Integral formulation of null-collision Monte Carlo algorithms

Article · August 2013

DOI: 10.1016/j.jqsrt.2013.04.001

CITATIONS
97

READS
788

12 authors, including:

Mathieu Galtier
Institut National des Sciences Appliquées de Lyon
48 PUBLICATIONS **399** CITATIONS
SEE PROFILE

Stéphane Blanco
Paul Sabatier University - Toulouse III
153 PUBLICATIONS **2,749** CITATIONS
SEE PROFILE

C. Caliot
French National Centre for Scientific Research
104 PUBLICATIONS **2,076** CITATIONS
SEE PROFILE

Christophe Coustet
méso-star
30 PUBLICATIONS **462** CITATIONS
SEE PROFILE

Integral formulation of null-collision Monte Carlo algorithms

M. Galtier[1], S. Blanco[2,3], C. Caliot[4], C. Coustet[5], J. Dauchet[6], M. El Hafi[1], V. Eymet[7], R. Fournier[2,3], J. Gautrais[8], A. Khuong[8], B. Piaud[5], G. Terrée[1]

1- Université de Toulouse, Mines Albi, UMR 5302 - Centre de Recherche d'Albi en génie des Procédés des Solides Divisés, de l'Energie et de l'Environnement (RAPSODEE), Campus Jarlard, F-81013, Albi CT cedex 09, France

2- Université de Toulouse ; UPS, INPT ; LAPLACE (Laboratoire Plasma et Conversion d'Energie) ; 118 route de Narbonne, F-31062 Toulouse cedex 9, France

3- CNRS ; LAPLACE ; F-31062 Toulouse, France

4- Processes, Materials and Solar Energy Laboratory (PROMES), CNRS, 7 rue du Four Solaire, Font-Romeu-Odeillo, F-66120, France

5- HPC-SA, 3 chemin du Pigeonnier de la Cépière, Bâtiment C, F-31100, Toulouse, France

6- Clermont Université, ENSCCF, Institut Pascal - UMR 6602, BP 10448, F-63000 Clermont-Ferrand, France

7- Université Bordeaux 1, UMR 5804 - Laboratoire d'Astrophysique de Bordeaux (LAB), 2 rue de l'Observatoire BP 89, F-33271, Floirac Cedex, France

8- Centre de Recherches sur la Cognition Animale, CNRS UMR5169, Université de Toulouse, France

Abstract

At the kinetic level, the meaning of null-collisions is straightforward: they correspond to pure-forward scattering events. We here discuss their technical significance in integral terms. We first consider a most standard null-collision Monte Carlo algorithm and show how it can be rigorously justified starting from a Fredholm equivalent to the radiative transfer equation. Doing so, we also prove that null-collision algorithms can be slightly modified so that they deal with unexpected occurrences of negative values of the null-collision coefficient (when the upper bound of the heterogeneous extinction coefficient is nonstrict). We then describe technically, in full details, the resulting algorithm, when applied to the evaluation of the local net-power density within a bounded, heterogeneous, multiple scattering and emitting/absorbing medium. The corresponding integral formulation is then explored theoretically in order to distinguish the statistical significance of introducing null-collisions from that of the integral-structure underlying modification.

1 Introduction

The introduction of null-collisions in the process of modelling photon transport consists in transforming the standard radiative transfer equation

$$\frac{\partial f}{\partial t} + c\boldsymbol{\omega}.\boldsymbol{\nabla} f = -(k_a + k_s)cf + \mathcal{S} + \int_{4\pi} k_s c f' p(\boldsymbol{\omega}|\boldsymbol{\omega'}) d\boldsymbol{\omega'} \tag{1}$$

into

$$\frac{\partial f}{\partial t} + c\boldsymbol{\omega}.\boldsymbol{\nabla} f = -(k_a + k_s + k_n)cf + \mathcal{S} + \int_{4\pi} k_s c f' p_S(\boldsymbol{\omega}|\boldsymbol{\omega'}) d\boldsymbol{\omega'} + \int_{4\pi} k_n c f' \delta(\boldsymbol{\omega} - \boldsymbol{\omega}') d\boldsymbol{\omega'} \tag{2}$$

where

- $f \equiv f(\mathbf{x}, \boldsymbol{\omega}, t)$ is the distribution function at location $\mathbf{x}$, propagation direction $\boldsymbol{\omega}$ and time t. The distribution function is used here, instead of the specific intensity $I = h\nu cf$, in order to help readers from other particle transport communities such as neutron transport, plasma physics and rarefied gas dynamics, that have made an intensive use of null-collision approaches [1, 2, 3] (see A for a brief description of the rather complex structure of the corresponding literature).
- c is the speed of light, $k_a(\mathbf{x}, t)$ the absorption coefficient, $k_s(\mathbf{x}, t)$ the scattering coefficient, $p_S(\boldsymbol{\omega}|\boldsymbol{\omega'}) \equiv p_S(\boldsymbol{\omega}|\boldsymbol{\omega'}, \mathbf{x})$ the single scattering phase function, that is to say the probability density that the scattering direction is $\boldsymbol{\omega}$ for a photon initially in the direction $\boldsymbol{\omega'}$. The notation f' in the scattering source integral stands for $f' \equiv f(\mathbf{x}, \boldsymbol{\omega'}, t)$.
- $\mathcal{S} \equiv \mathcal{S}(\mathbf{x}, \boldsymbol{\omega}, t)$ is any source term. We will define $s \equiv s(\mathbf{x}, \boldsymbol{\omega}, t)$ such that $\mathcal{S} = k_a cs$, and therefore $s = f^{eq}(\mathbf{x}, t)$ in the particular case of thermal emission under the assumption of local thermodynamic equilibrium, where $f^{eq}(\mathbf{x}, t)$ is the distribution function at equilibrium at local temperature (related to the Planck specific intensity B according to $B = h\nu cf^{eq}$).
- k_n is the null-collision coefficient and δ is the Dirac distribution.

Additional collisions are introduced via the term $-k_n cf$ but these collisions are cancelled out, as they are scattering events in the pure forward direction (the phase function is $\delta(\boldsymbol{\omega} - \boldsymbol{\omega}')$ in the scattering source integral), and leave the f field unchanged, which is a direct consequence of the property $\int_{4\pi} k_n c f' \delta(\boldsymbol{\omega} - \boldsymbol{\omega'}) d\boldsymbol{\omega'} = k_n cf$. To the best of our knowledge, outside the above mentioned transport physics literature, the only reported practical use of null-collision approaches for radiative transfer applications are in the fields of computer graphics and medical imaging [4] [5].

Such applications are related to Monte Carlo simulations in which the heterogeneity of the absorption and scattering coefficients does not allow the implementation of simple free path sampling algorithms. When defining the location of the next collision event, the common practice is indeed to first sample an extinction optical thickness τ according to the probability density function $p_T(\tau) = exp(-\tau)$, and then derive the corresponding path length λ by inverting the function relating τ to λ: $\tau(\lambda) = \int_0^\lambda k(\mathbf{x} + \sigma\boldsymbol{\omega}, \boldsymbol{\omega}, t + \frac{\sigma}{c}) d\sigma$, where $k = k_a + k_s$. However, if k_a and k_s are complex functions of space, this inversion is difficult to perform analytically. Most usually, k_a is then approximated with discretization approaches, but this implies a rigorous control of the corresponding approximation level. Introducing null-collisions is a way to avoid such approximations.

A null-collision k_n field can indeed be introduced so that the modified extinction optical thickness $\hat{k} = k_a + k_s + k_n$ (corresponding to absorption plus true scattering plus null-collision) allows tractable $\tau(\lambda)$ inversions (e.g. $\hat{k}$ uniform). Practically,

- $\hat{k}$ is arbitrarily chosen as an upper bound of the true extinction field k ($\hat{k} > k$) and k_n is then defined as $k_n = \hat{k} - k$ (note that the choice is made on $\hat{k}$, not on k_n, so that $\hat{k}$ has the expected inversion properties);

- a collision location is sampled by first sampling $\hat{\tau}$ according to p_T and inverting $\hat{\tau}(\lambda) = \int_0^\lambda \hat{k}(\mathbf{x}+\sigma\boldsymbol{\omega},\boldsymbol{\omega},t+\frac{\sigma}{c})d\sigma$;
- a random number r is sampled uniformly on the unit interval and the collision is considered as an absorption event if $0 < r < \frac{k_a}{\hat{k}}$, as a real scattering event if $\frac{k_a}{\hat{k}} < r < \frac{k_a+k_s}{\hat{k}}$, or as a pure forward scattering event if $\frac{k_a+k_s}{\hat{k}} < r < \frac{k_a+k_s+k_n}{\hat{k}} = 1$ (fortune wheel).

This technique is well suited to the recent Monte Carlo developments toward flexible validation tools for accuracy control of fast radiation solvers (interacting with chemistry and fluid mechanics). In such contexts, field representation is bound to the specificity of each solver in an intricate manner and null-collision algorithms make it possible to design transversal meshless[1] Monte Carlo codes that are immediately applicable whatever the retained solver numerics [6].

The present technical note addresses the question of using integral formulation techniques for refining Monte Carlo algorithms involving null-collisions. For didactic reasons, we first consider the academic question of evaluating the distribution function (at a given point in a given direction) in an heterogeneous emitting/absorbing infinite medium using a backward algorithm (Sec. 2). The corresponding integral formulation is constructed step by step as a translation of the above described null-collision algorithm. This formulation is then modified so that the constraint $\hat{k} > k$ is relieved: negative values of the null-collision coefficient are accepted. This is practically very significant because $\hat{k}$ must be chosen to match k as closely as possible (otherwise too many useless collisions are sampled), which is a delicate task when the constraint $\hat{k} > k$ is strict. This first technical proposition is synthesised in Sec. 3, with the complete description of a Monte Carlo algorithm evaluating the local net-power density within a bounded, heterogeneous, multiple scattering and emitting/absorbing medium. A second technical proposition is made in Sec. 4: an integral formulation is constructed that helps clarify the significance of introducing null-collisions, in particular as far as convergence is concerned. This formulation indicates that the problem of sampling free paths in heterogeneous fields could be bypassed without introducing any null-collision concept, but sign alternations would appear that would be sources of statistical variance. It is then shown that the benefit of introducing null-collisions is to break this sign alternation. We therefore suggest to preserve the idea of introducing a $\hat{k}$ field, but without imposing that free paths be sampled according to $\hat{k}$, or that the type of collision (absorption, true scattering or forward continuation) be sampled according to the respective proportions of k_a, k_s and $k_n = \hat{k} - k_a - k_s$. A wider class of Monte Carlo algorithms is therefore identified that could be explored for convergence enhancement.

[1] "meshless" is here used to indicate that the Monte Carlo algorithm requires no volume discretization. Therefore, if the input fields of temperature and extinction coefficients are analytical (as in benchmarking exercises) no mesh is used at all. However, if the input fields are provided using a volume discretization and an interpolation procedure, the grid is rigorously respected. The idea is that the input fields can take any form and that the Monte Carlo algorithm introduces no mesh by itself.

2 Theoretical justification and extension to negative values of the null-collision coefficient

In the particular case of stationary radiation[2] in a non-scattering infinite medium, the distribution function at location $\mathbf{x}$ in the direction $\boldsymbol{\omega}$ takes the following integral form (solution of Eq. 1):

$$f(\mathbf{x},\boldsymbol{\omega}) = \int_0^{+\infty} k_{a,\lambda} s_\lambda exp\left(-\int_0^\lambda k_{a,\sigma} d\sigma\right) d\lambda \tag{3}$$

Throughout this note, in all non-recursive integral formulations, the notations $k_{a,\alpha}$, $k_{s,\alpha}$, $k_{n,\alpha}$, $\hat{k}_\alpha$, s_α and f^{eq}_α are used to represent $k_a(\mathbf{x}-\alpha\boldsymbol{\omega})$, $k_s(\mathbf{x}-\alpha\boldsymbol{\omega})$, $k_n(\mathbf{x}-\alpha\boldsymbol{\omega})$, $\hat{k}(\mathbf{x}-\alpha\boldsymbol{\omega})$, $s(\mathbf{x}-\alpha\boldsymbol{\omega},\boldsymbol{\omega})$ and $f^{eq}(\mathbf{x}-\alpha\boldsymbol{\omega})$ respectively, where α is any propagation-distance along the line of sight. Standard backward Monte Carlo algorithms start from Eq. 3 and introduce the random variable Λ corresponding to the distribution of absorption free paths λ in the $-\boldsymbol{\omega}$ direction, of probability density $p_\Lambda(\lambda) = k_{a,\lambda} exp\left(-\int_0^\lambda k_{a,\sigma} d\sigma\right)$, to get

$$f(\mathbf{x},\boldsymbol{\omega}) = \int_0^{+\infty} p_\Lambda(\lambda) d\lambda \; s_\lambda \tag{4}$$

$f(\mathbf{x},\boldsymbol{\omega})$ is then interpreted as the expectation of $s(\mathbf{x}-\Lambda\boldsymbol{\omega},\boldsymbol{\omega})$ which leads to the Monte Carlo algorithm of Fig. 1. Even if one decides to make use of a null-collision technique, it does not appear explicitly in such a presentation: it is only implicit in the way the Beer sampling of λ is performed.

Alternatively, all the details of using null-collisions can be put forward as in the complete algorithm of the left part of Fig. 2. A strict formal translation of this algorithm is displayed on the right part of the figure, where the Heaviside notation $\mathcal{H}(test)$ is used to represent 1 if *test* is true and 0 otherwise. This integral formulation can be derived from the following Fredholm equation, a well-known translation of the radiative transfer equation (here of Eq. 2 at stationary state, including null-collisions interpreted as forward scattering events):

$$\begin{aligned} f(\mathbf{x},\boldsymbol{\omega}) = \int_0^{+\infty} & exp\left(-\int_0^\lambda (k_{a,\sigma}+k_{n,\sigma}) d\sigma\right) \times \\ & \left[k_{a,\lambda} s_\lambda + k_{n\lambda} \int_{4\pi} \delta(\boldsymbol{\omega}-\boldsymbol{\omega'}) f(\mathbf{x}-\lambda\boldsymbol{\omega},\boldsymbol{\omega'}) d\boldsymbol{\omega'}\right] d\lambda \end{aligned} \tag{5}$$

We now give all the details of this derivation, justifying meanwhile the corresponding null-collision Monte-Carlo algorithm of Fig. 2 and we then extend it in order to allow negative values of the null-collision coefficient.

The first step is solving the Dirac integration and using the recursive notations $\mathbf{x_0} \equiv \mathbf{x}$ and $\mathbf{x_{j+1}} = \mathbf{x_j} - \lambda_j \boldsymbol{\omega}$ to get

$$\begin{aligned} f(\mathbf{x_j},\boldsymbol{\omega}) = \int_0^{+\infty} & exp\left(-\int_0^{\lambda_j} \hat{k}(\mathbf{x_j}-\sigma_j\boldsymbol{\omega}) d\sigma_j\right) \times \\ & \left[k_a(\mathbf{x_{j+1}}) s(\mathbf{x_{j+1}},\boldsymbol{\omega}) + k_n(\mathbf{x_{j+1}}) f(\mathbf{x_{j+1}},\boldsymbol{\omega})\right] d\lambda_j \end{aligned} \tag{6}$$

[2]Transient radiation would induce no specific theoretical difficulty, but it would make the integral formulation much heavier. The extinction coefficients would indeed be functions of time and time would itself depend on path-length.

$\tilde{f}_N = 0;$
foreach i *in* $1 : N$ **do**
  Beer sampling of λ;
  $w = s_\lambda$;
  $\tilde{f}_N = \tilde{f}_N + w;$
end
$\tilde{f}_N = \tilde{f}_N / N;$

$$f(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{+\infty} p_\Lambda(\lambda) d\lambda \; w$$

with

$$w = s_\lambda$$

$$p_\Lambda(\lambda) = k_{a,\lambda} exp\left(- \int_0^\lambda d\sigma k_{a,\sigma} \right)$$

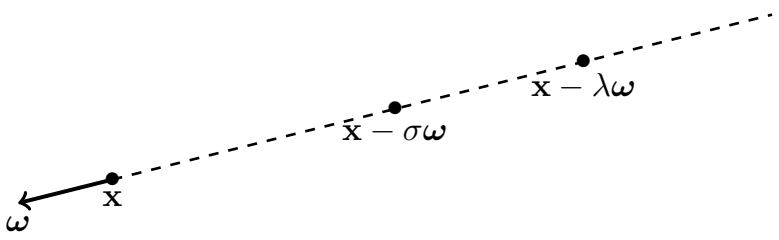


Figure 1: The reciprocal algorithm. $\tilde{f}_N$ is a Monte Carlo estimate of $f(\mathbf{x}, \boldsymbol{\omega})$ justified by Eq. 4. The integral formulation displayed on the right side of the algorithm box is a strict formal translation of the algorithm description.

$\tilde{f}_N = 0$;
foreach i *in* $1 : N$ **do**
  $j = 0$; $w = 0$; $\mathbf{x_0} = \mathbf{x}$;
  $abs = false$;
  while $abs = false$ **do**
    Beer sampling of λ_j;
    Uniform sampling of r_j;
    $\mathbf{x_{j+1}} = \mathbf{x_j} - \lambda_j \boldsymbol{\omega}$;
    if $r_j < \frac{k_a(\mathbf{x_{j+1}})}{\hat{k}(\mathbf{x_{j+1}})}$ **then**
      $w = s(\mathbf{x_{j+1}}, \boldsymbol{\omega})$;
      $\tilde{f}_N = \tilde{f}_N + w$;
      $abs = true$;
    end
    $j = j + 1$;
  end
end
$\tilde{f}_N = \tilde{f}_N / N$;

$$f(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{+\infty} p_{\Lambda_0}(\lambda_0) d\lambda_0 \int_0^1 p_{R_0}(r_0) dr_0$$
$$\Big\{ \mathcal{H}\Big(r_0 < \frac{k_a(\mathbf{x_1})}{\hat{k}(\mathbf{x_1})}\Big) w_1 + \mathcal{H}\Big(r_0 > \frac{k_a(\mathbf{x_1})}{\hat{k}(\mathbf{x_1})}\Big) \int_0^{+\infty} p_{\Lambda_1}(\lambda_1) d\lambda_1 \int_0^1 p_{R_1}(r_1) dr_1$$
$$\Big\{ \mathcal{H}\Big(r_1 < \frac{k_a(\mathbf{x_2})}{\hat{k}(\mathbf{x_2})}\Big) w_2 + \mathcal{H}\Big(r_1 > \frac{k_a(\mathbf{x_2})}{\hat{k}(\mathbf{x_2})}\Big) \int_0^{+\infty} p_{\Lambda_2}(\lambda_2) d\lambda_2 \int_0^1 p_{R_2}(r_2) dr_2$$
$$\ldots \Big\} \Big\}$$

with

$$\mathbf{x_0} = \mathbf{x} \ ; \ \mathbf{x_{j+1}} = \mathbf{x_j} - \lambda_j \boldsymbol{\omega}$$
$$p_{\Lambda_j}(\lambda_j) = \hat{k}(\mathbf{x_j} - \lambda_j \boldsymbol{\omega}) exp\Big(-\int_0^{\lambda_j} \hat{k}(\mathbf{x_j} - \sigma_j \boldsymbol{\omega}) d\sigma_j\Big)$$
$$p_{R_j}(r_j) = \tfrac{1}{1-0} = 1$$
$$w_j = s(\mathbf{x_j}, \boldsymbol{\omega})$$

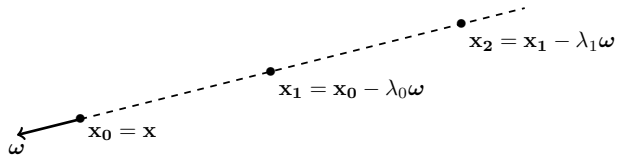


Figure 2: The standard null-collision algorithm. $\tilde{f}_N$ is a Monte Carlo estimate of $f(\mathbf{x}, \boldsymbol{\omega})$. The integral formulation displayed on the right side of the algorithm box is a strict formal translation of the algorithm description. The Monte Carlo weight is w_j when the j-th collision is the first true collision (the preceding collisions are null-collisions). The whole algorithm could also be presented as in Fig. 1 with $\lambda = \lambda_0 + \lambda_1 + ... + \lambda_{j-1}$, $\mathbf{x} - \lambda \boldsymbol{\omega} = \mathbf{x_j}$ and $s_\lambda = s(\mathbf{x_j}, \boldsymbol{\omega})$, and the appropriate change of the coefficient k used in $p_\Lambda(\lambda)$.

Then, the probability density of the j^{th} free path is introduced:

$$p_{\Lambda_j}(\lambda_j) = \hat{k}(\mathbf{x_j} - \lambda_j \boldsymbol{\omega}) exp\left(- \int_0^{\lambda_j} \hat{k}(\mathbf{x_j} - \sigma_j \boldsymbol{\omega}) d\sigma_j \right) \tag{7}$$

as well as non-zero probabilities P_j, to give

$$\begin{aligned} f(\mathbf{x_j}, \boldsymbol{\omega}) = \int_0^{+\infty} p_{\Lambda_j}(\lambda_j) d\lambda_j \Bigg[P_{j+1} \left(\frac{k_a(\mathbf{x_{j+1}})}{\hat{k}(\mathbf{x_{j+1}})} \frac{1}{P_{j+1}} s(\mathbf{x_{j+1}}, \boldsymbol{\omega}) \right) + \\ (1 - P_{j+1}) \left(\frac{k_n(\mathbf{x_{j+1}})}{\hat{k}(\mathbf{x_{j+1}})} \frac{1}{1 - P_{j+1}} f(\mathbf{x_{j+1}}, \boldsymbol{\omega}) \right) \Bigg] \end{aligned} \tag{8}$$

and a simple recursive expansion gives

$$f(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{+\infty} p_{\Lambda_0}(\lambda_0) d\lambda_0 \Big[P_1 w_1 + (1 - P_1) I_1 \Big] \tag{9}$$

with

$$I_j = \int_0^{+\infty} p_{\Lambda_j}(\lambda_j) d\lambda_j \Big[P_{j+1} w_{j+1} + (1 - P_{j+1}) I_{j+1} \Big] \tag{10}$$

and

$$w_j = \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})} \frac{1}{P_j} s(\mathbf{x_j}, \boldsymbol{\omega}) \prod_{m=1}^{j-1} \left(\frac{k_n(\mathbf{x_m})}{\hat{k}(\mathbf{x_m})} \frac{1}{1 - P_m} \right) \tag{11}$$

Eq. 10 and 11 lead to the equation of Fig. 2 in a straightforward manner as soon as the choice $P_j = \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})}$ is made. This is obviously only possible if $k_n > 0$, i.e. $k_a(\mathbf{x_j}) < \hat{k}(\mathbf{x_j})$, which insures $P_j < 1$ and $1 - P_j > 0$. The usual restriction to positive null-collisions is therefore very much meaningful. However, the fact that $\hat{k}$ must be a strict upper bound of the extinction coefficient k in standard null-collision algorithms is often a severe limitation of the technique. $\hat{k}$ has to be chosen as a compromise between approaching k closely enough to avoid numerous expensive iterative null-collisions, and preserving enough simplicity to allow fast free paths sampling procedures. From this point of view, the constraint that $\hat{k}$ must be strictly greater than k at all locations is a severe constraint. This is particularly true when the optical properties cannot be pre-computed across the field and are only evaluated at each collision location, once it is sampled. This is a typical requirement of meshless algorithms. In such cases, there is no fundamental problem associated to the construction of a nonstrict upper-bound of k, for instance by only pre-computing k on a rough grid across the field, but it is very difficult to impose that this upper bound is strict considering that absorption and scattering coefficients are commonly non-monotonous functions of pressure, temperature and concentrations.

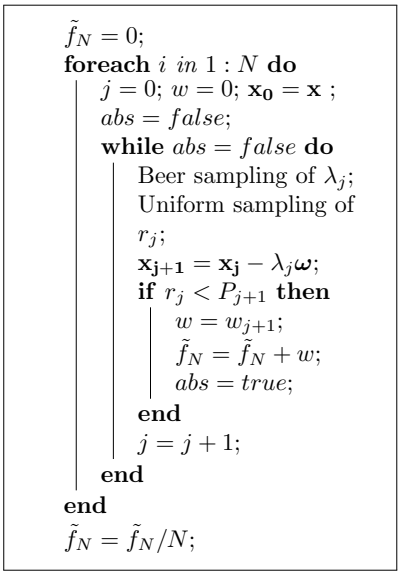
This difficulty can however be bypassed as soon as one observes that the choice of P_j in Eqs. 9 - 11 is not constrained: $P_j = \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})}$ is systematically used in the literature only because of its intuitive nature, in relation to the kinetic pictures of null-collisions. An alternative knowledgeable choice is:

$$P_j = \frac{k_a(\mathbf{x_j})}{k_a(\mathbf{x_j}) + |\hat{k}(\mathbf{x_j}) - k_a(\mathbf{x_j})|} \tag{12}$$

The immediate benefit is that we get rid of the constraint $k_n > 0$ (i.e. $\hat{k} = k_a + k_n$ is an upper bound of k_a): *negative values of the null-collision coefficient are now admitted*. Furthermore, this choice is consistent with the results presented above since using P_j of Eq. 12 leads to:

- the very same algorithm in cases when $\hat{k}$ is a strict upper bound of k_a
- a legible extension of the algorithm otherwise, which bypasses the difficulties encountered when $k_a > \hat{k}$.

The resulting algorithm is fully described in Fig.3 and its extension to multiple scattering in confined



$$f(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{+\infty} p_{\Lambda_0}(\lambda_0) d\lambda_0 \int_0^1 p_{R_0}(r_0) dr_0$$
$$\Big\{ \mathcal{H}\Big(r_0 < P_1\Big) w_1 + \mathcal{H}\Big(r_0 > P_1\Big) \int_0^{+\infty} p_{\Lambda_1}(\lambda_1) d\lambda_1 \int_0^1 p_{R_1}(r_1) dr_1$$
$$\Big\{ \mathcal{H}\Big(r_1 < P_2\Big) w_2 + \mathcal{H}\Big(r_1 > P_2\Big) \int_0^{+\infty} p_{\Lambda_2}(\lambda_2) d\lambda_2 \int_0^1 p_{R_2}(r_2) dr_2$$
$$\cdots \Big\} \Big\}$$

with

$$\mathbf{x_0} = \mathbf{x} \; ; \; \mathbf{x_{j+1}} = \mathbf{x_j} - \lambda_j \boldsymbol{\omega}$$
$$p_{\Lambda_j}(\lambda_j) = \hat{k}(\mathbf{x_j} - \lambda_j \boldsymbol{\omega}) exp\Big(- \int_0^{\lambda_j} \hat{k}(\mathbf{x_j} - \sigma_j \boldsymbol{\omega}) d\sigma_j \Big)$$
$$p_{R_j}(r_j) = 1$$
$$P_j = \frac{k_a(\mathbf{x_j})}{k_a(\mathbf{x_j}) + |\hat{k}(\mathbf{x_j}) - k_a(\mathbf{x_j})|}$$
$$w_j = \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})} \frac{1}{P_j} s(\mathbf{x_j}, \boldsymbol{\omega}) \prod_{m=1}^{j-1} \Big(\frac{k_n(\mathbf{x_m})}{\hat{k}(\mathbf{x_m})} \frac{1}{1 - P_m} \Big)$$

Figure 3: The generalized null-collision algorithm in which there is no more constraint on the $\hat{k}$ field. $\tilde{f}_N$ is a Monte Carlo estimate of $f(\mathbf{x}, \boldsymbol{\omega})$ justified by Eq. 9. The integral formulation displayed on the right side of the algorithm box is a strict formal translation of the algorithm description. Note that when k_n is always positive, $P_j = \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})}$, $\frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})} \frac{1}{P_j} = 1$ and $\frac{k_n(\mathbf{x_m})}{\hat{k}(\mathbf{x_m})} \frac{1}{1 - P_m} = 1$; the algorithm becomes identical to that of Fig. 2.

geometries is provided in the following section. One of its important features is that the Monte Carlo weight of Eq. 11 can take negative values : $\frac{k_n}{\hat{k}}$ is negative each time k_n is negative. So the proposed algorithm deals rigorously with the occurrence of unexpected negative values of the null-collision coefficient, but this is achieved at the price of increasing the weight-variance, therefore lowering the convergence rate. This is quantitatively examined in the following section.

3 Practical implementation

The algorithm described in this section evaluates the stationary net-power density $A(\mathbf{x})$ at a location $\mathbf{x}$ within the volume, i.e. the balance between the radiative power absorbed and the radiative power

emitted locally, per unit volume:

$$A(\mathbf{x}) = \int_{4\pi} h\nu c k_a(\mathbf{x}) \left[f(\mathbf{x}, \boldsymbol{\omega}) - s(\mathbf{x}, \boldsymbol{\omega}) \right] d\boldsymbol{\omega} \tag{13}$$

We restrict ourselves to thermal emission under the assumption of local thermodynamic equilibrium. Therefore $s(\mathbf{x}, \boldsymbol{\omega}) = f^{eq}(\mathbf{x})$ and

$$A(\mathbf{x}) = \int_{4\pi} h\nu c k_a(\mathbf{x}) \left[f(\mathbf{x}, \boldsymbol{\omega}) - f^{eq}(\mathbf{x}) \right] d\boldsymbol{\omega} \tag{14}$$

If the volume were still non-scattering and infinite as in Sec. 2, $A(\mathbf{x})$ could be evaluated using an algorithm very similar to that of Eqs. 9, 10 and 11 (see also Fig. 3). The only change would be that $\boldsymbol{\omega}$ would be first sampled according to an isotropic probability density function $p_{\boldsymbol{\Omega}}(\boldsymbol{\omega}) = \frac{1}{4\pi}$, and the Monte Carlo weight w_j would be modified by multiplying it by $4\pi h\nu c k_a(\mathbf{x})$ and replacing $f^{eq}(\mathbf{x_j})$ by $f^{eq}(\mathbf{x_j}) - f^{eq}(\mathbf{x})$. Equations 9, 10 and 11 would then become

$$A(\mathbf{x}) = \int_{4\pi} p_{\boldsymbol{\Omega}}(\boldsymbol{\omega}) d\boldsymbol{\omega} \int_0^{+\infty} p_{\Lambda_0}(\lambda_0) d\lambda_0 \Big[P_1 w_1 + (1 - P_1) I_1 \Big] \tag{15}$$

$$I_j = \int_0^{+\infty} p_{\Lambda_j}(\lambda_j) d\lambda_j \Big[P_{j+1} w_{j+1} + (1 - P_{j+1}) I_{j+1} \Big] \tag{16}$$

$$w_j = 4\pi h\nu c k_a(\mathbf{x}) \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})} \frac{1}{P_j} \left(f^{eq}(\mathbf{x_j}) - f^{eq}(\mathbf{x}) \right) \prod_{m=1}^{j-1} \left(\frac{k_n(\mathbf{x_m})}{\hat{k}(\mathbf{x_m})} \frac{1}{1 - P_m} \right) \tag{17}$$

Introducing multiple scattering can be performed by adding a branch to the collision test, and sampling a new direction when true scattering occurs. When dealing with opaque boundaries a test is added to check if a boundary is intersected before the next collision, in which case a new binary sampling procedure is implemented to either resume the algorithm, with a new sampled reflection direction, or stop the algorithm and compute the Monte Carlo weight using the value of the equilibrium distribution function at the surface impact. Altogether, the resulting algorithm is a quite standard backward Monte Carlo algorithm corresponding to the following recursive formulation:

$$\begin{aligned} A(\mathbf{x}) = & \int_{4\pi} p_{\boldsymbol{\Omega}}(\boldsymbol{\omega}) d\boldsymbol{\omega} \int_0^{+\infty} p_{\Lambda_0}(\lambda_0) d\lambda_0 \times \\ & \left\{ \begin{array}{l} \mathcal{H}(\mathbf{x_1} \in \mathcal{B}) \left\{ \begin{array}{l} P_{E,1} w_1 \\ + (1 - P_{E,1}) \int_{2\pi} p_R(\boldsymbol{\omega_0} | \boldsymbol{\omega_1}, \mathbf{x_1}) d\boldsymbol{\omega_1} I_1 \end{array} \right\} + \\ \mathcal{H}(\mathbf{x_1} \in \mathcal{V}) \left\{ \begin{array}{l} P_{A,1} w_1 \\ + P_{S,1} \int_{4\pi} p_S(\boldsymbol{\omega_0} | \boldsymbol{\omega_1}, \mathbf{x_1}) d\boldsymbol{\omega_1} I_1 \\ + P_{N,1} \int_{4\pi} \delta(\boldsymbol{\omega_0} - \boldsymbol{\omega_1}, \mathbf{x_1}) d\boldsymbol{\omega_1} I_1 \end{array} \right\} \end{array} \right. \end{aligned} \tag{18}$$

$$I_j = \int_0^{+\infty} p_{\Lambda_j}(\lambda_j) d\lambda_j \times$$
$$\left\{ \begin{array}{l} \mathcal{H}(\mathbf{x_{j+1}} \in \mathcal{B}) \left\{ \begin{array}{l} P_{E,j+1} w_{j+1} \\ + (1 - P_{E,j+1}) \int_{2\pi} p_R(\boldsymbol{\omega_j} | \boldsymbol{\omega_{j+1}}, \mathbf{x_{j+1}}) d\boldsymbol{\omega_{j+1}} I_{j+1} \end{array} \right\} + \\ \\ \mathcal{H}(\mathbf{x_{j+1}} \in \mathcal{V}) \left\{ \begin{array}{l} P_{A,j+1} w_{j+1} \\ + P_{S,j+1} \int_{4\pi} p_S(\boldsymbol{\omega_j} | \boldsymbol{\omega_{j+1}}, \mathbf{x_{j+1}}) d\boldsymbol{\omega_{j+1}} I_{j+1} \\ + P_{N,j+1} \int_{4\pi} \delta(\boldsymbol{\omega_j} - \boldsymbol{\omega_{j+1}}, \mathbf{x_{j+1}}) d\boldsymbol{\omega_{j+1}} I_{j+1} \end{array} \right. \end{array} \right. \tag{19}$$

$$w_j = 4\pi h\nu c k_a(\mathbf{x}) \times$$
$$\left[\mathcal{H}(\gamma_j{=}1) \frac{\varepsilon(\mathbf{x_j}, \boldsymbol{\omega_{m-1}})}{P_{E,j}} \left(f^{eq}(\mathbf{x_j}) - f^{eq}(\mathbf{x}) \right) + \mathcal{H}(\gamma_j{=}3) \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j}) P_{A,j}} \left(f^{eq}(\mathbf{x_j}) - f^{eq}(\mathbf{x}) \right) \right] \times \tag{20}$$
$$\prod_{m=1}^{j-1} \left[\mathcal{H}(\gamma_m{=}2) \frac{1 - \varepsilon(\mathbf{x_m}, \boldsymbol{\omega_{m-1}})}{1 - P_{E,m}} + \mathcal{H}(\gamma_m{=}4) \frac{k_s(\mathbf{x_m})}{\hat{k}(\mathbf{x_m}) P_{S,m}} + \mathcal{H}(\gamma_m{=}5) \frac{k_n(\mathbf{x_m})}{\hat{k}(\mathbf{x_m}) P_{N,j}} \right]$$

where $\mathcal{V}$ is the volume of the considered system and $\mathcal{B}$ its boundary (Fig. 4). The locations $\mathbf{x_{j+1}}$ and directions $\boldsymbol{\omega_j}$ are defined in the same way as in section 2 with the only difference that $\mathbf{x_{j+1}} = \mathbf{y_{j+1}}$ when $\mathbf{x_j} - \lambda \boldsymbol{\omega_j}$ is outside $\mathcal{V}$, where $\mathbf{y_{j+1}}$ is the intersection with the boundary of the straight ray starting at $\mathbf{x_j}$ in the direction $-\boldsymbol{\omega_j}$ (see Fig. 4). When $\mathbf{x_j}$ belongs to $\mathcal{B}$, $\varepsilon(\mathbf{x_j}, \boldsymbol{\omega_{j-1}})$ is the local value of the emissivity in the direction $\boldsymbol{\omega_{j-1}}$, and $p_R(\boldsymbol{\omega_{j-1}} | \boldsymbol{\omega_j}, \mathbf{x_j})$ is the probability density of the reflection direction $\boldsymbol{\omega_{j-1}}$ for an incidence along $\boldsymbol{\omega_j}$. In the absence of any specific convergence difficulty, the branching probability $P_{E,j}$ (the probability that the algorithm stops at the surface impact $\mathbf{x_j}$) can be taken as $P_{E,j} = \varepsilon(\mathbf{x_j}, \boldsymbol{\omega_{j-1}})$. In the expression of the weight, $\gamma_j = 1$ if the algorithm stops at the boundary, $\gamma_j = 2$ if the optical path sampling is continued backward after surface reflection, $\gamma_j = 3$ in case of "absorption" within the volume, $\gamma_j = 4$ in case of true scattering and $\gamma_j = 5$ in case of null-collision. The true originalities are the definition of the branching probabilities $P_{A,j}$, $P_{S,j}$ and $P_{N,j}$ when $\mathbf{x_j}$ belongs to $\mathcal{V}$ (probabilities that the j-th collision is an absorption, a true-scattering event, or a null-collision respectively), as well as the Monte Carlo weight expressions. As argued in Sec. 2, we suggest the use of $P_{A,j} = \frac{k_a(\mathbf{x_j})}{k_a(\mathbf{x_j}) + k_s(\mathbf{x_j}) + |k_n(\mathbf{x_j})|}$, $P_{S,j} = \frac{k_s(\mathbf{x_j})}{k_a(\mathbf{x_j}) + k_s(\mathbf{x_j}) + |k_n(\mathbf{x_j})|}$ and $P_{N,j} = \frac{|k_n(\mathbf{x_j})|}{k_a(\mathbf{x_j}) + k_s(\mathbf{x_j}) + |k_n(\mathbf{x_j})|}$. Except for that, the algorithmic structure strictly corresponds to the application of Skullerud and Woodcock's strategies. Note however that although we essentially play with probability choices, our proposition is nothing like an importance sampling strategy. As detailed at the end of Sec. 2, we do not propose to modify the branching probabilities and change the Monte-Carlo weight accordingly; we rather extend the applicability range of standard null-collision algorithms by preserving exactly the definitions of $P_{A,j}$, $P_{S,j}$ and $P_{N,j}$ in the usual range, and generalizing their definitions in order to handle rigorously the occurrences of $\hat{k} < k_a$.

We now present a parametric study in order to evaluate the numerical behaviour of the above presented algorithm. Monochromatic radiative budget densities are evaluated at two locations within a simple academic configuration. The algorithmic implementation is validated against a well mastered Monte Carlo algorithm, and the code is then used to analyse how the convergence levels and the computation times depend on the retained $\hat{k}$ field. The considered system is a cube, of side $2L$, with $0K$ diffuse-reflecting faces of uniform emissivity ε, that are perpendicular to the x, y and z axis of a Cartesian coordinate system originating at the center of the cube (see Fig. 5). The

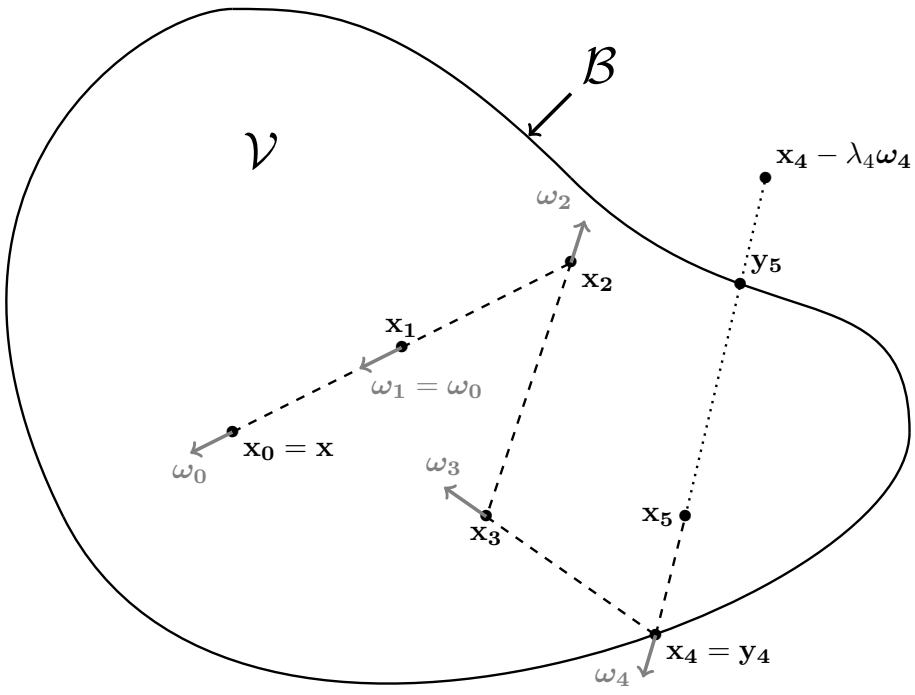


Figure 4: $\mathbf{y_{j+1}}$ is the intersection with the boundary of the straight ray starting at $\mathbf{x_j}$ in the direction $-\boldsymbol{\omega_j}$. $\mathbf{x_{j+1}}$ equals $\mathbf{x_j} - \lambda_j\boldsymbol{\omega_j}$ if this location belongs to $\mathcal{V}$. Otherwise $\mathbf{x_{j+1}} = \mathbf{y_{j+1}}$. If $\mathbf{x_{j+1}} \in \mathcal{V}$ the collision is either a null-collision and $\boldsymbol{\omega_{j+1}} = \boldsymbol{\omega_j}$ (see $j = 0$ in the figure), or a true scattering and $\boldsymbol{\omega_{j+1}}$ is sampled according to the single scattering phase function (see $j = 1$ and $j = 2$ in the figure), or an "absorption" and the algorithm stops (the exchange weight is computed, see $j = 4$ in the figure). If $\mathbf{x_{j+1}} \in \mathcal{B}$ the interaction with the boundary is either a reflection and $\boldsymbol{\omega_{j+1}}$ is sampled according to the directional reflectivity (see $j = 3$ in the figure), or an "absorption" and the algorithm stops (the exchange weight is computed).

enclosed medium is heterogeneous both in temperature and optical properties. The k_a, k_s and f^{eq}

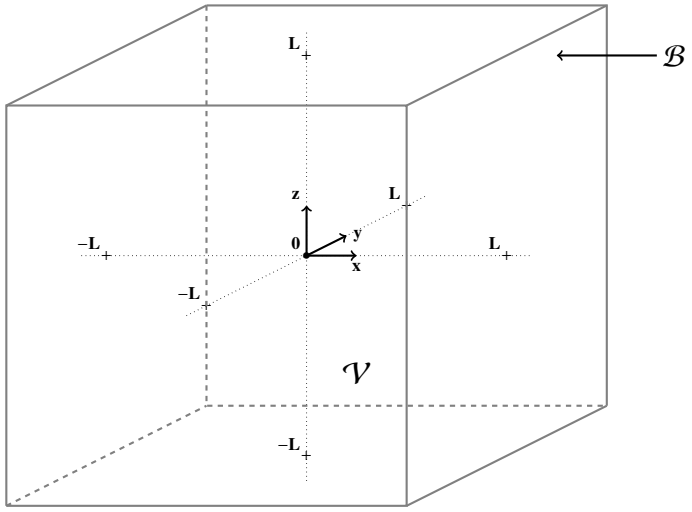


Figure 5: Considered system: a cube of side 2L, whose center is the Cartesian coordinate system origin.

fields are

$$k_a(x,y,z) = k_{a,max}\left(\frac{L-x}{2L}\right)\left(1-\sqrt{\frac{y^2+z^2}{2L^2}}\right), \tag{21}$$

$$k_s(x,y,z) = k_s(x,y,z) = k_{s,max}\left(\frac{L-x}{2L}\right)\left(1-\sqrt{\frac{y^2+z^2}{2L^2}}\right) \tag{22}$$

and

$$f^{eq}(x,y,z) = f^{eq}_{max}\left(\frac{L-x}{2L}\right)\left(1-\sqrt{\frac{y^2+z^2}{2L^2}}\right) \tag{23}$$

figuring an axisymmetric flame along the x axis (maximum temperature and maximum extinction along the axis, and a linear decay as function of the distance to the axis, down to zero at the corners). The Henyey-Greenstein single-scattering phase function is used with a uniform value of the asymmetry parameter g throughout the field. For simplicity, $\hat{k}$ is chosen uniform. As k_a and k_s take their maximum values at the same location, $k_{max} = k_{a,max} + k_{s,max}$ is the maximum value of the total extinction coefficient and the ratio $\rho = \frac{\hat{k}}{k_{max}}$ tells us whether negative values of the null-collision coefficient will occur ($\rho < 1$) or not. Because of the shape of the retained field of equilibrium distribution function, monochromatic radiative budgets are simply proportional to f^{eq}_{max} and the remaining numerically-meaningful free-parameters are (in nondimensional form) : ρ, $k_{a,max}L$, $k_{s,max}L$, g and ε. The analysis will be performed using $g = 0$ (isotropic scattering) and $\varepsilon = 1$ (black boundaries). The influence of g, ε, as far as numerical behaviour is concerned, will then be briefly described at the end of the section.

Table 1 displays the simulated values of $A(\mathbf{x})$ for $\mathbf{x} = [0,0,0]$ (the center of the cube) and $\mathbf{x} = [-L,0,0]$ (the location of the maximum values of the k_a, k_s and f^{eq} fields), using 10^6 independant

realizations, for $\rho = 1$, meaning that $k_n = 0$ at $\mathbf{x} = [-L, 0, 0]$ and $k_n > 0$ at all other locations (no negative values of the null-collision coefficient). Also given are the associated standard deviations, σ, and computation times, t. The columns labelled A_{ref} and σ_{ref} correspond to the simulation results obtained with a standard Monte Carlo algorithm in which the problem of inverting optical thicknesses is solved by fitting $k = k_a + k_s$ using an accurate spline decomposition. These solutions were only used to validate the implementation procedure : considering the values of σ and σ_{ref}, A and A_{ref} are indeed statistically compatible. The relative uncertainty $\frac{\sigma}{A}$ indicates that the convergence level is good for all the considered absorption and scattering optical thicknesses ($\frac{\sigma}{A}$ is below 0.2% in all cases). The computation times, that were measured without the use of any parallelization procedure, are typical of standard Monte Carlo simulations.

More open is the question of choosing $\hat{k}$, in particular the effect of modifying the Monte Carlo weight in order to deal with negative values of the null-collision coefficient when $\hat{k} < k$ at some locations. This question is addressed by reproducing the same simulations for different values of ρ, from $\rho = 0.5$ (i.e. $\hat{k}$ is a faulty overestimate of k, as low as $\frac{1}{2}k$ at some locations) to $\rho = 5$ (on the contrary $\hat{k}$ is a large overestimate of k). Figure 6 displays the evolution with ρ of $\frac{\sigma}{A}$, Fig. 7 displays the computation times, and Fig. 8 displays the computation times required to achieved a 1% accuracy. These results are interpreted as follows:

- Above $\rho = 1$, the standard deviation of the estimator is independant of ρ. This is expected since no negative values of the null-collision coefficient occur : as indicated from the start, standard null-collision algorithms can be rigorously interpreted as only practical ways to sample collision-locations according to Beer extinction. Adding supplementary null-collisions increases only the computation time but changes nothing to the resulting sampling statistics.

- Below $\rho = 1$, the standard deviation of the estimator increases when increasing the occurrence of negative values of the null-collision coefficient. Again, this is expected since the handling of negative values of the null-collision coefficient is achieved at the price of multiplying the Monte Carlo weight by the correction term[3] $\frac{+}{-}\frac{\hat{k}+2|k_n|}{\hat{k}}$. The module of this weight-correction factor is always greater than unity and the factor is positive when absorption or true scattering is retained, negative when null-collision is retained. If many scattering or null-collision events occur along the optical path, in regions where $k_n < 0$, before the algorithm stops because of absorption, then the Monte Carlo weight can take very high absolute values as it involves the product of a large number or correction terms greater than unity. The convergence toward the exact same solution of the radiative transfer equation is insured by the fact that positive weights are compensated by negative ones, but the convergence rate is smaller : much more statistical realisations are required to reach the same accuracy levels when no negative values of the null-collision coefficient occur. This is illustrated by the fact that for increasing values of the scattering optical thickness combined with high values of the single-scattering albedo (see $k_s L = 3$ and $k_a L = 0.1$ in Fig. 6), the standard deviation increases very fast when decreasing ρ below unity. This effect is of course much stronger when $\mathbf{x}$ is right at the center of the region where $k_n < 0$ (see $\mathbf{x} = [-L, 0, 0]$) than when optical paths starts from a region where $k_n > 0$ (see $\mathbf{x} = [0, 0, 0]$).

- For a given number of statistical realisations, the computation times (see Fig. 7) decrease when decreasing the number of null-collisions, and this is also true when decreasing k_n below

[3] With the choices we made for P_A, P_S and P_N, the correction terms in the weight expression of Eq. 17 verify the property $\frac{k_a}{\hat{k}P_A} = \frac{k_s}{\hat{k}P_S} = \left|\frac{k_n}{\hat{k}P_N}\right| = \frac{\hat{k}+2|k_n|}{\hat{k}}$

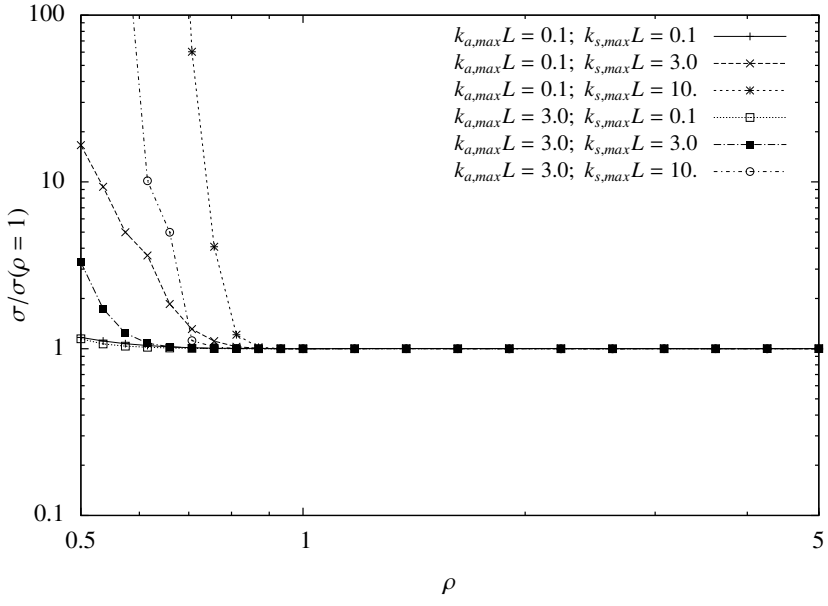
$k_{a,max}L$	$k_{s,max}L$	$\frac{A}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$	$\frac{\sigma}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$	t(s)	$\frac{A_{ref}}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$	$\frac{\sigma_{ref}}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$
0.1	0.1	-0.483813	8.52E-05	2.43	-0.483717	1.13E-05
0.1	1	-0.482031	8.97E-05	7.92	-0.481921	1.40E-05
0.1	3	-0.477997	9.90E-05	24.25	-0.477883	1.93E-05
0.1	10	-0.463027	1.27E-04	122.69	-0.463068	3.56E-05
1	0.1	-0.366086	2.09E-04	2.94	-0.365971	7.96E-05
1	1	-0.356169	2.13E-04	7.43	-0.356353	8.93E-05
1	3	-0.33585	2.20E-04	19.2	-0.335928	1.06E-04
1	10	-0.277205	2.28E-04	76.39	-0.27683	1.34E-04
3	0.1	-0.218989	2.21E-04	3.48	-0.218942	1.23E-04
3	1	-0.209261	2.18E-04	6.4	-0.209529	1.26E-04
3	3	-0.190256	2.10E-04	13.63	-0.190141	1.30E-04
3	10	-0.144073	1.84E-04	41.38	-0.143501	1.27E-04
10	0.1	-0.071271	1.19E-04	3.49	-0.07137	9.15E-05
10	1	-0.068662	1.15E-04	4.66	-0.068854	8.99E-05
10	3	-0.063501	1.07E-04	7.29	-0.063369	8.61E-05
10	10	-0.050674	8.49E-05	16.23	-0.050674	7.44E-05

(a)

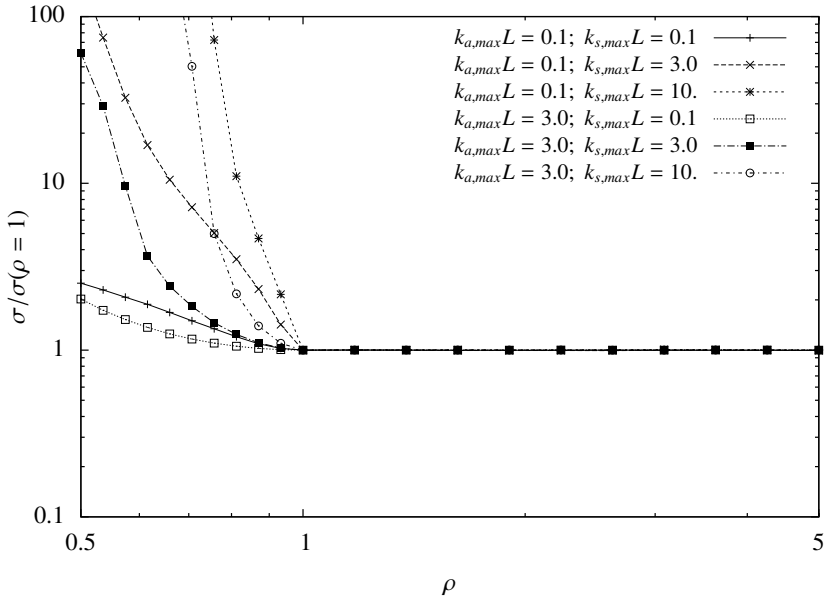
$k_{a,max}L$	$k_{s,max}L$	$\frac{A}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$	$\frac{\sigma}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$	t(s)	$\frac{A_{ref}}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$	$\frac{\sigma_{ref}}{4\pi k_a(\mathbf{x_0})f^{eq}_{max}}$
0.1	0.1	-0.977296	1.27E-04	2.24	-0.977336	2.58E-05
0.1	1	-0.97683	1.29E-04	6.18	-0.976679	2.79E-05
0.1	3	-0.975682	1.33E-04	15.3	-0.975767	3.22E-05
0.1	10	-0.974828	1.37E-04	44.9	-0.974733	4.36E-05
1	0.1	-0.822495	3.24E-04	2.38	-0.822111	1.97E-04
1	1	-0.822446	3.26E-04	5.13	-0.821846	2.03E-04
1	3	-0.823933	3.29E-04	10.75	-0.823994	2.14E-04
1	10	-0.83941	3.27E-04	26.32	-0.839533	2.29E-04
3	0.1	-0.658358	4.07E-04	2.22	-0.657242	3.64E-04
3	1	-0.66479	4.09E-04	3.73	-0.664704	3.62E-04
3	3	-0.67959	4.12E-04	6.67	-0.679703	3.58E-04
3	10	-0.72422	4.10E-04	14.49	-0.722886	3.42E-04
10	0.1	-0.544282	4.62E-04	1.98	-0.5438	4.60E-04
10	1	-0.551703	4.63E-04	2.47	-0.551153	4.57E-04
10	3	-0.567704	4.65E-04	3.54	-0.567366	4.48E-04
10	10	-0.61077	4.65E-04	6.76	-0.609865	4.27E-04

(b)

Table 1: Estimation, standard deviation and computation time obtained for 10^6 independant realizations and for $\rho = 1$ at two probe locations: $\mathbf{x_0} = [0, 0, 0]$ (see table (a)) and $\mathbf{x_0} = [-L, 0, 0]$ (see table (b)) for several values of the optical thicknesses $k_{a,max}L$ and $k_{s,max}L$. The computation was done with a processor "Intel Core i5 - 2,4GHz" without any parallelization.

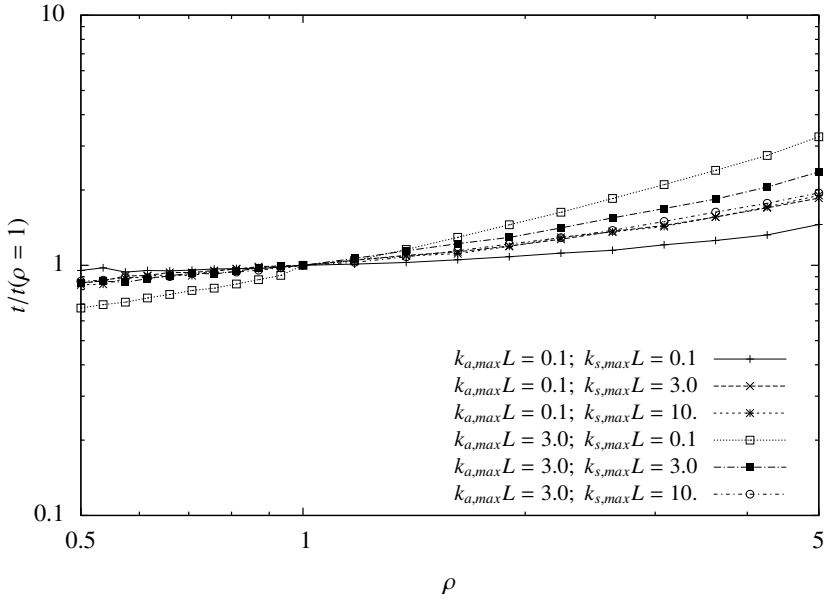


(a)

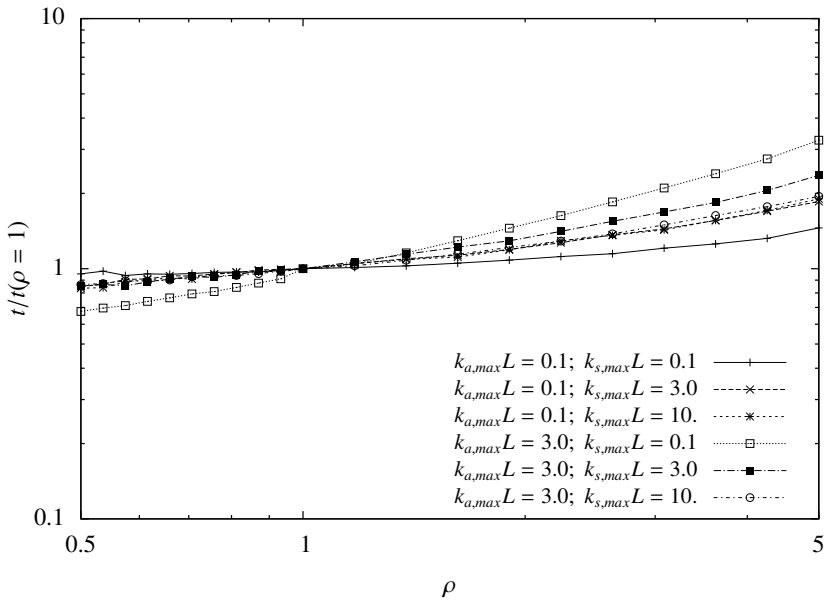


(b)

Figure 6: Standard deviation as a function of ρ, $k_{a,max}L$, $k_{s,max}L$ at (a) $\mathbf{x_0} = [0, 0, 0]$ and (b) $\mathbf{x_0} = [-L, 0, 0]$

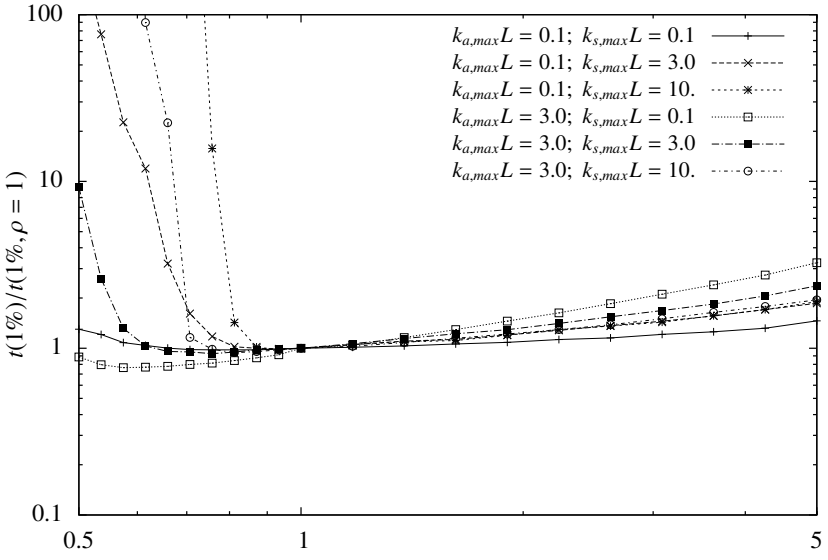


(a)

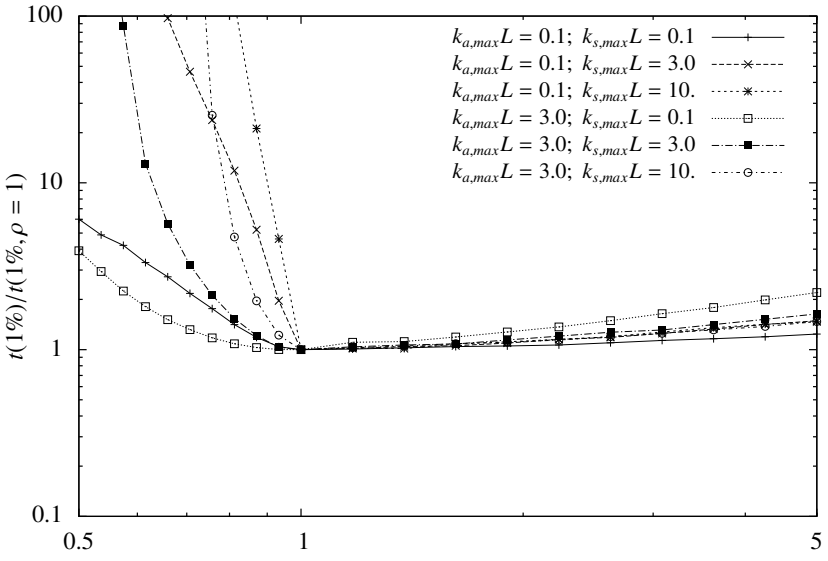


(b)

Figure 7: Computation time as a function of ρ, $k_{a,max}L$, $k_{s,max}L$ at (a) $\mathbf{x_0} = [0, 0, 0]$ and (b) $\mathbf{x_0} = [-L, 0, 0]$



(a)



(b)

Figure 8: Computation time in order to reach a 1% standard deviation as a function of ρ, $k_{a,max}L$, $k_{s,max}L$ at (a) $\mathbf{x_0} = [0, 0, 0]$ and (b) $\mathbf{x_0} = [-L, 0, 0]$

zero. This is a direct result of less collisions occurring, but this does not wholly compensate the degradation in standard deviation (see Fig. 8). For a given relative accuracy, the required computation time is then driven by the impact of ρ upon the standard deviation, and it is of course greater as $\hat{k}$ becomes a larger and larger overestimate of the true extinction coefficient.

Altogether, the use of negative values of the null-collision coefficient is fully relevant when the approximated upper-bound $\hat{k}$ can be astutely designed, since the convergence will be really reasonable: for $\hat{k} \simeq 0.9k$, the increase of the computing effort should not be a concern (see Fig. 8 (a)) except if domains where $k_n < 0$ are optically thick with a high single scattering albedo (see Fig. 8 (b)). Accordingly, most efforts design of $\hat{k}$ should focus on avoiding the occurrences of such domains. Bad approximates of the upper-bound ($\rho << 1$) would yield pathological behaviours, as expected.

The simulations performed with $g \neq 0$ and $\varepsilon < 1$ indicate that the shape of the single scattering phase function has very little influence (the values of A are affected but the numerical behaviour is unchanged), and that surface reflection acts like scattering : because of multiple reflections, more null-collision or scattering events can occur within the domain of negative null-collision coefficients before absorption and standard deviation increases (although less than when increasing scattering).

4 Formal developments

This section is addressed to the reader interested by the formal significance of null-collision algorithms. The physical meaning of null-collisions at the kinetic level is quite trivial : they are additional collisions that change nothing to the overall radiative transfer. But when looking at the corresponding integral formulations, several observations can be made, that could be useful in the process of enhancing statistical convergence. A renewed viewpoint can indeed be taken from which null-collisions are only of secondary importance compared to the associated integral reformulation. This reformulation alone suppresses the need for an optical-thickness inversion procedure and meshless algorithms can therefore be designed without introducing any null-collision. The next paragraph, entitled *step 1*, illustrates this point. In *step 2* we argue that it may still be useful to introduce a (non-strict) overestimate $\hat{k}$ of the extinction coefficient, but $\hat{k}$ is not used for sampling collision locations: it plays a role similar to that of a control variate [7], allowing to get rid of sign alternations that would otherwise be sources of convergence difficulties. In *step 3* we finally show how standard null-collision algorithms can be fully recovered by choosing to also make use of $\hat{k}$ for free-path sampling as well as for the weighting of branching tests. We advise however that this choice does not entail optimized convergence features.

Step 1 - Our starting point is the observation that the initial radiative transfer equation of Eq. 1 at stationary state can be integrated backward along the line of sight to give the following Fredholm equation:

$$f(\mathbf{x}, \boldsymbol{\omega}) = f(\mathbf{x} - L\boldsymbol{\omega}, \boldsymbol{\omega}) + \int_0^L \left[k_{a,\lambda} s_\lambda - k_{a,\lambda} f(\mathbf{x} - \lambda\boldsymbol{\omega}, \boldsymbol{\omega})\right] d\lambda \tag{24}$$

This equation is easy to demonstrate but its structure does not highlight the pictures of transport physics, which is probably the reason why it is seldom mentioned in the radiative transfer literature. Indeed, by comparison with Eq. 3, no Beer extinction appears and it is difficult to interpret physically the integration over space of the local emission $k_{a,\lambda} s_\lambda$. Of course, the exponentials are well recovered due to the Fredholm structure of this equation (f appearing within the integral).

Fredholm equations are common in photon transport physics but it is worth mentioning that they are usually the result of scattering or surface-reflection representations. In the present context the fact that Beer extinction does not appear explicitly is a strong advantage: the difficulties associated with the inversion of exponential extinctions in heterogeneous media are automatically by-passed. Let us consider the particular case where $f(\mathbf{x} - L\boldsymbol{\omega}, \boldsymbol{\omega}) = 0$. The same steps can then be followed as in Eq. 6 and 8, starting from Eq. 24 instead of Eq. 5, to give:

$$f(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{L_0} p_{\Lambda_0}(\lambda_0) d\lambda_0 \Big[P_1 w_1 + (1 - P_1) I_1 \Big] \tag{25}$$

with

$$I_j = \int_0^{L_j} p_{\Lambda_j}(\lambda_j) d\lambda_j \Big[P_{j+1} w_{j+1} + (1 - P_{j+1}) I_{j+1} \Big] \tag{26}$$

where the only changes by comparison with Eq. 9 and 10 are that the j-th free path is integrated between zero and $L_j = L - \sum_{m=0}^{j-1} \lambda_j$, the probability density function $p_{\Lambda_j}(\lambda_j)$ is now an arbitrary probability density on $[0, L_j]$, and the Monte Carlo weights are

$$w_j = k_a(\mathbf{x_j}) s(\mathbf{x_j}, \boldsymbol{\omega}) \frac{1}{P_j} \frac{1}{p_{\Lambda_{j-1}}(\lambda_{j-1})} \prod_{m=1}^{j-1} \left[-k_a(\mathbf{x_m}) \frac{1}{1 - P_m} \frac{1}{p_{\Lambda_{m-1}}(\lambda_{m-1})} \right] \tag{27}$$

Apart from the free paths being integrated over finite intervals, which we will comment later, the essential differences with the null-collision algorithm of section 2 are that no $\hat{k}$ field has yet been introduced and that the successive weights alternate signs ($w_1 > 0$; $w_2 < 0$; ...). In *step 2* we argue that *the first meaning and the main interest of introducing $\hat{k}$ is to break this sign alternation.*

Step 2 - As detailed in the literature about exponential transforms [8, 9, 10], it is shown in B that any arbitrary positive scalar field $\hat{k}$ can be introduced to transform Eq. 24 into

$$\begin{aligned} f(\mathbf{x}, \boldsymbol{\omega}) = f(\mathbf{x} - L\boldsymbol{\omega}, \boldsymbol{\omega}) exp\left(- \int_0^L \hat{k}_\lambda d\lambda \right) + \\ \int_0^L exp\left(- \int_0^\lambda d\sigma \hat{k}_\sigma \right) \Big[k_{a,\lambda} s_\lambda + \left(\hat{k}_\lambda - k_{a,\lambda} \right) f(\mathbf{x} - \lambda\boldsymbol{\omega}, \boldsymbol{\omega}) \Big] d\lambda \end{aligned} \tag{28}$$

Very much like when introducing control variates to modify the convergence features of Monte Carlo algorithms [7], we can play with the arbitrary choice of the $\hat{k}$ field:

- First, if $\hat{k} > 0$ the exponentials insure that improper integrals converge and L may be extended to infinity to recover the same problem as in Sec. 2: evaluating $f(\mathbf{x}, \boldsymbol{\omega})$ in the particular case of an infinite medium. Eq. 28 becomes

$$f(\mathbf{x}, \boldsymbol{\omega}) = \int_0^{+\infty} exp\left(- \int_0^\lambda \hat{k}_\sigma d\sigma \right) \Big[k_{a,\lambda} s_\lambda + (\hat{k}_\lambda - k_{a,\lambda}) f(\mathbf{x} - \lambda\boldsymbol{\omega}, \boldsymbol{\omega}) \Big] d\lambda \tag{29}$$

  which is Eq. 5 exactly, where the Dirac integration is solved (there is indeed no more need to highlight the physical picture of a forward scattering equivalent). Note that we only take the

limit $L \to +\infty$ for didactic reasons and that all further reasoning can be reproduced using Eq. 28 to address the question of evaluating $f(\mathbf{x}, \boldsymbol{\omega})$ in bounded domains. For instance, the term $f(\mathbf{x} - L\boldsymbol{\omega}, \boldsymbol{\omega}) exp\left(-\int_0^L \hat{k}_\lambda d\lambda\right)$ in Eq. 28 is the one that allows the representation of surface emission and surface reflection in Sec 3.

- Second, as in Sec. 2, $\hat{k}$ can be lower than k_a. But, as much as possible, $\hat{k}$ should still be chosen such that $\hat{k} > k_a$ at most locations. Indeed this ensures that both $k_{a,\lambda} s_\lambda$ and $(\hat{k}_\lambda - k_{a,\lambda}) f(\mathbf{x} - \lambda\boldsymbol{\omega}, \boldsymbol{\omega'})$ in Eq. 29 are positive terms, with the direct consequence that Monte Carlo weights are strictly positive: the convergence difficulties due to sign alternation vanish. The technical steps of Eq. 6 and Eq. 8 can again be taken, this time to recover Eq. 9 and Eq. 10 exactly, with the following new expression for w_j (which is strictly positive if $\hat{k} - k_a > 0$):

$$\begin{aligned} w_j = {} & k_a(\mathbf{x_j}) exp\left(-\int_0^{\lambda_{j-1}} \hat{k}(\mathbf{x_{j-1}} - \sigma\boldsymbol{\omega}) d\sigma\right) s(\mathbf{x_j}, \boldsymbol{\omega}) \frac{1}{P_j} \frac{1}{p_{\Lambda_{j-1}}(\lambda_{j-1})} \times \\ & \prod_{m=1}^{j-1} \left[\left(\hat{k}(\mathbf{x_m}) - k_a(\mathbf{x_m})\right) exp\left(-\int_0^{\lambda_{m-1}} \hat{k}(\mathbf{x_{m-1}} - \sigma\boldsymbol{\omega}) d\sigma\right) \frac{1}{1 - P_m} \frac{1}{p_{\Lambda_{m-1}}(\lambda_{m-1})}\right] \end{aligned} \tag{30}$$

where the $p_{\Lambda_j}(\lambda_j)$ probability densities and the P_j probabilities are now fully arbitrary [7, 11]. Note in particular that $\hat{k}$ appears in the weight expression, but that p_{Λ_j} and P_j can be chosen independently of $\hat{k}$.

- Third, choosing $\hat{k}$ as close to k_a as possible is useful, this time not as far as statistical convergence is concerned, but in terms of computational costs. Let us indeed admit that p_{Λ_j} and P_j could be ideally chosen according to a zero-variance strategy [11, 12, 13, 14]. If we temporary admit that $\hat{k}$ is strictly greater than k_a at all locations, then zero-variance is obtained with

$$\begin{aligned} p_{\Lambda_j}(\lambda_j) = {} & \frac{1}{f(\mathbf{x_j}, \boldsymbol{\omega})} exp\left(-\int_0^{\lambda_j} \hat{k}(\mathbf{x_j} - \sigma\boldsymbol{\omega}) d\sigma\right) \times \\ & \left[k_a(\mathbf{x_{j+1}}) s(\mathbf{x_{j+1}}, \boldsymbol{\omega}) + \left(\hat{k}(\mathbf{x_{j+1}}) - k_a(\mathbf{x_{j+1}})\right) f(\mathbf{x_{j+1}}, \boldsymbol{\omega})\right] \end{aligned} \tag{31}$$

and

$$P_j = \frac{k_a(\mathbf{x}_j) s(\mathbf{x_j}, \boldsymbol{\omega})}{k_a(\mathbf{x_j}) s(\mathbf{x_j}, \boldsymbol{\omega}) + \left(\hat{k}(\mathbf{x_j}) - k_a(\mathbf{x_j})\right) f(\mathbf{x_j}, \boldsymbol{\omega})} \tag{32}$$

(see C). Then only one sample is required to reach the exact solution and the remaining question is the computation cost of the sampling procedure itself. This cost is directly related to the average value of the recursion level: the value of the index j at which the sampling algorithm is exited. This average recursion level is obviously related to the value of P_j: there is ideally no recursion when $P_j = 1$, which is reached when $\hat{k}$ is strictly identical to k_a. Altogether, our conclusions match those of all previous publications: $\hat{k}$ should be greater than k_a and should be as close to k_a as possible. However, we reach these conclusions without any reference to $\hat{k}$ as an extinction coefficient to be used for the sampling of collision locations. So, not only the constraint $\hat{k} > k_a$ becomes non-strict (as illustrated in the previous sections), but

it is also no more required that the function $\hat{\tau}(\lambda) = \int_0^\lambda \hat{k}(\mathbf{x} - \sigma\boldsymbol{\omega}, \boldsymbol{\omega}, t + \frac{\sigma}{c})d\sigma$ be analytically invertible: all we need is that $\hat{\tau}(\lambda)$ be easily evaluated as it appears within the exponentials in the weight expression of Eq. 30.

Step 3 - To recover the standard null-collision algorithm of Sec. 2 (before extension to negative k_n values), it suffices to make the following choice for p_{Λ_j} and P_j (that were arbitrary up to now):

$$p_{\Lambda_j}(\lambda_j) = \hat{k}(\mathbf{x_{j+1}}) exp\left(-\int_0^{\lambda_j} \hat{k}(\mathbf{x_j} - \sigma\boldsymbol{\omega})d\sigma\right) \tag{33}$$

and

$$P_j = \frac{k_a(\mathbf{x_j})}{\hat{k}(\mathbf{x_j})} \tag{34}$$

This choice is well guided by the physical pictures, but nothing motivates this particular choice in terms of statistical convergence. We have indeed already mentioned that the ideally optimized choice (if it was practicable) would be that of Eq. 31 and Eq. 32, but for Eq. 33 and Eq. 34 to match Eq. 31 and Eq. 32, it is required that $f(\mathbf{x_j}, \boldsymbol{\omega}) \approx f(\mathbf{x_{j+1}}, \boldsymbol{\omega}) \approx s(\mathbf{x_j}, \boldsymbol{\omega}) \approx s(\mathbf{x_{j+1}}, \boldsymbol{\omega})$. This is a fair approximation only in the limit of thermodynamic equilibrium and this strongly limits the applicative potential.

5 Conclusions

Altogether, the null-collision concept was revisited, thinking more specifically of radiative transfer applications. The corresponding algorithms introduce no specific convergence difficulty, which is not surprising considering the well known similarities between photon-transport and neutron or electron-transport, the two particle-transport physics that motivated initially the introduction of null-collisions in Monte Carlo path-tracking algorithms.

It was also shown, by two different formal means, how null-collision algorithms provide exact unbiased statistical estimations of the solution of the radiative transfer equation. In both cases (in Sec. 2 and Sec. 4), thanks to its linearity properties, the radiative transfer equation was replaced by a rigorous integral-equivalent. In the first case, the radiative transfer equation included null-collisions from the start; in the second case, null-collisions were introduced at the integral level.

Beside their meaning in terms of algorithmic validation, these integral formulation efforts open two new fields of investigation. We first showed how null-collision algorithms can be slightly transformed in order to deal with the unexpected occurrence of negative values of the null-collision extinction coefficient. We checked that this transformation does not introduce pathological convergence difficulties that would make it impractical, and our conclusion is that difficulties will only be encountered when the domain of negative null-collision coefficients is optically thick with a high single-scattering albedo. Absorption reduces the difficulty because it reduces the number of times the Monte Carlo weight is multiplied by a negative correction term of absolute value greater than unity. Pathological behaviours will therefore only be encountered when $\hat{k}$ is a poor overestimate of the true extinction coefficient, for scattering dominated media.

If such difficulties were practically encountered, the question could first be addressed by adjusting the branching probabilities P_A, P_S and P_N (we made a practical proposition for these probabilities, but we did not explore alternative choices). Further investigations in this direction

would then certainly consist in transforming the integral structure. We suggest furthermore that this question should be enlarged by considering the meaning of the integral structure highlighted in Sec. 4. It seems indeed that the meshless feature of null-collision algorithms has very little to do with the null-collisions themselves, but rather with an underlying Fredholm formulation that bypasses the question of dealing with path-integrated extinction-coefficients appearing within the exponential function. Introducing null-collisions could then be viewed mainly as a practical way to enhance statistical convergence, very much like introducing control variates in standard Monte Carlo convergence-enhancement techniques. Accordingly, we propose that alternative solutions could be explored starting back from the primary Fredholm formulation. We only opened this investigation field in the last section, but we are convinced that it is worth a close attention.

Acknowledgements

The research presented in this paper was partially conducted with the support of the ITAAC project (Impact du Trafic Aérien sur l'Atmosphère et le Climat), funded by the *Fondation Sciences et Technologies pour l'Aéoronautique et l'Espace* (STAE), Toulouse, France, within the *Réseau Thématique de Recherche Avancée* (RTRA). We also acknowledge support from the European Research Council (Starting Grant 209622: E3ARTHs) and the FRAE (Fédération de Recherche Aéronautique et Espace) for the STRASS project.

A Terminology and bibliographic entries

Null-collision algorithms have been developed independently in two branches of physics: plasma physics and neutron transport. Consequently, according to disciplines and authors, they are found under different designations: null-collisions, fictitious-collisions, pseudo-collisions, null-events, Woodcock-tracking,
delta-scattering, pseudo-scattering, etc.

In the field of plasma physics, null-collision algorithms were first formulated by Skullerud in 1968 [1] to sample ion/molecule collision times. This publication led to further refinements in the same application field, for instance [3], [15] or [16]. These advances have also directly inspired the community studying the dynamics of rarefied gases [17].

Meanwhile, this technique was developed for neutron transport applications by Woodcock and co-workers [2]. They are legitimately recognized as the founders of null-collision algorithms in their field. A significant step was then the formalisation effort reported in [18], that enlarged the application potential of Woodcock algorithm. Today, the so-called "Woodcock tracking" is implemented in many transport simulation codes such as SERPENT [19] or MORET [20]. These ideas have also significantly impacted the communities of image synthesis and tomography research [4] [5] [21].

B Exponential transform

In the literature about exponential transforms [8, 9, 10], a new distribution function $g(\mathbf{x}, \boldsymbol{\omega}) = f(\mathbf{x}, \boldsymbol{\omega}) exp\left(\int_0^L \hat{k}_\sigma d\sigma\right)$ is introduced and is reported in transport equations such as Eq. 1 to get

(here in the particular case of stationary radiation in a non-scattering medium)

$$\boldsymbol{\omega}.\boldsymbol{\nabla} g(\mathbf{x},\boldsymbol{\omega}) = \left[\hat{k}(\mathbf{x}) - k_a(\mathbf{x})\right] g(\mathbf{x},\boldsymbol{\omega}) + k_a(\mathbf{x}) s(\mathbf{x},\boldsymbol{\omega}) exp\left(\int_0^L \hat{k}_\sigma d\sigma\right) \tag{35}$$

The problem is then solved in g instead of f, using Monte Carlo approaches, and the arbitrary $\hat{k}$ field is adjusted in order to minimize the variance of the estimator (essentially using adjoint formulation similar to that of the zero-variance literature). Here, we build a Fredholm equation starting from Eq. 35 (as in *Step 1*):

$$g(\mathbf{x},\boldsymbol{\omega}) = g(\mathbf{x} - L\boldsymbol{\omega},\boldsymbol{\omega}) + \int_0^L d\lambda \left[\hat{k}_\lambda - k_{a,\lambda}\right] g(\mathbf{x} - \lambda\boldsymbol{\omega},\boldsymbol{\omega}) + k_{a,\lambda} s_\lambda exp\left(\int_0^{L-\lambda} \hat{k}_\sigma d\sigma\right) \tag{36}$$

Reporting the expression of $g(\mathbf{x},\boldsymbol{\omega}) = f(\mathbf{x},\boldsymbol{\omega}) exp\left(\int_0^L \hat{k}_\sigma d\sigma\right)$ in Eq. 36 leads to Eq. 28.

C Zero-variance strategy

In the Monte Carlo literature, zero-variance refers to algorithms such that the Monte Carlo weight is strictly and systematically equal to the quantity to be estimated independently of the sampling occurrences. This corresponds to ideal convergence in the sense that perfect convergence is obtained with a single Monte Carlo sampling event. The design of such algorithms is always part of pure-theoretical reasoning and can be quite tedious. Here, starting from Eq. 29 in the restrictive case of $\hat{k} > k_a$ (so that, all terms are positive), such an algorithm can be easily designed using only an ideally optimized importance sampling procedure. Indeed, a random variable Λ of probability density function p_Λ on $[0, +\infty]$ can be introduced to give

$$f(\mathbf{x},\boldsymbol{\omega}) = \int_0^{+\infty} p_\Lambda(\lambda) d\lambda w(\lambda) \tag{37}$$

with

$$w(\lambda) = \frac{1}{p_\Lambda(\lambda)} exp\left(-\int_0^\lambda \hat{k}_\sigma d\sigma\right) \left[k_{a,\lambda} s_\lambda + \left(\hat{k}_\lambda - k_{a,\lambda}\right) f(\mathbf{x} - \lambda\boldsymbol{\omega},\boldsymbol{\omega})\right] \tag{38}$$

and $w(\lambda)$ is equal to f whatever the sampled value of λ as soon as

$$p_\Lambda(\lambda) = \frac{1}{f(\mathbf{x},\boldsymbol{\omega})} exp\left(-\int_0^\lambda \hat{k}_\sigma d\sigma\right) \left[k_{a,\lambda} s_\lambda + \left(\hat{k}_\lambda - k_{a,\lambda}\right) f(\mathbf{x} - \lambda\boldsymbol{\omega},\boldsymbol{\omega})\right] \tag{39}$$

This is Eq. 31 exactly, except for recursive notations.

If we now want that the algorithm branches between pure absorption and null-collisions (to recover the algorithmic structure of Eqs. 9 and 10), it suffices to introduce an absorption probability P and write

$$f(\mathbf{x},\boldsymbol{\omega}) = \int_0^{+\infty} p_\Lambda(\lambda) d\lambda \left\{P w_a(\lambda) + (1-P) w_n(\lambda)\right\} \tag{40}$$

with

$$w_a(\lambda) = \frac{1}{p_\Lambda(\lambda)} exp\left(-\int_0^\lambda \hat{k}_\sigma d\sigma\right) \frac{k_{a,\lambda} s_\lambda}{P} \tag{41}$$

and

$$w_n(\lambda) = \frac{1}{p_\Lambda(\lambda)} exp\left(-\int_0^\lambda \hat{k}_\sigma d\sigma\right) \frac{\left(\hat{k}_\lambda - k_{a,\lambda}\right) f(\mathbf{x} - \lambda\boldsymbol{\omega}, \boldsymbol{\omega})}{1 - P} \tag{42}$$

We keep the previous choice for p_Λ (Eq. 39), and we still want to achieve $w_a(\lambda) = w_n(\lambda) = f(\mathbf{x}, \boldsymbol{\omega})$, then we get

$$P = \frac{k_{a,\lambda} s_\lambda}{k_{a,\lambda} s_\lambda + \left(\hat{k}_\lambda - k_{a,\lambda}\right) f(\mathbf{x} - \lambda\boldsymbol{\omega}, \boldsymbol{\omega})} \tag{43}$$

References

[1] H. Skullerud, "The stochastic computer simulation of ion motion in a gas subjected to a constant electric field," *J. Phys. D: Appl. Phys.*, vol. 1, p. 1567, 1968.

[2] E. Woodcock, T. Murphy, P. Hemmings, and S. Longworth, "Techniques used in the gem code for monte carlo neutronics calculations in reactors and other systems of complex geometry," p. 557, 1965.

[3] S. Lin and J. Bardsley, "The null-event method in computer simulation," *Comput. Phys. Commun.*, vol. 15, no. 3-4, pp. 161–163, 1978.

[4] N. Rehfeld, S. Stute, M. Soret, J. Apostolakis, and I. Buvat, "Optimization of photon tracking in gate," pp. 4013–4015, 2008.

[5] A. Badal and A. Badano, "Monte carlo simulation of x-ray imaging using a graphics processing unit," pp. 4081–4084, 2009.

[6] V. Eymet, D. Poitou, M. Galtier, M. El Hafi, G. Terree, and R. Fournier, "Null-collision meshless monte-carlo - application to the validation of fast radiative transfer solvers embedded in combustion simulators," *Submitted to J. Quant. Spectrosc. Radiat. Transfer*, 2012.

[7] J. Hammersley and D. Handscomb, *Monte carlo methods*. Taylor & Francis, 1975.

[8] P. Sarkar and M. Prasad, "Prediction of statistical error and optimization of biased monte carlo transport calculations.[integral equations]," *Nucl. Sci. Eng.*, vol. 70, no. 3, 1979.

[9] S. Turner and E. Larsen, "Automatic variance reduction for three-dimensional monte carlo simulations by the local importance function transform–1: Analysis," *Nucl. Sci. Eng.*, vol. 127, no. 1, 1997.

[10] S. Turner and E. Larsen, "Automatic variance reduction for three-dimensional monte carlo simulations by the local importance function transform-ii: Numerical results," *Nucl. Sci. Eng.*, vol. 127, no. 1, pp. 36–53, 1997.

[11] J. Delatorre, J. Bézian, S. Blanco, C. Caliot, J. Cornet, J. Dauchet, M. El Hafi, V. Eymet, R. Fournier, J. Gautrais, O. Gourmel, F. Veynandt, N. Meilhac, A. Pajot, M. Paulin, P. Perez, B. Piaud, M. Roger, J. Rolland, and S. Weitz, "Monte-carlo advances and concentrated solar applications," *Sol. Energy*, 2013.

[12] J. Dauchet, S. Blanco, J.-F. Cornet, M. El Hafi, V. Eymet, and R. Fournier, "The practice of recent radiative transfer monte carlo advances and its contribution to the field of microorganisms cultivation in photobioreactors," *J. Quant. Spectrosc. Radiat. Transfer*, 2012.

[13] R. Assaraf and M. Caffarel, "Zero-variance principle for monte carlo algorithms," *Phys. Rev. Lett.*, vol. 83, no. 23, pp. 4682–4685, 1999.

[14] J. Hoogenboom, "Zero-variance monte carlo schemes revisited," *Nucl. Sci. Eng.*, vol. 160, no. 1, pp. 1–22, 2008.

[15] J.-P. Boeuf and E. Marode, "A monte carlo analysis of an electron swarm in a nonuniform field: the cathode region of a glow discharge in helium," *J. Phys. D: Appl. Phys.*, vol. 15, p. 2169, 1982.

[16] M. Brennan, "Optimization of monte carlo codes using null collision techniques for experimental simulation at low e/n," *Plasma Sci., IEEE Transactions on*, vol. 19, no. 2, pp. 256–261, 1991.

[17] K. Koura, "Null-collision technique in the direct-simulation monte carlo method," *Phys. Fluids*, vol. 29, p. 3509, 1986.

[18] W. Coleman, "Mathematical verification of a certain monte carlo sampling technique and applications of the technique to radiation transport problems.," tech. rep., Oak Ridge National Lab., Tenn., 1968.

[19] J. Leppänen, "Performance of woodcock delta-tracking in lattice physics applications using the serpent monte carlo reactor physics burnup calculation code," *Ann. Nucl. Energy*, vol. 37, no. 5, pp. 715–722, 2010.

[20] B. Forestier, J. Miss, F. Bernard, A. Dorval, O. Jacquet, and B. Verboomen, "Criticality calculations on pebble-bed htr-proteus configuration as a validation for the pseudo-scattering tracking method implemented in the moret 5 monte carlo code," 2008.

[21] B. Toth and M. Magdics, "Monte carlo radiative transport on the gpu," 2010.