Finite Element Methods for Global Illumination

Paul S. Heckbert
University of California at Berkeley[*]

James M. Winget
Silicon Graphics Computers, Inc.[†]

8 January 1991 [‡]

Abstract

The interreflection of light between surfaces is governed by an integral equation. Existing radiosity algorithms approximate the solution of this integral equation by transforming it into a system of linear equations. It is shown that such algorithms are simple applications of the finite element method.

Techniques are presented for applying more advanced finite element techniques to the global illumination problem in order to yield more accurate results. First, piecewise-linear, piecewise-quadratic, and higher order elements are discussed as a superior alternative to current piecewise-constant radiosity assumptions. Second, Galerkin techniques are a more robust alternative to current point collocation (point sampling) techniques. Finally, occlusions in a scene give rise to discontinuities such as shadow edges in the solution function. ***Discontinuity meshing*** is introduced as a technique for resolving these discontinuities by adaptive placement of element boundaries. Illustrations, algorithms, and results are given for two-dimensional ***radiosity in flatland*** problems.

1 Introduction

Many applications in computer graphics require realistic image synthesis. Lighting design for architectural CAD, product design, and special effects for entertainment all strive toward realistic simulation of illumination in complex three-dimensional scenes. One of the most difficult aspects of realistic image synthesis is the accurate and efficient simulation of ***global illumination effects***: the illumination of each object by every other object in the scene.

Previous methods for global illumination generally fall into three classes: ray tracing methods, radiosity methods, and hybrid methods. Ray tracing algorithms are generally best suited to scenes of specular reflecting and transmitting surfaces [Whi80], while radiosity methods are generally limited to diffuse scenes [SH81,GTGB84,NN85,CG85]. Ray tracing methods can be generalized to diffuse environments [Kaj86,WRC88], and radiosity methods can be generalized to specular environments [ICG86], but to date these extensions have not resulted in very efficient algorithms. The third class of algorithms are hybrids of ray tracing and radiosity techniques, typically employing multiple passes [WCG87,SP89,Shi90,Hec90].

These methods can be viewed as numerical approximation methods for solving the integral equation governing light transport. This integral equation is called the ***rendering equation*** in the computer graphics literature [Kaj86], or the ***mutual illumination equation*** in computer vision [KvD83]. Ray tracing solves the integral equation using point sampling and Monte Carlo integration, while radiosity methods solve the problem as a system of linear equations.

Parallel to this computer graphics research, many of the same problems have been studied in the thermal radiation literature, although with different emphasis. Thermal radiation problems typically are less concerned with appearances and details than with accurate measurements of light level or temperature, for example, so they typically do not admit the complex 3-D scenes encountered in graphics rendering algorithms. Instead, the early thermal radiation literature has focused on the simulation of simple scenes using analytic techniques [SH81]. More recently, application of the finite element method has facilitated the extension to more complex problems involving conduction and convection in addition to radiation.

[*] Dept. of Electrical Engineering and Computer Sciences U.C. Berkeley, Berkeley, CA 94720, ph@miro.berkeley.edu

[†] 2029 N. Shoreline Blvd., Mountain View, CA 94043, jmw@sgi.com

[‡] Minor revisions: July '91.

Such problems are governed by integro-differential equations [Chu88]. Some of these techniques have recently been applied to computer graphics problems for the simulation of absorption and scattering in participating media [Rus87].

Previous research on image synthesis has typically pursued one of the two goals: speed or visual realism. The speed of radiosity algorithms has been improved dramatically by the introduction of progressive radiosity techniques [CCWG88]. Visual realism, a subjective measure of image quality, has been pursued for applications in special effects for film and television. There has been relatively little emphasis among computer graphicists of objective, numerical accuracy (with a few exceptions: [WRC88,BRW89,HS90]). The relevance of accuracy in image synthesis will grow as computer graphics techniques increasingly come to be used for interdisciplinary scientific and engineering simulations.

Our goal in this paper is to develop more accurate algorithms for the simulation of global illumination. To achieve an accurate result, we must combat all significant sources of error, including those arising in the stages of problem statement, discretization or sampling, and solution.

Global illumination problems have a different character from many integral equations studied in engineering, because of the importance of occlusions. Occlusions in a scene cause discontinuities such as shadow edges in the resulting intensity function.

We borrow three main techniques from the finite element literature: the use of higher order elements to more accurately represent solution functions, matrix formulation techniques for robustly transforming an integral equation into a system of equations, and a priori meshing techniques to choose a finite element (subdivision) mesh sensitive to discontinuities and singularities in the problem.

Most of the techniques described in the paper are quite general and can be applied to global illumination problems involving diffuse and specular surfaces, participating media, and wavelength and time-dependence. Equations are given in their most general form, where possible, but to help intuition, some of the discussion is limited to the simulation of radiosity in a two-dimensional *flatland* world.

The remainder of the paper consists of the following parts. First, the integral equation governing global illumination is reviewed, and its properties are discussed. Then general approximation techniques and finite element techniques are reviewed and applied to the global illumination problem. Techniques for solving the equations and for intelligent meshing are then discussed. Finally, results and conclusions are given.

2 Physical Foundation

The primary source for the underlying equations used in radiosity work to date is the thermal engineering literature based on classical electromagnetic theory[SH81]. Unfortunately, in practice, the engineering accuracy requirements have been quite different from those needed for high quality renderings. In particular, discrete bulk transfer quantities under many simplifying assumptions are sufficient for many engineering calculations, while continuous quantities are desirable for computer graphics. To achieve this latter accuracy goal it is necessary to start from a more general physical model and carefully choose the allowable simplifications and approximations.

Consider the problem of global illumination by participating media in a closed domain Ω with boundary Γ in $\Re^3$. To reduce complexity, geometric optics is assumed ($\lambda \approx 0$), thus ignoring the effects of interference and diffraction. In general, all of the terms considered may have spectral, spatial, angular, temperature and time dependences. For many graphics needs, it is often possible to ignore many of these physical dependences.

2.1 Conservation of Energy

Conservation of energy at a boundary in the domain provides the definition for the outgoing intensity of light I_{out}, of wavelength λ, at position $\mathbf{x}$, in outgoing direction Θ_{out}, at time t, due to the emission, reflection and transmission:

$$I_{\mathrm{out}}(\lambda, \mathbf{x}, \Theta_{\mathrm{out}}, t) = \epsilon(\lambda, \mathbf{x}, \Theta_{\mathrm{out}}, t) \quad (1)$$
$$+ \int_{\omega} \sigma_{\mathrm{bd}}(\lambda, \mathbf{x}, \Theta_{\mathrm{in}}, \Theta_{\mathrm{out}}, t) I_{\mathrm{in}}(\lambda, \mathbf{x}, \Theta_{\mathrm{in}}, t) \cos(\theta_{\mathrm{in}})\, d\omega_{\mathrm{in}}$$

where the surface quantities at $\mathbf{x}$ are:

SYMBOL	MEANING
ϵ	emissivity
ω	solid angle domain for all incoming radiation
σ_{bd}	bidirectional scattering function
I_{in}	incoming light intensity
Θ_{in}	incoming direction
θ_{in}	angle between the incoming direction and surface normal

The scattering function σ_{bd} combines the bidirectional reflectivity, ρ_{bd}, and the bidirectional transmissivity, τ_{bd}. See [SH81,Kaj86,Rus87,Rus89] for a complete derivation.

In a closed domain, Eq (1) may be simplified by a change of variable: the integral of incident directions, ω, becomes an integral over all surfaces Γ, with the introduction of the visibility factor V, in the integrand:

$$
\begin{aligned}
I_{\text{out}}(\lambda, \mathbf{x}, \boldsymbol{\Theta}_{\text{out}}, t) &= \epsilon(\lambda, \mathbf{x}, \boldsymbol{\Theta}_{\text{out}}, t) \\
&+ \int_{\Gamma} \sigma_{\text{bd}}(\lambda, \mathbf{x}, \boldsymbol{\Theta}_{\text{in}}, \boldsymbol{\Theta}_{\text{out}}, t) I_{\text{out}}(\lambda, \bar{\mathbf{x}}, \bar{\boldsymbol{\Theta}}_{\text{out}}, t) \\
&\quad \cdot V(\mathbf{x}, \bar{\mathbf{x}}) \frac{\cos(\theta_{\text{in}}) \cos(\bar{\theta}_{\text{out}})}{\|\mathbf{x} - \bar{\mathbf{x}}\|^2} \, d\bar{\mathbf{x}} \qquad (2)
\end{aligned}
$$

where $\bar{\boldsymbol{\Theta}}_{\text{out}}$ and $\bar{\theta}_{\text{out}}$ are evaluated in the context of the surface at $\bar{\mathbf{x}}$. In the two-dimensional case, the change of variable results in a denominator of $2\,\|\mathbf{x} - \bar{\mathbf{x}}\|$ instead of $\|\mathbf{x} - \bar{\mathbf{x}}\|^2$.

In general, the visibility factor accounts for intensity gain or attenuation by any intervening participating media and requires a path integral. This results in an integro-differential equation system for the intensity in the domain [SH81,Rus87,Chu88]. When no participating media is present, V may be simply defined as

$$
V(\mathbf{x}, \bar{\mathbf{x}}) = \begin{cases} 1 & \text{if } \bar{\mathbf{x}} \text{ is visible from } \mathbf{x} \\ 0 & \text{otherwise} \end{cases}
$$

and the energy conservation Eq (2) reduces to integral equation form.

2.2 The Integral Equation

Eq (2) is a *Fredholm integral equation of the second kind* in $\mathbf{x}$ [KvD83,Kaj86,DM85] and may by written:

$$
u(\mathbf{x}) = e(\mathbf{x}) + \int_{\Gamma} \kappa(\mathbf{x}, \bar{\mathbf{x}}) u(\bar{\mathbf{x}}) \, d\bar{\mathbf{x}} \qquad (3)
$$

where the integral kernel, κ, is given by

$$
\begin{aligned}
\kappa(\mathbf{x}, \bar{\mathbf{x}}) &= \sigma_{\text{bd}}(\lambda, \mathbf{x}, \boldsymbol{\Theta}_{\text{in}}, \boldsymbol{\Theta}_{\text{out}}, t) \\
&\quad \cdot V(\mathbf{x}, \bar{\mathbf{x}}) \frac{\cos(\theta_{\text{in}}) \cos(\bar{\theta}_{\text{out}})}{\|\mathbf{x} - \bar{\mathbf{x}}\|^2}
\end{aligned}
$$

and u and e correspond to I_{out} and ϵ respectively.

For the case of perfect diffuse scattering (σ_{bd} independent of $\boldsymbol{\Theta}$) the correspondence traditionally includes a factor of π to account for the constant angular dependence and u is termed ***radiosity***: $u = \pi I_{\text{out}}$. Eq (3) is often abbreviated as

$$
u = e + \mathcal{K} u \qquad (4)
$$

where $\mathcal{K}u$ denotes the integral operator

$$
(\mathcal{K}u)(\mathbf{x}) = \int_{\Gamma} \kappa(\mathbf{x}, \bar{\mathbf{x}}) u(\bar{\mathbf{x}}) \, d\bar{\mathbf{x}}
$$

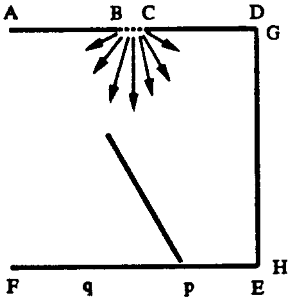

Figure 1: Flatland test scene. All edges are reflective except dashed edge BC at top, which is a light source, and angled edge, which is black. The angled obstacle causes a sharp shadow edge at p and a gradual penumbra at q.

3 Understanding the Integral Equation in Flatland

In a three-dimensional world, it is difficult to visualize the global illumination equation and to test algorithms for its solution because of the high dimensionality of the functions involved. In full generality, intensity is a function of three-dimensional position $\mathbf{x}$, two-dimensional direction $\boldsymbol{\Theta}_{\text{out}}$, wavelength, and time, for a total of seven variables. The kernel κ for such a problem would have even more dimensions. Clearly it is difficult to understand such complex functions.

To simplify the problem, we temporarily restrict our attention to ***radiosity in flatland***: a two-dimensional world consisting of opaque, diffuse objects [Abb84]. For now, we will restrict ourselves further to a static scene with closed polygonal shapes, diffuse light sources, no wavelength-dependence (i.e. grayscale), and no participating media.

In this flatland world the global illumination problem reduces to the determination of the radiosity (a scalar) at each point on the edges of the polygons. A flatland scene is shown in Fig 1. Instead of shading two-dimensional surfaces and computing two-dimensional integrals, as we do in 3-D graphics, in flatland graphics we shade one-dimensional edges and compute one-dimensional integrals. Relative to three-dimensional worlds, in flatland one finds that analytic results are easier to come by, algorithms are easier to debug, brute force techniques such as Monte Carlo integration converge faster, and it is possible to compute approximate solutions so accurate that they can be regarded as exact. This facilitates the use of

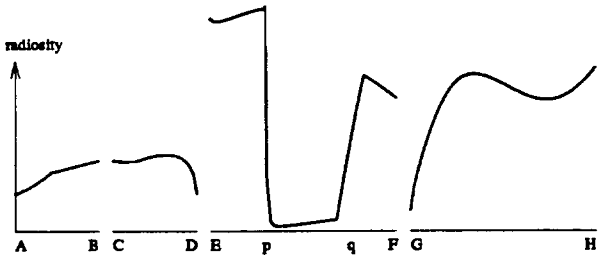

Figure 2: Radiosity as a function of arc length along the non-black edges of test scene. Note the sharp shadow edge at p and the gradual one at q.

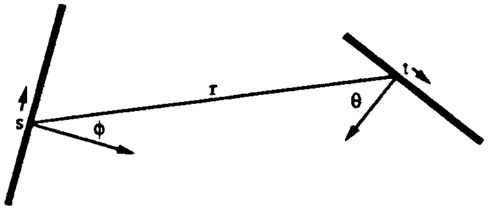

Figure 3: Visibility geometry for edge points with parameter values s and t.

quantitative error metrics for the objective comparison of algorithms.

3.1 Integral Equation for Radiosity in Flatland

Suppose the scene consists of m edges, and the length of edge i is L_i. Each edge i is parameterized by a variable s_i, which runs from 0 to L_i, and the radiosity along each edge is given by $u_i(s_i)$. For convenience, we abut the domains of these functions in arbitrary order to create a single function $u(s)$ parameterized by s, which runs from 0 to $L = \sum_i L_i$. Note that this concatenation introduces explicit discontinuities at edge endpoints. The radiosity function can now be plotted as a piecewise-continuous function as shown in Fig 2.

For simplicity, we assume that each edge has constant reflectivity ρ and constant emittance e. Edges are reflectors if $\rho > 0$, and light sources if $e > 0$, and occasionally both. Reflectivity is a unitless quantity between 0 (black) and 1 (perfect white). Generally, because it is difficult to keep surfaces clean, the maximum practical value for ρ is .85 .

In flatland, radiosity is determined by the Fredholm integral equation

$$u(s) = e(s)+\rho(s)\int_0^L dt\, u(t)\frac{cos\phi(s,t)\, cos\theta(s,t)V(s,t)}{2r(s,t)} \tag{5}$$

which is a special case of Eq (2). This integral equation can be written in the general form of second-kind equations (Eq (4)) by defining the kernel:

$$\kappa(s,t) = \rho(s)\frac{cos\phi(s,t)\, cos\theta(s,t)\, V(s,t)}{2r(s,t)}$$

Since we have abutted the domains of the edges in the scene, the kernel's domain consists of rectangular blocks corresponding to pairs (i,j) of edges. The kernel is discontinuous at the boundaries of the blocks, and also along occlusion curves that trace out hyperbolas in st space, (see Fig 15). Note also that the kernel is singular at reflex corners in the scene (where touching surfaces face each other), because $\kappa \to \infty$ as $r \to 0$.

3.2 Properties of the Solution Functions

We can derive many of the qualitative properties of the exact solution function $u(s)$ from the properties of the kernel and the geometry of the scene, even without solution algorithms. Eq (5) has a unique solution if the integral of the kernel is bounded [DM85].

We call discontinuities in the kth derivative of a function D^k discontinuities. A function has a D^k discontinuity at a point if it is C^{k-1} there but not C^k. Shadows due to a point light source can cause D^0 discontinuities in the value of the radiosity. Area light sources cast hard shadows with D^0 discontinuities when objects touch, and soft shadows with D^1 discontinuities when objects do not touch (Fig 4). The pattern generalizes to higher order discontinuities. If there is a D^k discontinuity at a point on one edge, then it can cause D^k discontinuities at all touching points visible to it, and D^{k+1} discontinuities at the projection of all of the silhouette points from its point of view. A D^k discontinuity in the normal of a curve can cause a D^k discontinuity in the radiosity at that point. If there are no occlusions in a scene then there are no shadows and the only discontinuities come from the edge endpoints.

3.3 Neumann Series Approximation

There is no cookbook solution method for integral equations; most cannot be solved analytically. Exact solutions to equation Eq (5) are known only in the simplest geometries. Even the case of two unit,

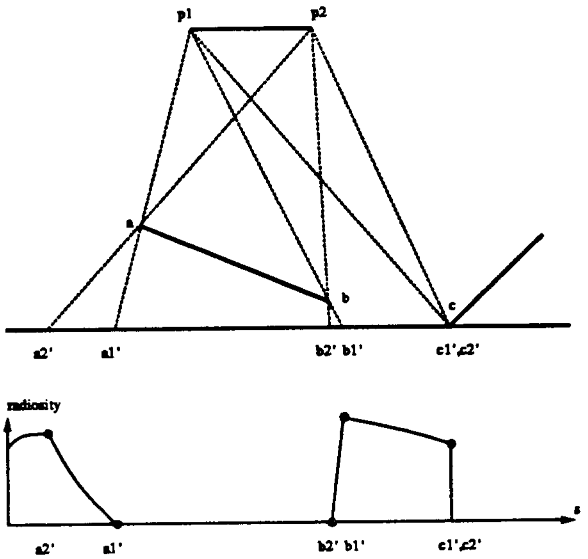

Figure 4: D^1 and D^0 discontinuities caused by area light source at top illuminating an occluded edge at bottom. D^1 discontinuities delimit the penumbras at a2', a1', b2', and b1'. D^0 discontinuity at the hard shadow edge c1'.

reflective edges forming a right-angle corner, illuminated from infinity, has no known analytic solution [Hor77].

Approximate solutions to many integral equations can be found iteratively. Starting with some initial guess $u^{(0)}(s)$, subsequent approximations are defined by

$$u^{(i)} = e + \mathcal{K}u^{(i-1)}$$

If we start with $u^{(0)} = e$, then the ith approximant is the truncated series

$$u^{(i)} = e + \mathcal{K}e + \mathcal{K}^2 e + \cdots + \mathcal{K}^i e$$

where $\mathcal{K}^i$ denotes i successive applications of the integral operator $\mathcal{K}$. If the kernel κ has largest eigenvalue with magnitude less than 1, then the sequence $u^{(i)}$ converges [DM85], and the exact solution is given by the *Neumann series*

$$u = u^{(\infty)} = \sum_{i=0}^{\infty} \mathcal{K}^i e \tag{6}$$

For global illumination, the ith term $(\mathcal{K}^i e)(s)$ is the light that reaches the point s after exactly i 'hops' [Kaj86], where a hop is an unoccluded straight-line path between surfaces. The approximant $u^{(i)}(s)$ is the light that reaches point s in i hops or fewer.

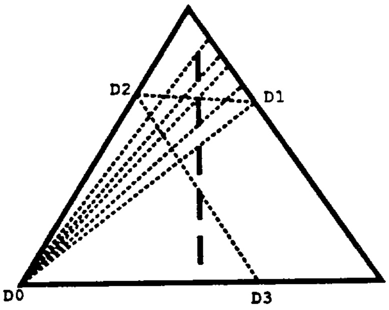

Figure 5: Propagation of discontinuities. Solid lines show edges in scene; dotted lines show rays of light leading to discontinuities.

Early illumination models (what Kajiya called the 'Utah approximation') simulated only direct illumination $u^{(1)} = e + \mathcal{K}e$; global illumination attempts to compute $u^{(\infty)}$. Unfortunately, it appears impossible to perform the multiple integrals $\mathcal{K}^i e$ analytically even for radiosity in flatland.

3.4 Propagation of Discontinuities

Using the Neumann series, however, we can discover more qualitative properties of the solution function. Earlier we noted that shadows from point and area light sources cause D^0 and D^1 discontinuities, respectively. Using the Neumann series we can show how higher order discontinuities arise after additional hops of light.

Theorem: There can be an infinite number of discontinuities of various orders in the radiosity function.

Proof: This is proven with an example. Consider the scene in Fig 5 that consists of three reflective and emissive edges in a triangle and $m - 3$ black edges on a line in their interior. Let q_i denote the number of D^i discontinuities in $u^{(i)}$. If each triangle edge has a different emittance, then $q_0 = 3$. After one hop, each corner creates one D^1 shadow edge from each end of the interior edges, so $q_1 = 3 \times 2(m - 3)$. After subsequent hops, each D^i discontinuity creates $2(m - 3)$ D^{i+1} shadow edges, but one of these is coincident with its 'grandparent' D^{i-1} discontinuity, and in general the remaining $2m - 7$ shadow edges will not be coincident with lower order discontinuities, so $q_i = 6(m - 3)(2m - 7)^{i-1}$ for $i \geq 1$. The exact solution $u^{(\infty)}$ will have all of these discontinuities. The number of discontinuities of all orders is thus infinite.

The preceding scene achieves the asymptotic upper bound on the number of discontinuities of each order. In any scene, any endpoint can cause at most $2m$ shadow edges, so $q_i \leq 2mq_{i-1}$, and $q_i = O(m^i)$.

The possibility of so many discontinuities suggests that no analytic solution to the integral equations for global illumination is possible in general. We therefore turn to numerical approximations.

4 Approximation Techniques

In the ensuing discussion of approximation techniques the assumptions made by existing radiosity algorithms will be identified and more advanced techniques introduced.

The underlying functions in the solution approximation subspaces are frequently referred to as *basis functions*. These functions may provide global support (non-zero anywhere) as typified by the Rayleigh-Ritz or spectral techniques [Fle84]. Alternately, techniques based on functions with only local support (non-zero only in a small portion of the domain) such as spline based or finite element methods are possible. These latter techniques have gained increasing popularity for their computational robustness and ability to easily model complex geometry.

Consider an approximate solution function $\tilde{u}$ defined by a linear combination of a finite number, n_{eq}, of linearly independent basis functions N_i:

$$\tilde{u} = \tilde{u}_1 N_1 + \tilde{u}_2 N_2 + \cdots = \sum_{i=1}^{n_{eq}} \tilde{u}_i N_i$$

where the $\tilde{u}_i$ are unknown generalized coefficients. The interpolation basis functions are not limited to spatial dependence but in fact may be functions of $\mathbf{x}$, λ, Θ, t, etc. For simplicity in the following presentation only spatial dependence will be considered (*shapes* $= N_i(\mathbf{x})$) and the domain will be restricted to Ω. For global illumination in the absence of participating media the domain may be further restricted to Γ. Furthermore, the presentation is limited to linear Fredholm integral equations of the second kind. The generalization to nonlinear equations and their associated solution using iterative techniques such as Newton-Raphson iteration is quite straightforward.

In general, no combination of $\tilde{u}_i$ values will exactly satisfy the governing equation Eq (2), since the space of all such $\tilde{u}$ is a proper subspace of the space of all piecewise-continuous functions. The residual error of the approximate solution $\tilde{u}$ is defined as:

$$r(\mathbf{x}) = e(\mathbf{x}) + \int_\Omega \kappa(\mathbf{x}, \bar{\mathbf{x}})\tilde{u}(\bar{\mathbf{x}})\, d\bar{\mathbf{x}} - \tilde{u}(\mathbf{x})$$

The exact solution has a residual that is identically zero. A "good" approximation is one for which r is small everywhere in Ω.

The following subsections define different approaches to determining the unknown coefficients to minimize the resultant residual error. The method of weighted residuals is the general approximation technique from which point collocation and Galerkin methods may be derived.

4.1 Method of Weighted Residuals

The method of weighted residuals forces the residual error due to the approximate solution to be as small as possible with respect to a specified set of weighting functions [Fin72]. Requiring the residual to be orthogonal to a given set of weighting functions $w_i(\mathbf{x})$ over Ω results in

$$\begin{aligned} 0 &= \int_\Omega r(\mathbf{x}) w_i(\mathbf{x})\, dx \\ &= \int_\Omega \left[e(\mathbf{x}) + \int_\Omega \kappa(\mathbf{x}, \bar{\mathbf{x}})\tilde{u}(\bar{\mathbf{x}})\, d\bar{\mathbf{x}} - \tilde{u}(\mathbf{x}) \right] w_i(\mathbf{x})\, dx \end{aligned}$$

After performing the double integration over the domain, the solution of the resulting algebraic equations in the generalized coefficients provides the approximate solution. For the linear finite dimensional case the problem may be expressed in equivalent matrix form as

$$\mathbf{A}\tilde{\mathbf{u}} = \tilde{\mathbf{e}} \tag{7}$$

where $\tilde{\mathbf{u}}$ consists of the unknown coefficients $\{\tilde{u}_i\}$ and the generalized stiffness matrix is composed of two terms

$$\mathbf{A} = \mathbf{M} - \mathbf{K} \tag{8}$$

the first part, $\mathbf{M}$, given by a single integral

$$M_{ij} = \int_\Omega w_i(\mathbf{x}) N_j(\mathbf{x})\, d\mathbf{x}$$

and the second, $\mathbf{K}$, requiring double integration

$$K_{ij} = \int_\Omega w_i(\mathbf{x}) \int_\Omega \kappa(\mathbf{x}, \bar{\mathbf{x}}) N_j(\bar{\mathbf{x}})\, d\bar{\mathbf{x}}\, d\mathbf{x}$$

Finally, the right-hand-side or forcing vector, $\tilde{\mathbf{e}}$, is given by

$$\tilde{e}_i = \int_\Omega e(\mathbf{x}) w_i(\mathbf{x})\, d\mathbf{x} \tag{9}$$

4.2 Point Collocation Method

In point collocation techniques the residual is forced to zero at a given set of points $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{n_{eq}}$ equal in number to the number of unknowns in the approximate solution [BD89]. This is equivalent to choosing weighting functions composed of Dirac deltas located at the collocation points, $w_i(\mathbf{x}) = \delta(\mathbf{x} - \mathbf{x}_i)$, resulting in n_{eq} equations $r(\mathbf{x}_i) = 0$ in n_{eq} unknowns:

$$\tilde{u}(\mathbf{x}_i) = e(\mathbf{x}_i) + \int_\Omega \kappa(\mathbf{x}_i, \bar{\mathbf{x}})\tilde{u}(\bar{\mathbf{x}})\, d\bar{\mathbf{x}} \qquad (10)$$

Note that one level of integration over the domain has been eliminated, substantially reducing the computational effort.

In principal, any arbitrary distribution of points may be used. Typically, the points are distributed uniformly with respect to basis function support. This improves the conditioning of the resultant matrix system. For radiosity problems, some care must be taken in the choice of collocation points as points near reflex corners may lead to evaluations of singular kernels.

Traditional radiosity algorithms in computer graphics [GTGB84,CG85,CCWG88] employ numerical approximations that correspond to elementary point collocation techniques with piecewise-constant basis functions. With these restrictions, the $\tilde{u}_i$ may be interpreted as polygon radiosities and $\mathbf{M}$ is the identity: $\mathbf{M} = \mathbf{I}$. Typically the matrix $\mathbf{K}$ is computed by rendering the scene from the point of view of the polygon centers, which are used as the collocation points. Rendering can be done using either a hemicube [CG85] or by ray tracing [Kaj86,WEH89,BF89]. Both are a form of numerical integration of the kernel in Eq (10). In certain geometries, it is possible to compute the integrals analytically [BRW89].

The vector $\tilde{\mathbf{e}}$ is a point sampling of $e(\mathbf{x}_i)$, corresponding to the constant emittance of each polygon.

After the system of equations is solved, the results are often displayed using Gouraud shading, yielding an image that is piecewise-linear in intensity. This is really just a display hack that helps conceal the errors of a piecewise-constant radiosity solution. The results may be free of artifacts and subjectively acceptable, but they will be much less accurate objectively than those of a true piecewise-linear formulation [Hec91].

4.3 Galerkin Method

In the Galerkin method the weighting functions are chosen equal to the basis functions for the solution [Fle84].

$$w_i = N_i \qquad (11)$$

The result is an approximate solution whose residual error is orthogonal to the space of solutions and thus to the solution itself. A desirable byproduct of this choice is that it generally minimizes the error in the natural energy norm. Additionally, for symmetric kernels the resultant equation system is also symmetric $\mathbf{A}_{ij} = \mathbf{A}_{ji}$. This both reduces the computational workload and considerably improves the robustness of associated numerical algorithms [Chu88]. Galerkin methods provide consistently accurate, robust solutions to a wide variety of engineering problems and will be used as the foundation for the finite element solution technique developed in the next section.

5 Finite Element Techniques

The preceding approximation techniques placed little restriction on the basis functions N_i. In this section the effect of constraining the basis functions to the finite element form is considered. The finite element method provides a comprehensive and systematic technique for the construction of piecewise interpolation functions over arbitrary domains [BCO81, Hug87]. When simulating global illumination with participating media, one uses volumetric finite elements, but when simulating non-participating media, as in our examples here, one uses two-dimensional elements embedded in a three-dimensional space, and the matrix formulation and computational procedure most closely resemble the *boundary element method* [Fle84,BD89]. In either case, the same basic steps apply.

In particular, the solution domain, Ω, is discretized into a *finite element mesh*, a collection of n_{el} subdomains or *elements*, Ω^e, and their associated *nodes*. As in the previous section, Γ may be substituted for Ω if no participating media are present. Nodes are special points within or on the boundary of elements that facilitate mesh construction and basis function definition. The piecewise basis functions are then limited to local support, i.e. they are only non-zero in the elements connected with their associated nodes. In finite elements these are often referred to as *shape* functions. Within each element, simple polynomial forms are typically used.

Unfortunately, a great deal of notation is required to precisely describe the relationships between global and element quantities and their associated calculations [Hug87]. The enormous payoff, however, in generality and ease of implementation is well worthwhile, and the bookkeeping required is quite straightforward.

For example, one-dimensional *line elements* ($n_{\mathrm{ed}} = 1$) are defined in terms of a sequence of element endpoints (nodes), x_i, where $0 = x_0 \le x_1 \le \ldots \le x_n = L$.

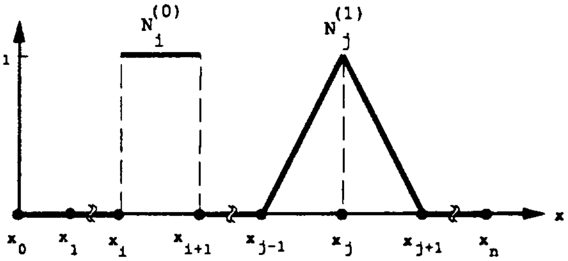

Figure 6: Piecewise-constant (box) and piecewise-linear (hat) functions.

These points are analogous to knot vectors for splines [BBB87]. The simplest basis functions, and those currently used by the majority of radiosity solution algorithms, are the piecewise constant (box) functions:

$$N_i^{(0)}(x) = \begin{cases} 1 & \text{if } x_i < x < x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

The next order higher, and more familiar to the average finite element user, are the piecewise linear (hat) functions given by:

$$N_j^{(1)}(x) = \begin{cases} \frac{x-x_{j-1}}{x_j-x_{j-1}} & \text{if } x_{j-1} \leq x \leq x_j \\ \frac{x_{j+1}-x}{x_{j+1}-x_j} & \text{if } x_j \leq x \leq x_{j+1} \\ 0 & \text{otherwise} \end{cases}$$

Examples of these function classes are shown in Fig 6, generalization to higher polynomial powers is straightforward.

The property of local support substantially reduces the computational effort required to evaluate the integrals in the preceding section. The integrals over the continuous domain Ω become sums of integrals over the set of elements Ω^e, which in turn only require evaluation for their associated basis functions since all other basis functions will be zero within that Ω^e.

To guarantee sufficient conditions for convergence as the mesh of elements is refined the following three requirements on the basis functions are imposed: smoothness on the interior of each element Ω^e; continuity across the boundary of each element Γ^e; and spanning of linear functions. Less necessary but still desirable attributes include well graded meshes (no abrupt changes in adjacent element areas) composed of elements with good aspect ratios (no slivers). In practice these conditions impose significant restrictions on the form of the finite element mesh. In fact, the vast majority of polygonal meshes constructed for radiosity solutions to date are inappropriate for finite element based solution techniques. In particular, due to poor mesh construction, attempts to use higher-order interpolation will violate the second continuity requirement wherever T-vertices occur [BMSW91].

By default, most meshes are constructed by maximally connecting the *degrees-of-freedom* of adjacent elements at shared nodes. This introduces an element-dependent degree of continuity in the solution. Optionally, selected adjacent elements may be connected with lower degree continuity in a manner similar to the use of double knots in splines. This proves to be a crucial capability for supporting the necessary D^k meshing discussed later. In radiosity solutions, the equations remain well posed even without any element connectivity. However, the effort required to obtain the solution may increase.

5.1 Isoparametric Elements

Isoparametric elements satisfy the above requirements and additionally provide a valuable degree of flexibility in modeling and programming convenience. Isoparametric elements use the same parametric basis functions for both the spatial and radiosity solution interpolation. A simple n_{ed}-dimensional domain, $\Box$, provides the master parametric domain for element-level function mapping and evaluation. In two dimensions for example, $n_{ed} = 2$, $\Box$ is chosen to be the bi-unit square, $\xi \in [-1,1] \times [-1,1]$. The spatial coordinates, $\mathbf{x} \in \Re^{n_{sd}}$, within an element, e, are interpolated by

$$\mathbf{x}(\xi) = \sum_{a=1}^{n_{en}} N_a(\xi)\mathbf{x}_a^e \qquad (12)$$

where n_{en} is the number of element nodes, N_a are the *local* element basis functions and $\mathbf{x}_a^e$ are the local nodal coordinates. Isoparametric elements interpolate the corresponding local approximate solution by

$$\tilde{u}(\xi) = \sum_{a=1}^{n_{en}} N_a(\xi)\tilde{u}_a^e \qquad (13)$$

where $\tilde{u}_a^e$ are the local nodal solution values. An example of the mapping for the simplest two-dimensional isoparametric element, the bilinear quadrilateral, is shown in Fig 7. Note that for valid invertible mappings $n_{ed} \leq n_{sd}$ and the element must be convex.

Implicit in the above definition is the standard finite element local-to-global index relationship, L. More specifically, global index is related to the local element index by $i = L(a,e)$. In theory, the L relationship may be expressed as a large sparse Boolean matrix. In practice, all element calculations may be performed in the small (a,e)-space and the results

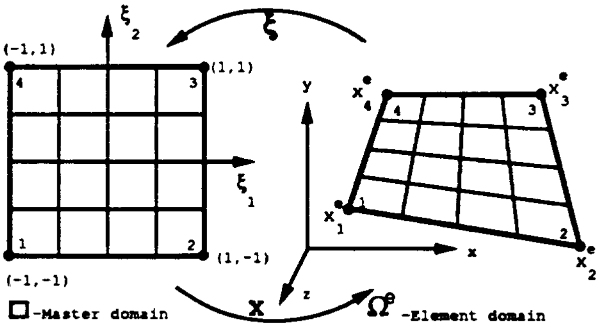

Figure 7: Isoparametric map for bilinear quadrilateral element.

assembled into the global space through indirection. This is one of the fundamental advantages of the finite element technique.

An advantage of the use of isoparametric interpolation becomes immediately apparent in the evaluation of element-level integration. A simple change of variables allows the integration to be performed over the fixed master domain:

$$\int_{\Omega^e} f(\mathbf{x})\,d\mathbf{x} = \int_{\Box} f(\mathbf{x}(\xi))J(\xi)\,d\xi$$

where $f(\mathbf{x})$ is an arbitrary function and $J(\xi)$ is the Jacobian determinant of the spatial mapping $J(\xi) = \det(\partial\mathbf{x}/\partial\xi)$. Other forms of parametric mappings in which the order of the spatial interpolation is lower (subparametric) or higher (superparametric) than that used for the solution interpolation are also possible. Additionally, the use of higher order spatial interpolation provides a natural mechanism for the geometric approximation of curved surfaces.

Finally, it should be noted that the basis functions are not limited to simple polynomial form (Lagrange family) but may be constructed in a variety of ways (Serendipity, rational, or Hermite families). However, the degree of polynomial completeness directly contributes to the overall accuracy of the element.

5.2 Numerical Integration Methods

For simple interpolation functions and constant Jacobian mappings it is possible to compute point to area integrals analytically [BRW89]. As the scene becomes more complex, and the basis functions higher order, analytic evaluation becomes infeasible. A variety of numerical integration techniques exist for evaluation of multi-dimensional integrals. The most common form for finite element domain integration involve the cartesian product of one-dimensional numerical Gaussian integration rules. Integration of an arbitrary function, f, on an isoparametric element in a one dimensional domain is optimally approximated by

$$\int_{\Omega^e} f(\mathbf{x})\,d\mathbf{x} = \int_{\Box} f(\mathbf{x}(\xi))J\,d\xi \approx \sum_{p=1}^{n_{\mathrm{pts}}} \bar{w}_p f(\mathbf{x}(\bar{\xi}_p^e))J(\bar{\xi}_p^e) \tag{14}$$

where n_{pts} is the number of Gaussian integration points and $\bar{w}_p$ and $\bar{\xi}_p^e$ are the corresponding weights and locations. The n_{pts} rule provides exact results for polynomials up to degree $2n_{\mathrm{pts}}+1$. For the multi-dimensional case, $n_{\mathrm{ed}} > 1$, one-dimensional rules are applied repeatedly for each dimension. The multiple indices, products of weights, and summations can be remapped into a single index $1,\ldots,n_{\mathrm{pts}}^{\mathrm{ed}} = (n_{\mathrm{pts}})^{n_{\mathrm{ed}}}$ for simplicity.

Recent boundary element research has investigated adaptive subdivision strategies for accurate evaluation of singular integrands. Experience indicates that the lowest order Gauss rules generating non-singular matrices are sufficient for the majority of the element to element calculations. Analytic or adaptive integration is needed for the special edge and corner cases or where elements are in close proximity [BRW89]. Fully adaptive methods based on *a posteriori* integration error estimates typically require many additional integrand evaluations even for the most common case in which adaptation is terminated at the lowest level of refinement. Hybrid adaptive strategies in which integration order is predicted *a priori* based on element proximity indicators provide reasonable cost alternatives.

5.3 Element Matrix Formulation

Combining Galerkin approximation methods with isoparametric finite element basis functions and numerical integration, Eqs (7)–(9), (11), (12), (13), and (14), allow us to approximate radiosity problems using basis functions of arbitrary order and Galerkin techniques. The approximation yields a system of linear equations: $\mathbf{A}\bar{\mathbf{u}} = \bar{\mathbf{e}}$ where $\mathbf{A} = \mathbf{M} - \mathbf{K}$. $\mathbf{M}$ may be computed using an assembly of small element level matrices, $\mathbf{M}^e$,

$$\mathbf{M} = \mathop{\mathcal{A}}_{e=1}^{n_{\mathrm{el}}} \mathbf{M}^e$$

where the assembly operator $\mathcal{A}$ combines summation and local to global index transformation using L.

Each element level contribution may be computed independently, thus $\mathbf{M}$ may be assembled one element at a time using $\mathbf{M}^e$ approximated by:

$$M^e_{ab} \approx \sum_{p=1}^{n^{\rm ed}_{\rm pts}} \bar{w}_p N_a(\bar{\xi}^e_p) N_b(\bar{\xi}^e_p)$$

where a and b are limited to $1, \ldots, n_{\rm en}$.

Similarly, the element level $\mathbf{K}^{ef}$ relating two elements e and f at a time is approximated as

$$K^{ef}_{ab} \approx \sum_{p=1}^{n^{\rm ed}_{\rm pts}} \bar{w}_p N_a(\bar{\xi}^e_p) \sum_{q=1}^{n^{\rm ed}_{\rm pts}} \bar{w}_q \kappa(\bar{\xi}^e_p, \bar{\xi}^f_q) N_b(\bar{\xi}^f_q)$$

and can be assembled into global form using

$$\mathbf{K} = \mathop{\mathcal{A}}_{e=1}^{n_{\rm el}} \mathop{\mathcal{A}}_{f=1}^{n_{\rm el}} \mathbf{K}^{ef}$$

Finally, $\tilde{\mathbf{e}}$ is formed by assembling the $\tilde{\mathbf{e}}^e$ approximations given by

$$\tilde{e}^e_a \approx \sum_{p=1}^{n^{\rm ed}_{\rm pts}} \bar{w}_p e(\bar{\xi}^e_p) N_a(\bar{\xi}^e_p)$$

Only one element at a time need be processed.

In practice, the small dense element level quantities are computed one at a time and assembled on the fly into the large sparse global equivalents. This elegantly structured "element view" of the problem tremendously simplifies the calculations and permits very fast, robust and general implementations. Care should be take to choose the lowest order rule providing sufficient accuracy since the worst case $\mathbf{K}$ evaluation cost is $O((n_{\rm el} n^{\rm ed}_{\rm pts})^2)$.

6 Solving Radiosity Systems

Many techniques are known for solving the resulting system of equations $\mathbf{A}\tilde{\mathbf{u}} = \tilde{\mathbf{e}}$ (see [GVL89]), but the fastest, most accurate methods must exploit the special properties of radiosity equations.

6.1 Radiosity Matrix Properties

The properties of the matrix $\mathbf{A} = \mathbf{M} - \mathbf{K}$ (Eq (8)) depend on the choices fofor mesh, basis, and formulation. When point collocation techniques and constant or linear elements are used, $\mathbf{M} = \mathbf{I}$, and the radiosity matrix has a particularly simple form: $\mathbf{A} = \mathbf{I} - \mathbf{K}$. The elements of $\mathbf{K}$ in this case are proportional to the *form factors* commonly referred to in the radiosity literature.

The radiosity matrix $\mathbf{A}$ for a flatland scene is shown in Fig 8. Since the matrix $\mathbf{K}$ is derived from the kernel κ, many of its properties are analogous to those of the kernel.

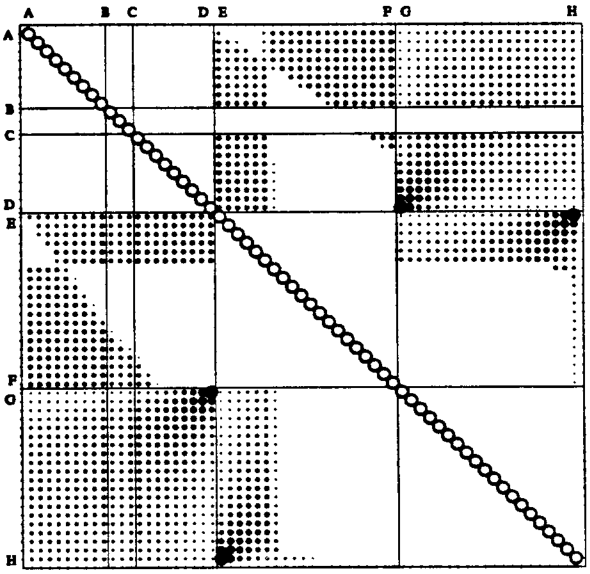

Figure 8: Nonzero elements of the sparse matrix $I-K$ for our test scene, with $n_{\rm eq} = 60$ equations. Diagonal elements (open circles) are all 1; off-diagonal elements are shown with dot area proportional to magnitude. White areas are zeros. Large dots indicate large form factors at reflex corners.

The matrices $\mathbf{A}$ encountered in radiosity problems are usually large, moderately sparse, diagonally dominant, non-symmetric positive definite, with largest eigenvalue less than 1. They are moderately sparse since, for most scenes of interest, each surface can "see" only a small fraction of the other surfaces. If a systematic element ordering scheme is used, then the radiosity matrix will have block character, where blocks containing nonzeros correspond to surfaces that are inter-visible. A block that is entirely nonzero comes from a pair of surfaces with no intervening occluders, while a block that is only partially nonzero comes from a pair of surfaces that are partially occluded and partially inter-visible. In flatland, the boundaries of the regions of zeros trace out hyperbolas in the matrix, just as in the kernel. Except for the block boundaries, and occlusion discontinuities, the kernel and hence the matrix values vary smoothly.

Any physically valid scene will have a diagonally dominant radiosity matrix $\mathbf{A}$. Physical validity constrains the range of reflectivity $(0 \leq \rho < 1)$, which implies that integrals of the kernel are bounded and that the largest eigenvalue is less than one. This condition implies that the Neumann series (Eq (6)) converges.

The above properties apply to radiosity matrices coming from a point collocation formulation. The matrices created by Galerkin and higher order finite element techniques are more difficult to characterize.

6.2 Linear System Solving

Systems of equations can be solved by either direct methods, indirect methods, or iterative methods [GVL89]. Direct methods such as Gaussian elimination do not exploit the sparseness of matrices as well as some other methods, so they are most suited to small or low-sparsity systems.

Indirect methods such as the conjugate gradient method generate a sequence of approximate solutions that are guaranteed to converge after n iterations, for $n \times n$ systems, and usually find accurate solutions far sooner. The conjugate gradient method is best suited to symmetric systems, but many radiosity problems can be *preconditioned* into symmetric form by the substitutions $\mathbf{K} = \mathbf{P}\mathbf{K}^*$ and $\tilde{\mathbf{u}} = \mathbf{P}\tilde{\mathbf{u}}^*$, where $\mathbf{P} = \text{diag}(\rho_1, \rho_2, \cdots, \rho_n)$. The matrix $\mathbf{K}^*$ is symmetric, so these substitutions transform the nonsymmetric system $(\mathbf{I} - \mathbf{K})\tilde{\mathbf{u}} = \tilde{\mathbf{e}}$ into the symmetric system $(\mathbf{P} - \mathbf{P}\mathbf{K}^*\mathbf{P})\tilde{\mathbf{u}}^* = \tilde{\mathbf{e}}$. This approach is an example of the *preconditioned conjugate gradient method*.

With iterative techniques, the third class of solution methods, convergence is guaranteed when the eigenvalues of the matrix satisfy certain conditions [GVL89]. For extremely sparse matrices iterative methods are often the fastest. The matrices that are encountered in radiosity problems have well-behaved eigenvalues, but they are not very sparse; not nearly as sparse as those for many multidimensional problems.

The simplest iterative algorithm is Jacobi's method, which, when applied to problems of the form $(\mathbf{I}-\mathbf{K})\tilde{\mathbf{u}} = \tilde{\mathbf{e}}$, computes the sequence of approximants

$$\tilde{\mathbf{u}}^{(i)} = \tilde{\mathbf{e}} + \mathbf{K}\tilde{\mathbf{u}}^{(i-1)}$$

for some initial guess $\tilde{\mathbf{u}}^{(0)}$. When applied to radiosity matrices of this form, Jacobi's method is a discrete approximation to the Neumann series. As in the Neumann series, the approximant $\tilde{\mathbf{u}}^{(i)}$ for a radiosity problem consists of the light reaching each point in i hops or fewer. The Gauss-Seidel iterative method is a simple variant of Jacobi's method, which, for a large class of matrices, converges twice as fast as Jacobi. A trivial extrapolating variation of Gauss-Seidel called *successive overrelaxation* accelerates convergence further. Even faster solution methods exist for specific problem domains [GVL89].

6.3 Radiosity-Specific Techniques

An advantage of many iterative techniques is that the only use they make of the matrix is the computation of matrix-vector products. Consequently, the matrix need not be stored explicitly, but can be computed on the fly, row-by-row, as the matrix product is computed.

The progressive radiosity approach is a radiosity-specific solution method of this type [CCWG88]. It computes selected columns of the radiosity matrix, using these to iterate toward convergence. Unfortunately, no rigorous theoretical analysis of progressive radiosity has been done to date.

Hanrahan's radiosity solution technique is another novel approach [HS90]. Instead of using a fixed mesh, shooting surfaces are subdivided adaptively relative to their distance to receiving surfaces. This method avoids the matrix formulation altogether, instead sampling the kernel adaptively in a quadtree-like pattern. For unoccluded environments of m polygons, the method has time cost $O(m)$. It has not yet been demonstrated for environments with occlusions, where the discontinuities in the kernel would complicate matters, but it holds promise as a fast, accurate solution technique.

7 Meshing

Early research in radiosity focused on the computation of form factors and efficient solution of the system of equations, but the issue of meshing or discretization of surfaces was little discussed; until recently it has remained a black art and a manual process for the most part [BMSW91].

7.1 Comparison of Meshing Techniques

Meshing techniques can be classified as either uniform or nonuniform (also known as adaptive). Simple radiosity systems typically employ uniform meshes; subdividing rectangular polygons into a uniform grid of rectangular elements, for example. Non-uniform meshing can be either *a priori*, in which the mesh is chosen before the solution is found, or *a posteriori*, in which a mesh is chosen based on a previous solution, a new solution is found, and the cycle is repeated as necessary.

A priori methods attempt to predict where additional subdivision is needed beyond a uniform mesh. Campbell's grid generation scheme predicts the location of shadow edges by approximating the light sources by one or more points and projecting all silhouette edges onto other surfaces in the scene [CF90]. This technique is a valuable step toward better meshing, but it will usually find lines down the center of a penumbra, rather than the boundaries of the penumbra, where the discontinuities occur. Subdividing near a discontinuity improves the potential accuracy, but does not improve it as much as subdivision directly on the discontinuity. The method we present shortly, ***discontinuity meshing***, attempts to position element boundaries to best resolve any such discontinuities in the solution.

A posteriori methods for global illumination have been examined more fully for ray tracing algorithms than for radiosity algorithms. In ray tracing algorithms one typically does not work with a mesh of elements chosen before solution, but with a set of samples that are accumulated during the course of the algorithm. Ward observed that the coherence of the diffuse component of surface radiance could be exploited by saving radiance samples at a scattering of points across each surface [WRC88]. Ward's algorithm samples most densely at corners and other regions where surfaces are in proximity. Radiosity information can also be regarded as a texture that is mapped onto the surface. Ray tracing algorithms can be used to adaptively subdivide and sample more densely near shadow edges [Hec90].

Another class of a posteriori methods are the *multigrid* methods. Multigrid methods, common in finite difference and finite element techniques, solve the system on a succession of scales, propagating low spatial frequency, slowly-varying components from coarse grids to fine grids, and propagating high-frequency, rapidly-varying components from fine grids back to coarse [PH85]. For many classes of partial differential equations, they provide the fastest known solution methods.

Cohen's substructuring method for radiosity is essentially a simple form of multigrid method [CGIB86]. The substructuring method first chooses a coarse mesh, solves for radiosities on the coarse mesh, then refines the mesh where the coarse solution suggests high gradients, and re-solves on the finer mesh using the coarse solution as a starting point. Substructuring typically solves the problem on only two grids, while multigrid methods solve the problem on a pyramid of grids.

7.2 Discontinuity Meshing

Radiosity problems are more complex than many finite element problems due to the discontinuities in the solution caused by occlusion.

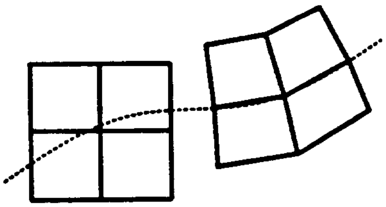

Figure 9: Left: Mesh does not resolve the discontinuity (dashed curve). Right: Mesh resolves discontinuity.

Discontinuity meshing is an approach to meshing that attempts to accurately resolve the most significant discontinuities in the solution by optimal positioning of element boundaries. In general, an approximation using polynomial elements will have discontinuities at element boundaries only. Different element families have different orders of boundary discontinuity. For a degree p element, the boundary discontinuities can range in order from D^0 to D^p. Lagrange elements are D^1 at their borders for all element orders, while Hermitian elements are $D^{(p+1)/2}$ at their borders for elements of degree p [BCO81].

When discontinuities in the true solution fall on element boundaries, the mesh is said to ***resolve*** the discontinuity (Fig 9). Elements cannot resolve discontinuities of orders above their degree. All discontinuities that can be resolved should be resolved. When using constant elements, discontinuity meshing should attempt to resolve all D^0 discontinuities. For linear elements, discontinuity meshing should attempt to resolve all D^0 discontinuities with ***double nodes*** [Hug87], and D^1 discontinuities should be resolved with a single node.

In general, to resolve a D^p discontinuity, elements of degree p or higher must be used, and a node must be placed at this point. The derivatives of order $p-1$ or less may be coupled across the element boundary at this point, but this is not necessary. It is not fruitful to resolve discontinuities of degree higher than the element degree because the errors caused by failure to resolve discontinuities are swamped by the discontinuities introduced at element boundaries.

The placement of mesh boundaries is especially important for problems such as radiosity, since its solution function contains discontinuities. Failure to resolve the discontinuities with the mesh will result in poorer solutions. For a given element size, discontinuity meshing will give a more accurate result than non-discontinuity meshing. For a given accuracy, discontinuity meshing does not require a mesh

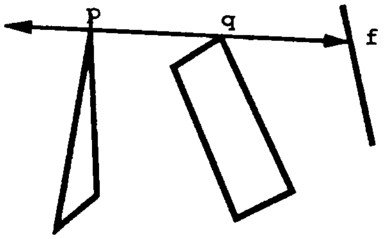

Figure 10: f is a critical point caused by the critical ray through endpoints p and q.

as fine as that of non-discontinuity meshing, and thus runs faster. The differences in accuracy and speed are both qualitative and quantitative in nature, as illustrated later in the results.

7.3 Flatland Discontinuity Meshing Algorithm

In flatland, discontinuity meshing can be done quite easily. As before, assume a scene consisting of m opaque line segments. We first locate D^0 discontinuities, then D^1 discontinuities, and so on up to the degree of the elements being used.

First, D^0 meshing is performed: meshing at all D^0 discontinuities. D^0 discontinuities come from intersecting edges or from the projection of silhouette points from point light sources. Intersecting edges can be found in $O(mlogm)$ time. To maintain the invariant that all D^0 discontinuities occur at edge endpoints, we split each intersecting edge into two edges at an intersection point. Point light sources are dismissed as non-physical. With this restriction, all remaining shadow edges will be soft penumbras (i.e. D^1), not hard (D^0).

Next, D^1 meshing is done (unless constant elements are used, in which case we can stop here). D^1 discontinuities all result from remote changes in visibility, where an edge endpoint becomes occluded. An edge point which is collinear with two edge endpoints visible from it is called a *critical point* and the line along which they lie is called a *critical line* (Fig 10). As proven earlier, there are $O(m^2)$ critical points in a scene. In practice, the number is often far smaller than this upper bound. A critical point will not cause a D^1 discontinuity if it is on a black edge ($\rho = 0$).

All critical points can be found as follows:

```
for each node p
    for each node q
        if no edge intersects line segment pq then {
            e = trace_ray(p, p-q)
            f = trace_ray(q, q-p)
            if e<>NULL then critical_point(e)
            if f<>NULL then critical_point(f)
        }
```

The routine **trace_ray(p, d)** traces a ray from point p in direction d and returns the first edge point hit, if any. The above algorithm, implemented straightforwardly, has $O(m^3)$ time cost. This can be reduced to $O(m^2 \log m)$ by performing a visibility determination for each point p using a radial sweepline algorithm.

Higher order discontinuities can be located in a similar fashion. To find all D^k discontinuities, one loops over all D^{k-1} discontinuities, tracing rays from that point through all edge endpoints which are silhouettes with respect to that point. The point on the first edge hit by such a ray is called an order k critical point, as it can be the site of a D^k discontinuity.

8 Results

We have implemented two programs for testing the ideas described above. The first focuses on discontinuity meshing in general flatland scenes using linear elements and point collocation techniques, and the second focuses on Galerkin formulations with elements of arbitrary degree for simple scenes consisting of two edges. The accuracy of the solutions achieved with these algorithms is measured with an error norm.

8.1 Error Measures

The error estimate used in this study is based on the L_2 norm of the difference between a reference (exact) solution u and the finite element approximation $\tilde{u}$. The measure is sensitized by only considering the non-emitted radiosity field, $u - \epsilon$. The relative error measure for the approximate solution is defined as:

$$E_{\mathrm{RMS}} = \frac{\|(u-\epsilon)-(\tilde{u}-\tilde{\epsilon})\|_2}{\|u-\epsilon\|_2}$$

where

$$\|u\|_q = \left(\int_\Omega |u|^q d\Omega\right)^{1/q}$$

To study the limiting case as the mesh is refined consider the asympotic estimate of error given by $\|E_{\mathrm{RMS}}\| \leq Ch^p$ where h is the varying mesh parameter (a measure of element size), p an indicator of the rate of convergence and C is a model-specific constant. The rate of convergence is strongly dependent on the polynomial completeness of the basis functions. For isoparametric elements in this error norm the optimal rates of convergence are $p = 1$ for piecewise constant, $p = 2$ for piecewise linear and $p = 3$ for piecewise quadratic.

8.2 Discontinuity Meshing Experiments

The first program was written for experimentation with a priori discontinuity meshing. This program simulates diffuse interreflection among non-intersecting, simple polygons, with colored, diffuse reflecting and/or emitting edges. The program solves for radiosities using a collocation formulation with analytic form factors (no numerical integration) and linear elements. Either uniform or D^1 discontinuity meshing can be used. Critical points on the m edges are found with the $O(m^3)$ ray tracing algorithm described above. Form factors are calculated with an object space visible edge algorithm using an $O(m^2)$ radial sweepline technique. For a scene discretized into n elements, the total cost of computing the matrices is $O(nm^2 + \alpha n^2)$, where α is the fraction of nonzero elements in the sparse matrix. The systems of equations are solved with successive overrelaxation. In the current implementation, the memory requirements are $6\alpha n^2$ bytes for each of the three (R, G, B) components. For the scenes tested, matrix density α typically ranged between 10% and 40%.

Three display views are supported, an interactive flatland view, which is a top view of the scene, a graph of the red, green, and blue solution curves as a function of arc length, and a schematic of the radiosity matrix. The program runs on a Silicon Graphics workstation, and is able to re-solve and redisplay a scene consisting of 100 elements in about one second, while scenes containing 1000 elements require several minutes.

Fig 15 is a snapshot of the program running on the scene in Fig 1. In the upper left is the top view of flatland, with thick shaded edges (colored on the screen). The white edge at top is a light source, the top right edge is cyan, the edge at right is yellow, the occluder in the middle is black, and the other edges are white. The thin lines and dots show the critical lines and points, respectively. The upper right window is the radiosity matrix (for a coarse mesh) with blocks boundaries and hyperbolic occlusion curves drawn in with lines. Dot area is proportional to $|\mathbf{A}_{ij}|$ here. The largest off-diagonal dots (matrix elements) come from reflex corners in the scene such as the upper left and upper right, which lie at singular points in the kernel of the system. The bottom window is the plot of the red, green, and blue solution functions as computed with a fine discontinuity mesh. White tick marks at bottom show the elements, and vertical red lines (very faint) mark critical points. The shaded strips at bottom are the edge colors.

Since analytic solutions are not known for radiosity problems involving interreflection, the "exact" solution u is taken to be the approximate solution resulting from an extremely fine mesh. Both uniform and discontinuity meshing converge to this solution, verifying that it is a valid reference.

When the test scene of Fig 1 was simulated with discontinuity meshing with $n_{\rm eq} = 91$ equations, it required 73K bytes of memory and 1.3 seconds of CPU time. To achieve the same accuracy with uniform meshing required $n_{\rm eq} = 775$ equations, 4.5M bytes of memory, and 74 seconds. Discontinuity meshing, for this test case, gave results of the same quality as uniform meshing using about 1/60th the time and 1/60th the memory. The above experiment used point collocation on linear elements. Further experiments are needed to determine the relative speed and accuracy of Galerkin techniques and higher order elements.

8.3 Galerkin Experiments

The majority of interesting two-dimensional solution characteristics may be adequately addressed by studying simple cases of the interaction between two "surfaces". Our second program allows experimentation with Galerkin solutions with either uniform or discontinuity meshing on such a two-edge scene, using elements of various degrees. To permit the exact solution to be accurately approximated by alternate means, the surface properties for a perfect diffuse emitter, $\rho_1 = 0$ and $\epsilon_1 = 1$, and perfect diffuse reflector $\rho_2 = 1$ and $\epsilon_2 = 0$ are chosen. The geometry and solution for the three cases considered are shown in Fig 11.

The first and by far the easiest case, the *parallel-model*, is that of two finite parallel surfaces. Second, the *perpendicular-model*, a geometry in which the emitter self shadows a portion of the receiver from a distance is considered. Finally, the *T-corner-model*, a "T"-like geometry in which the emitter intersects the receiver and introduces the kernel singularity is analyzed. For each of the three models the convergence rate of the Galerkin approximation is considered for each of three element types: piecewise-constant (traditional radiosity), piecewise-linear, and piecewise-quadratic.

In the parallel-model, the solution is very smooth and well behaved. Only minimal low order (three point) Gaussian integration rules are required. The results are as expected, uniform convergence of all

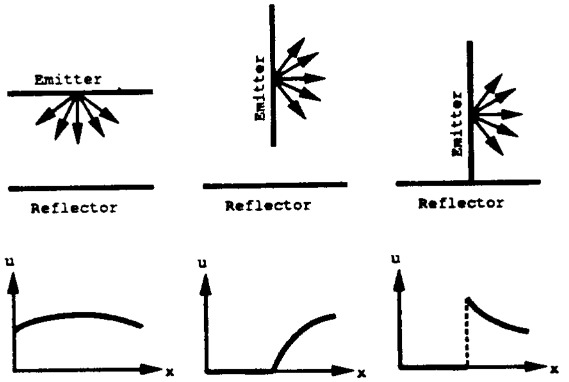

Figure 11: Three simple cases.

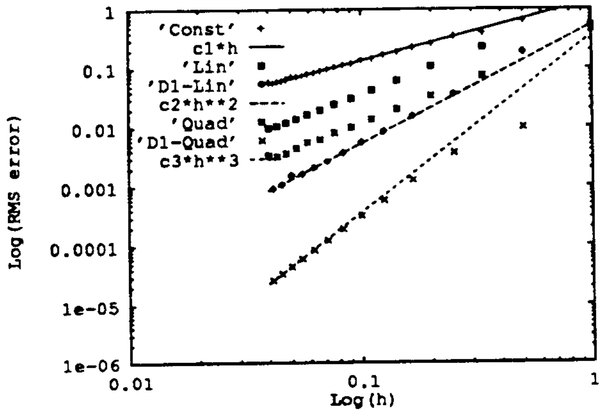

Figure 13: Convergence of perpendicular-model.

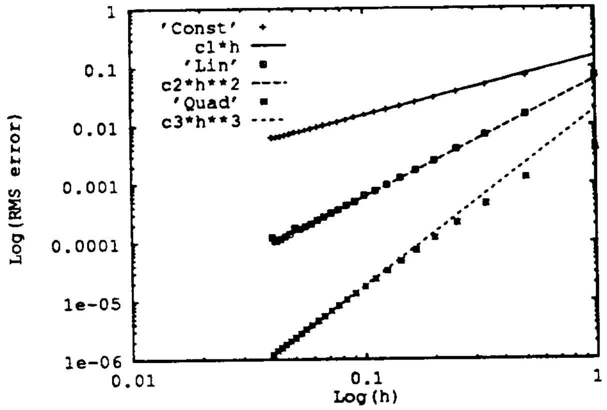

Figure 12: Convergence of parallel-model.

three solutions as the mesh is refined. The datapoints for the piecewise-constant (Const), piecewise-linear (Lin) and piecewise-quadratic (Quad) solutions along with the associated asympotic convergence rates are shown in Fig 12. Optimal convergence rates are achieved without special meshing. The accuracy advantage of higher order elements is quickly apparent for this model.

In the perpendicular-model a new and important wrinkle is added, D^1 discontinuities on the reflecting surface due to self-shadowing by the emitter from a distance. The piecewise-constant element maintains its slow steady $O(h)$ rate of convergence unperturbed by the introduction of additional higher order discontinuities. However, unless special care is taken, some of the additional accuracy advantage of the higher order methods may be lost. In particular, the convergence rate for non-D^1-meshes using higher order elements is shown by datapoints labeled 'Lin' and 'Quad' in Fig 13. Although the higher order elements retain better absolute accuracy, they no longer achieve their optimal convergence rates. Using D^1 discontinuity meshing the ideal rates of convergence may be maintained as shown by the datapoints labeled 'D1-Lin' and 'D1-Quad' in Fig 13.

Finally, in the T-corner-model two new behaviors are introduced. First, by locating one end of the emitter at the center of the reflector the discontinuity introduced by self shadowing is intensified to D^0. Second, although the exact solution is smooth and well behaved, the reflex corner necessitates additional effort to accurately integrate the singular kernel. These two solution features present a considerably more challenging numerical problem. To integrate the corner singularity combinations of both high and low order Gauss rules were utilized. For non-D^0 meshes, all of the solutions display poor convergence as shown by datapoints 'Const', 'Lin', and 'Quad' in Fig 14. With the introduction of D^0 discontinuities, even the piecewise-constant solution, the traditional radiosity workhorse, loses any pretense of accuracy. The higher order solutions exhibit Gibbs phenomena around the discontinuity and the results are no better then those of lower order. Once again the expected rates of convergence may be regained through the use of D^0 meshing as shown in datapoints 'D0-Const', 'D0-Lin' and 'D0-Quad' in Fig 14.

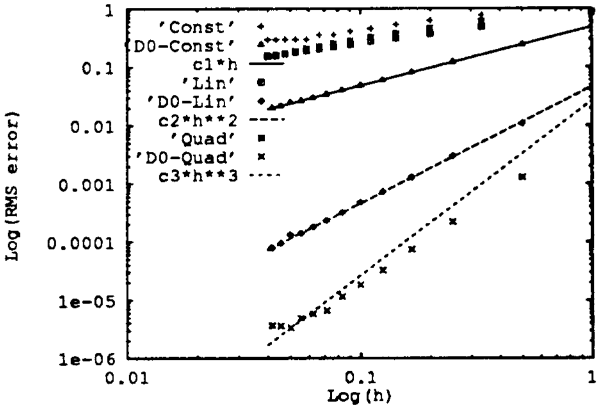

Figure 14: Convergence of T-corner-model.

9 Conclusions

The accuracy of global illumination simulations can be improved using finite element methods. Existing radiosity techniques correspond to the simplest finite element techniques: constant elements formulated with point collocation on a uniform mesh. This approach can be improved upon in several ways. First, the use of linear, quadratic, or higher order elements allows the solution function to be fit more accurately for a given number of elements. Second, the use of Galerkin methods instead of point collocation extracts more information from the kernel of the integral equation, yielding a discretized system of equations that represents the problem with higher fidelity. Third, the solution to many global illumination problems contain infinite numbers of discontinuities caused by occlusion. Shadows are the most obvious example. In order to achieve the theoretical accuracy limited by the chosen element degree, it is necessary to perform discontinuity meshing: locating mesh boundaries on low-order discontinuities in the solution. These three tools could be used independently, but higher order elements and Galerkin formulations will have reduced effect without discontinuity meshing.

Formulas and algorithms have been given for all three techniques. Higher order elements and Galerkin techniques can be applied to radiosity problems using numerical integration methods. Gaussian integration in combination with ray tracing for visibility testing has provided accurate numerical results. The hemicube could also be used, but it is less accurate than Gaussian integration.

Algorithms have been given for discontinuity meshing in two dimensions. Discontinuity meshing in 3-D is more complex [Hec91]. In three dimensions, D^1 discontinuities lie along curves. In a polyhedral environment, these critical curves can arise from edge-vertex events, and edge-edge-edge events, giving rise to straight line and conic discontinuity curves, respectively [GM90]. Discontinuity meshing in 3-D remains a challenging research problem.

Our experiments show that careful selection of element degree, formulation, and mesh give dramatic speed-ups for flatland scenes, suggesting that similar acceleration would be possible in 3-D. However, speed and accuracy comparisons against existing progressive and hierarchical radiosity algorithms remain to be done.

We expect that the finite element approach will provide a path toward very accurate radiosity solutions. In applications requiring only low-accuracy results, cheaper techniques such as progressive radiosity with hardware-assisted hemicube may yield faster results, but for high-accuracy results, as might be required for lighting design or thermal radiation simulations, the finite element approach appears promising.

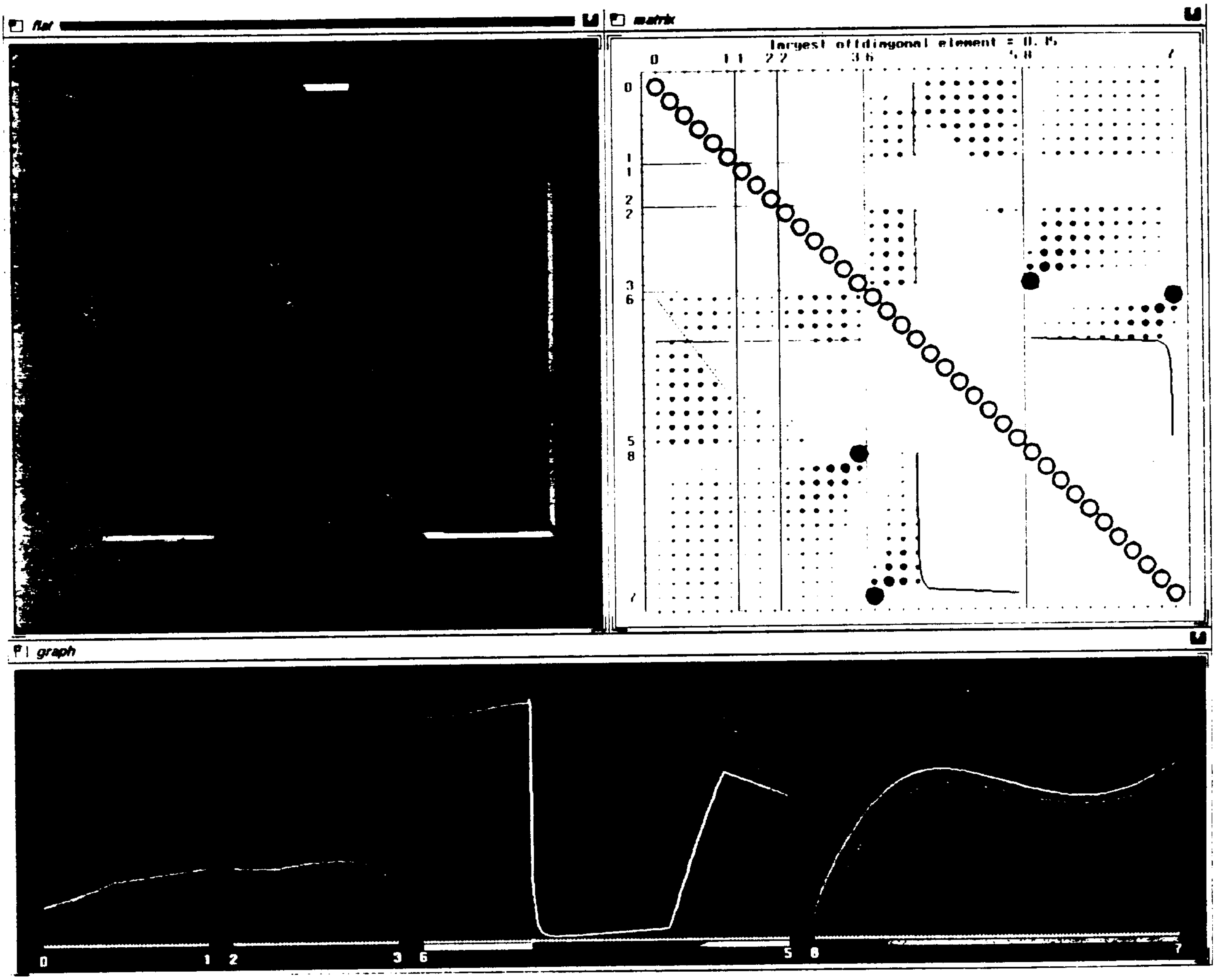

Figure 15: *Screen snapshot of flatland radiosity program.*

References

[Abb84] Edwin A. Abbott. *Flatland: A Romance of Many Dimensions*. Dover, New York, 1884.

[BBB87] Richard Bartels, John Beatty, and Brian Barsky. *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling*. Morgan Kaufmann Publishers, San Mateo, CA, 1987.

[BCO81] Eric B. Becker, Graham F. Cary, and J. Tinsley Oden. *Finite Elements: An Introduction*, volume 1. Prentice-Hall, Englewood Cliffs, NJ, 1981.

[BD89] C. A. Brebbia and J. Dominguez. *Boundary Elements: An Introductory Course*. Computational Mechanics Publications, Southampton, 1989.

[BF89] Chris Buckalew and Donald Fussell. Illumination networks: Fast realistic rendering with general reflectance functions. *Computer Graphics (SIGGRAPH '89 Proceedings)*, 23(3):89–98, July 1989.

[BMSW91] Daniel R. Baum, Stephen Mann, Kevin P. Smith, and James M. Winget. Making radiosity usable: Automatic preprocessing and meshing techniques for the generation of accurate radiosity solutions. *Computer Graphics (SIGGRAPH '91 Proceedings)*, July 1991. To appear.

[BRW89] Daniel R. Baum, Holly E. Rushmeier, and James M. Winget. Improving radiosity solutions through the use of analytically determined form factors. *Computer Graphics (SIGGRAPH '89 Proceedings)*, 23(3):325–334, July 1989.

[CCWG88] Michael F. Cohen, Shenchang Eric Chen, John R. Wallace, and Donald P. Greenberg. A progressive refinement approach to fast radiosity image generation. *Computer Graphics (SIGGRAPH '88 Proceedings)*, 22(4):75–84, Aug. 1988.

[CF90] A. T. Campbell, III and Donald S. Fussell. Adaptive mesh generation for global diffuse illumination. *Computer Graphics (SIGGRAPH '90 Proceedings)*, 24(4):155–164, Aug. 1990.

[CG85] Michael F. Cohen and Donald P. Greenberg. The hemi-cube: A radiosity solution for complex environments. *Computer Graphics (SIGGRAPH '85 Proceedings)*, 19(3):31–40, July 1985.

[CGIB86] Michael F. Cohen, Donald P. Greenberg, David S. Immel, and Philip J. Brock. An efficient radiosity approach for realistic image synthesis. *IEEE Computer Graphics and Applications*, pages 26–35, Mar. 1986.

[Chu88] T. J. Chung. Integral and integro-differential systems. In Minkowycz et al, editor, *Handbook of Numerical Heat Transfer*, pages 579–624. Wiley, 1988.

[DM85] L. M. Delves and J. L. Mohamed. *Computational methods for integral equations*. Cambridge University Press, Cambridge, U.K., 1985.

[Fin72] B. A. Finlayson. *The Method of Weighted Residuals and Variational Principles*. Academic Press, New York, 1972.

[Fle84] C. A. J. Fletcher. *Computational Galerkin Methods*. Springer-Verlag, New York, 1984.

[GM90] Ziv Gigus and Jitendra Malik. Computing the aspect graph for line drawings of polyhedral objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(2):113–122, Feb. 1990.

[GTGB84] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. *Computer Graphics (SIGGRAPH '84 Proceedings)*, 18(3):213–222, July 1984.

[GVL89] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD, 1989.

[Hec90] Paul S. Heckbert. Adaptive radiosity textures for bidirectional ray tracing. *Computer Graphics (SIGGRAPH '90 Proceedings)*, 24(4):145–154, Aug. 1990.

[Hec91] Paul S. Heckbert. *Simulating Global Illumination Using Adaptive Meshing*. PhD thesis, CS Dept, UC Berkeley, June 1991. Tech. Report UCB/CSD 91/636.

[Hor77] Berthold K. P. Horn. Understanding image intensities. *Artificial Intelligence*, 8:201–231, 1977.

[HS90] Pat Hanrahan and David Salzman. A rapid hierarchical radiosity algorithm for unoccluded environments. In *Proceedings Eurographics Workshop on Photosimulation, Realism and Physics in Computer Graphics*, pages 151–171, Rennes, France, June 1990.

[Hug87] Thomas J. R. Hughes. *The Finite Element Method*. Prentice-Hall, Englewood Cliffs, NJ, 1987.

[ICG86] David S. Immel, Michael F. Cohen, and Donald P. Greenberg. A radiosity method for non-diffuse environments. *Computer Graphics (SIGGRAPH '86 Proceedings)*, 20(4):133–142, Aug. 1986.

[Kaj86] James T. Kajiya. The rendering equation. *Computer Graphics (SIGGRAPH '86 Proceedings)*, 20(4):143–150, Aug. 1986.

[KvD83] J. J. Koenderink and A. J. van Doorn. Geometrical modes as a general method to treat diffuse interreflections in radiometry. *J. Opt. Soc. Am.*, 73(6):843–850, June 1983.

[NN85] Tomoyuki Nishita and Eihachiro Nakamae. Continuous tone representation of 3-D objects taking account of shadows and interreflection. *Computer Graphics (SIGGRAPH '85 Proceedings)*, 19(3):23–30, July 1985.

[PH85] D. J. Paddon and H. Holstein, editors. *Multigrid Methods for Integral and Differential Equations.* Clarendon Press, Oxford, 1985.

[Rus87] Holly E. Rushmeier. The zonal method for calculating light intensities in the presence of a participating medium. *Computer Graphics (SIGGRAPH '87 Proceedings)*, 21(4):293–302, July 1987.

[Rus89] Holly E. Rushmeier. Illumination models for computer graphics. Unpublished course notes from Silicon Graphics seminar, 1989.

[SH81] Robert Siegel and John R. Howell. *Thermal Radiation Heat Transfer.* Hemisphere Publishing Corp., Washington, DC, 1981.

[Shi90] Peter Shirley. A ray tracing method for illumination calculation in diffuse-specular scenes. In *Proceedings of Graphics Interface '90*, pages 205–212, May 1990.

[SP89] François Sillion and Claude Puech. A general two-pass method integrating specular and diffuse reflection. *Computer Graphics (SIGGRAPH '89 Proceedings)*, 23(3):335–344, July 1989.

[WCG87] John R. Wallace, Michael F. Cohen, and Donald P. Greenberg. A two-pass solution to the rendering equation: A synthesis of ray tracing and radiosity methods. *Computer Graphics (SIGGRAPH '87 Proceedings)*, 21(4):311–320, July 1987.

[WEH89] John R. Wallace, Kells A. Elmquist, and Eric A. Haines. A ray tracing algorithm for progressive radiosity. *Computer Graphics (SIGGRAPH '89 Proceedings)*, 23(3):315–324, July 1989.

[Whi80] Turner Whitted. An improved illumination model for shaded display. *CACM*, 23(6):343–349, June 1980.

[WRC88] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A ray tracing solution for diffuse interreflection. *Computer Graphics (SIGGRAPH '88 Proceedings)*, 22(4):85–92, Aug. 1988.