Path tracing
L'algorithme de Monte Carlo pour la résolution de l' équation du rendu

Babahenini Med Chaouki

Qu’est-ce que l’illumination globale ?

- Les algorithmes d’illumination globale sont ceux qui, en déterminant la lumière éclairant un point, tiennent compte non seulement de **la lumière qui a pris un chemin partant directement d’une source lumineuse** (illumination directe), mais également de **la lumière ayant subi la réflexion d’autres surfaces** dans la scène (illumination indirecte).

Qu’est-ce que l’illumination globale ?

- Dans la pratique, **des rayons émanent d’une source lumineuse** dans toutes les directions et bombardent la scène. Quand ces photons frappent un objet, certains sont bloqués et d’autres sont reflétés et réfractés.
- Ces derniers (les rayons non bloqués) **vont alors frapper d’autres surfaces**. Le processus continue ainsi pendant un certain nombre d’itérations. Ce procédé permet notamment de **voir les objets d’une scène qui ne sont pas illuminés directement par une lumière** ou les réflexions d’une surface brillante.

Qu’est-ce que l’illumination globale ?

(a) Illumination locale

(b) Illumination globale

Qu’est-ce que l’illumination globale ?

- L’objectif de l’illumination globale est de **simuler toutes les réflexions de lumière** dans un modèle, de manière à prédire correctement l’intensité en tout point de la scène.
- Plusieurs algorithmes ont été développés pour calculer l’illumination globale. Ceux-ci sont (la plus part du temps) répartis en deux familles :
  - les techniques de **lancer de rayon**
  - les techniques de **radiosité**

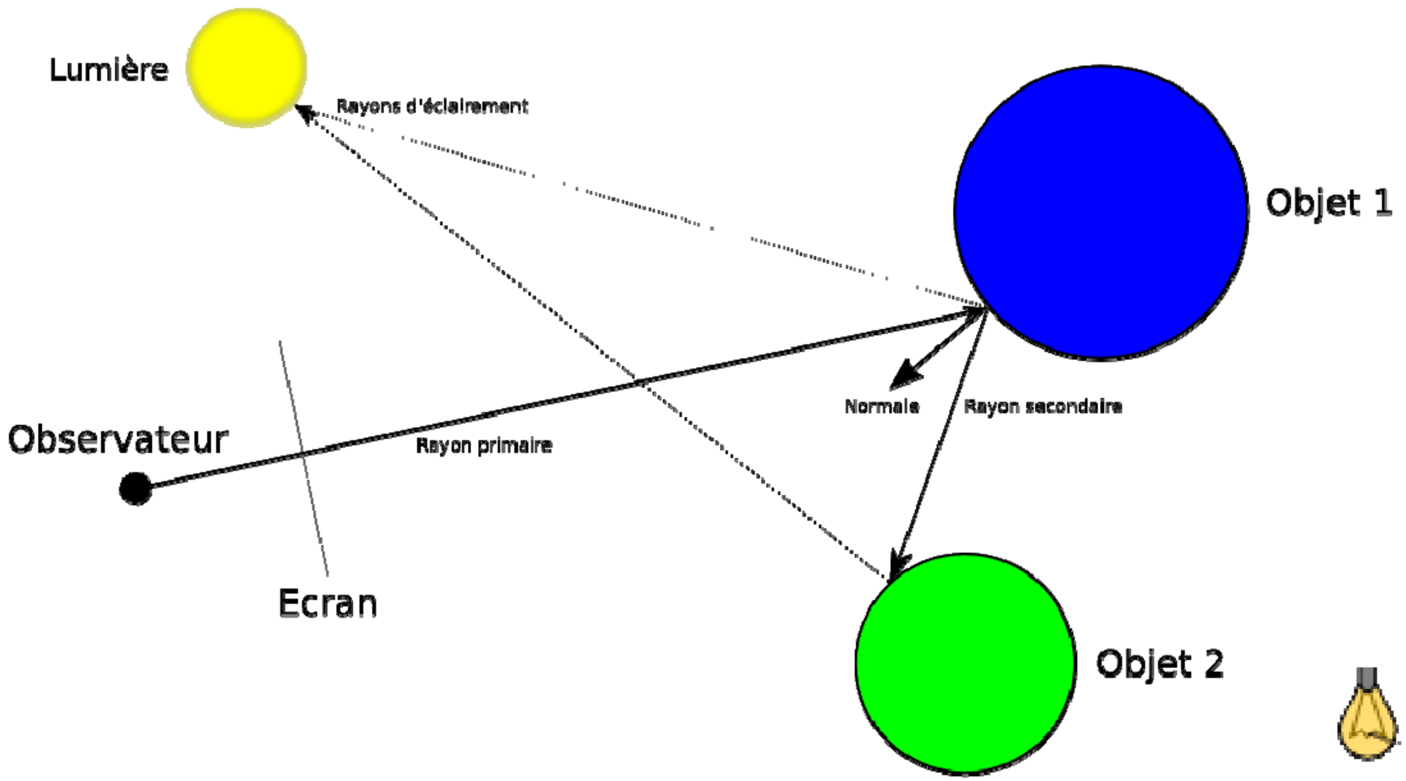
Rappels sur le lancer de rayon

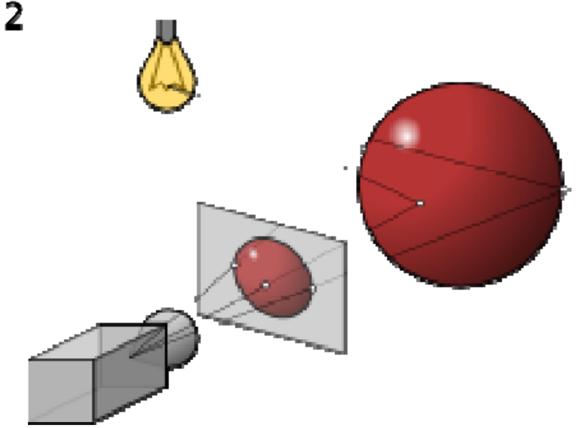
- Le principe derrière le lancer de rayon est que **le parcours de lumière peut être analysé en sens inverse**, en partant de l'observateur et en allant jusqu'à la source.
- Pour fonctionner, un lanceur de rayons doit connaître
  - la **position de l'observateur**
  - le **plan image** (direction de vue et champ de vision)
  - la **description de la scène** (géométrie, matériaux et sources de lumière)

Rappels sur le lancer de rayon

- L’objectif est d’**évaluer la couleur de chaque pixel dans le plan image**. Pour y arriver, on envoie un ou plusieurs rayons à travers chaque pixel, en partance de l’observateur.
- Par la suite, on appellera ces rayons les **rayons primaires**.
- Pour calculer la couleur associée à un rayon primaire, on trouve **le premier objet de la scène intersecté par le rayon**. On obtient alors le vecteur normal à la surface intersectée ainsi que les propriétés matérielles de l’objet.
- On effectue alors deux tâches :
  - Évaluer l’illumination **provenant des sources de lumière**
  - Évaluer l’illumination **obtenue par réflexion/réfraction spéculaire**

Rappels sur le lancer de rayon





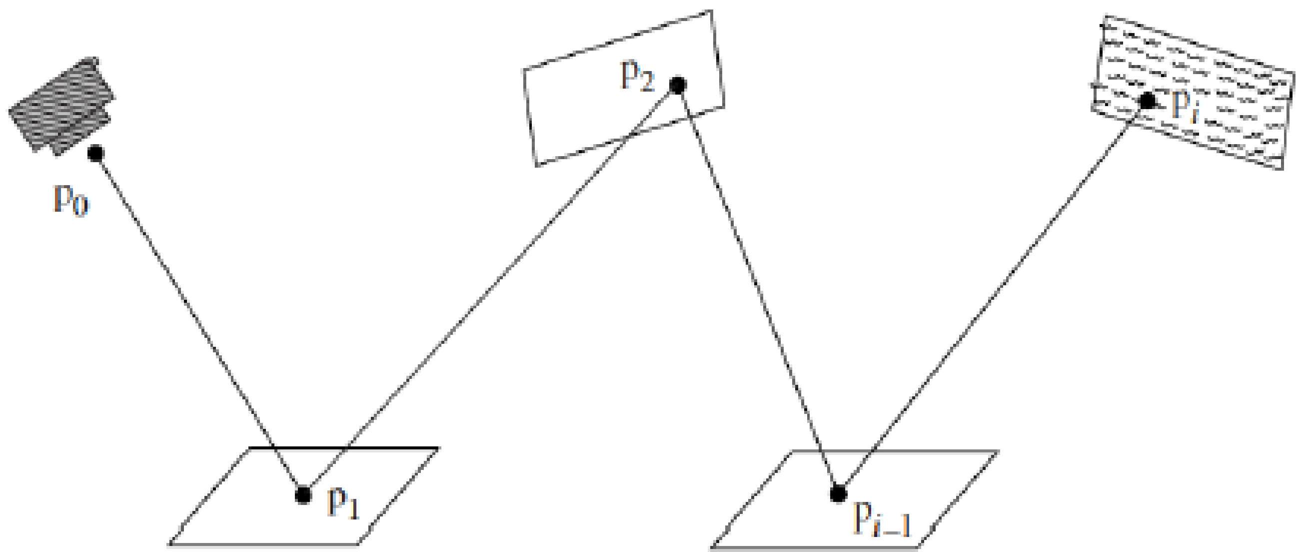
Rappels sur le lancer de rayon

- Le lancer de rayon ne peut pas calculer l’**illumination indirecte**, c’est-à-dire la contribution des surfaces diffuses dans l’éclairage d’un point.
- À la base, le lancer de rayon n’est donc **pas un algorithme d’illumination globale**, puisqu’il laisse de côté des sources d’éclairage. Le **tracé de chemins** permet d’incorporer des calculs d’illumination globale au lancer de rayon.

Tracé de chemins: Path tracing

- Le tracé de chemins est **une extension du lancer de rayon** qui rend possible le calcul d’une illumination globale complète.
- Pour évaluer l’éclairage complet en un point P, le tracé de chemins propose de:
  - Générer des chemins de transport de la lumière (une chaîne de rayons) à partir de points de surface visibles aux sources de lumière.
  - Les rayons sont tracés de manière récursive jusqu'à ce qu'il s atteignent une source de lumière.
  - Evaluation de façon récursive de l'équation de rendu long d'un chemin.
  - Génération des rayons dans un chemin est régi par :
    - des sources lumineuses
    - BRDFs
- **combien doit-on lancer de rayons aléatoires** ?
- Le tracé de chemins propose de faire ça simple : on va lancer **un seul rayon** pour estimer l’illumination indirecte, en plus des rayons dirigés vers les sources de lumière, pour les calculs d’illumination directe.

Fonctionnement du tracé de chemins



Fonctionnement du tracé de chemins

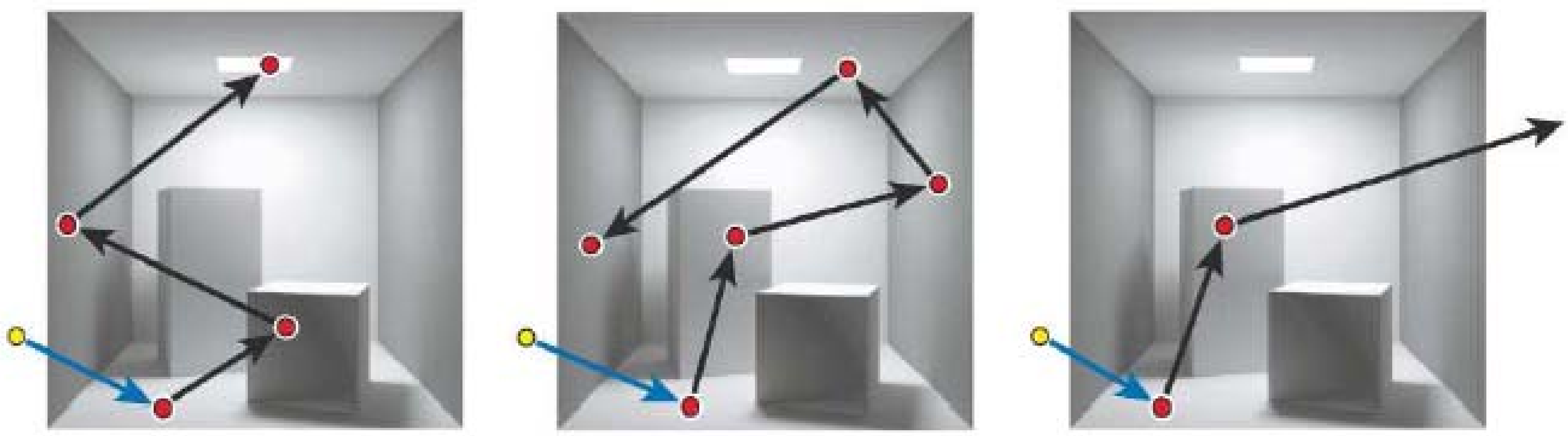
- Pour s’assurer d’avoir une bonne approximation de la véritable illumination, **on lancera un nombre fixe de rayons pour chaque pixel** (qu’on appellera **les échantillons**), et chacun de ses rayons générera un rayon aléatoire.
- La moyenne sera faite sur les intensités associées aux échantillons pour donner la valeur qu’on assignera au pixel.
- Il aurait été possible de ne lancer qu’un seul rayon primaire, pour ensuite **émettre plusieurs rayons aléatoires** à partir du point d’intersection.
- Toutefois, l’algorithme du tracé de chemins étant récursif, la complexité serait devenue exponentielle.

Algorithme

- Pour un pixel p, un rayon est lancé à partir de l'œil à travers p dans la scène.
- au point d’intersection, l'équation de rendu est évaluée en utilisant l’intégration de Monte Carlo.
- Pour approximer la luminance incidente, les rayons sont tracées dans des directions d'échantillons aléatoires.
- Les directions des échantillons sont choisies en fonction de:
  - La pondération des cosinus de la luminance incidente.
  - BRDFs
  - Sources de lumière.
- ce schéma est appliqué de manière récursive tant qu’il y a une quantité importante de luminance transportée long d'un rayon.

Algorithme

- Le suivi des rayons est fait de manière récursive le long d'un chemin afin d’estimer la luminance transportée
- La récursivité s’arrête si :
  - une source de lumière est touché
  - une profondeur maximale / minimum de la luminance est atteint
  - le rayon quitte la scène / frappe le fond



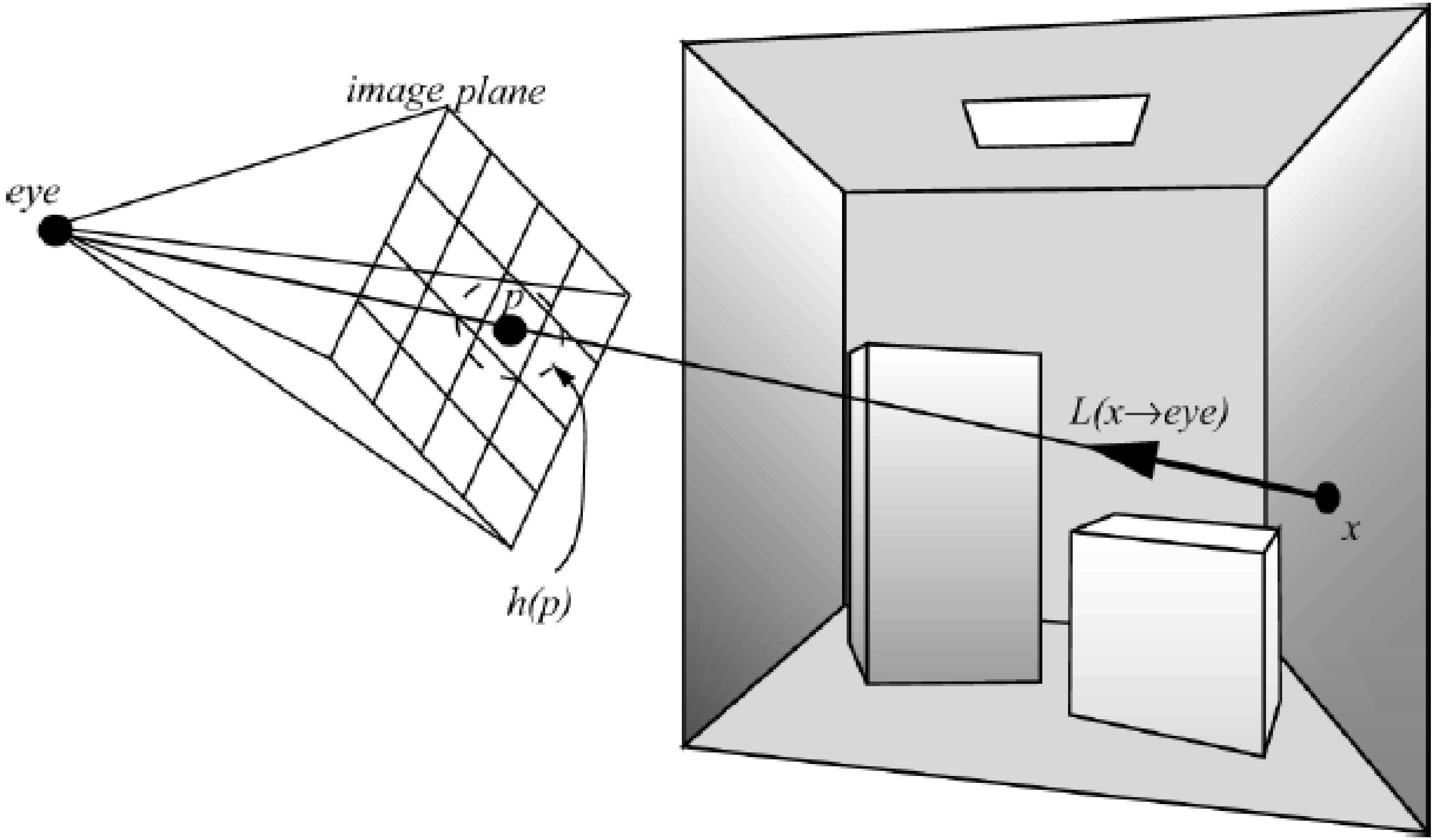
Calcul de la luminance

- Comment évaluer L?
  - L'intégration de Monte Carlo. erreur dans $O(N^{-0.5})$
  - Générer des directions aléatoires ω_i sur l'hémisphère Ω, en utilisant la fonction densité de probabilité $p(\omega)$.

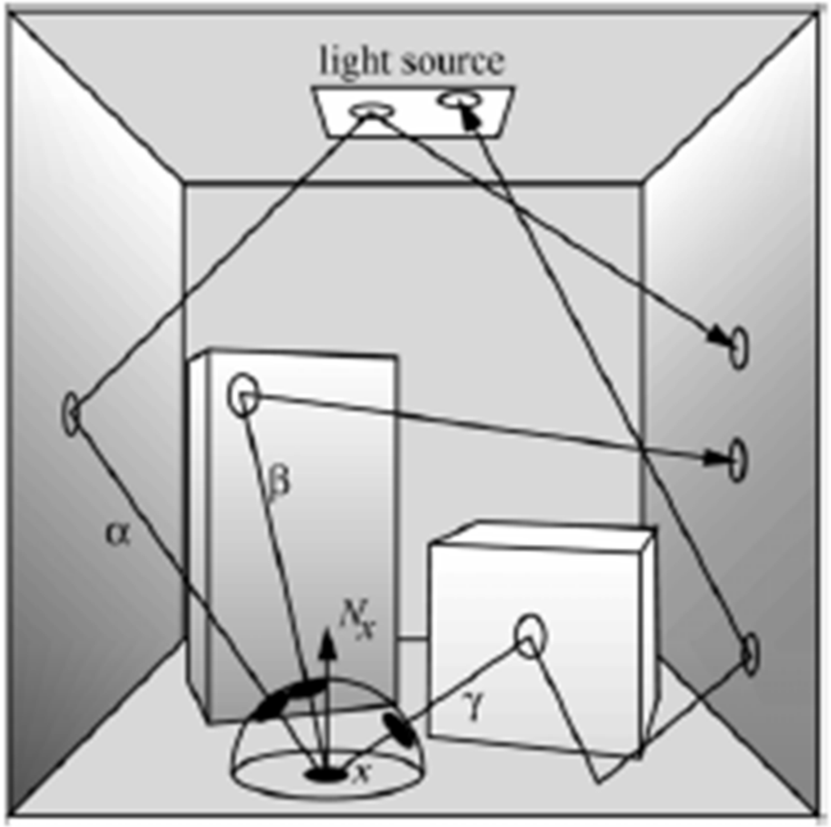
$$L(x,\vec{\omega}') = \int_{\Omega} f_r(x,\vec{\omega},\vec{\omega}') \cdot L_i(x,\vec{\omega}) \cdot (\vec{\omega} \cdot \vec{n}) d\vec{\omega}$$

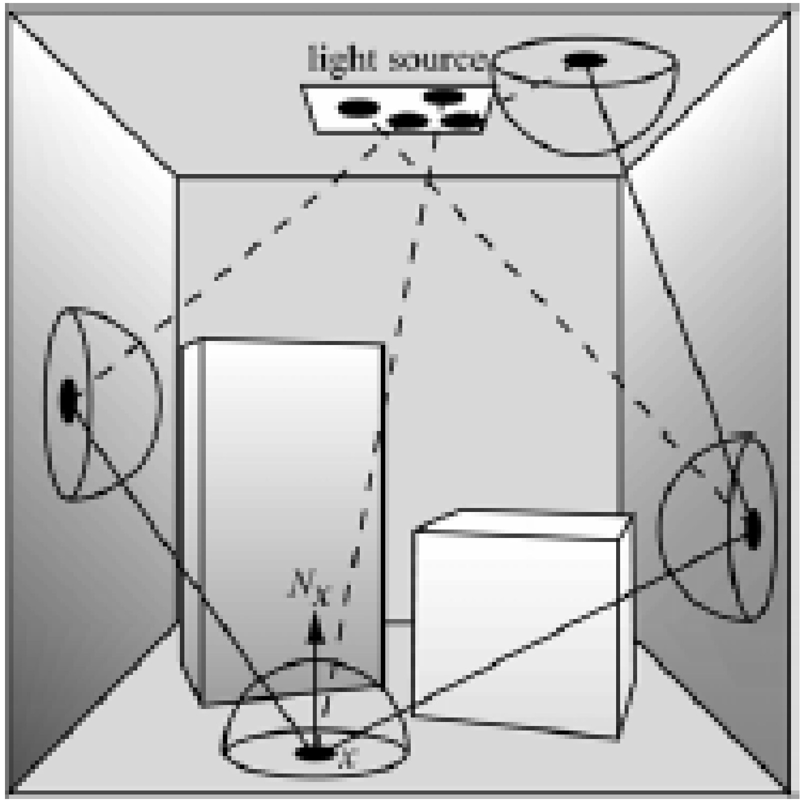
$$\left\langle L(x,\vec{\omega}') \right\rangle = \frac{1}{N} \sum_{i=1}^{N} \frac{f_r(x,\vec{\omega}_i,\vec{\omega}') \cdot L_i(x,\vec{\omega}_i) \cdot (\vec{\omega}_i \cdot \vec{n})}{p(\vec{\omega}_i)}$$

Echantillonnage: Résolution de l'équation intégrale



Echantillonnage: Résolution de l’équation intégrale



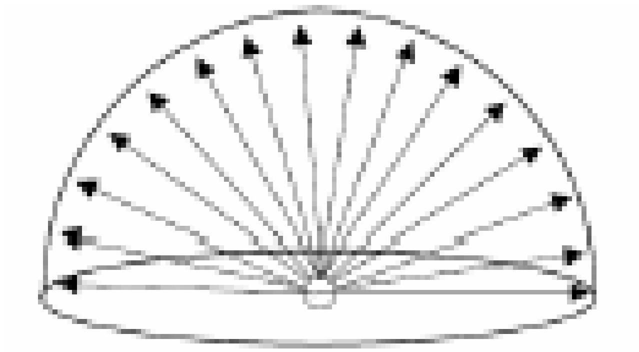


Tracé de rayon par Monte carlo

- Distribution uniforme:

$$p(\Psi) = \frac{1}{2\pi}$$

$$\left\langle L(x,\vec{\omega}')\right\rangle = \frac{2\pi}{N}\sum_{i=1}^{N} f_r(x,\vec{\omega}_i,\vec{\omega}')\cdot L(x,\vec{\omega}_i)\cdot(\vec{\omega}_i\cdot\vec{n})$$

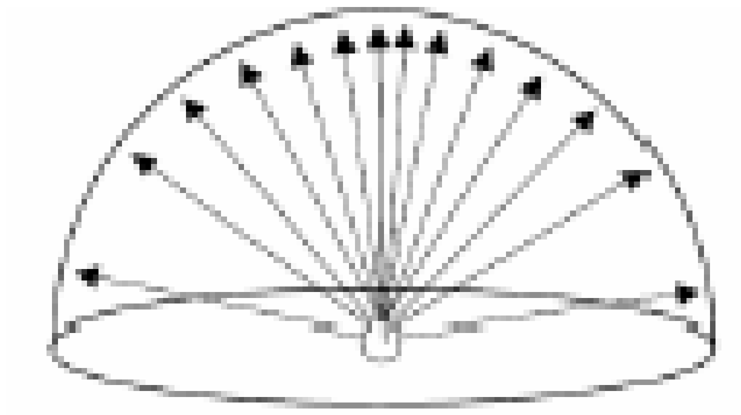


Tracé de rayon par Monte carlo

- Distribution cosinus

$$p(\vec{\omega}) = \frac{(\vec{\omega} \cdot \vec{n})}{\pi}$$

$$\left\langle L(x, \vec{\omega}') \right\rangle = \frac{2\pi}{N} \sum_{i=1}^{N} f_r(x, \vec{\omega}_i, \vec{\omega}') \cdot L(x, \vec{\omega}_i)$$

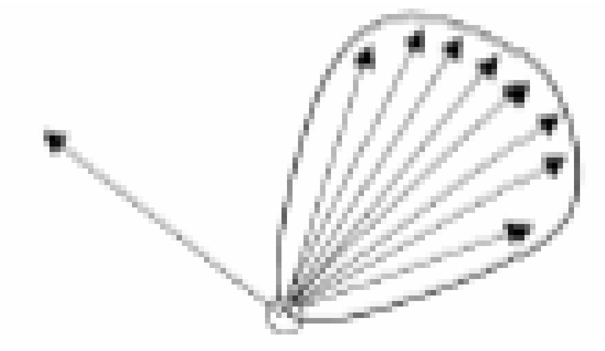


Tracé de rayon par Monte carlo

- Distribution des BRDF

$$p(\vec{\omega}) = f_r(\ldots)$$

$$\left\langle L(x,\vec{\omega}')\right\rangle = \frac{1}{N}\sum_{i=1}^{N} L(x,\vec{\omega}_i)\cdot(\vec{\omega}_i\cdot\vec{n})$$



Tracé de rayon par Monte carlo

- Distribution des cosinus BRDF:

$$p(\vec{\omega}) = f_r(\ldots).(\vec{\omega}_i \cdot \vec{n})$$

$$\langle L(x, \vec{\omega}') \rangle = \frac{1}{N} \sum_{i=1}^{N} L(x, \vec{\omega}_i)$$