Instructor
Dr. Raymond Rumpf
(915) 747-6958
rcrumpf@utep.edu

EE 5337
Computational Electromagnetics

Lecture #23
RCWA Extras

Outline
• One spatial harmonic: $P=Q=1$
• Simulation of 1D Gratings with 3D-RCWA
• Formulation of 2D-RCWA with fast Fourier factorization
• Danger of RCWA and convergence
• RCWA and curved structures
• Strategically truncating the set of spatial harmonics
• RCWA for generalized symmetries
• Modeling hexagonal gratings with rectangular RCWA
• Enhanced transmittance matrix approach
One Spatial Harmonic

\[ P = Q = 1 \]

Anatomy of the Convolution Matrix

Device \( \varepsilon_r(x, y) \)

Convolution Matrix \( [\varepsilon_r] \)

Higher order Fourier coefficients describe periodic variations in \( \varepsilon_r(x, y) \)

0-order Fourier coefficient is the average value
One Spatial Harmonic ($P=Q=1$)

When only one spatial harmonic is used, RCWA reduces to the 1D transfer matrix method, but uses the average value for the material properties in each layer. This is first-order “effective medium theory” and can make use of fast Fourier factorization so it is more sophisticated than the standard TMM.

$\varepsilon_1 < \varepsilon_2 < \varepsilon_3 < \varepsilon_4 < \varepsilon_r$

Simulation of 1D Gratings with 3D-RCWA
Grating Terminology

1D grating
Ruled grating
Requires a 2D simulation

2D grating
Crossed grating
Requires a 3D simulation

3D-RCWA for 1D Gratings

Three-dimensional RCWA simulates all polarizations at the same time.

For 1D Gratings, Maxwell’s equations decouple into the $E$ mode and the $H$ mode.

It is possible to reformulate RCWA specifically for 1D Gratings so that it will only simulate either the $E$ mode or $H$ mode, but not both. This approach will be several times faster due to smaller matrices.

It is my experience that 3D-RCWA is fast enough that few applications warrant formulating a 2D-RCWA code. Exceptions include when fast Fourier factorization is important or for running optimizations that require many thousands of simulations to be iterated.

There are some tricks that can be used when using RCWA to model 1D gratings to maximize the speed and efficiency.
Number of Spatial Harmonics

You must use some number of spatial harmonics in this direction.

\[ P \approx 7 \frac{\Lambda}{\lambda} \]

\[ N_x > 500 \]

There is no contrast in this direction. Only one spatial harmonic is needed.

\[ Q = 1 \]

\[ N_y = 1 \]

---

E and H Modes with 3D-RCWA

The field for both \( E \) and \( H \) modes are thought of completely in terms of the polarization vector for the electric field. If you have implemented your codes following these lectures, no changes to your code are needed outside of the dashboard.

The concept of \( E \) and \( H \) modes only apply when: (1) LHI or diagonally anisotropic materials, (2) device is uniform in \( y \) direction, and (3) propagation is restricted to \( x-z \) plane.
FDFD Vs. RCWA Representations

FDFD represents a device as the side view of a single unit cell.

In contrast, RCWA represents a device as a top view of each layer of a single unit cell.

Formulation of 2D RCWA with FFF (1D Gratings)
Starting Point for Derivation

We will start with the semi-analytical matrix form of Maxwell’s equations in Fourier-space. These are rigorous and valid even for 3D devices.

\[-j\mathbf{\tilde{K}}_y u_z - \frac{d}{dz} u_y = \begin{bmatrix} \varepsilon_r \end{bmatrix} s_x\]

\[\frac{d}{dz} u_x + j\mathbf{\tilde{K}}_x u_z = \begin{bmatrix} \varepsilon_r \end{bmatrix} s_y\]

\[\begin{bmatrix} \mathbf{\tilde{K}}_x \end{bmatrix} u_y - \mathbf{\tilde{K}}_y u_x = j\begin{bmatrix} \varepsilon_r \end{bmatrix} s_z\]

\[-j\mathbf{\tilde{K}}_y s_z - \frac{d}{dz} s_y = \begin{bmatrix} \mu_r \end{bmatrix} u_x\]

\[\frac{d}{dz} s_x + j\mathbf{\tilde{K}}_x s_z = \begin{bmatrix} \mu_r \end{bmatrix} u_y\]

\[\begin{bmatrix} \mathbf{\tilde{K}}_x \end{bmatrix} s_y - \mathbf{\tilde{K}}_y s_x = j\begin{bmatrix} \mu_r \end{bmatrix} u_z\]

Reduction to Two Dimensions

When devices are uniform in the \(y\)-direction and no wave propagation occurs in this direction, we have

\[\mathbf{\tilde{K}}_y = 0\]

Our matrix equations reduce to

\[-\frac{d}{dz} u_y = \begin{bmatrix} \varepsilon_r \end{bmatrix} s_x\]

\[-\frac{d}{dz} s_y = \begin{bmatrix} \mu_r \end{bmatrix} u_x\]

\[\frac{d}{dz} u_x + j\mathbf{\tilde{K}}_x u_z = \begin{bmatrix} \varepsilon_r \end{bmatrix} s_y\]

\[\frac{d}{dz} s_x + j\mathbf{\tilde{K}}_x s_z = \begin{bmatrix} \mu_r \end{bmatrix} u_y\]

\[\begin{bmatrix} \mathbf{\tilde{K}}_x \end{bmatrix} u_y = j\begin{bmatrix} \varepsilon_r \end{bmatrix} s_z\]

\[\begin{bmatrix} \mathbf{\tilde{K}}_x \end{bmatrix} s_y = j\begin{bmatrix} \mu_r \end{bmatrix} u_z\]
Two Independent Modes

We see that Maxwell’s equations have decoupled into two independent modes.

\[
\begin{align*}
\frac{d}{dz} \mathbf{u}_x &= \left[ \varepsilon_r \right] \mathbf{s}_x, \\
\frac{d}{dz} \mathbf{u}_y &= \left[ \varepsilon_r \right] \mathbf{s}_y, \\
\mathbf{K} \mathbf{s}_y &= j \left[ \mu_r \right] \mathbf{u}_z. \\
\frac{d}{dz} \mathbf{s}_y &= \left[ \mu_r \right] \mathbf{u}_x, \\
\frac{d}{dz} \mathbf{s}_z &= \left[ \mu_r \right] \mathbf{u}_y, \\
\mathbf{K} \mathbf{s}_z &= j \left[ \varepsilon_r \right] \mathbf{u}_y. \\
\end{align*}
\]

E Mode

H Mode

Orientation of the Field Components

For 1D gratings, the orientation of the field components relative to the interfaces is fixed. In this case, it is straightforward to incorporate fast Fourier factorization into the formulation.

Within the layer...

\( S_x \) is always perpendicular to interfaces.
\( S_y \) is always parallel to interfaces.
\( S_z \) is always parallel to interfaces.

Within the layer...

\( U_x \) is always perpendicular to interfaces.
\( U_y \) is always parallel to interfaces.
\( U_z \) is always parallel to interfaces.
Incorporating Fast Fourier Factorization

For 1D gratings, it is very straightforward to incorporate fast Fourier factorization rules. This will improve convergence rates.

**E Mode**
\[
\frac{d}{d\tilde{z}} \textbf{u}_x + j\tilde{K}_x \textbf{u}_z = [\varepsilon_x] \textbf{s}_y
\]
\[
\frac{d}{d\tilde{z}} \textbf{s}_y = [1/\mu_x]^{-1} \textbf{u}_x
\]
\[
\tilde{K}_x \textbf{s}_y = j[\mu_x] \textbf{u}_z
\]

**H Mode**
\[
\frac{d}{d\tilde{z}} \textbf{s}_x + j\tilde{K}_x \textbf{s}_z = [\mu_x] \textbf{u}_y
\]
\[
\frac{d}{d\tilde{z}} \textbf{u}_y = [1/\varepsilon_x]^{-1} \textbf{s}_x
\]
\[
\tilde{K}_x \textbf{u}_y = j[\varepsilon_x] \textbf{s}_z
\]

\( \textbf{s}_y \) is always parallel to interfaces so \([\varepsilon_x]\) is the standard convolution matrix.
\( \textbf{u}_y \) is always parallel to interfaces so \([\mu_x]\) is the standard convolution matrix.
\( \textbf{u}_z \) is always perpendicular so FFF rules are used to construct \([1/\mu_x]\).
\( \textbf{s}_z \) is always perpendicular so FFF rules are used to construct \([1/\varepsilon_x]\).

**Note:** FFF in E-mode with non-magnetic materials makes no difference.

Eliminate Longitudinal Components

We solve the third equation for \( \textbf{u}_z \) and the sixth equation for \( \textbf{s}_z \) to obtain

\[
\textbf{u}_z = -j[\mu_x]^{-1} \tilde{K}_x \textbf{s}_y
\]
\[
\textbf{s}_z = -j[\varepsilon_x]^{-1} \tilde{K}_x \textbf{u}_y
\]

We substitute these expressions into the remaining Maxwell’s equations.

**E Mode**
\[
\frac{d}{d\tilde{z}} \textbf{u}_x = [\varepsilon_x] \textbf{s}_y - \tilde{K}_x [\mu_x]^{-1} \tilde{K}_x \textbf{s}_y
\]
\[
\frac{d}{d\tilde{z}} \textbf{s}_y = -[1/\mu_x]^{-1} \textbf{u}_x
\]

**H Mode**
\[
\frac{d}{d\tilde{z}} \textbf{s}_x = [\mu_x] \textbf{u}_y - \tilde{K}_x [\varepsilon_x]^{-1} \tilde{K}_x \textbf{u}_y
\]
\[
\frac{d}{d\tilde{z}} \textbf{u}_y = -[1/\varepsilon_x]^{-1} \textbf{s}_x
\]
**Standard P and Q Form**

**E Mode**
\[
\frac{d}{dz} s_y = P u_x \\
\frac{d}{dz} u_x = Q s_y
\]

**H Mode**
\[
\frac{d}{dz} u_y = P s_x \\
\frac{d}{dz} s_x = Q u_y
\]

\[
P = -\left[ \frac{1}{\mu_r} \right]^{-1} \\
Q = \left[ \varepsilon_r \right] - \tilde{K}_x \left[ \mu_r \right]^{-1} \tilde{K}_x
\]

We see that FFF is incorporated solely into the matrix \( P \).

---

**Matrix Wave Equations**

Now that we have our equations in standard \( P \) and \( Q \) form, we derive the wave equations in the same manner as before.

**E Mode**
\[
\frac{d^2}{dz^2} s_y - \Omega^2 s_y = 0
\]

**H Mode**
\[
\frac{d^2}{dz^2} u_y - \Omega^2 u_y = 0
\]

\[
\Omega^2 = PQ
\]

...and now you know the rest of the story.
Convergence Study for 1D Gratings

- For $\mu_i = 1.0$, $\varepsilon_i = 3.0$, the ordinary and FFF methods converge.
- For $\mu_i = 1.0$, $\varepsilon_i = 40.0$, the ordinary method converges, but the FFF method does not.

Convergence Study for 1D Curved Structures

- For $\mu_i = 1.0$, $\varepsilon_i = 3.0$, the ordinary and FFF methods converge.
- For $\mu_i = 1.0$, $\varepsilon_i = 40.0$, the ordinary method converges, but the FFF method does not.

H-Mode
Danger of RCWA
and Convergence

In real-space, poor grid resolution led to fluctuations in conservation of power and other very recognizable signs that things are wrong.

The danger of RCWA is that results can “look” correct even with very few spatial harmonics.

Conservation of power will always be obeyed in RCWA even using just one spatial harmonic.

It must become habit to look for convergence, as there are few other signs that more spatial harmonics are needed.
Typical Convergence Plot

- Device from last lecture
  @ $f_0 = 47.6$ GHz

Convergence

- 250 harmonics
- $15 \times 15$ harmonics

RCWA and Curved Structures
EBG Material

Divide into Thin Layers

Layers 1 to 20

Staircase Approximation
Strategically Truncating the Set of Spatial Harmonics

Notes on Truncating the Set of Spatial Harmonics

- The choice of which spatial harmonics to include in the expansion is arbitrary.
- Improper choice can lead to slow convergence and inaccurate results.
- We chose directions consistent with the physics of diffraction and a rectangular Fourier-space grid for simplicity.
- The number of harmonics retained in a particular direction determines the spatial resolution of structures with contrast in that direction.
- It seems optimal to keep the number of spatial harmonics uniform in all directions.
Fourier-Space Grid Notation

The components of the wave vector expansion look like:

\[ \hat{k}_x (m) \quad \hat{k}_y (n) \quad \hat{k}_z (m, n) = \left( \sqrt{\mu' \varepsilon' - \hat{k}_z^2 (m)} - \hat{k}_z (m) \right) \]

We visualized it this way:

A simpler view of our 2D Fourier-space grid is:

---

Simple Grid Truncation Scheme

We will retain all spatial harmonics with indices that satisfy the following equation:

\[ \frac{m^2 \gamma}{M} + \frac{n^2 \gamma}{N} \leq 1 \]

Conventional \( |n| \leq N \) and \( |m| \leq M \)

Implementation

Step 1 – Build Unit Cell on High-Resolution Grid

Implementation

Step 2 – Calculate Fourier Coefficients Using FFT Technique
Implementation

Step 3 – Assemble Standard Convolution Matrix From Fourier Coefficients

\[ \mathbf{e} = \begin{bmatrix} \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \]

Implementation

Step 4 – Build Truncation Map

```
% Construct Truncation Map
TMAP = abs(m/M).^(2*p) + abs(n/N).^(2*p);
TMAP = (TMAP <= 1);
```
Implementation

**Step 5 – Extract Indices of Spatial Harmonics to Retain**

% Construct Truncation Map
TMAP = abs(m/M).^(2*p) ...
+ abs(n/N).^(2*p);
TMAP = (TMAP <= 1);

% Extract Array Indices
ind = find(TMAP(:));

---

Implementation

**Step 6 – Truncate Convolution Matrix**

% Truncate Convolution Matrix
ERCT = ERC(ind,ind);
Implementation

Step 7 – Perform RCWA

The rest of the RCWA algorithm remains virtually unchanged.

You will need to consider your truncation again:

1. When you calculate the source.
2. If you calculate the fields from the eigen-modes.

RCWA for Generalized Symmetries
Revised Fourier Transforms

The materials...

\[ e_r (\vec{r}) = \sum_{m=\infty}^{\infty} \sum_{n=\infty}^{\infty} a_{mn} e^{i(\varphi^{mn} + \varphi^{mn} \epsilon_i)} \]

\[ a_{mn} = \frac{1}{A} \oint_{\text{cell}} e_r (\vec{r}) e^{-i(\varphi^{mn} + \varphi^{mn} \epsilon_i)} d\vec{r} \]

\[ \mu_r (\vec{r}) = \sum_{m=\infty}^{\infty} \sum_{n=\infty}^{\infty} \beta_{mn} e^{i(\varphi^{mn} + \varphi^{mn} \epsilon_i)} \]

\[ \beta_{mn} = \frac{1}{A} \oint_{\text{cell}} \mu_r (\vec{r}) e^{-i(\varphi^{mn} + \varphi^{mn} \epsilon_i)} d\vec{r} \]

The fields...

\[ E_x (\vec{r}, z) = \sum_{m=\infty}^{\infty} \sum_{n=\infty}^{\infty} S_x (m,n,z) \cdot e^{-i(m \varphi_1 + n \varphi_2)} \]

\[ H_x (\vec{r}, z) = \sum_{m=\infty}^{\infty} \sum_{n=\infty}^{\infty} U_x (m,n,z) \cdot e^{-i(m \varphi_1 + n \varphi_2)} \]

Wave vector expansion...

\[ \vec{k} (m,n) = \vec{k}_0 - m \vec{T}_1 - n \vec{T}_2 \]

\[ m = -\infty, \ldots, -2, -1, 0, 1, 2, \ldots, \infty \]

\[ n = -\infty, \ldots, -2, -1, 0, 1, 2, \ldots, \infty \]

\[ \vec{k}_1 = \left( \sqrt{k_0^2 \mu_0^2 - \vec{k} (m,n)^2} \right)^\righthalfarrow \]

\[ \vec{T}_1, \vec{T}_2 = \text{reciprocal lattice vectors of the unit cell} \]

Modeling Hexagonal Gratings with Rectangular RCWA
Geometry of a Hexagon

![Hexagon Diagram]

Grating Vectors of Hexagonal Structures

Direct Lattice

Reciprocal Lattice

\[ \vec{t}_1 = \frac{a}{2} \hat{x} - \frac{a\sqrt{3}}{2} \hat{y} \]

\[ \vec{t}_2 = \frac{a}{2} \hat{x} + \frac{a\sqrt{3}}{2} \hat{y} \]

\[ \vec{t}_3 = c \hat{z} \]

\[ \vec{\tilde{t}}_1 = \frac{2\pi}{a} \hat{x} - \frac{2\pi}{a\sqrt{3}} \hat{y} \]

\[ \vec{\tilde{t}}_2 = \frac{2\pi}{a} \hat{x} + \frac{2\pi}{a\sqrt{3}} \hat{y} \]

\[ \vec{\tilde{t}}_3 = \left(\frac{2\pi}{c}\right) \hat{z} \]
We must identify a rectangular unit cell that reconstructs a hexagonal array.

This implies that we will need more spatial harmonics along the $y$ direction than $x$.

$$Q = \text{round}(P*Sy/Sx);$$

Enhanced Transmittance Matrix Approach

**Motivation**

The enhanced transmittance matrix (ETM) method involves less matrix manipulations so it is much faster than using scattering matrices. Maybe $\sim 10\times$. It also provides easier computation of internal fields.

ETM works by first stepping backward through each layer (backward analysis) and then stepping forward (forward analysis). Intermediate parameters must be stored for each layer during the backward analysis that are recalled during the forward analysis. This leads to severe memory limitations when many layers are used.

**Conclusion**  

Unless the device is composed of prohibitively large number of layers or if other features of scattering matrices are not needed, use ETM.

---

**The Problem**

The source of the instability is the following matrix.

$$X_i^+ = e^{-\Omega x_i},$$

$$X_i^- = e^{\Omega x_i}, \quad \text{Growing exponentials!!!}$$

The enhanced transmittance matrix (ETM) method was the first technique applied to RCWA that fixed the instability.

ETM is much faster than scattering matrices, but is much less memory efficient because it requires parameters to be stored for all layers at the same time. Scattering matrices can proceed one layer at a time, forgetting everything about previous layers.
Boundary Conditions

First interface:
\[ s + A \mathbf{r} = F_i \begin{bmatrix} 1 & 0 \\ 0 & X_i \end{bmatrix} \mathbf{c}_i \]

Intermediate interfaces:
\[ F_i \begin{bmatrix} X_j & 0 \\ 0 & 1 \end{bmatrix} \mathbf{c}_j = F_{i+1} \begin{bmatrix} 1 & 0 \\ 0 & X_{i+1} \end{bmatrix} \mathbf{c}_{i+1} \]

Last interface:
\[ F_N \begin{bmatrix} X_N & 0 \\ 0 & 1 \end{bmatrix} \mathbf{c}_N = B \mathbf{t} \]

Work Backward Through Layers (1 of 4)

The goal is to solve for \( \mathbf{r} \) and \( \mathbf{t} \) without using \( \mathbf{X}^{-1} \).

We start by solving the equation at the last interface for \( \mathbf{c}_N \):
\[ F_N \begin{bmatrix} X_N & 0 \\ 0 & 1 \end{bmatrix} \mathbf{c}_N = B \mathbf{t} \rightarrow \mathbf{c}_N = \begin{bmatrix} X_N^{-1} & 0 \\ 0 & 1 \end{bmatrix} F_N^{-1} B \mathbf{t} \]

We write this as
\[ \mathbf{c}_N = \begin{bmatrix} X_N^{-1} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} a_N \\ b_N \end{bmatrix} = a_N \begin{bmatrix} X_N^{-1} \\ b_N \end{bmatrix} = F_N^{-1} B \]
To eliminate the potentially ill conditioned matrix $X^{-1}$, we introduce an intermediate transmittance matrix parameter $t_N$ defined as

$$t = a_N^{-1}X_N t_N$$

$t$ and $t_N$ remain unknown.

Our equation for $c_N$ becomes

$$c_N = \begin{bmatrix} X^{-1}_N & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} a_N \\ b_N \end{bmatrix} t_N \Rightarrow c_N = \begin{bmatrix} I \\ b_N a_N^{-1}X_N \end{bmatrix} t_N$$

The boundary condition equation at the second-to-last interface is

$$F_{N-1} \begin{bmatrix} X_{N-1}^{-1} & 0 \\ 0 & I \end{bmatrix} c_{N-1} = F_N \begin{bmatrix} I & 0 \\ 0 & X_N \end{bmatrix} c_N$$

Substituting our expression for $c_N$ into the equation yields

$$F_{N-1} \begin{bmatrix} X_{N-1}^{-1} & 0 \\ 0 & I \end{bmatrix} c_{N-1} = F_N \begin{bmatrix} I & 0 \\ 0 & X_N \end{bmatrix} \begin{bmatrix} I \\ b_N a_N^{-1}X_N \end{bmatrix} t_N$$

Solving this for $c_{N-1}$ leads to

$$c_{N-1} = \begin{bmatrix} X_{N-1}^{-1} & 0 \\ 0 & I \end{bmatrix} F_{N-1}^{-1} F_N \begin{bmatrix} I & 0 \\ 0 & X_N \end{bmatrix} \begin{bmatrix} I \\ b_N a_N^{-1}X_N \end{bmatrix} t_N$$

We have now worked backward by one layer while avoiding $X_N^{-1}$.
Work Backward Through Layers (4 of 4)

This process continues through all the layers.

\[
c_{N-1} = \begin{bmatrix} X_{N-1}^{-1} & 0 \\ 0 & I \end{bmatrix} F_{N-1}^{-1} F_N \begin{bmatrix} 1 & 0 \\ 0 & X_N \end{bmatrix} \begin{bmatrix} I \\ b_N a_N^{-1} X_N \end{bmatrix} t_N
\]

\[
c_{N-1} = \begin{bmatrix} X_{N-1}^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} a_{N-1} \\ b_{N-1} \end{bmatrix} t_N
\]

\[
\begin{bmatrix} a_{N-1} \\ b_{N-1} \end{bmatrix} = F_{N-1}^{-1} F_N \begin{bmatrix} 1 & 0 \\ 0 & X_N \end{bmatrix} \begin{bmatrix} I \\ b_N a_N^{-1} X_N \end{bmatrix}
\]

\[
t_N = a_{N-1}^{-1} X_{N-1} t_{N-1} \quad \text{T}_{N-1} \text{ and } T_N \text{ remain unknown.}
\]

Solve for Reflected and Transmitted Fields

After working through all interfaces, we are left with

\[
s + A r = F_i \begin{bmatrix} 1 & 0 \\ 0 & X_i \end{bmatrix} \begin{bmatrix} I \\ b_i a_i^{-1} X_i \end{bmatrix} t_i
\]

We solve this matrix equation for \( r \) and \( t_i \).

\[
\begin{bmatrix} r \\ t_i \end{bmatrix} = \left[ -A \quad B' \right]^{-1} S \quad B' = F_i \begin{bmatrix} 1 & 0 \\ 0 & X_i \end{bmatrix} \begin{bmatrix} I \\ b_i a_i^{-1} X_i \end{bmatrix} t_i
\]

Now that we know \( t_i \), we can work forward through the layers to solve for \( t \).

\[
t_2 = a_1^{-1} X_1 t_1
\]

\[
t_3 = a_2^{-1} X_2 t_2
\]

\[
\vdots
\]

\[
t_N = a_{N-1}^{-1} X_{N-1} t_{N-1}
\]

\[
t = a_N^{-1} X_N t_N
\]

Note: we must store \( a \) and \( X \) for each layer. This leads to poor memory efficiency.
Calculating the Diffraction Efficiencies

First, we calculate the longitudinal field components.

\[ r_z = -\tilde{K}^{-1}_{z,\text{ref}} \left( \tilde{K}_r r_x + \tilde{K}_y r_y \right) \quad \quad t_z = -\tilde{K}^{-1}_{z,\text{ref}} \left( \tilde{K}_r t_x + \tilde{K}_y t_y \right) \]

Second, we calculate the diffraction efficiencies.

\[ R = \frac{\text{Re} \left[ -\tilde{K}_{z,\text{ref}} / \mu_{r,\text{inc}} \right] }{\text{Re} \left[ k_{z,\text{inc}} / \mu_{r,\text{inc}} \right]} |\vec{r}|^2 \quad |\vec{r}|^2 = |r_x|^2 + |r_y|^2 + |r_z|^2 \]

\[ T = \frac{\text{Re} \left[ \tilde{K}_{z,\text{im}} / \mu_{r,\text{inc}} \right] }{\text{Re} \left[ k_{z,\text{inc}} / \mu_{r,\text{inc}} \right]} |\vec{t}|^2 \quad |\vec{t}|^2 = |t_x|^2 + |t_y|^2 + |t_z|^2 \]

---

Block Diagram of ETM

1. Dashboard: Source, device, external regions, & RCWA
2. Build Device on High-Res Grid
3. Compute Convolution Matrices ER and UR
4. Compute Wave Vectors kinc, kx, ky, kzref, & kztrn
5. Construct Diagonal K Matrices Kx, Ky, Kzref, & Kztrn
6. Calculate Source Vector \( s = [sx \ sy \ ux \ uy]^T \)
7. Calculate A and B for External Regions
8. Work Backward Through All Layers
   - Build Eigen-Value Problem \( P, Q, \& \Omega_2 \)
   - Solve Eigen-Value Problem \( W, V, \& \LAM \)
   - Compute Layer Parameters F, X, a, and b
   - Update B
9. Compute \( r_x, r_y, \) and \( t \)
10. Work Forward Through All Layers
    - Extract \( tx \) and \( ty \)
    - Compute \( r_z \) and \( t_z \)
    - Compute Diffraction Efficiencies R and T
    - Calculate Overall REF and TRN
    - Verify Conservation
11. Finished!