Computational Photonics:

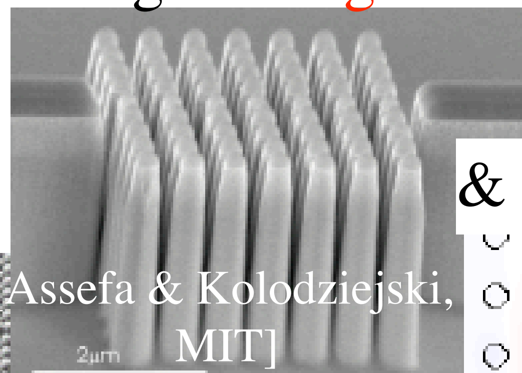
Frequency and Time Domain Methods

Steven G. Johnson

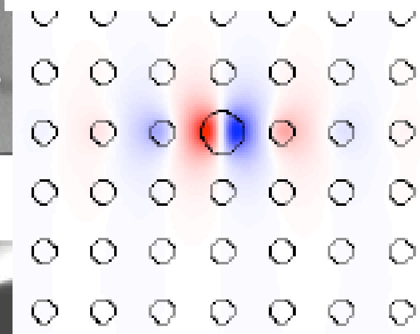
MIT Applied Mathematics

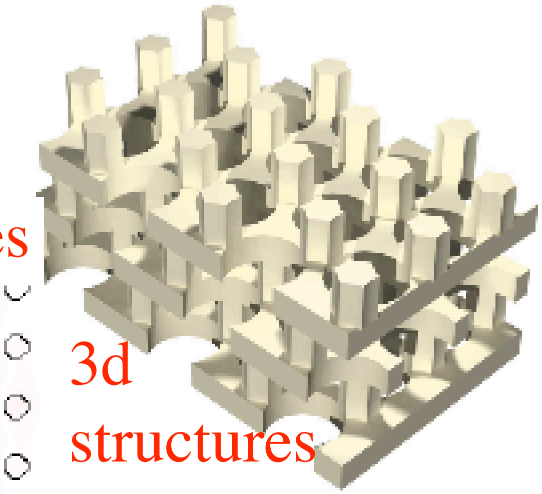
Nano-photonic media (λ-scale)

strange waveguides

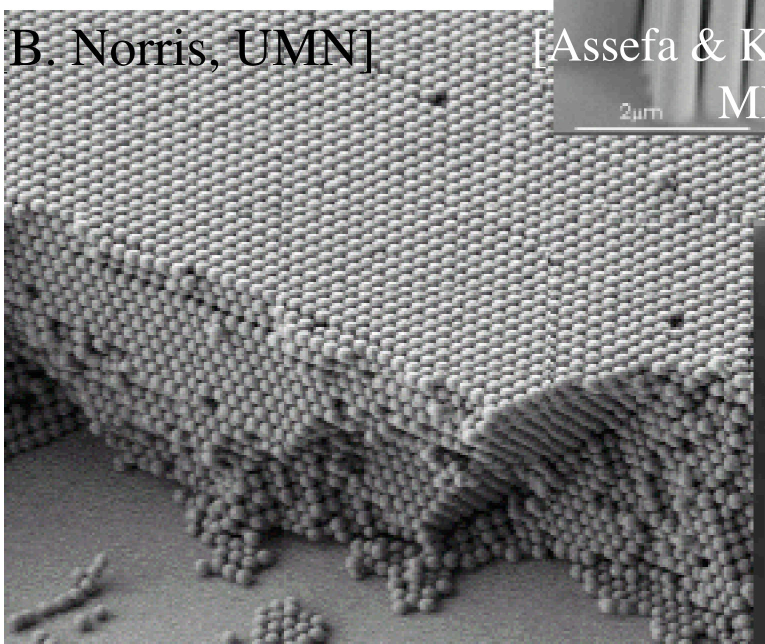


& microcavities

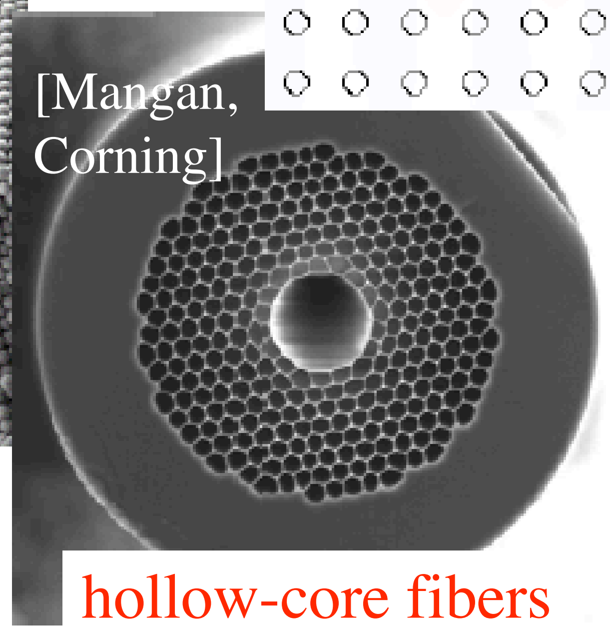




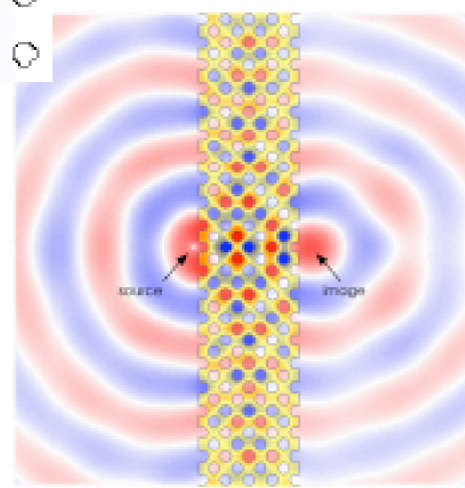
3d structures



synthetic materials



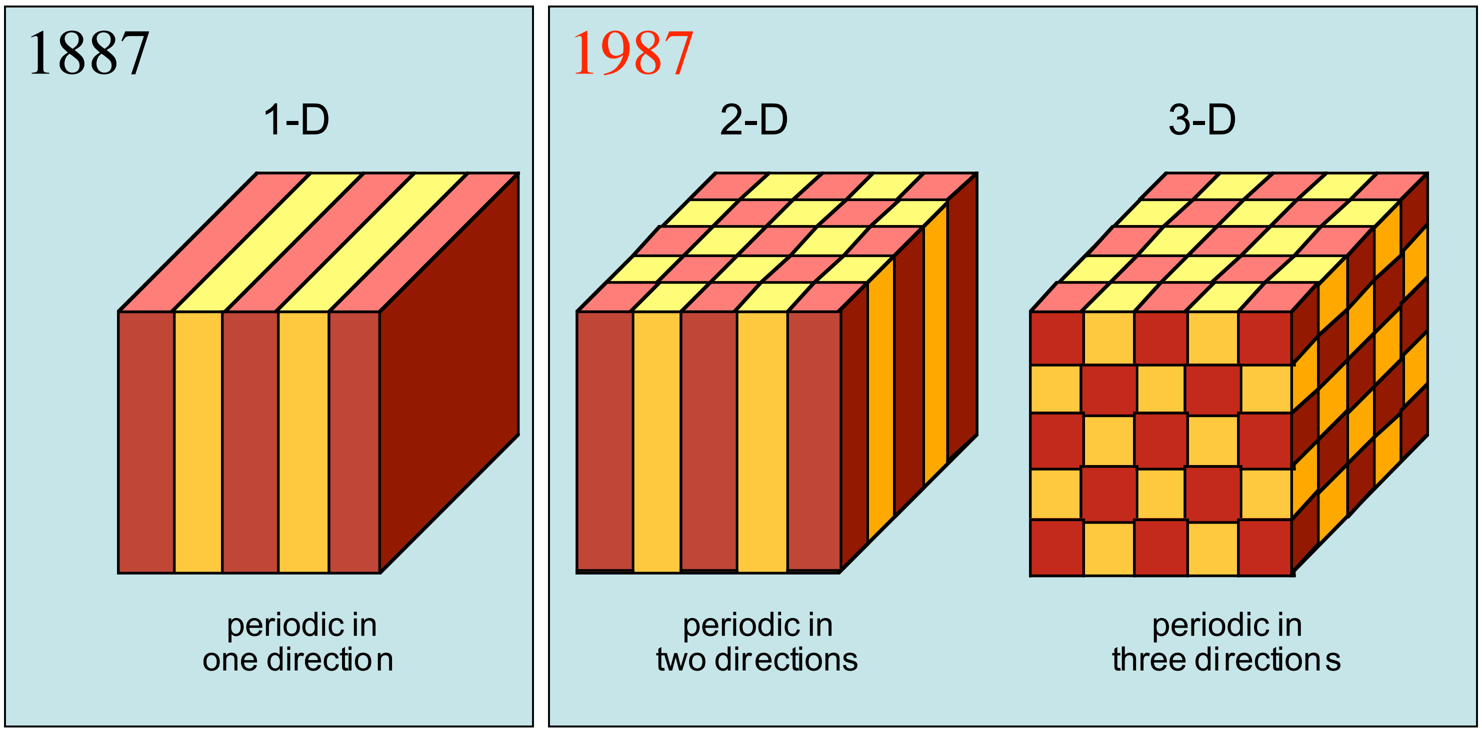
hollow-core fibers



optical phenomena

Photonic Crystals

periodic electromagnetic media

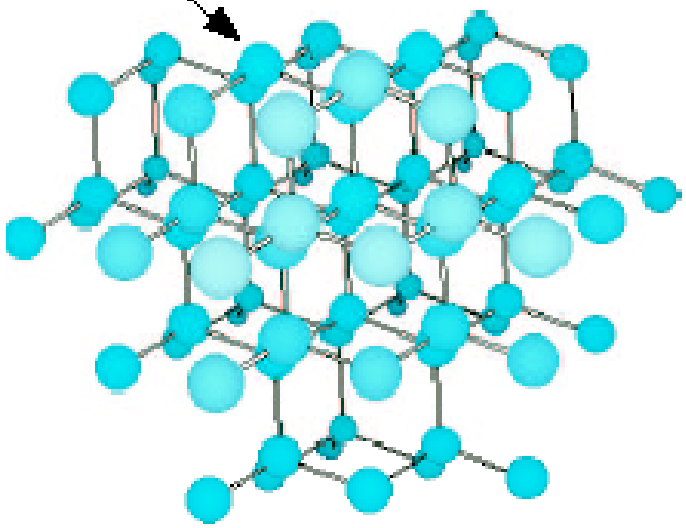


can have a band gap: optical “insulators”

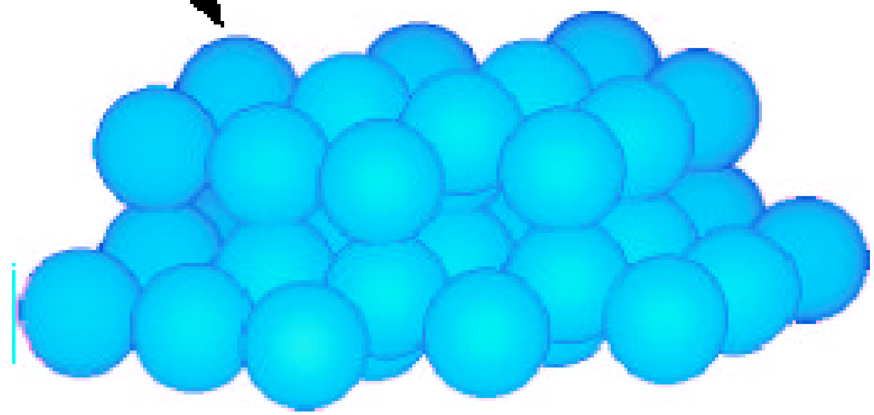
Electronic and Photonic Crystals

Periodic Medium

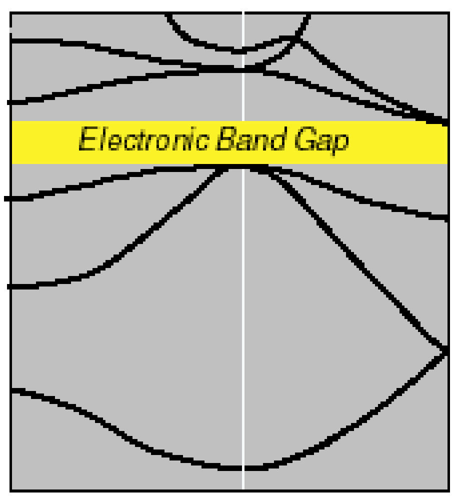
atoms in diamond structure

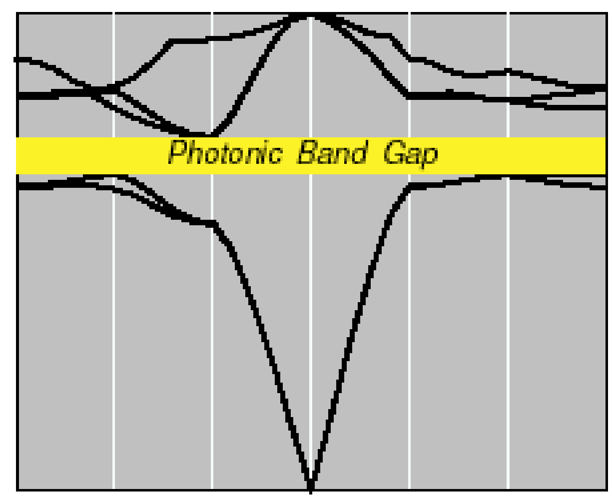


dielectric spheres, diamond lattice



Bloch waves: **Band Diagram**





Electronic & Photonic Modeling

Electronic

- strongly interacting
  - entanglement, Coulomb
  - tricky approximations
- lengthscale dependent (from Planck's *h*)

Photonic

- non-interacting (or weakly),
  - simple approximations (finite resolution)
  - any desired accuracy
- scale-invariant
  - *e.g.* size □10 □ □□10 (except materials may change)

Computational Photonics Problems

- Time-domain simulation
  - — start with current **J**(**x**,*t*)
  - — run "numerical experiment" to simulate **E**(**x**, *t*), **H**(**x**, *t*)

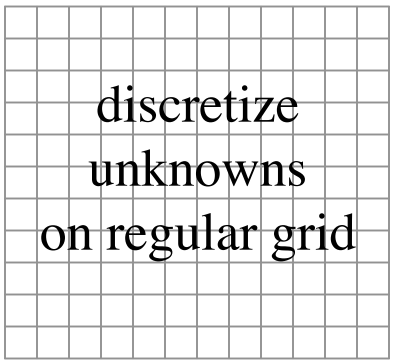
- Frequency-domain linear response
  - — start with harmonic current **J**(**x**, *t*) = $e^{-i\omega t}$ **J**(**x**)
  - — solve for steady-state harmonic fields **E**(**x**), **H**(**x**)
  - — involves solving linear equation Ax=b

- Frequency-domain eigensolver
  - — solve for source-free harmonic eigenfields

    E(**x**), **H**(**x**) ~ $e^{-i\omega t}$
  - — involves solving eigenequation $Ax=\omega^2 x$

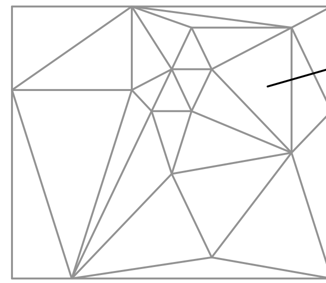
Numerical Methods: Basis Choices

finite difference



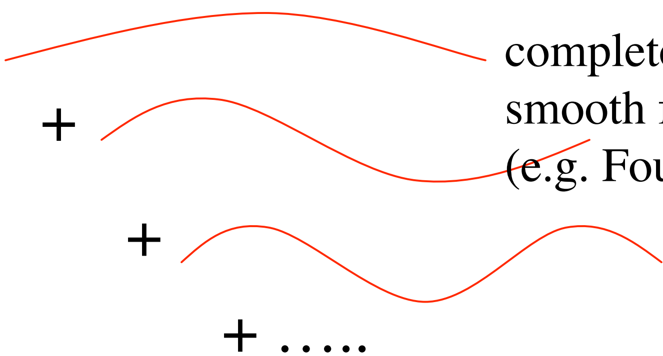
$$\frac{df}{dx} \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{\Delta x} + O(\Delta x^2)$$

finite elements



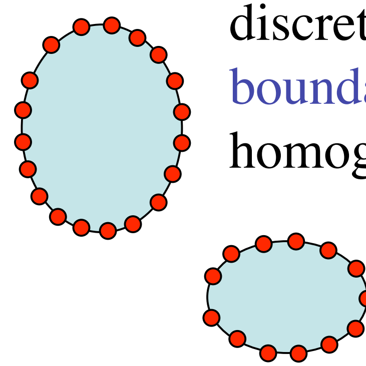
in irregular "elements," approximate unknowns by low-degree polynomial

spectral methods



complete basis of smooth functions (e.g. Fourier series)

boundary-element methods

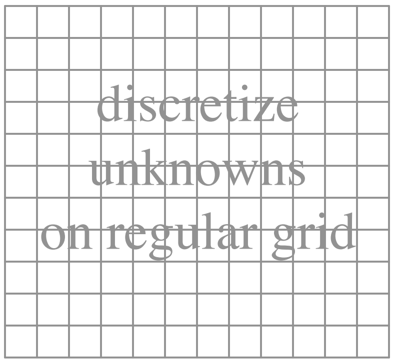


discretize only the boundaries between homogeneous media

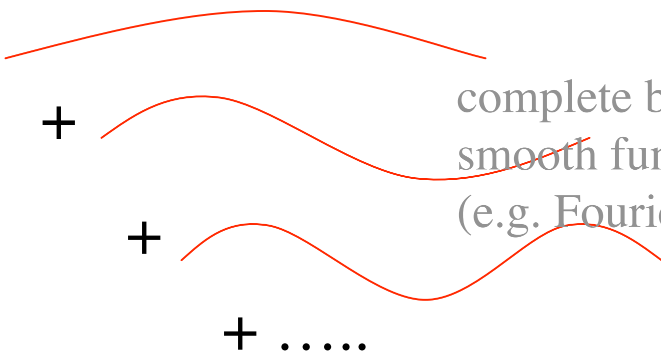
…solve integral equation via Green's functions

Numerical Methods: Basis Choices

finite difference



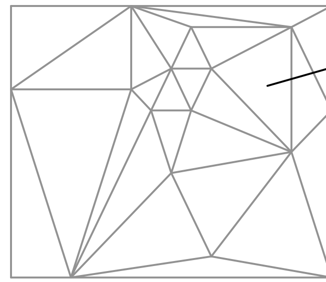
spectral methods



complete basis of
smooth functions
(e.g. Fourier series)

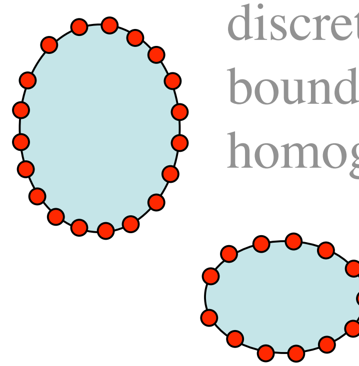
Much easier to analyze, implement, generalize, parallelize, optimize, …

finite elements



in irregular “elements,”
approximate unknowns
by low-degree polynomial

boundary-element methods



discretize only the
boundaries between
homogeneous media

…solve
integral equation
via Green’s functions

Potentially much more efficient, especially for high resolution

Computational Photonics Problems

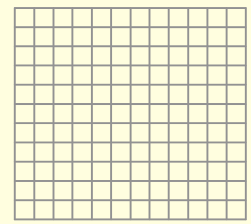
- Time-domain simulation
  - — start with current $\mathbf{J}(\mathbf{x},t)$
  - — run "numerical experiment" to simulate $\mathbf{E}(\mathbf{x}, t)$, $\mathbf{H}(\mathbf{x}, t)$

- Frequency-domain linear response
  - — start with harmonic current $\mathbf{J}(\mathbf{x}, t) = e^{-i\omega t}\, \mathbf{J}(\mathbf{x})$
  - — solve for steady-state harmonic fields $\mathbf{E}(\mathbf{x})$, $\mathbf{H}(\mathbf{x})$
  - — involves solving linear equation Ax=b

- Frequency-domain eigensolver
  - — solve for source-free harmonic eigenfields $\mathbf{E}(\mathbf{x})$, $\mathbf{H}(\mathbf{x}) \sim e^{-i\omega t}$
  - — involves solving eigenequation $Ax=\omega^2 x$

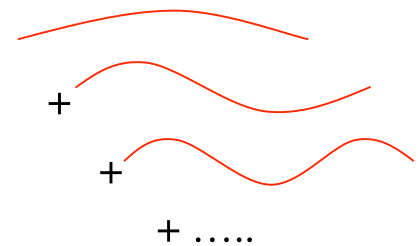
Numerical Methods: Basis Choices

finite difference

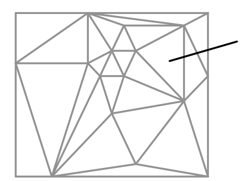


$$\frac{df}{dx} \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x} + O(\Delta x^2)$$

spectral methods

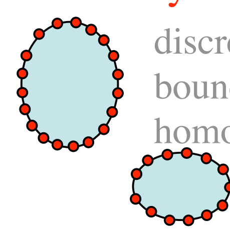


finite elements



in irregular "elements," approximate unknowns by low-degree polynomial

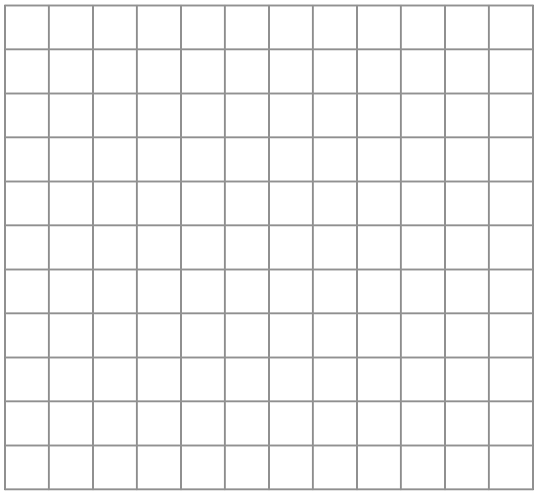
boundary-element methods



discretize only the boundaries between homogeneous media

FDTD

Finite-Difference Time-Domain methods



Divide *both* space and time into discrete grids

— spatial resolution Δx

— temporal resolution Δt

Very general: arbitrary geometries, materials, nonlinearities, dispersion, sources, …

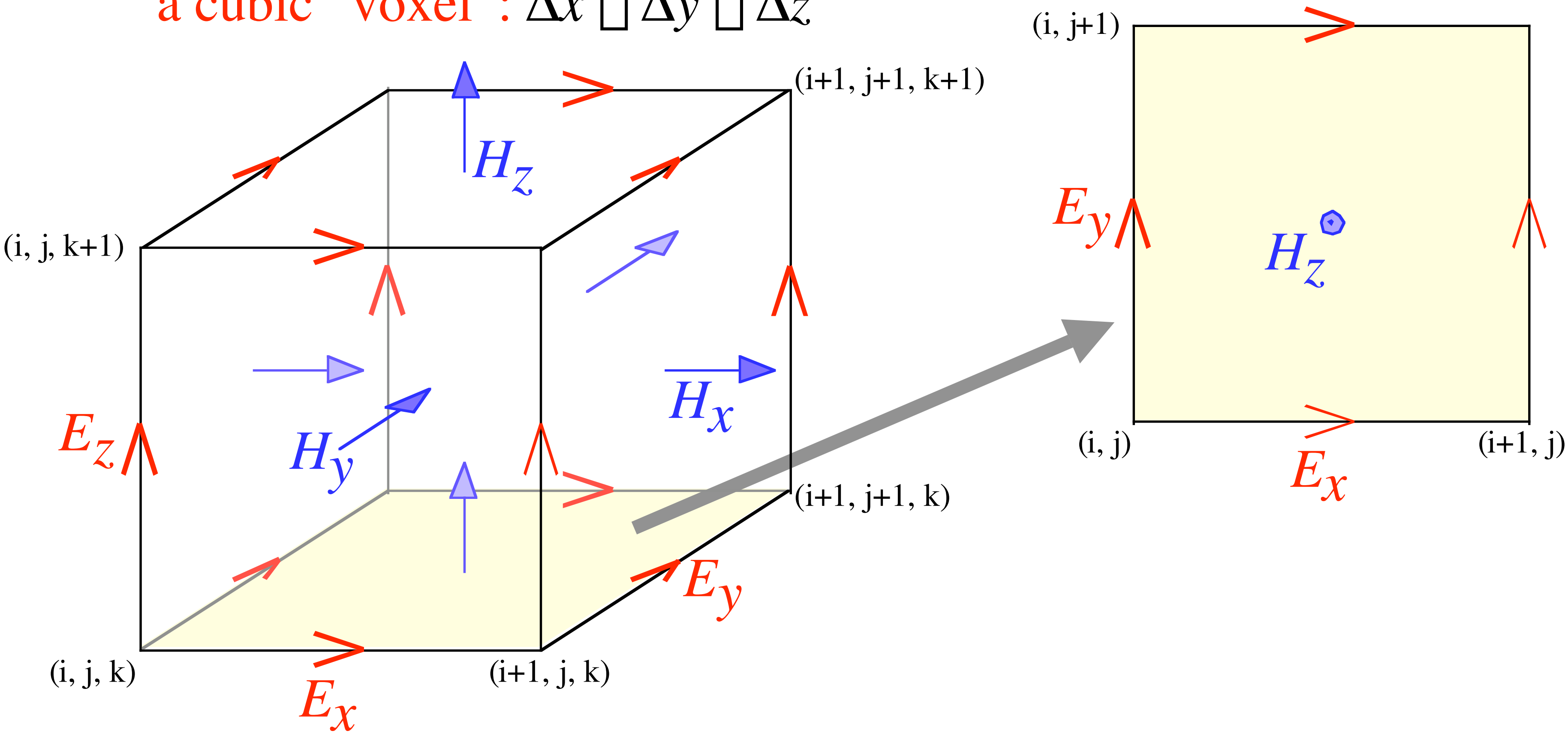
— *any photonics calculation, in principle*

$$\frac{\partial \mathbf{H}}{\partial t} = -\frac{1}{\mu}\vec{\nabla} \times \mathbf{E} \qquad \frac{\partial \mathbf{E}}{\partial t} = \frac{1}{\varepsilon}\vec{\nabla} \times \mathbf{H} - \frac{\mathbf{J}}{\varepsilon}$$

dielectric function $\varepsilon(\mathbf{x}) = n^2(\mathbf{x})$

The Yee Discretization (1966)

a cubic "voxel": Δx □ Δy □ Δz

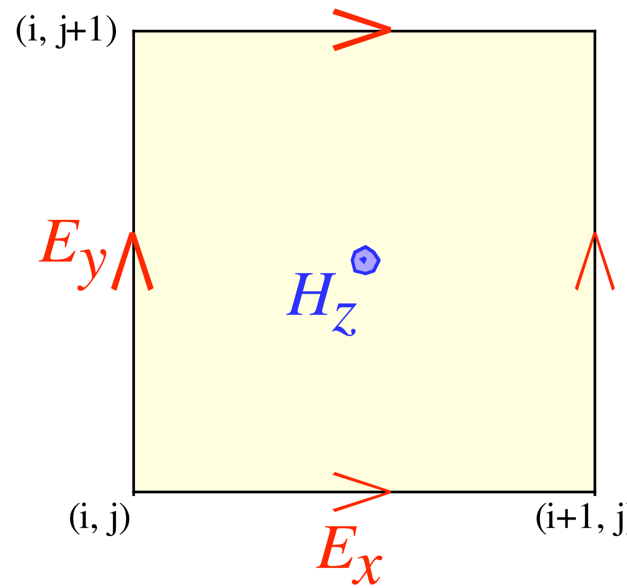


Staggered grid in space:

— every field component is stored on a different grid

The Yee Discretization (1966)

$$\frac{\partial \mathbf{H}}{\partial t} = -\frac{1}{\mu}\left(\nabla\times\mathbf{E}\right) \quad \cdots$$



$$\left.\frac{\partial H_z}{\partial t}\right|_{i+\frac{1}{2},\, j+\frac{1}{2}} = -\frac{1}{\mu}\left(\frac{\partial E_y}{\partial x} - \frac{\partial E_x}{\partial y}\right)$$

$$\approx -\frac{1}{\mu}\left[\frac{E_y(i+1, j+\frac{1}{2}) - E_y(i, j+\frac{1}{2})}{\Delta x} - \frac{E_x(i+\frac{1}{2}, j+1) - E_x(i+\frac{1}{2}, j)}{\Delta y}\right]$$

$$+\ \mathrm{O}(\Delta x^2) + \mathrm{O}(\Delta y^2)$$

all derivatives become *center differences*…

The Yee Discretization (1966)

all derivatives become *center differences*…

including derivatives in *time*

$$\left.\frac{\partial \mathbf{H}}{\partial t}\right|_{t=n\Delta t} = -\frac{1}{\mu} \nabla \times \left.\mathbf{E}\right|_{t=n\Delta t} \approx \frac{\mathbf{H}(n+\frac{1}{2}) - \mathbf{H}(n-\frac{1}{2})}{\Delta t}$$

$$+ O(\Delta t^2)$$

Explicit time-stepping:

stability requires $\Delta t < \frac{\Delta x}{\sqrt{\#\ \text{dimensions}}}$

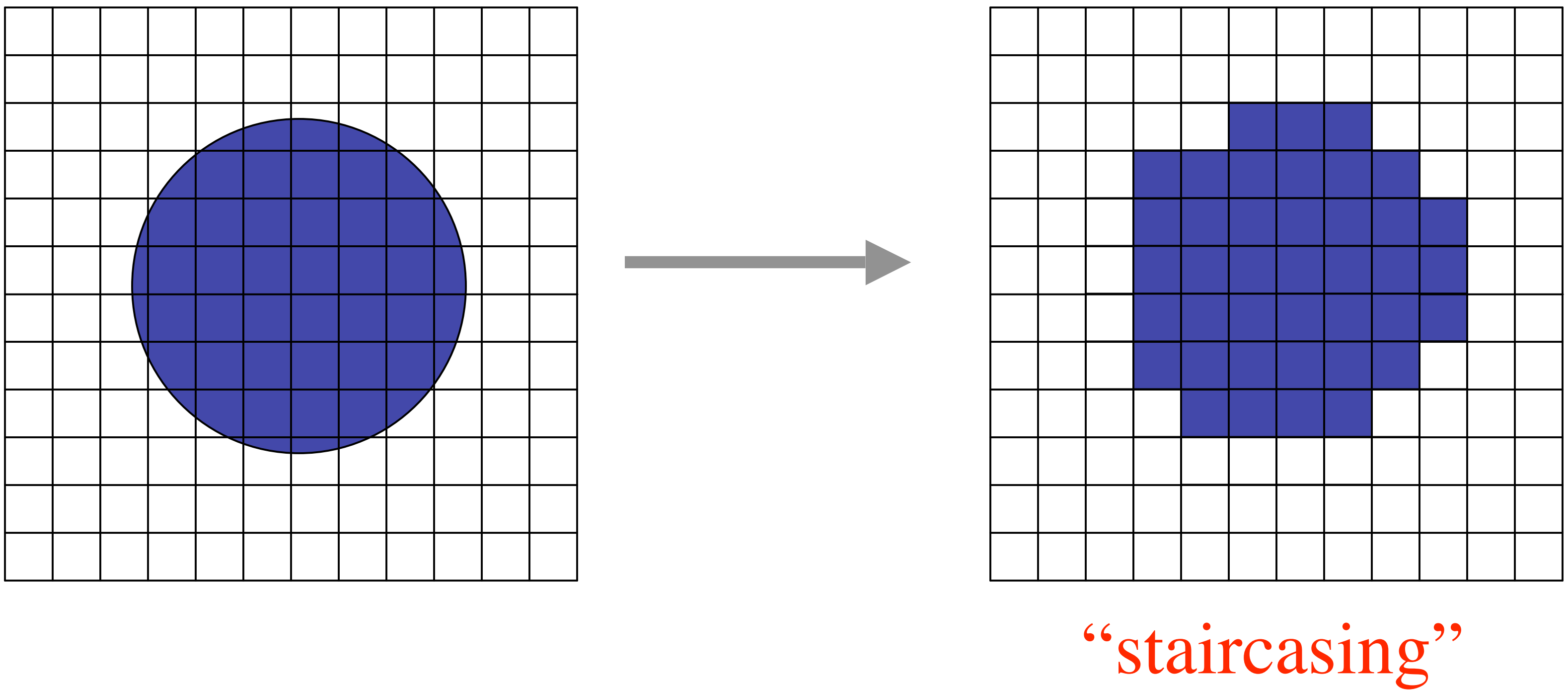
(vs. *implicit* time steps: invert large matrix at each step)

FDTD Discretization Upshot

- For stability, space and time resolutions are proportional
  — doubling resolution in 3d requires
  at least $16 = 2^4$ times the work!

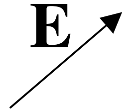
- But at least the error goes quadratically with resolution
  …right?
  …not necessarily!

Difficulty with a grid:
representing discontinuous materials?

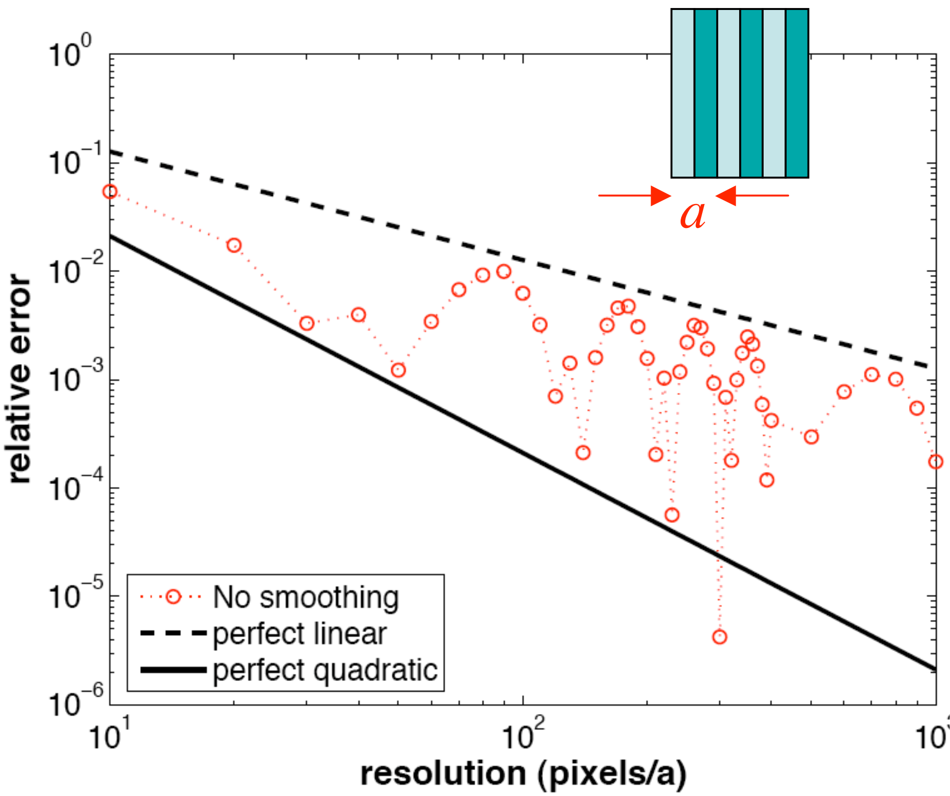


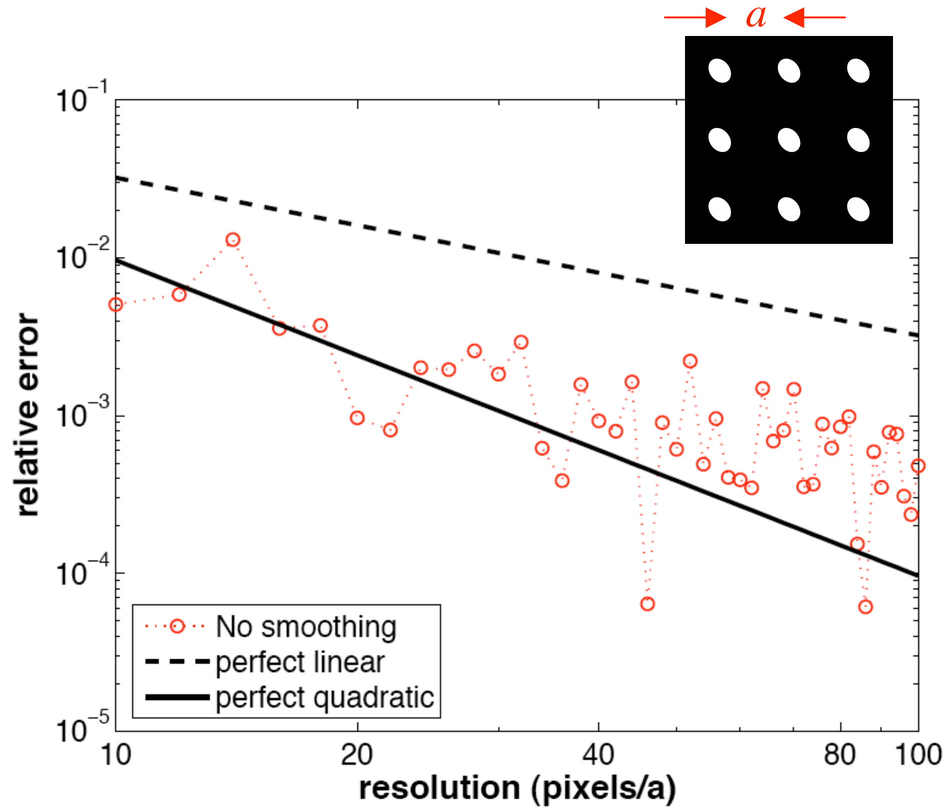
… how does this affect accuracy?

Field Discontinuity Degrades *Order* of Accuracy

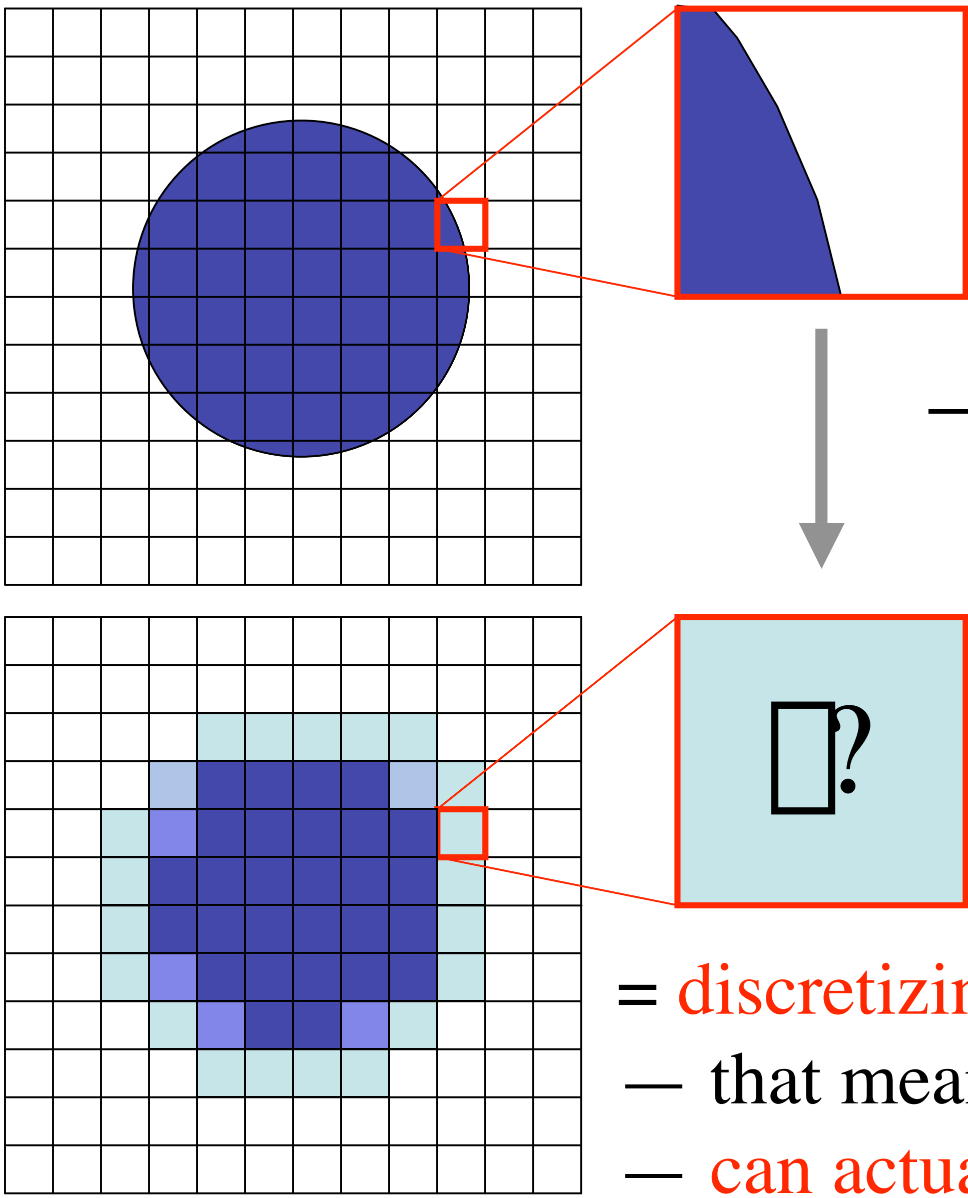


TE polarization (**E** in plane: discontinuous)





Sub-pixel smoothing



Can eliminate discontinuity by "grayscaling"

— assign some *average* □ to each pixel

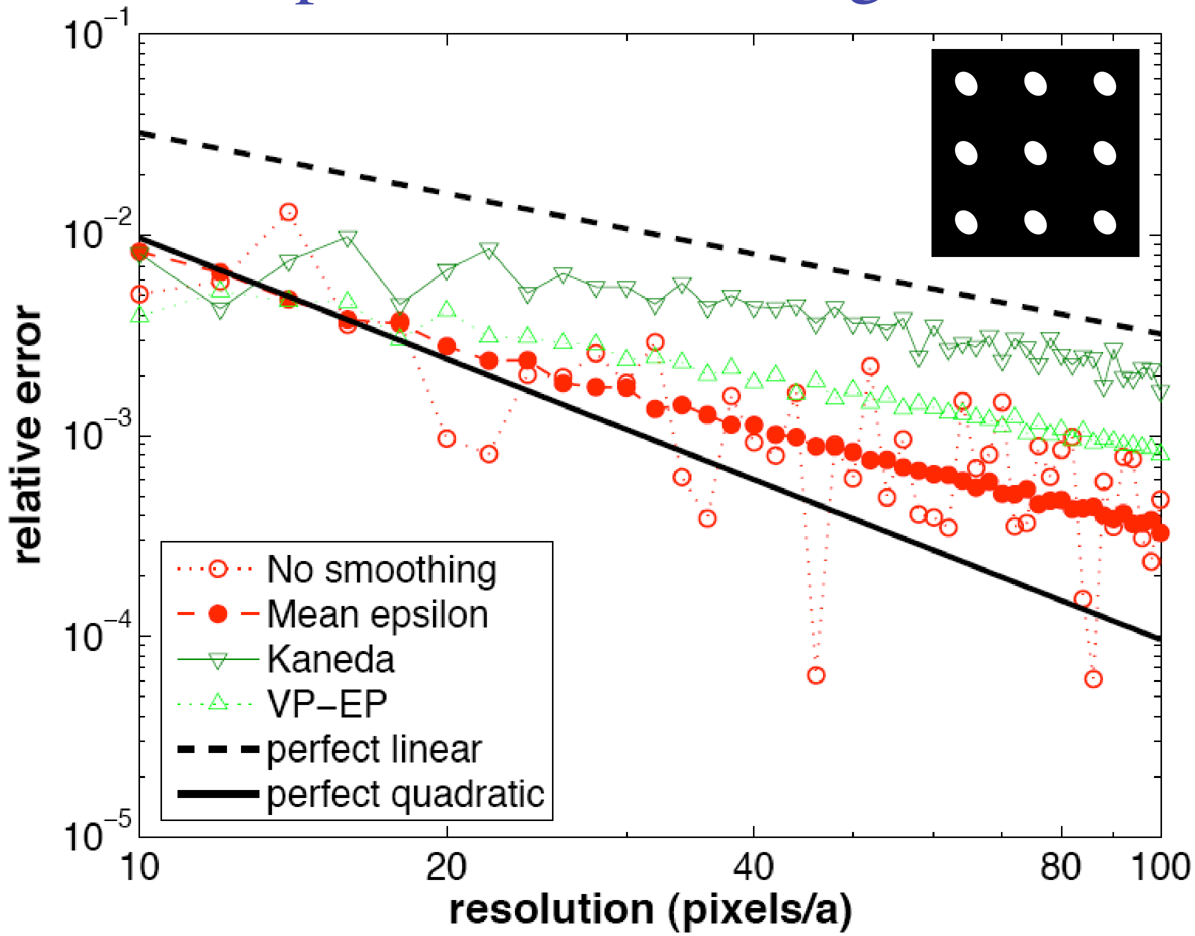
= discretizing a *smoothed* structure

— that means we are *changing* geometry

— can actually *add* to error

Past sub-pixel smoothing methods can make error *worse!*

Three previous smoothing methods



& convergence is still only linear

[Dey, 1999]
[Kaneda, 1997]
[Mohammadi, 2005]

A Criterion for Accurate Smoothing

1st-order errors from smoothing $\Delta\varepsilon$

$$\sim \int \left[\Delta\varepsilon \, |\mathbf{E}_{\|}|^2 - \Delta\left(\frac{1}{\varepsilon}\right) |D_{\perp}|^2 \right]$$

We want the smoothing errors to be *zero* to 1st order

— minimizes error *and* 2nd-order convergent!

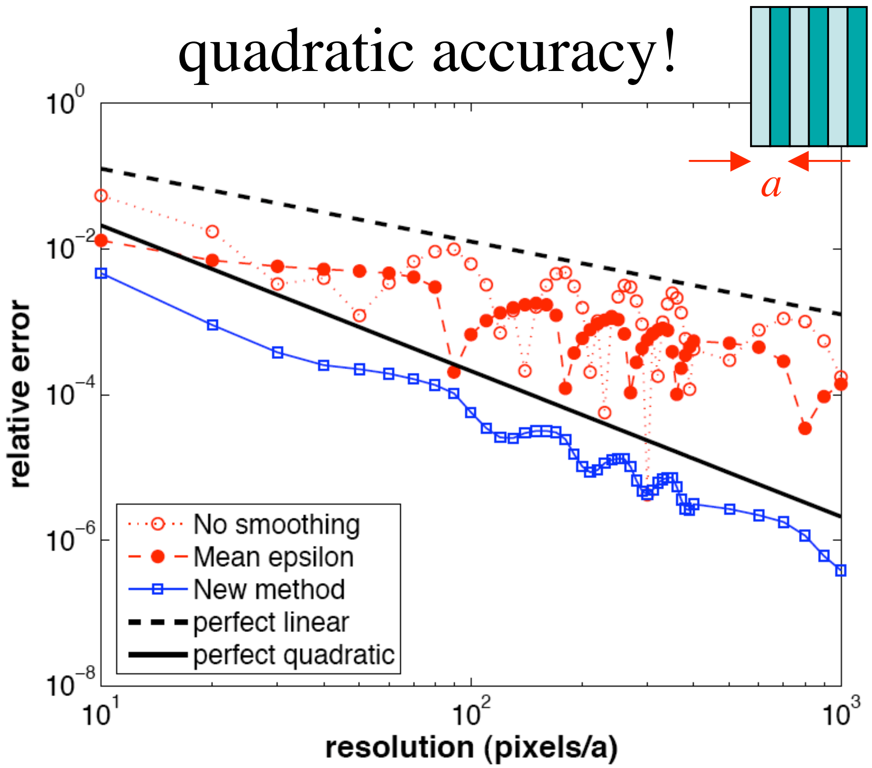
→ *Use a tensor* $\tilde{\varepsilon}$:

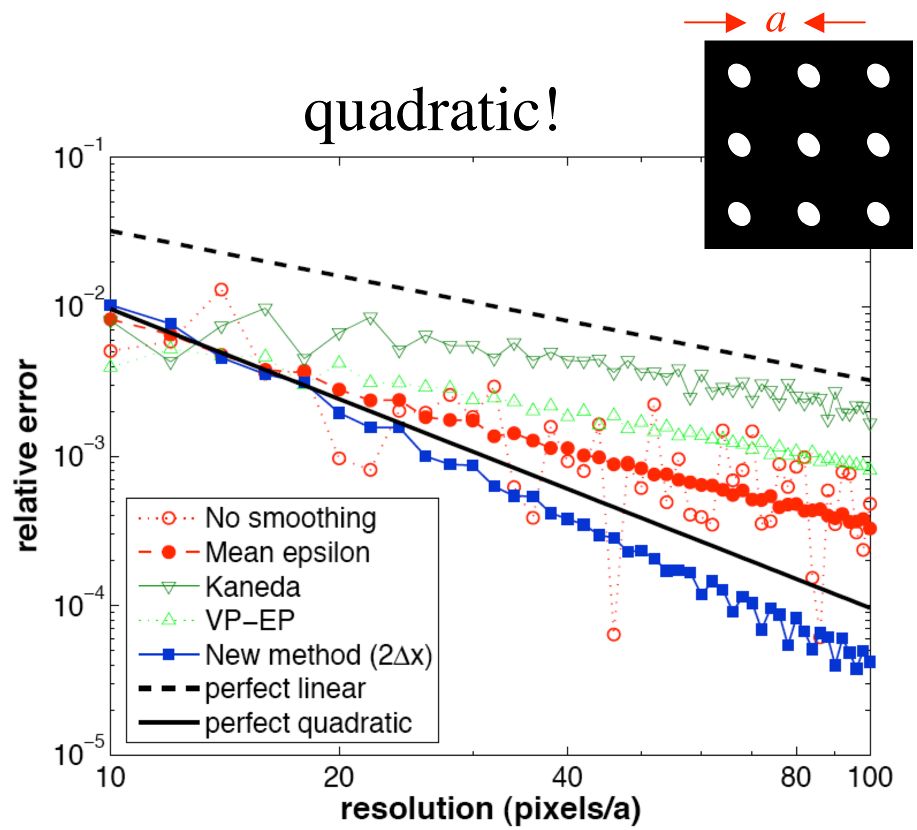
(in principal axes:)

$$\tilde{\varepsilon} = \begin{pmatrix} \langle\varepsilon\rangle & & \\ & \langle\varepsilon\rangle & \\ & & \langle\varepsilon^{-1}\rangle^{-1} \end{pmatrix} \begin{matrix} \mathbf{E}_{\|} \\ \\ \mathbf{E}_{\perp} \end{matrix}$$

[Meade et al., 1993]

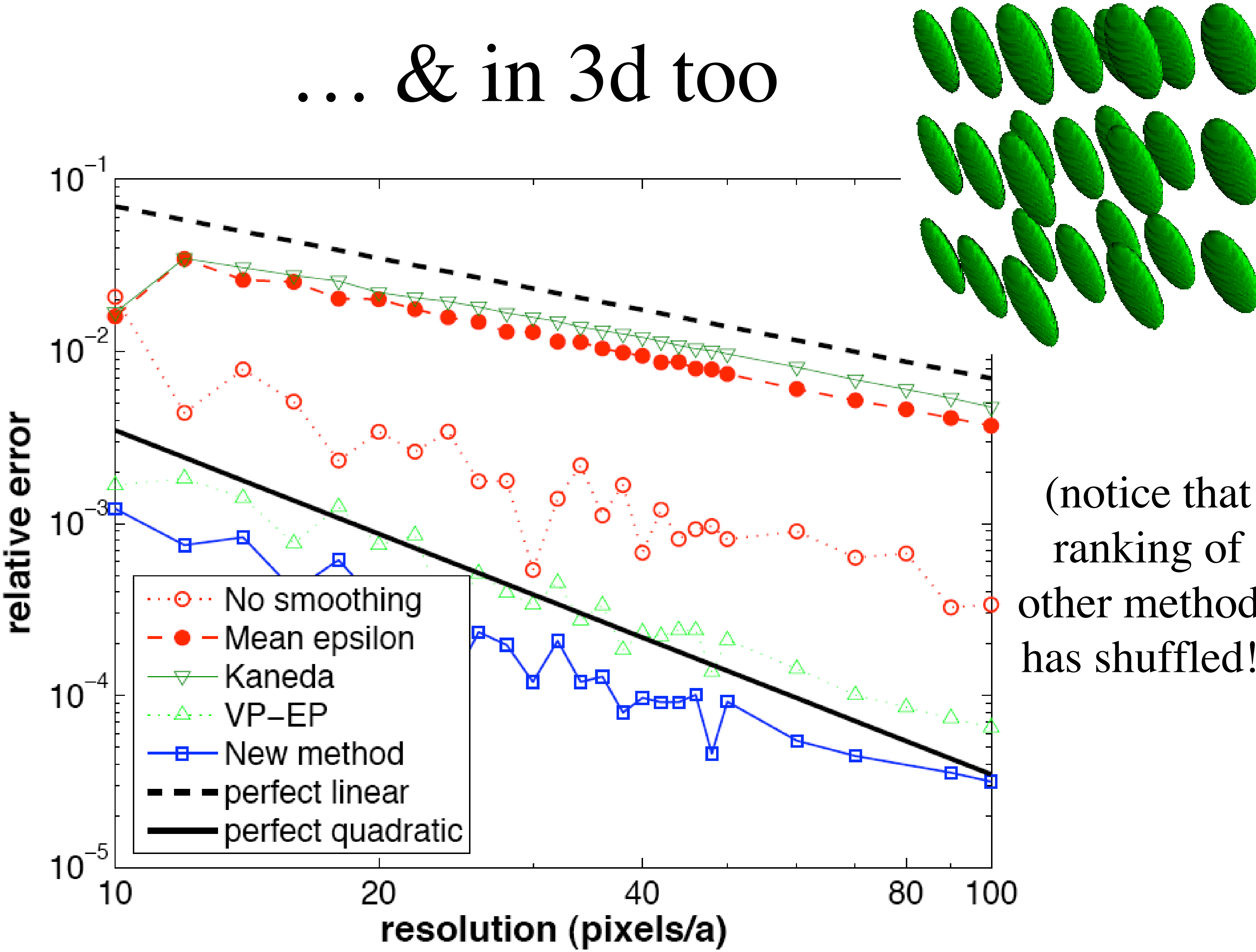
Consistently the Lowest Error





[Farjadpour *et al*., *Opt*. *Lett*. 2006]

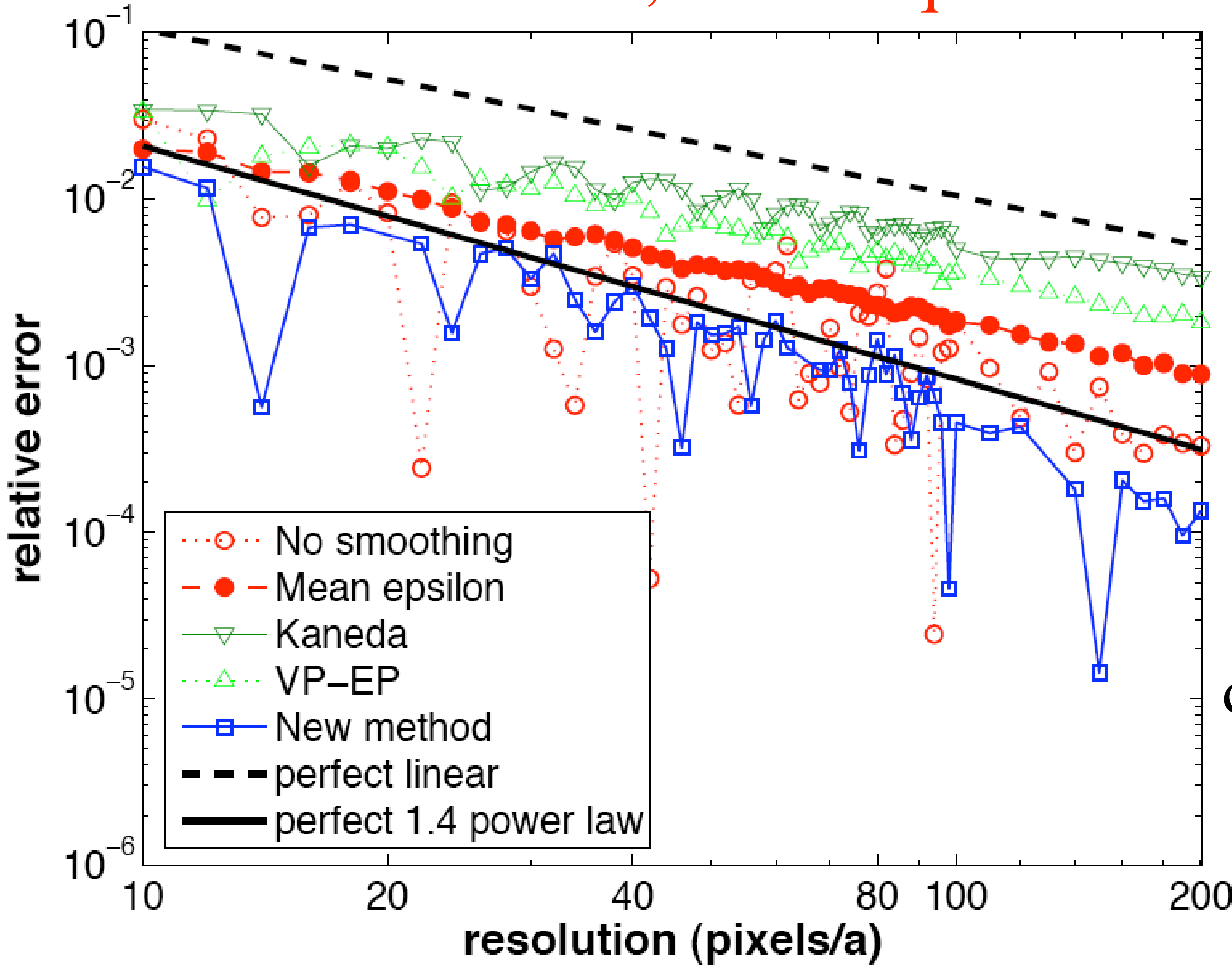
… & in 3d too

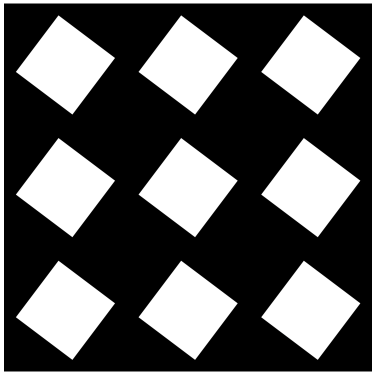


(notice that ranking of other methods has shuffled!)

A qualitatively different case: corners

still ~lowest error, but *not* quadratic





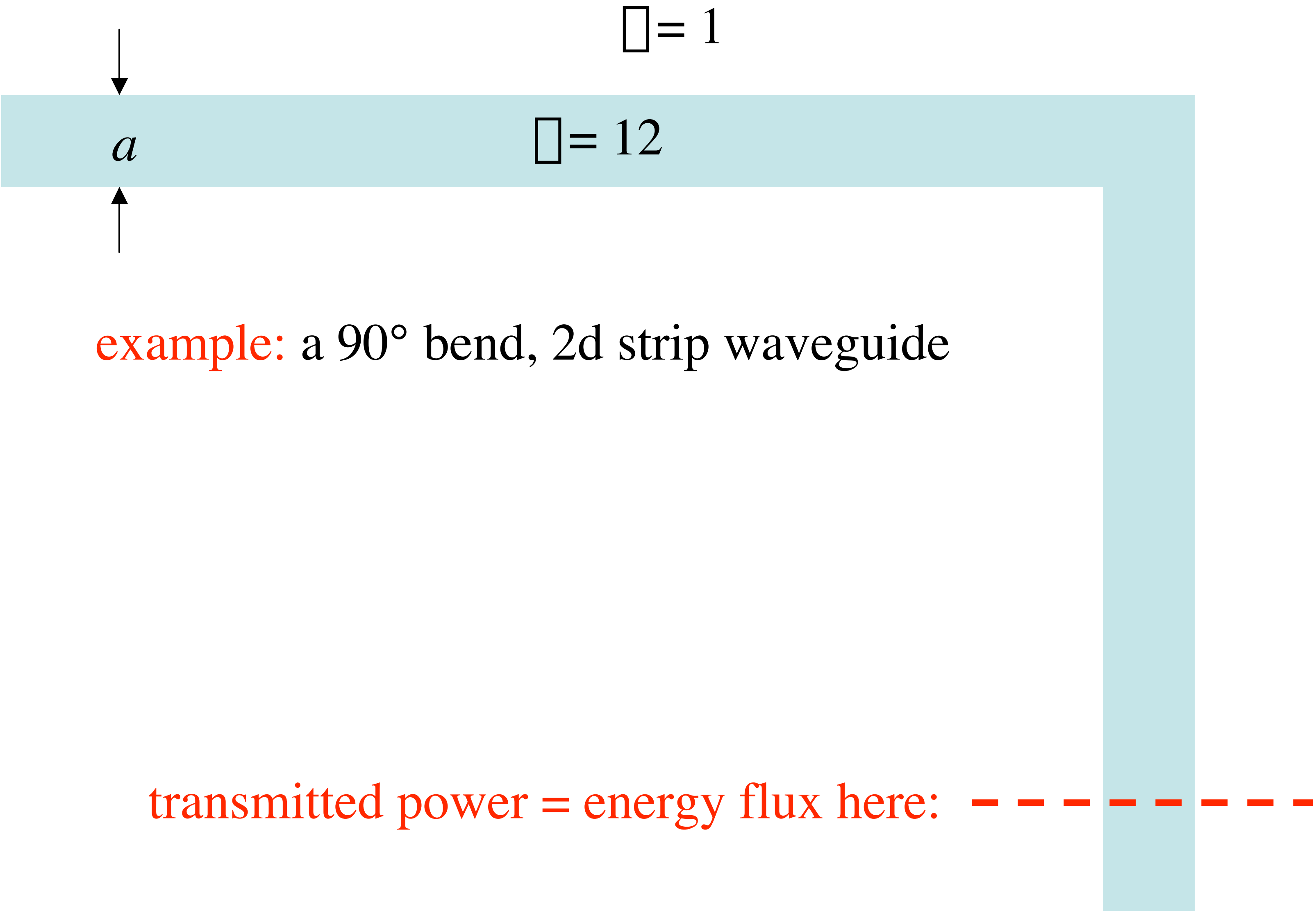
zero-perturbation criterion *not* satisfied due to **E** divergence at corner — analytically, error ~ $\Delta x^{1.404}$

Yes, but what can you *do* with FDTD?

Some common tasks:

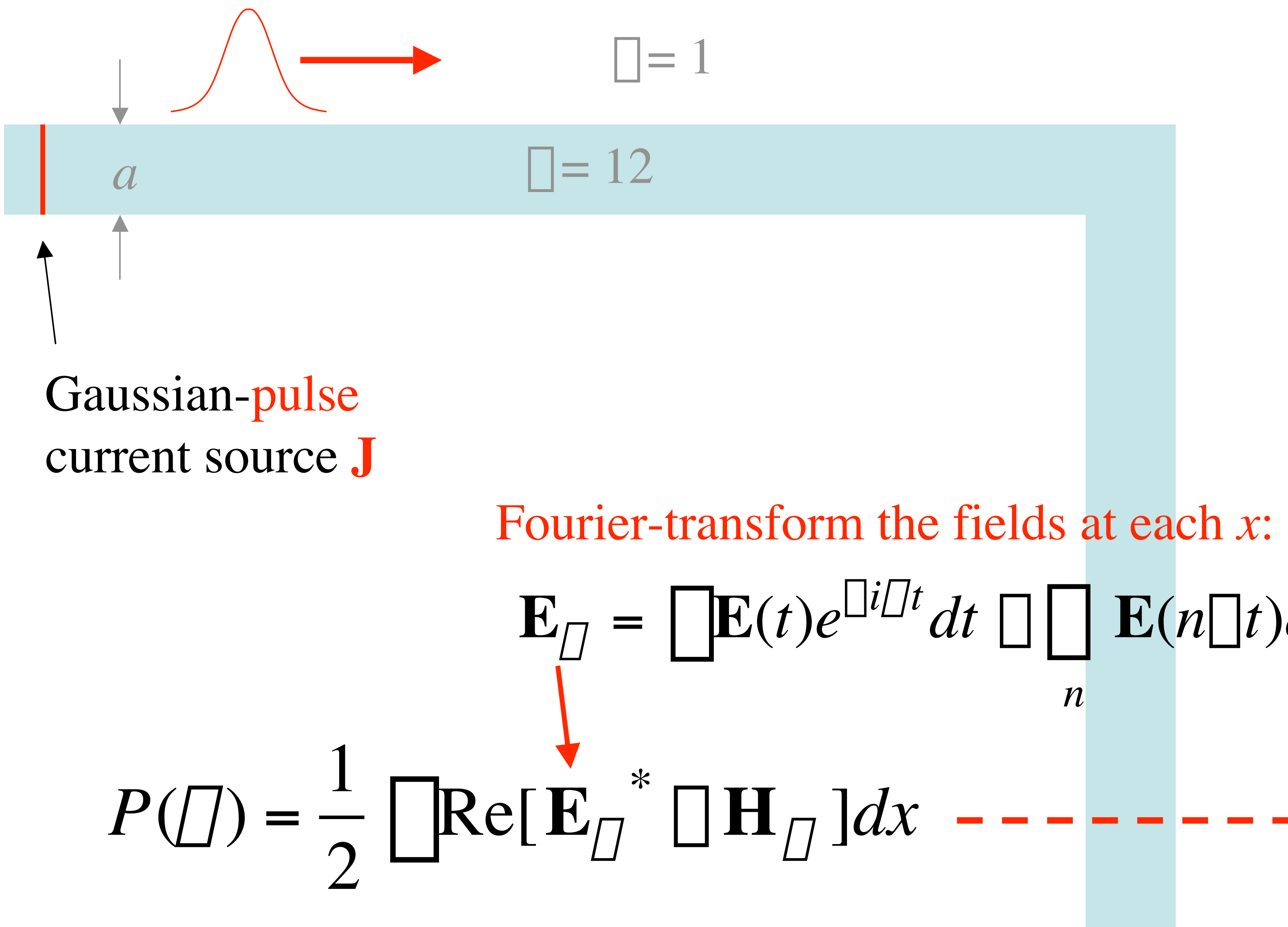
- Frequency-domain response:
  — put in harmonic source and wait for steady-state
- Transmission/reflection spectra:
  — get entire spectrum from a single simulation
  (Fourier transform of impulse response)
- Eigenmodes and resonant modes:
  — get all modes from a single simulation
  (some tricky signal processing)

Transmission Spectra in FDTD



example: a 90° bend, 2d strip waveguide

Transmission Spectra in FDTD



Fourier-transform the fields at each x:

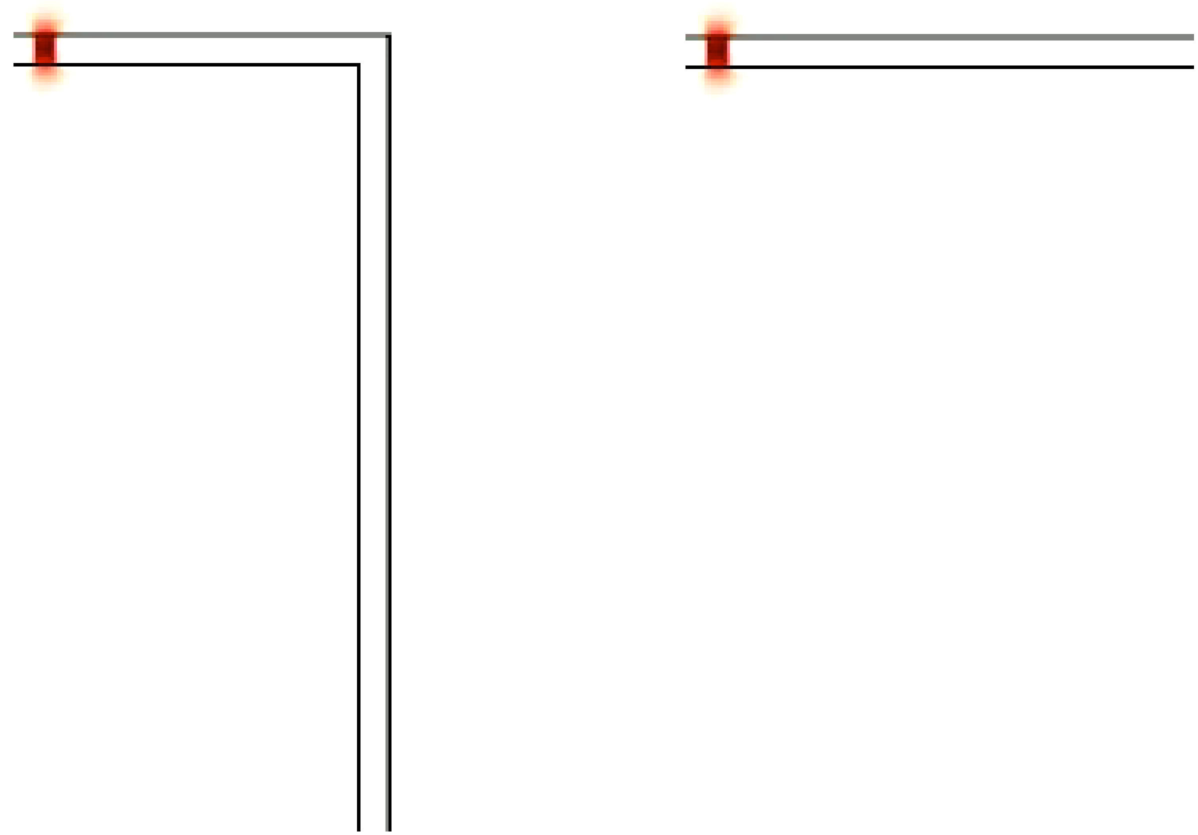
$$\mathbf{E}_{\omega} = \int \mathbf{E}(t) e^{-i\omega t} dt \approx \sum_{n} \mathbf{E}(n\Delta t) e^{-i\omega n\Delta t} \Delta t$$

$$P(\omega) = \frac{1}{2} \int \mathrm{Re}[\mathbf{E}_{\omega}^{*} \times \mathbf{H}_{\omega}] dx$$

Transmission Spectra in FDTD

must always do *two* simulations: one for normalization

electric field E_z:

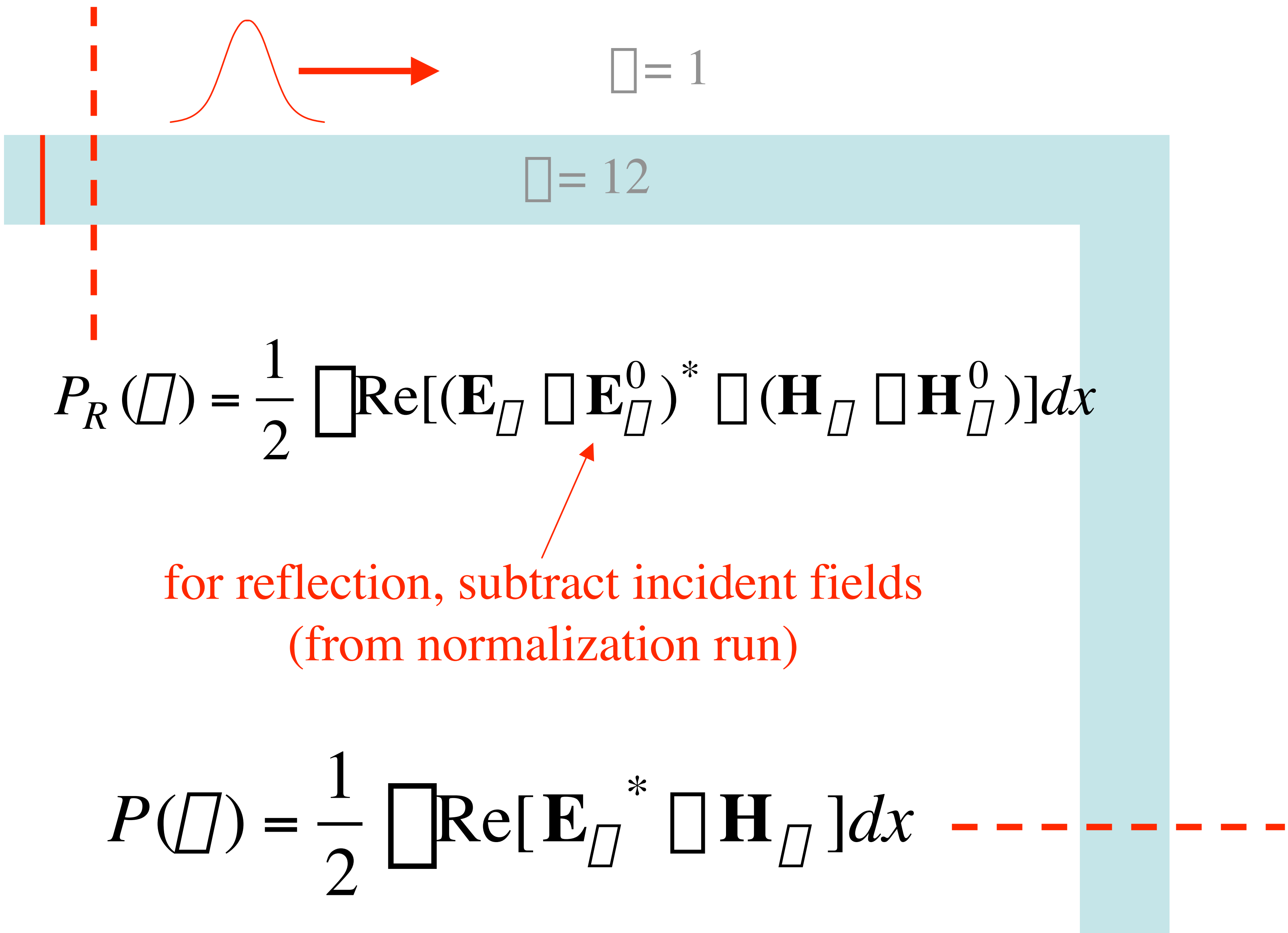


P_0(□)

P(□)

transmission = P(□) / P_0(□)

Reflection Spectra in FDTD

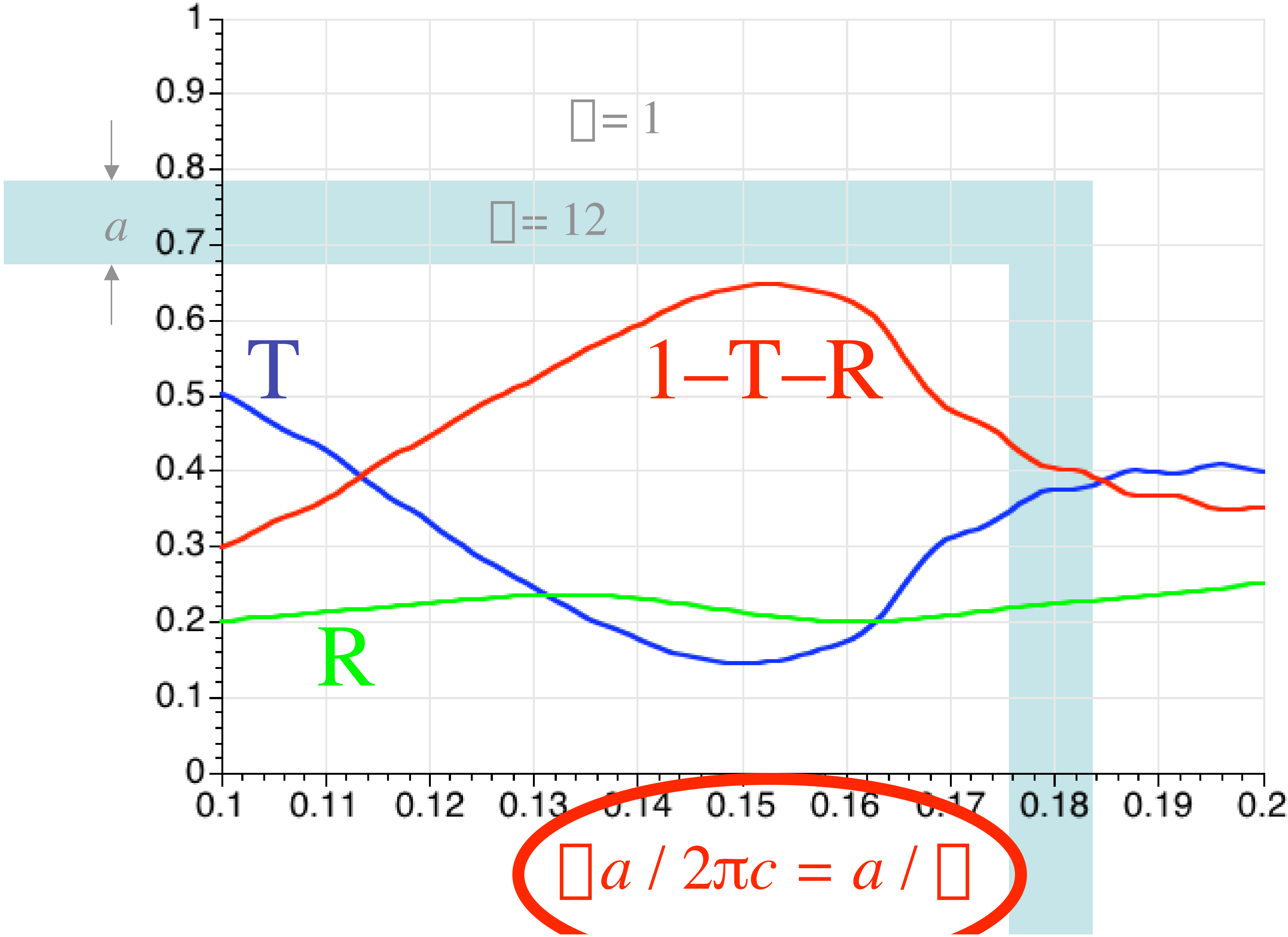


$$P_R(\omega) = \frac{1}{2}\int \mathrm{Re}[(\mathbf{E}_\omega - \mathbf{E}_\omega^0)^* \times (\mathbf{H}_\omega - \mathbf{H}_\omega^0)]\,dx$$

for reflection, subtract incident fields
(from normalization run)

$$P(\omega) = \frac{1}{2}\int \mathrm{Re}[\mathbf{E}_\omega^* \times \mathbf{H}_\omega]\,dx$$

Transmission/Reflection Spectra



Dimensionless Units

Maxwell's equations are *scale invariant*

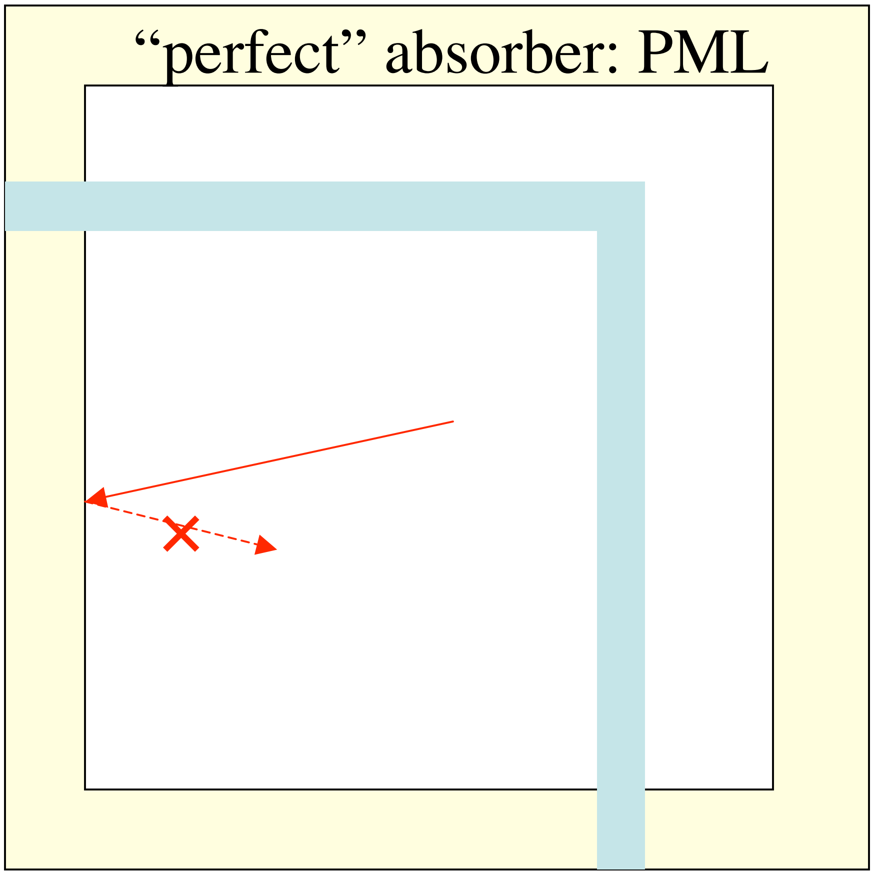
- most useful quantities are dimensionless ratios
  like a / □, for a characteristic lengthscale a
- same ratio, same □, □ = same solution
  regardless of whether a = 1μm or 1km

Our (typical) approach:

pick characteristic lengthscale a

- measure distance in units of a
- measure time in units of a/c
- measure □ in units of $2\pi c/a$ = a / □
- ....

Absorbing Boundaries: Perfectly Matched Layers



Artificial absorbing material *overlapping* the computation

Theoretically reflectionless

… but PML is no longer perfect with finite resolution,
so "gradually turn on" absorption over finite-thickness PML

Computational Photonics Problems

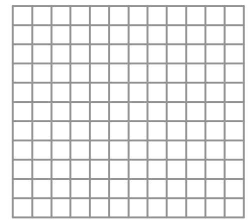
- Time-domain simulation
  - — start with current **J**(**x**,t)
  - — run "numerical experiment" to simulate **E**(**x**, t), **H**(**x**, t)

- Frequency-domain linear response
  - — start with harmonic current **J**(**x**, t) = $e^{-i\omega t}$ **J**(**x**)
  - — solve for steady-state harmonic fields **E**(**x**), **H**(**x**)
  - — involves solving linear equation Ax=b

- Frequency-domain eigensolver
  - — solve for source-free harmonic eigenfields **E**(**x**), **H**(**x**) ~ $e^{-i\omega t}$
  - — involves solving eigenequation $Ax=\omega^2 x$

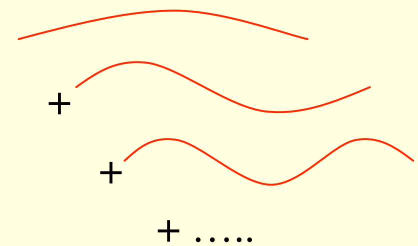
Numerical Methods: Basis Choices

finite difference

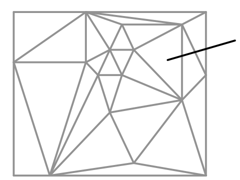


$$\frac{df}{dx} \approx \frac{f(x+\Delta x) - f(x-\Delta x)}{\Delta x} + O(\Delta x^2)$$

spectral methods

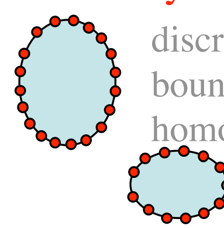


finite elements



in irregular "elements," approximate unknowns by low-degree polynomial

boundary-element methods



discretize only the boundaries between homogeneous media

A Maxwell Eigenproblem

$$\vec{\nabla} \times \vec{E} = -\frac{1}{c}\frac{\partial}{\partial t}\vec{H} = i\frac{\omega}{c}\vec{H}$$

$$\vec{\nabla} \times \vec{H} = \varepsilon\frac{1}{c}\frac{\partial}{\partial t}\vec{E} + \cancelto{0}{\vec{J}} = -i\frac{\omega}{c}\varepsilon\vec{E}$$

dielectric function $\varepsilon(\mathbf{x}) = n^2(\mathbf{x})$

First task:
get rid of this mess

$$\nabla \times \frac{1}{\varepsilon} \nabla \times \vec{H} = \left(\frac{\omega}{c}\right)^2 \vec{H}$$

eigen-operator — eigen-value — eigen-state

+ constraint

$$\nabla \cdot \vec{H} = 0$$

Electronic & Photonic Eigenproblems

Electronic

$$\left[-\frac{\hbar^2}{2m}\nabla^2 + V\right]\psi = E\psi$$

nonlinear eigenproblem

(V depends on e density $|\psi|^2$)

Photonic

$$\nabla \times \frac{1}{\varepsilon} \nabla \times \vec{H} = \left(\frac{\omega}{c}\right)^2 \vec{H}$$

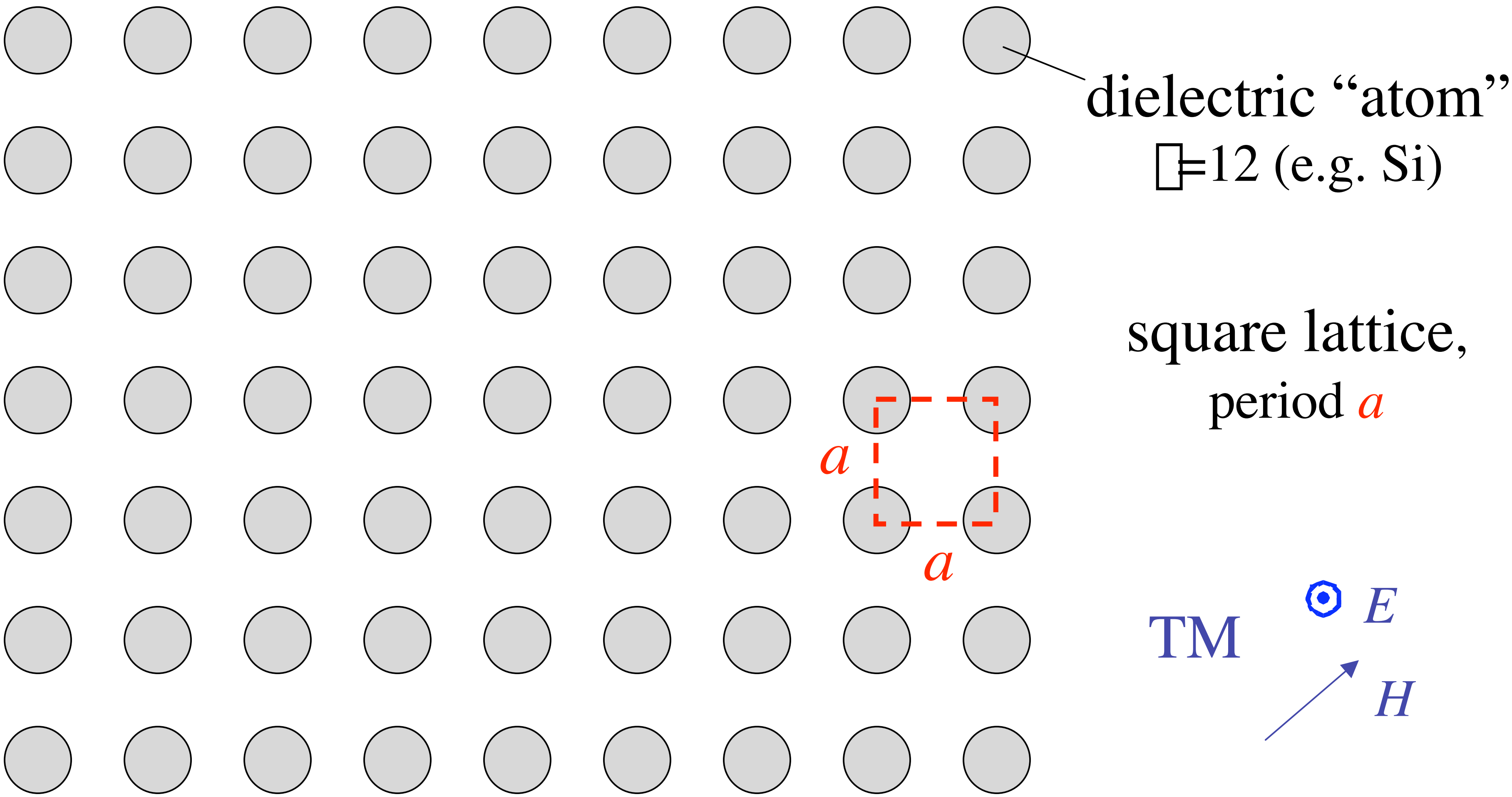
simple linear eigenproblem

(for linear materials)

—many well-known computational techniques

Hermitian = real E & ω, … *Periodicity* = *Bloch's theorem…*

A 2d Model System



Periodic Eigenproblems

if eigen-operator is periodic, then Bloch-Floquet theorem applies:

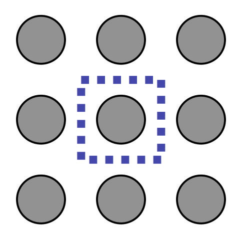
can choose: $$\vec{H}(\vec{x},t) = e^{i\left(\vec{k}\cdot\vec{x} - \omega t\right)} \vec{H}_{\vec{k}}(\vec{x})$$

planewave

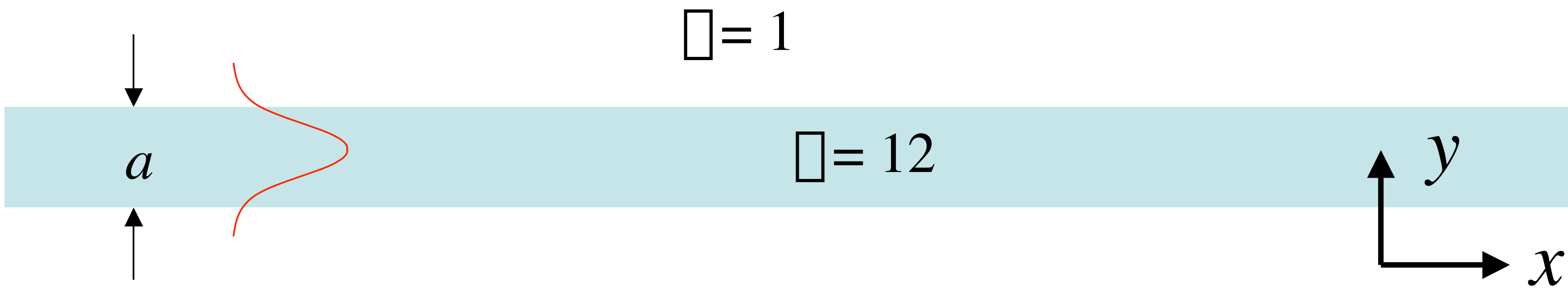
periodic "envelope"

Corollary 1: **k** is conserved, *i.e.* no scattering of Bloch wave

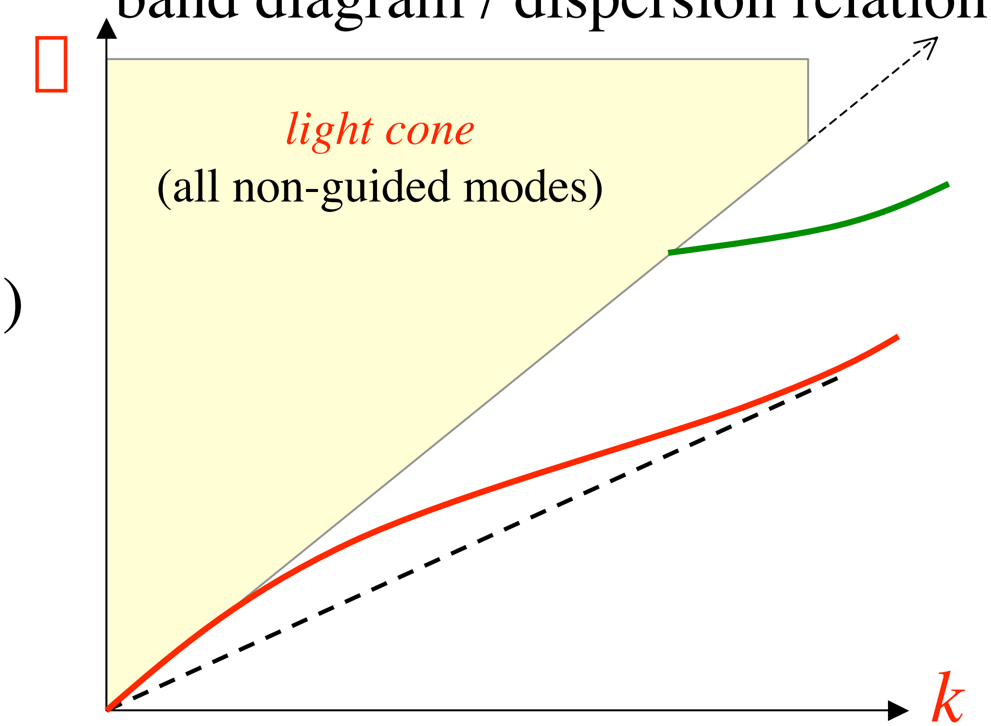
Corollary 2: $\vec{H}_{\vec{k}}$ given by finite unit cell, so ω are discrete $\omega_n(\mathbf{k})$



A More Familiar Eigenproblem



band diagram / dispersion relation



find the normal modes
of the waveguide:

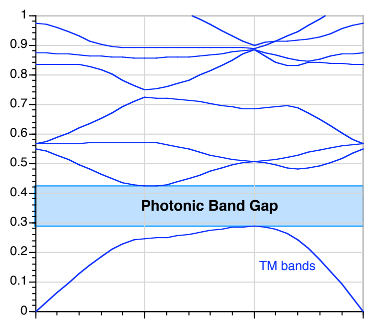
$$\mathbf{H}(y,t) = \mathbf{H}_k(y)e^{i(kx - \omega t)}$$

(propagation constant *k*
a.k.a. β)

Solving the Maxwell Eigenproblem

Finite cell ➔ *discrete* eigenvalues ω_n

Want to solve for $\omega_n(\mathbf{k})$,
& plot vs. "all" **k** for "all" n,



$$(\nabla + i\mathbf{k}) \times \frac{1}{\varepsilon}(\nabla + i\mathbf{k}) \times \mathbf{H}_n = \frac{\omega_n^2}{c^2}\mathbf{H}_n$$

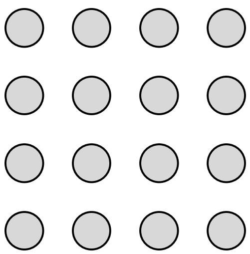
constraint: $(\nabla + i\mathbf{k}) \cdot \mathbf{H} = 0$

where: $\mathbf{H}(x,y)\, e^{i(\mathbf{k}\cdot\mathbf{x} - \omega t)}$

1. Limit range of **k**: irreducible Brillouin zone
2. Limit degrees of freedom: expand **H** in finite basis
3. Efficiently solve eigenproblem: iterative methods

Solving the Maxwell Eigenproblem: 1

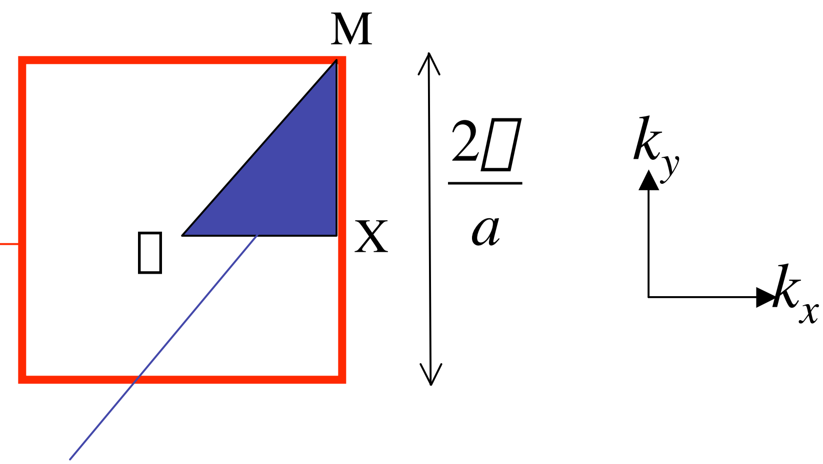
1 Limit range of **k**: irreducible Brillouin zone



—Bloch's theorem: solutions are periodic in **k**

first Brillouin zone

= minimum |**k**| "primitive cell"



irreducible Brillouin zone: reduced by symmetry

2 Limit degrees of freedom: expand **H** in finite basis

3 Efficiently solve eigenproblem: iterative methods

Solving the Maxwell Eigenproblem: 2a

1. Limit range of **k**: irreducible Brillouin zone

2. Limit degrees of freedom: expand **H** in finite basis (*N*)

$$|\mathbf{H}\rangle = \mathbf{H}(\mathbf{x}_t) = \sum_{m=1}^{N} h_m \mathbf{b}_m(\mathbf{x}_t) \qquad \text{solve: } \hat{A}|\mathbf{H}\rangle = \omega^2|\mathbf{H}\rangle$$

finite matrix problem: $Ah = \omega^2 Bh$

$$\langle \mathbf{f}|\mathbf{g}\rangle = \int \mathbf{f}^* \cdot \mathbf{g} \qquad A_{m\ell} = \langle \mathbf{b}_m|\hat{A}|\mathbf{b}_\ell\rangle \qquad B_{m\ell} = \langle \mathbf{b}_m|\mathbf{b}_\ell\rangle$$

3. Efficiently solve eigenproblem: iterative methods

Solving the Maxwell Eigenproblem: 2b

1. Limit range of $\mathbf{k}$: irreducible Brillouin zone

2. Limit degrees of freedom: expand $\mathbf{H}$ in finite basis

   — must satisfy constraint: $(\nabla + i\mathbf{k}) \cdot \mathbf{H} = 0$

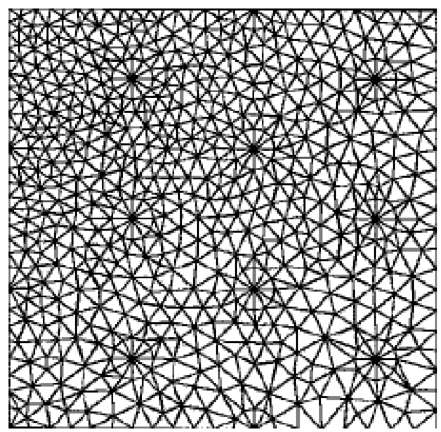
Planewave (FFT) basis

$$\mathbf{H}(\mathbf{x}_t) = \sum_{\mathbf{G}} \mathbf{H}_{\mathbf{G}} e^{i\mathbf{G}\cdot\mathbf{x}_t}$$

constraint: $\mathbf{H}_{\mathbf{G}} \cdot (\mathbf{G} + \mathbf{k}) = 0$

uniform "grid," periodic boundaries,
simple code, O(N log N)

Finite-element basis



[figure: Peyrilloux *et al.*, *J. Lightwave Tech.* **21**, 536 (2003)]

constraint, boundary conditions:

Nédélec elements

[Nédélec, *Numerische Math.* **35**, 315 (1980)]

nonuniform mesh,
more arbitrary boundaries,
complex code & mesh, O(N)

3. Efficiently solve eigenproblem: iterative methods

Solving the Maxwell Eigenproblem: 3a

1. Limit range of **k**: irreducible Brillouin zone
2. Limit degrees of freedom: expand **H** in finite basis
3. Efficiently solve eigenproblem: iterative methods

$$Ah = \Pi^2 Bh$$

Slow way: compute A & B, ask LAPACK for eigenvalues

— requires $O(N^2)$ storage, $O(N^3)$ time

Faster way:

— start with *initial guess* eigenvector h_0

— *iteratively* improve

— $O(Np)$ storage, ~ $O(Np^2)$ time for p eigenvectors

(p smallest eigenvalues)

Solving the Maxwell Eigenproblem: 3b

1. Limit range of **k**: irreducible Brillouin zone
2. Limit degrees of freedom: expand **H** in finite basis
3. Efficiently solve eigenproblem: iterative methods

$$Ah = \Pi^2 Bh$$

Many iterative methods:

— Arnoldi, Lanczos, Davidson, Jacobi-Davidson, …, Rayleigh-quotient minimization

Solving the Maxwell Eigenproblem: 3c

1. Limit range of **k**: irreducible Brillouin zone
2. Limit degrees of freedom: expand **H** in finite basis
3. Efficiently solve eigenproblem: iterative methods

$$Ah = \omega^2 Bh$$

Many iterative methods:

— Arnoldi, Lanczos, Davidson, Jacobi-Davidson, …,
Rayleigh-quotient minimization

for Hermitian matrices, smallest eigenvalue ω_0 minimizes:

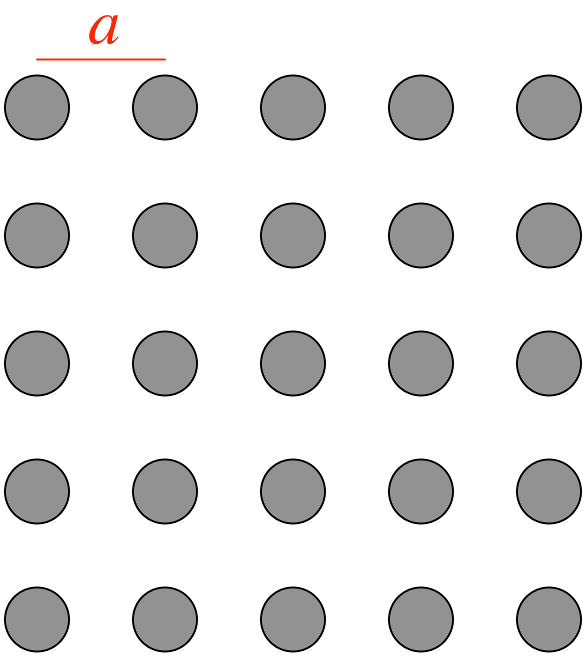
"variational theorem"

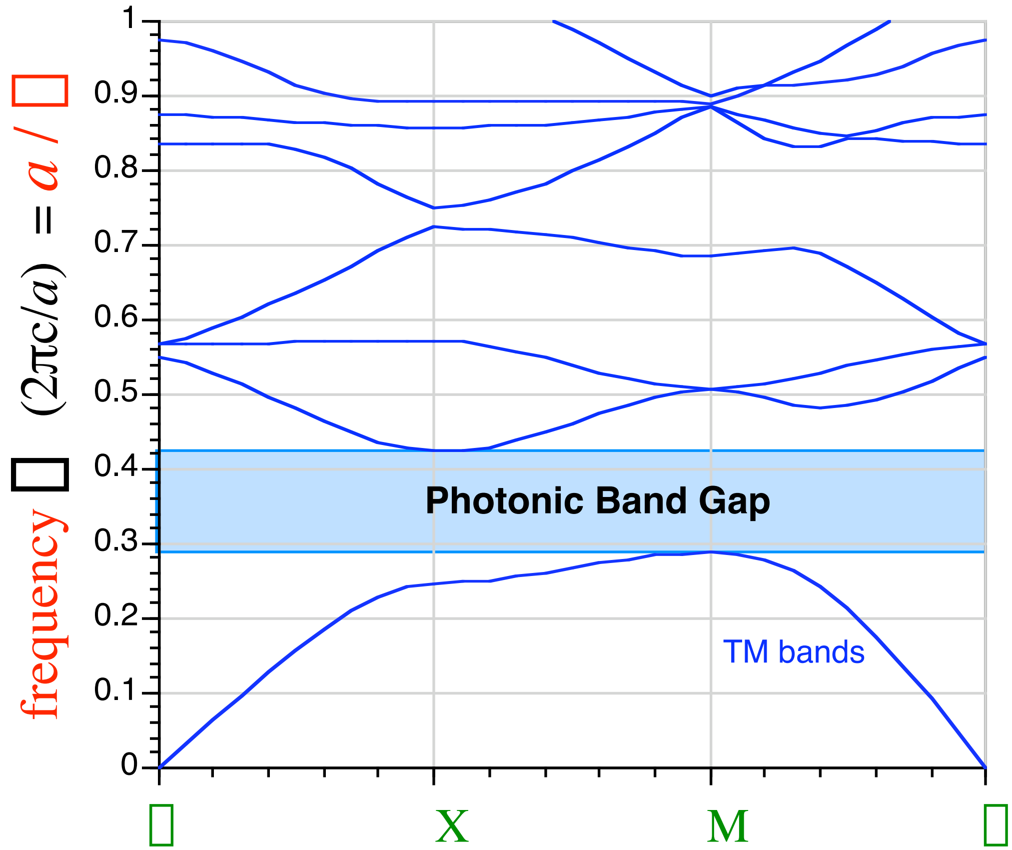
$$\omega_0^2 = \min_h \frac{h' Ah}{h' Bh}$$

minimize by preconditioned conjugate-gradient (or…)

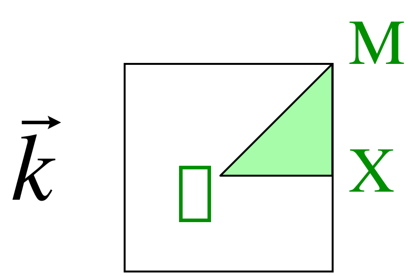
Band Diagram of 2d Model System

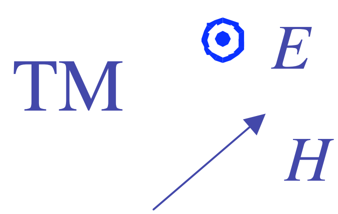
(radius 0.2a rods, ε=12)





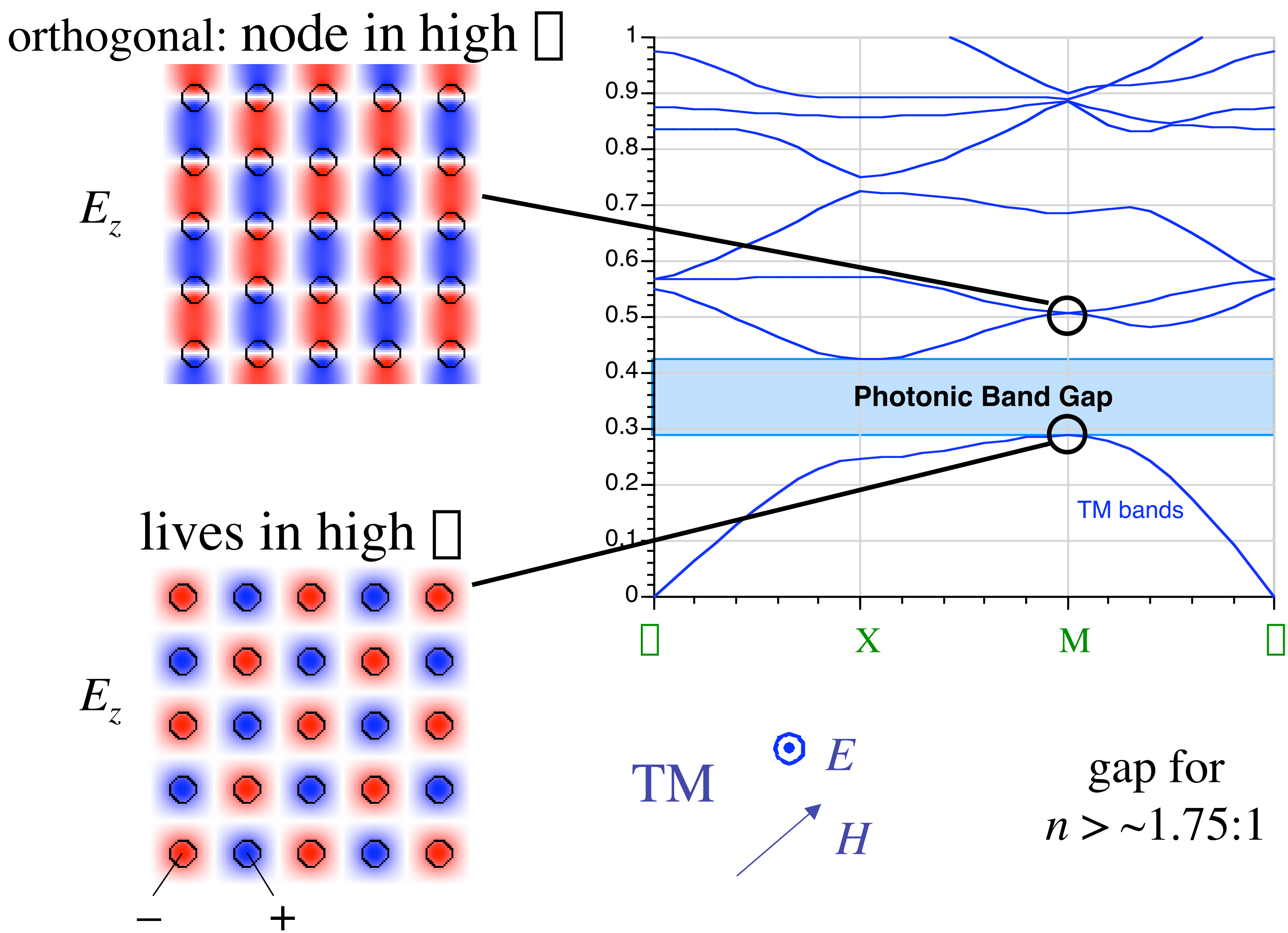
irreducible Brillouin zone





gap for
n > ~1.75:1

Origin of Gap in 2d Model System



gap for
$n > \sim 1.75:1$

The Iteration Scheme is *Important*

(minimizing function of 10^4–10^8+ variables!)

$$\omega_0^2 = \min_h \frac{h' A h}{h' B h} = f(h)$$

Steepest-descent: minimize $(h + \alpha \nabla f)$ over α … repeat

Conjugate-gradient: minimize $(h + \alpha\, d)$

— d is ∇f + (stuff): *conjugate* to previous search dirs

Preconditioned steepest descent: minimize $(h + \alpha\, d)$

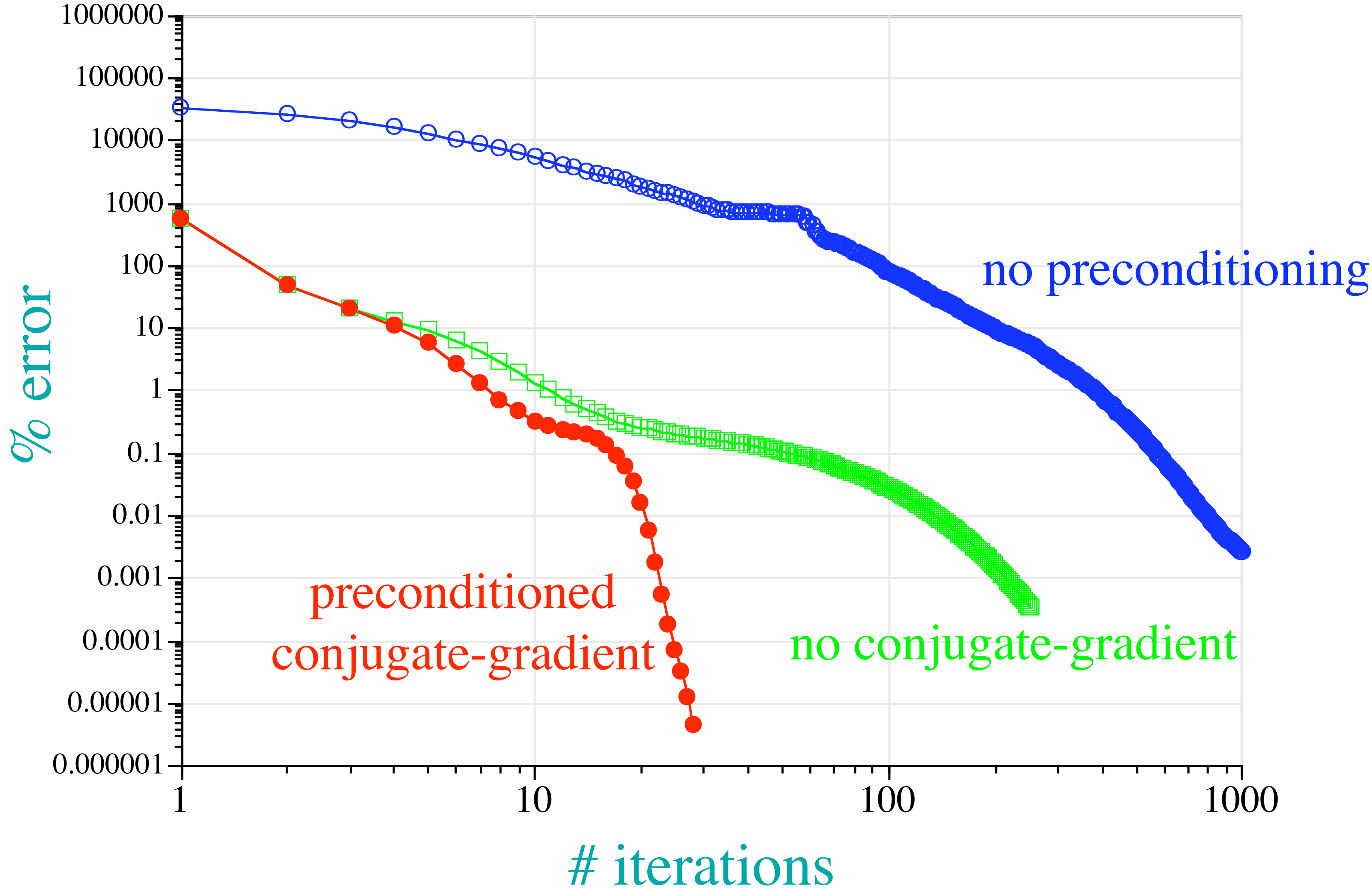
— d = (approximate A^{-1}) ∇f ~ Newton's method

Preconditioned conjugate-gradient: minimize $(h + \alpha\, d)$

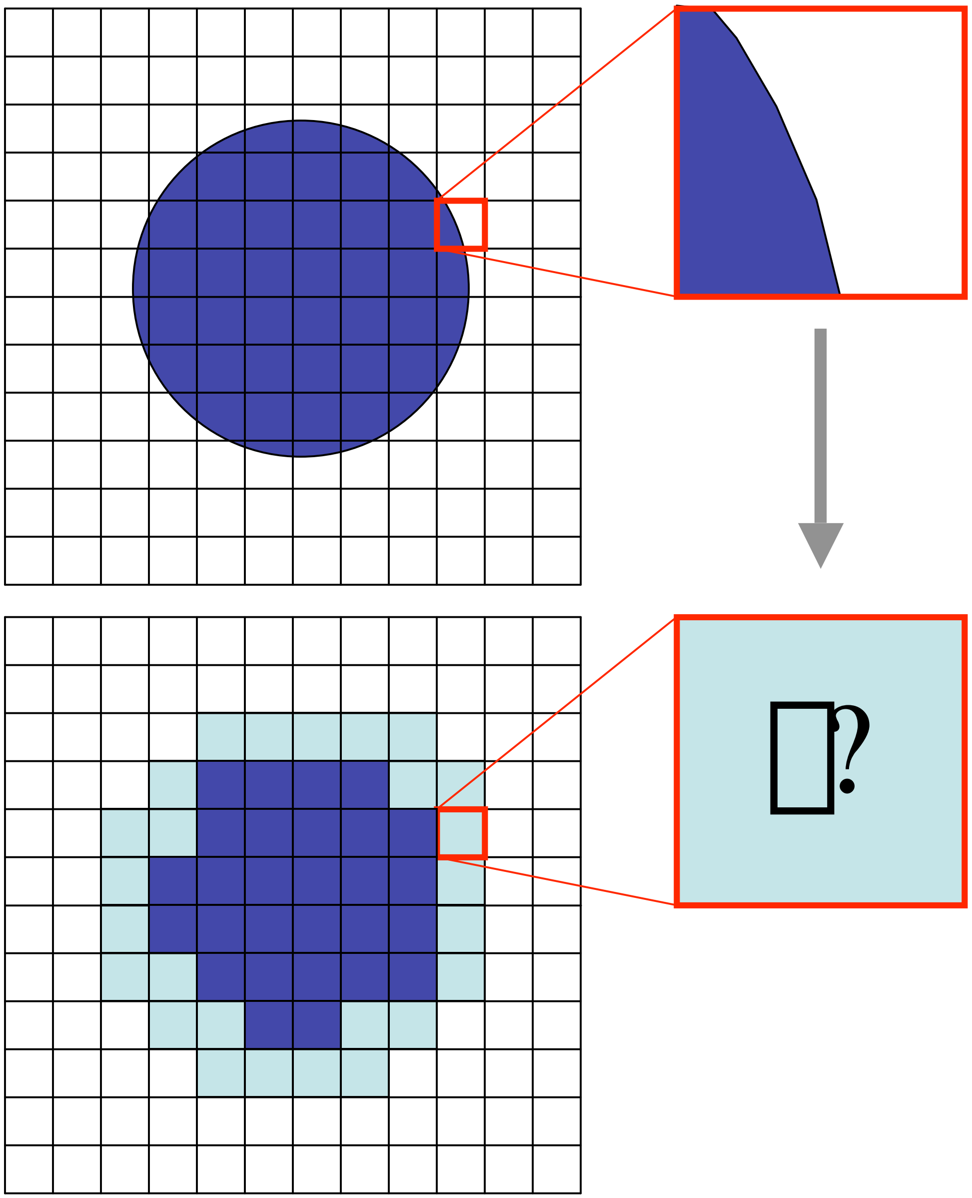
— d is (approximate A^{-1}) [∇f + (stuff)]

The Iteration Scheme is *Important*

(minimizing function of ~40,000 variables)



The Boundary Conditions are Tricky



$\mathbf{E}_{\parallel}$ is continuous

$\mathbf{E}_{\perp}$ is discontinuous

($\mathbf{D}_{\perp} = \varepsilon \mathbf{E}_{\perp}$ is continuous)

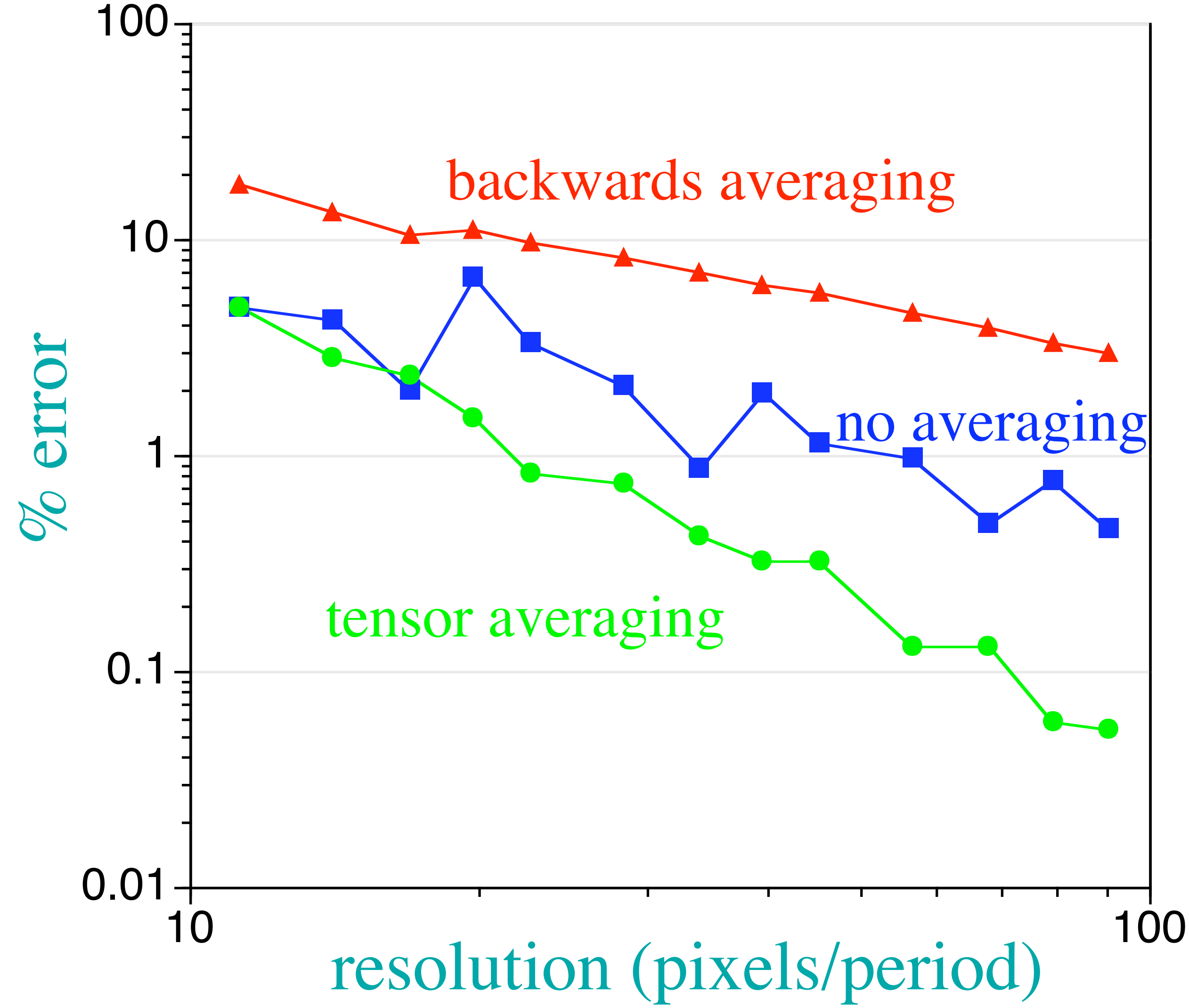
Any single scalar ε fails:

(mean **D**) ≠ (any ε) (mean **E**)

Use a tensor ε:

$$\varepsilon \rightarrow \begin{pmatrix} \langle \varepsilon \rangle & & \\ & \langle \varepsilon \rangle & \\ & & \langle \varepsilon^{-1} \rangle^{-1} \end{pmatrix} \begin{matrix} \mathbf{E}_{\parallel} \\ \\ \mathbf{E}_{\perp} \end{matrix}$$

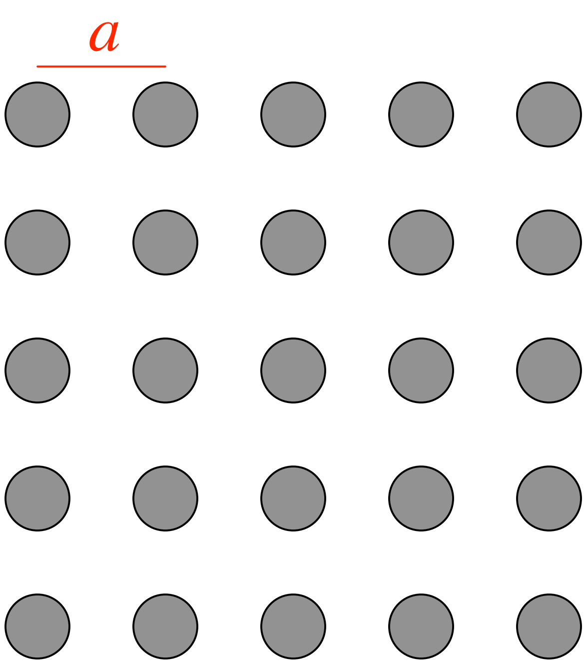
The □-averaging is *Important*

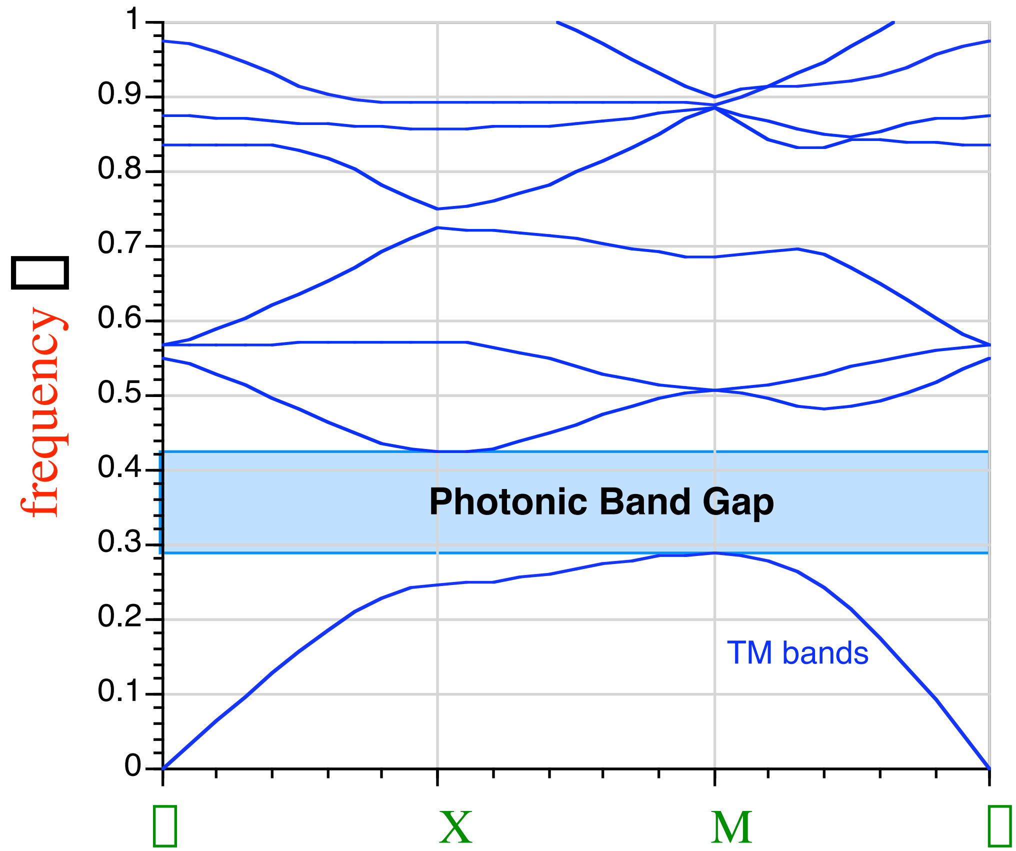


correct averaging
changes *order*
of convergence
from Δx to Δx^2

(similar effects
in other E&M
numerics & analyses)

Gap, Schmap?

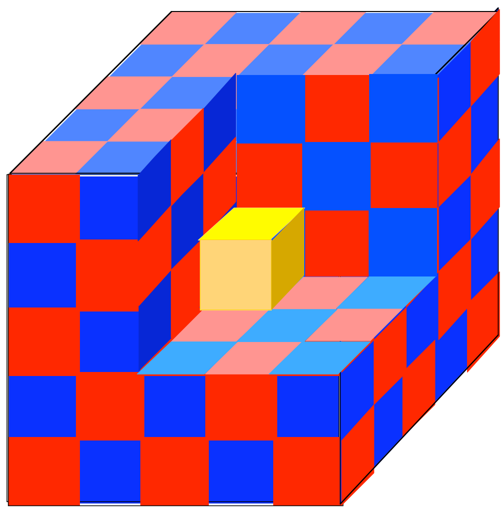




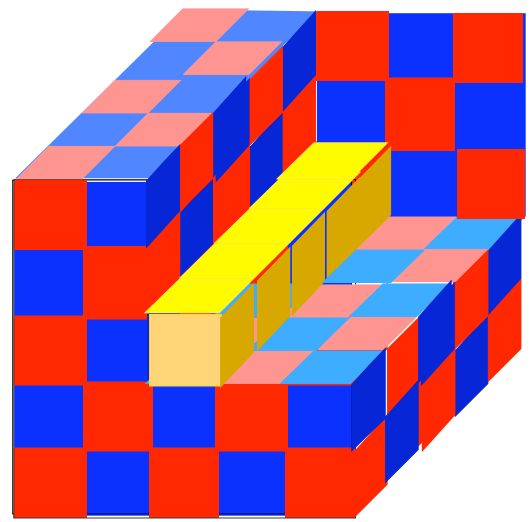
But, what can we *do* with the gap?

Intentional "defects" are good

microcavities

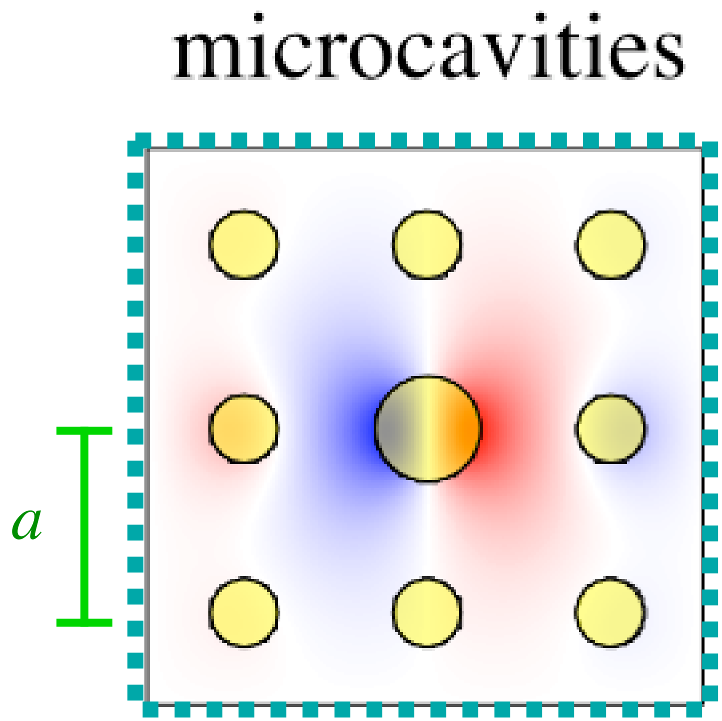


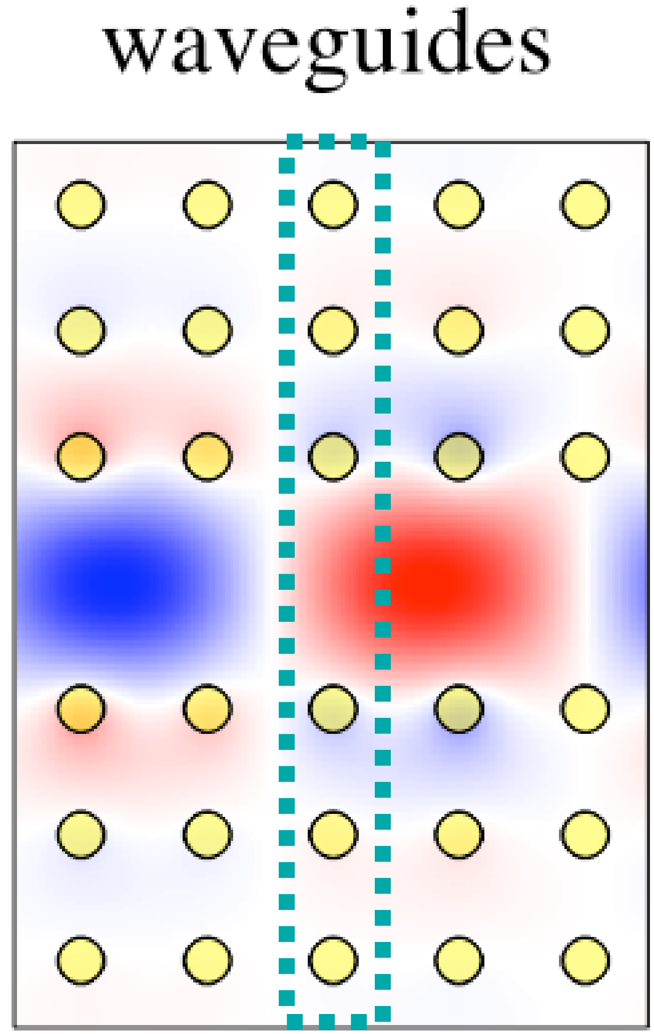
waveguides ("wires")



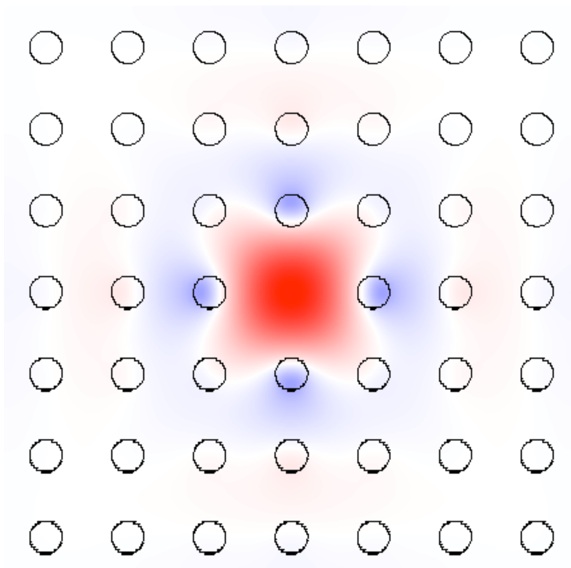
Intentional “defects” in 2d

(Same computation, with supercell = many primitive cells)





Microcavity Blues



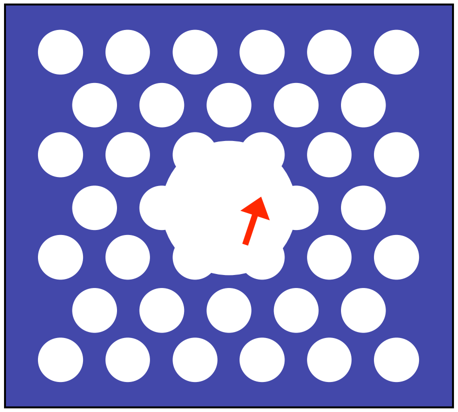
For cavities (*point defects*)
frequency-domain has its drawbacks:

- Best methods compute lowest-□ bands,
  but N^d supercells have N^d modes
  below the cavity mode — *expensive*

- Best methods are for Hermitian operators,
  but losses requires non-Hermitian

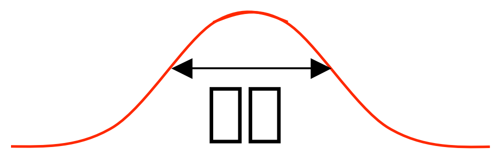
Time-Domain Eigensolvers

(finite-difference time-domain = FDTD)



Simulate Maxwell's equations on a discrete grid,
+ absorbing boundaries (leakage loss)

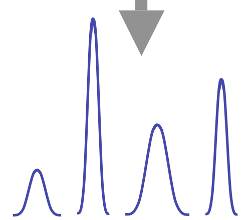
• Excite with broad-spectrum dipole (↑) source



signal processing

complex ω_n

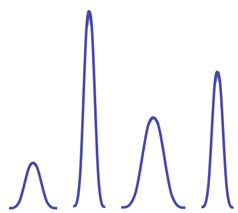
[Mandelshtam,
J. Chem. Phys. **107**, 6756 (1997)]



Response is many sharp peaks,
one peak per mode

decay rate in time gives loss

Signal Processing is Tricky

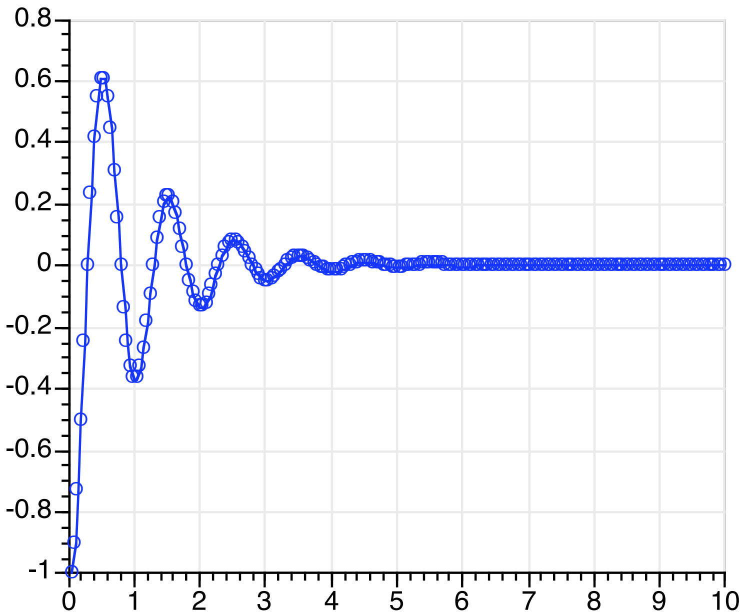


signal processing

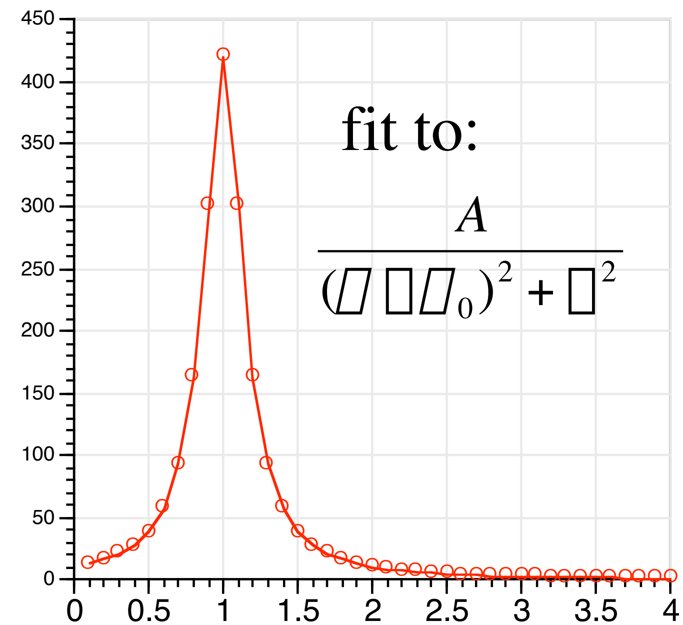
complex □$_n$

?

a common approach: least-squares fit of spectrum



FFT

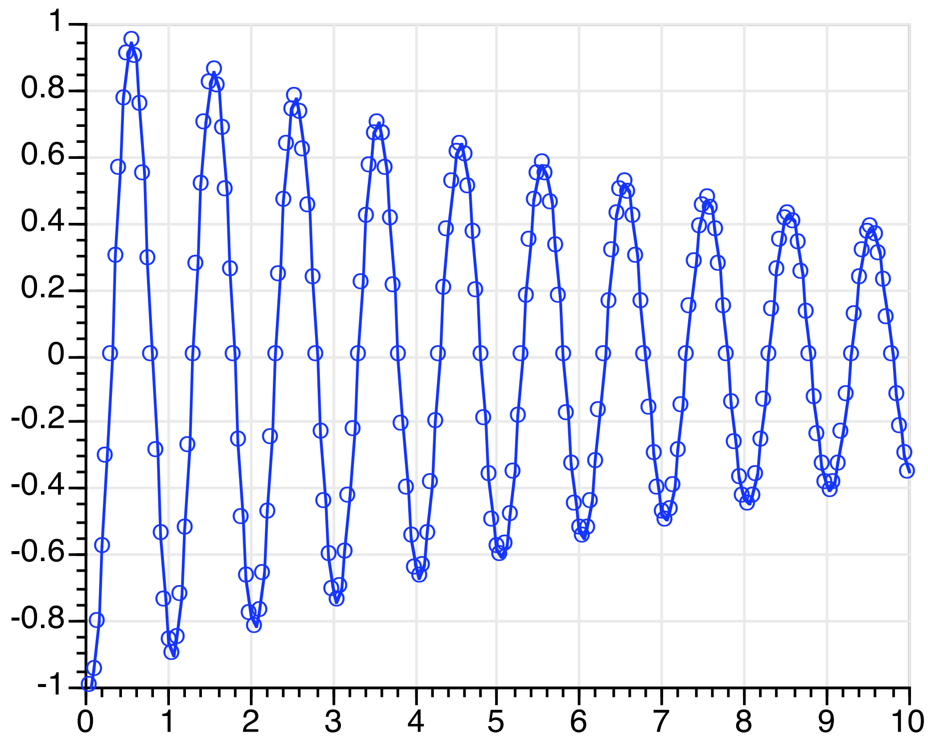


Decaying signal (*t*)

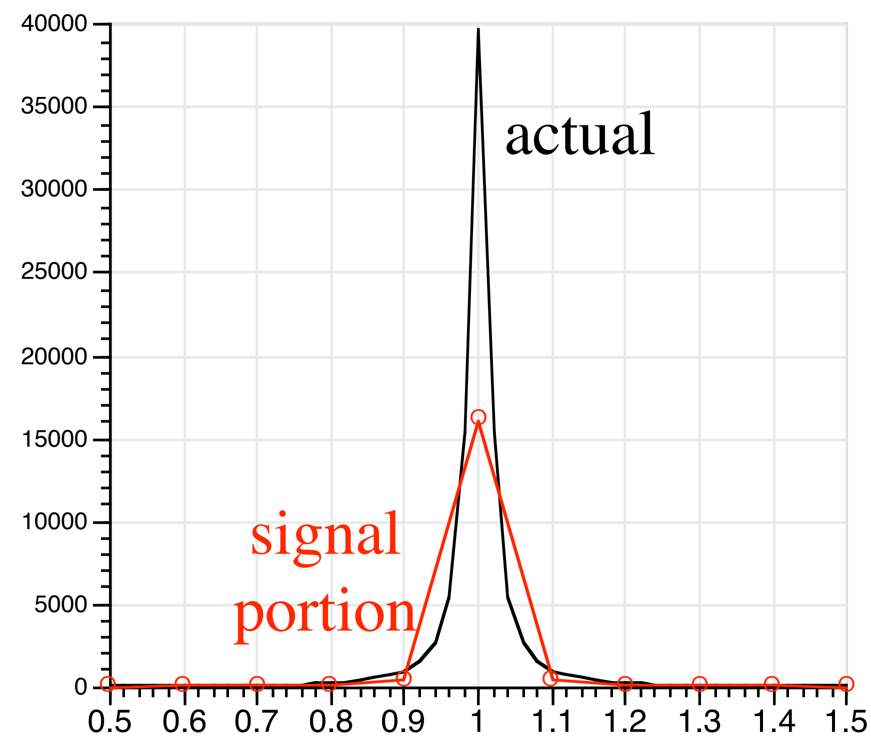
Lorentzian peak (□)

Fits and Uncertainty

problem: have to run long enough to *completely* decay



Portion of decaying signal (*t*)

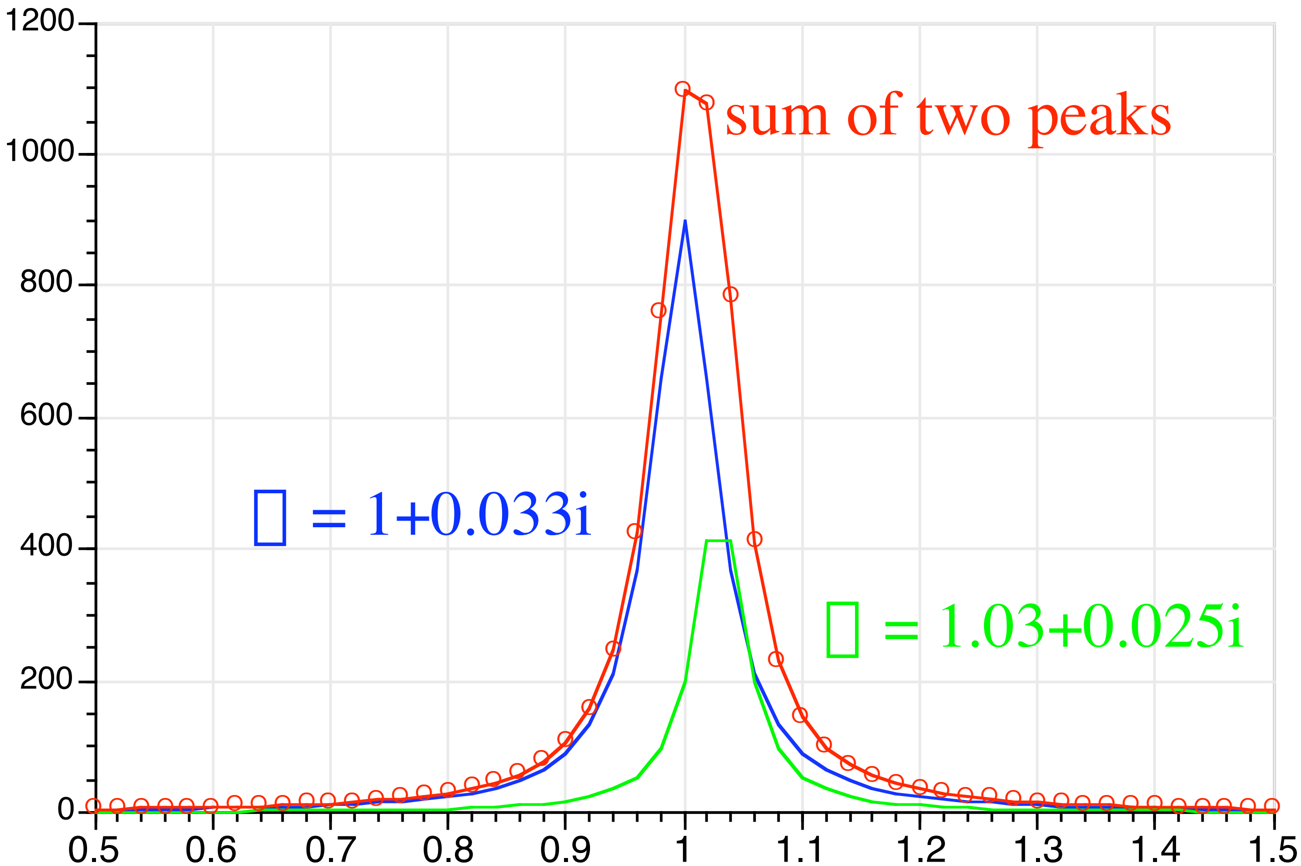


Unresolved Lorentzian peak (□)

There is a better way, which gets complex □ to > 10 digits

Unreliability of Fitting Process

Resolving two overlapping peaks is
near-impossible 6-parameter nonlinear fit
(too many local minima to converge reliably)



Sum of two Lorentzian peaks (ω)

There is a better way, which gets complex ω for *both* peaks to > 10 digits

Quantum-inspired signal processing (NMR spectroscopy):

Filter-Diagonalization Method (FDM)

[Mandelshtam, *J. Chem. Phys.* **107**, 6756 (1997)]

Given time series y_n, write: $$y_n = y(n\Delta t) = \sum_k a_k e^{-i\omega_k n\Delta t}$$

…find *complex* amplitudes a_k & frequencies ω_k

by a simple linear-algebra problem!

Idea: pretend $y(t)$ is autocorrelation of a quantum system:

$$\hat{H}|\psi\rangle = i\hbar\frac{\partial}{\partial t}|\psi\rangle$$

time-Δt evolution-operator: $\hat{U} = e^{-i\hat{H}\Delta t/\hbar}$

say: $$y_n = \langle\psi(0)|\psi(n\Delta t)\rangle = \langle\psi(0)|\hat{U}^n|\psi(0)\rangle$$

Filter-Diagonalization Method (FDM)

[Mandelshtam, *J. Chem. Phys.* **107**, 6756 (1997)]

$$y_n = \langle \Phi(0) | \Phi(n\Delta t) \rangle = \langle \Phi(0) | \hat{U}^n | \Phi(0) \rangle \qquad \hat{U} = e^{-i\hat{H}\Delta t/\hbar}$$

We want to diagonalize U: eigenvalues of U are $e^{i\omega\Delta t}$

…expand U in basis of $|\Phi(n\Delta t)>$:

$$U_{m,n} = \langle \Phi(m\Delta t) | \hat{U} | \Phi(n\Delta t) \rangle = \langle \Phi(0) | \hat{U}^m \hat{U} \hat{U}^n | \Phi(0) \rangle = y_{m+n+1}$$

U_{mn} given by y_n's — just diagonalize known matrix!

Filter-Diagonalization Summary

[Mandelshtam, *J. Chem. Phys.* **107**, 6756 (1997)]

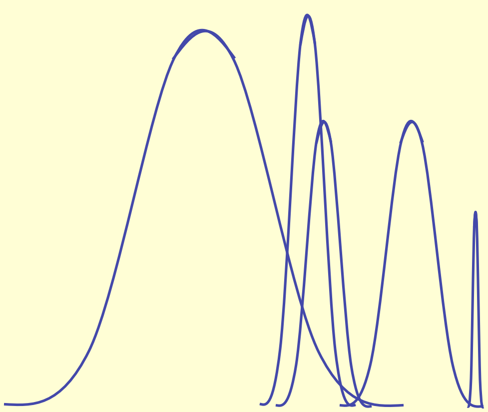
U_{mn} given by y_n's — just diagonalize known matrix!

A few omitted steps:

— Generalized eigenvalue problem (basis not orthogonal)

— Filter y_n's (Fourier transform):

small bandwidth = smaller matrix (less singular)



- resolves many peaks at once
- # peaks not known *a priori*
- resolve overlapping peaks
- resolution >> Fourier uncertainty

Do try this at home

FDTD simulation:

`http://ab-initio.mit.edu/meep/`

Bloch-mode eigensolver:

`http://ab-initio.mit.edu/mpb/`

Filter-diagonalization:

`http://ab-initio.mit.edu/harminv/`

Photonic-crystal tutorials (+ THIS TALK):

`http://ab-initio.mit.edu/`
`/photons/tutorial/`

Meep (FDTD)

- Arbitrary $\varepsilon(\mathbf{x})$ — including dispersive, loss/gain, and nonlinear [$\chi^{(2)}$ and $\chi^{(3)}$]
- Arbitrary $\mathbf{J}(\mathbf{x},t)$
- PML/periodic/metal bound.
- 1d/2d/3d/cylindrical
- power spectra
- eigenmodes

MPB (Eigensolver)

- Arbitrary periodic $\varepsilon(\mathbf{x})$ — anisotropic, magneto-optic, … (lossless, linear materials)
- 1d/2d/3d
- band diagrams, group velocities perturbation theory, …

Free/open-source software (GNU)

- MPI parallelism
- exploit mirror symmetries
- fully scriptable interface
- built-in multivariate optimization, integration, root-finding, …
- field output (standard HDF5 format)

Unix Philosophy

combine small, well-designed tools, via files

Input text file → MPB/Meep → standard formats (text + HDF5)

↓

Visualization / Analysis software (Matlab, Mayavi [vtk], command-line tools, …)

Disadvantage:

— have to learn several programs

Advantages:

— flexibility

— batch processing, shell scripting

— ease of development

Unix Philosophy

combine small, well-designed tools, via files

Input text file

MPB/Meep → standard formats (text + HDF5)

GNU Guile scripting interpreter (Scheme language)

Visualization / Analysis software (Matlab, Mayavi [vtk], command-line tools, …)

Embed a full scripting language:

— parameter sweeps

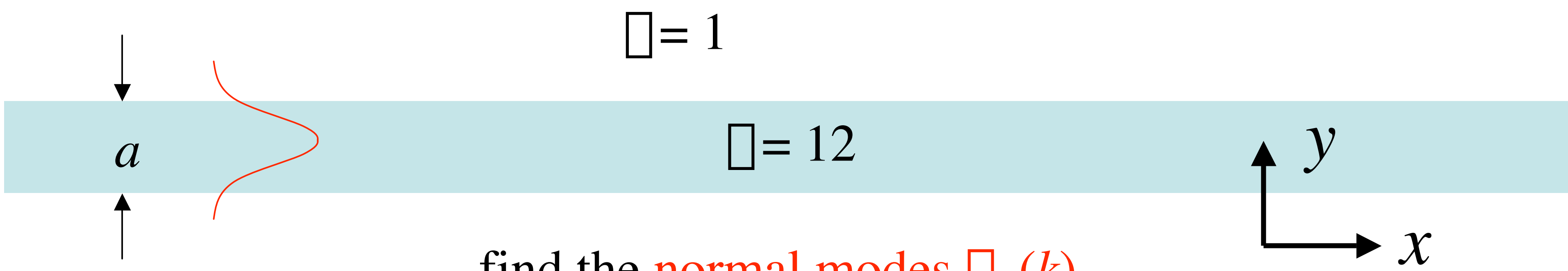
— complex parameterized geometries

— optimization, integration, etc.

— programmable $\mathbf{J}(\mathbf{x}, t)$, etc.

— … Turing complete

A Simple Example (MPB)



find the normal modes $\omega_n(k)$
of the waveguide:

$$\mathbf{H}(y,t) = \mathbf{H}_k(y)e^{i(kx-\omega t)}$$

Need to specify:

- computational cell size/resolution
- geometry, i.e. $\varepsilon(y)$
- what k values
- how many modes (n = 1, 2, … ?)

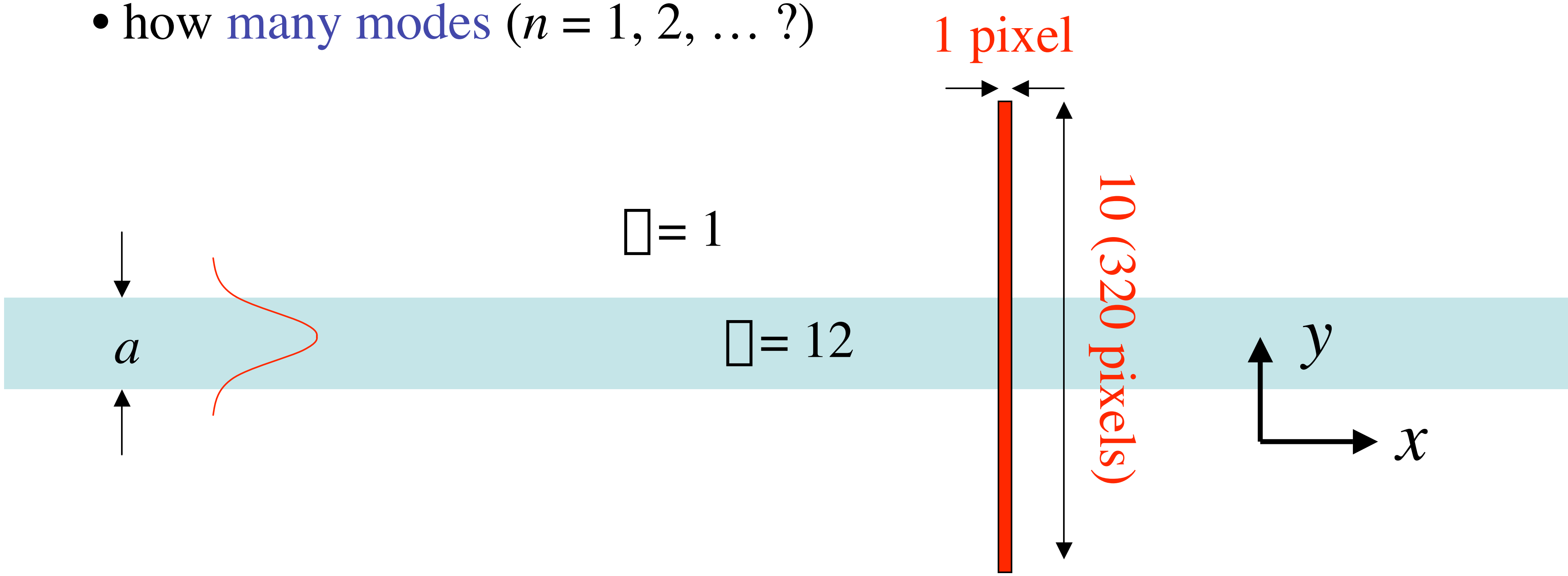
A File Format Made of Parentheses

Need to specify:

- computational cell size/resolution

```
(set! geometry-lattice (make lattice (size no-size 10 no-size)
(set! resolution 32)
```

- geometry, i.e. ε(*y*)
- what *k* values
- how many modes (*n* = 1, 2, … ?)



A File Format Made of Parentheses

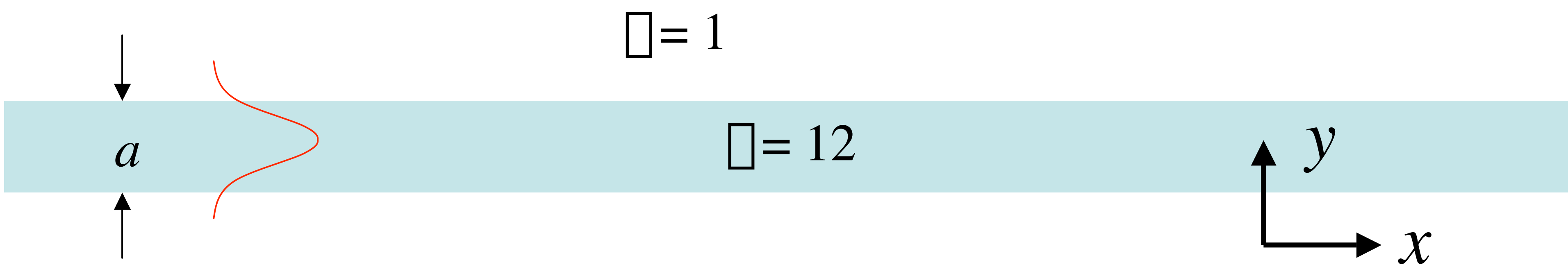
Need to specify:

- computational cell size/resolution
- geometry, i.e. $\varepsilon(y)$

(choose units of *a*)

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

- what *k* values
- how many modes (n = 1, 2, … ?)



A File Format Made of Parentheses

Need to specify:

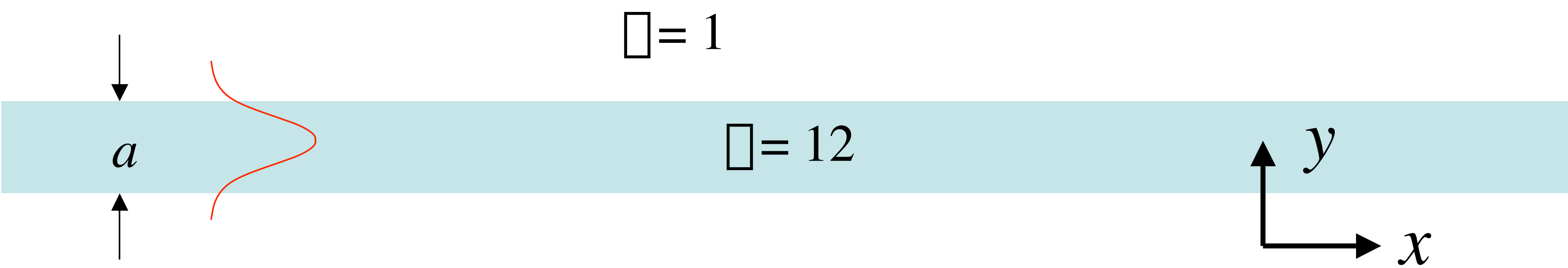
- computational cell size/resolution
- geometry, i.e. $\varepsilon(y)$
- what k values

(units of $2\pi/a$)

```
(set! k-points
  (interpolate 10 (list (vector3 0 0 0) (vector3 2 0 0))))
```

(built-in function)

- how many modes (n = 1, 2, … ?)



A File Format Made of Parentheses

Need to specify:

- computational cell size/resolution
- geometry, i.e. $\varepsilon(y)$
- what k values
- how many modes (n = 1, 2, … ?)

```
(set! num-bands 5)
```

…Then run:

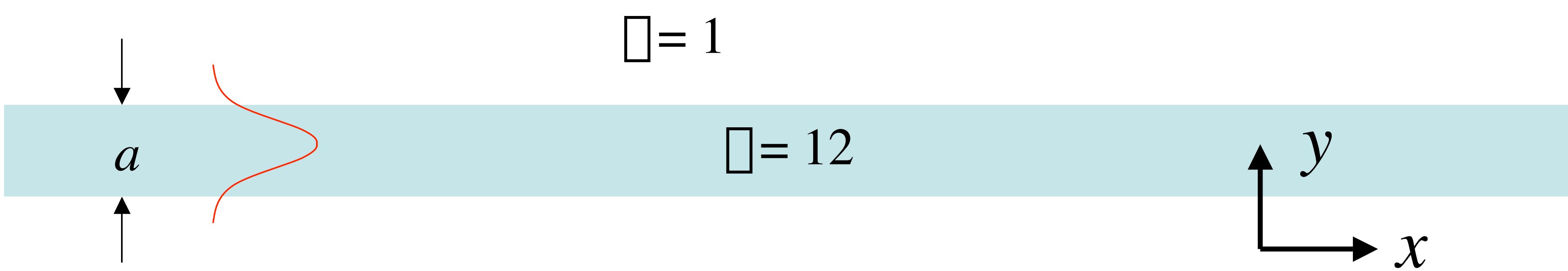
```
(run)
```

or only TM polarization:

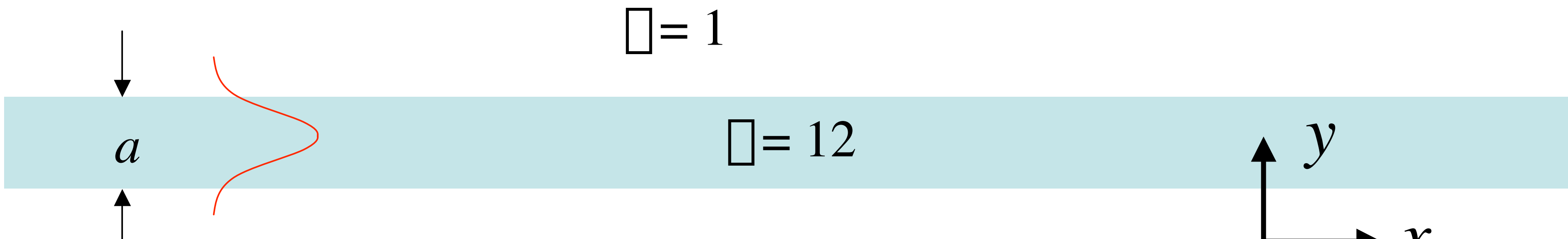
```
(run-tm)
```

or only TM, even modes:

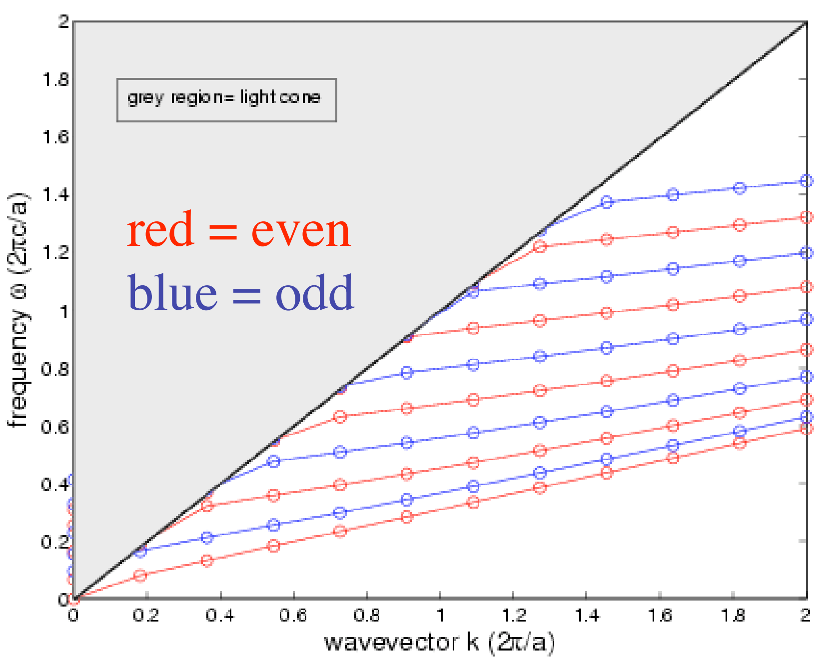
```
(run-tm-yeven)
```

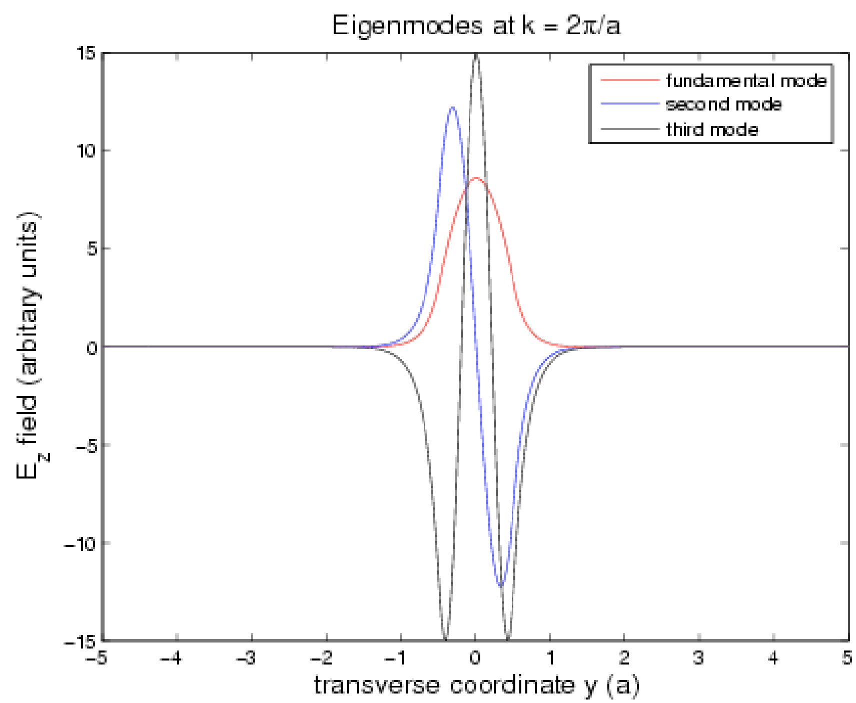


Simple Example (MPB) Results



find the normal modes $\omega_n(k)$
of the waveguide:





Do try this at home

FDTD simulation:

`http://ab-initio.mit.edu/meep/`

Bloch-mode eigensolver:

`http://ab-initio.mit.edu/mpb/`

Filter-diagonalization:

`http://ab-initio.mit.edu/harminv/`

Photonic-crystal tutorials (+ THIS TALK):

`http://ab-initio.mit.edu/`
`/photons/tutorial/`