Radiosity

CGG MFF UK Praha

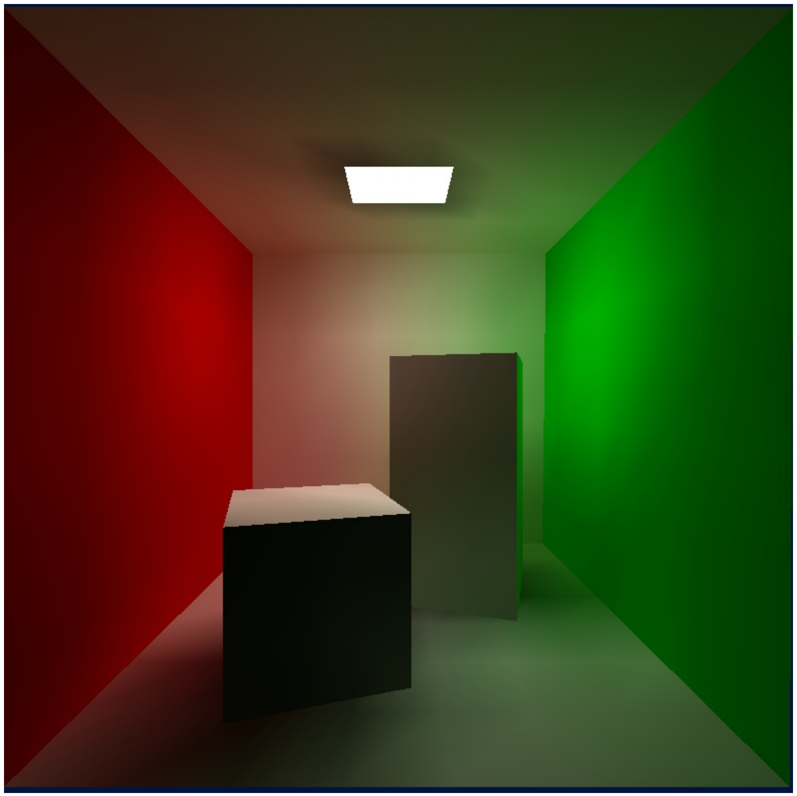
pepca@cgg.mff.cuni.cz

http://cgg.mff.cuni.cz/~pepca/

Global illumination, radiosity

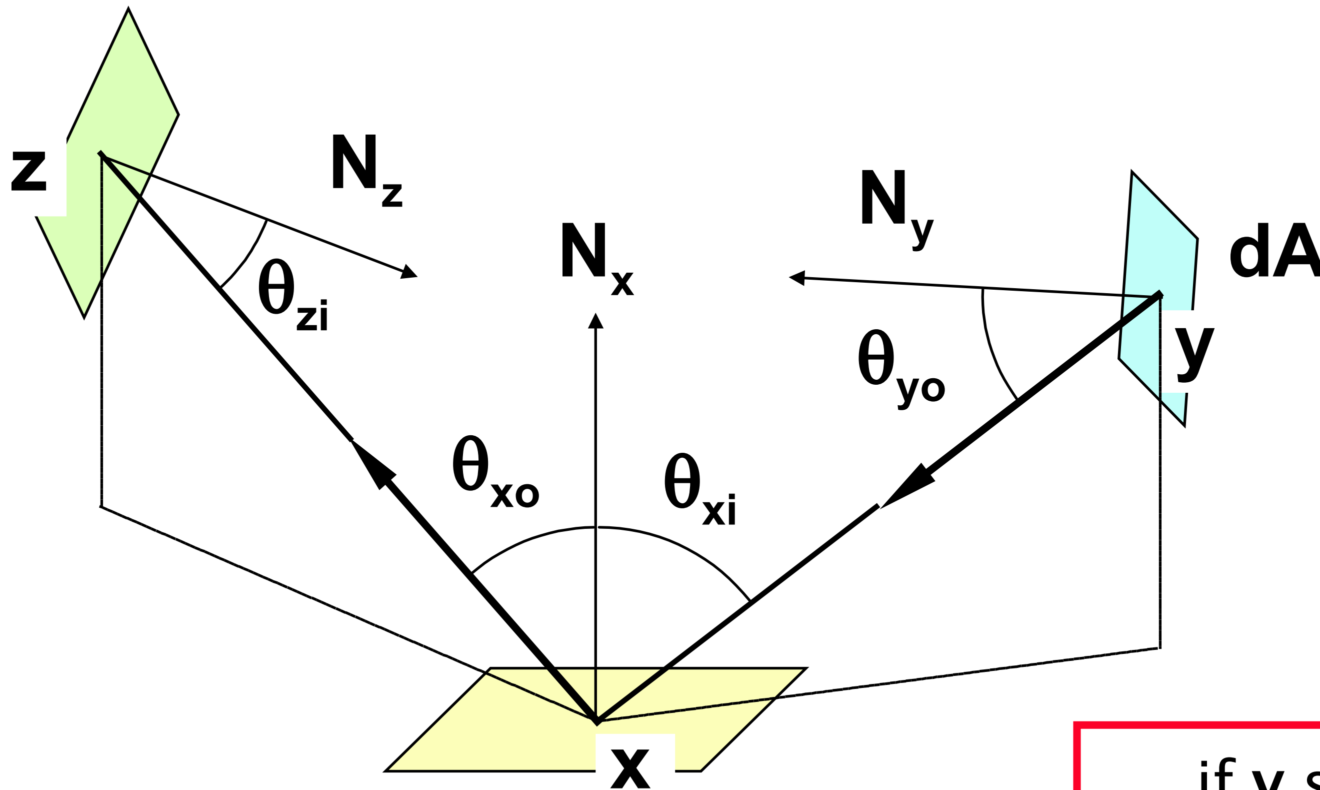
- based on **physics**
  - energy transport (light transport) in simulated environment
  - first usage of radiosity in image synthesis: Cindy Goral (SIGGRAPH 1984)
- radiosity is able to compute **diffuse light**, secondary lighting, ..
- basic **radiosity** cannot do sharp reflections, mirrors, ..
- time consuming computation
  - Radiosity: light propagation only

Radiosity – examples



© David Bařina (WiKi)

Reflected light



if y sees x

Terminology: $\underline{L(y,x)} = L_o(y, x-y) \boxed{= L_i(x, y-x)}$

$$\underline{f(y,x,z)} = f\big(x, (y-x) \to (z-x)\big)$$

Radiosity equation

- assumption – **ideal diffuse (Lambertian)** surface:
  - **BRDF** is not dependent on incoming/outgoing angles
  - outgoing radiance $\mathbf{L(y,\omega)}$ independent on direction $\boldsymbol{\omega}$

$$L(x, z) = L_e(x, z) + \rho(x) / \pi \cdot \int_S L(y, x) \cdot G(y, x) \cdot V(y, x)\, dA$$

$$L(x, z) = B(x) / \pi, \quad L_e(x, z) = E(x) / \pi$$

$$\boxed{B(x) = E(x) + \rho(x) \cdot \int_S B(y) \cdot \frac{G(y, x) \cdot V(y, x)}{\pi}\, dA}$$

Discrete solution

$$\boxed{B(x) = E(x) + \rho(x) \cdot \int_S B(y) \cdot g(y, x)\ dA}$$

where $g(y, x) = \dfrac{G(y, x) \cdot V(y, x)}{\pi}$

- solution **B** is infinit-dimensional
- discretization of the task:
  - **Monte-Carlo** ray-tracing (dependent on camera)
  - classical **radosity** (finite/boundary elements FEM)

General radiosity method

❶ object surfaces divided into set of **elements**

❷ definition of **knot points** on elements
  – **radiosity** will be computed there

❸ choice of an **approximation method** and error metric
  – basis functions for convex blend from knot points

❹ **coefficients** of linear equation system
  – “form-factors”

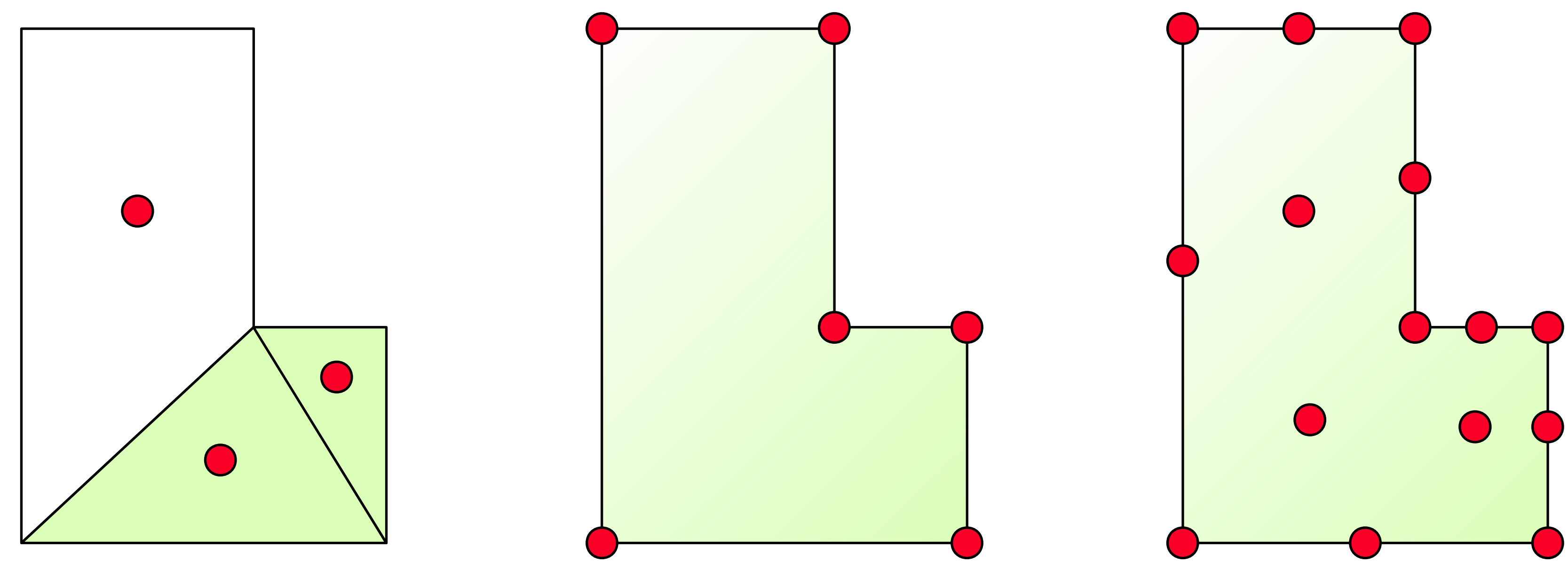
General radiosity method

5. solution of **linear equation system**
   - result: radiosity in knot points

6. reconstruction of values on **whole surfaces**
   - linear blends using basis functions and knot point radiosities

7. **rendering** of results (arbitrary view)
   - light is proportional to radiosity

Radiosity approximation



constant (knots in centers)

bilinear (knots in vertices)

quadratic (more knots in centers..)

Constant elements

- on every **element A_i constant** reflectivity is assumed ρ_i, radiosity **B** – average of **B(x)**:
  - terminology: ρ_i, B_i for $i = 1 .. N$

average over area A_i

$$B(x) = E(x) + \rho(x) \cdot \int_S B(y) \cdot g(y,x)\ dA$$

$$B_i = E_i + \rho_i \cdot \frac{1}{A_i} \int_{A_i} \underbrace{\left[\sum_{j=1}^{N} B_j \int_{A_j} g(y,x)\ dA_j \right]} dA_i$$

radiosity received in point **x** (lying on A_i)

Basic radiosity equation

switching sum and integral:

$$B_i = E_i + \rho_i \cdot \sum_{j=1}^{N} B_j \cdot \boxed{\frac{1}{A_i} \int_{A_i} \int_{A_j} g(y, x) \; dA_j \; dA_i}$$

geometric term – **form factor F_{ij}**
(part of energy irradiated from $\mathbf{A_i}$ received directly by $\mathbf{A_j}$)

$$\boxed{B_i = E_i + \rho_i \cdot \sum_{j=1}^{N} B_j \, F_{ij} \quad \left[\frac{W}{m^2} \right]}$$

System of linear equations

$$\underline{B_i} - \rho_i \cdot \sum_{j=1}^{N} \underline{B_j}\, F_{ij} = E_i \qquad i = 1..N$$

$$\begin{bmatrix} 1-\rho_1 F_{1,1} & -\rho_1 F_{1,2} & .. & -\rho_1 F_{1,N} \\ -\rho_2 F_{2,1} & 1-\rho_2 F_{2,2} & .. & -\rho_2 F_{2,N} \\ .. & .. & .. & .. \\ -\rho_N F_{N,1} & -\rho_N F_{N,2} & .. & 1-\rho_N F_{N,N} \end{bmatrix} \underline{\begin{bmatrix} B_1 \\ B_2 \\ .. \\ B_N \end{bmatrix}} = \begin{bmatrix} E_1 \\ E_2 \\ .. \\ E_N \end{bmatrix}$$

vector of unknown vars $[B_i]$

Quantities

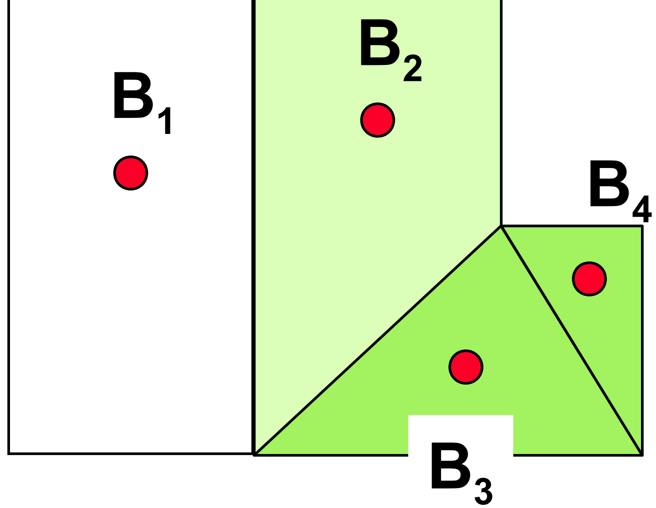
- B_i.. unknown **radiosity** values of individual faces
  – when calculating color, we need to calculate radiosity for all required wavelengths (color components - e.g. **R**,**G**,**B**)
- E_i .. **own** (emitted) **radiosity** (**R**,**G**,**B**)
- ρ_i .. **reflection coefficients** of an object (**R**,**G**,**B**)
- F_{ij} .. **form-factors**
  – they depends only on scene geometry

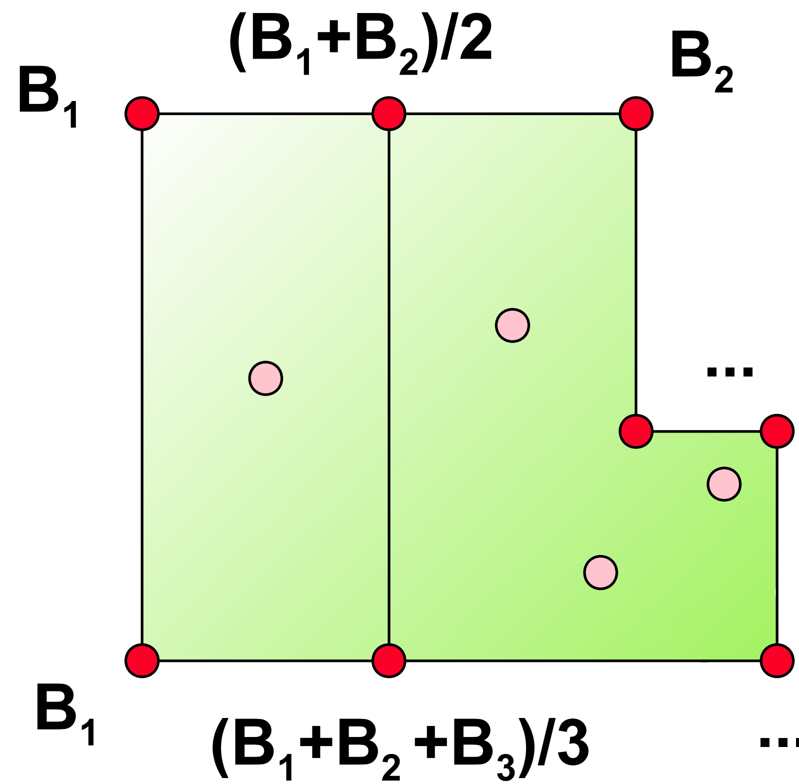
System of linear equations

- for **planar (convex) surfaces**: $F_{ii} = 0$
  - the diagonal contain only unit values
- **nondiagonal items** are usually very small (abs value)
  - matrix is “diagonally dominant”
  - ⇒ system is stable and can be solved by **iterative methods** (Jacobi, Gauss-Seidel)
- for **light change (light sources)** $[E_i]$ system needs not to be fully re-computed, only reverse phase could be done

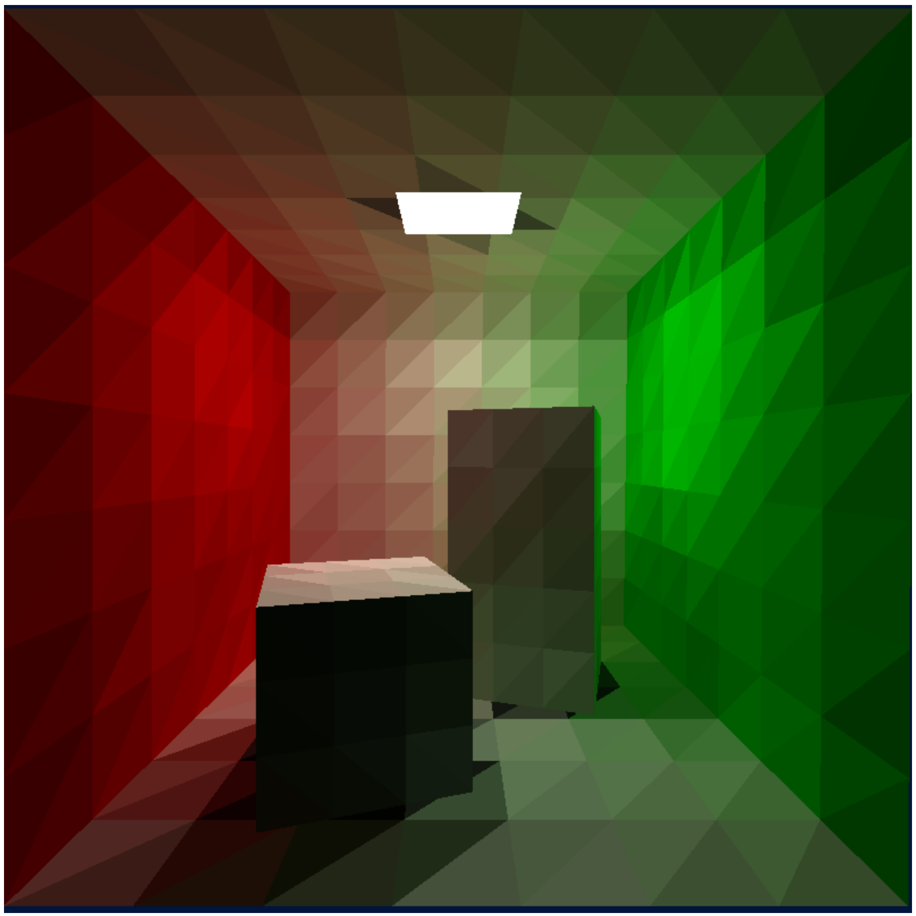
Radiosity to vertices

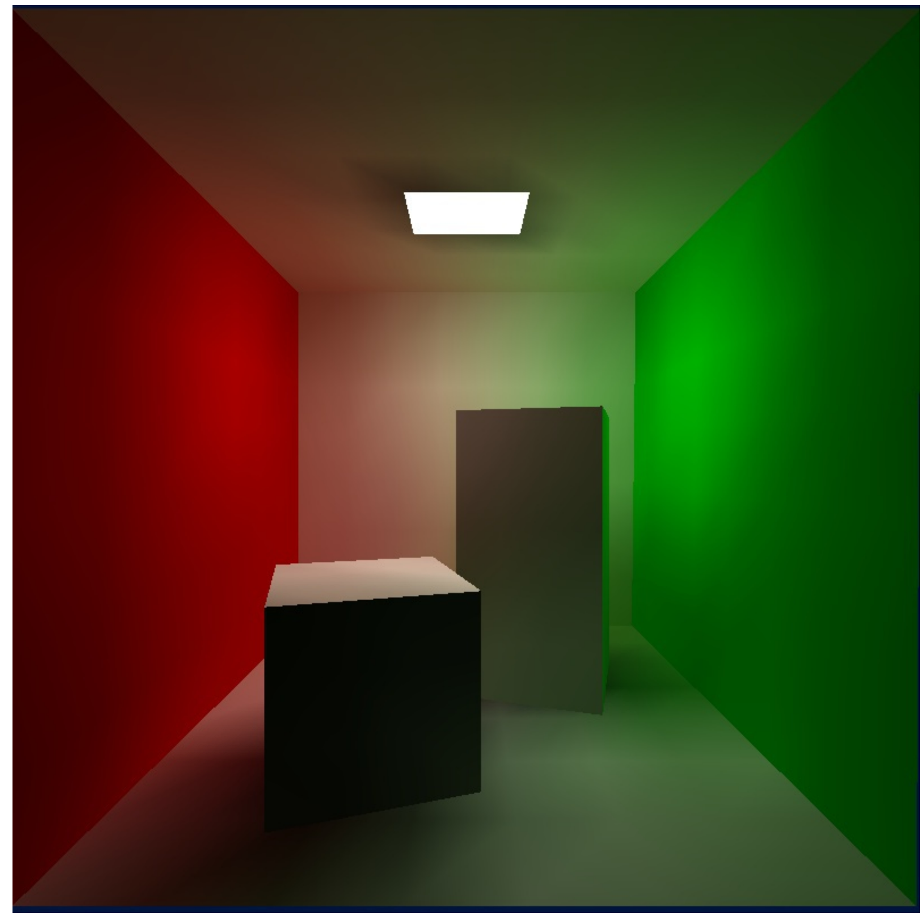
Even in constant element approach usage of some color interpolation method is recommended (**Gouraud**)





Linear color interpolation





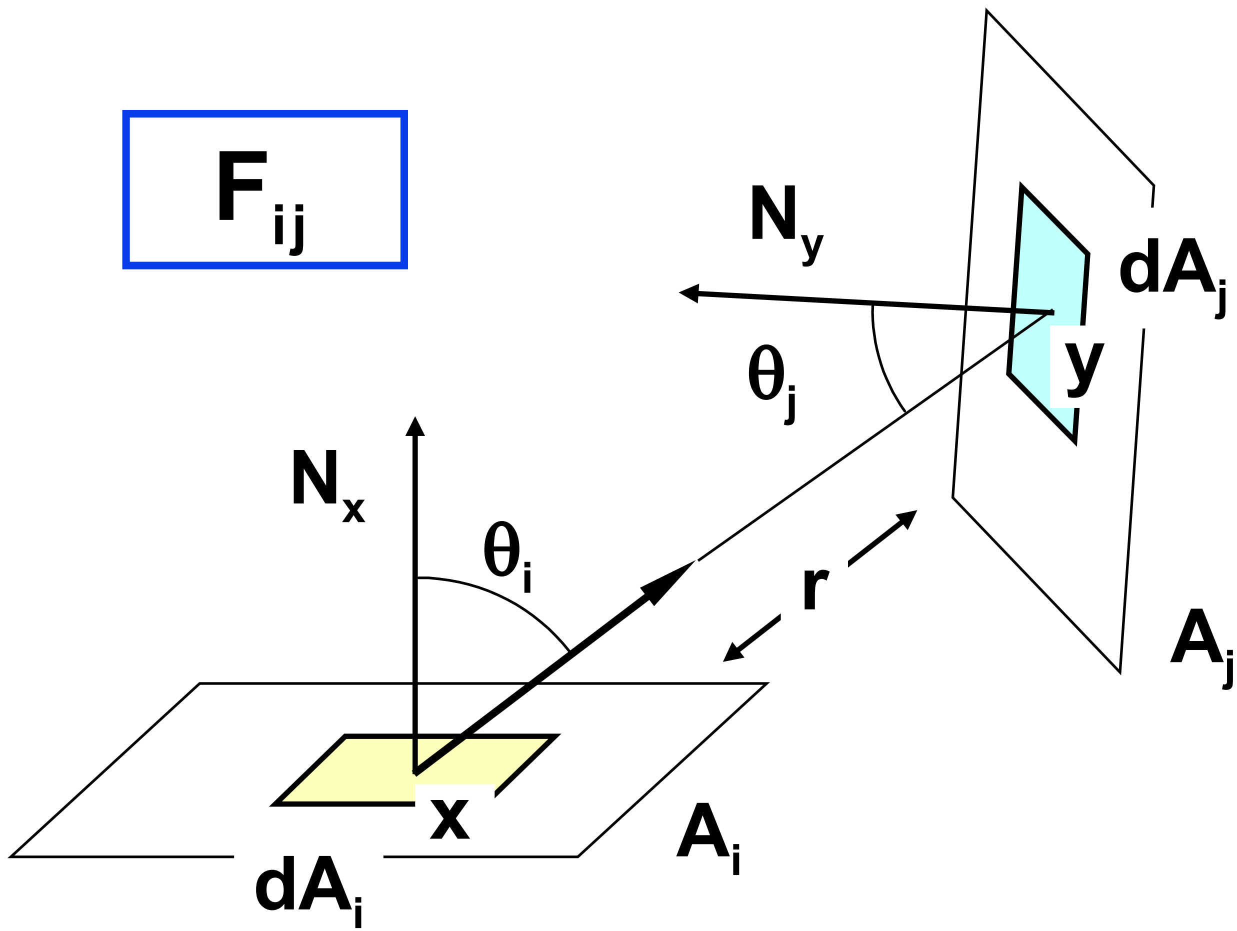
Form-factors

Form-factor F_{ij}

- It indicates the proportion of energy emitted from the surface **i** which will hit the the surface **j**
  - key value when creating a system of linear equations (searching for individual area radiosities)
  - first calculation (physics): Lambert 1760
- it depends only on the **geometry of the scene**
  - distance, inclination and slope of the areas
- $\mathbf{F_{ij}}$ is a dimensionless number from the interval $\langle \mathbf{0,1} \rangle$
  - for a convex polygon **i** is $\mathbf{F_{ii} = 0}$

Form-factor

$\boxed{F_{ij}}$



Form-factor

Radiosity equation (with constant elements):

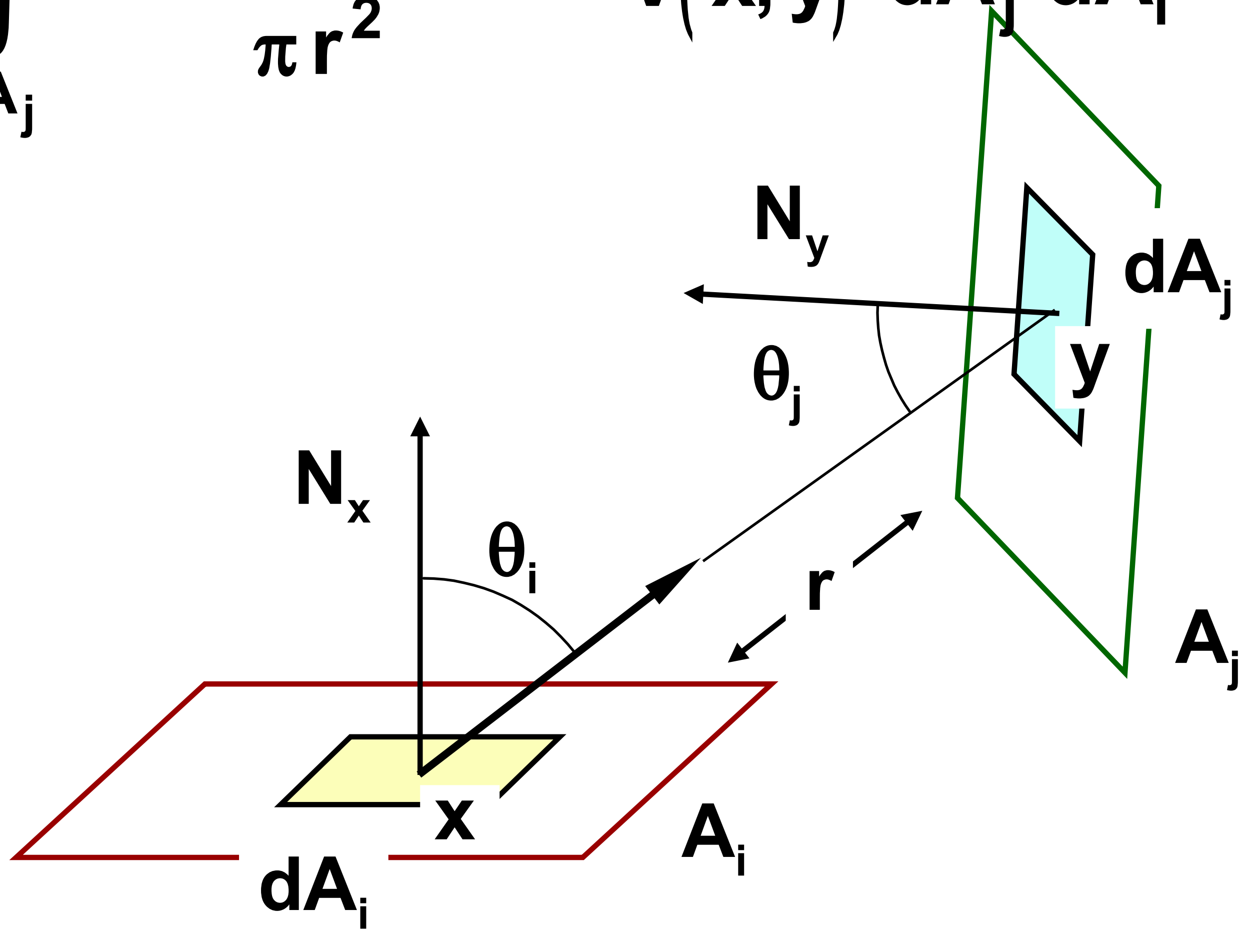
$$B_i = B_{e,i} + \rho_i \cdot \sum_{j=1}^{N} B_j \cdot \boxed{\frac{1}{A_i} \int_{A_i} \int_{A_j} G(y,x)\; dA_j\; dA_i}$$

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} G(y,x)\; dA_j\; dA_i =$$

$$\boxed{= \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\theta_i \cdot \cos\theta_j}{\pi \left\| x - y \right\|^2} \cdot V(x,y)\; dA_j\; dA_i}$$

Form-factor

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos\theta_i \cdot \cos\theta_j}{\pi r^2} \cdot V(x, y)\ dA_j\ dA_i$$



Calculation of form-factors

❶ **analytical methods**

❷ **numerical methods**

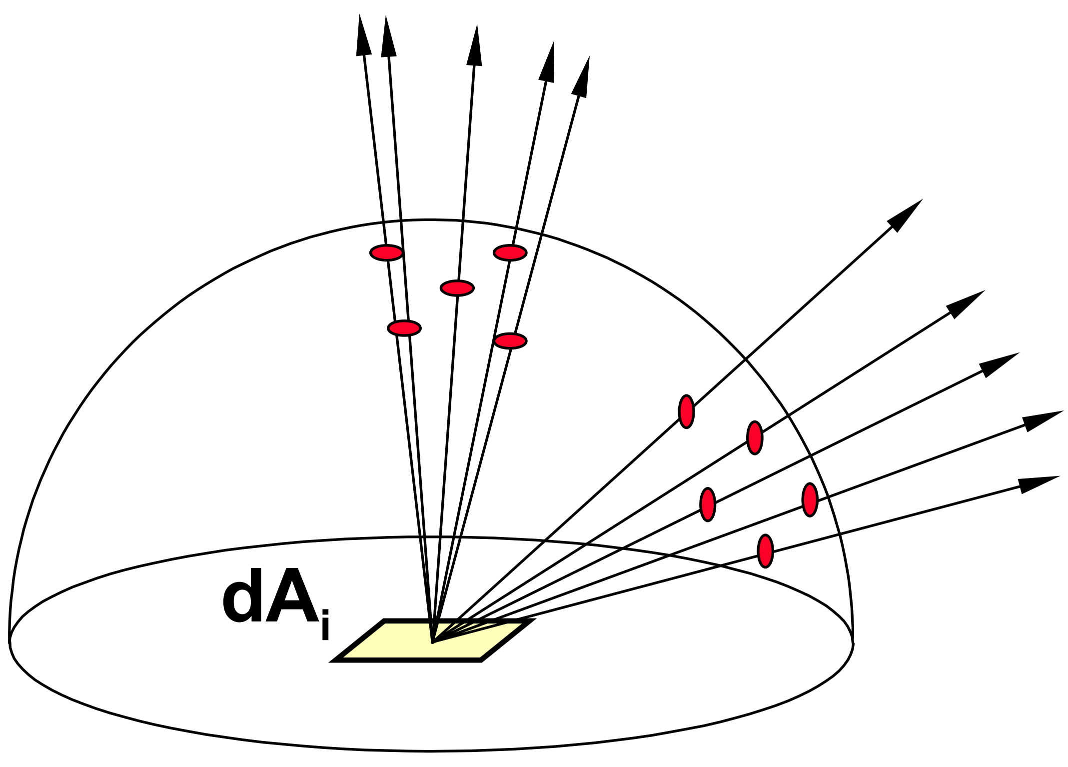
– hemicube (Nusselt analogue), projection into one plane, curve integral

❸ numerical **stochastic methods** (Monte-Carlo)

– sampling of a spatial angle or an area that receives energy

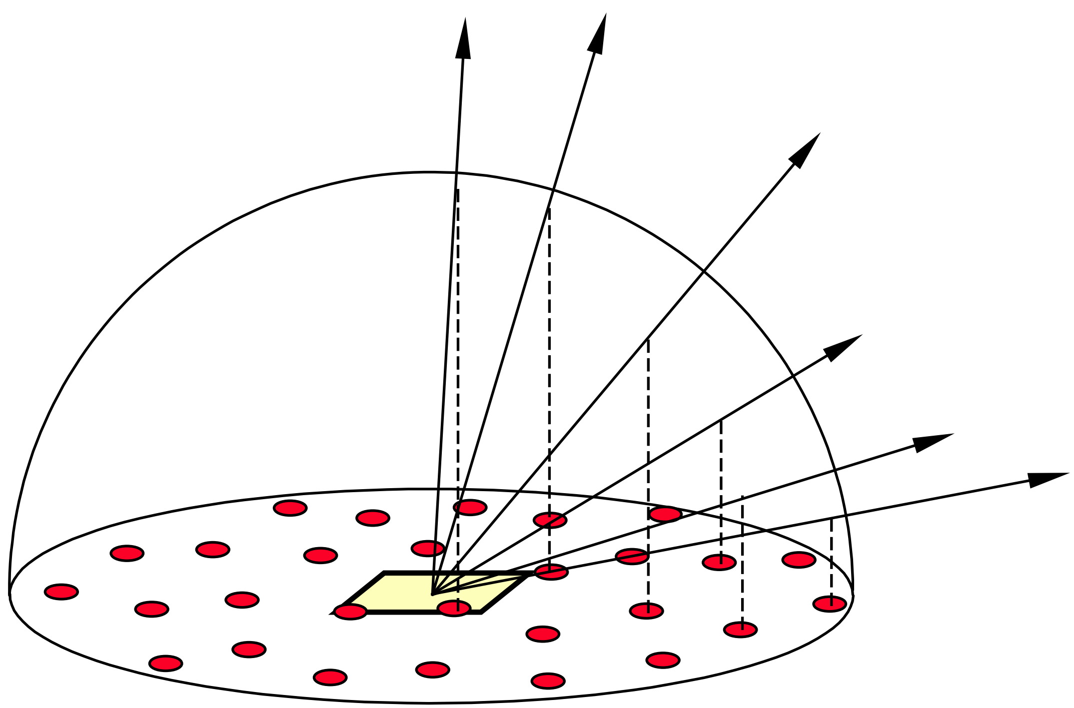
Sampling on the hemisphere

Uniform sampling of the solid angle with **cos** θ_k pdf

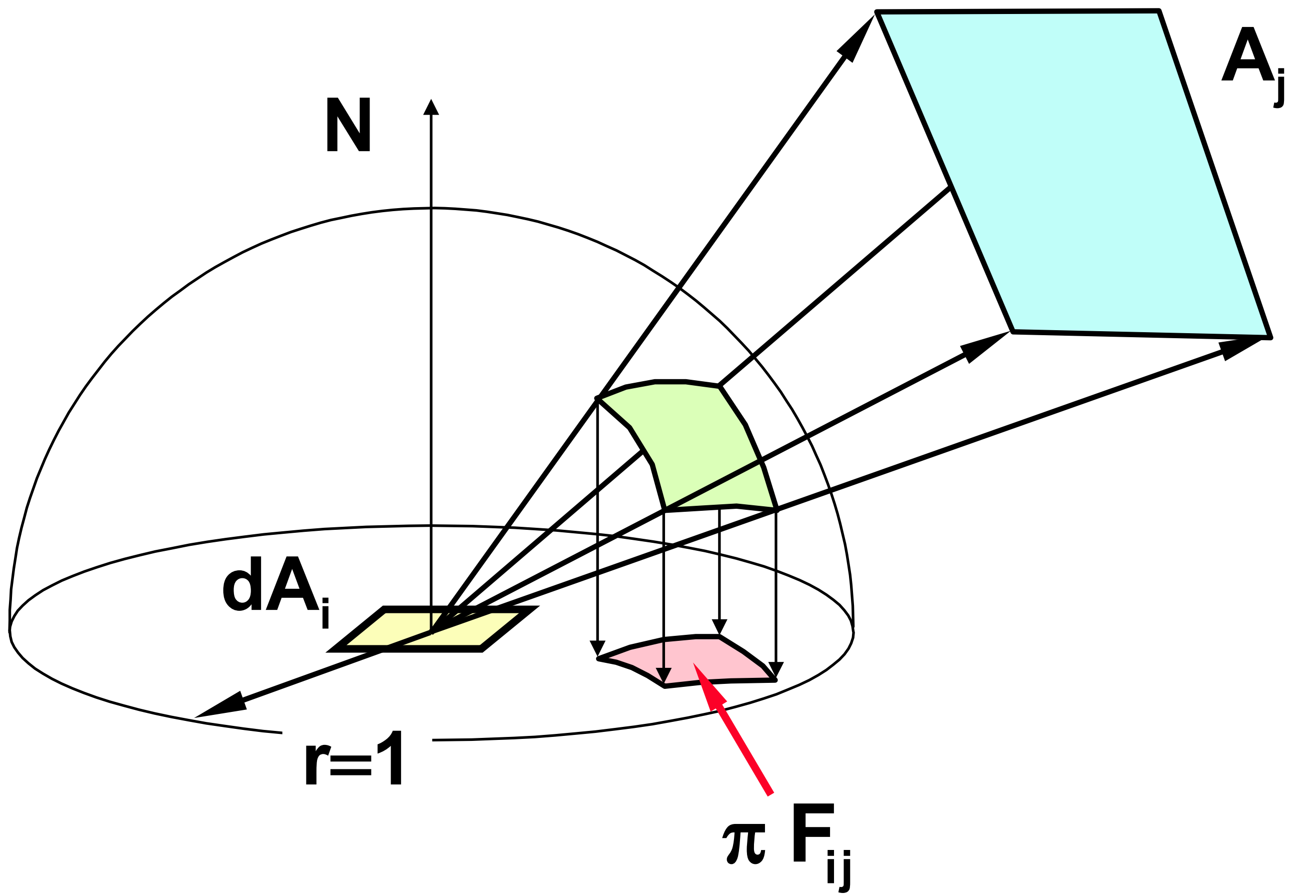


Sampling on the source

Non-uniform sampling of the solid angle
all rays have the same <u>importation</u> !



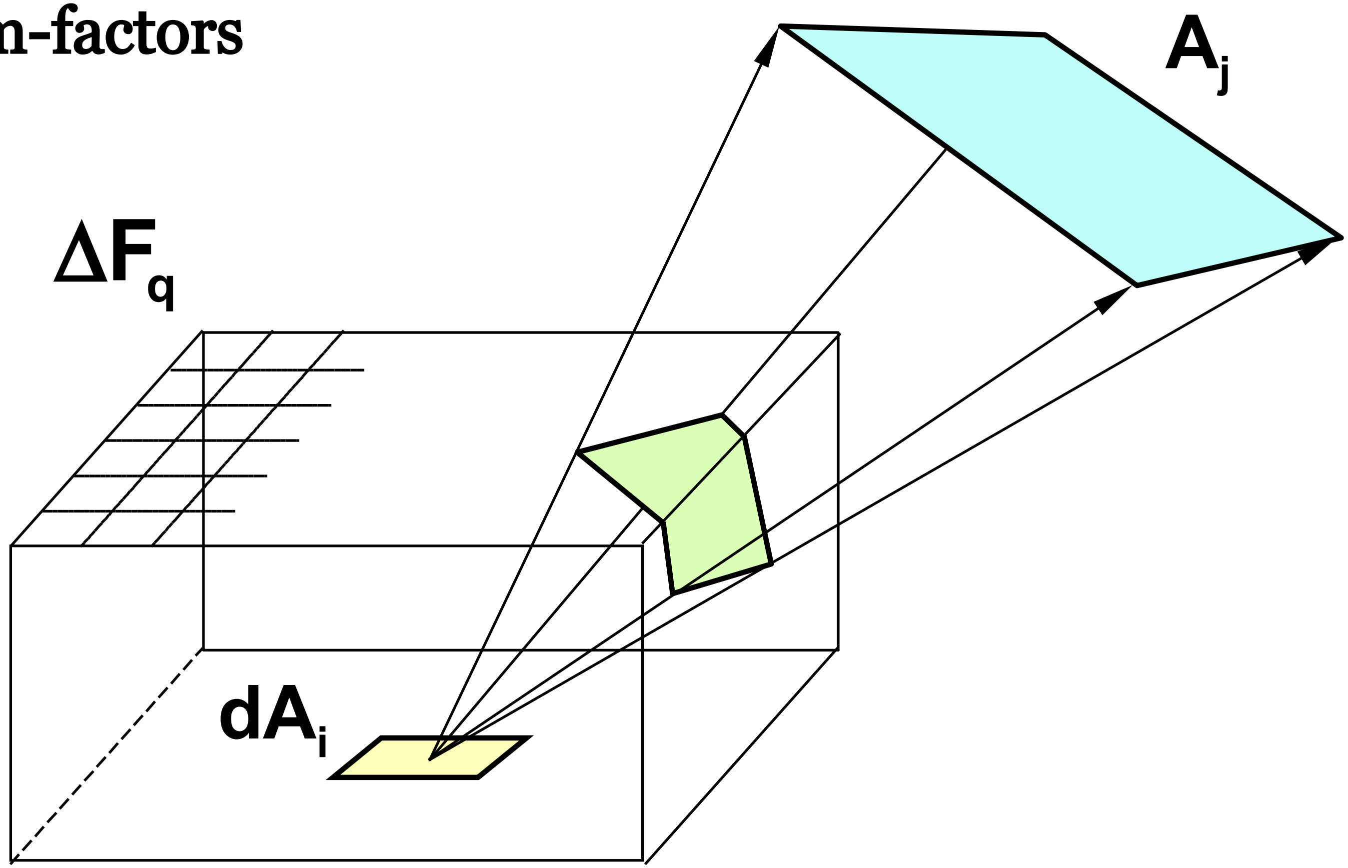
Nusselt analogue



Hemicube

Regular cell network:

delta form-factors



Hemicube

- calculation of all $\mathbf{F}_{ij}$ for given $\mathbf{i}$
  - we project all other faces $\mathbf{A}_j$ on the hemicube built around $\mathbf{dA}_i$
- we calculate the visibility of the individual faces on the surface of the hemicube (Z-buffer method)
- the surface of the hemicube is divided into a **regular network of cells** $\mathbf{C}_q$
  - for each cell we have calculated delta form-factor $\Delta\mathbf{F}_q$ in advance

Hemicube

- configuration factor $\mathbf{A_j}$ is estimated by cells that were covered by its projection:

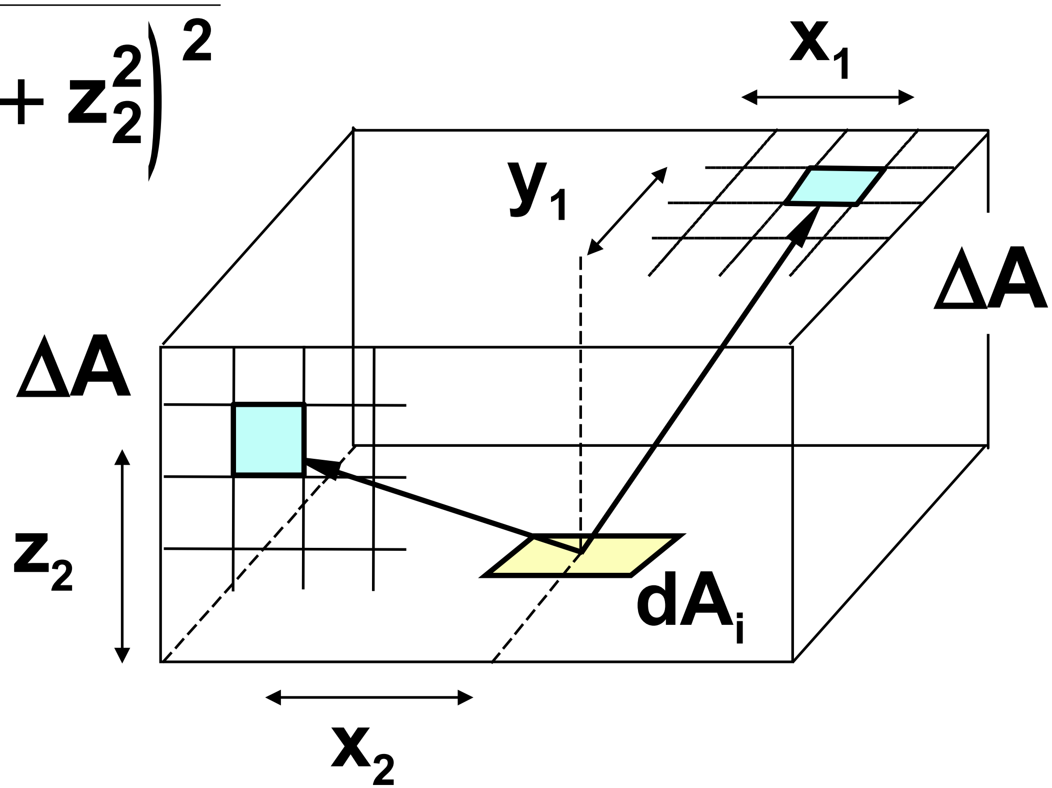
$$\mathbf{F_{ij}} \cong \sum_{\mathbf{q} \in \mathbf{J}} \Delta \mathbf{F_q}$$

- dividing the cube affects the accuracy of the estimation of the configuration factors
  – in practice from **64×64** to **2k×2k** cells were used

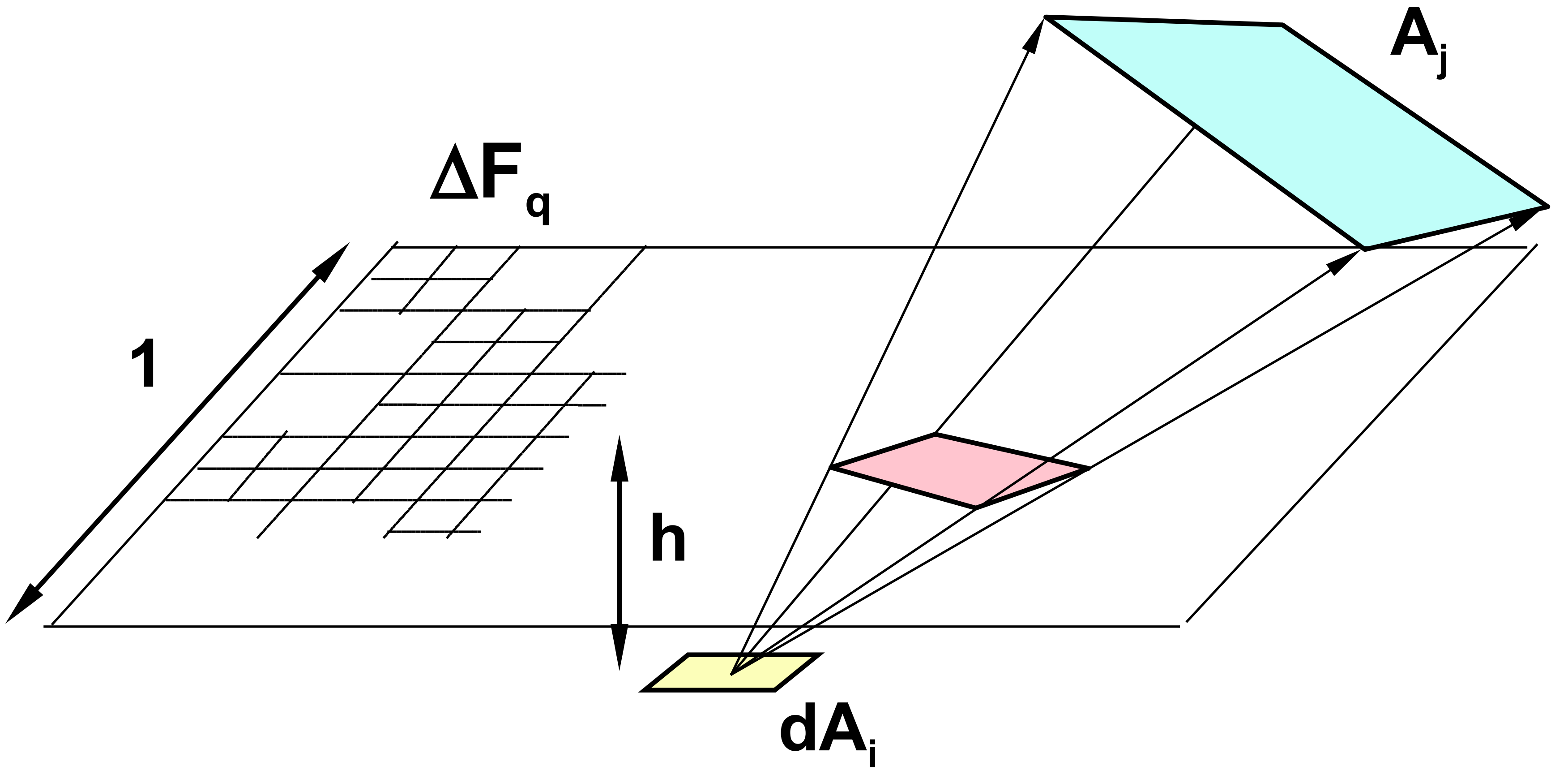
Delta form-factors

$$\Delta F_1 = \frac{\Delta A}{\pi \cdot \left(x_1^2 + y_1^2 + 1 \right)^2}$$

$$\Delta F_2 = \frac{z_2 \cdot \Delta A}{\pi \cdot \left(x_2^2 + 1 + z_2^2 \right)^2}$$



Sillion hemiplane method



Hemiplane method

- **faster implementation** (projection, trimming)
  - part of the spatial angle is neglected
  - the height of the projection plane should be maximum **0.1**
- visibility is calculated by **divide and conquer** method
  - Warnock algoritm analogue
  - adaptive projection plane division $\Rightarrow$ better efficiency
- **delta form-factor** are precalculated for different levels of division

Numerical solutions of the radiosity system of equations

The system of linear equations

$$\underline{B_i} - \rho_i \cdot \sum_{j=1}^{N} \underline{B_j}\, F_{ij} = E_i \qquad i = 1..N$$

$$\begin{bmatrix} 1-\rho_1 F_{1,1} & -\rho_1 F_{1,2} & .. & -\rho_1 F_{1,N} \\ -\rho_2 F_{2,1} & 1-\rho_2 F_{2,2} & .. & -\rho_2 F_{2,N} \\ .. & .. & .. & .. \\ -\rho_N F_{N,1} & -\rho_N F_{N,2} & .. & 1-\rho_N F_{N,N} \end{bmatrix} \underline{\begin{bmatrix} B_1 \\ B_2 \\ .. \\ B_N \end{bmatrix}} = \begin{bmatrix} E_1 \\ E_2 \\ .. \\ E_N \end{bmatrix}$$

vector of unknows **[B_i]**

Quantities

- B_i.. unknown **radiosity** values of individual faces
  - when calculating color, we need to calculate radiosity for all required wavelengths (color components - e.g. **R**,**G**,**B**)
- E_i .. **own** (emitted) **radiosity** (**R**,**G**,**B**)
- ρ_i .. **reflection coefficients** of an object (**R**,**G**,**B**)
- F_{ij} .. **form-factors**
  - they depends only on scene geometry

Matrix properties of the system

- For complex scenes is matrix **M** almost **sparse**
- **M** is **diagonally dominant** and well-conditioned
  - can be solved with iteration method (Jacobi, Gauss-Seidel)

$$\sum_{\substack{j=1 \\ j \neq i}}^{N} \rho_i F_{ij} \leq 1 - \rho_i F_{ii}$$

Gauss-Seidel method

Matrix form of the system:

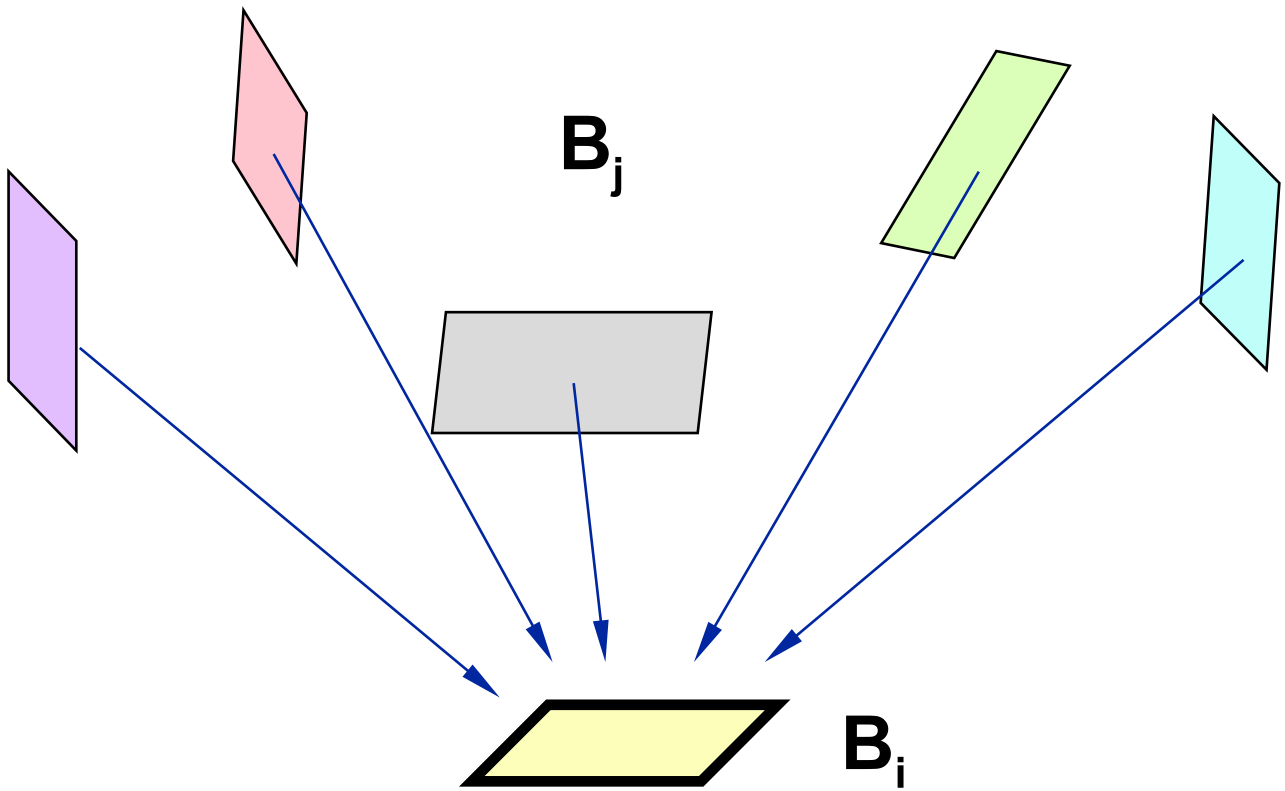
$$\mathbf{M} \cdot \mathbf{B} = \mathbf{E} \qquad \mathbf{M} = \left[\mathbf{M}_{ij} \right]_{i,j=1}^{N}$$

First estimation: $\mathbf{B}_i^{(0)} = \mathbf{E}_i$

Step:

$$\boxed{\mathbf{B}_i^{(k+1)} = \frac{\mathbf{E}_i}{\mathbf{M}_{ii}} - \sum_{j=1}^{i-1} \frac{\mathbf{M}_{ij}}{\mathbf{M}_{ii}} \mathbf{B}_j^{(k+1)} - \sum_{j=i+1}^{N} \frac{\mathbf{M}_{ij}}{\mathbf{M}_{ii}} \mathbf{B}_j^{(k)}}$$

Physical interpretation (gathering)



$$B_i = E_i + \underline{\rho_i} \cdot \sum_{j \neq i} \underline{B_j\, F_{ij}}$$

Southwell iteration method

- Jacobi and Gauss-Seidel's method updates the radiosity of single patch with the radiosities of other patches (at the expense of others!)
  - items are updated in order **1**, **2**, ... **N**
- Southwell's method always chooses the item with the **largest value of the unshooted radiosity**
- Items with a bigger radiosity are corrected earlier
  - faster convergence of the solution

Southwell's algorithm

```
double B[N], E[N], ΔB[N], M[N][N];

for ( int i=0; i<N; i++ ) {     // initialization B,ΔB[i]
  B[i] := 0.0;
  ΔB[i] := E[i];
  }

while ("does not converged") { // one calculation step
  "choose i so that ΔB[i]*A[i] == max(ΔB[i]*A[i])"
  B[i] += ΔB[i];
  for ( int j=0; j<N; j++ )
    ΔB[j] += M[j][i]*ΔB[i];
  }
  ΔB[i] = 0;
```

Southwell iteration method

❶ selection of the item with the maximum residue:

$| \mathbf{r}_i | = \max_j \{ | \mathbf{r}_j | \}$

❷ updating of the i-th solution item $\mathbf{B}_i$

❸ updating the residual vector $\mathbf{r}$

❹ steps ❶ to ❸ are repeated until the system meets the convergence criterion

Incremental calculation of residue

Updating the solution vector in a single calculation step:

$$\mathbf{B}^{(p+1)} = \mathbf{B}^{(p)} + \Delta\mathbf{B}^{(p)}$$

Residue correction:

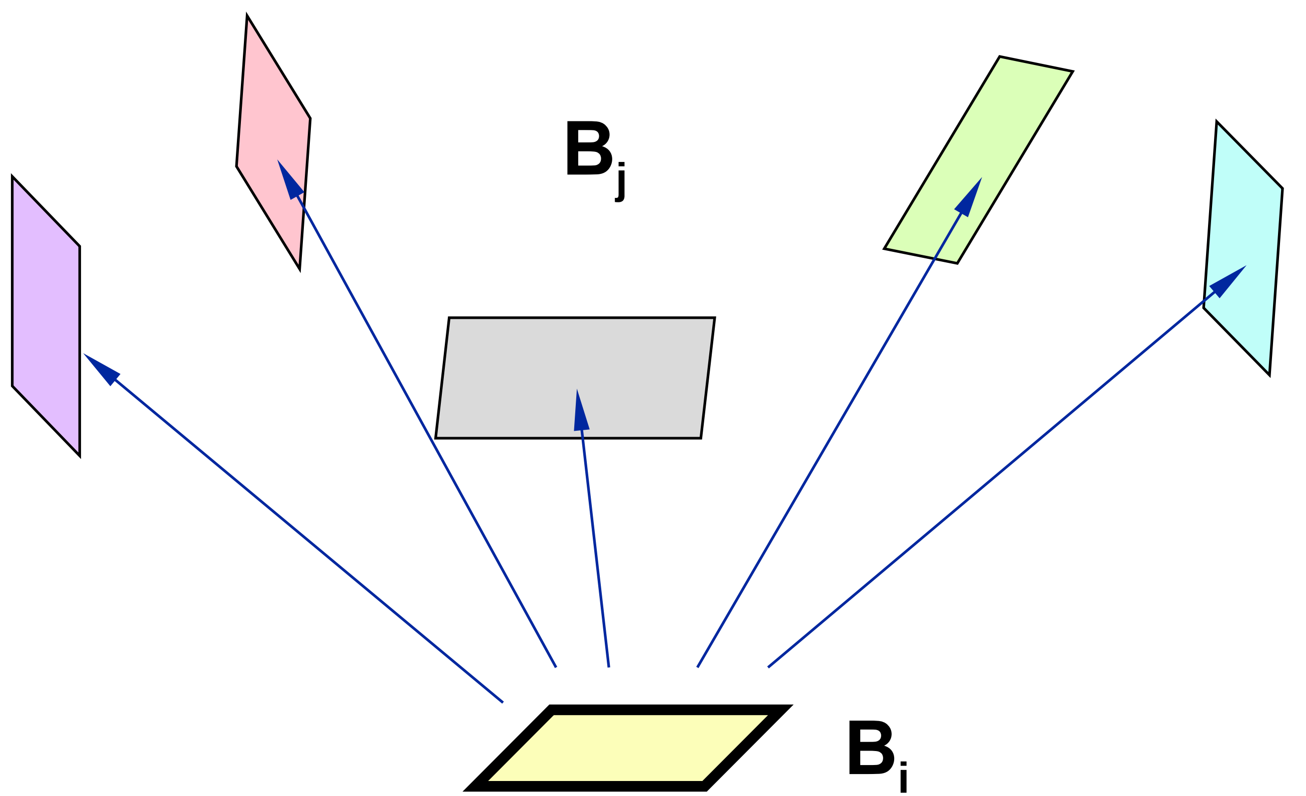
$$\underline{\mathbf{r}^{(p+1)}} = \mathbf{E} - \mathbf{M}\cdot\left(\mathbf{B}^{(p)} + \Delta\mathbf{B}^{(p)}\right) = \underline{\mathbf{r}^{(p)} - \mathbf{M}\cdot\Delta\mathbf{B}^{(p)}}$$

Physical interpretation (shooting)

- **B_i** .. radiosity of the i-th face (direct and indirect)
- **on step of the calculation** = radiosity (shoot) distribution of the i-th face to the neighborhood faces
- **ΔB_i** .. yet **unshooted radiosity** of the i-th face
- **method convergence** = total unshooted radiosity in the scene is decreasing

Physical interpretation (shooting)

$$B_j^{(p+1)} = B_j^{(p)} + \underline{\Delta B_i^{(p)} \cdot \rho_j \cdot F_{ji}}$$



Progressive radiosity

- M. Cohen at al., SIGGRAPH ‘88
- **interactive illumination calculation**
  - a progressive result is displayed after each step
  - an effort to estimate the solution in the first few steps

- Southwell's method alteration
  - selecting the face with the most **unshooted energy**
  - using of the ambient term of the illumination

Progressive radiosity

```
double B[N], E[N], ΔB[N], F[N][N], A[N], ro[N];

for ( int i=0; i<N; i++ ) {      // initialization B,ΔB[i]
  B[i] := E[i];
  ΔB[i] := E[i];
  }
while ("does not converged") {  // one calculation step
  "choose i so that ΔB[i]*A[i] == max(ΔB[i]*A[i])"
  for ( int j=0; j<N; j++ ) {
    double ΔRad = ΔB[i]*ro[j]*F[j][i];
    B[j] += ΔRad;
    ΔB[j] += ΔRad;
    }
  ΔB[i] = 0.0;
  "displaying halftime result using radiosity B[i]"
  }
```

Ambient term

- Improving the look of continuously displayed halftime results
- Approximation of the non-computed light reflections

Total already unshooted radiosity:

$$\overline{\Delta B} = \frac{\sum \Delta B_i \cdot A_i}{\sum A_i}$$

Ambient term

Average coefficient of reflection: $\bar{\rho} = \dfrac{\sum \rho_i \cdot A_i}{\sum A_i}$

Estimation of residual (ambient) radiosity:

$$\underline{B_{amb}} = \overline{\Delta B} \cdot \left(1 + \bar{\rho} + \bar{\rho}^2 + \ldots\right) = \underline{\frac{\overline{\Delta B}}{1 - \bar{\rho}}}$$

When the scene is displayed, the radiosity of each face is recalculted:

$$\boxed{B_i^{disp} = B_i + \rho_i \cdot B_{amb}}$$

References

- **C. M. Goral, K. E. Torrance, D. P. Greenberg, B. Battaile**: *Modeling the Interaction of Light Between Diffuse Surfaces*, CG vol 18(3), SIGGRAPH 1984
- **A. Glassner**: *Principles of Digital Image Synthesis*, Morgan Kaufmann, 1995, 871-937
- **M. Cohen, J. Wallace**: *Radiosity and Realistic Image Synthesis*, Academic Press, 1993, 13-64
- **J. Foley, A. van Dam, S. Feiner, J. Hughes**: *Computer Graphics, Principles and Practice*, 793-804
- **M. Feda, W. Purgathofer**: *Accelerating radiosity by overshooting*, The Third EG Workshop on Rendering, Bristol, 1992, 21-32