DIMM ray tracer

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

Spreadsheet for geometric simulation of DIMM system beams.

Parameters :

```
fig = figure();
```

Editable

Angle of inclination of prism wall

```
alpha = deg2rad(1.93); %° -> System value = 1.93
```

Possible Tilt

```
i = deg2rad(0/3600); %" d'arc
```

Focal lenght (achromatic)

```
Focal =50; %mm -> System value = 50
```

Distance between prism and lens

```
e = 144; %mm -> System value = 144
```

Selected parameters (cannot be changed)

Mask

```
% Minimum ratio of 5
D_holes = 1;
gap = 5;
```

Prism

```
n_p = 1.5163; %BK7
t_c = 1.92;   %Prism thickness

D = 25;     %12 mm     hauteur du prisme
t_e = t_c - tan(alpha)*D/2;
```

Lens

```
D_Lens = 6.25; %mm  diamètre de la lentille
```

Equations :

Prism

```
%Prism input coordinates
Xin11 = 0;
Xin12 = 0;
Xin21 = 0;
Xin22 = 0;

Yin11 = (gap + D_holes)/2;
Yin12 = (gap - D_holes)/2;
Yin21 = -(gap + D_holes)/2;
Yin22 = -(gap - D_holes)/2;

%Calculation of internal prism angles + exit angle
beta = asin(n_p*sin(alpha-asin(sin(i)/n_p)))-alpha;
beta_deg = rad2deg(beta);
r = asin(sin(i)/n_p);


m11 = -1/tan(pi/2-r); %high beam slope
m12 = -1/tan(alpha); %high prism slope

m21 = -1/tan(pi/2-r);%low beam slope
m22 = 1/tan(alpha); %low prism slope

%Ray 1 high
b11 = Yin11 - Xin11*m11;
b1 = D/2 - t_e*m12;

Xout11 = (b1-b11)/(m11-m12);
Yout11 = Xout11*m12+b1;

%Ray 2 high
b12 = Yin12 - Xin12*m12;
```

```
Xout12 = (b1-b12)/(m11-m12);
Yout12 = Xout12*m12+b1;

%Ray 1 low
b21 = Yin21 - Xin21*m21;
b2 = -D/2 - t_e*m22;

Xout21 = (b2-b21)/(m21-m22);
Yout21 = Xout21*m22+b2;

%Ray 2 low
b22 = Yin22 - Xin22*m22;

Xout22 = (b2-b22)/(m21-m22);
Yout22 = Xout22*m22+b2;
```

Prism -> Lens

```
xL_11 = e;
xL_12 = e;
xL_21 = e;
xL_22 = e;

mL1 = -1/tan(pi/2-beta);
mL2 = 1/tan(pi/2-beta);

bL_11 = Yout11 + 1/tan(pi/2-beta)*Xout11;
bL_12 = Yout12 + 1/tan(pi/2-beta)*Xout12;
bL_21 = Yout21 + 1/tan(pi/2-beta)*Xout21;
bL_22 = Yout22 + 1/tan(pi/2-beta)*Xout22;

yL_11 = mL1*e+b11;
yL_12 = mL1*e+b12;
yL_21 = mL2*e+b21;
yL_22 = mL2*e+b22;
```

Lens -> Screen

```
Ecran = e+Focal;
yE1 = -Focal*tan(beta);
yE2 = Focal*tan(beta);
```

Display :

Prism modeling

```
rectangle('Position',[0 -D/2 t_e D])
hold on;
line([t_e,t_c], [-D/2, 0], 'Color', 'k');
line([t_e,t_c], [D/2, 0], 'Color', 'k');
```

Lens

```
line([e,e], [D_Lens/2, -D_Lens/2], 'Color', 'k');
```

Screen

```
line([e+Focal,e+Focal], [yE1*(1+1/10), yE2*(1+1/10)], 'Color', 'b');
```

Rays

Before Prism

```
%High beam arrival (telescope exit)
line([-100,0], [(gap/2 + D_holes/2)*(1+100*tan(i)), gap/2 + D_holes/2], 'Color', 'g');
line([-100, 0], [(gap/2 - D_holes/2)*(1+100*tan(i)), gap/2 - D_holes/2], 'Color', 'g');
%Low beam arrival (telescope exit)
line([-100,0], [(-gap/2 + D_holes/2)*(1-100*tan(i)), -gap/2 + D_holes/2], 'Color', 'r');
line([-100,0], [(-gap/2 - D_holes/2)*(1-100*tan(i)), -gap/2 - D_holes/2], 'Color', 'r');
```

Prism

```
%Internal prism beam
line([Xin11,Xout11], [Yin11, Yout11], 'Color', 'g');
line([Xin12,Xout12], [Yin12, Yout12], 'Color', 'g');

line([Xin21,Xout21], [Yin21, Yout21], 'Color', 'r');
line([Xin22,Xout22], [Yin22, Yout22], 'Color', 'r');
```

Prism-Lens

```
%Prism exit beam
line([Xout11,e], [Yout11, yL_11], 'Color', 'g');
line([Xout12,e], [Yout12, yL_12], 'Color', 'g');
line([Xout21,e], [Yout21, yL_21], 'Color', 'r');
line([Xout22,e], [Yout22, yL_22], 'Color', 'r');
```

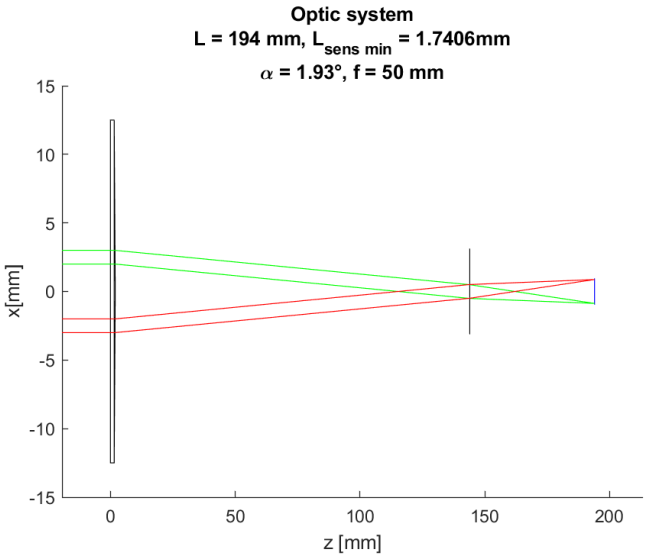
Lens-Screen

```
%Lens exit beam
line([xL_11,Ecran], [yL_11, yE1], 'Color', 'g');
line([xL_12,Ecran], [yL_12, yE1], 'Color', 'g');
line([xL_21,Ecran], [yL_21, yE2], 'Color', 'r');
line([xL_22,Ecran], [yL_22, yE2], 'Color', 'r');
```

Title and image scaling

```
xlim([-Ecran/10 Ecran*(1+1/10)])

title({'Optic system';['L = ',num2str(Ecran), ' mm, L_{sens min} = ',num2str(yE2-yE1),'mm'];['\alpha = ',num2str(rad2deg(alpha)),'°,
xlabel('z [mm]')
ylabel('x[mm]')
```



Calculating camera parameters

Parameters :

```
lambda = 600e-9;    %wavelength [m]
D = 305e-3;         %Telescope diameter [m]

D_holes = D_holes*1e-3;
D_sun = 20;      %Sunspots diameter [arc"]
F = Focal*1e-3;
H = (yE2-yE1)*1e-3; %gap between the 2 rays

r0 = [0.02,0.05,0.1,0.2]; %[m]
r0_600nm = r0*power(lambda/500e-9,6/5); %[m]
r0_550nm = r0*power(550/500,6/5); %[m]
```

Equations :

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%1.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Dmt = (D-20e-3)/6;      %Mask diameter transferred to telescope

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%2.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Gt = D_holes/Dmt;       %Image magnification

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%3.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%View from the telescope
sigma_t = sqrt(0.1698*power(lambda/Dmt,2)*power(Dmt./r0_600nm,5/3));
%View from the system
sigma_t_sec = sigma_t * (3600*180)/pi;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%4.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Fov_tel = 6*sigma_t_sec + D_sun;        %FOV calculation
Fov_mask = Fov_tel * 1/Gt;
Fov_mask_rad = Fov_mask / 3600 / 180*pi;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%5.
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Fov_sensor = F*Fov_mask_rad;    %Size on sensor

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%6.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
pixel_max = lambda*F/2/D_holes; %Maximum pixel size to comply with Niquist

No_Pixels = Fov_sensor/pixel_max;       %Number of pixels for 1 image

Screen_Size = (H/pixel_max + No_Pixels*2); %Minimum screen size [ft]
```

Camera parameters :

```
fprintf("---------------\nSpecs caméra :");
```

```
---------------
Specs caméra :
```

```
fprintf("---------------\n\nSensor size limit :\n %3.1f pi",ceil(max(Screen_Size)));
```

```
---------------

Sensor size limit :
 164.0 pi
```

```
fprintf("Minimum number of pixels for an image = %4.0f pi",min(No_Pixels));
```

```
Minimum number of pixels for an image =   17 pi
```

```
fprintf("Maximum pixel size :\n %2.2f [um]",pixel_max*1e6);
```

```
Maximum pixel size :
 15.00 [um]
```

Results for selected cameras

```
Pixel_size = 9e-6;

Useful_pixels = ceil(min(Fov_sensor/Pixel_size));
Screen_Size = ceil((H/Pixel_size + Useful_pixels*2));

fprintf("\n\n\n---------------\nCamera used :");
```

```
---------------
Camera used :
```

```
fprintf("---------------\n\nPixel size : %.0f X %.0f [um]",Pixel_size * 1e6,Pixel_size * 1e6);
```

```
---------------

Pixel size : 9 X 9 [um]
```

```
fprintf("Number of pixels used for an image = %4.0f pi",Useful_pixels);
```

```
Number of pixels used for an image =   28 pi
```

```
fprintf("Number of pixels used for entire image : %.0f pi",Screen_Size);
```

```
Number of pixels used for entire image : 250 pi
```