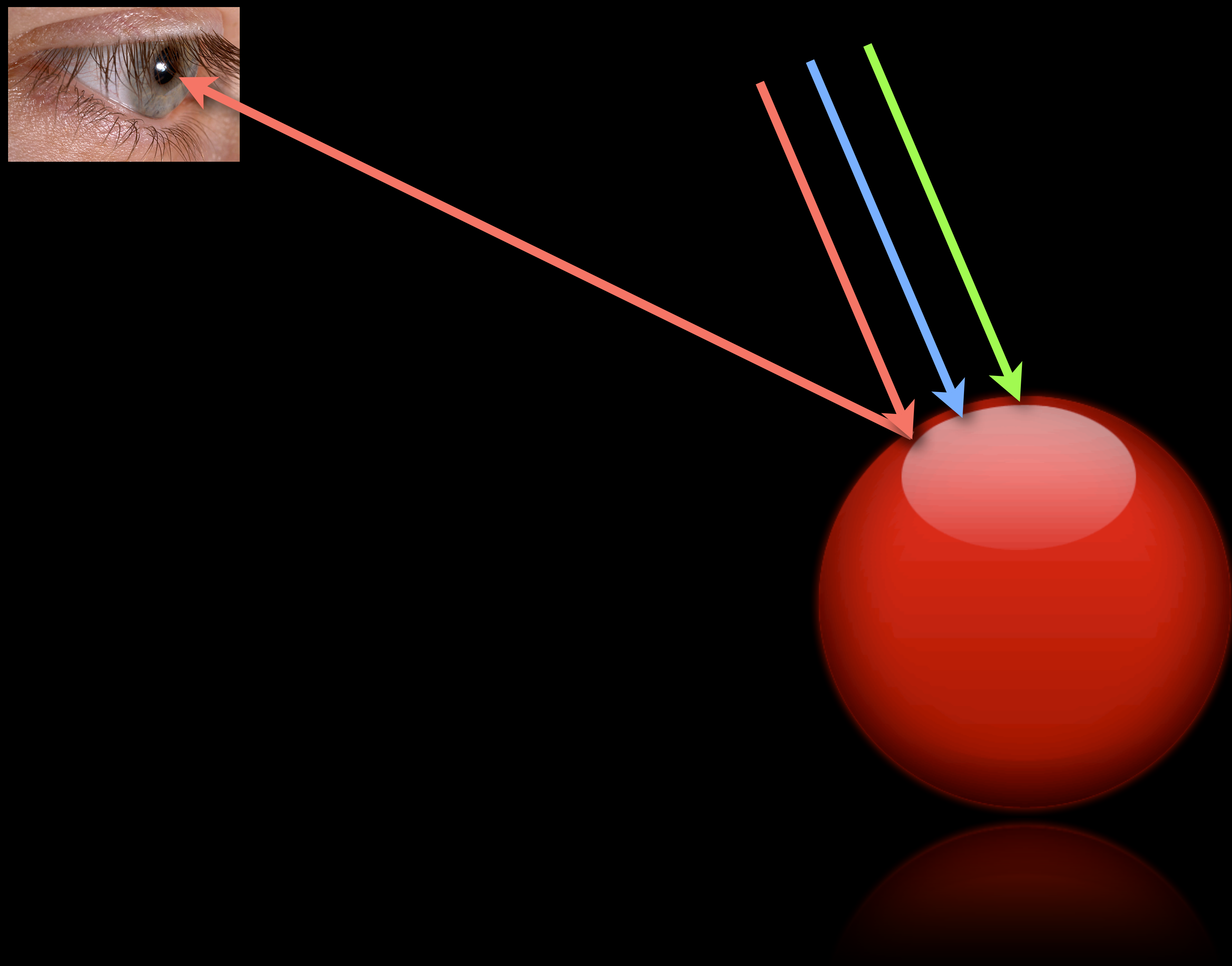
The Computational Complexity of
Hierarchical Radiosity

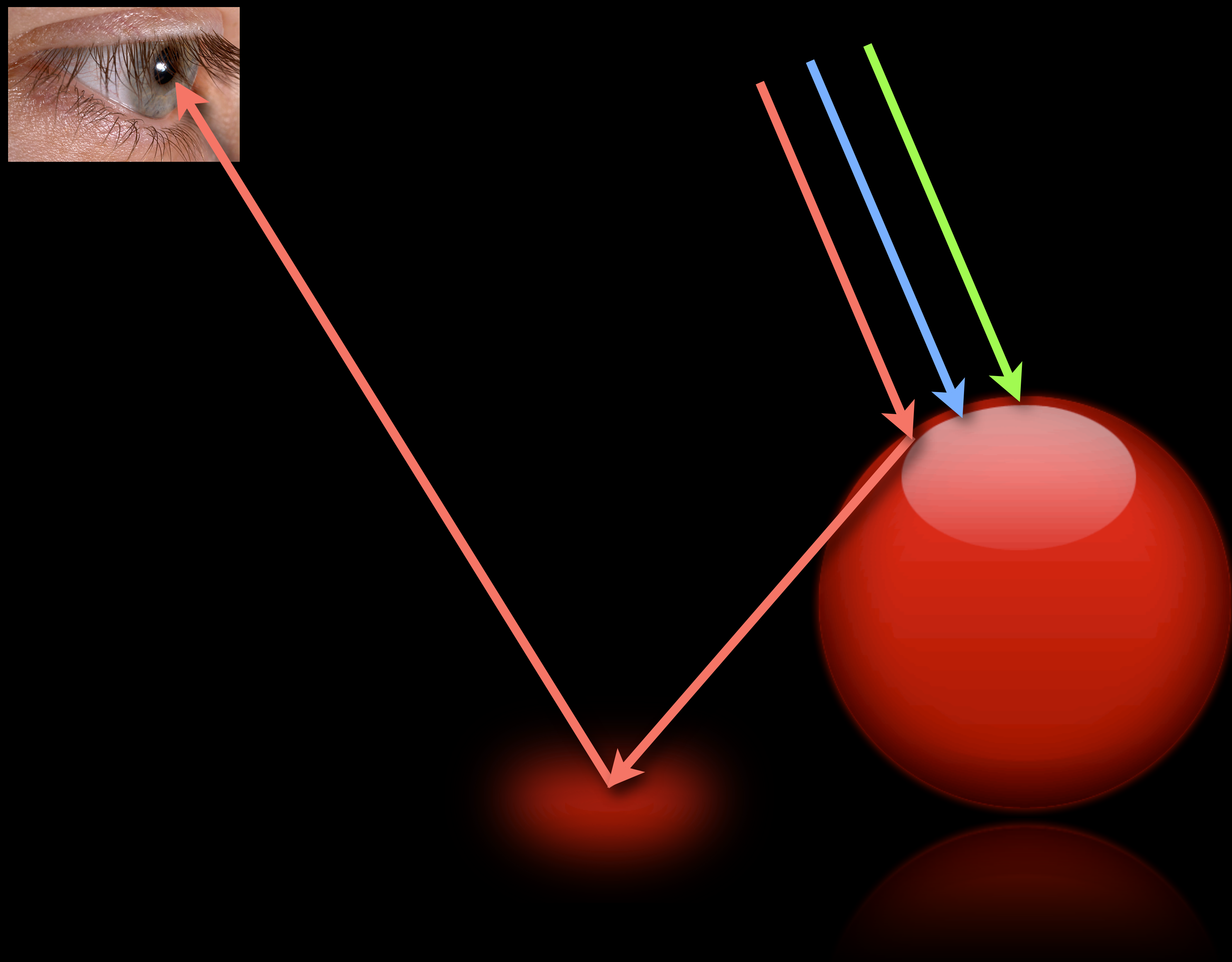
What is radiosity?

- Method for simulating light in an 3D environment.
- Realism as goal.

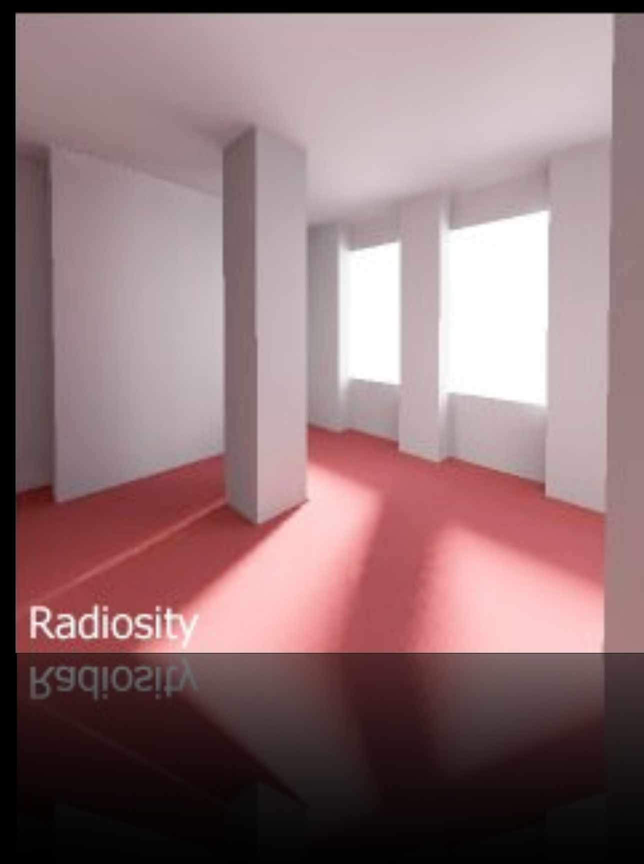


Direct illumination

Direct illumination



Radiosity



Radiosity

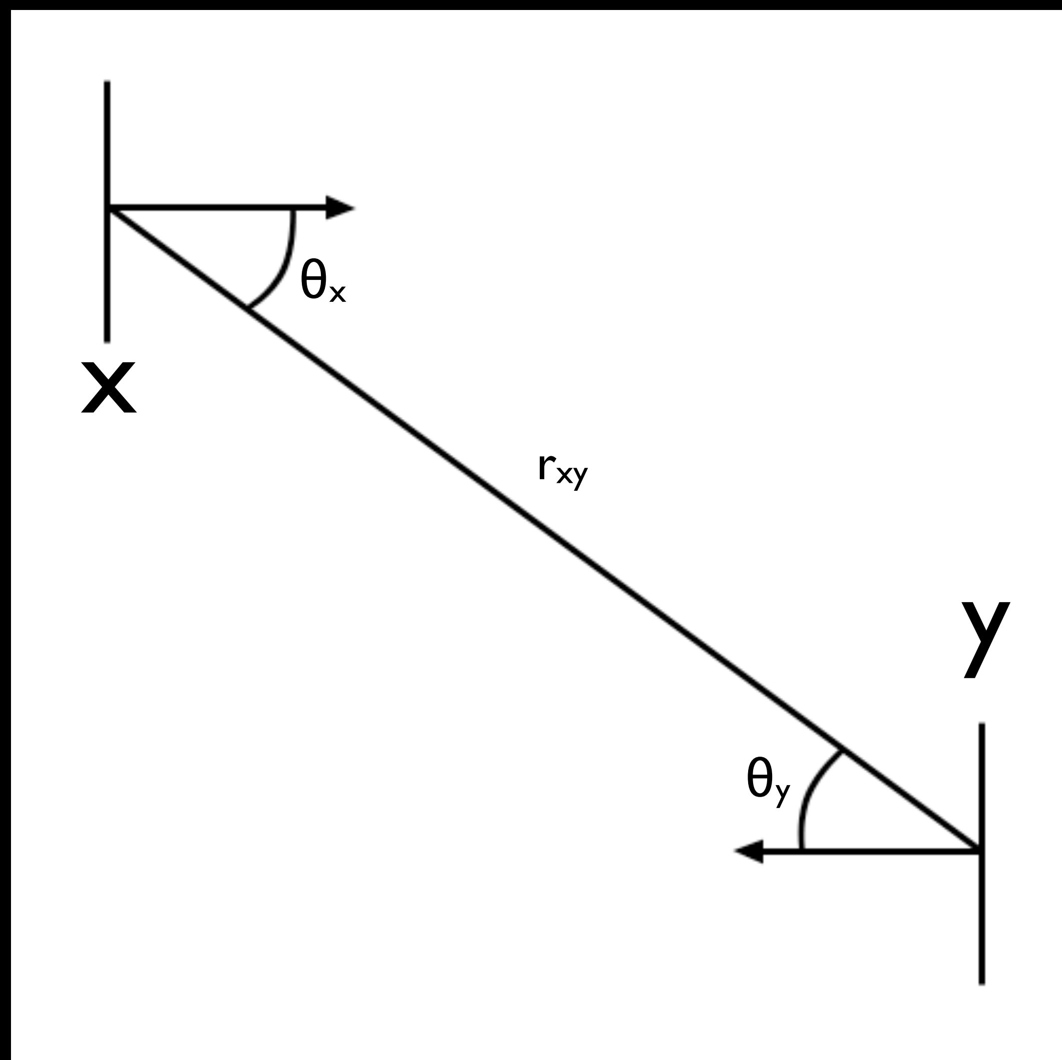
Radiosity Equation

$$B(y) = B_e(y) + \rho(y) \int_S G(x,y)V(x,y)B(x)dx$$

- B(y),B(x): Radiosity in point y and point x
- $B_e(y)$: Radiosity emitted by y if y is a lightsource
- ρ(y): Reflection coefficient
- G(x,y): Geometric relationship
- V(x,y): Visibility function

B(y): Ljusstyrkan i punkt y
$B_e(y)$: Ljusintensitet om ljuskälla
ρ(y): Reflaktiongrad, Rå
G(x,y): Geometriskt förhållande För viktning av överförd energi. Nästa slide.
V(x,y): Ifall punkterna ser varandra
B(x): Ljusstyrkan i punkt x

Geometric relationship



$$G(x,y) = \frac{\cos\theta_x \cos\theta_y}{2r_{xy}}$$

Detta förhållande används för att vikta överföringen av energi.

The Kernel

$$B(y) = B_e(y) + \rho(y) \underbrace{\int_S G(x,y)V(x,y)B(x)dx}$$

$$k(x,y) = \rho(y)G(x,y)V(x,y)$$

Funktionerna ρ, G och V kombineras till kärnan.

Calculating radiosity

The radiosity equation cannot be solved analytically in general.

- Approximation
- Hanrahan et al.
- One of many
- Effective?

Hanrahans rapport argumenterar effektiviteten.
Används iom beräkningseffektiviteten.

Hierarchical radiosity

Transform the radiosity equation to
a linear system of equations

$$B(y) = B_e(y) + \rho(y) \int_S G(x,y)V(x,y)B(x)dx$$

$$b_i = e_i + \sum_{j=1}^{n} k_{ij} b_j, \qquad i = 1, \dots, n$$

Diskreta approximationer av original ekvationen

This article

- Refines and generalizes the complexity analysis
- Shows a surprising worst-case behavior

Effektivitetsmässigt värre i praktiken

Hanrahan et al. procedure

```
ProjectKernel(Element i, Element j)
  error= Oracle(i,j);
  if (Acceptable(error) || RecursionLimit(i,j))
    link (i,j);
  else
    if (PreferredSubdivision(i,j) == i)
      ProjectKernel(LeftChild(i), j);
      ProjectKernel(RightChild(i), j);
    else
      ProjectKernel(i, LeftChild(j));
      ProjectKernel(i, RightChild(j)).
```

Oracle: felet på interaktionen (rimligheten i påverkan)
Acceptable(error): accepterar felet eller ej => länk
RecursionLimit(i,j): resursbegränsning => länk
PreferredSubdivision: avgör element att dela.

Element: Yta
Link: Interaktion mellan två ytor(element)

Modified algorithm

- Modified subdivision
- RecursionLimit = false for arbitrary pairs of elements
- Oracle function:

$$\frac{\max\left(A_x, A_y\right)}{r_{xy}}$$

Ändrad delning => <dubbelt med länkar
Ax/Ay: Storleken på elementen
rxy: avstånd mellan mittpunkterna

Complexity analysis

Hanrahan et al. showed that

$$N = \theta(g\varepsilon^{-1})$$

$$L = \theta(N\varepsilon^{-1})$$

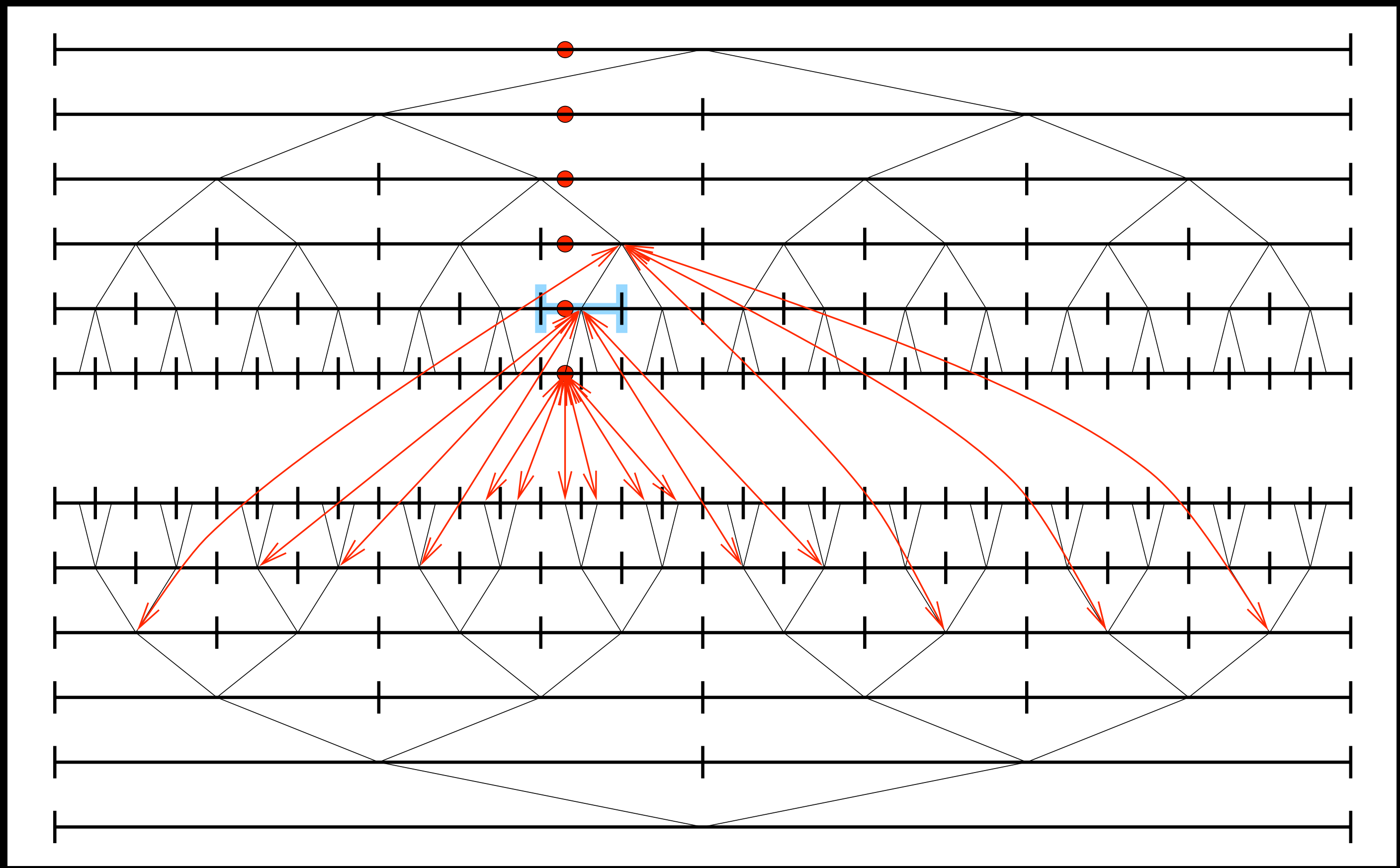
N: Antalet element
L: Antalet länkar
g: Beror på initialgeometrin av scenen

$$N = \theta(g\varepsilon^{-1})$$

Number of leaves: $N_{leaf} = \frac{l}{A_{leaf}} = \theta\left(\frac{l}{d}\varepsilon^{-1}\right)$

Number of nodes: $N = 2(2N_{leaf} - 1) = \theta\left(\frac{l}{d}\varepsilon^{-1}\right)$

$$g := \frac{l}{d} \Rightarrow N = \theta(g\varepsilon^{-1})$$



d: avståndet mellan ytorna.
l: längden på ursprungselementet
A_{leaf}: storleken på löven

$$L = \theta(N\varepsilon^{-1})$$

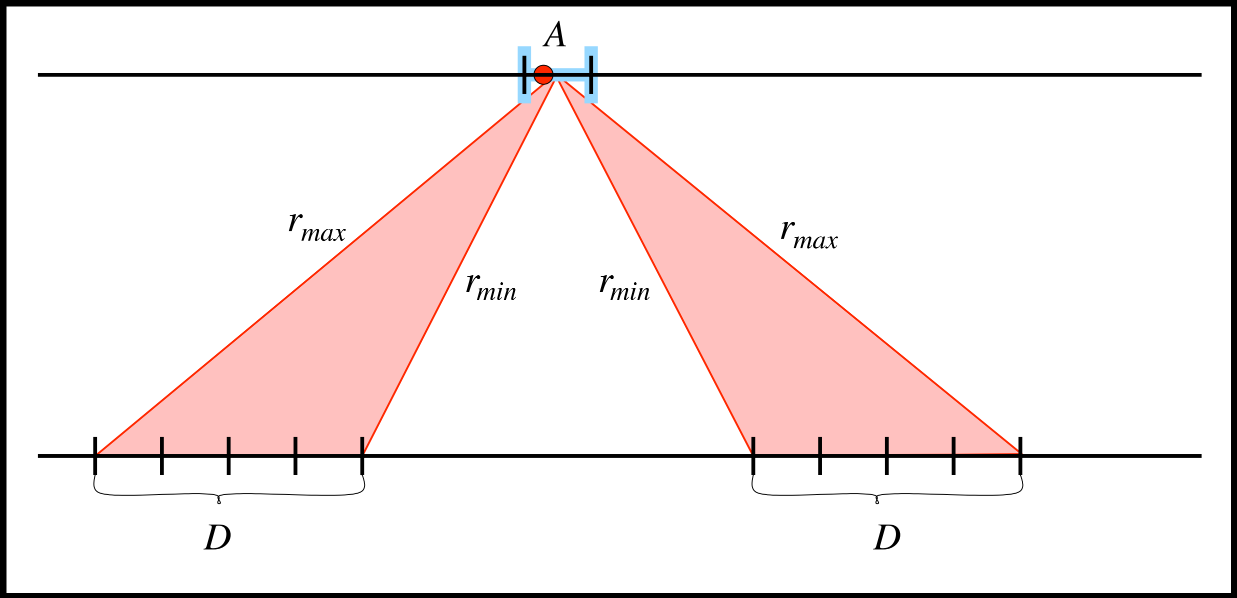
$$r_{max} - r_{min} \leq D \leq r_{max} + r_{min}$$

$$r_{max} - r_{min} = A(\varepsilon^{-1} + \delta)$$

$$r_{max} + r_{min} = A(3\varepsilon^{-1} + \delta)$$

$$A\theta(\varepsilon^{-1}) \geq D \geq A\theta(\varepsilon^{-1})$$

$$L = \theta\left(N\frac{D}{A}\right) = \theta(N\varepsilon^{-1})$$



rmax-rmin har samma O som rmax+rmin => D också får det O
Antalet länkar L är (D/A)*N alltså antalet grannar gånger antalet noder/element

Hanrahan Vs. Garmann

$$L = \theta(N\varepsilon^{-1})$$

Hanrahan et al. assumed a constant error threshold

$$L = \theta(N)$$

Garmann et al. assumed a variable error threshold

$$\varepsilon^{-1} = \theta(N) \Rightarrow$$
$$L = \theta(N^2)$$

Garmanns metod är mer realistisk