Accurate Error Bounds for Multi-Resolution Visibility

Cyril Soler, François Sillion

iMAGIS*, Laboratoire GRAVIR/IMAG-INRIA

Abstract: We propose a general error-driven algorithm to compute form factors in complex scenes equipped with a suitable cluster hierarchy. This opens the way for the efficient approximation of form factors in a controlled manner, with guaranteed error bounds at every stage of the calculation. In particular we discuss the issues of bounding the error in the form factor approximation using average cluster transmittance, combining subcluster calculations with proper treatment of visibility correlation, and the calculation and storage of the necessary information in the hierarchy. We present results from a 2D implementation, that demonstrate the validity of the approach; the form factor approximations are effectively bounded by the user-supplied threshold.

Keywords: Error bounds, Hierarchical radiosity, Multi-resolution visibility, Error-driven refinement, Visibility correlation.

1 Introduction

In recent years a vast body of research has been devoted to the refinement of advanced simulation techniques such as the radiosity method, offering either improved accuracy, faster computations, or the ability to progressively refine a solution [1]. However the inherent quadratic complexity of the radiosity method [8] requires the use of hierarchical formulations to obtain very accurate solutions in reasonable time [2]. For scenes consisting of large numbers of independent objects, *clustering* techniques must be employed to construct a suitable three-dimensional hierarchy throughout the scene [9, 5].

In all of the above techniques, the calculation of visibility relationships is one of the most time-consuming stages of the simulation. Visibility must be computed for each interaction to quantify the transfer of energy, and visibility information is also useful during the hierarchical refinement stage to orient the computation effort to areas of partial visibility.

In this paper we consider the case of very complex scenes consisting of a great number of (small) objects. Such scenes can be encountered in applications such as the simulation of energy fluxes under a vegetation cover. In these scenes, each visibility calculation entails the consideration of many potential occluders. Some calculations of visibility informations, based on geometrical configurations [11], or on the analogy with scattering volumes [5], have been proposed before, but without a precise characterization of the errors introduced.

Our work focuses on the acceleration of visibility calculations in complex scenes. Specifically, we seek to provide practical and accurate algorithms to approximate visibility while providing trusted bounds on the error incurred. A general error-driven algorithm for the computation of form factors is introduced, and the underlying issues are identified and discussed. We introduce explicit error bounds for the computation of approximate transmittance using a measure of occluder density, and suggest possible precomputation strategies to store the required information at the cluster level. Our results extend the notion of multi-resolution visibility [6], whereby an appropriate level

* iMAGIS is a joint research project of CNRS/INRIA/INPG/UJF. Postal address: B.P. 53, F-38041 Grenoble Cedex 9, France. Contact E-mail: `Cyril.Soler@imag.fr`.

of the cluster hierarchy is automatically selected to represent the set of occluders. Using our proposed error bounds, the visibility calculation is performed with the highest possible cluster level that ensures enough accuracy.

2 Notations for visibility and form factor estimation

2.1 Transmittance

We define the *transmittance* $\tau_C(L)$ of a cluster C along a line L as

$$I_{out} = \tau_C(L) I_{in}$$

where I_{in} and I_{out} are the incoming and outgoing intensities of a light ray supported by L. If C is totally opaque along L, the transmittance is 0, whereas if some light can travel through C along L, $\tau_C(L)$ is a positive value less than 1.

2.2 Form factor

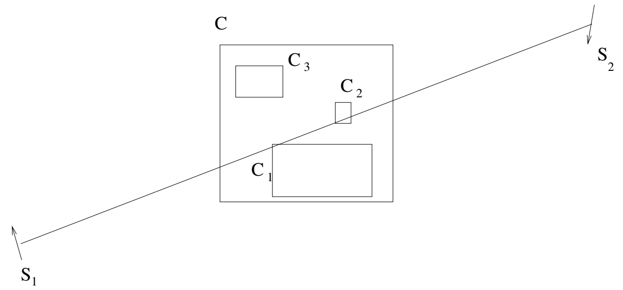

Fig. 1. General form factor notations.

Let S_1 and S_2 be two surfaces, and C a cluster partially occluding visibility between these surfaces (Fig.1). The form factor between S_1 and S_2 is

$$F_{12} = \int_{x_1 \in S_1} \int_{x_2 \in S_2} k(x_1, x_2) v(x_1, x_2) dx_1 dx_2$$

where k is the form factor kernel, depending on the distance between x_1 and x_2, and the incident angles of the line L_{x_1,x_2} on surfaces S_1 and S_2. The function v is 1 if x_1 and x_2 are mutually visible, 0 otherwise. If C is the only occluding object between the surfaces, the form factor can be expressed as

$$F_{12} = \int_{x_1 \in S_1} \int_{x_2 \in S_2} k(x_1, x_2) \tau_C(L_{x_1,x_2}) dx_1 dx_2 \qquad (1)$$

2.3 Form factor approximation

We define the *directional transmittance* of a cluster C in the direction d as the mean value of its transmittance on all lines of direction d that intersect C:

$$\overline{\tau}_C(d) = \frac{\int_{L \| d \wedge L \cap C \neq \emptyset} \tau_C(L) dL}{\int_{L \| d \wedge L \cap C \neq \emptyset} dL}$$

The main drawback of the integral expression (1) is its high computation cost, due to the unpredictable variations of the visibility function τ_C in the integration domain. An inexpensive way of approximating the form factor is to consider τ as constant on the integration domain, which leads to the following expression, where d_{12} is the "mean" direction of S_1 to S_2:

$$\tilde{F}_C = \overline{\tau}_C(d_{12}) \int_{S_1} \int_{S_2} k(x_1, x_2) dx_1 dx_2$$

If we call F_0 the unoccluded form factor from S_1 to S_2, we get

$$\tilde{F}_C = \overline{\tau}_C(d_{12}) F_0$$

Such an approximation is equivalent to ignoring the correlation between the form factor kernel k and the visibility component τ_C. Besides, we shall see later that it is possible to obtain a multi-level approximation of the form factor by deciding at which depth of the hierarchy we replace F_C by $\tilde{F}_C$.

3 An error-driven algorithm for multi-resolution visibility

We consider the problem of obtaining a controlled approximation of the form factor between S_1 and S_2, occluded by a cluster C that is the root of a cluster hierarchy. This means rapidly computing an approximate value $\mathcal{F}^\varepsilon$ whose distance to the real form factor is guaranteed to be less than a fixed bound ε.

For this purpose, we equip the hierarchy with the information needed to evaluate a bound on the visibility error incurred when replacing the exact transmittance of each cluster in the form factor computation by its directional transmittance, computed in the mean direction of the surfaces.

We denote by BVE a bound on the visibility error. This function therefore depends on both the computation configuration (surfaces relative positions, size orders of magnitude) and the characteristics of the cluster itself such as sparsity, uniformity, emptiness... The relation the BVE function must verify for each cluster is:

$$|\tilde{F}_C - F_C| \leq \text{BVE}(S_1, S_2, C)$$

We propose to compute a controlled approximation of the form factor with the following algorithm: if the current cluster verifies $\text{BVE}(S_1, S_2, C) \leq \varepsilon$, then the returned approximation of the form factor can be $\tilde{F}_C$. Otherwise, go down the hierarchy and compute for each subcluster $C_i, i = 1..n$ of C, an error-bounded approximation $\mathcal{F}_i^{\varepsilon_i}$, with appropriate ε_i, of the form factors obtained for each subcluster C_i of C, and combine these values to obtain an ε-bounded approximation of the requested form factor.

Such an algorithm requires the solution of the following problems:

- building and storing a reliable bound function of the error incurred when using $\tilde{F}_C$ instead of F_C for each cluster of the hierarchy;
- knowing how to compute the form factor between S_1 and S_2 occluded by a cluster C using the values of separately computed form factors obtained for all subclusters $C_1, ...C_n$ of C;
- computing and storing approximate visibility information for each cluster of the hierarchy, which means storing the directional transmittance of each cluster.

The following sections provide possible answers to all these questions.

4 Bound on the form factor approximation error

In this section we consider the BVE function. In order to evaluate BVE(S_1, S_2, C) we must store with each cluster some information related to its visibility characteristics. In [6], Sillion and Drettakis proposed to directly store geometrical information giving an acceptable but not mathematicaly controlled way to approximate the form factor.

What we do is compute and store the required elements to obtain a sufficiently accurate visibility error bound criterion. This information takes the form of a couple of directional functions coming from a mathematical bounding [10] of the error $|\tilde{F}_C - F_C|$[10]. In order to let the recursive algorithm offer a smooth control of visibility error as a function of the error bound ε, the BVE function must decrease when going down the hierarchy. Section 8 gives an example of such a function in 2D. We are currently developing a BVE function in 3D.

5 Recombination

When the error-bounded algorithm decides to use the contents of a cluster instead of the local approximation $\tilde{F}_C$, recursive application of the algorithm provides error-bounded estimates of the form factors considering each subcluster as unique occluder, respectively. These form factors must be combined to obtain an ε-bounded estimate of the form factor at higher level. In order to derive a recombination formula, we express the form factor using line set densities [3]:

$$F(S_1, S_2, C) = \frac{\int_{L \cap S_1 \neq \emptyset \wedge L \cap S_2 \neq \emptyset \wedge L \cap C = \emptyset} dL}{\int_{L \cap S_1 \neq \emptyset} dL}$$

Let $\mu(S_1, S_2, C_1, ..., C_k)$ denote the measure of the set of lines that intersect surfaces S_1, S_2 and clusters [2] $C_1, C_2, ...C_k$. The form factor becomes

$$F(S_1, S_2, C) = \frac{1}{\mu(S_1)}[\mu(S_1, S_2) - \mu(S_1, S_2, C)] \tag{2}$$

Which becomes, in the particular case of no occluding cluster:

$$F_0 = \frac{\mu(S_1, S_2)}{\mu(S_1)}$$

By using the measure of the intersection and union of sets, we obtain

$$\mu(S_1, S_2, C) = \sum_{1 \leq i_1 \leq n} \mu(S_1, S_2, C_{i_1}) - \sum_{1 \leq i_1 < i_2 \leq n} \mu(S_1, S_2, C_{i_1}, C_{i_2}) + ...$$
$$+ (-1)^n \mu(S_1, S_2, C_1, C_2, ..., C_n)$$

When replacing this expression in (2) we get

$$F(S_1, S_2, C) = \frac{1}{\mu(S_1)}[\mu(S_1, S_2) - \mu(S_1, S_2, C_1) - ... - \mu(S_1, S_2, C_n)] + \chi$$

with $\chi = \frac{1}{\mu(S_1)}[\sum_{1 \leq i_1 < i_2 \leq n} \mu(S_1, S_2, C_{i_1}, C_{i_2}) + ... + (-1)^n \mu(S_1, S_2, C_1, ..., C_n)]$

[2] For such intersections we consider clusters to be the set of their contained objects

i.e $$F(S_1, S_2, C) = \sum_{1 \le i \le n} F(S_1, S_2, C_i) - (n-1)F_0 + \chi$$

The term χ is called the correlation factor of subclusters $C_1, C_2, ...C_n$, and expresses a complex interaction between their visibility functions. We can therefore obtain an accurate value of F using $F_1, ..., F_n$ provided that χ is low enough:

$$F \approx F_1 + F_2 + ... + F_n - (n-1)F_0$$

Obtaining a accurate bound on χ is difficult for large values of n. At this time, we use an estimate of it for $n = 2$, which implies that we use only a binary hierarchy. For $n = 2$, the expression leads to

$$F = F_1 + F_2 - F_0 + \chi \quad \text{with} \quad \chi = \frac{\mu(S_1, S_2, C_1, C_2)}{\mu(S_1)} \tag{3}$$

We can express the correlation in terms of transmittance functions as

$$\chi = \frac{1}{\mu(S_1)} \int_{S_1} \int_{S_2} (1 - \tau_{C_1}(L_{x_1,x_2}))(1 - \tau_{C_2}(L_{x_1,x_2}))dx_1 dx_2$$

The correlation of subclusters is of great importance in any visibility computation. Consider for instance the 2D case of computing the form factor of two vertical segments occluded by a cluster with two subclusters made of regularly spaced segments (Fig.2), so that each subcluster lets nearly fifty percent of light travel along the horizontal direction. The form factors F_1 and F_2 will be both very close to $\frac{F_0}{2}$, depending on the relative

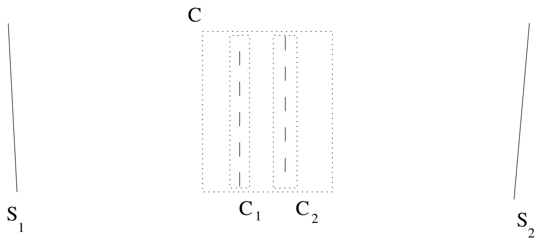

Fig. 2. A bad configuration for recursive computation.

position of C_1 and C_2. The correlation term χ can range from 0 to $\frac{F_0}{2}$. According to expression (3), this implies that the form factor can take values from 0 to $\frac{F_0}{2}$. In other words, the directional transmittance of a cluster does not depend only on that of its subclusters. The same problem occurs for the BVE criterion itself, which prevents us from using a simple recursive algorithm to compute the approximate visibility information.

When recombination is impossible for the required accuracy ε, we have to compute the form factor as a classical integration along the surfaces S_1 and S_2. In order to benefit from the hierarchy coherence, we can still use the BVE criterion to recursively compute an approximate transmittance $th_C(\rho, \theta)$ of the cluster C: if $\text{BVE}(S_1, S_2, C) \le \varepsilon$ we use the approximation $\overline{\tau}(\theta)$, otherwise we compute the product of such approximations for subclusters of C obtained by recursive calls. Then we obtain an approximation of the form factor F by integration over surfaces, which essentially amounts to explicitly computing the correlation:

$$HFormFactor(S_1, S_2, C) = \int_{S_1}\int_{S_2} \tau h_C(x_1, x_2)k(x_1, x_2)dx_1dx_2$$

6 Practical algorithm

The form factor calculation algorithm can now be expressed as a recursive function of surfaces S_1 and S_2, and cluster C. This is a two-stage algorithm: when correlation of the subclusters is low enough, we recombine recursively form factors obtained with the subclusters of C. Otherwise, we compute the form factor using the hierarchical transmittance of C, as described in the previous section. The algorithm is thus the following:

```
Function RFormFactor(S_1, S_2, C, ε)
   if BVE(C, S_1, S_2) < ε then
      return τ̄F_0
   else
      if χ(C_1, ..., C_n) < ε then compute
            F_1 = RFormFactor(S_1, S_2, C_1, ε_1)
            ...
            F_n = RFormFactor(S_1, S_2, C_n, ε_n)
            return F_1 + F_2 + ... + F_n − (n − 1)F_0
      else
            return HFormFactor(S_1, S_2, C)
```

When recursively computing $RFormFactor(S_1, S_2, C_i, \varepsilon_i)$, we must ensure that the total approximation error is still lower than ε after summation of all approximate form factors with the subclusters C_i of C. This can be done in two different ways. The simplest is to do each recursive call with an error bound of $\frac{\varepsilon}{n}$. This method causes the error bound to decrease exponentially with the current hierarchy depth, so that it rapidly goes below the value of BVE , and thus limits critically the number of recombination calls. Moreover, there exists subclusters for which the effective error bound is very small, for instance totally opaque clusters or leaves of the hierarchy. Thus, allocating $\frac{\varepsilon}{n}$ for such approximations is wasting a precious error margin.

A more efficient method consists in allocating a certain part of ε for the calculation of $RFormFactor(S_1, S_2, C_1)$, and getting back the effective error bound used. We next subtract this value from ε, which gives the remaining error margin ε' for subsequent approximations, and continue in the same manner. The effective error at this level itself will be the sum of all effective errors of the calculations with C_1, ...C_n. We have implemented this latest method, which gives satisfying results (Sec.8).

7 Managing multi-scale visibility information

The philosophy of pre-computing multi-scale visibility information is that it needs to be calculated once for the cluster hierarchy of a given object, and then can be used in every scene the object is placed in. This property comes directly from the fact that the visibility information only depends on clusters themselves and never on the computing configuration or surfaces.

In this section we discuss the different issues related to the problem of computing, storing and updating multi-scale visibility information in the hierarchy.

7.1 Computing

A natural way to compute the information would be to use a recursive algorithm, i.e compute the directional transmittance and criterion functions of each cluster using only those of its subclusters. We have seen in section 5 why it is generally not possible.

On the other hand, we could compute directly the visibility information for each cluster separately, but first this would not take any advantage of the coherence of the cluster hierarchy, and secondly would require extensive geometrical calculation on clusters that may contain a great number of objects.

This means that we should compute recursively the visibility information using a temporary more precise information that is passed along the hierarchy during the computation and then thrown away, from which we deduce at each level of the hierarchy the required approximate values and error criterion functions. The temporary information we use consists in a sampling of the current cluster transmittance function, which can be easily recursively computed. At the lowest level of the hierarchy, i.e for leaves, a geometrical computation gives us the necessary information that we propagate in the upper levels. In all cases, the computing method must be accurate enough to produce a lowest possible but true BVE criterion.

The computation cost of this method is linear with respect to the total number of clusters times the constant cost of computing the visibility information for each cluster. Moreover, the hierarchical structure allows one to implement a parallel version of the recursive form factor algorithm, which would reduce significantly the computation cost.

7.2 Storage

Two different questions hide under the storage problem: how to store the visibility information so that (a) it can be easily accessed during the form factor computation stage but (b) consumes the least possible disk space.

These two requirements have opposite solutions: the better way to store a function so that it can be accessed in constant time is to sample it. This is a very memory-expensive storage method. On the other hand, as the visibility information is for each cluster a set of directional functions, a function-based decomposition such as a spherical harmonic decomposition [7], or a spherical wavelet decomposition [4], can be a very memory-efficient storage solution, but significantly slows down the access to the values of the function itself.

Experiments in 2D and 3D show that the visibility information functions can be quite irregular, especially for periodic-like clusters. This prevents us from using periodic smooth function decompositions like spherical harmonics, which generally produces a "Gibbs effect" around derivative discontinuities. On the other hand, a low level wavelet basis seems to be an efficient method for storing directional functions.

8 Results

We have implemented the whole method in dimension 2. All objects are segments. Clusters are axis-aligned boxes. Any line L in the plane is represented by its polar coordinates ρ and θ, so that the linear equation of L is $x \cos\theta + y \sin\theta = \rho$. A direction is represented by θ. We can write all previously defined functions in terms of polar coordinates, keeping the same names: $\tau(\rho, \theta)$, $\overline{\tau}(\theta)$.

8.1 Refinement criterion

The refinement criterion function is obtained by bounding the expression:

$$|F(S_1,S_2,C)-\tilde{F}| = |\int_{S_1}\int_{S_2} k(\rho,\theta)\tau_C(\rho,\theta)dx_1dx_2-\overline{\tau}(\theta_{12})\int_{S_1}\int_{S_2} k(\rho,\theta)dx_1dx_2|$$

We have obtained the following bound [10]:

$$|\tilde{F} - F(S_1,S_2,C)| \leq \frac{2\delta_\theta J_{12}\sqrt{\Delta_\rho}}{L_1 r_{min}} T(\alpha,\beta,\theta_{12})[R_1(\theta_{12}) + R_2(\theta_{12})]$$

with

$$R_1(\theta) = \sup_{\theta' \in [\theta-\delta_\theta, \theta+\delta_\theta]} ||\tau_C(.,\theta') - \tau_C(.,\theta)||_{L^2}$$

$$R_2(\theta) = ||\tau_C(.,\theta) - \overline{\tau}(\theta)||_{L^2}$$

$$T(\alpha,\beta,\theta) = \tfrac{1}{2}\cos(\alpha-\beta) - \tfrac{1}{2}\cos(\alpha+\beta-2\theta)$$

L_1 is the length of S_1, r_{min} the distance of one segment to the other, δ_θ and Δ_ρ the diameters of the integration intervals in θ and ρ. θ_{12} is the middle of the integration interval in θ, and J_{12} is a bound on the jacobian of the cartesian-to-polar coordinate transform. The T term expresses the relative position of the two segments. The R_1 and R_2 functions expresses the irregularities of the transmittance of C in θ and ρ.

As required for our recursive form factor algorithm, this bound is a function of both the current configuration ($T(\alpha,\beta,\theta_{12}, J_{12})$, r_{min}, δ_θ, Δ_ρ, L_1) and the cluster itself ($R_1(\theta_{12})$ and $R_2(\theta_{12})$). We can therefore choose the following expression for BVE:

$$\text{BVE}(S_1,S_2,C) = \frac{2\delta_\theta J_{12}\sqrt{\Delta_\rho}}{L_1 r_{min}} T(\alpha,\beta,\theta_{12})[R_1(\theta_{12}) + R_2(\theta_{12})]$$

The only information stored for each cluster is thus the two directional functions R_1 and R_2 (See example on Fig.3). Although this approach gives acceptable error control on the form factor calculation, we think that better bounds can be obtained by considering a *feature based* error estimation method [6].

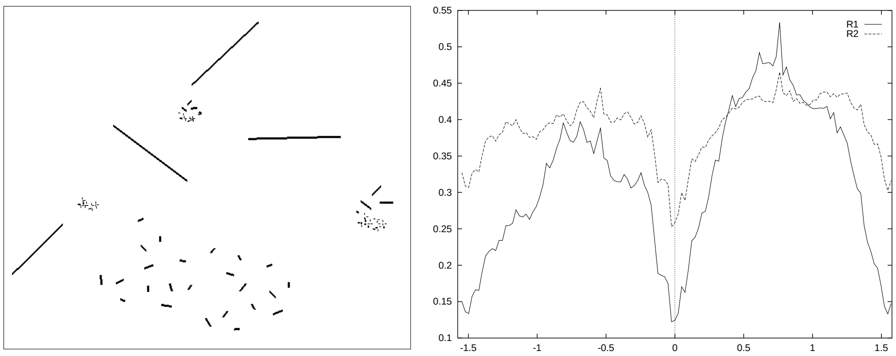

Fig. 3. A cluster and related functions R_1 and R_2

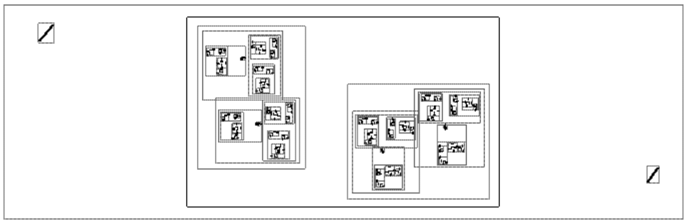

Fig. 4. Experimental configuration

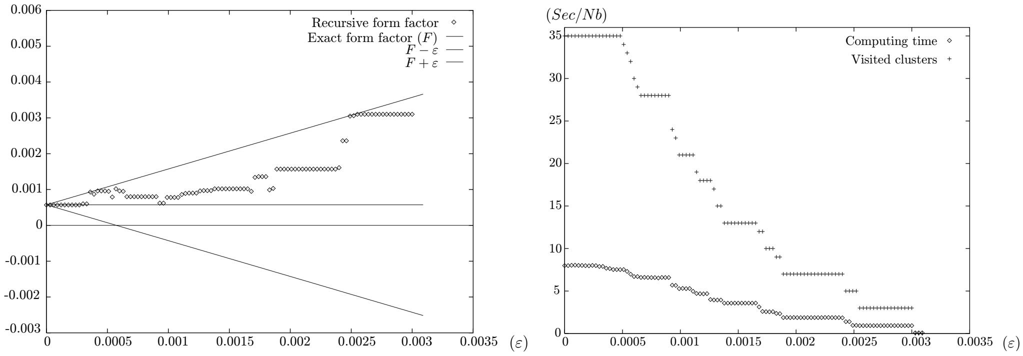

Fig. 5. Cpu computing time and number of used clusters

8.2 Example of form factor error control in dimension 2

We show here an example of form factor error control using multi-scale visibility information between two segments S_1 and S_2, occluded by a cluster hierarchy of root C (Fig.4). We have made the control error bound go from 0 up to 0.0035, which is the value above which the recursive form factor algorithm does not do any refinement.

As imposed by the algorithm, the result remains in the interval $[F-\varepsilon, F+\varepsilon]$ for any ε value. When ε goes to 0, the cpu computing time and the number of used clusters grow regularly, which shows that this algorithm allows a time/accuracy control of the form factor.

Conclusions & Future directions

In this paper we have suggested some avenues for the development of efficient and accurate algorithms to evaluate form factors using approximate visibility.

We have defined a general framework for error-driven visibility computation with a hierarchy of clusters, using an error-bounding function for each cluster. Such functions can be derived from mathematical bounds on the form factor error, provided the necessary information is stored at each cluster. The recombination problem must be addressed to cope with subcluster correlation and maintain the error bounds as we step through the hierarchy.

Results were presented from our first implementation, limited to 2D in order to demonstrate the applicability of the concepts. We are currently implementing a 3D version, for which we have derived adequate error bounds. It should be noted that the behaviour of correlation functions in 3D is even more favorable than in 2D: intuitively, this is because there are “more” directions for which the correlation is low.

The results show that we can effectively generate various approximations of the form factor by selecting different levels of the cluster hierarchy to compute visibility. It is especially remarkable that the approximate calculation actually converges to the true form factor value when the requested bound goes to zero. This is in contrast to previous approaches [5] where approximate visibility calculations never gave way to accurate, surface-based computations. Computing times for our examples are still quite large. In this study we focused on visibility approximation and always compute the form factor integral with a very expensive and precise quadrature, every time we reach a leaf of the hierarchy.

Future work includes the use of a "feature based" error estimation, which produces tighter error bounds on the form factor estimation error. The coupling of the multi-resolution visibility calculation method with the hierarchical refinement criterion is also a subject of great interest. In this paper we considered a fixed pair of surfaces without mentioning the possible subdivision of one or the other. In a real application form factors are rarely computed in isolation from the refinement process. It is very plausible that the insight gained using the error-driven analysis will help in defining more efficient, error-bounded subdivision criteria.

References

1. Michael F. Cohen, Shenchang Eric Chen, John R. Wallace, and Donald P. Greenberg. A progressive refinement approach to fast radiosity image generation. *Computer Graphics*, 22(4):75–84, August 1988. Proceedings SIGGRAPH '88.
2. Pat Hanrahan, David Saltzman, and Larry Aupperle. A rapid hierarchical radiosity algorithm. *Computer Graphics*, 25(4):197–206, August 1991. Proceedings SIGGRAPH '91 in Las Vegas (USA).
3. Luis A. Santalo. *Integral Geometry and Geometric Probability*, volume 1 of *Encyclopedia of Mathematics and its applications*. Addison-Wesley, 1976.
4. Peter Schröder and Wim Sweldens. Spherical wavelets: Efficiently representing functions on the sphere. In *Computer Graphics Proceedings:* SIGGRAPH '95, pages 161–172. ACM SIGGRAPH, New York, August 1995.
5. François Sillion. A unified hierarchical algorithm for global illumination with scattering volumes and object clusters. *IEEE Transactions on Visualization and Computer Graphics*, 1(3), September 1995.
6. François Sillion and George Drettakis. Feature-based control of visibility error: A multiresolution clustering algorithm for global illumination. In *Proceedings* SIGGRAPH '95, pages 145–152. ACM SIGGRAPH, New York, August 1995.
7. François Sillion, George Drettakis, and Cyril Soler. A clustering algorithm for radiance calculation in general environments. In P.M. Hanrahan and W. Purgathofer, editors, *Rendering Techniques '95*. Springer Verlag, Wien, August 1995. Proceedings of Sixth Eurographics Workshop on Rendering (Dublin, Ireland, June 1995).
8. François Sillion and Claude Puech. *Radiosity and Global Illumination*. Morgan Kaufmann publishers, San Francisco, 1994.
9. Brian Smits, James Arvo, and Donald P. Greenberg. A clustering algorithm for radiosity in complex environments. In *Computer Graphics Proceedings:* SIGGRAPH '94, pages 435–442. ACM SIGGRAPH, New York, July 1994.
10. Cyril Soler. Représentation multi-échelles d'informations de visibilité, June 1995. Mémoire de DEA de Mathématiques Appliquées. Univ. Joseph Fourier. Grenoble.
11. Seth J. Teller and Patrick M. Hanrahan. Global visibility algorithms for illumination computations. In *Computer Graphics Proceedings, Annual Conference Series:* SIGGRAPH '93 (Anaheim, CA, USA), pages 239–246. ACM SIGGRAPH, New York, August 1993.