Partially Implicit Time Integration Methods for the Compressible Euler Equations

Oswald Knoth, Stefan Jebens, Ruediger Weiner
Institute for Tropospheric Research Leipzig,
Institute for Mathematics, University Halle

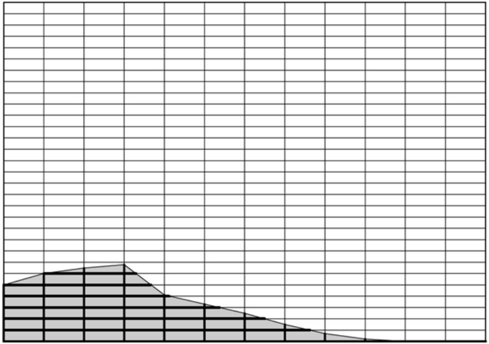
SNRWP Workshop Bad Orb

16.05.2011

LEIBNIZ-INSTITUT FÜR TROPOSPHÄRENFORSCHUNG

ASAM
All Scale Atmospheric Model

- Cut cell approach with small grid cells



- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
    - Energetically relevant slow waves (e.g. advection, Rossby waves)
    - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
    - Advantages: Every step is cheap, easy to implement, parallelization
    - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
    - Advantages: Allows very big time steps, order conditions and stability issues are obvious
    - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- In compressible models occur:
  - Energetically relevant slow waves (e.g. advection, Rossby waves)
  - Energetically irrelevant fast waves (e.g. sound waves)
- In explicit models the fast waves restrict the maximal time step size
- One ansatz to overcome this is operator splitting
  - Advantages: Every step is cheap, easy to implement, parallelization
  - Disadvantages: Still explicit (i.e. only small time steps allowed especially when used together with cut-cells), complicated derivation of order conditions and stability results
- Another ansatz is the use of implicit methods
  - Advantages: Allows very big time steps, order conditions and stability issues are obvious
  - Disadvantages: Requires solution of huge (non-)linear systems of equations, needs efficient (parallel) preconditioners

- Consider PDE discretized in space

$$y' = f(y)$$

- Rosenbrock Method

$$\begin{aligned} y_{n+1} &= y_n + \sum_{i=1}^{s} b_i k_i, \\ k_i &= \tau f\left(y_n + \sum_{j=1}^{i-1} a_{ij} k_j\right) + \Delta t W \sum_{j=1}^{i} \gamma_{ij} k_j, \quad i = 1, ..., s. \end{aligned}$$

where $W \approx J_n = f('y_n)$.

- Consider PDE discretized in space

$$y' = f(y)$$

- Rosenbrock Method

$$\begin{aligned} y_{n+1} &= y_n + \sum_{i=1}^{s} b_i k_i, \\ k_i &= \tau f\left(y_n + \sum_{j=1}^{i-1} a_{ij} k_j\right) + \Delta t W \sum_{j=1}^{i} \gamma_{ij} k_j, \quad i = 1, ..., s. \end{aligned}$$

where $W \approx J_n = f('y_n)$.

- Order conditions

Order p	Conditions
1	$\sum_{i=1}^{s} b_i = 1$
2	$\sum_{i=2}^{s} b_i a_i = 1/2$
	$\sum_{i=2}^{s} b_i d_i = 0$
3	$\sum_{i=2}^{s} b_i a_i^2 = 1/3$
	$\sum_{i=3}^{s} \sum_{j=2}^{i-1} b_i a_{ij} a_j = 1/6$
	$\sum_{i=3}^{s} \sum_{j=2}^{i-1} b_i a_{ij} d_j = 0$
	$\sum_{i=3}^{s} \sum_{j=2}^{i-1} b_i \gamma_{ij} a_j = 0$
	$\sum_{i=2}^{s} b_i d_i^2 = 0$

- Rosenbrock-W method based on RK3

$$
\begin{array}{c|ccc}
0 & & & \\
1/3 & 1/3 & & \\
1/2 & 0 & 1/2 & \\
\hline
 & 0 & 0 & 1
\end{array}
\qquad
\begin{array}{c|ccc}
\gamma & \gamma & & \\
\frac{1-9\gamma+24\gamma^2}{-9+36\gamma} & \frac{1-12\gamma^2}{-9+36\gamma} & \gamma & \\
0 & -1/4+2\gamma & 1/4-3\gamma & \gamma \\
\hline
 & 0 & 0 & 0
\end{array}
$$

$\mathcal{A}$-Matrix	Γ-Matrix

- Type of approximate Jacobian
  - Approximate matrix factorization

$$f = f_1 + f_2, \quad J = J_1 + J_2, \quad I - \gamma\tau W = (I - \gamma\Delta t J_1)(I - \gamma\Delta t J_2)$$

  - Jacobian from a low order discretization

$$f_L \approx f, \quad W = J_L$$

  - Partial Jacobian, split with respect to space or processes

$$f = f_1 + f_2, \quad W = J_1$$

  - Combine above ideas

- Type of approximate Jacobian
  - Approximate matrix factorization

$$f = f_1 + f_2, \quad J = J_1 + J_2, \quad I - \gamma\tau W = (I - \gamma\Delta t J_1)(I - \gamma\Delta t J_2)$$

  - Jacobian from a low order discretization

$$f_L \approx f, \quad W = J_L$$

  - Partial Jacobian, split with respect to space or processes

$$f = f_1 + f_2, \quad W = J_1$$

  - Combine above ideas

- Type of approximate Jacobian
  - Approximate matrix factorization

$$f = f_1 + f_2, \quad J = J_1 + J_2, \quad I - \gamma\tau W = (I - \gamma\Delta t J_1)(I - \gamma\Delta t J_2)$$

  - Jacobian from a low order discretization

$$f_L \approx f, \quad W = J_L$$

  - Partial Jacobian, split with respect to space or processes

$$f = f_1 + f_2, \quad W = J_1$$

  - Combine above ideas

- Type of approximate Jacobian
  - Approximate matrix factorization

$$f = f_1 + f_2, \quad J = J_1 + J_2, \quad I - \gamma\tau W = (I - \gamma\Delta t J_1)(I - \gamma\Delta t J_2)$$

  - Jacobian from a low order discretization

$$f_L \approx f, \quad W = J_L$$

  - Partial Jacobian, split with respect to space or processes

$$f = f_1 + f_2, \quad W = J_1$$

  - Combine above ideas

- Type of approximate Jacobian
  - Approximate matrix factorization

$$f = f_1 + f_2, \quad J = J_1 + J_2, \quad I - \gamma\tau W = (I - \gamma\Delta t J_1)(I - \gamma\Delta t J_2)$$

  - Jacobian from a low order discretization

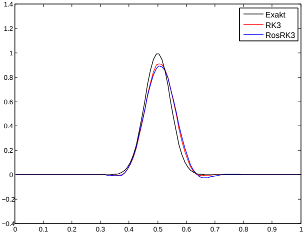
$$f_L \approx f, \quad W = J_L$$

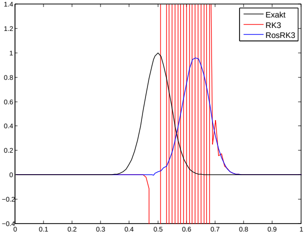
  - Partial Jacobian, split with respect to space or processes

$$f = f_1 + f_2, \quad W = J_1$$

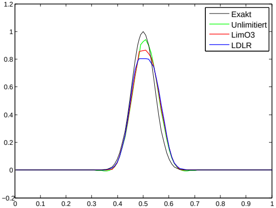
  - Combine above ideas

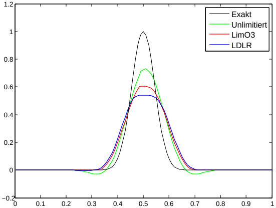
- Unit interval, 100 grid cells, comparison of a uniformly spaced grid and a uniformly grid cell with one small grid cell $h_{\text{small}} = 1/10000$.



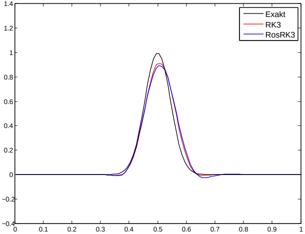


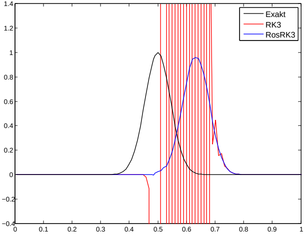
- Uniformly grid cell with one small grid cell $h_{\text{small}} = 1/10000$, one and tenth revolution of the profile, two different limiters



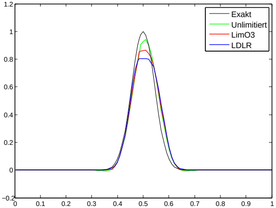


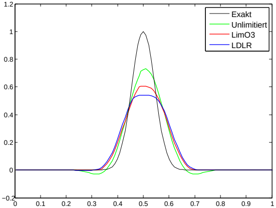
- Unit interval, 100 grid cells, comparison of a uniformly spaced grid and a uniformly grid cell with one small grid cell $h_{\text{small}} = 1/10000$.





- Uniformly grid cell with one small grid cell $h_{\text{small}} = 1/10000$, one and tenth revolution of the profile, two different limiters





- Peer Method
- Write numerical solutions as:

$$Y_m := \begin{pmatrix} Y_{m1} \\ \vdots \\ Y_{ms} \end{pmatrix} \approx \begin{pmatrix} y(t_m + c_1 \Delta t) \\ \vdots \\ y(t_m + c_s \Delta t) \end{pmatrix} \in \mathbb{R}^{s \times n}, \qquad F_m := f(Y_m) \in \mathbb{R}^{s \times n}$$

- Runge-Kutta methods (for autonomous systems) read:

$$Y_m = Y_{m-1,s} + \Delta t A F_m$$

- Explicit peer methods are defined by:

$$Y_{mi} = B_i Y_{m-1} + \Delta t A_i F_{m-1} + \Delta t R_i F_m$$

- Performing one Newton step results in the considered class of linearly implicit peer methods:

$$Y_m (I - h\gamma J)^T = B Y_{m-1} + \Delta t A F_{m-1} + \Delta t R F_m + \Delta t G Y_{m-1} J^T + \Delta t H Y_m J^T$$

- Peer Method
- Write numerical solutions as:

$$Y_m := \begin{pmatrix} Y_{m1} \\ \vdots \\ Y_{ms} \end{pmatrix} \approx \begin{pmatrix} y(t_m + c_1\Delta t) \\ \vdots \\ y(t_m + c_s\Delta t) \end{pmatrix} \in \mathbb{R}^{s\times n}, \qquad F_m := f(Y_m) \in \mathbb{R}^{s\times n}$$

- Runge-Kutta methods (for autonomous systems) read:

$$Y_m = Y_{m-1,s} + \Delta t A F_m$$

- Explicit peer methods are defined by:

$$Y_{mi} = B_i Y_{m-1} + \Delta t A_i F_{m-1} + \Delta t R_i F_m$$

- Performing one Newton step results in the considered class of linearly implicit peer methods:

$$Y_m (I - h\gamma J)^T = BY_{m-1} + \Delta t A F_{m-1} + \Delta t R F_m + \Delta t G Y_{m-1} J^T + \Delta t H Y_m J^T$$

- Peer Method
- Write numerical solutions as:

$$Y_m := \begin{pmatrix} Y_{m1} \\ \vdots \\ Y_{ms} \end{pmatrix} \approx \begin{pmatrix} y(t_m + c_1 \Delta t) \\ \vdots \\ y(t_m + c_s \Delta t) \end{pmatrix} \in \mathbb{R}^{s \times n}, \qquad F_m := f(Y_m) \in \mathbb{R}^{s \times n}$$

- Runge-Kutta methods (for autonomous systems) read:

$$Y_m = Y_{m-1,s} + \Delta t A F_m$$

- Explicit peer methods are defined by:

$$Y_{mi} = B_i Y_{m-1} + \Delta t A_i F_{m-1} + \Delta t R_i F_m$$

- Performing one Newton step results in the considered class of linearly implicit peer methods:

$$Y_m (I - h\gamma J)^T = B Y_{m-1} + \Delta t A F_{m-1} + \Delta t R F_m + \Delta t G Y_{m-1} J^T + \Delta t H Y_m J^T$$

- Peer Method
- Write numerical solutions as:

$$Y_m := \begin{pmatrix} Y_{m1} \\ \vdots \\ Y_{ms} \end{pmatrix} \approx \begin{pmatrix} y(t_m + c_1 \Delta t) \\ \vdots \\ y(t_m + c_s \Delta t) \end{pmatrix} \in \mathbb{R}^{s \times n}, \qquad F_m := f(Y_m) \in \mathbb{R}^{s \times n}$$

- Runge-Kutta methods (for autonomous systems) read:

$$Y_m = Y_{m-1,s} + \Delta t A F_m$$

- Implicit peer methods are defined by:

$$Y_{mi} = B_i Y_{m-1} + \Delta t A_i F_{m-1} + \Delta t R_i F_m + \Delta t \gamma f(Y_{mi})$$

- Performing one Newton step results in the considered class of linearly implicit peer methods:

$$Y_m (I - h\gamma J)^T = B Y_{m-1} + \Delta t A F_{m-1} + \Delta t R F_m + \Delta t G Y_{m-1} J^T + \Delta t H Y_m J^T$$

- Peer Method
- Write numerical solutions as:

$$Y_m := \begin{pmatrix} Y_{m1} \\ \vdots \\ Y_{ms} \end{pmatrix} \approx \begin{pmatrix} y(t_m + c_1 \Delta t) \\ \vdots \\ y(t_m + c_s \Delta t) \end{pmatrix} \in \mathbb{R}^{s\times n}, \qquad F_m := f(Y_m) \in \mathbb{R}^{s\times n}$$

- Runge-Kutta methods (for autonomous systems) read:

$$Y_m = Y_{m-1,s} + \Delta t A F_m$$

- Implicit peer methods are defined by:

$$Y_{mi} = B_i Y_{m-1} + \Delta t A_i F_{m-1} + \Delta t R_i F_m + \Delta t \gamma f(Y_{mi})$$

- Performing one Newton step results in the considered class of linearly implicit peer methods:

$$Y_m (I - h\gamma J)^T = BY_{m-1} + \Delta t A F_{m-1} + \Delta t R F_m + \Delta t G Y_{m-1} J^T + \Delta t H Y_m J^T$$

- Order conditions $AB(k)=0$, $\widehat{AB}(k)=0$, $k\le s$, can be written in compact matrix form

$$\begin{aligned} B\mathbb{1} &= \mathbb{1},\\ A &= CV_0D^{-1}V_1^{-1} - B(C-I)V_1D^{-1}V_1^{-1} - RV_0V_1^{-1},\\ G &= -\Gamma V_0V_1^{-1} - HV_0V_1^{-1} \end{aligned}$$

with $\mathbb{1}=(1,\dots,1)^T$, $C=\mathrm{diag}(c_1,\dots,c_s)$, $\Gamma=\gamma I$, $D=\mathrm{diag}(1,2,\dots,s)$,

$$V_0=\begin{pmatrix}1 & c_1 & \cdots & c_1^{s-1}\\ \vdots & \vdots & \ddots & \vdots\\ 1 & c_s & \cdots & c_s^{s-1}\end{pmatrix} \quad\text{and}\quad V_1=\begin{pmatrix}1 & c_1-1 & \cdots & (c_1-1)^{s-1}\\ \vdots & \vdots & \ddots & \vdots\\ 1 & c_s-1 & \cdots & (c_s-1)^{s-1}\end{pmatrix}.$$

- In the remainder we will concentrate on second-order methods with $s=2$ stages. Furthermore we choose $c_s=1$ so that $Y_{ms}\approx y(t_{m+1})$.
- Remaining parameters are c_1, γ, b_{11}, b_{21}, r_{21} and h_{21}. These will be optimized with respect to good stability properties.

- Order conditions $AB(k) = 0$, $\widehat{AB}(k) = 0$, $k \le s$, can be written in compact matrix form

$$
\begin{aligned}
B\mathbb{1} &= \mathbb{1},\\
A &= CV_0D^{-1}V_1^{-1} - B(C-I)V_1D^{-1}V_1^{-1} - RV_0V_1^{-1},\\
G &= -\Gamma V_0V_1^{-1} - HV_0V_1^{-1}
\end{aligned}
$$

with $\mathbb{1} = (1,\dots,1)^T$, $C = \mathrm{diag}(c_1,\dots,c_s)$, $\Gamma = \gamma I$,
$D = \mathrm{diag}(1,2,\dots,s)$,

$$
V_0 = \begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix} \quad \text{and} \quad V_1 = \begin{pmatrix} 1 & c_1-1 & \cdots & (c_1-1)^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s-1 & \cdots & (c_s-1)^{s-1} \end{pmatrix}.
$$

- In the remainder we will concentrate on second-order methods with $s = 2$ stages. Furthermore we choose $c_s = 1$ so that $Y_{ms} \approx y(t_{m+1})$.
- Remaining parameters are c_1, γ, b_{11}, b_{21}, r_{21} and h_{21}. These will be optimized with respect to good stability properties.

- Order conditions $AB(k)=0$, $\widehat{AB}(k)=0$, $k\le s$, can be written in compact matrix form

$$\begin{aligned} B\mathbb{1} &= \mathbb{1}, \\ A &= CV_0D^{-1}V_1^{-1} - B(C-I)V_1D^{-1}V_1^{-1} - RV_0V_1^{-1}, \\ G &= -\Gamma V_0V_1^{-1} - HV_0V_1^{-1} \end{aligned}$$

with $\mathbb{1}=(1,\dots,1)^T$, $C=\mathrm{diag}(c_1,\dots,c_s)$, $\Gamma=\gamma I$,
$D=\mathrm{diag}(1,2,\dots,s)$,

$$V_0 = \begin{pmatrix} 1 & c_1 & \cdots & c_1^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s & \cdots & c_s^{s-1} \end{pmatrix} \quad \text{and} \quad V_1 = \begin{pmatrix} 1 & c_1-1 & \cdots & (c_1-1)^{s-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & c_s-1 & \cdots & (c_s-1)^{s-1} \end{pmatrix}.$$

- In the remainder we will concentrate on second-order methods with $s=2$ stages. Furthermore we choose $c_s=1$ so that $Y_{ms}\approx y(t_{m+1})$.
- Remaining parameters are c_1, γ, b_{11}, b_{21}, r_{21} and h_{21}. These will be optimized with respect to good stability properties.

- One-dimensional compressible Euler equations in conservative form:

$$
\begin{aligned}
\dot{\rho} &= -\frac{\partial \rho u}{\partial x} \\
\dot{\rho u} &= -\frac{\partial \rho u u}{\partial x} - \frac{\partial p}{\partial x} \\
\dot{\rho\theta} &= -\frac{\partial \rho u \theta}{\partial x} \\
p &= \left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{1}{1-\kappa}}
\end{aligned}
$$

- Elimination of pressure:

$$
\begin{aligned}
\frac{\partial p}{\partial x} &= \frac{\partial p}{\partial \rho\theta}\frac{\partial \rho\theta}{\partial x} \\
\frac{\partial p}{\partial \rho\theta} &= \frac{R}{p_0^\kappa(1-\kappa)}\left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{\kappa}{1-\kappa}} = \frac{1}{\rho\theta(1-\kappa)}\left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{1}{1-\kappa}} = \frac{c_s^2}{\theta}
\end{aligned}
$$

with c_s the speed of sound

$$
c_s := \sqrt{\frac{1}{\rho(1-\kappa)}\left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{1}{1-\kappa}}}
$$

- One-dimensional compressible Euler equations in conservative form:

$$\dot{\rho} = -\frac{\partial \rho u}{\partial x}$$
$$\dot{\rho u} = -\frac{\partial \rho u u}{\partial x} - \frac{\partial p}{\partial x}$$
$$\dot{\rho\theta} = -\frac{\partial \rho u \theta}{\partial x}$$
$$p = \left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{1}{1-\kappa}}$$

- Elimination of pressure:

$$\frac{\partial p}{\partial x} = \frac{\partial p}{\partial \rho\theta}\frac{\partial \rho\theta}{\partial x}$$
$$\frac{\partial p}{\partial \rho\theta} = \frac{R}{p_0^\kappa(1-\kappa)}\left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{\kappa}{1-\kappa}} = \frac{1}{\rho\theta(1-\kappa)}\left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{1}{1-\kappa}} = \frac{c_s^2}{\theta}$$

with c_s the speed of sound

$$c_s := \sqrt{\frac{1}{\rho(1-\kappa)}\left(\frac{R\rho\theta}{p_0^\kappa}\right)^{\frac{1}{1-\kappa}}}$$

- Use of product rule for

$$\frac{\partial \rho u u}{\partial x} = -u^2 \frac{\partial \rho}{\partial x} + 2u \frac{\partial \rho u}{\partial x}$$
$$\frac{\partial \rho u \theta}{\partial x} = -u\theta \frac{\partial \rho}{\partial x} + \theta \frac{\partial \rho u}{\partial x} + u \frac{\partial \rho \theta}{\partial x}$$

- results in the nonlinear Euler equations in compact form:

$$\begin{pmatrix} \dot{\rho} \\ \dot{\rho u} \\ \dot{\rho\theta} \end{pmatrix} = -\begin{pmatrix} 0 & 1 & 0 \\ -u^2 & 2u & \frac{c_s^2}{\theta} \\ -u\theta & \theta & u \end{pmatrix} \begin{pmatrix} \rho_x \\ (\rho u)_x \\ (\rho\theta)_x \end{pmatrix}$$

- Linearization by considering the disturbed quantities (e.g. $\rho' := \rho - \overline{\rho}$) and dropping all nonlinear terms:

$$\begin{pmatrix} \dot{\rho}' \\ (\dot{\rho u})' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = -\underbrace{\begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix}}_{M} \begin{pmatrix} \rho'_x \\ (\rho u)'_x \\ \frac{1}{\overline{\theta}}(\rho\theta)'_x \end{pmatrix}$$

- Use of product rule for

$$\frac{\partial \rho u u}{\partial x} = -u^2 \frac{\partial \rho}{\partial x} + 2u \frac{\partial \rho u}{\partial x}$$
$$\frac{\partial \rho u \theta}{\partial x} = -u\theta \frac{\partial \rho}{\partial x} + \theta \frac{\partial \rho u}{\partial x} + u \frac{\partial \rho \theta}{\partial x}$$

- results in the nonlinear Euler equations in compact form:

$$\begin{pmatrix} \dot{\rho} \\ \dot{\rho u} \\ \dot{\rho\theta} \end{pmatrix} = -\begin{pmatrix} 0 & 1 & 0 \\ -u^2 & 2u & \frac{c_s^2}{\theta} \\ -u\theta & \theta & u \end{pmatrix} \begin{pmatrix} \rho_x \\ (\rho u)_x \\ (\rho\theta)_x \end{pmatrix}$$

- Linearization by considering the disturbed quantities (e.g. $\rho' := \rho - \overline{\rho}$) and dropping all nonlinear terms:

$$\begin{pmatrix} \dot{\rho}' \\ (\dot{\rho u})' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = -\underbrace{\begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix}}_{M} \begin{pmatrix} \rho'_x \\ (\rho u)'_x \\ \frac{1}{\overline{\theta}}(\rho\theta)'_x \end{pmatrix}$$

- Use of product rule for

$$\frac{\partial \rho u u}{\partial x} = -u^2 \frac{\partial \rho}{\partial x} + 2u \frac{\partial \rho u}{\partial x}$$
$$\frac{\partial \rho u \theta}{\partial x} = -u\theta \frac{\partial \rho}{\partial x} + \theta \frac{\partial \rho u}{\partial x} + u \frac{\partial \rho \theta}{\partial x}$$

- results in the nonlinear Euler equations in compact form:

$$\begin{pmatrix} \dot{\rho} \\ \dot{\rho u} \\ \dot{\rho\theta} \end{pmatrix} = - \begin{pmatrix} 0 & 1 & 0 \\ -u^2 & 2u & \frac{c_s^2}{\theta} \\ -u\theta & \theta & u \end{pmatrix} \begin{pmatrix} \rho_x \\ (\rho u)_x \\ (\rho\theta)_x \end{pmatrix}$$

- Linearization by considering the disturbed quantities (e.g. $\rho' := \rho - \overline{\rho}$) and dropping all nonlinear terms:

$$\begin{pmatrix} \dot{\rho}' \\ \dot{(\rho u)}' \\ \frac{1}{\overline{\theta}} \dot{(\rho\theta)}' \end{pmatrix} = \underbrace{- \begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix}}_{M} \begin{pmatrix} \rho'_x \\ (\rho u)'_x \\ \frac{1}{\overline{\theta}} (\rho\theta)'_x \end{pmatrix}$$

- To save storage and gain computational efficiency we make two simplifications for the Jacobian J:
    - Use Jacobian of the advection form of the Euler equations
    - Use first-order upwind scheme for spatial discretization
- Use $\overline{\rho}u' \approx (\rho u)' - \overline{u}\rho$ instead of $(\rho u)'$, i.e. use:

$$\widetilde{M} := -\begin{pmatrix} 1 & 0 & 0 \\ -\overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix}$$

- It holds:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\rho\dot{\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} \rho_x' \\ (\overline{\rho}u')_x \\ \frac{1}{\overline{\theta}}(\rho\theta)_x' \end{pmatrix}$$

- To save storage and gain computational efficiency we make two simplifications for the Jacobian J:
  - Use Jacobian of the advection form of the Euler equations
  - Use first-order upwind scheme for spatial discretization
- Use $\overline{\rho}u' \approx (\rho u)' - \overline{u}\rho$ instead of $(\rho u)'$, i.e. use:

$$\widetilde{M} := -\begin{pmatrix} 1 & 0 & 0 \\ -\overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix}$$

- It holds:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\rho\dot{\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} \rho'_x \\ (\overline{\rho}u')_x \\ \frac{1}{\overline{\theta}}(\rho\theta)'_x \end{pmatrix}$$

- To save storage and gain computational efficiency we make two simplifications for the Jacobian J:
  - Use Jacobian of the advection form of the Euler equations
  - Use first-order upwind scheme for spatial discretization
- Use $\overline{\rho}u' \approx (\rho u)' - \overline{u}\rho$ instead of $(\rho u)'$, i.e. use:

$$\widetilde{M} := -\begin{pmatrix} 1 & 0 & 0 \\ -\overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix}$$

- It holds:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\rho\dot{\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} \rho'_x \\ (\overline{\rho}u')_x \\ \frac{1}{\overline{\theta}}(\rho\theta)'_x \end{pmatrix}$$

- To save storage and gain computational efficiency we make two simplifications for the Jacobian J:
  - Use Jacobian of the advection form of the Euler equations
  - Use first-order upwind scheme for spatial discretization
- Use $\overline{\rho}u' \approx (\rho u)' - \overline{u}\rho$ instead of $(\rho u)'$, i.e. use:

$$\widetilde{M} := -\begin{pmatrix} 1 & 0 & 0 \\ -\overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix}$$

- It holds:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} \rho'_x \\ (\overline{\rho}u')_x \\ \frac{1}{\overline{\theta}}(\rho\theta)'_x \end{pmatrix}$$

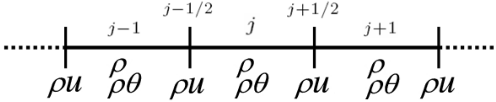
- To save storage and gain computational efficiency we make two simplifications for the Jacobian J:
  - Use Jacobian of the advection form of the Euler equations
  - Use first-order upwind scheme for spatial discretization
- Use $\overline{\rho}u' \approx (\rho u)' - \overline{u}\rho$ instead of $(\rho u)'$, i.e. use:

$$\widetilde{M} := -\begin{pmatrix} 1 & 0 & 0 \\ -\overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -\overline{u}^2 & 2\overline{u} & c_s^2 \\ -\overline{u} & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ \overline{u} & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix}$$

- It holds:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\rho\dot{\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u} & 1 & 0 \\ 0 & \overline{u} & c_s^2 \\ 0 & 1 & \overline{u} \end{pmatrix} \begin{pmatrix} \rho'_x \\ (\overline{\rho}u')_x \\ \frac{1}{\overline{\theta}}(\rho\theta)'_x \end{pmatrix}$$

- Variables are defined on a staggered grid



- For investigation of spatial discretizations perform von Neumann stability analysis, e.g. it holds:

$$\rho u(t, x_{j+1/2}) = \rho u(t) e^{ikx_{j+1/2}}$$

$$\Rightarrow \quad \left.\frac{\partial \rho u}{\partial x}\right|_{(t,x_j)} = \rho u(t) \frac{e^{ikx_j}}{\Delta x}\left(e^{\frac{ik\Delta x}{2}} - e^{-\frac{ik\Delta x}{2}}\right)$$

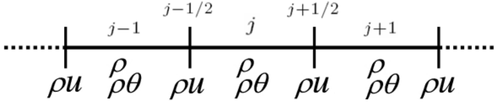
- Three spatial discretizations appear:

$$\mathcal{D}_1 = \frac{1}{\Delta x}\left(1 - e^{-ik\Delta x}\right)$$

$$\mathcal{D}_2 = \frac{1}{\Delta x}\left(e^{\frac{ik\Delta x}{2}} - e^{-\frac{ik\Delta x}{2}}\right)$$

$$\mathcal{D}_3 = \frac{1}{6\Delta x}\left(2e^{ik\Delta x} + 3 - 6e^{-ik\Delta x} + e^{-2ik\Delta x}\right)$$

- Variables are defined on a staggered grid



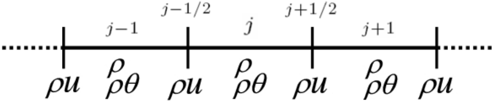
- For investigation of spatial discretizations perform von Neumann stability analysis, e.g. it holds:

$$
\begin{aligned}
&\rho u(t, x_{j+1/2}) = \rho u(t) e^{ikx_{j+1/2}} \\
\Rightarrow \quad &\left.\frac{\partial \rho u}{\partial x}\right|_{(t,x_j)} = \rho u(t) \frac{e^{ikx_j}}{\Delta x}\left(e^{\frac{ik\Delta x}{2}} - e^{-\frac{ik\Delta x}{2}}\right)
\end{aligned}
$$

- Three spatial discretizations appear:

$$
\begin{aligned}
\mathcal{D}_1 &= \frac{1}{\Delta x}\left(1 - e^{-ik\Delta x}\right) \\
\mathcal{D}_2 &= \frac{1}{\Delta x}\left(e^{\frac{ik\Delta x}{2}} - e^{-\frac{ik\Delta x}{2}}\right) \\
\mathcal{D}_3 &= \frac{1}{6\Delta x}\left(2e^{ik\Delta x} + 3 - 6e^{-ik\Delta x} + e^{-2ik\Delta x}\right)
\end{aligned}
$$

- Variables are defined on a staggered grid



- For investigation of spatial discretizations perform von Neumann stability analysis, e.g. it holds:

$$
\begin{aligned}
&\rho u(t, x_{j+1/2}) = \rho u(t) e^{ikx_{j+1/2}} \\
\Rightarrow \quad &\left.\frac{\partial \rho u}{\partial x}\right|_{(t,x_j)} = \rho u(t) \frac{e^{ikx_j}}{\Delta x}\left(e^{\frac{ik\Delta x}{2}} - e^{-\frac{ik\Delta x}{2}}\right)
\end{aligned}
$$

- Three spatial discretizations appear:

$$
\begin{aligned}
\mathcal{D}_1 &= \frac{1}{\Delta x}\left(1 - e^{-ik\Delta x}\right) \\
\mathcal{D}_2 &= \frac{1}{\Delta x}\left(e^{\frac{ik\Delta x}{2}} - e^{-\frac{ik\Delta x}{2}}\right) \\
\mathcal{D}_3 &= \frac{1}{6\Delta x}\left(2e^{ik\Delta x} + 3 - 6e^{-ik\Delta x} + e^{-2ik\Delta x}\right)
\end{aligned}
$$

- Using these operators results in the ODE:

$$\begin{pmatrix} \dot{\rho}' \\ (\dot{\rho u})' \\ \frac{1}{\overline{\theta}}(\rho\dot{\theta})' \end{pmatrix} = -\begin{pmatrix} 0 & \mathcal{D}_2 & 0 \\ -\overline{u}^2\mathcal{D}_3 & 2\overline{u}\mathcal{D}_3 & c_s^2\mathcal{D}_2 \\ -\overline{u}\mathcal{D}_3 & \mathcal{D}_2 & \overline{u}\mathcal{D}_3 \end{pmatrix} \begin{pmatrix} \rho' \\ (\rho u)' \\ \frac{1}{\overline{\theta}}(\rho\theta)' \end{pmatrix}$$

- For the Jacobian we instead use the matrix which belongs to:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\rho\dot{\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u}\mathcal{D}_1 & \mathcal{D}_2 & 0 \\ 0 & \overline{u}\mathcal{D}_1 & c_s^2\mathcal{D}_2 \\ 0 & \mathcal{D}_2 & \overline{u}\mathcal{D}_1 \end{pmatrix} \begin{pmatrix} \rho' \\ \overline{\rho}u' \\ \frac{1}{\overline{\theta}}(\rho\theta)' \end{pmatrix}$$

- **Remark:** While M and $\widetilde{M}$ are similar the matrices

$$A := \begin{pmatrix} 0 & \mathcal{D}_2 & 0 \\ -\overline{u}^2\mathcal{D}_3 & 2\overline{u}\mathcal{D}_3 & c_s^2\mathcal{D}_2 \\ -\overline{u}\mathcal{D}_3 & \mathcal{D}_2 & \overline{u}\mathcal{D}_3 \end{pmatrix} \quad \text{and} \quad \widetilde{A} := \begin{pmatrix} \overline{u}\mathcal{D}_1 & \mathcal{D}_2 & 0 \\ 0 & \overline{u}\mathcal{D}_1 & c_s^2\mathcal{D}_2 \\ 0 & \mathcal{D}_2 & \overline{u}\mathcal{D}_1 \end{pmatrix}$$

are not similar.

- Using these operators results in the ODE:

$$\begin{pmatrix} \dot{\rho}' \\ (\dot{\rho u})' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = - \begin{pmatrix} 0 & \mathcal{D}_2 & 0 \\ -\overline{u}^2\mathcal{D}_3 & 2\overline{u}\mathcal{D}_3 & c_s^2\mathcal{D}_2 \\ -\overline{u}\mathcal{D}_3 & \mathcal{D}_2 & \overline{u}\mathcal{D}_3 \end{pmatrix} \begin{pmatrix} \rho' \\ (\rho u)' \\ \frac{1}{\overline{\theta}}(\rho\theta)' \end{pmatrix}$$

- For the Jacobian we instead use the matrix which belongs to:

$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = - \begin{pmatrix} \overline{u}\mathcal{D}_1 & \mathcal{D}_2 & 0 \\ 0 & \overline{u}\mathcal{D}_1 & c_s^2\mathcal{D}_2 \\ 0 & \mathcal{D}_2 & \overline{u}\mathcal{D}_1 \end{pmatrix} \begin{pmatrix} \rho' \\ \overline{\rho}u' \\ \frac{1}{\overline{\theta}}(\rho\theta)' \end{pmatrix}$$

- **Remark:** While M and $\widetilde{M}$ are similar the matrices

$$A := \begin{pmatrix} 0 & \mathcal{D}_2 & 0 \\ -\overline{u}^2\mathcal{D}_3 & 2\overline{u}\mathcal{D}_3 & c_s^2\mathcal{D}_2 \\ -\overline{u}\mathcal{D}_3 & \mathcal{D}_2 & \overline{u}\mathcal{D}_3 \end{pmatrix} \quad \text{and} \quad \widetilde{A} := \begin{pmatrix} \overline{u}\mathcal{D}_1 & \mathcal{D}_2 & 0 \\ 0 & \overline{u}\mathcal{D}_1 & c_s^2\mathcal{D}_2 \\ 0 & \mathcal{D}_2 & \overline{u}\mathcal{D}_1 \end{pmatrix}$$

are not similar.

- Using these operators results in the ODE:

$$\begin{pmatrix} \dot{\rho}' \\ (\dot{\rho u})' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = -\begin{pmatrix} 0 & \mathcal{D}_2 & 0 \\ -\overline{u}^2\mathcal{D}_3 & 2\overline{u}\mathcal{D}_3 & c_s^2\mathcal{D}_2 \\ -\overline{u}\mathcal{D}_3 & \mathcal{D}_2 & \overline{u}\mathcal{D}_3 \end{pmatrix} \begin{pmatrix} \rho' \\ (\rho u)' \\ \frac{1}{\overline{\theta}}(\rho\theta)' \end{pmatrix}$$

- For the Jacobian we instead use the matrix which belongs to:

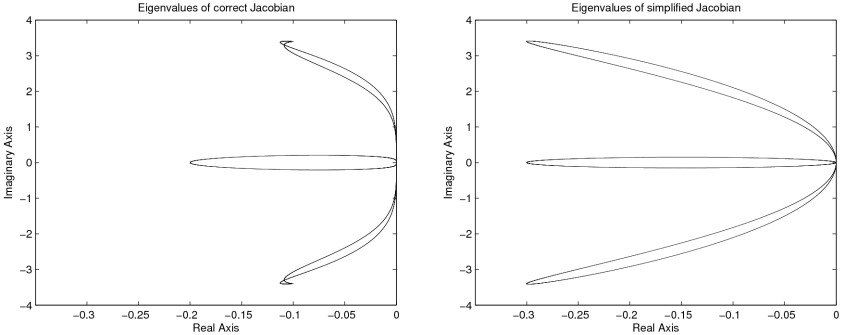
$$\begin{pmatrix} \dot{\rho}' \\ \overline{\rho}\dot{u}' \\ \frac{1}{\overline{\theta}}(\dot{\rho\theta})' \end{pmatrix} = -\begin{pmatrix} \overline{u}\mathcal{D}_1 & \mathcal{D}_2 & 0 \\ 0 & \overline{u}\mathcal{D}_1 & c_s^2\mathcal{D}_2 \\ 0 & \mathcal{D}_2 & \overline{u}\mathcal{D}_1 \end{pmatrix} \begin{pmatrix} \rho' \\ \overline{\rho}u' \\ \frac{1}{\overline{\theta}}(\rho\theta)' \end{pmatrix}$$

- **Remark:** While M and $\widetilde{M}$ are similar the matrices

$$A := \begin{pmatrix} 0 & \mathcal{D}_2 & 0 \\ -\overline{u}^2\mathcal{D}_3 & 2\overline{u}\mathcal{D}_3 & c_s^2\mathcal{D}_2 \\ -\overline{u}\mathcal{D}_3 & \mathcal{D}_2 & \overline{u}\mathcal{D}_3 \end{pmatrix} \quad \text{and} \quad \widetilde{A} := \begin{pmatrix} \overline{u}\mathcal{D}_1 & \mathcal{D}_2 & 0 \\ 0 & \overline{u}\mathcal{D}_1 & c_s^2\mathcal{D}_2 \\ 0 & \mathcal{D}_2 & \overline{u}\mathcal{D}_1 \end{pmatrix}$$

are not similar.

Eigenvalues of correct and simplified Jacobian



- Application of the peer method to the Dahlquist test equation

$$\dot{y} = \lambda y$$

leads to:

$$(1 - \Delta t \gamma J) Y_m = B Y_{m-1} + \Delta t A \lambda Y_{m-1} + \Delta t R \lambda Y_m + \Delta t G J Y_{m-1} + \Delta t H J Y_m$$

- With notations $z := \Delta t \lambda$ and $\widetilde{z} := \Delta t J$ it holds:

$$Y_m = (I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G) Y_{m-1}$$

- Side conditions for optimization are
  - A-stability in common sense, i.e. for $\widetilde{z} = z$
  - A-stability for simplified Jacobian, i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$

- Application of the peer method to the Dahlquist test equation

$$\dot{y} = \lambda y$$

leads to:

$$(1-\Delta t\gamma J)Y_m = BY_{m-1}+\Delta t A\lambda Y_{m-1}+\Delta t R\lambda Y_m+\Delta t GJY_{m-1}+\Delta t HJY_m$$

- With notations $z := \Delta t\lambda$ and $\widetilde{z} := \Delta t J$ it holds:

$$Y_m = (I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)Y_{m-1}$$

- Side conditions for optimization are
  - A-stability in common sense, i.e. for $\widetilde{z} = z$
  - A-stability for simplified Jacobian, i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$

- Application of the peer method to the Dahlquist test equation

$$\dot{y} = \lambda y$$

leads to:

$$(1-\Delta t\gamma J)Y_m = BY_{m-1}+\Delta tA\lambda Y_{m-1}+\Delta tR\lambda Y_m+\Delta tGJY_{m-1}+\Delta tHJY_m$$

- With notations $z := \Delta t\lambda$ and $\widetilde{z} := \Delta tJ$ it holds:

$$Y_m = (I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)Y_{m-1}$$

- Side conditions for optimization are
    - A-stability in common sense, i.e. for $\widetilde{z} = z$
    - A-stability for simplified Jacobian, i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$

- Application of the peer method to the Dahlquist test equation

$$\dot{y} = \lambda y$$

leads to:

$$(1-\Delta t\gamma J)Y_m = BY_{m-1}+\Delta tA\lambda Y_{m-1}+\Delta tR\lambda Y_m+\Delta tGJY_{m-1}+\Delta tHJY_m$$

- With notations $z := \Delta t\lambda$ and $\widetilde{z} := \Delta tJ$ it holds:

$$Y_m = (I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)Y_{m-1}$$

- Side conditions for optimization are
  - A-stability in common sense, i.e. for $\widetilde{z} = z$
  - A-stability for simplified Jacobian, i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$

- Application of the peer method to the Dahlquist test equation

$$\dot{y} = \lambda y$$

leads to:

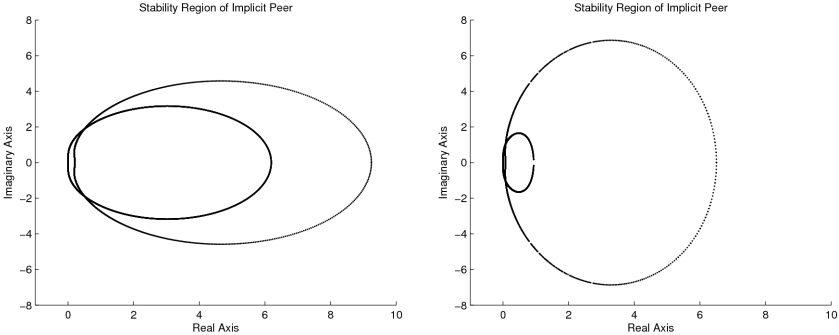
$$(1-\Delta t\gamma J)Y_m = BY_{m-1}+\Delta tA\lambda Y_{m-1}+\Delta tR\lambda Y_m+\Delta tGJY_{m-1}+\Delta tHJY_m$$

- With notations $z := \Delta t\lambda$ and $\widetilde{z} := \Delta tJ$ it holds:

$$Y_m = (I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)Y_{m-1}$$

- Side conditions for optimization are
  - A-stability in common sense, i.e. for $\widetilde{z} = z$
  - A-stability for simplified Jacobian, i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$

Stability regions for exact and simplified Jacobian



- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re}z} e^{i\mathrm{Im}z} y(t_{m-1}),$$

i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re}z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan \frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im}z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$)

- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re}z} e^{i\mathrm{Im}z} y(t_{m-1}),$$

  i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re}z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan \frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im}z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$)

- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re} z} e^{i \mathrm{Im} z} y(t_{m-1}),$$

i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re} z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan \frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im} z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re} z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im} z$)

- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re} z} e^{i \mathrm{Im} z} y(t_{m-1}),$$

  i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re} z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan \frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im} z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re} z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im} z$)

- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re}z} e^{i\mathrm{Im}z} y(t_{m-1}),$$

  i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re}z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan \frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im}z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$)

- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re}z} e^{i\mathrm{Im}z} y(t_{m-1}),$$

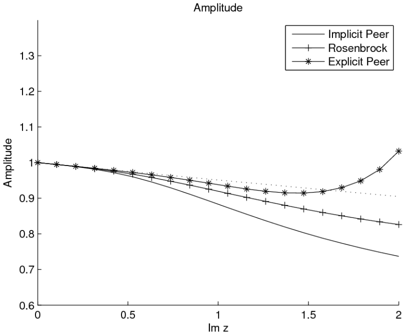
  i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re}z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan \frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im}z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$)

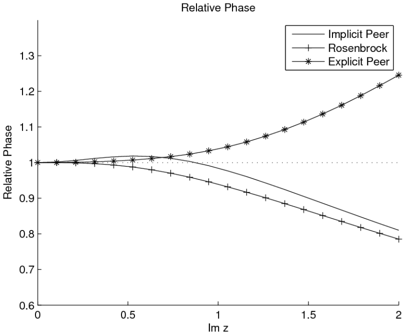
- The analytical solution of the Dahlquist test equation satisfies

$$y(t_m) = e^z y(t_{m-1}) = e^{\mathrm{Re}z} e^{i\mathrm{Im}z} y(t_{m-1}),$$

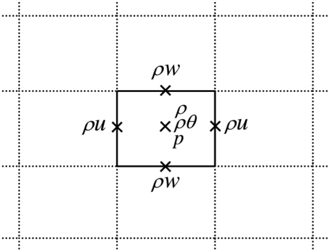
  i.e. the analytical solution has
  - the amplification factor $e^{\mathrm{Re}z}$
  - the relative phase speed 1
- Let λ be an eigenvalue of the amplification matrix $(I - zR - \widetilde{z}(\gamma I + H))^{-1}(B + zA + \widetilde{z}G)$ of a peer method applied to the Dahlquist test equation
  - The amplification factor is $|\lambda|$
  - The relative phase speed is $\frac{\arctan\frac{\mathrm{Im}\lambda}{\mathrm{Re}\lambda}}{\mathrm{Im}\lambda}$
- Optimization goal are good amplitude and phase errors for the case $z = (-0.05 + i)\mathrm{Im}z$ (i.e. eigenvalues of advection and acoustics) when using the simplified Jacobian (i.e. for $\mathrm{Re}\widetilde{z} = 2.5\mathrm{Re}z$, $\mathrm{Im}\widetilde{z} = \mathrm{Im}z$)

Amplitude and phase for the simplified Jacobian





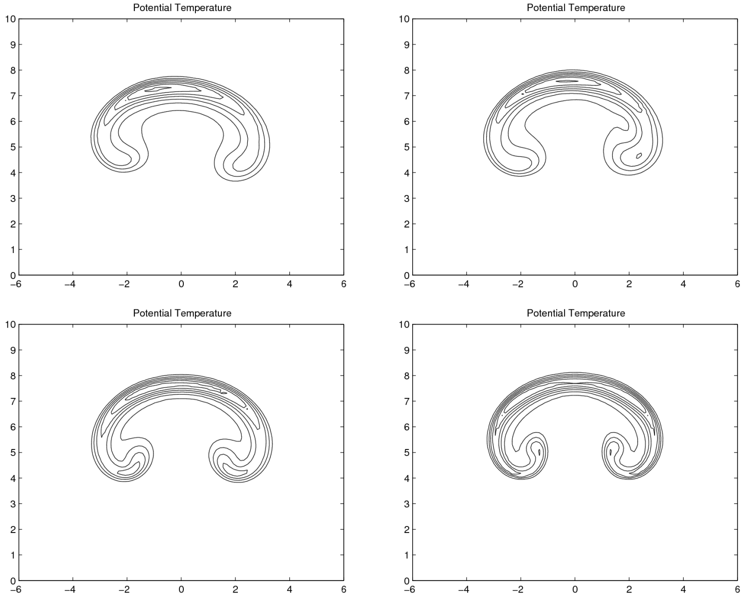
$$
\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\frac{\partial \rho u}{\partial x} - \frac{\partial \rho w}{\partial z} \\
\frac{\partial \rho u}{\partial t} &= -\frac{\partial \rho u u}{\partial x} - \frac{\partial \rho w u}{\partial z} - \frac{R}{1-\kappa}\pi\frac{\partial \rho\theta}{\partial x} \\
\frac{\partial \rho w}{\partial t} &= -\frac{\partial \rho u w}{\partial x} - \frac{\partial \rho w w}{\partial z} - \frac{R}{1-\kappa}\pi\frac{\partial \rho\theta}{\partial z} - \rho g \\
\frac{\partial \rho\theta}{\partial t} &= -\frac{\partial \rho u\theta}{\partial x} - \frac{\partial \rho w\theta}{\partial z} \\
\pi &= \left(\frac{R\rho\theta}{p_0}\right)^{\frac{\kappa}{1-\kappa}}
\end{aligned}
$$



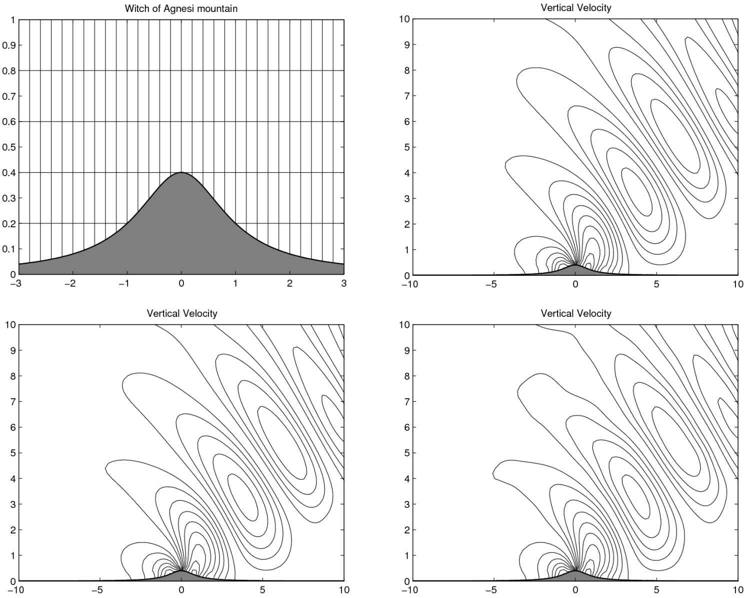
$$
\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\frac{\partial \rho u}{\partial x} - \frac{\partial \rho w}{\partial z} \\
\frac{\partial \rho u}{\partial t} &= -\frac{\partial \rho u u}{\partial x} - \frac{\partial \rho w u}{\partial z} - \frac{R}{1-\kappa}\pi\frac{\partial \rho\theta}{\partial x} \\
\frac{\partial \rho w}{\partial t} &= -\frac{\partial \rho u w}{\partial x} - \frac{\partial \rho w w}{\partial z} - \frac{R}{1-\kappa}\pi\frac{\partial \rho\theta}{\partial z} - \rho g \\
\frac{\partial \rho\theta}{\partial t} &= -\frac{\partial \rho u\theta}{\partial x} - \frac{\partial \rho w\theta}{\partial z} \\
\pi &= \left(\frac{R\rho\theta}{p_0}\right)^{\frac{\kappa}{1-\kappa}}
\end{aligned}
$$

	correct Jacobian	simplified Jacobian	ratio
1D	$3\mathcal{D}_2 + 4\mathcal{D}_3 = 22$	$3\mathcal{D}_2 + 3\mathcal{D}_1 = 12$	55%
2D	$6\mathcal{D}_2 + 14\mathcal{D}_3 = 68$	$6\mathcal{D}_2 + 8\mathcal{D}_1 = 28$	41%
3D	$9\mathcal{D}_2 + 30\mathcal{D}_3 = 138$	$9\mathcal{D}_2 + 15\mathcal{D}_1 = 48$	35%

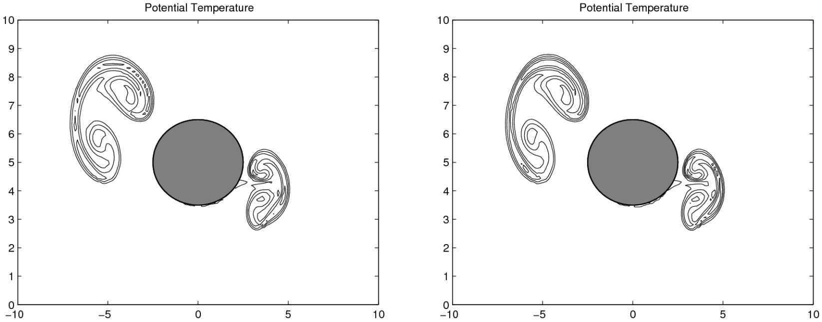
Rising bubble



Flow over mountain



Zeppelin test



Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Conclusions

- Development of a linearly implicit two-stage peer method which
  - is second-order independently of the Jacobian
  - is A-stable in the common sense and for the simplified Jacobian
  - has acceptable amplitude and phase errors
- Despite of the large CFL numbers the solutions of the linearly implicit peer method are as good as the solutions computed with the explicit method with tiny time steps
- Only exception is the transported rising bubble where the impact of damping and phase errors is visible, but
  - the explicit method is a three-stage method, there is no explicit two-stage method which is stable with the time steps used in the first test
  - the implicit peer method might not be the best one, perhaps there are better optimization criteria

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Outlook

- Determination of the practical speed-up when using the simplified Jacobian instead of the correct one
- Mixing of linearly implicit and explicit peer methods:
  - Use of full Jacobian in regions where orography results in cut-cells
  - In free regions without cut-cells only the parts of the Jacobian which come from acoustics have non-zeros entries
- Such a peer method should
  - compute with time step sizes restricted only by the CFL condition of the underlying explicit method in the free regions
  - produce as good results as the split-explicit peer method

Danke für eure/Ihre Aufmerksamkeit!