

1400008549
còpia 1

**Diffuse interreflections.
Techniques for Form-Factor Computation.**

Xavier Pueyo

Report LSI-90-16



Diffuse Interreflections. Techniques for Form-Factor Computation

Xavier Pueyo

INRIA-IRISA *

IG-LiSI †

*Institut National de Recherche en Informatique et Automatique. Institut de Recherche en Informatique et Systèmes Aléatoires. Projet SIAMES. Campus de beaulieu. F-35042 RENNES.

†Sec. Informàtica Gràfica. Dept. Llenguatges i Sistemes Informàtics. Univ.Politécnica de Catalunya. Av.Diagonal,647 pl8. E-08029-BARCELONA (on leave).

Abstract

The purpose of this report is to present a characterization of the techniques proposed until now to model diffuse interreflections. Basic approaches are presented as well as the evolution of implementation strategies. In the conclusions, open problems are enumerated.

1 Introduction

As well known [9, 7], **radiosity** was introduced in 1984 [6] as a new powerful method to model the behavior of light interacting between diffuse reflecting surfaces. This new technique dealt with a problem of illumination models not studied in depth up until this moment.

Radiosity was borrowed from thermal engineering [13] and its application to image synthesis has evolved very quickly. The bursting of the new technique into computer graphics has also activated the development of **direct and invers ray tracing** techniques to compute diffuse reflections and to obtain global illumination models with suitable implementation techniques [10, 1]. On the other hand, instead of rivaling, radiosity has collaborated with both types of ray tracing in search of a good implementation technique for **global illumination models** [14].

2 Radiosity

This section describes the basic notions of radiosity, the different methods proposed for its implementation and their evolution. Good survey-like papers in this field may be found in current literature [8, 7] as well as interesting discussions about drawbacks of proposed radiosity implementation techniques [2, 16].

2.1 First notions

As stated above, radiosity was introduced in computer graphics to model the interreflection of light between ideal diffuse (Lambertian) reflectors. The method is useful for an arbitrary environment and it can account for illumination from a variety of light sources. Derived from thermal engineering, its application to image synthesis was proposed by Goral et al. [6] with the following conclusions:

1.
$$B_i = E_i + \rho_i \sum_{j=1}^N B_j F_{ij}, \quad i = 1, \dots, N$$
 (1)

where (see figure 1)



- B_i is the radiosity of surface i
- E_i is the energy directly emitted from surface i
- ρ_i is the reflectivity of surface i
- F_{ij} is the form-factor and represents the fraction of radiant energy leaving surface j and landing on surface i
- N is the number of patches the whole environment is divided into (figure 2)

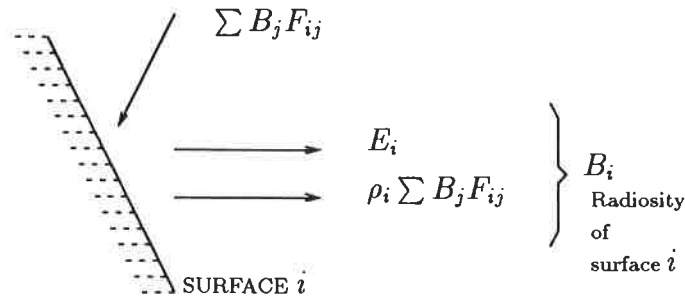


Figure 1: Energy exchange on a surface

Equation 1 yields a set of N linear equations with N B_i unknowns where E_i and ρ_i are application data, and F_{ij} must be derived from geometry of the environment,

$$\begin{bmatrix} 1 - \rho_1 F_{1,1} & -\rho_1 F_{1,2} & \dots & -\rho_1 F_{1,N} \\ -\rho_2 F_{2,1} & 1 - \rho_2 F_{2,2} & \dots & -\rho_2 F_{2,N} \\ \vdots & & & \vdots \\ -\rho_N F_{N,1} & -\rho_N F_{N,2} & \dots & 1 - \rho_N F_{N,N} \end{bmatrix} \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_N \end{bmatrix} = \begin{bmatrix} E_1 \\ E_2 \\ \vdots \\ E_N \end{bmatrix} \quad (2)$$

2. Figure 3 shows the parameters participating to the computation of form-factor which is obtained by,

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos \phi_i \cos \phi_j}{\pi r^2} dA_j dA_i \quad (3)$$

Instead of solving this double area integral, Goral et al. used Stoke's theorem and solved a contour integral. In fact, the resolution of the integral in equation 3 is the key to the different implementation methods as we will see in the following sections.

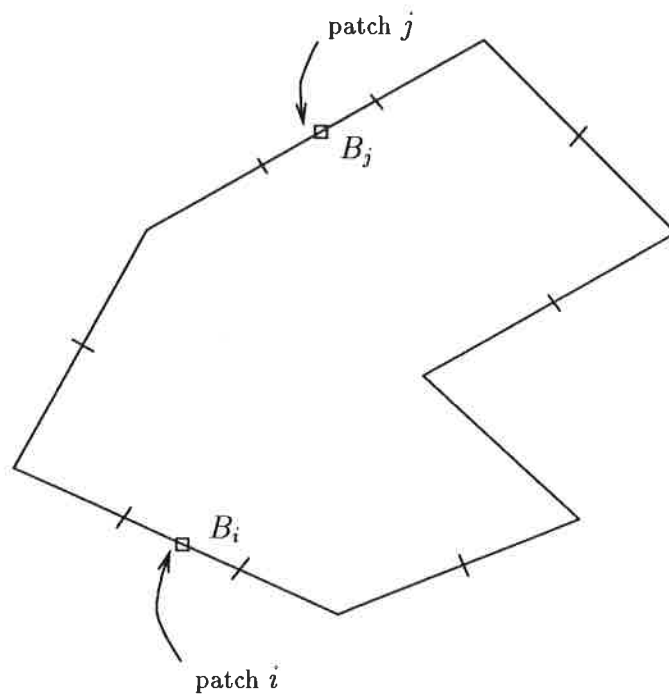


Figure 2: Environment subdivision

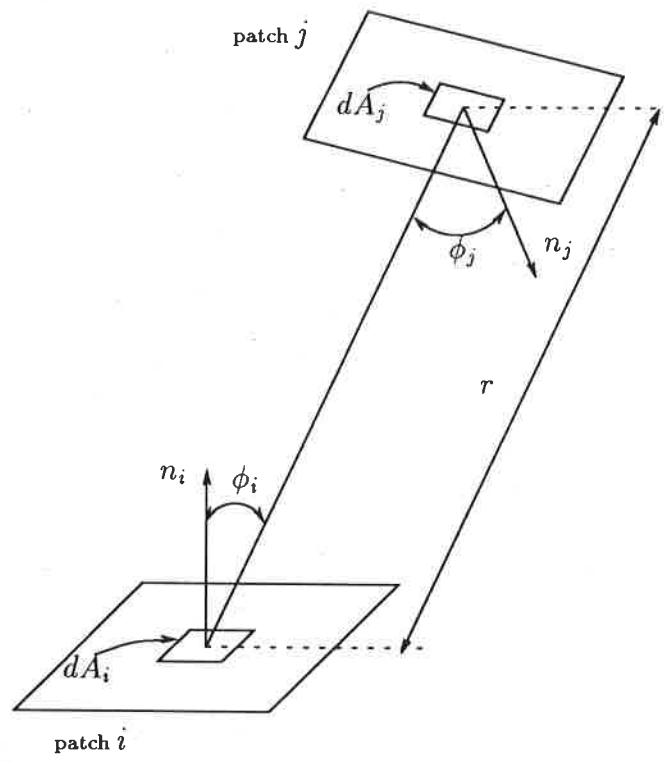


Figure 3: Geometry for form-factor derivation

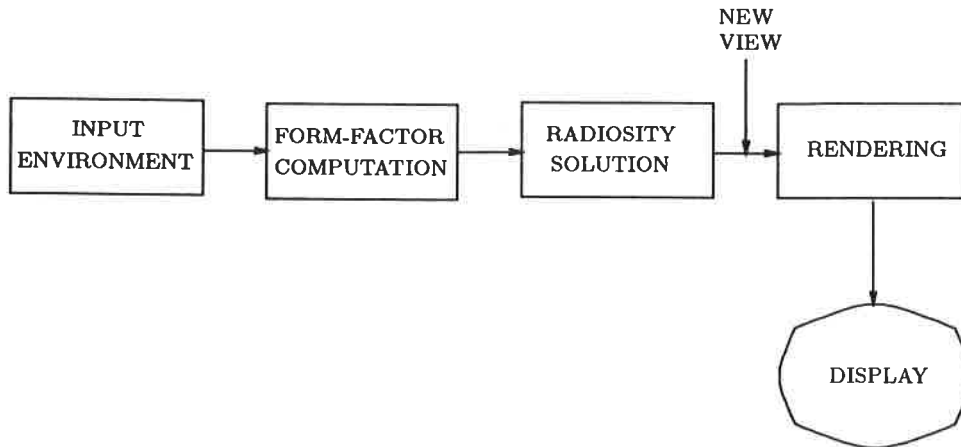


Figure 4: Program scheme

Some simple identities are also employed to simplify the computation of form-factors:

- $A_i F_{ij} = A_j F_{ji}$ (4)
- $\sum_{j=1}^N F_{ij} = 1, \quad i = 1, \dots, N$ (5)
- $F_{ii} = 0$ (6)

The computation of form factors and the resolution of equation 2 for a discretization N of the environment, gives the radiosity (B_i) of the patches centre points which are view independent. Finally, the rendering procedure computes the scene's visible points by z-buffering which intensities are calculated by linear interpolation of B_i 's in object space.

Figure 4 shows a scheme of the complete procedure, underlying the usefulness of the method to render several views of a scene with low computational cost.

2.2 Form-Factor computation

We have seen in the previous section that form-factor is given by the geometric relationship of two surface entities (patch to patch) and represents the ratio of energy leaving one of these entities that lands on the other. In real environment, light leaving a surface and arriving on another depends not only on this relationship but also on being elements of the scene that intercept light producing shadows (figure 5). Then form-factors need only to be computed between surfaces or pieces of surfaces which are visible one to another. So a hidden surface

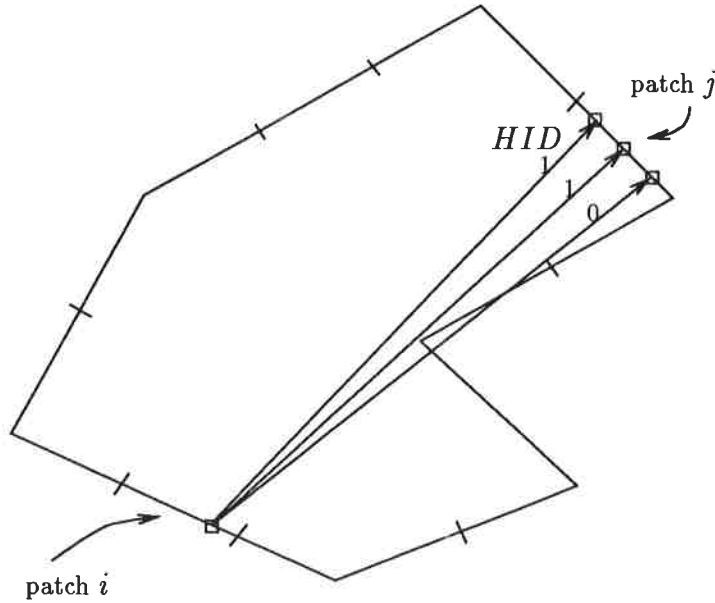


Figure 5: Considering the occlusions

removal process takes place for the computation of each form-factor. Thus, equation 3 must be modified in the following way,

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} \frac{\cos \phi_i \cos \phi_j}{\pi r^2} HID dA_j dA_i \quad (7)$$

where HID takes value one or zero depending on whether dA_j is visible from dA_i .

Three basic approaches have been used for the computation of form factors:

1. **Analytical solution.** This was the first approach, employed by Goral et. al. [6]. As stated above, it is applied to environments without occlusions and it is based on the used of Stoke's theorem.
2. **Hemi-space projection.** It is based on a geometric analog for the form-factor integral developed by Nusselt [13]. This states that, for a finite area the form-factor is equivalent to the fraction of circle (which is the base of the hemisphere surrounding the normal of the patch) covered by projecting the area onto the hemisphere and then ortographically down onto the circle (figure 6).

Discretizing the surface of the hemisphere and then computing a form-factor (**delta form-factor**) for each resulting element, gives a way of computing the form-factor to

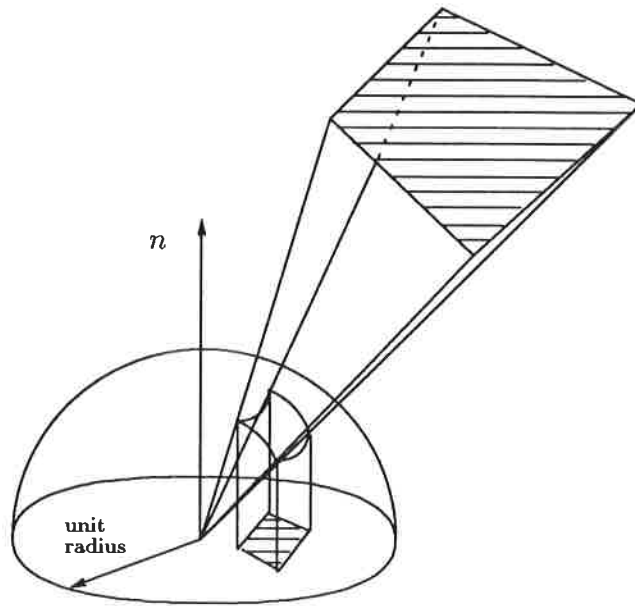


Figure 6: Nusselt's analog

the patch. This can be done projecting the patch onto the hemisphere and performing the summation of the delta form-factors of the covered elements of the hemisphere. Of course, occlusions must be taken into account.

3. **Ray Traced.** Now the environment is sampled tracing rays from a point. These rays are casted on the basis of a discretization of the hemi-sphere surrounding the point or a discretization of the environment (figure 7).

2.2.1 Hemi-space projection approach

The Hemicube

This is a technique proposed by Cohen and Greenberg [4], which is based on:

- If the distance between the two patches is large compared to their size and there is not occlusion between them, the inner integrand in equation 3 remains almost constant.

From this assertion, the patch-to-patch ($F_{A_i A_j}$) form-factor is approximated with a point-to-patch ($F_{dA_i A_j}$) form-factor,

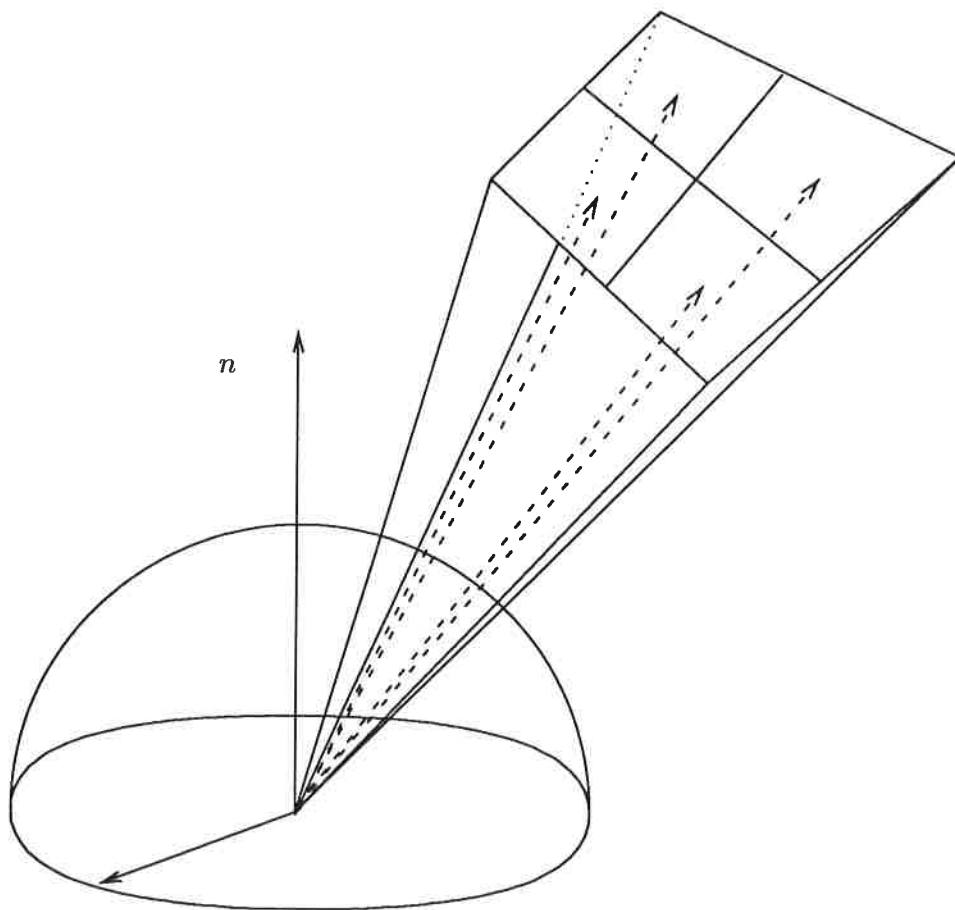


Figure 7: Ray traced approach

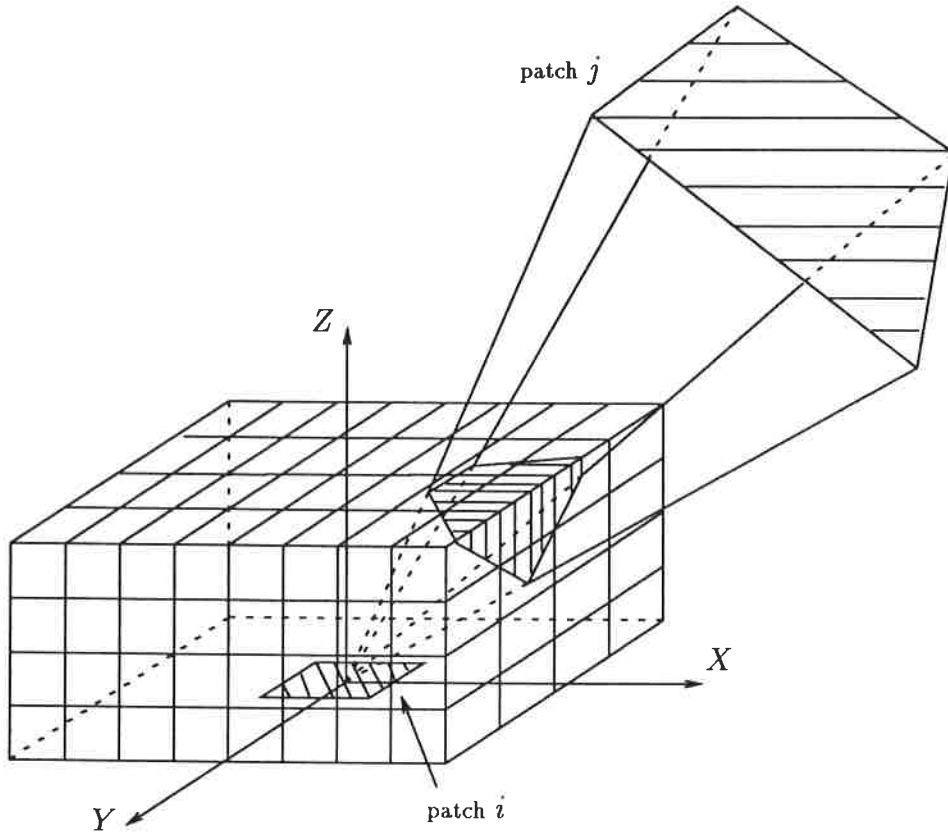


Figure 8: The hemi-cube

$$F_{ij} \approx F_{dA_i A_j} = \int_{A_j} \frac{\cos \phi_i \cos \phi_j}{\pi r^2} dA_j \quad (8)$$

which represents the ratio of light leaving patch j that lands on the centre point of patch i , and is supposed to be constant over patch i .

If the distance-to-size relationship or occlusion hypothesis fail, patches can be subdivided into smaller patches. As we will see below, other solutions, than subdividing, have been proposed to deal with this problem.

Problems created by the geometry of the hemisphere to create and to manage a convenient discretization of its surface are solved replacing the hemisphere by an **hemi-cube** (figure 8) which moreover provides a good way to treat occlusions.

Each of the five faces of the hemi-cube are subdivided into *pixels*, the whole environment is projected onto each face and a z-buffering process is applied to projected patches in order to perform visibility calculation. Finally, the form-factor to a patch j is given by,

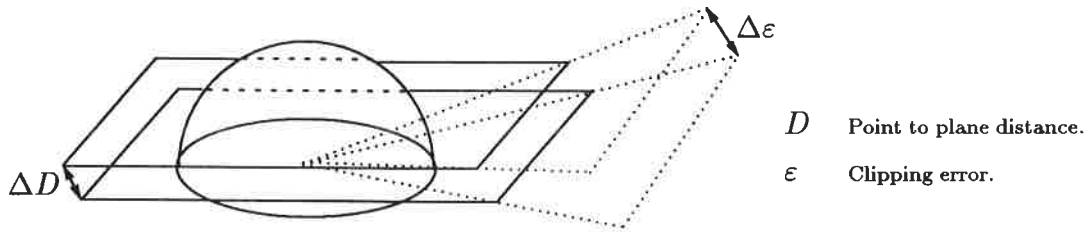


Figure 9: Approaching the hemisphere by a plane.

$$F_{ij} = \sum_{q=1}^R \Delta F_q \quad (9)$$

where,

- ΔF_q is the delta form-factor associated with pixel q on the hemi-cube
- R is the number of hemi-cube pixels covered by projection of the patch onto the hemi-cube

Hemi-space projection onto a plane

Sillion et al. [15] propose an alternative to the hemi-cube as projection surface. Now the hemisphere surrounding the point under study is replaced by a portion of a plane parallel to the tangent plane on the point (figure 9). The size of this piece of plane is defined depending on the accepted error introduced by the method. In fact, we can understand this solution as the substitution of the whole hemicube by a piece of plane which is an extension of the hemicube's top face (figure 10).

As in the hemi-cube technique, the projection plane is subdivided into *pixels*. Now these pixels (denoted as *proxels*) are of different sizes and "each one contributes for about the same amount to the form-factor" (figure 11).

2.2.2 Ray traced approach

Point to point method

Simultaneously to the hemi-cube technique, Nishita et al. [12] proposed another way to solve equation 7 for environment with convex polyhedra. Surfaces in the environment are also subdivided into rectangular patches; but now radiosities will be computed for each patch vertex instead of for centre points (figure 12).

Under the hypothesis that patches are usually very small, form-factors are approximated by,

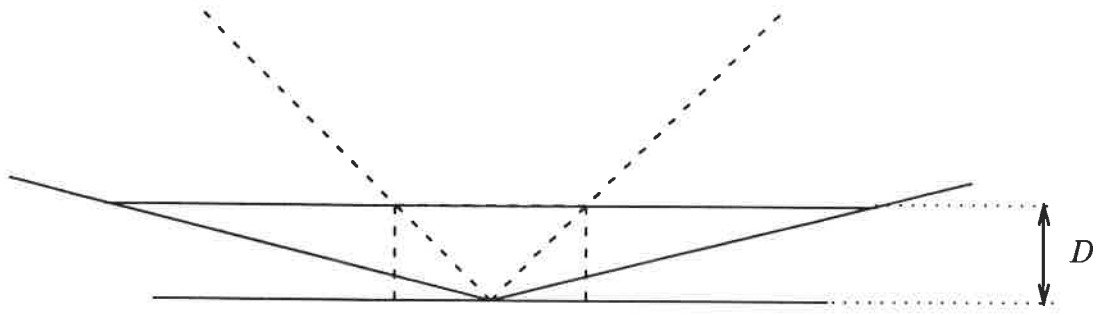


Figure 10: Hemicube - plane "relationship".

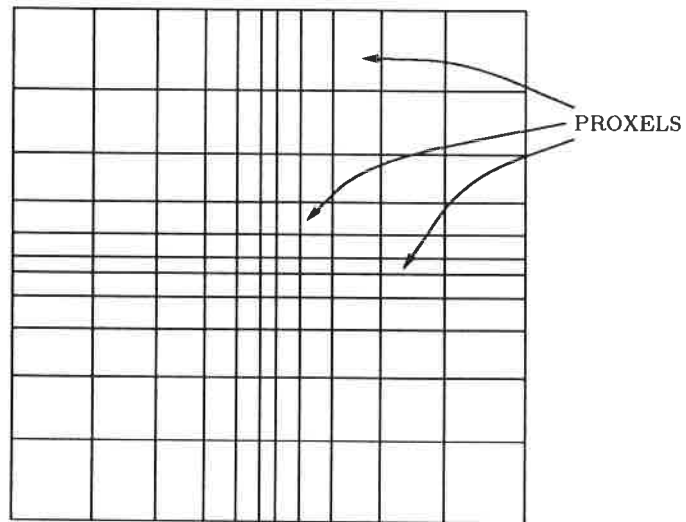


Figure 11: Subdivision of the projection plane into *proxels*

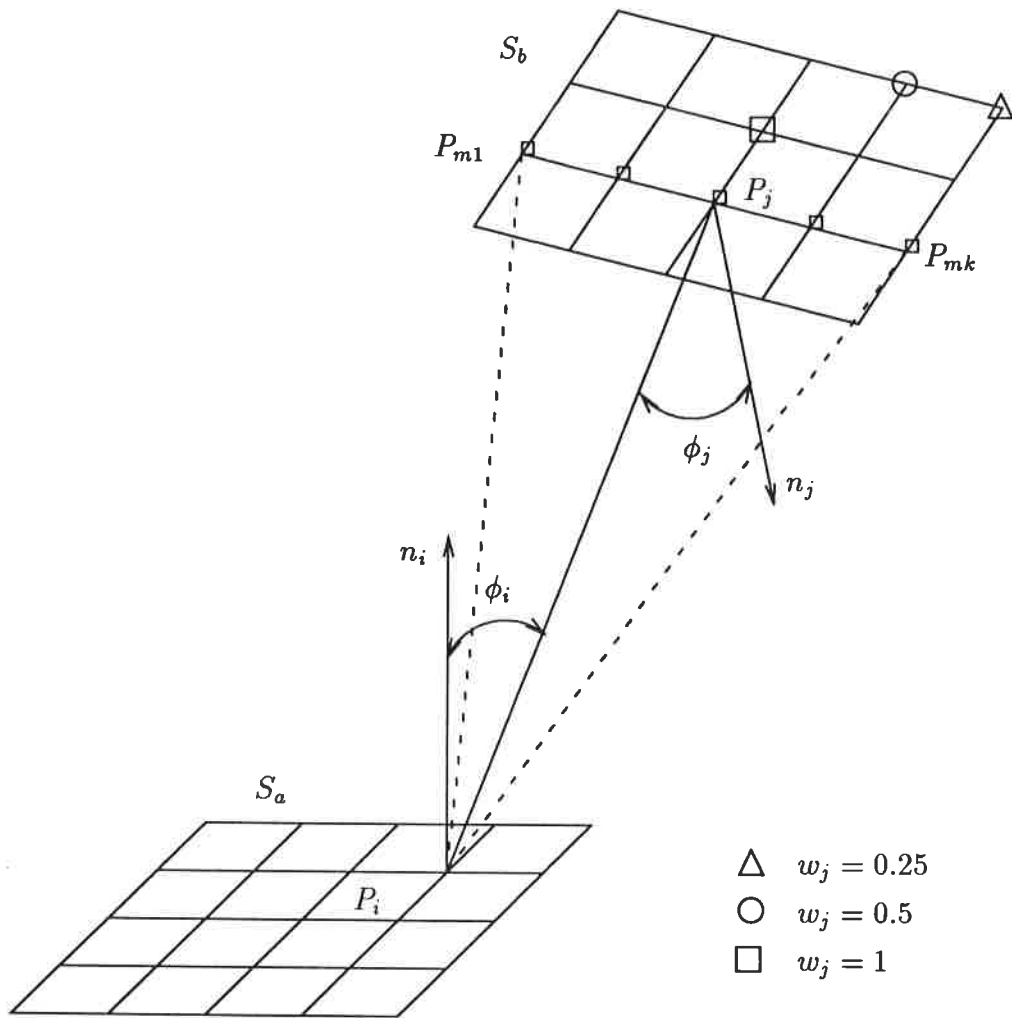


Figure 12: Nishita-Nakamae's configuration

$$F_{ij} = \frac{d(S_a, P_j)d(S_b, P_i)}{\pi r^4} A_j \quad (10)$$

where,

$d(S_a, P_j)$ is the distance from point P_j
to the plane of surface S_a

In fact, F_{ij} must be multiplied by a so called “weighting coefficient” w_j that takes into account whether P_j is a vertex of the surface, belongs to one of its edges or other (figure 12).

Function HID of equation 7 (referred in [12] as *shadow function* v_{ij}) is evaluated for a complete row of patches’ vertices at a time (points marked with small squares in figure 12). First, polyhedra between surfaces S_a and S_b are selected. A second selection gives a set of objects intersecting the triangle defined by P_i and the row $\overline{P_{ml}P_{mk}}$. Finally, contour lines (silhouettes) of these polyhedra are used to clip the row $\overline{P_{ml}P_{mk}}$.

Use of simple solutions computed analytically.

Now, points are casted from a point (vertex of a surface’s patch) to sample points of a patch on another surface, in order to determine which of the sample points are visible from the former, so to perform the HID function. In fact, this is conceptually equivalent to the HID function in Nishita-Nakamae’s method, without using object coherence as they do.

Each of the visible sample points is associated with a piece of patch (figure 13.a) for which a delta form-factor must be computed. The point-to-patch form-factor will, finally, be obtained by summation of the delta form-factor (figure 13.b). Wallace et al. [16] propose to approximate each piece of patch by a simple finite geometry for which an analytical solution for form-factor is available. Using disk geometry (figure 13.c), point-to-patch form-factor is approximated by,

$$F_{ij} = \frac{1}{n} \sum_{k=1}^n \frac{\cos \phi_{ik} \cos \phi_{jk}}{\pi r_k^2 + A_j/n} \quad (11)$$

2.2.3 Conclusions

We have presented different ways of approximating form-factors as well as of performing the HID function. Unfortunately, exhaustive comparisons of both efficiency and precision of the different methods are not available.

It is easy to see that other techniques can be derived by combining different form-factor approximations with different HID functions, either those presented in the above sections or other known rendering techniques and numerical methods.

2.3 Evolution of the hemi-cube based radiosity technique

Several improvements have been proposed to the basic hemi-cube technique [5, 3, 2]. Even though these improvements have been presented along with implementations where the hemi-cube technique was employed for the computation of form-factors, most of the basic ideas

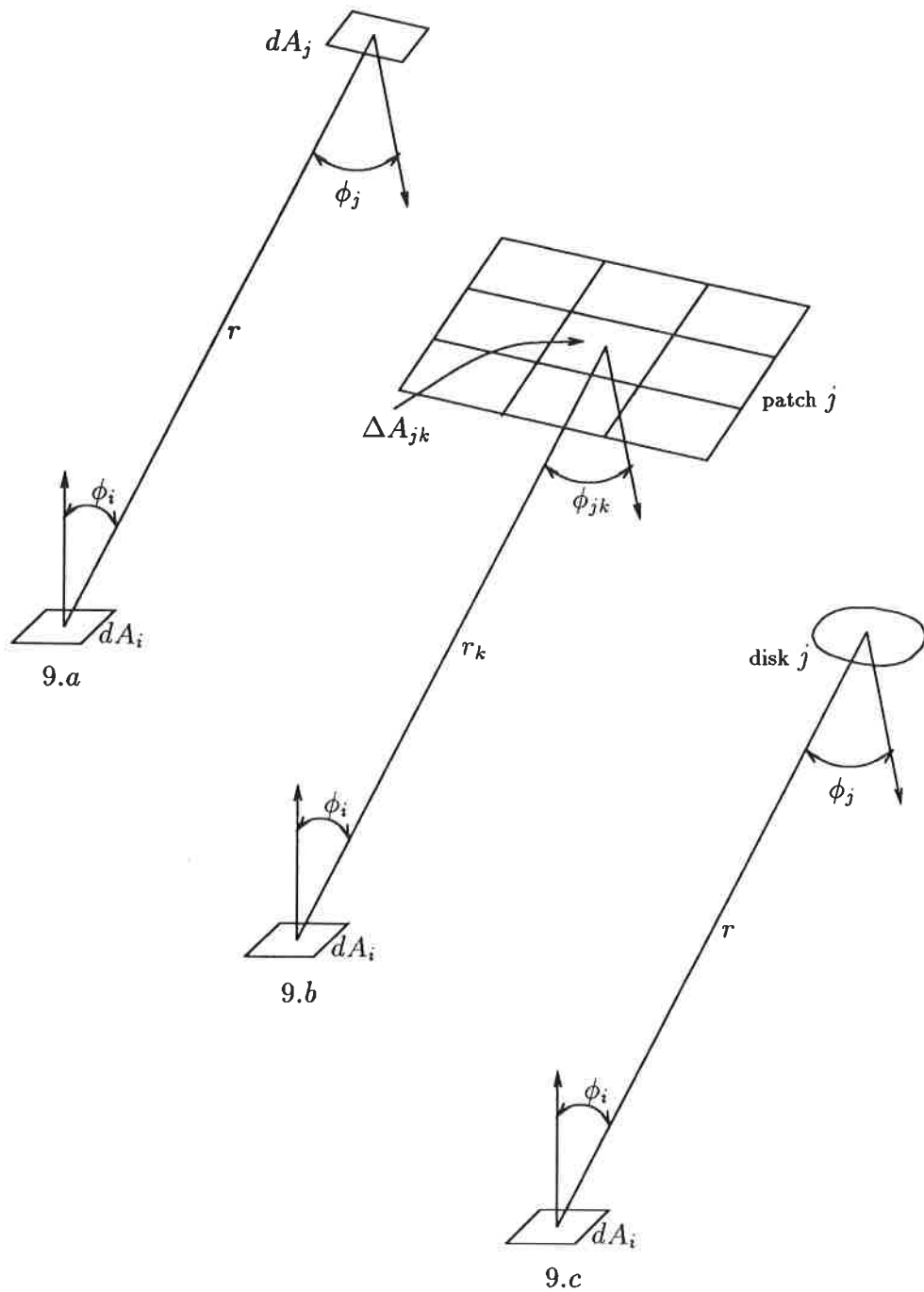


Figure 13: **a.** differential-to-differential area form-factor. **b.** Patch discretisation for form-factor computation from differential area to finite area. **c.** Approximating a piece of patch by an arbitrarily oriented disk.

which the improvements are founded on, could be easily applied to the other methods presented in the previous section.

2.3.1 Improving radiosity accuracy

The first two improvements were proposed by Cohen et al. [5] in order to:

1. Compute more accurate radiosity values with a small increase of computation cost.
2. Automatically compute form-factors with different levels of accuracy depending on the radiosity gradients.

As proposed in [4], the only way to improve the accuracy of radiosity was to subdivide patches into smaller ones. Because both form-factor calculations and simultaneous equations resolution result of order N^2 , other solutions different than increasing the number of patches are needed.

The solution proposed by Cohen et al. consist on considering the patch, from which form-factors are computed, subdivided in smaller entities, called *elements* (figure 14). Then, element-to-patch form-factors are computed and patch-to-patch form-factor is approximated by (figure 15),

$$F_{ij} = \frac{1}{A_i} \sum_{q=1}^R F_{qj} A_q \quad (12)$$

where

R is the number of elements.

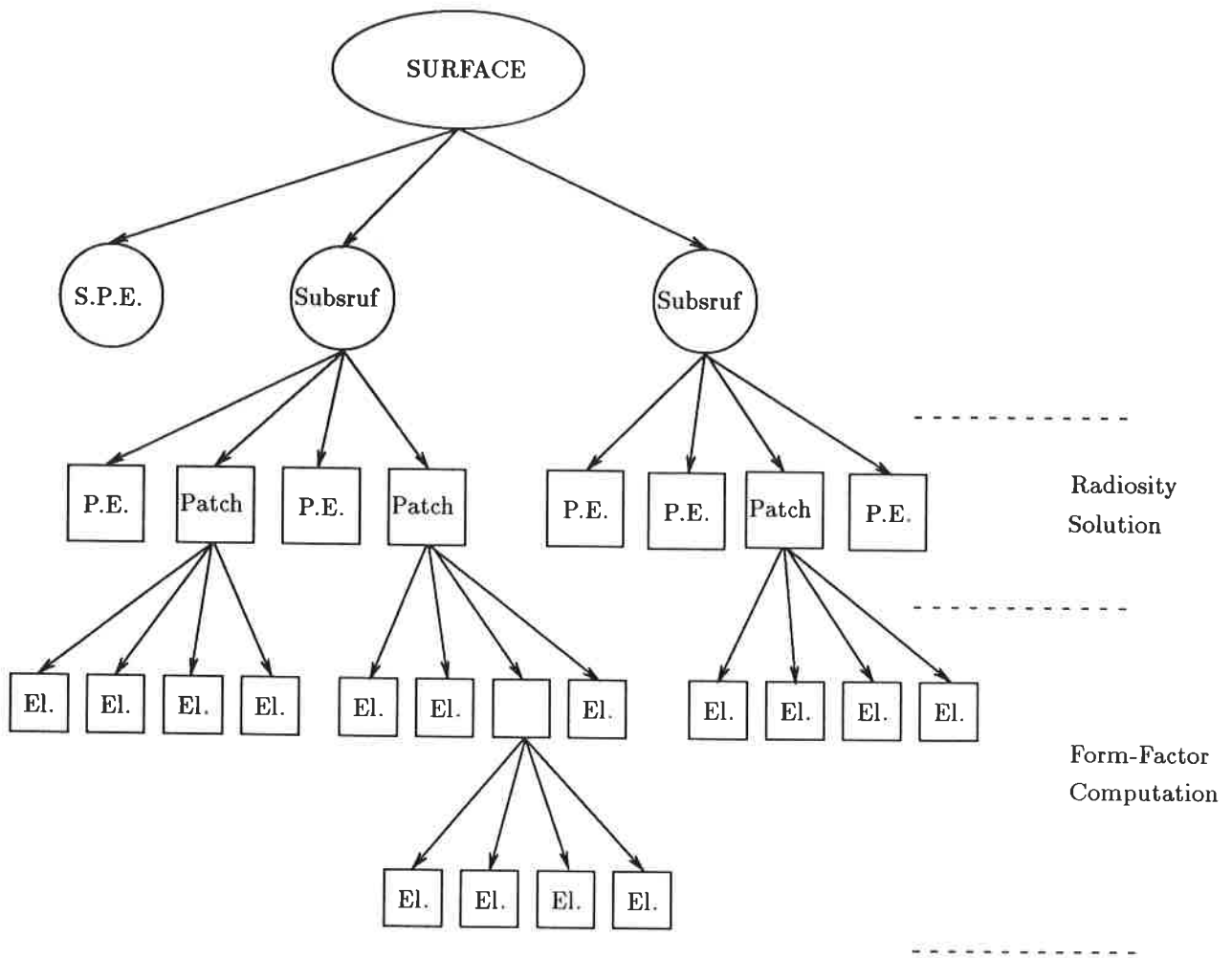
After that, the simultaneous equations are solved for patch-to-patch form-factors and the radiosity of each element q is found with,

$$B_q = E + \rho \sum_{j=1}^N B_j F_{qj} \quad (13)$$

where

Emission and reflectivity are assumed constant over the patch.

Computation of element-to-patch form-factors provides a way of predicting local variations of intensity within a patch without having to solve the global radiosity equations. Thus elements can be subdivided into smaller ones until local form-factor variations are small enough to predict small variations of radiosity.



- Surface* Bounded area with const. or continuous normal
- Subsurface* Part of a surface which can be divided by binary subdivision
- Patch* Part of a subsurface obtained by binary subdivision.
Reflectivity and emission are assumed constant over the patch.
- Element* Basic geometric unit acting as an illuminator (direct or indirect).
Part of a patch obtained by binary subdivision.
Smallest geometric unit acting as receiver of light from patches.
and as a discrete unit for numerical integration of the patch-to-patch
form-factor.

Figure 14: Surface discretisation

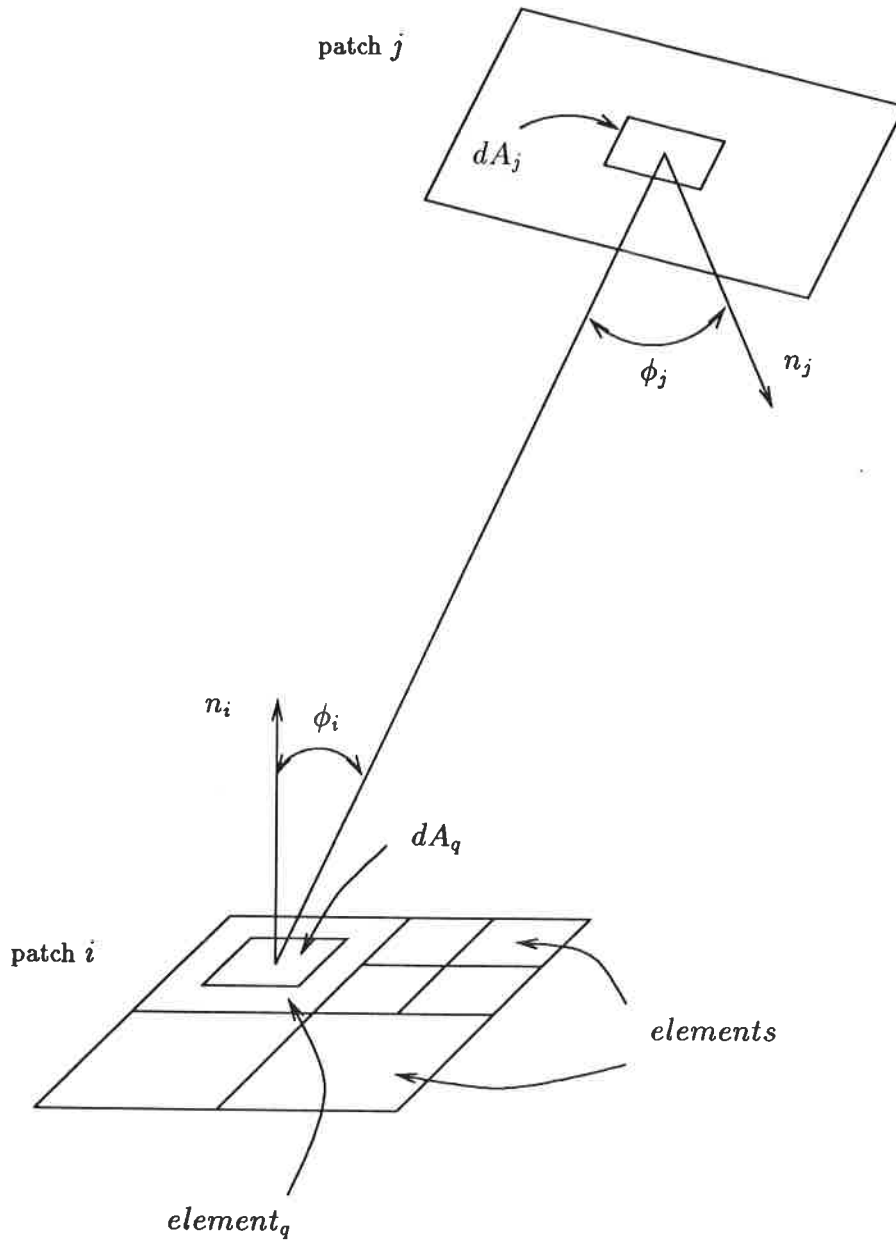


Figure 15: Patch subdivision into elements

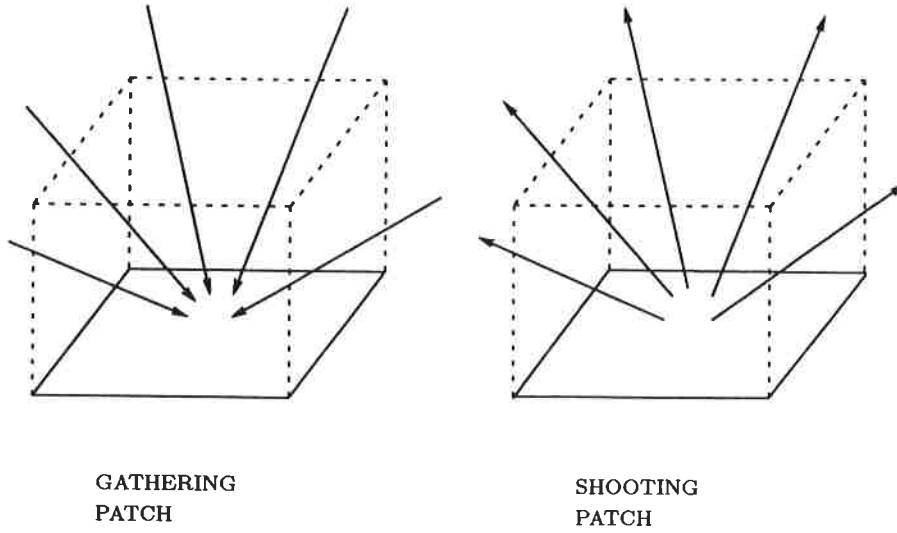


Figure 16:

2.3.2 Progressive refinement

Cohen et al. [3] presented a second set of improvements on the basis that “for rendering by *progressive refinement*, an important criterion is the time required to achieve a useful as opposed to complete solution”. Then they proposed a method which allows one to quickly obtain useful solutions using the radiosity technique. This strategy is based on the two following points:

1. The evaluation of a row in equation 2 gives an estimate of a patch radiosity (*gathering patch*) based on current estimates of the radiosity of the other patches. So, only the radiosity of one patch is updated at each step of one iteration by means of performing,

$$B_i = E_i + \sum_{j=1}^N \rho_i F_{ij} B_j \quad (14)$$

Instead of this, Cohen et al. proposed to reverse the strategy (figure 16) to find a method in which all but one patch (*shooting patch*) are updated at each step. This means to compute the contribution of one patch to the radiosity of all other patches, and is obtained in the following way,

$$B_j = B_j + B_i \rho_i F_{ji} \quad \forall j \neq i \quad (15)$$

where

$$F_{ji} = F_{ij} \frac{A_i}{A_j}$$

2. In a real environment, the contribution of each patch to the global illumination varies a lot from light sources to patches with a small reflectivity. So, a good estimation of the global illumination will be obtained faster if patches with bigger contributions are treated first in the procedure described in point 1.

This method proceeds gracefully to the complete radiosity solution and provides intermediate results good enough for certain applications. In order to improve these intermediate results, Cohen et al. proposed the use of an *ambient* term like in earlier illumination models. Now this term is calculated from,

- an estimated *reflectivity of the environment*,
- a *form-factor* of each patch *to the whole environment*,
- the current estimations of the radiosity values.

Thus, this term evolves with the global illumination approximations and it is just used for rendering purposes. Then, if B_i is the current estimated value for patch i , its centre point will be rendered with intensity,

$$B'_i = B_i + \rho_i \text{Ambient}$$

2.4 Recent advances

Recently, Baum et al. [2] presented an interesting discussion about errors introduced by *progressive refinement*. Then they propose an hybrid method to compute form-factors, which combines the hemi-cube technique with analytical solutions.

When geometric assumption used by the hemi-cube approach are violated, an analytical solution based on Stoke's theorem is applied. This approach is also applied directly when surfaces of high emittance (specially light sources) are treated as emitters, because of their high contribution to the global illumination solution.

Now an hemi-cube is still used as a way of sampling visible elements from the patch under study. When visible elements are known, a form-factor is computed analytically for each pair pixel-subelement. Finally, a summation of these form-factors, weighted by the area of the pixel projected onto the subelement, is performed.

Often, the visibility reciprocity between a shooting patch and an element is also violated. In this case, Baum et al. propose to subdivide the patch until each subpatch is "fully visible or fully hidden" from all elements in the environment, and present two methods to detect when subdivision is necessary.

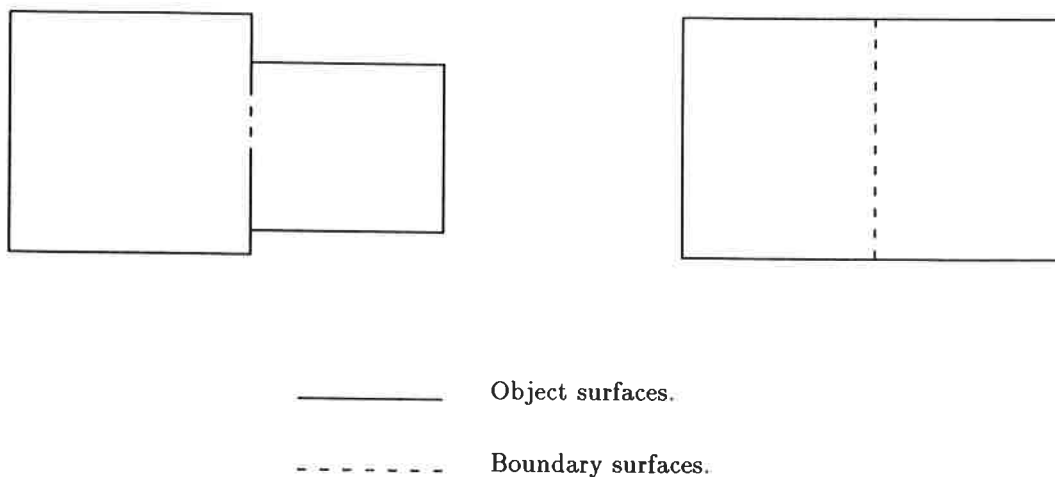


Figure 17: Two different examples of *local environment*

Wallace et al. [16] have also discussed recently the violation of the hemi-cube geometric assumptions and have proposed a ray tracing based technique to compute form-factors as we have explained in section 2.2. Now, the computation of form-factor is adapted to the required accuracy by means of shooting an adequate number of rays which may be different for the calculation of each form-factor. On the other hand, Wallace et al. sample receiving elements at vertices instead of centre points as an antialiasing method.

These two recent techniques [2, 16] should be discussed and compared quietly; but two first conclusions are clear:

1. Both retain the *progressive refinement* strategy.
2. Both tend, in some way, to *separate visibility computation from form-factor calculation*.

3 Subdividing the environment

Because both, form-factor computation and radiosity equation solution are $O(N^2)$, it seems obvious that one of the goals of research in this field is to reduce N . Two contributions [11, 17] have been made recently in this direction. In both papers authors propose to subdivide the environment in *subsystems* or *local environments*. Boundaries must then be defined to separate local environments, in order to be able to formulate the radiosity equations for each subsystem. In this way, patches of a subsystem are of two types: *object patches* and *boundary patches* (figure 17).

To solve the different local radiosity equations, the characteristics of boundary patches must be defined. It is easy to see intuitively that these characteristics depend on the properties of neighbouring local environments. So local radiosity equations can not be solved independently in the general case; even though they may become almost independent under special circumstances.

This kind of techniques must mainly deal with two aspects:

- Boundary patches modelisation.
- Systems of equations resolution.

In [11] authors propose a method (for special cases) where boundary patches are modelled by black surfaces, whereas [17] consider boundaries as being transparent. To solve the equations, the former propose to compute an approximated solution for each subsystem considering the others as light emitters. When all the subsystems have been treated, the process starts again with the first subsystem using the previous approximation. Nevertheless, [17] use the subdivision technique to express form factors between a patch i of a local environment and patch j of another as a combination of form factors of i and j to boundary patch k ($F_{ij} = f(F_{ik}, F_{kj})$). Finally they solve a set of global equations.

4 Conclusions

As summarized in figure 4 radiosity technique consist of four steps,

1. Discretization of the environment.
2. Form-factor computation.
3. Radiosity solution.
4. Rendering.

Radiosity literature deal, mainly, with steps 2 and 3 which have been analysed in this paper.

Form-factor computation (without accounting for occlusions) may be performed in several ways based on different hypothesis.

ccl.1 Comparisons between them should be made.

ccl.2 Feasibility of combining this different methods should be studied along with criteria allowing to choice the suitable way for each case. In fact, this kind of *hibrid form-factor computation techniques* has already been treated by Baum et al. [2]; but other solutions *combining hemi-cube and ray tracing* can be considered.

When occlusions take place function *HID* is also performed in different ways. Ray tracing and z-buffering (hemi-cube) methods have been basically kept; but,

ccl.3 Other known visibility computation strategies may be applied within *HID* (BSP like preprocess, for exemple).

ccl.4 New visibility strategies should be developped, based on different types of coherence;

- Between the five projection planes of an hemi-cube.
- Between neighbouring patches (patches belonging to the same surface).

As in “pure” form-factor computation, *hibrid HID functions* could be considered, where critical areas would be treated by more accurate (and expensive) *HID* functions.

From the point of view of step 3, **radiosity solution**, two major strategies have been proposed and kept by recent research contributions,

ccl.5 Adaptative subdivision depending on estimated radiosity gradient.

ccl.6 Progressive refinement.

Even though, these two techniques have been proposed along with the use of hemi-cube, they

ccl.7 Could be applied to other form-factor computation methods.

For both, reducing computation cost of steps 2 and 3, and possibly ofering good way for parallel solutions,

ccl.8 Strategies to divide the environment on subsystems to be processed “independently” should be studied deeper.

Moreover, the above conclusions derived from the analysed research contributions, other almost unexplored aspects of radiosity which should be studied are,

ccl.9 Relationship between surface subdivision (on patches and elements) and the different way of sampling the environment (by hemi-cube or ray tracing) from these surface entities, as well as the relationship between errors introduced in these different levels of discretization (figure 18).

ccl.10 Application of radiosity in dynamic environments.

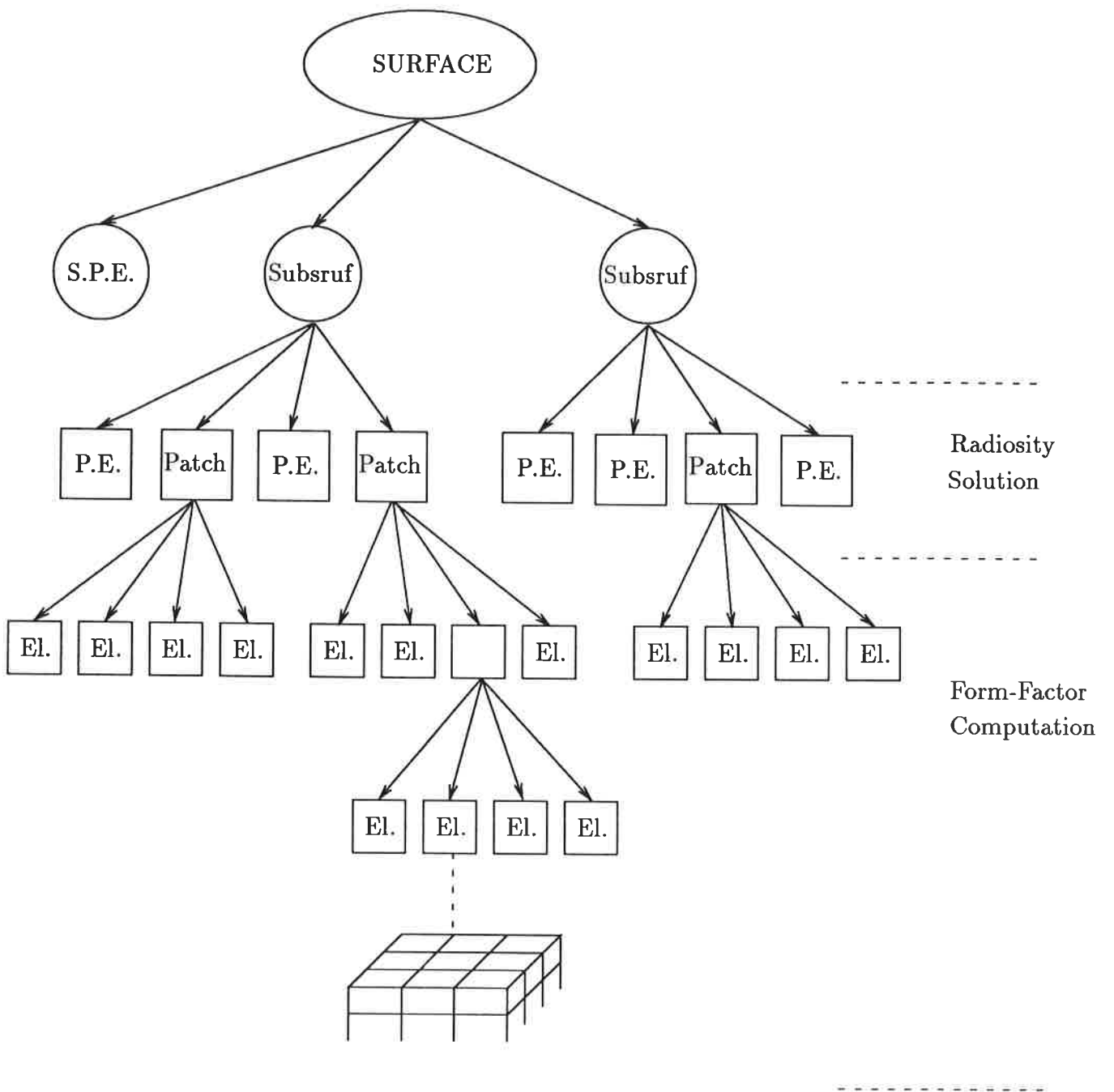


Figure 18: Discretization levels

References

- [1] ARVO, J. Backward ray tracing. In *Developments in Ray Tracing: Course Notes*, ACM SIGGRAPH, 1986.
- [2] BAUM, D. R., RUSHMEIER, H. E., AND WINGET, J. M. Improving radiosity solutions through the use of analytically determined form-factors. In *ACM SIGGRAPH Conf. Proc.* (july 1989), pp. 325–334. Computer Graphics vol.23 num.3.
- [3] COHEN, M. F., CHEN, S. E., WALLACE, J. R., AND GREENBERG, D. P. A progressive refinement approach to fast radiosity image generation. In *ACM SIGGRAPH Conf. Proc.* (august 1988), pp. 75–84. Computer Graphics vol.22 num.4.
- [4] COHEN, M. F., AND GREENBERG, D. P. The hemi-cube. a radiosity solution for complex environments. In *ACM SIGGRAPH Conf. Proc.* (july 1985), pp. 31–40. Computer Graphics vol.19 num.3.
- [5] COHEN, M. F., GREENBERG, D. P., IMMEL, D. S., AND BROCK, P. J. An efficient radiosity approach for realistic image synthesis. *IEEE Computer Graphics and Applications* 6, 3 (1986), 26–35.
- [6] GORAL, C. M., TORRANCE, K. E., GREENBERG, D. P., AND BATTAILE, B. Modeling the interaction of light between diffuse surfaces. In *ACM SIGGRAPH Conf. Proc.* (july 1984), pp. 213–222. Computer Graphics vol.18 num.3.
- [7] GREENBERG, D., Ed. *Radiosity: Course Notes*. ACM SIGGRAPH, 31 july - 4 august 1989.
- [8] GREENBERG, D. P., COHEN, M. F., AND TORRANCE, K. E. Radiosity: a method for computing global illumination. *The Visual Computer* 2, 5 (1986), 291–297.
- [9] HALL, R. *Illumination and Color in Computer Generated Imagery*. Springer-Verlag, New York, 1988. ISBN 0-387-96774-5.
- [10] KAJIYA, J. T. The rendering equation. In *ACM SIGGRAPH Conf. Proc.* (august 1986), pp. 143–150. Computer Graphics vol.20 num.4.
- [11] NEUMANN, L., AND NEUMANN, A. Photosimulation: interreflection with arbitrary reflectance models and illumination. *Computer Graphics Forum* 8, 1 (1989), 21–34.
- [12] NISHITA, T., AND NAKAMAE, E. Continuous tone representation of three-dimensional objects taking account of shadows and interreflection. In *ACM SIGGRAPH Conf. Proc.* (july 1985), pp. 23–30. Computer Graphics vol.19 num.3.
- [13] SEIGEL, R., AND HOWELL, J. R. *Thermal Radiation. Heat Transfer*. Hemisphere Publishing Corp., 1978.

- [14] SHAO, M., PENG, Q., AND LIANG, Y. A new radiosity approach by procedural refinements for realistic image synthesis. In *ACM SIGGRAPH Conf. Proc.* (august 1988), pp. 93–101. *Computer Graphics* vol.22 num.4.
- [15] SILLION, F., AND PUECH, C. A general two-pass method integrating specular and diffuse reflection. In *ACM SIGGRAPH Conf. Proc.* (july 1989), pp. 335–344. *Computer Graphics* vol.23 num.3.
- [16] WALLACE, J. R., ELMQUIST, K. A., AND HAINES, E. A. A ray tracing algorithm for progressive radiosity. In *ACM SIGGRAPH Conf. Proc.* (july 1989), pp. 315–324. *Computer Graphics* vol.23 num.3.
- [17] XU, H., PENG, Q., AND LIANG, Y. Accelerated radiosity method for complex environments. In *EUROGRAPHICS Conf. Proc.* (september 1989), pp. 51–61.