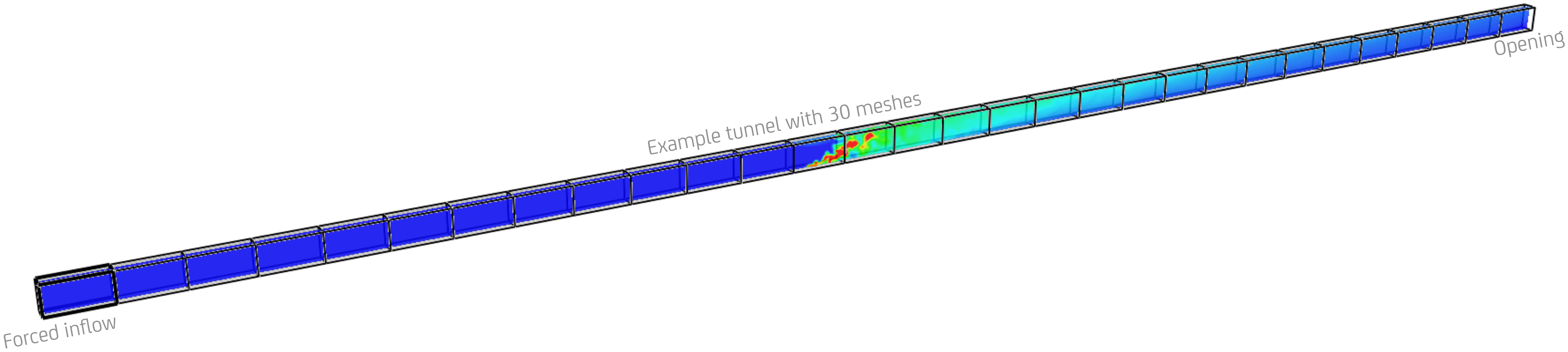
The Pressure Poisson Equation in FDS

Simplified versus complete version

Dr. Susanne Kilian
hhpberlin – Engineers for fire protection
10249 Berlin - Germany

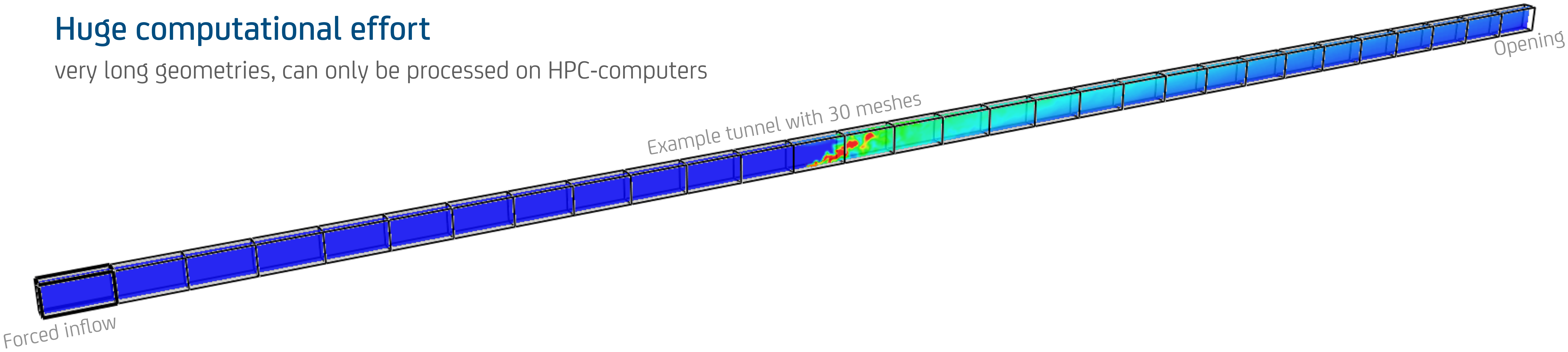
Tunnel simulations - a special challenge



Tunnel simulations - a special challenge

Huge computational effort

very long geometries, can only be processed on HPC-computers



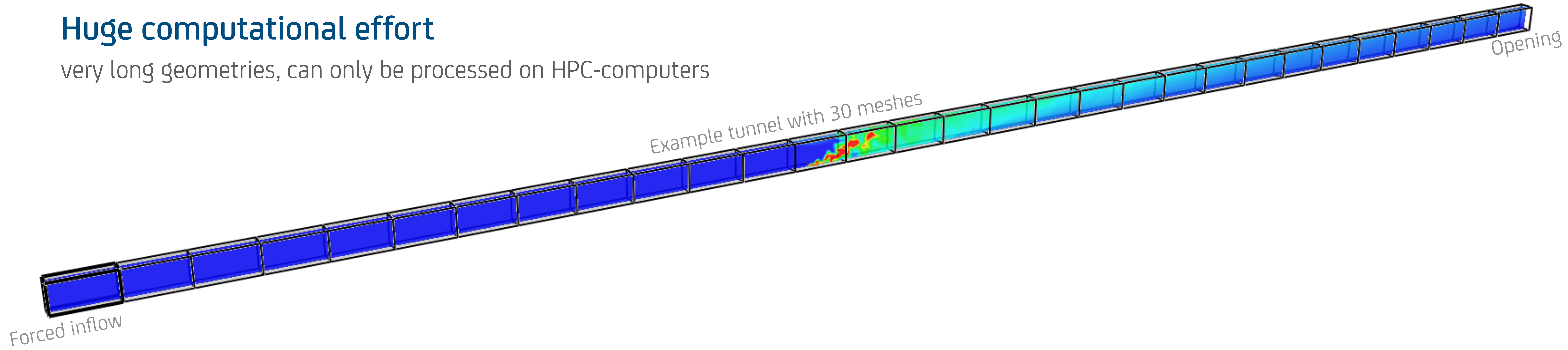
Tunnel simulations - a special challenge

Turbulently pulsating fire

non-steady heat release rate driving the flow

Huge computational effort

very long geometries, can only be processed on HPC-computers



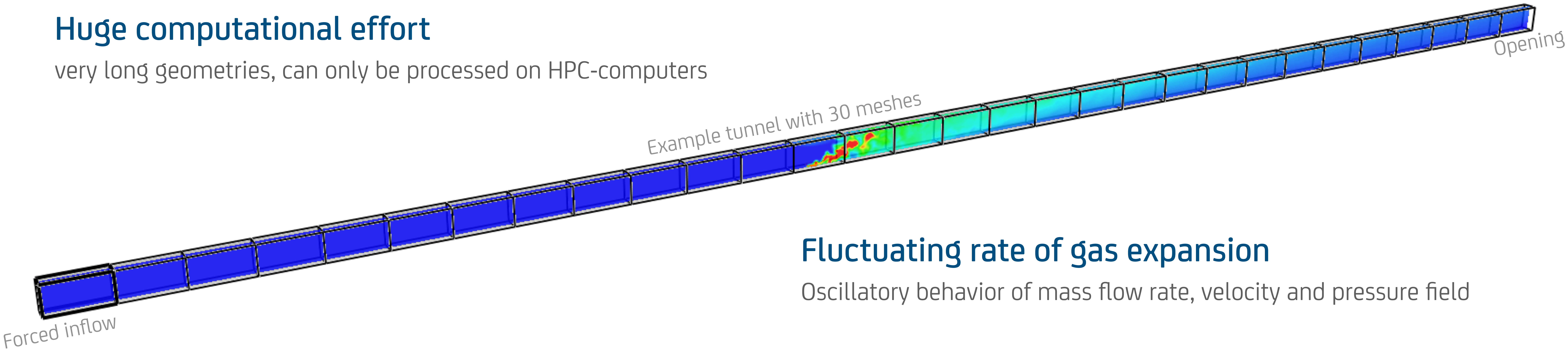
Tunnel simulations - a special challenge

Turbulently pulsating fire

non-steady heat release rate driving the flow

Huge computational effort

very long geometries, can only be processed on HPC-computers



Fluctuating rate of gas expansion

Oscillatory behavior of mass flow rate, velocity and pressure field

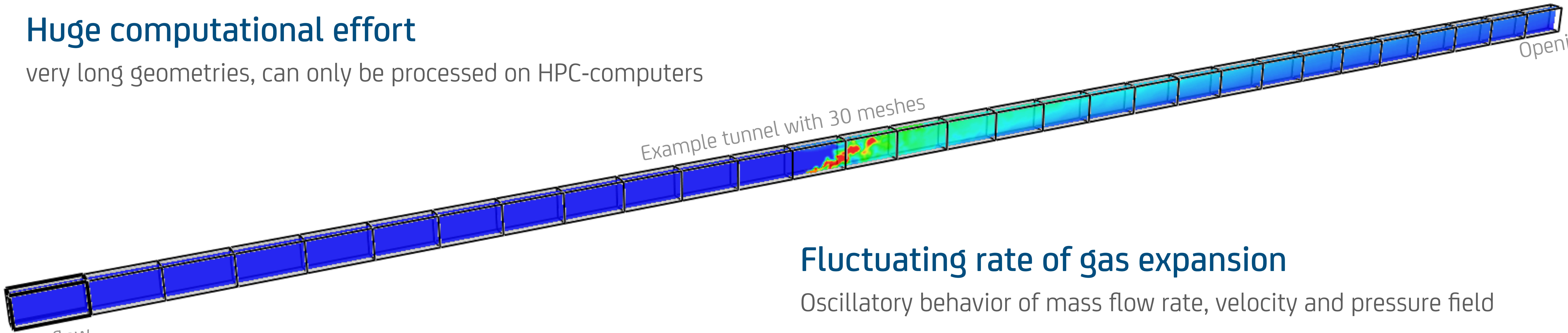
Tunnel simulations - a special challenge

Turbulently pulsating fire

non-steady heat release rate driving the flow

Huge computational effort

very long geometries, can only be processed on HPC-computers



Fluctuating rate of gas expansion

Oscillatory behavior of mass flow rate, velocity and pressure field

Long sealed box

flow is ‚trapped', excess volumes can only escape at the ends

Challenges related to tunnel simulations

Low-Mach formulation

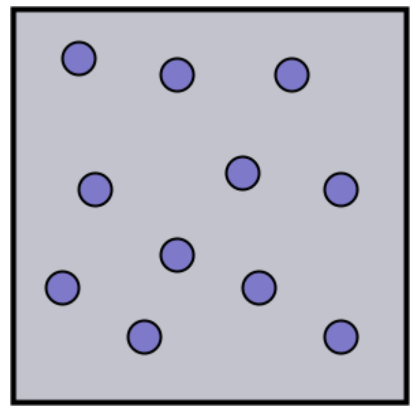
Parallelization

Interaction of different components

Simplified pressure equation

Momentum equation - Newton's 2. Law

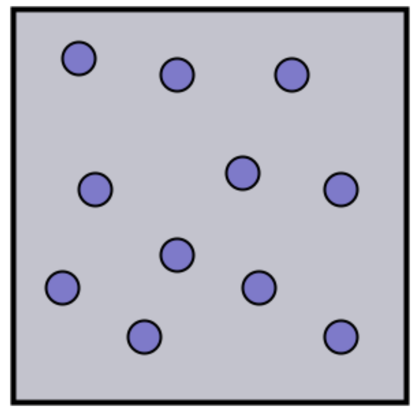
$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}^{\text{Force}}$$



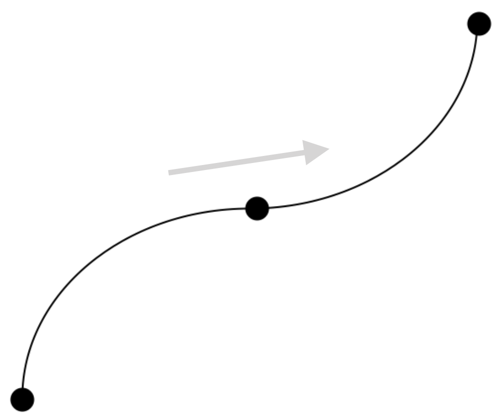
Density

Momentum equation - Newton's 2. Law

$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}^{\text{Force}}$$



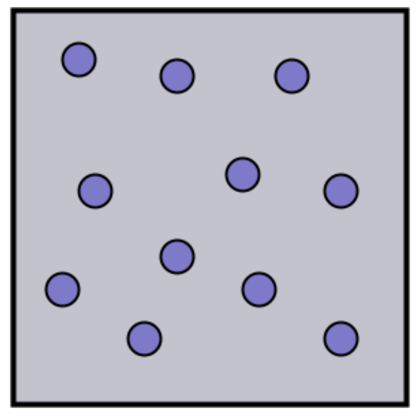
Density



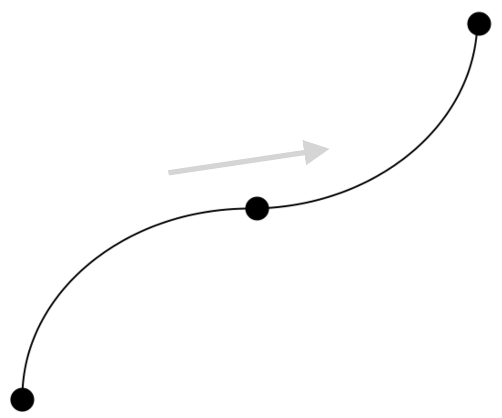
Time rate of
change of velocity

Momentum equation - Newton’s 2. Law

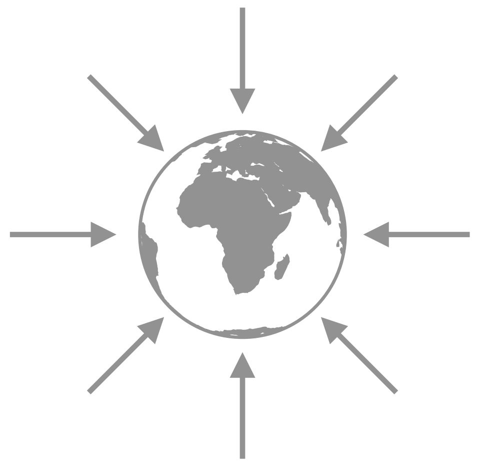
$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}^{\text{Force}}$$



Density



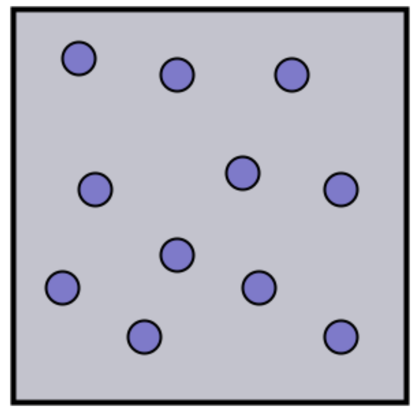
Time rate of change of velocity



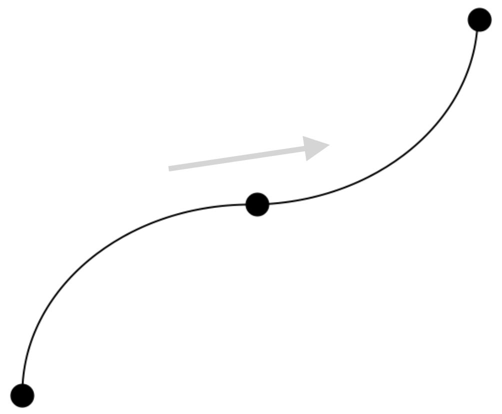
Force of gravity

Momentum equation - Newton's 2. Law

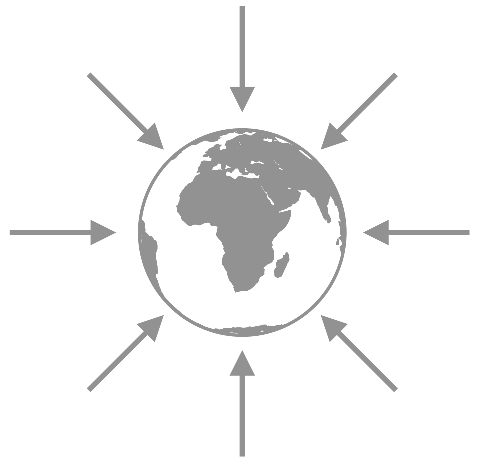
$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}^{\text{Force}}$$



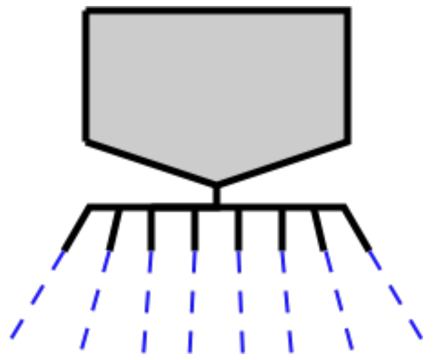
Density



Time rate of
change of velocity



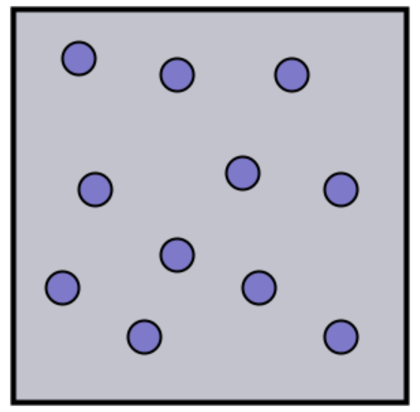
Force of
gravity



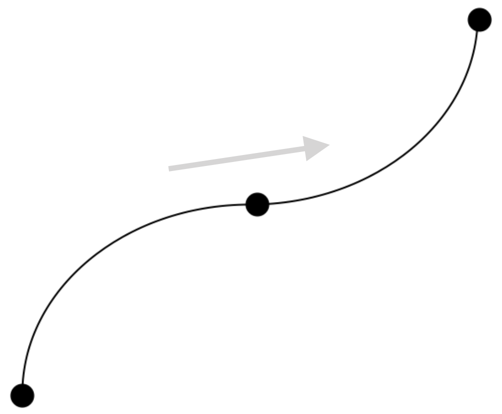
Moving particles
(e.g. water droplets)

Momentum equation - Newton's 2. Law

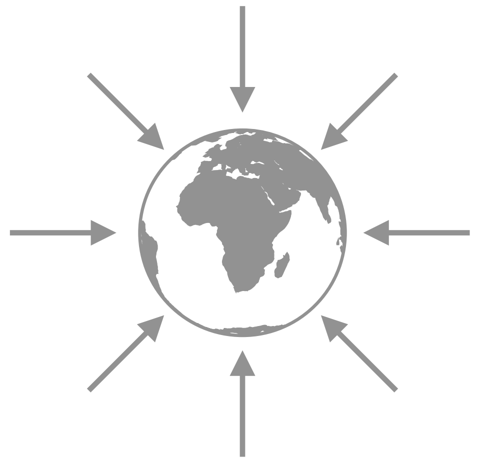
$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}^{\text{Force}}$$



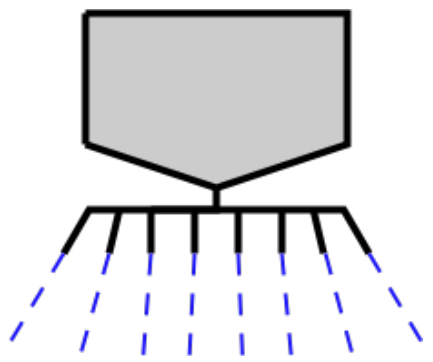
Density



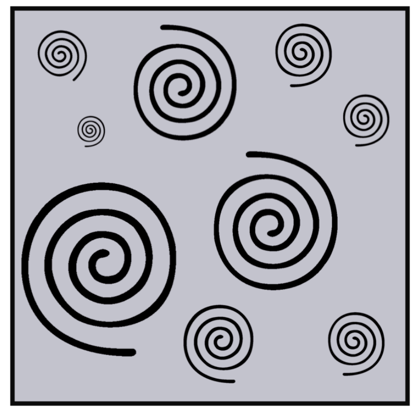
Time rate of change of velocity



Force of gravity



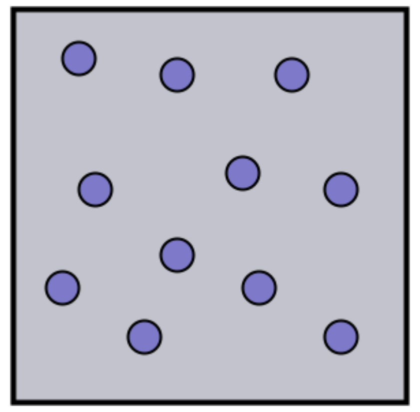
Moving particles (e.g. water droplets)



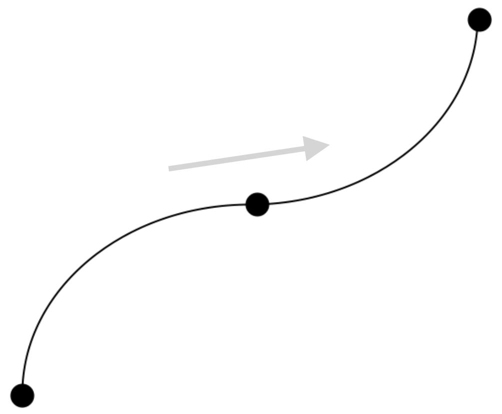
Viscosity (i.e. turbulence)

Momentum equation - Newton's 2. Law

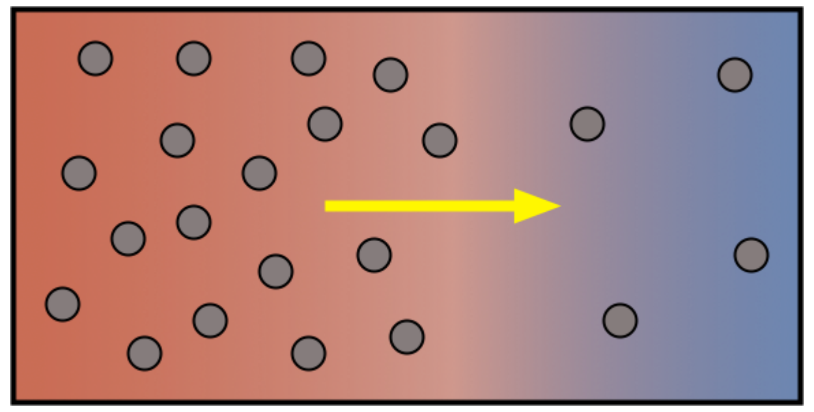
$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}^{\text{Force}}$$



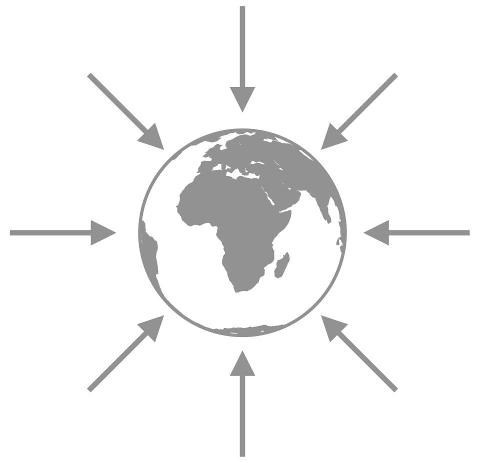
Density



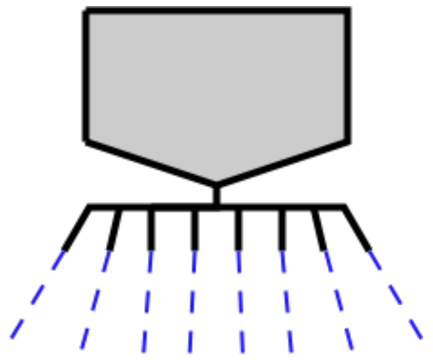
Time rate of change of velocity



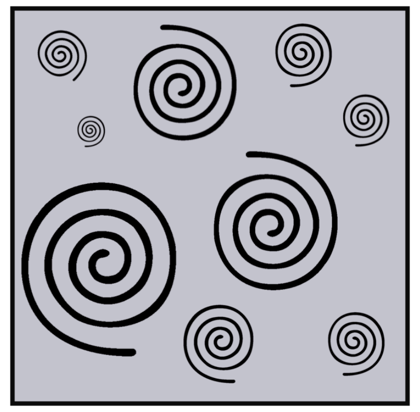
Pressure differences



Force of gravity



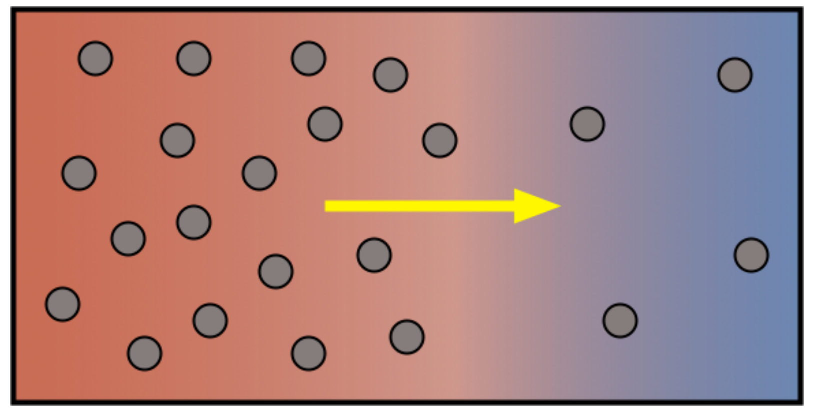
Moving particles (e.g. water droplets)



Viscosity (i.e. turbulence)

Momentum equation - Newton's 2. Law

$$\underbrace{\rho}_{\text{Mass}} \cdot \underbrace{\frac{D\mathbf{u}}{Dt}}_{\text{Acceleration}} = \underbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}}_{\text{Force}}$$

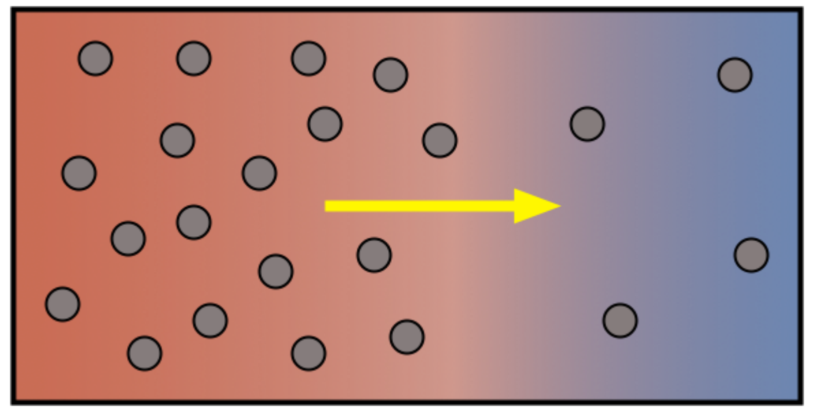


Pressure differences

Momentum equation - Newton's 2. Law

$$\overbrace{\rho}^{\text{Mass}} \cdot \overbrace{\frac{D\mathbf{u}}{Dt}}^{\text{Acceleration}} = \overbrace{-\nabla p + \rho\mathbf{g} + \mathbf{f} + \nabla \cdot \tau_{ij}}^{\text{Force}}$$

How to derive a governing equation for the pressure from this?



Pressure differences

Splitting of pressure for Low-Mach flows

Separation of effects on different scales

$$\rho \cdot \frac{D\mathbf{u}}{Dt} = \underbrace{-\nabla p}_{\nabla \tilde{p} \,+\, \nabla \overline{p}_m} + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}$$

Splitting of pressure for Low-Mach flows

Separation of effects on different scales

$$\rho \cdot \frac{D\mathbf{u}}{Dt} = \underbrace{-\nabla p}_{\nabla \tilde{p} \,+\, \nabla \overline{p}_m} + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}$$

Background pressure $\overline{p}_m(z,t)$

thermodynamic

Represents stratification of atmosphere

Splitting of pressure for Low-Mach flows

Separation of effects on different scales

$$\rho \cdot \frac{D\mathbf{u}}{Dt} = \underbrace{-\nabla p}_{\nabla \tilde{p} \,+\, \nabla \overline{p}_m} + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}$$

Perturbation pressure $\tilde{p}(\mathbf{x}, t)$

hydrodynamic

Resolves small-scaled fluctuations

Background pressure $\overline{p}_m(z, t)$

thermodynamic

Represents stratification of atmosphere

Splitting of pressure for Low-Mach flows

Separation of effects on different scales

$$\rho \cdot \frac{D\mathbf{u}}{Dt} = -\underbrace{\nabla p}_{\nabla \tilde{p} + \nabla \overline{p}_m} + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}$$

$$\nabla \tilde{p} + \nabla \overline{p}_m$$

multiple compartments
‚m' possible

Absorbed on average

Equation of state only considers background pressure

Perturbation pressure $\tilde{p}(\mathbf{x}, t)$
hydrodynamic
Resolves small-scaled fluctuations

Background pressure $\overline{p}_m(z, t)$
thermodynamic
Represents stratification of atmosphere

Splitting of pressure for Low-Mach flows

Separation of effects on different scales

$$\rho \cdot \frac{D\mathbf{u}}{Dt} = - \underbrace{\nabla p}_{\nabla \tilde{p} \; + \; \nabla \overline{p}_m} + \rho\mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}$$

multiple compartments ‚m' possible

Moved to the left

Pressure equation only considers perturbation pressure

Perturbation pressure $\tilde{p}(\mathbf{x}, t)$

hydrodynamic

Resolves small-scaled fluctuations

Absorbed on average

Equation of state only considers background pressure

Background pressure $p_m(z, t)$

thermodynamic

Represents stratification of atmosphere

Splitting of pressure for Low-Mach flows

Separation of effects on different scales

$$\rho \cdot \frac{D\mathbf{u}}{Dt} = \underbrace{-\nabla p}_{\nabla \tilde{p} \, + \, \nabla \bar{p}_m} + \rho \mathbf{g} + \mathbf{f}_b + \nabla \cdot \tau_{ij}$$

multiple compartments ‚m' possible

Moved to the left

Perturbation pressure $\tilde{p}(\mathbf{x}, t)$

Pressure equation only considers perturbation pressure

hydrodynamic

Resolves small-scaled fluctuations

Absorbed on average

Background pressure $\bar{p}_m(z, t)$

Equation of state only considers background pressure

thermodynamic

Represents stratification of atmosphere

$\longrightarrow$ CFL-condition only based on velocity of flow (10’s of m/s) and not velocity of sound (343 m/s)

Pressure equations - two variants

Complete version

$$\nabla \cdot \left[\frac{1}{\rho} \nabla \tilde{p} \right] = \hat{R}$$

Simplified version

$$\nabla^2 \underbrace{H}_{\frac{\tilde{p}}{\rho} + \frac{|\mathbf{u}|^2}{2}} = R$$

Bernoulli pressure

Pressure equations - two variants

Complete version

$$\nabla \cdot \left[\frac{1}{\rho} \nabla \tilde{p}\right] = \hat{R}$$

Simplified version

$$\nabla^2 \underbrace{H}_{\frac{\tilde{p}}{\rho} + \frac{|\mathbf{u}|^2}{2}} = R$$

Bernoulli pressure

Solved at least twice per time step

⟶ Elliptic Poisson-type equations with infinite propagation velocity

Discretized equations - two variants

Complete version

$$A_\rho^n \, x = \hat{b}$$

Simplified version

$$A \, x = b$$

Two variants of the pressure equation

Complete version

$$A_\rho^n \, x = \hat{b}$$

Different values

from cell to cell
→ very robust solver required

from timestep to timestep
→ continuous matrix rebuild needed

Less efficient to solve, but accurate

Simplified version

$$A \, x = b$$

Same values

from cell to cell
→ optimized FFT solver usable

from timestep to timestep
→ only one matrix build needed

Highly efficient to solve, but not fully accurate

Two variants of the pressure equation

Complete version

$$A_\rho \, x = b$$

Default in FDS

Different values

from cell to cell
→ very robust solver required

from timestep to timestep
→ continuous matrix rebuild needed

Less efficient to solve, but accurate

Simplified version

$$A \, x = b$$

Same values

from cell to cell
→ optimized FFT solver usable

from timestep to timestep
→ only one matrix build needed

Highly efficient to solve, but not fully accurate

From complete to simplified equation

$$\nabla \cdot \left[\frac{1}{\rho} \nabla \tilde{p} \right] = \hat{R}$$

How to get rid of the density term between the derivatives?

From complete to simplified equation

$$\nabla \cdot \left[\boxed{\frac{1}{\rho} \nabla \tilde{p}}\right] = \hat{R}$$

Pressure decomposition

$$\nabla \left(\frac{\tilde{p}}{\rho}\right) - \tilde{p} \nabla \left(\frac{1}{\rho}\right)$$

From complete to simplified equation

$$\nabla \cdot \left[\frac{1}{\rho} \nabla \tilde{p} \right] = \hat{R}$$

Pressure decomposition

Just algebraic rearranging,
but still the same equation!

$$\nabla \left(\frac{\tilde{p}}{\rho} \right) - \tilde{p} \, \nabla \left(\frac{1}{\rho} \right)$$

stays on the left

moved to the right

State after decomposition & resorting

$$\underbrace{\nabla^2 \left[\frac{\tilde{p}}{\rho} + \frac{|\mathbf{u}|^2}{2} \right]}_{H} = \underbrace{\dots \quad + \nabla \cdot \left[\overbrace{\tilde{p}\, \nabla \left(\frac{1}{\rho} \right)}^{\text{Baroclinic torque}} \right]}_{R}$$

Bernoulli Pressure

Right hand side of simplified equation

How to discretize this equation?

$$\nabla^2 \left[\frac{\tilde{p}}{\rho} + \frac{|\mathbf{u}|^2}{2} \right] = \quad \ldots \quad + \nabla \cdot \left[\tilde{p} \, \nabla \left(\frac{1}{\rho} \right) \right]$$

How to discretize this equation?

$$\nabla^2 \left[\frac{\tilde{p}}{\rho} + \frac{|\mathbf{u}|^2}{2} \right] = \quad \ldots \quad + \nabla \cdot \left[\tilde{p} \, \nabla \left(\frac{1}{\rho} \right) \right]$$

$\tilde{p}$ is the pressure term for which the system is currently being solved

$\tilde{p}$ isn‘t known when right hand side must be built !

Only delayed value on right side possible

$$\nabla^2 \left[\frac{\tilde{p}^n}{\rho} + \frac{|\mathbf{u}|^2}{2} \right] = \quad \ldots \quad + \nabla \cdot \left[\tilde{p}^{n-1} \nabla \left(\frac{1}{\rho} \right) \right]$$

$\tilde{p}$ is the pressure term for which the system is currently being solved

Value from last time step is used instead

Only delayed value on right side possible

$$\nabla^2 \left[\frac{\tilde{p}^n}{\rho} + \frac{|\mathbf{u}|^2}{2} \right] = \quad \dots \quad + \nabla \cdot \left[\tilde{p}^{n-1} \nabla \left(\frac{1}{\rho} \right) \right]$$

How big is the difference between the old and new pressure field?

Effects of Poisson equation

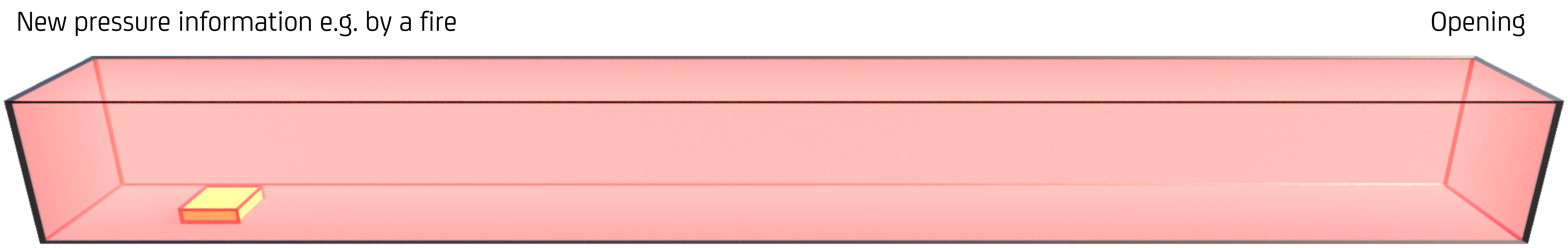
In reality, new pressure pulses would need seconds to pass the tunnel (~ speed of sound)

New pressure information e.g. by a fire

Opening

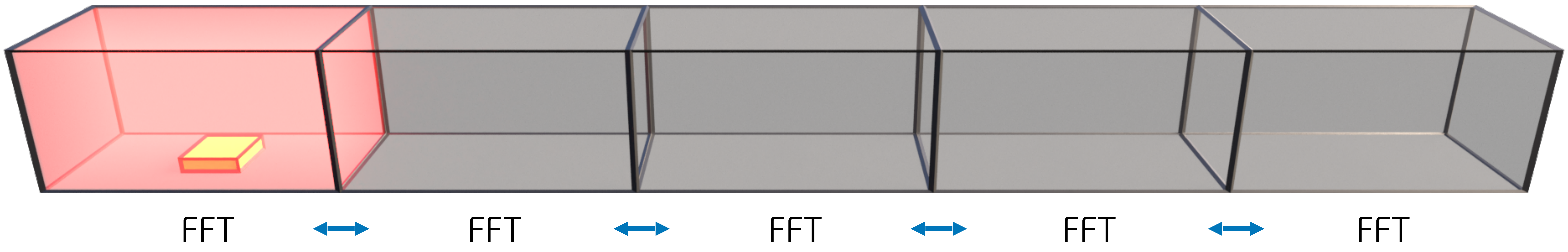
Effects of Poisson equation

In the Low-Mach formulation, new pressure pulses propagate infinitely fast



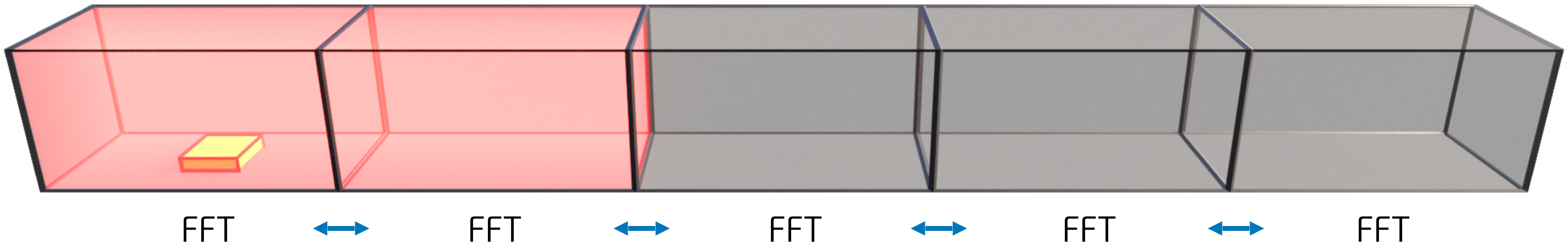
Default pressure solver - local FFT's

Each mesh performs its own local FFT method - highly efficient per mesh



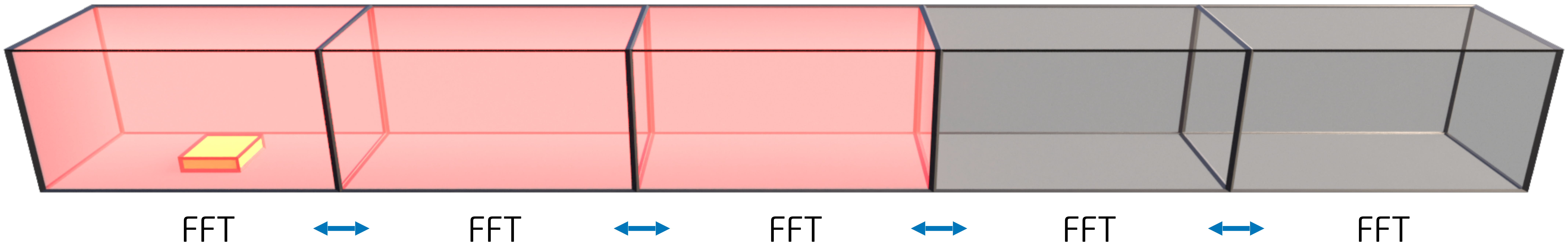
Default pressure solver - local FFT's

New pressure information can only be passed mesh-by-mesh via local data exchanges



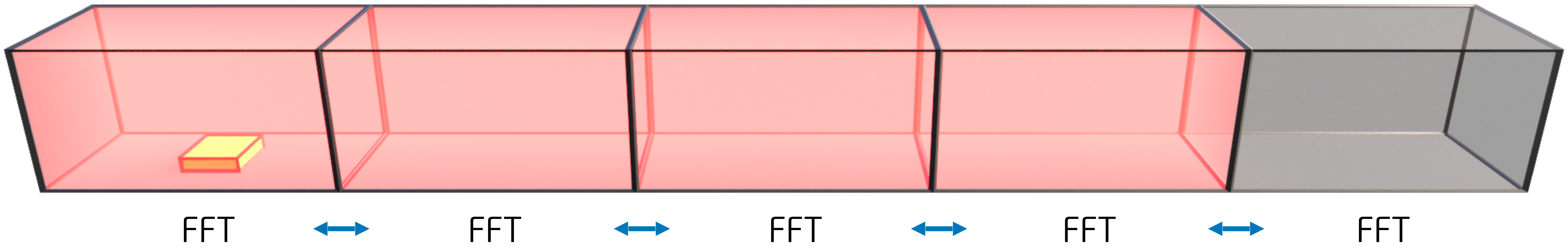
Default pressure solver - local FFT’s

New pressure information can only be passed mesh-by-mesh via local data exchanges



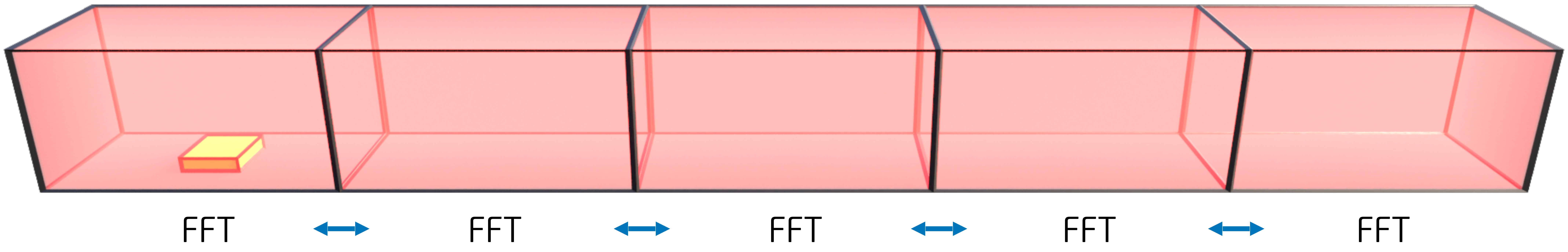
Default pressure solver - local FFT's

New pressure information can only be passed mesh-by-mesh via local data exchanges



Default pressure solver - local FFT's

New pressure information can only be passed mesh-by-mesh via local data exchanges



⟶ Does not conform to high propagation velocity of Poisson equation

FFT solver embedded in pressure iteration

```
&PRES VELOCITY_TOLERANCE = …, PRESSURE_TOLERANCE = …, MAX_PRESSURE_ITERATIONS = …
```

Tolerance for
mismatch of velocity normals
at mesh interfaces and solid obstructions

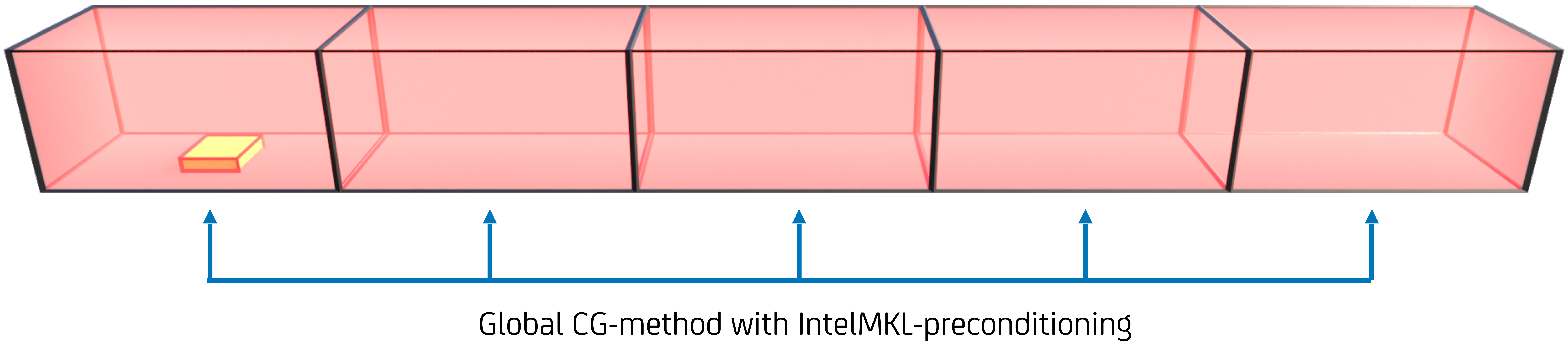
Tolerance for
difference between
old and new pressure field

Maximum number
of allowed
pressure iterations

⟶ **Velocity and pressure fields are forced to converge up to specified tolerances**

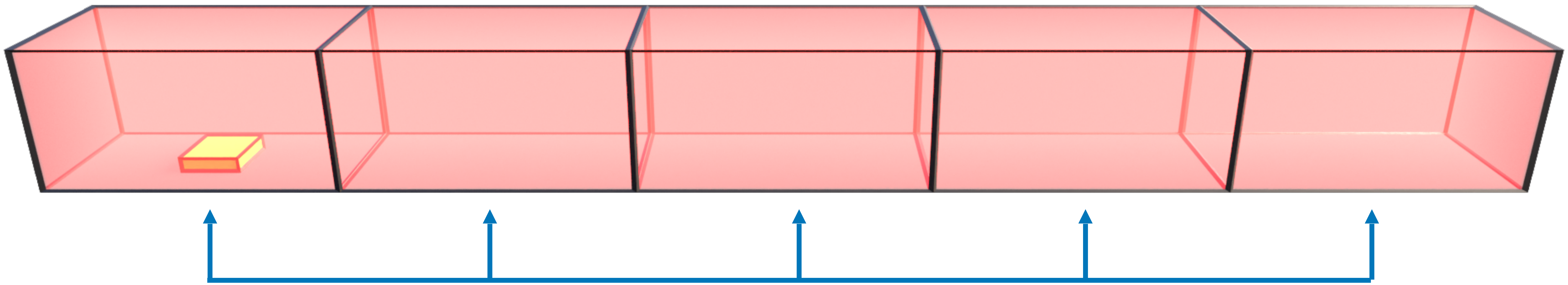
Additional feature of USCARC

Global solution of COMPLETE pressure equation on unstructured grids



Additional feature of USCARC

Global solution of COMPLETE pressure equation on unstructured grids

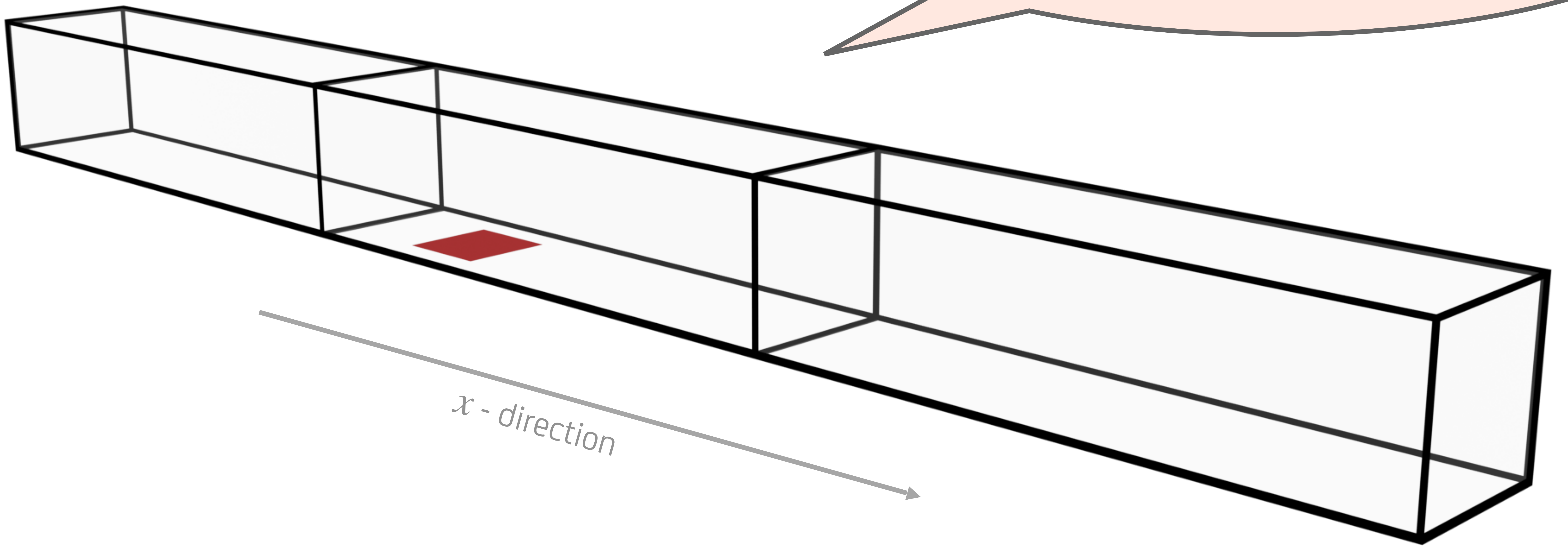


Global CG-method with IntelMKL-preconditioning

⟶ No pressure iteration needed, but higher effort per single solution

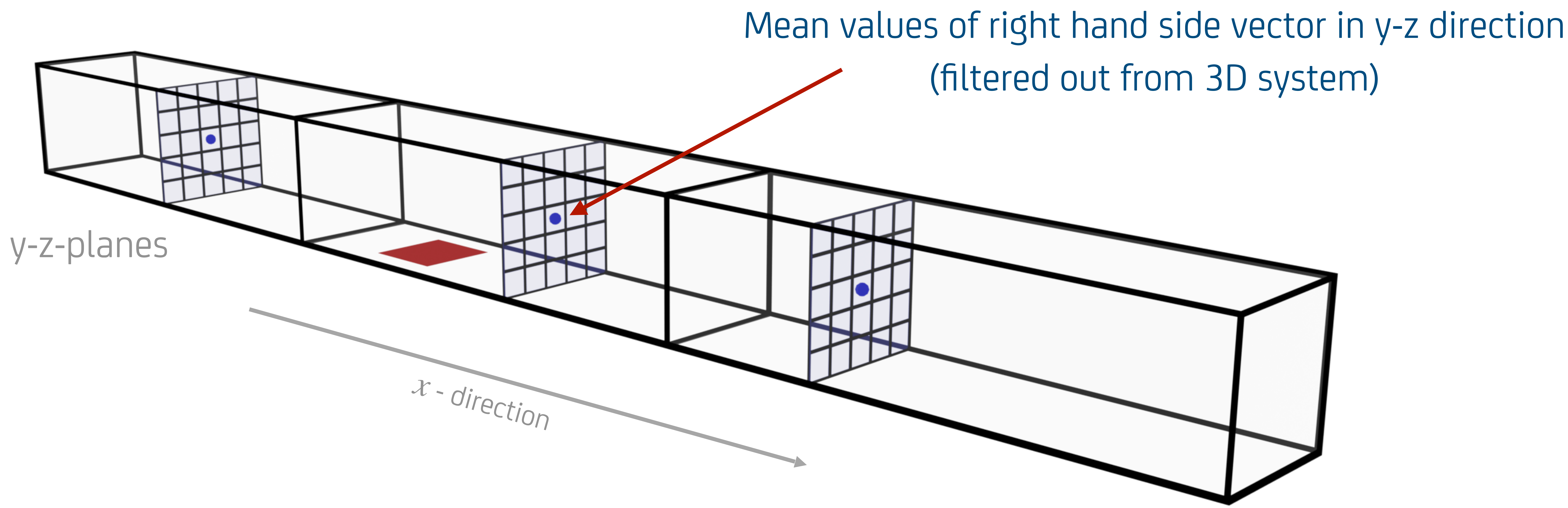
Special tunnel preconditioner

How can global information be transmitted at least roughly through the tunnel?



Special tunnel preconditioner

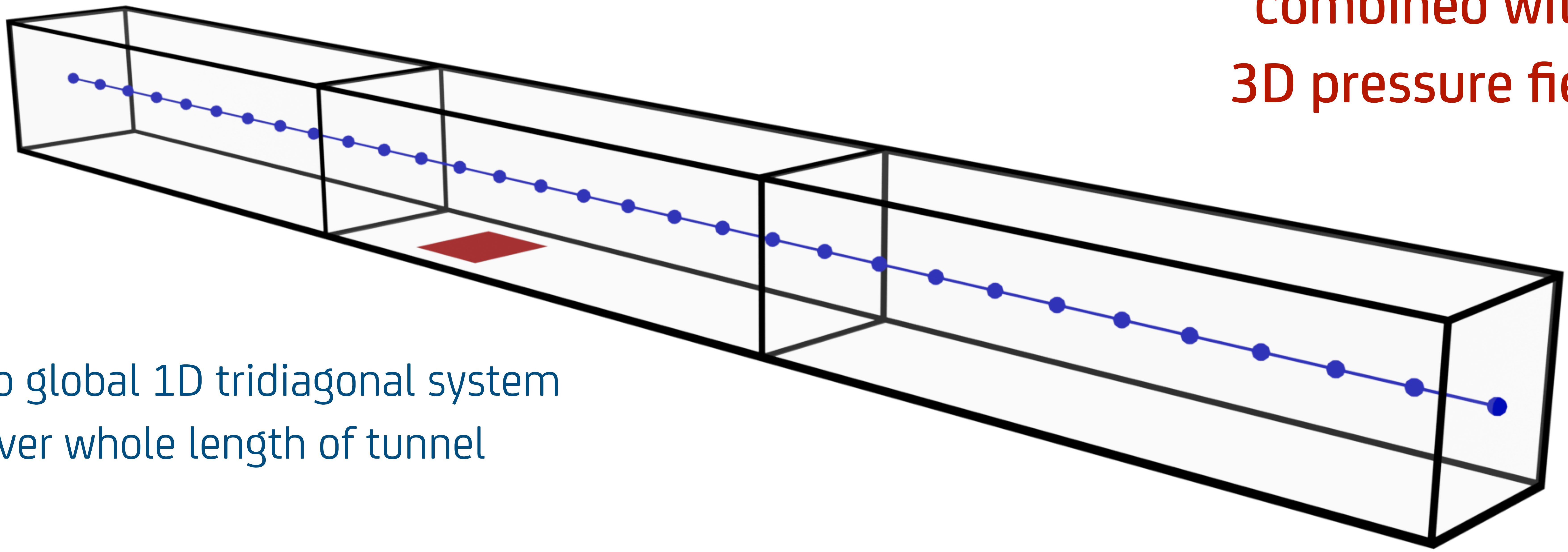
```
&PRES TUNNEL_PRECONDITIONER = .TRUE. /
```



Special tunnel preconditioner

```
&PRES TUNNEL_PRECONDITIONER = .TRUE. /
```

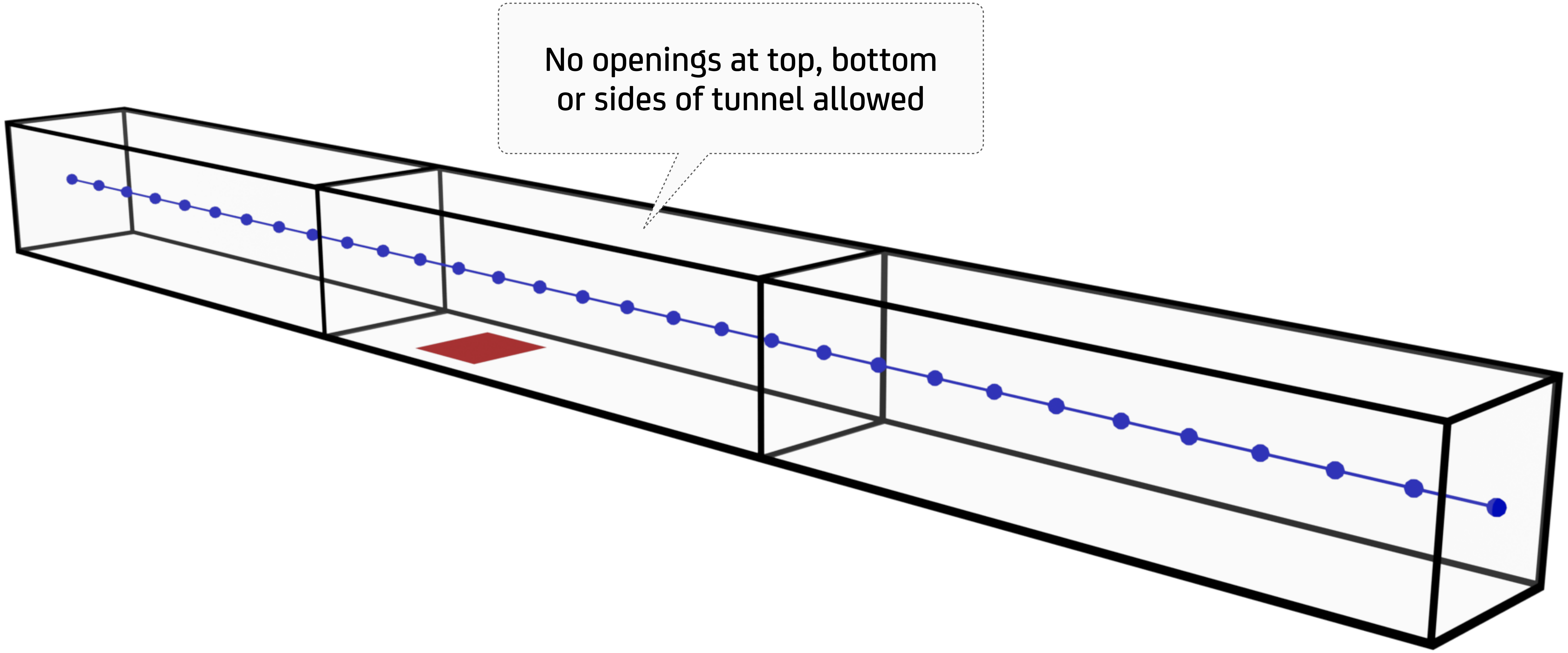
Average 1D pressure field
combined with
3D pressure field



Cheap global 1D tridiagonal system
over whole length of tunnel

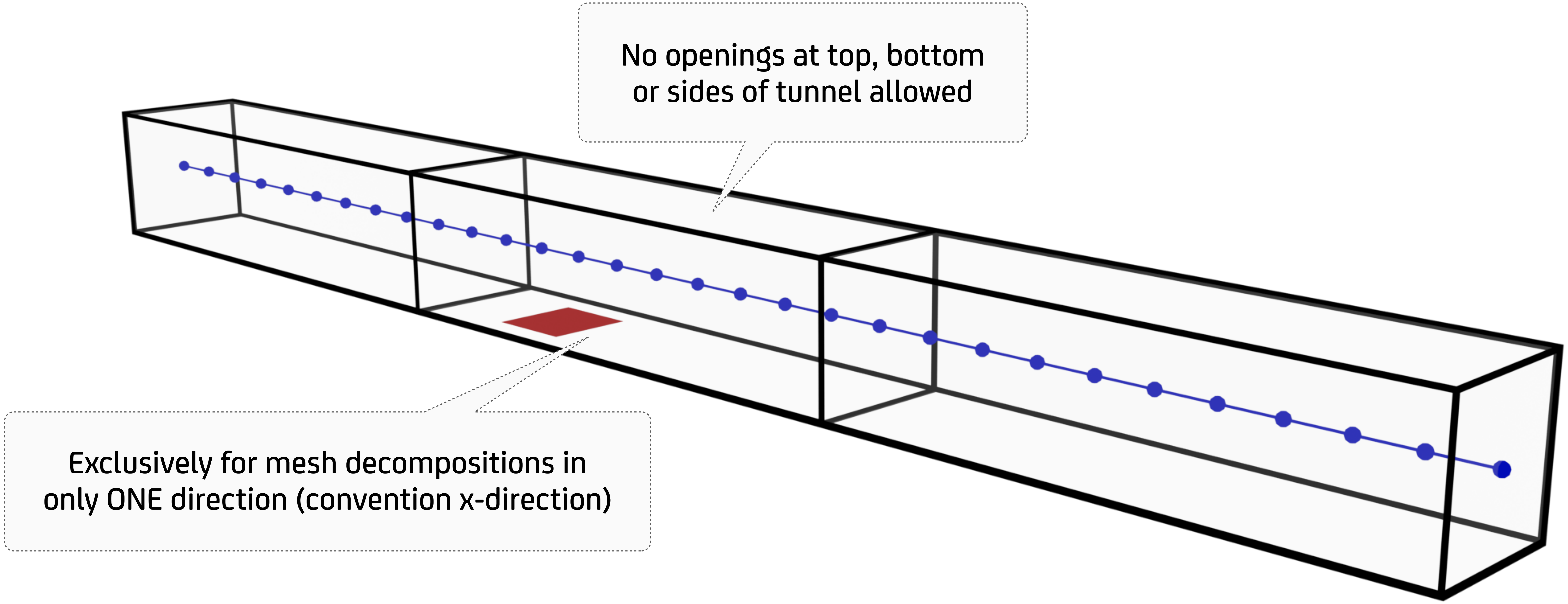
Special tunnel preconditioner

Restricted in its use, but very helpful if usable!



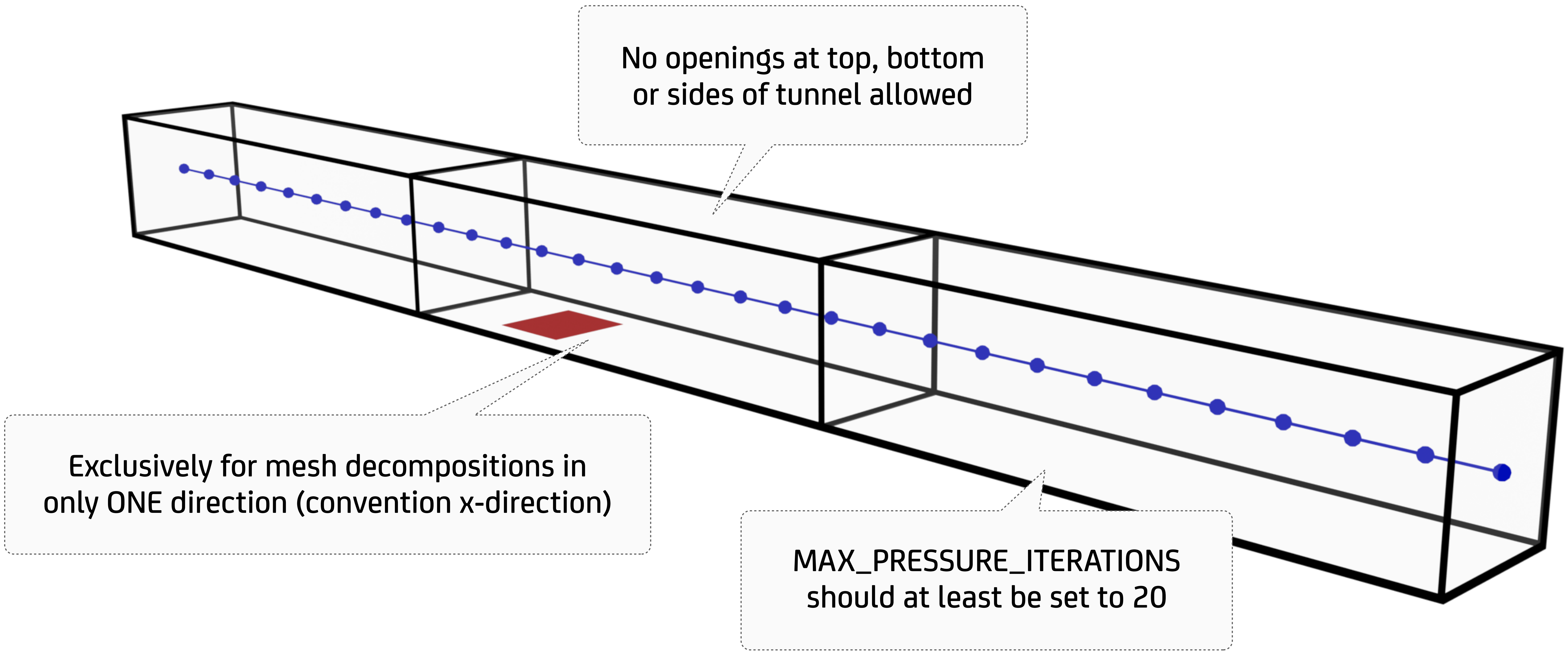
Special tunnel preconditioner

Restricted in its use, but very helpful if usable!



Special tunnel preconditioner

Restricted in its use, but very helpful if usable!

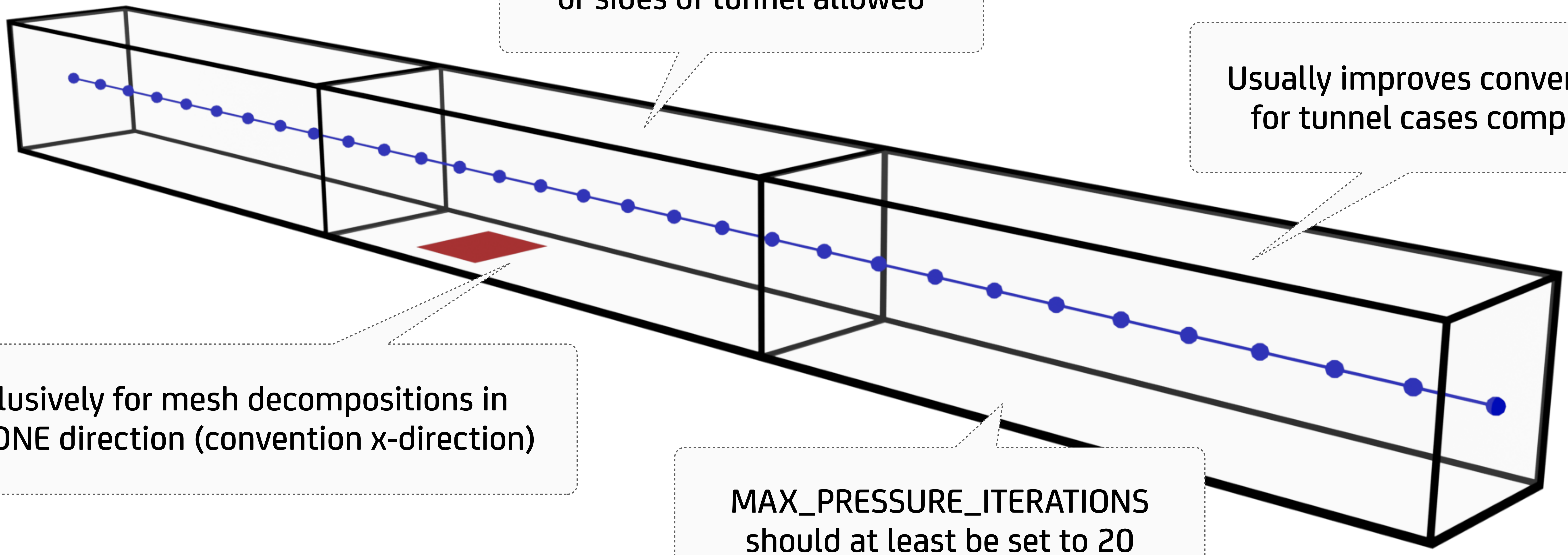


Special tunnel preconditioner

Restricted in its use, but very helpful if usable!

No openings at top, bottom or sides of tunnel allowed

Usually improves convergence speed for tunnel cases comprehensively



Exclusively for mesh decompositions in only ONE direction (convention x-direction)

MAX_PRESSURE_ITERATIONS should at least be set to 20

Thank you very much

for your attention