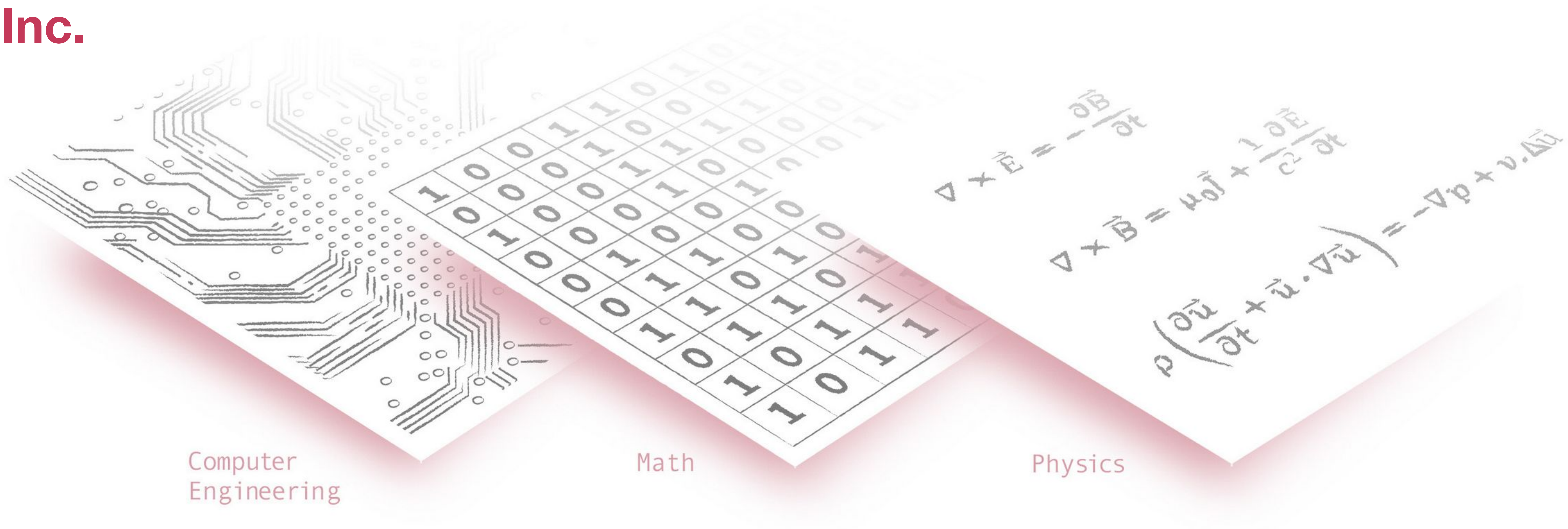
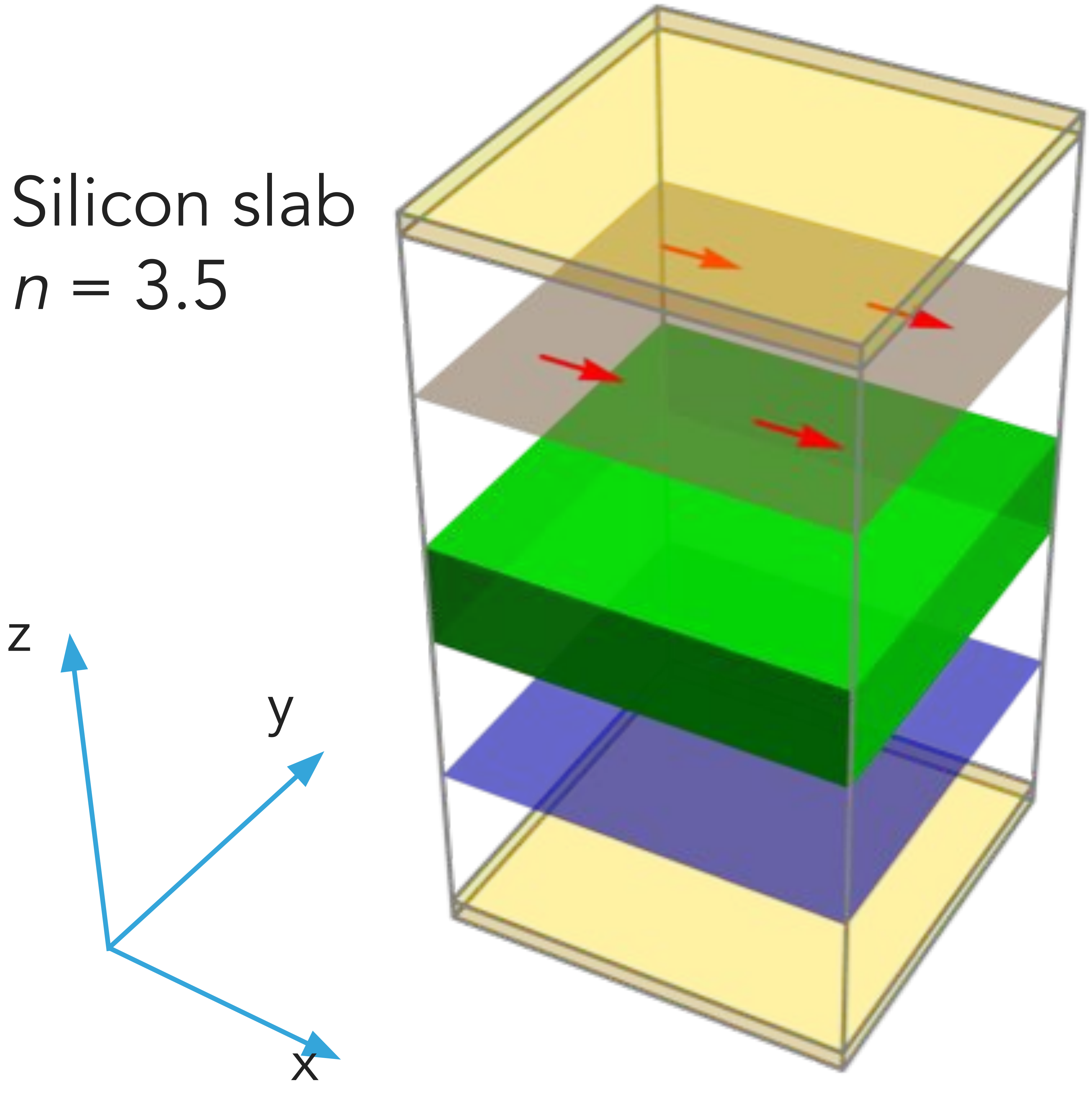
INTRO TO FDTD (10)

Flexcompute **Inc.**

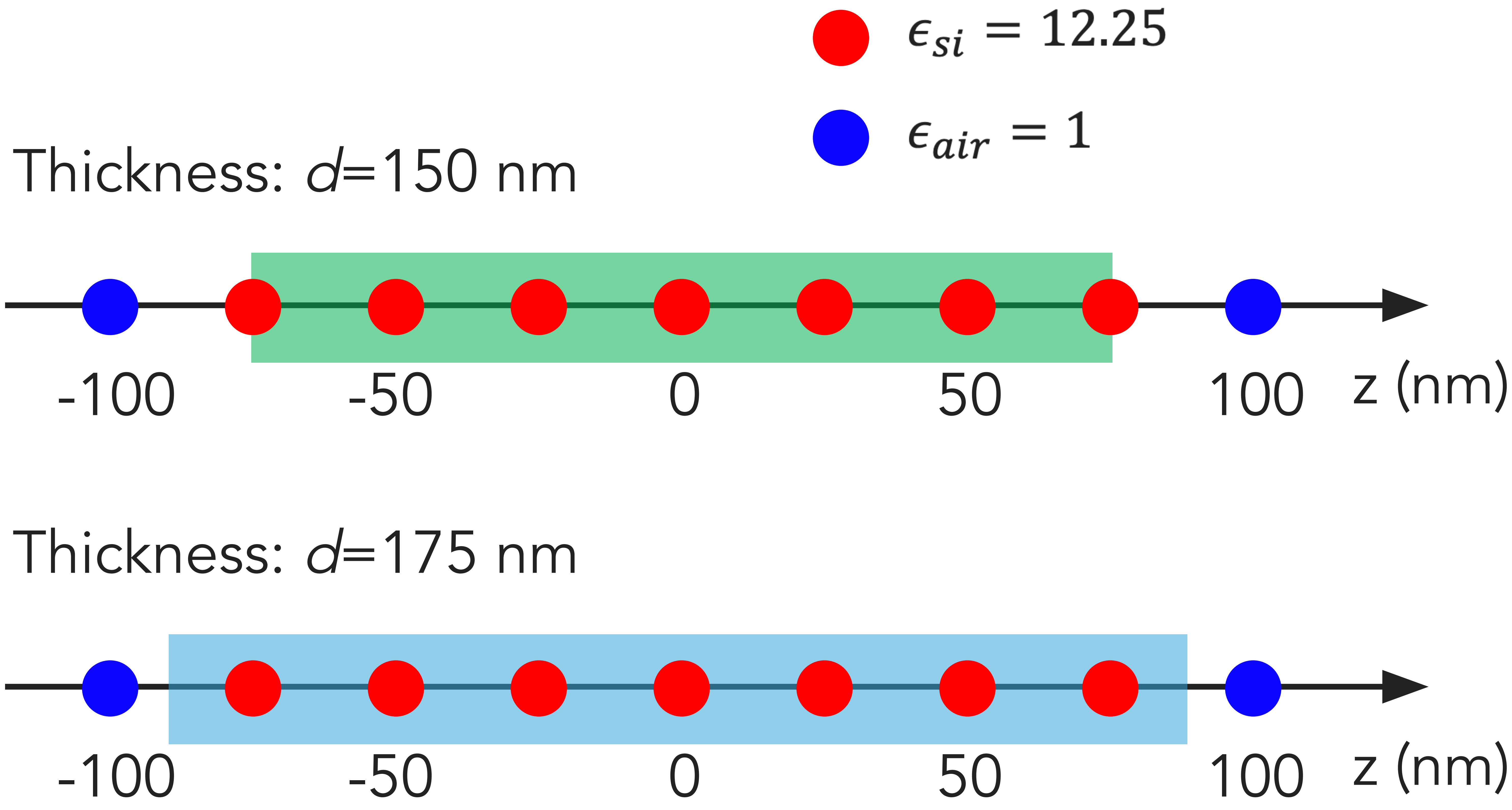


How to capture subpixel features in FDTD simulation?

Silicon slab
$n = 3.5$

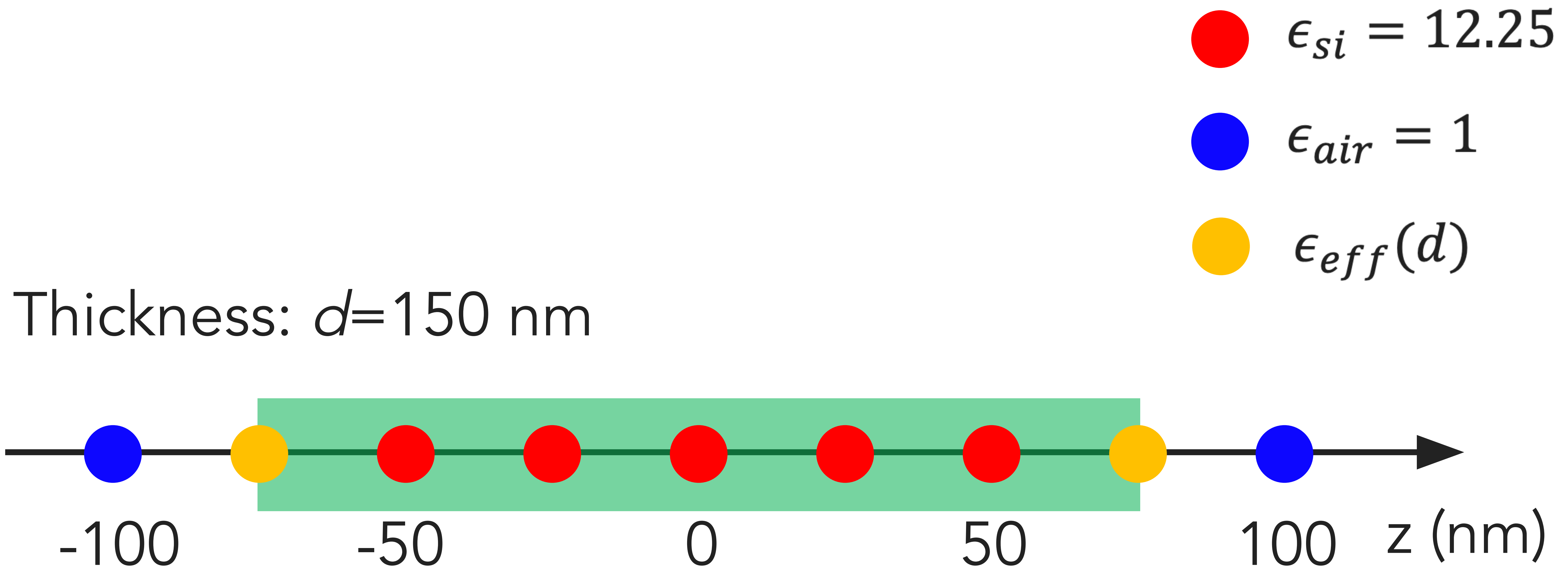


Spatial discretization: 25 nm



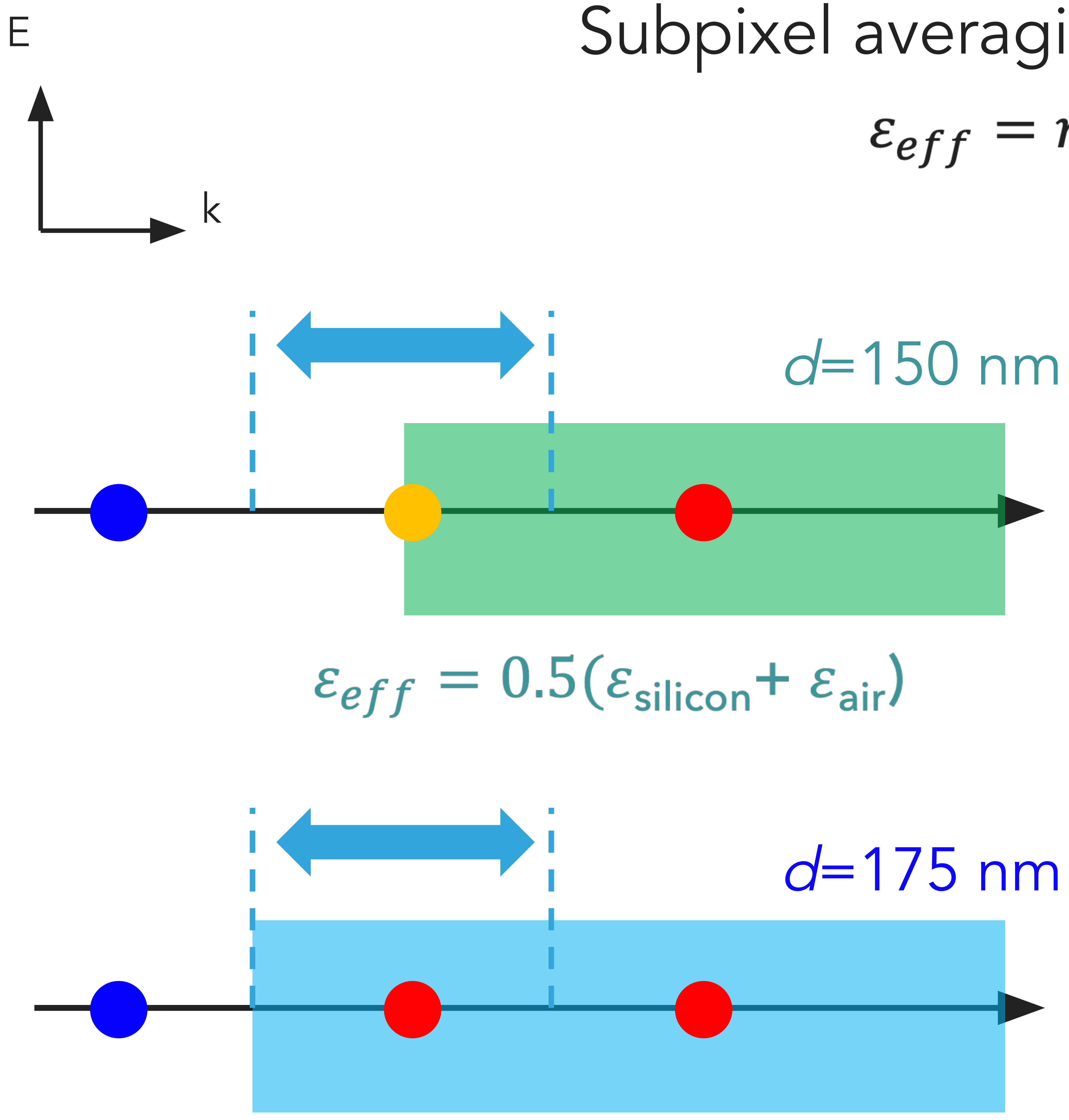
Both contain 7 grids: thickness 7×25 nm=175 nm

Subpixel averaging: effective permittivity near the boundary



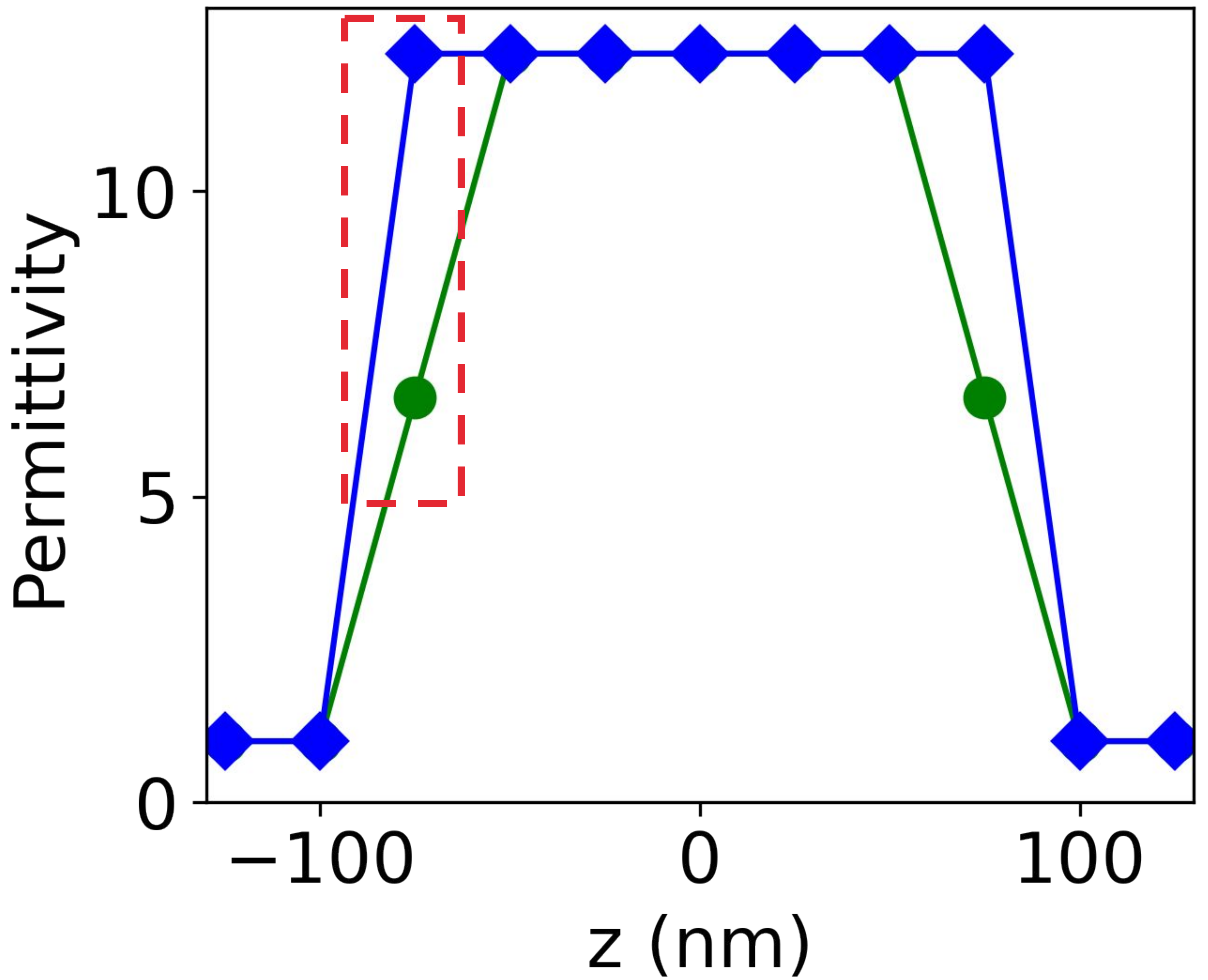
Subpixel averaging: a simple illustration

$$\varepsilon_{eff} = \eta\epsilon_{si} + (1-\eta)\epsilon_{air}$$



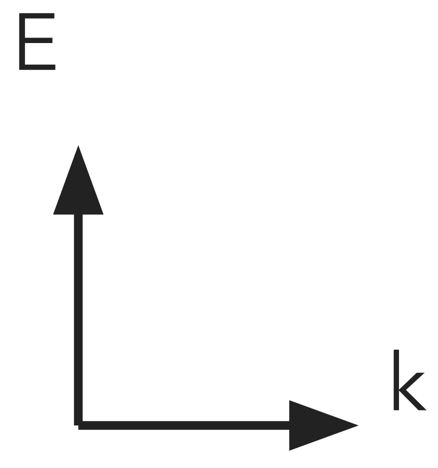
$\varepsilon_{eff} = 0.5(\varepsilon_{\text{silicon}} + \varepsilon_{\text{air}})$

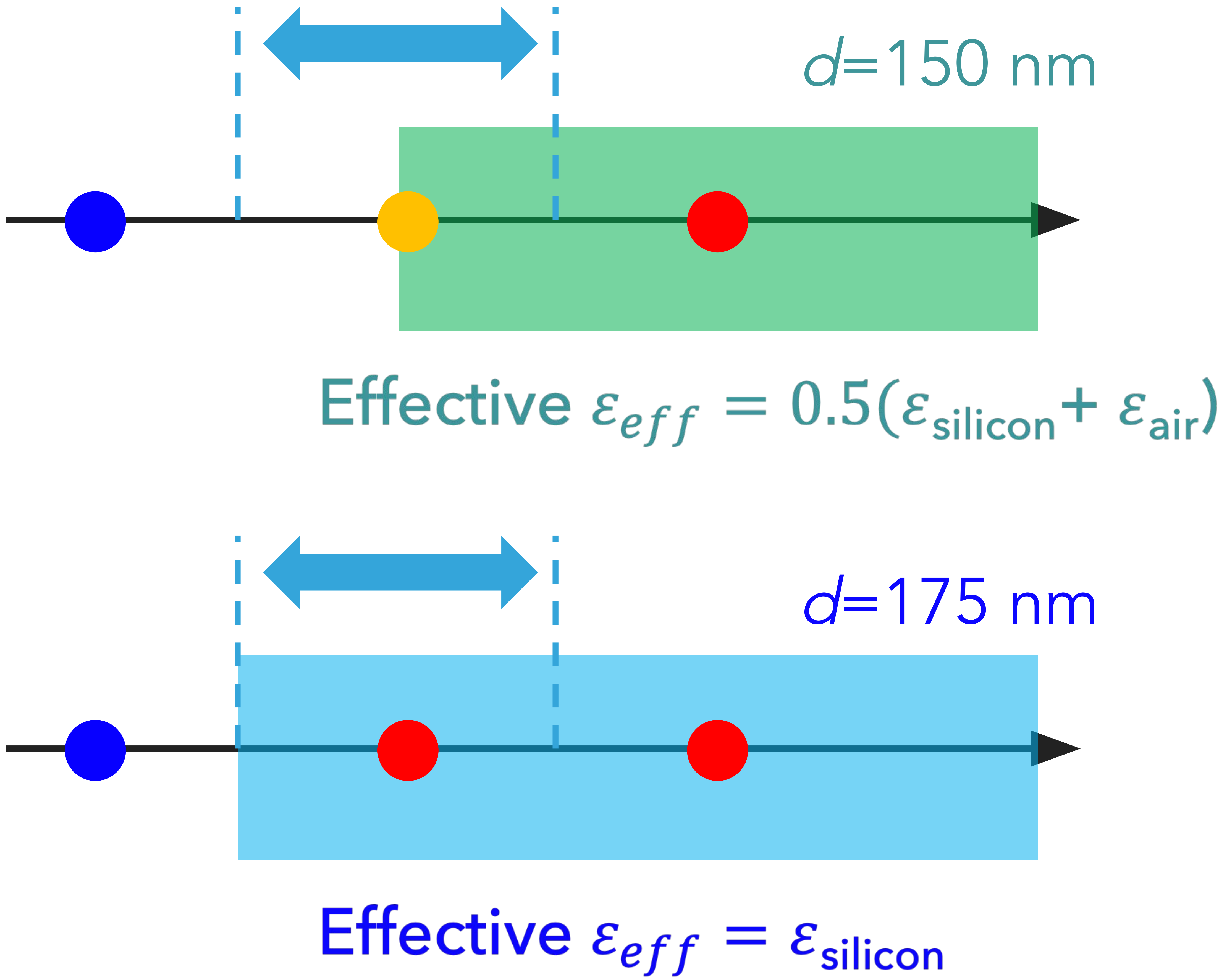
$\varepsilon_{eff} = \varepsilon_{\text{silicon}}$

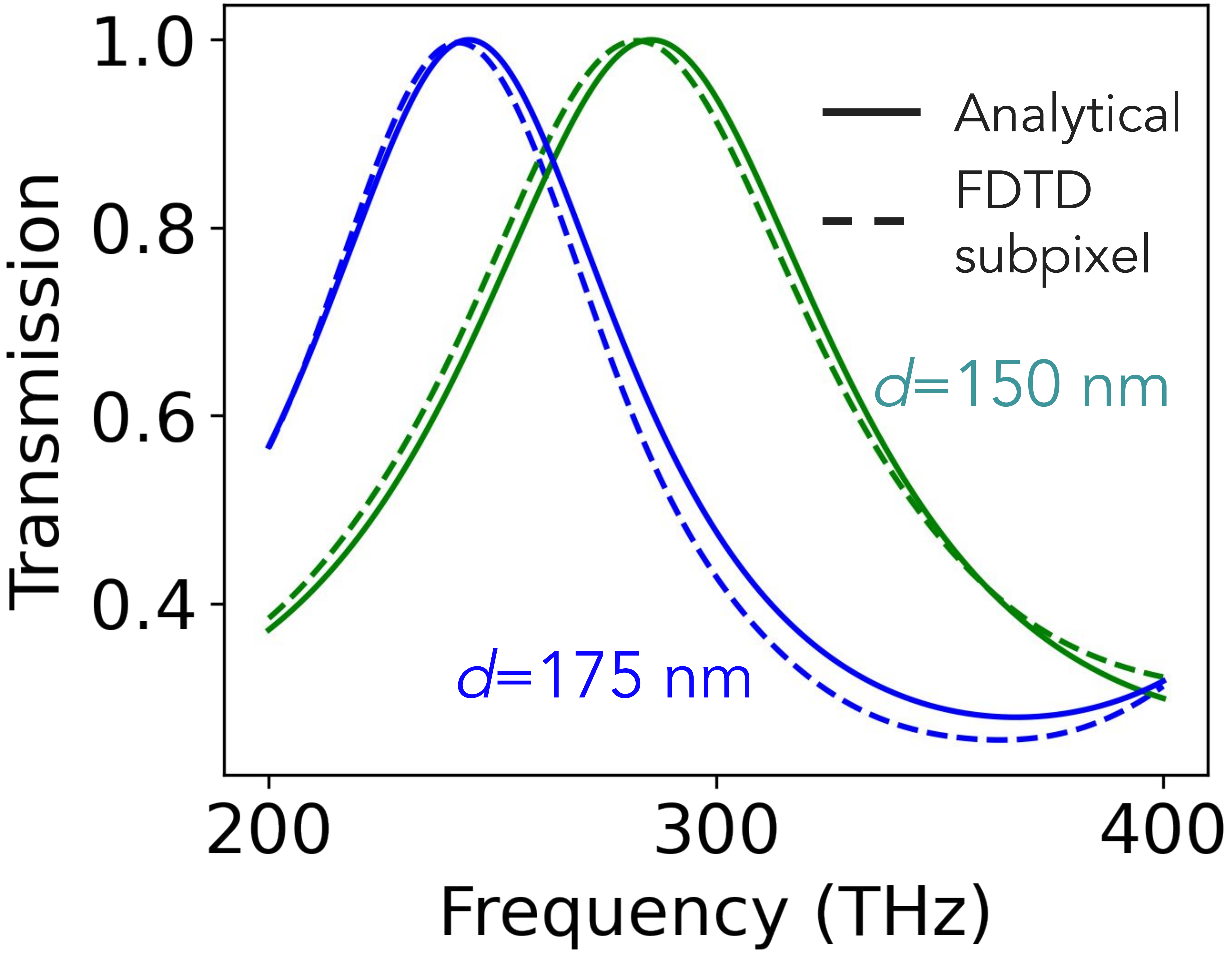


Subpixel averaging: a simple illustration

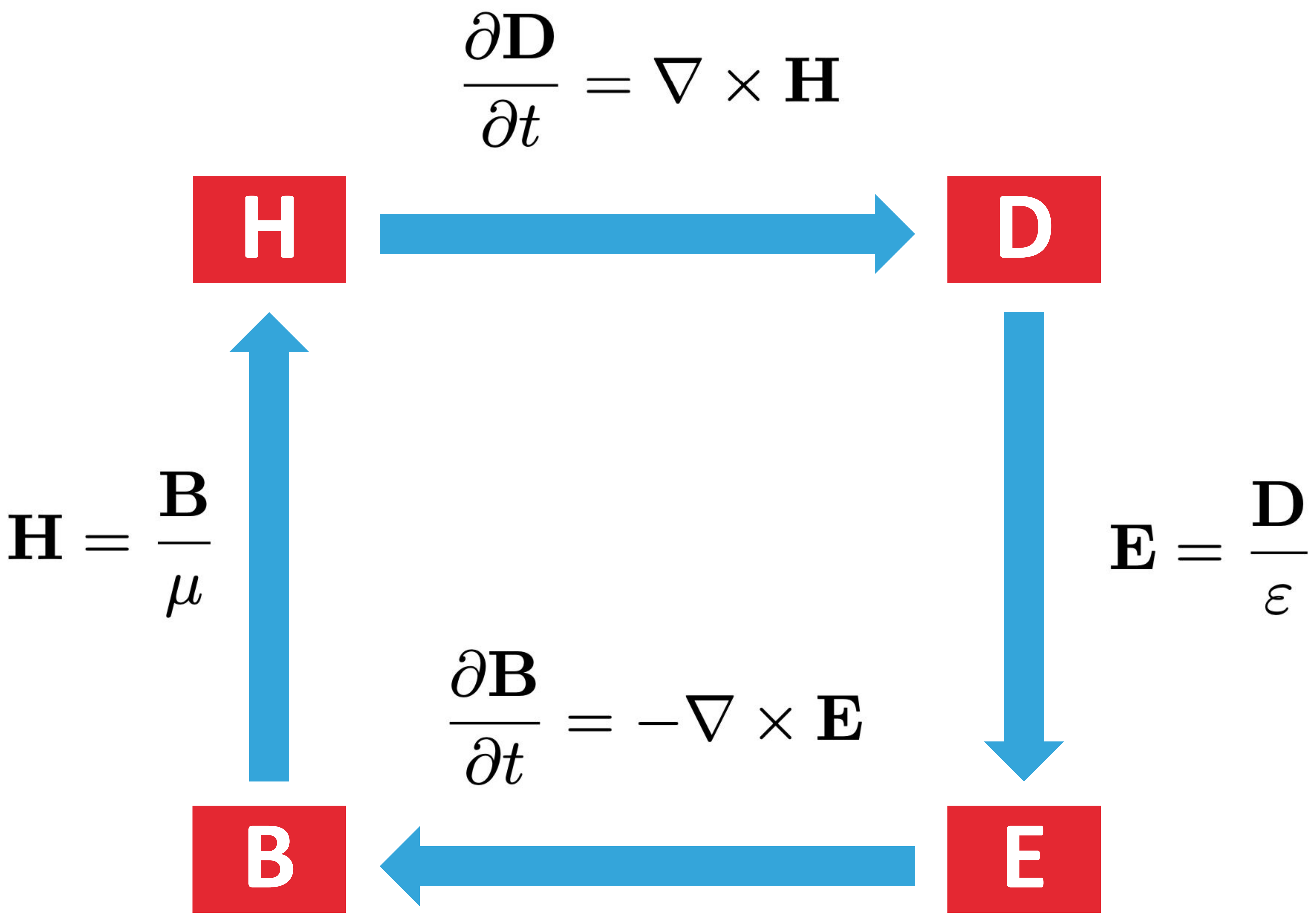
$$\varepsilon_{eff} = \eta\epsilon_{si} + (1-\eta)\epsilon_{air}$$



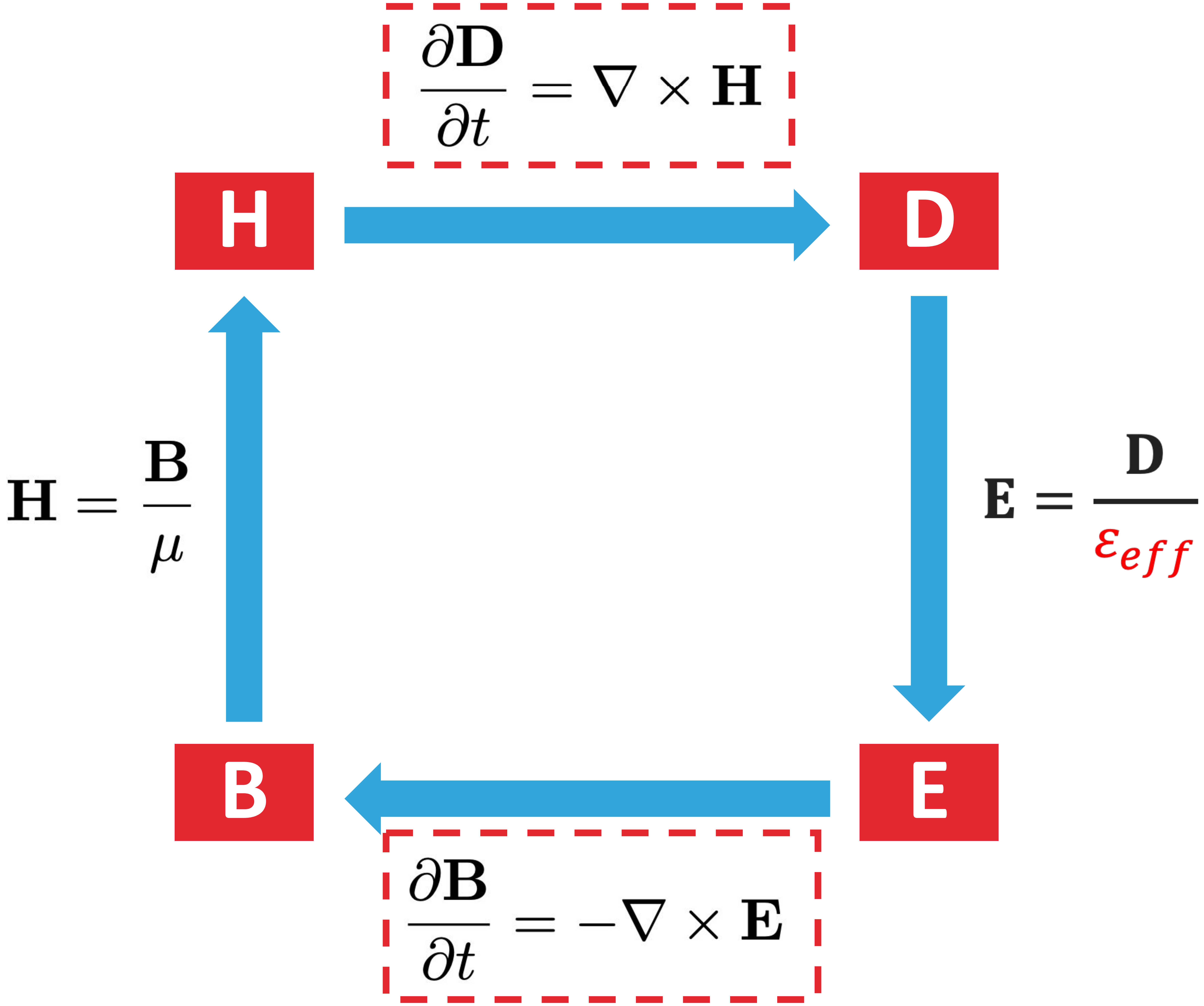




Field update procedures during time stepping in FDTD



How to capture subpixel features?

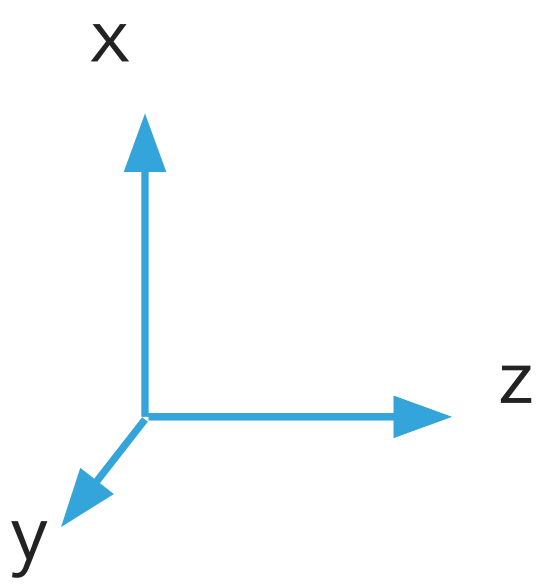


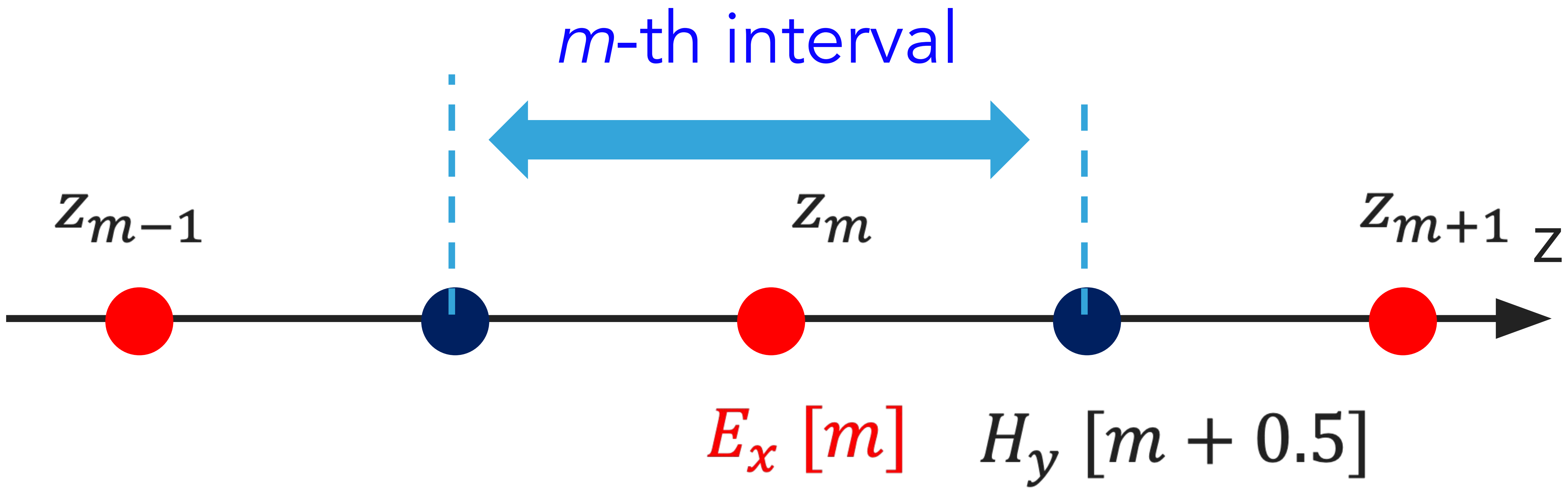
Average around the grid point to obtain ε_{eff} for approximating subpixel features.

Subpixel in 1D for tangential component

1D system: E_x and H_y

- Uniform along x and y-direction

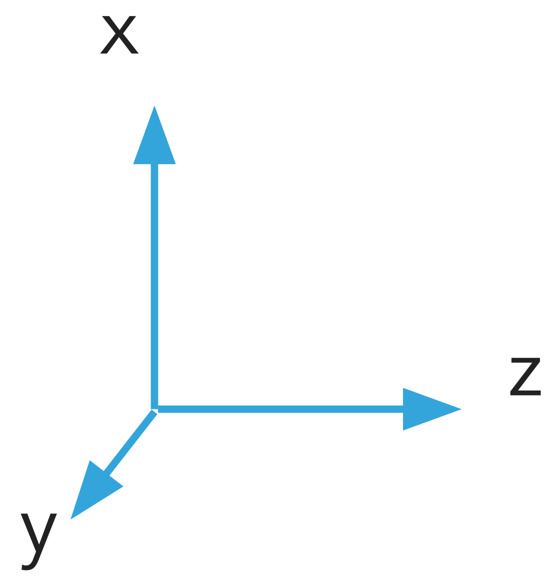




Integration over the *m*-th interval (Ampere's law):

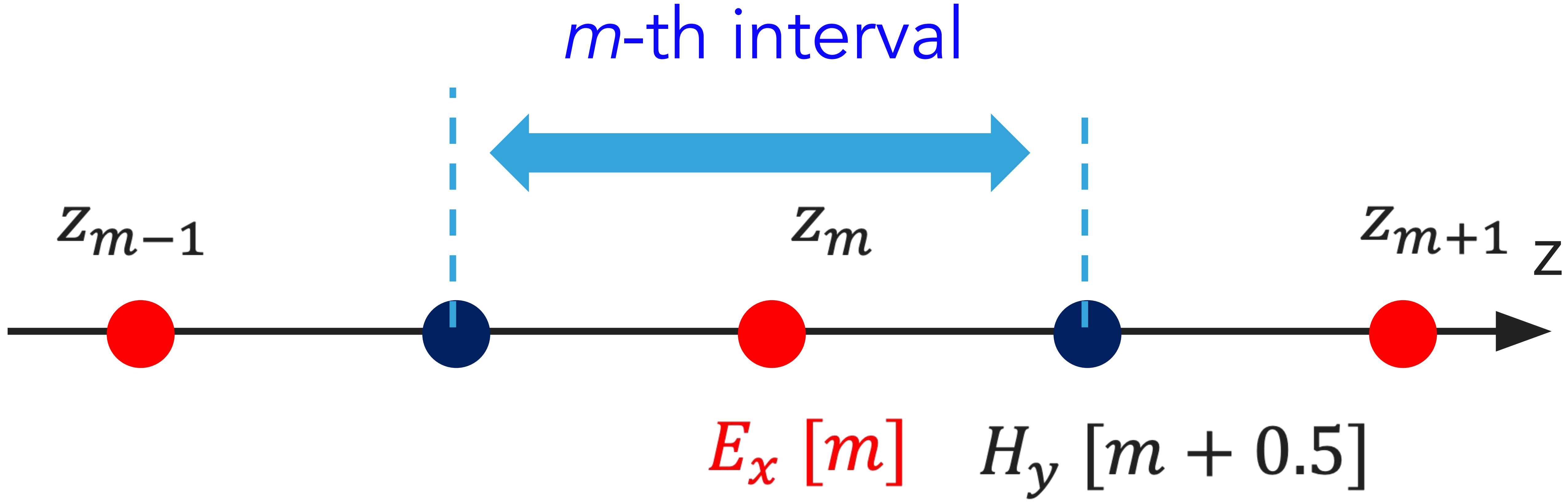
$$\frac{\partial \mathbf{D}}{\partial t} = \nabla \times \mathbf{H} \quad \Longrightarrow \quad \frac{\partial}{\partial t}\boxed{\int_{z_{m-0.5}}^{z_{m+0.5}} dz\, D_x(z)} = H_y(z_{m-0.5}) - H_y(z_{m+0.5})$$

Subpixel in 1D for tangential component



Interface condition for tangential fields :

- D_x is not continuous
- E_x is continuous

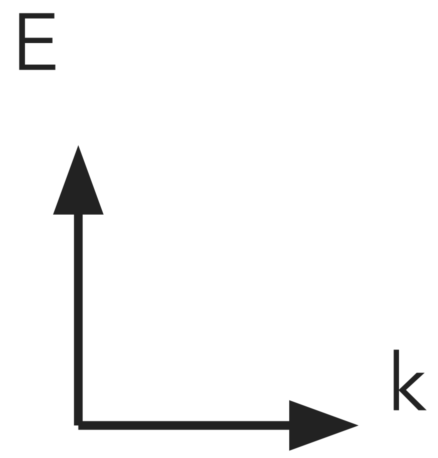


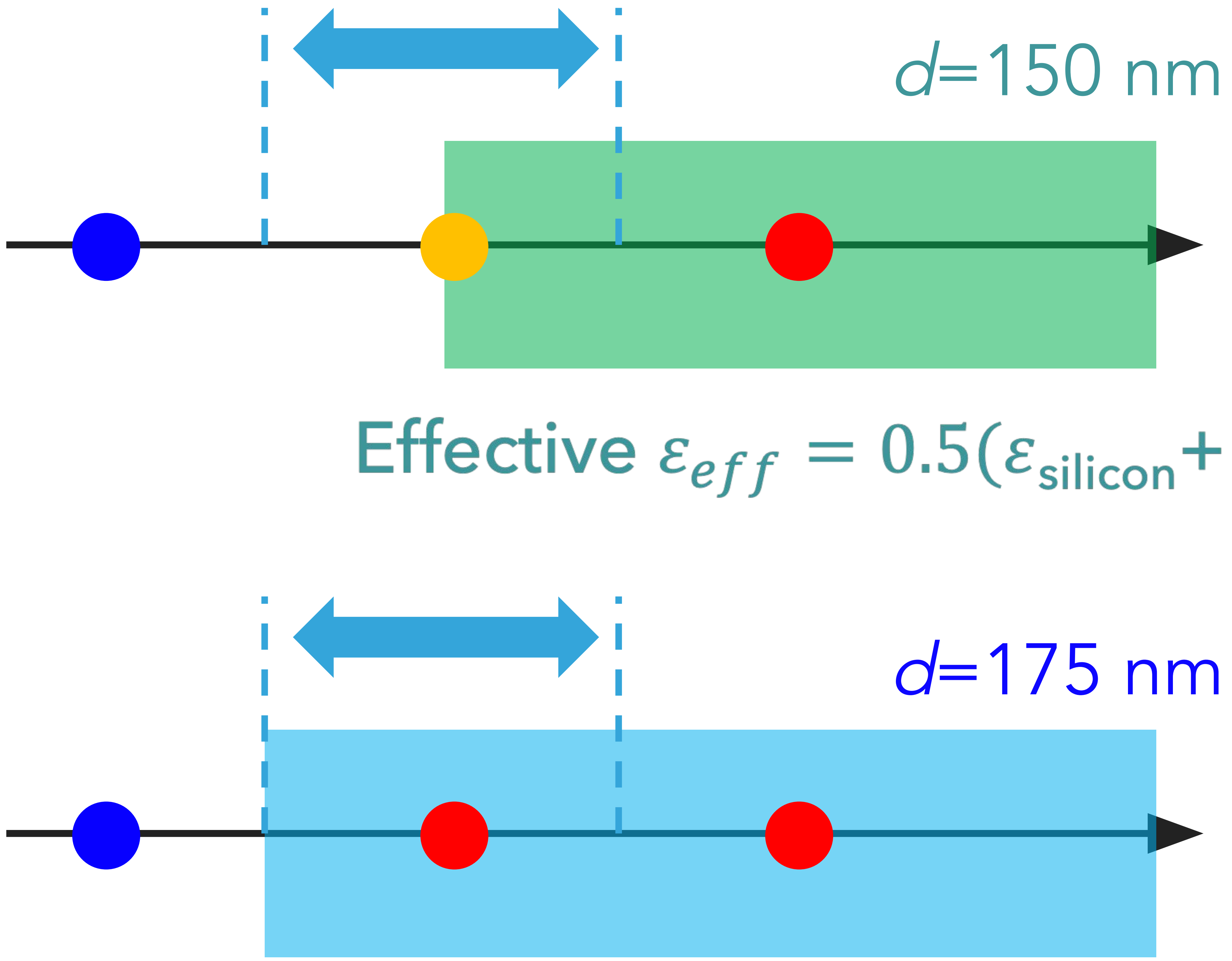
Approximate the continuous E_x as a constant in the interval:

$$\int_{z_{m-0.5}}^{z_{m+0.5}} dz\, D_x(z) \approx \varepsilon_{eff} E_x[z_m] \Delta z \quad \Longrightarrow \quad \varepsilon_{eff} = \int_{z_{m-0.5}}^{z_{m+0.5}} dz\, \varepsilon_{xx}(z)\ /\Delta z$$

Subpixel averaging: a simple illustration

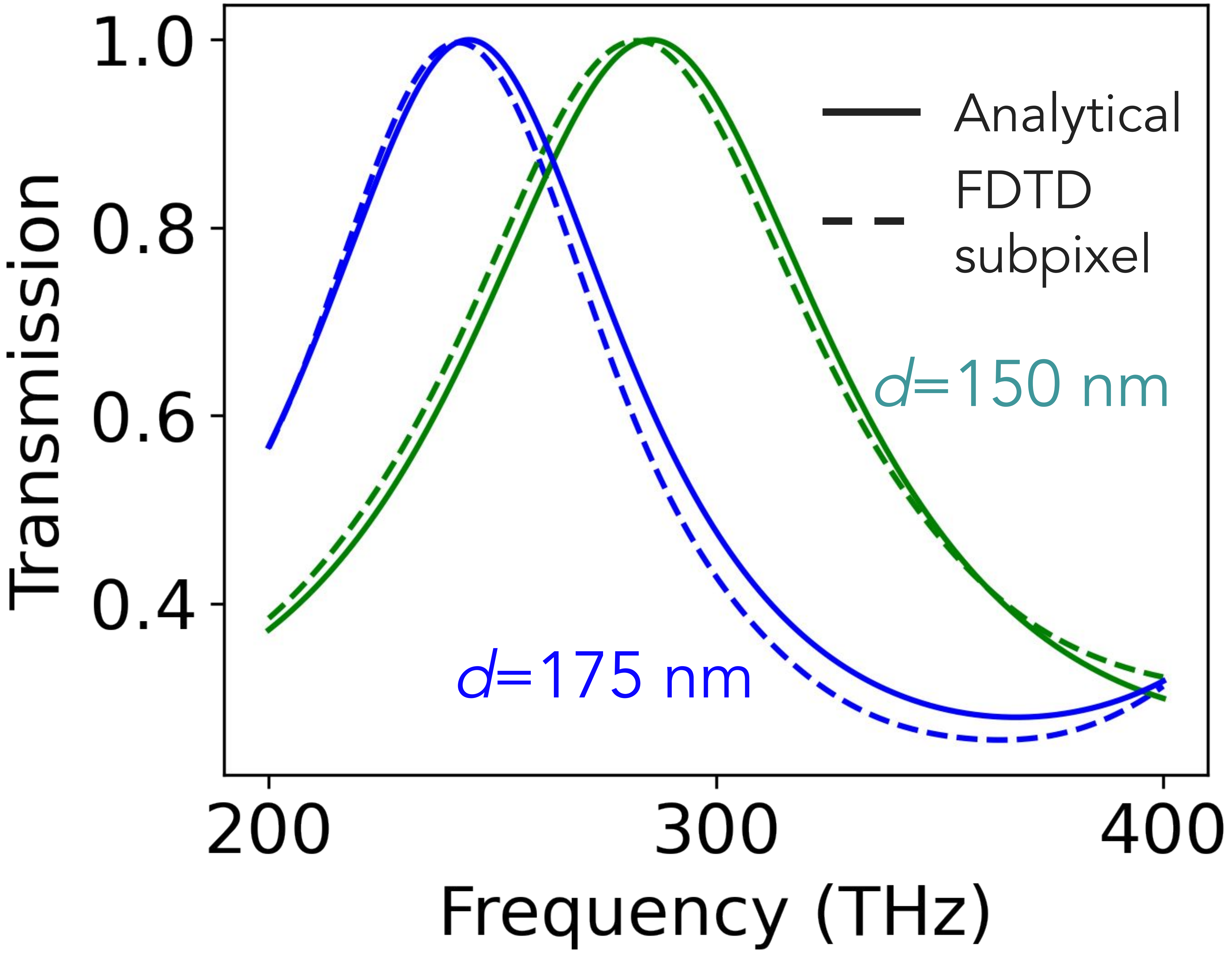
$$\varepsilon_{eff} = \eta\epsilon_{si} + (1-\eta)\epsilon_{air}$$





Effective $\varepsilon_{eff} = 0.5(\varepsilon_{silicon} + \varepsilon_{air})$

Effective $\varepsilon_{eff} = \varepsilon_{silicon}$

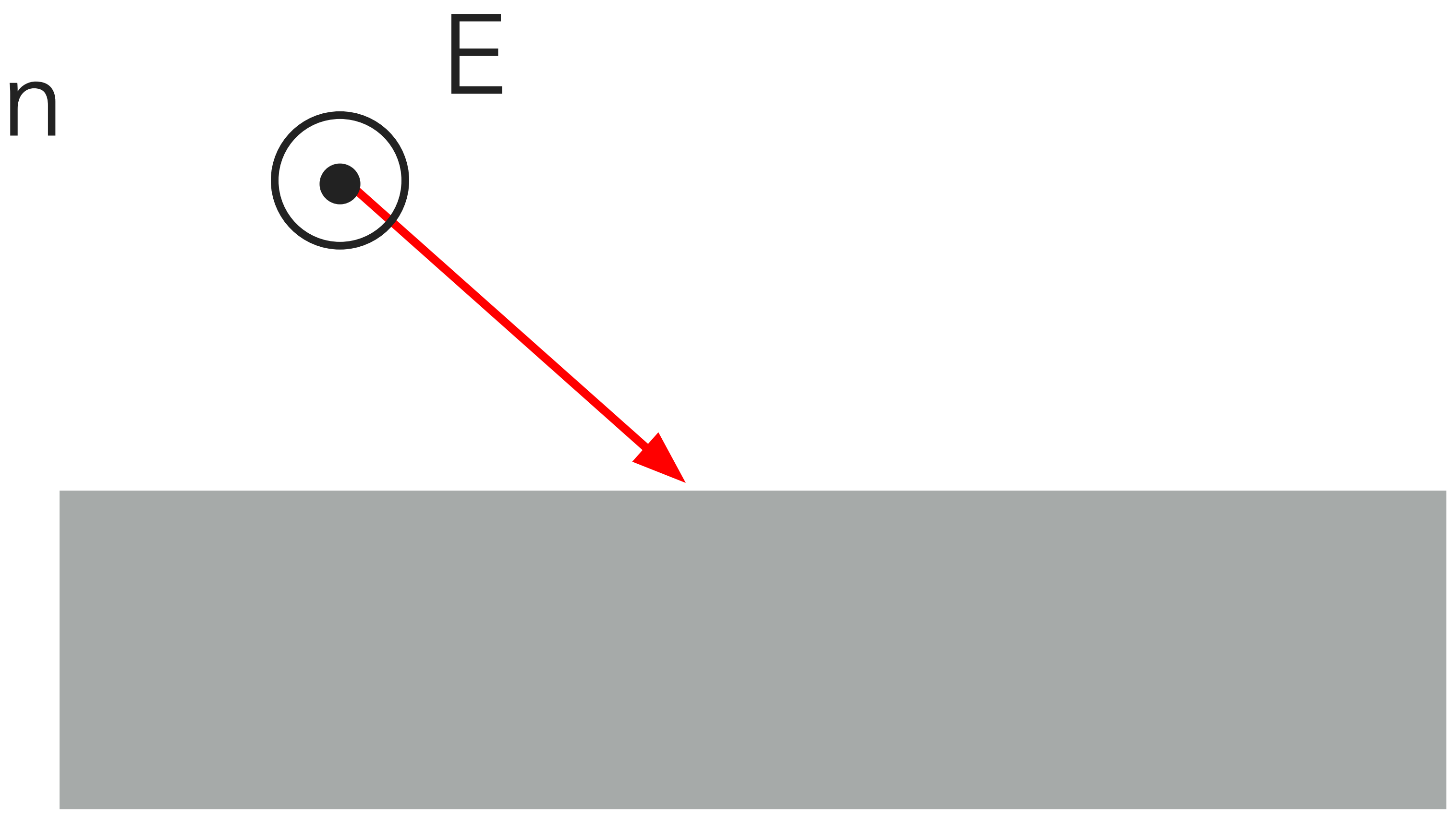


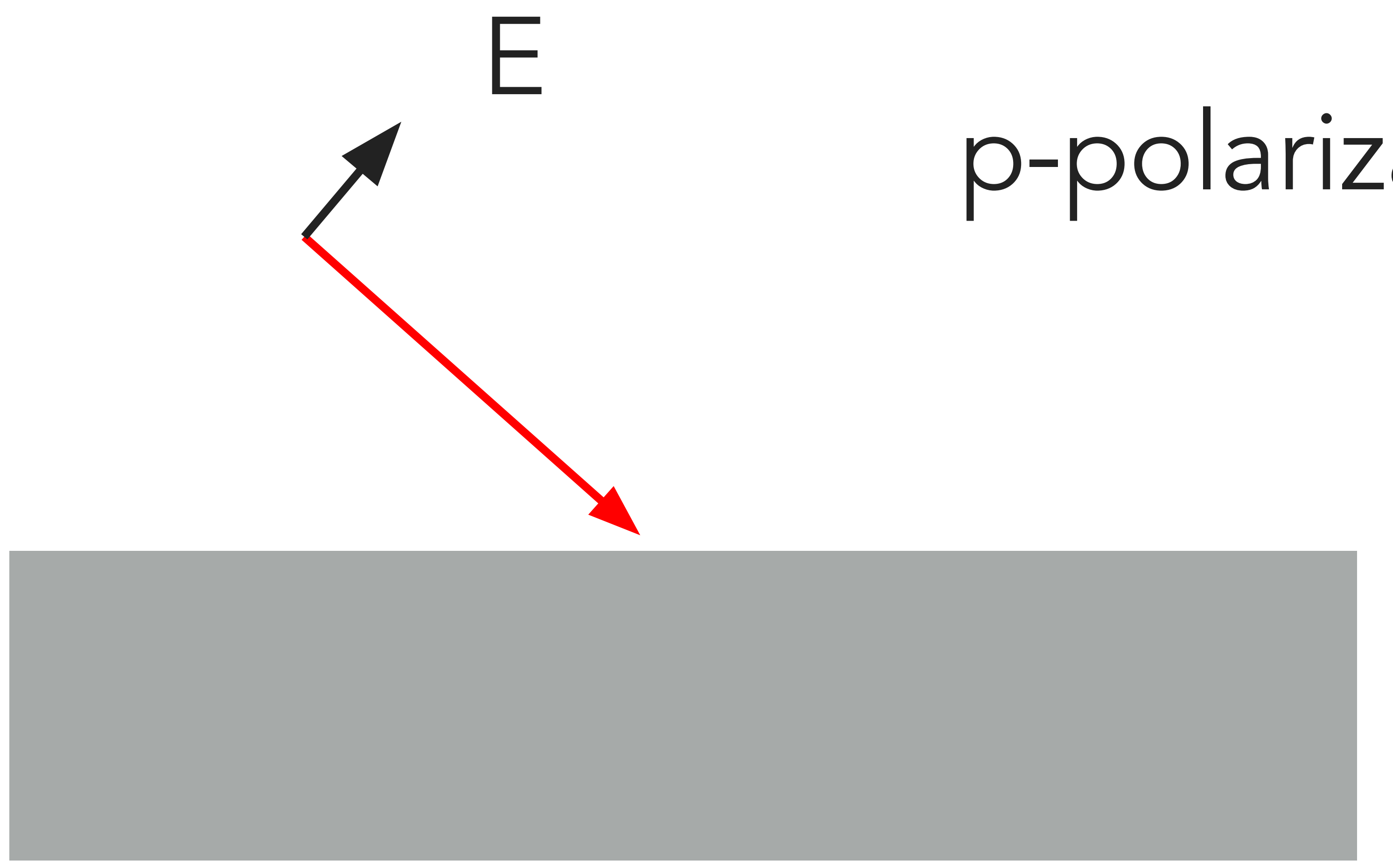
Subpixel averaging in practice

Depends on the field and surface orientation

- In 2D, the subpixel averaging for the s and p polarizations have different formulas.
- In 3D, the formulas require a tensorial average.

s-polarization





p-polarization