Computational Photonics
Tutorial 2:
Mode analysis by finite difference method

10 May 2023

Dr. Markus Nyman, markus.nyman@kit.edu
M.Sc. Nigar Asadova, nigar.asadova@kit.edu

Exercises for everyone:

- Task I: 1D FD mode analysis
- Task II: 2D FD mode analysis

Extended exercises:

- Task III*: Group velocity and dispersion
- Task IV*: Periodic boundary conditions

Pitfalls (courtesy of last tutorial)

Real vs complex arrays

```
a = np.zeros(5)
b = np.array([0, 1+1j, 0, 0, 0])
a[1] = b[1]  # ComplexWarning
print(a) # array([0., 1., 0., 0., 0.])
```

Can use `np.zeros(5, dtype='complex128')` to ensure complex arrays.

Not copying an array

```
a = np.zeros(5)
b = np.linspace(0,4,5)
for i in range(5):
    if i == 0:
        a = b
    else:
        a[i] = a[i] + 1
print(a) # [0. 2. 3. 4. 5.]
print(b) # [0. 2. 3. 4. 5.]
```

Can use `a = b.copy()` to ensure that a will be a new array.

Sparse matrices in Python / scipy

```
from scipy.sparse import diags, dok_matrix
from scipy.sparse.linalg import eigs

### Filling individual elements
M = dok_matrix((5,5))  # dict of keys: easy to modify, slow to use
for i in range(5):
    M[i,i] = 2.0
for i in range(3):
    M[i,i+2] = 5.0
M = M.tocsr()  # compressed sparse row: hard to modify, fast to use
print(M.todense())

### Using diagonal filling
M = diags([2.0*np.ones(5), 5.0*np.ones(5)], [0, 2], shape=(5,5))
print(M.todense())

### The three eigenvalues and eigenvectors, with largest real parts
eigenvalues, eigenvectors = eigs(M, 3, which='LR')
```

Computational Photonics

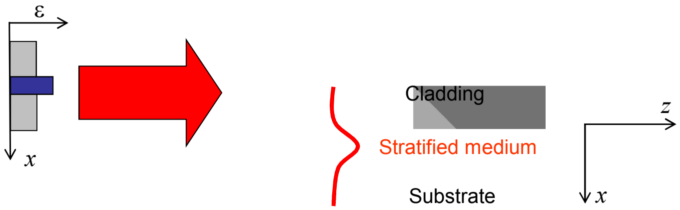
Seminar 04, May 15, 2020

Implementation of a Finite-Difference Mode Solver

- Implementation of 2^{nd} order finite difference schemes in matrix notation
- Calculation of the guided modes in a slab waveguide system
- Calculation of the guided modes in a strip waveguide system

Abbe School of Photonics | JENA
Friedrich-Schiller-Universität

Guided modes in 1+1 (=2D) systems (stratified media)



- no y-dependence
- phase evolution in the z-direction

➔ looking for beams without diffraction in x-direction

Guided modes in 1+1D systems (TE modes)

Assuming weak guiding ($\varepsilon_0 \varepsilon(\omega) \operatorname{div} \mathbf{E}(\mathbf{r},\omega) \approx 0$), we can use the

Helmholtz equation: $\Delta \mathbf{E}(\mathbf{r},\omega) + \frac{\omega^2}{c^2} \varepsilon(\mathbf{r},\omega) \mathbf{E}(\mathbf{r},\omega) = 0$

Ansatz for the fields:

$$\mathbf{E}(\mathbf{r},t) = \mathbf{E}_0(x,y) \exp\left(i\beta z - i\omega t\right)$$
$$\mathbf{H}(\mathbf{r},t) = \mathbf{H}_0(x,y) \exp\left(i\beta z - i\omega t\right)$$

For a 2D geometry & TE ($\mathbf{E}_0(x) = \begin{pmatrix} 0 \\ E_0 \\ 0 \end{pmatrix}$), inserting into Helmholtz eq. yields:

$$\boxed{\frac{1}{k_0^2} \frac{\partial^2 E_0(x,\omega)}{\partial x^2} + \varepsilon(x,\omega) E_0(x,\omega) = \varepsilon_{\text{eff}} E_0(x,\omega)}$$

$$\varepsilon_{\text{eff}} = \left(\frac{\beta}{k_0} \right)^2$$

Guided modes in 1+1D systems (TE modes)

This eigenvalue equation has to be solved for PEC, i.e. perfectly electric conducting boundaries ($E_0(x_{\min}) = E_0(x_{\max}) = 0$).

The mode operator is defined as: $L_{\mathrm{TE}} = \frac{1}{k_0^2}\frac{\partial^2}{\partial x^2} + \varepsilon(x,\omega)$

Numerical solution by discretizing functions and operators:

Discretizing transverse space: $x_j = x_{\min}, x_{\min} + h, \ldots, x_{\max}$

Discretizing the E-fields: $E_j = E_0(x_j, \omega)$

Discretizing 2nd order derivative: $\left.\frac{\partial^2 f}{\partial x^2}\right|_{x_j} \approx \frac{f(x_{j+1}) - 2f(x_j) + f(x_{j-1})}{h^2}$

➔ matrix diagonal: $-2/h^2$ ($+k_0^2\varepsilon$)
adjacent diagonals: $1/h^2$

Guided modes in 1+1D systems (matrix form)

$$\frac{1}{k_0^2}\begin{pmatrix} -\frac{2}{h^2}+k_0^2\varepsilon_1 & 1/h^2 & 0 & 0 & 0 & 0 & \cdots \\ 1/h^2 & -\frac{2}{h^2}+k_0^2\varepsilon_2 & 1/h^2 & 0 & 0 & 0 & \cdots \\ 0 & 1/h^2 & -\frac{2}{h^2}+k_0^2\varepsilon_3 & 1/h^2 & 0 & 0 & \cdots \\ 0 & 0 & \ddots & \ddots & \ddots & 0 & \cdots \end{pmatrix}\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{pmatrix} = \varepsilon_{\mathrm{eff}}\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \end{pmatrix}$$

Task I: Calculate TE eigenmodes of a film waveguide

```
def guided_modes_1DTE(prm, k0, h):
    """Computes the effective permittivity of a TE polarized guided eigenmode.
    All dimensions are in µm.
    Note that modes are filtered to match the requirement that
    their effective permittivity is larger than the substrate (cladding).

    Parameters
    ----------
    prm : 1d-array
        Dielectric permittivity in the x-direction
    k0 : float
        Free space wavenumber
    h : float
        Spatial discretization

    Returns
    -------
    eff_eps : 1d-array
        Effective permittivity vector of calculated modes
    guided : 2d-array
        Field distributions of the guided eigenmodes
    """
    pass
```

Task I: Calculate TE eigenmodes of a film waveguide

Input variables: permittivity profile; h, k_0

assume Gaussian waveguide profile

$$\varepsilon(x,\omega) = \varepsilon_{\text{Substrate}} + \Delta\varepsilon e^{-(x/W)^2}$$

Use the following parameters for testing:

ε = 2.25, Δε = 0.015, W = 15 µm, λ = 780 nm

Please select the modes guided inside the dielectric waveguide according to their eigenvalue:

$$\varepsilon_{\text{Substrate}} < \varepsilon_{\text{eff}} < \varepsilon(x)_{\max}$$

Guided modes in 2+1 (=3D) systems (strip waveguide)

in scalar approximation

scalar Helmholtz equation ➔ eigenvalue problem for scalar fields

$$\left[\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right]u(x,y)+\left[k_0^2\varepsilon(x,y,\omega)-\beta^2(\omega)\right]u(x,y)=0$$

transform to standard notation

$$\left[\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}+k_0^2\varepsilon(x,y,\omega)\right]u(x,y)=\beta^2(\omega)u(x,y)$$

This eigenvalue problem is to be solved by a finite difference scheme using electric conducting boundaries.

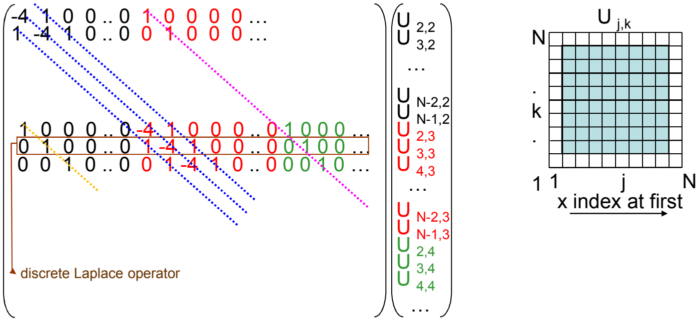
Discretizing 2D 2nd order derivative:

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{x_j,y_k}+\left.\frac{\partial^2 f}{\partial y^2}\right|_{x_j,y_k}\approx\frac{f(x_{j+1},y_k)+f(x_{j-1},y_k)+f(x_j,y_{k+1})+f(x_j,y_{k-1})-4f(x_j,y_k)}{h^2}$$

➔ matrix diagonal: $-4/h^2$ & adjacent diagonals: $1/h^2$

➔ 2 more bands with $1/h^2$

Numerical implementation of the 2D Laplace operator



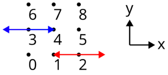
Differentiation with respect to x (same as 1D)

Differentiation with respect to y

Use the concept of sparse matrices since the matrix will be mainly filled with zeros.

Boundary conditions implementation examples

Perfect electric conductor
boundary conditions
in x-direction

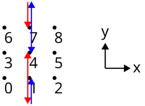


$$\frac{\partial^2}{\partial x^2} = \frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \textcircled{1} & -2 & \textcircled{0} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \textcircled{0} & -2 & \textcircled{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 \end{bmatrix}$$

nearest neighbour in x-direction

main diagonal: point itself

Periodic
boundary conditions
in y-direction



$$\frac{\partial^2}{\partial y^2} = \frac{1}{\Delta y^2}\begin{bmatrix} -2 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2 & 0 & 0 & \textcircled{1} & 0 & 0 & \textcircled{1} & 0 \\ 0 & 0 & -2 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -2 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -2 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & -2 & 0 & 0 \\ 0 & \textcircled{1} & 0 & 0 & \textcircled{1} & 0 & 0 & -2 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -2 \end{bmatrix}$$

nearest neighbour (y-dir) on the opposite side of domain

nearest neighbour in y-direction

main diagonal: point itself

Task II: Quasi-TE modes of a strip waveguide

```python
def guided_modes_2D(prm, k0, h, numb):
    """Computes the effective permittivity of a quasi-TE polarized guided
    eigenmode. All dimensions are in µm.

    Parameters
    ----------
    prm  : 2d-array
        Dielectric permittivity in the xy-plane
    k0 : float
        Free space wavenumber
    h : float
        Spatial discretization
    numb : int
        Number of eigenmodes to be calculated

    Returns
    -------
    eff_eps : 1d-array
        Effective permittivity vector of calculated eigenmodes
    guided : 3d-array
        Field distributions of the guided eigenmodes
    """
    pass
```

Task II 2D mode solver test parameters

Step index optical fiber (These fibers consist of a pure SiO_2 cladding and a core made of SiO_2 doped with a small amount of GeO_2. They are usually used around wavelengths 1300 - 1600 nm where they only have a single mode. Here we will use a shorter wavelength to see more guided modes.)

- $\epsilon(x, y) = \epsilon_{\text{silica}} + 0.01$ if $x^2 + y^2 \leq r_{\text{core}}^2$ with $\epsilon_{\text{silica}} = 2.11$
- $\epsilon(x, y) = \epsilon_{\text{silica}}$ otherwise
- $r_{\text{core}} = 6\,\mu\text{m}$
- $\lambda_0 = 780$ nm
- Make domain large enough so the fields do not touch the edge.
- Grid step size: make it smaller until fields do not change.
- There should be six guided modes.

Task III*: Group velocity and dispersion

Study the group velocity $v_g = \partial\omega/\partial\beta$ of the step index optical fiber from Task II. Pick the fundamental guided mode with the highest β and calculate v_g in the wavelength range 1.2 µm – 1.6 µm. Use realistic dispersion for ϵ_{silica}, given by the Sellmeier equation

```
def eps_SiO2(wl):
    # 'wl' wavelength in micrometers
    # Malitson J. Opt. Soc. Am. 55, 1205, 1965
    return (1 + 0.6961663*wl**2/(wl**2 - 0.0684043**2) +
           0.4079426*wl**2/(wl**2 - 0.1162414**2) +
           0.8974794*wl**2/(wl**2 - 9.896161**2))
```

Knowing that the dispersion parameter is defined as

$$D = \frac{\partial(1/v_g)}{\partial\lambda_0} = -\frac{\omega^2}{2\pi c}\frac{\partial^2\beta}{\partial\omega^2},$$

can you find the zero dispersion wavelength of this fiber?

Hint: if you want to avoid headaches, use equally-spaced *frequencies* to span the given wavelength range and perform the derivatives numerically with respect to angular frequency.

Task IV*: Periodic boundary conditions

Implement periodic boundary conditions for the 2D mode solver of Task II, along one of the coordinate directions (either x or y but not both). Check the result by reproducing the Task I result in this 2D geometry.