

UNIVERSIDAD DE GRANADA
ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA
DEPARTAMENTO DE LENGUAJES Y SISTEMAS INFORMÁTICOS

Métodos de Monte-Carlo Eficientes para Iluminación Global.

Tesis doctoral presentada por:
Carlos Ureña Almagro
para la obtención del grado de Doctor en Informática.

Directores
Juan Carlos Torres Cantero.
Xavier Pueyo Sáñez.

Granada
21 de Mayo de 1998

Agradecimientos.

Quiero agradecer a los directores de esta tesis, Juan Carlos Torres Cantero y Xavier Pueyo Sáñez, por el apoyo y los consejos recibidos durante la producción de los resultados descritos aquí, y también por el estímulo constante a la investigación en el campo de la Informática Gráfica que ambos han realizado.

También quiero agradecer a las personas involucradas en el desarrollo del sistema GIRT, que fué usado para probar los algoritmos descritos. Dichas personas son: Jorge Revelles Moreno, Pedro Cano Olivares, Marcelino Cabrera Cuevas, Juan Carlos Torres y Vicente del Sol Lopez. Asimismo, Antonio López Fernández y Miguel Lastra Leidingher contribuyeron en algunas partes del software. También quisiera agradecerles a todos el esfuerzo empleado en mantener el equipamiento informático de nuestro Departamento funcionando.

Quiero agradecer al Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Granada el permitirme dedicar parte de mi tiempo de trabajo a las tareas de investigación, y a usar su infraestructura para escribir esta memoria.

Este trabajo ha sido parcialmente subvencionado con el proyecto codificado como TIC95-0614-C03-02, de la Comisión Interministerial de Ciencia y Tecnología, y también con la Acción Integrada Hispano-Austriaca con código HU96-37.

Parte de los contenidos de esta memoria han sido previamente publicados [Urena97, Urena97a]. También hemos publicado textos que describen nuestro sistema original de síntesis de imágenes [Urena92, Urena93] y el software actual para Iluminación Global nombrado como GIRT [Urena97b]. Me gustaria agradecer a todos los miembros del Grupo de Investigación en Informática Gráfica su trabajo en dichas publicaciones, así como a José Parets su aportación relativa a la Programación Dirigida a Objetos.

Esta memoria está dedicada a mis padres, Antonio Ureña Muñoz y M^a del Carmen Almagro Pérez, y muy especialmente a mi mujer, Yolanda, que conoce bien el esfuerzo empleado en esta tesis, y le da sentido.

Contenidos

Agradecimientos.	3
Lista de Símbolos.	13
Introducción.	19
1 Transporte de luz y métodos computacionales relacionados.	23
1.1 Introducción.	23
1.1.1 Modelo del entorno.	23
1.1.2 Dominios y medidas	24
1.2 Reflexión local de la luz	26
1.2.1 La Radiancia y otras funciones relacionadas.	26
1.2.2 Propiedades de la BRDF.	29
1.2.3 Clases de BRDF	30
1.3 Transporte global de la luz.	34
1.3.1 Ecuaciones integrales para radiancia.	34
1.3.2 La ecuación de radiosidad	39
1.3.3 Operadores integrales y soluciones a ecuaciones integrales.	39
1.3.4 La solución de la ecuación de transporte.	42
1.4 Ecuaciones de transporte adjuntas.	43
1.4.1 Productos escalares y operadores adjuntos.	43
1.4.2 La ecuación integral dual.	44
1.4.3 Ecuaciones de transporte de importancias.	45
1.5 Métodos computacionales.	45
1.5.1 Bases y operadores de proyección.	46
1.5.2 Métodos de elementos finitos.	47
1.5.3 Métodos de Monte-Carlo.	49
1.6 Conclusiones.	51
2 Clasificación de métodos para Iluminación Global	53
2.1 Los conceptos de Imagen y Observador.	53
2.2 Soluciones	54
2.3 Métodos locales.	55
2.3.1 Métodos de Proyección.	55

2.3.2	Trazado de rayos sencillo.	56
2.4	Métodos de discretización o de Elementos Finitos	56
2.4.1	Radiosidad clásica.	56
2.4.2	Métodos de una pasada para entornos no puramente difusos.	58
2.4.3	Métodos de dos pasadas para entornos no difusos.	59
2.4.4	Métodos de Importancia y Multiresolución.	61
2.4.5	Proyección en bases Wavelet.	61
2.4.6	Familias de wavelets.	62
2.4.7	Refinamiento por brillo e importancia y métodos jerárquicos.	63
2.5	Métodos Estocásticos o de Monte-Carlo.	64
2.5.1	<i>Path-tracing</i> o Monte-Carlo desde el observador.	64
2.5.2	Métodos de fotosimulación o seguimiento de partículas.	66
2.6	Conclusiones.	68
3	La Varianza de los Métodos de Monte-Carlo.	69
3.1	Introducción.	69
3.2	La Varianza del Path-Tracing	69
3.2.1	Densidad de Cadenas	70
3.2.2	Una variable aleatoria sobre Cadenas de Markov	72
3.2.3	Análisis de la varianza	73
3.2.4	Probabilidad de transición proporcional al núcleo.	75
3.2.5	Estimación de productos escalares o <i>funcionales</i>	76
3.2.6	Una probabilidad de transición ideal con varianza nula.	78
3.2.7	Aplicaciones a la Iluminación Global.	79
3.3	Path-Tracing con muestreo directo de fuentes.	81
3.3.1	Densidad de cadenas extendidas.	81
3.3.2	Una variable aleatoria sobre cadenas extendidas.	82
3.3.3	La varianza de Y_r	84
3.3.4	Estimación de productos escalares.	86
3.4	Conclusiones.	87
4	Un procedimiento mejorado de refinamiento final para radiosidad.	89
4.1	Introducción.	89
4.2	Refinamiento final por Monte-Carlo.	91
4.3	Distribuciones de muestreo para refinamiento final.	92
4.3.1	La varianza de las distribuciones.	93
4.3.2	Muestreo del ángulo sólido.	93
4.3.3	Muestreo del ángulo sólido proyectado.	95
4.3.4	Muestreo del área.	96
4.3.5	Muestreo de áreas parametrizadas.	96
4.4	Funciones de probabilidad con reutilización de información.	97
4.4.1	Aproximaciones a la distribución de las irradiancias.	97
4.5	Distribuciones de probabilidad ponderadas.	98
4.5.1	Muestreo ponderado de área y área parametrizada.	99
4.5.2	Muestreo ponderado del ángulo sólido.	99
4.5.3	Muestreo ponderado del ángulo sólido proyectado.	101

4.6	Implementación y Resultados.	102
4.7	Conclusiones.	105
5	Refinamiento final para Iluminación Global.	107
5.1	Introducción	107
5.1.1	Trabajos anteriores.	108
5.1.2	El algoritmo propuesto.	108
5.2	Agrupamiento jerárquico.	109
5.3	Primera pasada.	111
5.3.1	Fotosimulación.	111
5.3.2	Información de estado de las partículas.	112
5.3.3	Probabilidades de supervivencia y absorción.	113
5.3.4	Aproximación a la función de radiancias.	113
5.3.5	Actualización de enlaces.	115
5.3.6	Refinamiento de enlaces.	116
5.4	Segunda pasada.	117
5.4.1	Formalización del problema.	117
5.4.2	Una <i>pdf</i> para refinamiento final.	118
5.4.3	Descenso del árbol.	119
5.5	Implementación y resultados.	120
5.6	Conclusiones.	124
6	Principales aportaciones y conclusiones.	125
	Bibliografía	129

Lista de Figuras

1.1	La medida σ para un conjunto de direcciones A .	25
1.2	La medida σ_x para un conjunto de direcciones A .	25
1.3	Reflexión local de radiancias.	28
1.4	Forma de la BRDF difusa.	31
1.5	Forma de la BRDF difuso-especular para $n = 1$	33
1.6	Forma de la BRDF difuso-especular para $n > 1$	33
1.7	La relación entre medidas para el cambio de variable.	36
1.8	La radiancia reflejada como integral de radiancia.	37
1.9	La radiancia como integral en el espacio de los rayos.	38
3.1	Ejemplo de una cadena de rayos.	72
3.2	Un ejemplo de una cadena extendida.	82
4.1	La radiosidad clásica por elementos finitos.	89
4.2	El paso de refinamiento final.	90
4.3	Forma de la PDF muestreo del ángulo sólido.	94
4.4	Forma de la PDF para muestreo del ángulo sólido proyectado.	95
4.5	Forma de la PDF muestreo ponderado del ángulo sólido.	100
4.6	Método de muestreo para el ángulo sólido ponderado.	100
4.7	Forma de la PDF para muestreo ponderado del ángulo sólido proyectado.	101
4.8	Escena simple: 8, 16 y 32 muestras por pixel.	102
4.9	Escena simple: 64, 128 y 256 muestras por pixel.	102
4.10	Escena simple por path-tracing: 64, 256 y 1024 muestras por pixel.	103
4.11	Gráfico de σ^2 con respecto al tiempo de CPU	103
4.12	Resultado de las comparaciones numéricas.	104
4.13	Habitación con 800 trozos, y 4×50 muestras de refinamiento final.	106
5.1	Un conjunto de grupos y objetos.	110
5.2	Arbol correspondiente a la figura 5.1.	110
5.3	El camino seguido por una partícula.	111
5.4	Nodos terminales visitados.5.3.	112
5.5	El mallado de la semiesfera.	114
5.6	Búsqueda del enlace que atraviesa una partícula.	115
5.7	Geometría para una transición entre dos superficies.	116

5.8	Refinamiento de un enlace.	117
5.9	Path-tracing con 4,16 y 32 muestras por pixel	121
5.10	Nuevo metodo con 1,2,4,8,16 y 32 muestras por pixel.	121
5.11	Comparación de métodos: error cuadrático frente a tiempo de CPU.	122
5.12	Una escena difusa compleja con Path-tracing y el nuevo método.	123
5.13	Comparación de eficiencia para la escena de dos mesas.	123
5.14	Escena compleja con 3,6 y 12 millones de partículas.	124

Lista de Tablas

2.1	Métodos de radiosidad clásica.	58
2.2	Métodos de una pasada para entornos no difusos.	59
2.3	Métodos de dos pasadas para entornos no difusos.	61
2.4	Métodos jerárquicos.	64
2.5	Trazado de caminos desde el observador.	66
2.6	<i>Path-tracing</i> desde las fuentes de luz	68

Lista de Símbolos.

- A Medida de área en las superficies de los objetos del entorno. Ver sección 1.1. En la sección 1.5 se usa para significar la matriz cuadrada obtenida como discretización del operador de transporte \mathcal{A} . Está relacionada con *matriz de factores de forma* y sus elementos son c_{ij} (ver definición 1.80).
- \mathcal{A} Operador integral lineal que actúa sobre funciones integrables en el dominio D . Está definido por la función k , según se observa en la ecuación (1.41).
- a Representa al estado de absorción. Es un elemento cualquiera tal que $a \notin D$.
- B Función de S en valores reales. Se le llama función de radiosidad. Cumple la ecuación integral (1.39) y equivale a la radiancia L cuando esta es independiente de la dirección.
- B' Función de S en valores reales. Se obtiene al proyectar la función B en una base ortonormal, como puede observarse en (4.1) (ver capítulo 4).
- \mathcal{B} Operador arbitrario que actúa sobre funciones en el dominio D
- \mathbf{B} Conjunto arbitrario de n funciones base ortonormales en el dominio D .
- B_e Función de S en valores reales. El valor $B_e(x)$ es la parte de $B(x)$ debida a emisión de radiación desde x .
- B_r Función de S en valores reales. El valor $B_r(x)$ es la parte de B debida a reflexión de radiación incidente sobre x .
- b_i Funciones en el dominio de D , pertenecientes a la base \mathbf{B}
- C Conjunto de cadenas infinitas de elementos de D' .
- \mathbf{C} Conjunto de funciones base ortonormales en D , cada una de las cuales es constante. Ver sección 2.4.
- C_i Funciones en D pertenecientes a la base \mathbf{C} (ver ecuación 2.3).
- c_{ij} Elementos de la matriz A . Se definen como $c_{ij} = \langle Ab_i | b_j \rangle$.
- \cos Función de $O \times O$ en valores reales. Aplicado a dos vectores unitarios, es el coseno del ángulo que forman entre sí, esto es, el producto escalar de ambos.
- D Dominio de la función de radiancia, esto es, el conjunto de todos los rayos con origen en las superficies de los objetos. Se define como $D = S \times O$.

- D' Es igual al conjunto D extendido con el estado de absorción a , esto es $D' = D \cup \{a\}$.
- \mathcal{D} Operador de transporte difuso. Es el operador asociado a las reflexiones difusas, cuya BRDF es f_{rd} .
- \mathbf{D}_i Una base ortonormal del espacio vectorial V_i . Ver sección 2.4
- E Función de irradiancia. Para cada punto x de S , $E(x)$ es la irradiancia sobre x , que se mide en Vatios por metro cuadrado. Ver definición (1.4). Aplicado a una variable aleatoria cualquiera X , $E(X)$ denota el valor medio de dicha variable (en el capítulo 3).
- E_n Función con dominio en D y valores reales. Para un elemento r en D y un natural n , el valor $E_n(r)$ es la media de la variable aleatoria X_r^n (ver sección 3.2). Se usan principalmente E_1 y E_2 . En el capítulo 4, designa a una función en S . El valor $E_n(x)$ es la parte de $E_g(x)$ debida a radiación proveniente del trozo n -ésimo (ver definición 4.20).
- E'_n Función con dominio en D y valores reales. Para un elemento r en D y un natural n , el valor $E'_n(r)$ es la media de la variable aleatoria Y_r^n (ver sección 3.3).
- E_g Función con dominio en S y valores reales. Aplicada a un punto x de S , $E_g(x)$ es el resultado exacto del *final-gather* sobre dicho punto (ver definición 4.3). Es un valor aproximadamente igual a la irradiancia sobre dicho punto $E(x)$.
- E'_g Función con dominio en S y valores reales. Aplicada a un punto x de S , $E'_g(x)$ es el resultado aproximado del *final-gather* sobre dicho punto, obtenido mediante muestreo por Monte-Carlo y que debe considerarse como un estimador de $E_g(x)$ (ver definición 4.6).
- F Vector columna de coeficientes de f con respecto a la base ortonormal \mathbf{B} .
- f Función solución de la ecuación integral de transporte (1.27)
- f' Función solución de la ecuación discretizada (1.74).
- f_i Elementos del vector F . Son los coeficientes de la función f' respecto de la base \mathbf{B} , y por tanto $f_i = \langle f' | b_i \rangle$ (ver capítulo 1). En el capítulo 3 se usa para significar la función obtenida al aplicar i veces el operador \mathcal{A} sobre f , esto es, $f_i = \mathcal{A}^i f$. Se usan sobre todo f_1 y f_2 .
- f_r Función bidireccional de distribución de la reflectancia. Se nombra como BRDF. Ver definición (1.10).
- f_{rd} BRDF difusa. Ver definición (1.21).
- f_{rp} BRDF de Phong, o *glossy*. Ver definición (1.23).
- f_{rs} BRDF especular perfecta. Ver definición (1.20).

- G Vector columna de los coeficientes de g respecto de la base \mathbf{B} (ver sección 1.5). También se usa para significar una función de $S \times S$ en valores reales que es el término geométrico incluido entre los factores la función K (ver definición 1.33).
- \mathcal{G} Operador de transporte *glossy*. Es el operador asociado a las reflexiones difuso-especulares, cuya BRDF es f_{rp} .
- g Función arbitraria integrable en el dominio D .
- g_i Elementos del vector G , es decir $g_i = \langle b_i | g \rangle$, donde b_i son los elementos de la base \mathbf{B} (ver sección 1.5). En el capítulo 3 se usa para significar la función obtenida al aplicar i veces el operador \mathcal{A} sobre la función g , esto es, $g_i = \mathcal{A}^i g$.
- H Función con dominio $S \times S$ y valores reales (ver definición 1.30).
- h Función arbitraria integrable en el dominio D .
- I Función definida en el dominio D , pertenece al espacio de funciones generado por la base \mathbf{W} . Se llama función *imagen*. El valor $I(\mathbf{r})$ es la parte de la radiancia que viaja por \mathbf{r} que percibe un observador. Se define en la ecuación (2.2).
- \mathcal{I} Operador identidad. Para cualquier función f en D , se cumple que $\mathcal{I}f = f$.
- K Núcleo del operador de transporte integral \mathcal{T} . Es una función de $D \times D$ en valores reales, definida en (1.36). El valor $K(\mathbf{r}, \mathbf{s})$ es la radiancia en \mathbf{r} debida a una unidad de radiancia en \mathbf{s} .
- k Núcleo del operador integral genérico \mathcal{A} . Es una función arbitraria de $D \times D$ en valores escalares.
- l Función solución de la ecuación integral (1.66), que es la dual de (1.27). Si l existe, se puede escribir como $l = (\mathcal{A}^*)^+ h$.
- l_i Coeficientes de la función I respecto de la base W . Se definen como $l_i = \langle I | W_{ei} \rangle$.
- L Función de D en valores escalares. Para cada \mathbf{r} de D , el valor $L(\mathbf{r})$ es su radiancia (ver definición 1.5).
- L_r Parte de L debida a reflexión de radiación. Se le llama *radiancia reflejada*.
- L_e Parte de L debida a emisión de radiación. Se le llama *radiancia emitida*.
- L_i Función de D en valores reales. Para cada \mathbf{r} de D , el valor $L_i(\mathbf{r})$ es la radiancia incidente en el origen en x proveniente de w , donde $(x, w) = \mathbf{r}$ (ver capítulo 1).
- M Función de S en valores escalares. Para cada punto $x \in S$, $M(x)$ es la energía radiante por unidad de área que abandona x en todas las direcciones. Se le llama radiancia total y está definida en la ecuación (1.3).

M_e	Función de S en valores escalares. $M_e(x)$ es la parte de $M(x)$ debida a emisión de radiación en el punto x .
M_r	Función de S en valores escalares. $M_r(x)$ es la parte de $M(x)$ debida a reflexión de radiación en el punto x .
m	Medida arbitraria definida en D .
m'	Extensión de m al dominio D' , asignando medida unitaria al estado de absorción.
m_K	Supremo esencial (del inglés <i>essential supremum</i>) del conjunto de valores de la función $\mathcal{T}u$, donde u es la función en D constante e igual a 1 en todos sus puntos. Coincide con el supremo esencial de los valores de $\rho(x, w)$. Ver definición (1.55).
m_k	Supremo esencial (del inglés <i>essential supremum</i>) del conjunto de valores de la función $\mathcal{A}u$, donde u es la función en D constante e igual a 1 en todos sus puntos. Ver definición (1.53).
n_x	Para un punto x de las superficies S , representa el vector normal a la superficie en x , con longitud unidad.
O	Conjunto de vectores unitarios o puntos de la esfera de radio unidad.
\mathcal{P}_B	Para cada base arbitraria B , representa el operador de proyección sobre dicha base.
p	Función con dominio en $D' \times D'$ y valores escalares. El valor $p(x, y)$ es la probabilidad condicional de que una cadena que ha visitado el estado x tenga como siguiente estado y . También se ha usado para significar una función de $S \times O$ en S . El punto $p(x, w)$ es el primer punto de S visible desde x en la dirección w .
p_0	Función con dominio en D y valores escalares. El valor $p_0(x)$ es la probabilidad de que x sea el primer estado de una cadena.
p_x	Función con dominio en S y valores escalares. Es la función de densidad de probabilidad usada para realizar el <i>final-gather</i> descrito en el capítulo 4. Cumple las condiciones (4.4) y (4.5).
Q_r	Para cada $r \in D$, es una medida de probabilidad definida en C , que cumple la igualdad (3.41), y sobre la cual se define la variable Y_r para cadenas extendidas.
Q_h	Para cada h (función arbitraria en D con valores reales), es una medida de probabilidad que cumple la igualdad (3.52), y que sobre la cual se define la variable aleatoria Y_h .
r_i	Energía total incidente en el trozo i -ésimo de la escena. (ver capítulo 4).
r_{ij}	Parte de r_i debida a energía incidente desde el trozo j (ver capítulo 4).

- $refl_x$ Función de O en O . Para un punto x en S y un vector w , el vector $refl_x(w)$ es el simétrico de w con respecto a la normal en x (esto es, el vector *reflejado* de w en x). Ver capítulo 1.
- S Conjunto de los puntos en la superficie (o frontera) de los objetos de la escena.
- \mathcal{S} Operador de transporte especular perfecto, inducido por la BRDF f_{rs} .
- \mathbf{S}_i Base ortonormal del espacio vectorial U_i . Ver sección 2.4.
- T Función con dominio en $S \times S$ y valores reales (ver sección 4.2), se define como $T(x, y) = G(x, y)B'(y)$.
- \mathcal{T} Operador integral lineal de transporte de radiancias. Esta definido por el núcleo K en la ecuación (1.38), o también por la ecuación (1.25) vista como un operador.
- \mathcal{U} Operador de transporte de varianzas, definido en el capítulo 3 (ver definición 3.10).
- U_r Para cada $r \in D$, es una medida de probabilidad definida en C , que cumple las igualdades que aparecen en (3.5), y sobre la cual se define la variable X_r .
- U_h Para cada h (función arbitraria en D con valores reales), es una medida de probabilidad que cumple la igualdad (3.32), y que sobre la cual se define la variable aleatoria X_h .
- V Función con dominio en $S \times S$ y valores reales. El valor $V(x, y)$ es uno si x e y son visibles, cero en otro caso. En el capítulo 3, V designa una función de D en los valores reales, de forma que $V(r)$ es la varianza de X_r .
- V_e Función de *varianza emitida* asociada al path-tracing (ver definición 3.24).
- V' Función con dominio en D y valores reales. $V'(r)$ es la varianza de Y_r .
- w_{xy} Para cualquier par de puntos x e y en S , el vector w_{xy} es el vector unitario desde x hacia y , esto es, $w_{xy} = (y - x)/|y - x|$.
- W Función de potencial o importancia. Se obtiene como la suma de todas las W_i .
- \mathbf{W} Conjunto de funciones base ortonormales en D , que representan a un observador. Sus elementos son W_{ei} . Ver capítulo 2.
- W_{ei} Cada una de las funciones base que forman \mathbf{W} . Se le suele llamar *potencial emitido* asociado al i -ésimo pixel de la imagen. Ver capítulo 2.
- W_i Conjunto de funciones con dominio en D y valores reales. A cada W_i se le llama *potencial* asociado al i -ésimo pixel de la imagen. Ver capítulo 2.
- X_r Variable aleatoria definida sobre cadenas infinitas (ver definición 3.6). El subíndice r es un elemento de D que interviene en su definición.

X_h	Variable aleatoria definida sobre cadenas infinitas (ver definición 3.29). El subíndice h designa una función de D en valores reales que interviene en su definición.
Y_r	Variable aleatoria definida sobre cadenas infinitas (ver definición 3.42). El subíndice r es un elemento de D que interviene en su definición.
Y_h	Variable aleatoria definida sobre cadenas infinitas (ver definición 3.53). El subíndice h designa una función de D en valores reales que interviene en su definición.
Z_r	Variable aleatoria definida sobre D (ver definición 3.43).
δ	La función delta de Dirac. Se cumple que $\int_I \delta(x)f(x)dx = f(0)$ para cualquier intervalo I de valores reales que incluyan el cero.
Δ	Función de S en valores reales. El valor $\Delta(x)$ es el error cometido al aproximar $E_g(x)$ por r_i , donde x está en el i -ésimo trozo (ver definición 4.21).
Δ_j	Función de S en valores reales. El valor $\Delta_j(x)$ es el error cometido al aproximar $E_j(x)$ por r_{ij} , donde x está en el i -ésimo trozo (ver definición 4.22).
Φ	Función de medida definida en S . Para cualquier región $S' \subseteq S$, el valor $\Phi(S')$ es la potencia total saliente de los puntos de S . Se mide en Vatios.
Φ_r	Función de medida en S . Parte de Φ debida a reflexión de radiación.
Φ_e	Función de medida en S . Parte de Φ debida a emisión de radiación.
Φ_i	Potencia total saliente del i -ésimo nodo en la jerarquía de grupos y trozos descrita en el capítulo 5.
σ	Función de medida en O . Es la medida de ángulo solido en la esfera. Para cualquier $O' \subseteq O$, $\sigma(O')$ es el ángulo sólido cubierto por O' .
σ_x	Función de medida en O . Esta medida se define por su relación con la medida σ . Para cualquier punto x de S y cualquier vector w de O , se cumple que $d\sigma_x(w)/d\sigma(w) = \cos(n_x, w)$.
μ	Función de medida en D . Se obtiene como la composición de la medida de área A y la medida σ_x . Para cualquier $\mathbf{r} = (x, w) \in D$, se cumple que $d\mu(\mathbf{r}) = d\mu(x, w) = dA(x)d\sigma_x(w)$.
ρ	Función de D en valores reales entre cero y uno. Se le llama reflectividad direccional-semiesférica. El valor $\rho(x, w)$ es la cantidad de energía total reflejada desde x en todas las direcciones por cada unidad de radiancia incidente desde la dirección w . (ver definición 1.13).
ρ_d	Parte de ρ debida a reflexión difusa.
ρ_s	Parte de ρ debida a reflexión especular perfecta.

Introducción.

Esta tesis está dedicada a la mejora de las técnicas de Monte-Carlo para la solución numérica de la ecuación integral que caracteriza el transporte de flujo electromagnético visible. Aunque esta parece ser una sentencia muy técnica, el propósito básico de estos métodos es muy simple: la creación de imágenes realistas de objetos del mundo real. De hecho, este ha sido el objetivo de muchos esfuerzos en el campo de las ciencias y el arte, desde el principio del ser humano. Téngase en cuenta que unos de los primeros resultados visibles de la cultura humana son las herramientas de caza y los dibujos en las paredes de las cuevas.

A lo largo de la historia del arte, podemos ver que los artistas se acercan, con un nivel de acierto creciente, a la creación de imágenes realistas. Algunas de las técnicas usadas no se expresaron explícitamente, y por tanto no se pudo transferir este conocimiento a otras personas. Las primeras expresiones explícitas de estos conceptos fueron realizadas en los siglos XV y XVI. En esta época se encontraron las leyes que rigen las proyecciones en perspectiva. Esto permitió realizar cuadros en los cuales los distintos elementos se perciben a diferentes profundidades. Este esfuerzo en la percepción de la profundidad también llevó al descubrimiento del *sfumato*. En este caso, la atenuación de la intensidad de la luz (debido a la dispersión atmosférica) fue usada para lograr una mayor percepción de la profundidad en los paisajes.

Todos estos son unos pocos ejemplos de como el estudio de la percepción humana, la geometría y la física son necesarios para la creación de imágenes realistas. Podría argumentarse que la invención de la fotografía hace inútiles estas técnicas. Pero aún así, es necesaria la creación de esta clase de imágenes, por ejemplo cuando el objeto a representar aún no existe, o no se pueden tomar fotos de él. El uso de los ordenadores ayuda en la mayoría de las actividades humanas, así que no es una sorpresa que también ocurra esto en la producción de imágenes. La Informática Gráfica es la rama de la Informática dedicada a la creación y procesado de imágenes y animaciones usando ordenadores. La creación de imágenes no es, con mucho, el único objetivo de la Informática Gráfica, pero se ha incluido como uno de sus aspectos. A este proceso se la ha llamado *rendering*, cuya traducción literal al Español es *revelado*, que parece poco apropiada. Nosotros usaremos simplemente *síntesis de imágenes* para este concepto. Más específicamente, podremos hablar de síntesis de imágenes realistas o simplemente de síntesis realista. La síntesis realista esta relacionada con otros aspectos de la informática gráfica, como son el modelado, el almacenamiento y visualización de imágenes, y la ingeniería de software para gráficos. Pero también la síntesis realista usa resultados tomados de la física, de las matemáticas y del estudio del sistema perceptivo visual del hombre.

El primer problema resuelto (directamente relacionado con síntesis realista) fue

la visualización en perspectiva de modelos de alambre de los objetos. El siguiente paso para conseguir realismo fue la eliminación de las partes no visibles debido a la oclusión por otras partes opacas (a esto se le llama el problema de eliminación de partes ocultas). Se propusieron e implementaron técnicas eficientes con este propósito. Sin embargo, las imágenes así conseguidas no parecen realistas debido a la intensidad constante de los colores dentro del trozo de imagen donde se proyecta cada objeto. Muy pronto se advirtió la necesidad de incluir alguna clase de variación en el color, simulando el sombreado, de manera que esto ayudase en la percepción de la forma y posición relativa de los objetos. Se propusieron algunas técnicas simples para producir sombreado. Estas técnicas se usaban debido a que eran suficientemente simples para el poder de cálculo de la época al tiempo que producían niveles aceptables de realismo. Según la potencia de cálculo iba en aumento, las técnicas descritas se usaron para producir animaciones e imágenes estáticas en muy poco tiempo de cálculo. La simplicidad de algunos de estos algoritmos permitió la implementación en hardware de los mismos. Todo esto llevó a la visualización interactiva y en tiempo real de animaciones, lo cual dio lugar a la realidad virtual.

Al mismo tiempo, otros investigadores se centraron en la creación de imágenes más realistas. Pronto se encontró que producir imágenes reales solo podría conseguirse prestando la necesaria atención a las leyes que gobiernan el transporte de radiación visible. Estas ecuaciones se expresan habitualmente en la forma de ecuaciones integrales de Fredholm, y se usan para derivar algoritmos de síntesis realista. Fueron usadas anteriormente para simular el flujo de calor por radiación, así como el flujo de partículas sub-atómicas, como los neutrones.

Actualmente son usadas dos familias de métodos: los de Elementos Finitos y los de Monte-Carlo. Los primeros de ellos se usan tradicionalmente en la solución de ecuaciones integrales y diferenciales. En esencia, estos métodos están basados en la reducción de las ecuaciones continuas originales a sistemas de ecuaciones discretos. Estos sistemas de ecuaciones se pueden resolver en tiempo finito, y su solución puede ser almacenada en la memoria de los ordenadores.

Los métodos de Monte-Carlo están basados en el muestreo de Cadenas de Markov para obtener estimadores del valor de la función desconocida en determinados puntos. Como en el resto de los métodos de Monte-Carlo, es necesario definir una variable aleatoria cuya media sea el valor desconocido. Esta variable aleatoria debe ser muestreada, y de esta forma se obtiene una aproximación a su media, promediando un número suficiente de muestras.

Ambas clases de métodos poseen ventajas y desventajas. Los métodos de elementos finitos son usualmente más rápidos cuando el entorno presenta una complejidad baja o media. Además, pueden hacer uso de una forma natural de la coherencia espacial en la función a calcular. Sin embargo, para entornos muy complejos, o bien para comportamientos reflectivos no difusos sus tiempos de cálculo se disparan. Los métodos de Monte-Carlo suelen presentar unos tiempos de ejecución muy altos, debido a la necesidad de tomar muchas muestras para reducir el ruido debido a la varianza. Sin embargo, el incremento de los tiempos de cálculo no depende tanto de la complejidad de la escena. Por lo tanto, los métodos de elementos finitos se han usado menos para escenas complejas con comportamiento reflectivo no puramente difuso, mientras en estos casos los métodos de Monte Carlo se usan más

frecuentemente.

En esta memoria proponemos técnicas que mejoran la eficiencia de los algoritmos de Monte-Carlo. La baja eficiencia es causada por el error o varianza tan alta que se obtiene para un número razonable (respecto al tiempo) de muestras. Esta varianza es una medida acerca de lo lejana que estará la solución que obtengamos de la solución exacta real. Por lo tanto, se obtendrá mas eficiencia cuanto más baja sea dicha varianza. Así, es necesario entender correctamente la naturaleza de la varianza. En esta memoria se estudia el problema de la Iluminación Global, profundizándose en la caracterización de dicha varianza. Encontramos que la varianza se obtiene como solución de una ecuación integral de segundo orden y usando este resultado se obtiene el algoritmo *ideal* de varianza nula. Este algoritmo es ideal en el sentido de que usa información desconocida *a priori*, por lo cual es irrealizable. A partir de aquí, tratamos de obtener algoritmos de dos pasadas de varianza reducida, basándonos en el resultado anterior. Para ello se usa la primera pasada, que obtiene la información necesaria en la segunda para guiar el proceso de muestreo.

La estructura de la memoria es como se describe a continuación: En el primer capítulo se define la notación a usar y se derivan las ecuaciones que caracterizan el transporte de radiosidad y radiancia. Partiendo de estas ecuaciones, en el capítulo segundo se obtiene una formalización del problema de Iluminación Global, y se usa esa formalización para clasificar los algoritmos que lo resuelven. Se presentan los algoritmos de forma tabular, junto con sus principales características, con el objetivo de facilitar su comparación. En el capítulo tercero el estudio se centra los métodos de Monte-Carlo. Se obtiene una caracterización de su varianza, así como una expresión de la estrategia ideal de muestreo. En el siguiente capítulo, se introduce un algoritmo de dos pasadas para radiosidad que aproxima esa estrategia ideal de muestreo. La idea central es rehusar la información de la primera pasada durante la segunda. Finalmente, en el capítulo quinto la anterior idea se extiende a entornos complejos de reflectividad no difusa. Aquí se usan las técnicas de agrupamiento jerárquico de objetos que han sido usadas en el contexto de los métodos de elementos finitos.

Capítulo 1

Transporte de luz y métodos computacionales relacionados.

1.1 Introducción.

En este capítulo, describimos las cantidades que modelan el transporte de luz, y derivamos las ecuaciones que los relacionan. Primero, introducimos el modelo del entorno en el cual el flujo energético tiene lugar, y después el dominio de las funciones y las funciones de medida usadas en las integrales. Usando estas herramientas matemáticas es fácil derivar dichas relaciones. Dichas derivaciones pueden encontrarse en la literatura sobre Iluminación Global, vease, especialmente, [Glassner95, Cohen93]. Hemos querido incluirlas aquí para introducir la notación que se usará en el resto de la memoria, al tiempo que deseábamos mostrar su coherencia. Además se llega a una expresión ligeramente diferente de la *ecuación de rendering*. Esta expresión tiene exactamente la forma de una *ecuación de Fredholm de segunda clase*, lo cual facilita la introducción de los algoritmos que la resuelven.

Los aspectos puramente matemáticos del discurso¹ no se tratan con gran profundidad. El lector que desee conocer más detalles acerca de los mismos, en los casos que no aparecen aquí, puede recurrir a un diccionario de matemáticas, como, por ejemplo [Ito93].

1.1.1 Modelo del entorno.

En adelante, supondremos un entorno constituido por un número de objetos en el espacio tridimensional. En algunas partes de la superficie de estos objetos se emite

¹Especialmente importantes son los espacios medibles, las funciones de medida, la integración de Lebesgue, las funciones de medida de probabilidad, los espacios vectoriales de funciones integrables y los conjuntos de funciones base, entre otros.

energía en forma de ondas o radiación electromagnética, con intensidad constante a lo largo del tiempo. Estas ondas se reflejan en otras superficies. Consideramos que se ha alcanzado un estado de equilibrio en dicho entorno, en el sentido de que la temperatura de los objetos no cambia, y la intensidad de la radiación emitida y reflejada por los objetos también permanece constante en el tiempo. Las superficies de los objetos son opacas, así que no hay radiación que las atraviese (toda ella es absorbida o reflejada, pero no refractada). Además, dichas superficies tienen definido un vector normal en cada punto, que es continuo excepto en un conjunto de puntos de medida nula.

Supondremos también que en el espacio libre entre los objetos no hay ningún medio o material que disperse la radiación. De esta forma, cualquier punto está en el espacio vacío o pertenece a algún objeto. Los objetos son conjuntos compactos de puntos, es decir la frontera de cada objeto pertenece al objeto. Además, las superficies son continuas y existe un vector normal a dicha superficie en cada uno de sus puntos.

Las ondas electromagnéticas que supondremos tienen una sola longitud de onda. Todas las funciones que describen el flujo de partículas dependen de la longitud de onda a la que sean medidas. Esta dependencia estará implícita en nuestras formulaciones, y la longitud de onda no aparece en la notación explícitamente.

Solo consideraremos entornos cerrados. Si hay que tratar un entorno abierto, podemos incluir en él una superficie perfectamente negra que rodee a todos los otros objetos. Este entorno modificado (cerrado) tiene exactamente el mismo comportamiento que el original (abierto), por que el flujo que abandona el entorno abierto es absorbido en el objeto negro que rodea al cerrado. El flujo es por tanto el mismo en ambos.

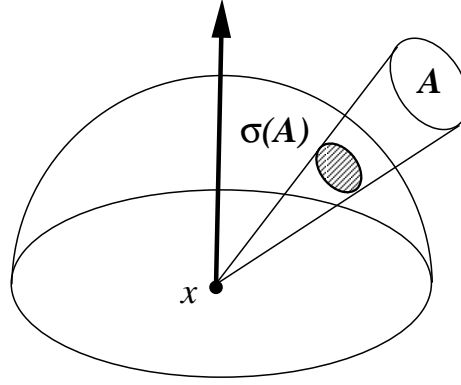
1.1.2 Dominios y medidas

Llamaremos S al conjunto de las superficies de los objetos descritos arriba. Para cada punto $x \in S$, existe un vector normal a la superficie en ese punto, que notaremos como n_x . Esto implica que podemos usar como función de medida en S la medida del área, que notaremos con A . Si $S' \subseteq S$, entonces $A(S') = \int_{S'} dA(x)$. Aquí, $dA(x)$ es un diferencial de superficie entorno al punto x .

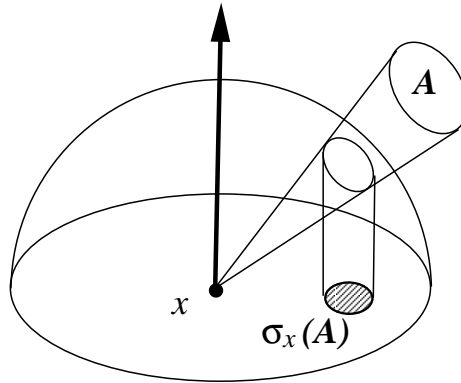
Las funciones en que vamos a tratar están definidas en el espacio de los vectores unitarios de dirección y los puntos de las superficies. Definiremos el dominio D , como $D = S \times O$, donde O es el conjunto de los vectores de longitud uno (equivalente a la superficie de la esfera).

En la esfera unidad existe la medida de ángulo sólido de nombre σ . Para un conjunto de direcciones A , el valor real $\sigma(A)$ es el área de la región de la esfera ocupada por dicho conjunto de direcciones. En la figura 1.1, la medida del conjunto A es el área de la superficie sombreada. Para cada punto $x \in S$ se puede definir otra medida de ángulo sólido proyectado. Llamaremos a esta medida σ_x . Para cada subconjunto O' de vectores unitarios, con $O' \subseteq O$, haremos la siguiente definición de la medida σ_x :

$$\sigma_x(O') = \int_{O'_x} \cos(w, n_x) d\sigma(w) \quad (1.1)$$

Figura 1.1: La medida σ para un conjunto de direcciones A .

donde $O'_x = \{v \in O' \mid \cos(v, n_x) > 0\}$. Esto es, O'_x es el subconjunto de vectores de O' que están en la semiesfera sobre x , siendo el eje de dicha semiesfera n_x . Esta definición implica que las direcciones en la semiesfera bajo el punto tienen medida nula. En la figura 1.2, se observa como, para un conjunto de direcciones A , el valor $\sigma_x(A)$ es el área de la región sombreada. Dicha región se encuentra en el círculo bajo la semiesfera positiva, y se obtiene proyectando verticalmente la porción de semiesfera cubierta por el conjunto de direcciones A . Una vez que las medidas A y

Figura 1.2: La medida σ_x para un conjunto de direcciones A .

σ_x han sido introducidas, definimos una nueva medida μ en D que será usada más tarde en la integral que gobierna el transporte de radiancias. Consideremos cualquier elemento $r \in D$. Esto implica que $r = (x, w)$ donde $x \in S$ y $w \in O$. La medida μ se obtiene como la composición de las dos medidas previamente introducidas,

cumpléndose lo siguiente:

$$d\mu(\mathbf{r}) = d\mu(x, w) = dA(x) d\sigma_x(w) \quad (1.2)$$

. La medida μ se llama *throughput measure* [Veatch97], que podría traducirse por *medida de la capacidad*. Es muy frecuentemente usada en Iluminación Global, y de ahí su importancia en este texto.

1.2 Reflexión local de la luz

1.2.1 La Radiancia y otras funciones relacionadas.

Para cualquier subconjunto S' de S , podemos medir la energía radiante total que abandona dicho trozo de superficies por unidad de tiempo (esto es, la potencia). Llamaremos a esta potencia $\Phi(S')$, que se mide en Vatios. El valor $\Phi(S')$ puede expresarse como la suma de otros dos: $\Phi(S') = \Phi_e(S') + \Phi_r(S')$, donde Φ_e es una medida de la potencia total debida a energía que ha sido generada o emitida desde S' , y $\Phi_r(S')$ mide la energía total reflejada en S' debida a energía que llega a S' desde el entorno. También podemos medir la energía total incidente en S' y que proviene del entorno. Notaremos esta medida como Φ_i . Todas estas funciones de medida tienen el mismo dominio y unidades.

Normalmente, la cantidad de energía que abandona una superficie difiere de unos puntos a otros. Para poder modelar esta variación, definimos la *radiancia saliente*, como una función M que se obtiene al diferenciar la medida Φ con respecto a la medida de área, obteniéndose potencia por unidad de área, esto es:

$$M(x) = \frac{d\Phi(x)}{dA(x)} \quad (1.3)$$

donde $x \in S$. De la misma forma podemos definir la radiancia emitida y la reflejada, haciendo $M_e(x) = d\Phi_e(x)/dA(x)$ y $M_r(x) = d\Phi_r(x)/dA(x)$. La función de *irradiancia* E se define igual que las anteriores, pero con respecto a la potencia incidente:

$$E(x) = \frac{d\Phi_i(x)}{dA(x)} \quad (1.4)$$

Todas estas funciones M, M_r, M_e y E tienen unidades de Vatios por metro cuadrado ($W m^{-2}$). La potencia incidente que alcanza cualquier punto de las superficies puede ser absorbida o reflejada. En este último caso supondremos que la potencia reflejada abandona la superficie exactamente desde el mismo punto donde se recibió la potencia incidente que causa la reflexión. Esto implica que $M(x) = M_e(x) + M_r(x)$, donde la potencia reflejada $M_r(x)$ es debida a reflexión en x de la potencia incidente $E(x)$.

El valor $M(x)$ mide la energía que abandona x en todas las direcciones posibles. Podemos también medir la energía saliente en un subconjunto de direcciones O' incluido en O . Llamaremos a este valor $M(x, O')$. Así podemos considerar a M

como una medida en O . Usando esta medida podemos definir la *radiancia*:

$$L(\mathbf{r}) = L(x, w) = \frac{dM(x, w)}{d\sigma_x(w)} \quad (1.5)$$

donde $\mathbf{r} = (x, w)$. Se puede ver que la radiancia tiene a D como dominio, esto es, está definida para cada par punto-dirección en D . Esta función, al ser evaluada en un punto $x \in S$ y una dirección $w \in O$, proporciona la potencia saliente de x en la dirección w , por unidad de área y por unidad de ángulo sólido. Por tanto, se mide en Vatios por estereoradián y por metro cuadrado, y se nota como L . En consistencia con previas definiciones, haremos las siguientes:

$$L_e(x, w) = \frac{dM_e(x, w)}{d\sigma_x(w)} \quad (1.6)$$

$$L_r(x, w) = \frac{dM_r(x, w)}{d\sigma_x(w)} \quad (1.7)$$

$$L_i(x, w) = \frac{dE(x, w)}{d\sigma_x(w)} \quad (1.8)$$

todas estas funciones se miden también en Vatios por estereoradián y por metro cuadrado. Las definiciones anteriores implican las siguientes igualdades, que incorporan integrales:

$$M_r(x) = \int_O L_r(x, w) d\sigma_x(w) \quad E(x) = \int_O L_i(x, w) d\sigma_x(w) \quad (1.9)$$

nótese que hemos establecido que $M_r(x)$ es debido a reflexión en x de $E(x)$. No existe una relación similar entre L_r y L_i , es decir, la energía que alcanza un punto en una dirección se puede reflejar en otras direcciones distintas de la de entrada. La relación entre la energía entrante en una dirección y la parte de ella que se refleja en otra dirección esta definida por la *Función Bi-direccional de Distribución de la Reflectancia* o simplemente BRDF (por las siglas en inglés).

Con el objetivo de definir la BRDF, podemos ver a L_r y a E como funciones de medida en O . Respecto a L_r , podemos medir la parte de $L_r(x, w)$ debida a energía incidente desde un conjunto de direcciones O' (es decir, la energía proveniente de direcciones en O' que se refleja en la dirección w), y llamar a esto $L_r(x, w, O')$. Igualmente, podemos considerar simplemente la cantidad de potencia incidente en x desde un conjunto de direcciones O' , llamando a esto $E(x, O')$. La BRDF (que llamaremos f_r) se define como la relación entre estas dos medidas:

$$f_r(x, w_o, w_i) = \frac{dL_r(x, w_o, w_i)}{dE(x, w_i)} \quad (1.10)$$

Con esta definición, vemos que $f_r(x, w_o, w_i)$ es la cantidad de radiancia reflejada desde x en la dirección w_o debido a una unidad de irradiancia proveniente de la dirección w_i . Podemos reescribir la definición de f_r como sigue:

$$f_r(x, w_o, w_i) = \frac{dL_r(x, w_o, w_i)}{L_i(x, w_i) d\sigma_x(w_i)} \quad (1.11)$$

usando esta última igualdad, podemos descomponer el valor $L_r(x, w_o)$ como la suma integral de las contribuciones a dicho valor desde todas las direcciones incidentes. Esto se hace integrando la igualdad (1.11) para todas las direcciones w_i en O :

$$L_r(x, w_o) = \int_O f_r(x, w_o, w_i) L_i(x, w_i) d\sigma_x(w_i) \quad (1.12)$$

en la figura 1.3 se puede observar como la radiancia reflejada L_r en un punto se obtiene como una integral sobre todas la direcciones de la semiesfera de la radiancia incidente en cada dirección, según se desprende de la ecuación (1.12).

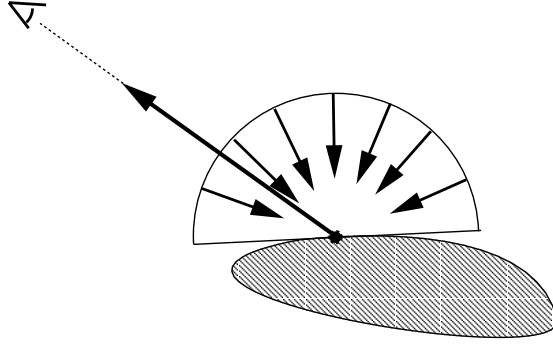


Figura 1.3: Reflexión local de radiancias.

Una función relacionada con la reflexión en un punto es la función de *reflexividad direccional-semiesférica*. Esta función proporciona, para cada punto x y dirección w_i , la fracción de radiancia incidente en x desde w_i que es reflejada en todas las direcciones sobre x . Esta nueva función se llamará ρ . Se puede definir alternativamente mediante integración o diferenciación.

Respecto a la versión integral, podemos ver $\rho(x, w_i)$ como la potencia reflejada en todas las direcciones desde x debido a una unidad de irradiancia sobre x proveniente de w_i . Podemos expresar entonces $\rho(x, w_i)$ como igual a $M_r(x)$ cuando la función $L_i(x, w)$ vale uno en $w = w_i$ y cero en todas las otras direcciones (esto es, $L_i(x, w) = \delta(w - w_i)$). Si sustituimos esta versión de L_i en (1.12), la integral desaparece por la presencia de una función δ y obtenemos que $L_r(x, w_o) = f_r(x, w_o, w_i)$. Podemos ahora usar (1.9) y obtenemos:

$$\rho(x, w_i) = M_r(x) = \int_O L_r(x, w_o) d\sigma_x(w_o) = \int_O f_r(x, w_i, w_o) d\sigma_x(w_o) \quad (1.13)$$

vemos por tanto que ρ esta completamente determinada por la BRDF.

Usando diferenciación, escribimos ρ como la relación entre radiancia saliente total desde x (debida a radiancia incidente desde w_i) respecto a radiancia incidente en x desde w_i

$$\rho(x, w_i) = \frac{dM'_r(x, w_i)}{dE(x, w_i)} = \frac{dM'_r(x, w_i)}{L_i(x, w_i) d\sigma_x(w_i)} \quad (1.14)$$

donde $M'_r(x, O')$ es una medida de la radiancia saliente desde x debido a reflexión de irradiancia proveniente de direcciones en O' y (no usamos M_r para distinguir esta medida de la que aparece en la ecuación (1.7)) y $E(x, O')$ es la misma medida usada en la ecuación (1.8). El valor M_r puede ponerse como la suma integral de $dM'_r(x, w_i)$ para todas las direcciones w_i en O . De esta forma obtenemos

$$\begin{aligned} M_r(x) &= \int_O dM'_r(x, w_i) \\ &= \int_O \rho(x, w_i) dE(x, w_i) \\ &= \int_O \rho(x, w_i) L_i(x, w_i) d\sigma_x(w_i) \end{aligned}$$

1.2.2 Propiedades de la BRDF.

Hasta ahora no se ha impuesto ninguna restricción a la BRDF. Pero existen algunas leyes físicas que llevan a propiedades que toda BRDF debe cumplir necesariamente. Dichas propiedades son la conservación de energía y la simetría.

Conservación de la energía

La potencia reflejada en un punto no puede ser mayor que la incidente sobre ese punto, ya que la potencia absorbida no puede ser negativa. Esta implica que la irradiancia sobre un punto x es mayor que la radiancia reflejada desde dicho punto. Formalmente, para todo $x \in S$ se cumple que:

$$M_r(x) \leq E(x) \quad (1.15)$$

la condición anterior se debe cumplir para todas las posibles distribuciones de radiancia entrante (L_i). Esto incluye el caso en el que $L_i(x, w) = \delta(x - w_i)$, esto es, toda la energía alcanza x desde una sola dirección w_i . Como hemos visto en una sección anterior en este caso $E(x) = 1$ y $M_r(x) = \rho(x, w_i)$. Por tanto, si se cumple la conservación de la energía entonces también se cumple que

$$\rho(x, w_i) \leq 1 \quad (1.16)$$

para todos las direcciones entrantes w_i . Se puede probar, integrando, que la ecuación anterior (1.16) implica la conservación de la energía (1.15), como está descrito en [Lewis93]. Para ello podemos integrar ambos lados de la desigualdad (1.16), multiplicando por una función de radiancia entrante cualquiera L_i . Se obtiene:

$$\int_O \rho(x, w_i) L_i(x, w_i) d\sigma_x(w_i) \leq \int_O L_i(x, w_i) d\sigma_x(w_i) \quad (1.17)$$

a partir de esta desigualdad, usando (1.9) y (1.15), se obtiene directamente (1.15). Como las implicaciones en ambos sentidos han sido demostradas, se ve que las dos desigualdades (1.15) y (1.16) son equivalentes, aunque la segunda de ellas es más fácil de comprobar pues solo depende de la BRDF. La función ρ , por tanto, siempre estará acotada por el valor 1.

Reciprocidad o simetría

La *Ley de Reciprocidad de Helmholtz* establece que la posibilidad que tiene una partícula de ser reflejada desde una dirección entrante a una dirección de salida es la misma si se intercambian ambas direcciones. Formalmente, esto se convierte en una propiedad de la BRDF, ya que se debe cumplir que, en todo punto x , y para todos los pares de direcciones w_i y w_o que:

$$f_r(x, w_i, w_o) = f_r(x, w_o, w_i) \quad (1.18)$$

1.2.3 Clases de BRDF

Tradicionalmente, en la literatura sobre Informática Gráfica se han usado varias clases bien conocidas de modelos de reflexión local. En esta sección, mostramos las expresiones de las BRDF que modelan esas clases de reflexiones, e intentamos ver si cumplen las anteriores restricciones. Estos son los modelos más conocidos y sencillos, aunque existen otros que no trataremos aquí por razones de espacio. Para más detalles, se pueden consultar los siguientes artículos: [Lewis93] y [Schlick94].

Reflexión especular perfecta.

La reflexión especular perfecta se produce en la superficie de los espejos perfectamente pulidos. En un punto cualquiera x , la radiancia reflejada en la dirección w_o es proporcional a la radiancia incidente desde la dirección simétrica (con respecto a n_x). Esto es totalmente independiente de la radiancia proveniente de otras direcciones. Por tanto, se cumple que:

$$L_r(x, w_o) = \rho_s(x) L_i(x, refl_x(w_o)) \quad (1.19)$$

donde $refl_x(w_o) = 2 \cos(n_x, w_o) n_x - w_o$, esto es, $refl_x(w_o)$ es el vector reflejado de w_o en el punto x , y ρ_s es la función que da la proporción de energía reflejada de esta forma, cumpliendo que $\rho_s(x) \leq 1$.

Esto implica que solo una dirección contribuye a la radiancia saliente. La BRDF debe ser cero para todas las direcciones menos una, pero su integral no debe ser cero. Esta propiedad se cumple por la función Delta de Dirac. Podemos definir la BRDF especular (que notaremos como f_{rs}) como sigue [Cohen93]:

$$f_{rs}(x, w_o, w_i) = \rho_s(x) \frac{\delta(w_i - refl_x(w_o))}{\cos(n_x, w_i)} \quad (1.20)$$

podemos demostrar que (1.20) implica (1.19) usando la ecuación (1.12), como sigue:

$$\begin{aligned} L_r(x, w_o) &= \int_O \rho_s(x) \frac{\delta(w_i - refl_x(w_o))}{\cos(n_x, w_i)} L_i(x, w_i) d\sigma_x(w_i) \\ &= \rho_s(x) \int_O \delta(w_i - refl_x(w_o)) L_i(x, w_i) d\sigma(w_i) \\ &= \rho_s(x) L_i(x, refl_x(w_o)) \end{aligned}$$

aquí hemos usado las propiedades de la función δ , y también hemos hecho la siguiente sustitución: $d\sigma_x(w) = \cos(n_x, w) d\sigma(x)$

La simetría de esta BRDF es fácil de probar, por la reciprocidad de la función de reflexión $refl_x$. Se cumple que $w = refl_x(w')$ si y solo si $w' = refl_x(w)$. Por tanto $\delta(w_i - refl_x(w_o)) = \delta(w_o - refl_x(w_i))$. Además, también es cierto que $\cos(n_x, w) = \cos(n_x, refl_x(w))$. Con estas propiedades se obtiene directamente la simetría de f_{rs} .

La conservación de la energía se puede comprobar viendo la reflectividad direccional-semiesférica de esta BRDF

$$\begin{aligned} \rho(x, w_i) &= \int_O f_{rs}(x, w_o, w_i) d\sigma_x(w_o) \\ &= \int_O \rho_s(x) \frac{\delta(w_i - refl_x(w_o))}{\cos(n_x, w_i)} d\sigma_x(w_o) \\ &= \rho_s(x) \int_O \delta(w_i - refl_x(w_o)) d\sigma(w_o) \\ &= \rho_s(x) \\ &\leq 1 \end{aligned}$$

Por lo tanto, $\rho(x, w_i) = \rho_s(x)$. La reflectividad es independiente de la dirección, y se cumple la conservación de la energía ya que $\rho_s(x)$ es menor que uno.

Reflexión difusa.

La BRDF difusa es independiente de las direcciones de entrada y salida. Esto es, el valor $f_r(x, w_o, w_i)$ solo depende del punto x . Esto puede escribirse como $f_r(x, w_o, w_i) = f_{rd}(x)$. En la figura 1.4 puede observarse un diagrama de esta BRDF. Nótese como, independientemente del ángulo de entrada, la radiancia reflejada para cualquier ángulo de salida es la misma. Con esta definición de f_r ,

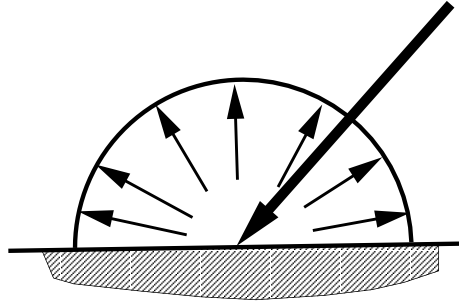


Figura 1.4: Forma de la BRDF difusa.

podemos usar (1.9) y (1.12) para obtener la radiancia saliente M_r como:

$$M_r(x) = \int_O L_r(x, w_o) d\sigma_x(w_o)$$

$$\begin{aligned}
&= \int_O \left[\int_O f_r(x) L_i(x, w_i) d\sigma_x(w_i) \right] d\sigma_x(w_o) \\
&= f_{rd}(x) \int_O \left[\int_O L_i(x, w_i) d\sigma_x(w_i) \right] d\sigma_x(w_o) \\
&= f_{rd}(x) \int_O E(x) d\sigma_x(w_o) \\
&= f_{rd}(x) E(x) \int_O d\sigma_x(w_o) \\
&= f_{rd}(x) E(x) \sigma_x(O) \\
&= f_{rd}(x) E(x) \pi
\end{aligned}$$

de lo anterior vemos que $M_r(x)/E(x) = f_{rd}(x)\pi$. A este valor lo llamaremos la *reflectancia* en x . Es la fracción de potencia reflejada respecto de la potencia incidente en x , y los escribiremos como $\rho_d(x)$. Con todo esto, podemos establecer la definición de $f_{rd}(x)$:

$$f_{rd}(x, w_o, w_i) = f_{rd}(x) = \frac{\rho_d(x)}{\pi} \quad (1.21)$$

es obvio que esta BRDF es simétrica. La conservación de la energía puede probarse expandiendo la función ρ . Es fácil probar que

$$\rho(x, w_i) = \rho_d(x) \quad (1.22)$$

para todas las direcciones de entrada w_i

Reflexión difuso-especular.

Las dos BRDF introducidas anteriormente son muy sencillas. Aunque algunos materiales reales muestran este comportamiento, hay también muchas otras clases de superficies que reflejan la energía de otras formas. En las publicaciones es posible encontrar muchos artículos que describen el modelo e implementación de otros tipos de BRDF. Probablemente, el modelo más simple y más antiguo de esta clase de superficies es el de *Phong* [Phong75]. Este modelo era una solución intermedia entre eficiencia y exactitud física. A este modelo se le llamó una *fórmula para sombreado*, esto es, un algoritmo destinado a calcular la intensidad de la luz reflejada, para un conjunto de luces puntuales dado.

Después de esto, en [Lewis93], esa fórmula para sombreado fue interpretada como el modelo de una BRDF, y se comprobó que este modelo no cumpliría la conservación de la energía. En este artículo, algunos modelos modificados de BRDF se proponían que cumplieran la conservación de la energía y la reciprocidad. Finalmente, en [Lafortune94], se analizaba el uso de estos modelos en el contexto de los algoritmos de Monte-Carlo, y se proponía otra BRDF, también basada en la fórmula de sombreado de Phong. A este modelo se le llama en dicho artículo modelo de *Phong modificado*. Usando la notación de este texto, se puede expresar como:

$$f_{rp}(x, w_o, w_i) = \rho_p(x) \frac{n+2}{2\pi} \cos^n(w_o, \text{refl}_x(w_i)) \quad (1.23)$$

donde $n \geq 0$ es un valor real llamado el *exponente de Phong*. Suponiendo una radiancia incidente L_i desde una única dirección w_i sobre x , el valor máximo de L_r se produce en la dirección reflejada de w_i , y decae al alejarnos de dicha dirección. El pico de radiancia que se produce en la dirección reflejada se hace mas estrecho según el valor de n crece. Para $n = 0$ se obtiene la BRDF difusa, introducida anteriormente, mientras que si n crece indefinidamente en el límite esta BRDF se aproxima a la función de reflexión especular perfecta. Este el motivo de ver esta BRDF como algo intermedio entre las dos opciones anteriores, y por este motivo se le llama reflexión difuso-especular.

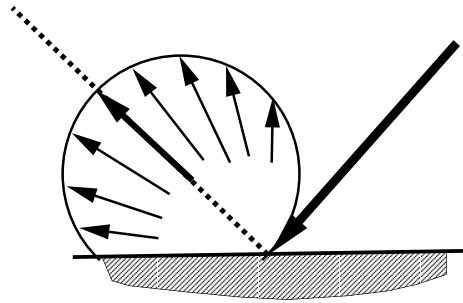


Figura 1.5: Forma de la BRDF difuso-especular para $n = 1$

En la figura 1.5 se puede apreciar un diagrama en coordenadas polares de la forma de la BRDF para el caso de $n = 1$. Para valores mas grandes de n , la BRDF toma aproximadamente la forma que se aprecia en la figura 1.6 La función ρ_p es

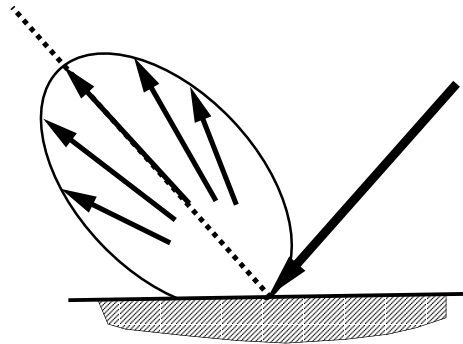


Figura 1.6: Forma de la BRDF difuso-especular para $n > 1$

una cota de la reflectancia direccional-semiesférica, esto es $\rho(x, w_i) \leq \rho_p(x)$. Por tanto, la conservación de la energía se cumple siempre que $\rho_p(x) \leq 1$. Esto es fácil de comprobar, ya que la integral del termino del coseno en toda la semiesfera (respecto de n_x) es menor que en la esfera completa, y esta a su vez es $2\pi/(n+2)$. Es difícil dar una expresión analítica exacta de ρ para esta BRDF, ya que los límites

de integración son complicados [Lafortune94]. Esto se debe a que el término del coseno puede ser no nulo para direcciones w tales que $\cos(n_x, w) < 0$, y dichas direcciones no contribuyen a la integral. Por otro lado, la reciprocidad de esta BRDF es fácil de comprobar, puesto que se cumple $\cos(w_o, refl_x(w_i)) = \cos(refl_x(w_o), w_i)$

Comportamiento reflectivo mixto.

Muchas veces, las superficies no muestran un comportamiento que se ajuste exactamente a alguno de estos modelos. Un ejemplo sería el de las superficies que reflejan la luz simultáneamente en más de una de estas formas. Este es el caso de una superficie difusa recubierta de una capa semitransparente. La capa superior puede reflejar parte de la luz de forma especular perfecta. Otra porción de la luz puede penetrar hasta el material difuso, reflejarse y volver a salir atravesando la capa transparente.

Si asumimos que, para todos los puntos x de un entorno, se cumple que

$$\rho_s(x) + \rho_d(x) + \rho_p(x) \leq 1 \quad (1.24)$$

entonces las BRDF definida por $f_r = f_{rs} + f_{rd} + f_{rp}$ cumple la conservación de la energía, como se puede deducir fácilmente de sus formulaciones.

Finalmente, el peso relativo de cada una de las componentes del modelo anterior puede incluso depender de la dirección de salida w_o . Un ejemplo sería el de algunos suelos de mármol pulido. Cuando son observados desde arriba, verticalmente, hacia abajo ($\cos(w_o, n_x)$ cercano a 1), se perciben como superficies difusas, pero sin embargo, cuando la dirección de observación es casi horizontal ($\cos(w_o, n_x)$ cercano a 0), entonces parecen especulares [Lafortune97].

1.3 Transporte global de la luz.

La sección anterior se llamó *transporte local de luz* por que la ecuación (1.12) modela la forma en que la luz se refleja localmente en un punto x . Esto caracteriza la función L_r sobre un punto en términos de L_i en el mismo punto. Sin embargo, nuestro objetivo es caracterizar L_r en todos los puntos del entorno, en función exclusivamente de L_e . Por esto se le llama transporte *global*. La ecuación que introducimos aquí es parecida a la ecuación de Kajiya, solo que en una forma modificada, de manera que su estructura es exactamente la de una ecuación integral de Fredholm de segunda clase. A su vez, la ecuación de Kajiya esta basada en la ecuación de transporte de Boltzman [Kajiya86, Arvo90].

1.3.1 Ecuaciones integrales para radiancia.

El primer paso para llegar a la ecuación integral completa es considerar el hecho de que la radiancia incidente en un punto desde una dirección se debe a la radiancia saliente de otro punto. Como la radiancia se conserva a lo largo de líneas rectas, obtenemos que $L_i(x, w)$ es igual a la $L(x', -w)$, donde x' es el primer punto visible desde x en la dirección w . Nótese que siempre existe x' debido a que el

entorno es cerrado. Esto puede escribirse formalmente definiendo una función p que proporciona, para cada punto x y dirección w el primer punto visible en la escena desde x en la dirección w [Arvo95a]. Así que $x' = p(x, w)$ y la igualdad anterior se puede expresar como $L_i(x, w) = L(p(x, w), -w)$. Usando esta igualdad podemos reescribir la ecuación (1.12), como sigue:

$$L_r(x, w_o) = \int_O f_r(x, w_o, w_i) L(p(x, w_i), -w_i) d\sigma_x(w_i) \quad (1.25)$$

la anterior ecuación se puede escribir en forma de operador como $L_r = \mathcal{T}L$, donde \mathcal{T} es un operador de transporte integral. Nótese que $L = L_e + L_r$. Expandiendo L_r por la anterior definición obtenemos

$$L = L_e + \mathcal{T}L \quad (1.26)$$

Esta es una ecuación integral. Para una BRDF f_r y una función L_e cualquiera, existe una única función L que satisface dicha ecuación, asumiendo que f_r cumple la conservación de la energía.

Con el fin de introducir los métodos computacionales que sirven para resolver la ecuación (1.25), es conveniente escribirla como una ecuación integral de segundo orden, debido a que estas ecuaciones han sido estudiadas previamente en la literatura. Estas ecuaciones tienen la siguiente forma:

$$f(x) = g(x) + \int_D k(x, y) f(y) dm(y) \quad (1.27)$$

donde D es un dominio, m es una medida definida en D , y g es una función integrable bajo la medida m en D . Si k cumple ciertas propiedades, entonces existe una única función f que cumple la ecuación integral.

Podemos ver en la ecuación (1.27) que la función desconocida f tiene como dominio el dominio de integración D , mientras que en (1.25), L (la función desconocida) está definida en $S \times O$ pero la integración se hace en O . Otra diferencia proviene de la ocurrencia de p en los argumentos de L en (1.25) mientras que en (1.27) el argumento de L es simplemente y .

Sin embargo, es posible escribir una ecuación equivalente a (1.25) pero con la misma estructura que (1.27). Esto se puede hacer en dos pasos: primero, se elimina la función p de los argumentos de L en (1.25), mediante un cambio de variable de integración. Finalmente, se hace que el dominio de la integración coincida con el dominio de L , usando un nuevo cambio.

Respecto al primer paso, haremos una integración en el dominio de los puntos de S en lugar de las direcciones sobre w_i . En la figura 1.7 vemos dos puntos x e y sobre las superficies de los objetos, donde $y = p(x, w_i)$. Para dos puntos cualesquiera a y b de las superficies, definamos w_{ab} como el vector unitario de dirección que va desde a hacia b , esto es $w_{ab} = (b - a)/|b - a|$. Por lo tanto tenemos que $w_{xy} = w_i$ y $w_{yx} = -w_{xy}$. Dado un conjunto de O' vectores de dirección salientes de x , vemos que su medida $\sigma(O')$ es el área del círculo sombreado sobre la semiesfera. El conjunto de direcciones cubre una región (sombreada en la figura) S' alrededor del

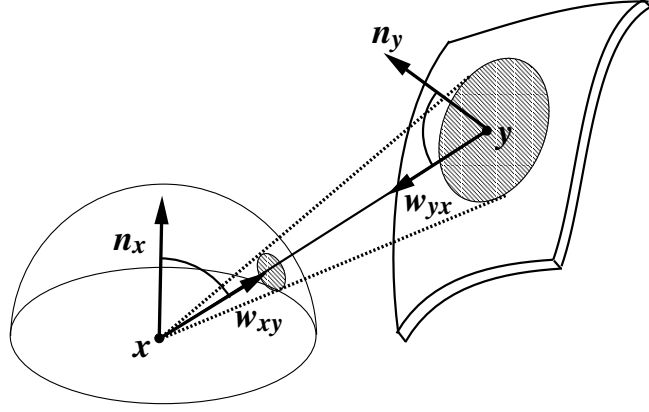


Figura 1.7: La relación entre medidas para el cambio de variable.

punto y cuyo área es $A(S')$. En el límite cuando $\sigma(O')$ tiende a cero, obtenemos la siguiente relación entre los diferenciales de las respectivas medidas

$$d\sigma(x) = \frac{\cos(n_y, w_{yx})}{|x - y|^2} dA(y) \quad (1.28)$$

A partir de aquí podemos obtener, a su vez, la relación entre la medida de ángulo proyectado σ_x y la medida de área $A(y)$:

$$d\sigma_x(w_i) = H(x, y) dA(y) \quad (1.29)$$

donde $y = p(x, w)$ y H es una función definida como sigue:

$$H(x, y) = \frac{\cos(n_x, w_{xy}) \cos(n_y, w_{yx})}{|x - y|^2} \quad (1.30)$$

Ya que la radiancia se conserva en las líneas rectas en el vacío entre los objetos, también se deduce que:

$$L_i(x, w_{xy}) = L(y, w_{yx}) \quad (1.31)$$

por lo tanto podemos cambiar la variable de integración en (1.12), sustituyendo w_i por y , y obtenemos:

$$L_r(x, w_o) = \int_{S_x} f_r(x, w_o, w_{xy}) H(x, y) L(y, w_{yx}) dA(y) \quad (1.32)$$

donde $S_x \subseteq S$ es el subconjunto de los puntos de S visibles desde x . Esta integral puede extenderse a todo S , usando un término geométrico adicional $G(x, y)$ definido por:

$$G(x, y) = H(x, y) V(x, y) \quad (1.33)$$

donde $V(x, y)$ es 1 cuando x e y son mutuamente visibles, y 0 si hay algún objeto entre ellos. Usando G obtenemos

$$L_r(x, w_o) = \int_S f_r(x, w_o, w_{xy}) G(x, y) L(y, w_{yx}) dA(y) \quad (1.34)$$

Esta integral se realiza sobre S , en lugar de O . En la figura 1.8 se puede observar como ahora la radiancia reflejada L_r no depende ya de las radiancias incidentes en el punto (como se veía en la figura 1.3) si no de las radiancias salientes de otros puntos.

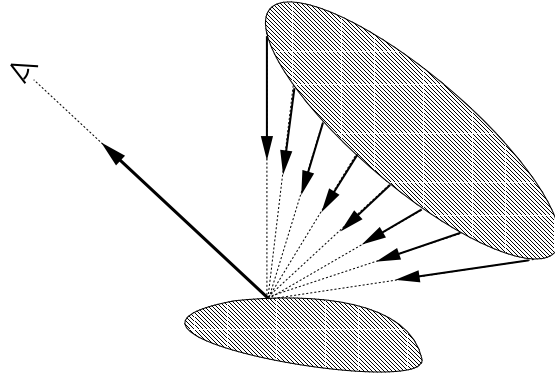


Figura 1.8: La radiancia reflejada como integral de radiancia.

Aún tenemos que transformar la integral para que el dominio sea $S \times O$. Este segundo paso se realiza usando las propiedades de la función δ . Dichas propiedades permiten establecer la siguiente igualdad:

$$L(y, w_{yx}) = \int_O \delta(w - w_{yx}) L(y, w) d\sigma(w) \quad (1.35)$$

Donde δ es la función delta de Dirac con respecto a la medida de ángulo sólido σ . Podemos sustituir $L(y, w_{yx})$ por la expresión anterior en (1.34), obteniéndose

$$\begin{aligned} L_r(x, w_o) &= \int_S f_r(x, w_o, w_{xy}) G(x, y) \left[\int_O \delta(w - w_{yx}) L(y, w) d\sigma(w) \right] dA(y) \\ &= \int_{S \times O} f_r(x, w_o, w_{xy}) \frac{\cos(n_x, w_{xy})}{|x - y|^2} V(x, y) \delta(w - w_{yx}) L(y, w) d\mu(y, w) \end{aligned}$$

para simplificar la notación, podemos aislar todos los términos que dependen de x e y en una función K definida como:

$$K(x, w_o, y, w) = f_r(x, w_o, w_{xy}) \frac{\cos(n_x, w_{xy})}{|x - y|^2} V(x, y) \delta(w - w_{yx}) \quad (1.36)$$

en adelante, usaremos símbolos en negrita para significar elementos del dominio de integración $D = S \times O$. Así, definimos $\mathbf{r} = (x, w_o)$, $\mathbf{s} = (y, w)$, y $K(\mathbf{r}, \mathbf{s}) = K(x, w_o, y, w)$. A los elementos de D les llamaremos *rayos*. Todo esto lleva a una forma simplificada de la ecuación anterior:

$$L_r(\mathbf{r}) = \int_D K(\mathbf{r}, \mathbf{s}) L(\mathbf{s}) d\mu(\mathbf{s}) \quad (1.37)$$

En la figura 1.9 podemos observar como la integración no se realiza ya sobre los puntos del entorno (como se veía en la figura 1.8) si no sobre todos los rayos. Solo

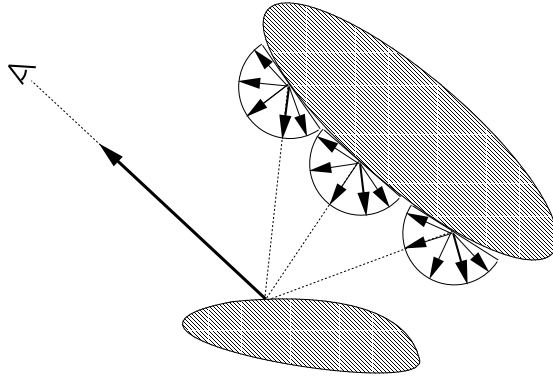


Figura 1.9: La radiancia como integral en el espacio de los rayos.

aquellos rayos que apuntan hacia el punto x contribuyen a dicha integral, gracias a la presencia de la función delta, que cancela el efecto de los demás, y reduce la dimensión de la integración.

Esta última ecuación (1.37) es equivalente a las formulaciones previas, esto es, a las ecuaciones (1.25), (1.26) y (1.34). Sin embargo, la ecuación (1.37) tiene la misma forma que la ecuación genérica (1.27), donde L cumple el papel de f , L_e el de g , y K el de k . En la ecuación (1.26) introdujimos el operador de transporte \mathcal{T} . Usando (1.37), podemos definir la acción del operador \mathcal{T} sobre la función L como sigue:

$$L_r(\mathbf{r}) = (\mathcal{T}L)(\mathbf{r}) = \int_D K(\mathbf{r}, \mathbf{s}) L(\mathbf{s}) d\mu(\mathbf{s}) \quad (1.38)$$

La función K se llama el *núcleo* del operador de transporte \mathcal{T} . La ecuación (1.37) puede también expresarse en forma diferencial. Supongamos que $S' \subseteq S$. Podemos medir la cantidad de radiancia en \mathbf{r} debido a radiancia de rayos en S' , llamando a esto $L_r(\mathbf{r}, S')$. Esta medida se puede diferenciar. En concreto, se puede demostrar que:

$$K(\mathbf{r}, \mathbf{s}) = \frac{dL_r(\mathbf{r}, \mathbf{s})}{L(\mathbf{s}) d\mu(\mathbf{s})}$$

de esta forma podemos establecer que $K(\mathbf{r}, \mathbf{s})$ es el diferencial de radiancia en el rayo \mathbf{r} debido a una unidad de radiancia en el rayo \mathbf{s} .

1.3.2 La ecuación de radiosidad

En el caso de que un entorno incluya únicamente superficies con BRDF difusa, tal y como está definida por la ecuación (1.21), y emisión difusa, la ecuación integral que gobierna el transporte de radiancia se simplifica, resultando en la clásica ecuación de radiosidad. Por emisión difusa entendemos una función L_e tal que los valores $L_e(x, w)$ son independientes de w . En estas condiciones, se puede demostrar que también L y L_r son independientes de w . Para ello se pueden usar las ecuaciones (1.21) y (1.12)

$$\begin{aligned} L_r(x, w) &= \int_O \frac{\rho_d(x)}{\pi} L_i(x, w_i) d\sigma_x(w_i) \\ &= \frac{\rho_d(x)}{\pi} E(x) \end{aligned}$$

así que en estas condiciones podemos eliminar la w de los argumentos de L_e, L_r y L (al ser esta última la suma de las dos anteriores). Escribiremos, por tanto, $L_e(x), L_r(x)$ y $L(x)$. Usualmente, a estas funciones se les llama funciones de *radiosidad*. Tenemos por tanto la radiosidad emitida $B_e = L_e$, la radiosidad reflejada $B_r = L_r$ y simplemente radiosidad para la suma $B = L$. La ecuación (1.34) se puede reescribir usando las definiciones anteriores:

$$B_r(x) = \frac{\rho_d(x)}{\pi} \int_S G(x, y) B(y) dA(y) \quad (1.39)$$

Sabemos que $B(x) = B_e(x) + B_r(x)$, así que podemos expandir B_r y G , obteniéndose

$$B(x) = B_e(x) + \rho_d(x) \int_S \frac{\cos(n_x, w_{xy}) \cos(n_y, w_{yx})}{\pi |x - y|^2} B(y) dA(y) \quad (1.40)$$

Está es la *ecuación de radiosidad*, que puede verse como una simplificación de (1.34).

1.3.3 Operadores integrales y soluciones a ecuaciones integrales.

Consideremos cualquier ecuación integral con la forma de (1.27). Dicha ecuación puede reescribirse más compactamente usando operadores. Para ello definiremos el operador \mathcal{A} como el operador integral inducido por el función núcleo k

$$(\mathcal{A}f)(x) = \int_D k(x, y) f(y) dy \quad (1.41)$$

con esta definición, la ecuación (1.27) puede reescribirse como:

$$f = g + \mathcal{A}f \quad (1.42)$$

Si g y k son conocidos (y por lo tanto también \mathcal{A}), podemos preguntarnos si existe una única función f cumpliendo la ecuación anterior. Con el objetivo de responder a esto, revisamos aquí brevemente los conceptos de norma de una función y un operador. Una explicación más detallada de la utilidad de estos conceptos en Iluminación Global puede encontrarse en [Arvo95a].

Operadores y normas

La L_p -norma de una función f , bajo una medida m , se escribe como $|f|_p$ y es un valor escalar definido como:

$$|f|_p = \left[\int f^p(x) dm(x) \right]^{1/p} \quad (1.43)$$

cada norma definida para las funciones induce una norma en los operadores que actúan sobre esas funciones. Para un operador \mathcal{A} , su L_p -norma se define como :

$$|\mathcal{A}|_p = \inf \{a : |\mathcal{A}f|_p \leq a|f|_p \quad \forall f\} \quad (1.44)$$

de la anterior definición se puede derivar la siguiente propiedad, que se cumple para todas las funciones f con $|f|_p$ finito:

$$|\mathcal{A}f|_p \leq |\mathcal{A}|_p |f|_p \quad (1.45)$$

esta es una propiedad interesante que usaremos más adelante.

Los operadores pueden ser sumados y compuestos entre ellos. Para cualesquiera dos operadores \mathcal{A} y \mathcal{B} , y cualquier función f , definimos $(\mathcal{A} + \mathcal{B})f = \mathcal{A}f + \mathcal{B}f$, y $(\mathcal{A}\mathcal{B})f = \mathcal{A}(\mathcal{B}f)$. Es fácil probar que:

$$|\mathcal{A}\mathcal{B}|_p \leq |\mathcal{A}|_p |\mathcal{B}|_p \quad (1.46)$$

$$|\mathcal{A} + \mathcal{B}|_p \leq |\mathcal{A}|_p + |\mathcal{B}|_p \quad (1.47)$$

Un operador *lineal* es uno que cumple $\mathcal{A}(f + g) = \mathcal{A}f + \mathcal{B}g$ para todos los f y g . Si \mathcal{A} es un operador lineal, se cumple la propiedad distributiva, esto es, $\mathcal{A}(\mathcal{B} + \mathcal{C}) = \mathcal{A}\mathcal{B} + \mathcal{A}\mathcal{C}$, para todos los operadores \mathcal{B} y \mathcal{C} . Un operador especial es el operador identidad, notado por \mathcal{I} . Este es el único que aplica toda función en si misma, esto es, para todas las f tenemos que $\mathcal{I}f = f$. La norma del operador identidad es siempre 1. También es posible definir el operador \mathcal{A}^n , para cada operador base \mathcal{A} y número natural n . Esto se hace iterativamente: $\mathcal{A}^0 = \mathcal{I}$, y $\mathcal{A}^{n+1} = \mathcal{A}\mathcal{A}^n$. Usando (1.46) derivamos lo siguiente:

$$|\mathcal{A}^n|_p \leq |\mathcal{A}|_p^n \quad (1.48)$$

Finalmente, definimos el operador $\mathcal{A}^{(n)}$ como $\sum_{i=0}^n \mathcal{A}^i$. Cuando \mathcal{A} es lineal se cumple la siguiente propiedad:

$$\mathcal{A}^{(n+1)} = \mathcal{I} + \mathcal{A}\mathcal{A}^{(n)} \quad (1.49)$$

Existencia de la solución.

La existencia y unicidad de la función f que satisface la ecuación (1.42), y tal que $\int_D f^p(x) dm(x)$ existe, esta garantizada cuando \mathcal{A} es lineal y $|\mathcal{A}|_p < 1$. Esto se puede demostrar examinando la serie de operadores $\mathcal{A}^{(n)}$. Por resultados previos vemos que:

$$|\mathcal{A}^{(n)}|_p \leq \sum_{i=0}^n |\mathcal{A}|_p^i \quad (1.50)$$

cuando $|\mathcal{A}|_p < 1$, la serie de normas $|\mathcal{A}^{(n)}|_p$ converge, y

$$\lim_{n \rightarrow \infty} |\mathcal{A}^{(n)}|_p \leq \frac{1}{1 - |\mathcal{A}|_p} \quad (1.51)$$

por lo tanto, para cualquier función h , la serie de funciones $\mathcal{A}^{(n)}h$ converge a una función h' tal que $|h'|_p \leq 1/(1 - |\mathcal{A}|_p)$. Podemos definir el operador \mathcal{A}^+ como el que aplica h en h' . La función h' es el límite de una serie funcional

$$h' = \mathcal{A}^+h = \lim_{n \rightarrow \infty} \mathcal{A}^{(n)}h$$

Si aplicamos ambos lados de (1.49) a cualquier función y después tomamos límites cuando $n \rightarrow \infty$, podemos derivar una interesante propiedad de \mathcal{A}^+

$$\mathcal{A}^+ = \mathcal{I} + \mathcal{A}\mathcal{A}^+ \quad (1.52)$$

si aplicamos los dos lados de esta igualdad a la función g en (1.42), obtenemos

$$(\mathcal{A}^+g) = g + \mathcal{A}(\mathcal{A}^+g)$$

y por tanto se deduce que \mathcal{A}^+g cumple la ecuación (1.42). De esto derivamos que su solución existe y puede ser expresada como $f = \mathcal{A}^+g$. La norma de \mathcal{A}^+ cumple

$$|\mathcal{A}^+|_p \leq \frac{1}{1 - |\mathcal{A}|_p}$$

El operador \mathcal{A}^+ se llama el *operador resolvente* de \mathcal{A} , como se describe en [Arvo95a].

La norma de los operadores lineales integrales.

Supongamos que \mathcal{A} es un operador integral lineal, tal y como se define en la ecuación (1.41). Podemos buscar cual es la norma de esta clase de operadores (o al menos una cota suya). Primero, definimos:

$$m_k = \operatorname{ess\,sup}_{y \in D} \left\{ \int_D k(x, y) \, dm(x) \right\} \quad (1.53)$$

donde *ess sup* significa *essential supremum*². Incluimos k como subíndice de m por que este valor depende de la función k . Ahora expandimos el valor $|\mathcal{A}h|_1$ para cualquier función h

$$\begin{aligned} |\mathcal{A}h|_1 &= \int_D (\mathcal{A}h)(x) \, dm(x) \\ &= \int_D \left[\int_D k(x, y) h(y) \, dm(y) \right] dm(x) \end{aligned}$$

²La noción de *essential supremum* (o *supremo esencial*) de un conjunto está brevemente descrita en [Arvo95a]. Es el supremo del conjunto (es decir, el ínfimo de las cotas superiores de los elementos del conjunto), ignorando subconjuntos del mismo con medida m nula.

$$\begin{aligned}
&= \int_D \left[\int_D k(x, y) dm(x) \right] h(y) dm(y) \\
&\leq \int_D m_k h(y) dm(y) \\
&= m_k |h|_1
\end{aligned}$$

la desigualdad anterior es cierta para cualquier h , así que podemos obtener una cota de $|A|_1$. Usando (1.44) se deduce

$$|A|_1 \leq m_k \quad (1.54)$$

1.3.4 La solución de la ecuación de transporte.

Una vez conocida las condiciones que aseguran la solución de cualquier ecuación integral, podemos preguntarnos acerca de la solución de la ecuación de transporte de radiancias, en la forma dada en (1.26). De resultados anteriores sabemos que la función L existe si \mathcal{T} es lineal y $|\mathcal{T}|_1 < 1$. La linealidad se cumple para todos los operadores integrales, como se puede deducir fácilmente (1.38). Con respecto a la norma, podemos usar (1.54). Si $m_K < 1$ entonces $|\mathcal{T}|_1 < 1$.

Así, debemos comprobar si es cierto que $m_K < 1$, donde K es la función definida en (1.36). Para hacer esto, expandimos $\int_D K(\mathbf{r}, \mathbf{s}) d\mu(\mathbf{r})$, donde $\mathbf{r} = (x, w_o)$ y $\mathbf{s} = (y, w)$ (nótese que y y w son valores fijos)

$$\begin{aligned}
&\int_D K(\mathbf{r}, \mathbf{s}) d\mu(\mathbf{r}) \\
&= \int_{S \times O} f_r(x, w_o, w_{xy}) \frac{\cos(n_x, w_{xy})}{|x - y|^2} V_{xy} \delta(w - w_{yx}) dA(x) d\sigma_x(w_o) \\
&= \int_S \left[\int_O f_r(x, w_o, w_{xy}) d\sigma_x(w_o) \right] \delta(w - w_{yx}) V_{xy} \frac{\cos(n_x, w_{xy})}{|x - y|^2} dA(x) \\
&= \int_O \left[\int_O f_r(x, w_o, -w') d\sigma_x(w_o) \right] \delta(w - w') d\sigma(w') \\
&= \int_O f_r(x, w_o, -w) d\sigma_x(w_o) \\
&= \rho(x, -w)
\end{aligned}$$

aquí hemos transformado una integración en S en una integración en O , usando la igualdad (1.29). Después de esto, hemos usado de nuevo las propiedades de la función δ . De lo anterior se deduce que

$$m_K = \sup_{(x, w) \in D} \{ \rho(x, w) \} \quad (1.55)$$

Se puede asegurar, por tanto, que si la BRDF cumple la conservación de la energía, entonces $m_K < 1$ y la ecuación (1.25) tiene solución.

1.4 Ecuaciones de transporte adjuntas.

La función f , solución de la ecuación (1.27) no puede ser expresada analíticamente para la mayoría de los casos. Usualmente el objetivo es conocer el producto escalar de f con alguna otra función h , que se escribe como $\langle f | h \rangle$. En esta sección introducimos el operador de transporte adjunto y mostramos como ese producto escalar puede obtenerse a partir de la solución de dos ecuaciones integrales duales.

1.4.1 Productos escalares y operadores adjuntos.

El producto escalar de dos funciones f y h , es un valor real³ definido como

$$\langle f | h \rangle = \int_D f(x) h(x) dm(x) \quad (1.56)$$

como puede observarse, el producto escalar depende de una función de medida usada en la integración. Se puede hacer referencia a dicha medida usando su nombre como subíndice, cuando pudiese ocurrir alguna confusión. A partir de su definición, es fácil deducir las siguientes propiedades del producto escalar, que se cumplen para todas las funciones integrables f, h y h' , y para todos los valores reales a .

$$\langle f | h \rangle = \langle h | f \rangle \quad (1.57)$$

$$\langle f | h + h' \rangle = \langle f | h \rangle + \langle f | h' \rangle \quad (1.58)$$

$$\langle af | h \rangle = a \langle f | h \rangle \quad (1.59)$$

El operador adjunto de un operador \mathcal{A} se escribe como \mathcal{A}^* , y se define como el operador que cumple lo siguiente

$$\langle \mathcal{A}f | h \rangle = \langle f | \mathcal{A}^*h \rangle \quad (1.60)$$

para todas las funciones integrables f y h . Téngase en cuenta que no todos los operadores tienen un adjunto. En el caso de que \mathcal{A} sea un operador integral con un núcleo k , tal y como se define en (1.41), entonces existe \mathcal{A}^* , y está dado por la siguiente ecuación:

$$(\mathcal{A}^*h)(x) = \int_D k(x, y) h(y) dm(y) \quad (1.61)$$

es fácil mostrar que \mathcal{A}^* cumple (1.60), expandiendo $\langle \mathcal{A}f | h \rangle$

$$\begin{aligned} \langle \mathcal{A}f | h \rangle &= \int_D (\mathcal{A}f)(x) h(x) dm(x) \\ &= \int_D \left[\int_D k(x, y) f(y) dm(y) \right] h(x) dm(x) \\ &= \int_{D \times D} k(x, y) f(y) h(x) dm(y) dm(x) \end{aligned}$$

³Suponemos que f y h son funciones integrables en el dominio D con la medida m .

$$\begin{aligned}
&= \int_D \left[\int_D k(x, y) h(x) dm(x) \right] f(y) dm(y) \\
&= \int_D (\mathcal{A}^* h)(y) f(y) dm(y) \\
&= \langle f | \mathcal{A}^* h \rangle
\end{aligned}$$

Supongamos que \mathcal{A} y \mathcal{B} son dos operadores arbitrarios con adjunto, y f y h dos funciones integrables. Es fácil probar que $\langle (\mathcal{A} + \mathcal{B})f | h \rangle = \langle f | (\mathcal{A}^* + \mathcal{B}^*)h \rangle$ y $\langle \mathcal{A}\mathcal{B}f | h \rangle = \langle f | \mathcal{B}^* \mathcal{A}^* h \rangle$. Usando esto se deducen las siguientes igualdades

$$(\mathcal{A} + \mathcal{B})^* = \mathcal{A}^* + \mathcal{B}^* \quad (1.62)$$

$$(\mathcal{A}\mathcal{B})^* = \mathcal{B}^* \mathcal{A}^* \quad (1.63)$$

$$(\mathcal{A}^n)^* = (\mathcal{A}^*)^n \quad (1.64)$$

$$(\mathcal{A}^{(n)})^* = (\mathcal{A}^*)^{(n)} \quad (1.65)$$

1.4.2 La ecuación integral dual.

Como se ha dicho anteriormente, supongamos que estamos interesados en el valor $\langle f | h \rangle$, donde f es la solución a la ecuación (1.42) y h es una función arbitraria. Sabemos que el operador lineal integral \mathcal{A} , definido en (1.41), tiene un adjunto \mathcal{A}^* . Este adjunto es también un operador lineal, como se ha demostrado en la sección anterior. Podemos asumir que $\|\mathcal{A}^*\|_1 < 1$ [Arvo95a]. Así, existe el operador resolvente de \mathcal{A}^* , que se escribe como $(\mathcal{A}^*)^+$. En estas condiciones la siguiente ecuación integral

$$l = h + \mathcal{A}^* l \quad (1.66)$$

tiene una solución, esto es, existe una función integrable l que la cumple, y que puede escribirse como $l = (\mathcal{A}^*)^+ h$. Nótese que g y h de las ecuaciones (1.42) y (1.66) son funciones integrables arbitrarias. Usando la propiedad (1.65), podemos obtener las siguientes igualdades

$$\begin{aligned}
\langle \mathcal{A}^+ g | h \rangle &= \left\langle \lim_{n \rightarrow \infty} \mathcal{A}^{(n)} g | h \right\rangle \\
&= \lim_{n \rightarrow \infty} \langle \mathcal{A}^{(n)} g | h \rangle \\
&= \lim_{n \rightarrow \infty} \langle g | (\mathcal{A}^{(n)})^* h \rangle \\
&= \lim_{n \rightarrow \infty} \langle g | (\mathcal{A}^*)^{(n)} h \rangle \\
&= \left\langle g | \lim_{n \rightarrow \infty} (\mathcal{A}^*)^{(n)} h \right\rangle \\
&= \langle g | (\mathcal{A}^*)^+ h \rangle
\end{aligned}$$

así que podemos establecer que $(\mathcal{A}^*)^+$ es el adjunto de \mathcal{A}^+ , esto es

$$(\mathcal{A}^+)^* = (\mathcal{A}^*)^+ \quad (1.67)$$

y obtenemos que

$$\langle f | h \rangle = \langle l | g \rangle \quad (1.68)$$

así que el cálculo de $\langle f | h \rangle$ es equivalente a el cálculo de $\langle l | g \rangle$. Esto tiene bastantes aplicaciones en Iluminación Global.

1.4.3 Ecuaciones de transporte de importancias.

Una vez que se han introducido las nociones de producto escalar y operador adjunto, podemos aplicarlas a la función de radiancia y a la ecuación integral que la define. Supongamos que W_e es cualquier función tal que deseamos calcular el valor $\langle L | W_e \rangle$. La contribución de $L(\mathbf{r})$ a este producto escalar está por tanto ponderada por $W_e(\mathbf{r})$. Debido a esto, a la función W_e se le llama *importancia emitida* o *potencial emitido*. El uso de la palabra "emitida" en los anteriores nombres (y del subíndice e) se debe a que W_e juega un papel similar al de L_e en una ecuación integral que veremos más adelante.

Por ejemplo, podemos definir $W_e(\mathbf{r})$ como una función distinta de cero solo para rayos \mathbf{r} que apunten al punto de vista y atraviesen el área de un pixel en el plano de visión (para los cuales W_e vale 1). De esta forma, el valor $\langle L | W_e \rangle$ es la radiancia media de los rayos que atraviesan un pixel en dirección al observador. Otro ejemplo sería el caso de una función W_e tal que solo valga 1 para rayos originados en un trozo de las superficies. En este caso, el valor $\langle L | W_e \rangle$ es igual a la potencia radiante total que abandona dicho trozo.

Sabemos que si una BRDF cumple la conservación de la energía, entonces existe \mathcal{T}^+ y la función de radiancia L . Es fácil probar que si a estas condiciones se le añade la simetría de dicha BRDF, entonces $|\mathcal{T}^*|_1 < 1$. Esto implica que la siguiente ecuación:

$$W = W_e + \mathcal{T}^* W \quad (1.69)$$

tiene solución, esto es, existe el operador $(\mathcal{T}^*)^+$, que es el adjunto de \mathcal{T}^+ , y también existe la función integrable W que se obtiene como $W = (\mathcal{T}^*)^+ W_e$. El valor $\langle L | W_e \rangle$ puede reescribirse como

$$\langle L | W_e \rangle = \langle \mathcal{T}^+ L_e | W_e \rangle = \langle L_e | (\mathcal{T}^*)^+ W_e \rangle = \langle L_e | W \rangle \quad (1.70)$$

La ecuación (1.69) es la ecuación adjunta a (1.26). Se le llama usualmente la ecuación de transporte de importancias. Como puede observarse, la estructura de ambas ecuaciones es la misma. Vemos que W_e hace el papel de L_e , y W el de L . La importancia puede verse como una cantidad transportada por el operador \mathcal{T}^* , al igual que la radiancia es transportada por \mathcal{T} . La función W se llama simplemente importancia o potencial. El valor $W(\mathbf{r})$ representa la fracción de $L_e(\mathbf{r})$ que contribuye al producto escalar (1.70).

1.5 Métodos computacionales.

Una vez que se han explicado las ecuaciones que definen la radiancia y la importancia, en esta sección introducimos brevemente las bases de los elementos que se usan

para obtener aproximaciones discretas a dichas funciones. Estos métodos pueden clasificarse en dos familias: Métodos de Monte-Carlo y Métodos de elementos finitos o de proyección. Ambas clases de técnicas están apoyadas en principios diferentes, que se introducen brevemente aquí.

1.5.1 Bases y operadores de proyección.

Las funciones integrables en un dominio cualquiera pueden ser sumadas entre ellas y pueden ser multiplicadas por valores reales, dando lugar a funciones que son también integrables. Esto da al conjunto la estructura de un espacio vectorial. Consideremos cualquier conjunto finito $\mathbf{B} = \{b_1, b_2, \dots, b_n\}$ formado por n funciones integrables en el dominio D . Supondremos que este conjunto es ortonormal, esto es $\langle b_i | b_j \rangle = \delta_{ij}$ para todos los $i, j \leq n$, donde δ_{ij} es 1 cuando $i = j$ y 0 en otro caso. El conjunto \mathbf{B} puede verse como la base de un subespacio vectorial n -dimensional $V_{\mathbf{B}}$. Para cualquier función h en $V_{\mathbf{B}}$, existe un conjunto de n valores reales $\{c_1, \dots, c_n\}$ tal que

$$h = \sum_{i=1}^n c_i b_i \quad (1.71)$$

usualmente se dice $V_{\mathbf{B}}$ es el espacio vectorial expandido por \mathbf{B} . También diremos que h es una *combinación lineal* de las funciones b_i .

Para cualquier función integrable f podemos encontrar la función f' más cercana a f tal que f' este en $V_{\mathbf{B}}$. Aquí el término *más cercana* se usa en el sentido inducido por la L_1 -norma. Esto es, el valor $|f' - f|_1$ es más pequeño que cualquier valor $|g - f|_1$ donde $g \in V_{\mathbf{B}}$. En el caso de que \mathbf{B} sea ortonormal, es posible obtener f' directamente de f y \mathbf{B} , ya que

$$f' = \sum_{i=1}^n \langle f | b_i \rangle b_i \quad (1.72)$$

La función f' se llama la *proyección*⁴ de f en $V_{\mathbf{B}}$. Podemos considerar el operador que toma una función cualquiera y produce su proyección en \mathbf{B} . Escribiremos este operador como $\mathcal{P}_{\mathbf{B}}$, y así tenemos que $f' = \mathcal{P}_{\mathbf{B}}f$. Este operador está definido por

$$(\mathcal{P}_{\mathbf{B}}f)(x) = \sum_{i=1}^n \langle f | b_i \rangle b_i(x) \quad (1.73)$$

Es fácil comprobar que esta clase de operadores son siempre lineales, debido a la linealidad de los productos escalares, como se deduce de la igualdad (1.58). Si $f \in V_{\mathbf{B}}$ entonces $\mathcal{P}_{\mathbf{B}}f = f$. Esto implica que, para todos los $n > 0$, se cumple que $\mathcal{P}_{\mathbf{B}}^n = \mathcal{P}_{\mathbf{B}}$.

Los valores $\langle f | b_i \rangle$ se llaman las *coordenadas* de f con respecto a \mathbf{B} . Nótese que cualquier función in $V_{\mathbf{B}}$ está completamente determinada por sus coordenadas, y esto permite representar dichas funciones en la memoria de un ordenador con una cantidad finita de espacio en la misma.

⁴El término *proyección* se usa aquí ya que esta operación, cuando se realiza en el espacio vectorial de los vectores del espacio tridimensional, es la proyección perpendicular de un vector en una línea o un plano

1.5.2 Métodos de elementos finitos.

En esencia, los métodos de elementos finitos están basados en la conversión de la ecuación (1.42) en una ecuación matricial discreta. Esto se hace intentando resolver una ecuación parecida a (1.26) pero no exactamente igual. La ecuación aproximada que se resuelve es la siguiente:

$$f' = \mathcal{P}_B (g + \mathcal{A}f') \quad (1.74)$$

La función g debe darse como dato de entrada, luego debe ser representable en la memoria del ordenador. Lo más frecuente es escoger un g tal que se encuentre en V_B , y por lo tanto se almacena como un vector de coeficientes. Por eso tenemos que $g = \mathcal{P}_B g$. Como el operador de proyección es lineal, podemos reescribir (1.74) de la siguiente forma

$$f' = g + (\mathcal{P}_B \mathcal{A}) f' \quad (1.75)$$

esta ecuación es muy similar a (1.42) excepto por la aparición del operador $\mathcal{P}_B \mathcal{A}$ en lugar de \mathcal{A} . La función solución f' se encuentra en V_B , esto es, $\mathcal{P}_B f' = f'$. Este método para encontrar f' se llama método de *Galerkin* en la literatura sobre elementos finitos. Existen otras posibilidades para discretizar la ecuación continua, aunque esta es la más frecuentemente usada en Iluminación Global.

Podemos comprobar que la ecuación (1.75) nos lleva al método de Galerkin viendo que la proyección de la función de *residuo* es cero en todos los puntos. La función de residuo r es $(\mathcal{I} - \mathcal{A})f' - g$, y se considera como una medida del error en la solución. Vemos que si f' fuese exactamente f , el residuo sería cero. Pero lógicamente es imposible encontrar la solución exacta. Cada método de elementos finitos establece una condición sobre la función residuo. En el caso concreto del método Galerkin, se establece que la proyección de dicha función residuo debe ser la función cero en todos los puntos, Esto es, $\mathcal{P}_B r = 0$. Para comprobar esto, podemos expandir r usando la definición implícita de f' dada por (1.74)

$$\begin{aligned} \mathcal{P}_B r &= \mathcal{P}_B ((\mathcal{I} - \mathcal{A})f' - g) \\ &= (\mathcal{P}_B f' - \mathcal{P}_B \mathcal{A}f') - \mathcal{P}_B g \\ &= (f' - \mathcal{P}_B \mathcal{A}f') - g \\ &= g - g \\ &= 0 \end{aligned}$$

donde el cero de la ultima igualdad representa la función igual a cero en todos los puntos.

Ahora obtendremos la ecuación discreta equivalente a (1.75). Llamemos $\{f_1 \dots f_n\}$ a los coeficientes de f' con respecto a B . Entonces la función $(\mathcal{P}_B \mathcal{A})f'$ puede reescribirse como

$$\begin{aligned} (\mathcal{P}_B \mathcal{A}) f' &= (\mathcal{P}_B \mathcal{A}) \sum_{i=1}^n f_i b_i \\ &= \sum_{i=1}^n f_i \mathcal{P}_B (\mathcal{A} b_i) \end{aligned}$$

$$= \sum_{i=1}^n \left[f_i \sum_{j=1}^n \langle \mathcal{A}b_i | b_j \rangle b_j \right]$$

hemos usado la linealidad de $\mathcal{P}_{\mathbf{B}}$ y \mathcal{A} , y después la definición de $\mathcal{P}_{\mathbf{B}}$. La ecuación (1.75) puede expandirse

$$\sum_{i=1}^n f_i b_i = \sum_{j=1}^n g_j b_j + \sum_{k=1}^n f_k \sum_{l=1}^n c_{lk} b_l \quad (1.76)$$

donde c_{lk} es $\langle \mathcal{A}b_k | b_l \rangle$ y $\{g_1 \dots g_n\}$ son las coordenadas de g con respecto a \mathbf{B} . Esta última ecuación es una ecuación funcional que también se cumple cuando evaluamos ambos miembros en un punto arbitrario x del dominio. Reorganizando los términos obtenemos que, en cualquier punto x , se cumple que:

$$\sum_{i=1}^n \left[f_i - g_i - \sum_{j=1}^n c_{ij} f_j \right] b_i(x) = 0 \quad (1.77)$$

esta última ecuación puede cumplirse únicamente cuando todos los factores que multiplican a los $b_i(x)$ son cero, esto es, cuando

$$f_i = g_i + \sum_{j=1}^n c_{ij} f_j \quad (1.78)$$

para todos los $i \in \{1, \dots, n\}$. Esta ecuación se escribe en forma matricial:

$$\begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} = \begin{pmatrix} g_1 \\ \vdots \\ g_n \end{pmatrix} + \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & & \vdots \\ c_{n1} & \cdots & c_{nn} \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} \quad (1.79)$$

Definamos $F = (f_1, \dots, f_n)^T$, esto es, F es el vector columna de los coeficientes de f con respecto a \mathbf{B} . también definimos $G = (g_1, \dots, g_n)^T$ y

$$A = \begin{pmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & & \vdots \\ c_{n1} & \cdots & c_{nn} \end{pmatrix} \quad (1.80)$$

de esta forma la ecuación (1.79) puede escribirse más compactamente como

$$F = G + A F \quad (1.81)$$

la ecuación anterior es equivalente a la ecuación continua original excepto que no involucra funciones sino los coeficientes de sus proyecciones en \mathbf{B}

1.5.3 Métodos de Monte-Carlo.

Una alternativa a los métodos de elementos finitos es usar Cadenas de Markov para resolver (1.42). En lugar de discretizar la ecuación, lo que se obtiene es una aproximación estocástica del producto escalar de la función incógnita f y alguna otra función h , esto es, el valor $\langle f | h \rangle$ [Rubinstein81].

Para desarrollar esto, es necesario antes introducir el concepto de Cadenas de Markov. Una cadena de Markov, (o un paseo aleatorio) c es una secuencia infinita de elementos del dominio de integración D de la integral que aparece en (1.27). Sea c una de estas cadenas, entonces $c = \{c_0, \dots, c_i, \dots\}$. Usualmente, a los elementos de la cadena se les llama estados.

Consideremos ahora el conjunto de todas las cadenas de Markov posibles. En los métodos de Monte-Carlo se realiza una selección aleatoria de estas cadenas. Para realizar dicha selección será necesario disponer de una *función de densidad de probabilidad* (en adelante una *pdf*), que llamaremos Q . Normalmente Q es separable, en el sentido de que existen otras dos funciones, que llamaremos p_0 y p tales que

$$Q(c) = p_0(c_0)p(c_0, c_1)p(c_1, c_2) \dots$$

donde $p_0(x)$ es la probabilidad de seleccionar x como primer punto de la cadena, y $p(y, z)$ es la probabilidad condicional de seleccionar z como punto siguiente cuando el anterior es y . Para que el método funcione correctamente es necesario imponer algunas condiciones sobre p_0 y p . Estas son ($\forall x, y \in D$)

$$\begin{aligned} \int_D p(y, z) dz &= 1 \\ \int_D p_0(z) dz &= 1 \\ h(x) \neq 0 &\rightarrow p_0(x) > 0 \\ k(x, y) \neq 0 &\rightarrow p(x, y) > 0 \end{aligned}$$

Donde h es una función cualquiera, integrable en D . Definamos ahora la función V , que se evalúa sobre Cadenas de Markov [Rubinstein81]

$$V(c) = \frac{h(c_0)}{p_0(c_0)} \sum_{i=0}^{\infty} U_i g(c_i)$$

donde:

$$U_i = \frac{K(c_0, c_1)}{p(c_0, c_1)} \dots \frac{K(c_{i-1}, c_i)}{p(c_{i-1}, c_i)}$$

Podemos ahora definir una variable aleatoria X , que a cada valor real $V(c)$ le asigna una densidad de probabilidad $Q(c)$. El hecho esencial acerca de X es que su valor promedio es exactamente el producto escalar que queremos calcular, esto es, $E(X) = \langle f | h \rangle$. Por tanto se puede muestrear la variable aleatoria para obtener estimadores de su valor promedio. Esto se realiza mediante el siguiente algoritmo:

1. Seleccionar la funciones de densidad de probabilidad p y P

2. Seleccionar aleatoriamente n cadenas $\{c^1, \dots, c^n\}$ con densidad de probabilidad Q , donde n es lo suficientemente grande.
3. Obtener una aproximación $x \in R$ del producto escalar mediante un estimador, esto es:

$$\langle h | f \rangle \approx a = \frac{1}{n} \sum_{i=1}^n V(c^i) \quad (1.82)$$

Nótese que hemos de tratar con cadenas de Markov de longitud infinita, lo cual no es abordable computacionalmente. Esto se puede resolver truncando las cadenas (lo cual provoca un error residual o sesgo más o menos grande) o bien extendiendo D mediante la inclusión de un nuevo elemento llamado *estado de absorción*. Llamando a a dicho estado, extendemos p_0, p, k y g para tenerlo en cuenta. para cualquier $x \in D$, tenemos que $p(x, a) > 0$, $p(a, x) = 0$, $p(a, a) = 1$, $p_0(a) = 0$, $k(a, x) = k(x, a) = 0$ y $g(a) = 0$. Con esto, todas las cadenas caerán en el estado de absorción después de un número finito de pasos, al tiempo que el método se mantiene no sesgado. Una vez que la cadena cae en dicho estado, no lo abandona, puesto que $p(a, a) = 1$.

Formulación adjunta.

En Iluminación Global muchas veces no se desea calcular simplemente $\langle h | f \rangle$, sino el conjunto de valores $\{\langle h_i | f \rangle\}$ para un conjunto de m funciones $\{h_1 \dots h_i \dots h_m\}$. Usando los resultados disponibles, tenemos al menos dos opciones para realizar esto último. La primera de ellas consiste en obtener, para cada i independientemente, el valor $\langle h_i | f \rangle$, como se establece en la sección anterior. Esto requiere la creación de mn cadenas en total.

Otra opción consiste en usar el adjunto de \mathcal{A}^+ , que es $(\mathcal{A}^*)^+$. Usando las propiedades del operador adjunto, sabemos que

$$\langle h_i | f \rangle = \langle h_i | \mathcal{A}^+ g \rangle = \langle g | (\mathcal{A}^*)^+ h_i \rangle \quad (1.83)$$

usando esta última forma obtenemos el siguiente procedimiento

1. Seleccionar las funciones de muestreo adecuadas p y P
2. Obtener aleatoriamente n cadenas (c^j) con la densidad de probabilidad anterior.
3. Para cada h_i , estimar el producto escalar como

$$\langle h_i | f \rangle = \langle g | \mathcal{A}_a^+ h_i \rangle \approx \frac{1}{n} \sum_{j=1}^n V_i(c^j) \quad (1.84)$$

donde

$$V_i(c) = \frac{g(c_0)}{p_0(c_0)} \sum_{j=0}^{\infty} U_j h_i(c_j)$$

y

$$U_j = \frac{K(c_1, c_0)}{p(c_0, c_1)} \dots \frac{K(c_j, c_{j-1})}{p(c_{j-1}, c_j)}$$

Esto requiere n cadenas, en lugar de mn . Nótese que los argumentos de k están invertidos con respecto a la formulación anterior. Esto se debe al uso del operador adjunto, en lugar del original (ver la ecuación 1.61).

1.6 Conclusiones.

Este capítulo incluye las definiciones de las funciones que caracterizan el flujo de energía electromagnética, visto desde la perspectiva del modelo de partículas de la radiación electromagnética. La definición de dichas funciones se hace usando relaciones entre funciones de medida.

Una vez introducidas las funciones, se presenta una derivación de la ecuación de radiancias en forma de una ecuación integral de segundo orden. Para ello damos la expresión adecuada del núcleo del operador de transporte K . Esta formulación presenta la ventaja de que las herramientas matemáticas existentes para dicha clase de ecuaciones se pueden usar fácilmente en el contexto de la Iluminación Global. Las formas anteriormente propuestas de dicha ecuación no presentan exactamente esta característica, bien por que el núcleo es una función de tres variables [Kajiya86], bien porque el operador de transporte aparece como la composición de otros dos [Arvo95].

Además de presentar la ecuación de esta forma, en este primer capítulo se detallan los fundamentos de los métodos de elementos finitos (fundamentalmente el método Galerkin) y de los métodos de Monte-Carlo. Esta introducción es fácil al usar la nueva forma propuesta para la ecuación de rendering.

Capítulo 2

Clasificación de métodos para Iluminación Global

2.1 Los conceptos de Imagen y Observador.

Una vez que hemos caracterizado L , sabemos que la radiancia en cada punto de una superficie esta perfectamente definida matemáticamente. Sin embargo, el objetivo de las técnicas de Iluminación Global no es sólo lograr dicha caracterización, sino además producir imágenes de los objetos. En esta sección intentamos formalizar el concepto de imagen y de observador, que son esenciales para establecer el problema de la Iluminación Global.

Una imagen I se define como una función cuyo dominio es D , tal que $I(\mathbf{r})$ es la cantidad de radiancia de \mathbf{r} percibida directamente por un observador dado. Con el fin de almacenar I en un ordenador, es necesario usar una proyección de dicha función en alguna base finita. Dicha proyección nos proporciona un conjunto de coeficientes que pueden usarse directa o indirectamente para iluminar los píxeles de una pantalla o imprimir hojas en una impresora.

La base usada para realizar la proyección anteriormente citada será un conjunto \mathbf{W} de n funciones ortonormales, con $\mathbf{W} = \{W_{e1}, \dots, W_{en}\}$, tales que el valor escalar $W_{ei}(\mathbf{r})$ es la fracción de radiancia que viaja a través de \mathbf{r} que contribuye al i -ésimo coeficiente de la imagen discreta. Si consideramos el operador de proyección $\mathcal{P}_{\mathbf{W}}$, entonces la función imagen puede escribirse como sigue

$$I = \mathcal{P}_{\mathbf{W}} L \quad (2.1)$$

Esto es, I pertenece a $V_{\mathbf{W}}$, y por tanto está determinada por un conjunto finito de valores (sus coeficientes respecto a \mathbf{W}). La anterior igualdad puede reescribirse usando la evaluación puntual de I en un rayo concreto

$$I(\mathbf{r}) = (\mathcal{P}_{\mathbf{W}} L)(\mathbf{r}) = \sum_{i=1}^n l_i W_{ei}(\mathbf{r})$$

donde $l_i = \langle W_{ei} \mid L \rangle$.

Idealmente (si usamos el modelo tradicional de cámara) cada función W_{ei} debe ser distinta de cero solo para rayos que pasen por el foco y el rectángulo (o círculo) ocupado en el plano de visión por el pixel número i . Desgraciadamente, solo es posible tomar un conjunto de muestras discretas de la radiancia que atraviesa cada pixel. En este caso, podemos definir \mathbf{W} como sigue:

$$W_{ei}(\mathbf{r}) = \sum_{j=1}^m w_{ij} \delta(\mathbf{r} - \mathbf{s}_{ij})$$

donde \mathbf{s}_{ij} es el rayo que pasa por el foco y por la posición de la j -ésima muestra en el pixel i -ésimo. El conjunto de pesos w_{ij} debe estar normalizado, en el sentido de que $\sum_{j=1}^m w_{ij}$ es 1 en todos los pixeles i . En el caso de otros modelos especiales de cámara, como por ejemplo una cámara que incluya lentes, se puede usar igualmente el modelo anterior de imagen, solo que las funciones W_{ei} cambian adecuadamente, y por tanto también cambia la posición de los rayos de muestra.

Con todos estos resultados, el problema de la Iluminación Global se presenta como sigue: dada una función de radiancia emitida L_e , un operador de transporte \mathcal{T} y un observador \mathbf{W} , calcular una representación finita de la función imagen I que el observador percibe. Dicha representación finita se obtiene como

$$I = \mathcal{P}_{\mathbf{W}} \mathcal{T}^+ L_e \quad (2.2)$$

como se ha dicho, I se corresponde con un conjunto finito de coeficientes respecto de \mathbf{W} . Estos coeficientes pueden interpretarse como intensidades de pixeles y por lo tanto se puede mostrar una representación de la imagen en diversos dispositivos hardware. Esta formulación es general, ya que incluye cualquier clase de observador o cámara y cualquier operador de transporte en un entorno arbitrario.

2.2 Soluciones

En este punto, el problema de la Iluminación Global ha quedado establecido, así como las herramientas necesarias para solucionarlo (en el capítulo anterior). Podemos entonces abordar una clasificación de los métodos usados para solucionar dicho problema:

Métodos locales

Esta familia de métodos no resuelve realmente el problema de la Iluminación Global, sino simplemente la iluminación local. Esto permite usar algoritmos simples y eficientes, con el coste de no calcular realmente Iluminación Global. La imagen calculada solo tiene en cuenta una reflexión en los objetos, y por tanto se usa un operador local \mathcal{T} en lugar del operador global \mathcal{T}^+ . Formalmente $I = \mathcal{P}_{\mathbf{W}} \mathcal{T} L_e$. Las técnicas de *ray-tracing*, *z-buffer* y los métodos de líneas de barrido caen en estas categorías.

Métodos de discretización o elementos finitos

Incluimos aquí todos los métodos basados en el uso de un operador proyectado

en lugar del original. El problema se convierte entonces en la solución de un sistema de ecuaciones discreto, de la cual se obtiene una aproximación discreta a L . A partir de esta aproximación se puede calcular una imagen proyectando L sobre \mathbf{W} . Formalmente podemos expresar estos dos pasos como sigue: (1) $L = (\mathcal{P}_{\mathbf{B}}\mathcal{T})^+ L_e$ y (2) $I = \mathcal{P}_{\mathbf{W}} L$. Nótese que aunque estos métodos se llaman métodos de discretización, es posible recurrir en algunos puntos a métodos de Monte-Carlo, como por ejemplo en el cálculo de los coeficientes del operador proyectado. (también llamados *factores de forma*).

Métodos estocásticos o de Monte-Carlo.

Estos métodos usan Cadenas de Markov para obtener la solución de la ecuación (2.2). Se obtiene una aproximación estocástica de la solución, bajo cierta varianza. Podemos, a su vez, dividirlos en dos subclases.

- **Path-Tracing desde el observador.** En este caso, para cada W_{ei} (independientemente) obtenemos $l_i = \langle W_{ei} | \mathcal{T}^+ L_e \rangle$, usando la formulación dada en el capítulo anterior.
- **Fotosimulación o Path-Tracing desde las fuentes de luz.** Obtenemos una solución simultánea para $l_i = \langle L_e | \mathcal{T}_a^+ b_i \rangle$ (para una base $\mathbf{B} = \{b_1 \dots b_n\}$) usando la segunda opción en la subsección 1.5.3. Formalmente, calculamos $L' = \mathcal{P}_{\mathbf{B}} \mathcal{T}^+ L_e$. Una vez que esto ha sido obtenido, se proyecta la función L usando un observador dado: $I = \mathcal{P}_{\mathbf{W}} L'$. Nótese que aunque usamos una base \mathbf{B} , esto es necesario solamente para obtener una representación finita de L' en la memoria del ordenador, y no se usan operadores de transporte proyectados.

En las siguientes secciones se dan detalles de cada familia de métodos, así como algunos de los artículos más importantes que los describen.

2.3 Métodos locales.

Estos métodos fueron los primeros en desarrollarse con el fin de sintetizar imágenes realistas. La restricción a un operador local produce un beneficio significativo en tiempo de cálculo y almacenamiento necesario. Subdividimos estos algoritmos en *métodos de proyección* (encontrar en que pixeles se ve cada objeto) y métodos de trazado de rayos o *ray-tracing* (encontrar que objeto u objetos se ven en cada pixel).

2.3.1 Métodos de Proyección.

En estos métodos, se calcula la porción de los pixeles de una imagen cubiertos por la proyección de cada objeto de la escena. En cada pixel, se calcula la contribución directa de luz sobre el observador más la contribución después de una única reflexión. Formalmente:

$$I = \mathcal{P}_{\mathbf{W}}(\mathcal{T} + \mathcal{I})L_e$$

normalmente se realizan las siguientes simplificaciones con el objetivo de simplificar los cálculos.

- L_0 está restringido a un conjunto de luces puntuales (con intensidad l_i y posición x_i), esto es $L_e(\mathbf{r}) = \sum_{i=1}^n l_i \delta(x_i - \mathbf{r}_s)$
- El operador \mathcal{T} incluye un término difuso-especular (Phong) y un término difuso, esto es $\mathcal{T} = \mathcal{G} + \mathcal{D}$, y puede no tener en cuenta oclusiones.

Aquí incluimos todos los algoritmos del tipo z-buffer o de líneas de barrido (*scan-line*) y que usan sombreado de Phong o Gouroud. Una referencia clásica es el artículo original de Phong [Phong75].

2.3.2 Trazado de rayos sencillo.

Esta técnica ampliamente difundida usa un árbol de rayos originado en el punto de vista para seguir (en el orden inverso) el camino que sigue la luz. Su ventaja respecto a los métodos de proyección reside en que de esta manera se pueden seguir las reflexiones especulares perfectas. La suposiciones que se hacen respecto a \mathbf{W} y L_0 son las mismas que en los métodos de proyección. El operador \mathcal{T} incluye ahora una componente especular perfecta. Esto da lugar a la siguiente imagen:

$$I = \mathcal{P}_{\mathbf{W}} (\mathcal{I} + \mathcal{D} + \mathcal{G} + \mathcal{S}^+) L_e$$

La primera referencia sobre la implementación de un trazador de rayos completo (incluyendo reflexiones especulares perfectas) es [Whitted80]

2.4 Métodos de discretización o de Elementos Finitos

En esta sección se incluyen todos los métodos en los cuales se usa un operador de transporte discreto para resolver el problema de la Iluminación Global. Esta ha sido probablemente la opción tomada por el mayor número de investigadores. Una vez que han sido dadas \mathbf{W} , \mathcal{T} y L_e , es necesario seleccionar algún conjunto finito de bases \mathbf{B} y entonces obtener $I = \mathcal{P}_{\mathbf{W}} (\mathcal{P}_{\mathbf{B}} \mathcal{T})^+ L_e$, resolviendo el sistema de ecuaciones asociado. Debido al amplio conjunto de artículos en esta categoría, los hemos dividido en varias secciones que se muestran en orden cronológico.

2.4.1 Radiosidad clásica.

En esta sección discutimos los métodos que tratan entornos con reflectividad puramente difusa y funciones base constantes. Nombraremos con \mathbf{C} a un conjunto cualquiera de funciones base constantes. Esto implica que $\mathbf{C} = \{C_1, \dots, C_n\}$ donde cada C_i es una función constante en un área de la superficies A_i (con $A_i \subseteq S$) y cero fuera de dicha área. Estas áreas son disjuntas y cubren la totalidad de las superficies. La definición de las C_i es la siguiente:

$$C_i(\mathbf{r}) = \begin{cases} 1/|A_i| & \text{si } \mathbf{r}_s \in A_i \\ 0 & \text{en otro caso} \end{cases} \quad (2.3)$$

siendo disjuntas las áreas, el conjunto \mathbf{C} es ortonormal. Se usa un operador de transporte que llamaremos \mathcal{D} para significar que solo se tienen en cuenta reflexiones difusas. Por lo tanto el problema no se plantea para rayos en general, si no solo en el dominio de los puntos, ya que al haber solo reflexiones difusas la radiancia no depende de la dirección del rayo. Por lo tanto, el objetivo es calcular la función B que es solución de la ecuación $B = B_e + \mathcal{D}B$, donde B_e es la radiosidad emitida (que también sigue una distribución difusa). Esta función B_e es inicialmente conocida, y se da como un conjunto de coeficientes e_i respecto de la base \mathbf{C} . Usando la formulación dada en el capítulo anterior, la ecuación continua se transforma en una discreta donde interviene una matriz A cuyas entradas son c_{ij} . En el contexto de entornos difusos, estas entradas se pueden escribir como $c_{ij} = F_{ji}R_i$, donde R_i se le llama *reflectividad* del área A_i y F_{ji} es el factor de forma. A todas estas entidades se les puede asignar un significado físico, que no se desarrollará aquí. El método procede en dos pasos, como sigue:

$$B' = (\mathcal{P}_\mathbf{C}\mathcal{D})^+ B_e \quad (2.4)$$

$$I = \mathcal{P}_\mathbf{W} B' \quad (2.5)$$

Una vez que B' (la función de radiosidad aproximada) se ha obtenido y almacenado en memoria, el segundo paso puede repetirse para más de un observador distinto \mathbf{W} . La solución del sistema de ecuaciones, dado su tamaño, suele hacerse por métodos numéricos iterativos [Gortler94]. Principalmente se usa alguno de estos dos métodos:

- Gauss-Seidel (GS) (Método de Jacobi con actualización directa)
Este método requiere el almacenamiento simultáneo de todos los factores de forma.
- Radiosidad progresiva o Iteración de Southwell (PR)
Solo es necesario el almacenamiento simultáneo de una fila de factores de forma. Además permite presentar los resultados intermedios como una aproximación.

El cálculo de los factores de forma usualmente demanda una gran cantidad de tiempo de CPU. Hay muchas opciones para realizar este cálculo, tales como:

- Método del Semicubo (HC).
El algoritmo que se usa para proyectar B' sobre \mathbf{W} (esto es, el método de z-buffer) se usa también para proyectar \mathcal{D} sobre \mathbf{C} . Cada fila de los factores de forma se obtiene simultáneamente.
- Métodos de cuadratura.
Cada factor de forma se obtiene (independientemente) tomando un número finito de muestras de la doble integral que lo define.
- Monte-Carlo local de acierto-fallo (MC Local)
Una fila de factores de forma se calcula cada vez, mediante la creación estocástica de rayos que parten de un área con la distribución de probabilidad adecuada.

Referencia	Fact.Forma	Solu.sist.ecuac.
[Goral84]	Stokes	Gauss-Seidel
[Cohen85]	Semicubo	Gauss-Seidel
[Cohen88]	Semicubo	Progresiva
[Malley88]	MC Local	
[Wallace89]	Cuadratura	Progresiva
[Baum89]	Stokes+SC	Progresiva
[Sbert93]	MC Global	

Tabla 2.1: Métodos de radiosidad clásica.

- Monte-Carlo global de acierto-fallo (MC Global)

En este caso, la matriz completa se obtiene mediante rayos que no están referidos a ningún área, y que siguen una distribución global. Ha sido llamado *método de las espadas*.

- Métodos analíticos (Stokes)

Usando el teorema de Stokes, el cálculo de los factores de forma puede realizarse mediante la evaluación de una expresión analítica, casi sin errores, en ausencia de oclusiones. Esto se usa en conjunción con otros métodos para poder manejar las oclusiones.

La tabla 2.1 muestra las contribuciones más importantes en esta línea, incluyendo cuales de las opciones de la mostradas arriba se han usado en cada artículo.

2.4.2 Métodos de una pasada para entornos no puramente difusos.

Estos métodos son extensiones de la radiosidad clásica que intentan incluir reflexiones especulares perfectas y/o reflexiones difuso-especulares además de reflexiones difusas. Formalmente, se calcula la siguiente imagen:

$$I = \mathcal{P}_{\mathbf{W}} (\mathcal{P}_{\mathbf{B}} \mathcal{T})^+ L_e \quad (2.6)$$

donde $\mathcal{T} = \mathcal{S} + \mathcal{D} + \mathcal{G}$, siendo \mathcal{S} el operador de reflexión especular perfecta, \mathcal{D} el difuso, y \mathcal{G} el difuso-especular. Se lleva a cabo un cálculo en dos fases, parecido a la radiosidad clásica, como se describe a continuación

$$(1) \quad L' = (\mathcal{P}_{\mathbf{B}} (\mathcal{S} + \mathcal{D} + \mathcal{G}))^+ L_e \quad (2.7)$$

$$(2) \quad I = \mathcal{P}_{\mathbf{W}} L' \quad (2.8)$$

En este caso, se calculan radiancias en lugar de radiosidades, y el problema tiene una dimensión adicional con respecto a la radiosidad clásica. Esto supone un tiempo de cálculo más grande. Hay dos opciones para escoger el conjunto de funciones base que se usan para el cálculo:

Referencia	Basds	Estim.Inic.
[Immel86]	Semicubo	Emitido
[Shao88]	Semicubo	Difuso
[Buckalew89]	Enlaces	Emitido

Tabla 2.2: Métodos de una pasada para entornos no difusos.

- Semicubos

Cada B_i es no cero en los rayos que parten del centro de un área y pasan por un determinado pixel del Semicubo. Por tanto la función de radiancia obtenida esta proyectada en los pixeles de las caras del Semicubo.

- Enlaces

Se usa una estructura muy densa de segmentos de recta (enlaces), cada uno de los cuales une dos áreas. Cada base B_i esta asociada a uno de estos enlaces (un rayo que llamaremos \mathbf{l}_i). La base es distinta de cero únicamente para ese rayo, esto es $B_i(\mathbf{r}) = \delta(\mathbf{r} - \mathbf{l}_i)$. Por tanto, la radiancia se obtiene como una función de la cual conocemos su valor en cada rayo o enlace.

Una vez establecido, el sistema de ecuaciones se resuelve por Gauss-Seidel. Esto requiere una estimación inicial de las radiancias antes de comenzar las iteraciones. Hay dos opciones para esto:

- Usar L_e , la función de radiancia emitida.
- Usar $L' = (\mathcal{P}_B \mathcal{D})^+ L_e$, esto es, la radiancia calculada teniendo en cuenta únicamente reflexiones difusas, mediante radiosidad clásica. En este caso, la estimación inicial de la radiancia puede estar más cerca del valor verdadero, sobre todo si no hay muchas superficies especulares.

La tabla 2.2 muestra algunas contribuciones relevantes en esta línea. Este método plantea dos problemas:

- Requiere una gran cantidad de memoria y tiempo de cálculo.
- Es difícil tener en cuenta correctamente el operador especular perfecto, ya que la correspondiente BRDF está basada en una función delta, de forma que la proyección del operador de transporte produce un operador discreto con una gran cantidad de error.

2.4.3 Métodos de dos pasadas para entornos no difusos.

Los métodos de dos pasadas se usan para tratar de solventar los problemas de los métodos descrito en un apartado anterior. Estos métodos están basados en la separación del operador general \mathcal{T} en la adición de otros dos:

$$\mathcal{T} = \mathcal{T}_1 + \mathcal{T}_2 \quad (2.9)$$

El uso de los subíndices 1 y 2 se debe al hecho de que \mathcal{T}_1 se trata en la primera pasada, mientras que en la segunda pasada se tiene en cuenta \mathcal{T}_2 . Con el fin de mostrar la validez del método, introducimos la siguiente propiedad que cumple esta descomposición de \mathcal{T} . La propiedad es:

$$\mathcal{T}^+ = \mathcal{T}_2^+ (\mathcal{T}_1 \mathcal{T}_2^+)^+ = \mathcal{T}_2^+ \mathcal{U}^+ \quad (2.10)$$

donde $\mathcal{U} = \mathcal{T}_1 \mathcal{T}_2^+$. Con esto, vemos que es posible calcular una aproximación a $\mathcal{U}^+ L_e$ usando métodos de proyección, y después obtener la imagen mediante un algoritmo parecido al trazado de rayos. Esto puede expresarse con el siguiente algoritmo de dos pasadas:

$$\begin{aligned} (1) \quad L' &= (\mathcal{P}_B \mathcal{U})^+ L_e \\ (2) \quad I &= \mathcal{P}_W \mathcal{T}_2^+ L' \end{aligned} \quad (2.11)$$

Usualmente, el operador \mathcal{D} (difuso) se encuentra incluido en \mathcal{T}_1 , y el operador \mathcal{S} (especular perfecto) se incluye en \mathcal{T}_2 .

El paso (1) es igual al primer paso de los métodos de elementos finitos ya introducidos, excepto por los coeficientes de $\mathcal{P}_B \mathcal{U}$, que han sido llamados *factores de forma extendidos*, ya que también tienen en cuenta el término \mathcal{T}_2^+ que aparece en \mathcal{U} . La función L' tiene en cuenta todos los posibles caminos de partículas que acaban en una reflexión correspondiente al operador \mathcal{T}_1 . El paso (2) es formalmente equivalente a un método de ray-tracing (o una método de path-tracing de Monte-Carlo), excepto por el uso de L' en lugar de L_e como función fuente. Esta segunda pasada simula los caminos de partículas que comienzan después de una reflexión tipo \mathcal{T}_1 , después pasan por un numero arbitrario de reflexiones tipo \mathcal{T}_2 y acaban en el observador. Se dice que estos métodos son una combinación de radiosidad (por la primera pasada) y ray-tracing (por la segunda pasada).

Hay dos opciones para la descomposición de \mathcal{T} en \mathcal{T}_1 y \mathcal{T}_2 según cual de estos dos operadores incluye el término difuso-especular (\mathcal{G})

- $\mathcal{T}_1 = \mathcal{D}$ y $\mathcal{T}_2 = \mathcal{G} + \mathcal{S}$

En este caso a L' (el resultado de la primera pasada) se le llama emisión difusa. Es necesario recurrir a un trazado de rayos estocástico en la segunda pasada, con el fin de tratar apropiadamente la presencia del operador \mathcal{G} en \mathcal{T}_2 .

- $\mathcal{T}_1 = \mathcal{D} + \mathcal{G}$ y $\mathcal{T}_2 = \mathcal{S}$

Aquí L' depende de la dirección. Consecuentemente la primera pasada es más complicada mientras que la segunda es un método de trazado de rayos sencillos que solo tiene en cuenta reflexiones especulares.

Para resolver el sistema de ecuaciones en la primera pasada, es frecuente usar un método progresivo. En cada paso, la energía del área fuente debe ser distribuida a través de cero o más reflexiones especulares hasta que se llega a una reflexión difusa. Hay dos opciones para realizar esto:

- Se puede usar *trozos virtuales*, que son duplicados especulares del trozo emisor en cada espejo. El factor de forma desde estos trozos virtuales se debe añadir al original.

Referencia	\mathcal{T}_1	\mathcal{T}_2	Funciones base	Fact.Form. Extendidos
[Wallace87]	\mathcal{D}	$\mathcal{S} + \mathcal{G}$	Constant	Trozos virtuales
[Sillion89]	\mathcal{D}	$\mathcal{S} + \mathcal{G}$	Constant	ray-casting recursivo
[Sillion91]	$\mathcal{D} + \mathcal{G}$	\mathcal{S}	Armónicos Esféricos	Trozos virtuales

Tabla 2.3: Métodos de dos pasadas para entornos no difusos.

- Usar trazador de rayos recursivo en el caso de que un trozo emisor "ve" un trozo especular perfecto.

La tabla 2.3 resume los trabajos que exponen estos métodos

2.4.4 Métodos de Importancia y Multiresolución.

El orden de complejidad para todos los anteriores métodos de elementos finitos es al menos de $O(n^2)$ donde n es el número de funciones base del conjunto \mathbf{B} . Al aumentar este número, también lo hace la calidad de la aproximación. Sin embargo, la complejidad crece cuadráticamente. En aplicaciones prácticas, el número de funciones base (trozos) requeridos para obtener una calidad razonable es tan grande que dichos métodos se vuelven impracticables. La complejidad viene principalmente del cálculo de los coeficientes del operador proyectado.

Una opción para resolver esto en el contexto de los métodos de elementos finitos es el uso de conjuntos jerárquicos de funciones base o bases tipo Wavelet. Estas técnicas proporcionan una complejidad menor para el cálculo del operador proyectado o la solución del sistema de ecuaciones. Dichas técnicas se han usado en otros campos de la Informática Gráfica y la Visión por Ordenador, como es el análisis y compresión de imágenes, el diseño de curvas y superficies, el almacenamiento y procesamiento de texturas, el modelado de sólidos, etc...

2.4.5 Proyección en bases Wavelet.

Como se describe en [Mallat89], un Análisis Multiresolución (MRA) es un conjunto de espacios vectoriales $\{V_i, i \in \mathbb{N}\}$ tal que $V_i \subset V_{i+1}$ y que $V_i \subset \mathcal{L}^2(D)$. En otras palabras, cada V_i es un conjunto de funciones cuyo cuadrado es integrable en el dominio D . Llamemos \mathbf{S}_i a cualquier base ortonormal de V_i . La definición de un MRA incluye restricciones adicionales que aseguran que la proyección de f sobre \mathbf{S}_i converge a f cuando i tiende a infinito. Cada \mathbf{S}_i es una base finita que sirve para aproximar cualquier función con una resolución creciente con i .

Ya que $V_i \subset V_{i+1}$ podemos considerar el espacio vectorial complementario a V_i en V_{i+1} . Lo llamaremos U_i . Si se suman funciones de V_i con funciones de U_i , obtenemos funciones en V_{i+1} . Podemos establecer, por tanto, que las funciones de U_i representan las diferencias entre las funciones de V_i y aquellas de V_{i+1} . Llamemos ahora \mathbf{D}_i a cualquier base ortonormal de U_i . Cada elemento de \mathbf{S}_i es ortonormal a cualquier elemento de \mathbf{D}_i , y $\mathbf{S}_i \cup \mathbf{D}_i$ es también una base de V_{i+1} . Las funciones

en \mathbf{S}_i se llaman *funciones de escalado* (del inglés *scaling functions*), y aquellas de \mathbf{D}_i se llaman *funciones de detalle*.

Podemos entonces construir una base \mathbf{H}_n de V_n uniendo las funciones de escala al más alto nivel con las funciones de detalle de sucesivos niveles, como sigue:

$$\mathbf{H}_n = \mathbf{S}_0 \bigcup_{i=0}^n \mathbf{D}_i$$

Cuando se proyecta una función cualquiera f contra \mathbf{H}_n , obtenemos una proyección muy simple de f (su proyección contra \mathbf{S}_0) más las sucesivas diferencias con proyecciones más detalladas (los coeficientes de las proyecciones contra las funciones de detalle \mathbf{D}_i), hasta una resolución máxima n . En el caso de que la proyección de f sea buena a una resolución cualquiera m , los coeficientes (diferencias) de las proyecciones contra las siguientes funciones de detalles son muy pequeños.

Para cualquier operador dado \mathcal{T} , el correspondiente operador discreto¹ $\mathcal{P}_n \mathcal{T}$ se representa mediante una matriz de coeficientes. Si se fijan a cero todas las entradas de dicha matriz cuyo valor esté por debajo de un valor límite previamente establecido, entonces se obtiene una matriz dispersa (muchas entradas a cero) [Beylkin91, Alpert93], tal que las operaciones con esa matriz se realizan con una complejidad lineal con el número de funciones base.

Los métodos jerárquicos usan esta propiedad, mediante el cálculo de operadores proyectados contra una base jerárquica, de forma que se evita el cálculo de los coeficientes que estén por debajo de cierto valor, y se logra alcanzar la complejidad lineal en lugar de cuadrática.

2.4.6 Familias de wavelets.

La base wavelet más simple usada en Iluminación Global es la base Haar. Por simplicidad, asumimos que el dominio de los wavelets es el intervalo unidad, $D = [0, 1]$. El conjunto \mathbf{S}_i tiene 2^i elementos (funciones base), que pueden definirse como:

$$\mathbf{S}_i = \{\phi_{ij}(x) = 2^{i/2} \phi_{00}(2^i x - j) \mid j = 0, \dots, 2^i - 1\}$$

donde $\phi_{00}(x)$ es 1 cuando $x \in [0, 1]$ y 0 en otro caso. Cada función es una versión trasladada y escalada de ϕ_{00} . Estas bases wavelets son fáciles de tratar. Desafortunadamente, los espacios V_i solamente incluyen funciones constantes a trozos y por tanto discontinuas. Con el fin de solventar esta limitación, las bases Haar pueden generalizarse de dos formas:

- En lugar de que \mathbf{S}_0 esté formado por una única función base, se puede incluir un conjunto de funciones polinomiales ortogonales, que forman una base de los polinomios de orden menor que k en el intervalo unidad. De esta forma V_i incluiría funciones discontinuas polinomiales a trozos. Esta clase de wavelets se llaman *Multiwavelets* de orden k , y se describen en [Alpert93].

¹En adelante, y para simplificar la notación, usaremos \mathcal{P}_n para significar el operador de proyección contra la base \mathbf{H}_n

- Otra posibilidad es usar también polinomios de orden k , pero, en este caso, asegurándonos también de que son continuos y k veces diferenciables en el intervalo unidad. Esto puede lograrse usando funciones base B-Spline de grado k . A estos se les llama *Wavelets Spline*, y son continuos y diferenciables. Desgraciadamente, no son ortonormales sino semiortonormales.

2.4.7 Refinamiento por brillo e importancia y métodos jerárquicos.

En teoría hay un único conjunto de funciones de escala al más alto nivel, es decir un único conjunto S_0 . Pero en la práctica cada superficie de cada objeto del entorno implica un conjunto distinto de bases S_0 . Si hay n objetos, la complejidad es cuadrática con el número de objetos. Aún usando funciones base jerárquicas, el orden de complejidad puede ser demasiado alto ya que el número de objetos puede ser también grande. Una solución es el uso de refinamiento de brillo e importancia.

En este contexto, la importancia es la función $f = (\mathcal{T}^*)^+ W_e$ (donde $W_e = \sum_i W_{ei}$), que proporciona para cada rayo r la fracción de radiancia de r que alcanza al observador (después de un número arbitrario de reflexiones). Como esta definición incluye el operador $(\mathcal{T}^*)^+$, el cálculo de la función de importancia es tan complejo como el cálculo de la radiancia. Sin embargo, esta función proporciona una valor que indica la contribución del error cometido en cada rayo al error en la imagen. Recuérdese que los métodos jerárquicos están basados en descartar entradas inferiores a un valor mínimo. Por tanto, si se multiplica los coeficientes por la importancia, estos valores se reducen y es posible descartar más coeficientes, y por tanto los cálculos pueden acelerarse más.

El refinamiento por brillo e importancia (en adelante, refinamiento BI) se basa en el cálculo simultáneo de brillo e importancias. Esto se hace usando iteración de Jacobi en sucesivas aproximaciones de ambas funciones. A cada paso el valor mínimo para descartar coeficientes se disminuye, con lo cual nuevos coeficientes son tenidos en cuenta (o, lo que es lo mismo, el transporte se simula a más baja resolución). Esta inclusión de nuevos coeficientes (o enlaces), es adaptativo en el sentido de que se realiza más esfuerzo en las zonas que más contribuyen al error en la imagen.

De hecho, este método produce una función discreta cuya resolución es más alta en las zonas más iluminadas (brillo) o con mayor contribución a la imagen (importancia). Esto se ha llamado una solución con resolución dependiente del punto de vista.

Los métodos jerárquicos se han usado para calcular radiosidad difusa (solo operador \mathcal{D}) y también para entornos difuso-especulares (operador \mathcal{G}). En el último caso, la radiancia puede definirse en el dominio de los pares de puntos-dirección (PD) o bien en el dominio de los pares de puntos de las superficies (PP). La tabla 2.4 muestra algunas contribuciones en esta línea:

Referencia	Oper.	Bases	refin-BI	Dominio
[Hanrahan91]	\mathcal{D}	Haar	No	P
[Smits92]	\mathcal{D}	Haar	Si	P
[Gortler93]	\mathcal{D}	Multiwavelets	No	P
[Aupperle93]	$\mathcal{D} + \mathcal{G}$	Haar	Si	PP
[Pattanaik94]	$\mathcal{D} + \mathcal{G}$	Haar	No	PP
[Christensen94]	$\mathcal{D} + \mathcal{G}$	Haar	Si	PD
[Schroeder94]	$\mathcal{D} + \mathcal{G}$	Multiwavelets	Si	PP
[Yu95]	\mathcal{D}	B-Spline	No	P

Tabla 2.4: Métodos jerárquicos.

2.5 Métodos Estocásticos o de Monte-Carlo.

Todos los métodos de esta sección están basados en el uso de cadenas de Markov como la herramienta básica para resolver la ecuación de Iluminación Global. Incluso cuando se usan cadenas de Markov es necesario recurrir a una base finita para almacenar la solución aunque no se use ningún operador de proyección.

2.5.1 *Path-tracing* o Monte-Carlo desde el observador.

Estos métodos pueden verse como una extensión del trazado de rayos, o bien como una instancia de los métodos MC (Monte-Carlo) para ecuaciones integrales. Sabemos que la intensidad l_i del i -ésimo pixel en la imagen se obtiene como $l_i == \langle W_{ei} | \mathcal{T}^+ L_e \rangle$. El método procede secuencialmente pixel a pixel, y calcula una aproximación (bajo una varianza dada) de su intensidad. Este valor se obtiene mediante una evaluación de (1.82) para un conjunto dado de cadenas de puntos o caminos. Una elección común para las funciones p_0 y p es la siguiente:

$$p_0(\mathbf{r}) = W_{ei}(\mathbf{r}) \quad (2.12)$$

$$p(\mathbf{r}, \mathbf{s}) = K(\mathbf{r}, \mathbf{s}) \quad (2.13)$$

$$p(\mathbf{r}, a) = 1 - \int_D K(\mathbf{r}, \mathbf{x}) d\mathbf{x} \quad (2.14)$$

El primer rayo (para el pixel i -ésimo) se selecciona con una *pdf* proporcional a W_{ei} . A partir de aquí, todos los demás de ese camino se crea a partir del anterior usando una P proporcional a K como función de probabilidad condicional para las transiciones. La probabilidad de absorción (esto es, la probabilidad de alcanzar el estado a) se hace igual a la fracción de energía absorbida. En estas condiciones, la expresión (1.82) se convierte en:

$$V(c) = \sum_{i=0}^k L_e(c_i)$$

donde k es el ultimo estado de c antes de la absorción, esto es, $c_j = a \forall j > k$. Existen otras opciones para fijar la probabilidad de absorción, tales como la técnica llamada *ruleta rusa* [Arvo90].

Con el objetivo de reducir la varianza, a veces se usan otras técnicas. Una de estas es el muestreo directo de las fuentes de luz, que es un algoritmo basado en la siguiente formulación:

$$\begin{aligned} I &= \mathcal{P}_{\mathbf{W}} \mathcal{T}^+ L_e \\ &= \mathcal{P}_{\mathbf{W}} (\mathcal{T}^+ \mathcal{T} + \mathcal{I}) L_e \\ &= \mathcal{P}_{\mathbf{W}} \mathcal{T}^+ (\mathcal{T} L_e) + \mathcal{P}_{\mathbf{W}} L_e \end{aligned}$$

Aquí se puede ver como es posible usar $\mathcal{T} L_e$ en lugar de L_e como el término fuente, siempre que se añada el término $\mathcal{P}_{\mathbf{W}} L_e$ (luz de las fuentes que alcanza directamente al observador).

Aquí hemos usado la igualdad $\mathcal{T}^+ = \mathcal{I} + \mathcal{T} \mathcal{T}^+$, que se ha establecido previamente en la ecuación (1.52). La evaluación del término fuente $\mathcal{T} L_e$ en cada punto de impacto puede realizarse a su vez mediante técnicas de MC (lo que se llama *muestreo de las fuentes de luz*). El término $\mathcal{P}_{\mathbf{W}} L_e$ se puede añadir sumando la radiancia emitida en el primer impacto de cada camino. Esta técnica reduce la varianza ya que la función L_e suele ser cero en la mayoría de los puntos de la escena, con zonas distintas de cero aisladas en las fuentes de luz. Esto implica una alta varianza ya que los caminos difícilmente alcanzan las fuentes. Por otra parte, la función $\mathcal{T} L_e$ (luz reflejada de la que proviene directamente de las fuentes) es una función más suave y distinta de cero en más puntos, lo cual implica menor varianza (una comparación de estas opciones puede encontrarse en [Shirley92]).

Existen otras opciones para la función de transición p . Cuando las fuentes de luz no iluminan directamente la escena, si no que lo hacen indirectamente, el método puede mostrar una alta varianza incluso con el muestro directo de las fuentes. Esto se debe al hecho de que en estas condiciones, incluso la función $\mathcal{T} L_e$ muestra picos aislados. Para solventar este problema se puede usar una p tal que las cadenas se construyen usando dos subcadenas, una que se comienza desde el observador (la *pdf* para el primer rayo es proporcional a W_{ei}) y otra que comienza en las fuentes de luz (la *pdf* para el primer rayo es proporcional a L_e). A esto se le llama *path-tracing bidireccional*, pues las cadenas se construyen en las dos direcciones.

La longitud de la cadena unión de las dos subcadenas debe ser establecida a priori, con lo cual las cadenas de distintas longitudes deben de muestrearse explícitamente. Esto implica que se puede calcular un número finito de iteraciones del operador. Por lo tanto se simula el operador $\mathcal{T}^{(k)}$ en lugar de \mathcal{T}^+ , para alguna longitud máxima de las cadenas k . Formalmente esta formulación se basa en las propiedades del operador adjunto, ya que:

$$l_t = \left\langle \mathcal{T}^{(k)} L_e \mid W_{et} \right\rangle \quad (2.15)$$

$$= \sum_{n=0}^k \left\langle \mathcal{T}^{(n)} L_e \mid W_{et} \right\rangle \quad (2.16)$$

Referencia	PDF	Coherencia	Sel.Optima
[Cook84]	unidireccional	No	No
[Kajiya86]	unidireccional	No	No
[Ward88]	unidireccional	Si	No
[Veach94]	bidireccional	No	No
[Lafortune94]	bidireccional	No	No
[Veach95]	bidireccional	No	Si

Tabla 2.5: Trazado de caminos desde el observador.

$$= \sum_{n=0}^k \sum_{i+j=n} w_{ij} \left\langle \mathcal{T}^{(i)} L_e \mid (\mathcal{T}^*)^{(j)} W_{et} \right\rangle \quad (2.17)$$

donde w_{ij} es un conjunto de pesos tal que $\sum_{i+j=n} w_{ij} = 1$ para todos los $n \geq 0$. Estos pesos pueden seleccionarse arbitrariamente, aunque en [Veach95] se describe un método para la selección óptima de los mismos (en el sentido de reducir la varianza).

Como se ha explicado, todos estos métodos proceden pixel a pixel independientemente, y no usan la posible coherencia presente en la función L . En el caso de que esta función no presente variaciones muy altas en una zona, es posible usar una estimación de su valor en unos puntos para predecir dicho valor en otros puntos. Esto se puede aprovechar guardando dichas muestras y reusándolas cuando sea necesario.

La tabla 2.5 muestra algunos trabajos publicados en esta línea.

2.5.2 Métodos de fotosimulación o seguimiento de partículas.

Este método ha sido llamado de varias formas, por ejemplo *Photon-tracking* (seguimiento de fotones), *MC path tracing from the light sources* (trazado de caminos desde las fuentes de luz), o *shooting random walk* (paseo aleatorio de disparo), o *backward ray-tracing* (trazado de rayos inverso o hacia atrás).

El presente método también está basado en cadenas de Markov. De hecho se usan los mismos principios que los métodos previos, excepto que en este caso se recurre a la formulación adjunta, y por tanto se encuentra involucrado el operador adjunto. (\mathcal{T}^* en lugar de \mathcal{T}). Podemos calcular la función $L' = \mathcal{P}_{\mathbf{B}} \mathcal{T}^+ L_0$ (para cualquier base \mathbf{B}) como un conjunto de coeficientes l_i usando lo siguiente

$$l_i = \langle b_i \mid \mathcal{T}^+ L_e \rangle = \langle L_e \mid (\mathcal{T}^*)^+ b_i \rangle$$

Así que el cálculo de estos coeficientes puede verse como una instancia de (1.84) donde $g = L_e$ y $h_i = b_i$. En este contexto, la selección de p_0 y p se hace usualmente como sigue:

$$p_0(\mathbf{r}) = \frac{L_e(\mathbf{r})}{\Phi} \quad (2.18)$$

$$p(\mathbf{r}, \mathbf{s}) = K(\mathbf{r}, \mathbf{s}) \quad (2.19)$$

$$p(a, \mathbf{s}) = 1 - \int_D K(\mathbf{x}, \mathbf{s}) d\mathbf{x} \quad (2.20)$$

donde Φ se introduce para normalizar p , esto es, $\Phi = \int_D L_e(\mathbf{r}) d\mu(\mathbf{r})$, la energía total emitida en el entorno. En el caso de usar funciones base constantes disjuntas, tal y como se definen en (2.3), los valores de b_i son bien $1/|A_i|$ o cero, y el cálculo de la sumatoria sobre el conjunto de funciones base no es necesario. En lugar de esto, simplemente hay que realizar una búsqueda del trozo que contiene el punto de impacto. Si llamamos n_i al número de impactos en el trozo número i para un total de m cadenas, entonces (1.84) se simplifica:

$$l_i \approx \frac{\Phi n_i}{m |A_i|}$$

Se puede dar una interpretación física de este método. Cada cadena se puede ver como un fotón o partícula. Dicho fotón *nace* en un estado o rayo \mathbf{r} con probabilidad proporcional a $L_e(\mathbf{r})$. A partir de aquí, la probabilidad de pasar de un estado \mathbf{s} a otro \mathbf{r} es proporcional a $K(\mathbf{r}, \mathbf{s})$. Finalmente, el fotón alcanza el estado de absorción a partir de otro estado con una probabilidad igual a la fracción de radiancia en el último estado que es absorbida por las superficies. Cada fotón transporta una energía igual a Φ/m que se *deposita* en cada trozo de las superficies que es alcanzado. La división por $|A_i|$ se utiliza para convertir energía total en energía por unidad de área. Por lo tanto, la última ecuación puede verse simplemente como un conteo de la energía de los fotones que llegan al trozo. Las primeras descripciones de este método se realizaron en este contexto [Spanier69, Kalos86], y por lo tanto puede llamarse fotosimulación.

En lugar de esta selección básica, podemos usar una función p basada en importancia, de forma que una densidad de impactos más alta se obtiene en las zonas de las superficies que más contribuyen a la imagen. Hay varias opciones para lograr esto. A modo de ejemplo, se puede ver el trabajo de Pattanaik [Pattanaik93].

Una vez que la función L' ha sido obtenida, es aún necesario proyectarla sobre algún observador para obtener una imagen. Este último paso se realiza usando un método de proyección, como los que se detallan en la sección 2.4. Si el operador \mathcal{T} incluye algún término especular perfecto, entonces L' no podría capturar los detalles de la solución exacta L . Esto se puede resolver mediante métodos de dos pasadas, como se describe en la sección 2.4.3. Si $\mathcal{T} = \mathcal{T}_1 + \mathcal{T}_2$ entonces el proceso es como sigue:

$$\begin{aligned} L' &= \mathcal{P}_B \mathcal{U}^+ L_e \\ I &= \mathcal{P}_W \mathcal{T}_2^+ L' \end{aligned}$$

donde $\mathcal{U} = \mathcal{T}_1 \mathcal{T}_2^+$. De nuevo, el trazado de rayos simple se usa para lograr este paso.

Finalmente existe un método de fotosimulación en una sola pasada para entornos sin reflexiones especulares perfectas. En este caso la imagen I se obtiene

Referencia	Operador	Imp.	Sol.Direct
[Arvo86]	$\mathcal{S}^+ \mathcal{D} \mathcal{S}^+$	No	No
[Heckbert90]	$\mathcal{S}^+ \mathcal{D} \mathcal{S}^+$	Si	No
[Pattanaik92]	$(\mathcal{S} + \mathcal{D} + \mathcal{G})^+$	No	No
[Pattanaik93]	$(\mathcal{S} + \mathcal{D} + \mathcal{G})^+$	Si	No
[Dutre94]	$(\mathcal{D} + \mathcal{G})^+$	Si	Si

Tabla 2.6: *Path-tracing* desde las fuentes de luz

directamente sin pasar por una L' intermedia. Puede verse como el método adjunto al muestreo directo de las fuentes de luz en Path-tracing. En lugar de usar fotosimulación para obtener una proyección con respecto a una base \mathbf{B} y después proyectar esta aproximación contra el observador \mathbf{W} , es posible proyectar directamente contra la base \mathbf{W} . Por lo tanto, la imagen se obtiene directamente como el resultado del proceso de simulación. Para cada estado \mathbf{r} del fotón se calcula cual es el pixel i tal que $\mathcal{T}^* W_{ei}(\mathbf{r})$ es distinto de cero (es decir, el pixel donde se proyecta el siguiente punto de impacto), y se añade la energía reflejada del fotón hacia el pixel al estimador asociado al pixel.

La tabla 2.6 incluye varios artículos que describen esta clase de métodos. La columna encabezada con *Sol.Direct* indica si el método anterior se usa o no.

2.6 Conclusiones.

En este segundo capítulo se presenta una formalización del problema de la Iluminación Global, introduciendo el concepto de observador como un conjunto de funciones que miden la radiancia en determinadas partes de la escena. Asimismo se introduce el concepto de imagen como la función que se obtiene al proyectar la función de radiancias en la base inducida por el observador.

Combinando este concepto con los introducidos en el capítulo anterior, obtenemos una imagen como el resultado de aplicar un operador lineal sobre la función de radiancia emitida.

Esta formulación representa una formalización del problema de Iluminación Global que ayuda a comprender y comparar mutuamente los algoritmos de Iluminación Global presentados en un amplio conjunto de artículos. Dichos artículos se presentan de forma tabular, de manera que se ponen de manifiesto las diferencias y semejanzas entre ellos.

Capítulo 3

La Varianza de los Métodos de Monte-Carlo.

3.1 Introducción.

En este capítulo, se estudia a fondo la naturaleza de la varianza de los métodos de Monte-Carlo basados en Cadenas de Markov. Para ello, definimos una función de medida de probabilidad en el espacio de todas las cadenas posibles. Esta medida dicta la probabilidad de seleccionar una cadena cualquiera. Usando dicha medida se construye la variable aleatoria que es muestreada en los métodos de path-tracing. Obtenemos a continuación la media y la varianza de dicha variable. Con esto se demuestra que el método no presenta sesgo, y conocemos su varianza. Vemos que, al igual que la función solución, la varianza cumple una ecuación integral de Fredholm segundo orden. Esto se puede usar para caracterizar la varianza de algunos algoritmos de Monte-Carlo.

Un procedimiento similar al anterior se usa para analizar la varianza del path-tracing con muestreo directo de las fuentes de luz.

3.2 La Varianza del Path-Tracing

En esta sección consideramos el path-tracing simple (o la fotosimulación, si consideramos la ecuación adjunta). Como se ha establecido antes, definiremos formalmente una variable aleatoria. Dicha variable estará compuesta de una función de medida de probabilidad en el espacio de las cadenas, y de una función de valores reales sobre dicho espacio. La expresión de la varianza que se obtiene coincide con la que aparece en [Mikhailov92], aunque en dicha obra no se incluye la demostración. Dicha demostración fue derivada independientemente por los autores, antes de conocer la citada referencia.

El primer estudio de la varianza de los métodos de Monte-Carlo en el contexto de la Informática Gráfica aparece en [Shirley91]. En este caso se analiza el proceso

de fotosimulación para entornos difusos, en el cual las partículas solo contribuyen en el último impacto. Aquí se dan cotas de la varianza que permiten establecer que el tiempo de cálculo tiene una complejidad lineal con el número de trozos de la escena, bajo ciertas condiciones sobre el tamaño relativo de dichos trozos.

Un estudio más amplio del caso difuso aparece en [Sbert97]. Aquí se proporciona la varianza para trazado de partículas y para path-tracing, tanto para estimadores con contribución en el último impacto como para aquellos con contribución en todos los impactos. En este caso, también se pueden establecer cotas que implican complejidad lineal en el tiempo en todos los casos. Sin embargo, el estudio se restringe a un tipo especial de path-tracing en el cual el punto de salida después de cada impacto está distribuido aleatoriamente en el trozo que contiene el punto impacto. Con esto, se obtiene una solución estocástica a la ecuación integral discretizada en lugar de la solución a la ecuación continua. Sin embargo, dichas soluciones pueden extenderse al caso continuo como se describe brevemente en [Sbert97b].

En los trabajos citados anteriormente no se aborda el problema de reflectores no difusos ni de probabilidades de transición arbitrarias. En este capítulo trataremos el problema de la forma más general posible, es decir, usando probabilidades de transición arbitrarias en entornos de comportamiento reflectivo también arbitrario. En realidad, se calcula la varianza para una ecuación general, y luego el resultado se puede instanciar para un número de problemas concretos en Iluminación Global que se pueden resolver usando muestreo de Cadenas de Markov

3.2.1 Densidad de Cadenas

La medida de probabilidad para cadenas.

Consideremos el conjunto $D' = D \cup \{a\}$, donde a es un estado especial llamado el *estado de absorción*, con $a \notin D$. Recordemos que D es el dominio donde se realiza la integración que aparece en la ecuación (1.27).

Las integrales en D pueden extenderse a integrales en D' extendiendo la definición de la función de medida m usada en (1.27) a una nueva medida m' definida en D' . Para cualquier conjunto $A \subseteq D'$, m' se define como

$$m'(A) = \begin{cases} m(A) & \text{si } a \notin A \\ m(A - \{a\}) + 1 & \text{si } a \in A \end{cases}$$

la definición previa implica que $m'(\{a\}) = 1$, la medida del estado de absorción es siempre 1, independientemente de la medida del resto de elementos en D . Esta definición implica que para cualquier función f definida en D' (cuya restricción a D sea integrable bajo m) se cumple que

$$\int_{D'} f(x) dm'(x) = f(a) + \int_D f(x) dm(x)$$

Es fácil comprobar que m' es una función de medida válida.

La función de transición.

Usando la medida anterior podemos definir una *función de probabilidad de transición* como una función p definida en $D' \times D'$, con las siguientes propiedades, que deben cumplirse para todos los r, s pertenecientes a D'

$$1 = \int_{D'} p(r, t) dm'(t) \quad (3.1)$$

$$1 = p(a, a) \quad (3.2)$$

$$k(r, s) > 0 \rightarrow p(r, s) > 0 \quad (3.3)$$

$$p(r, a) > 0 \quad (3.4)$$

El valor $p(r, s)$ es probabilidad condicional, en una cadena cualquiera de estados de D' , de que el siguiente estado sea s , suponiendo que el anterior es r .

La igualdad (3.1) asegura que p esta normalizada y puede usarse como medida de probabilidad. La igualdad (3.2) asegura que todas las transiciones desde el estado de absorción van de nuevo al estado de absorción. La implicación (3.3) asegura que todos los estados s que contribuyen a la función objetivo en r (es decir, todos los estados s tales que $k(r, s) > 0$), tienen una oportunidad no nula de ser seleccionados después de r . Finalmente, la condición (3.4) asegura que toda cadena alcanzará el estado de absorción con una probabilidad que converge a uno cuantas mas transiciones ocurren. Esto, a su vez, implica que es imposible encontrar una cadena con un número infinito de estados de estados que no alcance el estado de absorción.

Una medida de probabilidad para Cadenas de Markov.

Definamos C como el conjunto de todas las posibles listas infinitas de elementos en D' . Formalmente, C se define como la solución más pequeña (en el sentido de inclusión de conjuntos) a la siguiente ecuación de conjuntos

$$C = D' \times C$$

Consideremos ahora cualquier cadena $c \in C$ con $c = \{c_0, c_1, c_2, \dots\}$ y un elemento $x \in D'$. Obviamente, por la ecuación que define C , se cumple que $(x, c) \in C$. En adelante escribiremos simplemente xc para significar (x, c) , esto es, xc es la cadena infinita que se obtiene anteponiendo x al principio de c . Por lo tanto, $xc = \{x, c_0, c_1, \dots\}$. En la figura 3.1 se puede ver un ejemplo de una cadena en un entorno. Nótese que el primer elemento r de esta cadena es un rayo que apunta al observador.

Una vez que C ha sido definido, abordaremos la definición de una función de medida de probabilidad en el dominio C . Dicha medida se llamará U_r , y depende tanto de la función p como de un elemento r de D' . Dicho elemento se pone como subíndice del nombre para hacer explícita la dependencia. La medida de probabilidad U_r puede usarse para realizar integrales en C , en el sentido de Lebesgue. La definición de la medida viene dada por las dos siguientes restricciones

$$\frac{dU_r(sc)}{dm'(s) dU_s(c)} = p(r, s) \quad \wedge \quad U_r(C) = 1 \quad (3.5)$$

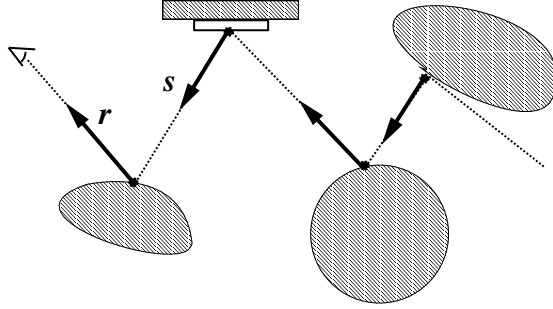


Figura 3.1: Ejemplo de una cadena de rayos.

Podemos ver el valor (diferencial) $dU_r(c)$ como la probabilidad de que una cadena pase por los estados de c después justo de haber pasado por r . De las anteriores restricciones deducimos que $dU_r(sc) = p(r, s)dm'(s)dU_s(c)$ por lo tanto, se ve que $dU_r(sc)$ es igual a la probabilidad de transición de r a s multiplicada por la probabilidad de que, una vez visitado s , se pasen por los estados de c . La condición $U_r(C) = 1$ se introduce para asegurar la unicidad ¹ de U_r y también para que sea una medida de probabilidad correctamente normalizada.

3.2.2 Una variable aleatoria sobre Cadenas de Markov

Una vez que tenemos la medida de probabilidad U_r , podemos definir variables aleatorias en C , referidas a dicha medida. Definiremos una variable aleatoria de nombre X_r cuyo dominio es C . La r en el subíndice nota de nuevo un elemento en D' , del cual depende la variable. La definición es como sigue:

$$X_r(sc) = g(r) + \frac{k(r, s)}{p(r, s)} X_s(c) \quad (3.6)$$

Las funciones g y k que aparecen aquí son las mismas que intervienen en la ecuación (1.27), la ecuación general de segundo orden. Necesitamos extender la definición de estas funciones para tener en cuenta el estado de absorción. Por definición, hacemos $g(a) = 0$ y $k(a, a) = k(a, r) = k(r, a) = 0$, para cualquier r en D . El valor medio de la variable aleatoria X_r con respecto a la medida de probabilidad U_r es:

$$E(X_r) = \int_C X_r(c) dU_r(c) = f(r)$$

donde f es la función solución de (1.27). Este es un resultado importante, puesto que permite usar métodos de Monte-Carlo para obtener estimaciones del valor de f para un elemento r . Esta clase de estimadores han sido ampliamente usados en problemas de transporte de neutrones y en Informática Gráfica.

¹si no se introdujese esta condición, entonces si U_r cumple la ecuación diferencial, kU_r también la cumpliría, para cualquier k real no nulo, y por tanto habría infinitas soluciones.

Con el objetivo de demostrar lo anterior (y posteriormente analizar la varianza) definimos el conjunto de funciones E_n con $n \geq 1$ como:

$$E_n(r) = E(X_r^n)$$

vemos que cada E_n es una función de valor real definida en el dominio D' , que equivale al momento n -ésimo de la variable aleatoria X_r . Nosotros estamos interesados especialmente en la funciones E_1 (la media) y en E_2 (la media de los cuadrados, que interviene en la varianza). Con respecto a la media, puede obtenerse como sigue:

$$\begin{aligned} E_1(r) &= E(X_r) \\ &= \int_C X_r(sc) dU_r(sc) \\ &= \int_C \left[g(r) + \frac{k(r, s)}{p(r, s)} X_s(c) \right] dU_r(sc) \\ &= \int_C g(r) dU_r(sc) + \int_C \frac{k(r, s)}{p(r, s)} X_s(c) dU_r(sc) \\ &= g(r) \int_C dU_r(sc) + \int_{D' \times C} k(r, s) X_s(c) dm'(s) dU_s(c) \\ &= g(r) + \int_{D'} k(r, s) \left[\int_C X_s(c) dU_s(c) \right] dm'(s) \\ &= g(r) + \int_{D'} k(r, s) E_1(s) dm'(s) \end{aligned} \quad (3.7)$$

La integración que se lleva a cabo en la ecuación (3.7) puede limitarse a D en lugar de D' , ya que $k(r, a) = 0$. Ahora, reescribimos (3.7):

$$E_1(r) = g(r) + \int_D k(r, s) E_1(s) dm(s) \quad (3.8)$$

Obsérvese que la ecuación de arriba tiene la misma forma que (1.27). Como esta última tiene un única solución, entonces, necesariamente se cumple que $E_1 = f$. Por lo tanto, para todos los puntos r , tenemos que $E(X_r) = f(r)$ lo cual demuestra que el método de muestreo usado no presenta sesgo.

3.2.3 Análisis de la varianza

Aunque la variable aleatoria introducida proporciona estimaciones no sesgadas de la función desconocida f , este estimador puede llegar a ser impracticable debido a una alta varianza. En esta sección analizamos la varianza de la variable aleatoria X_r . Su expresión es la siguiente:

$$\begin{aligned} Var(X_r) &= \int_C [X_r(c) - E(X_r)]^2 dU_r(c) \\ &= E(X_r^2) - E^2(X_r) \\ &= E_2(r) - f^2(r) \end{aligned} \quad (3.9)$$

Definimos explícitamente la función $V(r) = \text{Var}(X_r)$. También usaremos un operador integral lineal \mathcal{U} , definido como sigue:

$$\mathcal{U}F(r) = \int_{D'} \frac{k^2(r, s)}{p(r, s)} F(s) dm'(s) \quad (3.10)$$

En lo que sigue, probaremos que la función V puede obtenerse como la solución de una ecuación integral de segundo orden que involucra al operador \mathcal{U} . Para lograr esto, desarrollamos la expresión de $E_2(r)$:

$$\begin{aligned} E_2(r) &= E(X_r^2) \\ &= \int_C X_r^2(sc) dU_r(sc) \\ &= \int_C \left[g(r) + \frac{k(r, s)}{p(r, s)} X_s(c) \right]^2 dU_r(sc) \\ &= \int_C \left[g^2(r) + 2g(r) \frac{k(r, s)}{p(r, s)} X_s(c) + \frac{k^2(r, s)}{p^2(r, s)} X_s^2(c) \right] dU_r(sc) \end{aligned} \quad (3.11)$$

Si realizamos un estudio separado de los tres sumandos de la última integral, obtenemos, para el primero de ellos, lo siguiente:

$$\int_C g^2(r) dU_r(sc) = g^2(r) \int_C dU_r(sc) \quad (3.12)$$

$$= g^2(r) \quad (3.13)$$

Con respecto al segundo término de (3.11), podemos reescribirlo como sigue:

$$\int_C 2g(r) \frac{k(r, s)}{p(r, s)} X_s(c) dU_r(sc) = 2g(r) \int_C \frac{k(r, s)}{p(r, s)} X_s(c) dU_r(sc) \quad (3.14)$$

Usando las mismas sustituciones que convierten (3.7) en (3.7), encontramos que la integral (3.14) puede convertirse en:

$$2g(r) \int_D k(r, s) f(s) dm(s) = 2g(r) \mathcal{A}f(r) \quad (3.15)$$

Ahora, expandimos el tercer término en (3.11):

$$\int_C \frac{k^2(r, s)}{p^2(r, s)} X_s^2(c) dU_r(sc) \quad (3.16)$$

$$= \int_{D \times C} \frac{k^2(r, s)}{p^2(r, s)} X_s^2(c) [p(r, s) dm(s) dU_s(c)] \quad (3.17)$$

$$= \int_D \frac{k^2(r, s)}{p(r, s)} \left[\int_C X_s^2(c) dU_s(c) \right] dm(s) \quad (3.18)$$

$$= \int_D \frac{k^2(r, s)}{p(r, s)} E_2(s) dm(s) \quad (3.19)$$

$$= \mathcal{U}E_2(r) \quad (3.20)$$

Uniendo de nuevo los tres sumandos de (3.13), (3.15) y (3.20), podemos reescribir (3.11) de una forma simplificada:

$$E_2(r) = g^2(r) + 2g(r)\mathcal{A}f(r) + \mathcal{U}M(r)$$

si ponemos esta expresión de forma funcional, obtenemos:

$$E_2 = g^2 + 2g\mathcal{A}f + \mathcal{U}E_2 \quad (3.21)$$

$$= (g + \mathcal{A}f)^2 - (\mathcal{A}f)^2 + \mathcal{U}E_2 \quad (3.22)$$

$$= f^2 - (\mathcal{A}f)^2 + \mathcal{U}E_2 \quad (3.23)$$

con este último resultado, se puede encontrar una ecuación relacionada para V

$$\begin{aligned} V &= E_2 - f^2 \\ &= \mathcal{U}E_2 - (\mathcal{A}f)^2 \\ &= \mathcal{U}(V + f^2) - (\mathcal{A}f)^2 \\ &= \mathcal{U}f^2 - (\mathcal{A}f)^2 + \mathcal{U}V \end{aligned}$$

Aquí hemos usado la identidad $\mathcal{A}f = f - g$. Ahora definimos la función V_e como

$$V_e = \mathcal{U}f^2 - (\mathcal{A}f)^2 \quad (3.24)$$

y obtenemos la siguiente ecuación integral cuya solución es la función de varianzas V

$$V = V_e + \mathcal{U}V$$

siguiendo los mismos razonamientos que se hicieron para la ecuación integral (1.27), la anterior ecuación solo tendrá solución (esto es, la función V tendrá valores finitos) cuando $|\mathcal{U}|_1 < 1$, ya que en otro caso la norma de la serie de funciones $\mathcal{U}^n V_e$ divergiría al crecer n . Una condición necesaria y suficiente para que $|\mathcal{U}|_1 < 1$ es la siguiente

$$\int_D \frac{k^2(r, s)}{p(r, s)} dm(s) < 1 \quad (3.25)$$

en todos los puntos r del dominio D . Por tanto, la anterior debe considerarse como una restricción adicional de la función de transición p .

3.2.4 Probabilidad de transición proporcional al núcleo.

En la sección previa, se introdujo una expresión para la varianza cuando se usa una función de probabilidad de transición arbitraria, en un dominio arbitrario D . En esta subsección, aplicamos estos resultados al path-tracing, en el contexto de la iluminación global. La función de probabilidad de transición más usual es proporcional a la función K , en lo que se ha denomina usualmente como *muestreo por importancia* (*importance sampling*). Por lo tanto, se cumple que $p(r, s) = ck(r, s)$, donde c es una constante. La opción mas simple es hacer c igual a 1. Entonces $p(r, s) = k(r, s)$ para todos los s en D , y obtenemos

$$\int_{D'} p(r, s) dm'(s) = p(r, a) + \int_D k(r, s) dm(s)$$

La normalización de la función p , como se establece en la igualdad (3.1) esta asegurada ya que $m_k < 1$, como se detalló en el primer capítulo. Por lo tanto

$$\int_D k(r, s) dm(s) \leq m_k < 1$$

para todos los r y s en D . El valor $p(r, a)$ esta determinado por la igualdad (3.1), de la cual deducimos

$$p(r, a) = 1 - \int_D k(r, s) dm(s)$$

ahora, la función p esta plenamente determinada. Es fácil ver que esta p cumple las restricciones (3.1) hasta (3.4), y además también (3.25). Por lo tanto, para esta p la varianza es finita. Para obtener una expresión de V en este caso, desarrollamos primero $\mathcal{U}h$ para una función arbitraria h

$$\begin{aligned} \mathcal{U}h &= \int_D \frac{k^2(r, s)}{p(r, s)} h(s) dm(s) \\ &= \int_D k(r, s) h(s) dm(s) \\ &= \mathcal{A}h \end{aligned}$$

y deducimos por lo tanto que los operadores \mathcal{U} y \mathcal{A} coinciden, y por lo tanto la varianza es

$$V = V_e + \mathcal{A}V \quad (3.26)$$

donde

$$V_e = \mathcal{A}f^2 - (\mathcal{A}f)^2$$

como sabemos que $|\mathcal{A}|_1 < 1$, podemos por tanto escribir la ecuación (3.26) de la siguiente forma:

$$V = \mathcal{A}^+ V_e = \mathcal{A}^+ (\mathcal{A}f^2 - (\mathcal{A}f)^2) \quad (3.27)$$

además, usando (3.23) obtenemos que $E_2 = (f^2 - (\mathcal{A}f)^2) + \mathcal{T}E_2$ y por lo tanto

$$E_2 = \mathcal{A}^+ (f^2 - (\mathcal{A}f)^2) \quad (3.28)$$

3.2.5 Estimación de productos escalares o *funcionales*.

En la subsección anterior, hemos obtenido la varianza involucrada en las estimaciones puntuales de la función f . Sin embargo, lo mas usual en Iluminación Global es el cálculo de productos escalares o funciones de dicha función, esto es, el cálculo de $\langle f | h \rangle$, donde f es la función incógnita y h es una función integrable arbitraria en el dominio de D , tal y como se establece en el segundo capítulo. Esto puede llevarse a cabo igualmente mediante métodos de Monte-Carlo.

Con el objetivo de formalizar el método, definiremos una nueva variable aleatoria X_h cuya media es $\langle f | h \rangle$, de la forma

$$X_h(r) = \frac{h(r)}{p_0(r)} X_r(c) \quad (3.29)$$

donde p_0 una función de densidad de probabilidad con dominio en D , cumpliendo:

$$h(r) > 0 \Rightarrow p_0(r) > 0 \quad (3.30)$$

$$\int_D p_0(r) dm(r) = 1 \quad (3.31)$$

La función p_0 induce una medida de probabilidad U_h en C (el espacio de todas las cadenas), que cumple

$$dU_h(rc) = p_0(r) dm(r) dU_r(c) \quad (3.32)$$

Ahora, expandimos el valor medio de X_h con respecto a la medida de probabilidad U_h

$$\begin{aligned} E(X_h) &= \int_C X_h(rc) dU_h(rc) \\ &= \int_{C \times D} h(r) X_r(c) dm(r) dU_r(c) \\ &= \int_D h(r) \left[\int_C X_r(c) dU_r(c) \right] dm(r) \\ &= \int_D h(r) E(X_r) dm(r) \\ &= \int_D h(r) L(r) dm(r) \\ &= \langle f | h \rangle \end{aligned}$$

Así, vemos que la variable aleatoria X_h puede ser muestreada con el objetivo de obtener estimaciones de $\langle f | h \rangle$. Lógicamente, la eficiencia del método depende de la varianza. Para obtener una expresión de $Var(X_h)$, comenzamos expandiendo $E(X_h^2)$

$$\begin{aligned} E(X_h^2) &= \int_D X_h^2(rc) dU_h(rc) \\ &= \int_{C \times D} X_h^2(rc) p_0(r) dm(r) dU_r(c) \\ &= \int_D \left[\int_C X_r^2(c) dU_r(c) \right] \frac{h^2(r)}{p_0(r)} dm(r) \\ &= \int_D E(X_r^2) \frac{h^2(r)}{p_0(r)} dm(r) \\ &= \langle E_2 | h^2/p_0 \rangle \end{aligned}$$

y, por lo tanto, la varianza es

$$Var(X_h) = E(X_h^2) - E^2(X_h) \quad (3.33)$$

$$= \langle E_2 | h^2/p_0 \rangle - \langle f | h \rangle^2 \quad (3.34)$$

El valor $p_0(r)$ indica la probabilidad de que escogamos una cadena que comience en r . A partir de aquí, las siguientes transiciones de la cadena están gobernadas por p , la función de probabilidad de transición.

Cuando esto es posible, lo más frecuente es seleccionar p_0 proporcional a h , pues, a falta de más información a priori, esta es la selección que produce menor varianza. En este caso tenemos que $p_0 = h/c$ donde c es una constante introducida para normalizar, y por lo tanto $c = \int_D h(r) dm(r)$. Con este ejemplo concreto de p_0 , la varianza de X_h es

$$\text{Var}(X_h) = c \langle E_2 | h \rangle - \langle f | h \rangle^2$$

Si, además de elegir este p_0 se usan probabilidades de transición proporcionales al núcleo, como se ha explicado antes, entonces obtenemos

$$\text{Var}(X_h) = c \langle \mathcal{A}(f^2 - (\mathcal{A}f)^2) | h \rangle - \langle f | h \rangle^2 \quad (3.35)$$

$$= c \langle f^2 - (\mathcal{A}f)^2 | (\mathcal{A}^*)^+ h \rangle - \langle f | h \rangle^2 \quad (3.36)$$

3.2.6 Una probabilidad de transición ideal con varianza nula.

De lo expuesto en la sección anterior podemos deducir cuando la varianza de la variable aleatoria X_r es cero. Esto solo puede ocurrir si la función V_e es cero en todos los puntos.

A partir de (3.24) obtenemos que $V_e = 0$ si y solo si $\mathcal{U}f^2 = (\mathcal{A}f)^2$. Seleccionamos la siguiente función de probabilidad de transición

$$p(r, s) = \begin{cases} K(r, s) \frac{f(s)}{\mathcal{A}f(r)} & \text{si } \mathcal{A}f(r) > 0 \\ 0 & \text{si } \mathcal{A}f(r) = 0 \end{cases} \quad (3.37)$$

es fácil demostrar que el anterior ejemplo de p cumple las restricciones (3.1) hasta (3.4). Con respecto a la varianza, podemos expandir el valor de $\mathcal{U}f^2$:

$$\begin{aligned} \mathcal{U}f^2(r) &= \int_D \frac{K^2(r, s)}{p(r, s)} f^2(s) ds \\ &= \int_D K(r, s) \mathcal{A}f(r) f(s) ds \\ &= \mathcal{A}f(r) \int_D K(r, s) f(s) ds \\ &= (\mathcal{A}f(r))^2 \end{aligned}$$

Como esperábamos, hemos deducido que $\mathcal{U}f^2 = (\mathcal{A}f)^2$, y por tanto $V_e = 0$ cuando p se define como en (3.37). Obviamente, no se pueden seleccionar cadenas con esta p , ya que su definición incluye los valores desconocidos $f(s)$ y $\mathcal{A}f(r)$.

3.2.7 Aplicaciones a la Iluminación Global.

Los resultados obtenidos anteriormente permiten obtener la varianza en los estimadores, para el caso de la ecuación genérica (1.27). Existen varios métodos de Monte-Carlo que constituyen instancias de la solución genérica. La obtención de la varianza en el caso general permite, por lo tanto, obtenerla estos casos particulares.

Cálculo de la intensidad de un pixel.

En este caso, el problema es el cálculo de la proyección de la función imagen, como se estableció en la sección 2. Recuérdese que el conjunto de funciones $\mathbf{W} = \{W_{e1}, \dots, W_{en}\}$ se usa para proyectar la función de radiancia y obtener una imagen discreta. La intensidad de cada pixel es $l_i = \langle W_{ei} | L \rangle$. Por lo tanto, el problema puede establecerse como el cálculo del producto escalar de la función de radiancias (solución de una ecuación integral) y otra función, con lo cual se puede usar el *path-tracing*.

Esto puede llevarse a cabo muestreando la variable aleatoria X_h , tal y como se define en (3.29), con $h = W_{ei}$, $\mathcal{A} = \mathcal{T}$, $g = L_e$, $f = L$, $\mathcal{A}f = \mathcal{T}L = L_r$ y $m = \mu$. Asumiremos que W_{ei} está normalizada en el sentido de que $\int_D W_{ei}(\mathbf{r}) d\mu(\mathbf{r}) = 1$, así que podemos usar $p_0 = W_{ei}$, con $c = 1$. También se suelen usar probabilidades de transición proporcionales al núcleo, y por lo tanto $p = K$ y $\mathcal{T} = \mathcal{U}$. Tenemos que $E(X_h) = l_i$, y así el método es no sesgado. Con respecto a la varianza obtenemos, de la ecuación (3.28), (nótese que $\mathcal{U} = \mathcal{T}$) el valor de E_2

$$E_2 = \mathcal{T}^+ (L^2 - L_r^2)$$

usando $p_0 = h = W_{ei}$, la ecuación (3.36) se convierte, en este caso concreto, en:

$$\begin{aligned} Var(X_h) &= \langle L^2 - L_r^2 | (\mathcal{T}^*)^+ W_{ei} \rangle - \langle L | W_{ei} \rangle^2 \\ &= \langle L^2 - L_r^2 | W_i \rangle - \langle L | W_{ei} \rangle^2 \end{aligned}$$

nótese que $W_i = (\mathcal{T}^*)^+ W_{ei}$, es el potencial (o importancia) inducido por el i -ésimo pixel de la imagen, como se define en la sección 2.1.

Cálculo de la radiosidad de un trozo.

En este caso, el objetivo es el cálculo de la radiosidad de un trozo. Para esto se pueden usar dos métodos: bien se pueden enviar partículas desde las fuentes de luz, como se explica en 2.5.2, bien mediante *path-tracing* con caminos originados en el trozo

Asumimos que L_e tiene una distribución difusa (esto es, $L_e(x, w)$ es independiente de w) y que \mathcal{T} es operador de transporte difuso (por tanto $\mathcal{T} = \mathcal{D}$). En estas circunstancias, (como se puede ver en la sección 1.3.2) el transporte de radiancias está modelado por la ecuación de radiosidad. Recuérdese que dicha ecuación tiene como dominio el conjunto de puntos de las superficies, por lo tanto este será el dominio de nuestro problema (y no el conjunto de rayos).

La radiosidad de un trozo es el valor medio de la función de radiosidad en toda su área. Llamaremos A al trozo de superficie, y b a su radiosidad. Sabemos que:

$$b = \frac{1}{|A|} \int_A B(x) dA(x)$$

donde dA representa la medida de área. Si definimos la siguiente función W_e (sobre el conjunto de puntos de las superficies)

$$W_e(x) = \begin{cases} \frac{1}{|A|} & \text{si } x \in A \\ 0 & \text{en otro caso} \end{cases} \quad (3.38)$$

podemos escribir b como un producto escalar

$$b = \int_S W_e(x) B(x) dA(x) = \langle B | W_e \rangle$$

Sea $W = (\mathcal{T}^*)^+ W_e$, esto es, W es la función de potencial. Podemos establecer estas dos ecuaciones adjuntas:

$$B = B_e + \mathcal{D}B \quad (3.39)$$

$$W = W_e + \mathcal{D}^*W \quad (3.40)$$

y también expresar b de dos formas

$$b = \langle B | W_e \rangle = \langle \mathcal{D}^+ B_e | W_e \rangle = \langle B_e | (\mathcal{D}^*)^+ W_e \rangle = \langle B_e | W \rangle$$

existen dos formas distintas de estimar b , según cual de las dos formas de expresarlo usemos. Cada una de estas dos formas nos lleva a métodos distintos con varianzas distintas, que examinamos a continuación:

Path-tracing. En este caso, para obtener b usaremos la ecuación (3.39). Por lo tanto, tenemos que $f = B$, $\mathcal{A} = \mathcal{D}$, y $h = W_e$. En este caso b se escribe como $b = \langle B | W_e \rangle$. La varianza se obtiene instanciando (3.36) con $c = 1$ (recordar que W_e está normalizado):

$$\begin{aligned} Var(X_h) &= \langle B^2 - B_r^2 | (\mathcal{D}^*)^+ W_e \rangle - b^2 \\ &= \langle B^2 - B_r^2 | W \rangle - b^2 \end{aligned}$$

Fotosimulación. En este caso, usamos la ecuación (3.40). Por lo tanto, tenemos $f = W$, $\mathcal{A} = \mathcal{T}^*$, y $h = B_e$. El valor b se expresa ahora como $b = \langle B_e | W \rangle$. En este caso obtenemos algoritmos de fotosimulación o trazado de partículas. Al no estar B_e normalizada, aquí no se cumple que $c = 1$, por lo tanto tenemos que $c = \int_D B_e(x) dA(x)$. Esta es la energía total emitida en la escena, dividida por π . La varianza se obtiene de nuevo por instanciación de (3.36)

$$\begin{aligned} Var(X_h) &= \langle W^2 - W_r^2 | \mathcal{D}^+ B_e \rangle \phi_e - b^2 \\ &= \langle W^2 - W_r^2 | B \rangle \phi_e - b^2 \end{aligned}$$

donde $W_r = \mathcal{T}^*W$.

3.3 Path-Tracing con muestreo directo de fuentes.

Como se expuso en el capítulo 2 sobre algoritmos de Iluminación Global, el muestreo directo de las fuentes de luz es una mejora al Path-Tracing simple desde el observador. En este algoritmo cada impacto en una superficie conlleva una estimación de la radiancia proveniente de las fuentes de luz y reflejada en el punto de impacto. Dicha estimación se realiza a su vez con métodos de Monte-Carlo. Este método fue propuesto en [Kajiya86]. Después, se han publicado varias funciones alternativas de probabilidad de muestreo [Shirley96]. En esta sección veremos como se puede caracterizar la varianza de este método. Primero introducimos la densidad de cadenas apropiadas, y después mostramos que el método es no sesgado. Finalmente, veremos como a partir de la formulación general se derivan expresiones de la varianza para varios métodos de Iluminación Global.

3.3.1 Densidad de cadenas extendidas.

Ahora, introducimos una medida de probabilidad para las cadenas que se usan en el muestro directo de las fuentes de luz. Esta densidad es parecida a la que aparece en la sección 3.2.1, con la diferencia de que esta se define en un espacio extendido. Consideremos el espacio C' de las cadenas infinitas de pares de elementos del conjunto $D \times D'$, esto es, si $c \in C'$ entonces $c = (r_1, r_2, \dots)$ donde $r_i \in D \times (D \cup \{a\})$ (la segunda componente puede ser el estado absorción, no así la primera). También usaremos la notación rc para nombrar la cadena extendida formada al anteponer r a c , donde $r \in D \times D'$ y $c \in C'$. Por lo tanto, si $c = (r_1, r_2, \dots)$ entonces $rc = (r, r_1, r_2, \dots) \in C'$. Cada estado s de una cadena extendida es un par de dos elementos llamados s_d y s_i , respectivamente, esto es $s = (s_d, s_i)$. El elemento s_d se usa para el muestreo directo de las fuentes de luz (normalmente es un punto en dicha fuente), mientras que s_i es el siguiente elemento en la cadena, que se usa para muestro de la luz indirecta.

Supongamos que el estado actual de una cadena es $q = (r_d, r)$. La probabilidad de que el siguiente estado sea $s = (s_d, s_i)$ es independiente de r_d y solo depende de r . Además, dicha probabilidad puede descomponerse en un producto de las dos componentes de s (es separable con respecto a s). Podemos expresar, por tanto, dicha probabilidad como

$$p_d(r, s_d) p(r, s_i) dm(s_d) dm(s_i)$$

donde $p_d(r, s_d)$ es la probabilidad de seleccionar s_d como punto para muestreo directo, mientras que $p(r, s_i)$ es la probabilidad de seleccionar s_i para continuar la cadena y muestrear la componente indirecta. En la figura 3.2 se puede observar una cadena extendida que comienza en r (es el rayo que se dirige al observador) y continua en (s_d, s_i) . El rayo s_d tiene origen en la fuente de luz, mientras que s_i sirve para continuar dicha cadena. En cada impacto con las superficies, se muestrea la fuente de luz y se continua la cadena. Con todo esto podemos definir la medida de probabilidad Q_r , para cada $r \in D'$, como

$$dQ_r(sc) = p_d(r, s_d) dm(s_d) p(r, s_i) dm(s_i) dQ_{s_i}(c) \quad (3.41)$$

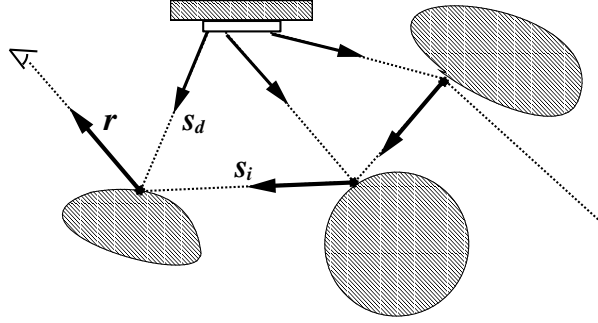


Figura 3.2: Un ejemplo de una cadena extendida.

El valor $dQ_r(sc)$ es la densidad de probabilidad de realizar una transición al estado $s = (s_d, s_i)$ y después recorrer todos los estados de la cadena extendida c , sabiendo que el estado actual tiene a r como segunda componente.

Esta función p cumple las restricciones que aseguran que es una función de probabilidad válida para nuestros propósitos, esto es, p cumple las propiedades (3.1) and (3.2). La función p_d también cumple una restricciones parecidas, aunque no exactamente iguales:

$$1 = \int_D p_d(r, s) dm(s)$$

$$k(r, s)g(s) > 0 \Rightarrow p_d(r, s) > 0$$

Se supone que las cadenas extendidas que se usan en esta sección siguen la distribución de probabilidad inducida por la anterior medida Q_r .

3.3.2 Una variable aleatoria sobre cadenas extendidas.

Una vez introducido Q_r podemos definir la variable aleatoria Y_r , para todos los elementos $r \in D$, y que se evalúa sobre cadenas extendidas:

$$Y_r(sc) = Z_r(s_d) + \frac{k(r, s_i)}{p(r, s_i)} Y_{s_i}(c) \quad (3.42)$$

donde Z_r es, a su vez, otra variable aleatoria, definida como sigue:

$$Z_r(s_d) = \frac{k(r, s_d)}{p_d(r, s_d)} g(s_d) \quad (3.43)$$

la variable Z_r se define en D en lugar de C' . Dicha variable aleatoria está definida respecto a la densidad de probabilidad $p_d(r, s) dm(s_d)$. Su media es la siguiente:

$$E(Z_r) = \int_D Z_r(s_d) p_d(r, s_d) dm(s_d)$$

$$\begin{aligned}
&= \int_D k(r, s_d) g(s_d) dm(s_d) \\
&= (\mathcal{A}g)(r)
\end{aligned}$$

Ahora podemos obtener la media de la variable Y_r con respecto de la medida Q_r :

$$E(Y_r) = \int_{C'} Y_r(sc) dQ_r(sc) \quad (3.44)$$

$$= \int_{C'} Z_r(s_d) dQ_r(sc) + \int_{C'} \frac{k(r, s_i)}{p(r, s_i)} Y_{s_i}(c) dQ_r(sc) \quad (3.45)$$

el primer término de arriba puede reescribirse como sigue:

$$\begin{aligned}
&\int_{C'} Z_r(s_d) dQ_r(sc) \\
&= \int_{D^2 \times C'} Z_r(s_d) p_d(r, s_d) dm(s_d) p(r, s_i) dm(s_i) dQ_{s_i}(c) \\
&= \int_D Z_r(s_d) \left(\int_D p(r, s_i) \left[\int_{C'} dQ_{s_i}(c) \right] dm(s_i) \right) p_d(r, s_d) dm(s_d) \\
&= \int_D Z_r(s_d) p_d(r, s_d) dm(s_d) \\
&= E(Z_r) \\
&= (\mathcal{A}g)(r)
\end{aligned}$$

mientras que el segundo término en (3.45) se puede a su vez expandir como:

$$\begin{aligned}
&\int_{C'} \frac{k(r, s_i)}{p(r, s_i)} Y_{s_i}(c) dQ_r(sc) \\
&= \int_{D^2 \times C'} k(r, s_i) Y_{s_i}(c) p_d(r, s_d) dm(s_d) dm(s_i) dQ_{s_i}(c) \\
&= \int_D k(r, s_i) \left[\int_{C'} Y_{s_i}(c) \left(\int_{D'} p(r, s_d) dm(s_d) \right) dQ_{s_i}(c) \right] dm(s_i) \\
&= \int_D k(r, s_i) \left[\int_{C'} Y_{s_i}(c) dQ_{s_i}(c) \right] dm(s_i) \\
&= \int_D k(r, s_i) E(Y_{s_i}) dm(s_i) \\
&= \int_D k(r, s_i) E'_1(s_i) dm(s_i) \\
&= (\mathcal{A}E'_1)(r)
\end{aligned}$$

donde hemos definido $E'_1(r) = E(Y_r)$ a semejanza con la sección anterior. Reuniendo de nuevo los dos términos de (3.45) obtenemos:

$$E(X_r) = E'_1(r) = (\mathcal{A}g)(r) + (\mathcal{A}E'_1)(r)$$

o, en forma funcional

$$E'_1 = \mathcal{A}g + \mathcal{A}E'_1$$

y, por lo tanto, deducimos que

$$E'_1 = \mathcal{A}^+(\mathcal{A}g) = \mathcal{A}(\mathcal{A}^+g) = \mathcal{A}f$$

aquí hemos usado la identidad $\mathcal{A}^+\mathcal{A} = \mathcal{A}\mathcal{A}^+$, que se deriva fácilmente de la definición de \mathcal{A}^+ . Por lo tanto tenemos que $E(Y_r) = (\mathcal{A}f)(r)$ y podemos muestrear la variable Y_r para calcular por Monte-Carlo el valor $(\mathcal{A}f)(r)$. Sin embargo, el objetivo es obtener valores de f en lugar de $\mathcal{A}f$, y esto puede hacerse sumando el valor $g(r)$ tras el muestro, o, lo que es lo mismo, muestreando la variable Y'_r , que se define como

$$Y'_r(c) = g(r) + Y_r(c) \quad (3.46)$$

y cuya media se obtiene a partir de la media de Y_r :

$$E(Y'_r) = g(r) + E(Y_r) = g(r) + (\mathcal{A}f)(r) = f(r)$$

Obviamente, se cumple que $\text{Var}(Y_r) = \text{Var}(Y'_r)$. En el siguiente apartado se analiza esta varianza, al igual que se analizó la varianza de X_r .

3.3.3 La varianza de Y_r

Con respecto a la varianza, como se ha establecido antes, se cumple que $\text{Var}(Y_r) = \text{Var}(Y'_r)$, así que analizaremos la primera de ellas. En coherencia con secciones anteriores, definimos $E'_2(r) = E(Y_r^2)$. Este valor puede expandirse como la suma de tres términos, y cada uno de ellos se puede simplificar, como veremos a continuación. Expandiendo E_2 obtenemos:

$$\begin{aligned} E_2(r) &= E(Y_r^2) \\ &= \int_{C'} Y_r^2(sc) dQ_r(sc) \\ &= \int_D Z_r^2(s_d) p_d(r, s_d) dm(s_d) + \int_D \frac{k^2(r, s_i)}{p(r, s_i)} E(Y_{s_i}^2) dm(s_i) \\ &\quad + 2 \int_D Z_r(s_d) \frac{k(r, s_i)}{p(r, s_i)} Y_{s_i}(c) dQ_r(sc) \end{aligned}$$

Los dos primeros términos de arriba se pueden escribir como $E(Z_r^2) + (\mathcal{U}E'_2)(r)$. El tercero se expande como sigue:

$$\begin{aligned} &2 \int_D Z_r(s_d) \frac{k(r, s_i)}{p(r, s_i)} Y_{s_i}(c) dQ_r(sc) \\ &= 2 \left[\int_D Z_r(s_d) p_d(r, s_d) dm(s_d) \right] \left[\int_D k(r, s_i) \left[\int_{C'} Y_{s_i}(c) dQ_{s_i}(c) \right] dm(s_i) \right] \\ &= 2 E(Z_r) (\mathcal{A}E'_1)(r) \\ &= 2 (\mathcal{A}g)(r) (\mathcal{A}^2 f)(r) \end{aligned}$$

Uniendo de nuevo los tres términos originales, obtenemos una expresión de E'_2 como:

$$E'_2(r) = E(Z_r^2) + (\mathcal{U}E'_2)(r) + 2(\mathcal{A}g)(r)(\mathcal{A}^2f)(r) \quad (3.47)$$

el valor $E(Z_r^2)$ puede escribirse como $(\mathcal{F}g^2)(r)$ donde \mathcal{F} es el operador integral con núcleo k^2/p_d , esto es:

$$(\mathcal{F}g^2)(r) = \int_D \frac{k^2(r, s_d)}{p_d(r, s_d)} g^2(s_d) dm(s_d)$$

Por simplicidad, definiremos $f_n = \mathcal{A}^n f$ y $g_n = \mathcal{A}^n g$, para todo $n \geq 0$. Aplicando \mathcal{A}^n a ambos lados de (1.42), obtenemos que $f_n = g_n + f_{n+1}$. Como $g_n \geq 0$, entonces también se cumple que $f_{n+1} \leq f_n$. Con estas definiciones podemos reescribir la última ecuación como sigue:

$$\begin{aligned} E'_2 &= \mathcal{F}g^2 + 2(\mathcal{A}g)(\mathcal{A}^2f) + \mathcal{U}E'_2 \\ &= \mathcal{F}g^2 + 2g_1f_2 + \mathcal{U}E'_2 \\ &= \mathcal{F}g^2 + f_1^2 - g_1^2 - f_2^2 + \mathcal{U}E'_2 \end{aligned}$$

La anterior es una ecuación integral, ya que E'_2 aparece a ambos lados de la igualdad. Si suponemos que el operador \mathcal{U} cumple las restricciones dadas en la desigualdad (3.25) entonces tenemos que $|\mathcal{U}|_1 < 1$, y por lo tanto existe el operador \mathcal{U}^+ . Usando dicho operador, ponemos finalmente E'_2 como:

$$E'_2 = \mathcal{U}^+ [\mathcal{F}g^2 - g_1^2 + f_1^2 - f_2^2] \quad (3.48)$$

Si definimos la función V' como $V'(r) = \text{Var}(Y_r)$ entonces tenemos que $E'_2(r) = V'(r) + E^2(Y_r) = V'(r) + (f_1(r))^2$. Sustituyendo en la ecuación (3.47) la función E'_2 por este valor y despejando V' obtenemos

$$\begin{aligned} V' &= \mathcal{F}g^2 - g_1^2 - f_2^2 + \mathcal{U}(V' + f_1^2) \\ &= \mathcal{F}g^2 - g_1^2 - f_2^2 + \mathcal{U}f_1^2 + \mathcal{U}V' \end{aligned}$$

o lo que es lo mismo:

$$V' = \mathcal{U}^+ [\mathcal{F}g^2 - g_1^2 - f_2^2 + \mathcal{U}f_1^2] \quad (3.49)$$

Por lo tanto vemos que la variable aleatoria Y_r presenta una varianza V' que es la solución de una ecuación integral de segundo orden.

En el caso de que la función de probabilidad de transición p sea proporcional al núcleo k , entonces tenemos (por lo visto en la sección anterior) que $\mathcal{A} = \mathcal{U}$, y por tanto lo tanto E'_2 y V' se escriben como:

$$E'_2 = \mathcal{A}^+ [\mathcal{F}g^2 - g_1^2 + f_1^2 - f_2^2] \quad (3.50)$$

$$V' = \mathcal{A}^+ [\mathcal{F}g^2 - g_1^2 - f_2^2 + \mathcal{A}f_1^2] \quad (3.51)$$

3.3.4 Estimación de productos escalares.

Como se ha dicho anteriormente, muchas veces el objetivo es calcular productos escalares de f con una función arbitraria h . Esto se puede llevar a cabo usando cadenas extendidas, mediante un esquema parecido al de las cadenas simples. A continuación definimos la medida de probabilidad y la variable aleatoria correspondiente a este algoritmo de muestreo.

Con respecto a la medida de probabilidad, la llamaremos Q_h y se encuentra definida en el espacio $D \times C'$. Esto quiere decir que se muestrea un elemento de D y a continuación una cadena. Por simplicidad escribiremos rc para denotar el elemento (r, c) del espacio $D \times C'$, donde $r \in D$ y c es una cadena extendida de C' . Al igual que antes, definiremos la nueva medida por su relación con la medida Q_r , como sigue:

$$dQ_h(rc) = p_0(r) dm(r) dQ_r(c) \quad (3.52)$$

donde p_0 cumple las propiedades establecidas en (3.30) y en (3.31). la función p_0 establece la probabilidad de seleccionar r como primer elemento en una cadena extendida. La variable aleatoria que usaremos será llamada Y_h que se evalúa también sobre elementos de $D \times C'$, como sigue:

$$Y_h(rc) = \frac{h(r)}{p_0(r)} Y'_r(c) \quad (3.53)$$

la media de Y_h con respecto Q_h se obtiene fácilmente:

$$\begin{aligned} E(Y_h) &= \int_{D \times C'} Y_h(rc) dQ_h(rc) \\ &= \int_D h(r) \left[\int_{C'} Y'_r(c) dQ_r(c) \right] dm(r) \\ &= \int_D h(r) E(Y'_r) dm(r) \\ &= \int_D h(r) f(r) dm(r) \\ &= \langle h | f \rangle \end{aligned}$$

con lo cual vemos que muestrear Y_h constituye un algoritmo sin sesgo para el cálculo de $\langle f | h \rangle$. Con respecto a la varianza, sabemos que $Var(Y_h) = E(Y_h^2) - \langle f | h \rangle^2$. Por tanto, desarrollaremos $E(Y_h^2)$:

$$\begin{aligned} E(Y_h^2) &= \int_{D \times C} Y_h^2(rc) dQ_h(rc) \\ &= \int_{D \times C} \frac{h^2(r)}{p_0(r)} Y_r'^2(c) dQ_h(rc) \\ &= \int_D \frac{h^2(r)}{p_0(r)} \left[\int_C Y_r'^2(c) dQ_r(c) \right] dm(r) \\ &= \int_D \frac{h^2(r)}{p_0(r)} E(Y_r'^2) dm(r) \end{aligned}$$

$$= \langle h^2/p_0 \mid E'_2 + f^2 - f_1^2 \rangle$$

donde hemos usado la igualdad $E(Y_r'^2) = E'_2(r) + f^2 - f_1^2$, que se obtiene fácilmente a partir de la definición de Y_r' dada en la igualdad (3.46).

Si suponemos que $\int_D h(r) dm(r) = 1$ (esto es, la función h esta normalizada), que p_0 es igual h en todos los puntos, y seleccionamos p proporcional a k entonces tenemos que $\mathcal{A} = \mathcal{U}$ y $h^2/p_0 = h$. En estas condiciones particulares, $E(Y_h^2)$ se puede escribir como sigue:

$$\begin{aligned} E(Y_h^2) &= \langle h \mid E'_2 + f^2 - f_1^2 \rangle \\ &= \langle h \mid E'_2 \rangle + \langle h \mid f^2 - f_1^2 \rangle \\ &= \langle (\mathcal{A}^+)^* h \mid \mathcal{F}g^2 - g_1^2 + f_1^2 - f_2^2 \rangle + \langle h \mid f^2 - f_1^2 \rangle \\ &= \langle l \mid \mathcal{F}g^2 - g_1^2 + f_1^2 - f_2^2 \rangle + \langle h \mid f^2 - f_1^2 \rangle \end{aligned}$$

donde $l = (\mathcal{A}^+)^* h$ es la solución de la ecuación $l = h + \mathcal{A}^* l$, adjunta a la ecuación original (1.42).

3.4 Conclusiones.

En este capítulo se analiza la varianza de los métodos de Monte-Carlo para ecuaciones integrales de segundo orden. Se presenta una función de medida de probabilidad sobre el espacio de todos los caminos posibles, lo cual permite explicar los métodos de Monte-Carlo como el muestreo de una variable aleatoria definida con respecto a dicha medida. Esto da la posibilidad además de expresar su media y la varianza como una integración en el espacio de todos los caminos posibles. A partir de aquí se caracteriza completamente la varianza de dichos métodos. En concreto, se obtiene dicha varianza como la solución de una ecuación integral de segundo orden.

Al aplicar ese resultado a varios problemas dentro del campo de la Iluminación Global, es fácil conocer la varianza exacta de los métodos usados en dichos problemas. En concreto, se caracteriza la varianza de la fotosimulación para el cálculo de radiosidades y del path-tracing para el cálculo de la intensidad de un pixel.

Finalmente, se obtiene un resultado completamente original que caracteriza la varianza cuando se usa path-tracing con muestreo directo de las fuentes, técnica habitual en Iluminación Global.

Capítulo 4

Un procedimiento mejorado de refinamiento final para radiosidad.

4.1 Introducción.

El cálculo de la radiosidad (como se ha detallado en el capítulo segundo) en entornos arbitrarios es un problema que se ha resuelto bien por métodos de elementos finitos, o bien por los métodos estocásticos detallados en los capítulos segundo y tercero. Los primeros de ellos pueden mejorarse si se usan conjuntos de funciones base jerárquicas [Gortler93], en lugar de funciones constantes no solapadas. La principal desventaja de este proceso es que produce una aproximación discreta (proyectada) de la función de radiosidad, que puede no capturar algunos detalles de la función exacta. En la figura 4.1 se observa un esquema del proceso. Primero se realizan cálculos de las transferencias de energía entre pares de trozos, y como resultado se obtiene un vector de radiosidades escalares B_i asociadas a cada uno de los trozos.

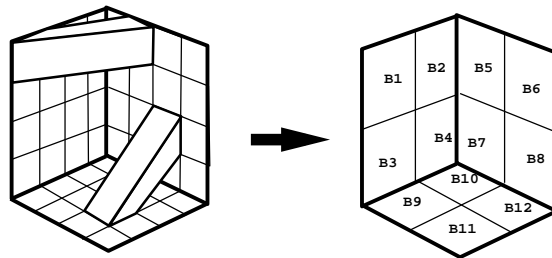


Figura 4.1: La radiosidad clásica por elementos finitos.

Esto puede resolverse usando un conjunto de funciones base del cual se sepa

a priori que aproxima mejor la función objetivo. Una de las soluciones propuestas es el uso del *discontinuity meshing* o mallado de discontinuidades, en el cual el dominio de cada base no incluye ninguna discontinuidad hasta un orden máximo predeterminado. Otra opción es el uso de la técnica de *shadow mask* o máscaras de sombras [Zatz93], para funciones base de orden superior a 0. Desafortunadamente, el cálculo de la malla de discontinuidades o las máscaras de sombras implica un alto coste para entornos con un número alto de superficies.

Con respecto a los métodos de Monte-Carlo, están basados en el muestreo de una determinada distribución de cadenas aleatorias [Spanier69, Kajiya86]. El proceso de cálculo conlleva cierta varianza en los estimadores, como se ha analizado en el capítulo anterior. En todos los casos, se necesitan un número muy alto de cadenas para reducir la varianza a niveles aceptables.

Con el objetivo de superar estas limitaciones, consideramos los métodos de dos pasadas, en los cuales la primera de ellas produce una aproximación a baja resolución de la función de radiosidad, y en la segunda pasada (también llamada pasada de visualización, *rendering pass*) dicha aproximación a baja resolución se mejora mediante un proceso de refinamiento final (o *final-gather*) a nivel de píxeles. El proceso de refinamiento final equivale al cálculo de la irradiancia en cada punto visible a través del plano de visión durante la fase de visualización. Esto supone una recolección de los valores de radiosidad de todos las posibles fuentes o reflectores del entorno sobre dicho punto (véase la figura 4.2)

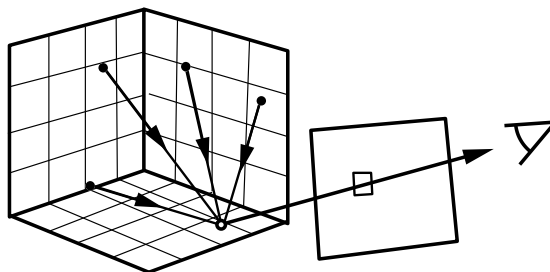


Figura 4.2: El paso de refinamiento final.

Varios artículos tratan este problema. En [Rushmeier88] se presentaba un método de Monte-Carlo, y en [Reichert92] se describía un sistema en el cual se usaban semicubos y hardware gráfico. Ambos métodos requieren una gran cantidad de cálculos en cada píxel.

Una mejora al esquema básico consiste en muestrear únicamente la luz directa, como se describe en [Shirley90], usando los valores almacenados en la primera pasada para aproximar la iluminación directa. Con el objetivo de mejorar más el algoritmo de muestreo es posible averiguar que reflectores o fuentes contribuyen más a la irradiancia en el punto objetivo, realizándose un mayor esfuerzo en el muestreo de dichas áreas. En [Kok91] se presentaba un método en el cual la primera pasada produce, para cada trozo, una lista de los otros trozos que se han clasificado como fuentes importantes para el primero. En la fase de visualización, esos otros trozos se

muestran directamente mediante *ray-casting* (o clasificación de rayos) adaptativo. Ward describe [Ward91] un método para evitar el muestreo de todas las fuentes, para escenas con muchas de ellas. También se puede usar un índice espacial, como se describe en [Zimmerman95, Shirley96]. Otras aproximaciones para el cálculo de la irradiancia incluyen el uso de Algoritmos Evolutivos [Beyer94], o geometría simplificada para la función fuente [Rushmeier93, Zimmerman95].

En este capítulo proponemos un método simple y eficiente de Monte-Carlo para el cálculo de la irradiancia en la segunda pasada. Para ello usamos información obtenida en la primera pasada, para averiguar la distribución de la irradiancia sobre cada trozo receptor con respecto al resto de trozos, considerados como trozos fuente. Mostramos como estos datos se pueden usar para construir una *pdf* válida, obteniéndose así un estimador sin sesgo con poca varianza. El método no necesita estructuras de datos complicadas, ni modificaciones a las técnicas estándar de cálculo de la función a baja resolución. El número de muestras que cada trozo fuente recibe es (aproximadamente), proporcional a la parte de irradiancia que produce en el trozo receptor donde se encuentra el punto. Dicho número de muestras puede reducirse gracias a la baja varianza, y de hecho puede llegar a ser mucho menor que el número de trozos. Así, usando esta *pdf* evitamos el muestreo explícito de cada trozo del entorno.

En las siguientes secciones, introducimos la notación y después damos detalles acerca de las varias distribuciones posibles para muestreo durante la fase de refinamiento final, incluyendo las distribuciones ponderadas que usan información obtenida en la primera pasada. Finalmente, se proporcionan imágenes de muestra que sirven para verificar el rendimiento de este método con respecto a otro método estándar.

4.2 Refinamiento final por Monte-Carlo.

Los métodos que usan refinamiento final para mejorar una función de radiosidad a baja resolución pueden expresarse como un cálculo de dos pasadas, usando la notación introducida en el capítulo 2:

$$\begin{aligned} (1) \quad B' &= (\mathcal{P}_{\mathbf{B}} \mathcal{D})^+ B_e \\ (2) \quad I &= \mathcal{P}_{\mathbf{W}} (\mathcal{D} B' + B_e) \end{aligned} \tag{4.1}$$

El primer paso es el cálculo de la versión proyectada de la función de radiosidad B' . Esto se realiza mediante el uso de la base \mathbf{B} , que produce una solución a baja resolución. En el segundo paso, para cada pixel, calculamos el primer punto visible x desde dicho pixel, y entonces se realiza un refinamiento final para obtener el valor $E_g(x)$ (que es una aproximación a la irradiancia), y añadimos la radiosidad emitida. Así la radiosidad se obtiene como:

$$B(x) = B_e(x) + \frac{\rho(x)}{\pi} E_g(x) \tag{4.2}$$

El problema del refinamiento final sobre los puntos visibles se puede establecer como sigue: nos es dada la función B' , que es la aproximación actual a B , la

función de radiosidad emitida B_e y el punto objetivo x . Con estos datos, debemos realizar un refinamiento final sobre x para obtener una aproximación a la irradiancia en dicho punto. Nótese que el resultado exacto del refinamiento final es diferente de la irradiancia exacta sobre x . Esto se debe al uso de B' en lugar de B como término fuente. De hecho, definimos el resultado exacto del refinamiento final sobre x como $E_g(x)$. Este valor viene dado por la siguiente expresión:

$$E_g(x) = \int_S G(x, y) B'(y) dA(y) = \int_S T(x, y) dA(y) \quad (4.3)$$

donde $G(x, y)$ se define en la ecuación (1.33), y hemos definido, por simplicidad $T(x, y) = G(x, y)B'(y)$. Aunque deseamos obtener $E_g(x)$, esto no es posible por que la integración anterior no se puede llevar a cabo en poco tiempo para entornos reales. Como usualmente, es necesario recurrir a métodos de muestreo por Monte-Carlo para lograr este objetivo. El resultado que se obtiene es una aproximación (con cierta varianza) a E_g , que llamaremos E'_g .

4.3 Distribuciones de muestreo para refinamiento final.

Con el objetivo de aproximar la integral (4.3) usando técnicas de Monte-Carlo, debemos seleccionar una función de densidad de probabilidad válida ¹ p_x sobre el dominio de integración S , cuya integral sea 1, esto es:

$$\int_S p_x(y) dy = 1 \quad (4.4)$$

y asigna una probabilidad mayor que cero a cualquier punto y cuya contribución a la integral sea mayor que cero, esto es, para todos los puntos x e y de S se cumple que:

$$T(x, y) > 0 \implies p_x(y) > 0 \quad (4.5)$$

Cualquier función que cumpla las dos restricciones anteriores es válida para nuestros propósitos. Una vez que p_x ha sido seleccionada, podemos muestrear puntos de las superficies y obtener una aproximación a $E_g(x)$. Esto se lleva a cabo seleccionando un número de muestras n , y después n puntos de S con distribución p_x (a los que se llama y_i). Entonces (como se establece en las reglas de integración por Monte-Carlo) el valor aproximado de la solución se obtiene como:

$$E'_g(x) = \sum_{i=0}^{n-1} \frac{T(x, y_i)}{p_x(y_i)} \quad (4.6)$$

El conjunto de valores posibles de $E'_g(x)$ sigue una distribución cuya media es exactamente $E_g(x)$, cuando los puntos y_i se encuentran distribuidos según $p_x(y)$.

¹esta función tiene como subíndice el punto objetivo donde queremos realizar el refinamiento final, ya que la definición de p_x usualmente incluye x como parámetro.

El problema establecido arriba esta muy relacionado con el problema del muestreo directo de las fuentes de luz. Durante el refinamiento final, todos los otros objetos deben de ser muestreados, mientras que en otros algoritmos, como el ray-tracing distribuido, solamente se muestrean las fuentes de luz. Excepto esto, ambos problemas son similares, ya que la irradiancia es recolectada desde los puntos fuente al punto destino. Muchos artículos, en el contexto de la visualización realista están dedicados al problema de encontrar un función eficiente $p_x(y)$, y la importancia de esto esta descrita en [Shirley96]. El principal objetivo es que el número de muestras que recibe cada región sea ² tan proporcional como sea posible a la energía que esa región radia sobre el punto destino. Pero de hecho esto requiere algún conocimiento de la irradiancia en el punto objetivo, la cual es desconocida.

4.3.1 La varianza de las distribuciones.

Como se ha dicho, cualquier P_x es válido, pero la eficiencia no es igual para todos ellos, ya que cada distribución p_x induce una varianza diferente en la estimación final. El objetivo es entonces una distribución de muestreo con varianza baja, usando datos conocidos. La varianza se escribe como $V(E_g(x))$, aunque escribiremos simplemente $V(x)$ en este capítulo. Este valor es la media de las distancias al cuadrado entre las muestras y el valor medio exacto. Se puede expresar como:

$$V(x) = \frac{1}{n} \left(\int_S \frac{T^2(x, y)}{p_x(y)} dy - E_g^2(x) \right) \quad (4.7)$$

Si usamos $p_x(y) = T(x, y)/E_g(x)$, entonces la función de arriba es cero. Esta sería la estrategia ideal de muestreo, pero es imposible de implementar puesto que la definición de p_x incluye el valor desconocido $E_g(x)$. Nuestro objetivo será usar una función p_x cuya forma sea tan similar como sea posible al valor ideal, con el objetivo de reducir la varianza.

4.3.2 Muestreo del ángulo sólido.

Una opción para seleccionar p_x es asignar una probabilidad a cada y que sea proporcional al ángulo sólido que cubre el punto y cuando se proyecta sobre x . Esto se llama usualmente *hemisphere sampling*, o muestreo de la semiesfera. En este caso p_x se puede expresar como sigue:

$$p_x(y) = \frac{1}{2\pi} \frac{\cos(n_y, w_{yx})}{|x - y|^2} V_{xy} \quad (4.8)$$

nótese que el factor $1/2\pi$ se introduce con el objetivo de cumplir la restricción de normalización (4.4), ya 2π es la superficie de una semiesfera de radio unidad. En la figura 4.3 se puede observar la forma de esta PDF.

²Aquí usamos el término *fente* en el sentido de objetos que radian hacia el punto objetivo x , no simplemente para referirnos a objetos con energía radiante emitida.

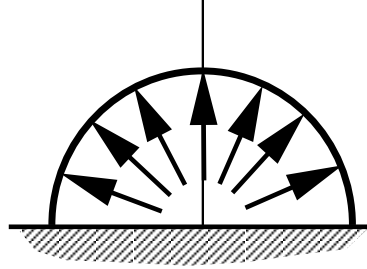


Figura 4.3: Forma de la PDF muestreo del ángulo sólido.

La varianza inducida por la distribución anterior se obtiene sustituyendo (4.8) en (4.7)

$$V_1(x) = \frac{1}{n} \left(2 \int_S \cos(n_x, w_{xy}) G(x, y) B'^2(y) dy - E_g^2(x) \right) \quad (4.9)$$

Esta expresión de la varianza no se puede simplificar más, ya que no podemos realizar ningún supuesto acerca de los valores de B'^2 . Para analizar la varianza, sin embargo, debemos hacer alguna simplificación. En este caso asumiremos que B' es constante, es decir, que todos los puntos y emiten energía con igual intensidad. Llamaremos b al valor constante de la radiosidad en el entorno. Para analizar la varianza podemos convertir la integración en la semiesfera en una integración en el círculo unidad. Una diferencial de ángulo sólido dw alrededor de w en la semiesfera se corresponde con un diferencial de área dp alrededor de un punto p en la base de dicha semiesfera. El punto p se obtiene proyectando verticalmente w en el círculo. Relacionando dy con dw y a su vez dw con dp podemos realizar un cambio de la variable de integración. En lugar de integrar en S , integramos en el círculo unidad C . La relación entre los diferenciales es la siguiente

$$\frac{dp}{dy} = \pi G(x, y) \quad (4.10)$$

y por lo tanto obtenemos

$$2 \int_S \cos(x, y) G(x, y) B_p'^2(y) dy = \frac{2b^2}{\pi} \int_C \cos(p) dp = \frac{2b^2}{\pi} \left(\frac{2}{3}\pi \right) = \frac{4}{3}b^2 \quad (4.11)$$

donde $\cos(p) = (1 - p_x^2 - p_y^2)^{1/2}$. La irradiancia en x se obtiene también mediante un cambio en la variable de integración:

$$E_g(x) = \int_S G(x, y) B_p'(y) dy = b \int_S G(x, y) dy = \frac{b}{\pi} \int_C dp = \frac{b}{\pi} \pi = b \quad (4.12)$$

Con todo esto se simplifica la expresión (4.7), y obtenemos

$$V_1(x) = \frac{b^2}{3n} \quad (4.13)$$

Para obtener un punto y con esta distribución, seleccionamos un vector de longitud unidad, aleatoriamente distribuido en la esfera unidad, y después obtenemos y como el primer punto visible en la escena desde x en la dirección w , esto es $y = p(x, w)$.

4.3.3 Muestreo del ángulo sólido proyectado.

Sabemos que la contribución de la irradiancia entrante desde una dirección w_i a un punto esta ponderada por el coseno del ángulo formado n_x y w . Por lo tanto, una selección mejor para muestrear consiste en tener en cuenta este ángulo y muestrear con mas probabilidad direcciones cercanas a la normal y con menos probabilidad direcciones más horizontales. La expresión de p_x es ahora:

$$p_x(y) = G(x, y) \quad (4.14)$$

en la figura 4.4 se observa la forma de esta PDF. Nótese como en la figura se que las direcciones alejadas de la normal tienen una menor probabilidad de ser seleccionadas.

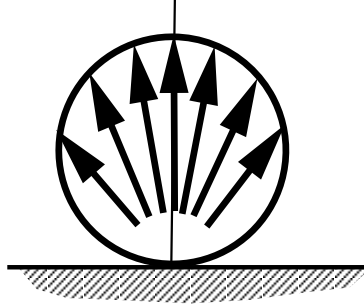


Figura 4.4: Forma de la PDF para muestreo del ángulo sólido proyectado.

Esta fácil demostrar que esta distribución esta normalizada. De nuevo la varianza se obtiene sustituyendo (4.14) into (4.7)

$$V_2(x) = \frac{1}{n} \left(\int_S G(x, y) B_p'^2(y) dy - E_g^2(x) \right) \quad (4.15)$$

Podemos realizar la misma suposición que hicimos para el muestreo de la semiesfera. Si suponemos que B' es constante e igual a B en todos los puntos, entonces este método de muestreo presenta varianza cero. De hecho en este caso particular, la distribución $p_x(y)$ es proporcional a $T(x, y)$, así que se convierte en el método de muestreo ideal, en el sentido descrito en 4.3, y bajo el supuesto de radiosidad constante.

Para seleccionar puntos con esta distribución, primero seleccionamos un punto aleatorio uniformemente distribuido en el círculo de radio unidad. Después se calcula el punto en la semiesfera situado sobre el primero en la dirección del eje Y. Este punto se interpreta como la dirección w de un rayo que parte de x . Usando ray-casting, se obtiene y_i como el primer punto visible en la dirección del rayo, esto es $y = p(x, w)$.

4.3.4 Muestreo del área.

Podemos asignar una densidad de probabilidad constante a todos los puntos de las superficies. Esta distribución es independiente de x , y resulta muy fácil de implementar. En este caso tenemos que:

$$P_x(y) = \frac{1}{|S|} \quad (4.16)$$

la varianza se obtiene sustituyendo p_x en (4.7), lo cual proporciona:

$$V(x) = \frac{|S|}{n} \left[\int_S G^2(x, y) B'^2(y) dy - E_g^2(x) \right] \quad (4.17)$$

El algoritmo para seleccionar puntos con esta distribución consta de dos pasos: primero se selecciona un objeto de la escena, de forma que cada uno de ellos tiene una probabilidad de ser seleccionado proporcional a su área. En el segundo paso se selecciona un punto en el interior del objeto. El algoritmo concreto que se debe usar para esto depende de la geometría del objeto. Se debe asegurar que regiones con igual área tienen siempre la misma oportunidad de ser elegidas, y dicha probabilidad es proporcional al área. Mientras que en el caso de objetos simple como esferas, planos o cilindro es esto es fácil, para objetos más complicados como una superficie B-spline esto es más difícil.

4.3.5 Muestreo de áreas parametrizadas.

Para algunas formas de objetos, el muestreo del área se vuelve difícil. Cuando los objetos están compuestos de superficies parametrizadas, se puede seleccionar el punto de muestreo en el espacio paramétrico, y después encontrar el punto de las superficies y como la imagen por la función de parametrización del punto original en el rectángulo unidad.

Supongamos que la escena consiste en m superficies paramétricas $\{S_1 \dots S_m\}$, cada una de ellas tiene area $|S_i|$ y se encuentra definida por una función de parametrización f_i . Sea (u, v) un punto en $[0, 1]^2$ (el plano unidad en 2D). Entonces $y = f_i(u, v)$ es un punto en la superficie numero i . Consideremos la medida de area A_i en S_i y la medida de area A en $[0, 1]^2$. La función J_i , que depende de f_i se define como

$$\frac{dA_i(y)}{dA(u, v)} = J_i(y) = \left| \frac{df_i(u, v)}{du} \times \frac{df_i(u, v)}{dv} \right| \quad (4.18)$$

es decir, $J_i(y)$ es la relación entre las áreas diferenciales que rodean un punto (u, v) y su imagen y . y se obtiene como el módulo del producto vectorial de los vectores tangentes a la superficie en y . Una vez definida J_i , podemos establecer la probabilidad de seleccionar un punto cualquiera como sigue:

$$p_x(y) = \frac{|S_i|}{|S|} \frac{1}{J_i(y)} \quad (4.19)$$

donde i es el índice de la superficie donde y reside. Hemos supuesto que cada superficie se selecciona con una probabilidad proporcional a su área, como en el muestreo de áreas. El punto u se selecciona uniformemente en $[0, 1]^2$.

4.4 Funciones de probabilidad con reutilización de información.

Es posible realizar simultáneamente el cálculo de E'_g con la recolección de información acerca de la distribución de la irradiancia. Esta información se puede usar durante la fase de refinamiento final para obtener distribuciones de probabilidad con menor varianza. En esta sección vemos como podemos extender fácilmente un algoritmo de fotosimulación con el objetivo de obtener aproximaciones a la energía radiante total que abandona un trozo j y alcanza otro trozo i . Después, definiremos distribuciones que están parametrizadas con respecto a estos valores.

4.4.1 Aproximaciones a la distribución de las irradiancias.

La distribución de la irradiancia sobre cualquier punto se describe con exactitud mediante la función $T(x, y)$, que da la parte de irradiancia en x debido a la radiosidad en y . Obviamente esta función no puede conocerse con exactitud de forma analítica. Sin embargo, el proceso de cálculo de B' puede extenderse para calcular además una aproximación discreta a T .

Con este objetivo, recurrimos a alguna clase de proyección en conjuntos finitos de funciones base. Por simplicidad, usaremos una partición de S en un conjunto de m áreas disjuntas o trozos, llamados A_i . Asimismo cada trozo A_i es plano, con un área igual a $|A_i|$. Los algoritmo que calculan la radiosidad en los trozos se pueden extender fácilmente para calcular las transferencias de energía entre dichos trozos. En todos los casos comenzamos el cálculo con un matriz de m^2 entradas r_{ij} inicializadas a cero.

- En el caso de la radiosidad progresiva, en cada disparo desde el trozo A_j hacia el trozo A_i , debemos añadir el incremento de radiosidad en el valor actual de b_i . Además de esto, podemos añadir la cantidad de energía transmitida al valor r_{ij} .
- En el caso de fotosimulación, cada vez que una partícula abandona el trozo A_j y alcanza el trozo A_i añadiremos su peso actual (el quantum de energía que transporta) a r_{ij} (incluso si la partícula se absorbe en A_i).

Como puede observarse, en ambos casos obtenemos una matriz de valores. Cada valor r_{ij} es un valor aproximado de la energía que abandona el trozo j y alcanza el trozo i . El cálculo de estos valores no incrementa demasiado el tiempo de cálculo. Desgraciadamente, la cantidad de memoria necesaria para guardarlos es proporcional al cuadrado del número de trozos. Sin embargo, dicho número de trozos no tiene por que ser muy alto ya que no se usan directamente en la producción final de la imagen.

La irradiancia sobre x debido a energía proveniente del trozo j es igual a $E_j(x)$, donde E_j es una función que se define como sigue:

$$E_j(x) = \int_{A_j} G(x, y) B'(y) dy \quad (4.20)$$

obviamente se cumple que $E_g(x) = \sum_j E_j(x)$.

También definimos la irradiancia media aproximada en el trozo i como $r_i = \sum_{j=1}^m r_{ij}$. La diferencia entre la irradiancia media aproximada y la exacta en un punto x debido al trozo j es una función Δ_j que se define como

$$\Delta_j(x) = r_{ij} - E_j(x) \quad (4.21)$$

donde i es el índice del trozo que contiene a x . El error total entre la irradiancia aproximada y la exacta es Δ , que se puede escribir como:

$$\Delta(x) = r_i - E_g(x) = \sum_{j=1}^m \Delta_j(x) \quad (4.22)$$

4.5 Distribuciones de probabilidad ponderadas.

Una vez que los valores r_{ij} se han obtenido en el primer paso del algoritmo, pueden ser usados durante la fase de refinamiento final. El objetivo es obtener una distribución de probabilidad cuya forma sea lo más parecida posible a la función objetivo actual $T(x, y)$, como se explica en la sección 4.3.

Supongamos que el punto objetivo x está situado en el trozo A_i . Podemos diseñar una distribución tal que la probabilidad de seleccionar y en el trozo A_j es proporcional a r_{ij} . Esto implica que p_x cumple la siguiente restricción:

$$\int_{A_j} p_x(y) dy = \frac{r_{ij}}{r_i} = \frac{E_j(x) + \Delta_j(x)}{I_g(x) + \Delta(x)} \quad (4.23)$$

Esta ecuación no define totalmente p_x ya que únicamente fija el valor de su integral en cada trozo, no su valor exacto en cada punto. De hecho se necesita un conjunto de distribuciones q_{xj} que de la distribución de las muestras en el interior del trozo j cuando el punto objetivo es x . Esta función se define en el trozo j y esta normalizada, esto es:

$$\int_{A_j} q_{xj}(y) dy = 1 \quad (4.24)$$

Una vez definida q_{xj} , podemos escribir una expresión que caracteriza totalmente p_x , como sigue:

$$p_x(y) = \frac{r_{ij}}{r_i} q_{xj}(y) \quad (4.25)$$

donde i es el trozo donde x se encuentra y j es el de y . Es fácil demostrar que p cumple los requisitos requeridos. Con el objetivo de seleccionar un punto y con esta distribución, es necesario primero seleccionar un trozo con probabilidad r_{ij}/r_i . Esto se lleva a cabo mediante una PDF discreta definida sobre el segmento de enteros de 1 hasta m . Después de esto, se selecciona un punto y en el interior del trozo j seleccionado, con distribución dada por q_{xj} .

Podemos usar varias distribuciones de probabilidad en el interior de los trozos. En las siguientes subsecciones se detallan varias de ellas.

4.5.1 Muestreo ponderado de área y área parametrizada.

La elección más sencilla es elegir un valor constante de q_{xj} en el interior de cada trozo j . Todos los puntos del trozo tienen la misma probabilidad de ser elegidos. Así q_{xj} se encuentra totalmente caracterizado por la ecuación (4.24), y se obtiene:

$$q_{xj}(y) = \frac{1}{|A_j|} \quad (4.26)$$

Esto implica que las regiones con igual área son seleccionadas con igual probabilidad. Como se ha establecido antes, en el caso de superficies arbitrarias esto puede ser difícil. Pero si existe una parametrización, se puede usar dicha propiedad para seleccionar puntos del entorno. Supongamos que el trozo número j es la imagen bajo una función f_j de una región H_j incluida en $[0, 1]^2$, con área $A(H_j)$. En estas condiciones q_{xj} se define como sigue:

$$q_{xj}(y) = \frac{1}{A(H_j)} \frac{dA(u, v)}{dA_j(y)} \quad (4.27)$$

donde $y = f_j(u, v)$ y A_j es la medida de área estándar en el trozo j . Para seleccionar un punto y con esta distribución, primero seleccionamos un punto aleatorio en el trozo H y después obtenemos y como $f(u)$.

4.5.2 Muestreo ponderado del ángulo sólido.

En este caso, la probabilidad de seleccionar un punto y es igual al diferencial de ángulo sólido cubierto por un diferencial de área alrededor de y cuando se proyecta sobre x . Esto implica que:

$$q_{xj} = \frac{1}{G_j(x)} \frac{\cos(y, w_{xy})}{|x - y|^2} \quad (4.28)$$

donde

$$G_j(x) = \int_{S_j} \frac{\cos(y, w_{xy})}{|x - y|^2} dy$$

nótese que $G_j(x)$ es el ángulo sólido que cubre el trozo j cuando se proyecta sobre x (sin tener en cuenta posibles oclusiones). La figura 4.5 muestra un ejemplo de la forma de esta PDF en un entorno cerrado sin oclusiones.

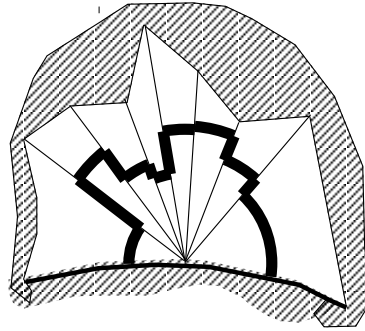


Figura 4.5: Forma de la PDF muestreo ponderado del ángulo sólido.

Es difícil seleccionar el punto y para cualquier tipo de geometría del trozo j . Pero en el caso de trozos triangulares planos podemos recurrir a las técnicas descritas en [Arvo95] para obtener puntos y con exactamente esta distribución. Hacemos esto seleccionando un punto A con distribución uniforme en $[0, 1]^2$, y después obtenemos B como la imagen de A bajo la función de parametrización para triángulos. Finalmente, el punto y se obtiene como el primer punto en el triángulo original visible desde x en la dirección dada por B .

Lo anterior también puede llevarse a cabo en el caso de trozos poligonales que puedan descomponerse en triángulos. La figura 4.6 ayuda a visualizar los elementos involucrados en este proceso de muestreo.

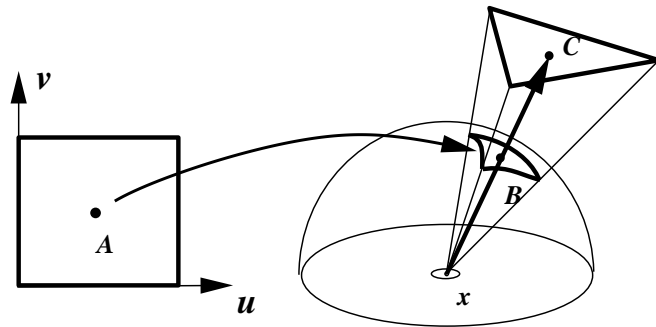


Figura 4.6: Método de muestreo para el ángulo sólido ponderado.

4.5.3 Muestreo ponderado del ángulo sólido proyectado.

En este caso, el valor $q_{xj}(y)$ es proporcional al valor $G(x, y)$. Si incluimos las constantes de normalización necesarias obtenemos:

$$q_{xj}(y) = \frac{1}{F_j(x)} \frac{\cos(x, w_{xy}) \cos(y, w_{yx})}{|x - y|^2} \quad (4.29)$$

donde

$$F_j(x) = \int_{S_j} \frac{\cos(x, w_{xy}) \cos(y, w_{yx})}{|x - y|^2} dy$$

nótese que $F_j(x)$ es igual al factor de forma del punto y al trozo j , multiplicado por π . En la figura 4.7 vemos la forma aproximada que toma esta PDF.

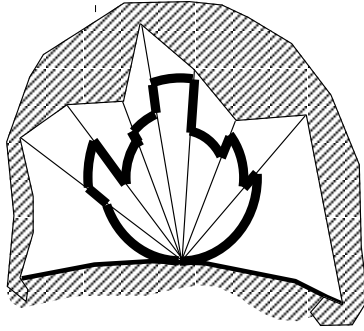


Figura 4.7: Forma de la PDF para muestreo ponderado del ángulo sólido proyectado.

Esta distribución es la más cercana a $G(x, y)B'_a(y)$. De hecho, cuando los términos de error Δ_i son todos cero, la varianza que obtenemos es también cero. Esto se puede demostrar obteniendo la expresión de la varianza. Combinando (4.20) y (4.21) con la definición anterior obtenemos:

$$V(x) = \frac{1}{n} \left([E_g(x) + \Delta(x)] \sum_{i=1}^m \frac{E_i^2(x)}{E_i(x) + \Delta_i(x)} - E_g^2(x) \right) \quad (4.30)$$

cuando $\Delta_i(x) = 0$, para todos los i , entonces $\Delta(x) = 0$ y $V(x) = 0$. Esto implica que cuanto más cercano este r_{ij} a $I_j(x)$, mas bajo es el valor que obtenemos para $V(x)$.

Desgraciadamente no existe aún un algoritmo para seleccionar puntos con esta distribución. Un algoritmo de esta clase debería estar basado en la expresión analítica del factor de forma de un punto a un polígono (sin occlusiones), y usando los mismos principios a los expuestos en [Arvo95] para muestreo del ángulo sólido.

4.6 Implementación y Resultados.

Para probar las distribuciones de muestreo descritas, hemos implementado un método de dos pasadas en nuestro sistema de visualización *girt*. En la primera pasada, calculamos la función E' como una aproximación discreta a la función de irradiancias real. Esto se lleva a cabo usando fotosimulación, como se describe en [Pattanaik92]. Hemos extendido el método detallado en dicho artículo con el objetivo de calcular la cantidad de irradiancia en cada trozo proveniente de otros trozos.

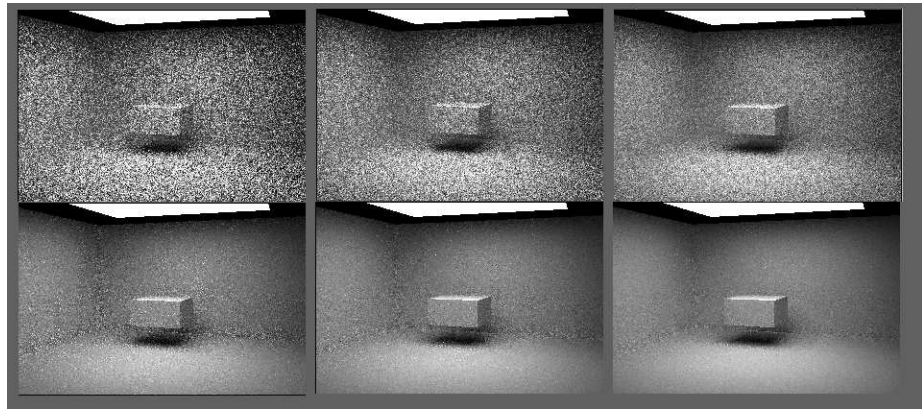


Figura 4.8: Escena simple: 8, 16 y 32 muestras por pixel.

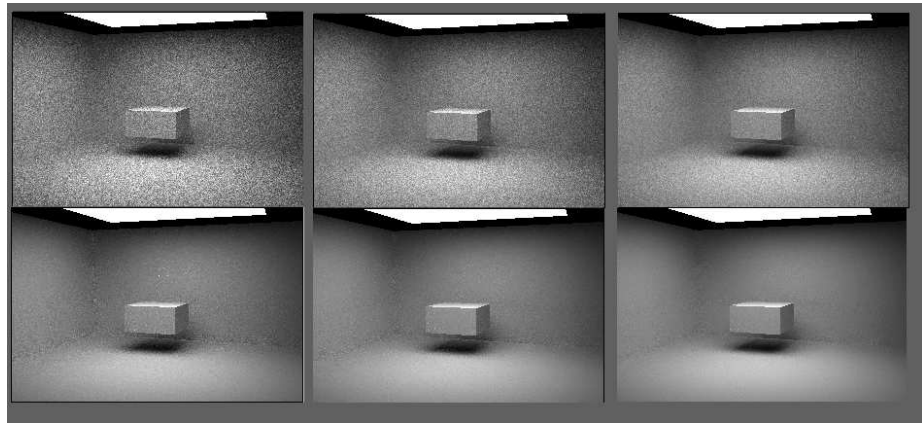


Figura 4.9: Escena simple: 64, 128 y 256 muestras por pixel.

Después de este primer paso se lleva a cabo la segunda pasada. Aquí se selecciona un punto de vista, y entonces, para cada pixel se muestrea el área del pixel mediante trazado de rayos. En cada posición de muestra calculamos el primer punto

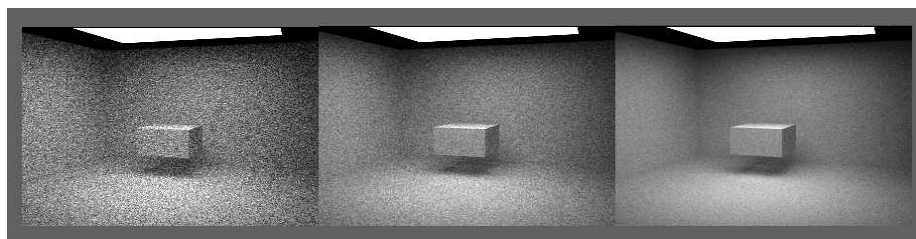


Figura 4.10: Escena simple por path-tracing: 64, 256 y 1024 muestras por pixel.

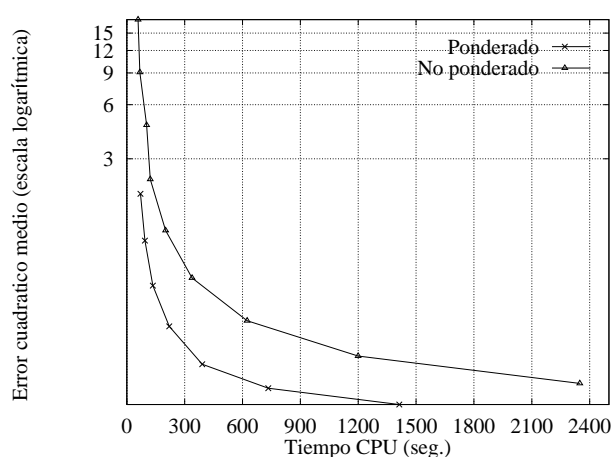


Figura 4.11: Gráfico de σ^2 con respecto al tiempo de CPU

visible x en la escena. Después de esto se realiza un refinamiento final sobre x seleccionando un número de puntos y_i en la escena, con distribución $p_x(y)$. Entonces, usando (4.6) obtenemos $E'_g(x)$ como un valor aproximado de $E_g(x)$. Finalmente se obtiene una aproximación a la radiosidad a partir de la irradiancia.

Hemos implementado tres métodos distintos para refinamiento final:

- **Muestreo del ángulo sólido proyectado.**, como se describe en la subsección 4.3.3. Este método tiene la desventaja de que las fuentes pequeñas que iluminan las escenas se muestrean muy pobremente. La imagen aparece muy ruidosa, ya que dichas fuentes tienen muy pocas posibilidades de ser seleccionadas, aunque pueden contribuir de forma importante a la irradiancia en el punto objetivo.
- **Muestreo ponderado de áreas paramétricas.**, como se describe en la subsección 4.5.1. En este caso cada fuente recibe un número de muestras que es aproximadamente proporcional a su contribución a la irradiancia en x . Desgraciadamente este método muestra una alta varianza cuando el punto

<i>ns</i>	<i>No ponderado</i>				<i>Ponderado</i>			
	<i>rays</i>	<i>CPU</i>	σ^2	σ	<i>rays</i>	<i>CPU</i>	σ^2	σ
1	384	58.2	17.818	4.221	384	70.6	1.915	1.384
2	441	67.3	9.078	3.013	441	93.6	1.052	1.026
4	556	102.7	4.617	2.148	555	134.8	0.591	0.769
8	784	121.4	2.307	1.518	783	219.9	0.351	0.593
16	1241	201.0	1.198	1.094	1239	390.8	0.216	0.465
32	2155	338.3	0.652	0.807	2152	733.6	0.159	0.399
64	3983	624.0	0.377	0.614	3976	1413.0	0.129	0.360
128	7639	1199.2	0.240	0.490	7625	2782.0	0.114	0.338
256	14952	2348.9	0.169	0.411	14924	5509.4	0.107	0.328

Figura 4.12: Resultado de las comparaciones numéricas.

x esta muy cerca de una fuente importante (una adyacente y perpendicular al trozo donde x está, por ejemplo). Esto se debe a las altas variaciones del factor de forma entre el punto receptor y el punto de muestra, que produce altas variaciones entre las contribuciones de cada muestra.

- **Muestreo ponderado del ángulo sólido proyectado.**, como se describe en la subsección 4.5.2. En este caso, el muestreo ponderado asegura que las fuentes importantes son correctamente muestreadas. Además, la función p_x incluye una parte del factor de forma entre puntos, con lo cual las variaciones de la contribución no son tan importantes.

El segundo método no es comparable al tercero, ya que en el segundo el ruido es muy grande cuando el punto x esta muy cerca del punto de muestreo seleccionado. Este ruido es menor cuando se usa el tercer método.

Hemos realizado simulaciones en una escena muy simple con dos muros, el suelo, una fuente que ocupa todo el techo y un cubo que bloquea la luz. Para cada punto objetivo, la fuente del techo cubre un ángulo sólido grande. De esta forma el muestreo del ángulo sólido muestra mucha menos varianza que en escenas con fuentes de luz pequeña que recibirían muy pocas muestras. Se lanzaron 100.000 partículas desde las fuentes de luz. La escena esta mallada en 100 trozos.

La figura 4.12 muestra los resultados que obtuvimos con la escena descrita. La primera columna indica el número de muestras, que coincide con las potencias de 2 entre 1 y 256. Después de esto se da el tiempo de CPU en segundos, y el número de rayos en miles. Los dos valores anteriores tienen en cuenta toda la simulación, incluyendo el trazado de partículas. Los errores se miden con respecto a una solución sin sesgo obtenida con path-tracing simple y número muy alto de muestras por pixel (1024). Mostramos el error cuadrático medio de todos los pixeles, y la raíz de dicho valor (desviación típica media). Nótese que los tiempos de CPU son mayores en el caso de muestreo ponderado, para un número igual de muestras. Esto se debe al coste involucrado en el muestreo de los triángulos esféricos ³.

³La implementación actual no está optimizada en el sentido de que un nuevo triángulo

Una gráfica de σ^2 (con el tiempo de CPU en eje X) para ambos métodos se muestra en 4.11. Como puede observarse, el muestreo ponderado del ángulo sólido es más eficiente que el muestreo del ángulo sólido.

Mas allá de los errores numéricos, las imágenes obtenidas mediante el muestreo del ángulo sólido muestran mucho más ruido que las obtenidas con el muestreo ponderado, incluso para tiempos de CPU más bajos. Esto puede observarse en las figuras 4.8 y 4.9. Cada una de estas figuras muestra dos filas de imágenes. La primera fila se obtuvo mediante muestreo no ponderado, y la segunda mediante muestreo ponderado. En cada columna, el número de muestras tomadas es el mismo en ambas imágenes. Estas son las imágenes tomadas para comparaciones numéricas.

La figura 4.10 muestra tres imágenes obtenidas con path-tracing simple, con 64, 256 y 1024 muestras por pixel. La última de ellas se usó como imagen de referencia para el cálculo de los errores.

Finalmente, la figura (4.13) muestra una habitación con 540 objetos, mallados en 800 trozos, y sintetizado usando muestreo ponderado. Para esta imagen se tomaron cuatro muestras por pixel, y en cada una de ellas se realizaron 50 muestras para refinamiento final.

4.7 Conclusiones.

El uso de información *a priori* sobre una función puede ayudar mucho en el diseño de estrategias de muestreo para la integración por Monte-Carlo de la misma. Esta es la idea básica presentada en este capítulo. En el contexto de los entornos difusos, hemos mostrado como la primera pasada (de un algoritmo de dos pasadas) puede usarse para obtener esa información *a priori*, que es tomada en la segunda pasada para dirigir el proceso de muestreo hacia las partes más interesantes de la escena, en el sentido que contribuyen más al punto objetivo.

Usando la idea anterior se ha diseñado un algoritmo de *final-gather* con varianza muy reducida comparado con algoritmos que no tienen información *a priori*, como el path-tracing. Dicho algoritmo se ha implementado en nuestro sistema de síntesis de imágenes, junto con el de Path-Tracing, obteniéndose una mejora significativa en los tiempos de cálculo frente a este último método.

esférico se crea para cada muestra. Hubiésemos obtenido menores tiempos de CPU mediante el almacenamiento temporal y la reutilización de los triángulos esféricos cuando fuese posible, como se describe en [Arvo95]



Figura 4.13: Habitación con 800 trozos, y 4×50 muestras de refinamiento final.

Capítulo 5

Refinamiento final para Iluminación Global.

5.1 Introducción

En este capítulo, proponemos un algoritmo de Monte-Carlo de dos pasadas para entornos difusos. En la primera pasada, una aproximación a baja resolución a la función de radiancias se obtiene usando un algoritmo de trazado de partículas. En la segunda pasada, se produce una imagen después de un paso de refinamiento final a nivel de pixels, al igual que se ha llevado a cabo previamente en algoritmos de elementos finitos para Iluminación Global. Al igual que el algoritmo descrito en el capítulo anterior, este paso de refinamiento final se ve mejorado al usar muestreo ponderado o por importancias, lo cual reduce la varianza. El muestreo de importancias está basado en la información sobre el intercambio de energía entre elementos de superficie, que se encuentra almacenada en una estructura jerárquica de enlaces.

Con respecto a algoritmos anteriores de elementos finitos, el uso de técnicas estocásticas evita el muestreo explícito de cada enlace durante el refinamiento final. Al igual que el algoritmo descrito en el capítulo anterior, el número de muestras que se deben de tomar no está relacionado con el número de enlaces en un elemento receptor, sino que está relacionado con la varianza. Dicha varianza se reduce bastante cuando se tiene en cuenta la importancia visual en la estructura de enlaces, y esto es especialmente cierto para entornos no puramente difusos.

En este capítulo proponemos un algoritmo que extiende el que se presentó en el capítulo previo, al tiempo que reduce la complejidad de tiempo y almacenamiento que exhibe aquel método. En dicho método, la mejora del paso de refinamiento final se obtenía mediante la representación explícita de las transferencias de energía entre todos los pares de elementos. Esto es muy ineficiente para escenas de complejidad media o alta. Sin embargo, la complejidad se puede reducir si se realiza una primera pasada dependiente del punto de vista (que proporciona la energía transferida en cada enlace, ponderada por su importancia para el observador), y también mediante el uso de agrupamiento jerárquico para la representación de dichas transferencias.

Al tener en cuenta radiancias en lugar de radiosidades, la dimensión del problema se incrementa, ya que la radiancia está definida en el espacio de los rayos mientras que la radiosidad está definida en el espacio bidimensional de los puntos de las superficies.

La ecuación de radiosidad clásica y la ecuación de radiancias pueden escribirse ambas como ecuaciones integrales de segundo orden, con un núcleo de dos variables, y una función desconocida de una variable. Esta función desconocida puede ser representada, en los dos casos, como una función a baja resolución almacenada mediante un conjunto de coeficientes con respecto a una base. Estos coeficientes se pueden calcular aproximadamente mediante trazado de partículas.

En el siguiente apartado se revisan los trabajos anteriores relacionados con el nuestro, y en el posterior se detallan las mejoras que ofrece el que aquí presentamos.

5.1.1 Trabajos anteriores.

Como se detalló en el capítulo anterior, el refinamiento de una aproximación a baja resolución ha sido tratado en varios artículos en la literatura sobre visualización realista. A los artículos allí citados, todos ellos relativos a entornos difusos, cabe añadir el trabajo de presentado en [Jensen95] para entornos arbitrarios. En este artículo se usa una primera pasada de fotosimulación. Los puntos y direcciones de impacto de todos los fotones son almacenados, de forma que en la segunda pasada dicha información se usa para realizar la generación de la imagen por Path-Tracing.

En concreto, en cada punto x alcanzado por una cadena durante la segunda pasada se construye una *pdf* con el objetivo de seleccionar la siguiente dirección de muestreo. La construcción de la *pdf* se realiza usando los fotones que impactaron en la primera pasada cerca de x . Dichos fotones codifican las direcciones mas importantes de donde proviene la energía del punto de impacto. Este esquema tiene la desventaja de que es necesario construir la *pdf* en cada punto de impacto. El almacenamiento temporal de los impactos también puede ocupar mucha memoria. Además, es posible que alguna fuente importante que ocupe un ángulo sólido pequeño no sea adecuadamente muestreada, debido a que la *pdf* es discreta y puede no tener la resolución adecuada para representar adecuadamente dicha fuente.

En la siguiente subsección se resume los fundamentos del algoritmo propuesto, y asimismo se detallan las ventajas que presenta.

5.1.2 El algoritmo propuesto.

Si consideramos una estructura de enlaces entre todos los posible elementos en los que el espacio de los rayos puede ser discretizado, entonces es necesario almacenar un valor escalar para cada par de dichos elementos. Esto implicaría un coste excesivo en términos de almacenamiento y tiempo, ya que el número de elementos en el espacio de los rayos es un orden de magnitud mayor que el necesario para aproximar la radiosidad.

Con el objetivo de solventar esta limitación, podemos considerar la forma del núcleo K involucrado en la definición de la radiancia. La presencia de una función delta en dicha definición implica que para la mayoría de los pares de elementos no hay

transferencia directa de energía entre ellos. De hecho solamente es necesario tener en cuenta transferencias de energía entre ternas de puntos de la superficie, mediante el uso de una formulación de transporte de 3 puntos, como puede observarse en [Aupperle93]. La complejidad resultante está entonces en el orden de $O(n^3)$, donde n es el número de elementos de superficie o trozos. Esta complejidad es aún muy alta para entornos medianos o complejos.

La complejidad puede reducirse aún más. De hecho en nuestro método no es necesario almacenar las transferencias entre todas las posibles ternas de elementos de superficie. Esto se debe a que la longitud de los caminos durante el refinamiento final es tres: desde el observador se va al punto receptor, y de aquí al punto donde se muestrea la función objetivo. Por tanto el observador es siempre el mismo se encuentra fijo durante toda la simulación. Aprovechando esto, podemos ver las transferencias como una función de dos variables, ya que el tercer punto es constante. Para cada par de puntos x e y , esta función representa la cantidad de energía que viaja de x hacia y y después es reflejada hacia el observador. Esto se hace a coste de obtener una solución dependiente del punto de vista, pero con la ventaja de reducir la complejidad a un orden de $O(n^2)$.

Esto es aún demasiado complejo para entornos con un número grande elementos de superficie. El siguiente paso consiste en considerar técnicas de agrupamiento jerárquico. La geometría de la escena se puede representar mediante una jerarquía donde cada nodo representa un conjunto de superficies o bien un elemento de una superficie. Para cada elemento terminal en dicha jerarquía, se mantiene una lista de enlaces a otros nodos de la misma, no necesariamente terminales, de forma que cada enlace tiene un peso mínimo. De esta forma se evita almacenar una representación explícita de todas las transferencias entre trozos, salvando así memoria y tiempo de ejecución.

5.2 Agrupamiento jerárquico.

Para nuestros sistemas hemos usado el mismo método de agrupamiento de objetos que se describe en [Christensen97]. Como resultado de este proceso de agrupamiento, se obtiene un árbol en el cual cada nodo está asociado a una parte de todas las superficies de los objetos. El área cubierta por cada nodo es un subconjunto de la cubierta por su nodo padre en el árbol. El nodo raíz cubre todas las superficies de los objetos.

Clasificamos los nodos del árbol en tres tipos: nodos grupo, nodos objeto y nodos trozo. Un nodo grupo cubre un conjunto de objetos, un nodo objeto cubre un solo objeto de la escena, y un nodo trozo cubre una superficie de un objeto o un trozo de ella. Los descendientes de un nodo grupo pueden ser otros nodos grupo o bien nodos objeto. Los descendientes de los nodos objeto son siempre nodos trozo. Finalmente, los nodos trozo pueden tener como descendientes solamente nodos trozo, o bien pueden no tener ningún descendiente (a estos últimos los llamamos nodos terminales).

El algoritmo usado para el agrupamiento es como sigue: primero se crea el nodo raíz (un nodo grupo), que se asocia a todos los objetos de la escena. Cada vez

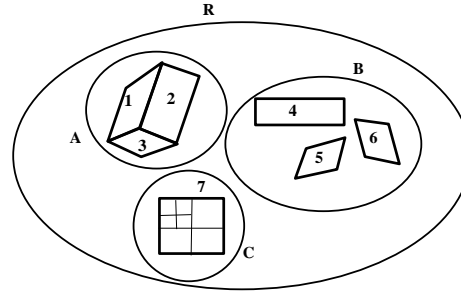


Figura 5.1: Un conjunto de grupos y objetos.

que se crea un nodo grupo se examinan cada uno de los objetos que incluye. Si su tamaño excede la mitad del tamaño de la caja englobante del nodo grupo, se construye un nodo objeto que lo contiene y se pone como descendiente directo del nodo grupo. Si el tamaño no excede el límite anterior, se examina en que suboctante ¹ del nodo padre cae el centro ² de cada objeto y se crean entonces 8 subgrupos hijos del grupo padre, uno para cada suboctante del mismo. Nótese que alguno de ellos puede estar vacío y no ser creado por tanto.

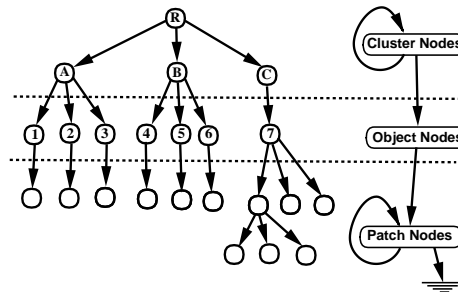


Figura 5.2: Arbol correspondiente a la figura 5.1.

Un nodo objeto está siempre compuesto de un conjunto con un nodo trozo como mínimo (puede tener más si el objeto está compuesto de varias superficies). Estos nodos trozo se dividen siempre en cuatro trozos iguales, cada uno cubriendo un cuadrante del nodo padre, excepto cuando el área del nodo trozo padre es inferior a un valor mínimo, en cuyo caso se considera un nodo terminal y no se subdivide más.

¹Para cada nodo grupo, se calcula la caja englobante de todos sus objetos y se almacenan los ocho octantes que se obtienen al dividir la misma en 8 partes iguales.

²El centro de un objeto es el punto central de su caja englobante.

5.3 Primera pasada.

La primera pasada del método propuesto es simplemente un algoritmo de fotosimulación o trazado de partículas, extendido con el objetivo de procesar los enlaces que se almacenan en la jerarquía de grupos y trozos. En los dos apartados siguientes detallamos varios aspectos del proceso de fotosimulación, y presentamos el método de actualización de los enlaces.

5.3.1 Fotosimulación.

El proceso de trazado de partículas que hemos implementado es similar al que se describe en [Pattanaik92]. Las partículas se crean en las fuentes de luz, y se simula su recorrido por la escena. Todas las fuentes producen energía con una distribución difusa, y con intensidad constante en el interior del área de cada una de ellas. Para la creación de partículas, cada fuente se selecciona con una probabilidad proporcional a la energía total que emite.

Después de seleccionar una fuente, se selecciona un punto en su interior, con una distribución de probabilidad uniforme con respecto al área. A continuación se calcula la dirección de salida inicial, según una distribución difusa. Si llamamos $dP(\mathbf{r})$ a la probabilidad de escoger r como primer rayo de una partícula, entonces se cumple que $dP(\mathbf{r}) = L_e(\mathbf{r})d\mu(\mathbf{r})/\phi_e$, donde ϕ_e es la energía total emitida en todas las fuentes. A cada partícula se le asigna un peso w , que al crear la partícula es igual a la energía total emitida en la escena dividida por el número de partículas.

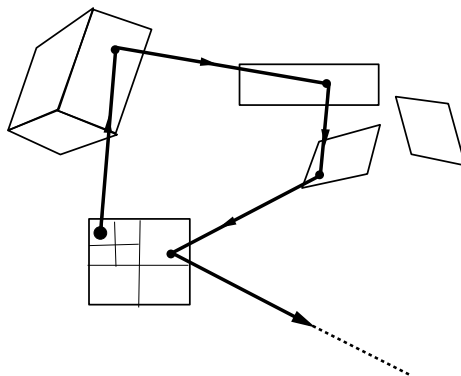


Figura 5.3: El camino seguido por una partícula.

Una vez que la partícula ha sido creada, se lleva a cabo una simulación de la trayectoria que dicha partícula sigue por el entorno. Esta simulación está compuesta de una serie de reflexiones en las superficies que acaban bien por que la partícula es absorbida, bien por que abandona la escena (que no es necesariamente cerrada). La selección de la nueva dirección de una partícula después de una reflexión se realiza mediante muestreo de la BRDF. Téngase en cuenta que la partícula alcanza, en cada reflexión, un punto de un nodo trozo terminal en la jerarquía de nodos.

Por lo tanto, podemos ver la trayectoria de la partícula como una serie de nodos terminales.

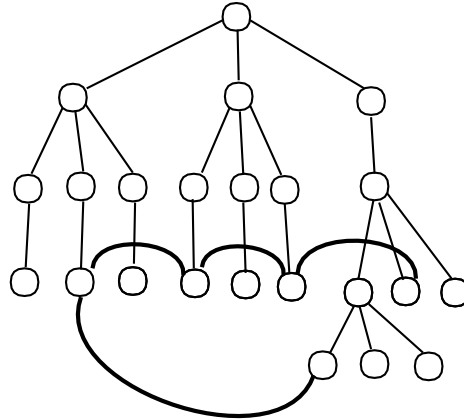


Figura 5.4: Nodos terminales visitados.5.3.

5.3.2 Información de estado de las partículas.

La información de estado de las partículas incluye todos los elementos necesarios para el cálculo de la función de radiancia aproximada y la actualización de los enlaces. En concreto, de cada partícula se conocen los siguientes datos:

- El punto donde ocurrió la última reflexión (o bien el punto donde la partícula ha sido creada si aún no ha pasado por ninguna reflexión)
- El vector de dirección de la partícula después del impacto.
- El peso de la partícula, que puede verse como la cantidad de energía que transporta.

Para entornos monocromos, un valor escalar es suficiente para guardar el peso de la partícula. Sin embargo, los entornos con los que se tratan presentan reflectividades no monocromáticas, esto es, que dependen de la longitud de onda. Esto puede resolverse repitiendo la simulación para un conjunto de longitudes de onda fijas y después combinando adecuadamente los resultados, como se hace habitualmente en los métodos de elementos finitos.

Otra opción es asignar a cada partícula una longitud de onda, usando una distribución aleatoria que sea proporcional a la distribución de emisividades de la fuente donde se crea, como se describe en [Pattanaik92]. Finalmente, un tercer método es asignar un triple RGB a cada partícula, como se describe en [Jensen95].

Esta última opción puede generalizarse para tener en cuenta distribuciones arbitrarias de reflectividades y energías, no solamente RGB. Esto implica que el peso

una partícula es una función w de la longitud de onda, de forma que $w(\lambda)$ es el peso actual de la partícula para la longitud de onda λ . Cuando la partícula alcanza una superficie, la reflectividad del punto esta también representada por una distribución ρ dependiente de la longitud de onda, de forma que $\rho(\lambda)$ es la reflectividad a la longitud de onda λ .

Esta representación se introduce para conseguir que el algoritmo de trazado de partículas sea independiente del modelo de distribución elegido, que puede ser distinto del modelo RGB, tal y como se requiere en nuestro sistema de rendering [Urena97].

5.3.3 Probabilidades de supervivencia y absorción.

En el caso de que el peso de las partículas fuese un simple valor escalar, las reflectividades también serían valores escalares entre 0 y 1. Dichos valores podrán, por tanto, ser usados como probabilidades de supervivencia en cada reflexión de una partícula. En el caso de que la energía sea una función de la longitud de onda, esto ya no puede ser así, aunque sigue siendo necesario establecer cual es la probabilidad de supervivencia en la reflexiones.

Para solucionar esto podemos considerar la curva de eficiencia luminosa ³ c . El producto escalar $\langle c | w \rangle$ (definido por una integral en el espacio de las longitudes de onda) es la luminancia de la distribución w . Cuando una partícula alcanza un punto cuya distribución de reflectividades es ρ , podemos considerar la luminancia de la distribución $w\rho$, que es $\langle w\rho | c \rangle$. Esta luminancia es necesariamente menor que la luminancia original de la partícula $\langle w | c \rangle$, ya que ρ es siempre menor que 1. Una posible probabilidad de supervivencia es entonces $r = \langle w\rho | c \rangle / \langle w | c \rangle$, esto es, la relación entre la luminancia de ambas distribuciones. Este valor r cumple el mismo papel que la reflectividad escalar de otros algoritmos descritos.

Este esquema implica que la luminancia de una partícula se mantiene constante después de cada reflexión. Otros métodos para decidir la supervivencia, como el de la *Ruleta Rusa* también son compatibles con la supervivencia con luminancias.

Para implementar esto es necesario que exista una clase objetos para almacenar distribuciones de energía. Esta clase debe de incorporar métodos para extraer su luminancia y para realizar producto punto a punto. El modelo de color concreto usado debe de estar basado en alguna proyección en bases finitas que siempre conlleven alguna pérdida de información.

5.3.4 Aproximación a la función de radiancias.

Cuando una partícula abandona un punto de las superficies después de una reflexión en él, es necesario actualizar los coeficientes usados en la aproximación de la función de radiancia. En principio, se puede usar cualquier función base definida en el espacio de los rayos. En nuestro caso hemos elegido la opción mas sencilla, que consiste en usar funciones base constantes no solapadas.

³Estos es, $c(\lambda)$ es proporcional a la sensibilidad del ojo humano a una unidad de energía radiante con longitud de onda λ

Como se ha explicado antes, todos los objetos de la escenas se encuentran mallados en un conjunto de trozos o elementos de superficie. Para cada trozo se guarda una matriz bidimensional de distribuciones de energía, que se encuentra inicializado a cero. Cada entrada en la matriz almacena la radiancia media en una región de direcciones sobre la esfera unidad. Asumamos que el elemento de superficie número i cubre un area $S_i \subseteq S$, y que existen n trozos de este tipo.

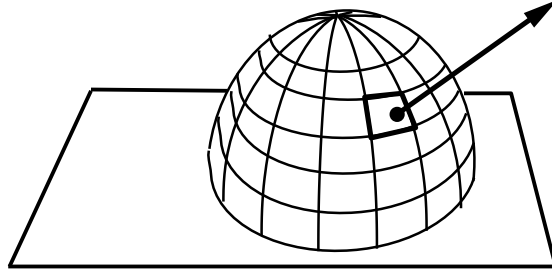


Figura 5.5: El mallado de la semiesfera.

Al igual que las superficies, la semiesfera de direcciones se divide en un conjunto de regiones, cuyos bordes coinciden con los meridianos y paralelos de la esfera, como se puede observar en la figura (5.5). La región entre el meridiano número j y el paralelo k cubre un subconjunto b_{jk} de la semiesfera. Al conjunto de rayos cuyo vector de dirección esta en b_{jk} y cuyo origen esta en el trozo i se les llamará D_{ijk} . Este conjunto puede escribirse como $D_{ijk} = S_i \times b_{jk}$. Supongamos que existen n_p paralelos y n_m meridianos. El conjunto de funciones base está definido como sigue:

$$\mathbf{B} = \{f_{ijk} \mid 1 \leq i \leq n, 1 \leq j \leq n_m, 1 \leq k \leq n_p\}$$

donde

$$f_{ijk}(\mathbf{r}) = \begin{cases} 1/\mu(D_{ijk}) & \text{si } \mathbf{r} \in D_{ijk} \\ 0 & \text{en otro caso} \end{cases}$$

Es fácil probar que el anterior conjunto \mathbf{B} es un conjunto ortonormal de funciones base. La radiancia reflejada media en la región jk del trozo i se llamará c_{ijk} y puede escribirse como sigue:

$$c_{ijk} = \frac{1}{\mu(D_{ijk})} \int_{D_{ijk}} L_r(\mathbf{r}) d\mu(\mathbf{r}) = \langle f_{ijk} \mid L_r \rangle \quad (5.1)$$

donde el producto escalar anterior esta definido con respecto a la medida μ . Nótese que la función L_r puede aproximarse mediante su proyección en B , que es:

$$\mathcal{P}_{\mathbf{B}} L_r = \sum_{ijk} c_{ijk} f_{ijk}$$

Los coeficientes c_{ijk} se pueden calcular mediante el proceso de fotosimulación. Cuando una partícula es reflejada en un punto de un trozo i en una dirección w ,

se busca la región b_{jk} que contiene a w . Entonces se añade el valor wf_{ijk} a c_{ijk} , donde w es el peso actual de la partícula. Hay que tener en cuenta que w no es un valor escalar, sino una función, por tanto también son funciones (o distribuciones de radiancia) los coeficientes c_{ijk} .

5.3.5 Actualización de enlaces.

Al principio de la simulación, la estructura de enlaces comienza con un enlace desde cada nodo trozo terminal hacia el nodo raíz, aunque posteriormente nuevos enlaces se crean, como veremos en el siguiente apartado. Cuando una partícula viaja desde un elemento emisor hasta alcanzar un elemento receptor, la información almacenada en la estructura de enlaces es actualizada. La primera tarea es encontrar el enlace al que la partícula contribuye. Como puede observarse en la figura 5.6, consideramos

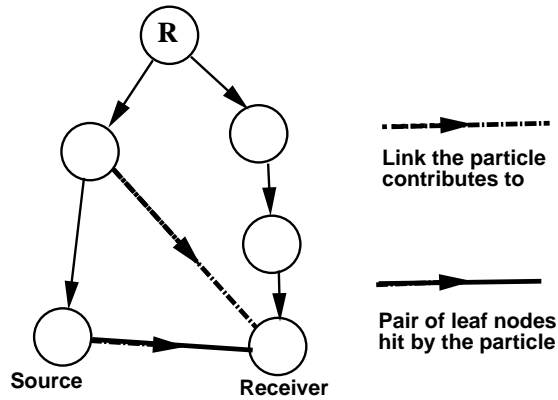


Figura 5.6: Búsqueda del enlace que atraviesa una partícula.

el par de nodos terminales visitados. Se busca entonces si hay algún enlace desde el receptor hacia el emisor. Si dicho enlace no se encuentra el proceso se repite recursivamente para el nodo padre del nodo emisor, hasta que se alcanza el nodo raíz. Necesariamente, el nodo emisor o alguno de sus ancestros debe ser el destino de un enlace desde el nodo receptor.

Cuando el enlace se ha encontrado, se actualiza su peso. El peso de un enlace es la suma de la luminancia total de todas las partículas que lo han atravesado, estando cada una de dichas luminancias ponderada por la fracción de energía de la partícula que se refleja en el receptor hacia el observador. Para actualizar el peso se añade el siguiente valor d al peso anterior del enlace

$$d = \langle w\rho \mid c \rangle \frac{n_t}{n_t - n_0} \quad (5.2)$$

donde w es el peso actual de la partícula, ρ es una distribución de reflectividades, obtenida al evaluar la BRDF en el punto de entrada, c es la curva de eficiencia luminosa, n_t es el número total de partículas lanzadas durante toda la simulación

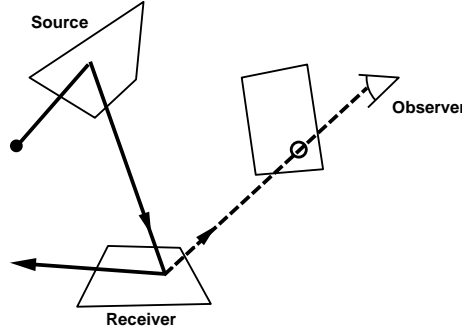


Figura 5.7: Geometría para una transición entre dos superficies.

y n_0 es el número de partículas ya lanzadas cuando el enlace fue creado, siendo $n_0 < n_t$. Esto es, d es la luminancia de $w\rho$, corregida por el factor $n_t/(n_t - n_0)$ que se introduce ya que $n_t - n_0$ es el número total de partículas lanzadas durante el tiempo de existencia del enlace, inferior a n_t , el número total de partículas.

5.3.6 Refinamiento de enlaces.

El refinamiento de enlaces consiste en la subdivisión adaptativa de los mismos, usando como criterio el número de partículas que los atraviesan.

Cada enlace se dice que va *desde* un nodo receptor de las partículas *hacia* otro nodo emisor de las mismas. Esto es así puesto que los enlaces que parten de un nodo están almacenados en una lista enlazada asociada a dicho nodo receptor. Dicha estructura sirve para la segunda pasada, en la cual se conoce el nodo receptor y después se busca el emisor, como veremos en la siguiente sección.

Antes de que comience el trazado de partículas, se crea un enlace desde cada nodo terminal al nodo raíz. Una vez que un número mínimo de partículas n_{min} han atravesado el enlace, este se subdivide. Si el enlace va de un nodo a hacia otro b , entonces, (si b no es terminal) se crean nuevos enlaces desde a hacia cada uno de los nodos hijos de b . El valor n_0 del nuevo enlace es igual al número de partículas que ya se habían lanzado en la simulación al crear el enlace.

Los enlaces que se han subdividido se ignoran desde ese instante, y solo los enlaces terminales, no divididos, se usan para la segunda pasada. Con este método, es posible que exista algún enlace con el valor n_0 muy cercano a n_t , esto es, el enlace se creó cerca del final de la simulación. Los pesos estimados de estos enlaces son una muy mala aproximación a los pesos reales, ya que el número de partículas usadas para realizar la aproximación es muy pequeño, y por tanto la varianza de la estimación es muy alta.

Para evitar esto, podemos fijar un número entero n_{max} (cumpliéndose que $n_{min} < n_{max} < n_t$) tal que ningún enlace es creado con $n_0 > n_{max}$. Esto significa que después de lanzar la partícula número n_{max} , no se crean nuevos enlaces. De esta forma, el número mínimo de partículas lanzadas durante la vida de un enlace

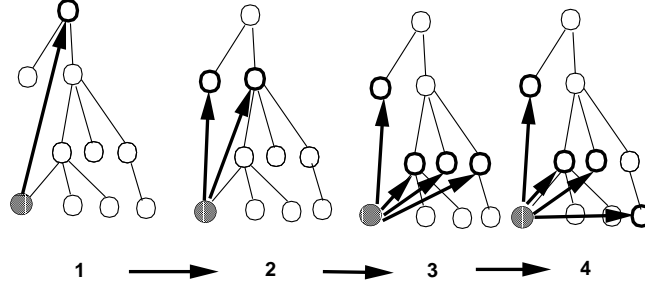


Figura 5.8: Refinamiento de un enlace.

es $n_t - n_{max}$. En nuestras simulaciones hemos seleccionado $n_{max} = n_t/2$

Al final de la simulación existe, para cada nodo terminal, un conjunto de enlaces que apuntan a otros nodos en la jerarquía. En la figura 5.8 puede observarse como un enlace (desde un nodo terminal al nodo raíz) es refinado en 3 subdivisiones durante la simulación, y también vemos los enlaces resultantes.

5.4 Segunda pasada.

La segunda pasada se usa para obtener una imagen del entorno tal y como se ve desde el observador. Esto se lleva a cabo mediante un proceso de refinamiento final. En este contexto, el refinamiento final es el problema de encontrar una aproximación a $L(\mathbf{r})$ para un rayo dado \mathbf{r} , aplicando el operador de transporte \mathcal{T} a una función de radiancias a baja resolución L' . Este refinamiento se realiza usando un método de Monte-Carlo con muestreo ponderado o por importancias, guiado por la información almacenada en los enlaces.

Durante la fase de visualización, se lanza un rayo a través de cada punto de muestreo en el plano de visión. Se obtiene entonces el primer punto visible x de la escena (si existe alguno). Entonces el problema es calcular la radiancia $L(x, w)$, donde w es el vector de dirección unitario que va de x hacia el observador. Esto se lleva a cabo muestreando un conjunto de puntos x_i en las superficies de la escena, y calculando la radiancia reflejada en x hacia el observador desde dichos puntos.

5.4.1 Formalización del problema.

Como en el capítulo anterior, este refinamiento final puede formalizarse como el calculo de $L(\mathbf{r})$ usando la siguiente expresión:

$$L(\mathbf{r}) = L_e(\mathbf{r}) + \int_D K(\mathbf{r}, \mathbf{s}) L'(\mathbf{s}) d\mu(\mathbf{s}) \quad (5.3)$$

donde L' es una aproximación a baja resolución de la función de radiancias, calculada en la primera pasada. El valor $L_e(\mathbf{r})$ se conoce para cualquier \mathbf{r} , luego el problema es

el calculo de la integral. Mediante el uso de los métodos de Monte-Carlo, usaremos una función de densidad de probabilidad $p_{\mathbf{r}}$ cumpliendo las siguientes propiedades:

$$\begin{aligned} 1 &= \int_D p_{\mathbf{r}}(\mathbf{s}) d\mu(\mathbf{s}) \\ K(\mathbf{r}, \mathbf{s}) L'(\mathbf{s}) > 0 &\Rightarrow p_{\mathbf{r}}(\mathbf{s}) > 0 \end{aligned}$$

Una vez que tenemos una $p_{\mathbf{r}}$ válida se puede diseñar un algoritmo de muestreo para aproximar el valor dado en (5.3). Esto se lleva a cabo seleccionando un conjunto de n rayos del entorno $\mathbf{s}_i = (y_i, w_i)$ con densidad de probabilidad $p_{\mathbf{r}}$. Después se calcula un estimador secundario mediante la siguiente variable aleatoria:

$$X_{\mathbf{r}}(\mathbf{s}_1, \dots, \mathbf{s}_n) = \frac{1}{n} \sum_{i=1}^n \frac{K(\mathbf{r}, \mathbf{s}_i) L'(\mathbf{s}_i)}{p_{\mathbf{r}}(\mathbf{s}_i)} \quad (5.4)$$

es fácil demostrar que $E(X_{\mathbf{r}}) = L(\mathbf{r})$, y por lo tanto $X_{\mathbf{r}}$ es un estimador sin sesgo de $L(\mathbf{r})$. El error obtenido en la aproximación depende de la varianza, que a su vez depende de $p_{\mathbf{r}}$ y del número de muestras n . Podemos disminuir dicha varianza seleccionando una $p_{\mathbf{r}}$ apropiada. La "mejor" *pdf* es aquella que presenta una varianza nula. Dicha función es la siguiente:

$$p_{\mathbf{r}}(\mathbf{s}) = \frac{K(\mathbf{r}, \mathbf{s}) L'(\mathbf{s})}{(\mathcal{T}L')(\mathbf{r})}$$

Este ejemplo de $p_{\mathbf{r}}$ cumple las condiciones (5.4) y (5.4), y además tiene varianza nula ya que es proporcional a $K(\mathbf{r}, \mathbf{s}) L'(\mathbf{s})$.

5.4.2 Una *pdf* para refinamiento final.

El objetivo es obtener una $p_{\mathbf{r}}$ óptimo en el sentido de que aproxima lo más posible el ideal, como se ha establecido en el apartado anterior. Esto puede llevarse a cabo usando la información almacenada en la estructura de enlaces. El conjunto de enlaces que salen de un nodo terminal representan una aproximación discreta a la distribución de la energía que llega a dicho nodo (ponderada por su importancia respecto al observador).

Por lo tanto, podemos usar una *pdf* que sea proporcional a dicho valor, con el objetivo de reducir la varianza. Esto implica que, si existe un enlace desde un nodo i y hacia otro nodo j con peso r_{ij} , entonces haremos $p_{\mathbf{r}}(\mathbf{s})$ proporcional a r_{ij} para todos los rayos \mathbf{r} con origen en el nodo i (y que apunten hacia el observador) y todos los rayos \mathbf{s} con origen en el nodo j (y que apunten hacia el origen de \mathbf{s}).

Necesariamente (debido al algoritmo de búsqueda usado) el enlace de i a j es un enlace que no ha sido dividido, esto es, no existe ningún enlace desde i hacia ningún descendiente de j . Por lo tanto, los pesos de los enlaces no son suficientes para caracterizar completamente la función $p_{\mathbf{r}}$, ya que no dan información sobre el trozo de superficie (o conjunto de objetos) cubierto por el nodo j . Una vez que el nodo fuente j ha sido seleccionado, es aún necesario descender el árbol hasta

encontrar un nodo terminal, y después debemos seleccionar un punto en el trozo de superficie cubierto por dicho nodo terminal.

Todo esto se puede expresar formalmente con la siguiente definición de $p_{\mathbf{r}}(\mathbf{s})$

$$p_{\mathbf{r}}(\mathbf{s}) = \frac{r_{ij}}{r_i} t_{\mathbf{rs}}(j, k) q_{k\mathbf{r}}(\mathbf{s}) \quad (5.5)$$

donde i es el índice del nodo terminal donde \mathbf{r} reside, k es el índice del nodo terminal donde \mathbf{s} reside, j es el primer ancestro de k tal que existe un enlace desde i hacia j , r_{ij} es el peso de dicho enlace, $q_{k\mathbf{r}}(\mathbf{s})$ es la probabilidad condicional de seleccionar punto de muestreo \mathbf{s} sabiendo que el nodo k ha sido seleccionado. Finalmente $t_{\mathbf{rs}}(j, k)$ da la probabilidad condicional de seleccionar k como trozo fuente de muestreo sabiendo que antes se ha seleccionado j . La función t se define iterativamente como sigue:

$$t_{\mathbf{rs}}(j, k) = \begin{cases} 1 & \text{si } j = k \\ f_{\mathbf{r}}(j, h) t_{\mathbf{rs}}(h, k) & \text{en otro caso} \end{cases}$$

donde h es el índice del nodo hijo de j que cubre el origen de \mathbf{s} y $f_{\mathbf{r}}(j, h)$ es una función que da la probabilidad condicional de seleccionar h sabiendo que hemos seleccionado j (esto es, la probabilidad de descender desde j hacia h)

Una vez que se conoce \mathbf{r} es necesario seleccionar un conjunto de puntos de muestra \mathbf{s} , con distribución $p_{\mathbf{r}}(\mathbf{s})$. Cada uno de estos \mathbf{s} se selecciona siguiendo los pasos que se detallan a continuación.

1. Hacer i igual al índice del nodo donde \mathbf{r} reside.
2. Seleccionar un nodo j con probabilidad proporcional a r_{ij} .
3. Mientras el nodo j no sea un nodo terminal:
 - (a) Seleccionar un nodo hijo de j (con índice h), con probabilidad $f_{\mathbf{r}}(j, h)$.
 - (b) Hacer j igual a h .
4. Seleccionar un rayo \mathbf{s} , con origen en el nodo j , con densidad de probabilidad $q_{j\mathbf{r}}(\mathbf{s})$.

En el siguiente apartado se detallan las funciones $f_{\mathbf{r}}$, que gobierna el descenso del árbol, y $q_{k\mathbf{r}}$ que se usa para seleccionar un punto en el interior del nodo k .

5.4.3 Descenso del árbol.

El algoritmo introducido en la sección anterior no está completo, en el sentido de que aún están sin especificar como son las funciones $f_{\mathbf{r}}$ y $q_{k\mathbf{r}}$.

Asumamos que j es un nodo no terminal con un nodo hijo h . El valor escalar $f_{\mathbf{r}}(j, h)$ es la probabilidad condicional de que el punto de muestreo resida en h , sabiendo que reside en j . La mejor estrategia sería usar el $f_{\mathbf{r}}$ definido como:

$$f_{\mathbf{r}}(j, h) = \frac{L_h(\mathbf{r})}{L_j(\mathbf{r})} \quad (5.6)$$

donde $L_i(\mathbf{r})$ es la radiancia en \mathbf{r} debido a la energía total proveniente de todo el área cubierta por el nodo i . Esta cantidad es difícil de conocer debido a la complejidad del operador de transporte. Con el objetivo de aproximar el valor en la ecuación (5.6) usamos una aproximación a L_i relativamente fácil de calcular, con el fin de evitar tiempos de ejecución muy largos, y también lo suficientemente cercana al valor exacto para reducir la varianza. En nuestro caso hemos usado la siguiente aproximación:

$$L_i(\mathbf{r}) \approx L'_i(\mathbf{r}) = f_r(x, w, w') \Phi_i \cos(n_x, w) s(x, i)$$

donde $\mathbf{r} = (x, w)$, f_r es la función BRDF, w' es el vector $c_i - x$ normalizado, Φ_i es la energía total de todas las partículas que abandonan el nodo i , y c_i es un punto considerado *centro* del nodo i , y que se define como sigue: si i es elemento plano de superficie, c_i es el centro de masas de dicho trozo, y en cualquier otro caso (el nodo i es un objeto o un grupo de ellos), c_i es el centro de su caja englobante. Finalmente, el valor $s(x, i)$ es una aproximación al ángulo sólido total cubierto por el nodo i cuando se proyecta sobre la semiesfera alrededor de x .

En el caso de que i sea un trozo plano con área a y normal n_i , aproximamos $s(x, i)$ asumiendo que el trozo i puede sustituirse por un disco plano de área a y radio r y cuya normal apunta a x . Este ángulo sólido es exactamente r^2/d^2 donde $d = |c_i - x|$. Dicho valor se multiplica por $\cos(n_i, -w')$, con el objetivo de tener en cuenta la reducción en el ángulo sólido debido a la inclinación relativa del trozo (su normal no tiene por que apuntar a x). Por lo tanto, para un trozo plano i , $s(x, i)$ se define como:

$$s(x, i) = \cos(n_i, -w') \frac{r^2}{d^2}$$

en el caso de que el nodo i sea un grupo de objetos, hacemos $s(x, i) = r^2/d^2$, donde d tiene la misma definición que antes, y r es el radio de una esfera cuya área coincide con el área total de todas las superficies en el grupo.

Finalmente, el valor $q_{k\mathbf{r}}(\mathbf{s})$ es la probabilidad de seleccionar un rayo fuente \mathbf{s} dentro del nodo terminal k . Para implementar esta probabilidad, usamos la parametrización de los triángulos esféricos descrita en [Arvo95], tal y como se detalló en el capítulo anterior.

5.5 Implementación y resultados.

Para probar los algoritmos descritos hemos diseñado una escena muy sencilla compuesta por un emisor y una superficie reflectora tipo Phong. El exponente de Phong se ha fijado al valor 2. En la figura 5.9 se muestran tres imágenes de referencia obtenidas con path-tracing simple, usando un número de muestras por pixel igual a 4, 16 y 32, de izquierda a derecha.

Por otro lado, en la figura 5.10 se pueden observar varias imágenes de la misma escena obtenidas con el método propuesto en este capítulo. Para visualizar estas imágenes se lanzaron 200.000 partículas desde la fuente de luz. El valor n_{min} se fijó a 100, y n_{max} a 100.000. El conjunto de imágenes puede observarse en la figura .

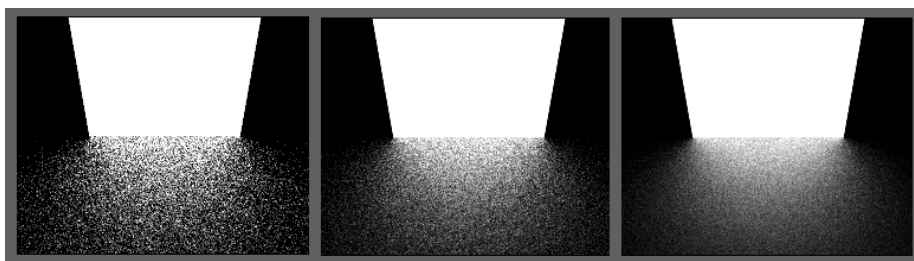


Figura 5.9: Path-tracing con 4,16 y 32 muestras por pixel

El número de muestras para refinamiento final es 1,2 y 4 para la fila de arriba (de izquierda a derecha) y 8,16 y 32 en la fila de abajo.

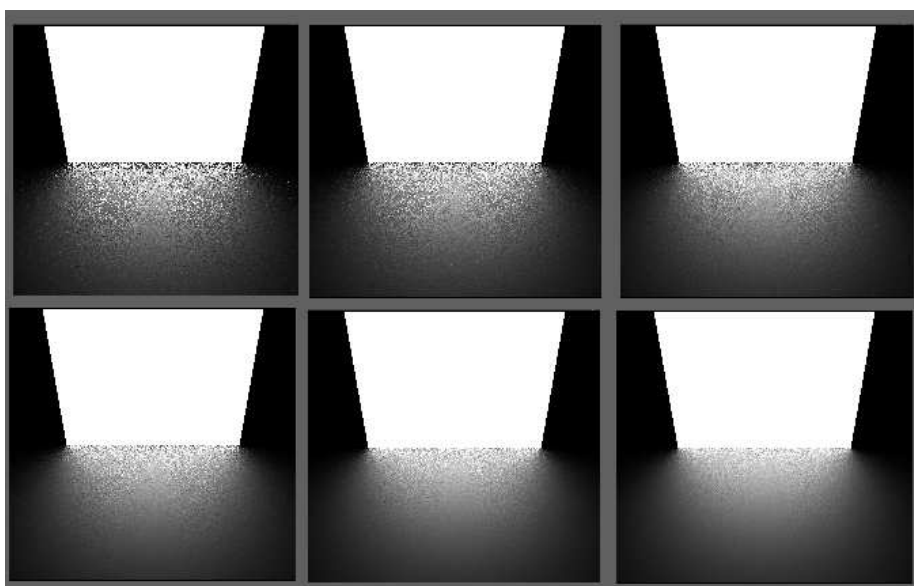


Figura 5.10: Nuevo metodo con 1,2,4,8,16 y 32 muestras por pixel.

Como puede observarse la varianza en el reflector es mayor para los puntos que están cerca del emisor, y muy baja para los puntos alejados del mismo. Esto se debe a que el factor de forma de punto a punto varía mucho en los puntos cercanos a la arista en común, mientras que para los puntos alejados dicho factor presenta menos variaciones. Se puede observar como en la región alejada la varianza es tan pequeña que apenas es observable incluso con una sola muestra por pixel. Esto implica que el algoritmo para descenso del árbol calcula los respectivas influencias con una gran aproximación si los puntos no están muy cercanos. En cualquier caso,

las varianza es menor que la que se obtendría usando muestreo de la BRDF para escoger las direcciones.

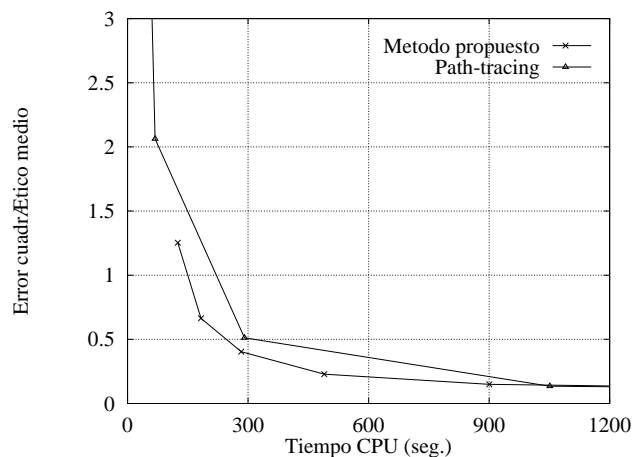


Figura 5.11: Comparación de métodos: error cuadrático frente a tiempo de CPU.

Con las baterías de imágenes citadas anteriormente se han realizado comparaciones numéricas del error. Se ha tomado como referencia una imagen obtenida con path-tracing usando 1024 muestras por pixel, sobre la misma escena. Se han calculado el error cuadrático medio (media de las diferencias en cada pixel al cuadrado) de imágenes obtenidas con path-tracing usando 1,4,16,64 y 256 muestras por pixel, respectivamente. Por otro lado se ha obtenido el error cuadrático medio usando el algoritmo propuesto con 1,2,4,8,16,32 y 64 muestras para refinamiento final. La gráfica en la figura 5.11 muestra una comparación de dicho error en las dos series, con respecto al tiempo total de CPU empleado (incluyendo, en el caso del algoritmo propuesto, el tiempo de la primera pasada). Como vemos, el algoritmo propuesto presenta menos error que path-tracing.

Como ejemplo de uso del método, en la figura (5.12) se observa una escena compleja (formada por dos conjuntos de sillas y mesas), obtenida usando path-tracing y el nuevo método propuesto. En la solución por path-tracing se usaron 256 cadenas por pixel y se emplearon 11 horas de tiempo de CPU, mientras que en la del nuevo método se usaron 4 muestras primarias por pixel, y 50 muestras para final gather (equivalente a unas 200 muestras por pixel). El tiempo empleado fue de unas 7 horas de CPU. Como vemos, la varianza es mucho menor para el nuevo método.

Sobre esta misma escena (el conjunto de dos mesas y sillas), se han sintetizado imágenes usando path-tracing con 1,4,16,64 y 256 cadenas por pixel, y con el nuevo método usando 8,16,32,64 y 128 muestras por pixel. En la figura 5.13 se observan dos gráficas. En el eje X esta el tiempo de CPU en segundos, y en el eje Y la raíz del error cuadrático medio de las imágenes, o desviación típica (tomando como

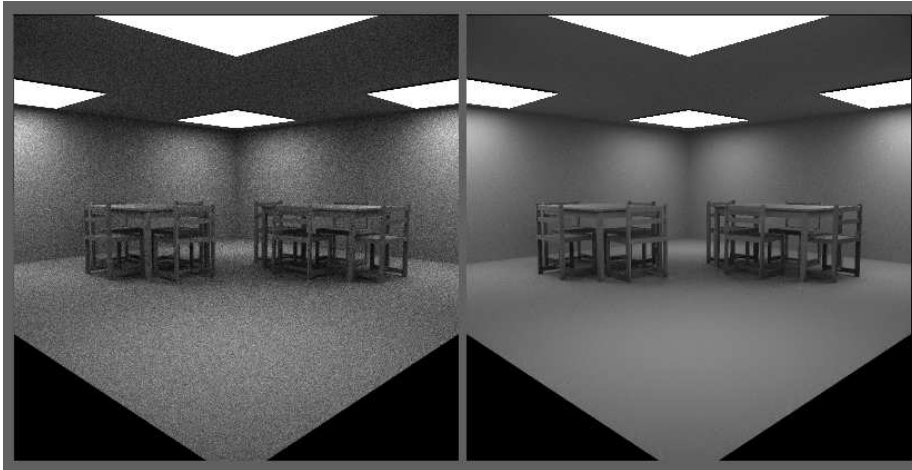


Figura 5.12: Una escena difusa compleja con Path-tracing y el nuevo método.

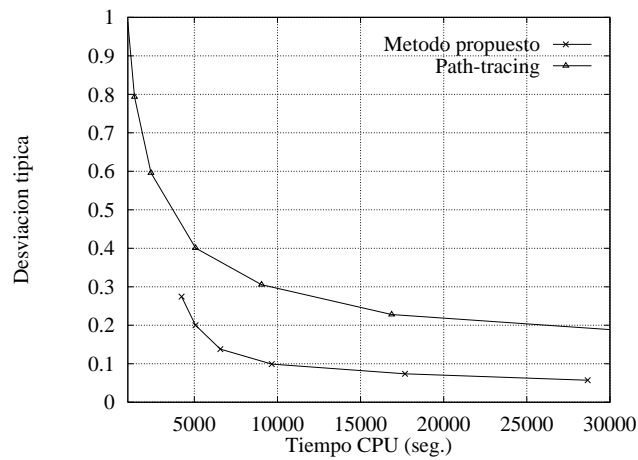


Figura 5.13: Comparación de eficiencia para la escena de dos mesas.

referencia, para cada método, la última imagen de su serie). Se observa como la eficiencia del nuevo método es claramente superior.

En la figura 5.14 se observa una escena compleja con una serie de conjuntos de mesas y sillas. Esta escena se ha sintetizado usando el nuevo método con 256 muestras para refinamiento final, y 4 muestras primarias por pixel. El número de partículas (de izquierda a derecha) fué de 3,6 y 12 millones. Se puede observar como las patas de las mesas y las sillas son más oscuras en la imagen a la izquierda y más claras a la derecha. Esto se debe a que estos objetos, muy pequeños, no han

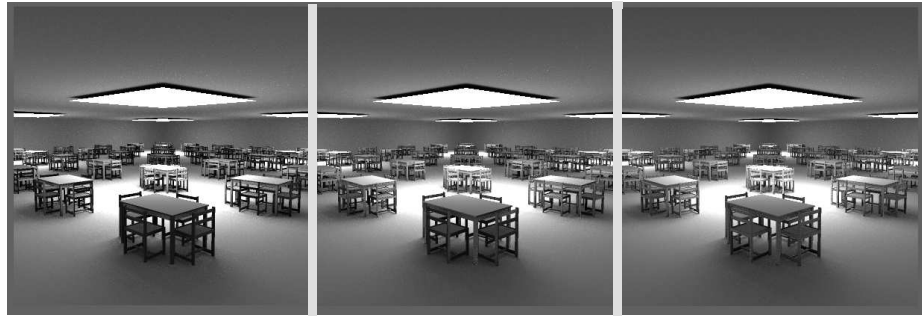


Figura 5.14: Escena compleja con 3,6 y 12 millones de partículas.

recibido un número suficiente de partículas en la simulación con 3 millones, pero si en la simulación con 12 millones. Esto provoca que las estimaciones de los pesos de los enlaces originados en ellos sean peores en el primer caso y mejores en el segundo, con lo cual la calidad de la imagen así obtenida varía.

5.6 Conclusiones.

Finalmente, en el último capítulo se detalla una técnica completamente nueva para extender el algoritmo anterior a entornos de radiosidad. Aquí se usa un método ya existente de creación de jerarquías de trozos y objetos, pero aplicada en un algoritmo de Monte-Carlo. Dichas jerarquías se han usado previamente solo para métodos de elementos finitos. Relacionado con este algoritmo, se propone una forma de actualizar la estructura de enlaces, necesariamente distinta a las existentes para métodos de elementos finitos.

Además, en este quinto y último capítulo se propone una nueva forma de muestreo por importancias en la segunda pasada, que usa la información almacenada en los enlaces y a su vez la jerarquía de objetos. En la simulaciones que se han realizado se demuestra como esta forma de muestreo puede reducir enormemente la varianza.

Capítulo 6

Principales aportaciones y conclusiones.

En este capítulo se detallan las conclusiones y aportaciones más relevantes realizadas en la presente memoria.

Como conclusión de esta memoria de tesis cabe decir que se han propuesto técnicas específicas que mejoran el rendimiento de las técnicas de Monte-Carlo. Tradicionalmente se considera a estas técnicas como poco eficientes frente a las de elementos finitos, lo cual ha limitado bastante la investigación en ellas y su uso práctico. Sin embargo en nuestra opinión, y para la clase de problemas a los que se enfrenta la visualización realista, en muchos casos las técnicas de Monte-Carlo pueden ser más eficientes que las técnicas de elementos finitos. Esto se debe fundamentalmente a que es un problema planteado en un espacio de muchas dimensiones. Aunque un uso de técnicas de Monte-Carlo *en bruto* es efectivamente poco eficiente, el análisis de la varianza y la propuesta de mejores algoritmos contribuye a la obtención de algoritmos eficientes.

En el capítulo primero se presenta una derivación de la ecuación de radiancias en forma de una ecuación integral de segundo orden. Para ello damos la expresión adecuada del núcleo del operador de transporte K . Esta formulación presenta la ventaja de que las herramientas matemáticas existentes para dicha clase de ecuaciones se pueden usar fácilmente en el contexto de la Iluminación Global. Las formas anteriormente propuestas de dicha ecuación no presentan exactamente esta característica, bien por que el núcleo es una función de tres variables [Kajiya86], bien porque el operador de transporte aparece como la composición de otros dos [Arvo95]. Además de presentar la ecuación de esta forma, en este primer capítulo se detallan los fundamentos de los métodos de elementos finitos (fundamentalmente el método Galerkin) y de los métodos de Monte-Carlo. Esta introducción es fácil al usar la nueva forma propuesta para la ecuación de rendering.

En el capítulo segundo se presenta una formalización del problema de la Iluminación Global, introduciendo el concepto de observador como un conjunto de funciones que miden la radiancia en determinadas partes de la escena. Combinando

este concepto con los introducidos en el capítulo anterior, obtenemos una imagen como el resultado de aplicar un operador lineal sobre la función de radiancia emitida. Otra aportación de este capítulo es una clasificación de los distintos métodos de cálculo de iluminación como diferentes formas de aplicar dicho operador lineal a la función de radiancias emitidas. Esto permite comparar entre si un conjunto amplio de publicaciones en este campo.

En el capítulo tercero se analiza la varianza de los métodos de Monte-Carlo para ecuaciones integrales de segundo orden. Se presenta una función de medida de probabilidad sobre el espacio de todos los caminos posibles, lo cual permite explicar los métodos de Monte-Carlo como el muestreo de una variable aleatoria definida con respecto a dicha medida. Esto da la posibilidad además de expresar su media y la varianza como una integración en el espacio de los caminos. A partir de aquí se caracteriza completamente la varianza de dichos métodos. En concreto, se obtiene dicha varianza como la solución de una ecuación integral de segundo orden.

Si bien este último resultado ya había sido previamente obtenido [Mikhailov92], no había sido puesto en relación con un conjunto de problemas de la Iluminación Global que se pueden resolver por Monte-Carlo, como nosotros hacemos en esta memoria. Finalmente, se obtiene un resultado completamente original que caracteriza de una forma parecida la varianza cuando se usa path-tracing con muestreo directo de las fuentes, técnica habitual en Iluminación Global.

La principal aportación presente en el capítulo cuarto es la demostración de como se puede reutilizar la información de la primera pasada para guiar el proceso de muestreo en la segunda en entornos puramente difusos. Aquí se ha propuesto una técnica específica para realizar esto, distinta de las previamente existentes. Su eficiencia ha quedado probada al ser comparada con el path-tracing estándar.

Finalmente, en el último capítulo se detalla una técnica completamente nueva para extender el algoritmo anterior a entornos de radiosidad. Aquí se usa un método ya existente de creación de jerarquías de trozos y objetos, pero aplicada en un algoritmo de Monte-Carlo. Dichas jerarquías se han usado previamente solo para métodos de elementos finitos. Relacionado con este algoritmo, se propone una forma de actualizar la estructura de enlaces, necesariamente distinta a las existentes para métodos de elementos finitos.

Además, en este quinto y último capítulo se propone una nueva forma de muestreo por importancias en la segunda pasada, que usa la información almacenada en los enlaces y a su vez la jerarquía de objetos. En la simulaciones que se han realizado se demuestra como esta forma de muestreo puede reducir enormemente la varianza.

Como resumen, cabe citar como aportaciones originales más relevantes las siguientes:

- Una nueva expresión de la ecuación de rendering con un núcleo de dos variables.
- Una clasificación de los métodos de Iluminación Global.
- Una caracterización de la varianza para una amplia familia de métodos de Monte-Carlo.

- Un nuevo algoritmo de varianza reducida para *final-gather* por Monte-Carlo en entornos difusos.
- Un nuevo algoritmo de Monte-Carlo en dos pasadas para Iluminación Global, basado en una discretización jerárquica de la escena, incluyendo *clustering*.

Bibliografía

- [Alpert93] Alpert, B.K.: A Class of Bases in L^2 for the Sparse Representation of Integral Operators. *SIAM Journal of Mathematical Analysis*, 24(1):246-262, January 1993.
- [Arvo86] Arvo, J.: Backward Ray-Tracing. *Siggraph'86 Developments in Ray Tracing, Seminar Notes*, Agosto 1986.
- [Arvo90] Arvo, J., Kirk, D.: Particle Transport and Image Synthesis *Computer Graphics*, 24(4):63-66, August 1990. ACM Siggraph '90 Conference Proceedings.
- [Arvo95] Arvo, J.: Stratified Sampling of Spherical Triangles. *Computer Graphics Proceedings, Annual Conference Series*, 1995 pp 437-438
- [Arvo95a] Arvo, J.: The Role of Functional Analysis in Global Illumination. *Sixth Eurographics Workshop on Rendering*. Dublin, June, 1995.
- [Arvo95b] Arvo, J.: Stratified Sampling of Spherical Triangles. *Computer Graphics Proceedings, Annual Conference Series*, 1995 pp 437-438
- [Aupperle93] Aupperle, L., Hanrahan, P.: Importance and Discrete Three Point Transport. *Fourth Eurographics Workshop on Rendering*. 1993.
- [Baum89] Baum, D.R., Rushmeier, H.E., Winget, J.M.: Improving Radiosity Solutions Through the Use of Analytically Determined Form-Factors. *Computer Graphics*, 23(3):325-334, July 1989. ACM Siggraph '89 Conference Proceedings.
- [Beyer94] Beyer, M., Lange, B.: Rayvolution: An Evolutionary Ray Tracing Algorithm. *Proceedings of 5th EG Workshop on Rendering*. Darmstadt, June 1994.
- [Beylkin91] Beylkin G., Coifman R., Rokhlin V.: Fast Wavelet Transforms and Numerical Algorithms I. *Communications on Pure and Applied Mathematics*, 44:141-183, 1991.
- [Buckalew89] Buckalew, C., Fussel, D.: Illumination Networks: Fast Realistic Rendering with General Reflectance Functions. *Computer Graphics*, 23(3):89-98, 1989. ACM Siggraph'89 Conference Proceedings.

- [Christensen93] Christensen, P.H., Salesin, D.H., DeRose, T.D.: A Continuous Adjoint Formulation for Radiance Transport. *Fourth Eurographics Workshop on Rendering*. 1993.
- [Christensen94] Christensen, P.H., Stollnitz, E.J., Salesin, D.H., DeRose, T.D.: Wavelet Radiance. In *Proceeding of the Fifth Eurographics Workshop on Rendering*. Darmstadt, Junio 1994.
- [Christensen97] Christensen, P.H., Lischinski, D., Stollnitz, E.J., Salesin: Clustering for Glossy Global Illumination. *ACM Transactions on Graphics*, Vol.16, n.1, January,1997, pp 3-33.
- [Cohen85] Cohen, M.F. Greenberg, D.P.: A Radiosity Solution for Complex Environments. *Computer Graphics*, 19(3): 31-40, 1985. ACM Siggraph'85 Conference Proceedings.
- [Cohen88] Cohen, M.F., Chen, S.E., Wallace, J.R., Greenberg, D.P.: A Progressive Refinement Approach to Fast Radiosity Image Generation. *Computer Graphics*, 22(4):75-84, 1988. ACM Siggraph'88 Conference Proceedings.
- [Cohen93] Cohen, M.F. Wallace J.R.: *Radiosity and Realistic Image Synthesis* Academic Press Professional, 1993.
- [Cook84] Cook, R.: Distributed Ray-Tracing. *Computer Graphics*, 18(3): pp.137-146, 1984. ACM Siggraph'84 Conference Proceedings.
- [Dutré94] Dutré, P., Willems, Y.D.: Importance-driven Monte Carlo Light Tracing *Fifth Eurographics Workshop on Rendering*. Darmstadt, 1994.
- [Glassner95] Glassner, A.S.: *Principles of Digital Image Synthesis*. Morgan Kaufmann Publishers, 1995.
- [Goral84] Goral, C.M., Torrance, K.E., Greenberg, D.P., Bennet, B.: Modelling the Interaction of Light Between Diffuse Surfaces. *Computer Graphics*, 18(3): 213-222, 1984. ACM Siggraph'84 Conference Proceedings.
- [Gortler93] Gortler, S., Schroder, P., Cohen, M.F., Hanrahan, P.: Wavelet Radiosity. *Computer Graphics*, ACM Siggraph'93 Conference Proceedings.
- [Gortler94] Gortler, S., Cohen, M.F., Slusallek, P.: Radiosity and Relaxation Methods (Progressive Refinement is Southwell Relaxation). *IEEE Computer Graphics & Applications* Noviembre'94.
- [Hanrahan91] Hanrahan, P., Salzman, D., Aupperle, L.: A Rapid Hierarchical Radiosity Algorithm. *Computer Graphics*, 25(4), July 1991. ACM Siggraph '91 Conference Proceedings.

- [Heckbert90] Heckbert, P.S.: Adaptive Radiosity Textures for Bidirectional Ray Tracing. *Computer Graphics*, 24(4), August 1990. ACM Siggraph '90 Conference Proceedings.
- [Immel86] Immel, D.S., Cohen M.F., Greenberg, D.P.: A Radiosity Method for Non-Diffuse Environments. *Computer Graphics*, 20(4):133-142, 1986. ACM Siggraph'86 Conference Proceedings.
- [Ito93] Itô, K., Ed.: *Encyclopedic Dictionary of Mathematics (two volumes)*. MIT Press, Cambridge, 1993.
- [Jensen95] Jensen H.W.: Importance Driven Path Tracing using the Photon Map. *Proceedings of the 6th EG Workshop on Rendering*, pp 349-368. Dublin, June 12-14, 1995.
- [Kajiya86] Kajiya, J.T.: The Rendering Equation. *Computer Graphics*, 20(4):143-150, 1986. ACM Siggraph'86 Conference Proceedings.
- [Kalos86] Kalos, M.H., Whitlock, P.A.: Monte Carlo Methods. Volume I: Basics John Wiley & Sons, 1986.
- [Kok91] Kok, A.J.F., Jansen F.W.: Source Selection for the direct Lighting Computation in Global Illumination. *Proceedings of the 2nd EG Workshop on Rendering*, Barcelona 1991.
- [Lafortune94] Lafortune, E.P., Willems, Y.D.: A Theoretical Framework for Physically Based Rendering. *Computer Graphics Forum* 13(2), June 1994.
- [Lafortune94a] Lafortune, E.P., Willems, Y.: Using the Modified Phong Reflectance Model for Physically Based Rendering. Report CW 197, November, 1994. Department of Computing Science, K.U. Leuven.
- [Lafortune97] Lafortune E.P., Foo S., Torrance K.E., Greenberg D.P.: Non-Linear Approximation of Reflectance Functions. SIGGRAPH 97 Conference Proceedings, Los Angeles, California, August 1997.
- [Languenou89] Languénou, E., Bouatouch, K., Tellier, P.: An adaptive Discretization Method for Radiosity. *Computer Graphics Forum* 11(3), September 1992.
- [Lewis93] Lewis, R.: Making Shaders more physically plausible. Proceedings of the Fourth EG Workshop on Rendering. Paris, France, 14-16 June, 1993. Also in *Computer Graphics Forum* 13(2), pp. 109-120, June, 1994.
- [Lischinski92] Lischinski, D., Tampieri, F., Greenberg D.P.: Discontinuity Meshing for Accurate Radiosity. *IEEE Computer Graphics & Applications*, November, 1992.

- [Mallat89] Mallat, S.G.: A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2(7):674-693, July 1989.
- [Malley88] Malley, T.J.: A Shading Method for Computer Generated Images. Master's Thesis, Dept. of Computer Science, University of Utah, June 1988.
- [Mikhailov92] Mikhailov, G.A.: *Optimization of Weighted Monte Carlo Methods*. Springer Series in Computational Physics. Springer Verlag, 1992.
- [Pattanaik92] Pattanaik, S.N., Mudur, S.P.: Computation of Global Illumination By Monte Carlo Simulation of The Particle Model of Light In *Proceedings of the Third Eurographics Workshop on Rendering*. Consolidation Express, Bristol, 1992.
- [Pattanaik93] Pattanaik, S.N., Mudur, S.P.: Efficient Potential Equation Solution for Global Illumination Computation. *Computer & Graphics*. 17(4):387-396, 1993.
- [Pattanaik94] Pattanaik, S., Bouatouch, K.: Haar Wavelet: A Solution to Global Illumination with General Surface Properties. In *Proceeding of the Fifth Eurographics Workshop on Rendering*. Darmstadt, Junio 1994.
- [Phong75] Phong, B.: Illumination for computer generated images. *Communications of the ACM*. 18(6):311-317, 1975.
- [Reichert92] Reichert, M.C.: *A two pass radiosity method driven by lights and viewer position*. Master's thesis, Program of Computer Graphics, Cornell University, Jan. 1992.
- [Rubinstein81] Rubinstein, R.Y.: *Simulation and the Monte Carlo Method*. Jhon Wiley & Sons, 1981.
- [Rushmeier88] Rushmeier, H.E.: *Realistic Image Synthesis for Scenes with Radiatively Participating Media*. PhD thesis, Program of Computer Graphics, Cornell University, 1988.
- [Rushmeier93] Rushmeier, H., Patterson, C., Veerasamy, A.: Geometric Simplification for Indirect Illumination Calculations. *Proceeding of Graphics Interface 93*, Canadian Information Processing Society, 1993.
- [Sbert93] Sbert, M.: An Integral Geometry Based Method for Fast Form Factor Computation. *Computer Graphics Forum*. 12(3):409-420, 1993. Eurographics '93 Conference Proceedings, Barcelona, September 1997.
- [Sbert97] Sbert, M.: Error and Complexity of Random-Walk Monte-Carlo Radiosity. *IEEE Transactions on Visualization and Computer Graphics* 3(1), March 1997.

- [Sbert97b] Sbert, M.: Optimal Source Selection in Shooting Random Walk Monte Carlo Radiosity. *Computer Graphics Forum*. 16(3), 1997. Eurographics '97 Conference Proceedings, Budapest, September 1997.
- [Schroder93] Schroder, P., Gortler, S.J, Cohen, M.F.: Wavelet Projections for Radiosity. *Proceedings of the Fourth Eurographics Workshop on Rendering*, Paris, Junio 1993.
- [Schlick94] Schlick, C.: A Survey of Shading and Reflectance Models. *Computer Graphics Forum* 13(2). pp 121-132. Blackwell, 1994.
- [Schroeder94] Schroeder, P., Hanrahan, P.: Wavelets Methods for Radiance Computations. *Proceedings of the Fifth Eurographics Workshop on Rendering*, Darmstadt, Junio 1993.
- [Shao88] Shao, M., Peng, Q., Liang, Y.: A New Radiosity Approach by Procedural Refinements for Realistics Image Synthesis. *Computer Graphics*, 22(4): 93-101, 1988. ACM Siggraph'88 Conference Proceedings.
- [Shirley90] Shirley P.: A Ray Tracing Method for Illumination Calculation in Diffuse-Specular Scenes. *Proceedings of Graphics Interface 90*, Canadian Information Processing Society, 1990.
- [Shirley91] Shirley P.: Time Complexity of Monte-Carlo Radiosity. *Eurographics'91 Conference Proceedings*. pp 459-465, September, 1991.
- [Shirley92] Shirley, P.S., Wang, C.: Distribution Ray-Tracing: Theory and Practice. In *Proceedings of the Third Eurographics Workshop On Rendering*. Consolidation Express, Bristol, 1992.
- [Shirley96] Shirley, P.S., Wang, C., Zimmerman, K.: Monte Carlo Techniques for Direct Lighting Calculations. *ACM Transactions on Graphics*, 15(1), January 1996. pp 1-36.
- [Sillion89] Sillion, F., Puech, C.: A general Two-Pass Method Integrating Specular and Diffuse Reflection. *Computer Graphics*, 23:(3):335-344, 1989. ACM'Siggraph'89 Conference Proceedings.
- [Sillion91] Sillion, F., Arvo, J.R., Westin S.H., Greenberg, D.P.: A global Illumination Solution for General Reflectance Distributions. *Computer Graphics*, 25(4):187-196, Julio 991. ACM Siggraph'91 Conference Proceedings.
- [Smits92] Smits, B.E., Arvo, J.R., Salesin, D.H.: An Importance-Driven Radiosity Algorithm. *Computer Graphics*, 25(4), July 1991. ACM Siggraph '91 Conference Proceedings.
- [Spanier69] Spanier, J., Gelbard, E.M.: *Monte Carlo Principles and Neutron Transport Problems*. Addison-Wesley 1969.

- [Urena92] Ureña, C., Parets, J., Torres, J.C., del Sol, V.: An Object-Oriented approach to Ray Tracing Image Synthesis Algorithm Implementation. *Computer & Graphics* 16(4). pp. 363-369. Pergamon Press, 1992.
- [Urena93] Ureña, C., Torres, J.C.: Un mtodo de Monte Carlo Bidireccional para Sntesis de Imgenes. *Actas del III Congreso Español de Informática Gráfica (CEIG'93)*. pp. 209-222. Granada, 1993.
- [Urena97] Ureña, C., Torres, J.C.: A Formalization and Classification of Global Illumination Methods. *Computer & Graphics* 21(2). Pergamon Press, 1997.
- [Urena97a] Ureña, C., Torres, J.C.: Improved Irradiance Computation by Importance Sampling. *Rendering Techniques'97*. Springer Verlag, Wien, 1997.
- [Urena97b] Ureña C., Torres, J.C., Revelles, J., Cano, P., del Sol, V., Cabrera, M.: Designing an Object-Oriented Rendering System. *7th Eurographics Workshop on Programming Paradigms in Graphics*, Budapest, Hungary, September, 1997.
- [Veach94] Veach, E., Guibas L.J.: Bidirectional Estimators for Light Transport. *Fifth Eurographics Workshop on Rendering*, Darmstadt, 1994.
- [Veach95] Veach, E., Guibas L.J.: Optimally Combining Sampling Techniques for Monte Carlo Rendering. *Computer Graphics Proceedings, Annual Conference Series*, 1995. pp. 419-428, 1995. ACM Siggraph'95 Conference Proceedings.
- [Veach97] Veach, E.: *Robust Monte Carlo Methods for Light Transport Simulation*. Ph.D. Thesis. Stanford University. December, 1997.
- [Wallace87] Wallace, J.R., Cohen, M.F., Greenberg, D.P. A Two Pass Solution to the Rendering Equation: A Synthesis of Ray-Tracing and Radiosity Methods. *Computer Graphics*, 21(4):311-320, July 1987. ACM Siggraph'87 Conference Proceedings.
- [Wallace89] Wallace, J.R., Elmquist, K.A., Haines, E.A.: A Ray Tracing Algorithm for Progressive Radiosity. *Computer Graphics*, 23(3):315-324, 1989. ACM Siggraph'89 Conference Proceedings.
- [Ward88] Ward, G.J., Rubinstein, F.M., Clear, R.D.: A Ray Tracing Solution for Diffuse Interreflection. *Computer Graphics*, 22(4):85-92, August 1988. ACM Siggraph'88 Conference Proceedings.
- [Ward91] Ward, G.J.: Adaptive Shadow Testing for Ray-Tracing *Proceedings of the 2nd EG Workshop on Rendering*, Barcelona, 1991.

- [Whitted80] Whitted, T., An Improved Illumination Model for Shaded Display. *Communications of The ACM*, 23(6):343-349, 1980.
- [Yu95] Yu, Y., Peng,Q.: Multiresolution B-Spline Radiosity. *Computer Graphics Forum*. 14(3):285-298, 1995. Eurographics'95 Conference Proceedings.
- [Zimmerman95] A Two-Pass Solution to the Rendering Equation with a Source Visibility Preprocess. *Proceedings of the 6th EG Workshop on Rendering*. Dublin, 1995.
- [Zatz93] Zatz, H.R.: Galerkin Radiosity. A Higher Order Solution Method for Global Illumination. *Computer Graphics Proceedings, Annual Conference Series*, 1993.