



PROJET 150 H

---

# Filtrage de distributions de normales pour du rendu sans aliasing

---

*Auteur :*

**Cyril LAJARGE**  
M2 CMI IIRVIJ  
Université de Strasbourg, France  
cyril.lajarge@etu.unistra.fr

*Encadrant :*

**Xavier CHERMAIN**  
Chercheur post-doctoral au laboratoire ICube  
Université de Strasbourg, France  
chermain@unistra.fr

Automne 2020

## Remerciements

Je tiens à remercier mon encadrant, Xavier CHERMAIN, chercheur post-doctoral à l'Université de Strasbourg, qui m'a accompagné tout au long de ce projet avec bienveillance. Il a su m'aider de façon constructive et pédagogique lors des difficultés que j'ai rencontrées et m'a permis d'approfondir mes connaissances en rendu basé physique.

## Notations

Symbole	Signification
$\omega_g$	normale géométrique
$\omega_m$	normale de la micro-surface
$\omega_o$	direction d'observation
$\omega_i$	direction incidente
$\omega_h$	direction à mi-chemin entre $\omega_i$ et $\omega_o$
$\omega_1 \cdot \omega_2$	produit scalaire
$\omega_1 \times \omega_2$	produit vectoriel
$\Omega$	domaine hémisphérique
$ x $	valeur absolue
$L$	radiance
$D(\omega_m)$	distribution de normales
$G(\omega_o, \omega_i, \omega_h)$	masquage-ombrage géométrique
$F(\omega_o, \omega_h)$	terme de Fresnel
$\mathcal{F}$	projection de l'empreinte du pixel sur la surface
$\mathcal{P}$	projection de l'empreinte du pixel dans le domaine des pentes

# Table des matières

<b>Remerciements</b>	<b>1</b>
<b>Notations</b>	<b>3</b>
<b>1 Contexte</b>	<b>4</b>
1.1 Rendu en informatique . . . . .	4
1.2 Présentation du sujet . . . . .	4
1.3 Moteur de rendu BGFX . . . . .	4
<b>2 Rendu basé physique</b>	<b>4</b>
2.1 BRDF à micro-facettes . . . . .	5
2.2 Distribution de normales . . . . .	5
2.2.1 Distribution des pentes . . . . .	6
2.3 Masquage-ombrage géométrique . . . . .	7
2.4 Terme de Fresnel . . . . .	7
2.5 Les différents repères en rendu . . . . .	7
<b>3 Anti-aliasing spéculaire</b>	<b>7</b>
3.1 Implémentation . . . . .	8
<b>4 Résultats</b>	<b>9</b>
<b>5 Bilan personnel</b>	<b>10</b>
<b>Annexes</b>	<b>11</b>
<b>A Procédé de Gram-Schmidt</b>	<b>11</b>
<b>B Calcul du cosinus et de la tangente pour l'angle theta</b>	<b>11</b>
<b>C Calcul du cosinus et du sinus pour l'angle phi</b>	<b>11</b>

# 1 Contexte

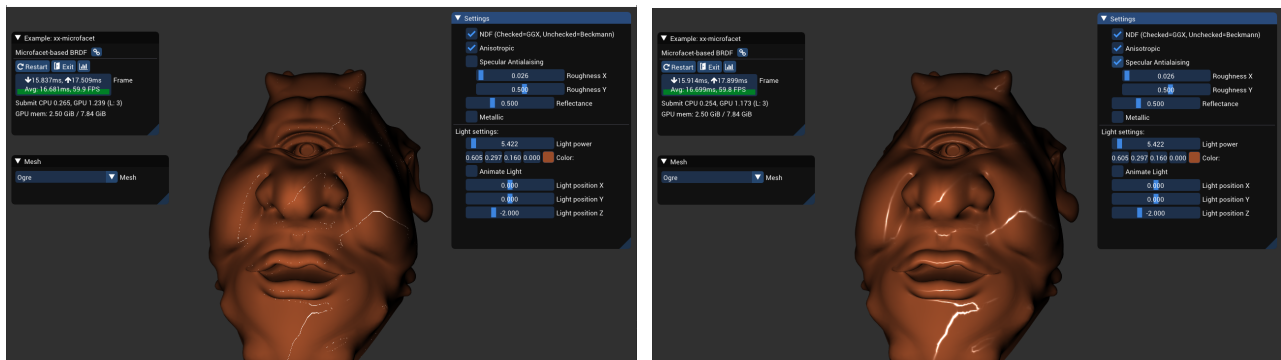
## 1.1 Rendu en informatique

Le rendu en informatique consiste à produire une image à partir d'une scène 3D virtuelle. Il existe une multitude de types de rendus différents. Nous pouvons comparer ce procédé à une tâche concrète comme la peinture où un artiste peut retranscrire sur sa toile la scène qu'il observe de façon réaliste ou au contraire selon un style particulier. Afin de générer ces images virtuelles, différents procédés peuvent être utilisés comme par exemple le lancer de rayons qui va simuler les rayons lumineux.

## 1.2 Présentation du sujet

Durant la dernière décennie, la BRDF spéculaire de Phong a été peu à peu remplacée par la BRDF spéculaire à micro-facettes car cette dernière donne des résultats plus proches de la réalité et est basée physique. Ce changement a été opéré pour les applications non temps-réel (lancer de rayons) et temps-réel (rastérisation). Ce passage à une BRDF basée physique n'est cependant pas une solution miracle et un problème subsiste : de par la discrétisation en pixels, de l'aliasing *spéculaire* peut être visible (figure 1a). En effet, ce phénomène touche souvent le rendu temps-réel de par le fait que peu d'échantillons sont utilisés par pixels. Il peut se manifester de manière géométrique lorsqu'un maillage est très fin mais il peut aussi être spéculaire quand une petite lumière est utilisée et que l'objet est proche d'un miroir parfait ainsi que complexe en terme de courbure.

Le travail de ce projet consiste dans un premier temps en l'implémentation d'une BRDF spéculaire à micro-facettes (figure 1a) [Heitz, 2014] puis d'une technique d'anti-aliasing spéculaire (figure 1b) développée par [Kaplanyan et al., 2016].



(a) Sans anti-aliasing spéculaire

(b) Avec anti-aliasing spéculaire

FIGURE 1 – Comparaison entre une BRDF à micro-facettes pourvue et dépourvue d'anti-aliasing spéculaire.

## 1.3 Moteur de rendu BGFX

Afin de se concentrer sur l'implémentation de la BRDF ainsi que l'antialiasing spéculaire, nous avons utilisé le moteur de rendu open-source BGFX, qui met à disposition une API C++ multi-plateforme bas niveau utilisée par de grands noms comme Minecraft. Des outils sont également mis à disposition comme **shaderc** qui permet de compiler les shaders de BGFX (dont la syntaxe est très proche de GLSL) dans d'autres langages de shader. L'utilisation de maillages est également simplifiée grâce aux outils **geomtryc** et **geometryv** qui permettent respectivement de convertir les maillages dans un format compréhensible par BGFX et de les visualiser (figure 3). Pour interagir en temps réel avec BGFX, la librairie d'interface utilisateur ImGui est intégrée au moteur (figure 2).

# 2 Rendu basé physique

Le rendu basé physique est un domaine qui a gagné en popularité ces dernières années car il permet un résultat visuel proche de la réalité. En effet, ce type de rendu s'appuie sur des modèles physiques de propagation et d'interaction de la lumière avec des matériaux, en utilisant des propriétés réelles telle que le coefficient d'absorption ou l'indice de réfraction. Comme à la manière d'un capteur d'appareil photo, nous ne cherchons pas ici à déterminer une couleur mais à mesurer la quantité d'énergie (le nombre de photons) reçue par notre capteur virtuel. Le rendu basé physique permet d'une part de modéliser de façon fidèle des objets du réel rendant

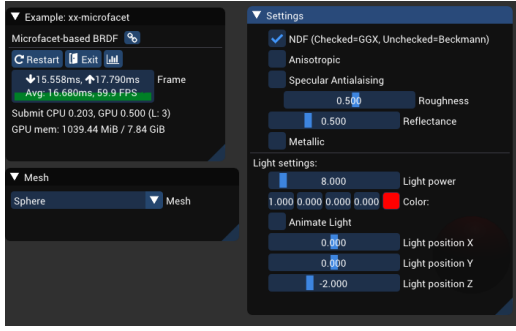


FIGURE 2 – Interface graphique avec ImGUI

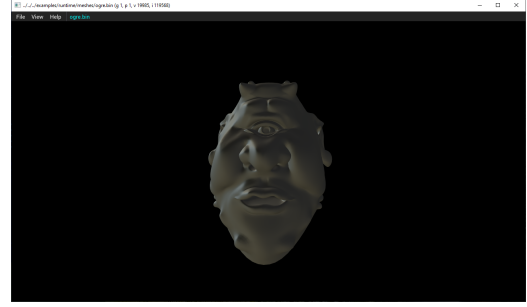


FIGURE 3 – Outil geometryv

possible la réalisation d’analyses et de simulations numériques [Heitz, 2014] et d’autre part, de déterminer des critères objectifs en terme d’évaluation du réalisme.

Une des équations les plus importantes en informatique graphique est probablement l’équation de rendu (1). Elle permet de modéliser le principe de conservation de l’énergie.

$$L_o(\omega_o) = L_e(\omega_o) + \int_{\Omega} f(\omega_i, \omega_o) L_i(\omega_i) (\omega_i \cdot n) d\omega_i \quad (1)$$

La radiance arrivant depuis la direction d’observation ( $L_o$ ) est égale à la radiance émise par le matériau dans la direction d’observation ( $L_e$ ), à laquelle on ajoute la somme de toutes les radiances incidentes ( $L_i$ ), celles-ci étant pondérées par l’angle entre la direction incidente et la normale, ainsi que la fonction de distribution de réflectance bidirectionnelle (Bidirectional Reflectance Distribution Function en anglais d’où **BRDF**) ( $f$ ).

La partie qui nous intéresse le plus dans cette équation est la BRDF. Souvent composée d’une partie diffuse et d’une partie spéculaire, c’est elle qui caractérise la manière dont le matériau diffuse et reflète la lumière. Pour la partie diffuse, c’est souvent une BRDF lambertienne constante de valeur  $\frac{1}{\pi}$  qui est utilisée. La partie spéculaire de la BRDF permet de représenter des matériaux métalliques ou plastiques. Les premiers modèles comme par exemple celui de Blinn-Phong ne se basent pas sur des propriétés physiques ce qui les rend difficilement utilisables pour du rendu réaliste. Les BRDFs utilisées pour faire du rendu basé physique sont quant à elles basées sur un modèle dit à micro-facettes.

## 2.1 BRDF à micro-facettes

Une BRDF à micro-facettes suppose que la surface d’un objet de normale géométrique  $\omega_g$  (macrosurface, figure 4, noir) est composée de micro-facettes orientées dans différentes directions  $\omega_m$  (microsurface, figure 4, rouge). Ce modèle bi-échelle est dit basé physique contrairement au modèle de Phong, car il utilise un paramètre de rugosité qui peut être mesuré sur des surfaces réelles. Une manière de représenter une micro-surface la plus fidèlement possible est d’avoir un maillage extrêmement fin pour modéliser ces aspérités microscopiques, mais cela engendre une quantité de données beaucoup trop importante et rend cette solution difficilement utilisable. La fonction de distribution des micro-normales permet d’approximer la micro-géométrie tout en gardant un maillage relativement simple. Les modèles de distribution analytique connus, comme par exemple une distribution gaussienne, permettent souvent d’obtenir une bonne approximation.

Dans le cas où les micro-facettes sont parfaitement spéculaires, c’est-à-dire qu’elles sont des miroirs parfaits, l’équation (2) permet de calculer la partie spéculaire d’une BRDF à micro-facettes [Walter et al., 2007]. Elle est composée d’un certain nombre de termes qui seront décrits dans les sections suivantes.

$$f(\omega_i, \omega_o) = \frac{F(\omega_o, \omega_h) G(\omega_o, \omega_i, \omega_h) D(\omega_h)}{4|\omega_g \cdot \omega_o||\omega_g \cdot \omega_i|} \quad (2)$$

## 2.2 Distribution de normales

Le terme  $D$  de la formule décrivant une BRDF à micro-facettes fait référence à la distribution des normales. Effectivement, ce terme, qui est une probabilité, détermine la proportion de facettes orientées selon la direction  $\omega_h = \frac{\omega_o + \omega_i}{|\omega_o + \omega_i|}$ . Ce dernier est le vecteur à mi-chemin entre la direction d’observation  $\omega_o$  et la direction incidente  $\omega_i$ . Plus  $\omega_h$  est proche de la normale géométrique  $\omega_g$ , plus la direction d’observation est alignée avec la direction de réflexion et plus la contribution spéculaire est importante.

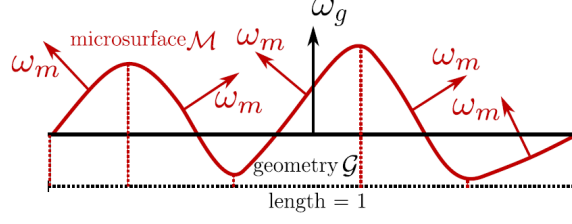


FIGURE 4 – Représentation schématique de la micro-surface [Heitz, 2014]  
(noir = échelle macroscopique, rouge = échelle microscopique)

Cette distribution possède un paramètre physique qui est contrôlable : la rugosité. Cette dernière représente l'écart-type des pentes des micro-normales. Plus la rugosité est importante, plus les micro-facettes vont avoir des pentes fortes, plus les rayons réfléchis vont partir dans des directions différentes, et plus la tâche spéculaire sera étendue et peu intense. À contrario, si la rugosité est faible, les rayons vont avoir tendance à être réfléchis dans une même direction et donc créer une tâche spéculaire plus concentrée et intense (figure 5).

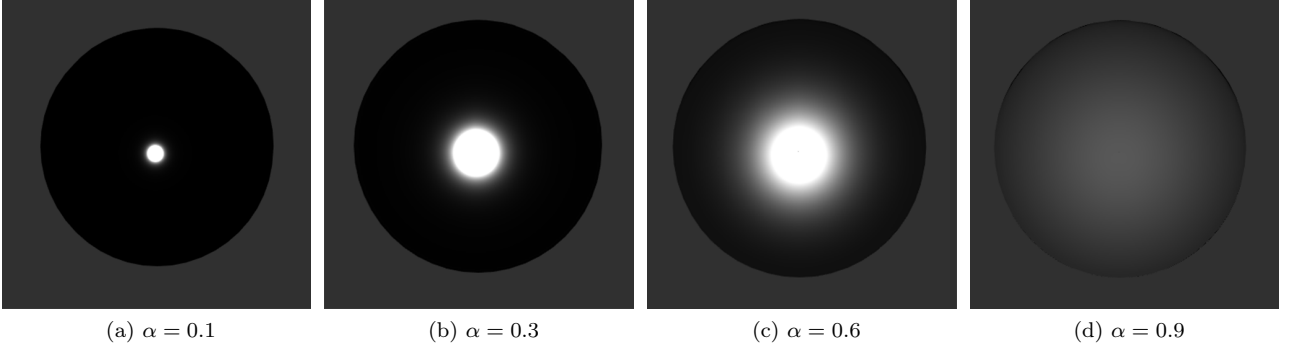


FIGURE 5 – Résultat visuel pour différentes rugosités  $\alpha$ . La lumière est petite et est située au niveau de la caméra.

### 2.2.1 Distribution des pentes

Les micro-facettes sont définies par leur normale  $\omega_m = (x_m, y_m, z_m)$ , mais étant donné que la normale est orthogonale à la pente, elles peuvent également être représentées par leur pente. Il est donc possible de modéliser la distribution des normales (3) par une distribution des pentes des micro-facettes  $P^{22}$ . La normale appartenant à  $\mathbb{R}^3$ , la pente  $\tilde{m}$  est dans  $\mathbb{R}^2$  et vaut  $(-\frac{x_m}{z_m}, -\frac{y_m}{z_m})$ . La distribution de normales est liée à la distribution de pentes par la relation suivante :

$$D(\omega_m) = \frac{P^{22}(x_{\tilde{m}}, y_{\tilde{m}})}{\cos^4 \theta_m}. \quad (3)$$

Deux modèles de distributions sont généralement utilisés pour la distribution des pentes  $P^{22}$  : la distribution de Beckmann [Beckmann and Spizzichino, 1963] et la distribution GGX [Walter et al., 2007]. Dans leurs formes les plus simples (4,5), ces distributions sont dites isotropes, c'est-à-dire que la rugosité  $\alpha$  est la même dans les deux directions de la surface. Elles peuvent cependant aussi avoir une forme anisotrope, c'est-à-dire avec un coefficient  $\alpha_x$  pour la rugosité selon  $x$  et un coefficient  $\alpha_y$  pour la rugosité selon  $y$ , déduite de l'équation isotrope.

$$P_{GGX}^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi \alpha^2 \left(1 + \frac{x_{\tilde{m}}^2 + y_{\tilde{m}}^2}{\alpha^2}\right)^2} \quad (4)$$

$$P_{BECKMANN}^{22}(x_{\tilde{m}}, y_{\tilde{m}}) = \frac{1}{\pi \alpha^2} \exp\left(-\frac{x_{\tilde{m}}^2 + y_{\tilde{m}}^2}{\alpha^2}\right) \quad (5)$$

## 2.3 Masquage-ombrage géométrique

Le terme *G* faire référence à ce qu'on appelle le masquage-ombrage. De par le fait que la surface n'est pas lisse au niveau microscopique, selon la direction d'observation, certaines micro-facettes sont cachées par d'autres. Effectivement, plus la direction d'observation est rasante, plus la probabilité que la micro-facette soit masquée est importante. Le masquage permet alors de donner la proportion de micro-facettes visibles depuis la direction d'observation. L'ombrage quant à lui est un masquage qui se base par rapport à la direction d'incidence. En effet, il permet de déterminer la proportion de micro-facettes visibles depuis la lumière. Plus la rugosité est importante, plus la pente des micro-facettes est importante et ainsi plus la probabilité que la micro-facette ne soit pas visible par la lumière est forte.

Il existe deux modèles respectant la règle de conservation de l'énergie permettant de modéliser ces phénomènes : le modèle de Smith et le modèle à V-cavités. Le premier suppose que les micro-facettes ne sont pas corrélées entre elles et que le masquage peut être séparé en un masquage local et un masquage global. Le second suppose que chaque point de la surface peut être décomposé en une infinité de micro-cavités en forme de V plus ou moins profondes. De nos jours, le modèle de Smith est majoritairement utilisé car les résultats obtenus sont plus proches de la réalité. Pour plus de détails techniques, [Heitz, 2014] a écrit un état de l'art sur les fonctions de masquage-ombrage pour les BRDFs à micro-facettes.

## 2.4 Terme de Fresnel

D'après les lois de conservation de l'énergie, une partie de celle-ci est restituée et l'autre est absorbée. L'absorption de l'énergie par les surfaces peut être modélisée par les équations de Fresnel. En effet, ces dernières permettent de déterminer la proportion de lumière réfléchi (restituée) par rapport à la lumière réfractée (absorbée). Le terme de Fresnel dépend de paramètres physiques tels que l'indice de réfraction et le coefficient d'absorption, ce qui le rend compatible pour du rendu basé physique. Il dépend également de la direction d'observation. En effet, plus l'angle polaire de la direction d'observation est proche de  $90^\circ$ , plus le matériau reflète la lumière. Nous pouvons par exemple remarquer cela lorsque nous observons une grande étendue d'eau. Si l'on regarde au loin, la lumière est très réfléchi par l'eau tandis que si l'on regarde juste devant nous, on peut facilement voir à travers l'eau. On mesure ce phénomène avec la réflectance, qui est le rapport entre le flux lumineux incident et le flux lumineux réfléchi. De par leur complexité, les équations de Fresnel sont implémentées à l'aide de l'approximation de Schlick [Schlick, 1994] qui dépend de la réflectance de base de la surface (la réflectance pour un angle de  $0^\circ$ ) ainsi que de la direction d'observation et la normale.

## 2.5 Les différents repères en rendu

En rendu, il est courant d'utiliser plusieurs repères. Afin de pouvoir visualiser une image à travers une caméra virtuelle, une succession de changements de repères est nécessaire. Ces transformations sont effectuées avec ce que l'on appelle des matrices de transformations. Les plus utilisées en rendu sont les suivantes :

1. Du repère de l'objet au repère du monde à l'aide de la transformation **Model**.
2. Du repère du monde au repère de la caméra à l'aide de la transformation **View**.
3. Du repère de la caméra à la projection sur le capteur de la caméra à l'aide de la transformation **Projection**.

La transformation totale peut être effectuée en multipliant successivement les matrices de transformations.

On peut notamment avoir recours à d'autres repères pour simplifier certains calculs. C'est le cas du repère **Tangent-Binormal-Normal (TBN)** qui simplifie le calcul des versions anisotropiques des distributions de normales, par exemple pour les fonctions trigonométriques (annexes B et C). Afin de transformer nos directions dans le repère TBN, il est nécessaire de définir une base orthonormée composée de la normale à la surface pour ce vertex ainsi que d'une tangente et d'une bitangente. Le procédé de Gram-Schmidt (annexe A) permet d'effectuer cela.

## 3 Anti-aliasing spéculaire

Le rendu de tâches spéculaires très brillantes de manière stable est un problème important en rendu. En effet, lorsque la surface a des zones de forte courbure et une rugosité très faible, c'est-à-dire que la surface est très réfléchissante, il est possible que la tâche spéculaire ait une taille inférieure à celle d'un pixel et que dans certains cas elle n'apparaisse pas après la discrétisation. Dans le cas du rendu par lancer de rayons, ce problème peut être surmonté en effectuant du sur-échantillonnage mais cela rallonge le temps de calcul. Dans le cas du rendu temps réel, pour garder un temps de calcul faible, le nombre d'échantillons par pixel est souvent très petit voire



même unique. Afin d'atténuer ce phénomène, [Kaplanyan et al., 2016] ont proposé une méthode basée sur un filtrage de la distribution de normales utilisée dans une BRDF spéculaire à micro-facettes.

L'idée est de simuler un sur-échantillonnage du pixel en intégrant la distribution de normales sur la projection  $\mathcal{F}$  du pixel sur la surface  $x$  (figure 6, gauche). Pour faire cela, nous allons calculer l'intensité lumineuse  $I$  en moyennant la BRDF  $f$  sur l'ensemble de l'empreinte du pixel  $\mathcal{F}$ . Initialement, d'autres termes entrent en jeu, mais comme la région d'intégration  $\mathcal{F}$  est très petite par rapport à la distance entre la surface et la caméra ou la source de lumière, ils sont supposés constants et peuvent donc être sortis de l'intégrale. Ces termes supposés constants sont représentés par  $C_p$  dans l'équation suivante :

$$I = C_p \int_{\mathcal{F}} f(\omega_o(x), \omega_i(x)) dx. \quad (6)$$

Ensuite, la BRDF  $f$  se décompose selon l'équation (2) et tous les termes, mis à part celui de la NDF, sont également considérés constants et mis dans  $C_p$  :

$$I = C_p \int_{\mathcal{F}} D(\omega_h(x)) dx. \quad (7)$$

Notez que le vecteur à mi-chemin  $\omega_h = (x_h, y_h, z_h)$  entre la lumière et l'oeil varie selon la position  $x$  (figure 6, gauche). En exprimant la NDF dans le domaine des pentes (equation 3) et en changeant le domaine d'intégration des positions sur la surface  $x \in \mathcal{F}$  vers les pentes à mi-chemin  $\tilde{h} \in \mathcal{P}$ , nous avons alors l'intégrale suivante (figure 6, droite) :

$$I = C_p \int_{\mathcal{P}} P^{22}(\tilde{h}) d\tilde{h}. \quad (8)$$

La pente à mi-chemin  $\tilde{h} = (x_{\tilde{h}}, y_{\tilde{h}}) = (-x_h/z_h, -y_h/z_h)$  est associée à la direction à mi-chemin  $\omega_h$ . On en déduit que pour moyennner les contributions lumineuses dans l'empreinte du pixel, il faut moyennner les distributions de normales dans le domaine des pentes. Cette dernière équation peut être transformée en appliquant un filtre passe-bas  $K$  à la NDF. Il a pour effet de borner l'intégrale au domaine  $\mathcal{P}$  et nous permet d'intégrer sur  $\mathbb{R}^2$  :

$$I = C_p \int_{\mathbb{R}^2} K(\tilde{h}) P^{22}(\tilde{h}) d\tilde{h}. \quad (9)$$

Maintenant, il faut déterminer la taille du filtre  $K$ , c'est-à-dire la région représentant l'empreinte du pixel  $\mathcal{F}$  dans le domaine des pentes. Pour cela, la direction à mi-chemin  $\omega_h$  est exprimée dans le plan tangent à la surface et sa pente associée  $\tilde{h}$  est calculée. Ensuite, afin d'obtenir la projection dans le domaine des pentes, il suffit de calculer les différences finies de  $\tilde{h}$  par rapport à un décalage d'un pixel  $p = (x_p, y_p)$  en  $x$  et en  $y$ . La différence finie selon la direction  $x$  en espace écran est notée  $\Delta\tilde{h}/\Delta x_p$  et la différence finie selon la direction  $y$  est notée  $\Delta\tilde{h}/\Delta y_p$ . Ces deux vecteurs 2D déterminent alors la taille du filtre passe-bas 2D  $K$ .

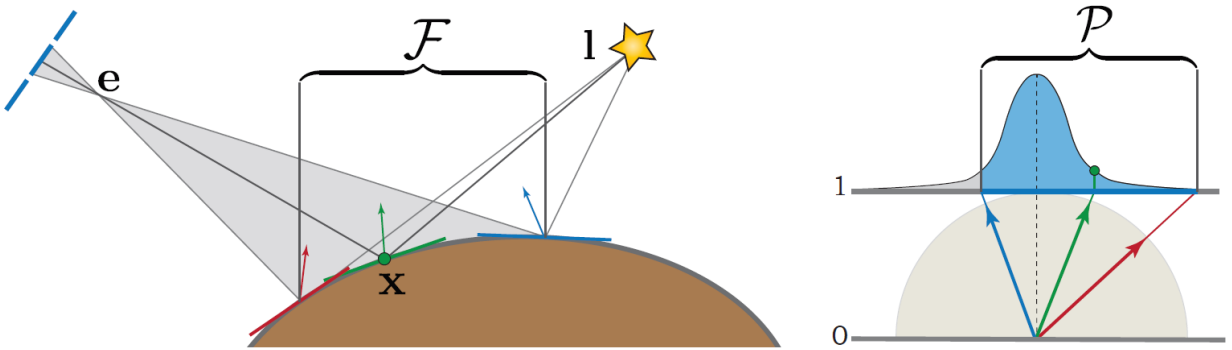


FIGURE 6 – À gauche, projection de l'empreinte du pixel sur la surface. À droite, projection de l'empreinte du pixel dans le domaine des pentes [Kaplanyan et al., 2016].

### 3.1 Implémentation

L'intégrale dans l'équation 9 a une forme close analytique si le filtre  $K$  est gaussien et si la distribution de pentes  $P^{22}$  est aussi gaussienne. Nous avons vu dans la section 2.2.1 que la distribution de Beckmann est une

gaussienne dans l'espace des pentes (équation 5). Sa matrice de covariance dans le cas anisotrope est :

$$\Sigma_B = \begin{pmatrix} \frac{\alpha_x^2}{2} & 0 \\ 0 & \frac{\alpha_y^2}{2} \end{pmatrix} \quad (10)$$

avec  $\alpha_x$  la rugosité le long de la tangente à la surface et  $\alpha_y$  la rugosité le long de la bitangente.

Nous pouvons considérer que le filtre  $K$  est gaussien et utiliser les différences finies  $\Delta\tilde{h}/\Delta x_p$  et  $\Delta\tilde{h}/\Delta y_p$  pour déterminer les écarts-types de cette gaussienne. Sur GPU et avec OpenGL, nous pouvons calculer efficacement les différences finies pixel à pixel avec les fonctions `dFdx` et `dFdy`. La matrice de covariance du filtre  $K$  est [Kaplanyan et al., 2016] :

$$\Sigma_K = \begin{pmatrix} 0,5(|\Delta x_{\tilde{h}}/\Delta x_p| + |\Delta x_{\tilde{h}}/\Delta y_p|)^2 & 0 \\ 0 & 0,5(|\Delta y_{\tilde{h}}/\Delta x_p| + |\Delta y_{\tilde{h}}/\Delta y_p|)^2 \end{pmatrix}. \quad (11)$$

Le coefficient 0,5 permet d'éviter que les filtres ne se chevauchent trop pixel à pixel.

Afin d'obtenir la matrice de covariance de la NDF de Beckmann filtrée, il faut calculer la convolution de la NDF avec le filtre passe-bas. Les deux distributions étant gaussiennes, la convolution donne une nouvelle gaussienne dont la matrice de covariance est l'addition des deux matrices de covariance (équations 10 et 11) :

$$\Sigma_{B_{filtre}} = \Sigma_K + \Sigma_B = \begin{pmatrix} 0,5(|\Delta x_{\tilde{h}}/\Delta x_p| + |\Delta x_{\tilde{h}}/\Delta y_p|)^2 + \frac{\alpha_x^2}{2} & 0 \\ 0 & 0,5(|\Delta y_{\tilde{h}}/\Delta x_p| + |\Delta y_{\tilde{h}}/\Delta y_p|)^2 + \frac{\alpha_y^2}{2} \end{pmatrix}. \quad (12)$$

Par identification, nous pouvons donc calculer les nouvelles rugosités de la distribution de Beckmann anisotrope avec les relations suivantes :

$$\alpha_{x_{filtre}} = \sqrt{(|\Delta x_{\tilde{h}}/\Delta x_p| + |\Delta x_{\tilde{h}}/\Delta y_p|)^2 + \alpha_x^2} \quad (13)$$

et

$$\alpha_{y_{filtre}} = \sqrt{(|\Delta y_{\tilde{h}}/\Delta x_p| + |\Delta y_{\tilde{h}}/\Delta y_p|)^2 + \alpha_y^2} \quad (14)$$

C'est cette formule qui est implémentée et qui permet d'augmenter la rugosité en fonction de la courbure.

## 4 Résultats

Les résultats obtenus en utilisant la méthode de [Kaplanyan et al., 2016] sont satisfaisants. En effet, des petites tâches spéculaires supprimées par la rasterisation sont récupérées, ce qui améliore le réalisme du rendu. Nous observons bien la quantité de détail perdue, due à l'aliasing spéculaire, en utilisant un matériau métallique (figure 7). Par ailleurs, le scintillement lié à l'aliasing spéculaire observé sur l'objet lors du mouvement d'une lumière disparaît en filtrant la NDF, laissant place à des tâches spéculaires continues. En soi, nous constatons que le filtrage de la NDF revient à augmenter la rugosité du matériau.



(a) Sans anti-aliasing spéculaire

(b) Avec anti-aliasing spéculaire

FIGURE 7 – Résultats pour un matériau métallique.

Dans notre cas, les distributions de normales sont utilisées avec un facteur de corrélation nul. Par conséquent, les rugosités sont alignées seulement selon la tangente et la bitangente. Pour avoir un résultat plus fin et moins grossier, c'est-à-dire avoir la possibilité d'aligner les rugosités sur des axes arbitraires dans le plan tangent, un coefficient de corrélation pourrait être intégré.

## 5 Bilan personnel

Ayant suivi le cours de Visualisation et Vision enseigné ce semestre par M. Dischler, j'ai à travers ce projet approfondi ma compréhension de certaines techniques et notions vues en cours, comme le rendu temps réel basé physique et les BRDFs. De par la nature du sujet, j'ai pris connaissance des limitations et problématiques liées au rendu temps réel notamment au niveau de l'aliasing *spéculaire*. Le fait de devoir implémenter une BRDF à micro-facettes de zéro m'a permis d'étudier plus en profondeur la signification et l'intérêt de chaque notion qui la compose. J'ai par la suite découvert et implémenté une technique permettant d'atténuer l'aliasing spéculaire par le biais du filtrage de la distribution de normales.

Au niveau de l'organisation du travail, j'ai pris conscience de l'intérêt de bien se documenter sur la méthode et de la comprendre avant de commencer à l'implémenter. J'ai également constaté l'importance de la recherche d'informations et de précisions complémentaires sur les notions complexes, afin de pouvoir les appréhender d'une autre manière. Pour conclure, ce projet m'a permis d'améliorer mes connaissances en rendu informatique par le biais des recherches que j'ai effectuées ainsi que ma méthodologie de travail, tant au niveau de l'organisation que de la communication avec les autres membres du projet.

L'ensemble du code source est disponible dans un exemple BGFX intitulé **xx-specular-antialiasing** à l'adresse Git suivante : <https://github.com/cyrillajarge/specular-antialiasing>.

## Références

- [Beckmann and Spizzichino, 1963] Beckmann, P. and Spizzichino, A. (1963). *The scattering of electromagnetic waves from rough surfaces*. Pergamon Press.
- [Heitz, 2014] Heitz, E. (2014). Understanding the Masking-Shadowing Function in Microfacet-Based BRDFs. *Journal of Computer Graphics Techniques*, 3(2) :48–107.
- [Kaplanyan et al., 2016] Kaplanyan, A. S., Hill, S., Patney, A., and Lefohn, A. (2016). Filtering Distributions of Normals for Shading Antialiasing. In *Proc. of ACM SIGGRAPH / Eurographics conference on High Performance Graphics*, page 151–162.
- [Schlick, 1994] Schlick, C. (1994). An inexpensive brdf model for physically-based rendering. *Computer Graphics Forum*, 13(3) :233–246.
- [Walter et al., 2007] Walter, B., Marschner, S. R., Li, H., and Torrance, K. E. (2007). Microfacet Models for Refraction Through Rough Surfaces. In *Proc. Eurographics Symposium on Rendering*, pages 195–206.

# Annexes

## A Procédé de Gram-Schmidt

Le procédé de Gram-Schmidt permet d'obtenir une base orthonormée à partir d'une famille libre de vecteurs. Le principe de ce procédé est de calculer le vecteur projeté d'un vecteur  $\vec{v}$  sur les autres et de soustraire ces vecteurs projetés à  $\vec{v}$ . Le projeté d'un vecteur  $\vec{v}$  sur un vecteur  $\vec{u}$  est défini de la manière suivante :

$$proj_{\vec{u}}(\vec{v}) = \frac{\vec{u} \cdot \vec{v}}{\vec{u} \cdot \vec{u}} \vec{u}. \quad (15)$$

Soient  $\vec{u}_1, \vec{u}_2$  et  $\vec{u}_3$  trois vecteurs dans  $\mathbb{R}^3$ . Afin d'obtenir une base orthonormée, les calculs suivants sont effectués :

$$\begin{aligned} \vec{v}_1 &= \vec{u}_1 \\ \vec{v}_2 &= \vec{u}_2 - proj_{\vec{v}_1}(\vec{u}_2) \\ \vec{v}_3 &= \vec{u}_3 - proj_{\vec{v}_1}(\vec{u}_3) - proj_{\vec{v}_2}(\vec{u}_3) \end{aligned} \quad (16)$$

Il suffit ensuite de normaliser les vecteurs. La figure 8 permet de visualiser le procédé pour deux vecteurs  $\vec{u}_1$  et  $\vec{u}_2$  dans  $\mathbb{R}^2$ . Nous obtenons bien deux vecteurs  $\vec{v}_1$  et  $\vec{v}_2$  orthogonaux.

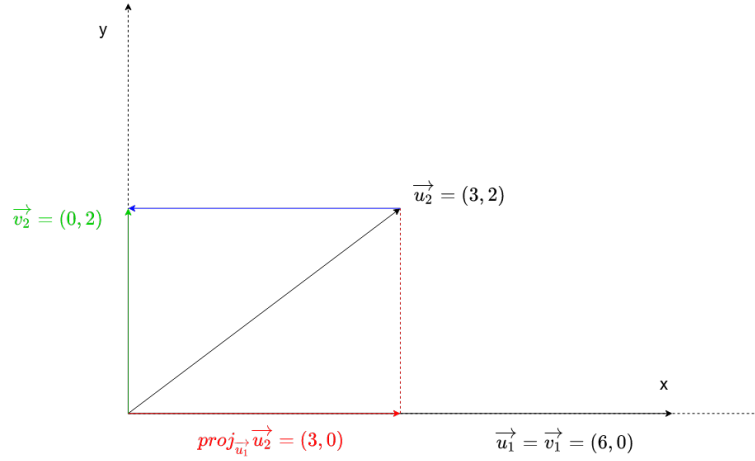


FIGURE 8 – Exemple du procédé de Gram-Schmidt dans  $\mathbb{R}^2$

## B Calcul du cosinus et de la tangente pour l'angle theta

Dans le repère TBN, la normale est représentée par le vecteur  $\text{vec3}(0.0, 0.0, 1.0)$ . Par conséquent, pour un vecteur  $\vec{k}$  :

$$\cos(\theta_k) = \vec{n} \cdot \vec{k} = z_k. \quad (17)$$

En se servant des équations trigonométriques,  $\sin(\theta_k)$  est aisément calculable :

$$\sin(\theta_k) = 1 - \cos(\theta_k)^2. \quad (18)$$

$\tan(\theta_k)$  se déduit de façon triviale de  $\sin(\theta_k)$  et  $\cos(\theta_k)$ .

## C Calcul du cosinus et du sinus pour l'angle phi

Afin de calculer les versions anisotropiques des distributions de normales en se servant de l'espace TBN, il est nécessaire de calculer  $\cos(\phi_k)$  et  $\sin(\phi_k)$ . Nous savons que :

$$(x_{\vec{k}}, y_{\vec{k}}) = (-\tan \theta_k \cos \Phi_k, -\tan \theta_k \sin \Phi_k). \quad (19)$$

Donc,

$$\begin{aligned} \cos \Phi_k &= -\frac{x_{\vec{k}} \cos \theta_k}{\sin \theta_k} \\ \sin \Phi_k &= -\frac{y_{\vec{k}} \cos \theta_k}{\sin \theta_k}. \end{aligned} \quad (20)$$

Or,  $x_{\tilde{k}} = -\frac{x_k}{z_k}$ ,  $y_{\tilde{k}} = -\frac{y_k}{z_k}$  et  $\cos \theta_k = z_k$ .

Par conséquent,  $\cos(\phi_k)$  et  $\sin(\phi_k)$  sont aisément calculables de la manière suivante :

$$\begin{aligned}\cos \Phi_k &= \frac{x_k}{\sin \theta_k} \\ \sin \Phi_k &= \frac{y_k}{\sin \theta_k}\end{aligned}\tag{21}$$