sources as they are viewed from $\vec{x}_{obs}$ (temporarily ignoring the extinction by absorption and scattering), each elementary solid angle $d\vec{\omega}$ defines an elementary surface $d\vec{x}_p$ at the boundary according to $d\vec{\omega} = \frac{(\vec{\omega}\cdot\vec{n})d\vec{x}_p}{r^2}$, where r is the distance between $\vec{x}_{obs}$ and $\vec{x}_p$: $r = ||\vec{x}_p - \vec{x}_{obs}||$. The angular integration becomes:

$$\begin{aligned}\int_{\Omega} \partial_{1,\vec{\chi}} I(\vec{x}_{obs},\vec{\omega}) d\vec{\omega} = &\int_{\partial\mathcal{G}^{\Omega}} \frac{(\vec{\omega}_p\cdot\vec{n})}{r^2} \dot{\mathrm{S}}_{b,\vec{\chi}} d\vec{x}_p \\ &+ \int_{\mathcal{L}^{\Omega}_{i,i'}} \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_i}{r^2} I_i d\vec{x}_l \\ &+ \int_{\mathcal{L}^{\Omega}_{i,i'}} \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_{i'}}{r^2} I_{i'} d\vec{x}_l\end{aligned} \tag{6.23}$$

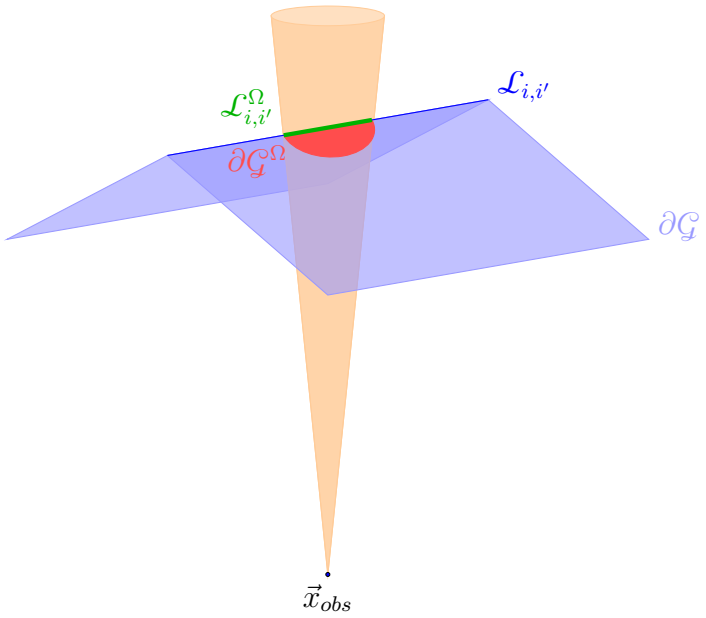


Figure 6.9 – An angular integral at $\vec{x}_{obs}$. A subpart of $\mathcal{L}_{i,i'}$ is viewed, noted $\mathcal{L}^{\Omega}_{i,i'}$. A subpart of $\partial\mathcal{G}$ is viewed, noted $\partial\mathcal{G}^{\Omega}$.

Following the same pattern, a solid angle $\Omega(\partial\mathcal{G}_i)$ can be considered. It corresponds exactly to the solid angle by which $\partial\mathcal{G}_i$ is viewed from $\vec{x}_{obs}$ (Fig.6.10). The angular integral over $\Omega(\partial\mathcal{G}_i)$ is only one part of the angular integral over 4π. Again, since the geometry is convex, the boundary $\partial\mathcal{G}$ can be totally viewed from the observation point $\vec{x}_{obs}$. Therefore, $\partial_{1,\vec{\chi}} G$ is the sum of all angular integrals over $\Omega(\partial\mathcal{G}_i), \forall i \in \{1,2,\ldots,n\}$:

$$\int_{4\pi} \partial_{1,\vec{\chi}} I(\vec{x}_{obs},\vec{\omega}) d\vec{\omega} = \sum_{i=1}^{n} \int_{\Omega(\partial\mathcal{G}_i)} \partial_{1,\vec{\chi}} I(\vec{x}_{obs},\vec{\omega}) d\vec{\omega} \tag{6.24}$$

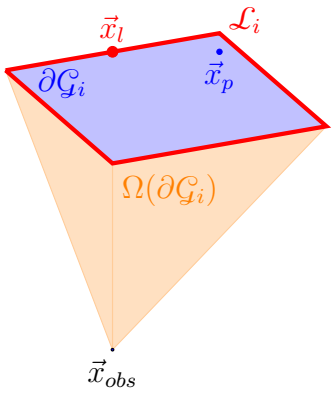


Figure 6.10 – The surface $\partial\mathcal{G}_i$ is viewed from the point $\vec{x}_{obs}$ by a solid angle $\Omega(\partial\mathcal{G}_i)$.

As it is shown in Eq.6.23 and Eq.6.21, the line source is separated into two parts, corresponding to the two plane surfaces sharing the same edge. In the angular integral over $\Omega(\partial\mathcal{G}_i)$, the part of line source corresponding to the outgoing intensity I_i on the plane surface $\partial\mathcal{G}_i$ is taken into account:

$$\begin{aligned}\int_{\Omega(\partial\mathcal{G}_i)} \partial_{1,\vec{\chi}} I(\vec{x}_{obs},\vec{\omega})d\vec{\omega} = &\int_{\partial\mathcal{G}_i} \frac{(\vec{\omega}_p\cdot\vec{n})}{r^2}\dot{\mathrm{S}}_{b,\vec{\chi}}d\vec{x}_p \\ &+ \oint_{\mathcal{L}_i} \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_i}{r^2} I_i d\vec{x}_l\end{aligned} \tag{6.25}$$

where $\mathcal{L}_i$ is the total set of the edges of the plane surface $\partial\mathcal{G}_i$.

We restrict here the analysis only to the line integral since the aim of this chapter is to transform line sources into surface sources:

$$\oint_{\mathcal{L}_i} \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_i}{r^2} I_i d\vec{x}_l \tag{6.26}$$

This line integral is along a closed contour, and Stokes' Theorem can be used to transform it into a surface integral. This application requires only an extension $\mathscr{I}_i$ of I_i to the entire surface $\partial\mathcal{G}_i$. It is chosen here to extend I_i in a straight forward way because I_i is also defined on the plane surface $\partial\mathcal{G}_i$: $\mathscr{I}_i(\vec{x}_p,\vec{\omega}_p) = I_i(\vec{x}_p,\vec{\omega}_p)$. For the outside of the surface $\partial\mathcal{G}_i$, an invariant value is taken for $\mathscr{I}_i$.

Then, we have:

$$\begin{aligned}
&\oint_{\mathcal{L}_i} \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_i}{r^2} I_i d\vec{x}_l = -\int_{\partial\mathcal{G}_i} \vec{\nabla}\times\left(\frac{\vec{\omega}_p\times\vec{\chi}}{r^2}\mathscr{I}_i\right)\cdot\vec{n}_i d\vec{x}_p \\
&= -\int_{\partial\mathcal{G}_i} \mathscr{I}_i\vec{\nabla}\times\left(\frac{(\vec{\omega}_p\times\vec{\chi})}{r^2}\right)\cdot\vec{n}_i d\vec{x}_p - \int_{\partial\mathcal{G}_i}\left[\vec{\nabla}(\mathscr{I}_i)\times(\frac{\vec{\omega}_p\times\vec{\chi}}{r^2})\right]\cdot\vec{n}_i d\vec{x}_p \\
&= -\int_{\partial\mathcal{G}_i} \mathscr{I}_i\vec{\nabla}\times\left(\frac{(\vec{\omega}_p\times\vec{\chi})}{r^2}\right)\cdot\vec{n}_i d\vec{x}_p - \int_{\partial\mathcal{G}_i}\left[(\frac{\vec{\omega}_p\times\vec{\chi}}{r^2})\times\vec{n}_i\right]\cdot\vec{\nabla}(\mathscr{I}_i)d\vec{x}_p \\
&= -\int_{\partial\mathcal{G}_i} \mathscr{I}_i\vec{\nabla}\times\left(\frac{(\vec{\omega}_p\times\vec{\chi})}{r^2}\right)\cdot\vec{n}_i d\vec{x}_p - \int_{\partial\mathcal{G}_i}\frac{(\vec{\omega}_p\cdot\vec{n}_i)\beta\vec{u}}{r^2}\cdot\vec{\nabla}(\mathscr{I}_i)d\vec{x}_p \\
&= -\int_{\partial\mathcal{G}_i} \mathscr{I}_i\vec{\nabla}\times\left(\frac{(\vec{\omega}_p\times\vec{\chi})}{r^2}\right)\cdot\vec{n}_i d\vec{x}_p - \int_{\partial\mathcal{G}_i}\frac{(\vec{\omega}_p\cdot\vec{n}_i)}{r^2}\beta\partial_{1,\vec{u}}\mathscr{I}_i d\vec{x}_p
\end{aligned} \tag{6.27}$$

In Eq.6.27, line 2 and line 3 can be referred to standard algebra; line 4 introduces the product $(\beta\vec{u})$ from Eq.6.16; line 5 introduces the definition of the spatial derivation. With $\vec{u}$ which is tangent to $\partial\mathcal{G}_i$ and $\mathscr{I}_i(\vec{x}_p,\vec{\omega}_p) = I_i(\vec{x}_p,\vec{\omega}_p)$ on $\partial\mathcal{G}_i$, we finally get:

$$\begin{aligned}\oint_{\mathcal{L}_i} \frac{(\vec{\omega}_l \times \vec{\chi})\cdot\vec{t_i}}{r^2} I_i d\vec{x}_l = &- \int_{\partial\mathcal{G}_i} I_i(\vec{x}_p,\vec{\omega}_p)\vec{\nabla}\times\left(\frac{(\vec{\omega}_p\times\vec{\chi})}{r^2}\right)\cdot\vec{n}_i d\vec{x}_p \\ &- \int_{\partial\mathcal{G}_i} \frac{(\vec{\omega}_p\cdot\vec{n}_i)}{r^2}\beta\partial_{1,\vec{u}} I_i(\vec{x}_p,\vec{\omega}_p) d\vec{x}_p\end{aligned} \tag{6.28}$$

We have made an effort to write the intensity I_i outside the term of curl. It is to build the curl term only with geometric terms ($\vec{\omega}_p$, $\vec{\chi}$, and r are geometric terms). The final line of Eq.6.28 can be applied directly to cases with a different profile of I_i. For example, when I_i is homogeneous on $\partial\mathcal{G}_i$, the second term of Eq.6.28 becomes null.

In this stage, the line sources have been transformed to surface sources via the application of Stokes' Theorem.

However, we note that the development of this section is based on temporarily ignoring the extinction by absorption and scattering. In the following sections, we will apply this development (from line sources to surface sources) to cases with different properties of the medium: transparent, absorbent, and scattering medium.

6.4 Example with a transparent medium

When the medium is transparent, the model of intensity I becomes:

$$\begin{cases}\vec{\omega}\cdot\partial_1 I(\vec{x},\vec{\omega}) = 0 \\ I(\vec{x}_p,\vec{\omega}_p) = \mathring{I}(\vec{x}_p,\vec{\omega}_p)\end{cases} \tag{6.29}$$

where the index p is used for the boundary conditions on the surfaces, and l is used for the boundary conditions on the edges.

The model of spatial derivative $\partial_{1,\vec{\chi}} I$ becomes:

$$\begin{cases}\vec{\omega}\cdot\partial_1(\partial_{1,\vec{\chi}} I(\vec{x}_p,\vec{\omega}_p)) = 0 \\ \partial_{1,\vec{\chi}} I(\vec{x}_p,\vec{\omega}_p) = \beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p,\vec{\omega}_p) \\ \partial_{1,\vec{\chi}} I(\vec{x}_l,\vec{\omega}_l) = (\vec{\omega}_l\times\vec{\chi})\cdot\vec{t_i}(I_i - I_{i'})\end{cases} \tag{6.30}$$

where i and i' are the two indexes of surfaces sharing the edge. The value of I_i and of $I_{i'}$ on the edge are expressed in Eq.6.19 and Eq.6.20. β and $\vec{u}$ are calculated based on Eq.6.16. It is noted that $\mathring{I}(\vec{x}_p,\vec{\omega}_p)$, I_i and $I_{i'}$ are all known by the boundary conditions of the model of intensity. The angular integral of intensity over 4π[sr] on the point $\vec{x}_{obs}$ is then:

$$G(\vec{x}_{obs}) = \int_{4\pi} \mathring{I}(\vec{x}_p,\vec{\omega}_p) d\vec{\omega}_p. \tag{6.31}$$

We now investigate two ways to estimate $\partial_{1,\vec{\chi}} G$. For the first one, we do not apply Stokes' theorem. The transport of surface sources and line sources are vectorized separately following the same strategy as in Chapters 4 and 5. The second one consists in applying Stokes' Theorem to transform the line sources into surface sources so that the transport of all sources can be vectorized together. Also, when we investigate to solve also the model of intensity, the sources in the model of intensity can also be vectorized together.

6.4.1 Vectorization with line and surface sources

The spatial gradient of $G(\vec{x}_{obs})$ towards $\vec{\chi}$ direction ($\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$) depends on both surface sources and the line sources:

$$\begin{aligned}\partial_{1,\vec{\chi}}G(\vec{x}_{obs}) = &\int_{\partial\mathcal{G}} \frac{(\vec{\omega}\cdot\vec{n})}{r^2}\beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p,\vec{\omega}_p)d\vec{x}_p \\ &+ \int_{\mathcal{L}} \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_i}{r^2}(I_i - I_{i'})d\vec{x}_l\end{aligned} \tag{6.32}$$

$\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$ corresponds to the gradient of the angular integral of intensity projected to the direction $\vec{\chi}$.

Similarly to what we have done in Chapter.4 and Chapter.5, we vectorize the transport of sources sharing the same path space. Therefore, the estimation of G and $\partial_{1,\vec{\chi}}G$ can be written in a vectorized form:

$$\begin{aligned}\begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}}G(\vec{x}_{obs})\end{bmatrix} = &\int_{\partial\mathcal{G}} \mathscr{P}_{\Omega_p}(\vec{\omega}_p)\begin{bmatrix}\mathring{I}(\vec{x}_p,\vec{\omega}_p) \\ \beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p,\vec{\omega}_p)\end{bmatrix}4\pi d\vec{\omega}_p \\ &+ \int_{\mathcal{L}} \mathscr{P}_{X_l}(\vec{x}_l)\begin{bmatrix}0 \\ \frac{(\vec{\omega}_l\times\vec{\chi})\cdot\vec{t}_i}{r^2}(I_i - I_{i'})\end{bmatrix}\mathcal{L}_{tot}d\vec{x}_l\end{aligned} \tag{6.33}$$

with the pdf:

$$\mathscr{P}_{\Omega_p}(\vec{\omega}_p) = \frac{1}{4\pi}; \mathscr{P}_{X_l}(\vec{x}_l) = \frac{1}{\mathcal{L}_{tot}} \tag{6.34}$$

where $\mathcal{L}_{tot}$ is the total length of all edges. The integral formulation Eq.6.33 can be seen as a Monte-Carlo algorithm. It could be noticed that the observable $G(\vec{x}_{obs})$ only requires a surface integration. Therefore the Monte Carlo weight of the line integral is null. On the other hand, the spatial gradient of the observable $\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$ is estimated through both surface and line integrations. Mathematically, two integral with different intervals appears. Physically, two path spaces are found for the transport of intensity and its spatial derivative. The corresponding Monte-Carlo algorithm is the Algo.8.

Algorithm 8 Evaluation of G and $\partial_{1,\vec{\chi}}G$ with line sources in transparent medium

Input: $\mathscr{G}$, n_{MC}, $\vec{x}_{obs}$

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{x}_{obs}$: the observation point.

1: $W_1 \leftarrow 0$, $W_1 \leftarrow 0$ and $V_2 \leftarrow 0$, $V_2 \leftarrow 0$ ▷ Initialize the sum and the sum of square
2: **for** $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ First Monte-Carlo for surface sources.
3: Initialize the weight of Monte-Carlo $w_1 \leftarrow 0$ and $w_2 \leftarrow 0$
4: Sample uniformly a direction $\vec{\omega}$
5: Get the intersection point of the ray $(\vec{x}_{obs},\vec{\omega})$ on $\partial\mathcal{G}$: $\vec{x}_{hit}$
6: Get the normal $\vec{n}$ on the point $\vec{x}_{hit}$
7: $\vec{x}_p \leftarrow \vec{x}_{hit}$
8: $\vec{\omega}_p \leftarrow -\vec{\omega}$
9: Calculate the length of optical path $r \leftarrow ||\vec{x}_p - \vec{x}_{obs}||$
10: Compute the weight $w_1 \leftarrow 4\pi\mathring{I}(\vec{x}_p,\vec{\omega}_p)$
11: Compute the weight $w_2 \leftarrow \beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p,\vec{\omega}_p)$

$W_1 \leftarrow W_1 + w_1$
$V_1 \leftarrow V_1 + w_1^2$
$W_2 \leftarrow W_2 + w_2$
$V_2 \leftarrow V_2 + w_2^2$
$i_{MC} \leftarrow i_{MC} + 1$
end for
for $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ Second Monte-Carlo for line sources.
Initialize the weight of Monte-Carlo $w_2 \leftarrow 0$
Sample uniformly a position $\vec{x}_l$ on $\mathscr{L}$
Get the intensity of two surfaces sharing the sampled point $\vec{x}_l$: I_i and $I_{i'}$
Calculate the length of optical path $r \leftarrow ||\vec{x}_l - \vec{x}_{obs}||$
Get $\vec{t}_i$
Calculate the direction $\vec{\omega}_p = \frac{\vec{x}_{obs} - \vec{x}_l}{||\vec{x}_{obs} - \vec{x}_l||}$
Compute the weight $w_2 \leftarrow \frac{(\vec{\omega}_p \times \vec{\chi}) \cdot \vec{t}_i}{r^2}(I_i - I_{i'})$
$W_2 \leftarrow W_2 + w_2$
$V_2 \leftarrow V_2 + w_2^2$
$i_{MC} \leftarrow i_{MC} + 1$
end for

Output: $G \leftarrow \frac{W_1}{n_{MC}}$, $\sigma[G] \leftarrow \sqrt{\frac{\frac{V_1}{n_{MC}} - (\frac{W_1}{n_{MC}})^2}{n_{MC}-1}}$, $\partial_{1,\vec{\chi}} G \leftarrow \frac{W_2}{n_{MC}}$, $\sigma[\partial_{1,\vec{\chi}} G] \leftarrow \sqrt{\frac{\frac{V_2}{n_{MC}} - (\frac{W_2}{n_{MC}})^2}{n_{MC}-1}}$

6.4.2 Vectorization with surface sources only

As it is shown in Eq.6.24, $\partial_{1,\vec{\chi}} G$ is also the sum of all angular integral over a solid angle $\Omega(\partial\mathcal{G}_i)$, which is already developed in Eq.6.25. We now substitute the result of the development in Eq.6.28 (from line sources to surface sources) into Eq.6.25 and we have:

$$\int_{\Omega(\partial\mathcal{G}_i)} \partial_{1,\vec{\chi}} I(\vec{x}_{obs}, \vec{\omega}) d\vec{\omega} = -\int_{\partial\mathcal{G}_i} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \left[\vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2}\right) \cdot \vec{n}_i\right] d\vec{x}_p \tag{6.35}$$

We substitute Eq.6.35 into Eq.6.24:

$$\partial_{1,\vec{\chi}} G(\vec{x}_{obs}) = -\int_{\partial\mathcal{G}} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \left[\vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2}\right) \cdot \vec{n}_i\right] d\vec{x}_p \tag{6.36}$$

Eq.6.36 is a spatial integral, while Eq.6.31 is an angular integral. In order to have the same interval for the two integrals, we choose to change Eq.6.36 into an angular integral. It is noted that it is also possible to change the angular integral into spatial integral. It is chosen to use angular integral here because it is more common in the community of radiative transfer. We remind that the solid angle is $d\vec{\omega} = \frac{(\vec{\omega} \cdot \vec{n}) d\vec{x}_p}{r^2}$:

$$\partial_{1,\vec{\chi}} G(\vec{x}_{obs}) = -\int_{4\pi} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \left[\vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2}\right) \cdot \vec{n}\right] \frac{r^2}{(\vec{\omega}_p \cdot \vec{n})} d\vec{\omega}_p \tag{6.37}$$

Finally, the value G and its spatial derivative $\partial_{1,\vec{\chi}} G$ share a same interval. Physically, they share the same path space. Therefore, their estimations can be vectorized:

$$\begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}} G(\vec{x}_{obs}) \end{bmatrix} = \int_{4\pi} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p \begin{bmatrix} 1 \\ \left(\vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2}\right) \cdot \vec{n}\right) \frac{r^2}{(\vec{\omega}_p \cdot \vec{n})} \end{bmatrix} 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p) \tag{6.38}$$

with the pdf $\mathscr{P}_{\Omega_p}(\vec{\omega}_p)$ already shown in Eq.6.34.

Eq.6.38 can then be translated into a Monte-Carlo algorithm. In order to estimate $G(\vec{x}_{obs})$, we sum all the source of intensity coming from the full solid angle 4π[sr]. The sources of intensity are on the surfaces. Concerning its spatial derivative $\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$, we sum also all the source of spatial derivative coming form the full solid angle 4π[sr]. The sources of the spatial derivative are also on the surfaces. Therefore, the transport of the sources of intensity and spatial derivative can be vectorized. The corresponding algorithm is the Algo.9.

To sum up, Eq.6.38 and Eq.6.33 represent two algorithms, which are both dedicated to estimate the observable $G(\vec{x}_{obs})$ and its spatial derivative $\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$ in a vectorized form. We will discuss their differences here.

In Eq.6.33, both the surface integral and line integral exist. They are the integrals of surface sources and line sources. The surface sources and the line sources have different integral domains, which means different path spaces. Therefore, from an algorithmic point of view, a Monte-Carlo estimation is needed for the surface sources and another for the line sources.

In Eq.6.38, only the integral of surface sources exists. It is because of the efforts that are made to transform the line sources in Eq.6.33 into surface sources. Therefore, from an algorithmic point of view, only one Monte-Carlo estimation is needed for all the surface sources because they now all share the same path space (or, mathematically, they share the same integral domain.)

Finally, by transforming Eq.6.33 to Eq.6.38, we optimized the algorithm of estimating $G(\vec{x}_{obs})$ and $\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$. We are successful to firstly avoid sampling on a line and secondly, vectorized completely the estimation of $G(\vec{x}_{obs})$ and $\partial_{1,\vec{\chi}}G(\vec{x}_{obs})$. Our strategy of transforming line sources into surface sources works well for transparent mediums.

Algorithm 9 Evaluation of G and $\partial_{1,\vec{\chi}}G$ with surface sources in transparent medium

---

Input: $\mathscr{G}$, n_{MC}, $\vec{x}_{obs}$

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{x}_{obs}$: the observation point.

1: $W_1 \leftarrow 0$, $W_1 \leftarrow 0$ and $V_2 \leftarrow 0$, $V_2 \leftarrow 0$
▷ Initialize the sum and the sum of square
2: **for** $i_{MC} = 0 \rightarrow n_{MC}$ **do**
3: Initialize the weight of Monte-Carlo $w_1 \leftarrow 0$ and $w_2 \leftarrow 0$
4: Sample uniformly a direction $\vec{\omega}$
5: Get the intersection point of the ray $(\vec{x}_{obs}, \vec{\omega})$ on $\partial\mathcal{G}$: $\vec{x}_{hit}$
6: Get the normal $\vec{n}$ on the point $\vec{x}_{hit}$
7: $\vec{x}_p \leftarrow \vec{x}_{hit}$
8: $\vec{\omega}_p \leftarrow -\vec{\omega}$
9: Calculate the length of optical path $r \leftarrow ||\vec{x}_p - \vec{x}_{obs}||$
10: Compute the weight $w_1 \leftarrow 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p)$
11: Compute the weight $w_2 \leftarrow 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p) \left(\vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2}\right) \cdot \vec{n}\right) \frac{r^2}{(\vec{\omega}_p \cdot \vec{n})}$
12: $W_1 \leftarrow W_1 + w_1$
13: $V_1 \leftarrow V_1 + w_1^2$
14: $W_2 \leftarrow W_2 + w_2$
15: $V_2 \leftarrow V_2 + w_2^2$
16: $i_{MC} \leftarrow i_{MC} + 1$
17: **end for**

Output: $G \leftarrow \frac{W_1}{n_{MC}}$, $\sigma[G] \leftarrow \sqrt{\frac{\frac{V_1}{n_{MC}} - (\frac{W_1}{n_{MC}})^2}{n_{MC}-1}}$, $\partial_{1,\vec{\chi}}G \leftarrow \frac{W_2}{n_{MC}}$, $\sigma[\partial_{1,\vec{\chi}}G] \leftarrow \sqrt{\frac{\frac{V_2}{n_{MC}} - (\frac{W_2}{n_{MC}})^2}{n_{MC}-1}}$

6.5 Example with an absorbent medium

When the medium is purely absorbent and described by a constant absorption coefficient k_a, the model of intensity I becomes:

$$\begin{cases} \vec{\omega} \cdot \partial_1 I(\vec{x}, \vec{\omega}) = -k_a I(\vec{x}, \vec{\omega}) \\ I(\vec{x}_p, \vec{\omega}_p) = \mathring{I}(\vec{x}_p, \vec{\omega}_p) \end{cases} \tag{6.39}$$

where the index p is used for the boundary conditions on the surfaces, and l is used for the boundary conditions on the edges.

The model of spatial derivative $\partial_{1,\vec{\chi}} I$ becomes:

$$\begin{cases} \vec{\omega} \cdot \partial_1 (\partial_{1,\vec{\chi}} I(\vec{x}_p, \vec{\omega}_p)) = -k_a \partial_{1,\vec{\chi}} I(\vec{x}, \vec{\omega}) \\ \partial_{1,\vec{\chi}} I(\vec{x}_p, \vec{\omega}_p) = -\alpha k_a \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \\ \partial_{1,\vec{\chi}} I(\vec{x}_l, \vec{\omega}_l) = (\vec{\omega}_l \times \vec{\chi}) \cdot \vec{t}_i (I_i - I_{i'}) \end{cases} \tag{6.40}$$

where i and i' are the two indices of surfaces sharing the edge. The value of I_i and of $I_{i'}$ can be referred to Eq.6.19 and Eq.6.20. β and $\vec{u}$ can be referred to Eq.6.16. The model of intensity can be solved following the process in Chapter.3 (by formulating a Fredholm equation and then writing in an expected value form). The solution of G is then:

$$G(\vec{x}_{obs}) = \int_{4\pi} \mathring{I}(\vec{x}_p, \vec{\omega}_p) e^{-k_a r} d\vec{\omega}_p. \tag{6.41}$$

where r is the length of the optical path: $r = ||\vec{x}_{obs} - \vec{x}_p||$. Compared to Eq.6.31, the exponential attenuation by absorption $e^{-k_a r}$ is taken into account in Eq.6.41. As it is done for the transparent medium in the last section, we will also provide the solution of $\partial_{1,\vec{\chi}} G$ with and without the application of Stokes' Theorem.

6.5.1 Vectorization with line and surface sources

Compared to the model of spatial derivative in a transparent model Eq.6.30, a surface source $-\alpha k_a \mathring{I}(\vec{x}_p, \vec{\omega}_p)$ appears on the boundary conditions of the model in absorbent medium. Also, the exponential attenuation due to the absorption along the optical path is also taken into account. Finally, we vectorize the transport of the sources sharing the same path space, and the following integral formulation is obtained:

$$\begin{aligned} \begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}} G(\vec{x}_{obs}) \end{bmatrix} = & \int_{\partial\mathcal{G}} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) \begin{bmatrix} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \\ -\alpha k_a \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \end{bmatrix} 4\pi e^{-k_a r} d\vec{\omega}_p \\ & + \int_{\mathcal{L}} \mathscr{P}_{X_l}(\vec{x}_l) \begin{bmatrix} 0 \\ \frac{(\vec{\omega}_l \times \vec{\chi}) \cdot \vec{t}_i}{r^2} (I_i - I_{i'}) \end{bmatrix} \mathcal{L}_{tot} e^{-k_a r} d\vec{x}_l \end{aligned} \tag{6.42}$$

with the pdf $\mathscr{P}_{\Omega_p}$ and $\mathscr{P}_{X_l}$ already shown in Eq.6.34. In fact, the algorithm of an absorbent medium with line sources is similar to the case of a transparent medium. Compared to Eq.6.33, only the factor of exponential attenuation $e^{-k_a r}$ is added in Eq.6.42. If we have $k_a = 0$, the medium becomes transparent, and the algorithm would be exactly the same as the case of a transparent medium.

The corresponding Monte-Carlo algorithm is the Algo.10.

Algorithm 10 Evaluation of G and $\partial_{1,\vec{\chi}} G$ with line sources in absorbent medium

Input: $\mathscr{G}$, n_{MC}, $\vec{x}_{obs}$, k_a

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{x}_{obs}$: the observation point.
▷ $\vec{k}_a$: coefficient of absorption.

$W_1 \leftarrow 0$, $W_1 \leftarrow 0$ and $V_2 \leftarrow 0$, $V_2 \leftarrow 0$
▷ Initialize the sum and the sum of square
for $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ First Monte-Carlo for surface sources.
  Initialize the weight of Monte-Carlo $w_1 \leftarrow 0$ and $w_2 \leftarrow 0$
  Sample uniformly a direction $\vec{\omega}$
  Get the intersection point of the ray $(\vec{x}_{obs}, \vec{\omega})$ on $\partial\mathcal{G}$: $\vec{x}_{hit}$
  Get the normal $\vec{n}$ on the point $\vec{x}_{hit}$
  $\vec{x}_p \leftarrow \vec{x}_{hit}$
  $\vec{\omega}_p \leftarrow -\vec{\omega}$
  Calculate the length of optical path $r \leftarrow ||\vec{x}_p - \vec{x}_{obs}||$
  Compute the weight $w_1 \leftarrow 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p) e^{-k_a r}$
  Compute the weight $w_2 \leftarrow (-\alpha k_a \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p)) \mathscr{L}_{tot} e^{-k_a r}$
  $W_1 \leftarrow W_1 + w_1$
  $V_1 \leftarrow V_1 + w_1^2$
  $W_2 \leftarrow W_2 + w_2$
  $V_2 \leftarrow V_2 + w_2^2$
  $i_{MC} \leftarrow i_{MC} + 1$
end for
for $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ Second Monte-Carlo for line sources.
  Initialize the weight of Monte-Carlo $w_2 \leftarrow 0$
  Sample uniformly a position $\vec{x}_l$ on $\mathscr{L}$
  Get the intensity of two surfaces sharing the sampled point $\vec{x}_l$: I_i and $I_{i'}$
  Get $\vec{t}_i$
  Calculate the direction $\vec{\omega}_p = \frac{\vec{x}_{obs} - \vec{x}_l}{||\vec{x}_{obs} - \vec{x}_l||}$
  Calculate the length of optical path $r \leftarrow ||\vec{x}_l - \vec{x}_{obs}||$
  Compute the weight $w_2 \leftarrow \frac{(\vec{\omega}_p \times \vec{\chi}) \cdot \vec{t}_i}{r^2} (I_i - I_{i'})$
  $W_2 \leftarrow W_2 + w_2$
  $V_2 \leftarrow V_2 + w_2^2$
  $i_{MC} \leftarrow i_{MC} + 1$
end for

Output: $G \leftarrow \frac{W_1}{n_{MC}}$, $\sigma[G] \leftarrow \sqrt{\frac{\frac{V_1}{n_{MC}} - (\frac{W_1}{n_{MC}})^2}{n_{MC}-1}}$, $\partial_{1,\vec{\chi}} G \leftarrow \frac{W_2}{n_{MC}}$, $\sigma[\partial_{1,\vec{\chi}} G] \leftarrow \sqrt{\frac{\frac{V_2}{n_{MC}} - (\frac{W_2}{n_{MC}})^2}{n_{MC}-1}}$

6.5.2 Vectorization with surface sources only

The angular integral of $\partial_{1,\vec{\chi}} I$ over $\Omega(\partial\mathcal{G}_i)$ is developed in Eq.6.25 without considering the absorption attenuation. When the absorption in the medium is taken into account, Eq.6.25 becomes:

$$\begin{aligned}\int_{\Omega(\partial\mathcal{G}_i)} \partial_{1,\vec{\chi}} I(\vec{x}_{obs}, \vec{\omega}) d\vec{\omega} = &\int_{\partial\mathcal{G}_i} \frac{(\vec{\omega}_p \cdot \vec{n})}{r^2} \left(-\alpha k_a \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p)\right) e^{-k_a r} d\vec{x}_p \\ &+ \oint_{\mathscr{L}_i} \frac{(\vec{\omega}_l \times \vec{\chi}) \cdot \vec{t}_i}{r^2} I_i e^{-k_a r} d\vec{x}_l\end{aligned} \tag{6.43}$$

Herein, as we have done in Section.6.3, we only focus on the analysis of the line integral, taking the absorption attenuation into account:

$$\oint_{\mathcal{L}_i} \frac{(\vec{\omega}_l \times \vec{\chi}) \cdot \vec{t}_i}{r^2} I_i e^{-k_a r} d\vec{x}_l \tag{6.44}$$

Following the same extension $\mathscr{I}_i$ of I_i (as it is chosen in Section.6.3) from the edges to the entire surface, we apply Stokes' Theorem to Eq.6.44:

$$\begin{aligned} \oint_{\mathcal{L}_i} \frac{(\vec{\omega}_l \times \vec{\chi}) \cdot \vec{t}_i}{r^2} I_i e^{-k_a r} d\vec{x}_l = &- \int_{\partial \mathcal{G}_i} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2} e^{-k_a r} \right) \cdot \vec{n}_i d\vec{x}_p \\ &- \int_{\partial \mathcal{G}_i} \frac{(\vec{\omega}_p \cdot \vec{n}_i)}{r^2} \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p) e^{-k_a r} d\vec{x}_p \end{aligned} \tag{6.45}$$

We then substitute Eq.6.45 to Eq.6.43 and the following integral is yielded:

$$\begin{aligned} \int_{\Omega(\partial \mathcal{G}_i)} \partial_{1,\vec{\chi}} I(\vec{x}_{obs}, \vec{\omega}) d\vec{\omega} = &\int_{\partial \mathcal{G}_i} \frac{(\vec{\omega}_p \cdot \vec{n})}{r^2} \left(-\alpha k_a \mathring{I}(\vec{x}_p, \vec{\omega}_p) \right) e^{-k_a r} d\vec{x}_p \\ &- \int_{\partial \mathcal{G}_i} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2} e^{-k_a r} \right) \cdot \vec{n}_i d\vec{x}_p \end{aligned} \tag{6.46}$$

After that, we extend this angular integral over $\Omega(\partial \mathcal{G}_i)$ to 4π, following Eq.6.24. The angular integral of $\partial_{1,\vec{\chi}} G$ without line sources in an absorbent medium is yielded:

$$\partial_{1,\vec{\chi}} G(\vec{x}_{obs}) = - \int_{4\pi} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \left(\alpha k_a e^{-k_a r} + \vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2} e^{-k_a r} \right) \cdot \vec{n} \frac{r^2}{(\vec{\omega}_p \cdot \vec{n})} \right) d\vec{\omega}_p \tag{6.47}$$

Finally, G and $\partial_{1,\vec{\chi}} G$ can be written in a vectorized form:

$$\begin{aligned} \begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}} G(\vec{x}_{obs}) \end{bmatrix} = &\int_{4\pi} \mathscr{P}_{\Omega_p}(\vec{\omega}_p) d\vec{\omega}_p \\ &\begin{bmatrix} e^{-k_a r} \\ \alpha k_a e^{-k_a r} + \vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2} e^{-k_a r} \right) \cdot \vec{n} \frac{r^2}{(\vec{\omega}_p \cdot \vec{n})} \end{bmatrix} 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p) \end{aligned} \tag{6.48}$$

with the pdf $\mathscr{P}_{\Omega_p}$ already defined in Eq.6.34.

Finally, the algorithm represented by Eq.6.48 is similar to that represented by Eq.6.38. If $k_a = 0$, the medium becomes transparent, and they become exactly the same. The minor differences between these two algorithms are that, firstly, the curl is applied to two different formulations. In the case of an absorbent medium, the curl operator is applied to a term with exponential attenuation. Also, an extra surface source $\alpha k_a e^{-k_a r}$ appears where the formulation of α is referred to Eq.6.16. The corresponding Monte-Carlo algorithm is the Algo.11

Algorithm 11 Evaluation of G and $\partial_{1,\vec{\chi}} G$ with surface sources in absorbent medium

Input: $\mathscr{G}$, n_{MC}, $\vec{x}_{obs}$, k_a

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{x}_{obs}$: the observation point.
▷ $\vec{k}_a$: coefficient of absorption.

$W_1 \leftarrow 0$, $W_1 \leftarrow 0$ and $V_2 \leftarrow 0$, $V_2 \leftarrow 0$
▷ Initialize the sum and the sum of square
for $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ First Monte-Carlo for surface sources.
  Initialize the weight of Monte-Carlo $w_1 \leftarrow 0$ and $w_2 \leftarrow 0$
  Sample uniformly a direction $\vec{\omega}$
  Get the intersection point of the ray $(\vec{x}_{obs}, \vec{\omega})$ on $\partial\mathcal{G}$: $\vec{x}_{hit}$
  Get the normal $\vec{n}$ on the point $\vec{x}_{hit}$
  $\vec{x}_p \leftarrow \vec{x}_{hit}$
  $\vec{\omega}_p \leftarrow -\vec{\omega}$
  Calculate the length of optical path $r \leftarrow ||\vec{x}_p - \vec{x}_{obs}||$
  Compute the weight $w_1 \leftarrow 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p) e^{-k_a r}$
  Compute the weight
    $w_2 \leftarrow \left(\alpha k_a e^{-k_a r} + \vec{\nabla} \times \left(\frac{(\vec{\omega}_p \times \vec{\chi})}{r^2} e^{-k_a r}\right) \cdot \vec{n} \frac{r^2}{(\vec{\omega}_p \cdot \vec{n})}\right) 4\pi \mathring{I}(\vec{x}_p, \vec{\omega}_p)$
  $W_1 \leftarrow W_1 + w_1$
  $V_1 \leftarrow V_1 + w_1^2$
  $W_2 \leftarrow W_2 + w_2$
  $V_2 \leftarrow V_2 + w_2^2$
  $i_{MC} \leftarrow i_{MC} + 1$
end for

Output: $G \leftarrow \frac{W_1}{n_{MC}}$, $\sigma[G] \leftarrow \sqrt{\frac{\frac{V_1}{n_{MC}} - (\frac{W_1}{n_{MC}})^2}{n_{MC}-1}}$, $\partial_{1,\vec{\chi}} G \leftarrow \frac{W_2}{n_{MC}}$, $\sigma[\partial_{1,\vec{\chi}} G] \leftarrow \sqrt{\frac{\frac{V_2}{n_{MC}} - (\frac{W_2}{n_{MC}})^2}{n_{MC}-1}}$

6.6 Example with a scattering medium

When the medium is purely diffusive, described by a constant scattering coefficient k_s, the model of intensity I becomes:

$$\begin{cases} \vec{\omega} \cdot \partial_1 I(\vec{x}, \vec{\omega}) = -k_s I(\vec{x}, \vec{\omega}) + k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}') \\ I(\vec{x}_p, \vec{\omega}_p) = \mathring{I}(\vec{x}_p, \vec{\omega}_p) \end{cases} \tag{6.49}$$

The model of spatial derivative $\partial_{1,\vec{\chi}} I$ becomes:

$$\begin{cases} \vec{\omega} \cdot \partial_1 (\partial_{1,\vec{\chi}} I(\vec{x}_p, \vec{\omega}_p)) = -k_s I(\vec{x}, \vec{\omega}) + k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}') \\ \partial_{1,\vec{\chi}} I(\vec{x}_p, \vec{\omega}_p) = -\alpha k_s \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \alpha k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}') + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p) \\ \partial_{1,\vec{\chi}} I(\vec{x}_l, \vec{\omega}_l) = (\vec{\omega}_l \times \vec{\chi}) \cdot \vec{t}_i (I_i - I_{i'}) \end{cases} \tag{6.50}$$

The solution of G is then:

$$G(\vec{x}_{obs}) = \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_0 \int_0^\infty k_s e^{-k_s \xi_0} d\xi_0 \Big\{ \mathcal{H}(\xi_0 - d_0) \mathring{I}(\vec{x}_{hit,0}, \vec{\omega}_0) 4\pi + \mathcal{H}(d_0 - \xi_0) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^\infty k_s e^{-k_s \xi_1} d\xi_1 \Big\{ \mathcal{H}(\xi_1 - d_1) \mathring{I}(\vec{x}_{hit,1}, \vec{\omega}_1) 4\pi + \mathcal{H}(d_1 - \xi_1) \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \ldots \Big\} \Big\} \quad (6.51)$$

where ξ_0 represents the path length of the photon before its 1^{st} scattering and ξ_1 represent the path length of the photon after its 1^{st} scattering; $\vec{\omega}_0$ is the direction of propagation of the photon before its 1^{st} scattering, and $\vec{\omega}_1$ and $\vec{\omega}_2$ represent the direction of propagation of the photon respectively after its 1^{st} and 2^{nd} scattering; d_0 represents the distance that the photon needs to go through to hit a boundary without scattering; d_1 represents the distance that, starting from the position where located the 1^{st} scattering, the photon need to go through to hit a boundary without 2^{nd} scattering. $\vec{x}_{hit,0}$ represents the hit point without scattering, and $\vec{x}_{hit,1}$ represents the hit point after one scattering.

As it is discussed in the previous chapters, a "random walk" algorithm corresponding to a physical picture is attached to Eq.6.51. G can be regarded as the sum of intensity carried by all photons coming from all directions (angular integration over 4π). In order to determine the intensity coming from one direction $\vec{\omega}_0$, we will inversely trace the arriving photons at the position, $\vec{x}_{obs}$ in the direction $\vec{\omega}_0$, searching from where they are coming. We will sample a path length according to the exponential law in the direction $-\vec{\omega}_0$ to determine the position where scattering happens. If the sampled path length is larger than the distance to the boundary following the direction of $-\vec{\omega}_0$, we consider that the photon is not scattered. In this case (the Heaviside function $\mathcal{H}(\xi_0 - d_0)$ will be true), the photon comes from the hit-point $\vec{x}_{hit,0}$ on the boundary, following the direction of $\vec{\omega}_0$. Otherwise, (the Heaviside function $\mathcal{H}(d_0 - \xi_0)$ will be true), we sample a direction of scattering and then another sample of the path length to determine the next scattering point. We repeat this process again and again until the photon reaches the boundary.

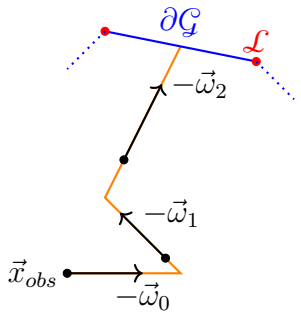


Figure 6.11 – After the random walks in the medium with a reverse point of view, a photon can hit a surface boundary $\partial\mathcal{G}$, but it can never hit a line boundary $\mathcal{L}$.

However, following the above algorithm, the photon can only intersect the surface $\partial\mathcal{G}$, but it can not intersect an edge $\mathcal{L}$ (seeing Fig.6.11). As a consequence, this "random walks" algorithm can not be implemented directly to the model of spatial derivative because the line sources on $\mathcal{L}$ can not be reached. In the following part of this section, we will implement a special treatment to deal with this problem.

6.6.1 Vectorization with line and surface sources

The line sources can be accumulated using the techniques developed to handle collimated Dirac sources for solar/laser applications or satellite observation [5, 11, 16, 52, 57]. At each scattering event, the directions of the Dirac sources are first sampled, specifically, before continuing the path in another sampled reflected or scattered direction. We apply this algorithm in our case in this section.

In the model of spatial gradient (Eq.6.50), sources are found on the surfaces $\partial\mathcal{G}$ and on the edges $\mathcal{L}$. We separate here the two contributions of $\partial_{1,\vec{\chi}}G$, corresponding to the sum of the two kinds of source, respectively:

$$\partial_{1,\vec{\chi}}G(\vec{x}_{obs}) = (\overset{\bullet}{\partial_{1,\vec{\chi}}G})(\vec{x}_{obs}) + (\widehat{\partial_{1,\vec{\chi}}G})(\vec{x}_{obs}) \tag{6.52}$$

For the surface source $(\overset{\bullet}{\partial_{1,\vec{\chi}}G})$, an integral formulation similar to Eq.6.51 can be built. Eq.6.51 and Eq.6.53 have the same integrals domains and pdf. The only difference is the surface sources on the boundary. Therefore, the surface sources in Eq.6.51 and in Eq.6.53 can be accumulated by the same "random walks" algorithm (same as Fig.6.11):

$$\begin{aligned}(\overset{\bullet}{\partial_{1,\vec{\chi}}G})(\vec{x}_{obs}) = \int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_0\int_0^\infty k_s e^{-k_s\xi_0}d\xi_0\Big\{&\mathcal{H}(\xi_0-d_0)\dot{\mathrm{S}}^{scat}_{b,\vec{\chi}}(\vec{x}_{hit,0},\vec{\omega}_0)4\pi+\\ \mathcal{H}(d_0-\xi_0)\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_1\int_0^\infty k_s e^{-k_s\xi_1}d\xi_1\Big\{&\mathcal{H}(\xi_1-d_1)\dot{\mathrm{S}}^{scat}_{b,\vec{\chi}}(\vec{x}_{hit,1},\vec{\omega}_1)4\pi+\\ &\mathcal{H}(d_1-\xi_1)\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_2\ldots\Big\}\Big\}\end{aligned} \tag{6.53}$$

where $\dot{\mathrm{S}}^{scat}_{b,\vec{\chi}}$ is the surface source in Eq.6.50:

$$\dot{\mathrm{S}}^{scat}_{b,\vec{\chi}}(\vec{x}_p,\vec{\omega}_p) = -\alpha k_s\mathring{I}(\vec{x}_p,\vec{\omega}_p) + \alpha k_s\int_{4\pi}\frac{1}{4\pi}d\vec{\omega}' I(\vec{x},\vec{\omega}') + \beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p,\vec{\omega}_p) \tag{6.54}$$

It is noticed that the physical image and the attached algorithm of Eq.6.53 are very similar to those attached to Eq.6.51. The only subtlety is that Eq.6.54 is coupled with the model of intensity (via $I(\vec{x},\vec{\omega}')$). Therefore, a new optical path is generated, which represents the coupling with the intensity of the boundary condition. The coupling is done through the double randomization process as shown in Fig.6.12.

However, the estimation of the sum of line source $(\widehat{\partial_{1,\vec{\chi}}G})$ is not that straightforward. The line sources are emitted on the boundary $\mathcal{L}$. They arrive at the point $\vec{x}_{obs}$ after different numbers of scattering in the medium. The sum of these arrival sources after 0 scattering, 1 scattering, ..., infinitive scatterings, is then $(\widehat{\partial_{1,\vec{\chi}}G})$. For example, Fig.6.13 shows an optical path of a line source, arriving at the point $\vec{x}_{obs}$, following $\vec{\omega}_0$, after two scattering events. Herein, we classify it following the number of scattering that happens before arriving at $\vec{x}_{obs}$, $\forall j\in\mathbb{N}$:

$$(\widehat{\partial_{1,\vec{\chi}}G})(\vec{x}_{obs}) = \sum_{j=0}^{+\infty}(\widehat{\partial_{1,\vec{\chi}}G})_j(\vec{x}_{obs}) \tag{6.55}$$

where j is the index of the number of scattering. $(\widehat{\partial_{1,\vec{\chi}}G})_j$ is then the sources of spatial derivative on the boundary $\mathcal{L}$ arriving to the point $\vec{x}_{obs}$, after j times of scattering:

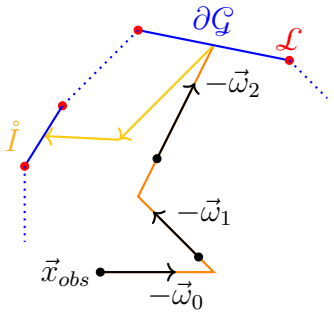


Figure 6.12 – The boundary condition of the spatial derivative on the surface $\dot{\mathrm{S}}^{scat}_{b,\vec{\chi}}$ is coupled with the model of intensity. A supplementary optical path (in yellow) is generated on the hit point to search for the boundary condition of I and complete the boundary condition of $\dot{\mathrm{S}}^{scat}_{b,\vec{\chi}}$. To be reminded that the source of intensity is noted $\mathring{I}$

$$\widehat{(\partial_{1,\vec{\chi}}G)}_j(\vec{x}_{obs}) = \int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_0\int_0^{d_0} k_s e^{-k_s\xi_0}d\xi_0\ldots \int_{4\pi}\frac{1}{4\pi}d\vec{\omega}_{j-1}\int_0^{d_{j-1}} k_s e^{-k_s\xi_{j-1}}d\xi_{j-1} \int_{\mathcal{L}} d\vec{x}_{l_j}\hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j,\vec{x}_{l_j}) \quad (6.56)$$

where

$$\hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j,\vec{x}_{l_j}) = \frac{(\vec{\omega}_{l_j}\times\vec{\chi})\cdot\vec{t}_i}{r_{l_j}^2}(I_i - I_{i'})e^{-k_s r_{l_j}}\frac{1}{4\pi}, \quad (6.57)$$

$$r_{l_j} = ||\vec{x}_j - \vec{x}_{l_j}||; \vec{\omega}_{l_j} = \frac{\vec{x}_j - \vec{x}_{l_j}}{r_{l_j}} \quad (6.58)$$

and

$$\vec{x}_j = \sum_{k=0}^{j-1}\vec{x}_{obs} + \vec{\omega}_k\xi_k, \forall j > 0; \vec{x}_0 = \vec{x}_{obs} \quad (6.59)$$

We trace inversely the path of a photon that arrives at $\vec{x}_{obs}$ after j scatterings. $\vec{x}_j$ is the position of the photon where j^{th} scattering happens; $\vec{x}_{l_j}$ is on the edge $\mathcal{L}$; r_{l_j} is the distance between $\vec{x}_{l_j}$ and $\vec{x}_j$; $\vec{\omega}_{l,j}$ is the vector of direction linking $\vec{x}_j$ and $\vec{x}_{l_j}$; $\hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}$ is the source emitted from $\vec{x}_{l_j}$, arriving to $\vec{x}_j$, which is attenuated (exponentially) along the optical path because of the scattering. For example, a schema of the contribution $\widehat{(\partial_{1,\vec{\chi}}G)}_j|_{j=2}$ is shown on Fig.6.13.

After that, we substitute Eq.6.56 into Eq.6.55 and we reformulate it as an expected value:

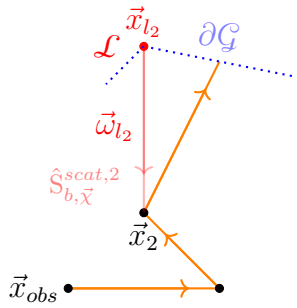


Figure 6.13 – A schema of $\widehat{(\partial_{1,\vec{\chi}}G)}_j|_{j=2}$. The line source $\hat{\mathrm{S}}^{scat,2}_{b,\vec{\chi}}$ arrives at $\vec{x}_{obs}$ after 2 scattering events.

$$\begin{aligned}\widehat{(\partial_{1,\vec{\chi}}G)}(\vec{x}_{obs}) = \int_{\mathcal{L}} \frac{1}{\mathcal{L}_{tot}} d\vec{x}_{l_0} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_0 \int_0^{+\infty} k_s e^{-k_s\xi_0} d\xi_0 \Big\{ \mathcal{H}(\xi_0 - d_0)\hat{\mathrm{S}}^{scat,0}_{b,\vec{\chi}} 4\pi\mathcal{L}_{tot} \\ + \mathcal{H}(d_0 - \xi_0) \int_{\mathcal{L}} \frac{1}{\mathcal{L}_{tot}} d\vec{x}_{l_1} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^{+\infty} k_s e^{-k_s\xi_1} d\xi_1 \{ \mathcal{H}(\xi_1 - d_1)(\hat{\mathrm{S}}^{scat,0}_{b,\vec{\chi}} + \hat{\mathrm{S}}^{scat,1}_{b,\vec{\chi}}) 4\pi\mathcal{L}_{tot} \\ + \mathcal{H}(d_1 - \xi_1) \int_{\mathcal{L}} \frac{1}{\mathcal{L}_{tot}} d\vec{x}_{l_2} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \int_2^{+\infty} k_s e^{-k_s\xi_2} d\xi_2 \dots \} \Big\}\end{aligned} \tag{6.60}$$

where $\mathcal{L}_{tot}$ is the total length of all the edges $\mathcal{L}$; The notations in this equation are the same as those in Eq.6.51. A "random walks" algorithm and a physical image are also attached on Eq.6.60. The photon is scattered in the medium until it hit the boundaries. But in this algorithm, the contribution of line sources is separated. At each scattering event, the corresponding contribution is accumulated. This technique is used also in [52, 57], shown in Fig.6.14.

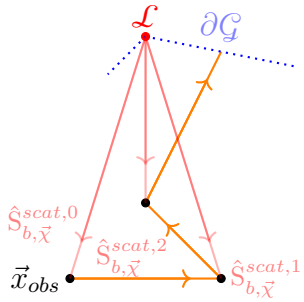


Figure 6.14 – In order to accumulate the line sources in a scattering medium, its contribution is accumulated at each position of the scattering events.

Finally, we write Eq.6.51, Eq.6.60 and Eq.6.53 into a vectorized form, reminding that $\partial_{1,\vec{\chi}}G(\vec{x}_{obs}) = \widehat{(\partial_{1,\vec{\chi}}G)}(\vec{x}_{obs}) + (\partial_{1,\vec{\chi}}\dot{G})(\vec{x}_{obs})$:

$$\begin{aligned}\begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}} G(\vec{x}_{obs}) \end{bmatrix} = \int_{\mathcal{L}} \frac{1}{\mathcal{L}_{tot}} d\vec{x}_{l_0} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_0 \int_0^{+\infty} k_s e^{-k_s \xi_0} d\xi_0 \\ \Big\{ \mathcal{H}(\xi_0 - d_0) \begin{bmatrix} \mathring{I} \\ \hat{\mathrm{S}}^{scat,0}_{b,\vec{\chi}} \mathcal{L}_{tot} + \dot{\mathrm{S}}^{scat}_{b,\vec{\chi}} \end{bmatrix} 4\pi \\ + \mathcal{H}(d_0 - \xi_0) \int_{\mathcal{L}} \frac{1}{\mathcal{L}_{tot}} d\vec{x}_{l_1} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^{+\infty} k_s e^{-k_s \xi_1} d\xi_1 \\ \Big\{ \mathcal{H}(\xi_1 - d_1) \begin{bmatrix} \mathring{I} \\ (\hat{\mathrm{S}}^{scat,0}_{b,\vec{\chi}} + \hat{\mathrm{S}}^{scat,1}_{b,\vec{\chi}}) \mathcal{L}_{tot} + \dot{\mathrm{S}}^{scat}_{b,\vec{\chi}} \end{bmatrix} 4\pi \\ + \mathcal{H}(d_1 - \xi_1) \int_{\mathcal{L}} \frac{1}{\mathcal{L}_{tot}} d\vec{x}_{l_2} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \int_2^{+\infty} k_s e^{-k_s \xi_2} d\xi_2 \dots \Big\} \Big\} \end{aligned} \tag{6.61}$$

At this stage, a Monte-Carlo algorithm dedicated to accumulating all surface sources and line sources is developed (Eq.6.61). The same "random walk" algorithm, which is discussed earlier in this section, is used to accumulate the surface sources. Also, the surface sources of intensity and those of spatial derivative are accumulated in a vectorized form.

Moreover, compared to Eq.6.53 and Eq.6.51, in Eq.6.61, an extra line sampling is needed at each scattering events, so that the contribution of line sources can be accumulated. The corresponding Algorithm is Algo.12.

Algorithm 12 Evaluation of G and $\partial_{1,\vec{\chi}} G$ with line sources in scattering medium

Input: $\mathscr{G}$, n_{MC}, $\vec{x}_{obs}$, k_s

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{x}_{obs}$: the observation point.
▷ $\vec{k}_s$: coefficient of scattering.

$W_1 \leftarrow 0$, $W_1 \leftarrow 0$ and $V_2 \leftarrow 0$, $V_2 \leftarrow 0$ ▷ Initialize the sum and the sum of square
for $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ First Monte-Carlo for surface sources.
  Initialize the weight of Monte-Carlo $w_1 \leftarrow 0$ and $w_2 \leftarrow 0$
  Initialize the keeprunning flag $keeprunning \leftarrow 1$
  Sample uniformly a position on $\mathcal{L}$: $\vec{x}_l$
  $\vec{x}_{l_j} \leftarrow \vec{x}_l$
  $\vec{x}_j \leftarrow \vec{x}_{obs}$
  Calculate $\hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j})$ (Eq.6.57).
  Accumulate the weight $w_2 \leftarrow w_2 + \hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}} 4\pi \mathcal{L}_{tot}$
  Sample uniformly a direction $\vec{\omega}$
  Initialize the position for ray tracing $\vec{x} \leftarrow \vec{x}_{obs}$
  while *keeprunning* **do**
    Get the intersection point of the ray $(\vec{x}, \vec{\omega})$ on $\partial\mathcal{G}$: $\vec{x}_{hit}$
    Get the distance between $\vec{x}_{hit}$ and $\vec{x}$: $d \leftarrow ||\vec{x}_{hit} - \vec{x}||$
    Sample a length based on $\mathcal{P}_\Sigma(\sigma) = k_s e^{-k_s \sigma}$
    if $\sigma < d$ **then**
      Update the position $\vec{x} \leftarrow \vec{x} + \vec{\omega}\sigma$
      $\vec{x}_j \leftarrow \vec{x}$
      Sample uniformly a position on $\mathcal{L}$: $\vec{x}_l$
      $\vec{x}_{l_j} \leftarrow \vec{x}_l$

Calculate $\hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j})$ (Eq.6.57).
Accumulate the weight $w_2 \leftarrow w_2 + \hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}} 4\pi \mathcal{L}_{tot}$
Sample uniformly a new direction $\vec{\omega}$
else
Get the normal on the hitting point $\vec{x}_{hit}$: $\vec{n}$
Get $\mathring{I}(\vec{x}_{hit}, \vec{\omega})$
Calculate the weight $w_1 \leftarrow \mathring{I}(\vec{x}_{hit}, \vec{\omega}) 4\pi$
$keeprunning \leftarrow 0$
Sample uniformly a new direction $\vec{\omega}'$
if $\vec{\omega}' \cdot \vec{n} > 0$ **then**
Compute α (Eq.6.16)
Get $I(\vec{x}, \vec{\omega}')$ by Algo.13
Compute the weight
$w_2 \leftarrow w_2 - \alpha k_s \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \alpha k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}') + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p)$
else
Compute α (Eq.6.16)
Compute the weight
$w_2 \leftarrow w_2 - \alpha k_s \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_p, \vec{\omega}_p)$
end if
end if
$W_1 \leftarrow W_1 + w_1$
$V_1 \leftarrow V_1 + w_1^2$
$W_2 \leftarrow W_2 + w_2$
$V_2 \leftarrow V_2 + w_2^2$
$i_{MC} \leftarrow i_{MC} + 1$
end while
end for

Output: $G \leftarrow \frac{W_1}{n_{MC}}$, $\sigma[G] \leftarrow \sqrt{\frac{\frac{V_1}{n_{MC}} - (\frac{W_1}{n_{MC}})^2}{n_{MC}-1}}$, $\partial_{1,\vec{\chi}} G \leftarrow \frac{W_2}{n_{MC}}$, $\sigma[\partial_{1,\vec{\chi}} G] \leftarrow \sqrt{\frac{\frac{V_2}{n_{MC}} - (\frac{W_2}{n_{MC}})^2}{n_{MC}-1}}$

Algorithm 13 Get $I(\vec{x}, \vec{\omega}')$ with the sampled $\vec{x}$ and $\vec{\omega}'$

Input: $\mathscr{G}$, $\vec{x}$, $\vec{\omega}'$, k_s

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{k}_s$: coefficient of scattering.

Initialize the keeprunning flag $keeprunning \leftarrow 1$
Initialize the ray-tracing process: $\vec{\omega} \leftarrow \vec{\omega}'$
while keeprunning **do**
Get the first intersection point of the ray $(\vec{x}, \vec{\omega})$: $\vec{x}_{hit}$
Get the distance between $\vec{x}_{hit}$ and $\vec{x}$: $d \leftarrow ||\vec{x}_{hit} - \vec{x}||$
Sample a length based on $\mathscr{P}_\Sigma(\sigma) = k_s e^{-k_s \sigma}$
if $\sigma < d$ **then**
Sample a direction $\vec{\omega}'$ based on $\mathscr{P}_{\Omega'}(\vec{\omega}') = \frac{1}{4\pi}$
Update the position $\vec{x} \leftarrow \vec{x} + \vec{\omega}\sigma$
Update the direction of propagation $\vec{\omega} \leftarrow \vec{\omega}'$
else
$I \leftarrow \mathring{I}(\vec{x}_{hit}, \vec{\omega})$
$keeprunning \leftarrow 0$
end if
end while

Output: I

6.6.2 Vectorization with surface sources only

In this section, we apply our method to transform the line sources in Eq.6.61 into surface sources. It helps to avoid the line sampling at each scattering event in Algo.13. In order to achieve this, we focus on the analysis of line integral in Eq.6.56:

$$\int_{\mathcal{L}} d\vec{x}_{l_j} \hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j}) = \int_{\mathcal{L}} d\vec{x}_{l_j} \frac{(\vec{\omega}_{l_j} \times \vec{\chi}) \cdot \vec{t}_i}{r^2_{l_j}} (I_i - I_{i'}) e^{-k_s r_{l_j}} \tag{6.62}$$

As it is shown in Section.6.3, the integral over $\mathcal{L}$ can be transformed into the sum of the integral over the edges of each plane surface:

$$\int_{\mathcal{L}} d\vec{x}_{l_j} \frac{(\vec{\omega}_{l_j} \times \vec{\chi}) \cdot \vec{t}_i}{r^2_{l_j}} (I_i - I_{i'}) e^{-k_s r_{l_j}} = \sum_{i=1}^{n} \oint_{\mathcal{L}_i} d\vec{x}_{l_j} \frac{(\vec{\omega}_{l_j} \times \vec{\chi}) \cdot \vec{t}_i}{r^2_{l_j}} I_i(\vec{x}_{l_j}, \vec{\omega}_{l_j}) e^{-k_s r_{l_j}} \tag{6.63}$$

The left side of Eq.6.63 accumulates the line sources by a line integral, while the right side accumulates the line sources surface by surface. Both sides of Eq.6.63 represent the sum of all line sources. The value of I_i and of $I_{i'}$ can be referred to Eq.6.19 and Eq.6.20; $\mathcal{L}_i$ is the edges of the i^{th} plane surface ($\partial\mathcal{G}_i$).

By applying the Stokes' Theorem to Eq.6.63 (following the same development of Eq.6.28 in Section.6.3), the line integral is transformed to a surface integral:

$$\begin{aligned}
&\sum_{i=1}^{n} \oint_{\mathcal{L}_i} d\vec{x}_{l_j} \frac{(\vec{\omega}_{l_j} \times \vec{\chi}) \cdot \vec{t}_i}{r^2_{l_j}} I_i(\vec{x}_{l_j}, \vec{\omega}_{l_j}) e^{-k_s r_{l_j}} \\
&\qquad = \sum_{i=1}^{n} \int_{\partial\mathcal{G}_i} d\vec{x}_{p_j} \vec{\nabla} \times \left(\frac{\vec{\omega}_{p_j} \times \vec{\chi}}{r^2_{l_j}} e^{-k_s r_{p_j}}\right) \cdot \vec{n}_i \mathring{I}(\vec{x}_{p_j}, \vec{\omega}_{p_j}) \\
&\qquad\qquad - \sum_{i=1}^{n} \int_{\partial\mathcal{G}_i} \frac{(\vec{\omega}_{p_j} \cdot \vec{n}_i)}{r^2_{p_j}} \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_{p_j}, \vec{\omega}_{p_j}) d\vec{x}_j
\end{aligned} \tag{6.64}$$

where

$$r_{p_j} = ||\vec{x}_j - \vec{x}_{p_j}||; \vec{\omega}_{p_j} = \frac{\vec{x}_j - \vec{x}_{p_j}}{r_{p_j}} \tag{6.65}$$

Therefore, Eq.6.62 can be reformulated in a surface integral:

$$\begin{aligned}
&\int_{\mathcal{L}} d\vec{x}_{l_j} \frac{(\vec{\omega}_{l_j} \times \vec{\chi}) \cdot \vec{t}_i}{r^2_j} (I_i - I_{i'}) e^{-k_s r_{l_j}} \\
&\qquad = \int_{\partial\mathcal{G}} d\vec{x}_{p_j} \vec{\nabla} \times \left(\frac{\vec{\omega}_{p_j} \times \vec{\chi}}{r^2_{p_j}} e^{-k_s r_{p_j}}\right) \cdot \vec{n} \mathring{I}(\vec{x}_{p_j}, \vec{\omega}_{p_j}) \\
&\qquad\qquad - \int_{\partial\mathcal{G}} d\vec{x}_{p_j} \frac{(\vec{\omega}_{p_j} \cdot \vec{n}_i)}{r^2_{p_j}} \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_{p_j}, \vec{\omega}_{p_j})
\end{aligned} \tag{6.66}$$

We then substitute Eq.6.66 into Eq.6.56 and Eq.6.55. The line contribution before applying the Stokes' Theorem is noted $(\widehat{\partial_{1,\vec{\chi}}G})$. After applying the Stokes' Theorem, the line integral no longer exists and the same contribution is noted $(\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G})$.

Therefore, referring to Eq.6.52, we have:

$$\partial_{1,\vec{\chi}}G = (\overset{\bullet}{\partial}_{1,\vec{\chi}}G) + (\widehat{\partial_{1,\vec{\chi}}G}) = (\overset{\bullet}{\partial}_{1,\vec{\chi}}G) + (\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G}). \tag{6.67}$$

Herein, we separate the source $(\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G})$ as we have done to separate the source $(\widehat{\partial_{1,\vec{\chi}}G})$ in Eq.6.55. We classify it following the number of scattering events before arriving at $\vec{x}_{obs}$, $\forall j \in \mathbb{N}$:

$$(\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G}) = \sum_{j=0}^{+\infty} (\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G})_j \tag{6.68}$$

where j is the index of the number of scattering. $(\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G})_j$ is then the sources of spatial derivative on the boundary $\partial\mathcal{G}$ arriving to the point $\vec{x}_{obs}$, after j scattering events:

$$\begin{aligned}(\widehat{\overset{\bullet}{\partial}_{1,\vec{\chi}}G})_j = \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_0 \int_0^{d_0} k_s e^{-k_s\sigma_0} d\sigma_0 \dots \\ \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_{j-1} \int_0^{d_{j-1}} k_s e^{-k_s\sigma_{j-1}} d\sigma_{j-1} \\ \int_{\partial\mathcal{G}} d\vec{x}_{p_j} \dot{\hat{\mathrm{S}}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j})\end{aligned} \tag{6.69}$$

where

$$\begin{aligned}\dot{\hat{\mathrm{S}}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j}) = \vec{\nabla} \times \left(\frac{\vec{\omega}_{p_j} \times \vec{\chi}}{r_{p_j}^2} e^{-k_s r_{p_j}}\right) \cdot \vec{n} \mathring{I}(\vec{x}_{p_j}, \vec{\omega}_{p_j}) \frac{1}{4\pi} \\ - \frac{(\vec{\omega}_{p_j} \cdot \vec{n})}{r_{p_j}^2} \beta \partial_{1,\vec{u}} \mathring{I}(\vec{x}_{p_j}, \vec{\omega}_{p_j}) \frac{1}{4\pi}\end{aligned} \tag{6.70}$$

In Eq.6.57, $\vec{x}_j$ is referred to Eq.6.59; $\vec{\omega}_{p_j}$ and r_{p_j} are referred to Eq.6.65. Finally, reminding $\partial_{1,\vec{\chi}}G = (\overset{\bullet}{\partial}_{1,\vec{\chi}}G) + (\widehat{\partial_{1,\vec{\chi}}G})$. we formulate the vectorized form to estimate $\begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}}G(\vec{x}_{obs}) \end{bmatrix}$:

$$\begin{aligned}
\begin{bmatrix} G(\vec{x}_{obs}) \\ \partial_{1,\vec{\chi}} G(\vec{x}_{obs}) \end{bmatrix} = & \int_{\partial\mathcal{G}} \frac{1}{\mathcal{S}_{tot}} d\vec{x}_{l_0} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_0 \int_0^{+\infty} k_s e^{-k_s \sigma_0} d\sigma_0 \\
& \Big\{ \mathcal{H}(\sigma_0 - d_0) \begin{bmatrix} \mathring{I} \\ \dot{\hat{\mathrm{S}}}^{scat,0}_{b,\vec{\chi}} \mathcal{S}_{tot} + \dot{\mathrm{S}}^{scat}_{b,\vec{\chi}} \end{bmatrix} 4\pi \\
& + \mathcal{H}(d_0 - \sigma_0) \int_{\partial\mathcal{G}} \frac{1}{\mathcal{S}_{tot}} d\vec{x}_{l_1} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_1 \int_0^{+\infty} k_s e^{-k_s \sigma_1} d\sigma_1 \\
& \Big\{ \mathcal{H}(\sigma_1 - d_1) \begin{bmatrix} \mathring{I} \\ (\dot{\hat{\mathrm{S}}}^{scat,0}_{b,\vec{\chi}} + \dot{\hat{\mathrm{S}}}^{scat,1}_{b,\vec{\chi}}) \mathcal{S}_{tot} + \dot{\mathrm{S}}^{scat}_{b,\vec{\chi}} \end{bmatrix} 4\pi \\
& + \mathcal{H}(d_1 - \sigma_1) \int_{\partial\mathcal{G}} \frac{1}{\mathcal{S}_{tot}} d\vec{x}_{l_2} \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}_2 \int_2^{+\infty} k_s e^{-k_s \sigma_2} d\sigma_2 \dots \Big\} \Big\}
\end{aligned} \tag{6.71}$$

Form Eq.6.71, a "random walks" algorithm and a physical image are proposed. The photon is scattered in the medium until it hits the boundaries. But in this algorithm, the contribution of surface sources on $\partial\mathcal{G}$ is separated. At each scattering event, the corresponding contribution is accumulated, seeing the schema in Fig.6.15. The corresponding Algorithm is Algo.14.

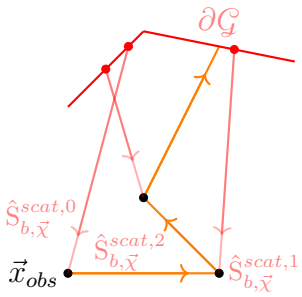


Figure 6.15 – In order to accumulate the surface sources in a scattering medium, its contribution is accumulated at each position of the scattering events.

To sum up, Eq.6.61 and Eq.6.71 are both dedicated to estimate the observable G and its spatial derivative $\partial_{1,\vec{\chi}} G$. In Eq.6.61, the line sources $(\widehat{\partial_{1,\vec{\chi}} G})$ are accumulated at each scattering event. In Eq.6.53, the line sources are transformed to surface sources $(\overset{\bullet}{\widehat{\partial_{1,\vec{\chi}} G}})$. However, the surface sources are accumulated at each scattering event. Therefore, if we compare Algo.12 and Algo.14, we avoid sampling on the lines by transforming the line sources into surface sources. At each scattering event, surface sampling is still required to accumulate the surface sources. We remind that the Fig.6.14 corresponds to Eq.6.61 and Algo.12, without applying the Stokes' theorem. Fig.6.15 corresponds to Eq.6.71 and Algo.14, after applying the Stokes' theorem.

Algorithm 14 Evaluation of G and $\partial_{1,\vec{\chi}} G$ with surface sources in scattering medium

Input: $\mathscr{G}$, n_{MC}, $\vec{x}_{obs}$, k_s

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{G}$: the convex geometry bounded by plane surfaces.
▷ $\vec{x}_{obs}$: the observation point.
▷ $\vec{k}_s$: coefficient of scattering.

$W_1 \leftarrow 0$, $W_1 \leftarrow 0$ and $V_2 \leftarrow 0$, $V_2 \leftarrow 0$ ▷ Initialize the sum and the sum of square
for $i_{MC} = 0 \rightarrow n_{MC}$ **do** ▷ First Monte-Carlo for surface sources.
  Initialize the weight of Monte-Carlo $w_1 \leftarrow 0$ and $w_2 \leftarrow 0$
  Initialize the keeprunning flag $keeprunning \leftarrow 1$
  Sample uniformly a position on $\partial\mathcal{G}$: $\vec{x}_p$
  $\vec{x}_{p_j} \leftarrow \vec{x}_p$
  $\vec{x}_j \leftarrow \vec{x}_{obs}$
  Calculate $\hat{\dot{\mathrm{S}}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j})$ (Eq.6.70).
  Accumulate the weight $w_2 \leftarrow w_2 + \hat{\dot{\mathrm{S}}}^{scat,j}_{b,\vec{\chi}} 4\pi\mathcal{S}_{tot}$
  Sample uniformly a direction $\vec{\omega}$
  Initialize the position for ray tracing $\vec{x} \leftarrow \vec{x}_{obs}$
  while *keeprunning* **do**
    Get the intersection point of the ray $(\vec{x}, \vec{\omega})$ on $\partial\mathcal{G}$: $\vec{x}_{hit}$
    Get the distance between $\vec{x}_{hit}$ and $\vec{x}$: $d \leftarrow ||\vec{x}_{hit} - \vec{x}||$
    Sample a length based on $\mathcal{P}_\Sigma(\sigma) = k_s e^{-k_s\sigma}$
    if $\sigma < d$ **then**
      Update the position $\vec{x} \leftarrow \vec{x} + \vec{\omega}\sigma$
      $\vec{x}_j \leftarrow \vec{x}$
      Sample uniformly a position on $\partial\mathcal{G}$: $\vec{x}_p$
      $\vec{x}_{p_j} \leftarrow \vec{x}_l$
      Calculate $\hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}}(\vec{x}_j, \vec{x}_{l_j})$ (Eq.6.70).
      Accumulate the weight $w_2 \leftarrow w_2 + \hat{\mathrm{S}}^{scat,j}_{b,\vec{\chi}} 4\pi\mathcal{S}_{tot}$
      Sample uniformly a new direction $\vec{\omega}$
    else
      Get the normal on the hitting point $\vec{x}_{hit}$: $\vec{n}$
      Get $\mathring{I}(\vec{x}_{hit}, \vec{\omega})$
      Calculate the weight $w_1 \leftarrow \mathring{I}(\vec{x}_{hit}, \vec{\omega})4\pi$
      $keeprunning \leftarrow 0$
      Sample uniformly a new direction $\vec{\omega}'$
      if $\vec{\omega}' \cdot \vec{n} > 0$ **then**
        Compute α (Eq.6.16)
        Get $I(\vec{x}, \vec{\omega}')$ by Algo.13
        Compute the weight
          $w_2 \leftarrow w_2 - \alpha k_s \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \alpha k_s \int_{4\pi} \frac{1}{4\pi} d\vec{\omega}' I(\vec{x}, \vec{\omega}') + \beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p, \vec{\omega}_p)$
      else
        Compute α (Eq.6.16)
        Compute the weight
          $w_2 \leftarrow w_2 - \alpha k_s \mathring{I}(\vec{x}_p, \vec{\omega}_p) + \beta\partial_{1,\vec{u}}\mathring{I}(\vec{x}_p, \vec{\omega}_p)$
      end if
    end if
    $W_1 \leftarrow W_1 + w_1$
    $V_1 \leftarrow V_1 + w_1^2$
    $W_2 \leftarrow W_2 + w_2$
    $V_2 \leftarrow V_2 + w_2^2$
    $i_{MC} \leftarrow i_{MC} + 1$
  end while
end for

Output: $G \leftarrow \frac{W_1}{n_{MC}}$, $\sigma[G] \leftarrow \sqrt{\frac{\frac{V_1}{n_{MC}} - (\frac{W_1}{n_{MC}})^2}{n_{MC}-1}}$, $\partial_{1,\vec{\chi}}G \leftarrow \frac{W_2}{n_{MC}}$, $\sigma[\partial_{1,\vec{\chi}}G] \leftarrow \sqrt{\frac{\frac{V_2}{n_{MC}} - (\frac{W_2}{n_{MC}})^2}{n_{MC}-1}}$

6.7 Validations, results and discussion

In this section, the developed algorithms are tested. We use a cube as our convex geometry (Fig.6.16). Also, all assumptions and definitions made in Section.6.2 still remain. $\mathscr{E} \subset \partial\mathcal{G}$ is the surface on the top of the cube, which is an emitter. $\mathscr{E}$ is a black body with a homogeneous temperature. It means that the intensity emitted on $\mathscr{E}$ is considered homogeneous and equal to I_0 (the source of intensity $\mathring{I} = I_0$ is independent of the position $\vec{x}_p$). Other surfaces of the cube are the "cold black" surfaces, noted $\mathscr{B} \subset \partial\mathcal{G}$. The observation point $\vec{x}_{obs}$ is initially set at the center of the cube. The differentiation direction in this section is chosen as $\vec{\chi}_0 \equiv \begin{bmatrix}1\\0\\0\end{bmatrix}$. We aim to estimate the angular integral of intensity arriving at $\vec{x}_{obs}$: $G \equiv G(\vec{x}_{obs})$ and its spatial derivative $\partial_{1,\vec{\chi}_0} G \equiv \partial_{1,\vec{\chi}_0} G(\vec{x}_{obs})$, referred to Eq.6.5 and Eq.6.6.

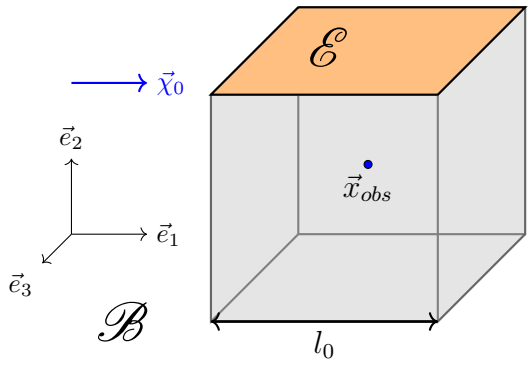


Figure 6.16 – The configuration for validations. Point $\vec{x}_{obs}$ is the observation point; The geometric center of the cube is the origin $O \in [0,0,0]^T$; $\vec{e}_1$, $\vec{e}_2$ and $\vec{e}_3$ compose the global coordinate system. $\vec{\chi}_0$ is the differentiation direction chosen for validations in this section. l_0 is the length of the border of the cube.

As it is introduced in Chapter.1, for validation purposes, the spatial derivative $\partial_{1,\vec{\chi}_0} G(\vec{x}_{obs})$ estimated will also be approximated by Finite Different method. The result of approximation is noted $\partial_{1,\vec{\chi}_0} \tilde{G}(\vec{x}_{obs})$:

When G is derivable around $\vec{x}_{obs}$:

$$\partial_{1,\vec{\chi}} G(\vec{x}_{obs}) = \lim_{\epsilon \to 0} \frac{G(\vec{x}_{obs} + \epsilon\vec{\chi}) - G(\vec{x}_{obs} - \epsilon\vec{\chi})}{2\epsilon} \tag{6.72}$$

The finite difference method approximates $\partial_{1,\vec{\chi}} G$ by giving a non-zero fixed value Δ to ϵ. We note the result of approximation as $\partial_{1,\vec{\chi}} \tilde{G}(\vec{x}_{obs})$:

$$\partial_{1,\vec{\chi}} G(\vec{x}_{obs}) \approx \frac{G(\vec{x}_{obs} + \Delta\vec{\chi}) - G(\vec{x}_{obs} + \Delta\vec{\chi})}{2\Delta} \equiv \partial_{1,\vec{\chi}} \tilde{G}(\vec{x}_{obs}) \tag{6.73}$$

Practically, the approximation of $\partial_{1,\vec{\chi}} \tilde{G}$ requires estimating G twice. Since G is estimated by the Monte-Carlo method, its statistical standard deviation $\sigma(G)$ is also estimated at the same time [14]. When approximating $\tilde{S}_j$, its standard deviation $\partial_{1,\vec{\chi}} \tilde{G}$ can be obtained [21]:

$$\sigma(\partial_{1,\vec{\chi}} \tilde{G}(\vec{x}_{obs})) \approx \frac{\sigma(G(\vec{x}_{obs} + \Delta\vec{\chi})) + \sigma(G(\vec{x}_{obs} - \Delta\vec{\chi}))}{2\Delta} \tag{6.74}$$

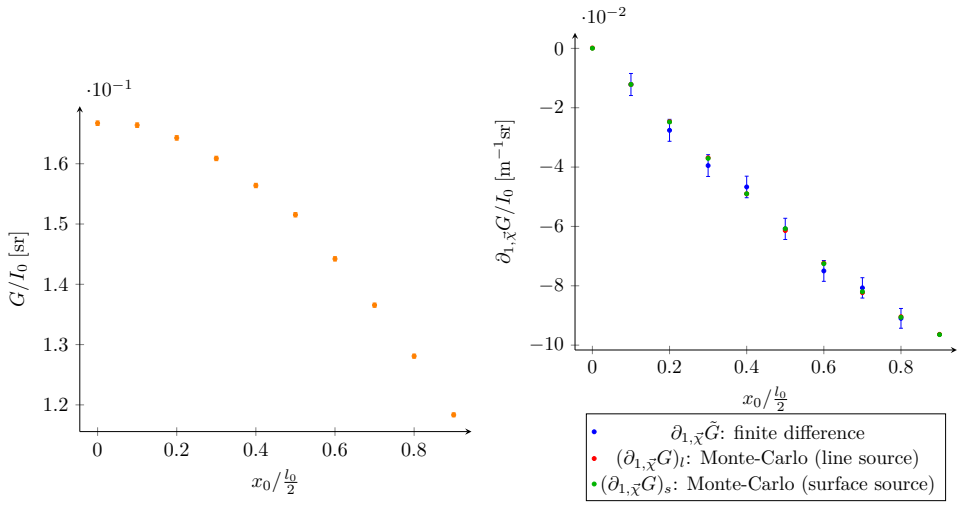
The limitations of this method have already been discussed in the previous chapters. However, we still use the results approximated by Finite Difference method $\partial_{1,\vec{\chi}} \tilde{G}$ to validate our estimations of $\partial_{1,\vec{\chi}} G$.

We remind our proposal for this chapter, which is transforming the line sources into surface sources to avoid the line integrals. Since the algorithms of estimating $\partial_{1,\vec{\chi}}G$ is built with or without the line sources, we note $(\partial_{1,\vec{\chi}}G)_l$ as the results obtained by the algorithm with line sources and $(\partial_{1,\vec{\chi}}G)_s$ as the results obtained by the algorithm with surface sources.

6.7.1 Transparent medium

Firstly, we will validate Algo.8 and Algo.9 for transparent medium. The observation point $\vec{x}_{obs}$ is defined as a function of x_0, which is the first component of $\vec{x}_{obs} \equiv \begin{bmatrix} x_0 \\ 0 \\ 0 \end{bmatrix}$, $x_0 \in [-\frac{l}{2}, \frac{l}{2}]$.

Results for different positions of observations are shown in Table.6.3 and Fig.6.17. In the Table.6.3, G and $(\partial_{1,\vec{\chi}}G)_l$ are estimated by Algro.8 and $(\partial_{1,\vec{\chi}}G)_s$ is estimated by Algo.9. The approximation is calculated following Eq.6.73, based on the results of G, estimated by Algo.8. The x-axis of Fig.6.17 is non-dimensional The y-axis of Fig.6.17a physically represents the solid angle, under which the surface $\mathcal{E}$ is viewed from the point $\vec{x}_{obs}$ [23]. The y-axis of Fig.6.17b represents the gradient of solid angle, under which the surface $\mathcal{E}$ is viewed from the point $\vec{x}_{obs}$.



(a) Results of G.

(b) Results of $\partial_{1,\vec{\chi}}G$.

Figure 6.17 – Results and validations of Algo.8 and of Algo.9 for transparent medium. Number of Monte-Carlo realization $n_{MC} = 100000$.

From Fig.6.17b, the results of Aglo.8, Algo.9 are validated. Our method of transforming the line sources into surface sources works well in the case of an absorbent medium. Following Algo.9, we successfully avoid the line samplings and vectorize completely the estimations of G and $\partial_{1,\vec{\chi}}G$.

In order to study the convergence performance, we realized tests in the condition of a transparent medium, with different positions of $\vec{x}_{obs}$. The interesting results we obtained are shown in Table.6.1, where the observation point $\vec{x}_{obs}$ is set as a function of z_0: $\vec{x}_{obs} = [0.25l_0, 0, z_0]^T$. It is found that with the $\vec{x}_{obs}$ closer to the emitter, the ratio $\sigma((\partial_{1,\vec{\chi}}G))_s/(\partial_{1,\vec{\chi}}G)_s$ increases significantly. This convergence issue can also be found when the medium is absorbent.

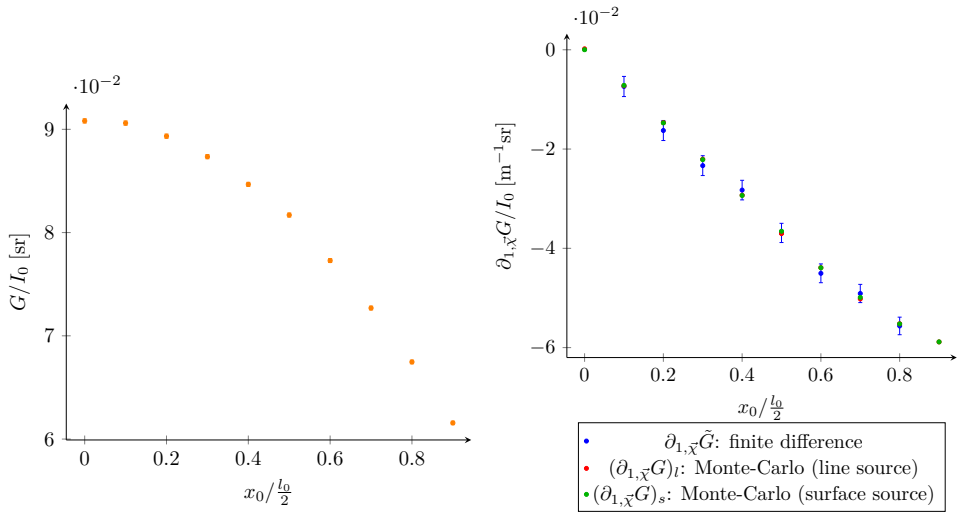
6.7.2 Absorbent medium

Then, we validate Algo.10 and Algo.11 for absorbent medium. Two absorption coefficients are chosen (for $k_a/l_0 = 0.25$ and $k_a/l_0 = 1.5$).

$z_0/\frac{l_0}{2}$	$(\partial_{1,\vec{\chi}}G)_s$	$\sigma((\partial_{1,\vec{\chi}}G))_s$	$\sigma((\partial_{1,\vec{\chi}}G))_s/(\partial_{1,\vec{\chi}}G)_s$
0	-61.12×10^{-3}	37.71×10^{-6}	0.06%
0.5	-112.79×10^{-3}	203.18×10^{-6}	0.18%
0.9	-53.70×10^{-3}	5.46×10^{-3}	10.17%
0.95	-29.18×10^{-3}	21.72×10^{-3}	74.40%
0.99	-265.49×10^{-3}	547.89×10^{-3}	200.36%

Table 6.1 – Following Algo.9, estimations of $(\partial_{1,\vec{\chi}}G)_s$ with the observation point closer or further from the surface where the surface sources locate.

Results for different positions of observations are shown in Table.6.4, Table.6.5, Fig.6.18 and Fig.6.19. In the Table.6.4, G and $(\partial_{1,\vec{\chi}}G)_l$ are estimated by Algo.10 and $(\partial_{1,\vec{\chi}}G)_s$ is estimated by Algo.11. The approximation is calculated following Eq.6.73, based on the results of G, estimated by Algo.10.



(a) Results of G.

(b) Results of $\partial_{1,\vec{\chi}}G$.

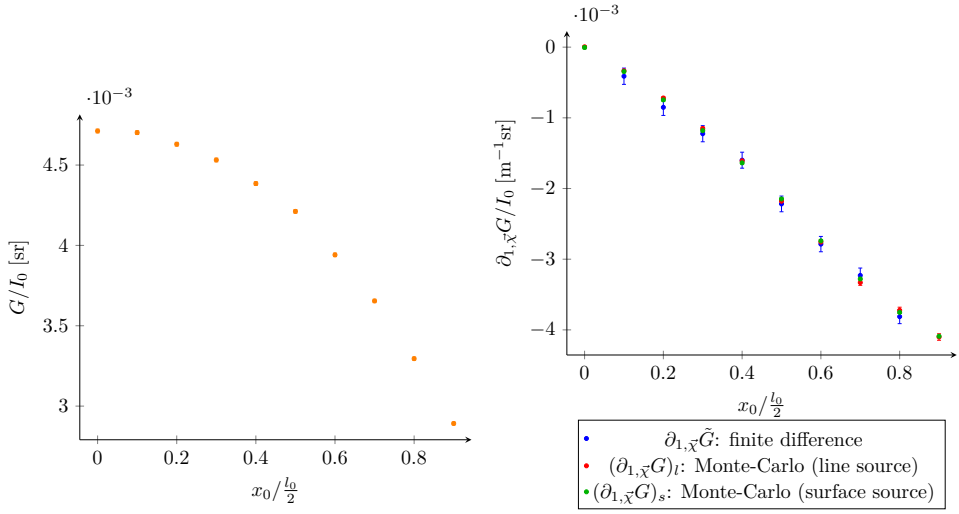
Figure 6.18 – Results and validations of Algo.10 and of Algo.11 for absorbent medium ($k_a/l_0 = 0.25$). Number of Monte-Carlo realization $n_{MC} = 100000$.

From Fig.6.18 and Fig.6.19, the results of Aglo.10, Algo.11 are validated. Our method of transforming the line sources into surface sources works well in the case of an absorbent medium. Following Algo.9, we successfully avoid the line samplings and vectorize completely the estimations of G and $\partial_{1,\vec{\chi}}G$.

6.7.3 Scattering medium

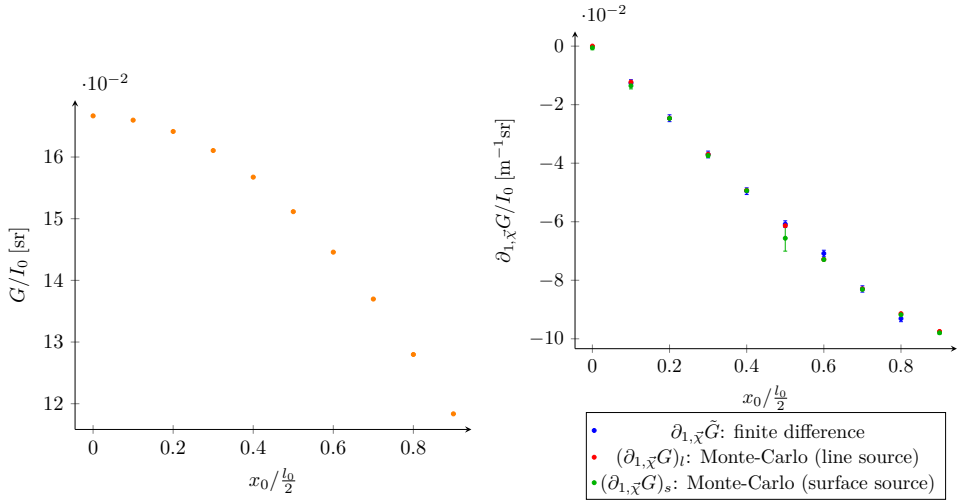
Then, we validate Algo.12 and Algo.14 for scattering medium. Only a small absorption coefficients is chosen (for $k_s = 0.01\text{m}^{-1}$), because of the convergence issues observed. Only a small absorption coefficients is chosen (for $k_s/l_0 = 0.005$), because of the convergence issues observed. This issue will be discussed later in this section.

Results for different positions of observations are shown in Table.6.6 and Fig.6.20. In the Table.6.6, G and $(\partial_{1,\vec{\chi}}G)_l$ are estimated by Algo.12 and $(\partial_{1,\vec{\chi}}G)_s$ is estimated by Algo.14. The approximation is calculated following Eq.6.73, based on the results of G, estimated by Algo.12.



(a) Results of G. **(b)** Results of $\partial_{1,\vec{\chi}}G$.

Figure 6.19 – Results and validations of Algo.10 and of Algo.11 for absorbent medium ($k_a/l_0 = 1.5$). Number of Monte-Carlo realization $n_{MC} = 100000$.



(a) Results of G. **(b)** Results of $\partial_{1,\vec{\chi}}G$.

Figure 6.20 – Results and validations of Algo.12 and of Algo.14 for scattering medium. Number of Monte-Carlo realization $n_{MC} = 1000000$.

It is found in Fig.6.20, the result of the point where $x_0/\frac{l_0}{2} = 0.5$ is not converged well for $(\partial_{1,\vec{\chi}}G)_s$. However, the result of $(\partial_{1,\vec{\chi}}G)_l$ does not have this problem of convergence. This convergence issue is discussed in the following section.

An issue of convergence is observed when estimating $(\partial_{1,\vec{\chi}}G)_s$ by Algo.14. As it is found in Fig.6.20 and Table.6.6, when the observation point is set at $[0.25l_0, 0, 0]^T$, the standard deviation $\sigma((\partial_{1,\vec{\chi}}G)_s)$ is very large. In order to better observe the convergence issue, a larger k_s is set ($k_s = 1\text{m}^{-1}$), remaining the position of the observation point $\vec{x}_{obs}$. Algo.14 is then tested for several number of Monte-Carlo realization (n_{MC}) and results are in Table.6.2.

In Table.6.2, results obtained through Algo.12 (the $(\partial_{1,\vec{\chi}}G)_l$) don't have specific convergence issue. In Algo.12, one part of the contribution from the line sources is accumulated at each

n_{MC}	$(\partial_{1,\vec{\chi}}G)_s$	$\sigma((\partial_{1,\vec{\chi}}G)_s)$	$(\partial_{1,\vec{\chi}}G)_l$	$\sigma((\partial_{1,\vec{\chi}}G)_l)$
100000	-437.42×10^{-3}	262.16×10^{-3}	-87.07×10^{-3}	1.47×10^{-3}
1000000	116.29×10^{-3}	166.38×10^{-3}	-85.38×10^{-3}	642.91×10^{-6}
10000000	-124.85×10^{-3}	41.66×10^{-3}	-86.07×10^{-3}	224.32×10^{-6}
100000000	-58.23×10^{-3}	36.75×10^{-3}	-85.38×10^{-3}	642.91×10^{-6}

Table 6.2 – Following Algo.14, results of $(\partial_{1,\vec{\chi}}G)_s$ are not reliable when $k_s[\mathrm{m}^{-1}]$ is larger because of the convergence issue.

scattering, even in the medium (Fig.6.14). As for Algo.14, the line sources are extended to surface sources. At each scattering event, one part of the contribution from the surface sources is accumulated (Fig.6.15). Therefore, it seems that the convergence issue is principally due to the process of transformation from line sources into surface sources.

Finally, we understand the convergence issue of Algo.14 in a scattering medium. In Algo.14, the contribution of surface sources is accumulated at each scattering event in the medium. When the scattering coefficient k_s increases, more scattering events happen in the medium. Therefore, some scattering events are very close to the emitter. The contributions accumulated for these scattering events will be divergent, as it is observed in Table.6.1. We temporarily do not have solutions for this convergence problem. The strategy to tackle this convergence issue could be using the variance control techniques of the Monte-Carlo method [14].

6.8 Conclusion of the chapter

The method of sensitivity model introduces both line and surface sources. Inconveniences are due to the line sources when applying the Monte Carlo method to solve this kind of problem. In this chapter, we investigate the transformation of line sources into surface sources with Stokes' theorem under some assumptions (such as convex geometry, cold medium, homogeneous properties, etc.). No more line sources exist in the system, and line sampling is avoided. Also, all surface sources share the same path space. The observable and its spatial derivative are then estimated in a vectorized form, and only one Monte Carlo estimation is needed. This is a significant advantage for the method of observable derivation, introduced in Chapter.2 and Chapter.3. Herein, following the method of sensitivity model, we regain this advantage completely in this Chapter.

However, a convergence issue is found when applying this technique to a scattering medium. We do not have solutions for it for now, but the potential solution could be applying the variance control techniques of the Monte-Carlo method [14].

6.9 Results of simulations

$x_0/\frac{l_0}{2}$	G $\sigma(G)$	$\partial_{1,\vec{\chi}}\tilde{G}$ $\sigma(\partial_{1,\vec{\chi}}\tilde{G})$	$(\partial_{1,\vec{\chi}}G)_l$ $\sigma((\partial_{1,\vec{\chi}}G)_l)$	$(\partial_{1,\vec{\chi}}G)_s$ $\sigma((\partial_{1,\vec{\chi}}G)_s)$
0.0	166.47×10^{-3} 118×10^{-6}		-10×10^{-6} 42×10^{-6}	3×10^{-6} 117×10^{-6}
0.1	166.09×10^{-3} 118×10^{-6}	-10.84×10^{-3} 1.18×10^{-3}	-12.18×10^{-3} 42×10^{-6}	-12.38×10^{-3} 116×10^{-6}
0.2	164.30×10^{-3} 117×10^{-6}	-24.28×10^{-3} 1.17×10^{-3}	-24.46×10^{-3} 44×10^{-6}	-24.69×10^{-3} 115×10^{-6}
0.3	161.23×10^{-3} 116×10^{-6}	-37.32×10^{-3} 1.16×10^{-3}	-36.86×10^{-3} 47×10^{-6}	-37.00×10^{-3} 113×10^{-6}
0.4	156.84×10^{-3} 115×10^{-6}	-49.13×10^{-3} 1.15×10^{-3}	-49.11×10^{-3} 50×10^{-6}	-49.27×10^{-3} 110×10^{-6}
0.5	151.41×10^{-3} 113×10^{-6}	-61.54×10^{-3} 1.13×10^{-3}	-61.08×10^{-3} 54×10^{-6}	-61.11×10^{-3} 107×10^{-6}
0.6	144.53×10^{-3} 111×10^{-6}	-72.27×10^{-3} 1.11×10^{-3}	-72.41×10^{-3} 58×10^{-6}	-72.59×10^{-3} 104×10^{-6}
0.7	136.95×10^{-3} 109×10^{-6}	-81.17×10^{-3} 1.09×10^{-3}	-82.45×10^{-3} 62×10^{-6}	-82.60×10^{-3} 102×10^{-6}
0.8	128.30×10^{-3} 106×10^{-6}	-90.62×10^{-3} 1.06×10^{-3}	-90.63×10^{-3} 65×10^{-6}	-90.87×10^{-3} 100×10^{-6}
0.9	118.83×10^{-3} 102×10^{-6}		-96.67×10^{-3} 67×10^{-6}	-96.67×10^{-3} 100×10^{-6}

Table 6.3 – Validations for different position of observation (from $(0, 0, 0)$ to $(0.45l_0, 0, 0)$), for a number of Monte-Carlo realization $n_{MC} = 10000000$.

$k_a = 0.5\text{m}^{-1}$				
$x_0/\frac{l_0}{2}$	G $\sigma(G)$	$\partial_{1,\vec{\chi}}\tilde{G}$ $\sigma(\partial_{1,\vec{\chi}}\tilde{G})$	$(\partial_{1,\vec{\chi}}G)_l$ $\sigma((\partial_{1,\vec{\chi}}G)_l)$	$(\partial_{1,\vec{\chi}}G)_s$ $\sigma((\partial_{1,\vec{\chi}}G)_s)$
0.0	90.89×10^{-3} 64.94×10^{-6}		-11×10^{-6} 29×10^{-6}	-76×10^{-6} 75×10^{-6}
0.1	90.30×10^{-3} 64.94×10^{-6}	-7.82×10^{-3} 642×10^{-6}	-7.26×10^{-3} 29×10^{-6}	-7.25×10^{-3} 75×10^{-6}
0.2	89.33×10^{-3} 64.94×10^{-6}	-14.13×10^{-3} 638×10^{-6}	-14.59×10^{-3} 31×10^{-6}	-14.58×10^{-3} 74×10^{-6}
0.3	87.48×10^{-3} 63.94×10^{-6}	-21.87×10^{-3} 633×10^{-6}	-21.95×10^{-3} 33×10^{-6}	-22.05×10^{-3} 72×10^{-6}
0.4	84.96×10^{-3} 63.94×10^{-6}	-29.72×10^{-3} 624×10^{-6}	-29.44×10^{-3} 37×10^{-6}	-29.46×10^{-3} 70×10^{-6}
0.5	81.53×10^{-3} 61.94×10^{-6}	-36.55×10^{-3} 613×10^{-6}	-36.82×10^{-3} 41×10^{-6}	-36.85×10^{-3} 67×10^{-6}
0.6	77.65×10^{-3} 60.94×10^{-6}	-43.11×10^{-3} 599×10^{-6}	-43.90×10^{-3} 44×10^{-6}	-43.96×10^{-3} 65×10^{-6}
0.7	72.91×10^{-3} 58.94×10^{-6}	-50.61×10^{-3} 581×10^{-6}	-50.19×10^{-3} 48×10^{-6}	-50.24×10^{-3} 63×10^{-6}
0.8	67.53×10^{-3} 56.94×10^{-6}	-55.04×10^{-3} 560×10^{-6}	-55.36×10^{-3} 51×10^{-6}	-55.34×10^{-3} 61×10^{-6}
0.9	61.90×10^{-3} 54.94×10^{-6}		-58.98×10^{-3} 53×10^{-6}	-59.04×10^{-3} 61×10^{-6}

Table 6.4 – Validations for different position of observation (from $(0, 0, 0)$ to $(0.45l_0, 0, 0)$), for a number of Monte-Carlo realization $n_{MC} = 10000000$.

$k_a = 3\mathrm{m}^{-1}$				
$x_0/\frac{l_0}{2}$	G $\sigma(G)$	$\partial_{1,\vec{\chi}}\tilde{G}$ $\sigma(\partial_{1,\vec{\chi}}\tilde{G})$	$(\partial_{1,\vec{\chi}}G)_l$ $\sigma((\partial_{1,\vec{\chi}}G)_l)$	$(\partial_{1,\vec{\chi}}G)_s$ $\sigma((\partial_{1,\vec{\chi}}G)_s)$
0.0	4.71×10^{-3} 4.09×10^{-6}		2×10^{-6} 5×10^{-6}	-3×10^{-6} 7×10^{-6}
0.1	4.68×10^{-3} 4.09×10^{-6}	-387×10^{-6} 36×10^{-6}	-358×10^{-6} 5×10^{-6}	-354×10^{-6} 7×10^{-6}
0.2	4.63×10^{-3} 4.09×10^{-6}	-717×10^{-6} 36×10^{-6}	-739×10^{-6} 5×10^{-6}	-735×10^{-6} 6×10^{-6}
0.3	4.54×10^{-3} 4.09×10^{-6}	-1.16×10^{-3} 36×10^{-6}	-1.16×10^{-3} -1.64×10^{-3}	-1.16×10^{-3} 6×10^{-6}
0.4	4.40×10^{-3} 4.09×10^{-6}	-1.66×10^{-3} 36×10^{-6}	-1.64×10^{-3} 8×10^{-6}	-1.64×10^{-3} 6×10^{-6}
0.5	4.20×10^{-3} 4.09×10^{-6}	-2.18×10^{-3} 35×10^{-6}	-2.17×10^{-3} 9×10^{-6}	-2.18×10^{-3} 6×10^{-6}
0.6	3.96×10^{-3} 3.09×10^{-6}	-2.71×10^{-3} 34×10^{-6}	-2.74×10^{-3} 11×10^{-6}	-2.75×10^{-3} 5×10^{-6}
0.7	3.66×10^{-3} 3.09×10^{-6}	-3.30×10^{-3} 33×10^{-6}	-3.31×10^{-3} 12×10^{-6}	-3.30×10^{-3} 5×10^{-6}
0.8	3.30×10^{-3} 3.09×10^{-6}	-3.74×10^{-3} 31×10^{-6}	-3.79×10^{-3} 14×10^{-6}	-3.77×10^{-3} 5×10^{-6}
0.9	2.91×10^{-3} 3.09×10^{-6}		-4.10×10^{-3} 15×10^{-6}	-4.10×10^{-3} 5×10^{-6}

Table 6.5 – Validations for different position of observation (from $(0, 0, 0)$ to $(0.45l_0, 0, 0)$), for a number of Monte-Carlo realization $n_{MC} = 10000000$.

$k_s = 0.01\mathrm{m}^{-1}$				
$x_0/\frac{l_0}{2}$	G $\sigma(G)$	$\partial_{1,\vec{\chi}}\tilde{G}$ $\sigma(\partial_{1,\vec{\chi}}\tilde{G})$	$(\partial_{1,\vec{\chi}}G)_l$ $\sigma((\partial_{1,\vec{\chi}}G)_l)$	$(\partial_{1,\vec{\chi}}G)_s$ $\sigma((\partial_{1,\vec{\chi}}G)_s)$
0.0	166.67×10^{-3} 118×10^{-6}		30×10^{-6} 42×10^{-6}	-565×10^{-6} 674×10^{-6}
0.1	165.98×10^{-3} 118×10^{-6}	-12.65×10^{-3} 1.18×10^{-3}	-12.27×10^{-3} 45×10^{-6}	-13.56×10^{-3} 1.04×10^{-3}
0.2	164.15×10^{-3} 117×10^{-6}	-24.63×10^{-3} 1.17×10^{-3}	-24.60×10^{-3} 44×10^{-6}	-24.69×10^{-3} 224×10^{-6}
0.3	161.06×10^{-3} 116×10^{-6}	-37.02×10^{-3} 1.16×10^{-3}	-36.94×10^{-3} 49×10^{-6}	-37.36×10^{-3} 564×10^{-6}
0.4	156.74×10^{-3} 115×10^{-6}	-49.56×10^{-3} 1.15×10^{-3}	-49.26×10^{-3} 51×10^{-6}	-49.38×10^{-3} 151×10^{-6}
0.5	151.14×10^{-3} 113×10^{-6}	-60.82×10^{-3} 1.13×10^{-3}	-61.37×10^{-3} 56×10^{-6}	-65.64×10^{-3} 4.41×10^{-3}
0.6	144.58×10^{-3} 111×10^{-6}	-70.85×10^{-3} 1.11×10^{-3}	-72.88×10^{-3} 60×10^{-6}	-72.93×10^{-3} 420×10^{-6}
0.7	136.97×10^{-3} 109×10^{-6}	-82.98×10^{-3} 1.08×10^{-3}	-82.94×10^{-3} 63×10^{-6}	-83.11×10^{-3} 134×10^{-6}
0.8	127.98×10^{-3} 106×10^{-6}	-93.10×10^{-3} 1.05×10^{-3}	-91.40×10^{-3} 69×10^{-6}	-91.79×10^{-3} 278×10^{-6}
0.9	118.35×10^{-3} 102×10^{-6}		-97.48×10^{-3} 68×10^{-6}	-97.96×10^{-3} 497×10^{-6}

Table 6.6 – Validations for different position of observation (from $(0, 0, 0)$ to $(0.45l_0, 0, 0)$), for a number of Monte-Carlo realization $n_{MC} = 10000000$.

Conclusion and perspectives

Conclusion

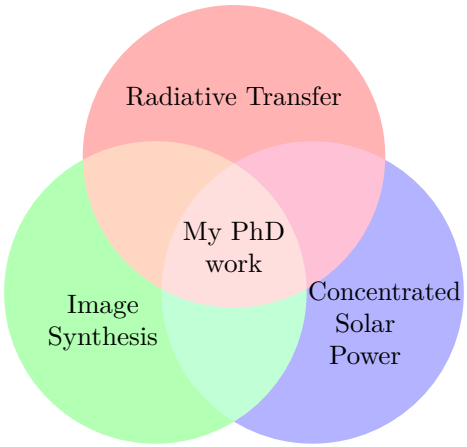


Figure 1 – The general positioning of this thesis.

In these three years of Ph.D. life, the author works between the interface of the CSP, the physics of radiative transfer, and the community of image synthesis (Fig.1). Because of the multi-disciplinary character of this thesis, it is concluded in these three viewpoints.

In the viewpoint of the CSP, a method to estimate sensitivities in a STP system is provided in Chapter.5. It extends the first attempt to estimate sensitivities suggested by the Ph.D. work of De la Torre, which is valid only when shadowing and blocking are neglected. The present work fills the blank concerning sensitivity estimation in the CSP community. The sensitivities of the translations, rotations and the size of each heliostat to the impacting power on the receiver can be estimated accurately. The vectorization strategy makes estimating thousands of sensitivities feasible within a reasonable calculating time. Also, each sensitivity can be interpreted physically, telling how the physical events in STP system (blocking, spillage, shadowing, etc.) affect the sensitivities, making some more sensitive than others. All kinds of STP systems with flat mirrors can benefit from this method. With the information provided by this method, engineers and researchers can:

1. Analyse the optical field of a STP system and improve the optical performance by adjusting the more sensitive parameters during operation.

2. Realize a detailed analysis of the optical field. The information on the contributions of physical events (blocking, spillage, shadowing, etc.) to the sensitivities is provided, helping engineers understand better the optical system and perform the sensitivity analysis.

3. Couple the information on sensitivities (the gradient) with a gradient-based algorithm for optimization purposes. This is of great interest to the CSP community since optimization mainly relies on gradient-free algorithms such as genetic algorithms or particle swarm optimization.

In the viewpoint of physics radiative transfer, the idea of vectorization has been proposed and implemented after the recent Ph.D. work of Paule Lapeyre [31–34]. Thanks to this vectorization strategy, the sensitivity model method can be applied to cases involving complex, triangulated geometries with a reasonable calculating time and less coding effort.

Nevertheless, a technique of transforming line sources into surface sources is presented in Chapter.6. This technique helps to avoid line sampling and vectorize the observable and its spatial derivation estimation completely.

In the viewpoint of image synthesis, all results obtained in this thesis are based on the computer graphic libraries in the community of image synthesis.

The technique studied in Chapter.6 helps to avoid the “sampling on segments,” which is regarded as a complex subject in the community. The researchers of image synthesis provide some complicated methods to overcome this difficulty [64]. However, herein, we try to solve this problem physically (by transforming the line sources into surface sources).

In Chapter.4, by applying the strategy of vectorization, a sensitivity image is rendered at the price of a low supplementary computation effort when the image of intensity is rendered. The image of sensitivity helps perform optimization and machine learning in the community of image synthesis[26].

Moreover, the literature on image synthesis about estimating sensitivities by Monte-Carlo is reviewed. Although the vocabularies and the viewpoints are different, we compared their methods and ours (developed in the EDstar group[1]) in Chapter.3. It is exciting to find out that even though the viewpoints are different, the method of differentiable rendering used by the image synthesis community and the method of sensitivity model converge to the same algorithm in the example developed in Chapter.3.

Perspectives

This Ph.D. work inspires the following perspectives:

1. The method developed in Chapter.5 is limited to flat mirrors. However, many STP systems are constructed based on curved mirrors (hemispheric or parabolic). There are two ways to describe the curvature of heliostat: 1) using the quadric surface of curved heliostat; 2) using a meshed geometry composed of triangles. Further work is required to highlight the differences and/or the practicability of the two approaches.

2. Even if it is possible to estimate only the impacting power at the top of a STP, it would also be possible to build a "flux map" of sensitivity. This would be useful to avoid the "hot spot" and to homogenize the flux on the receiver.

[1] http://www.edstar.cnrs.fr/prod/fr/

3. The sensitivities provided by our method have uncertainties. To our knowledge, uncertainties are not taken into account in gradient-based optimization algorithms. This would become crucial when the gradient tends to zero since it would not be possible to distinguish if the estimated gradient is positive or negative. A naive way to handle it would be to increase the Monte Carlo iterations. However, this approach requires more development.

4. In Chapter.6, the convergence problem is distinguished when the technique of transforming line sources to surface sources is applied in a scattering medium. Applying variance reduction methods might be a potential solution for this convergence problem.

5. The study of Chapter.6 is restricted to convex geometries. Applying the technique of transforming line sources to surface sources in a non-convex geometry is challenging. Stokes' theorem can not be applied directly when the triangles are blocked from each others.

6. Finally, the method of differentiable rendering is well adapted to complex geometry. Applying it in CSP applications can be a promising way of estimating sensitivities. The main challenge would be linking the different research problems in the two communities. In image synthesis, the main challenge is to render images which means that algorithms are based on reverse ray-tracing. Therefore, efforts are required to adapt those algorithms to the CSP field.

A

Vectorized transport models

A.1 Introduction

The idea of vectorization is that, instead of propagating a scalar as the descriptor of the transport model, we propagate a vector as the descriptor of the transport. In this appendix, we build the transport models in a vectorized form (see Fig.1). This way of vectorization requires great formal efforts. Further studies are needed if we want to apply it to specific applications.

In this appendix, we will firstly introduce the method of sensitivity model developed during the PhD work of Paule Lapeyre (Lapeyre, 2021b; Lapeyre et al., 2019). In this section, the general transport model of intensity will be reviewed. Depending on the need of different applications, observables are estimated based on the general transport model of intensity. Herein, the sensitivities of an observable with respect to different parameters are also estimated, following the method of sensitivity model. The method of sensitivity model aims to build a transport model for the sensitivity of intensity, which is similar to the transport model of intensity. The observable is usually formulated as the spatial or angular integral of the intensity. While following the method of sensitivity model, the sensitivity of an observable is usually formulated as the spatial or angular integral of the sensitivity of the intensity. Therefore, we introduce then the general transport model for the sensitivity of intensity. However, the model of sensitivity is coupled with two other quantities: the spatial derivative of intensity and the angular derivative of intensity. Their definitions are also given, and the general transport models are introduced respectively. Finally, the estimation of the sensitivity of an observable requires solving the four coupled models of transport: the model of intensity, the model of sensitivity, the model of spatial derivative, and the model of angular derivative.

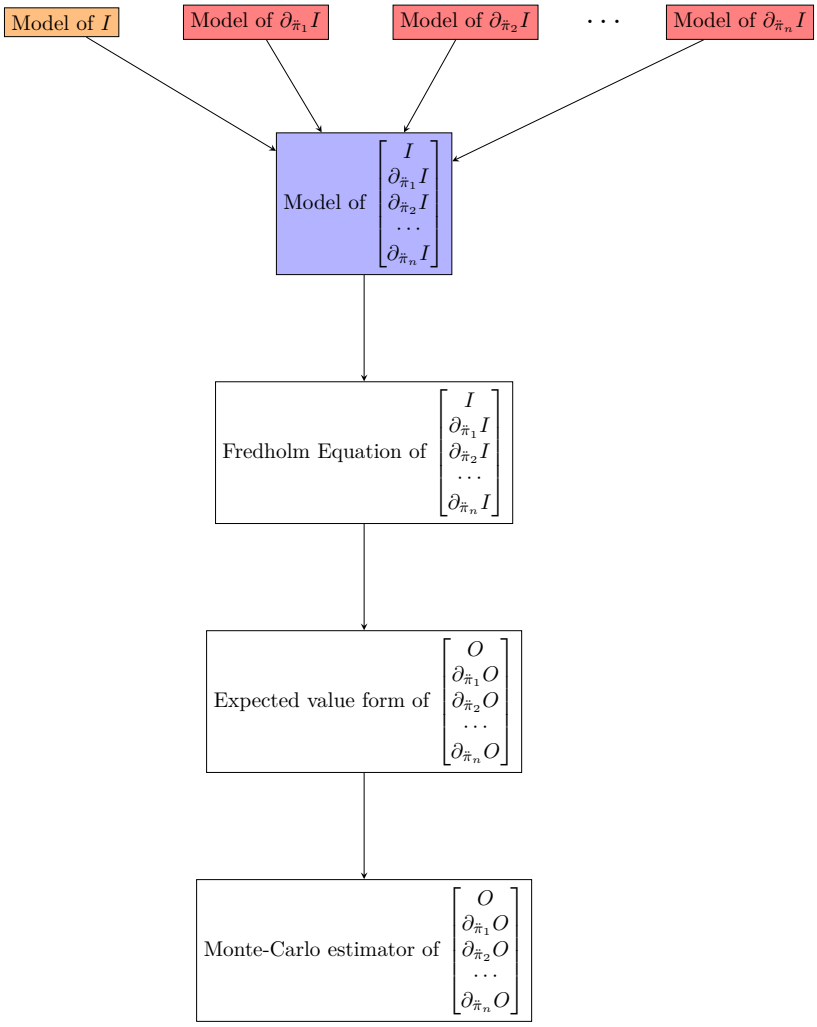


Figure 1 – A roadmap for the first way of vectorization. It aims to formulate a vectorized transport model and formulate the Fredolhme equation, expected value form, and the Monte-Carlo estimator in a vectorized form. I is the intensity, $\ddot{\pi}_1$, $\ddot{\pi}_2$, ..., $\ddot{\pi}_n$ are the parameters of the system and $\partial_{\ddot{\pi}_1} I$, $\partial_{\ddot{\pi}_2} I$, ..., $\partial_{\ddot{\pi}_n} I$ are the sensitivities corresponding. O is observable, which is usually formulated as the integral of intensity I. The sensitivity of O is usually formulated as the integral of the sensitivity of intensity.

After that, we will introduce the strategy of vectorization. The four models introduced describe the transport of four scalars: intensity, sensitivity, spatial derivative, and angular derivative. Following the strategy of vectorization, we simply rewrite the four models into a vectorized form, where the four models are built for a vector of sensitivities, a vector of spatial derivatives, and a vector of angular derivatives. This simple gesture allows us to transport the descriptors in a vectorized form. For example, instead of transporting each sensitivity corresponding to a parameter by its own path in the system, we can transport a vector of sensitivities corresponding to each parameter by only one pathway.

A.2 The method of sensitivity model

The method of sensitivity model aims to estimate the sensitivity of an observable with respect to a parameter in a radiative system. In this section, we review this method developed

by Paule Lapeyre during her PhD work [31–34]. It has already been introduced briefly in Chapter.3, and it will be introduced in detail in this section.

Beginning with the definition of a general radiative system, the general model of intensity will be introduced. After that, observables and the sensitivities of observables will be formulated in integral forms. The transport model of the sensitivity of intensity is then introduced, from which the sensitivity of the observable can be estimated. Lastly, the transport model of sensitivity is coupled with two other transport models: the transport model of spatial derivative and the transport model of angular derivative, which are also introduced. The estimation of an observable needs to solve the model of intensity. While the estimation of the sensitivity of an observable needs to solve the four mentioned coupled models.

A.2.1 Definition of a radiative system

We consider a radiative system. The descriptor of the system is the intensity $I \equiv I(\vec{x}, \vec{\omega}, \ddot{\pi})$ which has 3 independent variables:

- The position $\vec{x}$ is a vector of $\mathbb{R}^3$ that belongs to the set $\mathcal{G} \subset \mathbb{R}^3$. The geometric domain is bounded by $\partial\mathcal{G}$. The set $\mathcal{G}$ includes then the interior of the geometric domain and its boundary: $\mathcal{G} = \mathring{\mathcal{G}} \cup \partial\mathcal{G}$.
- The direction of propagation $\vec{\omega}$ is an element of the unit sphere $\mathcal{S}$.
- The parameter $\ddot{\pi}$ is an element of $\mathbb{R}$. The intensity in the system is sensitive to a parameter for all positions $\vec{x}$ of $\mathcal{G}$ and every direction $\vec{\omega}$ of $\mathcal{S}$. It will then be referred as $I \equiv I(\vec{x}, \vec{\omega}, \ddot{\pi})$ and the domain $\mathcal{G} \equiv \mathcal{G}(\ddot{\pi}) = \mathring{\mathcal{G}}(\ddot{\pi}) \cup \mathcal{G}(\ddot{\pi})$.

A.2.2 Model of intensity

In the medium, noting $\mathcal{C}$ as the collision operator, the stationary RTE has already been announced in Chapter.1.

$$\vec{\omega} \cdot \partial_1 I = \mathcal{C}[I] + \mathrm{S} \qquad \vec{x} \in \mathcal{G} \tag{A.1}$$

with

$$\mathcal{C}[I(\vec{x}, \vec{\omega})] = -k_a(\vec{x}) I(\vec{x}, \vec{\omega}) - k_s(\vec{x}) I(\vec{x}, \vec{\omega}) + k_s(\vec{x}) \int_{\Omega'} \mathcal{P}(-\vec{\omega}' | \vec{x}, -\vec{\omega}) d\vec{\omega}' I(\vec{x}, \vec{\omega}') \tag{A.2}$$

k_a is the absorption coefficient, and k_s is the scattering coefficient. S is the source of emission in the medium. It is null when the medium is considered "cold". The function of $\mathcal{P}(-\vec{\omega}'|\vec{x}, -\vec{\omega})$ is called the *Phase-function*. It characterizes the probability of the fact that a photon is scattered to the direction $-\vec{\omega}'$, coming from the direction $-\vec{\omega}$. This formulation favors a reciprocal/adjoint interpretation thanks to the micro-reversibility relation $\mathcal{P}(-\vec{\omega}' | -\vec{\omega}) = \mathcal{P}(\vec{\omega} | \vec{\omega}')$.

On the boundary, noting $\mathcal{C}_b$ as the reflection operator and $\mathring{I}$ as the source of emission on the boundary, the incoming intensity equation is:

$$I = \mathcal{C}_b[I] + \mathring{I} \qquad \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \tag{A.3}$$

with

$$\mathcal{C}_b[I] = \rho(\vec{x}, -\vec{\omega}) \int_{2\pi(\vec{n})} \mathcal{P}_{\Omega',b}(-\vec{\omega}'|\vec{x}, -\vec{\omega}) d\vec{\omega}' I(\vec{x}, \vec{\omega}') \tag{A.4}$$

where $\vec{n}$ is the normal to the boundary at $\vec{x}$, $\vec{\omega}$ the direction within the hemisphere 2π characterised by $\vec{n}$, $\rho(\vec{x}, -\vec{\omega})$ the surface reflectivity for a photon impacting the boundary in direction $-\vec{\omega}$, and $\mathcal{P}_{\Omega',b}$ the probability density that the reflection direction is $-\vec{\omega}'$ for a photon reflected at $\vec{x}$ coming from direction $-\vec{\omega}$ (the product of $\rho\mathcal{P}_{\Omega',b}$ is called BRDF).

To summarize, the model of transport for intensity is:

$$\begin{cases} \vec{\omega} \cdot \partial_1 I = \mathcal{C}[I] + \mathrm{S} & \vec{x} \in \mathcal{G} \\ I = \mathcal{C}_b[I] + \mathring{I} & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.5}$$

with S the source in the medium (Fig.2) and $\mathring{I}$ the source on the boundaries (Fig.3). Also, distinguished from the sources of the system, I is the transport term of intensity.

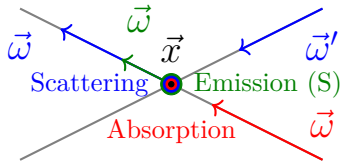


Figure 2 – Source (emission) and collisions (absorption and scattering) within the volume at point $\vec{x}$.

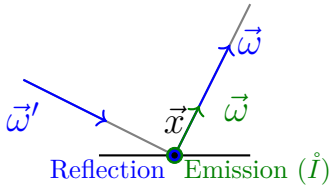


Figure 3 – Source (emission) and collisions (reflection) on the boundary at point $\vec{x}$. The collision of absorption is described by the reflectivity $\rho \in [0, 1]$, which is a factor of intensity reflected.

A.2.3 Observable in a radiative system

Some typical observables in a radiative system are reviewed in Chapter.1. Herein, we have a brief reminder.

The observable can be simply the descriptor itself: intensity I on a location $\vec{x}_{obs}$ towards a direction $\vec{\omega}_{obs}$: $I(\vec{x}_{obs}, \vec{\omega}_{obs}, \ddot{\pi})$. Also, it can be some macroscopic value. The transformation from the mesoscopic descriptor (intensity I) to different macroscopic values is done by integrations.

Scalar flux G $[\mathbf{w \cdot m^{-2}}]$ The irradiance of a unit surface is the integral of I for all incoming solid angles:

$$G(\vec{x}_{obs}) = \int_{\Omega} I(\vec{x}_{obs}, \vec{\omega}, \ddot{\pi}) d\vec{\omega} \tag{A.6}$$

When we consider a 3D radiative system, $\Omega = 4\pi$.

Total incident irradiance ϕ $[\mathbf{w \cdot m^{-2}}]$ The radiance flux is the energy flux traveling perpendicularly to a unit surface:

$$\phi(\vec{x}_{obs}) = \int_{\Omega} I(\vec{x}_{obs}, \vec{\omega}, \ddot{\pi}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} \tag{A.7}$$

Impacting power P $[\mathbf{w}]$ The impacting power on a surface is the integral of radiance flux over all the surface:

$$P = \int_{\mathcal{S}} \int_{\Omega} I(\vec{x}, \vec{\omega}, \ddot{\pi}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} d\vec{x} \tag{A.8}$$

where $\mathcal{S}$ is the area of the surface being impacted.

A.2.4 Sensitivity of an observable with respect to a parameter

Firstly, we define the sensitivity of intensity $s(\vec{x}, \vec{\omega}, \ddot{\pi})$. It is the derivative of intensity with respect to a parameter $\ddot{\pi}$:

$$s(\vec{x}, \vec{\omega}, \ddot{\pi}) = \partial_{\ddot{\pi}} I(\vec{x}, \vec{\omega}, \ddot{\pi}) = \partial_3 I(\vec{x}, \vec{\omega}, \ddot{\pi}) \tag{A.9}$$

The observables are expressed as the integral of the descriptor I in Section.A.2.3. We now differentiate the observables introduced in Section.A.2.3, with respect to the parameter $\ddot{\pi}$. We will find that the sensitivity of the observable with respect to the parameter $\ddot{\pi}$ is expressed as the integral of s, the sensitivity of intensity defined in Eq.A.9.

Sensitivity of irradiance $\partial_{\ddot{\pi}} G$ The sensitivity of the irradiance of a unit surface is the integral of s for all incoming solid angles:

$$\partial_{\ddot{\pi}} G(\vec{x}_{obs}) = \int_{\Omega} s(\vec{x}_{obs}, \vec{\omega}, \ddot{\pi}) d\vec{\omega} \tag{A.10}$$

When we consider a 3D radiative system, $\Omega = 4\pi$.

Sensitivity of radiance flux $\partial_{\ddot{\pi}} \phi$ The sensitivity of the radiance flux is the sensitivity of energy flux traveling perpendicularly to a unit surface:

$$\partial_{\ddot{\pi}} \phi(\vec{x}_{obs}) = \int_{\Omega} s(\vec{x}_{obs}, \vec{\omega}, \ddot{\pi}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} \tag{A.11}$$

Sensitivity of impacting power P The sensitivity of impacting power on a surface is the integral of the sensitivity of radiance flux over all the surface:

$$\partial_{\ddot{\pi}} P = \int_{\mathcal{S}} \int_{\Omega} s(\vec{x}, \vec{\omega}, \ddot{\pi}) |\vec{\omega} \cdot \vec{n}| d\vec{\omega} d\vec{x} \tag{A.12}$$

where $\mathcal{S}$ is the area of the surface being impacted.

It is noted that the development from the expressions of observable in Section.A.2.3 to the expressions of sensitivity beyond is mathematically straightforward, based on the definition of the sensitivity of intensity s (in Eq.A.9). However, in the framework of the method of sensitivity model, Paule Lapeyre tried to read the expressions of sensitivity physically: The sensitivity of intensity s and also be regarded as a descriptor of a radiative system, and the sensitivities of the observable are then the measure of this descriptor. Therefore, it is decided to build a transport model for s. The way of building a model for s is then differentiating the transport equation (the RTE) and the boundary conditions of the model of intensity (Eq.A.5) with respect to the parameter $\ddot{\pi}$.

A.2.5 Model of sensitivity

The model of sensitivity is developed in [32–34]. The sensitivity of intensity s is used as the descriptor of this model. In the medium, s is then absorbed, scattered in the medium, governed by the same collision operator $\mathcal{C}$ in the model of intensity. On the boundary, s is reflected, absorbed, and governed by the same reflection operator $\mathcal{C}_b$ in the model of intensity:

$$\begin{cases} \vec{\omega} \cdot \partial_1 s = \mathcal{C}[s] + \mathrm{S}_{\ddot{\pi}}[I] & \vec{x} \in \mathcal{G} \\ s = \mathcal{C}_b[s] + \mathring{s}[I, \partial_{1,\vec{\chi}} I, \partial_{2,\vec{\gamma}} I] & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.13}$$

with the source of sensitivity in the medium $\mathrm{S}_{\ddot{\pi}}[I]$ (Fig.4) and the source of sensitivity on the boundary $\mathring{s}[I, \partial_{1,\vec{\chi}} I, \partial_{2,\vec{\gamma}} I]$ (Fig.5).

In the model of intensity, the sources in the medium S and on the boundary $\mathring{I}$ are given (functions of the properties in the medium and on the boundaries). But in the model of sensitivity, the sources $\mathrm{S}_{\ddot{\pi}}$ and $\mathring{s}$ are coupled with the model of I, $\partial_{1,\vec{\chi}} I$, and $\partial_{2,\vec{\gamma}} I$, which are respectively the model of intensity, the model of spatial derivative, and the model of angular derivative. More details can be found in [34].

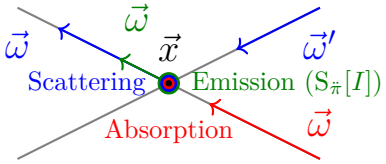


Figure 4 – Source (emission) and collisions (absorption and scattering) within the volume at point $\vec{x}$. The source of sensitivity ($\mathrm{S}_{\ddot{\pi}}$) is a function of intensity at this point $\vec{x}$ and following the same direction $\vec{\omega}$. The model of s is then coupled with the model of I.

A.2.6 Model of spatial derivative

The spatial derivative is made along a given direction, namely along a unit vector $\vec{\chi}$, which means that:

$$\partial_{1,\vec{\chi}} I = \vec{\chi} \cdot \vec{\nabla} I = \partial_{1,\vec{\chi}} I(\vec{x}, \vec{\omega}) = \lim_{\tau \to 0} \frac{I(\vec{x} + \tau\vec{\chi}) - I(\vec{x}, \vec{\omega})}{\tau} \tag{A.14}$$

Therefore, the spatial derivative $\partial_{1,\vec{\chi}} I$ has tree independent variables: $(\vec{x}, \vec{\omega}, \vec{\chi})$ they are the position $\vec{x}$, the direction of propagation $\vec{\omega}$ and the direction of differentiation $\vec{\chi}$ (Fig.6).

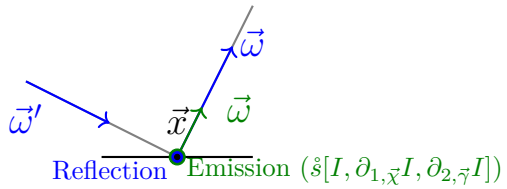


Figure 5 – Source (emission) and collisions (reflection) on the boundary at point $\vec{x}$. The collision of absorption is described by the reflectivity $\rho \in [0, 1]$, which is a factor of intensity reflected. The source of sensitivity on the boundary ($\mathring{s}$) is a function of intensity, the spatial derivative of intensity and angular derivative of intensity at this point $\vec{x}$ and following the same direction $\vec{\omega}$. The model of s is then coupled with the model of I, $\partial_{1,\vec{\chi}} I$ and $\partial_{2,\vec{\gamma}} I$.

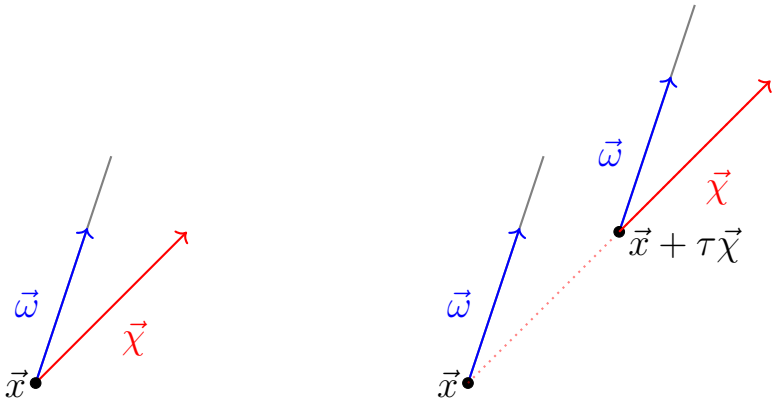


Figure 6 – The spatial derivative $\partial_{1,\vec{\chi}} I$ pictured as an elementary displacement following the direction of differentiation $\vec{\chi}$ (Eq.A.14). When picturing the physical model of spatial derivative of intensity, we need to draw the location $\vec{x}$, the direction of propagation $\vec{\omega}$ and also the direction of differentiation $\vec{\chi}$.

The model of spatial derivative is developed in [34]:

$$\begin{cases} \vec{\omega} \cdot \partial_1(\partial_{1,\vec{\chi}} I) = \mathcal{C}[\partial_{1,\vec{\chi}} I] + \mathrm{S}_{\vec{\chi}}[I] & \vec{x} \in \mathcal{G} \\ \partial_{1,\vec{\chi}} I = \beta \mathcal{C}_b[\partial_{1,\vec{u}} I] + \mathrm{S}_{b,\vec{\chi}}[I] & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.15}$$

with the source in the medium $\mathrm{S}_{\vec{\chi}}[I]$ (Fig.7).

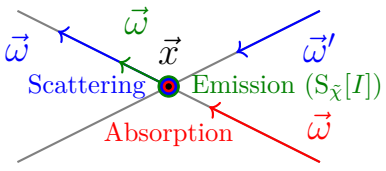


Figure 7 – Source (emission) and collisions (absorption and scattering) within the volume at point $\vec{x}$. The source of spatial derivative ($\mathrm{S}_{\vec{\chi}}$) is a function of intensity at this point $\vec{x}$ and following the same direction $\vec{\omega}$. The model of $\partial_{1,\vec{\chi}} I$ is then coupled with the model of I.

On the boundary, $\vec{\chi}$ is decomposed as the sum of two vectors, one parallel to the direction of $\vec{\omega}$, the other parallel to a direction $\vec{u}$, which is parallel to the boundary:

$$\vec{\chi} = \alpha\vec{\omega} + \beta\vec{u} \tag{A.16}$$

with

$$\alpha = \frac{\vec{\chi}\cdot\vec{n}}{\vec{\omega}\cdot\vec{n}}; \beta = ||\vec{\chi} - \alpha\vec{\omega}||; \vec{u} = \frac{\vec{\chi} - \alpha\vec{\omega}}{\beta} \tag{A.17}$$

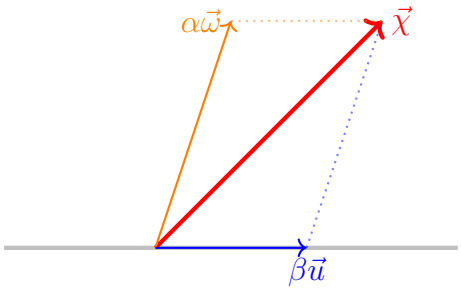
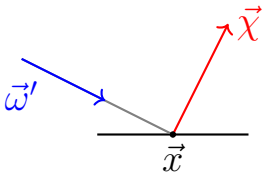
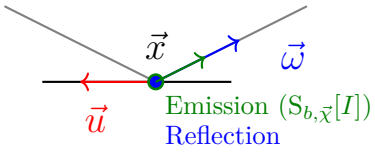


Figure 8 – Decomposition of the spatial velocity of deformation. The gradient of intensity projected to $\vec{\chi}$ is unknown. It is then decomposed to the direction of $\vec{\omega}$, and $\vec{u}$, where the gradients are described by RTE and the boundary condition.

The source on the boundary $\mathrm{S}_{b,\vec{\chi}}$ is coupled with the model of I (see more details in [34] and Fig.9). It is noted that in Eq.A.15, the reflection operator on the boundary $\mathcal{C}_b$ is applied to $\partial_{1,\vec{u}}I$, instead of $\partial_{1,\vec{\chi}}I$ (seeing Fig.9). In other words, the differentiation direction for the incoming spatial derivative is $\vec{\chi}$, but after the reflection, the differentiation direction becomes $\vec{u}$, which is parallel to the boundary. The corresponding physical image of reflection still exists, but the direction of the differentiation changes, and the collision operator is multiplied by β after each reflection (see Eq.A.15).



(a) Incoming spatial derivative.



(b) Reflected spatial derivative.

Figure 9 – Source (emission) and collisions (reflection) on the boundary at point $\vec{x}$. The collision of absorption is described by the reflectivity $\rho \in [0, 1]$, which is a factor of the spatial derivative reflected. The source of spatial derivative on the boundary ($\mathrm{S}_{b,\vec{\chi}}$) is a function of intensity at this point $\vec{x}$ and following the same direction $\vec{\omega}$. The model of $\partial_{1,\vec{\chi}}I$ is then coupled with the model of I.

$\mathrm{S}_{\vec{\chi}}[I]$ is the source of spatial derivative in the medium, and $\mathrm{S}_{b,\vec{\chi}}[I]$ is the source of spatial derivative on the boundaries. It is noted that the source $\mathrm{S}_{b,\vec{\chi}}[I]$ will include a surface part and a lineic part when the boundary has a discontinuity at the junction of two plane surfaces (for example two triangles). When the boundary is smooth, only the surface source exists, noted $\dot{\mathrm{S}}_{b,\vec{\chi}}$:

$$\mathrm{S}_{b,\vec{\chi}}[I] = \dot{\mathrm{S}}_{b,\vec{\chi}}[I]. \tag{A.18}$$

When the boundary is discretized as an ensemble of plane surfaces, typically an ensemble of triangles, then the intensity in a given direction becomes discontinuous at the edge of the surface. It induces line sources on the edges:

$$\mathrm{S}_{b,\vec{\chi}}[I] = \dot{\mathrm{S}}_{b,\vec{\chi}}[I] + \hat{\mathrm{S}}_{b,\vec{\chi}}[I]. \tag{A.19}$$

A.2.7 Model of angular derivative

The angular derivative corresponds to a rotation around the direction of the unit vector $\vec{\gamma}$:

$$\partial_{2,\vec{\gamma}} I = \lim_{\delta\phi \to 0} \frac{I(\vec{x}, \vec{\omega}^{\vec{\gamma}}_{\delta\phi}) - I(\vec{x}, \vec{\omega})}{\delta\phi} \tag{A.20}$$

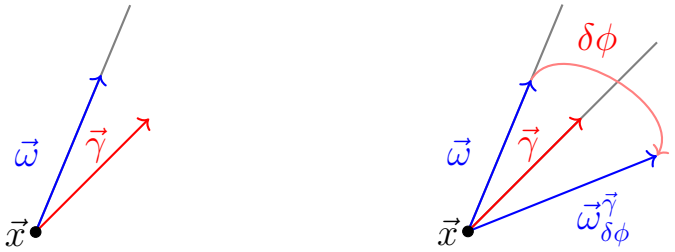


Figure 10 – The angular derivative $\partial_{2,\vec{\gamma}} I$ pictured as an elementary rotation around the rotation direction $\vec{\gamma}$ (Eq.A.14). When picturing the physical model of spatial derivative of intensity, we need to draw the location $\vec{x}$, the direction of propagation $\vec{\omega}$ and also the direction of rotation $\vec{\gamma}$.

The model of angular derivative is developed in [34]:

$$\begin{cases} \vec{\omega} \cdot \partial_1(\partial_{2,\vec{\gamma}} I) = \tilde{C}[\partial_{2,\vec{\gamma}} I] + \mathrm{S}_{\vec{\gamma}}[I] & \vec{x} \in \mathcal{G} \\ \partial_{2,\vec{\gamma}} I = \mathrm{S}_{b,\vec{\gamma}}[I] & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.21}$$

The transport of the angular derivative is not classical radiative transport. The corresponding collision operator $\tilde{C}$ is:

$$\tilde{C}[\partial_{2,\vec{\gamma}} I] = -k_a(\vec{x})\partial_{2,\vec{\gamma}} I - k_s(\vec{x})\partial_{2,\vec{\gamma}} I \tag{A.22}$$

and the source $\mathrm{S}_{\vec{\gamma}}$ in the medium is expressed as:

$$\mathrm{S}_{\vec{\gamma}}[I] = \partial_{2,\vec{\gamma}} C[I] + \partial_{2,\vec{\gamma}} \mathrm{S} - \tilde{C}[I] \tag{A.23}$$

There is no reflection for angular derivative on the boundary, but only a source of angular derivative $\mathrm{S}_{b,\vec{\gamma}}[I]$ can be found on the boundary (see [34] for more details).

A.2.8 A road map of the method

The objective of using the method of sensitivity model is to estimate the sensitivity of an observable with respect to a parameter. The model of sensitivity is built by differentiating the model of intensity. The observable is then formulated as an integral of intensity, which is the descriptor of the model of intensity. The sensitivity is then formulated as an integral of

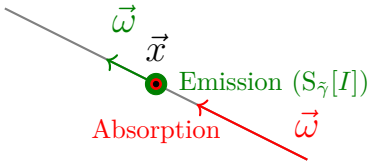


Figure 11 – Source (emission) and collisions (absorption) within the volume at point $\vec{x}$. The angular derivative is not scattered in the medium. The source of angular derivative ($S_{\vec{\gamma}}$) is a function of intensity at this point $\vec{x}$ and following the same direction $\vec{\omega}$. The model of $\partial_{2,\vec{\gamma}} I$ is then coupled with the model of I.

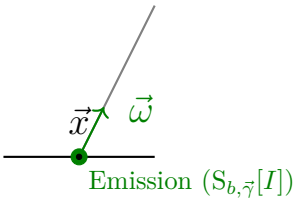


Figure 12 – Source (emission) on the boundary at point $\vec{x}$. The angular derivative is not reflected on the boundary, and the source of angular derivative on the boundary ($S_{b,\vec{\gamma}}$) is a function of intensity at this point $\vec{x}$ and following the same direction $\vec{\omega}$. The model of $\partial_{2,\vec{\gamma}} I$ is then coupled with the model of I.

the sensitivity of the intensity, which is the descriptor of the model of sensitivity. However, the model of sensitivity is coupled with three other models: the model of intensity, the model of spatial derivative, and the model of angular derivative. We deal with this coupling by the statistical technique of Double Randomization, which is already introduced in Chapter.1 and Chapter.3.

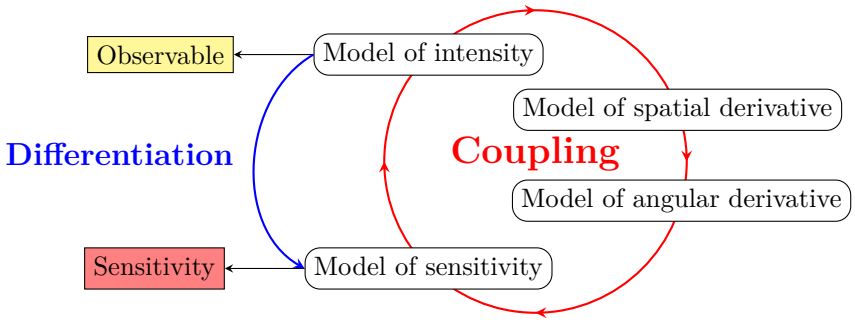


Figure 13 – A roadmap for the method of sensitivity model, of which the objective is to estimate the sensitivity of an observable.

A.3 Vectorization of the transports

Following the method of sensitivity model, in order to estimate the sensitivity of an observable in a radiative system, the four coupling models are built. They are built following the standard radiative transport, where the descriptors are scalars (I, s, $\partial_{1,\vec{\chi}} I$ or $\partial_{2,\vec{\gamma}} I$). This is because, as it is mentioned in Section.A.2, the parameter we focus on is an element $\ddot{\pi} \in \mathbb{R}$. However, we are usually interested in more than one parameter in a radiative system. We note all these parameters as $\ddot{\boldsymbol{\pi}} \in \mathbb{R}^n$ and $\ddot{\boldsymbol{\pi}} \equiv [\ddot{\pi}_j], \forall j \in \{1, 2, \ldots, n\}$, where n is the number of parameters. It will be convenient to formulate the model in vectorized forms rather than formulating

$4 \times n$ coupling models to be solved. In this chapter, the letters in bold are used to represent a vector (for example, $\ddot{\boldsymbol{\pi}}$ is a vector of $\ddot{\pi}_j$).

We define now the vector of sensitivity of $\ddot{\boldsymbol{\pi}}$:

$$\boldsymbol{s}(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) = \partial_{\ddot{\boldsymbol{\pi}}} I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) = \partial_3 I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}}) \tag{A.24}$$

with $\boldsymbol{s} \in \mathbb{R}^n$ and $\boldsymbol{s} \equiv [s_j], \forall j \in \{1, 2, \dots, n\}$.

The model of sensitivity in a vectorized form is then developed:

$$\begin{cases} \vec{\omega} \cdot \partial_1 \boldsymbol{s} = \mathcal{C}[\boldsymbol{s}] + \mathbf{S}_{\ddot{\boldsymbol{\pi}}}[I] & \vec{x} \in \mathcal{G} \\ \boldsymbol{s} = \mathcal{C}_b[\boldsymbol{s}] + \mathring{\boldsymbol{s}}[I, \partial_{1,\boldsymbol{\chi}} I, \partial_{2,\boldsymbol{\gamma}} I] & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.25}$$

where $\mathbf{S}_{\ddot{\boldsymbol{\pi}}}[I]$ is the source in the medium in a vectorized form: $\mathbf{S}_{\ddot{\boldsymbol{\pi}}} \equiv [\mathrm{S}_{\ddot{\pi}_j}]$ and $\mathring{\boldsymbol{s}}$ is the source on the boundary in a vectorized form: $\mathring{\boldsymbol{s}} \equiv [\mathring{s}_j]$. Also, the source $\mathring{\boldsymbol{s}}$ is a function of I, $\partial_{1,\boldsymbol{\chi}} I$ and $\partial_{2,\boldsymbol{\gamma}} I$, where $\boldsymbol{\chi} \equiv [\vec{\chi}_j]$ and $\boldsymbol{\gamma} \equiv [\vec{\gamma}_j]$.

$\partial_{1,\boldsymbol{\chi}} I$ is the spatial derivative in a vectorized form:

$$\partial_{1,\boldsymbol{\chi}} I = \boldsymbol{\chi} \cdot \vec{\nabla} I \tag{A.26}$$

and $\partial_{2,\boldsymbol{\gamma}} I$ is the angular derivative in a vectorized form:

$$\partial_{2,\boldsymbol{\gamma}} I = \lim_{\delta\phi \to 0} \frac{I(\vec{x}, \vec{\omega}^{\boldsymbol{\gamma}}_{\delta\phi}) - I(\vec{x}, \vec{\omega})}{\delta\phi} \tag{A.27}$$

The model of spatial derivative in a vectorized form is then developed:

$$\begin{cases} \vec{\omega} \cdot \partial_1(\partial_{1,\boldsymbol{\chi}} I) = \mathcal{C}[\partial_{1,\boldsymbol{\chi}} I] + \mathbf{S}_{\boldsymbol{\chi}}[I] & \vec{x} \in \mathcal{G} \\ \partial_{1,\boldsymbol{\chi}} I = \boldsymbol{\beta} \mathcal{C}_b[\partial_{1,\boldsymbol{u}} I] + \mathbf{S}_{b,\boldsymbol{\chi}}[I] & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.28}$$

with $\boldsymbol{\beta} \in \mathbb{R}^n$ and $\boldsymbol{u} \in \mathbb{R}^{3 \times n}$. Each component of $\boldsymbol{\beta}$ and $\boldsymbol{u}$ are referred to Eq.A.16 and Eq.A.17. $\mathbf{S}_{\boldsymbol{\chi}}$ and $\mathbf{S}_{b,\boldsymbol{\chi}}$ are then the sources in the medium and on the boundaries respectively: $\mathbf{S}_{\boldsymbol{\chi}} \equiv [\mathrm{S}_{\vec{\chi}_j}]$ and $\mathbf{S}_{b,\boldsymbol{\chi}} \equiv [\mathrm{S}_{b,\vec{\chi}_j}]$.

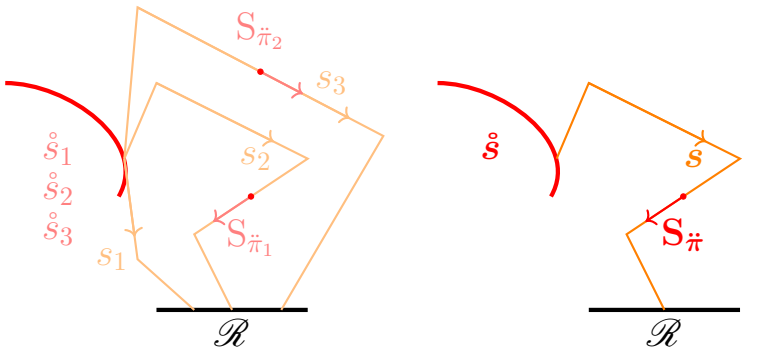
Also, the model of angular derivative in a vectorized form is developed:

$$\begin{cases} \vec{\omega} \cdot \partial_1(\partial_{2,\boldsymbol{\gamma}} I) = \tilde{\mathcal{C}}[\partial_{2,\boldsymbol{\gamma}} I] + \mathbf{S}_{\boldsymbol{\gamma}}[I] & \vec{x} \in \mathcal{G} \\ \partial_{2,\boldsymbol{\gamma}} I = \mathbf{S}_{b,\boldsymbol{\gamma}}[I] & \vec{x} \in \partial\mathcal{G};\ \vec{\omega} \cdot \vec{n} > 0 \end{cases} \tag{A.29}$$

with $\mathbf{S}_{\boldsymbol{\gamma}} \in [S_{\vec{\gamma}_j}]$ and $\mathbf{S}_{b,\boldsymbol{\gamma}} \in [S_{b,\vec{\gamma}_j}]$ the source in the medium and on the boundary respectively.

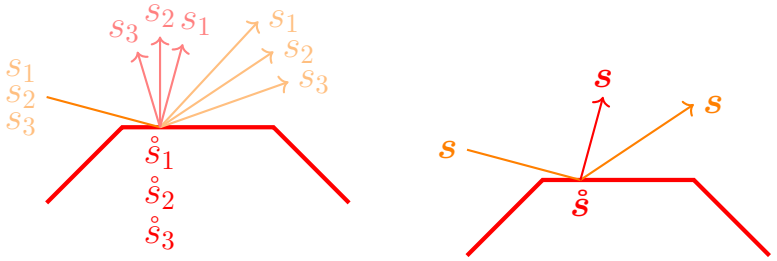
At this stage, the four coupling models in vectorized forms are built. Mathematically, the models are simply written in a vectorized form, but there is also a physical image attached to this reformulation. In fact, we are regrouping the propagations of the sources by their path spaces.

For the model of sensitivities, different sensitivities $s_1, s_2, s_3, \dots$ share the same path space in the medium and on the boundaries because they have the same model of transport in the medium and on the boundaries. Therefore, the transport of different sensitivities can be vectorized, seeing Fig.14. The propagation of the vectorized source $\mathring{\boldsymbol{s}}$ has, therefore, only one path space in the system.



(a) Transport in the medium before vectorization.

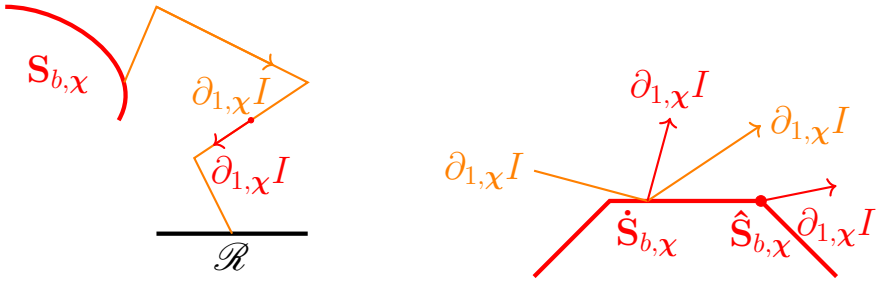
(b) Transport in the medium after vectorization.



(c) Transport on the boundary before vectorization.

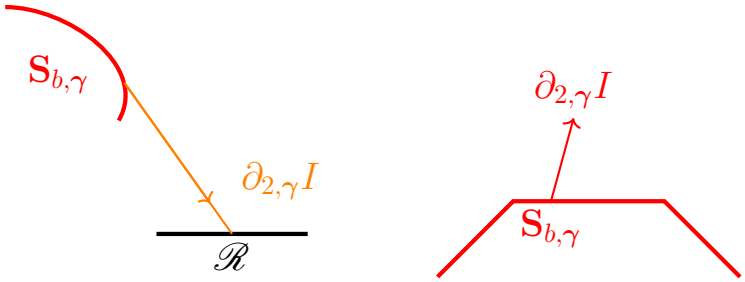
(d) Transport on the boundary after vectorization.

Figure 14 – The sensitivities propagate from the source (in red) to the sensor (noted $\mathscr{R}$ in black). Before vectorization, three models of transport for s are needed. After vectorization, only one model is needed to be solved. This is because the sensitivities s_1, s_2, s_3 in a radiative system have the same model of transport in the medium.



(a) Transport in the medium. **(b)** Transport on the boundaries.

Figure 15 – The spatial derivatives propagate from the source (in red) to the sensor (noted $\mathscr{R}$ in black). When the boundary is discontinued, the surface source $\dot{\mathbf{S}}_{b,\boldsymbol{\chi}}$ and the lineic source $\hat{\mathbf{S}}_{b,\boldsymbol{\chi}}$ have their own path space.



(a) Transport in the medium. **(b)** Transport on the boundaries.

Figure 16 – The angular derivatives propagate from the source (in red) to the sensor (noted $\mathscr{R}$ in black).

For the model of spatial derivative, different spatial derivatives $\partial_{1,\vec{\chi}_1} I$, $\partial_{1,\vec{\chi}_2} I$, $\partial_{1,\vec{\chi}_3} I$, ... share the same path space in the medium and on the boundaries, because they have the same model of transport in the medium and on the boundaries. Therefore, the transport of different spatial derivatives can be vectorized in the medium and on the boundaries, seeing Fig.15. When the boundary is discontinued, the vectorized source $\mathbf{S}_{b,\boldsymbol{\chi}}$ consists of two different sources: $\dot{\mathbf{S}}_{b,\boldsymbol{\chi}} \equiv [\dot{\mathrm{S}}_{b,\vec{\chi}_j}]$ on the surface and $\hat{\mathbf{S}}_{b,\boldsymbol{\chi}} \equiv [\hat{\mathrm{S}}_{b,\vec{\chi}_j}]$ on the edge:

$$\mathbf{S}_{b,\boldsymbol{\chi}} = \dot{\mathbf{S}}_{b,\boldsymbol{\chi}} + \hat{\mathbf{S}}_{b,\boldsymbol{\chi}} \tag{A.30}$$

As it is shown in Fig.15, the two sources are propagated in the medium and reflected on the boundaries. However, they have different path spaces because $\dot{\mathbf{S}}_{b,\boldsymbol{\chi}}$ is emitted on the surface and $\hat{\mathbf{S}}_{b,\boldsymbol{\chi}}$ is emitted on the edge.

For the model of angular derivative, different angular derivatives $\partial_{2,\vec{\gamma}_1} I$, $\partial_{2,\vec{\gamma}_2} I$, $\partial_{2,\vec{\gamma}_3} I$, ... share the same path space in the medium and on the boundaries, because they have the same model of transport in the medium and on the boundaries. Therefore, the transport of different spatial derivatives can be vectorized, seeing Fig.16. The propagation of the vectorized source $\mathbf{S}_{\boldsymbol{\gamma}}$ has, therefore, only one path space in the system.

Sometimes, we can vectorize them together. If we observe the RTE of intensity (Eq.A.1) and the RTE of sensitivity (Eq.A.25), they are almost the same, except that a source is emitted in the medium in Eq.A.1. Therefore, when the medium is "cold" (no emission in the medium), the intensity and the sensitivity will share the same model of

transport in the medium. The path space of I and of $\boldsymbol{s}$ will then be the same in the medium. The following equation of transport can be formulated:

$$\vec{\omega} \cdot \partial_1 \begin{bmatrix} I \\ \boldsymbol{s} \end{bmatrix} = C[\begin{bmatrix} I \\ \boldsymbol{s} \end{bmatrix}] \tag{A.31}$$

Furthermore, we observe the RTE of spatial derivative (Eq.A.28). When the medium is homogeneous (k_a and k_s and the phase function is independent of $\vec{x}$), the spatial derivative will share the same model of transport with the intensity and the sensitivity in the medium. The path space of $\partial_{1,\boldsymbol{\chi}} I$, I and $\boldsymbol{s}$ will then be the same in the medium:

$$\vec{\omega} \cdot \partial_1 \begin{bmatrix} I \\ \boldsymbol{s} \\ \partial_{1,\boldsymbol{\chi}} I \end{bmatrix} = -k_a \begin{bmatrix} I \\ \boldsymbol{s} \\ \partial_{1,\boldsymbol{\chi}} I \end{bmatrix} - k_s \begin{bmatrix} I \\ \boldsymbol{s} \\ \partial_{1,\boldsymbol{\chi}} I \end{bmatrix} + k_s \int_{\Omega'} \mathcal{P}(-\vec{\omega}' | -\vec{\omega}) d\vec{\omega}' \begin{bmatrix} I \\ \boldsymbol{s} \\ \partial_{1,\boldsymbol{\chi}} I \end{bmatrix} \tag{A.32}$$

Moreover, we observe the RTE of angular derivative (Eq.A.29). When the medium is purely absorbent ($k_s = 0$), the angular derivative will share the same model of transport with others in the medium. The path space of $\partial_{2,\boldsymbol{\gamma}} I$, $\partial_{1,\boldsymbol{\chi}} I$, I and $\boldsymbol{s}$ will then be the same in the medium:

$$\vec{\omega} \cdot \partial_1 \begin{bmatrix} I \\ \boldsymbol{s} \\ \partial_{1,\boldsymbol{\chi}} I \\ \partial_{2,\boldsymbol{\gamma}} I \end{bmatrix} = -k_a \begin{bmatrix} I \\ \boldsymbol{s} \\ \partial_{1,\boldsymbol{\chi}} I \\ \partial_{2,\boldsymbol{\gamma}} I \end{bmatrix} \tag{A.33}$$

To sum up, the basic strategy of vectorization is to vectorize the propagation of the sources having the same path space.

We can vectorize the model of sensitivity, the model of spatial derivatives, and the model of angular derivatives separately. For example, for the sensitivities, instead of constructing statistic paths for each component $s_1, s_2, \ldots$, only one statistic path for $\boldsymbol{s}$ is then enough to solve the model.

Furthermore, the vectorization of different descriptors is not always possible. The corresponding strategy of vectorization is very flexible, depending on the specific problem.

In the following part of this chapter, we will practice this technique of vectorization in two examples: calculate the sensitivities of reflectivity and the sensitivities of geometric parameters. The strategy of vectorization is then adapted to these two examples.

B

Rendering an image of intensity and an image of sensitivity simultaneously by vectorized Monte-Carlo method

B.1 Introduction

In the first example of Chapter.4, the estimations of observable I and its sensitivity s_ρ are vectorized. In this section, we benefit from this vectorized form (Eq.4.15) to render an image of I and another image of s_ρ simultaneously. In other words, at the same time that an image of I is rendered, the image of s_ρ can also be rendered at a price of a low supplementary computation effort. The image s_ρ can serve as necessary information to process optimization and machine learning process in the community of image synthesis [26].

B.2 Render an image of intensity

Our algorithm of rendering is based on the calculation of the mean intensity (which is our observable) captured by each pixel of a grid within a solid angle corresponding to this pixel:

$$< I >_j = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \{I(\vec{x}, \vec{\omega}, \rho)\} \tag{B.1}$$

Eq.B.1 is a formulation of the mean intensity $< I >_j$ captured by a pixel of j as its index. This formulation can be read directly as an algorithm of Monte-Carlo estimation, consisting of uniformly sampling an initial direction $\vec{\omega}$ within a solid angle $\Delta\Omega_j$ corresponding to the pixel. This correspondence depends on the camera. In this example, we will use a pinhole

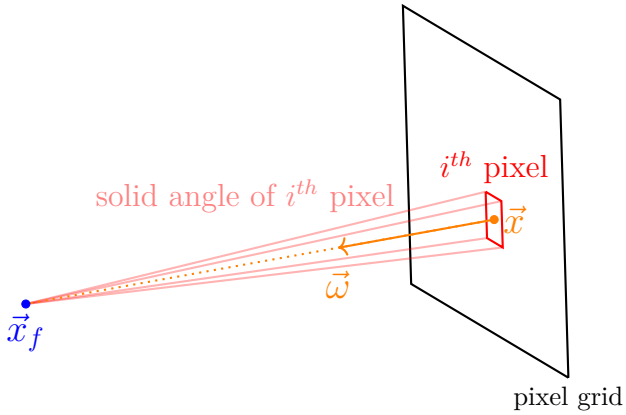


Figure 1 – Schema of a pinhole camera used in this example: the position of departure position of ray-tracing $\vec{x}$ and the direction of departure $\vec{\omega}$ from a given pixel is linked by the camera focal $\vec{x}_f$.

camera. The position of the departure of the ray from the pixel $\vec{x}$ and its initial direction are linked via a camera focal $\vec{x}_f$, seeing Fig.1 for the definition of a camera.

In Section.4.2.3, Eq.4.6 signifies the fact that the intensity $I(\vec{x},\vec{\omega},\rho)$ has three possible origins. It can be from the three boundary conditions, seeing Fig.2.

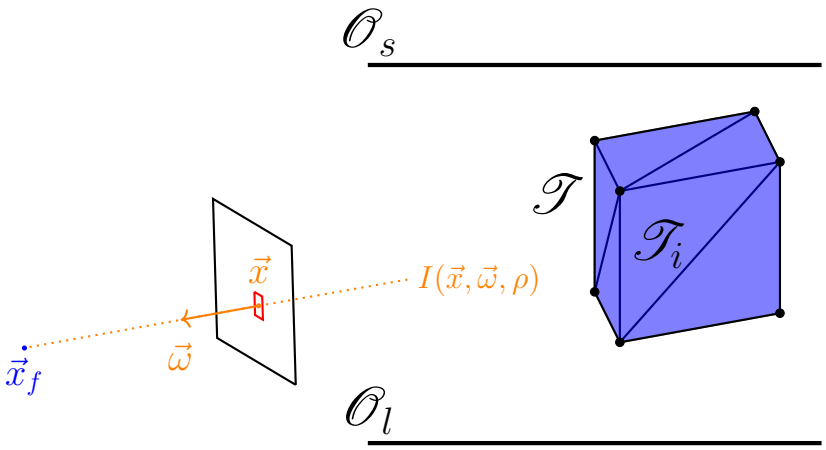


Figure 2 – A pinhole camera (Fig.1) is put in the scene. The intensity to be integrated is $I(\vec{x},\vec{\omega},\rho)$, the intensity propagated from $\vec{x}$, following $\vec{\omega}$. Three possible events can happen: the ray hits the $\mathscr{O}_s$, and $I(\vec{x},\vec{\omega},\rho)=1$; the ray hits the $\mathscr{O}_l$ and $I(\vec{x},\vec{\omega},\rho)=0$; the ray hits the $\mathscr{T}_i$, $I(\vec{x},\vec{\omega},\rho)=I_i(\vec{x}_p,\vec{\omega}_p,\rho)$.

Finally, the expected value form of $< I >_j$ can be formulated:

$$< I >_j = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \begin{cases} I_0 \mathscr{H}(\vec{x}_{hit} \in \mathscr{O}_s) + \\ 0 \mathscr{H}(\vec{x}_{hit} \in \mathscr{O}_l) + \\ \displaystyle\int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega'_p}(\vec{\omega}'_p) d\vec{\omega}'_p \rho I_i(\vec{x}_p,\vec{\omega}'_p,\rho) \mathscr{H}(\vec{x}_{hit} \in \mathscr{T}_i) \end{cases} \tag{B.2}$$

A Monte-Carlo algorithm estimating the mean intensity impacting the j^{th} pixel can be read from Eq.B.2. For each realization, we sample a direction of observation $\vec{\omega}$ following the pdf

$\frac{1}{\Omega_j}$, and we calculate the associated position of observation $\vec{x}$ according to the position focal point. After that, we proceed with a ray-tracing process. If the first intersection is found on $\mathscr{O}_s$, we count I_0 for this realization; if the first intersection is found on $\mathscr{O}_l$, we count 0; if the first intersection is found on $\mathscr{T}_i$, we will continue to search for the incoming intensity, sampling a direction following $\mathscr{P}_{\Omega'_p}$.

We will finally reformulate Eq.B.2 iteratively to interpret a final algorithm and an attached physical picture:

$$< I >_j = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \Big\{ \rho^0 I_0 \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{O}_s) + \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{T}_i) \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega_p}(\vec{\omega}_{p1}) d\vec{\omega}'_{p1} \left[\rho^1 I_0 \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{O}_s) + \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{T}_i) \dots \right] \Big\} \quad \text{(B.3)}$$

where $\vec{x}_{0,hit}$ and $\vec{x}_{1,hit}$ are the points of intersection after 0 and 1 reflection; $\vec{\omega}_{p1}$ is the direction of propagation after 1 reflection.

Systematically, for each Monte-Carlo realization, if we note $\vec{x}_{k,hit}$, $\forall k \in \mathbb{N}_0$ as the point of intersection after k reflections, when $\vec{x}_{k,hit} \in \mathscr{O}_s$, we count $\rho^k I_0$ for this realization; when $\vec{x}_{k,hit} \in \mathscr{O}_l$, we count 0; when $\vec{x}_{k,hit} \in \mathscr{T}$, we continue the $(k+1)^{th}$ reflection.

B.3 Render an image of sensitivity

Similar to the Eq.B.1, the rendering of an image of sensitivity s_ρ is based on the estimation of the mean sensitivity captured by each pixel of a grid within a solid angle corresponding to this pixel:

$$< s_\rho >_j = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \left\{ s_\rho(\vec{x}, \vec{\omega}, \rho) \right\} \quad \text{(B.4)}$$

In Section.4.2.5, Eq.4.12 signifies the fact that 2 events can happen when a ray is lanced from the position $\vec{x}$ towards the direction $-\vec{\omega}$: hitting or not hitting the object $\mathscr{T}$ (Fig.3).

Therefore, the expected value form of s_ρ can be formulated:

$$< s_\rho >_j = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \begin{cases} 0 \mathcal{H}(\vec{x}_{hit} \notin \mathscr{T}) + \\ \int_{2\pi(\vec{n}_i)} \mathscr{P}_{\Omega'_p}(\vec{\omega}'_p) d\vec{\omega}'_p \left(\rho s_{\rho i}(\vec{x}_p, \vec{\omega}'_p, \rho) + I_i(\vec{x}_p, \vec{\omega}'_p, \rho) \right) \mathcal{H}(\vec{x}_{hit} \in \mathscr{T}_i) \end{cases} \quad \text{(B.5)}$$

A Monte-Carlo algorithm estimating the mean sensitivity impacting the j^{th} pixel can be read from Eq.B.5. For each realization, we sample a direction of observation $\vec{\omega}$ following the pdf $\frac{1}{\Omega_j}$, and we calculate the associated position of observation $\vec{x}$ according to the position focal point. After that, we proceed with a ray-tracing process from $\vec{x}$ towards $-\vec{\omega}$. If the first intersection is not found on $\mathscr{T}$, we count 0 for this realization; if the first intersection is found on $\mathscr{T}_i$, we will continue to search for the incoming sensitivity s_ρ, sampling a direction $\vec{\omega}'_p$ following $\mathscr{P}_{\Omega'_p}$ and in the same time, we need to also search for the intensity I following the direction $\vec{\omega}'_p$. In other words, we continue the ray-tracing process after the reflection, searching for the incoming s_ρ the I at the same time.

We will herein reformulate Eq.B.5 iteratively to interpret a final algorithm:

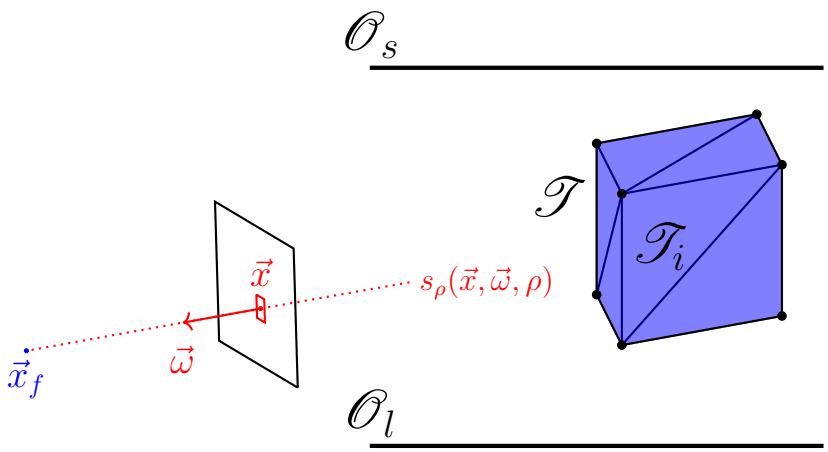


Figure 3 – A pinhole camera (Fig.1) is put in the scene. The sensitivity to be integrated is $s_\rho(\vec{x},\vec{\omega},\rho)$, the sensitivity of intensity propagated from $\vec{x}$, following $\vec{\omega}$. Two possible events can happen: the ray hits the $\mathscr{O}_s$ or $\mathscr{O}_l$, then $s_\rho(\vec{x},\vec{\omega},\rho)=0$; the ray hits the $\mathscr{T}_i$, $s_\rho(\vec{x},\vec{\omega},\rho)=s_{\rho i}(\vec{x}_p,\vec{\omega}_p,\rho)$.

$$
\begin{aligned}
< s_\rho >_j = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \Big\{ & 0\mathcal{H}(\vec{x}_{0,hit} \notin \mathscr{T}) \\
& + \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{T}_i) \int_{2\pi(\vec{n}_i)} \mathcal{P}_{\Omega_p}(\vec{\omega}_{p1}) d\vec{\omega}'_{p1} \big[\rho^0 I_0 \mathcal{H}(\vec{x}_{1,hit} \notin \mathscr{T}) + \\
& \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{T}_i) \int_{2\pi(\vec{n}_i)} \mathcal{P}_{\Omega_p}(\vec{\omega}_{p2}) d\vec{\omega}'_{p2} \big(2\rho^1 I_0 \mathcal{H}(\vec{x}_{2,hit} \notin \mathscr{T}) + \\
& \mathcal{H}(\vec{x}_{2,hit} \in \mathscr{T}_i) \dots \big) \big] \Big\}
\end{aligned}
\tag{B.6}
$$

where $\vec{x}_{0,hit}$, $\vec{x}_{1,hit}$, $\vec{x}_{2,hit}$ are the points of intersection after 0, 1 and 2 reflection(s) respectively; $\vec{\omega}_{p1}$ and $\vec{\omega}_{p2}$ are the directions of propagation after 1 and 2 reflection(s).

Systematically, for each Monte-Carlo realization, if we note $\vec{x}_{k,hit}$, $\forall k \in \mathbb{N}_0$ as the point of intersection after k reflection, when $\vec{x}_{k,hit} \notin \mathscr{T}$, we count $k\rho^{(k-1)} I_0$ for this realization; when $\vec{x}_{k,hit} \in \mathscr{T}$, we continue the $(k+1)^{th}$ reflection.

B.4 Render the two images simultaneously

In this stage, we have formulated the expected value form for estimating $< I >_j$ and $< s_\rho >_j$. Instead of proceeding two Monte-Carlo to estimate $< I >_j$ and $< s_\rho >_j$ for each pixel, in fact, only one Monte-Carlo per pixel is needed to render an image of s_ρ and an image of I. It is because the propagation of I and of s_ρ can be vectorized.

When we compare Eq.B.2 and Eq.B.5 (or compare Eq.B.3 or Eq.B.6), we found that their **domains of integration are exactly the same**. It means that they have the same path

space of propagation so that we can vectorize their propagations by simply writing Eq.B.2 and Eq.B.5 into a vectorized form:

$$\begin{bmatrix} < I >_j \\ < s_\rho >_j \end{bmatrix} = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \begin{cases} \begin{bmatrix} I_0 \\ 0 \end{bmatrix} \mathcal{H}(\vec{x}_{hit} \in \mathscr{O}_s) + \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix} \mathcal{H}(\vec{x}_{hit} \in \mathscr{O}_l) + \\ \int_{2\pi(\vec{n}_i)} \mathcal{P}_{\Omega'_p}(\vec{\omega}'_p) d\vec{\omega}'_p \begin{bmatrix} \rho I_i(\vec{x}_p, \vec{\omega}'_p, \rho) \\ \rho s_{\rho i}(\vec{x}_p, \vec{\omega}'_p, \rho) + I_i(\vec{x}_p, \vec{\omega}'_p, \rho) \end{bmatrix} \mathcal{H}(\vec{x}_{hit} \in \mathscr{T}_i) \end{cases} \tag{B.7}$$

By formulating a vectorized form, from an algorithmic point of view, the ray-tracing test for I and for s_ρ can be done by the same ray (Fig.4).

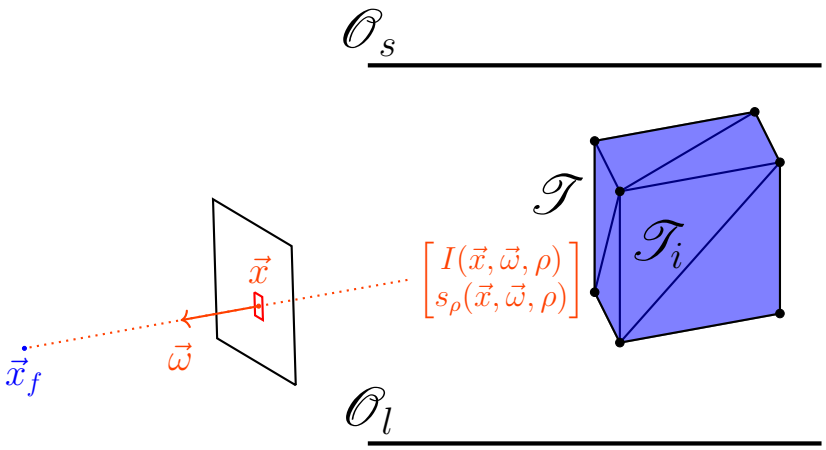


Figure 4 – A pinhole camera (Fig.1) is put in the scene. Instead of launching a ray for the estimation of I and another ray for the estimation of s_ρ. The same ray is used for the estimation of the vector of these two descriptors.

Finally, we reformulate Eq.B.7 iteratively:

$$\begin{aligned} \begin{bmatrix} < I >_j \\ < s_\rho >_j \end{bmatrix} = \int_{\Delta\Omega_j} \frac{1}{\Omega_j} d\vec{\omega} \Big\{ & \begin{bmatrix} I_0 \\ 0 \end{bmatrix} \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{O}_s) \\ & + \mathcal{H}(\vec{x}_{0,hit} \in \mathscr{T}_i) \int_{2\pi(\vec{n}_i)} \mathcal{P}_{\Omega_p}(\vec{\omega}_{p1}) d\vec{\omega}'_{p1} \Big[\begin{bmatrix} \rho I_0 \\ I_0 \end{bmatrix} \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{O}_s) + \\ & \mathcal{H}(\vec{x}_{1,hit} \in \mathscr{T}_i) \int_{2\pi(\vec{n}_i)} \mathcal{P}_{\Omega_p}(\vec{\omega}_{p2}) d\vec{\omega}'_{p2} \Big(\begin{bmatrix} \rho^2 I_0 \\ 2\rho I_0 \end{bmatrix} I_0 \mathcal{H}(\vec{x}_{2,hit} \in \mathscr{O}_s) + \\ & \mathcal{H}(\vec{x}_{2,hit} \in \mathscr{T}_i) \dots \Big) \Big] \Big\} \end{aligned} \tag{B.8}$$

Systematically, for each Monte-Carlo realization, if we note $\vec{x}_{k,hit}$, $\forall k \in \mathbb{N}_0$ as the point of intersection after k reflection, when $\vec{x}_{k,hit} \in \mathscr{O}_s$, we count $\rho^k I_0$ for intensity and $k\rho^{(k-1)} I_0$ for sensitivity for this realization; when $\vec{x}_{k,hit} \in \mathscr{O}_l$, we count 0 for intensity and sensitivity; when $\vec{x}_{k,hit} \in \mathscr{T}$, we continue the $(k+1)^{th}$ reflection. The corresponding Algorithm is the Algo.15.

Algorithm 15 Evaluation of $< I >_j$ and $< s_\rho >_j$ (for the j^{th} pixel)

Input: $\mathscr{T}$, n_{MC}, $\vec{x}_f$, $\mathscr{P}_i$

▷ n_{MC}: number of Monte-Carlo realization.
▷ $\mathscr{T}$: the triangulated geometry of an object.
▷ $\mathscr{P}_i$: the geometry of the j^{th} pixel.
▷ $\vec{x}_f$: the position of the focal point.

$W_I \leftarrow 0$, $W_s \leftarrow 0$ and $V_I \leftarrow 0$, $V_s \leftarrow 0$
▷ Initialize the sum and the sum of square
for $i_{MC} = 0 \rightarrow n_{MC}$ **do**
  Initialize the weight of Monte-Carlo $w_I \leftarrow 0$ and $w_s \leftarrow 0$
  Initialize the reflection counter $c \leftarrow 0$
  Initialize the keeprunning flag $keeprunning \leftarrow 1$
  Sample uniformly a position $\vec{x}$ on $\mathscr{P}_i$
  Calculate the direction based on $\vec{x}$ and $\vec{x}_f$: $\vec{\omega} \leftarrow \frac{\vec{x}-\vec{x}_0}{||\vec{x}-\vec{x_0}||}$
  while keeprunning **do**
    get the first intersection point of the ray $(\vec{x}, \vec{\omega})$: $\vec{x}_{hit}$
    if $\vec{x}_{hit} \in \mathscr{T}_i$ **then**
      $c \leftarrow c + 1$
      Sample a direction of diffuse reflection $\vec{\omega}'_p$ based on $\mathscr{P}'_\Omega(\vec{\omega}')$
      Update the position $\vec{x} \leftarrow \vec{x}_{hit}$
      Update the direction of propagation $\vec{\omega} \leftarrow \vec{\omega}'_p$
    else if $\vec{x}_{hit} \in \mathscr{O}_s$ **then**
      $w_I \leftarrow \rho^c I_0$
      $w_s \leftarrow c\rho^{c-1} I_0$
      $keeprunning \leftarrow 0$
    else
      $w_I \leftarrow 0$
      $w_s \leftarrow 0$
      $keeprunning \leftarrow 0$
    end if
  end while
  $W_I \leftarrow W_I + w_I$
  $V_I \leftarrow V_I + w_I^2$
  $W_s \leftarrow W_s + w_s$
  $V_s \leftarrow V_s + w_s^2$
  $i_{MC} \leftarrow i_{MC} + 1$
end for

Output: $< I >_j \leftarrow \frac{W_I}{n_{MC}}$, $\sigma[< I >_j] \leftarrow \sqrt{\frac{\frac{V_I}{n_{MC}} - (\frac{W_I}{n_{MC}})^2}{n_{MC}-1}}$, $< s_\rho >_j \leftarrow \frac{W_s}{n_{MC}}$, $\sigma[< s_\rho >_j] \leftarrow \sqrt{\frac{\frac{V_s}{n_{MC}} - (\frac{W_s}{n_{MC}})^2}{n_{MC}-1}}$

C

Boundary conditions of sensitivity on each triangle for Example 2 in Chapter.4

In order to define the boundary condition of $s_{i,j}$ on $\mathscr{T}_i$ (which is the sensitivity of intensity with respect to the j^{th} parameter on the i^{th} triangle), we differentiate the boundary condition of I, but this process is not that straight forward. This is because the geometric vector $\ddot{\boldsymbol{\pi}}$ characterises the geometric status of $\mathscr{T}_i$:

$$\vec{x}_p \equiv \vec{x}_p(\ddot{\boldsymbol{\pi}}); \vec{\omega}_p \equiv \vec{\omega}_p(\ddot{\boldsymbol{\pi}}) \tag{C.1}$$

However, the sensitivity s_i is defined as the derivative of intensity $I(\vec{x}, \vec{\omega}, \ddot{\boldsymbol{\pi}})$ where $\vec{x}$, $\vec{\omega}$ and $\ddot{\boldsymbol{\pi}}$ are independent. The Ph.D. thesis [32] studies the development of boundary conditions for sensitivity in detail. The boundary condition of sensitivity for a black surface is formulated in a general form:

$$s_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = -\partial_{1,\vec{\chi}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) - \partial_{2,\vec{\gamma}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{C.2}$$

where $\partial_{1,\vec{\chi}_j}$ is the operator for the spatial gradient projected to the vector $\vec{\chi}_j$, $\partial_{2,\vec{\gamma}_j}$ is the operator for the angular gradient projected to the vector $\vec{\gamma}_j$. $\vec{\chi}_j$ are the spatial velocities of deformation, and $\vec{\gamma}_j$ are the angular velocities of deformation.

The definition of $\vec{\chi}_j$, $\vec{\gamma}_j$ and the demonstration of Eq.C.2 are in the following box:

Formulating the boundary conditions of sensitivity

Following [32], we note the domain that we defined $\partial\mathcal{G}$ as the geometric space, and we define another domain as material space $\partial\mathcal{M}$. Special notations are needed to formulate the boundary condition of sensitivity on $\mathscr{T}_i$. In geometric space, the positions $\vec{y}_p \in \partial\mathcal{G}$ and directions $\vec{\omega}_p \in \mathcal{S}$ are functions of the geometric parameter $\ddot{\pi}_j$, while in material space, the positions are noted $\vec{y}_p \in \partial\mathcal{M}$ and directions are noted $\vec{\omega} \in \mathcal{S}$. They are not dependent on $\ddot{\pi}_j$.
The function $Z : \partial\mathcal{M} \times \mathbb{R} \to \partial\mathcal{G}$ and $\Omega : \partial\mathcal{M} \times \mathbb{R} \times \mathcal{S} \to \mathcal{S}$ link the two space, so that the positions and directions in one space can refer to the other space:

$$\vec{y}_p = Z(\vec{y}_p, \ddot{\pi}_j); \vec{\omega} = \Omega(\vec{y}_p, \vec{\omega}, \ddot{\pi}_j) \tag{C.3}$$

Also, we denote the intensity in geometric space as $I_i(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j)$ and in material space as $L_i(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j)$, so that we can build the following equation:

$$I_i(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j) = L_i(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j) \tag{C.4}$$

We take the derivative of Eq.C.4 with respect to $\ddot{\pi}_j$:

$$\partial_{\ddot{\pi}_j} I_i(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j) = \partial_{\ddot{\pi}_j} L_i(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j) \tag{C.5}$$

implying to the following equation, where we find the boundary condition of sensitivity $s_j(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j)$:

$$\begin{aligned}\partial_1 I_i(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j)\partial_{\ddot{\pi}_j} Z(\vec{y}_p, \ddot{\pi}_j) + \partial_2 I_i(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j)\partial_{\ddot{\pi}_j}\Omega(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j) \\ + s_j(\vec{\ddot{y}}_p, \vec{\ddot{\omega}}_p, \ddot{\pi}_j) = \partial_{\ddot{\pi}_j} L_i(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j)\end{aligned} \tag{C.6}$$

We define the derivative of Z with respect to $\ddot{\pi}_j$ as the **spatial velocity of deformation** $\vec{\chi}_j$:

$$\vec{\chi}_j \equiv \partial_{\ddot{\pi}_j} Z(\vec{y}_p, \ddot{\pi}_j) \tag{C.7}$$

For different $\ddot{\pi}_j$, the velocities of deformations $\vec{v}_j$ are summarized in Table.C.1. Furthermore, we define the derivative of Ω with respect to $\ddot{\pi}_j$ as the **angular velocity of deformation** $\vec{\gamma}_j$:

$$\vec{\gamma}_j \equiv \partial_{\ddot{\pi}_j}\Omega(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j) \tag{C.8}$$

Also, we notice that the material properties are not dependent on the geometric parameters $\ddot{\pi}_j$ in this example, which means:

$$\partial_{\ddot{\pi}_j} L_i(\vec{y}_p, \vec{\omega}_p, \ddot{\pi}_j) = 0 \tag{C.9}$$

Finally, Eq.C.2 is yielded.

j	$Z(\vec{y}, \ddot{\pi}_j)$	$\vec{\chi}_1 = \partial_{\ddot{\pi}_1} Z$
1	$\begin{bmatrix}\ddot{\pi}_1\\0\\0\end{bmatrix} + \vec{y}$	$\begin{bmatrix}1\\0\\0\end{bmatrix}$
2	$\begin{bmatrix}0\\\ddot{\pi}_2\\0\end{bmatrix} + \vec{y}$	$\begin{bmatrix}0\\1\\0\end{bmatrix}$
3	$\begin{bmatrix}0\\0\\\ddot{\pi}_3\end{bmatrix} + \vec{y}$	$\begin{bmatrix}0\\0\\1\end{bmatrix}$
4	$\vec{y} + \ddot{\pi}_4(\vec{a} \times (\vec{y} - \vec{x}_a)$	$\vec{a} \times (\vec{y} - \vec{x}_a)$

Table C.1 – The functions Z the spatial velocity of deformation corresponding to 4 geometric parameters of the object $\mathscr{T}$.

Following Eq.C.2, the boundary conditions for sensitivity consist of a term of spatial gradient and a term of the angular gradient.

The first term on the right of Eq.C.2 is a spatial gradient. It quantifies the variation of the intensity $I_i(\vec{y}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ with respect to a translation of the position $\vec{x}_p$ for an infinitely short length path in the domain $\mathcal{G}$. The direction of the translation follows the unit vector $\vec{\chi}_j$, which is the spatial velocity of deformation:

$$\partial_{1,\vec{\chi}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \lim_{\delta r \to 0} \frac{I_i(\vec{x}_p + \vec{\chi}_j \delta r, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) - I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})}{\delta \phi} \tag{C.10}$$

The second term on the right of Eq.C.2 is an angular gradient. It quantifies the variation of the intensity $I_i(\vec{y}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ with respect to a rotation of the direction $\vec{\omega}_p$ for an infinitely small angle on the unit sphere $\mathcal{S}$. The rotation of $\vec{\omega}_p$ is around the direction of the unit vector $\vec{\gamma}_i$, which is the angular velocity of deformation:

$$\partial_{2,\vec{\gamma}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \lim_{\delta \phi \to 0} \frac{I_i(\vec{x}_p, \vec{\omega}_{p\delta\phi}^{\vec{\gamma}_j}, \ddot{\boldsymbol{\pi}}) - I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})}{\delta \phi} \tag{C.11}$$

In this case, the object $\mathscr{T}$ is a Lambertian emitter. Therefore, the angular gradient is null:

$$\partial_{2,\vec{\gamma}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = 0 \tag{C.12}$$

The boundary condition of sensitivity is then only about the spatial gradient, which can be regarded as the emission term of the sensitivity (the source of sensitivity is noted $\mathring{s}_{i,j}$):

$$s_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \mathring{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = -\partial_{1,\vec{\chi}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{C.13}$$

The spatial gradient of intensity is studied in [34]. In our example, the object is triangulated. The intensity on the borders of triangles is discontinuous, and its gradient induces a Dirac term. The source of sensitivity $\mathring{s}_{i,j}$ is then composed with a surface source $\dot{s}_{i,j}$ and a line source $\hat{s}_{i,j,k}$:

$$\mathring{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \dot{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) + \sum_{k=1}^{3} \hat{s}_{i,j,k}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}) \tag{C.14}$$

For the surface source:

$$\dot{s}_{i,j}(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = -\alpha_{i,j} \mathcal{C}[I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})] \tag{C.15}$$

$\mathcal{C}$ is the collision operator, referring to Eq.4.22. $\alpha_{i,j}$ is the norm of the projected vector $\vec{\chi}_j$ on the vector $\vec{\omega}_l$, seeing Fig.3:

$$\alpha_{i,j} = \frac{\vec{\chi}_j \cdot \vec{n}_i}{\vec{\omega}_l \cdot \vec{n}_i} \tag{C.16}$$

For the line source:

$$\hat{s}_{i,j,k}(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}) = \big(I_0 - I'(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}})\big)(\vec{\omega}_l \times \vec{\chi}_j) \cdot \vec{t}_{i,k} \mathcal{H}(\vec{\omega}_l \cdot \vec{n}_{i'} < 0) \tag{C.17}$$

A position on the edge of $\mathscr{T}_i$ is noted $\vec{x}_l \in \partial \mathscr{T}_i$. Respectively, the direction of emission on edge is noted $\vec{\omega}_l \in \mathcal{S}$. $\vec{t}_{i,k}$ is the vector of circulation of the triangle, where index k

signifies the k^{th} edge of the triangle $\mathscr{T}_i$. The vectors of $\vec{t}_{i,k}$ follow the convention made in [34]: $\vec{m}_{i,k} = \vec{t}_{i,k} \times \vec{n}_i$, seeing Fig.1. $\vec{n}_{i'}$ is the normal of the triangle $\mathscr{T}_{i'}$ who shares the edge (where located the line source) with $\mathscr{T}_i$ and I' is the intensity coming backward, following the direction of $\vec{\omega}_l$ and passing through $\vec{x}_l$, seeing Fig.2.

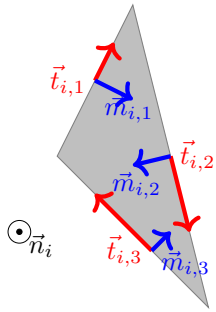
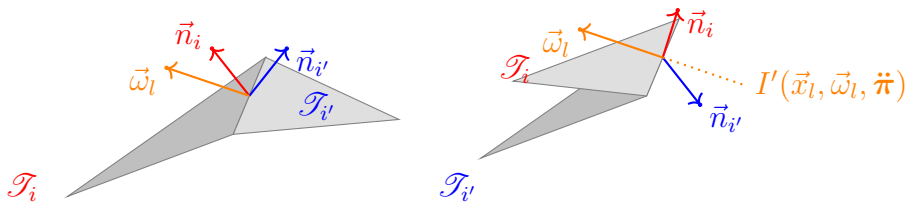


Figure 1 – convention of $\vec{t}_{i,k}$, $\vec{m}_{i,k}$ and $\vec{n}_i$ for $\vec{m}_{i,k} = \vec{t}_{i,k} \times \vec{n}_i$



(a) when $\vec{\omega}_l \cdot \vec{n}_{i'} > 0$, the line source is null in this case, because the triangles have homogeneous emitted intensity.

(b) when $\vec{\omega}_l \cdot \vec{n}_{i'} < 0$, the line source is not null. I' is then the intensity coming backward, following $\vec{\omega}_l$, passing through $\vec{x}_l$.

Figure 2 – The schema of the line source on edge shared by $\mathscr{T}_i$ and $\mathscr{T}_{i'}$.

The demonstration of Eq.C.14,C.15 and C.17 are given in the following box.

Developing the boundary condition of spatial gradient

Following Eq.C.13, the boundary condition of sensitivity is only about the spatial gradient:

$$\partial_{1,\vec{\chi}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \vec{\chi}_j \cdot \vec{\nabla} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{C.18}$$

The gradient of intensity $\vec{\nabla} I_i$ following the direction $\vec{\chi}_j$ is unknown. We will then decompose the vector $\vec{\chi}_j$ into two other vectors, where the gradient of intensity is given, seeing Fig.3:

$$\vec{\chi}_j = \alpha_{i,j}\vec{\omega}_p + \beta_{i,j}\vec{u}_p \tag{C.19}$$

$\vec{u}_p$ is the tangential vector with a norm of $\beta_{i,j}$, following which the gradient $\vec{u}_p \cdot \vec{\nabla} I_i$ is given by the boundary condition of intensity. On the direction of $\vec{\omega}_p$ with a norm of $\alpha_{i,j}$, the gradient $\vec{\omega}_p \cdot \vec{\nabla} I_i$ is given by the RTE (Eq.4.23). The following equation is then yielded:

$$\partial_{1,\vec{\chi}_j} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \alpha \mathcal{C}[I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})] + \beta \partial_{1,\vec{u}_p} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{C.20}$$

The second term on the right of Eq.C.20 is the spatial gradient of intensity following the direction of $\vec{u}_p$ (which is a tangential vector on the surface of $\mathscr{T}_i$). This term

induces a Dirac term because of the discontinuous intensity profile on the boundary of $\mathscr{T}_i$, seeing Fig.4. This Dirac term becomes a line source of sensitivity. The demonstration is following.

Line source We decompose vector $\vec{\chi}_j$ using the non-orthogonal basis on an edge of the triangle $\mathscr{T}_i$: $(\vec{\omega}_p, \vec{m}_i, \vec{t}_i)$, following the convention made in Fig.4. Since we focus on only one edge, the index of edges k is not specified. For the k^{th} edge of the i^{th} triangle:

$$\vec{\chi}_j = \alpha_{i,j}\vec{\omega}_p + \zeta_{i,j}\vec{m}_i + \psi_{i,j}\vec{t}_i \tag{C.21}$$

$$\beta_{i,j}\vec{u}_p = \zeta_{i,j}\vec{m}_i + \psi_{i,j}\vec{t}_i \tag{C.22}$$

where

$$\zeta_{i,j} = -\frac{\vec{\chi}_j \cdot (\vec{\omega}_p \times \vec{t}_i)}{\vec{\omega}_p \cdot \vec{n}_i}; \psi_{i,j} = \frac{\vec{\chi}_j \cdot (\vec{\omega}_p \times \vec{m}_i)}{\vec{\omega}_p \cdot \vec{n}_i} \tag{C.23}$$

For any location $\vec{x}_p \in \mathscr{T}_i$, we note here (y, l) the coordinates of $\vec{x}_p$ in a two dimension cartesien system of $(\vec{m}_i, \vec{t}_i)$. On the edge, the term $\beta_{i,j}\partial_{1,\vec{u}_p} I(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}})$ in Eq.C.20 induces a Dirac term of y, because $\vec{m}_i$ is orthogonal to the edge.

$$\zeta_{i,j}\partial_{1,\zeta_{i,j}} I_i(\vec{x}_p, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) = \zeta_{i,j}\delta(y)(I_{in} - I_{out}) \tag{C.24}$$

where $I_{in} = I_i$ is the intensity emitted on $\mathscr{T}_i$ and I_{out} is the intensity following the direction $\vec{\omega}_p$ out of the edge.

When $\vec{\omega}_p \cdot \vec{n}_{i'} > 0$ (Fig.5a):

$$I_{out} = I_{i'} = \lim_{\epsilon \to 0} I(\vec{x} + \epsilon\vec{m}_{i'}, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{C.25}$$

and when $\vec{\omega}_p \cdot \vec{n}_{i'} < 0$ (Fig.5b):

$$I_{out} = I'_i = \lim_{\epsilon \to 0} I(\vec{x} - \epsilon\vec{m}_i, \vec{\omega}_p, \ddot{\boldsymbol{\pi}}) \tag{C.26}$$

where $I_{i'}$ is the emitted intensity of the $\mathscr{T}_{i'}$ and I'_i is the backward incoming intensity following $\vec{\omega}_p$.

We then multiply Eq.C.24 by $(\vec{\omega}_p \cdot \vec{n}_i)$ to get a flux density of sensitivity, and we substitute Eq.C.23 into it.

When $\vec{\omega}_p \cdot \vec{n}_{i'} > 0$:

$$(\vec{\omega}_p \cdot \vec{n}_i)\zeta\delta(y)(I_i - I_{i'}) = (\vec{\omega}_p \times \vec{\chi}_j) \cdot \vec{t}_i(I_i - I_{i'})\delta(y) \tag{C.27}$$

and when $\vec{\omega}_p \cdot \vec{n}_{i'} < 0$:

$$(\vec{\omega}_p \cdot \vec{n}_i)\zeta\delta(y)(I_i - I'_i) = (\vec{\omega}_p \times \vec{\chi}_j) \cdot \vec{t}_i(I_i - I'_i)\delta(y) \tag{C.28}$$

After that, we integrate Eq.C.27 and Eq.C.28 over the surface of $\mathscr{T}_i$, with the differential surface $d\xi = dydl$.

When $\vec{\omega}_l \cdot \vec{n}_{i'} > 0$:

$$\int_{\mathscr{T}_i} (\vec{\omega}_p \times \vec{\chi}_j) \cdot \vec{t}_i(I_i - I_{i'})\delta(y)d\xi = \int_{\partial\mathscr{T}_{i,i'}} (\vec{\omega}_l \times \vec{\chi}_j) \cdot \vec{t}_i(I_i - I_{i'})dl \tag{C.29}$$

and when $\vec{\omega}_p \cdot \vec{n}_{i'} < 0$:

$$\int_{\mathscr{T}_i} (\vec{\omega}_p \times \vec{\chi}_j) \cdot \vec{t}_i (I_i - I'_i) \delta(y) d\xi = \int_{\partial \mathscr{T}_{i,i'}} (\vec{\omega}_l \times \vec{\chi}_j) \cdot \vec{t}_i (I_i - I'_i) dl \tag{C.30}$$

where $\partial \mathscr{T}_{i,i'} \equiv \partial \mathscr{T}_i \cup \partial \mathscr{T}_{i'}$, is the edge shared by $\mathscr{T}_i$ and $\mathscr{T}_{i'}$. The integral on the surface is then changed to the integral on edge. We then note the direction as $\vec{\omega}_l$ instead of $\vec{\omega}_p$.
In this example, the emitter is homogeneous: $I_i = I_{i'} = I_0$. Therefore, when $\vec{\omega}_l \cdot \vec{n}_{i'} < 0$, the line source is null and when $\vec{\omega}_p \cdot \vec{n}_{i'} < 0$, Eq.C.30 becomes:

$$\int_{\partial \mathscr{T}_{i,i'}} (\vec{\omega}_l \times \vec{\chi}_j) \cdot \vec{t}_i (I_0 - I'(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}})) dl \tag{C.31}$$

In Eq.C.31, $(\vec{\omega}_p \times \vec{\chi}_j) \cdot \vec{t}_i (I_0 - I'(\vec{x}_l, \vec{\omega}_l, \ddot{\boldsymbol{\pi}}))$ is then the line source of the boundary condition of $\mathscr{T}_i$.
Finally, the boundary condition on $\mathscr{T}_i$, Eq.C.14, C.15 and C.17 are yielded.

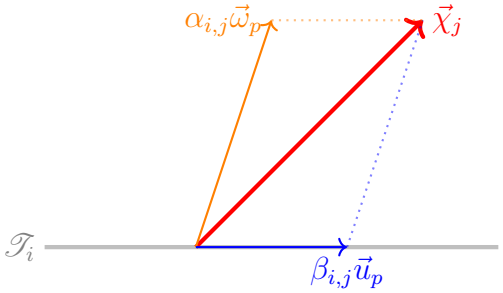


Figure 3 – Decomposition of the spatial velocity of deformation. The gradient of intensity projected to $\vec{\chi}_j$ is unknown. It is then decomposed to the direction of $\vec{\omega}_p$, and $\vec{u}_p$, where the gradients are described by RTE and the boundary condition.

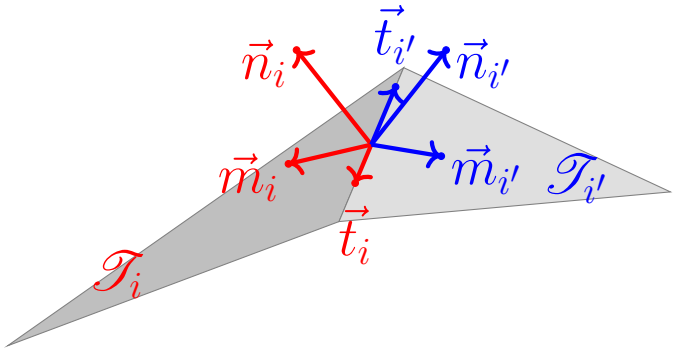
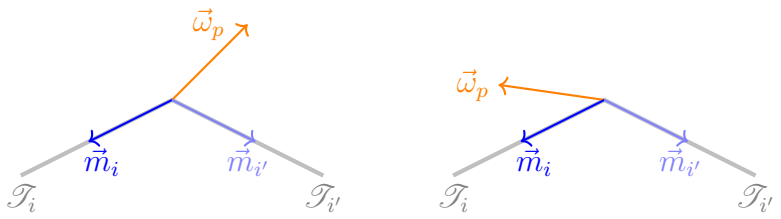


Figure 4 – The units vectors attached to $\mathscr{T}_i$ and $\mathscr{T}_{i'}$ form two orthonormal basis: $(\vec{m}_i, \vec{t}_i, \vec{n}_i)$ and $(\vec{m}_{i'}, \vec{t}_{i'}, \vec{n}_{i'})$.



(a) $\vec{\omega}_p \cdot \vec{n}_{i'} > 0$ **(b)** $\vec{\omega}_p \cdot \vec{n}_{i'} < 0$

Figure 5 – Schema of I_{out}. I_{out} is the intensity following $\vec{\omega}$ out of the edge of $\mathscr{T}_i$. When $\vec{\omega}_p \cdot \vec{n}_{i'} > 0$, I_{out} will take the value of the intensity emitted on $\mathscr{T}_{i'}$: $I_{i'}$. When $\vec{\omega}_p \cdot \vec{n}_{i'} < 0$, I_{out} will take the value of intensity coming backward, following $\vec{\omega}_p$: I'_i.

List of Figures

List of Tables

Acronyms

BRDF *Bidirectional Reflectance Distribution Function* 28, 84, 203

cdf *cumulative distribution function* 15

CSP *Concentrated Solar Power* . . . . iii, 1–4, 39–46, 48, 49, 80, 107, 159, 197–199, 230, 247, 252, 254, 256, 258–260

IFMC *Integral Formulation Monte-Carlo* 41–43

MCRT *Monte-Carlo Ray Tracing* 40, 41, 43, 230

PBR *Physically Based Rendering* 46

pdf *probability density function* 8–17, 19, 20, 30, 31, 33, 45, 55–58, 62, 64, 67, 84, 87, 89, 98, 100, 102, 103, 170, 171, 173, 175, 178, 216, 217, 229, 254

RTE *Radiative Transfer Equation* 23, 27, 28, 34, 41, 43, 46, 47, 52, 53, 70, 71, 75, 82, 83, 86, 93–95, 107, 162, 165, 203, 205, 208, 213, 214, 224, 226, 232–234, 254–256

STP *Solar Tower Power* 1, 3, 5, 40, 43, 48, 79, 111, 197, 198, 230, 252, 253, 256–259

Bibliography

[1] Seungryul Baek, Kwang In Kim, and Tae-Kyun Kim. *Pushing the Envelope for RGB-based Dense 3D Hand Pose Estimation via Neural Rendering*. 2019 (cit. on pp. 4, 44, 254).

[2] Richard Bellman and George Milton Wing. *An introduction to invariant imbedding*. SIAM, 1992 (cit. on pp. 51, 54).

[3] Manuel J Blanco, Amaia Mutuberria, Pierre Garcia, Raquel Gastesi, and Victor Martin. "Preliminary validation of Tonatiuh." In: *SolarPACES conference*. 2009 (cit. on pp. 40, 41).

[4] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J. Black. *Keep it SMPL: Automatic Estimation of 3D Human Pose and Shape from a Single Image*. 2016 (cit. on pp. 4, 44, 254).

[5] Cyril Caliot, Hadrien Benoit, Emmanuel Guillot, Jean-Louis Sans, Alain Ferriere, Gilles Flamant, Christophe Coustet, and Benjamin Piaud. "Validation of a Monte Carlo Integral Formulation Applied to Solar Facility Simulations and Use of Sensitivities." In: *Journal of Solar Energy Engineering* 137.2 (Apr. 2015), 8 p. (Cit. on pp. 1, 40, 43, 178, 252).

[6] Cyril Caliot, Stéphane Blanco, Christophe Coustet, Mouna El-Hafi, Vincent Eymet, Vincent Forest, Richard Fournier, and Benjamin Piaud. "Combined conductive-radiative heat transfer analysis in complex geometry using the Monte Carlo method." In: *CTRPM-VI-6th Computational Thermal Radiation in Participating Media VI*. 2018, 8–p (cit. on pp. 1, 252).

[7] Carlo Cercignani. "The boltzmann equation." In: *The Boltzmann equation and its applications*. Springer, 1988, pp. 40–103 (cit. on p. 24).

[8] Jérémi Dauchet. "Analyse radiative des photobioréacteurs." Theses. Université Blaise Pascal - Clermont-Ferrand II, Dec. 2012 (cit. on p. 24).

[9] Jérémie De la Torre. "Calculs de sensibilités par méthode de Monte-Carlo, pour la conception de procédés à énergie solaire concentrée." PhD thesis. 2011 (cit. on pp. 2, 3, 42, 46, 61, 76, 80, 252, 253, 255).

[10] A De Lataillade, Stéphane Blanco, Y Clergent, Jean-Louis Dufresne, Mouna El Hafi, and Richard Fournier. "Monte Carlo method and sensitivity estimations." In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 75.5 (2002), pp. 529–538 (cit. on pp. 3, 20–22, 45, 252).

[11] Jérémie Delatorre, Germain Baud, Jean-Jacques Bézian, Stéphane Blanco, Cyril Caliot, Jean-François Cornet, Christophe Coustet, Jérémi Dauchet, Mouna El Hafi, Vincent Eymet, et al. "Monte Carlo advances and concentrated solar applications." In: *Solar Energy* 103 (2014), pp. 653–681 (cit. on pp. 41, 42, 178).

[12] Theodore A Dellin and Miriam J Fish. *User's manual for DELSOL: a computer code for calculating the optical performance, field layout, and optimal system design for solar central receiver plants*. Tech. rep. Sandia National Lab.(SNL-CA), Livermore, CA (United States), 1979 (cit. on p. 40).

[13] Miguel Diago, Nicolas Calvet, and Peter R Armstrong. "Net power maximization from a faceted beam-down solar concentrator." In: *Solar Energy* 204 (2020), pp. 476–488 (cit. on p. 43).

[14] William L Dunn and J Kenneth Shultis. *Exploring monte carlo methods*. Elsevier, 2011 (cit. on pp. 10, 13, 14, 21, 41, 42, 106, 187, 191, 258).

[15] Olivier Farges. "Conception optimale de centrales solaires à concentration: application aux centrales à tour et aux installations" beam down"." PhD thesis. Ecole des Mines d'Albi-Carmaux, 2014 (cit. on pp. 1, 2, 252).

[16] Olivier Farges, Jean-Jacques Bézian, Hélène Bru, Mouna El Hafi, Richard Fournier, and Christophe Spiesser. "Life-time integration using Monte Carlo Methods when optimizing the design of concentrated solar power plants." In: *Solar Energy* 113 (2015), pp. 57–62 (cit. on pp. 42–44, 178).

[17] Y Favennec, F Dubot, D Le Hardy, B Rousseau, and DR Rousse. "Space-dependent sobolev gradients as a regularization for inverse radiative transfer problems." In: *Mathematical Problems in Engineering* 2016 (2016) (cit. on pp. 43, 44).

[18] Mathieu Galtier. "Approche statistique du rayonnement dans les milieux gazeux hétérogènes : de l'échantillonnage des transitions moléculaires au calcul de grandeurs radiatives." Theses. Ecole des Mines d'Albi-Carmaux, Dec. 2014 (cit. on p. 25).

[19] Pierre Garcia, Alain Ferriere, and Jean-Jacques Bezian. "Codes for solar flux calculation dedicated to central receiver system applications: A comparative review." In: *Solar Energy* 82.3 (2008), pp. 189–197 (cit. on pp. 40, 41).

[20] Kyle Genova, Forrester Cole, Aaron Maschinot, Aaron Sarna, Daniel Vlasic, and William T. Freeman. "Unsupervised Training for 3D Morphable Model Regression." In: (2018) (cit. on pp. 4, 44, 254).

[21] Emmanuel Gobet. "Stochastic differential equations and Feynman-Kac formulas." In: *Monte-Carlo Methods and Stochastic Processes*. Num Pages: 46. Chapman and Hall/CRC, 2016 (cit. on pp. 20, 21, 106, 187).

[22] Zili He, Paule Lapeyre, Stephane Blanco, Simon Eibner, Mouna El Hafi, and Richard Fournier. "Monte-Carlo estimation of geometric sensitivities in Solar Power Tower systems of flat mirrors." In: *Solar Energy* 253 (2023), pp. 9–29 (cit. on p. 111).

[23] John R Howell, Kyle Mengd Daun, and Robert Siegel. *Thermal radiation heat transfer*. CRC press, 2020 (cit. on p. 188).

[24] Loris Ibarrart, Cyril Caliot, Mouna El-Hafi, Richard A Fournier, Stéphane Blanco, Sébastien Dutour, Jeremi Dauchet, Jean-Marc Tregan, Vincent Eymet, and Forest Vincent. "Combined conductive-convective-radiative heat transfer in complex geometry using the Monte Carlo method." In: *IHTC 16 - 16th International Heat Transfer Conference*. The heat and mass transfer society of china and qsinghua university. Beijing, China, Aug. 2018, 8 p. (Cit. on pp. 1, 252).

[25] Frederick James. "A review of pseudorandom number generators." In: *Computer physics communications* 60.3 (1990), pp. 329–344 (cit. on p. 14).

[26] Hiroharu Kato, Deniz Beker, Mihai Morariu, Takahiro Ando, Toru Matsuoka, Wadim Kehl, and Adrien Gaidon. *Differentiable Rendering: A Survey*. 2020 (cit. on pp. 4, 44, 46, 82, 90, 198, 215, 254, 259).

[27] Hiroharu Kato and Tatsuya Harada. *Learning View Priors for Single-view 3D Reconstruction*. 2018 (cit. on pp. 4, 44, 254).

[28] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. “Neural 3D Mesh Renderer.” In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2018 (cit. on pp. 4, 44, 254).

[29] Milton Kerker. *The scattering of light and other electromagnetic radiation: physical chemistry: a series of monographs*. Vol. 16. Academic press, 2013 (cit. on p. 25).

[30] Mirosław Lachowicz. “Microscopic, mesoscopic and macroscopic descriptions of complex systems.” English. In: *Probabilistic Engineering Mechanics* 26.1 (2011), pp. 54–60 (cit. on p. 24).

[31] P Lapeyre, Stéphane Blanco, Cyril Caliot, J Dauchet, Mouna El Hafi, Richard Fournier, Olivier Farges, Jacques Gautrais, and Maxime Roger. “Monte-Carlo and sensitivity transport models for domain deformation.” In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 251 (2020), p. 107022 (cit. on pp. 2, 47, 198, 203, 252, 259).

[32] Paule Lapeyre. “Un modèle de transfert radiatif pour la sensibilité géométrique: lecture physique des algorithmiques de Monte-Carlo via la double randomisation.” PhD thesis. Perpignan, 2021 (cit. on pp. 5, 47, 48, 51, 70–72, 80, 95, 107, 198, 203, 206, 221, 255, 256, 259).

[33] Paule Lapeyre, Stéphane Blanco, Cyril Caliot, Jérémi Dauchet, Mouna El Hafi, Richard Fournier, Olivier Farges, and Maxime Roger. “Monte-Carlo and Domain-Deformation Sensitivities.” In: *Proceedings of the 9th International Symposium on Radiative Transfer, RAD-19*. Begel House Inc. 2019 (cit. on pp. 51, 70, 72, 80, 95, 107, 198, 203, 206, 256, 259).

[34] Paule Lapeyre, Zili He, Stéphane Blanco, Cyril Caliot, Christophe Coustet, Jérémi Dauchet, Mouna El Hafi, Simon Eibner, Eugene d’Eon, Olivier Farges, Richard Fournier, Jacques Gautrais, Nada Chems Mourtaday, and Maxime Roger. *A physical model and a Monte Carlo estimate for the specific intensity spatial derivative, angular derivative and geometric sensitivity*. 2022 (cit. on pp. 45, 47, 48, 72, 80, 81, 95, 107, 161, 164–166, 198, 203, 206–209, 223, 224, 256, 259).

[35] PL Leary and JD Hankins. *User’s guide for MIRVAL: a computer code for comparing designs of heliostat-receiver optics for central receiver solar power plants*. Tech. rep. Sandia National Lab.(SNL-CA), Livermore, CA (United States), 1979 (cit. on pp. 40, 41).

[36] Gaël Levêque, Roman Bader, Wojciech Lipiński, and Sophia Haussener. “High-flux optical systems for solar thermochemistry.” In: *Solar Energy* 156 (2017), pp. 133–148 (cit. on p. 41).

[37] Lifeng Li, Joe Coventry, Roman Bader, John Pye, and Wojciech Lipiński. “Optics of solar central receiver systems: a review.” In: *Opt. Express* 24.14 (July 2016), A985–A1007 (cit. on p. 41).

[38] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. “Differentiable monte carlo ray tracing through edge sampling.” In: *ACM Transactions on Graphics (TOG)* 37.6 (2018), pp. 1–11 (cit. on pp. 46, 51).

[39] FW Lipps and Lorin L Vant-Hull. “A cellwise method for the optimization of large central receiver systems.” In: *Solar Energy* 20.6 (1978), pp. 505–516 (cit. on p. 40).

[40] AJ Marston, KJ Daun, and MR Collins. “Geometric optimization of concentrating solar collectors using monte carlo simulation.” In: *Journal of solar energy engineering* 132.4 (2010) (cit. on p. 43).

[41] M Mehos, C Turchi, J Vidal, M Wagner, Z Ma, C Ho, W Kolb, C Andraka, and A Kruizenga. “Concentrating solar power Gen3 demonstration roadmap. 2017.” In: *NREL (National Renewable Energy Laboratory (NREL), Golden, CO (United States))* () (cit. on p. 2).

[42] Nicholas Metropolis and Stanislaw Ulam. “The monte carlo method.” In: *Journal of the American statistical association* 44.247 (1949), pp. 335–341 (cit. on p. 10).

[43] David Mills. “Advances in solar thermal electricity technology.” In: *Solar energy* 76.1-3 (2004), pp. 19–31 (cit. on p. 40).

[44] Merlin Nimier-David, Sébastien Speierer, Benoı
t Ruiz, and Wenzel Jakob. “Radiative backpropagation: an adjoint method for lightning-fast differentiable rendering.” In: *ACM Transactions on Graphics (TOG)* 39.4 (2020), pp. 146–1 (cit. on pp. 76, 255).

[45] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. *Learning to Estimate 3D Human Pose and Shape from a Single Color Image*. 2018 (cit. on pp. 4, 44, 254).

[46] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016 (cit. on pp. 3, 4, 11, 46, 66, 80, 253).

[47] Benjamin Piaud. “Modélisation mésoscopique des écoulements avec changement de phase à partir de l’équation de Boltzmann-Enskog. Introduction des effets thermiques.” Theses. Université Toulouse III, July 2007 (cit. on p. 24).

[48] Daniel F Potter, Jin-Soo Kim, Alex Khassapov, Ric Pascual, Lachlan Hetherton, and Zikai Zhang. “Heliosim: An integrated model for the optimisation and simulation of central receiver CSP facilities.” In: *AIP Conference Proceedings*. Vol. 2033. 1. AIP Publishing LLC. 2018, p. 210011 (cit. on p. 40).

[49] Maria Simona Răboacă, Gheorghe Badea, Adrian Enache, Constantin Filote, Gabriel Răsoi, Mihai Rata, Alexandru Lavric, and Raluca-Andreea Felseghi. “Concentrating solar power technologies.” In: *Energies* 12.6 (2019), p. 1048 (cit. on p. 40).

[50] Maxime Roger. “Modèles de sensibilité dans le cadre de la méthode de Monte-Carlo: illustrations en transfert radiatif.” PhD thesis. 2006 (cit. on pp. 3, 51, 61–63, 252).

[51] Maxime Roger, Stéphane Blanco, Mouna El Hafi, and Richard Fournier. “Monte Carlo estimates of domain-deformation sensitivities.” In: *Physical review letters* 95.18 (2005), p. 180601 (cit. on pp. 3, 5, 20, 21, 45, 51, 61, 63, 252, 255).

[52] Morgan Sans, Mouna El Hafi, V Eymet, V Forest, Richard Fournier, and N Villefranque. “Null-collision meshless Monte Carlo-A new reverse Monte Carlo algorithm designed for laser-source emission in absorbing/scattering inhomogeneous media.” In: *Journal of Quantitative Spectroscopy and Radiative Transfer* 271 (2021), p. 107725 (cit. on pp. 178, 180).

[53] Peter Schwarzbözl, Robert Pitz-Paal, and Mark Schmitz. “Visual HFLCAL-A software tool for layout and optimisation of heliostat fields.” In: *Proceedings*. 2009 (cit. on pp. 40, 41).

[54] Ping Sheng. *Introduction to wave scattering, localization and mesoscopic phenomena*. Vol. 88. Springer Science & Business Media, 2006 (cit. on p. 25).

[55] Jean-Marc Tregan. “Thermique non-linéaire et Monte-Carlo.” Theses. Université Toulouse 3 Paul Sabatier, Dec. 2020 (cit. on p. 13).

[56] Shubham Tulsiani, Tinghui Zhou, Alexei A. Efros, and Jitendra Malik. *Multi-view Supervision for Single-view Reconstruction via Differentiable Ray Consistency*. 2017 (cit. on pp. 4, 44, 254).

[57] Najda Villefranque, Richard Fournier, Fleur Couvreux, Stéphane Blanco, Céline Cornet, Vincent Eymet, Vincent Forest, and Jean-Marc Tregan. “A Path-Tracing Monte Carlo Library for 3-D Radiative Transfer in Highly Resolved Cloudy Atmospheres.” In: *Journal of Advances in Modeling Earth Systems* 11.8 (2019), pp. 2449–2473. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2018MS001602 (cit. on pp. 1, 178, 180, 252).

[58] Najda Villefranque, Frédéric Hourdin, Louis d'Alençon, Stéphane Blanco, Olivier Boucher, Cyril Caliot, Christophe Coustet, Jérémi Dauchet, Mouna El Hafi, Vincent Eymet, Olivier Farges, Vincent Forest, Richard Fournier, Jacques Gautrais, Valéry Masson, Benjamin Piaud, and Robert Schoetter. "The teapot in a city: A paradigm shift in urban climate modeling." In: *Science Advances* 8.27 (2022), eabp8934. eprint: https://www.science.org/doi/pdf/10.1126/sciadv.abp8934 (cit. on pp. 1, 2, 252).

[59] CN Vittitoe and F Biggs. *User's guide to HELIOS: a computer program for modeling the optical behavior of reflecting solar concentrators. Part I. Introduction and code input.* Tech. rep. Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 1981 (cit. on p. 40).

[60] Michael J Wagner and Tim Wendelin. "SolarPILOT: A power tower solar field layout and characterization tool." In: *Solar Energy* 171 (2018), pp. 185–196 (cit. on pp. 40, 41).

[61] Ye Wang, Charles-Alexis Asselineau, Joe Coventry, and John Pye. "Optical Performance of Bladed Receivers for CSP Systems." In: June 2016 (cit. on p. 41).

[62] Ye Wang, Daniel Potter, Charles-Alexis Asselineau, Clotilde Corsi, Michael Wagner, Cyril Caliot, Benjamin Piaud, Manuel Blanco, Jin-Soo Kim, and John Pye. "Verification of optical modelling of sunshape and surface slope error for concentrating solar power systems." In: *Solar Energy* 195 (2020), pp. 461–474 (cit. on pp. 1, 40–42, 252).

[63] Tim Wendelin, Aron Dobos, and Allan Lewandowski. "SolTrace: a ray-tracing code for complex solar optical systems." In: (2013) (cit. on pp. 40, 41).

[64] Kai Yan, Christoph Lassner, Brian Budge, Zhao Dong, and Shuang Zhao. "Efficient Estimation of Boundary Integrals for Path-Space Differentiable Rendering." In: *ACM Trans. Graph.* 41.4 (July 2022) (cit. on pp. 198, 259).

[65] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. *Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction without 3D Supervision*. 2016 (cit. on pp. 4, 44, 254).

[66] Seonghyeok Yang, Kyungeun Lee, and Ikjin Lee. "Pattern-free heliostat field layout optimization using physics-based gradient." In: *Solar Energy* 206 (2020), pp. 722–731 (cit. on p. 43).

[67] Qiang Yu, Zhifeng Wang, and Ershu Xu. "Analysis and improvement of solar flux distribution inside a cavity receiver based on multi-focal points of heliostat field." In: *Applied energy* 136 (2014), pp. 417–430 (cit. on p. 44).

[68] Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. "Monte Carlo estimators for differential light transport." In: *ACM Transactions on Graphics (TOG)* 40.4 (2021), pp. 1–16 (cit. on pp. 46, 51).

[69] Cheng Zhang, Zhao Dong, Michael Doggett, and Shuang Zhao. "Antithetic Sampling for Monte Carlo Differentiable Rendering." In: *ACM Trans. Graph.* 40.4 (July 2021) (cit. on p. 160).

[70] Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. "Path-Space Differentiable Rendering." In: *ACM Trans. Graph.* 39.4 (July 2020) (cit. on pp. 4, 5, 51, 254, 255).

[71] Cheng Zhang, Lifan Wu, Changxi Zheng, Ioannis Gkioulekas, Ravi Ramamoorthi, and Shuang Zhao. "A differential theory of radiative transfer." In: *ACM Transactions on Graphics (TOG)* 38.6 (2019), pp. 1–16 (cit. on pp. 45, 46, 51).

[72] Xiong Zhang, Qiang Li, Hong Mo, Wenbo Zhang, and Wen Zheng. *End-to-end Hand Mesh Recovery from a Monocular RGB Image*. 2019 (cit. on pp. 4, 44, 254).

[73] Yiyi Zhou and Yuhong Zhao. “Heliostat field layout design for solar tower power plant based on GPU.” In: *IFAC Proceedings Volumes* 47.3 (2014), pp. 4953–4958 (cit. on p. 43).

Contents

Résumé long en français

Contexte Le réchauffement climatique et la crise climatique sont des défis majeurs de notre époque, qui menacent la santé de notre planète et de ses habitants. Les activités humaines, en particulier les émissions de gaz à effet de serre résultant de la combustion de combustibles fossiles, sont la principale cause du réchauffement climatique et de l'augmentation des températures moyennes à travers le monde.

Pour lutter contre cette crise, une transition énergétique mondiale est nécessaire, visant à remplacer les sources d'énergie fossile par des sources d'énergie renouvelable. Les sources d'énergie renouvelable, telles que l'énergie solaire, l'énergie éolienne et l'énergie hydroélectrique, sont des solutions durables qui peuvent réduire les émissions de gaz à effet de serre et atténuer l'impact du réchauffement climatique.

La transition numérique et énergétique est un défi majeur de notre époque, visant à transformer notre société en une économie plus durable, résiliente et économe en ressources. Cette transition implique une utilisation accrue des technologies numériques pour améliorer l'efficacité énergétique, augmenter la part des énergies renouvelables et réduire les émissions de gaz à effet de serre.

Dans le cadre de cette transition, les simulations numériques sont devenues un outil essentiel pour optimiser la performance des systèmes énergétiques. En particulier, la méthode de Monte-Carlo est largement utilisée pour simuler des systèmes énergétiques complexes, en raison de sa capacité à générer des résultats fiables même dans des conditions d'incertitude importantes.

Tout d'abord, permettez-moi de vous présenter la méthode de Monte-Carlo. Il s'agit d'une technique de simulation numérique utilisée pour résoudre des problèmes complexes en générant des échantillons aléatoires. Cette méthode est couramment utilisée dans les domaines de la physique, des mathématiques, de l'informatique et de l'ingénierie, entre autres. L'idée centrale de la méthode de Monte-Carlo est d'estimer une quantité inconnue en générant de nombreux échantillons aléatoires et en calculant une moyenne. Cette méthode est particulièrement utile pour les problèmes qui ne peuvent pas être résolus analytiquement ou qui nécessitent des calculs complexes.

Parmi les sources d'énergie renouvelable, la CSP est une technologie prometteuse qui peut offrir une production d'électricité constante et prévisible, contrairement à d'autres sources d'énergie renouvelable telles que l'énergie solaire photovoltaïque ou l'énergie éolienne. Cependant, la performance de la CSP dépend de nombreux facteurs, tels que les conditions météorologiques, la géométrie du système et les propriétés du matériau utilisé. Par conséquent, il est crucial de simuler la performance de ces systèmes de manière précise pour maximiser leur efficacité.

La synthèse d'images est un autre outil numérique important qui peut être utilisé pour visualiser les résultats de simulations numériques. En utilisant des techniques de modélisation, d'éclairage et de rendu, la synthèse d'images peut créer des images numériques réalistes et convaincantes qui peuvent aider les chercheurs à comprendre les performances des systèmes énergétiques et à communiquer leurs résultats de manière efficace.

Résumé Les technologies d'énergie solaire à concentration (CSP) ont un potentiel indéniablement important pour produire de l'électricité et de la chaleur. Ce doctorat. La thèse est motivée par des problèmes liés à l'optimisation et aux problèmes inverses, où les sensibilités de différents paramètres dans un système CSP sont nécessaires. Ces paramètres pourraient être, par exemple, les positions et les propriétés de réflexion des héliostats. Dans une étude récente [5], Caliot *et al.* calcule inversement la propriété réfléchissante d'un héliostat à partir d'une carte de flux mesuré sur le récepteur d'une installation CSP. Pour y parvenir, la sensibilité de la propriété réfléchissante au flux était nécessaire. De plus, des sensibilités aux paramètres de conception (tels que les positions, les rotations et les tailles des héliostats) sont nécessaires pour optimiser les performances optiques d'un système STP. Cependant, à la connaissance de l'auteur, aucune méthode n'est disponible dans la communauté CSP pour estimer les sensibilités impliquant des paramètres géométriques. Cette thèse vise donc à combler cette lacune.

Avant de se concentrer sur les sensibilités, l'évaluation des performances optiques dans les systèmes CSP est principalement basée sur la méthode de Monte Carlo [62]. Cet outil numérique spécifique a l'avantage de traiter la complexité de la géométrie étudiée, le couplage de différents phénomènes physiques, ou la grande échelle de temps [6, 15, 24, 57]. La complexité de la géométrie étudiée fait que les échelles géométriques des différents éléments d'un système radiatif sont très différentes. Par exemple, des études ont été faites sur l'analyse thermique d'une théière dans une ville. Les échelles géométriques des éléments sont très différentes, d'un grand bâtiment moderne à un arbre près de la route et une petite fenêtre sur le mur d'un garage dans une ville. Villefranque *et al.* a étudié l'état thermique de cette théière [58]. Le rayonnement thermique entre tous les éléments, même la diffusion thermique dans un mur isolé thermiquement, et la transmission des fenêtres à double vitrage doivent être pris en compte. Une telle méthode de Monte Carlo rend possible la modélisation du transfert thermique avec une complexité géométrique. La géométrie peut également être très complexe dans un système CSP. Par exemple, la centrale solaire d'Ivanpah est la plus grande centrale solaire au monde en 2022 et est composée de 173 500 héliostats. Chaque héliostat suit la position du soleil pendant la journée et peut ombrager d'autres héliostats. La méthode de Monte Carlo est bien adaptée pour traiter cette complexité géométrique.

Les travaux de cette thèse sont développés au sein d'un groupe de recherche (RAPSODEE[1], LAPLACE[2], Mesostar[3] et le groupe EDstar [4]) où les principales activités portent sur la simulation numérique du transfert radiatif par la méthode de Monte Carlo. L'estimation de la sensibilité par la méthode Monte Carlo dans les systèmes CSP reste un problème de recherche ouvert, et très peu d'études ont été réalisées [9, 15, 31]. Les médecins du groupe travaillent depuis 2002 sur l'estimation des sensibilités par la méthode de Monte Carlo. De Lataillade a étudié la sensibilité du rapport de chauffe à la concentration en espèces dans une flamme CH_4 [10]. Cependant, l'estimation des sensibilités impliquant des paramètres géométriques devait encore être possible. En 2006, Roger *et al.* a développé une méthode de Monte Carlo pour estimer les sensibilités Villefranque2022impliquant des paramètres géométriques [50, 51]. Cette méthode est notée comme la " méthode de dérivation observable " dans cette thèse. La signification physique de ce nom sera discutée au chapitre.2. Après cela, De la Torre a tenté d'appliquer la méthode de dérivation observable pour estimer les

[1]Recherche d'Albi en génie des Procédés, des SOlides Divisés, de l'Énergie et de l'Environnement
[2]Le Laboratoire Plasma et Conversion d 'Energie
[3]https://www.meso-star.com/
[4]http://www.edstar.cnrs.fr/prod/fr/

sensibilités géométriques dans les systèmes CSP dans sa thèse en 2011 [9]. Malheureusement, l'auteur n'a pas tenu compte des effets de blocage et d'ombrage. C'est que l'application de la méthode de dérivation observable pour les cas impliquant une géométrie complexe demande un gros effort de développement formel. La géométrie d'un système CSP avec héliostats est trop complexe pour prendre en compte tous les blocages et ombrages, ce qui limite l'application de cette méthode dans le système CSP. Jusqu'en 2021, un modèle général de transport de la sensibilité géométrique était proposé par Lapeyre *et al.*, également dédié à l'estimation des sensibilités dans les systèmes radiatifs. Cette méthode est notée "méthode du modèle de sensibilité" dans cette thèse. La signification physique du nom sera discutée au chapitre.2. Contrairement à la méthode de dérivation observable, un développement moins formel est nécessaire pour que la méthode du modèle de sensibilité soit appliquée dans les cas impliquant des géométries complexes. Cette méthode a le potentiel d'être appliquée aux applications CSP.

Cependant, la même méthode a une faiblesse par rapport à celle de la dérivation observable. Pour la méthode de dérivation d'observables, lorsqu'une observable est évaluée, sa sensibilité à tout paramètre est également évaluée au prix d'un faible effort de calcul supplémentaire. Cet avantage est dû au fait que la méthode de Monte Carlo estime un vecteur contenant l'observable et toutes ses sensibilités à différents paramètres. Par conséquent, l'estimation de l'observable et toutes ses sensibilités sont complètement "vectorisées". Cette méthode sera détaillée au chapitre.3. Pour la méthode du modèle de sensibilité, on perd cet avantage de vectorisation complète. Les utilisateurs doivent construire un modèle pour chaque sensibilité correspondant à chaque paramètre. Par conséquent, au moins une estimation de Monte Carlo est nécessaire pour chaque modèle. Néanmoins, lors de l'application de la méthode dans des cas impliquant des géométries complexes, les sources d'un modèle de sensibilité peuvent se trouver sur la surface ou sur les segments, ce qui rend le modèle difficile à résoudre par la méthode de Monte Carlo. Contrairement à la méthode de dérivation observable, le modèle de sensibilité effectue des estimations de Monte Carlo pour les scalaires. Au moins une estimation de Monte Carlo est nécessaire pour chaque source pour résoudre un tel modèle de sensibilité. Cet inconvénient rend le développement du codage plus lourd et le temps de calcul nécessaire très consommateur. Ici vient un problème intéressant à résoudre, qui est de retrouver l'avantage de la vectorisation pour la méthode du modèle de sensibilité. L'avantage de la vectorisation complète est difficile à retrouver en suivant la méthode du modèle de sensibilité, mais une stratégie de vectorisation partielle est fournie au Chapitre.4. Suivant cette stratégie, une méthode d'estimation des sensibilités pour un système STP est développée au chapitre.5. Enfin, dans le chapitre.6, nous tentons d'obtenir une vectorisation complète en suivant la méthode du modèle de sensibilité en transformant toutes les sources linéaires du système en sources surfaciques. Quelques cas académiques sont développés, et certaines limites seront discutées.

Enfin, des efforts sont également faits pour comprendre et comparer les recherches dans la communauté de la synthèse d'images pour des problèmes de transfert radiatif similaires sur les sensibilités. Le rendu d'images de haute qualité est l'un des principaux défis de la synthèse d'images. Il s'agit de produire une image 2D à partir d'une description d'une scène 3D. Il existe de nombreuses façons de rendre des images, mais le rendu physique est l'un des plus réalistes. C'est parce que le rendu basé sur la physique suit les principes de la physique pour modéliser l'interaction de la lumière et de la matière (par lancer de rayons) pour simuler la réalité [46]. Le lancer de rayons est un algorithme simple basé sur le suivi du chemin d'un rayon de lumière à travers une scène lorsqu'il interagit avec et rebondit sur des objets dans un environnement. De nos jours, presque toutes les images de synthèse dans l'industrie cinématographique sont réalisées par la méthode de Monte Carlo et les algorithmes de lancer de rayons. Grâce aux outils numériques et librairies [46] développées dans le domaine de la synthèse d'images, l'implémentation de la méthode de Monte Carlo dans cette thèse devient possible. Par ailleurs, les chercheurs en synthèse d'images travaillent également sur l'estimation des sensibilités. La méthode correspondante est développée et

étudiée au Chapitre.2 et au Chapitre.3 et est appelée rendu différentiable [70]. Cette méthode a déjà été utilisée dans de nombreuses applications impliquant des géométries complexes. Couplées aux algorithmes d'optimisation et réseau neutre [26], les sensibilités sont utilisées pour la reconstruction d'objets 3D [27, 28, 56, 65], les morphologies [4, 45], formes de main [1, 72], formes de visage [20], etc.

Cependant, bien que nous partagions des problèmes de recherche similaires dans l'estimation des sensibilités, les points de vue et les vocabulaires entre les chercheurs en synthèse d'images et notre groupe présentent des différences significatives. La méthode de rendu différentiel se concentre principalement sur la solution mathématique et les algorithmes. Moins d'attention a été accordée à l'interprétation physique des sensibilités. Cette méthode est comparée aux deux autres méthodes mentionnées dans le dernier paragraphe du chapitre.3, où nous expliquons, analysons et comparons les trois approches. On constate que l'analyse de sensibilité est difficile à réaliser en suivant la méthode de rendu différentiable. Cependant, l'analyse de sensibilité est effectuée en suivant la méthode du modèle de sensibilité au chapitre.5 pour une application CSP. Nous verrons dans cette thèse que les deux communautés ont de nombreux liens dans les problématiques de recherche de sensibilité et peuvent bénéficier des recherches de l'autre.

Enfin, cette thèse est organisée en six chapitres :

- Chapitre.1 vise à préparer une base théorique pour les lecteurs non familiarisés avec la physique des rayonnements ou la méthode de Monte Carlo. Nous passerons en revue les approches intégrales et statistiques du rayonnement et introduirons la méthode de Monte Carlo.

  Après un rapide examen des statistiques de base, nous avons introduit la méthode de Monte Carlo, qui est utilisée pour estimer une intégrale sous une forme de valeur attendue. Le résultat est toujours donné avec un intervalle de confiance qui repose sur la variance. Lorsque la variance est trop élevée, des techniques de réduction de la variance telles que *échantillonnage d'importance* peuvent être appliquées. Lorsque plusieurs valeurs attendues doivent être estimées simultanément, la technique de *vectorisation* peut parfois être mise en œuvre pour réutiliser l'ensemble des nombres aléatoires. Cette méthode peut être appliquée lorsque les valeurs attendues partagent le même pdf. En réduisant la génération de nombres aléatoires, le temps de calcul est grandement amélioré.

  Les différences finies peuvent naïvement atteindre des sensibilités ou des dérivées approximatives. Elle repose sur l'estimation de deux valeurs attendues avec une variation infinitésimale d'un paramètre choisi. Cette approche inefficace entraîne une estimation biaisée de la sensibilité avec une variance très élevée. Nous avons montré que la différenciation directe de la valeur attendue est beaucoup plus appropriée puisque l'estimation est sans biais et donne une barre d'erreur acceptable.

  Dans la deuxième partie de ce chapitre, la physique du transfert radiatif est introduite d'un point de vue corpusculaire et mésoscopique. La physique du transfert radiatif dans le milieu est décrite par RTE, en utilisant l'intensité comme descripteur. La condition aux limites régit le transport de l'intensité sur les frontières. Afin de construire un algorithme de Monte Carlo pour estimer une observable physique, l'observable doit être reformulée comme une intégrale sous une forme de valeur attendue.

  Un exemple de résolution de problème de transfert radiatif par la méthode de Monte Carlo est donné à la fin. De même que la réciprocité des rayons, l'algorithme de Monte Carlo peut être construit différemment. Il peut être construit de l'émetteur vers le récepteur ou, à l'inverse, du récepteur vers l'émetteur. Ces deux stratégies font référence aux algorithmes de lancer de rayons direct et inverse. Enfin, nous relions le vocabulaire des formulations intégrales, de la physique et des algorithmes pour préparer d'autres études sur la *vectorisation* dans les systèmes radiatifs dans les chapitres suivants.

- Chapitre.1 introduit l'objectif et la problématique de recherche de cette thèse. Nous passons d'abord en revue les différents systèmes optiques CSP, les outils numériques standard pour évaluer les performances optiques et les besoins d'estimation des sensibilités dans le système CSP. Cette thèse vise à développer une méthode pour estimer les sensibilités dans le système CSP. Pour atteindre cet objectif, nous passons en revue les trois méthodes de la communauté du transfert radiatif et de la synthèse d'images : la méthode de dérivation observable, la méthode de rendu différentiable et la méthode de modèle de sensibilité. Les avantages et les inconvénients sont répertoriés pour chaque méthode. Après cela, nous définissons la problématique de recherche de cette thèse : retrouver l'avantage de la vectorisation pour la méthode du modèle de sensibilité.

- Chapitre.2 compare et passe en revue les trois points de vue pour les estimations de sensibilités dans un système radiatif en les appliquant à un problème radiatif simple à une dimension. Les trois points de vue consistent en la méthode de dérivation observable développée dans la thèse de Maxime Roger [51], la méthode de rendu différentiable utilisée dans la communauté de synthèse d'images [70], et la méthode de modèle de sensibilité développée dans la thèse de Paule Lapeyre [32]. Le lien théorique entre les trois méthodes sera distingué, et les avantages et inconvénients de chaque méthode seront discutés.

  Dans ce chapitre, nous étudions trois méthodes pour calculer la sensibilité de l'intensité dans une étude de cas radiative unidimensionnelle. La solution analytique est connue à des fins de comparaison. RTE est utilisé pour décrire le transport d'intensité dans le milieu, et les conditions aux limites sont fixées. Le milieu est supposé froid avec des propriétés homogènes d'absorption et de diffusion. Deux conditions aux limites de Dirichlet sont posées sur les frontières. Après avoir construit le modèle de transport, le I observable est formulé dans une équation de Fredholm à l'aide d'outils mathématiques standard. Après cela, l'équation de Fredholm est transformée en une valeur attendue. Enfin, comme mentionné au chapitre.1, la valeur attendue peut être estimée par la méthode de Monte Carlo. De plus, l'algorithme de Monte Carlo permet de construire une image physique puisque chaque chemin mime le transport des photons.

  Ensuite, trois méthodes d'estimation de la sensibilité sont introduites et comparées. La méthode de dérivation observable repose sur la différenciation directe de l'observable sous une forme de valeur attendue. La méthode de rendu différentiable est basée sur la dérivation de l'observable sous la forme d'une équation de Fredholm. La formulation résultante est ensuite reformulée sous une forme de valeur attendue. La dernière méthode repose sur la différenciation du modèle d'intensité pour construire le modèle de sensibilité. La sensibilité de l'intensité devient le descripteur du modèle de sensibilité, absorbée et diffusée dans le système. Le modèle de sensibilité résultant est ensuite résolu, suivant le même processus que la résolution du modèle d'intensité. Le modèle d'intensité et de sensibilité sont couplés, mais cela est géré de manière algorithmique par la double randomisation.

  Par rapport aux deux autres méthodes, la méthode de dérivation observable est parfaitement vectorisée pour estimer simultanément l'intensité et la sensibilité (comme discuté au Chapitre.2). Cependant, le développement formel correspondant est très exigeant. Peu d'applications sont développées suivant cette méthode [9].

  La méthode de rendu différentiable est plus pertinente que la méthode précédente pour traiter des géométries complexes. D'autres applications sont trouvées pour améliorer la reconstruction de scènes 3D et concevoir des matériaux aux propriétés optiques spécifiées [44]. Cependant, il perd l'avantage de la vectorisation.

  La méthode du modèle de sensibilité partage le même algorithme que la méthode de rendu différentiable et reste adapté pour des applications plus complexes. De plus, cette méthode est d'un grand intérêt pour les médecins et les ingénieurs en raison

de l'image physique attachée. La sensibilité peut être décrite physiquement comme l'intensité aidant les médecins et les ingénieurs à mieux comprendre leurs systèmes.

Cependant, cette méthode perd également l'avantage de la vectorisation. Un système radiatif peut être caractérisé par des milliers de paramètres (par exemple, le système solaire à concentration). Les sensibilités de tous ces paramètres sont d'une grande importance à des fins d'optimisation. Afin d'estimer toutes les sensibilités, nous devons construire un modèle de sensibilité pour chacune d'entre elles. Si on les estime une à une, cela devient très coûteux en ressources de calcul. Ici, nous suggérons de **vectoriser partiellement les estimations de sensibilités**. Nous étudierons et analyserons avec soin les modèles de sensibilité dans le système radiatif, en essayant de regrouper et de vectoriser les estimations de sensibilités.

Dans les chapitres suivants, nous appliquerons la méthode du modèle de sensibilité dans certains systèmes géométriquement complexes. Nous étudierons et analyserons avec soin les modèles de sensibilité, regrouperons différentes sortes de sensibilités et vectoriserons leurs estimations. Néanmoins, nous aborderons également l'estimation de milliers de sensibilités qui reste hors gamme si la technique de vectorisation n'est pas appliquée.

- Chapitre.3 vise à appliquer la méthode du modèle de sensibilité dans des cas impliquant une géométrie complexe. Pour y parvenir, nous allons introduire une stratégie de vectorisation pour traiter un grand nombre de sources dans les modèles de sensibilités. Ce chapitre préparera l'application la plus complexe dans un système STP pour le chapitre suivant.

  Ce chapitre est consacré à l'introduction de la méthode de Monte-Carlo vectorisée, qui permet de préparer l'application de la méthode du modèle de sensibilité à un système CSP. Nous implémentons la méthode du modèle de sensibilité dans deux exemples impliquant une géométrie complexe et triangulée en utilisant la technique de vectorisation. Cette technique permet d'accélérer les estimations et de faciliter le développement du codage.

  Premièrement, nous avons construit un modèle d'intensité où les sources d'intensité sont spécifiées. Sur la base de ce modèle, des modèles de sensibilités peuvent être construits en utilisant les outils de [32–34], où les sources de sensibilités sont spécifiées.

  Dans un second temps, en suivant les trois étapes mentionnées au Chapitre.3 pour résoudre un modèle radiatif, nous avons formulé la formulation intégrale itérative suivant les propagations de toutes les sources (Eq.4.7 et Eq.4.14 dans le premier exemple et Eq.4.17,4.32,4.34,4.36 et Eq.4.37 dans le deuxième exemple).

  Enfin, nous avons distingué les intégrales ayant les mêmes domaines d'intégrations, et nous les avons écrites sous une forme vectorisée (Eq.4.15, Eq.4.44 et Eq.4.36). A partir de cette forme vectorisée, des algorithmes de Monte Carlo peuvent être conçus pour résoudre les modèles radiatifs.

  Cette stratégie de vectorisation consiste à **vectoriser le transport des sources qui partagent le même espace de chemin.** Si l'on veut distinguer les sources qui partagent le même espace de chemin, il faut vérifier les **localisations** des sources sur les frontières et le **transport** des sources dans le milieu. Lorsque les sources sont aux mêmes endroits sur la frontière et transportées par le même RTE dans le milieu, les sources peuvent être vectorisées.

  Pour le premier exemple, I et s_ρ sont transportés de la même manière dans le milieu (Eq.4.2 et Eq.4.8). Sur la frontière, la source $\mathring{I}$ et $\mathring{s}_\rho$ sont toutes sur toutes les surfaces de $\mathscr{T}$. Ainsi, la propagation de I et s_ρ peut être vectorisée.

  Pour le deuxième exemple, I et $s_i, \forall j \in \{1,2,3,4\}$ sont transportés également de la même manière dans le milieu (Eq.4.20 et Eq.4.23). Cependant, les sources de

sensibilités $\dot{s}_{i,j}, \forall i \in \{1, 2, \ldots, n_{\mathscr{T}}\}, \forall j \in \{1, 2, 3, 4\}$ sont trouvés sur chaque surface de $\mathscr{T}$ et $\hat{s}_{i,j,k}, \forall i \in \{1, 2, \ldots, n_{\mathscr{T}}\}, \forall j \in \{1, 2, 3, 4\}, \forall k \in \{1, 2, 3\}$ sont trouvés sur chaque bord de chaque triangle $\mathscr{T}_i$. Ainsi, les propagations de $\dot{s}_{i,j}$ et I (qui est aussi une source surfacique) et les propagations des sources linéaires $\hat{s}_{i,j,k}$ sont vectorisés séparément. Même avec cette stratégie de vectorisation, il n'est toujours pas facile de gérer toutes les sources surfaciques et les sources linéaires sur chaque triangle d'une géométrie complexe et triangulée. Par conséquent, plus tard dans le chapitre.6, des efforts seront faits pour transformer toutes les sources linéaires en sources surfaciques. Par conséquent, il ne reste que des sources de surface, et les estimations de l'observable et de toutes ses sensibilités peuvent être complètement vectorisées.

- Au chapitre.5, nous appliquons la méthode du modèle de sensibilité dans un système STP, en suivant la stratégie de vectorisation présentée dans le dernier chapitre. Aussi, grâce aux modèles de sensibilités, l'interprétation physique des sensibilités peut être réalisée. Les contributions de différents événements physiques dans un système STP (blocage, ombrage, déversement, etc.) aux sensibilités peuvent être analysées en détail, aidant les ingénieurs à comprendre et à optimiser le système. Nous avons présenté des modèles généraux pour l'intensité I et la sensibilité d'intensité $s_{i,j}$ dans un système radiatif d'un système SPT, ainsi que les algorithmes complets d'estimation de la matrice de sensibilité de puissance $\boldsymbol{S}$. Ces algorithmes sont validés par la méthode des différences finies à l'aide de quatre exemples types. Puisque la sensibilité d'un héliostat à un paramètre géométrique $S_{i,j}$ est estimé par le cumul de 4 contributions : $S_{i,j}^{tar}$, $S_{i,j}^{blo}$, $S_{i,j}^{shad-b}$ et $S_{i,j}^{shad-f}$, les contributions de différents événements physiques à la sensibilité $S_{i,j}$ dans un système SPT (blocage, masquage, déversement) peut être analysée en détail. Après cela, un système SPT fonctionnel réel a été utilisé comme étude de cas test. Dans un premier temps, une comparaison des temps de calcul a été faite pour l'estimation de la sensibilité par notre méthode et son approximation par la méthode des différences finies. Dans un second temps, la matrice de sensibilité de puissance $\boldsymbol{S}$ a été calculée. Les ingénieurs peuvent bénéficier de ces informations sur les sensibilités pour optimiser la conception optique et la stratégie de visée du système SPT. Enfin et surtout, à la suite de ce travail, l'intégration d'optimisations basées sur le gradient pour le système optique dans un SPT devient possible. Cependant, le modèle de sensibilité développé dans le présent article n'est appliqué qu'à un champ d'héliostats plats. Ainsi, une perspective prometteuse serait d'étendre ce modèle aux héliostats courbes.

- Suivant la méthode du modèle de sensibilité, le Chapitre.6 vise à retrouver l'avantage de la méthode de dérivation observable : une vectorisation complète. Cela se fera en transformant les sources linéaires en sources superficielles dans le modèle de sensibilité. Quelques exemples académiques sont développés, et certaines limites sont discutées.

  La méthode du modèle de sensibilité introduit à la fois des sources linéaires et de surface. Les inconvénients sont dus aux lignes sources lors de l'application de la méthode de Monte Carlo pour résoudre ce genre de problème. Dans ce chapitre, nous étudions la transformation de sources linéaires en sources surfaciques avec le théorème de Stokes sous certaines hypothèses (telles que géométrie convexe, milieu froid, propriétés homogènes, etc.). Il n'existe plus de source de ligne dans le système et l'échantillonnage de ligne est évité. De plus, toutes les sources de surface partagent le même espace de chemin. L'observable et sa dérivée spatiale sont alors estimées sous une forme vectorisée, et une seule estimation de Monte Carlo est nécessaire. C'est un avantage non négligeable pour la méthode de dérivation observable, introduite au Chapitre.2 et au Chapitre.3. Ici, en suivant la méthode du modèle de sensibilité, nous retrouvons complètement cet avantage dans ce chapitre.

  Cependant, un problème de convergence est rencontré lors de l'application de cette technique à un milieu diffusant. Nous n'avons pas de solutions pour le moment, mais la

solution potentielle pourrait être d'appliquer les techniques de contrôle de la variance de la méthode de Monte-Carlo [14].

- Annexe A vises à construire les modèles de transport pour la sensitbilité, le gradient spatiale et le gradient angulaire. L'idée de la vectorisation est qu'au lieu de propager un scalaire en tant que descripteur du modèle de transport, nous propageons un vecteur en tant que descripteur du transport. Dans cette annexe, nous construisons les modèles de transport sous forme vectorielle (voir Fig.1). Cette méthode de vectorisation nécessite de grands efforts formels. Des études supplémentaires sont nécessaires si nous voulons l'appliquer à des applications spécifiques.

  Dans cette annexe, nous allons d'abord présenter la méthode de modèle de sensibilité développée lors du travail de thèse de Paule Lapeyre. Dans cette section, le modèle général de transport d'intensité sera examiné. Selon les besoins des différentes applications, les observables sont estimés en fonction du modèle général de transport d'intensité. Ici, les sensibilités d'un observable par rapport à différents paramètres sont également estimées, en suivant la méthode du modèle de sensibilité. La méthode du modèle de sensibilité vise à construire un modèle de transport pour la sensibilité de l'intensité, similaire au modèle de transport de l'intensité. L'observable est généralement formulé comme l'intégrale spatiale ou angulaire de l'intensité. En suivant la méthode du modèle de sensibilité, la sensibilité d'un observable est généralement formulée comme l'intégrale spatiale ou angulaire de la sensibilité de l'intensité. Nous présentons donc le modèle général de transport pour la sensibilité de l'intensité. Cependant, le modèle de sensibilité est couplé avec deux autres quantités : la dérivée spatiale de l'intensité et la dérivée angulaire de l'intensité. Leurs définitions sont également données, et les modèles de transport généraux sont introduits respectivement. Enfin, l'estimation de la sensibilité d'un observable nécessite la résolution des quatre modèles de transport couplés : le modèle d'intensité, le modèle de sensibilité, le modèle de dérivée spatiale et le modèle de dérivée angulaire.

  Ensuite, nous allons introduire la stratégie de vectorisation. Les quatre modèles introduits décrivent le transport de quatre scalaires : l'intensité, la sensibilité, la dérivée spatiale et la dérivée angulaire. En suivant la stratégie de vectorisation, nous réécrivons simplement les quatre modèles sous forme vectorisée, où les quatre modèles sont construits pour un vecteur de sensibilités, un vecteur de dérivées spatiales et un vecteur de dérivées angulaires. Ce geste simple nous permet de transporter les descripteurs sous forme vectorisée. Par exemple, au lieu de transporter chaque sensibilité correspondant à un paramètre par son propre chemin dans le système, nous pouvons transporter un vecteur de sensibilités correspondant à chaque paramètre en un seul chemin.

Au cours de ces trois années, l'auteur travaille entre l'interface du CSP, la physique du transfert radiatif, et la communauté de la synthèse d'images (Fig.1). Du fait du caractère pluridisciplinaire de cette thèse, elle se conclut sur ces trois points de vue.

Du point de vue du CSP, une méthode pour estimer les sensibilités dans un système STP est fournie au chapitre.5. Il prolonge la première tentative d'estimation des sensibilités proposée par la thèse. travail de De la Torre, qui n'est valable que lorsque l'ombrage et le blocage sont négligés. Le présent travail comble le vide concernant l'estimation de la sensibilité dans la communauté CSP. Les sensibilités des translations, des rotations et la taille de chaque héliostat à la puissance d'impact sur le récepteur peuvent être estimées avec précision. La stratégie de vectorisation rend possible l'estimation de milliers de sensibilités dans un temps de calcul raisonnable. De plus, chaque sensibilité peut être interprétée physiquement, indiquant comment les événements physiques dans le système STP (blocage, déversement, ombrage, etc.) affectent les sensibilités, rendant certaines plus sensibles que d'autres. Tous les types de systèmes STP avec des miroirs plats peuvent bénéficier de cette

méthode. Grâce aux informations fournies par cette méthode, les ingénieurs et les chercheurs peuvent :

1. Analysez le champ optique d'un système STP et améliorez les performances optiques en ajustant les paramètres les plus sensibles pendant le fonctionnement.
2. Réaliser une analyse détaillée du champ optique. Les informations sur les contributions des événements physiques (blocage, déversement, ombrage, etc.) aux sensibilités sont fournies, aidant les ingénieurs à mieux comprendre le système optique et à effectuer l'analyse de sensibilité.
3. Associez les informations sur les sensibilités (le gradient) à un algorithme basé sur le gradient à des fins d'optimisation. Ceci est d'un grand intérêt pour la communauté CSP car l'optimisation repose principalement sur des algorithmes sans gradient tels que les algorithmes génétiques ou l'optimisation par essaim de particules.

Du point de vue de la physique des transferts radiatifs l'idée de vectorisation a été proposé et mis en œuvre après le récent doctorat. oeuvre de Paule Lapeyre [31–34]. Grâce à cette stratégie de vectorisation, la méthode du modèle de sensibilité peut être appliquée aux cas impliquant des géométries complexes et triangulées avec un temps de calcul raisonnable et un moindre effort de codage.

Néanmoins, une technique de transformation de sources linéaires en sources surfaciques est présentée au chapitre.6. Cette technique permet d'éviter l'échantillonnage linéaire et de vectoriser complètement l'observable et son estimation de dérivation spatiale.

Du point de vue de la synthèse d'images , tous les résultats obtenus dans cette thèse s'appuient sur les bibliothèques d'infographie de la communauté de la synthèse d'images.

La technique étudiée au chapitre.6 permet d'éviter le "sampling on segments" considéré comme un sujet complexe dans la communauté. Les chercheurs en synthèse d'images proposent des méthodes compliquées pour surmonter cette difficulté [64]. Cependant, ici, nous essayons de résoudre physiquement ce problème (en transformant les sources linéaires en sources surfaciques).

Au chapitre.4, en appliquant la stratégie de vectorisation, une image de sensibilité est rendue au prix d'un faible effort de calcul supplémentaire lors du rendu de l'image d'intensité. L'image de sensibilité permet d'effectuer de l'optimisation et du machine learning dans la communauté de la synthèse d'images[26].

De plus, la littérature sur la synthèse d'images concernant l'estimation des sensibilités par Monte-Carlo est passée en revue. Bien que les vocabulaires et les points de vue soient différents, nous avons comparé leurs méthodes et les nôtres (développées dans le groupe EDstar[5]) au chapitre.3. Il est passionnant de constater que même si les points de vue sont différents, la méthode de rendu différentiable utilisée par la communauté de la synthèse d'images et la méthode du modèle de sensibilité convergent vers le même algorithme dans l'exemple développé au chapitre.3.

Ce doctorat. travail inspire les perspectives suivantes :

1. La méthode développée au chapitre.5 est limitée aux miroirs plans. Cependant, de nombreux systèmes STP sont construits à partir de miroirs courbes (hémisphériques ou paraboliques). Il existe deux manières de décrire la courbure de l'héliostat : 1) en utilisant la surface quadrique de l'héliostat courbe ; 2) en utilisant une géométrie maillée composée de triangles. Des travaux supplémentaires sont nécessaires pour mettre en évidence les différences et/ou la praticabilité des deux approches.

[5] http://www.edstar.cnrs.fr/prod/fr/

2. Même s'il est possible d'estimer uniquement la puissance d'impact au sommet d'un STP, il serait également possible de construire une "flux map" de sensibilité. Cela serait utile pour éviter le "point chaud" et homogénéiser le flux sur le récepteur.

3. Les sensibilités fournies par notre méthode comportent des incertitudes. A notre connaissance, les incertitudes ne sont pas prises en compte dans les algorithmes d'optimisation par gradient. Cela deviendrait crucial lorsque le gradient tend vers zéro puisqu'il ne serait pas possible de distinguer si le gradient estimé est positif ou négatif. Une façon naïve de gérer cela serait d'augmenter les itérations de Monte Carlo. Cependant, cette approche nécessite plus de développement.

4. Au chapitre.6, le problème de convergence est distingué lorsque la technique de transformation des sources linéaires en sources surfaciques est appliquée dans un milieu diffusant. L'application de méthodes de réduction de la variance pourrait être une solution potentielle à ce problème de convergence.

5. L'étude de Chapitre.6 est restreinte aux géométries convexes. L'application de la technique de transformation de sources linéaires en sources de surface dans une géométrie non convexe est un défi. Le théorème de Stokes ne peut pas être appliqué directement lorsque les triangles sont bloqués les uns des autres.

6. Enfin, la méthode de rendu différentiable est bien adaptée à la géométrie complexe. L'appliquer dans des applications CSP peut être un moyen prometteur d'estimer les sensibilités. Le principal défi serait de lier les différents problèmes de recherche dans les deux communautés. En synthèse d'images, le principal défi est de rendre les images ce qui signifie que les algorithmes sont basés sur le lancer de rayons inversé. Par conséquent, des efforts sont nécessaires pour adapter ces algorithmes au champ CSP.

Résumé

La méthode Monte-Carlo vectorisée pour les modèles de sensibilité en transfert radiatif : application en solaire à concentration

Les travaux présentés dans ce manuscrit abordent la question de la vectorisation des évaluations de sensibilité pour les systèmes de transfert radiatif impliquant des milieux semi-transparents. La luminance est la descriptrice principale du transfert radiatif. Dans ce travail, les sensibilités de la luminance par rapport aux paramètres caractérisant le système radiatif sont également utilisées comme descripteurs du transfert radiatif. Elles portent l'information comme les perturbations de la luminance causées par les perturbations des paramètres. Des modèles de transport de la luminance et de sensibilité peuvent alors être construits. Ensuite, une traduction standard de l'intégrale de chemin est effectuée pour concevoir des algorithmes de Monte-Carlo résolvant le modèle de luminance et les modèles de sensibilités. Des difficultés sont trouvées lorsque le nombre de sensibilités augmente et lorsque la dérivation de propriété radiative fait intervenir des discontinuités. Les chemins de rayons statistiques nécessaires deviennent nombreux.

Dans cette thèse, les espaces des chemins statistiques de la luminance et des sensibilités sont étudiés et reformulés. Le transport de la luminance et des sensibilités peut être donc vectorisé. Cette technique de vectorisation permet de transporter un vecteur (luminance et sensibilités sont écrites dans le même vecteur) au lieu d'un scalaire (luminance ou sensibilité séparément). Enfin, cette technique est ensuite appliquée pour estimer les sensibilités géométriques dans les systèmes d'énergie solaire concentrée, et la physique des sensibilités est interprétée et discutée.

Mots-clés : Méthodes de Monte Carlo, Transfert radiatif, Vectorisation, Solaire à concentration, Modèle de sensibilité

Abstract

Vectorized Monte-Carlo Method for sensitivity models in radiative transfer: application to Concentrated Solar Power

The present manuscript addresses the question of vectorizing the sensitivity evaluations for radiative transfer systems involving semi-transparent media. Intensity is the usual descriptor of radiative transfer. In this work, the sensitivities of intensity with respect to parameters characterizing the radiative system are also used as descriptors of radiative transfer. They characterize the perturbations to intensity, originating from the perturbations of parameters. Transport models of intensity and sensitivity can then be built. After that, a standard path integral translation is made to design Monte-Carlo algorithms solving the model of intensity and the models of sensitivities. Inconveniences are found when the number of sensitivities increases and when the discontinuities of radiative properties are encountered because the number of required statistical ray paths becomes very high.

In this thesis, the statistical path space of intensity and sensitivities are studied and reformulated, and then their transports can be vectorized. Instead of constructing ray paths for the transport of a scalar (intensity or sensitivity separately), ray paths are constructed for the transport of a vector (intensity and sensitivities are written into the same vector). Last but not least, this technique is then applied to estimate the geometric sensitivities in concentrated solar power systems, and the physics of the sensitivities is interpreted and discussed.

Keywords: Monte Carlo methods, Radiative transfer, Vectorization, Concentrated solar power, Model of sensitivity