Université de Montréal

# Simulation du transfert de radiance en harmoniques sphériques dans des scènes dynamiques

Jean-Sébastien Guay
Département de génie informatique et génie logiciel
École polytechnique de Montréal

Mémoire présenté en vue de l'obtention
du diplôme de maîtrise ès sciences appliquées
(génie informatique et génie logiciel)

Avril 2008

© Jean-Sébastien Guay, 2008

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

# SIMULATION DU TRANSFERT DE RADIANCE EN HARMONIQUES SPHÉRIQUES DANS DES SCÈNES DYNAMIQUES

présenté par: GUAY Jean-Sébastien
en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées
a été dûment accepté par le jury d'examen composé de:

Mme CHERIET, Farida, Ph.D., président
M. OZELL, Benoît, Ph.D., membre et directeur de recherche
M. POULIN, Pierre, Ph.D., membre

# REMERCIEMENTS

J'aimerais premièrement remercier mon directeur de recherche, Benoît Ozell, qui a su à la fois me guider et me laisser la liberté de mener mon projet comme je le voulais, à parts égales.

Ensuite, je n'aurais pu compléter ce projet sans la collaboration et l'encouragement de ma conjointe, Annabella Cengarle, et de mes deux fils, Raphaël et Marc-Antoine.

J'aimerais également remercier les organismes suivants, qui m'ont fourni une aide financière indispensable sous la forme de bourses d'études au cours de ma maîtrise: la Fondation de Polytechnique, Ubisoft, Fondation des bourses du millénaire.

Finalement, j'aimerais souligner l'aide que m'ont fournie certains gens, par leurs propres projets publiés.

- Vincent Domurado, dont le mémoire de recherche "Étude et modélisation de la réflectance de la surface d'objets réels" fut pour moi un résumé intéressant des différents modèles d'illumination au début de mon travail.
- Stéphane Metz et Fabien Houlmann, pour les nuanceurs de *tone mapping* qui m'ont aidé à implanter cette fonctionnalité.
- Sévan Dalkian, pour les discussions intéressantes et le projet de fin d'études qui m'a inspiré à améliorer la qualité d'image de mon travail.

# Résumé

Dans les dernières années, plusieurs travaux ont été faits afin de faire le pont entre les techniques d'illumination globale et le rendu temps réel. Par exemple, le *transfert de radiance précalculé* consiste à découpler l'illumination de la réaction des surfaces à la lumière, pour ensuite encoder ces deux termes en harmoniques sphériques, permettant ainsi certaines opérations sans avoir à faire un recalcul de l'éclairage. Un autre exemple, le *tampon d'illumination*, vise à calculer l'illumination locale d'une scène en utilisant un rendu temps réel classique, puis à utiliser une fonction de rendu de haute qualité pour calculer les effets de l'illumination globale, qui seront superposés à l'illumination locale. Ensuite, si la scène est modifiée, les parties affectées de la scène pourront être recalculées en arrière-plan, sans ralentir l'interaction temps réel, et les résultats de ce recalcul seront affichés à mesure qu'ils sont disponibles.

Nous proposons d'utiliser ces deux techniques de façon complémentaire. À la base, le tampon d'illumination utilise le tracé de chemins bidirectionnel comme fonction de rendu de haute qualité, et cette option présente des avantages, mais nous constatons que la remplacer par le transfert de radiance précalculé peut également avoir des avantages. Parmi ceux-ci, on compte le fait de pouvoir changer totalement l'illumination de la scène sans avoir à faire de recalcul, en plus des autres situations où le transfert de radiance précalculé permet de faire des opérations sur la scène tout en restant valide. En combinant cette technique au tampon d'illumination, les autres cas, où l'illumination devrait être recalculée, sont également couverts sans toutefois bloquer l'interaction utilisa-

teur. Dans le but de faire avancer les recherches dans le domaine de l'illumination globale en temps réel dans des scènes dynamiques, nous avons donc combiné ces deux techniques pour vérifier si ces observations s'avéraient correctes.

La méthode développée pour ce travail de recherche permet, étant donnée une scène infographique, d'effectuer un calcul des ombres à l'aide du transfert de radiance en harmoniques sphériques. L'utilisateur peut ensuite visualiser la scène et modifier les objets qu'elle contient. Si l'utilisateur déplace un objet, l'ombre que cet objet dessinait sur les autres objets de la scène disparaîtra progressivement, et s'il doit former une nouvelle ombre ailleurs, celle-ci apparaîtra après un certain temps. Cette mise à jour progressive pourrait être utile dans plusieurs autres contextes, surtout lorsque le point de vue est relativement loin de ce qui est mis à jour.

Le barème utilisé pour valider nos observations était séparé en trois parties:

a) Vitesse d'interaction. L'utilisateur doit pouvoir continuer à se déplacer et interagir avec la scène même pendant une mise à jour de l'illumination résultant d'une interaction antérieure.

b) Temps de mise à jour. La mise à jour de l'illumination doit prendre un temps raisonnable, ce qui sera défini dans le travail.

c) Qualité d'image. La représentation des ombres doit être valide, et doit comporter un minimum d'artéfacts visuels.

L'expérimentation a donné lieu à des résultats intéressants. Premièrement, la vitesse d'interaction ne diminue pas de façon perceptible. Deuxièmement, la mise à jour

complète peut prendre entre quinze et soixante secondes pour des scènes typiques, mais l'image sera en général acceptable après cinq à dix secondes, et l'utilisateur ne perd pas l'interactivité. Finalement, les paramètres de qualité d'image donnant ces résultats présentent quelques artéfacts visuels minimes. Si on est prêt à une diminution de performance, il est possible d'augmenter la qualité d'image pour donner des résultats visuels impressionnants.

Il est important de noter que le calcul d'ombres a été choisi simplement pour démontrer le processus; celui-ci s'appliquerait tout autant à d'autres types de calculs faits à la surface d'un objet, tel que les interréflexions diffuses. Appliquer cette technique à un effet dépendant du point de vue, comme par exemple les réflexions spéculaires sur une surface miroir, impliquerait par contre un recalcul de toutes les surfaces spéculaires de la scène chaque fois que le point de vue change. De plus, l'une des caractéristiques du tampon d'illumination, la subdivision du maillage aux endroits où l'illumination était mal représentée, n'a pas fait partie de notre travail.

L'expérimentation a permis de constater que même si le transfert de radiance précalculé présente des avantages, surtout par son encodage, lorsque utilisé en combinaison avec un tampon d'illumination, il nécessite également de faire un plus grand nombre de tracés de rayons lors d'une mise à jour. En plus de ralentir la mise à jour comparativement à un tracé de rayons normal en espace image, cela fait en sorte qu'il serait difficile d'utiliser cette technique dans un contexte où il faut représenter l'illumination dépendant du point de vue, car il serait alors nécessaire de faire des mises à jour très fréquemment.

La méthode présentée dans ce travail comporte quand même de grands avantages dans d'autres contextes. Notre travail se compare avantageusement à la technique des textures d'illumination, qui est couramment utilisée dans les simulations et jeux afin de donner l'apparence d'une illumination de haute qualité à une scène. Les avantages que sont la mise à jour progressive lorsqu'un changement survient dans la scène et la possibilité de changer l'illumination de la scène rapidement pèsent en la faveur du concept présenté dans ce travail de recherche, car les textures d'illumination ne s'appliquent qu'à des scènes et une situation d'illumination totalement statiques. De plus, l'illumination précalculée par notre travail sera moins gourmande en termes d'espace de stockage par rapport à la grande quantité de textures requises pour appliquer les textures d'illumination à une scène en entier.

# ABSTRACT

In the last few years, researchers have worked on combining global illumination techniques with real-time rendering. For example, *Precomputed Radiance Transfer* decouples the lighting environment from surfaces' reaction to light and encodes these two terms in spherical harmonics, which allows certain operations on the objects without having to recalculate the objects' shading. Another technique, the *Shading Cache*, calculates a scene's local illumination using classic real-time rendering and uses a high-quality sample renderer to calculate global illumination effects which can then be used in addition to the local illumination. If the objects are modified, only the affected parts of the scene will be recalculated in a background process, which will have no effect on the interactivity, and the results will be displayed as they are completed.

We present a method which uses the two techniques described above together. Originally, the Shading Cache used bidirectional path tracing to render high quality samples. This has its advantages, but using Precomputed Radiance Transfer instead has many advantages as well. For example, the fact that the scene's lighting environment can be totally changed without having to recalculate the scene's shading, as well as some situations where changes can be made to the scene while still keeping valid shading. By combining it to the Shading Cache, we account for the cases where valid shading cannot be preserved without recalculating it by offering a way of doing this recalculation without blocking the user interaction. We combined these two techniques to verify that our ob-

servations were correct, and with the greater goal of helping the field of real-time global illumination in dynamic scenes move forward.

The technique developed for this thesis calculates soft shadows using Precomputed Radiance Transfer in the spherical harmonics basis. The user can then view the scene and modify the objects it contains. If the user moves an object, the shadow which that object projected onto other objects will disappear, and if a new shadow should form somewhere else, it will appear gradually. This progressive update could be useful in many contexts, especially when the viewpoint is relatively far from the objects to be updated.

Our observations were validated against the following three criteria:

a) Interactivity. The user should be able to continue moving around in the scene and interacting with it even if an update, caused by a previous interaction, is in progress.

b) Update time. The time it takes to update the shading should be reasonable. This will be defined in the thesis.

c) Image quality. Shadows should be valid and show only minimal visual artifacts.

Interesting results were obtained through experimentation. First of all, there is no significant change in interaction speed during an update. Second, a complete update of a typical scene's shading can take between 15 and 60 seconds, but the image generally reaches an acceptable state after about 5 to 10 seconds, and the user does not lose interactivity. Finally, the image quality settings which give these results show some visual arti-

facts, but these are small. If the trade off of longer update times is acceptable, impressive results, with practically no visual artifacts, are possible.

Shadow rendering was chosen to demonstrate the process of Precomputed Radiance Transfer calculation and progressive update. Other shading calculations, such as diffuse interreflection, would apply just as well. However, applying this technique to a view-dependent shading calculation, for example specular reflection on mirror surfaces, would imply recalculating all specular surfaces in the scene each time the viewpoint changes. Also, adaptive mesh subdivision to refine shading in certain areas, which was one of the features of the Shading Cache, was not part of this thesis.

We noted through experimentation that even though Precomputed Radiance Transfer has several advantages when used in combination with the Shading Cache (mostly because of its encoding), it also requires to trace a large number of rays for each update. This not only slows down the update compared to a normal ray tracing, but it also means that this technique could not be used in a context where view-dependent shading must be calculated, because it would then be necessary to update the shading very often.

The technique presented in this thesis has many advantages in other contexts. When compared to light maps, which is a technique often used in simulations and games to give the appearance of high quality lighting to a scene, our technique is superior in many respects. The progressive update when a change happens in the scene and the possibility to change the lighting environment very quickly are features that do not apply to light maps, which can only be used if the scene and the lighting environment are totally static. Also, the results of the precomputation in our technique are more compact in terms

of storage than using a large number of light maps which would be required for a large scene, as well as for representing multiple lighting situations.

# Table des matières

Remerciements....iv

Résumé....v

Abstract....ix

Table des matières....xiii

Liste des tableaux....xvii

Liste des figures....xviii

Liste des sigles et abréviations....xix

Formules....xix

Glossaire....xx

Introduction....1

Motivation de la recherche....1

Plan du document....2

Définitions et antécédents....4

1.1 Ombrage et illumination....4

1.2 Techniques de génération d'images....6

1.3 Harmoniques sphériques....9

1.3.1 Propriétés....9

Revue historique et littéraire....11

2.1 Effets de base....11

2.1.1 Illumination globale....11

2.1.2 Illumination précalculée....13
2.1.3 Accélération du tracé de rayons....13
2.1.4 Tracé de rayons interactif....14
2.1.4.1 Optimisation bas niveau....14
2.1.4.2 Optimisation des algorithmes....15
2.1.4.3 Méthodes de tampons....16
2.2 Transfert de radiance précalculé....17
2.2.1 Précalcul....18
2.2.2 Visualisation....24
2.3 Tampon d'illumination....26
2.4 Observations....31
2.5 Objectifs généraux et spécifiques de la recherche....32
2.5.1 Objectif général....32
2.5.2 Objectifs spécifiques....32
2.5.3 Quantification des objectifs....33
2.5.4 Éléments exclus de la recherche....34
Méthodologie et implantation....36
3.1 Implantation....36
3.1.1 Implantation du transfert de radiance précalculé....36
3.1.1.1 Précalcul....36
3.1.1.2 Tracé de rayons....38
3.1.1.3 Visualisation....40

3.1.1.4 Différences....42
3.1.2 Implantation du tampon d'illumination....44
3.1.2.1 Différences....48
3.2 Expérimentation....49
3.2.1 Scènes....50
3.2.1.1 Sphère et cube....50
3.2.1.2 Échiquier....50
3.2.1.3 Optimisation des scènes....52
3.2.2 Environnements d'éclairage....53
3.2.3 Rotation des coefficients en harmoniques sphériques....55
Résultats et discussion....56
4.1 Résultats....56
4.1.1 Temps de calcul....56
4.1.2 Vitesse d'interaction....58
4.1.3 Taille des données....58
4.1.4 Taille des données en fonction du nombre de bandes....59
4.1.5 Comparaison entre la version originale et optimisée des scènes....61
4.1.6 Comparaison entre 400 et 10 000 échantillons par sommet....61
4.1.7 Changement de l'environnement d'éclairage....64
4.1.8 Rotation des harmoniques sphériques....64
4.2 Discussion....65
4.2.1 Technique de rendu....66

4.2.2 Subdivision du maillage.....68

4.2.3 Comparaison aux textures d'illumination.....69

Conclusion.....73

Bibliographie.....75

# Liste des tableaux

Tableau 1: Composition de la scène "Sphère et cube".....50

Tableau 2: Composition de la scène "Échiquier".....52

Tableau 3: Temps de calcul et de mise à jour.....57

Tableau 4: Vitesse d'interaction au repos et lors d'une mise à jour.....58

Tableau 5: Taille des données.....58

Tableau 6: Taille des données en fonction du nombre de bandes.....60

Tableau 7: Effet de l'optimisation des scènes.....61

Tableau 8: Comparaison entre 400 et 10 000 échantillons par sommet.....62

Tableau 9: Temps requis pour changer la situation d'éclairage.....64

Tableau 10: Vitesse d'affichage lors d'une rotation de la fonction d'éclairage.....65

# Liste des figures

Figure 1: Image rendue par tracé de chemins. [Kajiya1986].....12

Figure 2: Image générée par illumination globale par tracé de photons. [Jensen1996]....12

Figure 3: Exemple de visualisation des fonctions de transfert avec ombres et interréflexions [Sloan2002].....24

Figure 4: Exemples de transfert diffus et spéculaire [Sloan2002].....25

Figure 5: Exemple de la sélection des échantillons et du raffinement du maillage du tampon de rendu [Tole2002] (légende des images traduites).....30

Figure 6: Diagramme de séquence présentant le processus de calcul des coefficients des fonctions de transfert.....38

Figure 7: Diverses façons de représenter le transfert de radiance précalculé.....42

Figure 8: Création de l'objet ChangeSet lors de la manipulation d'un objet.....46

Figure 9: Traitement lors d'une mise à jour.....47

Figure 10: Scène "Sphere et cube".....52

Figure 11: Scène "Échiquier".....52

Figure 12: Environnements d'éclairage.....54

Figure 13: Processus d'interaction et de mise à jour. (a,b,c) Scène "Sphère et cube"; (d,e,f) Scène "Échiquier".....57

Figure 14: Comparaison de la qualité d'image en fonction du nombre de bandes.....60

Figure 15: Comparaison de la scène "Sphère et cube" à 400 et 10 000 échantillons par sommet.....63

# Liste des sigles et abréviations

SIMD: Une instruction qui opère sur plusieurs données à la fois (*Single instruction, multiple data*)

MIMD: Un groupe d'instructions qui opèrent sur plusieurs données à la fois (*Multiple instructions, multiple data*)

HS ou SH: Harmoniques sphériques (*spherical harmonics*)

## Formules

Notez que certains acronymes anglais sont utilisées dans les formules de la revue littéraire afin qu'elles soient identiques aux formules données dans les ouvrages respectifs. Dans le texte, les acronymes français sont utilisées.

DI: Diffus avec interréflexion (*diffuse interreflected*)

DO ou DS: Diffus avec ombres (*diffuse shadowed*)

DSO ou DU: Diffus sans ombres (*diffuse unshadowed*)

SI ou GI: Spéculaire avec interréflexion (g*lossy interreflected*)

SO ou GS: Spéculaire avec ombres (*glossy shadowed*)

SSO ou GU: Spéculaire sans ombres (*glossy unshadowed*)

# GLOSSAIRE

Cette section présentera une liste de termes et définitions afin d'éviter toute ambiguïté dans le texte. Puisque souvent, en infographie, les termes originaux sont anglophones, ces termes seront donnés entre parenthèses après le terme francophone qui sera utilisé par la suite.

**Arbre k-dimensionnel (*kd-tree*):** Structure de données qui partitionne l'espace pour accélérer certaines recherches dans un ensemble quelconque de points (par exemple, le tracé de rayons).

**Crénelage (*aliasing*):** Effet visuel indésirable prenant la forme de dentelures, qui apparaît lors de l'affichage d'images courbes ou de diagonales, à cause d'une définition insuffisante de l'image ou de l'écran.

**Effet de bande (*banding*):** Défaut d'image qui se manifeste surtout lorsqu'il n'y a pas assez de précision pour représenter fidèlement un dégradé de couleur ou d'intensité.

**Graphe de scène (*scene graph*):** Structure hiérarchique d'une scène qui permet d'appliquer certaines optimisations lors du rendu temps réel, ainsi que de faciliter la manipulation de scènes très complexes. Un exemple d'un graphe de scène est OpenSceneGraph [Osfield2007].

**Nuanceur (*shader*):** Programme qui peut être exécuté par la carte graphique pour chaque primitive, chaque sommet ou chaque fragment (pixel) au cours d'un processus de rendu temps réel [Proudfoot2001][Rost2006].

**Ombrage (*shading*):** La réaction d'une surface à la lumière. En infographie, le terme anglais *shading* signifie les variations de couleur qu'on calcule à la surface d'un objet afin de lui donner un aspect tridimensionnel, en particulier par rapport à une ou plusieurs sources de lumière. Ce terme n'a pas d'équivalent direct en français; nous utiliserons donc le terme *ombrage* en ce sens, mais il ne faut pas alors présumer qu'il n'est question que de l'ombre portée d'un objet. Nous l'utiliserons dans le sens plus général. Voir la section **1.1 Ombrage et illumination** pour plus de détails.

**Sonde d'illumination (*light probe*):** Image représentant l'éclairage dans un environnement donné, dans toutes les directions. Souvent représentée soit sous forme de texture cubique où chaque face du cube représente une direction (devant, derrière, haut, bas, gauche, droite), ou sous forme de texture angulaire où les directions sont paramétrisées en termes de longitude et latitude comme pour une carte géographique. Les sondes d'illumination sont souvent acquises à l'aide d'une ou plusieurs photos d'une sphère de chrome, prises dans l'environnement dont on veut représenter l'éclairage [Debevec1998].

**Textures d'illumination (*light maps*):** Textures dans lesquelles l'éclairage provenant de sources de lumière statiques a été précalculé, et qui sont appliquées à la surface des objets statiques d'une scène comme éclairage de base [Segal1992]. Ceci permet d'avoir un éclairage réaliste sur ces objets sans faire de calculs complexes lors de la visualisation. On peut ensuite y ajouter l'effet de sources de lumière dynamiques, ainsi que les ombres portées par des objets dynamiques de la scène. Voir la section **2.1.2 Illumination précalculée** pour plus de détails.

**Tracé de rayons (*ray tracing*):** Processus de génération d'images qui consiste à lancer des rayons à partir d'un point de vue donné, à travers un certain nombre de points sur un *plan image*, puis dans la scène à représenter. L'interaction du rayon avec les surfaces dans la scène permettra de déterminer la couleur à placer à l'endroit où le rayon passait dans le plan image, qui deviendra l'image résultante. Voir la section **1.2 Techniques de génération d'images** pour plus de détails.

**Tramage (*rasterization*):** Processus de génération d'images qui consiste à afficher chaque primitive (polygone, par exemple) de chaque objet le plus rapidement possible. La plupart du temps, le tramage est effectué simplement en projetant les primitives de l'espace tridimensionnel à l'espace bidimensionnel, en déterminant celles qui devront être affichées dans l'image résultante (*clipping*) et en remplissant les polygones ligne par ligne (*scan conversion*). Voir la section **1.2 Techniques de génération d'images** pour plus de détails.

# INTRODUCTION

L'infographie permet, depuis quelques décennies déjà, de générer des images en simulant le comportement de la lumière à divers niveaux de réalisme. Les chercheurs en infographie font des efforts considérables pour que l'illumination soit représentée de la meilleure façon possible compte tenu des ressources disponibles (temps, puissance de calcul, mémoire). Les deux techniques les plus répandues de génération d'images sont le tracé de rayons et le tramage. La première a été longtemps reléguée à la génération d'images en différé, bien que les résultats soient de très grande qualité. La seconde, quant à elle, permet de générer des images à une vitesse suffisante pour permettre l'interactivité, et la qualité visuelle ne cesse de s'améliorer.

Depuis quelques années, on note une convergence des travaux de certains chercheurs vers un but ultime: la simulation de l'illumination globale, et donc du comportement réel de la lumière, dans des logiciels interactifs. Ceci présente plusieurs défis, car en illumination globale, par définition, l'apparence d'un objet peut dépendre de la scène en entier (objets, effets atmosphériques, éclairage). Deux approches ont été étudiées jusqu'à maintenant: simplifier le problème en retirant certains éléments, ou encore optimiser le processus à l'extrême.

## *MOTIVATION DE LA RECHERCHE*

Puisque les calculs d'illumination globale sont assez longs et complexes, une approche hybride est présentée dans plusieurs travaux des dernières années. Dans ce type

de système, on combine des calculs d'illumination globale faits par un engin de tracé de rayons à un engin temps réel basé sur le tramage. Ce faisant, il est alors possible de conserver une vitesse acceptable pour l'interaction avec la scène tout en ayant un rendu d'illumination globale plus ou moins complet.

Nous proposons, dans un premier temps, de découpler l'éclairage dans une scène de la réaction des surfaces à la lumière, l'encodant en harmoniques sphériques [Sloan2002]. Ceci fait en sorte que les calculs d'illumination restent valides après certaines opérations, telles la modification du point de vue et de l'éclairage. D'autre part, nous proposons de refaire les calculs d'illumination qui ne seraient plus valides, par exemple si un objet est déplacé, en parallèle à l'interaction de l'utilisateur [Tole2002]. Ceci permettra à l'utilisateur de continuer à interagir avec la scène, qui se mettra à jour progressivement.

Notre travail consiste donc à combiner ces deux techniques hybrides qui semblent à première vue complémentaires, notre hypothèse étant que le système qui en résultera offrira une qualité d'image et une performance intéressantes.

## *PLAN DU DOCUMENT*

Ce mémoire est composé de quatre chapitres. Le premier fera un survol de la génération d'images en présentant les bases du tracé de rayons et du tramage, ainsi qu'un survol des harmoniques sphériques. Le second chapitre tracera une trame de fond plus précise pour le travail effectué par une revue bibliographique et se terminera par l'énoncé des objectifs généraux et spécifiques du présent travail de recherche. Le troisième cha-

pitre présentera les idées utilisées pour valider les hypothèses données dans les objectifs et discutera de leur implantation. Finalement, le dernier chapitre présentera les résultats obtenus et l'interprétation de ceux-ci par rapport à nos objectifs de recherche.

# CHAPITRE 1

# DÉFINITIONS ET ANTÉCÉDENTS

## *1.1 OMBRAGE ET ILLUMINATION*

À la base, puisque la perception des images par le système visuel humain est basée sur la réaction des surfaces à la lumière, la génération d'images par ordinateur se doit de reproduire le comportement de la lumière et des surfaces, ou du moins l'apparence des surfaces sous différentes conditions d'illumination.

Il existe un grand nombre de modèles d'ombrage qui permettent de reproduire plusieurs caractéristiques des surfaces, soit de façon analytique (par exemple [Cook1981],[Poulin1990],[He1991],[Oren1994]) ou de façon empirique (par exemple [Lambert1760],[Phong1975],[Blinn1977]). Le modèle d'ombrage utilisé le plus souvent en infographie est un modèle empirique paramétrisable à trois composantes: un terme ambiant, un terme diffus et un terme spéculaire. L'ombrage peut dépendre de plusieurs facteurs tels:

- la visibilité de la source de lumière à partir d'un point de la surface;
- l'angle relatif entre les rayons lumineux et la surface;
- la position et l'angle de vue de l'observateur par rapport à la surface;
- les propriétés de la source lumineuse et de la surface;
- les propriétés d'autres surfaces.

La visibilité entre un point de la surface et la source de lumière détermine si ce point de la surface est dans l'ombre ou non, ou plus précisément si cette source de lumière contribue à l'illumination de ce point de la surface ou non. La distinction est importante, car l'illumination est additive. Par exemple, si trois sources de lumière éclairent directement une surface, celle-ci sera plus fortement illuminée que si l'une des sources de lumière est obstruée par un objet, c'est à dire que la surface est dans l'ombre d'un autre objet par rapport à l'une des sources de lumière.

L'angle entre les rayons lumineux et la surface détermine, en gros, la puissance de l'illumination à ce point. Lorsque les rayons arrivent perpendiculairement à la surface, celle-ci sera plus illuminée que lorsque les rayons arrivent à un angle rasant par rapport à la surface.

La position et l'angle de vue de l'observateur, eux, n'ont d'influence que sur le terme spéculaire. C'est pourquoi on dit que ce terme est *dépendant du point de vue*. La plupart des techniques de précalcul d'ombrage ne précalculent que les termes ambiant et diffus, car ils seront toujours valides même si le point de vue change, alors que le terme spéculaire dépend de celui-ci et donc devrait être recalculé trop souvent.

La surface peut avoir des propriétés telles une couleur, une texture ou une réflexivité données. La source de lumière peut également avoir une couleur ainsi qu'une intensité; souvent ces propriétés sont différentes pour chaque terme du modèle d'ombrage.

Finalement, une image peut considérer seulement les surfaces par rapport aux sources de lumière (calcul local de l'ombrage), ou considérer la scène en entier (calcul global de l'ombrage ou illumination globale). Dans ce dernier cas, l'ombrage d'une surface peut dépendre de toutes les autres surfaces dans la scène. Des effets comme des réflexions miroir ou des ombres sont la base de l'illumination globale, mais en général, quand on parle d'illumination globale, ça implique une interréflexion diffuse ou spéculaire entre les surfaces, et d'autres effets semblables.

## *1.2 TECHNIQUES DE GÉNÉRATION D'IMAGES*

En infographie, la génération d'images se fait généralement par deux techniques de base: le tracé de rayons et le tramage.

Le tracé de rayons permet de simuler le comportement de la lumière d'une façon aussi réaliste qu'on le désire, dans les limites de la puissance de l'ordinateur. C'est une technique intuitive et conceptuellement simple qui permet de reproduire des effets de façon réaliste. Le principe est le suivant: pour chaque pixel d'un plan image (qui deviendra l'image finale), on lance un rayon partant d'une caméra virtuelle et passant par ce pixel. S'il y a intersection avec un objet de la scène, l'objet le plus près du point de vue est celui qui est vu par la caméra dans ce pixel. On peut alors:

- calculer l'ombrage en utilisant un modèle d'ombrage de notre choix;
- lancer d'autres rayons, par exemple pour:
    - déterminer si ce point de l'objet est dans l'ombre d'un autre objet;

- trouver s'il y a une réflexion dans le cas d'une surface spéculaire;
- trouver s'il y a réfraction dans le cas d'une surface transparente;
- déterminer l'interréflexion à ce point de la surface en lançant d'autres rayons par rapport au type de surface.

Le tracé de rayons comporte plusieurs avantages intéressants. Il permet de représenter tout type d'objet, incluant des surfaces paramétriques et analytiques; il permet de représenter tout type de surface par le type de modèle d'ombrage choisi; il permet d'obtenir plusieurs effets simplement en lançant d'autres rayons.

Par contre, malgré sa puissance et sa flexibilité, cette technique demande beaucoup de ressources par pixel et peut difficilement exploiter la cohérence spatiale entre les rayons. Conséquemment, elle est surtout utilisée dans des applications de rendu différé, malgré que la tendance soit à essayer de la rendre interactive. Une caractéristique importante est que le tracé de rayons est facilement parallélisable, puisqu'il y a peu de dépendance entre des échantillons différents. C'est pourquoi la majorité des travaux de recherche sur le tracé de rayons en temps réel jusqu'à maintenant ont utilisé des ensembles d'ordinateurs interconnectés pour accélérer le processus. Par contre, les processeurs multi-coeur devenant de plus en plus répandus, cette caractéristique est maintenant exploitable sur un seul ordinateur.

Par opposition, le tramage est une technique rapide permettant d'obtenir un affichage à haut débit et normalement utilisé dans des applications interactives. Cette technique consiste à considérer chaque primitive (dont le type le plus couramment utilisé est

le polygone et plus spécifiquement le triangle) et à l'afficher le plus rapidement possible. Dans cette optique, plusieurs paramètres qu'il était possible d'utiliser pour le tracé de rayons sont éliminés ou simplifiés, et jusqu'à récemment, les librairies de programmation limitaient la paramétrisation du rendu. La venue des nuanceurs a changé cet état des choses, par contre.

La génération d'images par tramage, par défaut, ne considère qu'un modèle d'ombrage local. De plus, l'implantation de la majorité des effets n'est pas aussi intuitive qu'en tracé de rayons. Conséquemment, les effets implantés dans un rendu par tramage sont souvent des approximations, car on ne peut se permettre de reproduire le comportement réel de la lumière.

Prenons l'exemple du calcul des ombres portées par un objet dans une scène infographique, par exemple. Ce calcul peut se faire, entre autres, soit en évaluant la profondeur des objets à partir du tampon de profondeur à partir de la source de lumière, ce qui nécessite une passe de rendu additionnelle (*shadow map*), ou en créant des volumes d'ombres et en calculant le nombre d'entrées et de sorties de ces volumes (*shadow volume*).

Dans certains cas, une combinaison du tracé de rayons et du tramage est possible. Un exemple est l'utilisation de textures pour représenter l'ombrage d'objets statiques. Ces textures sont prégénérées par un logiciel externe par un processus de rendu tel que le tracé de rayons et sont ensuite utilisées dans un logiciel interactif de rendu par tramage pour représenter l'ombrage des objets statiques, auquel sont ajoutées les ombres

provenant d'objets dynamiques (personnages, etc.) à l'aide de l'une des techniques ci-haut. Par contre, ceci suppose que les scènes en question sont *totalement* statiques, c'est à dire que les objets qu'elles contiennent ne peuvent être ni transformés (translation, rotation, mise à l'échelle), ni déformés (percés, brisés, pliés), et que les sources de lumière par rapport auxquelles cette illumination a été calculée sont également statiques.

## *1.3 HARMONIQUES SPHÉRIQUES*

L'une des techniques formant la base de ce travail utilise les harmoniques sphériques, ce qui mérite une brève introduction.

La base des harmoniques sphériques (HS) est une base orthonormale sur une sphère $\boldsymbol{S}$. On peut faire analogie entre la transformation de Fourier sur un cercle et les harmoniques sphériques sur une sphère. On définit des fonctions de base en harmoniques sphériques pour un certain nombre de bandes $n$.

Le fait que la base des HS est orthonormale implique qu'une fonction scalaire $f$ peut être projetée en coefficients d'HS à l'aide d'une intégrale. Un signal de basse fréquence pourra être représenté de façon fidèle en peu de bandes d'HS. Par contre, un signal de haute fréquence sera limité en bande (lissé sans crénelage) lorsque projeté en trop peu de bandes.

### 1.3.1 PROPRIÉTÉS

La propriété la plus importante de la base des HS est son invariance rotationnelle. En pratique, ceci implique que la projection ne cause aucun crénelage lorsque les

points d'échantillonnage auxquels on a évalué la fonction $f$ ont subi une rotation. Cette propriété a des implications sur les opérations qu'il est possible de faire sans avoir besoin de recalculer les coefficients en HS [Sloan2002][Ivanic1996].

De plus, l'orthonormalité de la base implique que l'intégration d'un produit de fonctions dans la base des HS se réduit à un produit scalaire de leurs coefficients. Ceci sera important pour l'implantation des algorithmes de combinaison des fonctions d'illumination et fonctions de transfert dans notre travail.

# CHAPITRE 2

# REVUE HISTORIQUE ET LITTÉRAIRE

Ayant déjà résumé le sujet de la génération infographique d'images, nous allons maintenant nous attarder aux travaux de recherche ayant mené à l'état actuel des choses. Par la suite, les travaux qui forment la base du présent projet de recherche seront présentés et décrits en détail.

## *2.1 EFFETS DE BASE*

Tel que mentionné précédemment, il est possible de reproduire plusieurs facettes du comportement réel de la lumière par le tracé de rayons. Cette technique puissante a été la base d'un nombre impressionnant d'effets qui, tout en augmentant chaque fois la qualité et le réalisme des images, nous démontrent aussi l'élégance du concept du tracé de rayons.

### 2.1.1 ILLUMINATION GLOBALE

Le concept d'illumination globale s'applique lorsque l'apparence d'un objet dépend d'autres objets (à la limite, de tous les objets de la scène). En ce sens, même un effet simpliste comme les ombres peut être considéré comme de l'illumination globale. Par contre, il a fallu attendre [Kajiya1986], qui a présenté l'*équation du rendu*, avant de pouvoir parler d'une scène rendue complètement en illumination globale. D'après cette équation, la lumière provenant d'un point $x'$ atteignant un point $x$ dans la scène est composée à la fois de la lumière émise par le point $x'$ lui-même et de la lumière provenant d'un point

$x''$ et atteignant le point $x'$. Puisque ce dernier terme dépend de la même équation du rendu, il est impossible d'obtenir une solution analytique à cette équation pour tout point d'une scène dans le cas général, bien que ce soit possible dans certains cas très limités (par exemple [Goral1984]).

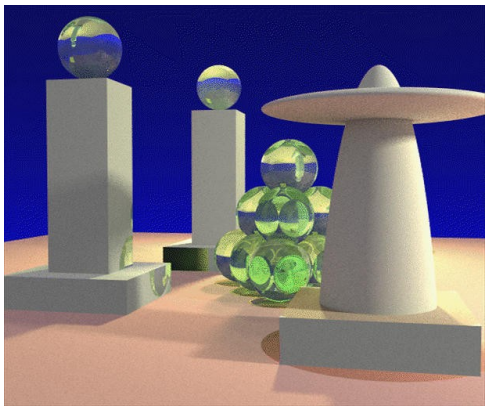


*Figure 1: Image rendue par tracé de chemins.* [Kajiya1986]

Dans le même article, on présente également une technique qui permet d'approximer une solution à l'équation du rendu par une méthode *Monte Carlo*. Cette technique, nommée tracé de chemins (*path tracing*) (Figure 1), a par la suite subi plusieurs améliorations [Lafortune1993] [Veach1997], mais c'est toujours un processus qui demande énormément de ressources (temps calcul, mémoire) avant de converger vers une image acceptable. Avec un nombre trop petit d'échantillons, l'image comportera des régions de bruit visuel.

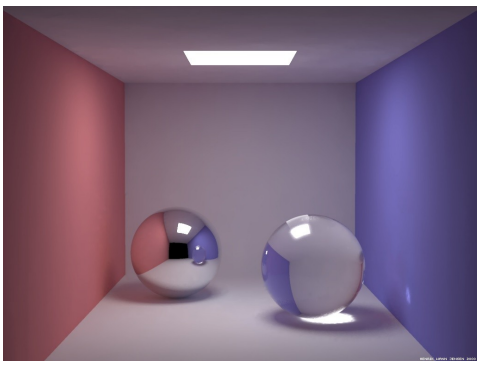


*Figure 2: Image générée par illumination globale par tracé de photons. [Jensen1996]*

D'autres stratégies ont été développées pour tenter de donner des solutions à l'équation du rendu. [Goral1984] tente de simplifier le problème en discrétisant la scène en éléments de surface et en ne considérant que les surfaces diffuses. [Jensen1996] tente de conserver les avantages et la généralité des techniques Monte Carlo tout en convergeant plus rapidement (Figure 2). Malheureusement, dans tous ces cas, l'approche directe pour générer une

image ainsi que le temps requis impliquent qu'on ne peut pas utiliser ces techniques en rendu temps réel.

### 2.1.2 Illumination précalculée

Le concept de précalculer de l'illumination complexe pour l'utiliser dans un rendu temps réel n'est pas nouveau lui non plus. [Segal1992] présenta l'idée d'utiliser des textures dans lesquelles ont été précalculés les effets de l'illumination, et qui sont appliquées à la géométrie comme éclairage de base. Ces textures, surnommées textures d'illumination, sont souvent générées à l'aide d'un engin de rendu basé sur la radiosité [Goral1984] et donnent généralement de bons résultats, mais encore une fois, la technique ne fonctionne que sous l'hypothèse que l'environnement est totalement statique.

### 2.1.3 Accélération du tracé de rayons

L'une des structures d'accélération spatiales traditionnellement utilisées en infographie est l'arbre k-dimensionnel (*kd-tree*) [Bentley1975]. Cette structure est généralement assez coûteuse à construire (comparativement à n'avoir aucune structure accélératrice), mais peut réduire considérablement le temps d'exécution du tracé de rayons, surtout pour les scènes distribuées non uniformément, une fois construite. Certains travaux ont été réalisés pour tenter de l'adapter à une scène constituée de géométrie dynamique ou semi-dynamique (voir par exemple [Bentley1990]), mais la structure de base est utilisée plus fréquemment.

Récemment, certains travaux ont montré qu'une hiérarchie de volumes englobants simples (sphères, boîtes orientées aux axes) peut être une structure d'accélération

valable pour le tracé de rayons, surtout dans des scènes dynamiques [Wald2007]. Les avantages sont une construction plus simple et moins coûteuse, la possibilité de faire des mises à jour plus fréquemment, ainsi qu'une plus grande stabilité à travers le temps (la structure n'étant pas strictement spatiale, mais également une hiérarchie d'objets distincts).

Nous verrons dans la section 3.1.1.2 comment ces deux méthodes d'accélération du tracé de rayons ont été utilisées dans notre travail.

### 2.1.4 Tracé de rayons interactif

Plusieurs travaux récents ont tenté de trouver un moyen de faire du tracé de rayons interactif. En général, ceux-ci se sont divisés en trois grandes catégories: optimisations bas niveau, optimisations des algorithmes à un niveau supérieur et méthodes de tampons.

#### 2.1.4.1 Optimisation bas niveau

La micro-optimisation des algorithmes de tracé de rayons consiste, par exemple, à prendre avantage des instructions opérant sur plusieurs données à la fois (SIMD et MIMD) ainsi qu'à porter une attention particulière à la localité des données afin d'optimiser l'utilisation des mémoires tampon des microprocesseurs (par exemple, [Wald2002] [Wald2003] et autres). Malgré une performance impressionnante, le travail d'optimisation doit souvent être refait lorsqu'une nouvelle génération de microprocesseurs voit le jour (ce qui est le cas à tous les deux ans en général) car les caractéristiques des mémoires tampon ainsi que les instructions de bas niveau de ceux-ci changent. Il est

d'ailleurs reconnu dans le domaine du génie logiciel que ce genre d'optimisation peut entraver les optimisations que les compilateurs peuvent faire automatiquement, et il est alors concevable que dans certains cas, la performance se dégrade (malgré le fait que les résultats semblent contredire cet argument).

#### *2.1.4.2 Optimisation des algorithmes*

À un plus haut niveau, certains travaux visent à changer ou à améliorer les algorithmes de rendu des images afin d'obtenir des vitesses interactives. Parmi ceux-ci, notons par exemple [Dmitriev2002], qui prend des échantillons aléatoires dans la scène afin de détecter les changements faits par l'utilisateur, ainsi que [Keller1997], qui tente d'améliorer la méthode de radiosité [Goral1984] afin de la rendre quasi-interactive. [Sloan2002] et [Ng2003] expriment l'illumination en utilisant la base des harmoniques sphériques et des ondelettes de Haar (respectivement), et arrivent ainsi à découpler l'illumination de la réflexion de la lumière sur les surfaces, et ainsi à la changer sans nécessiter un recalcul complet de la scène (voir également les travaux subséquents: [Sloan2003] [Sloan2003A][Sloan2005][Sloan2006] et [Ng2004]). [Sloan2002] fera l'objet de la section 2.2 du présent chapitre.

Nous incluons également dans cette catégorie les méthodes utilisant les processeurs graphiques programmables afin d'accélérer certaines parties du calcul, dont [Sloan2002] et [Ng2003] font partie, ainsi que [Ren2006] et la multitude de travaux visant à faire le tracé de rayons directement par le processeur graphique (par exemple [Purcell2003] et [Carr2006]).

À la base, ces travaux changent la façon dont nous approchons les algorithmes de tracé de rayons et d'illumination globale. Contrairement aux méthodes de la sous-section 2.1.4.1, nous prévoyons intuitivement que ces dernières pourraient avoir un impact important à long terme (ce qu'on ne peut affirmer de façon certaine, par contre).

#### *2.1.4.3 Méthodes de tampons*

Parallèlement à ces deux extrêmes, une autre stratégie a vu le jour, celle d'utiliser différents types de tampons afin de stocker certains résultats qui peuvent ensuite être réutilisés et mis à jour lorsque nécessaire. Certains de ces travaux feront l'objet de la section 2.3 de ce chapitre [Walter1999][Walter2002][Tole2002], mais il en existe d'autres (par exemple [Granier2004], [Pellacini2005] et [Hasan2006]).

Ces méthodes sont généralement plus complexes à généraliser, car un tampon stockant une certaine donnée sera efficace dans le cas d'une certaine d'utilisation mais pas une autre. Par exemple, le tampon de rendu [Walter1999] est utile pour des cas relativement simples (observateur fixe, petits changements dans la scène) mais la performance se dégrade rapidement lorsque des sources importantes d'illumination indirecte sont déplacées (par exemple, le plafond d'une pièce), car dans ce cas tous les échantillons du tampon deviennent invalides simultanément et doivent être recalculés. Le tampon d'illumination [Tole2002] donne une meilleure performance dans ce genre de cas car il permet de réduire le nombre d'échantillons par objet jusqu'à ce que la situation se stabilise; par contre, il peut difficilement gérer les objets ayant une composante spéculaire (réflexion miroir).

La méthode présentée dans le cadre de ce travail de recherche combine des éléments des techniques présentées dans les sous-sections 2.1.4.2 et 2.1.4.3. Il est en effet évident que l'optimisation de bas niveau n'était pas considérée cruciale à ce travail, malgré qu'une stratégie intéressante de tracé de rayons fut employée (voir section 3.1.1.2).

Ayant fait un survol rapide des techniques de base impliquées dans le présent travail, il est utile de s'attarder un peu plus sur les deux techniques formant la base du travail, soit le transfert de radiance précalculé et le tampon d'illumination. Voici une description détaillée de celles-ci.

## *2.2 TRANSFERT DE RADIANCE PRÉCALCULÉ*

[Sloan2002] présente le transfert de radiance précalculé (*Precomputed Radiance Transfer*). Nous présenterons dans cette section les grandes lignes de cette technique, qui est à la base du travail décrit dans le présent document.

À l'aide de cette technique, il est possible de simuler certains effets d'illumination globale, tels les ombres portées, les interréflexions de lumière sur un objet et la diffusion interne de la lumière dans des objets translucides. De plus, ces effets peuvent être calculés par rapport à l'illumination provenant d'une sonde d'illumination [Debevec1998] ou d'une source de lumière classique. Le résultat est une illumination qui semble très naturelle, en plus d'un objet d'apparence réelle et solide, plutôt que d'apparence "infographique". Par contre, hormis certaines modifications à la technique qui enlèvent des effets intéressants [Sloan2005], cette méthode ne s'applique pas à des objets déformables, ni à des environnements dynamiques où des objets se déplacent par rapport à d'autres. On ne

peut l'appliquer qu'à un seul objet rigide. De plus, la façon de stocker l'information du précalcul (la base des harmoniques sphériques, en l'occurrence) limite son utilisation à des situations d'éclairage de basse fréquence. En d'autres mots, l'illumination ne créera jamais d'ombres nettes, même lorsque l'éclairage de la scène est très intense.

### 2.2.1 PRÉCALCUL

Tel que son nom l'indique, la première étape nécessaire pour mettre en oeuvre le transfert de radiance précalculé est le précalcul des fonctions de transfert sur toute la surface de l'objet. Ceci peut être fait à l'aide d'un algorithme de tracé de rayons modifié pour stocker le transfert de radiance en espace objet en la représentant dans la base des HS (harmoniques sphériques). Ce précalcul est fait en projetant des rayons à partir de chaque sommet de l'objet dans un certain nombre de directions d'échantillonnage, formant une sphère autour du sommet. Le but est de simplement stocker la réaction de la surface à la lumière lorsque celle-ci provient d'une direction donnée, et non pas de calculer l'effet d'une situation d'illumination donnée en termes de couleur ou d'intensité lumineuse résultante.

Le transfert de radiance final, noté $L'_p$, doit être calculé différemment selon le type de transfert (diffus ou spéculaire[1]), car le transfert diffus se fait en fonction d'un vecteur normal fixe, ce qui n'est pas le cas pour un transfert spéculaire. Dans ce cas, on calcule le transfert de radiance en fonction d'un vecteur réfléchi variable (dépendant du point de vue de l'observateur). Le résultat est donc une valeur scalaire représentant la

1 On utilise le terme spéculaire, mais il faut se rappeler que nous travaillons en basse fréquence, et donc les réflexions nettes seront impossibles. Le terme *glossy* est utilisé en anglais plutôt que *specular* pour faire cette distinction.

radiance sortante dans le cas diffus, et une matrice de coefficients de projection pour la fonction sphérique en entier dans le cas spéculaire. Les équations sont:

$$L'_p = \sum_{i=1}^{n^2} (M_p^D)_i (L_p)_i \quad \text{(transfert diffus)} \qquad (1)$$

où

- n est le nombre de bandes utilisées dans l'encodage en harmoniques sphériques (n bandes donnent $n^2$ coefficients),
- $M_p^D$ représente le vecteur de transfert diffus
- $L_p$ est la radiance incidente au point *p* en supposant que l'objet *O* n'est pas dans la scène,
- $L'_p$ est le résultat dans le cas diffus, une valeur scalaire représentant l'illumination sortante au point *p*.

$$(L'_p)_i = \sum_{j=1}^{n^2} (M_p^G)_{ij} (L_p)_j \quad \text{(transfert spéculaire)} \qquad (2)$$

où

- n est le nombre de bandes utilisées dans l'encodage en harmoniques sphériques (n bandes donnent $n^2$ coefficients),
- $M_p^G$ représente la matrice de transfert spéculaire,
- $L_p$ est la radiance incidente au point *p* en supposant que l'objet *O* n'est pas dans la scène,

. $(L'_p)_i$ est le résultat dans le cas spéculaire, le coefficient *i* d'un vecteur de $n^2$ coefficients (toujours nommé $L'_p$) représentant l'illumination sortante au point p.

On note que dans le cas diffus, $M_p^D$ est un vecteur. Par contre, dans le cas spéculaire, $M_p^G$ est une matrice. De plus, alors que la radiance incidente $L_p$ est un vecteur de coefficients dans les deux cas, le résultat est une valeur scalaire dans le cas diffus et un vecteur de coefficients dans le cas spéculaire.

Pour définir les fonctions de transfert, [Sloan2002] considère la façon dont un objet *O* projette une ombre sur lui-même (*self-shadowing*) et comment la lumière est réfléchie d'une partie de l'objet à une autre (interréflexion). Il faut d'abord paramétriser l'illumination incidente aux points $p \in O$, notée $L_p(s)$, dans la base des harmoniques sphériques (HS). Ici, *s* représente un échantillon sphérique. Afin de pouvoir calculer les valeurs d'illumination en HS, il faut déterminer un nombre cible d'échantillons par point $p \in O$, qui doivent être de préférence uniformément espacés afin de donner un bon échantillonnage sphérique.

L'illumination incidente est donc représentée par un vecteur de $n^2$ coefficients, notés $(L_p)_i$. Les fonctions de transfert sont ensuite définies en prenant compte des effets que nous voulons obtenir.

La fonction de transfert la plus simple

$$T_{DU}(L_p) = (\rho_p / \pi) \int L_p(s) H_{N_p}(s)\, ds\,, \tag{3}$$

permet un transfert diffus sans ombre, où

- $s$ représente une direction d'échantillonnage sphérique,
- $\rho_p$ représente l'albedo de l'objet au point $p$,
- $L_p$ est la radiance incidente au point $p$ en supposant que l'objet $O$ n'est pas dans la scène,
- $N_p$ est la normale au point $p$ et
- $H_{N_p}(s)=max(N_p \cdot s, 0)$ est le noyau cosinus hémisphérique autour de $N_p$.

En se référant aux fonctions de transfert finales (équations 1 et 2), on voit qu'en projetant $L_p$ et $H_{N_p}$ séparément, on peut ensuite calculer le résultat en faisant un produit interne de leurs vecteurs de coefficients. Les facteurs résultants sont appelés la fonction illumination, $L_p$ et la fonction transfert, $M_p$. Dans ce cas simple, $M_p^{DU}(s)=H_{N_p}(s)$.

Une autre fonction

$$T_{DS}(L_p)=(\rho_p/\pi)\int L_p(s)\,H_{N_p}(s)\,V_p(s)\,ds\,, \qquad (4)$$

permet un transfert diffus avec ombres, où la fonction $V_p(s)\rightarrow\{0,1\}$ indique la visibilité ($V_p(s)$ = 1 si un rayon dans la direction $s$ ne réintersecte pas l'objet $O$) et les autres termes sont les mêmes que dans l'équation 3. Comme dans le cas précédent, on décompose le calcul en deux fonctions, la fonction illumination $L_p$ et la fonction transfert $M_p^{DS}(s)=H_{N_p}(s)V_p(s)$.

La fonction

$$T_{DI}(L_p) = T_{DS}(L_p) + (\rho_p / \pi) \int \bar{L}_p(s) H_{N_p}(s)(1 - V_p(s)) ds \qquad (5)$$

permet un transfert diffus avec ombres et interréflexions diffuses, où $\bar{L}_p(s)$ dénote la radiance de *O* lui-même dans la direction *s* et les autres termes sont les mêmes que dans les équations 3 et 4. Ceci est difficile à évaluer dans le cas général, car nous ne pouvons connaître cette valeur étant donné seulement la radiance incidente au point *p* que si la source de lumière est infiniment éloignée. Dans d'autres cas, $\bar{L}_p(s)$ peut dépendre de la radiance sortante à plusieurs points sur *O* et cette radiance peut varier localement. En supposant que la variation locale est petite, $\bar{L}_p(s)$ peut être approximé en supposant que *O* est illuminé uniformément par $L_p$. Il est alors possible de factoriser comme dans les deux cas précédents en deux fonctions projetées, la fonction illumination et la fonction transfert.

Noter que la fonction de transfert pour le transfert diffus avec ombres (mais pas d'interré-flexions) ne nécessite pas de supposer que l'illumination incidente varie peu. La raison est que cette fonction de transfert ne dépend que de l'illumination incidente, alors que celle pour les interréflexions dépend de plusieurs points $q \neq p$ pour lesquels $L_q \neq L_p$. Dans le cas sans interréflexions, il suffit donc d'échantillonner l'illumination incidente avec une précision suffisamment fine pour capturer la variation locale, ce qui donnera un résultat correct.

Pour le cas spéculaire, les fonctions de transfert dépendront d'une fonction *G(s,R,r)* plutôt que d'une fonction $H_{N_p}$, où *s* est la direction d'échantillonnage, *R* est le

vecteur réfléchi (dépendant du point de vue) et *r* est le degré de spécularité de la surface, tel que défini dans les équations du modèle d'illumination de Phong [Phong1975]. Des fonctions analogues aux trois fonctions de transfert données pour le cas diffus peuvent être définies ainsi:

$$T_{GU}(L_p, R, r) = \int L_p(s) G(s, R, r)\, ds \tag{6}$$

$$T_{GS}(L_p, R, r) = \int L_p(s) G(s, R, r) V_p(s) ds \tag{7}$$

$$T_{GI}(L_p, R, r) = T_{GS}(L_p) + \int \bar{L}_p(s) G(s, R, r)(1 - V_p(s))\, ds \tag{8}$$

où les termes sont les mêmes que dans les équations 3, 4 et 5.

Ces fonctions ne sont plus seulement dépendantes de la direction d'échantillonnage *s*, ce qui implique qu'elles ne peuvent plus être décomposées en vecteurs de composants dans la base des HS. Une solution est de paramétriser la radiance incidente $L_p$ sur une sphère, ce qui sera noté $L'_p$. En supposant alors que le noyau *G* est circulairement symétrique autour de *R*, c'est à dire qu'il représente un modèle d'illumination semblable à celui de Phong, $L'_p(s)$ peut être convolué avec la fonction $G_r(s) = G(s, (0,0,1), r)$ et évalué à *R* pour obtenir le résultat final.

Ceci donne alors une matrice de transfert plutôt qu'un vecteur. Par exemple, la fonction transfert pour le cas spéculaire avec ombres est le suivant:

$$M_p^{GS}(L_p, s) = L_p(s) V_p(s)\ . \tag{9}$$

### 2.2.2 VISUALISATION

*Figure 3: Exemple de visualisation des fonctions de transfert avec ombres et interréflexions [Sloan2002]*

Par la suite, la visualisation peut être effectuée utilisant les résultats stockés lors du précalcul pour calculer l'illumination sur l'objet en tenant compte de la situation d'illumination voulue. Un environnement prédéfini (*environment map*) [Blinn1976] sera souvent utilisé pour donner un aspect très naturel à l'illumination de la scène. Cette illumination est utilisée pour calculer la couleur à un point de la surface étant donné le précalcul fait précédemment (voir les Figures 3 et 4).

Les étapes à suivre lors de la visualisation sont:

1. calculer l'illumination incidente $L_{p_i}$ à un ou plusieurs points d'échantillonnage dans le voisinage de *O* dans la base des HS.
2. Effectuer une rotation de ces échantillons vers l'espace de coordonnées de *O* et les moduler ensemble pour produire le champ d'éclairage incident $L_p$ sur *O*.
3. Effectuer une transformation linéaire sur $(L_p)_i$ à chaque point *p* de *O* pour obtenir la radiance sortante. Ceci nécessite un produit scalaire avec $(M_p^D)_i$ pour les surfaces diffuses ou la multiplication d'un vecteur avec la matrice $(M_p^G)_{ij}$ pour les surfaces spéculaires.

4. Les surfaces spéculaires nécessitent une étape supplémentaire: le vecteur résultant de l'étape 3 est convolué avec les propriétés de la surface de *O* au point *p* puis évalué par rapport au vecteur réflexion *R* (dépendant du point de vue).

À l'étape 1, l'illumination peut être générée soit en utilisant une sonde d'illumination, un modèle analytique basé sur la physique de la lumière, un modèle empirique tel que celui de Phong/Blinn, ou une combinaison de ces méthodes.

La rotation à l'étape 2 est faite une fois par objet, et non pas pour chaque *p* indépendamment, car le résultat du précalcul a été stocké dans l'espace objet de *O*, ce qui fait en sorte qu'il est plus efficace d'effectuer la rotation sur les échantillons $L_{p_i}$ plutôt que sur toutes les fonctions de transfert précalculées de *O*.

*Figure 4: Exemples de transfert diffus et spéculaire [Sloan2002]*

Pour le transfert diffus (voir l'image de gauche de la Figure 4), l'étape 3 est effectuée dans un nuanceur. La façon la plus simple de faire est de stocker les vecteurs de transfert par sommet et faire les calculs dans un nuanceur de sommets. Les résultats de ce calcul seront interpolés sur le polygone automatiquement (méthode de Gouraud). Lorsque l'environnement est coloré ou qu'on veut une simulation de l'interréflexion diffuse colorée entre les surfaces, on doit faire ce travail en trois passes, une pour

chacune des composantes (rouge, vert et bleu), car le nombre de données requises pour trois bandes dépasse ce qu'on peut accéder dans un nuanceur; sinon, une passe suffit.

Dans le cas du transfert spéculaire (voir l'image de droite de la Figure 4), l'étape 3 ne peut pas présentement être effectuée dans un nuanceur car la matrice de transfert est trop volumineuse pour être manipulée dans les nuanceurs de la génération en vigueur au moment de l'écriture de [Sloan2002] (GeForce série 5x00). Par contre, la convolution et l'évaluation à l'étape 4 peuvent être faites dans un nuanceur.

Après avoir fait une revue détaillée du transfert de radiance précalculé, nous ferons de même pour la seconde technique qui forme la base du travail présenté dans ce document, soit le tampon d'illumination.

## *2.3 TAMPON D'ILLUMINATION*

Deux systèmes ont été développés à l'université Cornell permettant d'intégrer le rendu différé (en illumination globale ou non) à un affichage temps réel. Le premier système est nommé le tampon de rendu (*Render Cache*) [Walter1999] [Walter2002] et fonctionne en prenant des échantillons de la scène à l'aide d'un engin de rendu différé (tracé de rayons, Monte Carlo) puis, lorsque l'utilisateur interagit avec le système (déplacement du point de vue ou d'objets), il reprojette les points existants qui avaient été sauvegardés dans le tampon de rendu, puis détecte les changements et refait le rendu où c'est nécessaire seulement. La qualité d'image est généralement bonne, mais lors d'interactions avec le système, une image approximative est affichée pendant qu'une partie de l'image est recalculée, ce qui peut prendre plusieurs secondes, voire quelques minutes.

Le second système développé par ce groupe se nomme le tampon d'illumination (*Shading Cache*) [Tole2002] et vise à corriger les lacunes du tampon de rendu. Dans ce système, le matériel graphique et les techniques classiques du rendu temps réel s'occupent de faire le calcul et l'affichage de l'illumination directe, alors qu'un engin de rendu différé fait les calculs d'illumination globale, par la technique du tracé de chemins bidirectionnel, en parallèle lorsque des changements sont effectués dans l'image (déplacement du point de vue ou d'objets). Ceci est beaucoup plus efficace, car l'interaction se fait toujours à une vitesse acceptable alors que les résultats des calculs d'illumination globale sont affichés à mesure qu'ils sont disponibles. L'image résultante présente donc toujours une illumination locale complète, même immédiatement après une interaction, et l'illumination indirecte (globale) se met à jour graduellement. Ce système est particulièrement utile pour la prévisualisation et la conception d'éclairages, car l'utilisateur peut se déplacer dans l'environnement ou déplacer des objets et des sources de lumière et avoir le résultat du calcul d'illumination globale quelques secondes plus tard. Il peut ensuite se déplacer encore une fois pour voir l'environnement d'un autre point de vue et seuls les calculs nécessaires à la mise à jour de l'image auront à être refaits.

En plus d'être le nom de la technique, le terme tampon d'illumination désigne également la structure de données utilisée. Il est décrit comme un maillage hiérarchique en espace objet dans lequel les échantillons de haute qualité (les résultats du rendu d'illumination globale sur l'objet) sont stockés. Chaque élément du maillage (*patch*) contient les échantillons calculés à ses sommets. Les auteurs précisent que la structure développée

supporte les triangles, les quadrilatères et les surfaces bicubiques, ce qui peut vouloir dire qu'il faut adapter la structure de données à la représentation des objets de la scène.

Lors de l'initialisation du logiciel, chaque primitive de chaque objet de la scène se fait attribuer une telle structure de données ne contenant qu'un seul élément de maillage non subdivisé. Lors de la mise à jour de l'image, les échantillons sont calculés aux sommets de l'élément de maillage et le système détermine, par exemple en examinant la variance entre les échantillons d'un même élément de maillage, si celui-ci doit être subdivisé pour raffiner l'image.

Lors de l'affichage de la scène, les données du tampon d'illumination sont transférées à un tampon de sommets standard qui est affiché directement. Le transfert entre le processus d'échantillonnage et le processus d'interaction ne se fait qu'à certains moments (par exemple, lorsqu'il y a eu un changement significatif) afin de ne pas avoir à synchroniser les deux processus outre-mesure, car le but est de découpler l'interaction du rendu des échantillons d'illumination globale.

Les échantillons sont conservés en espace objet afin qu'ils puissent être réutilisés dans des images subséquentes, même si à un moment donné cette partie de l'objet n'est pas visible. Ceci fonctionne bien pour l'illumination diffuse, car elle est indépendante du point de vue, mais pour l'illumination spéculaire, même un changement très minime du point de vue fera en sorte que l'échantillon n'est plus valide et doit être recalculé.

La méthode du tampon d'illumination décrite dans [Tole2002] utilise un système de priorités assez sophistiqué afin de faire en sorte que l'effort de rendu soit concentré

dans les régions où il y a plus de changement dans l'image. En gros, avant de faire le rendu d'un échantillon, sa priorité est calculée à partir de la variance entre les échantillons voisins pondérée par le nombre de pixels que l'élément de maillage occupe dans l'image. L'échantillon est ensuite mis dans une queue à la position correspondant à sa priorité. Le processus de rendu prend les échantillons un par un, en ordre de priorité, et stocke les résultats dans l'élément de maillage correspondant.

Lorsqu'il y a interaction, les échantillons existants sont conservés et le système rééchantillonne ce qui a changé. La Figure 5 illustre ce processus.

- Si la caméra se déplace, seuls les objets ayant une composante spéculaire doivent être mis à jour car l'illumination diffuse est indépendante du point de vue. La priorité des échantillons appartenant à des surfaces spéculaires est augmentée et ces échantillons sont remis dans la queue.
- Si un objet a été déplacé, le problème est plus complexe, car en illumination globale, l'illumination de tout objet dans la scène peut dépendre de tous les autres objets. Ainsi, par exemple, si le plafond d'une pièce est déplacé, la scène au complet devra très probablement être recalculée, car le plafond est une source importante d'illumination indirecte. La stratégie employée pour tenir compte des déplacements est de prendre un certain nombre d'échantillons aléatoires dans l'image à chaque mise à jour. Si l'illumination de la scène a fortement changé, ces échantillons auront un résultat très différent des échantillons voisins (qui n'ont pas été mis à jour depuis l'interaction) et donc leur priorité sera grandement augmen-

tée. L'effet est que de tels changements seront détectés rapidement et l'augmentation de priorité fera en sorte que les régions où il y a le plus de changement seront recalculées le plus tôt possible.

Le principe de raffinement du maillage des objets fonctionne à l'inverse également. Lorsqu'une situation comme celle dans l'exemple précédent (le déplacement du plafond) se produit, le système verra qu'un trop grand nombre d'échantillons doivent être calculés afin de ramener la qualité d'image à un niveau acceptable. La résolution des maillages concernés sera diminuée jusqu'à un niveau acceptable afin que moins de travail soit nécessaire pour améliorer la qualité d'une grande portion de l'image rapidement et le processus de subdivision et de raffinement des maillages reprendra, une fois qu'une certaine stabilité aura été rétablie, afin de capturer des détails plus fins et subtils.

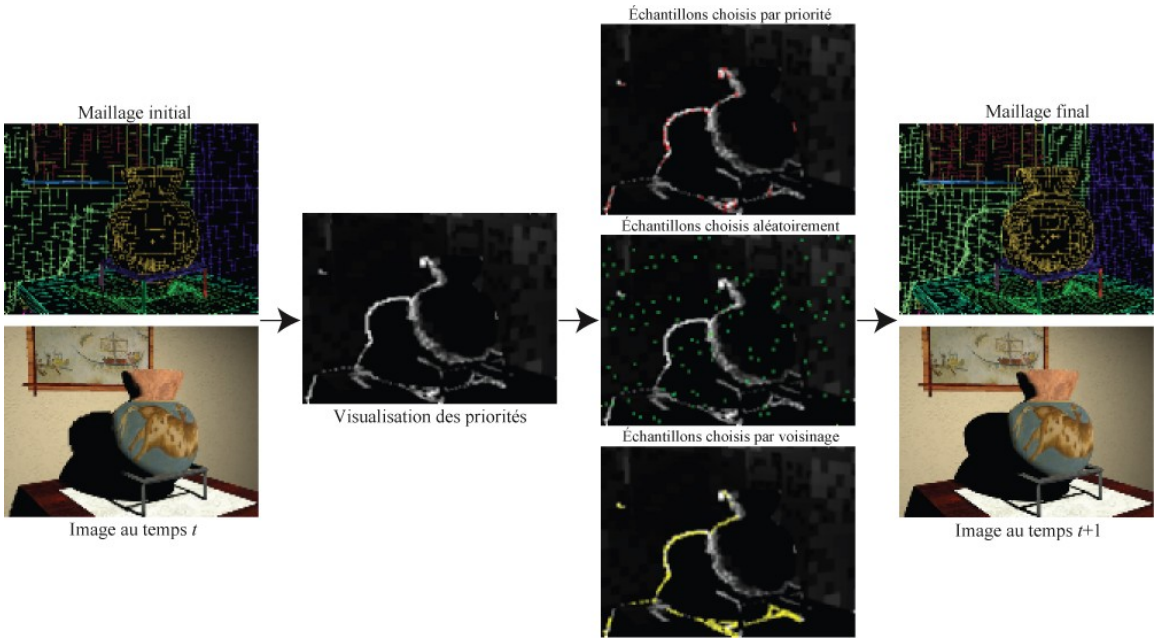



*Figure 5: Exemple de la sélection des échantillons et du raffinement du maillage du tampon de rendu [Tole2002] (légende des images traduites)*

## 2.4 *OBSERVATIONS*

La revue détaillée des travaux dans les sections 2.2 et 2.3 nous permet de faire certaines observations qui permettront de mieux cerner les objectifs de la présente recherche.

1. Le tampon d'illumination permet de tirer avantage à la fois d'un rendu temps réel, pour la vitesse d'interaction, et d'un rendu en différé (en l'occurrence, un tracé de chemins bidirectionnel), pour la qualité d'image. Il permet à l'utilisateur de faire des modifications à la scène en refaisant les calculs nécessaires pour remettre à jour l'illumination de haute qualité en parallèle à l'interaction.

2. La fonction de calcul du transfert de radiance précalculé (dans son mode diffus avec ombres) n'est pas un tracé de rayons complet, car il ne considère pas les propriétés des surfaces. Ce n'est, en fait, qu'un test de visibilité encodé de façon sphérique à chaque sommet. En théorie, donc, il devrait être moins coûteux à faire (en termes de temps de calcul) qu'un tracé de rayons.

3. Le calcul est fait en espace objet, ce qui fait en sorte que tous les points de vue sont calculés plutôt que seulement ce qui est visible à partir de la position présente de l'observateur comme dans un tracé de rayons normal. Ceci implique que le calcul est valide même si la position de l'observateur change, en autant que seuls des effets d'illumination diffuse (donc indépendante du point de vue) sont représentés.

4. L'encodage utilisé a certaines propriétés qui font en sorte que dans certains cas, une modification à la scène est possible sans avoir à recalculer quoi que ce soit. Dans le cas de la technique originale, tout changement impliquait un certain re-calcul, et certains changements simples pouvaient engendrer le recalcul complet de l'image.

## *2.5 OBJECTIFS GÉNÉRAUX ET SPÉCIFIQUES DE LA RECHERCHE*

### 2.5.1 OBJECTIF GÉNÉRAL

Ayant une vision plus claire de ce qui a été fait dans le passé, ainsi que de la terminologie en cause, nous pouvons maintenant présenter les objectifs de la recherche en termes plus précis. Le but général de la recherche est de permettre un rendu d'illumination globale en temps réel dans des scènes dynamiques. Dans ces termes, *illumination globale* signifie un rendu avec interréflexion diffuse, réflexion spéculaire, ombres, caustiques, et autres effets ne dépendant pas seulement de l'environnement local d'un objet; *temps réel* implique une vitesse permettant l'interaction, soit un minimum de 30Hz; et *scènes dynamiques* indique que l'utilisateur a le contrôle total pour modifier ce qu'il veut dans la scène, sans restriction.

### 2.5.2 OBJECTIFS SPÉCIFIQUES

Dans le but de s'approcher de cet objectif de manière incrémentale, nous émettons l'hypothèse selon laquelle la technique de précalcul de l'illumination de [Sloan2002] (le transfert de radiance précalculé) serait une alternative intéressante au tracé de rayons

traditionnel dans le contexte d'un système d'affichage temps réel avec rendu différé tel que celui de [Tole2002] (le tampon d'illumination). Puisque ces deux techniques permettent de faire du rendu d'illumination globale à divers degrés, et que les deux ont une composante temps réel, on peut espérer qu'une telle combinaison nous permettra d'avancer vers le but décrit plus haut.

Voici une division plus précise du travail de recherche pour montrer le travail à effectuer:

Transfert de radiance précalculé :

- Faire un précalcul de l'illumination afin de représenter un ou plusieurs types de transfert lumineux.
- Offrir la possibilité d'effectuer une rotation des coefficients en harmoniques sphériques (ce qui est l'une des opérations qui peut être faite sans nécessiter un recalcul, plus de détails plus loin).

Tampon d'illumination :

- Refaire les calculs d'illumination lorsque la situation change, en faisant encore une fois appel à la technique de rendu du transfert de radiance précalculé.

### 2.5.3 Quantification des objectifs

Dans l'optique de s'assurer que nous pouvons vérifier notre hypothèse, voici certains critères qui devront faire partie de notre expérimentation. Les trois critères principaux seront la vitesse d'interaction, le temps de mise à jour et la qualité d'image.

1. Il est d'abord important que la vitesse d'interaction (en termes d'images par seconde) reste constante, peu importe s'il y a un recalcul en cours ou non. Une différence de moins de 5% entre la vitesse au repos et la vitesse lors d'une mise à jour sera jugée acceptable.

2. Un temps de mise à jour de moins de 15 secondes pour une scène relativement simple d'environ 10 000 sommets sera jugé raisonnable suite à une modification de la scène par l'utilisateur (déplacement ou déformation d'un objet, par exemple).

3. Notre travail se concentre surtout sur les ombres. Nous dirons donc qu'une image donnant une bonne impression des ombres (c'est à dire, de la provenance de la lumière et de la forme des objets qui bloquent celle-ci) avec un minimum d'artéfacts visuels (crénelage, effet de bande) sera acceptable. De plus, l'application de la technique présentée ici ne devrait pas limiter les effets de rendu d'illumination locale. Elle devrait se combiner à ceux-ci, afin que les effets visuels qu'il est possible de représenter en temps réel de nos jours restent toujours applicables.

### 2.5.4 Éléments exclus de la recherche

Certains aspects exposés dans les deux articles ne font pas partie de nos objectifs de recherche, et nous tenons à préciser ce point. L'amélioration des techniques d'illumination constitue notre objectif premier en faisant ce travail de recherche.

En particulier, le mécanisme de subdivision adaptative de maillages employé par le tampon d'illumination ne sera pas utilisé dans notre prototype car les objectifs de la recherche n'incluent pas la construction/subdivision de géométrie.

De plus, afin de concentrer l'effort là où il sera le plus utile, nous ne considérerons que le cas diffus avec ombres dans le cas du transfert de radiance précalculé. Un transfert spéculaire impliquerait:

a) des calculs additionnels lors de la visualisation du transfert de radiance précalculé;

b) le recalcul d'un grand nombre de fonctions de transfert chaque fois que le point de vue change lors de la visualisation.

On conviendra ainsi que le transfert diffus avec ombres sera suffisant pour démontrer notre hypothèse.

Pour la même raison, nous ne considérerons que deux types de sources de lumière dans nos calculs: les sources directionnelles et les sondes d'illumination. Ces deux types simulent un éclairage provenant de l'infini. Les sources de type point et spot, étant des sources de lumière positionnelles, présentent des problèmes lors de l'implantation du transfert de radiance précalculé, et n'apportent rien de plus par rapport à notre hypothèse.

# CHAPITRE 3

# MÉTHODOLOGIE ET IMPLANTATION

## *3.1 IMPLANTATION*

Nous avons utilisé les idées exposées dans [Sloan2002] et [Tole2002] pour démontrer la validité de nos hypothèses. Le présent chapitre donne notre interprétation de ces idées, ce qui s'est traduit par une implantation que nous avons pu utiliser pour expérimenter.

### 3.1.1 IMPLANTATION DU TRANSFERT DE RADIANCE PRÉCALCULÉ

L'implantation du transfert de radiance précalculé se divise en deux parties: le précalcul et la visualisation. Voici le processus pour chacune des deux parties.

#### *3.1.1.1 PRÉCALCUL*

Le précalcul est fait selon les étapes suivantes (voir [Green2003] pour une explication détaillée de l'implantation d'un générateur d'illumination en harmoniques sphériques) :

1. Le générateur de coefficients en harmoniques sphériques est initialisé avec les paramètres donnés sur la ligne de commande, et les directions d'échantillonnage sont générées de façon aléatoire mais uniforme.
2. La scène ainsi que l'environnement d'éclairage prédéfini sont chargés.

3. Les coefficients d'éclairage ($L_p$) sont calculés. Ces coefficients sont ensuite assignés à la scène en tant que variables uniformes (quatre coefficients sont combinés dans un vecteur à quatre composantes).

4. Partant de la racine du graphe de scène chargé à l'étape 2, à chaque objet,

    a) Pour chaque sommet, les coefficients de la fonction de transfert voulue sont calculés et ces coefficients sont ensuite assignés en tant qu'attributs de sommet (encore une fois, quatre coefficients sont combinés dans un vecteur à quatre composantes).

    b) De plus, on effectue une rotation des coefficients d'éclairage par rapport à la transformation locale de l'objet. Ceci permet de tenir compte de la hiérarchie du graphe de scène afin de conserver des objets distincts, plutôt que de limiter le traitement à un seul objet statique et rigide.

5. Les coefficients des fonctions de transfert sont par la suite sauvegardés sur le disque pour chaque objet, afin de pouvoir les réutiliser la prochaine fois que le programme sera démarré, car ceci constitue la situation initiale, avant que survienne toute interaction de l'utilisateur.

La Figure 6 présente l'étape 4.a) en détail, sous forme d'un diagramme de séquence.

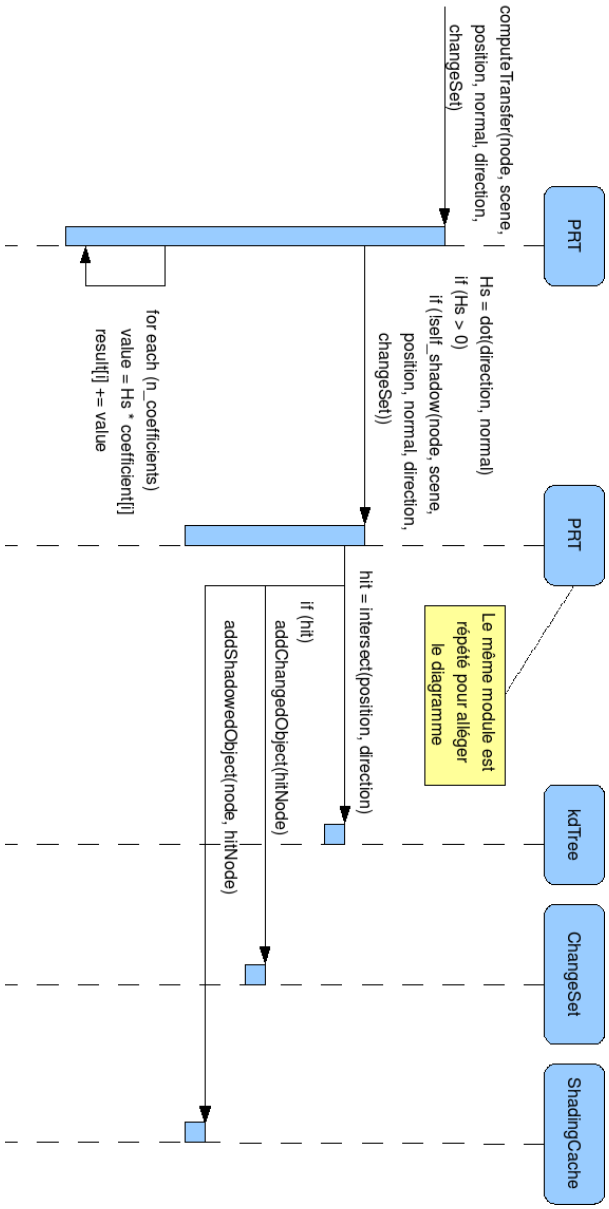



*Figure 6: Diagramme de séquence présentant le processus de calcul des coefficients des fonctions de transfert*

#### *3.1.1.2 TRACÉ DE RAYONS*

Dans la section 2.1.3, nous avons présenté deux méthodes d'accélération du tracé de rayons. Notre travail étant centré sur les scènes dynamiques, notre algorithme de tracé de rayons devait être conçu en conséquence.

Dans notre cas, puisque le rendu temps réel est basé sur un graphe de scène, la création et la mise à jour de la hiérarchie de volumes englobants est déjà faite, et il est facile de faire des tracés de rayons à travers cette hiérarchie (dans notre cas, en utilisant des tests d'intersection entre un rayon et le volume englobant, qui sont très peu coûteux).

Nous avons donc opté pour une stratégie hybride. Nous construisons un kd-tree pour chacun des noeuds terminaux du graphe de scène, en espace objet. Les rayons traversent d'abord la hiérarchie de volumes englobants, et lorsque l'on détermine qu'un rayon fait intersection avec un noeud terminal du graphe, le rayon est transformé par l'inverse de la transformation du noeud, ce qui le ramènera en espace objet pour ce noeud, et il sera testé contre le kd-tree. Ceci a plusieurs avantages:

- Les rayons ne faisant intersection avec aucun objet peuvent être rejetés rapidement grâce aux tests entre rayons et volumes englobants, qui sont très peu coûteux.
- La hiérarchie des volumes englobants donne généralement une bonne localité des volumes, ce qui veut dire que les rayons n'auront pas à aller très bas dans la hiérarchie avant d'être rejetés, sauf s'ils doivent vraiment intersecter l'un des noeuds terminaux.
- La transformation d'un objet n'entraîne pas la mise à jour (dans notre cas, la reconstruction) du kd-tree pour cet objet, puisque le kd-tree est en espace objet.
- Les tests d'intersection entre un polygone et un rayon sont très rapides, car ils utilisent le kd-tree.

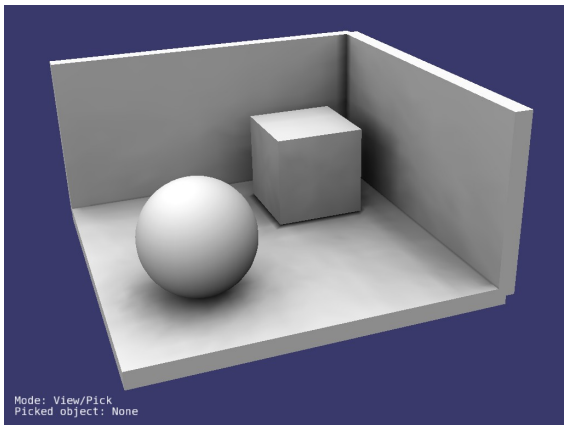
Le désavantage est que dans des scènes très complexes, le kd-tree est très efficace pour les tests d'intersection. En ayant potentiellement à tester plusieurs volumes englobants, puis en séparant les kd-trees des objets distincts, on pourrait avoir à faire plus de tests (la hiérarchie de volumes englobants convergeant moins rapidement vers les polygones précis que le rayon intersecte que le kd-tree) et à tester plusieurs kd-trees dans le cas où leurs volumes englobants coïncident dans l'espace. Néanmoins, le fait de ne pas avoir à faire la mise à jour d'un seul grand kd-tree pour toute la scène est un avantage significatif.

#### *3.1.1.3 VISUALISATION*

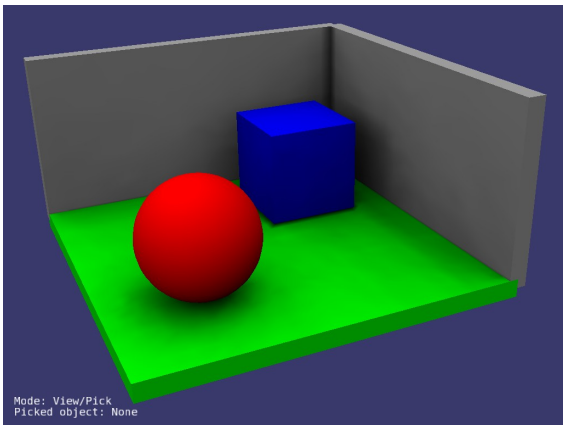
L'engin de visualisation est, à la base, un engin 3D relativement simple. Il permet d'observer la scène et de faire certaines modifications qui seront décrites plus loin. La visualisation des résultats du transfert de radiance précalculé se fait en utilisant un nuanceur de sommets qui, pour chaque sommet, effectue le produit scalaire entre les coefficients d'éclairage et les coefficients de la fonction de transfert à ce sommet. Dans le cas présent, puisque nous ne considérons que le transfert diffus avec ombres, le résultat de ce produit scalaire nous donnera une valeur entre 0 et 1 représentant la quantité de lumière atteignant ce sommet.

Par la suite, ayant cette information, on peut déterminer la couleur d'un point à la surface de l'objet dans un nuanceur de fragments. La valeur calculée par le nuanceur de sommets sera interpolée automatiquement à la position du fragment. Dépendant du résultat désiré, le nuanceur de fragments peut faire tout type de calcul. Par exemple:

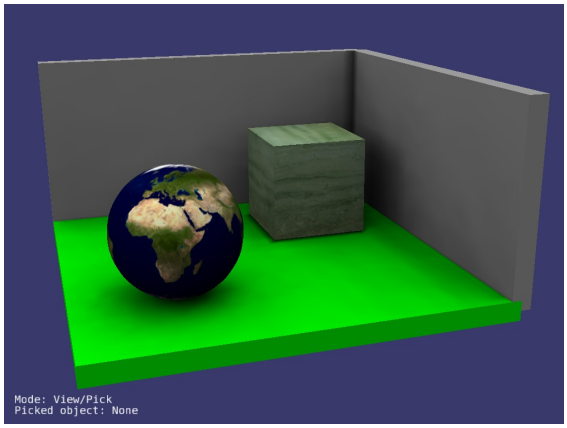
- Simplement afficher la valeur, ce qui donne un rendu monochromatique en dégradés de gris (Figure 7a).
- Afficher la valeur multipliée par la couleur diffuse de la surface (Figure 7b).
- Afficher la valeur combinée à une texture appliquée sur la surface de l'objet (Figure 7c).
- Utiliser l'une des options précédentes, plus un terme d'éclairage basé sur la même image qui a servi à faire le précalcul, à la fois pour les surfaces diffuses et pour les surfaces spéculaires. Ceci donne un aspect plus naturel, car les objets semblent faire partie de leur environnement, mais en même temps, le résultat du transfert de radiance précalculé est plus difficile à voir (Figure 7d).



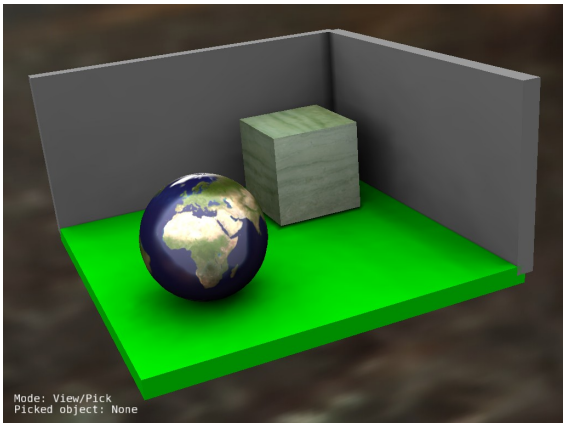

*a) Résultat du calcul du transfert de radiance pré-calculé.*




*b) Transfert de radiance précalculé plus couleur diffuse.*




*c) Transfert de radiance précalculé plus couleur diffuse plus textures.*




*d) Transfert de radiance précalculé plus couleur diffuse plus textures plus illumination basée sur une image.*

*Figure 7: Diverses façons de représenter le transfert de radiance précalculé*

#### *3.1.1.4 DIFFÉRENCES*

Premièrement, tel que mentionné à la section 2.5.4, dans notre implantation du transfert de radiance précalculé, nous n'avons pas inclus tous les types de fonctions de transfert décrites dans [Sloan2002]. Afin de démontrer que notre implantation fonctionnait et qu'elle pouvait ensuite être améliorée en implantant les autres types de transfert, nous nous sommes concentrés sur le transfert diffus avec ombres. Ceci a simplifié l'implantation principalement par rapport aux deux aspects suivants:

- le processus de tracé de rayons nécessaire ne consistait qu'en un test d'ombrage, ce qui revient à lancer un rayon à partir de la surface de l'objet dans chaque direction d'échantillonnage sphérique pour voir s'il y a intersection avec le reste de la scène. Le tracé de rayons devient plus complexe pour le transfert avec interréflexion;
- on peut présumer que les ombres sont toujours monochromatiques (un point sur la surface sera soit dans l'ombre, soit dans la lumière, ou dans une zone de la pénombre). Il n'est donc pas nécessaire de tester et de stocker un résultat pour chaque bande de couleur, ce qui aurait aussi nécessité trois passes de rendu lors de la visualisation.

Par contre, il est clair que si notre implantation fonctionne pour le transfert diffus avec ombres, les autres types de transfert sont également possibles. Il serait par contre alors nécessaire d'optimiser le tracé de rayons, qui prendrait plus de temps pour des types de transfert plus complexes.

Autre différence par rapport à [Sloan2002], également mentionnée à la section 2.5.4, nous ne considérons que deux types de sources de lumière: lumière directionnelle et sonde d'illumination. L'article décrit une façon de supporter les autres types de sources de lumière souvent utilisées en rendu temps réel (*point* et *spot*) en échantillonnant l'éclairage en un certain nombre d'endroits sur la surface de l'objet à chaque image. Par contre, l'article décrit aussi plusieurs problèmes lorsque la source de lumière entre dans l'enveloppe convexe de l'objet, ce qui ne peut pas arriver pour des sources de lumière directionnelles.

### 3.1.2 Implantation du tampon d'illumination

Dans la phase d'initialisation du logiciel, on assigne à chaque objet géométrique de la scène un ensemble de données qui serviront à gérer cet objet par rapport au tampon d'illumination. Cet ensemble de données comporte, entre autres, la matrice de transformation de l'objet ainsi que sa matrice inverse, et aussi l'ensemble d'échantillons correspondant à cet objet. Chaque échantillon est composé du sommet et de la normale à ce sommet, prétransformés en espace monde afin d'accélérer le processus de rendu.

De plus, le tampon d'illumination lui-même possède certaines informations à propos de la scène. Parmi ces informations, il y a entre autres la liste des sources de lumière, la liste des objets ainsi que la liste des objets qui sont visibles à partir de la surface d'un autre objet. Cette dernière liste est remplie lors du calcul des coefficients des fonctions de transfert (voir section précédente) et sauvegardée dans les mêmes fichiers que le résultat du précalcul du transfert de radiance pour chaque objet, afin de pouvoir être lue lors du démarrage du logiciel lorsqu'on n'a pas besoin de faire le précalcul.

À la base, le tampon d'illumination ne fait rien tant qu'une mise à jour n'est pas nécessaire. Une mise à jour peut être causée par les événements suivants:

1. déplacement, rotation ou mise à l'échelle d'un objet (changement de l'une des matrices de transformation du graphe de scène);
2. déformation d'un objet (déplacement d'un ou plusieurs sommets);
3. déplacement ou rotation du point de vue (changement dans les matrices régissant la caméra).

Lorsque l'un de ces événements survient, une mise à jour commence. Les mises à jour de type 1 et 2 sont basées sur des objets, et donc peuvent engendrer le recalcul d'un grand nombre d'échantillons, alors que les mises à jour de type 3 n'entraîneront que le recalcul d'échantillons appartenant à des objets avec un matériel ayant une composante spéculaire, car ce sont les seuls échantillons affectés par un changement du point de vue.

Prenons l'exemple d'une mise à jour de type 1. De plus, considérant que notre implantation du transfert de radiance précalculé ne considère que le cas diffus avec ombres, nous ne parlerons que d'ombres, mais tout ceci s'applique tout aussi bien aux cas d'interréflexion diffuse ou de réflexion spéculaire.

Lorsqu'un objet est déplacé dans la scène, par exemple, l'objet lui-même, ainsi que les objets qui étaient visibles à partir de sa surface, sont placés dans une queue de mise à jour. Tous ces objets doivent être mis à jour car il était possible que l'un d'eux ait une ombre causée par le premier objet. Par la suite, la liste d'objets visibles à partir de la surface de cet objet sera vidée, et lors de sa mise à jour, les objets intersectés par le tracé de rayons seront mis dans la liste, ainsi qu'ajoutés à la queue de mise à jour, car il se peut que l'objet, à sa nouvelle position, cause une ombre sur l'un d'eux. Une fois l'illumination de tous ces objets mise à jour, la mise à jour du tampon d'illumination est terminée.

La Figure 8 expose graphiquement ce processus. Un objet de type ChangeSet est créé. Cet objet représente de façon atomique toutes les données reliées à un changement de la scène. L'objet qui a été modifié est ajouté au ChangeSet, puis tous les objets qui étaient visibles de la surface du premier sont également ajoutés. Le diagramme de la Fi-

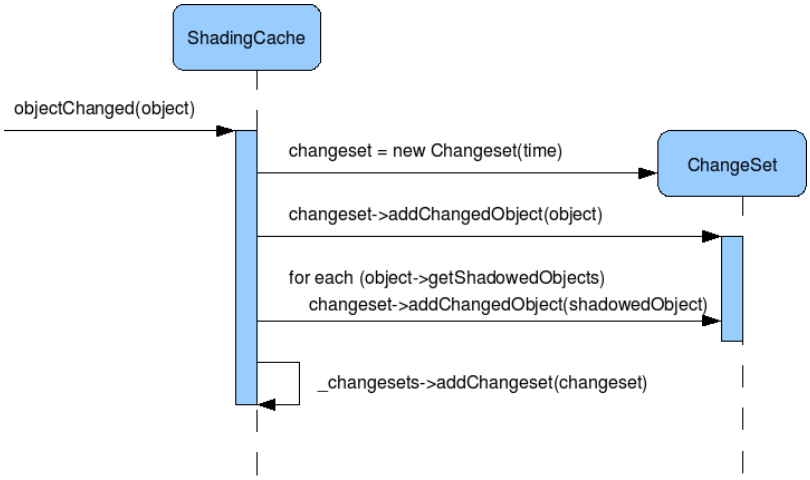



*Figure 8: Création de l'objet ChangeSet lors de la manipulation d'un objet*

gure 6 montre que par la suite, lors de la mise à jour, les objets qui sont visibles à partir de la surface du premier lors du tracé de rayons seront aussi ajoutés au ChangeSet. Notez qu'un objet ne sera pas ajouté deux fois au même ChangeSet, même si les deux diagrammes ne le montrent pas de façon explicite.

La Figure 9 montre ensuite ce qui se passe dans le fil d'exécution séparé dédié au tampon d'illumination (*ShadingCacheThread*) lorsqu'une modification survient. Ce fil d'exécution observe la liste de changements (variable `_changesets`) et commence à travailler lorsqu'elle n'est pas vide. Il met alors à jour la queue du tampon d'illumination, puis prend chaque échantillon un à un et le traite. Ce traitement est le même que dans la Figure 6.

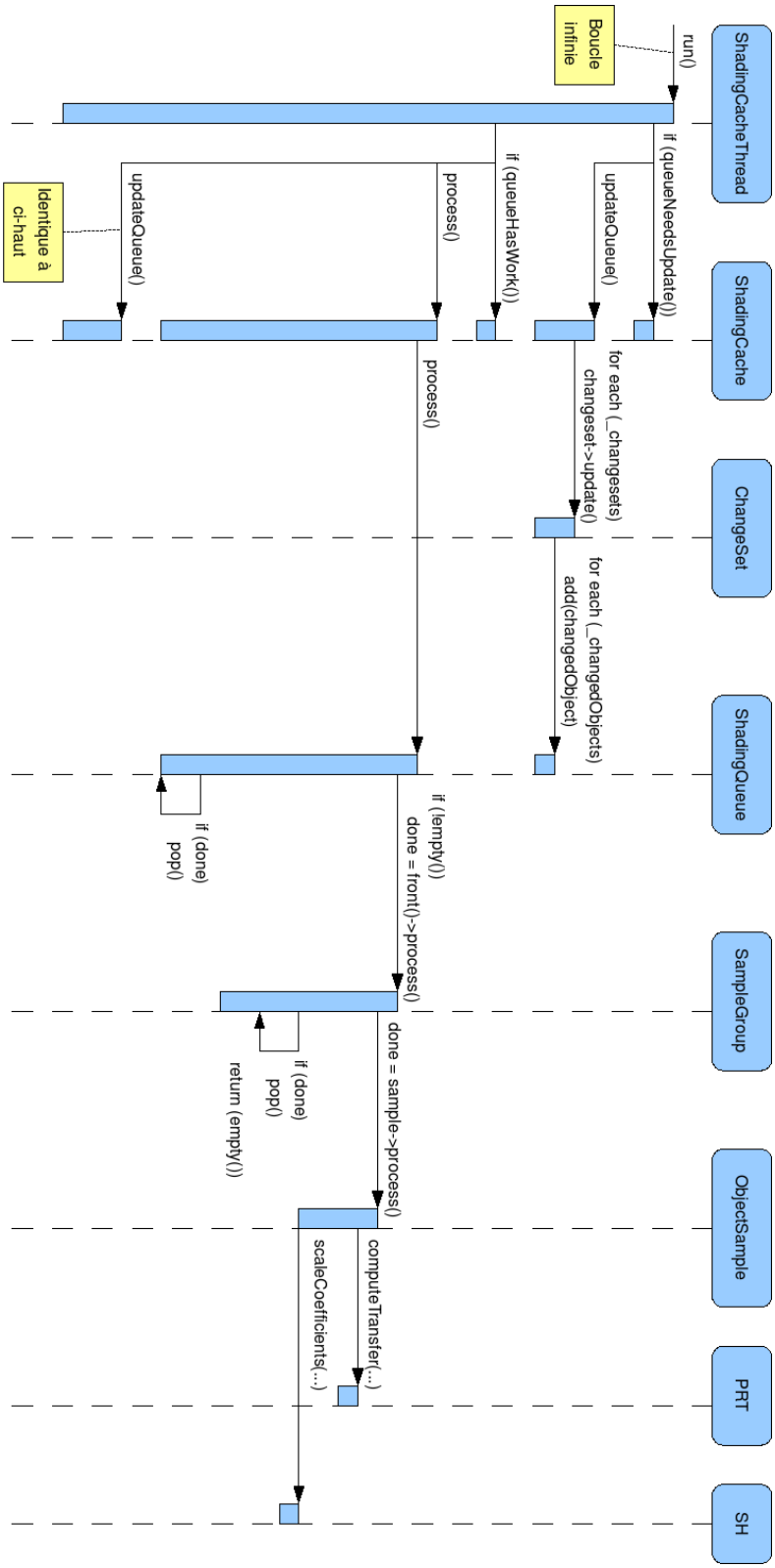



*Figure 9: Traitement lors d'une mise à jour.*

#### *3.1.2.1 DIFFÉRENCES*

Encore une fois, dans l'optique de montrer que le prototype fonctionne, certaines modifications ont été faites par rapport à la technique de base lors de l'implantation.

Premièrement, le tampon d'illumination tel que présenté dans [Tole2002] opérait en espace image, c'est à dire que le tracé de rayons se faisait à partir du point de vue vers la scène, et ce, seulement dans la pyramide de vue. Cette façon de faire, en plus de présenter quelques problèmes lors du déplacement du point de vue, ne s'applique pas au transfert de radiance précalculé, qui lui doit opérer en espace objet. Nous avons donc décidé de faire les mises à jour strictement par objet, et de les faire même pour des objets qui ne seraient pas visibles, afin qu'après le recalcul, la scène soit totalement valide même si le point de vue se déplace. Notez que ceci est aussi une conséquence du fait que notre implantation du transfert de radiance précalculé ne considère que le cas diffus avec ombres. Si les cas spéculaires avaient été implantés, la décision aurait pu être différente.

Ensuite, la technique de base fait un raffinement progressif du maillage des objets là où la variance entre deux échantillons est trop grande. Tel que mentionné à la section 2.5.4, ceci ne s'inscrivait pas dans nos objectifs de recherche. De plus, considérant que le transfert de radiance précalculé opère dans la base des harmoniques sphériques, et que pour des questions d'efficacité, nous devions utiliser un nombre de bandes relativement petit pour nos calculs, ce qui nous limitait à une représentation de basse fréquence de l'illumination, une stratégie de raffinement progressif n'aurait pas vraiment aidé à améliorer la qualité d'image. Nous avons donc plutôt construit les objets géométriques de façon à ce qu'il y ait un nombre suffisant de sommets aux endroits nécessaires pour

montrer l'illumination de façon acceptable. Ceci nous permet d'éviter la question du raffinement progressif, mais il y a tout de même des artéfacts visuels dans certaines parties des images où le niveau de subdivision des maillages n'est pas suffisant.

Notez que dans la description de ce qui arrive lorsqu'une mise à jour est nécessaire suite au déplacement d'un objet, il est mentionné que les coefficients de la fonction de transfert de tous les objets qui sont visibles de la surface de celui qui a été déplacé devront être recalculés. Ceci est causé par le fait que la fonction que notre implantation du tampon d'illumination doit recalculer n'est pas qu'un simple tracé de rayons comme dans le tampon d'illumination tel que décrit dans [Tole2002], mais un transfert de radiance précalculé. Puisque ce calcul considère un ensemble d'échantillons disposés en sphère autour de chaque sommet, on ne peut pas simplement recalculer les objets qui sont présentement dans l'ombre du premier, car l'ombre peut se déplacer lors d'une rotation des coefficients d'éclairage. Tout objet visible de la surface d'un autre pourrait potentiellement être dans l'ombre de celui-ci étant donné un certain ensemble de coefficients d'éclairage. Donc, il est vraiment nécessaire de vérifier si tous les objets qui étaient visibles de la surface de l'objet déplacé le sont encore, et vice versa.

## *3.2 EXPÉRIMENTATION*

Afin de pouvoir vérifier les hypothèses de départ, nous avons créé quelques scènes relativement simples. De plus, nous avons intégré une bonne variété d'environnements d'éclairage différents afin de pouvoir vérifier que l'illumination est bien représentée et que les ombres sont représentatives de la direction de la lumière incidente.

### 3.2.1 Scènes

Nous avons utilisé les scènes ci-dessous pour tester le prototype.

#### *3.2.1.1 Sphère et cube*

Cette scène simpliste permet de vérifier que les ombres sont valides, ainsi que de déterminer quel niveau de subdivision est nécessaire pour donner une bonne impression de clarté dans les ombres. Les murs projettent leur ombre sur le plancher ainsi que sur les autres objets dépendant de la source de lumière. La sphère projette une ombre sur le plancher, alors que le cube en projette une sur le plancher et les deux murs.

Le Tableau 1 montre la composition de la scène "Sphère et cube", et la Figure 10 en présente une image.

*Tableau 1: Composition de la scène "Sphère et cube".*

| Nom de l'objet | Nombre de sommets |
|---|---|
| Mur 1 | 1520 |
| Mur 2 | 1520 |
| Plancher | 4784 |
| Sphère | 1090 |
| Cube | 1536 |
| Total | 10450 |

#### *3.2.1.2 Échiquier*

Cette scène est plus lourde que la précédente. De plus, presque toutes les pièces sont visibles à partir d'une pièce donnée. La scène a donc permis de vérifier le temps nécessaire pour une mise à jour complète d'une scène relativement complexe. Comme vous le verrez, les objets sont beaucoup plus complexes qu'ils ne devraient l'être, ce qui

affecte directement le temps de mise à jour, car un calcul de transfert de radiance doit être fait à partir de chaque sommet.

Visuellement, cette scène est peut-être plus intéressante que la précédente, mais elle n'est pas optimale d'un point de vue de la représentation de la technique. Les pièces étant relativement petites, elles ne projettent jamais une très grande ombre et donc ne démontrent pas une grande amélioration visuelle par rapport à un rendu sans ombres, du moins comparé à la scène précédente où les ombres sont très évidentes. De plus, lorsqu'une pièce est déplacée, la mise à jour prend beaucoup de temps mais semble très petite et localisée, ce qui donne l'impression que le système est très lent.

Par contre, le but de cette scène est vraiment de montrer ce qui serait l'un des pires cas pour notre prototype. En plus du fait que les objets ont plus de sommets que nécessaire pour représenter adéquatement les pièces de jeu, la visibilité à partir d'une pièce fait en sorte que la majorité du temps, toutes les pièces devront être recalculées, ce qui met en évidence le fait que le tracé de rayons n'est pas aussi optimisé qu'il aurait pu l'être dans notre implantation.

Le Tableau 2 montre la composition de la scène "Échiquier", et la Figure 11 en présente une image.

*Tableau 2: Composition de la scène "Échiquier".*

| Nom de l'objet | Nombre de sommets |
|---|---|
| Cases blanches et interstices | 6000 |
| Cases noires | 2048 |
| Extérieur de l'échiquier | 2616 |
| Pion (x16) | 3744 |
| Cavalier (x4) | 2718 |
| Fou (x4) | 5196 |
| Tour (x4) | 4098 |
| Reine (x2) | 7782 |
| Roi (x2) | 6180 |
| Total | 146540 |

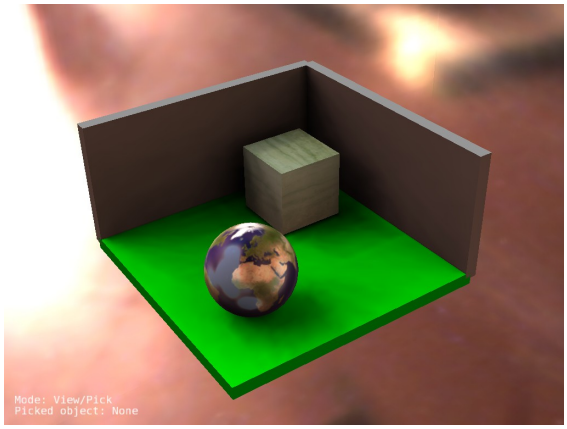



*Figure 10: Scène "Sphere et cube".*

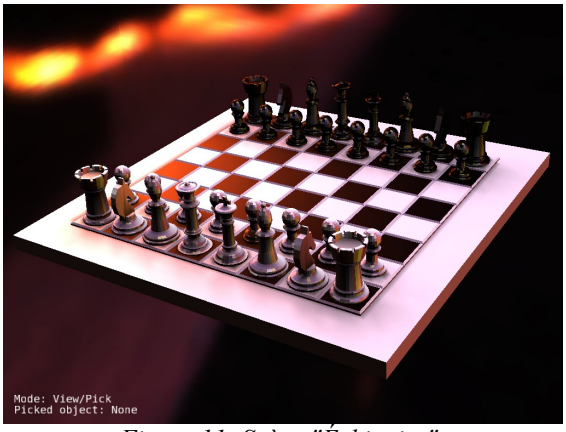



*Figure 11: Scène "Échiquier".*

#### *3.2.1.3 Optimisation des scènes*

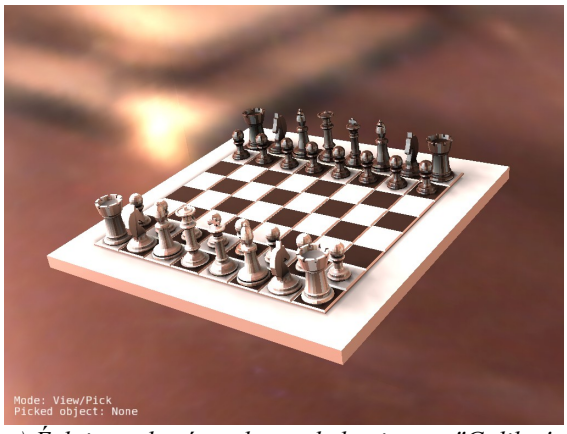
Puisque le traitement se fait pour chaque sommet, il est évident que réduire le nombre de sommets a un grand effet sur le temps de mise à jour total. Nous avons pu faire une optimisation automatique des deux scènes, qui consistait en la création de bandes de triangles (*triangle strips*). Ceci ne devrait pas réduire le nombre de sommets en soi, mais ce faisant, certains sommets coïncidents pouvaient être fusionnés. Cette opti-

misation a donné de meilleurs résultats sur la scène "Sphère et cube" que sur la scène "Échiquier", probablement parce qu'un grand nombre de sommets font partie d'un même objet géométrique continu, ce qui n'est pas le cas pour la seconde scène, qui comporte plus de détails qui ont nécessité un plus grand nombre de petits objets géométriques, et même des objets ayant des trous. Dans le premier cas, le nombre total de sommets a diminué de 46%, et dans le second, de 5%. Le chapitre suivant, Résultats et discussion, présente plusieurs tableaux qui permettront de récapituler les différentes données.

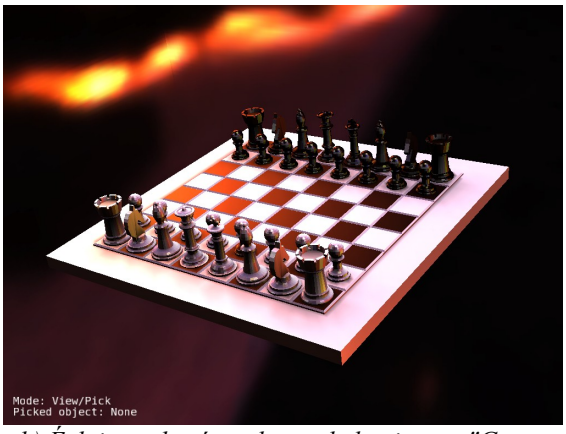
### 3.2.2 Environnements d'éclairage

Les environnements d'éclairage utilisés sont interchangeables. Nous en avons intégré plusieurs surtout pour l'intérêt visuel, car en réalité, ils n'affectent pas l'algorithme lui-même. Par contre, ils permettent de montrer que différents types d'éclairage donnent des résultats différents lorsque représentés sous forme d'harmoniques sphériques.

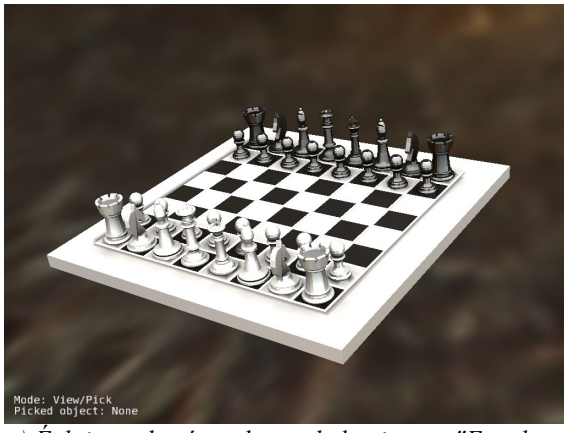
Les environnements basés sur des images viennent du site de sondes d'illumination de Paul Debevec (Figure 12, images a à e). L'autre type de source de lumière supporté est une source directionnelle avec un vecteur incident prédéfini (Figure 12f).



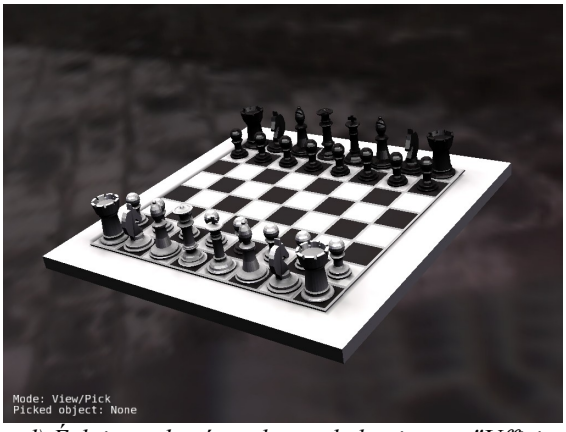

*a) Éclairage basé sur la sonde lumineuse "Galileo's Tomb".*



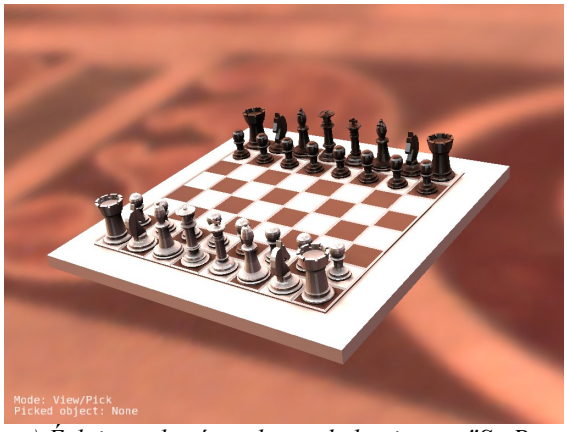

*b) Éclairage basé sur la sonde lumineuse "Grace Cathedral".*



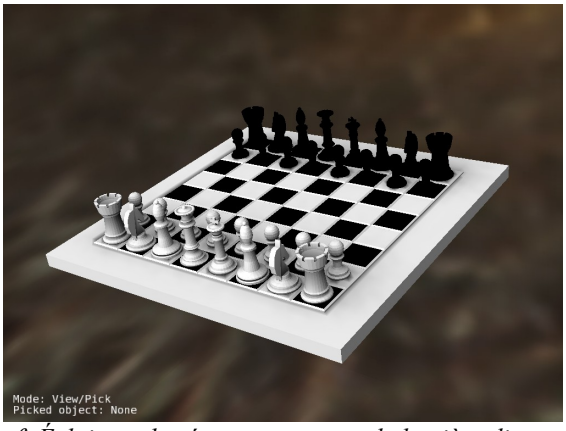

*c) Éclairage basé sur la sonde lumineuse "Eucalyptus Grove".*




*d) Éclairage basé sur la sonde lumineuse "Uffizi Gallery".*




*e) Éclairage basé sur la sonde lumineuse "St. Peter's Basilica".*




*f) Éclairage basé sur une source de lumière directionnelle.*

*Figure 12: Environnements d'éclairage*

### 3.2.3 Rotation des coefficients en harmoniques sphériques

Nous avons intégré une méthode de rotation des coefficients en harmoniques sphériques [Sloan2002], afin de montrer l'un des avantages du transfert de radiance pré-calculé par rapport à d'autres techniques de précalcul de l'éclairage. Lorsqu'une rotation s'effectue, ce sont en réalité les coefficients d'éclairage auxquels on fait faire une rotation, plutôt que de l'effectuer sur les coefficients des fonctions de transfert de chaque objet géométrique de la scène. Notons encore une fois que la technique de base du transfert de radiance précalculé ne s'applique qu'à une scène en entier (objet statique et rigide), et c'est pourquoi on ne peut faire faire de rotation qu'à la scène en entier dans ce cas-ci.

Il est important de noter que la technique de rotation implantée n'est pas la seule, c'était seulement la plus directe. La rotation d'harmoniques sphériques constitue apparemment un sujet important en chimie, à en croire les publications d'où proviennent la majorité des articles à ce sujet, notamment [Ivanic1996].

# CHAPITRE 4

# RÉSULTATS ET DISCUSSION

## *4.1 RÉSULTATS*

Cette section présente les résultats de notre expérimentation, à partir desquels nous pourrons tirer des conclusions reliées à nos objectifs décrits dans la section 2.5. Les résultats reliés aux trois critères principaux seront présentés, ainsi que des résultats secondaires qui comparent notre technique aux techniques sur lesquelles nous nous sommes basés.

Tous les résultats qui seront donnés dans ce chapitre ont été obtenus dans l'environnement matériel et logiciel suivant:

Ordinateur: Intel QX6600 (quadruple coeur, 2,4GHz), 3 Go de mémoire vive, carte graphique nVidia GeForce 8800GTX 768 Mo.

Système d'exploitation: Ubuntu Linux, noyau 2.6.22, pilotes nVidia (non libres).

### 4.1.1 TEMPS DE CALCUL

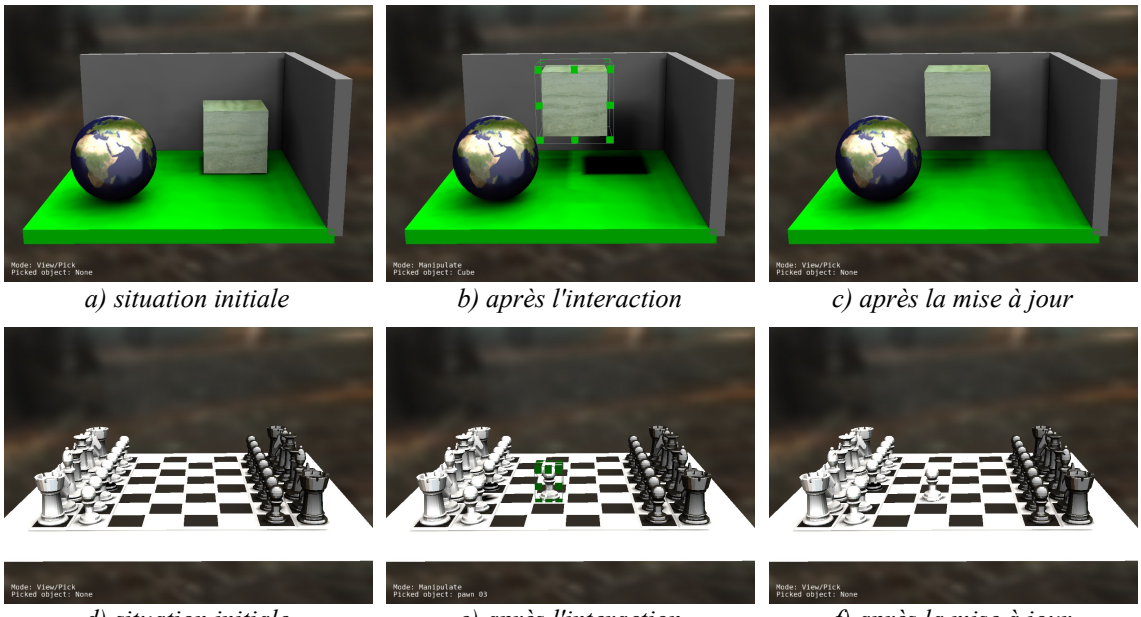
Le Tableau 3 donne une comparaison du temps de calcul des deux scènes que nous utilisons pour l'expérimentation, tant au niveau du calcul initial qu'à la mise à jour. Ces résultats ont été obtenus à 400 échantillons par sommet, avec des harmoniques sphériques à 5 bandes (25 coefficients). De plus, ils ont été obtenus avec les scènes optimi-

sées. Voir le Tableau 7 pour une comparaison des mêmes résultats entre la version originale et la version optimisée des scènes.

*Tableau 3: Temps de calcul et de mise à jour.*

| Scène | Sphère et cube | Échiquier |
|---|---|---|
| Nombre de sommets | 5612 | 139070 |
| Nombre d'échantillons | 2244800 | 55628000 |
| Précalcul initial (sec) | 12.31 | 570.12 |
| Nb d'échantillons/sec | 182327.89 | 97573.27 |
| Mise a jour (sec) | 14.67 | 731.66 |
| Nb d'échantillons/sec | 152993.70 | 76030.37 |

La Figure 13 illustre la mise à jour. La première image représente la situation initiale, avec les paramètres donnés plus haut. La seconde a été prise immédiatement après le déplacement d'un objet, et la troisième à la fin de la mise à jour (le Tableau 3 donne le temps requis, qui représente le temps entre la deuxième et la troisième image ci-dessous).



*a) situation initiale* *b) après l'interaction* *c) après la mise à jour*

*d) situation initiale* *e) après l'interaction* *f) après la mise à jour*

*Figure 13: Processus d'interaction et de mise à jour. (a,b,c) Scène "Sphère et cube"; (d,e,f) Scène "Échiquier".*

### 4.1.2 Vitesse d'interaction

Tel que mentionné à la section 2.5.3, l'un des buts premiers de la technique présentée dans ce projet de recherche est de permettre à l'illumination de se mettre à jour sans nuire à l'interaction. Il est donc important de s'assurer que la vitesse d'interaction lorsqu'une mise à jour est en train de se produire est semblable à celle lorsque le système de tampon d'illumination est au repos. C'est ce que montre le Tableau 4.

*Tableau 4: Vitesse d'interaction au repos et lors d'une mise à jour.*

| Scène | Sphère et cube | Échiquier |
|---|---|---|
| Au repos (images/sec) | 187-192 | 23.6-24.2 |
| Mise à jour (images/sec) | 185-193 | 23.4-24.1 |

On voit ici que la différence de débit est négligeable (environ 1%).

### 4.1.3 Taille des données

Dans [Sloan2002], on mentionne que l'espace requis pour le stockage du précalcul est important. Le Tableau 5 montre que ce n'est pas le cas pour notre implantation.

*Tableau 5: Taille des données.*

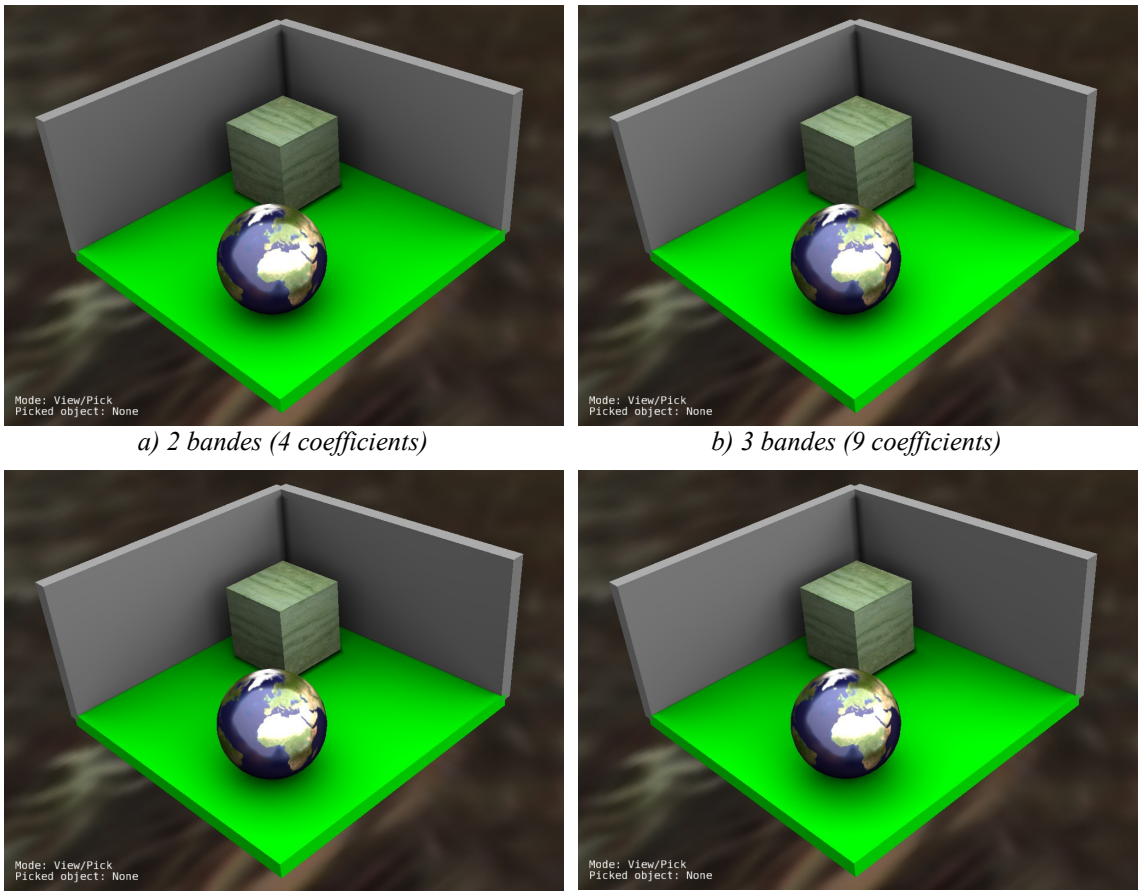
| Scène | Sphère et cube | Échiquier |
|---|---|---|
| Nombre de sommets | 5612 | 139070 |
| Taille des données (mégaoctets) | 1.26 | 30.06 |
| Espace requis par sommet (octets) | 235.96 | 226.68 |

Notez bien que notre format de données étant textuel plutôt que binaire pour des raisons de simplicité, il est normal que l'espace requis par sommet varie quelque peu. Par contre, peu importe le nombre d'échantillons qui seront calculés, l'espace requis pour le

stockage ne variera pas vraiment, car ce sont les coefficients d'harmoniques sphériques pour chaque sommet qui sont stockés. Donc, l'espace requis varie plutôt par rapport au nombre de bandes, comme le présente le Tableau 6.

### 4.1.4 Taille des données en fonction du nombre de bandes

Dans [Sloan2002], on fait des comparaisons de qualité d'image en fonction du nombre de bandes utilisées lors du calcul des coefficients en harmoniques sphériques. Dans les images présentées dans cet article, on peut voir qu'il y a peu de différences entre 4 et 5 bandes (16 et 25 coefficients). Dans notre cas, nous avons comparé de 2 à 5 bandes, et avons vu encore moins de différence, même lorsque l'illumination est calculée avec 10 000 échantillons par sommet. La Figure 14 présente cette comparaison, mais avec des images de si petite taille, il sera pratiquement impossible de déceler une différence entre les images (la plus grande différence visible est que les parties éclairées sont quelque peu plus claires plus il y a de bandes).




*a) 2 bandes (4 coefficients)* *b) 3 bandes (9 coefficients)*

*c) 4 bandes (16 coefficients)* *d) 5 bandes (25 coefficients)*

*Figure 14: Comparaison de la qualité d'image en fonction du nombre de bandes.*

Par contre, la taille des données qu'il faut stocker change avec le nombre de bandes utilisées, et c'est ce que le Tableau 6 montre.

*Tableau 6: Taille des données en fonction du nombre de bandes.*

| Bandes (n) | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| Coefficients ($n^2$) | 4 | 9 | 16 | 25 |
| Taille à 400 éch/sommet (mo) | 0.19 | 0.44 | 0.8 | 1.26 |
| Facteur d'augmentation | | 2.28 | 1.8 | 1.58 |
| Taille à 10 000 éch/sommet (mo) | 0.2 | 0.47 | 0.85 | 1.35 |
| Facteur d'augmentation | | 2.29 | 1.82 | 1.58 |

### 4.1.5 Comparaison entre la version originale et optimisée des scènes

Tel que mentionné à la section 3.2.1.3, il fut possible de faire une optimisation des scènes qui a eu comme effet de réduire le nombre de sommets. Le Tableau 7 montre l'impact que cette optimisation a eu sur le temps de calcul de l'illumination.

*Tableau 7: Effet de l'optimisation des scènes.*

| Scène | Sphère et cube | | | Échiquier | | |
|---|---|---|---|---|---|---|
| | Avant | Après | Amélioration | Avant | Après | Amélioration |
| Nombre de sommets | 10450 | 5612 | 46.3% | 146540 | 139070 | 5.1% |
| Nombre d'échantillons | 4180000 | 2244800 | | 58616000 | 55628000 | |
| Précalcul initial (sec) | 22.97 | 12.31 | 46.4% | 587.84 | 570.12 | 3.0% |
| Nb d'échantillons/sec | 182000.58 | 182327.89 | | 99713.58 | 97573.27 | |
| Mise a jour (sec) | 28.19 | 14.67 | 47.9% | 786.58 | 731.66 | 7.0% |
| Nb d'échantillons/sec | 148290.58 | 152993.70 | | 74520.26 | 76030.37 | |

On peut constater que la diminution du nombre de sommets donne une diminution proportionnelle du temps de calcul. On peut également supposer que si l'on pouvait calculer plus d'échantillons en parallèle, par exemple sur un processeur à quatre coeurs (ce qui devient de plus en plus courant), ça donnerait également une amélioration linéaire du temps de calcul. Ceci est à prévoir pour toute technique basée sur le tracé de rayons.

### 4.1.6 Comparaison entre 400 et 10 000 échantillons par sommet

Nous avons comparé deux niveaux de qualité d'image en termes de temps de calcul. La littérature au sujet de l'illumination précalculée en harmoniques sphériques mentionne que 10 000 échantillons par sommet donne normalement de bons résultats ([Sloan2002],[Green2003] et autres), et du point de vue qualité d'image c'est certainement vrai, mais nous avons remarqué que 400 échantillons par sommet donne tout de même un relativement bon résultat (bonne impression de l'ombre malgré des artéfacts

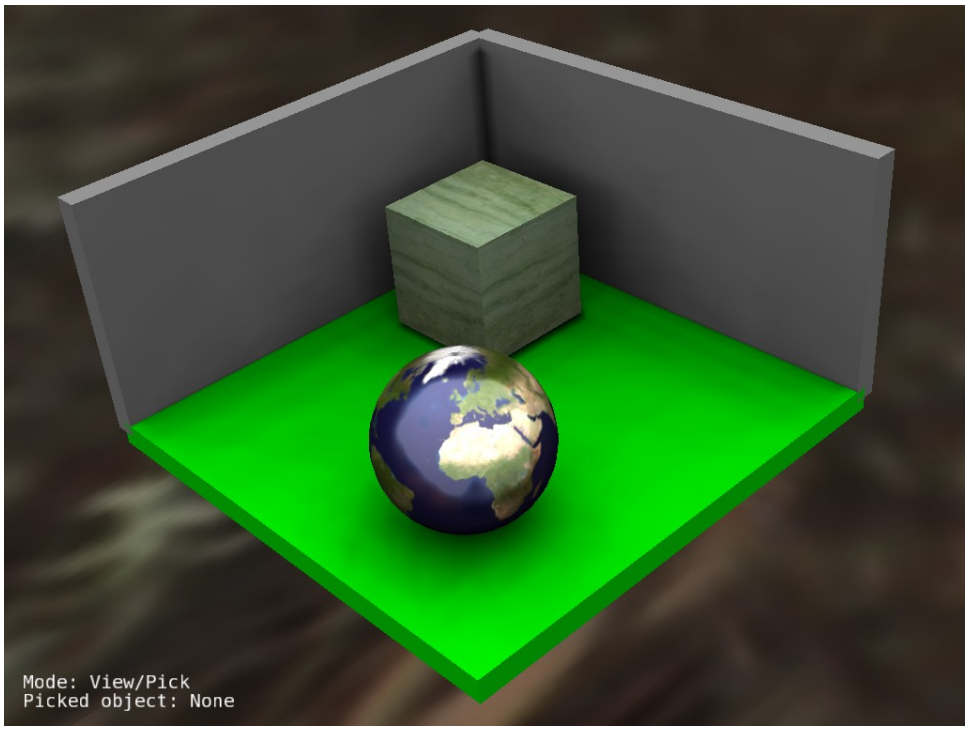
d'effet de bande mineurs), et le temps de calcul s'en trouve fortement réduit. Le Tableau 8 donne donc une comparaison du temps de calcul pour ces deux scénarios.

*Tableau 8: Comparaison entre 400 et 10 000 échantillons par sommet.*

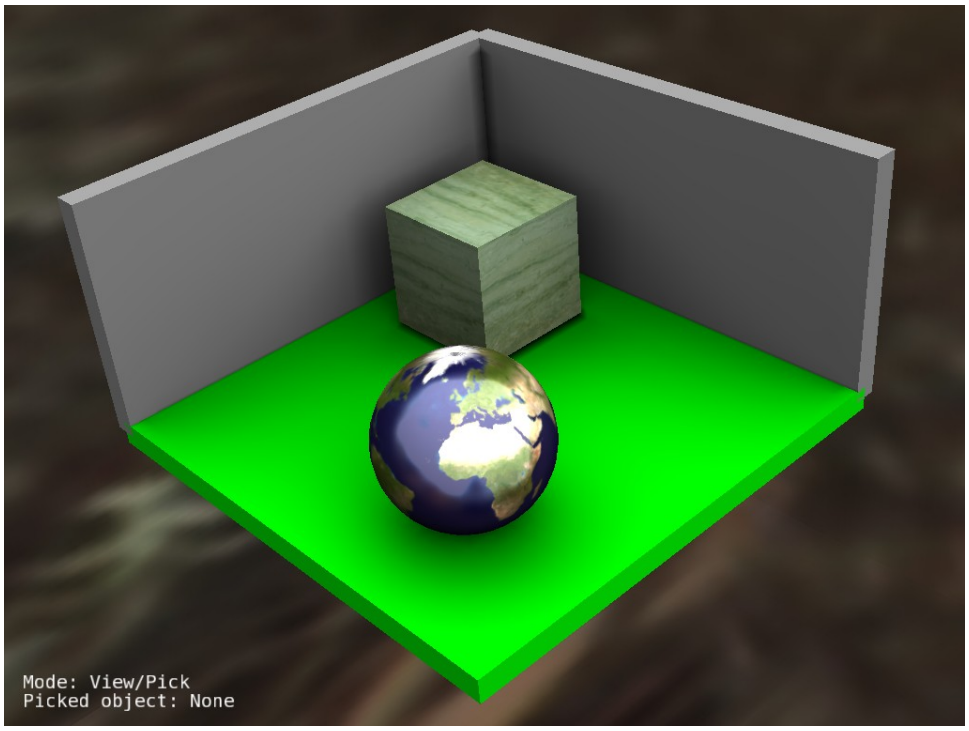
| Scène | Sphère et cube | Sphère et cube | Échiquier | Échiquier |
|---|---|---|---|---|
| Nombre d'échantillons par sommet | 400 | 10000 | 400 | 10000 |
| Nombre de sommets | 5612 | 5612 | 139070 | 139070 |
| Nombre d'échantillons | 2244800 | 56120000 | 55628000 | 1390700000 |
| Précalcul initial (sec) | 12.31 | 267.19 | 570.12 | 12944.91 |
| Nb d'échantillons/sec | 182327.89 | 210036.64 | 97573.27 | 107432.20 |
| Mise a jour (sec) | 14.67 | 356.43 | 731.66 | 17725.27 |
| Nb d'échantillons/sec | 152993.70 | 157449.39 | 76030.37 | 78458.61 |

On note une petite augmentation du nombre d'échantillons par seconde à 10 000 échantillons par sommet, qui est probablement causée par une augmentation d'efficacité au niveau de l'utilisation de la mémoire tampon du processeur. Puisque plus d'échantillons sont calculés au même sommet, les données nécessaires sont déjà chargées pour un plus grand nombre de calculs subséquents avant de changer de sommet.

La Figure 15 démontrent la différence de qualité d'image. Notez surtout la douceur des dégradés dans la pénombre où 10 000 échantillons par sommet sont utilisés, alors que pour 400 échantillons par sommet, les dégradés sont légèrement saccadés (artéfacts d'effet de bande).




*a) 400 échantillons par sommet*




*b) 10 000 échantillons par sommet*

*Figure 15: Comparaison de la scène "Sphère et cube" à 400 et 10 000 échantillons par sommet*

### 4.1.7 Changement de l'environnement d'éclairage

L'un des avantages d'utiliser le transfert de radiance précalculé comme fonction de rendu est que la fonction illumination est découplée des fonctions de transfert, ce qui permet de changer cette première indépendamment de la seconde. Le Tableau 9 donne le temps requis pour recalculer la fonction éclairage lorsque l'environnement d'éclairage change (par exemple, si on charge une nouvelle sonde d'illumination). Puisque dans notre cas, l'éclairage est global à la scène, il est représenté par une seule fonction sphérique, et donc par un seul ensemble de 400 ou 10 000 échantillons, ce qui prend très peu de temps à recalculer (contrairement au recalcul du grand nombre d'échantillons qui représentent les fonctions de transfert à chaque sommet).

*Tableau 9: Temps requis pour changer la situation d'éclairage.*

| Nombre d'échantillons | 400 | 10000 |
|---|---|---|
| Temps maximal (sec) | 0.004034 | 0.005908 |

Notez que nous avons pris le temps maximum parmi 6 changements de situation d'éclairage (pour chacune des situations d'éclairage que nous avons utilisées), et que ce temps comprend le calcul des échantillons mais ne comprend pas le temps de chargement des images pour l'éclairage basé sur des images, qui n'est fait qu'une fois au démarrage du programme.

### 4.1.8 Rotation des harmoniques sphériques

Tel que mentionné, l'un des avantages de stocker le résultat du rendu en harmoniques sphériques est qu'aucun recalcul n'est nécessaire dans certaines situations. L'une

de ces situations est une rotation de tous les objets (par rapport à l'éclairage), dans quel cas on peut effectuer une rotation inverse des coefficients d'éclairage. Le Tableau 10 compare la vitesse d'affichage au repos et lors d'une rotation continue de 5 degrés par seconde de la scène.

*Tableau 10: Vitesse d'affichage lors d'une rotation de la fonction d'éclairage.*

| Nombre d'échantillons | Sphère et cube | Échiquier |
|---|---|---|
| Au repos (images/sec) | 523-545 | 63-64 |
| En rotation (images/sec) | 181-185 | 26-27 |

Il y a eu beaucoup de littérature au sujet de la rotation d'harmoniques sphériques, surtout dans le domaine de la chimie quantique. Notre objectif était simplement de démontrer que c'est un avantage de notre technique par rapport aux autres. Ainsi, nous avons seulement implanté la méthode de rotation telle que décrite dans [Sloan2002]. Si nos objectifs de recherche avaient compris l'amélioration de ces techniques, il aurait été intéressant de voir pourquoi c'est si demandant et d'essayer d'y remédier. On note effectivement une forte diminution du taux d'affichage, ce que les travaux à ce sujet tentent de remédier.

## *4.2 DISCUSSION*

Comme les deux techniques utilisées comme base à ce travail apportent chacune leur lot d'avantages et de désavantages, nous pouvons en tirer des conclusions.

### 4.2.1 Technique de rendu

La technique du tampon d'illumination utilisait, dans l'article initial, un tracé de chemins bidirectionnel [Lafortune1993] comme technique de rendu de haute qualité. En plus d'opérer en espace image à la base, ce qui permettait au système de se concentrer dans les parties de la scène qui sont visibles en premier, cette technique avait comme avantage qu'il était possible de remonter le chemin d'un rayon afin de voir quelles surfaces avaient contribué à la valeur finale, et donc de ne recalculer que ce qui avait réellement changé.

Dans notre cas, la technique de rendu est un calcul de transfert de radiance "précalculé" (bien que le terme précalculé ne s'applique plus). Le tracé de rayons se fait dans un ensemble d'échantillons sphériques autour de chaque sommet de l'objet. Cela, combiné au fait qu'il est difficile de déterminer exactement quels objets ont contribué à la valeur finale d'un tracé de rayons, fait en sorte que dans bien des cas, un grand nombre d'objets devront être mis à jour en utilisant la technique décrite dans ce travail.

Il faut également garder en tête que le transfert de radiance précalculé implique qu'il faut lancer plusieurs rayons par sommet (400 à 10 000 dans notre cas). Même si on a découplé le calcul de la fonction illumination du calcul des fonctions de transfert, puisque l'opération la plus coûteuse est le tracé de rayons, le gain est négligeable.

Comparons le nombre de tracés de rayons que le tampon d'illumination pourrait faire au nombre donné dans les résultats pour notre technique, plus haut. Le tampon d'illumination, qui opère en espace image, lancera au maximum un rayon de premier ni-

veau par pixel du plan image (au niveau de subdivision maximal, voir la section suivante), puis un certain nombre de rayons additionnels pour chaque rayon de premier niveau, selon les propriétés de la surface que celui-ci a frappé. Par exemple, si le plan image final avait une résolution de 1024x768, 786 432 rayons de premier niveau doivent être lancés dans le pire des cas.

Pour ce qui est de notre technique, on voit dans le Tableau 3 que même après simplification, une scène simpliste requiert 2 244 800 rayons à 400 échantillons par sommet. Ceci veut dire que pour égaler cette valeur, il faudrait en moyenne 1,85 tracés de rayons additionnels par rayon de premier niveau, dans le pire des cas. On se souviendra que le tampon d'illumination permet de spécifier jusqu'à quel point les surfaces devraient être subdivisées afin d'en raffiner l'apparence; si on suppose plutôt qu'il y aura en moyenne un élément de maillage par deux pixels du plan image, c'est plutôt 4,71 rayons additionnels par rayon de premier niveau qui peuvent être lancés avant d'obtenir le même nombre de tracés de rayons que notre technique. Tout ceci est purement hypothétique bien sûr, car nous ne savons pas le nombre exact de tracés de rayons effectués par le tampon d'illumination, mais cela met en évidence l'un des désavantages d'avoir choisi le transfert de radiance précalculé comme technique de rendu.

Il aurait été intéressant d'explorer une autre technique, comme par exemple le tracé de photons [Jensen1996], qui opère à la fois en espace objet et en espace image. Lors d'une modification de la scène, il aurait alors été possible de relancer les photons, ce qui est peu coûteux, puis de ne refaire le rendu que de la partie de l'image qui a changé, ce qui aurait réduit le nombre de tracés de rayons nécessaires de façon significative.

Néanmoins, le transfert de radiance précalculé comporte plusieurs avantages, dont le fait qu'il reste valide lorsque les conditions d'éclairage sont modifiées ou lorsque l'environnement d'éclairage ou les objets font des rotations l'un par rapport à l'autre.

### 4.2.2 SUBDIVISION DU MAILLAGE

D'autre part, dans l'article initial, le tampon d'illumination créait un maillage conceptuel par objet géométrique sur lequel les échantillons du rendu étaient stockés à chaque sommet. Le système subdivisait le maillage lorsque la variance entre deux échantillons voisins était trop grande, jusqu'à une certaine limite (probablement jusqu'à ce que la subdivision ait la taille d'un pixel à l'écran). Ensuite, lorsqu'une mise à jour était nécessaire, le système calculait le nombre d'échantillons à recalculer et s'il y en avait trop, il réduisait le niveau de subdivision afin de pouvoir améliorer la qualité de l'image plus rapidement. Ensuite, le processus de subdivision recommençait.

Tel que mentionné à la section 2.5.4, cet aspect ne faisait pas partie des objectifs de notre recherche. Nous avons plutôt présumé que les objets chargés par le logiciel avaient un niveau de subdivision suffisant pour donner l'effet escompté, mais pas si grand qu'un trop grand nombre d'échantillons devaient être calculés lors d'une mise à jour.

Par exemple, un plancher ne doit pas être composé d'un seul quadrilatère, car à ce moment, l'illumination ne sera capturée qu'aux quatre sommets et interpolée sur le polygone, ce qui ne représentera pas l'ombre qu'un objet porte sur sa surface de façon adéquate. Ça revient à un problème d'échantillonnage, semblable à celui auquel font face

les systèmes de rendu basés sur la radiosité [Goral1984], avec les mêmes problèmes et les mêmes artéfacts visuels. On voit ce problème facilement dans la scène "Sphère et cube" (voir, par exemple, la Figure 15): l'ombre du cube bleu continue à côté de celui-ci dans une région qui reçoit beaucoup de lumière, car il y a interpolation entre les sommets qui sont sous le cube et ceux qui ne le sont pas.

### 4.2.3 Comparaison aux textures d'illumination

Ce travail, malgré ses lacunes qui ont été énumérées plus haut, compte plusieurs avantages en comparaison à d'autres techniques qui permettent de précalculer l'éclairage. Hormis le fait qu'une mise à jour est possible en cours d'exécution du logiciel, voici certains autres avantages que nous pouvons voir. Nous pouvons comparer la technique décrite dans ce travail à l'autre technique le plus souvent utilisée pour précalculer de l'éclairage de haute qualité et l'appliquer à des surfaces en temps réel, les textures d'illumination [Segal1992]. Pour faire cette comparaison, nous devons dans un premier temps décrire quelques cas d'utilisation des textures d'illumination. Ces cas d'utilisation iront du plus simple au plus complexe. Puis, nous montrerons comment la technique décrite dans ce travail simplifie les choses même pour les cas les plus complexes.

Décrivons d'abord l'utilisation de base des textures d'illumination. L'idée est d'effectuer un rendu de haute qualité de l'environnement statique d'une scène (une pièce et son mobilier, qui restera immobile à l'exécution de l'application, par exemple) et de stocker des valeurs d'éclairage pour chaque surface dans des textures. Lors de la visualisation, ces textures sont appliquées aux surfaces et l'éclairage provenant de sources de lu-

mière dynamiques est ensuite appliqué par-dessus (multiplicativement). Ainsi, les parties statiques de la scène ont une illumination réaliste ainsi que des ombres par rapport à des sources de lumière statiques, mais les objets dynamiques peuvent encore les éclairer, et un calcul d'ombres dynamiques (*shadow map*, *shadow volume*, etc.) doit être fait pour que ces objets puissent ombrager les parties statiques et les parties dynamiques de la scène.

Examinons maintenant un exemple plus complexe d'utilisation des textures d'illumination. Si la situation d'éclairage générale de la scène n'est pas statique mais varie de façon prévisible, il est nécessaire de précalculer une série de textures d'illumination et de les interchanger, et on peut même faire de l'interpolation entre les différentes situations d'éclairage. Un exemple de ce genre d'effet pourrait être l'éclairage extérieur durant la journée. On peut alors précalculer, par exemple, vingt-quatre situations d'éclairage, une pour chaque heure, dans lesquelles on ferait varier l'angle du "soleil" et de la "lune". On peut aussi, dans le même contexte, précalculer deux séries de vingt-quatre images, l'une pour un ciel clair et l'autre pour un ciel ennuagé. On voit que ce genre de technique est flexible, mais les contraintes d'espace peuvent limiter les effets possibles dans certaines circonstances.

La situation est différente lorsqu'il devient possible de mettre à jour les portions statiques ou quasi-statiques (c'est à dire statiques la plupart du temps) dans la scène. Pour faire une comparaison valide, notons qu'il est possible d'utiliser, pour le transfert de radiance précalculé, le même calcul que celui utilisé pour faire le précalcul des textures d'illumination. En admettant qu'on applique un processus semblable à celui décrit dans ce

travail à une scène où certains objets sont définis comme étant statiques ou quasi-statiques, il faudrait encore une fois calculer des ombres dynamiques projetées soit par les objets dynamiques ou celles projetées sur ceux-ci par les objets statiques, ce qui est semblable à ce qui est nécessaire lorsque les textures d'illumination sont utilisées.

Pour l'exemple utilisé plus haut, où l'éclairage d'une scène extérieure change par rapport à l'heure et aux conditions atmosphériques, aucun espace additionnel de stockage ne serait nécessaire. Dans ce cas, le temps relativement minime requis pour recalculer les coefficients de la fonction d'éclairage (disons 6 millisecondes, voir le Tableau 9) signifie que ce pourra être fait plus souvent afin de donner un effet plus réaliste, en plus d'éliminer le besoin de stocker des textures d'illumination pour un grand nombre de situations d'éclairage. De plus, une interpolation entre des situations d'éclairage serait moins coûteuse qu'une interpolation entre des ensembles de textures d'illumination.

Un autre avantage vient lorsqu'on veut soit déplacer ou déformer certains objets normalement statiques par rapport aux autres ou modifier l'environnement d'éclairage.

Comme nous l'avons vu dans le Tableau 3, le temps de recalcul des objets ayant une visibilité directe sur un objet modifié est raisonnable. Ceci offre donc la possibilité de définir des scénarios qui n'auraient pas été possibles en n'utilisant que les textures d'illumination, comme par exemple un char d'assaut qui tire sur le côté d'un bâtiment et qui le détruit partiellement. Dans ce cas, l'ombre causée par le bâtiment devra être recalculée, mais au moins, après un certain temps, l'image sera correcte, alors que dans un

système où les bâtiments sont totalement rigides, ce niveau de réalisme (de pouvoir modifier l'environnement par les actions de l'utilisateur) ne serait pas possible.

# CONCLUSION

Pour représenter de manière réaliste des scènes du monde réel en infographie, une simulation de plus en plus complète du comportement réel de la lumière devient nécessaire. Par contre, un grand nombre d'applications nécessitent d'interagir avec les scènes en temps réel, ce qui rend peu utiles les solutions directement liées aux techniques de tracé de rayons et autres techniques de rendu en différé. Les techniques de rendu par tramage, quant à elles, doivent s'attaquer à chaque effet de la lumière un à un et trouver une façon efficace de le reproduire de façon parfois détournée; si intéressant et stimulant que soit l'exercice, il devient lourd de représenter dans le même système un grand nombre d'effets d'illumination.

Dans le contexte de cette apparente impasse, le présent travail de recherche offre de combiner une technique de précalcul de l'illumination qui peut tirer profit de n'importe quel système de rendu en différé à une technique permettant de refaire des calculs d'illumination qui ne seraient plus valides suite à une interaction de l'utilisateur. Nous avons remarqué qu'elles étaient complémentaires et que leur combinaison permettrait d'atteindre le but d'une illumination réaliste dans des scènes dynamiques. De plus, elles présentaient des avantages intéressants qui rendraient certaines opérations plus efficaces à effectuer.

Les résultats expérimentaux ont vérifié nos hypothèses. Premièrement, nous avons démontré que le recalcul des fonctions de transfert en différé ne ralentissait pas

l'interaction de l'utilisateur, ce qui était l'un des objectifs premiers de ce travail. Ensuite, étant donnés les critères établis, la qualité d'image et le temps de mise à jour dans les scènes utilisées sont acceptables.

Finalement, il faut dire que l'utilisation du transfert de radiance précalculé amène certains avantages qui amenuisent quelque peu le désavantage qu'est le fait d'avoir à calculer des centaines, voire des milliers d'échantillons par sommet. La possibilité de changer totalement l'environnement d'éclairage ou d'effectuer une rotation de celui-ci, ainsi que le fait que les calculs se font en espace objet et donc ne doivent pas être recalculés lors d'un déplacement du point de vue si les surfaces n'ont que des propriétés diffuses, font en sorte que si les opérations permises à l'utilisateur sont bien choisies, le temps de recalcul sera acceptable.

Des recherches se font constamment sur le sujet de l'illumination globale en temps réel dans des scènes dynamiques. Même si nous n'avons pas atteint ce but avec ce travail, nous espérons avoir tracé un chemin qui pourra mener à d'autres solutions intéressantes et à des améliorations des techniques présentées dans les dernières années. Il est déjà possible de voir le transfert de radiance précalculé en action dans certains engins de jeux, surtout pour des éléments de décor dans des interludes cinématiques entre les niveaux. Il serait pertinent de voir ces techniques se généraliser et leur utilisation s'intensifier afin d'offrir une qualité d'image accrue dans des situations temps réel.

# BIBLIOGRAPHIE

[Bentley1975] BENTLEY, JON LOUIS. 1975. «Multidimensional binary search trees used for associative searching». Communications of the ACM. 18: 9. pp. 509-517.

[Bentley1990] BENTLEY, JON LOUIS. 1990. «K-d trees for semidynamic point sets». *SCG '90: Proceedings of the sixth annual symposium on Computational geometry*. New York, NY, USA : ACM Press. 1, pp. 187-197.

[Blinn1976] BLINN, JAMES F ET NEWELL, MARTIN E. 1976. «Texture and reflection in computer generated images». Communications of the ACM. 19: 10. pp. 542-547.

[Blinn1977] BLINN, JAMES F. 1977. «Models of light reflection for computer synthesized pictures». *SIGGRAPH '77: Proceedings of the 4th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 192-198.

[Carr2006] CARR, NATHAN A, HOBEROCK, JARED, CRANE, KEENAN ET HART, JOHN C. 2006. «Fast GPU ray tracing of dynamic meshes using geometry images». *GI '06: Proceedings of Graphics Interface 2006*. Toronto, Ont., Canada : Canadian Information Processing Society. 1, pp. 203-209.

[Cook1981] Cook, Robert L et Torrance, Kenneth E. 1981. «A reflectance model for computer graphics». *SIGGRAPH '81: Proceedings of the 8th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 307-316.

[Debevec1998] Debevec, Paul. 1998. «Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography». *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 189-198.

[Dmitriev2002] Dmitriev, Kirill, Brabec, Stefan, Myszkowski, Karol et Seidel, Hans-Peter. 2002. «Interactive global illumination using selective photon tracing». *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*. Aire-la-Ville, Switzerland : Eurographics Association. 1, pp. 25-36.

[Goral1984] Goral, Cindy M, Torrance, Kenneth E, Greenberg, Donald P et Battaile, Bennett. 1984. «Modeling the interaction of light between diffuse surfaces». *SIGGRAPH '84: Proceedings of the 11th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 18, pp. 213-222.

[Granier2004] Granier, Xavier et Drettakis, George. 2004. «A final reconstruction approach for a unified global illumination algorithm». ACM

Transactions on Graphics. 23: 2. pp. 163-189.

[Green2003] GREEN, ROBIN. 2003. «Spherical Harmonic Lighting: The Gritty Details». Archives of the Game Developers Conference. 1: 1. pp. 1-47.

[Hasan2006] HASAN, MILOS, PELLACINI, FABIO ET BALA, KAVITA. 2006. «Direct-to-indirect transfer for cinematic relighting». *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*. New York, NY, USA : ACM Press. 1, pp. 1089-1097.

[He1991] HE, XIAO D, TORRANCE, KENNETH E, SILLION, FRANÇOIS X ET GREENBERG, DONALD P. 1991. «A comprehensive physical model for light reflection». *SIGGRAPH '91: Proceedings of the 18th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 175-186.

[Ivanic1996] IVANIC, JOSEPH ET RUEDENBERG, KLAUS. 1996. «Rotation Matrices for Real Spherical Harmonics. Direct Determination by Recursion». Journal of Physical Chemistry. 100: 15. pp. 6342-6347.

[Jensen1996] JENSEN, HENRIK W. 1996. «Global Illumination using Photon Maps». *Rendering Techniques '96*. New York, NY, USA : Springer-Verlag Wien. 1, pp. 21-30.

[Kajiya1986] KAJIYA, JAMES T. 1986. «The rendering equation». *SIGGRAPH '86: Proceedings of the 13th annual conference on Computer graphics and interactive techniques*. New York, NY, USA :

ACM Press. 20, pp. 143-150.

[Keller1997] KELLER, ALEXANDER. 1997. «Instant radiosity». *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press/Addison-Wesley Publishing Co. 1, pp. 49-56.

[Lafortune1993] LAFORTUNE, ERIC P ET WILLEMS, YVES D. 1993. «Bi-directional Path Tracing». *Proceedings of Third International Conference on Computational Graphics and Visualization Techniques (Compugraphics '93)*. Alvor, Portugal : Compugraphics. 1, pp. 145-153.

[Lambert1760] LAMBERT, J H. 1760. *Photometria sive de mensure de gratibus luminis, colorum umbrae*. 1. Augsberg, Allemagne : Eberhard Klett. ? p.

[Ng2003] NG, REN, RAMAMOORTHI, RAVI ET HANRAHAN, PAT. 2003. «All-frequency shadows using non-linear wavelet lighting approximation». ACM Transactions on Graphics. 22: 3. pp. 376-381.

[Ng2004] NG, REN, RAMAMOORTHI, RAVI ET HANRAHAN, PAT. 2004. «Triple product wavelet integrals for all-frequency relighting». ACM Transactions on Graphics. 23: 3. pp. 477-487.

[Oren1994] OREN, MICHAEL ET NAYAR, SHREE K. 1994. «Generalization of Lambert's reflectance model». *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 239-246.

[Osfield2007] OSFIELD, ROBERT ET CONTRIBUTEURS. 2008. *OpenSceneGraph*. (2.4) [Logiciel]. Perthshire, Scotland : OpenSceneGraph Professional Services. Bibliothèque graphique temps réel et graphe de scène basé sur OpenGL. http://www.openscenegraph.org/

[Pellacini2005] PELLACINI, FABIO, VIDIMCE, KIRIL, LEFOHN, AARON, MOHR, ALEX, LEONE, MARK ET WARREN, JOHN. 2005. «Lpics: a hybrid hardware-accelerated relighting engine for computer cinematography». ACM Transactions on Graphics. 24: 3. pp. 464-470.

[Phong1975] PHONG, BUI-THONG. 1975. «Illumination for computer generated images». Communications of the ACM. 18: 6. pp. 311-317.

[Poulin1990] POULIN, PIERRE ET FOURNIER, ALAIN. 1990. «A model for anisotropic reflection». *SIGGRAPH '90: Proceedings of the 17th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 273-282.

[Proudfoot2001] PROUDFOOT, KEKOA, MARK, WILLIAM R, TZVETKOV, SVETOSLAV ET HANRAHAN, PAT. 2001. «A real-time procedural shading system for programmable graphics hardware». *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 1, pp. 159-170.

[Purcell2003] PURCELL, TIMOTHY J, DONNER, CRAIG, CAMMARANO, MIKE, JENSEN, HENRIK WANN ET HANRAHAN, PAT. 2003. «Photon Mapping on Pro-

grammable Graphics Hardware». *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*. San Diego, California : Eurographics Association. 1, pp. 41-50.

[Ren2006] REN, ZHONG, WANG, RUI, SNYDER, JOHN, ZHOU, KUN, LIU, XINGUO, SUN, BO, SLOAN, PETER-PIKE, BAO, HUJUN, PENG, QUNSHENG ET GUO, BAINING. 2006. «Real-time soft shadows in dynamic scenes using spherical harmonic exponentiation». *SIGGRAPH '06: ACM SIGGRAPH 2006 Papers*. New York, NY, USA : ACM Press. 1, pp. 977-986.

[Rost2006] ROST, RANDI J. 2006. *OpenGL(R) Shading Language (2nd Edition)*. 2e édition. Toronto, Ontario, Canada : Addison-Wesley Professional. 740 p.

[Segal1992] SEGAL, MARK, KOROBKIN, CARL, VAN WIDENFELT, ROLF, FORAN, JIM ET HAEBERLI, PAUL. 1992. «Fast shadows and lighting effects using texture mapping». *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 26, pp. 249-252.

[Sloan2002] SLOAN, PETER-PIKE, KAUTZ, JAN ET SNYDER, JOHN. 2002. «Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments». *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 21, pp.

527-536.

[Sloan2003] SLOAN, PETER-PIKE, HALL, JESSE, HART, JOHN ET SNYDER, JOHN. 2003. «Clustered principal components for precomputed radiance transfer». ACM Transactions on Graphics. 22: 3. pp. 382-391.

[Sloan2003A] SLOAN, PETER-PIKE, LIU, XINGUO, SHUM, HEUNG-YEUNG ET SNYDER, JOHN. 2003. «Bi-scale radiance transfer». ACM Transactions on Graphics. 22: 3. pp. 370-375.

[Sloan2005] SLOAN, PETER-PIKE, LUNA, BEN ET SNYDER, JOHN. 2005. «Local, deformable precomputed radiance transfer». ACM Transactions on Graphics. 24: 3. pp. 1216-1224.

[Sloan2006] SLOAN, PETER-PIKE. 2006. «Normal mapping for precomputed radiance transfer». *SI3D '06: Proceedings of the 2006 symposium on Interactive 3D graphics and games*. New York, NY, USA : ACM Press. 1, pp. 23-26.

[Tole2002] TOLE, PARAG, PELLACINI, FABIO, WALTER, BRUCE ET GREENBERG, DONALD P. 2002. «Interactive global illumination in dynamic scenes». *SIGGRAPH '02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*. New York, NY, USA : ACM Press. 21, pp. 537-546.

[Veach1997] VEACH, ERIC ET GUIBAS, LEONIDAS J. 1997. «Metropolis light transport». *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. New

York, NY, USA : ACM Press/Addison-Wesley Publishing Co. 1, pp. 65-76.

[Wald2002] WALD, INGO, KOLLIG, THOMAS, BENTHIN, CARSTEN, KELLER, ALEXANDER ET SLUSALLEK, PHILIPP. 2002. «Interactive global illumination using fast ray tracing». *EGRW '02: Proceedings of the 13th Eurographics workshop on Rendering*. Aire-la-Ville, Switzerland : Eurographics Association. 1, pp. 15-24.

[Wald2003] WALD, INGO, BENTHIN, CARSTEN ET SLUSALLEK, PHILIPP. 2003. «Interactive global illumination in complex and highly occluded environments». *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*. Aire-la-Ville, Switzerland : Eurographics Association. 1, pp. 74-81.

[Wald2007] WALD, INGO, BOULOS, SOLOMON ET SHIRLEY, PETER. 2007. «Ray Tracing Deformable Scenes using Dynamic Bounding Volume Hierarchies». ACM Transactions on Graphics. 26: 1. p. 6.

[Walter1999] WALTER, BRUCE, DRETTAKIS, GEORGE ET PARKER, STEVEN. 1999. «Interactive Rendering using the Render Cache». *Rendering techniques '99 (Proceedings of the 10th Eurographics Workshop on Rendering)*. New York, NY : Springer-Verlag/Wien. 10, pp. 235-246.

[Walter2002] WALTER, BRUCE, DRETTAKIS, GEORGE ET GREENBERG, DONALD P. 2002. «Enhancing and optimizing the render cache». *EGRW '02: Pro-*

*ceedings of the 13th Eurographics workshop on Rendering*. Aire-la-Ville, Switzerland : Eurographics Association. 1, pp. 37-42.