

# Traçado de raios estocástico

Aldo René Zang

5 de julho de 2007

# 1 Introdução

## 1.1 Traçado de raios

O método de Ray tracing é uma das técnicas mais poderosas de rendering, pois além de ser um algoritmo de visualização, pode resolver de forma efetiva e relativamente simples vários fenômenos luminosos como sombras, reflexão e refração.

O método é baseado no princípio da ótica geométrica em que a energia luminosa propaga-se em forma retilínea num meio isotrópico. A energia que chega ao observador numa direção é calculada lançando um raio nesta direção. Como o olho é sensível à radiância, e esta é constante ao longo do raio, a energia luminosa que o raio lançado transporta é caracterizada pela radiância que sai da superfície interceptada pelo raio. O algoritmo de Ray tracing simula o processo de propagação da luz no sentido inverso a como ele ocorre. Isto é feito por motivos de eficiência, já que só os raios que atingem o observador serão lançados, embora esta forma de proceder dificulte a simulação de alguns mecanismos de transporte de energia.

A idéia de utilizar o algoritmo de Ray-tracing para o cálculo da iluminação foi sugerida por Whitted como uma extensão ao algoritmo de visualização de Ray-casting utilizado para determinar as superfícies visíveis da cena. Na geração de imagens por Ray-tracing a intensidade de cada ponto da imagem é calculada lançando-se um raio em direção ao ponto, desde a posição da câmera. Na primeira interseção deste raio com os objetos da cena é calculada a intensidade luminosa utilizando algum modelo de iluminação local. A contribuição luminosa de cada fonte é computada se o ponto do objeto for visível, i.e não estiver na sombra. Isto é determinado lançando-se um raio desde o ponto em questão em direção a cada fonte, e verificando se ele não é interceptado por outros objetos.

Se a superfície for especular o raio refletido é seguido de modo de capturar a radiância vinda de outras superfícies. Se a superfície for translúcida procede-se de forma similar com o raio refratado.

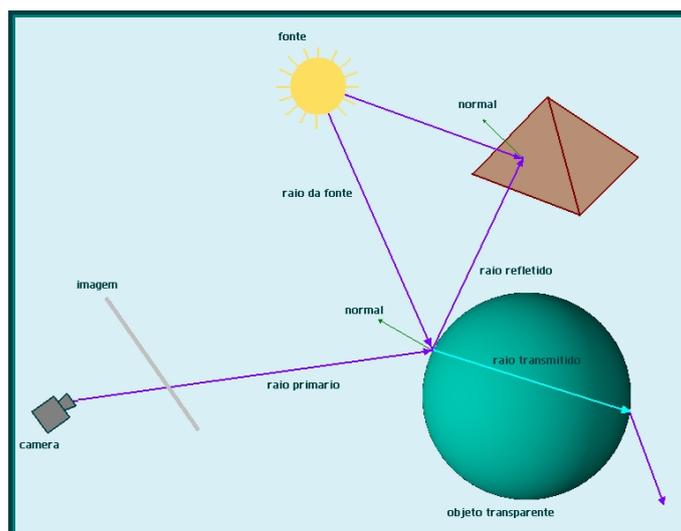


figura 1.1: traçado de raios

## 2 Novas abordagens

### 2.1 Traçado de raio estocástico

O algoritmo tradicional de Ray-tracing, é baseado numa técnica de amostragem pontual. Um único raio é lançado por cada pixel para determinar a superfície visível pela câmera, e a radiância que sai dela. Um único raio também, é lançado desde cada ponto visível da superfície em direção às fontes de luz, para determinar a incidência direta destas e as regiões que estão com sombra. A radiância indireta, vinda de outras superfícies por reflexão ou por transmissão, também é calculada lançando-se um único raio em cada uma destas direções. A técnica de amostragem pontual limita a resolução de alguns fenômenos que envolvem cálculos de integrais para sua representação. O efeito de aliasing, que surge na representação discreta da imagem por pixels e pode ser observado por linhas em forma de escadinha ao invés de traços contínuos, é um exemplo desta limitação. A intensidade de cada pixel é representada pela integral da função de intensidade na área do pixel. Fenômenos luminosos como o espalhamento de luz que ocorre na propagação e na reflexão da luz, não podem ser tratados pela técnica tradicional por envolverem também integrais no seu cálculo.

A técnica de Ray-tracing estocástico, também denominada Ray-tracing distribuído, introduzida por [Cook84], propõe uma solução geral para este tipo de problemas: distribuindo raios em forma estocástica, as integrais são calculadas por métodos de Monte Carlo. [Cook84] aplicou esta técnica para a simulação de vários efeitos luminosos:

#### 2.1.1 Simulação de fontes extensas:

Distribuindo raios nos ângulos sólidos das fontes de iluminação pode ser simulado efeitos de penumbras.

#### 2.1.2 Reflexão especular espalhada:

Distribuindo raios segundo a função de distribuição do fluxo refletido pode ser simulado o espalhamento da reflexão especular em superfícies rugosas, observado como reflexão embaçada.

#### 2.1.3 Refração espalhada:

Distribuindo raios segundo a função de distribuição do fluxo transmitido pode ser simulado o efeito de transmissão, observado como transparência difusa.

#### 2.1.4 Profundidade de Campo:

Considerando um modelo de câmera de lente fina, [Cook84] incluiu, por meio desta técnica, efeitos utilizados em fotografia como a profundidade de campo, o qual pode ser simulado distribuindo raios sobre a superfície da lente.

#### 2.1.5 Borrão de movimento (Motion Blur):

[Cook84] também conseguiu incluir o efeito fotográfico de motion blur, resultante do tempo de exposição do filme fotográfico numa cena não estática, agregando

uma dimensão temporal no cálculo do traçado de raios e distribuindo raios no espaço do tempo.

## Teoria de amostragem

A idéia fundamental da técnica de Ray-tracing estocástico é o tratamento de problemas de integração por métodos de Monte Carlo. De acordo com estes métodos, a integral multidimensional que determina o valor da intensidade do pixel, pode ser aproximada a partir de várias amostras distribuídas em forma aleatória no espaço do pixel.

### 3 Métodos de Monte Carlo

A idéia da integração de Monte Carlo é a de que a solução de uma integral  $I = \int f(x)dx$  pode ser aproximada tomando uma quantidade grande de amostras do integrando

$$I = \int_{\omega} f(x)dx = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N \frac{f(\xi_i)}{p(\xi_i)},$$

Onde as amostras  $\xi_i$  estão distribuídas de acordo com a função de distribuição de probabilidade (pdf)  $p(x)$  (i.e.  $p(x) \geq 0$ ,  $\int p(x) = 1$ ) [KW86]. Além disso, é necessário que a pdf seja não nula nas regiões do domínio  $\Omega$  onde  $f(x)$  é não nula. De acordo com a lei dos grandes números, a probabilidade de que a aproximação seja igual ao valor exato da integral converge para 1 para  $N$  grande

$$P = \left( \left| \frac{1}{N} \sum_{i=1}^N \frac{f(\xi_i)}{p(\xi_i)} - I \right| < \varepsilon \right) \rightarrow 1, \text{ para } \varepsilon > 0, N \rightarrow \infty$$

A técnica de Monte Carlo é particularmente conveniente para resolver equações integrais em dimensões altas com pouca coerência.

#### 3.1 Amostragem por importância e estratificação

A confiabilidade da amostragem Método de Monte Carlo é medida pela variância dos estimadores. A variância dos estimadores é dada por

$$\text{var} \left( \frac{1}{N} \sum_{i=1}^N \frac{f(\xi_i)}{p(\xi_i)} \right) = \frac{\text{var} \left( \frac{f(\xi_i)}{p(\xi_i)} \right)}{N},$$

Onde  $\frac{f(\xi_i)}{p(\xi_i)}$  é o estimador primário, entanto que sua media é o estimador secundário.

Como consequência do acima fórmula, a técnica de Monte Carlo requer um elevado número de amostras par diminuir a variância do estimador principal.

O erro  $\varepsilon = I - \frac{1}{N} \sum_{i=1}^N \frac{f(\xi_i)}{p(\xi_i)}$  da aproximação é também uma variável aleatória com medida, i.e. o estimador é imparcial. O desvio estándar do estimador é

reduzido proporcionalmente a  $1/\sqrt{N}$ , que é conhecido como diminishing return da técnica de Monte Carlo, i.e. o número de amostras deve ser quadruplicado para diminuir o desvio estándar pela metade.

A formulação do método de Monte Carlo é relativamente simples e a mesma é verdadeira para uma implementação direta. Contudo, o método de Monte Carlo é conhecido pela sua ineficiência e o seu alto custo computacional, devido a o elevado número de amostras que ter ser tomado. Este custo pode ser reduzido escolhendo adequadamente as pdf's para o problema específico, a técnica é conhecida como amostragem por importância.

Para amostragem por importância nós ajustamos a pdf  $p$  para ser semelhante à função integral  $f$ . Se formos capazes de escolher  $p(x) = Cf(x)$ , com um fator constante  $C = 1/I$ , a variância do estimador principal seria zero e uma única amostra nos daria sempre o resultado correto. Infelizmente, o constante  $C$  é determinada pela função integral que pretendemos computar sendo portanto indisponível. Por outro lado, se tem conhecimento a priori sobre a forma de  $f$  e pode ser utilizada para ajustar  $p$  a fim de diminuir a variância do estimador.

Outra técnica para reduzir a variância é a *amostragem estratificada*. Aqui, o domínio de integração  $\Omega$  é subdividido num número de domínios menores domínios ou estratos  $\Omega_i$ , e cada estrato é amostrado independentemente. Desta forma, a integral é substituída por um somatório de integrais sobre cada um dos domínios menores, i.e.

$$\text{var} \left( \frac{1}{MN} \sum_{i=1}^{MN} \frac{f(\xi_i)}{p(\xi_i)} \right) = \frac{1}{N} \sum_{k=1}^N \text{var} \left( \frac{1}{M} \sum_{i=1}^M \frac{f(\xi_{ik})}{p(\xi_{ik})} \right)$$

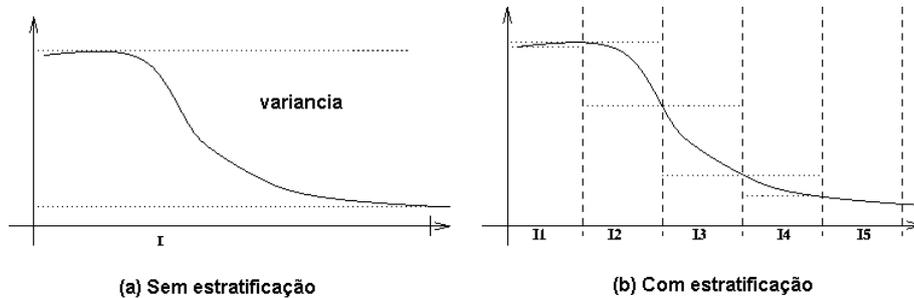


figura 3.1: A variância da amostragem sobre um intervalo  $I$  sem estratificação (a), é maior do que estratificando o intervalo  $I$  em intervalos  $I_i$  (b)

Na maioria dos casos a variância em cada domínio  $\Omega_i$  é muito menor do que sobre o domínio completo, reduzindo assim a variância global (comparar para Figura 3.2). Infelizmente, o sucesso da estratificação depende muitíssimo do integrando  $f$  (e.g. a variância de ruído branco é constante para qualquer domínio), mas frequentemente é melhor que amostragem por importância. Amostragem por estratificação e importância podem ser facilmente combinados, resultando numa função de importância diferente para ser utilizada em cada estrato. Exemplos de amostragem estocástica são mostradas na figura 3.2.

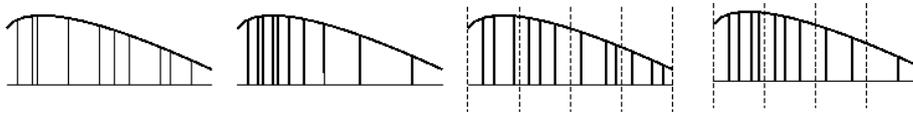


figura 3.2: Exemplos de técnicas de amostragem estocástica: amostragem aleatória simples (esquerda) , amostragem por importância (centro esquerda), combinação de amostragem por importância e estratificação.

### 3.2 Amostragem multidimensional

Se a integração multidimensional é realizável, que é habitualmente a o caso do rendering, a técnica de estratificação pode ser estendida para cada dimensão, uma de cada vez. Numa aplicação ingênua esta explode pelo fato do número de amostras ser levado. Amostragem completa de todos os estratos em  $D$  dimensões requer  $N^D$  amostras se cada dimensão estiver dividida em  $N$  regiões.

Uma alternativa é o método de jittering não correlacionado [Cook86], que é essencialmente a mesma idéia do path-tracing [Kaj86], onde apenas um estrato de cada dimensão é utilizado para cada amostra da integral multidimensional. Devido a que amostras são independente de aquelas tomadas para outras dimensões , esta modificado calcular é imparcial.

Outro método para escolher amostras multidimensionais é a amostragem N-rook [Shi90a], onde somente  $N$  das  $N^D$  combinações possíveis de domínios são amostradas. Se  $\pi_1, \dots, \pi_D$  é uma permutação diferente da seqüência , então a amostra na dimensão  $j$  é tomada no estrato  $\pi_j(i)$ .

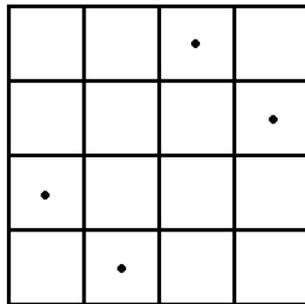


figura 3.2.1: Uma configuração para o método N-Rook para  $N = 4$  e  $D = 2$ .

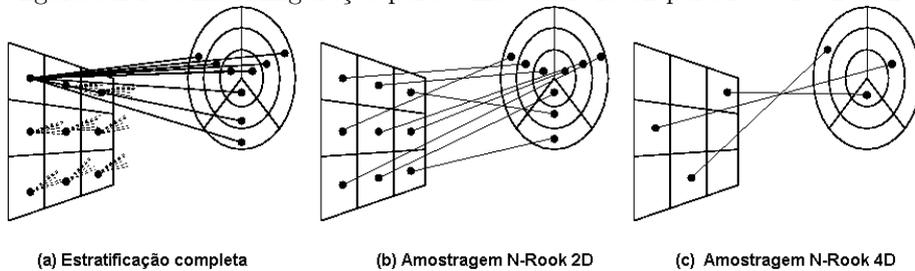


figura 3.2.2: Técnicas de amostragem estocástica multidimensional: (a) Estratificação completa gerando 81 amostras, (b) Amostragem N-Rook com 9

estratos bidimensionais cada, (c) Amostragem N-Rook com 3 estratos unidimensionais no espaço 4-dimensional.

Sabe-se pouco sobre o que é uma boa ou má estratégia de amostragem, mas correlação linear definitivamente deve ser evitada. Mitchel [Mit91] desenvolveu uma condição para otimização e sugere vários métodos para gerar uma amostragem perto da ótima modelando-a eficientemente.

### 3.3 Geração de amostras com densidade não uniforme

Uma vez que tivermos uma pdf adequada para fazer uma integração de Monte Carlo, o problema restante é gerar amostras de acordo com esta distribuição a partir de amostras uniformemente distribuídas  $\xi_i$  sobre o intervalo unitário.

Se a função de densidade  $p(x)$  é definida sobre o intervalo  $[a, b]$ , temos que a distribuição acumulada  $P(x)$  é dada por

$$P(x) = \int_a^x p(x') dx'$$

Amostras  $\xi'_i$  distribuídas segundo  $p$  podem ser geradas pela transformação

$$\xi'_i = P^{-1}(\xi_i)$$

Para uma função  $p(x, y)$  de densidade bidimensional sobre o intervalo  $[a, b] \times [c, d]$  a função de distribuição é dada por

$$P(x, y) = \int_c^y \int_a^x p(x', y') dx' dy'$$

Amostras apropriadamente distribuídas  $\xi' = (x_i, y_i)$  podem ser geradas em dois passos, primeiro escolhe-se  $x_i$  segundo  $P(x, d)$ , logo escolhe-se  $y_i$  segundo  $P(x_i, y)/P(x_i, d)$ . No caso especial em que  $p$  é separável isto é simplificado utilizando o caso unidimensional em cada dimensão [KW86].

Seguidamente apresentamos a geração de distribuições freqüentemente utilizadas para renderização.

#### 3.3.1 Amostragem uniforme do disco de raio R:

Amostragem uniforme sobre o disco de raio  $R$  da a função de densidade  $p(r, \theta) = 1/(\pi R^2)$  e a distribuição  $P(x, y) = (\theta r^2)/(2\pi R^2)$

A transformação de amostras  $(\xi_1, \xi_2)$  uniformemente distribuídas sobre o quadrado unitário é dada então por  $(r, \theta) = (R\sqrt{\xi_1}, 2\pi\xi_2)$ .

Na seção de Profundidade de campo estudaremos outro tipo de distribuição uniforme sobre o disco unitário, mais conveniente para os fins de nosso trabalho.

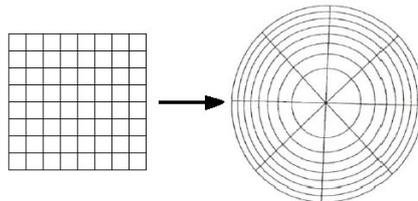


figura 3.3.1: Amostragem uniforme sobre o disco unitário.

### 3.3.2 Transformação para uma potencia do coseno:

A função de densidade é dada por

$$p(\theta, \phi) = \frac{n+1}{2\pi} \cos^n \theta$$

Em coordenadas polares é dada por

$$(\theta, \phi) = \left( \arccos \left( (1 - \xi_1)^{\frac{1}{n+1}} \right), 2\pi\xi_2 \right)$$

## 4 Profundidade de campo

Câmeras reais tem sistemas de lentes que focalizam a luz a traves de uma abertura finita no plano da imagem. Como a abertura tem uma área, um ponto da cena é projetado numa área no plano da imagem chamada círculo de confusão. Correspondentemente uma área finita da cena é visível desde um ponto do plano da imagem, dando uma imagem borrada.

O tamanho do círculo de confusão é afetado pelo raio de abertura da lente e pela distancia entre o objeto e a lente. A distancia focal é a distancia entre a lente e o plano que contem os objetos cujas projeções no plano da imagem tem círculo de confusão com raio igual a zero. Na realidade, os objetos permanecem em foco, em tanto o círculo de confuso seja menor que o tamanho de um pixel.

O tamanho da faixa na qual os objetos permanecem em foco é chamada de profundidade de campo. Câmeras projetivas precisam de 2 parâmetros extras para definir profundidade de campo: o tamanho de abertura da lente e a distancia focal.

A matemática envolvida no cômputo de círculos de confusão para lentes simples não é complicada , e pode ser facilmente modelada no Ray tracer. Somente é preciso escolher um ponto na lente e encontrar o raio do ponto no filme que passa na lente pelo ponto dado tal que os objetos no plano focal continuem em foco. Infelizmente é preciso traçar muitos raios para cada pixel da imagem a fim de obter um resultado apreciável.

O modelo físico estudado para o caso de uma lente delgada é o representado na figura 4.1.a, entanto que a efeitos de simplificação, implementamos o modelo apresentado na figura 4.2. derivando-o do modelo 4.1.b. Como pode ser observado, no modelo físico temos a lente entre o plano do filme e os objetos da cena, enquanto que no modelo projetivo utilizado para descrever cenas 3D computacionalmente costuma-se utilizar um modelo no qual o plano de projeção está entre o ponto de vista (lente) e os objetos da cena a fim de simplificar efeitos físicos tais como inversão da imagem. Com a finalidade de simplificar os cálculos envolvidos, implementamos o modelo 4.2 considerando que vamos tomar amostras bem distribuídas sobre a superfície da lente, e portanto não é preciso se preocupar com fenômenos de inversão.

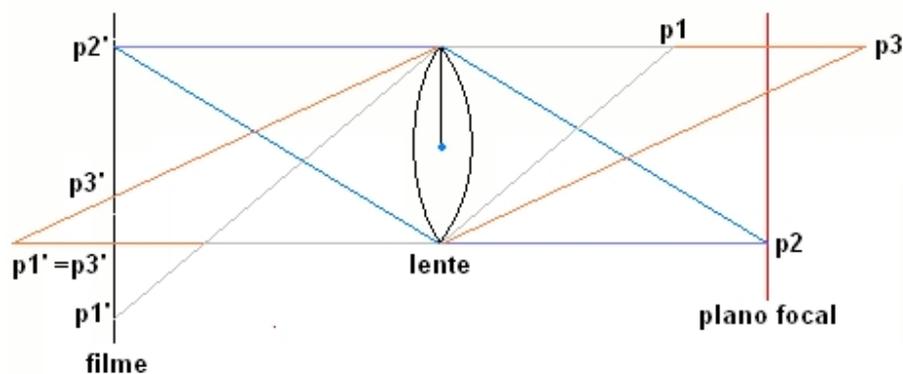


figura 4.1a: Modelo físico de uma lente delgada. Os pontos p1 e p3 não estão no plano focal, assim a região do filme onde p1 e p3 são mapeados aparece borrada. O ponto p2 está no plano focal, assim sua projeção no plano da imagem p2' está

em foco. A medida que aumentamos o raio da lente ou nos afastamos do plano focal, os objetos ficam mais desfocados e portanto mais borrados na imagem.

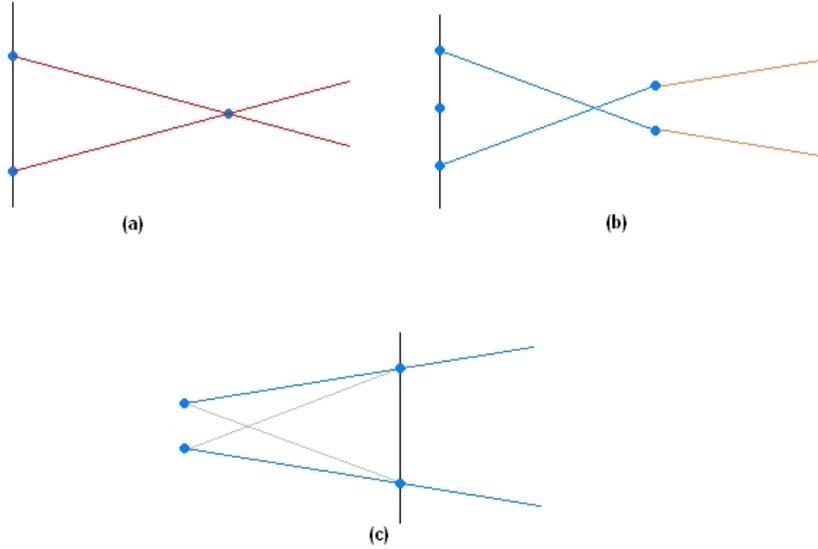


figura 4.1b: (a) Para uma câmara de furo, cada ponto no plano da imagem tem associado com ele um único raio da câmara, que passa pelo furo da camera. (b) Para o modelo de abertura finita, amostramos o ponto no disco da lente e encontramos as direções dos raios calculando a direção do ponto na lente para o ponto no plano focal tal que o raio que passa pelo centro da lente também o intersecta. (c) Simplificação do modelo (b) colocando o plano da imagem entre a lente e a cena.

Considerando os pontos da lente como equiprováveis, utilizaremos amostragem estratificada para amostrar pontos na lente. Estudaremos dois tipos de estratificação do disco unitário tais que os estratos tem áreas iguais:

#### 4.1 Estratificação angular:

Trata-se de uma amostragem uniforme do disco unitário pois divide o mesmo em regiões de igual área (figura 4.1.1). Temos a função de probabilidade considerada para  $r \in [0, 1]$   $\theta \in [0, 2\pi]$   $p(r, \theta) = 1/(\pi R^2)$  e  $P(x, y) = (\theta r^2)/(2\pi R^2)$  A transformação de amostras  $(\xi_1, \xi_2)$  uniformemente distribuídas sobre o quadrado unitario é dada então por  $(r, \theta) = (R\sqrt{\xi_1}, 2\pi\xi_2)$ .

#### 4.2 Estratificação concêntrica:

Trata-se também de uma amostragem uniforme do quadrado unitário  $[0, 1]^2$  no disco unitário. Como pode ser observado na figura 4.2.1 estratificação concêntrica mapeia quadrados em círculos, dividindo o círculo unitário em regiões não só de igual área senão também de formas semelhantes. Em outras palavras, a estratificação concêntrica aplica distorções menores na forma relativa da regiões quando estas passam do quadrado unitário e são mapeadas no círculo.

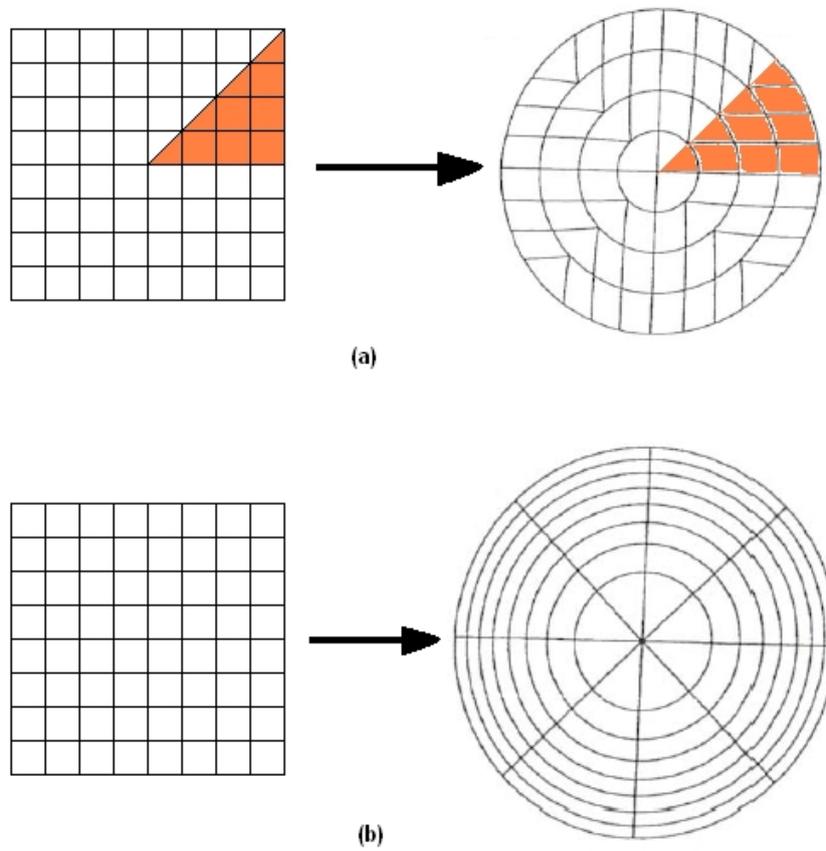


figura 4.4.1: (a) Amostragem uniforme por estratificação angular (b) Amostragem por estratificação concêntrica.

#### 4.2.1 Implementação do ConcentricSampleDisk

---

Point2 ConcentricSampleDisk(Real u1, Real u2)

```
Point2 p;
Real r, theta;
// Map uniform random numbers to  $[-1, 1]^2$ 
Real sx = 2. * u1 - 1.;
Real sy = 2. * u2 - 1.;
// Map square to  $(r, \theta)$ 
// Handle degeneracy at the origin
if (sx == 0.0 AND sy == 0.0)
    p.x = 0.0;
    p.y = 0.0;
    return;
if (sx >= -sy)
    if (sx > sy)
        // Handle first region of disk
        r = sx;
        if (sy > 0.0)
            theta = sy/r;
        else
            theta = (Real)8.0 + sy/r;
    else
        // Handle second region of disk
        r = sy;
        theta = (Real)2.0 - sx/r;
else
    if (sx <= sy)
        // Handle third region of disk
        r = -sx;
        theta = (Real)4.0 - sy/r;
    else
        // Handle fourth region of disk
        r = -sy;
        theta = (Real)6.0 + sx/r;
        theta *= ((Real)PI / (Real)4.);
p.x = r*cos(theta);
p.y = r*sin(theta);
return p;
```

---

## 5 Borrão de movimento (Motion blur)

Distribuir os raios ou os pontos amostrais no tempo resolve o problema de *motion blur*. Numa primeira tentativa, gostaríamos de solucionar o problema amostrando a imagem no tempo inicial e final e logo durante a renderização realizar uma interpolação entre as imagens no tempo inicial e final. Infelizmente isto não soluciona o problema de forma conveniente, pois dois objetos podem estar separado no tempo inicial e no tempo final mas ter sido parte de uma interação mútua algum momento durante a exposição.

Existem varias tentativas de resolver estes problemas, varias delas muito caras e que não resolvem completamente o problema. A solução proposta por Cook é a de distribuir pontos amostrais ao longo do tempo, o que implica agregar mais uma dimensão no tratamento do Ray tracing. Esta solução, embora pareça complicada pois temos que aumentar ainda mais a dimensão do nosso espaço de cálculo, resulta muito conveniente ao se utilizar métodos de Monte Carlo e técnicas de amostragem, como veremos a seguir. O único requisito caro computacionalmente é poder calcular a posição de um objeto num momento dado.

A idéia de distribuir pontos amostrais no tempo não implica em agregar mais raios a serem traçados, mas sim em distribuir os raios existente convenientemente ao longo do tempo. O único cálculo adicional é o de atualizar a posição dos objetos cada vez que um raio é traçado. Resolvemos isto computacionalmente simplesmente atualizando para cada objeto a matriz de transformação leva o objeto seu referencial local para o referencial da câmera.

### 5.1 Movimentos implementados:

**Incluimos os seguintes movimentos:** movimento retilíneo, movimento parabólico, rotação sobre um eixo passando pelo centro de coordenadas do objeto. Todos os movimentos podem ter aceleração.

As posições são dadas pelas seguintes fórmulas

**movimento retilíneo e parabólico:**

$$p(t) = v.t + \frac{1}{2}a.t^2 \text{ onde } p = (x_t, y_t, z_t), v = (v_x, v_y, v_z) \text{ e } a = (a_x, a_y, a_z)$$

**movimento de rotação:**

$$\alpha(t) = v.t + \frac{1}{2}a.t^2 \text{ onde } \alpha \text{ é o ângulo de rotação realizado no instante } t \text{ em torno do eixo de rotação, } v \text{ é a velocidade angular e } a \text{ a aceleração angular.}$$

## 6 Penumbra

Penumbra ocorre quando a fonte de luz é parcialmente obstruída. Para que isto aconteça precisamos de fontes de luz extensas, i.e que tenham uma área ou volume definido. A intensidade refletida é proporcional ao ângulo sólido da porção visível da fonte de luz. Falamos aqui em ângulo sólido, mas na prática o cálculo da porção do ângulo visível é muito complexa computacionalmente,

portanto resolvemos o problema tomando amostras aleatórias na fonte de luz e traçando-se raios. Segundo a proposta de [Cook84], as sombras podem ser calculadas traçando raios desde o ponto da superfície para a fonte de luz e as penumbras distribuindo estes raios secundários. O raio de sombra pode ser traçado para qualquer ponto na fonte de luz, somente não numa única localização. A distribuição dos raios de sombra deve ter peso de acordo à área projetada e ao brilho das diferentes partes da fonte luminosa. O número de raios traçados a cada região deve ser proporcional à quantidade de energia luminosa que chega daquela região se a luz estivesse completamente visível. A proporção de pontos amostrais numa região da superfície é então igual à proporção da intensidade da luz que é visível nessa região.

## 6.1 Luzes de área

A simulação de fontes extensas de iluminação gera o efeito de penumbra quando a luz for parcialmente bloqueada por um objeto. A radiância refletida num ponto de uma superfície, devida a uma fonte, pode ser calculada a partir de uma equação de Fredholm de segundo tipo para  $\omega = \Omega$  como:

$$L_r = \int_{\omega} \rho L_i \cos\theta_i d\omega$$

Onde  $\omega$  é o ângulo sólido formado pela fonte na direção do ponto.

A fim de simplificar as implementações, vamos considerar dois tipos de fontes luminosas de área: fontes com forma de paralelogramo plano e fontes com forma de discos planares.

**Fonte paralelogramo:** é facilmente definida com três parâmetros vetoriais *loc* é a posição de um vértice da fonte, *p1* e *p2* são os vetores que definem o paralelogramos com vértice em *loc* e lados *p1* e *p2*. Como pode-se observar é quase imediato realizar uma estratificação uniforme no paralelogramo a partir do quadrado unitário, como pode ser observado na figura. Observamos ainda que esta estratificação não é a mais conveniente no caso que os lados do paralelogramos sejam muito diferentes no seu comprimento e quando um dos ângulos do paralelogramo é muito pequeno.

**Fonte circular:** Será definida pelo centro do círculo com o parâmetro *loc* e dois vetores *e1* e *e2* que serão ortonormalizados para definir uma base sobre o círculo. Finalmente temos um parâmetro escalar *radio* que define o raio do disco luminoso. Para amostrar a fonte circular, utilizaremos a amostragem concêntrica definida anteriormente na seção 4.1.

## 7 Reflexão difusa

As superfícies especulares rugosas têm a sua componente especular distribuída em torno da direção de reflexão, e não inteiramente concentrada nesta direção, como em superfícies especulares perfeitas. O efeito de espalhamento na reflexão, ocorre devido a esta distribuição da energia luminosa refletida podendo ser observado como reflexões embaçadas que se atenuam com o aumento de distância entre os objetos.

Considerando apenas a componente especular da reflexão na equação integral de Fredholm do segundo tipo, a radiância refletida num ponto da superfície é:

$$L_r = s \int_{\Omega} \rho L_i \cos\theta_i d\omega$$

Onde  $\Omega$  é o hemisfério que contém o vetor normal à superfície. Esta integral somente é significativa num pequeno ângulo sólido em torno da direção de reflexão, devido à acentuada distribuição direcional da função de refletância bidirecional  $\rho_s$

Existem vários modelos que descrevem a reflexão, entre eles podemos citar : Modelo de Torrance e Sparrow, de base física, e bastante utilizada na síntese de imagens foto realistas; e o modelo de Phong de base empírica. Pela simplicidade de cálculo, escolhemos o modelo de Phong, que pode ser normalizada para ser utilizada como função de densidade de probabilidade,

$$p(\theta, \phi) = \frac{n+1}{2\pi} \cos^n \theta$$

Onde  $n$  é o expoente de Phong, que tem o importante papel de caracterizar o grau de espalhamento da reflexão. Quanto menor  $n$ , maior será o espalhamento da componente especular, ou seja a superfície será mais rugosa.

Da equação anterior obtemos a transformação

$$(\theta, \phi) = \left( \arccos \left( (1 - \xi_1)^{\frac{1}{n+1}} \right), 2\pi\xi_2 \right)$$

de pontos  $(\xi_1, \xi_2)$  uniformemente distribuídos sobre  $[0, 1]^2$  sobre a região de espalhamento da reflexão.

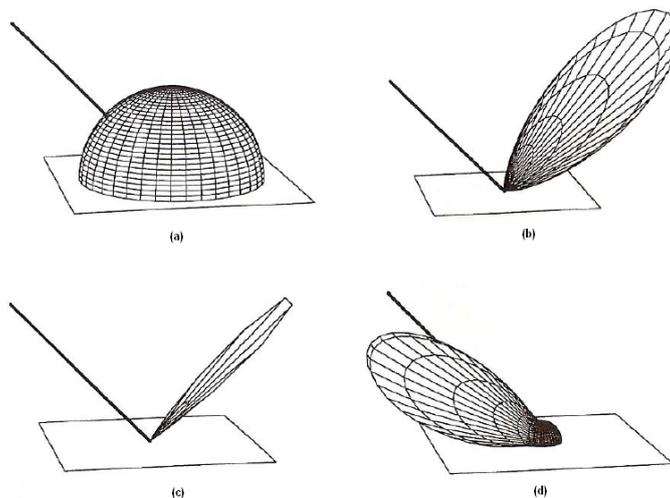


figura 7.1: (a) Reflexão difusa, modelada pelo modelo de Lambert (b) Reflexão especular rugosa, (c) reflexão especular perfeita, (d) Distribuição especular retro-reflexiva.

Devemos notar também que existem dois tipos de função de distribuição de refletância: isotrópicas e anisotrópicas. O modelo isotrópico é aplicado em materiais que emitem a mesma reflexão em qualquer direção em torno e um ponto para um ângulo de incidência dado. O modelo anisotrópico considera materiais que refletem mais luz em certas direções, como por exemplo os cd's, e metais polidos.

## 8 Refração

A refração, também conhecida como fenômeno de transmissão, acontece quando um objeto deixa passar uma certa porção da luz nele incidente. A figura 8.1 mostra o fenômeno de transmissão, que obedece à lei de Snell, segundo a qual  $n_i \text{sen} \theta_i = n_t \text{sen} \theta_t$ , onde  $n_i$  e  $n_t$  são os índices de refração dos meios delimitados pela superfície: e  $\theta_i$  e  $\theta_t$  são respectivamente os ângulos que o raio incidente  $\omega_i$  e o raio transmitido  $\omega_t$  formam com a normal  $n$  da superfície. A efeitos de simplificar o modelo de transmissão utilizaremos uma função de distribuição semelhante à utilizada para reflexão, sendo

$$p(\theta, \phi) = \frac{m+1}{2\pi} \cos^m \theta$$

Onde  $m$  é um expoente grande o suficiente para caracterizar adequadamente o espalhamento da transmissão. Ao igual que na reflexão quanto menor  $m$ , maior será o espalhamento da componente transmitida, ou seja a superfície será mais dispersiva e em certo modo *opaca*.

Da equação anterior obtemos a transformação

$$(\theta, \phi) = \left( \arccos \left( (1 - \xi_1)^{\frac{1}{m+1}} \right), 2\pi\xi_2 \right)$$

de pontos  $(\xi_1, \xi_2)$  uniformemente distribuídos sobre  $[0, 1]^2$  sobre a região de espalhamento da refração.

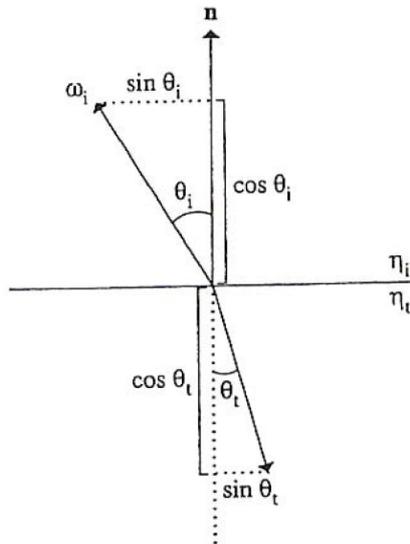


figura 8.1: Geometria para calcular a direção de transmissão  $\omega_t$  a partir da direção de incidência  $\omega_i$ .

## Referências

- [1] [AS00] Ashikhmin, M. and Shirley P. *An anisotropic Phong BRDF model*. 2000.
- [2] [Cook84] Cook R. L., Porter T., Carpenter L. *Distributed Ray Tracing*. Computer Graphics (SIGGRAPH '84 Proceedings), 18(3):137-144, July 1984.
- [3] [Coo86] Cook, R. L. *Stochastic sampling in computer graphics*. ACM TOC, 5(1):51-72, January 1986.
- [4] [Kw86] Kalos, M. H. and Whitlock, P. A. *Monte Carlo Methods*. John Wiley and Sons, 1986.
- [5] [Mit91] Mitchell, D. P. Spectrally optimal sampling for distributed ray tracing. In Sederberg, T. W., editor, Computer Graphics (SIGGRAPH '91 Proceedings), volume 25, pages 157-164, July 1991.
- [6] [PH] Pharr, M and Humphreys, G. *Physically based rendering. From theory to implementation*. Elsevier.
- [7] [Shi90a] Shirley, P. *Physically Based Lighting Calculations for Computer Graphics*. PhD thesis, Dept. of Computer Science, U. of Illinois, Urbana-Champaign, November 1990.
- [8] [VG] Velho, L. and Gomes, J. *Sistemas gráficos 3D*. Instituto de Matemática Pura e Aplicada - IMPA. Serie computação e Matemática.