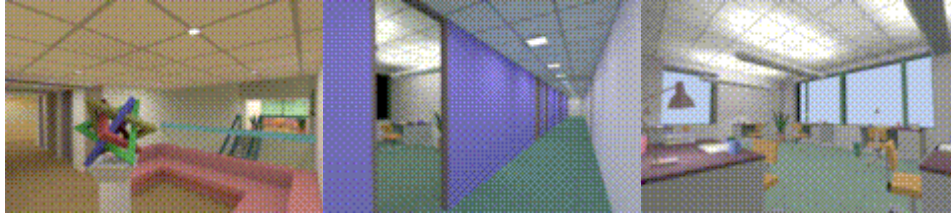


Radiosity Lighting Simulation



[Thomas Funkhouser](#)

Overview:

Radiosity methods accurately simulate diffuse indirect illumination and shadows, and thus are used to generate realistic-looking lighting models for a variety of virtual environments, including building interiors. A difficult challenge for radiosity systems is managing the algorithmic complexity ($O(n^2)$) and massive data storage requirements (GBs) typical in such computations. We have developed a radiosity system that computes radiosity solutions for very large polygonal models.

The first innovation in this system is that it uses visibility oracles and hierarchical methods to: 1) reduce the number of polygon-polygon interactions considered, and 2) partition the computation into a sequence of subcomputations each requiring a relatively small working set. Unlike any other system, the radiosity solver stores the evolving solution in a disk-resident database and loads only the working set for the current subcomputation into memory as the computation proceeds. Subcomputations are ordered so as to minimize the impact of database I/O operations. Using these techniques, the system is able to cull over 99.999999% of the potential interactions and requires only 0.24% of the database (14.5MB) to be stored in memory at any given time during experiments with large architectural models.

The second innovation is that it supports execution of multiple hierarchical radiosity solvers working on the same radiosity solution in parallel. The system is based on a group iterative approach that repeatedly: 1) partitions patches into groups, 2) distributes a copy of each group to a slave processor which updates radiosities for all patches in that group, and 3) merges the updates back into a master solution. The primary advantage of this approach is that separate instantiations of a hierarchical radiosity solver can gather radiosity to patches in separate groups in parallel with very little contention or communication overhead. This feature, along with automatic partitioning and dynamic load balancing algorithms, enables our implemented system to achieve significant speedups running on moderate numbers of workstations connected by a local area network.

This system has been used to compute the radiosity solution for a furnished model of Soda Hall. The model represents five floors of a large building with approximately 250 rooms containing furniture. It was constructed with 14,234 clusters comprising 280,836 patches, 8,542 of which were emitters and served as the only light sources. The total area of all surfaces was 75,946,664 square inches. Three complete iterations were made through all patches using an average of 4.96 slave processors in 168 hours. The entire computation generated 7,649,958 mesh elements and evaluated 374,845,618 element-to-element links.

This is joint work with [Seth Teller](#), Celeste Fowler, and [Pat Hanrahan](#).

Related Links:

- Soda Hall Model

- [UC Berkeley Walkthru Group](#)
 - [MIT Synthetic Imagery Group](#)
 - [Stanford Computer Graphics Laboratory](#)
-

Related Publications:

- Thomas A. Funkhouser.

[Coarse-Grained Parallelism for Hierarchical Radiosity Using Group Iterative Methods.](#)

Computer Graphics (SIGGRAPH '96), New Orleans, LA, August, 1996.

- Seth Teller, Celeste Fowler, Thomas Funkhouser, and Pat Hanrahan.

[Partitioning and Ordering Large Radiosity Computations.](#)

Computer Graphics (SIGGRAPH '94), Orlando, FL, August, 1994, p. 443-450.

- Thomas A. Funkhouser.

[Database and Display Algorithms for Interactive Visualization of Architectural Models.](#)

PhD Thesis, Computer Science Division, UC Berkeley, September, 1993.

Also available as UC Berkeley Technical Report UCB/CSD93/771.

This document was created with Win2PDF available at <http://www.win2pdf.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.