Recall: Adaptive Subdivision

- Subdivide the mesh at places where the radiosity gradient is high
- Use the same mesh for all interactions
- But, for any given patch, a fine mesh may not be needed for all interactions
  - A patch may contain the shadow boundary of one light source, but not others
  - Gathering from diffuse reflectors (e.g. walls) needs less resolution

Hierarchical Meshing

- Build hierarchy of elements for each patch
- Record interactions between elements at appropriate levels
- Essentially replaces blocks in the form factor matrix with average values

Building the Hierarchy

- Start with "top level" patches
- Estimate the error incurred by that transfer. Use:
  - Form factor estimate: $\tilde{F} = \frac{\cos\theta \cos\theta'}{\pi r^2}$
  - Radiosity $\times$ Form factor
  - Importance
- If error too large, break one element and recurse

Hierarchical Solution

- For each Gauss-Seidel iteration
  - Gather energy to a patch along all the links
  - Update radiosity values for each node in the patch hierarchy
    - Push radiosity down from parents to children
    - Pull radiosity up from children to parents
- Cost proportional to number of links, which is O(N) where N is number of leaves

Brightness-weighted refinement

- Subdivide based on an estimate of energy transferred
- Need radiosity estimates, which we don't know beforehand:
  - Start with high error threshold, and get approximate solution
  - Refine hierarchy using radiosity estimates
  - Repeat with successively lower thresholds

Computing Form Factors

- Only need form factors for each link
  - Hemicube does more work than necessary, because all elements must be projected for all other elements
  - Use patch-patch computations (more later)
- Error control
  - More and more subdivision not necessarily good, because errors accumulate in push-pull, and element boundaries can cause artifacts

Importance

- For a single image, only the visible surfaces are directly important
  - Don't care what's "down the hall"
- Surfaces illuminating visible surfaces are also important, and other surfaces illuminating those surfaces…
- *Importance propagates backwards from the "primary" sources of importance*

Solving for Importance

$$I_i = R_i + \sum_{j=1}^{N} \rho_j F_{ji} I_j$$

- R_i is the inherent importance of a given element
- When radiosity is gathered, importance is shot
- Use *BIF* as error estimate in determining subdivision
- Note: must use different push-pull operations for importance

Form Factor Errors

- Ref: Baum, Rushmeier, Winget 89
- Hemicube replaces the form factor double integral with a point to area form factor, assuming small changes over the patches
  - Violated if patches are close
  - Violated if visibility varies across patches
- Limited hemicube resolution results in aliasing
  - Small patches may not be seen, or may "beat"

Decoupling Hemicube Duties

- Hemicube computes:
  - Visibility
  - Sub-area form factors
- Replace one of both with alternate methods
  - Sub-area form factors from analytical methods
  - Visibility from ray-casting
  - Sub-area from factors from disc approximation

Shooting to Vertices

- "Standard" progressive radiosity shoots from points to areas (due to hemicubes)
- Better to shoot from area to points:
  - Progressive radiosity really needs F_{ji}
  - Less aliasing
  - Radiosity is needed at vertices for display
  - Greater control over receivers

Ray-casting

- Aim: compute $F_{y,P_i} = \int_{x \in P_i} \frac{\cos\theta \cos\theta'}{r^2} V(x, y)\, dx$
- Cast rays from y to sub-patches of P_i:
  - Ray determines (constant) visibility for each sub-patch
- Then:

$$\hat{F}_{y,P_i} = \sum_{k=1}^{n_i} V_k \delta F_k$$

$$\delta F_k = \int_{x \in P_i^k} \frac{\cos\theta \cos\theta'}{\pi r^2} dx$$

Delta Form Factors

- Several possibilities
  - Assume constant (can be very bad)
  - Use analytic methods (costly)
  - Use disc approximation
    - Assume sub-patch is a small disc
    - Form factor for point-disc is known

$$\delta F_k = \frac{A_i^k \cos\theta \cos\theta'}{A_i^k + \pi r^2}$$

Enhancements

- Sample points on source can be varied
  - More samples at high gradients
- Location of receiving points can be varied
  - Points on all surfaces - no chance to miss surfaces entirely
- Arbitrary precision with more rays

More ways to improve

- Ray casting is expensive
- We are still assuming constant radiosity across patches, and computing meshes on the fly
  - Allow different basis functions (non constant)
  - Better meshing