The Visual Computer manuscript No.
(will be inserted by the editor)

Richard Sharp · Raghu Machiraju

Accelerating Subsurface Scattering Using Cholesky Factorization

Abstract In this paper we present a simplified subsurface scattering model which exploits a diffusion mechanism to provide a simpler solution to the transport equation. Our model is based on numerical analysis techniques which are amenable to Cholesky factorization. We treat the factorization as a precomputed scattering quantity which can be used to significantly speed up multiple scattering calculations as the global light source changes. On low resolution meshes we have been able to achieve real-time solutions of the subsurface scattering while still maintaining good visual quality of the solution.

Keywords Subsurface Scattering · Picture/Image Generation · Three-Dimensional Graphics and Realism

1 Introduction

Recently practical computational subsurface scattering models [12] have been introduced in computer graphics literature. This has been in part a drive to increase the quality of realistic image synthesis, although there has has been interest from the biomedical community to simulate and predict the behavior of light scattering in human tissue.

Previous research in this area has yielded various models of light transport, each suited for modeling scattering and substrate transport in different types of materials [1, 2, 6, 8, 12, 20, 28].

Richard Sharp
395 Dreese Laboratories
2015 Neil Avenue
Columbus, OH 43210
Tel.: (614)688-3766
Fax: (614)292-2911
E-mail: sharpr@cse.ohio-state.edu

Raghu Machiraju
395 Dreese Laboratories
2015 Neil Avenue
Columbus, OH 43210
Tel.: (614)292-6730
Fax: (614)292-2911
E-mail: raghu@cse.ohio-state.edu

Realistic looking images have been obtained through these models. Unfortunately, many of them have limitations such as restrictions to simple geometries [1, 2, 20], homogeneous materials [2, 12, 20] or are limited to one dimensional transport theory or BRDFs [1, 2, 6, 8, 18, 24].

Our transport model, which incorporates material modeling and addresses the issues raised above, has been previously presented in [26]. In this version, we present enhancements which drastically reduce the computation time for the underlying scattering calculations. These include a method to factor the scattering matrix and an implementation of spatial subdivision techniques to update source values due to changes in lighting in near real-time. The model presented in this paper assumes that the material properties can be modeled by a diffusion process (the scattering coefficient is much larger than the absorption coefficient) and, to ensure that our precomputation techniques can be used, we assume that the geometry and material scattering properties are fixed during the simulation.

In the computer graphics literature diffusion has been used to model subsurface scattering in media where diffusion dominates [12, 28]. Although we also use diffusion, we employ a simpler scattering framework which can can handle inhomogeneous materials. In essence we employ a restricted model of transport where flux propagates along a 3D grid of cells. Although this may seem restrictive it should be noted that in the limiting case our cell transport diffusion model will model the complete diffusion process and hence subsurface scattering in its entirety.

1.1 Paper Overview

The rest of our paper is as follows. First in Section 2 we discuss previous works including different subsurface light scattering models. Next in Section 3 we provide an introduction to our cell transport theory and tie our model with previous models Following this, in Section 4 we describe the computational methods we employed,

provide details on our precomputation and acceleration techniques. We present our final results in Section 6 and summarize our conclusions in Section 7.

2 Previous Work

S. Chandrasekhar first presented the equation of transport in his classic work *Radiative Transfer* [4] (see Equation 5). This equation essentially accounts for all radiance flowing through a surface by accounting for absorption, outscattering, inscattering and a source term. Any effort to model subsurface scattering or substrate transport of light is in reality an attempt to solve this equation. Kajiya was one of the first to use this mathematical machinery and proposed a stochastic approach of path tracing and Monte Carlo methods [15].

Blinn created the first subsurface scattering model in computer graphics to realistically model light scattering effects caused by the rings of Saturn [2]. Blinn's model uses forward and backward biased Henyey-Greenstein functions and physically based measurements to create probability distributions for particle scattering directions. Although the resulting model is elegant and generates impressive images, it is limited to thin surfaces of low albedo and does not account for higher order scattering events. On a similar note Kajiya and Von Herzen present the classic volume ray tracing algorithm [14] while Rushmeier and Torrance present a radiosity based solution for a similar scattering problem [25].

Baranoski and Rokne developed a model similar to Blinn for light transport in plant tissue [1]. However, the phase functions are more complex than that of Blinn's and are obtained from large number of physically based measurements. Once again only thin materials can be modeled.

Kniss *et al.* [19] has developed a model that uses forward scattering and volumetric light attenuation to improve the quality of volume shading. A model of this nature is useful in materials where forward scattering dominates, such as highly translucent material, clouds or even atmospheric effects [18, 24].

Researchers in biomedical engineering have also examined the transport of electromagnetic radiation and waves in human tissue [22, 23]. In [23] a diffusion approximation, the Delta-Eddington model, was used to model light transport. Specific boundary conditions were used to model various tissue arrangements and configurations. Also, measurements were included in the model for more accurate modeling of the transport phenomena. It should be noted that circuits are often used to model light propagation in media [22].

Jos Stam presented an implementation of multiple scattering as a diffusion processes in [28]. Stam's diffusion model was also derived from Ishimaru [9] but solved the diffusion process through a multi-grid finite difference scheme and a finite-element blob method. Ishimaru shows that one can use the diffusion approximation to accurately simulate light propagation through materials when scattering events are frequent. These occur in optically thick materials like skin or milk. Since we also use a diffusion approximation it should be noted that this transport model is similar to ours. However, in our model we treat light flux as potentials across neighboring voxels rather than continuous flux throughout the substrate.

Hanrahan and Kruger developed a bidirectional reflectance distribution function (BRDF) and a transport model for subsurface scattering that is a complete solution to the total first order scattering [8] which is limited to flat, uniformly lit, homogeneous slabs and is based on one dimensional transport theory.

Jensen *et al.* [12] introduced a fast Monte-Carlo subsurface scattering model that combined the diffusion approximation presented by Ishimaru [9] and the single order scattering BRDF from Hanrahan and Kruger [8]. Furthermore, the authors used a dipole source to satisfy the boundary condition in Ishimaru's diffusion approximation. As a result, Jensen *et al.* propose the use of the bidirectional surface scattering distribution function (BSSRDF) which is like a BRDF but allows flux to exit a substrate in a different location than it entered. It should be noted that the model which uses the dipole formulation is limited to homogeneous materials only.

Jensen proposed a variant of photon mapping for highly scattering materials in [10]. In this model diffuse photons are discretely traced through the substrate where at each time step a photon is either scattered, absorbed or left untouched for the next time step. The exiting diffuse light is estimated by gathering photons in the region that is to be lighted. This technique could be adapted to handle non-homogeneous materials, but would require variable stepsizes depending on the scattering properties of the current material the photon is being traced through. Also, during the gathering step one must account for the contribution of photons differently depending on the scattering properties of the material that that photon resides.

Haber *et al.* [7] use a multigrid solution to solve the diffusion equation directly on arbitrarily shaped meshes. The multigrid method is a good method for quickly solving the diffusion equation, however if the source changes, the entire solution must be recalculated. Our work addresses this issue by prefactoring the linear system independently of the light source.

Jensen and Buhler present an accelerated subsurface scattering model in [11] which uses a two pass hierarchical sampling technique to significantly speed up the evaluation of the BSSRDF presented in [12]. Furthermore, Sloan *et al.* have developed techniques to precompute the radiance, and in some cases subsurface scattering, in scenes with low frequency dynamic light sources [27].

Finally there has been a significant amount of research to accelerate subsurface scattering calculations on

modern graphics hardware [3, 5, 21] through simplifications of the scattering model or through implementation of numerical methods on the graphics processing unit.

3 Theory

To recapitulate, our transport model is realized as a steady state diffusion process and can be conceived as an electrical circuit. Please note that this is only used as an analogy. Our transport model can also be realized numerically. As a result inhomogeneous materials, complex geometries and measurements can be easily included. This section provides necessary justification for our model.

$p(\hat{\mathbf{s}}, \hat{\mathbf{s}}')$	the phase function of the angle between $\hat{\mathbf{s}}$ and $\hat{\mathbf{s}}'$.
$\overline{\mu}$	the mean cosine of the scattering angle
ρ	particle density
σ_a	absorption cross section
σ_s	scattering cross section
σ_t	extinction cross section $(\sigma_a + \sigma_s)$
σ_{tr}	transport cross section $(\sigma_s(1-\overline{\mu}) + \sigma_a)$
$I_d(\mathbf{r}, \hat{\mathbf{s}})$	diffuse intensity
$I_{ri}(\mathbf{r}, \hat{\mathbf{s}})$	reduced incident intensity
$\varepsilon_{ri}(\mathbf{r}, \hat{\mathbf{s}})$	source function due to reduced incident intensity
$U_d(\mathbf{r})$	average diffuse intensity
$U_{ri}(\mathbf{r})$	average reduced intensity
κ	$(\rho\sigma_{tr})^{-1}$
α	$3\rho\sigma_a$

Table 1 Table of terms.

3.1 Cell Transport Model

Our model assumes that the solution for light propagation through highly scattering media is analogous voltage propagation through a resistive network. This section will show how light is related to current and how the solution of a resistive network is the same as the solution to the transport equation.

3.1.1 Irradiance is Related to Voltage

Irradiance (radiant power per unit area) is the integration of incoming radiance over all directions, its units are W/m^2. Hence, we can write irradiance as power per unit area:

$$\frac{d\phi}{dA} = E = \frac{p}{dA}. \tag{1}$$

Likewise in we can write power as a product of voltage and current:

$$p = \frac{dw}{dt} = \frac{dw}{dq} \cdot \frac{dq}{dt} = vi. \tag{2}$$

From Ohm's law we can rewrite this equation in terms of voltage only:

$$p = \frac{v^2}{R}. \tag{3}$$

Resistor R has no physical significance in our optical network other than the initial value of the voltage and hence can be chosen arbitrarily. This leads us to understand the relationship between voltage and irradiance as:

$$E \propto \frac{v^2}{dA} \Rightarrow v \propto \sqrt{E \cdot dA}. \tag{4}$$

3.1.2 Transport Equation in terms of Kirchoff Current Laws

In this section we formulate the Transport Equation can as a diffusion equation, and then formulate the diffusion equation in terms of Kirchoff Current Laws.

Ishimaru shows that single scattering and first order solutions are applicable when the volume density (ratio of the volume occupied by particles to the total volume of the media) is much less than 0.1% [9] (such as Blinn's Saturn rings). The diffusion approximation gives good solutions when the density is much greater than 1% which exists in highly scattering media, such as blood [13].

Intensity in a random medium can be divided into two parts, the reduced incident intensity I_{ri} and the diffuse intensity I_d. Reduced incident intensity is the part of the flux that remains after scattering and absorption. We denote it by $I_{ri}(\mathbf{r}, \mathbf{s})$ where $\mathbf{r}$ is the point at which the flux is measured and $\mathbf{s}$ is the unit vector in which it is propagating. Its behavior satisfies the equation

$$\frac{dI_{ri}(\mathbf{r}, \mathbf{s})}{ds} = -\rho\sigma_t I_{ri}(\mathbf{r}, \mathbf{s})$$

which shows that the value of I_{ri} decays exponentially as we travel away from its origin where ρ is the particle density and σ_t is the extinction cross section. (Note that terms are described in Table 1).

The diffuse intensity must satisfy the equation of transfer:

$$\frac{dI_d(\mathbf{r}, \mathbf{s})}{ds} = -\rho\sigma_t I_d(\mathbf{r}, \mathbf{s}) + \frac{\rho\sigma_t}{4\pi} \int_{4\pi} \rho(\mathbf{s}, \mathbf{s}') I_d(\mathbf{r}, \mathbf{s}) \mathrm{d}\omega' + \varepsilon_{ri}(\mathbf{r}, \mathbf{s}). \tag{5}$$

In this case ε_{ri} is the reduced source term which can be calculated from the reduced intensity:

$$\varepsilon_{ri}(\mathbf{r}, \hat{\mathbf{s}}) = \frac{\rho\sigma_t}{4\pi} \int_{4\pi} p(\hat{\mathbf{s}}, \hat{\mathbf{s}}') I_{ri}(\mathbf{r}, \hat{\mathbf{s}}) \, d\omega'. \tag{6}$$

By expanding I_d into the first two terms of its Taylor's series expansion and substituting back into Equation 5 we end up with the steady state diffusion approximation (see the details of the derivation in [9] pp 175-178):

$$\nabla^2 U_d(\mathbf{r}) - \kappa^2 U_d(\mathbf{r}) = -3\rho^2\sigma_s\sigma_{tr}U_{ri}(\mathbf{r}) - \frac{3}{4\pi}\rho\sigma_{tr}E(\mathbf{r}) + \frac{3}{4\pi}\nabla\cdot\int_{4\pi}\varepsilon_{ri}(\mathbf{r},\mathbf{s})\mathrm{s}\mathrm{d}\omega. \quad (7)$$

Where, U_d and U_{ri} are the uniform diffuse and reduced intensities respectively. By collecting the scattering terms, we write this equation in a more general form,

$$\nabla^2 u - au = \varepsilon \quad (8)$$

where
- u diffusion term
- a absorption coefficient
- ε source term.

We present Equation 8 in finite difference form. Assuming we divide our volume up into equal Cartesian cubes of width h:

$$\varepsilon_{ijk} = -au_{ijk} + \frac{1}{h^2}\left[\triangleright_x(u_{ijk}) + \triangleleft_x(u_{ijk}) + \triangleright_y(u_{ijk}) + \triangleleft_y(u_{ijk}) + \triangleright_z(u_{ijk}) + \triangleleft_z(u_{ijk})\right] \quad (9)$$

Where the forward and backward difference operators $\triangleright_x$ and $\triangleleft_x$ are defined as

$$\triangleright_x(u_{i,j,k}) = u_{i+1,j,k} - u_{i,j,k}$$
$$\triangleleft_x(u_{i,j,k}) = u_{i-1,j,k} - u_{i,j,k}.$$

As shown previously it is possible to represent irradiance in terms of voltage or current, knowing this we treat $u_{i,j,k}$ as a current flowing through the center of a finite volume centered at position (i,j,k). To model the potential for current to spread to neighboring finite elements we connect the nodes through discrete resistors whose values represent the likelihood of current to pass through the space between the two points, much like a phase function.

Thus we can rewrite Equation 9 as a current equation in terms of node voltages and resistances for every cell (i,j,k):

$$\frac{\Delta_V Vs_{i,j,k}}{R_{s_{i,j,k}}} = -\frac{V_{i,j,k}}{R_{g_{i,j,k}}} + \frac{1}{h^2}\left(\frac{\Delta_V V_{i+1,j,k}}{R_{i+,j,k}} + \frac{\Delta_V V_{i-1,j,k}}{R_{i-,j,k}} + \frac{\Delta_V V_{i,j+1,k}}{R_{i,j+,k}} + \frac{\Delta_V V_{i,j-1,k}}{R_{i,j-,k}} + \frac{\Delta_V V_{i,j,k+1}}{R_{i,j,k+}} + \frac{\Delta_V V_{i,j,k-1}}{R_{i,j,k-}}\right). \quad (10)$$

where $\Delta_V v \equiv v - V_{i,j,k}$ and $R_{i+,j,k}$ is defined as the resistor connecting nodes (i,j,k) and $(i+1,j,k)$, similarly, $R_{i-,j,k}$ indicates the resistor connecting nodes (i,j,k) and $(i-1,j,k)$ and so on. These resistor values represent preferred scattering directions in the material, where the lower the value, the higher the scatter. These resistors are generalization of the extinction and scattering coefficient.

Boundary conditions in the diffusion equation are formulated by enforcing that the inward integral of the diffuse intensity on the boundary must be zero [9, 28]. Essentially this condition states that the reduced intensity must be generated from inside the medium. We enforce this condition by using the following equation for boundary nodes

$$0 = \frac{V_{i',j',k'} - V_{i,j,k}}{h} \quad (11)$$

where node (i',j',k') is the closest node to the boundary along the normal.

Since this model is based on a diffusion approximation it should be noted that that it will not be able to handle scattering in following material types

- Transparent materials - Since we are modeling subsurface scattering as a diffusion process, transparent materials would not render correctly in our model.
- Thin materials - In this case less computationally complex single scattering effects dominate these solutions.

To validate our model we employ a discrete 15×15 grid to compare our model and the diffusion approximation given by the Jensen *et al.* model [12]. The optical properties that were chosen were within range of the "Wholemilk" data recorded in the same paper.

We stimulate the two dimensional grid with a step input shown in Figure 1(a). In the model presented in [12] this corresponds to a unit level of irradiance incident on the surface, while in our model it corresponds to setting the middle group of cells' source voltage to approximately 1 volt.

The response from the [12] model and our impedance network is shown in Figure 1 which are quite similar.

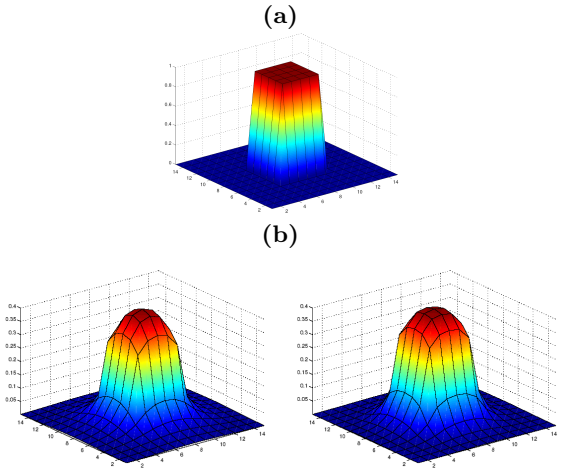


Fig. 1 (a) The step initial source value. (b) Response of Jensen *et al.* [12] model with $\sigma_a = 0.0041$ and $\sigma_s' = 2.6$ (left) vs. Response of our model with 1.18V step input and all resistors set to 1kΩ (right).

4 Solution Methods

This section discusses various data structures and algorithms we used to implement our transport impedance network.

4.1 Storage

As presented earlier, Equation 10 defines the scattering equation for every cell. Since our model limits flux transport to only neighboring cells, each equation will be dependent on at most six other equations. The resulting system is sparse where an impedance grid of size $i \times j \times k$ will generate a matrix of dimensions $(i \cdot j \cdot k)^2$ with at most $6i \cdot j \cdot k$ non-zero elements.

To store our linear system we use the compressed column storage mode used by the TAUCS linear solver library [29]. This technique requires approximately $2d + 2n$ space to store n non-zero elements in a matrix of size $d \times d$.

4.2 Solving the system

Once we establish our linear system of equations we solve a system of the form

$$Ax = b. \tag{12}$$

Where matrix A is defined by Equation 10 and vector b is the source values found at each node. In general, Equation 10 generates a matrix that is both symmetric and positive definite (so long as resistor values are chosen such that the matrix remains diagonally dominant).

4.3 Efficiency Enhancements

In this section we discuss our enhancements to [26] which include precomputation of the scattering matrix, A (from Equation 12) and the use of spatial subdivision and approximations to accelerate source calculations. These enhancements combined with a scattering grid of reasonable size can generate subsurface scattering updates in real-time as the light source moves.

4.3.1 Precomputation

There are many methods for solving systems of the form $Ax = b$. However if A is positive-definite, Cholesky decomposition is an economical method for decomposing A into the form

$$A = LL^T. \tag{13}$$

To use the factor to solve $Ax = b$ we first solve $Ly = b$ for y, then $L^T x = y$ for x. Solving is also an inexpensive process since both vectors y and x can be directly solved using back-substitution. Furthermore, if L is sparse we can decrease the computation time even further.

Unfortunately, simply because A is sparse does not guarantee that its factor will also be. In fact, in worse case the factor may be a full matrix! Fortunately, graph partitioning algorithms exist which can be used to precondition a matrix such that the fill ordering of its factor is also sparse [17]. In our implementation we use the sparse matrix package, TAUCS [29], which preconditions the matrix using the the multilevel graph partitioning algorithm implemented in the METIS [16] library. Table 2 shows our fill rates given original number of non-zeros in the matrix. In general, Cholesky factorization increases the fill of the factor by less than an order of magnitude. The "Upper Bound Fill" column was estimated by assuming that in the worst case, every unknown would fill an upper triangular row completely. Note that actual fill rates are at least two orders of magnitude smaller than that of a potentially full factor.

4.3.2 Efficient Source Calculation

The vector b defines the value of source voltages at all the nodes in the scattering mesh. The value of the voltage at a node should be proportional to that of the value of ε_{ri} at the same point. However, reasonable simplifications can be made to reduce the complexity of calculating b. First, the diffusion approximation is only valid in materials where scattering is the dominant transport mechanism. Practically this means that the mean path, defined as the average distance a photon travels before scattering, is quite short with respect to the size of the global geometry. Thus the value of ε_{ri}, which is defined as the amount of light which is unscattered and unattenuated, drops off significantly a small distance beneath the surface where it has little effect on the final solution. We take advantage of this exponential falloff by only calculating values for b in cells which lie on, or touch the surface of the object. Internal cells are assumed to have no source.

Finally, to determine if a cell has any light incident upon it, we trace a shadow ray from the corners of the cell to the light source. If any of the corners of the cell are illuminated we initialize the voltage based on the inner product of the average normal of the polygons with the light vector. In order to accelerate the ray test, which is critical for interactive scattering updates, we use an octree with eight levels and allow up to six primitives to be stored in the leaves (see Figure 2).

Since the scattering matrix A is independent of any light source, we can keep the factor L, update b when the light source changes and reuse the factor to determine the new scattering distribution. In Table 2 we show the speedup obtained by using the factor during the scattering rendering step.

To summarize, our precomputation algorithm is as follows:

Al model

Grid Size	Factor Time	U_{ri} Time	Solve Time	Non-Zeros in A	Non-Zeros in L	Upper Bound Fill for L
64×64×32	$25.438s$	$7.250s$	$0.453s$	2.40×10^5	5.37×10^6	8.00×10^8
32×32×16	$1.593s$	$1.422s$	$0.046s$	3.92×10^4	3.99×10^5	2.13×10^7
16×16×8	$0.031s$	$0.281s$	$< 0.001s$	6.76×10^3	5.35×10^4	6.35×10^5
8×8×4	$< 0.001s$	$0.046s$	$< 0.001s$	1.09×10^3	4.70×10^3	1.65×10^4

Dolphin model

Grid Size	Factor Time	U_{ri} Time	Solve Time	Non-Zeros in A	Non-Zeros in L	Upper Bound Fill for L
64×64×32	$29.750s$	$2.578s$	$0.359s$	1.26×10^5	1.54×10^6	2.20×10^8
32×32×16	$0.610s$	$0.594s$	0.047	2.21×10^4	1.35×10^5	6.78×10^6
16×16×8	$0.032s$	$0.141s$	$0.016s$	4.60×10^3	1.56×10^4	2.94×10^5
8×8×4	$< 0.001s$	$0.031s$	$< 0.001s$	9.00×10^2	3.46×10^3	1.13×10^4

Table 2 Run time statistics for Al and Dolphin model (See Figures 3 and 4). A is the scattering matrix, while L is its Cholesky factor. Note that the performance gain can be considered to be the amount of time taken to solve the entire system (the sum of "factor", "solve" and "source calculation" time) as compared to factoring once, and reusing the factor to repeatedly solve the system. There is additional run time overhead involved in copying the solution to a 3D texture in the graphics card.

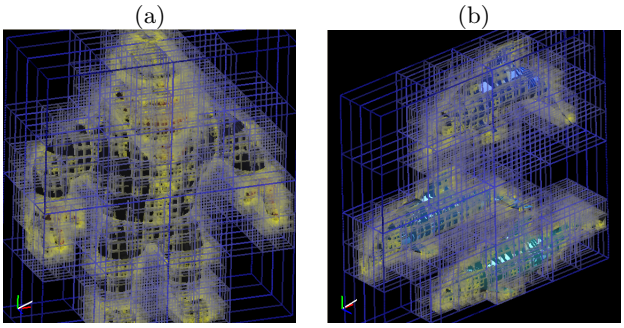


Fig. 2 Octree used in calculating source vector for (a) Al model and (b) Dolphin model. Levels range from blue (lowest) to yellow (highest).

1. Build global matrix A based on Equation 10.
2. Compute Cholesky factorization, LL^T.
3. Compute light source vector b (using octree data structure to accelerate intersections).
4. Solve for x using L.
5. If the location or intensity of the light source changes go to step 3.

5 Rendering

Our general rendering algorithm is as follows:

1. Load a 3D mesh.
2. Build a scattering grid to fit inside the mesh[1].
3. Set the resistor values on the internal grid[2].
4. Perform precomputation algorithm described in Section 4.3.2.
5. Render the image by treating the solved grid as a 3D luminare and projecting it onto the surface of the 3D model[3]. The nodal voltages are proportional to the intensity values exiting the material.
6. If the location or intensity of the light source[4] changes go to step 6.

6 Results

Figures 3 and 4 show the differences between scattering and no scattering on public domain `.obj` files with constant internal impedance. Although our model can be used to calculate subsurface scattering effects on inhomogeneous models such as Figure 5 and 6, we have found that the calculation of the light source vector is too expensive to calculate in real time due to the higher grid resolution required to accurately capture the self shadowing effects in the subsurface.

The time expended on precomputation and solving and non-zero fill stages are described in Table 2. Although denser meshes should give more accurate results, we have found that low density meshes give appealing visual results while also being coarse enough to update the subsurface scatter at interactive rates (approximately

[1] The mesh can be generated with many methods, but for our implementation we embedded a cube of cells around the model, then removed cells which were outside of the surface.

[2] In several examples in this paper we set the values based on a mapping from the MRI scalar data with the transfer function, see Figures 5 and 6 and [26] for details.

[3] In Figures 2, 3 and 4 we used OpenGL to do the rendering, while Figures 5 and 6 are rendered in Maya

[4] There is no limitation that the light source be a point source. In fact, Figures 5 and 6 both are illuminated from area light sources.

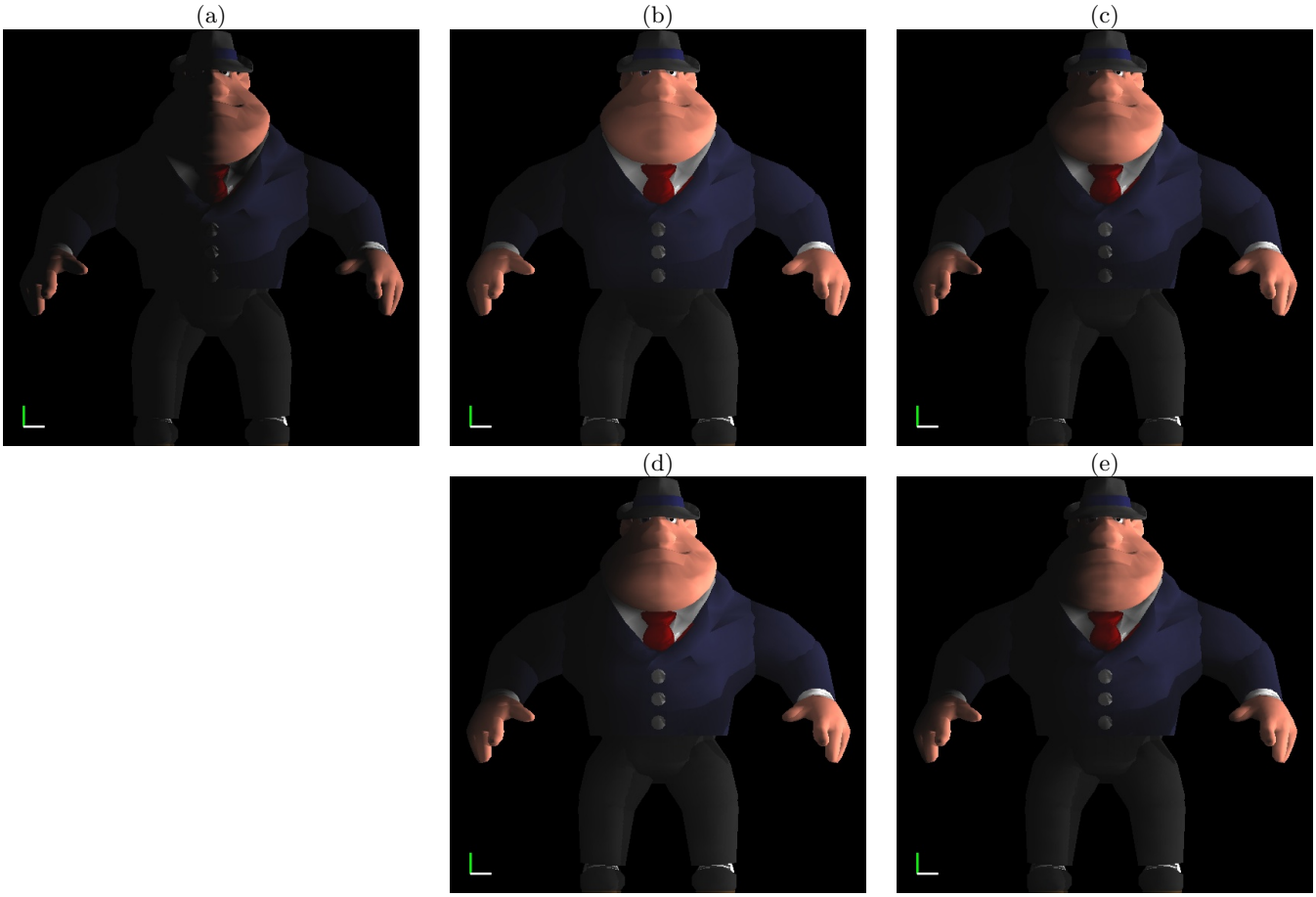


Fig. 3 (a) Al model with no subsurface scattering and with embedded subsurface grids of size (b) 8×8×4, (c) 16×16×8, (d) 32×32×16 and (e) 64×64×32.

0.5-1 frames per second (fps) on a $16 \times 16 \times 8$ grid). Since our implementation was realized in OpenGL we incurred additional overhead in updating the 3D texture per frame as the light source moved, thus our render times are not only the source calculation plus solve times as shown in Table 2 but also the time necessary to copy the 3D texture to the graphics card. However, if the light source is fixed the rendering becomes trivial since the 3D texture need not be updated. In this case completely interactive (30+ fps) rates can be achieved.

7 Conclusions and Future Work

We have shown that by using a simplified scattering model in conjunction with Cholesky factorization and ray acceleration techniques we can greatly accelerate the subsurface scattering calculations, even to the point of interactive framerates. Our model is not only amenable to precomputation but the presence of a resistor network allows the user to easily model inhomogeneous materials and build user defined impedance networks.

In future work we are considering accelerating the light source calculations further by pushing the shadow computation onto the graphics hardware. We also plan to explore other methods of solving the linear system such as an iterative solver based approach implemented on the GPU.

References

1. Baranoski, G., Rokne, J.: A simplified model for light interaction with plant tissue. In: Eighth International Conference on Computer Graphics and Visualization - GraphiCon'98, pp. 154–161 (1998)
2. Blinn, J.F.: Light reflection functions for simulation of clouds and dusty surfaces. Computer Graphics pp. 21–29 (1982)
3. Carr, N., Hall, J., Hart, J.: Gpu algorithms for radiosity and subsurface scattering. In: HWWS '03: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware, pp. 51–59. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2003)
4. Chandrasekhar, S.: Radiative Transfer. Dover Publications, Inc. (1960)
5. Dachsbacher, C., Stamminger, M.: Translucent shadow maps. In: Proceedings of the 14th Eurographics work-

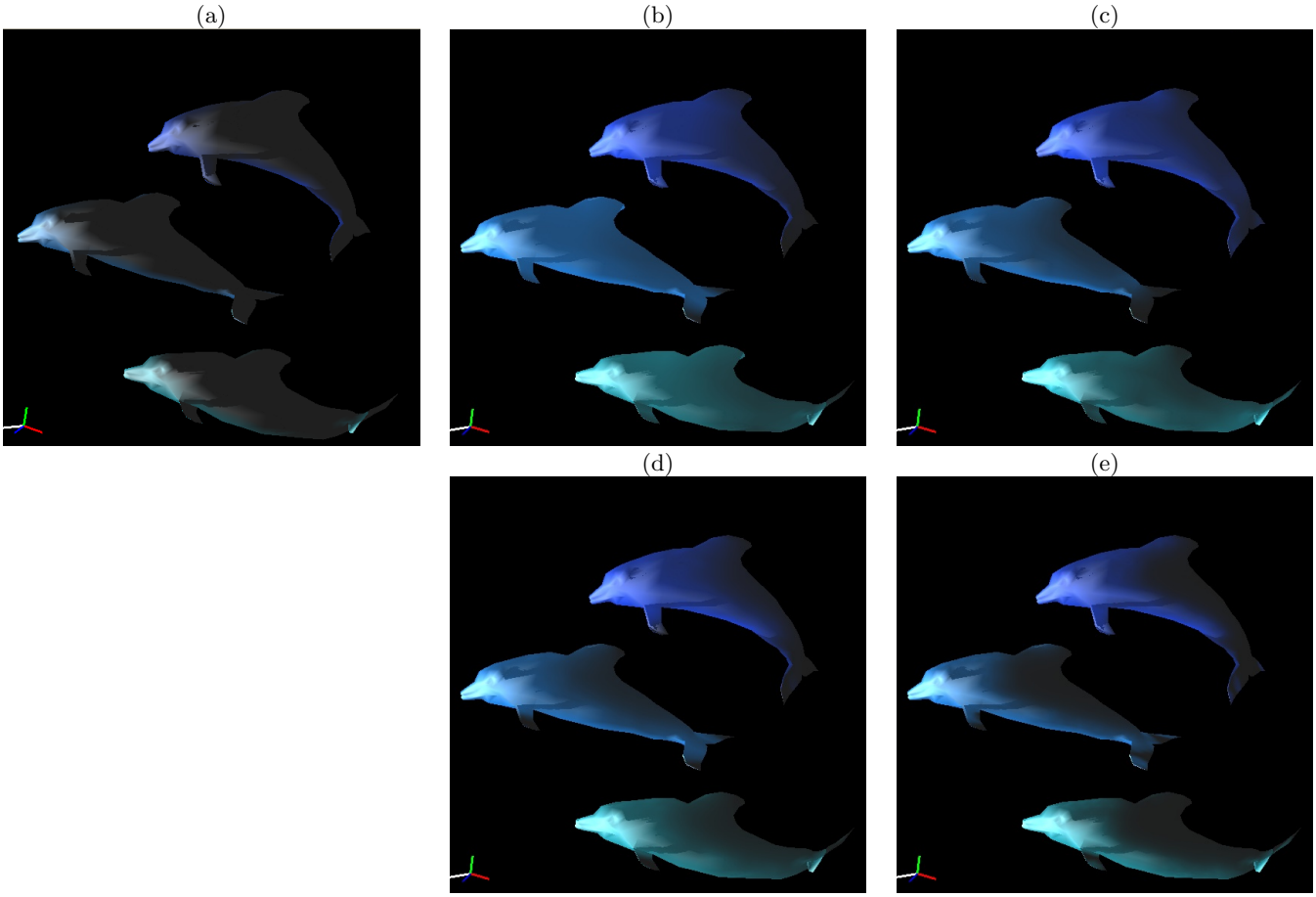


Fig. 4 (a) Dolphin model with no subsurface scattering and with embedded subsurface grids of size (b) 8×8×4, (c) 16×16×8, (d) 32×32×16 and (e) 64×64×32.

Fig. 5 Side lit foot with no scattering vs. scattering.

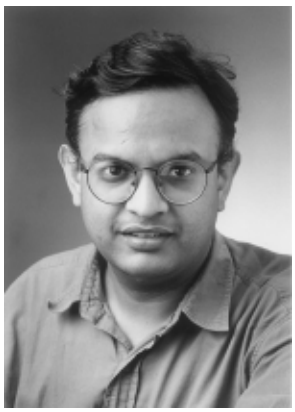
shop on Rendering, pp. 197–201. Eurographics Association (2003)

6. Dorsey, J., Hanrahan, P.: Modeling and rendering of metallic patinas. In: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, pp. 387–396. ACM Press (1996). DOI http://doi.acm.org/10.1145/237170.237278
7. Haber, T., Meterns, T., Bekaert, P., Reeth, F.V.: A computational approach to simulate subsurface light diffusion in arbitrarily shaped objects. In: Graphics Interface (2005)
8. Hanrahan, P., Krueger, W.: Reflection from layered surfaces due to subsurface scattering. In: Proceedings of the 20th annual conference on Computer graphics and interactive techniques, pp. 165–174. ACM Press (1993). DOI http://doi.acm.org/10.1145/166117.166139
9. Ishimaru, A.: Wave Propagation and Scattering in Random Media. Academic Press, New York (1978)
10. Jensen, H.W.: Realistic Image Synthesis using Photon Mapping. AK Peters (2001)
11. Jensen, H.W., Buhler, J.: A rapid hierarchical rendering technique for translucent materials. SIGGRAPH **21**(3), 576–581 (2005)

Fig. 6 Bottom lit foot with scattering but uniform impedance vs. bottom lit foot with non-uniform impedance values based on MRI density values.

12. Jensen, H.W., Marschner, S.R., Levoy, M., Hanrahan, P.: A practical model for subsurface light transport. In: Proceedings of the 28th annual conference on computer graphics and interactive techniques, pp. 511–518. ACM Press (2001). DOI http://doi.acm.org/10.1145/383259.383319
13. Johnson, C.C.: Optical diffusion in blood. In: IEEE Trans., pp. 129–133 (1970)
14. Kajiya, J., Von Herzen, B.: Ray tracing volume densities. In: H. Christiansen (ed.) Computer Graphics (SIGGRAPH '84 Proceedings), vol. 18, pp. 165–174 (1984)
15. Kajiya, J.T.: The rendering equation. In: SIGGRAPH '86, pp. 143–150. ACM Press (1986)
16. Karypis, G.: Metis: Family of multilevel partitioning algorithms. http://www-users.cs.umn.edu/ karypis/metis/
17. Karypis, G., Kumar, V.: A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J. Sci. Comput. **20**(1), 359–392 (1998). DOI http://dx.doi.org/10.1137/S1064827595287997
18. Kniss, J., Premoze, S., Hansen, C., Ebert, D.: Interactive translucent volume rendering and procedural modeling. In: Proceedings of the conference on Visualization '02, pp. 109–116. IEEE Computer Society (2002)
19. Kniss, J., Premoze, S., Hansen, C., Shirley, P., McPherson, A.: A model for volume lighting and modeling. IEEE Transactions on Visualization and Computer Graphics **9**(2), 150–162 (2003). DOI http://dx.doi.org/10.1109/TVCG.2003.1196003
20. Max, N.L.: Efficient Light Propagation for Multiple Anisotropic Volume Scattering. In: Fifth Eurographics Workshop on Rendering, pp. 87–104. Darmstadt, Germany (1994). URL citeseer.nj.nec.com/max94efficient.html
21. Mertens, T., Kautz, J., Bekaert, P., Seidelz, H., Van Reeth, F.: Interactive rendering of translucent deformable objects. In: Proceedings of the 14th Eurographics workshop on Rendering, pp. 130–140. Eurographics Association (2003)
22. Panfilov, A.V., Holden, A.V. (eds.): Computational Biology of the Heart. John Wiley & Sons (1997)
23. Prahl, S.A.: Light transport in tissue. Ph.D. thesis, University of Texas at Austin (1988)
24. Riley, K., Ebert, D., Hansen, C., Levit, J.: Visually accurate multi-field weather visualization. In: Proceedings IEEE Visualization, pp. 279–286 (2003)
25. Rushmeier, H., Torrance, K.: The zonal method for calculating light intensities in the presence of a participating medium. Computer Graphics (SIGGRAPH '87 Proceedings) **21**(4), 293–302 (1987)
26. Sharp, R., Machiraju, R.: A simplified model for inhomogeneous subsurface scattering. In: Volume Graphics, pp. 63–72. EuroGraphics/IEEE Computer Society VGTC (2005)
27. Sloan, P.P., Kautz, J., Snyder, J.: Precomputed radiance transfer for real-time rendering in dynamic, low-frequency lighting environments. In: SIGGRAPH (2002)
28. Stam, J.: Multiple scattering as a diffusion process. In: Proceedings of the 6th Eurographics Workshop on Rendering, pp. 51–58. Dublin, Ireland (1995)
29. Toledo, S., Chen, D., Rotkin, V.: Taucs: A library of sparse linear solvers. http://www.tau.ac.il/ stoledo/taucs/

Richard Sharp Richard Sharp is a Ph.D. candidate at The Ohio State University. His research interests include physics modeling, bioinformatics and software visualization.



Raghu Machiraju Raghu Machiraju is an Associate Professor of Computer Science and Engineering at The Ohio State University since January 2000. Earlier, he was at the NSF Engineering Research Center, Mississippi State University. His research interests include rendering, feature detection and computational methods and modeling.

Figure 1 (a) The step initial source value. (b) Response of Jensen *et al.* [12] model with $\sigma_a = 0.0041$ and $\sigma'_s = 2.6$ (left) vs. Response of our model with 1.18V step input and all resistors set to 1kΩ (right).
Figure 2 Octree used in calculating source vector for (a) Al model and (b) Dolphin model. Levels range from blue (lowest) to yellow (highest).
Figure 3 (a) Al model with no subsurface scattering and with embedded subsurface grids of size (b) 8×8×4, (c) 16×16×8, (d) 32×32×16 and (e) 64×64×32.
Figure 4 (a) Dolphin model with no subsurface scattering and with embedded subsurface grids of size (b) 8×8×4, (c) 16×16×8, (d) 32×32×16 and (e) 64×64×32.
Figure 5 Side lit foot with no scattering vs. scattering.
Figure 6 Bottom lit foot with scattering but uniform impedance vs. bottom lit foot with non-uniform impedance values based on MRI density values.

Table 3 Figure Legend.