Diffraction Through a Rectangular Aperture – Simplest Theory

In this example, we show plots of the sound intensity *vs*. angles and observer/listener position x_{screen}, y_{screen} on a screen for the simplest theory of Fraunhofer diffraction through a rectangular aperture of vertical height *a* and horizontal width *b*. The observer/listener is located far from the rectangular aperture, at a perpendicular distance $L\ (m)$ away, such that the conditions $b, a \ll L$ **.and.** $\lambda \ll L$ both hold simultaneously, where $\lambda\ (m)$ is the wavelength of the sound – this is the so-called "far-field" limit.

The expression for diffraction through a rectangular aperture in this simplest theory is given by:

$$I_{tot}(\theta) = I_o \left\{ \frac{\sin\delta_y}{\delta_y} \right\}^2 \cdot \left\{ \frac{\sin\delta_x}{\delta_x} \right\}^2 \equiv I_o \operatorname{sinc}^2\delta_y \cdot \operatorname{sinc}^2\delta_x \text{ and } SIL(\theta) \equiv 10\log_{10}\left(I_{tot}(\theta)/I_{ref}\right)\ (dB)$$

where $I_o\ \left(Watts/m^2\right)$ is the maximum sound intensity associated with the single slit, the phases $\delta_y = (\pi a/\lambda)\sin\theta_y$ and $\delta_x = (\pi b/\lambda)\sin\theta_x\ (radians)$ where $\theta_y = \tan^{-1}(y_{screen}/L)$ and $\theta_x = \tan^{-1}(x_{screen}/L)$. Then: $r_{screen} = \sqrt{x_{screen}^2 + y_{screen}^2}$ and $\theta = \tan^{-1}(r_{screen}/L)$.
$I_{ref} \equiv 10^{-12}\ \left(Watts/m^2\right)$ is the reference sound intensity for the sound intensity level (SIL).

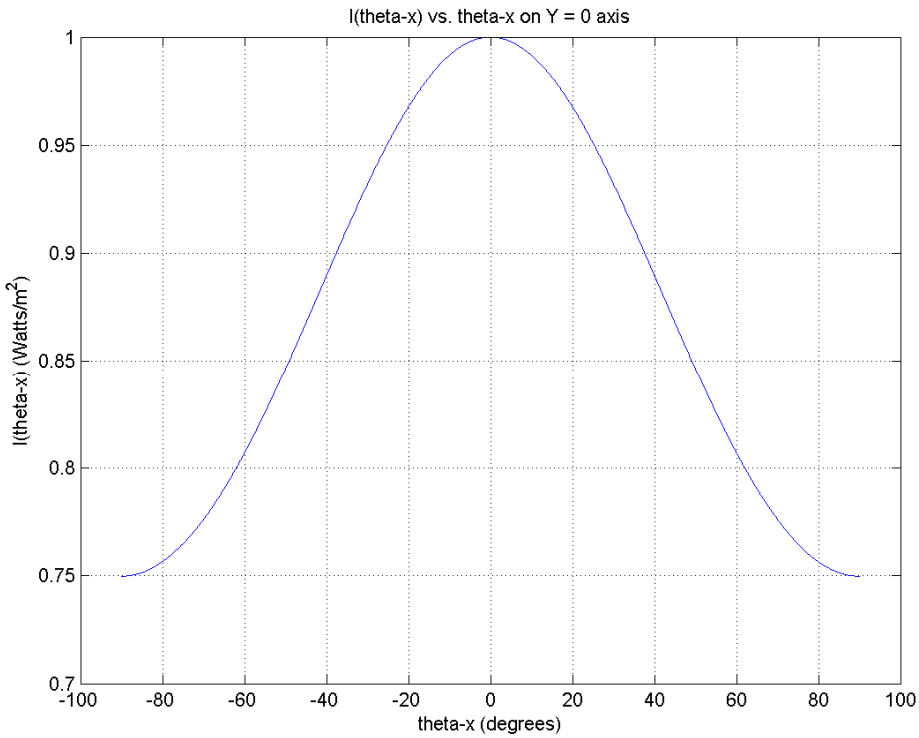
Diffraction <u>minima</u> (intensity zeroes) occur when $\delta_y = (\pi a/\lambda)\sin\theta_y$ = ±π, ±2π, ±3π, … = ±*m*π, *m* = 1, 2, 3, …. and $\delta_x = (\pi b/\lambda)\sin\theta_x$ = ±π, ±2π, ±3π, … = ±*n*π, *n* = 1, 2, 3, ….

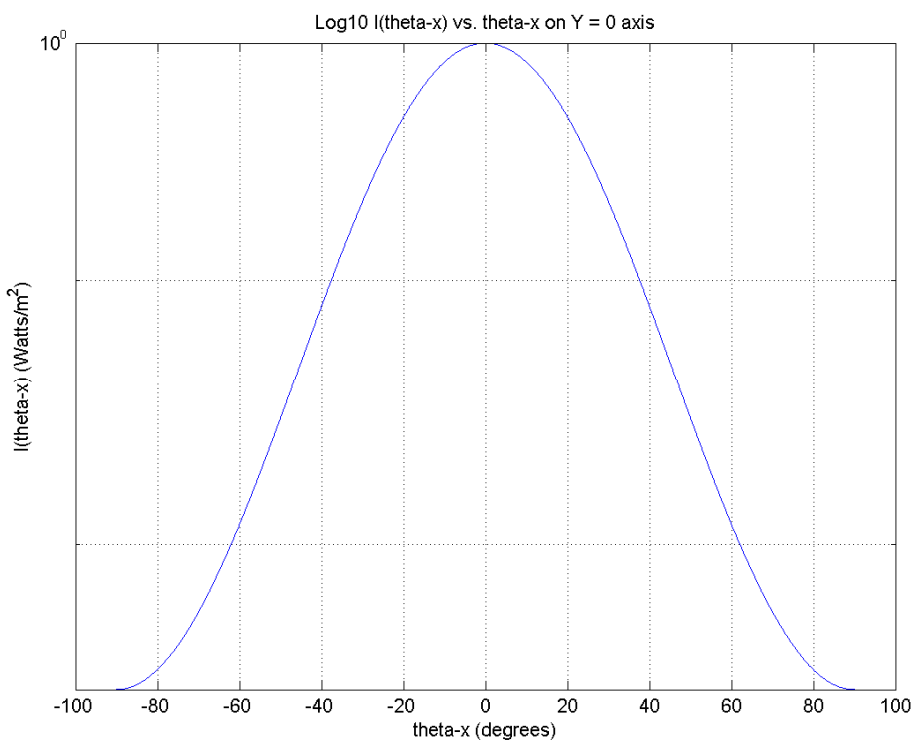
We coded up the above formulas using MATLAB to make plots of I_{tot} *vs.* θ_x, θ_y and I_{tot} *vs.* x_{screen}, y_{screen} *e.g.* for (*b* , *a*) = (0.1 *m*, 0.2 *m*) , (1.0 *m*, 0.5 *m*) and (1.0 *m*, 1.0 *m*) with the following parameter values: $I_o = 1\ Watt/m^2$, observer/listener distance (at $\theta_y = \theta_x = \theta = 0$) of $L = 10\ m$, the speed of propagation in free air/great-wide open: $v_{air} = 343\ m/s$ and frequency of $f = 1000\ Hz$, thus $\lambda = v_{air}/f = 0.345\ m$.

In the following figures, note that the angular width of the central maxima in the *y*- (*x*-) direction decreases as the aperture width *b* (aperture height *a*) increases, since the phases $\delta_y = (\pi a/\lambda)\sin\theta_y$ and ($\delta_x = (\pi b/\lambda)\sin\theta_x$) are linearly proportional to *a*. (*b*) respectively. Note also that the angular width of the central maxima in the *y*- (*x*-) direction increases linearly with increasing frequency *f*, since the phase differences both increase linearly with frequency:

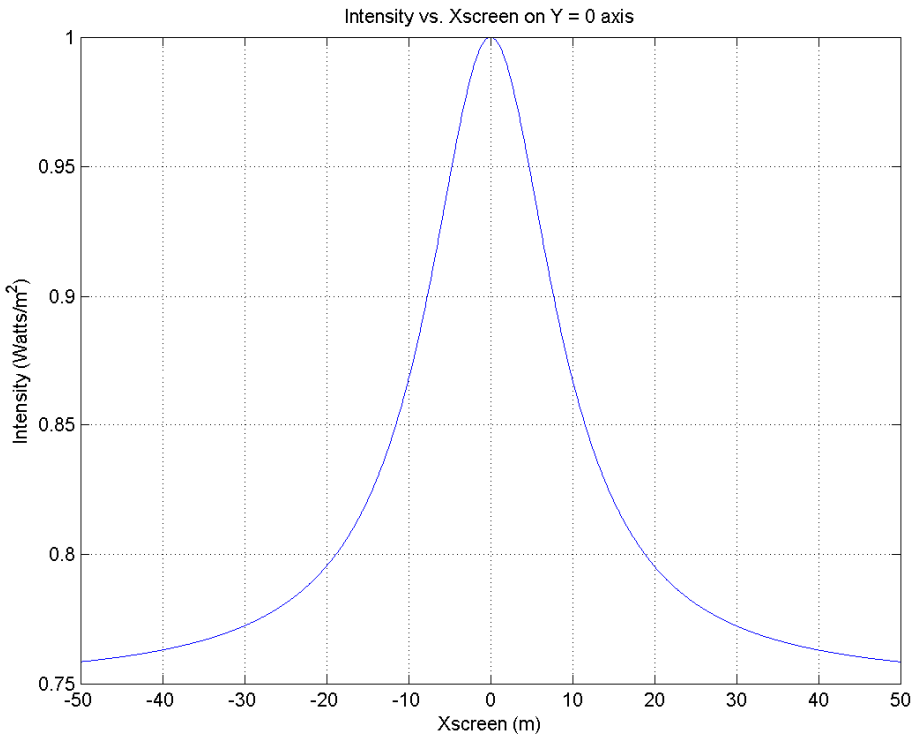
$$\delta_y = (\pi a/\lambda)\sin\theta_y = (\pi a f/v_{air})\sin\theta_y \quad \text{and} \quad \delta_x = (\pi b/\lambda)\sin\theta_x = (\pi b f/v_{air})\sin\theta_x$$

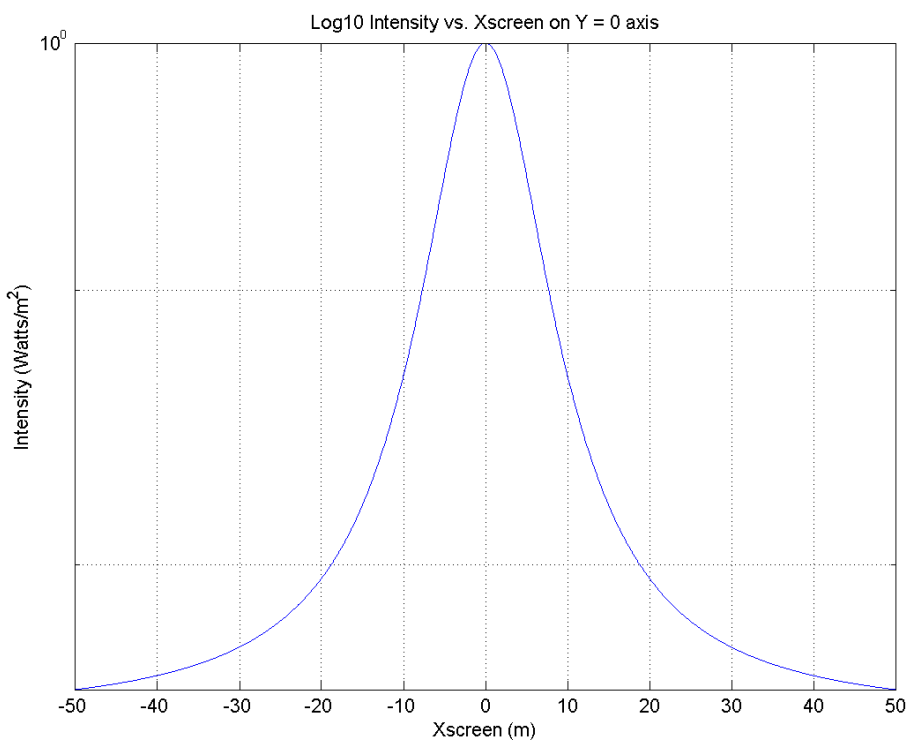
Linear and Semilog Plots of I_{tot} *vs.* θ_x for (b , a) = (0.1 m, 0.2 m):



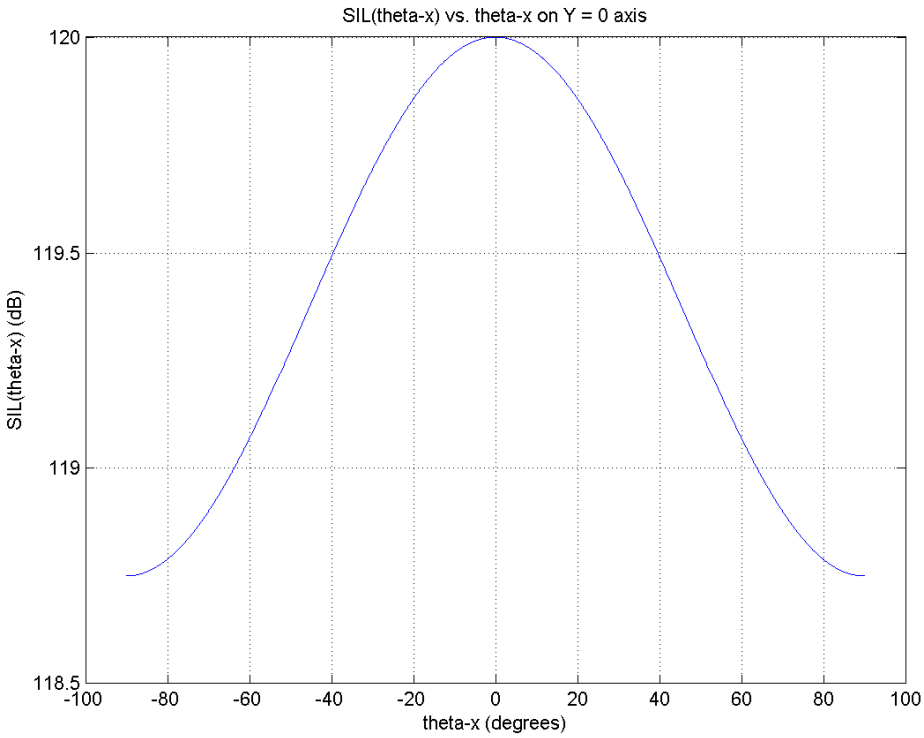


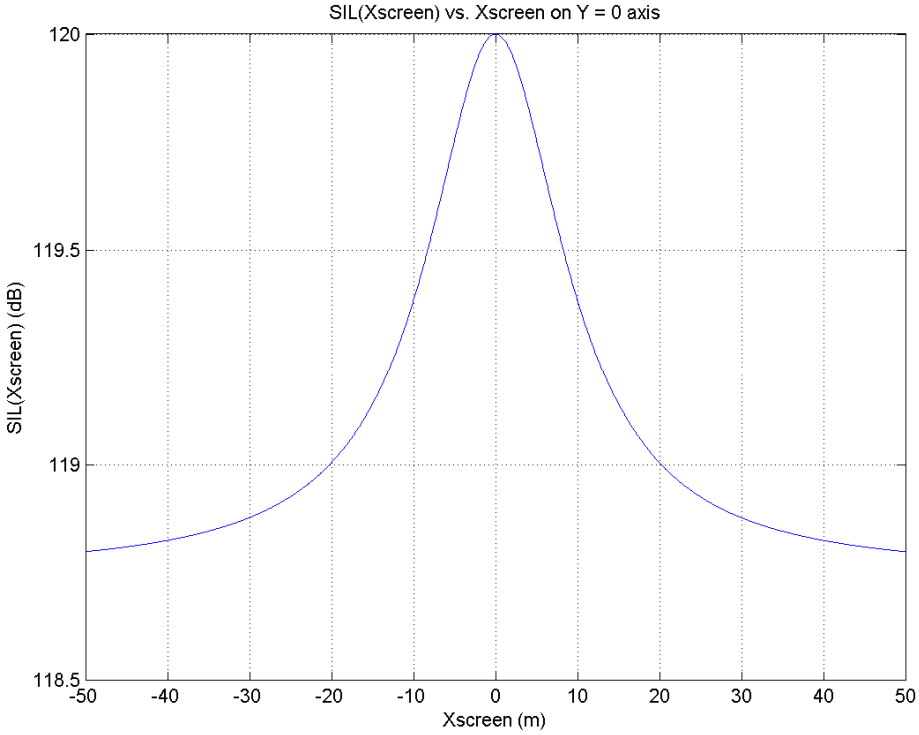
Linear and Semilog Plots of I_{tot} *vs.* x_{screen} for (b , a) = (0.1 m, 0.2 m):



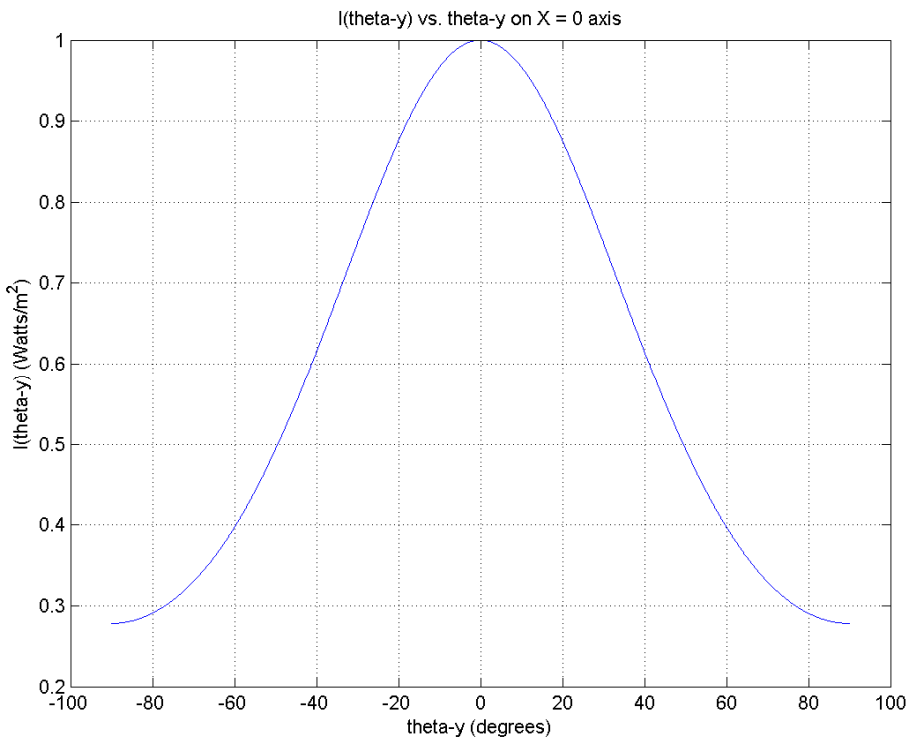


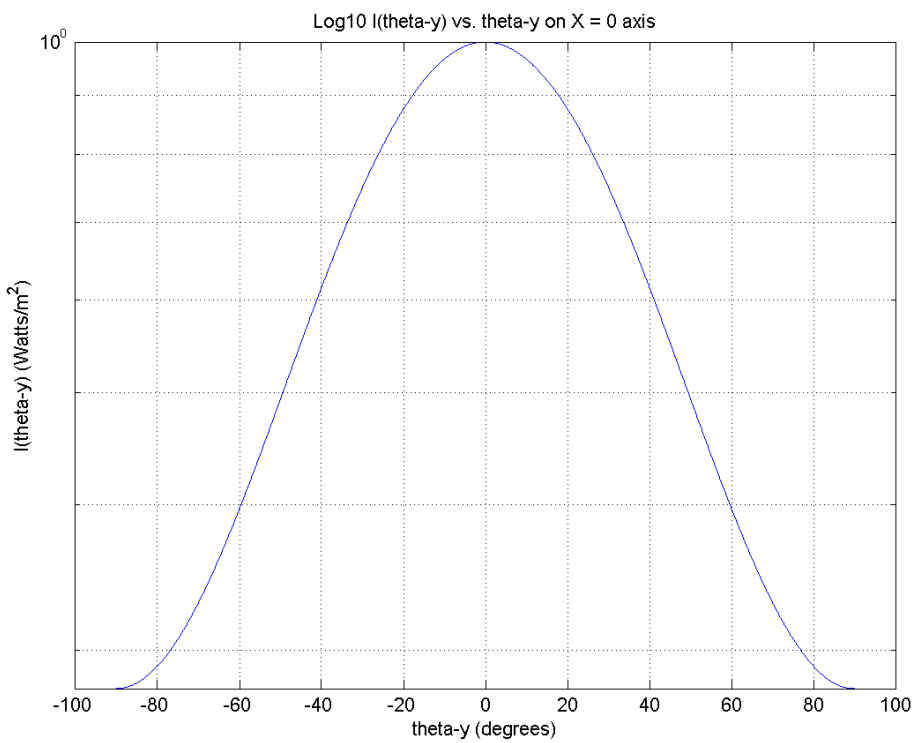
Plots of SIL *vs.* θ_x and SIL *vs.* x_{screen} for (b , a) = (0.1 m, 0.2 m):



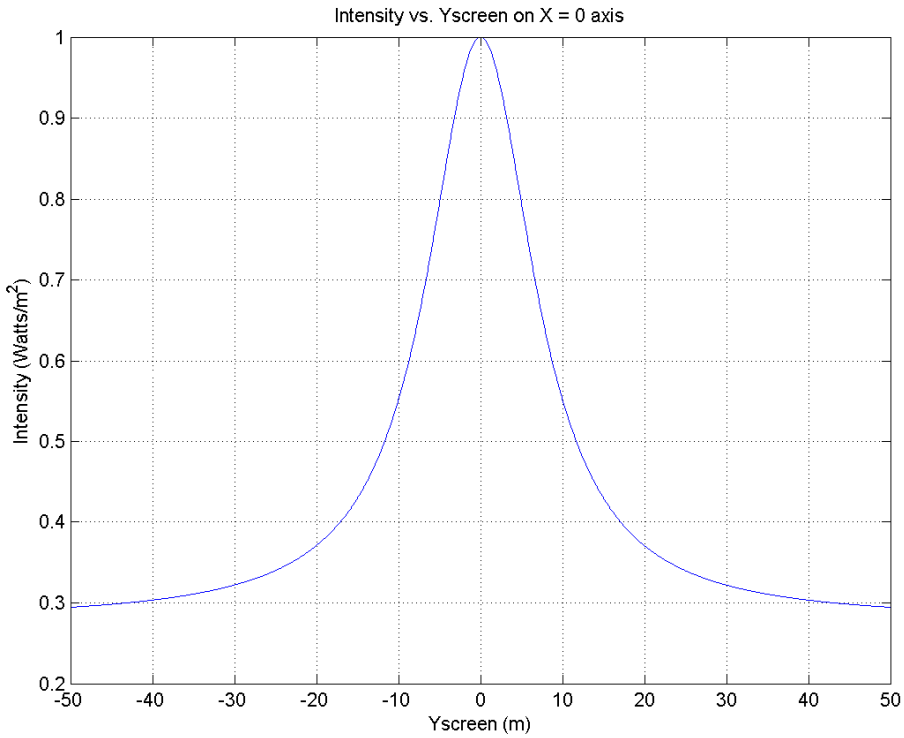


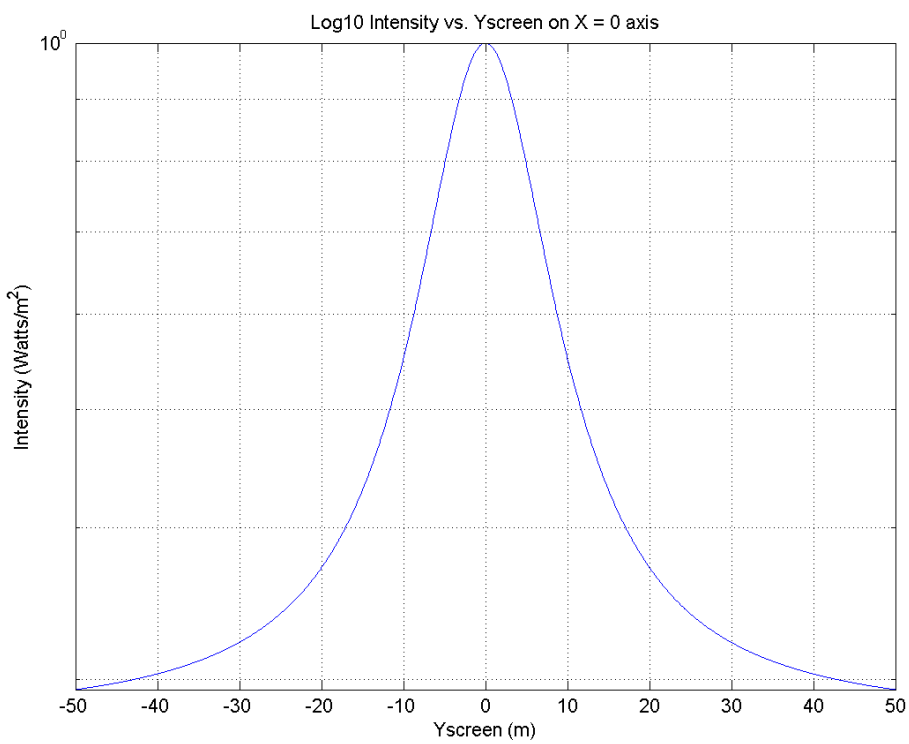
Linear and Semilog Plots of I_{tot} *vs.* θ_y for (b , a) = (0.1 m, 0.2 m):



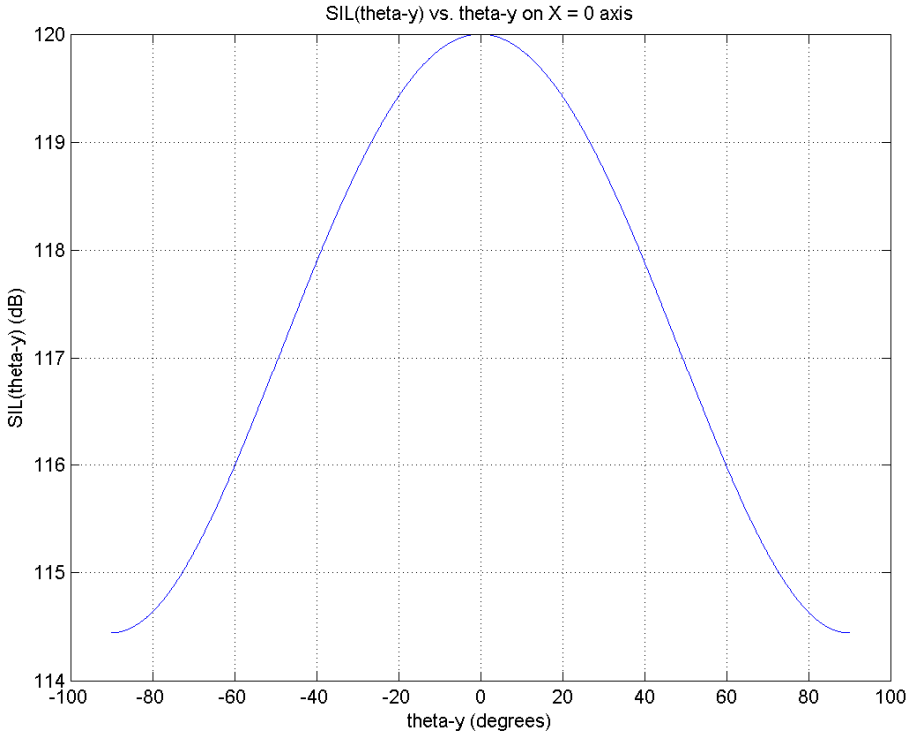


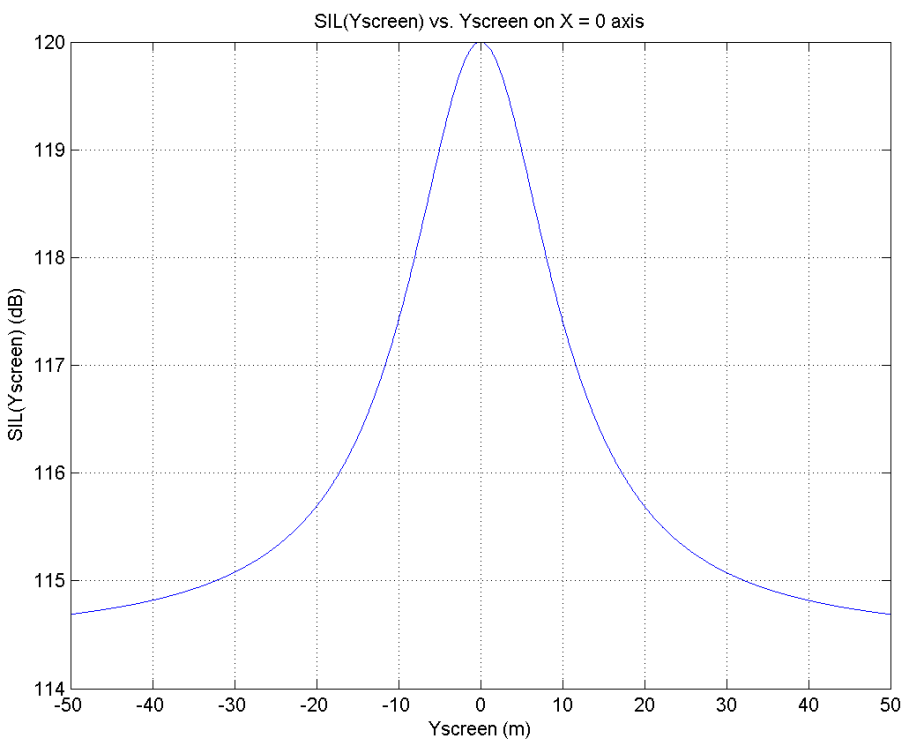
Linear and Semilog Plots of I_{tot} *vs.* y_{screen} for (b , a) = (0.1 m, 0.2 m):



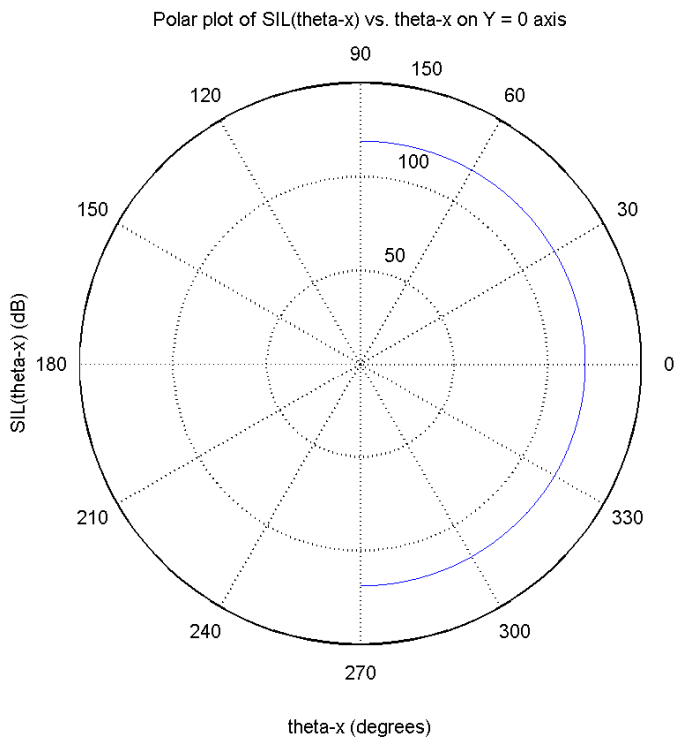


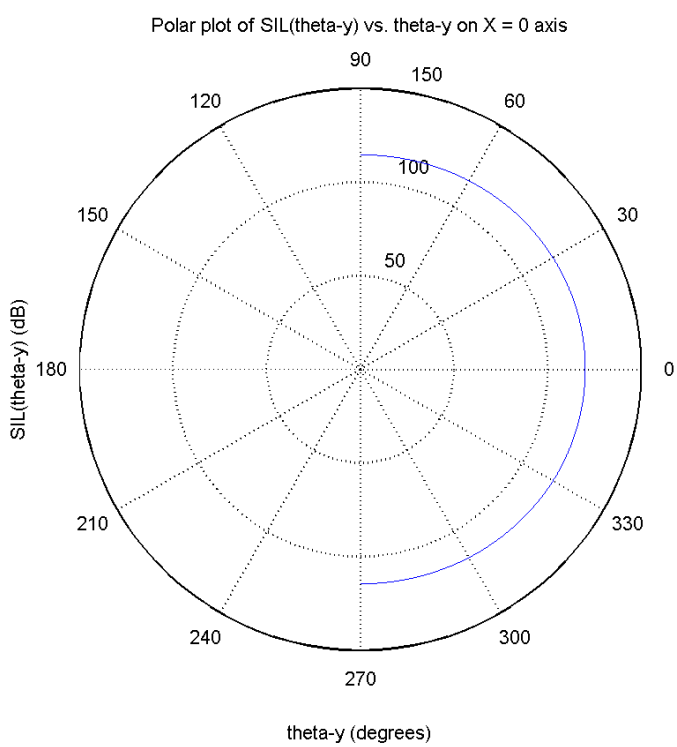
Plots of SIL *vs.* θ_y and SIL *vs.* y_{screen} for (b , a) = (0.1 m, 0.2 m):



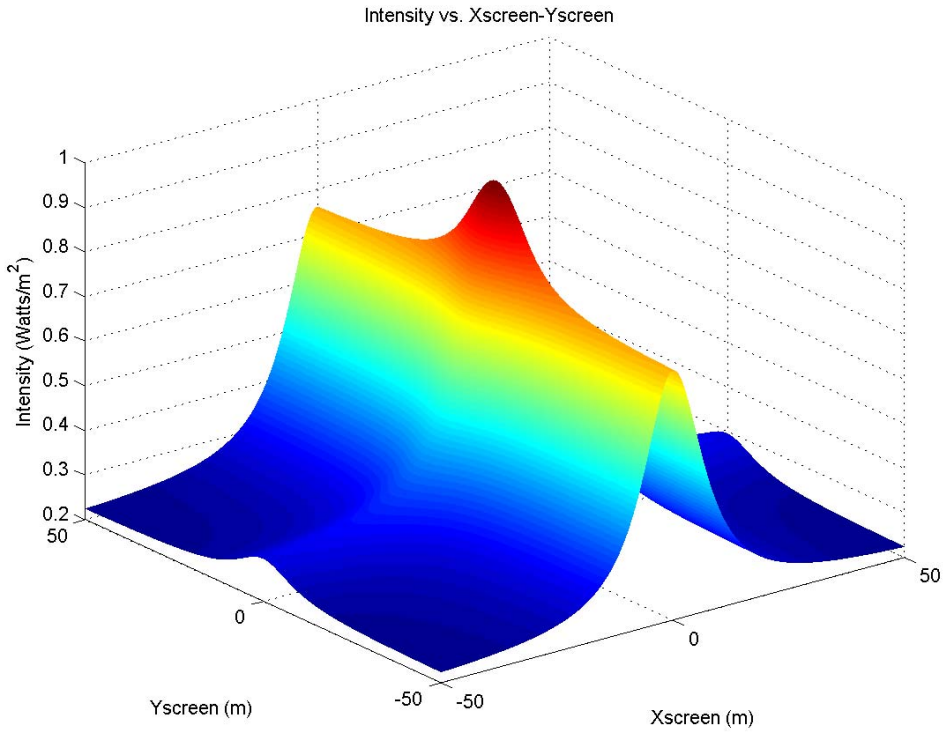


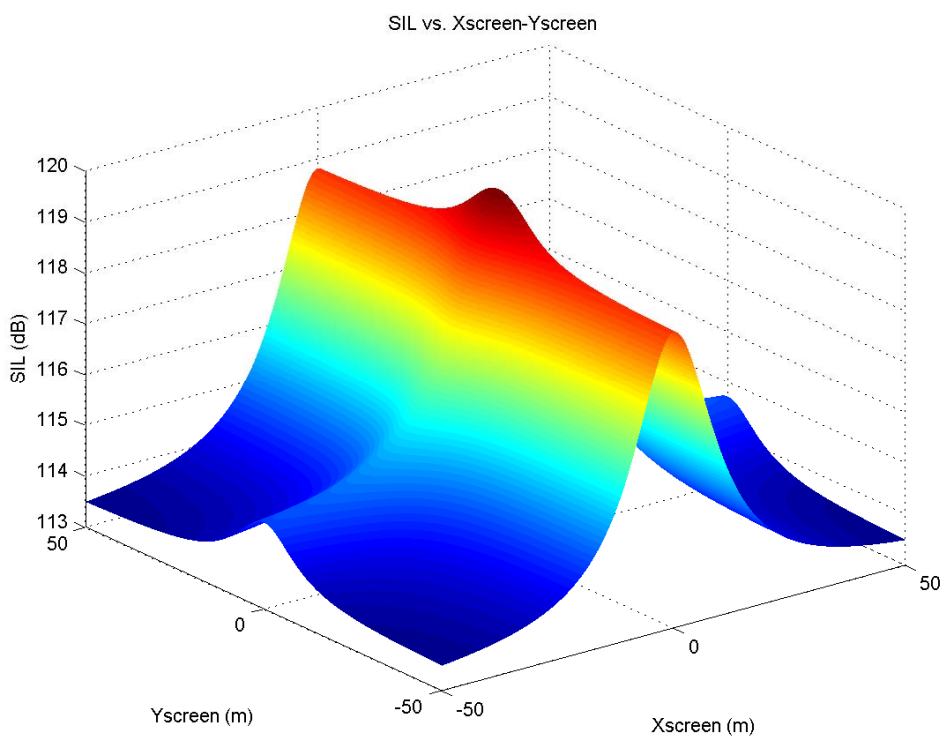
Plots of SIL *vs.* θ_x and SIL *vs.* θ_y for $(b\ ,\ a) = (0.1\ m,\ 0.2\ m)$:



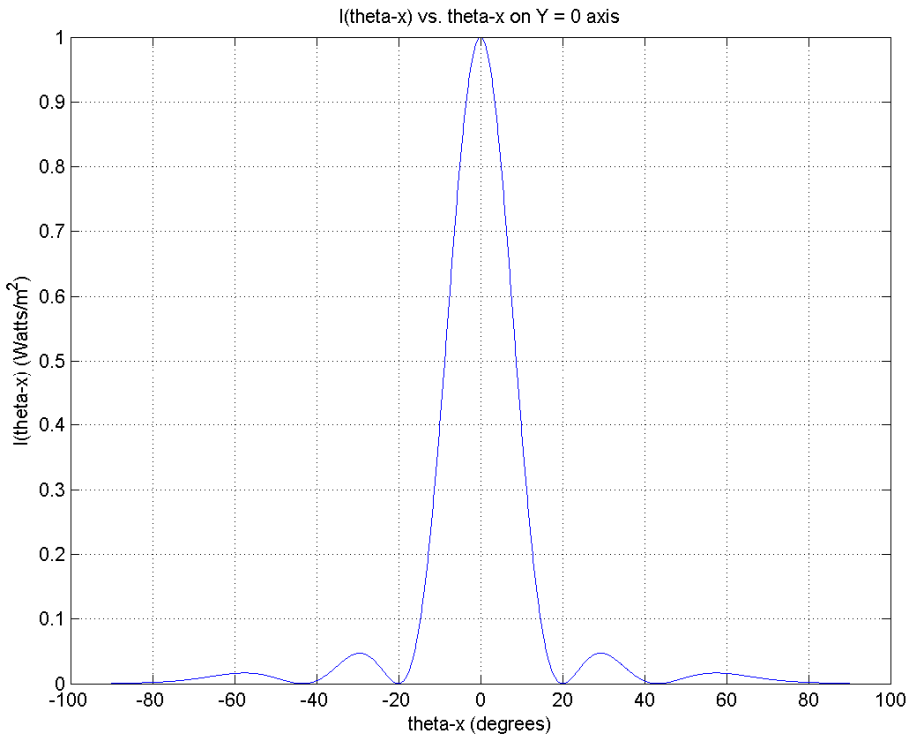


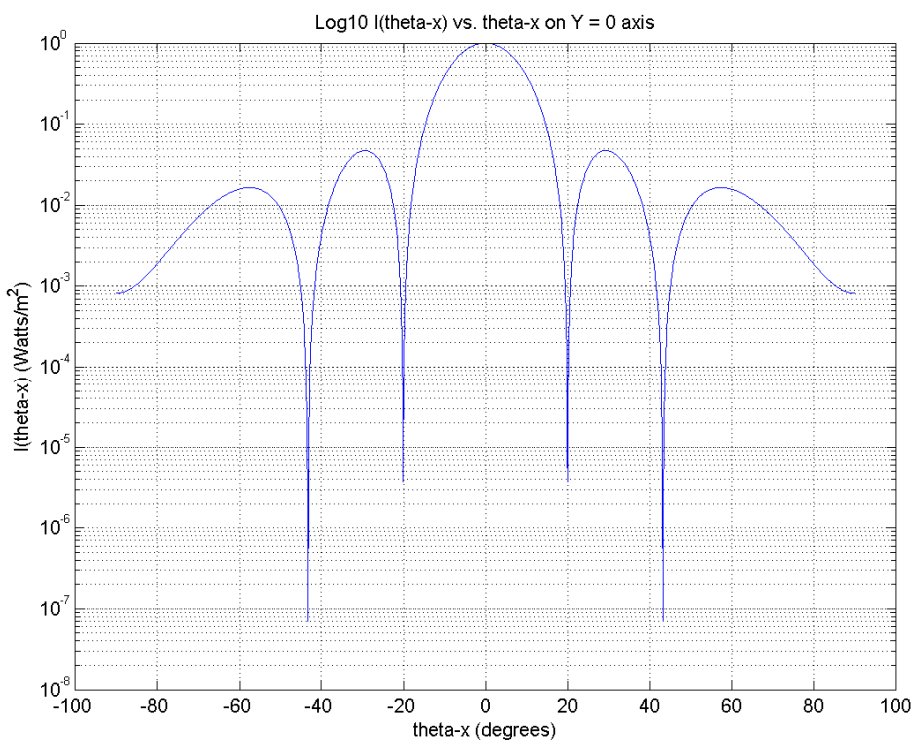
Plots of I_{tot} *vs.* $\left(x_{screen}, y_{screen}\right)$ and *SIL vs.* $\left(x_{screen}, y_{screen}\right)$ for (b , a) = (0.1 m, 0.2 m):



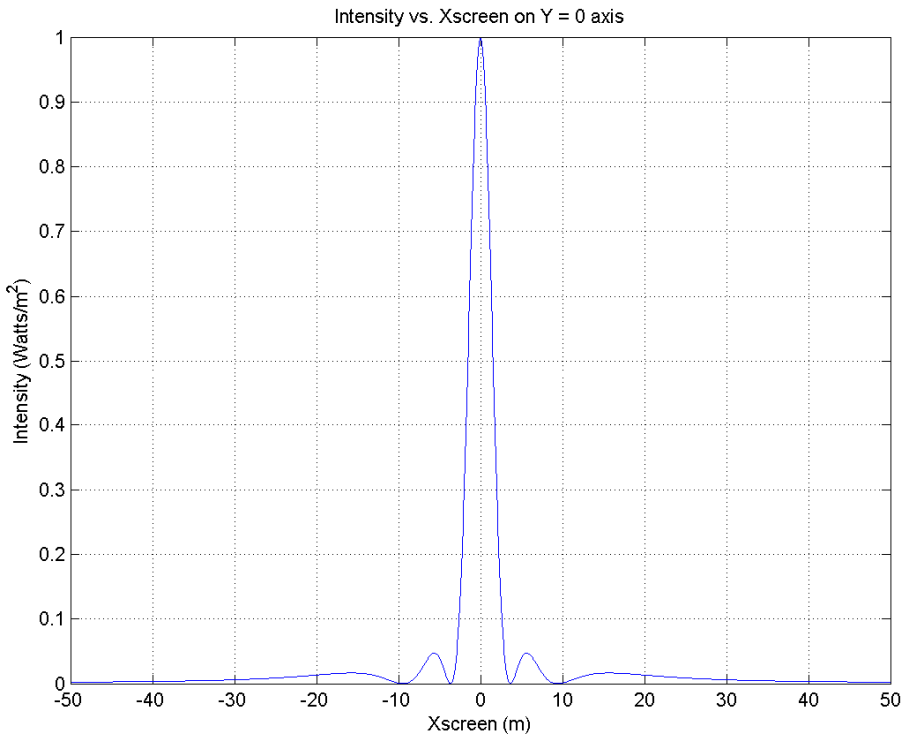


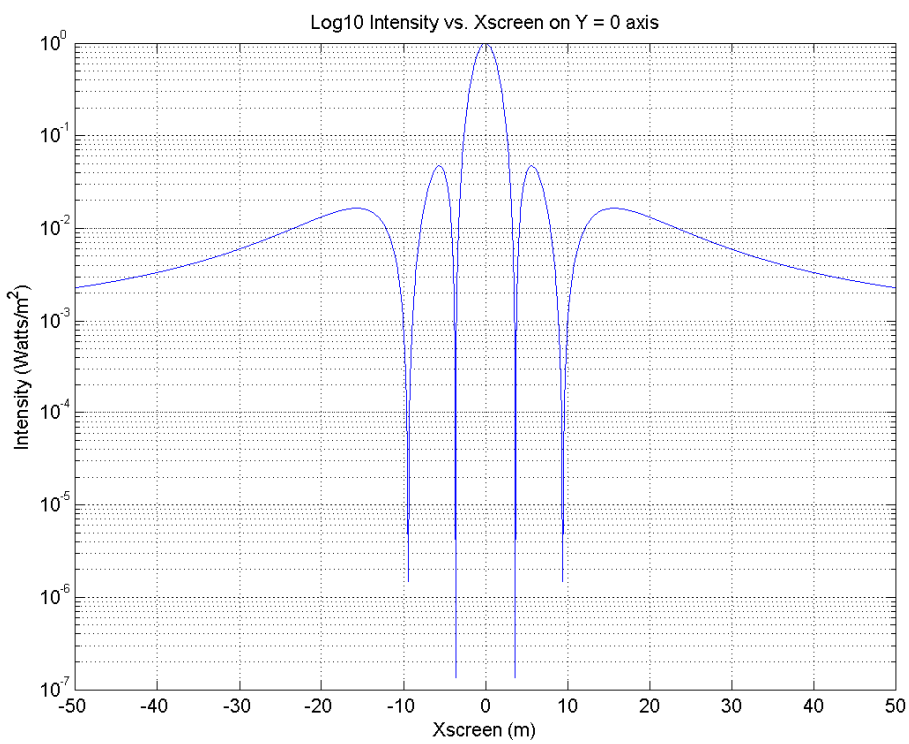
Linear and Semilog Plots of I_{tot} *vs.* θ_x for (b , a) = (1.0 m, 0.5 m):



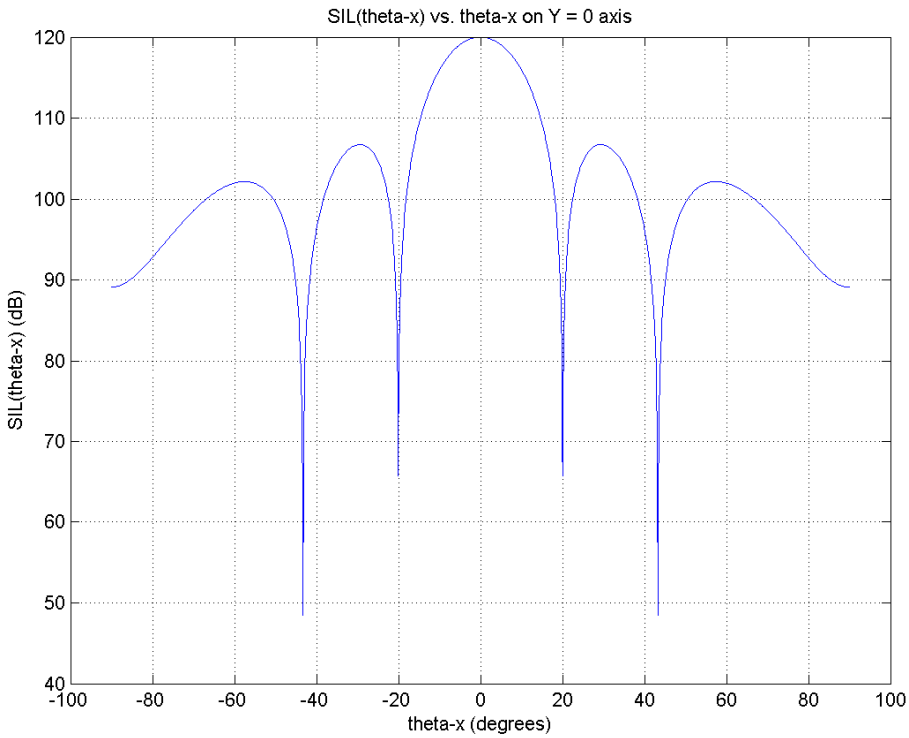


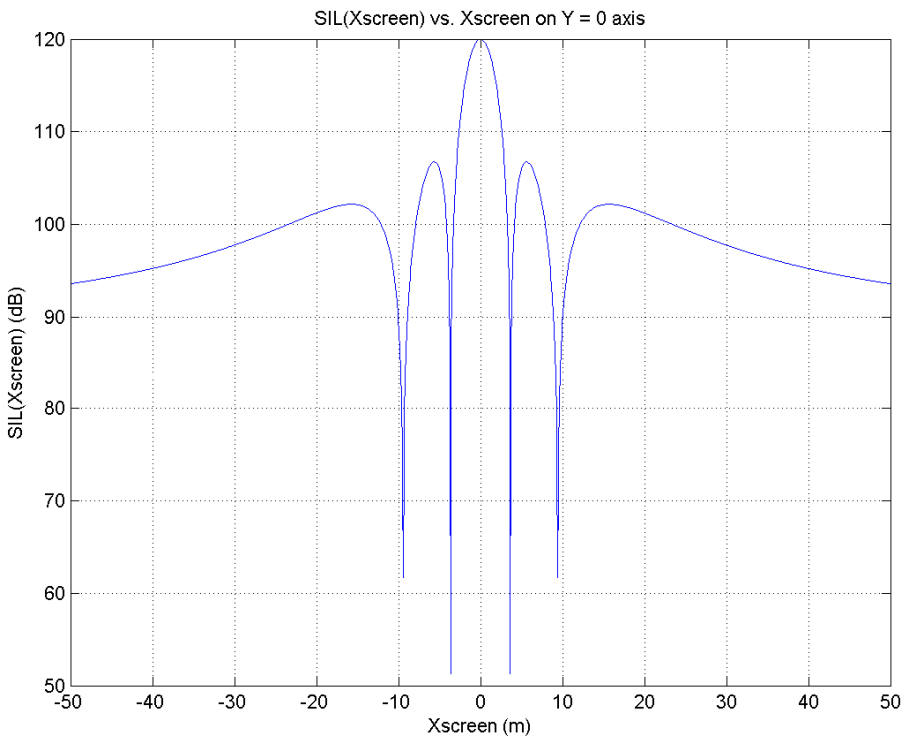
Linear and Semilog Plots of I_{tot} *vs.* x_{screen} for (b , a) = (1.0 m, 0.5 m):



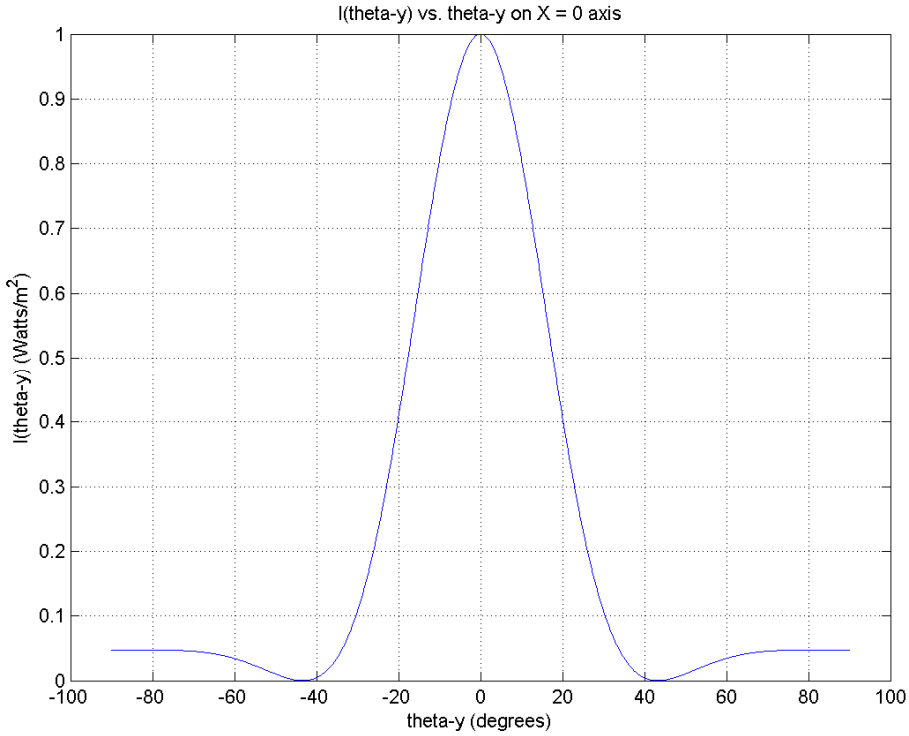


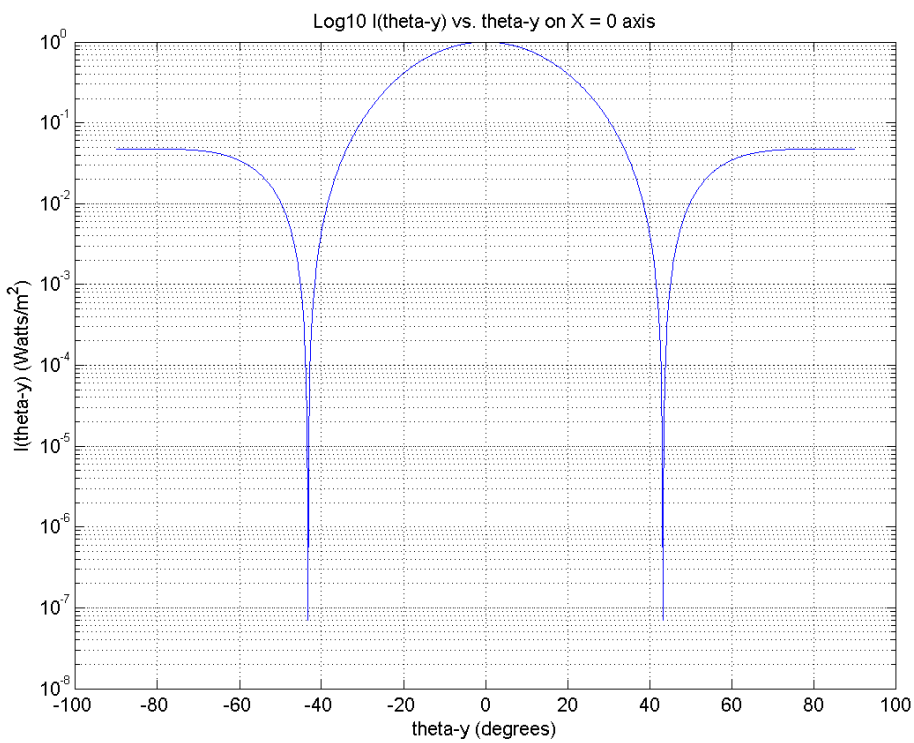
Plots of SIL *vs.* θ_x and SIL *vs.* x_{screen} for (b , a) = (1.0 m, 0.5 m):



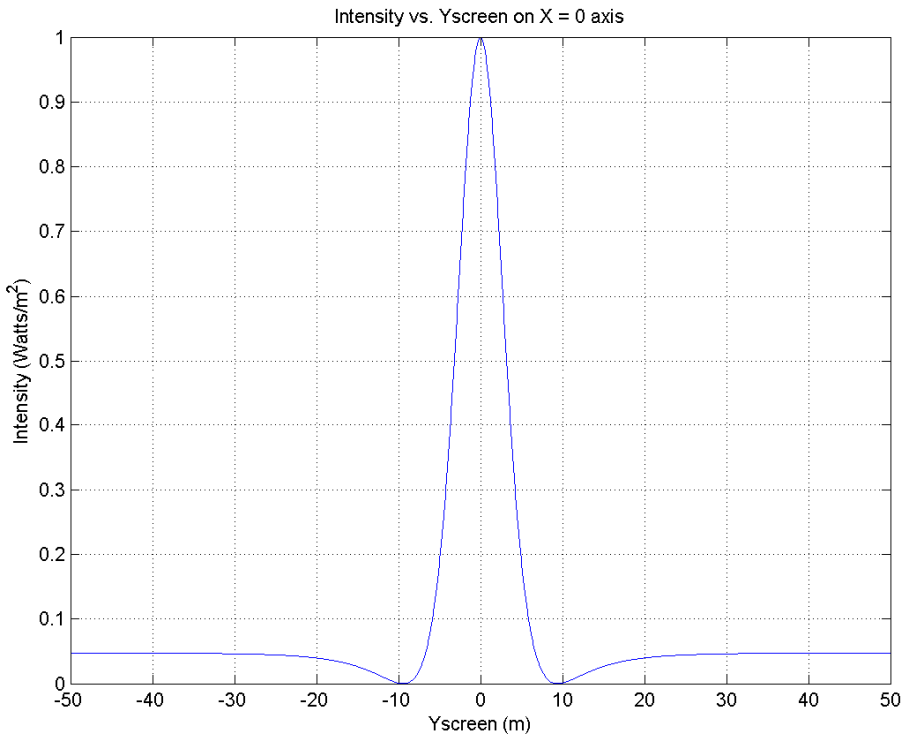


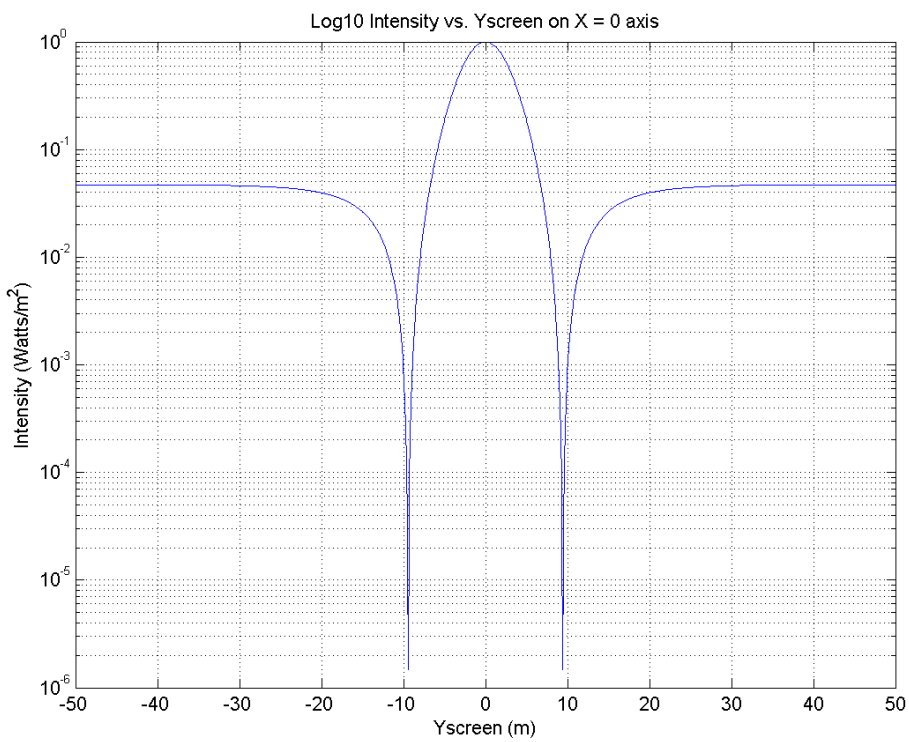
Linear and Semilog Plots of I_{tot} *vs.* θ_y for (b , a) = (1.0 m, 0.5 m):



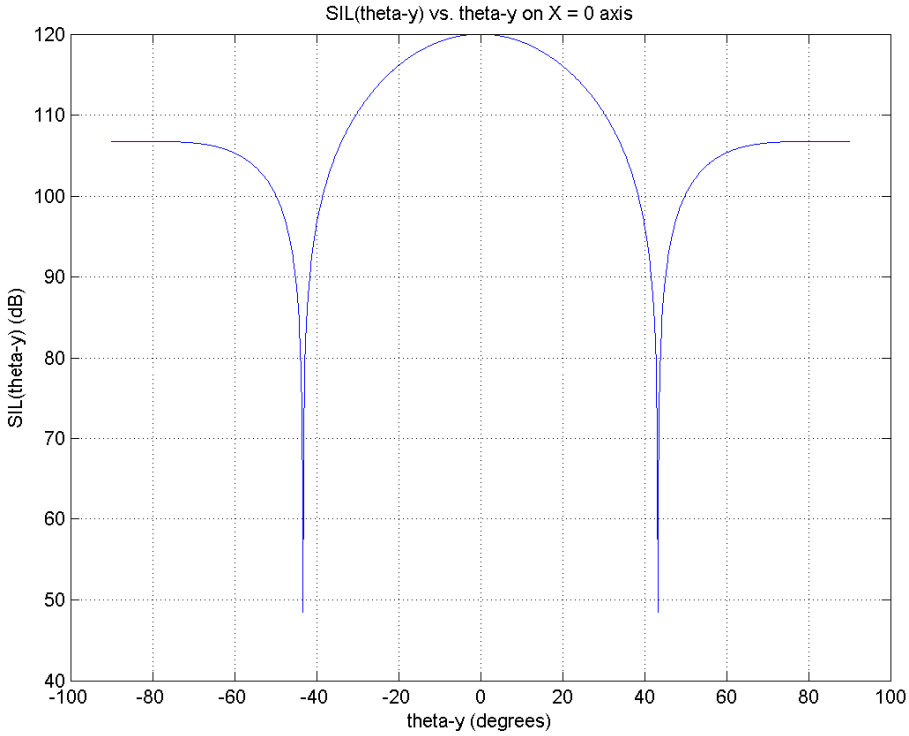


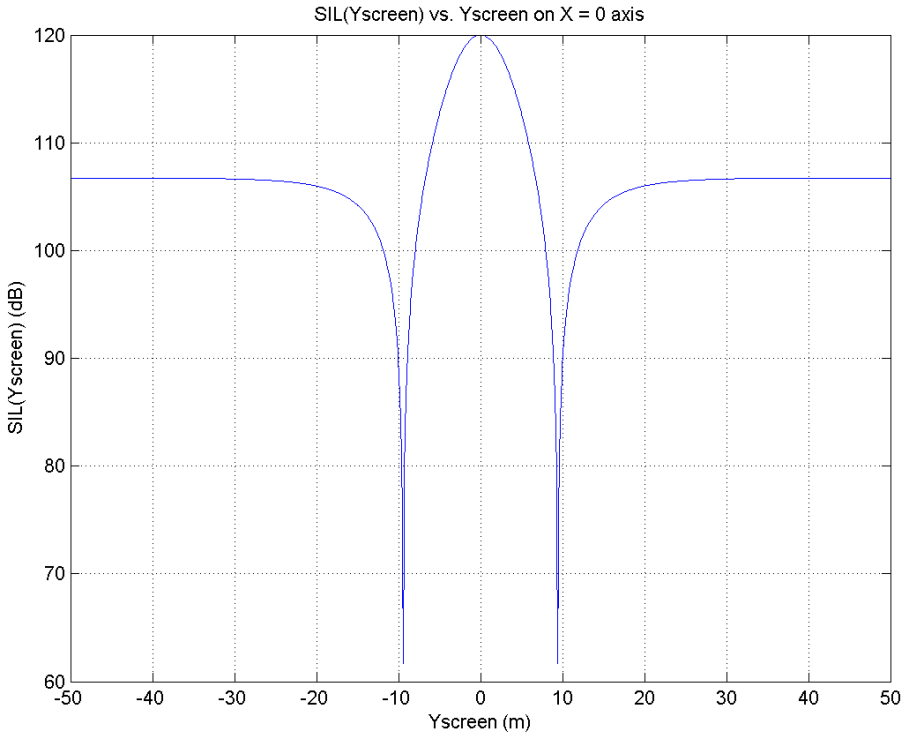
Linear and Semilog Plots of I_{tot} *vs.* y_{screen} for (b , a) = (1.0 m, 0.5 m):



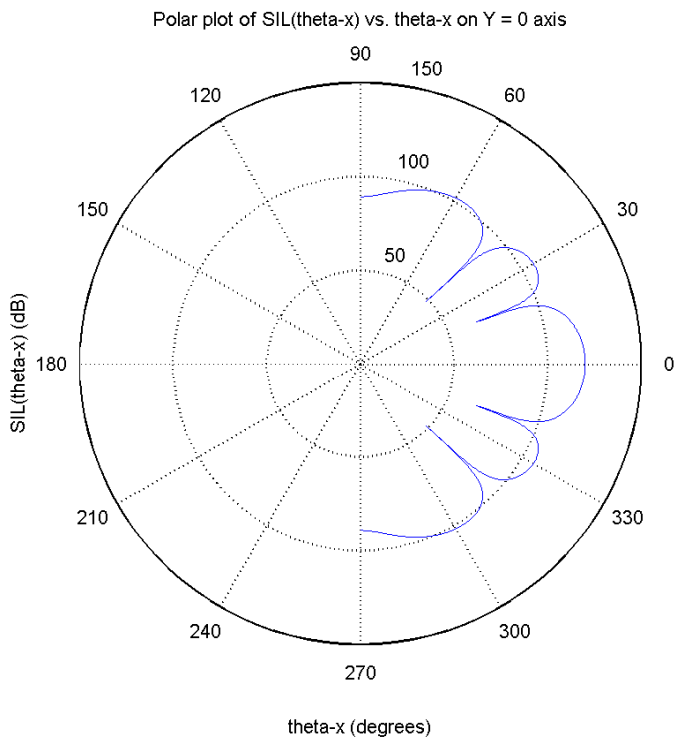


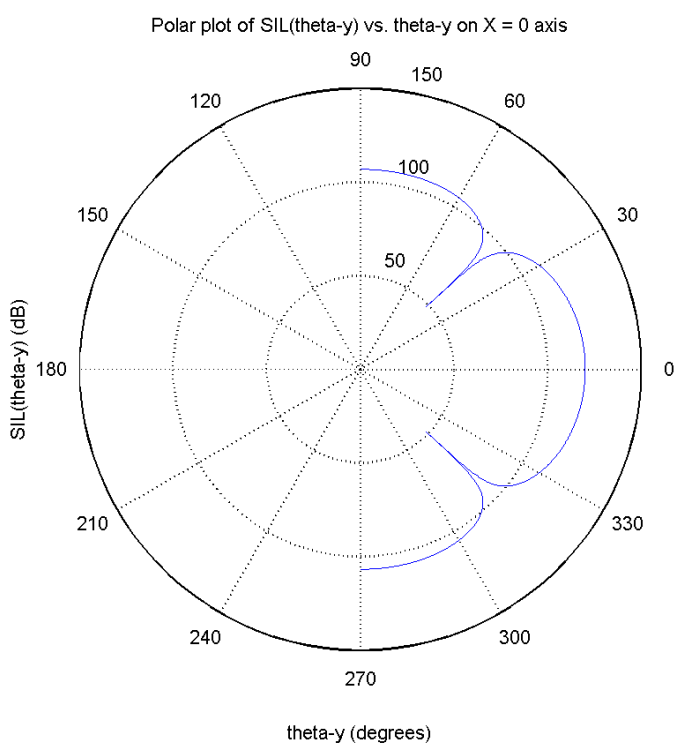
Plots of SIL *vs.* θ_y and SIL *vs.* y_{screen} for (b , a) = (1.0 m, 0.5 m):



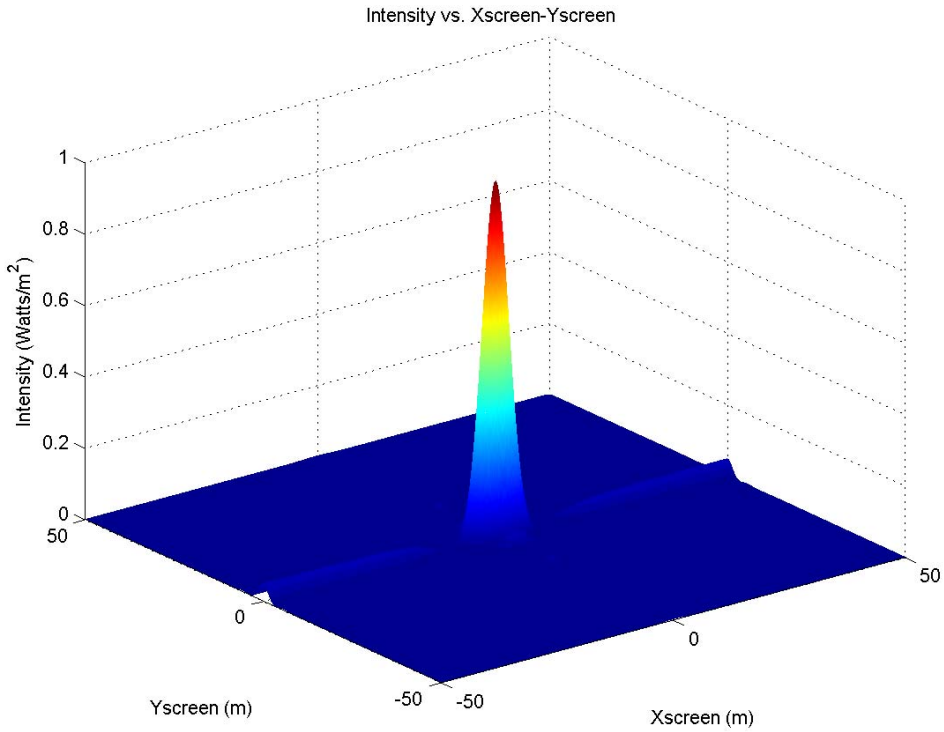


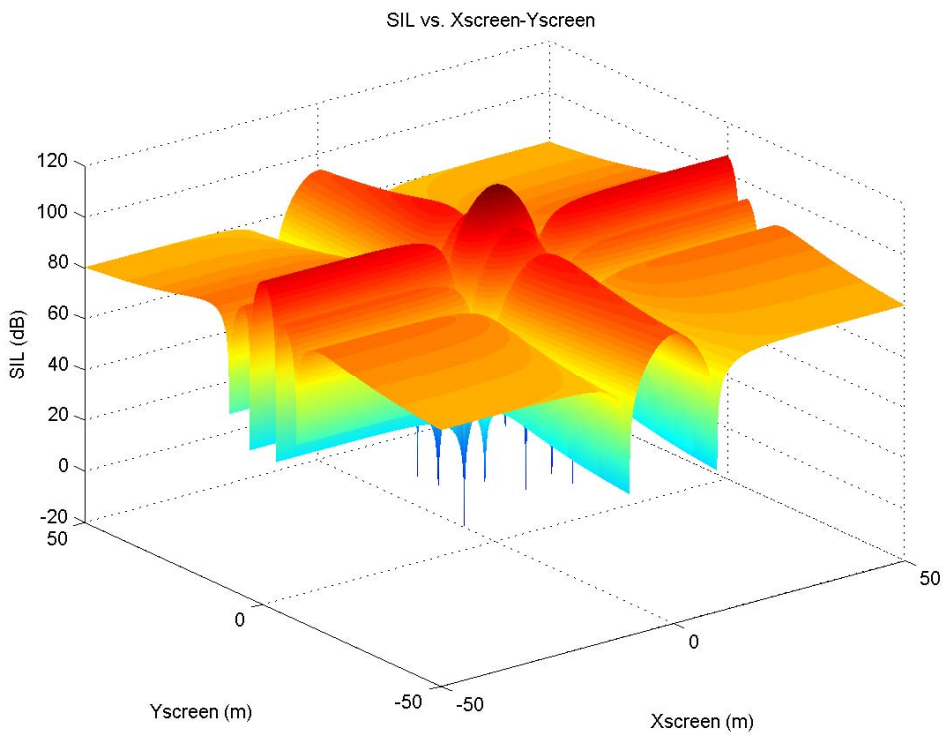
Plots of SIL *vs.* θ_x and SIL *vs.* θ_y for $(b\ ,\ a) = (1.0\ m,\ 0.5\ m)$:



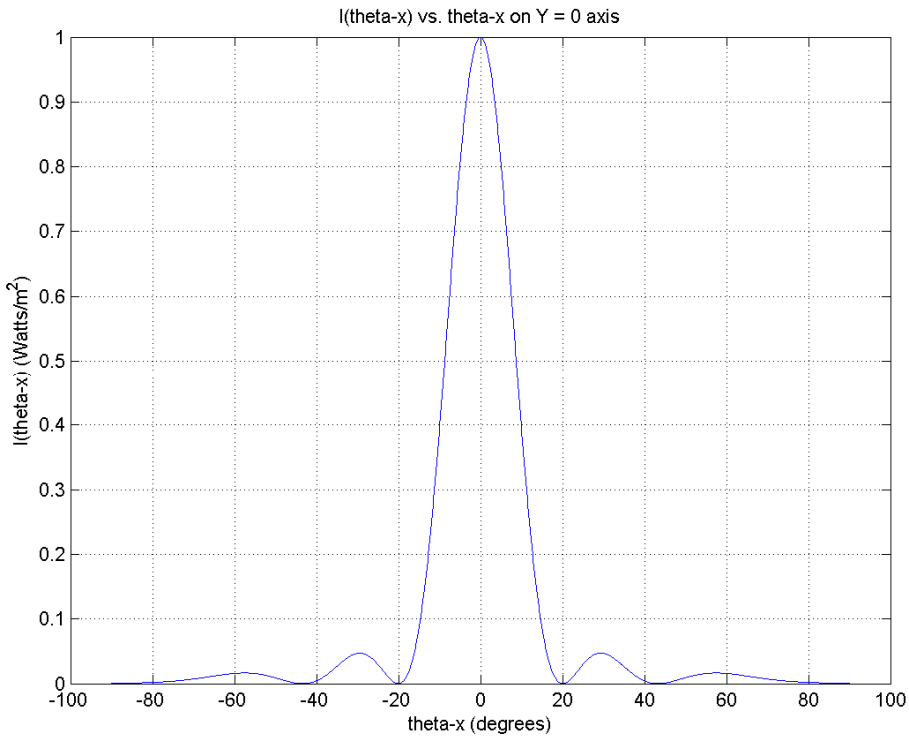


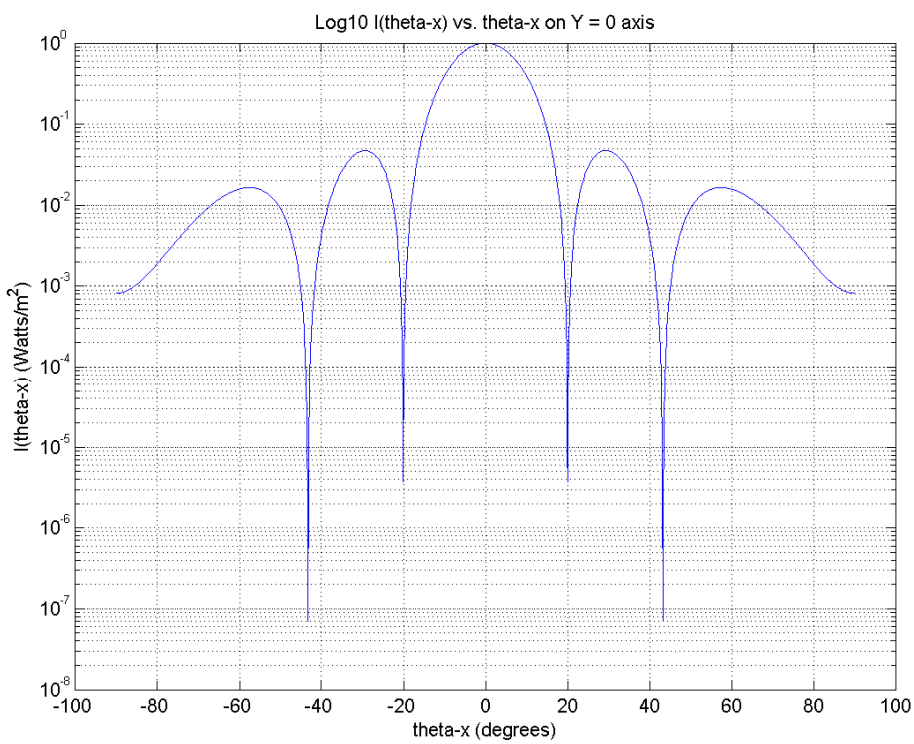
Plots of I_{tot} *vs.* $\left(x_{screen}, y_{screen}\right)$ and SIL *vs.* $\left(x_{screen}, y_{screen}\right)$ for (b , a) = (1.0 m, 0.5 m):



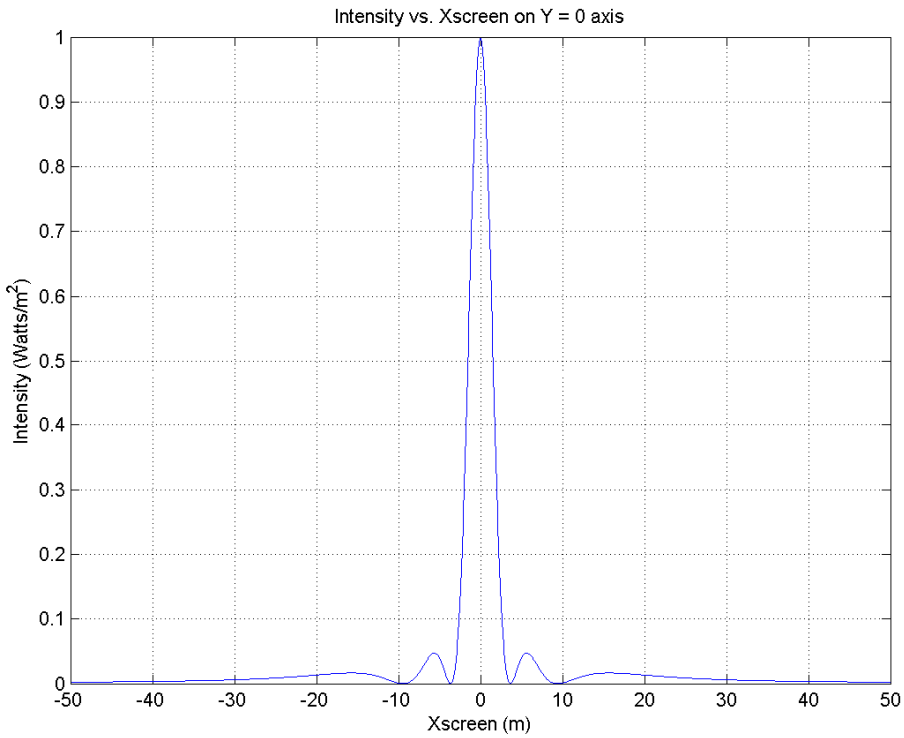


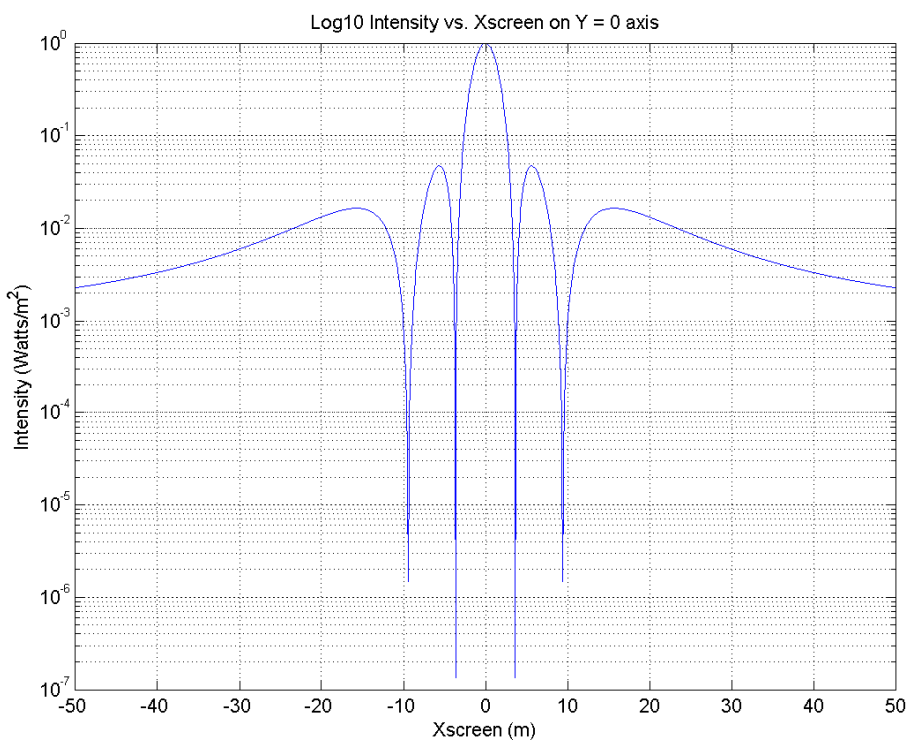
Linear and Semilog Plots of I_{tot} *vs.* θ_x for (b , a) = (1.0 m, 1.0 m):



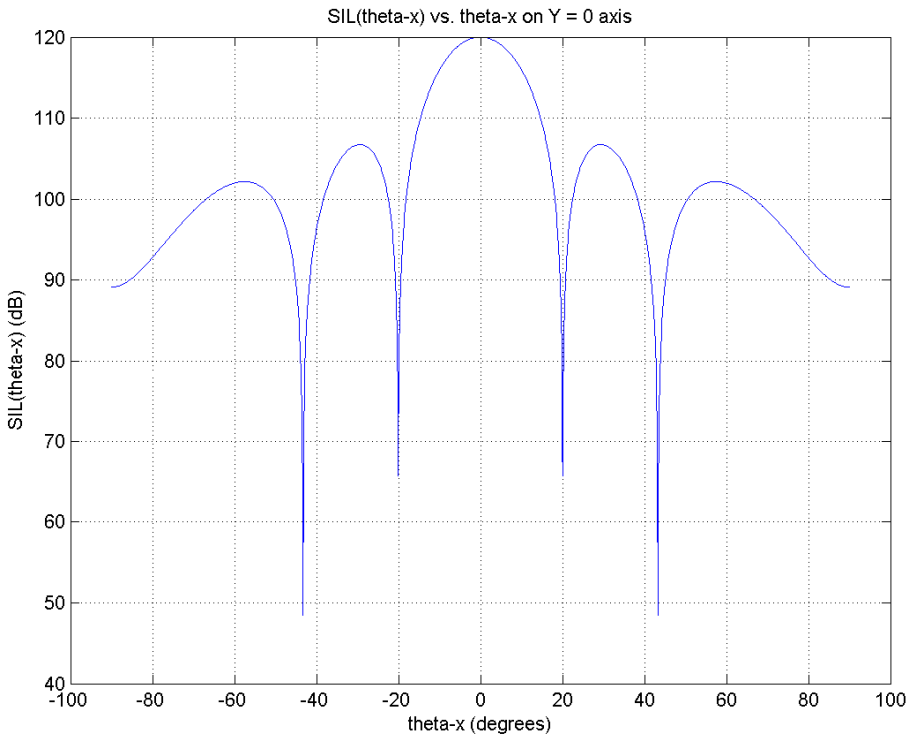


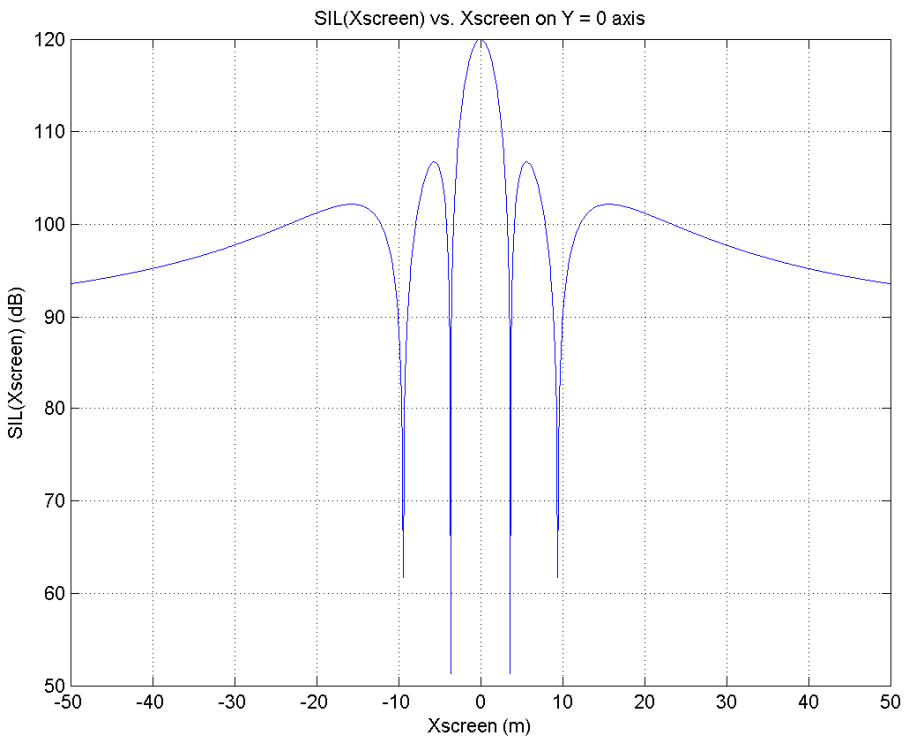
Linear and Semilog Plots of I_{tot} *vs.* x_{screen} for (b , a) = (1.0 m, 1.0 m):



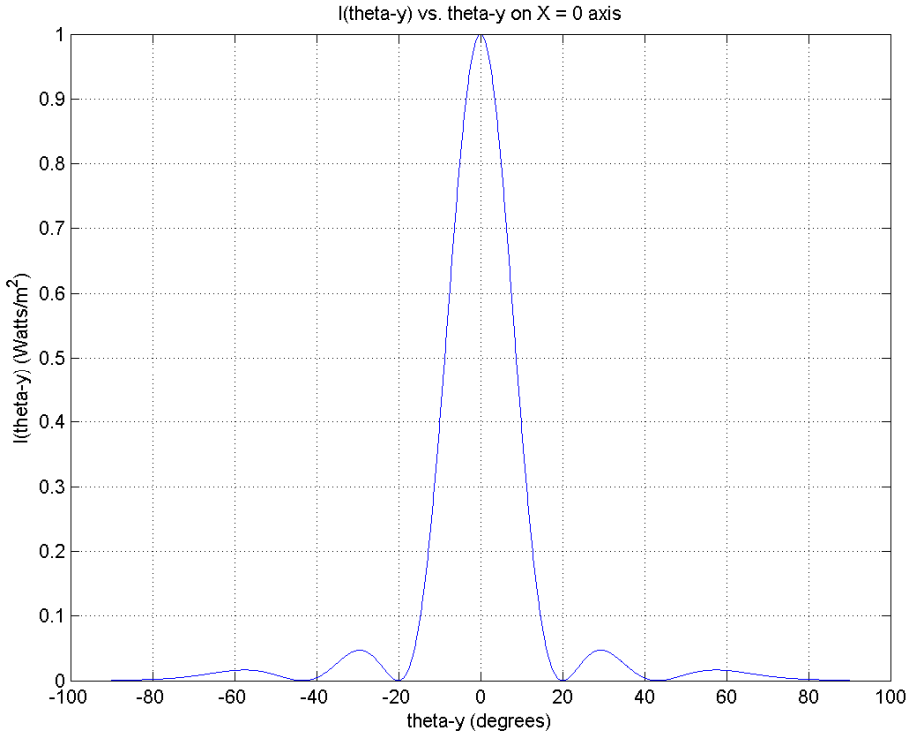


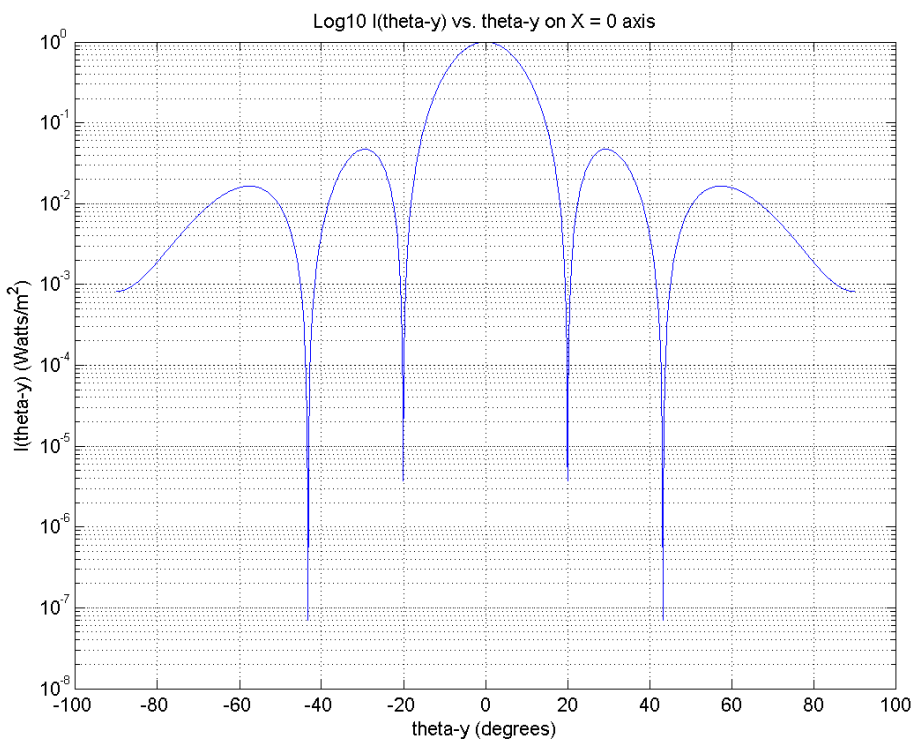
Plots of *SIL vs.* θ_x and *SIL vs.* x_{screen} for (b , a) = (1.0 m, 1.0 m):



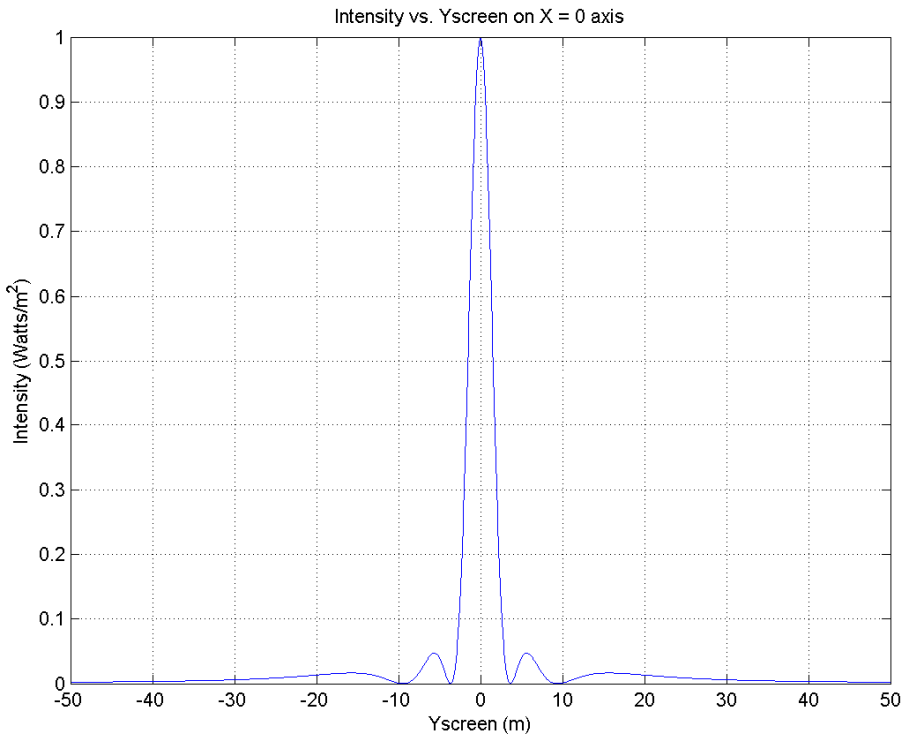


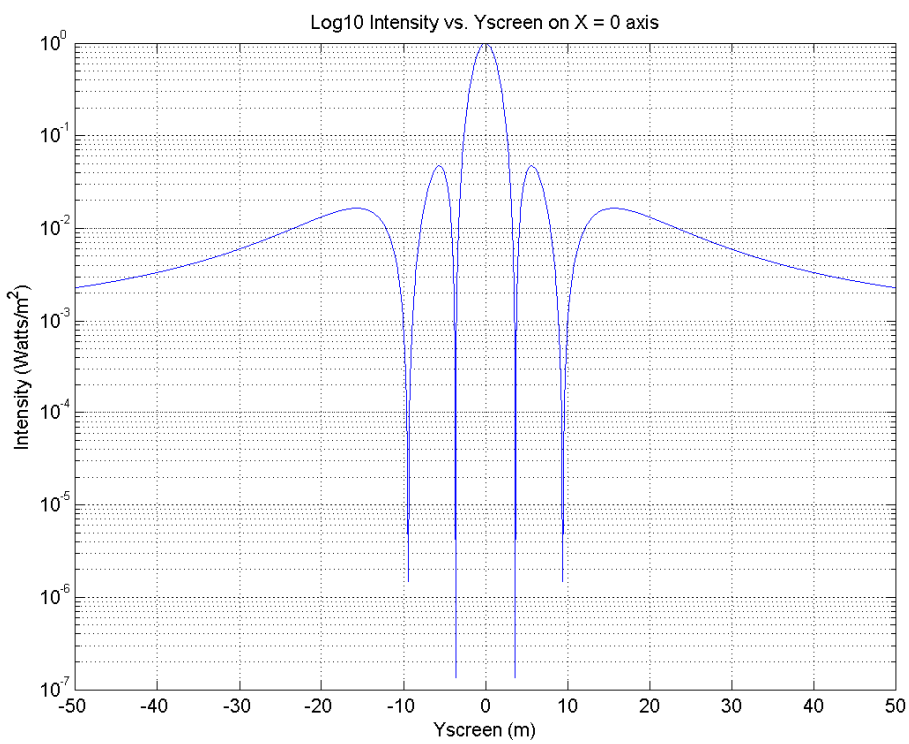
Linear and Semilog Plots of I_{tot} *vs.* θ_y for (b , a) = (1.0 m, 1.0 m):



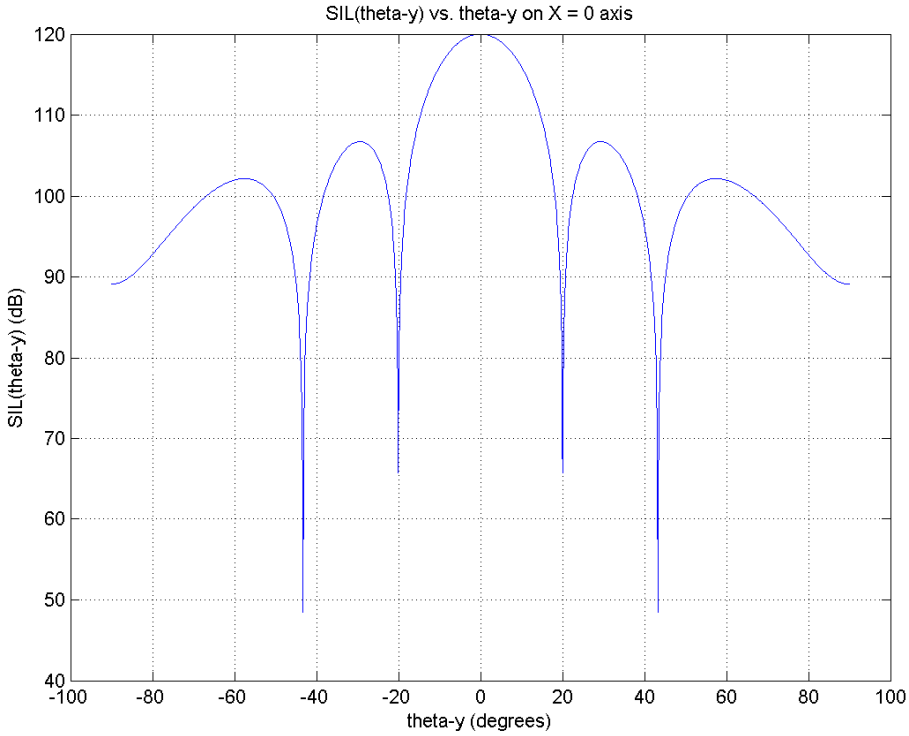


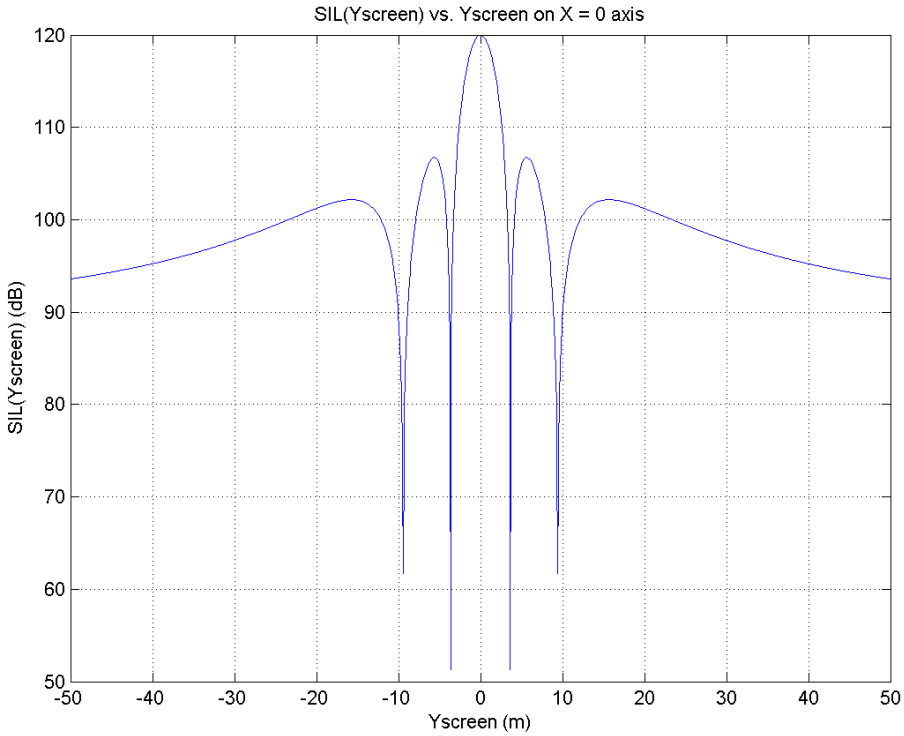
Linear and Semilog Plots of I_{tot} *vs.* y_{screen} for (b , a) = (1.0 m, 1.0 m):



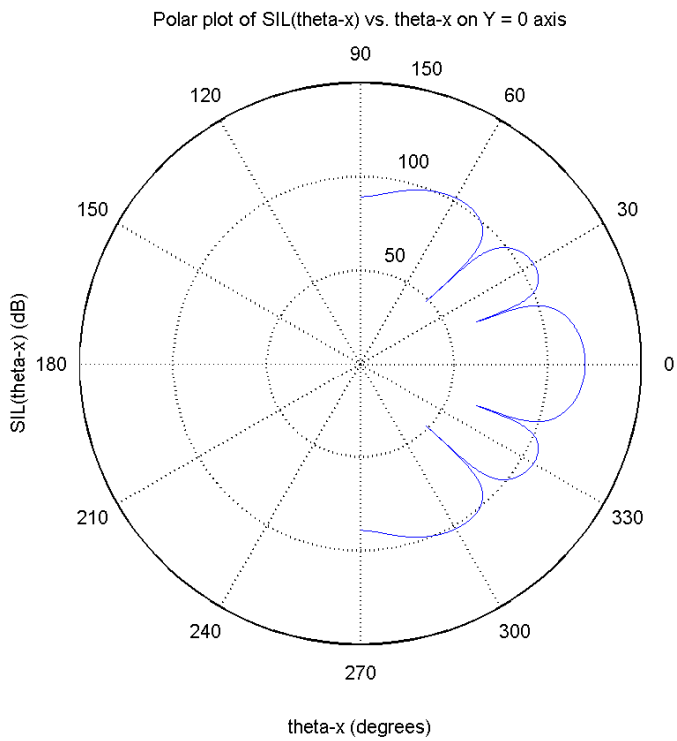


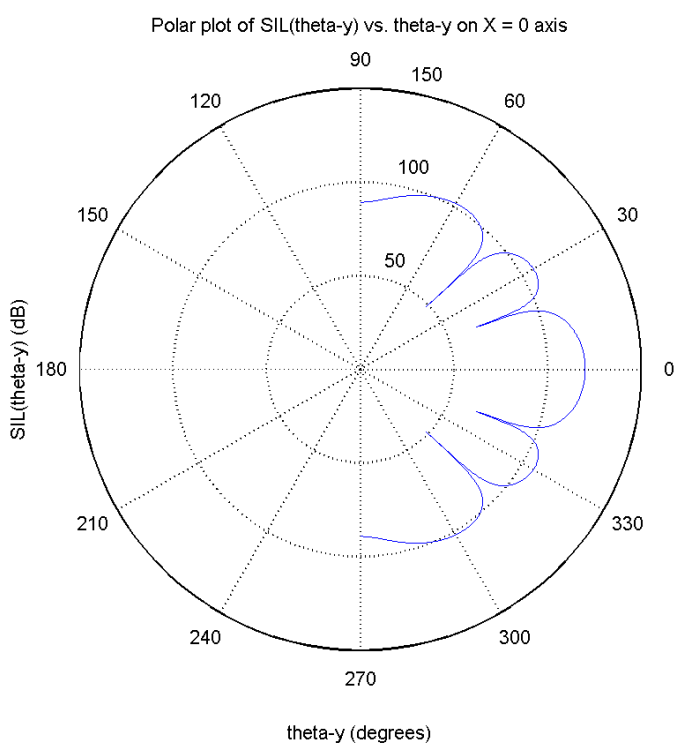
Plots of SIL *vs.* θ_y and SIL *vs.* y_{screen} for (b , a) = (1.0 m, 1.0 m):



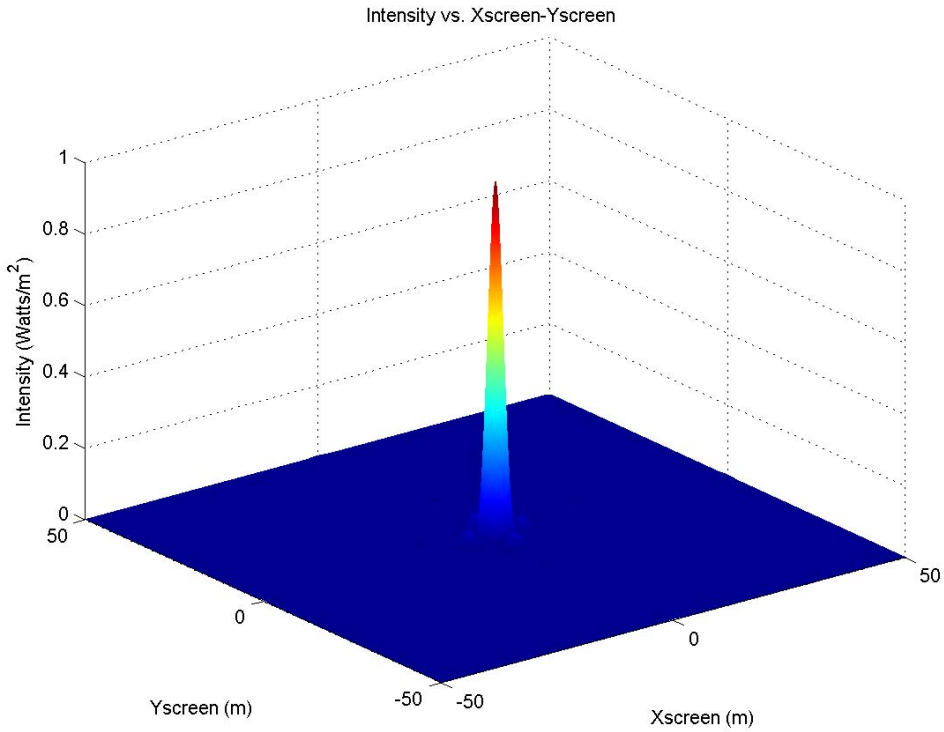


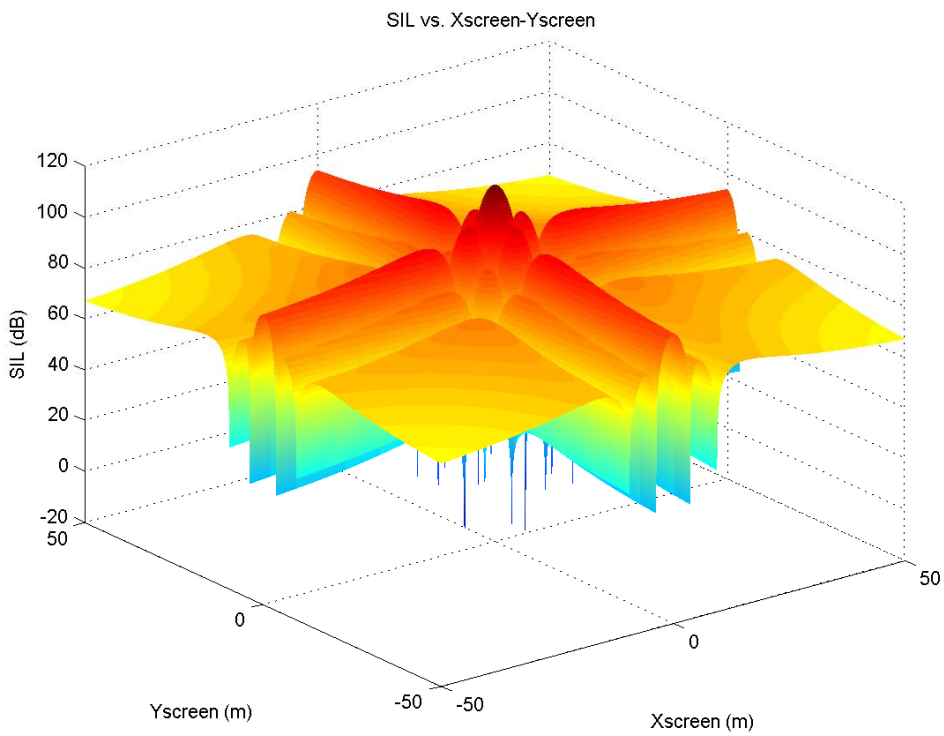
Plots of SIL *vs.* θ_x and SIL *vs.* θ_y for $(b\ ,\ a) = (1.0\ m,\ 0.5\ m)$:





Plots of I_{tot} *vs.* $\left(x_{screen}, y_{screen}\right)$ and SIL *vs.* $\left(x_{screen}, y_{screen}\right)$ for (b , a) = (1.0 m, 1.0 m):





Listing of the MATLAB code:

```
%==========================================================================
% Diffn_Rect_Aperture_Thy.m
%
% Diffraction through a rectangular aperture
% - simplest theory - far-field/plane-wave approx!
% Sound waves assumed to be propagating in free air/great wide-open!
%
%==========================================================================
%
% Written by Prof. Steven Errede  Last Updated: Feb. 7, 2011 13:15 hr
%
%==========================================================================
close all;
clear all;

single     thtxr(1800);
single     thtxd(1800);
single     thtyr(1800);
single     thtyd(1800);
single    Itotxt(1800);
single    Itotyt(1800);
single     SILxt(1800);
single     SILyt(1800);

single      xscr(2000);
single      yscr(2000);
single    Itotxs(2000);
single    Itotys(2000);
single     SILxs(2000);
single     SILys(2000);

single    Itotxy(2000,2000);
single LogItotxy(2000,2000);
single     SILxy(2000,2000);

% Specify x, y dimensions of the rectangular aperture:
Bx = 0.1; % 0.1; 1.0; 1.0; % x-axis is horizontal
Ay = 0.2; % 0.2; 0.5; 1.0; % y-axis is   vertical

% Specify other needed parameters:
Io     =    1.0;    % intensity from single slit (Watts/m^2)
Ir     = 1.0*10^-12;% reference sound intensity  (Watts/m^2)
Vair   =  343.0;    % speed of propagation of sound - free air (m/s)
freq   = 1000.0;    % frequency (Hz or cps)
lambda = Vair/freq; % wavelength (m)
Lobs   =   10.0;    % perp. distance observer from slits (m) n.b. lambda << Lobs

%=====================================
% Calculate 1-D Itot vs. theta-x,y:
% along x = 0 y-axis and y = 0 x-axis:
%=====================================
Thetad =  -90.0;    % x-angle of observer in degrees
dTheta =    0.1;    % x-step angle in degrees
for i = 1:1800;
    thtxd(i) = Thetad;            % x-angle of observer in degrees
    Thetax   = (pi/180.0)*Thetad; % x-angle of observer in radians
    thtxr(i) = Thetax;

     deltax  = (pi*Bx/lambda)*sin(Thetax); % resultant x-phase diff (radians)

   Itotxt(i) = Io*(sin(deltax)/deltax)^2; % total x-intensity (Watts/m^2)
    SILxt(i) = 10.0*log10(Itotxt(i)/Ir);  % Sound Intensity Level (dB)

     Thetad  = Thetad + dTheta;   % increment x-angle for next calculation
end

Thetad =  -90.0;    % y-angle of observer in degrees
dTheta =    0.1;    % y-step angle in degrees
```

```
for i = 1:1800;
    thtyd(i) = Thetad;           % y-angle of observer in degrees
    Thetay  = (pi/180.0)*Thetad; % y-angle of observer in radians
    thtyr(i) = Thetay;

     deltay  = (pi*Ay/lambda)*sin(Thetay); % resultant y-phase diff (radians)

   Itotyt(i) = Io*(sin(deltay)/deltay)^2; % total y-intensity (Watts/m^2)
    SILyt(i) = 10.0*log10(Itotyt(i)/Ir);  % Sound Intensity Level (dB)

     Thetad  = Thetad + dTheta;   % increment angle for next calculation
end

figure(01);
plot(thtxd,Itotxt,'b');
grid on;
xlabel('theta-x (degrees)');
ylabel('I(theta-x) (Watts/m^{2})');
title('I(theta-x) vs. theta-x on Y = 0 axis');

figure(02);
semilogy(thtxd,Itotxt,'b');
grid on;
xlabel('theta-x (degrees)');
ylabel('I(theta-x) (Watts/m^{2})');
title('Log10 I(theta-x) vs. theta-x on Y = 0 axis');

figure(03);
plot(thtyd,Itotyt,'b');
grid on;
xlabel('theta-y (degrees)');
ylabel('I(theta-y) (Watts/m^{2})');
title('I(theta-y) vs. theta-y on X = 0 axis');

figure(04);
semilogy(thtyd,Itotyt,'b');
grid on;
xlabel('theta-y (degrees)');
ylabel('I(theta-y) (Watts/m^{2})');
title('Log10 I(theta-y) vs. theta-y on X = 0 axis');

figure(05);
plot(thtxd,SILxt,'b');
grid on;
xlabel('theta-x (degrees)');
ylabel('SIL(theta-x) (dB)');
title('SIL(theta-x) vs. theta-x on Y = 0 axis');

figure(06);
plot(thtyd,SILyt,'b');
grid on;
xlabel('theta-y (degrees)');
ylabel('SIL(theta-y) (dB)');
title('SIL(theta-y) vs. theta-y on X = 0 axis');

figure(07);
polar(thtxr,SILxt,'b');
grid on;
xlabel('theta-x (degrees)');
ylabel('SIL(theta-x) (dB)');
title('Polar plot of SIL(theta-x) vs. theta-x on Y = 0 axis');

figure(08);
polar(thtyr,SILyt,'b');
grid on;
xlabel('theta-y (degrees)');
ylabel('SIL(theta-y) (dB)');
title('Polar plot of SIL(theta-y) vs. theta-y on X = 0 axis');


%=====================================
```

```
% Calculate 1-D Itot vs. x,y-screen:
% along x = 0 y-axis and y = 0 x-axis:
%=====================================
 x = -50.00; % starting position on screen (m)
dx =   0.05; % step size on screen (m);
for i = 1:2000;
     xscr(i) = x;            % position of observer on perp. screen (m)
     Thetax  = atan(x/Lobs); % angle theta of observer in radians

     deltax  = (pi*Bx/lambda)*sin(Thetax); % resultant x-phase diff (radians)

   Itotxs(i) = Io*(sin(deltax)/deltax)^2; % total x-intensity (Watts/m^2)
    SILxs(i) = 10.0*log10(Itotxs(i)/Ir);  % Sound Intensity Level (dB)

     x  = x + dx;   % increment screen position for next calculation
end

 y = -50.00; % starting position on screen (m)
dy =   0.05; % step size on screen (m);
for i = 1:2000;
     yscr(i) = y;            % position of observer on perp. screen (m)
     Thetay  = atan(y/Lobs); % angle theta of observer in radians

     deltay  = (pi*Ay/lambda)*sin(Thetay); % resultant y-phase diff (radians)

   Itotys(i) = Io*(sin(deltay)/deltay)^2; % total y-intensity (Watts/m^2)
    SILys(i) = 10.0*log10(Itotys(i)/Ir); % Sound Intensity Level (dB)

     y  = y + dy;   % increment screen position for next calculation
end

figure(11);
plot(xscr,Itotxs,'b');
grid on;
xlabel('Xscreen (m)');
ylabel('Intensity (Watts/m^{2})');
title('Intensity vs. Xscreen on Y = 0 axis');

figure(12);
semilogy(xscr,Itotxs,'b');
grid on;
xlabel('Xscreen (m)');
ylabel('Intensity (Watts/m^{2})');
title('Log10 Intensity vs. Xscreen on Y = 0 axis');

figure(13);
plot(yscr,Itotys,'b');
grid on;
xlabel('Yscreen (m)');
ylabel('Intensity (Watts/m^{2})');
title('Intensity vs. Yscreen on X = 0 axis');

figure(14);
semilogy(yscr,Itotys,'b');
grid on;
xlabel('Yscreen (m)');
ylabel('Intensity (Watts/m^{2})');
title('Log10 Intensity vs. Yscreen on X = 0 axis');

figure(15);
plot(xscr,SILxs,'b');
grid on;
xlabel('Xscreen (m)');
ylabel('SIL(Xscreen) (dB)');
title('SIL(Xscreen) vs. Xscreen on Y = 0 axis');

figure(16);
plot(yscr,SILys,'b');
grid on;
xlabel('Yscreen (m)');
ylabel('SIL(Yscreen) (dB)');
```

```
title('SIL(Yscreen) vs. Yscreen on X = 0 axis');


%==========================================================================
beep;
fprintf('\n Very CPU-intensive I(x,y) vs. x,y calcs - please be patient!! \n')
%==========================================================================

%=========================================
% Calculate 2D Itot vs. x,y-screen:
%=========================================
 x = -50.00; % starting position on screen (m)
dx =   0.05; % step size on screen (m);
for j = 1:2000;
     xscr(j) = x;            % x-pos'n of observer on perp. screen (m)
     Thetax  = atan(x/Lobs); % x-angle of observer in radians

     deltax  = (pi*Bx/lambda)*sin(Thetax); % resultant x-phase diff (radians)

      Itotx  = Io*(sin(deltax)/deltax)^2; % total x-intensity (Watts/m^2)

     y = -50.00; % starting position on screen (m)
    dy =   0.05; % step size on screen (m);
    for i = 1:2000;
         yscr(i) = y;            % y-pos'n of observer on perp. screen (m)
         Thetay  = atan(y/Lobs); % y-angle of observer in radians

         deltay  = (pi*Ay/lambda)*sin(Thetay); % resultant y-phase diff (radians)

          Itoty  = Io*(sin(deltay)/deltay)^2; % total y-intensity (Watts/m^2)

     Itotxy(j,i) = Itotx*Itoty/Io; % total intensity (Watts/m^2)
  LogItotxy(j,i) = log10(Itotxy(j,i));
      SILxy(j,i) = 10.0*log10(Itotxy(j,i)/Ir); % Sound Intensity Level (dB)

         y  = y + dy;   % increment screen position for next calculation
    end
     x  = x + dx;   % increment screen position for next calculation
end


figure(21);
surf(xscr,yscr,Itotxy);
shading interp;
xlabel('Xscreen (m)');
ylabel('Yscreen (m)');
zlabel('Intensity (Watts/m^{2})');
title ('Intensity vs. Xscreen-Yscreen');

figure(22);
surf(xscr,yscr,LogItotxy);
shading interp;
xlabel('Xscreen (m)');
ylabel('Yscreen (m)');
zlabel('Log Intensity (Watts/m^{2})');
title ('Log Intensity vs. Xscreen-Yscreen');

figure(23);
surf(xscr,yscr,SILxy);
shading interp;
xlabel('Xscreen (m)');
ylabel('Yscreen (m)');
zlabel('SIL (dB)');
title ('SIL vs. Xscreen-Yscreen');

%==========================================================================
beep;
fprintf('\n Diffn Thru Rect Aperture Simple Thy Calculation completed !!! \n')
%==========================================================================
```