Numerical solutions of the radiosity system of equations

© 1996-2016 Josef Pelikán
CGG MFF UK Praha

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

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:

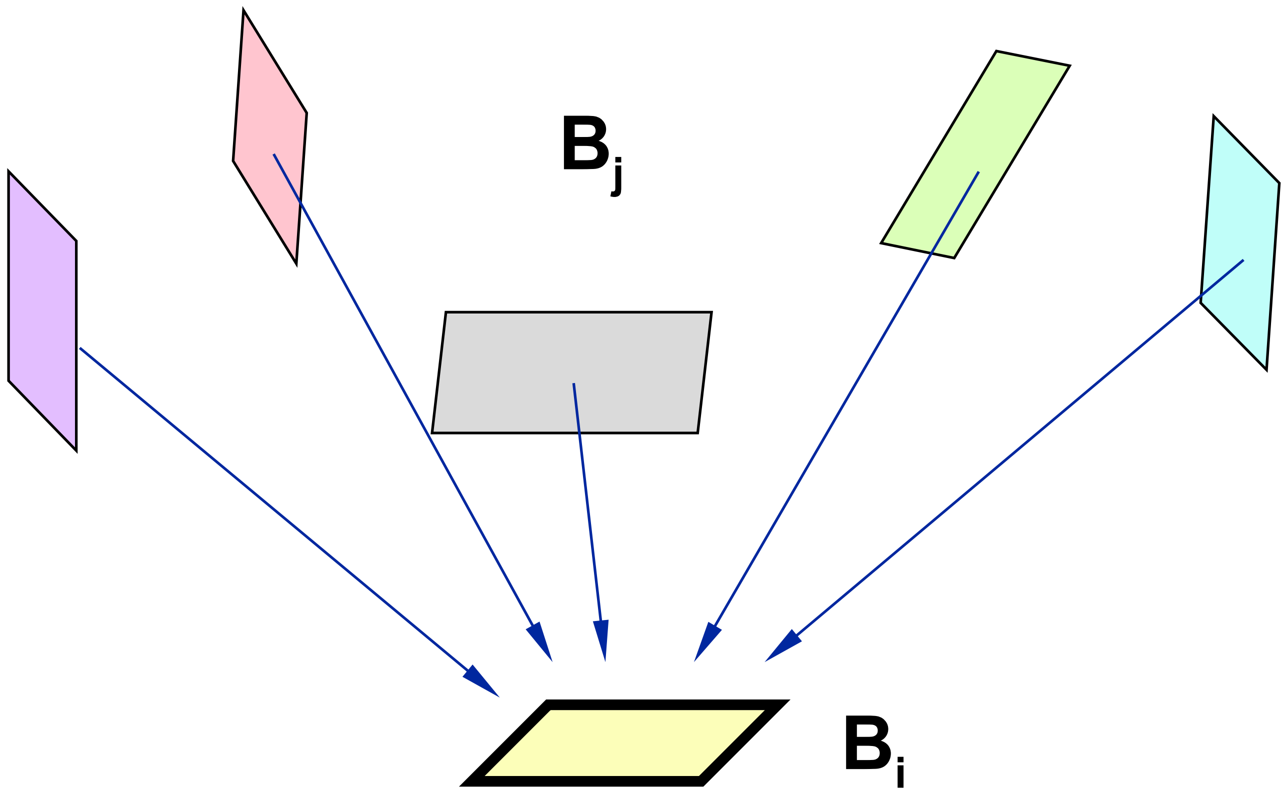
$$\mathbf{M} \cdot \mathbf{B} = \mathbf{E} \qquad \mathbf{M} = \left[\mathbf{M}_{ij}\right]_{i,j=1}^{N}$$

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

Step:

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

Physical interpretation (gathering)



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

Residue

Residue (error estimation) of the k-th iteration:

$$\mathbf{r}^{(k)} = \mathbf{E} - \mathbf{M} \cdot \mathbf{B}^{(k)}$$

In one calculation step, one item of the vector of the solution $\mathbf{B}_i$ is updated:

$$\mathbf{B}_i^{(k+1)} = \mathbf{B}_i^{(k)} + \frac{\mathbf{r}_i^{(k)}}{\mathbf{M}_{ii}}$$

(Jacobi method .. residues are corrected after iteration
Gauss-Seidel .. correction is calculated after each step)

Southwell iteration method

- Jacobi and Gauss-Seidel's method zeroes one component of the residue at each calculation step (at the expense of others!)
  - items are updated in order **1**, **2**, … **N**

- Southwell's method always chooses the item with the **largest absolute value of the residue**

- Items with a bigger errors are corrected more often
  - faster convergence of the solution

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:

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

Residue correction:

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

Since only the i-th component of the solution vector has changed:

$$r_j^{(p+1)} = r_j^{(p)} - M_{ji} \cdot \frac{r_i^{(p)}}{M_{ii}} \qquad j = 1..N$$

Southwell's algorithm

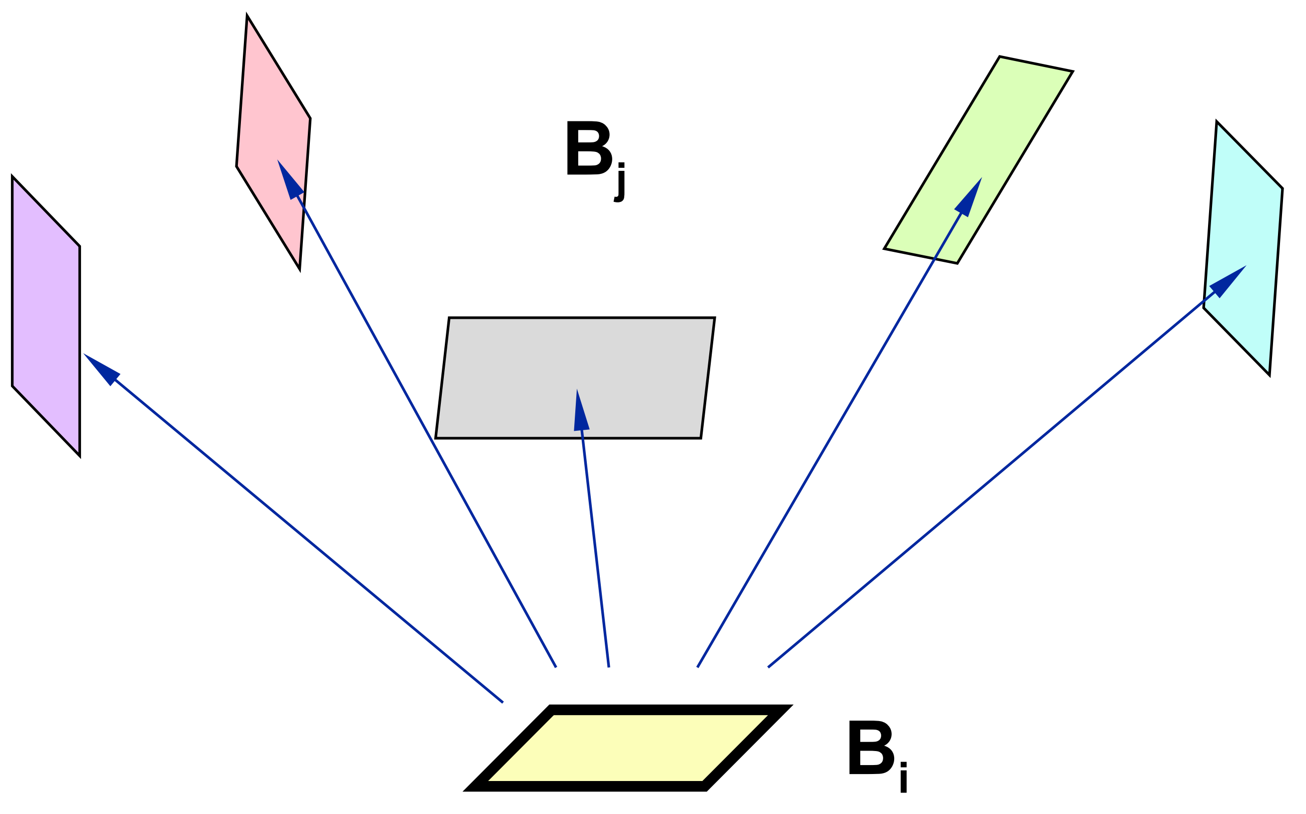
```
double B[N], E[N], r[N], M[N][N];

  //solutions and residues initialization
for ( int i=0; i<N; i++ ) {
  B[i] := 0.0;
  r[i] := E[i];
  }
while ("does not converged") {
    // one step of calculation
  "choose i so that fabs(r[i])== max(fabs(r[i]))"
  double delta = r[i]/M[i][i];
  B[i] += delta;
  for ( int j=0; j<N; j++ )
    r[j] -= M[j][i]*delta;
  }
```

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
- r_i .. yet **unshooted radiosity** of the i-th face
- **method convergence** = total unshooted energy in the scene is decreasing

Physical interpretation (shooting)

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



Total unshooted energy

According to the reciprocal rule for form-factors:

$$\underline{r_j^{(p+1)}} = r_j^{(p)} + \rho_j \cdot F_{ji} \cdot r_i^{(p)} = \underline{r_j^{(p)} + \rho_j \cdot F_{ij} \frac{A_i}{A_j} \cdot r_i^{(p)}}$$

Energy distribution in one calculation step:

$$r_j^{(p+1)} \cdot A_j = r_j^{(p)} \cdot A_j + \underbrace{\rho_j \cdot F_{ij}}_{<1} \cdot \underline{r_i^{(p)} \cdot A_i} \qquad j = 1..N$$

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], dB[N], F[N][N], A[N], ro[N];

for ( int i=0; i<N; i++ ) {      // initialization B, dB
  B[i] := E[i];
  dB[i] := E[i];
  }
while ("does not converged") {   // one calculation step
  "choose i so that dB[i]*A[i]== max(dB[i]*A[i])"
  for ( int j=0; j<N; j++ ) {
    double dRad = dB[i]*ro[j]*F[j][i];
    B[j] += dRad;
    dB[j] += dRad;
    }
  dB[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:

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

Ambient term

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

Estimation of residual (ambient) radiosity:

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

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

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

Literature

- **M. Cohen, J. Wallace**: ***Radiosity and Realistic Image Synthesis***, Academic Press, 1993, 109-130 (chyby!)
- **M. Cohen, S. E. Chen, J. R. Wallace, D. P. Greenberg**: ***A progressive refinement approach to fast radiosity image generation***, SIGGRAPH '88, 75-84

Literature

- **A. Glassner**: ***Principles of Digital Image Synthesis***, Morgan Kaufmann, 1995, 900-916
- **J. Foley, A. van Dam, S. Feiner, J. Hughes**: ***Computer Graphics, Principles and Practice***, 800-803
- **M. Feda, W. Purgathofer**: *Accelerating radiosity by overshooting*, The Third EG Workshop on Rendering, Bristol, 1992, 21-32