N-Slit Interference in 2-Dimensions – Simplest Theory

In this example, we show plots of the sound intensity *vs*. angles θ_x, θ_y and observer/listener positions x_{screen}, y_{screen} on a screen for the simplest 2-D theory of *N*-slit interference – where N_x and N_y sound sources, all in phase with each other, are assumed to be comprised of infinitely long, parallel slits of infinitely narrow width – *i.e*. infinitesimally thin slits, each separated from each other by a distance *d*, oriented in the *x* and *y* directions, respectively. The observer/listener is located far from the sound sources, a perpendicular *z*-distance $L\ (m)$ away, and such that the conditions $d \ll L$ **.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 1-D *N*-slit interference in this simplest theory is given by the product of two ***independent*** interference factors (see P406POM Lecture Notes P406POM_Lect3_Part2):

$$I_{tot}\left(x_{screen}, y_{screen}\right) = I_o \left\{\frac{\sin^2\left(N_x\delta_x/2\right)}{\sin^2\left(\delta_x/2\right)}\right\}\cdot\left\{\frac{\sin^2\left(N_y\delta_y/2\right)}{\sin^2\left(\delta_y/2\right)}\right\} \text{ and } SIL \equiv 10\log_{10}\left(I_{tot}/I_{ref}\right)\ (dB)$$

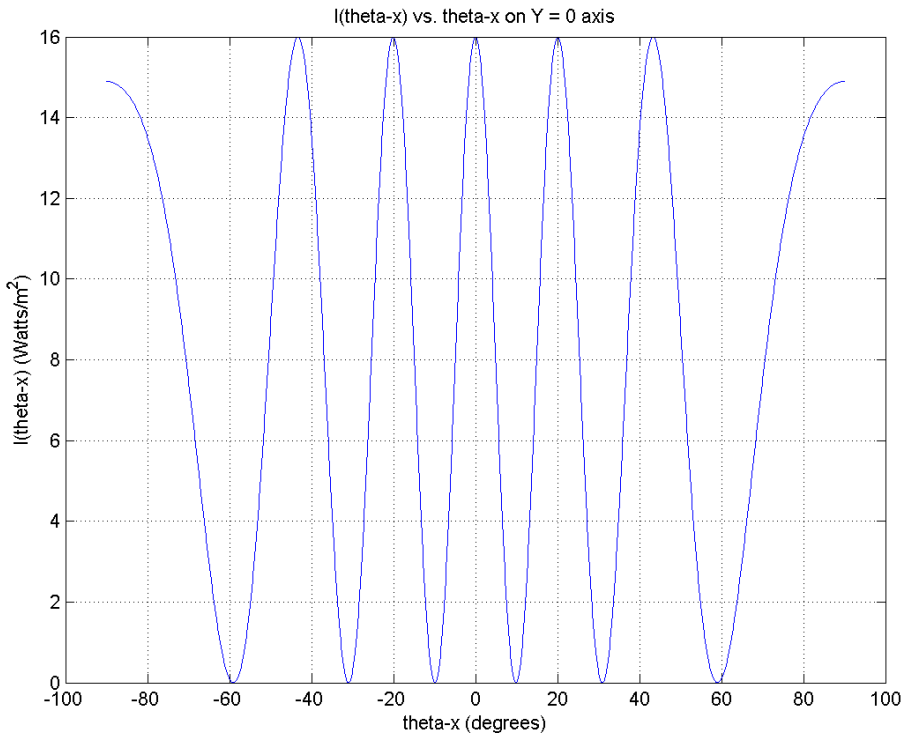
where $I_o\ \left(Watts/m^2\right)$ is the sound intensity associated with a single slit, the phase $\delta_{x,y} = k\Delta L_{x,y}$ $(radians)$, $k = 2\pi/\lambda\ (radians/m)$ is the wavenumber and $\Delta L_{x,y}\ (m)$ is the path length ***difference*** between adjacent sound sources to the observer/listener, located far away from the sound sources, in the *x*, *y* direction, respectively. $I_{ref} \equiv 10^{-12}\ \left(Watts/m^2\right)$ is the reference for the sound intensity level (SIL).

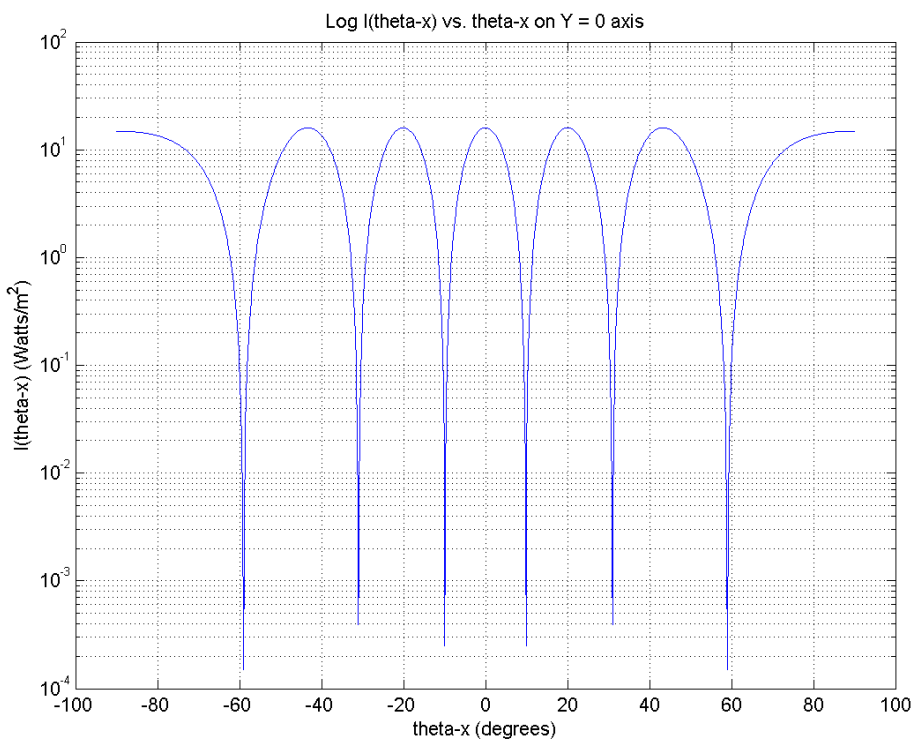
Minima – *i.e*. intensity zeroes (complete destructive interference) occur when the numerator factor $N\,\delta/2 = \pm\pi, \pm 2\pi, \pm 3\pi, \ldots = n\pi$, $n = \pm 1, \pm 2, \pm 3, \ldots$ <u>*except*</u> when the denominator factor simultaneously has $\delta/2 = \pm\pi, \pm 2\pi, \pm 3\pi, \ldots = n\pi$, $n = \pm 1, \pm 2, \pm 3, \ldots$ then have <u>*global maxima*</u> of the intensity, where $I_{tot} = N^2 I_o$.

From simple trigonometry, it is easy to show that the path length difference $\Delta L_{x,y} = d\sin\theta_{x,y}$, where $\theta_{x,y}$ is the angle the observer/listener makes with respect to the normal, or forward axis of the array of *N* slits relative to the *x*, *y* direction, respectively. The corresponding location of the observer's position $\left(x_{screen}, y_{screen}\right)$ on a screen located a perpendicular distance *L* away from the *N* sound sources is: $x_{screen} = L\tan\theta_x\ \left(y_{screen} = L\tan\theta_y\right)$, or conversely: $\theta_x = \tan^{-1}\left(x_{screen}/L\right)\ \left(\theta_y = \tan^{-1}\left(y_{screen}/L\right)\right)$, respectively.

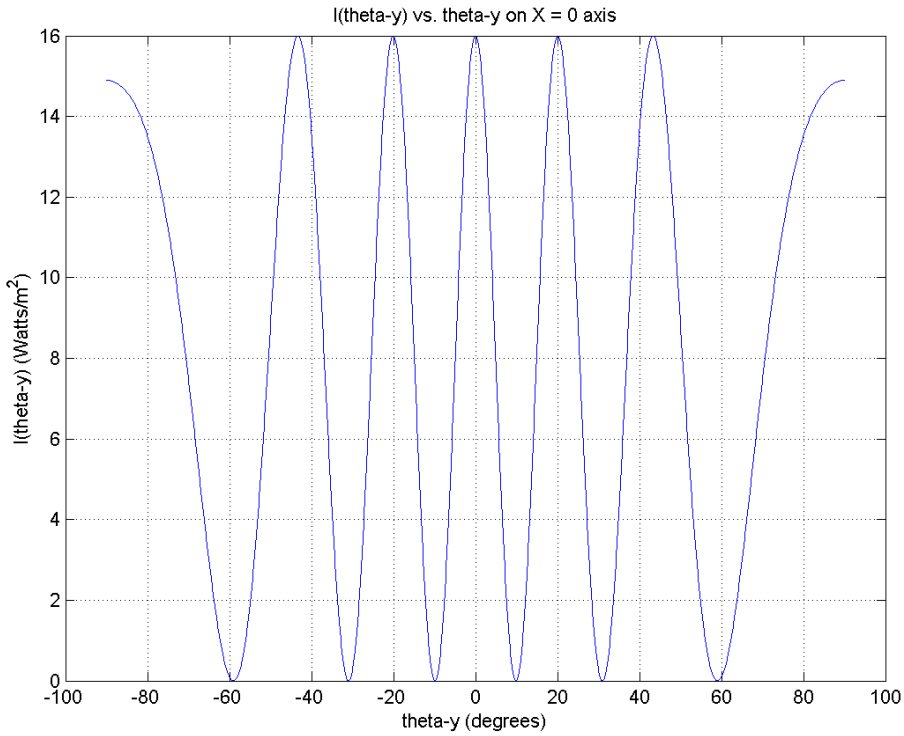
We coded up the above formulas using MATLAB to make plots of I_{tot} *vs.* $\theta_{x,y}$ and I_{tot} *vs.* x_{screen}, y_{screen} *e.g*. for the combinations (N_x, N_y) = (2,2), (5,3) and (10,4) slits, with the following parameter values: $I_o = 1\ Watt/m^2$, slit separation distance(s) of $d = 1\,m$, observer/listener distance (at $\theta_x = \theta_y = 0$) 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 $\lambda = v_{air}/f = 0.345\,m$.

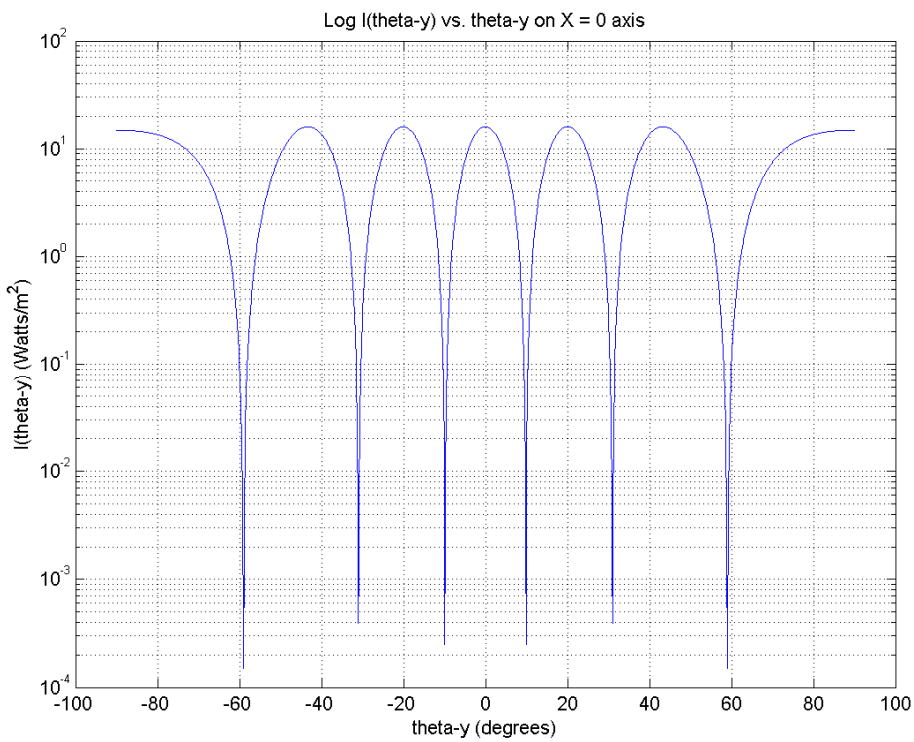
Plots of I_{tot} *vs.* θ_x for $(N_x, N_y) = (2,2)$ slits:



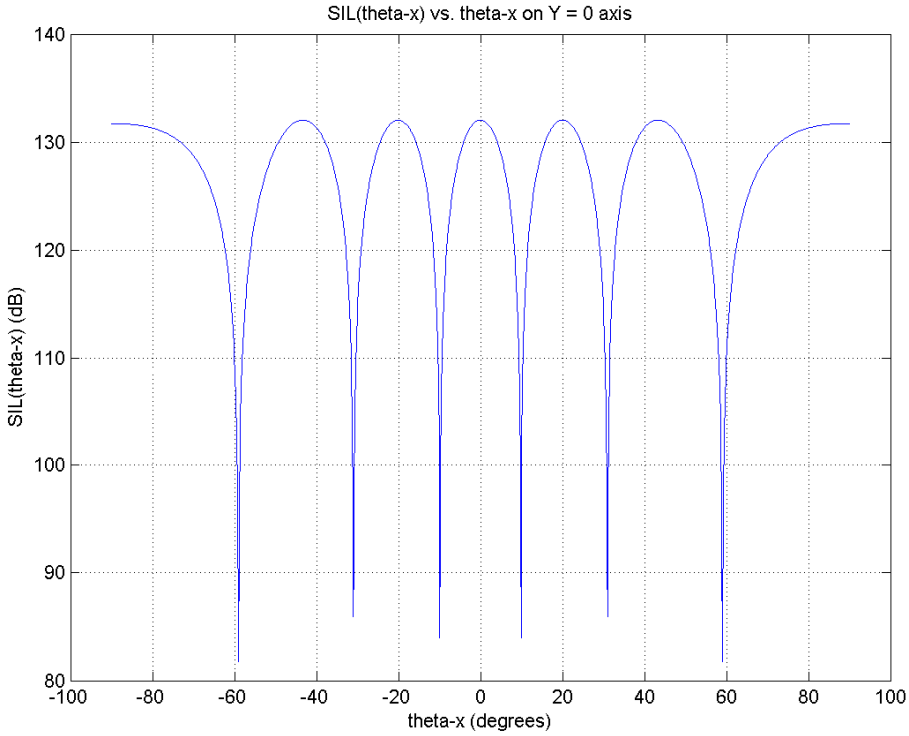


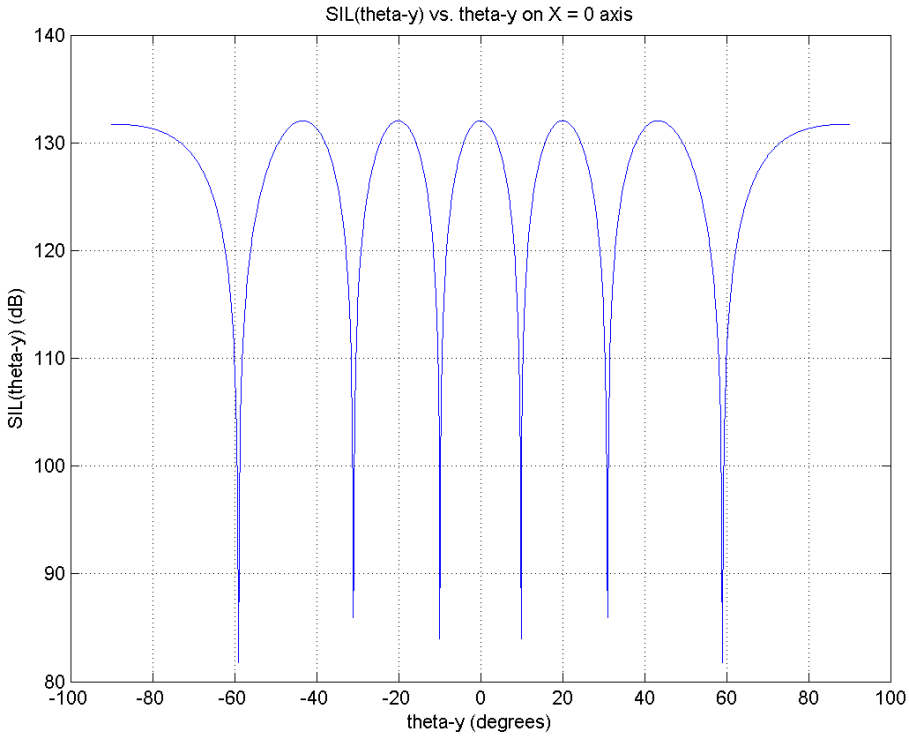
Plots of I_{tot} *vs.* θ_y for $(N_x, N_y) = (2,2)$ slits:



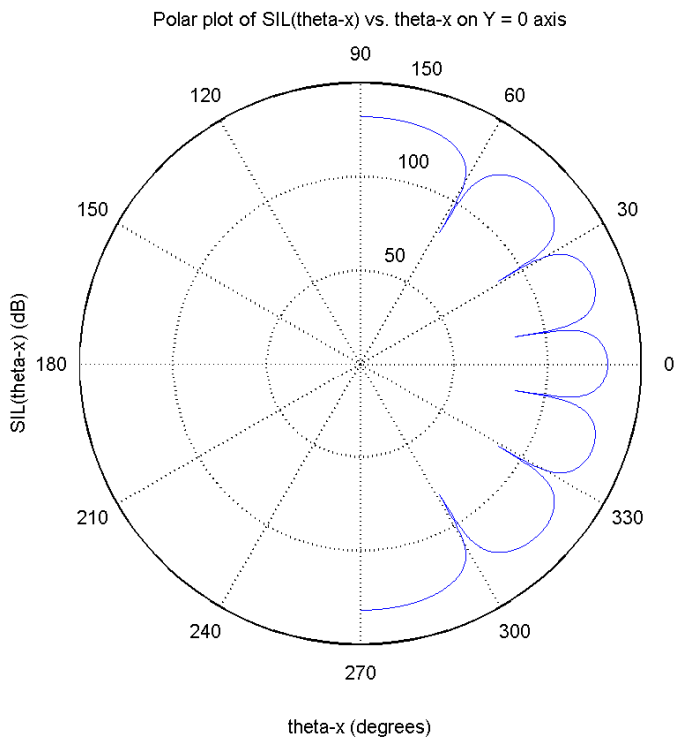


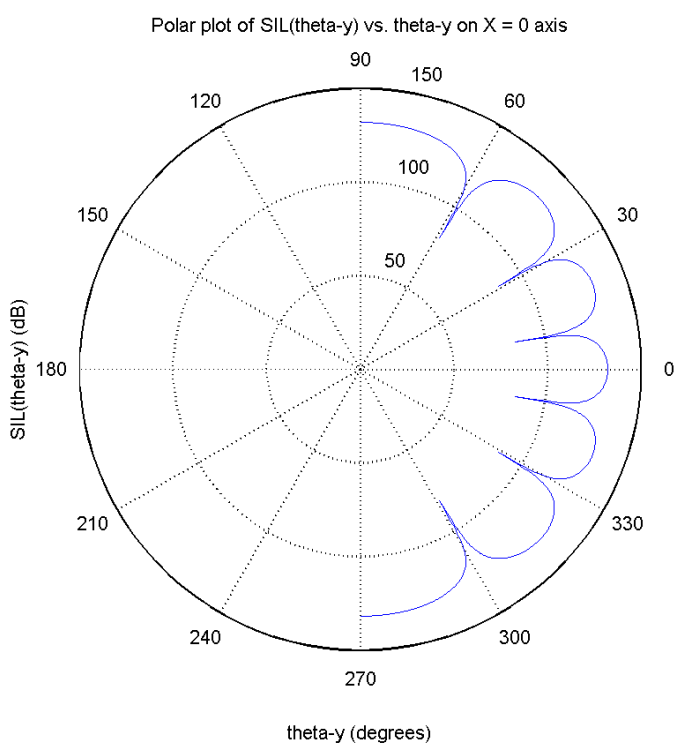
Plots of *SIL vs.* θ_x and *SIL vs.* θ_y for $(N_x, N_y) = (2,2)$ slits:



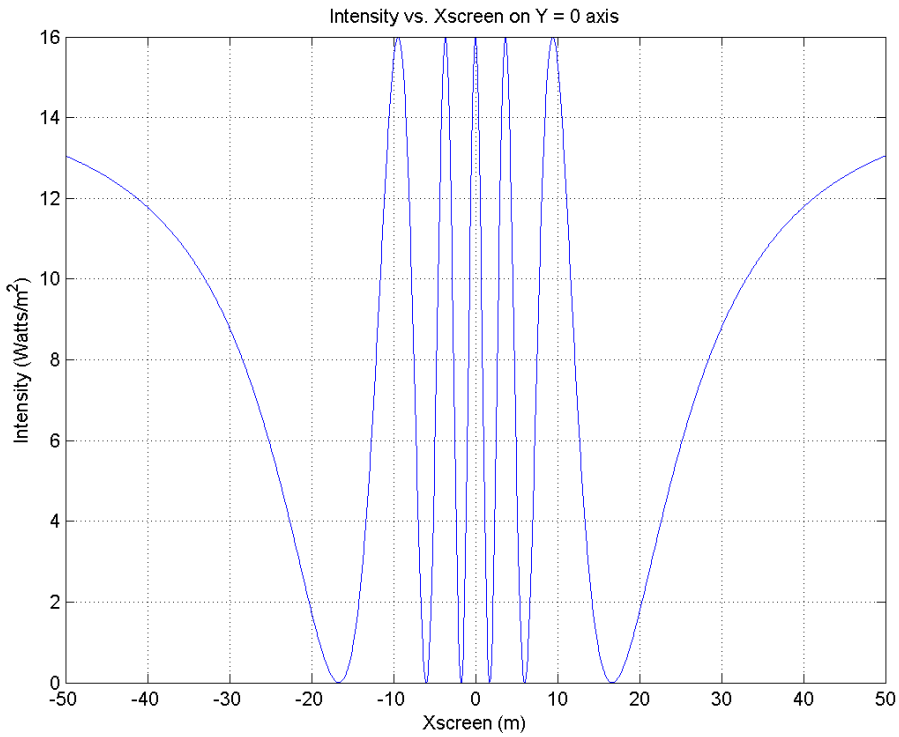


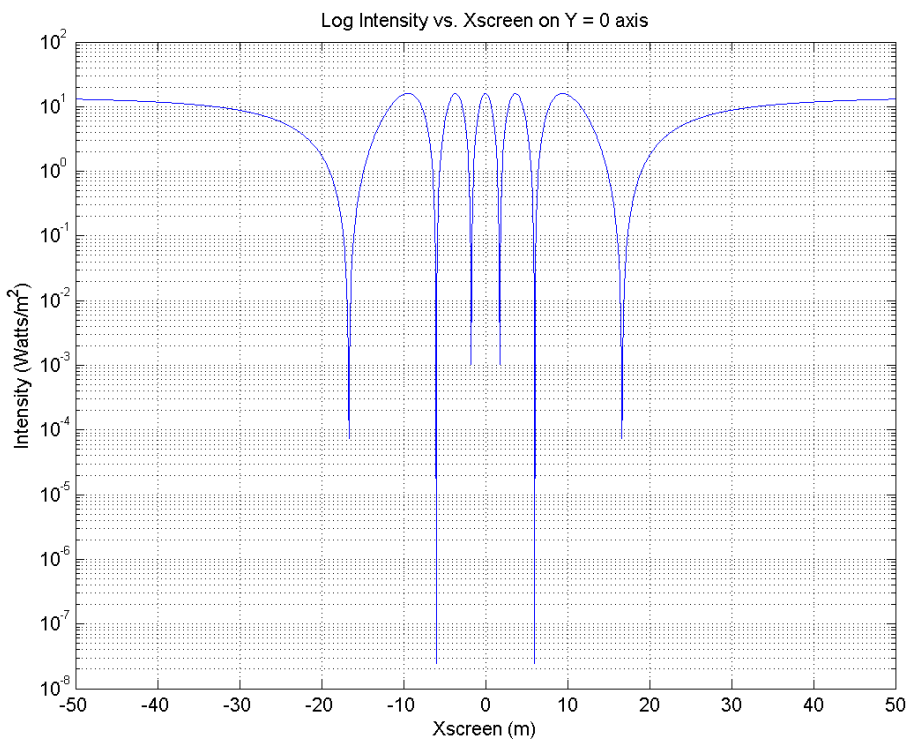
Polar plots of SIL *vs.* θ_x and SIL *vs.* θ_y for $(N_x, N_y) = (2,2)$ slits:



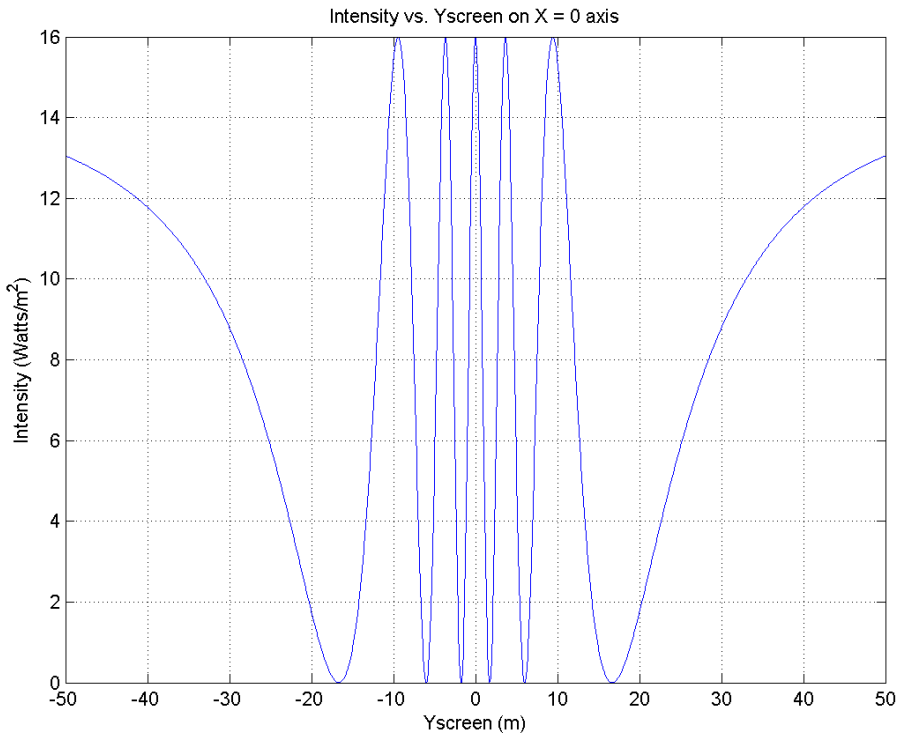


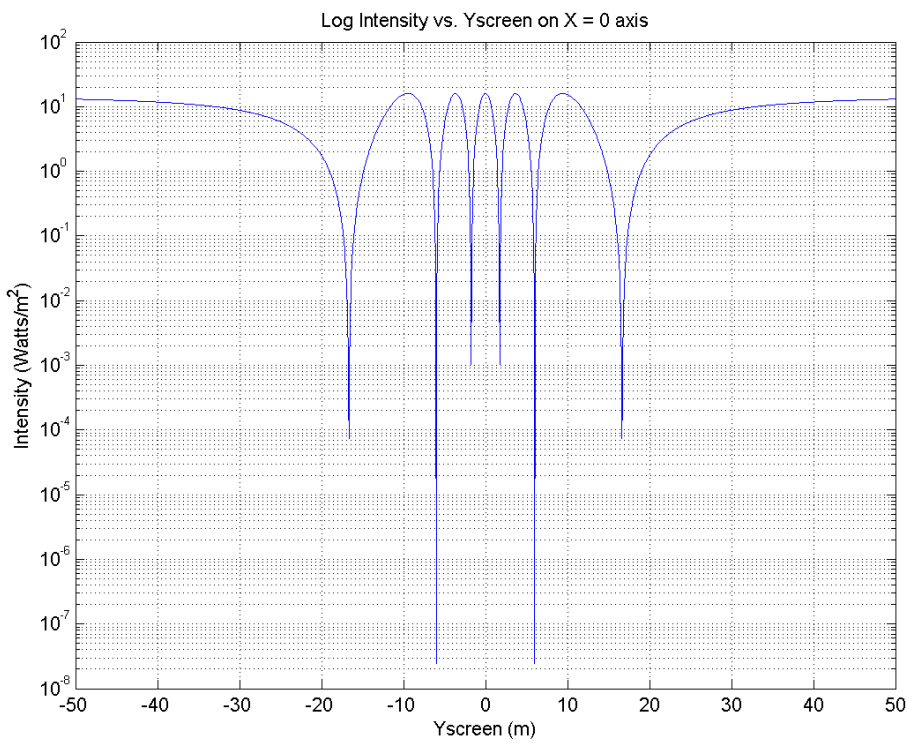
Plots of I_{tot} *vs.* x_{screen} for $(N_x, N_y) = (2,2)$ slits:



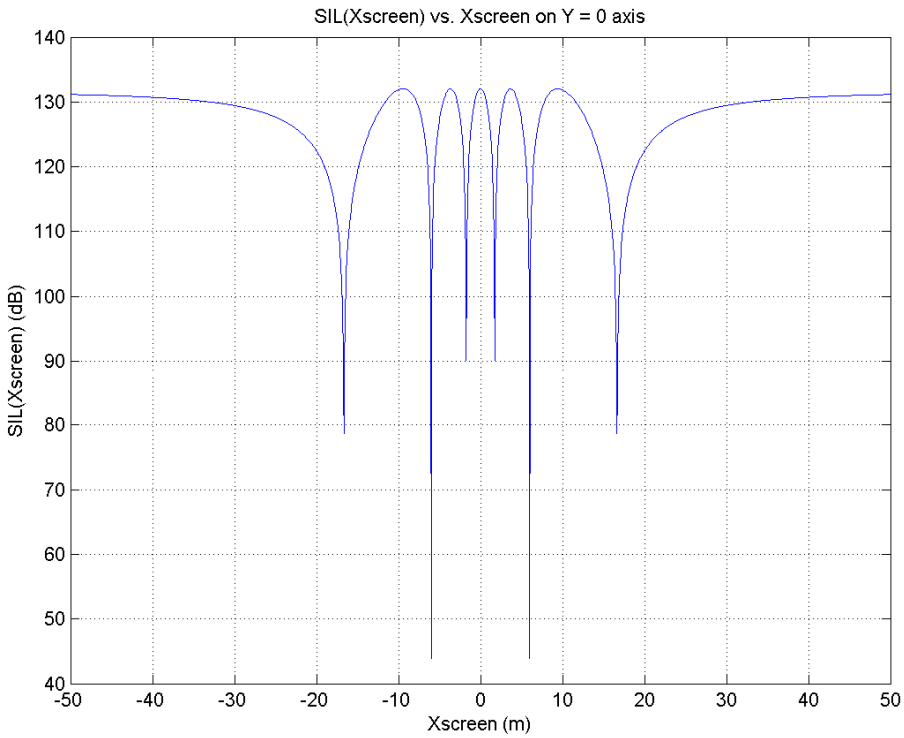


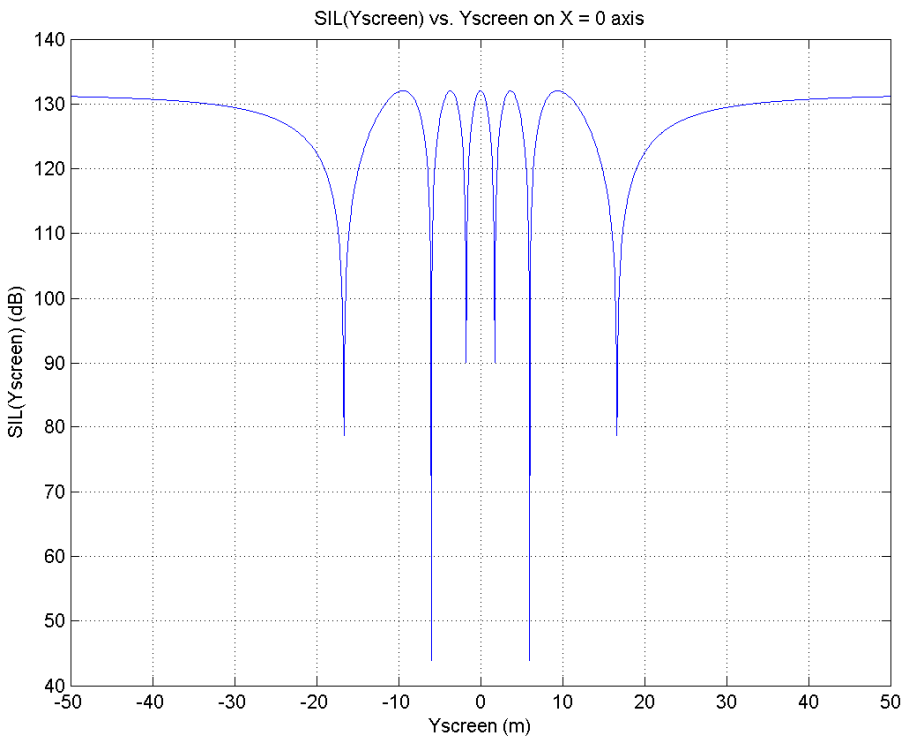
Plots of I_{tot} *vs.* y_{screen} for $(N_x, N_y) = (2,2)$ slits:



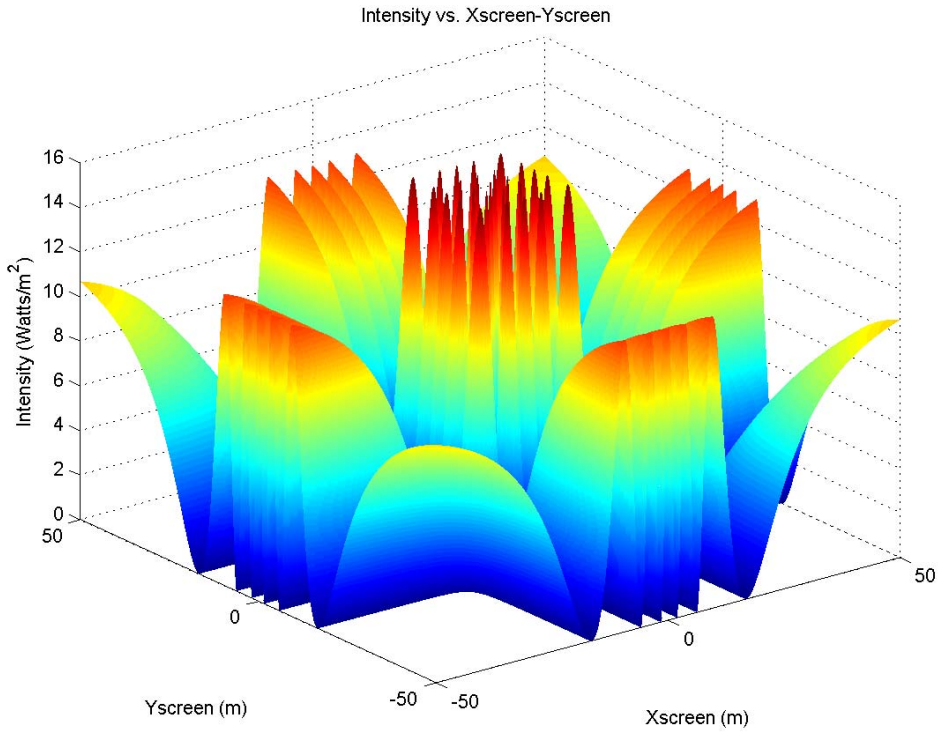


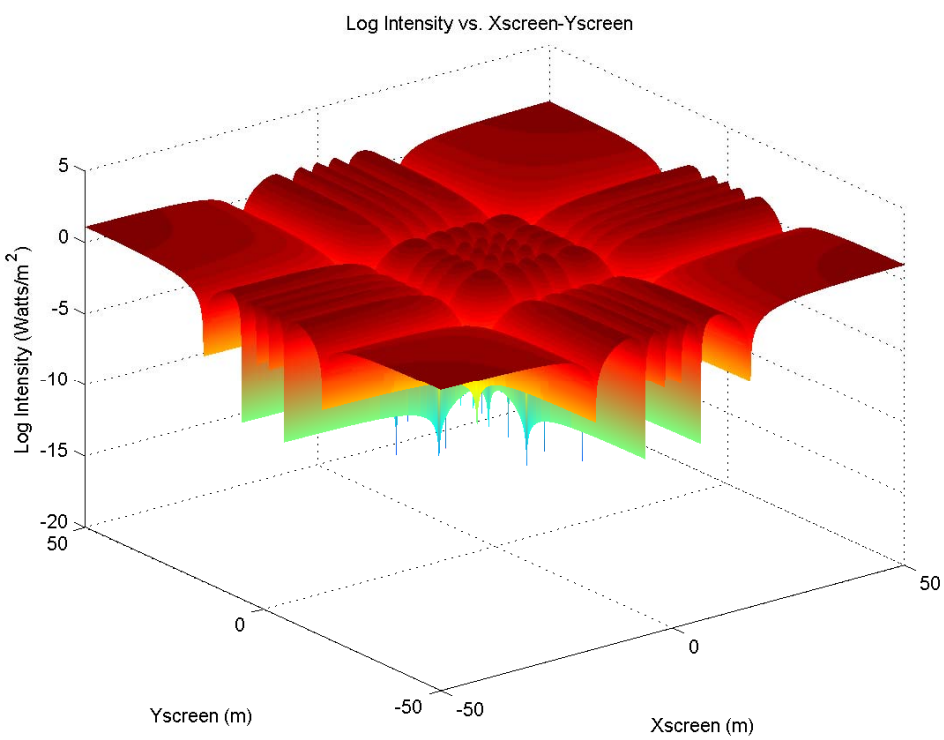
Plots of *SIL vs.* x_{screen} and *SIL vs.* y_{screen} for $(N_x, N_y) = (2,2)$ slits:



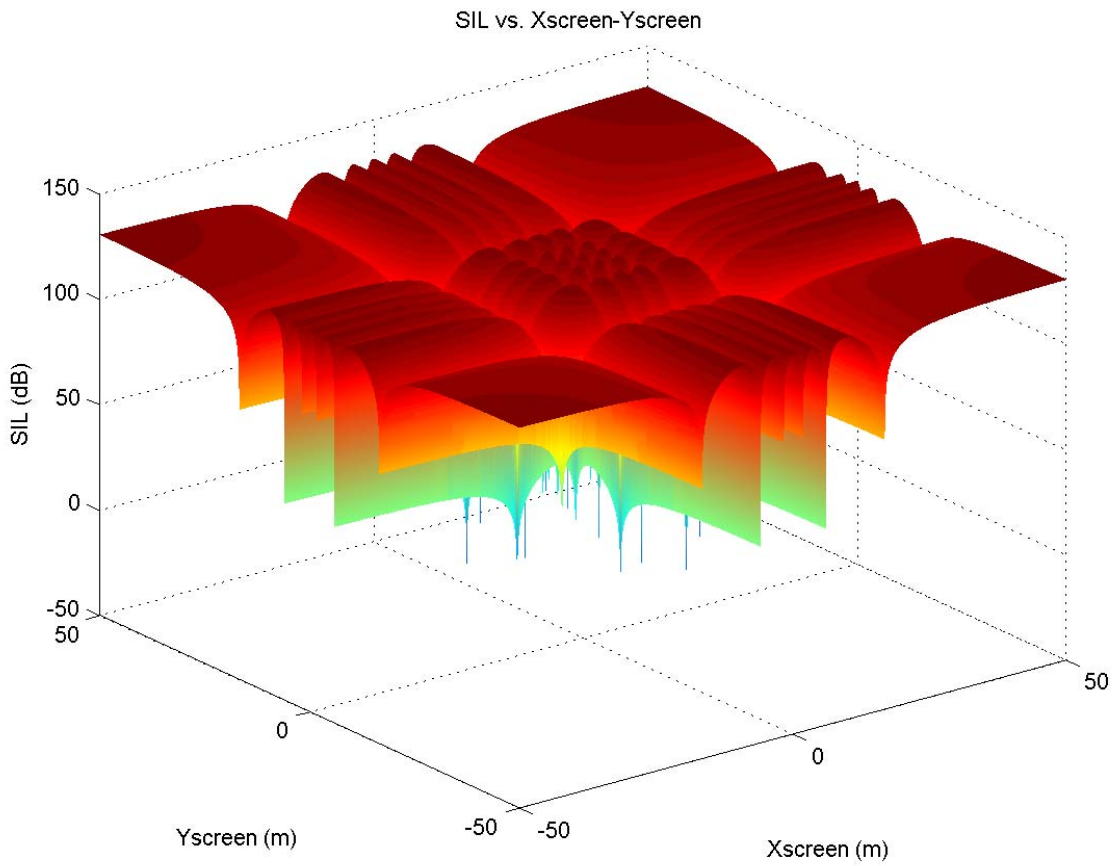


3-D Plot of $I_{tot}\left(x_{screen}, y_{screen}\right)$ *vs.* $\left(x_{screen}, y_{screen}\right)$ for (N_x, N_y) = (2,2) slits:

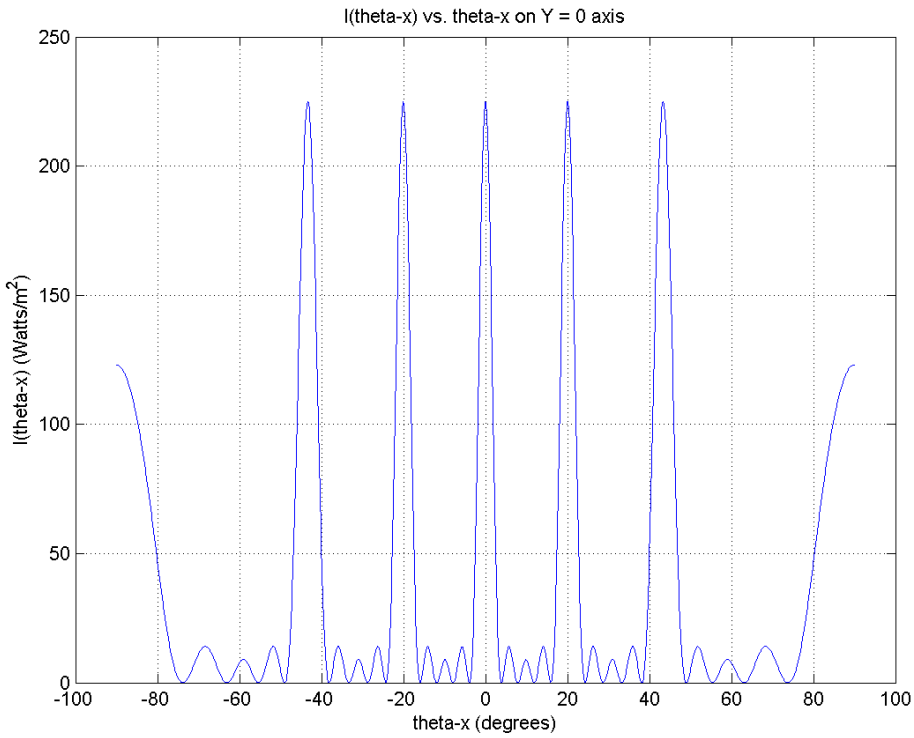


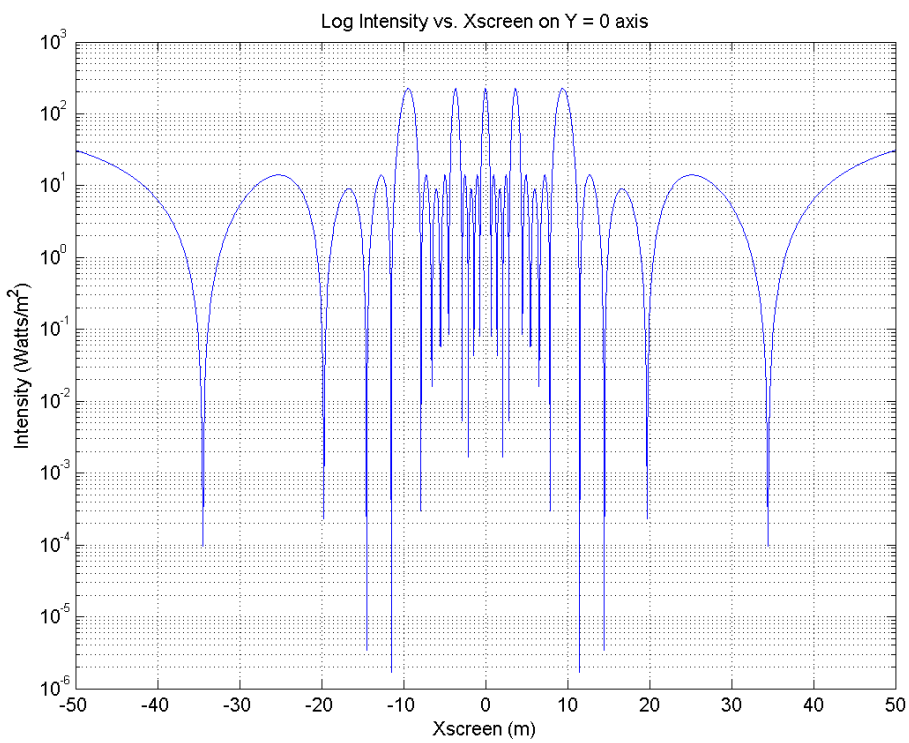


3-D Plot of $SIL\left(x_{screen}, y_{screen}\right)$ *vs.* $\left(x_{screen}, y_{screen}\right)$ for (N_x, N_y) = (2,2) slits:

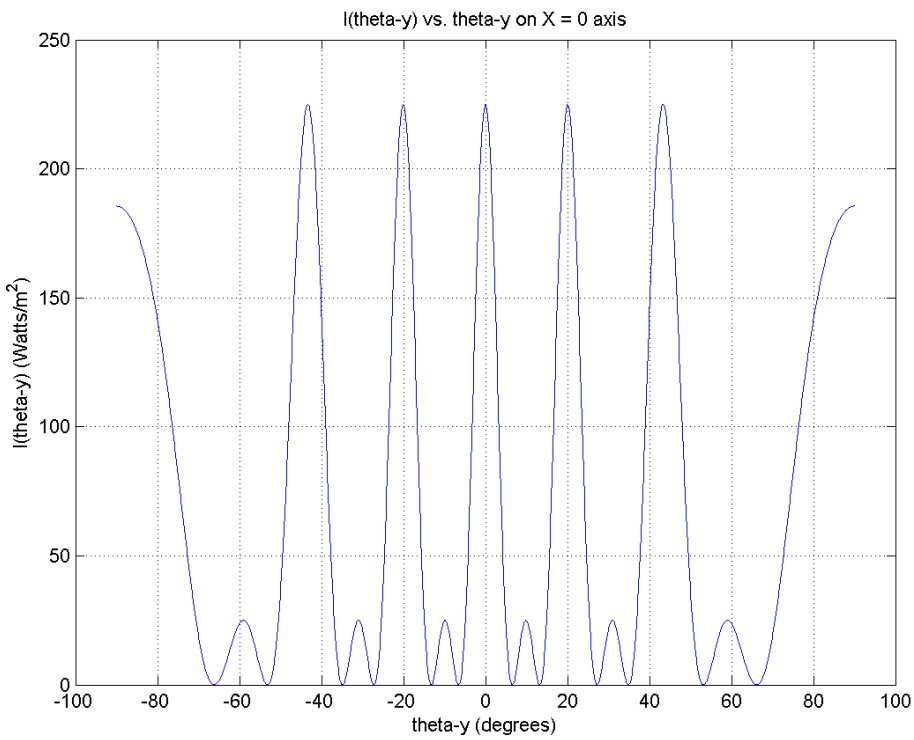


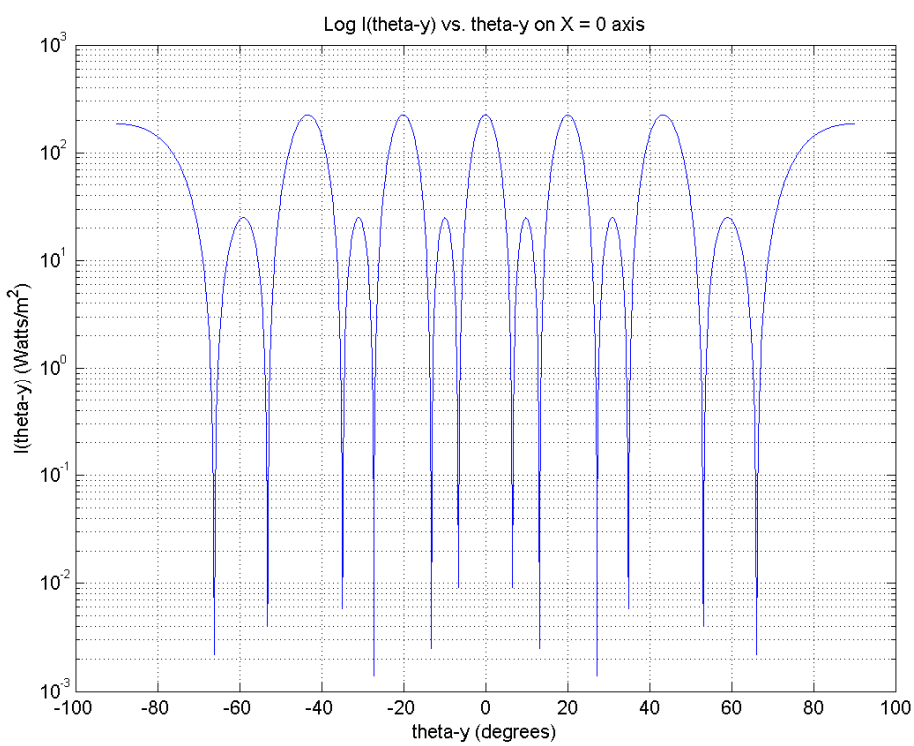
Plots of I_{tot} *vs.* θ_x for $(N_x, N_y) = (5,3)$ slits:



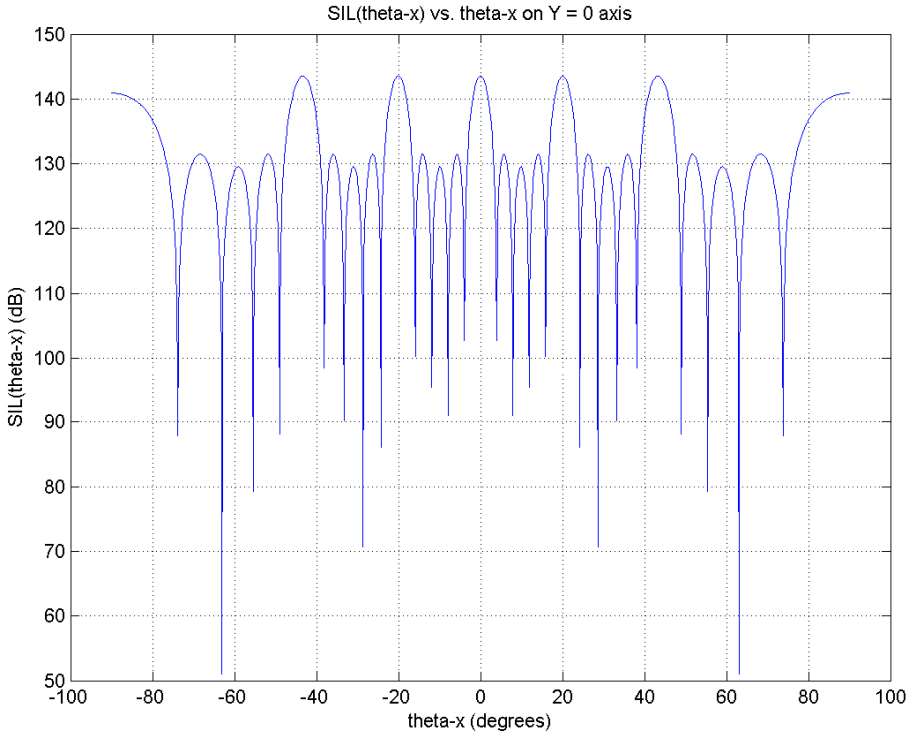


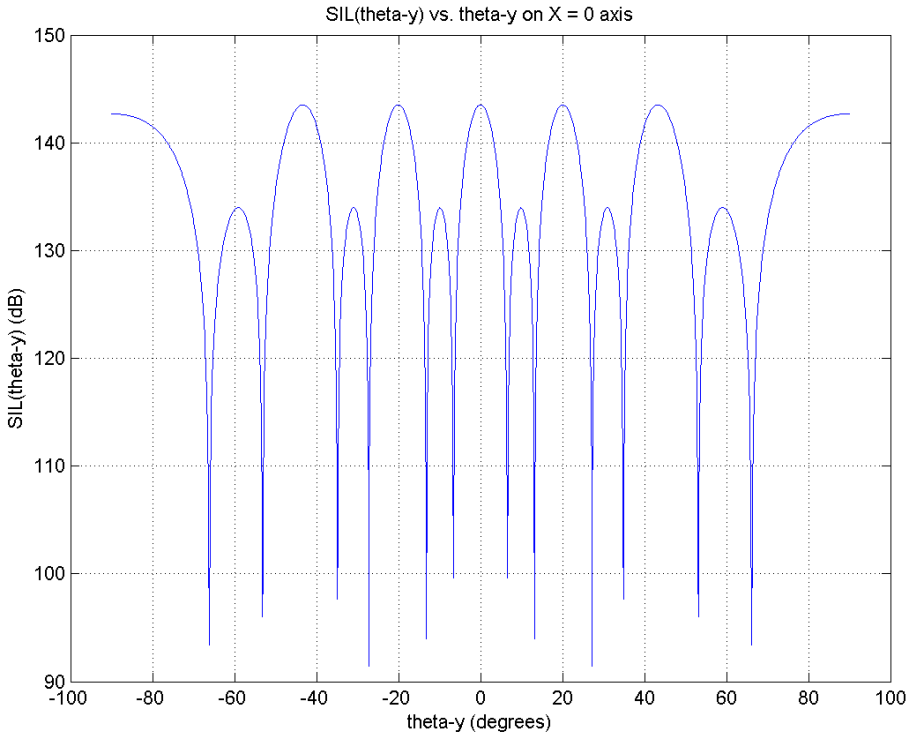
Plots of I_{tot} *vs.* θ_y for $(N_x, N_y) = (5,3)$ slits:



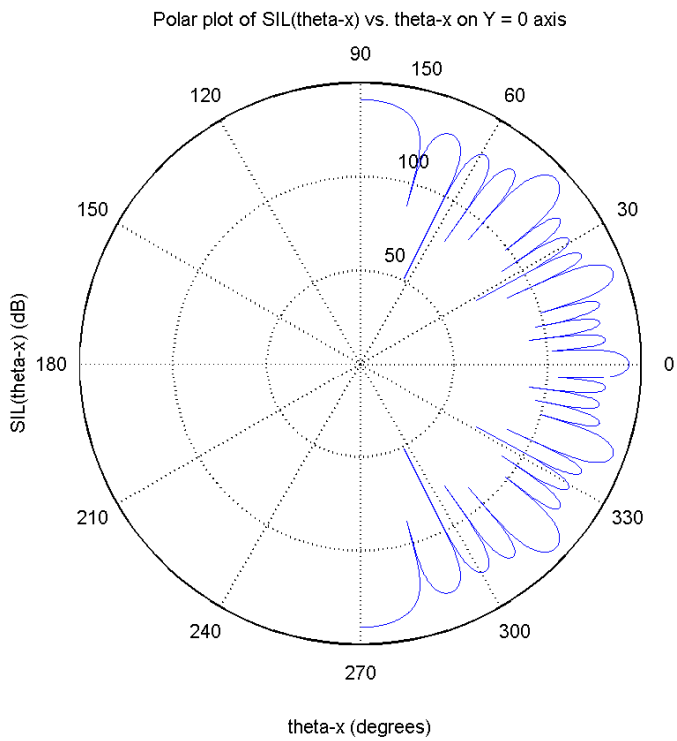


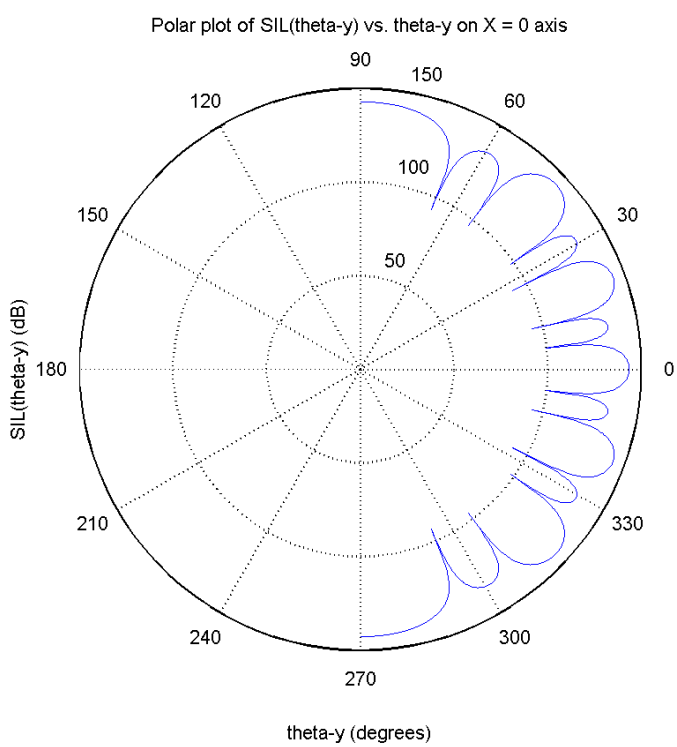
Plots of *SIL vs.* θ_x and *SIL vs.* θ_y for (N_x, N_y) = (5,3) slits:



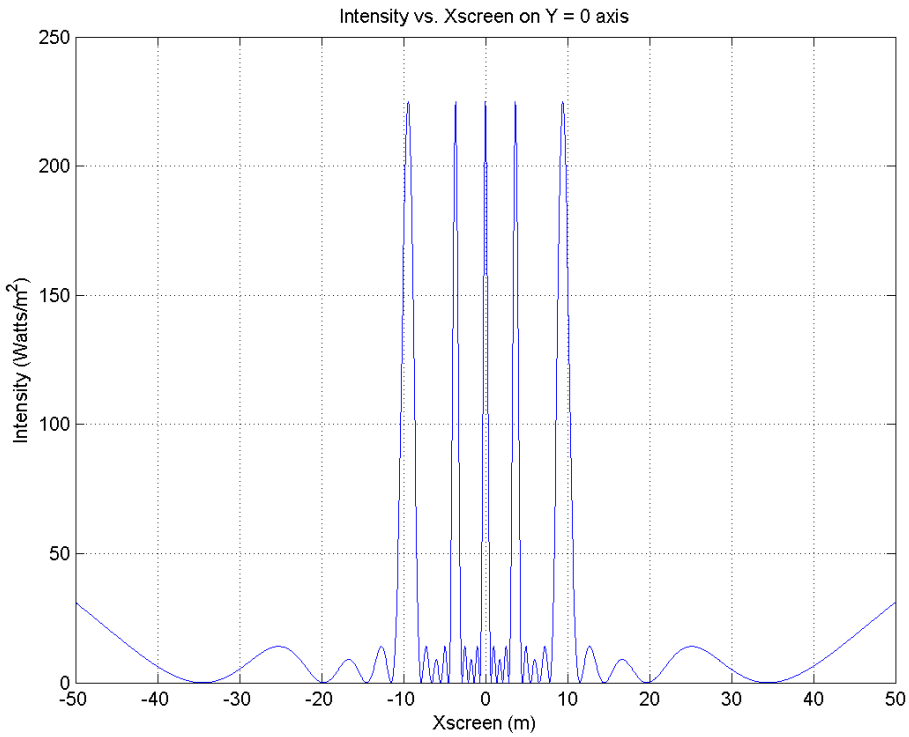


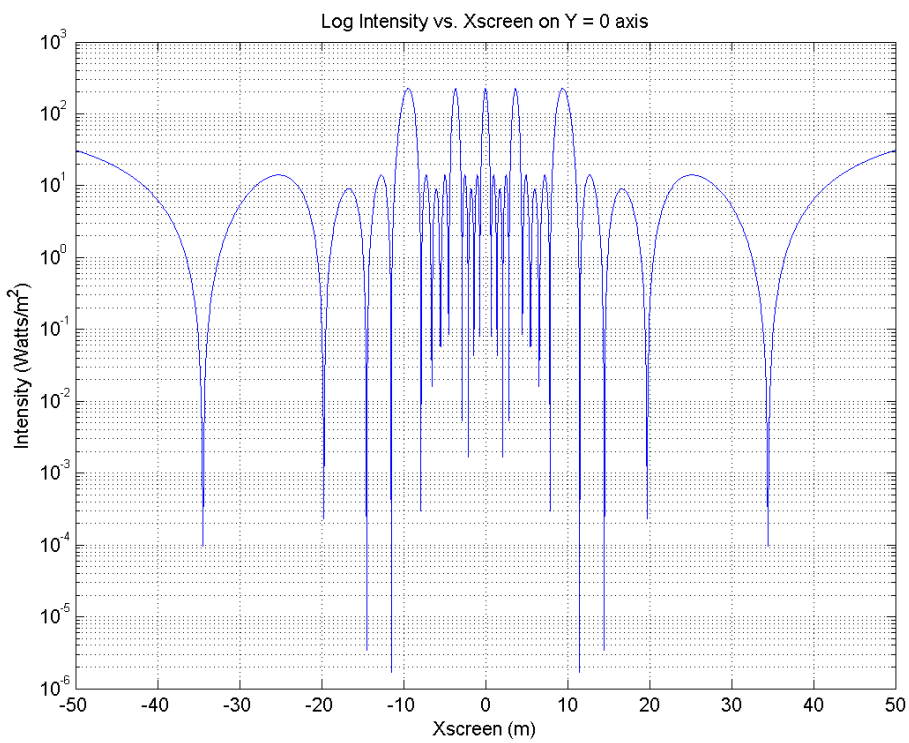
Polar plots of SIL *vs.* θ_x and SIL *vs.* θ_y for $(N_x, N_y) = (5,3)$ slits:



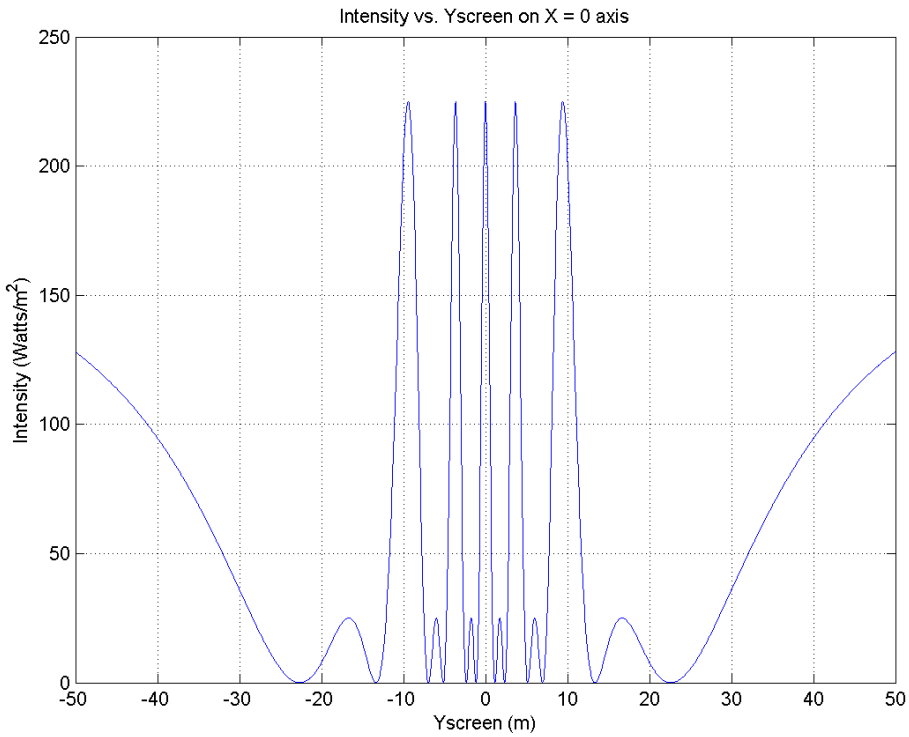


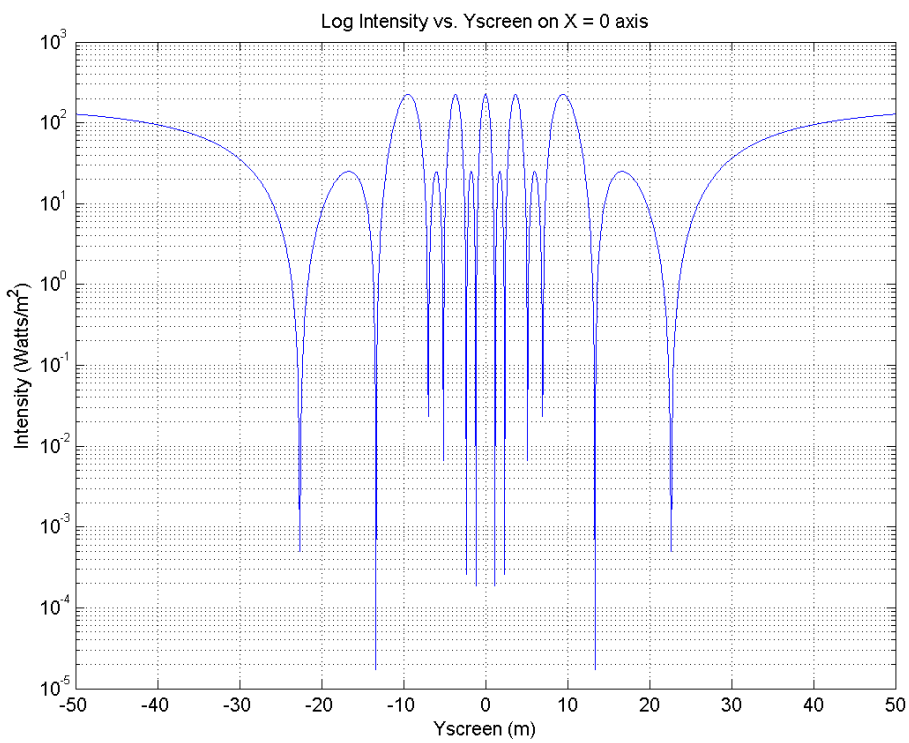
Plots of I_{tot} *vs.* x_{screen} for $(N_x, N_y) = (5,3)$ slits:



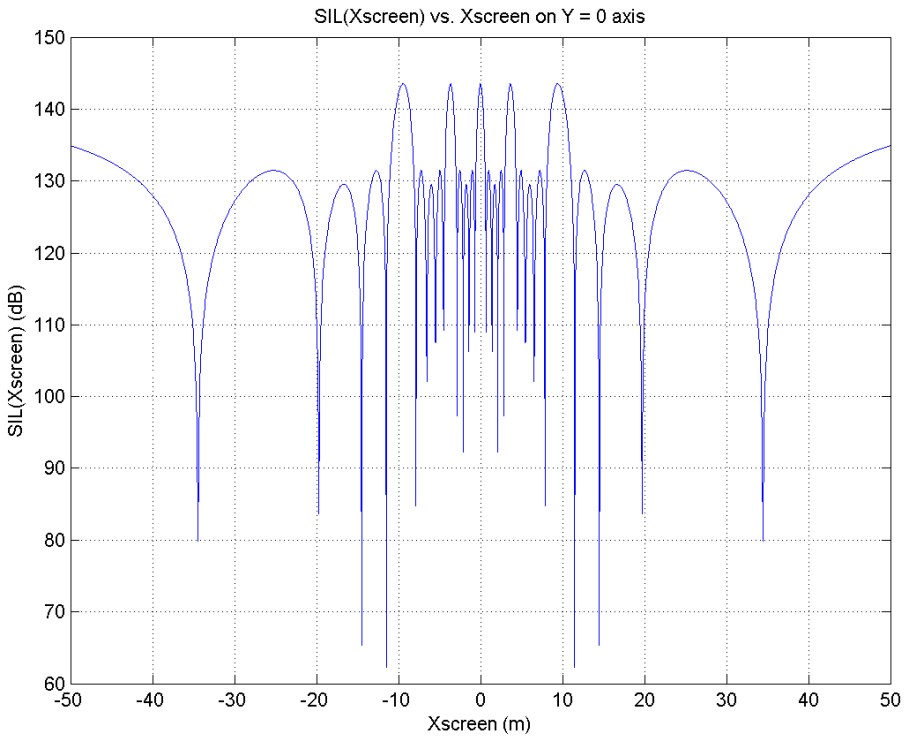


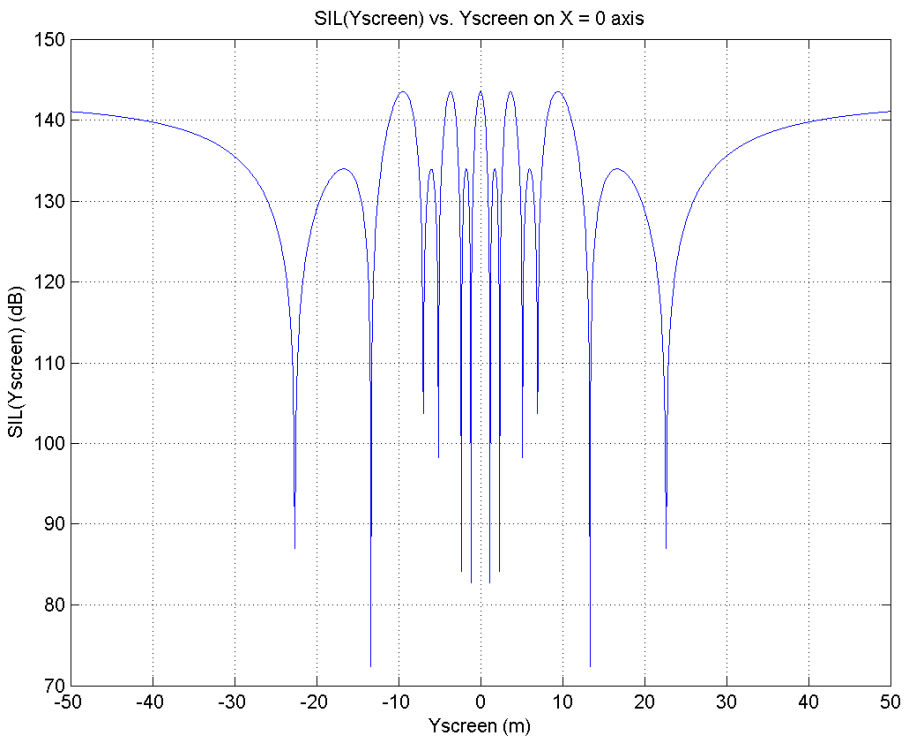
Plots of I_{tot} *vs.* y_{screen} for $(N_x, N_y) = (5,3)$ slits:



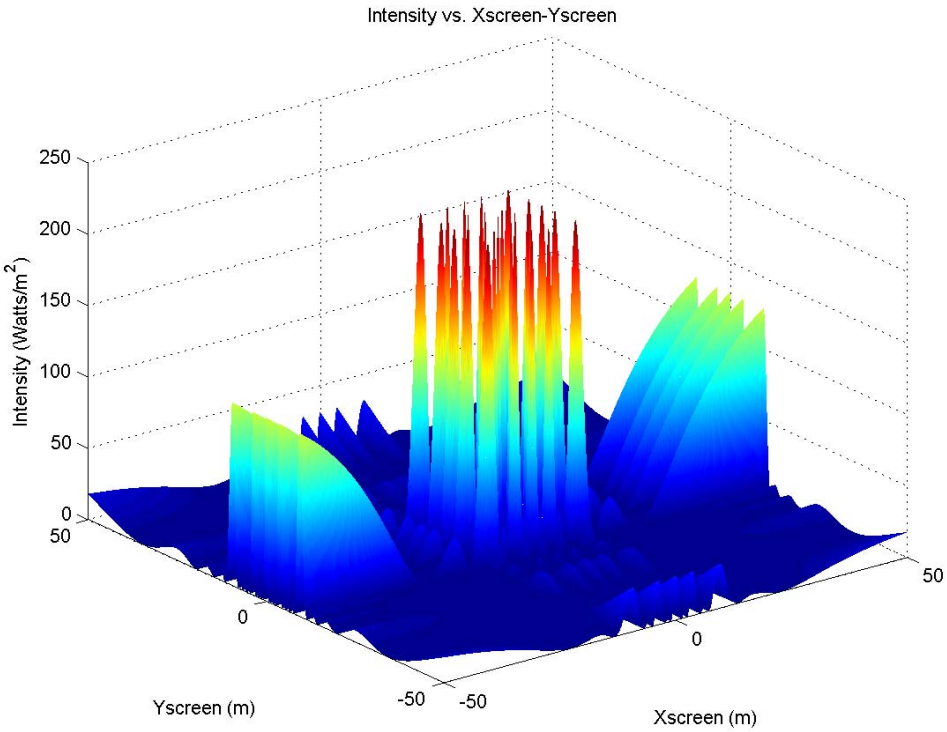


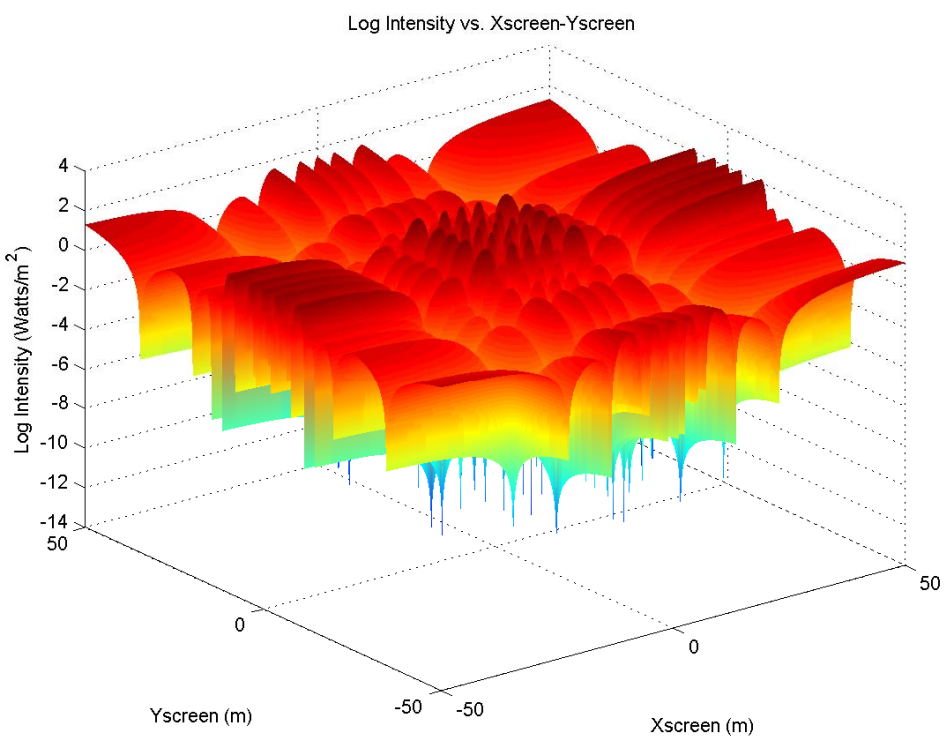
Plots of I_{tot} *vs.* y_{screen} for $(N_x, N_y) = (5,3)$ slits:



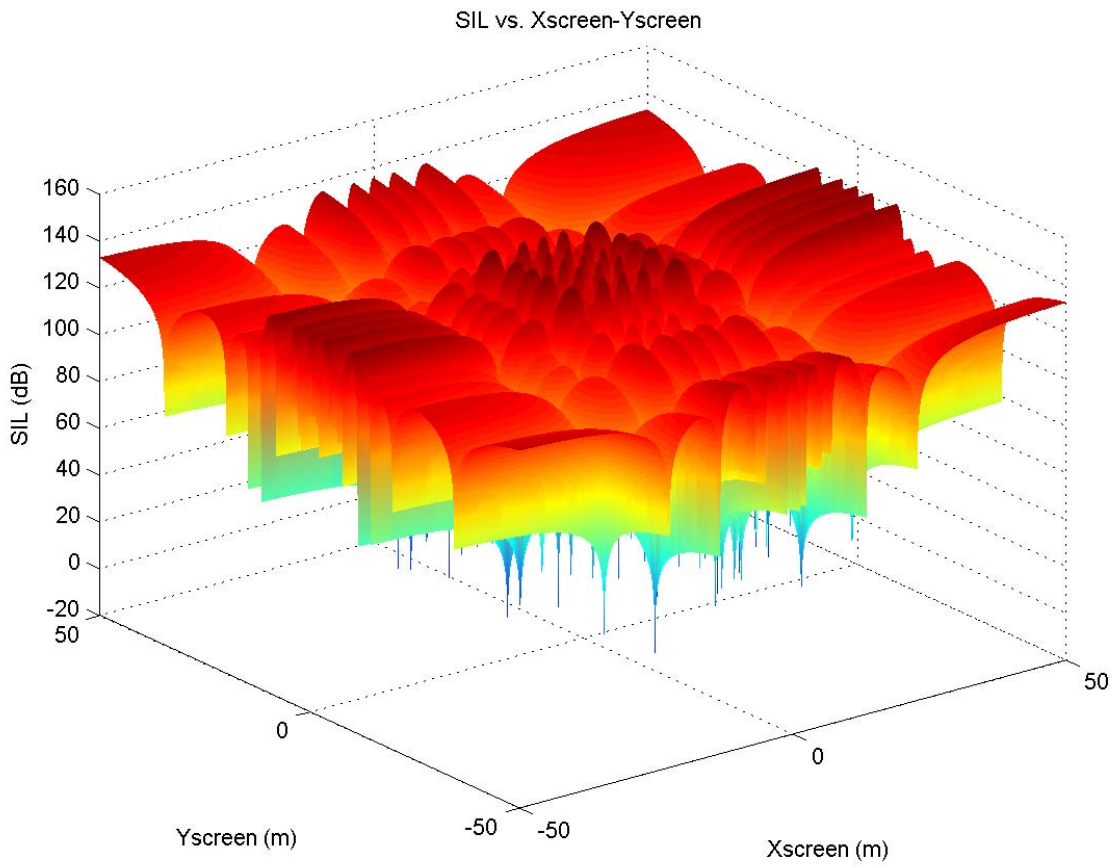


3-D Plot of $I_{tot}\left(x_{screen}, y_{screen}\right)$ *vs.* $\left(x_{screen}, y_{screen}\right)$ for (N_x, N_y) = (5,3) slits:

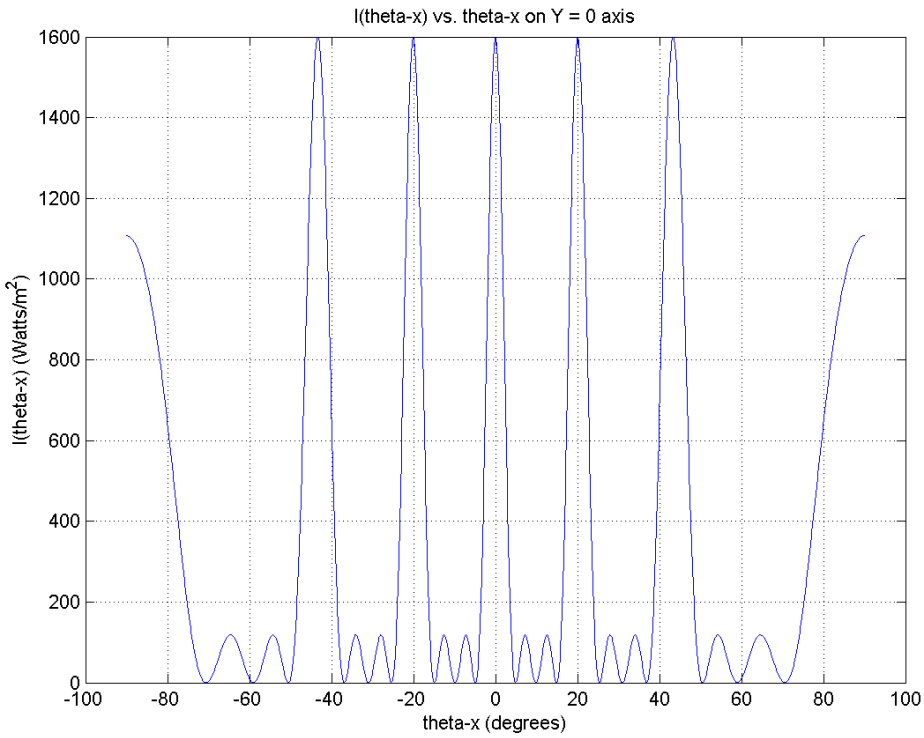


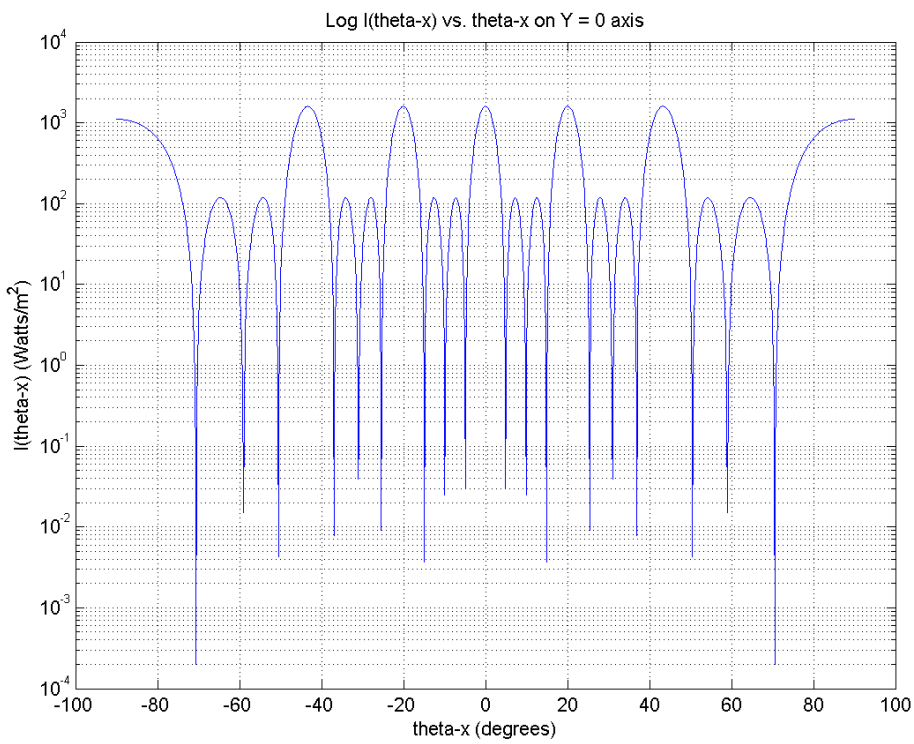


3-D Plot of $SIL\left(x_{screen}, y_{screen}\right)$ *vs.* $\left(x_{screen}, y_{screen}\right)$ for (N_x, N_y) = (5,3) slits:

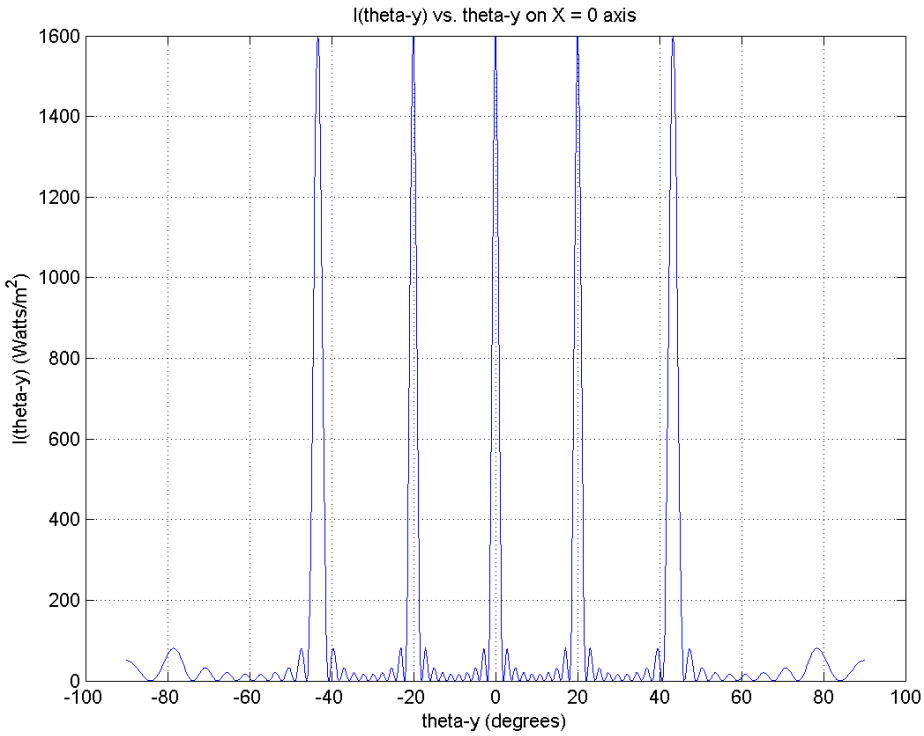


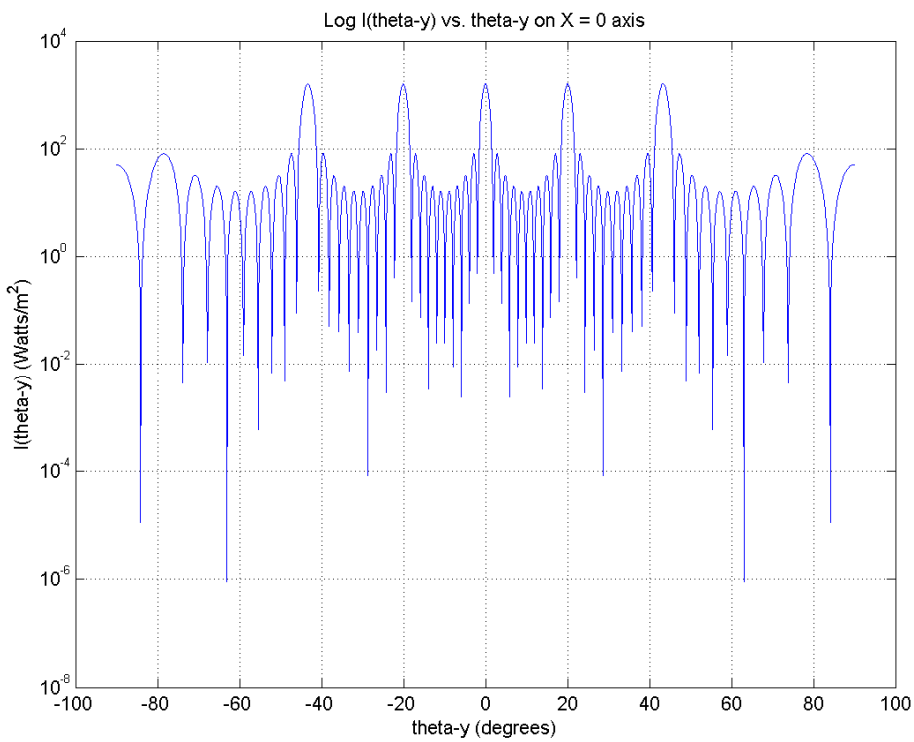
Plots of I_{tot} *vs.* θ_x for (N_x, N_y) = (4,10) slits:



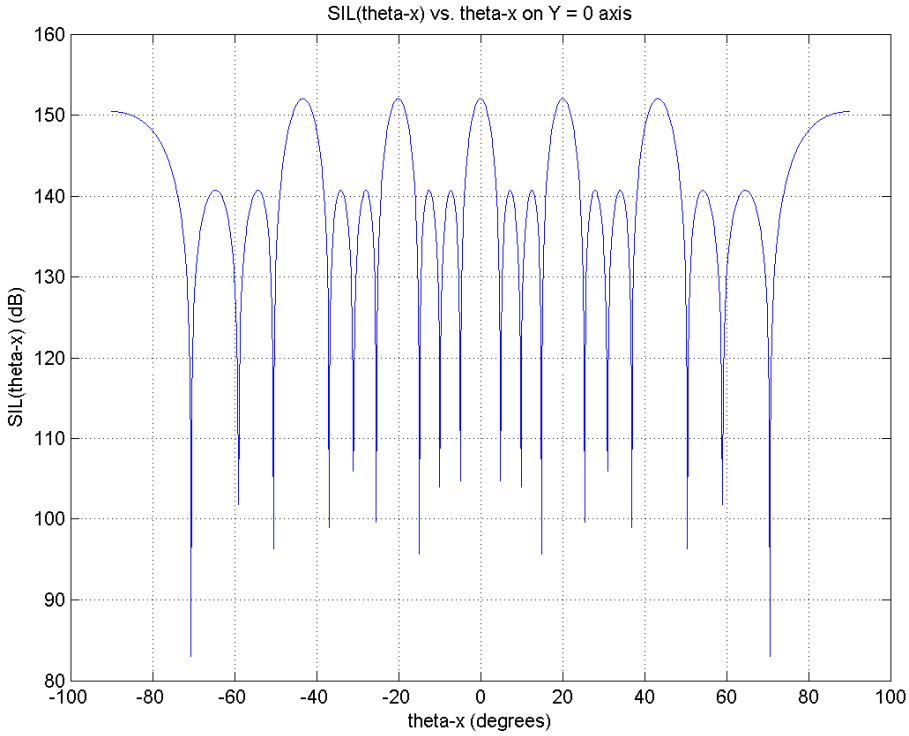


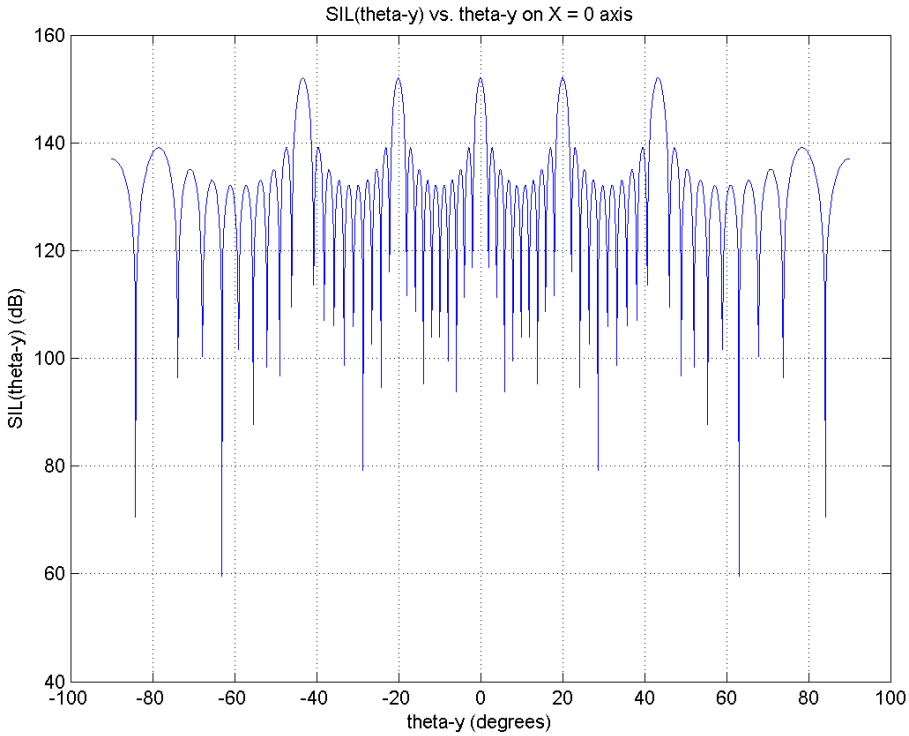
Plots of I_{tot} *vs.* θ_y for (N_x, N_y) = (4,10) slits:



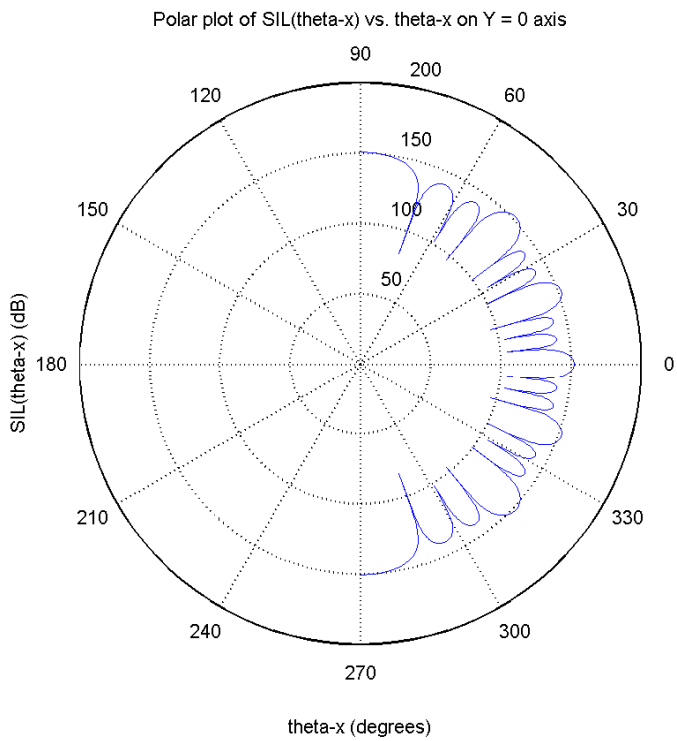


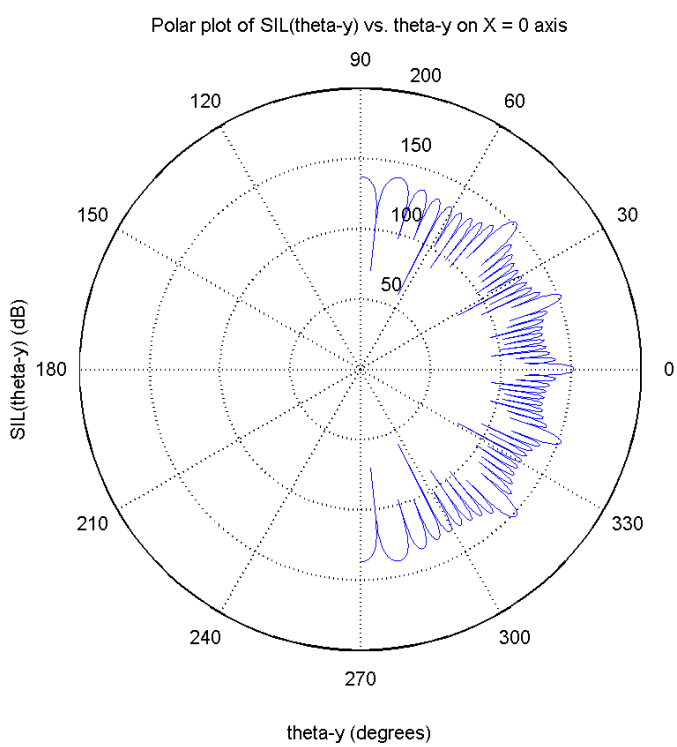
Plots of *SIL vs.* θ_x and *SIL vs.* θ_y for $(N_x, N_y) = (4,10)$ slits:



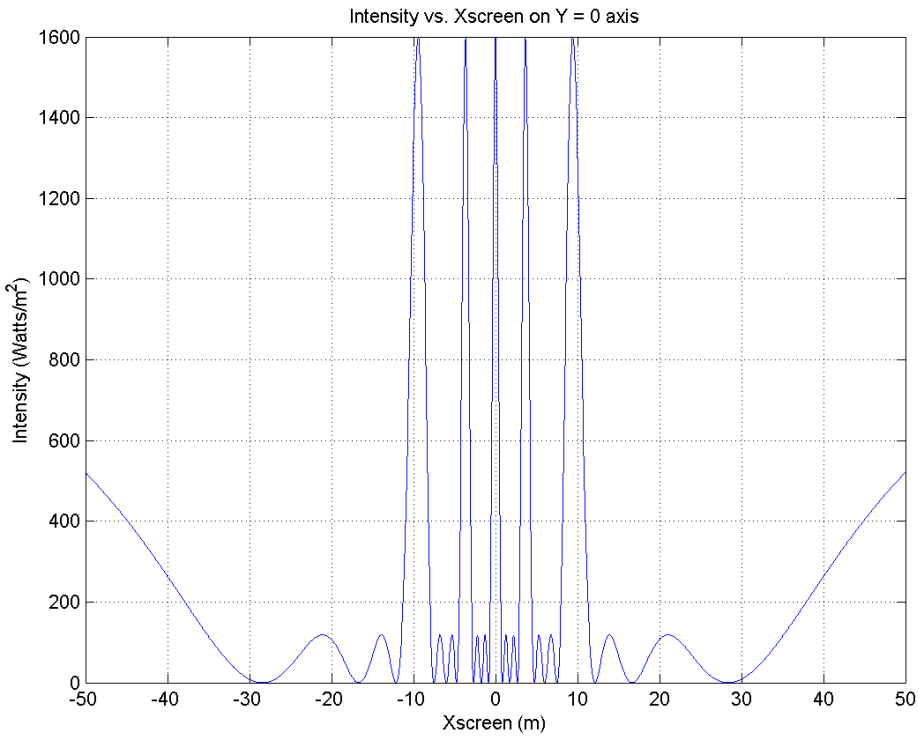


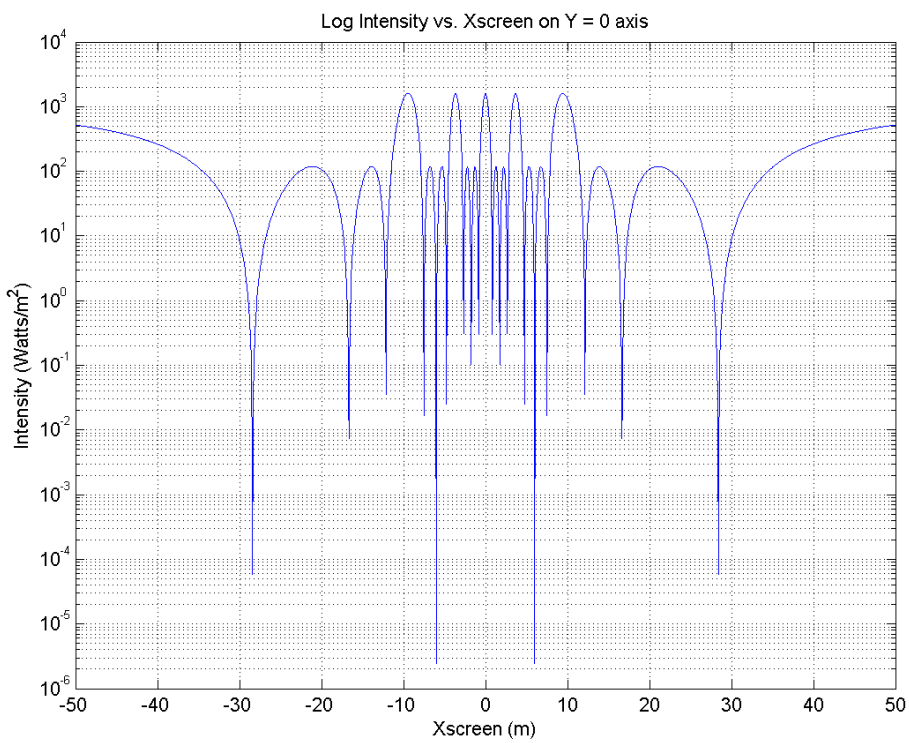
Polar plots of *SIL vs.* θ_x and *SIL vs.* θ_y for $(N_x, N_y) = (4,10)$ slits:



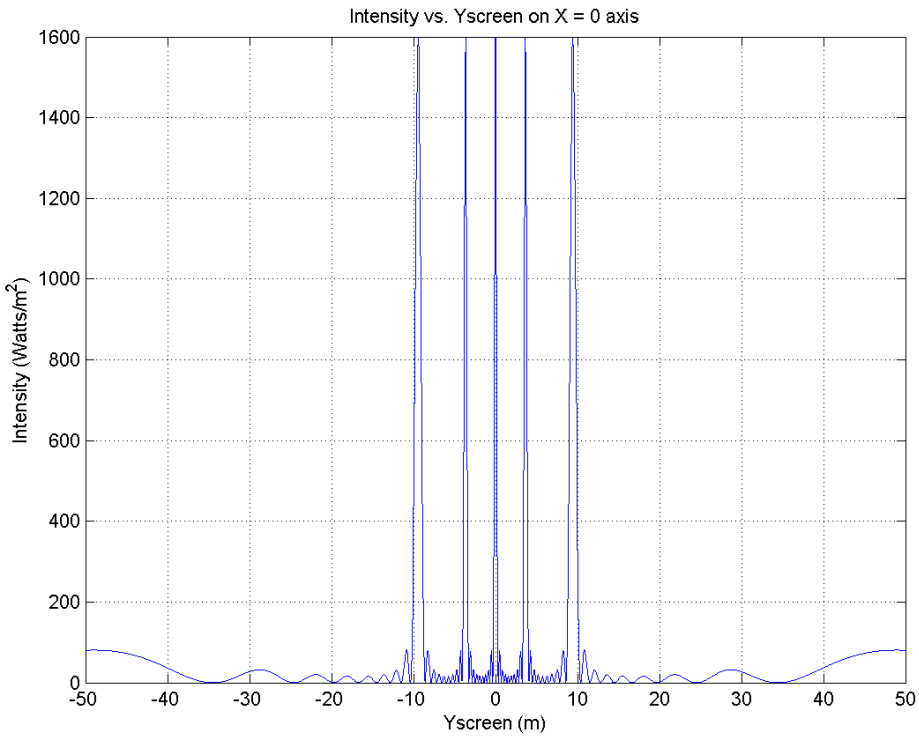


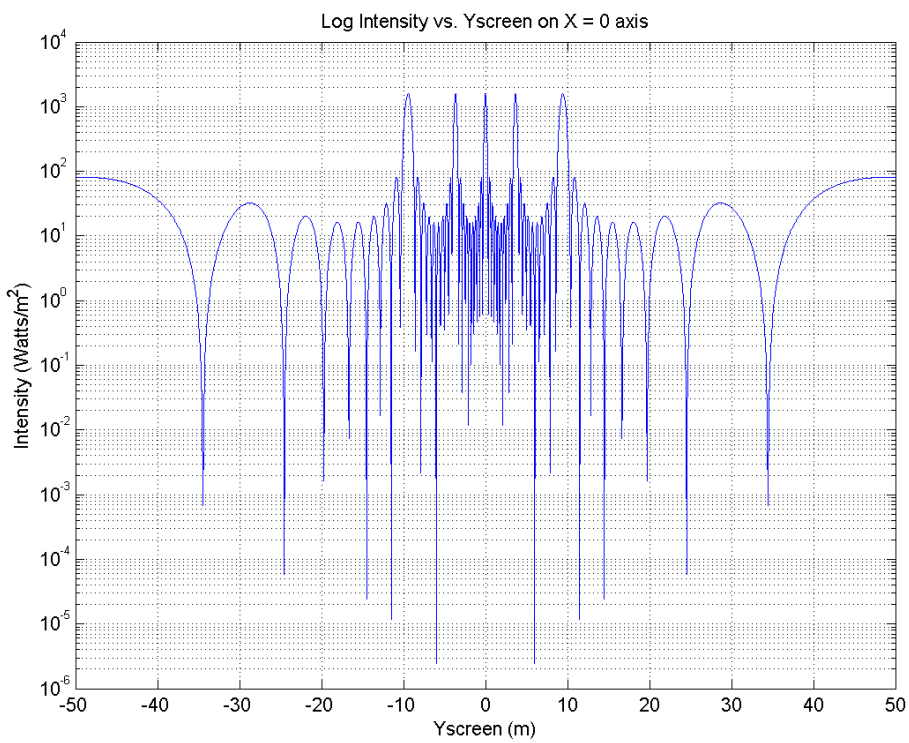
Plots of I_{tot} *vs.* x_{screen} and I_{tot} *vs.* y_{screen} for $(N_x, N_y) = (4,10)$ slits:



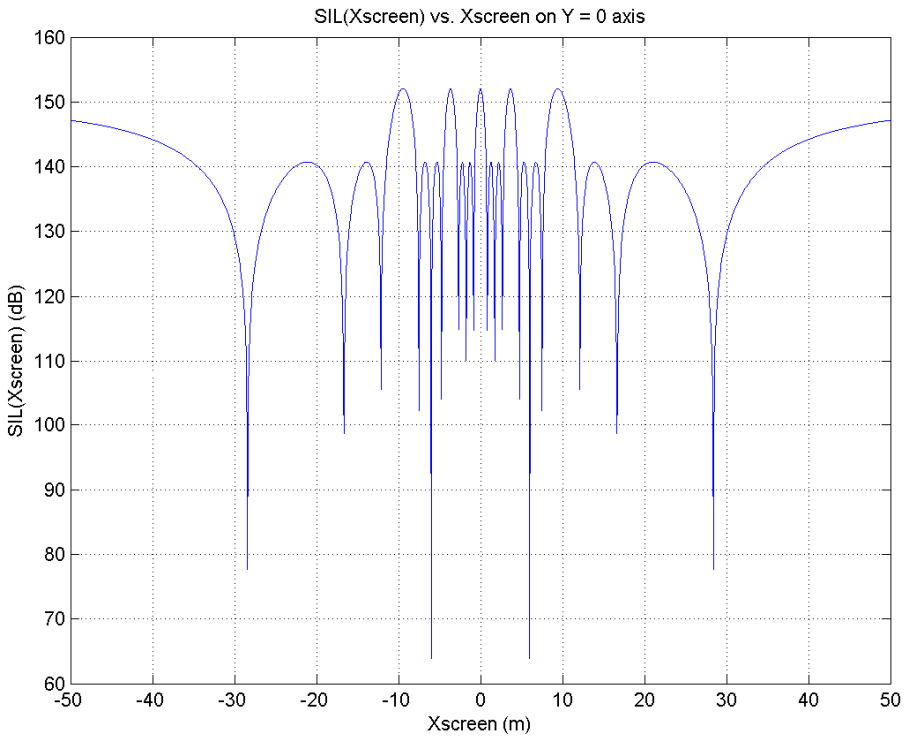


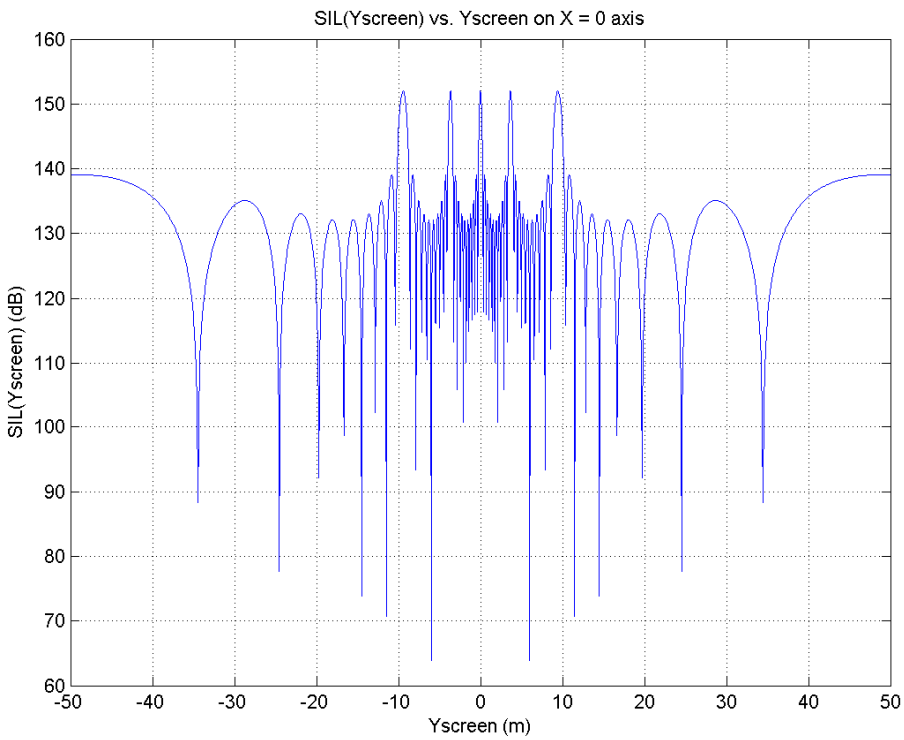
Plots of I_{tot} *vs.* y_{screen} for $(N_x, N_y) = (4,10)$ slits:



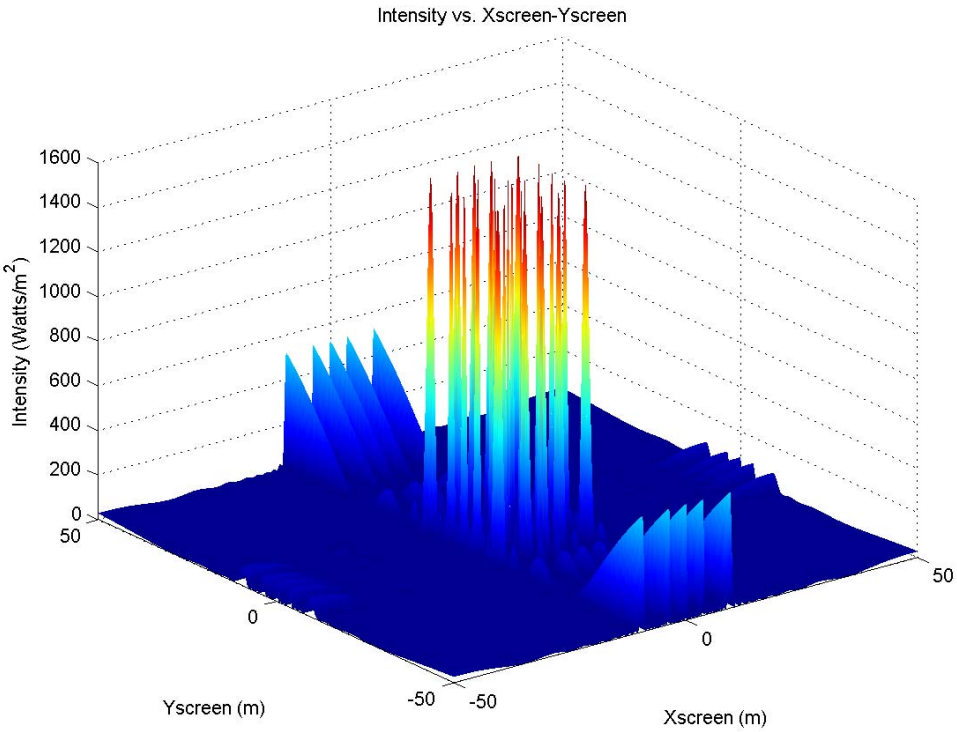


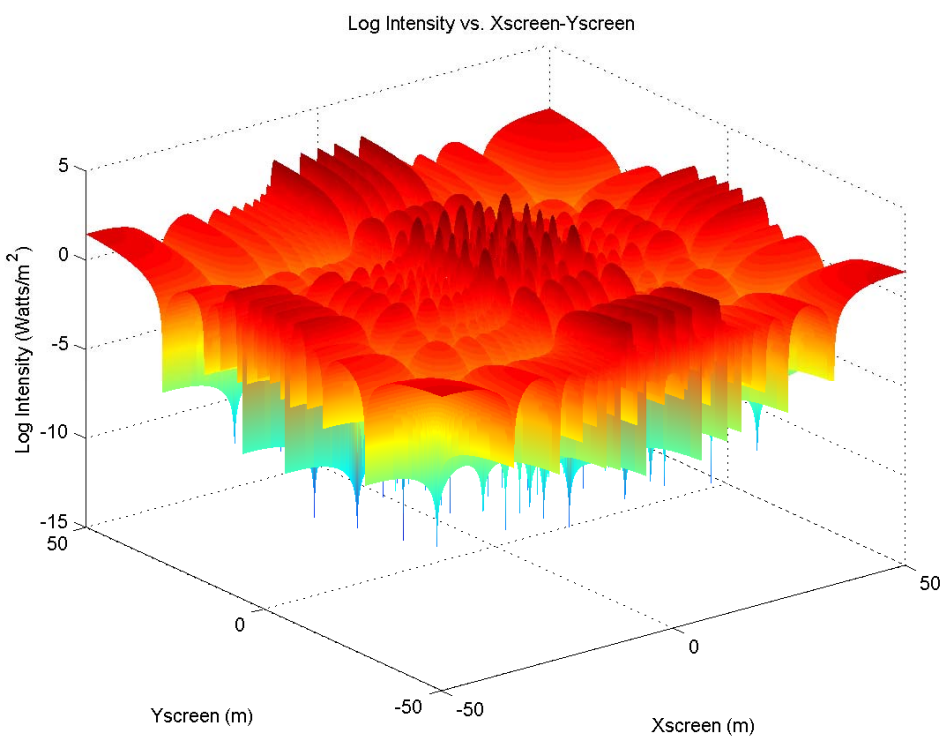
Plots of *SIL vs.* x_{screen} and *SIL vs.* y_{screen} for $(N_x, N_y) = (4,10)$ slits:



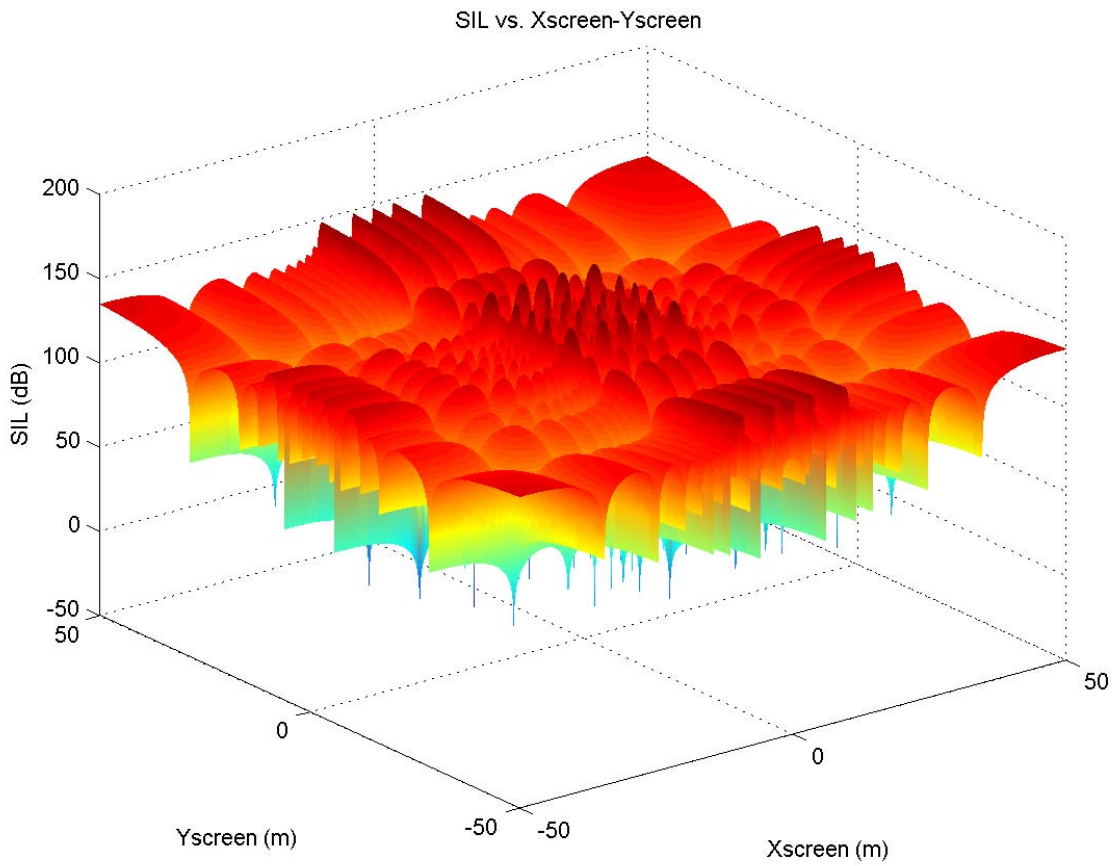


3-D Plot of $I_{tot}\left(x_{screen}, y_{screen}\right)$ *vs.* $\left(x_{screen}, y_{screen}\right)$ for $(N_x, N_y) = (4,10)$ slits:





3-D Plot of $SIL\left(x_{screen}, y_{screen}\right)$ *vs.* $\left(x_{screen}, y_{screen}\right)$ for (N_x, N_y) = (4,10) slits:



In the above figures, note that the angular width of the maxima decreases as the number of slits *N* increases. Note also that the number of maxima/minima increases linearly with increasing frequency *f*, since the phase difference increases linearly with frequency:

$$\delta_{x,y} = k\Delta L_{x,y} = \left(2\pi/\lambda\right)\Delta L_{x,y} = \left(2\pi f/v_{air}\right)\Delta L_{x,y} = \left(2\pi f/v_{air}\right) d\sin\theta_{x,y}\,.$$

Listing of the MATLAB code:

```
%==========================================================================
% Nslit_2D_Intf_Smpl_Thy.m
%
% 2D N-slit interference - simplest theory - far-field/plane-wave approx!
% sound waves assumed to be propagating in free air/great wide-open!
% each "slit" treated simplistically as point source - no spatial extent!
%
%==========================================================================
%
% Written by Prof. Steven Errede  Last Updated: Feb. 7, 2011 11:25 hr
%
%==========================================================================
close all;
clear all;

single     thtxr(1800);
single     thtxd(1800);
single     thtyr(1800);
single     thtyd(1800);

single    Itotx1(1800);
single    Itoty1(1800);
single    Itotxt(1800);
single    Itotyt(1800);
single     SILxt(1800);
single     SILyt(1800);

single      xscr(2000);
single      yscr(2000);

single    Itotx2(2000);
single    Itoty2(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 # of slits in x and y (n.b. both must be >= 2 !):
Nxslits =  2; % 2; 5;  4;
Nyslits =  2; % 2; 3; 10;

% 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
Dsrc   =    1.0;    % transv. distance between slits     (m) n.b.   Dsrc << Lobs

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

     deltax  = (2.0*pi/lambda)*Dsrc*sin(Thetar); % resultant Nslit phase diff (radians)

   Itotx1(i) = Io*(sin(Nxslits*deltax/2.0)/sin(deltax/2.0))^2;
   Itoty1(i) = Io*(Nyslits)^2;
```

```
   Itotxt(i) =  Itotx1(i)*Itoty1(i)/Io;  % total intensity (Watts/m^2)
    SILxt(i) = 10.0*log10(Itotxt(i)/Ir); % Sound Intensity Level (dB)

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

Thetad =  -90.0;    % angle theta of observer in degrees
dTheta =    0.1;    %  step angle in degrees
for i = 1:1800;
    thtyd(i) = Thetad;            % angle theta of observer in degrees
    Thetar   = (pi/180.0)*Thetad; % angle theta of observer in radians
    thtyr(i) = Thetar;

     deltay  = (2.0*pi/lambda)*Dsrc*sin(Thetar); % resultant Nslit phase diff (radians)

   Itotx1(i) = Io*(Nxslits)^2;
   Itoty1(i) = Io*(sin(Nyslits*deltay/2.0)/sin(deltay/2.0))^2; % total intensity (Watts/m^2)
   Itotyt(i) =  Itotx1(i)*Itoty1(i)/Io;  % total 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('Log 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('Log 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 Itot vs. x,y-screen:
%=====================================
 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)
     Thetar  = atan(x/Lobs); % angle theta of observer in radians

     deltax  = (2.0*pi/lambda)*Dsrc*sin(Thetar); % resultant Nslit phase diff (radians)

   Itotx2(i) = Io*(sin(Nxslits*deltax/2.0)/sin(deltax/2.0))^2; % total intensity (Watts/m^2)
   Itoty2(i) = Io*(Nyslits)^2;
   Itotxs(i) =  Itotx2(i)*Itoty2(i)/Io;  % total 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)
     Thetar  = atan(y/Lobs); % angle theta of observer in radians

     deltay  = (2.0*pi/lambda)*Dsrc*sin(Thetar); % resultant Nslit phase diff (radians)

   Itotx2(i) = Io*(Nxslits)^2;
   Itoty2(i) = Io*(sin(Nyslits*deltay/2.0)/sin(deltay/2.0))^2; % total intensity (Watts/m^2)
   Itotys(i) =  Itotx2(i)*Itoty2(i)/Io;  % total 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('Log 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('Log 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;            % position of observer on perp. screen (m)
     Thetar  = atan(x/Lobs); % angle theta of observer in radians

     deltax  = (2.0*pi/lambda)*Dsrc*sin(Thetar); % resultant Nslit phase diff (radians)

   Itotx2(j) = Io*(sin(Nxslits*deltax/2.0)/sin(deltax/2.0))^2; % total 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;            % position of observer on perp. screen (m)
         Thetar  = atan(y/Lobs); % angle theta of observer in radians

         deltay  = (2.0*pi/lambda)*Dsrc*sin(Thetar); % resultant Nslit phase diff (radians)

     Itoty2(i)   = Io*(sin(Nyslits*deltay/2.0)/sin(deltay/2.0))^2; % total intensity (Watts/m^2)

     Itotxy(j,i) =    Itotx2(j)*Itoty2(i)/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 2-D Nslit interference simple thy calculation completed !!! \n')
%=========================================================================
```