References

Faux I.D., Pratt M.J., Computational Geometry for Design and Manufacture, John Wiley & Sons, 1980.

Farin, G., Smooth interpolations of scattered 3D data, Surfaces in Computer Aided Geometric Design, Barnhill R.E., Boehm W. (eds), North Holland, 1983.

Nielson G.M., Franke R., A method for construction of surfaces under tension, Rocky Mountain Journal of Mathematics 14, 203-222, 1984.

Piegl L., Key developments in computer-aided geometric design, Computer-Aided Design 21, 262-273, 1989.

Farin G., Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide, Academic Press, New York, 1989.

Waren J., Creating Multisided Rational Bezier Surfaces Using Base Points, ACM Transactions on Graphics, Vol 11, No 2, 127-139, 1992.

Feriancova M., Design of Multisided Rational Bezier Patches, Conference Proceedings of Computer Graphics '93, Budmerice, Slovakia, June 2-4, 1993.

Cubic Monte Carlo Radiosity

Pavol Eliáš, Martin Feda, Peter Ferschin, Werner Purgathofer

Department of Applied Mathematics,
Faculty of Mathematics and Physics, Comenius University,
842 15 Bratislava, Slovakia
e-mail: elias@delta.dcs.fmph.uniba.sk

Institute of Computer Graphics
Technical University Vienna
Karlsplatz 13/186, A-1040, Vienna, Austria
email: elias@eigvs4.una.ac.at

Abstract

A revised radiosity method for curved surfaces is proposed, based on the Monte Carlo approach. In order to improve the accuracy of the solution, a smoothly reconstructed illumination function with selected discontinuities is used during the radiosity computation. The reconstructed function is used as a random number distribution for position sampling to overcome the constant radiosity assumption syndrome. Illumination information stored at the surface control points is used to preserve continuity of the illumination across the boundary of adjacent surfaces and to avoid Mach band effects. Implementation in Flatland is discussed.

1.Introduction

Realistic simulation of illumination effects is often a primary goal in computer generated imagery. Because of the complexity of light behavior interacting with an environment, one can't hope for closed-form solutions to global illumination problems. The techniques for global illumination have to consider the basic interdependency of the light energy transfer problem: the radiance of every object is determined by the radiance of all other objects in the scene visible from this object. Radiosity methods have been shown to be an effective solution to the global illumination problem in diffuse environments. They have been introduced to computer graphics from the simulation of radiative heat transfer. The method is based on the principle of energy conservation. All light energy emitted within the scene is reflected off surfaces and transfered between surfaces within this environment. Thus, the basic building brick of the radiosity solution is the energy transfer between two surfaces as an instance of the rendering equation [7]. The energy transfer is expressed using the concept of formfactors [18].

To simplify the rendering equation and speed up computation, several approaches have been developed. Conventional radiosity requires the scene to be discretized into small areas (called patches) and approximates the illumination function over a patch as a piecewise constant. Much work has been spent to increase the accuracy of the solution by discontinuity meshing [9] and ray-tracing form-factor evaluation [18]. Further improvements can be achieved by introducing piecewise linear approximation of the illumination function [10] and extending this technique to higher order finite element methods [19] [16]. The demand for increasing visual quality of a result led to the development of post-processing techniques for reconstruction of the illumination function from Gouraud linear interpolation shading to higher order reconstruction [13], [2].

Another effort has been devoted to the invention of stachastic methods for radiosity and has resulted in Monte Carlo radiosity techniques [14], [12]. A difference to conventional radiosity is that instead light energy transfer is modelled by shooting photons (packets of energy) from emitting surfaces, instead of using form-factor computation. Photons are absorbed at the surface of the receiver and transfered energy is stored for rendering.

2.Higher order finite element methods and the reconstruction of the illumination functions

The conventional radiosity algorithm computes the energy transport at a collection of points and uses this information for a piecewise constant approximation to the illumination function. This leads to a "faceted" appearance of the rendered objects. Through Gouraud interpolation as a rendering post-process, a continous solution is achieved. A more accurate illumination function for each surface can be obtained using a higher number of patches. Thus, meshing is a crucial part of a radiosity pre-process. Lischinski et al. in [9] have introduced a discontinuity meshing, which means that surfaces are subdivided along the lines of discontinuity caused by shadow boundaries. The main disadvantage of this approach is an increasing computational complexity. Higher order radiosity methods (Galerkin Radiosity [16], [19] and The Point Collocation Method [16]) have exchanged a large number of low-order elements (patches) with a smaller number of higher order elements using methods of finite element analysis. Rather than assuming a piecewise constant illumination function, they use a set of basis functions defined on the element. Thus, the illumination function over a patch is described as a linear combination of the relative contributions of each basis function to the final radiosity value. In order to maintain energy transfers in the environment, basis function-to-basis function form-factors were introduced, instead of patch-to-patch form-factors of conventional radiosity. Although higher order finite element methods produce a more accurate

solution, there is still a lot of problems left for further research. For example, the Galerkin method does not mathematically guarantee continuity between adjacent coplanar surfaces. This leads to the complication using any of the well known meshing techniques. This is an important problem, because environments usually contain a lot of singularities, which can be simply handled by discontinuity meshing. Another possibility is to use basis functions of significantly higher orders, which is, on the other hand, computationally very expensive. Another significant disadvantage is shadow treatment. Shadow masking (suggested in [19]) is only a local approximation to the true shadow edge generation. If an important shadow is missing, a smooth illumination function over a patch will lead to a "wave like" appearance caused by Gibbs ringing.

To overcome discontinuity of the illumination function in the rendered picture, several post-processing reconstruction techniques have been developed. The first attempt to solve this problem uses Gouraud shading for smoothing the radiance value over a patch. Although this approach doesn't increase the accuracy of the radiosity solution, the rendered pictures are visually more pleasant. In order to eliminate Mach banding, linear interpolation has been extendend to higher order interpolation methods (e.g. Bicubic Hermite Interpolation [2]) and reconstruction functions with selected discontinuities (e.g. Interpolation using Bezier triangles [13]). The main disadvantage of these techniques is their post-processing behavior. In contrast to higher order finite element methods, smooth reconstruction of the illumination function is not used during the radiosity solution, but as a blurring step for the rendering process only.

3.Revised Monte Carlo Radiosity

3.1 Problem Formulation:

- Given: A scene described as a set of Bezier triangles **T** with a set of Bezier control vertices **V**. A light sources described as a set of emittance functions $E_i: R^2 \rightarrow C$ for each Bezier triangle where C is a color space.
- To find: A set of functions $B_i: R^2 \rightarrow C$ for each Bezier triangle, such that:

$$B_i(X) = E_i(X) + \rho_i(X) \sum_i \int_\Omega B_i(X')G(X,X')dX'$$

where:

B - radiance (energy/unit time/unit area)
E - emittance (energy/unit time/unit area)
ρ - reflectivity (unitless)
G - geometry term (unitless)

3.2 The Monte Carlo Algorithm

The Monte Carlo algorithm is based on the energy transfer simulation using a finite number of samples called photons. It is a stochastic method, therefore it is only possible to predict the expected value and the variance of the converged radiosity solution. Shirley in [14] has proved that the computation of the radiosity solution with any variance requires only O(N) shooted photons (N is the number of patches in the scene), if the scene satisfies the following conditions:

1. The maximum radiosity in the scene is bounded.
2. The maximum reflectance in the scene is less than one.
3. The ratio of the largest to the smallest patch area is bounded.

These conditions are reasonable and make Monte Carlo approach competitive with other radiosity methods, especially for extremely fine meshed scenes.

Algorithm:

For each photon repeat steps 1-6 below:

1. Choose a light emitter, from which a photon will be shot (each ray transports approximately the same amount of energy and hence has the same effect on the radiosity solution, therefore it is not necessary to select the patch which emits the most unshot energy within the scene. Nonetheless, the selected patches should be equally distributed according to the remaining unshot light energy).
2. Reconstruct the illumination function over the emitting surface using piecewise cubic interpolation (reconstruction is done using a method introduced in [13] and is described in detail in chapter 3.3).
3. Choose a random position on the light emitter using a position sampling of the emitter surface geometry according to the reconstructed radiosity distribution (this is a crucial part of the revised Monte Carlo algorithm and therefore it is described in detail in chapter 3.4).
4. Choose a random direction in which the photon is emitted by directional sampling of the emission distribution of the emitter (a good discussion of sampling approaches for different surface geometries can be found in [12]).
5. Compute the photon's energy. (energy is given by the product of the total unshot radiosity of the emitter and the emitter's area divided by the number of rays emitted).
6. Find the nearest patch hit by the ray and store the transfered energy at the receiving patch (this is simply done using the ray-casting method. Some improvements have been proposed in [17] and are described in chapter 3.5).

3.3 Reconstruction of the illumination functions with selected discontinuities using Bezier triangles

The complete reconstructing algorithm is described in [13], so this paper presents only a summary of the method. The original paper proposes an interpolation technique for triangular meshing, although this method can be easily extended for curved surfaces modelled using Bezier triangles. Instead of interpolating across a real triangle, reconstruction is performed in parametrical space.

Problem formulation:

- Given: A triangulation with vertex set, edge set and triangular face set. Intensity and normal vectors of intensity for every vertex. A continuity flag C, for every edge in triangulation.
- To find: An interpolation function over a triangulation with continuity C^1 everywhere, except of edges with continuity flag C^{-1}, C^0.

A cubic Bernstein-Bezier polynomial P defined over a triangle (u,v,w) is given by the equation:

$$P(\beta_u,\beta_v,\beta_w) = \sum_{\substack{0\le i,j,k\le 3\\ i+j+k=3}} \frac{3!}{i!\,j!\,k!}\beta_u^i\beta_v^j\beta_w^k b_{ijk}$$

where: β_u,β_v,β_w are the barycentric coordinates with respect to the domain triangle, and the scalar values b_{ijk} are called the Bezier ordinates of P.

According to [13] we will denote the Bezier ordinates as follows:

$uuu = b_{300}$	$vvv = b_{030}$	$www = b_{003}$		
$uuv = b_{210}$	$uvv = b_{120}$	$uuw = b_{210}$		
$uww = b_{102}$	$vvw = b_{021}$	$vww = b_{012}$	and	$uvw = b_{111}$

The Clough-Tocher interpolant (see Fig.1), which ensures a C^1 (continuously differentiable) continuity everywhere across the surface, is based on splitting each triangle at the centroid into three subtriangles. A cubic Bernstein-Bezier polynomial is defined for each subtriangle. The construction requires the following constraints:

1. The ordinate vvv for each vertex v is set to correct intensity value for this vertex.
2. The ordinate uvv for each ordered edge uv must lie in tangent plane defined by intensity normal vector in vertex v.
3. The points (uuv,uvv,cuv,c'uv) must lie in the same plane, for each edge uv whose adjoining triangles have centroids c and c'.
4. The quadruples (cuu,uuu,uuv,uuw),(ccu,cuu,cuv,cuw), and (ccc,ccu,ccv,ccw) must each lie in the same plane, for each ordered triangle (u,v,w) with centroid c.

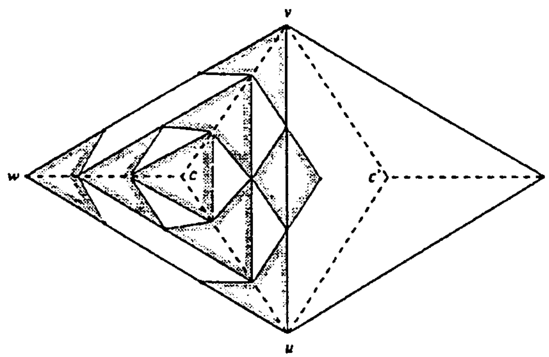

Figure 1. The Clough-Tocher construction

Discontinuities can be introduced as follows:

- Suppose that vertices u and v should be C^1, but edge uv should be C^0. In this case, the coupling between cuv and c'uv is removed by eliminating constraint (3).
- Suppose that vertex v should also be C^0. In this case, constraint (2) is removed. Each ordinate vvx for any vertex x in the original triangulation can be set to any arbitrary value.
- If an edge uv is to be discontinuous in position C^{-1}, then the ordinates uuv and uvv split into two independent values each.
- If a vertex v is also to be C^{-1}, then the ordinate vvv splits into different values for each triangle incident at v.

Complete reconstructing algorithm can be found in [13].

3.4 Positional and Directional sampling for revised Monte Carlo Radiosity.

Directional emissivity of the light source can be easily supported by the Monte Carlo radiosity approach. The simplest way is to use it as a random number distribution for directional sampling of photons. In diffuse environments, the emitted intensity is uniform in all directions. Thus, for the photon emitted from a diffuse emitter, the direction is given by the pair $(2\pi\xi_1, \arcsin(\sqrt{\xi_2}))$, where ξ_1, ξ_2 are uniform random variables in the range <0,1>.

If an assumption about uniform emission intensity over the emitter surfaces is made (that means, the radiosity function is constant), then photons must be generated uniformly over the surface. Therefore, the proper sampling strategy for arbitrary surface geometry is needed. In the radiosity method for scenes consisting of Bezier triangles a sampling based on

parametric space can be used. For each Bezier triangle, one can consider the domain triangle P_0, P_1, P_2 sampled uniformly by the following formula: $P_0 + (1-v)u(P_1 - P_0) + v(P_2 - P_0)$, where u is a uniform random number in the range <0,1> and v is sampled as $(1-\sqrt{\xi})$, where ξ is another uniform random number in the range <0,1>. The sample is projected from parametric space onto the Bezier triangle as a photon origin. The derivation of the directional distribution and positional distribution for various geometries can be found in [12].

The problem is more complicated for emitters with non-constant illumination over the surface. Instead of uniform sampling of the domain triangle, sampling can be dependent on the illumination function of the surface. Thus, the reconstructed radiosity function must be used as a random number distribution for positional sampling. That means, the density of photons' origins is higher in those parts of the emitter, which have a greater illumination. This method allows us to overcome a constant radiosity assumption syndrome, which is described in [15]. This is a basic difference to well known illumination reconstructing methods, which used the additional radiosity information only in the post-processing step.

The derivation of the positional sampling with reconstructed illumination function can be done using the principle of transformation of the random variable. Let's consider a 2 dimensional parametric space and the illumination function B(x,y) defined over this space. Let's define a function C(x) as follows:

$$C(x) = \int_a^b B(x,y)dy$$

If function B(x,y) defines a probability of the sample located in the point (x,y), than the function C(x) represents a probability of different x-slices in parametric space (Note, that functions B(x,y) and C(x) are not normalized, that means they don't describe the probability directly, only a sampling importance). In order to choose the most probable x-slice, uniform random variable ξ_1 must be transformed in the following way:

$$\xi_1 = \frac{\int_c^u C(x)dx}{\int_c^d C(x)dx}$$

As a result u value for the parametric space is obtained. After choosing a x-slice, the same method can be applied to the selection of the v value for parametric space:

$$\xi_2 = \frac{\int_a^v B(u,y)dy}{\int_a^b B(u,y)dy}$$

Note, that ξ_1, ξ_2 are uniform random variables in the range <0,1>.

Using this approach, a non uniformly distributed 2 dimensional random value (u,v) is obtained. Because of the random value distribution based on the illumination function B(x,y), this method results in the exact position sampling for Monte Carlo radiosity with non-constant illumination function over the surface.

Another, more straightforward, method should use the reconstracted illumination function as a weight of the energy carried by a photon. An important disadvantage of this second approach is violation of the Monte Carlo radiosity assumption, that each ray carries the same amount of energy. In this case, time complexity proved to be O(N) by Shirley in [14], may not be achieved.

3.5 Ray-object intersection using extended rays.

In the standard Monte Carlo Radiosity papers, a ray is understood to be a one dimensional carrier of an amount of energy like a photon. Vesel et al. introduced in [17] an extended ray for Monte Carlo Radiosity. Each ray is considered to have a certain width and energy inside a ray is distributed according to some distribution function (see Fig. 2). Each sample point inside a ray describes a percentage of the whole energy carried by the ray, which belongs to the interval determined by the current sample point. An extended ray intersection with objects have to be considered. This problem can be solved with direct calculation of band of rays with an object intersection. Another approach for speeding up computation is to consider a single ray with an extended object intersection.

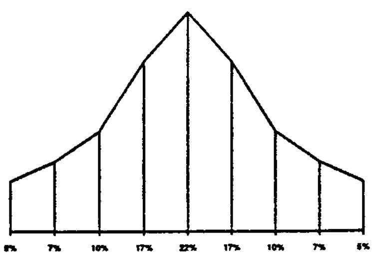

Figure 2. A typical energy distribution for the extended ray

After finding a ray-object intersection, the energy carried by the ray is stored at the surface of the object. In order to use extended rays, a way of energy distribution among the receiving surface control points must be defined. This is done in the following way:

- In the first step, control vertices which determine the hit surface are found out. Thus, a transformation from 3D to parametric space is needed.
- In the second step, the energy carried by the ray is distributed to all of the control vertices. The distance of the ray-object intersection to the control vertex can be used as a weight for energy distribution. It is important to note that the total transferred energy must be preserved. That means the sum of the energy received by the control vertices must be equal to the energy carried by the ray.

4. Implementation in Flatland

We have implemented the Monte Carlo algorithm using piecewise cubic illumination functions on a PC in C as an extension of an algorithm described by Vesel et al. in [17]. The crucial part of the algorith is a positional and directinal sampling. For positional sampling, a distribution function has been derived according to [12] in the following form:

$$\xi = \frac{\int_0^v B(x)dx}{\int_0^1 B(x)dx},$$

where v means a position of the sample point inside of the interval <0,1> and ξ is a uniformly distributed random value in the range <0,1>. The principle of transformation of the random variable has been used for derivation of positional sampling.

5. Summary

The revised radiosity algorithm takes advantage of the stochastic nature of Monte Carlo methods. Since the origin and the direction of each ray are selected randomly, it is possible to incorporate a smooth illumination function into the previous concepts, using it as a direction sampling distribution for shot rays. It is important to note that in the Monte Carlo simulation, the role of the meshing structure is only to store illumination information. Therefore computation time depends only on the number of simulated photons and doesn't depend on the size of the meshing. The Monte Carlo simulations of the particle model of light are intuitively simple and quite straightforward to implement. In spite of this, they are suitable to model global illumination effects for scenes consisting of curved surfaces even with a additional special demand on the solution (smooth illumination function, specular reflection and refraction, etc.). Another advantage is the possibility to use suitable acceleration techniques like spatial subdivision and progressive ray refinement (see [6]) which were very intensively studied in the context of ray tracing methods.

6. References

[1] H. Bao, Q. Peng: "A Progressive Radiosity Algorithm for Scenes Containing Curved Surfaces", Computer Graphics Forum, Vol. 12, No. 3, 1993.

[2] R. Bastos, A. De Sousa, F. Ferreira: "Reconstruction of Illumination Functions using Bicubic Hermite Interpolation", 4th Eurographics Workshop on Rendering, Paris, France, Eurographics Technical Report Series, 1993.

[3] S. Butt, K. Brodlie: "Preserving Positivity using Piecewise Cubic Interpolation", Computer and Graphics, Vol. 17, No. 1, 1993.

[4] P. Eliáš, M. Feda, P. Ferschin, W. Purgathofer: "Monte Carlo Radiosity using Piecewise Cubic Illumination Functions", submmited to Computer Graphics Forum, Special Issue on Rendering, 1993.

[5] G. Farin: "Triangular Bernstein-Bezier Patches", Computer Aided Geometric Design, No.3, 1986.

[6] M. Feda, W. Purgathofer: "Progressive Ray Refinement for Monte Carlo Radiosity", 4th Eurographics Workshop on Rendering, Paris, France, Eurographics Technical Report Series, 1993.

[7] J. Kajiya: "The Rendering Equation", Computer Graphics, Vol. 20, No. 4, 1986.

[8] A. Kok, C. Yilmaz, L. Bierens: "A Two-Pass Radiosity Method for Bezier Patches", Eurographics Workshop on Photosimulation, Realism and Physics in Computer Graphics, Rennes, France, Springer-Verlag, 1990.

[9] D. Lischinski, F. Tampieri, D. Greenberg: "Discontinuity Meshing for Accurate Radiosity", IEEE Computer Graphics & Applications, Vol. 12, No. 6, 1992.

[10] N. Max, M. Allison: "Linear Radiosity: Approximation using Vertex-to-Vertex Form Factors", Graphics Gems III., Academic press, 1992.

[11] T. Nishita, E. Nakamae: "A New Radiosity Approach using Area Sampling for Parametric Patches", Computer Graphics Forum, Vol. 12, No. 3, 1993.

[12] S. Pattanaik, S. Mudur: "Computation of Global Illumination by Monte Carlo Simulation of the Particle Light", 3rd Eurographics Workshop on Rendering, Bristol, United Kingdom, Eurographics Technical Report Series, 1992.

[13] D. Salesin, D. Lischinski, T. DeRose: "Reconstructing Illumination Functions with Selected Discontinuities", 3rd Eurographics Workshop on Rendering, Bristol, United Kingdom, Eurographics Technical Report Series, 1992.

[14] P. Shirley: "Time Complexity of Monte Carlo Radiosity", Eurographics'91, Vienna, Austria, 1991.

[15] F. Tampieri, D. Lischinski: "The Constant Radiosity Assumption Syndrome", 2nd Eurographics Workshop on Rendering, Barcelona, Spain, 1991.

[16] R. Troutman, N. Max: "Radiosity Algorithms Using Higher Order Finite Element Methods", Computer Graphics, Vol. 27,No. 4, 1993.

[17] A. Vesel, M. Feda, M. Gervautz, W. Purgathofer: "Radiosity Computation in Flatland using a Monte Carlo Approach with Extended Rays", Proceedings of the V. International Conference on Computer Graphics'93, Budmerice, Slovakia, 1993.

[17] Wallace, K. Elmquist, E. Haines:" A Ray Tracing Algorithm for Progressive Radiosity", Computer Graphics, Vol. 23, No. 3, 1989.

[18] H. Zatz: "Galerkin Radiosity: A Higher Order Solution Method for Global Illuminatior.", Computer Graphics, Vol. 27,No. 4, 1993.