

Échantillonnage préférentiel multiple exploitant un squelette curviligne pour le lancer de rayons

Laurent Noël¹, John Chaussard², Venceslas Biri¹

¹ Université Paris Est, LIGM, A3SI-ESIEE

2, boulevard Blaise Pascal, 93162 Noisy le Grand CEDEX, France
laurent.noel@univ-paris-est.fr, v.biri@esiee.fr

² Université Paris 13, Sorbonne Paris Cité, LAGA, CNRS(UMR 7539),
F-93430, Villetaneuse, France
chaussard@math.univ-paris13.fr

Résumé

We present an importance sampling strategy for Monte-Carlo path tracing based on a curvilinear skeleton of the void of the scene. This skeleton allows us to build a probability density function that is able to sample directions according to the distribution of light in the scene. We discuss our results and expose some drawbacks of our method. Then we introduce a new strategy based on multiple importance sampling that permits us to correct the artifacts and gives better results.

Nous présentons une méthode d'échantillonnage préférentiel pour l'algorithme du lancer de rayons. Nous nous basons sur un squelette curviligne du vide de la scène nous permettant de construire une densité de probabilité adaptée à la distribution de lumière. Nous présentons nos résultats et discutons les artefacts générés par la méthode. Nous introduisons alors une nouvelle stratégie basée sur l'échantillonnage préférentiel multiple qui corrige ces artefacts et améliore la qualité des images.

Mots clé : Synthèse d'images, Illumination globale, Lancer de rayons, Squelettisation, Échantillonnage préférentiel multiple

1. Introduction

Le lancer de rayons ou *path tracing* est un algorithme de rendu utilisé en illumination globale. Il a été introduit par Kajiyama en 1986 [Kaj86] et vise à générer des images photoréalistes en mimant le comportement et les trajets de la lumière, au prix d'un temps de calcul relativement long, notamment dans le cas où la lumière est fortement indirecte. Si de nombreuses variantes ont été développées parmi lesquelles le *bidirectional path tracing* [LW93], le *photon mapping* [Jen96], le *metropolis light transport* [VG97] ou encore l'emploi de *virtual point lights* [Kel97], le lancer de rayons reste toujours la méthode de référence comme le prouve l'emploi d'Arnold (de la société *Solid Angle*) chez *Dreamworks*, les ouvrages comme [PH10] ou encore les récentes implantations GPU comme le moteur de rendu temps-réel *Brigade*.

Cet algorithme se fonde sur la résolution d'une équation intégrale appelée équation du rendu [Kaj86]. Le calcul de luminosité se fait le long de directions échantillonnées à l'aide

d'une densité de probabilité et l'intégrale est estimée grâce à la méthode de Monte-Carlo. Le choix de ces directions est crucial pour l'amélioration de la qualité du résultat et la réduction du bruit généré dans l'image finale. L'échantillonnage préférentiel (ou *importance sampling*) est alors utilisé [BS94] afin de choisir des directions pertinentes pour accélérer la convergence de l'algorithme. Cette méthode exploite toute l'information disponible sur la fonction à intégrer afin de choisir préférentiellement des directions où celle-ci a de hautes valeurs. Ces informations sont, la plupart du temps, dérivées du modèle de réflexion utilisé pour les matériaux composant les surfaces. La distribution de lumière dans la scène est moins utilisée pour l'échantillonnage préférentiel car beaucoup plus difficile à déterminer. Or, les figures 1.a et b illustrent bien que, dans le cas de zones éloignées de la lumière, il devient nécessaire d'exploiter cette distribution de lumière.

Plusieurs méthodes ont été développées afin de construire des chemins en accord avec la distribution de lumière. Des travaux [Jen95,HP02] en ce sens exploitent une carte de photon afin de construire une estimation de l'irradiance affectant un point d'intersection lors du lancer de rayon. Cette estimation est utilisée afin d'échantillonner efficacement l'hémisphère autour du point en accord avec la distribution de

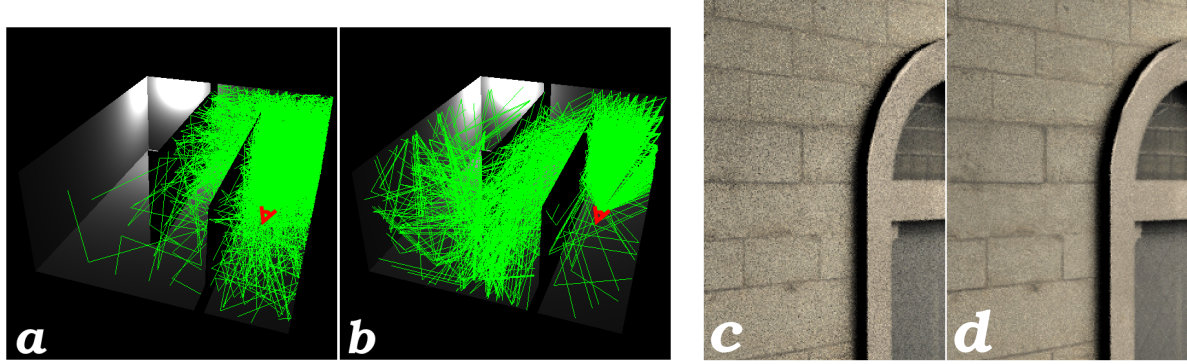


Figure 1: Les rayons s'orientent vers les parties éclairées de la scène grâce à notre squelette curviligne (image b), comparé à l'algorithme classique du lancer de rayon (image a). Les images c et d montrent le bruit généré par le lancer de rayons classique (c) et notre méthode (d).

lumière dans la scène. Cette méthode est efficace mais peut être coûteuse en espace (stockage de la carte de photons) et de plus, ne peut être exploitée que si le nombre de photons est suffisant ce qui n'est pas le cas dans des zones très occultées. Le *bidirectional path tracing* [LW93] construit des chemins en partant à la fois des lumières et de la caméra, permettant ainsi d'assurer une contribution non nulle le long de ces chemins. L'algorithme du *metropolis light transport* [VG97] exploite également cette idée en introduisant des opérateurs de mutation permettant l'exploration de nouveaux chemins à partir d'un chemin fortement lumineux déjà identifié. L'*instant radiosity* [Kel97] tente de répartir efficacement des points lumineux virtuels dans la scène en suivant des chemins partant des lumières. Ces points sont ensuite utilisés comme des sources de lumière primaire pour le rendu final. La luminosité indirecte est ainsi virtuellement transformée en contribution directe.

Nous présentons une méthode d'échantillonnage pour l'algorithme du lancer de rayons [BC12]. Cette méthode est basée sur l'utilisation d'un squelette topologique curviligne du vide de la scène. Il nous permet de choisir des directions porteuses d'énergie lors d'un rebond et ainsi améliorer la vitesse de convergence de l'algorithme. La figure 1 illustre la distribution des rayons dans la scène et la réduction du bruit obtenue avec notre algorithme. Notre méthode est complémentaire aux stratégies présentées ci-dessus [Jen95, LW93, VG97, Kel97] dans le sens où chacune d'entre elles peut également bénéficier de notre approche. Nous verrons néanmoins que l'introduction de cette stratégie d'échantillonnage crée certains artefacts. En conséquence, nous introduirons une stratégie d'échantillonnage préférentiel multiple combinant notre densité de probabilité construite à partir du squelette et une densité en accord avec le modèle de réflexion utilisé. Cette stratégie nous permet d'éliminer certains artefacts produits par la méthode originale et d'obtenir de meilleurs résultats.

2. Lancer de rayons

Le lancer de rayons consiste à lancer des rayons virtuels à travers chacun des pixels de l'image à calculer. On calcule la quantité de lumière traversant chacun des rayons en se

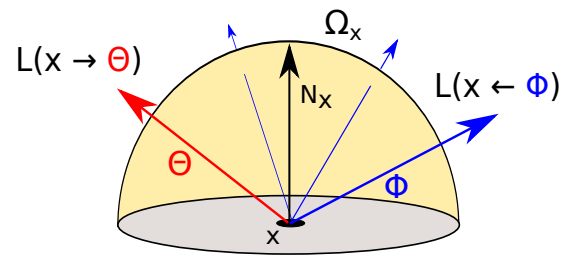


Figure 2: L'équation du rendu modélise la réflexion de lumière en un point de surface

basant sur un modèle de réflexion et sur les matériaux composant la scène. La qualité du résultat dépend du nombre de rayons lancés et des chemins traversés dans la scène.

2.1. Équation du rendu

L'algorithme du lancer de rayon résout à chaque point d'impact entre un rayon et une surface, l'équation du rendu [Kaj86] (figure 2). Elle décrit la manière dont la lumière est réfléchie à chaque rebond :

$$L(x \rightarrow \Theta) = L_e(x \rightarrow \Theta) + L_r(x \rightarrow \Theta)$$

$$L_r(x \rightarrow \Theta) = \int_{\Phi \in \Omega_x} f_r(x, \Theta \leftrightarrow \Phi) L(x \leftarrow \Phi) \cos(N_x, \Phi) d\omega_\Phi$$

Le terme $L(x \rightarrow \Theta)$ représente la radiance en x envoyée dans la direction Θ , quantité devant être calculée pour chaque point de l'image vue par la caméra. En fait, l'équation du rendu exprime une idée simple et intuitive : la radiance du point x dans la direction Θ est la somme de la radiance émise directement par x ($L_e(x \rightarrow \Theta)$) et de la radiance réfléchie par x ($L_r(x \rightarrow \Theta)$). La quantité $L_e(x \rightarrow \Theta)$ est la radiance émise. Elle est non nulle uniquement sur les sources de lumière et est fournie en entrée de l'algorithme pour chaque lumière.

La radiance réfléchie $L_r(x \rightarrow \Theta)$ doit être calculée. Pour cela on doit prendre en compte la lumière arrivant en x , c'est à dire provenant des directions de l'hémisphère englobant x . On modélise donc ce calcul par une intégrale sur l'hémisphère Ω_x centré en N_x , la normale de la surface au point x . Pour chaque direction Φ de Ω_x , $L(x \leftarrow \Phi)$ est la radiance incidente dans cette direction. Cette quantité est multipliée par le facteur $f_s(x, \Theta \leftrightarrow \Phi) = f_r(x, \Theta \leftrightarrow \Phi) \cos(N_x, \Phi)$. La fonction f_r est appelée fonction de réflectance bi-directionnelle (ou plus couramment BRDF pour *bidirectional reflectance distribution function*). Le terme $f_r(x, \Theta \leftrightarrow \Phi)$ exprime la proportion de lumière réfléchie dans la direction Θ lorsqu'elle affecte x selon la direction Φ . C'est une quantité qui dépend du matériau en x . En pratique on la calcule à partir d'un modèle de réflexion et, éventuellement, de textures. Le cosinus atténue la lumière provenant des directions rasantes.

Dans le vide la radiance reste constante le long des lignes droites ce qui nous permet d'exprimer la radiance incidente $L(x \leftarrow \Phi)$ en fonction de la radiance :

$$L(x \leftarrow \Phi) = L(r(x, \Phi) \rightarrow -\Phi)$$

Le terme $r(x, \Phi)$ représente le point visible depuis x dans la direction Φ . En pratique on le calcule en lançant un rayon dans la scène et en cherchant l'intersection la plus proche.

L'équation du rendu nous permet en théorie de calculer la radiance en tout point de surface et pour toute direction. En pratique, il n'existe pas de solution analytique et il est nécessaire d'utiliser des techniques d'approximation numérique pour calculer la partie intégrale de l'équation. La méthode la plus utilisée est l'intégration de Monte-Carlo qui fournit un estimateur général pour le calcul des intégrales.

2.2. Intégration de Monte-Carlo

Soit f une fonction définie sur un domaine D . On désire calculer l'intégrale de f sur D :

$$F = \int_{x \in D} f(x) dx$$

L'intégrateur de Monte-Carlo est fondé sur une distribution de probabilité p nous permettant échantillonner D . Soit $(x_i)_{i=1 \dots N}$ une famille d'échantillons indépendants et distribués selon p . L'estimateur de Monte-Carlo est le suivant :

$$\langle F \rangle = \frac{1}{N} \sum_{i=1}^N \frac{f(x_i)}{p(x_i)}$$

Cet estimateur est non-biaisé : l'espérance $E[\langle F \rangle]$ est égale à F quel que soit le nombre d'échantillons et quelle que soit la distribution de probabilité utilisée à condition que $p(x)$ soit non nul pour tout x dès que $f(x)$ est non nul.

La variance $\sigma^2[\langle F \rangle]$ de l'estimateur nous permet d'évaluer son efficacité. On a :

$$\sigma^2[\langle F \rangle] = \frac{1}{N} \int_{x \in D} \left(\frac{f(x)}{p(x)} - F \right)^2 dx$$

Plus cette variance est faible, meilleure est l'efficacité de l'estimateur. La partie intégrale de l'équation étant une

constante, on voit qu'en augmentant le nombre d'échantillons on diminue la variance.

L'algorithme du lancer de rayons stochastique utilise cette méthode pour estimer l'intégrale $L_r(x \rightarrow \Theta)$ avec $f(\Phi) = f_s(x, \Theta \leftrightarrow \Phi) L(r(x, \Phi) \rightarrow -\Phi)$. Malheureusement l'évaluation de l'intégrande entraîne un appel récursif de L . Cela nous impose de n'utiliser qu'un seul échantillon afin que l'algorithme n'ait pas une complexité exponentielle. L'estimateur utilisé en lancer de rayons est donc :

$$\langle L_r(x \rightarrow \Theta) \rangle = \frac{f_s(x, \Theta \leftrightarrow \Phi) L(r(x, \Phi) \rightarrow -\Phi)}{p(\Phi)}$$

2.3. Échantillonnage préférentiel

Une stratégie couramment utilisée pour réduire la variance de l'estimateur est d'utiliser une distribution p adaptée à la fonction à intégrer. Cette méthode est appelée échantillonnage préférentiel ou *importance sampling*.

Une distribution uniforme sur l'hémisphère ($p(\Phi) = \frac{1}{2\pi}$) conduit généralement à de mauvais résultats : si peu de rayons sont lancés, l'image de sortie sera très bruitée. Cela est dû au fait que toutes les directions sont traitées de la même manière. Or l'intégrande est très faible pour certaines directions comme par exemple les directions rasantes, le cosinus avec la normale étant nul ; ou les directions n'apportant que peu d'énergie. Une bonne distribution doit donner aux directions pour lesquelles l'intégrande est forte plus de chances d'être choisies. Cela conduit à une réduction du bruit et une convergence plus rapide vers la véritable image.

Dans l'algorithme du lancer de rayons, la fonction à intégrer est un produit de deux fonctions :

$$f(\Phi) = f_s(x, \Theta \leftrightarrow \Phi) L(r(x, \Phi) \rightarrow -\Phi)$$

Malheureusement, il est plus difficile de choisir une distribution adaptée à un produit qu'à une fonction simple. La solution la plus couramment utilisée est d'échantillonner en accord avec la fonction f_s . Ce choix se justifie par le fait que la fonction f_s est souvent représentée par un modèle connu de réflexion et des textures. On peut donc utiliser cette donnée immédiatement, l'évaluation du terme $f_s(x, \Theta \leftrightarrow \Phi)$ se faisant en temps constant. Au contraire, l'évaluation du terme $L(r(x, \Phi) \rightarrow -\Phi)$ relance la récursion, sa valeur n'est donc pas immédiatement accessible.

Notre méthode conduit à la construction d'une distribution adaptée au terme $L(r(x, \Phi) \rightarrow -\Phi)$. La procédure est détaillée dans la section 3.

3. Échantillonnage préférentiel guidé par le squelette du vide

Comme nous l'avons évoqué dans la section 2.3, notre objectif est de construire une densité de probabilité permettant d'échantillonner préférentiellement les directions porteuses d'énergie, c'est à dire les directions Φ pour lesquelles le terme $L(x \leftarrow \Phi)$ est fort. Notre méthode utilise un squelette du vide de la scène pour identifier les chemins porteurs d'énergie. Ce squelette est pré-traité pour construire un ensemble de points d'importance. Ces points sont fournis en

entrée de l'algorithme de lancer de rayons et nous permettent à chaque rebond de construire une distribution nous faisant rebondir dans leur direction.

Le squelette utilisé est curviligne et possède la même topologie que le vide. Ces choix sont justifiés par deux raisons :

- Il est important de garder la même topologie que le vide afin de ne pas construire des chemins de lumière inexistantes (passant à travers les murs).
- Un squelette curviligne est un objet à une dimension. Cela nous permet de le représenter à l'aide d'un graphe composé d'un nombre de nœuds réduit. Nous avons ainsi une représentation légère en mémoire et les pré-calculs sont rapides.

3.1. Voxelisation et squelettisation

La première étape consiste à voxeliser la scène puis à inverser le résultat (les voxels pleins deviennent vides et inversement). L'inversion est nécessaire afin d'avoir une voxelisation du vide. Une telle voxelisation binaire peut se faire très rapidement [FBP09].

Nous calculons ensuite un squelette à partir de la voxelisation précédente. Un squelette d'un objet X est un sous ensemble de points de X possédant la même topologie que X , soit, en 3 dimensions, le même nombre de composantes connexes, trous et cavités. De plus un squelette de X doit être centré au sein de X et au plus de dimension 2. Les travaux en squelettisation se concentrent sur l'obtention d'un squelette ressemblant géométriquement à l'objet initial : dans ce cas la difficulté est d'obtenir un squelette ne comportant pas d'artefacts pouvant être vus comme du bruit.

Les travaux récents en squelettisation se placent dans le cadre des cubiques complexes afin d'obtenir un squelette robuste [BC06]. Nous utilisons l'algorithme de squelettisation présenté dans [CC09] prenant en entrée un objet voxelisé et produisant un squelette de dimension inférieure centré dans l'objet et représenté par un cubique complexe. Nous avons modifié l'algorithme dans le but de ne conserver que les éléments de dimension 1 conduisant ainsi à l'obtention d'un squelette curviligne.

Le squelette est filtré en utilisant la méthode présentée dans [Cha10]. Cette méthode ne requiert aucun paramètre d'entrée : durant la squelettisation nous calculons l'espérance de vie de chaque point de l'objet (le nombre d'itérations nécessaires pour supprimer le point du squelette). Nous calculons également pour chaque point p de l'objet la taille de la boule maximale contenant dans l'objet et contenant p (appelée la valeur d'ouverture de p). Le filtrage consiste à garder, durant la squelettisation, tous les points ayant une espérance de vie supérieure à leur valeur d'ouverture. Cela nous permet d'obtenir un squelette robuste au bruit et conservant l'aspect visuel de l'objet original.

La dernière étape consiste à convertir le squelette obtenu, représenté par un cubique complexe, en un graphe dont les nœuds sont des points de \mathbb{R}^3 et dont les arêtes encodent la topologie du vide. La figure 3 illustre le squelette curviligne du

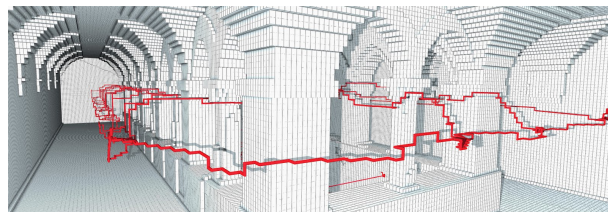


Figure 3: Squelette du vide de Sponza (en rouge)

vide de Sponza. On observe que la topologie est bien conservée (le squelette passe entre chacun des piliers) et qu'il est bien centré dans la scène, conservant ainsi une apparence visuelle proche de la scène.

En plus du squelette nous conservons une grille 3D couvrant la scène, dans chaque case, l'indice du nœud du squelette le plus proche. Ainsi nous pouvons récupérer en temps constant le nœud le plus proche de chaque point 3D de la scène. Cette grille se calcule automatiquement lors de l'étape de squelettisation.

3.2. Calcul des points d'importance

Soit $G = (V, E)$ le graphe représentant le squelette. Pour chaque nœud $n \in V$ un point d'importance imp_n est construit en direction de la lumière L .

Soit n_L le nœud le plus proche de la lumière. On commence par calculer un arbre des plus courts chemins de racine n_L . Pour cela on utilise l'algorithme de Dijkstra [Dij71] et une distance d basée sur l'illumination. Soit $u = (u_1, u_2) \in E$ une arête du graphe. Si le nœud u_2 est visible depuis n_L on fixe $d(u) = 1$, sinon $d(u) = 10$. Ainsi les chemins illuminés ont une distance associée plus courte et nous conduisent plus rapidement à la lumière. À l'issue de cette étape on possède pour chaque nœud le plus court chemin vers le nœud n_L .

Soit n un nœud du graphe. On note $V(n)$ l'ensemble des nœuds visibles depuis n le long du plus court chemin vers n_L . Le point d'importance imp_n associé à n est le barycentre de $V(n)$ (voir figure 4).

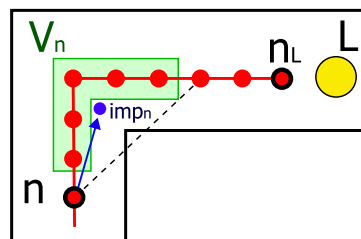


Figure 4: Construction des points d'importance. Les points visibles depuis n le long du plus court chemin jusqu'à la lumière sont encadrés en vert. Leur barycentre est le point d'importance imp_n associé à n .

Scene	Corridor 1	Corridor 2	Sponza
MSE 100	145 / 57	301 / 156	881 / 826
MSE 40	434 / 260	924 / 628	2678 / 2570

Table 1: Comparaison des temps de calcul pour obtenir un MSE de 100 et de 40 avec les images de référence. Dans chacune des cases le nombre de gauche est le temps associé au lancer de rayons standard, celui de droite est celui associé à notre méthode.

3.3. Utilisation des points d'importance

Les points d'importance sont utilisés pour échantillonner l'hémisphère Ω_x lors d'un rebond en x . Pour cela on récupère le nœud n le plus proche de x (faisable en temps constant dans notre implantation en utilisant une grille 3D). On construit alors la direction d'importance \vec{ximp}_n associée à x . La distribution utilisée est alors :

$$p_{skel}(\Phi) = \frac{s+2}{2\pi} \cos^s(\vec{ximp}_n, \Phi)$$

Le paramètre s permet de concentrer les tirs de rayons autour de la direction d'importance \vec{ximp}_n .

3.4. Résultats et discussion

Notre méthode conduit à une bonne réduction du bruit, en particulier dans les zones sombres où l'illumination est exclusivement indirecte (voir figure 5). Au niveau des temps de calcul, notre méthode converge plus vite que le lancer de rayons standard. le tableau 1 indique les temps requis pour atteindre différentes valeurs de MSE entre nos résultats et une image de référence.

Le tableau 2 indique les coûts supplémentaires en terme de mémoire et de temps de calcul associées aux scène "Sponza" et "Corridor". Le temps indiqué inclut la voxelisation, la squelettisation et le calcul des points d'importance. Ce dernier est négligeable comparé aux deux premiers. Notre algorithme de squelettisation n'a pas encore été parallélisé, ce qui explique le temps de calcul relativement élevé.

Notre méthode produit un artefact : les speckles. Les speckles sont des pixels anormalement illuminés, dû à un mauvais échantillonnage. Lorsqu'une direction possédant une forte valeur mais une faible probabilité est choisie, l'estimateur de Monte-Carlo donne une valeur très élevée conduisant à l'apparition d'un speckle. Ce type d'artefact s'estompe en augmentant le nombre de rayons par pixels grâce à la convergence de l'estimateur mais au prix d'un long temps de calcul. Une solution simple pour supprimer ces artefacts est de borner l'illumination à une valeur fixée. On introduit alors un

Scene	Coût mémoire	résolution grille	temps calcul
Corridor	18 MB	93x205x250	13s
Sponza	130 MB	210x308x500	30s

Table 2: Coûts associés aux scène Sponza et Corridor (image la plus à gauche sur la figure 5)

biais dans la méthode puisque une partie de l'énergie lumineuse ne sera jamais distribuée. Une meilleure solution est présentée dans la section 4.

D'autre part, plusieurs cas pathologiques, relativement rares, peuvent nous conduire à nous rabattre sur un échantillonnage basé sur la BRDF :

- La direction d'importance peut pointer du côté opposé à la normale en x . Cela signifie que le point d'importance est situé de l'autre côté de la surface.
- Le point d'importance peut ne pas être visible depuis x . Dans ce cas il est inutile de tirer aux alentours de sa direction car notre rayon a de fortes chances d'être intersecté avant d'atteindre la lumière.

Enfin, en concentrant nos tirs de rayons vers les zones éclairées de la scène, nous pouvons rater des échanges lumineux entre surfaces proches comme le montre la figure 6.

Ces problèmes nous ont naturellement conduit à exploiter une autre solution appelée échantillonnage préférentiel multiple qui répond au problème des speckles et des cas dégénérés.

4. Échantillonnage préférentiel multiple

Comme nous l'avons évoqué dans la section précédente, notre méthode conduit à l'apparition de speckles. Le fait d'ignorer totalement la BRDF au moment d'échantillonner est une des causes de l'apparition de cet artefact. Nous proposons donc d'utiliser une méthode appelée échantillonnage préférentiel multiple (MIS pour *multiple importance sampling*) pour prendre en compte à la fois notre stratégie d'échantillonnage par squelette curviligne et celle fondée sur la BRDF.

4.1. Cadre théorique

Le MIS est une technique utilisée pour échantillonner efficacement dans le cas de produit de fonctions comme c'est le cas pour le lancer de rayons. L'idée est d'utiliser une densité de probabilité construite à partir de la combinaison de plusieurs densités adaptées à chacun des facteurs de la fonction.

Soit $f(x) = f_1(x)f_2(x)...f_n(x)$ une fonction à intégrer sur un domaine D . Soit p_1, \dots, p_n les densités de probabilité adaptées à chacune des fonctions f_i . L'estimateur combiné pour l'intégrale de f est :

$$\langle F \rangle = \frac{\omega_I(X_I)f(X_I)}{c_I p_I(X_I)}, I = \min\{i \mid U < \sum_{j=1}^i c_j\}$$

Les ω_i sont des fonctions de pondération positives telles que $\sum_i \omega_i = 1$ et les c_i des probabilités associées à chacune des densités p_i . Le nombre réel U est choisi aléatoirement et uniformément dans l'intervalle $[0, 1]$. Cet estimateur est non biaisé et permet de construire une famille de stratégies d'échantillonnage en modifiant les ω_i et les c_i .

L'idée derrière cet estimateur est simple : on choisit aléatoirement une stratégie p_I en se basant sur les probabilités c_i . Cette stratégie est utilisée pour tirer un échantillon X_I . Cet échantillon est alors utilisé pour calculer l'estimation.

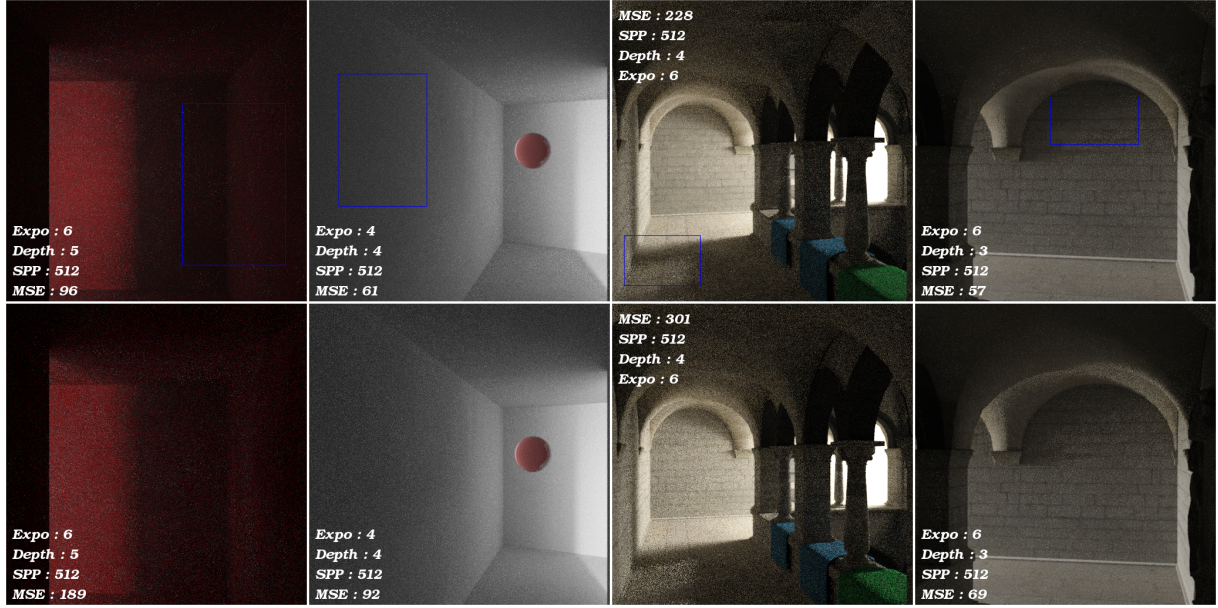


Figure 5: Comparaison du résultat de notre algorithme (images du haut) avec un lancer de rayons standard (images du bas). Sur chaque image l'erreur quadratique moyenne (MSE) avec une image de référence est affichée. Notre algorithme obtient de meilleurs résultats sur les quatre scènes.

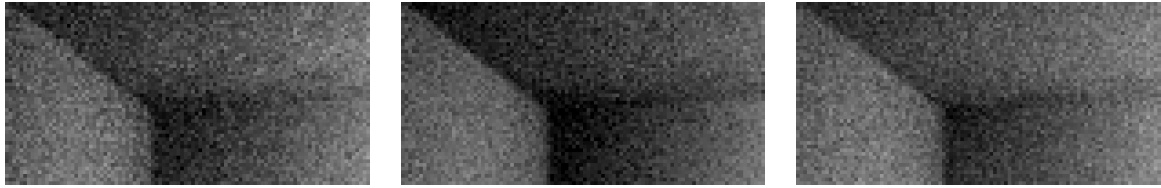


Figure 6: Notre méthode (image centrale) sans MIS conduit à rater des échanges lumineux entre surfaces proche. Le path tracing standard (image de gauche) génère un résultat moins sombre. L'emploi du MIS (image de droite) corrige cet artefact en donnant une chance d'être utilisée à la stratégie basée sur le modèle de réflexion

Lorsque l'on utilise un seul échantillon, comme c'est le cas en lancer de rayons, il a été démontré que les fonctions de pondération suivantes sont optimales (en terme de variance) :

$$\hat{\omega}_i(x) = \frac{c_i p_i(x)}{\sum_j c_j p_j(x)}$$

L'utilisation de ces fonctions conduit à l'estimateur suivant appelé "balance heuristic" :

$$\langle F \rangle = \frac{f(X_I)}{\sum_i c_i p_i(X_I)}, I = \min\{i \mid U < \sum_{j=1}^i c_j\}$$

4.2. Utilisation de la méthode

Dans le cas du lancer de rayons la fonction à intégrer est :

$$f(\Phi) = f_s(x, \Theta \leftrightarrow \Phi) L(x \leftarrow \Phi)$$

Notre méthode nous a conduit à la densité p_{skel} pour échantillonner en accord avec L . Un grand nombre de stratégies ont été étudiées pour échantillonner efficacement en accord avec f_s en fonction du modèle de réflexion utilisé.

N'importe quelle densité p_s adaptée à f_s peut être choisie. L'estimateur est alors le suivant :

$$\langle L_r(x \rightarrow \Theta) \rangle = \frac{f_s(x, \Theta \leftrightarrow \Phi) L(x \leftarrow \Phi)}{c_{skel} p_{skel}(\Phi) + (1 - c_{skel}) p_s(\Phi)}$$

La direction Φ est échantillonnée avec la densité choisie uniformément selon les probabilités c_{skel} et $1 - c_{skel}$. Le paramètre c_{skel} peut être fixé ou bien calculé en fonction du point x (par exemple par rapport à la distance à la lumière).

4.3. Résultats et discussion

L'emploi du MIS conduit à une forte réduction des speckles comme le montre la figure 7. On peut observer un très grand nombre de speckles dû au paramètre $s = 2$, entraînant une plus forte concentration autour de la direction d'importance. Le MIS supprime ces speckles tout en conservant la réduction du bruit comme on peut le voir sur le mur de gauche. Les valeurs de MSE confirment ces observations : les speckles constituent l'artefact majeur de notre méthode originale et leur suppression est cruciale pour l'amélioration des images calculées.

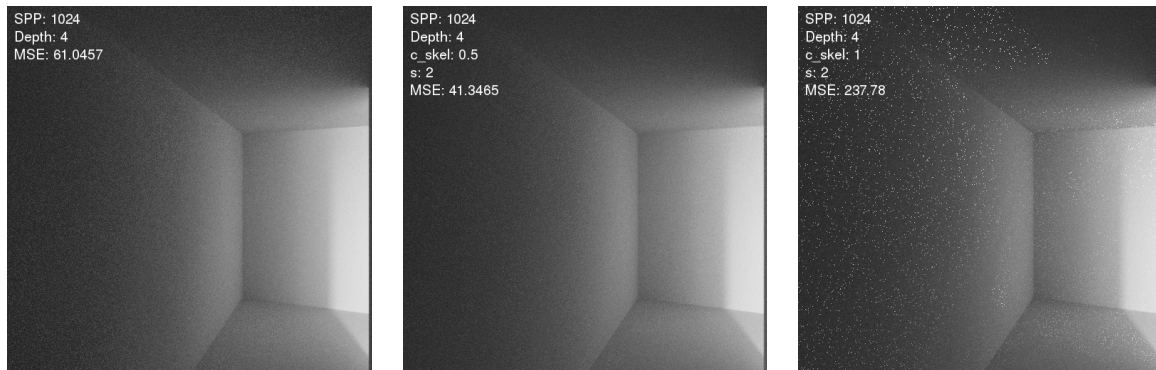


Figure 7: À gauche le lancer de rayons standard. Au centre le MIS appliqué à notre méthode. À droite la méthode de base ($c_{skel} = 1$ équivaut à n'utiliser que le squelette). On observe une très forte réduction des speckles.

La figure 6 est un gros plan sur le coin du couloir. La méthode originale donne un résultat plus sombre que celui obtenu avec le lancer de rayons standard. Cet artefact est dû au fait que l'on n'échantillonne pas l'hémisphère complet autour du point d'impact mais seulement une sous partie centrée autour de la direction d'importance. Au contraire le MIS permet de visiter la totalité des chemins de lumière (si la densité de probabilité utilisée pour le modèle de réflexion est valide).

5. Conclusion et travaux futurs

Nous avons montré que l'utilisation d'un squelette curviligne du vide de la scène permet d'extraire des informations pertinentes sur la distribution de lumière. Ces informations nous permettent d'échantillonner efficacement des chemins porteurs d'énergie dans la scène. La méthode génère néanmoins des artefacts du fait que l'on ignore totalement les propriétés locales de réflexion associées aux points d'impacts. L'emploi de l'échantillonnage préférentiel multiple permet de corriger ces artefacts en combinant notre stratégie et la stratégie associée au modèle de réflexion.

Le paramètre c_{skel} utilisé dans le MIS permet de pondérer l'utilisation du squelette par rapport aux caractéristiques locales. Nous avons utilisé des valeurs fixes pour nos résultats. Or il est évident que faire varier la contribution du squelette en fonction de la position dans la scène peut permettre d'obtenir de meilleurs résultats. Par exemple, il est inutile de l'utiliser dans les zones directement éclairées puisque la lumière est déjà atteinte : il suffit de calculer la contribution directe pour optimiser l'estimation. Nous cherchons à identifier des stratégies pour faire varier ce paramètre, par exemple par rapport à la distance à la lumière ou aux occlusions locales.

Nous nous orientons également vers l'utilisation du squelette pour d'autres algorithmes d'illumination globale, en particulier les algorithmes basés sur les *virtual point lights* ou sur le *photon mapping*. Le squelette peut nous permettre de guider plus efficacement les photons vers la caméra ou bien de placer les VPLs de manière à maximiser leur contribution à l'image finale.

Références

- [BC06] BERTRAND G., COUPRIE M. : A new 3d parallel thinning scheme based on critical kernels. In *Discrete Geometry for Computer Imagery* (2006), Springer, pp. 580–591.
- [BC12] BIRI V., CHAUSSARD J. : Skeleton Based Importance Sampling for Path Tracing. Andujar C., Puppo E., (Eds.), Eurographics Association, pp. 1–4.
- [BS94] BLASI P. B. L. S., SCHLICK C. : An importance driven monte-carlo solution to the global illumination problem. In *Eurographics Workshop on Rendering 5* (1994).
- [CC09] CHAUSSARD J., COUPRIE M. : Surface thinning in 3d cubical complexes. In *Proceedings of the 13th International Workshop on Combinatorial Image Analysis* (Berlin, Heidelberg, 2009), IWCI '09, Springer-Verlag, pp. 135–148.
- [Cha10] CHAUSSARD J. : *Topological tools for discrete shape analysis*. PhD thesis, Université Paris-Est, December 2010.
- [Dij71] DIJKSTRA E. : *EWD316 : A Short Introduction to the Art of Programming*. Technische Hogeschool, 1971.
- [FBP09] FOREST V., BARTHE L., PAULIN M. : Real-time hierarchical binary-scene voxelization. *J. Graphics, GPU, & Game Tools*. Vol. 14, Num. 3 (2009), 21–34.
- [HP02] HEY H., PURGATHOFER W. : Importance sampling with hemispherical particle footprints. In *Proceedings of the 18th spring conference on Computer graphics* (New York, NY, USA, 2002), SCCG '02, ACM, pp. 107–114.
- [Jen95] JENSEN H. W. : Importance driven path tracing using the photon map. In *Eurographics Rendering Workshop* (1995), Springer-Verlag, pp. 326–335.
- [Jen96] JENSEN H. W. : Global illumination using photon maps. In *7th Eurographics Workshop on Rendering, Technical Rendering 96* (juin 1996), Springer-Verlag, pp. 21–30.
- [Kaj86] KAJIYA T. : The Rendering Equation. In *Computer Graphics (ACM SIGGRAPH '86 Proceedings)* (août 1986), vol. 20(4), pp. 143–150.

- [Kel97] KELLER A. : Instant radiosity. In *Proceedings of the 24th conference on Computer graphics and interactive techniques* (New York, NY, USA, 1997), SIGGRAPH '97, ACM Press/Addison-Wesley Publishing Co., pp. 49–56.
- [LW93] LAFORTUNE E. P., WILLEMS Y. D. : Bi-directional path tracing. In *3rd International Conference on Computational Graphics and Visualization Techniques (COMPUGRAPHICS '93)* (1993), pp. 145–153.
- [PH10] PHARR M., HUMPHREYS G. : *Physically Based Rendering, Second Edition : From Theory To Implementation*, 2nd ed. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2010.
- [VG97] VEACH E., GUIBAS L. J. : Metropolis light transport. In *SIGGRAPH '97 : Proceedings of the 24th conference on Computer graphics and interactive techniques* (1997), pp. 65–76.