M2-Images

notions d'intégration numérique

J.C. Iehl

November 30, 2016

résumé des épisodes précédents

simulation :

- tout les objets éclairés recoivent de la lumière,
- cette lumière est réfléchie dans la scène ...
- et éclaire les objets visibles (par les objets éclairés)...
- sommer tout ça pour chaque point visible de la camera.

l'équation qui fait peur

$$L_r(p, \vec{\omega}_o) = L_e(p, \vec{\omega}_o) + \int_{\Omega^+} L_i(p, \vec{\omega}) f_r(p, \vec{\omega} \rightarrow \vec{\omega}_o) |\cos\theta| d\omega$$

avec :

- $L_r(p, \vec{\omega}_o)$: énergie réfléchie par p dans la direction $\vec{\omega}_o$,
- $L_e(p, \vec{\omega}_o)$: énergie émise par p dans la direction $\vec{\omega}_o$,
- Ω^+ : ensemble de directions autour du point p,
- $L_i(p, \vec{\omega}) = L_r(q, -\vec{\omega})$: énergie incidente en p dans la direction $\vec{\omega}$ = énergie réfléchie par q vers p,
- $f_r(p, \vec{\omega} \rightarrow \vec{\omega}_r)$: matière de p,
- θ : angle entre la normale en p et $\vec{\omega}$.

en clair :

en clair :

- pour chaque point p visible par la camera,
- trouver toutes les sources visibles,
  $L_i(p, \vec{\omega}) > 0 \equiv L_e(s, \vec{-\omega}) > 0$,
  avec s un point sur une source,
  visible par p dans la direction $\vec{\omega}$, $V(p, s) \equiv 1$
- puis, trouver tout les objets visibles,
  et pour chaque point sur un objet visible, trouver quelles sources de lumière sont visibles.

$$L_r(p, \vec{\omega}_o) = L_e(p, \vec{\omega}_o) + \int_{\Omega^+} V(p, s) L_e(s, -\vec{\omega}) f_r(p, \vec{\omega} \to \vec{\omega}_o) |\cos\theta| d\omega$$

qu'est ce qu'on calcule ?

calculer :

- tester un ensemble de directions, $\vec{\omega} \in \Omega^+$,
- si une source de lumière est visible dans cette direction $L_e(s, -\vec{\omega}) > 0$,
- calculer la lumière réfléchie vers la camera, $L_e(s, -\vec{\omega}) f_r(p, \vec{\omega} \to \vec{\omega}_o) |\cos\theta|$,
- sommer pour toutes les directions.

choisir des directions...

un ensemble de directions $\vec{\omega} \in \Omega^+$:

- autour d'un point sur un objet,
- ??

idée :

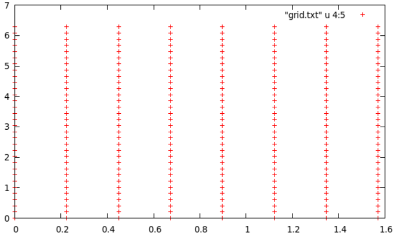
- rappel: intégration avec "la méthode des rectangles",
- fonctionne pour des fonctions 1d,
- mais les directions sont définies en 2d...

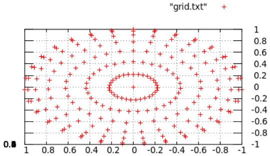
choisir des directions...

construire une grille 2d :

- coordonnées polaires $\vec{\omega} \equiv (\theta, \phi)$

grille en (θ, ϕ)

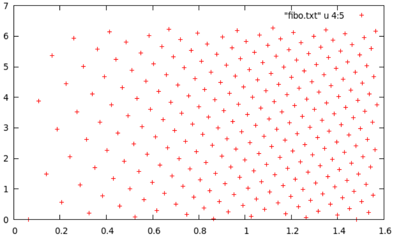


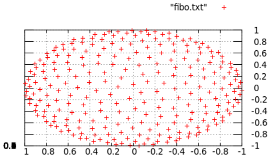


est ce que les cellules sont de même taille ?

est ce que c'est une grille ?

spirale de Fibonacci en (θ, ϕ)





est ce que les cellules sont de même taille ?

est ce que c'est une grille ?

spirale de Fibonacci

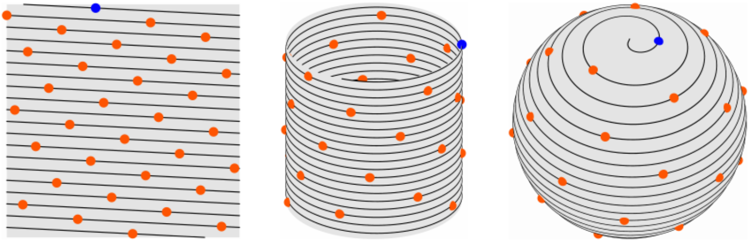


Figure 1: *Illustration of the construction of the point set* $\mathbf{SF}_i^{32}$*, showing the mapping from the grid through a cylinder to the sphere. The first point of the point set is marked blue.*

"Spherical Fibonacci Mapping"

spirale de Fibonacci

construction :

- pour $i \in N$ points
- $\cos\theta = 1 - \frac{2i+1}{2N}$
- $\phi = 2\pi \left[\frac{i}{\Phi}\right] \equiv 2\pi \left(\frac{i}{\Phi} - \left\lfloor \frac{i}{\Phi} \right\rfloor\right)$
- $(x, y, z) = (\cos\phi \sin\theta, \sin\phi \sin\theta, \cos\theta)$
- avec $\Phi = \frac{(\sqrt{5}+1)}{2}$
- avec $\sin\theta = \sqrt{1 - \cos^2\theta}$

"Spherical Fibonacci Mapping"
B. Keinert, M. Innmann, M. Sanger
"Spherical Fibonacci Point Sets for Illumination Integrals"
R. Marques, C. Bouville, M. Ribardière, L.P. Santos, K. Bouatouch

retour au calcul...

rappel :

tester un ensemble de directions, $\vec{\omega} \in \Omega^+$...

dans quel repère connait les directions $\vec{\omega}$?
dans quel repère faut-il connaitre la direction du rayon ?

changement de repère...

$\vec{\omega}$ dans le repère de la scène :

- on connait les coordonnées de $\vec{\omega}$ dans un repère local aligné sur la normale du point p,
- comment calculer les coordonnées de $\vec{\omega}$ dans le repère de la scène ?

idée :

- on connait la normale n au point p,
- construire un 2ième vecteur en "modifiant" les coordonnées de n ?

changement de repère...

idée :

- remplacer la plus petite composante (en valeur absolue) par 1,
- 2 produits vectoriels permettent enfin de construire 2 autres vecteurs orthogonaux...

plus rapide / plus simple :

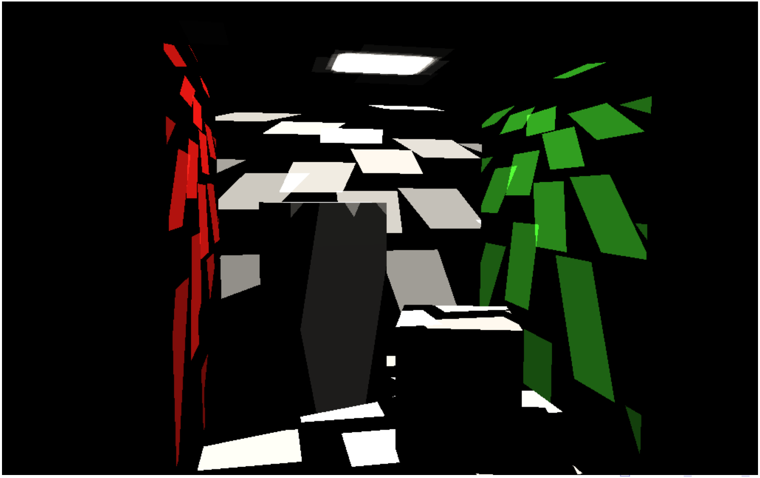
"Building an Orthonormal Basis from a 3D Unit Vector Without Normalization"
J.R. Frisvald

retour au calcul...

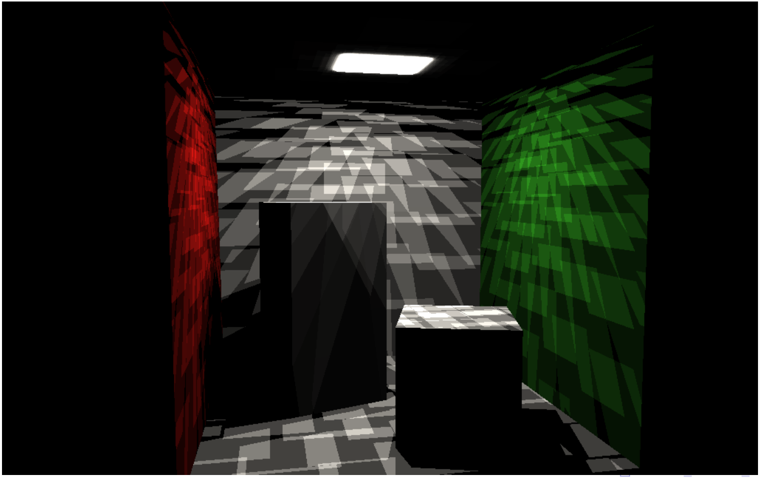
rappel :

- tester un ensemble de directions, $\vec{\omega} \in \Omega^+$...
- sommer la lumière émise par les sources visibles dans les N directions...

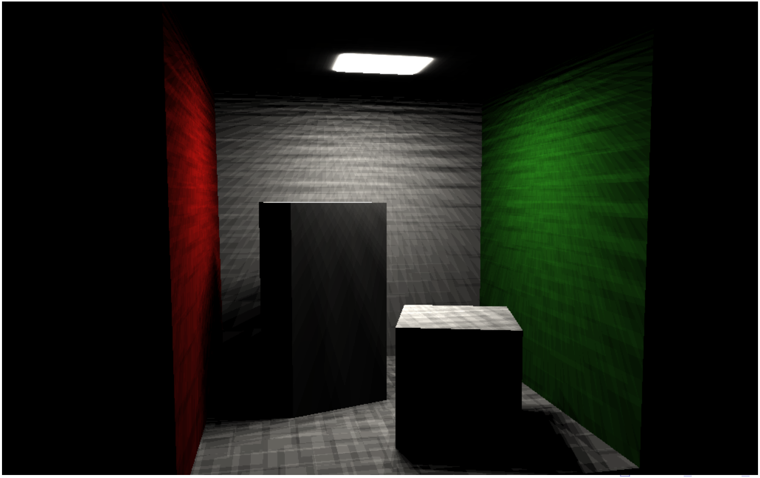
exemple : 64 directions / Fibonacci



exemple : 256 directions / Fibonacci



exemple : 1024 directions / Fibonacci

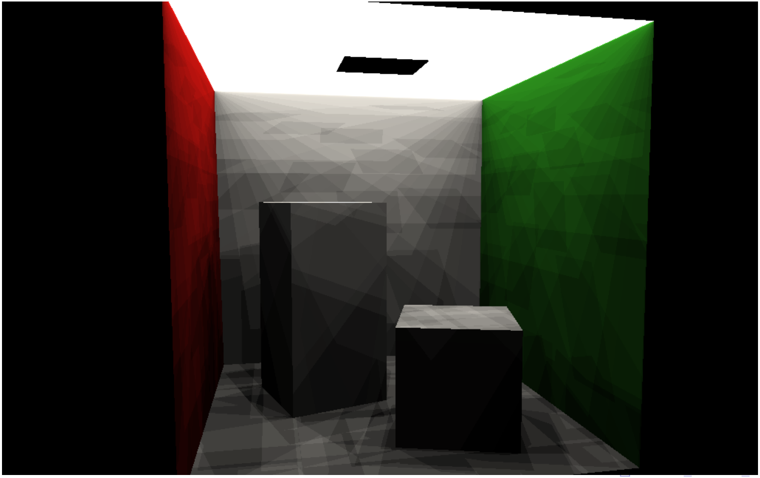


ça marche, mais...

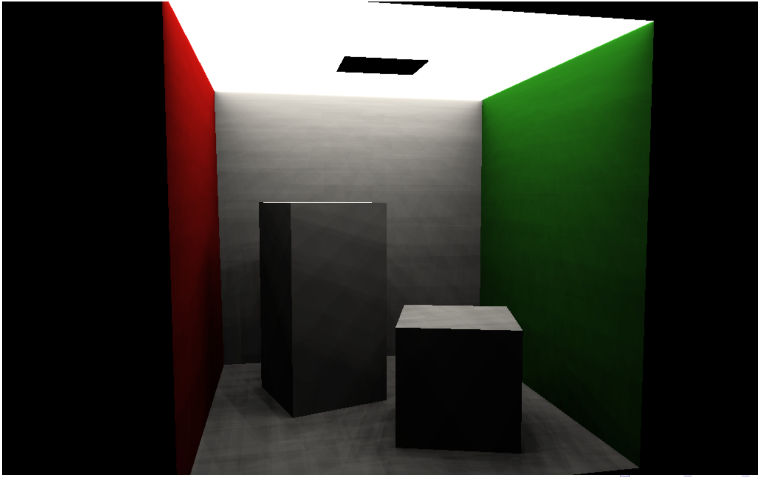
et alors ?

- si la source est "petite", il faut beaucoup de directions,
- le temps de calcul explose...

exemple : 64 directions / Fibonacci



exemple : 256 directions / Fibonacci



ça marche, mais...

idée :

- pas assez de précision pour un nombre raisonnable de directions...
- ??

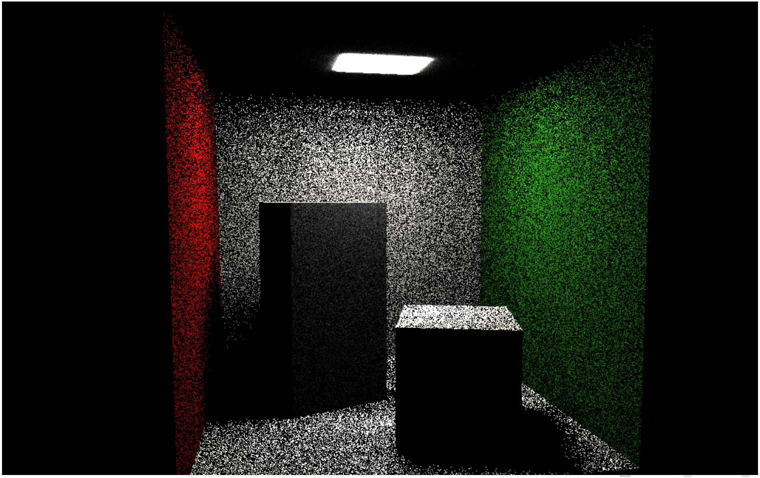
perturber l'orientation de la spirale ?

perturber la spirale de Fibonacci...

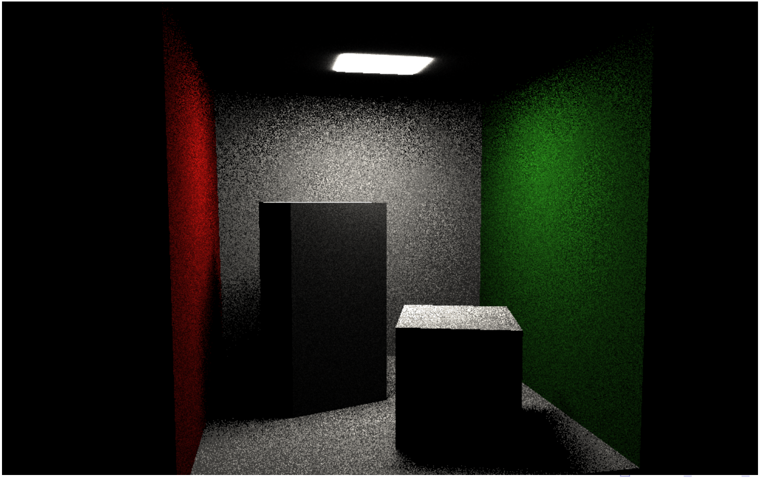
- rappel : $\phi = 2\pi \left[\frac{i}{\Phi}\right]$
- $\phi = 2\pi \left[\frac{i+u}{\Phi}\right]$
- avec u valeur aléatoire $\in [0..1]$, constante pour une spirale,
- plus difficile de perturber θ, pourquoi ?

chaque point p / pixel utilise des directions $\vec{\omega}$ différentes...

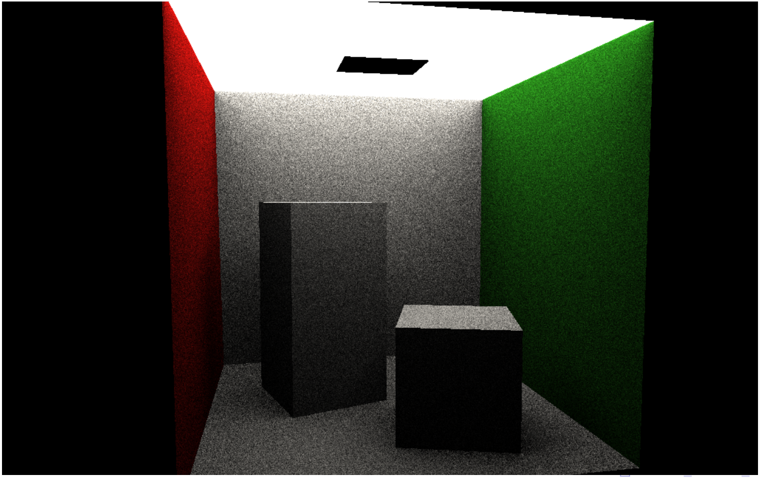
exemple : 64 directions / Fibonacci + perturbation ϕ



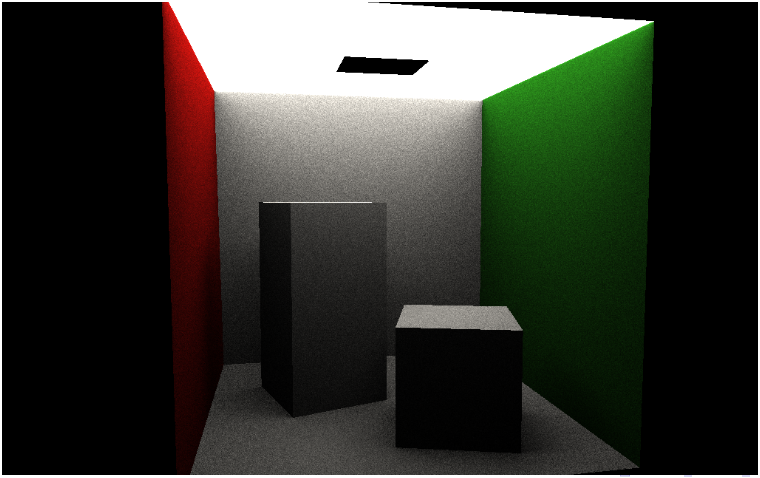
exemple : 256 directions / Fibonacci + perturbation ϕ



exemple : 64 directions / Fibonacci + perturbation ϕ



exemple : 256 directions / Fibonacci + perturbation ϕ



rappel :

rappel :

- pour chaque point p visible par la camera,
- trouver toutes les sources visibles,
  $L_i(p, \vec{\omega}) > 0 \equiv L_e(s, -\vec{\omega}) > 0$,
  avec s un point sur une source,
  visible par p dans la direction $\vec{\omega}$, $V(p, s) \equiv 1$
- puis, trouver tout les objets visibles,
  et pour chaque point sur un objet visible, trouver quelles sources de lumière sont visibles.

et alors ?

combien de rayons par pixel ?

- pour chaque pixel, N directions, pour chaque point trouvé, encore N directions...
- soit N^2 rayons par pixel,
- pour $\approx 1M$ pixel...
- et $N \approx 1000$,
- et pour plus de rebonds ? N^3, N^4, etc.

pas très raisonnable...

contrôler le nombre de rayons par pixel

bilan :

- pour chaque pixel,
- suivre un arbre *complet* de rayons (avec N^2 feuilles),
- et si on ne suivait que "quelques" chemins au lieu de N^2 existants dans l'arbre complet ?

comment ?

contrôler le nombre de rayons par pixel

comment :

- choisir aléatoirement les $P \ll N^2$ chemins à suivre,
- et utiliser une spirale avec beaucoup de directions ($N = 32768$, par exemple), pour obtenir suffisament de précision.

remarque : $N = \infty$, c'est encore mieux...

choisir aléatoirement P directions

choisir P directions :

- intuition : pour P directions combien sont choisies sur la ligne $\theta = \pi/2$?
  et sur la ligne $\theta = \pi/4$?
- intuition : combien de colonnes pour chaque ligne ?

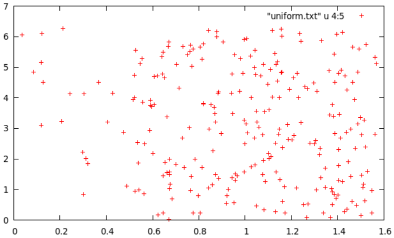
on peut faire la même construction, choisir une direction parmi les 2π qui existent sur une hémisphère.

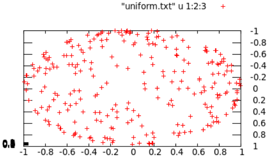
choisir aléatoirement P directions

recueil de formules :

- "Global Illumination Compendium"
  P. Dutre
- par exemple, eq 34 et 35
- $\cos\theta = u_1$
- $\phi = 2\pi u_2$
- $(x, y, z) = (\cos\phi \sin\theta, \sin\phi \sin\theta, \cos\theta)$
- avec $\sin\theta = \sqrt{1 - \cos^2\theta}$,
- avec u_1, u_2 valeurs aléatoires $\in [0..1]$

directions uniformes





et alors ?

et alors ?

- on peut "maitriser" le temps de calcul en controlant le nombre de chemins suivis par pixel,
- mais la qualité est toujours assez mauvaise...

il serait peut être plus efficace de viser exactement les sources, au lieu de les chercher au hasard ?

éclairage direct efficace

éclairage direct efficace :

- comment "viser" une source de lumière ?
- trouver les directions pour lesquelles la source est visible depuis un point p...
- ??
- et si on choisit directement un ensemble de points à la surface de la source de lumière ?
- il ne reste plus qu'à vérifier que la lumière arrive au point p...

choisir un point sur une source de lumière ?

??

- il faut trouver des points qui auraient pu être testés par les directions de la spirale de Fibonacci...
- au lieu de sommer sur un ensemble de directions, il faut sommer sur l'ensemble des points de la source, vus depuis le point p...
- quand on change de direction sur la spirale, comment "bouge" le point sur la source ?

choisir un point sur une source de lumière ?

changement de variable :

- au lieu d'intégrer sur les directions autour de p, qui correspondent à des points sur la source,
- intégrer sur les points de la source et vérifier qu'ils sont visibles par p.

$$L_r(p, \vec{\omega}_o) = L_e(p, \vec{\omega}_o) + \int_S L_e(s, -\vec{\omega}) f_r(p, \vec{\omega} \rightarrow \vec{\omega}_o) G(p, s) ds$$

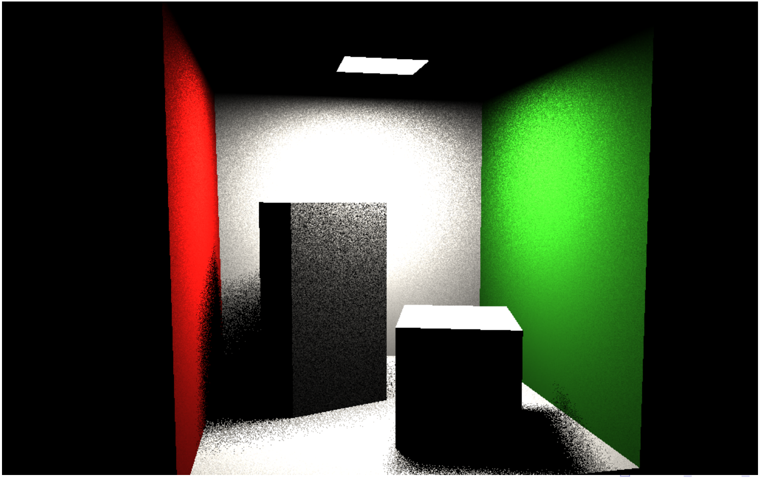
avec $G(p, s) = V(p, s) \frac{|\cos\theta||\cos\theta_s|}{d^2(p,s)}$

choisir un point sur une source de lumière ?

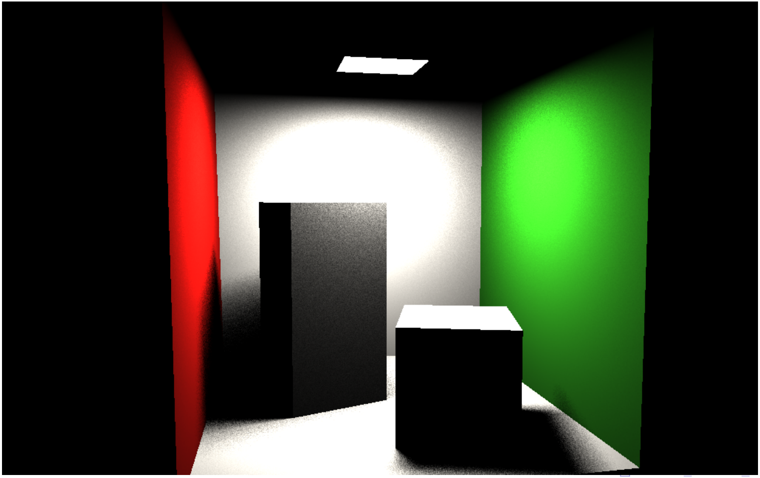
c'est malin, comment on génère un point dans un triangle ?

- on peut utiliser la même idée que la spirale de Fibonacci :
- découper le triangle en lignes et construire les cellules contenues dans le triangle,
- ou on peut aussi faire la construction pour $N = \infty$,
- cf GI Compendium, eq 18
- $\beta = (1 - u_2)\sqrt{u_1}$
- $\gamma = u_2\sqrt{u_1}$
- $p(\beta, \gamma) = (1 - \beta - \gamma)a + \beta b + \gamma c$

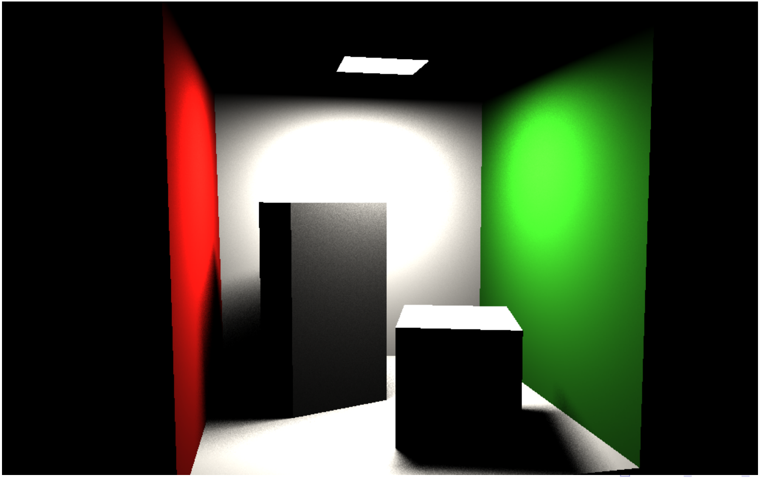
exemple : 1 point par source



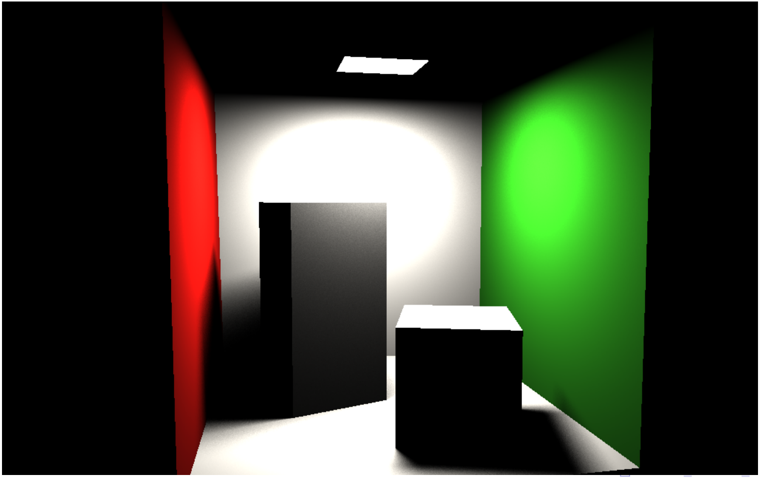
exemple : 16 points par source



exemple : 64 points par source



exemple : 256 points par source



rappel : 1024 directions / Fibonacci