Integration Methods for Galerkin Radiosity Couplings

Reid Gershbein
Department of Computer Science, Princeton University

Abstract

Computing energy transfer between objects is the most expensive operation in radiosity systems. This energy transfer operation, known as the irradiance operator, is an integral that, in general, must be calculated numerically. We perform a study of numerical integration techniques to increase the speed of this computation without severely compromising fidelity. A theoretical discussion of numerical integration is presented followed by details of the studied methods. The results of our study give us insight into greatly reducing the cost of the irradiance operator while maintaining accuracy. An adaptive method for choosing Gauss quadrature rules is presented, and our performance analysis of the new adaptive algorithm shows that it can be up to 10 times faster than previous methods.

1 Introduction

The radiosity equation can be written as a simplification of the rendering equation, introduced by Kajiya [8], by assuming that all surfaces are perfectly diffuse (Lambertian):

$$B(x) = B^e(x) + \rho(x) \int dA_{x'} G(x, x')\pi^{-1} B(x')$$

$B(x)$ is the radiosity at point x, consisting of emitted and reflected radiosity, $\rho(x)$ is the reflectance, and $G(x, x') = \frac{cos\theta_x cos\theta_{x'}}{\|x-x'\|^2} V(x, x')$ characterizes the radiant coupling between points x and x'. G accounts for relative surface orientation, distance, and visibility ($V = 0$ or $V = 1$, depending on whether x can or cannot see x'). The integral is taken over the hemisphere about x and represents the amount of energy per unit area received from other surfaces, thus computing irradiance.

Radiosity algorithms are usually based on finite element methods. It is assumed that all functions will be represented in the chosen orthonormal basis $\{M_i\}_{i=1,\ldots,n}$. The linear system is

$$B_i = B_i^e + \sum_{j=1}^{n} K_{ij} B_j$$

$$K_{ij} = \int dA_x \int dA_{x'} \rho(x) G(x, x')\pi^{-1} M_i(x) M_j(x') \tag{1}$$

B_i is coefficient of the radiosity projection onto the basis, B_i^e is the emission function's projection, and K_{ij} is the global irradiance projection. Here, we consider $\rho(x)$ to be constant since it can be removed from the global irradiance integral K_{ij} and applied locally as shown by Gershbein *et al.* [3]. The computation of K_{ij} is by far the most expensive operation of any radiosity system.

There have been many studies of finite-element methods for radiosity [4, 1, 6, 7, 19, 20, 5, 12, 15], and the irradiance projection. A survey of methods for computing the irradiance integral for a constant basis (called the form factor) is discussed by Cohen and Wallace [2]. For non-constant bases, the methods of choice have been Gauss quadrature product rules, except when there are singularities in the kernel [20, 19, 5, 13, 11].

In a hierarchical Galerkin radiosity system, a subdivision strategy is invoked so that each K_{ij} has the same amount of error. When a degree N basis is used, this strategy tries to make the kernel $G(x, x')$ close to a degree N polynomial and the integrand in (1) is at most a degree $2N$ polynomial along each axis since it is simply the product of a basis function and G. This assumption about $G(x, x')$ is not valid when there is a change in visibility, but the subdivision tries to resolve this problem. Previously, the computation of K_{ij} for an order N basis used $(N+1)^4$ kernel evaluations. The locations and weights of these evaluations were derived from Gauss quadrature rules.

We explore the efficiency of the computation of the irradiance integral by studying a number of quadrature methods. First, we establish a framework for our study through a brief discussion of numerical integration theory. The theoretical merits of different methods are presented and a number of tests are conducted to evaluate their performance on the irradiance integral. Based on these results we construct an adaptive method for choosing Gauss-Legendre quadrature rules and discuss its performance, which can result in a factor of ten speedup.

2 Numerical Integration Theory

2.1 Numerical Integration With One Variable

A quadrature rule is any numerical method that approximates the integration of a single variable function, $f(x)$. We look at quadrature rules that evaluate the function at a finite set of points. These quadrature rules have the following form

$$\int dx\, w(x) f(x) = \sum_{i=1}^{N} C_i f(x_i) + Error \tag{2}$$

where $w(x)$ is a weight function, the points $\{x_i\}_{i=0,\ldots,N}$ are the quadrature points at which we evaluate the function, $\{C_i\}_{i=0,\ldots,N}$ are the quadrature weights, and error value necessary for to get the correct value of the integral.

If we look at deterministically generated points and weights, a criteria for judging the methods is needed. Typically, this criteria is given in terms of the maximum-degree polynomial the quadrature rule will exactly integrate. We are looking for the cheapest way of evaluating these integrals. Based on this, Gauss integration rules are the best. An N point Gauss rule will exactly integrate polynomials of up to degree $2N - 1$. As stated in [16], Gauss quadrature rules compared to methods such as Newton-Cotes and extrapolation yield the most accurate results for the same amount of computation.

When generating points and weights stochastically the error term for a given $f(x)$ and $w(x)$ will be in a range of values, occuring with some probability distribution. In designing Monte Carlo quadratures, most effort is spent reducing the error through techniques such as stratified sampling and importance sampling to reduce variance [9].

2.2 Multi-Variate Numerical Integration

The multiple variable version of a quadrature rule is a cubature rule. For a function of M variables

$$\int \ldots \int dx_1 \ldots dx_m w(x_1, \ldots, x_m) f(x_1, \ldots, x_m) = \sum_{i=1}^{N} C_i f(x_1, \ldots, x_m) + Error \tag{3}$$

For deterministic cubature rules we can require a maximum-degree polynomial exactness criteria to judge our rules as we did with quadratures. A detailed account of such rules can be found in [17] and we now present an abridged discussion of the topic.

A cubature rule in the form (3) has degree d if it is exact for all polynomials in $x_1, \ldots, x_m$ whose exponents sum to $\leq d$. If a cubature rule is of degree d there exists a monomial of degree $d+1$ for which the rule is not exact. This definition has a subtlety that will be important when we look at $product$ and $non\text{-}product$ rules in the following discussion. A cubature rule may be of degree d and additionally be exact for all polynomials that have exponents whose sum is greater than d but has no single exponent greater than d.

Usually, the first step is to take the tensor product of single variable rules. These define a class of cubature rules called product rules. It is not necessary to use the same rule for each variable of the domain and if we choose to do this we reduce the degree of exactness of the formula to that of the lowest degree rule. If we know that the degree of the function in certain variables will be less than others then this is exactly what we want to do. This will be the main property we exploit in our adaptive quadrature approach.

Non-product cubature rules also exist. It is possible to create a rule of degree N that has the advantage of not being exponential in the number of variables being integrated. For example if we look at the 4-D radiosity case, there exists a degree 3 non-product rule that has 8 points versus the degree 3 Gauss product rule which has 16.

Monte Carlo methods can also be extended to multiple integrals in a straightforward manner. We simply define a probability distribution function (PDF) $p(x_1, \ldots, x_m)$ over multiple variables. Once again this error term is a random variable. If the choice of a value for each variable is independent of all other variables, our PDF $p(x_1, \ldots, x_m)$ is the product of the set of single variable PDFs. This is a useful property because it simplifies the process of probabilistically chosing the values of the variables.

3 Intergration Methods Study

All of the methods described in the paper have positive weights and points in the integration domain. These methods typically have less error than methods that have some negative weights and/or have points outside the integration domain [17]. The numerical integration methods we have studied can be broken down into three main categories: Gauss product rules, non-product Rules, and Monte Carlo methods. The goal was integration methods with fewer quadrature points than Gaussian product rules.

3.1 Gauss Product Rules

Univariate Gaussian quadrature rules can integrate degree $\leq 2N - 1$ functions with N function evaluations, which is fewer evaluations than other methods. Multivariate

Gauss product rules are constructed through the tensor product of univariate rules. We exclusively studied Gauss-Legendre quadrature rules since they have been the most commonly used method when determining the non-singular irradiance integral we are studying, though other Gauss rules can be considered. The irradiance integral is 4-dimensional, two variables on the source surface and two on the receiver, so we may construct a product rule from four different univariate rules.

Previously, when a degree N basis was used a degree $2N - 1$ Gauss rule, $N + 1$ quadrature points, was used for each variable resulting in $(N+1)^4$ total points. We call this the $FullGauss$ method. Noting that often the source's higher order basis functions have small coefficients, we test Gauss product rules that assume that their contribution is negligible. The rule $ConstSrcGauss$ (constant source) uses 1-point rules on the source and $N+1$ point rules on the receiver resulting in $(N+1)^2$ total points. This rule exactly integrates polynomials of degree $2N - 1$ over each of the receiver's variables. The second is $LinSrcGauss$ (linear source) which uses an 2-point rule for each variable on the source and an $N+1$ rule for each variable on the receiver: $(4*(N+1)^2)$ total points. Similar to $ConstSrcGauss$, this method exactly integrates polynomials up to degree $2N - 1$ on the receiver. We always use an $N + 1$ point rule on the receiver because we assume that all the coefficients of receiver's basis functions are significant. This assumption is made because, essentially, these are the coefficients we are computing.

3.2 Non-Product Rules

There exist fixed point cubature rules that can be classified as degree N and not be exponential in the dimension of the integral. For a given basis, we test a number of these rules that are of the same degree as the $FullGauss$ rule and have fewer function evaluations.

The non-products methods tested are presented by Stroud [17] and our experiments used linear and quadratic bases. In the linear case, $FullGauss$ is degree 3 and uses 16 function evaluations (N = 2, 2^4). The non-product methods that had the least error were $StroudN + 1$, a degree 2 rule that had 5 kernel evaluations (Stroud's C_n 2-1), and $Stroud2N$, a degree 3 rule that had 8 kernel evalutions (Stroud's C_n 3-1). In the quadratic case, $FullGauss$ is degree 5 and used 81 function evaluations (3^4). The best performing non-product rules were $Stroud65$, a degree 5 rule with 65 function evaluations (Stroud's C_n 5-7), and $HierStroud$, which is created by stratifying the receiver's domain into 4 regions, applying $Stroud2N$ to each, and taking the sum.

3.3 Monte Carlo Methods

Monte Carlo integration techniques' error and performance can not be classified in the same way as fixed point rules because they are stochastic. We tested Monte Carlo methods with the same number of samples as the $FullGauss$ rules with uniform, N-Rook [14], and Stratified PDFs. We only show the results of the stratified sampling method, $Stratified$, because it reduces variance most and generally has the smallest error. The $Stratified$ methods for an degree N basis breaks each variable domain into $N + 1$ regions and perform one function evaluation in each of the $(N + 1)^4$ regions from randomly chosen point.

4 Experiments

4.1 Error Metric

We compute the irradiance integral in a variety of situations with linear and quadrative multiwavelets and measure the amount of error. Since, in general, no closed form solution of this integral exists for non-constant bases, we compute our estimate of the "true" answer through a stratified Monte Carlo method. We measure the amount of error of a numerical integration method, I, by computing an upper bound on the L_1 norm of the difference error of the radiosity computed by the "true" solution and I. This upper bound is derived as follows. First, we have an approximation of the radiosity function in terms of our orthonormal multiwavelet basis of degree $n+1$. We expand the L_1 norm and derive an upper bound. This upper bound can be computed by taking the differences of the coefficients of the individual basis functions and is defined as follows:

$$\int dx \mid (b_1 B_1(x) + \ldots b_n B_n(x)) - (\hat{b}_1 B_1(x) + \ldots + \hat{b}_n B_n(x)) \mid$$

$$\leq \mid b_1 - \hat{b}_n \mid (\int dx \mid B_1(x) \mid) + \ldots + \mid b_n - \hat{b}_n \mid (\int dx \mid B_n(x) \mid) \qquad (4)$$

We call this error metric "Abs L1 error".

4.2 Experiment Description

We perform our tests with the Legendre polynomials as our basis, these are a multiwavelet basis. In all of our experiments, we separately measure the contribution from the source's constant, linear, and (when we use a quadratic basis) quadratic Legendre polynomials to the error and assume that their coefficients are 1. In the figures, each method has a cost less than or equal to those to the left. Consult Section 3 for the number of quadrature points used in each rule. Each figure's bar's total height represents the total amount of error. The bar's subsections represent the partial contribution of the source's basis functions.

The methods in the first 8 figures are compared against a 160,000 sample stratified Monte Carlo method. Figures 1 and 2 measure the error with two parallel unit area quadrilaterals that are a unit apart. Figures 3 and 4 show two perpendicular unit quadrilaterals that are .25 units away from each other (they were placed .25 units away from each other in order to avoid the singularity). To test partial occlusion, Figures 5 and 6, depict the parallel scenario with an occluding parallel quadrilateral with area .25 square units .25 units away from the source. Figures 7 and 8 have the occluder halfway.

Figures 9 through 12 compare a number of methods in a scenario where subdivision occurs. This scenario uses a BF refinement [6] oracle with visibility criteria based on the Teller-Hanrahan visibility classification [18] and a quadtree subdivison. A unit quadrilateral source is 10 units above a parallel 400 unit receiver. There is a unit occluder halfway between them and it is oriented 45 degree around it's normal in order to go against the quadtree meshing. Figures 9 through 12 show the sum Abs L_1 error in all of the unoccluded or partially occluded links where each link is compared against a 10,000 point stratified Monte Carlo method.

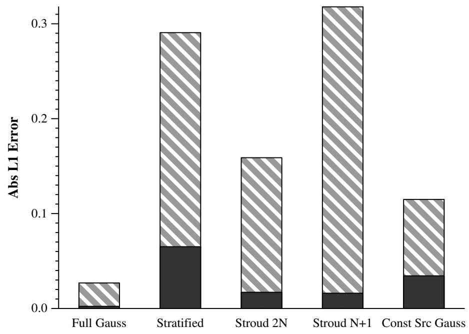

Figure 1: Abs L1 Error: Unoccluded Parallel Quads: Linear Multiwavelets, Solid Grey = Constant Source, Striped Grey = Linear Source

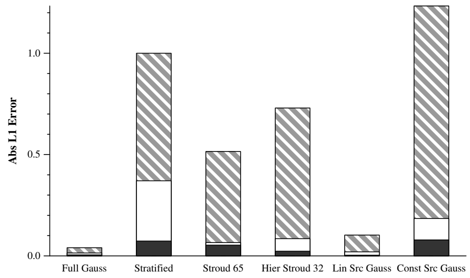

Figure 2: Abs L1 Error: Unoccluded Parallel Quads: Quadratic Multiwavelets, Solid Grey = Constant Source, Solid White = Linear, Striped Grey = Quadratic

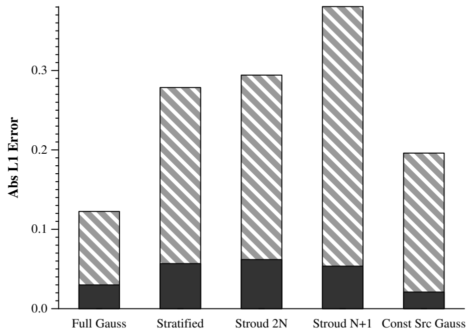

Figure 3: Abs L1 Error: Unoccluded Perpendicular Quads: Linear Multiwavelets

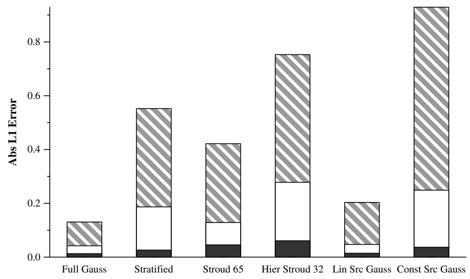

Figure 4: Abs L1 Error: Unoccluded Perpendicular Quads: Quadratic Multiwavelets.

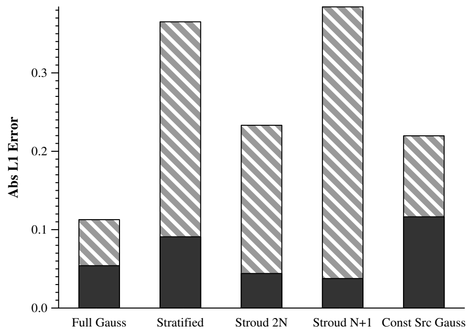

Figure 5: Abs L1 Error: Partially Occluded Parallel Quads, Blocker .25 From Source: Linear Multiwavelets

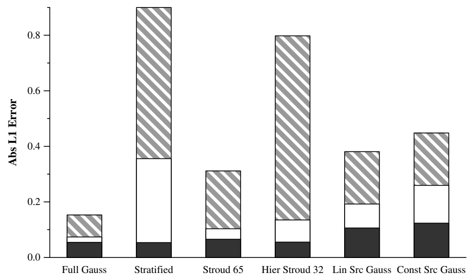

Figure 6: Abs L1 Error: Partially Occluded Parallel Quads, Blocker .25 From Source: Quadratic Multiwavelets

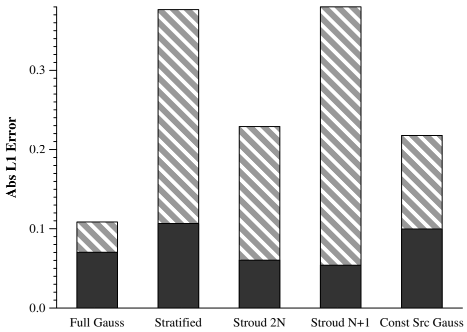

Figure 7: Abs L1 Error: Partially Occluded Parallel Quads, Blocker .5 From Source: Linear Multiwavelets

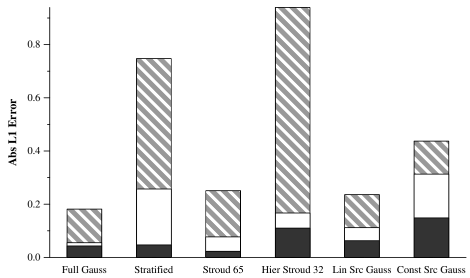

Figure 8: Abs L1 Error: Partially Occluded Parallel Quads, Blocker .5 From Source: Quadratic Multiwavelets

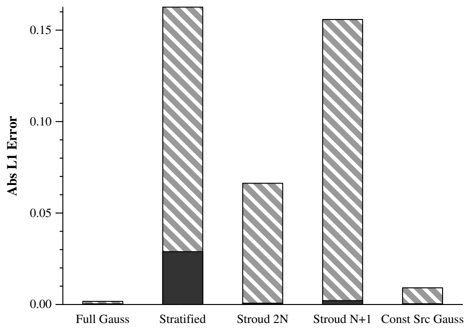

Figure 9: Abs L1 Error - Unoccluded Links: Subdivision Around Occluder : Linear Multiwavelets

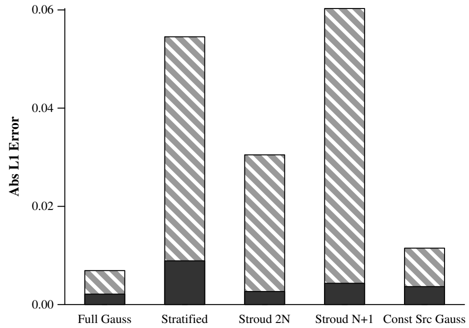

Figure 10: Abs L1 Error - Partially Occluded Links: Subdivision Around Occluder : Linear Multiwavelets

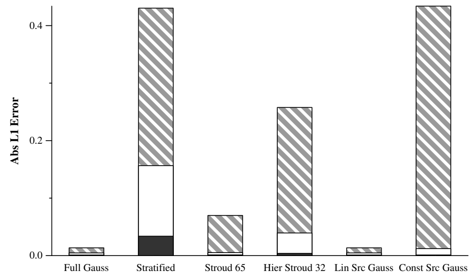

Figure 11: Abs L1 Error - Unoccluded Links: Subdivision Around Occluder : Quadratic Multiwavelets

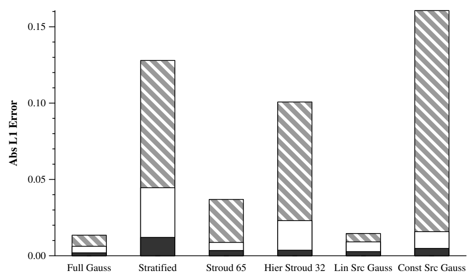

Figure 12: Abs L1 Error - Partially Occluded Links: Subdivision Around Occluder : Linear Multiwavelets

4.3 Experiment Result

Our results show the overall strength of Gauss methods. Predictably, the Gauss methods perform extremely well in the unoccluded scenarios often having comparable amounts of error with fewer quadrature points than other methods. In the linear case, $ConstSrcGauss$ is the least expensive method tested and always outperforms the more expensive non-Gauss rules. In the quadratic case, $LinSrcGauss$ rule has less error than the more expensive non-Gauss rules. Additionally, Gauss methods perform extremely well in the partially occluded scenarios. We find that the variance of the stratified Monte Carlo methods of 16 or 81 points causes so much error that we get better performance from the Gauss rules. The Gauss superiority during partial visibility occurs in the presence of a non-constant basis . When the coupling is partially visible and a constant basis is used the Gauss rules do not outperform the Monte Carlo.

We see more benefits of Gauss rules by looking at the breakdown of error due to the degree of the source's basis function. In the linear case, we can look at the amount of error that the $FullGauss$ methods gives in a specific scenario. By looking at the dark section of the bar we can see which methods give us the the same or less error when the source is constant. Often $ConstSrcGauss$ and the other methods fullfill that property. Similarly, the same can be done for the quadratic case. Now, we can compare error of a source's constant (low black bar) and linear (middle white bar) for a particular method with the total error of $FullGauss$. We see that the $LinSrcGauss$ rule performs quite well when the source is constant or linear. The $ConstSrcGauss$ often performs comparably when the source is constant.

5 Adaptive Algorithm

Based on the above observations, we now present an adaptive way of choosing Gauss quadrature rules that compute the irradiance coupling between basis functions in a hierarchical Galerkin wavelet radiosity system. The magnitudes of the source's basis functions' coefficients and occlusion factors will be used to choose a Gauss quadrature method. The choice of quadrature method is made after we have performed subdivision and have determined that the geometric kernel G is sufficiently smooth. The oracle that makes this decision could be, for example, BF refinement described by Hanrahan *et al.* [6] or Gortler *et al.*'s polynomial fitting approximation [5]. The subdivision strategy could be a quadtree subdivison, or discontinuity meshing as presented by Lischinski *et al.* [10].

Once we have determined the level at which we should compute the irradiance coupling between basis functions, we must choose a quadrature method. This can be done by noting that if M of the highest order basis functions for a variable have very small coefficients, then any integration in the presence of these functions is also small. Therefore, in that variable's domain the basis is essentially of degree $N - M$ instead of degree N and we can use an $(N - M) + 1$ point rule rather than an $N + 1$ point rule. When computing this integral we know the radiosity coefficients of the basis functions of the source. We may take advantage of this fact and use the $(N - M) + 1$ rule along each variable of the source surface (possibly with a different M for each variable, M_0 and M_1) when the coefficients are sufficiently small. This results in $(N - M_0 + 1)(N - M_1 + 1)(N + 1)^2$ kernel evaluations instead of the $(N + 1)^4$

of previous methods. Similar knowledge of the receiver's basis function is not available, so we must assume that it is a higher-frequency function.

We now construct an adaptive algorithm based on this criterion. When the basis is of degree N, we use an $N + 1$ point Gauss rule along each axis of the receiver. We then choose a rule of 1 to $N + 1$ points for each axis of the source based on an error threshold, $ErrThres$, set by the user. The coefficients of the source are checked along one of its axes and we find the highest order basis function, M, that has a coefficient above $ErrThres$. An $M + 1$ point rule is then used along that axis and this process is repeated for the other axis. Finally, tensor products of the source and receiver rules are used evaluate the integral.

There is more error in partially occluded regions and this should be accounted for when choosing a quadrature rule. To reduce this error, when the integral is partially occluded we randomly choose an axis on the source that has fewer than $N+1$ points and increase the number of points along the axis by one. After an iteration of the radiosity solver, we subdivide if necessary and check the coefficients of all couplings to see if we need a better quadrature rule due to the energy gain. This additional test is necessary because the higher order coefficients may now be above the threshold.

We have tried the adaptive algorithm in a range of scenes with linear, quadratic, and cubic multiwavelets. These tests were done with a quadtree subdivision. The subdivision oracle was BF refinement [6] and visibility was determined by Teller and Hanrahan's visibility classification [18]. We find that the speedup is half the theoretical best-case maximum of 4, 9, and 16 times for linear, quadratic, and cubic bases, respectively. This theoretical maximum occurs when you have only constant sources. The images in Plate 1 show a comparision of the fixed point method and the adaptive method using a cubic multiwavelet basis. The rightmost is the fixed point method $FullGauss$. The leftmost image used the adaptive algorithm without accounting for partial visibility and had 12 times fewer kernel evaluations. The center image is the adaptive rule accounting for partial visibility and is 8 times faster than the non-adaptive method.

6 Summary and Conclusion

We have presented a study of numerical integration methods for computing irradiance couplings in Galerkin radiosity systems. An adaptive method for choosing Gauss product rules is constructed and tested due to the fact that Gauss rules performed extremely well in our experiments. This algorithm takes advantage of the characteristics of the source's basis functions and accounts for partial visibility. Typically, our algorithm achieves a speedup of half the theoretical maximum (when all sources are constant) , which can be translate into an order of magnitude or greater speedup over non-adaptive methods.

The study and algorithm should be extended to deal with subdivision strategies and oracles other than quadtree and BF refinement. The error due to the current assumptions of the kernel should be looked at more closely, more numerical integration methods can be explored, and new basis functions tried.

We look optimistically upon extensions of the algorithm for Galerkin radiance systems. The adaptive approach's speedup is proportional to the basis degree and the dimensionality of the integral since more kernel evaluations can be avoided. This situation improves for adaptive approachs applied to the 6-dimensional angularly-dependent radiance case. This is due to the extraordinary amount of samples previously used to compute couplings, $(N+1)^6$ for a degree N basis.

7 Acknowledgements

I would like to thank Pat Hanrahan for support, discussions, and creating a fruitful environment in which to work. Peter Schröder and Seth Teller created the wavelet radiosity infrastructure upon which this work was developed. Thanks go to Peter Schröder and Don Mitchell for many patient hours of teaching, discussing, stimulating, and befriending. Phil Lacroute saved me with LaTeXsettings. This research was supported by equipment grants from Apple and Silicon Graphics Computer Systems, and a research grant from the National Science Foundation (CCR-9208966).

References

[1] COHEN, M. F., AND GREENBERG, D. P. The hemi-cube: A radiosity solution for complex environments. *Computer Graphics 19*, 3 (July 1985), 31–40.

[2] COHEN, M. F., AND WALLACE, J. R. *Radiosity and Realistic Image Synthesis*. Academic Press Professional, Boston, 1993.

[3] GERSHBEIN, R., SCHRÖDER, P., AND HANRAHAN, P. Textures and radiosity: Controlling emission and reflection with texture maps. In *Computer Graphics 1994* (August 1994), Siggraph, pp. 51–58.

[4] GORAL, C. M., TORRANCE, K. E., GREENBERG, D. P., AND BATTAILE, B. Modelling the interaction of light between diffuse surfaces. *Computer Graphics 18*, 3 (July 1984), 212–222.

[5] GORTLER, S., SCHRÖDER, P., COHEN, M., AND HANRAHAN, P. Wavelet radiosity. In *Computer Graphics 1993* (August 1993), Siggraph, pp. 221–230.

[6] HANRAHAN, P., SALZMAN, D., AND AUPPERLE, L. A rapid hierarchical radiosity algorithm. *Computer Graphics 25*, 4 (July 1991), 197–206.

[7] HECKBERT, P. S. Radiosity in flatland. *Computer Graphics Forum 2*, 3 (1992), 181–192.

[8] KAJIYA, J. T. The rendering equation. *Computer Graphics 20*, 4 (1986), 143–150.

[9] KALOS, M. H., AND WHITLOCK, P. A. *Monte Carlo Methods Volume I: Basics*. Whiley-Interscience, New York, 1986.

[10] LISCHINSKI, D., TAMPIERI, F., AND GREENBERG, D. P. Combining hierarchical radiosity and discontinuity meshing. In *Computer Graphics 1993* (August 1993), Siggraph, pp. 199–208.

[11] SCHRÖDER, P. Numerical integration for radiosity in the presence of singularities. In *Fourth Eurographics Workshop on Rendering* (1993).

[12] SCHRÖDER, P., GORTLER, S. J., COHEN, M. F., AND HANRAHAN, P. Wavelet projections for radiosity. In *Fourth Eurographics Workshop on Rendering* (June 1993), Eurographics, pp. 105–114.

[13] SCHRÖDER, P., AND HANRAHAN, P. On the form factor between two polygons. In *Computer Graphics 1993* (August 1993), Siggraph, pp. 163–164.

[14] SHIRLEY, P. Discrepancy as a quality measure for sampling distributions. In *Eurographics '91* (September 1991), pp. 183–193.

[15] SMITS, B., ARVO, J., AND GREENBURG, D. A clustering algorithm for radiosity in complex environments. In *Computer Graphics 1994* (August 1994), Siggraph, pp. 435–442.

[16] STOER, J., AND BULIRSCH, R. *Introduction to Numerical Analysis*. Springer Verlag, New York, 1980.

[17] STROUD, A. H. *Approximate Calculation of Multiple Integrals*. Prentice-Hall, New Jersey, 1971.

[18] TELLER, S., AND HANRAHAN, P. Global visibility algorithms for illumination computations. In *Computer Graphics 1993* (August 1993), Siggraph, pp. 239–246.

[19] TROUTMAN, R., AND MAX, N. Radiosity algorithms using higher-order finite elements. In *Computer Graphics 1993* (August 1993), Siggraph, pp. 209–212.

[20] ZATZ, H. R. Galerkin radiosity: A higher-order solution method for global illumination. In *Computer Graphics 1993* (August 1993), Siggraph, pp. 213–220.