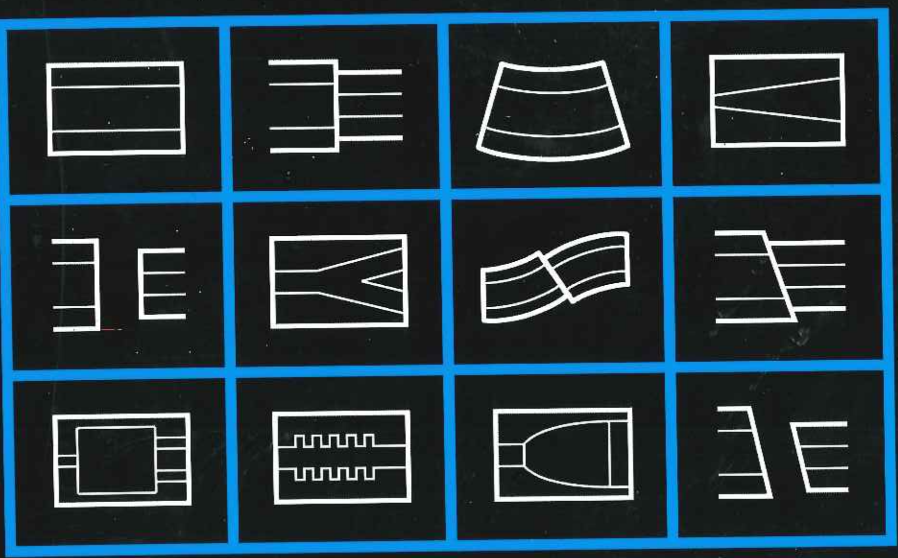


PHOTONICS CAD TOOLKITS

Photon Design®

FIMMWAVE

an introductory tutorial

Photon Design®

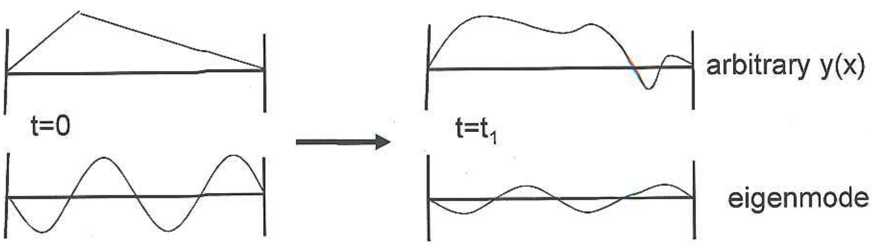
Introduction

- FIMMWAVE is a waveguide *mode solver*
- ***mode*** - (eigenmode) a solution of Maxwell's Equations in a *z-invariant* refractive index profile of the form:

$$\Phi(x, y, z) = \psi_m(x, y) \cdot e^{i\beta z}$$

β: *propagation constant* of mode

Mode of violin string:



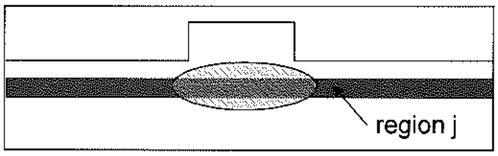
Photon Design®

Some Properties of Modes

n_{eff}: *effective index* of mode

$$n_{eff} = \beta / k_0 \qquad k_0 = 2\pi / \lambda$$

Γ: *confinement factor*



define Γ:
mode loss $\alpha_{mode} = \Gamma_j \cdot \alpha_{material,j}$

- modes are *orthogonal*:

$$\int \phi_i(x,y)\, \phi_k(x,y)\, dx\, dy \quad = 1, \text{ if } i=k \qquad = 0, \text{ if } i \neq k$$

- This implies that if mode *i* is propagating down a straight waveguide, power will never couple to mode *k*.
- modes travel at different speeds: $v = c/n_{eff}$

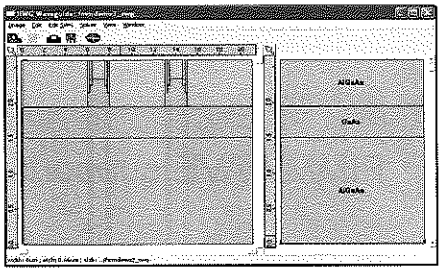
"Fill factor" is power in region
Different than confinement factor

Photon Design®

The Waveguide Editors

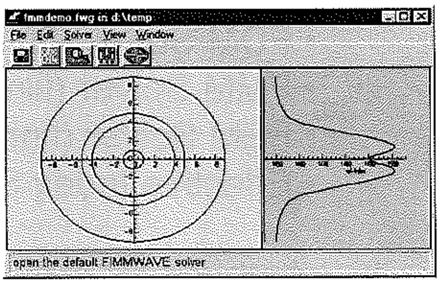
Photon Design®

Waveguide Geometries



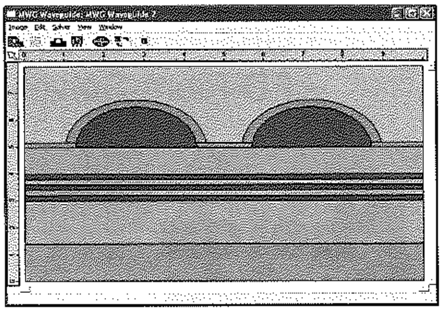
RWG: Rectangular geometry

epitaxially grown waveguides



FWG: Cylindrical geometry

radial index profile: RIX(r)



MWG: General geometry

rectangles + ellipses + polygons
graded RIX profiles

Photon Design®

The Mode Solvers

Photon Design®

Mode Solvers: an overview

FIMMWAVE supports different mode solvers

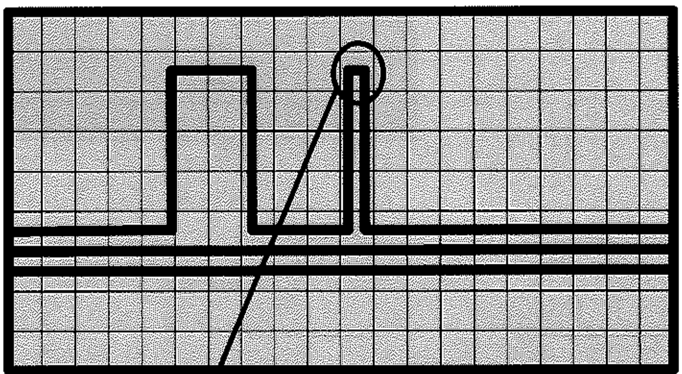
In Cartesian co-ordinates:
- FDM Solver (finite difference)
- FMM Solver (film mode matching)
- FEM Solver (finite element)
- Effective Index Solver

In cylindrical co-ordinates:
- GFS Solver (Bessel functions) - semi analytic w/ Bessel
- FD Fibre Solver (finite difference)

Photon Design®

The FDM Solver: features

- fully vectorial
- full diagonal dielectric tensor (anisotropy) $\begin{pmatrix} \epsilon_{xx} & 0 & 0 \\ 0 & \epsilon_{yy} & 0 \\ 0 & 0 & \epsilon_{zz} \end{pmatrix}$
- good PMLs   light @ boundary
- finds many modes at same time
- guaranteed not to miss any modes (to our knowledge!)
- real and complex versions
- bend mode version
- symmetry planes available
- uniform mesh spacing:
can be used with any geometry

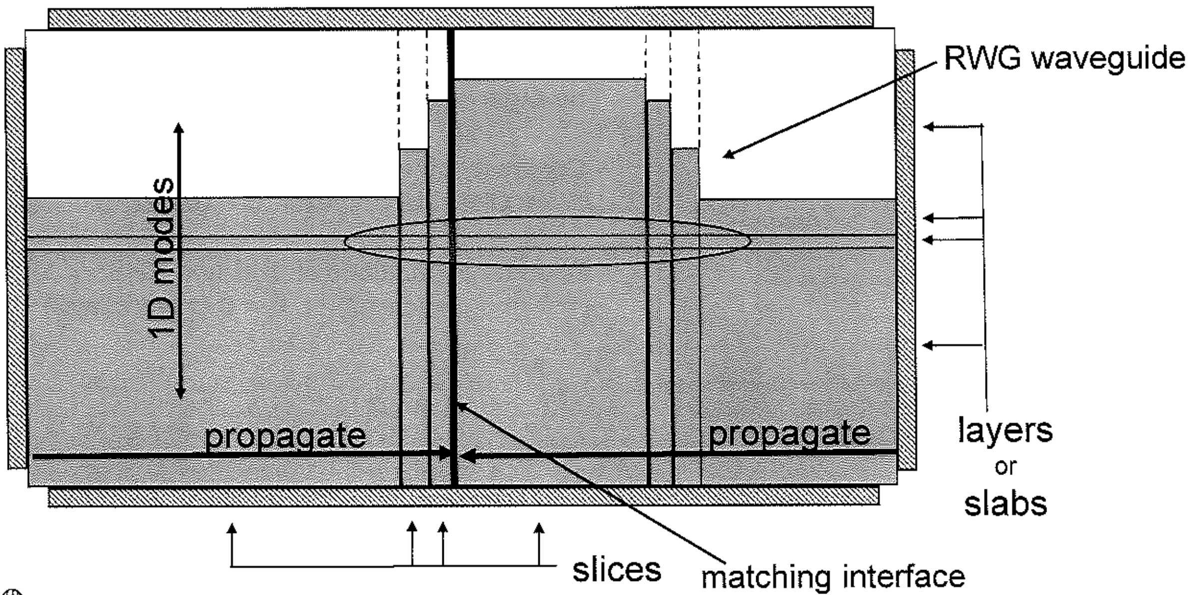


Takes account of structure smaller than grid squares – not just averaging

Photon Design®

The FMM Solver

Suitable for rectangular geometries



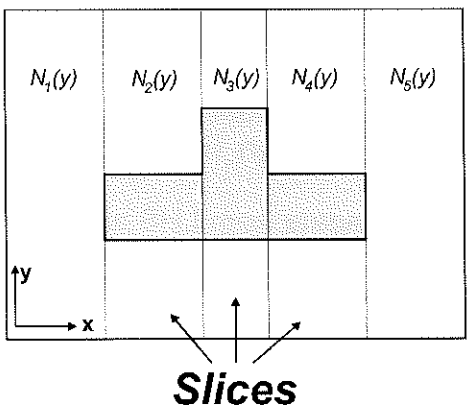
Photon Design®

7 slices, invariant in x direction within slab
vertical structure

Propagate trial β from B.C. to matching edge

The FMM Solver: theory

Assume RIX profile N(x,y) is **piecewise constant in x direction**.



In slice m, the field can be expressed as a linear combination of 1d TE and TM modes

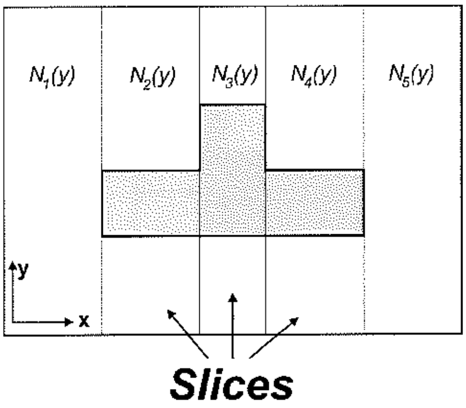
$$\{e,h\} = \sum \left[u_j^{TE} \exp(ik_{xj}^{TE})\phi_j^{TE}(y,\beta_{yj}^{TE}) + u_j^{TM} \exp(ik_{xj}^{TM})\phi_j^{TM}(y,\beta_{yj}^{TM}) \right]$$

Not necessarily travelling in z direction

Photon Design®

The FMM Solver: theory

The 2D mode is generated by “propagating” the vertical modes in the x-direction.



The eigensystem is obtained using:

- Continuity of $\mathbf{e}_t$, $\mathbf{h}_t$ at slice interfaces :
$\mathbf{U}^{m+1} = \mathbf{O}^{m,m+1}\,\mathbf{U}^{m}$, with $\mathbf{U} = \{\mathbf{u}^{TE}, \mathbf{u}^{TM}\}$

- Boundary conditions at LHS and RHS:
$\mathbf{e}_t = 0$ (magnetic wall) or $\mathbf{h}_t = 0$ (electric wall)

Photon Design®

The FMM Solver: algorithm

- Find all (N) TE and TM 1D modes for each slice
- Build overlap matrices between 1D modes at each slice interface
- Guess start beta
- Using given beta, propagate from LHS and RHS to the matching interface (near the middle)
- Generate error function at matching interface
- Loop (varying beta) until error is small
- When the error is zero: mode found!

Photon Design®

The FMM Solver: features

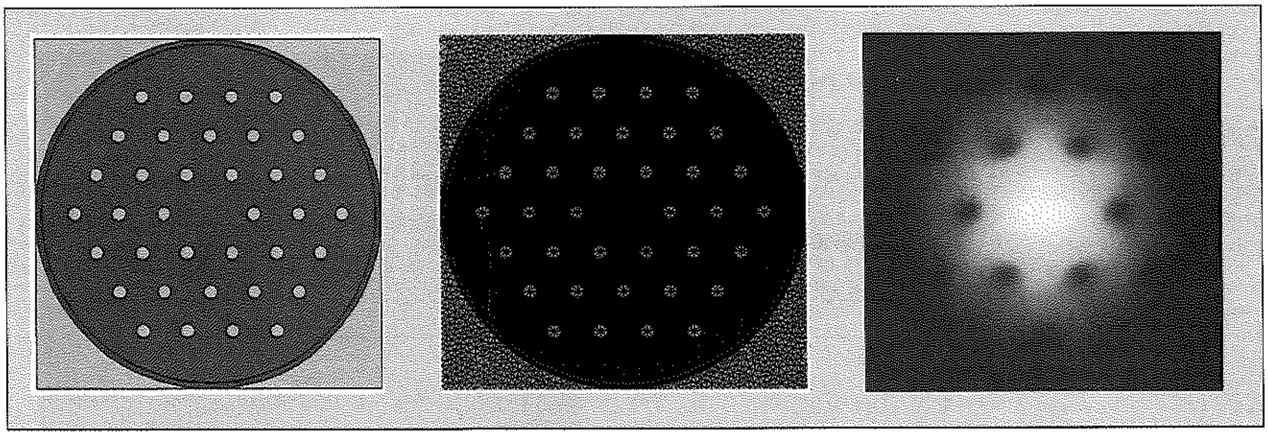
- fully vectorial
- real and complex versions
- bend mode version (rigorous cylindrical co-ordinates)
- PMLs
- symmetry planes available

- very fast if you need only a few eigenmodes
- exact solution to Maxwell as N_{1d} tends to infinity
- N_{1D} typically small - 20 to 50
- ideal for epitaxial devices with natural rectangular geometry.
- very accurate for planar structures - used as benchmark for other methods

Not a mesh, good for planar

Photon Design®

The FEM Solver



- fully vectorial
- complex solver
- adaptive unstructured mesh
- good for tilted or curved interfaces
- good for graded index structures
- mesh can be inspected - Engine Info
- guaranteed not to miss modes (to our knowledge!)
- bend mode version (conformal transformation)

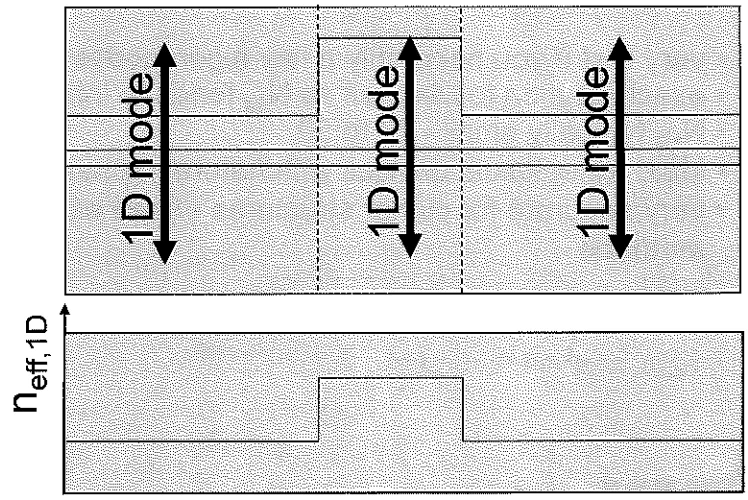
Photon Design®

The Effective Index Solver

How it works:

1. Finds vertical 1D modes in each slice
2. Builds horizontal 1D waveguide based on effective index of vertical modes
3. Solves horizontal 1D waveguide

- Approximation: $\psi(x,y) = A(x).B(y)$
- Should be used for:
  - 1D+Z calculations
  - small index contrast Δn
  - structures with w>>h or h>>w
- Very fast (both real and complex)



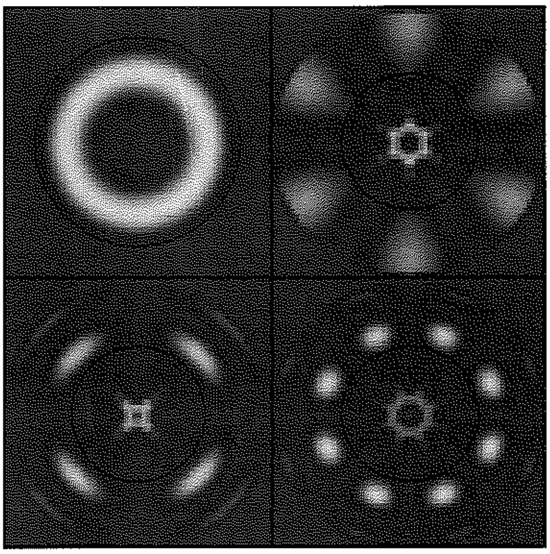
Photon Design®

Fibre Solvers (Cylindrical Geometry)

FWG waveguide:
radial index profile: $n = n(r)$

Two choices:

- GFS Solver (Bessel functions)
- FD Fibre Solver (finite differences)



Photon Design®

Fibre Solvers (Cylindrical Geometry)

- Solutions of the form:

$\Phi(\theta,r) = \psi(r)\cos(m.\theta)$,   p-order=1, m: m-order

$= \psi(r)\sin(m.\theta)$,   p-order=2

m-order and p-order are "quantum numbers"

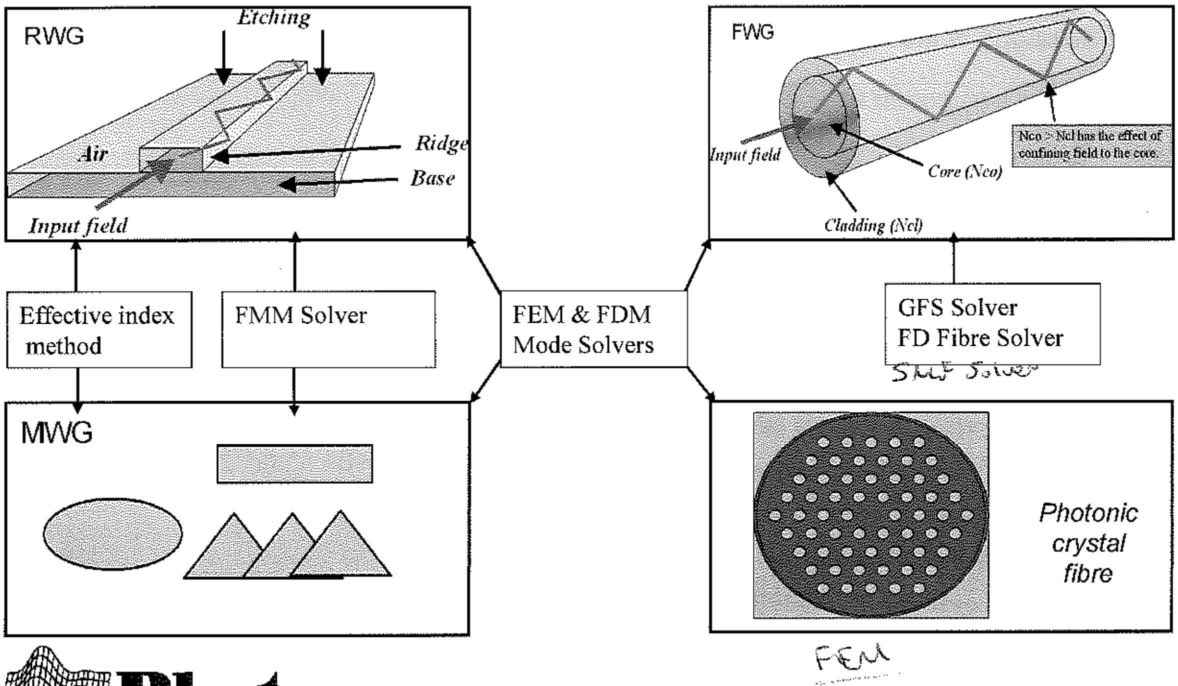
- Different p-order and m-order modes are **orthogonal**
- Features:
  - → arbitrary n(r)
  - → full vectorial formula (also semi-vec versions)
  - → very fast (2D reduced to 1D)
  - → real and complex versions for FDFS, GFS has real version only
  - → PMLs (FDFS only)
  - → 1D overlaps in FIMMPROP (if symmetry maintained at joint)

Photon Design®

Which Mode Solver?

Photon Design®

Waveguide structures and Mode Solvers



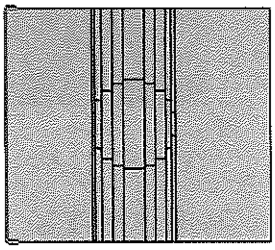
Photon Design®

General advice

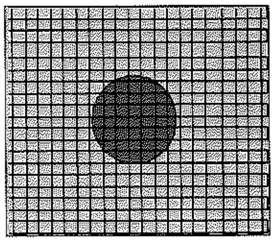
- For the general 2D+Z case, usually we would recommend that you use the FDM Solver as a start-point solver. You can then try to improve accuracy by switching to another solver (FMM or FEM). In some simple low delta-n structures the Effective Index Solver can give you even faster initial numbers.
- For FIMMPROP simulations, start with FDM Solver unless you have good reason to do otherwise, then check with other solvers if possible.
- If you have a planar structure and just need a few real modes, then FMM will give you the best accuracy.
- In the 1D+Z case use the Effective Index Solver in Quasi2D (XZ) or Quasi2D (YZ) mode.

Photon Design®

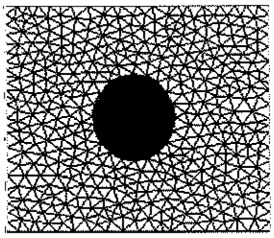
Shape discretisation



FMM: Has to use a staircase approximation for slanted or curved interfaces.



FDM: uses advanced techniques to better deal with slanted and curved interfaces but still (small) stairs so not as accurate as FEM.



FEM: can deal with interfaces at any angle and curved interfaces just as easily as horizontal or vertical interfaces.

Comment
For slanted or curved interfaces, FEM best, then FDM, then FMM.

Photon Design®

Rectangular structures

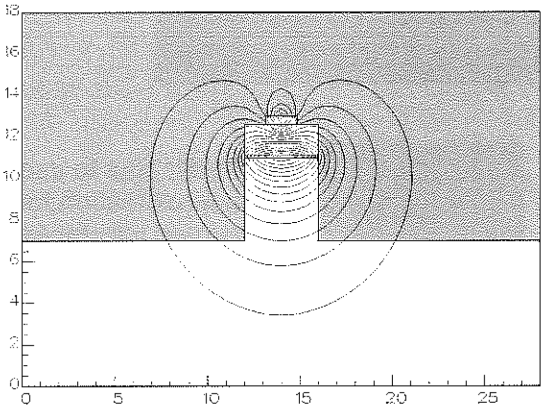
FMM: Uses semi-analytic methods that can provide high accuracy when used on waveguides within their domain of validity.

FDM/FEM: FDM and FEM will never be a match in accuracy for semi-analytical techniques where such techniques are used within their domain of competence.
↳ rectangular structures

Comment
Thus, you are likely to get higher accuracy with the FMM Solver for waveguides that can be described by a relatively small number of uniform rectangles, such as en epitaxially grown/etched ridge waveguide.

Photon Design®

Modelling very thin layers



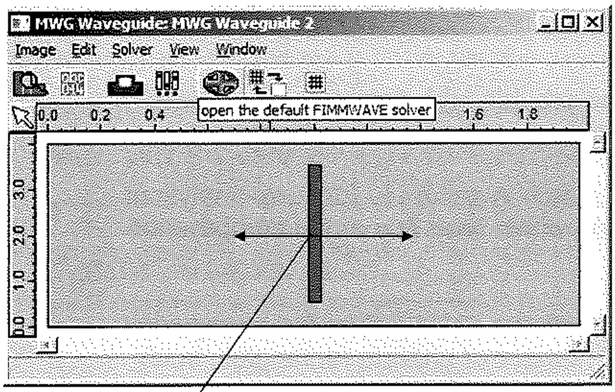
FMM: ideal choice
- No discretisation
- Very thin layers simply require higher number of 1D modes

FEM: 2nd choice, or if there are additional features FMM cant deal with
- Adaptive mesh
- Computation time increases at a rate approximately $(\text{nodes})^{1.5}$.

FDM:
- Uniform mesh but advanced thin layer modelling technique
- Computation time increases at a rate approximately $(\text{nodes})^{1.5}$
- Still needs a high resolution to model thin layers really accurately.

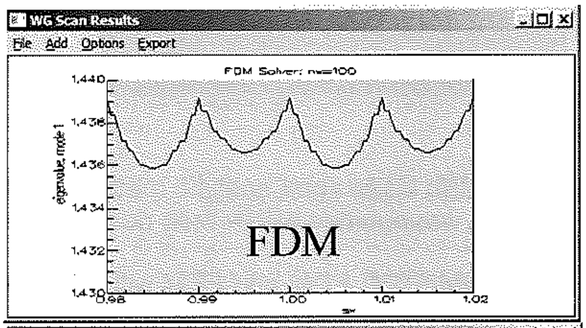
Photon Design®

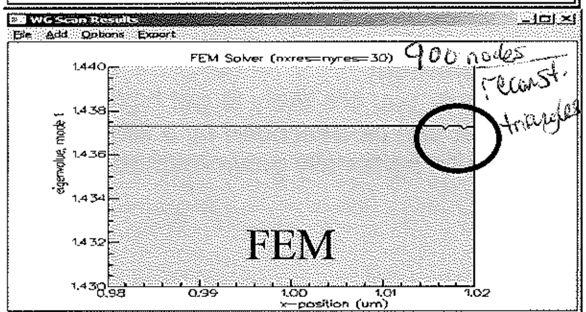
Modelling very thin layers

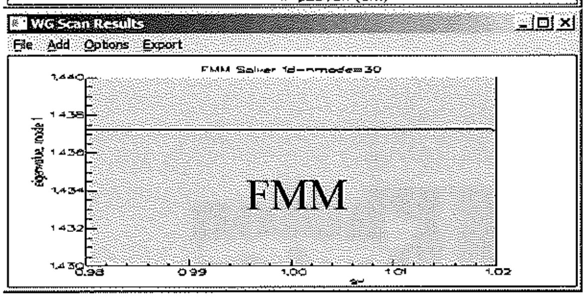


Move thin slice across waveguide
Slice width 50 nm, delta-n = 2.5

FDM sample 10nm – gives accuracy in neff of ~0.1% (worst case scenario). Raw sampling gives 20%.







Photon Design®

Complex modes

The complex version of the mode solvers is needed to model:
- material absorption
- metallic structures (surface plasmons)
- bend modes
- optical leakage
- optical radiation (FIMMPROP)

FMM:
Complex version much slower than the real.
Does not guarantee to find all the modes (iterative).

FDM/FEM:
Find complex modes in about 4x the time of real solvers
More robust than FMM Solver: guarantee to find the modes.

Comment
Thus if you want to model complex index waveguides, the FEM and FDM Solvers will be faster and more reliable. In general, it is not practical to use the Complex FMM Solver in FIMMPROP for reliability reasons.

Photon Design®

Modelling metallic structures

FMM: FMM Solver uses basis set of 1D modes which can converge slowly when metals are present, unless the metal extends as a flat plane through all slices. trouble if metal splitting/decoupling structure

FEM/FDM: use a local basis set (the mesh points)

Comment
FMM is not good at modelling structures where regions are decoupled from each other, e.g. by the presence of thick metals. FDM and FEM do not have this problem and can model decoupled structures and those containing large amounts of metal.

Photon Design®

Boundary conditions

Each wall has a boundary condition.

All Cartesian solvers support:
- metal wall - $E_{//} \rightarrow 0$
- magnetic wall - $H_{//} \rightarrow 0$
- PML – perfectly matched layers

FMM/FDM support:
- periodic wall - $\underline{E}(0) = \underline{E}(L)$, $\underline{H}(0) = \underline{H}(L)$
- impedance wall - perfectly matched for one particular angle of incidence or mode beta (FMM: only top and bottom)
- transparent - useful for bends

Photon Design®

Absorbing boundary conditions

FMM:
PMLs on 4 sides but top and bottom PMLs mess up the basis set of the FMM method if too big – narrow range of operation.
Transparent boundary conditions can be used through polishing (not with MOLAB).

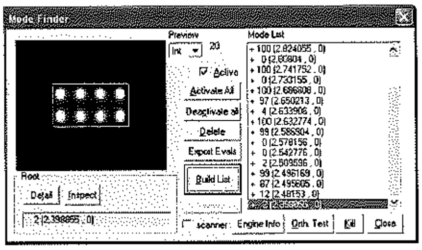
FDM:
PMLs on 4 sides are well behaved.
Transparent boundary conditions can be used with MOLAB. maybe w/ bend mode solver?

FEM:
PMLs on 4 sides are well behaved.
No transparent boundary conditions.

Comment
If you have radiation going up and/or down, then FDM/FEM are easier to use.

Photon Design®

Finding multiple modes



Looking for a large number of modes?

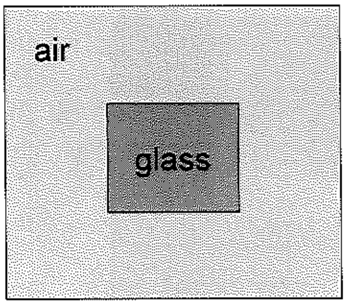
FEM and FDM
- will find many modes at the same time (fast!)
- will guarantee to find all the modes

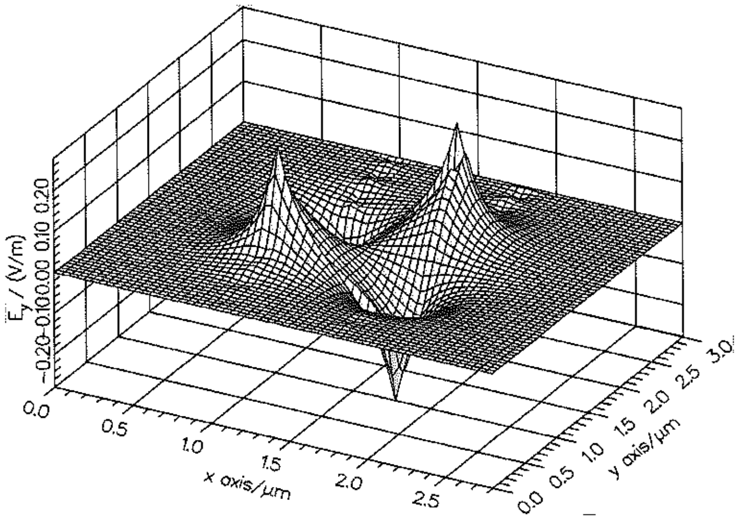
FMM
- finds one mode at a time
- may miss some modes
- can benefit from polishing in taper calculations (EVScan)

Comment
The FEM and FDM Solvers may be faster and more reliable when you require a large number of modes.

Photon Design®

Vectorial & Anisotropy





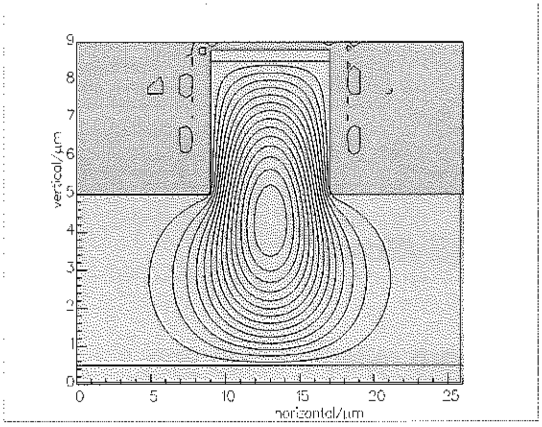
All solvers are fully vectorial.

Anisotropy:
FMM: two term diagonal tensor $\varepsilon_{xx}=\varepsilon_{zz}$
FDM/FEM: 3 term diagonal tensor

Photon Design®

High Δn waveguides

A Si/SiO_2 (SOI) waveguide



FMM, FDM and FEM solvers can model SOI waveguides.

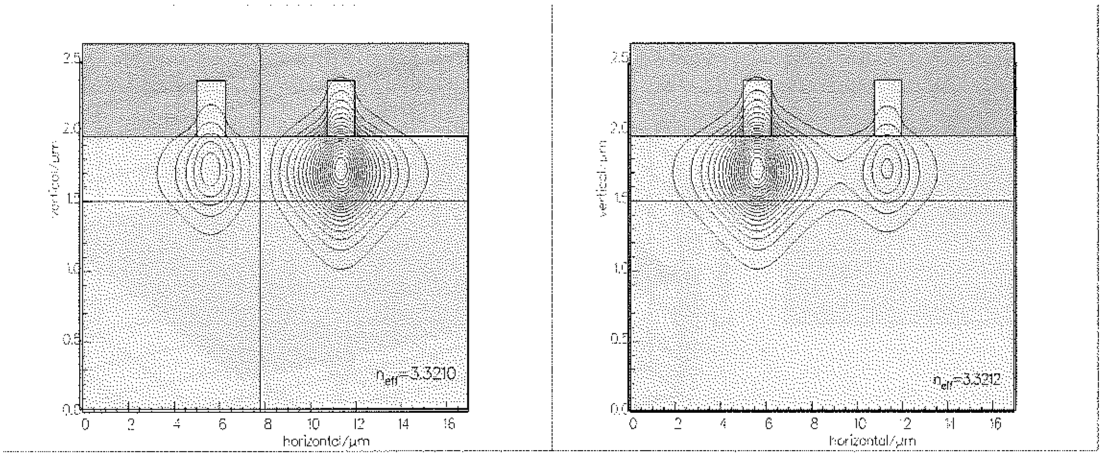
FMM and FEM are most accurate.

Vectorial versions of the solvers give best results with high index contrast.

Effective Index Solver is not valid.

Photon Design®

Weakly coupled (horizontally) waveguides



FMM: can model weakly coupled waveguides but not fully decoupled ones – might miss a mode.

FDM/FEM: can model fully decoupled waveguides.

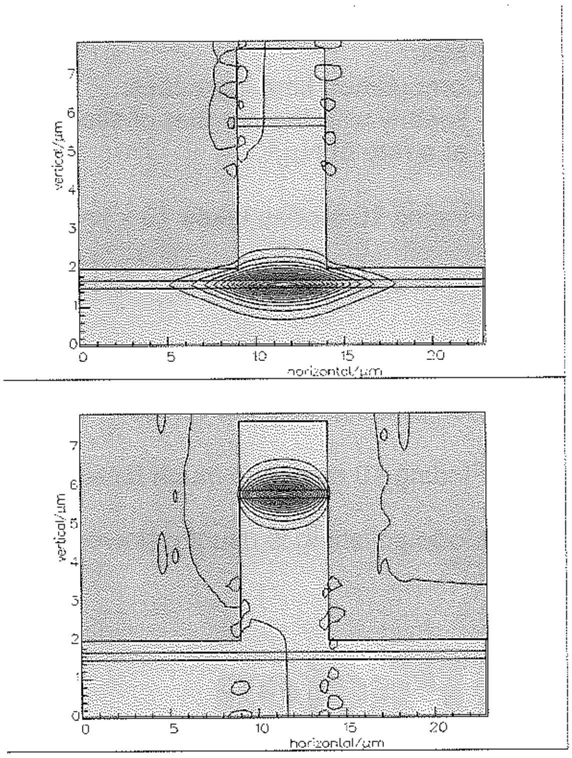
Photon Design®

Vertically decoupled waveguides

Handled by FMM, FDM and FEM Solvers.

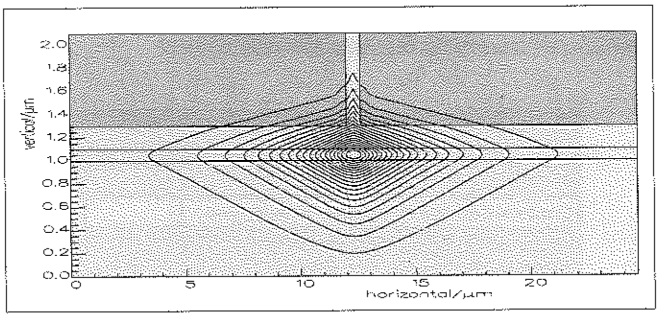
But beware - is this useful?

Why not solve the decoupled waveguides separately?
⇨ much faster



Photon Design®

Near cut-off modes



Modes near cut-off expand strongly and usually need a large computational window.

FMM:
No discretisation: slice width does not impact speed or resolution.
Can efficiently deal with modes that extend very far in the lateral direction.

FDM:
Uniform rectangular grid.
Increasing the window at a constant grid size makes the FDM Solver rather slower.

FEM:
The mesh size can be adjusted.
The increase in computation time is not as bad as with FDM.

Photon Design®

Comment
You may find the FMM Solver much more efficient for modelling modes that extend a long way from the waveguide.

A few words on computation time

FMM:
Computation time increases:
- linearly with the number of slices
- at a rate of N_{1D}^3

Finding 10 modes takes 10 x time needed for to find one mode.

Mode polishing is very fast and can be used to increase N_{1D} efficiently.

FDM/FEM:
Computation time increases at a rate of approximately $(\text{nodes})^{1.5}$.

Finding 10 modes is much faster than ten times the time to find one mode.

Mode polishing is usually slower than calculating the modes from scratch.

Photon Design®

Mode Solver Feature Table

Feature	FMM Solver	FDM Solver	FEM Solver	Effective Index Solver	FDM Fibre Solver	General Fibre Solver
Rectangular Geometry Waveguides	+++	+++	+++	++		
Cylindrical Geometry Waveguides – arbitrary Ne(r)	+	+	++		+++	+++
Mixed Geometry Waveguides (ellipses and polygons)	+	++	+++	+		
Graded index waveguides including diffused waveguides	+	+++	+++	++	++	++
Metals, gainy or absorbing waveguides (with Complex Engine option)	++	+++	+++	++	+++	
Anisotropic refractive index	++	++	++			
Bend modes	+++	++	++			
Compatibility with PMLs (with Complex Engine option)	++	+++	++	++	++	
Transparent boundary conditions (with Complex Engine option)	++	++				++
Periodic boundary conditions	+++	+++		+++		
Compatibility with FIMMPROP	+++	+++	+++	+++	+++	+++
Very thin layers	+++	++	++	+++	+++	+++

+++ : full support – best performance
++ : full support – fair performance
+ : some capability

Photon Design®

FIMMWAVE utilities

Photon Design®

The WG Scanner

A) Scan between two waveguides

Waveguide A: set of parameters $\{p_{1,a}, p_{2,a}, p_{3,a}, p_{4,a} \cdots \}$
Waveguide B: set of parameters $\{p_{1,b}, p_{2,b}, p_{3,b}, p_{4,b} \cdots \}$

An intermediate waveguide (e.g. 10% A, 90% B):
weighting parameter w = 0.1

Then the intermediate waveguide is given by:

$$p_j = (1-w).p_{j,a} + w.p_{j,b}$$

B) Scan a Parameter

Photon Design®

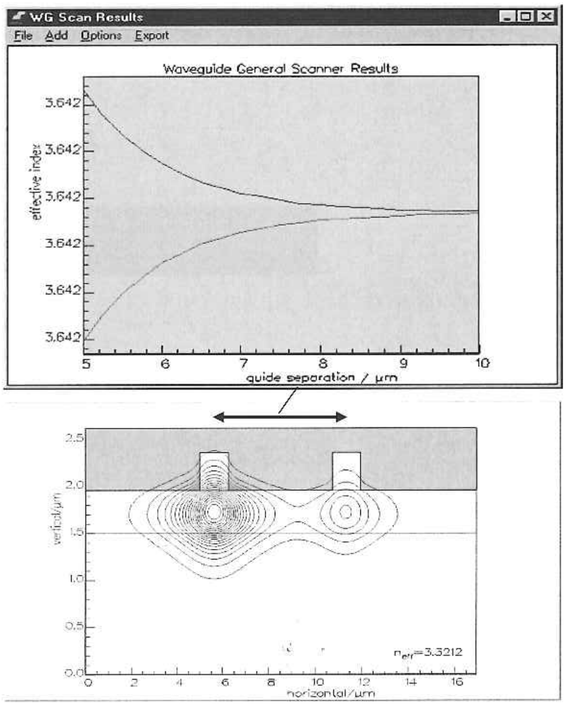
The WG Scanner

- can generate design curves versus any parameter
- very fast algorithm

Quantities:

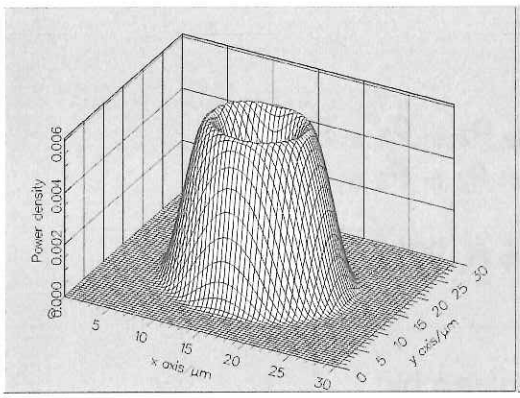
- n_{eff} and β
- confinement factor
- filling factor
- mode loss
- etc.

symmetric profile

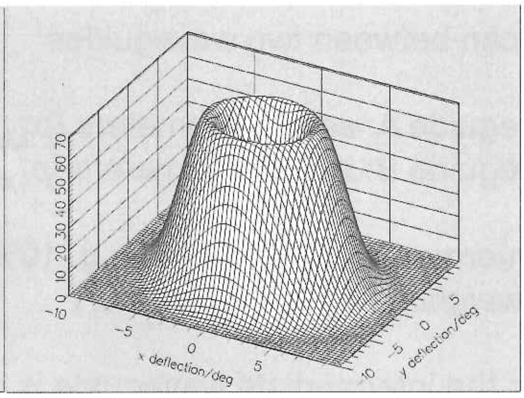


Photon Design®

The Far Field Calculator



"nearfield"



"far-field"

- fully vectorial formula
- aperture function

Photon Design®

The Far Field Calculator

Stage 1: build k-space profile

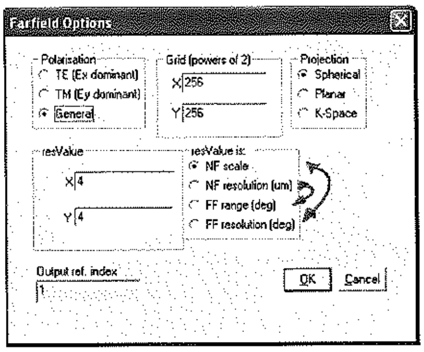
- FF resolution proportional to NF range/Scale
- FF range proportional to NF resolution
- NF range = waveguide-size x Scale

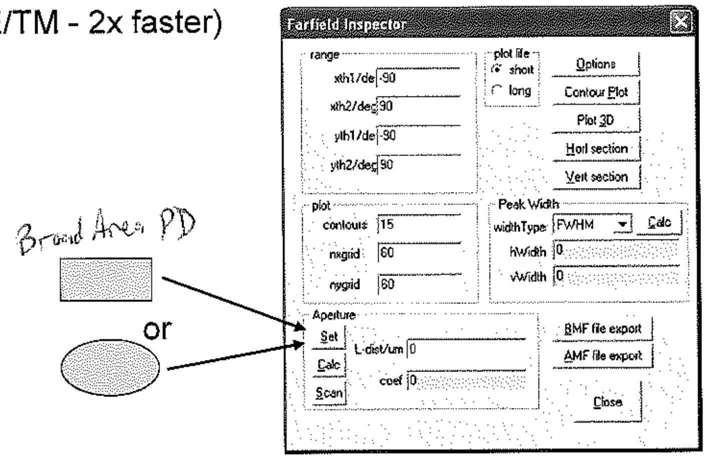
Always check these 4 quantities are OK

Also TE/TM/General option (TE/TM - 2x faster)

Stage 2

- regrid to smoothe
- builds a profile of any part of generated farfield

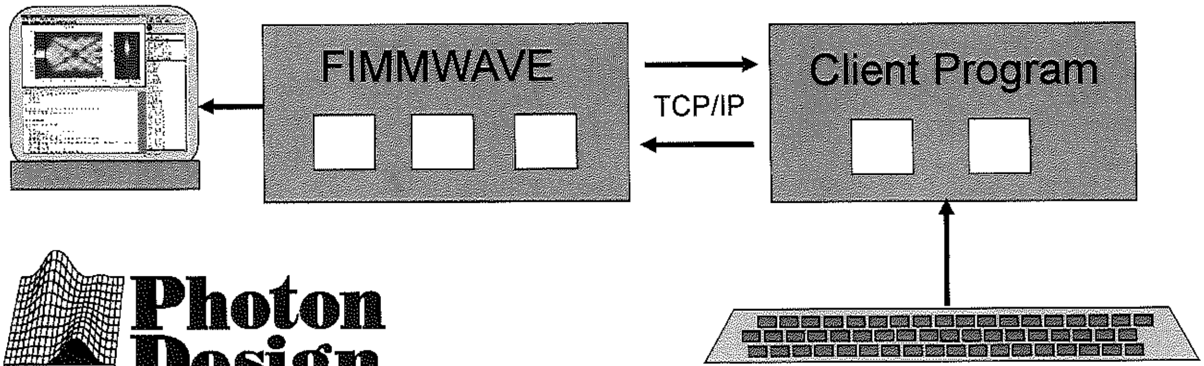




The Client-Server Framework

All of FIMMWAVE and FIMMPROP can be controlled either from:

- the command line interface, or
- from your own programs via a TCP/IP.
- Integrate our tools with your own code or in-house expertise
- extend with powerful optimisation techniques
- also macro-recorder for rapid script creation.



FIMMWAVE

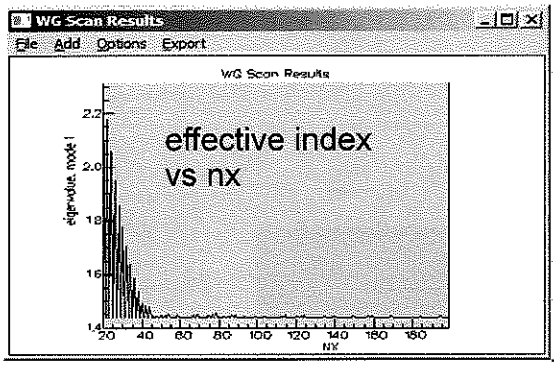
an advanced tutorial

Photon Design®

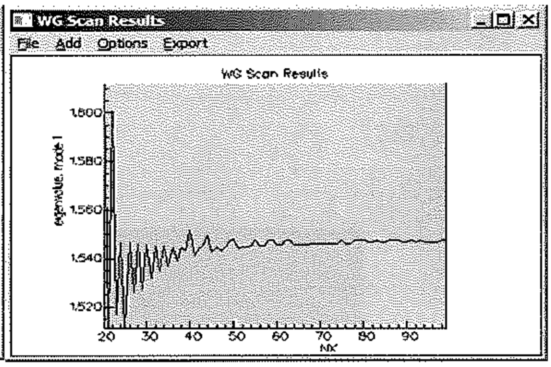
A few words on speed and accuracy

Photon Design®

Accuracy and Speed of FDM Solver



50nm Si layer in air



A similar but graded structure – smaller sampling error

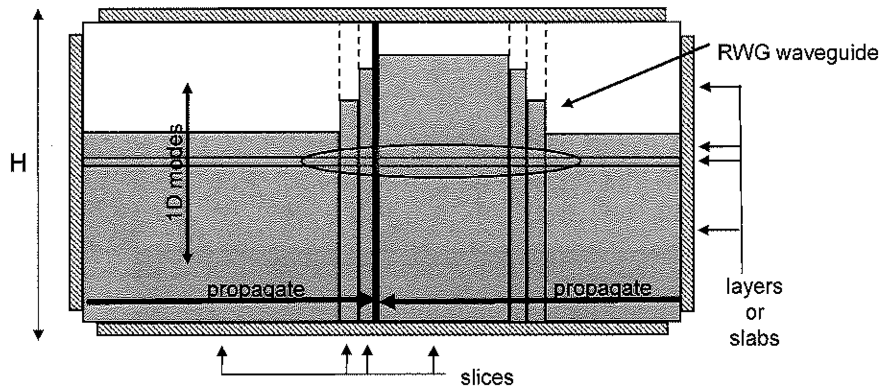
Sampling error is usually the limitation of FDM Solver

Time ∝ (number of grid points)$^{1.25}$
∝ 1/(gridCellWidth)$^{2.5}$, if gridCellHeight=gridCellWidth

Photon Design®

Smoothly varying structures better w/ FDM

Accuracy and Speed of FMM Solver



decr decr

Precision ~ N1dmode / H / "RIX contrast"

Speed ~ 1 / nslices / N1dmode^3

Layers / Vertical resolution for free

Photon Design®

Guidelines for Accurate Mode Calculations

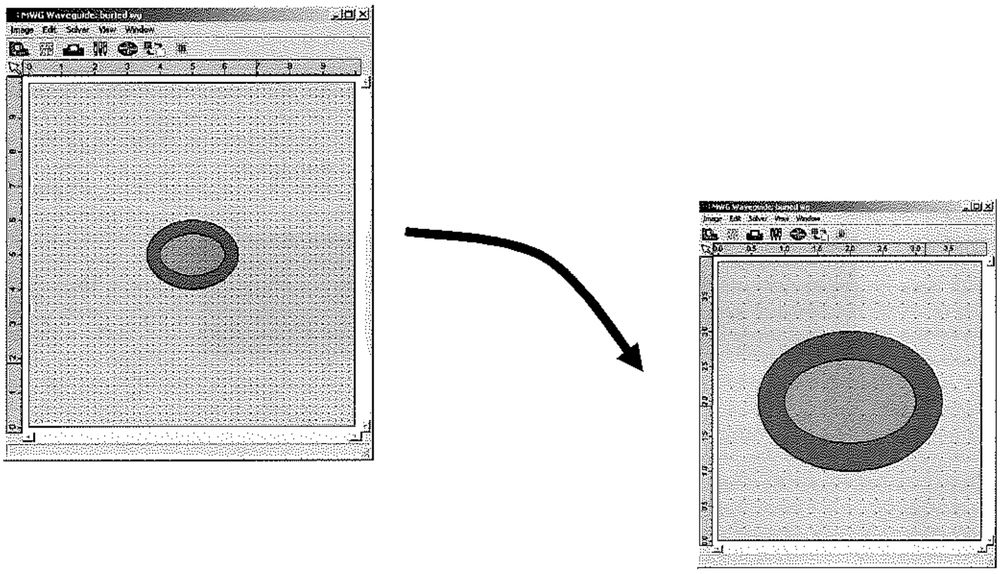
Photon Design®

General Procedures

- Always try to make the problem as simple as possible, at least initially
- Keep box size (width, height) as small as possible
- Identify symmetries
- Decide best boundary condition (ideally this would not make a difference)
- Decide whether semi-vectorial or effective index approximations are adequate
- Check convergence of the results
  - FMM Solver: check effect of (1d)nmodes (N_{1D}) - Error Analysis tool
  - FDM Solver: check effect of (nx, ny)
  - FEM Solver: check effect of (nxres, nyres)

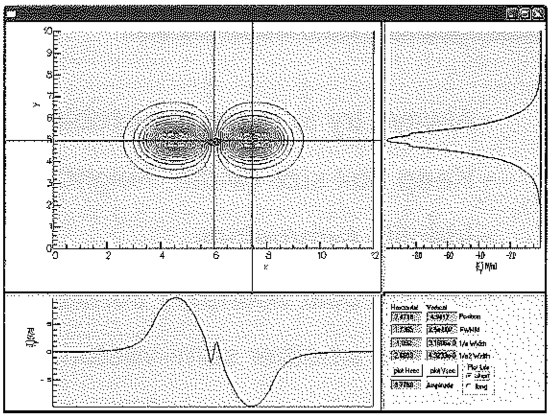
Photon Design®

Smallest Possible Window Size



Photon Design®

Position of the boundaries



Two principles:

1. Keep box size as small as possible
2. Boundary must not disturb the guided modes

Rule of thumb to remember:

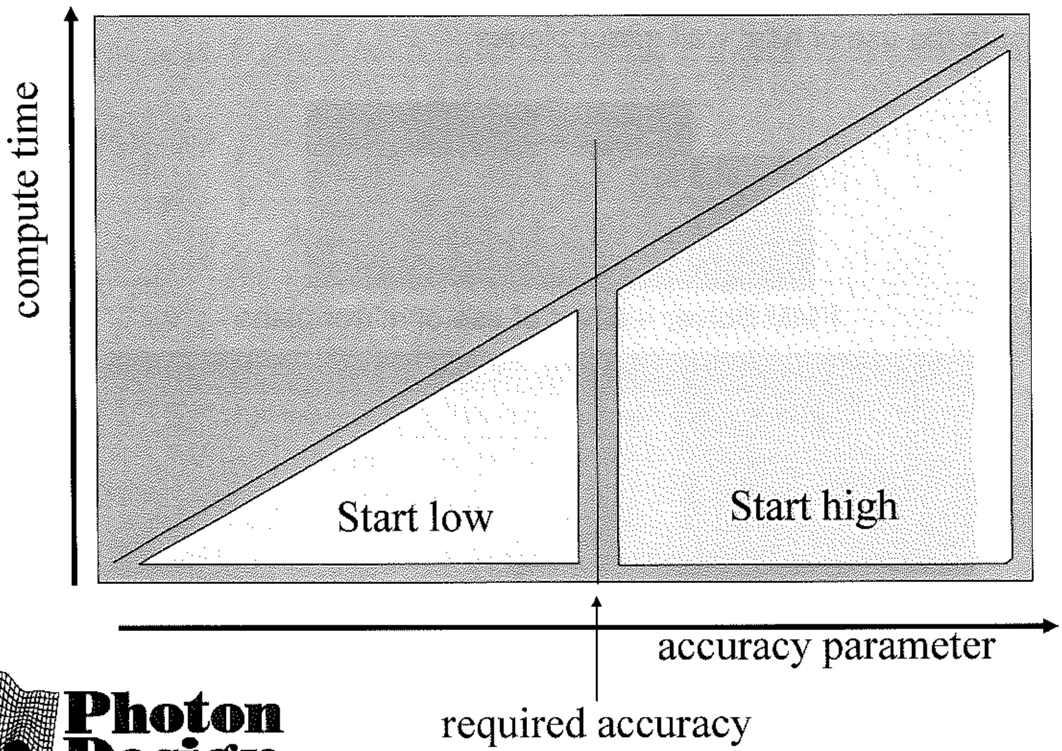
Boundary is far enough away if $\psi=0$ and $d\psi/dx=0$ at the boundary for the guided modes

ψ being the main component of the electric field.

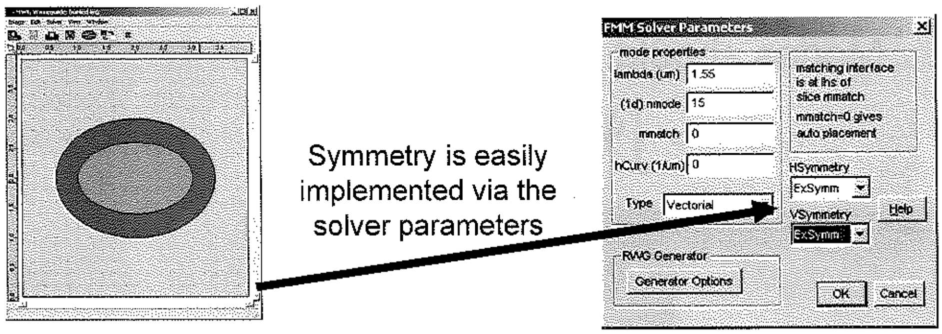
This can be checked using the "Plot Section" tool (Inspect Mode)

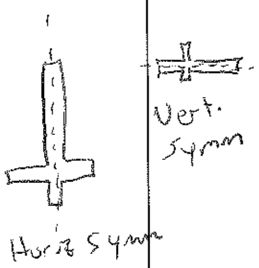
Photon Design®

Don't start with highest accuracy!



Take advantage of symmetries





FMM

- Symmetry in the vertical plane halves the number of 1d modes needed

⇨ Computation 8x faster for FMM

- Symmetry in the horizontal plane halves the number of slices needed

⇨ Computation 2x faster

FDM/FEM

Symmetry in the horizontal or vertical plane halves the number of grid points (FDM/FEM)

⇨ Computation ~3x faster (or ~9x faster for two symmetry planes)

All Solvers

Symmetry planes can remove modes of wrong symmetry

Symmetries and mirror planes

- Horizontal (HSymmetry) and vertical (VSymmetry) symmetry options are equivalent to the use of hard wall mirror planes

	Speed	Profile Memory
Y=H/2	8x	½ x
X=W/2	2x	½ x

FMM Solver – effect of use of mirror plane

	TE Mode (Ex,Hy)	TM Mode (Ey,Hx)
MetalWall at x=W/2	even about W/2	odd about W/2
MagWall at x=W/2	odd about W/2	even about W/2
MetalWall at y=H/2	odd about H/2	even about H/2
MagWall at y=H/2	even about H/2	odd about H/2

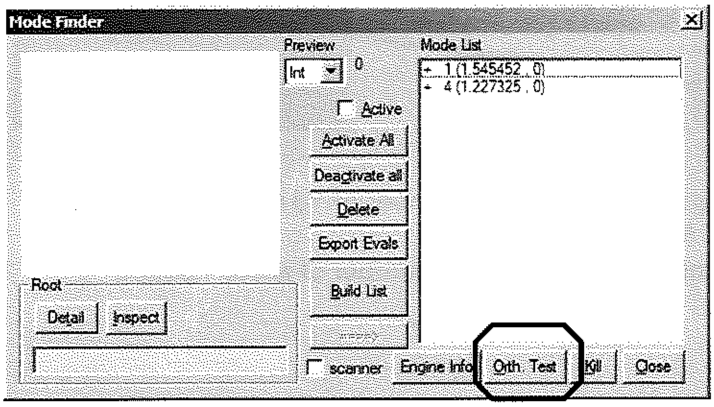
Symmetry boundaries – implemented ***automatically*** by HSymmetry & VSymmetry flags

Photon Design®

Checking your modes: orthogonality test

Modes should be orthogonal:

$$\delta_{i,j} = \int \Psi_i . \Psi_j \, dS$$

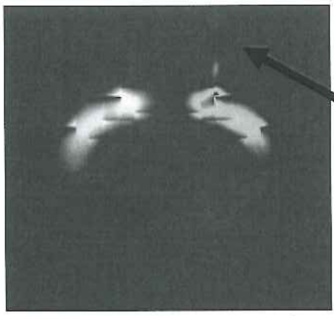


Built in orthogonality checker!
Get into the habit of using this.

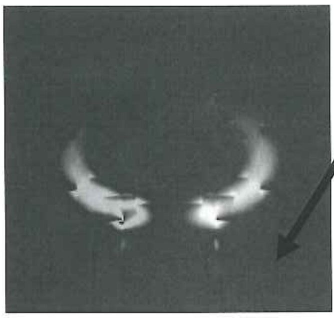
Photon Design®

Checking your modes: mode profiles

Mode 1



Mode 2



Look for Discontinuities!

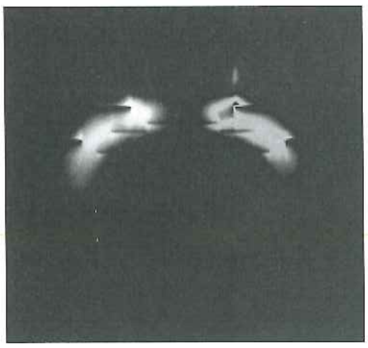
Check Orthogonality!

Not enough 1d modes!

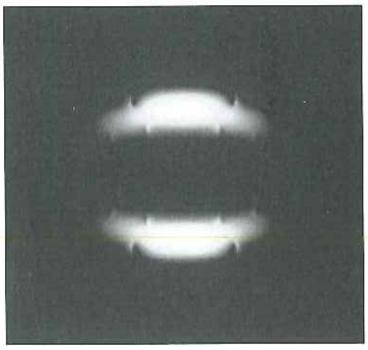
Photon Design®

Checking your modes: mode profiles

FMM: Increase (1d)nmodes and polish modes



Much Better

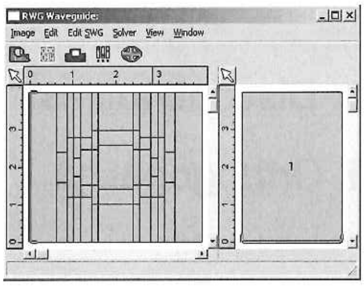


FDM/FEM: Increase grid density + rerun MOLAB

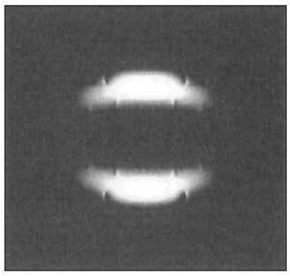
Photon Design®

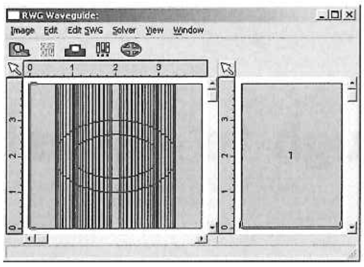
Checking your modes: Generated RWG

FMM: Increase discretisation and polish modes

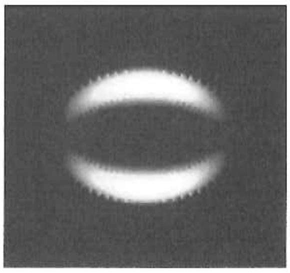


Default settings →





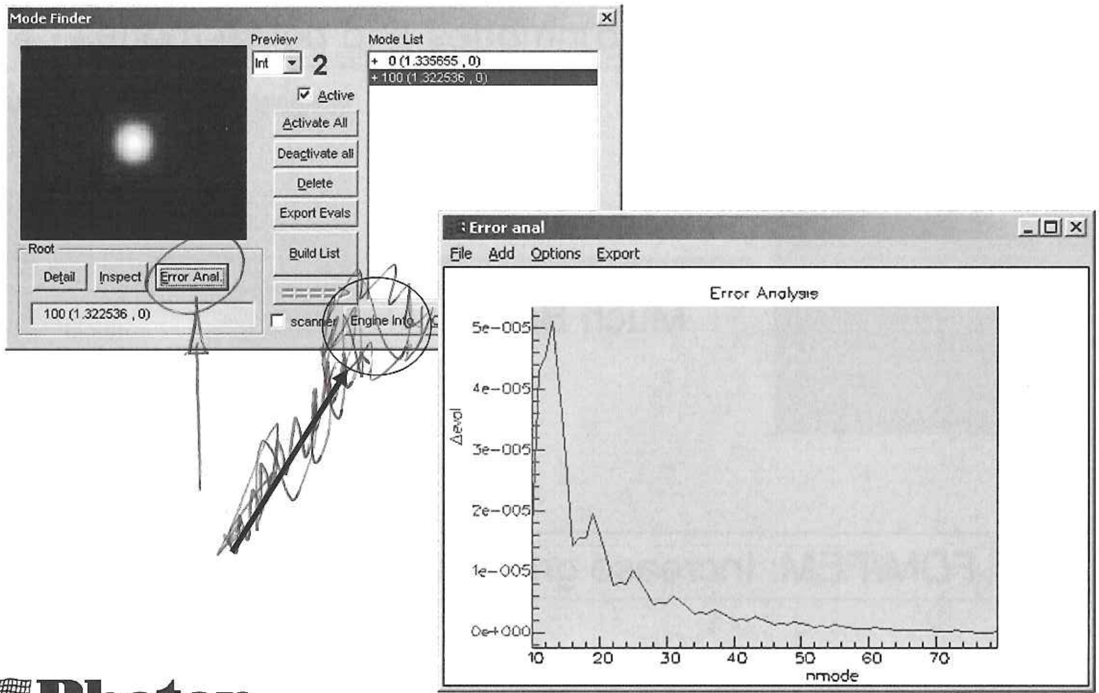
Increase discretisation →



RWG Generator

Photon Design®

FMM: (1d)nmodes Sensititivy

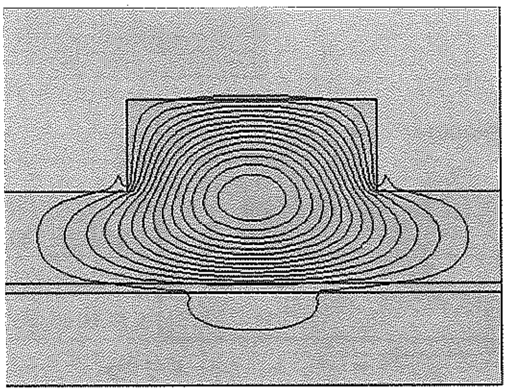


Photon Design®

Finding Leaky Modes

Photon Design®

A Leaky Structure



$F \sim e^{ik_y y}$

The ridge waveguide is a silicon on insulator waveguide which has leakage through the oxide into the underlying silicon

Photon Design®

Absorbing boundary conditions

All solvers support Perfectly Matched layers (PML)

An artificial complex thickness is introduced near the boundary

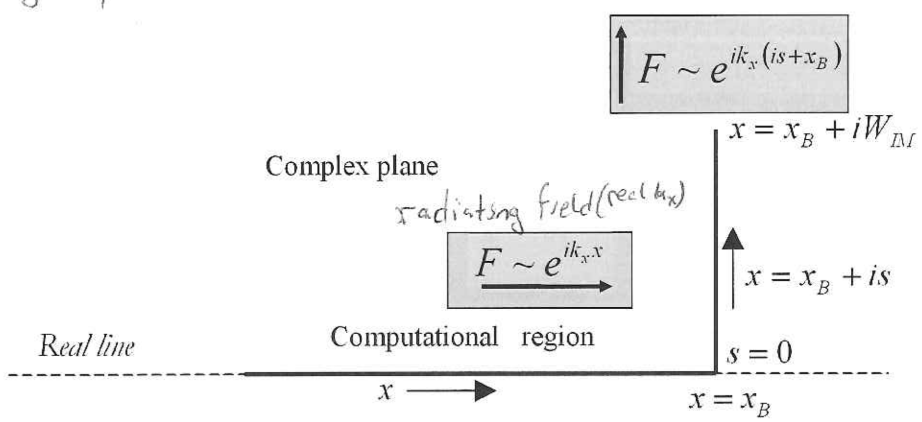
Any wave travelling towards the boundary is attenuated

PMLs in different solvers work slightly differently

Photon Design.

PML boundary conditions

Imaginary thickness that causes attenuation



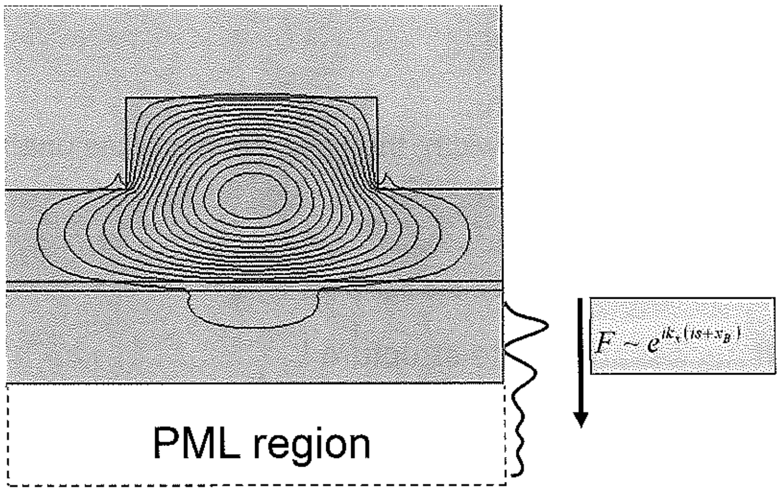
How wide should the PML be?

⇨ Wide enough to absorb the radiation

⇨ Some trial and error needed

Photon Design.

A Leaky Structure



Photon Design®

PMLs with FDM Solver

FDM Solver supports PMLs on all 4 sides

- PML width must be thick enough
- If too thin then may reflect off PML or off boundary (not absorbed by PML)
- If too thick then waste grid points (time)

We will illustrate this with bend loss...

Photon Design®

PMLs with FMM Solver

With the FMM Solver:

- PML on top or bottom:
when increasing PML width, the basis set becomes unable to accurately represent the field

- PML on side walls:
PML width has no effect on the basis set

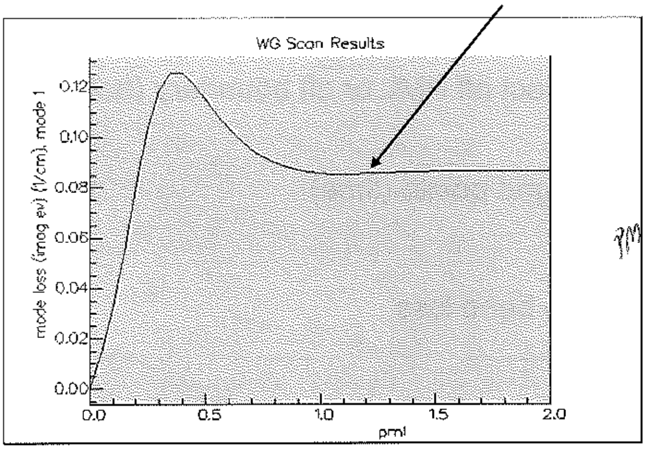
⇨ If necessary: rotate structure !

Photon Design®

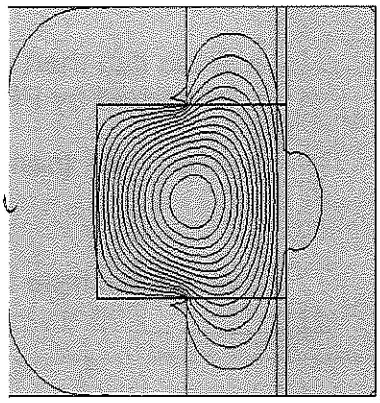
PMLs with FMM Solver

On the sides: Good

As we increase PML width, propagation constant converges



PML width on side



PML PML

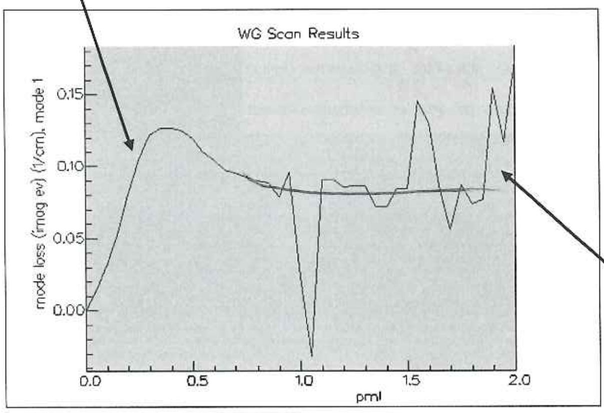
~ µm's

Photon Design®

PMLs with FMM Solver

On top and bottom: Not so good

As we increase PML width, absorption increases...



misbehaved

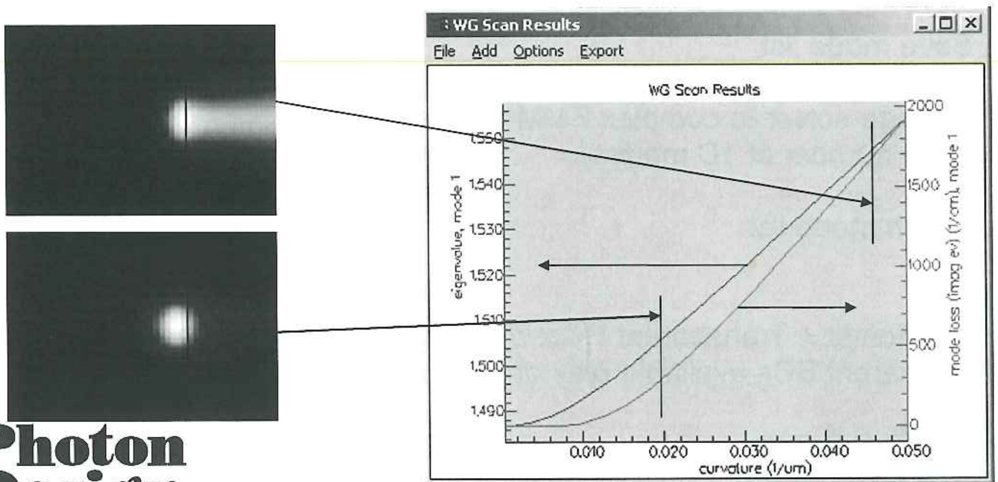
But then 1d mode basis set becomes degenerate and results are noisy

Photon Design®

Bend Mode Solvers

Three bend mode solvers:
- FMM Solver in cylindrical co-ordinates (rigorous)
- FDM & FEM use approximations

Computing bend loss for given radius of curvature:
- FEM/FDM: use with PMLs (left or right) and Complex Solver
- FMM: use transparent BCs and Complex Solver in domain of FMM

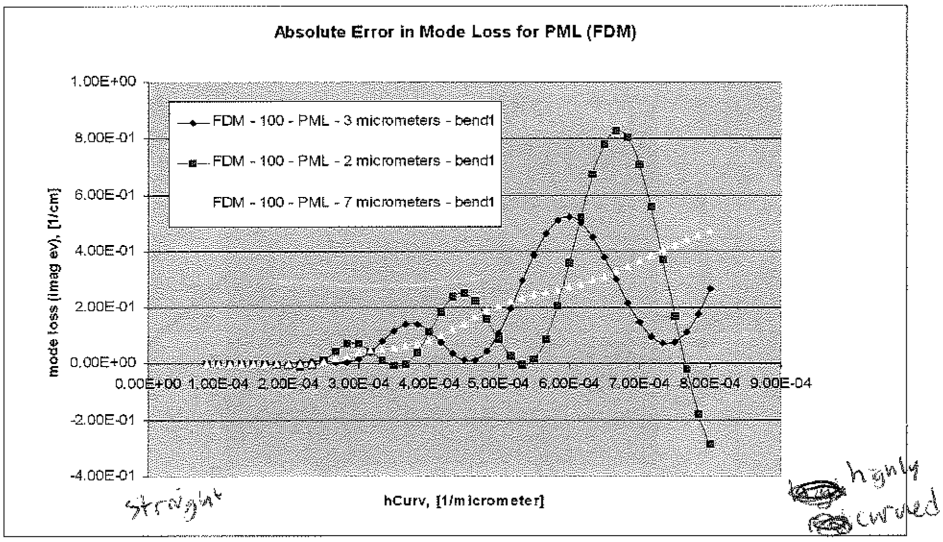


Waveguide Scanner Output

Photon Design®

Using the FDM Bend Mode Solver

- Light will radiate towards outside of curve – must put PML there.
- PML thickness is critical:



Photon Design®

- Start with a PML thickness around 10% of size of compute domain – but will depend on structure.

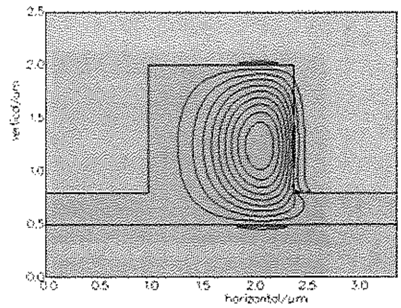
Nx & Ny set by window (set by user)
Adding PML automatically adds extra grid / computation

Using the FMM Bend Mode Solver

Finding very lossy bend modes with the FMM Solver can be tricky!

Here is how to proceed:

1. Start with complex FDM Solver with Transparent BC or PML
   (high spatial resolution nx, ny)
2. Save mode list
3. Change solver to complex FMM Solver
   (high number of 1D modes)
4. Polish mode list



Notes:

•FMM Solver + Transparent BCs: most accurate option if geometry allows
•Transparent BCs available only on sides. No PML thickness to guess!

Photon Design®

Orth. TEST Special overlap uses same grid & gives test of basis set of simulation
Non special overlap uses separate grid to test orthogonality

FIMMPROP

an introductory tutorial

Photon Design®

What is FIMMPROP?

- a tool for optical propagation
- rigorous solutions of Maxwell's Equations
- compare ray-tracing and BPM – these solve approximate equations
- sub-wavelength effects, diffraction/interference, good for small cross-sections, not for telescope lenses!
- 3D
- full vectorial
- based on EME (EigenMode Expansion)
- much faster than previous techniques for many applications
- much more accurate in many cases too

Photon Design®

Propagation Algorithms: BPM and EME

Photon Design®

The Beam Propagation Method

Best known in industry.

Assuming:

1. Slow Z variation of refractive index. $\Rightarrow \nabla(\mathbf{e} \cdot \nabla \ln n^2) \sim \nabla_t(\mathbf{e}_t \cdot \nabla_t \ln n^2)$
2. Small angle of incidence of initial signal
3. Paraxial assumption:

$$\mathbf{e}(x,y,z) = \Psi(x,y,z)\exp(i k n_o z)$$

slow variation of $\Psi \Rightarrow (\partial^2/\partial z^2)\,\mathbf{e} \sim 2 i k n_o \partial\Psi/\partial z - k^2 n_o^2 \Psi$

Wide angle BPM: K large angle from z, but only traveling at one angle

Then wave equations can be reduced to the form:

$$\partial\Psi_t/\partial z = f_t(\Psi_t, \nabla_t\Psi_t, \nabla_t^2\Psi_t, z)$$

Photon Design®

The Beam Propagation Method

$$\partial\Psi_t/\partial z = f_t(\Psi_t, \nabla_t\Psi_t, \nabla_t^2\Psi_t, z)$$

- This is a parabolic equation.
- Can easily solve by propagating an initial input profile in small steps.
- Very good for slowly varying tapers , y junctions, D couplers, ...
- Completely useless for discrete structures!

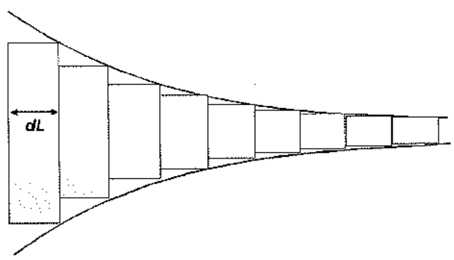
Abrupt On

Photon Design®

EingenMode Expansion - EME The Mode Matching Method

Strategy:

- Consider generic structure as a sequence of small uniform **subsections**. For continuous structures, such as tapers, some sort of discretization is needed.

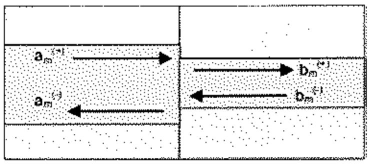


- In each section m, calculate set of eigenmodes $\{ \Psi_i^m \}$; $\Psi^m = [\mathbf{e}, \mathbf{h}]^m$

$$\Psi^m(x,y,z) = \sum_{i=1}^{N} \left(a_i^m \Psi_i^m(x,y)\, e^{i\beta_i^m z} + b_i^m \Psi_i^m(x,y)\, e^{-i\beta_i^m z} \right)$$

FWD BWD

Photon Design®



- Calculate the transition matrix **S** across interfaces:

$$S^{(m+1,m)} = < \Psi^{m+1}, \Psi^{m} >$$

- Calculate the transition matrix $D^{(m)}$ across a section:

$$D^{(m)} = \text{diag}\, [\, \exp(i\beta^{m} z)\, ,\, \exp(-i\beta^{m} z)\,]$$

- Then relationship between the field coefficients **c** = [**a**,**b**] is

$$c^{m+1} = S^{(m+1,m)}\, D^{(m)}\, c^{m}$$

- Compose the transition matrices of all subsections to obtain transition matrix for entire device:

$$S^{(M,1)} = S^{(M,M-1)}\, D^{(M)} \ldots S^{(2,1)}\, D^{(1)}$$

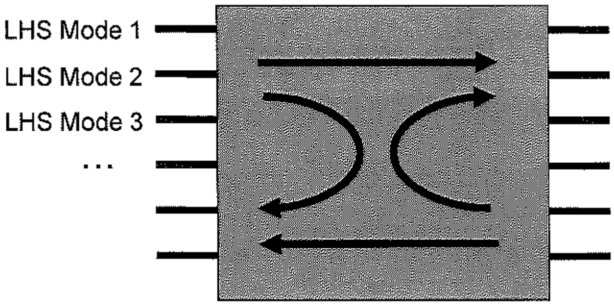
+ reflection elements

Photon Design®

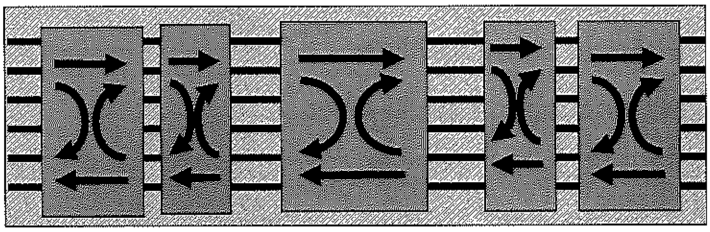
Benefits of EingenMode Expansion

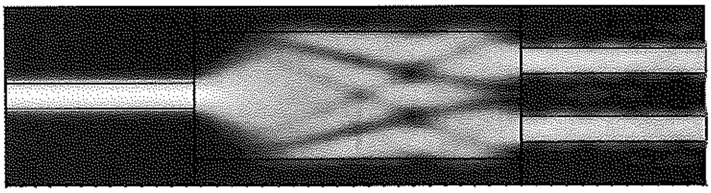
Photon Design®

Scattering Matrix Approach



- Solves for *all* inputs
- Component framework
- Port = mode (usually)
- Alter parts quickly





Photon Design®

MMI: only include guided modes so that radiation modes don't recouple from wall reflections

Solves for All Inputs

Traditional Tool:

- Input 1 → calculate → Result 1
- Input 2 → calculate → Result 2
- Input 3 → calculate → Result 3

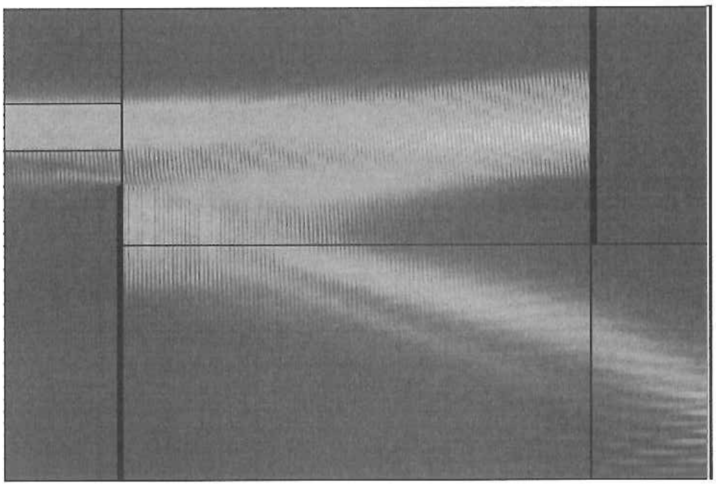
FIMMPROP:

- Calculate S matrix once
- Input 1 → Result 1
- Input 2 → Result 2
- Input 3 → Result 3

Photon Design®

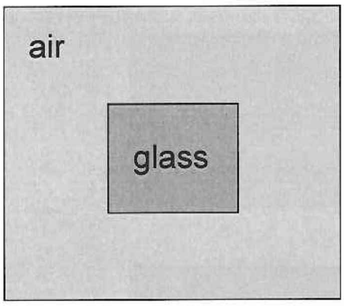
Bi-Directional Capability

- Unconditionally stable
- Takes *any number* of reflections into account
- NOT iterative
- Even resonant cavities
- Mirror coatings, multi-layer

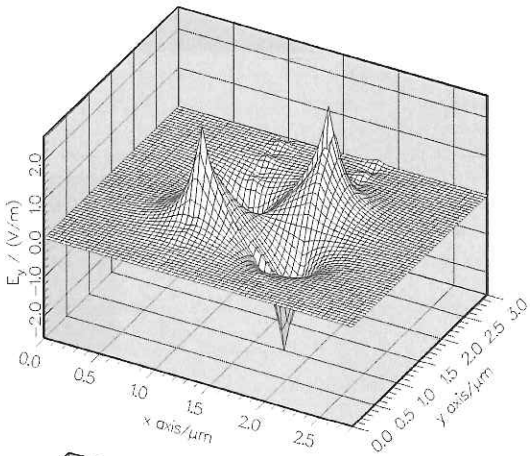


Photon Design®

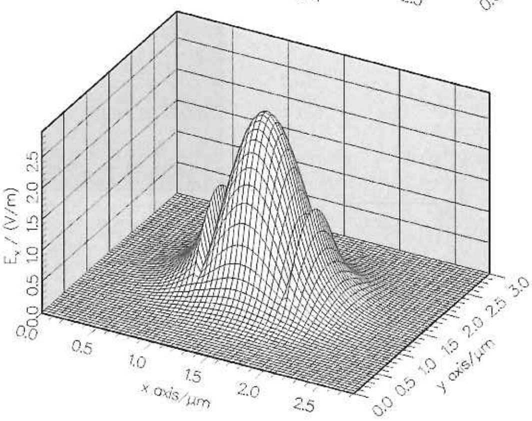
Fully Vectorial



E_y Field



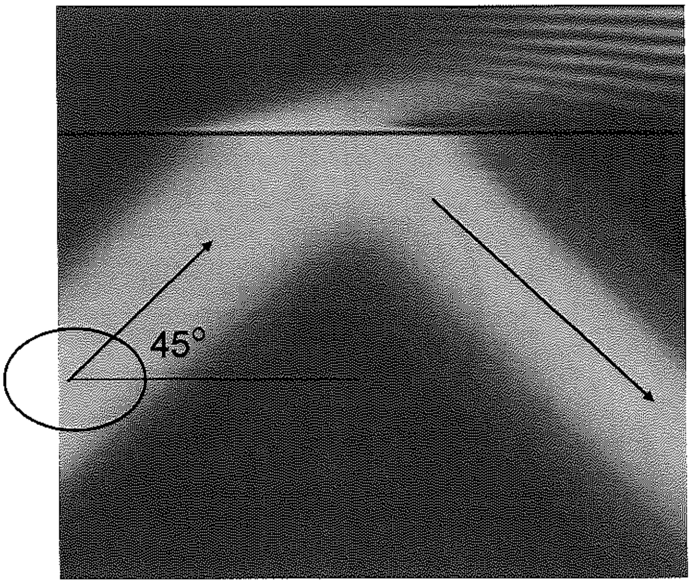
E_x Field



Photon Design®

Wide Angle Propagation

- No paraxial approximation
- Just add more modes

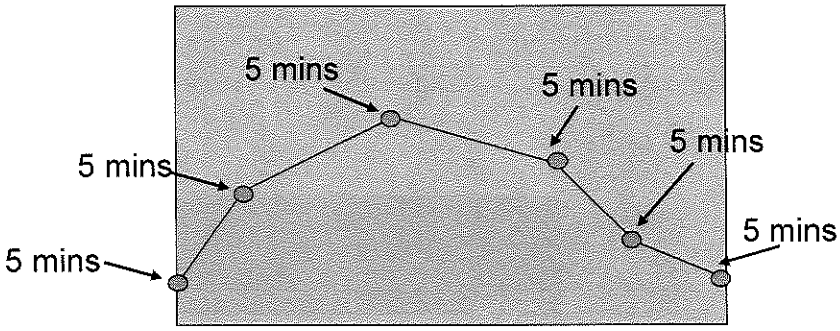


Photon Design®

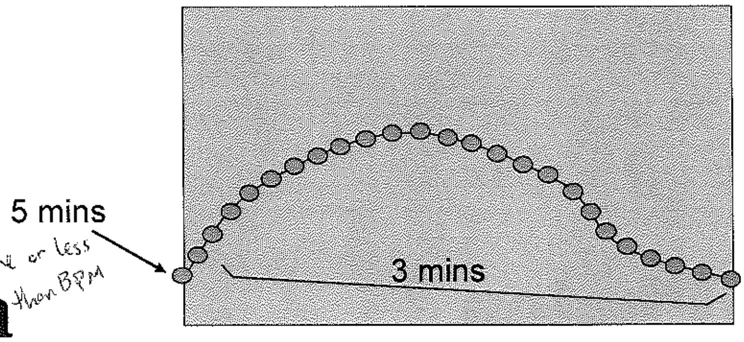
Design Curve Generation

Not λ, but length/physical parameters good to reuse S-matrix response

Traditional Tool:



FIMMPROP:

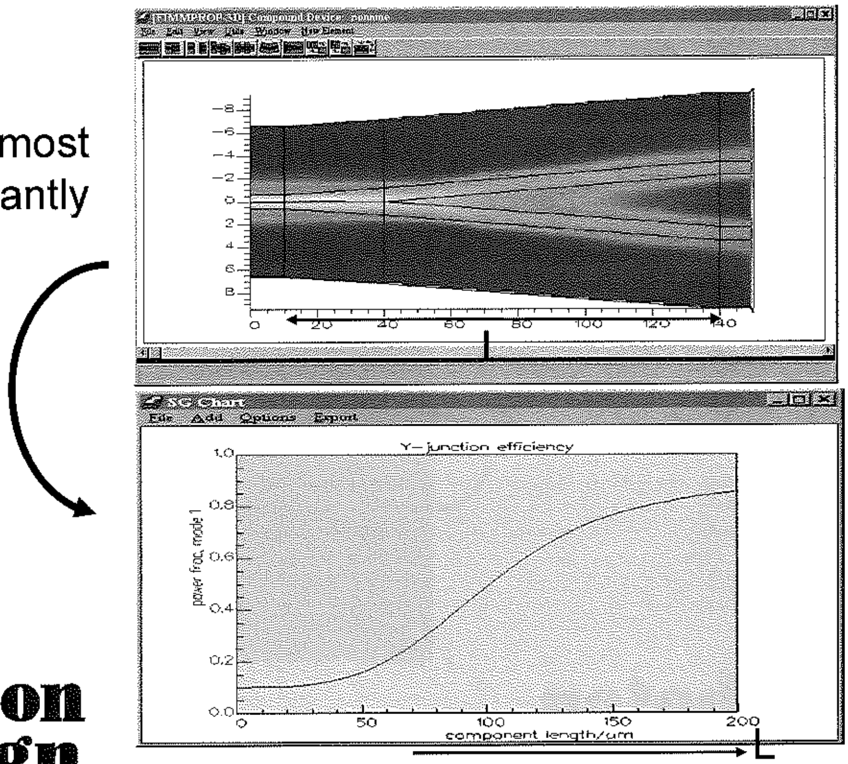


More or less than BPM

Photon Design®

Design Curve Generation

Stretch almost
instantly



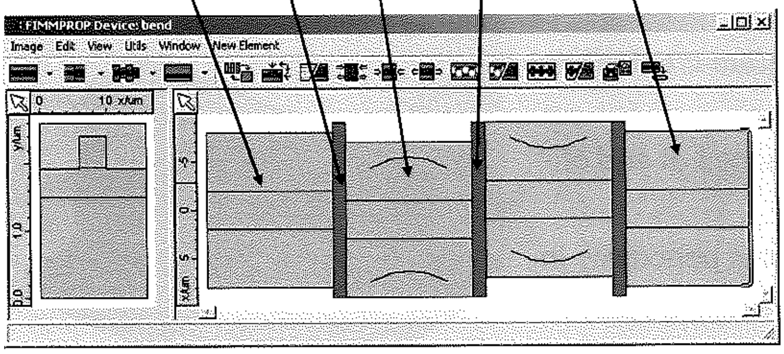
Photon Design®

Working with FIMMPROP

Photon Design®

Basics

Device = (Section,Joint,Section, Joint,…Section)

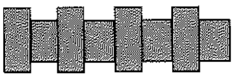


Section = one of:

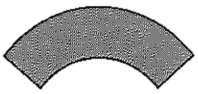
- Waveguide Section (straight)
- Bend Section
- Taper/Planar Section (also Y-branch etc)
- Periodic Section
- Other FIMMPROP Device

Propagation - 4 Cases

Straight sections
- trivial
- instant propagation even along 1km!



Periodic Structures

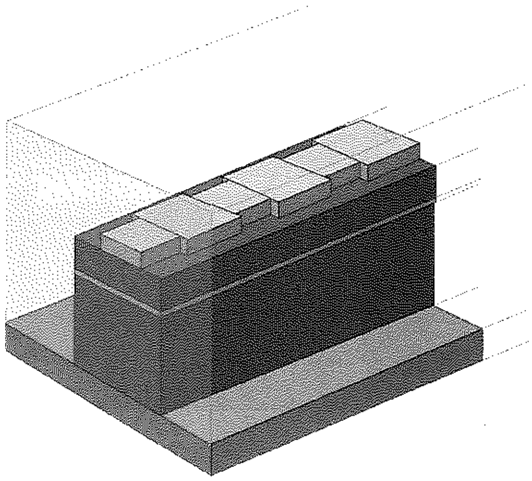


Bends: optimised algorithms

Z varying - harder!

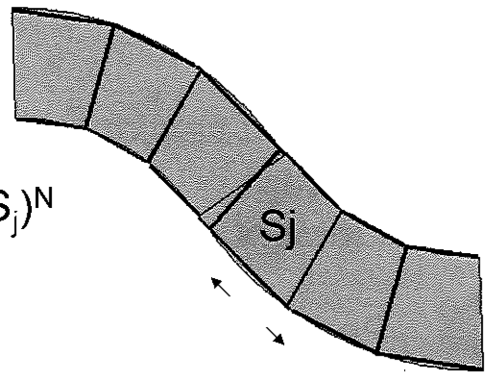
Periodic Structures

- Very efficient - repeat period
- arbitrary complexity within the period
- "Simple Periodic" - program creates joint between periods
- "Periodic" - user defines joint at end of period
- *always* look for and exploit periodicity in the device



Photon Design®

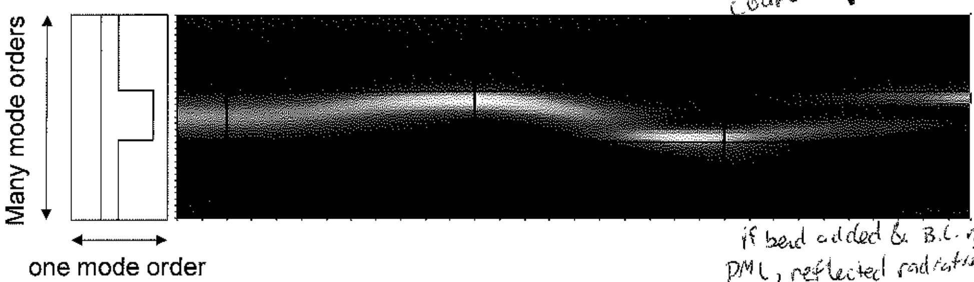
Bends



Transmission: $T = (S_j)^N$

exact answer as N→infinity

if only straight, reflected light will not couple



if bend added & B.C. not PML, reflected radiation mode may couple back!

- Need lots of mode orders in plane of bend
- Keep outside wall away or use PMLs
- Keep height (width) as small as possible

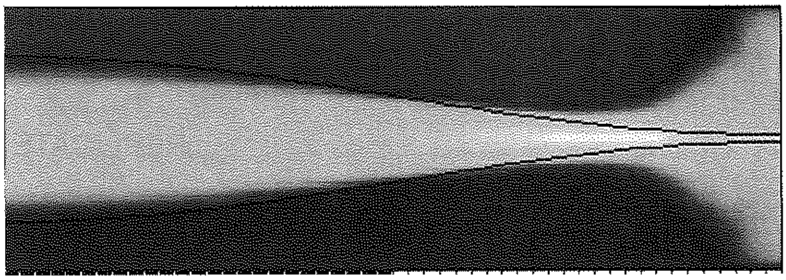
(rotated)

Can minimize vertical coupling if transverse bend

Photon Design®

Z-varying

- Advanced taper algorithm – not staircase; few adaptive steps
- Choice of user interface for specifying Z-variations:
  - ➢Taper Section between two waveguides (Z-dependent weighting function)
  - ➢Taper Section with a Z-parameterised waveguide
  - ➢Planar Section: arbitrary etch and grow paths

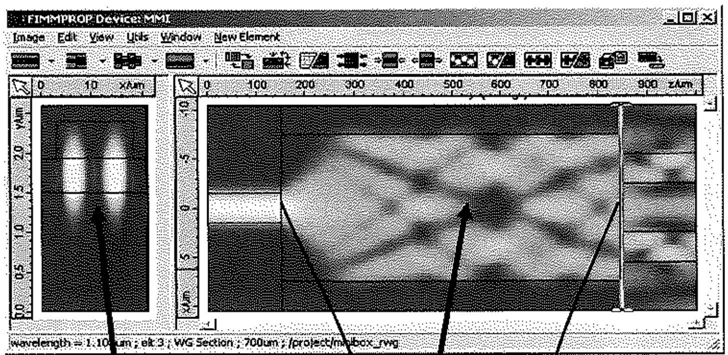


Z=0.0 Z=1.0

Photon Design®

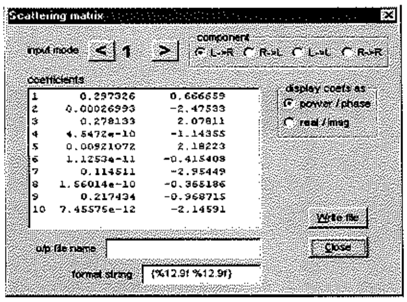
Analysis Tools

- Scattering Matrix
- XY, XZ and YZ Intensity Profiles
- Z Profiles:
  - mode betas
  - mode powers
- Calculation Diagnostic
  - check on accuracy, enough modes etc
  - sometimes loss is OK - we want to ignore radiation modes

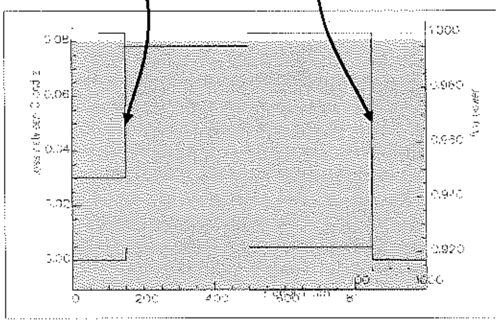


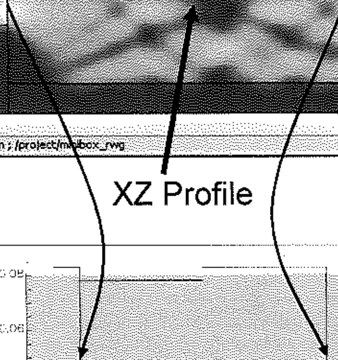
XY profile

XZ Profile



Scattering Matrix





Calculation Diagnostic

Photon Design®

The FIMMPROP Scanner

A) Scan Between Devices

Device A: set of parameters $\{p_{1,a}, p_{2,a}, p_{3,a}, p_{4,a} \cdots \}$
Device B: set of parameters $\{p_{1,b}, p_{2,b}, p_{3,b}, p_{4,b} \cdots \}$

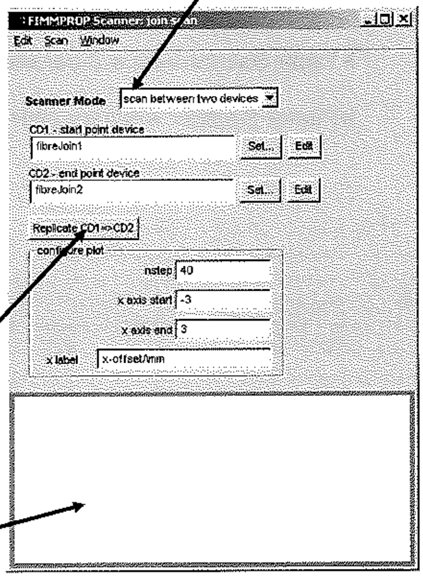
An intermediate device (e.g. 10% A, 90% B)

weighting parameter w = 0.1

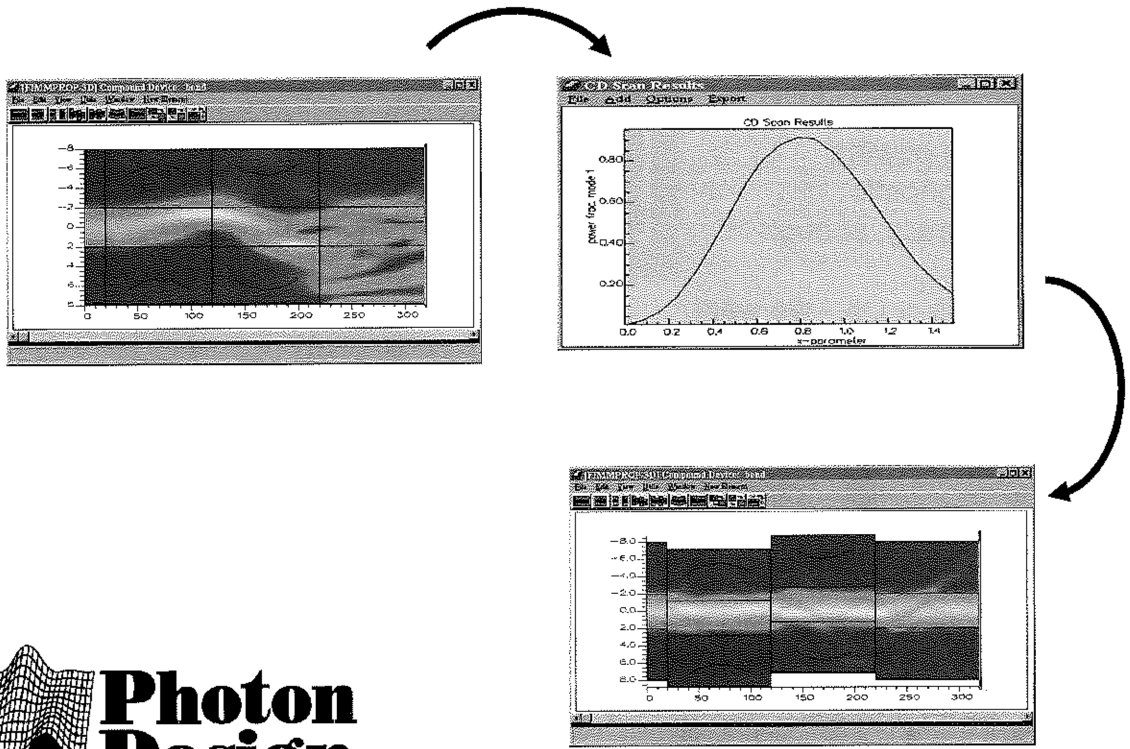
Then the intermediate device is given by:

$$p_j = (1\text{-}w).p_{j,a} + w.p_{j,b}$$

B) Scan A Parameter (Named Variable)



An S-Bend Scan



Loss and Gain

- If there is loss or gain in the waveguide then the modes have a complex β
- Real engine ignores loss and gain
- Real engine can estimate the imaginary part using an overlap integral of the mode profile with the loss/gain profile

So we have two choices for modelling loss or gain in FIMMPROP:

- real engine + overlap integral
  - fast
  - MOLAB works well/fast for real engine
  - good for low loss/gain - ≈100/cm $\alpha = 100\,cm^{-1}$
  - if the real engine and complex engine give similar mode profiles then the overlap integral will be a good approximation
  - no PMLs – still have side reflections
- complex engine
  - exact, required for metals
  - MOLAB good for FDM, FEM, Effective Index Solvers
  - complex MOLAB for FMM solver is slow, - not practical
  - can have PMLs – boundary reflections absorbed

Photon Design®

Memory and Speed

Memory:

increase area by factor of 2

- need 2x number of modes

- each mode needs 2x number of grid points

therefore memory proportional to A^2

Speed:

increase area by factor of 2

- need 2x number of modes

- each mode needs time A^n, (1.3<n<3, depending on method)

therefore time to build modes proportional to $A.A^n$ $= A^{n+1}$

overlaps: # of points X # of modes

therefore time to calculate overlaps proportional to A^3

- overlap integrals will eventually limit EME for very large calculations.

Poor scaling in EME

BPM better at larger cross sections

Photon Design®

FIMMPROP

an advanced tutorial

Photon Design®

Choosing a Mode Solver for FIMMPROP

Planar Structures (3D)

Use FDM Solver to begin with:
- Robust – unlikely to fail in middle of long calculation
- Stores only Ex, Ey fields if you only calculate S-matrix (less memory)
- Fast overlap integrals
- Builds lots of modes at same time (c.f. FMM)
- Has good PMLs for leaky structures

Circular Section Structures

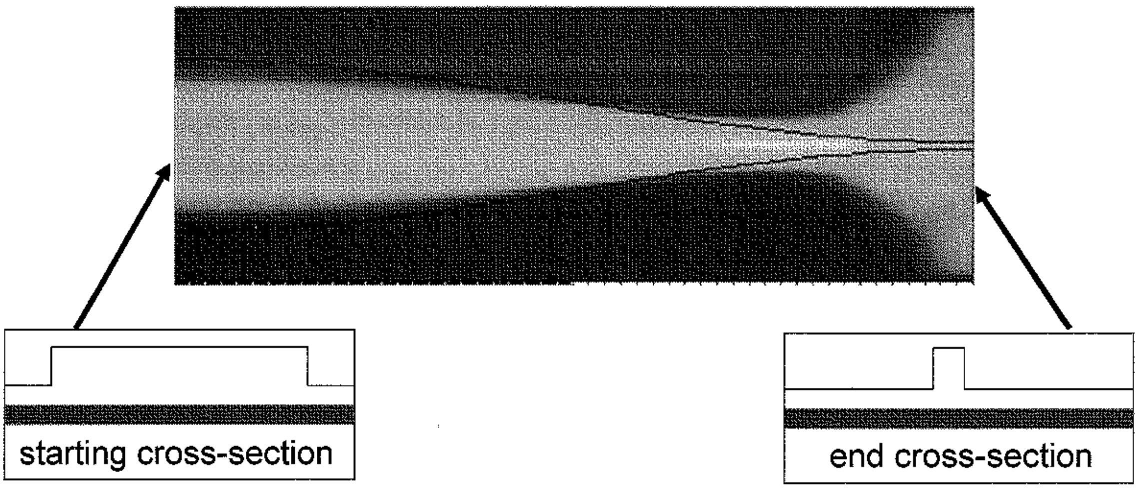
Use FD Fibre Solver:
- Fully vectorial
- Fast 1D overlap integrals
- Builds lots of modes at same time (c.f. GFS)
- Supports PMLs

Photon Design®

Modelling tapers

Tapers - using two cross-sections

Using Taper Section with “interpolation taper”



$\{p_{1,a}, p_{2,a}, p_{3,a}, p_{4,a} \cdots\}$ $\{p_{1,b}, p_{2,b}, p_{3,b}, p_{4,b} \cdots\}$

intermediate waveguide is given by:

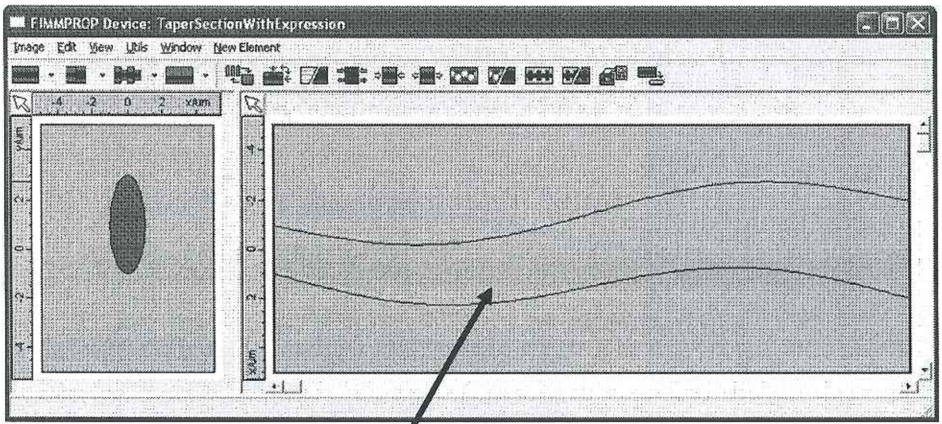
$p_j = (1\text{-}w).p_{j,a} + w.p_{j,b}$

$w=f(z/L)$,

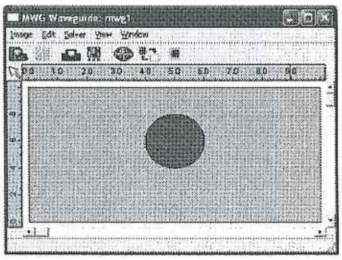
$f(z/L)$: “taper function” - linear,exponential, “S” curve...

Tapers - using a Z-parameterised cross-section

Using Taper Section with "expression taper"



Z=0 Z=1



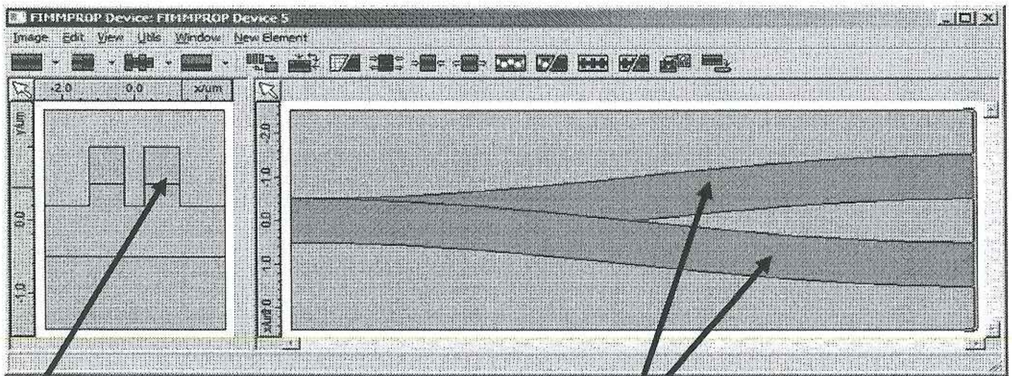
Size and position of shape defined as a function of "Z"

Z is the fractional co-ordinate: varies from 0 (lhs) to 1 (rhs)

per section

Photon Design®

Tapers - using "Planar Section"



One "SWG" defines vertical layer structure for whole taper

Paths - define etch profile in x/z plane

Path properties:

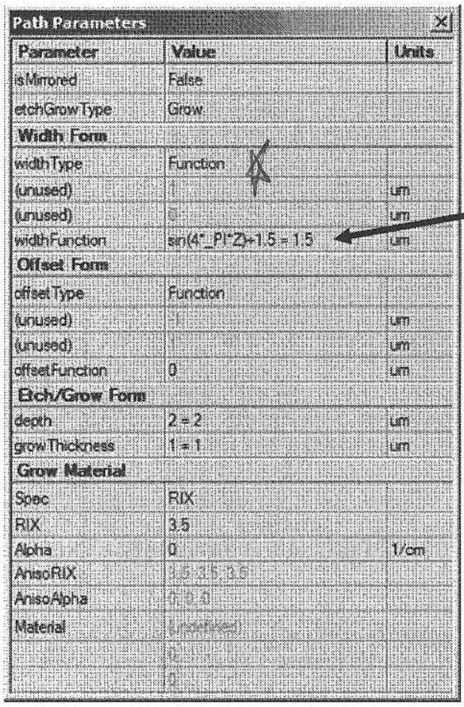
- width(Z)
- offset(Z)
- Mirroring
- Etch or Grow

Photon Design®

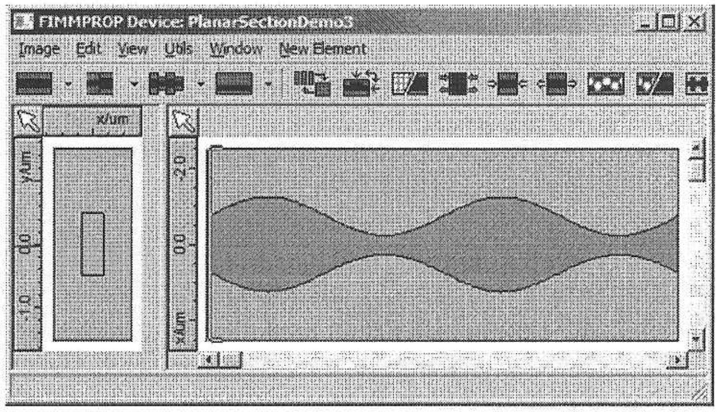
Using expressions in the "Planar Section"

Many path parameters can be of form "expression(Z)".

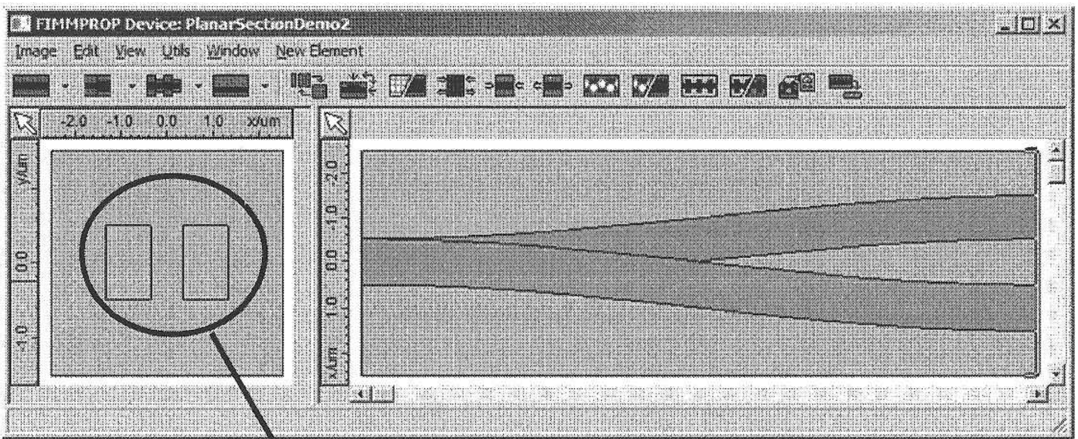
Even SWG can have Z-expressions, so layer thicknesses and rix can be controlled.



width = sin(4*_PI*Z)+1.5



"Planar Section" - "Grow Paths"



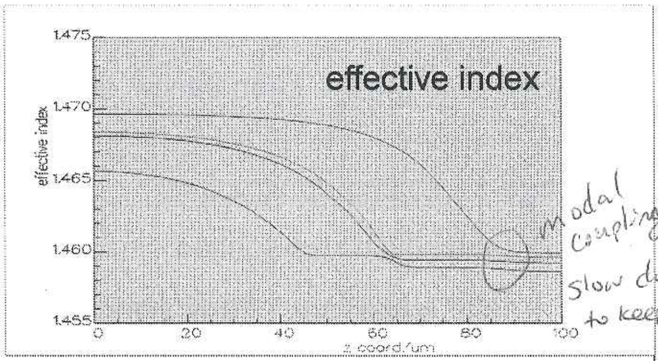
Si rectangles "grown" (superimposed) over background from SWG.

Can combine etching and grown paths to create complex geometries.

Photon Design®

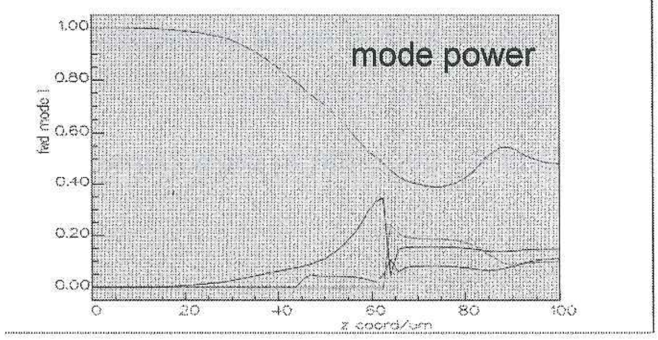
Tapers: Principles

- goal: adiabatic (no loss) - power stays in zero-order mode
- need only a few modes when near the adiabatic limit (loss<10%)
- use loss vs length plot - v. fast
- modal analysis is very powerful tool for taper analysis - mode properties tell us a lot



modal coupling

Slow down taper to keep lossless



Already computed

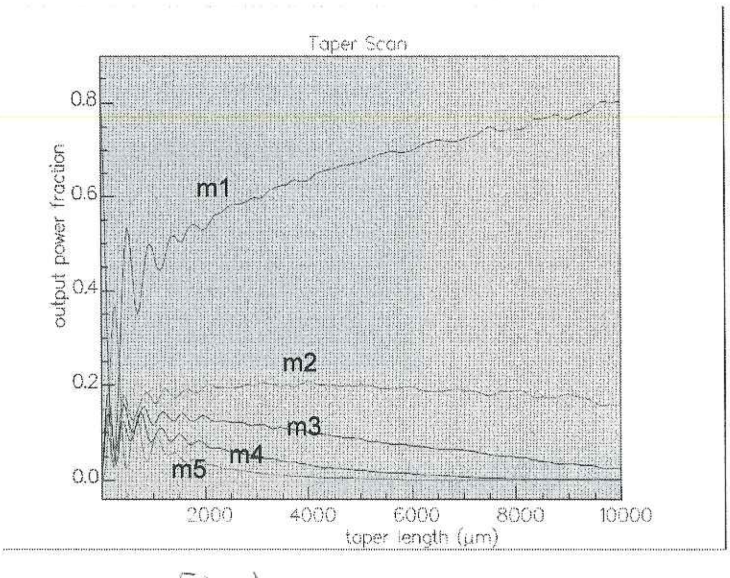
Photon Design®

The "Taper Curve"

- FIMMPROP framework allows us to compute a set of similar structures very quickly
- Very important for design process - determines adiabatic length

Note how higher order modes die out more quickly

Scanner



Fiber taper

Photon Design®

Tapers: Planar Geometry

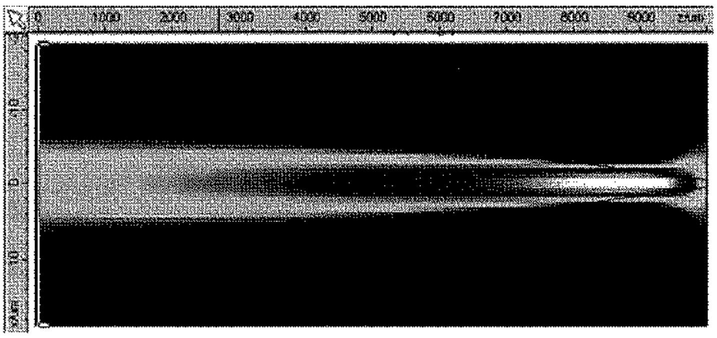
Golden rule: always keep things as simple as possible, especially initially

- Stage 1 - setting up structure
  - minimise cross-section - inspect size of zero order mode
  - take advantage of any symmetries - an odd mode will never couple to an even mode
- Stage 2 - look at mode propagation constants with WG-Scanner
- Stage 3 - calculate taper
- Stage 4 - generate a "taper curve" - efficiency v.s. length

Photon Design®

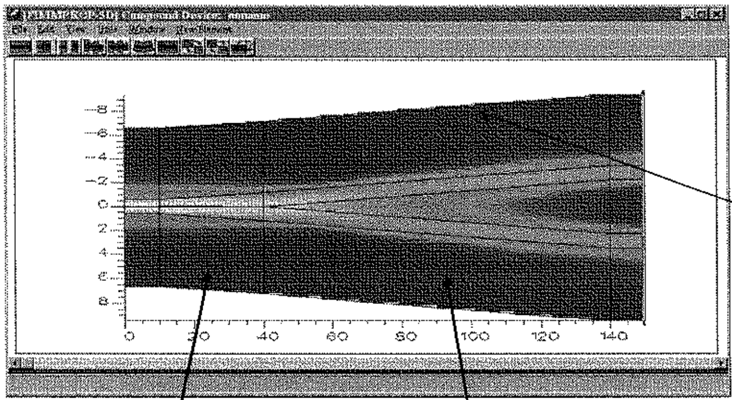
Tapers: Circular Geometry

- Must use the GFS Fibre Solver or FDM-Fibre Solver
- Fibre modes of different quantum numbers (m-order, p-order) do not couple - you should only include one m-order and one p-order



Photon Design®

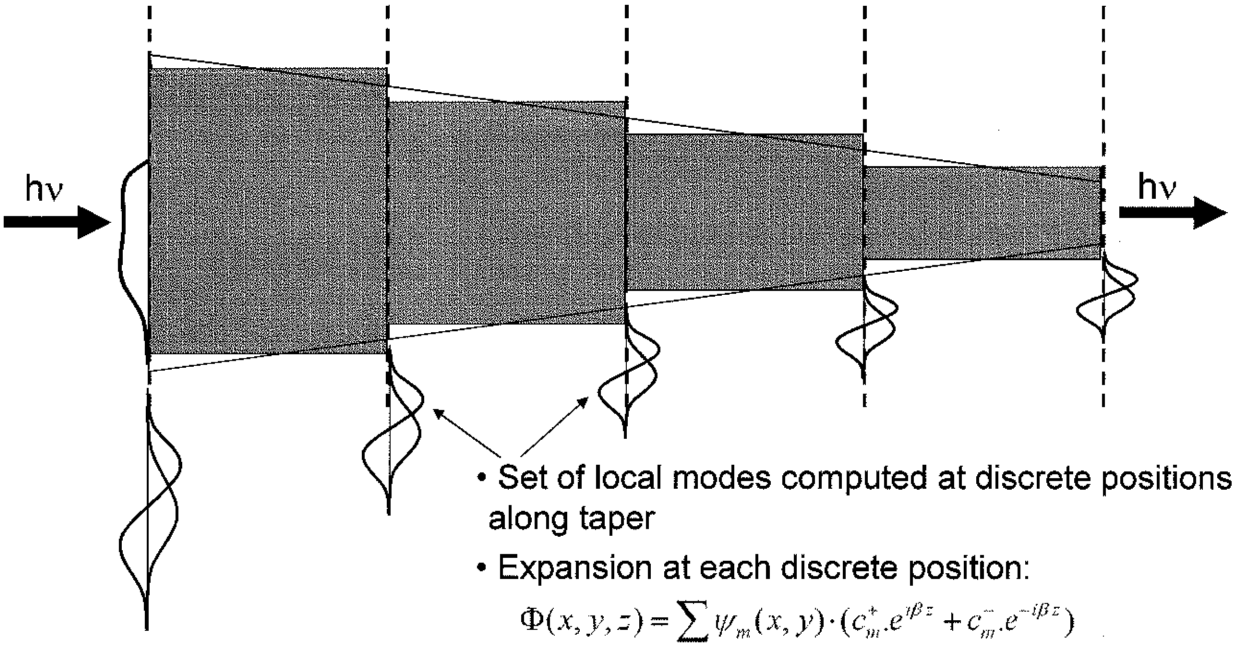
Other continuously varying structures...



- "Taper" algorithm is used for all continuously varying structures
- may need more than one "taper section" (RWG tapers)

The taper algorithm

Taper: Zero Order - Staircase Approximation



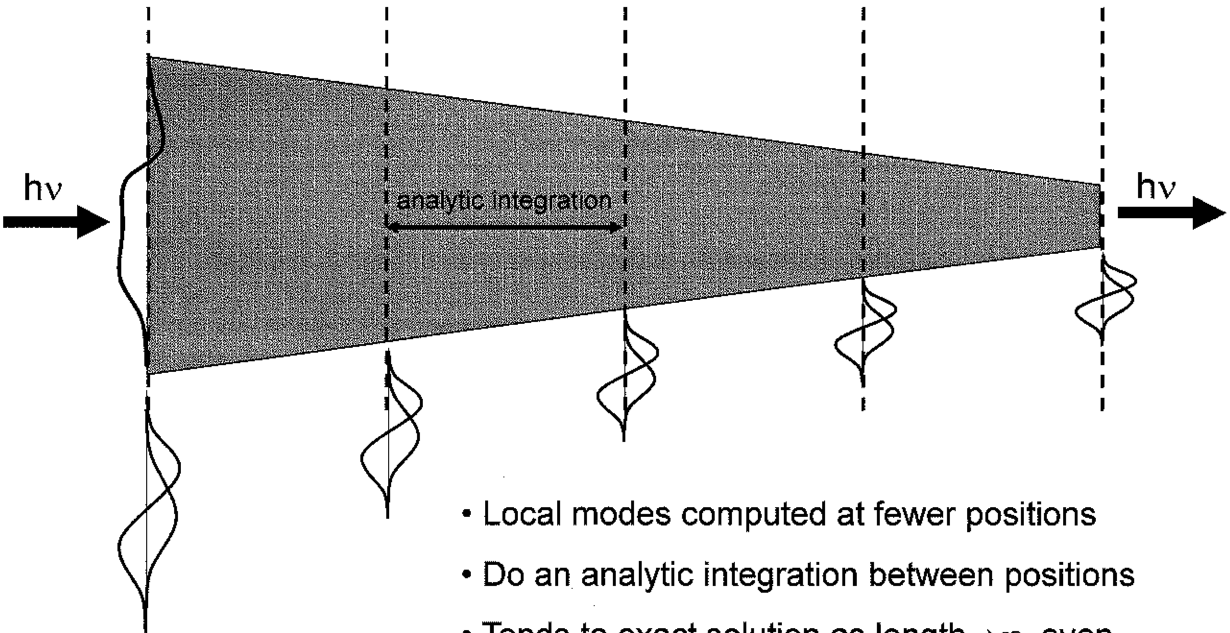
- Set of local modes computed at discrete positions along taper
- Expansion at each discrete position:

$$\Phi(x,y,z)=\sum_m \psi_m(x,y)\cdot(c_m^+.e^{i\beta z}+c_m^-.e^{-i\beta z})$$

- In theory, exact as number of sections $\rightarrow\infty$
- In practise limited by numerical errors as $N_{sec}\rightarrow\infty$
- Ideal for calculations of non-adiabatic tapers

Photon Design®

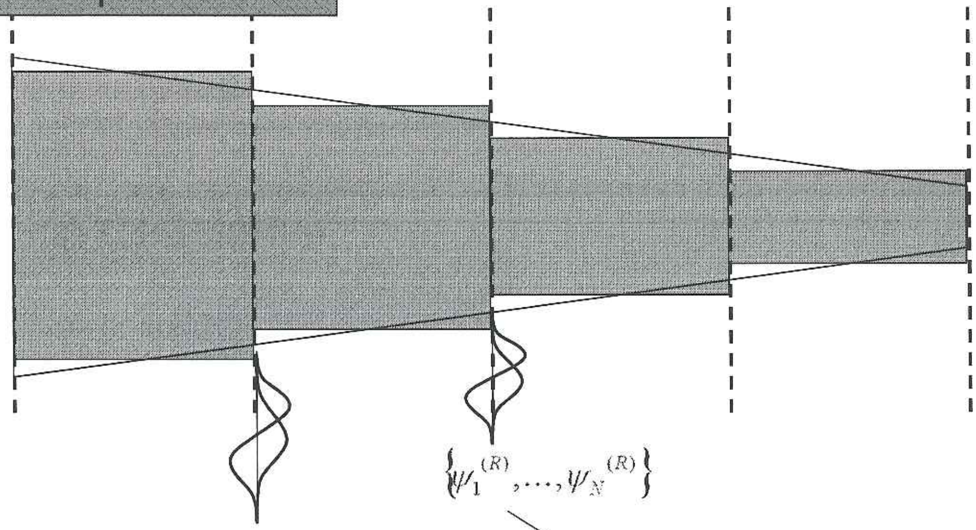
Taper: First Order Integration



- Local modes computed at fewer positions
- Do an analytic integration between positions
- Tends to exact solution as length$\rightarrow\infty$, even for finite number of sections
- Ideal for near-adiabatic devices - need only few modes

Photon Design®

Precision parameters



$\{\psi_1^{(R)},\dots,\psi_N^{(R)}\}$

$\{\psi_1^{(L)},\dots,\psi_N^{(L)}\}$ → $J_{ij} = \int \psi_i^{(L)} \psi_j^{(R)} dS$

As number of sections increases, **J→Identity**

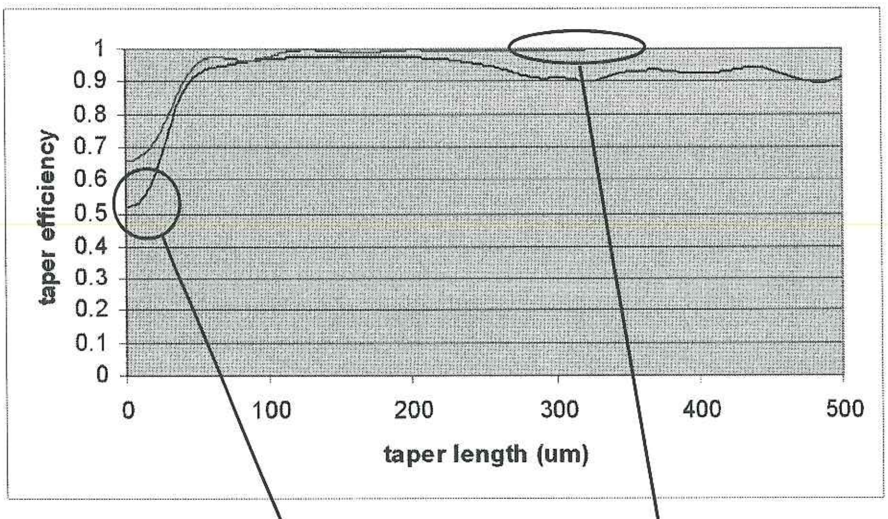
Tolerance = upper bound of **||J- Identity||**

Close to identity matrix

Tolerance can be set

Photon Design®

Comparison zero order / first order



first-order is more accurate at long length

zero-order is more accurate at short length

Photon Design®

Calculation Diagnostics

Photon Design®

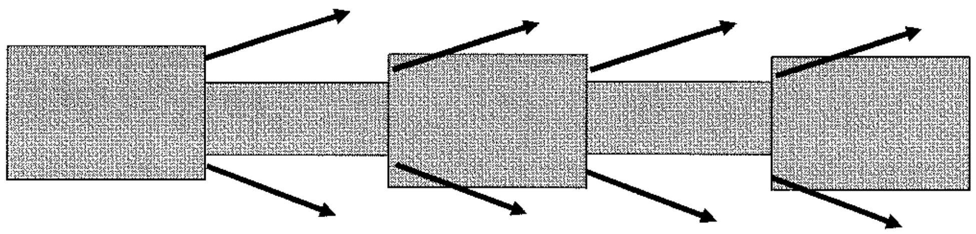
Using the Calculation Diagnostics

Two types of Calculation Diagnostics:

- Power – detects when and where power is disappearing from device.
- Joint – detects losses only at joints. Use this if your device has gain or loss.

BUT:
Sometimes loss is ok. E.g. you might want to throw away radiation modes.

Long period grating – can ignore radiation modes since once radiated, very little couples back in

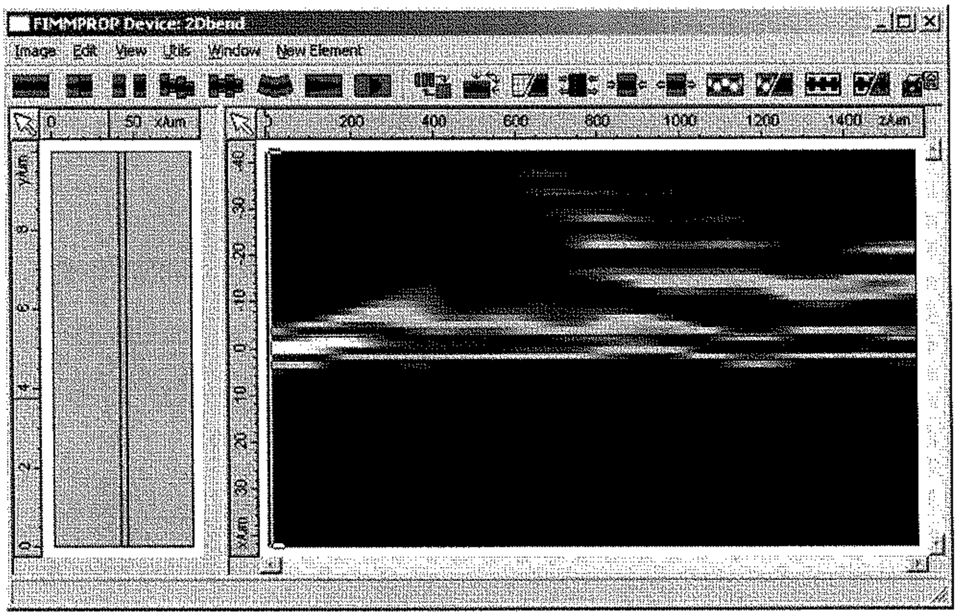


Photon Design®

Using PMLs with FIMMPROP

Photon Design®

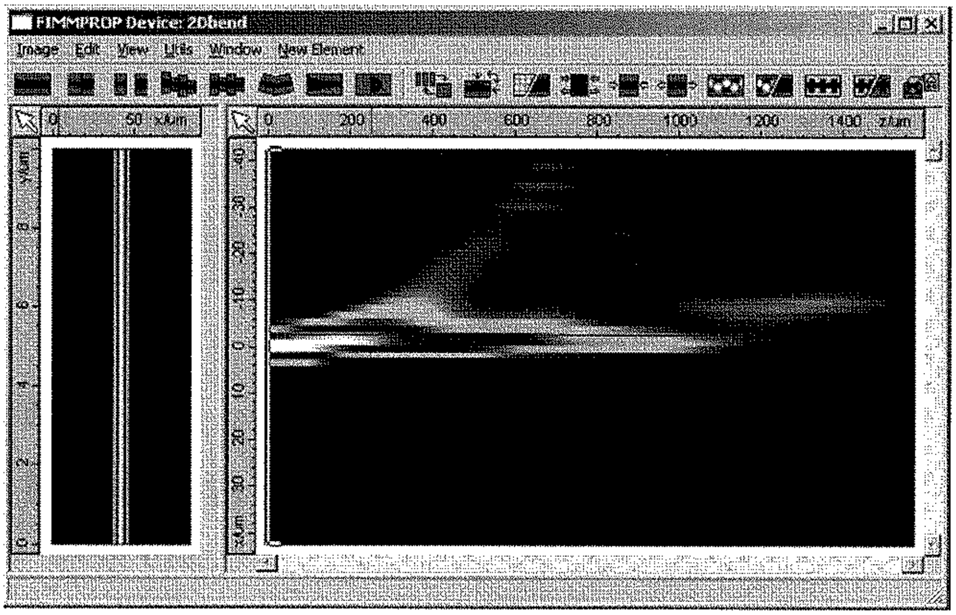
Without Absorbing Boundaries (PMLs)



Photon Design®

Light is radiated away from the waveguide but is reflected back by the sidewalls

With Absorbing Boundaries (PMLs)



The unwanted light is absorbed

Photon Design

Using the I/O Port Section

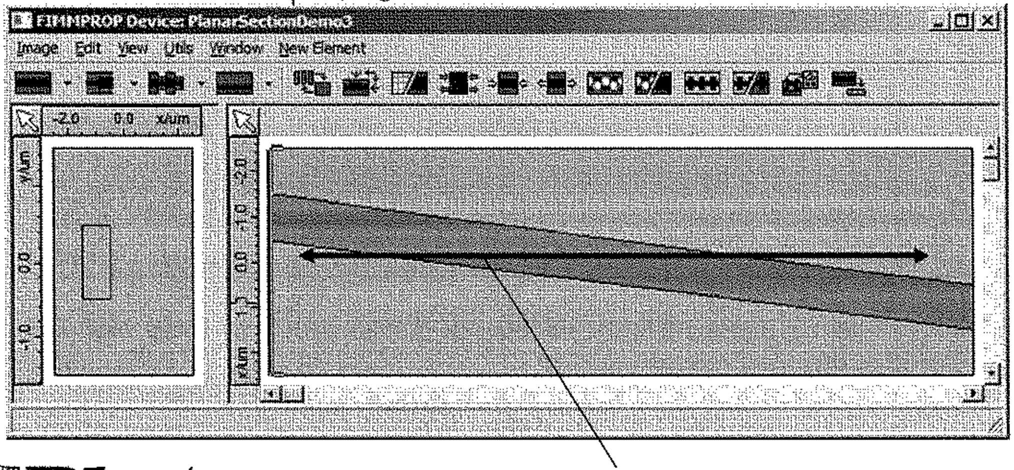
Photon Design

Ports and the I/O Section

S matrix only for z ↔ only
Port to collect light along other directions

- Normally s-matrix is in terms of z-propagating modes – exp(j.β.z)
- Ports allow you to create s-matrices in terms of more natural modes of your structure.

Ports allow mode propagation not in z

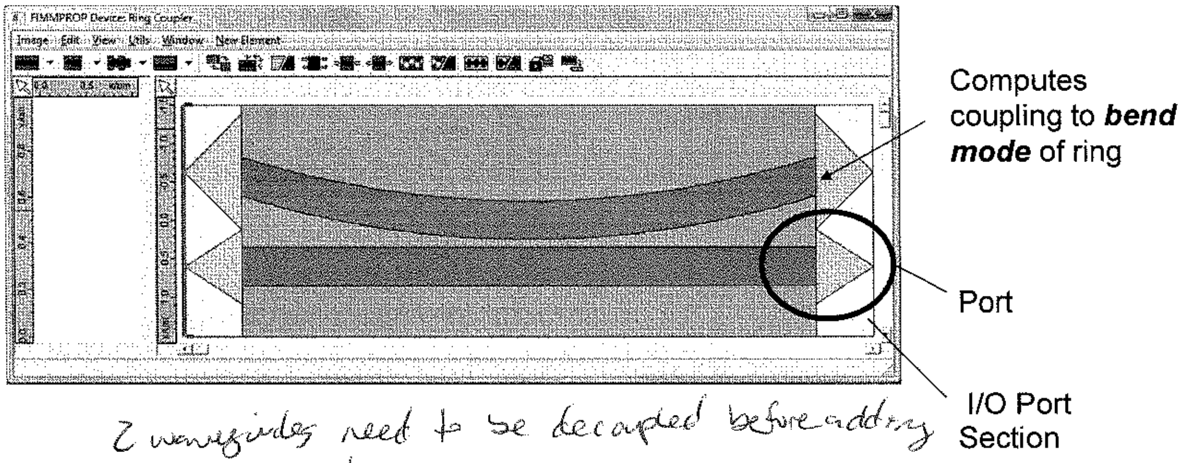


basis set travels in this direction

PML to thick

joint Gauge? No
Like BPM excitation?

Modelling a ring coupler in FIMMPROP

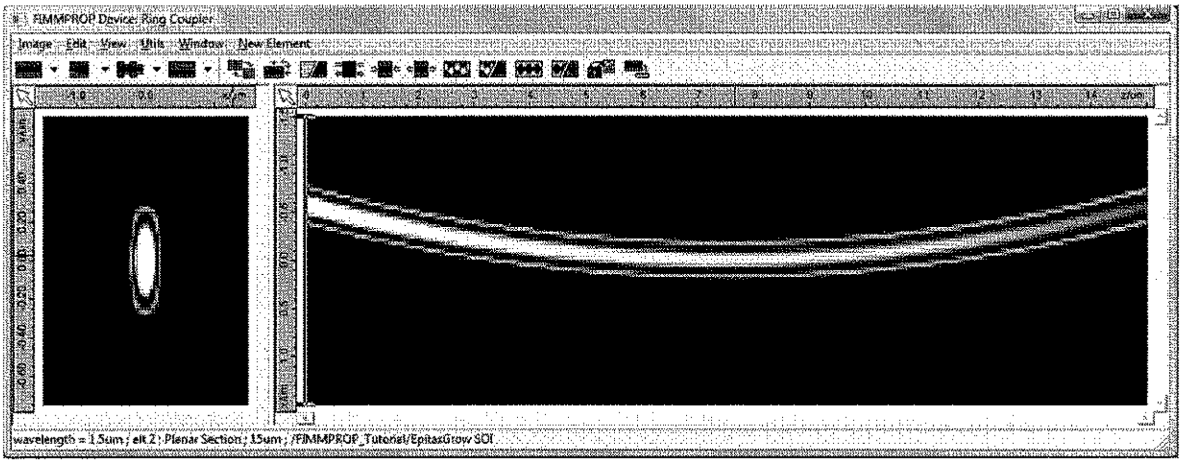


2 waveguides need to be decoupled before adding ports

- Result is an S-matrix of coupling coefficients between the 4 waveguides
- A bend-mode solver computes bend propagation and bend loss
- Results can then be plugged into a simple equation for the ring solution

Can tell which mode & port to excite

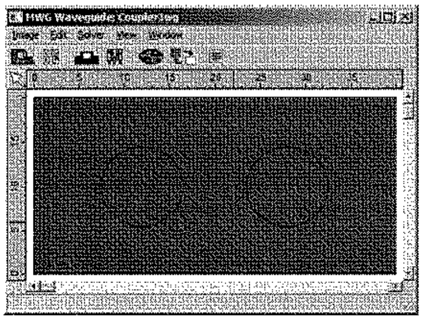
Modelling a ring coupler in FIMMPROP



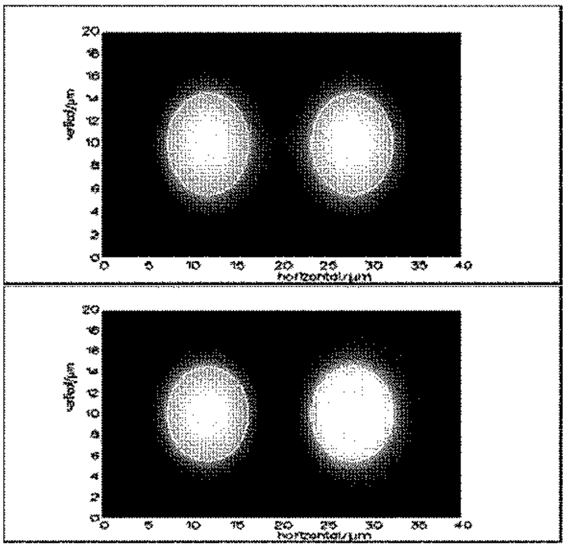
Above: simulation shows light launched into bend mode of ring, with a small coupling to the straight guide visible.

Photon Design®

Modelling a fibre coupler in FIMMPROP



- Breaking cylindrical symmetry
- Use MWG
- Define structure by cross-sections
- Use FDM or FEM Solvers
- Basis set: supermodes

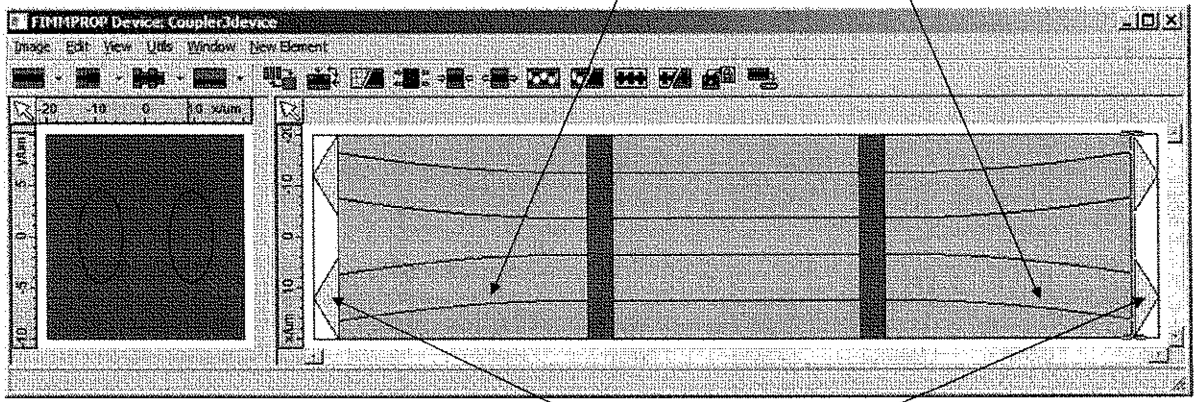


Photon Design®

Modelling a fibre coupler in FIMMPROP

Taper Section
“FORMULA (Z) 1-(1-Z)**2”

Taper Section
“FORMULA (Z) (Z)**2”

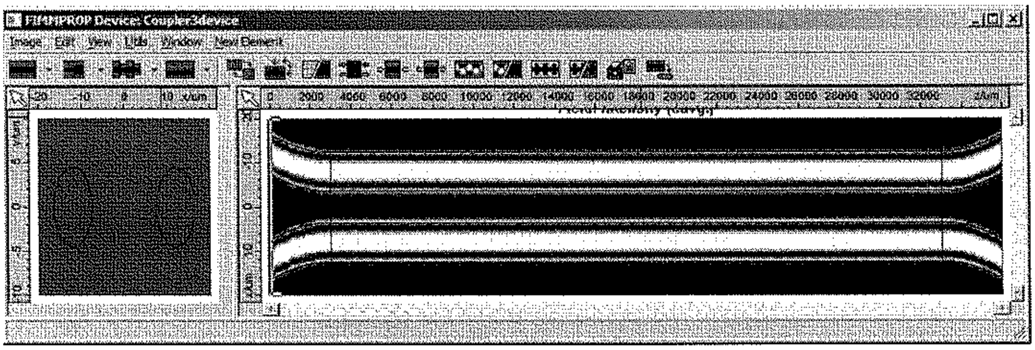


ports

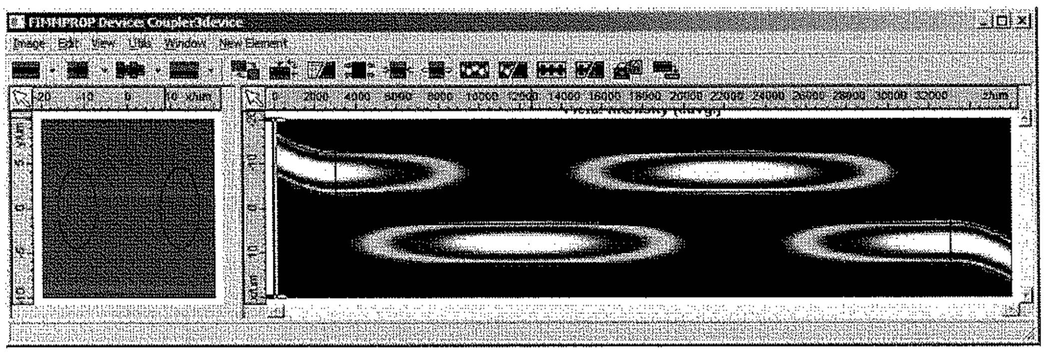
Photon Design

Modelling a fibre coupler in FIMMPROP

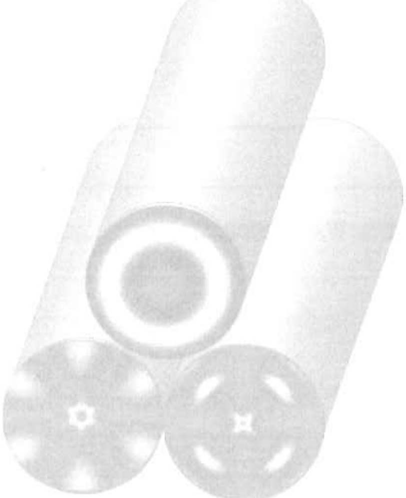
Inject Mode 1 on LHS



without ports

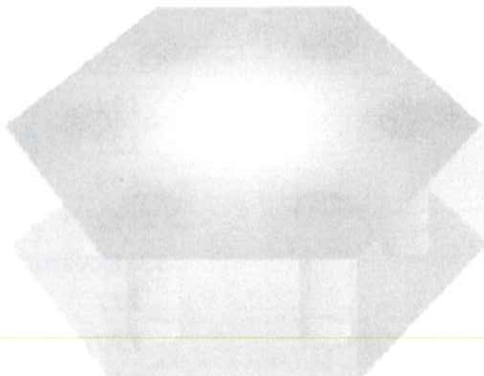


with ports



Training Day 2012

FIMMWAVE / FIMMPROP



CONTENTS

Chapter 1

Introduction

The tutorials contained in this document are centered upon the Photon Design software toolkits FIMMWAVE, and FIMMPROP.

FIMMWAVE is a fully vectorial mode finder designed to model a wide variety of 2D and 3D waveguide structures such as SOI, polymer and etched GaAs/AlGaAs waveguides, as well as single and multicore fibres. The program features several different mode solver methods. It is using these mode solvers that concerns us in the following tutorials Basic FIMMWAVE Usage and Advanced FIMMWAVE Usage.

FIMMPROP is the propagation module integrated with FIMMWAVE. It uses an eigenmode expansion algorithm (EME), where the optical properties of a structure are fully characterised by the local modes of the structure, and coupling matrices between different local modes. It allows great flexibility in constructing complex structures, is fully **bi-directional** and allows extensive visual inspection of the propagating fields. This powerful technique implies that once you have a local mode description of your structure, you can obtain a wide variety of information with little further computation. Becoming skilful in this application is the subject of the following tutorials Basic FIMMPROP Usage and Advanced FIMMPROP Usage.

This document is divided into four sections, each should take approx. 1 hour to work through:

- Basic FIMMWAVE
  - an overview of the core concepts of using the mode solvers.
- Basic FIMMPROP
  - an overview of the core concepts of using the propagation module.
- Advanced FIMMWAVE
  - a more involved discussion of concepts of using the mode solver.
- Advanced FIMMPROP
  - a more involved discussion of concepts of using the propagation module.

Chapter

2

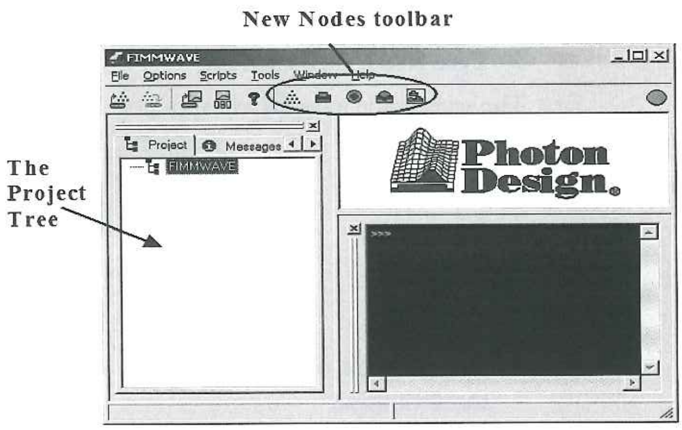
Basic FIMMWAVE Usage

2.1 Example 1 - Finding the modes of a Ridge Waveguide

We will begin with creating a **simple ridge waveguide**. This is done using an *RWG* (**R**ectangular **W**ave**G**uide) *Waveguide node.*

2.1.1 Getting started

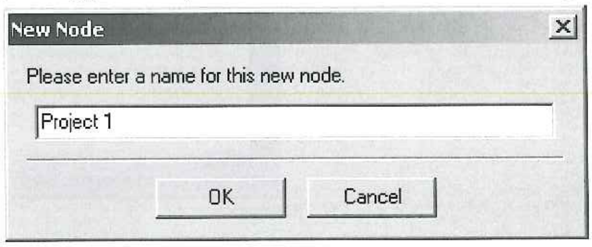
➢ Start up FIMMWAVE. The main window will appear. This should look something like this:



The *new nodes toolbar* shows the node types that can be constructed at this stage. A *node* is any structure that can be created in FIMMWAVE, e.g. waveguides, scanners, etc.

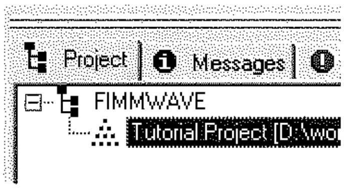
➢ Click on the ⛬ button on the New Nodes toolbar.

The *New Node dialog* will appear for you to enter the name of the new project:



➢ Type **Tutorial Project**, then click **OK**.

The *project* will now be shown with the project symbol ⛬ under the root of the *Project Tree*, shown with the tree symbol (you may have to scroll to the left in the *Project window* to see these symbols):

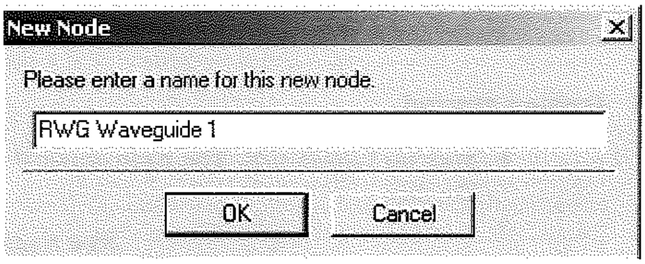


FIMMWAVE supports various waveguide formats. For this example the most appropriate is the *RWG Waveguide* (*RWG*) type.

- To create a new *RWG*, click on the button in the *new nodes toolbar* (it is the first button).

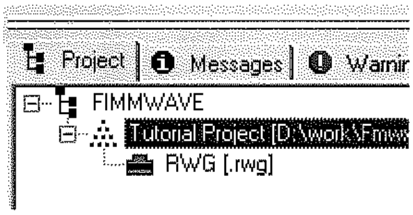
Note that this button will only appear if the Tutorial Project is selected in the *Project Tree*; if it is not shown, click once on the Tutorial Project.

Again, the *New Node dialog* will appear for you to enter the name of the new device:

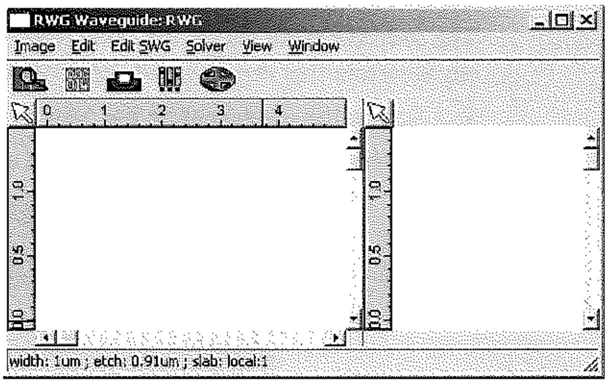


- Type **RWG**, then click **OK**.

The waveguide icon will appear in the *Project Tree* as shown



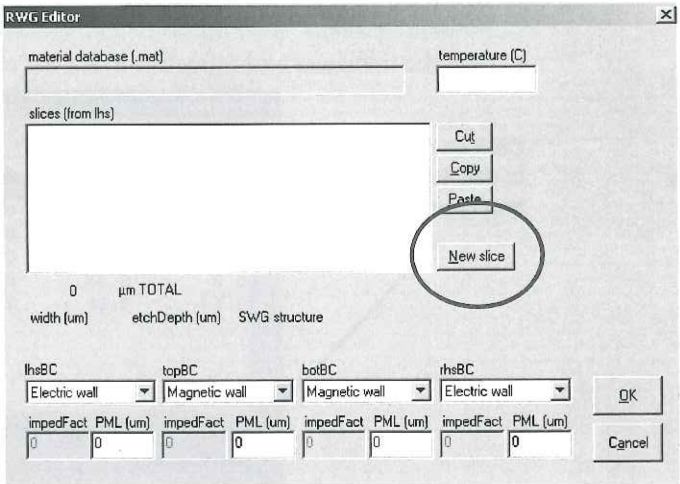
- Double click on this waveguide icon to open the *Waveguide Editor*. The following should appear.



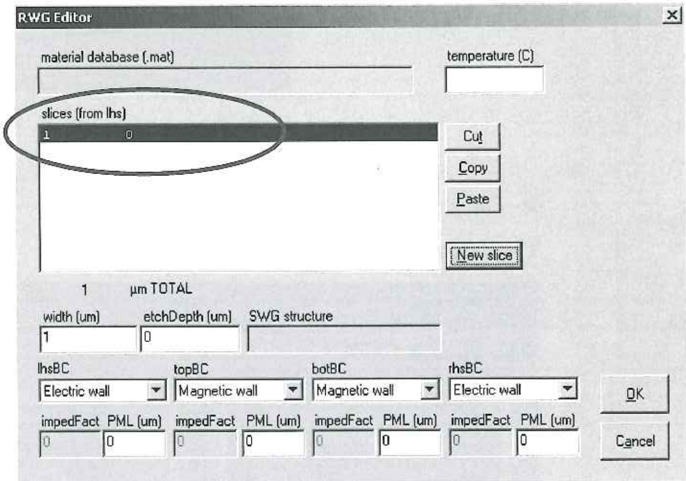
2.1.2 Building the waveguide

An *RWG* is constucted by a *slice*, or a series of *slices*. It is within these *slices* that the epitaxial layer structure is defined.

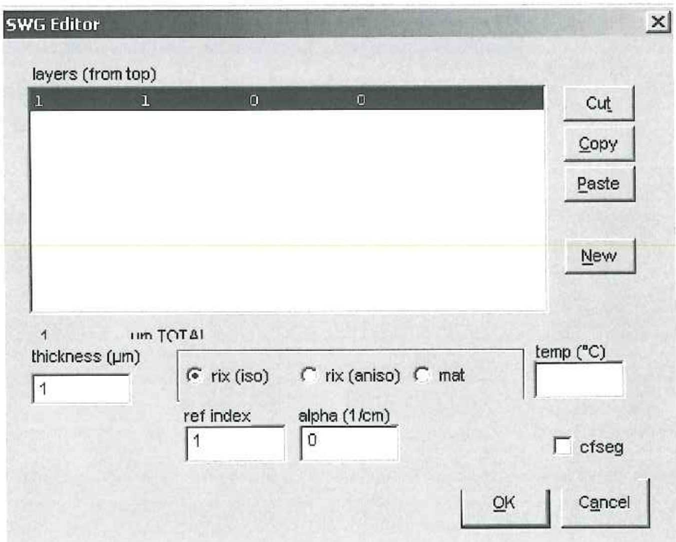
➢ Click on the  button. This will show the *RWG* Editor panel.



➢ To add a new slice click on the **New Slice** button (highlighted above).

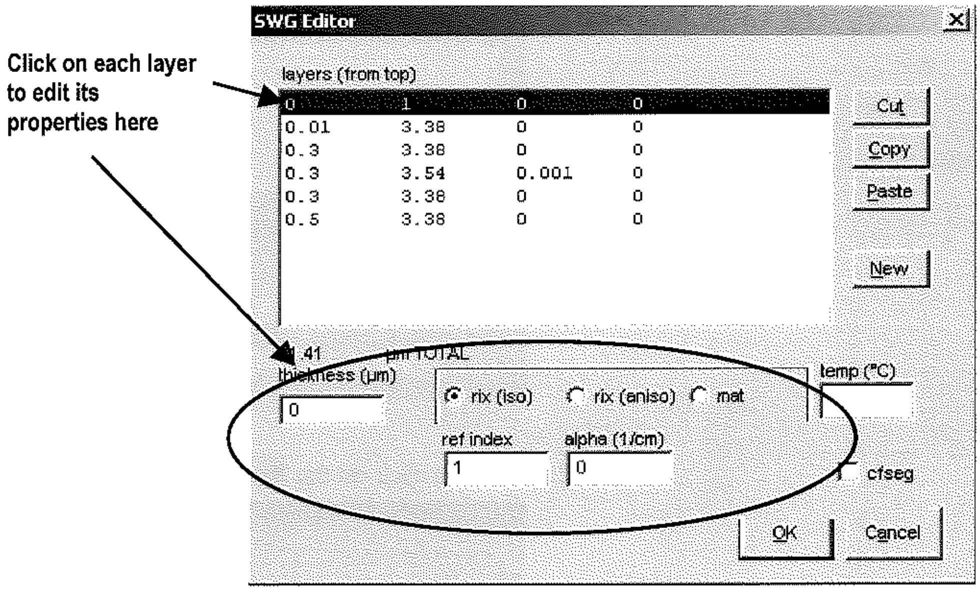


➢ Double-click on the slice in the list (highlighted above). This will show the *SWG Editor* panel.



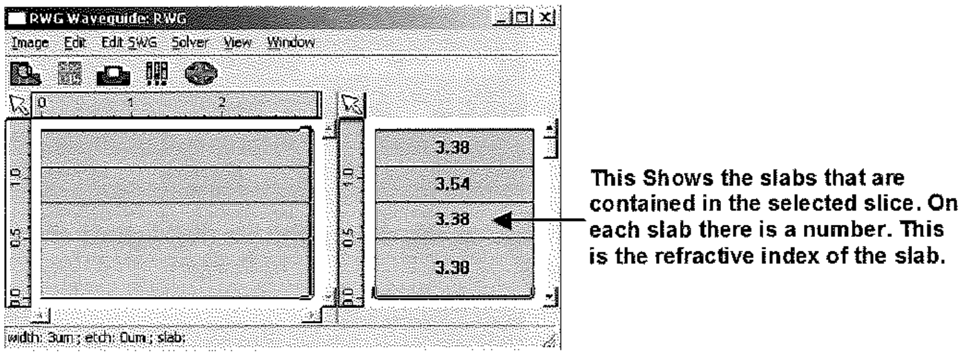
It is on this panel that the epitaxial layer structure is set.

- Click on the **New** button 5 times to create 5 new *layers* (giving a total of 6).
- Edit the thickness and refractive index for each *layer* to what is shown here:



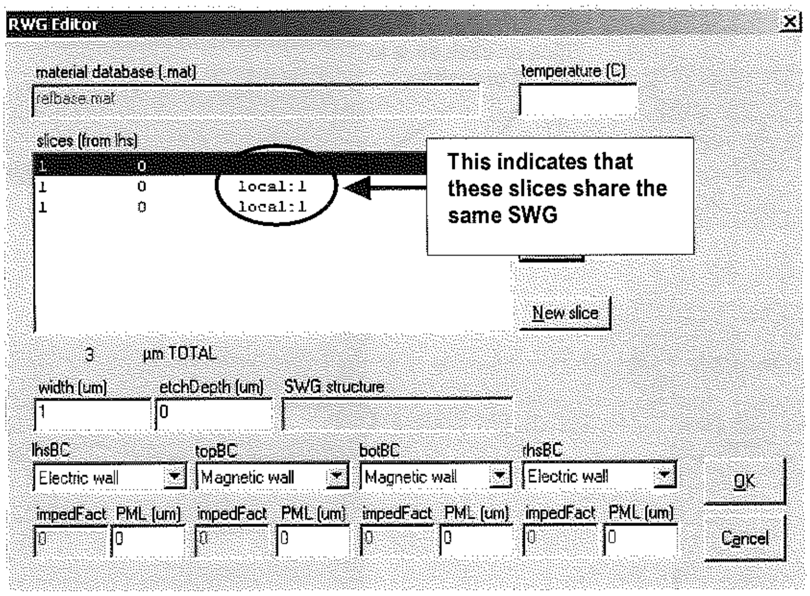
- Click **OK** and **OK** again, when finished.

The *Waveguide Editor* should now look like this:



Whenever possible, it is convenient for all slices to reference one *SWG* layer structure. This is because changing the *SWG* structure of one slice will automatically update all of them.

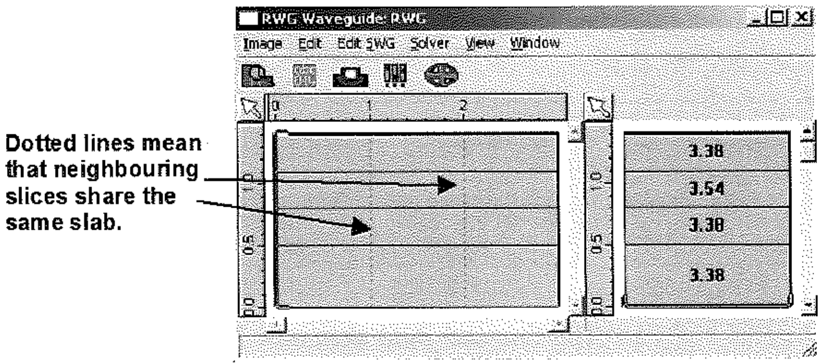
- Again click on the [icon] button. This will show the *RWG* Editor panel.
- Highlight the first slice in the list and click on **Copy**, then click on **Paste** two times. This will create two copies of this *slice*.



The second and third slice refer to the same *SWG* structure of the first slice.

➢ Click **OK** when finished.

The *Waveguide Editor* should now look like this.

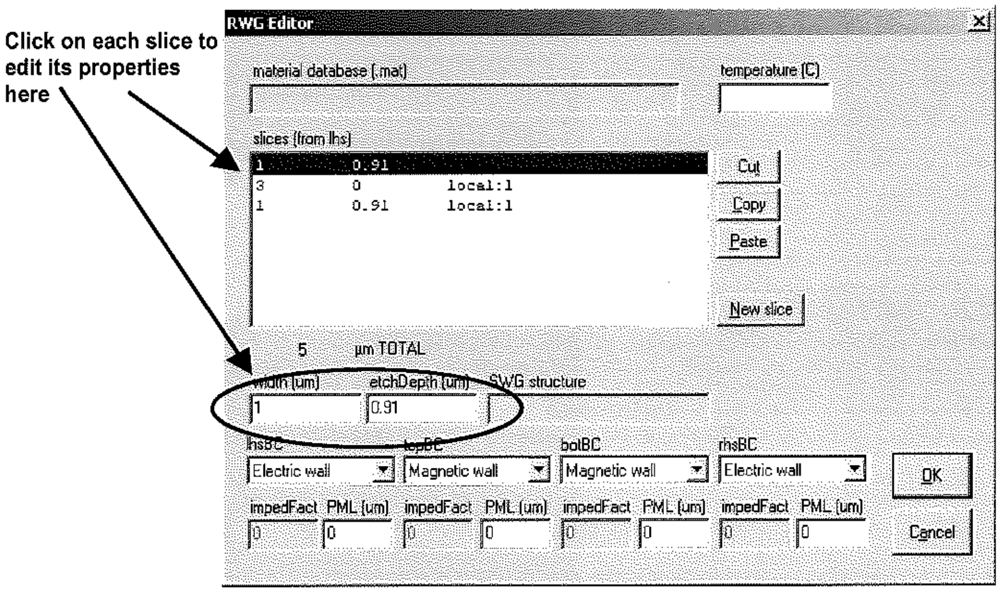


2.1.3 The Etch Mechanism

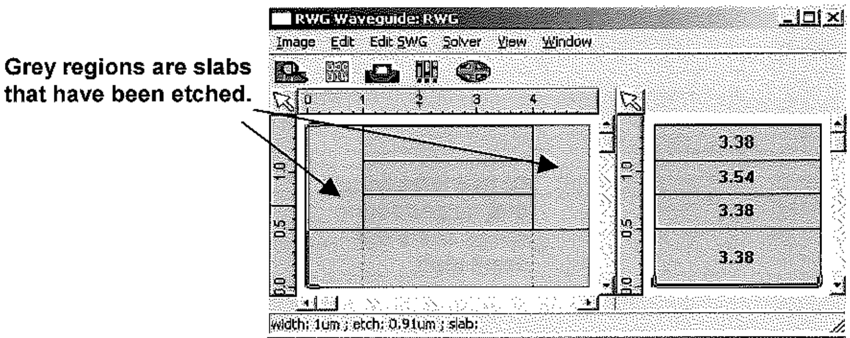
Next we need to define the ridge structure. We do this by **etching** the first and third *slices*. We must also set the *slice* widths.

➢ Click on the [button icon] button again - this will show the *RWG* Editor panel.

➢ Set the **width** and **etchDepth** of each slice as shown here:



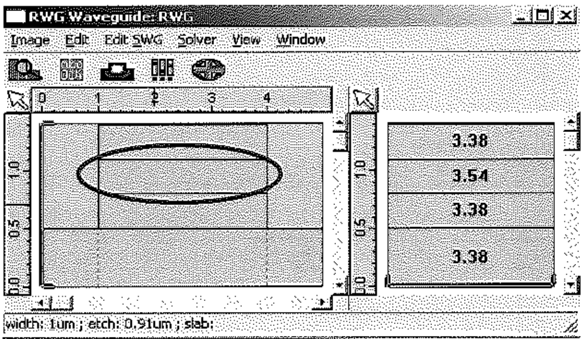
➢ Click **OK** when finished. You should now see the following.



The first and third *slices* now have a grey region extending 0.91 µm from the top surface. This indicates an *etched region*. The refractive indices of these regions are defined by the refractive index of the top layer. In this case, it is air. Using this functionality, you can create quite complex waveguides with just one vertical *SWG* layer structure and different etch depths in each *slice*.

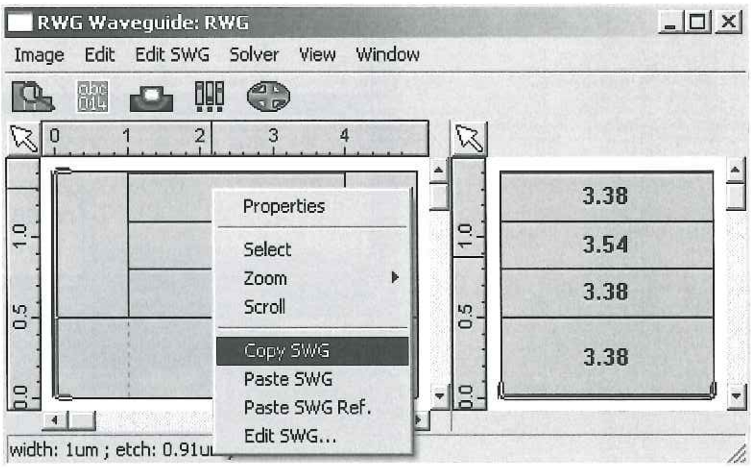
2.1.4 Specifying a confinement factor region

For this example, we wish to specify a region for which we want to calculate the confinement factor for modes. We want to specify this for the waveguide core. Ths layer is highlighted in the figure below.

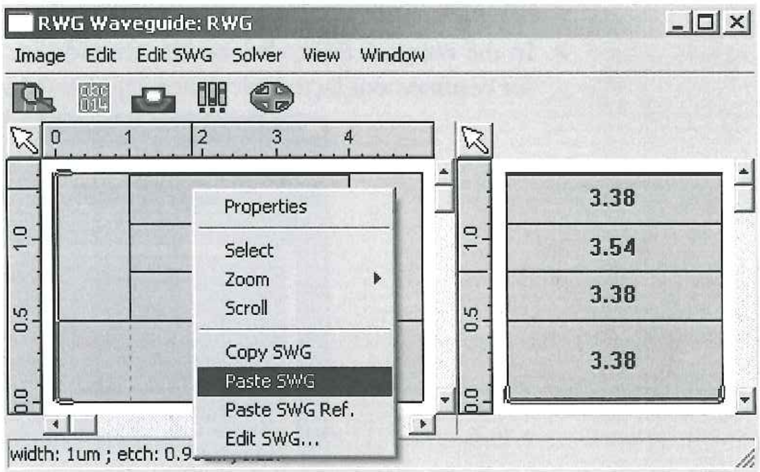


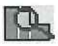
Currently, the middle slice references the first slice in the list, thus any changes made will affect all the slices in the waveguide. Therefore, we need to cancel this reference. This is easily done.

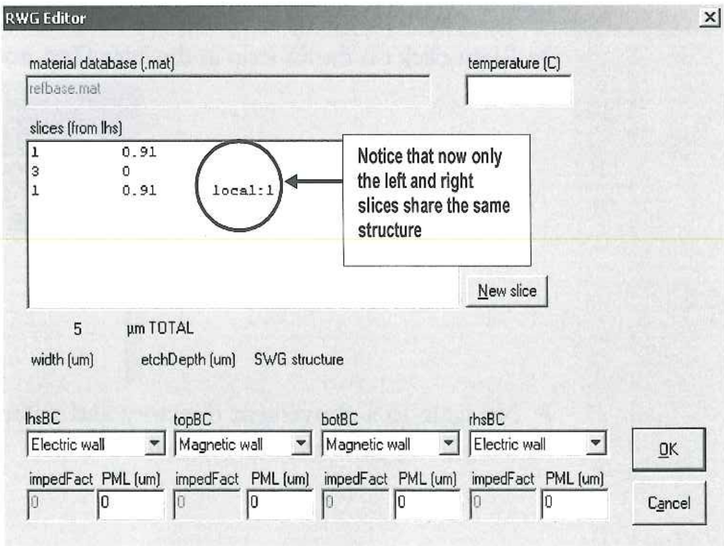
➢ Right-click on the middle slice, select **Copy SWG**



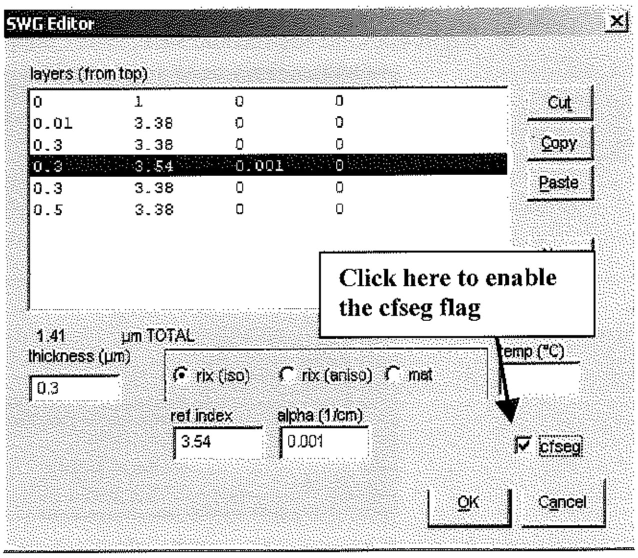
➢ Right-clicking on the middle slice and selecting **/Paste SWG**.



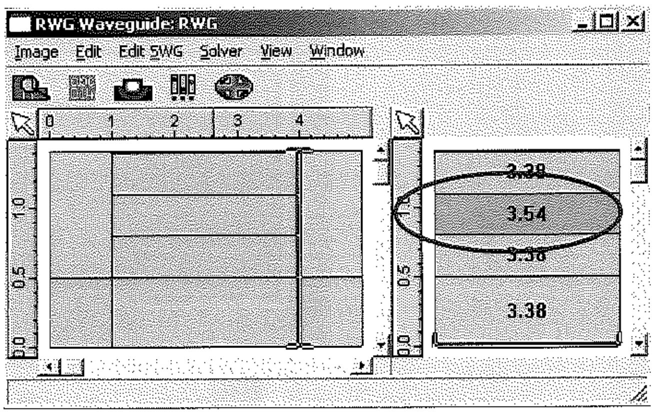
➢ Click on the  button - this will show the *RWG* Editor panel again.



- Double-click on the middle slice. This will show the *SWG Editor* panel.
- Highlight the core region and click in the **cfseg** box. (as shown below).

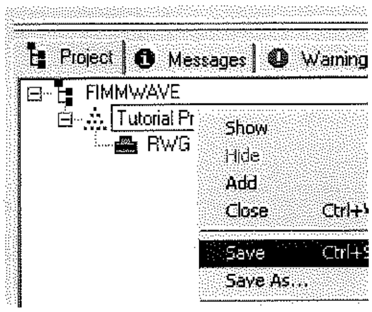


- Press **OK** when finished.
- In the *Waveguide Editor*, click on the second slice. You should see the region selected for confinement factor calculation highlighted in a different colour.



Before we go any further, we should save the project.

- Right-click on the ∴ icon in the *Project Tree* and select **/Save**.



- Navigate to a convenient directory and enter **Tutorial Project** in the file selector and press **Save**. The project is now saved.

2.1.5 Finding the modes

We now go on to locate the modes of the waveguide.

In this tutorial, we want to display the eigenvalues of the calculated modes as effective indices as opposed to propagation constants.

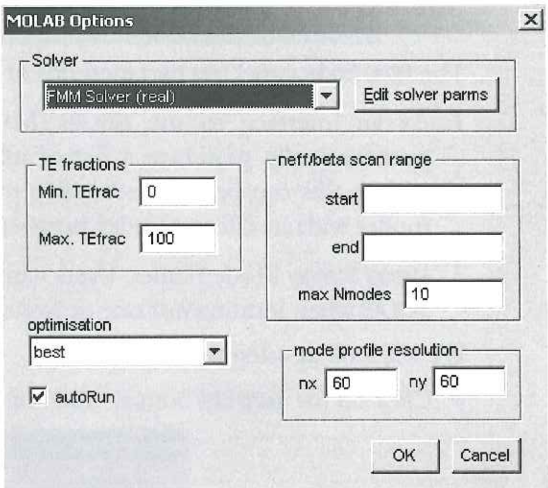
➢ In order to do so, go to the main FIMMWAVE window and go to the menu **Options/Application...**

➢ Set **evalType** to *effective index*.

➢ Click **OK** to close the panel.

➢ Back to the *Waveguide Editor*, click on the button. This will show the *MOLAB Options* panel.

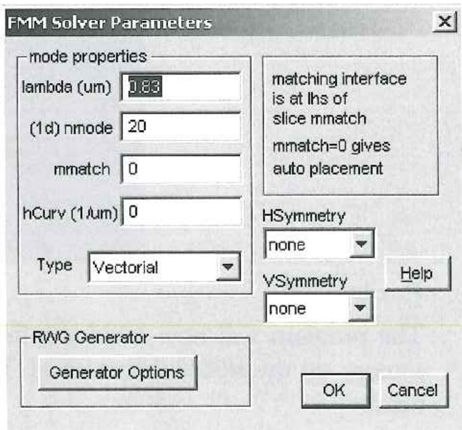
➢ Select the *FMM Solver (real)* in the list of solvers.



➢ Click on the **Edit solver parms** button. This will show the *FMM Solver parameters* panel.

➢ In the **lambda (um)** box, type in 0.83.

➢ In the **(1d) nmode** box type in 20.

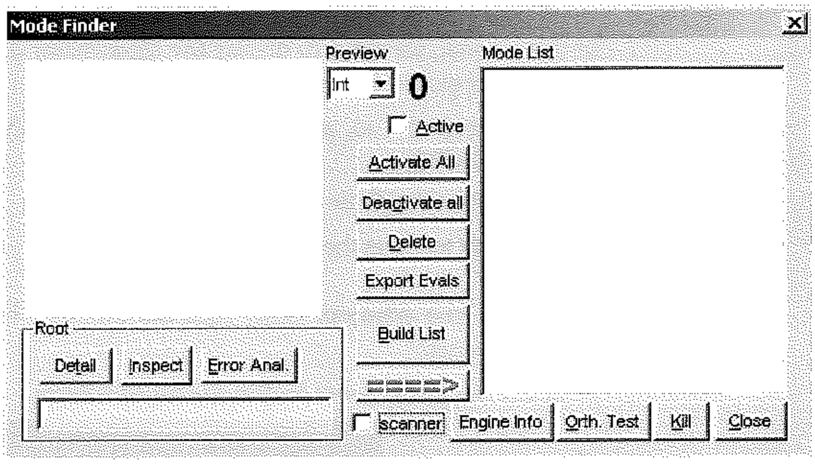


➢ Click **OK**.

➢ Click **OK** again.

➢ In the *Waveguide Editor*, click on the button.

The program will do some preliminary calculations and after a short delay, the *Mode Finder* panel will appear.

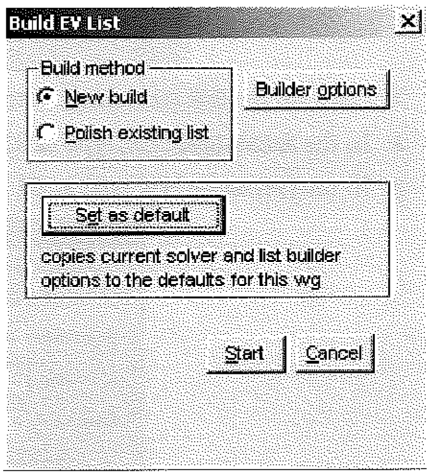


The *Mode Finder* panel has two methods of operation:

1. As an interface to the *MOLAB* (**MO**de **L**ist **A**uto **B**uilder). **The *MOLAB* can automatically generate a list of all the modes of your structure**. If the user wishes, this can be done according to certain criteria, e.g. the first 5 modes, or all modes with an effective index between, for example, 3.25 and 3.0 etc.
2. *Manual Scanner* Mode Finder. Users with more experience often find it faster than the *MOLAB* when locating just one or two modes.

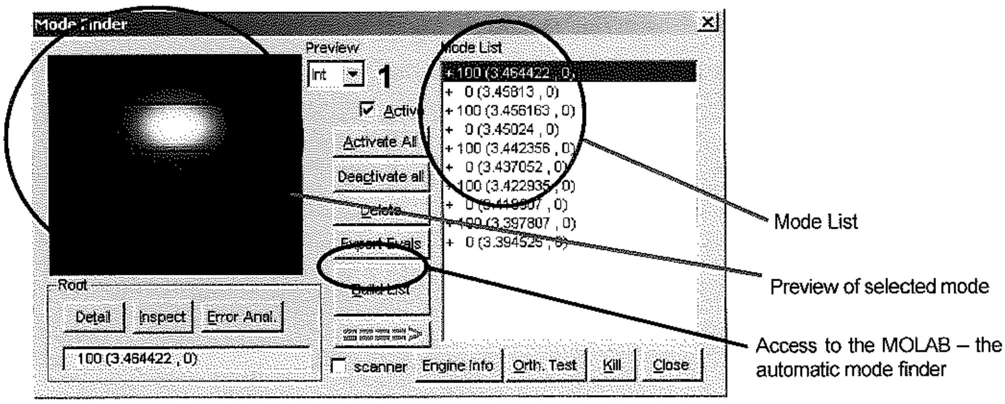
We will look at using the *MOLAB*.

➢ Click on the **Build List** button - the Build EV List panel will appear.



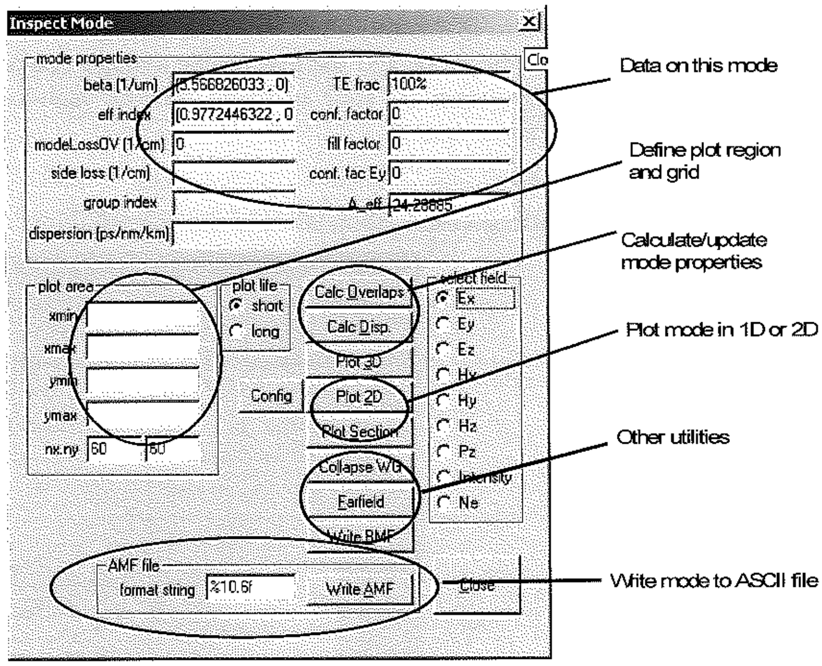
➢ Click **Start**.

The program will now build the first 10 modes of this structure, according to the settings on the *MOLAB Options* panel. These will be inserted into the *Mode List* as shown below.



➢ Double-click on the first mode in the Mode List. The *Inspect Mode* **Panel** will appear.

2.1.6 The Inspect Mode Panel



You use this panel to do any of the following:

α) Plot a 2D profile of the mode - any field component or the mode intensity;

β) Plot a 1D cross section of the mode at a given horizontal or vertical position;

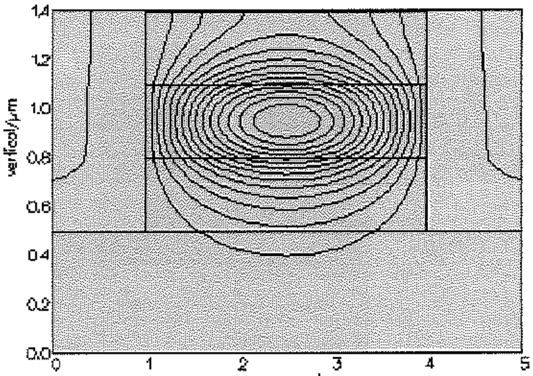
χ) Produce an ASCII file of either of the above;

δ) Calculate the confinement factor and/or loss of your mode;

ε) Calculate the Group Index and/or Dispersion;

φ) Calculate the Effective Area.

➢ Click the **Config** button beside the Plot 2D button and ensure that Plot Type is set to *Contour Plot.* Then click **Plot 2D** - a contour plot will appear, indicating that this is the zero-order TE mode.



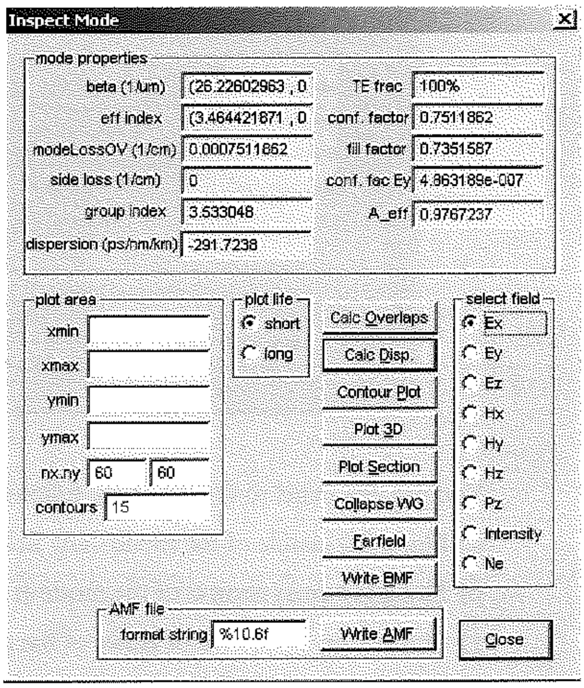
➢ Click **Calc Overlaps.** After a short delay, some data will appear in the region called *Mode Properties*. This will indicate a confinement factor of 0.751 in the waveguide core (remember the **cfseg** flag in the vertical layer structures).

It will also show TE frac = 100%, i.e. this is a purely Ex polarised mode. This has been calculated by analysing the whole mode profile and is an accurate value.

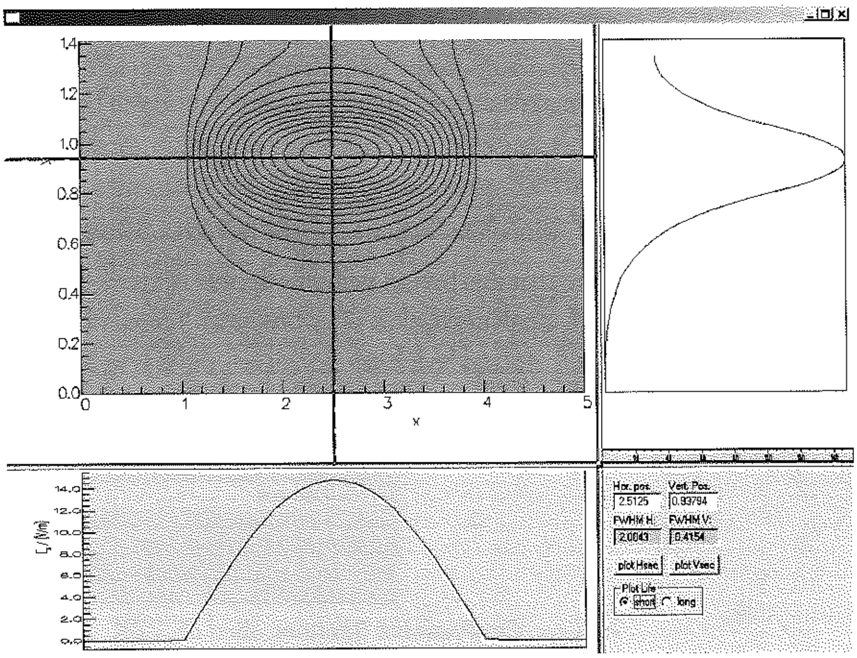
Notice also that **modeLossOV** = 0.00075/cm. This is obtained using an overlap integral between the mode and the loss profile and is a reasonably accurate estimate for low-loss waveguides. (If your structure contains high-loss materials, you will need the *Complex Engine* option).

➢ Click **Calc Disp**

The dispersion and group index of this waveguide are also calculated.



To investigate the 1D cross section of your mode, select *Ex field*, click on *Plot Section*. You can specify the desired horizontal and vertical cross section by using the cross hairs.

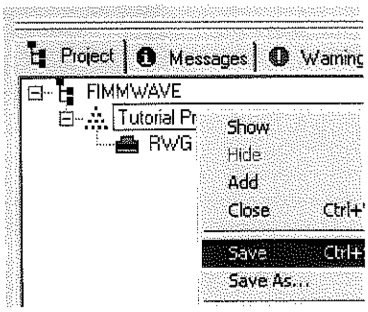


2.1.7 Saving the project

- Close the *Plot Section* panel
- Click **Close** on the *Inspect Mode* panel
- Click **Close** on the *Mode Finder* panel
- Click **Yes** to save the Mode List.

The program will remember the Mode List when you close the program – ready for when you next open the waveguide.

- Right-click on the ∴ icon in the *Project Tree* and select **/Save**.



The project is now saved.

2.2 Example 2 - Using the WG Scanner

This extremely useful tool will **automatically** allow you to see how a variety of mode parameters change as you alter the shape or refractive index of the waveguide.

The WG Scanner is given two versions of the waveguide, representing the structure at the beginning and the end of the scan. Internally, a third structure is then constructed which is a weighted average of the two waveguides. By this means, you can scan almost any parameter you wish.

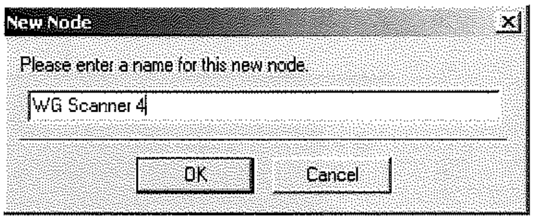
In this example, the WG Scanner will be used to investigate the change in mode confinement factor as a function of the refractive index of the core region. The waveguide created in the previous section will be used.

2.2.1 Getting started

- Click on the button in the *new nodes toolbar*

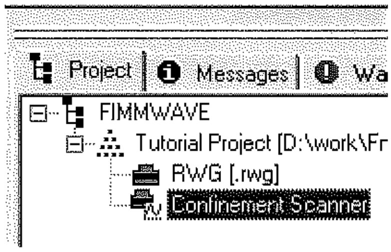
Note that this button will only appear if the Tutorial Project is selected in the *Project Tree*; if it is not shown, click once on the Tutorial Project.

Again, the *New Node dialog* will appear for you to enter the name of the new device:

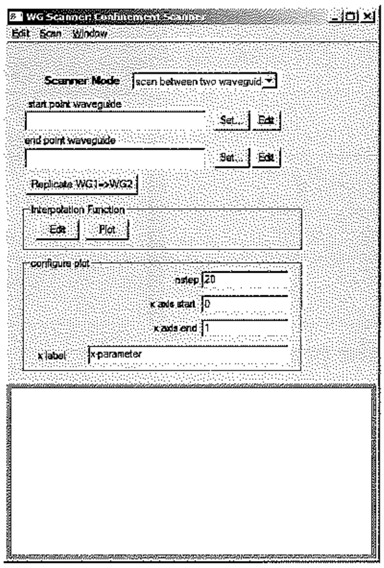


- Type **Confinement Scanner**, then click **OK**.

The waveguide icon will appear in the *Project Tree* as shown below.

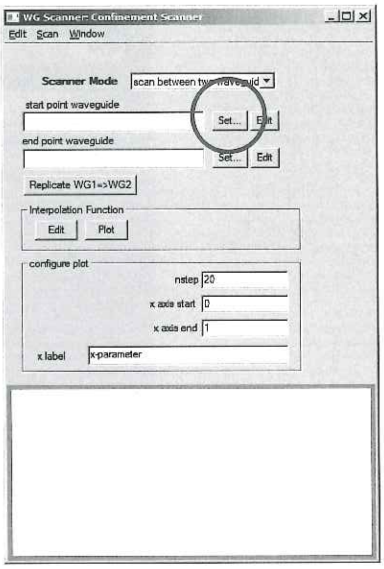


- Double click on this waveguide icon to open the *WG Scanner*. The following should appear.

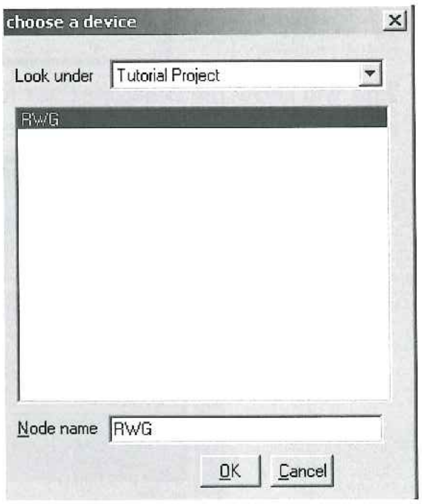


2.2.2 Setting up the WG Scanner

- Click on the **Set** button next to the **start point waveguide** box.



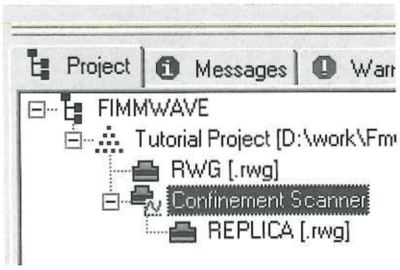
This will bring up the Choose a Device panel.



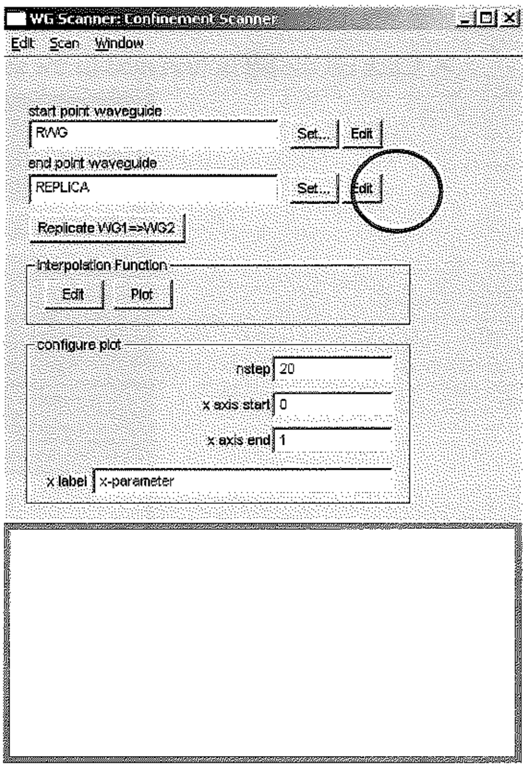
- Highlight **RWG** – the waveguide we set up in the first example.
- Click **OK**

You should now see "RWG" in the **start point waveguide** box.

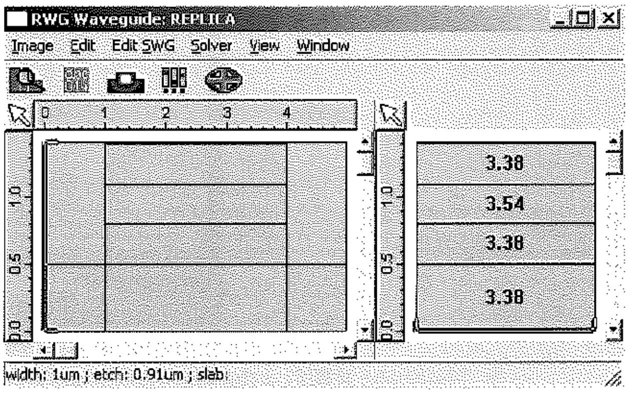
- Click on the **replicate WG1=>WG2** button. This will make a copy of the waveguide. You should see this copy appear in the *Project Tree*.



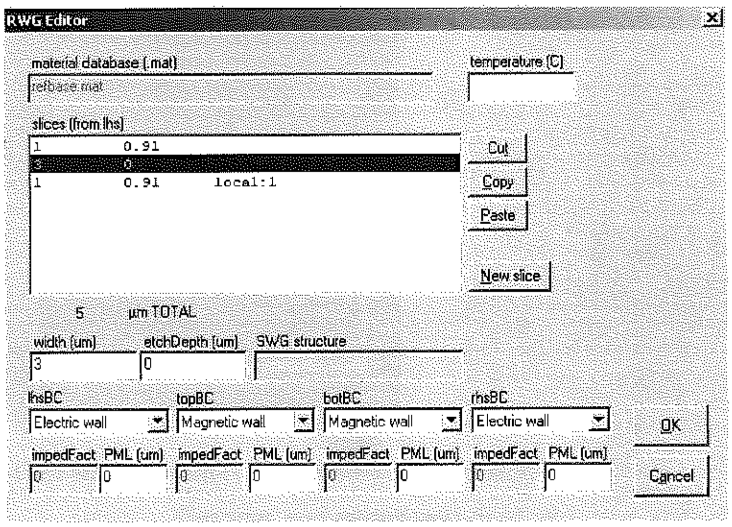
- Click on the **Edit** button next to the **end point waveguide** box.



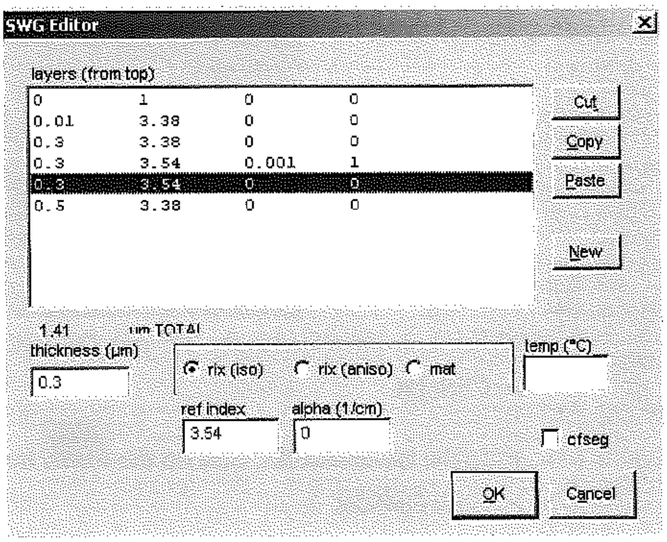
This will show the *Waveguide Editor* for the copied waveguide. It is currently the same as the original, however we will now change the refractive index one of the layers.



➢ Click on the button. This will show the *RWG* Editor panel.



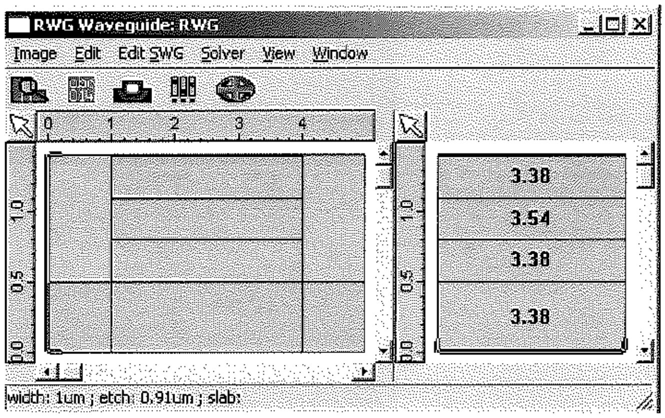
- Double-click on the middle slice in the list (highlighted above). This will show the *SWG Editor* panel.
- Set the **ref index** of the fifth layer (from the top) to be 3.54 as shown below



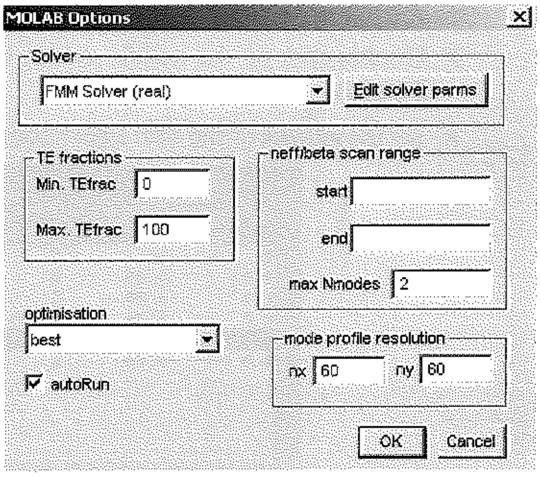
- Click **OK**
- Click **OK** again

Before we proceed with the scan, we need to determine which waveguide modes we are interested in studying.

- Click on the **Edit** button next to the **start point waveguide** box. This will show the *Waveguide Editor* for the waveguide that we created in the first section.



- Click on the button. This will show the *MOLAB Options* panel.
- In the **maxNmodes** box, type in 2.



We are only interested in the TE and TM fundamental modes.

- Click **OK**
- In the *Waveguide Editor*, click on the button. The *Mode Finder* panel will appear.
- Click **Build List**
- Click **Start**
- Click **Yes** when asked "delete all existing modes?"

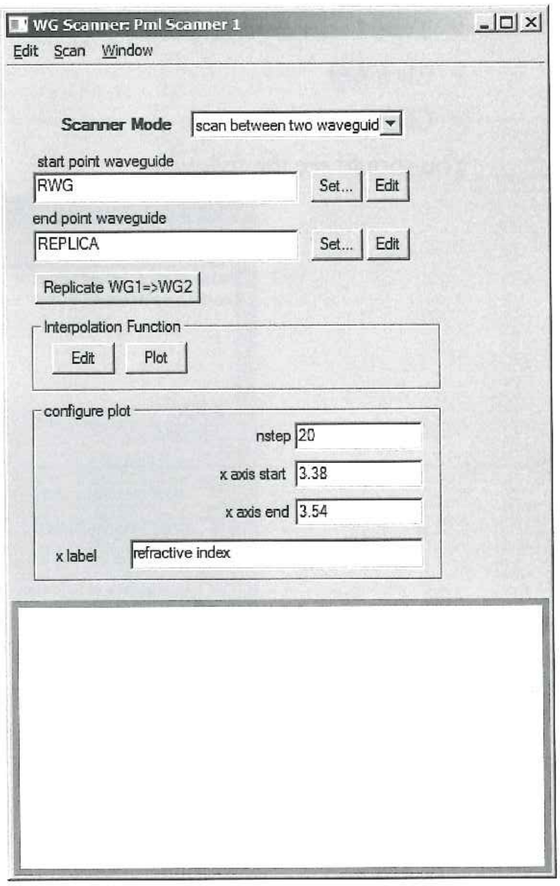
The MOLAB will find only the two fundamental modes.

- Click **Close**
- Click **Yes** when asked "Save mode list changes?"

The *WG Scanner* tool does not know which parameters are being scanned (you could change several things at once). We need to define the labels for the x-axis. In addition, we need to specify the number of points we wish to calculate.

- Type 3.38 in the **x-axis start** box
- Type 3.54 in the **x-axis end** box
- Type "refractive index" in the **x-label** box

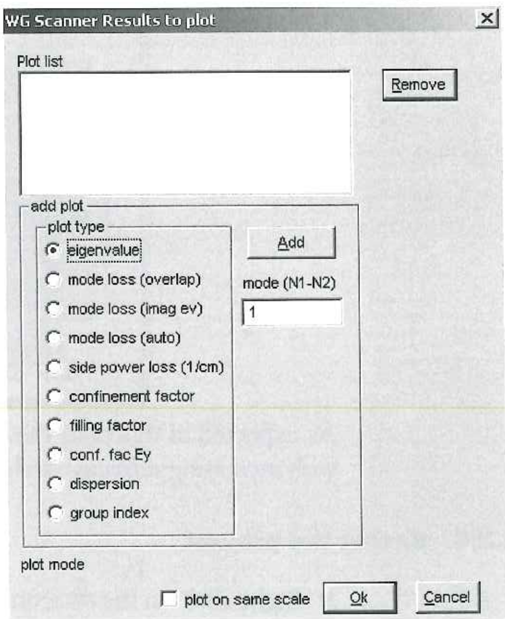
You should see the following.



We are now ready to begin the scan.

2.2.3 Performing the scan

➢ On the *WG Scanner* select **/Scan/Start**. You should see the following.



We want to display the **confinement factor** for the two modes about to be scanned.

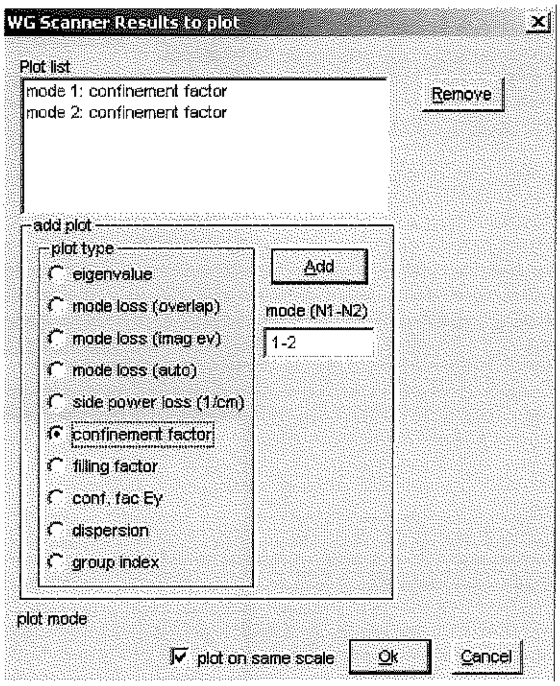
➢ Click on **confinement factor** in the **add plot** box

➢ Type 1-2 in the **mode (N1-N2)** box.

➢ Click **Add**.

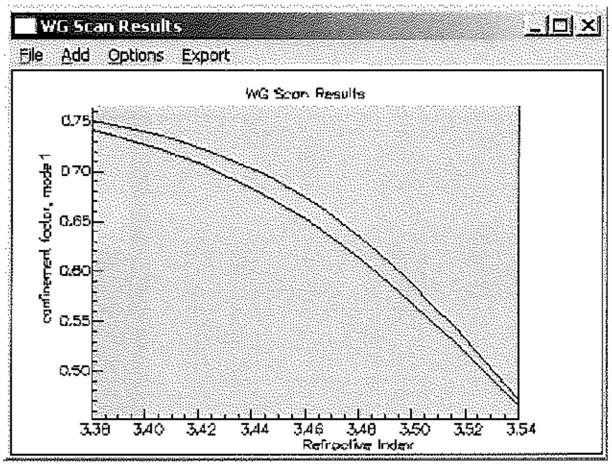
➢ Click **plot on same scale**

You should see the following



➢ Click **OK**

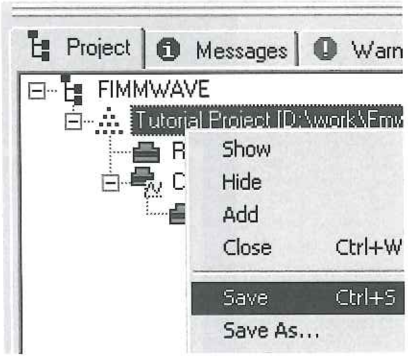
After a minute or two, you will have a plot of the confinement factors of the two modes shown below.



As expected in this case, the confinement factors for both modes in the core decrease with increasing refractive index of this underlying slice.

2.2.4 Saving the project

➢ Right-click on the icon in the *Project Tree* and select **/Save**.



The project is now saved.

Note: The scan data is also saved with the project file so that when you close and re-open the project you can again plot the above graph by simply simply selecting **/scan/Display data** from the *WG Scanner* window.

2.3 Example 3 - Creating a diffused waveguide

In FIMMWAVE you can easily define structures with mixed geometries and diffuse refractive index profiles. This is done using the **M**ixed **W**aveguide (*MWG*) format.

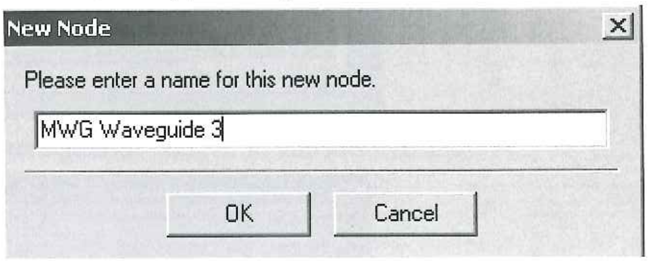
In this example, we will create a diffused buried waveguide (typically produced in lithium niobate processes).

2.3.1 Getting Started

Ensure that the Tutorial Project is selected in the *Project Tree.* Click on it once if it isnt.

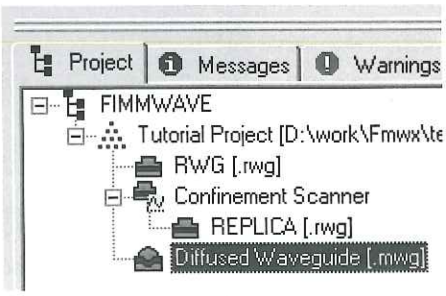
- Click on the button in the *new nodes toolbar*

Again, the *New Node dialog* will appear for you to enter the name of the new device:

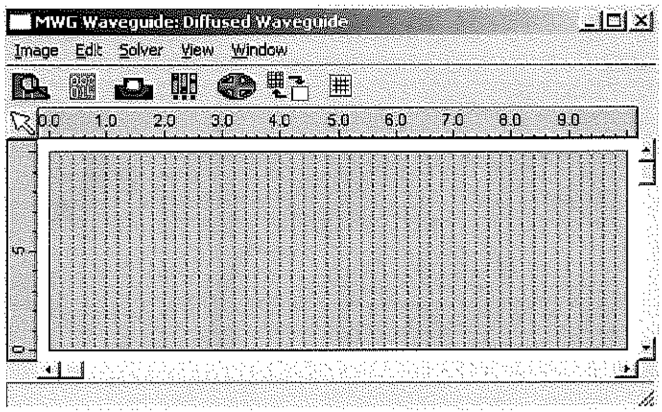


- Type **Diffused Waveguide**, then click **OK**.

The waveguide icon will appear in the *Project Tree* as shown



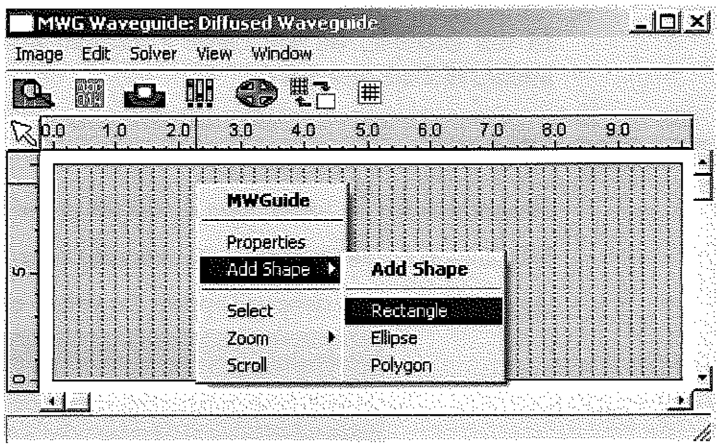
- Double click on this waveguide icon to open the *MWG Waveguide Editor.* The following should appear.



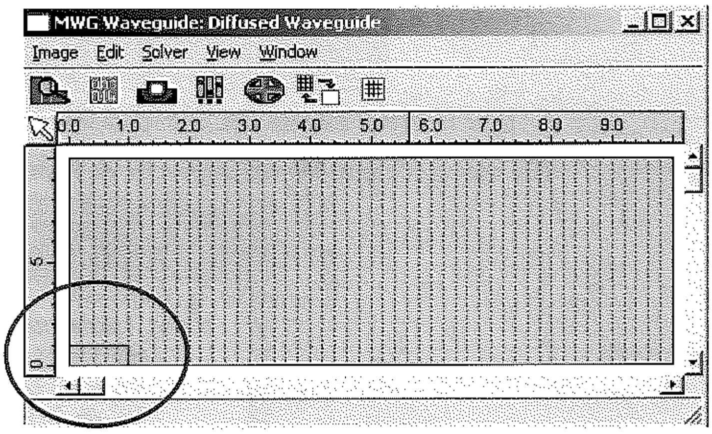
This is currently an empty region with refractive index 1.0 and window size 10μm by 10μm.

2.3.2 Building the waveguide

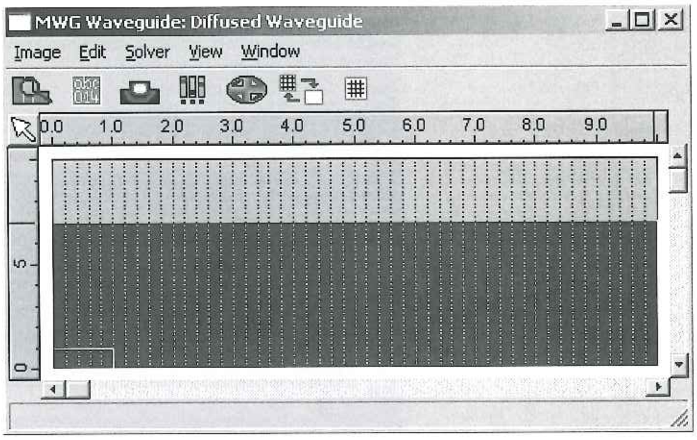
- First of all click on the button in the toolbar then set the grid parameter to 0.1 in both the x- and y-directions. Click **OK** to return to the MWG Waveguide.
- Add a rectangular shape by right clicking in the client area as shown.



- Click in the centre of the new rectangle and drag it to the bottom left hand corner.

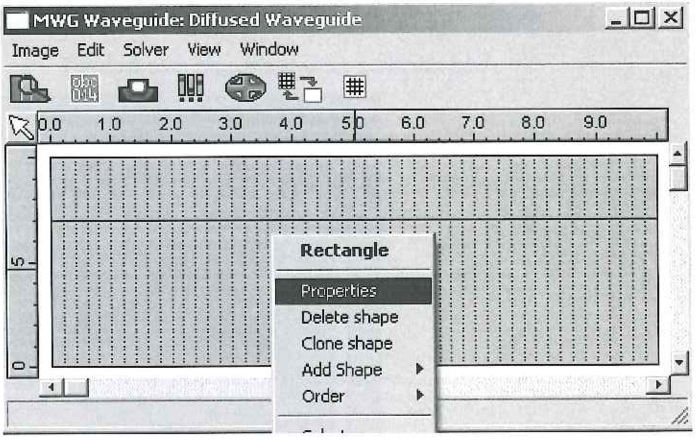


- Click just inside the top right corner of the rectangle, hold down the mouse button and resize the rectangle until it has width 10um and height 7um.

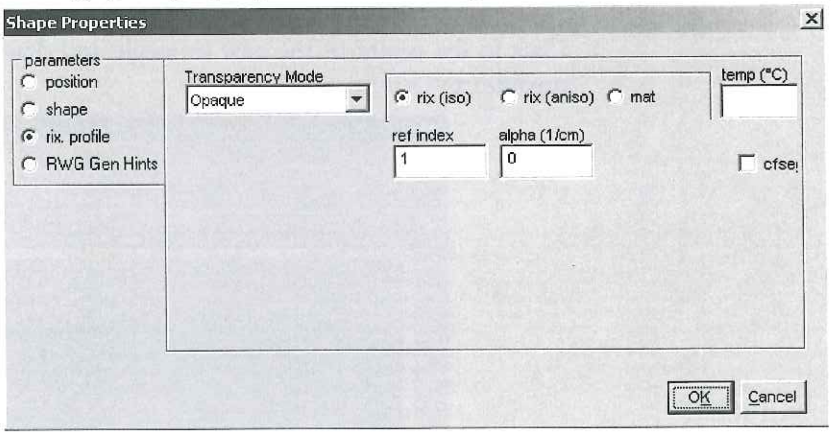


The refractive index of the new shape is set by default to *1.0*. We want to change this.

➢ Right-click on the shape and select **properties**.



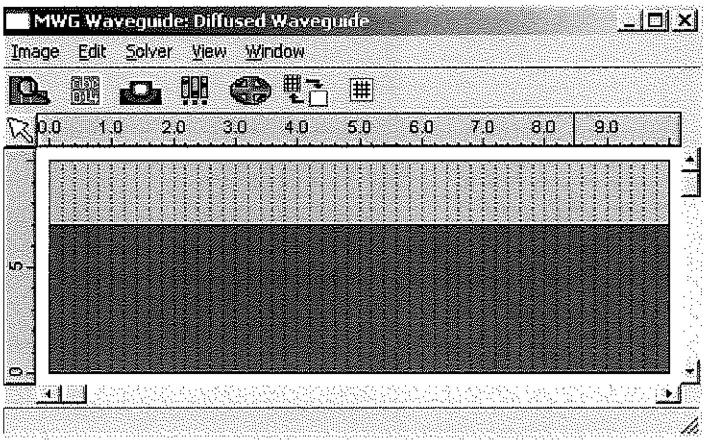
This will bring up the property editor for the shape. You should see the following.



➢ Set **ref index** to *1.5*.

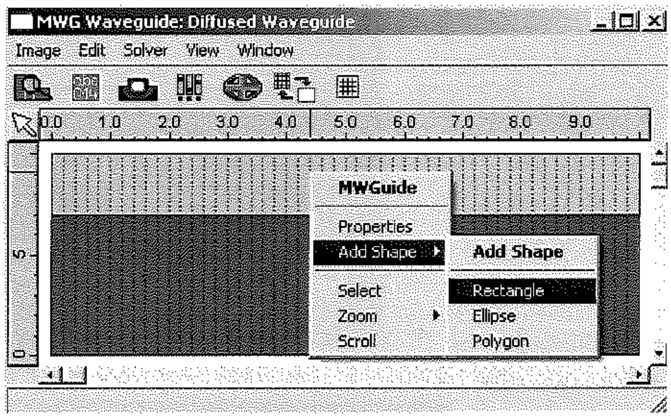
➢ Click **OK**

The colour of the rectangle should now change to reflect this new value as shown.

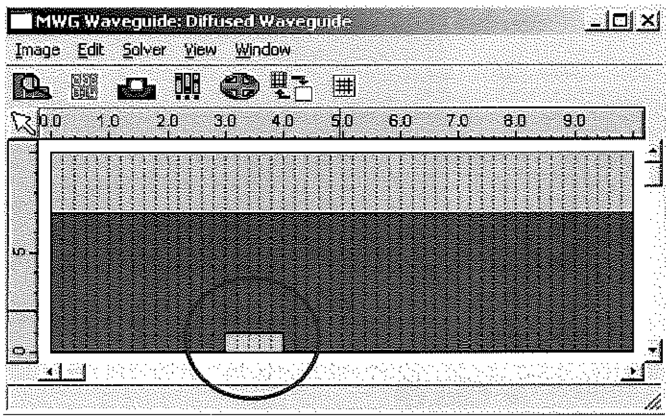


Next, we need to add a diffused refractive index profile so that the refractive index is 1.6 on the top-centre of this rectangle, and decreases gradually to 1.5 away from this point. We do this the following way:

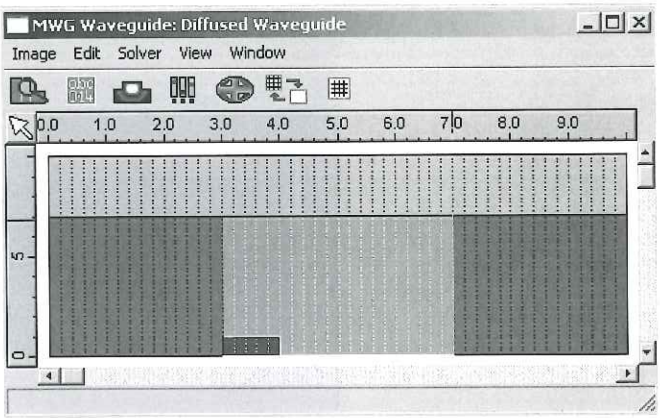
- Add another new rectangle (see below)



- Click in the center of the new rectangle and drag it so that the left side edge is at poxition x = 3.0um.

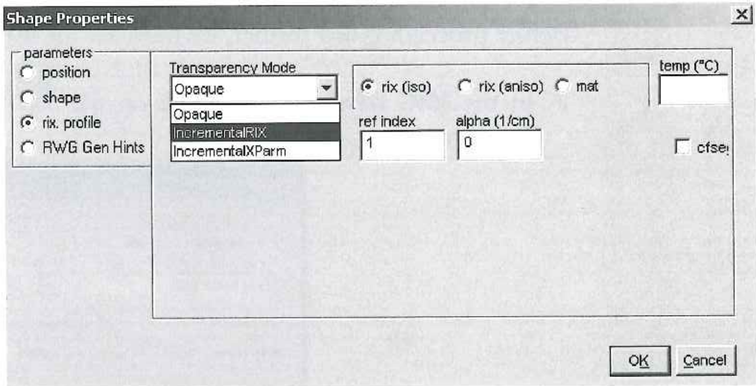


- Using the mouse, resize the new rectangle: Height = 7.0um, Width = 4.0um



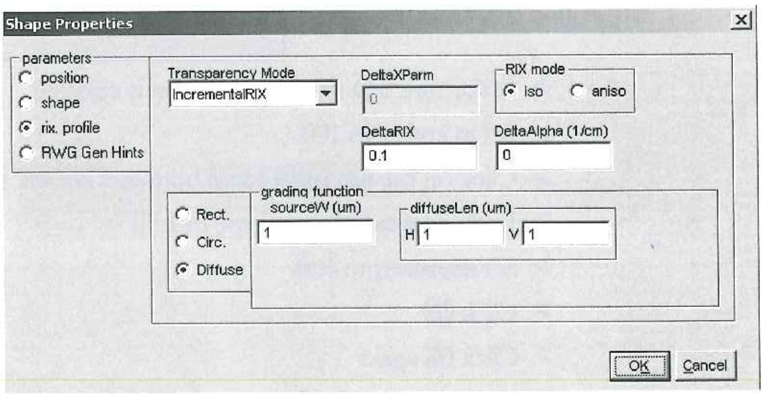
The refractive index of the new shape is set by default to *1.0*. We want to change this.

- Right-click on the shape and select **properties**.
- Set **Transparency Mode** to *IncrementalRIX*,

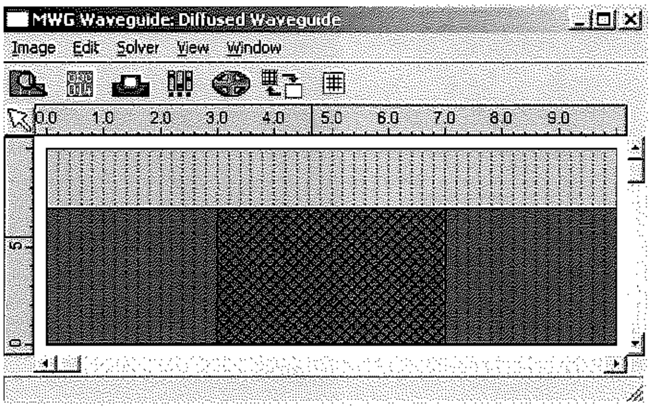


- **Set DeltaRIX** to *0.1*
- Set **grading function** to *Diffuse.*

The dialog box should look as shown.



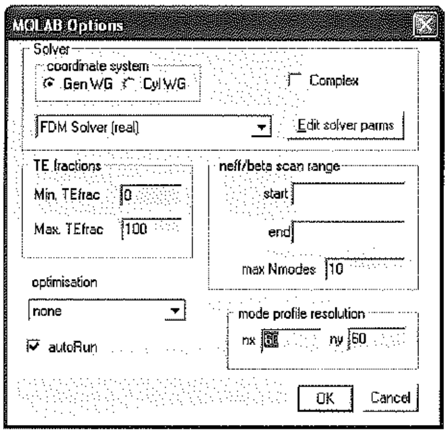
- Click **OK**. You should see the following.



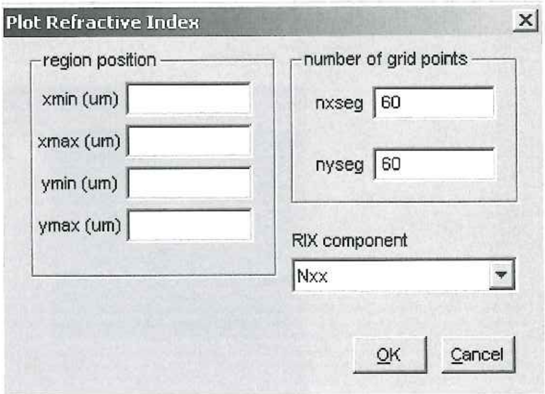
The second rectangle will now **not** have a constant refractive index, but will contribute to the background refractive index in the way specified in the dialog box. This contribution is an increment of *0.1* on the topside, tailing off to zero towards the other sides.

Before proceeding any further, we must set the wavelength.

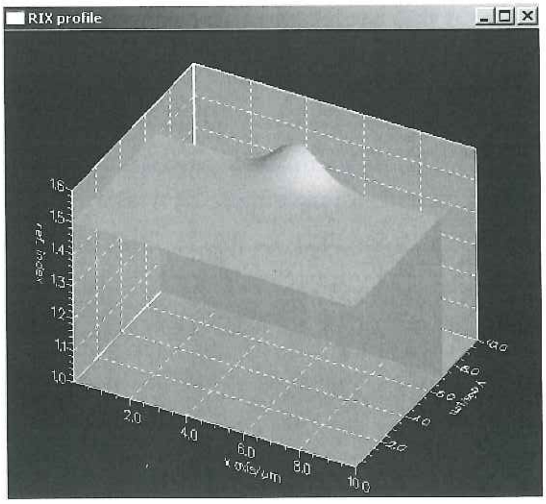
- In the *MWG Waveguide Editor*, click on the button. This will show the *MOLAB Options* panel.



- Make sure that the real *FDM Solver* is selected.
- Set **nx** and **ny** to 100.
- Click on the **Edit Solver parms** button. This will show the *FDM Solver parameters* panel.
- In the **lambda (um)** box, type in 1.55
- Set **HSymmetry** to *both*
- Click **OK**
- Click **OK** again
- You can now visualise exactly what you have done so far by viewing the refractive index profile.
- In the *MWG Waveguide Editor*, click on the button. This will show the *Plot Refractive Index* panel.



- Click **OK**
- The following plot should appear.

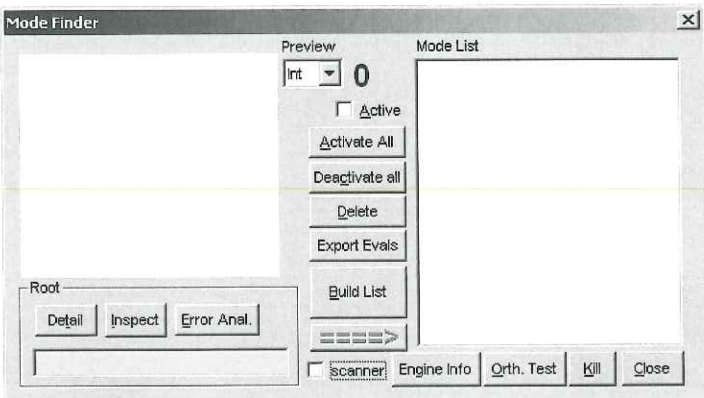


2.3.3 Finding the modes

You are now ready to find the modes of this waveguide.

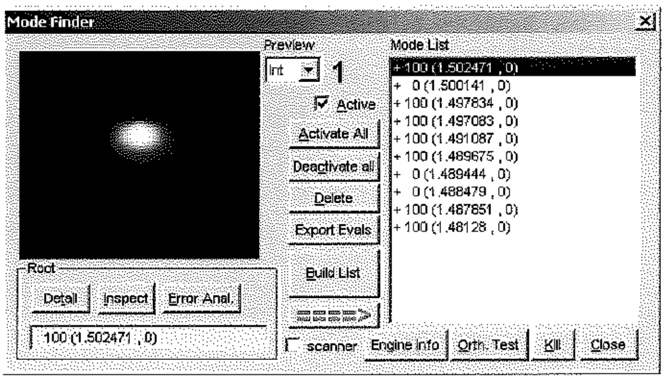
- In the *MWG Waveguide Editor*, click on the button.

The program will do some preliminary calculations and after a short delay, the *Mode Finder* panel will appear.



- Click on the **Build List** button,
- Click **Start**.

The fundamental TE and TM modes will appear in the Mode List along with some radiation/"box" modes.

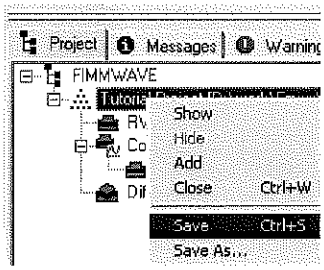


2.3.4 Saving the project

- Click **Close** on the *Mode Finder* panel
- Click **Yes** to save the Mode List.

The program will remember the Mode List when you close the program – ready for when you next open the waveguide.

- Right-click on the ∴ icon in the *Project Tree* and select **/Save**.



The project is now saved.

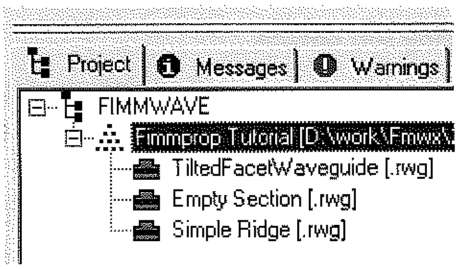
Chapter 3

Basic FIMMPROP Usage

We will construct all our components within a project provided with this document, so the first thing you must do is open it.

- Click on the icon on the tool bar.
- Navigate to the Tutorial Examples directory and select "`BasicFimmprop.prj`"
- Click **Open**.

The following should appear in the *Project Tree*.



You will contruct all your components within this project. For reference purposes, the file "Complete_BasicF`immprop.prj`" contains all the examples constructed in this tutorial section.

3.1 Example 1 - Reflections: Simulating a Tilted Facet

A major feature of FIMMPROP is its fully bi-directional capability. This is illustrated by the following simple example that looks at the reflections from a tilted facet.

3.1.1 Getting started

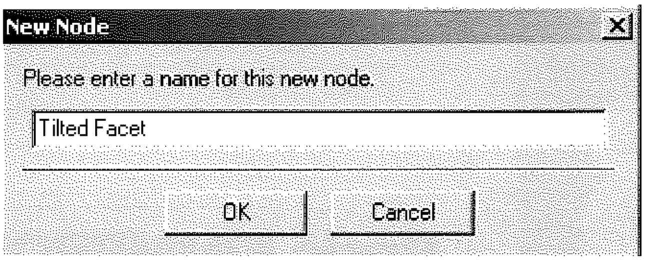
Most FIMMPROP devices will be created using the *node type* "*FIMMPROP Device*". So the first thing you must do is to create a node of this type.

- To create a new *FIMMPROP Device*, click on the button in the *new nodes toolbar*

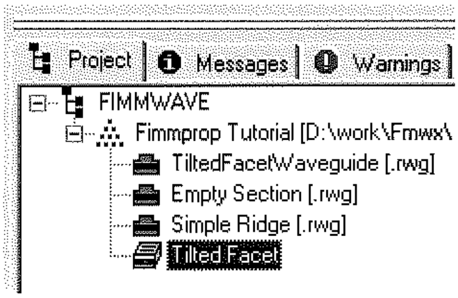
Note that this button will only appear if the Fimmprop Tutorial is selected in the *Project Tree*; if it is not shown, click once on the Fimmprop Tutorial

The *New Node dialog* will appear for you to enter the name of the new device:

- Type **Tilted Facet**, then click **OK**.



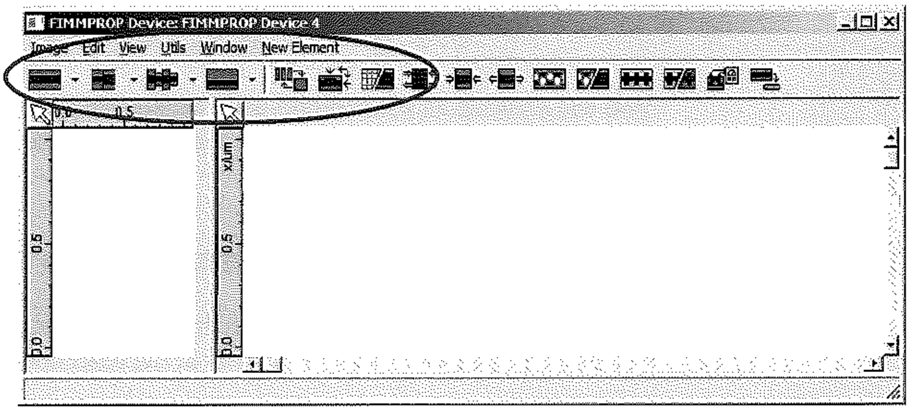
The *FIMMPROP Device* icon will appear in the *Project Tree* as shown



➢ Double click on this icon to open the *FIMMPROP Device* Editor.

3.1.2 Building the device

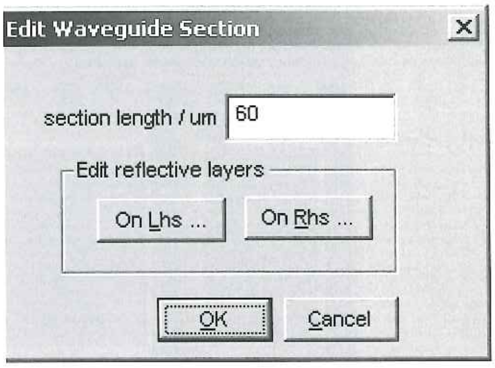
A structure (a *FIMMPROP Device)* is formed by joining together various types of simple *elements*. There are different types of elements, which you can insert by using the buttons on the toolbar (highlighted below).



The first set of icons on the toolbar contain the list of sections needing a waveguide cross-section. The set defaults to the *WG Section* (**W**ave**G**uide section) shown. We wish to create a *WG Section* and give it a length of 60um

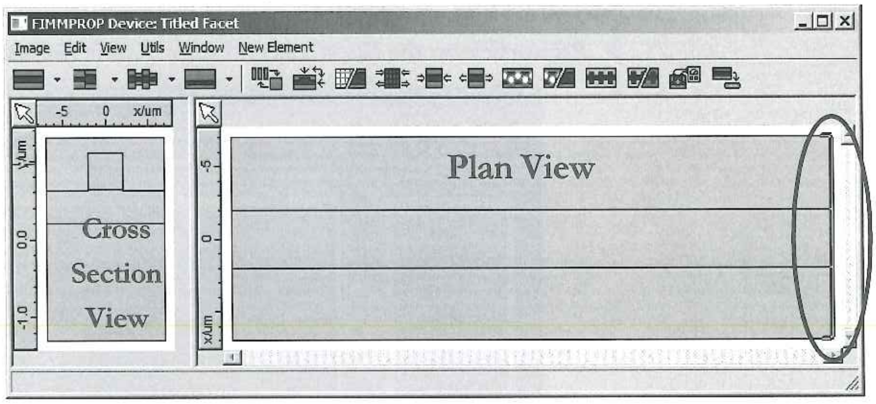
➢ Click on the icon. You should see the following.

- Highlight "TiltedFacetWaveguide"
- Click **OK**
- Type 60 in the **section length / um** box



- Click **OK**

You should see the following.



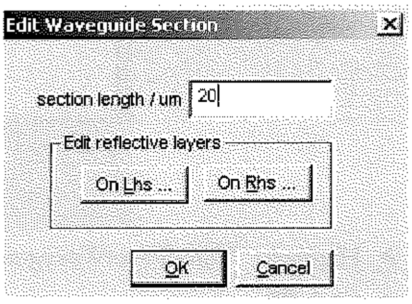
A *FIMMPROP Device* can be made up of many *Sections* but they have to be separated by *Joints*. Therefore we now need to append a *Joint*.

- Click to the right of the plan view of the WG Section so that the cursor is placed at the right hand side (highlighted above).
- Click on the icon.
- Clik **OK**

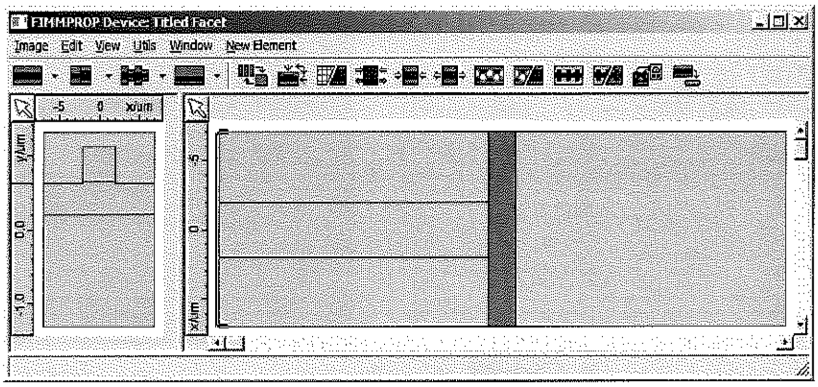
A new blue element will appear after the *WG Section*. Note that the *Joints* have a symbolic length, since in reality they are zero.

Now we wish to append another waveguide and give it a length of 20um.

- Click on the icon.
- Highlight "Empty Section"
- Click **OK**
- Type 20 in the **section length / um** box

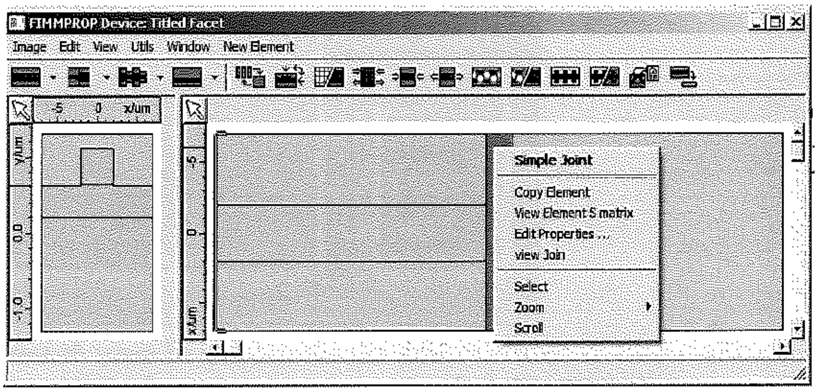


You should end up with a *Device* with two *Sections*: the LHS *Section* has a guiding core with a relatively high refractive index, while the RHS one is just air (an empty *Section*).

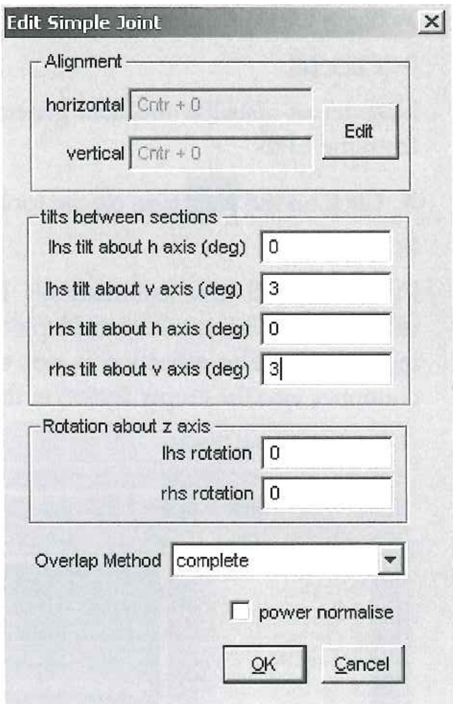


Joints contain all the information needed to define the interface between consecutive *Sections*, such as the alignment, titling, or rotation. Now let us introduce a tilt in the join:

- Right-click on the *Simple Joint* and select **/Edit Properties...**



- In the **lhs tilt about v xis box** type 3.
- In the **rhs tilt about v xis box** type 3.



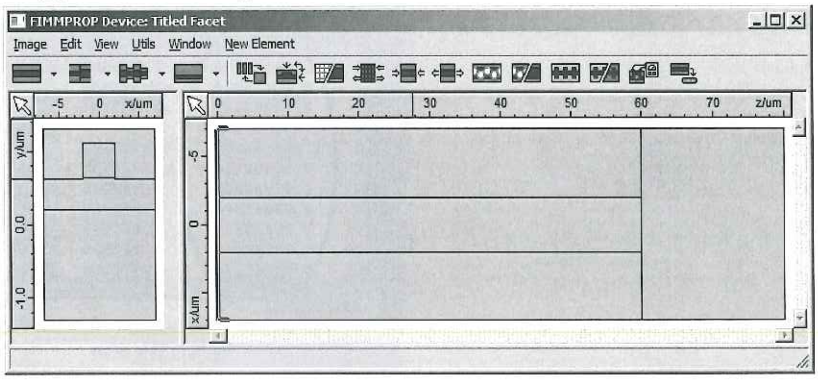
➢ Click **OK**

The *Joint* now has a total tilt of 6 degrees, although you cannot see this in the *FIMMPROP Device* Editor.

Also notice that the graphical lengths of the *Sections* are the same even though we have set one to be 60um and the other to be 20um. This is because you are in *symbolic view* mode which is generally more convenient for editing. In order to see the physical size of the *Device* do the following.

➢ Click on icon on the toolbar.

You should now see the *WG Sections* scaled to their respective lengths. The horizontal ruler also appears above the plan view.



Note that the joint is now invisible since it has zero length. Hence in general if you want to access the parameters of an element with zero or comparatively small length, you have to revert back to *symbolic view* to be able to select it.

3.1.3 Performing the simulation

We now need to set the wavelength.

➢ From the *FIMMPROP Device* editor select **/Edit/options** from the menu.

➢ Type 1.103 in the **Lambda/um** box

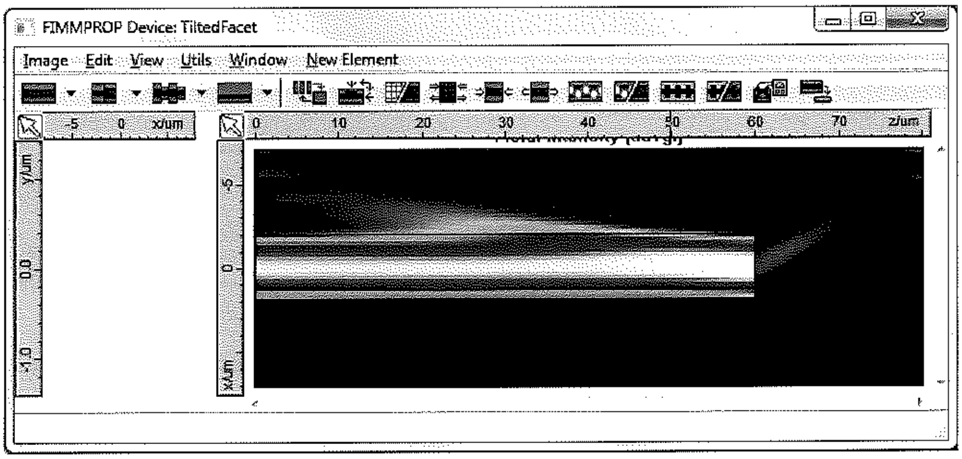
➢ Click **OK**.

Now let us visualise the field generated by injecting the fundamental mode injected from the LHS.

➢ Click on the icon on the toolbar

➢ Click **OK**.

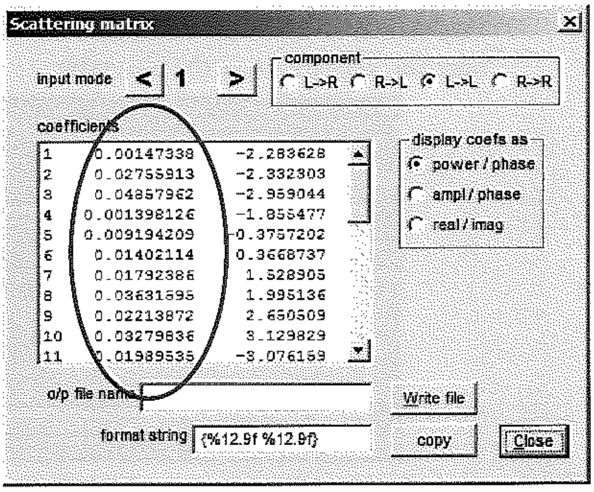
FIMMPROP will now automatically find the necessary modes and do the necessary calculations. The resulting field plot shown here clearly shows the reflection of the input mode. The reflection is not total, since you can see that some of the power continues into the empty *Section* on the right.



The amount of power that is reflected can be found by inspecting the *Scattering Matrix*.

➢ Click on icon on the toolbar.

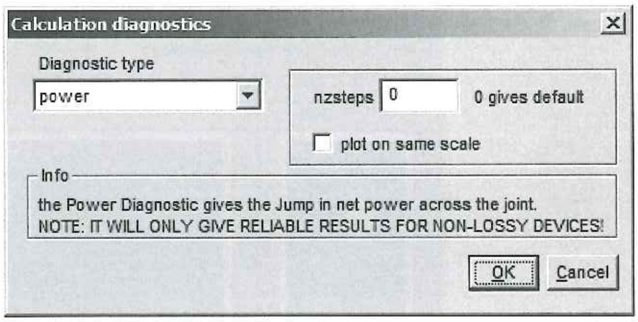
➢ Click on **L->L** (Reflection from left to left)



If you add up all the coefficients, you will find that the total power reflected is about 25%.

➢ Click **Close**

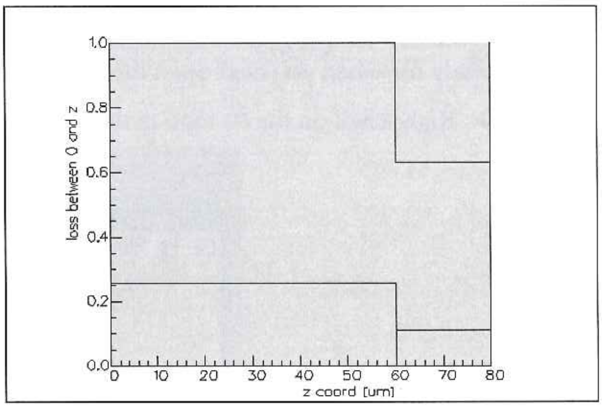
➢ Click on the icon on the toolbar. This brings up the Calculation Diagnostics.



➢ Enable (tick) **plot on same scale**.

➢ Click **OK**

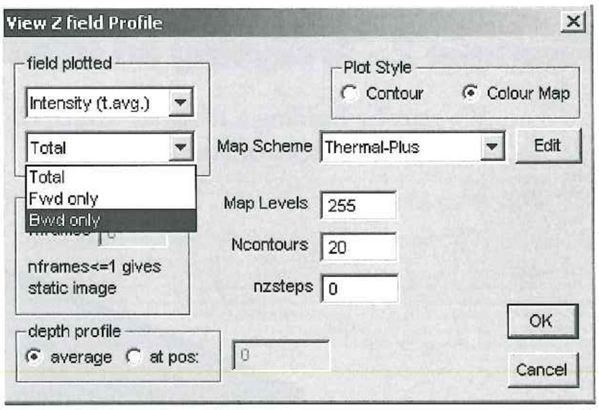
You should see the following graph. The red line is the power propagating from right to left i.e. the reflection from the facet. Notice that this is approx. 25%. This agrees with the *Scattering Matrix*.



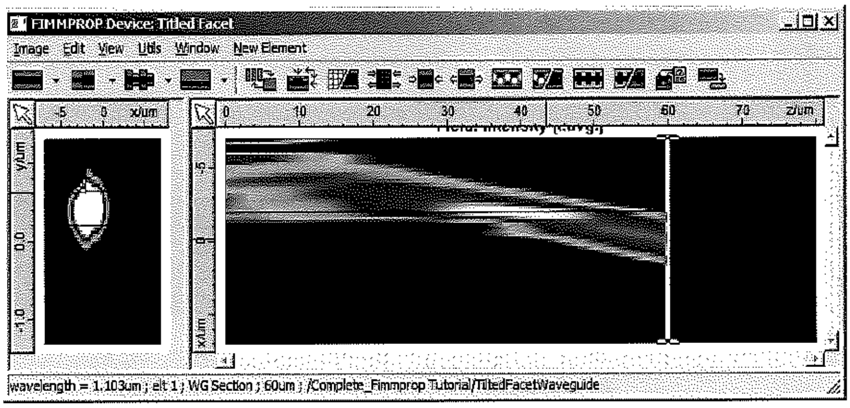
You can visualise the reflection more clearly by plotting only the backward field

➢ Click on the icon on the toolbar

➢ In the **field plotted** box select **Bwd only**.



➢ Click **OK**. You should see:

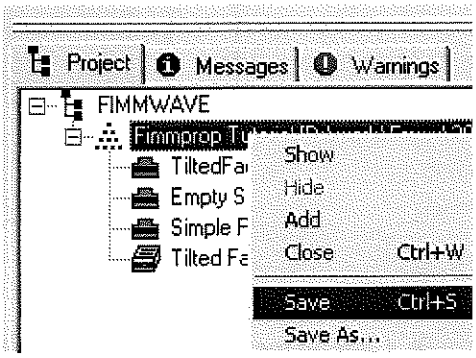


3.1.4 Saving the project

- Close the *FIMMPROP Device* Edit

Note that the program will remember the Mode List(s) when you close the program – ready for when you next open the project.

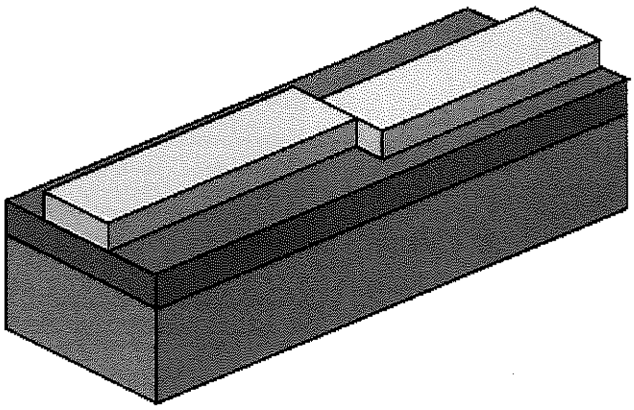
- Right-click on the icon in the *Project Tree* and select **/Save**.



The project is now saved.

3.2 Example 2 - Analysing an offset join

We start by building a relatively simple component made of two identical waveguides joined with a lateral offset.

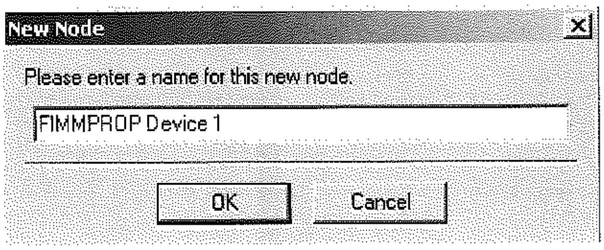


3.2.1 Getting started

- To create a new *FIMMPROP Device*, click on the button in the *new nodes toolbar*

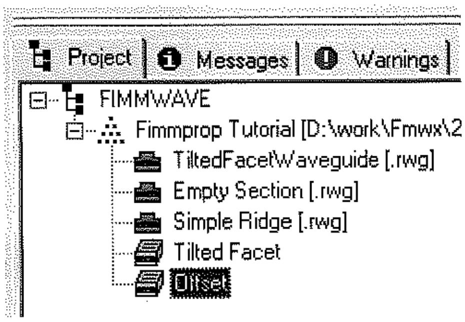
Note that this button will only appear if the Fimmprop Tutorial is selected in the *Project Tree*; if it is not shown, click once on the Fimmprop Tutorial

Again, the *New Node dialog* will appear for you to enter the name of the new device:

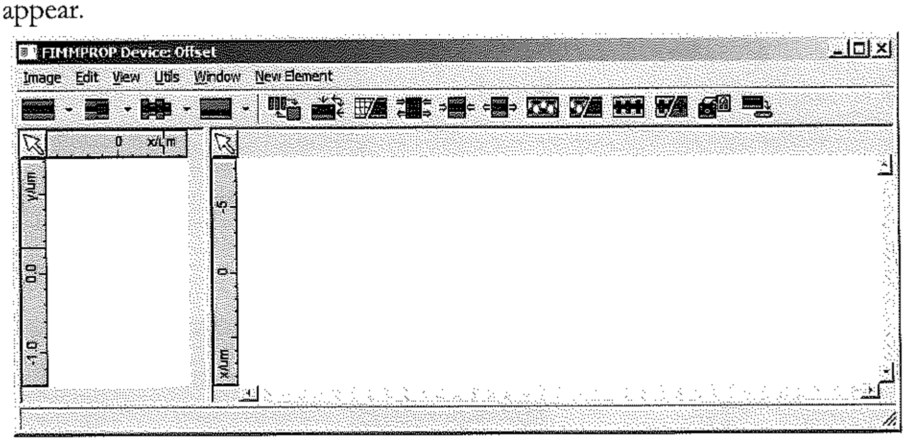


- Type **Offset**, then click **OK**.

The *FIMMPROP Device* icon will appear in the *Project Tree* as shown.



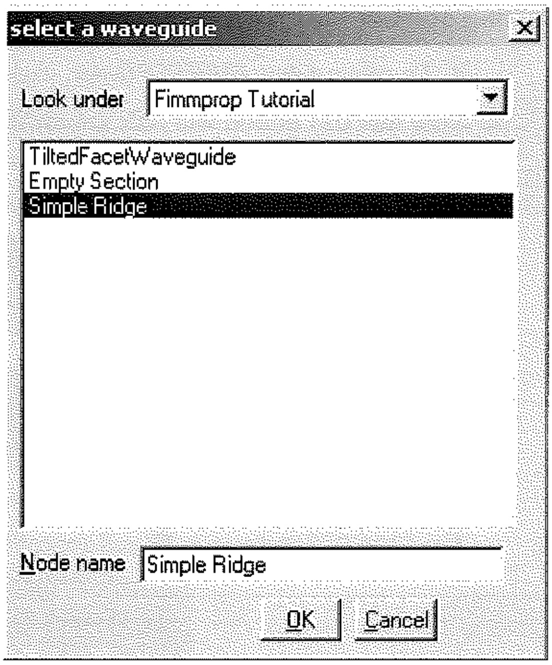
- Double click on this icon to open the *FIMMPROP Device* editor. The following should appear.



3.2.2 Building the device

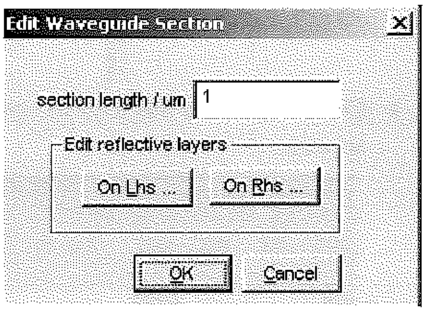
We wish to create a *WG Section* and give it a length of 10um

- Click on the icon. You should see the following.



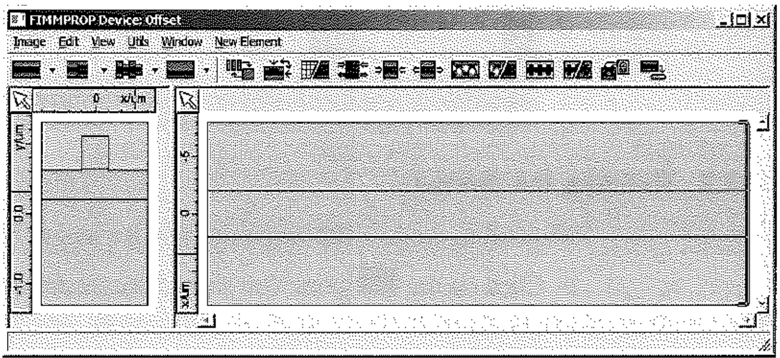
- Highlight "Simple Ridge"
- Click **OK**

You should see the following



- Type 10 in the **section length / um** box
- Click **OK**

You should now see the following.



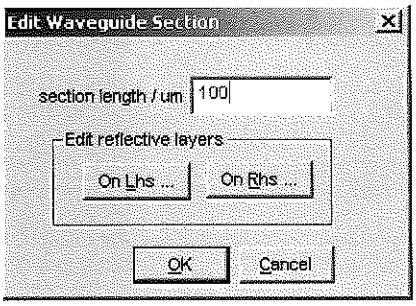
Now we wish to add a new identical *WG Section.*

- Click to the right of the plan view of the WG Section so that the cursor is placed at the right hand side.
- Click on the  icon again.

➢ Highlight "Simple Ridge"

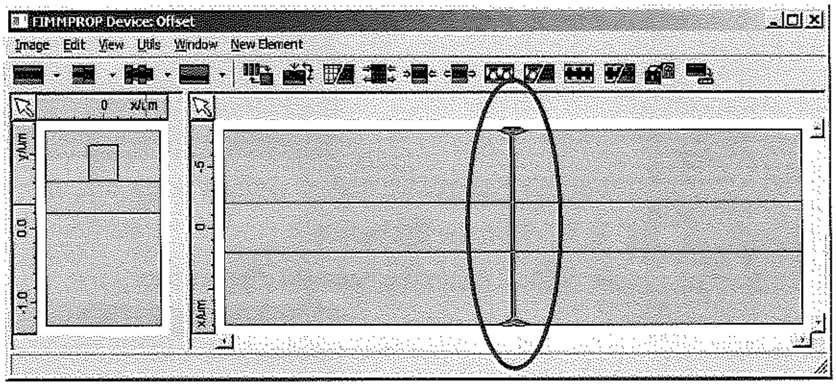
➢ Click **OK**

➢ Type 100 in the **section length / um** box



➢ Click **OK**

You should now see two consecutive identical *WG Sections.*

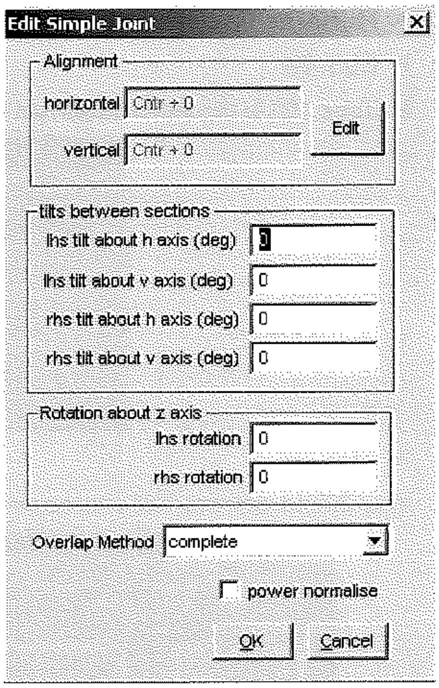


Note the vertical line at the interface with small green triangles on the top and bottom (highlighted above). This is an Error Symbol. This reminds us that we need to insert a *joint.*

➢ Click on the **Error Symbol.**

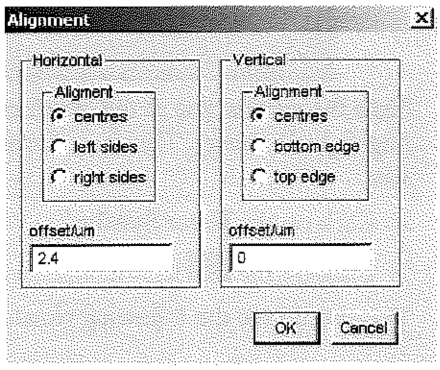
➢ Click on the  icon to insert a *Joint.*

You should see the following panel.



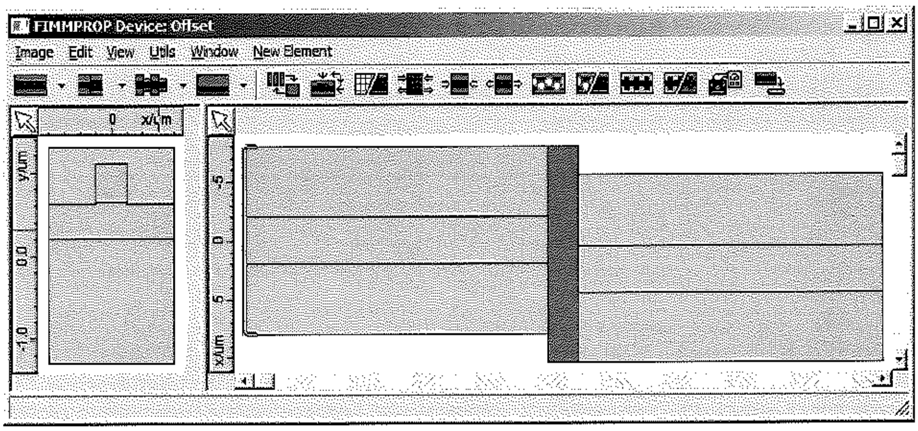
For this example we wish to define a horizontal offset between the two waveuides.

- Click on the **Edit** button.
- Type 2.4 into the **Horizontal offset/um** box



- Click **OK**
- Click **OK** again

You should see the following:



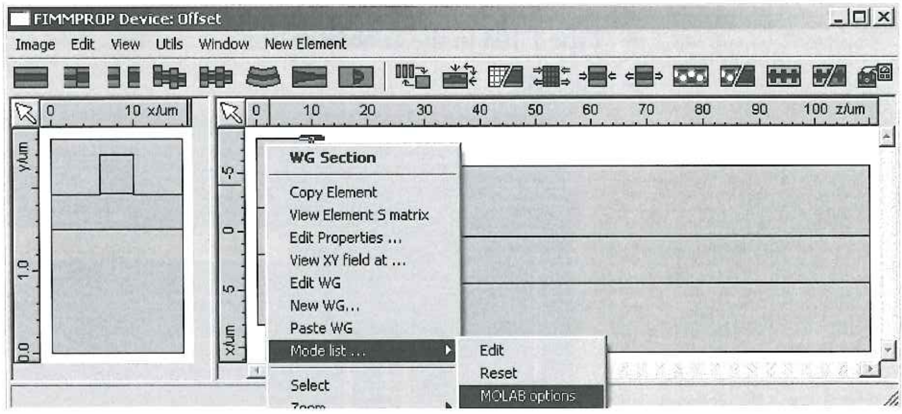
The offset should also show in the *FIMMPROP Device* window. Note that you are viewing the structure from the top, so the positive x direction is downwards; the offset should reflect this.

3.2.3 Performing the simulation

For each *WG Section* in the *FIMMPROP Device* a set of modes needs to be calculated. FIMMPROP can do this automatically, we need to set the *MOLAB* (Mode List Auto Builder) Options:

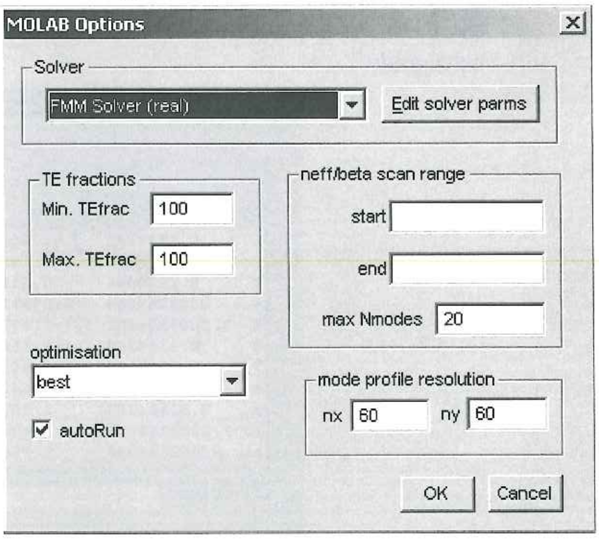
- Click [icon] to show the device in ***real view***.
- Right-click on the first *WG Section* and select **/Mode list/MOLAB options**.

This will show the *MOLAB Options* panel.



- Type 20 in the **max Nmodes** box. When we run the simulation the solver will then attempt to find the first 20 eigenmodes for the waveguide.
- Type 100 in the **Min. Tefrac** box. When we run the simulation we will only consider 100% Ex polarised modes (I.e. TE modes only).
- Make sure that **autoRun** is ticked;
- Make sure the **solver** box is set to *FMM Solver* (real). This will be the solver used to automatically find the modes.

You should see the following



- Now click on **Edit solver parms** to display the FMM Solver Parameters panel.
- Set Type to **Semivec TE**.

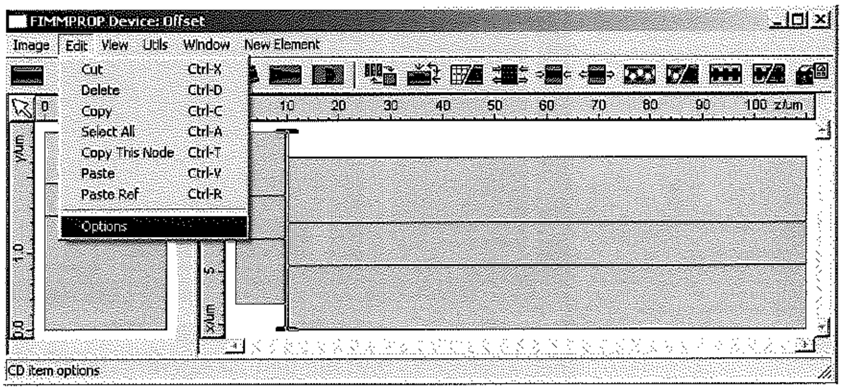
This tells the solver to use a semi-vectorial method that is an adequate approximation for the present example. Press on OK.

- Click **OK**
- Click **OK again**

The *MOLAB* is now configured to find 20 TE eigenmodes, with the real *FMM Solver* using a semi-vectorial TE approximation. Note that we do not have to repeat this procedure for the second waveguide section as it references the same waveguide as the first.

We now need to set the wavelength.

- From the *FIMMPROP Device* Editor select **/Edit/Options** from the menu.
- Type 1.103 in the **Lambda/um** box
- Click **OK**.

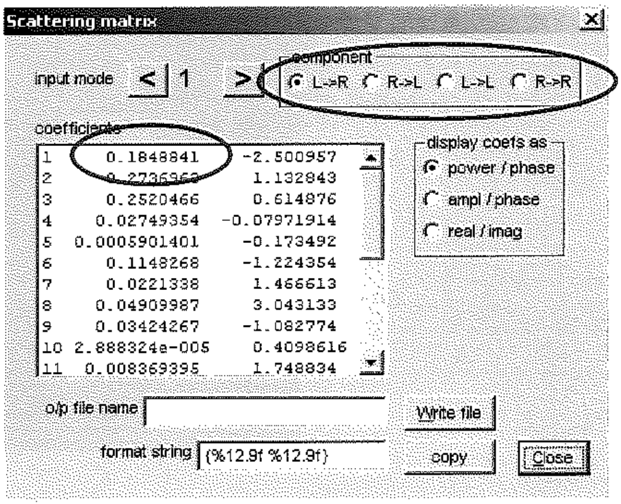


We are now ready to do some calculations.

- To calculate the *Scattering Matrix*, click on icon on the toolbar.

FIMMPROP will then run the MOLAB to find the 20 modes for the WG cross-section *"Simple Ridge"*

Then, after a few seconds, the *Scattering Matrix* will appear showing the **L->R** (Transmission from left to right) coefficients of the excited modes of the output on the RHS. Notice about 18.5% of the power remains in the fundamental mode of the waveguide.



If you wish, you can also inspect the other coefficients by selecting:

- **R->L** (Transmission from right to left)
- **L->L** (Reflection from left to left)
- **R->R** (Reflection from right to right).

(Highlighted above)

➢ Click **Close**

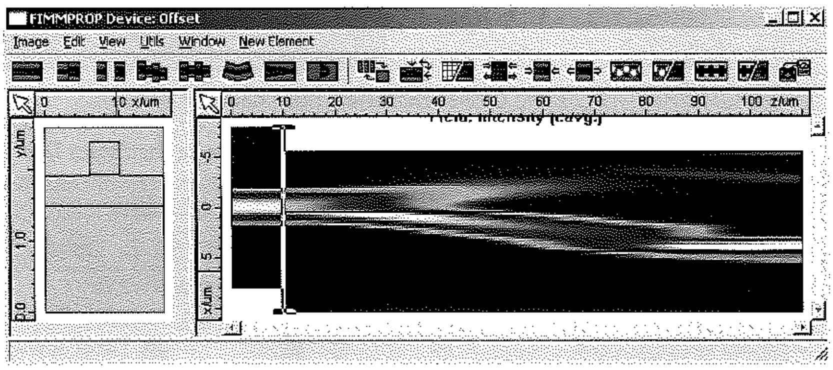
Now let us visualise the field generated by injecting the fundamental mode injected from the LHS.

➢ Click on the icon on the toolbar

➢ Make sure that **Total** is selected in the **field plotted** box.

➢ Click **OK**.

After a few seconds a graphical representation of the propagated field with the LHS fundamental mode as input will appear on the graph. You can clearly see much of the injected light being lost in radiation.



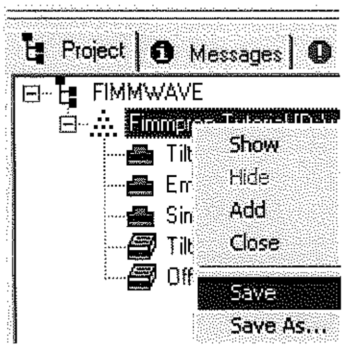
You can also get a cross-sectional view of the field at any z position by clicking on the longitudinal view at the desired point. The field cross-section at that point will appear in the pane on the left.

3.2.4 Saving the project

➢ Close the *FIMMPROP Device* Editor

Note that the program will remember the Mode List(s) when you close the program – ready for when you next open the project.

➢ Right-click on the icon in the *Project Tree* and select **/Save**.



The project is now saved.

3.3 Example 3 – Using the FIMMPROP Scanner

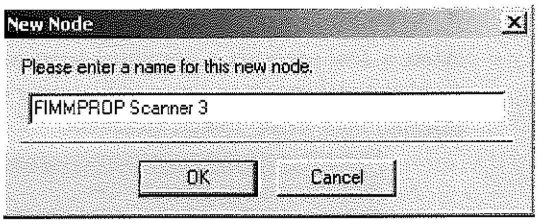
This quick example demonstrates how to use the FIMMPROP Scanner to study the transmission through the structure set up in the previous section as a function of the horizontal offset.

3.3.1 Getting Started

➢ Click on the button in the *new nodes toolbar*

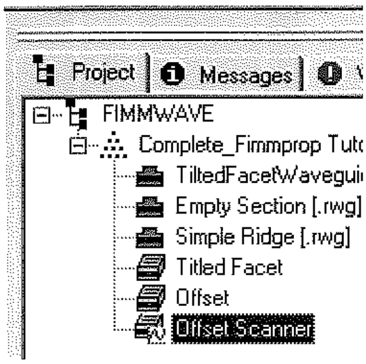
Note that this button will only appear if the Tutorial Project is selected in the *Project Tree*; if it is not shown, click once on the Tutorial Project.

Again, the *New Node dialog* will appear for you to enter the name of the new device:

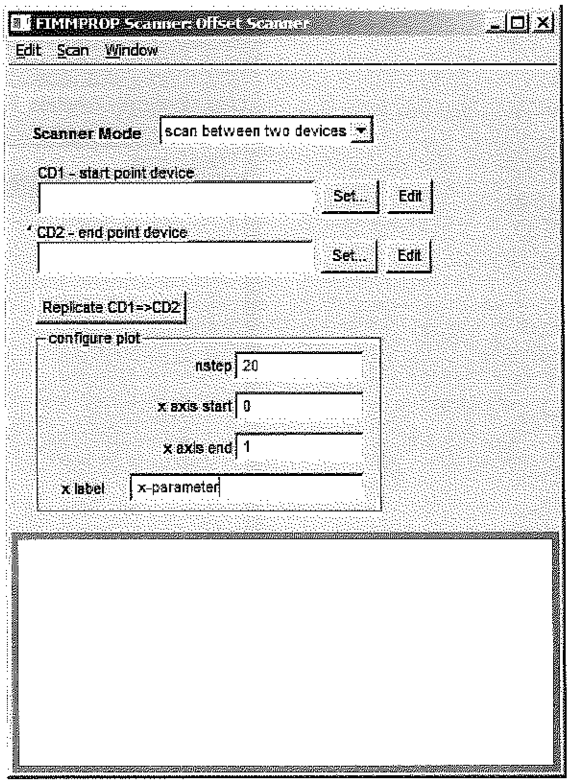


➢ Type **Offset Scanner**, then click **OK**.

The waveguide icon will appear in the *Project Tree* as shown below.

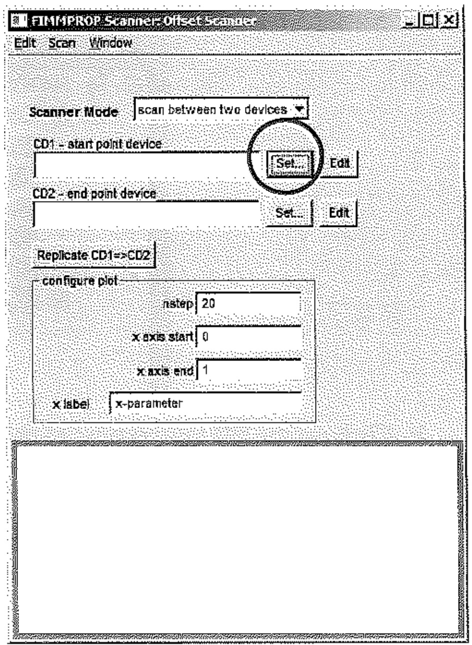


➢ Double click on this waveguide icon to open the *WG Scanner*. The following should appear.

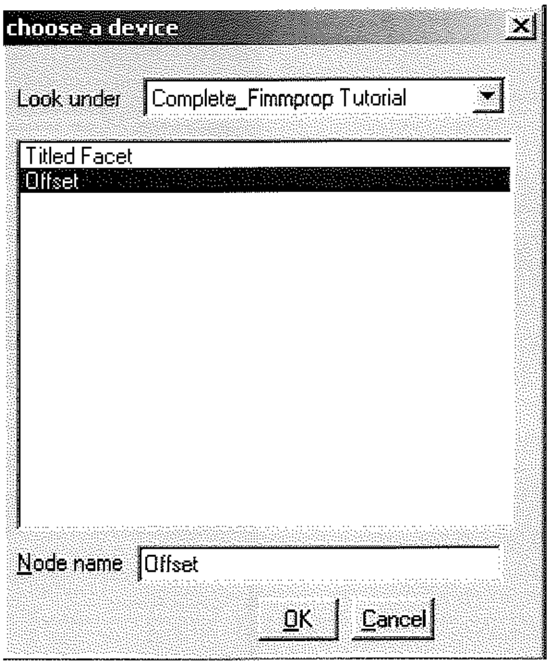


3.3.2 Setting up the FIMMPROP Scanner

➢ Click on the **Set** button next to the **start point device** box.



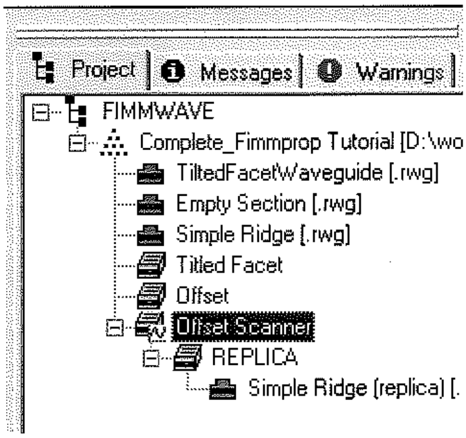
This will bring up the Choose a Device panel.



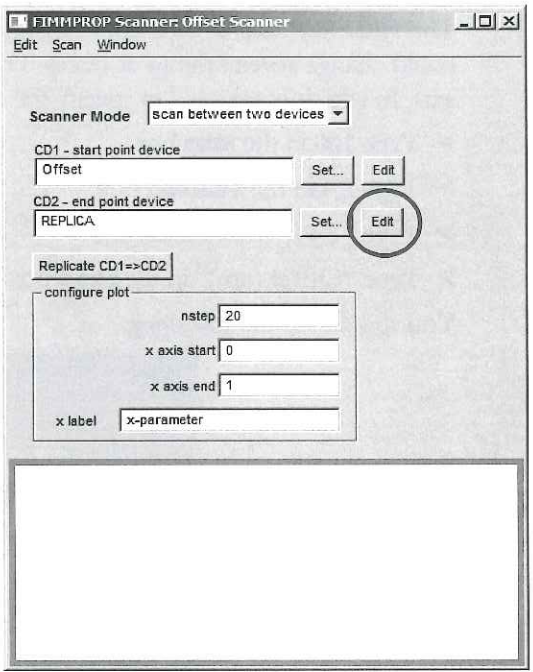
- Highlight **Offset** – the waveguide we set up in the first example.
- Click **OK**

You should now see "Offset" in the **start point device** box

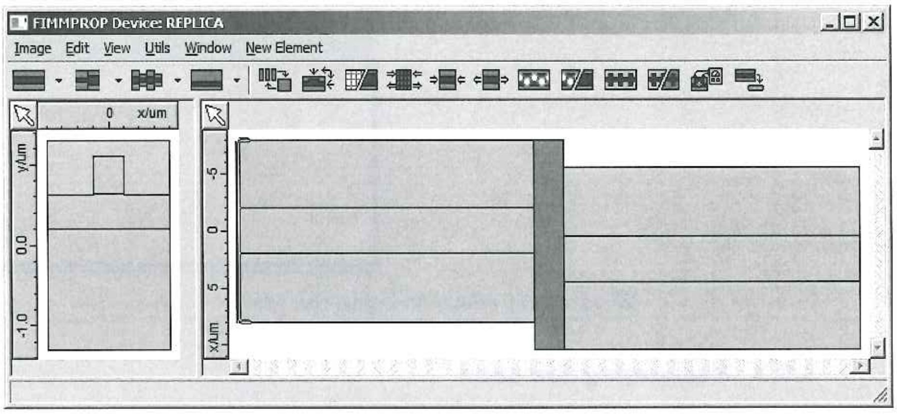
- Click on the **replicate CD1=>CD2** button. This will make a copy of the waveguide. You should see this copy appear in the *Project Tree*.



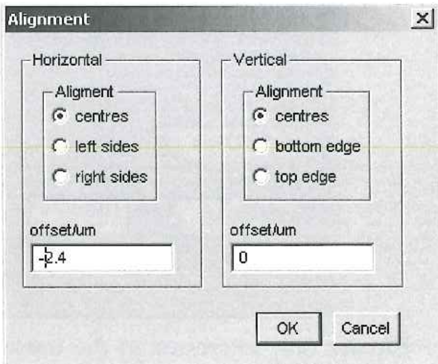
- Click on the **Edit** button next to the **end point device** box.



This will show the *Device Editor* for the copied device. It is currently the same as the original.



- Right-click on the *Simple Joint* and select **/edit properties...**
- Click on the **Edit** button.
- Type -2.4 into the **Horizontal offset/um** box

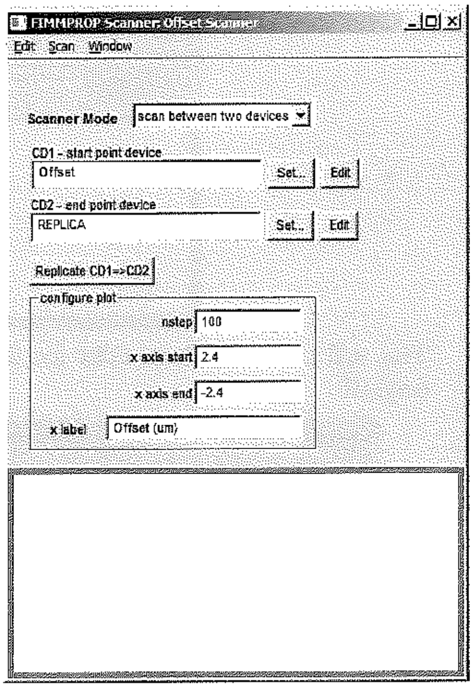


- Click **OK**
- Click **OK** again

The *FIMMPROP Scanner* tool does not know which parameters are being scanned (you could change several things at once). Therefore we need to define the labels for the x-axis. In addition, we need to specify the number of points we wish to calculate.

- Type 100 in the **nstep** box
- Type 2.4 in the **x-axis start** box
- Type –2.4 in the **x-axis end** box
- Type "Offset (um)" in the **x-label** box

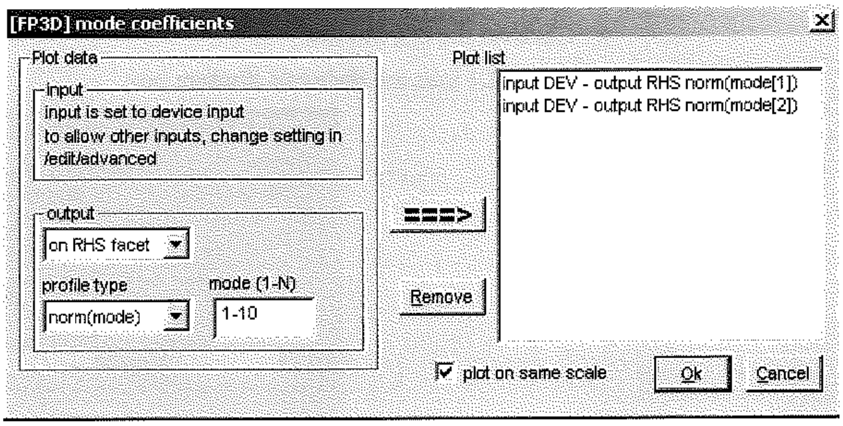
You should see the following.



We are now ready to begin the scan.

3.3.3 Performing the scan

- On the *FIMMPROP Scanner* select **/Scan/Start**. You should see the following.



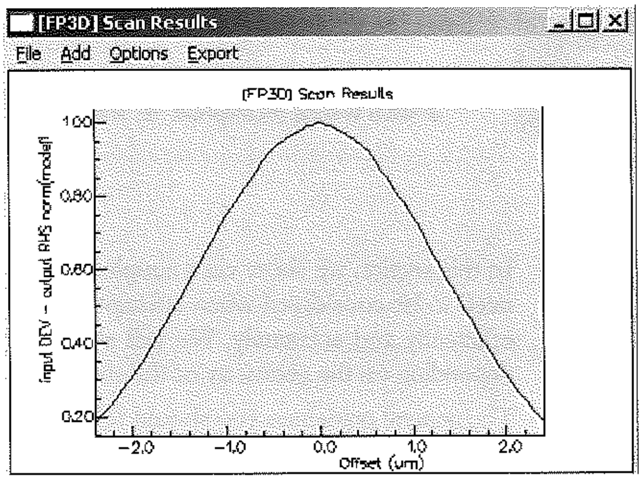
We are only interested in the transmission for the fundamental mode. Therefore we need to remove "input DEV – output RHS norm(mode[2])" from the **Plot list.**

- Click on "input DEV – output RHS norm(mode[2])" to highlight it.
- Click on the **Remove** button.

➢ Click **OK**.

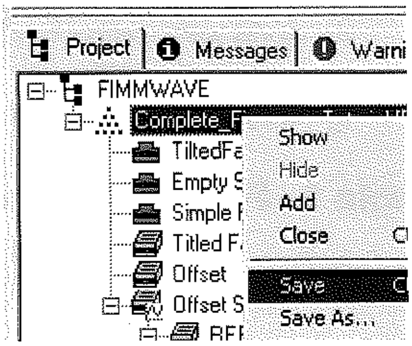
➢ After a minute or two the scan will have completed, click **OK**.

You will now see a plot the transmission as a function of the offset. As expected the transmission peaks at unity when the offset is zero.



3.3.4 Saving the project

➢ Right-click on the ∴ icon in the *Project Tree* and select **/Save**.



The project is now saved.

Chapter 4

Advanced FIMMWAVE usage

In this section we show via a couple of examples, some of the more subtle points that need to be considered when attempting to obtain results using the FDM mode solver efficiently and accurately. We will also look at the option of using a different solver depending on the geometry of the structure.

The general guidelines you should follow when using the FDM solver are:

- Minimise the size of the computational window.
- Exploit any symmetries of your structure.
- After finding modes, check their orthogonality.
- If the latter is not good enough, increase the number of 1d modes used, and find the modes again – either by running the *MOLAB* from scratch, or "polishing" your mode list (much faster!).

The following example will illustrate the above points.

4.1 Example 1 - Finding the modes in an elliptical Ring

4.1.1 Simulating the full ring

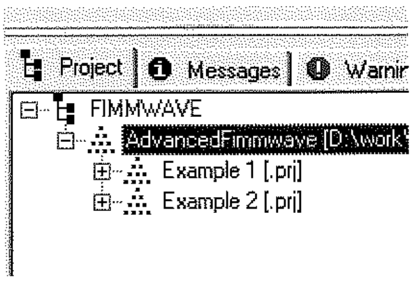
We will construct all our components within a project provided with this document, so the first thing you must do is open it.

➢ Click on the icon on the tool bar.

➢ Navigate to the Tutorial Examples directory and select "`AdvancedFimmwave.prj`"

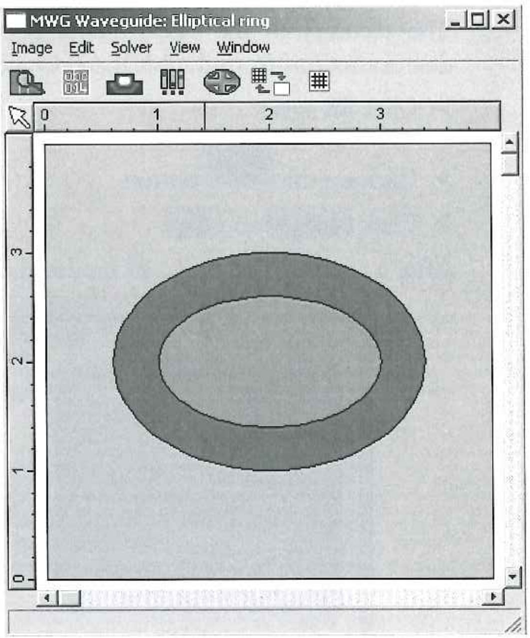
➢ Click **Open**.

The following should appear in the *Project Tree.*

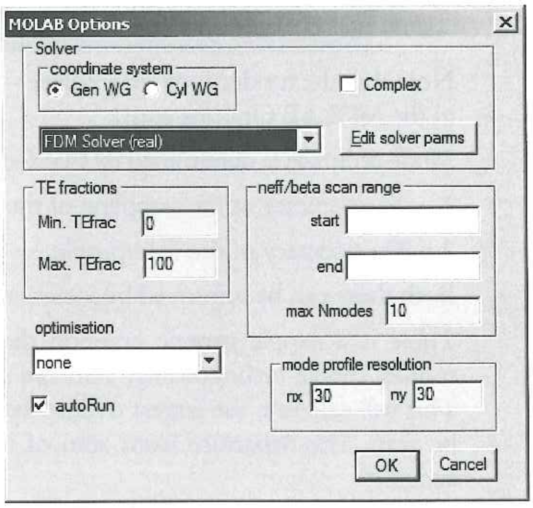


You will contruct all your components within this project.

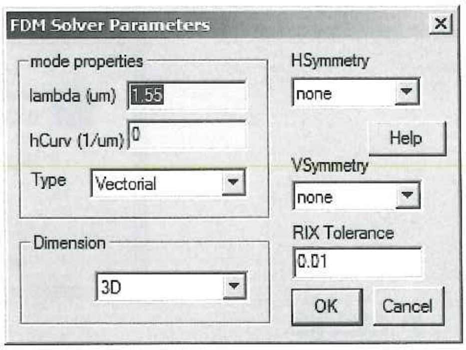
➢ Open the waveguide *"Example1\Elliptical ring."* You should see the following.



➢ Click on the  button. This will show the *MOLAB Options* panel.



➢ Click on the **Edit Solver parms** button. This will show the *FDM Solver parameters* panel. You should see the following.



Note "Dimension" is defaulted to "3D" - this means 2 lateral dimensions plus exp(i.β.z) z-dependence – used for 3D FIMMPROP simulations.

➢ Click **OK**

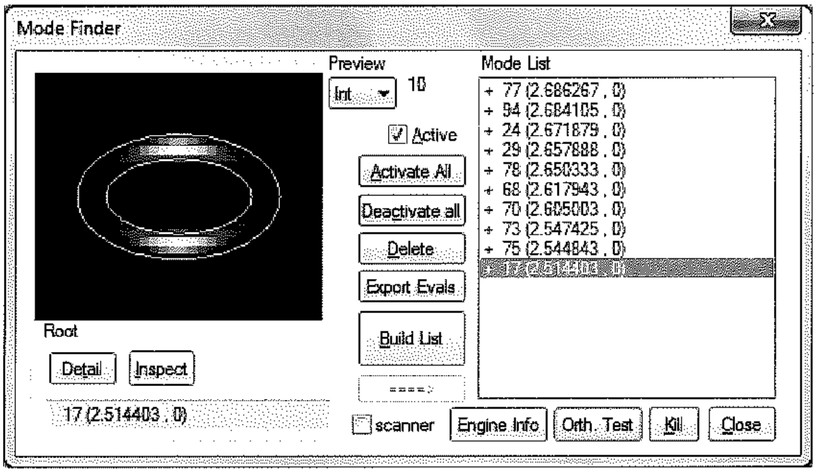
The accuracy of the FDM Solver is controlled by the grid size. The FDM Solver determines this from the MOLAB Options nx, ny parameters in this panel.

➢ Click **OK** again.

➢ Click on the  button.

➢ Click **Build List** and **Start**.

After a couple of seconds, 10 modes should appear in the mode list.



Note that the modes look very coarse – this is due to the coarse nx, ny parameters set in the MOLAB Options panel.

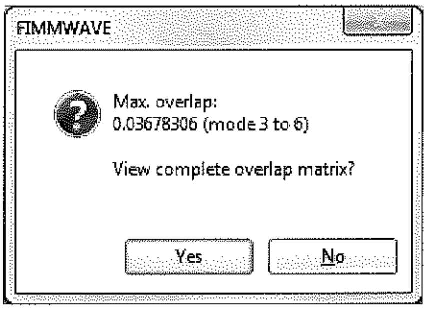
Mode accuracy is determined by two factors:

1. The accuracy of the sampling of the underlying structure.
2. The accuracy of the field profile.

Both these can be improved by increasing nx, ny.

There is a simple generic criterion that can be used to check the accuracy of the modes: modal orthogonality. You can do this in FIMMWAVE via the **Orth.Test** button. This will calculate the largest overlap between the different modes. Ideally this should be zero. The departure from zero of this value gives the user an idea of the mode accuracy.

➢ Click the **Orth. Test.** Button and say **No** to special complete overlaps. You should see the following.



➢ Click **No.**

A number rather larger than zero has been returned. This suggests that the modes are not very accurate.

We could simply increase nx, ny to increase the accuracy of the computation, however we need to be careful – the calculation time will increase dramatically if this is too big. Memory use will also increase.

➢ Close the Mode Finder window (no need to save modes).

4.1.2 Using symmetries

In structures like this one, we can take advantage of the vertical and horizontal symmetry planes to ease the computation.

When taking advantage of the symmetry plane in the vertical direction, the calculations are equivalent to doubling the resolution. Thus the computation will be more than 2x times faster.

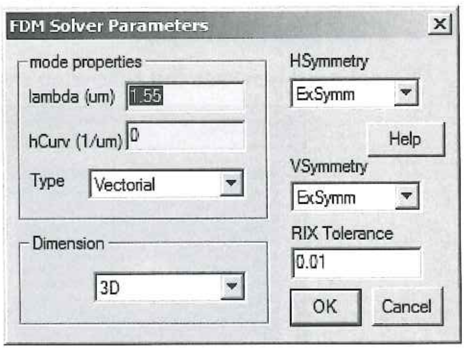
Using the symmetry plane in the horizontal direction reduces the calculation time by a factor of >two.

➢ Click on the button. This will again show the *MOLAB Options* panel.

➢ Click on the **Edit Solver parms** button. This will show the *FDM Solver parameters* panel.

➢ Set Hsymmetry = ExSymm

➢ Set Vsymmetry = ExSymm.



➢ Click **OK**

➢ Now set nx=60, ny=60 in the MOLAB Options panel. With the symmetry planes we will actually only use a 30x30 grid internally – computing just one quarter of the structure. So computation time will be the same as before but with better resolution.

➢ Click **OK**

Having chosen to take advantage of both the horizontal and vertical symmetries of the structure, we can double our grid resolution for the same computational effort.

➢ Click on the button.

➢ Click **Build List** and **start**

➢ Click **Yes** if you are asked if you want to delete all modes.

➢ When the 10 modes are found, you should see modes with much better resolution than before.

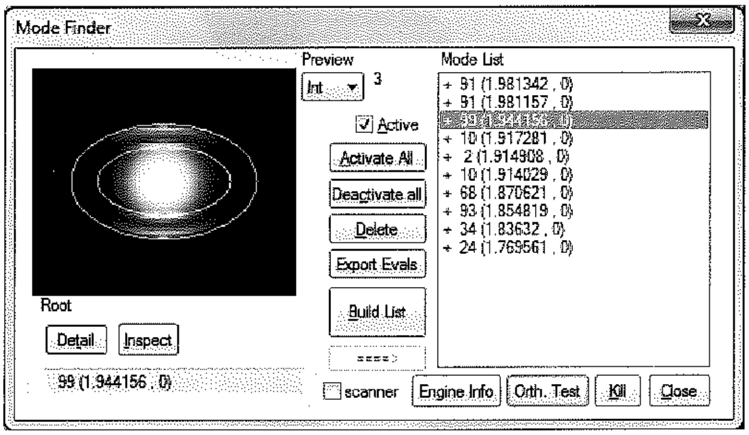
However note that only a subset of these modes will be calculated by the MOLAB, depending on the boundary conditions that have been set at these symmetry planes. Here we choose to find the modes that are Ex symmetric in both planes. In this case we choose to still find 10 modes, but now the modes go to higher order. Alternatively we could ask it to find fewer modes, speeding up the simulation time.

4.1.3 Finding modes in a specific effective index range

The centre of this structure has a refractive index of 2. What if we only wanted to find the mode that is concentrated in this central ellipse. We could ask the solver to find lots of modes until it reaches the mode of interest. FIMMWAVE has a better way.

- Close the project via the menu in the main program window named "/File/Close AdvancedFimmwave. Say NO if it asks you if you want to save.
- Re-open the project and re-display the "**elliptical ring [mwg]**" window.
- Click on the  button. This will again show the *MOLAB Options* panel.
- Set **neff/beta scan range start** to 2.0
- Set **neff/beta scan range end** to 1.5
- Set nx = ny = 60
- Click OK to close.
- Click on the  button to open the Mode Finder.
- Click **Build List** and **start**

You should now see the following mode list:



The modes all have effective indices between 2.0 and 1.5 as we requested in the MOLAB options.

4.1.4 Changing mode solver to suit the problem

FIMMWAVE gives the user the option to use any of a variety of solvers. The geometry of the structure that you wish to model should be a factor in choosing the mode solver.

For structures that are not efficiently described by layers and slices, it is perhaps better to use the FEM (finite element) Mode Solver. This solver uses a fine triangular grid, thus making it ideal for the following sorts of waveguides:

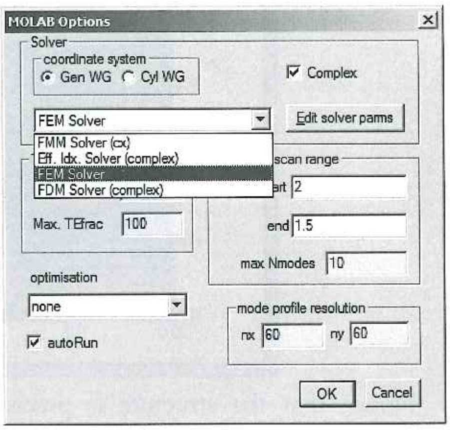
- Holey Fibres
- Elliptical Fibres
- Odd-Shaped waveguide with curved and/or slanting interfaces

- Diffused waveguides and other structures with smoothly varying refractive index

> Click on the button. This will show the *MOLAB Options* panel.

> Click the **Complex** box.

> In the Solver box select FEM. Solver.

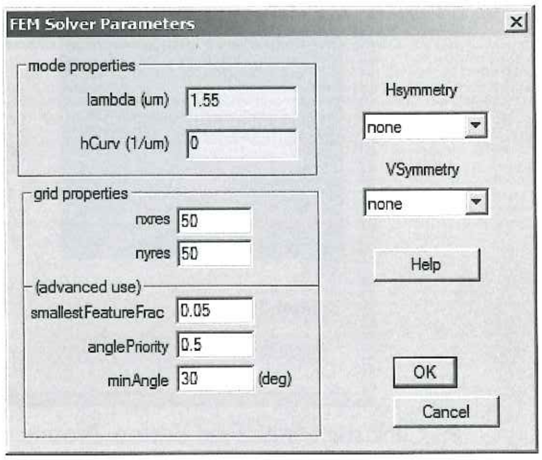


> Delete the values in the **start** and **end** boxes.

> Click on the **Edit Solver params** button.

> Type 50 in the **nxres** and **nyres** boxes.

You should see the following:

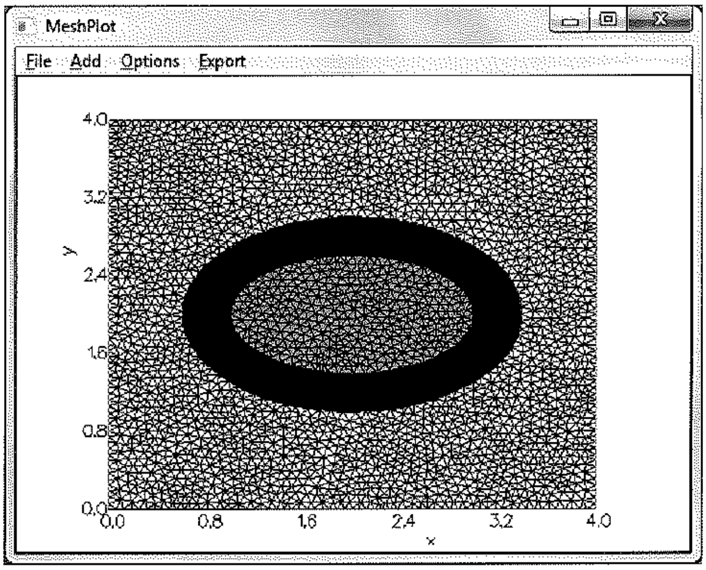


> Click **OK** and **OK** again.

> Click on the button.

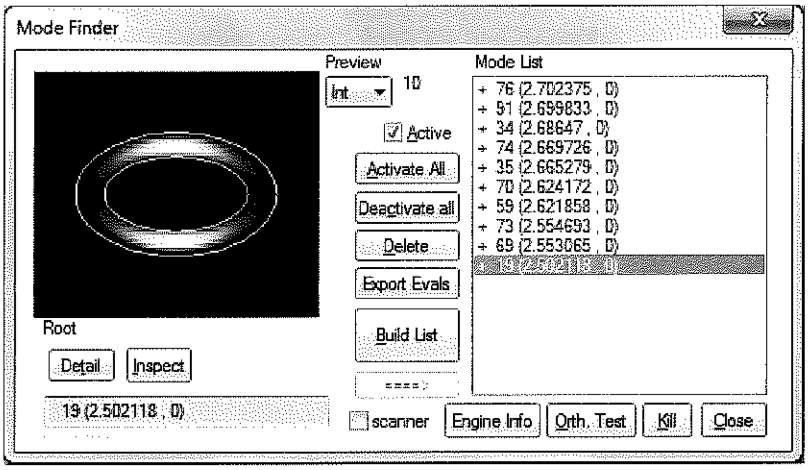
> Click on **Engine Info**.

You should see the following, showing the triangular mesh discretisation used by the FEM solver:



Notice that the structure is precisely discretised, with the triangles following the contours of the objects.

- Click **Build List**.
- Click **Start**, making sure that **New Build** is checked.



- Click the Orth. Test. button. Notice that the orthogonality is much better than with the FDM Solver.

The FEM Solver can accurately discretise even curved shapes such as this.

4.2 Example 2 - Transparent BCs: finding leaky modes

4.2.1 PML boundary conditions.

FIMMWAVE supports *absorbing boundary conditions* for simulating radiation losses off the side walls. The technique used is usually referred to as the *Perfectly Matching Layers (PML)* boundary conditions.

PML's can be used for simulating transparent walls in propagation problems when using the propagation tool *FIMMPROP*, and also for determining the mode loss of a "Leaky mode." The absorption is controlled via the **PmlWidth**, which can be positive or zero. Zero value indicates no absorption (i.e. a hard wall).

This simulation of absorption at the boundaries is achieved by introducing an artificial "complex thickness" near the boundary. This amounts to a "co-ordinate stretching" in the complex plain, and has the effect of attenuating any wave travelling towards the computational boundaries. The following simple argument shows why this is the case.

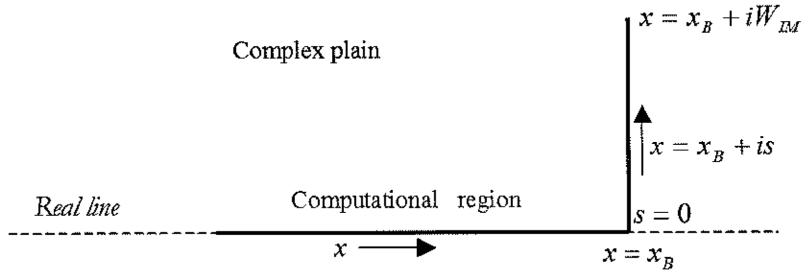
Consider the outward radiating field in a computational region outside the waveguide. At any point in this region, the radiating field can be considered as a linear combination of outward travelling waves of the form:

$$F \sim e^{ik_x x}$$

Where "x" is the lateral co-ordinate, and k_x is a lateral propagation constant. Now near the boundary the co-ordinate "x" becomes imaginary:

$$x = is + x_B$$

Where x_B is the value of x at the boundary of the computational region, and "s" is the distance along the imaginary axis.



Within this region, the outward travelling waves will take the form:

$$F \sim e^{ik_x (is + x_B)}$$

So the field will decrease exponentially along the imaginary axis. At the boundary of the imaginary thickness, the wave will have attenuated by a factor of

$$e^{-k_x W_{IM}}$$

Reflections of this wave off the boundary will also be attenuated by this factor. By the time the reflected wave exits the complex boundary region, the reflected wave will have attenuated by a total factor of

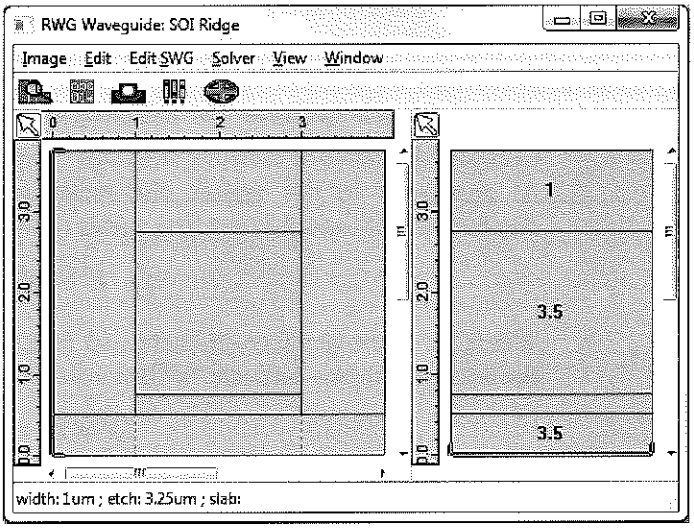
$$e^{-2k_x W_{IM}}$$

Hence the complex co-ordinate stretching has a similar effect to an absorbing material, with the important distinction that no reflections occur at the interface of the complex boundary region (*s=0*).

4.2.2 Finding leaky modes

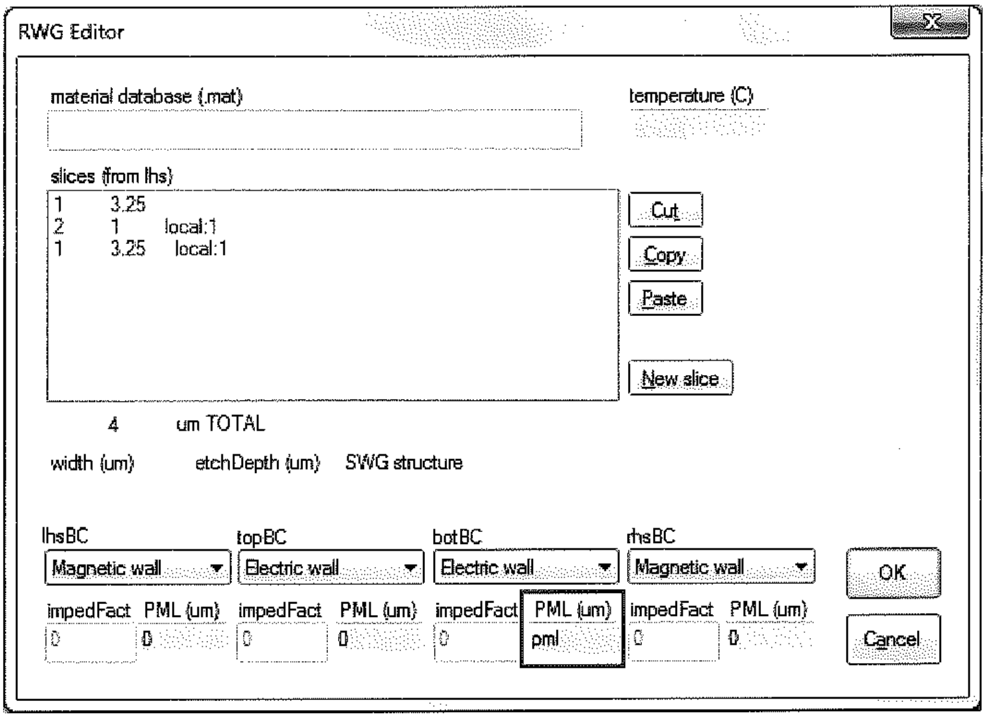
We now show how to find the mode loss for a leaky mode efficiently and accurately. We can do this by using the PML boundary conditions.

- Open the waveguide *'Example2\SOI Ridge'*' You should see the following.



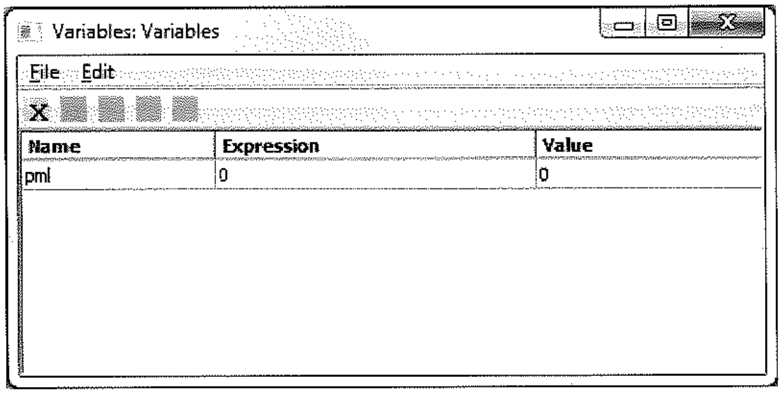
In this example the guided mode will "leak" through the 0.25um SiO2 layer into Si substrate. To obtain a value for this loss we therefore must put a PML boundary condition on this on the bottom boundary. We will need to use a complex solver in order to use PMLs, as the corresponding modes will have a complex propagation constant. In this case we will choose to use the complex FDM Solver.

- Click on the  button. This will show the *RWG* Editor panel.
- In the **botBC PML (um)** box you can see that the value is set to the variable "*pml*". Click **OK** to close the window.



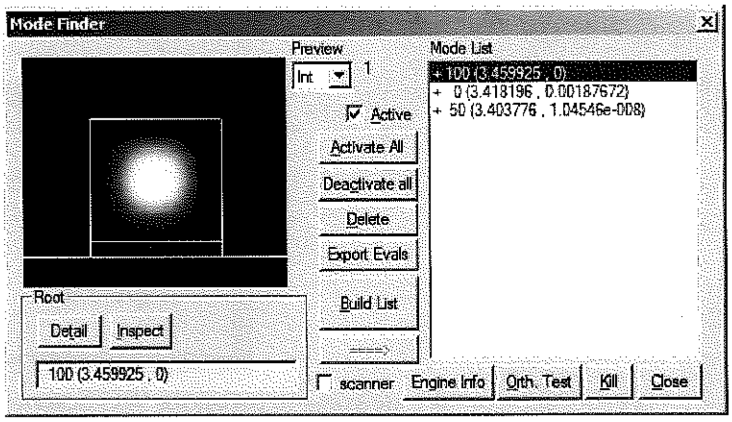
First let us find the mode with hard wall boundary conditions, i.e. with a PML thickness of zero.

- From the project tree, open the Variables node "*Example 2/Variables*." Please ensure that the variable "*pml*" is set to zero; this value will be varied later on using a WG Scanner.



➢ In the RWG Waveguide, click on the button to open the Mode Finder panel.

➢ Click **Build List** then **Start**.

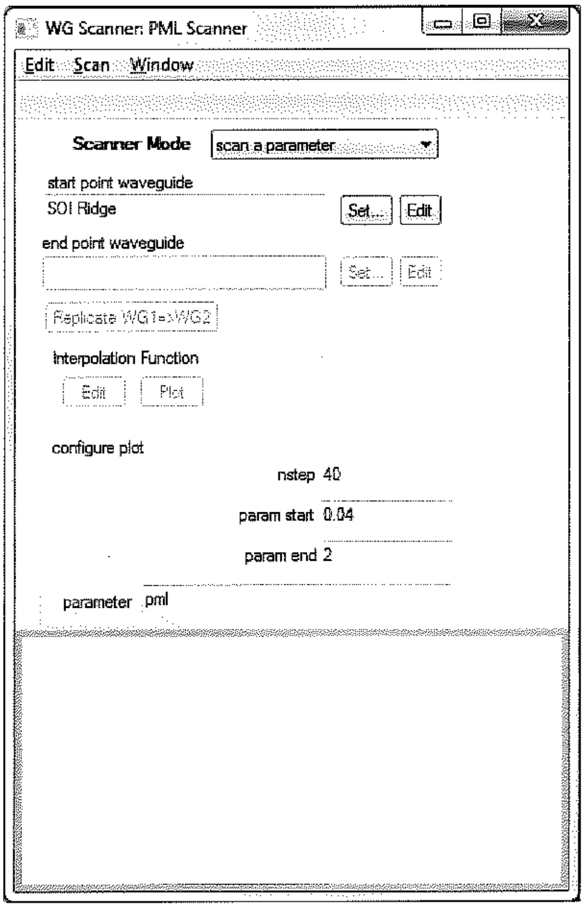


➢ The first mode in the list is the fundamental TE mode of interest.

➢ Click **Close** then **Yes** to save the mode list.

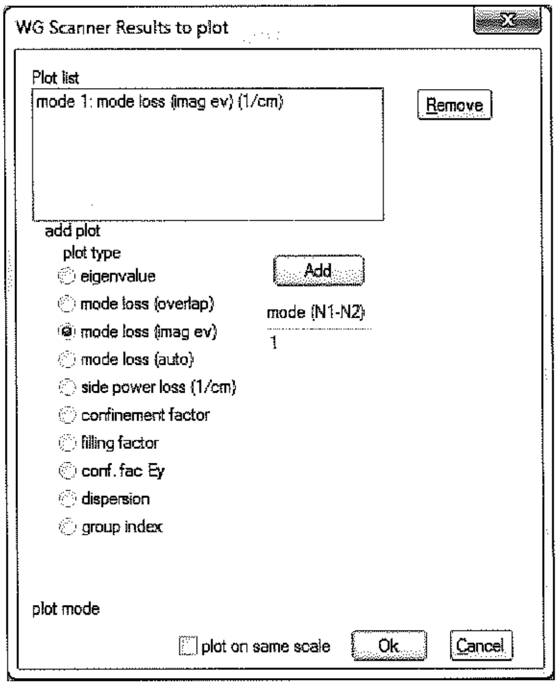
Now we will introduce a PML layer on the bottom boundary. The main problem when using PMLs is to find out what value to give to the PML width. Typically a thickness of 10% of the simulation size is a good guide but the only way to know for sure is to test it, which we will do by studying the evolution of the mode loss when varying the PML width.

➢ Open the WG Scanner "*Example 2/PML Scanner*". You should see the following. Note that we have chosen to "scan a parameter" and that the **parameter** to scan is set to the variable "*pml*", which will be varied between 0.04 and 2um.



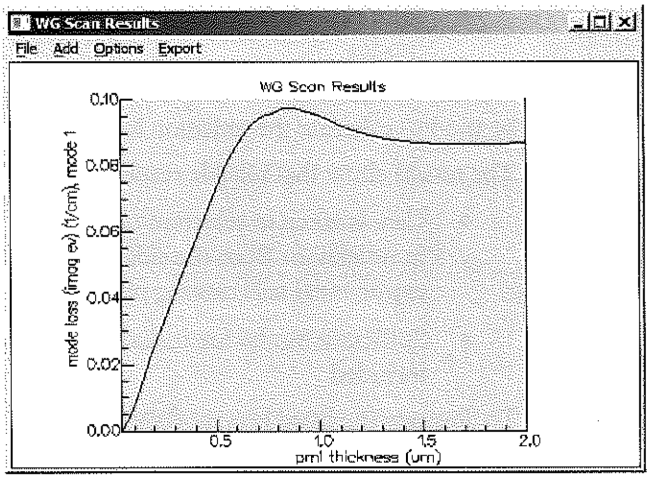
We are going to set up the WG Scanner to plot the Mode Loss as we increase the PML on the bottom boundary.

- Select **/Edit/Options**, and set **enable polishing** to *disabled.*
- From the WG Scanner select **Scan/Start.**
- We are interested in plotting the mode loss of mode 1, so insert this into the **Plot list** box.



➢ Click **OK**.

After a few minutes you should see the following plot.



Notice that the Mode Loss starts off very low and then increases before levelling off above a PML thickness of ~ 1.2 um. Clearly for this structure we need a 1.5 or 2 um thick PML – 50% of the height of the computational window. Usually we don't need quite such a thick PML but there is no way of determining the required thickness in general. So if you want to use PMLs, you should always include a test similar to this in your design study.

Chapter

5

Advanced FIMMPROP Usage

5.1 Example 1 - Bend Radiation Loss Modelling

In this section we consider a bent waveguide section, show how the radiation bend loss can be calculated, and investigate the effect of PML width.

For a curved waveguide section between two straight waveguide sections, there are two types of bending loss.

- Transition loss where the modes of the straight waveguide couple to the modes of the bent waveguide (and vice-versa at the end of the bend)
- Pure bend loss given by the loss coefficient of each of the bent waveguide modes

Herein, we consider an example to illustrate how the bend section in FIMMPROP can be used to calculate the radiation loss

The results show that reliable results can be obtained provided that

- Care is taken in selection of the PML width. Too narrow a PML width, results in reflections from the sidewalls and too wide a PML width results in spurious exponential tails causing physically unreasonable results.
- Sufficient modes are included in the basis set, so that radiation modes are included

PML too close? →

To illustrate the bend loss modelling capability in FIMMPROP, an example is explored based on silica waveguides. The objective is to determine the minimum refractive index delta N (dn), between the core and the cladding, to develop low loss optical components using curves of 20 mm radius. Typically, such a waveguide will be used in the development of directional couplers, Mach Zehnders and Y junctions.

The challenge for the design is to balance the requirements of compact device length (high dn) against low loss coupling to optical fibre (low dn). The starting waveguide has a cladding of refractive index of 1.447 and a core which is 5um x 5um with a refractive index of 1.453 and the components will be operating at a wavelength of 1.5 um.

5.1.1 Getting started

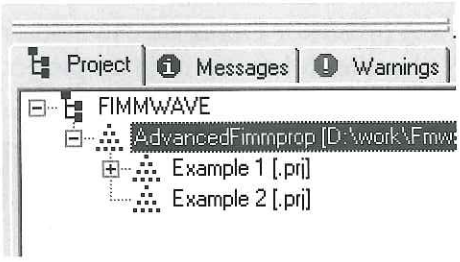
We will construct all our components within a project provided with this document, so the first thing you must do is open it.

➢ Click on the icon on the tool bar.

➢ Navigate to the Tutorial Examples directory and select "`AdvancedFimmprop.prj`"

➢ Click **Open**.

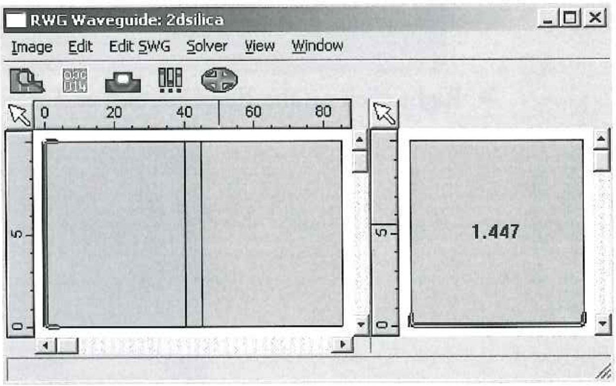
The following should appear in the *Project Tree*.



For reference purposes, the file "Complete_AdvancedF`immwave.prj`" contains all the examples constructed in this tutorial section.

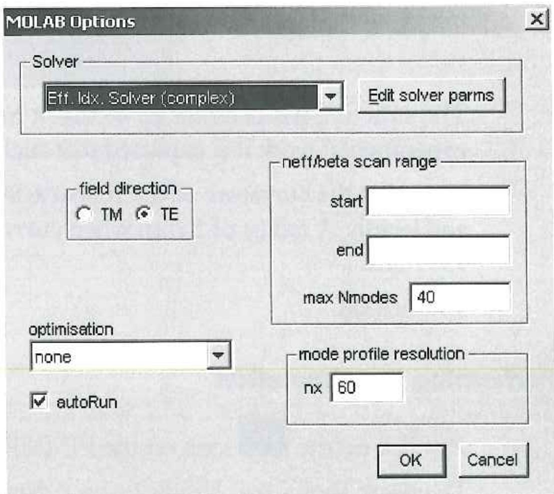
To simplify the computations, the problem has been reduced from a 3D problem to a 2D problem.

➢ Open the waveguide *"Example1\2D Silica."* You should see the following.



The 1D waveguide, "2dSilica," is described by two 40 um layers of refractive index 1.447 and a central 5 um layer of refractive index 1.4506.

➢ Click on the  button. This will show the ***MOLAB Options*** panel.



Note that the mode solver is set to "**Eff. Idx Solver (Complex)**"

➢ Click **Edit solver parms.**

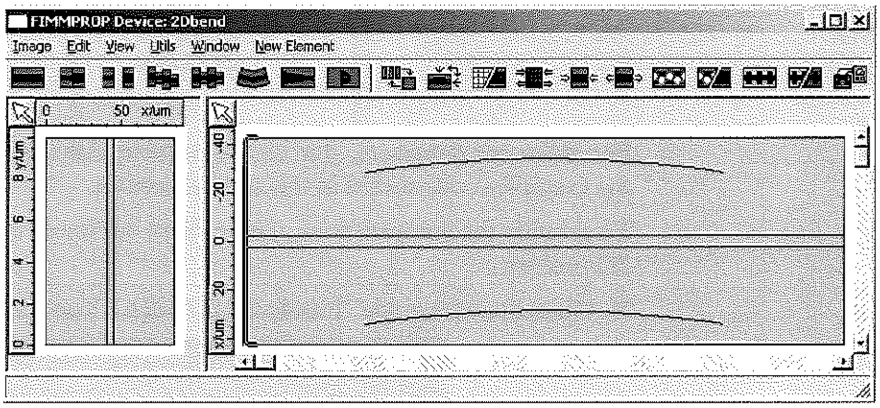
Note that the quasi-2D (XZ) option is selected, this allows the modes of the 1d waveguide to be found analytically.

➢ Click **OK**.

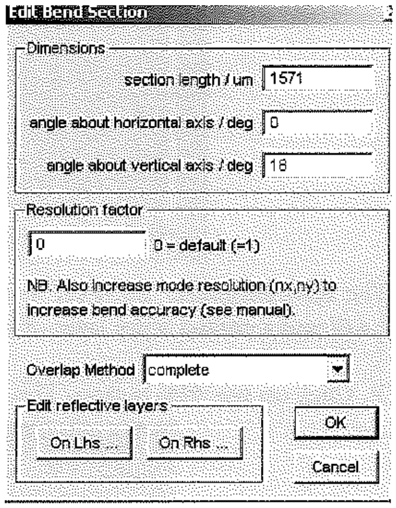
➢ Click **OK**.

A FIMMPROP Device used to model the bend loss using the FIMMPROP Bend Section has already been set up in the project.

➢ Open the FIMMPROP Device *"Example1\2Dbend."* You should see the following.



➢ Right-click on the Bend Section and select **Edit Properties...**



The structure has been set up so that it has a 5-mm bend radius. From previous experimental work it is expected that such a bend will be lossy.

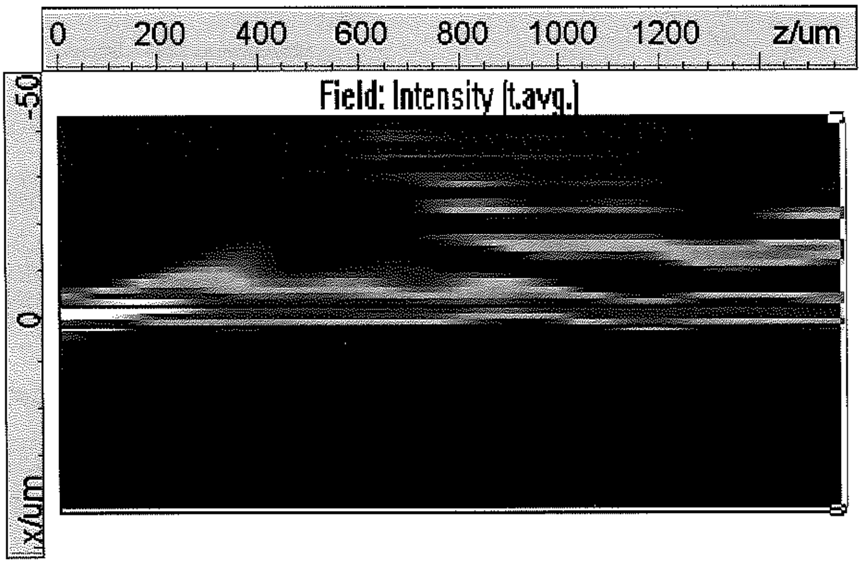
Note that the curvature of the FIMMPROP Bend Section is specified in terms of angle and length. A radius of 5 mm when curved through 18 degrees will have a length of 1571 um.

➢ Click **OK**.

5.1.2 Performing the simulation

➢ Click on the icon on the FIMMPROP Device toolbar

The result shows that light is being radiated out of the waveguide, as expected. However the light is reflected at the boundaries and some of the light is coupled back into the waveguide to give an incorrect answer.



Reflections from boundaries can be significantly reduced by the use of absorbing boundary conditions (PMLs). These are set as properties of the waveguide *"2dsilica"*

- From the project tree, open the waveguide *"2dsilica"* again (if it is not already open).
- Click on the button. This will show the *RWG* Editor panel.
- Type 1.0 in the **lhsBC PML (um)** box
- Type 1.0 in the **rhsBC PML (um)** box
- Click **OK**.
- Repeat the above simulation.

The PML has reduced the reflections from the side wall significantly.

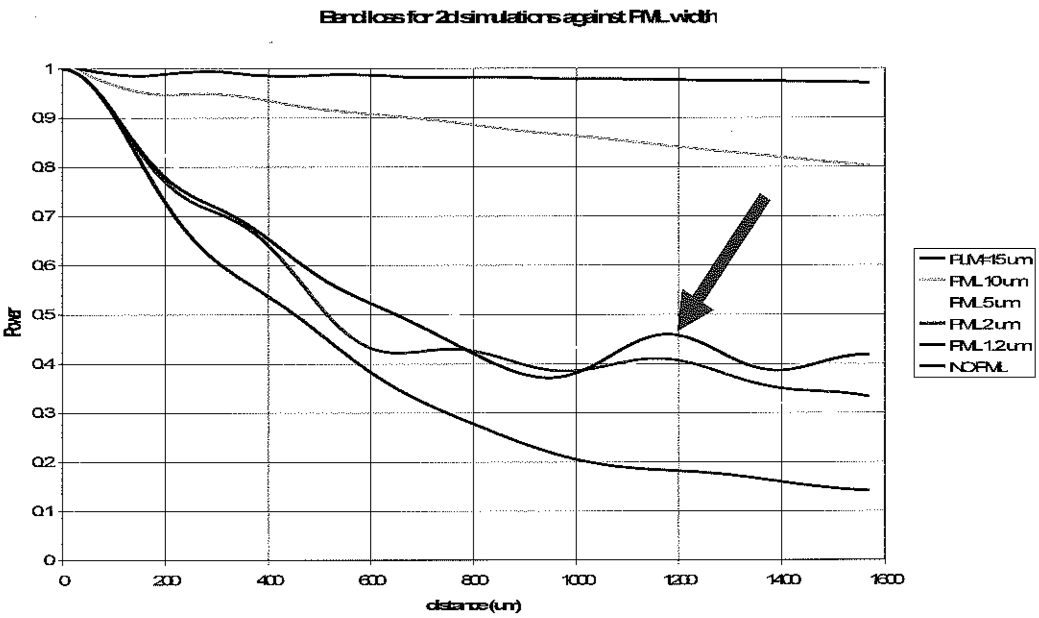
It would be natural to consider increasing the size of the pml layer even further.

- Click on the button on the *"2dsilica"* waveguide.
- Type 10 in the **lhsBC PML (um)** box
- Type 10 in the **rhsBC PML (um)** box
- Click **OK**.
- Repeat the above simulation.

In this case, spurious exponential tails have been created, leading to a physically unreasonable condition.

Caution is needed when determining the pml settings.

To investigate the effect of PML width, plots have previously been made of power in the fundamental mode against PML width. The results of these simulations are shown below.



Without a PML, oscillations occur as the reflected light couples back into the fundamental (highlighted above). Increasing the PML width reduces the oscillations and the smoothest result is obtained for a PML width of about 1.2 um. Further increase in the PML width leads to less light being coupled out of the waveguide due to the errors introduced by the exponential tails of the radiation modes.

Thus the following conclusion can be drawn from these simulations.

When simulating a curved waveguide with high radiation loss, it is important to use PML's correctly to reduce boundary reflections. If the PML is too small then reflections will occur and if the PML is too strong then light is not coupled correctly into radiation modes. In the example above a PML width of 1.2 um is optimum, but different waveguide geometries will require different widths.

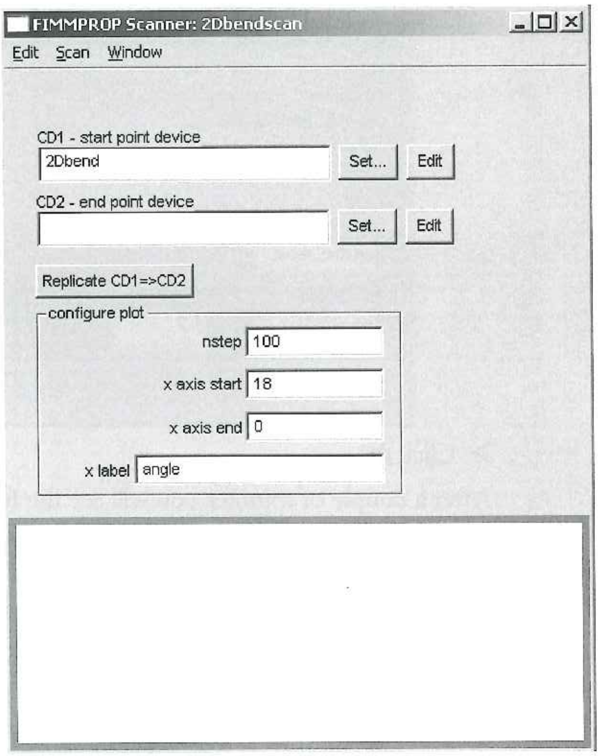
5.1.3 Further study

We will now proceed to looking at the bend loss as a function of bending angle using a PML width of 1.2 um.

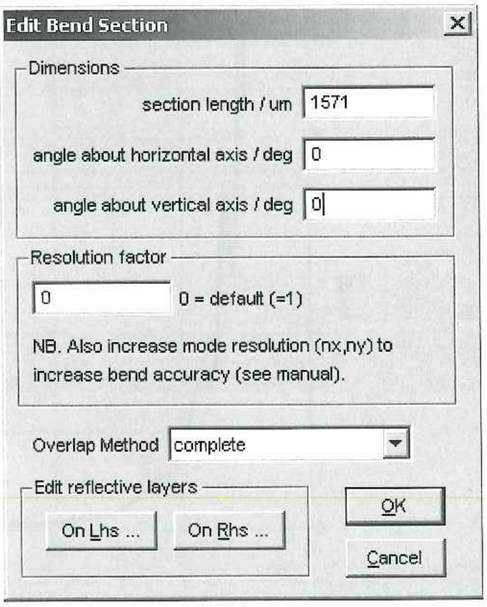
- Click on the  button on the *"2dsilica"* waveguide.
- Type 1.2 in the **lhsBC PML (um)** box
- Type 1.2 in the **rhsBC PML (um)** box
- Click **OK**.

The FIMMPROP Scanner will be used to vary the angle turned through by the bend (and hence the radius). Scanning from an angle of 18 degrees to zero represents bend radii from 5 mm to infinity (straight waveguide).

- Open the Fimmprop Scanner *"Example 1/2Dbendscan."* You should see the following.



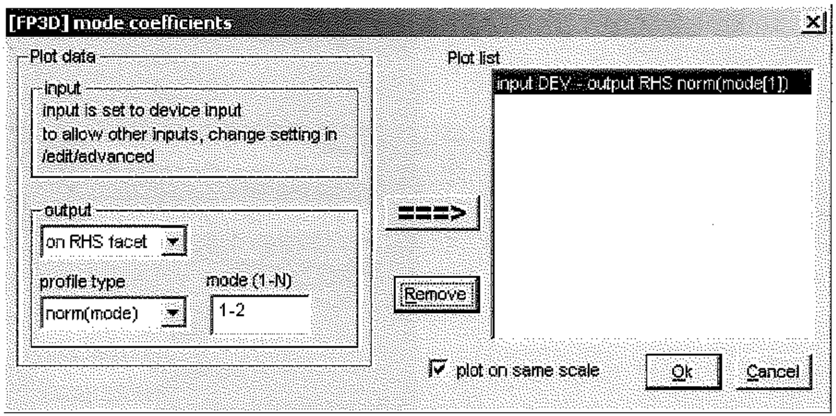
- Click on the **replicate CD1=>CD2** button. This will make a copy of the device. You should see this copy appear in the *Project Tree*.
- Click on the **Edit** button next to the **CD2 – end point device** box.
- Right-click on the FIMMPROP Bend Section and select **Edit Properties...**
- Type 0 in **angle about vertical axis / deg** box



- Click **OK.**

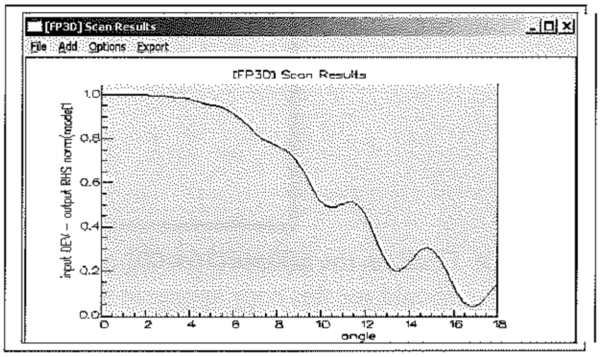
We are now ready to begin the scan

- From the FIMMPROP Scanner select **Scan/Start.**
- We are only interested in the fundamental mode transmission, highlight the second line in the **Plot list** and press **Remove.**

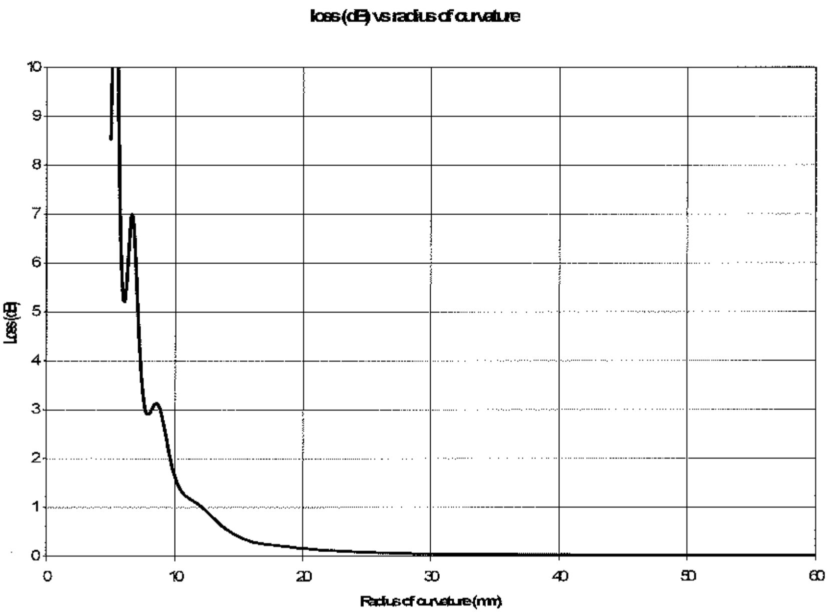


➢ Click **OK**.

After a couple of minutes you will see the following:



For reference this is plotted on a dB scale below.



The results show that the initial waveguide design has good bend performance down to about 30 mm, but is showing some loss at 20 mm.

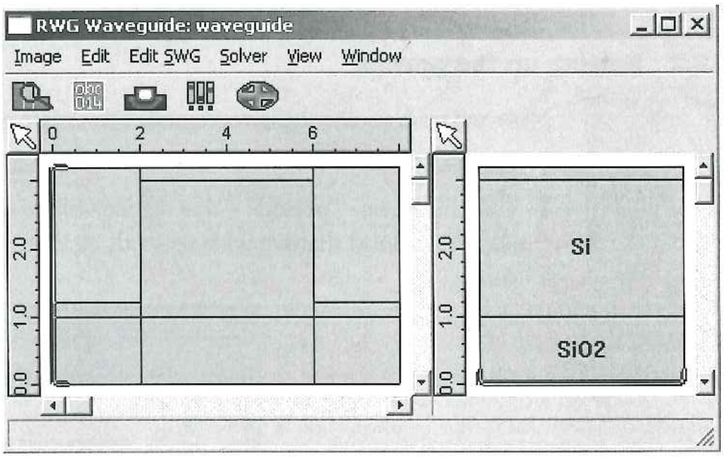
FIMMPROP has been used to model radiation loss in a silica waveguide. The results show that careful selection of PML width is important to achieve an accurate result.

5.2 Example 2 - Modelling Gratings

This example is based on an etched grating produced in an SOI ridge waveguide. The grating will be set up using periodic structures and scan the length of the grating to look at the grating transmission reflection and also examine the birefringence. The grating is produced by periodically etching the top of the SOI ridge waveguide 0.2 um deeper.

5.2.1 Getting started

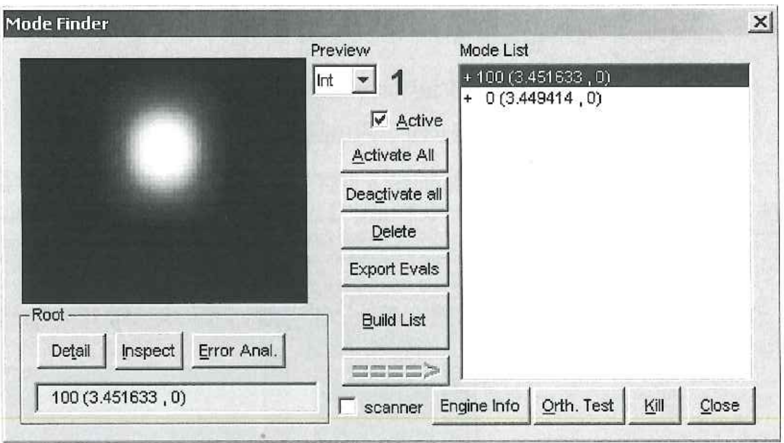
➢ Open the waveguide *"Example2\waveguide."* You should see the following.



In this example we are only going to consider light in the fundamental TE and TM modes.

➢ Click on the  button.

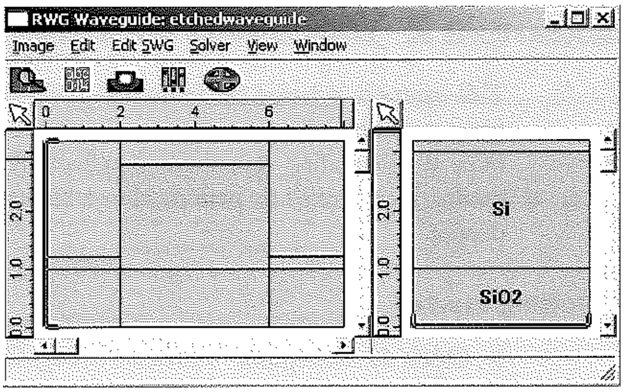
Notice that these two modes are already in the Mode List.



➢ Click **Close**.

➢ Open the waveguide *"Example2\etchedwaveguide."*

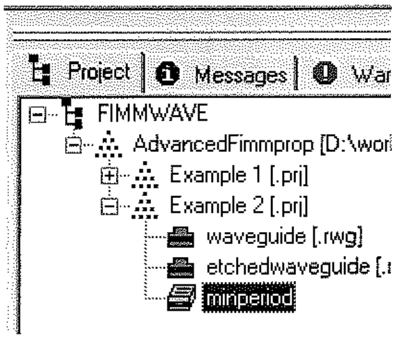
➢ Notice that the etching over the ridge is 0.2um larger than in the previous waveguide.



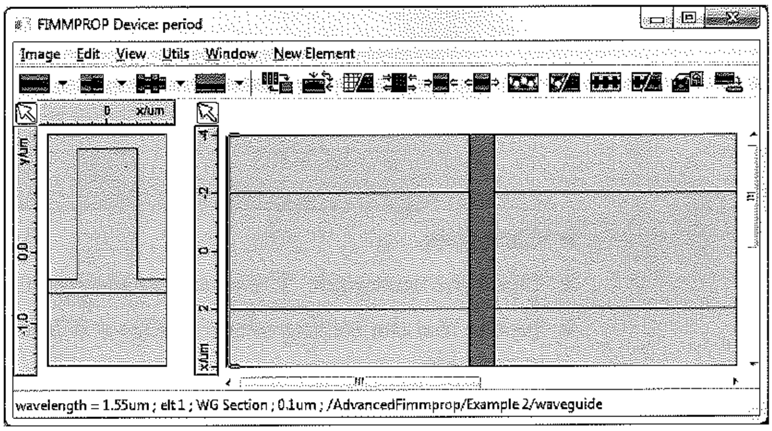
5.2.2 Setting up the grating

Now we wish to use these waveguides to create a grating.

- Right click on "*Example 2*" in the project tree and add a FIMMPROP Device.
- Call this device "period" - this device will be used to create a grating which has the shortest period that we wish to study (0.2 um). The project tree should look like this.

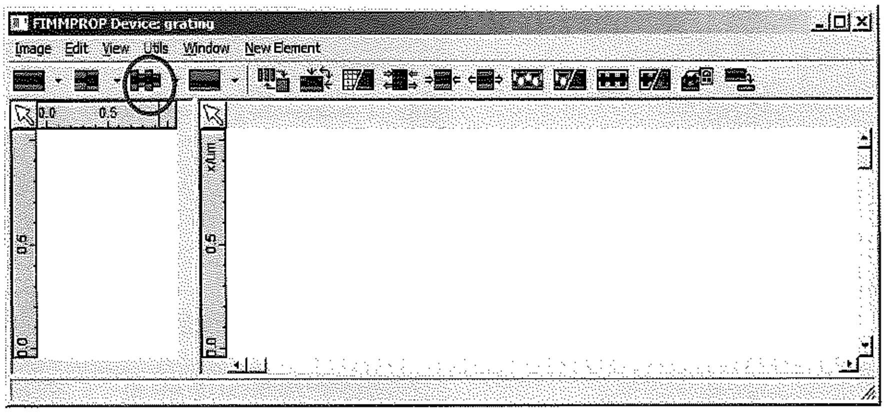


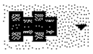
- Open the "period" Device and add a new WG Section – select "*waveguide*" from the list.
- Type 0.1 in the **section length/um** box.
- Append a new Simple Joint
- Follow this by a second WG Section - select "*etchedwaveguide*" and again set the length to 0.1 um. The device should look as shown below.

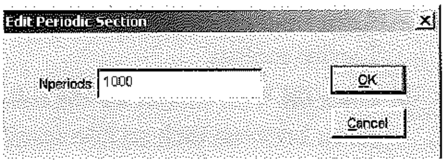


This FIMMPROP Device will be the periodic unit of our grating. We will insert this into another FIMMPROP Device as a Periodic Section.

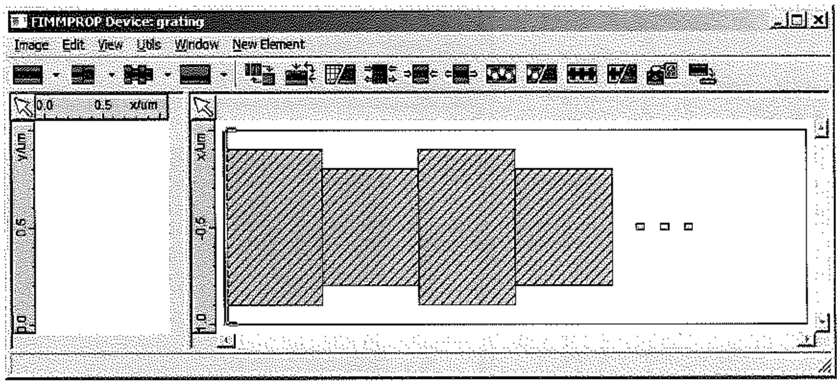
- Add another FIMMPROP Device to the project "*Example 2*"
- Call it "*grating*" and open this device.



- Add a new Periodic Section: click on the arrow next to  and select the second icon .
- Set the number of elements to 1000.



- Click **OK**.



- Click on the device "*period*" and click **Edit/Copy This Node**
- Click on the device "*grating*" device and right-click on the Periodic Section and select **\Paste Sub Element.**
- We will now introduce the joint in-between the sections: right-click on the Periodic Section and select **\New Sub Element\Simple Joint** then click **OK**.

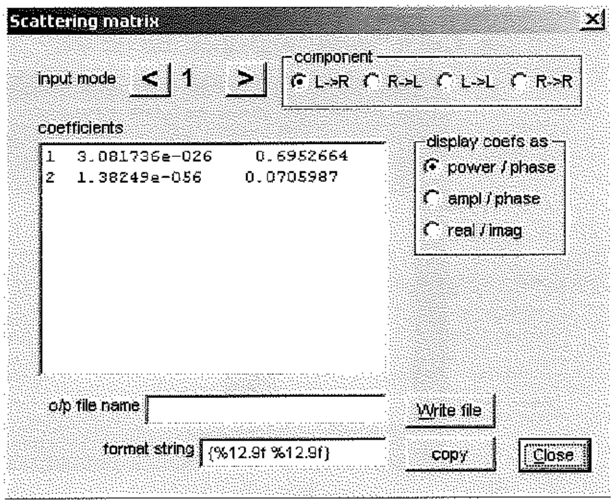
A periodic grating with a 1000 periods (of "*period*") has now been created.

5.2.3 Performing the simulation

Before calculating the Scattering Matrix of this device we first need to set the wavelength

- Select **Edit/Options** and Type 1.55 in the **Lambda/um** box.

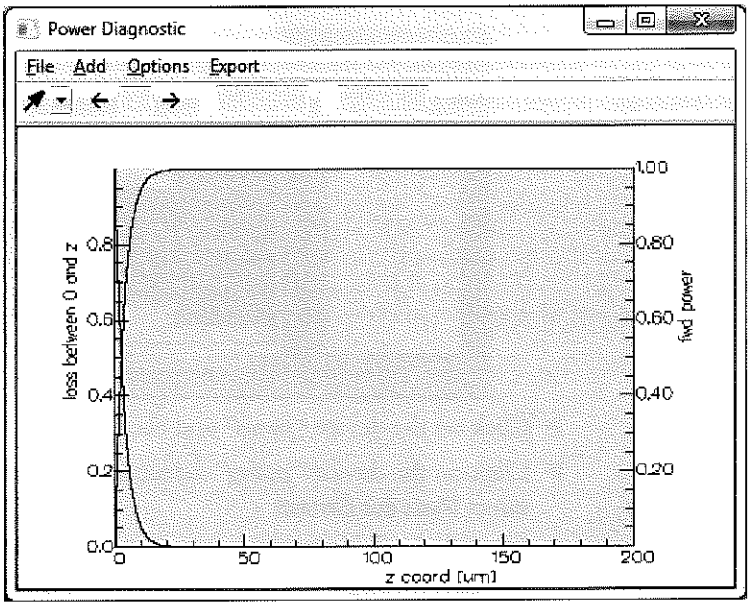
➢ To calculate the *Scattering Matrix*, click on icon on the toolbar. After a couple of seconds, the following will appear.



The transmission is very low!!! This is because there are many overlap integrals at each of the joints and small numerical losses build up. This can be easily seen by plotting the Calculation Diagnostics.

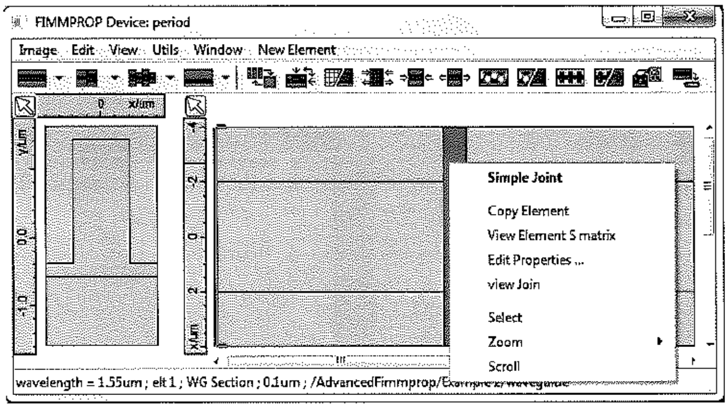
➢ Click on the Loss Diagnostic button towards the right end of the FIMMPROP Device toolbar.

➢ Click **OK**. You should see the following. Notice that due to numerical losses at the joints between each section result in almost all power being lost from the simulation after 20um or so.



However we know that first order gratings are not lossy, so we can use power normalisation to eliminate these artificial numerical losses so that we can actually see the resonance(s).

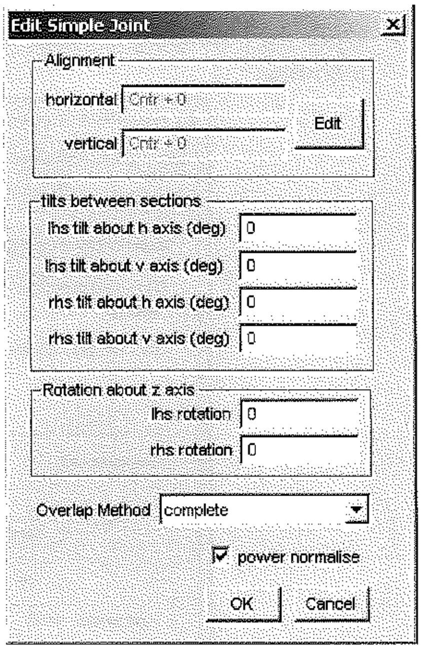
➢ Right-click on the joint in the device "*period*" and select **Edit Properties...**.



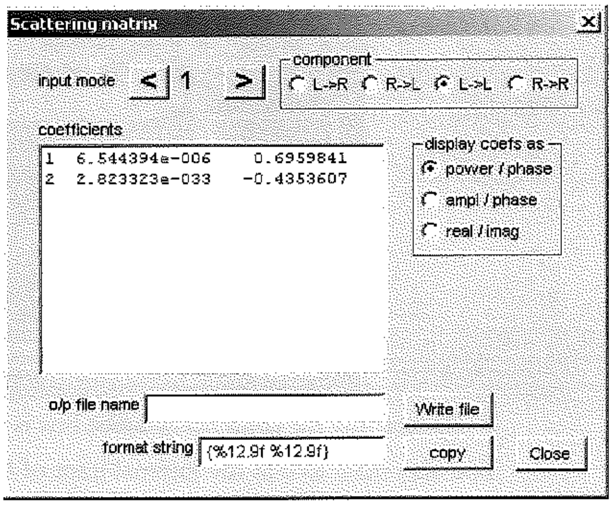
- Ensure that the **power normalisation** box is ticked as below.

You also need to edit the internal joint of the Periodic Section (in-between successive periods of *"period"*) of the device *"grating"*.

- Click on the Periodic Section in the device *"grating"* device and select **edit sub Joint/Edit properties**.
- Ensure that the **power normalisation** box is ticked.



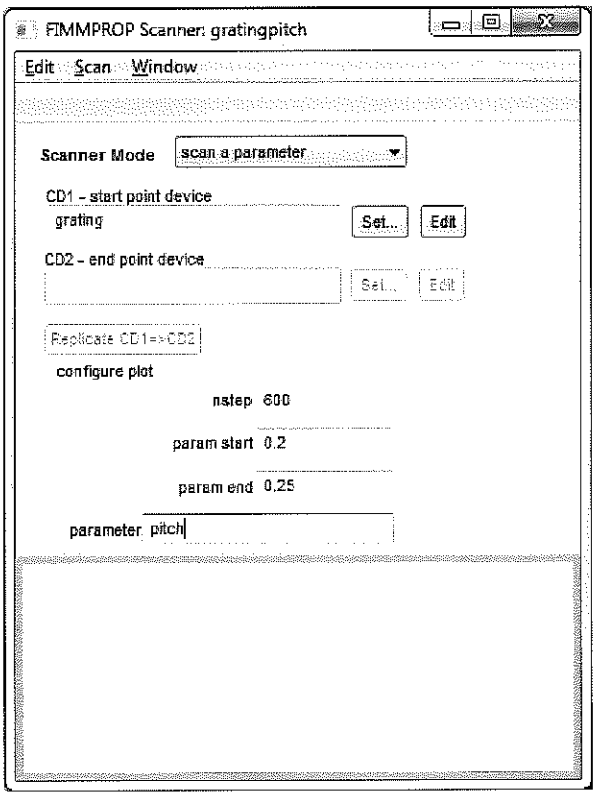
- Recalculate the Scattering Matrix, the transmission is now virtually 100% (by definition).
- Selecting L>L in the scattering matrix shows that the reflection is very low as below. This is because the grating pitch is not at a resonant condition, hence no reflection occurs.



It is much quicker to scan the grating period than the wavelength. This is because the modes only need to be calculated once and the scan will be significantly faster.

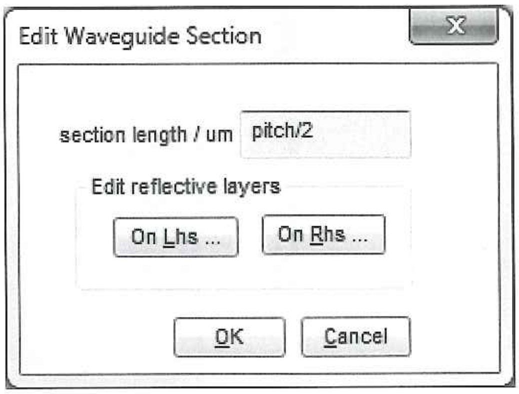
The silicon has a refractive index of 3.45 at a wavelength of 1.55 um and so we expect to have a first order grating with a period of 1.55/3.45/2=0.225 um.

➢ Open the Fimmprop Scanner *"Example 2/gratingpitch"*. You should see the following.

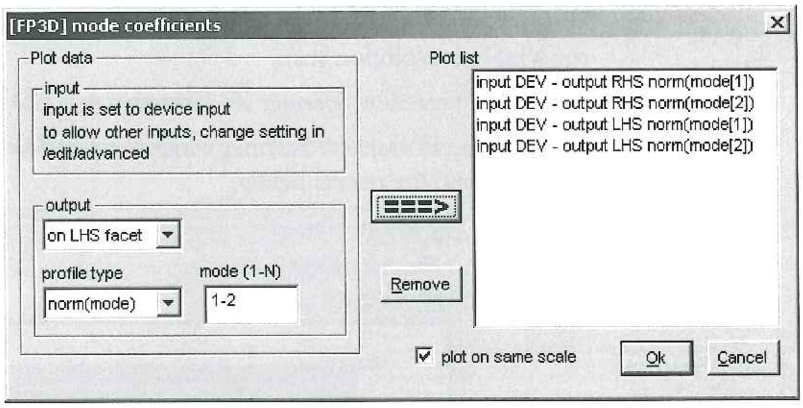


➢ Open the Variables node *"Example 2/Variables"*; you will see that that there is a variable named "pitch". We will use this variable to parameterise the length of our period; note that the same name is used for **parameter** in the FIMMPROP Scanner.

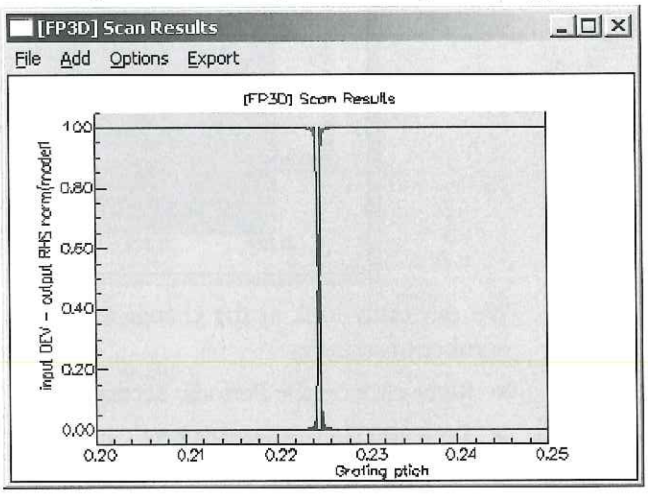
➢ Open *"period"* then right-click on the first waveguide section and select **\Edit Properties...**, then set **section length** to *"pitch/2"* then click **OK**.



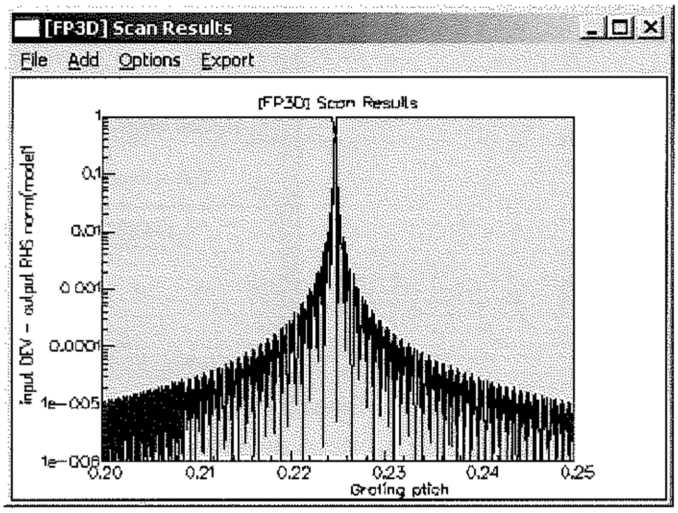
- Repeat this operation for the second waveguide section.
- Back to the FIMMPROP Scanner, select **Scan/Start**.
- We want to look at both the transmission and reflection so select the output **on LHS facet** and type 1-2 in the **mode (1-N)** box.
- Click on the **===>** to add the reflected modes.



- Click **OK**.
- When the scan completes, click **OK** again and the plot will appear



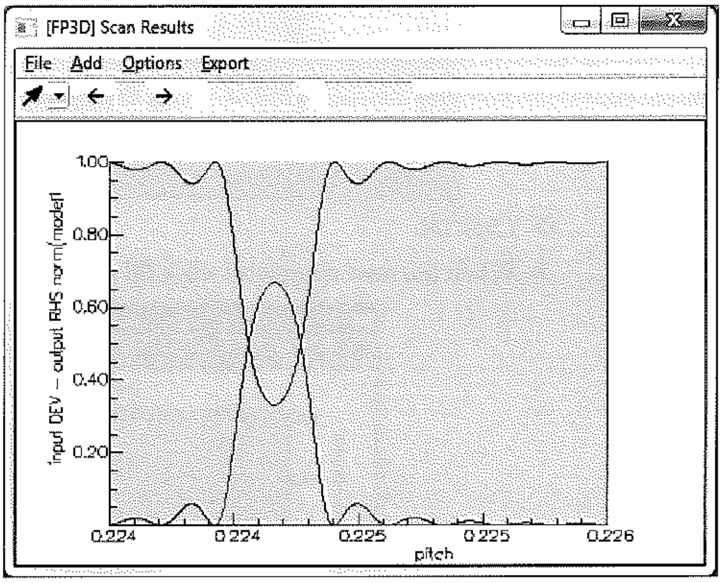
Often these plots are more useful plotted on a log scale. This has been done for you below. The log scale allows the sidebands to be seen and also the depth of the grating can be seen more clearly.



5.2.4 Further study

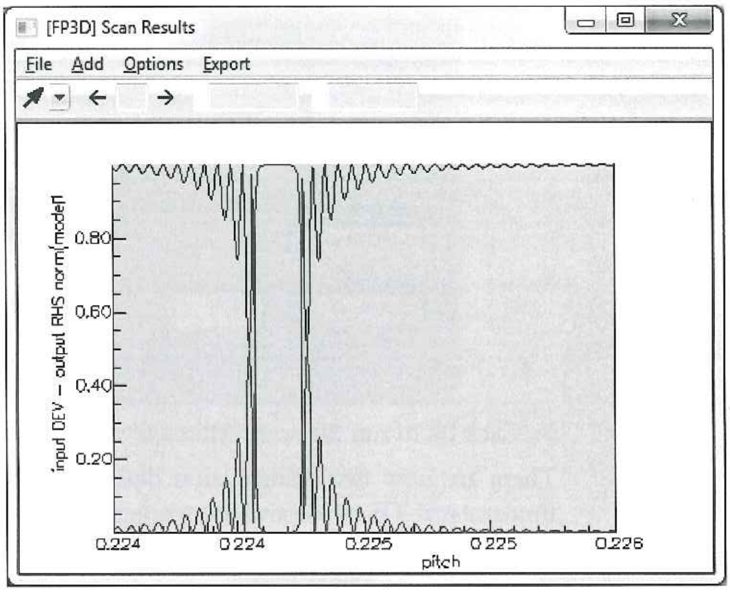
We can see that the grating pitch is between 0.224 um and 0.226 um so we will now run a higher resolution scan.

- In the FIMMPROP Scanner set **param start** to 0.224 and **param end** to 0.226.
- From the FIMMPROP Scanner window select **Scan/Start** to run the simulation again and obtain the results below.



We can easily look at the change in behaviour for a structure consisting of a larger number of sections.

- Right-click on the Periodic Section of the "*grating*" device. Select **Edit Properties...**
- Type 4000 in the **Nperiods** box
- Click **OK**.
- Re-run the scan by selecting **Scan/Start** from the FIMMPROP Scanner window. You should see the following.

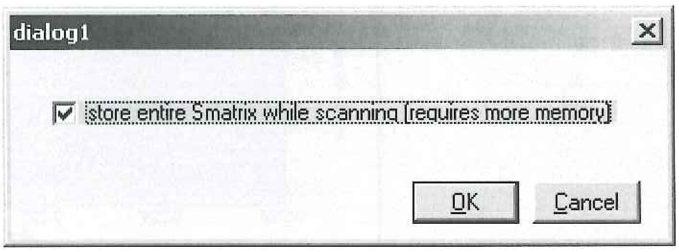


➢ Reset **Nperiods** to 1000 in "*grating*".

We would now like to look at the birefringence effect in the grating device. This requires storing the whole of the matrix data for the duration of the scan. This can sometimes be quite memory intense so by default this is switched off.

➢ In the FIMMPROP Scanner window click **Edit/Advanced** and tick the box.

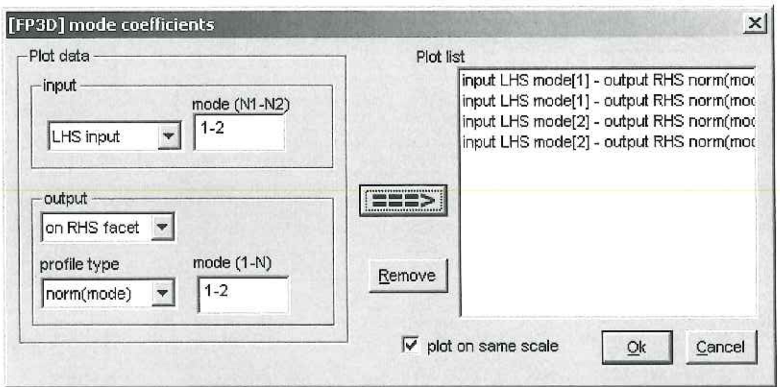
➢ Click **OK**.



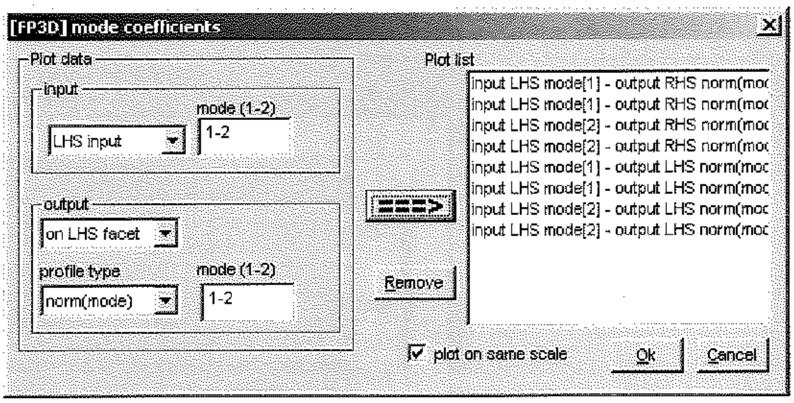
➢ Select **Scan/Start**

➢ Delete everything from the **plot list**.

➢ Select the input as **LHS input mode 1-2** and output on **RHS facet mode 1-2** click the **===>** button.

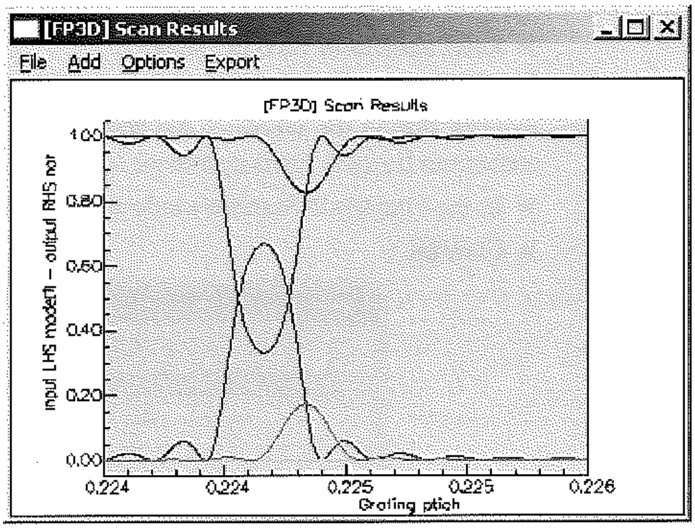


➢ Change the output to **on LHS facet** and click the **===>** button a second time. The mode coefficients dialog box should look as below.



➤ Click **OK** to run the scan. After a few seconds you should see the following plot.

There are now two transmission dips. The stronger dip in red corresponds to the fundamental TE mode and the weaker reflection dip corresponds to the fundamental TM mode.



The approach described could be used to study polarisation effects in gratings and the birefringence could be modified by changing the waveguide width, the etch depth for the waveguide or the grating. For example in an add-drop filter for WDM applications, the birefringence would want to be minimised whereas in a polarisation beam splitter, the birefringence would need to be increased to separate the two peaks. Those two design examples are left as an exercise for the student!