ECE 595, Section 10
Numerical Simulations
Lecture 35: MEEP Tutorial

Prof. Peter Bermel

April 8, 2013

Recap from Wednesday & Friday

- Introduction to FDTD
- Special features of MEEP:
  - Perfectly matched layers
  - Subpixel averaging
  - Symmetry
  - Scheme (programmable) interface
- Periodic and randomly textured light-trapping structures
  - Overview
  - Experimental motivation
  - Computational setup
  - Simulated field evolution
  - Absorption spectra
- Front coatings
- Correlated random structures

Outline

- MEEP Interfaces
- MEEP Classes
- Tutorial examples:
  - Waveguide
  - Bent waveguide

MEEP Interfaces

- C/C++ interface
  - Original interface developed
  - Executable compiled with MEEP library
  - Provides tremendous flexibility and speed
- Scheme interface
  - Most widely used interface
  - Built on Scheme and Libctl to make simple problems easy and hard problems soluble
  - Scripted and interpreted at runtime
- nanoHUB GUI
  - Reduces complexity of input at expense of flexibility

MEEP Classes

- Lattice: defines the size of the computational cell, when used
- Material-type: basic materials (3 choices)
  - Medium
  - Perfect-metal
  - Material-function
- Geometric-object: basic structures
  - Block
  - Cylinder
  - Sphere / Ellipsoid
  - Cone

MEEP Classes (Cont'd)

- Symmetry (3 choices)
  - Mirror-symmetry
  - Rotate2-symmetry
  - Rotate4-symmetry
- Pml: key properties
  - Thickness
  - Direction

MEEP Classes (Cont'd)

- Source (3 types):
  - Gaussian-src
  - Continuous-src
  - Custom-src
- Flux-region: used to calculate

$$P(\omega) = \int d\boldsymbol{A} \cdot [\boldsymbol{E}^*(\omega) \times \boldsymbol{H}(\omega)]$$

- Flux-region properties include:
  - Center
  - Size
  - Direction
  - Weight

Run Functions

- Run-until
- Run-sources
- Run-sources+
- Stop-when-fields-decayed
- (run-k-point T k)

Selected Step functions

- Output-epsilon
- Output-efield-z
- Output-tot-pwr
- (output-png component h5topng-options)

Example: Index-Guided Waveguide

(set! geometry-lattice (make lattice (size 16 8 no-size)))

(set! geometry (list (make block (center 0 0) (size infinity 1 infinity) (material (make dielectric (epsilon 12))))))

(set! sources (list (make source (src (make continuous-src (frequency 0.15))) (component Ez) (center -7 0))))

(set! pml-layers (list (make pml (thickness 1.0))))

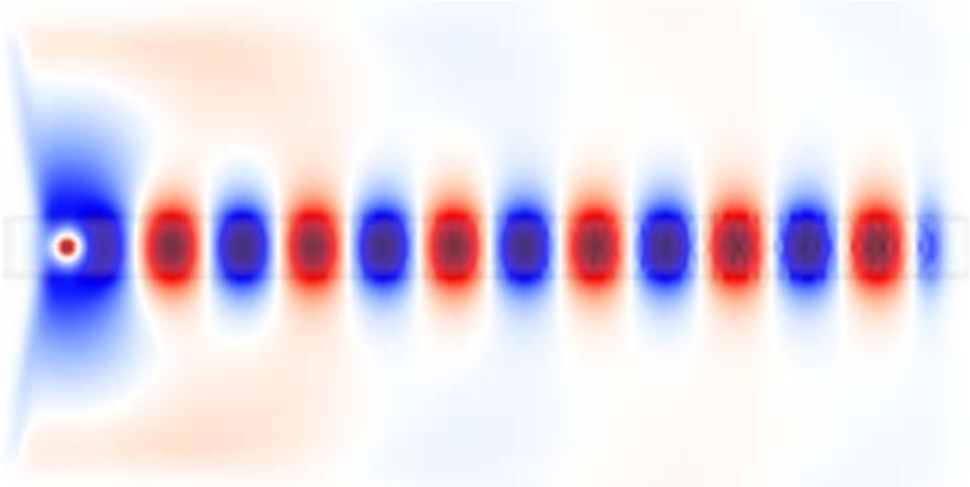
(set! resolution 10)

(run-until 200 (at-beginning output-epsilon) (at-end output-efield-z))

Example: Index-Guided Waveguide

- Dielectric function is as expected:

- Ez field propagates nicely along waveguide as expected:



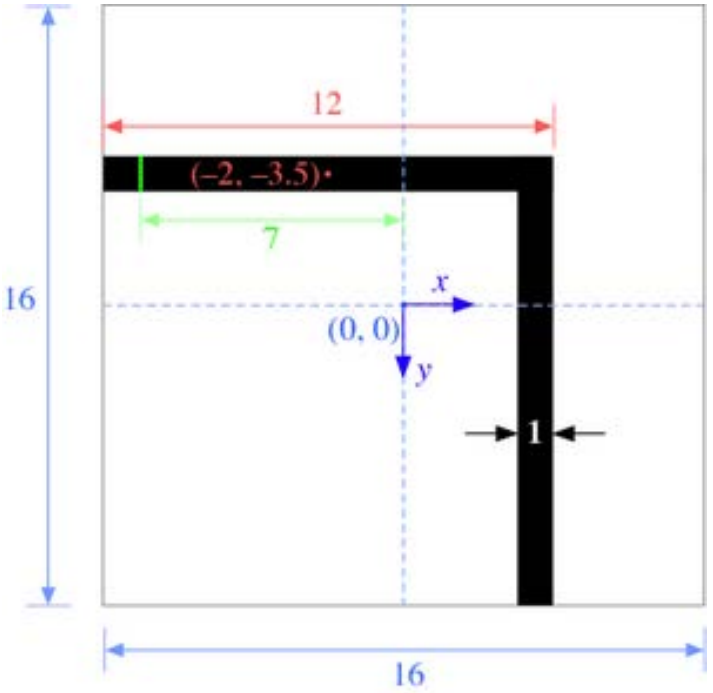
Example: Index-Guided Bend

(set! geometry-lattice (make lattice (size 16 16 no-size)))

(set! geometry (list (make block (center -2 -3.5) (size 12 1 infinity) (material (make dielectric (epsilon 12)))) (make block (center 3.5 2) (size 1 12 infinity) (material (make dielectric (epsilon 12))))))

(set! resolution 10)

- Resulting geometry (as shown on right):



Example: Index-Guided Bend

```
(set! pml-layers (list (make pml (thickness 1.0))))
(set! sources (list (make source (src (make continuous-src (wavelength (* 2 (sqrt 12))) (width 20))) (component Ez) (center -7 -3.5) (size 0 1))))
(run-until 200 (at-beginning output-epsilon) (to-appended "ez" (at-every 0.6 output-efield-z)))
```

Example: Index-Guided Bend

- Can create movie from this (as shown below):

Example: Index-Guided Bend

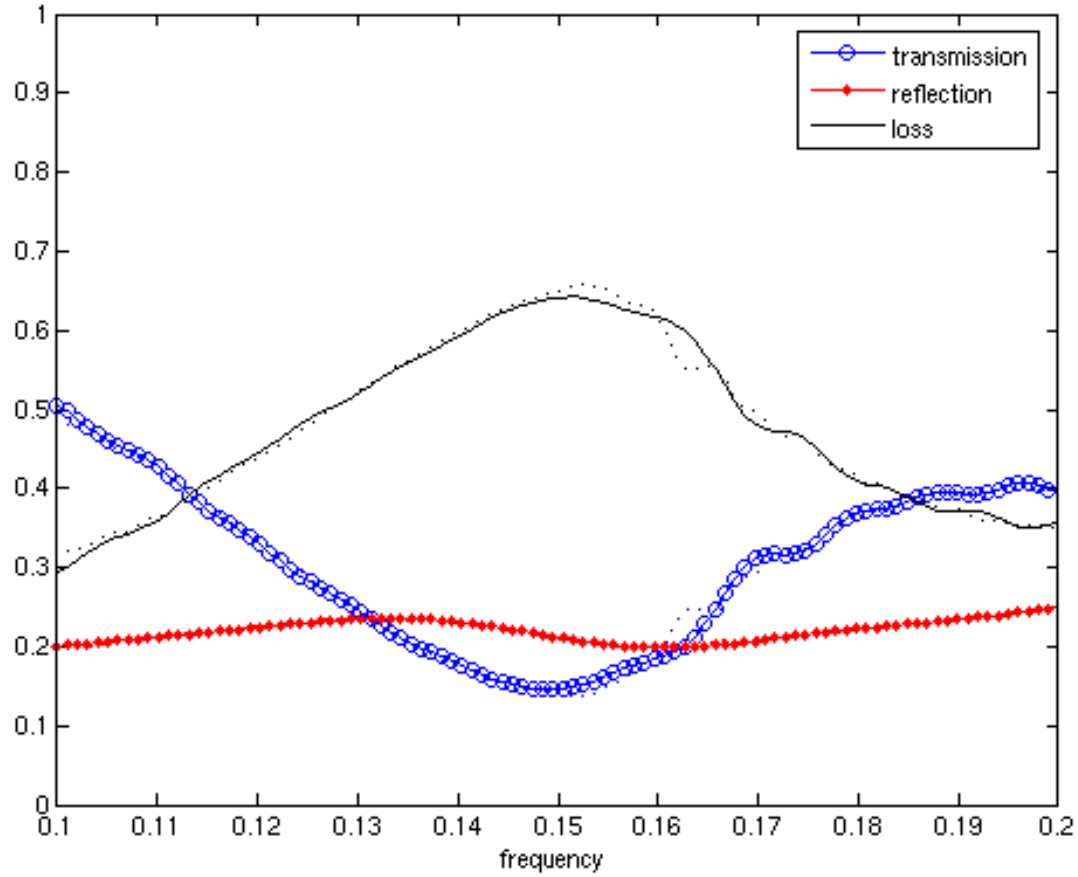
```
(define-param no-bend? false)
(set! geometry (if no-bend?
  (list (make block (center 0 wvg-ycen)
          (size infinity w infinity)
          (material (make dielectric (epsilon 12)))))
  (list (make block (center (* -0.5 pad) wvg-ycen)
          (size (- sx pad) w infinity)
          (material (make dielectric (epsilon 12))))
         (make block (center wvg-xcen (* 0.5 pad))
          (size w (- sy pad) infinity)
          (material (make dielectric (epsilon 12)))))))
```

Example: Index-Guided Bend

```
(define-param nfreq 100)
(define trans ; transmitted flux
  (add-flux fcen df nfreq
     (if no-bend?
       (make flux-region
         (center (- (/ sx 2) 1.5) wvg-ycen) (size 0 (* w 2)))
       (make flux-region
         (center wvg-xcen (- (/ sy 2) 1.5)) (size (* w 2) 0)))))
(define refl ; reflected flux
  (add-flux fcen df nfreq
      (make flux-region
        (center (+ (* -0.5 sx) 1.5) wvg-ycen) (size 0 (* w 2)))))
```

Example: Index-Guided Bend

Transmission, reflection, and loss spectrum for the bend



Next Class

- Is on Wednesday, April 10
- Next time: we will discuss using finite-difference time domain software: MEEP
- Suggested reference: MEEP tutorial, http://jdj.mit.edu/wiki/index.php/Meep_Tutorial