Table of Contents


```
clear all;
clc;
close all;
```

Annexe

```
Q = 10000; %Taux emission photon en photons/s
hs = 40; %Hauteur source en cm
he = 40; %Hauteur ecran en cm
le = 40; %Largeur ecran en cm
ld = 4; %Arete du detecteur
hd = 4; %Hauteur du detecteur
sigma = 0.5; %Section efficace ecran en cm^-1
time = 5*60;

r_e  = [0 -10 0;
        0 0 0;
        0 10 0];
```

Nombre de photons theorique detecte

```
Q_l = Q/hs;
Det_Area = ld*ld;
l1 = -20;
l2 = 20;

% Experience 1
x_d = 50;
x = [sqrt(x_d^2 + 10^2); x_d; sqrt(x_d^2 + 10^2)];
for i = 1:length(x)
    theta1(i) = atan(l1/x(i));
    theta2(i) = atan(l2/x(i));
    I1_th(i) = (Q_l/(4*pi*x(i)))*(theta2(i) - theta1(i));
    C1_th(i) = I1_th(i)*time*Det_Area;
end
C1_th = C1_th';

% Experience 2
x_d = 100;
x = [sqrt(x_d^2 + 10^2); x_d; sqrt(x_d^2 + 10^2)];
for i = 1:length(x)
    theta1(i) = atan(l1/x(i));
```

```
    theta2(i) = atan(l2/x(i));
    I2_th(i) = (Q_l/(4*pi*x(i)))*(theta2(i) - theta1(i));
    C2_th(i) = I2_th(i)*time*Det_Area;
end
C2_th = C2_th';

% Experience 3
x_d = 50;
x_e = 20;
e = 2;
x = [sqrt(x_d^2 + 10^2); x_d; sqrt(x_d^2 + 10^2)];
theta1 = atan(l1./x);
theta2 = atan(l2./x);
y = x.*sigma*(e/x_d);


for i = 1:3
    F = @(theta) exp(-y(i)*sec(theta));
    I3_th(i) = (Q_l/(4*pi*x(i)))*(quad(F,0,theta2(i)) -
 quad(F,0,theta1(i)));
    C3_th(i) = I3_th(i)*time*Det_Area;
end
C3_th = C3_th';

% Experience 4
x_d = 50;
x_e = 40;
e = 2;
x = [sqrt(x_d^2 + 10^2); x_d; sqrt(x_d^2 + 10^2)];
theta1 = atan(l1./x);
theta2 = atan(l2./x);
y = x.*sigma*(e/x_d);

for i = 1:3
    F = @(theta) exp(-y(i)*sec(theta));
    I4_th(i) = (Q_l/(4*pi*x(i)))*(quad(F,0,theta2(i)) -
 quad(F,0,theta1(i)));
    C4_th(i) = I4_th(i)*time*Det_Area;
end
C4_th = C4_th';

% Experience 5
x_d = 50;
x_e = 40;
e = 4;
x = [sqrt(x_d^2 + 10^2); x_d; sqrt(x_d^2 + 10^2)];
theta1 = atan(l1./x);
theta2 = atan(l2./x);
y = x.*sigma*(e/x_d);

for i = 1:3
    F = @(theta) exp(-y(i)*sec(theta));
    I5_th(i) = (Q_l/(4*pi*x(i)))*(quad(F,0,theta2(i)) -
 quad(F,0,theta1(i)));
```

```
    C5_th(i) = I5_th(i)*time*Det_Area;
end
C5_th = C5_th';

% Nombre de photons detectes
Nph_th = [C1_th C2_th C3_th C4_th C5_th]; % Pour chaque source
Nph_th_total = [sum(C1_th); sum(C2_th); sum(C3_th); sum(C4_th);
 sum(C5_th)]; % Nb photon total provenant des 3 sources pour chaque
 experience

fprintf ('Nombre de photons detecte theoriquement par le detecteur:
\n');
fprintf ('                             %d\n', round(Nph_th_total));
fprintf ('\n \n');
```

Nombre de photons detecte theoriquement par le detecteur:
4254
1124
1507
1507
534

Simulation numerique

```
N = 10;
NbPhotonsIni = Q*time;

for rko = 1:N
    % Experience 1
    x_d = 50;
    NbPhotons_Exp1 = 0;

    for i = 1:size(r_e,2)
        for j = 1:NbPhotonsIni
            photon = r_e(i,:) + [0, 0, (hs*(rand() - 0.5))]; %
 Generation du photon
            phi = 2*pi*rand();
            mu = (2*rand()) - 1;

            omega_S = [cos(phi)*sqrt(1-mu^2), sin(phi)*sqrt(1-mu^2),
 mu];
            photonDir = omega_S; % Direction du photon
            t = (x_d - photon(1))/photonDir(1);
            photonProp = photon + t*photonDir; % Propagation photon
            if ((t > 0) && (photonProp(2)<=(ld/2)) &&
 (photonProp(2)>=(-ld/2)) && (photonProp(3)<=(hd/2)) &&
 (photonProp(3)>=(-hd/2)))
                NbPhotons_Exp1 = NbPhotons_Exp1 + 1;
            end
        end
    end
```

```
    % Experience 2
    x_d = 100;
    NbPhotons_Exp2 = 0;

    for i = 1:size(r_e,2)
        for j = 1:NbPhotonsIni
            photon = r_e(i,:) + [0, 0, (hs*(rand() - 0.5))]; %
Generation du photon
            phi = 2*pi*rand();
            mu = (2*rand()) - 1;

            omega_S = [cos(phi)*sqrt(1-mu^2), sin(phi)*sqrt(1-mu^2),
mu];
            photonDir = omega_S; % Direction du photon
            t = (x_d - photon(1))/photonDir(1);
            photonProp = photon + t*photonDir; % Propagation photon
            if ((t > 0) && (photonProp(2)<=(ld/2)) &&
(photonProp(2)>=(-ld/2)) && (photonProp(3)<=(hd/2)) &&
(photonProp(3)>=(-hd/2)))
                NbPhotons_Exp2 = NbPhotons_Exp2 + 1;
            end
        end
    end

    % Experience 3
    x_d = 50;
    e = 2;
    x_e = 20;
    NbPhotons_Exp3 = 0;

    for i = 1:size(r_e,2)
        for j = 1:NbPhotonsIni
            photon = r_e(i,:) + [0, 0, (hs*(rand() - 0.5))]; %
Generation du photon
            phi = 2*pi*rand();
            mu = (2*rand()) - 1;
            omega_S = [cos(phi)*sqrt(1-mu^2), sin(phi)*sqrt(1-mu^2),
mu];
            photonDir = omega_S; % Direction du photon
            t = (x_d - photon(1))/photonDir(1);
            photonProp = photon + t*photonDir; % Propagation photon
            if ((t > 0) && (photonProp(2)<=(ld/2)) &&
(photonProp(2)>=(-ld/2)) && (photonProp(3)<=(hd/2)) &&
(photonProp(3)>=(-hd/2)))
                t_entre = (x_e - photon(1))/photonDir(1);
                photon_entreEcran = photon + t_entre*photonDir;
                t_ecran = (x_e + e - photon(1))/photonDir(1);
                photon_exitEcran = photon + t_ecran*photonDir;
                l_ecran = sqrt((photon_exitEcran(1)
- photon_entreEcran(1))^2 + (photon_exitEcran(2)
- photon_entreEcran(2))^2 + (photon_exitEcran(3) -
photon_entreEcran(3))^2);
```

```
                if (rand() < exp(-0.5*l_ecran))
                    NbPhotons_Exp3 = NbPhotons_Exp3 + 1;
                end
            end
        end
    end

    % Experience 4
    x_d = 50;
    e = 2;
    x_e = 40;
    NbPhotons_Exp4 = 0;

    for i = 1:size(r_e,2)
        for j = 1:NbPhotonsIni
            photon = r_e(i,:) + [0, 0, (hs*(rand() - 0.5))]; %
Generation du photon
            phi = 2*pi*rand();
            mu = (2*rand()) - 1;
            omega_S = [cos(phi)*sqrt(1-mu^2), sin(phi)*sqrt(1-mu^2),
mu];
            photonDir = omega_S; % Direction du photon
            t = (x_d - photon(1))/photonDir(1);
            photonProp = photon + t*photonDir; % Propagation photon
            if ((t > 0) && (photonProp(2)<=(ld/2)) &&
(photonProp(2)>=(-ld/2)) && (photonProp(3)<=(hd/2)) &&
(photonProp(3)>=(-hd/2)))
                t_entre = (x_e - photon(1))/photonDir(1);
                photon_entreEcran = photon + t_entre*photonDir;
                t_ecran = (x_e + e - photon(1))/photonDir(1);
                photon_exitEcran = photon + t_ecran*photonDir;
                l_ecran = sqrt((photon_exitEcran(1)
- photon_entreEcran(1))^2 + (photon_exitEcran(2)
- photon_entreEcran(2))^2 + (photon_exitEcran(3) -
photon_entreEcran(3))^2);

                if (rand() < exp(-0.5*l_ecran))
                    NbPhotons_Exp4 = NbPhotons_Exp4 + 1;
                end
            end
        end
    end

    % Experience 5
    x_d = 50;
    e = 4;
    x_e = 40;
    NbPhotons_Exp5 = 0;

    for i = 1:size(r_e,2)
        for j = 1:NbPhotonsIni
            photon = r_e(i,:) + [0, 0, (hs*(rand() - 0.5))]; %
Generation du photon
            phi = 2*pi*rand();
```

```
            mu = (2*rand()) - 1;
            omega_S = [cos(phi)*sqrt(1-mu^2), sin(phi)*sqrt(1-mu^2),
 mu];
            photonDir = omega_S; % Direction du photon
            t = (x_d - photon(1))/photonDir(1);
            photonProp = photon + t*photonDir; % Propagation photon
            if ((t > 0) && (photonProp(2)<=(ld/2)) &&
 (photonProp(2)>=(-ld/2)) && (photonProp(3)<=(hd/2)) &&
 (photonProp(3)>=(-hd/2)))
                t_entre = (x_e - photon(1))/photonDir(1);
                photon_entreEcran = photon + t_entre*photonDir;
                t_ecran = (x_e + e - photon(1))/photonDir(1);
                photon_exitEcran = photon + t_ecran*photonDir;
                l_ecran = sqrt((photon_exitEcran(1)
 - photon_entreEcran(1))^2 + (photon_exitEcran(2)
 - photon_entreEcran(2))^2 + (photon_exitEcran(3) -
 photon_entreEcran(3))^2);

                if (rand() < exp(-0.5*l_ecran))
                    NbPhotons_Exp5 = NbPhotons_Exp5 + 1;
                end
            end
        end
    end

    Nph_Exp1(rko) = NbPhotons_Exp1;
    Nph_Exp2(rko) = NbPhotons_Exp2;
    Nph_Exp3(rko) = NbPhotons_Exp3;
    Nph_Exp4(rko) = NbPhotons_Exp4;
    Nph_Exp5(rko) = NbPhotons_Exp5;
end

% Nombre de photons obtenus numeriquement pour chaque experience pour
 10 iterations
Nph_numeric = [Nph_Exp1; Nph_Exp2; Nph_Exp3; Nph_Exp4; Nph_Exp5];

% Nombre de photons moyenne obtenu numeriquement pour chaque
 experience
Nph = [sum(Nph_Exp1); sum(Nph_Exp2); sum(Nph_Exp3); sum(Nph_Exp4);
 sum(Nph_Exp5)]./N;

% Ecart-type provenant de la simulation numerique
for wwe = 1:length(Nph)
    Std(wwe) = (100/Nph(wwe))*sqrt((1/N)*sum((Nph_numeric(wwe,:) -
 Nph(wwe)).^2));
end

% Ecart-type theorique
for jjj = 1:length(Nph_th_total)
    Std_t(jjj) = 100/(sqrt(N*Nph_th_total(jjj)));
end
```

```
fprintf ('Nombre de photons moyen detecte numeriquement par le
 detecteur:\n');
fprintf ('                          %d\n', round(Nph));
fprintf ('\n \n');
fprintf ('Ecart-type provenant de la simulation numerique: \n');
fprintf ('         %.3f %% \n', Std);
fprintf ('\n \n');
fprintf ('Ecart-type theorique: \n');
fprintf ('         %.3f %% \n', Std_t);
fprintf ('\n \n');
```

```
Nombre de photons moyen detecte numeriquement par le detecteur:
                          4084
                          1100
                          1451
                          1466
                          515


Ecart-type provenant de la simulation numerique:
         2.005 %
         2.767 %
         2.297 %
         3.032 %
         4.332 %


Ecart-type theorique:
         0.485 %
         0.943 %
         0.815 %
         0.815 %
         1.368 %


Differences relatives:
                     -3.995 %
                     -2.076 %
                     -3.761 %
                     -2.766 %
                     -3.624 %
```

Differences relatives

```
for jj = 1:length(Nph)
    D(jj) = 100*((Nph(jj) - Nph_th_total(jj))/Nph_th_total(jj));
end

fprintf ('Differences relatives: \n');
fprintf ('                     %.3f %% \n', D);
fprintf ('\n \n');
```

Published with MATLAB® R2018b