RCWT Calculator Manual

In this manual, rigorous coupled-wave theory (RCWT) is explained. Firstly, the background and theory of RCWT is briefly explained. Then applications of RCWT are discussed, as well as the related problems when using RCWT. After understanding RCWT, a short description of the RCWT calculator in OptiScan is given with the brief descriptions of the parameters setting and geometries design. At last, an example for calculating the diffraction efficiency of dielectric gratings is given by using RCWT calculator. These results are compared with the published data, and they show great agreement. The descriptions of the theory of RCWT and RCWT calculator are brief, and interested users should refer the former detailed description of RCWT calculator associated with OptiScan. [1, 2]

1. Background of RCWT [1]

Rigorous coupled-wave theory (RCWT) is an exact, rigorous solution to Maxwell's equations for a plane wave incident on a physical, possibly many layered structure that is infinite in extend and periodic in one dimension. [3,4,5]

The basic idea behind RCWT is that a complex structure is divided into several layers, and the permittivity variation of each layer is expressed a Fourier series. Furthermore, the electric fields and magnetic fields, above, below and within each layer, are also decomposed as a Fourier series of plane waves. The base frequency of each of these plane waves is determined by applying the Floquet condition (also called grating equation) given by

$$k_{xj} = k_0 \left[n_I sin\theta - j \frac{\lambda_0}{\Lambda} \right]$$

Where,

n_I is the complex index of the incident medium

θ is the angle of the incident plane wave related to surface normal

Λ is the period of the permittivity variation

j is an indexing integer (note: not the imaginary number)

k_0 is wave vector of the incident light in vacuum

This condition forces each layer to have the same base period. However the Since electric and magnetic fields, and permittivity can be expressed in Fourier series, when applying Maxwell's equations, the differential equations can be solved by using the eigenvalue and eigenvector approach. Once the Maxwell's equations have been solved for each layer, the boundary conditions are applied to get the exact solution. When applying the boundary conditions, a set of algebraic coupled linear equations is resulted, which can be solved normal matrix technique. However, a direct inversion of these linear equations can result in numerical stability problems. [4]

2. Applications of RCWT [1]

RCWT can accurately solve the Maxwell's equations for a linear medium in which the permittivity variation can be expressed as a multiply layered binary structure. It should be pointed out that RCWT is best suited for binary grating structure. However, other common grating structures, such as cosine grating, blazed grating, can be approximated by layering multiple binary structures, which requires longer calculation time. And also, metal layers can be simulated in RCWT, and this requires more orders of Fourier series such that evanescent waves can be accurately calculated.

RCWT was originally developed to calculate the diffraction efficiency of binary grating structure. If multiple layers are used, diffraction efficiency for other common grating structure can be calculated. Metal grating can also be calculated for the diffraction efficiency if enough orders are defined.

RCWT can also be used to directly calculate the electric fields reflected from and transmitted through the structure. This capability extends the applicability of RCWT to calculate the effect of a grating structure on an arbitrary electric field. The electric field incident on the surface of the grating structure is decomposed into an angular spectrum of plane waves for each of the standard x, y and z polarization states. The incident angular spectrum is transformed into s-type and p-type polarization states. The RCWT algorithm is then applied to the s and p polarization components of each incident plane wave. The results of each of these calculations are a spectrum of s-type and p-type wavefronts. The s-type and p-type spectra are recombined and transformed to x, y, and z polarization states. The transformed electric field is then calculated by and inverse Fourier transform of the resultant spectra.

3. Problems when using RCWT [1]

Two factors determine the accuracy of the RCWT algorithm, the number of orders kept in the truncated series representing the electric field (and the media permittivity), and binary layering used to simulate the structure. In the limit of an infinite number of orders and layers, RCWT results in an exact solution to Maxwell's equations. However, the number of orders required to achieve an accurate solution differs depending on the problem in question. The minimum number of orders required for any configuration is $2\Lambda/_{\lambda}$, where Λ is the base period of the structure. Therefore, large period structures, require a large number of orders and therefore, longer computation times. In general, for dielectric media, the minimum number of orders is often sufficient. For metal structures, a large number of evanescent orders are required to achieve an accurate result. RCWT is known to handle s-type (TE) polarization much better than p-type (TM) polarization, which generally requires a larger number of orders.

There is also an inherent instability in the basic RCWT algorithm**.** This instability occurs when using a non-periodic layer (such as an air-gap), if the base period times the index of the layer is exactly equal to an integer times the wavelength. This problem occurs because the algorithm attempts to invert a matrix of the form, which is singular. As long as physical values for refractive index are used this instability is unlikely to be a problem, except in the case of an air gap. When this situation occurs using a value of 1.0001 for the index of air will result in a stable solution.

4. RCWT calculator in OptiScan [1, 2]

Nearly every simulation will begin by defining the physical structure of the diffraction grating being analyzed. The parameters in the grating structure are completely up to the user. That is, any structure that can be constructed using a stack of binary layers can be simulated using the Optiscan RCWT calculator. Thin film stacks can also be included simply by setting the two indices in a given layer to the same value. However, it should be kept in mind that the minimum number of orders used in the RCWT algorithm is $^{2\Lambda}/_{\lambda}$. Therefore, it must be emphasized that large period grating structures will require a large number of orders and will likewise require a large calculation time. Also note that a large number of layers will also increase the calculation time.

To define a grating:

4.1) In the main RCWT calculator panel, click "RCWT settings" to open the RCWT settings window.

4.2) Input the base period of the grating, keeping in mind that the minimum number of orders required for a calculation is $^{2\Lambda}/_{\lambda}$.

4.3) Input the complex index of refraction for the incident media and the substrate. If you plan on performing wavelength dependent calculations, you will likely want to use a dispersion file. The dispersion file is a tab delimited ASCII text file in which the first column is wavelength in microns, the second column is the real part of the complex index and the third column is the imaginary part of the complex index. The format for the dispersion file is:

```
1.23450000000000	1.21200000000000	12.46400000000000
1.12227272727273	1.20100000000000	11.18100000000000
1.02875000000000	1.26000000000000	10.01000000000000
0.94961538461538	1.46800000000000	8.94900000000000
0.88178571428571	2.23700000000000	8.21200000000000
```

Optiscan will automatically interpolate between the discrete values given in the data file.

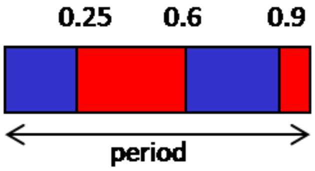
4.4) Click on the "RCWT Layers" Tab to begin defining the layers of your grating.

4.5) Use the "New Layer" button to add as many layers is required for your desired grating structure.

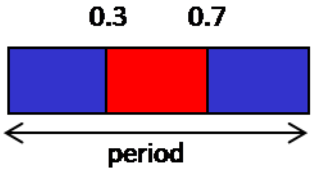
4.6) The height is the thickness of the layer.

4.7) Input the two index values that define the modulation of your grating structure. If desired, you can use a dispersion file for each index. If you want to use an unmodulated layer, you simply set Index 1 and Index2 to the same value.

4.8) Define the duty cycle vector for the layer. The values in the duty cycle vector are sequential values between zero and one that define the transition points between Index 1 and Index 2. For example (the last number is always 1)



Duty cycle vector = [0.25 0.6 0.9 1]



Duty cycle vector = [0.3 0.7 1]

4.9) View the grating structure using the preview button. A grating structure for cosine grating is given in Fig 4-1. Five layers are used for this structure, with height, indices and duty cycle vector given in the following table. Please keep in mind the last number in duty cycle vector is always 1.

Example: A 1-micron deep cosine grating with a 1-micron period

Layer number	Height	Index 1	Index 2	Duty cycle vector
1	0.2	1.0	Aluminum.dat	0.39758 0.60242 1
2	0.2	1.0	Aluminum.dat	0.31549 0.68451 1
3	0.2	1.0	Aluminum.dat	0.25 0.75 1
4	0.2	1.0	Aluminum.dat	0.18451 0.81549 1
5	0.2	1.0	Aluminum.dat	0.10242 0.89758 1

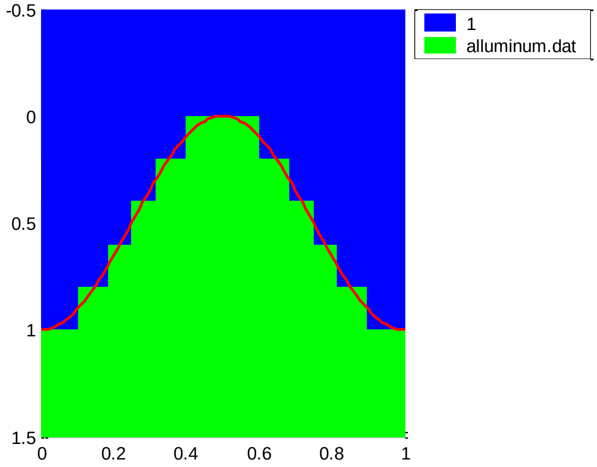


Fig 4-1. RCWT geometry for a cosine grating with five layers

The number of orders required to achieve an accurate result from the RCWT calculator is dependent on three factors.

1. The period of the grating: The minimum number of orders is $^{2\Lambda}/_{\lambda}$.
2. The material being used: For dielectric materials, the minimum number of orders is generally sufficient, although it is often a good idea to include at least a few orders above the minimum. For metals, more orders are required, especially with P-type polarization, to account for any plasmon and surface wave effects. Generally with metal structures, a good place to start is twice the minimum number of orders. For materials in which the imaginary part of the complex index is high (above 1) even more orders will likely be required.
3. Number of layers: Generally structures with several layers require a few more orders than single surface structures.

Obviously the user wants to minimize the number of orders required to produce an accurate result. The best way to determine the correct number of orders is to perform a diffraction efficiency calculation for p-type polarization and increase the number of orders until the answer stops changing.

Once the grating structure has been defined, the transmitted and reflected electric fields can be easily calculated in terms of the four parameters in the main window. Lambda is the wavelength of the illuminating electric field. The three angles define the plane wave that is incident on the grating, as shown in Fig 4-2.

θ is the angle of incidence with respect to the z-axis

ϕ is the rotational angle about the z-axis

ψ is the polarization angle 90° = s-polarization 0°=p-polarization.

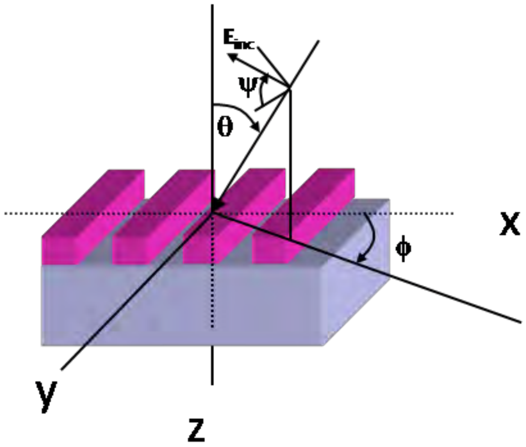


Fig 4-2. Angle definitions for incident light in RCWT.

5. Example in RCWT calculator

After understanding the procedures to use RCWT calculator in OptiScan, an example is given in this section to calculate the diffraction efficiency for binary dielectric grating for TE, TM polarization and conical mount. The results generated from RCWT calculator are then compared with the published data in Moharm's paper. []

The procedures for the example in RCWT calculator is given in the following:

1. In order to get started, open OptiScan in MATLAB, and generate a new project. An OptiScan main project workspace window is given in the Fig 5-1. Click on the Accessories -- > RCWT Calculator from the top menu bar. This opens the main calculator panel.

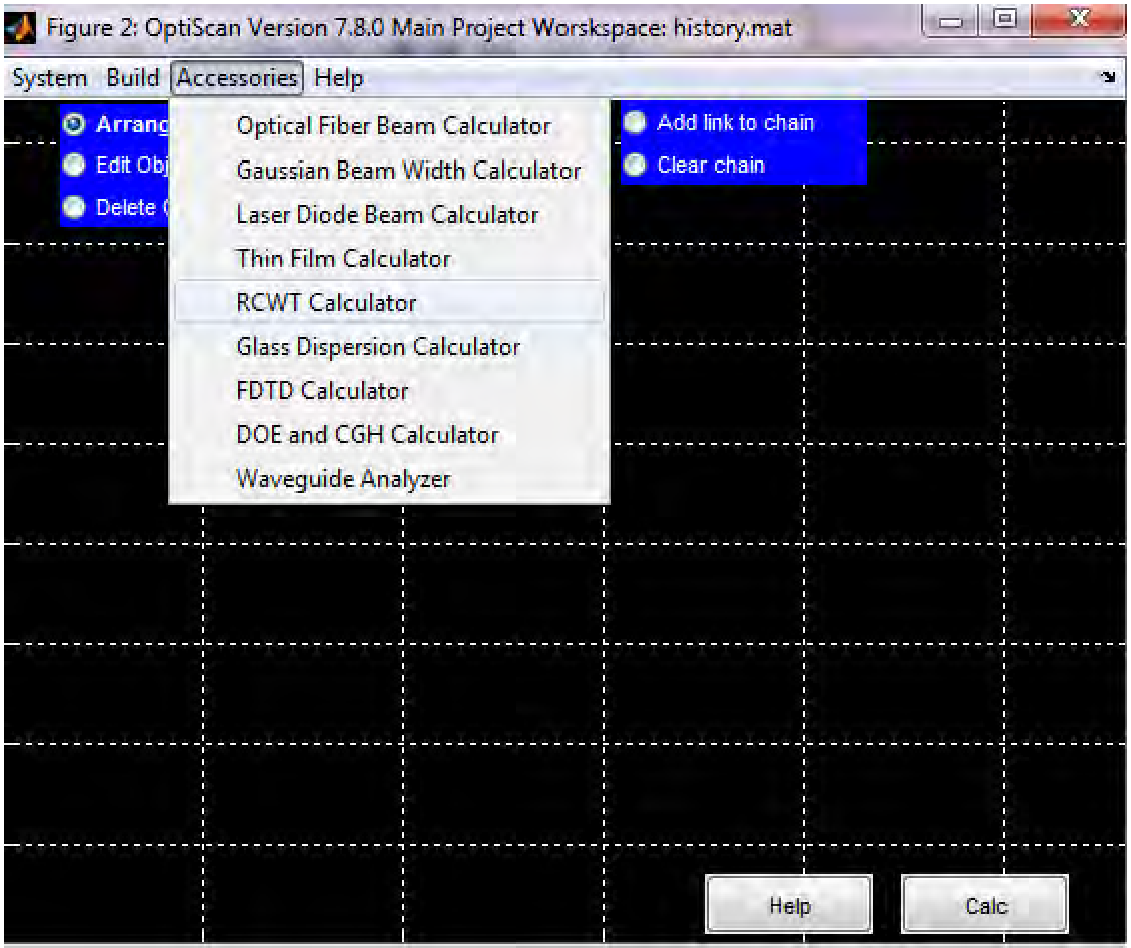


Figure 2: OptiScan Version 7.8.0 Main Project Worskspace: history.mat
System
Build
Accessories
Help
Arrang
Edit Obj
Delete (
Optical Fiber Beam Calculator
Gaussian Beam Width Calculator
Laser Diode Beam Calculator
Thin Film Calculator
RCWT Calculator
Glass Dispersion Calculator
FDTD Calculator
DOE and CGH Calculator
Waveguide Analyzer
Add link to chain
Clear chain
Help
Calc

Fig 5-1. RCWT Calculator in OptiScan Project Workspace window

2. The main calculator panel is the primary interface for the RCWT calculator. Form this panel, there are buttons which allow the user to define a grating structure, generate a basis set, calculate diffraction efficiency, and simulate a single plane wave incident upon the grating structure. In this example, the diffraction efficiency of a binary dielectric grating is demonstrated. Other functions are similar, and interested users can refer the OptiScan help.

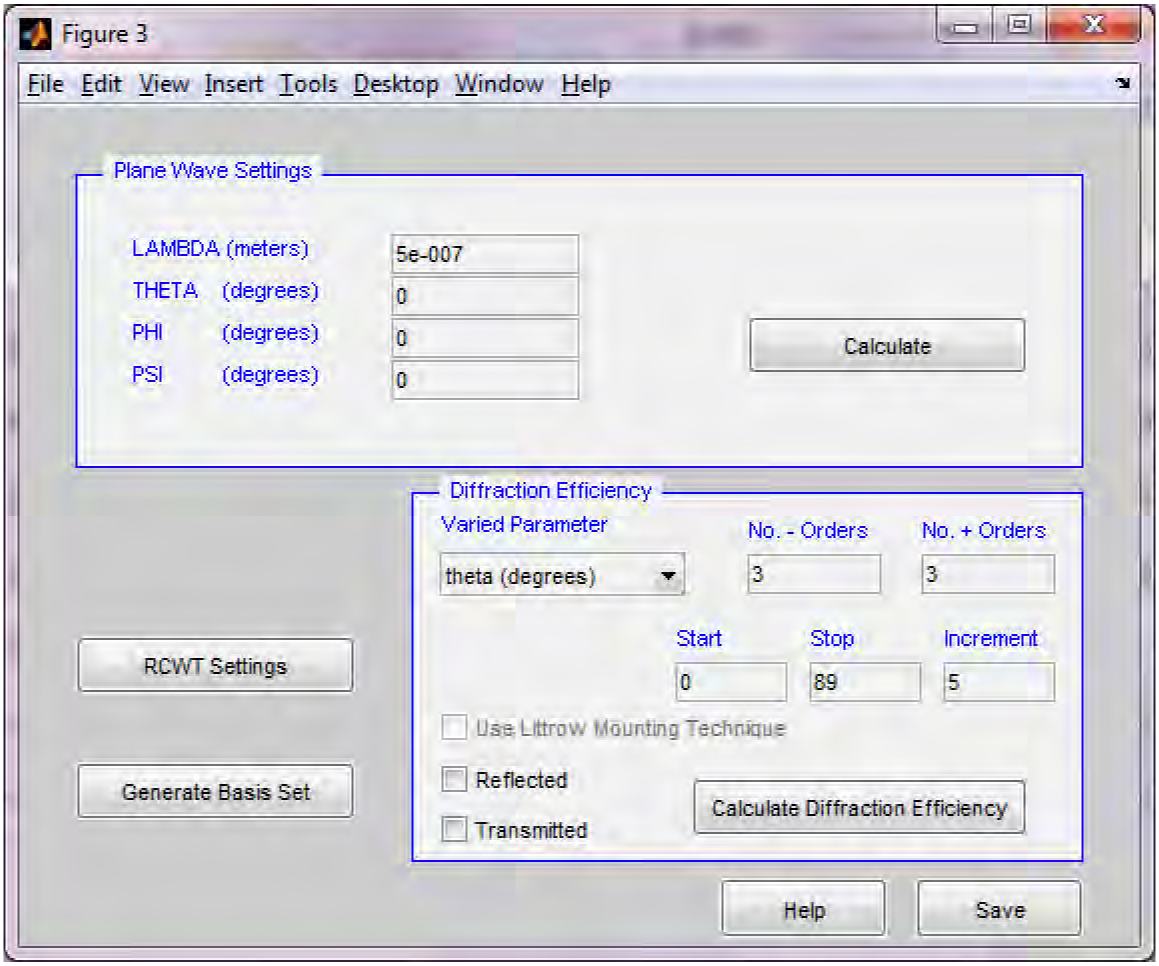


Figure 3
File Edit View Insert Tools Desktop Window Help
Calculate
RCWT Settings
Generate Basis Set
Calculate Diffraction Efficiency
Help
Save

Plane Wave Settings

LAMBDA (meters) 5e-007

THETA (degrees) 0

PHI (degrees) 0

PSI (degrees) 0

Diffraction Efficiency

Varied Parameter: theta (degrees)

No. - Orders: 3

No. + Orders: 3

Start: 0

Stop: 89

Increment: 5

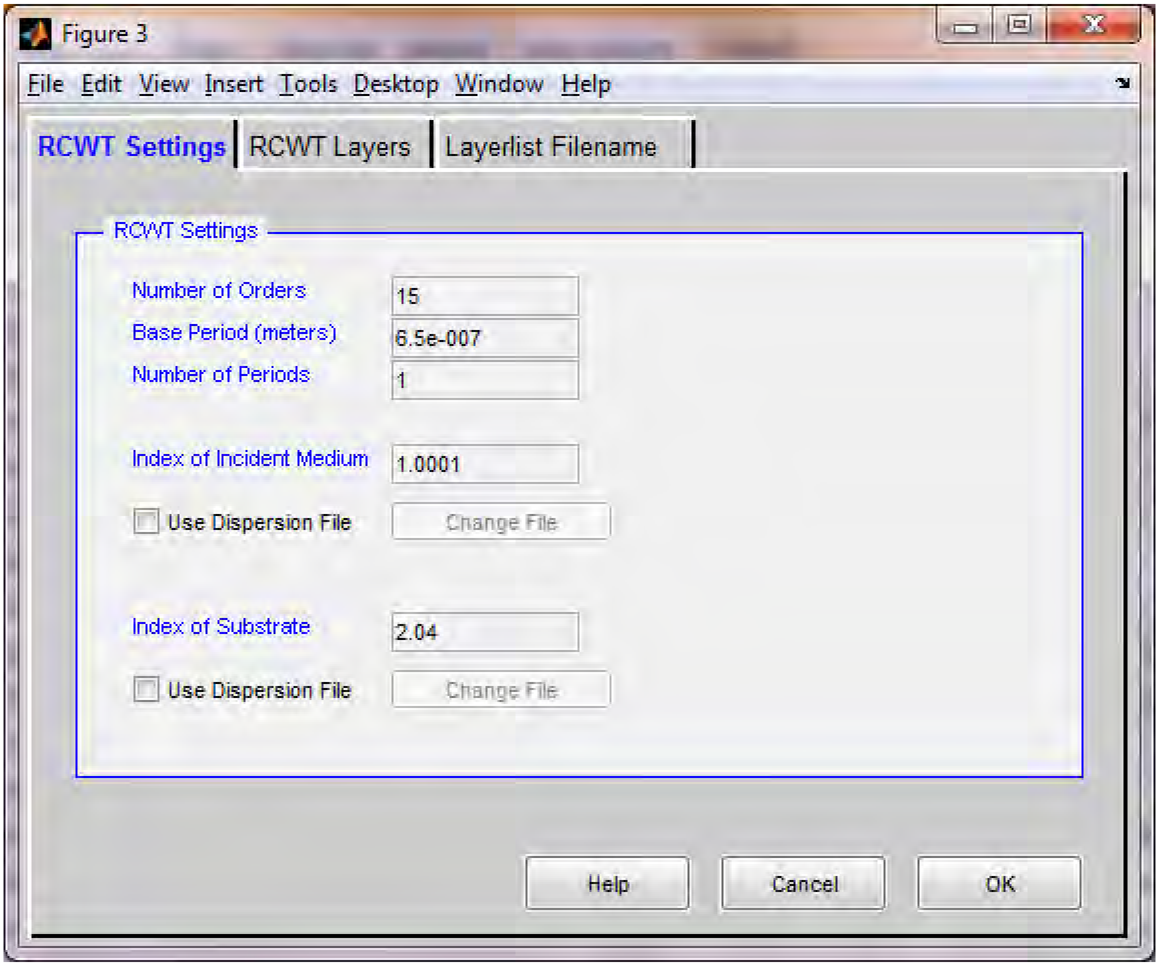
☐ Use Littrow Mounting Technique

☐ Reflected

☐ Transmitted

Fig 5-2. RCWT Calculator panel

3. Click on the RCWT Settings in the panel to work on the general information for grating structure and calculation, as shown in Fig 5-2. In the RCWT Settings, choose “Number of orders” to be 15, “Base Period (meters)” to be 650e-9, “Number of Periods” to be 1. The “Index of Incident Medium” is set to be 1.0001 instead of 1, which is used to solve the possible singularity problem in RCWT calculator. And the “Index of Substrate” is set to be 2.04. All the parameters can be seen in Fig 5-2.



RCWT Settings

Number of Orders 15

Base Period (meters) 6.5e-007

Number of Periods 1

Index of Incident Medium 1.0001

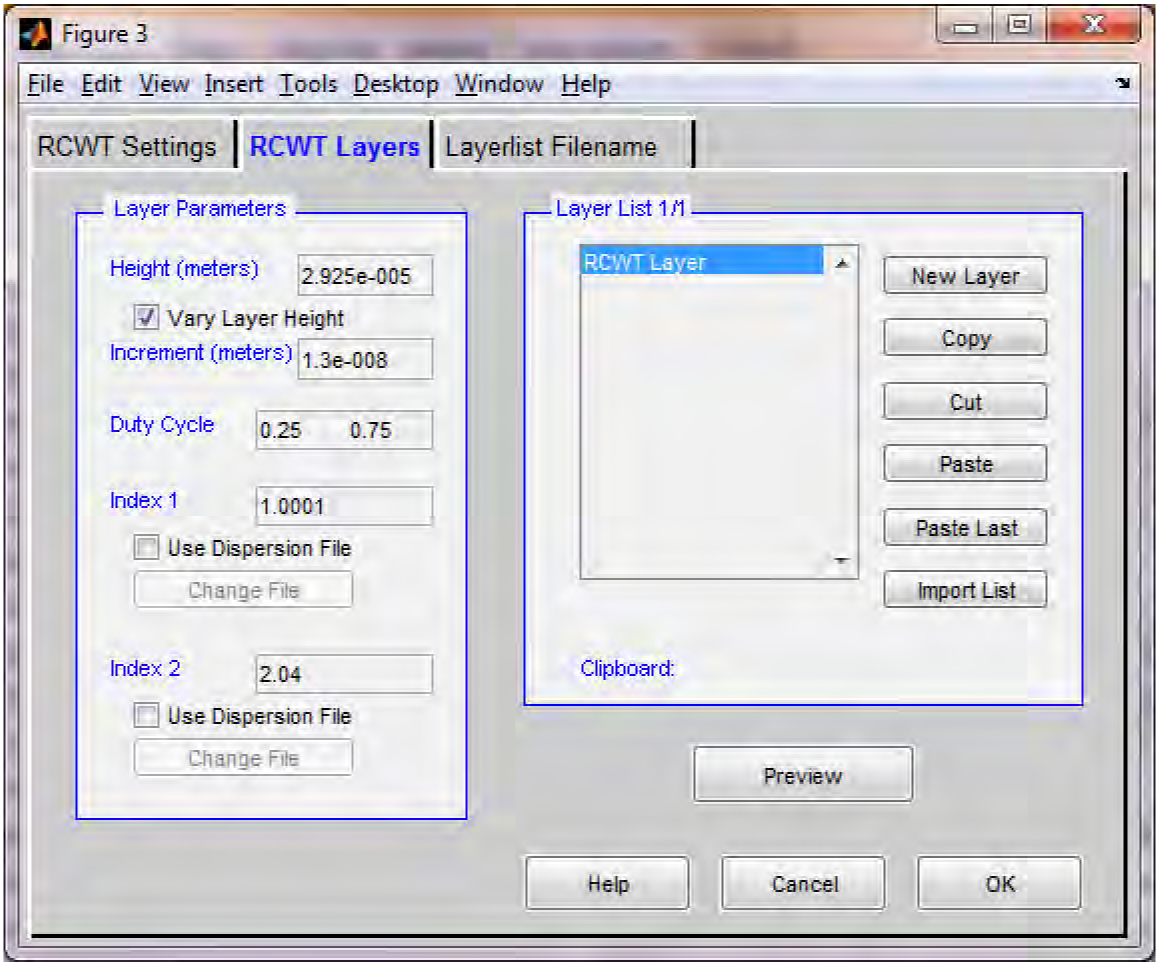
□ Use Dispersion File Change File

Index of Substrate 2.04

□ Use Dispersion File Change File

Fig 5-3. Panel for RCWT Settings

4. Then click on the RCWT Layers to define the height, duty cycle and indices for the grating. The panel is given in Fig 5-4. Set the Height (meters) to be 45λ, which is 2.925e-5, click on Vary Layer Height so that different grating height can be calculated with the defined Increment (meters). In this example, it is set to be 0.02λ, which is 1.3e-8. Duty cycle is [0.25 0.75 1]. "Index 1" is set to be 1.0001, and "Index 2" is set to be 2.04. All parameters can be seen in Fig 5-4.



Layer Parameters

Height (meters) 2.925e-005

☑ Vary Layer Height

Increment (meters) 1.3e-008

Duty Cycle 0.25 0.75

Index 1 1.0001

☐ Use Dispersion File

Index 2 2.04

☐ Use Dispersion File

Layer List 1/1

RCWT Layer

Clipboard:

Fig 5-4. RCWT Layers panel

5. Click on the Preview to see the grating structure, as shown in Fig 5-5. After checking the grating structure, click on OK to return back to the RCWT main panel to assign the wavelength, orientation, and polarization of incident light.

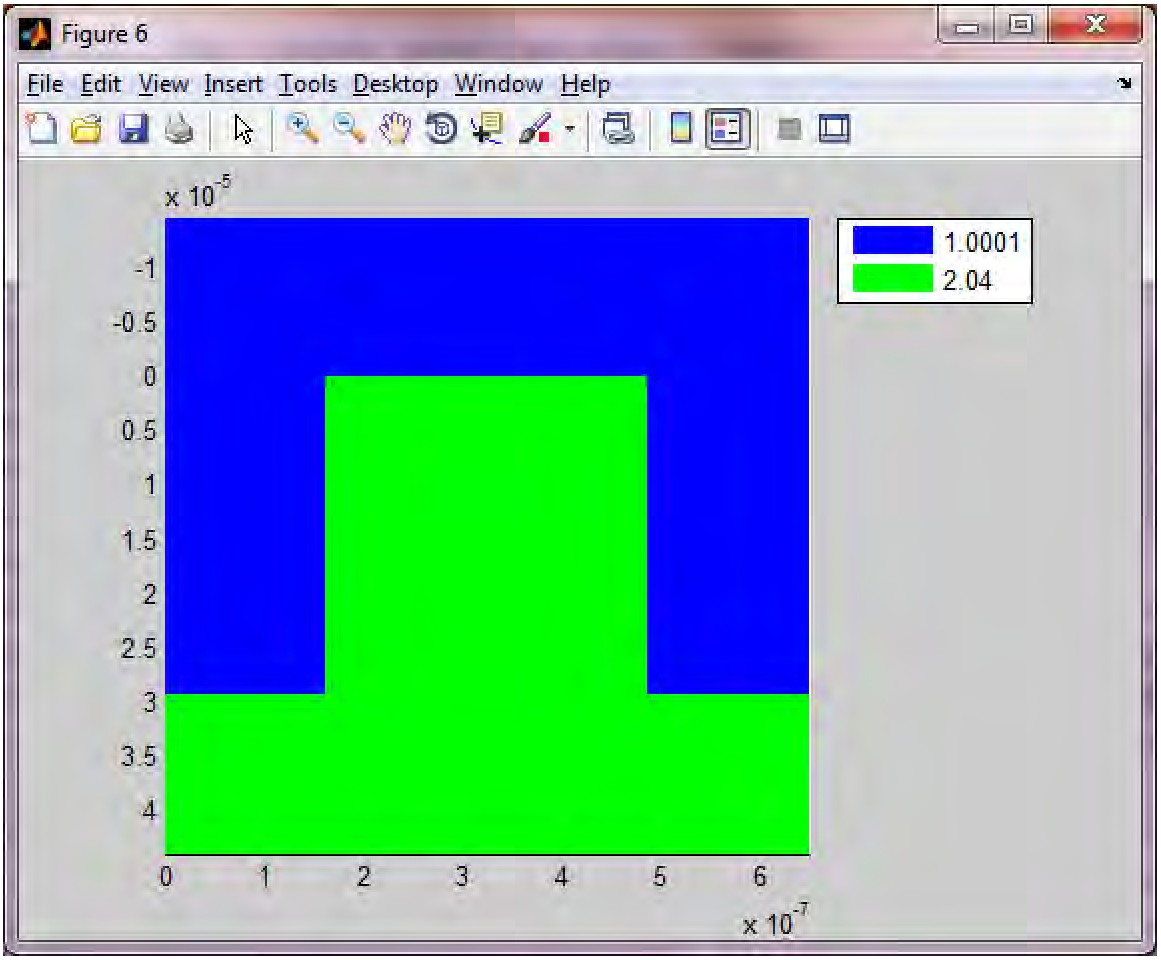


Fig 5-5. Binary dielectric grating geometry plotted by RCWT Calculator in OptiScan

6. Set the wavelength, orientation, polarization of incident light in the RCWT main panel. Also set the parameter in the Diffraction Efficiency. In this example, LAMBDA (meters) is set to be 650e-9 for Plane Wave Settings. In Diffraction Efficiency, select Varied Parameter to be "layer height (meters)", No.-orders to be 1, No.+order to be 1, and Stop for the grating height to be 50λ, which is 3.25e-5. Transmitted button is selected to calculate the diffraction efficiency. Three orientation and polarization incident light are used, which is given in the following table with TE, TM polarization and conical mount. The panel for TE is shown in Fig 5-6.

Three Types of Incident light

Incident light	LAMBDA (meters)	THETA (degrees)	PHI (degrees)	PSI (degrees)
TE	650e-9	10	0	90
TM	650e-9	10	0	0
Conical mount	650e-9	10	30	45

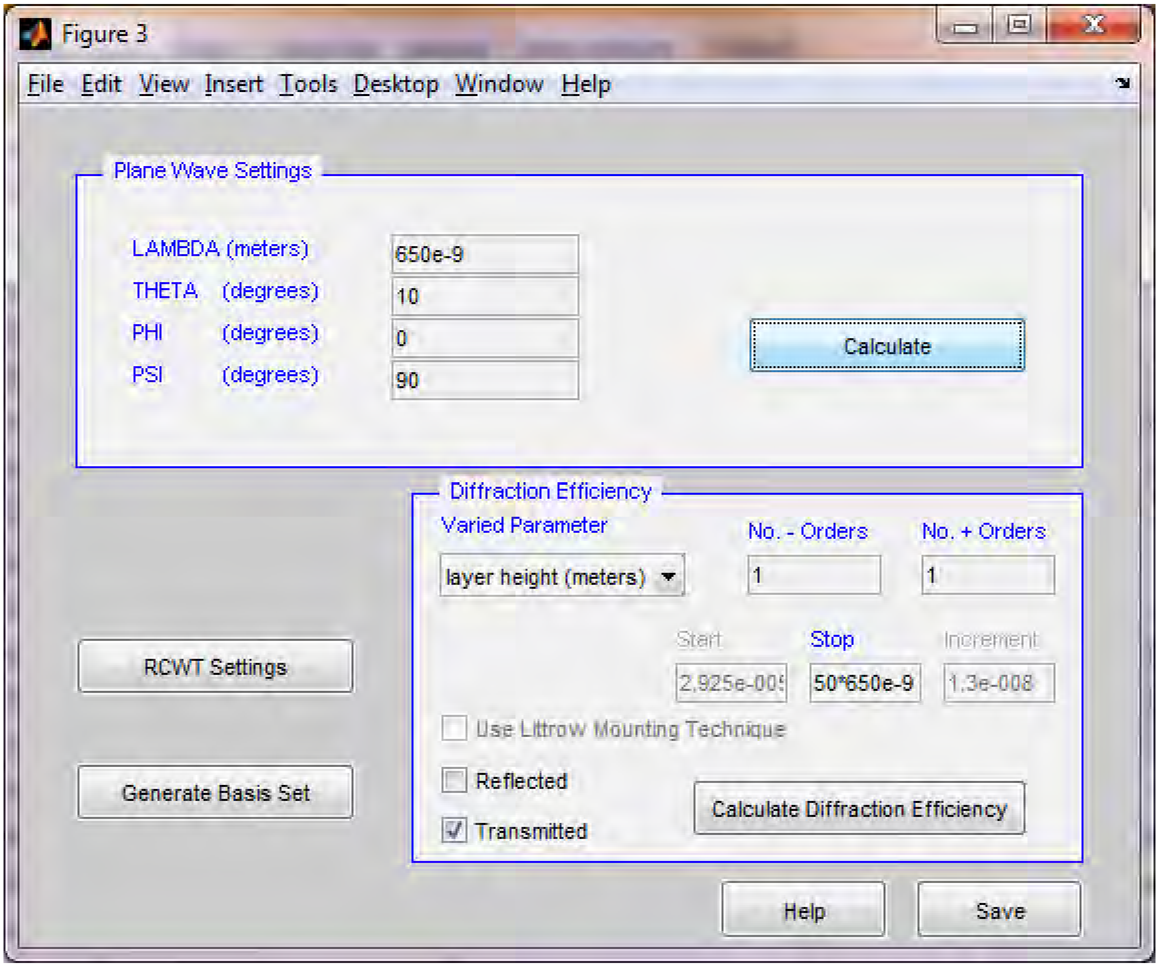


Figure 3
File Edit View Insert Tools Desktop Window Help
Calculate
RCWT Settings
Generate Basis Set
Calculate Diffraction Efficiency
Help
Save

Plane Wave Settings

LAMBDA (meters) 650e-9

THETA (degrees) 10

PHI (degrees) 0

PSI (degrees) 90

Diffraction Efficiency

Varied Parameter: layer height (meters)

No. - Orders: 1

No. + Orders: 1

Start: 2.925e-005

Stop: 50*650e-9

Increment: 1.3e-008

□ Use Littrow Mounting Technique

□ Reflected

☑ Transmitted

Fig 5-6. RCWT panel for TE illumination

7. Calculate the diffraction efficiency. Click on Transmitted, and click on button "Calculate Diffraction Efficiency", the results window is given in Fig 5-7.

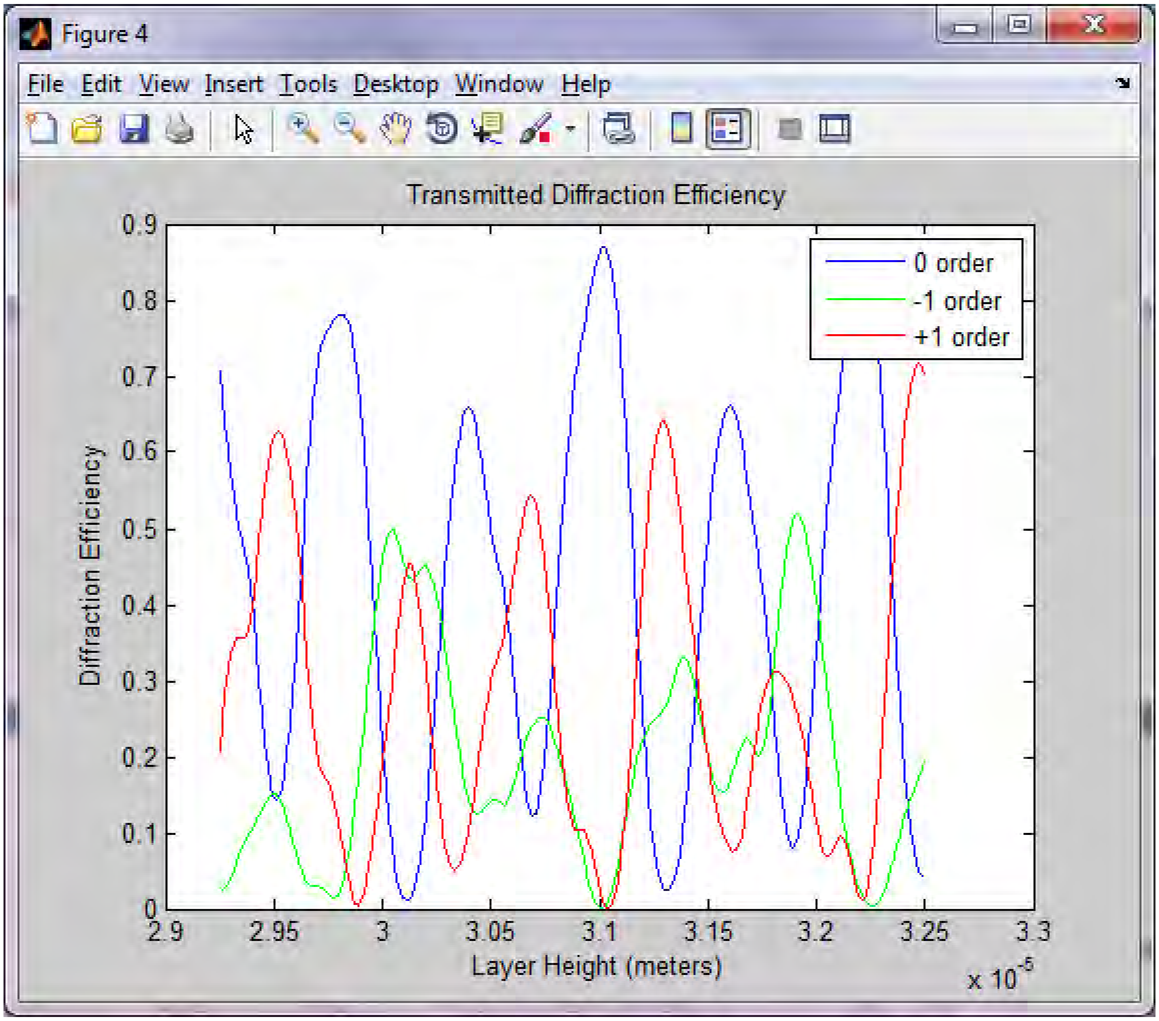


Fig 5-7. Transmitted diffraction efficiency for 0th, 1st, -1st orders for TE illumination with theta=10, phi=0, and psi=90 degrees.

The incident light orientation and polarization can be changed in the RCWT calculator main panel. Use the parameters given in the previous table for TM polarization and conical mount. The results are given in the Fig 5-8 (TM) and Fig 5-9 (Conical mount).

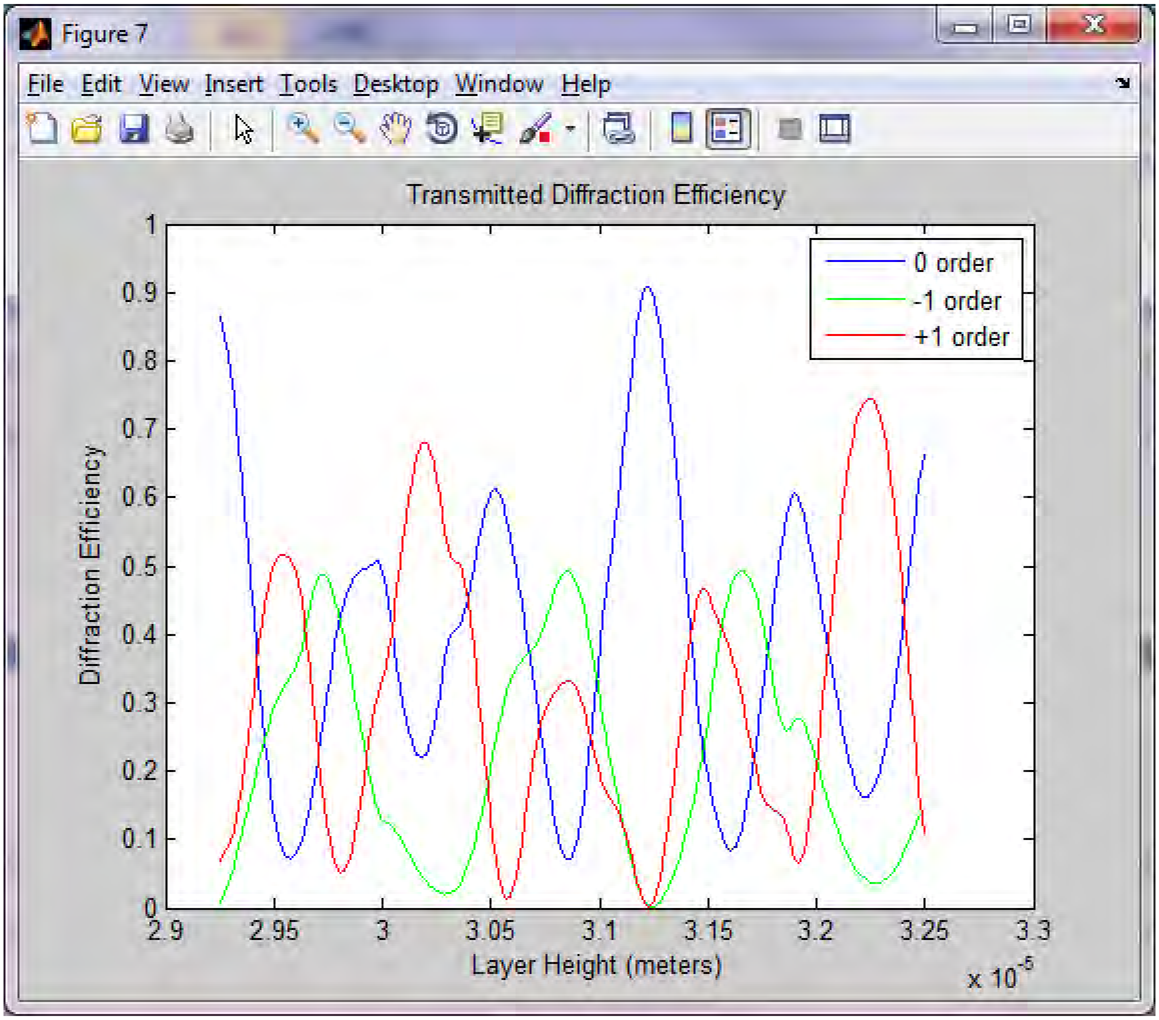


Fig 5-8 Transmitted diffraction efficiency for 0th, 1st, -1st orders for TM illumination with theta=10, phi=0, and psi=0 degrees.

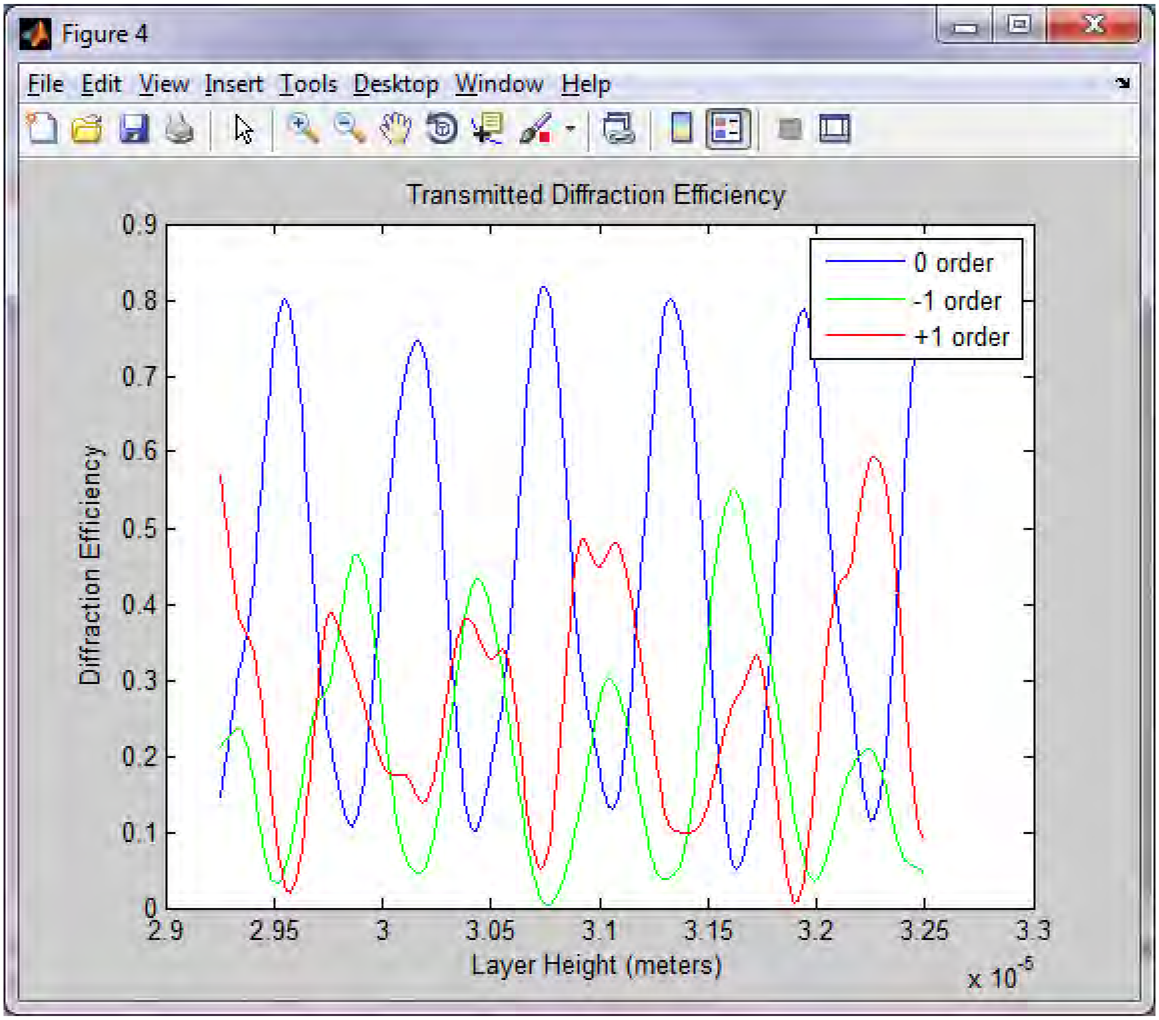
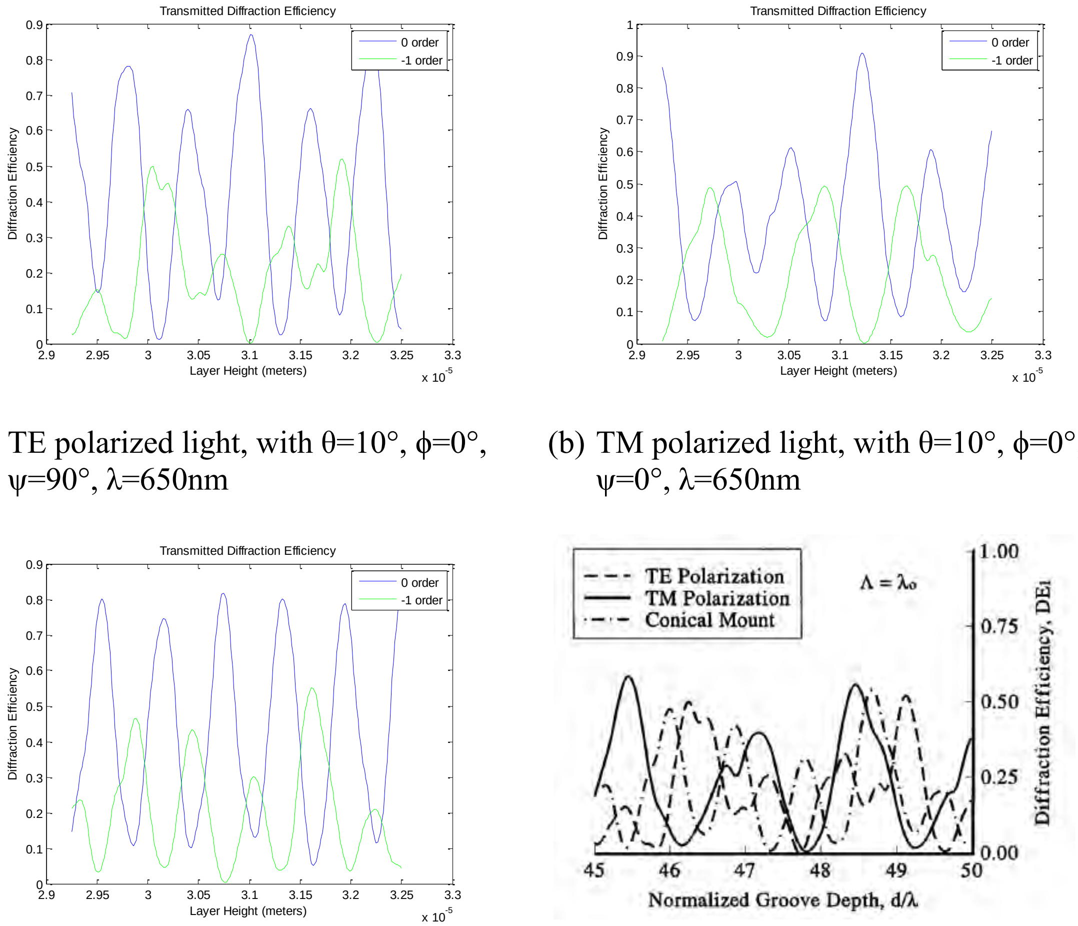


Fig 5-9 Transmitted diffraction efficiency for 0th, 1st, -1st orders for conical mount illumination with theta=10, phi=30 degrees, and psi=45 degrees.

The -1st order diffraction efficiency is chosen for TE, TM, and conical mount to compare with the results in Moharam's paper. However, in OptiScan, zeroth order is always plotted. The results are summarized in Fig 5-10.



(a) TE polarized light, with θ=10°, ϕ=0°, ψ=90°, λ=650nm

(b) TM polarized light, with θ=10°, ϕ=0°, ψ=0°, λ=650nm

(c) Conical mount light, with θ=10°, ϕ=30°, ψ=45°, λ=650nm

(d) -1st order diffraction efficiency for TE, TM, Conical Mount incident light from Moharam's paper

Fig 5-10 Diffraction efficiency dependence on the normalized grating depth of a binary dielectric grating ($n_{grating} = n_{substrate} = 2.04,\ n_{incident} = 1$) for TE, TM polarization and conical mount ($\phi = 30$ and $\psi = 45$) at $\theta = 10$. (a), (b), (c) are results from RCWT calculator in OptiScan, and (d) is from Moharam's paper. Only -1st order are compared. (green color)

6. MATALB code and examples

In this section, MATLAM program is given to calculate the diffraction efficiency for the same binary dielectric grating as before. The results are given in Fig 6-1, which show great agreement with the published data as shown in Fig 6-2.

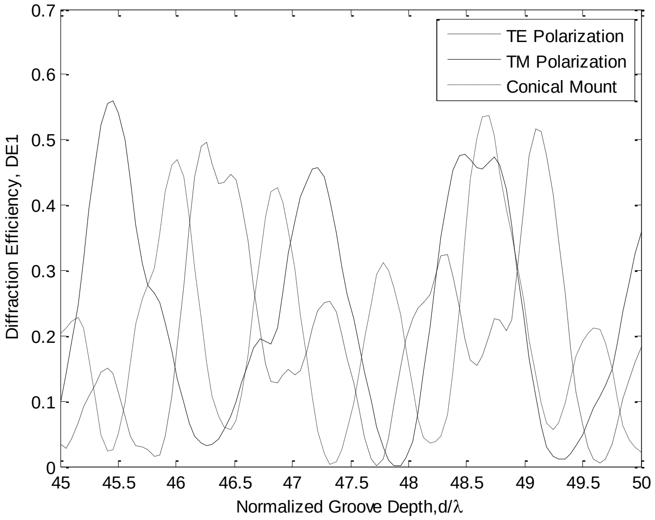


Fig 6-1. Results from RCWT calculator in OptiScan.

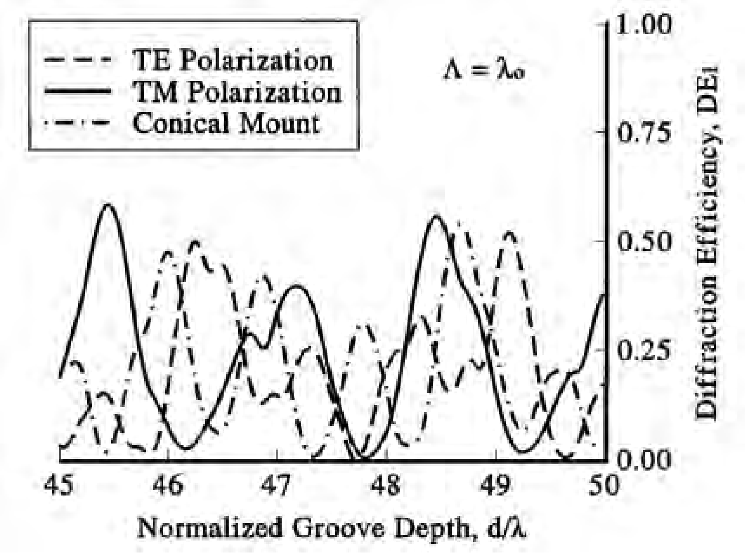


Fig 6-2. Results from Moharam's paper.

6.1 Moharam_1.m

```
%=======================================================================
% Evaluation of  Moharam's Paper
% "Formulation for stable and efficient implementation of
% the rigorous coupled-wave analysis of binary gratings
%
=======================================================================

clear all; close all;

Ndata   = 100;
n1      = 1.00001;
n2      = 2.04;
n_g2    = 1.00001;
n_g1    = 2.04;
cperiod = [0.25 0.75 1];

Lambda  = 650;

grating_period  = 1*Lambda;

theta           = 10;
phi             = 0;
m               = 31;

DE1TE           = [];
DE1TM           = [];
DE1Con          = [];
Start           = 45;
Stop            = 50;
for grating_h       = linspace(Start*Lambda,Stop*Lambda,Ndata),
    Grating.n1      = n1;
    Grating.n2      = n2;
    Grating.ng1     = n_g1;
    Grating.ng2     = n_g2;
    Grating.h       = grating_h;
    Grating.cperd   = cperiod;
    Grating.d       = grating_period;

    psi             = 90;
    phi             = 0;
    [DErs,DErp,DEts,DEtp]   = ...
        DiffractionEfficiencyRCWT(Grating,m,phi,psi,theta,Lambda);
    DE1TE           = [DE1TE; abs(DEts(m+1+1))+abs(DEtp(m+1+1))];


    psi             = 0;
    phi             = 0;
    [DErs,DErp,DEts,DEtp]   = ...
        DiffractionEfficiencyRCWT(Grating,m,phi,psi,theta,Lambda);
    DE1TM           = [DE1TM; abs(DEts(m+1+1))+abs(DEtp(m+1+1))];
    phi              = 30;
    psi              = 45;
    [DErs,DErp,DEts,DEtp]   = ...
        DiffractionEfficiencyRCWT(Grating,m,phi,psi,theta,Lambda);
    DE1Con           = [DE1Con; abs(DEts(m+1+1))+abs(DEtp(m+1+1))];

end
h = linspace(Start,Stop,Ndata)';
plot(h,DE1TE,'--k',h,DE1TM,'-k',h,DE1Con,'-.k');
figure,plot(h,DE1TE,'k');
```

6.2 DiffractionEfficiencyRCWT.m

```
function
[DErs,DErp,DEts,DEtp]=DiffractionEfficiencyRCWT(Grating,m,phiin,psiin,t
hetain,lambda);

n1      = Grating.n1;
n2      = Grating.n2;
ng1     = Grating.ng1;
ng2     = Grating.ng2;
h       = Grating.h;
d       = Grating.d;
cperd   = Grating.cperd;

%fix index issue for positive imag index
n1      = conj(n1);
n2      = conj(n2);
ng1     = conj(ng1);
ng2     = conj(ng2);



multiconipara;
[l, cl]=size(cperd);
if floor(cl/2)==ceil(cl/2)
    disp('Number of transition points must be odd!');
end

O       = zeros(2*m+1);
I       = eye(2*m+1);
O2      = zeros(4*m+2);
I2      = eye(4*m+2);
Y1      = diag(k1ziperk0);
Y2      = diag(k2ziperk0);
Z1      = diag(k1ziperk0/(n1^2));
Z2      = diag(k2ziperk0/(n2^2));
fg      = [ [I O];
            [1i*Y2 O];
            [O I];
            [O 1i*Z2]];
M       = I2;
for L   = l:-1:1,
    [E, invE]    = multilev(m, ng1(L),ng2(L), cperd(L,:));
%     E(abs(E)<1e-13)    = 0;
    A    = Kx2-E;
    %Use Backslash instead of invE for more reliability     by Seung-
Hune
    B    = Kx*(E\Kx)-eye(2*m+1);
    %B    = Kx*inv(E)*Kx-eye(2*m+1);

    [W1,Q12]    = eig(Ky2+Kx2-E,'nobalance');

    %Use Backslash instead of invE for more reliability     by Seung-
Hune
    KEBmat      = Ky2+(Kx*(E\Kx))*E-E;
    %KEBmat      = Ky2+(Kx*inv(E)*Kx)*E-E;                % <<Moharam>>
    %KEBmat=Ky2+(Kx*inv(E)*Kx)*inv(invE)-inv(invE);     % <<Neil>>

    [W2,Q22]    = eig(KEBmat,'nobalance');

    Q12     = diag(Q12);                               % <<Moharam>>
    Q22     = diag(Q22);                               % <<Moharam>>
    %Q12=real(diag(Q12))+1i*imag(diag(Q12));           % <<Neil>>
    %Q22=real(diag(Q22))+1i*imag(diag(Q22));           % <<Neil>>

    Q1      = k0*(diag(sqrt(Q12)));
    Q1perk0 = (diag(sqrt(Q12)));
    Q2      = k0*(diag(sqrt(Q22)));
    Q2perk0 = (diag(sqrt(Q22)));


    %Use Backslash instead of invE for more reliability     by Seung-
Hune
    V11     = (A\W1)*Q1perk0;
    V12     = kyperk0*(A\Kx)*W2;
    %V11     = inv(A)*W1*Q1perk0;
    %V12     = kyperk0*inv(A)*Kx*W2;

    %Use Backslash instead of invE for more reliability     by Seung-
Hune
    V21     = kyperk0*(B\Kx)*(E\W1);
    V22     = (B\W2)*Q2perk0;
    %V21     = kyperk0*inv(B)*Kx*inv(E)*W1;
    %V22     = inv(B)*W2*Q2perk0;

    X1      = diag(exp(-diag(Q1)*h(L)));
    X2      = diag(exp(-diag(Q2)*h(L)));
    X       = [[X1 O];[O X2]];

    if theta == 0
        phii        = atan(kyperk0./kxperk0(1:m));
        phii(m+1)   = 0;
        phii(m+2:2*m+1)     = atan(kyperk0./kxperk0(m+2:2*m+1));
    elseif kyperk0 == 0
        phii=atan2(kyperk0,kxperk0);
    else
        phii=atan2(kyperk0,kxperk0);
    end

    Fc      = diag(cos(phii));
    Fs      = diag(sin(phii));
    Vss     = Fc*V11;
    Wss     = Fc*W1+Fs*V21;
    Vsp     = Fc*V12-Fs*W2; %NOTE a typo in Ref. 1!

    Wsp     = Fs*V22;
    Wpp     = Fc*V22;
    Vpp     = Fc*W2+Fs*V12;
    Wps     = Fc*V21-Fs*W1;
    Vps     = Fs*V11;
    W11     = [[Vss Vsp]; [Wss Wsp]];
    W21     = [[Wps Wpp]; [Vps Vpp]];
    W12     = [[Vss Vsp]; [-Wss -Wsp]];
    W22     = [[-Wps -Wpp]; [Vps Vpp]];
    if (theta==0)||(phiin==0)||(phiin==180)
%Use Backslash instead of invE for more reliability     by Seung-Hune
        c1  =.5*(Vss\eye(size(Vss)));
        c3  =.5*(Wss\eye(size(Wss)));
        c6  =.5*(Wpp\eye(size(Wpp)));
        c8  =.5*(Vpp\eye(size(Vpp)));
%         c1  =.5*inv(Vss);
%         c3  =.5*inv(Wss);
%         c6  =.5*inv(Wpp);
%         c8  =.5*inv(Vpp);

        invers  =   [   [c1 c3 O O];
                        [O O c6 c8];
                        [c1 -c3 O O];
                        [O O -c6 c8]    ];
        ab  = invers*fg;
    else
        invers  =([  [Vss Vsp Vss Vsp];
                        [Wss Wsp -Wss -Wsp];
                        [Wps Wpp -Wps -Wpp];
                        [Vps Vpp Vps Vpp]   ]);
        ab  = invers\fg;
    end

    a   = ab(1:4*m+2,:);
    b   = ab(4*m+3:8*m+4,:);
    warning off;
%Use Backslash instead of invE for more reliability     by Seung-Hune
    fg  = [ [W11+W12*X*b*(a\X)];    % by Seung-Hune
            [W21+W22*X*b*(a\X)]  ];
    M   = M*(a\X);
    warning on;
%    fg  = [ [W11+W12*X*b*inv(a)*X];    % by Neil
%            [W21+W22*X*b*inv(a)*X]  ];
%    M   = M*inv(a)*X;
end
f11     = fg(1:2*m+1,1:2*m+1);
f12     = fg(1:2*m+1,2*m+2:4*m+2);
f21     = fg(2*m+2:4*m+2,1:2*m+1);
f22     = fg(2*m+2:4*m+2,2*m+2:4*m+2);
g11     = fg(4*m+3:6*m+3,1:2*m+1);
g12     = fg(4*m+3:6*m+3,2*m+2:4*m+2);
g21     = fg(6*m+4:8*m+4,1:2*m+1);
g22     = fg(6*m+4:8*m+4,2*m+2:4*m+2);

matr=[  [-I O f11 f12]; %this is the original  one
        [1i*Y1 O f21 f22];
        [O -I g11 g12];
        [O 1i*Z1 g21 g22]   ];

RT      = matr\[aa; bb; cc; dd];
Rs      = RT(1:2*m+1);
```

```
Rp      = RT(2*m+2:4*m+2);
Ts      = RT(4*m+3:6*m+3);
Tp      = RT(6*m+4:8*m+4);
TsTp    = M*[Ts;Tp];
Ts      = TsTp(1:2*m+1);
Tp      = TsTp(2*m+2:4*m+2);

DEr=abs(Rs).^2.*real(k1ziperk0/(n1*cos(theta)))'+abs(Rp).^2.*real(k1zip
erk0/(n1^3*cos(theta)))';
DEt=abs(Ts).^2.*real(k2ziperk0/(n1*cos(theta)))'+abs(Tp).^2.*real(k2zip
erk0/(n1*n2^2*cos(theta)))';
eta0=DEt(m+1);
delta0=(angle(Ts(m+1))-angle(Tp(m+1)))/pi*180;
alpha0=atan(abs(Ts(m+1))/abs(Tp(m+1)))/pi*180;

DErs=abs(Rs).^2.*real(k1ziperk0/(n1*cos(theta)))';
DErp=abs(Rp).^2.*real(k1ziperk0/(n1^3*cos(theta)))';
DEts=abs(Ts).^2.*real(k2ziperk0/(n1*cos(theta)))';
DEtp=abs(Tp).^2.*real(k2ziperk0/(n1*n2^2*cos(theta)))';
```

6.3 multiconipara.com

```
phi      = phiin/360*2*pi;
theta    = thetain/360*2*pi;
psi      = psiin/360*2*pi;
k0       = 2*pi/lambda;
kyperk0 = n1*sin(theta)*sin(phi);

if abs(kyperk0)<1e-12
    kyperk0=0;
end
ky       = kyperk0*k0;
ky2      = kyperk0.^2;
Ky2      = ky2*eye(2*m+1);

% Calculate kx0 with n of incident material, n1*sin();
i=-m:m;
kxperk0     = n1*sin(theta)*cos(phi)-i*(lambda/d);  % Grating Equation

indx     = find(abs(kxperk0)<1e-14);
kxperk0(indx)    = 0;

% k vectors
kx       = kxperk0*k0;
Kx       = diag(kxperk0);
Kx2      = diag(kxperk0.^2);

k1zi        = k0*sqrt(n1^2-(kxperk0.^2+ky2));
if imag(n1) == 0,
    k1zi        = conj(k1zi);       %for dielectric
end
k1ziperk0   = sqrt(n1^2-(kxperk0.^2+ky2));
if imag(n1) == 0,
    k1ziperk0   = conj(k1ziperk0);  %for dielectric
end

k2zi        = k0*sqrt(n2^2-(kxperk0.^2+ky2));
if imag(n2) == 0,
    k2zi        = conj(k2zi);       %for dielectric
end
k2ziperk0   = sqrt(n2^2-(kxperk0.^2+ky2));
if imag(n2) == 0,
    k2ziperk0   = conj(k2ziperk0);  %for dielectric
end

aa       = zeros(2*m+1,1);
bb       = aa;
cc       = aa;
dd       = aa;
O        = aa;

aa(m+1)  = sin(psi);
bb(m+1)  = 1i*sin(psi)*n1*cos(theta);
cc(m+1)  = -1i*cos(psi)*n1;
dd(m+1)  = cos(psi)*cos(theta);
```

6.4 multilev.m

```
function    [E,invE] = multilev(m,ng1,ng2,cperd)

cl                  = length(cperd);

for cc  = 1:cl+1
   multip(cc)      = (-1)^(1+ceil(cc/2)-floor(cc/2));
end

epsil(1)          = transpose((ng2^2-
ng1^2)*cperd*multip(2:cl+1)')+ng2^2;
inveps(1)         = (1/ng2^2-1/ng1^2)*cperd*multip(2:cl+1)'+1/ng2^2;

for i = 1:2*m
   fac(1)          = 1;
   fac(2:cl+1)     = exp(-1i*2*pi*(i)*(cperd)); %shift in grating
   fac2(1)         = 1;
   fac2(2:cl+1)    = exp(1i*2*pi*(i)*(cperd));
   epsil(i+1)      = 1i ./(2*pi*i)*(ng2^2-ng1^2)*fac*transpose(multip);
   epsil(4*m+2-i)  =  -1i  ./(2*pi*i)*(ng2^2-ng1^2)*fac2*transpose(multip);
   inveps(i+1)     = 1i ./(2*pi*i)*(1/ng2^2-
1/ng1^2)*fac*transpose(multip);
   inveps(4*m+2-i)= -1i ./(2*pi*i)*(1/ng2^2-
1/ng1^2)*fac2*transpose(multip);
end

%===========================================================
%epsil=epsil.*exp(-1i*2*pi*[0:4*m]*(2*m+1)/(4*m+1));        <<Neil>>
%inveps=inveps.*exp(-1i*2*pi*[0:4*m]*(2*m+1)/(4*m+1));      <<Neil>>
%===========================================================

for i = 1:2*m+1
   E(i:2*m+1,i)    = transpose(epsil(1:2*m+2-i));
   invE(i:2*m+1,i)= transpose(inveps(1:2*m+2-i));
end

for i = 2:2*m+1
   E(1:i-1,i)      = transpose(epsil(4*m+3-i:4*m+1));
  invE(1:i-1,i)   = transpose(inveps(4*m+3-i:4*m+1));
end
```