ECE 595, Section 10
Numerical Simulations
Lecture 30: Applications of the Cavity Modeling Framework (CAMFR)

Prof. Peter Bermel

March 27, 2013

Recap from Monday

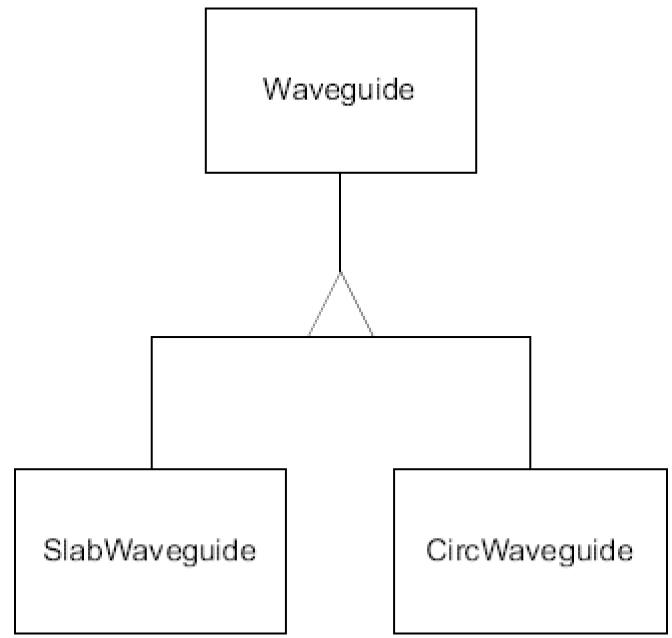
- Rationale for CAMFR: natural form of solutions is semi-analytic in 4 types of eigenmodes
- Subsequent steps:
  - Apply Lorentz reciprocity to match BC's
  - Propagate within layers using S-matrix method
  - Apply inputs to calculate physical outputs

CAMFR Architecture

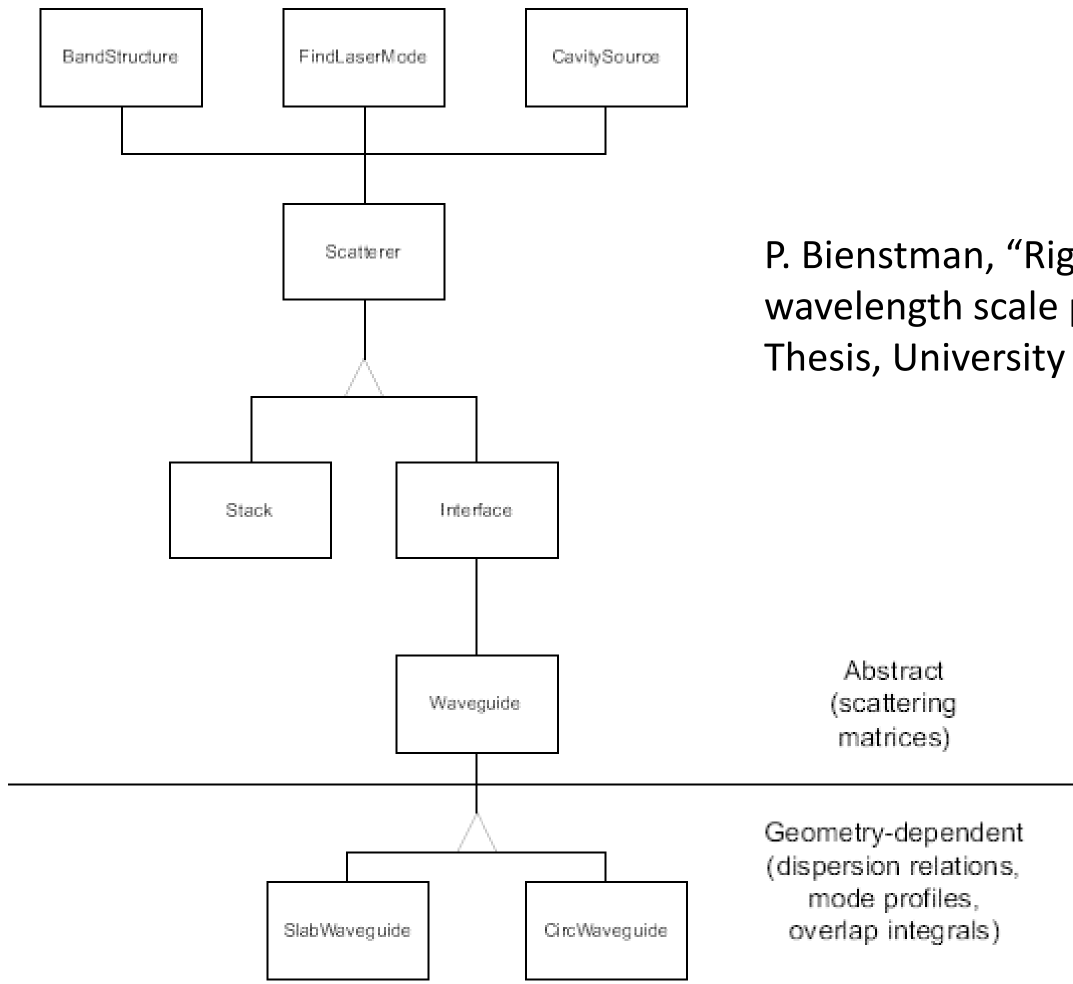
- Three key architectural elements:
  - User interface
  - Core logic
  - Low-level numerical routines
- Basic concept is to make each level independent yet interlocking with the others

CAMFR Architecture

- Built on object-oriented framework:
  - Abstract data types including slabs, waveguides
  - Encapsulated/reusable code
  - Polymorphism
- Implemented as Python library



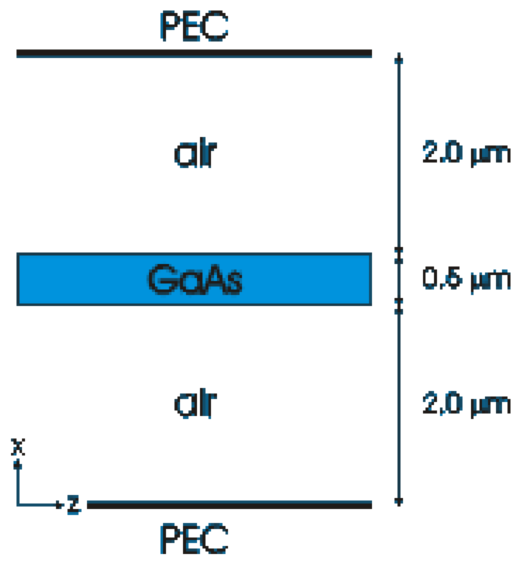
CAMFR Architecture



P. Bienstman, "Rigorous and efficient modeling of wavelength scale photonic components," Ph.D. Thesis, University of Ghent (2001).

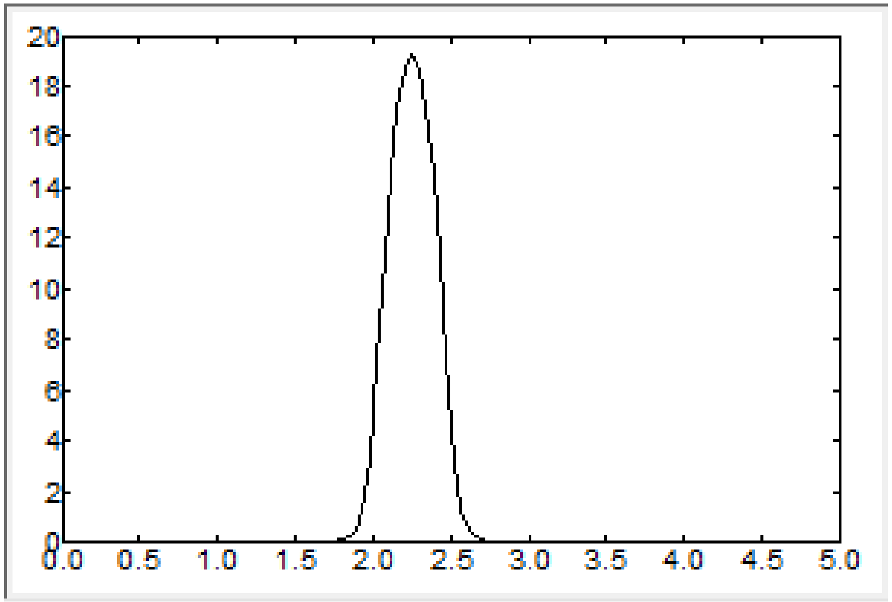
Code: 1D Waveguide

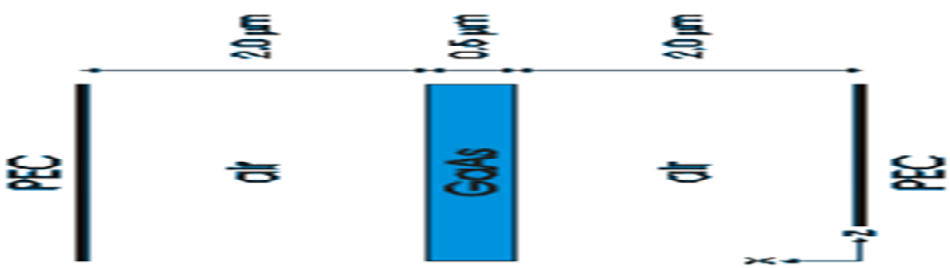
```
#!/usr/bin/env python
from camfr import *
set_lambda(1)
set_N(20)
set_polarisation(TE)
GaAs = Material(3.5)
air = Material(1.0)
slab = Slab(air(2) + GaAs(0.5) + air(2))
slab.calc()
....
slab.plot()
```



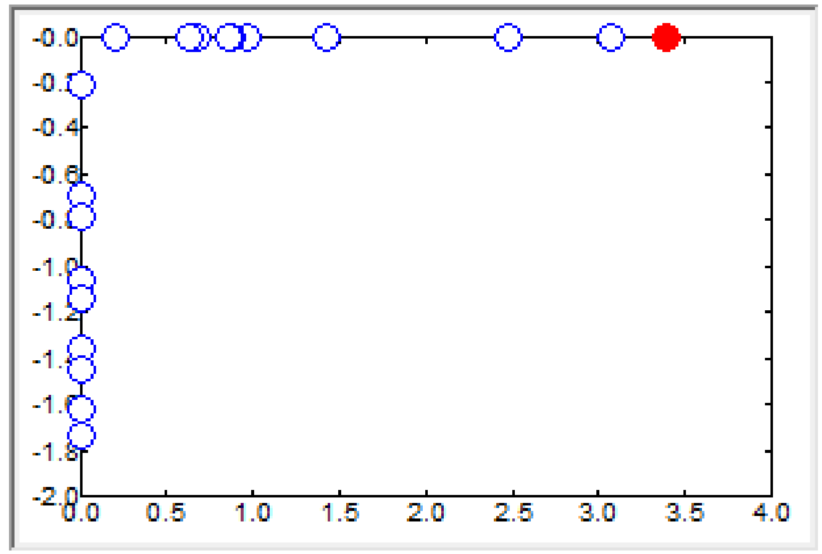
Results: 1D Waveguide

E-field spatial distribution

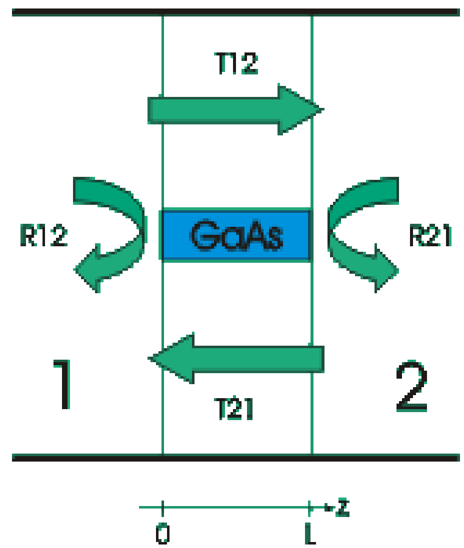




Effective index profile

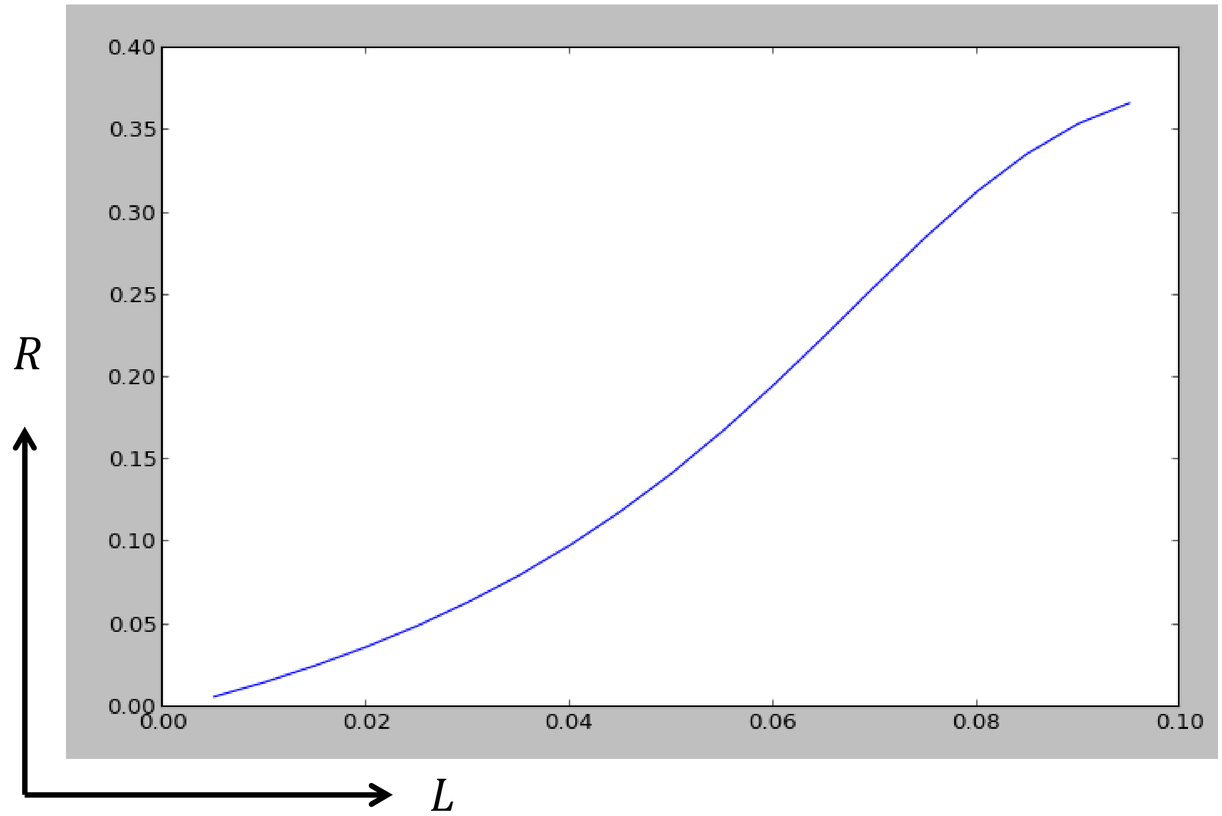


Code: 2D Waveguide




```
...
slab = Slab(air(2) + GaAs(0.5) + air(2))
space = Slab(air(2) + GaAs(0.5) + air(2))
For L in arange(.005,.1,.005):
        stack = Stack(space(0) + slab(L) + space(0))
        stack.calc()
        print L, abs(stack.R12(0,0))
```

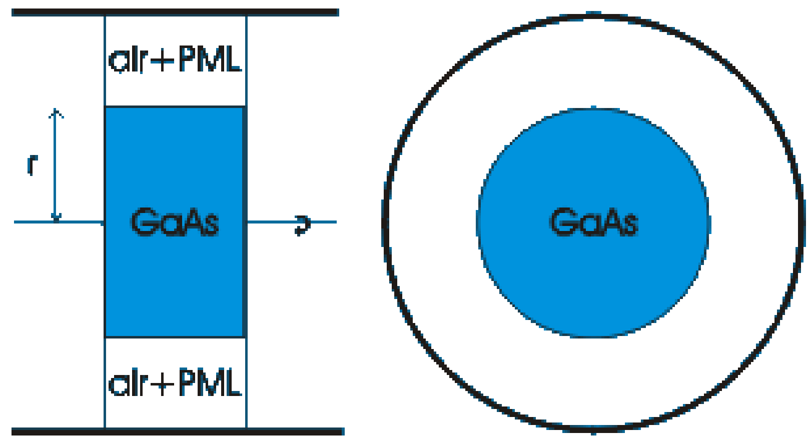
Results: 2D Waveguide



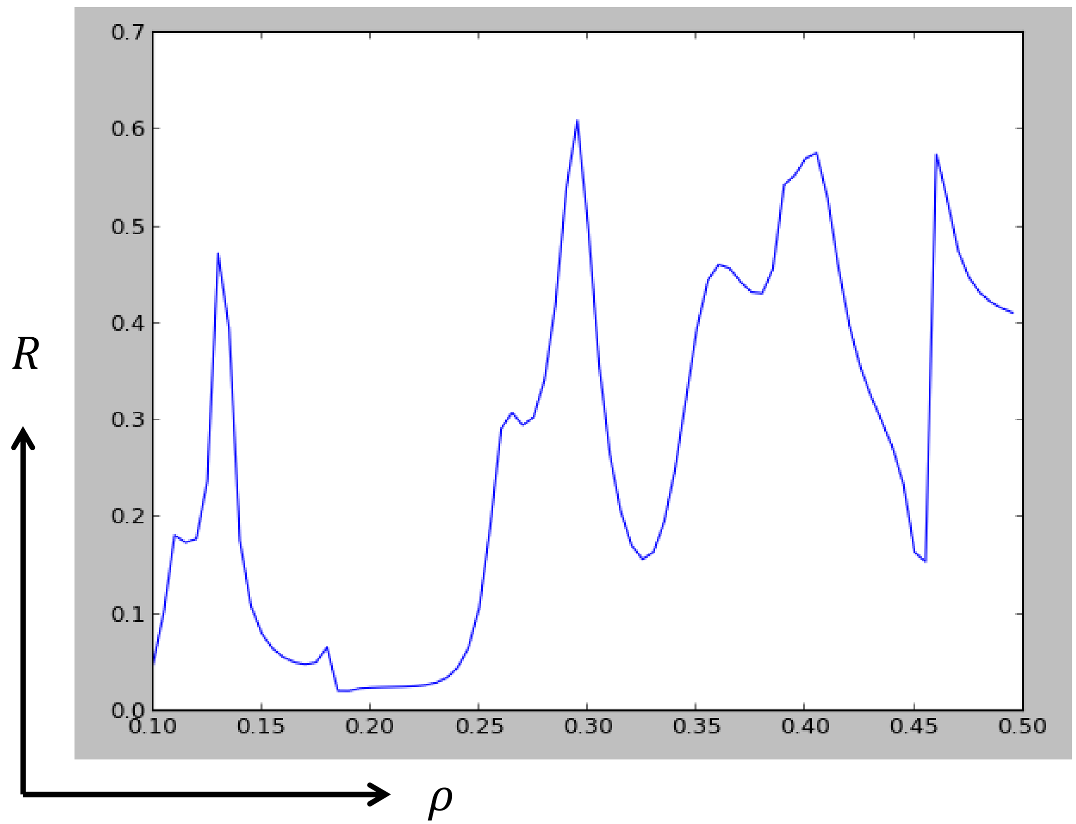
- Can see smooth increase from 0, with nonlinearities at larger *L*'s from interference

Code: Cylindrical Stack

```
...
set_circ_order(0)
set_polarisation(TE)
...
Set_circ_PML(-0.1)
Space = Circ(air(1))
for r in arange(.1,.5,.05):
        circ = Circ(GaAs(r) + air(1-r))
        stack = Stack(space(0) + circ(0.5) + space(0))
        stack.calc()
        print r, abs(stack.R12(0,0))
```



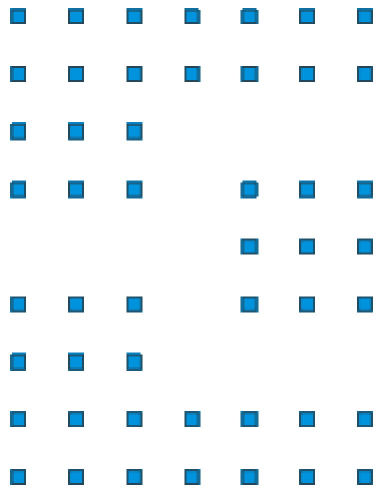
Results: Cylindrical Stack



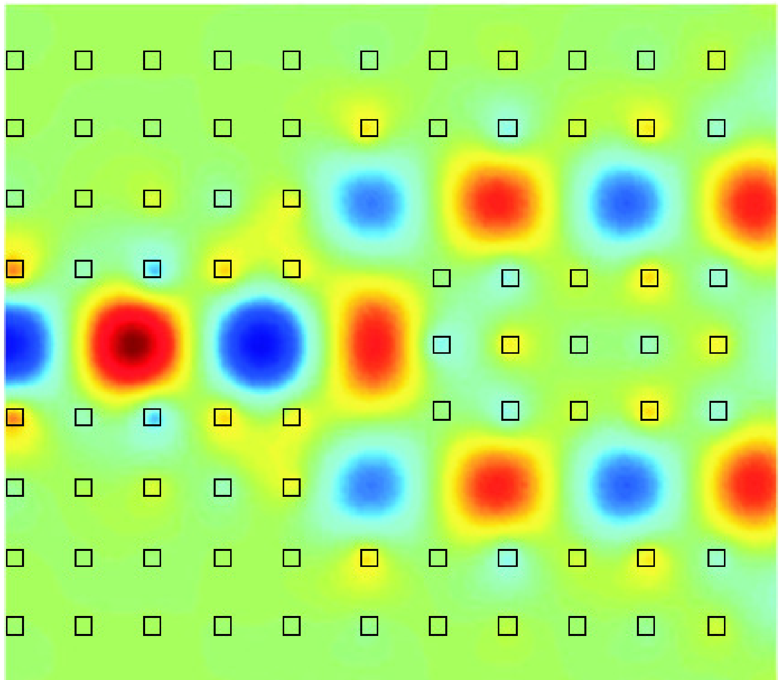
Can see heightened sensitivity to details of cylindrical geometry, with multiple reflection pekas

Code: Photonic Crystal Splitter

```
…
set_lower_wall(slab_H_wall)
periods = 3 # periods above outer waveguide
sections = 1 # intermediate 90 deg sections
no_rods = Slab(air(a-r+(sections+1+periods)*a+cl))
cen = Slab(air(a-r)+(sections+1+periods)*(GaAs(2*r) +
air(a-2*r))+air(cl) ) # Central waveguide
ver = Slab(air(a-r + (sections+1)*a) + periods*(GaAs(2*r)
+ air(a-2*r) )+air(cl) ) # Vertical section.
arm = Slab( GaAs(r) + air(a-2*r) + sections*(GaAs(2*r) +
air(a-2*r))+air(a)+periods*(GaAs(2*r) + air(a-
2*r))+air(cl) ) # Outer arms.
wg = BlochStack(cen(2*r) + no_rods(a-2*r))
wg.calc() # Find lowest order waveguide mode.
```



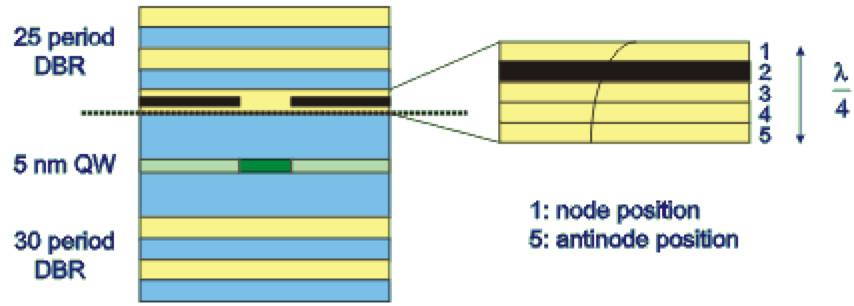
Results: Photonic Crystal Splitter



P. Bienstman, “Rigorous and efficient modeling of wavelength scale photonic components,” Ph.D. Thesis, University of Ghent (2001).

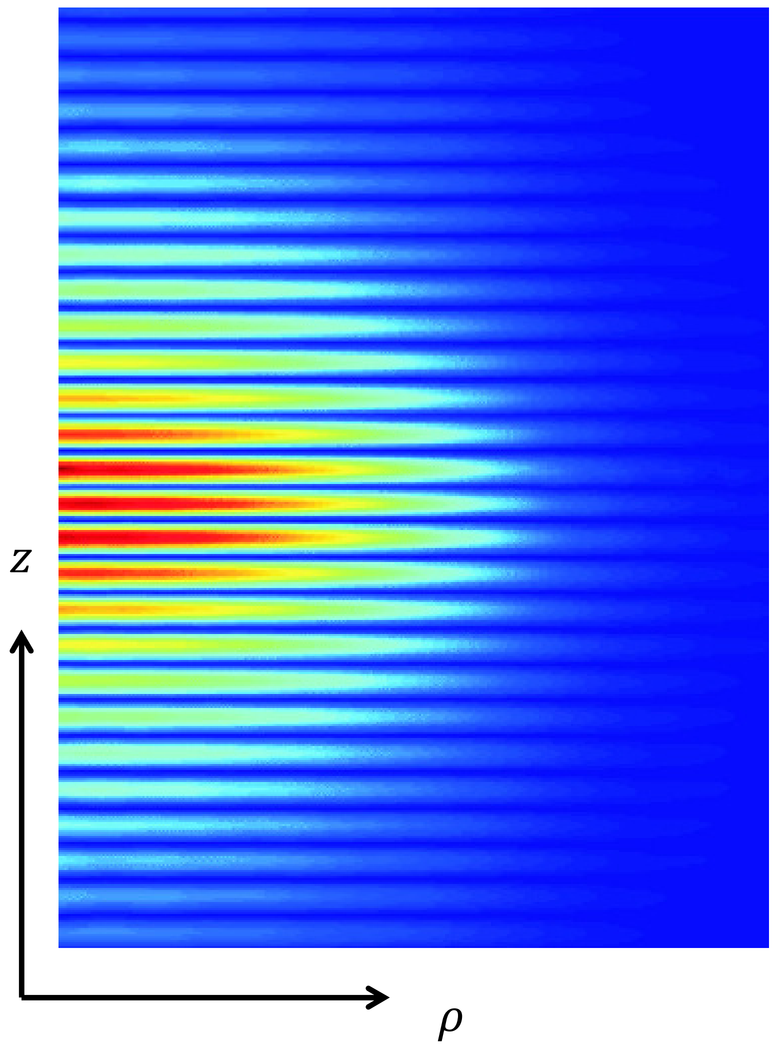
- Can demonstrate a low loss (3 dB) split within wavelength scale – compares favorably with index-guided fibers

Code: Vertical Cavity Surface Emitting Lasers (VCSELs)



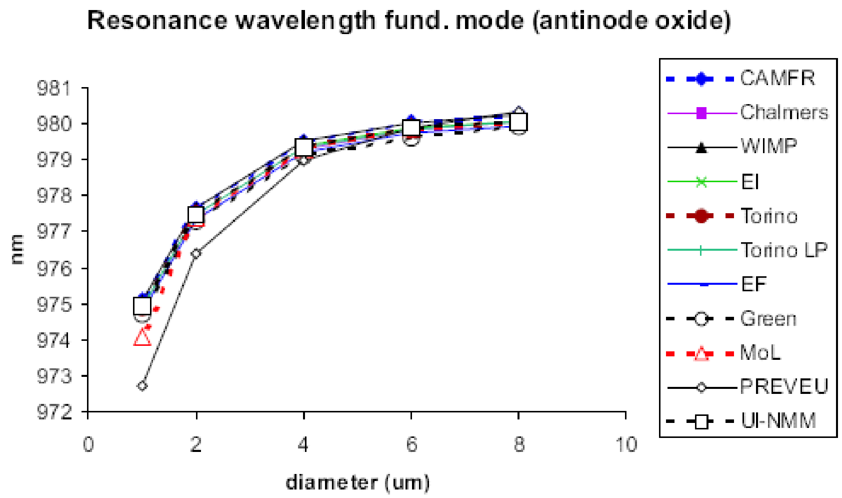

```
set_N(100)
set_circ_order(1)
set_circ_PML(-0.1)
GaAs=Circ (GaAs_m(r+d_cladding))
AlGaAs=Circ(AlGaAs_m(r+d_cladding))
…
top = Stack( (GaAs(0) + AlGaAs(x)) + ox(.2*d_AlGaAs) +
(AlGaAs(.8*d_AlGaAs - x) + GaAs(d_GaAs) +
24*(AlGaAs(d_AlGaAs) + GaAs(d_GaAs)) + air(0)) )
bottom = Stack(GaAs(.13659) + QW(.00500) \ + (GaAs(.13659)
+ 30*(AlGaAs(d_AlGaAs) + GaAs(d_GaAs) + GaAs(0))) )
cavity = Cavity(bottom, top)
cavity.find_mode(.980, .981)
```

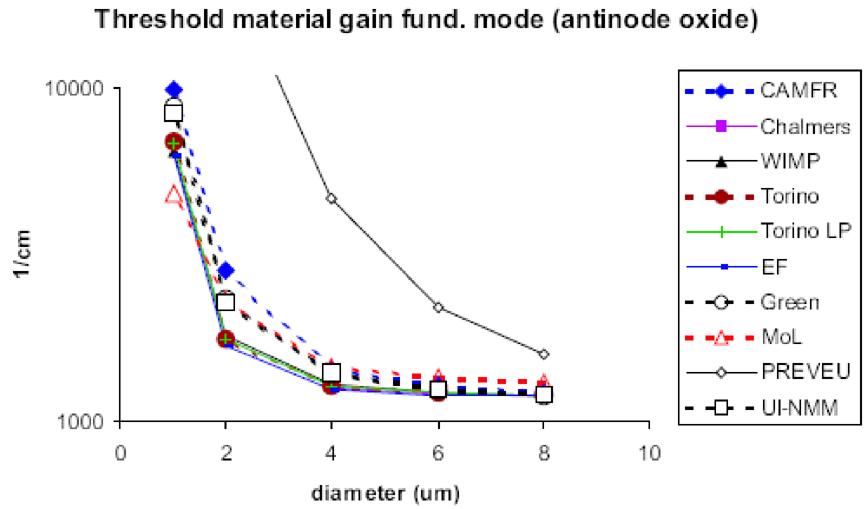
Results: VCSEL



- Field profile resulting from this design (in ρ-z plane)

Results: VCSEL





Next Class

- Is on Friday, March 29
- Next time: we will discuss Coupled Mode Theory (Joannopoulos, Chapter 10)