Numerical Simulation Methods in Geophysics, Lecture 9: from 1D over 2D to 3D EM

1. MGPY+MGIN

thomas.guenther@geophysik.tu-freiberg.de

TUBAF
Die Ressourcenuniversität. Seit 1765.

Recap

- Maxwells equations in time- and frequency domain
- single frequency approach (decomposition)
- Helmholtz term ($\frac{\partial^2 u}{\partial x^2}$ and u term)
- u term needs mass matrix (like in time-stepping)
- complex-valued matrix problem

Inductive EM

Assume: $\omega^2\mu\epsilon < \omega\mu\sigma$, no sources ($\boldsymbol{\nabla}\cdot\mathbf{j}_s = 0$), + vector identity

$$\boldsymbol{\nabla}\times\boldsymbol{\nabla}\times\mathbf{F} = \boldsymbol{\nabla}\boldsymbol{\nabla}\cdot\mathbf{F} - \boldsymbol{\nabla}^2\mathbf{F}$$

leads with $\boldsymbol{\nabla}\cdot\mathbf{E} = 0 = \boldsymbol{\nabla}\cdot\mathbf{B}$ to the vector Helmholtz PDE

$$-\nabla^2\mathbf{E} + \imath\omega\mu\sigma\mathbf{E} = 0$$

$$-\boldsymbol{\nabla}\cdot\sigma^{-1}\nabla\mathbf{H} + \imath\omega\mu\mathbf{H} = 0$$

Helmholtz equation $-\nabla^2\mathbf{H} - k^2\mathbf{H} = 0$ with $k^2 = -\imath\omega\mu\sigma$

TM polarization (scalar problem)

Transverse magnetic (TM) mode

Assume the source field is oscillating perpendicular to the modelling plane, i.e.

$\mathbf{H} = [H_x, 0, 0]^T e^{\imath\omega t}$.

Then the PDE holds for the scalar H_x (now only H)

$$-\boldsymbol{\nabla} \cdot \sigma^{-1} \boldsymbol{\nabla} H_x(y, z) + \imath\omega\mu H_x(y, z) = 0$$

Halfspace solution

$$\frac{\partial^2 H}{\partial z^2} + k^2 H = 0 \quad \text{with} \quad k^2 = -\imath\omega\mu\sigma$$

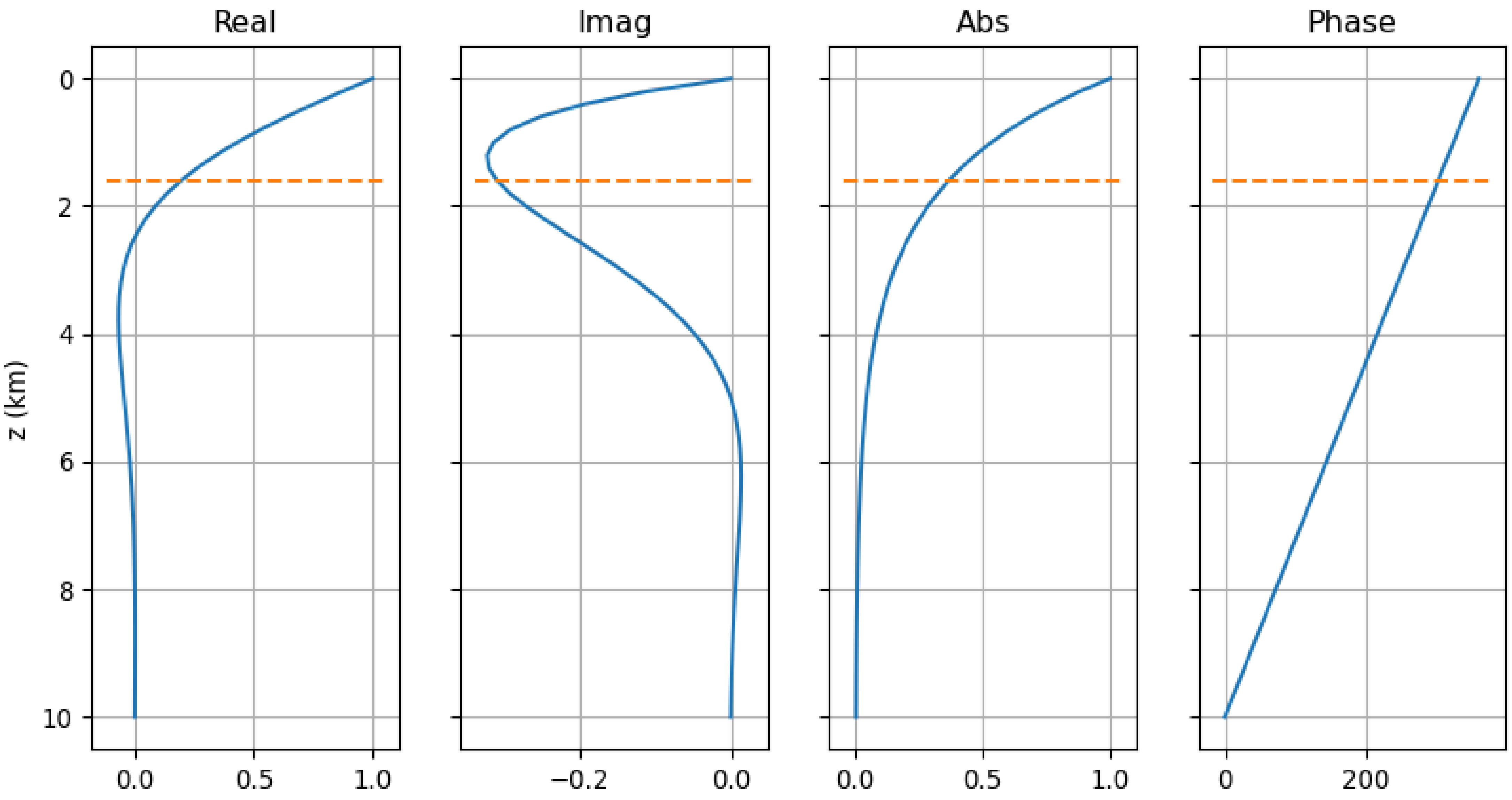
In the Earth, the solution is $H(z) = H_0 e^{-\imath k z} \quad \partial_{zz} H = -k^2 H$

$k = \sqrt{-\imath\omega\mu\sigma} = \sqrt{\omega\mu\sigma/2}(1-\imath) = 1/d(1-\imath)$

with skin depth (1/e decay) $d = \sqrt{2/\omega\mu\sigma} = \sqrt{T/\pi\mu\sigma}$ (~500$\sqrt{\rho/f}$)

$$-\imath k = -(1+i)/d \Rightarrow H = H_0 e^{-z/d} e^{-\imath z/d} \quad \equiv \cos(\omega t - z/d)$$

Analytical solution



EM with finite elements

$$-\boldsymbol{\nabla}^2 u + \imath\omega\mu\sigma u = f$$

$$-\int_\Omega w\boldsymbol{\nabla}^2 u \mathrm{d}\Omega + \int_\Omega w\imath\omega\mu\sigma u \mathrm{d}\Omega = \int_\Omega w f \mathrm{d}\Omega$$

Gauss's integral law

$$\int_\Omega \boldsymbol{\nabla} w \cdot \boldsymbol{\nabla} u \mathrm{d}\Omega + \imath\omega \int_\Omega \mu\sigma w u \mathrm{d}\Omega = \int_\Omega w f \mathrm{d}\Omega$$

Weak formulation

$u = \sum_i u_i \mathbf{v}_i$ and $w_i \in \{v_i\}$ leads to

$$\int_\Omega \boldsymbol{\nabla} v_i \cdot \boldsymbol{\nabla} v_j \mathrm{d}\Omega + \imath\omega\mu \int_\Omega \sigma v_i v_j \mathrm{d}\Omega = \int_\Omega v_i f \mathrm{d}\Omega$$

$$\langle \boldsymbol{\nabla} v_i | \boldsymbol{\nabla} v_j \rangle + \imath\omega\mu \langle v_i | \sigma v_j \rangle = \langle v_i | f \rangle \quad \text{inner products}$$

representation by matrix-vector product $(\mathbf{A} + \imath\omega\mathbf{M}^\sigma)\mathbf{u} = \mathbf{b}$

with $A_{ij} = \langle \boldsymbol{\nabla} v_i | \boldsymbol{\nabla} v_j \rangle$, $M^\sigma_{ij} = \langle v_i | v_j \rangle$ and $b_i = \langle v_i | f \rangle$

The problem in 1D

use self-implemented matrices from `stiffnessMatrix1DFE` & `massMatrix1DFE`

```
from poisson1d import stiffnessMatrix1DFE, massMatrix1DFE
T = 0.1  # 0.1
w = 2 * np.pi / T
sigma0 = 1/100  # 1/100
mu = np.pi * 4e-7
z = np.arange(-10000, 0.1, 200)
A, b = stiffnessMatrix1DFE(x=z, uR=1)
M = massMatrix1DFE(x=z, a=mu*sigma0)
AM = A + M * 1j * w
u = np.linalg.solve(AM, b)
```

Complex or real-valued?

The complex-valued system

$$(\mathbf{A} + \imath\omega\mathbf{M})\mathbf{u} = (\mathbf{A} + \imath\omega\mathbf{M})(\mathbf{u}_r + \imath\mathbf{u}_i) = \mathbf{b}_r + \imath\mathbf{b}_i$$

can be transferred into a doubled real-valued system

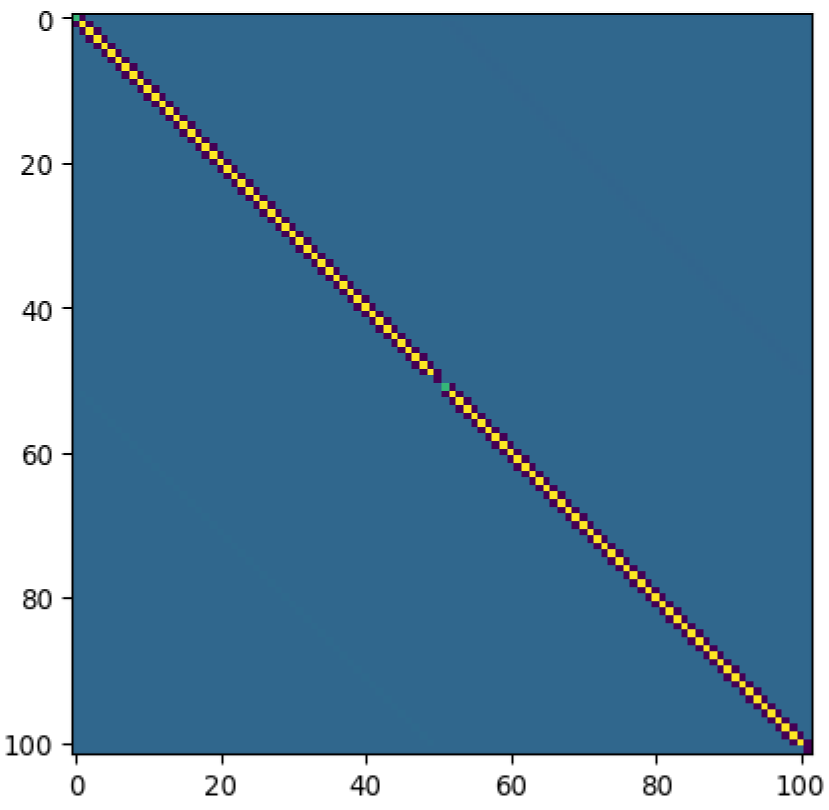
$$\mathbf{A}\mathbf{u}_r + \imath\mathbf{A}u_i + \imath\omega\mathbf{M}\mathbf{u}_r - \omega\mathbf{M}\mathbf{u}_i = \mathbf{b}_r + \imath\mathbf{b}_i$$

$$\begin{pmatrix} A & -\omega M \\ \omega M & A \end{pmatrix} \begin{pmatrix} u_r \\ u_i \end{pmatrix} = \begin{pmatrix} b_r \\ b_i \end{pmatrix}$$

Complex-to-real

$$\mathbf{B} = \begin{pmatrix} \mathbf{A} & -\omega\mathbf{M} \\ \omega\mathbf{M} & \mathbf{A} \end{pmatrix}$$

```
D = np.vstack([np.hstack([A, -M*w]),
               np.hstack([M*w, A])])
plt.imshow(D)
d = np.hstack([b, b*0])
uri = np.linalg.solve(D, d)
u = uri[:len(z)] + uri[len(z):] * 1j
```

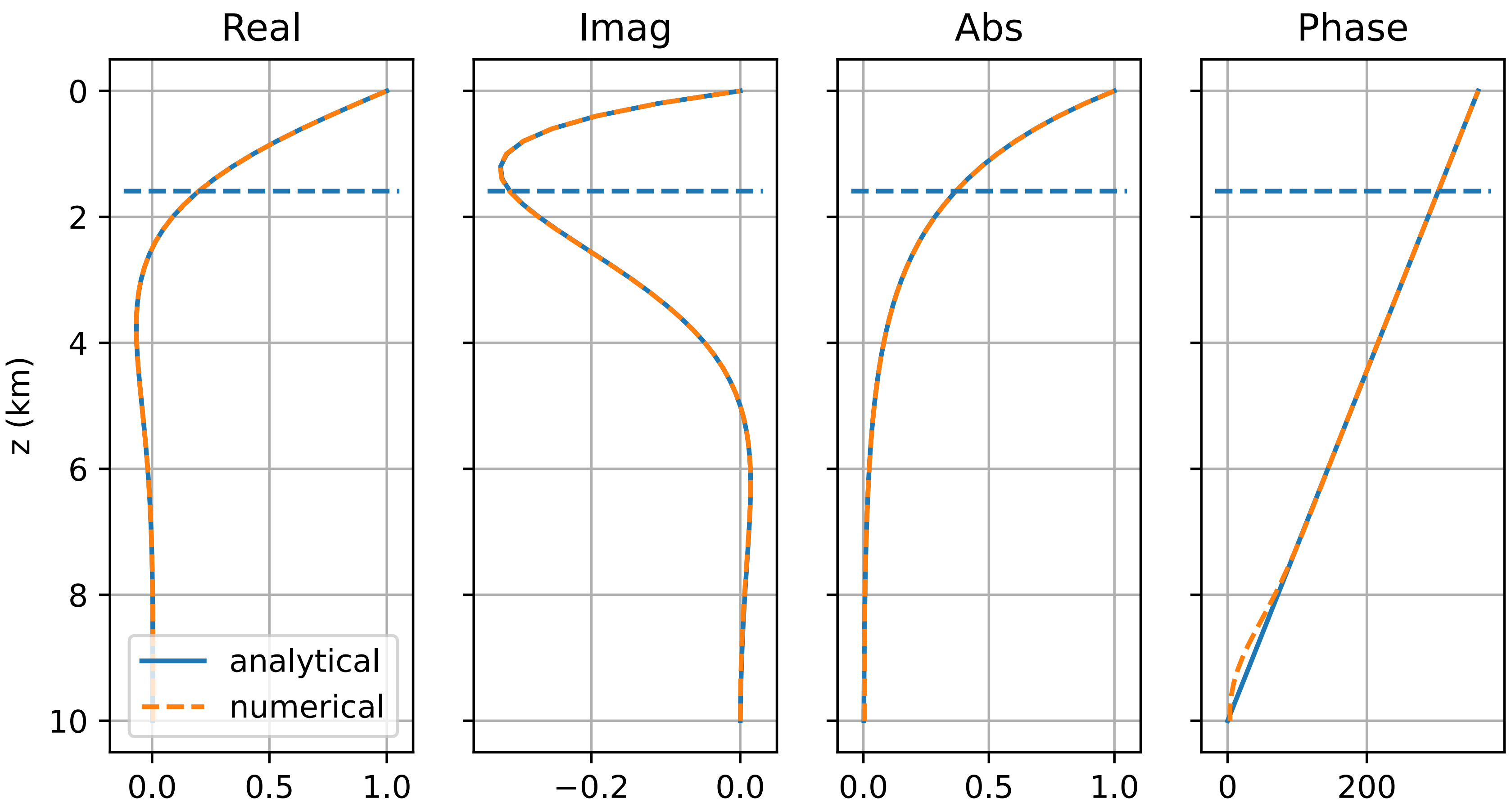


Complex-to-real in pyGIMLi

$$\mathbf{B} = \begin{pmatrix} \mathbf{A} & -\omega \mathbf{M} \\ \omega \mathbf{M} & \mathbf{A} \end{pmatrix}$$

Complex in pyGIMLi

```
C = pg.matrix.CSparseMatrix(A.vecColPtr(), A.vecRowIdx(),
     pg.core.toComplex(A.vecVals(), M.vecVals()*w))
c = pg.core.toComplex(b, b*0)
u = ps.linSolve(C, c).array()  # important (bug in pyGIMLi!)
```



Secondary field for EM

Let E_0 be the solution to the equation for $\sigma = \sigma_0$

$$-\boldsymbol{\nabla}^2 E_0 + \imath\omega\mu\sigma_0 E_0 = 0$$

and $E = E_0 + E_a$ for $\sigma = \sigma_0 + \delta\sigma$

$$-\boldsymbol{\nabla}^2 (E_0 + E_a) + \imath\omega\mu(\sigma_0 + \delta\sigma)(E_0 + E_a) = 0$$

subtracting leads to

$$-\boldsymbol{\nabla}^2 E_a + \imath\omega\mu(\sigma_0 + \delta\sigma) E_a = -\imath\omega\mu\delta\sigma E_0$$

Secondary field for EM

$$-\boldsymbol{\nabla}^2 E_a + \imath\omega\mu\sigma E_a = -\imath\omega\mu\delta\sigma E_0$$

Note

Source terms only arise at anomalous conductivities and increase with primary field

is solved for $E_0(\mathbf{u})$ using $\mathbf{A}$-stiffness, $\mathbf{M}$-mass and $E_0(\mathbf{u}_0)$

$$(\mathbf{A} + \imath\omega\mathbf{M}_\sigma)\mathbf{u} = -\imath\omega\mathbf{M}_{\delta\sigma}\mathbf{u}_0$$

```
A = stiffnessMatrix1DFE(x=z)
M = massMatrix1DFE(x=z, a=w*mu*sigma)
dM = massMatrix1DFE(x=z, a=w*mu*(sigma-sigma0))
u = uAna + solve(A+M*w*1j, dM@uAna * w*1j)
```

EM vector modelling

$$\boldsymbol{\nabla} \times \mu^{-1} \boldsymbol{\nabla} \times \mathbf{E} + \imath\omega\sigma\mathbf{E} - \omega^2 \epsilon \mathbf{E} = -\imath\omega \mathbf{j}_s$$

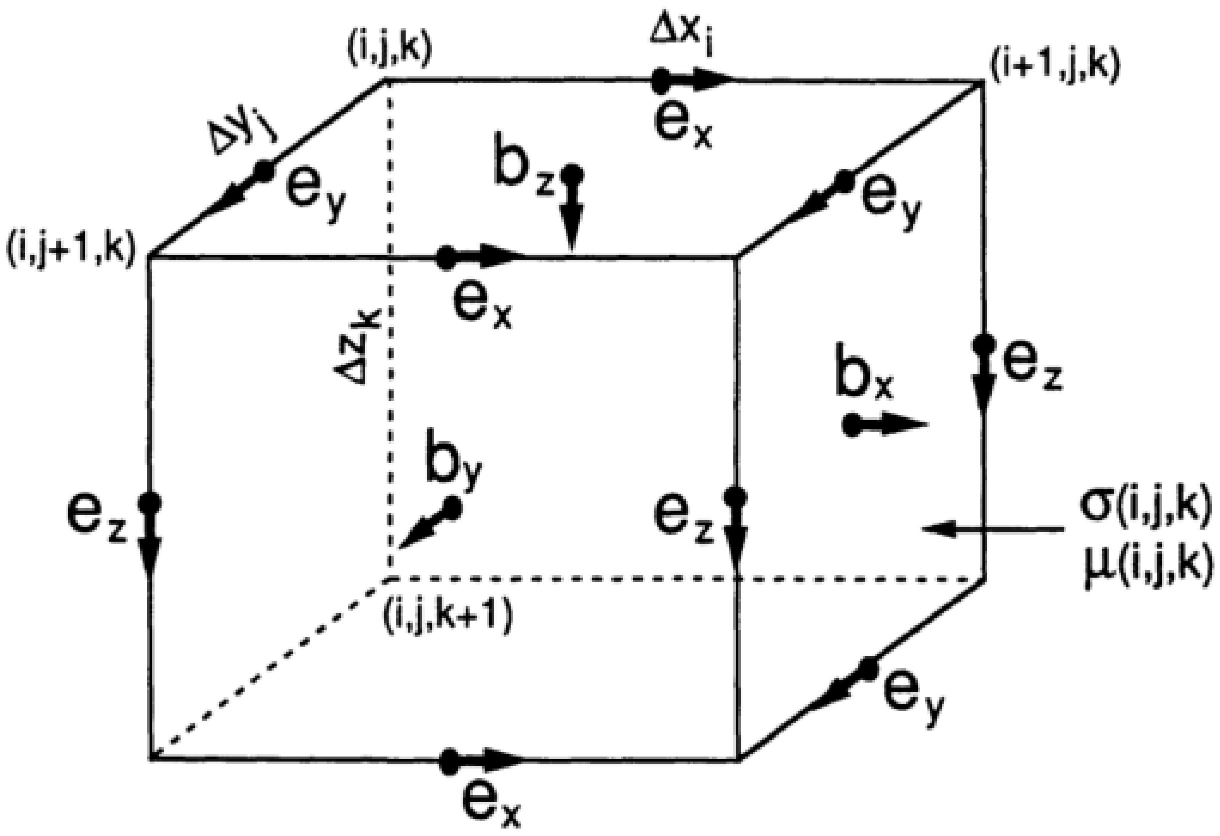
$$\boldsymbol{\nabla} \times \sigma^{-1} \boldsymbol{\nabla} \times \mathbf{H} + \imath\omega\mu\mathbf{H} - \omega^2 \mu\epsilon \mathbf{H} = \boldsymbol{\nabla} \times \sigma^{-1} \mathbf{j}_s$$

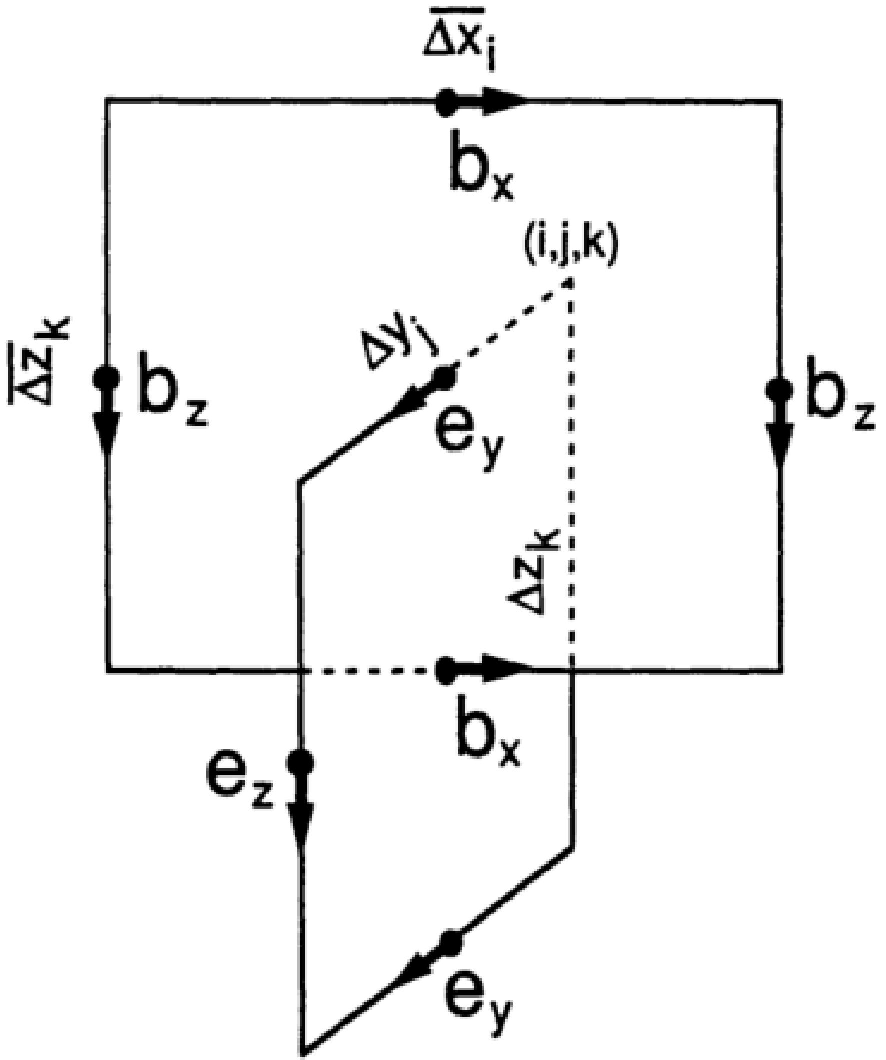
History

> **James Clerk Maxwell, Treatise on Electricity & Magnetism, 1891**
>
> “Physical vector quantities may be divided into two classes, in one of which the quantity is defined with reference to a line, while in the other the quantity is defined with reference to an area.”

- integral equations (IE), finite differences (FD) and elements (FE)
- nodal and vector basis functions, secondary field
- decompose 3D source (2D inverse) in wavenumber domain
- improvement of equation solvers and preconditioners
- parallel solvers, high-performance computing

Solving EM problems with staggered grid





Staggered grid cell after Yee (1966)

The way from FD to FE

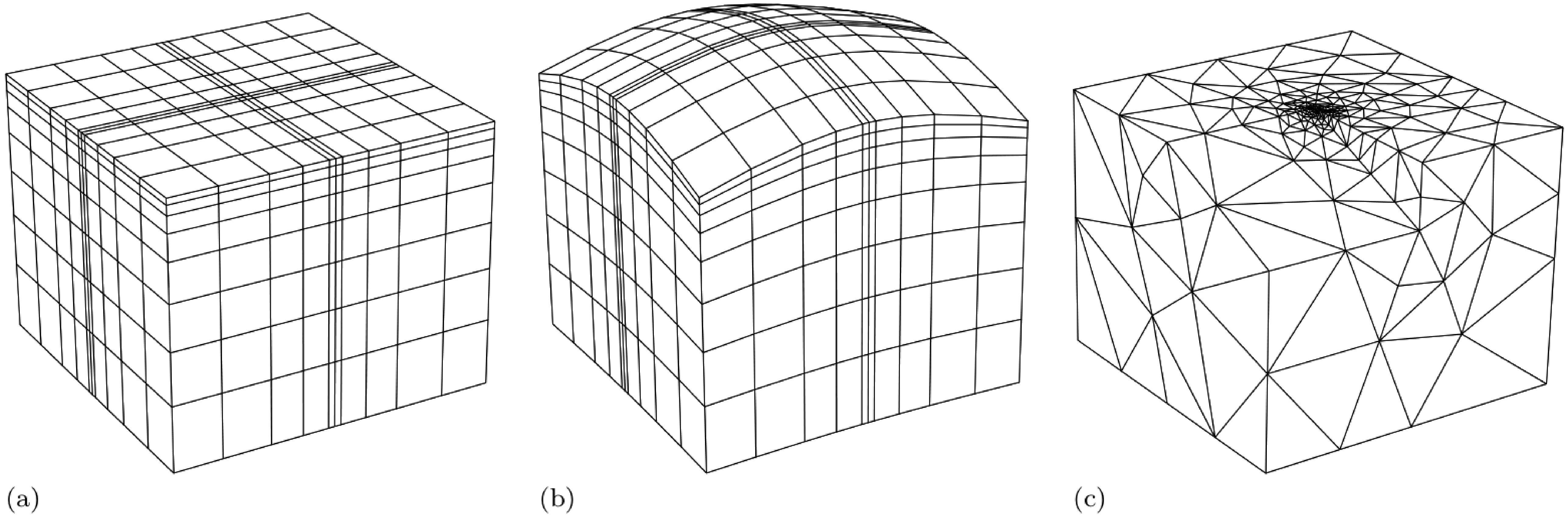
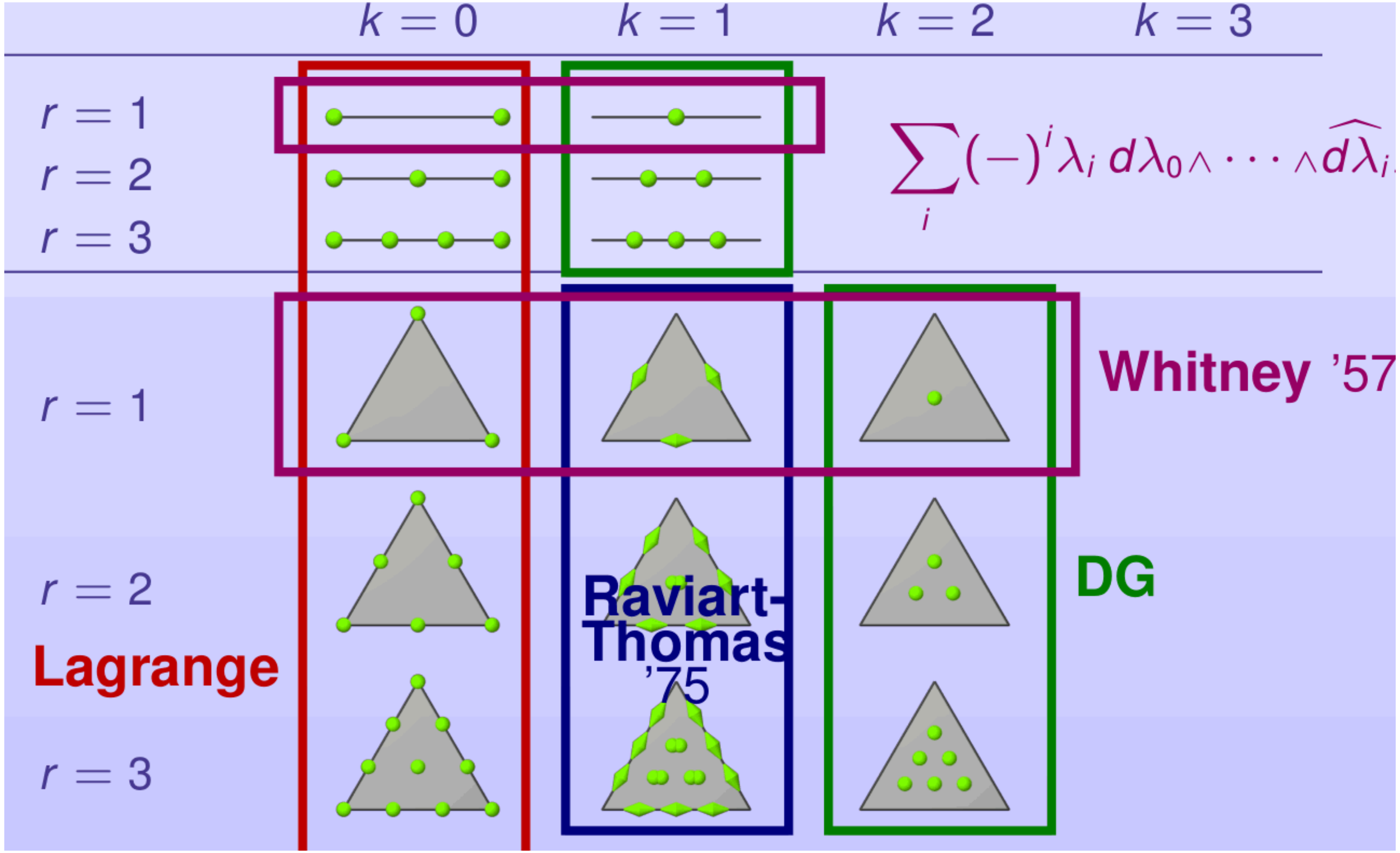


Figure 1. Different grid types: Orthogonal cuboid (a), non-orthogonal hexahedral (b) and unstructured tetrahedral (c) grid.

Orthogonal, non-orthogonal and irregular grids (Rücker et al., 2006)

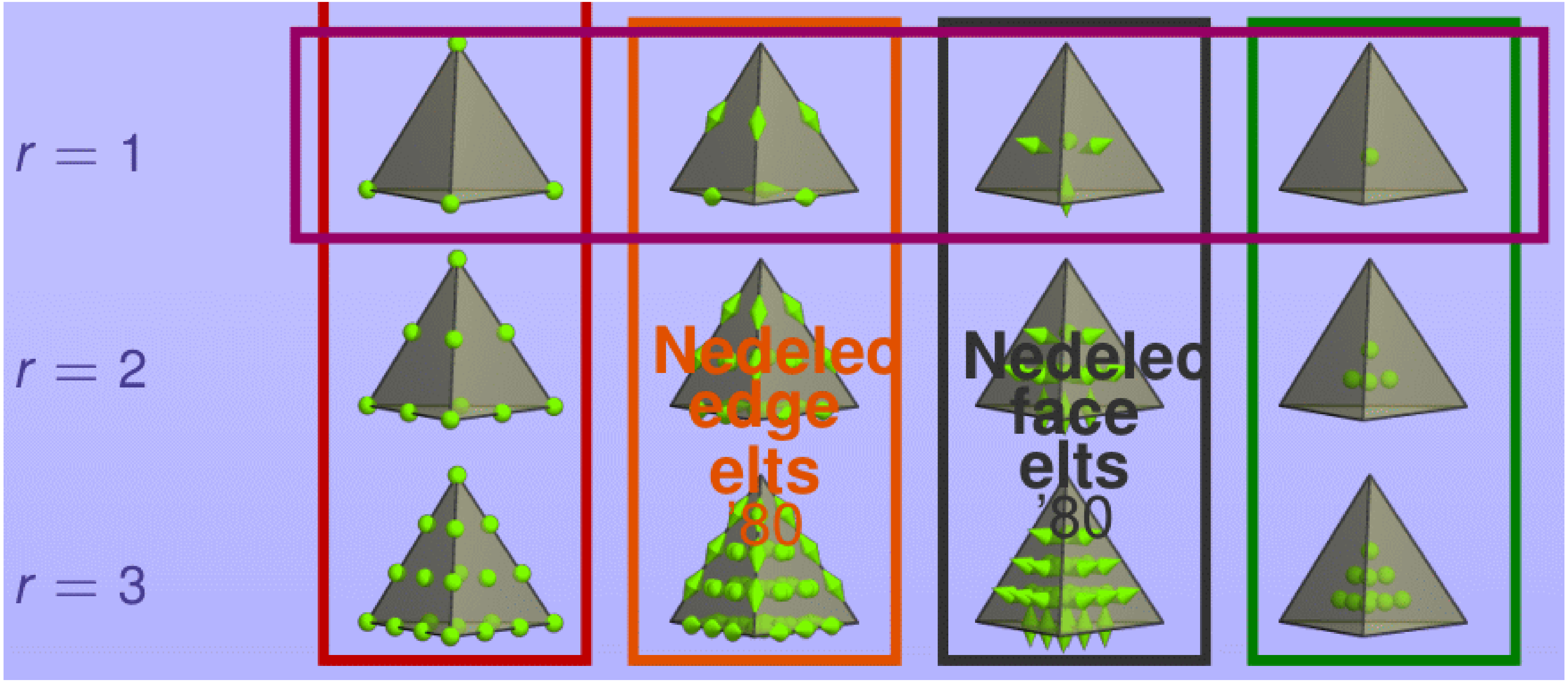
The Finite Element zoo (1D & 2D)



Arnold (2013): Periodic table of elements

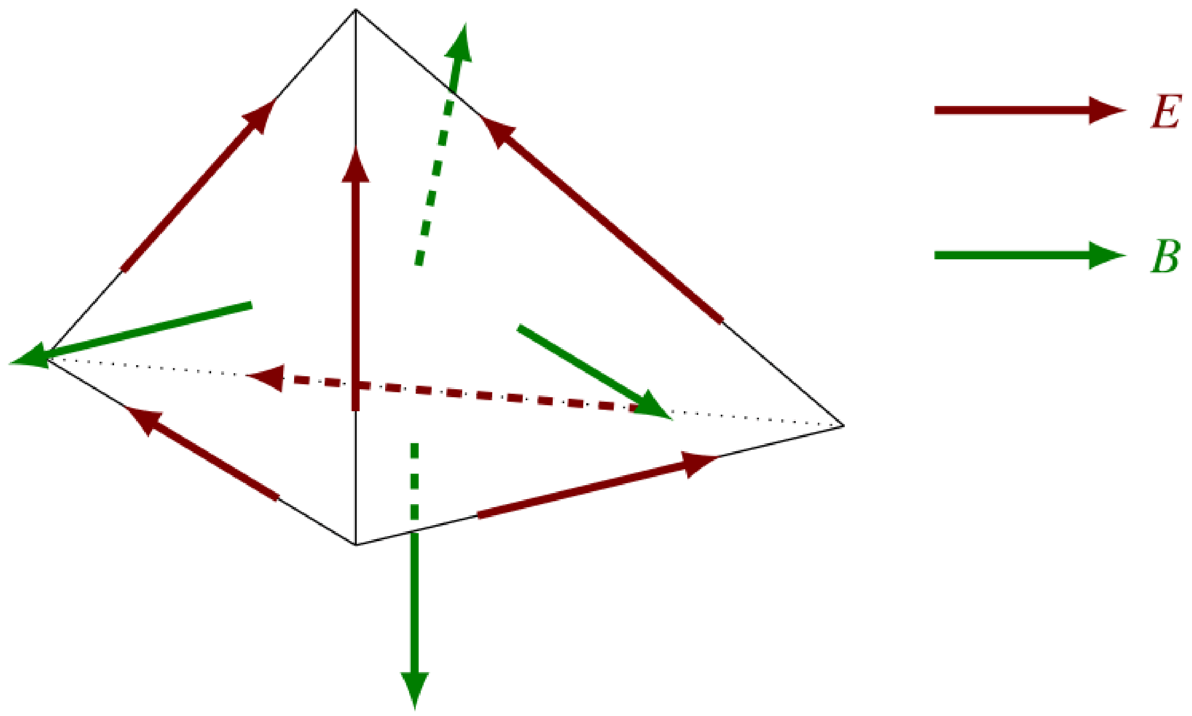
- k=0 Nodal elements
- k=1 edge elements
- k=2 face elements (Discontinuous Galerkin)
- r/p higher order

The Finite Element zoo (3D)



Arnold (2013): Periodic table of elements

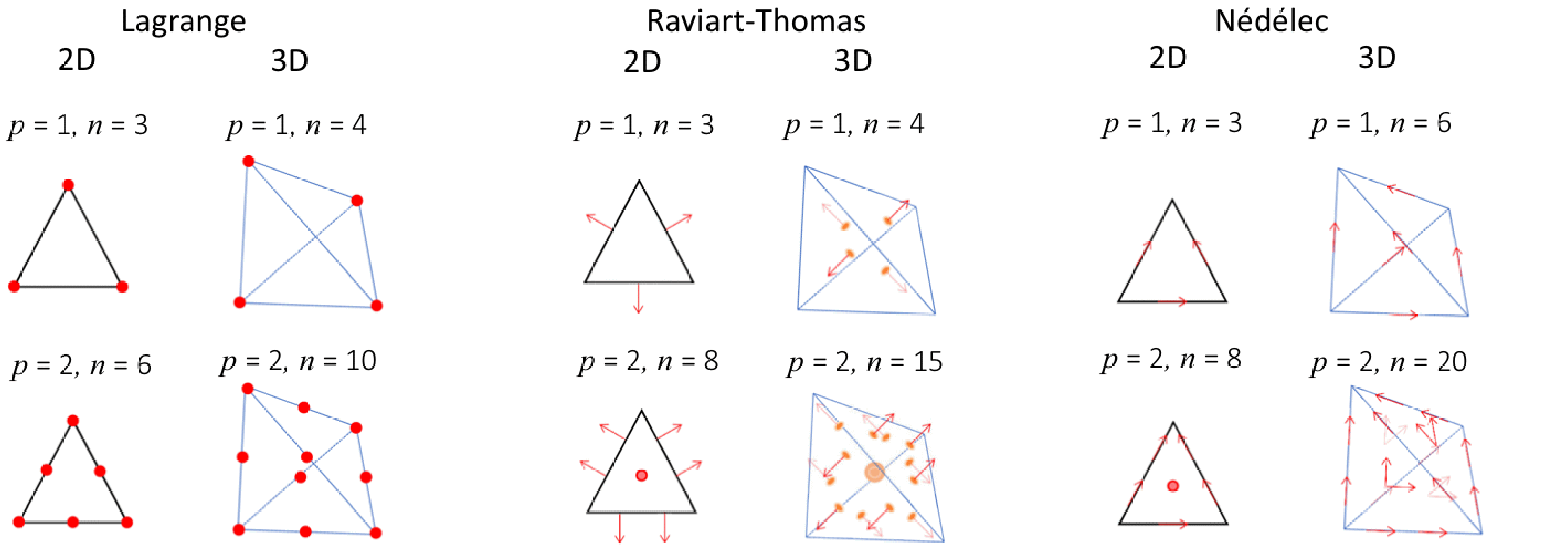
Vectorial solution for EM fields



Schwarzbach & Haber (2013)

- $\mathbf{E} \in H^{\text{curl}}(\Omega)$
- $\mathbf{B} \in H^{\text{div}}(\Omega)$
- natural BC: Dirichlet $\mathbf{n} \times \mathbf{E} = 0$
- Neumann BC:
  $\mathbf{n} \times \mu^{-1} \boldsymbol{\nabla} \times \mathbf{E} = 0$

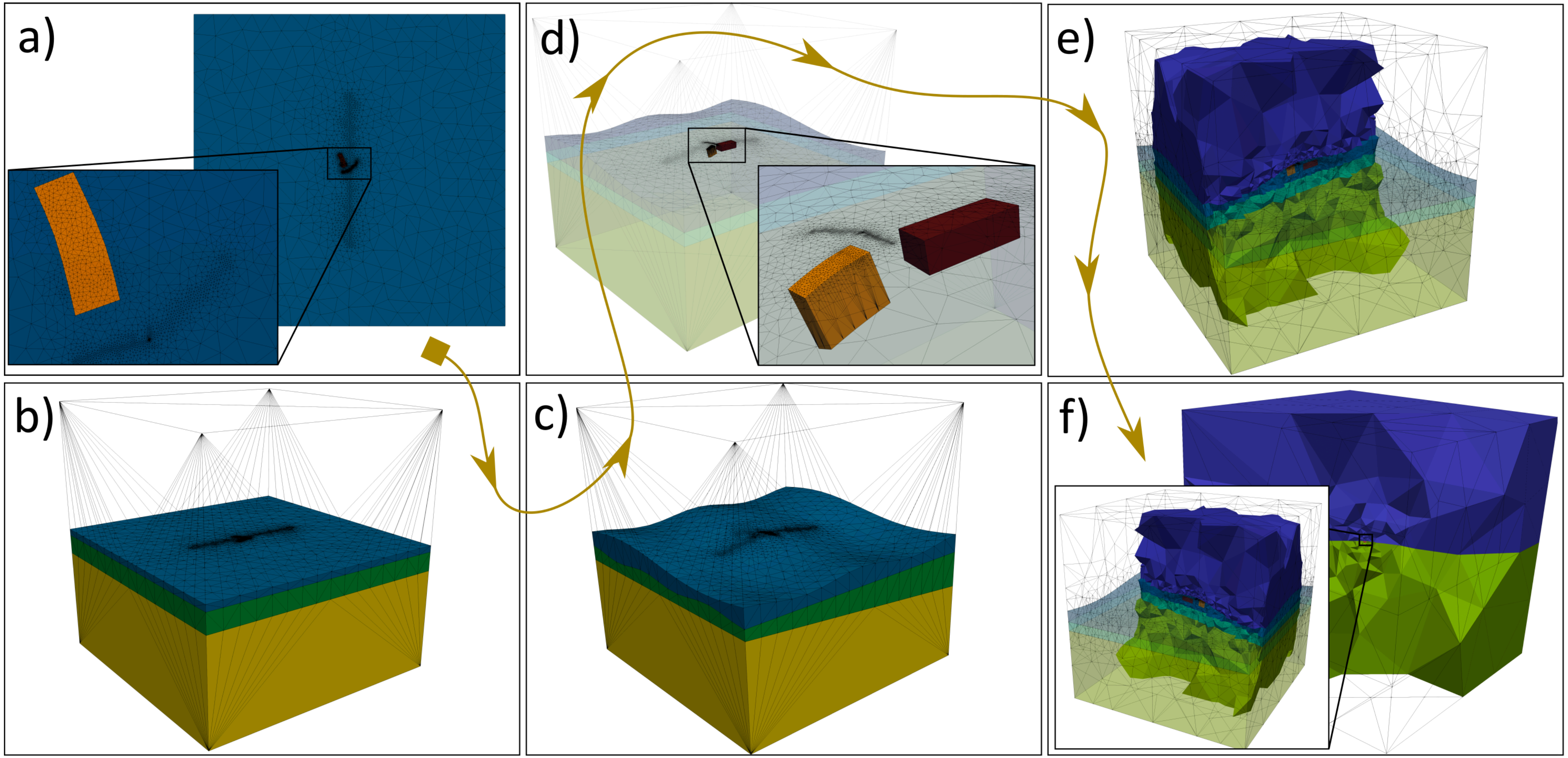
Elements used in EM modelling



$\mathcal{H}^1(\Omega) := \{v \in L^2(\Omega) : \nabla v \in [L^2(\Omega)]^3\}$ $\mathcal{H}(\mathrm{div};\Omega) := \{v \in [L^2(\Omega)]^3 : \nabla \cdot v \in L^2(\Omega)\}$ $\mathcal{H}(\mathrm{curl};\Omega) := \{v \in [L^2(\Omega)]^3 : \nabla \times v \in [L^2(\Omega)]^3\}$

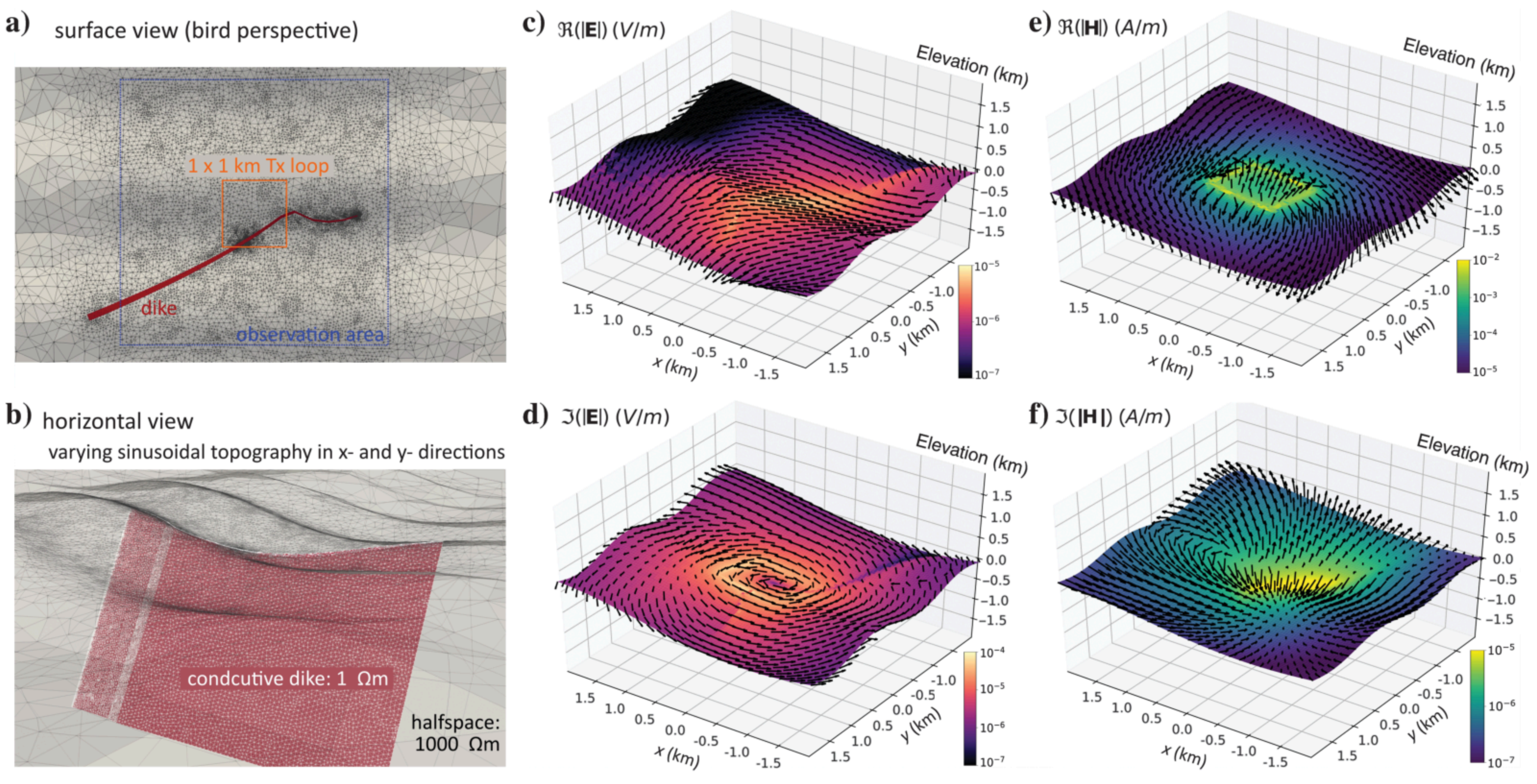
Types of elements relevant for EM problems (Spitzer, 2024)

Meshing complicated geometries



Meshing workflow in custEM (Rochlitz et al., 2019)

Modelling example



Modelling example of a conductive dike (Rochlitz et al., 2019)

Packages

Mesh generation: TetGen (3D), GMsh (2D/3D)

FE packages: FEniCS, NETGEN/NGsolve

Equation solvers: Suitesparse, MUMPS, SciPy

Computational frameworks: PetSc, MPI

EM modelling (and inversion) packages: Mare2DEM, emg3d, GoFEM, PETGEM, custEM, SimPEG, ModEM, FEMTIC

Boundary conditions

Mixed boundary conditions

So far…

- Dirichlet Boundary conditions $u = u_0$
- Neumann Boundary conditions $\frac{\partial u}{\partial n} = g_B$
- for vectorial problems $\mathbf{n} \cdot \mathbf{E} = 0$ or $\boldsymbol{\nabla} \times \mathbf{E} = 0$

In general mixed, also called Robin (or impedance convective) BC

$$au + b\frac{\partial u}{\partial n} = c$$

Example DC resistivity with point source

$$\boldsymbol{\nabla}\cdot\sigma\boldsymbol{\nabla}u=\boldsymbol{\nabla}\cdot\mathbf{j}=I\delta(\mathbf{r}-\mathbf{r}_s)$$

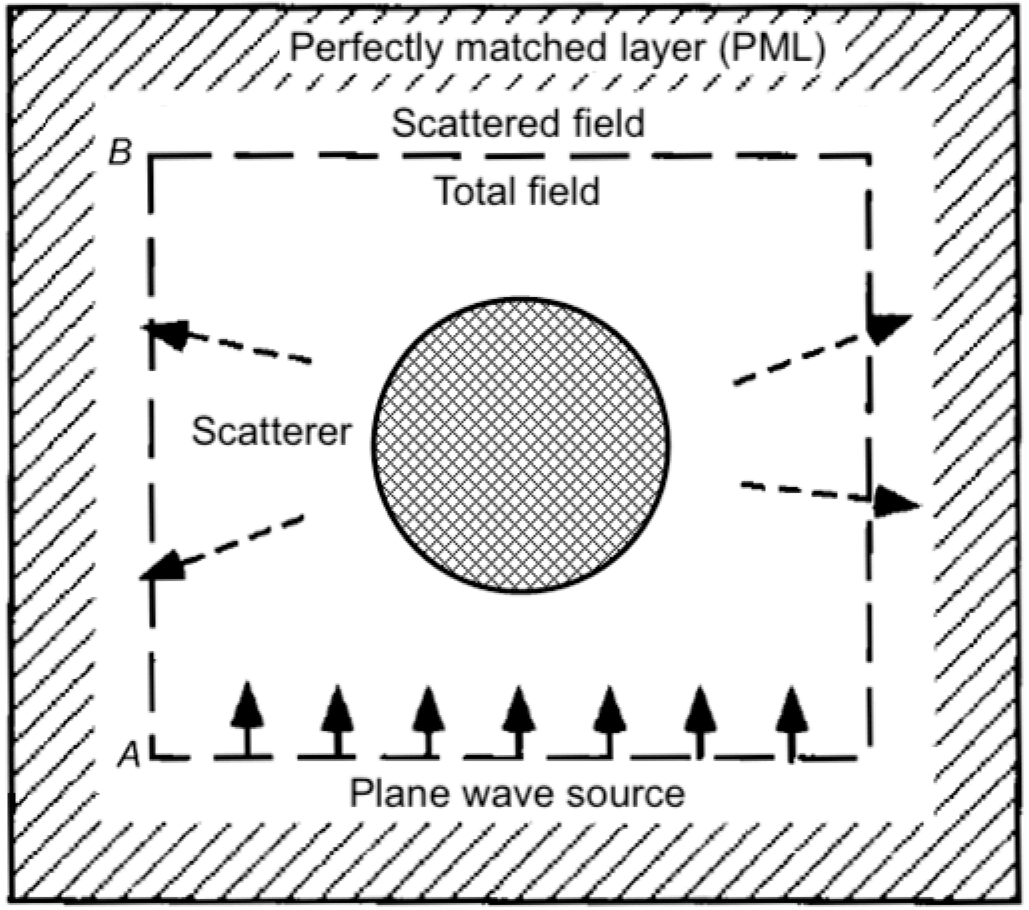
solution for homogeneous σ on surface: $u=\frac{I}{2\pi\sigma}\frac{1}{|\mathbf{r}-\mathbf{r}_s|}$

E-field $\mathbf{E}=-\frac{I}{2\pi\sigma}\frac{\mathbf{r}-\mathbf{r}_s}{|\mathbf{r}-\mathbf{r}_s|^3}$

normal direction $\mathbf{E}\cdot\mathbf{n}=-\frac{u}{|\mathbf{r}-\mathbf{r}_s|}\cos\phi$ purely geometric

$$\frac{\partial u}{\partial n}+\frac{\cos\phi}{|\mathbf{r}-\mathbf{r}_s|}=0$$

Perfectly matched layers



$$\frac{\partial}{\partial x} \to \frac{1}{1 + i\sigma/\omega} \frac{\partial}{\partial x}$$

$$x \to x + \frac{i}{\omega} \int^{x} \sigma(x')\mathrm{d}x'$$

Absorbing boundary conditions

wave equation (e.g. in 2D)

$$\frac{\partial^2 u}{\partial t^2} - v^2 \nabla^2 u = 0$$

Fourier transform in t and y (boundary direction) $\Rightarrow \omega, k$

$$\omega^2 \hat{u} - v^2 \frac{\partial^2 \hat{u}}{\partial x^2} + v^2 k^2 \hat{u} = 0$$

ordinary DE with solution $\hat{u} = \sum a_i e^{\lambda x}$ with $\lambda^2 = k^2 - \omega^2 / v^2$