Computational Photonics

Outro

A practical guide to Computational Photonics

Strategies to solve Maxwell's equations I

- frequency-domain eigenproblems

  ⟹ elementary solution to Maxwell's equations without source

  $$\omega = \omega(\mathbf{k})$$

  dispersion relation

  $$Ax = \omega^2 Bx$$

  field profile

- frequency-domain responses

  time harmonic current source $\mathbf{J}(\mathbf{r})e^{-i\omega t}$

  ⟹ field all electric and magnetic fields

- time-domain responses

  current source $\mathbf{J}(\mathbf{r}, t)$

  ⟹ field all electric and magnetic fields

Strategies to solve Maxwell's equations II

- finite differences: represent functions in space at discrete points and derivatives as differences

$$f(x) \;\rightarrow\; f_n \approx f(n\Delta x)$$

$$\frac{\partial f}{\partial x} \;\rightarrow\; \frac{\partial f_n}{\partial x} \approx \frac{f_{n+1} - f_{n-1}}{2\Delta x}$$

- finite elements: divide space into finite regions and expand solution in these regions

- spectral methods: represent functions as a series expansion in a complete basis set, truncate this series

- boundary-element method: discretize space at interfaces, analytical expressions in homogenous space

exploiting symmetries in geometry in the solution strategy

Calculating a band structure

MIT Photonic-Bands (MPB)

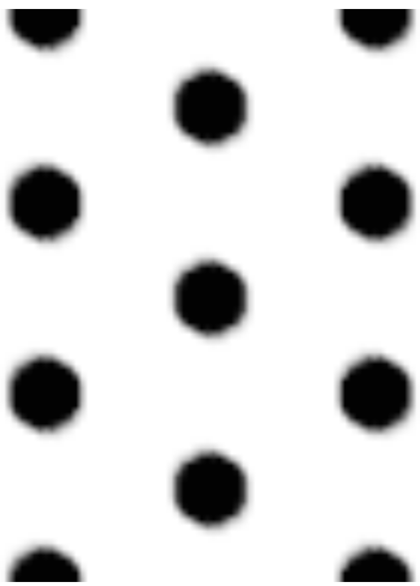
http://ab-initio.mit.edu/wiki/index.php/MIT_Photonic_Bands

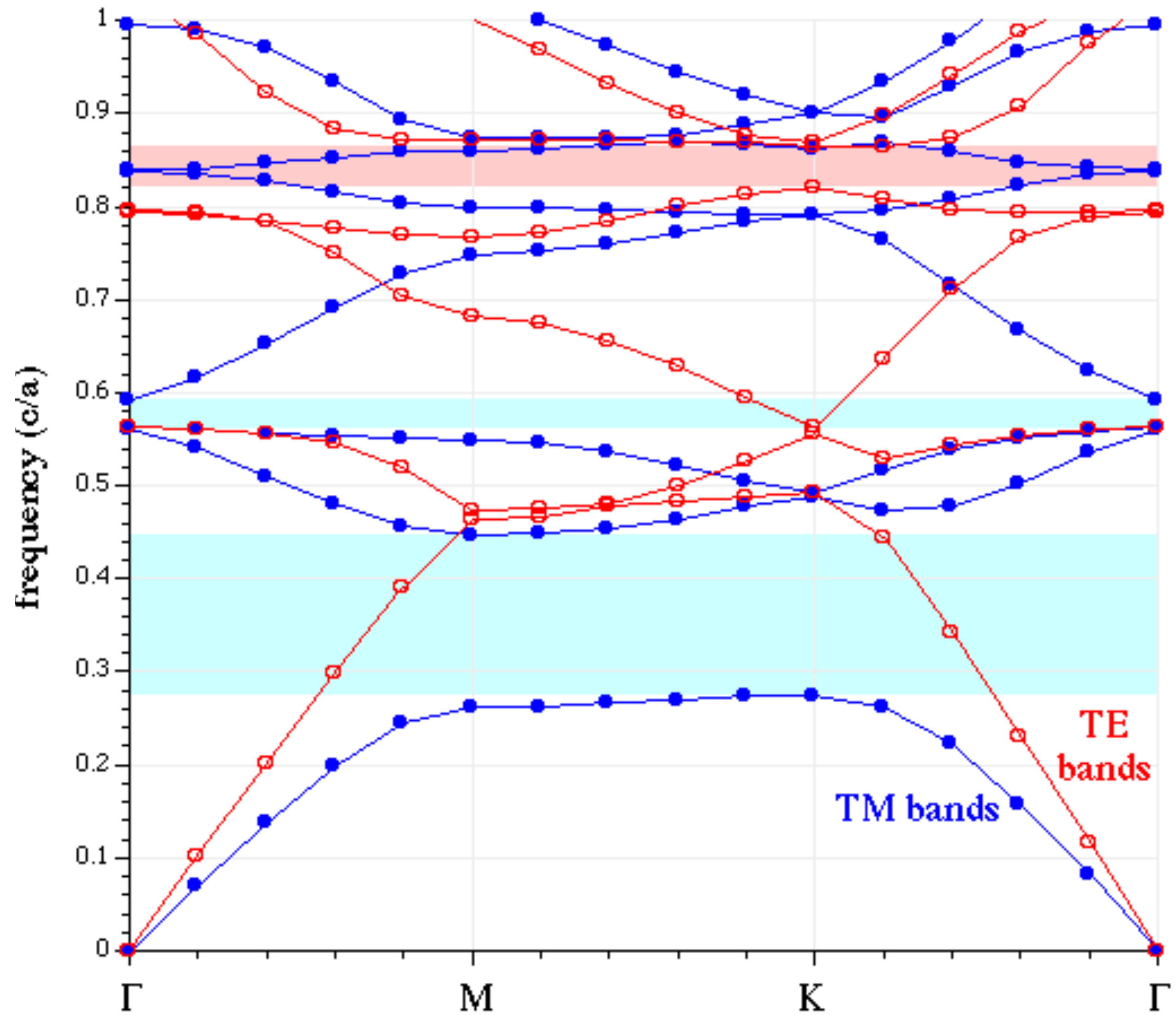
- frequency domain
- plane wave expansion, eigenvalue solver
- photonic crystals, waveguides, integrated optics
- guile scheme for programming a task
- isotropic, anisotropic materials, no dispersion
- visualisation using different h5 utilities

Example MPB

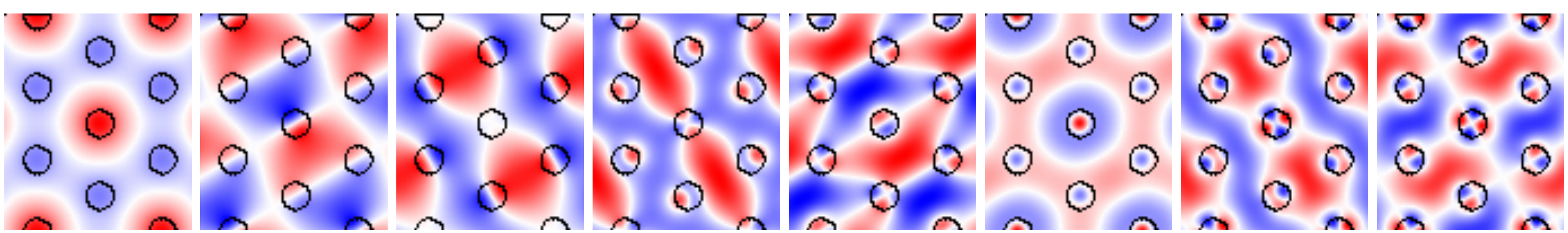
```
(set! num-bands 8)

(set! geometry-lattice (make lattice (size 1 1 no-size)
                         (basis1 (/ (sqrt 3) 2) 0.5)
                         (basis2 (/ (sqrt 3) 2) -0.5)))
(set! geometry (list (make cylinder
                       (center 0 0 0) (radius 0.2) (height infinity)
                       (material (make dielectric (epsilon 12))))))

(set! k-points (list (vector3 0 0 0)          ; Gamma
                     (vector3 0 0.5 0)        ; M
                     (vector3 (/ -3) (/ 3) 0) ; K
                     (vector3 0 0 0)))        ; Gamma
(set! k-points (interpolate 4 k-points))

(set! resolution 32)

(run-tm (output-at-kpoint (vector3 (/ -3) (/ 3) 0)
                          fix-efield-phase output-efield-z))
(run-te)
```

Example MPB





TM band 1 TM band 2 TM band 3 TM band 4 TM band 5 TM band 6 TM band 7 TM band 8



Finite-difference time-domain

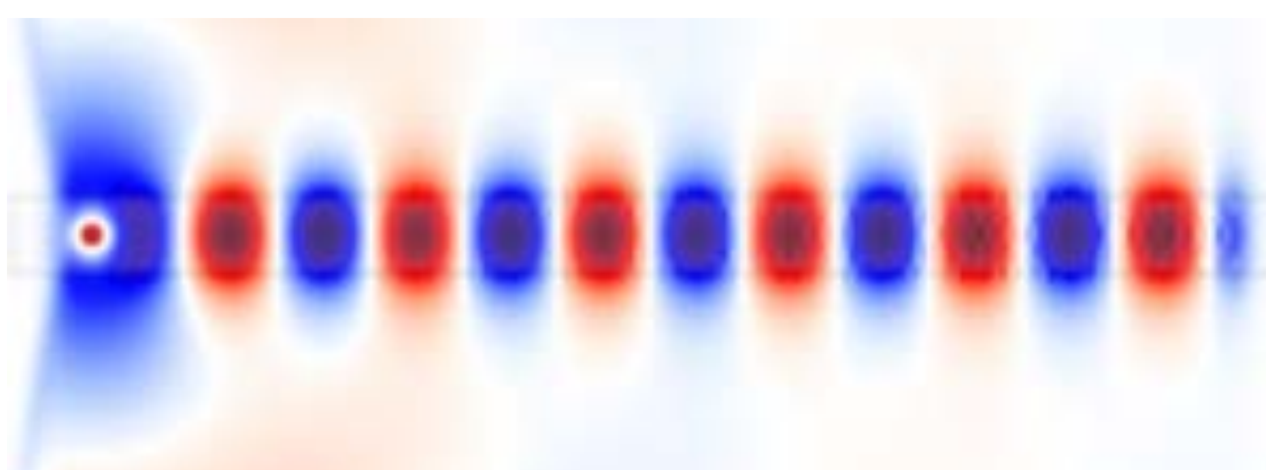
MEEP (also from MIT)

http://ab-initio.mit.edu/wiki/index.php/Meep

- time domain, 1d, 2d, 3d, cylindrical
- parallelised using MPI standard
- space and time discretisation and a leap-frog-algorithm
- basically everything when it comes to scattering and diffraction
  (less efficient for grating structures)
- guile scheme for programming a task
- isotropic, anisotropic materials, dispersion, nonlinear materials
- periodic and PML absorbing boundaries
- visualisation using different h5 utilities

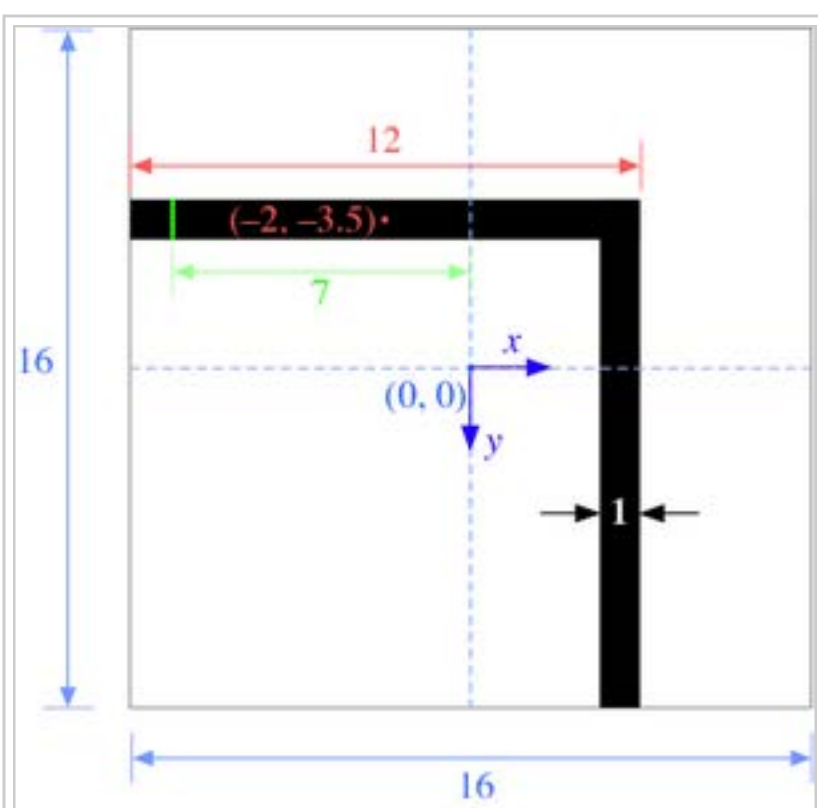
Example MEEP

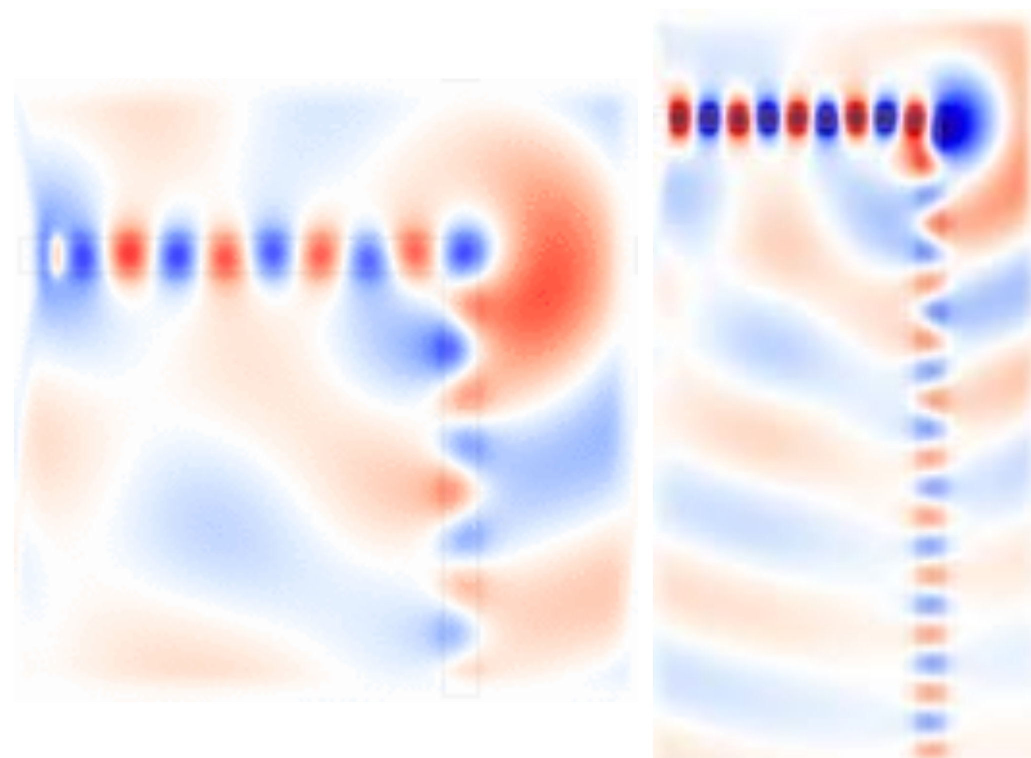
```
(set! geometry-lattice (make lattice (size 16 8 no-size)))

(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))
```



```
(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! pml-layers (list (make pml (thickness 1.0))))
(set! resolution 10)

(set! sources (list
               (make source
                 (src (make continuous-src
                        (wavelength (* 2 (sqrt 12))) (width 20)))
                 (component Ez)
                 (center -7 -3.5) (size 0 1))))
```

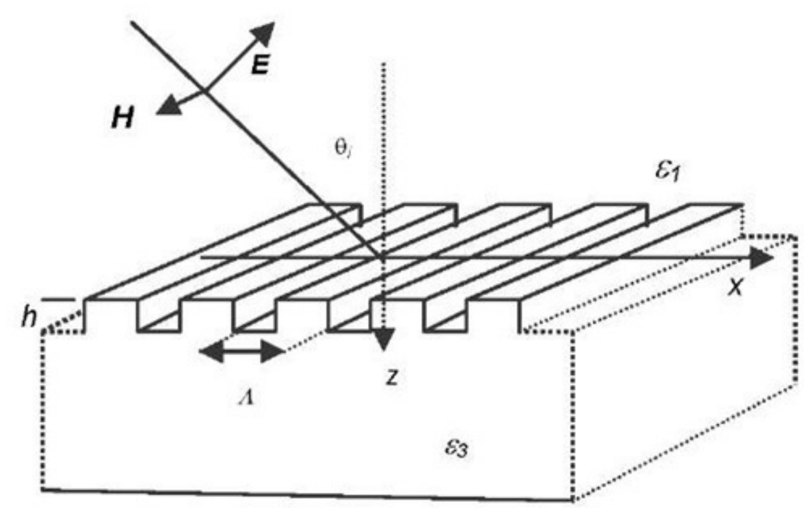
```
(run-until 200
           (at-beginning output-epsilon)
           (to-appended "ez" (at-every 0.6 output-efield-z)))
```





Grating algorithms

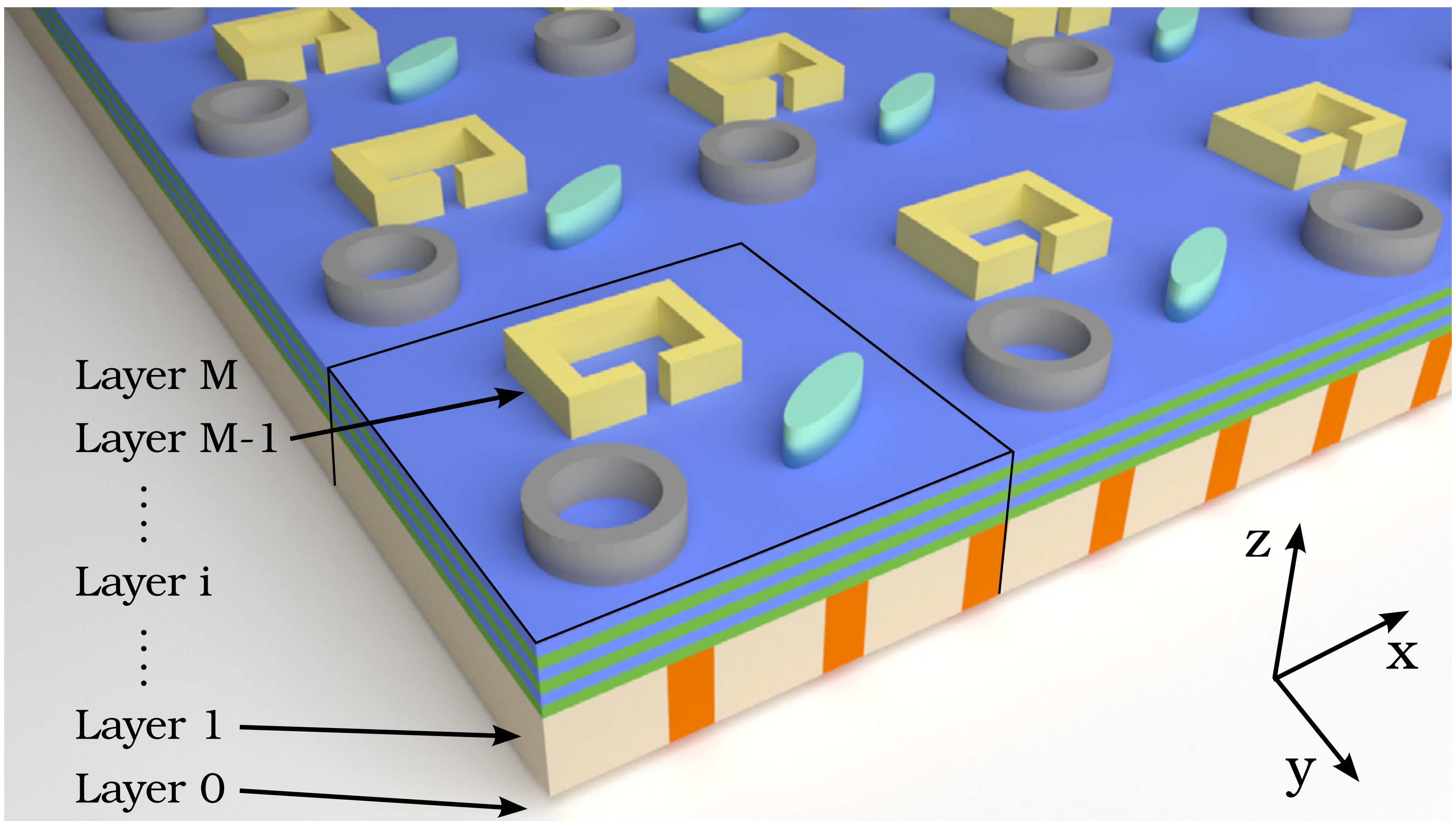
- based on rigorous coupled wave analysis (or Fourier Modal Method)
- frequency domain solver
- strictly periodic objects
- all linear materials usually considered, lossy and dispersive
- plane wave illumination
- many codes available, partially also free
- some provide also scattering matrices



EMUstack

http://www.physics.usyd.edu.au/emustack/

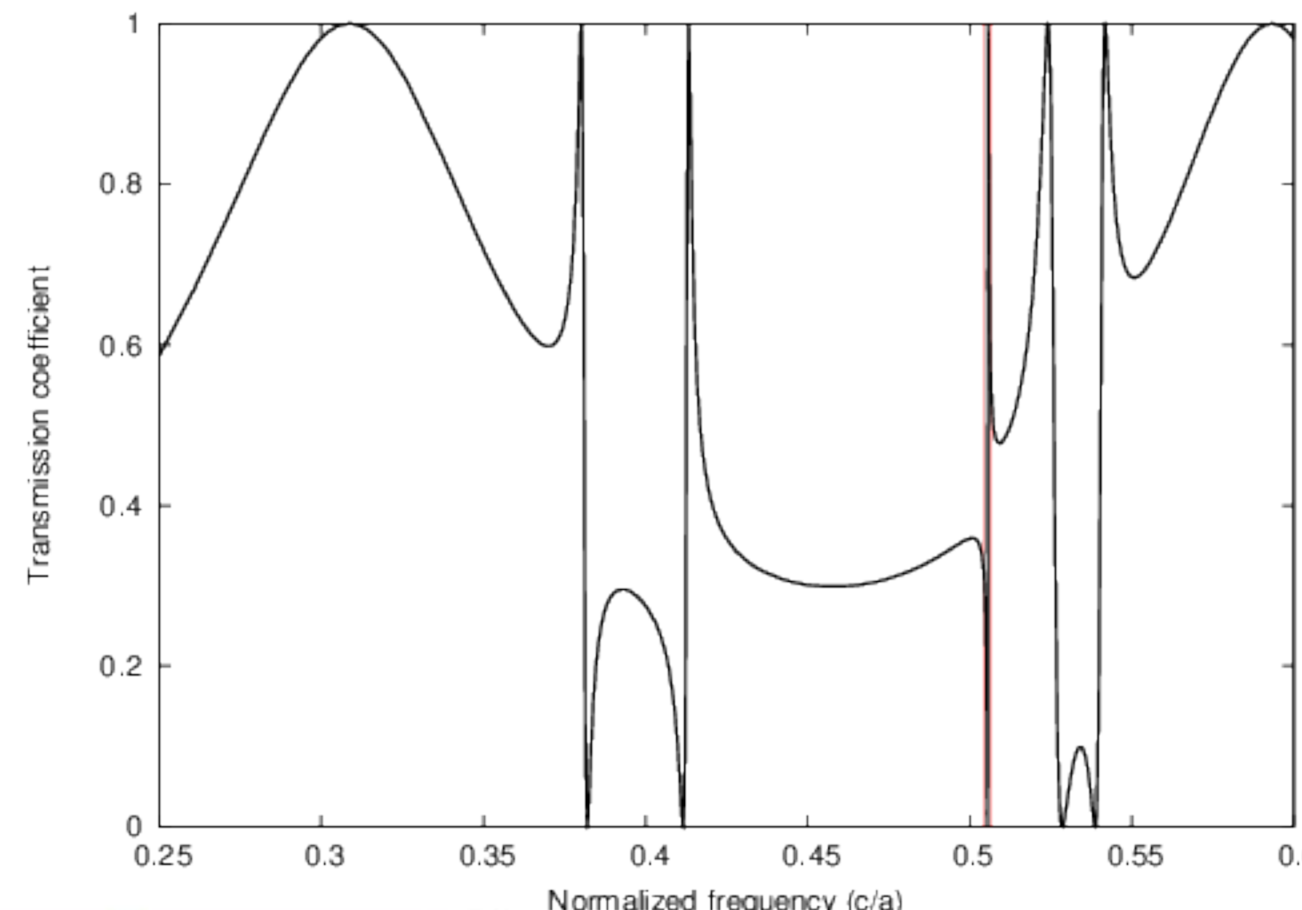
Computer Physics Communications 202 (2016) 276–286

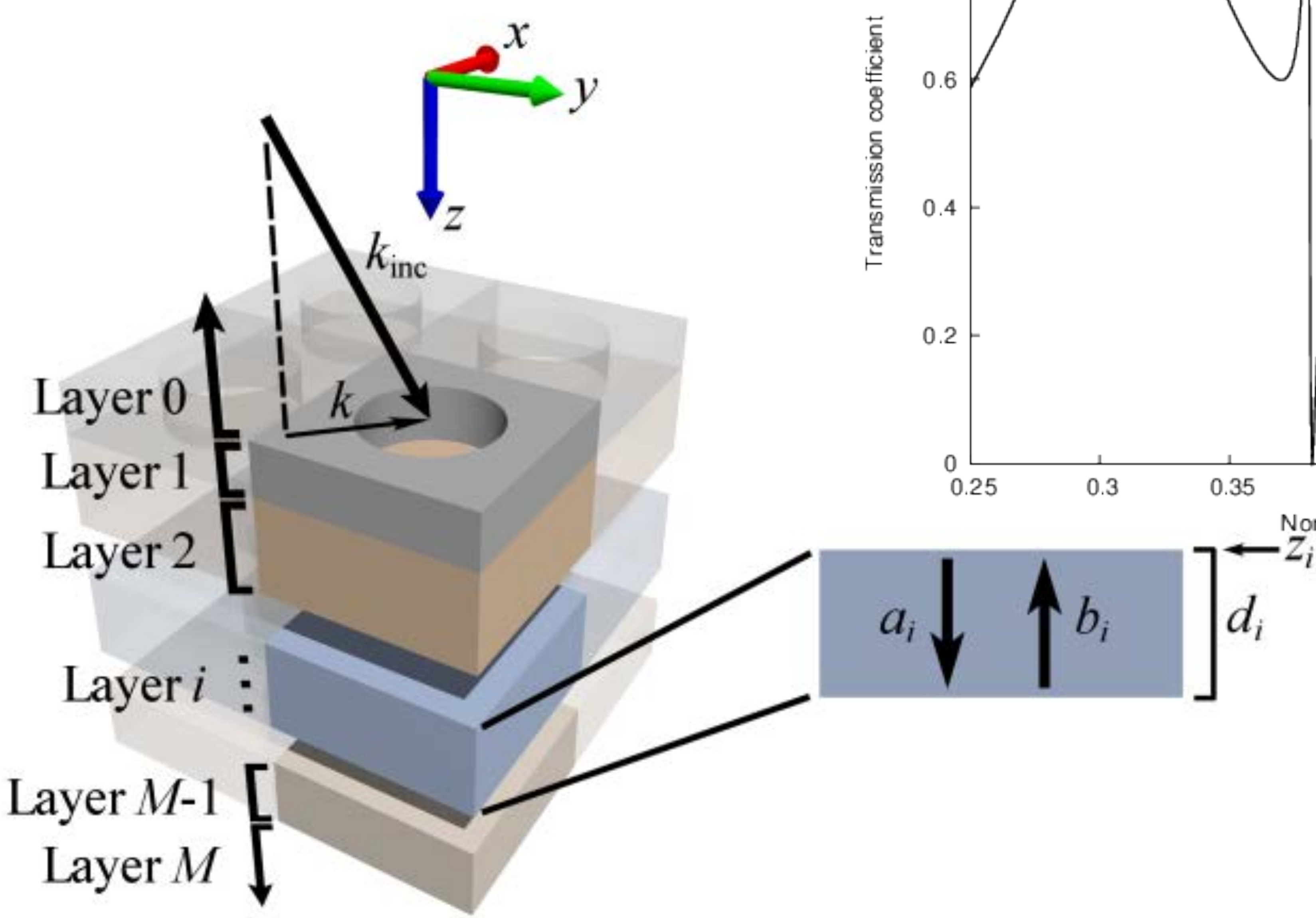


S4

https://web.stanford.edu/group/fan/S4/

Victor Liu and Shanhui Fan, “S[4]: A free electromagnetic solver for layered periodic structures,” Computer Physics Communications **183**, 2233-2244 (2012)



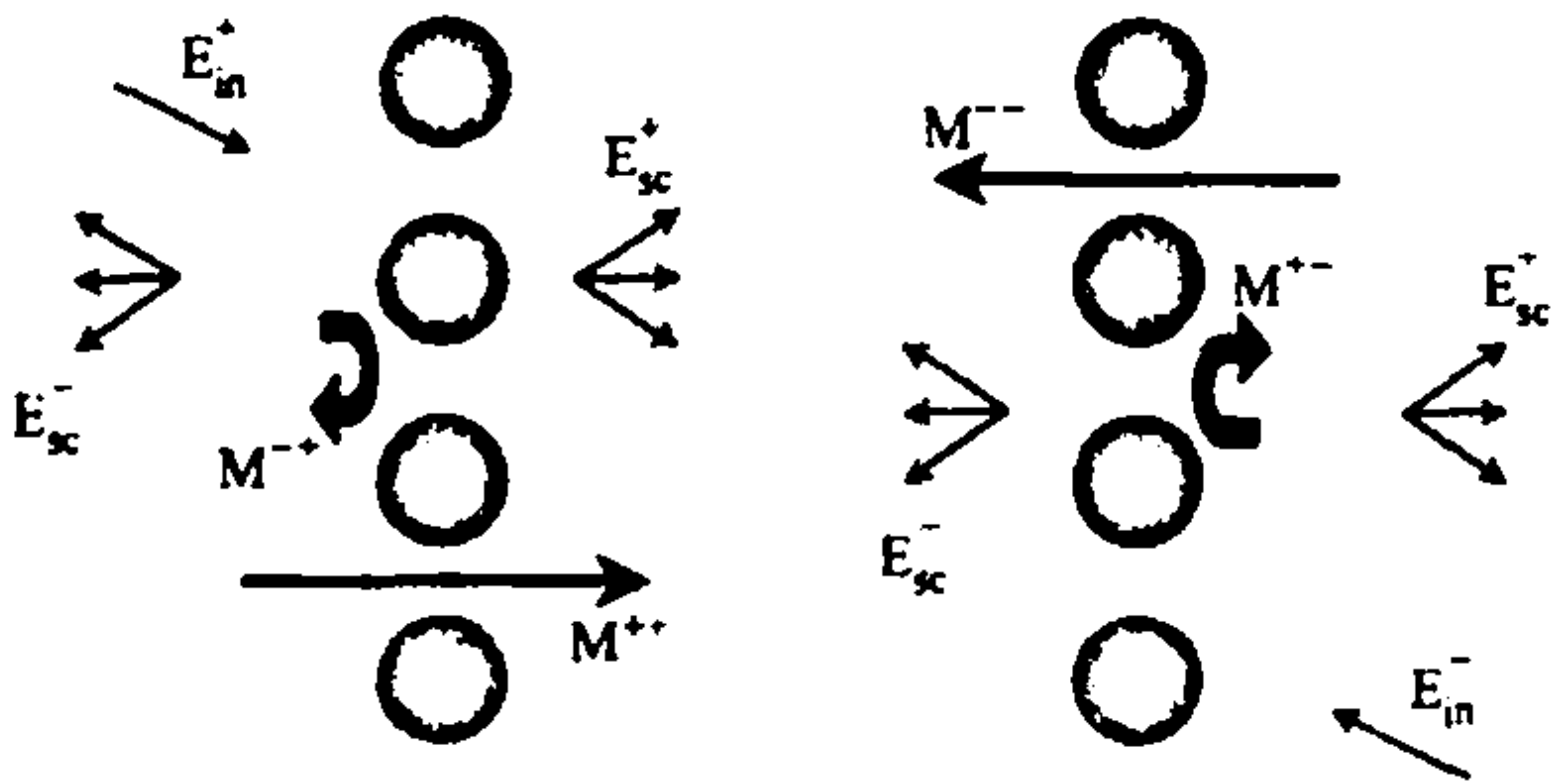


Resources on Mie Scattering http://www.scattport.org

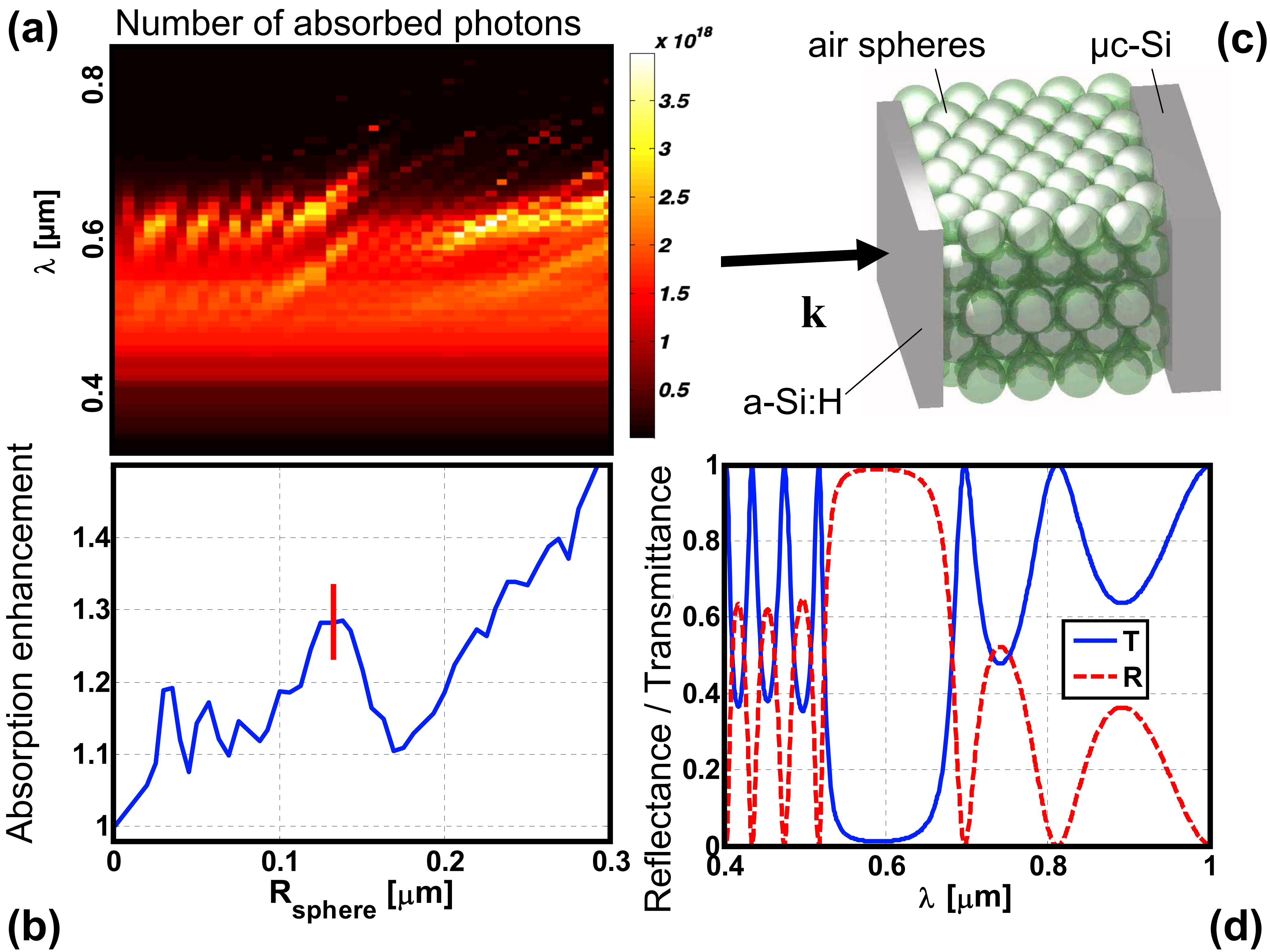
Resources on Mie Scattering

layered Korringa Kohn Rostoker method

Computer Physics Communications 113 (1998) 49–77



Resources on Mie Scattering

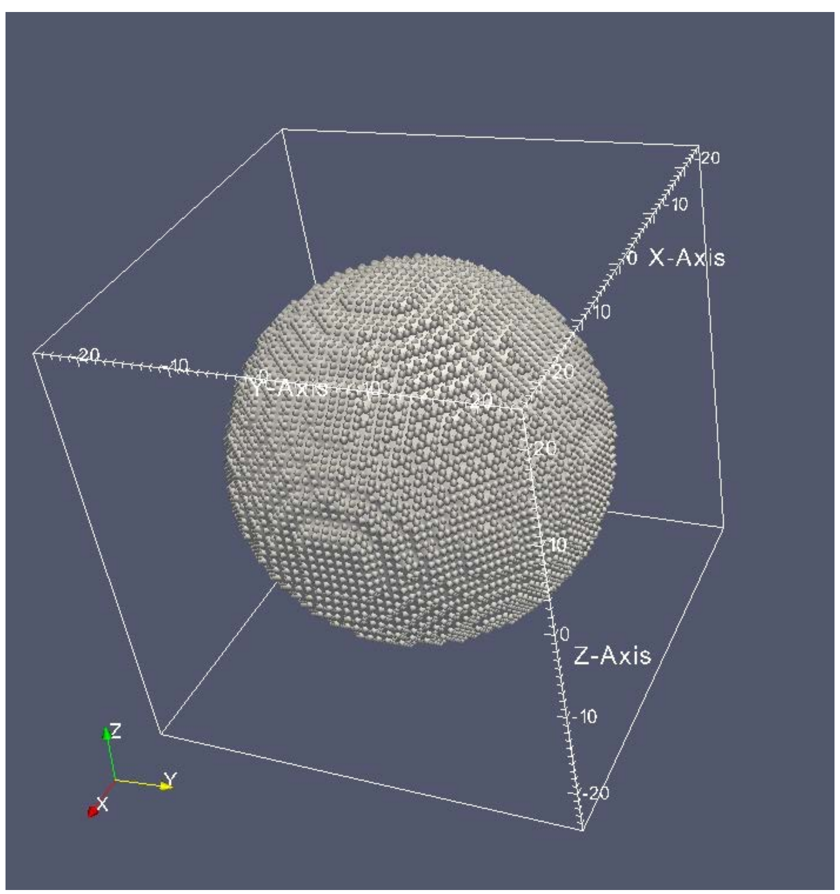


Resources on Mie Scattering

discrete dipole approximation

http://www.ddscat.org

<u>Draine, B.T., & Flatau, P.J., "Discrete dipole approximation for scattering calculations", J. Opt. Soc. Am. A, 11, 1491-1499 (1994)</u>



Resources on Mie Scattering

discrete dipole approximation

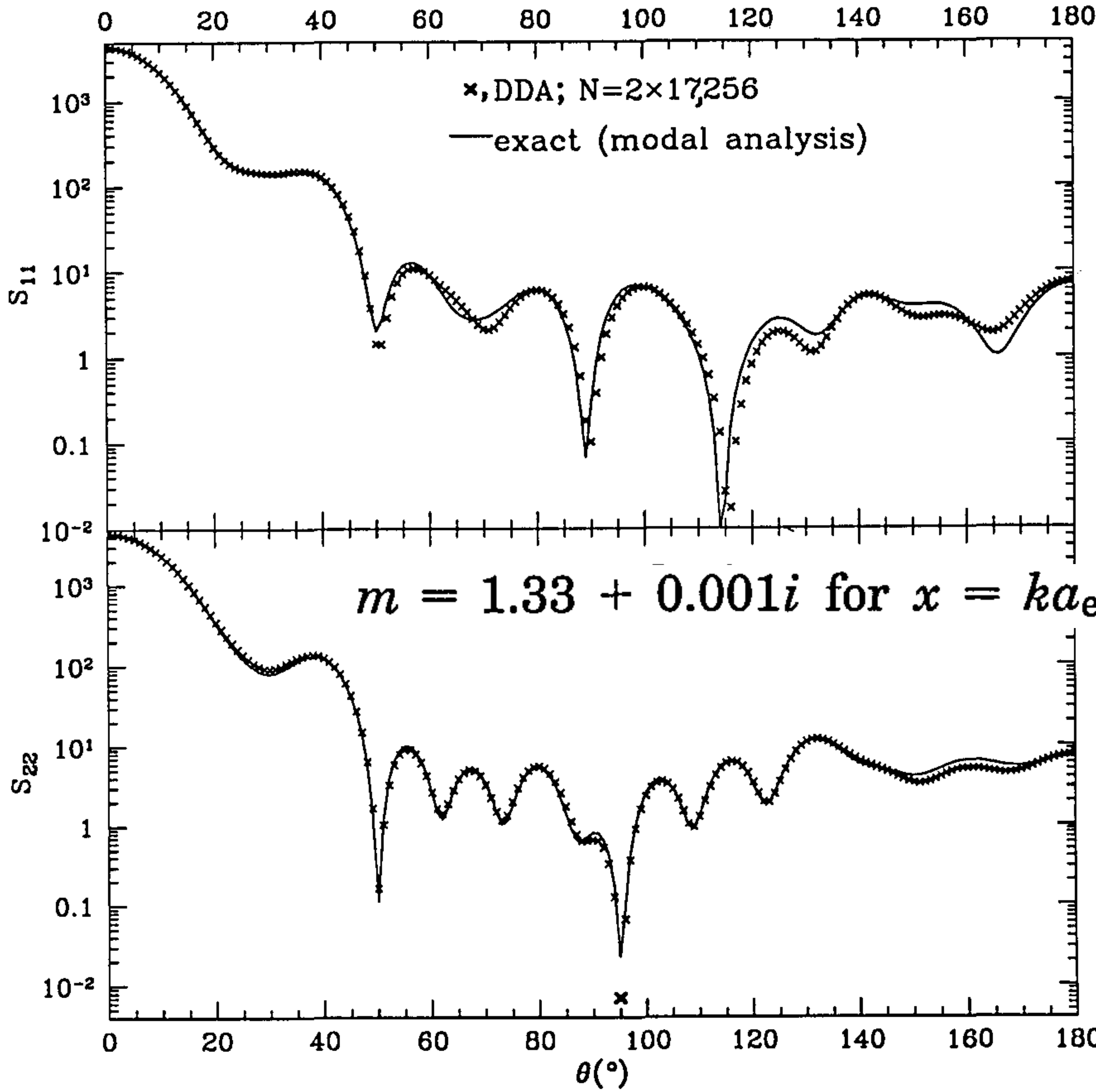
$$\alpha_j^{\mathrm{CM}} = \frac{3d^3}{4\pi}\frac{\epsilon_j - 1}{\epsilon_j + 2}$$

$$\mathbf{E}_j = \mathbf{E}_{\mathrm{inc},j} - \sum_{k \neq j} \mathbf{A}_{jk}\mathbf{P}_k$$

$$\mathbf{A}_{jk} = \frac{\exp(ikr_{jk})}{r_{jk}} \times \left[k^2(\hat{r}_{jk}\hat{r}_{jk} - \mathbf{1}_3) + \frac{ikr_{jk} - 1}{r_{jk}^2}(3\hat{r}_{jk}\hat{r}_{jk} - \mathbf{1}_3) \right]$$

Resources on Mie Scattering

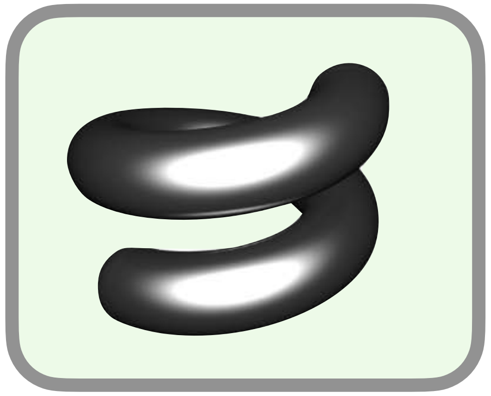
discrete dipole approximation



$$m = 1.33 + 0.001i \text{ for } x = ka_{\mathrm{eff}} = 10.$$

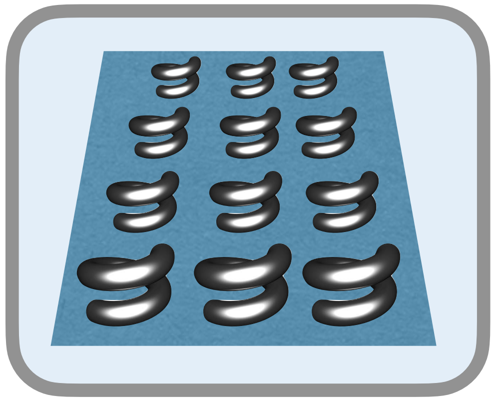
Resources on Mie Scattering

our own T-matrix based scattering code



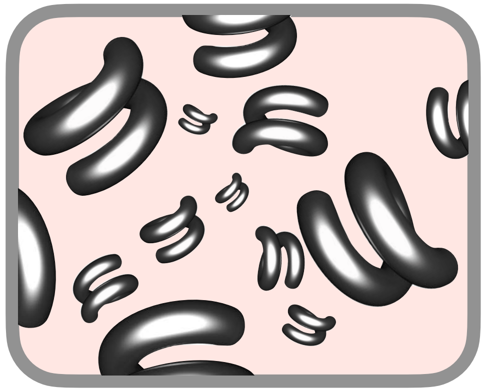
single particle

$$\mathbf{b}(\omega) = \mathbf{T}(\omega)\mathbf{q}(\omega)$$



particle in lattice

$$\mathbf{b} = \left(\mathbb{1} - \boldsymbol{T} {\sum_{\boldsymbol{R}}}' \boldsymbol{C}^{(3)}(-\boldsymbol{R})\mathrm{e}^{\mathrm{i}\boldsymbol{k}_\parallel \boldsymbol{R}}\right)^{-1} \boldsymbol{T}\mathbf{q}$$



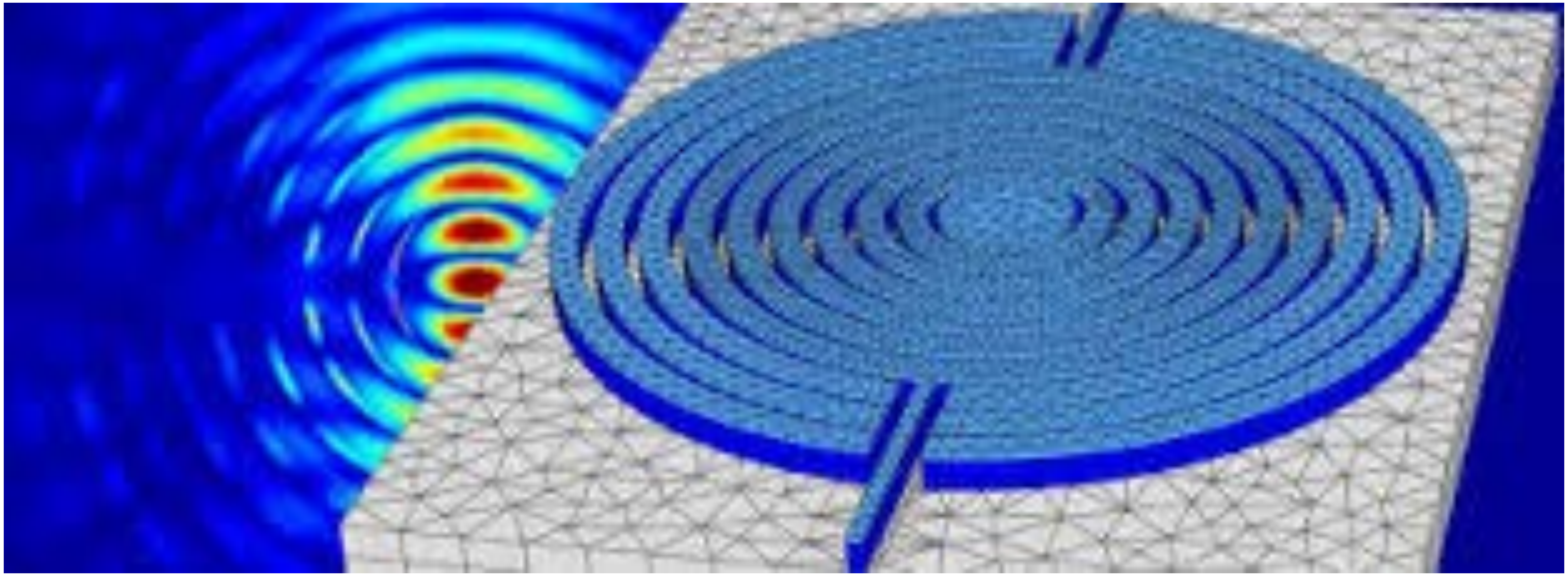
disordered particles

$$\mathbf{b}_{\text{local}} = (\mathbb{1} - \boldsymbol{T}_{\text{local}}\boldsymbol{C}^{(3)}{}_{\text{local}})^{-1}\boldsymbol{T}_{\text{local}}\mathbf{q}_{\text{local}}$$

https://tfp-photonics.github.io/treams

Finite Element Method

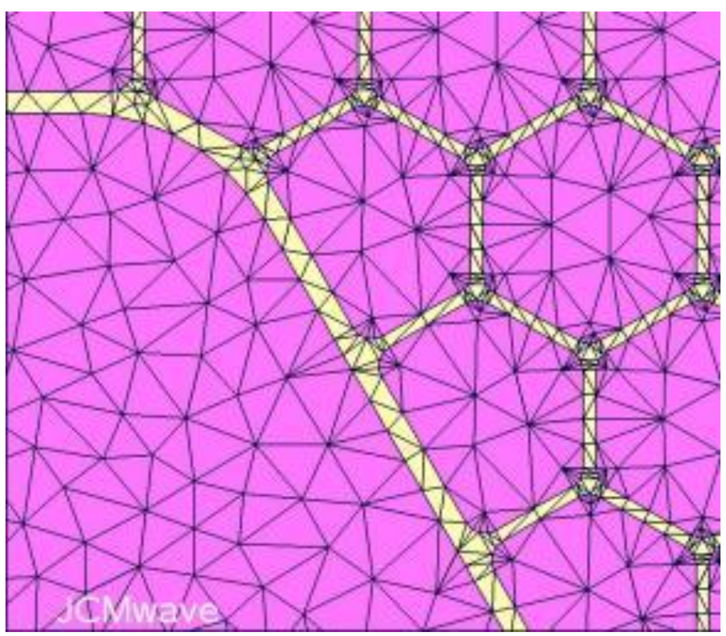
JCM suite

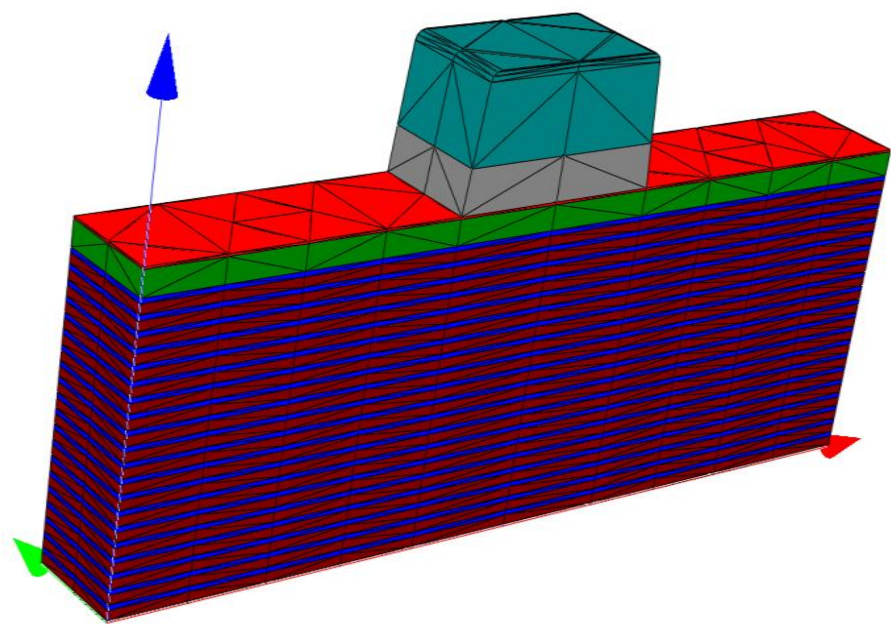
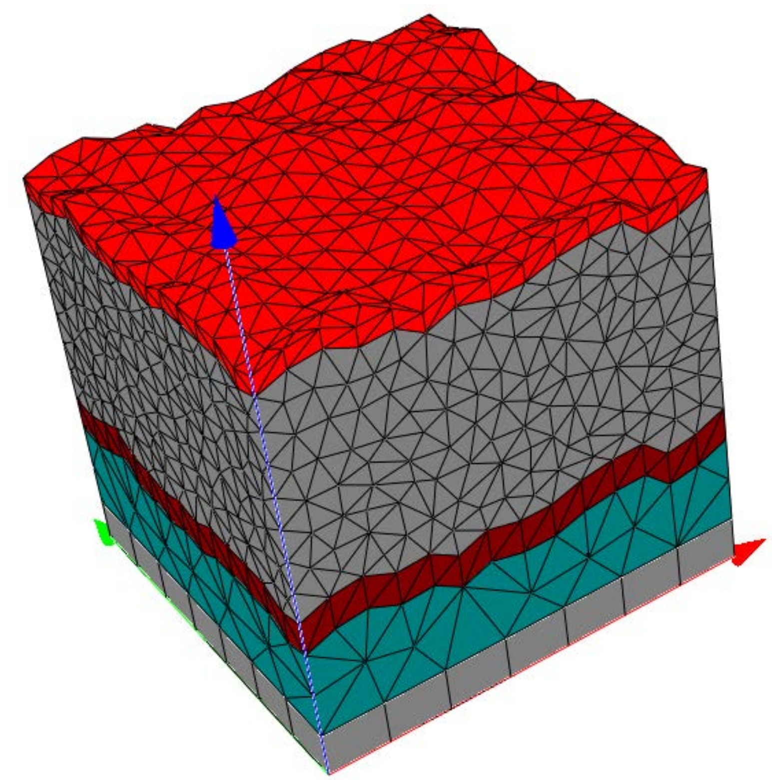


material taken from slides from S. Burger (JCM)

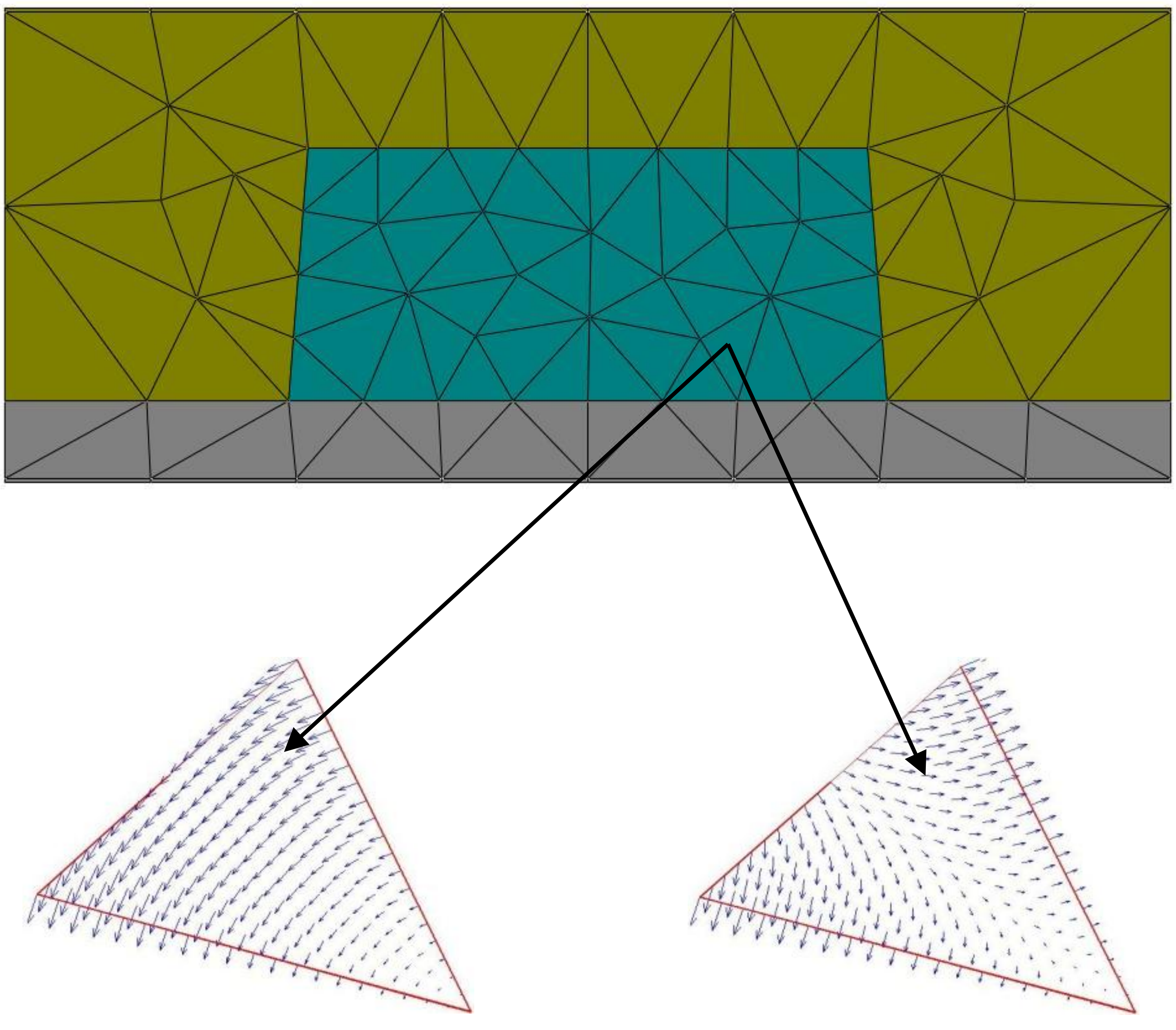
Finite Element Method

JCM suite

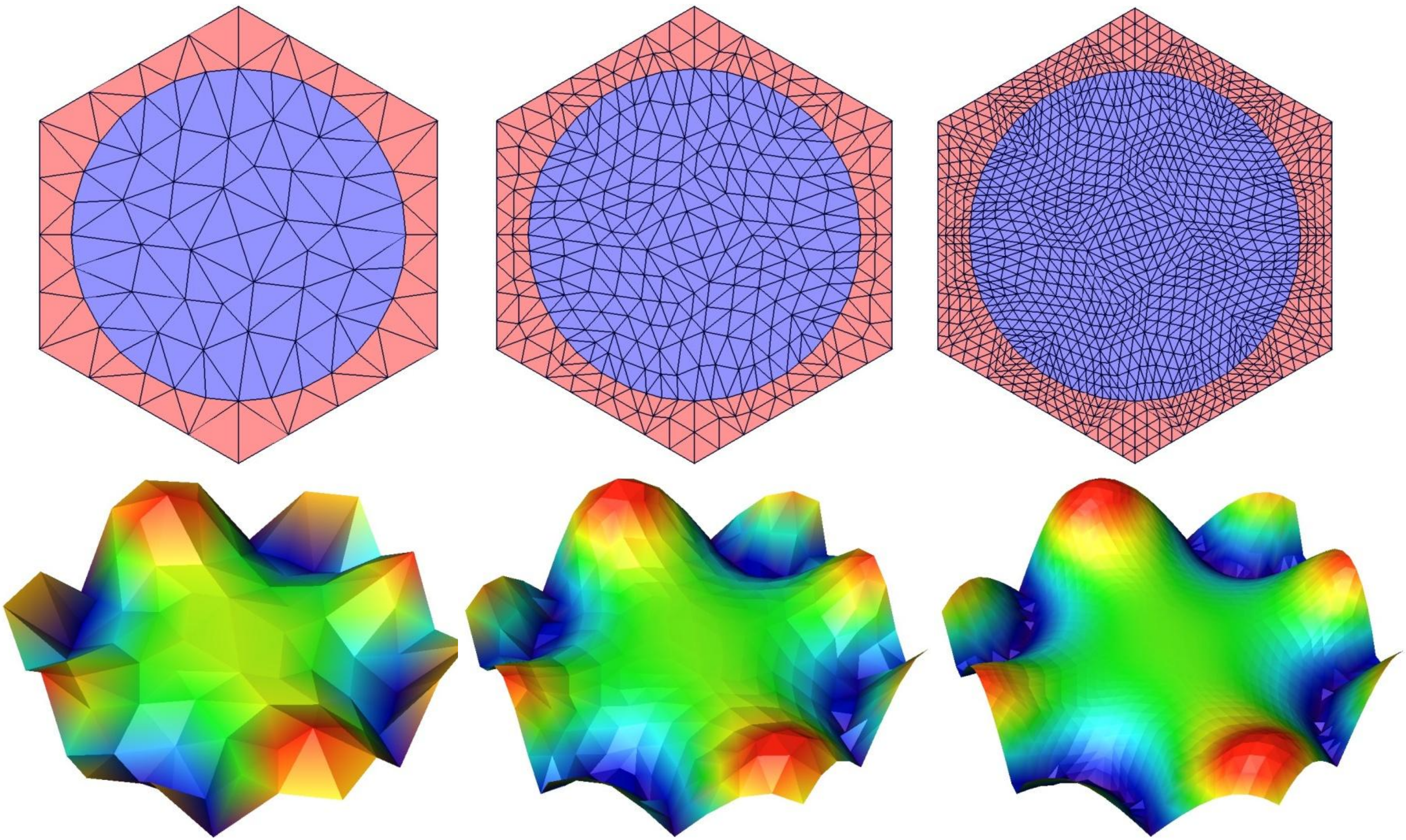


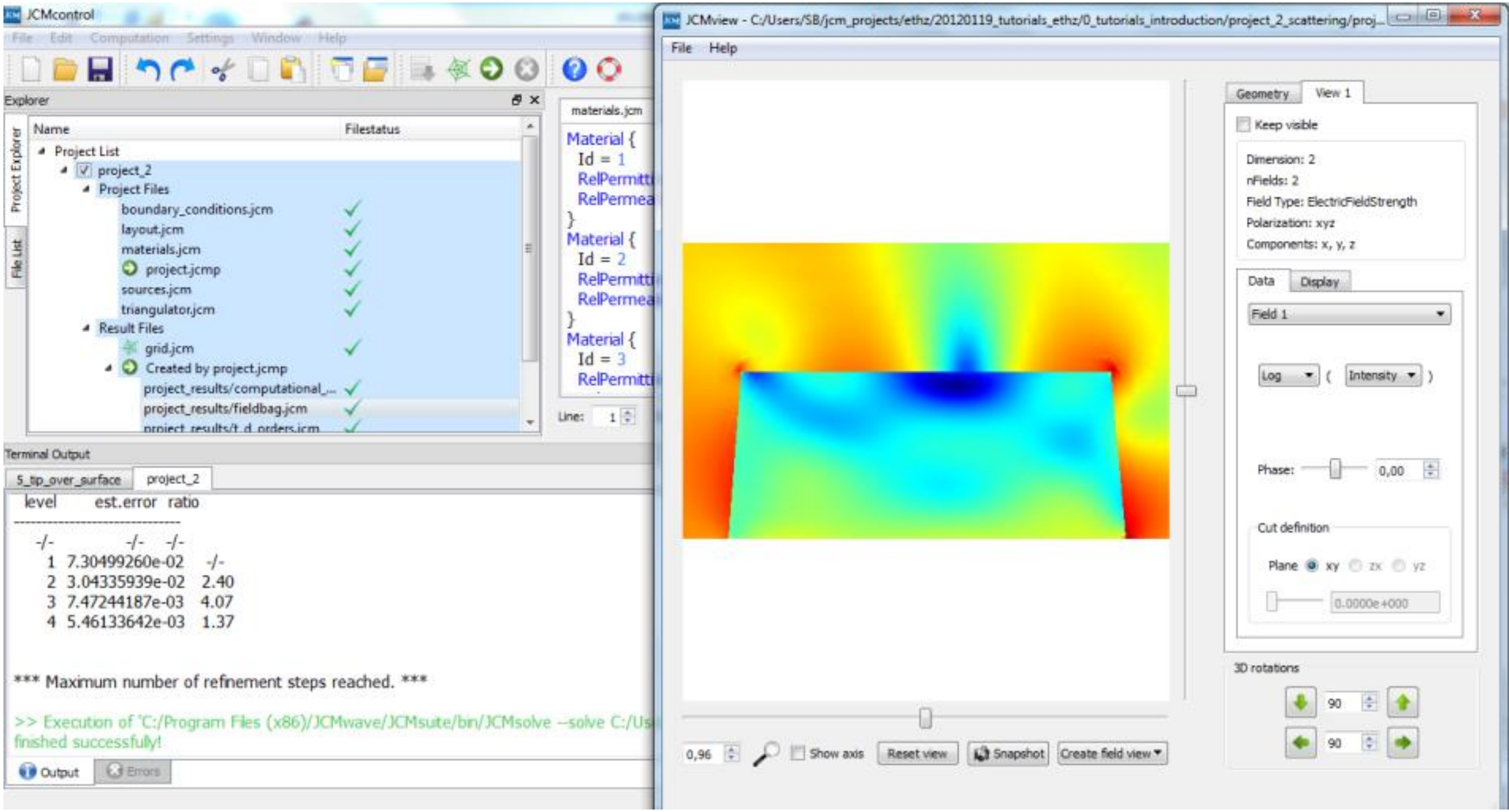
Finite Element Method

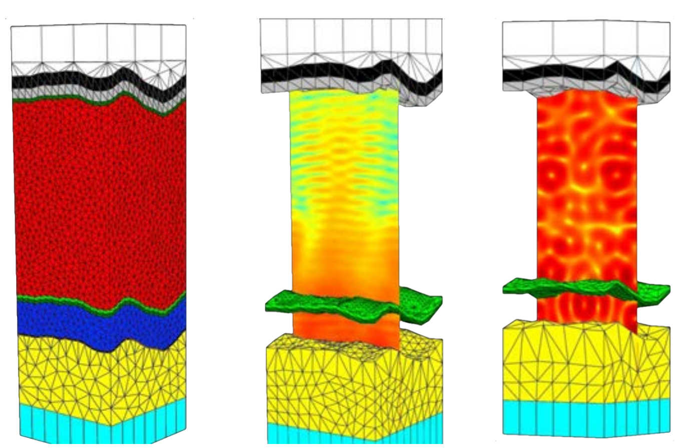


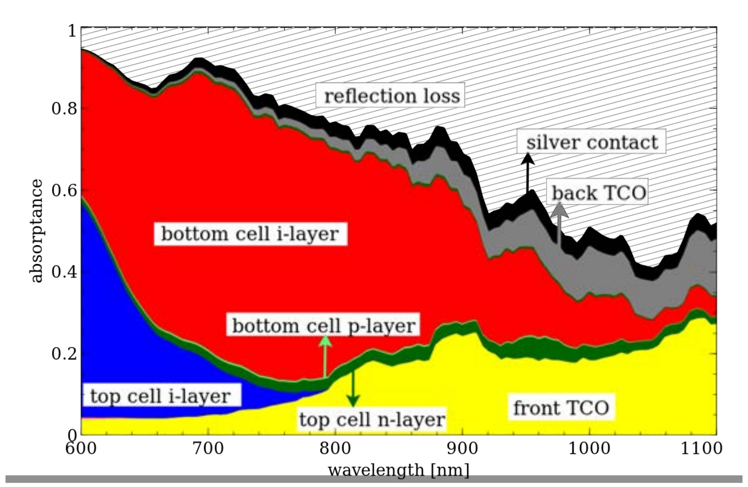
Finite Element Method



Finite Element Method



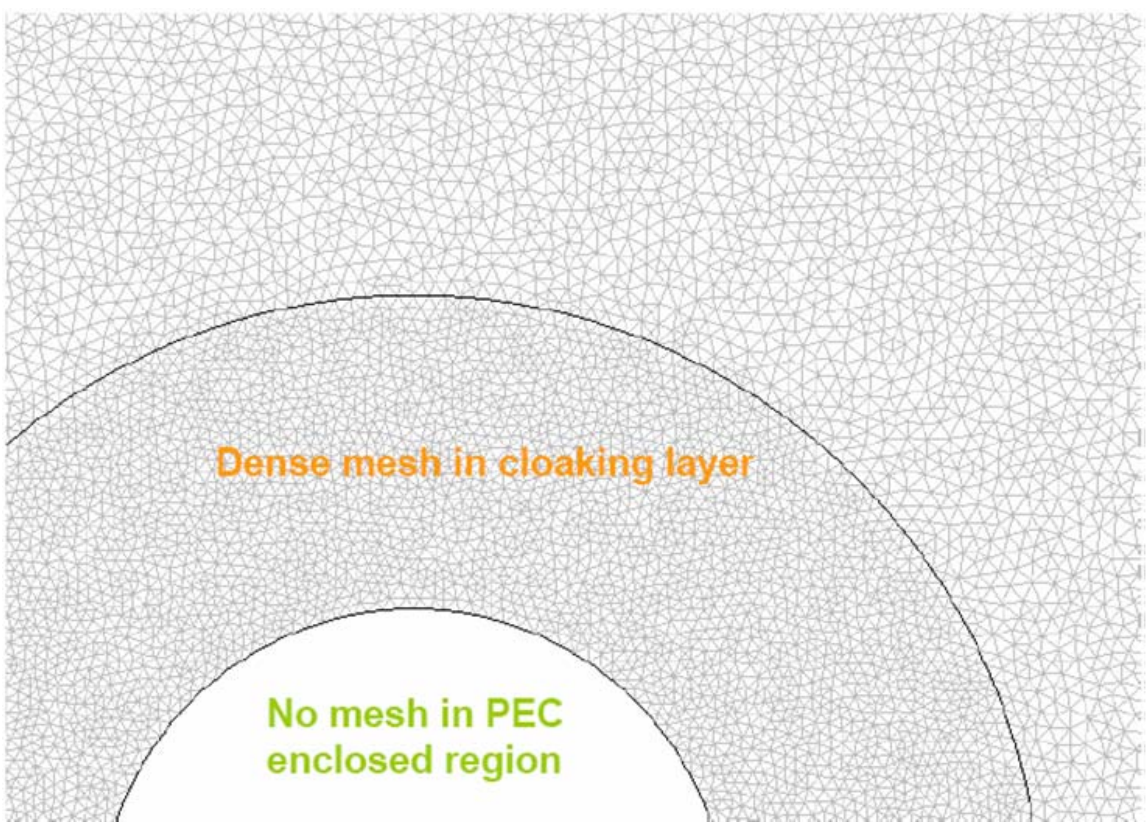


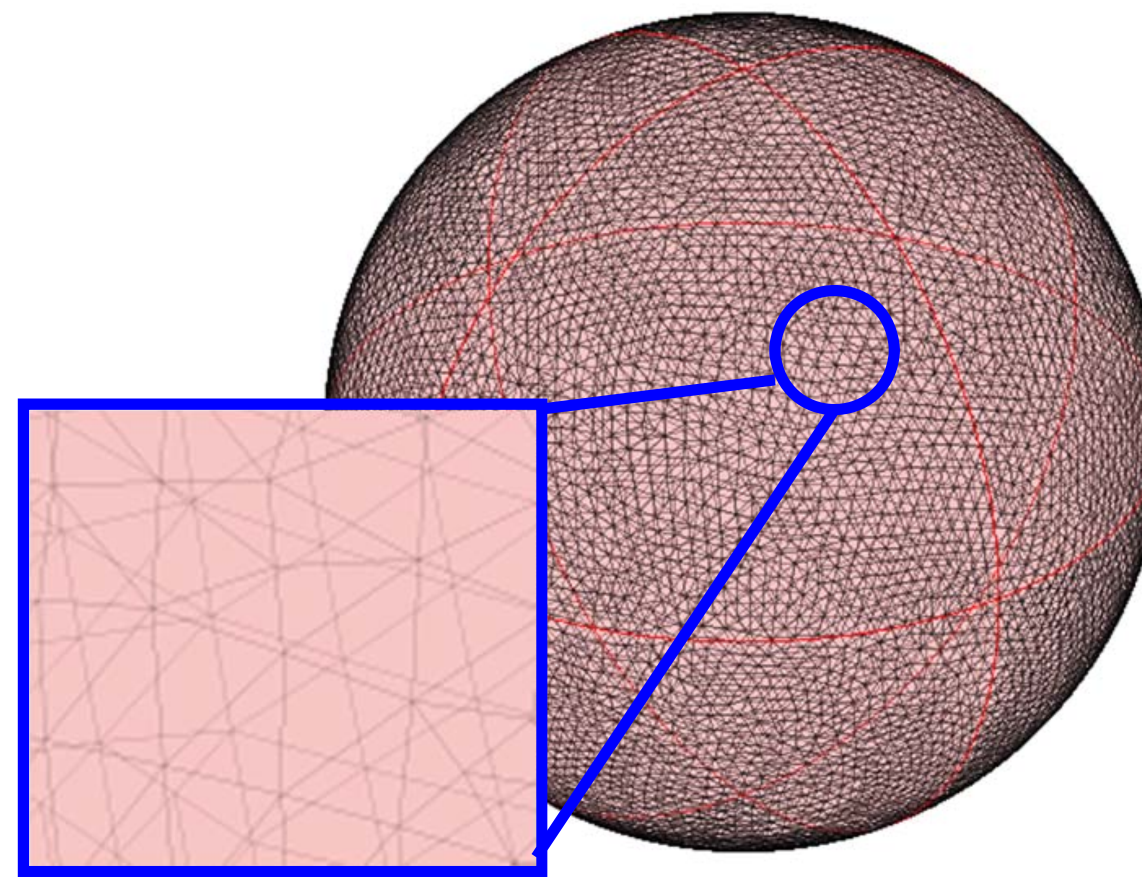


Finite Element Method

COMSOL

COMSOL multi physics

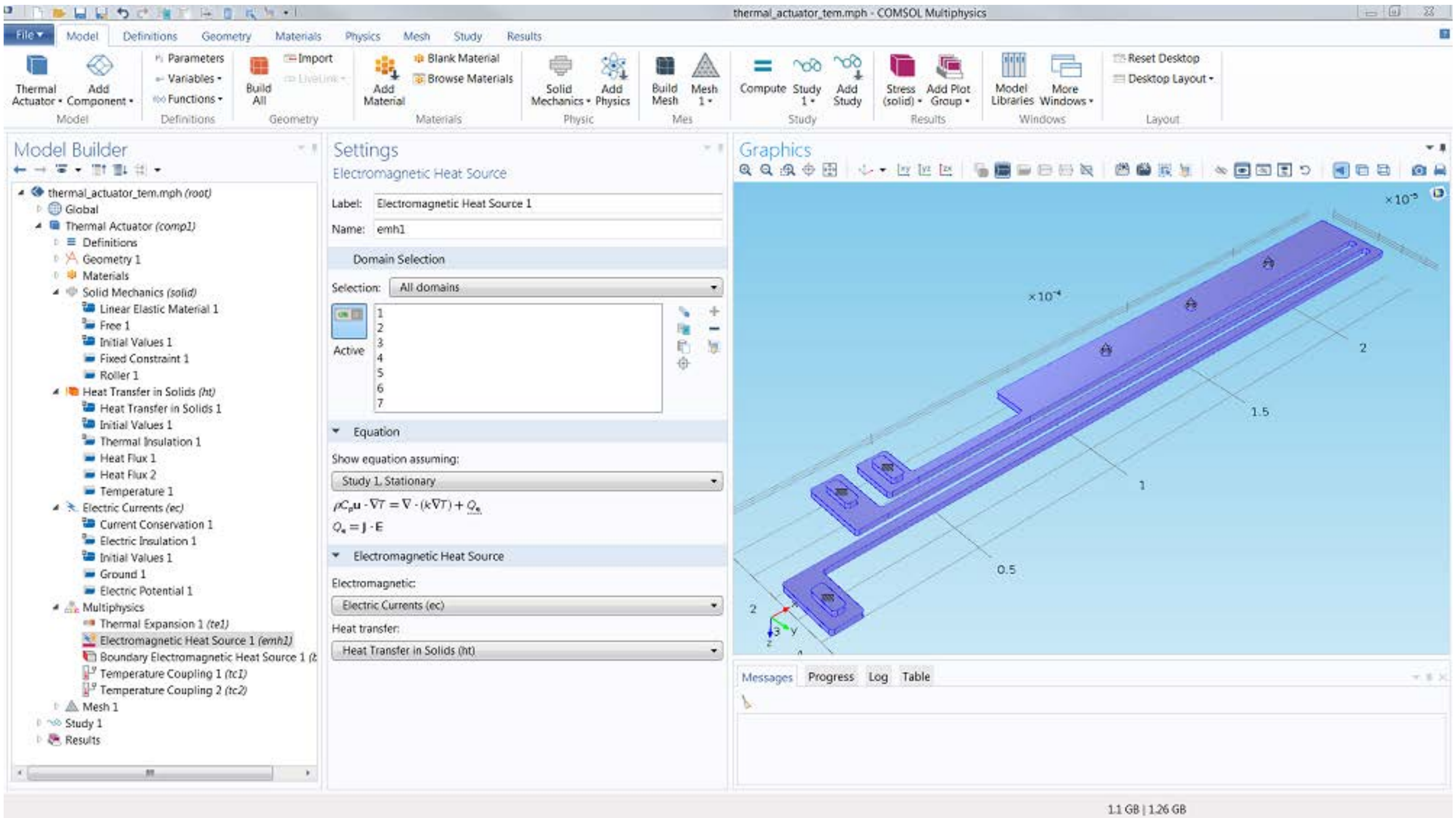




material taken from slides from Shulin Sun and Guang-yu Guo

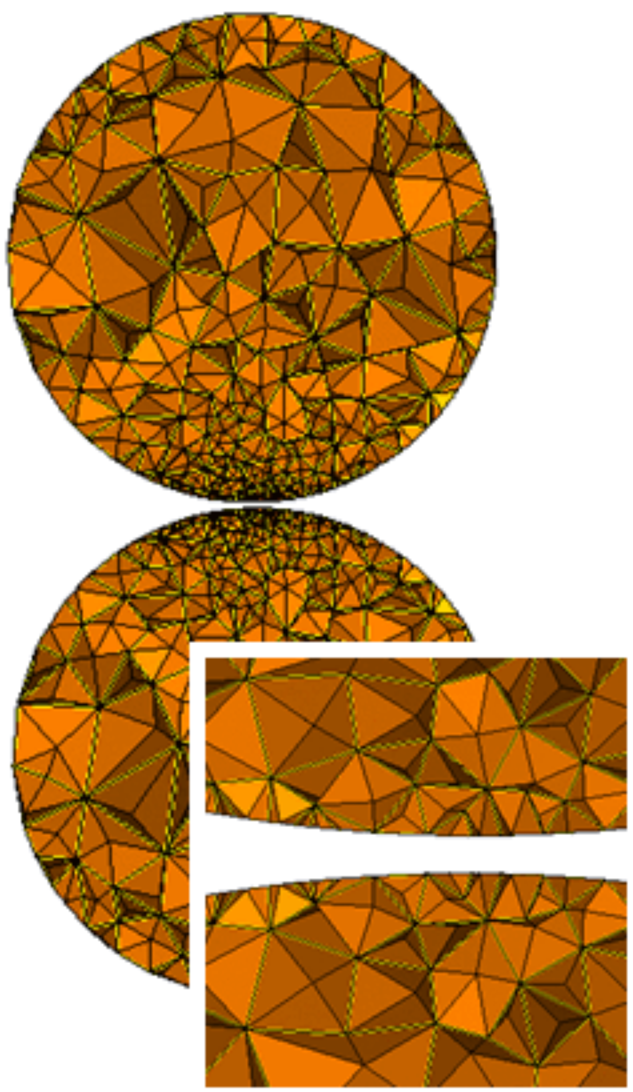
Finite Element Method

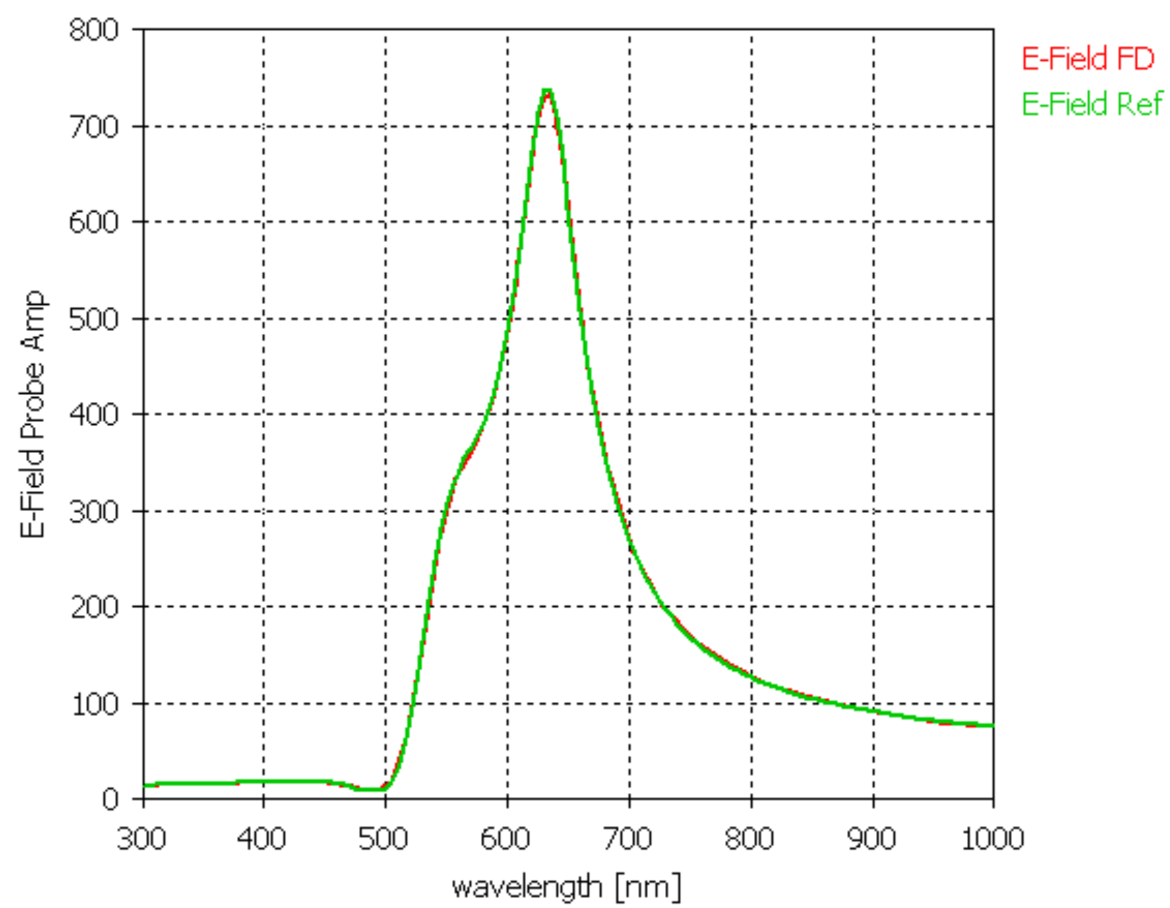
COMSOL



Finite Integration Method

Computer Simulation Technology

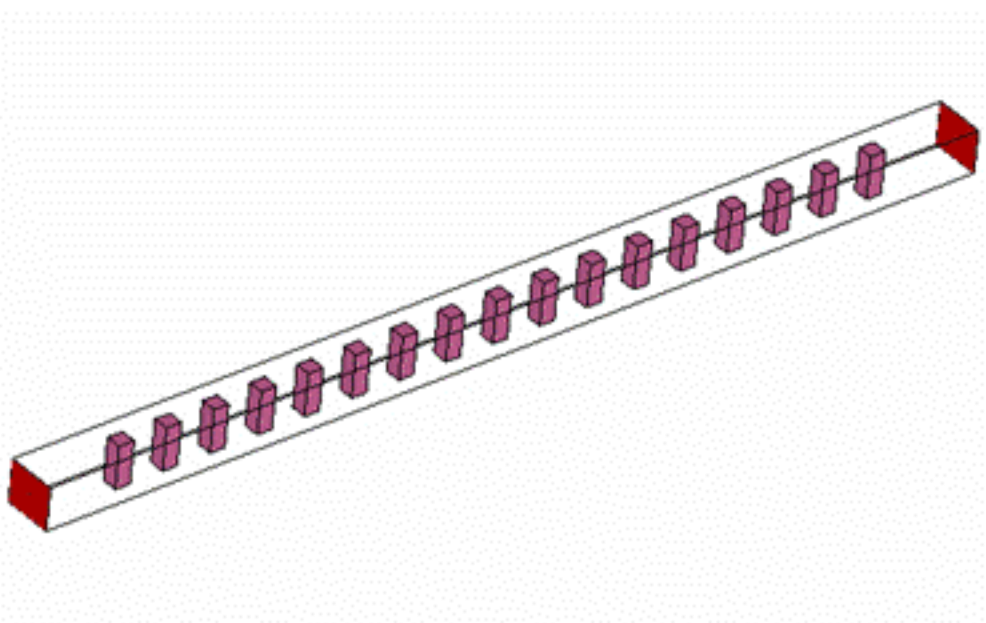


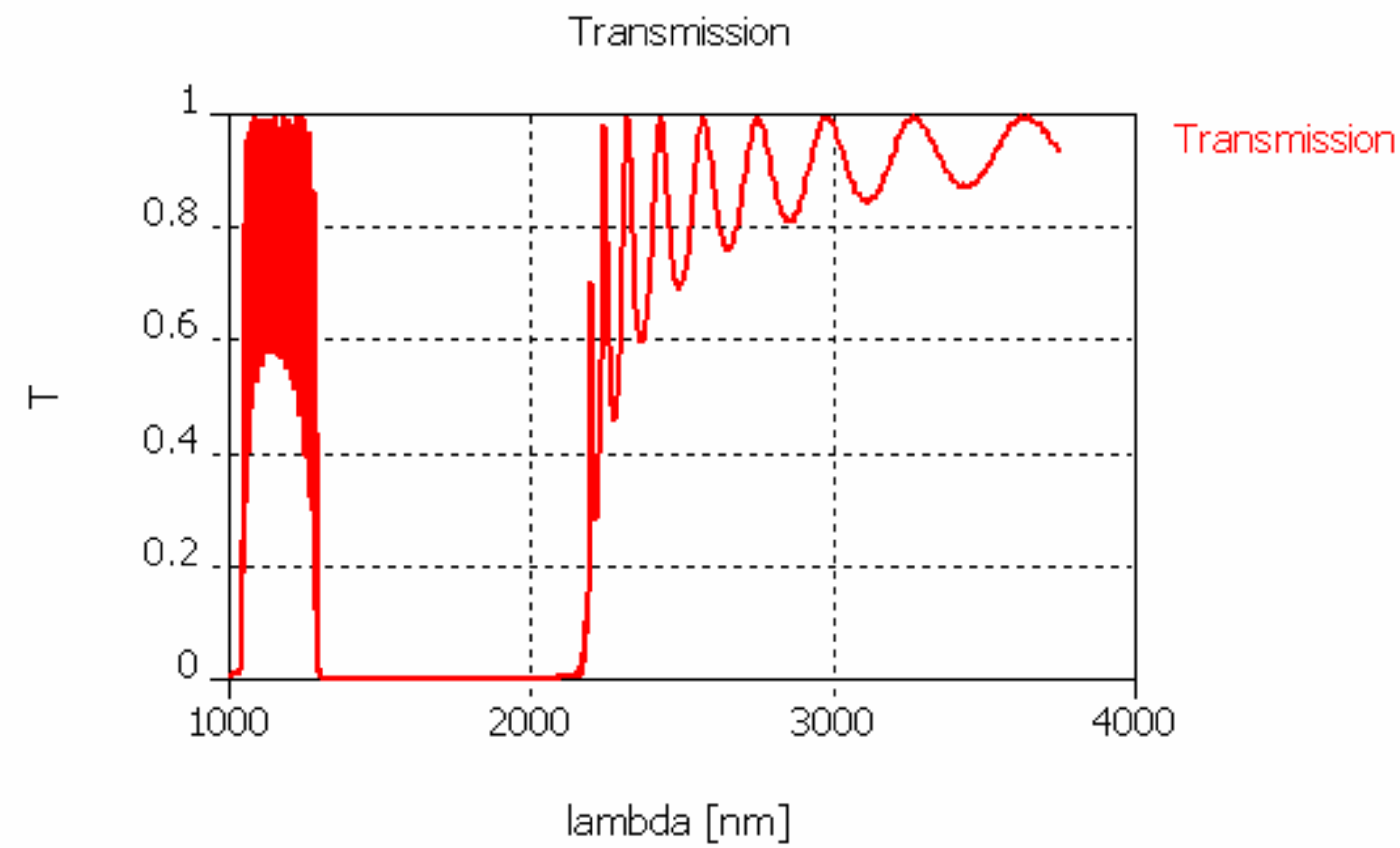


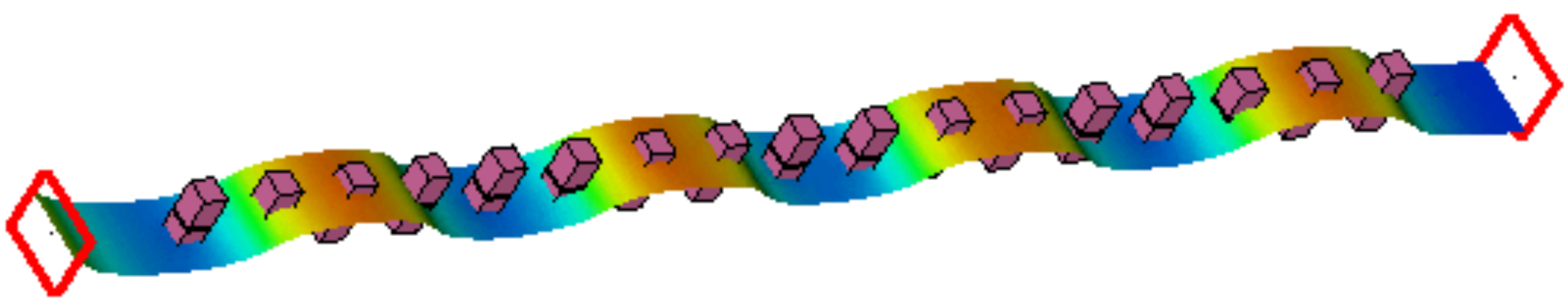
Finite Integration Method

CST

Computer Simulation Technology







Boundary Element Method

material taken from slides from U. Hohenester (Graz)

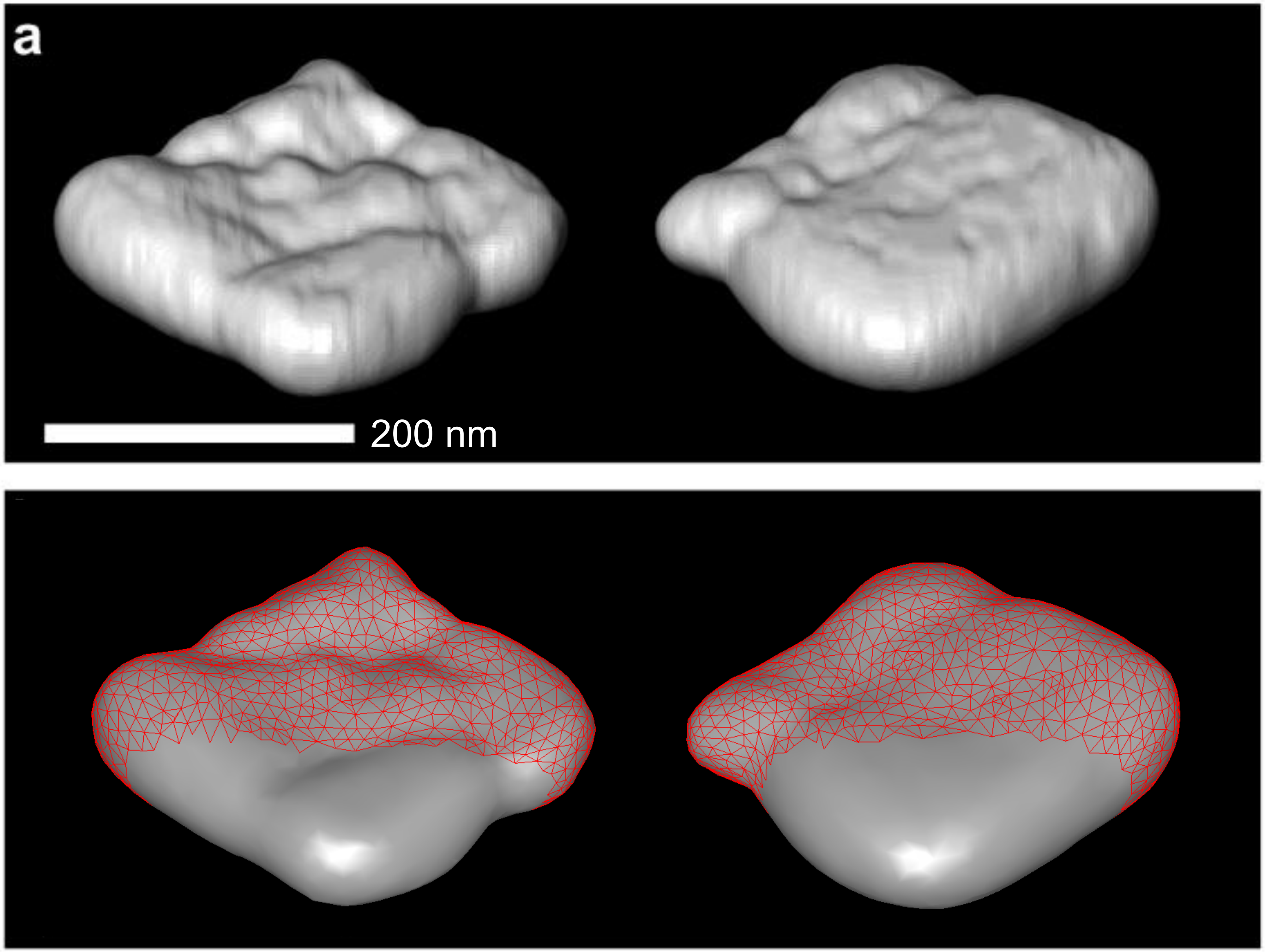
http://physik.uni-graz.at/mnpbem

U. Hohenester and J. Krenn, Phys. Rev. B 72, 195429 (2005).

U. Hohenester and A. Trügler, Comp. Phys. Comm. 183, 370 (2012); ibid 185, 1177 (2014).

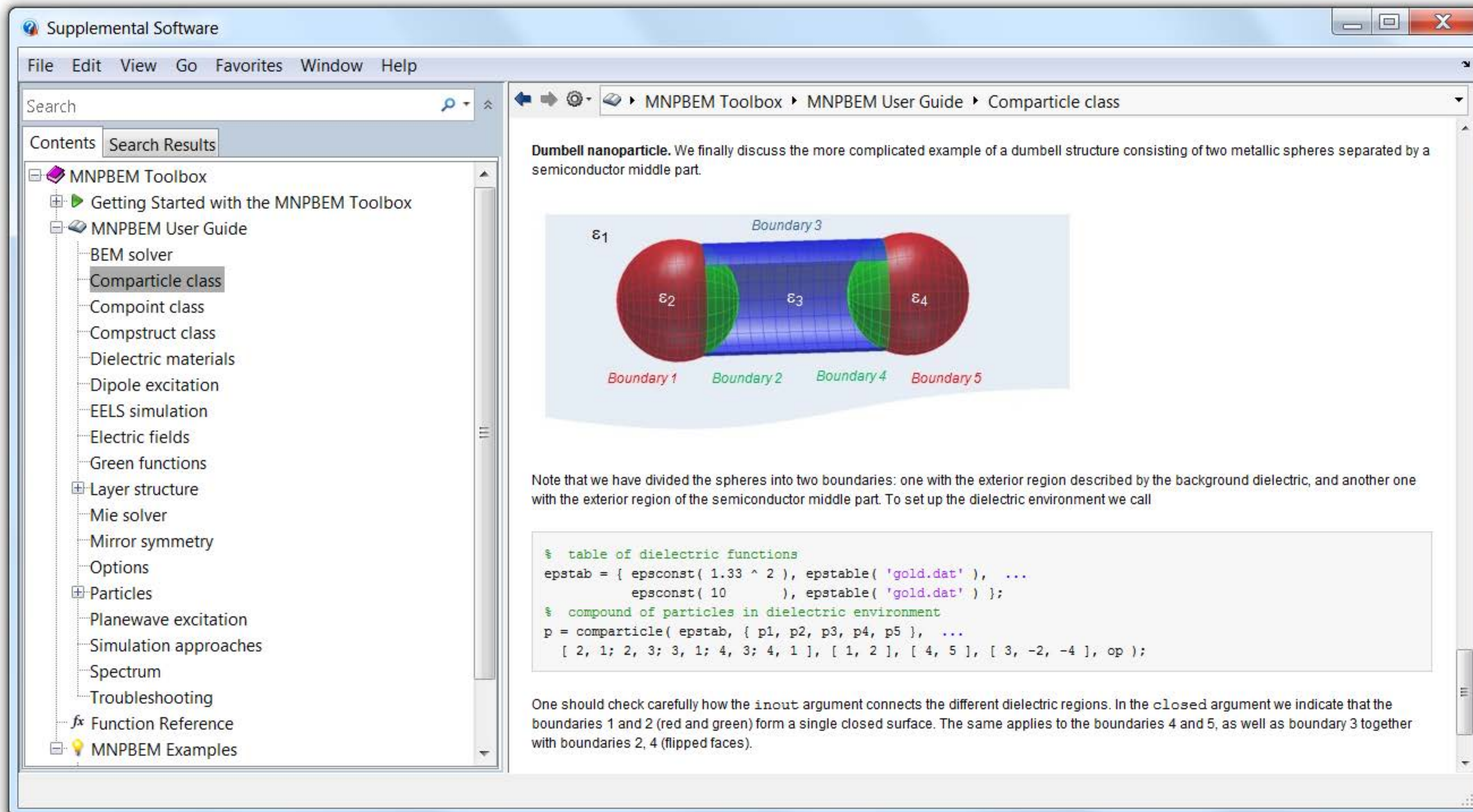
- perfect for scattering problems
- discretises the problem on a surface instead of a bulk medium
- assumes a homogenous scatterer
- frequency domain
- nonlocal and nonlinear extensions

Boundary Element Method



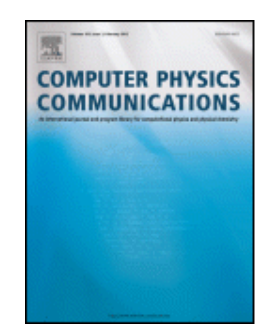
Haberfehlner et al, Nano Lett. **15**, 7726 (2015).

Boundary Element Method



U. Hohenester and J. Krenn, Phys. Rev. B 72, 195429 (2005).

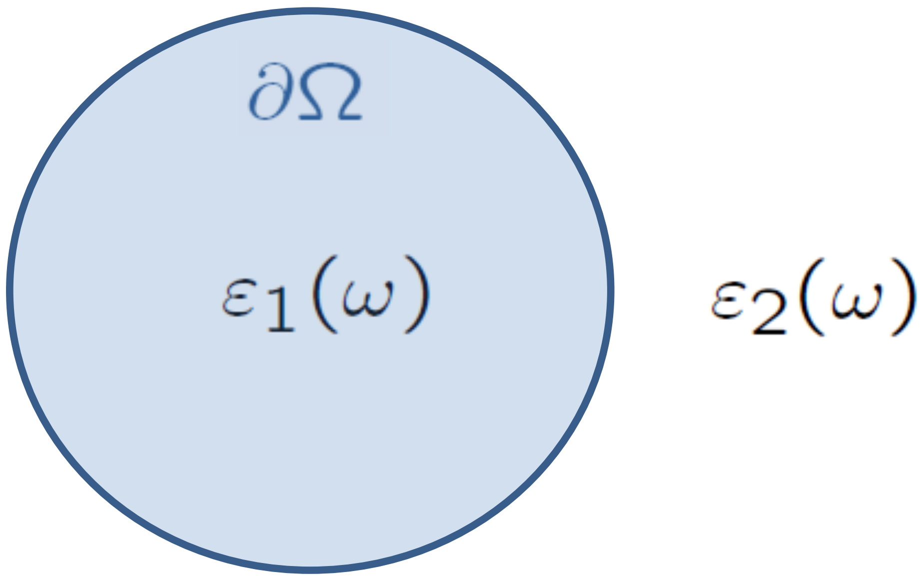
U. Hohenester and A. Trügler, Comp. Phys. Comm. 183, 370 (2012); ibid 185, 1177 (2014).

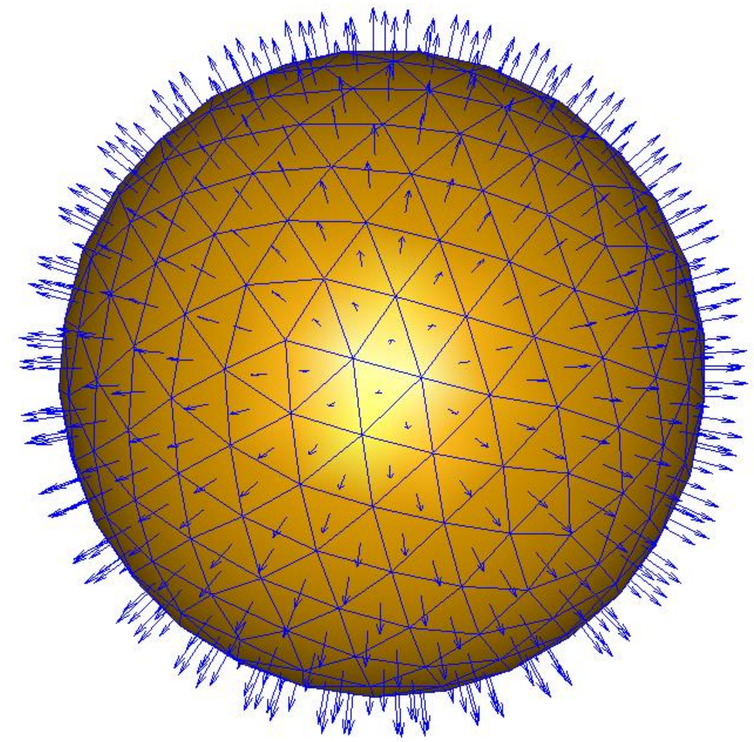


http://physik.uni-graz.at/mnpbem

Boundary Element Method

The ingredients of the BEM approach :





Implementation within the MNPBEM toolbox

```
%  table of dielectric functions
epstab = { epsconst( 1 ), epstable( 'gold.dat' ) };
%  nanosphere with a diameter of 50 nm
p = trisphere( 256, 50 );
%  initialize dielectic environment
p = comparticle( epstab, { p }, [ 2, 1 ], 1 );

%  plot particle boundary with outer surface normals
plot( p, 'EdgeColor', 'b', 'nvec', 1 );
```

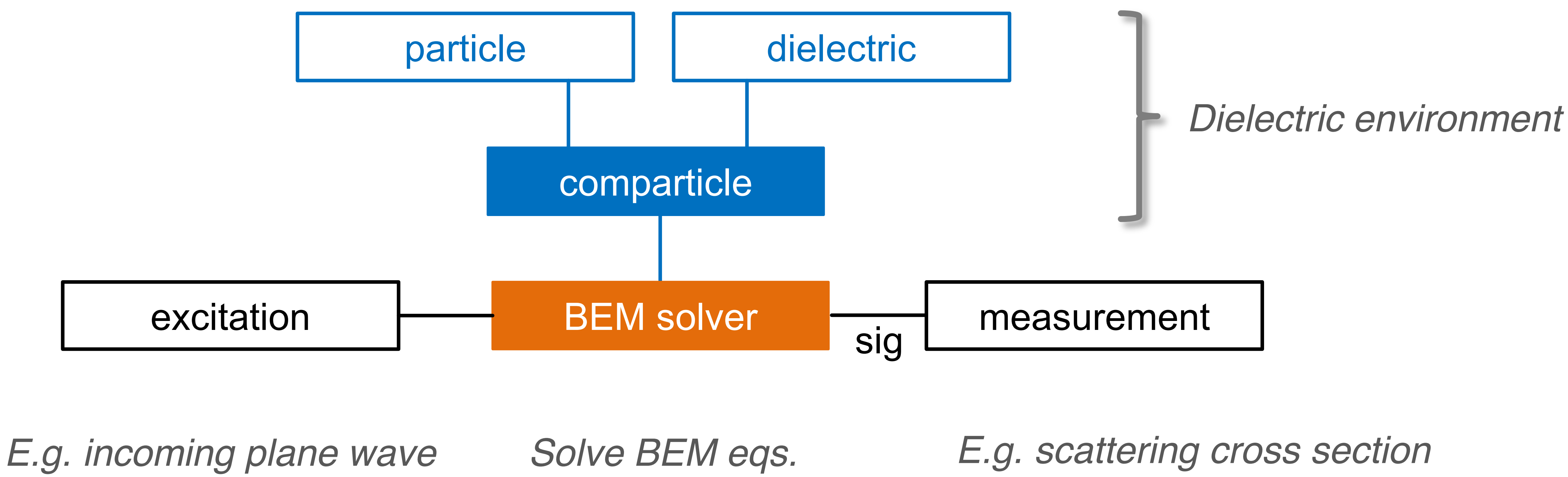
closed particle boundary

Pointer to dielectric function at in- and outside of boundary

Boundary Element Method

Flow chart for MNPBEM simulation

A typical MNPBEM simulation



Quasistatic BEM equations

$$\sum_j \left[\Lambda \, \delta_{ij} + \left(\frac{\partial G}{\partial n} \right)_{ij} \right] \sigma_j = -\varepsilon_0 \left(\frac{\partial V^{\rm ext}}{\partial n} \right)_i$$

The MNPBEM toolbox uses a collocation scheme for the quasistatic or full Maxwell's equations (see below)

Computational Photonics

Outro