

** Todays Menu **

- Mail boxing
- Cost functions summary
- Grids

---

** Last Time **

- BSP trees
- Cost functions

---

** Mailboxing **

- Tag objects to avoid multiple tests with the same ray

---

** BSP Trees **

---

** Surface Area Heuristic **

\[
C = \frac{C_n \sum_n SA_n + C_l \sum_l SA_l + C_o \sum_i SA_i \cdot N_i}{SA_{root}}
\]

[MacDonald and Booth, 1990]
BVH SAH

\[
C = \sum_i \frac{A(i)}{A_{\text{root}}} C_{\text{box}} + \sum_i \frac{A(l)}{A_{\text{root}}} C_{\text{tri}}
\]

C is total cost of BVH according to SAH.

BVH SAH Construction Step

Use top down splitting approach. Create 2 (in this example) child boxes:

Minimize the following cost:

\[
C = 2 \cdot C_{\text{box}} + \frac{A_1}{A_c} N_1 C_{\text{tri}} + \frac{A_2}{A_c} N_2 C_{\text{tri}}
\]

Here \(A_1\), \(A_2\), and \(A_c\) are the area of child box 1, child box 2, and the current node box.

BVH Configurations

Fig. 2. Different BVHs for 4 triangles. The siblings are allowed to spatially overlap (unlike spatial subdivisions). Other possibilities include splitting to size 1 and 3 triangle list, and recursively splitting lists of 2 or 3 triangles.

What is this?

Spatial Subdivision

- Subdivide all space into cells
- Cells can be empty
- Fast traversal algorithms
- Objects are checked in order

Spatial Subdivision

- Octrees
  [Glassner, IEEE CG&A 1984]
- Grids
  [Fujimoto et al., "ARTs", 1986]
- Hierarchical grids
- Adaptive Grids
  [Klimaszewski and Sederberg, IEEE CG&A 1997]
**Grids**

- Building a grid
  Bounding box, resolution...
- Traversing a grid (3D-DDA)

**Building a Grid**

- Find grid bounding box
- Make grid cells (voxels)
- Insert triangles into grid

**Triangle-Voxel Intersection**

- Vertices outside box test
- Vertices inside box test
- Triangle-edge box intersection test
- Box diagonal triangle intersection test

**Triangle Lists**

```c
struct cell {
    int ntriangles;
    int *triangles;
}
```

**Grid Traversal**

Step through grid using 3D-DDA

- Find line through grid
- Test all cells along this line

**Grid Traversal 2D**

Intersect grid bounding box

```c
if (t_xmin > t_ymin) {
    // hit x side
    i = 0
    y = o_y + t_xmin * d_y
    j = ( y - y_min ) / (y_max - y_min) * n_y

    dtx = (t_xmax - t_xmin) / nx
    dty = (t_ymax - t_ymin) / ny
    t_xnext = t_xmin + dtx
    t_ynext = t_ymin + (j+1)*dty
}```

CSE168: Rendering Algorithms
Henrik Wann Jensen
Grid Traversal 2D cont’d

while (1) {
    cell_index = j*n_y + i
    intersect objects in cell[ cell_index ]
    if (t_xnext < t_ynext) {
        t_xnext += dtx
        i += 1
        if (i == n_x) break;
    } else {
        t_ynext += dty
        j += 1
        if (j == n_y) break;
    }
}

Hierarchical Grids

- Scenes have varying density of objects
- Use subgrids in dense areas

HG Statistics

No grid: 6321 intersection tests per ray (total = 371082127)
One grid: 44.86 intersection tests per ray (total = 26336575)
2-level grid: 12.05 intersection tests per ray (total = 7072774)

Adaptive Grids

- Scenes have varying density of objects
- Use compact grids in dense areas
- Grid merging

Coherent Ray Tracing of Grids
Coherent BVH Ray Tracing

Fig. 1. Screenshots from an animated 180,000 triangle scene with moving dragonsfly, fairy, and planes. At 1024 x 1024 pixels the animated scene is ray traced at roughly 5.7 frames per second on a dual-2GHz Opteron desktop PC including shadows and texturing.

Dynamic BVH Ray Tracing

Fig. 5. When the objects move, a BVH can keep the same hierarchy and only needs to update the bounding volumes. Though the new hierarchy may not be as good as the old one, it will always be correct. For all but some worst-case examples, even worse deformations do not significantly deteriorate the BVH quality. By considering different primitive positions during the build, we can also make sure that the chosen BVH will be reasonably good for scenes in which a good hierarchy is not apparent from a single pose.

Ray Tracing Large Models

Source: 3D data provided by and used with permission of the Boeing Company.

Boeing 777 - 360 million triangles

Next time

Texture mapping, environment mapping, bump mapping