Interactive Walk-Through Using Particle Tracing

S. N. Pattanaik and Kadi Bouatouch
IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France

Abstract

In this paper, we present an efficient particle tracing method for interactive walk-through in a large environment. The efficiency is derived using the concept of *importance* similar to that used in importance driven radiosity. We propose the importance driven particle tracing as an add-on module to the standard particle tracing process, which is activated on demand as the user walks-through from one region to another in the environment. This module derives the *importance* information from the normal particle tracing process.
Key Words: **Radiance equation, Potential equation, Adjointness, Random Walk, Function Approximation, and Importance**.

1 Introduction

Particle tracing is a stochastic solution method for global illumination computation. In this method particles are emitted from the light source and propagated in the environment according to the light interaction property of the environment. The illumination is estimated by tracking the particles during their propagation. The tracking of a particle amounts to carrying out the ray/environment intersection operation which is quite an well understood operation. This makes particle tracing an extremely simple method for implementation.

The very first implementation of particle tracing [1] originated as the simulation of the particle model of light and using it view independent illumination of the environment was estimated by fitting piece-wise constant functions over the surfaces of the environment. Later, this simulation was related to the random walk solution of the potential equation [2, 3]. Using this relationship, it has been possible to derive a number of improvements to the particle tracing method. These various improvements are [3, 4]: direct estimation of higher order illumination function, introduction of better estimators for variance reduction, importance sampling and multi-resolution reconstruction for noise-free visual reconstruction of the illumination function etc.. All these improvements make particle tracing a powerful illumination computation method.

As a further extension to this particle tracing work, in this paper, we propose a simple and efficient particle tracing method for interactive walk-through. Here we use the concept similar to *importance* introduced by Smits *et al* in [5]. This *importance* has already been shown to be a view-independent potential function[6]. In particle tracing, one is carrying out the solution of the potential equation. So the *importance* value is already available in some form or other during the normal particle tracing operation. No extra effort is needed to carry out its computation. It is just a matter of extracting and using that information in a way such as to improve the view specific computation results during the walk-through. Though a general framework of an importance driven particle tracing has been proposed in [2, 4], so far, no efficient implementation is available. In the following parts of this paper, we propose and put in to work an extremely simple and efficient implementation of importance driven particle tracing. At its current form, it only improves the selection of the light source depending on their *importance* towards the view. We propose the importance driven particle tracing as an add-on module, which works in mutual co-operation with a normal particle tracer. The importance driven module, draws the necessary *importance* information from the normal module, computes and contributes to the estimates derived by the normal module. Both the modules, refine the illumination estimation progressively. But they differ in the fact that importance driven one refines the estimates for a small subset of the scene much faster than that would be possible by the normal one. The normal module refines for the whole environment uniformly.

The organisation of this article is as follows. For the completeness sake we first summarise the concepts of function approximation, adjoint equation and particle tracing. Then we proceed to derive the basis for importance driven particle tracing and follow it by an implementation demonstrating the efficiency.

2 Particle tracing and Higher Order Approximation of Illumination

The illumination function, L, defined over a surface k of an environment can be approximated as follows:

$$L^k(\bar{x},\Theta_{\bar{x}}) \approx \hat{L}^k(\bar{x},\Theta_{\bar{x}}) = \sum_{m=1}^{M} a_m \mathcal{N}_m(\bar{x},\Theta_{\bar{x}}) \tag{1}$$

where $\Theta_{\bar{x}}$ is the outgoing direction over the point $\bar{x}$ of surface k, $\mathcal{N}_m(\bar{x},\Theta_{\bar{x}})$'s are the basis functions used for approximation and a^k_m are the unknown coefficients of approximation. Using Galerkin method, one can estimate a^k_m's by solving a set of linear equations $\mathbf{Ka} = \mathbf{f}$ where the $K^k_{l,m}$'s, the elements of the matrix $\mathbf{K}$, are

$$K^k_{l,m} = \int_{\text{surface}_k} \int_{\Omega_{\bar{x}}} \mathcal{N}_l(\bar{x},\Theta_{\bar{x}}) \mathcal{N}_m(\bar{x},\Theta_{\bar{x}}) d\omega dA_{\bar{x}}$$

and f^k_m's, the elements of the matrix $\mathbf{f}$, are

$$f^k_m = \int_{\text{surface}_k} \int_{\Omega_{\bar{x}}} L^k(\bar{x},\Theta_{\bar{x}}) \mathcal{N}_m(\bar{x},\Theta_{\bar{x}}) d\omega dA_{\bar{x}}$$

Computation of the approximation coefficients, a^k_m, require the evaluation of $K^k_{l,m}$ and f^k_m. The evaluation of $K^k_{l,m}$ is independent of the illumination function and hence it does not pose a problem. The main problem lies in the evaluation of f^k_m.

In the discussion above, all the functions and domain of surface integration were over the surface k. We can however generalise and extend the domain of integration to the whole environment simply by assuming that the basis functions are sets of global functions having compact support and the extents of support of these functions are at best limited to the surface k. This modification makes the basis functions different for different surfaces, and to distinguish them we add the superscript k to each of them. Thus, with this modification the expression of f^k_m takes the following form.

$$f^k_m = \int_{\text{Env}} \int_{\Omega_{\bar{x}}} L(\bar{x},\Theta_{\bar{x}}) \mathcal{N}^k_m(\bar{x},\Theta_{\bar{x}}) d\omega dA_{\bar{x}} \tag{2}$$

Note that L is no more restricted to the surface k, and this has been indicated in this latter formulation by removing the superscript k from L^k.

Using the potential function[1] $\mathcal{P}$, the adjoint of radiance function, we can write a dual equation to that given in 2 as:

$$f^k_m = \int_{\text{Env}} \int_{\Omega_{\bar{x}}} \mathcal{P}^{(k,m)}(\bar{x},\Theta_{\bar{x}}) L^e(\bar{x},\Theta_{\bar{x}}) d\omega dA_{\bar{x}} \tag{3}$$

where $L^e(\bar{x},\Theta_{\bar{x}})$ is the global emission function, $\mathcal{P}^{(k,m)}$ is given by the expression

$$\mathcal{P}^{(k,m)}(\bar{x},\Theta_{\bar{x}}) = \mathcal{N}^k_m(\bar{x},\Theta_{\bar{x}}) + \cos\theta_{\bar{x}} \int_{\Omega_{\bar{y}}} f_r(\bar{y},\Theta_{\bar{x}},\Theta_{\bar{y}}) \mathcal{P}^{(k,m)}(\bar{y},\Theta_{\bar{y}}) d\omega \tag{4}$$

[1] For the details on the potential function see [2, 4].

and $\bar{y}$ is the nearest surface point from $\bar{x}$ along the direction $\Theta_{\bar{x}}$.
The superscript (k,m) in the potential function $\mathcal{P}$ indicates that it is the potential of any $(\bar{x},\Theta_{\bar{x}})$ towards the m-th basis function of the k-th surface.

2.1 Illumination Function Estimation

As explained above, the computation of an approximation of illumination function, requires the estimation of values of f_m^k. Here we describe a Monte Carlo/random-walk method for their estimation. This estimation involves sampling some probability distribution function from equation 3 to initiate a random walk followed by sampling of another distribution function from the integral component of equation 4 for continuing this walk. We first look at the equation 4. It has f_r, the bidirectional reflectance distribution function (*brdf*), as a known function in the integral term and can be used to generate new directions for the random walk. From the definition of *brdf* it is known that

$$\int_\Omega f_r(y,.,.)\cos\theta d\omega = \rho_y < 1 \qquad \text{where } \rho_y \text{ is the reflectivity at } y.$$

So we rewrite equation 4 to bring in the cosine term along with $f_r()$ inside the integral.

$$\begin{aligned} \frac{\mathcal{P}^{(k,m)}(\bar{x},\Theta_{\bar{x}})}{\cos\theta_{\bar{x}}} &= \frac{\mathcal{N}_m^k(\bar{x},\Theta_{\bar{x}})}{\cos\theta_{\bar{x}}} + \int_{\Omega_{\bar{y}}} f_r(\bar{y},\Theta_{\bar{x}},\Theta_{\bar{y}})\cos\theta_{\bar{y}}\frac{\mathcal{P}^{(k,m)}(\bar{y},\Theta_{\bar{y}})}{\cos\theta_{\bar{y}}}d\omega \qquad \text{or} \\ \mathcal{P}'^{(k,m)}(\bar{x},\Theta_{\bar{x}}) &= \frac{\mathcal{N}_m^k(\bar{x},\Theta_{\bar{x}})}{\cos\theta_{\bar{x}}} + \int_{\Omega_{\bar{y}}} f_r(\bar{y},\Theta_{\bar{x}},\Theta_{\bar{y}})\cos\theta_{\bar{y}}\mathcal{P}'^{(k,m)}(\bar{y},\Theta_{\bar{y}})d\omega \qquad (5) \\ &\text{where} \quad \mathcal{P}'^{(k,m)}(\bar{x},\Theta_{\bar{x}}) = \frac{\mathcal{P}^{(k,m)}(\bar{x},\Theta_{\bar{x}})}{\cos\theta_{\bar{x}}}. \end{aligned}$$

Now we look at equation 3. We accommodate $\mathcal{P}'^{(k,m)}$ in the equation as:

$$f_m^k = \int_{\text{Env}}\int_{\Omega_{\bar{x}}} \mathcal{P}'^{(k,m)}(\bar{x},\Theta_{\bar{x}})\cos\theta_{\bar{x}}L^e(\bar{x},\Theta_{\bar{x}})d\omega dA_{\bar{x}} \qquad (6)$$

In this equation[2], $L^e(\bar{x},\Theta_{\bar{x}})\cos\theta_{\bar{x}}$ is a known function and can be used to sample the starting position and direction for a random walk.

In a real life environment, the emission function, L^e, is non-zero only over the emitting surfaces. As the emitting surfaces make a very small fraction of the total surfaces, we shall write equation 6 as a sum of integrals as follows:

$$\begin{aligned} f_m^k &= \sum_{l=1}^{N_{src}} \int_{\text{emitter}_l}\int_{\Omega_{\bar{x}}} \mathcal{P}'^{(k,m)}(\bar{x},\Theta_{\bar{x}})\cos\theta_{\bar{x}}L_l^e(\bar{x},\Theta_{\bar{x}})d\omega dA_{\bar{x}} \\ &= \sum_{l=1}^{N_{src}} \Phi_l \int_{\text{emitter}_l}\int_{\Omega_{\bar{x}}} \mathcal{P}'^{(k,m)}(\bar{x},\Theta_{\bar{x}})\frac{L_l^e(\bar{x},\Theta_{\bar{x}})\cos\theta_{\bar{x}}}{\Phi_l}d\omega dA_{\bar{x}} \text{ where } \Phi_l = \int_{\text{emitter}_l}\int_{\Omega_{\bar{x}}} L_l^e(\bar{x},\Theta_{\bar{x}})\cos\theta_{\bar{x}}d\omega dA_{\bar{x}} \\ &= \Phi_{\text{Env}}\sum_{l=1}^{N_{src}} \frac{\Phi_l}{\Phi_{\text{Env}}} \int_{\text{emitter}_l}\int_{\Omega_{\bar{x}}} \mathcal{P}'^{(k,m)}(\bar{x},\Theta_{\bar{x}})\frac{L_l^e(\bar{x},\Theta_{\bar{x}})\cos\theta_{\bar{x}}}{\Phi_l}d\omega dA_{\bar{x}} \text{ where } \Phi_{\text{Env}} = \sum_l \Phi_l \qquad (7) \end{aligned}$$

N_{src}, in the above equations is the number of light sources, L_l^e is the emittance function of source l, Φ_l is the total emission power of source l and Φ_{Env} is the total emission power of the whole environment.

[2]Note that [2, 3] used equation 6 as the expression for potential equation.

Applying random walk estimation technique to equation pair (7) and (5) we can estimate f_m^k as

$$f_m^k \approx \frac{\Phi_{\mathrm{Env}}}{n} \sum_{i=1}^{n} \left[\sum_{j=0}^{m_i} \frac{\mathcal{N}_m^k(\bar{x}_{i,j}, \Theta_{\bar{x}_{i,j}})}{\cos\theta_{\bar{x}_{i,j}}} \right] \tag{8}$$

where the random samples $(\bar{x}_{i,0}, \Theta_{\bar{x}_{i,0}})$ are generated by : first randomly deciding on a light source l by sampling discrete probability distribution function

$$\left\{ \frac{\Phi_1}{\Phi_{\mathrm{Env}}}, \ldots, \frac{\Phi_{N_{src}}}{\Phi_{\mathrm{Env}}} \right\} \tag{9}$$

and then by drawing samples $(\bar{x}_{i,0}, \Theta_{\bar{x}_{i,0}})$ from the selected light source l by sampling the continuous distribution function $\frac{L_l^e(\bar{x}, \Theta_{\bar{x}}) \cos\theta_{\bar{x}}}{\Phi_l}$. For each i, sample $(\bar{x}_{i,j}, \Theta_{\bar{x}_{i,j}})$ is generated from the previous sample $(\bar{x}_{i,j-1}, \Theta_{\bar{x}_{i,j-1}})$ by equating the nearest surface point $\bar{y}$, from $\bar{x}_{i,j-1}$ along the direction $\Theta_{\bar{x}_{i,j-1}}$, to $\bar{x}_{i,j}$ and sampling $f_r(\bar{y}, \Theta_{\bar{x}_{i,j-1}}, \Theta_{\bar{y}})$ for getting the direction $\Theta_{\bar{x}_{i,j}}$.

Thus we have derived a particle tracing algorithm for estimating f_m^k of a illumination function belonging to a surface k. We initialise f_m^k to 0, initiate a random walk from a random position over an emitting surface, along a random direction and continue the walk by drawing random directions from the surfaces along the path. If the surface encountered along the path is k then increment its f_m^k estimate by $\frac{\mathcal{N}_m^k(\bar{x}_{i,j}, \Theta_{\bar{x}_{i,j}})}{\cos\theta_{\bar{x}_{i,j}}}$. If we carry out a large number of such random walks we will arrive at a good estimate.

The important aspect of this technique is that at each time a random walk originates by sampling the emission function and continues its walk by sampling the reflection function. Thus the walk at no stage has any preference towards any surface. So **the same random walk can be used for the estimation of all f_m^k's corresponding to the illumination functions over all the surfaces, and thus for the estimation of the approximation of the illumination function over all surfaces**. Figure 1 shows the resulting algorithm.

3 Importance Driven Particle Tracing

In the algorithm discussed above we have generated particles by sampling the light source according to their emitting strength. Thus, if we have a number of light sources with equal emission power, then statistically the same number of particles will be emitted from each of the light sources. Generating particles in this fashion is essential so as to estimate the illumination of the whole environment. However, this approach will take a very long time to compute the illumination for a virtual walk-through inside a large environment. Other than solving for the whole environment in offline and only creating the image during the walk-through time from these solved illumination information, this method cannot be used for any real-time application. We could possibly modify the particle tracing algorithm to better.

In a large environment it is generally not true that every light source contributes towards the illumination of every surface. For any given part of the environment, only a small fraction of the light sources makes actual contribution. Even among these few there is likely to be a variation in the amount of contribution. In such case, if one is viewing the neighbourhood of a small part of the environment, then it will be worthwhile to consider only those fraction of the light sources having significant contribution for the emission of the particle and also it will be better to give emphasis to the each light source according to its contribution to that small part of the environment. It is possible to incorporate this idea into the particle tracing framework by introducing a concept of illuminating capacity ($\mathcal{I}_l$) of each light source l towards a region. This is similar to what one calls *importance* in *importance driven radiosity* [5]. This capacity is actually the potential of the light source towards illuminating the specified region. However, here we will not be very rigorous about its expression. If we can specify our region of interest, say the one currently on the view during a walk-through, and if we can compute an approximate estimate of this capacity towards this region of interest, then we can generate a modified probability distribution of emission by taking in to consideration both

main
begin

- Initialise;
- Normal Particle Trace;
- Compute Coefficients;

end main.

Initialise
begin

- For each surface k initialise its f_m^k's to zero.
- Compute the discrete power distribution (equation 9).

end Initialise;
Normal Particle Trace
begin

- do
  - /*... Emit a particle ...*/
    - Choose an emitter by sampling the discrete power distribution.
    - Choose the position and direction $(\bar{x}_{i,0}, \Theta_{\bar{x}_{i,0}})$ for a particle on the emitter surface by sampling its emitted power distribution function.
    - Increment all f_m^k's corresponding to the emitting surface k by $\frac{\mathcal{N}_m^k(\bar{x}_{i,0}, \Theta_{\bar{x}_{i,0}})}{\cos\theta_{\bar{x}_{i,0}}}$.
  - /*... Track the particle until the particle is absorbed... /*.
    do
    - Find the nearest surface k and the surface point y along the particle path.
    - Choose the type of interaction i.e. *absorption* or *reflection*, by sampling the discrete interaction distribution function $\{\rho_{\bar{y}}, 1 - \rho_{\bar{y}}\}$ where $\rho_{\bar{y}}$ is the reflection coefficient at $\bar{y}$.
    - If the interaction is *reflection*
      - Set $\bar{x}_{i,j} = y$.
      - Assign a reflected direction, $\Theta_{\bar{x}_{i,j}}$ to the particle by sampling $\frac{f_r \cos\theta_{\bar{y}}}{\rho_{\bar{y}}}$.
      - Increment all f_m^k terms corresponding to the reflecting surface k by $\frac{\mathcal{N}_m^k(\bar{x}_{i,j}, \Theta_{\bar{x}_{i,j}})}{\cos\theta_{\bar{x}_{i,j}}}$.

    while(interaction is not *absorption*);

  until(a large number of particles are emitted);

end Normal Particle Trace;
Compute Coefficients
begin

- For each surface k do
  - Evaluate all $K_{l,m}^k$'s and compute $\left(\mathbf{K}^k\right)^{-1}$.
  - Evaluate $\mathbf{a}^k$, the coefficients of illumination function from computed $K_{l,m}^k$'s and estimated f_m^k's as $\mathbf{a}^k = \frac{\Phi_{\mathrm{Env}}}{n}\left(\mathbf{K}^k\right)^{-1}\mathbf{f}^k$, where n is the total number of emitted particles.

  end for;

end Compute Coefficients;

Figure 1: Normal Particle Tracing Algorithm

the illuminating capacities and the power of the light sources. Sampling of this modified distribution will certainly generate a particle distribution favourable to the illumination estimation of the specified region. However, this modification must to be compensated so as to keep the energy equilibrium in the environment. The discrete power distribution function was discussed in connection with the solution of equation 7. So we shall look at the equation 7 and see how the above proposed modification can be incorporated.

$$\begin{aligned} f_m^k &= \sum_{l=1}^{N_{src}} \Phi_l \int_{\text{emitter}_l} \int_{\Omega_{\bar{x}}} \mathcal{P}'^{(k,m)}(\bar{x}, \Theta_{\bar{x}}) \frac{L_l^e(\bar{x}, \Theta_{\bar{x}}) \cos\theta_{\bar{x}}}{\Phi_l} d\omega dA_{\bar{x}} \\ &= \sum_{l=1}^{N_{src}} \mathcal{I}_l \Phi_l \int_{\text{emitter}_l} \int_{\Omega_{\bar{x}}} \frac{\mathcal{P}'^{(k,m)}(\bar{x}, \Theta_{\bar{x}})}{\mathcal{I}_l} \frac{L_l^e(\bar{x}, \Theta_{\bar{x}}) \cos\theta_{\bar{x}}}{\Phi_l} d\omega dA_{\bar{x}} \\ &= \Phi'_{\text{Env}} \sum_{l=1}^{N_{src}} \frac{\mathcal{I}_l \Phi_l}{\Phi'_{\text{Env}}} \int_{\text{emitter}_l} \int_{\Omega_{\bar{x}}} \frac{\mathcal{P}'^{(k,m)}(\bar{x}, \Theta_{\bar{x}})}{\mathcal{I}_l} \frac{L_l^e(\bar{x}, \Theta_{\bar{x}}) \cos\theta_{\bar{x}}}{\Phi_l} d\omega dA_{\bar{x}} \quad \text{where } \Phi'_{\text{Env}} = \sum \mathcal{I}_l \Phi_l \end{aligned} \tag{10}$$

The only difference this modification makes is : instead of estimating the unknown $\mathcal{P}'^{(k,m)}(\bar{x}, \Theta_{\bar{x}})$ we are faced with the estimation of $\frac{\mathcal{P}'^{(k,m)}(\bar{x}, \Theta_{\bar{x}})}{\mathcal{I}_l}$. But it will pose no problem if the term in the denominator $\mathcal{I}_l$ is known. Thus the importance driven estimate of f_m^k will be:

$$f_m^k \approx \frac{\Phi'_{\text{Env}}}{n} \sum_{i=1}^{n} \frac{1}{\mathcal{I}_{l_i}} \sum_{j=0}^{m_i} \frac{\mathcal{N}_m^k(\bar{x}_{i,j}, \Theta_{\bar{x}_{i,j}})}{\cos\theta_{\bar{x}_{i,j}}} \tag{11}$$

where l_i is the index of the light source from which the i-th particle originated.
Using this estimate amounts to carrying out a modification of the earlier algorithms in that, each particle now carries weight equivalent to $\frac{1}{\mathcal{I}_l}$ where l is the light source from which the particle originated. Thus we can use the same algorithm given in figure 1 provided we carry out the following:

- For the specified region of the environment, compute the approximate illuminating capacity $\mathcal{I}_l$ for each light source.
- Compute the modified power distribution, *i.e.*

$$\left\{ \frac{\mathcal{I}_1 \Phi_1}{\Phi'_{\text{Env}}}, \ldots, \frac{\mathcal{I}_{N_{src}} \Phi_{N_{src}}}{\Phi'_{\text{Env}}} \right\} \tag{12}$$

- Sample the modified power distribution for particle emission.
- Choose the appropriate increments (given in equation 11) to compensate for the modified power distribution.

We are now ready to write the importance driven particle tracing algorithm which is as follows :

4 Interactive Walk-Through Architecture

We provide here the outline of our proposed interactive walk-through system based on particle tracing. Figure 3 provides a rough view of the system. We assume that there is a main process carrying out normal particle tracing continuously. In addition to this process we have another high-priority process, could be interrupt driven, which becomes active on user interaction. When invoked this process derives *importance* from the information gained by the main process and traces a reasonable number of particles to provide an improved illumination estimate for the region of interest. The actual number of particles that are needed to be traced by this importance driven module will depend on the current estimation status.

Importance Driven Particle Trace
begin

- For the given view, compute $\mathcal{I}_l$'s corresponding to each light source l.
- Compute a modified discrete power distribution (equation 12).
- for N times do
  - /*... Emit a Particle... */
    1. Choose a light source, p, by sampling the discrete distribution given in equation 12;
    2. Choose a position and direction for a particle on the light source p as done in figure 1;
    3. Increment all f_m^p's of surface p by $\frac{1}{\mathcal{I}_p} \frac{\mathcal{N}_m^p(\bar{x}_{i,0}, \Theta_{\bar{x}_{i,0}})}{\cos\theta_{\bar{x}_{i,0}}}$;
  - /*... Track the particle until the particle is absorbed... */
    do
    1. Find the nearest surface k and surface point y along the particle path.
    2. Choose the type of interaction as done in figure 1;
    3. If the interaction is *reflection*
       (a) Set $\bar{x}_{i,j} = y$.
       (b) Assign a reflected direction, $\Theta_{\bar{x}_{i,j}}$ as done in figure 1;
       (c) Increment all f_m^k terms corresponding to the surface k by $\frac{1}{\mathcal{I}_p} \frac{\mathcal{N}_m^k(\bar{x}_{i,j}, \Theta_{\bar{x}_{i,j}})}{\cos\theta_{\bar{x}_{i,j}}}$

    while(interaction is not *absorption*);

  end for;

end Importance Driven Particle Trace;

Figure 2: Importance Module.

Intialise

New view

Importance Module

Normal Particle Trace
(View Independent Computation)

Importance Information

Importance Driven Particle Trace
(View Dependent Computation)

Compute Coefficients

Visualisation

View Specifications

file=sketch.pstex

Figure 3: Interactive Walk-through using Particle Tracing.

4.1 Computation of $\mathcal{I}_l$'s

The implementation of above discussed idea requires an estimation of the illuminating capacity $\mathcal{I}$ for each light source, towards the region on which the view is focussed at any particular time of the walk through. The calculation procedure must take into account that the viewed region is most likely to change continuously during the walk-through. That means we can not make any preassumptions about the region of interest. Also, the strategy must allow immediate extraction of $\mathcal{I}_l$'s so as to be able to devote maximum possible effort towards the importance driven particle tracing process. Towards this goal, we propose of a very simple method of computing an approximate illuminating capacity associated with each light source towards the region of interest.

We assume that the environment could be enclosed inside a rectangular box and could be subdivided by uniform grids parallel to the bounding plane. Given this 3D grid structure, at any time in the walk-through process, we can quickly find a rough area of interest as a collection of integral number of voxels by intersecting the view volume with the 3D grid. Once we decide that a subset of voxels make up our region of interest, then we can have an extremely simple way of capturing and extracting the illuminating capacity $\mathcal{I}$ of each light source towards this region of interest.

We shall now associate with each voxel v an counter array, $\mathcal{C}_v$ containing as many elements as there are the light sources in the environment. During the normal particle tracing operation, every time a particle, originating at a light source l, enters a voxel v during its propagation, the item corresponding to the light source l of the counter array $\mathcal{C}_v$, *i.e.* $\mathcal{C}_v[l]$, is incremented. After a reasonable number of normal particle traces, $\mathcal{C}_v[l]$ will give an approximate illuminating capacity of a light source l towards the voxel v. From the $\mathcal{C}_v[l]$'s values corresponding to all the voxels, we can estimate of illuminating capacity, $\mathcal{I}_l$'s, towards the region of interest as:

$$\mathcal{I}_l = \sum_{v \in \text{view\_volume}} \mathcal{C}_v[l]$$

This latter evaluation is hardly any effort. Of course, it is essential that this evaluation step is carried out at a time when one is certain that the normal particle tracing process has sampled each and every light sources of the environment. Thus one must take in to account an initial latency period.

The incorporation of the additional grid structure and tracking the entry of particle inside a voxel for the capture of $\mathcal{C}$ is not at all an additional problem. For faster trace of particles we are already using a spatially enumerated 3D grid structure and use of such structures are quite common in ray tracing. And also for faster ray-surface interaction we are carrying out the grid traversal. Thus we have provided an extremely easy and fast method of acquiring $\mathcal{C}_v$'s for estimation of $\mathcal{I}$.

5 Implementation

We have implemented the above mentioned idea for the radiosity function estimation inside an environment containing 100 interconnected rooms with 100 light sources (figure 4). We have simulated the interrupt driven process as follows: we added an option to the particle tracer program to accept the *importance* data associated with each voxel and the number of voxel(s) intersecting the hypothetical view volume. This option modifies the emission probability associated with the light sources according to the $\mathcal{I}$ computed from $\mathcal{C}$ of the voxels and a hypothetical view volume. We have captured $\mathcal{C}$'s by carrying out an independent normal run such that at least 1000 particles are emitted from each light source.

For approximation of radiosity functions we have chosen direction independent spatial basis functions for $\mathcal{N}$. We have used the scaling functions of Multi-wavelets [7] as the necessary basis function. The reason of chosing of wavelets in our particle tracing work has been given in [4]. The choice of Multi-wavelets in this implementation comes from our familiarity with this basis. However, Multi-wavelets are not essential to the method discussed here. We construct our basis function from the 1D multi-wavelet basis functions as follows.

$$\mathcal{N}(\bar{x}, \Theta_{\bar{x}}) = \mathcal{S}(u)\mathcal{R}(v)$$

where $\mathcal{S}$ and $\mathcal{R}$ are respectively the 1D multiwavelet scaling functions with variable u and v.

Evaluation of $\mathbf{K^k}$:

Because of the orthogonality of the multi-wavelet basis functions, $K^k_{l,m}$ vanish for $l \neq m$. It leaves us to compute only $K^k_{m,m}$ values[3]

$$\begin{aligned} K^k_{m,m} &= \int_{\text{surface}_k} \int_{\Omega_x} N^k_m(\bar{x}, \Theta_{\bar{x}}) N^k_m(\bar{x}, \Theta_{\bar{x}}) d\omega dA_{\bar{x}} = \int_{\text{surface}_k} N^k_m(\bar{x}, \Theta_{\bar{x}}) N^k_m(\bar{x}, \Theta_{\bar{x}}) \left[\int_{\Omega_x} d\omega \right] dA_{\bar{x}} \\ &= 2\pi \int_{\text{surface}_k} N^k_m(\bar{x}, \Theta_{\bar{x}}) N^k_m(\bar{x}, \Theta_{\bar{x}}) dA_{\bar{x}} = 2\pi \int_0^1 \int_0^1 \mathcal{S}_m(u) \mathcal{R}_m(v) \mathcal{S}_m(u) \mathcal{R}_m(v) \left\| \frac{d\bar{x}}{du} \times \frac{d\bar{x}}{dv} \right\| du\, dv \end{aligned}$$

Because of the polynomial nature of the multi-wavelet basis functions it is possible to carry out a closed form evaluation of this integral over a bi-parametric polynomial surface. Otherwise it will be necessary to use some numerical quadrature techniques.

5.1 Results

To demonstrate the achieved improvement, we show two figures 5 and 6. Figure 5 shows the rendering based on the illumination estimation obtained from normal particle tracing operation with 20,000 particles and figure 6 shows the rendering based on the illumination estimation obtained from the importance driven particle tracing with the same number of particles. The *importance* has been based on a simulated situation where the view is focussed on the room at the left most corner. In each of the figures, the surface illumination has been captured by using lowest resolution multiwavelet with vanishing moment 4. Figures 7 and 8 are respectively the closeup views of figures 5 and 6.

6 Discussion

It is essential that a noise free illumination function is reconstructed from the estimated results before carrying out any viewing operation. The particle tracing method using Multi-wavelets, provides techniques [4] such as: multi-resolution wavelet reconstruction (from the coefficients captured at higher resolution), projection to lower dimension (if the number of particles contributing to the estimates are statistically low) for carrying out this reconstruction. However, it still does not remove the stochastic noise with surface elements which do not even have enough number of outgoing particles to produce a reasonable estimation of the lowest order illumination function (*i.e.* constant illumination). We propose a post simulation inspection and to resort to one level deterministic/stochastic gathering on these uncertain surfaces from the bright and already estimated surfaces. This latter step of one level gathering can be justified by saying that the surfaces are small enough to have any important illumination contribution over the environment and also small enough to worry about their illumination distribution other than an average constant illumination.

7 Conclusion

The method discussed in this paper suggests improvements to particle tracing for view dependent illumination computation by bringing changes to the particle emission pattern. There are still a lot of scopes to improve the efficiency of an importance driven particle tracing process by modifying the particle-medium interaction. We are working on similar simple and efficient algorithm and data-structure combination to bring out these improvements.

[3]It must be noted that in Multi-wavelet basis each of these coefficients $K^k_{m,m}$ represents a matrix of $M^2 \times M^2$ terms, where M is the number of vanishing moments of the basis function. So in fact, one would be computing $K^k_{m,m}[i,j] = \int_{\text{surface}_k} \int_{\Omega_x} N^k_m[i](\bar{x}, \Theta_{\bar{x}}) N^k_m[j](\bar{x}, \Theta_{\bar{x}}) d\omega dA_{\bar{x}}$.

8 Acknowledgements

We have made use of a scene whose geometry was made available by Peter Shirley. We are thankful to him.

References

[1] Sumanta N. Pattanaik and S. P. Mudur. Computation of global illumination by monte carlo simulation of the particle model of light. In *Proceedings of the Third Eurographics Workshop on Rendering*, pages 71–83, Bristol, UK, May 1992.

[2] Sumanta N. Pattanaik. *Computational Methods for Global Illumination and Visualisation of Complex 3D Environments.* PhD thesis, Feb 1993.

[3] Sumanta N. Pattanaik and S. P. Mudur. Adjoint equations and random walks for illumination computation. *ACM Transactions on Graphics*, 14(1):77–102, January 1995.

[4] Sumanta N. Pattanaik and Kadi Bouatouch. Adjoint equations and particle tracing for global illumination. Technical report, (http://www.irisa.fr/EXTERNE/bibli/pi/pi95.html/PI-903) IRISA, Rennes, France, Jan 1995.

[5] Brian E. Smits, James R. Arvo, and David H. Salesin. An importance driven radiosity algorithm. *Computer Graphics (SIGGRAPH '92 Proceedings)*, 26(2):273–282, July 1992.

[6] Sumanta N. Pattanaik and S. P. Mudur. The potential equation and importance in illumination computations. *Computer Graphics Forum*, 12(2):131–136, 1993.

[7] Steven Gortler, Peter Schroder, Michel F. Cohen, and Pat Hanrahan. Wavelet radiosity. *Computer Graphics (SIGGRAPH '93 Proceedings)*, 27(4):221–230, 1993.

Figure 4: A bird's eye view of an environment with 100 interconnected rooms.

Figure 5: 20000 normal particles traced.

Figure 6: 20000 importance driven particles traced. The partition on the left lower corner has been taken as the region of importance.

Figure 7: Close-up of the normal particle traced figure.

Figure 8: Close-up of importance driven particle traced figure.