LA-UR-02-6927

Approved for public release; distribution is unlimited.

Title: A 3-D Ray-Trace Model for an AMR Code on Distributed Processors

Author(s): Rod Mason, X-1

Submitted to: For presentation at the 44th Annual Meeting of the APS Division of Plasma Physics, Orlando, FL, November 11-15, 2002

Los Alamos
NATIONAL LABORATORY

A 3-D RAY-TRACE MODEL FOR AN AMR CODE ON DISTRIBUTED PROCESSORS

R. J. Mason

Los Alamos National Laboratory

Los Alamos, NM 87545

44TH Annual Meeting of the Division of Plasma Physics, Orlando, FL

Nov. 11-15, 2002

Rays_02_a

ABSTRACT

Distributed memory AMR codes provide a special challenge for laser ray-trace modeling. For RAGE we follow the energy-carrying rays through each cell, checking for crossings which require collection a new cell index (1 of 9 in 2D). Density gradients for ray deflections can be calculated "on the fly." Substantial excursions must be made from the legacy PIC particle area-weighting approach, but this serves as a useful 1st step. Similarly, IDL now offers a quick graphical rendering, while ENSIGHT graphics beautifully captures the 3D light refraction and deposition.

Rays_02_b

A new ray-trace model is under construction for the 3-D Eulerian AMR code RAGE. The model employs:

- **Ray particles are deflected by density gradients volume-weighted to local positions along their trajectories**
- **Energy absorption by inverse-bremsstrahlung and dump-all**
- **Possible particle-front stringing for smoothed deposition**
- **Particle splitting for matching to adaptively refined cells**
- **Rich 3-D perspective with ENSIGHT graphics**

Approach: A PC first, then coding on the SGI

We first built a basic 3D algorithm *off line, on a PC* to probe the basic issues.

We used Compaq Digital Fortran.

The mesh was structured and uniform. Neighbors were i ±1, j ±1, k ±1. A sphere was established with a max density of 2 n_{crit} on a 50 x 50 x 50 mesh. Later this was placed inside a Hohlraum.

A single fan of rays was run against the target.

First IDL, and later ENSIGHT graphics was employed.

Ray energy was deposited with area weighting at the cell centers. As ray intensity diminished, the plotted ray color was changed from red to blue.

The density gradients and deposition were linearly volume-weighted to the 3-D mesh

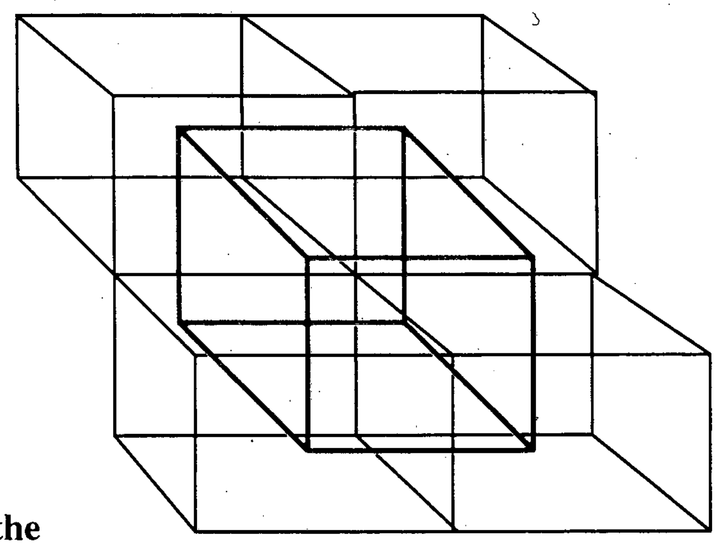
• Locate a ray – "particle's" base cell
via its integer order x/Δx, y/Δy, z/Δz

• Compute density gradients from their cell-centered differences

• Store density gradients at the cell-face centers

• Select the contributing gradients
the ray - particle's position relative
to its cell-center

• Area weight the gradients to the Instantaneous ray - particle's position



Rays_02_e

The ray-trace refraction and absorption equations are quite straightforward

Ray trajectories:

$$\frac{d\vec{v}}{dt_r} = -\frac{c^2}{2n_{crit}}\nabla n$$

Time steps for the rays are chosen to sample the conditions in each cell at least eight times (dt_r is reduced with refinement)

$$\frac{d\vec{x}}{dt_r} = \vec{v}$$

Inverse-Bremsstrahlung:

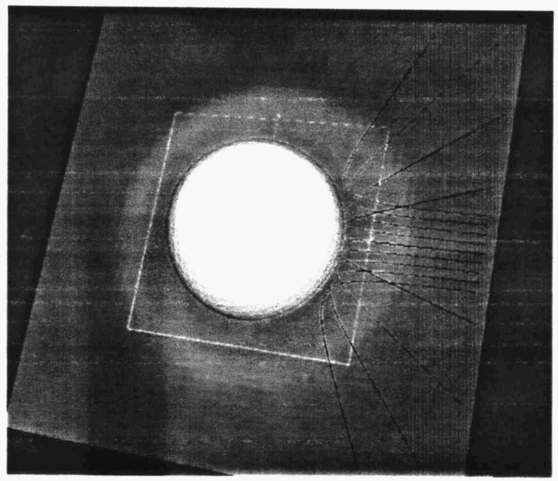
$$\frac{dI}{ds} = -K_a, \qquad K_a = \frac{c_2 n_e n_i Z^2 g_f s_p}{(k_B T_e)^{3/2}(c/\lambda)^3(1-n_e/n_{crit})^{1/2}}$$

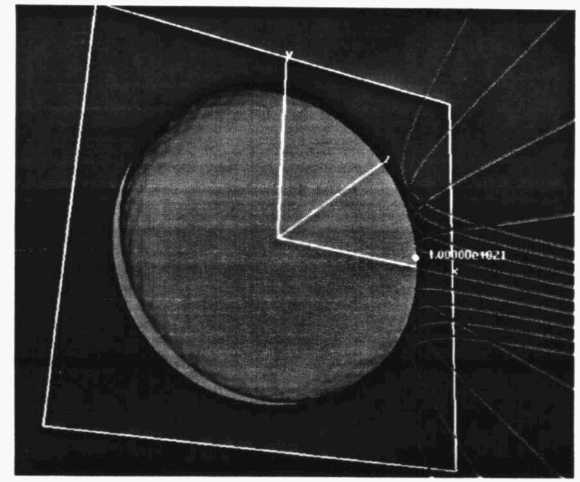
Plus dump-all for $n > n_{dmp}$

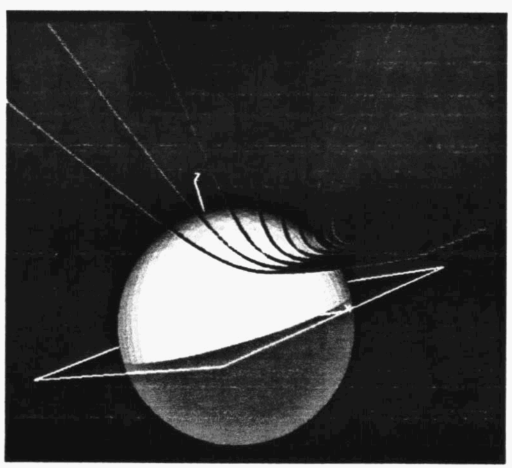
Rays_02_f

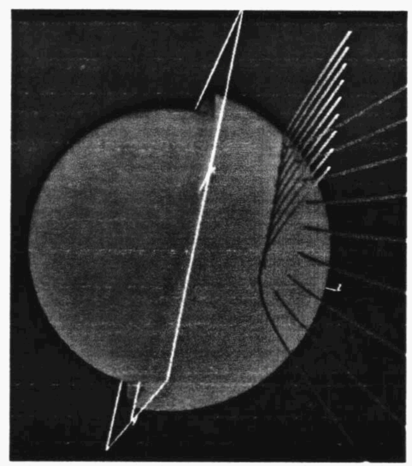
Various tests bear out equatorial in-plane reflection and full 3-D refraction

Equatorial interaction







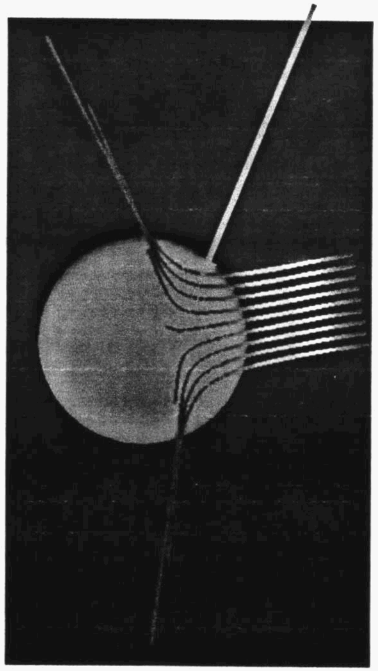


Full 3-D refraction and reflection

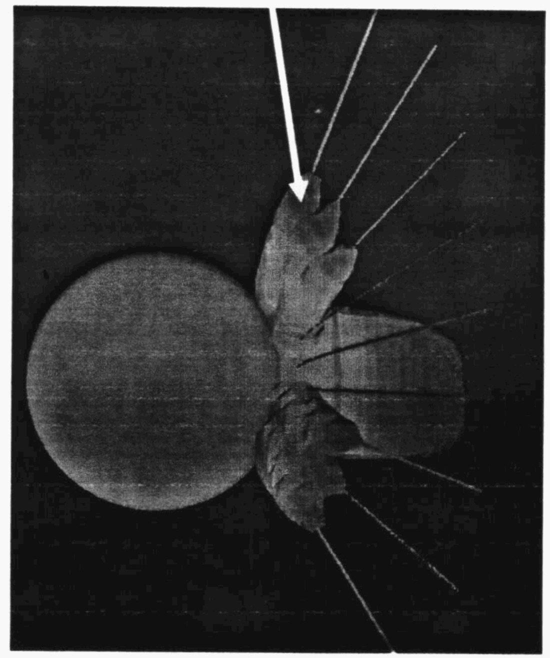
Rays_02_g

This approach renders a clear 3-D record of the dominant laser-matter interactions

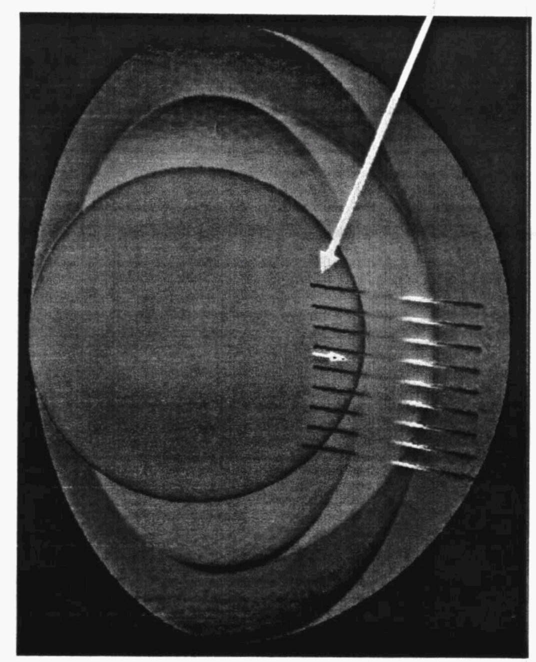
Refraction



Sourced deposition

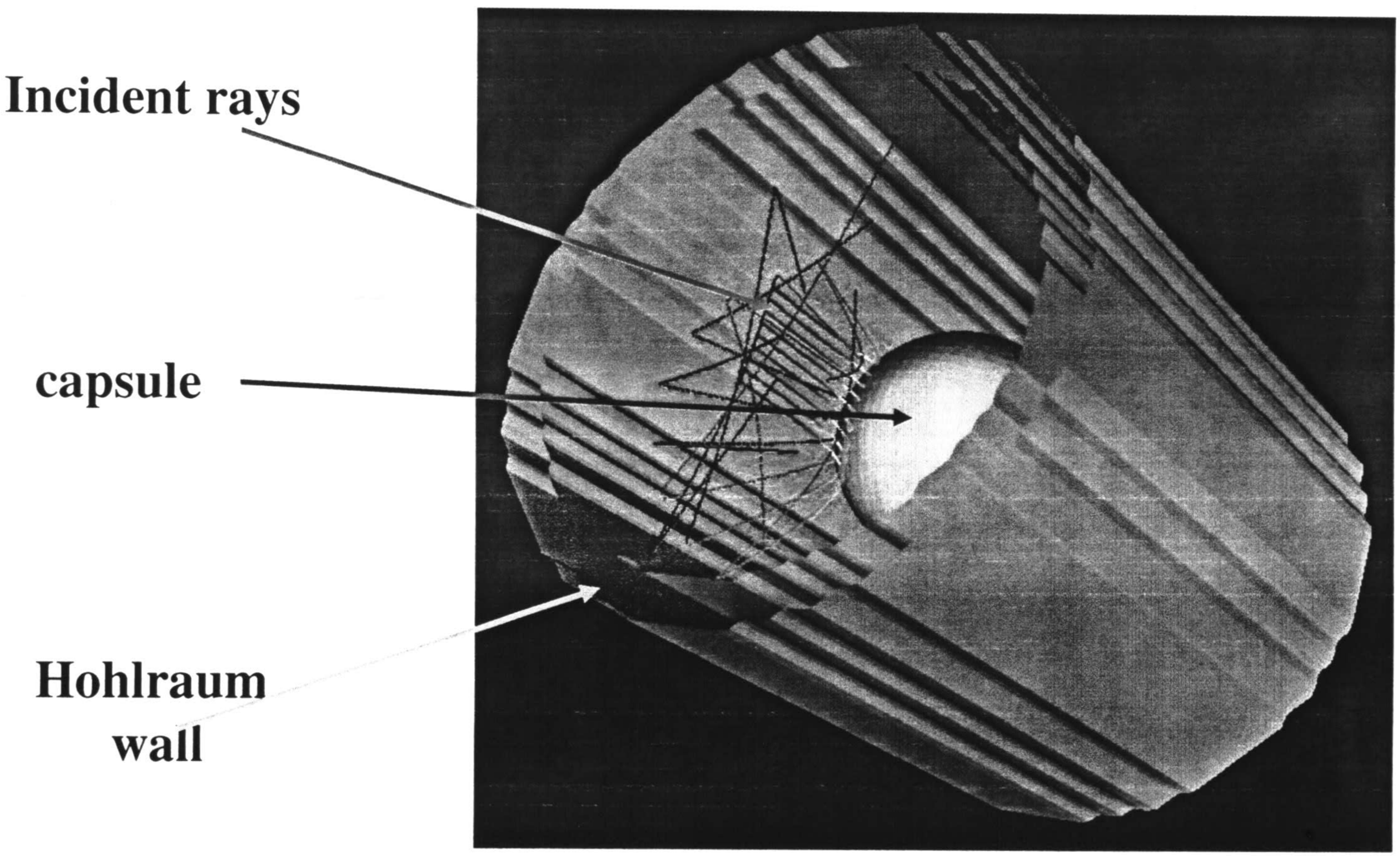


Dump-all at n_{crit}



Rays_02_h

Virtual Hohlraum applications show the rays losing energy in navigation between the capsule and the wall



Rays_02_i

The AMR modeling on a distributed computer architecture presents some unique challenges

• Neighboring cells can be refined. For these we use lower level data to provide the density gradients.

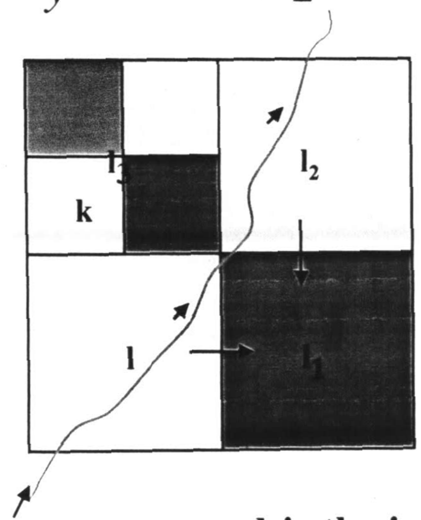
• The neighboring cells can "reside" on a different processor. For these we make a subroutine call to extract a copy of the data on the 'calling processor' via MPI. This allows continual rebalancing of the underlying mesh.

• Particles regularly cross onto new processors. We are considering a parallelization paradigm which places multiple copies of each ray-particle on all the processors, with a recorded update only on the processor containing the density gradient data for that particle.

Finding the neighboring cells for the density gradient calculations requires references to the neighboring indices

$$\nabla n_x = (n(l_1) - n(l)) / \Delta x$$

$$\nabla n_y = (n(l_2) - n(l_1)) / \Delta y$$



l_3 is the index for the upper cell in the y-direction

k is the 1st daughter of l_3

Ray trajectory
(8+ stops/cell)

$l_1$is the index for the right cell in the x-direction

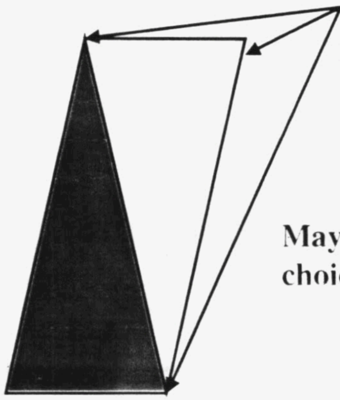
l_2=is the top index of l_1 in the y-direction

Rays_02_k

With AMR, an unstructured mesh and distributed memory we must choose our density gradient modeling optimally

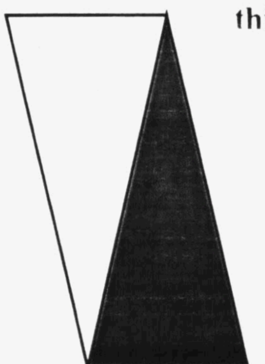
Lagrangian quads for earlier codes have been divided randomly

Three nodes define a plane which gives the density gradients.



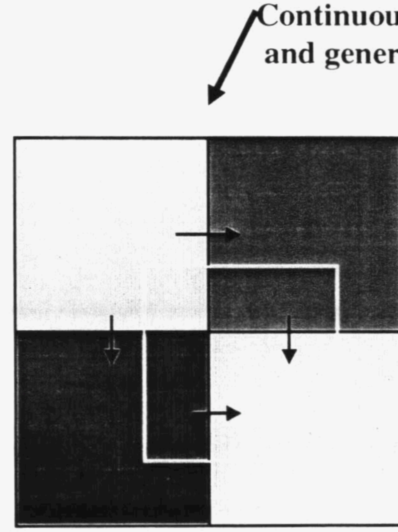
May get this choice,

or possibly this one.



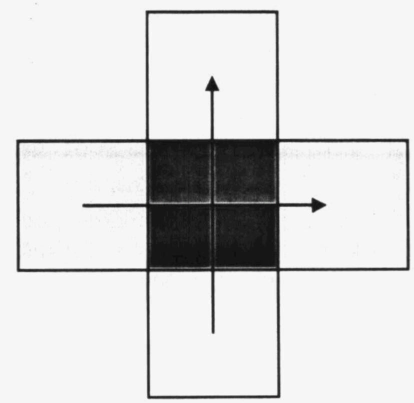
Could build 4 quads, but gradients change discontinuously at interfaces and 3D generalization is tricky.

Bi-linear interpolation is Continuous for gradients and generalizes readily to 3D.



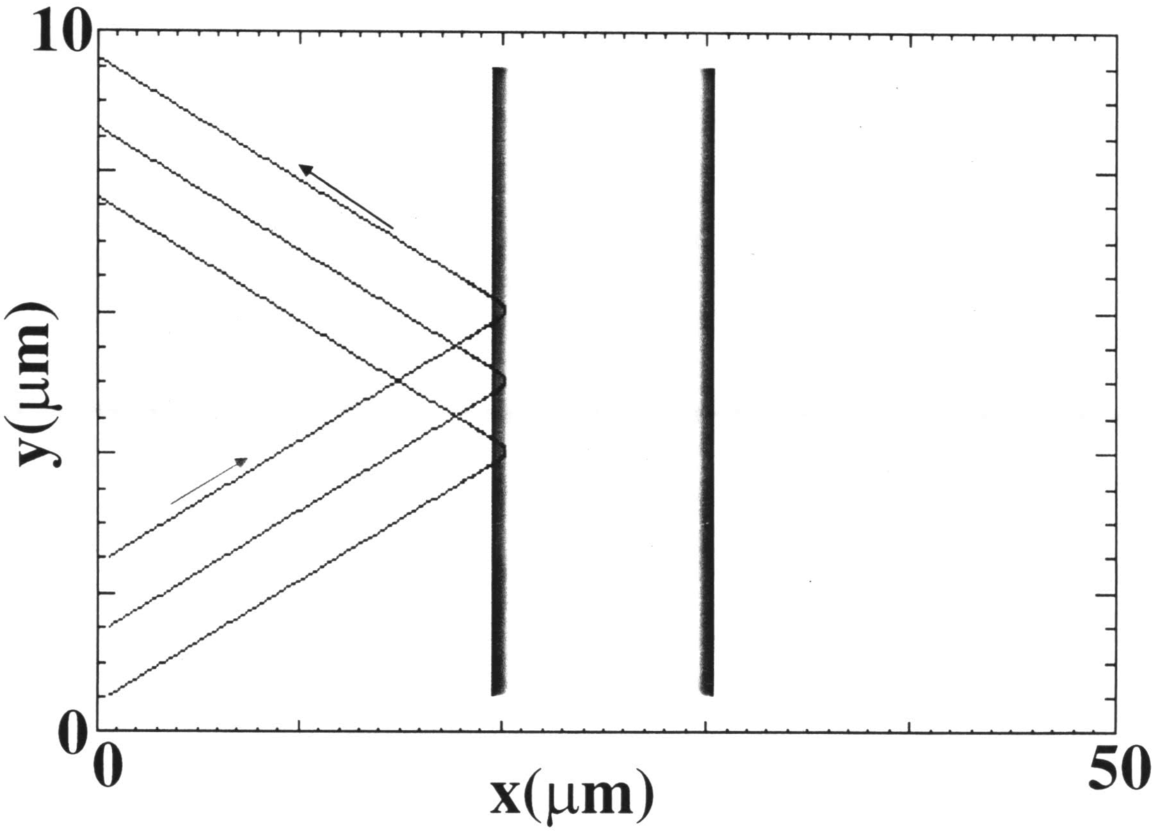
Work at same level with AMR, and interpolate if level changes.

Simplest gradient choices, to start.



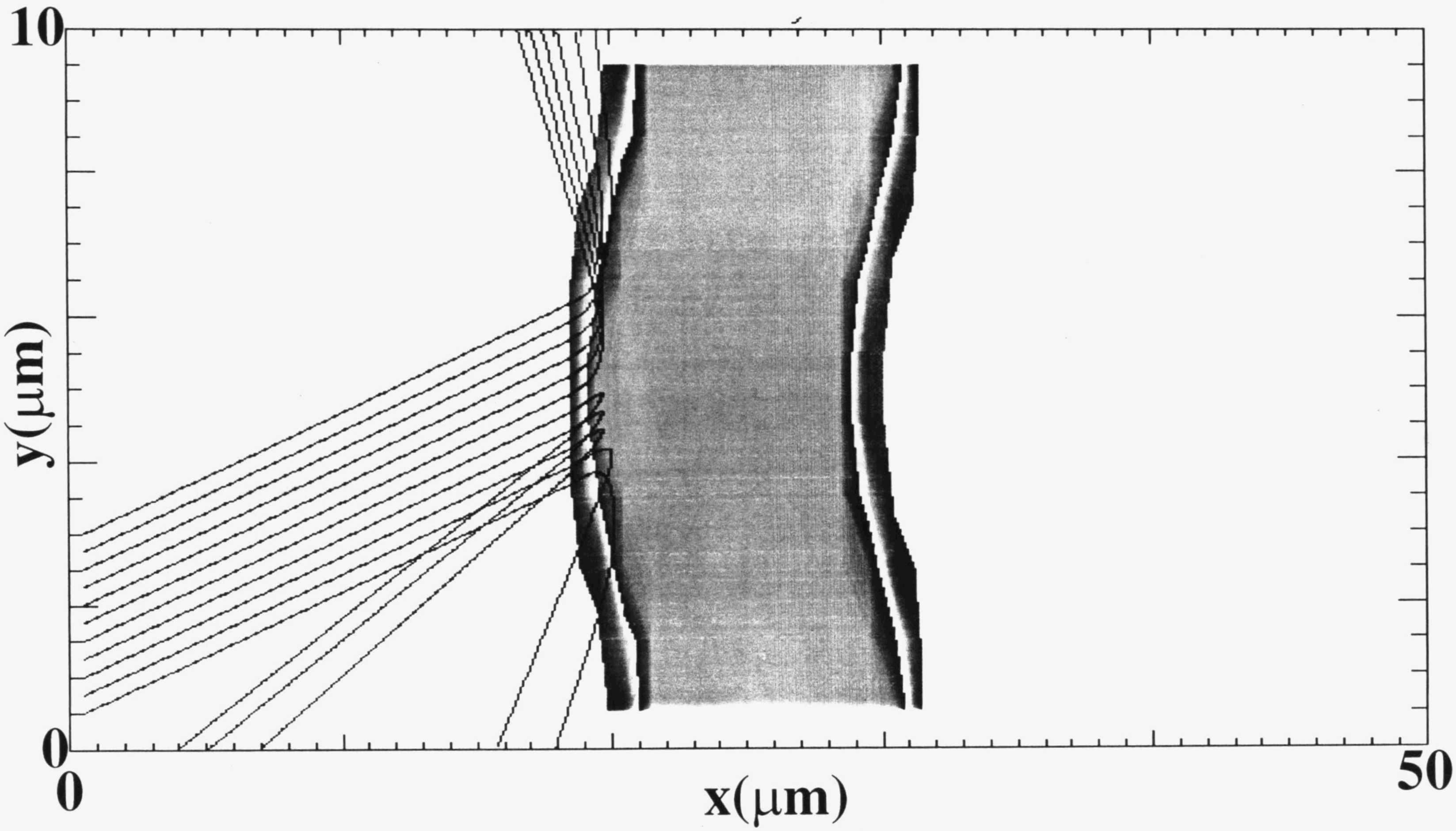
Rays_02_1

The simplest gradient choice worked well on a planar Problem with three levels of refinement for the density



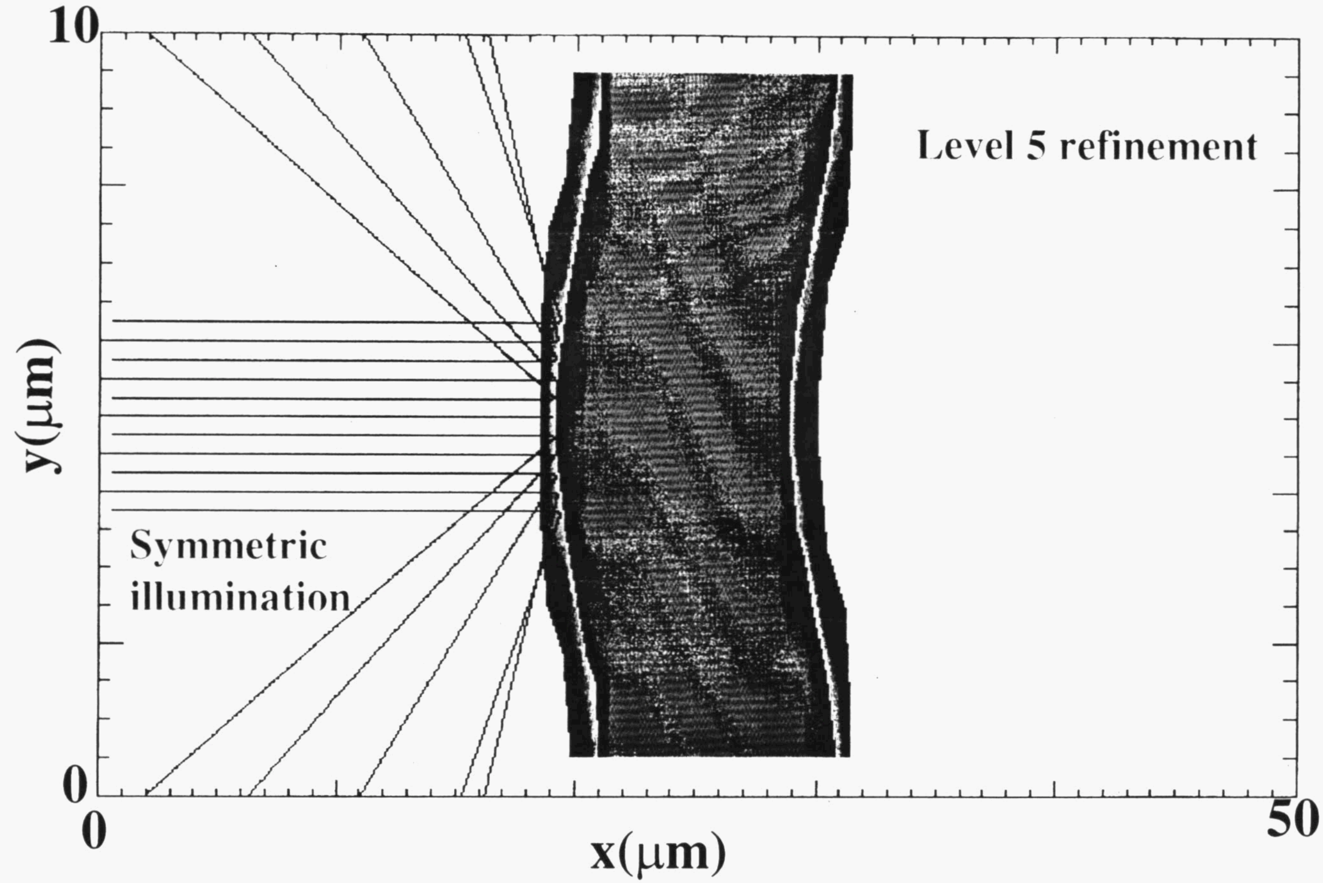
Rays_02_m

However, the simple density gradients give unphysical deflections near stair-stepped regions (levels 1 and to 3)



Rays_02_n

The area-weighted gradients give smooth, symmetric reflections which converge as refinement improves



Rays_02_o

Conclusions

- 3-D ray-trace for RAGE is on track; smoother gradients are being implemented

- The basic modeling is a straight-forward extension of old methods

- Refraction, inverse-bremsstrahlung, and dump-all work smoothly with conventional PIC volume-weighting (so far in our PC version)

- AMR extensions, including ray "particle" division and coalescence are planned

- Linear, frontal stringing of the ray particles offers potential smoothing