

UNIVERSITÉ DE LIMOGES

ÉCOLE DOCTORALE Sciences et Ingénierie pour l'Information

FACULTÉ DES SCIENCES ET TECHNIQUES

Année : 2019

Thèse N°X

Thèse

pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE LIMOGES

Discipline : Informatique et applications

Xavier Chermain

27 novembre 2019

Rendu basé physique de micro-reflets

Thèse dirigée par Frédéric Claux et Stéphane Mérillou

JURY :

| | | |
|--------------------------|---|-----------------------|
| Mathias PAULIN | Professeur des Universités, Université de Toulouse | Rapporteur |
| Pascal BARLA | Chargé de Recherche HDR, Inria Bordeaux | Rapporteur |
| Éric HEITZ | Chercheur, Unity Technologies | Examineur |
| Stéphane MÉRILLOU | Professeur des Universités, Université de Limoges | Directeur de thèse |
| Frédéric CLAUX | Maître de conférences, Université de Limoges | Co-directeur de thèse |
| Tamy BOUBEKEUR | Directeur de recherche, Adobe, Professeur, Télécom Paris | Président du jury |

Remerciements

J'aimerais remercier mon encadrant Frédéric Claux pour avoir toujours été présent. Sa persévérance, sa dévotion, sa passion ainsi que sa réactivité ont grandement contribué à l'aboutissement de cette thèse.

Je remercie aussi mon autre encadrant Stéphane Mérillou pour m'avoir accordé sa confiance aveugle sur ce projet, avant même qu'il ne commence.

Je tiens aussi à remercier Mathias Paulin, Pascal Barla, Tamy Boubekeur et Éric Heitz d'avoir accepté d'être dans mon jury de thèse.

Ce travail n'aurait pas pu être accompli sans l'aide technique et théorique de mes collègues et amis : Arthur Cavalier, Adrien Hauteville et Florent Nacry.

J'aimerais aussi remercier ma famille et Laura, pour leur soutien durant toutes ces années.

Table des matières

| | |
|---|-----------|
| Table des matières | 1 |
| Table des figures | 4 |
| Table des figures | 4 |
| Chapitre 1 : Introduction au rendu de micro-reflets | 7 |
| 1.1 Rendu et synthèse d'images | 8 |
| 1.2 Interactions lumière-surface | 10 |
| 1.3 Notion d'échelle dans le rendu | 11 |
| 1.3.1 Empreinte de rayon | 12 |
| 1.3.2 Surfaces augmentées | 14 |
| 1.4 Rendu de micro-reflets | 15 |
| 1.5 La conservation de l'énergie comme critère de qualité | 17 |
| 1.6 Objectif de la thèse | 18 |
| 1.7 Contributions | 18 |
| Chapitre 2 : État des connaissances dans le rendu de micro-reflets | 21 |
| 2.1 Rappels généraux sur les BRDFs | 22 |
| 2.1.1 Configuration géométrique | 22 |
| 2.1.2 Quantités énergétiques | 23 |
| 2.1.3 Équation d'illumination locale | 25 |
| 2.1.4 Définition et propriétés des BRDFs | 26 |
| 2.1.5 BRDFs dans un moteur de rendu basé physique | 28 |
| 2.2 Les modèles de BRDFs idéaux | 30 |
| 2.3 Modéliser l'absorption des matériaux | 31 |
| 2.4 Modèles de BRDFs à base de micro-facettes | 32 |
| 2.4.1 Distribution de normales (NDF) | 34 |
| 2.4.2 Distribution de pentes | 34 |
| 2.4.3 Propriétés des NDFs | 35 |
| 2.4.4 Distribution de Beckmann | 36 |
| 2.4.5 Distribution GGX | 36 |
| 2.4.6 Distributions anisotropes | 37 |
| 2.4.7 Distributions anisotropes non-alignées sur les axes | 38 |
| 2.4.8 Distributions GTR, STD et autres | 39 |
| 2.4.9 Distribution de normales visibles (VNDF) | 39 |
| 2.4.10 Fonctions de masquage | 40 |
| 2.4.11 Masquage-ombrage et simple rebond | 43 |
| 2.4.12 Multi-rebonds dans les micro-facettes | 44 |
| 2.5 Rendu de cartes de normales | 48 |
| 2.5.1 Distributions décentrées | 50 |
| 2.5.2 Distributions à deux normales | 52 |
| 2.6 Modèles de BRDFs surfaciques pour le rendu de micro-reflets | 55 |
| 2.6.1 Empreinte de rayon | 56 |
| 2.6.2 Distribution de normales pour une empreinte (\mathcal{P} -NDF) | 57 |

| | | |
|--|--|------------|
| 2.6.3 | Matériaux à micro-reflets basés cartes de normales | 59 |
| 2.6.4 | Matériaux à micro-reflets non basés cartes de normales | 63 |
| 2.7 | Objectif formel de la thèse | 64 |
| Chapitre 3 : Rendu efficace et précis de cartes de normales spéculaires et haute définition | | 65 |
| 3.1 | Introduction | 66 |
| 3.2 | Travaux précédents | 67 |
| 3.2.1 | Surfaces pailletées | 67 |
| 3.2.2 | Surfaces rayées | 67 |
| 3.2.3 | Surfaces basées sur une carte de normales | 68 |
| 3.3 | Définition de la micro-surface | 68 |
| 3.3.1 | Représentation des normales par des pentes | 68 |
| 3.3.2 | Mixture de distributions de normales de Beckmann | 69 |
| 3.3.3 | Reconstruction locale de NDF | 70 |
| 3.4 | \mathcal{P} -BRDF | 71 |
| 3.4.1 | \mathcal{P} -NDF | 71 |
| 3.4.2 | Masquage-ombrage de Smith surfacique | 72 |
| 3.5 | Résultats et discussions | 75 |
| 3.5.1 | \mathcal{P} -NDF | 76 |
| 3.5.2 | Rendus | 76 |
| 3.6 | Conclusion | 79 |
| Chapitre 4 : Rendu de micro-reflets basé sur une BRDF surfacique et multi-rebonds | | 83 |
| 4.1 | Introduction | 84 |
| 4.2 | Travaux connexes | 86 |
| 4.2.1 | Intégrateurs de micro-reflets | 86 |
| 4.2.2 | Rendu de cartes de normales basé micro-facettes | 87 |
| 4.2.3 | BRDFs multi-rebonds | 89 |
| 4.3 | BRDF multi-rebonds locale | 89 |
| 4.3.1 | Aperçu du modèle de micro-surface | 89 |
| 4.3.2 | Distribution de normales et aires projetées | 89 |
| 4.3.3 | Masquage et ombrage | 90 |
| 4.3.4 | BRDF simple rebond | 91 |
| 4.3.5 | BRDF multi-rebonds | 91 |
| 4.3.6 | Échantillonnage préférentiel de la BRDF locale | 93 |
| 4.4 | BRDF surfacique multi-rebonds | 94 |
| 4.4.1 | Définition de la BRDF surfacique | 94 |
| 4.4.2 | Évaluation de la BRDF surfacique | 94 |
| 4.4.3 | Échantillonnage de la BRDF surfacique | 97 |
| 4.5 | Résultats et discussions | 98 |
| 4.5.1 | Validation de la BRDF locale | 98 |
| 4.5.2 | Résultats utilisant la BRDF surfacique | 99 |
| 4.6 | Conclusion | 106 |
| Chapitre 5 : Conclusion | | 107 |

| | |
|--|------------|
| Annexe A : Code des matériaux pbrt-v3 | 111 |
| A.1 Introduction | 112 |
| A.2 Rendu de cartes de normales | 112 |
| A.2.1 Classique | 112 |
| A.2.2 Rendu de cartes de normales utilisant des facettes symétriques | 114 |
| A.3 Intégrateurs de micro-reflets | 115 |
| A.3.1 Simple rebond | 116 |
| A.3.2 Multi-rebonds | 118 |
| A.4 Dépendances additionnelles | 120 |
| A.4.1 Tracer de chemins compatible avec les poids d'échantillons | 120 |

Table des figures

| | | |
|------|--|----|
| 1.1 | La synthèse d'images | 8 |
| 1.2 | Interaction lumière-surface | 10 |
| 1.3 | Limitations des modèles classiques | 11 |
| 1.4 | Empreintes de rayons | 13 |
| 1.5 | Surfaces augmentées | 14 |
| 1.6 | Les détails diffus et spéculaires | 16 |
| 1.7 | La conservation de l'énergie | 17 |
| 1.8 | Comparaison entre les méthodes | 18 |
| 2.1 | Configuration géométrique | 23 |
| 2.2 | Quantités radiométriques | 24 |
| 2.3 | Équation d'illumination locale | 26 |
| 2.4 | Illustration d'une BRDF | 26 |
| 2.5 | Comparaisons BRDFs simple et multi-rebonds | 28 |
| 2.6 | Implémentation d'une BRDF | 29 |
| 2.7 | Les modèles idéaux | 31 |
| 2.8 | Modéliser l'absorption | 32 |
| 2.9 | Représentation spatiale et statistique d'une micro-surface | 34 |
| 2.10 | Normalisation sur la macro-surface de la NDF | 36 |
| 2.11 | Rugosité de Beckmann et de GGX | 37 |
| 2.12 | Distributions anisotropiques | 38 |
| 2.13 | Comparaison entre GGX et Beckmann | 39 |
| 2.14 | Comparaison NDF et VNDF | 40 |
| 2.15 | Modèle à V-cavités | 41 |
| 2.16 | La surface de Smith | 42 |
| 2.17 | L'ombrage | 43 |
| 2.18 | Les BRDFs classiques perdent de l'énergie | 45 |
| 2.19 | Les marches aléatoires | 45 |
| 2.20 | La compensation d'énergie | 46 |
| 2.21 | Mise à l'échelle d'une BRDF simple rebond | 47 |
| 2.22 | Carte de normales | 49 |
| 2.23 | Problèmes avec les cartes de normales | 49 |
| 2.24 | Distribution de Beckmann décentrée | 50 |
| 2.25 | Rendus d'une carte de normales avec des NDFs décentrées | 51 |
| 2.26 | Micro-surface en un point d'une carte de normales classique et à micro-facettes | 52 |
| 2.27 | Probabilités d'intersections des facettes pour la méthode de Schüssler. | 53 |
| 2.28 | Rendus avec la méthode de Schüssler et al. | 55 |
| 2.29 | Construction d'une empreinte de rayon. | 57 |
| 2.30 | Visualisation de différentes \mathcal{P} -NDFs | 58 |
| 2.31 | Comparaisons entre des P-NDFs exactes et leur approximation obtenue avec la méthode de Yan et al. 2016 | 61 |
| 2.32 | Rendus avec la méthode de Yan et al. 2016 | 62 |
| 3.1 | Problème lié à l'échantillonnage naïf | 66 |
| 3.2 | Distribution de normales locale | 70 |
| 3.3 | Influence du masquage sur le rendu | 73 |

| | | |
|------|--|-----|
| 3.4 | Comparaison de \mathcal{P} -NDFs | 77 |
| 3.5 | Comparaison de l'impact sur le rendu de notre \mathcal{P} -NDF et celle de Yan et al. | 78 |
| 3.6 | Cartes de normales utilisées dans le chapitre | 79 |
| 3.7 | Comparaison entre notre masquage approximatif et exact | 80 |
| 3.8 | Comparaison de différents rendus de vans | 81 |
| 3.9 | Comparaison de différents rendus d'avions | 81 |
| 3.10 | Comparaisons de différents rendus d'orbes rayés | 82 |
| 4.1 | Aguiche du chapitre 4 | 85 |
| 4.2 | Représentation du modèle de carte de normales classique, de celui de Schüssler et al. et du nôtre. | 88 |
| 4.3 | Comparaison entre la BRDF de Schüssler et la nôtre | 88 |
| 4.4 | Illustration des probabilités d'intersection et de la fonction d'ombrage. | 90 |
| 4.5 | Différents cas d'échantillonnage de la BRDF multi-rebonds locale. | 93 |
| 4.6 | Ensemble discret de BRDFs pondérées | 95 |
| 4.7 | Pré-calculs d'intégrales pour évaluer l'énergie perdue dans le patch | 97 |
| 4.8 | Échantillonnage par visibilité d'une carte de normales | 98 |
| 4.9 | Comparaisons entre une carte de normales classique et symétrique. | 100 |
| 4.10 | Des couverts de plusieurs matériaux utilisant notre méthode | 101 |
| 4.11 | Comparaison entre le modèle exact et approximé | 102 |
| 4.12 | Rendus montrant la disparition des fireflies avec notre méthode. | 103 |
| 4.13 | Compensation diffuse et spéculaire | 104 |
| 4.14 | Comparaisons entre notre méthode et les précédentes pour des surfaces rayées. | 105 |
| 4.15 | Limitations de l'approximation de l'énergie perdue dans le patch | 106 |
| A.1 | Rendus et tests du four blanc d'une carte de normales classique et basée micro-facettes. | 113 |
| A.2 | Rendus avec notre méthode d'intégration de micro-reflets, et les précédentes. | 116 |

Chapitre 1 :

Introduction au rendu de micro-reflets

Le rendu de micro-reflets est un domaine récent qui fait appel à plusieurs notions en informatique graphique et en transport de la lumière. Dans ce chapitre d'introduction, nous présentons de manière intuitive ce domaine et ses notions relatives. Nous repoussons les définitions mathématiques et l'état des connaissances techniques au chapitre 2 dans le but de donner ici un aperçu léger et global des travaux effectués durant cette thèse. Nous commençons par présenter ce qu'est le rendu et la synthèse d'images dans la section 1.1. Une partie importante du rendu et de ce manuscrit traite des interactions lumière-surface (section 1.2) qui sont liées à la notion d'échelle géométrique (section 1.3). Nous expliquons ce qu'est le rendu de micro-reflets dans la section 1.4 et nous introduisons le concept de conservation de l'énergie dans la section 1.5. Finalement, nous expliquons l'objectif de la thèse, ses contributions associées et son plan dans la section 1.6.

1.1 Rendu et synthèse d'images

Le rendu est un processus informatique qui consiste à produire une image d'une scène 3D virtuelle (figure 1.1, gauche). Une analogie peut être faite avec l'action de peindre un paysage ou un portrait, où le peintre observe son environnement et le retranscrit sur sa toile, avec plus ou moins de réalisme (figure 1.1, droite). Comme pour la peinture, il existe différentes façons de faire du rendu. Certaines méthodes s'attachent à reproduire la réalité de la manière la plus fidèle possible, d'autres s'attachent à produire un effet artistique. Nous pouvons faire le corollaire avec les différents mouvements artistiques, comme le réalisme, l'impressionnisme ou le cubisme. Le rendu peut être interactif, avec comme application principale le jeu vidéo. À contrario, les images de synthèse dans les films sont souvent issues d'un rendu non-interactif.

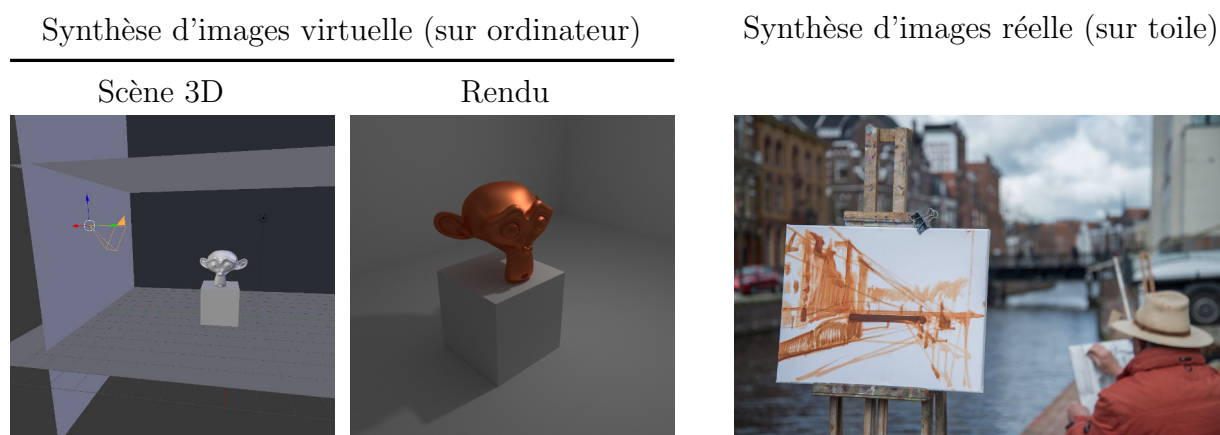


FIGURE 1.1 – La synthèse d'images sur ordinateur (gauche, avec Blender [Roo18]) peut facilement être vue comme l'action de peindre sur une toile ce que nous voyons (droite).

Rendu photoréaliste et perception. Le rendu photoréaliste a pour but de produire des images par ordinateur qui tendent à ressembler à une photo. La comparaison se fait à scène 3D équivalente, par exemple une sphère dans une boîte, avec un certain matériau et sous un certain éclairage. Si tous les paramètres sont identiques dans la scène virtuelle, alors l'image de synthèse doit être identique à la photo. Malheureusement,

il est impossible de reproduire exactement une scène 3D réelle car il y a énormément de paramètres (beaucoup de lumières complexes, beaucoup de matériaux différents en tout point de l'espace). La mesure du réalisme se fait donc par la capacité de l'image de synthèse à tromper l'observateur, en lui faisant croire que l'image qu'il observe est une vraie photo. C'est l'objectif principal du rendu photoréaliste, mais la métrique est très subjective. D'un observateur à un autre, la perception de la réalité peut être très différente. De plus, l'observation se fait souvent à travers du matériel qui est de plus ou moins bonne qualité : écran d'ordinateur ou de télévision, casque de réalité virtuelle, etc. Ces équipements sont limités dans la reproduction du spectre visuel. Parfois, c'est l'observateur qui a des problèmes de vue (par exemple le daltonisme). Les étapes liées au jugement du réalisme des images rendus sur ordinateur sont donc beaucoup trop sujet à interprétation en utilisant comme seule métrique la perception humaine. Pour éviter ce problème, une autre métrique peut être utilisée, qui elle est basée sur la physique.

Rendu basé physique. Les problèmes liés à la perception du réalisme d'une image ont été en partie résolus grâce au rendu basé physique. Durant la dernière décennie, ce type de rendu a reçu un intérêt croissant en informatique graphique [PJH16, GA19], car il cherche à simuler réellement la propagation de la lumière sur ordinateur en analysant, dans la vie réelle, comment un rayon lumineux interagit avec la matière. Beaucoup de concepts issus du domaine de la physique sont utilisés : la luminance énergétique, la longueur d'onde de la lumière, le coefficient d'absorption d'un matériau ou son indice de réfraction, etc. Lors d'un rendu basé physique, nous cherchons à connaître l'énergie arrivant dans le pixel d'une caméra virtuelle, plutôt qu'à lui donner une « couleur ». Nous attribuons des paramètres physiques aux objets, comme la rugosité, plutôt qu'une taille de lobe spéculaire. Juger du réalisme en utilisant la physique est un processus plus objectif que de juger du réalisme en utilisant la perception humaine.

Algorithme du lancer de rayons. Actuellement, le rendu basé physique s'appuie sur l'algorithme du lancer de rayons. Cet algorithme est simple : il consiste à lancer des rayons lumineux depuis la caméra dans une scène 3D virtuelle. Ensuite, l'ordinateur simule les rebonds et les interactions avec les objets, jusqu'à une potentielle source de lumière. En retraçant le chemin inverse et en connaissant la quantité d'énergie qui a été absorbée à chaque rebond lumineux, nous pouvons connaître la luminance énergétique arrivant à chaque pixel de l'image.

Même s'il existe plusieurs façons de coder ce type d'algorithme, les objets et phénomènes suivants sont souvent modélisés :

Les lumières ont une importance capitale dans un moteur de rendu. Sans elles, nous ne pouvons pas voir les objets. Il faut donc spécifier leur position, leur forme, leur type et la façon dont elles distribuent l'énergie et les rayons lumineux dans l'environnement.

Les intersections rayon-objet permettent de connaître où les rayons lumineux rencontrent pour la première fois un objet. Au point d'intersection, les informations de l'objet touché sont utilisées pour continuer la propagation de la lumière. L'algorithme devra donc fournir des fonctions d'intersections rayon-objet.

La caméra reçoit les rayons lumineux avec des informations énergétiques (de couleurs). La caméra a une position dans l'espace et une orientation. C'est à travers

elle que nous voyons la scène 3D.

Les interactions lumière-surface qui ont lieu sur les objets sont simulées en utilisant les propriétés microscopiques des matériaux qui les composent (figure 1.2). Ce sont les caractéristiques du matériau qui permettent de savoir comment la lumière est *répartie* après une intersection.

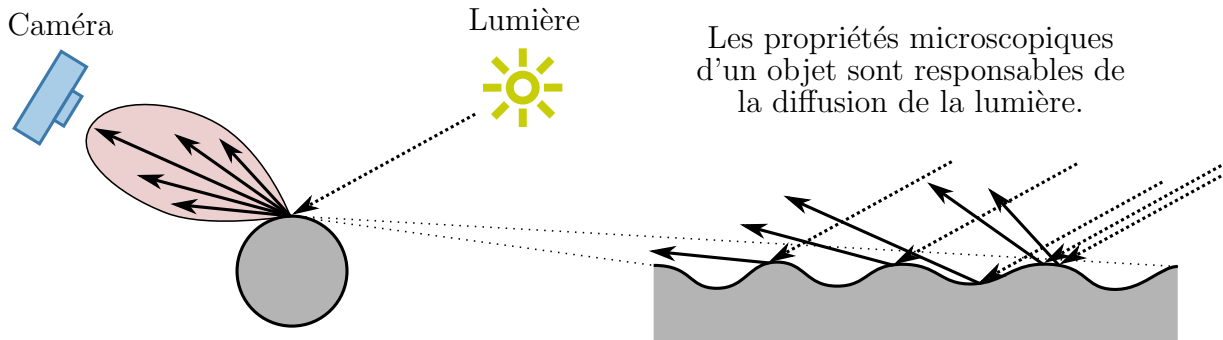


FIGURE 1.2 – Un rayon lumineux se diffuse après avoir été au contact d'un matériau d'un objet (gauche). Cette diffusion est due à sa micro-surface (droite).

Cette liste n'est pas exhaustive et d'autres phénomènes peuvent être modélisés, comme la propagation de la lumière dans des volumes de matière. Dans cette thèse, nous nous concentrons sur les *interactions lumière-surface* (le dernier point). Cette composante essentielle du lancer de rayons permet de modéliser l'apparence des objets (métallique, plastique, rayée, rugueuse, etc.). Nous nous attacherons aussi à la réalité physique que peut avoir notre modélisation de l'apparence.

1.2 Interactions lumière-surface

Dans un algorithme de lancer de rayons, les interactions lumière-surface sont gérées par des *modèles de réflexion*. Ces modèles seront abordés plus en détail dans le chapitre 2. Ils décrivent comment la lumière se disperse au niveau d'un point sur la surface. Pour une longueur d'onde de la lumière donnée (dit autrement, pour une couleur donnée), ils donnent la *distribution de réflectance* (le lobe rouge de la figure 1.2). Par exemple, pour un miroir parfait, cette distribution est un Dirac. C'est-à-dire que l'énergie lumineuse qui arrive depuis une direction incidente est reflétée dans une seule direction : la direction de réflexion par rapport à la normale géométrique. À l'inverse, un mur en béton aura plutôt tendance à distribuer de manière uniforme la lumière incidente. Le matériau qui compose l'objet et sa microstructure vont directement impacter la forme de la distribution lumineuse.

Données mesurées. Les distributions de réflectance peuvent être mesurées dans le monde réel pour ensuite être utilisées dans un moteur de rendu [MPBM03a, MPBM03b, DJ18]. Des bases de données sont disponibles contenant des mesures pour des matériaux classiques, comme du plastique, du métal et du bois, mais aussi pour des matériaux moins courants comme le papier, la soie ou les ailes des papillons. Ces mesures sont utilisées soit pour faire le rendu d'un matériau en particulier, soit pour tester l'exactitude d'un modèle analytique, moins coûteux en mémoire et plus versatile. Actuellement, il est encore difficile

de faire la capture de la distribution de réflectance d'un matériau car elle nécessite un matériel assez sophistiqué. De plus, le temps de capture peut être long et les données en mémoire importantes car il faut scanner un domaine 4D, possédant plus ou moins de hautes fréquences. Pour finir, les artistes n'ont pas de contrôle rapide sur l'apparence. Par exemple, pour changer la couleur d'un bois, il faudrait faire une nouvelle mesure sur un échantillon qui a la couleur voulue. Les modèles de réflexion peuvent aussi être mesurés à partir d'une surface *virtuelle*, créée sur ordinateur [WAT92]. Ces données simulées sont aussi un bon moyen de valider un modèle analytique [HHdD16].

Modèles analytiques. En utilisant des hypothèses simples sur la micro-géométrie et le matériau de la surface, la distribution de réflectance peut être modélisée par des expressions mathématiques. Un des modèles les plus utilisés en informatique graphique [CT81] suppose une distribution gaussienne des normales de la surface et un matériau parfaitement spéculaire (miroir parfait). L'utilisateur peut contrôler la rugosité du matériau en modifiant l'écart-type de la distribution gaussienne. Ce type de modèle simule bien la diffusion de la lumière sur plusieurs types de matériaux, comme le métal et le plastique.

C'est pour ces raisons que nous avons fait le choix de travailler sur des modèles analytiques et non pas sur des modèles utilisant des données mesurées de distribution de réflectances. De plus, il reste beaucoup de problèmes ouverts dans ce domaine, car ils supposent souvent des matériaux sans défaut et vus d'assez loin, pour que l'approximation gaussienne de la distribution des normales de la surface soit correcte (figure 1.3, gauche). Actuellement, peu de travaux ont proposés des solutions face à cette limitation qui arrive fréquemment dans la vie de tous les jours. Par exemple, à niveau de zoom moyen, où nous voyons individuellement les bosses de la surface (figure 1.3, milieu), ou des défauts comme des rayures ou des accros (figure 1.3, droite).

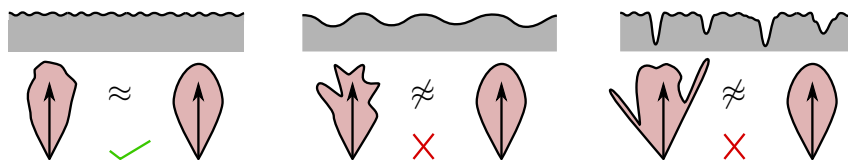


FIGURE 1.3 – Différents types de micro-surfaces (haut) possédant différentes distributions de normales (bas), où l'approximation gaussienne est plus ou moins efficace.

1.3 Notion d'échelle dans le rendu

L'étude de la micro-surface des matériaux pour en déduire des modèles de réflexion de la lumière se fait pour une aire de surface donnée (figure 1.3). L'apparence d'un objet change si nous sommes plus ou moins loin de lui. En se rapprochant, nous observons souvent beaucoup de détails. En nous éloignant, ces détails peuvent devenir invisibles. La notion d'échelle est donc importante et doit être prise en compte lors du rendu. La gestion classique des distances implique une échelle définie de manière absolue, où la géométrie a différentes représentations en fonction de sa taille [WAT92] :

La micro-géométrie, de l'ordre du micromètre, est souvent modélisée par une distribution de normales avec un ou deux paramètres. C'est l'échelle microscopique.

La méso-géométrie, de l'ordre du millimètre, est dans la majorité des moteurs de rendu représentée par une carte de hauteurs ou de normales, qui est plaquée sur la surface d'un objet. C'est l'échelle mésoscopique.

La macro-géométrie, supérieure au millimètre, est modélisée dans une scène 3D par une soupe de triangles, aussi appelée maillage, ou par des fonctions analytiques comme des courbes de Bézier. C'est l'échelle macroscopique.

Traditionnellement, les modèles de réflexion simulent la diffusion de la lumière sur de la micro-géométrie, à l'échelle microscopique. Ils donnent en moyenne la quantité de luminance sortante dans une direction. C'est comme si la micro-géométrie avait été explicitement modélisée, puis bombardée de rayons dans le but de connaître le ratio entre les rayons sortant dans une direction donnée et le nombre total de rayons incidents. À contrario, pour l'échelle macroscopique, des rayons lumineux sont explicitement lancés dans la scène 3D pour connaître globalement l'éclairage. Pour la diffusion de la lumière à l'échelle mésoscopique avec une carte de hauteurs ou de normales, différentes techniques existent, utilisant soit une représentation explicite de la méso-géométrie [TIS08], soit en la simulant implicitement dans le modèle de réflexion [Bli78, SHHD17]. Certaines techniques mixent même les deux [DHI⁺13].

En résumé, l'algorithme du lancer de rayons fait la distinction entre la macro-échelle qui est modélisée explicitement par des formes ou des maillages et la micro-échelle qui est gérée par les modèles de diffusion de la lumière. Les rayons lumineux intersectent la macro-géométrie (figure 1.2, gauche) et sont diffusés selon la micro-géométrie (figure 1.2, droite). La méso-échelle est gérée soit par de la géométrie, soit par des modèles de réflexions (ou un peu des deux).

Échelle relative au pixel. Précédemment, l'échelle de la géométrie a été introduite en utilisant des distances absolues. Cette représentation des ordres de grandeur est intuitive, mais en réalité elle n'est pas utilisée comme telle. Les distances vont être définies par rapport à l'observateur. Est-ce qu'un artiste aura besoin de modéliser tous les cratères et cailloux de la Lune pour en faire le rendu depuis la Terre ? Il se contentera au mieux de la considérer géométriquement comme une sphère, possédant une micro-géométrie très rugueuse et diffuse. En informatique graphique, il est courant de considérer que tout ce qui est sous pixel est de la micro-géométrie modélisée par une distribution de réflectance, et tout ce qui est sur pixel est de la macro-géométrie. Par exemple, dans l'industrie, le moteur de rendu photoréaliste Pixar RenderMan [Pix18] subdivise les cartes de déplacements et les NURBS en polygones de la taille d'un pixel de l'écran (les micro-polygones projetés sur l'écran ont une taille inférieure ou égale au pixel).

1.3.1 Empreinte de rayon

Nos travaux portent sur la définition d'un modèle de réflexion de la lumière prenant en compte toute la méso-géométrie sous pixel. Lors de la synthèse d'une image, nous cherchons à connaître la quantité d'énergie arrivant dans chaque pixel. Pour cela, nous tirons des rayons depuis les capteurs de la caméra virtuelle. Ces rayons vont toucher plusieurs positions d'une surface. Si la surface est uniforme et possède peu de détails,

alors l'apparence de l'objet sera correctement rendue (figure 1.4, gauche). Si la surface possède énormément de détails sous pixel (figure 1.4, milieu), comme des rayures, alors il faudra l'échantillonner jusqu'à ce que tous les détails soient retranscrits (la fréquence d'échantillonnage doit être égale à deux fois la fréquence de Nyquist). Ne pas avoir assez d'échantillons cause des problèmes d'aliasing de rendu, où l'apparence est bruitée et non lisse (figure 1.4, milieu). C'est un problème d'autant plus grave lors du rendu d'une séquence d'images, où la cohérence spatiale des détails surfaciques doit être maintenue. Si le taux d'échantillonnage du pixel n'est pas augmenté, alors la micro-surface doit être filtrée. Pour cela, il faudra connaître le domaine spatial d'étude et c'est l'empreinte du rayon qui le définira. Dans cette thèse, nous utilisons cette empreinte et nous ne simplifions pas la méso-géométrie (ou surface augmentée) pour filtrer la surface. Le but est de faire des rendus sans aliasing d'une apparence complexe (figure 1.4, droite). Les reflets spéculaires peuvent avoir différentes formes et révèlent la structure surfacique de l'objet (rayures, micro-bosses).

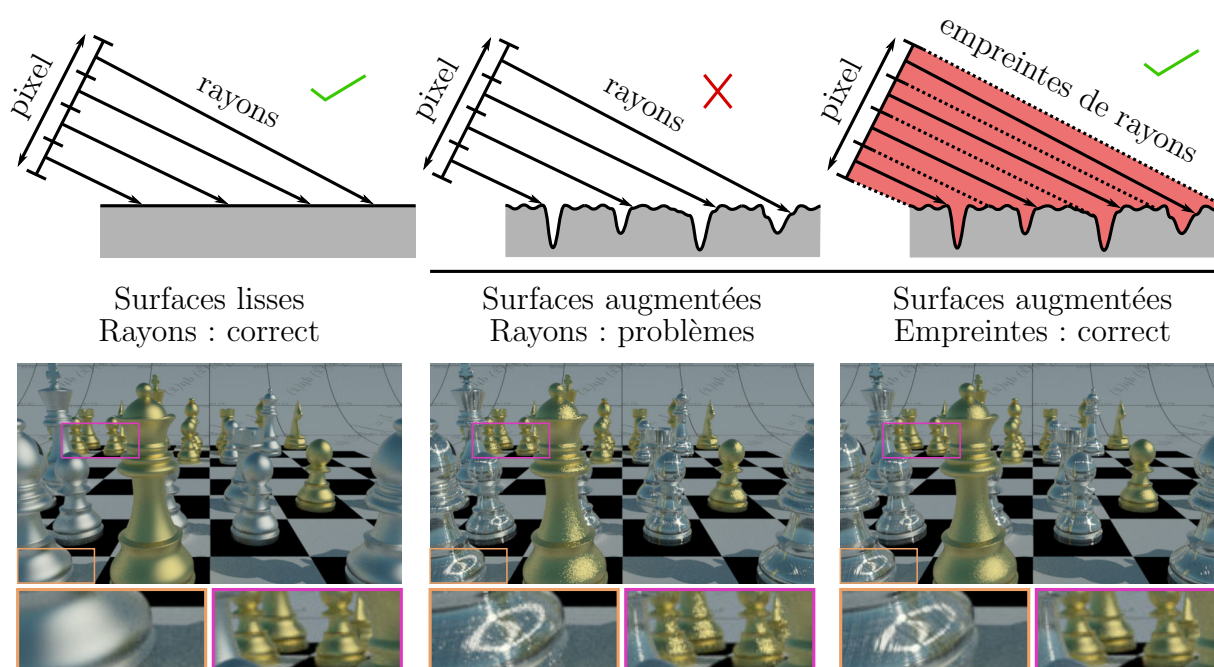


FIGURE 1.4 – Calculer l'éclairage au point d'intersection des rayons est correct si la micro-surface est basse fréquence (gauche). Certains détails peuvent être loupés s'il n'y a pas assez d'échantillons par pixel (milieu). Une technique classique d'anti-aliasing de surface utilise l'empreinte du rayon pour capter tous les détails (droite).

Calcul de l'empreinte du rayon. La projection d'une partie d'un pixel sur une surface n'est pas triviale. Le fait que le pixel ne soit pas uniformément échantillonné, que les objets soient incurvés et donc que les projections exactes aient des formes complexes n'y aide pas. La solution à ce problème est de considérer que localement, au point d'intersection, la surface est plane et que l'empreinte est représentée par un filtre de reconstruction. Par exemple, le filtre peut être gaussien, où la contribution de chaque point d'une surface est pondérée par une gaussienne centrée sur le rayon et s'étalant un peu sur les empreintes de rayon voisines [Hec89, Ige99].

Apparence multi-échelles. Les modèles de réflexion classiques, évalués pour un point sur une surface, ne sont pas multi-échelles et ne prennent pas en compte l’empreinte du rayon. Avec ces modèles, l’apparence des objets est la même de près ou de loin, car ils supposent le même nombre de détails sous pixel, quelle que soit la distance caméra-objet. Ce nombre est très grand pour que le théorème central limite, et donc, les statistiques gaussiennes, s’appliquent. Dans cette thèse, nous avons proposé des modèles qui ne dépendent plus d’un point, mais d’une aire sur une surface (une empreinte de rayon). Par définition, ils sont multi-échelles. De plus, nos modèles restent cohérents avec ceux qui sont statistiques, car vu de loin, les résultats sont égaux (figure 1.4, encarts violets).

1.3.2 Surfaces augmentées

Dans le but d’introduire plus de réalisme dans les scènes 3D, des cartes de hauteurs ou de normales sont appliquées sur les objets. Avec cette technique, des surfaces relativement plates (figure 1.4, gauche) peuvent être complexifiées (figure 1.4, droite) pour un coût mémoire inférieur à une modélisation géométrique explicite. Ces surfaces augmentées ont été introduites lors de l’émergence de l’informatique graphique [Bli78]. Aujourd’hui encore, c’est un sujet de recherche très actif [DHI⁺13, SHHD17] car il est difficile de faire le rendu sans aliasing et sans artéfact de ce type de méso-surfaces. En effet, elles introduisent souvent des hautes fréquences ou des situations difficiles à prendre en charge. Par exemple, des normales tournant le dos à la caméra alors que la géométrie, elle, est bien visible. Il existe deux types de modélisation : les cartes de déplacements ou les cartes de normales.

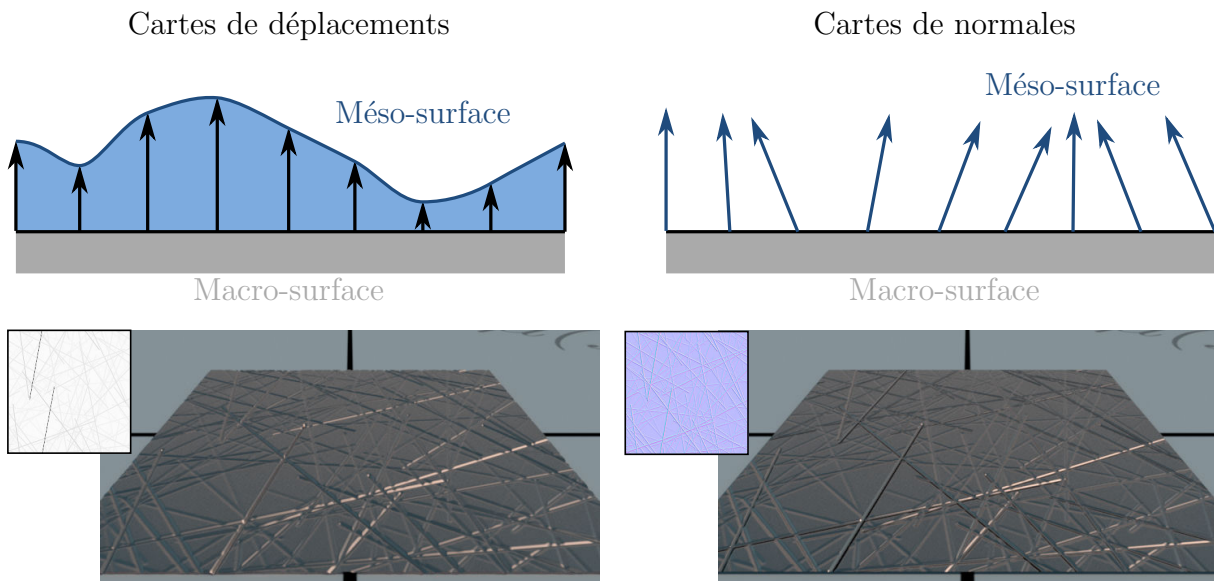


FIGURE 1.5 – Les surfaces augmentées permettent d’ajouter des détails surfaciques à moindre coût, soit en surélevant les points de la surface macroscopique (gauche), soit en perturbant la normale géométrique (droite). Les cartes de normales restent néanmoins la représentation la moins chère en temps de calcul. De plus, elles donnent des résultats proches du champ de hauteurs.

Cartes de déplacements. Cette représentation de la géométrie élève chaque point d’une surface d’une certaine valeur (figure 1.5, gauche). Souvent, ces valeurs sont contenues

dans une texture. Pour faire le rendu de cette primitive, la géométrie est reconstruite à la volée. Les rayons peuvent réellement se propager dans la géométrie et la complexité est gérée au niveau macroscopique. Cette méthode reste assez coûteuse comparée à une carte de normales mais elle donne des rendus de meilleures qualités. Dans le contexte de notre travail, où nous souhaitons faire le rendu efficace de méso-structures, cette représentation des détails est encore trop complexe et coûteuse. Elle implique une corrélation entre les hauteurs voisines, où une zone haute peut masquer un creux dans la surface.

Cartes de normales. Le rendu de cartes de normales ne nécessite pas forcément de connaître le voisinage du point éclairé. Tout se passe localement. C’est la représentation de détails surfaciques la plus économe qui existe. Contrairement aux cartes de déplacements, la géométrie n’est pas explicitement créée. Elle est simulée en perturbant la normale géométrique (figure 1.5, droite) et c’est au niveau du modèle de réflexion de la lumière (échelle microscopique) que la géométrie est simulée. Un point commun entre les cartes de déplacements (hauteurs) et les cartes de normales existe. Dans la plupart des cas, ces dernières sont construites à partir des premières. Le gradient d’un champ de hauteurs est une carte de pentes, qui peut être facilement transformée en carte de normales. Dans la figure 1.5, les normales ont été construites à partir de la carte de déplacements.

1.4 Rendu de micro-reflets

Lorsqu’il y a beaucoup de détails géométriques sous pixel (introduits par exemple par une surface augmentée), il est difficile de faire un rendu les retranscrivant en totalité et fidèlement. Il vaut mieux considérer la géométrie de manière statistique dans ce cas, en analysant par exemple la distribution de normales dans l’empreinte du pixel. Cette tâche n’est pas triviale car convertir un ensemble de triangles ou de normales sous pixel en une distribution de normales gérable par un modèle de réflexion est un problème difficile et pas encore totalement résolu. Deux catégories de méthodes s’opposent dans ce domaine : celles qui simplifient la micro-géométrie et celles qui ne la simplifient pas. La simplification est possible (et conseillée) dans le cas où le matériau est diffus. Même si sous pixel, la complexité géométrique est grande, la diffusion lumineuse dans toutes les directions de chaque détail surfacique va donner en moyenne une distribution de réflectance lisse et basse fréquence (figure 1.6, gauche). Cependant, si le matériau est parfaitement spéculaire, chaque détail surfacique va refléter la lumière dans une direction et la distribution de réflectance sera hérissée et haute fréquence (figure 1.6, droite). Les lobes (parfois des centaines) de la distribution devront être modélisés et retranscrits à la volée pour chaque empreinte de rayon. Dans ce cas, la simplification de la micro-géométrie et donc de la distribution de réflectance n’est pas correcte si le rendu se veut fidèle.

La simplification de détails très spéculaires sous pixel ne permet pas le rendu de *matériaux à micro-reflets*. Cette catégorie de matériaux est composée des surfaces pailletées (figure 1.6, droite) comme le papier brillant, la neige, le sable, ou encore certaines roches ou peintures métallisées. Les surfaces métalliques ou plastiques qui sont rayées, brossées ou bosselées rentrent aussi dans cette catégorie. Le rendu sans simplification de détails surfaciques spéculaires permet donc le rendu de matériaux à micro-reflets. Les détails peuvent être introduits avec une carte de déplacements ou une carte de normales.

Au début de cette thèse en 2015, seulement deux méthodes très récentes [JHY⁺14,

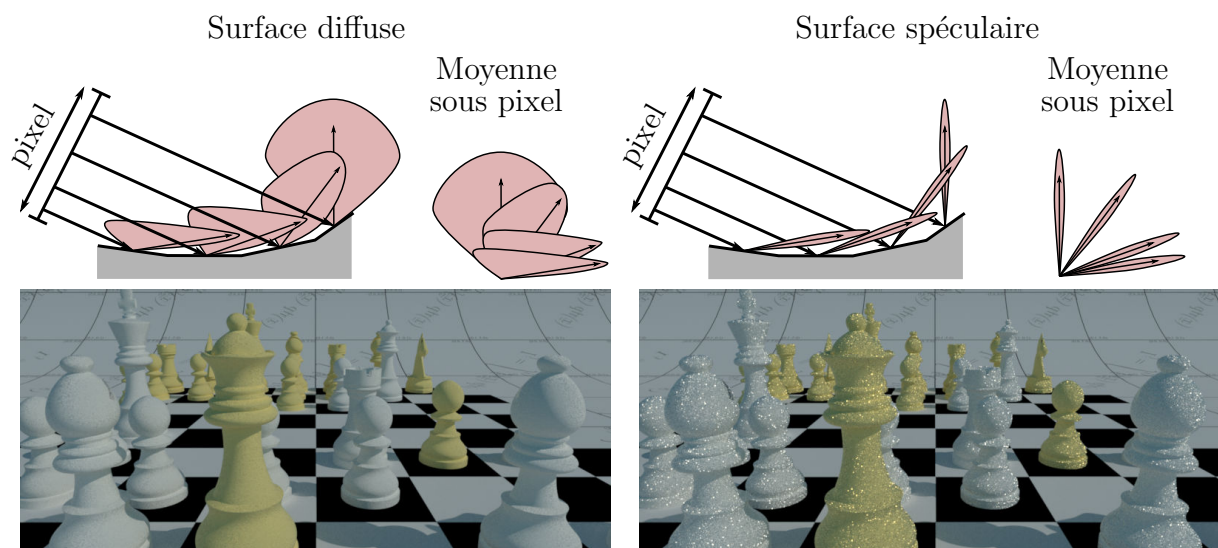


FIGURE 1.6 – Surfaces possédant beaucoup de détails surfaciques sous pixel. Lorsqu'ils sont diffus, la distribution moyenne de réflectance est lisse et le rendu est similaire à celui d'une surface plate et diffuse (gauche). Ce n'est pas le cas si les détails sont spéculaires. Dans ce cas, une infime zone sous pixel peut renvoyer beaucoup de lumière et la surface produit beaucoup de micro-reflets (droite).

[YHJ⁺14] permettaient le rendu de ce type de matériaux, alors qu'ils sont largement présents dans la vie de tous les jours. Le peu d'intérêt à l'époque pouvait s'expliquer par la haute complexité du problème. Lorsque ces méthodes sont sorties, elles n'étaient pas encore très au point et les performances étaient mauvaises. Mais rapidement, des extensions ont été publiées diminuant les temps de calcul par 100 environ [YHMR16, AK16]. Malgré tout, des problèmes n'étaient toujours pas résolus. Comment garantir que les modèles proposés étaient physiquement réalistes ? Une bonne manière de le vérifier est de reproduire la micro-géométrie, en vrai ou sur ordinateur et de mesurer ou calculer des références (par exemple en lançant plusieurs millions de rayons et en calculant numériquement la distribution). Comparer ces références avec les modèles permet de les valider ou non. Malheureusement, créer une référence réelle qui correspond à ce que modélise le modèle et mesurer sa réflectance est très compliqué : c'est rarement la méthode de validation qui est retenue. Les références obtenues virtuellement sont plus faciles à produire et peuvent être considérées comme physiquement réalistes. Il n'y a pas de perte ou de création d'énergie lumineuse et le multi-rebonds dans la surface est pris en compte (si l'algorithme simulant les réflexions lumineuses est correct). Là encore, peu de travaux utilisent cette méthode de validation, car elle est lourde à mettre en place et elle n'avantage pas souvent le modèle, qui est dans la plupart des cas moins respectueux des lois de la physique que la référence. C'est le cas des méthodes permettant le rendu de micro-reflets [YHMR16, AK16], où des principes simples comme la conservation de l'énergie ne sont pas respectés.

1.5 La conservation de l'énergie comme critère de qualité

Lors de la conception d'un modèle de réflexion de la lumière, il est important de respecter plusieurs propriétés physiques si le rendu se veut réaliste. L'une des plus importantes est la conservation de l'énergie. Si 100 watts d'énergie lumineuse arrivent sur un point, il faut qu'il y ait 100 watts qui repartent de ce point dans le cas où la surface est non absorbante. Si elle absorbe de l'énergie, alors la somme des énergies réfléchies doit être inférieure à 100 watts. La conservation de l'énergie à 100 % n'a pas été un critère de qualité depuis le début de l'informatique graphique. Les chercheurs utilisaient la contrainte plus faible qui autorisait la perte d'énergie, et donc les premiers modèles de réflexion analytiques ne simulaient qu'un rebond lumineux dans la micro-géométrie [CT81, WMLT07] (figure 1.7, gauche). Quelques méthodes récentes [HHdD16, RGB16] ont introduit durablement dans le domaine du rendu le concept de multi-rebonds pour un modèle de réflexion. Le but de ces méthodes était d'obtenir des représentations mathématiques plus en adéquation avec le comportement réel de la lumière. Pour valider ces modèles, le test du four blanc [Hei14] est utilisé. Il consiste à mettre un objet non absorbant dans une carte d'environnement blanche, distribuant la même quantité d'énergie uniformément dans toutes les directions. Le test est réussi si l'objet se confond avec son environnement : l'énergie est conservée (figure 1.7, droite). S'il est plus foncé, alors le modèle de réflexion fait disparaître de l'énergie lumineuse (figure 1.7, gauche) et s'il est plus clair, il en crée. Dans les deux derniers cas, ce n'est pas physiquement réaliste et l'apparence des objets est soit trop sombre (figure 1.7, gauche), soit trop claire, alors qu'elle ne doit pas l'être (figure 1.7, droite).

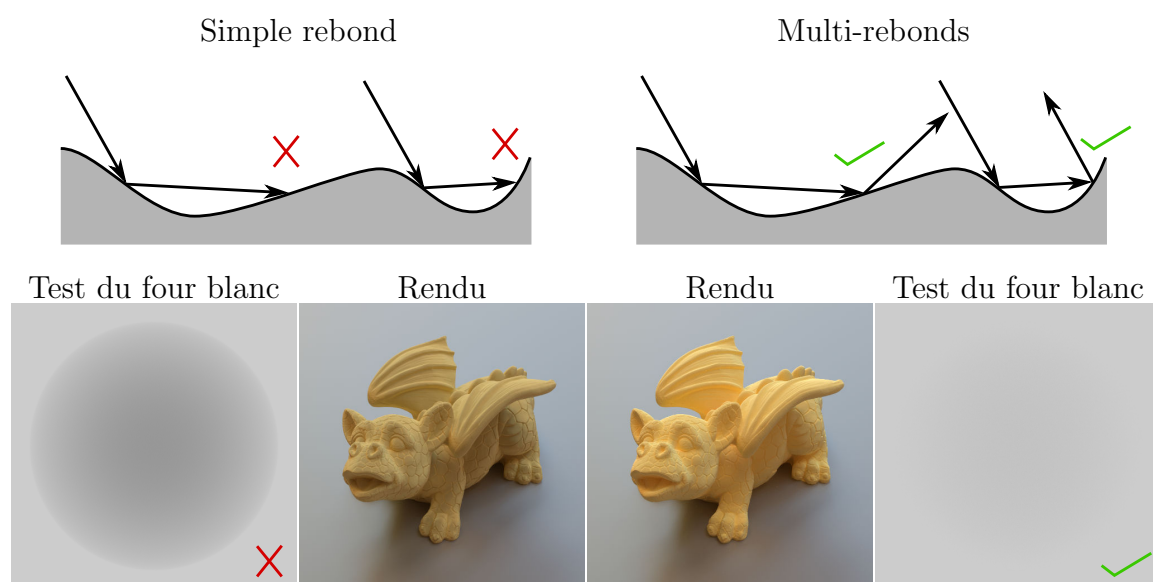


FIGURE 1.7 – La conservation de l'énergie joue un rôle important en rendu basé physique. Elle permet de valider un modèle de réflexion de la lumière. À gauche, le modèle de Cook et Torrance [CT81] modélisant qu'un rebond lumineux dans la micro-surface : il ne passe pas le test du four blanc et le rendu a des couleurs insaturées (grises). À droite, le modèle multi-rebonds de Heitz et al. [HHdD16]. Il passe le test du four blanc et les couleurs sont saturées et naturelles.

1.6 Objectif de la thèse

L'objectif de la thèse a été de développer des modèles de réflexion de la lumière permettant le rendu de matériaux à micro-reflets qui conservent l'énergie (figure 1.8, gauche et milieu). Nous avons choisi la carte de normales comme primitive de la micro-surface, car elle permet de modéliser une grande variété de matériaux : micro-rayures, surfaces pailletées, surfaces rugueuses, etc. Elle est moins coûteuse en mémoire qu'une carte de déplacements et suffisante pour notre objectif de rendu. L'état des connaissances sur le domaine nous a permis de remarquer que les modèles existants [YHJ⁺14, YHMR16] ne se focalisaient pas sur la conservation de l'énergie. Des créations et des fuites d'énergies étaient possibles dans certaines situations, par exemple à haute rugosité (figure 1.8, encart à droite). Ils n'étaient donc pas complètement basés physique.

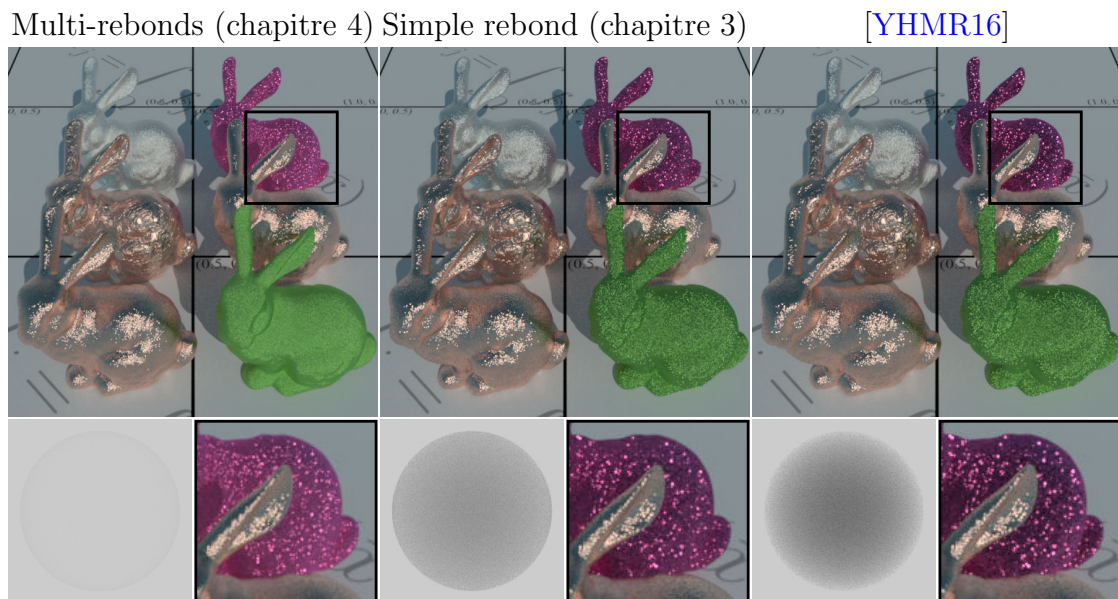


FIGURE 1.8 – Matériaux à micro-reflets obtenus avec la méthode précédente [YHMR16] (droite) et nos deux méthodes développées durant la thèse (gauche et milieu). Droite : La conservation de l'énergie n'est pas prise en compte. Milieu : Notre méthode du chapitre 3 qui n'autorise que la perte d'énergie. Droite : Notre méthode du chapitre 4 qui modélise le multi-rebonds dans la micro-surface et qui par conséquent, est plus réaliste.

1.7 Contributions

Nous avons d'abord conçu un modèle qui ne créait pas d'énergie mais qui en autorisait la perte (figure 1.8, milieu). Ce modèle simple rebond est présenté dans le chapitre 3. Les contributions de ce travail sont :

- La dérivation exacte du terme de normalisation (terme de masquage) d'une distribution de normales construite à partir d'une empreinte de rayon sur une carte de normales. Ce terme garantit la conservation de l'énergie.
- Une approximation de ce terme permettant le rendu efficace et précis d'une carte de normales spéculaire.
- La comparaison avec un rendu brute force de référence.

Ce travail a donné lieu à une publication dans la revue internationale *The Visual Computer* [CCM18] et a été présenté à l’AFIG (Association Française d’Informatique Graphique) en 2018. Même si les résultats étaient meilleurs que les travaux précédents, nous avons quand même remarqué que le manque d’intégration du multi-rebonds dans la carte de normales ne permettait pas un rendu réaliste, surtout à haute rugosité.

Ce premier travail a été une première étape vers l’objectif final que nous nous étions fixés : la conservation de l’énergie à 100 %. Cet objectif a été (presque) atteint (figure 1.8, gauche) grâce à une nouvelle façon d’interpréter géométriquement la carte de normales. Au lieu de considérer en chaque point de la surface *une* normale, nous en considérons *deux* : la primale et sa duale, qui dans notre cas est la symétrique par rapport à la normale géométrique. Nous avons trouvé l’idée dans les travaux de Schüssler et al. [SHHD17] qui eux utilisent une normale duale tangente pour faire le rendu basé physique d’une carte de normales. Dans leur modèle et le nôtre, l’équation d’illumination surfacique d’une carte de normales est simple et ne nécessite pas les normalisations complexes que nous avons trouvées dans notre premier travail (chapitre 3). Cependant, l’utilisation des normales tangentes pour le rendu de micro-reflets introduit des artefacts, contrairement aux normales symétriques. Pour modéliser le multi-rebonds dans la micro-surface, nous quantifions analytiquement la quantité d’énergie perdue à cause d’une modélisation simple rebond. Cette quantité est ensuite réinjectée à l’aide d’une évaluation pondérée d’un modèle de réflexion simulant les rebonds d’ordre deux et supérieur. Ce travail a été publié dans la revue internationale *Computer Graphics Forum* [CCM19] et a été présenté à EGSR (Eurographics Symposium on Rendering) en 2019. Il constitue le chapitre 4 et apporte les contributions suivantes :

- Il propose un modèle de réflexion de la lumière multi-rebonds permettant le rendu de matériaux à micro-reflets.
- Ce modèle a une procédure d’échantillonnage qui est optimale. Aucun échantillon de direction lumineuse sortante n’est invalide ou transporte une énergie nulle.
- Il permet un rendu réaliste et sans artefact.

Avant de décrire les deux méthodes que nous avons développées durant cette thèse (chapitres 3 et 4), nous présentons un état des connaissances complet et les notions théoriques nécessaires à leur compréhension (chapitre 2). Après la conclusion de la thèse (chapitre 5), nous proposons une annexe permettant d’installer nos deux méthodes dans le moteur de rendu `pbrt-v3` (annexe A).

Chapitre 2 :

État des connaissances dans le rendu de micro-reflets

Le rendu de micro-reflets et les notions relatives à ce domaine ont été introduits de manière intuitive dans le chapitre précédent. Dans le présent chapitre, nous introduisons le socle théorique nécessaire et présentons l'état des connaissances pour comprendre les deux méthodes développées durant cette thèse. Les notations, symboles et équations mathématiques liés aux modèles de réflexions sont introduits ici. Dans la section 2.1, nous rappelons les généralités à connaître sur les fonctions de *distribution de réflectance bidirectionnelles* (Bidirectional Reflectance Distribution Functions), abrégées BRDFs. Dans les sections 2.2, 2.3 et 2.4, nous présentons les modèles de BRDF couramment utilisés en informatique graphique. Ils sont généralement définis pour un point sur une surface plane, centrés sur la normale géométrique, et reliés à la *théorie des micro-facettes*. Pour ajouter des détails surfaciques à moindre coût, les surfaces peuvent être augmentées avec des cartes de normales. En chaque point de la surface, la BRDF est calculée non plus dans le repère défini par la normale géométrique de l'objet, mais avec une normale perturbée par rapport à celle-ci. Dans la section 2.5, nous passons en revue les modèles de réflexion qui tiennent compte de cette perturbation. Le rendu de micro-reflets est lié à ces modèles, car les micro-miroirs ont des normales perturbées par rapport à la normale géométrique, mais ce type de matériau nécessite une évaluation surfacique de la BRDF (liée à l'empreinte du rayon), et non plus une évaluation en un point donné d'une surface (section 2.6). Pour finir ce chapitre, nous présentons de manière formelle l'objectif de la thèse (section 2.7), c'est-à-dire quelles équations nous essayons de résoudre efficacement et quelles contraintes nous essayons de respecter. Ces contraintes ont été présentées de manière informelle dans le chapitre précédent.

2.1 Rappels généraux sur les BRDFs

Le terme BRDF est l'abréviation de Bidirectional Reflectance Distribution Function que nous pouvons traduire en français par *fonction de distribution de réflectance bidirectionnelle*. Les modèles de réflexion de la lumière présentés intuitivement avec des lobes rouges dans le chapitre précédent sont des BRDFs. Ces fonctions ont deux paramètres, une direction d'observation et une direction incidente. Pour pouvoir les définir, il faut dans un premier temps donner la configuration géométrique et la paramétrisation de l'espace. Les BRDFs donnent un ratio d'énergie, nous donnerons donc la définition des différentes quantités physiques qui sont impliquées lors d'un rendu basé physique ainsi que les propriétés énergétiques à respecter. La distribution de réflexion (BRDF) est une sous-partie de l'équation du rendu, que les moteurs 3D tentent d'évaluer le plus efficacement et le plus précisément possible. La BRDF doit être évaluable, et si le moteur de rendu est basé sur l'intégrateur de Monte Carlo, alors la BRDF doit pouvoir être échantillonnée et avoir une fonction de densité (Probability Density Function, abrégé PDF) analytique associée à la procédure d'échantillonnage. Sinon, le temps de synthèse de l'image sera beaucoup trop long.

2.1.1 Configuration géométrique

Système de coordonnées tangent. Pour définir des directions, des points et des vecteurs, il faut d'abord définir un système de coordonnées. Lors de l'intersection d'un rayon avec un objet, nous considérons que localement la surface est plane et nous

souhaitons connaître les échanges radiatifs qui ont lieu entre cette surface et les directions d'observation et d'incidence. Les équations modélisant ces échanges sont exprimées dans un repère orthonormé et cartésien (x, y, z) . En considérant que la surface est plane localement, les directions tangentes peuvent être définies autour de la normale géométrique de l'objet $\omega_g = (0, 0, 1)$. La sécante $s = (1, 0, 0)$ et la tangente $t = (0, 1, 0)$ sont orthogonales et appartiennent au plan de réflexion. Dans cette paramétrisation de l'espace 3D, les directions ω peuvent être exprimées en coordonnées sphériques (θ, ϕ) , où le vecteur unitaire et cartésien correspondant est $(\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta)$. La configuration géométrique est schématisée dans la figure 2.1.

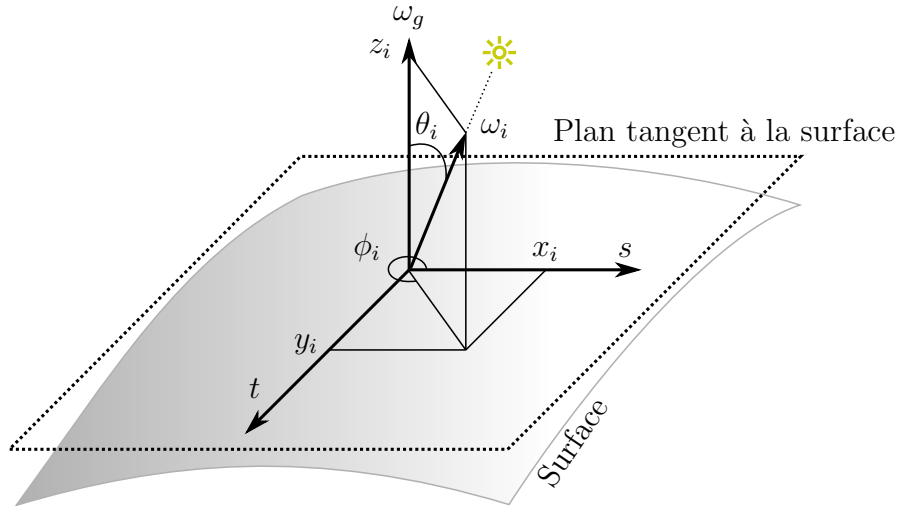


FIGURE 2.1 – Illustration du plan tangent et de son système de coordonnées associé. Ici, le vecteur incident ω_i est exprimé avec des coordonnées cartésiennes (x_i, y_i, z_i) et sphériques (θ_i, ϕ_i) . Les équations de cette thèse sont exprimées dans ce repère orthonormé.

Conventions pour les directions. Une pratique courante pour les notations de direction dans le domaine de la BRDF est la suivante [DHI+13, JHY+14, PJH16] :

- Le symbole ω (avec ou sans indice) représente un vecteur de longueur 1.
- La normale de la surface géométrique est notée ω_g , où « g » signifie « géométrique ».
- La direction d'observation est notée ω_o , où « o » signifie « observation ».
- La direction d'incidence ou d'illumination est notée ω_i , où « i » signifie « incidence ».
- Les indices, par exemple o ou i , sont aussi utilisés pour lier des coordonnées ou des quantités à une direction. Pour désigner les composantes cartésiennes de ω_i , nous notons (x_i, y_i, z_i) et pour ses composantes sphériques, nous notons (θ_i, ϕ_i) (figure 2.1).

Nous utilisons ces conventions de notation pour tous les chapitres de la thèse. À noter que d'autres conventions existent, utilisant d'autres notations [WMLT07] ou inversant ω_o et ω_i [HHd16].

2.1.2 Quantités énergétiques

L'étude de la diffusion de la lumière fait appel à des quantités physiques du domaine de la radiométrie : l'énergie, le flux radiant, l'irradiance et la radiance (figure 2.2). Elles

sont toutes liées entre elles et sont nécessaires pour définir formellement une BRDF. Plus généralement, elles interviennent dans l'équation du rendu, qui dépend de la BRDF.

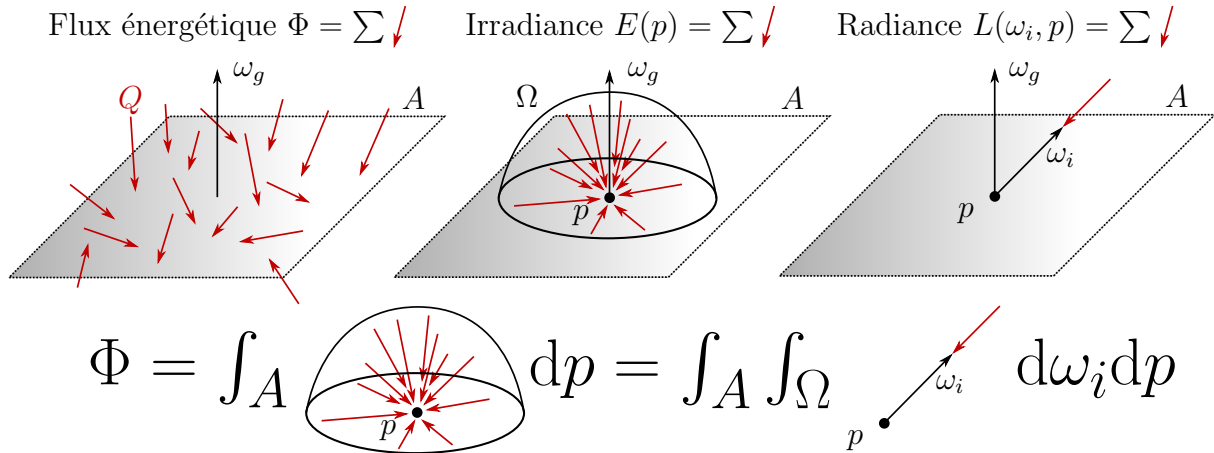


FIGURE 2.2 – Illustration des quantités radiométriques et des relations entre elles.

Énergie. C'est le point de départ pour faire du rendu. Le soleil et les lumières sont des sources d'énergies qui émettent des photons. Chaque photon transporte de l'énergie Q , qui est dépendante de sa longueur d'onde. Elle est mesurée en joules (J).

Flux énergétique. En anglais *Radiant Flux*, noté Φ , c'est la quantité d'énergie totale passant à travers une surface par unité de temps :

$$\Phi = \frac{\partial Q}{\partial t}. \quad (2.1.1)$$

Son unité est la joule/seconde (J/s), ou plus communément, le watt (W). Une source lumineuse est souvent caractérisée par cette valeur. Par exemple, si une lumière émet 100 J pendant 100 secondes, alors son flux énergétique est $\Phi = 1\text{W}$.

L'irradiance, ou l'éclairement énergétique, noté E , est la densité moyenne de flux radiant incident par unité de surface ∂A :

$$E(p) = \frac{\partial \Phi(p)}{\partial A}. \quad (2.1.2)$$

Son unité est le W/m² et cette quantité est définie pour un point p . Nous pouvons retrouver le flux radiant grâce à l'irradiance en l'intégrant sur une aire A :

$$\Phi = \int_A E(p) dp. \quad (2.1.3)$$

Si une lumière ponctuelle possédant un flux énergétique Φ est au centre d'une sphère de rayon r , alors chaque point à l'intérieur de la sphère reçoit un éclairement énergétique égal à

$$E = \frac{\Phi}{4\pi r^2}. \quad (2.1.4)$$

Cette dernière équation permet d'expliquer pourquoi la quantité d'énergie arrivant sur un point depuis une lumière décroît avec le carré de la distance entre ce point et la source.

La radiance, ou luminance énergétique, notée L , est la quantité la plus importante car c'est l'unité dans laquelle nous exprimons les valeurs des pixels d'une image. C'est la valeur que nous tentons de calculer en lançant des rayons depuis la caméra. Contrairement à l'irradiance, la radiance prend en compte la distribution directionnelle de l'énergie. Elle mesure l'irradiance par angle solide infinitésimal (un cône extrêmement petit, c'est-à-dire une direction ω_i) :

$$L(\omega_i, p) = \frac{\partial E(p)}{\cos \theta_i \partial \omega_i} = \frac{\partial^2 \Phi(p)}{\cos \theta_i \partial \omega_i \partial A}. \quad (2.1.5)$$

Son unité est le Watt par stéradian par mètre carré ($W/(m^2sr)$). Sa définition inclut un terme $\cos \theta$ issu de la loi de Lambert, qui prend en compte le fait que la surface illuminée augmente proportionnellement avec l'angle d'incidence θ . L'irradiance est obtenue en intégrant toutes les directions incidentes Ω :

$$E(p) = \int_{\Omega} L(\omega_i, p) \cos \theta_i d\omega_i. \quad (2.1.6)$$

Nous pouvons retrouver le flux radiant en intégrant l'irradiance sur l'aire A :

$$\Phi = \int_A \int_{\Omega} L(\omega_i, p) \cos \theta_i d\omega_i dp = \int_A E(p) dp. \quad (2.1.7)$$

À partir de la radiance, nous pouvons connaître toutes les autres quantités radiométriques en intégrant sur des aires et des directions (figure 2.2). De plus, la radiance est constante le long d'un rayon qui se propage dans le vide. C'est ces raisons qui font que cette quantité est la plus utilisée en lancer de rayons.

2.1.3 Équation d'illumination locale

L'objectif final en lancer de rayons est de calculer la radiance observée $L(\omega_o)$ en un point du capteur de la caméra depuis un rayon pointant vers $-\omega_o$ (figure 2.3). C'est l'équation d'illumination locale, aussi appelée l'équation du rendu, formalisée par Kajiya [Kaj86], qui permet ce calcul. Elle implique une intégration de toutes les radiances incidentes $L(\omega_i)$ au point d'intersection du rayon, pondérées par la BRDF f du matériau¹ (La dépendance à p n'apparaît pas mais elle est implicite) :

$$L(\omega_o) = \int_{\Omega} L(\omega_i) f(\omega_o, \omega_i) \cos \theta_i d\omega_i. \quad (2.1.8)$$

Comme évoqué dans la section précédente, la radiance L est la quantité énergétique la plus importante en lancer de rayons car cette valeur est constante le long d'un rayon se déplaçant dans le vide. L'équation du rendu permet de connaître l'équilibre entre une

1. Si le point p émet de la lumière, de la radiance est ajoutée au membre de droite de l'équation 2.1.8. Si le point p n'est pas opaque, l'intégrale a lieu sur la sphère des directions incidentes et non pas juste sur l'hémisphère supérieure. Ici, l'équation 2.1.8 est en fait l'équation de réflexion locale.

radiance sortante $L(\omega_o)$ et des radiances incidentes $L(\omega_i)$ en un point p . La BRDF f va permettre de connaître le pourcentage d'énergie reflétée.

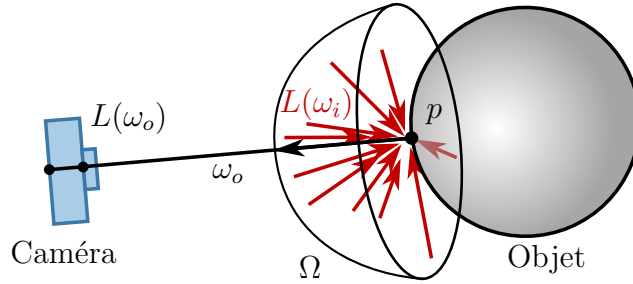


FIGURE 2.3 – L'équation d'illumination locale permet de calculer l'intensité lumineuse arrivant sur un capteur d'une caméra virtuelle. C'est l'intégrale de toutes les radiances incidentes pondérées par la BRDF au point d'intersection du rayon.

2.1.4 Définition et propriétés des BRDFs

Les réflexions sur les surfaces sont modélisées par des BRDFs (Bidirectional Reflectance Distribution Functions). Ce sont des fonctions de distribution bidirectionnelles 4D qui prennent deux directions en paramètre : une direction d'observation ω_o et une direction incidente ω_i . Elles définissent comment la lumière est reflétée *en un point* sur une surface opaque. Cette fonction de distribution est nécessaire pour évaluer l'équation du rendu (équation 2.1.8) et ainsi connaître la quantité d'énergie arrivant sur la caméra. Elle est souvent visualisée en 3D en fixant la direction incidente ω_i (figure 2.4).

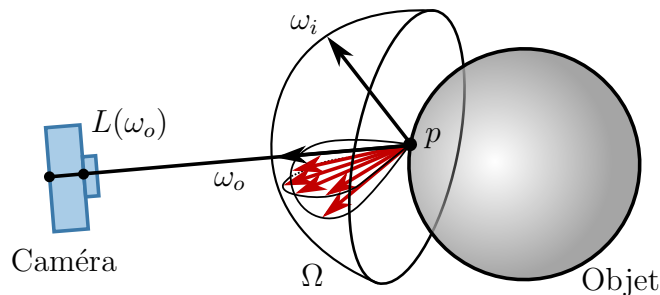


FIGURE 2.4 – La BRDF est une distribution qui peut être visualisée en fixant un angle incident ω_i . Ici, elle est représentée par le lobe formé par les directions de réflexion (flèches rouges). Elle permet de connaître la proportion d'énergie réfléchiée dans la direction ω_o .

Définition formelle. La BRDF permet de savoir combien de radiance $L(\omega_o)$ quitte la surface dans la direction d'observation ω_o en fonction de la radiance incidente $L(\omega_i)$. Sa définition est basée sur la dérivée partielle de l'irradiance $\partial E(\omega_i)$, elle-même basée sur un cône de directions infinitésimal $\partial\omega_i$:

$$\partial E(\omega_i) = L(\omega_i) \cos \theta_i \partial\omega_i. \quad (2.1.9)$$

Une quantité infinitésimale de radiance $\partial L(\omega_o)$ est reflétée dans la direction ω_o de manière proportionnelle à cette irradiance différentielle $\partial E(\omega_i)$. La BRDF donne ce ratio entre la

radiance différentielle sortante le long de ω_o et l'irradiance différentielle incidente depuis la direction ω_i :

$$f(\omega_o, \omega_i) = \frac{\partial L(\omega_o)}{\partial E(\omega_i)} = \frac{\partial L(\omega_o)}{L(\omega_i) \cos \theta_i \partial \omega_i}. \quad (2.1.10)$$

Cette définition formelle introduite pour la première fois par Fred Nicodemus [Nic65], utilisée dans l'équation du rendu (équation 2.1.8), permet de

- caractériser une surface opaque,
- définir son apparence,
- calculer la radiance sortante à partir des radiances incidentes en un point
- et donc de calculer l'intensité d'un pixel d'une caméra virtuelle.

Une BRDF est dite *basée physique* si elle respecte les trois propriétés fondamentales suivantes :

Positivité. Une BRDF étant une distribution, toutes ses valeurs sont positives ou nulles. Si des valeurs de BRDF sont négatives, alors de l'énergie négative est créée, ce qui est impossible dans la réalité.

Réciprocité. Le ratio d'énergie lumineuse est identique dans un sens comme dans l'autre : $f(\omega_o, \omega_i) = f(\omega_i, \omega_o)$. Si la BRDF n'est pas réciproque (formulation asymétrique), alors un rendu obtenu en lançant des rayons de la caméra vers les lumières ne sera pas identique à un rendu obtenu avec des rayons allant des lumières vers la caméra. C'est un problème lorsque des algorithmes bidirectionnels sont utilisés en animation, où la continuité entre les images risque de ne plus être maintenue. Cependant, lors de l'utilisation d'un lancer de rayons unidirectionnel avec des BRDFs non-réciproques, il n'y a pas d'artéfacts visibles. Cette propriété n'est pas respectée par toutes les BRDFs en informatique graphique, et nos modèles présentés dans le chapitre 3 et 4 ne font pas exception.

Conservation de l'énergie. Il est facile de s'imaginer un rayon lumineux venant de ω_o , avec $L(\omega_o) = 1$, touchant un point sur une surface non absorbante et se réfléchissant dans plusieurs directions différentes (comme dans la figure 2.4 en inversant ω_o et ω_i). En fixant ω_o , les directions de réflexion forment une distribution 3D égale à $f(\omega_o, \omega_i) \cos \theta_i$, dont le volume doit être égal à un :

$$\int_{\Omega} f(\omega_o, \omega_i) \cos \theta_i d\omega_i = 1. \quad (2.1.11)$$

Cette équation souligne le fait que toute l'énergie est reflétée, pas plus, pas moins. Elle suppose que les rayons lumineux rebondissent dans la micro-surface, et arriveront toujours à sortir. Si cette propriété n'est pas respectée, alors l'apparence des objets peut être anormalement trop sombre (perte d'énergie) ou trop claire (création d'énergie). Modéliser le multi-rebonds dans la micro-géométrie est compliqué, et beaucoup de modèles de BRDF ne modélisent qu'un rebond :

$$\int_{\Omega} f(\omega_o, \omega_i) \cos \theta_i d\omega_i \leq 1. \quad (2.1.12)$$

La conservation de l'énergie, la réciprocité et la positivité permettent de vérifier la qualité d'un modèle de réflexion analytique. C'est un idéal vers lequel tendre et en pratique, très peu de BRDFs respectent ces trois règles. Seuls les modèles idéaux (Lambert, réflexion parfaite), et les modèles récents pour des matériaux rugueux [HHdD16, LJJ+18, XH18] ou rayés [RGB16] sont *réellement* basés physique. Par exemple, le modèle de Phong original [Pho75] n'était pas réciproque ni normalisé (c'est-à-dire qu'il ne conservait pas l'énergie). Le modèle de Cook et Torrance [CT81] et ses extensions [WMLT07] ne conservent pas totalement l'énergie, ainsi que la majorité des modèles à micro-reflets [JHY+14, YHJ+14, YHMR16]. Lorsque la micro-surface est rugueuse, les modèles simple rebond ont une apparence globale qui est trop sombre (figure 2.5), car les rayons intersectant deux fois ou plus la micro-géométrie avant d'en sortir sont nombreux.

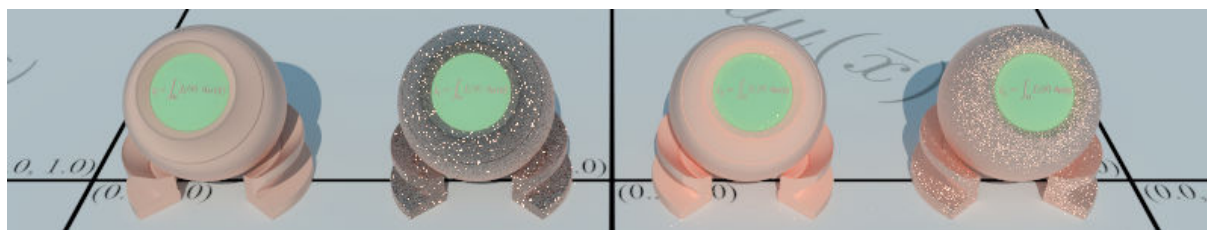


FIGURE 2.5 – Gauche : Modèles simple rebond, avec une apparence sombre [WMLT07, YHMR16]. Droite : Modèles multi-rebonds [HHdD16, CCM19], conservant l'énergie.

2.1.5 BRDFs dans un moteur de rendu basé physique

Incorporer une BRDF dans un moteur de rendu basé physique nécessite l'implémentation de trois fonctions (figure 2.6). Ces fonctions sont souvent utilisées une ou plusieurs fois pour chaque rayon lancé depuis la caméra, lors d'une intersection avec une surface. Elles doivent donc être rapides à évaluer. Par exemple, pour générer une image HD possédant 1920×1080 pixels, avec 16 échantillons par pixel, il faut lancer plus de 33 millions de rayons : ces fonctions sont donc utilisées énormément de fois lors d'un rendu. De plus, ces calculs se font pour une longueur d'onde donnée de la lumière.

Évaluation. Cette fonction retourne la densité d'énergie réfléchie $f(\omega_o, \omega_i)$ depuis une direction incidente ω_i vers une direction d'observation ω_o (figure 2.6, gauche). Pour des modèles de réflexion classique [WMLT07], cette fonction consiste en une évaluation d'une formule mathématique relativement simple. L'évaluation peut être plus complexe, par exemple pour modéliser du multi-rebonds avec des marches aléatoires [HHdD16, SHHD17], où la valeur de la BRDF est l'espérance des densités de toutes les marches possibles. Dans le cas d'un modèle de réflexion à micro-reflets [YHMR16] (chapitre 3 et 4), l'évaluation est une somme pondérée de plusieurs BRDFs locales.

Échantillonnage. L'équation de réflexion locale (équation 2.1.8) n'a pas de solution analytique (à part pour des configurations extrêmement simples). Des méthodes de résolution numérique sont donc nécessaires et la plus utilisée est celle de Monte-Carlo, car son taux de convergence est indépendant de la dimension de l'intégrale. De plus,

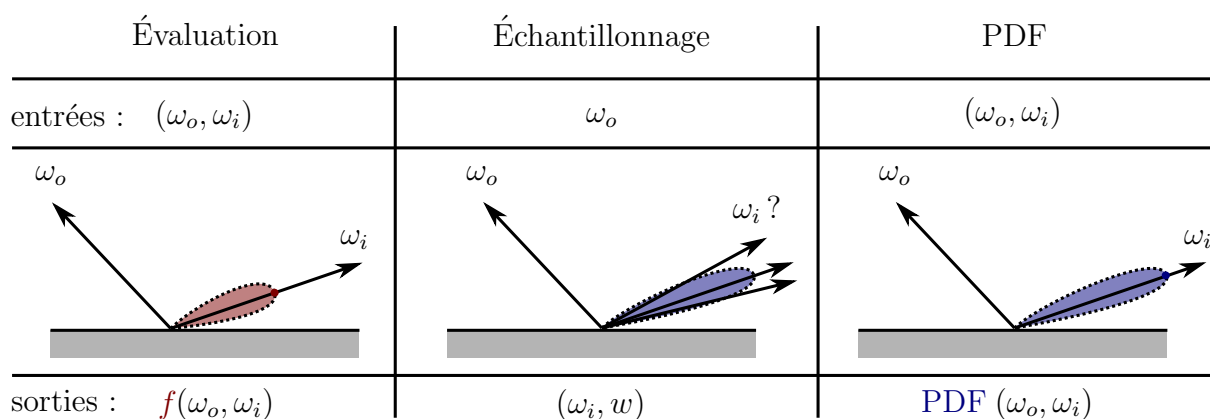


FIGURE 2.6 – Implémenter une BRDF dans un moteur de rendu basée physique nécessite l'implémentation de trois fonctions : l'évaluation (gauche), l'échantillonnage (milieu) et la PDF (droite).

l'intégration de Monte Carlo permet l'utilisation de l'*échantillonnage préférentiel* pour réduire grandement la variance de l'estimateur. Fournir une procédure d'échantillonnage optimale de la BRDF est donc important dans le but de réduire les temps de rendu. Concrètement, la fonction d'échantillonnage renvoie une direction de sortie ω_i en fonction d'une direction d'observation ω_o . Pour cela, elle s'appuie sur la forme de la distribution de réflexion (figure 2.6, milieu). Malheureusement, lorsque la BRDF a une forme trop complexe, il est difficile de trouver un algorithme l'échantillonnant parfaitement. Échantillonner une distribution implique le calcul de l'inverse de sa fonction de distribution cumulative, nommée CDF (Cumulative Distribution Function en anglais). Parfois, il n'y a pas de solution analytique à ce calcul et nous pouvons soit stocker cette fonction en mémoire, soit échantillonner une sous-partie de la BRDF qui est plus simple (c'est ce que nous faisons dans notre méthode décrite dans le chapitre 3). Dans le premier cas, l'impact mémoire est important, dans le deuxième cas, l'échantillonnage est sous-optimal, ce qui augmente la variance de l'estimateur et cause des artéfacts, comme des pixels brillants (fireflies). Dans le chapitre 4, nous proposons un modèle de BRDF avec une procédure d'échantillonnage optimale qui élimine ces problèmes.

Pour finir, la fonction doit aussi retourner le poids de l'échantillon w , qui est égal à la valeur de la BRDF pondérée par le cosinus de l'angle polaire incident, divisée par la PDF de la procédure d'échantillonnage préférentiel :

$$w(\omega_o, \omega_i) = \frac{f(\omega_o, \omega_i) \cos \theta_i}{\text{PDF}(\omega_o, \omega_i)}. \quad (2.1.13)$$

Ce poids est utilisé par l'estimateur de Monte Carlo pour prendre en compte le fait que certaines directions seront plus échantillonnées que d'autres.

Fonction de densité de probabilité (PDF). En anglais : Probability Density Function. Lors de l'estimation de l'intégrale de l'équation de réflexion locale (équation 2.1.8) avec Monte Carlo, l'échantillonnage préférentiel n'est souvent possible que pour des sous-parties de l'intégrande. Typiquement, la BRDF peut être échantillonnée « facilement » et la lumière aussi, mais pas les deux en même temps. Échantillonner les deux séparément puis combiner chaque contribution s'appelle l'*échantillonnage*

préférentiel multiple [Vea97] (MIS, pour Multiple Importance Sampling en anglais). Lors d'un rendu, le moteur échantillonne la BRDF *et* la lumière, et utilise la PDF de chaque échantillon pour calculer un poids final. C'est pourquoi il est nécessaire de coder une fonction qui retourne la PDF de la procédure d'échantillonnage évaluée pour un couple de directions (ω_o, ω_i) . Il est préférable que la PDF ait la même forme que celle utilisée pour échantillonner la BRDF (figure 2.6). Si ce n'est pas le cas, l'estimateur converge quand même vers la solution exacte, même si sa variance augmente et que des pixels blancs peuvent apparaître (troisième orbe en partant de la gauche dans la figure 2.5). Ils apparaissent lorsque la BRDF a une grande valeur (numérateur de l'équation 2.1.13) et que la PDF a une petite valeur (située au dénominateur).

En plus de respecter les principes physiques de la réciprocité, de la conservation *totale* de l'énergie et de la positivité (section 2.1.4), il faut trouver une formulation de BRDF rapide à évaluer, qui s'échantillonne de manière optimale et qui possède une PDF analytique. Là encore, c'est un idéal vers lequel tendre, et peu de modèles ont toutes ces propriétés en même temps. Dans les sections suivantes, nous allons passer en revue les différents modèles de BRDFs qui existent actuellement et voir leurs points forts et leurs points faibles.

2.2 Les modèles de BRDFs idéaux

Une BRDF donne la densité de probabilité d'un photon incident d'être réfléchi dans une direction d'observation. La surface est considérée comme suffisamment opaque dans le but d'ignorer la transmission dans le matériau et ainsi éviter un décalage spatial notable entre le point d'incidence et de réflexion. Autrement dit, la réflectivité (BRDF) est caractérisée pour un point p sur la surface. Cette définition est opposable à celle de la \mathcal{P} -BRDF, où la BRDF n'est pas définie pour un point p mais un patch \mathcal{P} , c'est-à-dire une surface, qui dans notre cas sera l'empreinte du rayon associée à un pixel. En réalité, que ce soit pour une BRDF ou une \mathcal{P} -BRDF, il y a toujours une notion d'aire de surface associée à la définition de ces modèles. Mais dans le premier cas, l'aire de la surface est considérée comme infinitésimale. Dans les quatre prochaines sections, nous passons en revue les BRDFs définies pour un point p sur une surface. Nous pouvons dire qu'elles sont invariantes relativement à la surface considérée.

Modèles idéaux. Les modèles idéaux de BRDF sont la réflexion et la diffusion parfaite (figure 2.7). Ils sont idéaux car ils respectent *toutes* les propriétés énoncées précédemment (section 2.1.4 et 2.1.5). La conservation de l'énergie est totale si le matériau est non-absorbant, la réciprocité est respectée, l'évaluation est rapide, l'échantillonnage est parfait et la PDF est analytique.

Réflexion parfaite. Naturellement, nous avons tendance à dire qu'un rayon lumineux est reflété dans une seule et unique direction : la direction de réflexion ω_r (figure 2.7, gauche). C'est ce que modélise la BRDF de réflexion spéculaire parfaite. Sa définition implique une distribution de Dirac qui ne laisse passer l'énergie que dans la direction de réflexion ω_r :

$$f(\omega_o, \omega_i) = \frac{\delta(\omega_i - \omega_r)}{\cos \theta_r}. \quad (2.2.1)$$



FIGURE 2.7 – Les modèles de BRDFs idéaux sont la réflexion parfaite (gauche) et la diffusion parfaite (droite).

La direction de réflexion se calcule à l'aide de la direction d'observation ω_o et de la normale géométrique ω_g :

$$\omega_r = -\omega_o + 2(\omega_g \cdot \omega_o)\omega_g. \quad (2.2.2)$$

Ce modèle suppose que la micro-surface est parfaitement plate, comme un miroir parfait. Il sert aussi à construire des modèles plus complexes, supposant par exemple un ensemble de facettes parfaitement spéculaires [CT81].

Diffusion parfaite. La diffusion parfaite modélise des matériaux diffus ayant une micro-surface extrêmement rugueuse, comme du béton, du papier ou une peinture matte. Ce modèle diffuse uniformément la lumière dans toutes les directions (figure 2.7, droite). C'est un modèle théorique qui n'a pas vraiment de plausibilité physique, mais c'est une approximation tout à fait raisonnable pour beaucoup de matériaux réels. Sa définition est

$$f(\omega_o, \omega_i) = \frac{1}{\pi}. \quad (2.2.3)$$

La division par π permet de normaliser correctement la BRDF et est obtenue en résolvant l'intégrale suivante :

$$\int_{\Omega} 1 \cos \theta_i d\omega_i = \pi. \quad (2.2.4)$$

2.3 Modéliser l'absorption des matériaux

L'absorption de l'énergie par la surface est modélisée par les *équations de Fresnel*. Ces équations utilisent les *indices de réfraction* des matériaux, qui dans le cas des *diélectriques* est une valeur réelle, et dans le cas des *conducteurs* est une valeur complexe. Les diélectriques ne conduisent pas l'électricité et transmettent une portion de la radiance incidente dans la surface. Dans cette catégorie, nous avons par exemple le verre, l'huile, l'eau et l'air. Ces matériaux sont caractérisés par leur indice de réfraction et sont modélisés par une *BSDF* (Bidirectional *Scattering* Distribution Function). Dans cette thèse, nous nous intéressons seulement à la réflexion, et donc nous ne détaillons pas plus précisément le sujet des BSDFs.

Les matériaux conducteurs, comme les métaux, reflètent une grosse partie de l'énergie

incidente. Le terme de Fresnel donne la proportion d'énergie qui n'est pas absorbée. Ces matériaux sont caractérisés par un coefficient de réfraction η et un coefficient d'absorption k . Ces coefficients ont été mesurés dans la vie réelle et peuvent être directement utilisés dans un moteur de rendu, pour simuler l'apparence de l'or par exemple. En plus du coefficient de réfraction et d'absorption, le terme de Fresnel dépend aussi de l'angle d'observation θ_o . L'impact lors du rendu de l'absorption de l'énergie par une surface est simple : les matériaux se « colorisent ». Les équations de Fresnel étant complexes, l'approximation de Schlick [Sch94] est couramment utilisée en informatique graphique car elle est rapide à évaluer et relativement précise en RGB (Red Green Blue) [Hof19].

Pour modéliser encore plus simplement l'absorption, il est courant de spécifier un albedo ρ qui indique la proportion d'énergie reflétée pour chaque longueur d'onde de la lumière. Les deux modèles idéaux présentés précédemment sont maintenant reformulés pour prendre en compte l'absorption. Pour la réflexion parfaite, le terme de Fresnel $F(\theta_o)$ est souvent utilisé car ce type de réflexion a lieu principalement sur des métaux (figure 2.8, gauche) :

$$f(\omega_o, \omega_i) = F(\theta_o) \frac{\delta(\omega_i - \omega_r)}{\cos \theta_r}. \quad (2.3.1)$$

Pour la diffusion parfaite (modèle de Lambert), un albedo ρ est utilisé (figure 2.8, droite) :

$$f(\omega_o, \omega_i) = \frac{\rho}{\pi}. \quad (2.3.2)$$

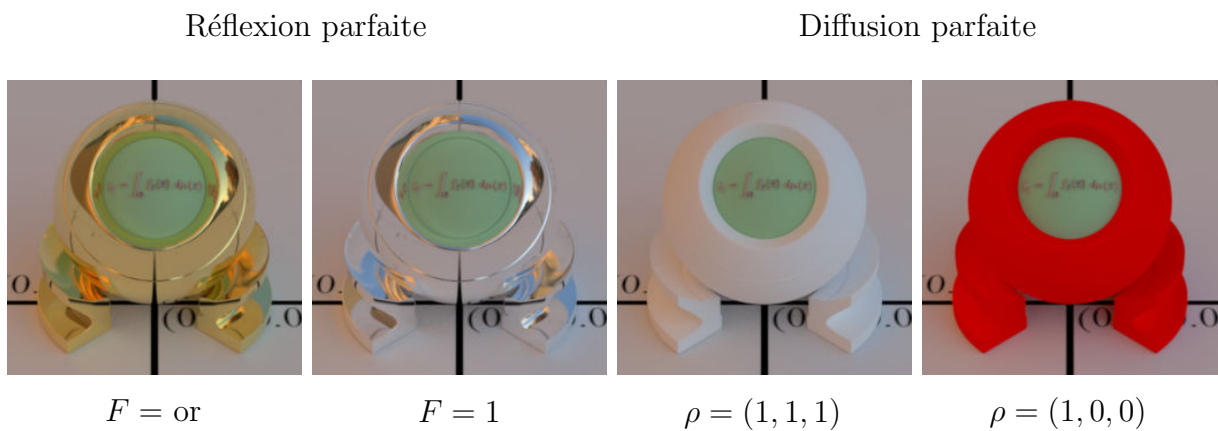


FIGURE 2.8 – À gauche : la réflexion parfaite avec et sans absorption. À droite : la diffusion parfaite, sans et avec absorption.

2.4 Modèles de BRDFs à base de micro-facettes

Les modèles idéaux présentés précédemment sont trop simples pour représenter une grande variété de matériaux. La micro-surface est soit parfaitement plate (réflexion parfaite) ou soit extrêmement rugueuse (diffusion parfaite). Rapidement, des chercheurs ont essayé de trouver un modèle de BRDF permettant de modéliser une surface qui est arbitrairement rugueuse. Phong en 1975 propose un modèle empirique [Pho75] avec un paramètre qui contrôle en quelque sorte la rugosité du matériau. Même s'il ne conserve

pas l'énergie et qu'il n'est pas réciproque, il permet pour la première fois de reproduire des surfaces brillantes. Le modèle de Torrance-Sparrow en 1967 utilise pour la première fois un ensemble aléatoire de micro-facettes parfaitement spéculaires [TS67]. Ce modèle a été introduit en informatique graphique par Blinn en 1977 [Bli77] et une variante plus connue a été proposée par Cook et Torrance en 1981 [CT81]. Ces modèles supposent une distribution de micro-facettes possédant chacune une micro-normale ω_m . La surface est donc caractérisée par une *distribution de normales* (NDF : pour Normal Distribution Function en anglais). Cette distribution de normales D permet de savoir si en général les pentes des facettes sont plutôt fortes ou faibles. La forme de la NDF impacte directement l'apparence des objets.

Si les micro-facettes sont parfaitement spéculaires, nous pouvons alors utiliser une BRDF analytique relativement simple, très connue et très utilisée en informatique graphique [WMLT07] :

$$f(\omega_o, \omega_i) = \frac{F(\omega_o, \omega_h)G_2(\omega_o, \omega_i, \omega_h)D(\omega_h)}{4(\omega_o \cdot \omega_g)(\omega_i \cdot \omega_g)}. \quad (2.4.1)$$

La distribution de normales $D(\omega_h)$ donne la proportion de facettes orientées selon la direction ω_h . Cette quantité est importante pour connaître le nombre de micro-miroirs qui reflètent la lumière, c'est-à-dire ceux qui ont une orientation à mi-chemin entre la direction d'observation ω_o et la direction incidente ω_i . Cette direction est notée ω_h et s'écrit :

$$\omega_h = \frac{\omega_o + \omega_i}{\|\omega_o + \omega_i\|}. \quad (2.4.2)$$

Le terme de Fresnel F permet de modéliser l'absorption de la lumière par la surface (section 2.3). Les autres termes servent à correctement normaliser la BRDF, pour que son intégrale pondérée par le cosinus soit inférieure ou égale à un [CT81], faisant l'impasse sur le multi-rebonds. Comme nous ne modélisons que la réflexion, les produits scalaires $\omega_o \cdot \omega_g$ et $\omega_i \cdot \omega_g$ sont toujours strictement supérieurs à zéro car les directions ω_o et ω_i ont toujours une composante z positive.

La diffusion de la lumière dans une micro-surface représentée par une distribution de normales D est aussi possible avec un modèle à micro-facettes [WMLT07]. Dans ce cas, la BSDF permet de modéliser du verre dépoli. Enfin, les micro-facettes peuvent être parfaitement diffuses, mais cela implique que la BRDF n'a pas de forme close. Oren et Nayar [ON94] ont proposé une fonction analytique qui approxime plutôt bien la forme exacte.

Dans cette thèse, nous avons développé des modèles de BRDF se basant sur la théorie des micro-facettes. Dans le chapitre 3, nous avons adapté l'équation 2.4.1 pour qu'elle soit toujours valide avec des NDFs calculées depuis une empreinte de rayon sur une carte de normales. Se baser sur l'équation 2.4.1 est une bonne idée car cette BRDF respecte toutes les bonnes propriétés énoncées dans les sections 2.1.4 et 2.1.5, sauf la conservation *totale* de l'énergie. Dans les sections suivantes, nous décrivons les principaux termes de cette équation.

2.4.1 Distribution de normales (NDF)

Caractériser une micro-surface par un ensemble de micro-facettes est au premier abord peu intuitif. Il est plus naturel de vouloir dessiner ou sculpter la micro-surface, puis de regarder comment les rayons lumineux se diffusent à son contact. C'est ce qu'ont fait par exemple Westin et al. [WAT92], en précalculant des BRDFs à partir de micro-géométries virtuelles et en les stockant avec des harmoniques sphériques. Ce processus est assez intuitif mais complexe à mettre en œuvre et peu performant. Pour différentes rugosités, il faut sculpter une micro-surface, lancer des rayons dedans, étudier les statistiques de réflectance et stocker une BRDF 4D qui est assez coûteuse en mémoire.

La théorie des micro-facettes est en quelque sorte une réponse à ce problème. Comment représenter une micro-surface avec peu de paramètres en mémoire ? Comment déduire une BRDF directement depuis cette représentation, sans faire de pré-calculs ? La réponse trouvée par la communauté scientifique est la suivante : la distribution de normales (NDF). Avec juste un paramètre contrôlant la variance des micro-normales par rapport à la normale géométrique, nous pouvons représenter une micro-surface dont les micro-normales suivent une distribution gaussienne, par exemple. Utiliser les statistiques de la micro-surface (moyenne et variance des micro-normales) au lieu d'une représentation spatiale explicite (figure 2.9) est beaucoup plus performant, plus compact et tout à fait suffisant pour modéliser la réflexion de la lumière sous pixel (pour des matériaux ne possédant pas de défauts de surface).

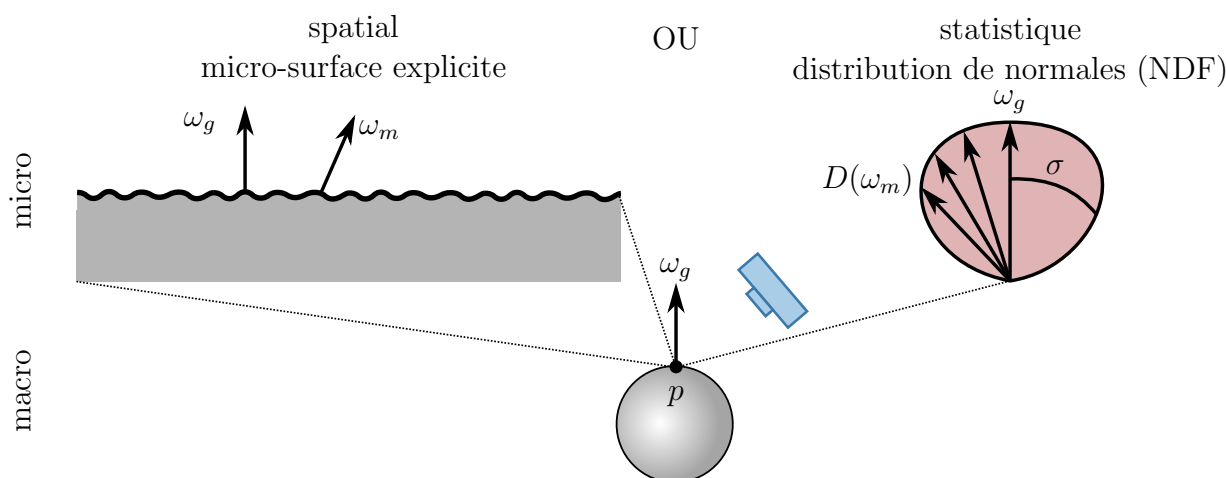


FIGURE 2.9 – Pour représenter la micro-surface en un point p d'une surface, nous pouvons soit utiliser une micro-géométrie explicite (gauche), soit utiliser les statistiques de cette micro-surface paramétrant une distribution de normale $D(\omega_m)$. Ici, l'écart type σ des micro-normales ω_m contrôle l'envergure de la NDF. En pratique, utiliser une représentation statistique est suffisant pour faire le rendu d'une large variété de matériaux.

2.4.2 Distribution de pentes

Dans la littérature, plusieurs distributions de normales D sont proposées, avec chacune leurs avantages et inconvénients et un ou plusieurs paramètres. Souvent, ces distributions 3D se basent sur une distribution 2D de pentes P^{22} . Une micro-normale ω_m est un vecteur 3D normalisé avec une composante z strictement positive pouvant être représenté avec

deux paramètres. Plusieurs représentations existent, mais la plus courante est la pente \tilde{m} . Pour calculer la pente $\tilde{m} = (x_{\tilde{m}}, y_{\tilde{m}})$ associée à une normale $\omega_m = (x_m, y_m, z_m)$, nous utilisons la transformation suivante :

$$\tilde{m} = \begin{pmatrix} -x_m/z_m \\ -y_m/z_m \end{pmatrix}. \quad (2.4.3)$$

Réciproquement, nous pouvons retrouver la normale ω_m à partir de la pente \tilde{m} :

$$\omega_m = \begin{pmatrix} -x_{\tilde{m}} \\ -y_{\tilde{m}} \\ 1 \end{pmatrix} \frac{1}{\sqrt{x_{\tilde{m}}^2 + y_{\tilde{m}}^2 + 1}}. \quad (2.4.4)$$

À partir d'une distribution de pentes $P^{22}(x_{\tilde{m}}, y_{\tilde{m}})$, nous pouvons construire une distribution de normales $D(\omega_m)$ avec la relation suivante

$$D(\omega_m) = \frac{P^{22}(x_{\tilde{m}}, y_{\tilde{m}})}{\cos^4 \theta_m} \quad (2.4.5)$$

où le dénominateur vient du jacobien de la transformation normale-pente $1/\cos^3 \theta_m$ multiplié par l'inverse de la projection des normales sur la macro-surface $1/\cos \theta_m$.

2.4.3 Propriétés des NDFs

Nous pouvons d'ores et déjà dresser la liste des propriétés que la NDF doit respecter dans l'idéal. Actuellement, seules les distributions GGX et de Beckmann respectent *toutes* ces propriétés, même si les performances peuvent être améliorées pour la partie multi-rebonds.

Normalisation sur la macro-surface. La normalisation de l'aire projetée des micro-normales sur la macro-surface est la propriété la plus importante. Il faut que l'intégrale de la distribution des micro-normales projetées sur le plan tangent soit égale à un (figure 2.10) :

$$\begin{aligned} 1 &= \int_{\Omega} D(\omega_m) \langle \omega_m, \omega_g \rangle d\omega_m \\ &= \int_{\mathbb{R}} \int_{\mathbb{R}} P^{22}(x_{\tilde{m}}, y_{\tilde{m}}) dx_{\tilde{m}} dy_{\tilde{m}}. \end{aligned} \quad (2.4.6)$$

Cette distribution doit pouvoir s'échantillonner facilement, pour réduire la variance de l'estimateur de Monte Carlo. La formulation $\langle \omega_m, \omega_g \rangle$ est le produit scalaire ramené à 0 pour les valeurs négatives et renforce le fait que les normales et directions sont définies dans l'hémisphère supérieure Ω .

Normalisation sur la direction d'observation. Dans le but de connaître rapidement la proportion de normales visibles d'une NDF depuis une direction d'observation ω_o , la NDF doit avoir une fonction de *masquage analytique* (section 2.4.10). Cette fonction peut se calculer à partir de la distribution de normales visibles (VNDF, section 2.4.9, figure 2.14). Échantillonner la VNDF au lieu de la NDF permet de réduire encore plus

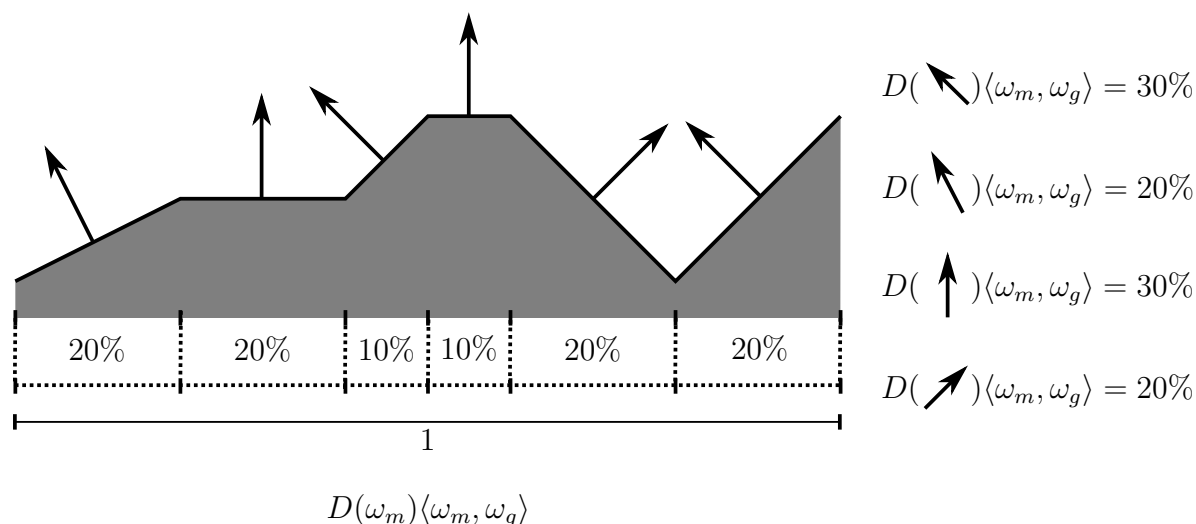


FIGURE 2.10 – La projection des normales de la NDF sur la macro-surface doit être égale à un.

la variance de l'estimateur de Monte Carlo mais là encore, il faut que la procédure d'échantillonnage soit rapidement évaluable.

Multi-rebonds. Dans l'idéal, le *multi-rebonds* doit pouvoir être calculable facilement depuis la micro-géométrie que représente la NDF. C'est l'étape la plus difficile et actuellement, beaucoup de modèles de NDF ne fournissent pas d'algorithme ou de formule le permettant. Si la fonction de densité de réflectance multi-rebonds issue de la NDF existe, elle doit pouvoir s'échantillonner facilement. Nous allons maintenant passer en revue les principales NDFs existantes en informatique graphique.

2.4.4 Distribution de Beckmann

Les propriétés idéales définies au-dessus sont classées par ordre de difficulté croissante. Par définition, une distribution de normales est *normalisée* car c'est une *distribution*. La distribution de Beckmann est l'une des plus connues en informatique graphique. Créée à la base dans le domaine de la physique en 1963 par Beckmann et Spizzichino [BS63], elle modélise une distribution gaussienne des pentes (c'est la densité de probabilité de la loi normale) :

$$P^{22}(x_{\bar{m}}, y_{\bar{m}}) = \frac{1}{\pi\alpha^2} \exp\left(-\frac{x_{\bar{m}}^2 + y_{\bar{m}}^2}{\alpha^2}\right), \quad (2.4.7)$$

où le paramètre de rugosité α est lié à l'écart type des pentes σ avec la relation suivante $\alpha = \sqrt{2}\sigma$. Augmenter ce paramètre augmente donc la rugosité du matériau (figure 2.11).

2.4.5 Distribution GGX

La distribution GGX (Ground Glass unknown) a été démocratisée en informatique graphique par Walter et al. [WMLT07] en 2007, où elle est utilisée pour modéliser la

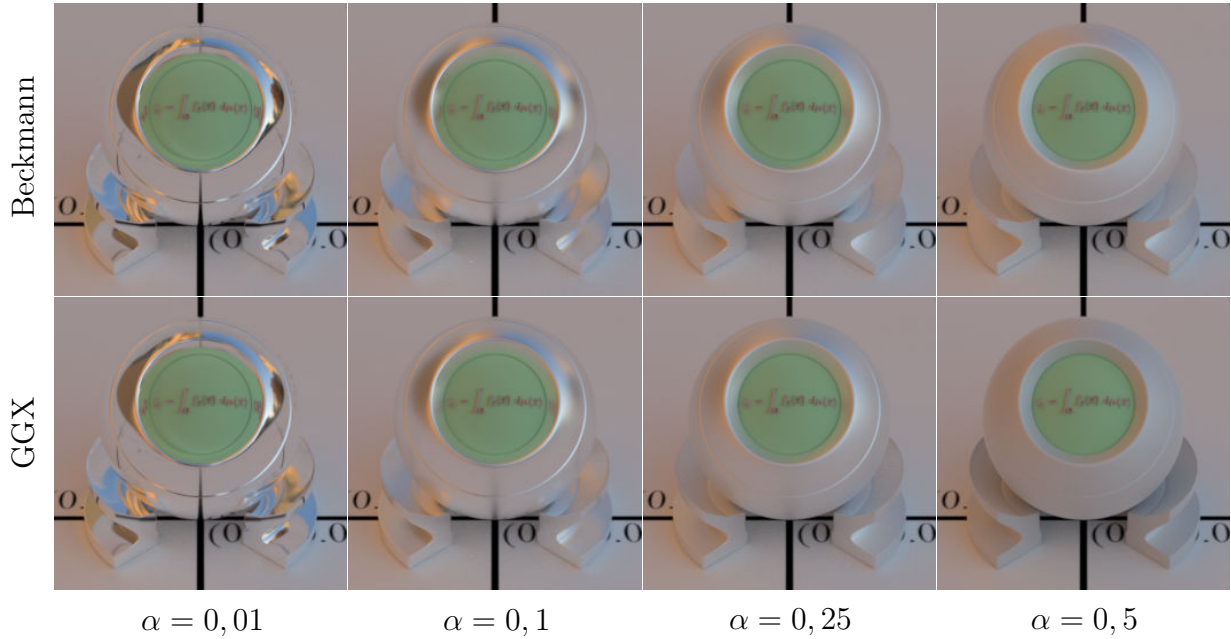


FIGURE 2.11 – Augmenter la rugosité α de Beckmann et de GGX rend le matériau de plus en plus diffus.

diffusion de la lumière dans le verre. Cette distribution a de meilleures correspondances avec des données mesurées et est plus rapide à évaluer que la NDF de Beckmann, et c'est pourquoi elle est la plus utilisée dans l'industrie à l'heure actuelle. Elle a d'abord été introduite en physique par Trowbridge et Reitz [TR75] et une distribution équivalente a été utilisée pour des matériaux volumétriques par Neyret [Ney98]. La distribution GGX donne la densité d'une normale ω_m sur la surface d'une ellipsoïde :

$$P^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi\alpha^2 \left(1 + \frac{x_{\tilde{m}}^2 + y_{\tilde{m}}^2}{\alpha^2}\right)^2}. \quad (2.4.8)$$

Le paramètre α modifie la rugosité du matériau, mais d'une façon différente que le paramètre α de Beckmann (figure 2.11).

2.4.6 Distributions anisotropes

Les distributions de facettes-normales précédentes sont isotropes, c'est-à-dire que la rugosité est la même en x et en y dans l'espace de la surface. Des versions anisotropes de la NDF de Beckmann et de GGX existent, utilisant une rugosité α_x et α_y . La formulation de Beckmann anisotrope est :

$$P^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi\alpha_x\alpha_y} \exp\left(-\frac{x_{\tilde{m}}^2}{\alpha_x^2} - \frac{y_{\tilde{m}}^2}{\alpha_y^2}\right), \quad (2.4.9)$$

où la variation des paramètres α_x et α_y permet de contrôler l'anisotropie du matériau (figure 2.12). La distribution des pentes GGX anisotrope est :

$$P^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi\alpha_x\alpha_y \left(1 + \frac{x_{\tilde{m}}^2}{\alpha_x^2} + \frac{y_{\tilde{m}}^2}{\alpha_y^2}\right)^2}. \quad (2.4.10)$$

Avec deux paramètres au lieu d'un, ce modèle représente plus de surfaces réelles, comme par exemple du métal brossé.

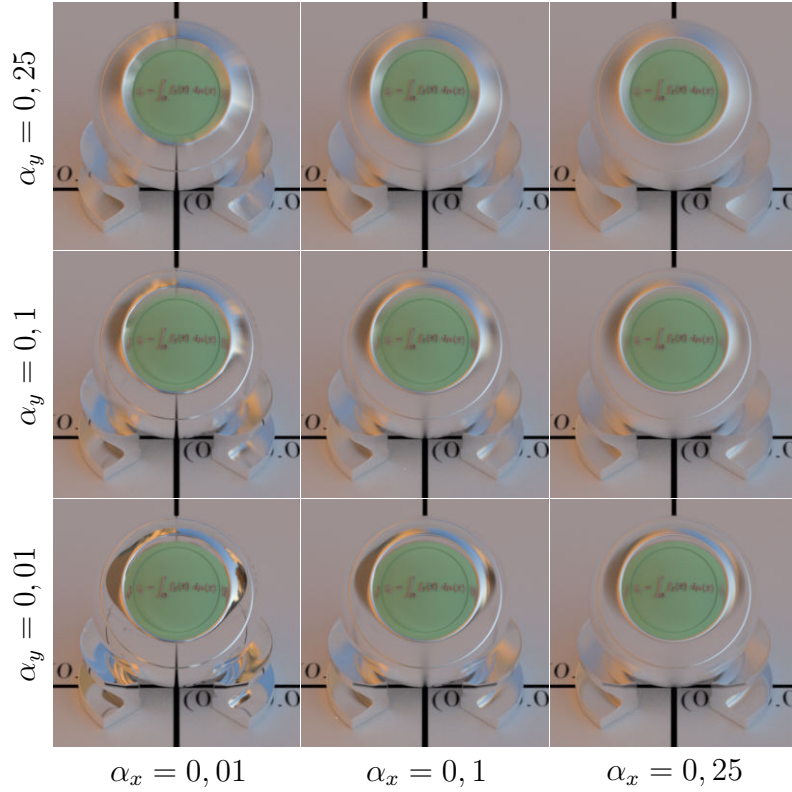


FIGURE 2.12 – Variation en x et en y de la rugosité α de Beckmann, donnant des matériaux anisotropes.

2.4.7 Distributions anisotropes non-alignées sur les axes

Dans le repère planaire tangent à la surface (figure 2.1), nous pouvons faire varier la rugosité en x et en y avec α_x et α_y . En utilisant un coefficient de corrélation c_{xy} entre les axes, ces deux rugosités peuvent être définies pour des axes orthogonaux arbitraires. La NDF de Beckmann non-alignée sur les axes est basée sur une distribution de pentes ayant une PDF égale à une loi normale 2D [Hei14] :

$$P^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi\alpha_x\alpha_y\sqrt{1-c_{xy}^2}} \exp\left(-\frac{1}{1-c_{xy}^2} \left(\frac{x_{\tilde{m}}^2}{\alpha_x^2} + \frac{y_{\tilde{m}}^2}{\alpha_y^2} - \frac{2c_{xy}x_{\tilde{m}}y_{\tilde{m}}}{\alpha_x\alpha_y}\right)\right) \quad (2.4.11)$$

Pour la distribution GGX, une version non-alignée sur les axes existe aussi [TK19]. Avec seulement 3 paramètres, nous pouvons maintenant représenter beaucoup de micro-surfaces différentes.

2.4.8 Distributions GTR, STD et autres

Même si les distributions de Beckmann et GGX sont les plus utilisées, il existe d'autres NDFs. La distribution GGX a une « queue » plus longue que celle de Beckmann (figure 2.13). Dit autrement, pour une rugosité α égale, la densité des normales les plus pentues est plus grande pour GGX que pour la NDF de Beckmann. Les distributions GTR (Generalized-Trowbridge-Reitz) [Bur12] et STD (Student's t-Distribution) [RBMS17] sont des distributions qui ont chacune un paramètre γ leur permettant de contrôler la longueur de la queue de la distribution. La première n'est pas normalisable et ne possède pas de procédure d'échantillonnage analytique alors que STD a ces deux propriétés. Ces deux distributions restent néanmoins peu utilisées en production car comparées à Beckmann et GGX, elles ne possèdent pas de procédure d'échantillonnage des normales *visibles* et actuellement, aucune méthode utilisable en production n'a été proposée pour gérer le multi-rebonds dans la micro-surface qu'elles représentent. Une liste exhaustive des NDFs existantes en informatique graphique ainsi que leurs propriétés se trouve dans le papier de Ribardi re et al. [RBSM19]. Les auteurs proposent une m thode g n rant plusieurs instances de micro-g om tries   partir de NDFs arbitraires dans le but de valider des potentiels futurs mod les, incluant le multi-rebonds.

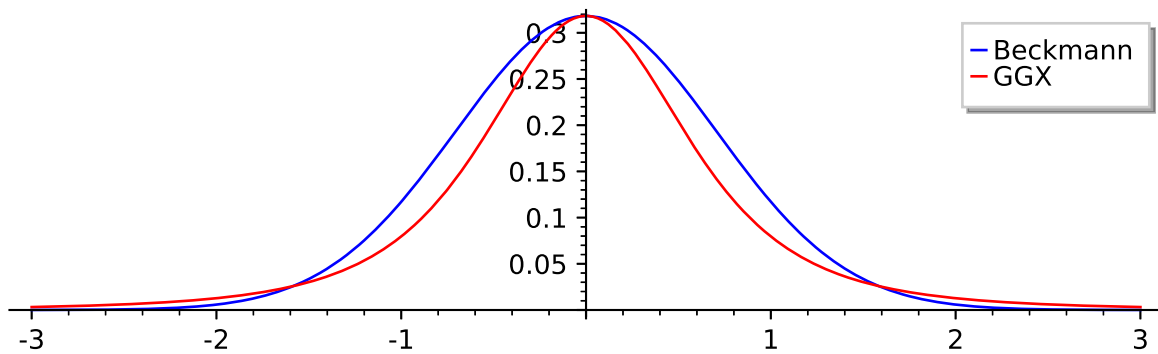


FIGURE 2.13 – Comparaison entre la distribution de pentes $P^{22}(x_{\tilde{m}}, \alpha = 1, y_{\tilde{m}} = 0)$ de Beckmann et de GGX. La queue de la distribution GGX est plus « longue ».

2.4.9 Distribution de normales visibles (VNDF)

La densité de normales en un point p d'une surface décrit de manière compacte une micro-g om trie explicite. Mais lors d'un rendu, nous cherchons en priorit    conna tre ce qui est *visible* depuis la cam ra. Cette affirmation est aussi vraie pour la micro-g om trie, et c'est pour cela que des  tudes relativement r centes [Hd14, Hei18] ont propos  d'utiliser la distribution de normales *visibles* (VNDF, pour Visible Normal Distribution Function en anglais) au lieu de la NDF, pour l' chantillonnage. La VNDF $D_{\omega_o}(\omega_m)$, illustr e en figure 2.14, est une sous-partie de la BRDF   base de micro-facettes. Elle est utile pour am liorer l' chantillonnage et elle est d finie comme suit [Hei14] :

$$D_{\omega_o}(\omega_m) = \frac{G_1(\omega_o, \omega_m)D(\omega_m)\langle\omega_o, \omega_m\rangle}{\cos\theta_o}. \quad (2.4.12)$$

La fonction de masquage $G_1(\omega_o, \omega_m)$ est choisie pour que la VNDF soit normalisée :

$$\int_{\Omega} D_{\omega_o}(\omega_m) d\omega_m = 1. \quad (2.4.13)$$

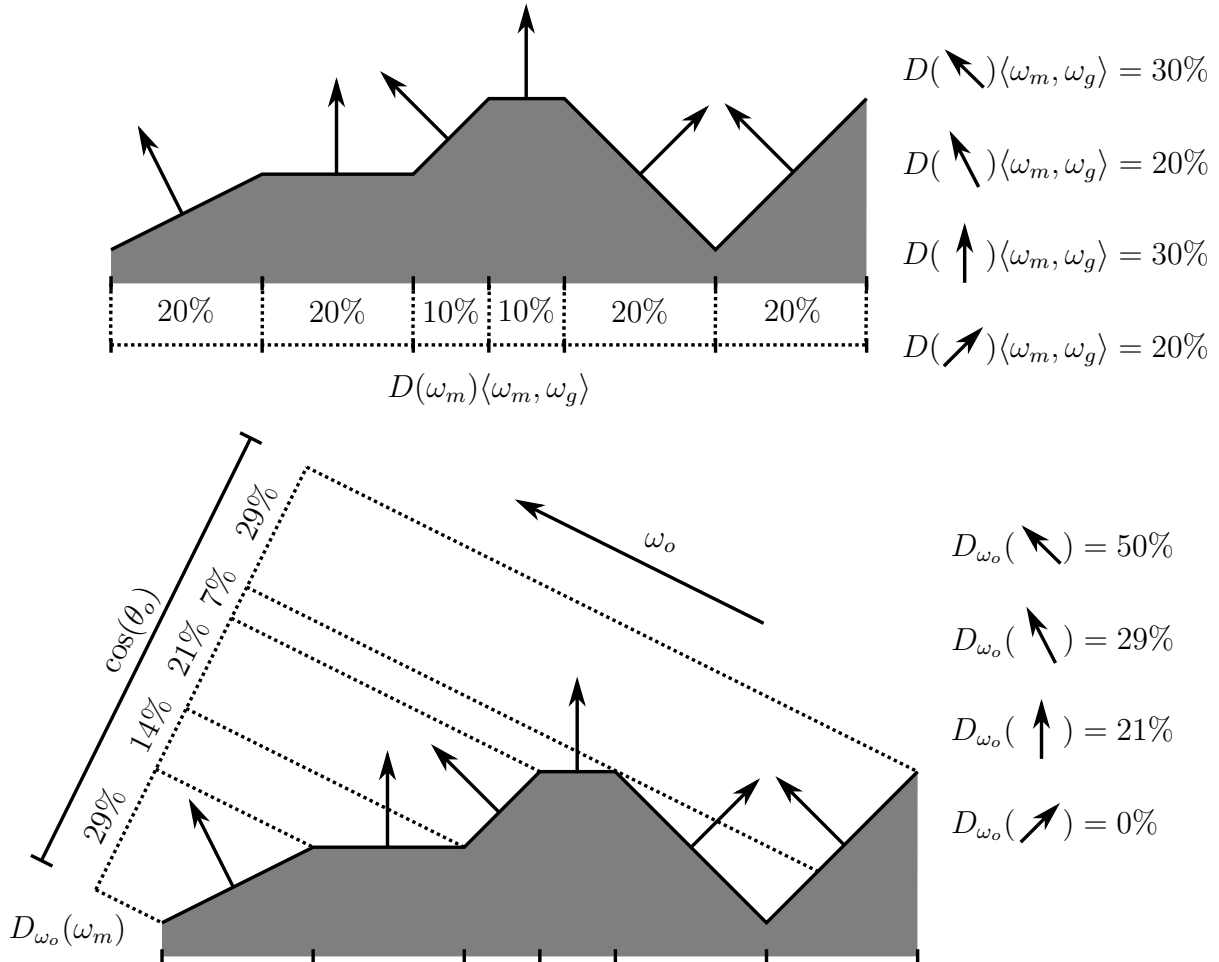


FIGURE 2.14 – La NDF (haut) donne la proportion de micro-normales ω_m projetées sur la macro-surface. La VNDF (bas) donne la proportion de normales projetées vers une direction d’observation ω_o . À angle rasant, ces deux distributions sont très différentes pour une même micro-surface.

2.4.10 Fonctions de masquage

La VNDF permet d’échantillonner la partie visible de la micro-géométrie, évitant ainsi le tirage de normales tournant le dos et ne renvoyant pas la lumière (figure 2.14). Elle implique une fonction de masquage qui donne la proportion de facettes visibles ω_m depuis une direction ω_o . Cette fonction de masquage $G_1(\omega_o, \omega_m)$ a des valeurs comprises entre 0 et 1 et permet à la BRDF de ne pas créer d’énergie. Plusieurs formulations permettent de satisfaire l’équation 2.4.13. Nous devons choisir une contrainte supplémentaire pour qu’il n’y en ait plus qu’une. Cette contrainte est le *profil de la micro-surface*. Un état de l’art complet sur les fonctions de masquage a été fait par Heitz [Hei14].

Modèle à V-cavités. Le modèle à base de V-cavités permet de trouver une solution unique à G_1 pour la normalisation de la VNDF. Plusieurs modèles connus en informatique graphique les utilisent, que ce soit avec des micro-facettes parfaitement spéculaires [CT81] ou parfaitement diffuses [ON94]. Ce profil de micro-surface suppose qu'au niveau d'un point, il existe une infinité de V-cavités, plus ou moins creuses. Une V-cavité a deux normales : une normale ω_m et sa symétrique ω_s (figure 2.15).

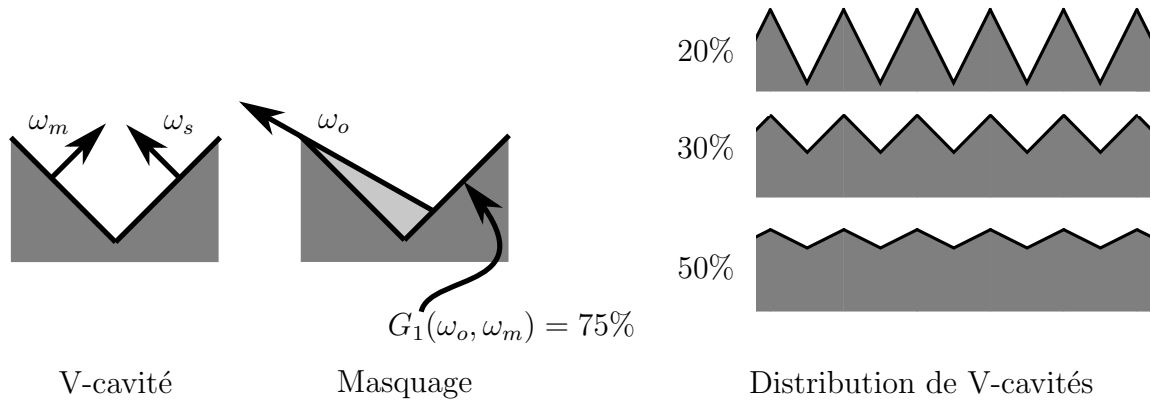


FIGURE 2.15 – Une V-cavité est définie par deux normales symétriques par rapport à la normale géométrique (gauche). Cette configuration simple permet de calculer le masquage G_1 assez facilement (milieu). Lors du calcul de la BRDF, il suffit de connaître la réflectance d'une V-cavité orientée selon ω_h et de la pondérée par sa densité, qui est donnée par la NDF (droite).

Définir la micro-surface par un ensemble de V-cavités permet de dériver un masquage G_1 simple et indépendant de la NDF, car le masquage n'est pas lié à un ensemble de micro-facettes suivant une distribution. Dans ce cas, il est juste lié à deux facettes symétriques (figure 2.15) et est facilement dérivable en utilisant soit de la trigonométrie, soit la conservation de l'aire projetée de la micro-surface visible. La formule du masquage d'une V-cavité est la suivante :

$$G_1(\omega_o, \omega_m) = H(\omega_o \cdot \omega_m) \min \left(1, 2 \frac{(\omega_m \cdot \omega_g)(\omega_o \cdot \omega_g)}{\langle \omega_o, \omega_m \rangle} \right). \quad (2.4.14)$$

Dans l'équation précédente, H est la fonction de Heaviside. Bien que cette formulation permette de correctement normaliser la VNDF et la BRDF, elle reste néanmoins peu représentative de la réalité. Il y a des différences importantes à angles d'observation rasants entre une BRDF utilisant ce modèle et une BRDF de référence, calculée à partir d'une micro-géométrie explicite ayant la même NDF. Néanmoins, avec les V-cavités, la conservation de l'énergie est respectée et le multi-rebonds peut être modélisé assez facilement [XH18, LJJ+18]. Nous utilisons ce profil de micro-surface dans le chapitre 4 pour représenter des matériaux à micro-reflets.

Modèle de Smith. Le modèle de Smith [Smi67] est aussi très utilisé en informatique graphique et est une bonne approximation d'une surface réelle. Il suppose que la surface est un ensemble de facettes orientées aléatoirement, sans jonction entre elles (figure 2.16). Cette supposition sous-entend que la surface n'est pas réellement continue. Une facette et sa voisine peuvent donc être orientées de manière très différente. Cette représentation

permet de séparer le masquage G_1 en un masquage local G_1^{local} et un masquage distant G_1^{dist} , car les propriétés locales et distantes sont décorréées :

$$G_1(\omega_o, \omega_m) = G_1^{\text{local}}(\omega_o, \omega_m) G_1^{\text{dist}}(\omega_o). \quad (2.4.15)$$

Le masquage local renvoie un si la normale est visible depuis la direction d'observation, zéro sinon :

$$G_1^{\text{local}}(\omega_o, \omega_m) = H(\omega_o \cdot \omega_m). \quad (2.4.16)$$

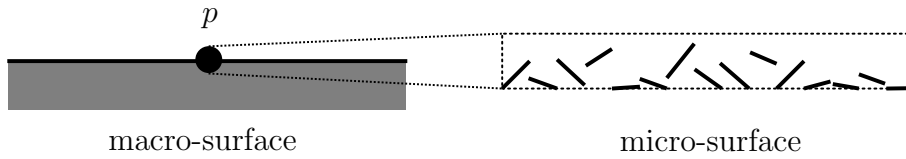


FIGURE 2.16 – La surface de Smith suppose que la micro-surface est un ensemble de micro-facettes non corrélées.

La fonction de masquage distante $G_1^{\text{dist}}(\omega_o)$ ne dépend pas de la micro-normale ω_m . Elle donne la probabilité de visibilité d'un point de la micro-surface en fonction de la direction d'observation ω_o . Plus cette direction sera rasante, plus cette probabilité sera faible, car des points distants cachent de plus en plus de micro-géométrie. En utilisant le profil de Smith et le fait que la VNDF doit être normalisée, c'est-à-dire que la projection vers ω_o de la macro-surface doit être égale à la projection de la micro-surface (équation 2.4.13), nous pouvons dériver le masquage distant :

$$\int_{\Omega} \frac{G_1^{\text{dist}}(\omega_o) G_1^{\text{local}}(\omega_o, \omega_m) D(\omega_m) \langle \omega_o, \omega_m \rangle}{\cos \theta_o} d\omega_m = 1. \quad (2.4.17)$$

L'aire projetée de la macro-surface $\cos \theta_o$ ne dépend pas de ω_m . De plus, avec le profil de Smith, le masquage distant G_1^{dist} peut aussi sortir de l'intégrale. L'équation précédente peut donc être réécrite pour isoler le masquage distant, et après simplification, donne :

$$G_1^{\text{dist}}(\omega_o) = \frac{\cos \theta_o}{\int_{\Omega} D(\omega_m) \langle \omega_o, \omega_m \rangle d\omega_m}. \quad (2.4.18)$$

La fonction de masquage de Smith est au final :

$$G_1(\omega_o, \omega_m) = H(\omega_o, \omega_m) \frac{\cos \theta_o}{\int_{\Omega} D(\omega_m) \langle \omega_o, \omega_m \rangle d\omega_m}. \quad (2.4.19)$$

Cette formulation avec une intégrale [APS00] sur les micro-normales ω_m n'est pas la plus connue dans la littérature. La forme générale du masquage de Smith est [Bro80, WMLT07] :

$$G_1(\omega_o, \omega_m) = \frac{H(\omega_o, \omega_m)}{1 + \Lambda(\omega_o)}, \quad (2.4.20)$$

où cette fois l'intégrale est « cachée » à l'intérieur du terme Λ , définie ici dans l'espace

des pentes :

$$\Lambda(\omega_o) = \frac{1}{\cot \theta_o} \int_{\cot \theta_o}^{\infty} \int_{-\infty}^{+\infty} (x_{\bar{m}} - \cot \theta_o) P^{22}(x_{\bar{m}}, y_{\bar{m}}) dy_{\bar{m}} dx_{\bar{m}}. \quad (2.4.21)$$

Contrairement au modèle utilisant des V-cavités, le masquage est *lié* à la NDF. L'intégrale a une solution analytique pour la distribution de normales GGX et de Beckmann et des approximations polynomiales très précises et plus rapides à évaluer existent [WMLT07, Hei14]. Notre méthode présentée dans le chapitre 3 utilise le modèle de micro-surface de Smith pour des matériaux à micro-reflets.

2.4.11 Masquage-ombrage et simple rebond

À l'origine, les modèles de BRDFs à base de micro-facettes simulent un rebond lumineux dans la micro-surface, par simplicité. Pour cela, la fonction d'*ombrage* est utilisée. Elle donne la proportion de facettes visibles depuis la lumière. Elle est égale à la fonction de masquage, mais elle prend en paramètre la direction incidente ω_i au lieu de la direction d'observation ω_o . Pour ne simuler qu'un rebond lumineux, il suffit « d'interdire » toutes les réflexions lumineuses dans les zones d'ombre, car ces zones reflètent seulement de la lumière ayant fait plus d'un rebond dans la micro-surface (figure 2.17). Modéliser les rebonds directs est donc assez simple. Il suffit de connaître la proportion de facettes visibles depuis la direction d'observation ω_o et d'incidence ω_i , ce que donne la fonction de masquage-ombrage $G_2(\omega_o, \omega_i)$.

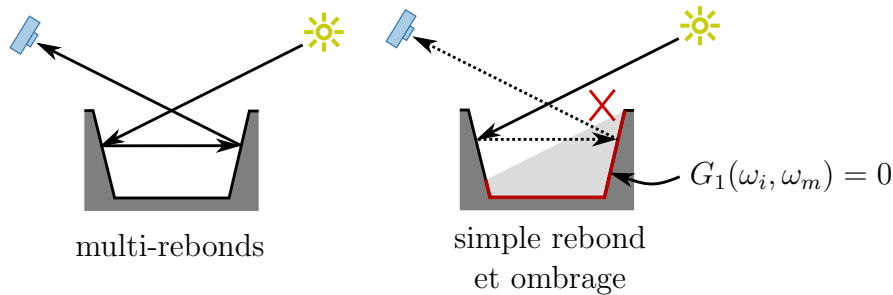


FIGURE 2.17 – L'ombrage permet de simuler qu'un rebond dans la micro-géométrie. Ici les zones rouges sont à l'ombre. Nous considérons qu'elles ne reflètent pas la lumière.

La forme séparable du masquage-ombrage est la plus simple et la plus utilisée en informatique graphique [WMLT07] :

$$G_2(\omega_o, \omega_i, \omega_m) = G_1(\omega_o, \omega_m)G_1(\omega_i, \omega_m). \quad (2.4.22)$$

Cette forme a tendance à surestimer la proportion de micro-géométrie qui est ombrée, car elle ne prend pas en compte les corrélations possibles entre le masquage et l'ombrage. Pour le modèle de Smith, une formulation relativement simple existe prenant en compte les corrélations dues aux hauteurs des micro-facettes [RDP05] :

$$G_2(\omega_o, \omega_i, \omega_m) = \frac{H(\omega_o \cdot \omega_m)H(\omega_i \cdot \omega_m)}{1 + \Lambda(\omega_o) + \Lambda(\omega_i)}. \quad (2.4.23)$$

Malgré tout, les corrélations dues aux directions ne sont pas considérées. Le cas le

plus défavorable est quand l'observateur se situe au même endroit que la lumière, à angle rasant. Dans ce cas, tout ce qui est visible n'est pas à l'ombre, mais l'équation précédente considérera qu'il y a des zones d'ombre. Certains modèles prennent en compte ces corrélations [vGSK98, APS00, HBP13].

2.4.12 Multi-rebonds dans les micro-facettes

Comme mentionné dans la section précédente, le modèle classique des micro-facettes ne simule qu'un rebond lumineux. C'est-à-dire que quand la surface n'est pas absorbante ($F = 1$), la BRDF f_1 peut perdre de l'énergie car les rayons intersectant deux fois ou plus la micro-géométrie ne sont pas pris en compte, comme évoqué dans la section 2.1.4 :

$$\int_{\Omega} f_1(\omega_o, \omega_i) \cos \theta_i d\omega_i \leq 1. \quad (2.4.24)$$

Par exemple, avec une distribution GGX et une rugosité de $\alpha = 1$, la perte d'énergie peut atteindre 60 %. Récemment, la communauté scientifique en informatique graphique essaye d'avoir une BRDF f_{∞} conservant l'énergie à 100 %, ce qui revient à prendre en charge le multi-rebonds (figure 2.18) :

$$\int_{\Omega} f_{\infty}(\omega_o, \omega_i) \cos \theta_i d\omega_i = 1. \quad (2.4.25)$$

Surface de Smith et marches aléatoires. Heitz et al. ont proposé une méthode multi-rebonds utilisant les distributions de Beckmann et GGX et le modèle de Smith [HHdD16]. Dans ce travail, ils considèrent la surface comme étant un volume de matière et utilisent la théorie des micro-paquettes (micro-flakes) [JAM⁺10, HDCD15] pour diffuser les rayons jusqu'à ce qu'ils quittent la surface (figure 2.19). Ces rayons n'existent pas vraiment et sont plutôt des marches aléatoires (random walks). Les probabilités d'intersection avec la micro-géométrie et les fonctions de diffusion (fonctions de phase) permettent respectivement de savoir si un chemin aléatoire touche une facette et vers où la lumière est reflétée au point d'intersection. L'aspect aléatoire réside dans le fait que pour chaque micro-chemin, des nombres aléatoires décident si le chemin est intersecté ou non, et vers où il est reflété. Pour connaître la BRDF multi-rebonds f_{∞} , il faut simuler plusieurs marches aléatoires, calculer leur contribution $E_{1,\dots,N}(\omega_o, \omega_i)$ pour N rebonds, et prendre la moyenne $\mathbb{E}[\]$ sur l'espace des marches :

$$f_{\infty}(\omega_o, \omega_i) = \frac{\mathbb{E}[E_{1,\dots,N}(\omega_o, \omega_i)]}{\cos \theta_i}. \quad (2.4.26)$$

Pour réduire la variance de l'estimateur, Heitz et al. proposent d'utiliser la forme close du premier rebond f_1 pour des micro-facettes parfaitement spéculaires :

$$f_{\infty}(\omega_o, \omega_i) = f_1(\omega_o, \omega_i) + \frac{\mathbb{E}[E_{2,\dots,N}(\omega_o, \omega_i)]}{\cos \theta_i} \quad (2.4.27)$$

où f_1 est l'équation 2.4.1. Ils proposent aussi de faire des marches aléatoires bidirectionnelles et de forcer les intersections avec les hauteurs pour parcourir plus

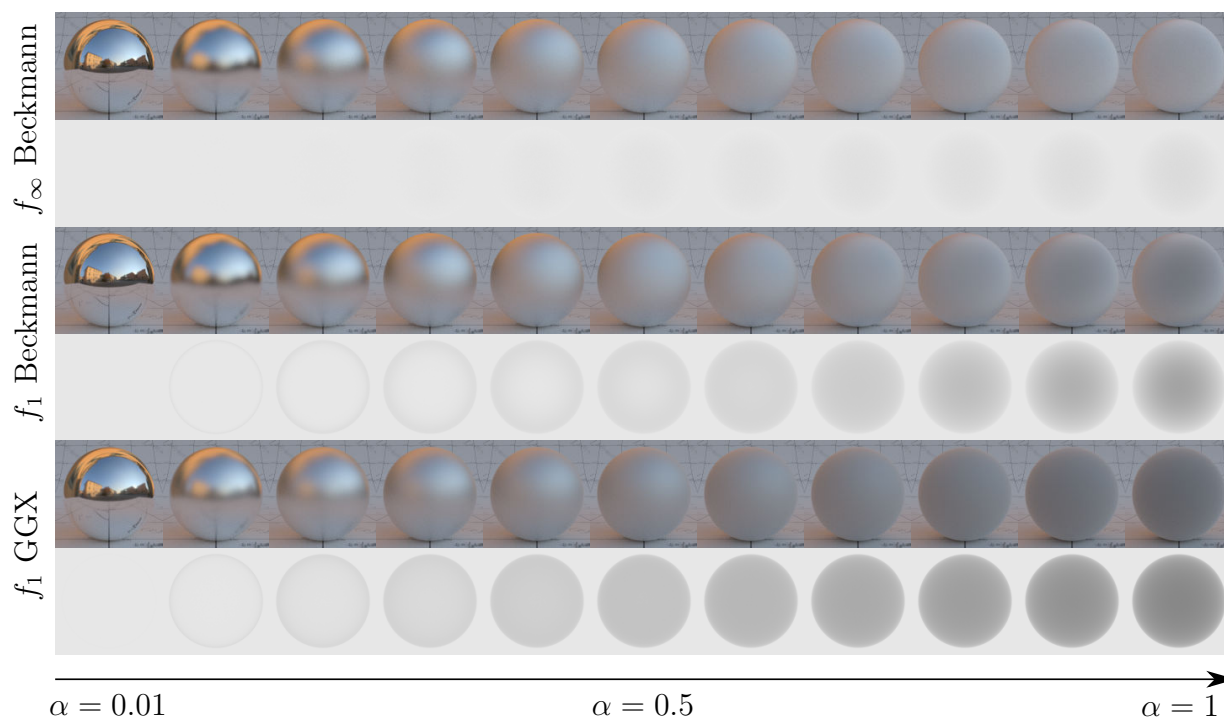


FIGURE 2.18 – Rendus et tests du four blanc. Le modèle classique de BRDF à micro-facettes f_1 , simple rebond, perd de l'énergie à cause de l'ombrage. La distribution de Beckmann engendre une perte moins importante que la distribution GGX car la queue de la distribution de Beckmann est moins longue, et donc en moyenne, les facettes sont moins pentues. La référence f_∞ en haut utilise la méthode de Heitz [HHdD16] prenant en charge le multi-rebonds. Le profil de micro-surface est celui de Smith.

rapidement l'espace des marches possibles. Malgré ces techniques, l'évaluation reste un processus assez coûteux comparé à une évaluation directe d'un modèle simple rebond. Cependant, cette méthode est utilisable dans un moteur de rendu basée sur Monte Carlo et donne des résultats comparables à une référence obtenue numériquement à partir d'une micro-géométrie gaussienne. Cette méthode peut être considérée comme une référence rapide à évaluer pour valider des futurs modèles multi-rebonds. Une étude plus approfondie sur la relation entre le transport de la lumière dans un volume et le multi-rebonds dans une surface de Smith est disponible dans le papier de Dupuy et al. [DHd16].

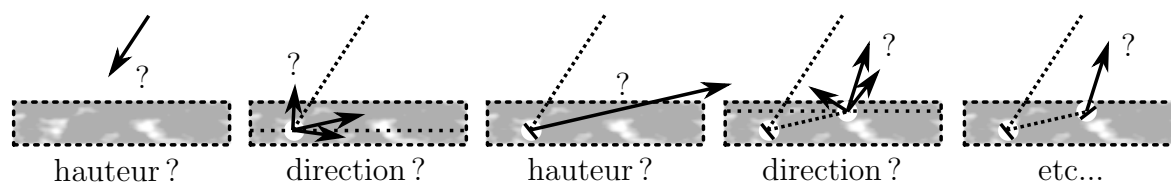


FIGURE 2.19 – Modéliser le multi-rebonds dans une micro-surface de Smith peut se faire en utilisant des marches aléatoires (random walks). Pour cela, il faut échantillonner une hauteur (profondeur) puis une direction jusqu'à ce que le rayon quitte la micro-surface.

V-cavités. La méthode précédente de prise en charge du multi-rebonds suppose un profil de micro-surface de Smith. Des méthodes utilisant le profil à V-cavités existent aussi. Oren et Nayar [ON94] ont proposé un modèle 2-rebonds pour des facettes parfaitement diffuses. Pour des facettes parfaitement spéculaires, des formules analytiques permettent de connaître la réflectance après N rebonds [Zip66, LJJ⁺18, XH18]. Comparées à Heitz [HHdD16], ces méthodes sont plus performantes, mais la solution existe seulement grâce à la configuration très simple des V-cavités. Malheureusement, la forte corrélation entre les facettes symétriques provoque des artéfacts visuels qui sont dus à des singularités (Dirac) et des discontinuités du modèle. De plus, le profil des V-cavités étant peu représentatif de la réalité, les apparences obtenues sont relativement lisses comparées à l'apparence d'une micro-surface gaussienne (de Beckmann) réelle, qui apparaît beaucoup plus rugueuse. Ce manque de correspondance avec une référence était déjà visible avec le modèle simple rebond, mais le multi-rebonds l'accentue encore plus. Il est à noter que la réflectance analytique après N rebonds existe aussi dans des cavités sphériques [KVDDN99].

Compensation d'énergie. Une autre méthode pour modéliser le multi-rebonds consiste à calculer la perte d'énergie induite par une formulation simple rebond f_1 , puis à la réintroduire directement (figure 2.20). C'est Kelemen et Szirmay-Kalos [KSK01] qui ont proposé cette méthode pour la première fois, mais pour des matériaux opaques et isotropes. Ensuite Jakob et al. [JdJM14] ont proposé un modèle général pour gérer des matériaux multicouches, où chaque couche est rugueuse. Pour tenir compte du multi-rebonds, ils ont généralisé le concept de Kelemen et Szirmay-Kalos pour la transmission. Les résultats sont concluants, mais le modèle est lourd, n'est pas adapté spécifiquement aux matériaux à une couche et nécessite des pré-calculs. Pour remédier à cela, Conty et Kulla [CK17] ont proposé un modèle dédié aux BSDFs à micro-facettes simple couche. Leur objectif a été de réduire le plus possible l'empreinte mémoire et de pouvoir faire varier la rugosité, sans refaire de pré-calculs complexes.

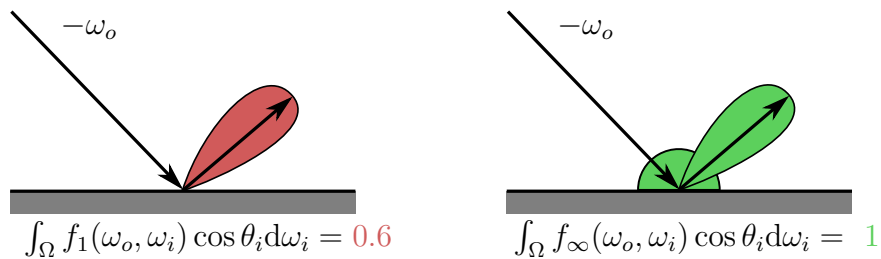


FIGURE 2.20 – Une BRDF simple rebond f_1 (gauche) peut être compensée en lui ajoutant un autre terme, donnant une BRDF f_∞ conservant l'énergie (droite).

La compensation se base sur l'*albédo directionnel* E_1 , appelé aussi *terme d'énergie* ou *réflectance directionnelle-hémisphérique*, qui est l'intégrale de la BRDF simple rebond, pondérée par le cosinus $\cos \theta_i$, sur toutes les directions incidentes ω_i :

$$E_1(\omega_o) = \int_{\Omega} f_1(\omega_o, \omega_i) \cos \theta_i d\omega_i. \quad (2.4.28)$$

Ce terme donne la quantité d'énergie reflétée après un rebond dans la micro-surface dans une direction d'observation et en prenant en compte toutes les directions incidentes. Ses

valeurs varient entre 0 et 1 en fonction de la direction et de la BRDF. La BRDF multi-rebonds f_∞ de Kulla et Conty [CK17] est :

$$f_\infty(\omega_o, \omega_i) = f_1(\omega_o, \omega_i) + \frac{(1 - E_1(\omega_o))(1 - E_1(\omega_i))}{\pi(1 - 2\pi E_{\text{moy}})} \quad (2.4.29)$$

où

$$E_{\text{moy}} = \frac{1}{\pi} \int_{\Omega} E_1(\omega) \cos \theta d\omega \quad (2.4.30)$$

est la *réflectance hémisphérique-hémisphérique*, donnant l'énergie simple rebond moyenne pour toutes les directions. Cette formulation sera reprise par Stephen Hill [Hil18a] et Carmelo J. Fdez-Agüera [FA19]. Cette dernière méthode est optimisée pour le rendu temps réel utilisant des cartes d'environnement. Ces trois dernières méthodes ont une formulation réciproque, mais ne sont pas faciles à échantillonner car la forme de la compensation est complexe. Turquin [Tur19] propose de mettre à l'échelle la BRDF simple rebond (figure 2.21) :

$$\begin{aligned} f_\infty(\omega_o, \omega_i) &= f_1(\omega_o, \omega_i) + \frac{1 - E_1(\omega_o)}{E_1(\omega_o)} f_1(\omega_o, \omega_i) \\ &= \underbrace{\frac{1}{E_1(\omega_o)}}_{\text{mise à l'échelle}} f_1(\omega_o, \omega_i). \end{aligned} \quad (2.4.31)$$

Dans ce cas, il peut utiliser la procédure d'échantillonnage de f_1 . Néanmoins, elle n'est pas parfaite car des échantillons de direction peuvent être perdus. Ce sont ceux qui sont dirigés vers l'intérieur de la surface ou ombrés par la micro-surface.

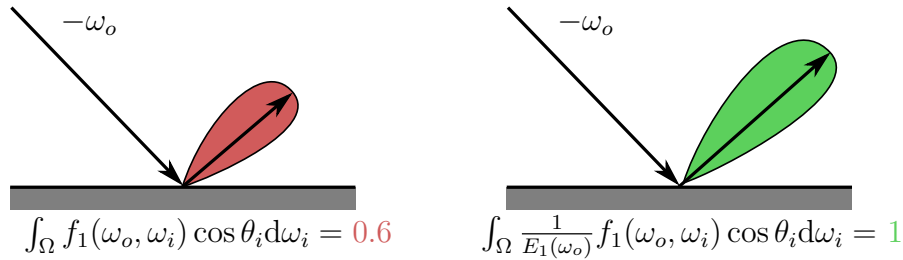


FIGURE 2.21 – Turquin propose de mettre à l'échelle la BRDF simple rebond f_1 (gauche) pour que son intégrale fasse un (droite).

Dans le chapitre 4, nous proposons une méthode permettant d'utiliser la compensation d'énergie et une procédure d'échantillonnage optimale. Pour cela, nous utilisons une BRDF de compensation f_{2+} qui conserve l'énergie à 100%. Notre formulation est :

$$f_\infty(\omega_o, \omega_i) = f_1(\omega_o, \omega_i) + (1 - E_1(\omega_o)) f_{2+}(\omega_o, \omega_i). \quad (2.4.32)$$

Malheureusement, la méthode de Turquin et la nôtre ont une formulation non réciproque, ce qui contraint l'utilisation de ces BRDFs à du tracer de chemins unidirectionnels. Ces techniques de compensation sont prometteuses car les résultats obtenues sont proches de ceux de Heitz [HHd16], et plus rapides.

Les équations 2.4.29, 2.4.31 et 2.4.32 ne prennent pas en compte l'absorption d'énergie au-delà du premier rebond. Kulla et Conty [CK17] s'inspirent de Jakob et al. [JdJM14] et utilisent le fait que leur compensation est diffuse. En s'appuyant sur le terme de Fresnel moyen

$$F_{\text{moy}} = \int_{\Omega} F(\omega) \cos \theta d\omega, \quad (2.4.33)$$

ils calculent un terme de Fresnel multi-rebonds

$$F_{2+} = \frac{F_{\text{moy}} E_{\text{moy}}}{1 - F_{\text{moy}}(1 - E_{\text{moy}})} \quad (2.4.34)$$

et l'utilisent dans l'équation 2.4.29 pour mettre à l'échelle la BRDF multi-rebonds :

$$f_{\infty}(\omega_o, \omega_i) = f_1(\omega_o, \omega_i) + F_{2+} \frac{(1 - E_1(\omega_o))(1 - E_1(\omega_i))}{\pi(1 - 2\pi E_{\text{moy}})}. \quad (2.4.35)$$

Cette mise à l'échelle est trop claire, et Hill [Hil18a] propose d'utiliser F_{moy}^2 au lieu de F_{moy} au numérateur :

$$F_{2+} = \frac{F_{\text{moy}}^2 E_{\text{moy}}}{1 - F_{\text{moy}}(1 - E_{\text{moy}})}. \quad (2.4.36)$$

Cette fois, comparés aux références, les résultats sont trop sombres et Hill [Hil18b] propose de précalculer un terme d'énergie incorporant Fresnel, donnant une bonne correspondance avec la référence, mais obligeant l'utilisation d'une formulation asymétrique et des pré-calculs supplémentaires.

Turquin [Tur19] propose, pour son modèle, de simplement utiliser le terme de Fresnel simple rebond F_1 comme approximation du terme multi-rebonds, donnant la BRDF suivante :

$$f_{\infty}(\omega_o, \omega_i) = f_1(\omega_o, \omega_i) + F_1 \frac{1 - E_1(\omega_o)}{E_1(\omega_o)} f_1(\omega_o, \omega_i). \quad (2.4.37)$$

Dans le chapitre 4, nous utilisons la même approximation, car elle donne des résultats acceptables, où l'effet de saturation de la couleur est visible.

2.5 Rendu de cartes de normales

Les définitions des BRDFs données dans les sections précédentes supposent que les réflexions ont lieu sur une surface macroscopiquement plane, dont la normale est la normale géométrique ω_g . Dans le but de rajouter des détails aux surfaces des objets sans modifier ou complexifier leur macro-géométrie, Blinn [Bli78] propose de perturber la normale géométrique. Cette nouvelle normale ω_p est utilisée pour calculer la réflectance, à la place de ω_g . Elle est définie pour chaque point u de la surface (figure 2.22) grâce à une *carte de normales*, qui est soit une texture, soit une fonction procédurale. Comparée à une carte de déplacements utilisant la tessellation pour reconstruire la géométrie, la carte de normales est plus performante mais moins réaliste. L'ajout de détails surfaciques

a lieu au niveau microscopique, c'est-à-dire au niveau de la BRDF, alors qu'une carte de déplacements les ajoute au niveau macroscopique. Les deux techniques, macro et micro, sont encore utilisables dans les moteurs de rendu professionnels, preuve que les cartes de hauteurs et les cartes de normales sont encore très utilisées aujourd'hui. Cette thèse a pour centre d'intérêt la BRDF, c'est-à-dire la micro-géométrie. Nous traiterons donc exclusivement des cartes de normales et de leurs BRDFs basées physique associées.

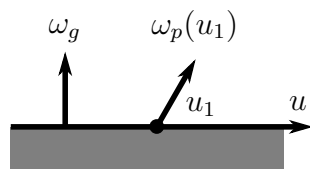


FIGURE 2.22 – Une carte de normales définit une normale perturbée ω_p par rapport à la normale géométrique ω_g pour chaque point u de la surface.

Calculer la réflectance autour de la normale perturbée ω_p au lieu de la normale géométrique ω_g n'est pas trivial. Il faut se rappeler que l'algorithme du tracer de chemins calcule les intersections avec de la macro-géométrie, même si une carte de normales est plaquée sur la surface de l'objet. Nous ne pouvons pas directement utiliser ω_p au lieu de ω_g sans prendre certaines précautions. Si la surface est opaque et que la direction incidente est orientée vers l'intérieur de l'objet, il se peut que macroscopiquement, la lumière ne soit pas visible mais que « mesoscopiquement », elle le soit (figure 2.23, gauche). L'inverse est aussi possible, où la normale perturbée ω_p peut tourner le dos à la direction incidente ω_i mais pas à la normale géométrique ω_g (figure 2.23, droite).

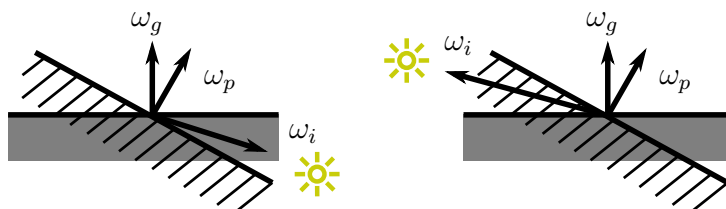


FIGURE 2.23 – L'utilisation de cartes de normales crée des situations paradoxales. La lumière peut être visible depuis la géométrie perturbée par ω_p mais pas par la géométrie macroscopique (gauche). Inversement, la lumière peut être visible par le plan tangent défini par ω_g et pas par celui défini par ω_p (droite).

La géométrie implicite générée par ω_p est appelée ici méso-géométrie, car elle se situe à la frontière entre l'échelle macroscopique (la géométrie) et l'échelle microscopique (la BRDF). Une façon simple et courante de résoudre les problèmes liés à l'utilisation d'une carte de normales est de n'autoriser la réflexion que quand la macro et la méso-géométrie sont visibles. Dans ce cas, la portion de la BRDF qui est sous la surface définie par ω_g est amputée et la BRDF retourne zéro pour des directions pointant vers la zone sous la méso-surface. L'équation de réflexion locale (équation 2.1.8) n'est plus évaluée dans le repère tangent à ω_g mais dans celui de ω_p . Même si cette technique est très largement utilisée, elle ne garantit plus la conservation de l'énergie et la réciprocité du transport de la lumière, comme décrit par Veach [Vea96]. De plus, les situations où les normales tournent le dos à la caméra créent des artéfacts visuels comme des franges noires. Pour

finir, l'échantillonnage des directions sortantes devient hasardeux, car des directions allant dans la macro-géométrie peuvent être échantillonnées. Ces échantillons qui sont invalides d'un point de vue macroscopique ne l'étaient pas d'un point de vue mésoscopique.

2.5.1 Distributions décentrées

Si la BRDF est basée sur les micro-facettes, nous pouvons résoudre en partie les problèmes liés à l'utilisation de cartes de normales en centrant les NDFs autour de la normale perturbée. Ainsi, nous pouvons échantillonner que des micro-normales orientées vers l'extérieur de la macro-surface *et* de la méso-surface. Le décentrage par rapport à ω_g doit se faire au niveau de la distribution des pentes P^{22} , comme ceci (ici pour Beckmann) :

$$P^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi\alpha_x\alpha_y\sqrt{1-c_{xy}^2}} \exp\left(-\frac{1}{1-c_{xy}^2}\left(\frac{(x_{\tilde{m}}-x_{\tilde{p}})^2}{\alpha_x^2} + \frac{(y_{\tilde{m}}-y_{\tilde{p}})^2}{\alpha_y^2} - \frac{2c_{xy}(x_{\tilde{m}}-x_{\tilde{p}})(y_{\tilde{m}}-y_{\tilde{p}})}{\alpha_x\alpha_y}\right)\right), \quad (2.5.1)$$

où la pente associée à la perturbation $\tilde{p} = (x_{\tilde{p}}, y_{\tilde{p}})$ décale la moyenne de la distribution par rapport à la pente de la normale géométrique $\tilde{g} = (0, 0)$ (figure 2.24). Dans la méthode LEADR [DHI⁺13], les auteurs filtrent des cartes de hauteurs en approximant la géométrie sous pixel avec cette distribution, où la normale moyenne est potentiellement différente de la normale géométrique.

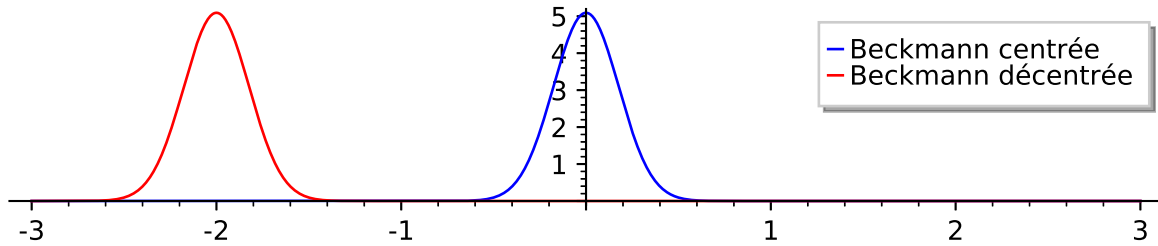


FIGURE 2.24 – Distributions de pentes de Beckmann où la pente moyenne est $x_{\tilde{p}} = 0$ (bleu) et $x_{\tilde{p}} = -2$ (rouge).

Dans cette méthode, ils expliquent comment modifier la BRDF à base de micro-facettes (équation 2.4.1) pour qu'elle reste correctement normalisée, même avec une distribution non-centrée. La BRDF décentrée est

$$f(\omega_o, \omega_i) = \frac{(\omega_p \cdot \omega_g)}{\langle \omega_o, \omega_p \rangle} \frac{F(\omega_o, \omega_h)G_2(\omega_o, \omega_i, \omega_h)D(\omega_h)}{4(\omega_i \cdot \omega_g)}, \quad (2.5.2)$$

où l'aire projetée de la macro-surface $\omega_o \cdot \omega_g$ a été remplacée par l'aire projetée de la méso-surface $\langle \omega_o, \omega_p \rangle / (\omega_p \cdot \omega_g)$. Le terme $(\omega_i \cdot \omega_g)$ au dénominateur reste inchangé car ici nous continuons à utiliser l'équation 2.1.8 pour calculer localement la réflexion lumineuse, et $(\omega_i \cdot \omega_g)$ ne sert qu'à faire disparaître $\cos \theta_i$ (rappel : $(\omega_i \cdot \omega_g) = \cos \theta_i$). Le masquage-ombrage G_2 a une solution avec le profil de micro-surface de Smith. Plus de détails sont disponibles dans LEADR [DHI⁺13]. Le décentrage est théoriquement possible avec la distribution

GGX [RBSM19] ou toute autre NDF invariante de forme mais à notre connaissance, personne n'a encore utilisé (et donc proposé) la formulation décentrée de GGX.

Utiliser l'équation 2.5.2 permet de faire le rendu d'une carte de normales, où en chaque point u de la surface, nous utilisons ω_p pour paramétrer la BRDF et simuler des perturbations géométriques. En comparaison avec un remplacement pur et simple de ω_g par ω_p , il y a des avantages importants. Nous pouvons utiliser la VNDF pour échantillonner la BRDF [Hd14], dans le cas où la distribution est alignée sur les axes (pas de rotation). De plus, l'énergie est conservée pour de faibles rugosités. Cependant, le modèle est non réciproque et il reste indéfini lorsque la méso-surface tourne le dos à la caméra, causant des franges noires sur la surface (figure 2.25). Une implémentation de ce modèle est disponible sur la [page du projet du chapitre 4](#), dans un fichier matériau de `pbrt-v3` nommé `normalmap_classic`. Comme nous n'utilisons que des BRDFs à base de micro-facettes, nous considérons cet algorithme comme étant la méthode classique de rendu de carte de normales.

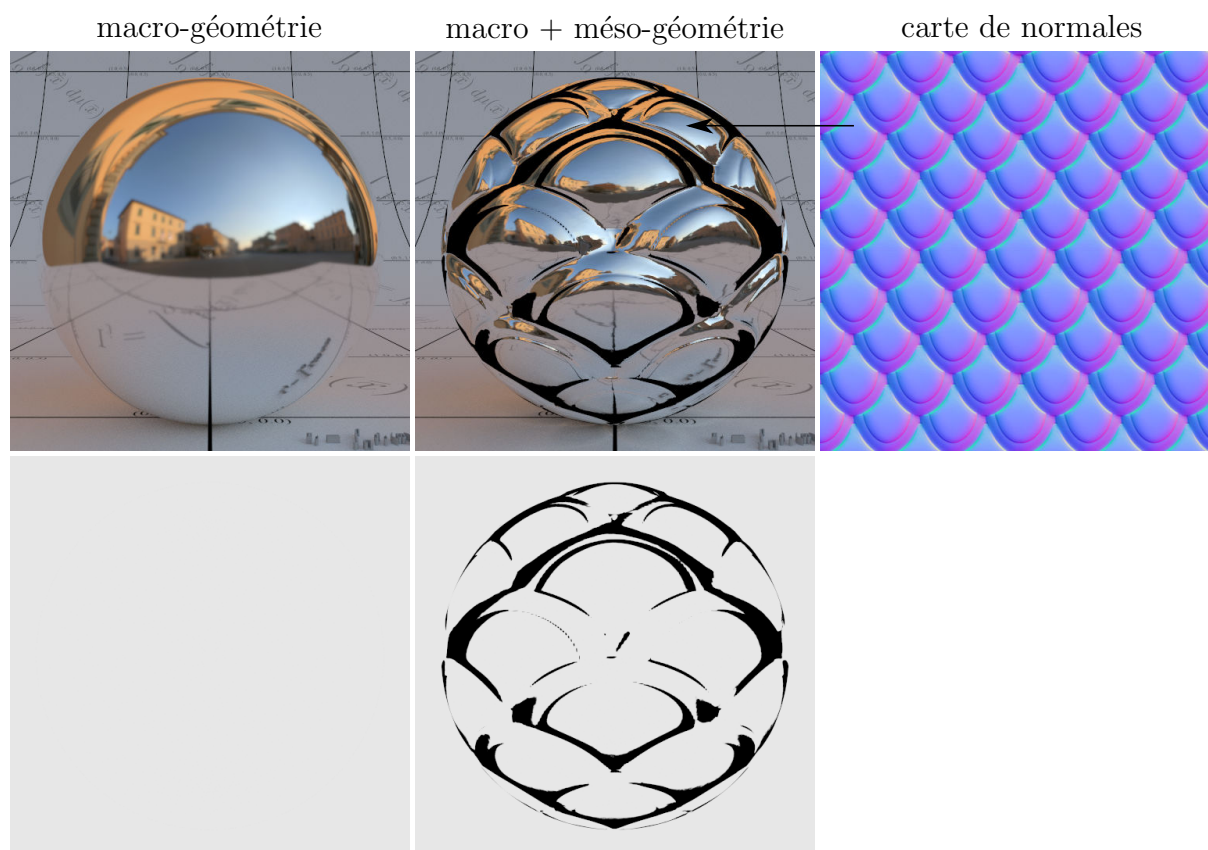


FIGURE 2.25 – Faire des rendus de cartes de normales est possible avec des distributions décentrées, en utilisant les équations 2.5.1 et 2.5.2. Le test du four blanc est presque parfait. Il y a juste des franges noires, qui sont dues aux normales tournant le dos à la caméra.

Nous utilisons une NDF décentrée (équation 2.5.1) dans le chapitre 3. Nous précalculons la variance et la pente moyenne de chaque texel d'une carte de normales dans le but de la représenter par une mixture de NDFs. Cette mixture permet le rendu de matériaux à micro-reflets.

2.5.2 Distributions à deux normales

Pour éviter le problème des franges noires lié au rendu classique de cartes de normales, Schüssler et al. [SHHD17] proposent de donner une interprétation géométrique à la carte de normales et de modéliser du multi-rebonds dedans. Ils nomment leur méthode : *Rendu basé micro-facettes de cartes de normales*, car elle utilise une NDF. Contrairement à la NDF de la section précédente (équation 2.5.1), la moyenne de leur distribution est toujours égale à la normale géométrique ω_g , dans le but d'avoir une BRDF symétrique. Donner une réalité géométrique à la micro-surface est important. Une interprétation possible de la micro-géométrie d'une carte de normales classique est la suivante. Nous pouvons imaginer qu'un point u de la carte de normales est constitué d'une soupe de facettes et que toutes ces facettes ont la même orientation, égale à ω_p (figure 2.26, gauche). Les paramètres α de l'équation 2.5.1 permettent de contrôler la rugosité de ces micro-facettes. Malheureusement, cette configuration crée une situation où la normale perturbée peut tourner le dos à la caméra, créant des franges noires sur la surface. De plus, la micro-surface n'est pas continue (pas de réalité géométrique). À chaque point u de la surface, Schüssler et al. proposent de fermer cette micro-surface en ajoutant des facettes tangentes ω_t aux normales perturbées ω_p (figure 2.26, droite).

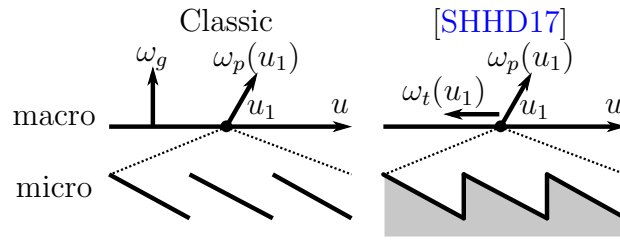


FIGURE 2.26 – Micro-surface en un point d'une carte de normales classique (gauche) et à micro-facettes (droite), selon le modèle de Schüssler et al. [SHHD17]. Cette dernière représentation est plus plausible car elle crée une continuité de la micro-surface et un observateur verra toujours quelque chose (pas de normales tournant le dos).

Dans la figure 2.26, la micro-surface peut être caractérisée par une NDF (si les facettes ne sont pas rugueuses). À gauche, la NDF est composée d'une fonction de Dirac :

$$D(\omega_m) = \frac{\delta(\omega_p - \omega_m)}{\omega_p \cdot \omega_g}, \quad (2.5.3)$$

signifiant que l'ensemble des micro-facettes a la même orientation ω_p . Le dénominateur permet à la NDF de respecter la contrainte de l'équation 2.4.6, stipulant que l'aire projetée des micro-normales sur la macro-surface doit faire un. À droite de la figure 2.26, la distribution de micro-normales est composée de deux fonctions de Dirac :

$$D(\omega_m) = \frac{\delta(\omega_p - \omega_m)}{\omega_p \cdot \omega_g} + \frac{\delta(\omega_t - \omega_m) \sqrt{1 - (\omega_p \cdot \omega_g)^2}}{\omega_p \cdot \omega_g}. \quad (2.5.4)$$

Ici, la NDF est une distribution à seulement deux normales. Si leur moyenne donne la normale géométrique ω_g , comme c'est ici le cas, alors la BRDF est symétrique et il n'y a plus de cas où potentiellement toutes les micro-normales tournent le dos à l'observateur. De plus, cette distribution a une représentation géométrique continue, dans laquelle nous

pouvons simuler du multi-rebonds. Dans le chapitre 4, nous utilisons aussi une distribution à deux normales pour faire le rendu d'une carte de normales, mais la deuxième micro-normale est symétrique à ω_p par rapport à ω_g .

En imaginant que la micro-surface est composée de facettes, nous pouvons considérer que le matériau de chaque facette est arbitraire : ω_p a une micro-BRDF f_p et ω_t a une micro-BRDF f_t . Il en découle une BRDF simple rebond induite par la NDF de l'équation 2.5.4 (la dérivation complète est dans le papier de Schüssler et al. [SHHD17]) :

$$f_1(\omega_o, \omega_i)(\omega_i \cdot \omega_g) = \lambda_p(\omega_o) f_p(\omega_o, \omega_i) \langle \omega_i, \omega_p \rangle G_1(\omega_i, \omega_p) + \lambda_t(\omega_o) f_t(\omega_o, \omega_i) \langle \omega_i, \omega_t \rangle G_1(\omega_i, \omega_t). \quad (2.5.5)$$

Les termes $\lambda_p(\omega_o)$ et $\lambda_t(\omega_o)$ sont les probabilités d'intersection de respectivement la facette ω_p et ω_t depuis la direction d'observation ω_o (figure 2.27). Nous avons

$$\lambda_p(\omega_o) = \frac{a_p(\omega_o)}{a_p(\omega_o) + a_t(\omega_o)} \quad \text{et} \quad \lambda_t(\omega_o) = \frac{a_t(\omega_o)}{a_p(\omega_o) + a_t(\omega_o)}, \quad (2.5.6)$$

où $a_p(\omega_o)$ et $a_t(\omega_o)$ sont les aires projetées de chaque facette. Elles sont définies comme suit :

$$a_p(\omega_o) = \frac{\langle \omega_o, \omega_p \rangle}{(\omega_p \cdot \omega_g)} \quad \text{et} \quad a_t(\omega_o) = \frac{\langle \omega_o, \omega_t \rangle \sqrt{1 - (\omega_p \cdot \omega_g)^2}}{(\omega_p \cdot \omega_g)}. \quad (2.5.7)$$

L'ombrage G_1 de l'équation 2.5.5 est similaire (mais pas identique) à celui d'une V-cavité (équation 2.4.14) :

$$G_1(\omega_i, \omega_m) = H(\omega_i \cdot \omega_m) \min \left(1, \frac{(\omega_i \cdot \omega_g)}{a_p(\omega_i) + a_t(\omega_i)} \right). \quad (2.5.8)$$

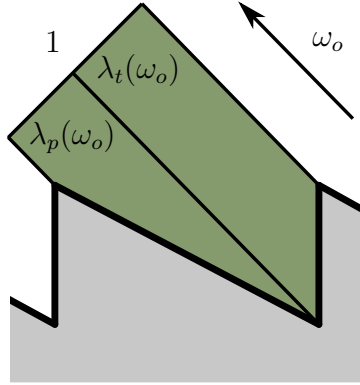


FIGURE 2.27 – Illustration des probabilités d'intersection de chaque micro-facette du modèle de Schüssler et al. [SHHD17].

Pour éliminer complètement les franges noires, il faut modéliser du multi-rebonds dans la micro-surface. Schüssler et al. [SHHD17] utilisent des marches aléatoires, comme dans la méthode de Heitz et al. [HHdD16]. Les apparences obtenues ne sont pas sombres comparées aux résultats obtenues avec un rendu classique de carte de normales, mais le

surcoût de temps de calcul peut être important, surtout pour des matériaux peu rugueux (figure 2.28, droite).

Les méthodes utilisant des marches aléatoires ont recours à une PDF approximative, car la BRDF n'a pas de solution analytique. Sa valeur exacte s'obtient seulement en moyennant plusieurs évaluations de différentes instances de marches aléatoires. La PDF retournée par une BRDF doit être en accord avec sa procédure d'échantillonnage. Les algorithmes utilisant des marches aléatoires pour modéliser le multi-rebonds ont une procédure d'échantillonnage optimale, c'est-à-dire qu'ils échantillonnent directement la BRDF multi-rebonds, et non une sous-partie ou une approximation de cette dernière. Dans ce cas, la PDF à retourner est la valeur de la BRDF sans absorption pondérée par le cosinus. Cette valeur de PDF est utilisée quand nous faisons de l'échantillonnage multiple (MIS), c'est-à-dire que nous échantillons la BRDF et la lumière, puis que nous combinons la contribution des deux avec les PDFs respectives. Utiliser une PDF approximative, comme proposé par Schüssler et al., est possible et correct, mais lorsque la rugosité est faible, le ratio entre la valeur de la BRDF et la PDF approximée peut être très grand et causer l'apparition de pixels blancs. Dans ce cas, il faut utiliser beaucoup d'échantillons pour compenser le pic de radiance, expliquant le surcoût de temps de calcul important de la figure 2.28, droite, par rapport à un rendu classique de carte de normales (à gauche).

Schüssler et al. proposent une solution analytique pour la valeur de la BRDF et de la PDF lorsque le matériau de la facette tangente est un miroir parfait et que un et deux rebonds sont modélisés (figure 2.28, milieu) :

$$\begin{aligned}
 f_2(\omega_o, \omega_i)(\omega_i \cdot \omega_g) = & \lambda_p(\omega_o) f_p(\omega_o, \omega_i, \omega_p) \langle \omega_i, \omega_p \rangle G_1(\omega_i, \omega_p) \\
 & + \lambda_p(\omega_o) f_p(\omega_o, \omega'_i, \omega_p) \langle \omega'_i, \omega_p \rangle (1 - G_1(\omega'_i, \omega_p)) G_1(\omega_i, \omega_t) \\
 & + \lambda_t(\omega_o) f_p(\omega'_o, \omega_i, \omega_p) \langle \omega_i, \omega_p \rangle G_1(\omega_i, \omega_p).
 \end{aligned} \tag{2.5.9}$$

Dans l'équation précédente, les directions ω'_o et ω'_i sont issues de réflexions parfaites par rapport à la micro-facette tangente : $\omega' = \text{reflect}(\omega, \omega_t)$. Cette forme close permet de ne pas avoir les problèmes liés aux marches aléatoires et entraîne qu'un faible surcoût en temps de calcul par rapport à un rendu classique de carte de normales (11 % pour la scène de la figure 2.28). De plus, elle atténue grandement les franges noires, même si elle ne les élimine pas complètement.

Rajouter une micro-normale pour fermer la micro-surface a un impact sur le rendu, qui peut être perçu comme un artéfact visuel. Il y a des différences d'apparence entre le rendu classique et les rendus de Schüssler et al., *en dehors des franges noires*. Dans le modèle classique, il n'y a qu'une direction de réflexion possible, alors que dans le modèle de Schüssler, il y en a trois. Pour s'en convaincre, il suffit de regarder l'équation 2.5.9, où trois valeurs de BRDFs sont pondérées par de l'ombrage G_1 et des probabilités d'intersection λ . Parfois, les deux facettes sont visibles, comme dans le cas de la figure 2.27. Rappelons que l'utilisation d'une carte de normales pour faire du rendu n'est pas sans risque, car nous modifions de manière arbitraire le comportement de la lumière en modifiant les normales d'un objet, pouvant créer des situations paradoxales (figure 2.23). Dans tous les cas, accepter la présence de ces mélanges de réflexion permet de supprimer les franges noires. C'est donc à l'utilisateur de choisir quel artéfact est le moins gênant.

Nous avons vu dans cette section que le rendu de cartes de normales peut se faire avec des modèles à base de micro-facettes, soit avec des distributions décentrées (section 2.5.1), soit avec des distributions à deux normales (section 2.5.2). Nous utilisons ce premier type

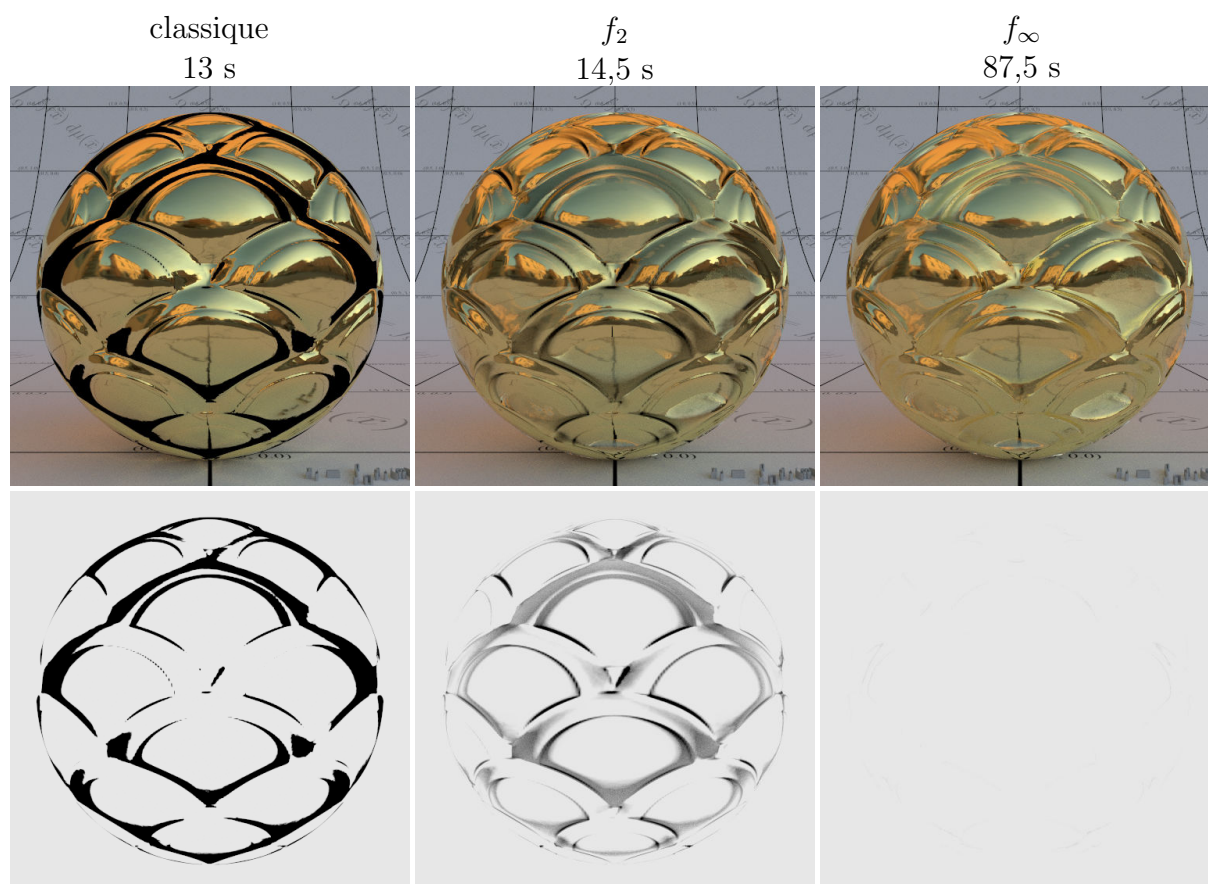


FIGURE 2.28 – Rendus d’une carte de normales avec la méthode classique (gauche) et la méthode de Schüssler et al. avec la forme close à deux rebonds (milieu) et avec des rebonds infinis utilisant des marches aléatoires coûteuses (droite). La BRDF à deux rebonds donne une apparence plus réaliste que la BRDF classique, pour un surcoût de temps de calcul négligeable (11 %).

de NDF dans le chapitre 3 et le deuxième dans le chapitre 4, car le rendu de matériaux à micro-reflets est étroitement lié au rendu de carte de normales, où des micro-miroirs discrets ont une normale différente de la normale géométrique.

2.6 Modèles de BRDFs surfaciques pour le rendu de micro-reflets

Les modèles de BRDFs précédents ont tous un point en commun : ils sont définis pour un point donné de la surface. Nous avons vu que baser une BRDF sur une distribution de normales permet d’avoir des formes closes paramétrables et rapides à évaluer. Nous avons aussi vu que ces NDFs ont plusieurs représentations géométriques possibles, en fonction du choix du profil de la micro-surface (Smith ou V-cavité, par exemple). En fait, en utilisant une NDF, nous supposons forcément que la distribution est liée à une aire. Pour être parfaitement rigoureux, il faudrait parler de distribution de normales *par mètre carré*, où à chaque fois nous multiplions implicitement par un. Dans les sections précédentes, nous avons toujours choisi les NDFs de sorte que la BRDF soit facilement normalisable et

paramétrable. Mais nous pouvons inverser le raisonnement. Nous pouvons construire une NDF à partir d'une surface explicite (c'est-à-dire non statistique), et l'utiliser pour calculer la réflectance. Nous avons dit dans l'introduction qu'il était souhaitable de toujours considérer ce qui était sous pixel de manière statistique, avec une BRDF. Nous pouvons appliquer ce principe pour définir une NDF unique pour chaque empreinte de pixel sur une surface. Par exemple, si les normales d'une surface sont définies par une *carte de normales*, nous pouvons assez facilement reconstituer la NDF pour un patch donné. Nous appelons cette distribution la \mathcal{P} -NDF, car elle est définie pour un patch \mathcal{P} . Cette \mathcal{P} -NDF peut être utilisée, non plus dans une BRDF définie pour un point u sur la surface, mais dans une \mathcal{P} -BRDF, qui sera définie pour un patch \mathcal{P} . Ce patch sera l'empreinte du pixel, ou l'empreinte du rayon si nous utilisons plusieurs rayons par pixel.

2.6.1 Empreinte de rayon

Avant de pouvoir dériver des \mathcal{P} -NDFs depuis une carte de normales, il faut définir formellement ce qu'est une empreinte de rayon sur une surface. Il existe beaucoup de méthodes différentes pour les calculer, comme par exemple le tracé de cônes (cone-tracing) [Ama84] ou le tracé de faisceaux (beam-tracing) [HH84]. Mais la plus utilisée est la méthode utilisant des rayons différentiels (ray differentials) de Igehy [Ige99]. Elle consiste à lancer deux rayons supplémentaires en plus du rayon principal : le premier est décalé horizontalement d'un échantillon de pixel et le deuxième est décalé verticalement. En supposant que la surface sous rayon est plane localement, nous pouvons calculer les points d'intersection de ces trois rayons et estimer les dérivées partielles des coordonnées de surface (u, v) par rapport à x et y , les coordonnées en espace image. Ces dérivées partielles $\partial(u, v)/\partial x$ et $\partial(u, v)/\partial y$ permettent de connaître le taux de changement de (u, v) lorsque nous passons d'un rayon à un autre. À partir de ces deux vecteurs 2D, nous pouvons construire une ellipse, un parallélogramme [JHY⁺14] ou une gaussienne [Hec89], approximant ainsi l'*empreinte du rayon* (figure 2.29).

Si nous prenons deux empreintes de rayon adjacentes, elles se chevauchent. Il vaut donc mieux pondérer chaque point de la surface en fonction de sa distance avec le point d'intersection du rayon. Cette pondération peut se faire en utilisant un filtre gaussien $k_{\mathcal{P}}$ qui représentera l'empreinte du rayon. Calculer les écarts types σ_u et σ_v ainsi que le coefficient de corrélation c_{uv} de l'empreinte $k_{\mathcal{P}}$ se fait en utilisant directement les dérivées partielles [Hec89] :

$$\sigma_u = \frac{\sqrt{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2}}{\sqrt{\chi^2(1-d)}} \quad , \quad \sigma_v = \frac{\sqrt{\left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2}}{\sqrt{\chi^2(1-d)}} \quad \text{et} \quad c_{uv} = \frac{\frac{\partial u}{\partial x} \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial v}{\partial y}}{\chi^2(1-d)\sigma_u\sigma_v} \quad (2.6.1)$$

où $\chi^2(1-d)$ est le quantile de la loi du χ^2 pour un degré de liberté de 2 et une valeur $1-d$. Le terme d permet de contrôler la proportion de densité de la gaussienne contenue dans l'ellipse formée par les dérivées partielles. Dans le cas où nous voulons 99 % de l'empreinte du rayon $k_{\mathcal{P}}$ à l'intérieur de l'ellipse (figure 2.29), nous avons $\sqrt{\chi^2(0,01)} \approx 3$ et nous pouvons considérer que n'importe quelle quantité liée à ce rayon (couleur, réflectance) a une valeur nulle à l'extérieur de l'ellipse (règle des 3σ). La PDF d'une loi normale bivariée nous donne la forme de notre empreinte, où $p = (u_p, v_p)$ est le point d'intersection du

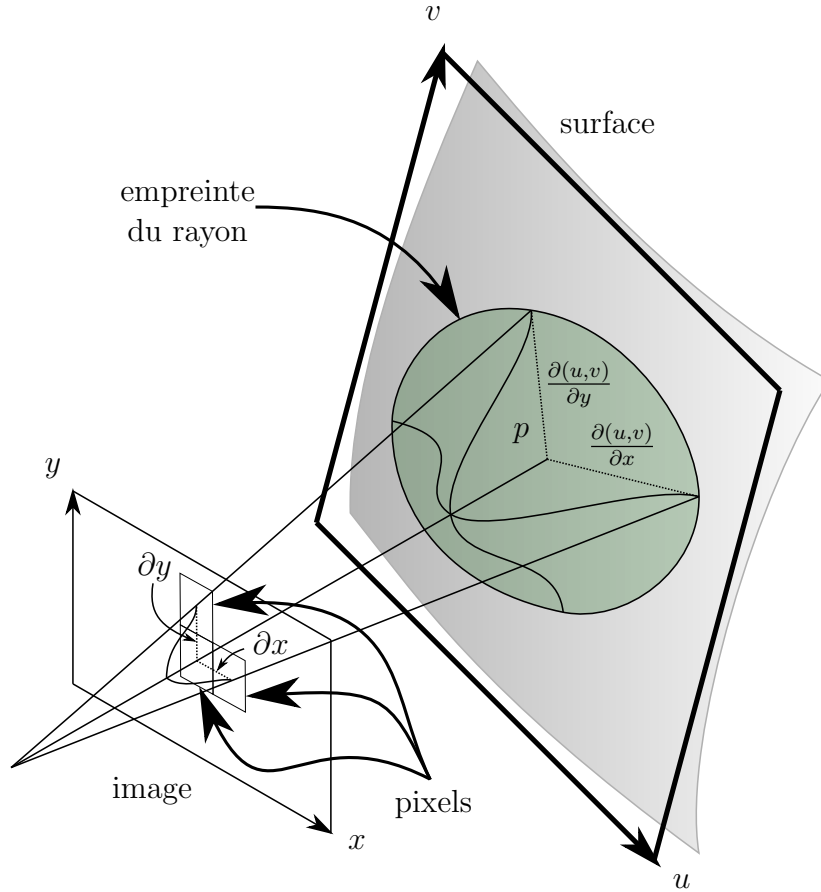


FIGURE 2.29 – Construction d’une empreinte de rayon (ici c’est même une empreinte de pixel) à partir de rayons différentiels.

rayon principal sur la surface :

$$k_{\mathcal{P}}(u, v) = \frac{1}{2\pi\sigma_u\sigma_v\sqrt{1-c_{uv}^2}} \exp\left(-\frac{1}{2(1-c_{uv}^2)}\left(\frac{(u-u_p)^2}{\sigma_u^2} + \frac{(v-v_p)^2}{\sigma_v^2} - \frac{2c_{uv}(u-u_p)(v-v_p)}{\sigma_u\sigma_v}\right)\right). \quad (2.6.2)$$

Le filtre $k_{\mathcal{P}}(u, v)$ représentant l’empreinte du rayon peut avoir d’autres formes, comme par exemple une boîte isotrope ou anisotrope. Dans cette thèse, nous utilisons la forme gaussienne anisotrope et non-alignée sur les axes (équation 2.6.2) car c’est la forme qui approxime le mieux l’empreinte réelle. De plus, cette formulation permet d’avoir des formes closes quand nous la convoluons avec des filtres locaux gaussiens.

2.6.2 Distribution de normales pour une empreinte (\mathcal{P} -NDF)

Définir une \mathcal{P} -NDF (ou même une NDF, tout simplement) se fait en utilisant un Dirac, qui permet de passer du domaine spatial (les normales) au domaine statistique (la densité de normales). La fonction de Dirac permet de « compter le nombre » de micro-normales

ω_m dans notre empreinte de rayon $k_{\mathcal{P}}$:

$$D_{\mathcal{P}}(\omega_m) = \int_{\mathbb{R}} \int_{\mathbb{R}} \delta(\omega_n(u, v) - \omega_m) k_{\mathcal{P}}(u, v) du dv. \quad (2.6.3)$$

Ici, nous utilisons $\omega_n(u, v)$ pour désigner la normale de la carte de normales au point (u, v) sur la surface. Si la carte de normales a été générée en suivant des statistiques gaussiennes (avec une rugosité α de Beckmann, par exemple [Hei15]) et que l'empreinte est très grande, alors la \mathcal{P} -NDF est une distribution de Beckmann (équation 2.4.7). En revanche, si l'empreinte a une taille ne permettant pas d'appliquer la loi des grands nombres, alors la \mathcal{P} -NDF a une forme non lisse et complexe (figure 2.30), comme mentionné dans Yan et al. [YHJ+14].

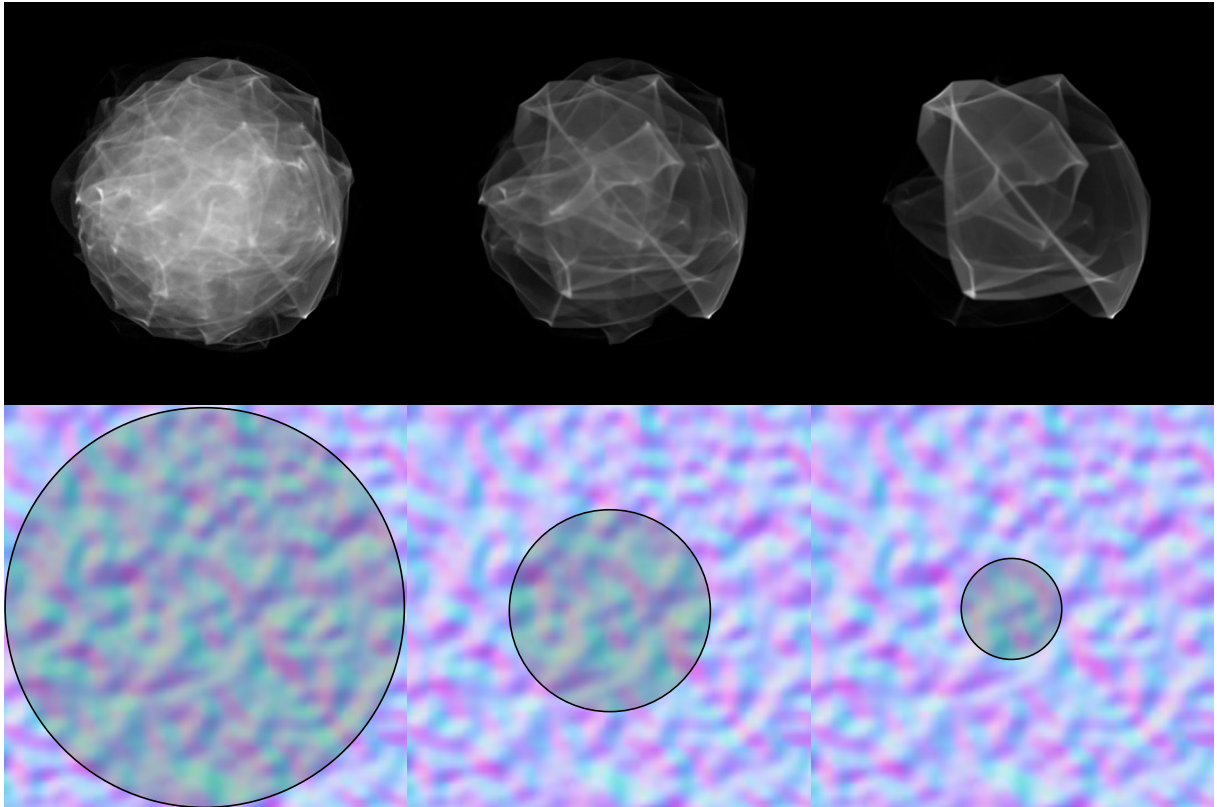


FIGURE 2.30 – Visualisation de différentes P-NDFs (haut), pour une carte de normales gaussienne ($\alpha = 0, 5$) et différentes empreintes de rayon (cercles transparents).

Calculer une \mathcal{P} -NDF à partir d'une empreinte sur une carte de normales se fait en classant les normales en fonction de leur direction (binning). Pour cela, il faut échantillonner plusieurs millions d'échantillons de positions dans l'empreinte $k_{\mathcal{P}}$ (équation 2.6.2), regarder les normales correspondantes $\omega_n(u, v)$ et les classer par direction dans une table. La table peut être en 2D : par exemple, dans la figure 2.30 en haut, les normales sont classées en fonction de leur composante $(x_n, y_n) \in [-1, 1]^2$. Plus le niveau de gris est élevé, plus la normale correspondante $(x_n, y_n, \sqrt{1 - x_n^2 - y_n^2})$ est présente dans l'empreinte du rayon.

Même si utiliser une \mathcal{P} -NDF à la place d'une NDF paramétrique semble plus naturelle pour faire le rendu sous pixel d'une carte de normales, ce n'est pas une pratique très

courante en informatique graphique. La raison est simple : les méthodes les utilisant sont coûteuses en temps de calcul et en mémoire, et des méthodes comme LEADR [DHI+13] approximent très rapidement une \mathcal{P} -NDF exacte par une distribution de Beckmann, même si l'approximation est très grossière. De plus, utiliser une \mathcal{P} -NDF dans une BRDF n'est pas trivial. Dans la section 2.4 sur les BRDFs à micro-facettes, nous avons vu que peu de NDFs permettaient d'être facilement utilisables au sein d'une BRDF. Seules les distributions lisses de Beckmann et GGX ont un terme de masquage relativement simple et ont une VNDF qui s'échantillonne de manière efficace. Si la distribution de normales est arbitraire, comme c'est le cas avec des \mathcal{P} -NDFs, alors il faut soit calculer numériquement son terme de masquage (pour chaque empreinte), soit l'approximer par une formulation plus simple. C'est la deuxième solution qui est utilisée en pratique [YHJ+14, JHY+14, YHMR16], où le terme de masquage-ombrage choisi est celui d'une distribution de Beckmann ayant comme rugosité α la rugosité globale de la carte de normales. Dans ce cas, la \mathcal{P} -BRDF a une formulation assez simple :

$$f_{\mathcal{P}}(\omega_o, \omega_i) = \frac{F(\omega_o, \omega_h)G_2(\omega_o, \omega_i, \omega_h)D_{\mathcal{P}}(\omega_h)}{4(\omega_o \cdot \omega_g)(\omega_i \cdot \omega_g)}. \quad (2.6.4)$$

Cette BRDF surfacique est similaire à l'équation 2.4.1 et rentre dans le cadre théorique des micro-facettes. Le seul changement étant la définition surfacique de la distribution de normales $D_{\mathcal{P}}$ et l'approximation du masquage-ombrage G_2 . Cette BRDF est principalement utilisée pour le rendu de matériaux à micro-reflets. Dans le chapitre 3, nous proposons une formulation où le masquage-ombrage n'est pas approximé, donnant des résultats plus proches d'un rendu de référence où la carte de normales est échantillonnée massivement.

2.6.3 Matériaux à micro-reflets basés cartes de normales

Yan et al. [YHJ+14] proposent d'utiliser l'équation 2.6.4 pour faire le rendu de matériaux à micro-reflets, comme des surfaces pailletées, de la peinture métallisée ou du métal brossé ou rayé. Pour calculer efficacement la \mathcal{P} -NDF associée à une empreinte sur une carte de normales, ils triangulent la carte de normales et interpolent linéairement les normales entre chaque sommet. Ils choisissent de diviser chaque texel en deux triangles, si le contenu de la carte de normales n'est pas trop haute fréquence. Sinon, ils subdivisent chaque texel en 4×4 sous-texels en utilisant une interpolation de Catmull-Rom, donnant finalement 32 triangles par texel. La carte de normales est considérée comme presque parfaitement spéculaire, car le but est de représenter des matériaux brillants. Considérer que la carte de normales est un miroir parfait crée des situations où la \mathcal{P} -NDF est un Dirac (par exemple, dans des zones plates), ce qui empêche toute évaluation. Donc, à chaque point de la surface, la micro-géométrie est modélisée par une distribution de normales gaussienne $D(\omega_m, \omega_n(u, v))$ centrée sur $\omega_n(u, v)$ et ayant une microrugosité très faible. Cette modélisation a pour but de mollifier le Dirac de l'équation 2.6.3 en une distribution gaussienne avec un très faible écart type :

$$D_{\mathcal{P}}(\omega_m) = \int_{\mathbb{R}} \int_{\mathbb{R}} D(\omega_m, \omega_n(u, v))k_{\mathcal{P}}(u, v)dudv. \quad (2.6.5)$$

La discrétisation de la carte de normales en triangles permet de transformer l'intégrale sur l'espace de la surface en une somme d'intégrales sur les N triangles :

$$D_{\mathcal{P}}(\omega_m) \approx \sum_{i=1}^N \int_{\Delta} D(\omega_m, \omega_n(u, v)) k_{\mathcal{P}}(u, v) du dv. \quad (2.6.6)$$

Yan et al. proposent une méthode de calcul efficace de l'intégrale précédente, mais le parcours de tous les triangles reste nécessaire et est coûteux. Il est cependant facile de hiérarchiser chaque triangle à l'aide d'un BVH 4D (Bounding Volume Hierarchy) en fonction du minimum et du maximum de ses positions (u, v) et de ses directions ω_n . C'est ce que font Yan et al., sachant qu'en plus chaque triangle occupe une boîte englobante 4D assez petite, rendant très efficace ce type de structure accélératrice. Lors de l'évaluation de la \mathcal{P} -BRDF (équation 2.6.4), la \mathcal{P} -NDF (équation 2.6.6) est évaluée avec le vecteur à mi-chemin ω_h . Nous demandons donc au BVH de retourner tous les triangles qui ont un intervalle de normales comprenant ω_h . Pour ce qui est de l'échantillonnage, ils proposent d'échantillonner l'empreinte du pixel, de choisir la normale à la position échantillonnée, puis de s'en servir pour refléter la direction d'observation ω_o . Cette méthode est comparable à un échantillonnage classique de NDF.

Deux ans après, Yan et al. [YHMR16] proposent une méthode pour accélérer par cent environ l'évaluation de la \mathcal{P} -NDF (équation 2.6.6). Pour cela, ils choisissent de ne plus discrétiser la carte de normales en N triangles, mais de la représenter avec une mixture de N gaussiennes 4D. Ces gaussiennes 4D sont des distributions de normales 2D *et de positions* 2D. Chacune de ces distributions 4D approxime la distribution réelle de normales liée à un sous-espace de la carte de normales (typiquement un texel). Elles sont dérivées en utilisant l'espacement h entre chaque gaussienne D_i , distribuées uniformément dans l'espace de la texture. Ils utilisent aussi la jacobienne de la carte de normale, à l'emplacement (u_i, v_i) de chaque gaussienne 4D centrée sur $(u_i, v_i, x_n(u_i, v_i), y_n(u_i, v_i))$. Ce changement de représentation permet d'avoir une \mathcal{P} -NDF

$$D_{\mathcal{P}}(\omega_m) \approx \sum_{i=1}^N \int_{\mathbb{R}} \int_{\mathbb{R}} D_i(\omega_m, \omega_n(u_i, v_i)) k_{\mathcal{P}}(u, v) du dv \quad (2.6.7)$$

impliquant une somme d'intégrales ayant une solution analytique relativement simple, car l'intégrale sur un domaine infini du produit de deux gaussiennes a une forme close. Bien sûr, cette formulation suppose un filtre $k_{\mathcal{P}}$ gaussien. Comme pour la méthode précédente, ils utilisent un BVH pour accélérer l'évaluation. Mais cette fois, ce sont des gaussiennes 4D qui sont hiérarchisées, où leur contribution est considérée comme nulle après 3σ .

Le nombre de gaussiennes 4D nécessaire pour retranscrire correctement la \mathcal{P} -NDF exacte dépend du contenu fréquentiel de la carte de normales. En pratique, il vaut mieux utiliser une gaussienne par texel puis adapter le contenu de la carte de normales pour que l'approximation soit « acceptable ». La figure 2.31 compare les \mathcal{P} -NDFs exactes de la figure 2.30 avec celles obtenues en utilisant la méthode de Yan et al. [YHMR16].

Comme dit au début de la section, utiliser des \mathcal{P} -NDFs est utile pour faire le rendu de matériaux à micro-reflets. Ces matériaux ont des BRDFs qui varient spatialement et qui ont des hautes fréquences, c'est-à-dire qu'elles n'ont pas une forme lisse, à l'inverse d'une BRDF basée sur une distribution de Beckmann, par exemple. La forme d'une BRDF à micro-facettes étant directement liée à sa NDF implique qu'utiliser des NDFs hautes

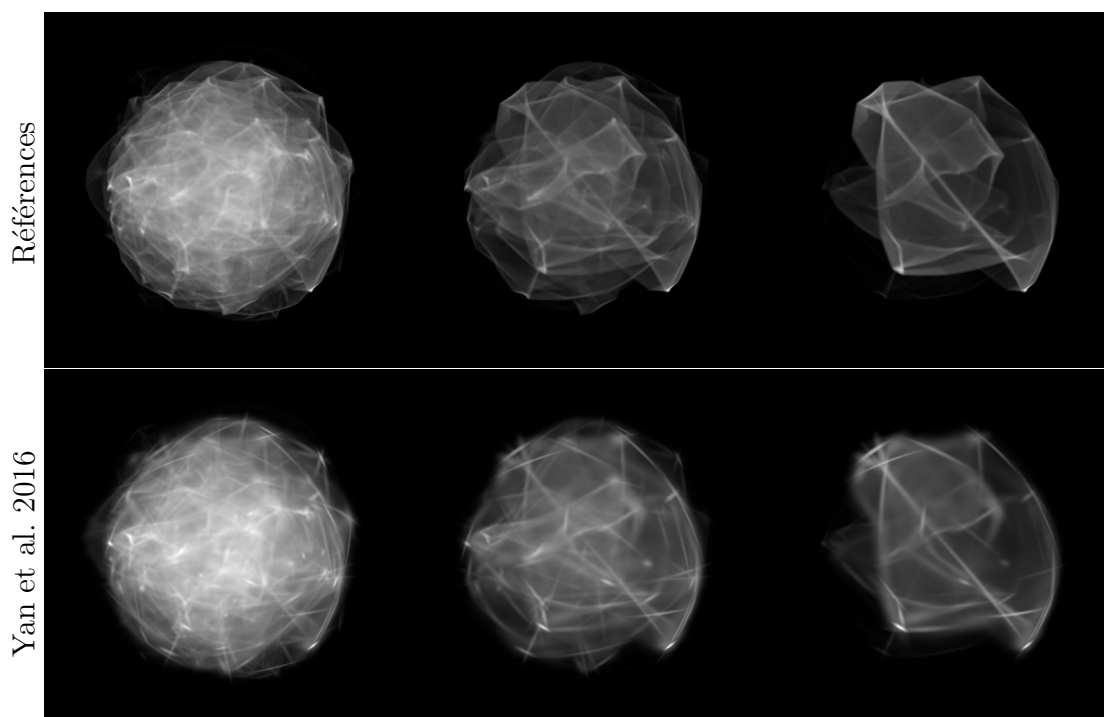


FIGURE 2.31 – Comparaisons entre des P-NDFs exactes et leur approximation obtenue avec la méthode de Yan et al. 2016. Dans les approximations, nous pouvons distinguer chaque gaussienne individuellement. Il y en a une par texel.

fréquences et variant spatialement permet de reproduire l'apparence de matériaux à micro-reflets. Les \mathcal{P} -NDFs obtenues à partir d'empreintes de rayon sur des cartes de normales spéculaires ont ces deux propriétés. Des exemples de matériaux sont présentés sur la figure 2.32, où différentes cartes de normales sont utilisées. Nous utilisons des cartes d'assez haute résolution (2048×2048 texels) et répétables, car elles doivent être répétées plusieurs fois sur la surface (15 répétitions sur la sphère de la figure 2.32). Pour éviter les artefacts de répétition et avoir une empreinte mémoire moins importante, une méthode très récente utilise de la synthèse de texture par l'exemple [WHHY19].

Les performances sont très variables avec la méthode de Yan et al. 2016. La distribution des directions des normales doit être la plus uniforme possible pour optimiser le BVH. Par exemple, le rendu de rayures à gauche de la figure 2.32 est long à obtenir car la carte de normales a beaucoup de zones plates. Beaucoup de gaussiennes 4D seront identiques et occuperont la même zone dans le BVH. Lors de l'échantillonnage de la \mathcal{P} -BRDF, la probabilité d'échantillonner une normale des zones plates est grande. Ensuite, lors de l'évaluation, il y a de fortes chances de demander au BVH de retourner toutes les gaussiennes 4D de l'empreinte du pixel ayant pour direction principale la normale géométrique : il y en a beaucoup dans ce cas. Une autre situation peut gravement atténuer les performances : c'est quand les jacobiniennes ont des valeurs fortes, c'est-à-dire quand les variations entre les texels sont importantes. En effet, des grandes variations de normales créent des gaussiennes 4D ayant un écart type spatial et directionnel très grand. Par conséquent, ces gaussiennes auront des boîtes englobantes qui seront aussi très grandes, augmentant leur probabilité d'être retournées par la structure accélératrice. Plus il y a de gaussiennes 4D retournées par le BVH, plus il y a de termes à sommer lors de l'évaluation

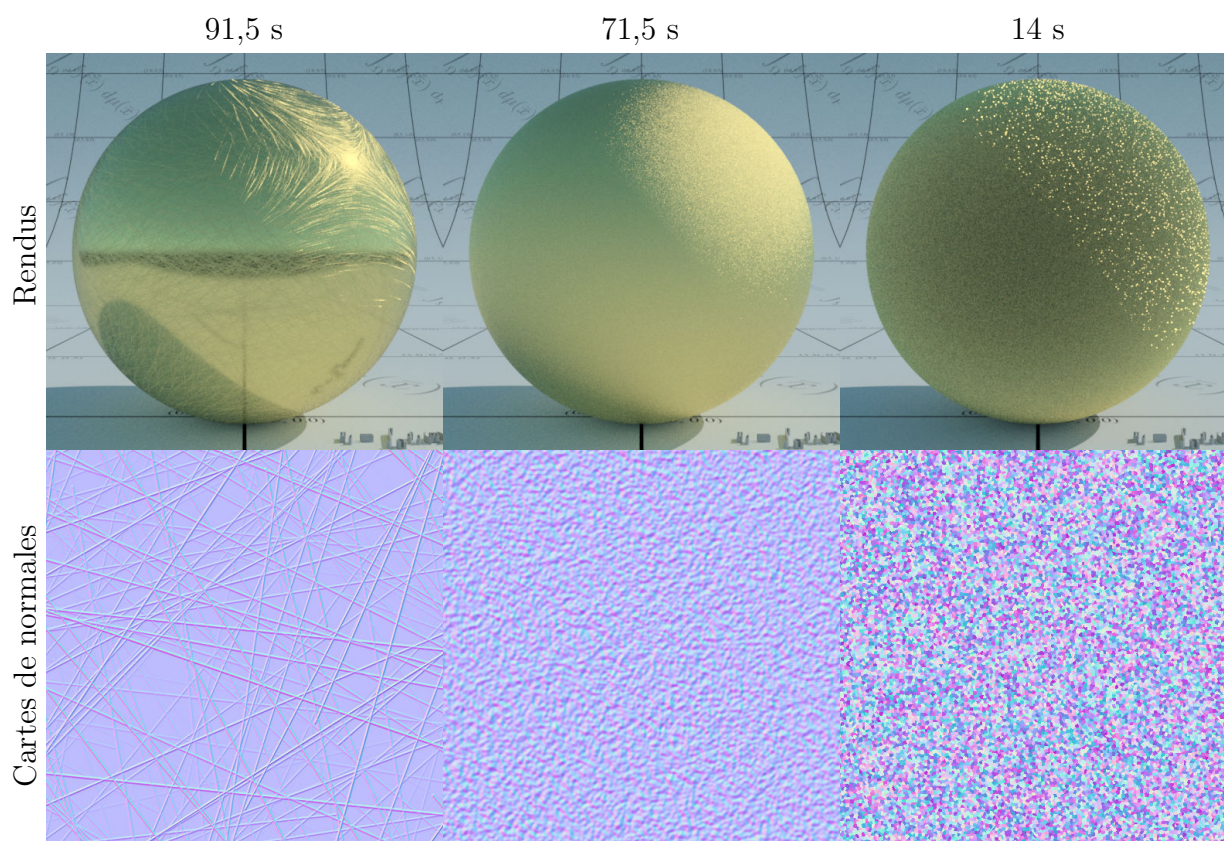


FIGURE 2.32 – Rendus utilisant 256 échantillons par pixel (epp) avec la méthode de Yan et al. 2016 (haut), où la micro-géométrie est représentée par des cartes de normales spéculaires haute définition (bas). Le rendu à droite est plus rapide car des gaussiennes fines sont utilisées (variante « flat element » de la méthode de Yan et al. 2016).

de la \mathcal{P} -NDF, et plus le rendu est long. C'est le cas des rendus à gauche et au milieu de la figure 2.32. Le rendu de droite n'a pas ce problème, car l'écart type spatial et directionnel de chaque gaussienne 4D est fixé à une petite valeur. Yan et al. proposent une variante où les gaussiennes 4D ne sont pas étirées (incurvées, « curved element » dans le papier), c'est-à-dire qu'ils supposent une variation constante et faible entre texels. Cette supposition est acceptable dans le cas où nous voulons faire le rendu de matériaux à micro-paillettes, basé par exemple sur une carte de normales possédant un pavage de Voronoï (figure 2.32, droite). Chaque cellule de Voronoï est un micro-miroir, et il ne faut pas que la transition entre deux micro-miroirs adjacents soit lisse et continue, contrairement à des rayures, par exemple. Utiliser des éléments plats améliore grandement les performances et permet à la méthode de Yan et al. d'être compétitive face à un rendu utilisant une BRDF à micro-facettes classique. Par exemple, il faut 13,5 secondes pour faire le rendu de la scène de la figure 2.32 avec une NDF lisse (Beckmann), contre 14 secondes avec la méthode de Yan et al. permettant de faire des paillettes (gaussiennes 4D plates).

Même si dans certaines situations, la méthode de Yan et al. peut être très performante, elle nécessite néanmoins une grande quantité de mémoire. Une gaussienne 4D incurvée est représentée par une matrice de covariance 4×4 symétrique, utilisant dix nombres à décimales (float). En utilisant des textures faisant 2048×2048 texels, avec une gaussienne 4D par texel, nous devons stocker une mixture de gaussiennes faisant 160 MiB, une carte

de normales faisant 32 MiB (2 floats par texel) et un BVH faisant 512 MiB. L’empreinte mémoire est donc au total de 704 MiB pour *un* matériau à micro-reflets. Si nous utilisons des gaussiennes à écart type fixe, il n’y a pas besoin de stocker la mixture de gaussiennes. Juste pour rappel, une BRDF à base de micro-facettes classique n’a besoin que de deux paramètres de rugosité α_x et α_y .

Yan et al. ont proposé une extension de leur travail pour le rendu spectral [YHW⁺18]. Les BRDFs précédentes se basent sur de l’optique géométrique, qui est une approximation du modèle ondulatoire de la lumière (plus précis). Si la carte de hauteurs est de l’ordre du micron, alors les réflexions lumineuses deviennent très colorées à cause d’un phénomène de diffraction. C’est le cas pour des objets de la vie quotidienne, comme du métal finement rugueux ou rayé. Dans cette thèse, nous ne nous intéressons pas au rendu spectral, et nous nous concentrons uniquement sur des BRDFs supposant un modèle linéaire de la lumière.

Pour finir, Gamboa et al. [GGN18] proposent d’intégrer toute la radiance incidente d’un seul coup, au lieu d’utiliser l’échantillonnage multiple (MIS). Pour cela, ils approximent l’équation de réflexion locale (équation 2.1.8) et utilisent une texture d’histogrammes couplée avec des harmoniques sphériques, assez coûteuses en mémoire.

2.6.4 Matériaux à micro-reflets non basés cartes de normales

\mathcal{P} -NDFs procédurales. Des méthodes non basées sur des cartes de normales utilisent une \mathcal{P} -BRDF (équation 2.6.4) basée sur une \mathcal{P} -NDF qui est « procédurale ». Jakob et al. proposent d’utiliser une distribution de normales discrète pour simuler l’apparence de matériaux à paillettes [JHY⁺14]. La méthode suppose que dans une unité de surface, il existe N micro-facettes parfaitement spéculaires, N étant très grand (par exemple, $N = 10^9$). Elles ne sont pas explicitement stockées en mémoire. Calculer la \mathcal{P} -NDF revient à compter le nombre de micro-facettes tombant dans l’empreinte du pixel, et correctement orientées pour refléter la lumière. Les facettes sont générées aléatoirement et à la volée, à travers le parcours d’un arbre (quadtree), en commençant à la racine. La cohérence spatiale des paillettes étant primordiale, le processus est initialisé de manière déterministe, pour que deux évaluations avec des empreintes identiques retournent toujours le même résultat. Pour connaître le nombre de facettes dans les quatre nœuds fils de la racine, ils échantillonnent une distribution multinomiale avec un vecteur de probabilité $p = (1/4, 1/4, 1/4, 1/4)$ et N essais, car ils supposent une distribution uniforme des facettes sur la surface. Le processus est appliqué récursivement.

Le comptage du nombre de facettes ayant pour direction ω_h pour une empreinte donnée est plus compliqué, car la distribution de normales n’est pas uniforme mais plutôt gaussienne. Ils proposent d’utiliser des pré-calculs permettant de garantir que les directions de toutes les facettes dans les feuilles de l’arbre suivent la distribution de normales choisie par l’utilisateur. L’échantillonnage par importance de la \mathcal{P} -BRDF est fait en échantillonnant cette distribution lisse de normales, et non pas la NDF discrète, ce qui n’est pas très optimal. Pour éviter l’utilisation de pré-calculs, Atanasov et Koylazov [AK16] proposent de parcourir toutes les facettes contenues dans l’empreinte de rayon. Ce parcours leur permet de calculer la VNDF discrète et améliore donc l’échantillonnage. Zirr et Kaplayan [ZK16] utilisent aussi une \mathcal{P} -NDF procédurale, mais dans un contexte temps-réel.

2.7 Objectif formel de la thèse

Le problème avec les méthodes se basant sur l'équation 2.6.4 est qu'elles supposent toutes que la normale moyenne dans l'empreinte du rayon est égale à la normale géométrique, ce qui n'est pas toujours vrai. Cela rend la normalisation encore plus approximative (elle l'était déjà un peu car le masquage n'a pas de solution analytique avec les \mathcal{P} -NDFs précédentes). De plus, ces méthodes utilisant des \mathcal{P} -BRDFs ne modélisent qu'un rebond de la lumière dans la micro-surface, donnant des apparences trop sombres quand la micro-géométrie est très rugueuse. Pour finir, l'échantillonnage n'est pas optimal, car il ne prend pas en compte la direction d'observation. Seule la méthode de Raymond et al. [RGB16], spécialisée dans le rendu de rayures, a un échantillonnage optimal, est multi-rebonds et est correctement normalisée. L'objectif formel de cette thèse est de résoudre ces problèmes en deux étapes, dans le cas où la \mathcal{P} -BRDF est basée sur une carte de normales (cette représentation permet de modéliser plus de matériaux qu'avec des \mathcal{P} -BRDFs procédurales).

La première étape consista à trouver une meilleure représentation d'une carte de normales que celle proposée par Yan et al. 2016 [YHMR16], permettant la dérivation d'un terme de masquage analytique. Dans leur travail, ils utilisent une mixture de gaussiennes 4D qui permet de reproduire avec beaucoup de précision la complexité des cartes de normales. Malheureusement, la \mathcal{P} -NDF est complexe et empêche la dérivation d'un terme de masquage analytique. Dans le chapitre 3, nous proposons une nouvelle mixture, basée sur des distributions de Beckmann décentrées (équation 2.5.1). Cette formulation donne un masquage avec une forme close, mais impliquant le parcours de toutes les normales discrètes. Nous proposons une approximation, qui utilise la rugosité *dans l'empreinte de rayon*, et non pas la rugosité globale de la carte de normales, comme le font les méthodes précédentes. Cette approximation nous a permis d'avoir des résultats correspondant avec ceux de la formulation exacte, tout en gardant des performances proches de celles de Yan et al.

La deuxième étape de la thèse a été de trouver une formulation multi-rebonds d'une \mathcal{P} -BRDF basée carte de normales, permettant la conservation totale de l'énergie et possédant un échantillonnage optimal. Nous proposons cette formulation dans le chapitre 4. Elle permet de faire le rendu basé physique d'une grande catégorie de matériaux à micro-reflets. Notre méthode ne se base plus sur une \mathcal{P} -NDF, car nous avons remarqué que cette représentation entraînait beaucoup de problèmes (par exemple, le calcul d'un masquage nécessitant la connaissance de toute la micro-géométrie dans l'empreinte du rayon pour correctement normaliser la \mathcal{P} -BRDF, un peu comme pour le profil de micro-surface de Smith). Notre méthode s'inspire plus du modèle à base de V-cavités, où le masquage se calcule localement et se moyenne. De plus, nous avons remarqué que l'utilisation d'une carte de normales posait des problèmes de normales tournant le dos à l'observateur, créant des situations où localement, la BRDF est indéfinie. Pour éviter ce problème, nous avons changé la représentation de la carte de normales, comme l'ont fait Schüssler et al. [SHHD17], mais en utilisant des normales symétriques, et non pas des normales tangentes. Pour finir, nous avons utilisé des techniques de compensation d'énergie pour modéliser le multi-rebonds.

Chapitre 3 :

Rendu efficace et précis de cartes de normales spéculaires et haute définition

3.1 Introduction

Les matériaux qui nous entourent dans la vie quotidienne sont reconnaissables par leur manière de refléter la lumière : leur apparence est définie par leur micro-géométrie. Les microstructures spéculaires (paillettes métalliques ou micro-rayures) peuvent être visibles sous un éclairage ponctuel et puissant, comme par exemple le soleil. Les réflexions lumineuses intenses apparaissent et disparaissent très rapidement, même avec un faible déplacement de l'observateur. Reproduire ce phénomène en informatique graphique est possible en modélisant la micro-surface par une carte de normales spéculaire et haute définition. L'échantillonnage naïf du pixel n'est pas efficace, car les échantillons peuvent tomber à côté des minuscules zones qui reflètent de la lumière. Par ailleurs, le filtrage de cartes de normales utilisant le MIP mapping moyenne les pentes de la surface et adoucit les reflets spéculaires. Des modèles dédiés capturant les reflets lumineux sont donc nécessaires (figure 3.1).

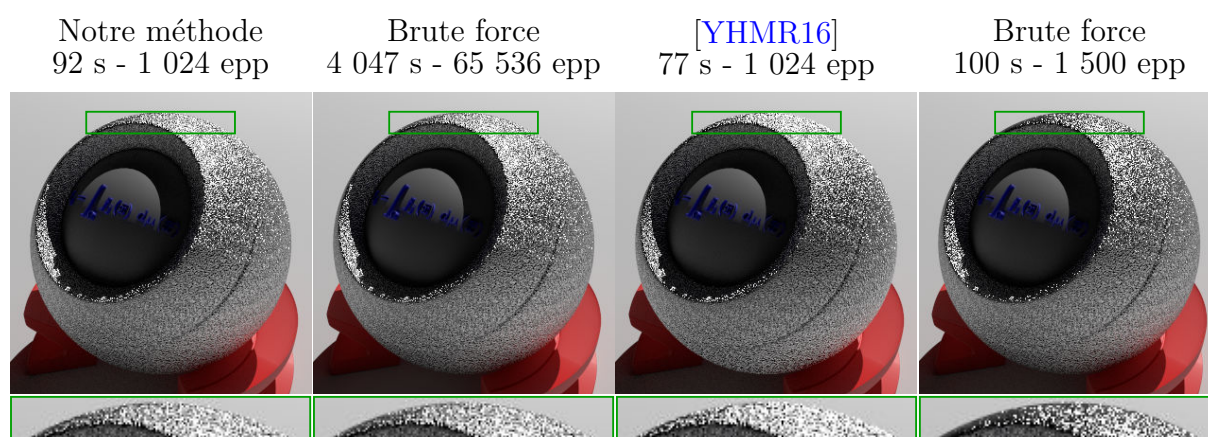


FIGURE 3.1 – Un orbe métallique dont la surface est modélisée par une carte de normales spéculaire et haute définition, éclairé par une lumière ponctuelle. L'échantillonnage naïf du pixel avec 1 500 échantillons par pixel (droite) utilisant à peu près le même budget temps que notre méthode (gauche) ne capture pas tous les reflets et a besoin de beaucoup plus d'échantillons : 65 536 epp pour cette scène (milieu gauche). Notre méthode capture rapidement tous les éclats et reproduit plus fidèlement l'apparence de la référence que la méthode précédente [YHMR16] (milieu droit).

Plusieurs travaux permettent le rendu efficace et multi-échelles de microstructures spéculaires ciblées, comme par exemple les paillettes [JHY⁺14] ou les micro-rayures [RGB16]. Cependant, les artistes ont tendance à préférer les approches basées sur une carte de normales, car elles permettent la création de surfaces complexes et variées. Les travaux de Yan et al. [YHJ⁺14, YHMR16] utilisent cette représentation pour modéliser la micro-surface du matériau : ils ont pour objectif d'obtenir de manière plus efficace le rendu « brute force » d'une carte de normales haute définition et spéculaire. Bien que leurs résultats arrivent à retranscrire la richesse et la complexité de l'apparence d'objets spéculaires réels, ils ne reproduisent pas fidèlement celle obtenue en échantillonnant massivement le pixel/la carte de normales. Cette différence est causée par un problème de normalisation qui ne leur permet pas de conserver l'énergie. Leur méthode utilise une BRDF basée sur les micro-facettes. Heitz [Hei14] a montré que pour être bien défini, ce modèle doit utiliser l'aire projetée et la fonction de masquage correspondante à la

distribution de normales. Ce lien n'est pas pris en compte dans les travaux de Yan et al. [YHJ⁺14, YHMR16].

Nous introduisons dans ce chapitre une définition de la micro-surface qui induit une solution analytique pour le terme de masquage et l'aire projetée, permettant la conservation de l'énergie. Nous proposons aussi une approximation de ces fonctions impliquant seulement un surcoût de temps de calcul de 20 % au plus par rapport à la méthode précédente (c'est la plus mauvaise performance que nous avons eue). Enfin, nous comparons nos résultats et ceux de Yan et al. [YHMR16] avec une référence, c'est-à-dire des rendus obtenus en échantillonnant naïvement et en grande quantité le pixel et la carte de normales.

3.2 Travaux précédents

Les surfaces spéculaires complexes sont difficiles à reproduire dans un moteur de rendu, surtout quand des détails sont introduits avec une carte de normales. Dans une empreinte de rayon, la distribution de normales peut avoir plusieurs lobes pointus (parfois plus d'une centaine) quand la microstructure n'est pas lisse. Elle est souvent approximée à tort par un [Tok05, OB10, DHI⁺13] ou une dizaine de lobes [HSRG07, XWZB17], entraînant la perte de nombreux détails créant pourtant la richesse de l'apparence d'un objet.

3.2.1 Surfaces pailletées

La méthode de Jacob et al. [JHY⁺14] et son amélioration [AK16] décrit la micro-surface par un ensemble aléatoire de petits miroirs. Leur modèle est aussi basé sur la théorie des micro-facettes et consiste à compter le nombre de paillettes spéculaires possédant une normale à mi-chemin entre la direction d'observation et d'éclairage. Pour la fonction de masquage, aucune information n'est donnée et nous pensons qu'ils utilisent celle d'une distribution de Beckmann possédant une rugosité égale à celle contenue en moyenne dans leur surface. Ils ne se comparent pas à une référence, car les paillettes de leur matériau n'ont pas d'emplacement précis, seul leur densité est contrôlable. Ils ne peuvent pas modéliser des surfaces rayées, brossées ou bosselées.

3.2.2 Surfaces rayées

Tout comme les surfaces pailletées, le rendu de métal rayé est aussi un défi important en informatique graphique [MDG01, BPMG04, BP10] car de minuscules rayures peuvent influencer l'intensité d'un pixel. Raymond et al. [RGB16] proposent un modèle dédié pour ce type de matériaux qui gère le multi-rebond dans la micro-surface. La BRDF 3D d'une rayure spéculaire extrudée depuis un profil 2D est approximée pendant le rendu à partir de pré-calculs contenus dans les trois canaux d'une texture 256×256 . En combinant linéairement plusieurs BRDFs de rayures, ils obtiennent un matériau possédant des hautes fréquences angulaires. La prise en compte des hautes fréquences spatiales et de leur filtrage se fait par l'utilisation d'un masque indiquant les zones rayées. L'utilisateur peut contrôler le profil des rayures, leur orientation, leur densité et leur microrugosité. Leur méthode est très efficace, conserve l'énergie, mais elle est limitée à la représentation de rayures.

3.2.3 Surfaces basées sur une carte de normales

L'une des représentations la plus flexible pour les artistes est la texture. Le premier travail utilisant une carte de normales arbitraire pour modéliser la micro-surface est celui de Yan et al. [YHJ⁺14]. Pour une partie \mathcal{P} de la surface associée à un rayon, appelée *empreinte du rayon*, ils évaluent sa distribution de normales associée, appelée la *Patch-Normal Distribution Function* (\mathcal{P} -NDF). Cette évaluation implique une convolution entre le filtre représentant \mathcal{P} et la fonction de densité de normales locale. La carte de normales ne pouvant être utilisée comme telle, ils la triangulent et calculent des intégrales finies (ce qui demande beaucoup de temps). Dans une publication ultérieure [YHMR16], ils proposent une autre approximation permettant une évaluation jusqu'à cent fois plus rapide de la \mathcal{P} -NDF. La carte de normales est approximée par une somme de distributions de normales pondérées par des gaussiennes spatiales 2D, amenant une solution analytique pour la convolution. Mais le problème de la conservation de l'énergie n'est toujours pas réglé car leur terme de normalisation ne prend pas en compte l'empreinte du rayon. Les modèles de réflexion de la lumière basés sur la théorie des micro-facettes doivent être normalisés par l'aire projetée de la micro-surface qui se calcule à partir de la moyenne de la distribution de normales [Hei14]. Dans leur méthode, ils utilisent toujours l'aire projetée d'une surface qui est plane en moyenne alors qu'en réalité, elle l'est rarement, surtout quand l'objet est proche de la caméra. Il en résulte une différence notable avec la référence obtenue en échantillonnant massivement la carte de normales. Calculer le terme de normalisation correct en utilisant leur approximation de la carte de normales n'est pas possible car ils utilisent une somme de distributions de normales 4D. Ces distributions dépendent d'une direction, mais aussi d'une position, et cette dépendance spatiale n'implique pas de solution analytique dans le calcul du terme de normalisation.

Contrairement au travail de Yan et al. [YHMR16], appelé dans ce chapitre la méthode précédente, nous proposons une définition de la micro-surface qui permet d'avoir une forme close lors du calcul du facteur de normalisation. Nous prenons en compte les données relatives à l'empreinte du rayon pour calculer une aire projetée et un terme de masquage correct, nous permettant d'obtenir des rendus très proches de la référence tout en ne créant pas d'énergie ($\int_{\Omega} f(\omega_o, \omega_i) \cos \theta_i d\omega_i \leq 1$).

3.3 Définition de la micro-surface

Utiliser une carte de normales pour définir la micro-surface est une approche simple et intuitive. En interpolant les normales, nous obtenons un ensemble continu qui peut être approximé par une somme de distributions de normales pondérées par des gaussiennes spatiales 2D. Notre représentation de la surface est proche mais pas complètement similaire à celle de Yan et al. [YHMR16]. Les différences sont importantes, car notre définition permet d'avoir un modèle bien défini. Nous listons les principales notations mathématiques pour ce chapitre dans le tableau 3.1.

3.3.1 Représentation des normales par des pentes

La carte de normales peut être vue comme une fonction donnant une pente $\tilde{\mathbf{n}}(\mathbf{u}) \in \mathbb{R}^2$ pour chaque coordonnée de surface \mathbf{u} . Cette représentation permet de mieux intégrer le modèle dans la théorie des micro-facettes, où des distributions de pentes sont souvent

| | |
|---|---|
| $\boldsymbol{\omega}_1 \cdot \boldsymbol{\omega}_2$ | produit scalaire |
| $\langle \boldsymbol{\omega}_1 \cdot \boldsymbol{\omega}_2 \rangle$ | produit scalaire borné : 0 si $\boldsymbol{\omega}_1 \cdot \boldsymbol{\omega}_2 < 0$ |
| $\chi^+(a)$ | fonction de Heaviside : 1 si $a > 0$ et 0 si $a \leq 0$ |
| $\boldsymbol{\omega}_g$ | normale géométrique |
| $\boldsymbol{\omega}_o$ | direction d'observation |
| $\boldsymbol{\omega}_i$ | direction incidente |
| $\boldsymbol{\omega}_n$ | normale de la carte de normales |
| $\tilde{\mathbf{n}}$ | pende d'une normale |
| $\boldsymbol{\omega}_h$ | vecteur à mi-chemin $\frac{\boldsymbol{\omega}_o + \boldsymbol{\omega}_i}{\ \boldsymbol{\omega}_o + \boldsymbol{\omega}_i\ }$ |
| \mathbf{u} | position sur la surface |
| $k_{\mathcal{P}}$ | filtre associé à l'empreinte du rayon |
| $D_{\mathcal{P}}$ | \mathcal{P} -NDF |
| $G_2(\mathcal{P})$ | fonction de masquage et d'ombrage liée à une partie \mathcal{P} de la surface |
| $\Lambda_{\mathcal{P}}$ | fonction Λ de Smith pour un patch |
| $A_{\mathcal{P}}$ | aire projetée d'une surface \mathcal{P} |
| D_i | $i^{\text{ème}}$ distribution de normales de Beckmann |
| k_i | $i^{\text{ème}}$ gaussienne 2D pondérant D_i |
| W_i | $i^{\text{ème}}$ poids : $\int_{\mathbb{R}^2} k_i(\mathbf{u})k_{\mathcal{P}}(\mathbf{u})d\mathbf{u}$ |
| Λ_i | $i^{\text{ème}}$ fonction Λ de Smith |
| $\boldsymbol{\omega}_{\tilde{\mathbf{n}}_i}$ | moyenne de la $i^{\text{ème}}$ distribution de normales |

Tableau 3.1 – Notations importantes du chapitre

utilisées. Pour calculer une pente $\tilde{\mathbf{n}} = (x_{\tilde{\mathbf{n}}}, y_{\tilde{\mathbf{n}}})^T$ associée à une normale $\boldsymbol{\omega}_n = (x_n, y_n, z_n)^T$, nous utilisons les relations mathématiques $x_{\tilde{\mathbf{n}}} = -x_n/z_n$ et $y_{\tilde{\mathbf{n}}} = -y_n/z_n$ (section 2.4.2). Toutes les normales et directions sont définies dans l'hémisphère \mathcal{H}^2 centrée autour de la normale géométrique de la surface $\boldsymbol{\omega}_g = (0, 0, 1)^T$.

3.3.2 Mixture de distributions de normales de Beckmann

Pour évaluer une BRDF basée sur les micro-facettes, nous avons besoin de connaître la densité d'une micro-normale $\boldsymbol{\omega}_m$ pour chaque point \mathbf{u} de la surface. Combiner m distributions de normales de Beckmann décentrées $D_i(\boldsymbol{\omega}_m)$ donne une bonne approximation de la distribution de normales spatiale $D_n(\boldsymbol{\omega}_m, \mathbf{u})$:

$$D_n(\boldsymbol{\omega}_m, \mathbf{u}) = \delta(\boldsymbol{\omega}_n(\mathbf{u}) - \boldsymbol{\omega}_m) \approx \sum_i^m D_i(\boldsymbol{\omega}_m)k_i(\mathbf{u}). \quad (3.3.1)$$

Cette somme incarne notre définition de la micro-surface où $k_i(\mathbf{u})$ est le poids gaussien associé à la distribution de Beckmann $D_i(\boldsymbol{\omega}_m)$. Pour une valeur m assez grande, la distribution de normales spatiale $D_n(\boldsymbol{\omega}_m, \mathbf{u})$ est bien approximée, surtout si les distributions de Beckmann $D_i(\boldsymbol{\omega}_m)$ sont presque égales à la distribution locale réelle associée au filtre $k_i(\mathbf{u})$.

1. Dans ce chapitre, nous utilisons la notation T pour spécifier que le vecteur est un vecteur colonne. Le gras indique que c'est un vecteur. Dans les autres chapitres, ni le gras, ni la notation T sont utilisés, mais les vecteurs sont aussi des vecteurs colonnes. Cette particularité dans ce chapitre permet d'être cohérent avec la publication associée.

3.3.2.1 Définition du poids gaussien

La carte de normales est uniformément échantillonnée par m points \mathbf{u}_i espacés par une distance de h . Pour avoir une transition douce entre les valeurs aux points discrets \mathbf{u}_i , elles sont pondérées par des gaussiennes 2D $k_i(\mathbf{u})$ centrées autour de \mathbf{u}_i (haut de la figure 3.2). Leur écart type σ_h est fixé à $h/\sqrt{8 \log 2}$, ce qui permet d'avoir une contribution de 50 % pour chacune des deux valeurs associées à deux échantillons \mathbf{u}_i adjacents, lorsque l'évaluation se fait à mi-chemin entre les deux. L'intégrale de la gaussienne $k_i(\mathbf{u})$ sur \mathbb{R}^2 doit aussi être égale à l'aire qu'elle représente : $\int_{\mathbb{R}^2} k_i(\mathbf{u}) d\mathbf{u} = h^2$.

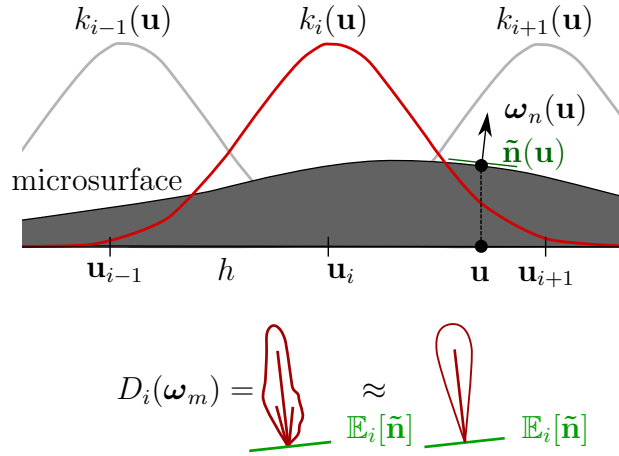


FIGURE 3.2 – Haut : Illustration des gaussiennes 2D $k_i(\mathbf{u})$. Bas : Nous approximos la distribution de normales D_i associée à $k_i(\mathbf{u})$ par une distribution de Beckmann décentrée.

3.3.3 Reconstruction locale de NDF

Une distribution de normales peut être construite à partir d'une distribution de pentes P^{22} où

$$D(\boldsymbol{\omega}_m) = \frac{P^{22}(\tilde{\mathbf{m}})}{(\boldsymbol{\omega}_m \cdot \boldsymbol{\omega}_g)^4}. \quad (3.3.2)$$

Pour une gaussienne 2D $k_i(\mathbf{u})$ pondérant les normales de la texture, la distribution de pentes associée est

$$P_i^{22}(\tilde{\mathbf{m}}) = \int_{\mathbb{R}^2} k_i(\mathbf{u}) \delta(\tilde{\mathbf{n}}(\mathbf{u}) - \tilde{\mathbf{m}}) d\mathbf{u}. \quad (3.3.3)$$

Cette distribution peut être approximée par une gaussienne de pentes 2D en calculant numériquement la pente moyenne, l'écart type et le coefficient de corrélation relatif au noyau $k_i(\mathbf{u})$:

$$P_i^{22}(\tilde{\mathbf{m}}) \approx \frac{\exp\left(-\frac{1}{2}(\tilde{\mathbf{m}} - \mathbb{E}_i[\tilde{\mathbf{n}}])^T \Sigma_i^{-1} (\tilde{\mathbf{m}} - \mathbb{E}_i[\tilde{\mathbf{n}}])\right)}{2\pi \sqrt{|\Sigma_i|}}. \quad (3.3.4)$$

La matrice de covariance de cette distribution normale est $\Sigma_i = \begin{bmatrix} \sigma_{x,i}^2 & c_{xy,i} \\ c_{xy,i} & \sigma_{y,i}^2 \end{bmatrix}$ et où ses coefficients sont

$$\begin{aligned} \sigma_{x,i}^2 &= \mathbb{E}_i[x_{\tilde{n}}^2] - \mathbb{E}_i[x_{\tilde{n}}]^2, \\ \sigma_{y,i}^2 &= \mathbb{E}_i[y_{\tilde{n}}^2] - \mathbb{E}_i[y_{\tilde{n}}]^2, \\ c_{xy,i}^2 &= \mathbb{E}_i[x_{\tilde{n}}y_{\tilde{n}}] - \mathbb{E}_i[x_{\tilde{n}}]\mathbb{E}_i[y_{\tilde{n}}]. \end{aligned} \quad (3.3.5)$$

Les $i^{\text{ème}}$ espérances mathématiques sont calculées à partir de $k_i(\mathbf{u})$: par exemple, la pente moyenne est

$$\mathbb{E}_i[\tilde{\mathbf{n}}] = \int_{\mathbb{R}^2} \tilde{\mathbf{n}}(\mathbf{u})k_i(\mathbf{u})d\mathbf{u}. \quad (3.3.6)$$

Les définitions des autres espérances mathématiques sont similaires. La $i^{\text{ème}}$ distribution de normales locale s'écrit $D_i(\boldsymbol{\omega}_m) = \frac{P_i^{22}(\tilde{\mathbf{m}})}{(\boldsymbol{\omega}_m \cdot \boldsymbol{\omega}_g)^4}$. Nous utilisons l'approximation gaussienne (équation 3.3.4) pour P_i^{22} dans le but d'utiliser que des distributions de normales de Beckmann décentrées (figure 3.2, bas), donnant une solution analytique pour le masquage.

Les paramètres des distributions de normales locales sont précalculés en utilisant une intégration numérique de Monte-Carlo. Chaque gaussienne 2D $k_i(\mathbf{u})$ est échantillonnée plusieurs fois donnant un ensemble de positions \mathbf{u} . Ces positions sont utilisées pour calculer un ensemble de valeurs $x_{\tilde{n}}, y_{\tilde{n}}, x_{\tilde{n}}^2, y_{\tilde{n}}^2$ et $x_{\tilde{n}}y_{\tilde{n}}$ et leurs moyennes sont respectivement $\mathbb{E}_i[x_{\tilde{n}}], \mathbb{E}_i[y_{\tilde{n}}], \mathbb{E}_i[x_{\tilde{n}}^2], \mathbb{E}_i[y_{\tilde{n}}^2]$ et $\mathbb{E}_i[x_{\tilde{n}}y_{\tilde{n}}]$. Quand la surface est parfaitement plate localement, les écarts types $\sigma_{x,i}$ et $\sigma_{y,i}$ valent 0, créant une singularité. Dans ce cas, ils sont fixés à une petite valeur $\sigma_\epsilon = 0,01$. Intuitivement, cela implique que la surface de la carte de normales possède une rugosité extrêmement faible, si faible qu'elle semble parfaitement spéculaire.

3.4 \mathcal{P} -BRDF

Une \mathcal{P} -BRDF est une BRDF à base de micro-facettes [WMLT07, Hei14], avec une fonction de masquage et d'ombrage associée G_2 (section 3.4.2) et une aire projetée $A_{\mathcal{P}}$ relative à une surface \mathcal{P} . La définition mathématique de la \mathcal{P} -BRDF est

$$f_{\mathcal{P}}(\boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = \frac{F(\boldsymbol{\omega}_o, \boldsymbol{\omega}_h)G_2(\mathcal{P}, \boldsymbol{\omega}_h, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i)D_{\mathcal{P}}(\boldsymbol{\omega}_h)}{4A_{\mathcal{P}}(\boldsymbol{\omega}_o)A_{\mathcal{P}}(\boldsymbol{\omega}_i)} \quad (3.4.1)$$

où $\boldsymbol{\omega}_h$ est le vecteur à mi-chemin, $D_{\mathcal{P}}$ est la \mathcal{P} -NDF, F est le facteur de Fresnel et \mathcal{P} l'empreinte du rayon définie dans la section suivante. Normalement, les aires projetées $A_{\mathcal{P}}(\boldsymbol{\omega}_o)$ et $A_{\mathcal{P}}(\boldsymbol{\omega}_i)$ sont les cosinus de l'angle polaire de respectivement la direction sortante et incidente, à part quand une carte de normales modifie la surface [DHI⁺13], ce qui est précisément notre cas.

3.4.1 \mathcal{P} -NDF

Déterminer toute la luminance énergétique quittant une empreinte de rayon \mathcal{P} sur une carte de normales revient en premier lieu à évaluer la \mathcal{P} -NDF. Sa définition est basée sur celle de l'empreinte de rayon et de son noyau associé $k_{\mathcal{P}}(\mathbf{u})$. Plusieurs méthodes de

filtrage de surface donnent l'orientation et la taille de ce noyau, comme par exemple les rayons différentiels [Ige99], les chemins différentiels [SW01] ou le filtrage par covariance [BYRN17]. Il est défini dans le plan tangent et nous considérons ici un noyau gaussien 2D normalisé sur \mathbb{R}^2 , satisfaisant l'équation $\int_{\mathbb{R}^2} k_{\mathcal{P}}(\mathbf{u})d\mathbf{u} = 1$ (voir la section 2.6.1 pour plus de détails).

En utilisant cette définition de l'empreinte du rayon, nous pouvons maintenant définir formellement la \mathcal{P} -NDF (ici avec un vecteur à mi-chemin $\boldsymbol{\omega}_h$) :

$$\begin{aligned} D_{\mathcal{P}}(\boldsymbol{\omega}_h) &= \int_{\mathbb{R}^2} D_n(\boldsymbol{\omega}_h, \boldsymbol{\omega}_n(\mathbf{u}))k_{\mathcal{P}}(\mathbf{u})d\mathbf{u} \\ &= \int_{\mathbb{R}^2} \delta(\boldsymbol{\omega}_n(\mathbf{u}) - \boldsymbol{\omega}_h)k_{\mathcal{P}}(\mathbf{u})d\mathbf{u}. \end{aligned} \quad (3.4.2)$$

En utilisant notre somme de distributions de Beckmann (équation 3.3.1), nous obtenons l'expression plus pratique

$$\begin{aligned} D_{\mathcal{P}}(\boldsymbol{\omega}_h) &\approx \int_{\mathbb{R}^2} \sum_i^m D_i(\boldsymbol{\omega}_h)k_i(\mathbf{u})k_{\mathcal{P}}(\mathbf{u})d\mathbf{u} \\ &\approx \sum_i^m D_i(\boldsymbol{\omega}_h) \int_{\mathbb{R}^2} k_i(\mathbf{u})k_{\mathcal{P}}(\mathbf{u})d\mathbf{u} \\ &\approx \sum_i^m D_i(\boldsymbol{\omega}_h)W_i \end{aligned} \quad (3.4.3)$$

où la distribution de normales $D_i(\boldsymbol{\omega}_h)$ est sortie de l'intégrale car elle ne dépend pas de la position \mathbf{u} . Chaque poids W_i a une solution analytique car c'est le produit de deux gaussiennes. La somme de tous les poids est normalisée, donnant $\sum_{i=1}^m W_i = 1$. Si la déviation standard de l'empreinte de rayon est inférieure à σ_h , alors la somme ne fait pas exactement 1 (il peut y avoir 10% d'erreur environ). Mais en pratique, l'empreinte de rayon recouvre toujours plusieurs gaussiennes 2D k_i .

3.4.2 Masquage-ombrage de Smith surfacique

Une partie importante de la théorie des micro-facettes est le masquage et l'ombrage associé au profil de micro-surface. Ce terme a une influence sur le rendu (figure 3.3) et normalise la \mathcal{P} -BRDF, garantissant la conservation de l'énergie. Dans cette section, nous dérivons exactement le terme de Smith surfacique pour notre modèle de réflexion de la lumière.

3.4.2.1 Fonction de masquage surfacique

Le profil de micro-surface de Smith [Smi67, Hei14] considère que la fonction de masquage G_1 ne dépend pas des micro-normales $\boldsymbol{\omega}_m$ quand elles ne sont pas auto-masquées ($\boldsymbol{\omega}_o \cdot \boldsymbol{\omega}_m > 0$). Cela implique l'existence d'un masquage *local* et *distant*. Le premier est lié à la micro-normale $\boldsymbol{\omega}_m$ et est défini par $\chi^+(\boldsymbol{\omega}_m \cdot \boldsymbol{\omega}_o)$ alors que le second dépend juste de la direction d'observation $\boldsymbol{\omega}_o$. Dans notre cas, le masquage distant $G_1^{\text{dist}}(\mathcal{P}, \boldsymbol{\omega}_o)$ dépend aussi de l'empreinte du rayon \mathcal{P} . La fonction de masquage surfacique est la combinaison

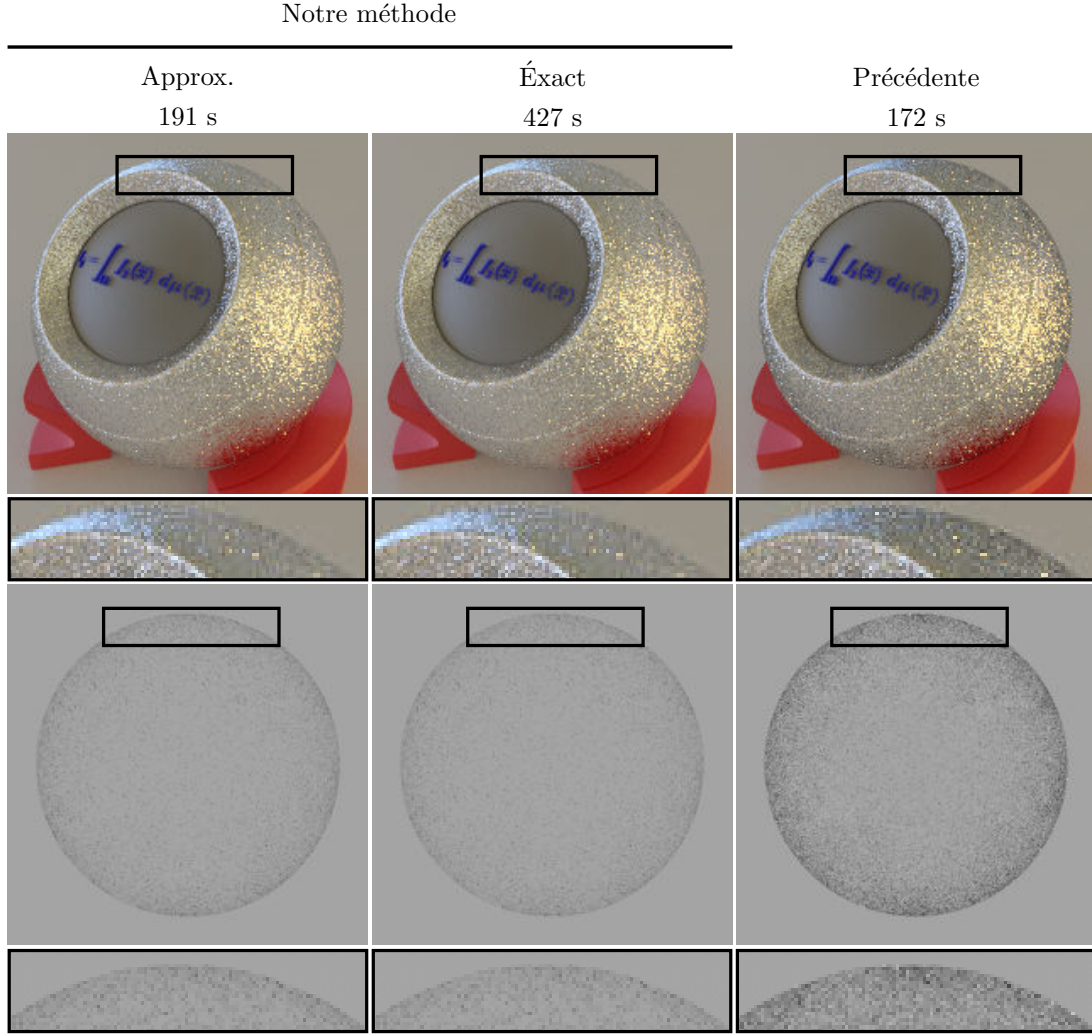


FIGURE 3.3 – Comparaison entre différents masquage-ombrages et aires projetées pour une même \mathcal{P} -NDF. À gauche : utilisation de nos approximations. Au centre : utilisation de nos termes exacts. À droite : utilisation des termes de la méthode précédente. Ligne du haut : rendus d'un orbe métallique rugueux. Ligne du bas : résultats du test du four blanc [Hei14] pour ce même matériau. Nous obtenons de meilleurs résultats en matière de conservation de l'énergie, car la carte de normales est globalement extrêmement rugueuse, impliquant une surestimation du masquage-ombrage pour la méthode précédente.

de ces deux termes, où la micro-normale est le vecteur à mi-chemin ω_h :

$$G_1(\mathcal{P}, \omega_h, \omega_o) = \chi^+(\omega_h \cdot \omega_o) G_1^{\text{dist}}(\mathcal{P}, \omega_o). \quad (3.4.4)$$

3.4.2.2 Masquage distant

La fonction de masquage de Smith distante et surfacique normalise la BRDF : elle implique une intégrale sur le domaine hémisphérique \mathcal{H}^2 :

$$G_1^{\text{dist}}(\mathcal{P}, \omega_o) = \frac{A_{\mathcal{P}}(\omega_o)}{\int_{\mathcal{H}^2} \langle \omega_o, \omega_h \rangle D_{\mathcal{P}}(\omega_h) d\omega_h}. \quad (3.4.5)$$

Cette forme est similaire à celle de l'équation 2.4.18, où le cosinus est devenu une aire projetée et où la NDF est maintenant une \mathcal{P} -NDF. L'aire projetée $A_{\mathcal{P}}(\boldsymbol{\omega}_o)$ de l'empreinte du rayon est étudiée dans la section 3.4.2.4. Au dénominateur, nous injectons notre formulation de la \mathcal{P} -NDF (équation 3.4.3) et sortons le poids W_i de l'intégrale car il ne dépend que de la position \mathbf{u} sur la surface :

$$\begin{aligned} G_1^{\text{dist}}(\mathcal{P}, \boldsymbol{\omega}_o) &= \frac{A_{\mathcal{P}}(\boldsymbol{\omega}_o)}{\int_{\mathcal{H}^2} \langle \boldsymbol{\omega}_o, \boldsymbol{\omega}_h \rangle \sum_i^m D_i(\boldsymbol{\omega}_h) W_i d\boldsymbol{\omega}_h} \\ &= \frac{A_{\mathcal{P}}(\boldsymbol{\omega}_o)}{\sum_i^m \int_{\mathcal{H}^2} \langle \boldsymbol{\omega}_o, \boldsymbol{\omega}_h \rangle D_i(\boldsymbol{\omega}_h) d\boldsymbol{\omega}_h W_i}. \end{aligned} \quad (3.4.6)$$

L'intégrale peut s'exprimer par la combinaison d'une aire projetée locale $\frac{\langle \boldsymbol{\omega}_{\bar{n}_i}, \boldsymbol{\omega}_o \rangle}{\boldsymbol{\omega}_{\bar{n}_i} \cdot \boldsymbol{\omega}_g}$ et d'un terme de Smith Λ_i [Hei14] correspondant à la $i^{\text{ème}}$ NDF :

$$G_1^{\text{dist}}(\mathcal{P}, \boldsymbol{\omega}_o) = \frac{A_{\mathcal{P}}(\boldsymbol{\omega}_o)}{\sum_i^m \frac{\langle \boldsymbol{\omega}_{\bar{n}_i}, \boldsymbol{\omega}_o \rangle}{\boldsymbol{\omega}_{\bar{n}_i} \cdot \boldsymbol{\omega}_g} (1 + \Lambda_i(\boldsymbol{\omega}_o)) W_i} \quad (3.4.7)$$

où $\boldsymbol{\omega}_{\bar{n}_i}$ est la normale moyenne de D_i , calculée à partir de la pente moyenne $\mathbb{E}_i[\bar{\mathbf{n}}]$. Grâce à notre formulation de la \mathcal{P} -NDF sous forme d'une somme de distributions de Beckmann décentrées, non-alignées sur les axes et anisotropes, une solution analytique existe pour $\Lambda_i(\boldsymbol{\omega}_o)$ [DHI⁺13].

3.4.2.3 Forme généralisée de la fonction de masquage de Smith surfacique

En développant la somme de l'équation 3.4.7 et en divisant par l'aire projetée $A_{\mathcal{P}}$ définie en section 3.4.2.4, nous obtenons la forme généralisée du masquage de Smith surfacique :

$$G_1^{\text{dist}}(\mathcal{P}, \boldsymbol{\omega}_o) = \frac{1}{1 + \Lambda_{\mathcal{P}}(\boldsymbol{\omega}_o)} \quad (3.4.8)$$

où le terme de Smith surfacique $\Lambda_{\mathcal{P}}$ est :

$$\Lambda_{\mathcal{P}}(\boldsymbol{\omega}_o) = \frac{\sum_i^m \frac{\langle \boldsymbol{\omega}_{\bar{n}_i}, \boldsymbol{\omega}_o \rangle}{\boldsymbol{\omega}_{\bar{n}_i} \cdot \boldsymbol{\omega}_g} \Lambda_i(\boldsymbol{\omega}_o) W_i}{A_{\mathcal{P}}(\boldsymbol{\omega}_o)}. \quad (3.4.9)$$

Nous pouvons maintenant injecter l'équation 3.4.8 dans l'équation 3.4.4 et l'utiliser pour définir la fonction de masquage et d'ombrage surfacique :

$$G_2(\mathcal{P}, \boldsymbol{\omega}_h, \boldsymbol{\omega}_o, \boldsymbol{\omega}_i) = G_1(\mathcal{P}, \boldsymbol{\omega}_h, \boldsymbol{\omega}_o) G_1(\mathcal{P}, \boldsymbol{\omega}_h, \boldsymbol{\omega}_i). \quad (3.4.10)$$

Ce terme est utilisé pour évaluer la \mathcal{P} -BRDF (équation 3.4.1). Nous devons maintenant définir l'aire projetée $A_{\mathcal{P}}$.

3.4.2.4 Aire projetée

La fonction de masquage-ombrage est une partie importante de la théorie des micro-facettes. Pour être bien définie, elle doit utiliser l'aire projetée $A_{\mathcal{P}}(\boldsymbol{\omega}_o)$ de la surface. Il est à noter qu'elle est à la fois utilisée dans les équations 3.4.1 et 3.4.5.

En utilisant notre formulation de la \mathcal{P} -NDF (équation 3.4.3), l'aire projetée $A_{\mathcal{P}}(\boldsymbol{\omega}_o)$ est la somme des $i^{\text{ème}}$ aires projetées $\frac{\langle \boldsymbol{\omega}_{\tilde{\mathbf{n}}_i}, \boldsymbol{\omega}_o \rangle}{\boldsymbol{\omega}_{\tilde{\mathbf{n}}_i} \cdot \boldsymbol{\omega}_g}$ correspondant à chaque distribution de normales D_i , pondérées par W_i :

$$A_{\mathcal{P}}(\boldsymbol{\omega}_o) = \sum_{i=1}^m \frac{\langle \boldsymbol{\omega}_{\tilde{\mathbf{n}}_i}, \boldsymbol{\omega}_o \rangle}{\boldsymbol{\omega}_{\tilde{\mathbf{n}}_i} \cdot \boldsymbol{\omega}_g} W_i. \quad (3.4.11)$$

Ce terme remplace le classique cosinus de l'angle polaire de la direction d'observation (utilisé par Yan et al. [YHMR16]) qui, dans le cas où la surface est augmentée par une carte de normales, ne peut être utilisé [DHI⁺13].

3.4.2.5 Approximations

Le terme de masquage et l'aire projetée que nous venons de dériver nécessitent le parcours des échantillons discrets contenus dans l'empreinte du pixel. Lorsque que la surface est éloignée de l'observateur, ce nombre est très grand, réduisant les performances. Si la conservation de l'énergie stricte n'est pas un besoin critique, nous proposons une approximation pour le masquage-ombrage (équation 3.4.10) et l'aire projetée $A_{\mathcal{P}}$ (équation 3.4.11).

De manière identique à Dupuy et al. [DHI⁺13], nous utilisons les moments d'ordre un et deux des pentes pour connaître à la volée leur moyenne, leur écart type et leur facteur de corrélation dans une empreinte de rayon. Ces statistiques servent à calculer une distribution de normales de Beckmann moyenne relative à l'empreinte du rayon \mathcal{P} . Son terme de masquage et d'ombrage de Smith est utilisé comme approximation de G_2 . L'aire projetée $A_{\mathcal{P}}$ est calculée à partir de la pente moyenne. La structure pyramidale (MIP) donne rapidement les statistiques des pentes dans \mathcal{P} .

3.5 Résultats et discussions

Notre \mathcal{P} -BRDF est implémentée dans le moteur de rendu basé physique PBRT-v3 [PJH16]. L'échantillonnage par importance est réalisé comme dans Yan et al. [YHMR16]. Comme pour beaucoup de BRDFs à micro-facettes, il passe par l'échantillonnage de la distribution de normales. Dans notre cas, il peut se faire en échantillonnant directement la carte de normales ayant servi à construire la mixture de NDFs. L'empreinte du rayon, qui est une gaussienne 2D, est échantillonnée, donnant une position \mathbf{u} . La normale associée $\boldsymbol{\omega}_n(\mathbf{u})$ est ensuite utilisée pour refléter la direction d'observation, donnant un bon candidat pour la direction de la lumière. La PDF associée à la procédure d'échantillonnage est $D_{\mathcal{P}}(\boldsymbol{\omega}_n)(\boldsymbol{\omega}_n \cdot \boldsymbol{\omega}_g)$ multiplié par le jacobien de l'opération de réflexion.

L'évaluation de $D_{\mathcal{P}}(\boldsymbol{\omega}_h)$ est accélérée grâce à une structure accélératrice presque identique à celle de la méthode précédente. Une hiérarchie de boîtes englobantes de NDFs dans l'espace $(\mathbf{u}, \tilde{\mathbf{n}})$ permet d'élaguer rapidement une grande quantité de NDFs qui n'ont pas une contribution significative dans la somme de l'équation 3.4.3. La requête sur la structure se fait pour un vecteur à mi-chemin $\boldsymbol{\omega}_h$ et une empreinte de rayon $k_{\mathcal{P}}$. Nous considérons que le filtre gaussien $k_i(\mathbf{u})$ a une valeur non nulle pour des positions \mathbf{u} comprises dans la boîte 2D $(\mathbf{u}_i \pm 3\boldsymbol{\sigma}_h)$. Pour la distribution de normales D_i , nous considérons qu'elle retourne des valeurs non nulles pour des pentes $\tilde{\mathbf{n}}$ comprises dans la boîte 2D englobant la région contenant 99 % de la masse de P_i^{22} .

3.5.1 \mathcal{P} -NDF

Dans cette section, nous testons la précision de notre \mathcal{P} -NDF en la comparant avec la référence et le travail précédent. Nous mesurons aussi sa performance et son occupation mémoire.

3.5.1.1 Précision

Notre \mathcal{P} -NDF approxime précisément la référence comme nous pouvons le voir dans la figure 3.4. Le nombre m d'échantillons nécessaires dépend du contenu fréquentiel de la carte de normales. Dans tous nos résultats, nous fixons $h = 1$, c'est-à-dire une distribution de normales par texel. La \mathcal{P} -NDF de la méthode précédente est légèrement plus précise que la nôtre, mais les différences n'impactent pas le rendu (figure 3.5). Une correspondance parfaite n'est pas nécessaire pour reproduire l'apparence obtenue en échantillonnant naïvement la carte de normales. Par contre, l'utilisation d'un masquage-ombrage incorrect crée une différence notable avec la référence (section 3.5.2.2).

3.5.1.2 Performance et coût mémoire

La mixture de distributions de normales de la méthode précédente utilise des gaussiennes de pentes 4D alors que notre méthode utilise des gaussiennes de pentes 2D. Contrairement à nous, Yan et al. [YHMR16] doivent extraire une tranche 2D pour évaluer la \mathcal{P} -NDF, impliquant un coût en temps de calcul plus important pour eux. De plus, leur représentation est plus lourde en mémoire, car elle nécessite 12 valeurs décimales par gaussienne. Il en faut 10 pour la matrice de covariance 4×4 qui est symétrique, plus 2 pour stocker la normale moyenne de chaque gaussienne. Comparativement, notre représentation requiert 5 valeurs décimales par NDF ($\mathbb{E}_i[x_{\bar{n}}], \mathbb{E}_i[y_{\bar{n}}], \mathbb{E}_i[x_{\bar{n}}^2], \mathbb{E}_i[y_{\bar{n}}^2]$ et $\mathbb{E}_i[x_{\bar{n}}y_{\bar{n}}]$).

3.5.2 Rendus

Les rendus sont faits sur un ordinateur possédant 16 unités d'exécution (2 processeurs Intel Xeon E5-2650 v2 à 8 cœurs). Nous utilisons des textures MIP de taille 2048×2048 (figure 3.6) et une distribution de normales par texel, nécessitant 80 Mib de mémoire pour les gaussiennes et 512 Mib pour la structure accélératrice. Pour la même configuration, la méthode précédente utilise 160 Mib pour les gaussiennes et la même quantité de mémoire pour la structure accélératrice.

3.5.2.1 Impact des approximations

Nous montrons ici la capacité de notre masquage-ombrage et de notre aire projetée basés sur une pyramide MIP à bien approximer les termes exacts (équations 3.4.10 et 3.4.11). Les figures 3.3 et 3.7 montrent que nous avons presque la même apparence en utilisant soit nos approximations, soit nos termes exacts. Des différences minimes sont visibles à angles rasants, où le masquage n'est pas nul. L'approximation tend à le surestimer, car la moyenne des distributions de Beckmann augmente la rugosité de l'empreinte de rayon. En revanche, l'utilisation d'un niveau de pyramide adapté à cette empreinte permet de borner les temps de calcul, augmentant les performances. De plus, contrairement à la méthode précédente, notre approximation ne crée ou ne perd pas

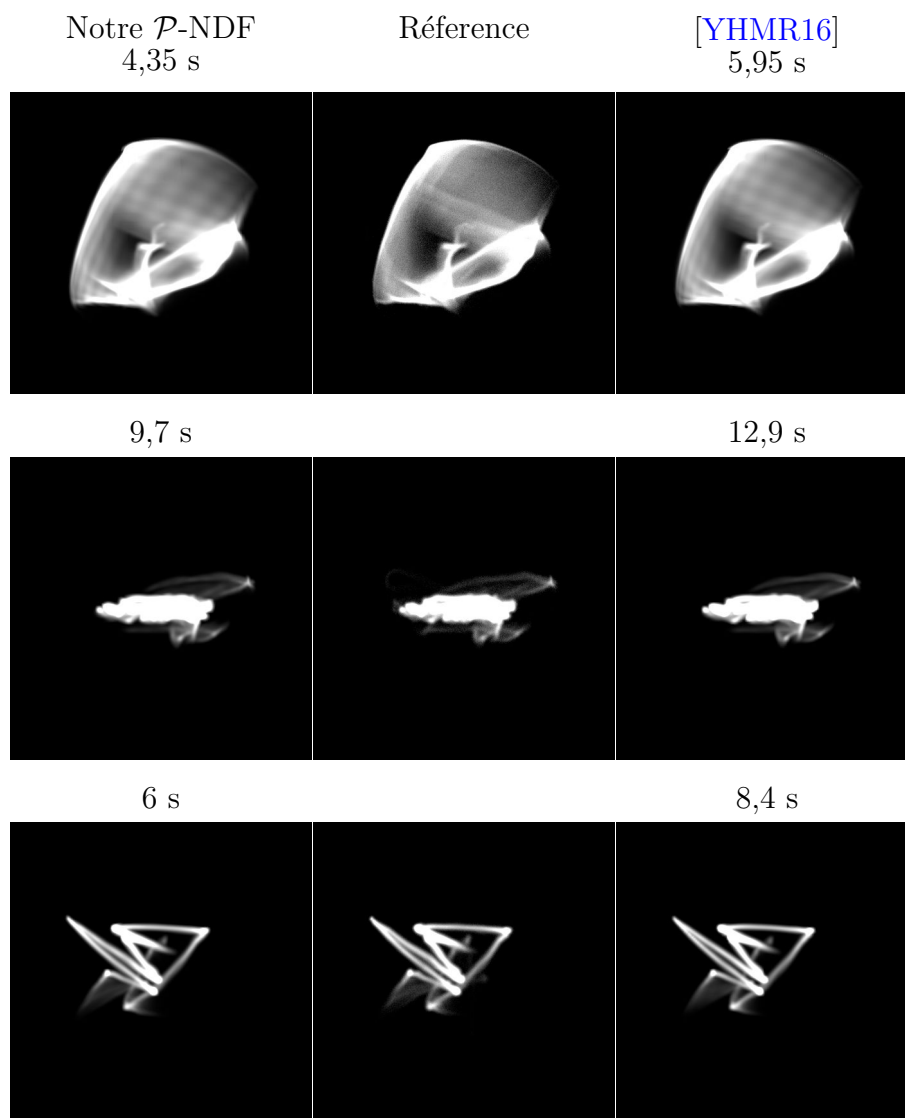


FIGURE 3.4 – Comparaison de notre \mathcal{P} -NDF (à gauche), de celle obtenue en échantillonnant massivement la carte de normales interpolées (au milieu) et de celle proposée par la méthode précédente (à droite). Les temps affichés correspondent aux temps de construction des données et reflètent la performance d'évaluation de la \mathcal{P} -NDF utilisant la structure accélératrice.

d'énergie de manière visible, comme le montre la figure 3.3. Les résultats qui suivent utilisent par défaut les termes approximés.

3.5.2.2 Comparaison avec la référence

Pour valider notre méthode, nous nous comparons avec le rendu naïf obtenu en échantillonnant massivement le pixel. Nous comparons aussi la méthode précédente avec la référence pour montrer l'impact non négligeable sur le rendu qu'à une mauvaise normalisation de la BRDF.

Le rendu classique d'une carte de normales requiert au moins 65 536 epp pour capturer la majorité des reflets lumineux dans nos scènes, à part pour la figure 3.10 où nous en

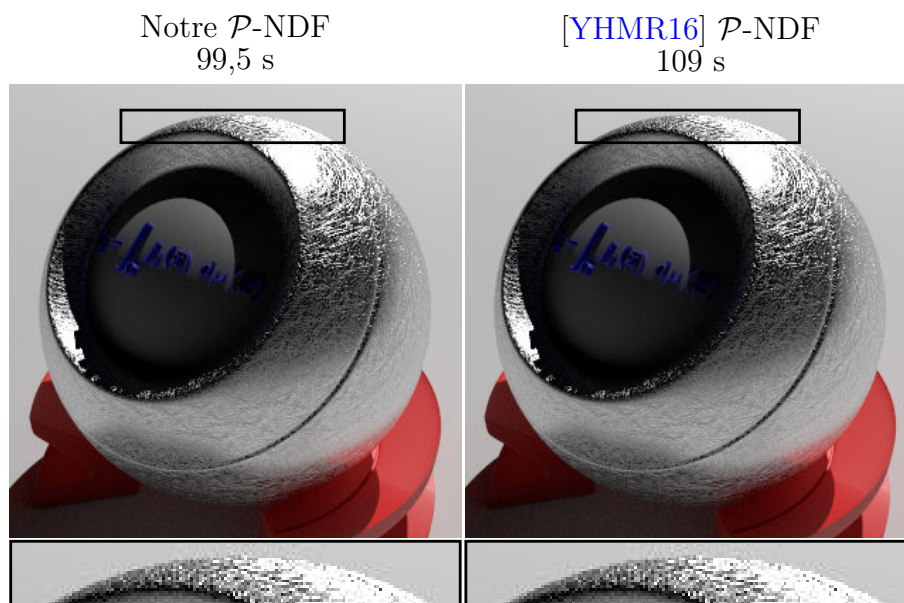


FIGURE 3.5 – Un orbe métallique rayé. À gauche, c’est notre \mathcal{P} -NDF qui est utilisée, et à droite, celle de la méthode précédente. Le même écart h est utilisé entre les échantillons. Le masquage-ombrage et l’aire projetée sont ceux de la méthode précédente. Nous obtenons une correspondance forte entre les deux rendus, montrant que notre mixture de distributions est assez précise pour reproduire les matériaux obtenus avec celle de la méthode précédente.

utilisons plus d’un million. Le matériau de la carte de normales est presque parfaitement spéculaire en utilisant une BRDF à micro-facettes possédant une rugosité proche de zéro ($\sigma_\epsilon = 0,01$). La réflectance est calculée en utilisant la normale perturbée issue de la carte de normales, au point d’intersection du rayon lumineux avec la surface. Le même paramètre de rugosité est utilisé pour construire la mixture de distributions de normales pour notre méthode (section 3.3) et celle de la méthode précédente. La figure 3.8 montre notre capacité à modéliser des surfaces pailletées comme une peinture métallisée ou une surface brossée de façon plus fidèle que Yan et al. [YHMR16]. C’est aussi le cas des objets rayés et métalliques, comme illustré dans les figures 3.9 et 3.10. Des versions temporelles des figures 3.8 et 3.9 sont disponibles dans les résultats vidéo (voir le [site du projet](#)).

Des différences minimes sont visibles entre nos rendus et ceux de la référence, comme par exemple aux angles rasants où l’échantillonnage naïf donne des bords plus sombres pour les objets. Ces divergences sont dues à la combinaison de trois facteurs. Le premier, comme pour la méthode précédente, est lié à l’approximation de la carte de normales et de l’empreinte du pixel. L’empreinte réelle peut avoir une forme différente d’une ellipse plane, surtout là où la surface est très incurvée. Le deuxième facteur à prendre en compte est le fait que l’échantillonnage naïf ne peut pas capturer complètement tous les reflets lumineux à angles rasants, car ils peuvent représenter une portion infinitésimale dans le pixel. Par exemple, à angles extrêmement rasants, l’empreinte du pixel peut faire environ $320\,000^2$ texels, ce qui nécessiterait $2^{19} \times 2^{19}$ échantillons par pixel pour avoir au moins un échantillon par texel. En pratique, nous ne pouvons pas en utiliser autant, ce qui implique que les bords des objets sont plus sombres pour la référence. Enfin, le dernier facteur produisant des différences est lié à notre approximation du masquage-ombrage

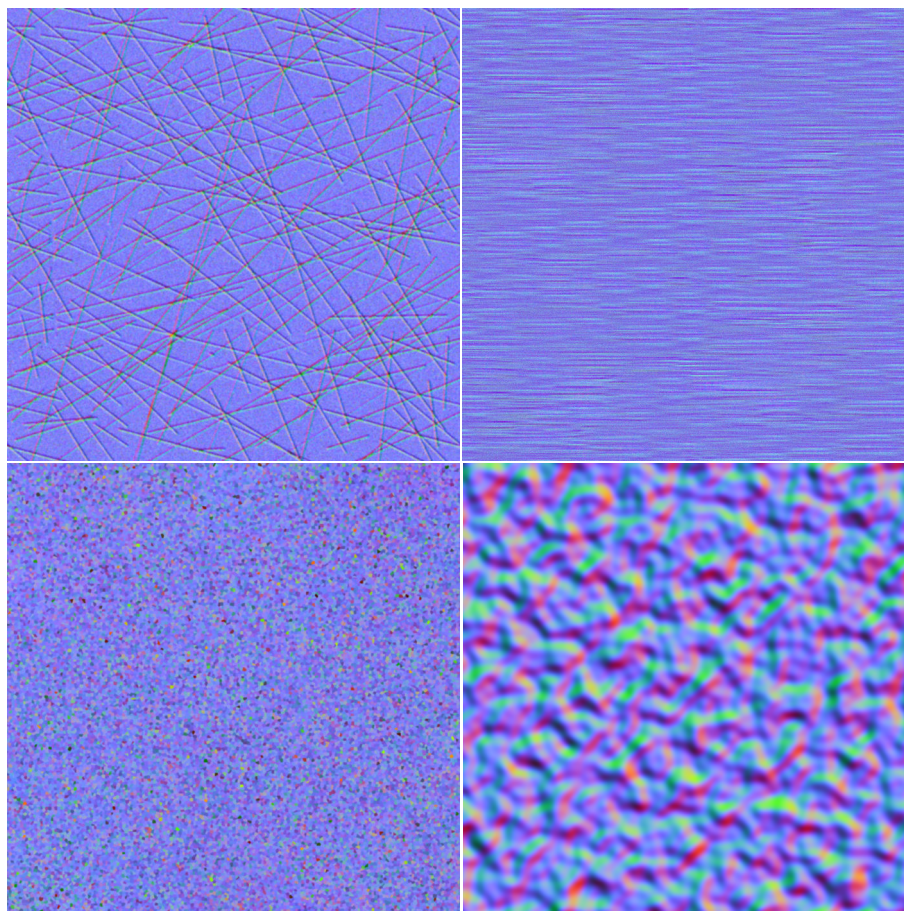


FIGURE 3.6 – Les cartes de normales utilisées pour nos résultats. En haut à gauche : surface rayée utilisée dans les figures 3.5, 3.9 et 3.10. En haut à droite : surface brossée utilisée dans les figures 3.7 et 3.8. En bas à gauche : surface très rugueuse utilisée pour les figures 3.7 et 3.8. Une version encore plus rugueuse est utilisée dans la figure 3.3. En bas à droite : surface rugueuse utilisée dans la figure 3.1.

qui moyenne les distributions de normales en une seule distribution, avec un écart type important. En utilisant son terme de masquage-ombrage associé, nous obtenons forcément une surestimation qui reste toutefois convenable comparée à la méthode précédente. Ces comportements peuvent être observés dans la plupart des rendus.

3.6 Conclusion

Nous avons proposé dans ce chapitre un nouveau modèle de réflexion de la lumière basé sur la théorie des micro-facettes où la micro-surface est définie par une mixture de distributions de Beckmann. Cette somme de distributions de normales approxime de manière assez précise la carte de normales spéculaire interpolée. Notre représentation s'intègre dans un moteur de rendu basé physique et permet la conservation de l'énergie grâce à un terme de masquage-ombrage exact possédant une solution analytique. Nous proposons une approximation précise de ce terme permettant d'avoir des performances proches de celles de la méthode précédente, mais en ayant une meilleure correspondance

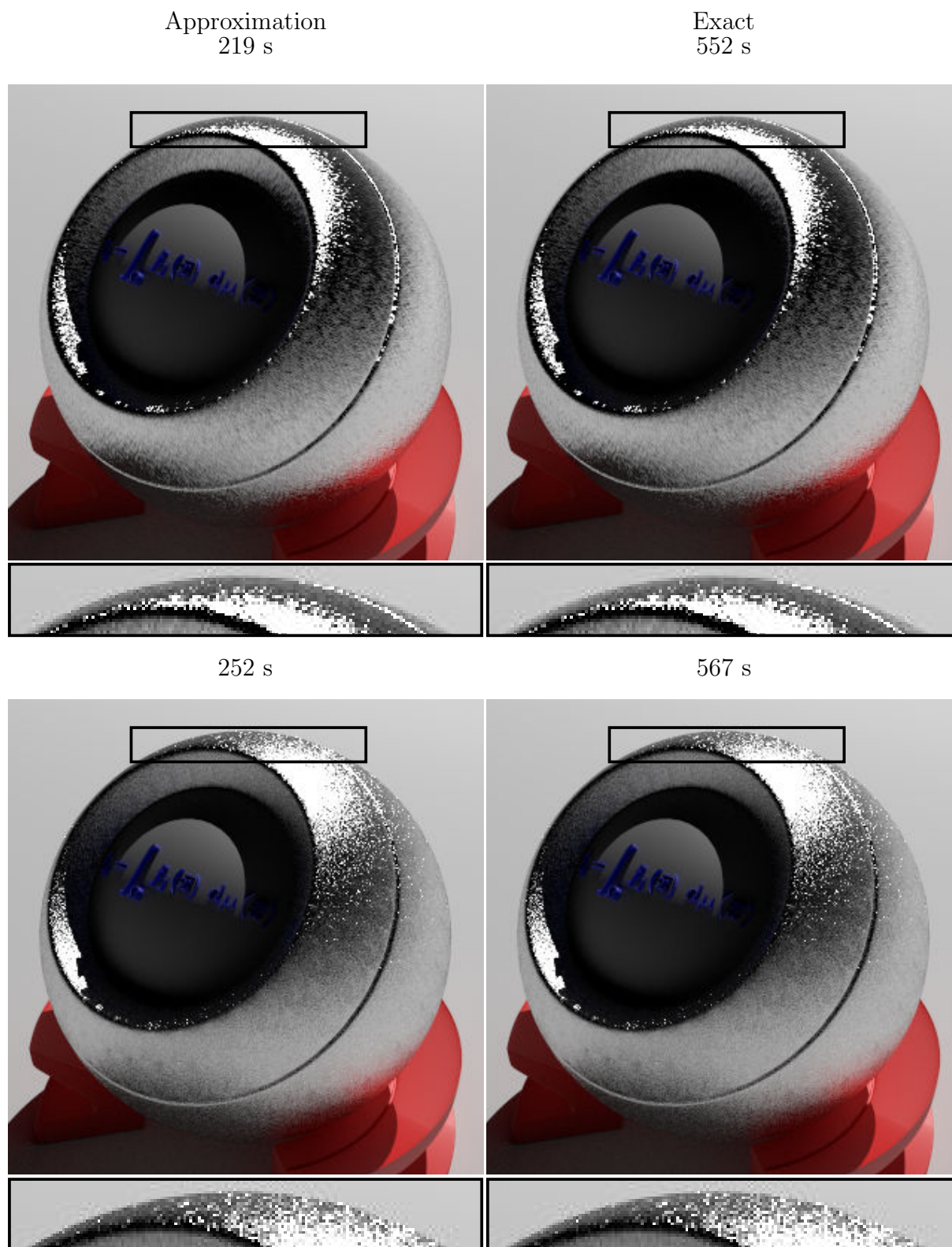


FIGURE 3.7 – Rendus obtenus en utilisant la version approximée (à gauche) et exacte (à droite) du masquage-ombrage et de l'aire projetée. Le matériau est soit anisotrope (en haut) ou isotrope (en bas). L'approximation ne change pas l'apparence de manière significative, mais elle permet une réduction de plus de la moitié du temps de calcul.

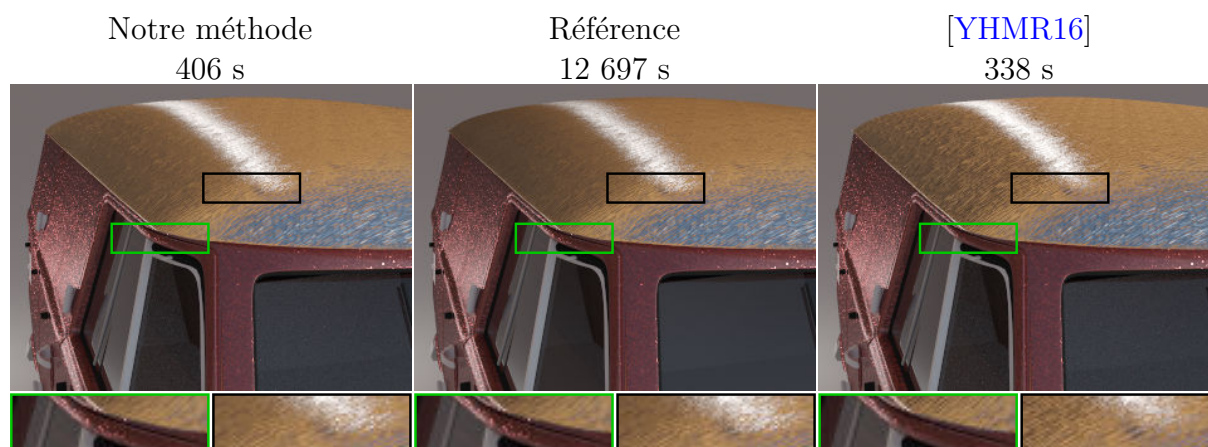


FIGURE 3.8 – Un van avec une peinture métallisée et un toit métallique brossé, rendu avec notre méthode (à gauche, 1 024 epp), rendu de manière classique (au milieu, 65 536 epp) et rendu avec la méthode précédente (à droite, 1 024 epp). Certains reflets lumineux ne sont pas rendus avec la méthode de Yan et al. (encarts verts). Dans l’encart noir, le lobe spéculaire de notre méthode est plus proche de la référence que celui de la méthode précédente.

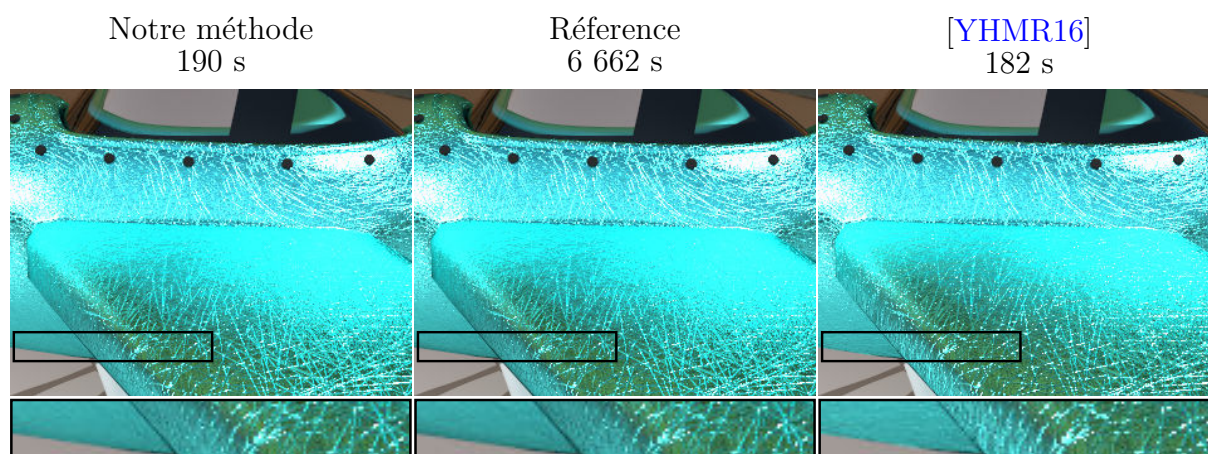


FIGURE 3.9 – Rendus d’un avion métallique rayé et rugueux, où le matériau est modélisé avec notre méthode (à gauche, 1 024 epp), un plaquage de normales classique (au milieu, 65 536 epp) et la méthode précédente (à droite, 1 024 epp). Cette dernière reflète de la lumière à certains endroits (comme dans l’encart) alors que la référence ne le fait pas et inversement. Notre BRDF reproduit plus fidèlement l’apparence de la référence pour un coût additionnel de temps de calcul de seulement 5% pour cette scène.

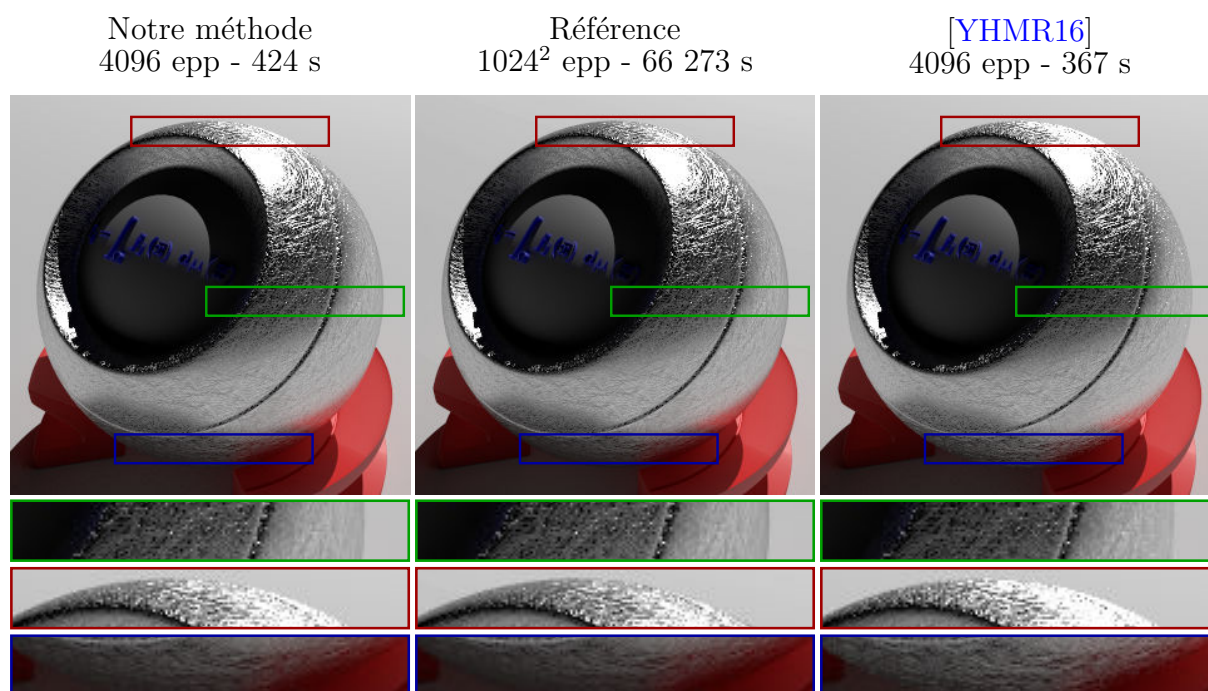


FIGURE 3.10 – Un orbe métallique et rayé illuminé par une lumière ponctuelle. Notre méthode (à gauche) capture mieux l'apparence du matériau de référence (au centre) que la méthode de Yan et al. (à droite).

avec la référence. Notre BRDF permet le rendu de surfaces pailletées (peinture métallisée), brossées, rayées ou rugueuses. Elle possède des hautes fréquences spatiales et angulaires, comme celle des matériaux scintillants visibles dans la vie quotidienne.

La mixture de NDFs utilisée dans notre méthode peut être améliorée. En effet, la formulation pourrait être plus adaptative au contenu fréquentiel local, ce qui aurait pour effet de sauver de la mémoire et de permettre une évaluation plus rapide.

Notre méthode ne gère que le simple rebond dans la micro-surface à cause du terme d'ombrage. Il cause une apparence trop sombre quand la rugosité est élevée. Dans le chapitre suivant, nous proposons une méthode qui corrige cet artéfact de rendu à l'aide d'une modélisation multi-rebonds. Cette modélisation permet aussi de gérer les normales tournant le dos à l'observateur. Ici, dans ce chapitre, elles causent des pertes d'énergie car elles empêchent toute réflexion lumineuse. Les cas les plus défavorables sont les angles rasants, où la moitié des normales peuvent potentiellement tourner le dos à la caméra, donnant des bords sombres aux modèles 3D. Ce problème n'est pas lié directement à la méthode présentée ici, mais plutôt à la représentation classique de la carte de normales. Dans le chapitre suivant, nous utilisons une représentation différente nous permettant, d'une part, de ne plus avoir des normales tournant le dos et, d'autre part, de dériver une \mathcal{P} -BRDF multi-rebonds. L'apparence n'est plus trop sombre et paraît naturelle.

Chapitre 4 :

Rendu de micro-reflets basé sur une BRDF surfacique et multi-rebonds

4.1 Introduction

Les matériaux à paillettes, la peinture métallisée, le métal ou le plastique rugueux, brossé ou rayé sont des surfaces qui sont difficiles à représenter en informatique graphique. Ces matériaux ont de minuscules détails très brillants qui ne sont visibles que sous un éclairage petit et puissant. La propriété principale de ces matériaux est la cohérence spatiale de leur centaine de lobes spéculaires très fins, où de petites translations de la caméra ou de la lumière révèlent la micro-surface.

Utiliser des distributions de normales lisses pour modéliser l'apparence est une pratique commune pour rendre des surfaces spéculaires, car ces fonctions s'intègrent bien dans les modèles d'éclairage standards et les architectures des moteurs de rendu. Elles sont cependant incapables de reproduire des surfaces réelles ayant des microstructures complexes. L'utilisation de cartes de normales permet de résoudre ce problème, mais leur filtrage simplifie la NDF sous pixel en une représentation trop simple qui ne permet pas de retranscrire l'apparence réelle du matériau qu'elle représente.

Pour capturer les reflets sous pixel provoqués par une surface spéculaire, le modèle de réflectance ne doit pas utiliser de BRDFs classiques. Des modèles plus sophistiqués doivent être utilisés, se basant sur des *intégrateurs de micro-reflets*. Cette famille de méthodes ne simplifie pas la micro-surface et garantit le rendu de tous les micro-détails dans un pixel de l'écran. La complexité du matériau est totalement prise en compte et son apparence est reproduite de manière fidèle.

Un intérêt récent. Le rendu de microstructures spéculaires a reçu un intérêt croissant durant les cinq dernières années. Les contributions de Jakob et al. [JHY⁺14] et de Yan et al. [YHJ⁺14] ont inspiré beaucoup de travaux récents explorant les matériaux procéduraux [AK16, ZK16, WWH18], basés sur des primitives [RGB16] ou basés sur des intégrateurs de micro-reflets utilisant des cartes de normales [YHMR16, CCM18, GGN18]. Chaque type de méthode a ses forces et ses faiblesses. Quelques méthodes se spécialisent dans le rendu d'un type particulier de microstructures (rayures, paillettes miroitantes), simulant parfois le multi-rebonds à l'intérieur et proposant des méthodes d'échantillonnage optimales. D'autres sont plus polyvalentes, et peuvent représenter une large gamme de matériaux (surfaces pailletées, brossées et du métal rayé), souvent au prix d'un rendu moins réaliste et d'une consommation mémoire importante.

Problèmes restants. La plupart des intégrateurs de micro-reflets basés sur des cartes de normales ou des surfaces procédurales ne simulent qu'un rebond lumineux dans la micro-surface. Les images rendues avec ces intégrateurs sont anormalement sombres, surtout quand la surface est très rugueuse (figure 4.1, au milieu et à droite). Beaucoup de normales de la surface tournent le dos à l'observateur et à la lumière quand les directions sont rasantes. Ces cas empêchent la lumière d'être reflétée. Un autre problème important est la procédure d'échantillonnage qui est sous-optimale, donnant des poids d'échantillon forts, et donc introduisant des pixels blancs (fireflies en anglais) isolés mais gênants (figure 4.12). Les intégrateurs de micro-reflets les plus récents échantillonnent une normale de la micro-surface sans prendre en compte le masquage et les inter-réflexions qui sont inhérents à la micro-géométrie. Lorsque les directions sont très rasantes, plus de la moitié des échantillons sont perdus et des poids forts compensent ces pertes.

Intégrateur de reflets
multi-rebonds

Chermain et al. [CCM18]

Yan et al. [YHMR16]

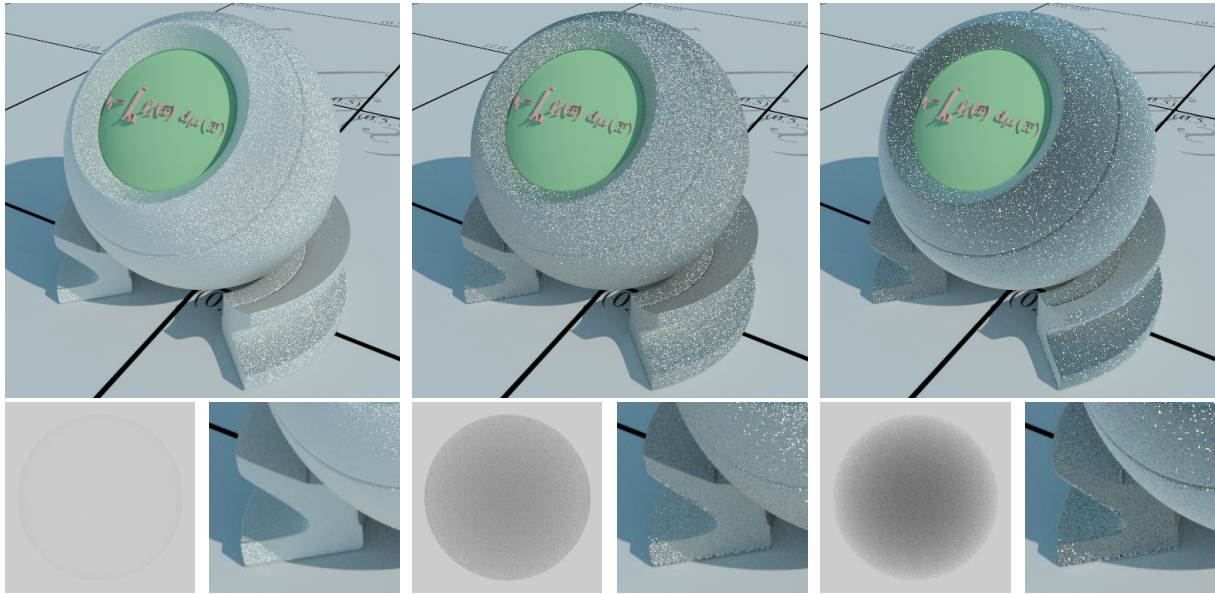


FIGURE 4.1 – Orbe pailleté sous l’éclairage d’une carte d’environnement rendu avec 1 024 échantillons par pixel (epp). La micro-surface est modélisée par une carte de normales spéculaire avec une rugosité RMS (Root Mean Square) forte ($\sigma = 1$). Les méthodes précédentes basées sur un rendu classique de cartes de normales et des évaluations de BRDFs simple rebond noircissent l’apparence et créent même des trous noirs sur la surface (encarts et tests du four blanc). L’échantillonnage préférentiel est sous-optimal et crée des pixels blancs. Notre nouvel intégrateur de micro-reflets multi-rebonds résoud ces problèmes. Il produit des rendus sans artefact, et passe presque totalement le test du four blanc, pour un surcoût de temps de calcul de 36 % dans cette scène.

Résumé et contributions. Le but de ce travail est de résoudre tous ces problèmes. Nous avons choisi de représenter la micro-surface par une *carte de normales*, pour sa polyvalence. Notre représentation est basée sur un modèle *dual*, inspiré du travail de Schüssler et al. [SHHD17] (section 2.5.2). Une différence majeure est que nous utilisons un dual symétrique et non tangent, dans le but de ne pas altérer la rugosité du matériau et son apparence relative. Nous prenons en compte le multi-rebonds en réintroduisant l’énergie perdue par une BRDF simple rebond (quantité que nous dérivons analytiquement pour une surface parfaitement spéculaire). La réintroduction de l’énergie perdue se fait en utilisant une BRDF de compensation d’énergie, qui est ajoutée à l’expression simple rebond, formant finalement notre BRDF *locale*. Pour l’échantillonnage, nous échantillonnons la BRDF simple rebond et nous gardons l’échantillon seulement s’il n’est pas ombré ou transmitt (figure 4.5). Sinon, nous échantillonnons la BRDF de compensation d’énergie qui modélise le multi-rebonds.

Notre modèle local est ensuite utilisé par l’intégrateur de micro-reflets basé sur une empreinte de rayon. Notre \mathcal{P} -BRDF est une somme pondérée de différentes instances de BRDFs locales. Elle conserve l’énergie car c’est aussi le cas des BRDFs locales.

Nos contributions dans le domaine du rendu de micro-reflets sont les suivantes :

- Nous proposons le premier intégrateur de micro-reflets multi-rebonds basé sur une carte de normales.

- La \mathcal{P} -BRDF proposée ne génère pas les problèmes de rendu des méthodes précédentes : trous noirs sur la surface, apparence trop sombre et fireflies.
- L'échantillonnage est optimal : il n'y a pas d'échantillons perdus, la PDF est analytique et le poids des échantillons vaut un si la surface est non absorbante.

Notre représentation n'utilise pas de jacobienne, c'est-à-dire qu'elle ne prend pas en compte la courbure locale de la surface. Elle est asymétrique, et donc son utilisation est limitée à du tracer de chemins (path tracing) unidirectionnels. De plus, la méthode de compensation est restreinte aux cartes de normales très spéculaires : l'utiliser pour faire autre chose que du rendu de micro-reflets n'est pas conseillé. Enfin, l'approche duale combine deux apparences : l'une issue de la carte de normales originale (primale), et l'autre issue de la carte de normales symétrique (duale). Cependant, dans nos résultats, l'apparence finale reste quand même proche de celle obtenue avec la carte de normales primale.

Après une revue détaillée des travaux connexes (section 4.2), nous présentons notre modèle de BRDF local et multi-rebonds dans la section 4.3. Nous utilisons cette BRDF locale pour construire la \mathcal{P} -BRDF dans la section 4.4, qui filtre la surface et garantit la cohérence spatiale des micro-reflets. Nous évaluons notre méthode et discutons des résultats dans la section 4.5.

4.2 Travaux connexes

Notre travail est lié à trois domaines de recherche différents en informatique graphique : les intégrateurs de micro-reflets, le rendu de cartes de normales basé micro-facettes et les BRDFs multi-rebonds. Nous passons en revue les travaux de ces trois domaines.

4.2.1 Intégrateurs de micro-reflets

Les intégrateurs de micro-reflets peuvent être vus comme des types spéciaux de BRDFs où des microstructures spéculaires sont rendues en utilisant soit des primitives géométriques placées aléatoirement sur la surface, soit des cartes de normales haute définition. Au lieu de se baser sur Monte Carlo pour calculer la réflectance (en échantillonnant la micro-surface), les intégrateurs de micro-reflets calculent d'un seul coup toute la radiance quittant une empreinte de rayon de manière beaucoup plus efficace.

Surfaces stochastiques. La première méthode utilisant des microstructures placées aléatoirement est celle de Jakob et al. [JHY⁺14]. Elle est spécialisée dans le rendu de matériaux à paillettes. La micro-surface est composée de micro-paillettes miroitantes, placées aléatoirement sur la surface. Elles sont utilisées pour calculer une \mathcal{P} -NDF discrète qui fait ensuite partie d'une BRDF à micro-facettes. Les résultats donnent des micro-reflets cohérents spatialement. En revanche, leur méthode ne modélise qu'un rebond lumineux, et lorsque nous zoomons fortement, nous pouvons voir des micro-miroirs qui tournent le dos à l'observateur, formant des trous noirs sur la surface. La normalisation de la \mathcal{P} -BRDF est globale et ne dépend pas de l'empreinte du rayon. Pour finir, l'échantillonnage n'est pas optimal, car des réflecteurs non contributifs peuvent être échantillonnés par inadvertance. L'extension proposée par Atanasov et Koylazov [AK16] évite ce problème, mais au prix du parcours de toutes les paillettes contenues dans

l’empreinte, ce qui n’est pas efficace. Certaines méthodes améliorent les performances du modèle [WWH18], d’autres utilisent une formulation différente et sont utilisables en temps-réel [ZK16, WB16], au prix de ne plus être basées physique du tout.

Le travail de Raymond et al. [RGB16] modélise des rayures spéculaires distribuées aléatoirement sur la surface. L’utilisateur a le contrôle sur le profil, l’orientation et la densité des rayures. Les auteurs simulent le multi-rebonds en précalculant la distribution de réflectance d’un profil d’une rayure avec un lancer de rayons 2D. Pour connaître l’aire relative des rayures dans l’empreinte du rayon, ils utilisent un filtrage MIP d’une texture noire et blanche, indiquant les zones avec et sans rayures. Leur méthode donne des résultats réalistes, où le multi-rebonds est pris en compte et où l’échantillonnage est optimal. Leur méthode ne gère cependant que les surfaces rayées.

Intégrateurs de micro-reflets basés cartes de normales. À l’inverse des méthodes utilisant des micro-surfaces procédurales, les intégrateurs de micro-reflets basés sur des cartes de normales donnent plus de flexibilité sur l’apparence du matériau. Le métal ou plastique rayé, brossé ou rugueux et les surfaces pailletées peuvent être modélisés en changeant seulement le contenu de la carte de normales. Le travail de Yan et al. 2014 [YHJ⁺14] a été le premier à introduire cette représentation de la micro-surface, sans la simplifier et obtenant des rendus réalistes d’une large gamme de matériaux. Plusieurs papiers ont amélioré les performances de leur méthode en utilisant soit des mixtures de NDFs 4D [YHMR16] ou des harmoniques sphériques [GGN18]. Le travail du chapitre 3 [CCM18] quant à lui améliore la qualité du rendu en proposant une normalisation relative à l’empreinte du rayon. La procédure d’échantillonnage de ces méthodes est efficace car ces dernières se basent toutes sur la NDF de l’empreinte. Cependant, les informations de visibilité ne sont pas utilisées, ce qui cause des fireflies et génère des chemins ne transportant pas de radiance. La méthode de Gamboa [GGN18] n’utilise pas l’échantillonnage préférentiel et n’a pas ces problèmes. Ils intègrent directement toute la radiance incidente, au prix de pré-calculs coûteux en temps et en mémoire. Malgré tout, toutes ces méthodes ne modélisent qu’un rebond lumineux dans la microstructure, provoquant une apparence globale trop sombre lorsque le matériau est rugueux. Elles utilisent aussi une représentation classique de la carte de normales, où les normales tournant le dos à l’observateur produisent des franges noires sur la surface. Notre travail corrige ces problèmes.

4.2.2 Rendu de cartes de normales basé micro-facettes

Le rendu classique de cartes de normales peut faire apparaître des franges noires sur les surfaces et implique des facteurs de renormalisation locaux [SHHD17] qui deviennent extrêmement complexes lorsque les normales sont filtrées dans l’empreinte du rayon [BN12, DHI⁺13]. Pour éviter ces problèmes, Schüssler et al. [SHHD17] proposent une BRDF basée micro-facettes qui simule des perturbations de normales par rapport à la normale géométrique, en utilisant des normales tangentes pour « fermer » la micro-surface. En un point de la surface, les facettes tangentes rendent la micro-géométrie réellement continue, alors que sans, la surface est constituée de facettes discrètes non reliées (figure 4.2). Ils utilisent des marches aléatoires pour modéliser des rebonds infinis dans la micro-surface, et dérivent une forme close intéressante pour des rebonds d’ordre deux. Ils obtiennent des résultats plus réalistes que le rendu classique de cartes de

normales, exempts de franges noires. Notre modèle s’inspire de leur travail, mais nous utilisons des normales symétriques qui sont plus appropriées pour le rendu de micro-reflets. En effet, les facettes verticales impactent négativement l’apparence car elles introduisent arbitrairement beaucoup plus de rugosité, selon certains angles d’observation (figure 4.3). De plus, nous n’utilisons pas de marches aléatoires coûteuses pour modéliser le multi-rebonds. À la place, nous estimons l’énergie des rebonds d’ordre deux et plus puis nous l’ajoutons dans la BRDF simple rebond à l’aide d’une BRDF de compensation d’énergie. La figure 4.2 donne un aperçu rapide de la différence entre le modèle classique d’une carte de normales, le modèle de Schüssler et al. et le nôtre.

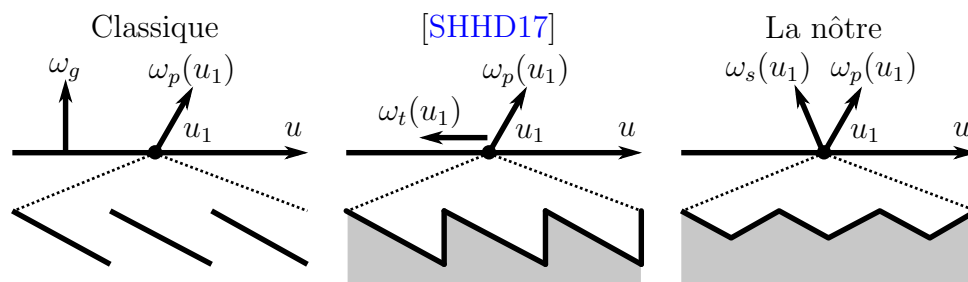


FIGURE 4.2 – À gauche : carte de normales classique. Au milieu : carte de normales de Schüssler et al. [SHHD17]. À droite : notre modèle. Les facettes symétriques ω_s ne changent pas la rugosité de base du matériau, contrairement aux facettes tangentes ω_t . Pour du rendu de micro-reflets où la carte de normales est hautement répétée, les facettes tangentes altèrent trop l’apparence.

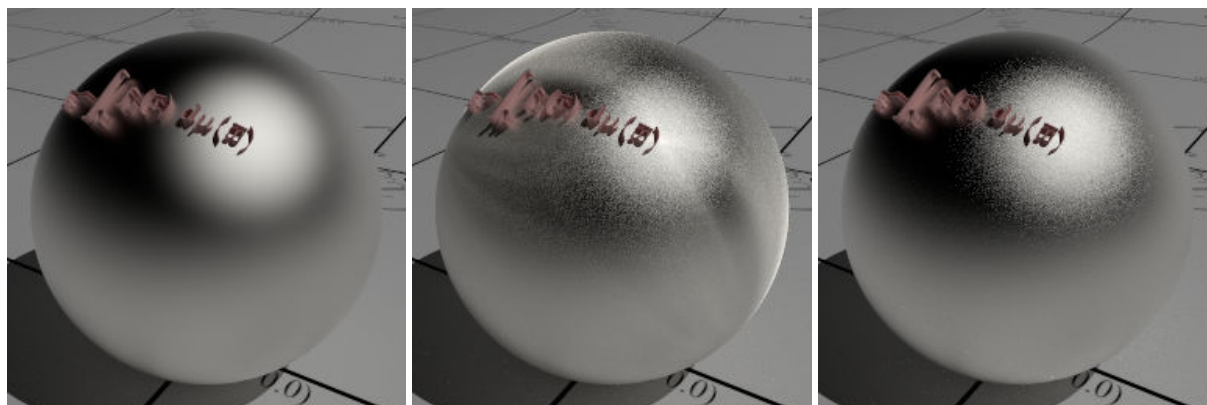


FIGURE 4.3 – Comparaison entre une surface de Beckmann modélisée par une NDF lisse (à gauche), une carte de normales utilisant des facettes tangentes (au milieu, [SHHD17]) et des facettes symétriques (à droite). La carte de normales étant très dense dans cette scène, les facettes tangentes introduisent de la rugosité non voulue comparé au modèle lisse, alors que ce n’est pas le cas de notre modèle. Remarquez aussi que les marches aléatoires, utilisées au milieu et à droite, entraînent des fireflies, même ici avec 262 144 epp.

4.2.3 BRDFs multi-rebonds

Récemment, la prise en charge du multi-rebonds de la lumière dans les micro-surfaces a gagné beaucoup d'intérêt en informatique graphique. C'est en partie grâce au travail de Heitz et al. [HHdD16] sur les BSDFs à base de micro-facettes utilisant le modèle de Smith, intégrant pour la première fois des diffusions multiples dans la micro-géométrie. Les auteurs utilisent des fonctions de réflectance stochastiques qui donnent des résultats correspondant à des données issues d'une simulation brute force. La faiblesse de leur modèle est l'approche stochastique utilisant des marches aléatoires, qui nécessite plusieurs évaluations pour donner des résultats qui ont convergés. Des méthodes plus récentes proposent des modèles multi-rebonds plus rapides à évaluer en utilisant des distributions de V-cavités parfaitement spéculaires [LJJ⁺18, XH18, SPSH19]. Elles restent néanmoins trop coûteuses pour notre objectif, et génèrent des artéfacts d'apparence. Finalement, pour modéliser très rapidement le multi-rebonds, plusieurs travaux réintègrent directement l'énergie perdue par une modélisation simple rebond, en se concentrant soit sur la réciprocité [CK17, Hil18a, FA19], soit sur l'échantillonnage [Tur19]. Nous utilisons aussi la compensation d'énergie, mais dans le contexte du rendu de micro-reflets utilisant des cartes de normales basées micro-facettes. De plus, notre méthode offre une procédure d'échantillonnage optimale et laisse le choix de la BRDF de compensation d'énergie.

4.3 BRDF multi-rebonds locale

Notre modèle de BRDF *surfacique* est une somme pondérée de BRDFs *locales*. Nous commençons par décrire la formulation de la BRDF *locale*.

4.3.1 Aperçu du modèle de micro-surface

Notre modèle de micro-surface est basé sur une carte de normales continue, construite à partir d'une texture discrète dont les valeurs sont interpolées. Chaque position u sur la carte a une normale primale ω_p , qui est *perturbée* par rapport à la normale géométrique ω_g , et une normale duale ω_s . Cette normale duale est la *symétrique* de ω_p par rapport à la normale géométrique ω_g , et l'ensemble forme une V-cavité (figure 4.2). Toutes les V-cavités ont le même matériau presque parfaitement spéculaire, c'est-à-dire qu'elles ont toutes la même micro-BRDF. Dans une V-cavité, nous détectons les situations où de l'énergie est perdue à cause de notre formulation simple rebond, et nous prenons en compte cette perte en évaluant une BRDF additionnelle. Notre modèle tire parti de l'estimateur de Monte Carlo en proposant une méthode d'échantillonnage locale qui est optimale. Localement, à n'importe quelle position u sur la surface, les orientations des micro-normales ω_p et ω_s sont utilisées pour dériver la probabilité qu'un rayon d'observation touche l'une ou l'autre des facettes, en prenant toujours en compte le masquage induit par l'autre facette.

4.3.2 Distribution de normales et aires projetées

Les distributions de normales sont largement utilisées pour modéliser des micro-surfaces. Elles donnent la densité de micro-normales ω_m à une position u . Dans notre cas, cette fonction est composée de deux fonctions de Dirac centrées sur la normale perturbée

ω_p et la normale symétrique ω_s (la dépendance à la position u est implicite pour le reste de la section) :

$$D(\omega_m) = \frac{1}{2} \frac{\delta_{\omega_p}(\omega_m)}{\omega_p \cdot \omega_g} + \frac{1}{2} \frac{\delta_{\omega_s}(\omega_m)}{\omega_s \cdot \omega_g}. \quad (4.3.1)$$

Cette distribution de normales est celle d'une V-cavité. L'intégrale des aires projetées des deux micro-normales sur la surface macroscopique donne exactement un [CT81, Hei14]. Les aires projetées d'une facette et de sa duale vers la direction d'observation ω_o sont

$$a_p(\omega_o) = \frac{\langle \omega_o, \omega_p \rangle}{\langle \omega_p, \omega_g \rangle}, \quad a_s(\omega_o) = \frac{\langle \omega_o, \omega_s \rangle}{\langle \omega_s, \omega_g \rangle}. \quad (4.3.2)$$

À partir de ces définitions, nous pouvons dériver les probabilités $\lambda_p(\omega_o)$ et $\lambda_s(\omega_o)$ qu'un rayon d'observation touche ω_p ou ω_s (figure 4.4) :

$$\lambda_p(\omega_o) = \frac{a_p(\omega_o)}{a_p(\omega_o) + a_s(\omega_o)}, \quad \lambda_s(\omega_o) = \frac{a_s(\omega_o)}{a_p(\omega_o) + a_s(\omega_o)}. \quad (4.3.3)$$

Les termes λ_p et λ_s seront utilisés pour évaluer (section 4.3.4) et échantillonner (section 4.3.6) la BRDF.

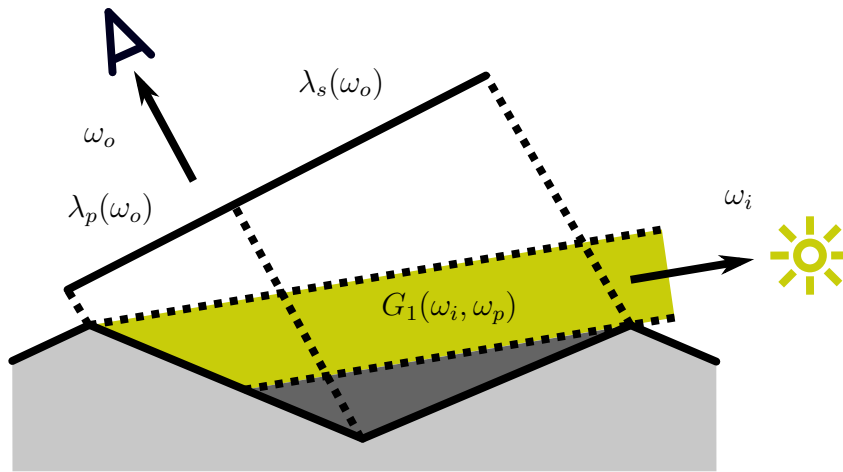


FIGURE 4.4 – Illustration des probabilités d'intersection et de la fonction d'ombrage.

4.3.3 Masquage et ombrage

La fonction de masquage donne la proportion de la micro-facette ω_p ou ω_s qui est visible depuis la direction d'observation ω_o (section 2.4.10). Elle est appelée la fonction d'ombrage quand elle est évaluée avec la direction incidente ω_i comme paramètre (figure 4.4). La

fonction de masquage pour une V-cavité est [CT81, Hei14] :

$$\begin{aligned} G_1(\omega_o, \omega_p) &= H(\langle \omega_o, \omega_p \rangle) \min \left(1, \frac{\langle \omega_o, \omega_g \rangle}{\frac{1}{2}a_p(\omega_o) + \frac{1}{2}a_s(\omega_o)} \right) \\ &= H(\langle \omega_o, \omega_p \rangle) \min \left(1, \frac{2\langle \omega_o, \omega_g \rangle \langle \omega_p, \omega_g \rangle}{\langle \omega_o, \omega_p \rangle + \langle \omega_o, \omega_s \rangle} \right) \end{aligned} \quad (4.3.4)$$

pour la facette orientée selon ω_p . Pour la micro-normale ω_s , elle est définie comme suit :

$$G_1(\omega_o, \omega_s) = H(\langle \omega_o, \omega_s \rangle) \min \left(1, \frac{2\langle \omega_o, \omega_g \rangle \langle \omega_s, \omega_g \rangle}{\langle \omega_o, \omega_p \rangle + \langle \omega_o, \omega_s \rangle} \right). \quad (4.3.5)$$

Comme dans la méthode de Schüssler et al. [SHHD17], nous supposons qu'il n'y a pas de corrélation entre le masquage et l'ombrage, ce qui nous permet d'utiliser la forme séparable simple de la fonction de masquage-ombrage (ici pour la micro-normale ω_p) :

$$G_2(\omega_o, \omega_i, \omega_p) = G_1(\omega_o, \omega_p) G_1(\omega_i, \omega_p). \quad (4.3.6)$$

Ce choix a une implication importante pour la procédure d'échantillonnage préférentiel (section 4.3.6) car nous devons déterminer quand un rayon est ombré, et avec cette supposition, nous pouvons directement utiliser la fonction d'ombrage G_1 au lieu de calculer des intersections explicites et coûteuses entre un rayon et une facette [SHHD17].

4.3.4 BRDF simple rebond

Dans le contexte du rendu de micro-reflets, le matériau des deux facettes est très spéculaire. En supposant que les deux facettes de notre micro-surface ont la même micro-BRDF f_m , la BRDF simple rebond de notre V-cavité pondérée par le cosinus est

$$\begin{aligned} f_1(\omega_o, \omega_i, \omega_p) \langle \omega_i, \omega_g \rangle &= \lambda_p(\omega_o) f_m(\omega_o, \omega_i) \langle \omega_i, \omega_p \rangle G_1(\omega_i, \omega_p) \\ &\quad + \lambda_s(\omega_o) f_m(\omega_o, \omega_i) \langle \omega_i, \omega_s \rangle G_1(\omega_i, \omega_s). \end{aligned} \quad (4.3.7)$$

L'expression f_1 est le résultat de l'intégrale des contributions de chaque micro-normale ω_m (ici il y en a deux) sur l'hémisphère Ω [SHHD17]. Les fonctions d'ombrage et les probabilités d'intersection de chaque facette (figure 4.4) pondèrent la micro-BRDF f_m centrée sur ω_p ou ω_s . Il faut noter que f_1 ne crée pas d'énergie et est réciproque tant que f_m l'est aussi.

4.3.5 BRDF multi-rebonds

Dans cette section, nous montrons comment estimer f_∞ , la BRDF multi-rebonds, pour une seule V-cavité. Sa définition est basée sur f_1 et une BRDF de compensation d'énergie f_{2+} qui représente les rebonds d'ordre deux et plus. Nous estimons l'énergie $1 - E_1$ allant dans cette dernière, et nous pondérons son évaluation avec cette valeur.

Terme d'énergie. En supposant que la micro-BRDF f_m est parfaitement spéculaire et non absorbante, l'albedo directionnel de f_1 , appelé aussi le terme d'énergie, est

$$\begin{aligned} E_1(\omega_o, \omega_p) &= \int_{\Omega} f_1(\omega_o, \omega_i, \omega_p) \langle \omega_i, \omega_g \rangle d\omega_i \\ &= \lambda_p(\omega_o) G_1(\omega_r(\omega_o, \omega_p), \omega_p) + \lambda_s(\omega_o) G_1(\omega_r(\omega_o, \omega_s), \omega_s) \end{aligned} \quad (4.3.8)$$

où $\omega_r(\omega_1, \omega_2)$ est la direction réfléchiée de ω_1 par rapport à la normale ω_2 . Le terme E_1 donne la radiance qui s'échappe de la micro-surface après un rebond de lumière, dans la direction ω_o . Pour la BRDF f_1 , cette valeur varie entre 0 et 1 pour différents couples de (ω_o, ω_p) .

Dans le but d'arriver à une conservation totale de l'énergie, nous définissons notre BRDF locale et multi-rebonds comme

$$f_{\infty}(\omega_o, \omega_i, \omega_p) = f_1(\omega_o, \omega_i, \omega_p) + (1 - E_1(\omega_o, \omega_p)) f_{2+}(\omega_o, \omega_i). \quad (4.3.9)$$

L'intégrale sur toutes les directions incidentes de cette BRDF pondérée par le cosinus est égale à un :

$$\begin{aligned} \int_{\Omega} f_{\infty}(\omega_o, \omega_i, \omega_p) \langle \omega_i, \omega_g \rangle d\omega_i &= E_1(\omega_o, \omega_p) + (1 - E_1(\omega_o, \omega_p)) \int_{\Omega} f_{2+}(\omega_o, \omega_i) \langle \omega_i, \omega_g \rangle d\omega_i \\ &= 1 + E_1(\omega_o, \omega_p) - E_1(\omega_o, \omega_p) = 1, \end{aligned} \quad (4.3.10)$$

seulement quand l'intégrale de la BRDF de compensation d'énergie $f_{2+}(\omega_o, \omega_i) \langle \omega_i, \omega_g \rangle$ donne aussi un. Le choix de f_{2+} est arbitraire et est discuté dans la section 4.5.2. Notre formulation pour f_{∞} a une forme close simple, et peut être évaluée efficacement. Cependant, elle est asymétrique (non réciproque) comme nous pouvons le voir dans l'équation 4.3.9 où échanger ω_o et ω_i donne un terme E_1 qui est différent ($E_1(\omega_o, \omega_p) \neq E_1(\omega_i, \omega_p)$). C'est un problème seulement si nous utilisons un algorithme de rendu bidirectionnel. Dans nos résultats utilisant du lancer de rayons unidirectionnels, nous n'avons pas remarqué d'artéfacts visuels causés par cette asymétrie.

En pratique, nous mollifions la distribution de Dirac contenue dans f_m en une NDF avec une rugosité minuscule α_m [YHJ⁺14, YHMR16, CCM18] dans le but d'évaluer f_1 (équation 4.3.7). Il faut noter que le terme d'énergie de l'équation 4.3.8, utilisant le Dirac, permet quand même la conservation totale de l'énergie de f_{∞} lorsque l'évaluation de f_1 est faite avec un Dirac mollifié. La section 4.5.1 montre que f_{∞} passe le test du four blanc quand la NDF de f_m a une rugosité minuscule, ce qui permet de valider cette approximation de la fonction de Dirac.

Terme de Fresnel. Le modèle présenté ci-dessus n'incorpore pas les effets de Fresnel qui donnent aux objets leurs couleurs et qui, dans le cas du multi-rebonds, sont responsables de la saturation des couleurs sur des surfaces rugueuses. Nous devons choisir le coefficient d'absorption de la BRDF de compensation d'énergie f_{2+} , mais il n'est pas trivial à définir. Nous choisissons de l'approximer avec le terme de Fresnel F_1 de la micro-BRDF f_m . Ainsi, la BRDF multi-rebonds f_{∞} pour une carte de normales absorbante est maintenant

$$f_{\infty}(\omega_o, \omega_i, \omega_p) = f_1(\omega_o, \omega_i, \omega_p) + (1 - E_1(\omega_o, \omega_p)) F_1(\omega_o, \omega_g) f_{2+}(\omega_o, \omega_i), \quad (4.3.11)$$

où $F_1(\omega_o, \omega_g)$ colore la BRDF non absorbante f_{2+} de la même manière que f_1 . Notez que Turquin [Tur19] utilise une approximation similaire pour ses BRDFs à micro-facettes compensées et obtient des résultats similaires à la référence [HHdD16].

4.3.6 Échantillonnage préférentiel de la BRDF locale

Pour échantillonner de manière préférentielle f_∞ , nous échantillons d'abord la BRDF simple rebond f_1 en choisissant une facette selon les probabilités d'intersection $\lambda_p(\omega_o)$ et $\lambda_s(\omega_o)$. Ensuite, une direction ω_i est choisie en échantillonnant par importance la micro-BRDF f_m de la micro-normale sélectionnée. Si la direction échantillonnée ω_i est ombrée (figure 4.5, milieu) ou se dirige vers l'intérieur de la surface (figure 4.5, droite), alors du multi-rebonds se produit et donc nous échantillons f_{2+} . La probabilité que l'échantillonnage de la BRDF simple rebond f_1 échoue, c'est-à-dire que ω_i est ombrée ou transmise, vaut exactement $1 - E_1(\omega_o, \omega_p)$. Notre procédure d'échantillonnage est résumée dans l'algorithme 1 (où \mathcal{U} est un nombre aléatoire uniforme tiré dans l'intervalle $[0, 1[$) et est illustrée dans la figure 4.5. Elle retourne des poids d'échantillonnage w entre 0 et 1, si c'est le cas de f_m et f_{2+} . Les poids seront toujours égaux à 1 s'il n'y a pas d'effets de Fresnel et en considérant que f_m a une rugosité égale à 0. Sinon, le poids w est multiplié par le terme de Fresnel et peut être très légèrement inférieur à 1 s'il y a une rugosité très faible (ce qui est notre cas). En revanche, des poids larges ne peuvent pas être retournés et les fireflies sont évités.

Algorithm 1: Échantillonnage de $f_\infty(\omega_o, \omega_i, \omega_p)\langle\omega_i, \omega_g\rangle$

```

 $\omega_m \leftarrow [\mathcal{U} < \lambda_p(\omega_o) ? \omega_p : \omega_s]$  // Choix d'une facette
 $(\omega_i, w) \leftarrow$  échantillonnage de  $f_m(\omega_o, \omega_i, \omega_m)\langle\omega_i, \omega_m\rangle$  // Échantillonnage de  $f_m$ 
if  $\omega_i \cdot \omega_g \leq 0$  ou  $\mathcal{U} \geq G_1(\omega_i, \omega_m)$  then
    /* Le rayon va dans la surface ou il est ombré */
     $(\omega_i, w) \leftarrow$  échantillonnage de  $f_{2+}(\omega_o, \omega_i)\langle\omega_i, \omega_g\rangle$  // Échantillonnage de  $f_{2+}$ 
end
    
```

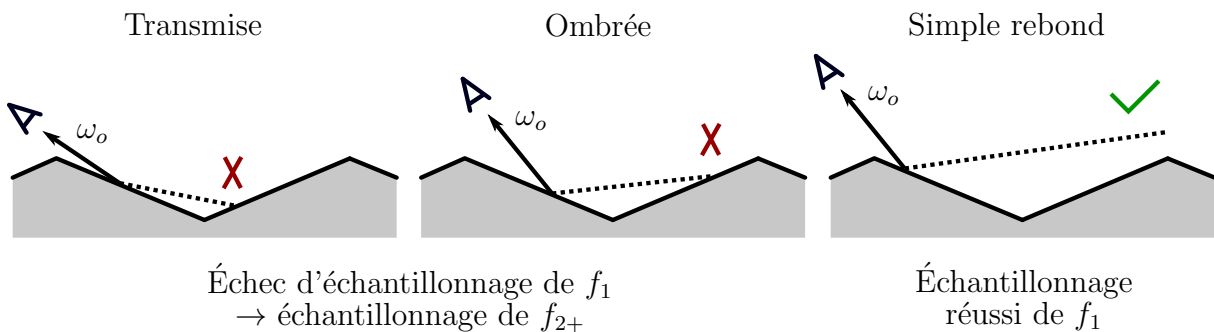


FIGURE 4.5 – Différents cas d'échantillonnage de la BRDF multi-rebonds locale.

4.4 BRDF surfacique multi-rebonds

Dans cette section, nous introduisons le cadre théorique nécessaire pour définir et évaluer des BRDFs surfaciques (\mathcal{P} -BRDFs) basées sur une carte de normales. Nous expliquons aussi comment calculer efficacement notre \mathcal{P} -BRDF basée sur la BRDF locale multi-rebonds f_∞ introduite dans la section précédente. Nous clôturons cette section par la procédure d'échantillonnage.

4.4.1 Définition de la BRDF surfacique

Une \mathcal{P} -BRDF calcule la densité de radiance sortante d'une surface géométrique plane \mathcal{P} , au lieu d'un point u . Le patch \mathcal{P} représente l'empreinte de rayon. C'est un filtre passe-bas normalisé $k_{\mathcal{P}}$, c'est-à-dire qu'il permet l'anti-aliasing de la radiance des reflets de l'empreinte, et que $\int_{\mathbb{R}^2} k_{\mathcal{P}}(u) du = 1$. Plusieurs techniques de filtrage de surface [Ige99, SW01, BYRN17] donnent des méthodes pour construire $k_{\mathcal{P}}$ (section 2.6.1) durant l'algorithme du tracer de chemins (path tracing).

Notre \mathcal{P} -BRDF $f_{\mathcal{P}}$ est définie comme l'intégrale des BRDFs locales pondérées par l'empreinte du rayon $k_{\mathcal{P}}(u)$ à chaque position u :

$$f_{\mathcal{P}}(\omega_o, \omega_i) = \int_{\mathbb{R}^2} f(\omega_o, \omega_i, \omega_p(u)) k_{\mathcal{P}}(u) du \quad (4.4.1)$$

où chaque $f(\omega_o, \omega_i, \omega_p(u))$ est la BRDF de la surface perturbée par $\omega_p(u)$. Il faut noter que cette définition est très différente des précédentes [JHY⁺14, YHJ⁺14, YHMR16, CCM18, GGN18] où la \mathcal{P} -BRDF utilise une \mathcal{P} -NDF (comme dans le chapitre 3). Notre formulation est beaucoup plus simple, et n'est possible que grâce au recentrage de chaque normale perturbée. Si la BRDF locale conserve l'énergie à 100%, est réciproque et s'échantillonne de manière optimale, alors la \mathcal{P} -BRDF a aussi ces trois propriétés.

4.4.2 Évaluation de la BRDF surfacique

En remplaçant f par notre BRDF locale multi-rebonds f_∞ perturbée par $\omega_p(u)$, nous obtenons

$$\begin{aligned} f_{\mathcal{P}}(\omega_o, \omega_i) &= \int_{\mathbb{R}^2} f_\infty(\omega_o, \omega_i, \omega_p(u)) k_{\mathcal{P}}(u) du \\ &= f_1(\omega_o, \omega_i, \mathcal{P}) + E_{2+}(\omega_o, \mathcal{P}) F_1(\omega_o, \omega_g) f_{2+}(\omega_o, \omega_i). \end{aligned} \quad (4.4.2)$$

Le terme de Fresnel $F_1(\omega_o, \omega_g)$ et la BRDF de compensation d'énergie $f_{2+}(\omega_o, \omega_i)$ peuvent être sortis de l'intégrale car ces termes ne dépendent pas de u . Le terme $f_1(\omega_o, \omega_i, \mathcal{P})$ est la BRDF surfacique simple rebond, défini comme suit :

$$f_1(\omega_o, \omega_i, \mathcal{P}) = \int_{\mathbb{R}^2} f_1(\omega_o, \omega_i, \omega_p(u)) k_{\mathcal{P}}(u) du. \quad (4.4.3)$$

Le terme $E_{2+}(\omega_o, \mathcal{P})$ est l'énergie perdue par la BRDF simple rebond f_1 dans l'empreinte, et son expression est

$$\begin{aligned} E_{2+}(\omega_o, \mathcal{P}) &= \int_{\mathbb{R}^2} 1 - E_1(\omega_o, \omega_p(u)) k_{\mathcal{P}}(u) du \\ &= 1 - \int_{\mathbb{R}^2} E_1(\omega_o, \omega_p(u)) k_{\mathcal{P}}(u) du. \end{aligned} \quad (4.4.4)$$

Nous développons maintenant comment calculer efficacement $f_1(\omega_o, \omega_i, \mathcal{P})$ et $E_{2+}(\omega_o, \mathcal{P})$.

4.4.2.1 Évaluation de la BRDF surfacique simple rebond

Jusqu'à maintenant, notre modèle, basé sur l'équation 4.4.3, utilise une infinité de BRDFs locales. En pratique, nous utilisons un ensemble discret de BRDFs locales en supposant qu'il existe un ensemble fini de m normales perturbées ω_p :

$$f_1(\omega_o, \omega_i, u) = \sum_j^m f_1(\omega_o, \omega_i, \omega_p(u_j)) k_j(u). \quad (4.4.5)$$

Des positions discrètes u_j séparées par une distance h sont distribuées uniformément dans la carte de normales. Chaque BRDF discrète définie à chaque position u_j est pondérée par sa distance à u pour garantir des transitions lisses entre deux BRDFs locales adjacentes. Pour avoir une pondération analytique relativement simple de f_1 , nous utilisons une empreinte $k_{\mathcal{P}}$ et un poids de BRDF k_j gaussiens. La déviation standard de k_j est fixée à $\sigma_h = h/\sqrt{8 \log 2}$, comme dans la méthode du chapitre 3 et de Yan et al. [YHMR16] dans le but d'avoir une contribution identique entre deux évaluations adjacentes de BRDF discrète lorsque le point d'évaluation u est à mi-chemin entre les deux (figure 4.6).

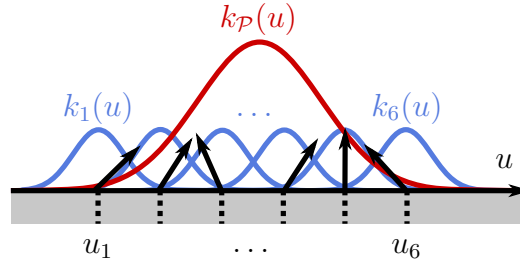


FIGURE 4.6 – Un ensemble discret de BRDFs pondérées par une gaussienne k_j , et leur normale associée $\omega_p(u_j)$.

En utilisant l'équation 4.4.5 (utilisant la somme discrète) dans l'équation 4.4.3 qui est la BRDF surfacique simple rebond, nous obtenons

$$\begin{aligned} f_1(\omega_o, \omega_i, \mathcal{P}) &= \int_{\mathbb{R}^2} \sum_j^m f_1(\omega_o, \omega_i, \omega_p(u_j)) k_j(u) k_{\mathcal{P}}(u) du \\ &= \sum_j^m f_1(\omega_o, \omega_i, \omega_p(u_j)) \int_{\mathbb{R}^2} k_j(u) k_{\mathcal{P}}(u) du \\ &= \sum_j^m f_1(\omega_o, \omega_i, \omega_p(u_j)) W_j \end{aligned} \quad (4.4.6)$$

où le poids W_i est la solution analytique de l'intégrale du produit de deux gaussiennes, ici le poids local k_j et l'empreinte du rayon $k_{\mathcal{P}}$.

Hiérarchie accélératrice. Comme expliqué dans la section 4.3, la BRDF locale f_1 a deux facettes symétriques avec un matériau f_m presque parfaitement spéculaire. Cette micro-BRDF est donc une fonction de Dirac mollifiée en une NDF avec une rugosité minuscule α_m . Cette NDF est utilisée dans une BRDF à base de micro-facettes f_m . Cela implique que f_1 a une valeur différente de zéro seulement quand le vecteur à mi-chemin $\omega_h = (\omega_o + \omega_i)/\|\omega_o + \omega_i\|$ vaut approximativement la normale perturbée ω_p ou sa symétrique ω_s . Nous pouvons utiliser cette propriété pour accélérer l'évaluation de la somme de l'équation 4.4.6 en élaguant les BRDFs discrètes non-contribuantes.

Une hiérarchie de volumes englobants 4D (BVH 4D) est construite en utilisant l'ensemble discret de normales perturbées ω_p . Chaque normale a une position u_j , une pente \tilde{p}_j et une boîte englobante 4D définissant le domaine (u, \tilde{p}) où la facette perturbée de f_1 reflète une quantité non-négligeable de lumière, pour un couple de position u et de pente de vecteur à mi-chemin \tilde{h} . La boîte englobante 4D est $(u_j \pm 3\sigma_h, \tilde{p}_j \pm 3\sigma_m)$, où $\sigma_m = \alpha_m/\sqrt{2}$, en supposant que la contribution vaut zéro après 3σ . Cette hiérarchie est très similaire à celle du chapitre 3 et de Yan et al. [YHMR16]. Par contre, l'utilisation d'une facette symétrique oblige à intersecter le BVH 4D deux fois : une fois avec (u, \tilde{h}) , comme dans les méthodes précédentes, mais aussi avec $(u, -\tilde{h})$ pour prendre en compte la contribution des micro-normales symétriques ω_s .

4.4.2.2 Énergie perdue dans la surface

Évaluer l'énergie perdue dans le patch requiert le parcours de toutes les normales perturbées ω_p se trouvant dans l'empreinte du rayon. Pour des raisons de performance, nous supposons que les normales ω_p suivent une distribution gaussienne D ayant une rugosité anisotrope $(\alpha_x(\mathcal{P}), \alpha_y(\mathcal{P}))$. Ces rugosités sont issues de l'écart type des pentes des normales dans l'empreinte de rayon $k_{\mathcal{P}}$ [DHI+13]. En d'autres termes, nous changeons le domaine d'intégration, depuis des positions u vers des normales ω_p , de manière similaire à Heitz et al. [HNPN13] :

$$\begin{aligned} E_{2+}(\omega_o, \mathcal{P}) &= 1 - \int_{\mathbb{R}^2} E_1(\omega_o, \omega_p(u)) k_{\mathcal{P}}(u) du \\ &\approx 1 - \int_{\Omega} E_1(\omega_o, \omega_p) D(\omega_p, \alpha_x(\mathcal{P}), \alpha_y(\mathcal{P})) d\omega_p \\ &\approx 1 - \left(\cos^2(\phi_o) \int_{\Omega} E_1(\theta_o, \omega_p) D(\omega_p, \alpha_x(\mathcal{P})) d\omega_p \right. \\ &\quad \left. + \sin^2(\phi_o) \int_{\Omega} E_1(\theta_o, \omega_p) D(\omega_p, \alpha_y(\mathcal{P})) d\omega_p \right). \end{aligned} \quad (4.4.7)$$

Les termes θ_o et ϕ_o sont respectivement l'angle polaire et azimutal de la direction d'observation ω_o . Nous utilisons ϕ_o pour pondérer deux intégrales hémisphériques impliquant une distribution isotrope $D(\omega_p, \alpha)$ de normales ω_p . En faisant ainsi, nous pouvons précalculer des intégrales de la forme $\int_{\Omega} E_1(\theta_o, \omega_p) D(\omega_p, \alpha) d\omega_p$, et stocker leurs valeurs dans des petits tableaux 2D (128×128 floats, figure 4.7) pour différents couples de rugosités α et d'angles polaires. Pour calculer $\alpha_x(\mathcal{P})$ et $\alpha_y(\mathcal{P})$ lors du rendu, nous

utilisons la méthode décrite dans LEADR [DHI+13], basée sur les moments d'ordre un et deux d'une carte de pentes.

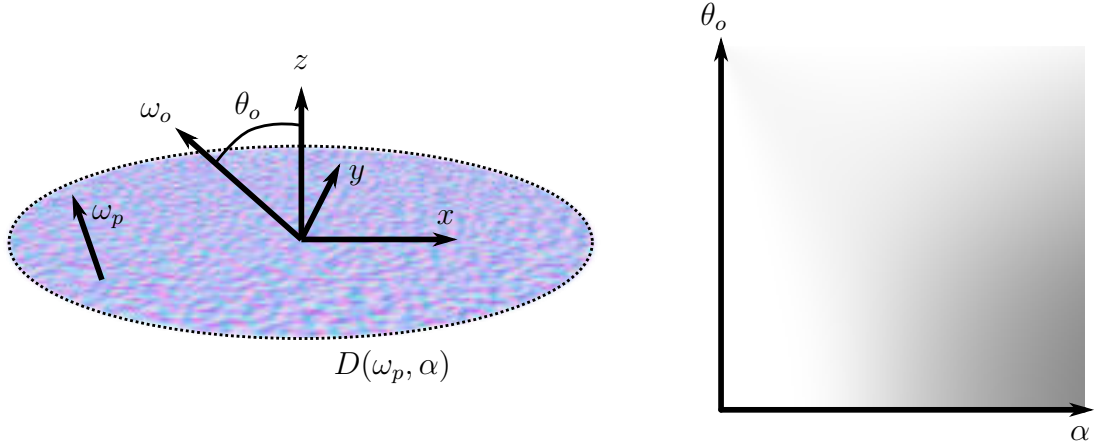


FIGURE 4.7 – Soit $D(\omega_p, \alpha)$ une distribution de normales gaussienne avec une rugosité isotrope α dans l'espace des normales ω_p . La NDF D approxime la distribution exacte des normales dans l'empreinte du rayon. Nous l'utilisons pour précalculer des termes d'énergies surfaciques $\int_{\Omega} E_1(\theta_o, \omega_p) D(\omega_p, \alpha) d\omega_p$. Les valeurs sont stockées dans un tableau 2D, reliant des quantités d'énergie à des paires de valeurs (α, θ_o) .

4.4.3 Échantillonnage de la BRDF surfacique

L'échantillonnage préférentiel est utilisé pour résoudre efficacement l'équation surfacique de réflexion :

$$\begin{aligned} L(\omega_o) &= \int_{\Omega} L(\omega_i) f_{\mathcal{P}}(\omega_o, \omega_i) \langle \omega_i, \omega_g \rangle d\omega_i \\ &= \int_{\Omega} L(\omega_i) \int_{\mathbb{R}^2} f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle k_{\mathcal{P}}(u) du d\omega_i \end{aligned} \quad (4.4.8)$$

où $L(\omega_i)$ est la radiance incidente depuis une direction ω_i . L'échantillonnage de la BRDF surfacique $f_{\mathcal{P}}(\omega_o, \omega_i)$ est fait en échantillonnant un couple composé d'une position et d'une direction (u, ω_i) en suivant la fonction de densité de probabilité suivante (PDF) :

$$\text{PDF}(\omega_i, u) = f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle k_{\mathcal{P}}(u). \quad (4.4.9)$$

La variable aléatoire u est d'abord tirée, suivie par la variable aléatoire ω_i sachant u . Pour la première étape, la densité marginale est

$$\begin{aligned} \text{PDF}(-, u) &= \int_{\Omega} f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle d\omega_i k_{\mathcal{P}}(u) \\ &= E(\omega_o, \omega_p) k_{\mathcal{P}}(u) \end{aligned} \quad (4.4.10)$$

où E est le terme d'énergie. Cette équation se simplifie en $k_{\mathcal{P}}(u)$ seulement si la BRDF locale f conserve l'énergie à 100 %, c'est-à-dire si $\int_{\Omega} f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle d\omega_i = 1$ pour tous les $u \in \mathbb{R}^2$. Comme notre BRDF locale honore cette condition, l'échantillonnage de u

consiste simplement à échantillonner l’empreinte du rayon $k_{\mathcal{P}}(u)$. Pour la seconde étape, la densité conditionnelle de ω_i sachant u est

$$\text{PDF}(\omega_i|u) = \frac{f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle}{\int_{\mathbb{R}^2} E(\omega_o, \omega_p) k_{\mathcal{P}}(u) du}. \quad (4.4.11)$$

Si la BRDF locale est multi-rebonds ($E(\omega_o, \omega_p) = 1$), nous avons

$$\begin{aligned} \text{PDF}(\omega_i|u) &= \frac{f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle}{\int_{\mathbb{R}^2} k_{\mathcal{P}}(u) du} \\ &= f(\omega_o, \omega_i, \omega_p(u)) \langle \omega_i, \omega_g \rangle, \end{aligned} \quad (4.4.12)$$

où ici u est la position échantillonnée durant la première étape.

À notre connaissance, il n’existe pas de procédure d’échantillonnage de \mathcal{P} -BRDF aussi simple et efficace. Les autres méthodes ignorent la direction d’observation lors de l’échantillonnage [YHJ⁺14, JHY⁺14, YHMR16, CCM18], ou doivent parcourir toutes les normales si elles souhaitent le faire [AK16] (figure 4.8), ce qui est inefficace.

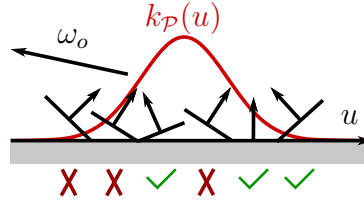


FIGURE 4.8 – Carte de normales classique. L’échantillonnage des normales visibles ne peut être fait que si toutes les normales sont examinées une à une dans l’empreinte du rayon. Notre méthode n’a pas besoin de toutes les parcourir.

4.5 Résultats et discussions

Dans cette section, nous évaluons notre méthode et nous la comparons avec les précédentes. Tous nos résultats ont été rendus avec PBRT-v3 [PJH16] sur un ordinateur avec un système Intel Xeon Gold 6138 biprocesseur. Les figures 4.3 et 4.9 n’utilisent pas d’empreintes de rayon pour faire de l’anti-aliasing, alors que les autres figures les utilisent. Après un premier rebond de la lumière, nous ne calculons plus d’empreintes et nous utilisons une BRDF lisse utilisant la rugosité globale et le terme de Fresnel de la carte de normales, comme dans Jakob et al. [JHY⁺14]. Nous utilisons uniquement du tracer de chemins unidirectionnels car notre BRDF n’est pas réciproque. Toutes les figures utilisent $h = 1$ (section 4.4.2.1), à l’exception des figures 4.14 et 4.15 (partie avec des rayures et utilisant notre méthode, où $h = 0,5$).

4.5.1 Validation de la BRDF locale

Dans cette section, nous validons notre BRDF locale multi-rebonds f_{∞} . Pour cela, nous comparons les rendus et les tests du four blanc de cartes de normales classiques et basées micro-facettes. Dans la figure 4.9, la carte de normales est une surface de Beckmann [Hei15] ($\alpha = 0.5$). Nous comparons f_1 et f_{∞} à une carte de normales classique. Les

franges noires sont visibles sur la surface avec cette dernière méthode. La BRDF simple rebond f_1 élimine partiellement ces artéfacts alors que la BRDF multi-rebonds f_∞ les fait complètement disparaître. De plus, la BRDF f_∞ passe le test du four blanc. Nous utilisons les paramètres suivants : f_m est une BRDF classique à base de micro-facettes avec une NDF de Beckmann ($\alpha_m = 0,02$). Son terme de Fresnel F_1 modélise un matériau en or. La BRDF de compensation d'énergie f_{2+} est lambertienne (diffuse), avec un albedo égal à F_1 . Notez que la compensation diffuse et sa coloration par F_1 reste assez naturelle et plausible pour simuler du multi-rebonds dans la micro-surface. Pour tous les autres résultats, nous utilisons aussi $\alpha_m = 0,02$ et une compensation diffuse, à part si nous mentionnons le contraire. Dans la figure 4.9, lignes d'en bas, la carte de normales est répétée 20 fois de plus que dans les lignes du haut. Les autres paramètres sont inchangés. Les normales tournant le dos à la caméra, visibles lors d'un rendu classique, s'agrègent et donnent une apparence trop sombre à la surface (figure 4.9, en bas à gauche). Notre BRDF locale f_∞ n'a pas ce problème (figure 4.9, en bas à droite).

4.5.2 Résultats utilisant la BRDF surfacique

La figure 4.10 montre différents matériaux que notre méthode peut modéliser. Une animation est disponible sur le [site du projet](#) pour évaluer la cohérence spatiale et temporelle des micro-reflets qu'offre notre méthode. Le métal brossé avec des dents peut être obtenu en utilisant une instance d'une surface de Beckmann avec une forte anisotropie pour la rugosité ($\alpha_x = 0,5$, $\alpha_y = 0,01$, voir la grosse cuillère). Le couteau utilise une texture de rayures orientées dans une seule direction, alors que la cuillère à dessert utilise des directions aléatoires, reproduisant une apparence complètement différente. Les fourchettes ont une BRDF de compensation f_{2+} très spéculaire, permettant de reproduire une apparence de plastique. La fourchette verte est extrêmement rugueuse et utilise une carte de normales de Beckmann isotrope avec $\alpha = 1$, et la fourchette violette utilise une texture de Voronoï avec une rugosité $\alpha = 1,41$. Pour finir, la cuillère en argent a une surface de Beckmann isotrope où $\alpha = 0,1$.

Empreinte mémoire. Pour la représentation de la carte de normales et l'estimation de la rugosité dans l'empreinte du rayon, nous utilisons deux textures MIP-mappées : une stockant les pentes ($x_{\bar{p}}, y_{\bar{p}}$) et une autre stockant les moments d'ordre deux ($x_{\bar{p}}^2, y_{\bar{p}}^2, x_{\bar{p}}y_{\bar{p}}$). Tous nos résultats utilisent des textures 2048×2048 MIP-mappées, utilisant chacune 96 MiB. L'approximation de l'énergie perdue dans le patch de la section 4.4.2.2 requiert 128×128 floats (64 KiB). Cette petite texture est la même quelle que soit la carte de normales. La structure accélératrice de la section 4.4.2 utilise une taille identique à celle utilisée dans les méthodes précédentes [YHMR16, CCM18] pour la même valeur de h (512 MiB pour $h = 1$ et 2048 MiB pour $h = 0,5$). Pour finir, nous n'avons pas besoin de stocker des mixtures de gaussiennes comparées aux méthodes précédentes.

Impact de l'approximation du terme d'énergie surfacique. Notre \mathcal{P} -BRDF utilise une approximation de l'énergie perdue dans le patch (section 4.4.2.2) pour accélérer les évaluations. La figure 4.11 compare cette approximation avec la quantité exacte, qui est calculée en sommant tous les termes de la somme de f_∞ sur toute l'empreinte de rayon. Notre \mathcal{P} -BRDF *exacte* passe le test du four blanc. Notre approximation permet de réduire les temps de calcul par deux environ, à part pour les rayures où beaucoup de

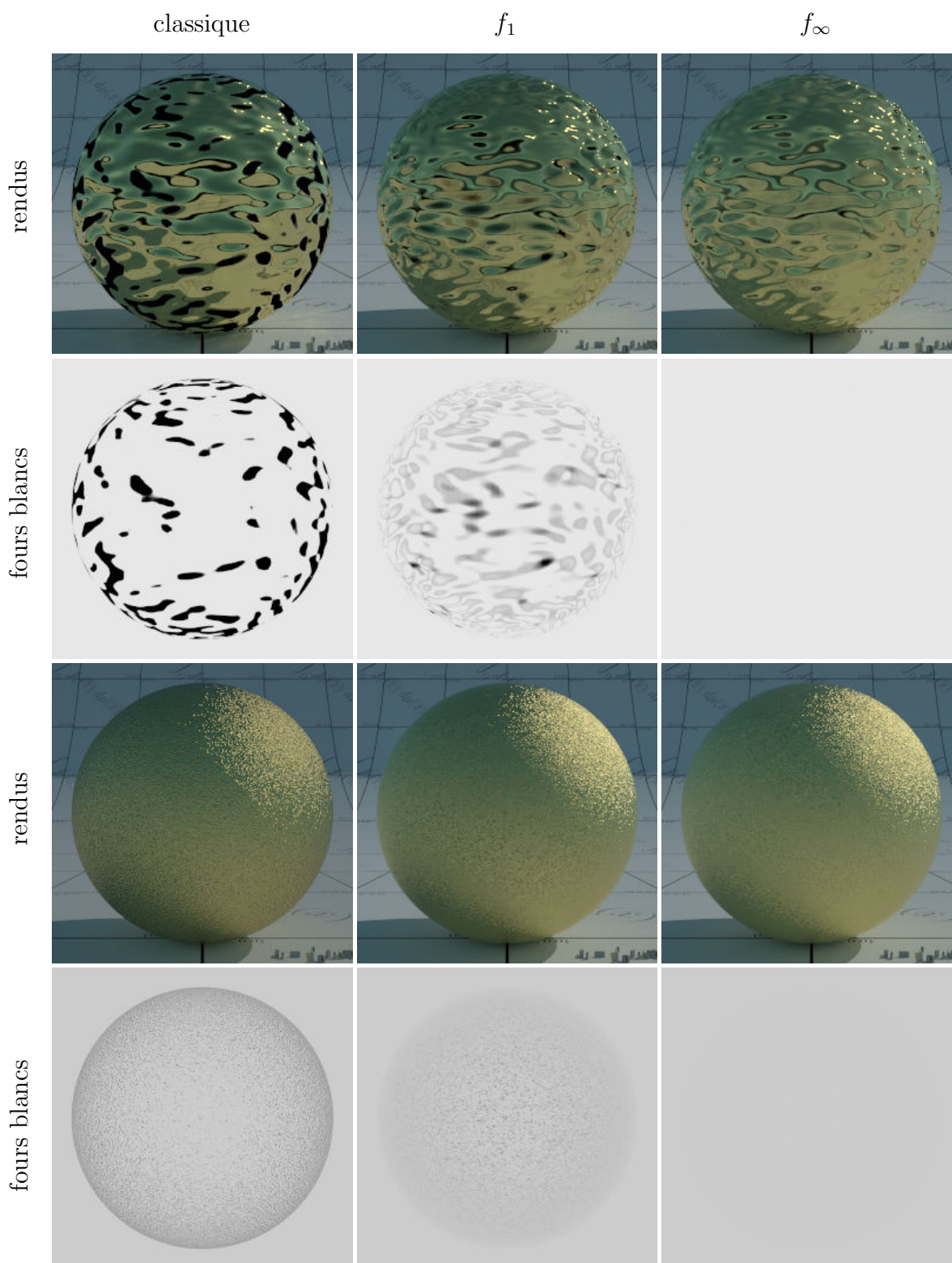


FIGURE 4.9 – Comparaisons entre une carte de normales de Beckmann classique (à gauche), et sa version basée micro-facettes simple rebond (au milieu) et multi-rebonds (à droite). Notre BRDF multi-rebonds f_∞ conserve l'énergie à 100 %, supprime les franges noires (lignes du haut) et l'apparence trop sombre (lignes du bas) sur la surface. En bas, la carte de normales est répétée 20 fois, donnant un lobe de Beckmann « rugueux ». Comme ici nous n'utilisons pas d'empreinte de rayon dans le but de valider seulement la BRDF locale, nous utilisons 131 072 epp pour rendre la majorité des reflets et ne plus avoir de bruits.

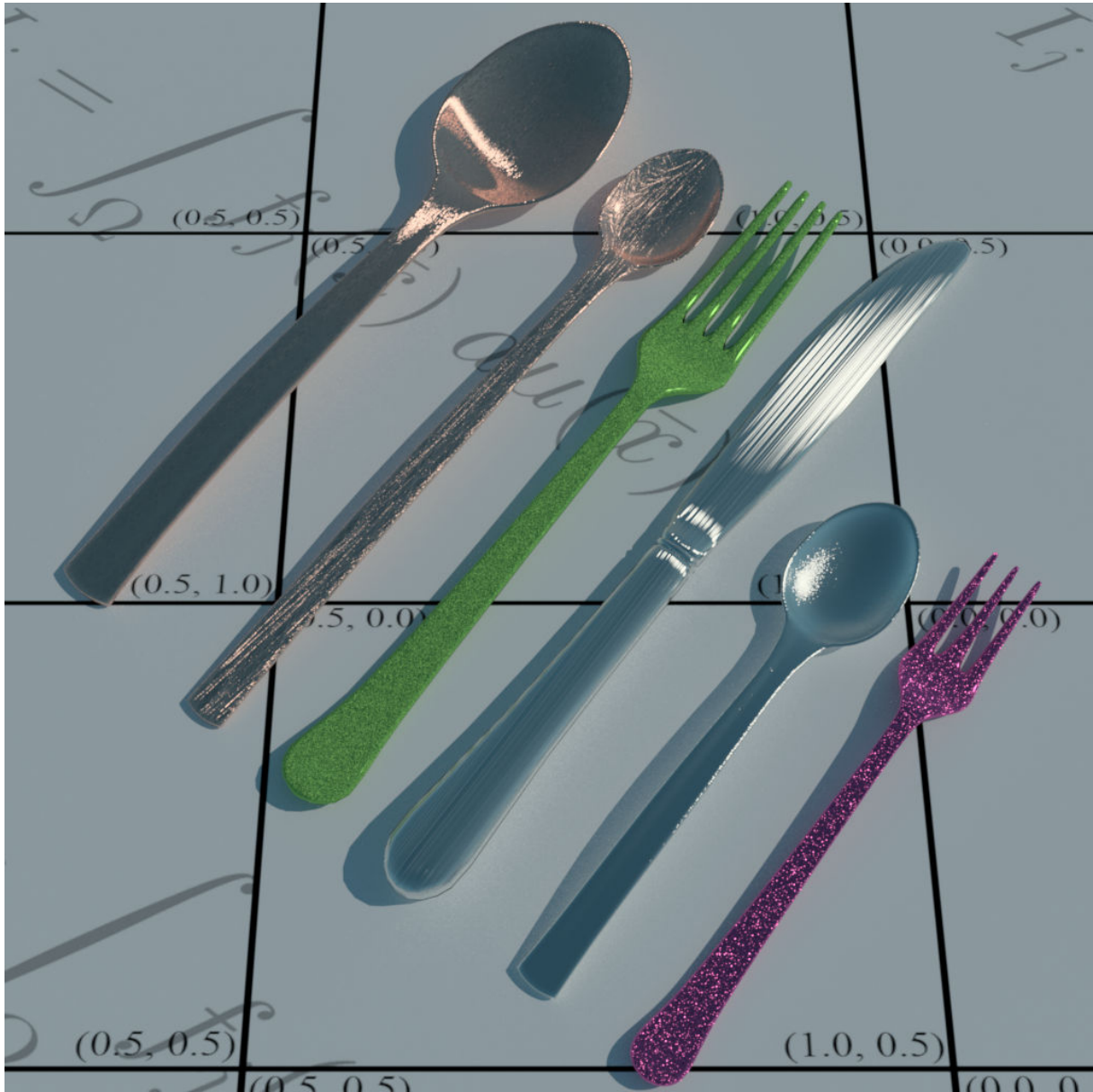


FIGURE 4.10 – Une scène avec des couverts rendue avec 1 024 epp et exhibant une large gamme de matériaux que notre méthode peut reproduire. De la gauche vers la droite : du métal brossé, rayé, du plastique vert rugueux et vernie, du métal machiné, du métal rugueux et une surface pailletée violette vernie.

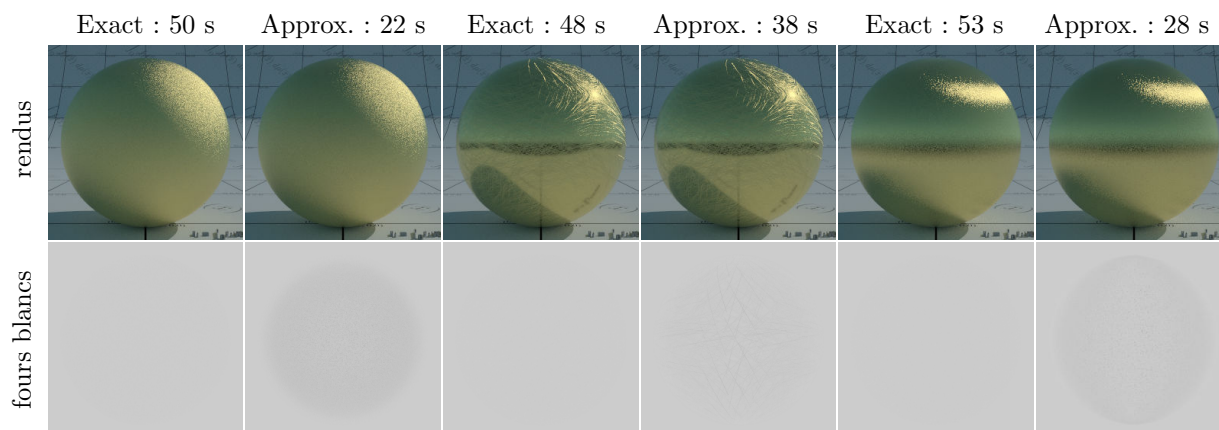


FIGURE 4.11 – Rendus et tests du four blanc de notre \mathcal{P} -BRDF utilisant le terme d'énergie surfacique exact ou approximé, pour différents types de cartes de normales. De la gauche vers la droite : une surface de Beckmann isotrope avec $\alpha = 0,5$, une surface rayée et une surface de Beckmann anisotrope avec $\alpha_x = 0,5$ et $\alpha_y = 0,1$. Notre modèle exact passe le test du four blanc. Cependant, notre \mathcal{P} -BRDF approximée n'est pas parfaite, mais a des bonnes propriétés énergétiques, des meilleures performances et donne des résultats similaires à la référence.

zones sont plates. Même si son utilisation ne permet pas d'avoir des tests du four blanc parfaits, l'approximation permet d'être compétitif face aux autres méthodes et donne des tests du four blanc qui sont meilleurs. De plus, l'apparence est plus plausible, car il n'y a plus de trous noirs, de couleurs insaturées ou d'apparences trop sombres.

Échantillonnage préférentiel. Les poids de notre procédure d'échantillonnage préférentiel ne sont jamais supérieurs à un, ce qui garantit un rendu sans pixels blancs non naturels (figure 4.12, droite). Au contraire, les méthodes précédentes peuvent donner des poids forts pour compenser les échantillons ne transportant pas de radiance (figure 4.12, gauche). Quand la surface est partiellement ombrée comme dans la figure 4.12, aucun pixel blanc ne doit être visible. C'est le cas pour notre modèle, mais pas pour la méthode précédente.

Choisir la BRDF de compensation. Presque toutes les figures dans le chapitre utilisent une BRDF de compensation f_{2+} diffuse pour modéliser le multi-rebonds. Ce choix donne des résultats permettant de mieux se comparer aux méthodes précédentes où l'apparence globale sombre peut être vue comme un matériau diffus avec un petit albédo. La figure 4.13 montre la différence entre l'utilisation d'une compensation f_{2+} diffuse et spéculaire, où $\alpha = 0,02$. La peinture métallisée peut être modélisée avec un f_{2+} très spéculaire, donnant l'impression que les paillettes en métal sont recouvertes d'un vernis transparent. Utiliser un f_{2+} lambertien convient mieux pour modéliser des matériaux très rugueux et sans vernis, comme des boules de Noël.

Comparaison avec les méthodes précédentes. Nous comparons les rendus de matériaux à paillettes en utilisant notre méthode et deux méthodes précédentes dans la figure 4.1 (et dans la vidéo). Les méthodes du chapitre 3 et de Yan et al. souffrent des normales tournant le dos à la caméra provoquant des trous noirs et une apparence

Chermain et al. : 3,5 s

Notre méthode : 4,5 s

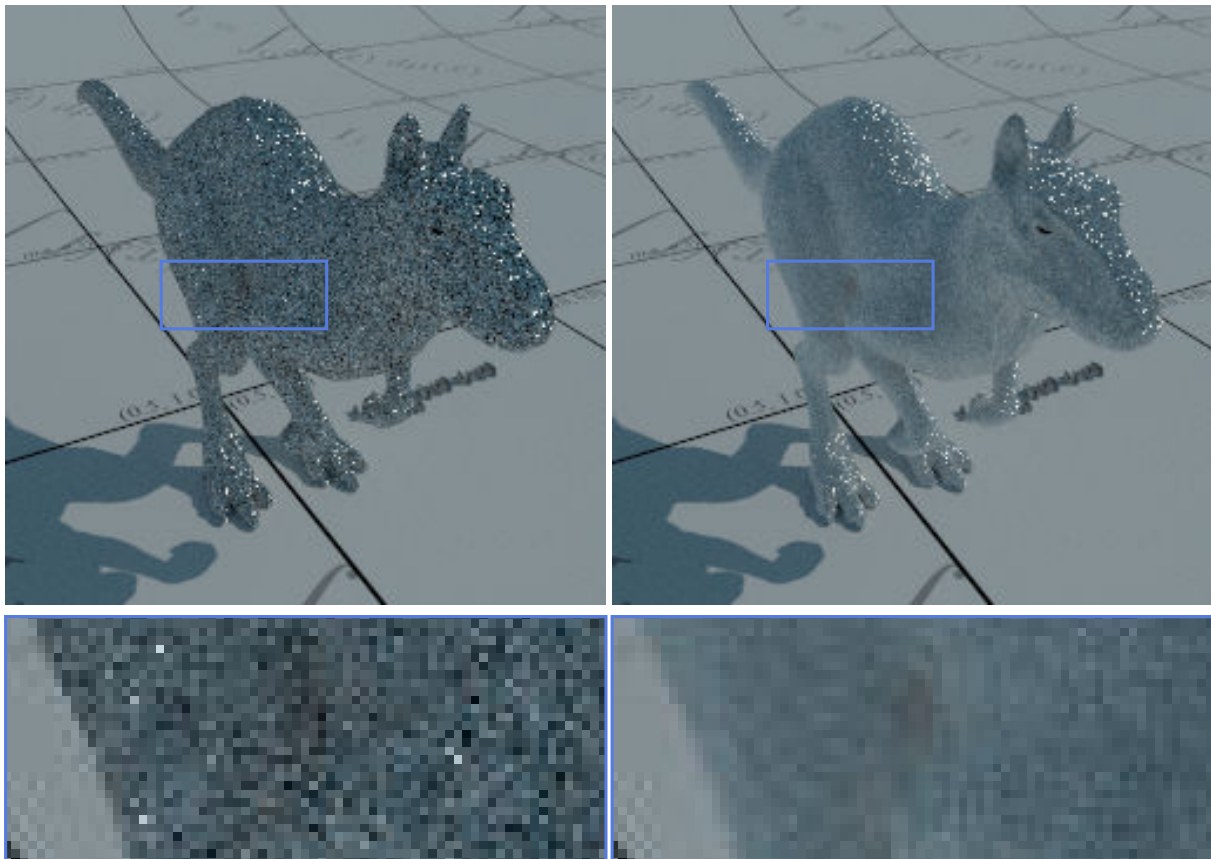


FIGURE 4.12 – Le Killeroo rendu à 256 échantillons par pixel. Des fireflies sont visibles quand la \mathcal{P} -BRDF a une procédure d'échantillonnage n'utilisant pas la direction d'observation (gauche). Notre méthode l'utilise et donc ne génère pas de fireflies (droite).

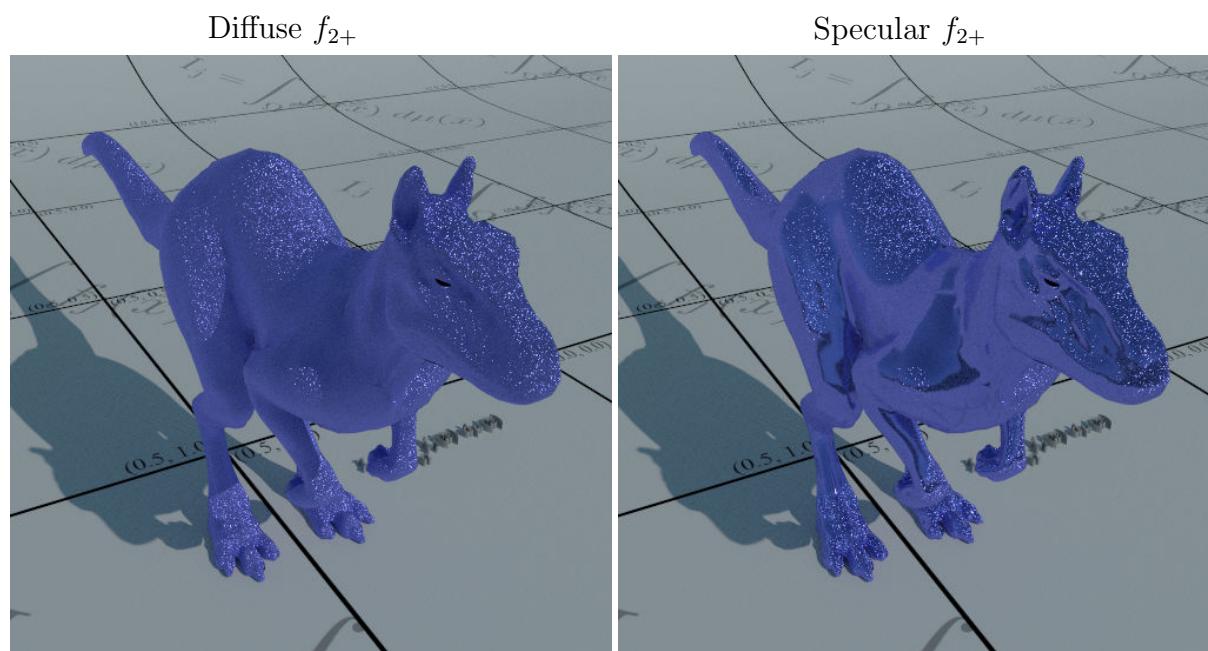


FIGURE 4.13 – Le Killeroo rendu à 256 échantillons par pixel avec notre \mathcal{P} -BRDF, utilisant soit une compensation diffuse (gauche) soit une compensation presque parfaitement spéculaire (droite), donnant des apparences très différentes.

sombre. Le résultat du test du four blanc est meilleur avec notre méthode. La figure 4.1 utilise la variante plate des mixtures de gaussiennes [YHMR16], qui est mieux adaptée pour représenter les surfaces pailletées (microrugosité constante $\alpha_m = 0,02$).

La figure 4.14 (et sa vidéo associée) montre un objet avec une micro-surface structurée, rendu en utilisant différentes méthodes. Les méthodes basées sur des mixtures de gaussiennes [YHMR16, CCM18] ont des rayures anormalement sombres ou trop claires. Nous pouvons l’observer dans les zones rasantes de l’orbe. Ce phénomène est dû aux normales tournant le dos ou à des normalisations incorrectes.

Les rendus de matériaux structurés utilisent une valeur $h = 0,5$ (figures 4.14 et 4.15) pour notre méthode pour une raison que nous allons maintenant expliquer. Les modèles du chapitre 3 et de Yan et al. reconstruisent une carte de normales continue avec des mixtures de gaussiennes (que nous pouvons aussi appeler mixtures de NDFs). Chaque NDF est une gaussienne qui a une variance calculée à partir des jacobiennes locales de la carte de normales, ce qui permet une meilleure correspondance avec la surface interpolée réelle (continue). Notre modèle n’utilise pas les jacobiennes, et ressemble plus à la variante plate de Yan et al., où la rugosité par normale est fixe. L’équation 4.3.8 nous empêche d’utiliser des rugosités locales trop grandes pour notre micro-BRDF. Notre méthode ne peut donc compter que sur les filtres k_j pour interpoler les valeurs de BRDF adjacentes lors du rendu. Nous pouvons aussi compter sur le fait que deux normales au lieu d’une sont définies par position discrète. Mais malgré tout, les hautes fréquences locales sont moins bien approximées : les mixtures de gaussiennes sont plus précises grâce à leurs NDFs locales dont la rugosité varie. Les rayures sont moins lisses et continues avec notre modèle. Ce problème peut être évité en réduisant la distance d’échantillonnage h de la carte de normales (ce qui augmente le nombre de V-cavités). De plus, l’ombrage intrinsèque de nos V-cavités pondère naturellement les intensités des micro-reflets, favorisant leur

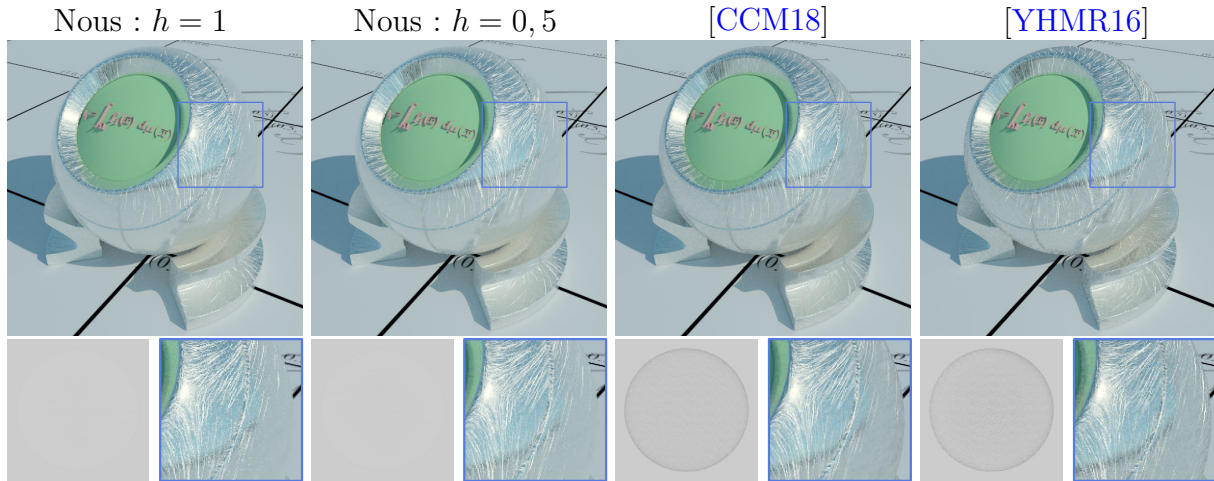


FIGURE 4.14 – Des rendus d’orbres métalliques rayés et des tests du four blanc utilisant notre méthode et les méthodes précédentes. Notez que e notre intégrateur de micro-reflets n’assombrit pas la silhouette des objets ou les rayures, comparativement aux méthodes précédentes. Les temps de rendu sont, de la gauche vers la droite : 129 s, 202 s, 296 s et 258 s.

continuité quand ils sont animés. Il faut aussi noter que les hautes rugosités permettant la continuité impactent sévèrement les performances des méthodes précédentes, car les boîtes englobantes du BVH deviennent très larges. Par exemple, l’orbe de la figure 4.14 rendu avec une micro-BRDF constante où $\alpha = 0,02$ (variante plate) ne requiert que 90 s pour être rendu avec les méthodes précédentes, contre plus de 250 s avec une microrugosité adaptative (variante incurvée). Notre proposition est donc de modifier le contenu de la texture ou la valeur de h pour améliorer la continuité localement, au lieu de modifier la microrugosité par texel. Bien sûr, la réduction de h ou l’ajout de texels dans la texture affectent les performances, mais d’une manière moins importante qu’en utilisant des mixtures de gaussiennes incurvées. Pour finir, notre méthode peut quand même utiliser des microrugosités α_m adaptatives (et perdre un peu d’énergie), même si nous avons fait le choix de ne pas le faire ici, pour les raisons évoquées ci-dessus.

La qualité de l’approximation de l’énergie perdue dans le patch (section 4.4.2.2) dépend de la taille de l’empreinte du rayon. Des empreintes larges permettent d’avoir des approximations gaussiennes de la distribution de normales très avantageuses (équation 4.4.7). En revanche, lorsque le zoom est important et que les empreintes sont petites, nous avons des sous-estimations de l’énergie perdue dans le patch, ce qui ne permet pas à notre méthode de passer le test du four blanc (figure 4.15). À cette échelle, la rugosité dans l’empreinte n’est pas un indicateur sûr pour estimer l’énergie perdue. Il faut néanmoins nuancer ces mauvais résultats avec ceux des méthodes précédentes, qui sont encore plus mauvais. En effet, Yan et al. normalisent leur BRDF dépendante de l’empreinte par un terme qui n’en dépend pas et dans le chapitre 3, nous souffrons tout simplement du manque de multi-rebonds.

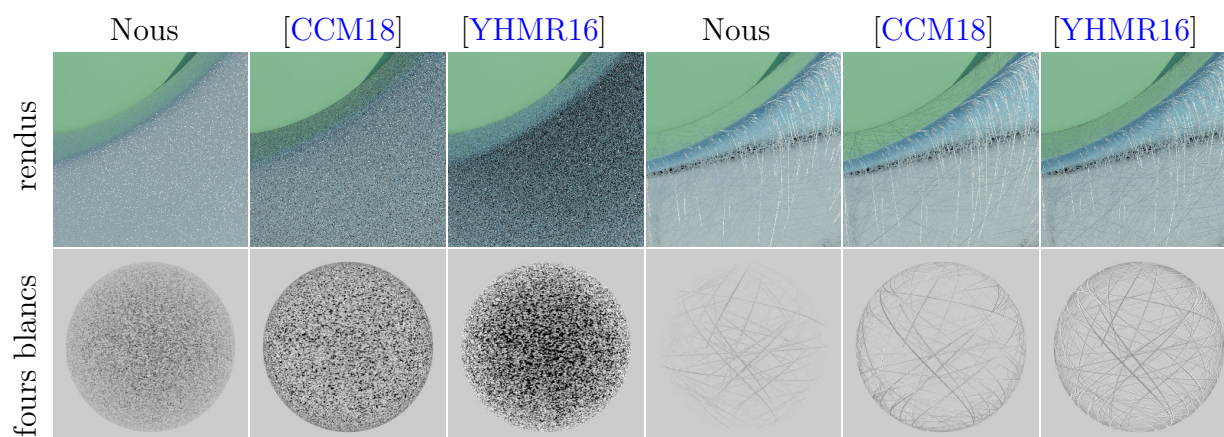


FIGURE 4.15 – Pour des vues proches, notre approximation de l'énergie perdue dans le patch produit des valeurs sous-estimées, donnant des tests du four blanc assez mauvais, mais meilleurs que ceux des autres méthodes. Les temps de rendu de la gauche vers la droite sont : 84 s, 65 s, 52 s, 194 s, 407 s et 398 s.

4.6 Conclusion

Dans ce travail, nous avons présenté le premier intégrateur de micro-reflets multi-rebonds basé sur une carte de normales. Notre méthode se base sur une BRDF locale multi-rebonds qui simule des perturbations de la normale géométrique. Notre \mathcal{P} -BRDF peut être utilisée dans un intégrateur de Monte Carlo unidirectionnel, et possède une procédure d'échantillonnage optimale, prenant en compte la visibilité des normales, contrairement aux méthodes précédentes (incluant la méthode du chapitre 3). Une large collection d'apparences de matériaux peut être reproduite en utilisant différentes textures. Contrairement aux méthodes précédentes basées sur des cartes de normales, nous obtenons des rendus qui n'ont pas d'artéfacts, comme des trous noirs ou une apparence globale trop sombre.

Chapitre 5 :

Conclusion

Le rendu de microstructures spéculaires a été abordé dans cette thèse. Nous avons vu que les modèles classiques de BRDF à micro-facettes ne sont pas aptes à reproduire l'apparence de matériaux à micro-reflets opaques, comme par exemple le métal rayé ou brossé, la peinture métallisée ou les surfaces pailletées. Nous avons vu que l'ajout de détails surfaciques spéculaires par le biais d'une carte de normales était possible. La carte de normales représente la micro-géométrie de ces matériaux. Cette représentation permet de reproduire une large gamme d'apparences mais est difficile à filtrer correctement. Les méthodes classiques de filtrage de surfaces augmentées simplifient la microstructure pour éliminer les hautes fréquences et donnent une apparence lisse à l'objet. Ce comportement est acceptable pour la plupart des situations, mais lorsque la micro-géométrie est très spéculaire dans le but de modéliser des effets scintillants, alors l'apparence filtrée est trop éloignée de la référence non simplifiée.

Faire le rendu fidèle et exact d'une carte de normales très spéculaire et non simplifiée permet de synthétiser l'apparence de matériaux à micro-reflets. Quelques méthodes [YHJ⁺14, YHMR16] ont proposé des algorithmes permettant de calculer la réflectance depuis ce type de surface, sans aliasing et sans simplification. Pour cela, les auteurs ont introduit le concept de \mathcal{P} -BRDF, basé sur une \mathcal{P} -NDF qui est la distribution de normales dans une empreinte de pixel sur une carte de normales. L'évaluation de la \mathcal{P} -NDF est compliquée, car elle implique de parcourir tous les texels dans l'empreinte, ce qui n'est pas très efficace. Mais des structures accélératrices hiérarchisant les texels (ou leur représentation, comme des NDFs 4D ou 2D) permettent de ne parcourir que ceux qui sont orientés à mi-chemin entre la lumière et l'observateur, rendant ces algorithmes performants face à un modèle classique.

Les modèles permettant le rendu de matériaux à micro-reflets basés sur des cartes de normales ont cependant quelques problèmes de normalisation entraînant une différence avec un rendu de référence obtenu en échantillonnant massivement la carte de normales. Dans la méthode de Yan et al. [YHMR16], la carte de normales est modélisée par une mixture de NDFs 4D (fonction d'une position et d'une normale), dans le but de calculer la \mathcal{P} -NDF rapidement. Malheureusement, la complexité de la formulation ne permet pas de normaliser facilement la \mathcal{P} -BRDF. De plus, dans leur travail, il n'y a pas de comparaisons avec des rendus de référence. Leur objectif est seulement de reproduire le plus fidèlement possible la \mathcal{P} -NDF de référence, pas la \mathcal{P} -BRDF.

Résumé des contributions. Dans le chapitre 3, nous avons résolu les problèmes énoncés précédemment, causés par la mixture de NDFs 4D. Nous avons choisi d'utiliser une mixture de NDFs 2D pour représenter une carte de normales, en supposant que localement, chaque NDF est constante. Nous avons utilisé des distributions de Beckmann décentrées et non-alignées sur les axes, car cette représentation est plus compacte et assez malléable pour approximer la NDF réelle dans un texel. Elle a un terme de masquage analytique, qui nous a permis de dériver un \mathcal{P} -masquage pour la \mathcal{P} -NDF. Même si sa formulation oblige à parcourir tous les texels de l'empreinte du rayon, nous avons obtenu des résultats très proches d'un rendu de référence. Dans le but d'être compétitif avec Yan et al. 2016, nous avons proposé une approximation du \mathcal{P} -masquage-ombrage exact, qui utilise le masquage-ombrage d'une distribution de Beckmann ayant une rugosité égale à celle contenue dans l'empreinte. La rugosité de l'empreinte du rayon est obtenue avec la méthode LEADR. Nous obtenons des résultats plus proches de la référence, avec une mixture de gaussiennes deux fois plus compacte.

Notre premier travail, présenté dans le chapitre 3 et publié dans *The Visual Computer* [CCM18], devait servir de socle pour une nouvelle \mathcal{P} -BRDF, qui devait cette fois être *multi-rebonds*. Le nouveau modèle ne devait pas ajouter trop de temps de calcul : comparé à l'état de l'art, nous ne voulions pas dépasser leur temps de calcul multiplié par deux. Les travaux précédents et le nôtre étant basés sur un échantillonnage de la \mathcal{P} -NDF, nous avons aussi essayé de dériver une \mathcal{P} -VNDF simple et pouvant être facilement échantillonnée. Nous avons rapidement remarqué que la tâche était vaine, tout simplement parce que le \mathcal{P} -masquage exact implique le parcours de tous les texels dans l'empreinte. Ce parcours est beaucoup trop coûteux, et sans le \mathcal{P} -masquage exact, nous n'avions pas de \mathcal{P} -VNDF accessible. Notre premier travail nous a permis de comprendre que nous allions dans la mauvaise direction, et que nos objectifs étaient irréalisables avec cette approche.

Dans le chapitre 4, nous avons fait table rase du premier travail et nous avons remarqué que le problème principal était le décentrage des normales par rapport à la normale géométrique. Ce décentrage implique une équation de filtrage de cartes de normales trop complexe (impliquant la \mathcal{P} -BRDF du chapitre 3). Nous avons rétabli en chaque point de la surface le centrage sur la normale géométrique en compensant toute perturbation de normale par une normale symétrique, nous donnant en ensemble de V-cavités, variant spatialement. Dans ce cas, la \mathcal{P} -BRDF devient une somme pondérée de BRDFs locales.

Cette formulation beaucoup plus simple nous a permis de proposer le premier modèle de \mathcal{P} -BRDF basé sur *une carte de normales, multi-rebonds* et avec un *échantillonnage optimal*. Il a été publié dans *Computer Graphics Forum* [CCM19] et présenté à EGSR 2019. Il est optimisé pour des cartes de normales presque parfaitement spéculaires. Il simule l'apparence de matériaux à micro-reflets avec un budget temps et mémoire presque similaire aux méthodes précédentes. Notre modèle se base sur des BRDFs locales multi-rebonds, simulant des fausses perturbations géométriques et pouvant être échantillonnées de manière optimale. Elles sont issues d'une dérivation d'une BRDF simple rebond d'une V-cavité possédant une micro-BRDF très spéculaire. Pour qu'elles soient multi-rebonds, nous compensons l'énergie perdue à l'aide d'une BRDF arbitraire et multi-rebonds (par exemple une BRDF de Lambert), modélisant les rebonds d'ordre deux et plus. Comparativement aux techniques de compensation précédentes, la nôtre a une procédure d'échantillonnage ne gaspillant pas d'échantillons.

Limitations et travaux futurs. Les limitations de la méthode du chapitre 3 ont été en partie résolues avec la méthode du chapitre 4, mais nos deux \mathcal{P} -BRDFs sont non réciproques. La première à cause du décentrage des normales, la deuxième à cause de la forme de la compensation d'énergie. Un premier travail futur serait de résoudre ce problème de réciprocité.

Les deux travaux proposés dans cette thèse sont par ailleurs gourmands en mémoire et ne sont pas utilisables en tant que tel en temps réel. Plus l'empreinte du rayon sera grande, plus il peut y avoir de BRDFs locales à calculer. Normalement, un modèle multi-échelles doit avoir un temps de calcul constant quelque soit la distance d'observation : nos algorithmes n'ont pas cette propriété. Des méthodes similaires à celles de Zirr et Kaplayan [ZK16], basées sur du MIP mapping, pourraient résoudre ce problème et en même temps réduire énormément l'empreinte mémoire. L'idéal serait même de créer de toutes pièces une \mathcal{P} -NDF analytique, variant spatialement de manière continue. Cette distribution devra aussi prendre en paramètre la taille de l'empreinte du rayon. Elle sera équivalente à une NDF de Beckmann quand l'empreinte sera très grande et sera un Dirac

centré sur une normale perturbée aléatoire quand l’empreinte sera un point. Bien sûr, la normale aléatoire devra toujours être la même en un point donné, et le changement de la forme de la NDF entre les tailles d’empreinte devra être continu et lisse. Pour finir, cette \mathcal{P} -NDF analytique devra être rapidement normalisable. Si une telle fonction existe, nous pourrions dans un premier temps l’utiliser dans une BRDF à micro-facettes utilisant un terme de masquage approximatif. Mais idéalement, il faudra une \mathcal{P} -NDF possédant un terme de masquage exact et analytique, et donnant une \mathcal{P} -VNDF pouvant être facilement échantillonnée. De plus, cette formulation devra être multi-rebonds.

Le contrôle de l’apparence avec des cartes de normales n’est pas une chose aisée et intuitive, contrairement à ce que l’on pourrait penser. Pour paramétrer la BRDF, il vaut mieux utiliser une densité de rayures ou de paillettes, par exemple. Si un artiste utilise notre méthode avec des textures de rayures faites à la main ou issues d’une photo, il n’aura sûrement pas le résultat escompté. Les textures utilisées dans le chapitre 3, faites à la main, ont nécessité plusieurs heures de travail et n’ont pas permis d’obtenir les apparences souhaitées à la base. Cependant, contrôler l’apparence est beaucoup plus facile à l’aide d’algorithmes de synthèse de textures, où nous pouvons contrôler les statistiques des pentes des normales, la densité des rayures et leur distribution dans l’espace. C’est ce que nous avons fait pour obtenir les résultats du chapitre 4. Ce besoin de contrôler directement les statistiques de la carte de normales rend notre travail plus difficile d’accès, selon nous. De plus, il nous fait réaliser que la texture n’est qu’une étape intermédiaire entre un processus aléatoire et le calcul d’une réflectance. Un travail futur pourrait être de lier directement les deux, sans passer par la texture. Mais dans ce cas, il faudrait se passer de structure accélératrice, ou bien en créer une nouvelle qui soit procédurale. Une autre motivation à ne pas utiliser de textures est la répétition de l’apparence sur la surface qu’elle induit.

La dernière limitation notable de nos travaux est l’absence de prise en compte de la diffusion de la lumière lorsque le matériau est diélectrique. Nous n’avons développé que des modèles de \mathcal{P} -BRDF et les généraliser pour en faire des \mathcal{P} -BSDFs serait intéressant. Malheureusement, cette généralisation n’est pas triviale et engendrerait plus de requêtes avec le BVH. Il faudrait retourner tous les texels ayant une normale proche du vecteur à mi-chemin de réflexion et du vecteur à mi-chemin de *transmission*. Ce qui reviendrait à quatre requêtes avec notre modèle multi-rebonds (chapitre 4). La compensation doit aussi être adaptée mais surtout, il faut pouvoir avoir facilement accès aux empreintes de rayon au-delà du premier rebond lumineux. Ce n’est pas le cas dans les moteurs de rendu actuels, basés sur les rayons différentiels de Igehy [Ige99] (à part pour des surfaces parfaitement spéculaires ou diffuses). Belcour et al. [BYRN17] proposent de les approximer en utilisant des matrices de covariance 4D. Le premier modèle de \mathcal{P} -BSDF pour le rendu de matériaux à micro-reflets transmis devra forcément implémenter cette méthode complexe, à part si les surfaces transparentes sont considérées comme infiniment fines.

Annexe A :
Code des matériaux pbrt-v3

A.1 Introduction

Nous donnons tout le code nécessaire pour tester les intégrateurs de micro-reflets du chapitre 3, du chapitre 4 et de Yan et al. [YHMR16] dans le moteur de rendu de `pbrt-v3` [PJH16]. Le code est disponible sur la [page du projet du chapitre 4](#).

Après un clone du dépôt de `pbrt-v3`, il faut copier nos fichiers sources dans les dossiers de `pbrt-v3`. Notre intégrateur de micro-reflets multi-rebonds dépend de matériaux locaux simple (section A.2.2.1) et multi-rebonds (section A.2.2.2), permettant le rendu de cartes de normales. Il faut aussi s'assurer d'avoir installé toutes les dépendances additionnelles (section A.4). Nous utilisons que des cartes de pentes, car beaucoup de nos algorithmes et de nos structures de données (LEADR mapping [DHI+13], BVH) sont plus efficaces avec cette représentation. Transformer une carte de normales en une carte de pentes est trivial [Dup17].

A.2 Rendu de cartes de normales

Dans cette section, nous donnons les instructions pour ajouter un support de rendu à des cartes de normales classiques à `pbrt-v3`, en se basant sur des distributions de normales décentrées (section A.2.1, figure A.1, droite). Nous donnons aussi les BRDFs pour rendre des cartes de normales basées micro-facettes, utilisant des facettes symétriques, et modélisant un rebond lumineux (section A.2.2.1, figure A.1, milieu) ou une infinité (section A.2.2.2, figure A.1, droite). Les deux sont utilisées par notre intégrateur de micro-reflets multi-rebonds. Nous décrivons aussi tous les paramètres de chaque matériau.

A.2.1 Classique

Nous donnons ici les instructions pour installer le support pour rendre des cartes de normales classiques, utilisant des distributions de Beckmann décentrées [DHI+13]. Cette représentation ne crée pas d'énergie. Pour installer ce matériau, il faut copier les fichiers `materials/normalmap_classic.*` dans le dossier `src/materials/` de `pbrt-v3`. Ensuite, il faut référencer ces fichiers dans l'API, en ajoutant dans le fichier `core/api.cpp` de `pbrt-v3` les lignes suivantes :

```
#include "materials/normalmap_classic.h"
```

à côté des autres headers et

```
else if (name == "normalmap_classic")
    material = CreateNormalMapClassicMaterial(mp);
```

dans la fonction `MakeMaterial()`. La scène `pbrt` est localisée dans `scenes/normalmap_classic/` et donne les images de droite dans la figure A.1. Le matériau `normalmap_classic` a les paramètres suivants :

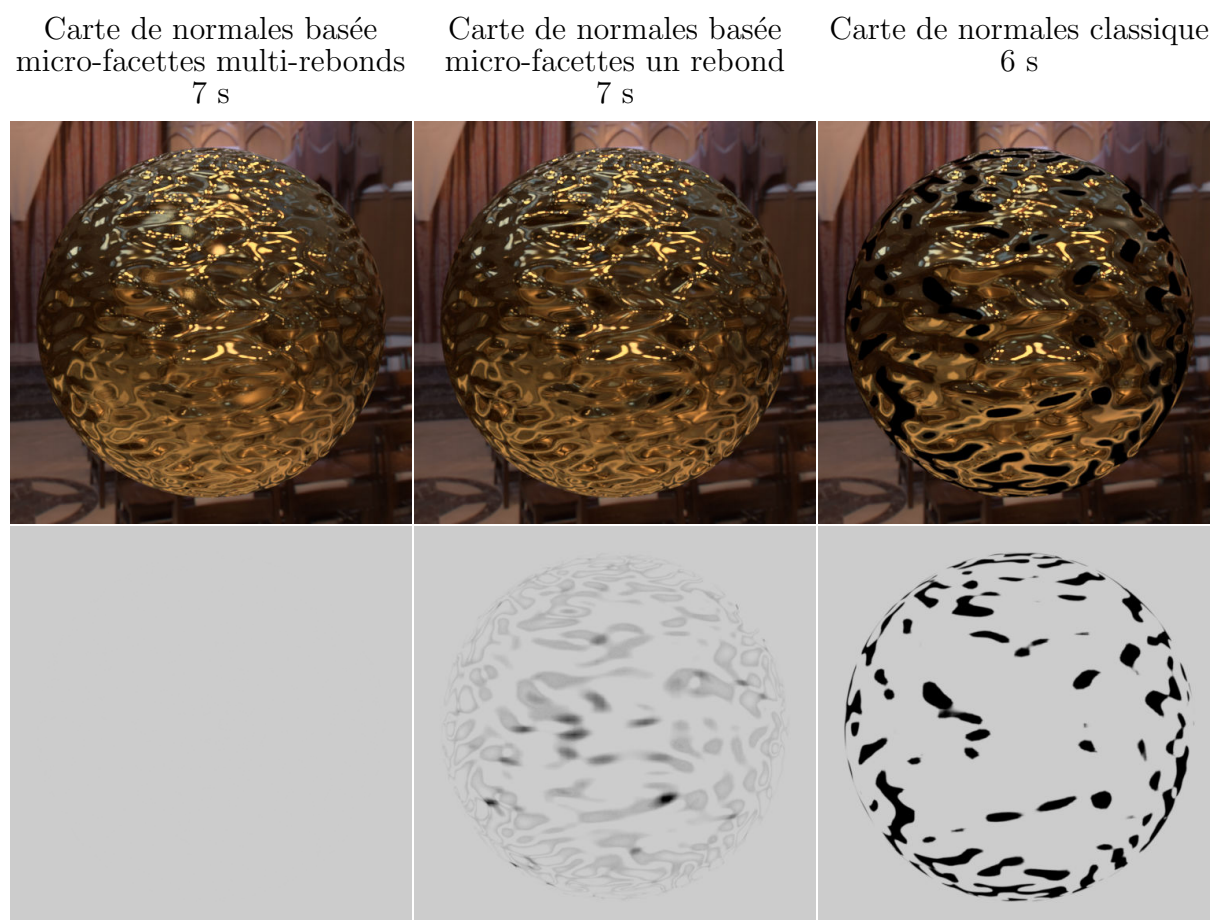


FIGURE A.1 – Résultats obtenus avec notre BRDF multi-rebonds f_∞ (gauche), simple rebond f_1 (milieu) et classique (droite). Haut : Résultats des rendus. Bas : Tests du four blanc.

| Type | Nom | Valeur par défaut | Description |
|------------------|-------------|-------------------|--|
| Spectrum texture | eta | (copper) | Comme dans <code>pbrt-v3</code> . Indice de réfraction du matériau. |
| Spectrum texture | k | (copper) | Comme dans <code>pbrt-v3</code> . Coefficient d'absorption du matériau. |
| Float texture | alpha_x | 0.02 | Rugosité microscopique α du matériau de la carte de normales dans la direction x . |
| Float texture | alpha_y | 0.02 | Rugosité microscopique α du matériau de la carte de normales dans la direction y . |
| Spectrum texture | slopes | None | La carte de pentes utilisant le canal rouge et vert pour respectivement les composantes x et y de la pente. Utilisez seulement des textures <code>.exr</code> carrées avec 2^n texels. |
| boolean | filtering | false | Active ou désactive le filtrage des pentes. |
| boolean | fresnelnoop | false | Active ou désactive l'absorption de Fresnel. Mettre à vrai pour les tests du four blanc. |

A.2.2 Rendu de cartes de normales utilisant des facettes symétriques

Les BRDFs simple rebond f_1 et multi-rebonds f_∞ permettant le rendu de cartes de normales sont utilisées par l'intégrateur de micro-reflets multi-rebonds.

A.2.2.1 Simple rebond

Pour installer le matériau f_1 permettant le rendu simple rebond de cartes de normales, il faut copier les fichiers `materials/normalmap_microfacet_sym_ss.*` dans le dossier `src/materials/` de `pbrt-v3`. Il faut ensuite les référencer dans l'API, en ajoutant dans le fichier `core/api.cpp` les lignes suivantes :

```
#include "materials/normalmap_microfacet_sym_ss.h"
```

dans la partie header et

```
else if (name == "normalmap_microfacet_sym_ss")
    material = CreateNormalMapMicrofacetSymSSMaterial(mp);
```

dans la fonction `MakeMaterial()`. La scène `pbrt` de ce matériau est localisée dans `scenes/normalmap_microfacet_sym_ss/` et donne les images du milieu de la figure A.1. Le matériau `normalmap_microfacet_sym_ss` a un paramètre additionnel, comparé au matériau `normalmap_classic` :

| Type | Nom | Valeur par défaut | Description |
|------------------|-----|-------------------|--|
| Spectrum texture | R | 1 | Couleur de base du matériau. Si inférieure à 1, mettre <code>fresnelnoop</code> à faux pour éviter des mélanges de couleurs. |

Ce matériau doit être utilisé avec l'intégrateur `pathiw` (section A.4.1).

A.2.2.2 Multi-rebonds

Pour installer ce matériau, il faut copier les fichiers `materials/normalmap_microfacet_sym_ms.*` dans le dossier `src/materials/` de `pbrt-v3`. Il faut ensuite les référencer dans l'API, en ajoutant au fichier `core/api.cpp` les lignes suivantes :

```
#include "materials/normalmap_microfacet_sym_ms.h"
```

dans la partie header et

```
else if (name == "normalmap_microfacet_sym_ms")
    material = CreateNormalMapMicrofacetSymMSMaterial(mp);
```

dans la fonction `MakeMaterial()`. La scène `pbrt` de ce matériau est localisée dans le dossier `scenes/normalmap_microfacet_sym_ms/` et donne les images à gauche de la figure A.1. Le matériau `normalmap_microfacet_sym_ms` a les paramètres suivants :

| Type | Nom | Valeur par défaut | Description |
|------------------|--------------------------|-------------------|--|
| Spectrum texture | <code>eta</code> | (copper) | Comme dans <code>pbrt-v3</code> . Indice de réfraction du matériau. |
| Spectrum texture | <code>k</code> | (copper) | Comme dans <code>pbrt-v3</code> . Coefficient d'absorption du matériau. |
| Spectrum texture | <code>R</code> | 1 | Couleur de base du matériau. Mettre le paramètre <code>fresnelnoop</code> à faux quand il est utilisé pour éviter de mixer deux couleurs. |
| Float texture | <code>ss_alpha_x</code> | 0.02 | Rugosité simple rebond microscopique α du matériau de la carte de normales dans la direction x . Utiliser des valeurs au-dessus de 0.02 n'est pas recommandé, car elles entraînent des fuites d'énergie. Ce paramètre est isotrope dans le papier, et s'appelle α_m . |
| Float texture | <code>ss_alpha_y</code> | 0.02 | Rugosité simple rebond microscopique α dans la direction y . Plus d'informations dans la ligne du dessus. |
| Float texture | <code>ms_alpha_x</code> | 0.0 | Mettre à 0 pour avoir une compensation f_{2+} lambertienne. Sinon, la compensation spéculaire est utilisée, modélisée par une BRDF à base de micro-facettes simple rebond avec une rugosité α . Ce paramètre contrôle cette rugosité, et il faut utiliser des valeurs faibles (≤ 0.02) pour ne pas avoir de fuites d'énergies. |
| Float texture | <code>ms_alpha_y</code> | 0.0 | Comme pour la ligne du dessus, mais pour la direction y de la surface. |
| Spectrum texture | <code>slopes</code> | None | La carte de pentes utilisant le canal rouge et vert pour respectivement les composantes x et y de la pente. Utilisez seulement des textures <code>.exr</code> carrées avec 2^n texels. |
| boolean | <code>filtering</code> | false | Contrôle si les pentes sont filtrées ou non. |
| boolean | <code>fresnelnoop</code> | false | Active ou désactive l'absorption de Fresnel. Mettre à vrai pour les tests du four blanc. |

Doit être utilisé avec l'intégrateur `pathiw` (section A.4.1).

A.3 Intégrateurs de micro-reflets

Dans cette section, nous donnons les instructions pour installer

- notre intégrateur de micro-reflets multi-rebonds (section A.3.2, figure A.2, gauche),
- et la variante plate de Yan et al. [YHMR16] (section A.3.1.1, figure A.2, droite),
- et du chapitre 3 [CCM18] (section A.3.1.2, figure A.2, milieu).

Les matériaux de cette section doivent être utilisés avec l'intégrateur `pathiw` (section A.4.1).

Intégrateur de micro-reflets
multi-rebonds
20,5 s

Chapitre 3
15 s

Yan et al.
14 s

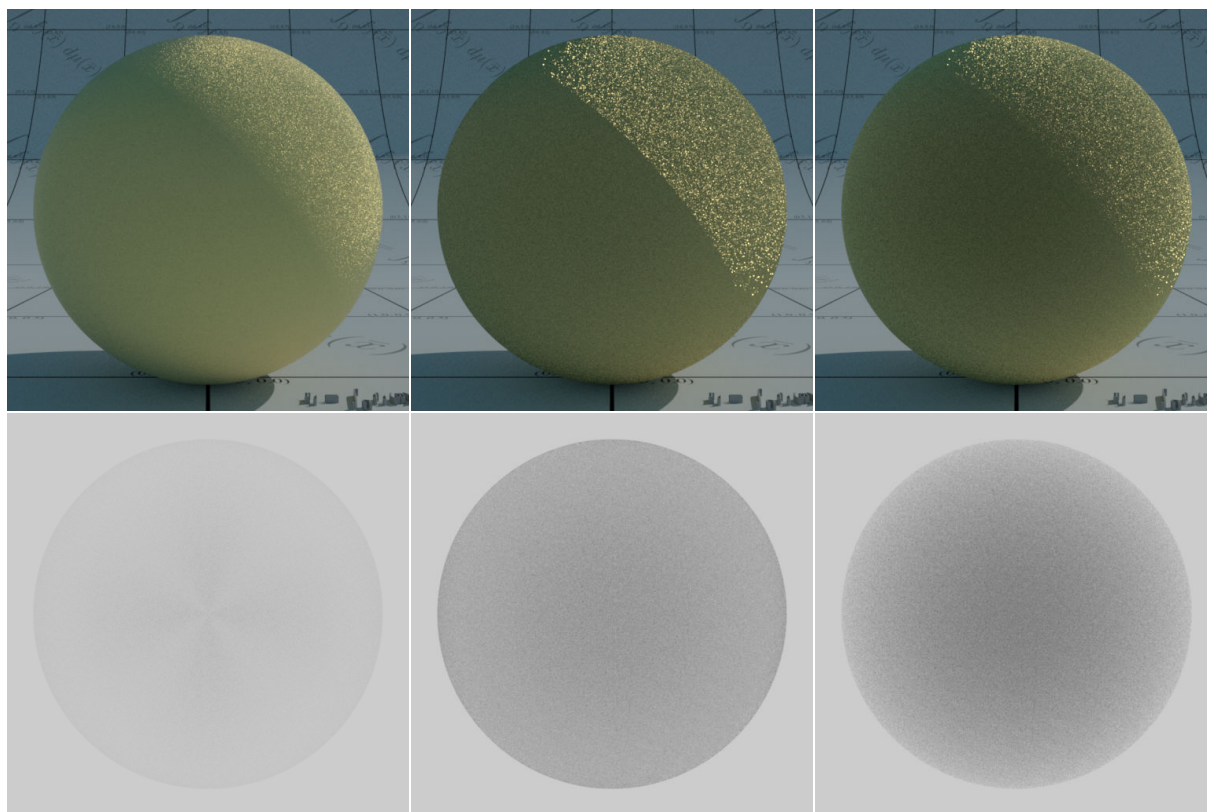


FIGURE A.2 – Résultats obtenus avec notre intégrateur multi-rebonds (gauche, chapitre 4), la méthode du chapitre 3 (milieu) et la méthode de Yan et al. 2016 [YHMR16] (droite). Les deux dernières méthodes modélisent seulement un rebond lumineux dans la micro-surface. Haut : Rendus. Bas : Tests du four blanc.

A.3.1 Simple rebond

A.3.1.1 Yan et al. 2016

Pour installer ce matériau, il faut copier les fichiers `materials/yan_et_al_2016.*` dans le dossier `src/materials/` de `pbrt-v3`. Il faut les référencer dans l'API, en ajoutant dans le fichier `core/api.cpp` les lignes suivantes :

```
#include "materials/yan_et_al_2016.h"
```

dans la partie header et

```
else if (name == "yan_et_al_2016")
    material = CreateYanEtAl2016Mat(mp);
```

dans la fonction `MakeMaterial()`. La scène `pbrt` de ce matériau est localisée dans le dossier `scenes/yan_et_al_2016/` et donne les images de droite de la figure A.2. Le matériau `yan_et_al_2016` a les paramètres suivants :

| Type | Nom | Valeur par défaut | Description |
|------------------|--------------------------|-------------------|--|
| Spectrum texture | <code>eta</code> | (copper) | Comme dans <code>pbrt-v3</code> . Indice de réfraction du matériau. |
| Spectrum texture | <code>k</code> | (copper) | Comme dans <code>pbrt-v3</code> . Coefficient d'absorption du matériau. |
| Float | <code>sigma_r</code> | 0.014 | Rugosité RMS (Root Mean Square) microscopique du matériau de la carte de normales dans la direction x et y . Augmenter cette valeur réduit les performances car les boîtes englobantes du BVH deviennent plus larges. |
| Float | <code>step_h</code> | 1 | Distance entre deux normales discrètes lors de l'échantillonnage uniforme de la carte de normales. Dans le papier, ce terme est noté h . Les valeurs doivent être égale à 0.25×2^n , par exemple, 0.25, 0.5, 1, 2, etc. |
| Spectrum texture | <code>slopes</code> | None | La carte de pentes utilisant le canal rouge et vert pour respectivement les composantes x et y de la pente. Utilisez seulement des textures <code>.exr</code> carrées avec 2^n texels. |
| boolean | <code>fresnelnoop</code> | false | Active ou désactive l'absorption de Fresnel. Mettre à vrai pour les tests du four blanc. |

A.3.1.2 Chermain et al. 2018

Pour installer ce matériau, il faut copier les fichiers `materials/chermain_et_al_2018.*` dans le dossier `src/materials/` de `pbrt-v3`. Il faut ensuite les référencer dans l'API en ajoutant dans le fichier `core/api.cpp` les lignes suivantes :

```
#include "materials/chermain_et_al_2018.h"
```

dans la partie header et

```
else if (name == "chermain_et_al_2018")
    material = CreateChermainEtAl2018Mat(mp);
```

dans la fonction `MakeMaterial()`. La scène `pbrt` de ce matériau est localisée dans `scenes/chermain_et_al_2018/` et donne les images du milieu de la figure A.2. Le matériau `chermain_et_al_2018` a un paramètre en plus comparé à `yan_et_al_2016` :

| Type | Nom | Valeur par défaut | Description |
|------------------|-------------|-------------------|--|
| Spectrum texture | 2nd_moments | None | Les moments d'ordre deux de la carte des pentes. Voir le papier LEADR [DHI+13] pour plus d'informations. Le canal rouge stocke la composante x de la pente au carrée, le canal vert stocke la composante y au carrée, et le canal bleu stocke le produit des composantes x et y . Utilisez seulement des textures <code>.exr</code> carrées avec 2^n texels. |

A.3.2 Multi-rebonds

Pour installer notre intégrateur de micro-reflets multi-rebonds décrit dans le chapitre 4, copiez les fichiers `materials/glint.*` dans le dossier `src/materials/` de `pbrt-v3`. Référez les dans l'API en ajoutant les lignes suivantes dans le fichier `core/api.cpp` :

```
#include "materials/glint.h"
```

dans la partie header et

```
else if (name == "glint")
    material = CreateGlintMat(mp);
```

dans la fonction `MakeMaterial()`. La scène `pbrt` du matériau est localisée dans `scenes/glint/` et donne les images à gauche de la figure A.2. Le matériau `glint` a les paramètres suivants :

| Type | Nom | Valeur par défaut | Description |
|------------------|--------------------------|-------------------|---|
| Spectrum texture | <code>eta</code> | (copper) | Comme dans <code>pbrt-v3</code> . Indice de réfraction du matériau. |
| Spectrum texture | <code>k</code> | (copper) | Comme dans <code>pbrt-v3</code> . Coefficient d'absorption du matériau. |
| Spectrum texture | <code>R</code> | 1 | Couleur de base du matériau. Si inférieur à 1, mettre <code>fresnelnoop</code> à faux. |
| Float | <code>alpha_x_m</code> | 0.02 | Rugosité simple rebond microscopique α du matériau de la carte de normales dans la direction x . Ce paramètre s'appelle α_m dans le papier. |
| Float | <code>alpha_y_m</code> | 0.02 | Rugosité simple rebond microscopique α dans la direction y . Plus d'informations dans la ligne du dessus. |
| Float | <code>ms_alpha_x</code> | 0.0 | Mettre à 0 pour avoir une compensation f_{2+} lambertienne. Sinon, la compensation spéculaire est utilisée, où ce paramètre contrôle la rugosité. |
| Float | <code>ms_alpha_y</code> | 0.0 | Comme pour la ligne du dessus, mais pour la direction y de la surface. |
| Float | <code>step_h</code> | 1 | Distance entre deux normales discrètes lors de l'échantillonnage uniforme de la carte de normales. Dans le papier, ce terme est noté h . Les valeurs doivent être égale à 0.25×2^n , par exemple, 0.25, 0.5, 1, 2, etc. |
| Spectrum texture | <code>slopes</code> | None | La carte de pentes utilisant le canal rouge et vert pour respectivement les composantes x et y de la pente. Utilisez seulement des textures <code>.exr</code> carrées avec 2^n texels. |
| Spectrum texture | <code>2nd_moments</code> | None | Les moments d'ordre deux de la carte des pentes. Voir le papier LEADR [DHI ⁺ 13] pour plus d'informations. Le canal rouge stocke la composante x de la pente au carrée, le canal vert stocke la composante y au carrée, et le canal bleu stocke le produit des composantes x et y . Utilisez seulement des textures <code>.exr</code> carrées avec 2^n texels. |
| Spectrum texture | <code>energy</code> | None | Texture contenant le terme d'énergie d'une surface ayant une distribution de normales isotrope et un matériau f_1 où la rugosité RMS de la distribution varie sur l'axe des x de 0.01 à 1, et où l'angle polaire de la direction d'observation varie de 0 à $\frac{\pi}{2}$ sur l'axe des y . La texture fournie a 128 par 128 texels, ce qui est suffisant. |
| boolean | <code>fresnelnoop</code> | false | Active ou désactive l'absorption de Fresnel. Mettre à vrai pour les tests du four blanc. |

A.4 Dépendances additionnelles

Les matériaux donnés ici requièrent des dépendances additionnelles qui ne sont pas incluses dans la version de base de `pbrt-v3` :

- `core/frame.h` : transforme les vecteurs d'un repère à un autre. Pour l'installer, copiez `core/frame.h` dans le dossier `core/` de `pbrt-v3`. Ajoutez le fichier aux `PBRT_CORE_HEADERS` du fichier `CMakeLists.txt`.
- `core/microfacet_extended.*` : Extension du module `core/microfacet.*` de `pbrt-v3`. Définit les fonctions `normalToSlope` et `slopeToNormal` et les distributions de Beckmann non-centrées et non-alignées sur les axes. Pour installer ce module, copiez `core/microfacet_extended.*` dans le dossier `core/` de `pbrt-v3` et ajoutez les fichiers dans `PBRT_CORE_SOURCE` et `PBRT_CORE_HEADERS` du fichier `CMakeLists.txt`.
- `textures/imagemap_extended.*` : Extension du module `textures/imagemap.*` de `pbrt-v3`. Ajoute des fonctionnalités utiles (`ComputeOverallRoughness`, `Texel`, `Width`, ...). Pour installer ce module, copiez `textures/imagemap.*` dans le dossier `textures/` de `pbrt-v3`. Dans le fichier `api.cpp`, ajoutez


```
else if (name == "imagemap_extended")
    tex = CreateImageFloatTextureExtended(tex2world, tp);
```

 dans la fonction `MakeFloatTexture` et


```
else if (name == "imagemap_extended")
    tex = CreateImageSpectrumTextureExtended(tex2world, tp);
```

 dans la fonction `MakeSpectrumTexture`.
- `core/geometry_extended.h` : Extension du module `core/geometry.h` de `pbrt-v3`. Ajoute les classes `Vector4`, `Point4` et `Bounds4`, permettant d'utiliser des vecteurs, des points et des boîtes englobantes 4D dont dépend le BVH 4D (item suivant). Pour installer ce module, copiez le fichier `core/geometry_extended.h` dans le dossier `core/` de `pbrt-v3`. Ajoutez le dans `PBRT_CORE_HEADERS` du fichier `CMakeLists.txt`.
- `accelerators/bvhvslope.*` : Un BVH 4D donnant rapidement les normales contribuant dans une empreinte de rayon. Il est utilisé par les intégrateurs de micro-reflets. Pour l'installer, copiez les fichiers `accelerators/bvhvslope.*` dans le dossier `accelerators/` de `pbrt-v3`.
- `core/rayfootprint.*` : Définit les fonctions associées au calcul de l'empreinte du rayon, utilisées par les intégrateurs de micro-reflets. Pour les installer, copiez les fichiers `core/rayfootprint.*` dans le dossier `core/` de `pbrt-v3`. Ajoutez les dans `PBRT_CORE_SOURCE` et `PBRT_CORE_HEADERS` du fichier `CMakeLists.txt`.

A.4.1 Tracer de chemins compatible avec les poids d'échantillons

Les matériaux fournis ici retournent directement le poids des échantillons, qui est la valeur de la BRDF évaluée avec la direction échantillonnée, divisée par la densité de probabilité (PDF) de la procédure d'échantillonnage. Nous avons modifié l'algorithme du tracer de chemins et la fonction `BSDF::Sample_f` pour qu'ils utilisent directement cette

valeur. Dans `pbrt-v3`, `BxDF::Sample_f` doit retourner la valeur de la BxDF et de la PDF de l'échantillon séparément. En pratique, retourner le ratio de ces valeurs est plus efficace car des simplifications ont lieu lors de la division (voir l'implémentation de Mitsuba, par exemple).

Le nom de notre intégrateur est `pathiw` pour path importance weight. Il est localisé dans le dossier `integrators` et il faut le copier dans celui de `pbrt`. Pour l'utiliser lors d'un rendu d'une scène `pbrt`, il faut remplacer `path` par `pathiw` quand nous spécifions l'`Integrator`. N'oubliez pas de le référencer dans le fichier `core/api.cpp` en ajoutant

```
#include "integrators/pathiw.h"
```

dans la partie header et

```
else if (IntegratorName == "pathiw")
    integrator = CreatePathIWIntegrator(IntegratorParams, sampler, camera);
```

dans la fonction `RenderOptions::MakeIntegrator()`. Finalement, dans la fonction `BxDF::Sample_f` du fichier `core/reflection.cpp`, remplacez le fragment suivant

```
// Compute value of BSDF for sampled direction
if (!(bxdf->type & BSDF_SPECULAR)) {
    bool reflect = Dot(*wiWorld, ng) * Dot(woWorld, ng) > 0;
    f = 0.;
    for (int i = 0; i < nBxDFs; ++i)
        if (bxdfs[i]->MatchesFlags(type) &&
            ((reflect && (bxdfs[i]->type & BSDF_REFLECTION)) ||
             (!reflect && (bxdfs[i]->type & BSDF_TRANSMISSION))))
            f += bxdfs[i]->f(wo, wi);
}
```

par celui-ci

```
// Compute value of BSDF for sampled direction
if (!(bxdf->type & BSDF_SPECULAR) && matchingComps > 1) {
    bool reflect = Dot(*wiWorld, ng) * Dot(woWorld, ng) > 0;
    for (int i = 1; i < nBxDFs; ++i)
        if (bxdfs[i] != bxdf && bxdfs[i]->MatchesFlags(type) &&
            ((reflect && (bxdfs[i]->type & BSDF_REFLECTION)) ||
             (!reflect && (bxdfs[i]->type & BSDF_TRANSMISSION))))
            f += bxdfs[i]->f(wo, wi);
}
```

dans le but d'éviter de recalculer la valeur de la BSDF échantillonnée (déjà fait lors de l'échantillonnage de la BSDF). Cet intégrateur et la modification de la fonction `BxDF::Sample_f` sont compatibles avec la fonction standard `Sample_f` des BRDFs de base de `pbrt-v3`, c'est-à-dire que les rendus obtenus avec notre intégrateur et l'original sont les mêmes. Quand nous retournons le poids, nous mettons la `pdf` à 1 lors de l'échantillonnage, pour garantir la compatibilité avec les fonctions standards de l'API. Finalement, comparé à l'intégrateur original `path`, nous échantillons toutes les lumières dans le but de rendre tous les reflets sous pixel.

Bibliographie

- [AK16] Asen Atanasov and Vladimir Koylazov. A Practical Stochastic Algorithm for Rendering Mirror-like Flakes. In *ACM SIGGRAPH 2016 Talks*, SIGGRAPH '16, pages 67 :1–67 :2, New York, NY, USA, 2016. ACM.
- [Ama84] John Amanatides. Ray Tracing with Cones. *SIGGRAPH Comput. Graph.*, 18(3) :129–135, January 1984.
- [APS00] Michael Ashikmin, Simon Premože, and Peter Shirley. A Microfacet-based BRDF Generator. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 65–74, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.
- [Bli77] James F. Blinn. Models of Light Reflection for Computer Synthesized Pictures. *SIGGRAPH Comput. Graph.*, 11(2) :192–198, July 1977.
- [Bli78] James F. Blinn. Simulation of Wrinkled Surfaces. *SIGGRAPH Comput. Graph.*, 12(3) :286–292, August 1978.
- [BN12] E. Bruneton and F. Neyret. A Survey of Nonlinear Prefiltering Methods for Efficient and Accurate Surface Shading. *IEEE Transactions on Visualization and Computer Graphics*, 18(2) :242–260, Feb 2012.
- [BP10] Carles Bosch and Gustavo Patow. Real-time path-based surface detail. *Computers and Graphics*, 34(4) :430 – 440, 2010. Procedural Methods in Computer Graphics Illustrative Visualization.
- [BPMG04] Carles Bosch, Xavier Pueyo, Stéphane Mérillou, and Djamchid Ghazanfarpour. A Physically Based Model for Rendering Realistic Scratches. *Computer Graphics Forum*, 23(3) :361–370, 2004.
- [Bro80] G. Brown. Shadowing by non-Gaussian random surfaces. *IEEE Transactions on Antennas and Propagation*, 28(6) :788–790, November 1980.
- [BS63] P. Beckmann and A. Spizzichino. *The scattering of electromagnetic waves from rough surfaces*. International series of monographs on electromagnetic waves. Pergamon Press; [distributed in the Western Hemisphere by Macmillan, New York], 1963.
- [Bur12] Brent Burley. Physically-based Shading at Disney. In *ACM SIGGRAPH course notes*, 2012.
- [BYRN17] Laurent Belcour, Ling-Qi Yan, Ravi Ramamoorthi, and Derek Nowrouzezahrai. Antialiasing Complex Global Illumination Effects in Path-Space. *ACM Trans. Graph.*, 36(1) :9 :1–9 :13, January 2017.
- [CCM18] Xavier Chermain, Frédéric Claux, and Stéphane Mérillou. A microfacet-based brdf for the accurate and efficient rendering of high-definition specular normal maps. *The Visual Computer*, Oct 2018.
- [CCM19] Xavier Chermain, Frédéric Claux, and Stéphane Mérillou. Glint Rendering based on a Multiple-Scattering Patch BRDF. *Computer Graphics Forum*, 2019.
- [CK17] Alejandro Conty and Christopher Kulla. Revisiting physically based shading at Imageworks. In *ACM SIGGRAPH 2017 Courses*, SIGGRAPH '17, pages 7 :1–7 :8, New York, NY, USA, 2017. ACM.

- [CT81] Robert L. Cook and Kenneth E. Torrance. A Reflectance Model for Computer Graphics. *SIGGRAPH Comput. Graph.*, 15(3) :307–316, August 1981.
- [DHd16] Jonathan Dupuy, Eric Heitz, and Eugene d’Eon. Additional Progress Towards the Unification of Microfacet and Microflake Theories. In *EGSR (EI&I)*, pages 55–63, 2016.
- [DHI⁺13] Jonathan Dupuy, Eric Heitz, Jean-Claude Iehl, Pierre Poulin, Fabrice Neyret, and Victor Ostromoukhov. Linear Efficient Antialiased Displacement and Reflectance Mapping. *ACM Trans. Graph.*, 32(6) :211 :1–211 :11, November 2013.
- [DJ18] Jonathan Dupuy and Wenzel Jakob. An Adaptive Parameterization for Efficient Material Acquisition and Rendering. *ACM Trans. Graph.*, 37(6) :274 :1–274 :14, December 2018.
- [Dup17] Jonathan Dupuy. A C++ Microfacet BRDF Fitting and Rendering Library, may 2017. https://github.com/jdupuy/dj_brdf.
- [FA19] Carmelo J. Fdez-Agüera. A Multiple-Scattering Microfacet Model for Real-Time Image Based Lighting. *Journal of Computer Graphics Techniques (JCGT)*, 8(1) :45–55, January 2019.
- [GA19] Romain Guy and Mathias Agopian. Filament, 2019. <https://google.github.io/filament/Filament.md.html>.
- [GGN18] Luis E. Gamboa, Jean-Philippe Guertin, and Derek Nowrouzezahrai. Scalable Appearance Filtering for Complex Lighting Effects. *ACM Trans. Graph.*, 37(6) :277 :1–277 :13, December 2018.
- [HBP13] Eric Heitz, Christophe Bourlier, and Nicolas Pinel. Correlation effect between transmitter and receiver azimuthal directions on the illumination function from a random rough surface. *Waves in Random and Complex Media*, 23(3) :318–335, 2013.
- [Hd14] E. Heitz and E. d’Eon. Importance Sampling Microfacet-Based BSDFs using the Distribution of Visible Normals. *Computer Graphics Forum*, 33(4) :103–112, 2014.
- [HDCD15] Eric Heitz, Jonathan Dupuy, Cyril Crassin, and Carsten Dachsbacher. The SGX Microflake Distribution. *ACM Trans. Graph.*, 34(4) :48 :1–48 :11, July 2015.
- [Hec89] Paul S Heckbert. Fundamentals of Texture Mapping and Image Warping. Technical report, EECS Department, University of California, Berkeley, Jun 1989.
- [Hei14] Eric Heitz. Understanding the Masking-Shadowing Function in Microfacet-Based BRDFs. *Journal of Computer Graphics Techniques (JCGT)*, 3(2) :48–107, June 2014.
- [Hei15] Éric Heitz. Generating Procedural Beckmann Surfaces, 2015. <https://eheitzresearch.wordpress.com/research/>.
- [Hei18] Eric Heitz. Sampling the GGX Distribution of Visible Normals. *Journal of Computer Graphics Techniques (JCGT)*, 7(4) :1–13, November 2018.

- [HH84] Paul S. Heckbert and Pat Hanrahan. Beam Tracing Polygonal Objects. *SIGGRAPH Comput. Graph.*, 18(3) :119–127, January 1984.
- [HHdD16] Eric Heitz, Johannes Hanika, Eugene d’Eon, and Carsten Dachsbacher. Multiple-scattering Microfacet BSDFs with the Smith Model. *ACM Trans. Graph.*, 35(4) :58 :1–58 :14, July 2016.
- [Hil18a] Stephen Hill. A multi-faceted exploration (part 2), jun 2018. <https://blog.selfshadow.com/2018/06/04/multi-faceted-part-2/>.
- [Hil18b] Stephen Hill. A multi-faceted exploration (part 3), jun 2018. <https://blog.selfshadow.com/2018/08/05/multi-faceted-part-3/>.
- [HNPN13] Eric Heitz, Derek Nowrouzezahrai, Pierre Poulin, and Fabrice Neyret. Filtering Color Mapped Textures and Surfaces. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D ’13*, pages 129–136, New York, NY, USA, 2013. ACM.
- [Hof19] Naty Hoffman. Fresnel Equations in RGB Considered Harmful, 7 2019. <http://renderwonk.com/publications/mam2019/>.
- [HSRG07] Charles Han, Bo Sun, Ravi Ramamoorthi, and Eitan Grinspun. Frequency Domain Normal Map Filtering. *ACM Trans. Graph.*, 26(3), July 2007.
- [Ige99] Homan Igehy. Tracing Ray Differentials. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH ’99*, pages 179–186, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.
- [JAM⁺10] Wenzel Jakob, Adam Arbree, Jonathan T. Moon, Kavita Bala, and Steve Marschner. A Radiative Transfer Framework for Rendering Materials with Anisotropic Structure. *ACM Trans. Graph.*, 29(4) :53 :1–53 :13, July 2010.
- [JdJM14] Wenzel Jakob, Eugene d’Eon, Otto Jakob, and Steve Marschner. A Comprehensive Framework for Rendering Layered Materials. *ACM Trans. Graph.*, 33(4) :118 :1–118 :14, July 2014.
- [JHY⁺14] Wenzel Jakob, Miloš Hašan, Ling-Qi Yan, Jason Lawrence, Ravi Ramamoorthi, and Steve Marschner. Discrete Stochastic Microfacet Models. *ACM Trans. Graph.*, 33(4) :115 :1–115 :10, July 2014.
- [Kaj86] James T Kajiya. The rendering equation. In *ACM SIGGRAPH computer graphics*, volume 20, pages 143–150. ACM, 1986.
- [KSK01] Csaba Kelemen and Laszlo Szirmay-Kalos. A microfacet based coupled specular-matte BRDF model with importance sampling. In *Eurographics Short Presentations*, volume 25, page 34, 2001.
- [KVDDN99] Jan J. Koenderink, Andrea J. Van Doorn, Kristin J. Dana, and Shree Nayar. Bidirectional Reflection Distribution Function of Thoroughly Pitted Surfaces. *International Journal of Computer Vision*, 31(2) :129–144, Apr 1999.
- [LJJ⁺18] Joo Ho Lee, Adrian Jarabo, Daniel S. Jeon, Diego Gutierrez, and Min H. Kim. Practical Multiple Scattering for Rough Surfaces. *ACM Trans. Graph.*, 37(6) :275 :1–275 :12, December 2018.

- [MDG01] S. Merillou, J.M. Dischler, and D. Ghazanfarpour. Surface scratches : measuring, modeling and rendering. *The Visual Computer*, 17(1) :30–45, Feb 2001.
- [MPBM03a] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A Data-driven Reflectance Model. *ACM Trans. Graph.*, 22(3) :759–769, July 2003.
- [MPBM03b] Wojciech Matusik, Hanspeter Pfister, Matthew Brand, and Leonard McMillan. Efficient Isotropic BRDF Measurement. In *Proceedings of the 14th Eurographics Workshop on Rendering, EGRW '03*, pages 241–247, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.
- [Ney98] F. Neyret. Modeling, animating, and rendering complex scenes using volumetric textures. *IEEE Transactions on Visualization and Computer Graphics*, 4(1) :55–70, Jan 1998.
- [Nic65] Fred E. Nicodemus. Directional Reflectance and Emissivity of an Opaque Surface. *Appl. Opt.*, 4(7) :767–775, Jul 1965.
- [OB10] Marc Olano and Dan Baker. LEAN Mapping. In *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D '10*, pages 181–188, New York, NY, USA, 2010. ACM.
- [ON94] Michael Oren and Shree K. Nayar. Generalization of Lambert’s Reflectance Model. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '94*, pages 239–246, New York, NY, USA, 1994. ACM.
- [Pho75] Bui Tuong Phong. Illumination for Computer Generated Pictures. *Commun. ACM*, 18(6) :311–317, June 1975.
- [Pix18] Pixar. Pixar RenderMan, 2018. <https://renderman.pixar.com/>.
- [PJH16] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically Based Rendering : From Theory to Implementation*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2016.
- [RBMS17] M. Ribardière, B. Bringier, D. Meneveaux, and L. Simonot. STD : Student’s t-Distribution of Slopes for Microfacet Based BSDFs. *Computer Graphics Forum*, 36(2) :421–429, 2017.
- [RBSM19] Mickaël Ribardière, Benjamin Bringier, Lionel Simonot, and Daniel Meneveaux. Microfacet BSDFs Generated from NDFs and Explicit Microgeometry. *ACM Trans. Graph.*, 38(5) :143 :1–143 :15, June 2019.
- [RDP05] Vincent Ross, Denis Dion, and Guy Potvin. Detailed analytical approach to the Gaussian surface bidirectional reflectance distribution function specular component applied to the sea surface. *J. Opt. Soc. Am. A*, 22(11) :2442–2453, Nov 2005.
- [RGB16] Boris Raymond, Gaël Guennebaud, and Pascal Barla. Multi-scale Rendering of Scratched Materials Using a Structured SV-BRDF Model. *ACM Trans. Graph.*, 35(4) :57 :1–57 :11, July 2016.
- [Roo18] Ton Roosendaal. Blender (2.79), 2018. www.blender.org.
- [Sch94] Christophe Schlick. An Inexpensive BRDF Model for Physically-based Rendering. *Computer Graphics Forum*, 13(3) :233–246, 1994.

- [SHHD17] Vincent Schüssler, Eric Heitz, Johannes Hanika, and Carsten Dachsbacher. Microfacet-based Normal Mapping for Robust Monte Carlo Path Tracing. *ACM Trans. Graph.*, 36(6) :205 :1–205 :12, November 2017.
- [Smi67] B. Smith. Geometrical shadowing of a random rough surface. *IEEE Transactions on Antennas and Propagation*, 15(5) :668–671, September 1967.
- [SPSH19] Dorian Saint-Pierre, Lionel Simonot, and Mathieu Hébert. Reflectance Computation for a Specular Only V-Cavity. In Shoji Tominaga, Raimondo Schettini, Alain Trémeau, and Takahiko Horiuchi, editors, *Computational Color Imaging*, pages 289–303, Cham, 2019. Springer International Publishing.
- [SW01] Frank Suykens and Yves D. Willems. Path differentials and applications. In Steven J. Gortler and Karol Myszkowski, editors, *Rendering Techniques 2001*, pages 257–268, Vienna, 2001. Springer Vienna.
- [TIS08] Art Tevs, Ivo Ihrke, and Hans-Peter Seidel. Maximum Mipmaps for Fast, Accurate, and Scalable Dynamic Height Field Rendering. In *Proceedings of the 2008 Symposium on Interactive 3D Graphics and Games, I3D '08*, pages 183–190, New York, NY, USA, 2008. ACM.
- [TK19] Yusuke Tokuyoshi and Anton S. Kaplanyan. Improved Geometric Specular Antialiasing. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games, I3D '19*, pages 8 :1–8 :8, New York, NY, USA, 2019. ACM.
- [Tok05] Michael Toksvig. Mipmapping Normal Maps. *Journal of Graphics Tools*, 10(3) :65–71, 2005.
- [TR75] T. S. Trowbridge and K. P. Reitz. Average irregularity representation of a rough surface for ray reflection. *J. Opt. Soc. Am.*, 65(5) :531–536, May 1975.
- [TS67] K. E. Torrance and E. M. Sparrow. Theory for Off-Specular Reflection From Roughened Surfaces. *J. Opt. Soc. Am.*, 57(9) :1105–1114, Sep 1967.
- [Tur19] Emmanuel Turquin. Practical multiple scattering compensation for microfacet models, 2019. https://blog.selfshadow.com/publications/turquin/ms_comp_final.pdf.
- [Vea96] Eric Veach. Non-symmetric Scattering in Light Transport Algorithms. In Xavier Pueyo and Peter Schröder, editors, *Rendering Techniques '96*, pages 81–90, Vienna, 1996. Springer Vienna.
- [Vea97] Eric Veach. *Robust Monte Carlo methods for light transport simulation*, volume 1610. Stanford University PhD thesis, 1997.
- [vGSK98] Bram van Ginneken, Marigo Stavridi, and Jan J. Koenderink. Diffuse and Specular Reflectance from Rough Surfaces. *Appl. Opt.*, 37(1) :130–139, Jan 1998.
- [WAT92] Stephen H. Westin, James R. Arvo, and Kenneth E. Torrance. Predicting Reflectance Functions from Complex Surfaces. *SIGGRAPH Comput. Graph.*, 26(2) :255–264, July 1992.

- [WB16] Beibei Wang and How Bowles. A Robust and Flexible Real-time Sparkle Effect. In *Proceedings of the Eurographics Symposium on Rendering : Experimental Ideas & Implementations*, EGSR '16, pages 49–54, Goslar Germany, Germany, 2016. Eurographics Association.
- [WHHY19] Beibei Wang, Miloš Hašan, Nicolas Holzschuch, and Ling-Qi Yan. Example-Based Microstructure Rendering with Constant Storage, 2019. <https://cs.ucsb.edu/research/tech-reports/2019-08>.
- [WMLT07] Bruce Walter, Stephen R. Marschner, Hongsong Li, and Kenneth E. Torrance. Microfacet Models for Refraction Through Rough Surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques*, EGSR'07, pages 195–206, Aire-la-Ville, Switzerland, Switzerland, 2007. Eurographics Association.
- [WWH18] Beibei Wang, Lu Wang, and Nicolas Holzschuch. Fast Global Illumination with Discrete Stochastic Microfacets Using a Filterable Model. *Computer Graphics Forum*, 37(7) :55–64, 2018.
- [XH18] Feng Xie and Pat Hanrahan. Multiple Scattering from Distributions of Specular V-grooves. *ACM Trans. Graph.*, 37(6) :276 :1–276 :14, December 2018.
- [XWZB17] Chao Xu, Rui Wang, Shuang Zhao, and Hujun Bao. Real-Time Linear BRDF MIP-Mapping. *Computer Graphics Forum*, 36(4) :27–34, 2017.
- [YHJ⁺14] Ling-Qi Yan, Miloš Hašan, Wenzel Jakob, Jason Lawrence, Steve Marschner, and Ravi Ramamoorthi. Rendering Glints on High-resolution Normal-mapped Specular Surfaces. *ACM Trans. Graph.*, 33(4) :116 :1–116 :9, July 2014.
- [YHMR16] Ling-Qi Yan, Miloš Hašan, Steve Marschner, and Ravi Ramamoorthi. Position-normal Distributions for Efficient Rendering of Specular Microstructure. *ACM Trans. Graph.*, 35(4) :56 :1–56 :9, July 2016.
- [YHW⁺18] Ling-Qi Yan, Miloš Hašan, Bruce Walter, Steve Marschner, and Ravi Ramamoorthi. Rendering Specular Microgeometry with Wave Optics. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2018)*, 37(4), 2018.
- [Zip66] Richard B Zipin. The apparent thermal radiation properties of an isothermal V-groove with specularly reflecting walls. *J. Res. NBS C*, 70 :275–280, 1966.
- [ZK16] Tobias Zirr and Anton S. Kaplanyan. Real-time Rendering of Procedural Multiscale Materials. In *Proceedings of the 20th ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, I3D '16, pages 139–148, New York, NY, USA, 2016. ACM.

Rendu basé physique de micro-reflets

Résumé : Le rendu de micro-reflets, utile pour simuler l'apparence de matériaux pailletés, de métal brossé ou de plastique rayé, est un défi théorique et technique en informatique graphique. Il implique l'utilisation de fonctions de distribution de réflectance bidirectionnelles surfaciques (\mathcal{P} -BRDFs) hautes fréquences et qui varient spatialement. Dans cette thèse, nous proposons deux nouvelles \mathcal{P} -BRDFs basées sur des cartes de normales presque parfaitement spéculaires. La première empêche toute création d'énergie grâce à une normalisation dépendante de l'empreinte du rayon, contrairement à la méthode précédente [YHMR16]. Cette normalisation est possible grâce à une nouvelle représentation d'une carte de normales en une mixture de NDFs de Beckmann décentrées et non-alignées sur les axes. La deuxième méthode améliore la première et empêche, pour la première fois, toute création *et perte* d'énergie, en simulant du multi-rebonds dans la micro-géométrie du matériau. Elle permet donc un rendu sans artefacts de surfaces opaques possédant des micro-reflets. De plus, nous donnons un algorithme d'échantillonnage optimal, utilisant la visibilité des normales. L'idée clé de cette méthode est la définition d'une V-cavité en chaque point de la surface. Pour simuler le multi-rebonds à l'intérieur, nous compensons l'énergie perdue par une modélisation simple rebond, en la réintégrant à l'aide d'une BRDF de compensation d'énergie. Nos méthodes ont le même ordre de grandeur que la méthode précédente en matière de temps de rendu et d'empreinte mémoire.

Physically based glint rendering

Abstract : Glint rendering, useful for simulating the appearance of glittery materials, brushed metal or scratched plastic, is a theoretical and technical challenge in computer graphics. It involves the use of spatially varying patch bidirectional reflectance distribution functions (\mathcal{P} -BRDFs) with high frequencies. In this thesis we propose two new \mathcal{P} -BRDFs based on specular normal maps. Unlike the previous method [YHMR16], our first BRDF prevents any creation of energy through footprint-dependent normalisation. This normalisation is possible thanks to a new representation of the normal map based on a mixture of non-centered and non-axis aligned Beckmann NDFs. The second method improves the first one and prevents, for the first time, any creation *and loss* of energy, by simulating multiple scattering in the microgeometry. It enables artifact-free rendering of opaque and sparkling surfaces. In addition, we provide an optimal sampling algorithm using the visibility information of the normals. The key idea of this method is the definition of a V-cavity for each point of the surface. To simulate multiple scattering inside it, we compensate for the energy lost by a single scattering model, by reintegrating lost energy with an energy compensation BRDF. The rendering time and memory footprint of our methods are in the same order of magnitude than previous methods.

Keywords : Reflectance functions, BRDF, microfacet, glint

Univ. Limoges, CNRS, XLIM, UMR 7252, F-87000 Limoges, France.