# Numerical considerations when configuring Fluidity

Stephan Kramer

Applied Modelling and Computation Group
Department of Earth Science and Engineering
Imperial College London

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# Time integration

Consider the advection diffusion equation:

$$\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T - \kappa \nabla^2 T = 0.$$

After finite element discretisation, we obtain a semi-discrete equation:

$$\mathrm{M}\frac{\partial \underline{\mathbf{T}}}{\partial t} + \mathrm{A}(\vec{u})\underline{\mathbf{T}} + \mathrm{D}\underline{\mathbf{T}} = 0.$$

which is an ODE in matrix form for the solution vector $\underline{\mathbf{T}}$ with time-dependent coefficients. The finite element method tells us how to assemble the mass matrix $\mathrm{M}$, the advection matrix $\mathrm{A}(\vec{u})$ and diffusion matrix $\mathrm{D}$.

# Explicit Euler

Time integration with the explicit **forward Euler** time stepping method:

$$\mathrm{M}\frac{\underline{\mathbf{T}}^{n+1}-\underline{\mathbf{T}}^{n}}{\Delta t}+\mathrm{A}(\vec{u})\underline{\mathbf{T}}^{n}+\mathrm{D}\underline{\mathbf{T}}^{n}=0$$

or

$$\mathrm{M}\underline{\mathbf{T}}^{n+1}=\mathrm{M}\underline{\mathbf{T}}^{n}-\Delta t\left(\mathrm{A}(\vec{u})\underline{\mathbf{T}}^{n}+\mathrm{D}\underline{\mathbf{T}}^{n}\right)$$

Even for an explicit method we still end up with a matrix equation to solve! For Continuous Galerkin FEM the inverse mass matrix is dense. In some cases we approximate the mass matrix by a diagonal matrix in a process called **mass lumping**.

# Explicit Euler

Time integration with the explicit **forward Euler** time stepping method:

$$\mathrm{M}\frac{\underline{\mathbf{T}}^{n+1}-\underline{\mathbf{T}}^{n}}{\Delta t}+\mathrm{A}(\vec{u})\underline{\mathbf{T}}^{n}+\mathrm{D}\underline{\mathbf{T}}^{n}=0$$

or

$$\mathrm{M}\underline{\mathbf{T}}^{n+1}=\mathrm{M}\underline{\mathbf{T}}^{n}-\Delta t\left(\mathrm{A}(\vec{u})\underline{\mathbf{T}}^{n}+\mathrm{D}\underline{\mathbf{T}}^{n}\right)$$

## Courant condition

The explicit method is only **conditionally** stable. For advection, a Courant condition applies similar to that for finite differences:

$$\frac{u\Delta t}{\Delta x}<1$$

# Courant condition & unstructured

Courant condition depends on numerical scheme.

## Example

Consider first–order–upwind finite volume for advection of a concentration $T$. In cell 2:

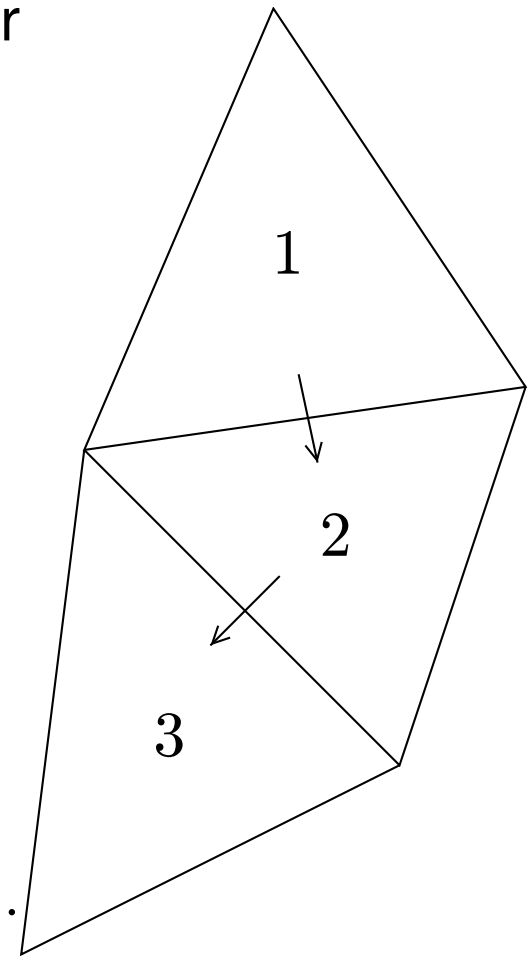
$$A_2\, T_2^{n+1} = A_2\, T_2^n + \Delta t\ \vec{u}_{12}\cdot\vec{n}_{12}\ w_{12}\ T_1^n - \Delta t\ \vec{u}_{23}\cdot\vec{n}_{23}\ w_{23}\ T_2^n.$$

For incompressible flow however

$$\vec{u}_{12}\cdot\vec{n}_{12}\ w_{12} = \vec{u}_{23}\cdot\vec{n}_{23}\ w_{23}.$$

This can be rewritten as an interpolation

$$T_2^{n+1} = (1-C)\, T_2^n + C\ T_1^n,\ C = \frac{\vec{u}_{12}\cdot\vec{n}_{12}\ w_{12}}{A_2}.$$



# Courant condition & unstructured

Courant condition depends on numerical scheme.

## Example

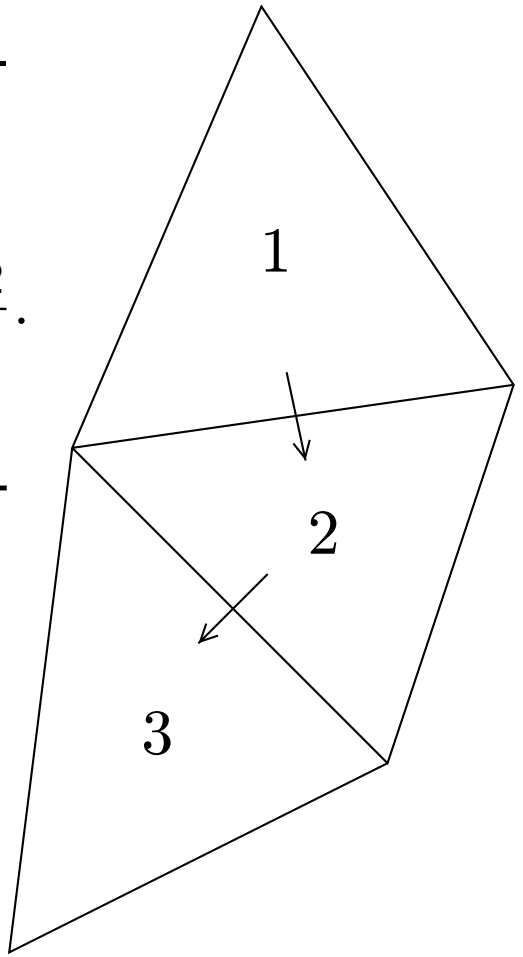
For first order upwind finite volume, the concentration in cell 2 is given by

$$T_2^{n+1} = (1 - C)\, T_2^n + C\, T_1^n, \quad C = \frac{\vec{u_{12}} \cdot \vec{n_{12}}\; w_{12}}{A_2}.$$

Rewriting $\vec{u} \cdot \vec{n} = \|u\| \hat{u} \cdot \vec{n}$, this leads to a CFL condition:

$$C = \frac{\Delta t\; \|u\| \hat{u} \cdot \vec{n}\; w}{A_2} < 1$$

In other words $\Delta x = \frac{A_2 \hat{u} \cdot \vec{n}}{w}$.



# Implicit Euler

The implicit **backward Euler** time-integration method gives:

$$\mathrm{M}\frac{\underline{\mathbf{T}}^{n+1}-\underline{\mathbf{T}}^{n}}{\Delta t}+\mathrm{A}(\vec{u})\underline{\mathbf{T}}^{n+1}+\mathrm{D}\underline{\mathbf{T}}^{n+1}=0$$

or

$$[\mathrm{M}+\Delta t\,(\mathrm{A}(\vec{u})+\mathrm{D})]\,\underline{\mathbf{T}}^{n+1}=\mathrm{M}\underline{\mathbf{T}}^{n}$$

This method is **unconditionally** stable, i.e. no time step restriction.

However it may introduce excessive damping!

# Theta method

The theta method is given by:

$$\mathrm{M}\frac{\underline{\mathbf{T}}^{n+1}-\underline{\mathbf{T}}^{n}}{\Delta t}+\mathrm{A}(\vec{u})\underline{\mathbf{T}}^{n+\theta}+\mathrm{D}\underline{\mathbf{T}}^{n+\theta}=0$$

where

$$T^{n+\theta}=(1-\theta)T^{n}+\theta T^{n+1}.$$

| | | | |
|---|---|---|---|
| | $\theta=0$ | Explicit Euler | conditionally stable |
| For | $\theta=1$ | Implicit Euler | uncond. stable, dissipative |
| | $\theta=0.5$ | Crank-Nicolson | second order! |

Unconditionally stable for $\theta >= 0.5$. In practice, a value just above 0.5 is often chosen.

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# Conservative vs non-conservative

The scalar advection diffusion equation can be solved in two forms:

Non-conservative form: $$\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T - \kappa \nabla^2 T = 0$$

Conservation form: $$\frac{\partial T}{\partial t} + \nabla \cdot (\vec{u} T) - \kappa \nabla^2 T = 0.$$

For smooth fields $T$ in incompressible flow the two should be the same. However, numerically $\nabla \cdot \vec{u}$ is only zero in approximation (depends on pressure discretisation). We provide a choice via a parameter $\beta$:

$$\frac{\partial T}{\partial t} + (1-\beta)\vec{u} \cdot \nabla T + \beta \nabla \cdot (\vec{u} T) - \kappa \nabla^2 T = 0.$$

**$\beta = 0$, non-conservative**

Depending on scheme, may be bounded.

**$\beta = 1$, conservative**

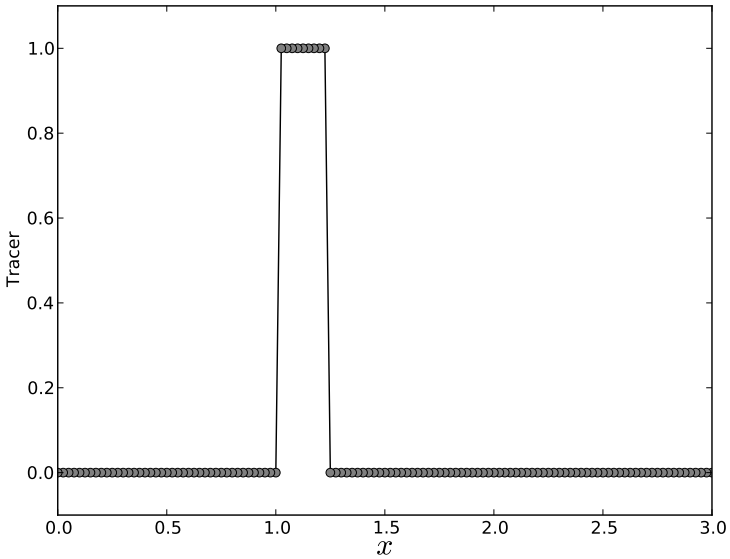
Depending on scheme, typically **not** bounded.

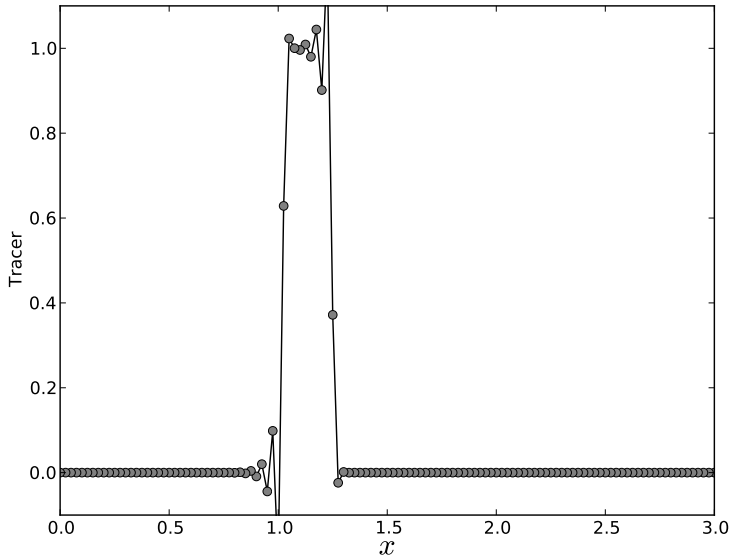
# Discretisation for advection-diffusion

Three methods:

- **Continuous Galerkin** Finite Element Method
- **Control Volumes** / Finite Volume Method
- **Discontinuous Galerkin** Finite Element Method

# Continuous Galerkin




Unstabilised Continuous Galerkin is not very good at handling discontinuities and large gradients. It is accurate and fairly efficient for smooth fields.

## Stabilisation methods

- SU (recommended)
- SUPG

**More details see:**
Donea and Huerta [2003]

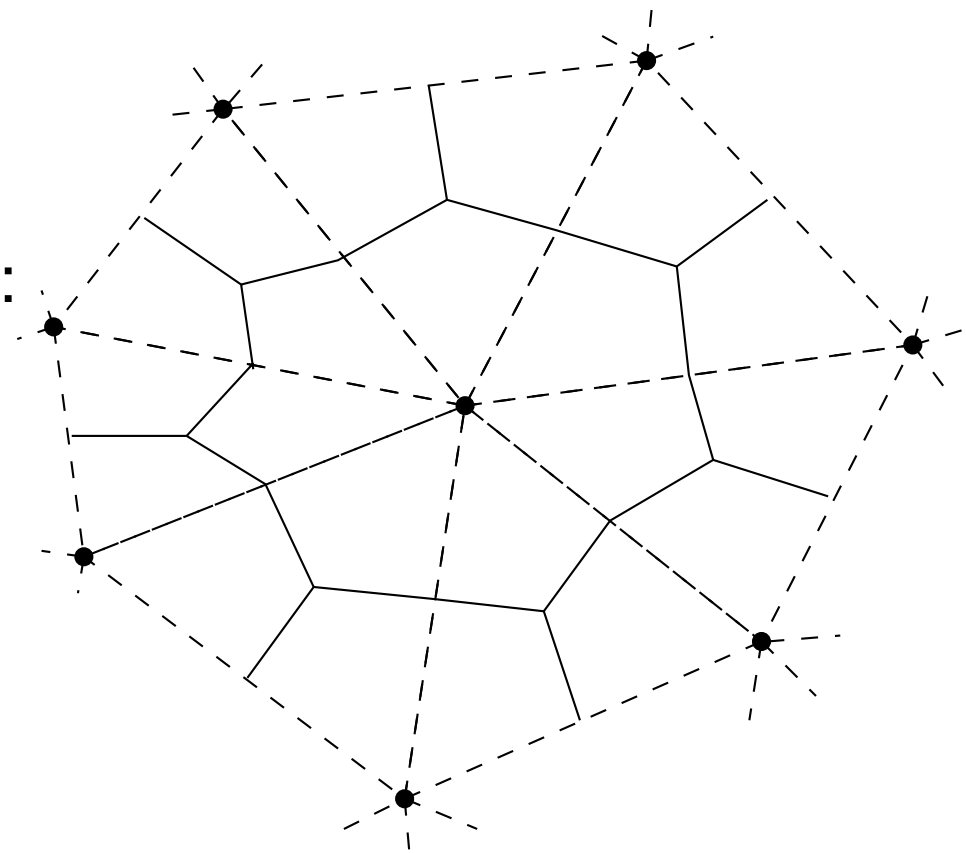
# Control volumes

Vertex based control volumes

Face values computed as:

- First order upwind
- Finite element (linear interpolation)

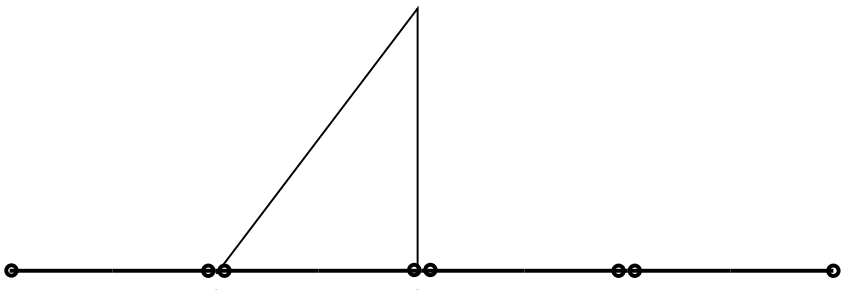
In combination with slope limiter: Sweby

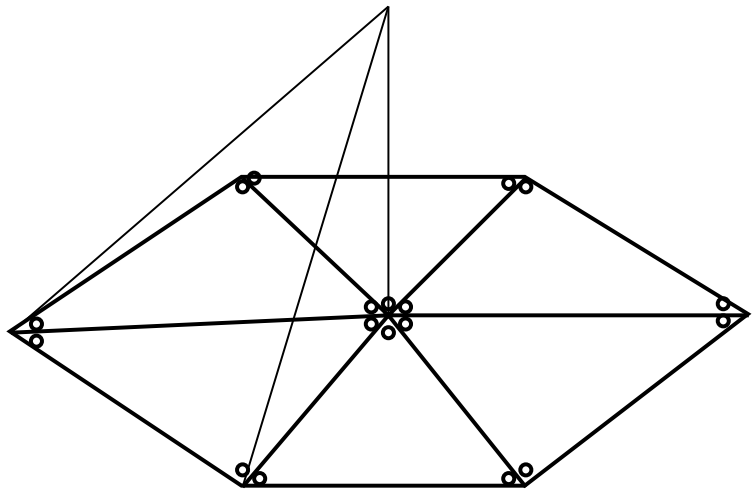


**References:**

LeVeque [2002] (general finite volume and slope limiters)
Wilson [2009] (Fluidity specific details, multi-material)

# Discontinuous Galerkin

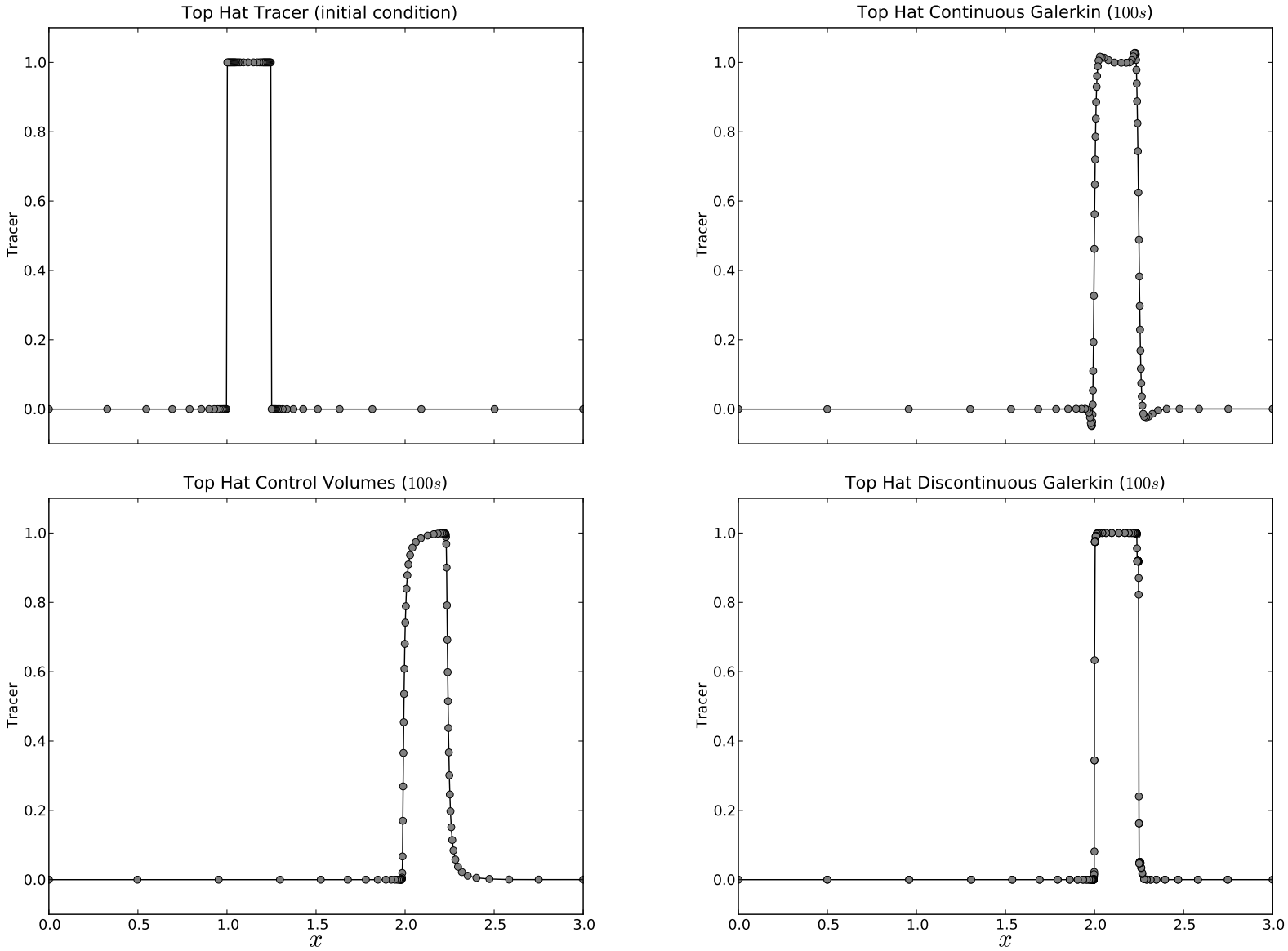




Discontinuous Galerkin methods are much better at handling discontinuities. Requires:

- subcycling: advection is divided in small sub-timesteps with CFL number smaller than one.
- slope limiting: changes the slope within each element to prevent overshoots.

**Reference:** Cockburn and Shu [2001]




Top hat example from the Fluidity manual, run with CG+SUPG, and CV+Sweby, and DG+slope limiting.

# Diffusion/Viscosity

## Remark:

Numerical considerations for the scalar advection diffusion equation are similar to those for the advection and viscosity terms in the momentum equation:

$$\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T - \kappa \nabla^2 T = 0.$$

$$\rho \left( \frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} - \nu \nabla^2 \vec{u} \right) + \nabla p = -g \rho \hat{z}$$

# Diffusion/Viscosity

Péclet (Reynolds) number, ratio of the rate of advection over the rate of diffusion (viscosity):

$$\mathrm{Pe} = \frac{LU}{\kappa}$$

$$\mathrm{Re} = \frac{LU}{\nu}$$

Typically, the length scale at which diffusion (viscosity) is dominant is much smaller than the grid scale:

$$\mathrm{Pe}_{\mathrm{grid}} = \frac{\Delta x U}{\kappa} \gg 1$$

$$\mathrm{Re}_{\mathrm{grid}} = \frac{\Delta x U}{\nu} \gg 1$$

## Sub-grid/turbulence models

- RANS models: k-$\varepsilon$ model
- Large Eddy simulation (LES)
- Vertical parameterisations for oceans: GLS model.

**References:** LES and k-$\varepsilon$: Pope [2000], Wilcox [1998], GLS: Hill et al. [2012], Umlauf and Burchard [2005]

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# Incompressible NS equations

Incompressible Navier Stokes equations:

$$\rho \frac{\partial \vec{u}}{\partial t} + \rho \vec{u} \cdot \nabla \vec{u} - \mu \nabla^2 \vec{u} + \nabla p = -g \rho \hat{z}$$

$$\nabla \cdot \vec{u} = 0$$

with density $\rho$, velocity $\vec{u}$, dynamic viscosity $\mu$, pressure $p$, gravitational acceleration $g$, and $\hat{z}$ the upwards unit vector. Density from equation of state:

$$\rho = \rho_0 \left(1 - \alpha T + \gamma S\right),$$

where $\rho_0$ is the reference density, $T$ and $S$ are temperature and salinity, and $\alpha$ and $\gamma$ are the thermal expansion and haline contraction coefficients.

# Incompressible NS equations

Incompressible Navier Stokes equations in the **Boussinesq** approximation: ($|\rho - \rho_0| \ll \rho_0$)

$$\rho_0 \frac{\partial \vec{u}}{\partial t} + \rho_0 \vec{u} \cdot \nabla \vec{u} - \mu \nabla^2 \vec{u} + \nabla p = -g \rho \hat{z}$$

$$\nabla \cdot \vec{u} = 0$$

with density $\rho$, velocity $\vec{u}$, dynamic viscosity $\mu$, pressure $p$, gravitational acceleration $g$, and $\hat{z}$ the upwards unit vector. Density from equation of state:

$$\rho = \rho_0 \left(1 - \alpha T + \gamma S\right),$$

where $\rho_0$ is the reference density, $T$ and $S$ are temperature and salinity, and $\alpha$ and $\gamma$ are the thermal expansion and haline contraction coefficients.

# Incompressible NS equations

Incompressible Navier Stokes equations in the **Boussinesq** approximation: ($|\rho - \rho_0| \ll \rho_0$)

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} - \nu \nabla^2 \vec{u} + \nabla P = -g \frac{\rho}{\rho_0} \hat{z}$$

$$\nabla \cdot \vec{u} = 0$$

with density $\rho$, velocity $\vec{u}$, kinematic viscosity $\nu = \mu/\rho_0$, kinematic pressure $P = p/\rho_0$, gravitational acceleration $g$, and $\hat{z}$ the upwards unit vector. Density from equation of state:

$$\rho = \rho_0 (1 - \alpha T + \gamma S),$$

where $\rho_0$ is the reference density, $T$ and $S$ are temperature and salinity, and $\alpha$ and $\gamma$ are the thermal expansion and haline contraction coefficients.

# Hydrostatic pressure

Large part of pressure is hydrostatic, i.e.

$$\frac{\partial p_h}{\partial z} = -\rho g, \quad p_h = 0 \text{ at } z = 0 \implies p_h = \int_{z'=z}^{z'=0} \rho g \, dz'$$

In other words, $p_h$ is proportional to the weight of the water column above each point in the fluid. If we only look at the weight of the reference fluid, then

$$p_{h,0} = -\rho_0 g z \text{ and } \nabla p_{h,0} = -\rho_0 g \hat{z}$$

# Hydrostatic pressure

Large part of pressure is hydrostatic, i.e.

$$\frac{\partial p_h}{\partial z} = -\rho g, \quad p_h = 0 \text{ at } z = 0 \implies p_h = \int_{z'=z}^{z'=0} \rho g \, dz'$$

In other words, $p_h$ is proportional to the weight of the water column above each point in the fluid. If we only look at the weight of the reference fluid, then

$$p_{h,0} = -\rho_0 g z \text{ and } \nabla p_{h,0} = -\rho_0 g \hat{z}$$

# Subtract hydrostatic part

Pressure is split according to

$$P = P_{h,0} + P'$$

where the reference hydrostatic pressure is given by:

$$P_{h,0} = -gz \text{ so that } \nabla P_{h,0} = -g\hat{z}$$

Substitution in the momentum equation:

$$\frac{\partial \vec{u}}{\partial t} + \vec{u}\cdot\nabla\vec{u} - \nu\nabla^2\vec{u} + \nabla P_{h,0} + \nabla P' = -\frac{g}{\rho_0}(\rho_0 + \Delta\rho)\hat{z}$$

# Subtract hydrostatic part

Pressure is split according to

$$P = P_{h,0} + P'$$

where the reference hydrostatic pressure is given by:

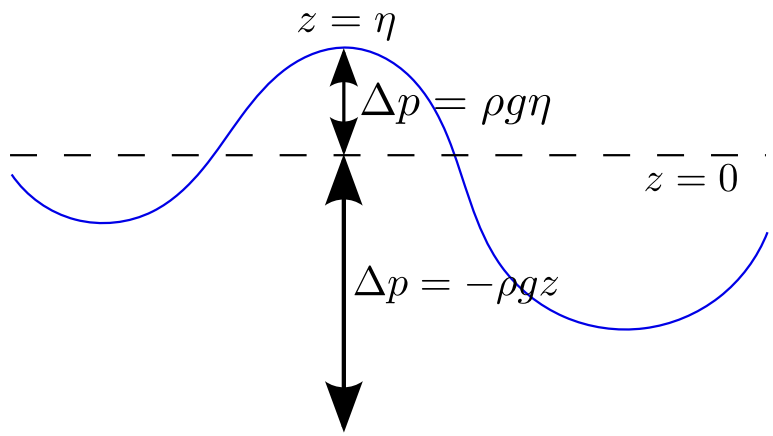
$$P_{h,0} = -gz \text{ so that } \nabla P_{h,0} = -g\hat{z}$$

Substitution in the momentum equation:

$$\frac{\partial \vec{u}}{\partial t} + \vec{u}\cdot\nabla\vec{u} - \nu\nabla^2\vec{u} + \cancel{\nabla P_{h,0}} + \nabla P' = -\frac{g}{\rho_0}\left(\cancel{\rho_0} + \Delta\rho\right)\hat{z}$$

# Free surface and Pressure

If a free surface is included, with elevation $\eta$, the part of the hydrostatic pressure between $z = 0$ and $z = \eta$ is **not** subtracted.




In other words, the barotropic part of pressure $\rho_0 g \eta$ is included in the pressure solved for in Fluidity.

This means that P' is nonzero at the free-surface, because at $z = \eta$:

$$P' = P - P_{h,0} = 0 + gz|_{z=\eta} = g\eta$$

# Discretisation of NS

Boussinesq equations with hydrostatic subtracted:

$$\frac{\partial \vec{u}}{\partial t} + \vec{u}\cdot\nabla\vec{u} - \nu\nabla^2\vec{u} + \nabla P' = -g\frac{\Delta\rho}{\rho_0}\hat{z}$$

$$\nabla\cdot\vec{u} = 0$$

Discretised in time and space:

$$\mathrm{M}\frac{\underline{\mathbf{u}}^{n+1} - \underline{\mathbf{u}}^{n}}{\Delta t} + \mathrm{A}(\vec{u})\underline{\mathbf{u}}^{n+\theta} + \mathrm{K}\underline{\mathbf{u}}^{n+\theta} + \mathrm{C}\underline{\mathbf{P}}'^{n+\frac{1}{2}} = \underline{\mathbf{f}}(\Delta\rho)$$

$$-\mathrm{C}^T\underline{\mathbf{u}}^{n+1} = 0$$

with matrices:

| | |
|---|---|
| $\mathrm{M}$ | mass matrix |
| $\mathrm{A}(\underline{\mathbf{u}})$ | advection matrix |
| $\mathrm{K}$ | viscosity matrix |
| $\mathrm{C}$ | gradient matrix |
| $-\mathrm{C}^T$ | divergence matrix |

# Pressure correction approach

$$\frac{\partial \vec{u}}{\partial t} + \vec{u}\cdot\nabla\vec{u} - \nu\nabla^2\vec{u} + \nabla P' = -g\frac{\Delta\rho}{\rho_0}\hat{z}$$

$$\nabla\cdot\vec{u} = 0$$

Solved in three steps:

1. Solve a preliminary $\underline{\mathbf{u}}^*$ (not divergence free):

$$\mathrm{M}\frac{\underline{\mathbf{u}}^* - \underline{\mathbf{u}}^n}{\Delta t} + \mathrm{A}(\vec{u})\underline{\mathbf{u}}^{n+\theta} + \mathrm{K}\underline{\mathbf{u}}^{n+\theta} + \mathrm{C}\underline{\mathbf{P}}'^{n-\frac{1}{2}} = \underline{\mathbf{f}}$$

2. Solve pressure correction $\delta\underline{\mathbf{P}}'$:

$$\mathrm{C}^T\mathrm{M}^{-1}\mathrm{C}\ \delta\underline{\mathbf{P}}' = -\mathrm{C}^T\underline{\mathbf{u}}^*$$

3. Correct velocity, and update pressure:

$$\mathrm{M}\frac{\underline{\mathbf{u}}^{n+1} - \underline{\mathbf{u}}^*}{\Delta t} + \mathrm{C}\delta\underline{\mathbf{P}}' = 0, \quad \underline{\mathbf{P}}'^{n+\frac{1}{2}} = \underline{\mathbf{P}}'^{n-\frac{1}{2}} + \delta\underline{\mathbf{P}}'$$

**More details:**Gresho and Sani [1998]

# Choice of velocity, pressure pair

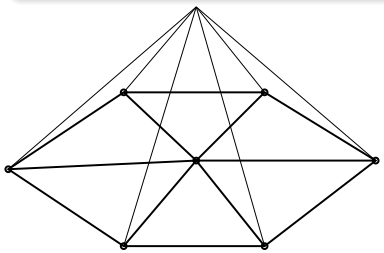
## P1-P1

Advantages:

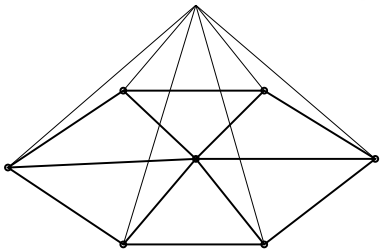
1. Simple

Disadvantages:

1. Unstable (LBB criterion), requires additional stabilisation
2. Mass lumping required



Velocity



Pressure

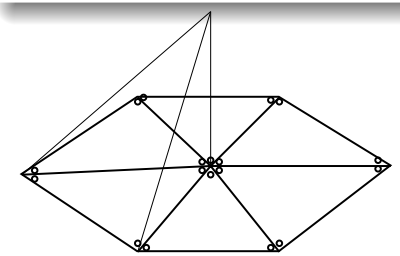
## $P1_{DG}$-P2

Advantages:

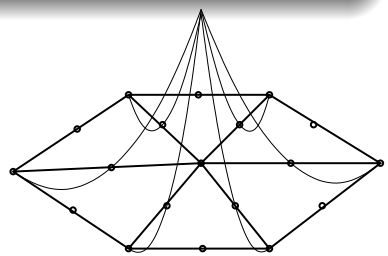
1. Higher order
2. Geostrophic balance
3. Stable
4. No mass lumping

Disadvantages:

1. More DOFs



Velocity



Pressure

**References:** Gresho and Sani [1998] (element pairs, LBB stability), Cotter et al. [2009] ($P1_{DG}$-P2), Pain et al. [2005] (P1-P1 stabilisation)

# Geostrophic balance

In larger scale ocean simulations, the flow is dominated by a **geostrophic balance** between the pressure gradient and Coriolis term in the horizontal, and **hydrostatic balance** in the vertical:

$$\frac{\partial \vec{u}}{\partial t} + \vec{u}\cdot\nabla\vec{u} - \nu\nabla^2\vec{u} + 2\vec{\Omega}\times\vec{u} + \nabla P' = -g\frac{\Delta\rho}{\rho_0}\hat{z},$$

here $\vec{\Omega}$ is the angular velocity vector of the Earth's rotation.

The gradient of a P1 pressure is P0 (piece-wise constant); Therefore, no exact balance between a P1 Coriolis term (or P1 buoyancy). With **P1-P1** we solve for a separate **geostrophic pressure** that is piecewise quadratic (P2, still no exact balance). With **$\mathbf{P1_{DG}}$-P2** an exact balance is possible without a separate geostrophic pressure solve.

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

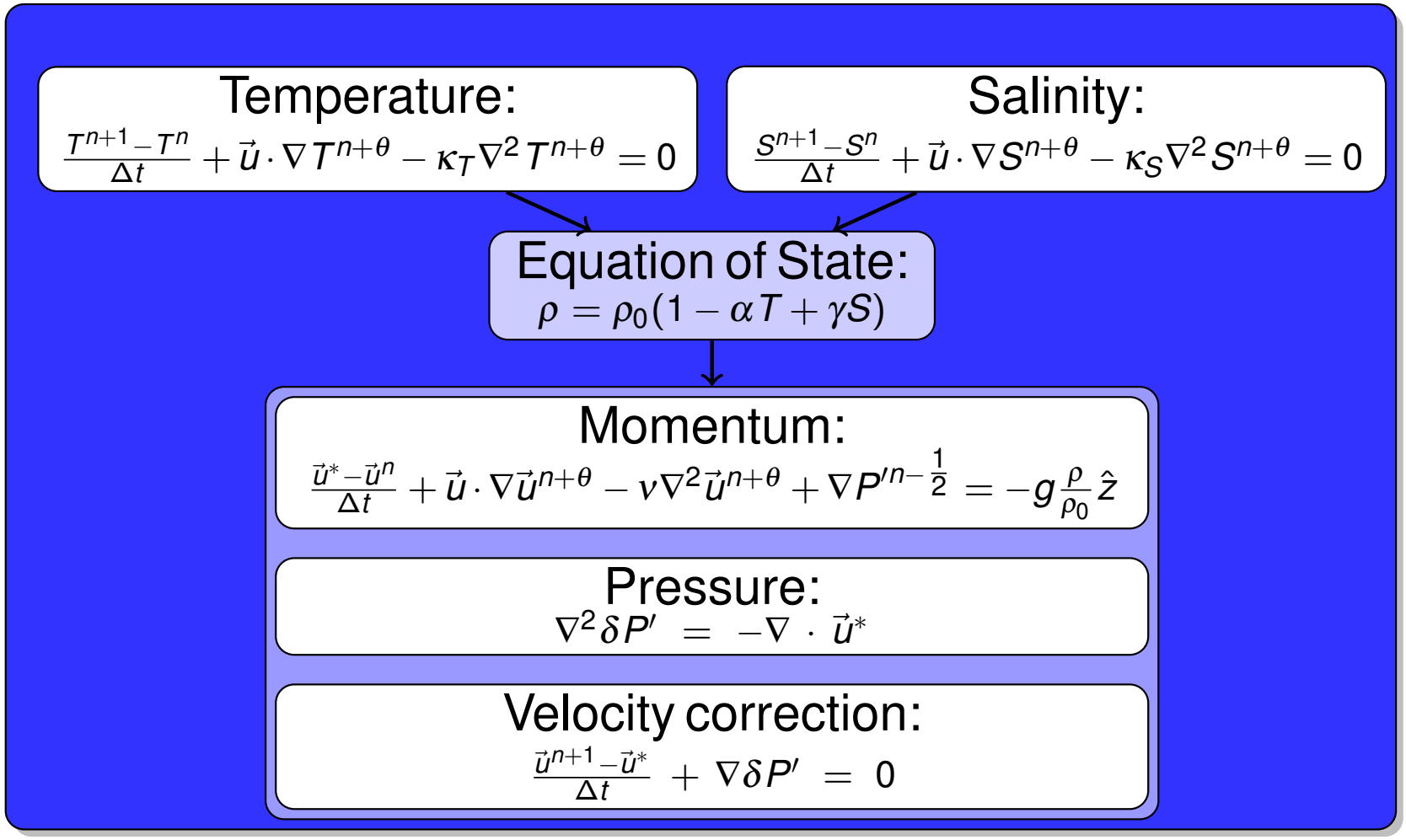
Boundary conditions

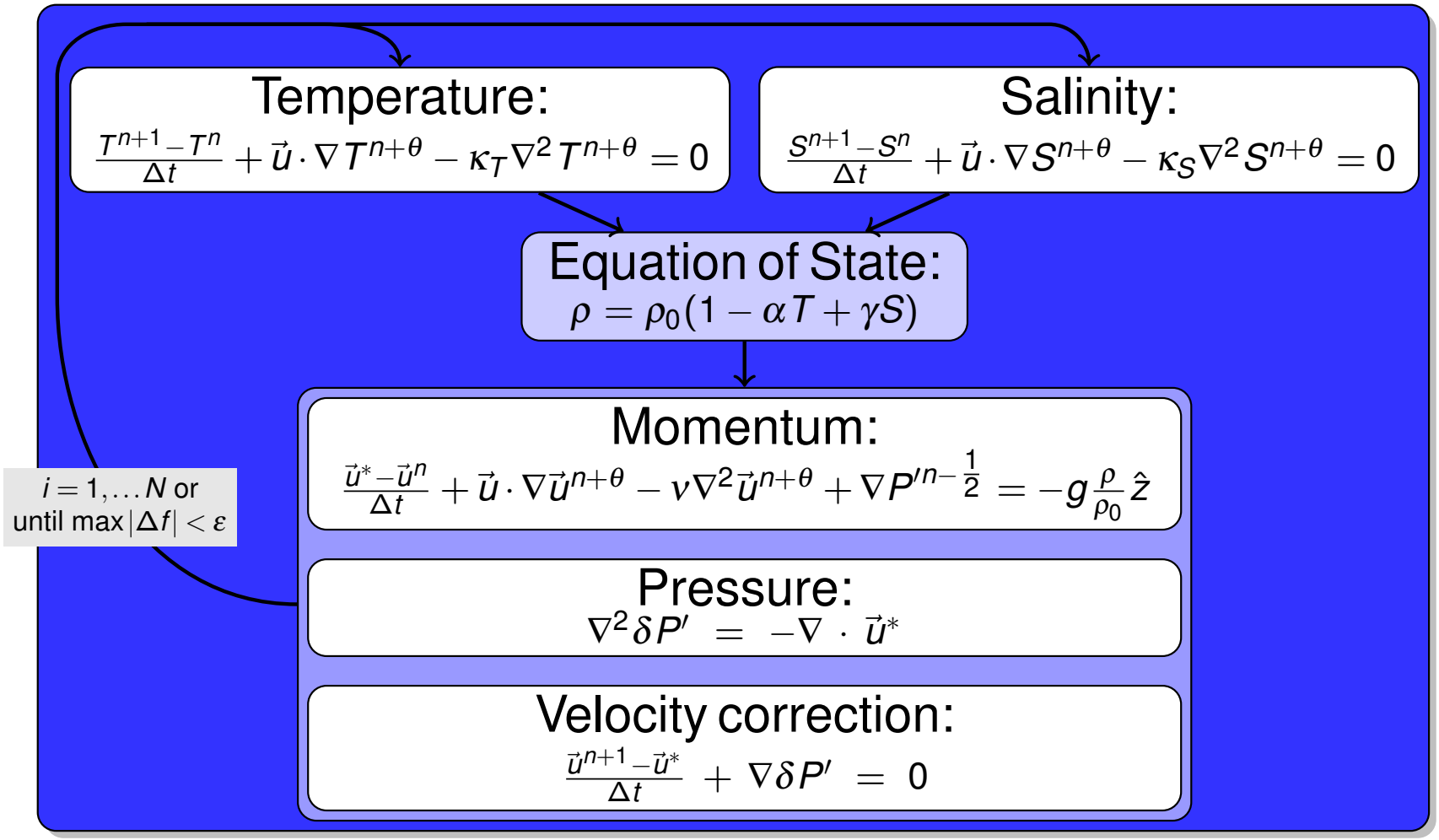
Linear solvers

References

Imperial College
London

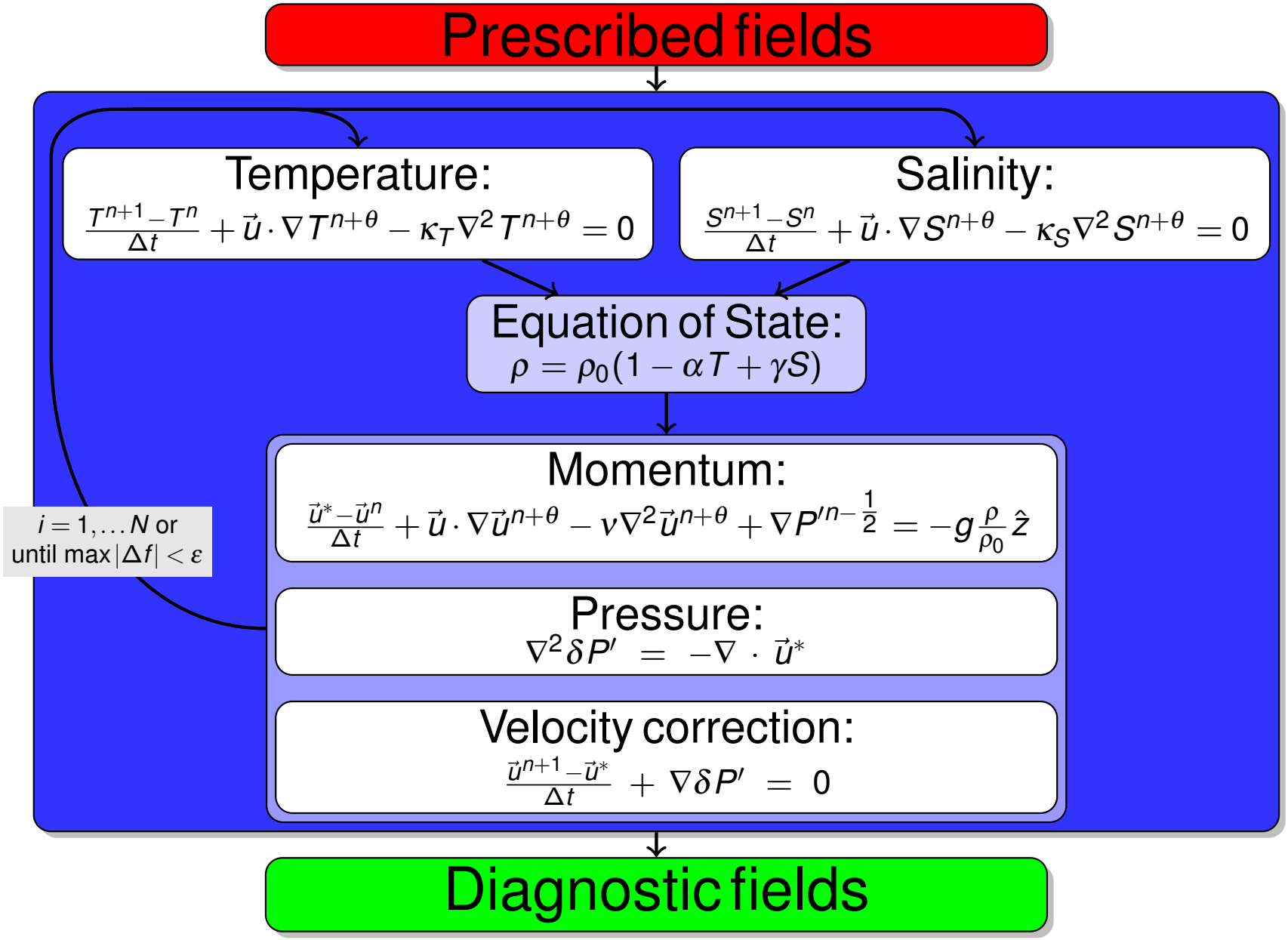
# Time loop






## Non-linear (Picard) iteration

Specified number of iterations $N$ (often $N = 2$) or until changes in prognostic field values are small enough. Advective velocity is updated according to a seperate $\theta$, the **relaxation** parameter.


Prescribed fields
Temperature:
$\frac{T^{n+1}-T^n}{\Delta t} + \vec{u}\cdot\nabla T^{n+\theta} - \kappa_T \nabla^2 T^{n+\theta} = 0$
Salinity:
$\frac{S^{n+1}-S^n}{\Delta t} + \vec{u}\cdot\nabla S^{n+\theta} - \kappa_S \nabla^2 S^{n+\theta} = 0$
Equation of State:
$\rho = \rho_0(1 - \alpha T + \gamma S)$
Momentum:
$\frac{\vec{u}^*-\vec{u}^n}{\Delta t} + \vec{u}\cdot\nabla\vec{u}^{n+\theta} - \nu\nabla^2\vec{u}^{n+\theta} + \nabla P'^{n-\frac{1}{2}} = -g\frac{\rho}{\rho_0}\hat{z}$
Pressure:
$\nabla^2\delta P' = -\nabla\cdot\vec{u}^*$
Velocity correction:
$\frac{\vec{u}^{n+1}-\vec{u}^*}{\Delta t} + \nabla\delta P' = 0$
$i = 1,\ldots N$ or
until $\max|\Delta f| < \varepsilon$
Diagnostic fields

# Baroclinic instability

Due to a lack of coupling between momentum and temperature equation, there is a numerical instability associated with the buoyancy frequency

$$N^2 = -\frac{g}{\rho_0}\frac{\partial \rho}{\partial z}.$$

This leads to a time step restriction of

$$N\Delta t < 1.$$

Fluidity provides an option, called **implicit_buoyancy**, to overcome this restriction.

Imperial College London

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# Strong vs. weak boundary conditions

- **Strong** boundary conditions are imposed exactly in each node of the mesh
- **Weak** boundary conditions are imposed in a weak integral sense that occur naturally in the finite element discretisation

## Example

Weak FEM discretisation of advection equation:

$$\int_\Omega \psi \frac{\partial T}{\partial t} + \int_\Omega \psi \vec{u} \cdot \nabla T = 0, \quad \text{for all } \psi \in V_{\text{test}}$$

Integrate by parts:

$$\int_\Omega \psi \frac{\partial T}{\partial t} - \int_\Omega \nabla \psi \cdot \vec{u} T + \int_\Gamma \psi \vec{n} \cdot \vec{u} T = 0, \quad \text{for all } \psi \in V_{\text{test}}$$

# Strong vs. weak boundary conditions

- **Strong** boundary conditions are imposed exactly in each node of the mesh
- **Weak** boundary conditions are imposed in a weak integral sense that occur naturally in the finite element discretisation

## Example

Weak FEM discretisation of advection equation:

$$\int_\Omega \psi \frac{\partial T}{\partial t} + \int_\Omega \psi \vec{u} \cdot \nabla T = 0, \quad \text{for all } \psi \in V_{\text{test}}$$

Integrate by parts, substitute boundary condition $T = T_D$:

$$\int_\Omega \psi \frac{\partial T}{\partial t} - \int_\Omega \nabla \psi \cdot \vec{u} T + \int_\Gamma \psi \vec{n} \cdot \vec{u} T_D = 0, \quad \text{for all } \psi \in V_{\text{test}}$$

# Strong vs. weak boundary conditions

- **Strong** boundary conditions are imposed exactly in each node of the mesh
- **Weak** boundary conditions are imposed in a weak integral sense that occur naturally in the finite element discretisation

Weak are generally better behaved. Although the boundary condition is not satisfied exactly, integrated quantities such as fluxes through the boundaries are exact. Neumann boundary conditions are always weakly imposed.
Strong boundary conditions should only be used in special cases.

# BCs for scalar advection-diffusion

- **Dirichlet**: weakly or strongly implied. Required at inflow boundary. Unstable for outflow boundary. At a closed boundary ($\vec{u} \cdot \vec{n} = 0$) only works in combination with diffusion.
- **Neumann**: typically only applied at a closed boundary. The value $g$ that is provided specifies the diffusive flux:

$$\kappa \vec{n} \cdot \nabla T = g$$

- **Robin**: Only works with CV and CG. Specify values $C_0$ and $C_1$ such that:

$$C_1 T + \kappa \vec{n} \cdot \nabla T = C_0$$

# BCs for incompressible Navier Stokes

Dirichlet bcs are applied to velocity components individually. Choice between:

- **Cartesian aligned**. Specify bcs for $u$,$v$ and $w$ components (in $x, y$ and $z$ directions). Works well in Cartesian, box-like domains or if all components are specified.
- **Surface aligned** (also known as **rotated boundary conditions**). Specify bcs in normal and tangential directions respectively. Requires computation of normal vectors on the nodes. Can usually be avoided.

# BCs for incompressible Navier Stokes

## Closed boundaries

- **No slip:** Apply Dirichlet boundary conditions in all directions (weakly or strongly), setting all components to zero.
- **Free slip:** Recommended: apply the so called no_normal_flow boundary condition to weakly impose the no normal flow condition, g. This condition can be imposed strongly, but for non-Cartesian aligned domains this requires rotated boundary conditions.
- **Bottom or wind drag:** Same as free slip but additionally specifying a bottom/wind drag:

$$\tau_{\text{bottom}} = C_D |u| \vec{u},$$
$$\tau_{\text{wind}} = C_w \left\| \vec{u}_{\text{wind}} - \vec{u} \right\| \left( \vec{u}_{\text{wind}} - \vec{u} \right)$$

# BCs for incompressible Navier Stokes

Considerations for **open boundaries**:

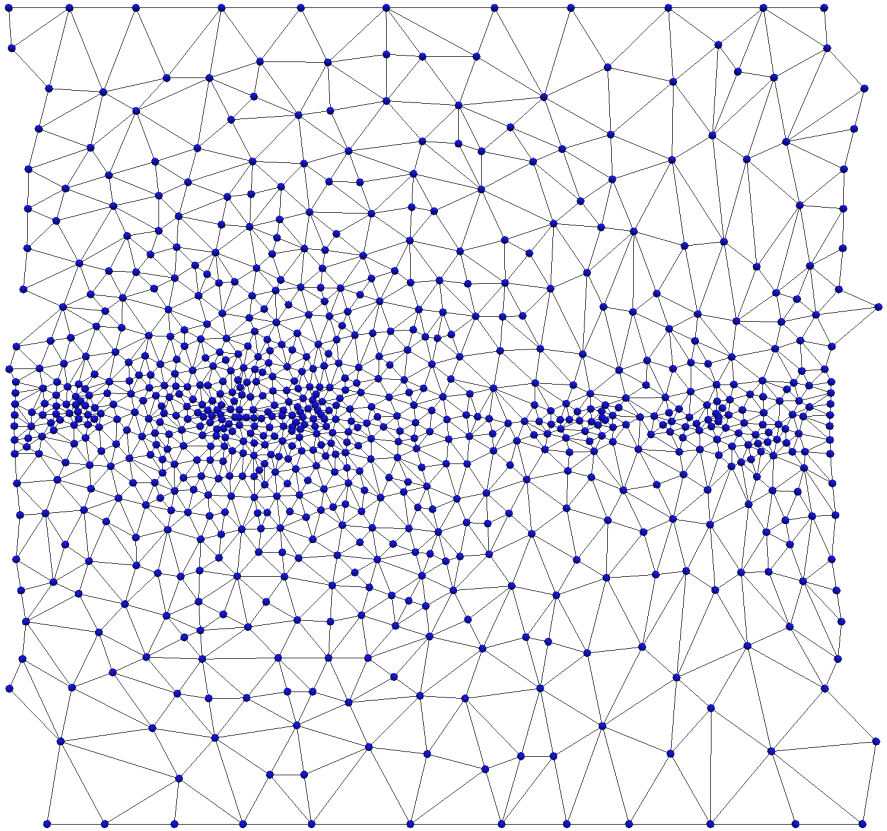
- For incompressible flows, the inflow has to be the same as the outflow.
- Ideally, only the inflow should be specified.
- The influence of the inflow boundary condition is immediately felt throughout the domain.
- The outflow condition is a free stress condition, which means that a zero pressure is assumed on the outside of the domain.

# BCs for incompressible Navier Stokes

Lateral open boundary conditions are often better behaved in combination with a **free surface** condition at the top of the domain. In that case:

- Inflow does not travel immediately across the domain (finite wave speed) and storage of water is possible.
- Can specify a velocity boundary condition in both directions (in- and outflow), e.g. periodic tidal forcing. Better avoided in areas with strong non-linear effects.
- Free surface boundary condition is specified through pressure boundary condition: $p = g\eta$. Assumes no non-hydrostatic effects and geostrophic pressure should be subtracted.
- Velocity boundary conditions on all open sides easily leads to a drift in the total volume.

# Periodic bcs



Periodic boundary conditions are implemented by identifying nodes on opposite sides of the periodic boundary. This leads to a new function space in which all solutions are automatically periodic.

# Source/Absorption

Two terms can be added to the advection diffusion equation:

$$\frac{\partial T}{\partial t} + u \cdot \nabla T - \kappa \nabla^2 T + \alpha T = s,$$

with absorption coefficient $\alpha$ and source $s$. Similarly, in the momentum equation

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} - \nu \nabla^2 \vec{u} + \nabla P + \vec{\sigma} \cdot \vec{u} = \vec{f},$$

with absorption vector $\vec{\sigma}$ and momentum source (force) $\vec{f}$.

# Sponge regions

Source and absorption terms can be used to define sponges in some regions. To relax to a value $T_r$, choose:

$$s = \alpha T_r, \quad \alpha = 1/\tau$$

where $\tau$ is the relaxation time. This relaxation time should be larger than your time step and smaller than the time scale of the dynamics you want to dampen out.
Best to ramp up the value of $\alpha$ towards the sponge region to avoid reflections.

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# Linear solvers

## Computational methods to solver linear systems

- Direct solvers. Only smallish problems (typically 2D)
- Iterative solvers - we use the PETSc library

For iterative solvers we need to choose:

- Krylov subspace method: CG or GMRES
- Preconditioners: SOR or MG (algebraic multigrid)
- Stopping criterion:
  - relative: $r < C_{\text{rel}}\ r_0$,
  - absolute: $r < C_{\text{abs}}$,

  where $r$ is the preconditioned residual.
- Maximum number of iterations.

**Overview iterative methods:** Saad [1996]

# Pressure solve options

The Pressure Poisson equation yields a symmetric positive definite matrix. This means we can use CG + SSOR (symmetric SOR=default). For bigger and more complex problems CG + MG is more efficient.

For large aspect ratio problems, where the horizontal length scales are orders bigger than the vertical, use CG + MG + vertical_lumping.[Kramer et al., 2010]

If all boundaries are closed (or have specified normal velocity) the pressure equation is not well-posed; We can add any constant to the solution. We need to tell this to the solver with remove_null_space option.

# Momentum and scalar equations

Due to the advection term the matrices resulting from discretising the scalar advection-diffusion equation and the momentum equation are non-symmetric. Therefore we choose GMRES+SOR.

High Courant numbers lead to high number of iterations or non-convergence.

For pure diffusion equations (heat equation) CG+SOR/MG can be used.

# Outline

Time integration

Scalar advection-diffusion

Incompressible Navier Stokes

Time loop

Boundary conditions

Linear solvers

References

# References

B. Cockburn and C-W. Shu. Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems. **Journal of Scientific Computing**, 16(3):173–261, 2001.

C. J. Cotter, D. A. Ham, and C. C. Pain. A mixed discontinuous/continuous finite element pair for shallow-water ocean modelling. **Ocean Modelling**, 26:86–90, 2009.

Jean Donea and Antonio Huerta. **Finite Element Methods for Flow Problems**. John Wiley & Sons, Ltd., 2003. ISBN 0471496669.

P. M. Gresho and R. L. Sani. **Incompressible Flow and the Finite Element Method**. John Wiley and Sons, 1998.

Jon Hill, MD Piggott, David A Ham, EE Popova, and MA Srokosz. On the performance of a generic length scale turbulence model within an adaptive finite element ocean model. **Ocean Modelling**, 2012.

Stephan C. Kramer, Colin J. Cotter, and Christopher C. Pain. Solving the poisson equation on small aspect ratio domains using unstructured meshes. **Ocean Modelling**, 35(3):253–263, 2010. doi: 10.1016/j.pepi.2012.01.001.

R J LeVeque. **Finite volume methods for hyperbolic problems**. Cambridge University Press, 2002.

CC Pain, MD Piggott, AJH Goddard, F Fang, GJ Gorman, DP Marshall, MD Eaton, PW Power, and CRE De Oliveira. Three-dimensional unstructured mesh ocean modelling. **Ocean Modelling**, 10(1):5–33, 2005.

Stephen B. Pope. **Turbulent Flows**. Cambridge University Press, 2000.

Yousef Saad. **Iterative methods for sparse linear systems**, volume 620. PWS publishing company Boston, 1996.

L Umlauf and H Burchard. Second-order turbulence closure models for geophysical boundary layers. A review of recent work. **Continental Shelf Research**, 25:795–827, 2005. doi: {10.1016/j.csr.2004.08.004}.

D.C. Wilcox. **Turbulence modeling for CFD**. DCW industries La Canada, CA, 1998.

C. R. Wilson. **Modelling multiple-material flows on adaptive, unstructured meshes**. PhD thesis, Imperial College London, 2009.

Many more details and references in the Fluidity manual!