A study about the form factors kernel function

Eric Zéghers[1] **Christophe Renaud**[2]

[1]Laboratoire de Logique, Algorithmique et Informatique de Clermont-Ferrand 1
zeghers@iutsux01.u-clermont1.fr

[2]Laboratoire d'Informatique du Littoral
Université du Littoral Côte d'Opale
renaud@lil.univ-littoral.fr

ABSTRACT

This paper presents a study of the form factors (FF) function kernel. The accuracy of FF estimate is known as a difficult problem when simulating radiative energy exchanges between objects inside an enclosure. By carefully studying the FF function between two polygons we are able to propose a very interesting characterization of its behaviour according to the relative distance between those polygons (general form of the function, location and height of its unique maximum, effect of polygons orientation and distance, ...). According to the results of this study we estimate the FFs between any two polygons by distinguishing the areas where the kernel has smooth variations from the those where it changes quickly. A fine integration is thus performed for the more varying parts of the kernel whereas the other parts are computed more easily. We show that even a very simple implementation of our approach provides accurate estimates of the FF close to the results provided by the Schroeder analytical formula [8] in a time 8 up to 10 times faster. Moreover our approach does not suffer from lack of accuracy when surfaces are very closed from each other thus outperforming classical methods.

Keywords: Form factors, kernel function, integration

1 Introduction

Radiosity simulates global illumination by taking into account the light exchanges between object surfaces assuming perfectly lambertian surfaces. Energy exchanges require to compute geometrical quantities known as form factors (FF). FFs are often the heart of a lot of application code, both in terms of quality (a high accuracy is required for correctly taking into account energy exchanges) and in terms of quantity (their estimate generates a very high computational cost). Previous researches about FF estimate have been performed through the heat transfer theory [2]. Due to the high complexity of their formula (double area integral), catalogs [5] have been proposed which list analytical expressions for some particular geometrical shapes. FFs estimate for more complex problems are then generally extrapolated from those simplest environments.
More recently researches have been performed in the field of computers graphics where FFs have to be computed for 3D scene illumination. Many specific algorithms have been proposed (hemicube, hemiplane, disk-to-point ray tracing, Monte Carlo, etc) [1] [9] providing an extended survey about these techniques. However all those methods suffer from a lack of accuracy especially when surfaces are closed from each other because of the inconstancy of their estimate. Furthermore they use sampling approaches for computing the FFs between distant surfaces, those approaches being prone to unaccuracy. Recent work [8] has provided a general analytical formula for the FF estimate between any two planar polygons. But this formula requires still to approximate some computation and has a very high computational cost. Then it has not been designed to take into account partial visibility between the two polygons. However due to its accuracy we will use this formula as a reference when evaluating the results provided by our approach.
The rest of the paper is organized as follows; section 2 recalls some basics about FFs and intro-

duce a new formulation of the FFs kernel. In section 3 we study the FF kernel function extensively and hightlight the parameters that have the main influence on its value. Then we exploit the acquired knowledge of the kernel function for developping in section 4 a first simple way of estimating the FF between any two polygons. The results we present in this section highlight the interest of our approach and allow us to expect large new enhancements in FF computation.

2 Form factor between two polygons

We assume for our study that the two polygons are fully visible from each other. The way in which occlusions could be taken into account will be suggested later in the paper.

2.1 Form factors kernel from the receiver

According to the notations of figure 1 the FF between two polygons is given by :

$$F_{ij} = \frac{1}{\pi A_i} \int_{A_i} \int_{A_j} \frac{\cos(\theta_x)\cos(\theta_y)}{r^2}\, dA_j\, dA_i$$

Furthermore, we define a new coordinate system $(O, \vec{s}, \vec{t}, \vec{u})$ attached to the receiving polygon. This coordinate system and notations we will use in the following are both represented in figure 1.

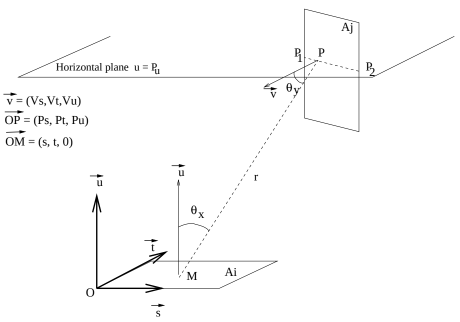


Figure 1: Geometry for the form factors kernel study

Thus we can write :

$$\overrightarrow{OM} = (s, t, 0)^T \quad \overrightarrow{OP} = (P_s, P_t, P_u)^T$$

$$r = \|\overrightarrow{MP}\| \quad \vec{v} = (V_s, V_t, V_u)^T \text{ with } \quad \|\vec{v}\| = 1$$

$$\cos(\theta_x) = \frac{\vec{u}.\overrightarrow{MP}}{\|\overrightarrow{MP}\|} \quad \text{and} \quad \cos(\theta_y) = \frac{\vec{v}.\overrightarrow{PM}}{\|\overrightarrow{MP}\|}$$

The FF kernel between emitter A_j and receiver A_i is defined as $K = \frac{\cos(\theta_x)\cos(\theta_y)}{r^2}$. It can thus be rewritten as :

$$K_{\vec{v}}(s, t, P_s, P_t, P_u) = -P_u \frac{Vs(P_s - s) + Vt(P_t - t) + V_u P_u}{((P_s - s)^2 + (P_t - t)^2 + P_u^2)^2} \quad (1)$$

2.2 New form for the kernel

In order to study the properties of the FF kernel we chose to define K as a function f with two variables X and Y and 4 parameters V_s, V_t, V_u and P_u. According to this choice we assume that we study the properties of f in the plane $(u = P_u)$, $P_u \neq 0$. X and Y are defined as $X = P_s - s$ and $Y = P_t - t$, the relative gap between any two points of emitter and receiver respectively. Thus :

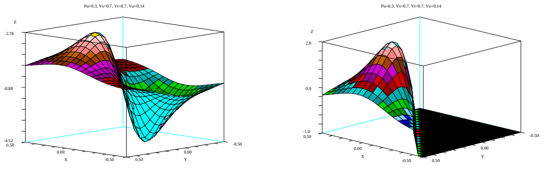
$$f(X, Y) = -P_u \frac{V_s X + V_t Y + V_u P_u}{(X^2 + Y^2 + P_u^2)^2} \quad (2)$$

This function will allow us to study the kernel changes in the plane $(u = P_u)$ according to the changes in $(P_s - s)$ and $(P_t - t)$. The intersection between this plane and the emitter is generally[1] a segment $[P_1, P_2]$ (eventually of zero-length) with $P1 = (P1_s, P1_t, P_u)$ and $P2 = (P2_s, P2_t, P_u)$.

2.2.1 Kernel's properties

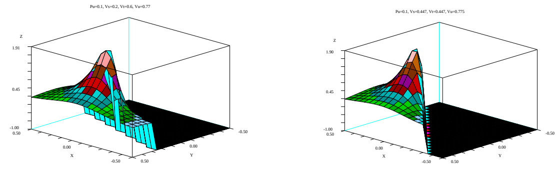
Function $f(X, Y)$ has the following obvious symmetries that allows us to limit our study to the cases in which $V_s \geq 0$ and $V_t \geq 0$:$f_{V_s,V_t,V_u,P_u}(X, Y) = f_{-V_s,V_t,V_u,P_u}(-X, Y) = f_{-V_s,-V_t,V_u,P_u}(-X, -Y) = f_{V_s,-V_t,V_u,P_u}(X, -Y)$. When $f(X, Y) \leq 0$ the two points are mutually unvisible either because the two polygons are backfaced from each other or a part of one of them is "below" the plane of the other. Care must thus be taken to discard negative values of the kernel during our study (and further during the FF estimate). In the following when showing the f's changes, the (X, Y) values that generate a negative value will be represented with the constant value -1. This will allow us to visually differentiate low f values from negative one's. Figures 2.a and 2.b show examples of changes for $f(X, Y)$ with $P_u = 0.3, V_s = 0.7, V_t = 0.7, V_u = 0.14$:

[1] If the emitter is horizontal with height P_u (in the $(O, \vec{s}, \vec{t}, \vec{u})$ coordinate system), the intersection is represented by the emitter's area.



(a) (b)

Figure 2: The kernel changes (a) with negative values (b) by removing negative values.



(a) (b)

Figure 3: Influence of V_s and V_t changes on the kernel, highlighting the rotation of f around the vertical axis with origin in $(0, 0)$.

3 Studying the FF kernel function

3.1 Extrema of function f

We derived in equation 2 a new expression for the FF kernel function :

$$f(X, Y) = -P_u \frac{V_s X + V_t Y + V_u P_u}{(X^2 + Y^2 + P_u^2)^2}$$

In order to find f extrema we have to find values such that f's partial derivative are zero values. Assuming $P_u \neq 0$ we have :

$$\frac{\partial f}{\partial X} = 0 \Leftrightarrow 3V_s X^2 + 4V_t XY - V_s Y^2 + 4P_u V_u X - V_s P_u^2 = 0 \quad (3)$$

$$\frac{\partial f}{\partial Y} = 0 \Leftrightarrow 3V_t Y^2 + 4V_s XY - V_t X^2 + 4P_u V_u Y - V_t P_u^2 = 0 \quad (4)$$

By studying quadratic forms associated with equations 3 and 4 it can be shown that those two equations correspond to a system of two separate hyperbolas when $V_s \neq V_t$.
Thus there are potentially 0 up to 4 points open to be an extrema. In order to know whether each of these points is a minima, a maxima or a stationnary point, we should compute the second partial derivatives. But the equations we obtained were too complex and thus useless for give any information about the sign of these one's.

3.2 Influence of the V_s and V_t parameters

In this part we assume that V_u and P_u are fixed and that only V_s and V_t are varying.
We have the following relation between the components of the emitter normal vector $\vec{v} = (V_s, V_t, V_u)^T$: $V_s^2 + V_t^2 + V_u^2 = 1$. Thus when only V_s and V_t change it physically means that the emitter normal extemity is moved along a circle of radius $R = \sqrt{1 - V_u^2}$ lying in an horizontal plane (orthogonal with $\vec{u}$). This rotation is characterized by an angle θ with :

$$\begin{pmatrix} V_s' \\ V_t' \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} V_s \\ V_t \end{pmatrix} \quad (5)$$

Using system 5 it is easy to show that $f_{V_s,V_t}(X, Y) = f_{V_s',V_t'}(X', Y')$ with X' and Y' issued from the same rotation of angle θ around the origin $X = Y = 0$. This property appears also in figure 3 : when V_s and V_t change function f is rotated around the vertical axis with its origin in $X = Y = 0$. This property is very important for the study of f : we can now study only special cases for V_s and V_t and then extrapolate more general results by a rotation.

3.3 Kernel study for $V_s = V_t$

For $V_s = V_t$ (and thus $V_u^2 = 1 - 2V_s^2$), function $f(X, Y)$ is symmetrical about the first bisecting plane. By subtracting equation 4 from 3 an extrema of f verifies :

$$4V_s X^2 + 4P_u V_u (X - Y) - 4V_s Y^2 = 0$$

$$\text{thus} \qquad (X - Y)(V_s(X + Y) + P_u V_u) = 0 \quad (6)$$

a) If $V_s = V_t = 0$ we have $V_u = \pm 1$ and using equation 6 the maxima verifies $X = Y$ ($P_u \neq 0$). We got thus $f(X, X) = \frac{-P_u^2 V_u}{(2X^2 + P_u^2)^2}$ and inequality $f(X, X) \geq 0$ implies that $V_u = -1$. Thus the maxima of f verifies $X = Y = 0$ and his value is $f(0, 0) = \frac{1}{P_u^2}$

b) If $Y = \frac{-P_u V_u}{V_s} - X$ (and $V_s \neq 0$) equation 3 gives :

$$2V_s^2 X^2 + 2P_u V_u V_s X + P_u^2 (V_u^2 + V_s^2) = 0 \quad (7)$$

The value of equation 7's discriminant is $\Delta = -V_s^2 P_u^2 (V_u^2 + 2V_s^2)$. This value is always negative. Equation 7 thus don't have any real solution and the potential maxima cannot belong to this particular cross section.

c) $Y = X$ (and $V_s \neq 0$) : If function f presents some extrema they should lye along the first bisecting line. Let us recall that we have $2V_s^2 =$

$1 - V_u^2$. We now set function g in the following way :

$$g(X) = f(X, X) = \frac{-P_u(2V_s X + P_u V_u)}{(2X^2 + P_u^2)^2} \quad (8)$$

$$\text{with } g'(X) = 0 \Leftrightarrow 6V_s X^2 + 4P_u V_u X - V_s P_u^2 = 0 \quad (9)$$

The discriminant of equation 9 gives thus :

$$\Delta' = 4P_u^2 V_u^2 + 6P_u^2 V_s^2 = P_u^2(4V_u^2 + 3(1 - V_u^2)) = P_u^2(3 + V_u^2)$$

P_u being strictly positive there are two possible extrema :

$$X_{1,2} = -P_u \frac{2V_u \pm \sqrt{3 + V_u^2}}{6V_s}$$

We now compute $g(X)$ for those two points :

$$g(X_1) = \frac{1}{P_u^2} \frac{-V_u/3 + \sqrt{3 + V_u^2}}{\left(\frac{(2V_u + \sqrt{3 + V_u^2})^2}{9(1 - V_u^2)} + 1\right)^2}$$

We have $\sqrt{3 + V_u^2} > \sqrt{3}$ and $-V_u/3 > -1/3$ and thus $\boxed{g(X_1) > 0}$. In the same way, we have $\boxed{g(X_2) < 0}$. Thus function $g(X)$ has an unique positive maxima in point X_1 and an unique negative one in point X_2. Point X_2 has no interest because the corresponding points on the polygons are not visible from each other (see subsection 2.2). Thus function $f(X, Y)$ has an unique maxima when $V_s = V_t$. Its coordinates are :

$$MAX_{(V_s = V_t)} = (X = X_1, Y = X_1, Z = g(X_1)) \quad (10)$$

3.4 Maxima of the kernel function for any V_s and V_t values

The previous result is valid only when $V_s = V_t$. But we shown in subsection 3.2 that any change in the values of V_s and V_t was corresponding to a rotation of function f. Thus for any V_s and V_t we can find the unique maximum of f (equation 11) assuming that $V_s \neq 0$ or $V_t \neq 0$.

$$MAX = \begin{pmatrix} X_M = -P_u V_s \frac{2V_u + \sqrt{3 + V_u^2}}{3(1 - V_u^2)} \\ Y_M = -P_u V_t \frac{2V_u + \sqrt{3 + V_u^2}}{3(1 - V_u^2)} \\ Z_M = f(X_M, Y_M) \end{pmatrix} \quad (11)$$

Let us recall that for $V_s = V_t = 0$ the maxima is located in $(0, 0)$ with $f(0, 0) = \frac{1}{P_u^2}$.
Thus we are able both to locate and to measure the maxima for any given height P_u.

3.5 General shape of the kernel function : influence of parameters V_u and P_u

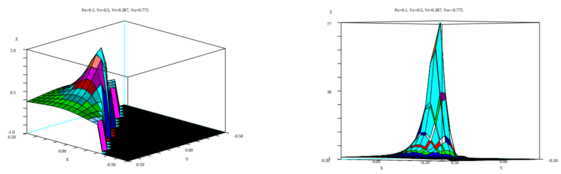
It is easy to show using polar coordinate system that function f tends to 0 for each radial directions. Thus the kernel function has a typical shape for any plane of height P_u : this shape is close from a tent linen that would be pulled up from a point for positive values and pulled down from another point for negative values. By restricting ourself to positive values (negative values involve no visibility between points) we will now characterize more precisely this shape by studying the influence of parameters V_u and P_u.

3.5.1 Influence of V_u

When both V_s and V_t have been set only two opposite values are available for parameter V_u :

$$V_u = \pm\sqrt{1 - V_s^2 - V_t^2}$$

$V_u < 0$ means that emitter is oriented towards the receiver (backwards for $V_u > 0$). As highlighted in figure 4 large changes on the kernel function can be expected from a change in the sign of V_u. However it is interesting to note that changing the



(a) (b)

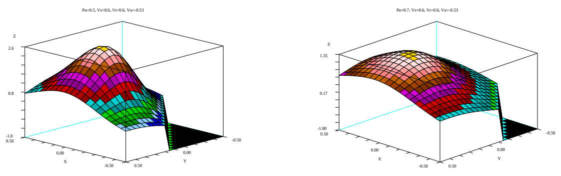
Figure 4: Influence of the sign change for V_u
$P_u = 0.1, V_s = 0.5, V_t = 0.387, V_u = \mp 0.775$

sign of V_u has a low influence on the tent spread (or cone spread) but a very large one on the value of the maxima.

3.5.2 Influence of P_u

Parameter P_u describes the relative height of emitter from the receiver. It is clearly dependent of the distance between the two polygons and thus it has a large influence on the kernel function (see the term $\frac{1}{r^2}$ in the kernel function definition). In figure 5 we drawn the kernel function for 2 values of parameter P_u (V_s, V_t and V_u remaining unchanged). The changes of the kernel function appear both in the location and the value of the maxima (see equation 11). But more important the spread of the cone is modified when

P_u changes. Using the tent linen analogy we could say that P_u is concerned with the strength used to pull up the tent. Another interesting feature



(a) (b)

Figure 5: Influence of P_u : $P_u = 0.5(a)$ or $0.7(b)$ $V_s = V_t = 0.6, V_u = -0.53$

is concerned with the cone spread : it appears to be correctly delimited by a circle of center the kernel function maxima and of radius $R = P_u$. This empirical feature could be more precisely investigated in futur works, allowing us to improve results of section 4.

Finally even though V_s, V_t and V_u are constant for any emitter it is important to note that P_u' value generally change across the emitter (except for parallel polygons). Thus P_u will be one of the main parameter to take into account when estimating the form factor.

3.6 Studying the kernel on $[P_1, P_2]$

In this part of our study we assume both that the two polygons are not parallel and that $P1_s \neq P2_s$.

3.6.1 Kernel expression on $[P_1, P_2]$

The equation of the projection of straight line (P_1, P_2) on $(O, \vec{s}, \vec{t})$ is given by (assuming $P2_s \neq P1_s$) :

$$Y = aX + b \text{ with } a = \frac{P2_t - P1_t}{P2_s - P1_s},\ b = P1_t - aP1_s \tag{12}$$

By restricting the kernel to this straight line and using equation 2, the kernel can be rewritten as :

$$f(X, aX+b) = -P_u \frac{(V_s + aV_t)X + V_u P_u + V_t b}{((1+a^2)X^2 + 2abX + Pu^2 + b^2)^2}$$

$\vec{v} = (V_s, V_t, V_u)^T$ is the emitter's normal vector. Thus we have $\vec{v}.\ \overrightarrow{P_1P_2} = 0$. But $\overrightarrow{P_1P_2}$ is colinear to vector $(1, a, 0)^T$. Thus we have $V_s + aV_t = 0$ and the kernel can be rewritten as :

$$f(X, aX+b) = -P_u \frac{V_u P_u + V_t b}{((1+a^2)X^2 + 2abX + Pu^2 + b^2)^2} \tag{13}$$

3.6.2 Constant values and kernel sign

Let us recall that vector $\vec{v}$ is constant across the emitter just as coefficient **a** of equation 12. Indeed the intersection of the different planes of relative height O_u with the emitter sustaining plane provides colinear straight lines. The projection of those lines onto the horizontal plane $(O, \vec{s}, \vec{t})$ provides parallel lines which slope **a** is constant. Furthermore P_u is constant and positive for any segment $[P_1, P_2]$.

If $V_t \neq 0$ the kernel value is nil only if $b = \frac{-V_u P_u}{V_t}$ (see equation 13). The straight line Δ whose equation is $Y = aX - \frac{V_u P_u}{V_t}$ has value zero in every points for the kernel and is parallel to line (P_1, P_2). When we study the change in the kernel value for segment $[P_1, P_2]$ parameter **b** is fixed. Thus the kernel has always the same sign along this segment. In other words, assuming no obstructions between the polygons a segment issued from the intersection between an horizontal plane of height P_u and the emitter is either fully visible or totally unvisible.

If $V_t = 0$ the kernel has always the same sign than $-P_u^2 V_u$. Thus emitter is fully visible if $V_u < 0$ and totally unvisible otherwise.

4 A first implementation

We derived a first implementation from the study we have previously presented. Although our current implementation appears to be simple it provides good results and allows us to expect interesting perspectives of research.

During our study of the FFs kernel we shown that its changes could be predicted. It is thus possible to integrate accurately this kernel in the parts where it changes strongly and reciprocally to integrate more roughly the parts where the changes are smooth. The interest of this approach is that we reduce considerably the computational cost while keeping an accurate estimate of the FF.

4.1 Determining the change zones

We previously shown that for any height P_u the kernel shape had typical features (see 3.5); we are able to locate its unique maxima and to know its value (see 3.4). Furthermore we provided an approximation of the conic shape spread centered around the maxima. Figure 6 shows the different relative locations of segment $[P_1, P_2]$ with regard to the cone spread (see 3.6.2), highlighting 5 representative cases.

The pseudo-algorithm that appears in figure 7

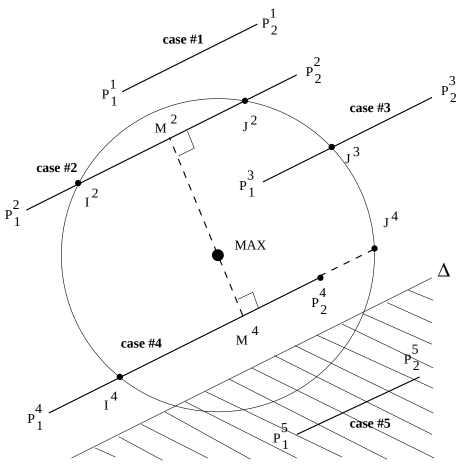


Figure 6: Top view of the kernel function : the circle represents the conic aperture of the function ; the kernel is negative in the hatched part.

summarizes our approach for intregating along $[P_1, P_2]$. Indices i of $P^i_{1,2}$, $M^i_{1,2}$, $I^i_{1,2}$ and $J^i_{1,2}$ are used according to one of the five representative cases previously mentioned. The parameters R (radius) and MAX (maxima of the function) are required for the computation of points I and J used during integration. The function **lin** per-

Integration_on_[P1,P2]$(P1, P2, R, MAX)$
case [P1, P2]
#1 : val=lin(P^1_1,P^1_2)
#2 : val=lin(P^2_1,I^2) + simp(I^2,J^2) + lin(J^2,P^2_2)
#3 : val=simp(P^3_1,J^3) + lin(J^3,P^3_2)
#4 : val=lin(P^4_1,I^4) + simp(I^4,J^4) - lin(P^4_2,J^4)
#5 : val=0
return val

Figure 7: Pseudo-code for the integration along [P1,P2]

formed an integration of order 1 and is used for any segment lying outside the cone. When inside the cone, we use the function **simp** that performs an integration of order 2, by using a third point M^i which is the center of the integrated segment. In case #2 the point M^2 is used ; its the center of $[I^2, J^2]$ and is the maxima along this segment (M^2 is the point of $[I^2, J^2]$ that is the closest of MAX, the global function maxima). The polynome of order 2 which runs through points I^2, M^2, J^2 can then be considered to close to the conic shape, providing a good accuracy during integration. For this reason we don't integrate case #4 along $[I^4, P^4]$, using a integration of order 2. We choose rather to integrate along $[I^4, J^4]$ and then to subtract the integration (of order 1) along $[P^2, J^4]$. In case #5 we do not have to integrate anything since the kernel has negative values. Thus integrating along segment $[P_1, P_2]$ requires 2 up to 5 points.

4.1.1 Some results

We compared our approach with some other numerical integration methods. More precisely we used the Gauss quadrature method [10] which gives better results for the same number of points than classical approaches like trapezium, Simpson or Newton-Cotes[2]. In the following table we put the numerical results we obtained with our approach as compared with thoses provided by a Gauss Quadrature using $2 * 10$ points used as the reference solution (10 point for each half segment), 6 points and 10 points. The different cases that appear in the table are those of figure 6. Our

Case	our approach	Gauss 2*10 pts	Gauss 6 pts	Gauss 10 pts
#1	**0.3062** (2pts)	0.3301	0.3301	0.3301
#2	**16.005** (5pts)	15.676	9.6580	15.103
#3	**5.8319** (4pts)	5.8320	5.83198	5.8320
#4	**7.3008** (5pts)	7.3727	7.36669	7.3727

Table 1: Numerical results from some typical cases for the integration along $[P_1, P_2]$ (see figure 6 for an overview of the cases).

approach provides thus results close from the values obtained by a Gauss method using 6 to 10 points, but with less points. Furthermore it ensures a general stability of its numerical results. Let us note that for frequently encountered case #2 Gauss Quadrature requires up to 10 points in order to capture the kernel features. In case #1 however our approach sometimes appears to provide inaccurate results. This problem appears only when the maxima is not reached in P^1_1 nor in P^1_2 (when the line $\Delta^\perp$, running through MAX, cut $[P^1_1, P^1_2]$). This problem could be solved by linearly integrating each new part of the segment. But this would increase the computation cost and we suspect that the low part of this case on the global value of the FF does not justify such approach.
A great advantage of our approach appears when polygons are close from each others. Indeed a low distance between polygons involves large changes in the kernel function with a narrow cone spread.

[2]Our approach does not use Gauss integration method because it requires to compute new points (it does not use the extremities of the segment), thus reducing the efficiency of the computation

The knowledge *a priori* of the kernel function allows us to control efficiently such cases[3].

4.1.2 Full FF estimate

We give in figure 8 the pseudo algorithm we use for computing the form factor between two fully visible patches A_i and A_j. In order to

```
Compute_FF(Ai, Aj)
  {s[1],s[2],s[3]}x{t[1],t[2],t[3]}=PtsOnReceiv(Ai);
  {Pu[1],...,Pu[10],h}= AltAndHeightTrapez(Aj);
  V⃗ = Normal(Aj);
  ValFormFactor = 0;

  for (k = 1; k ≤ 10; k++){
    R = ConicAperture(Pu[k]);            // see 3.5.2
    MAX = Maximum(Pu[k], V⃗);             // see 3.4
    sum1 =0;
    for (m = 1; m ≤ 3; m++){
      sum2 =0;
      for (n = 1; n ≤ 3; n++){
        [P1, P2] =Inc[P1,P2](Pu[k], Aj, s[m], t[n]);
        sum2 += w3[n]*Integrate(P1,P2,R,MAX);
      }
      sum1 += w3[m]*sum2;
    }
    ValFormFactor += sum1;
  }
  return h/(πAi) ValFormFactor;
```

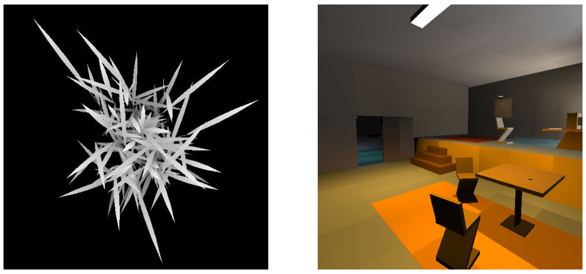
Figure 8: Pseudo algorithm : full FF estimate

fully integrate the FF we choose 3 Gauss points along each receiver's dimension (thus 9 integration points across surface A_i)(function **PtsOnReceiv**). According to our study however integration along emitter height (P_u) must be very precise. For the results we present emitter has been cut along 10 altitudes, providing 10 segments $[P1, P2]$[4]. An integration is performed for each of those segments using the previously described integration method. Integration across emitter area A_i is then performed by a trapezoidum integration for each altitude weighted by the results of the integration along the corresponding segment $[P1, P2]$. The height of the trapezoidum (denoted **h**) is the minimal distance on polygon A_j between the straight lines sustained by (colinear) segments $[P1, P2]$ taken at two successive altitudes. Thus the FF is mathematically estimated by :

$$F_{ij} \simeq \frac{h}{\pi Ai} \sum_{k=1}^{10} \sum_{m,n=1}^{3} w_3[m] w_3[n] \int_{\mathcal{S}} f_{s_m, t1_n, Pu_k}(X, Y) dX dY \tag{14}$$

with w_3 an array storing the weights used by the 3 points Gauss method, $\mathcal{S}$ a segment $[P_1, P_2]$ and the integration on segment $[P_1, P_2]$ performed by the method described in this paper. The function **Inc**[**P1**,**P2**] compute incrementally each segment $[P_1, P_2]$ on surface A_j (according to $(s[m], t[n])$ on A_i), integration being then performed for each of those segments.

We test our approach onto two 3D scenes that offer different geometrical properties (figure 9 (a) and (b)). The first scene is a part of a plant canopy used by agronomists. It hihglights a lot of complex geometric configurations, with several close polygons. The second scene is a more classical room. For both scenes we do not take into account occlusions and we compute the FF between each pair of polygons. In the following table 2 we put some results concerning the full estimate of those FFs using three different methods. The results we chose to report are significant of the general results, more specifically of the worst cases that can arise.



(a) (b)

Figure 9: (a) A part of a plant canopy. (b) A classical room.

The first method that appears in table 2 is a Gauss Quadrature using 10.000 points (10 points per dimension) for integrating each FF. Due to the large number of points it uses we assume the results the method provides are accurate enough to be used as the reference value. The second method is the analytical Schroeder's method and the third results are those obtained with our approach. Those results show that Schroeder's method appears to be generally more accurate. But in some cases it generates large errors (example 3 of table 2). Those cases seem to be connected with very low distance between polygons but we did not find any accurate explanation to this problem. Nevertheless, our approach even

[3] Of course another method could be used when patches are far enough from each other.

[4] Obviously the number of slices that are required for an accurate integration depends of several parameters : the size of the emitter, it's distance to the receiver... Furthermore according to those parameters the cutting out could be irregular, the distance between successive slices being smallest when those slices are close to the receiver.

	Gauss10000	Schroeder	Our approach
Ex 1	**0.0171674**	0.0172208	0.0165294
Ex 2	**0.0015258**	0.0015026	0.0014834
Ex 3	**0.0039079**	0.0050707	0.0037070
Ex 4	**0.0033659**	0.0033570	0.0031045
Ex 5	**0.0089509**	0.0089011	0.0083274
Ex 6	**0.0214502**	0.0209425	0.0207442
Ex 7	**0.0105754**	0.0103185	0.0100214

Table 2: Selected FF estimate from the canopy (1 to 4) and the room (5 to 7)

currently over-simply implemented has never be found to provide results far from 10% of its reference value. This behaviour is due to its ability of capturing essential features of the FF kernel even when it changes strongly. The advantage of our approach is thus to always control the induced error. Another advantage as compared to Schroeder's method is that it estimates any FF 8 to 10 times faster. This point is obviously important when recalling that FFs estimate take 90% of the computation time when simulating illumination.

5 Perspectives and conclusion

The main contribution of this paper is to provide a characterization of the FF kernel according to the relative altitudes of the surfaces that exchange energy. This better knowledge of the kernel features allows us to consider interesting new ways of research. Then we derived a simple implementation from our study and shown that despite its simplicity it provides accurate and stable results independently of both the size and the distance of the interacting polygons. Stability is ensured by the capture of the kernel features even when sharp variations appear. We could certainly improve the integration quality by searching for uncostly function that match better with the kernel function.

Occlusions have been omitted during this study. Obviously they must be taken into account during the FFs estimate. This problem is clearly difficult for any FFs estimate algorithm and it is generally processed wrongly[5]. In our approach, integration is performed along any $[P_1, P_2]$ segment. In case of partial occlusion along this segment, one can easily cut it into several subsegments, removing the subsegments that are fully occluded and applying our integration method onto each fully visible subsegment. A_j integration can then be correctly processed, providing accurate point(A_i)-surface(A_j) delta form factors. Like other algorithms, a weighted summation could be performed through A_i integration. But this will produce the same problems than those highlighted in footnote 5. We are currently investigating the idea of computing only delta FFs and then to regroup those that highlight closed variations. Then the receptor could be cut along the discontinuities that have been found, providing something like an adaptative discontinuity meshing [3]. Moreover this could be included into a hierarchical algorithm in which the classical **oracle** function [7] [4] which decides whether a patch should be divide or not, will be based on the kernel features.

Another interesting perspective to our work is that it should be easy to enclose any FF with our method by considering both the convexity and the concavity of the kernel. This would provide an error *a priori* on the FF and then, as highlighted by Lischinski [6], a minimum and maximum value for the resulting radiosity function. We would thus be able to have an idea of the error in the radiosity value, this property being important when a user has to judge of the results.

[5] Traditionnally the percentage of visibility is roughly estimated ; then the FF value that assumes full visibility is weighted by this percentage. However $\int\int \frac{\cos(\theta_x)\cos(\theta_y)}{r^2} V(x,y) \quad \# \quad \%V \int\int \frac{\cos(\theta_x)\cos(\theta_y)}{r^2}$

REFERENCES

[1] M. Cohen and J. Wallace. *Radiosity and realistic image synthesis.* Academic Press, 1993.

[2] K. S. H.D. Baehr. *Heat and mass transfer.* Berlin, Springer-Verlag, 1988.

[3] P. S. Heckbert. Discontinuity meshing for radiosity. *In Proceedings of the Third Eurographics Workshop on Rendering*, pages 203–216, May 1992.

[4] N. Holszschuch. *Le contrôle de l'erreur dans la méthode de radiosité hiérarchique.* PhD thesis, University J.Fourier, Grenoble, 1996.

[5] J. Howell. *A catalog of radiation configuration factors.* New York, McGraw-Hill, 1982.

[6] D. Lischinski. Bounds and error estimates for radiosity. *Proceedings of SIGGRAPH 94*, 99(7):1–100, January 1994.

[7] D. Lischinski, F. Tampieri, and D. Greenberg. Combining hierarchical radiosity and discontinuity meshing. *In Computer Graphics Proceedings*, Annual Conference Series:199–208, 1993.

[8] P. Schroeder and P. Hanrahan. On the form factor between two polygons. *Proceedings of SIGGRAPH 93*, pages 163–164, 1993.

[9] F. Sillion and C. Puech. *Radiosity and Global Illumination.* Morgan Kaufmann, 1994.

[10] W. V. W. Press, S. Teukolsky and B. Flannery. *Numerical recipes in C.* Cambridge University Press, 1992.