universität wien

FWF
Der Wissenschaftsfonds.

Implicit-Explicit Time Integration Methods for Astrophysical Applications

Friedrich Kupka

jointly with

**Natalie Happenhofer, Hannes Grimm-Strele,
Herbert Muthsam**

Faculty of Mathematics

University of Vienna, Austria

Contributors and Collaborators

Inmaculada Higueras

Universidad Pública de Navarra, Pamplona, Spain

Othmar Koch

ASC, Vienna University of Technology, Austria

Bernhard Löw-Baselli, Patrick Blies, Eva Mundprecht

Faculty of Mathematics, University of Vienna, Austria

Jérôme Ballot

Laboratoire d'Astrophysique, Obs. de Midi-Pyréenées, Toulouse, France

Outline

- Motivation
- Kwatra’s Method
- IMEX RK Methods
- Applications: Semiconvection
- New SSP IMEX-RK Methods
- Results
- Applications: Cepheids

Main Sources

- Kupka, F., Happenhofer, N., Higueras, I., Koch, O., *Total–Variation–Diminishing Implicit–Explicit Runge–Kutta Methods for the Simulation of Double-Diffusive Convection in Astrophysics*, J. Comput. Phys. 231, 3561–3586 (2012)
- Happenhofer, N., Grimm-Strele, H., Kupka, F., Löw-Baselli, B., Muthsam, H., *A Low Mach Number Solver: Enhancing Stability and Applicability*, J. Comput. Phys. 236, 96–118, (2013)
- Higueras, I., Happenhofer, N., Koch, O., Kupka, F., *Optimized Strong Stability Preserving IMEX Runge–Kutta Methods*, submitted (2013) (extended preprint: http://www.asc.tuwien.ac.at/preprint/2012/asc14x2012.pdf)
- Preprints: ASC Reports: Koch et al. (32/2010), Happenhofer et al. (27/2011), Higueras et al. (14/2012)
- N. Happenhofer, PhD thesis, Univ. of Vienna, 2013, to be submitted
- J.F.B.M. Kraaijevanger, *Contractivity of Runge–Kutta methods*, BIT 31, 482–528 (1991)

Motivation I

The problem of time step restrictions

- *Given a physical system and a set of dynamical equations $dy(t)/dt = f(y(t))$ describing its evolution in time*
- compute approximate solution as a function of time: $Y_n = Y_n(t) = Y(t_n) \sim y(t_n)$
- and choose $t_{n+1} - t_n$ such that the solution $Y(t)$ changes only by a few percent: $|Y_{n+1} - Y_n| < C\,|Y_n|$ with $C \sim 0.05 \ldots 0.1$ → sufficiently accurate advancing
- Explicit integration methods require to resolve the time dependence of all the processes described by the equations (independent of their amplitude, etc.): $|Y_{n+1} - Y_n| < D\,|Y_n|$ with $D \ll C$.
- Implicit methods can be less restrictive and ideally reach $D \sim C$ with $D \leq C$.
- But additional properties are often required for acceptable approximations.
- And fully implicit methods are expensive... so, how to get around this ?

Motivation II

Motivation for methods discussed below

- for a two-species flow as in the problem of double-diffusive convection the Navier-Stokes equation can be recast as

$$\underbrace{\frac{d}{dt}\begin{pmatrix}\rho\\ \rho c\\ \rho\mathbf{u}\\ e\end{pmatrix}}_{\dot{y}(t)} = \underbrace{-\nabla\cdot\begin{pmatrix}\rho\mathbf{u}\\ \rho c\mathbf{u}\\ \rho\mathbf{u}\otimes\mathbf{u}+P-\sigma\\ e\mathbf{u}+P\mathbf{u}-\mathbf{u}\cdot\sigma\end{pmatrix} - \begin{pmatrix}0\\ 0\\ \rho g\\ \rho g\mathbf{u}\end{pmatrix}}_{F(y(t))} \underbrace{+\nabla\cdot\begin{pmatrix}0\\ \rho\kappa_c\nabla c\\ 0\\ K\nabla T\end{pmatrix}}_{G(y(t))}$$

- perhaps split off pressure terms (→ Kwatra's method) & viscosity, too
- solve this system numerically as follows:
  - Method of lines (MOL): spatial derivatives discretized by, e.g., a finite difference or finite volume approach such as the weighted essentially non-oscillatory (WENO) method, likewise for boundary conditions.
  - Result: a set of non-linear, coupled ordinary differential equations (ODEs).

Motivation III

Time steps in simulations of convection

– time step limits: $\Delta t = \min\{\Delta t_c, \Delta t_T, \Delta t_{\rm visc}, \Delta t_{\rm snd}, \Delta t_{\rm ad}\}$,

where

$$\Delta t_c = \frac{C_c}{\kappa_c} \min\{(\Delta x)^2, (\Delta y)^2, (\Delta z)^2\}, \quad \Delta t_T = \frac{C_T}{\kappa_T} \min\{(\Delta x)^2, (\Delta y)^2, (\Delta z)^2\},$$

$$\Delta t_{\rm visc} = \frac{C_{\rm visc}\,\rho}{\eta} \min\{(\Delta x)^2, (\Delta y)^2, (\Delta z)^2\}, \quad \Delta t_{\rm ad} = \frac{C_{\rm advec}}{\max(|\mathbf{u}|)} \min\{\Delta x, \Delta y, \Delta z\},$$

$$\Delta t_{\rm diff} = \min(\Delta t_c, \Delta t_T), \quad \Delta t_{\rm snd} \frac{C_{\rm snd}}{\max(|\mathbf{u}| - c_{\rm s}, |\mathbf{u}|\,, |\mathbf{u}| + c_{\rm s})} \min\{\Delta x, \Delta y, \Delta z\}$$

and

$$C_{\rm diff} = \min(C_{\rm c}, C_{\rm T}), \quad C_{\rm ad} = \min(C_{\rm advec}, C_{\rm snd})$$

Motivation IV

Astrophysical convection simulations

- typical simulations for M dwarf, idealised semiconvection (Pr=0.1), a Cepheid

$c_{ad} = c_{diff} = 0.2$	Δt_{ad}	Δt_{snd}	Δt_{diff}
Convection (M-Star)	7 s	0.52 s	383 s
Semiconvection	19.47 s	2.45 s	3.72 s
Cepheid	2.31 s	1.73 s	0.057 s

- M dwarf: time step limited by sound waves: low Mach number flow !
- Semiconvection: low Mach number flow, limited by heat diffusion until the layer becomes convective (transition diffusive → advective during simulation) !
- Cepheid: limited by radiative diffusion in layers near / at the surface (also in A-type stars, red giants, AGB stars, ...).

Motivation V

Why splitting in time instead of fully implicit ?

- in hydrodynamics: little gain from implicitly integrating the advection operator
  - less of a problem for viscous flows where viscous stress limits Δt
  - if, as often in low viscosity flows, Δt_{adv} limits Δt → solution changes on Δt
  - for order in time $p > 1$: strong stability (SSP) property is lost, if $\Delta t > 4\ \Delta t_{adv}$

    → Kraaijevanger: BIT 31, 482 (1991); Ferracina & Spijker: Math. Comp. 74, 201 (2004), Appl. Num. Math. 58, 1675 (2008); Ketcheson et al.: Appl. Num. Math. 59, 373 (2008), J. Sci. Comp. 38, 251 (2009)
- if the NSE are integrated in time with IMEX RK-methods, generalized (linear or quasilinear) Helmholtz equations are obtained (scalar, computationally cheaper)
- the same holds for the method of Kwatra et al. (2009), JCP 228, 4146 used for semi-implicit integration of terms containing ∇P
- these are (quasilinear) elliptic equations (even V-coercive and V-elliptic)
- fast, well-converging methods available: CG-type, multigrid methods

Kwatra's method I

Summary

- Following Kwatra et al. (2009), JCP 228, 4146 use Dp/Dt, equation of state, and conservation laws to derive a dynamical equation for $\partial P/\partial t$.
  - For $\partial P/\partial t$ discretize P in "diffusion-like" terms at t_{n+1}, evaluate other terms at t_n
  - and construct (linear system of) equation(s) $\Phi(P^{n+1})$ for P^{n+1}.
  - Additive pressure-term splitting as in incompressible methods → $(u_{n+1})^*$, ...
  - use fractional step results → evaluate $\Phi(P^{n+1})$ at $(P^{n+1})_{pred} = (P^{n+1})^*$.
  - Use $(P^{n+1})^*$ to complete time step → u_{n+1}, evaluate final P^{n+1} from EOS.
  - Repeat at each RK stage; for IMEX RK: P^{n+1} belongs to "explicit term".
- Buoyancy source term → "explicit term", e.g.
- while diffusion terms → right-hand side of generalized Helmholtz equations
- Method can be used in low and high Mach number regime,
- time step limit from advection and steep pressure gradients, not sound waves (!).

Kwatra’s method II

$$\frac{(\rho u)^{n+1} - (\rho u)^*}{\Delta t} = -\nabla P \tag{1}$$

$$\frac{e^{n+1} - e^*}{\Delta t} = -\nabla \cdot (Pu) \tag{2}$$

$$\nabla \cdot u^{n+1} = \nabla \cdot u^* - \Delta t \nabla \cdot \left(\frac{\nabla P}{\rho^{n+1}} \right) \tag{3}$$

$$\frac{\partial P}{\partial t} + u \cdot \nabla P = -(\nabla \cdot u)\rho c_\mathrm{s}^2 - \left(\frac{\partial P}{\partial \varepsilon} \right)_\rho \frac{1}{\rho^2} (u \cdot \nabla \tau - \nabla \cdot (u \cdot \tau) + \nabla \cdot (K \nabla T)) \tag{4}$$

$$\frac{\partial P}{\partial t} + u \cdot \nabla P = -(\nabla \cdot u)\rho c_s^2 - \frac{2}{3\rho} (u \cdot \nabla \tau - \nabla \cdot (u \cdot \tau) + \nabla \cdot (K \nabla T)) \tag{5}$$

$$\frac{\partial P}{\partial t} + u \cdot \nabla P = -\rho c_s^2 \nabla \cdot u^* + \rho c_s^2 \nabla \cdot \left(\frac{\nabla P}{\rho} \right) - \frac{2}{3\rho} (u \cdot \nabla \tau - \nabla \cdot (u \cdot \tau) + \nabla \cdot (K \nabla T)) \tag{6}$$

Kwatra et al. (2009), JCP 228, 4146 as extended by Happenhofer et al. (2013), JCP 236, 96: goal: compute intermediate $(P^{n+1})^*$ to integrate Eq. (1), then Eq. (2). Fractional step splitting → $\rho^*=\rho^{n+1}$, $c^*=c^{n+1}$, u^*, e^* → Eq. (1), (2) → Eq. (3). Evaluation of DP/Dt with EOS → Eq. (4). For a perfect gas EOS → Eq. (5). Use Eq. (3) to rewrite Eq. (5) → Eq. (6). (likewise for Eq. (4)).

Kwatra's method III

$$\tilde{c}P_{\text{pred}}^{n+1} - \nabla\cdot(\kappa\nabla P_{\text{pred}}^{n+1}) = f \tag{7}$$

$$\tilde{c} = \frac{1}{\Delta t^2 \rho^n c_s^2} \tag{8}$$

$$\kappa = \frac{1}{\rho^{n+1}} \tag{9}$$

$$\begin{aligned} f &= \frac{P^n - \Delta t\ u^n\cdot\nabla P^n}{\Delta t^2\rho^n(c_s^n)^2} - \frac{1}{\Delta t}(\nabla\cdot u^*) \\ &-\frac{2}{3\Delta t\ (\rho^n)^2(c_s^n)^2}(u^n\cdot\nabla\cdot\tau^n - \nabla\cdot(u^n\cdot\tau^n) + \nabla\cdot(K\nabla T^n)) \end{aligned} \tag{10}$$

$$\frac{\partial P}{\partial t} + u\cdot\nabla P = -(\nabla\cdot u)\rho c_{\text{s}}^2 - \frac{2}{3\rho}(u\cdot\nabla\tau - \nabla\cdot(u\cdot\tau) + \nabla\cdot(K\nabla T)) - \frac{3}{4}R_{\text{gas}}T\nabla\cdot(\rho\kappa_{\text{c}}\nabla c) \tag{11}$$

$$\begin{aligned} f &= \frac{P^n - \Delta t\ u^n\cdot\nabla P^n}{\Delta t^2\rho^n(c_{\text{s}}^n)^2} \\ &- \frac{1}{\Delta t}(\nabla\cdot u^*) - \frac{2}{3\Delta t(\rho^n)^2(c_{\text{s}}^n)^2}(u^n\cdot\nabla\cdot\tau^n - \nabla\cdot(u^n\cdot\tau^n) + \nabla\cdot(K\nabla T^n)) \\ &- \frac{3}{4\rho^n\Delta t(c_{\text{s}}^n)^2}R_{\text{gas}}T^n\nabla\cdot(\rho^n\kappa_{\text{c}}\nabla c^n) \end{aligned} \tag{12}$$

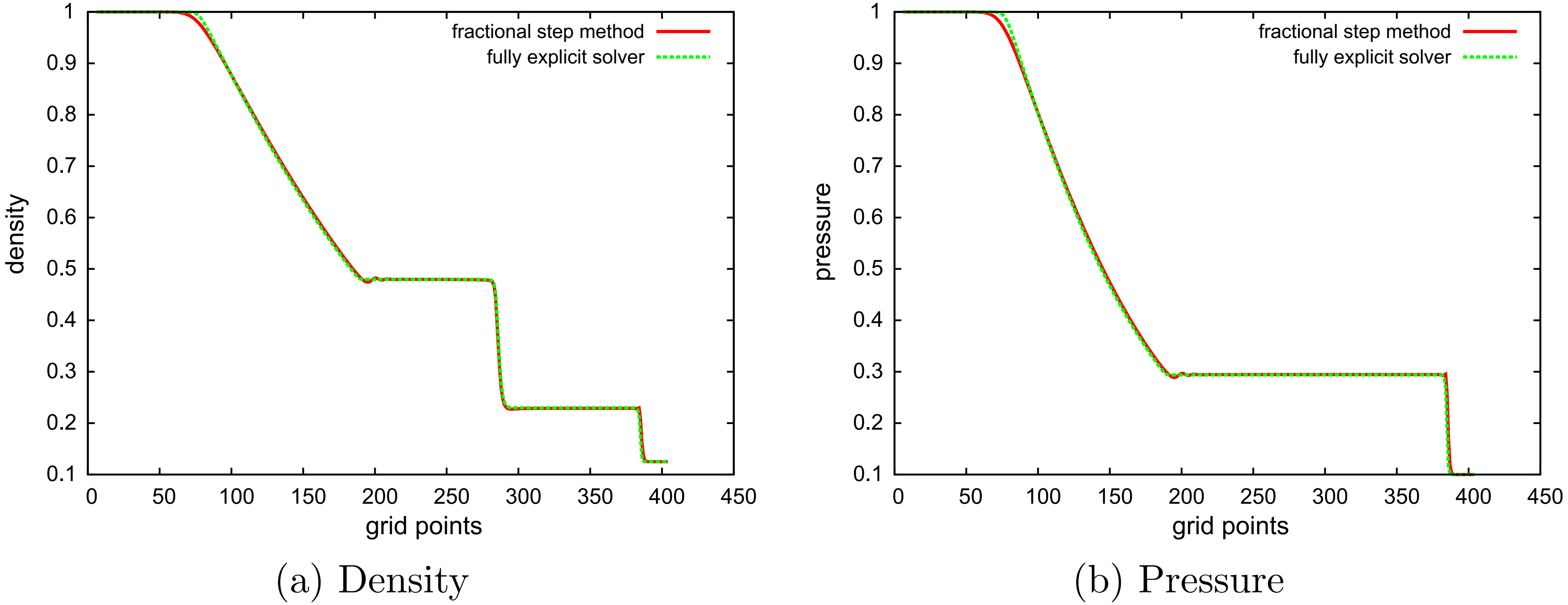
continue: forward difference for ∂P/∂t, P at t_{n+1} only in diffusion-like term ➔ Eq. (7) with Eq. (8), (9), and Eq. (10). Two-component fluid: Eq. (6) ⇔ Eq. (11), Eq. (10) ⇔ Eq. (12).

Kwatra's method IV

Remarks on suitable linear solvers

- The elliptic equation Eq. (7) with coefficients given by Eq. (8) and Eq. (9) and the right-hand side Eq. (10) or Eq. (12) is rather benign:
  - Meaningful physical solutions → $\check{c} > 0$, $\kappa > 0$: generalized Helmholtz equation
  - $-\Delta u + \lambda u = f$, $\lambda > 0$: (scalar, constant coefficient, linear) Helmholtz equation H
  - Eq. (7)-(10): scalar and linear PDE → shares many properties with H
- Discretization: bilinear (triangular shaped for pCG, rectangular for MG solver discussed later on) finite elements → symmetric, positive definite (SPD) matrix A for the linear system $A\,u = v$ (Grimm-Strele, MSc Thesis, 2010)
- Solve linear system by preconditioned conjugate gradient method (pCG) with incomplete Cholesky decomposition ($A = LL^T$) for preconditioning
- parallelization: use the Schur Complement decomposition
- the Schur matrix S_A is SPD because A is SPD → use the (plain) CG solver to solve the Schur complement equation $S_A\,u = v$; local systems: solved with pCG
- implemented into ANTARES (Muthsam et al. 2010, New Astronomy 15, 460)

Kwatra's method V



(a) Density

(b) Pressure

Fig. 8. Results of the Sod shocktube test at $t = 0.25$ s. Density and Pressure are plotted against an explicit reference solution. The results indicate well resolved shock waves.

1D Sod shocktube test with 405 grid points taken from Happenhofer et al. (2013), JCP 236, 96 confirming Kwatra et al. (2009), JCP 228, 4146

Kwatra's method VI

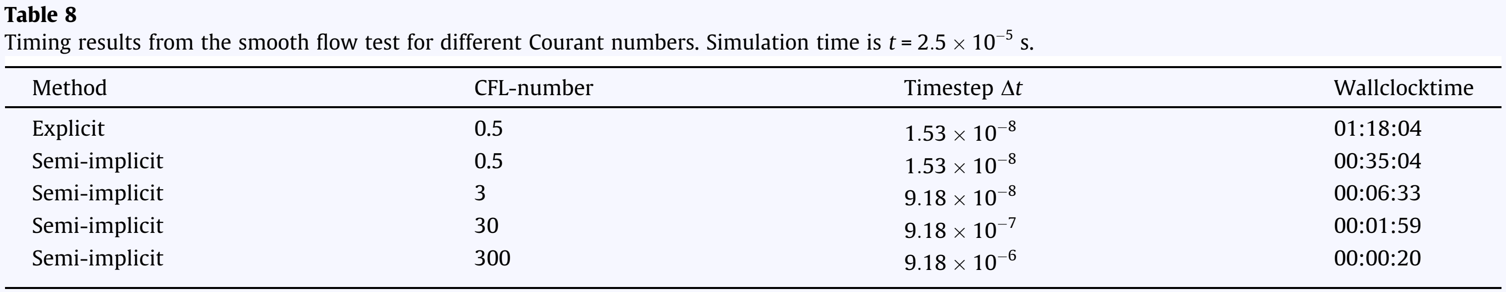


Table 8
Timing results from the smooth flow test for different Courant numbers. Simulation time is $t = 2.5 \times 10^{-5}$ s.

Method	CFL-number	Timestep Δt	Wallclocktime
Explicit	0.5	1.53×10^{-8}	01:18:04
Semi-implicit	0.5	1.53×10^{-8}	00:35:04
Semi-implicit	3	9.18×10^{-8}	00:06:33
Semi-implicit	30	9.18×10^{-7}	00:01:59
Semi-implicit	300	9.18×10^{-6}	00:00:20

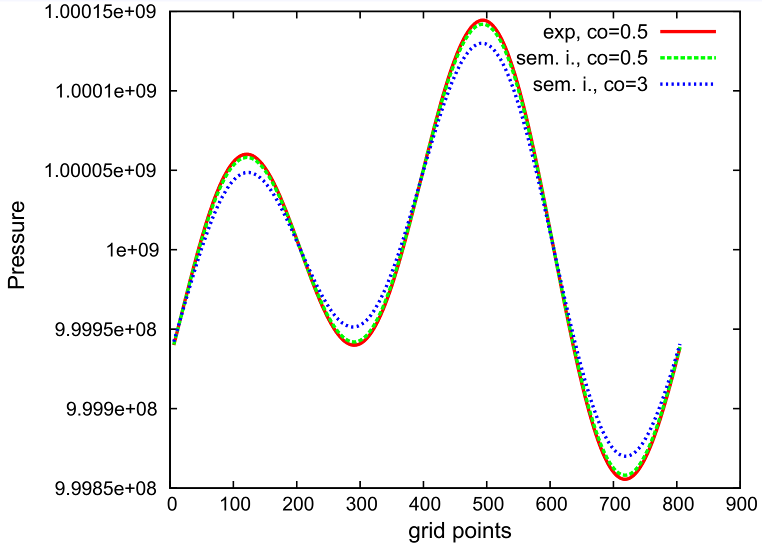
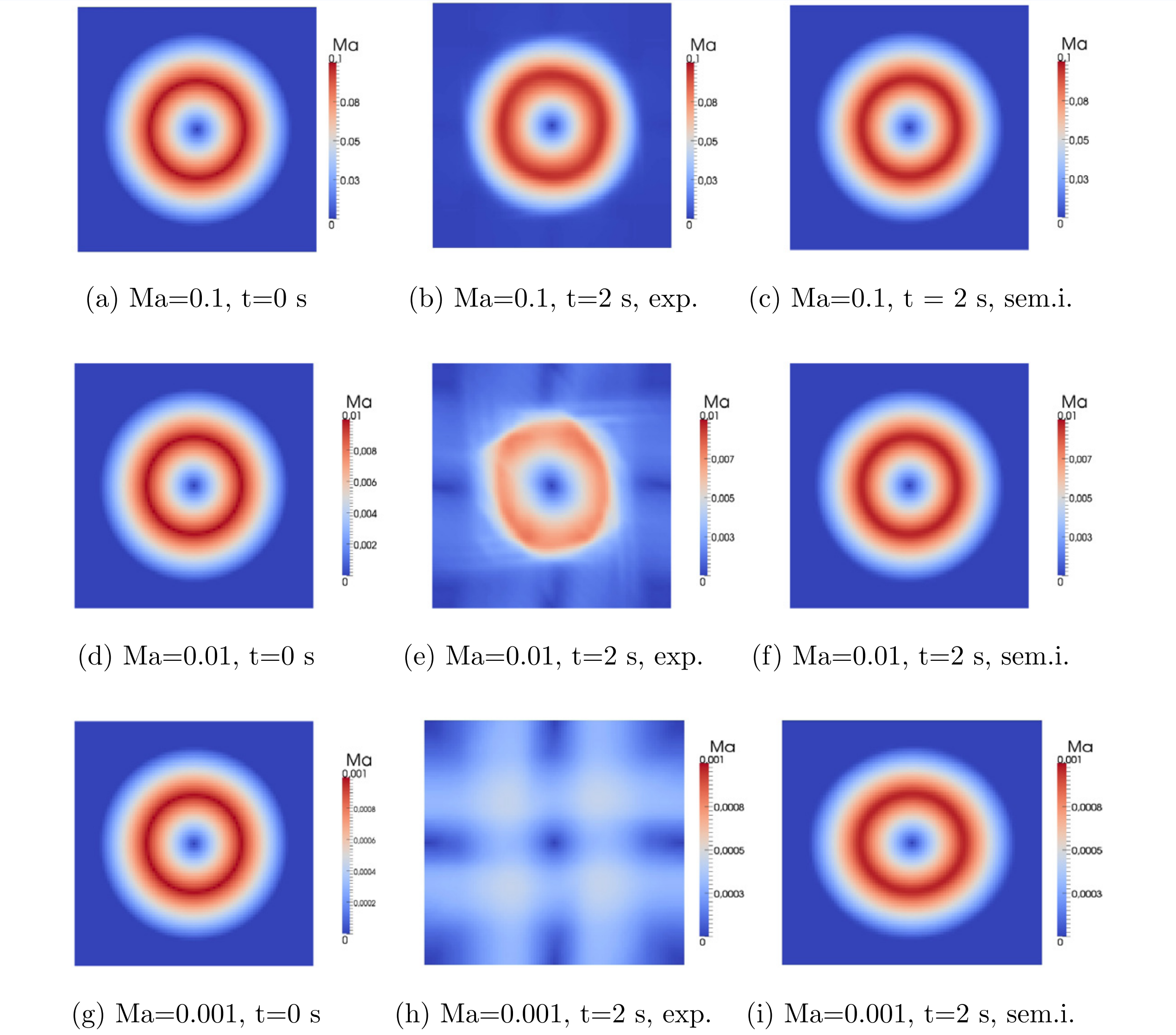


Fig. 10. Numerical results comparing the pressure at $t = 1.25 \times 10^{-5}$ s in the smooth flow test run with different CFL numbers.

1D smooth, low Mach number flow test test from Happenhofer et al. (2013), JCP 236, 96 confirming Kwatra et al. (2009), JCP 228, 4146

Kwatra’s method VII



(a) Ma=0.1, t=0 s (b) Ma=0.1, t=2 s, exp. (c) Ma=0.1, t = 2 s, sem.i.

(d) Ma=0.01, t=0 s (e) Ma=0.01, t=2 s, exp. (f) Ma=0.01, t=2 s, sem.i.

(g) Ma=0.001, t=0 s (h) Ma=0.001, t=2 s, exp. (i) Ma=0.001, t=2 s, sem.i.

Fig. 11. Comparison of the Gresho vortex test performed with the explicit and the semi-implicit solver with different Mach numbers. On the left, the initial configuration is depicted. In the center and on the right, the simulation is advanced 2 s with the explicit solver and the semi-implicit solver, respectively.

Time development of the Gresho vortex at three different Mach numbers (0.1, 0.01, and 0.001):

initial state (left), explicit time integration with TVD2 (centre) and Kwatra’s method (right) after t=2 s.

From Happenhofer et al. (2013), JCP 236, 96.

Kwatra’s method VIII

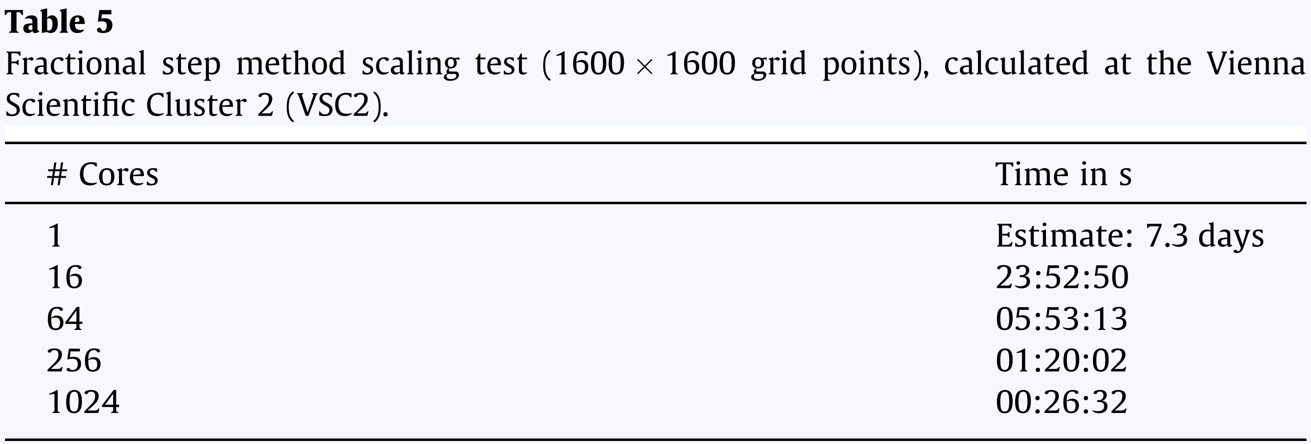


Table 5
Fractional step method scaling test (1600 × 1600 grid points), calculated at the Vienna Scientific Cluster 2 (VSC2).

# Cores	Time in s
1	Estimate: 7.3 days
16	23:52:50
64	05:53:13
256	01:20:02
1024	00:26:32

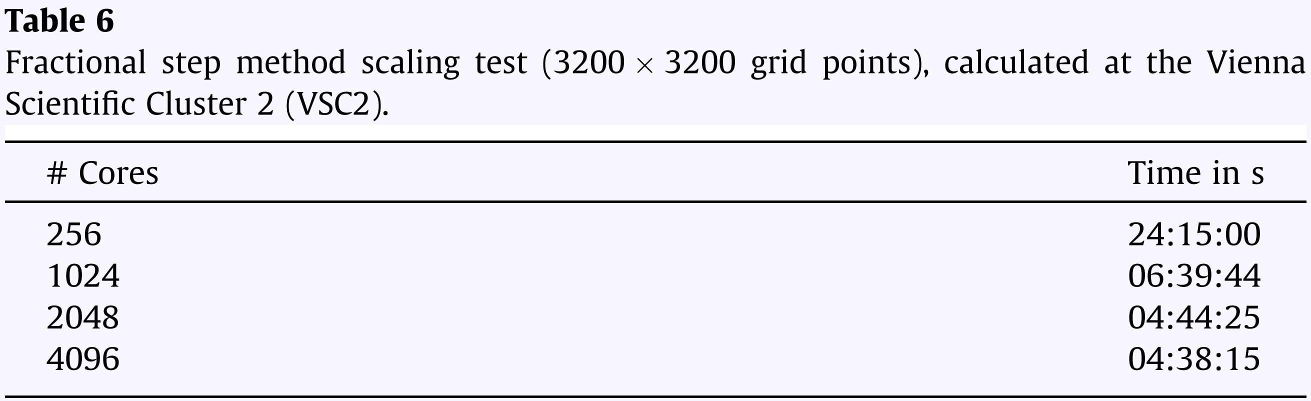


Table 6
Fractional step method scaling test (3200 × 3200 grid points), calculated at the Vienna Scientific Cluster 2 (VSC2).

# Cores	Time in s
256	24:15:00
1024	06:39:44
2048	04:44:25
4096	04:38:15

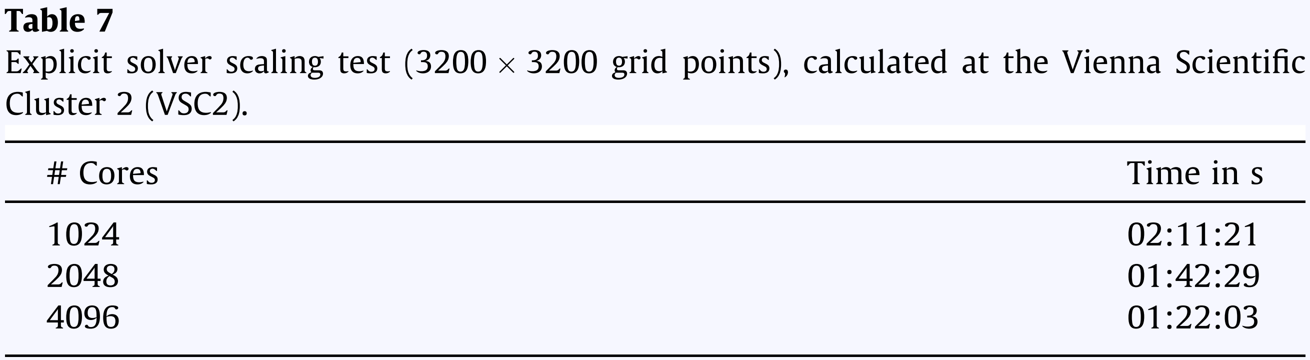


Table 7
Explicit solver scaling test (3200 × 3200 grid points), calculated at the Vienna Scientific Cluster 2 (VSC2).

# Cores	Time in s
1024	02:11:21
2048	01:42:29
4096	01:22:03

scaling test of simulation of fully compressible convection (Ra=1.6 × 10^6, Pr = 0.1, run over 0.5 sound crossing times), from Happenhofer et al. (2013), JCP 236, 96

further tests passed: convergence rate, 2D circular shock (from Kwatra et al.), 2D Gresho vortex test (new test, for Ma = 0.1 to 0.001)

IMEX RK-Methods I

Definition of IMEX RK-Methods

Consider the ODE initial value problem

$$\dot{y}(t) = F(y(t)) + G(y(t)), \qquad y(0) = y_0, \tag{1}$$

where the vector fields F and G have different stiffness properties.

An s-stage Runge–Kutta method characterized by coefficient matrices $A = (a_{i,j})$ and $\tilde{A} = (\tilde{a}_{i,j})$ defines one step $y_{\text{old}} \to y_{\text{new}}$ by

$$y_i = y_{\text{old}} + \Delta t \sum_{j=1}^{s} a_{i,j} F(y_j) + \Delta t \sum_{j=1}^{s} \tilde{a}_{i,j} G(y_j), \quad i = 1, \ldots, s, \tag{2}$$

$$y_{\text{new}} = y_{\text{old}} + \Delta t \sum_{j=1}^{s} b_j F(y_j) + \Delta t \sum_{j=1}^{s} \tilde{b}_j G(y_j). \tag{3}$$

If $a_{i,j} = 0$ for $j \geq i$, the method is called an *implicit–explicit (IMEX)* method.

(following Kupka et al. (2012), JCP 231, 3561)

IMEX RK-Methods II

Strong Stability Preserving (SSP) property:

- monotonicity: $\|u(t)\| \leqslant \|u(t_0)\|$ $\forall\ t \geqslant t_0$ with $\|.\|$ some (semi-)norm
- contractivity: $\|u(t)-v(t)\| \leqslant \|u(t_0)-v(t_0)\|$ $\forall\ t \geqslant t_0$
- boundedness: $m \leqslant u(t) \leqslant M$, if $m \leqslant u(t_0) \leqslant M$ $\forall\ t \geqslant t_0$
  (special cases: $m = 0$ → positivity, likewise: $\|u(t)\| \leqslant M\ \forall\ t \geqslant t_0$)
- monotonicity u(t) with respect to v(t): $u(t) \leqslant v(t)$, if $u(t_0) \leqslant v(t_0)$ $\forall\ t \geqslant t_0$

Preserve these using SSP schemes for ODEs:

- if the exact solution is monotonic → also require the same from RK scheme !
- numerical monotonicity: strong stability means $\|U_{ni}\| \leqslant \|u_n\|$ for any stage i and internal stage solution U_{ni} computed at time t_n and also $\|u_{n+1}\| \leqslant \|u_n\|$
  → radius of absolute monotonicity (Kraaijevanger 1991) → restricts step size
- reformulate RK schemes (Shu & Osher 1988) → if for a spatial discretization the Euler forward scheme is strongly stable, then, with step size restrictions, RK schemes are strongly stable, too which have positive coefficients α_{ik}, β_{ik}

IMEX RK-Methods III

Pareschi & Russo (2005) give an IMEX SSP2(2,2,2) method with nontrivial region of absolute monotonicity ($\gamma = 1 - \frac{1}{\sqrt{2}}$):

$$
\begin{array}{c|cc}
0 & 0 & 0 \\
1 & 1 & 0 \\
\hline
A & \frac{1}{2} & \frac{1}{2}
\end{array}
\qquad\qquad
\begin{array}{c|cc}
\gamma & \gamma & 0 \\
1-\gamma & 1-2\gamma & \gamma \\
\hline
\tilde{A} & \frac{1}{2} & \frac{1}{2}
\end{array}\,. \tag{6}
$$

The coefficients imply $\mathcal{R}(A) = 1,\ \ \mathcal{R}(\tilde{A}) = 1 + \sqrt{2}$, and

$$\mathcal{R}(A, \tilde{A}) = \{(r, \tilde{r}) : 0 \leq r \leq 1,\ 0 \leq \tilde{r} \leq \sqrt{2}(1 - r)\},$$

see Higueras (2006).

(from Kupka et al. (2012), JCP 231, 3561)

IMEX RK-Methods IV

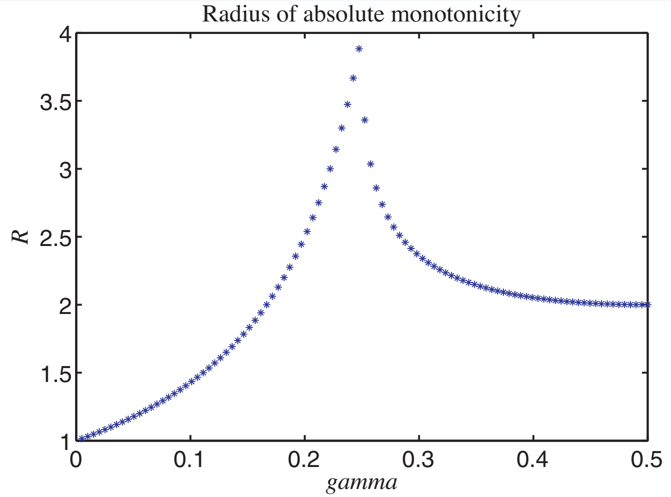


Fig. 2. Radius of absolute monotonicity $\mathcal{R}(\tilde{A})$ as a function of γ for (29).

stability properties as a function of γ for the 1D constant coefficient heat equation

Kupka et al. (2012), JCP 231, 3561

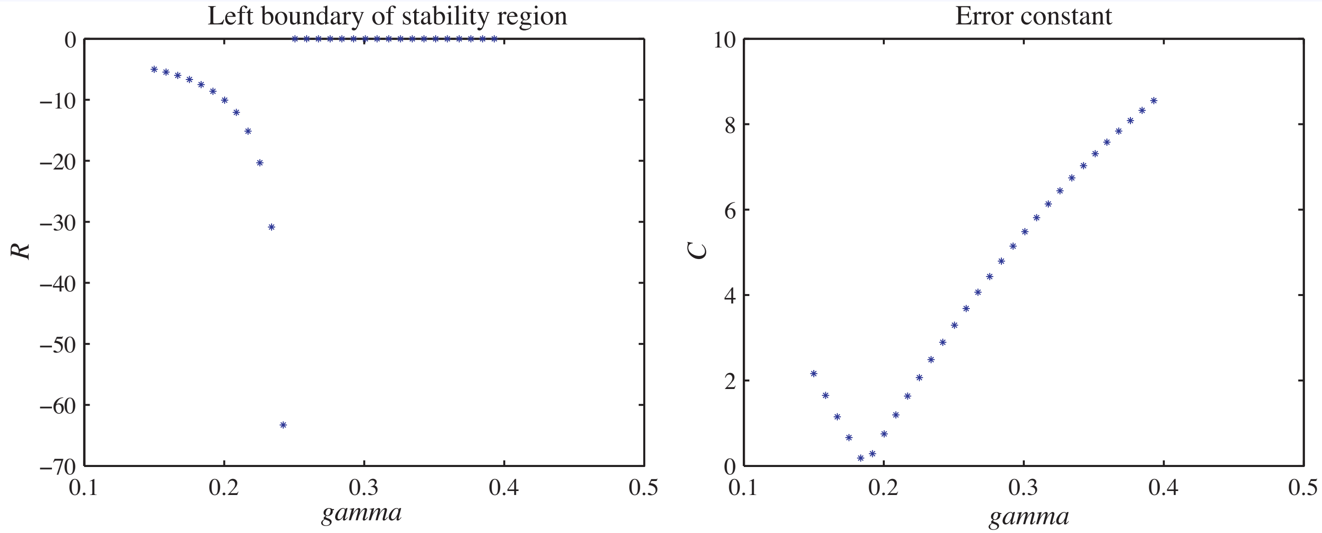


Fig. 4. Left boundary of the stability region (left) and error constant computed for (7) (right) as a function of γ for (29).

IMEX RK-Methods V

Higueras (2006) gives an IMEX SSP2(3,3,2) method with nontrivial region of absolute monotonicity:

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} & 0 \\ \hline A & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array} \qquad \begin{array}{c|ccc} \frac{1}{5} & \frac{1}{5} & 0 & 0 \\ \frac{3}{10} & \frac{1}{10} & \frac{1}{5} & 0 \\ 1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \hline \tilde{A} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array} \tag{8}$$

This is a modification of a scheme from Pareschi & Russo (2005), where the latter turned out to have a trivial region of absolute monotonicity. It holds that $\mathcal{R}(A) = 2$ and $R(\tilde{A}) = \frac{5}{9}(\sqrt{70} - 4)$, and

$$\mathcal{R}(A, \tilde{A}) = \{(r, \tilde{r}) : 0 \leq r \leq 1,\ 0 \leq \tilde{r} \leq \phi(r)\},$$

where

$$\phi(r) = \left\{\frac{1}{4}(-28 + 9r) + \frac{1}{4}\sqrt{1264 - 984r + 201r^2}\right\}.$$

We note that the latter is a correction with respect to Higueras (2006), since we have found r to be necessarily bound by 1 in $\mathcal{R}(A, \tilde{A})$.

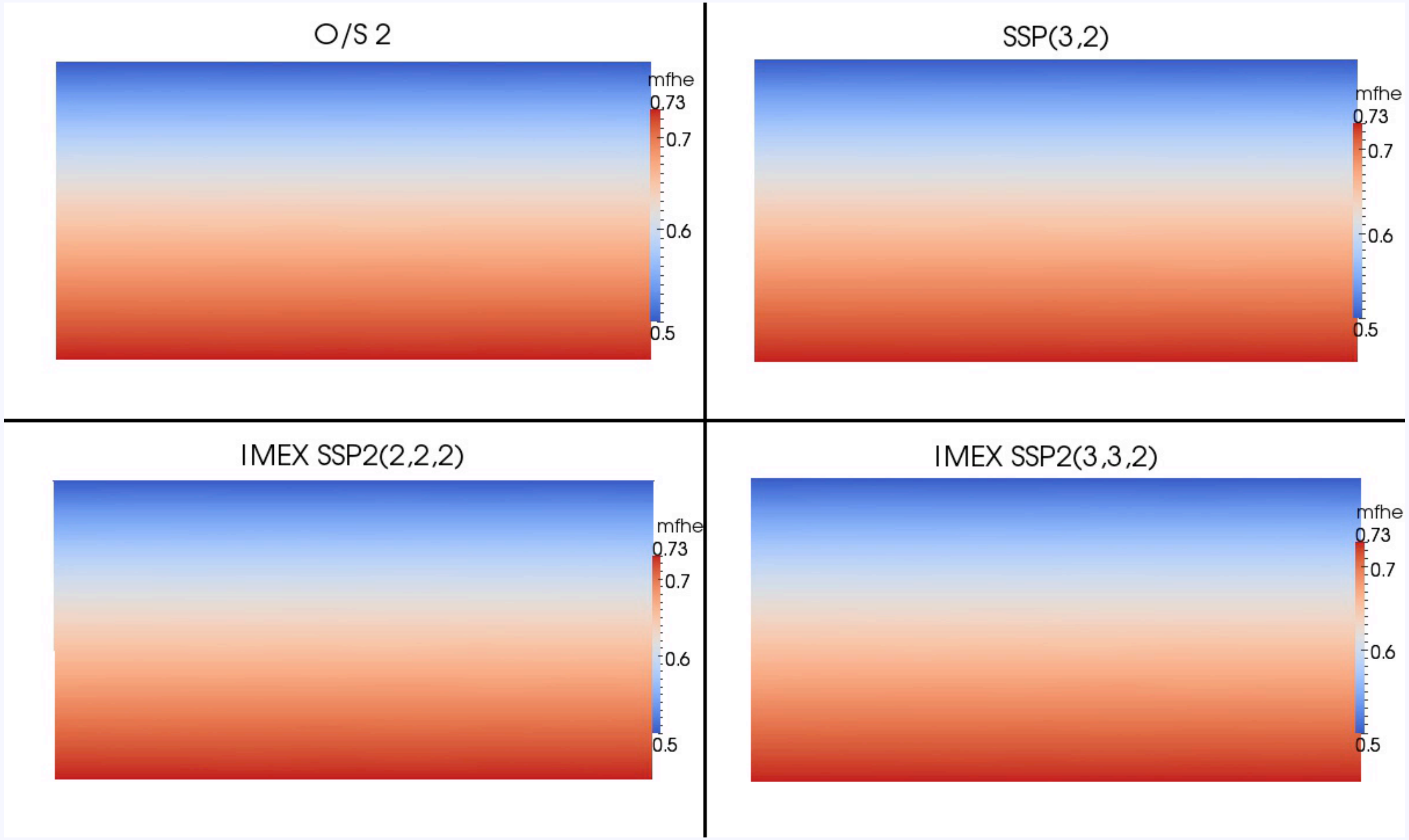
(from Kupka et al. (2012), JCP 231, 3561)

Applications: Semiconvection I

Simulations of Double-Diffusive Convection

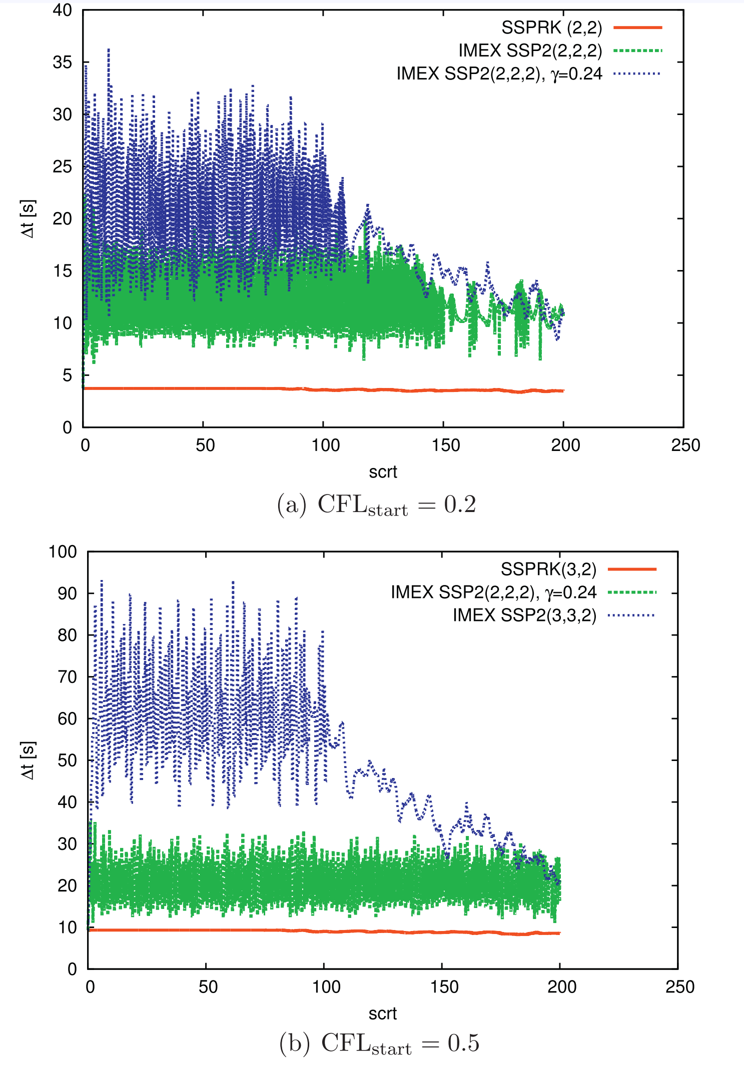
- simulation of semiconvection (based on two-species NSE)
  - 400 × 400 points, Cartesian grid, horizontally periodic, DNS
  - compressible, Kwatra's method to deal with ∇P terms
  - time-steps: 24000 to 300000; ~ 6 hours @ 64 MPI-processes; 3 GB data per run
- using the ANTARES code (Muthsam et al. (2010), New Astronomy 15, 460)
  - default method: conservative 5th order WENO (also with Marquina flux splitting)
  - parabolic terms consistent with WENO (Happenhofer et al. (2013), JCP 236, 96)
  - 1D, 2D, 3D and various grids (Cartesian, co-moving polar, curvi-linear in prep.)
  - optional: grid refinement, DNS/iLES/LES, radiative transfer, MHD mode
  - various open and closed boundary conditions, one- and two-component fluids
  - flexible microphysics modules, compressible or Boussinesq approximation
  - hybrid parallelization (MPI, optionally OpenMP), modular Fortran95, parallel I/O

Applications: Semiconvection II



Semiconvection in a compressible layer: explicit vs. semi-implicit time integration. Computations: N. Happenhofer @ VSC1 (from Kupka et al. 2012, JCP 231, 3561).

Applications: Semiconvection III



(a) $\mathrm{CFL}_{\mathrm{start}} = 0.2$

(b) $\mathrm{CFL}_{\mathrm{start}} = 0.5$

Time step size as a function of scrt for the case Pr = 0.1

from Kupka et al. 2012, JCP 231, 3561

New SSP IMEX RK-Methods I

Is the SSP property really needed for such flows?

- semiconvection simulations (400 × 400 points, Kupka et al. 2012, JCP 231, 3561)
  - Kennedy & Carpenter (2003), Appl. Num. Math. 44, 139: ARK3(2)4L[2]SA method
    - fails at CFL$_{\text{start}}$ used for IMEX-RK SSP schemes (after 78 time steps)
    - even after 10 scrt: never permits to exceed CFL values of explicit schemes !
    - designed for such systems (L-stable, stiffly accurate) — but non-SSP
  - original Pareschi & Russo, J. Sci. Comp. 25, 129 (2005)) SSP2(3,3,2) scheme
    - at original resolution works (SSP in an asymptotic sense: stiffness ➜ infinity)
    - at 100 x 100 points resolution: works during the diffusive phase, fails during the advective phase (Δt ➜ 0 at 113 scrt), no problem for true SSP schemes
- case of solar surface convection (2D model with 219 × 159 points, ibidem)
  - the explicit, non-SSP 3rd order RK method by Heun: found to fail after 9.2 scrt while the SSPRK(3,3) method by Fehlberg (1970) (= TVD3) works for ⩾ 20 scrt
- Cylindrical MHD explosion test (Aloy et al., at ASTRONUM 2013):
  - only modified SSP2(3,3,2) scheme (from Higueras 2006) works with MCL limiter
  - 3 to 4 time larger time step than original SSP2(3,3,2) scheme in shock-tube test

New SSP IMEX RK-Methods II

Motivations for further improvements

- Degrees of freedom of RK scheme can be used to fulfil many properties simultaneously instead of just maximizing the region of absolute monotonicity !
- Suggestion (Higueras et al. 2013, see also their ASC Report 14/2012):

1. p=2: overall scheme second order in time with small error constant C (if accuracy is limited by spatial resolution, p=2 with small C sufficient → p=3 not that important).
2. SSP: IMEX scheme should be SSP and have a large region of absolute monotonicity (Higueras 2006, 2009) → both the explicit and the implicit schemes must be SSP. The Kraaijevanger radius of each of them should be large, too.
3. Ensured by L-stability: the stability function of the implicit scheme should tend to zero at infinity, its stability region containing a large subinterval of the negative real axis.
4. Positivity: for both schemes, the stability function g should be nonnegative for a large interval of the negative real axis (related to step-size restrictions due to dissipativity). This prevents spurious oscillations of the numerical solution.

Properties 3 & 4 are guided by exact solutions of the heat equation !

New SSP IMEX RK-Methods III

- Further design goals of optimization:

5. Ensure uniform convergence (to guarantee higher than first order convergence for arbitrarily stiff terms, see Boscarino (2008), SIAM J. Num. Analysis 45, 1600).
6. For the explicit scheme, the stability region should a) contain large subintervals of the negative real axis, $[-z, 0]$ with $z > 0$, and b) also of the imaginary axis, $[-w\,i,\ w\,i]$ with $w > 0$. The latter requirement is associated with a stable integration of the hyperbolic advection terms (see Motamed et al. 2009, Wang & Spiteri (2007)).
7. The region of absolute stability of the combined IMEX scheme should be large.
8. For a convenient and memory-efficient implementation, the coefficients of the scheme should be rational numbers which recombine the stages in a suitable way.

- Perhaps optimize only the coefficients of the implicit scheme ?

9. Use optimal second order three stage method by Kraaijevanger (1991), the SSPRK(3,2) method, as explicit method and abandon requirement 6b) from above.

- Possibly consider additional optimizations, if sufficient degrees of freedom are still available ...

New SSP IMEX RK-Methods IV

We give the coefficients of the IMEX scheme obtained from our earlier considerations, that is

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{5}{6} & \frac{5}{6} & 0 & 0 \\ \frac{11}{12} & \frac{11}{24} & \frac{11}{24} & 0 \\ \hline \mathcal{A} & \frac{24}{55} & \frac{1}{5} & \frac{4}{11} \end{array} \qquad \begin{array}{c|ccc} \frac{2}{11} & \frac{2}{11} & 0 & 0 \\ \frac{289}{462} & \frac{205}{462} & \frac{2}{11} & 0 \\ \frac{751}{924} & \frac{2033}{4620} & \frac{21}{110} & \frac{2}{11} \\ \hline \tilde{\mathcal{A}} & \frac{24}{55} & \frac{1}{5} & \frac{4}{11} \end{array} \qquad (22)$$

and summarize its properties.

This is a second order IMEX scheme such that the implicit method is L–stable. The stability functions for the explicit and implicit schemes are

$$R_{\mathbb{A}}(z) = 1 + z + \frac{z^2}{2} + \frac{5}{36}\,z^3\,, \qquad R_{\tilde{\mathbb{A}}}(z) = \frac{11\left(13\,z^2 + 110\,z + 242\right)}{2\,(11 - 2\,z)^3}\,. \qquad (23)$$

(see Higueras et al. (2013), submitted, and their ASC Report 14/2012)

New SSP IMEX RK-Methods V

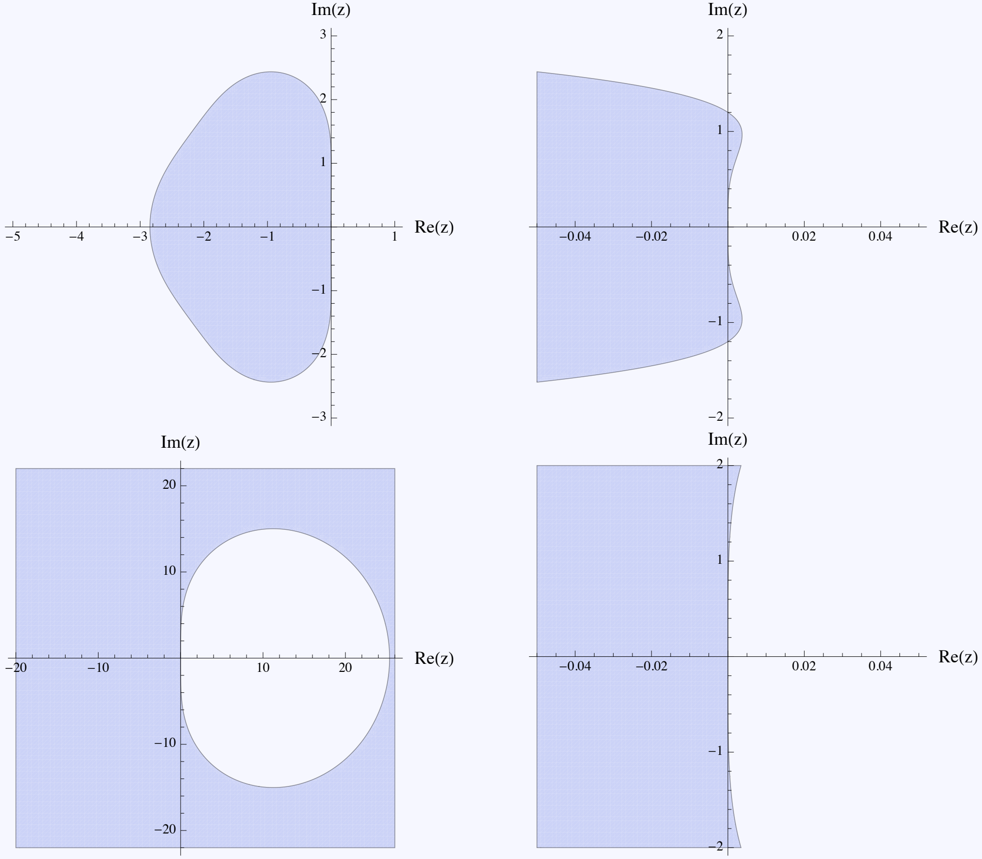


FIG. 3.5. *Stability region and a zoom of the stability region of the explicit scheme in* (3.30) *(top) and the implicit scheme in* (3.30) *(bottom).*

(see Higueras et al. (2013), submitted, and their ASC Report 14/2012)

New SSP IMEX RK-Methods VI

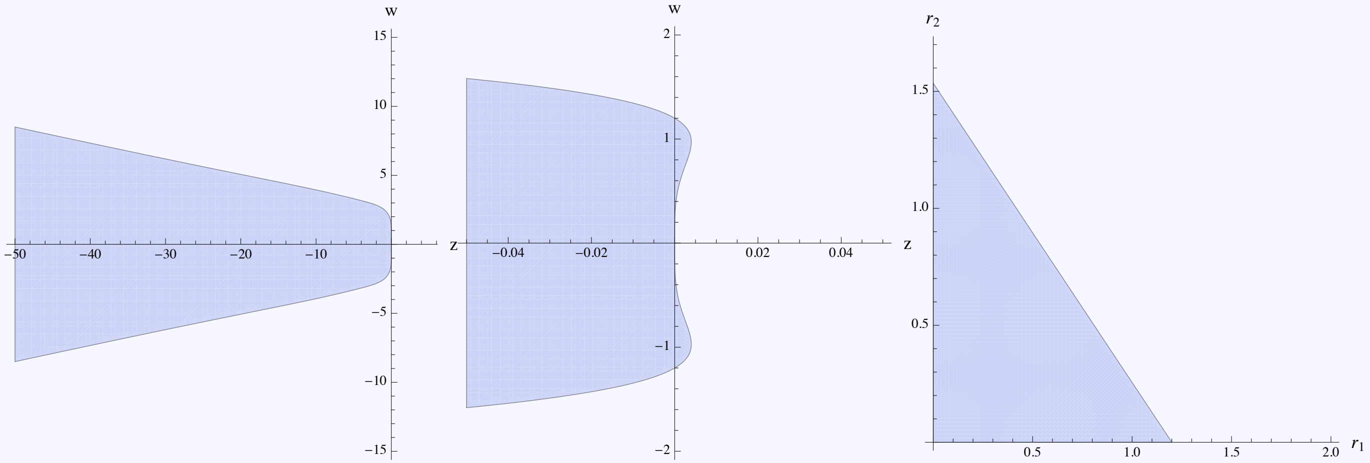


FIG. 3.6. *For IMEX* (3.30)*: stability region (right), a zoom of the region at the origin (center) and region of absolute monotonicity (right).*

Due to the properties listed above, we will refer to to this scheme as SSP2(3,3,2)–LSPUM, where the letters have the following meanings:

’L’: L–stable;
’S’: the stability region for the explicit part contains an interval
on the imaginary axis;
’P’: the amplification factor g is always positive;
’U’: the IMEX method features uniform convergence (21);
’M’: the IMEX method has a nontrivial region of absolute monotonicity.
(24)

(see Higueras et al. (2013), submitted, and their ASC Report 14/2012)

New SSP IMEX RK-Methods VII

IMEX method based on the explicit SSP(3,2) scheme with uniform convergence: in this case, the IMEX scheme obtained by imposing (21) is given by the coefficient tableaux

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} & 0 \\ \hline \mathcal{A} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array} \qquad \begin{array}{c|ccc} \frac{2}{11} & \frac{2}{11} & 0 & 0 \\ \frac{69}{154} & \frac{41}{154} & \frac{2}{11} & 0 \\ \frac{67}{77} & \frac{289}{847} & \frac{42}{121} & \frac{2}{11} \\ \hline \tilde{\mathcal{A}} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{array} \tag{25}$$

Due to its properties, we refer to method as SSP2(3,3,2)–LPUM.

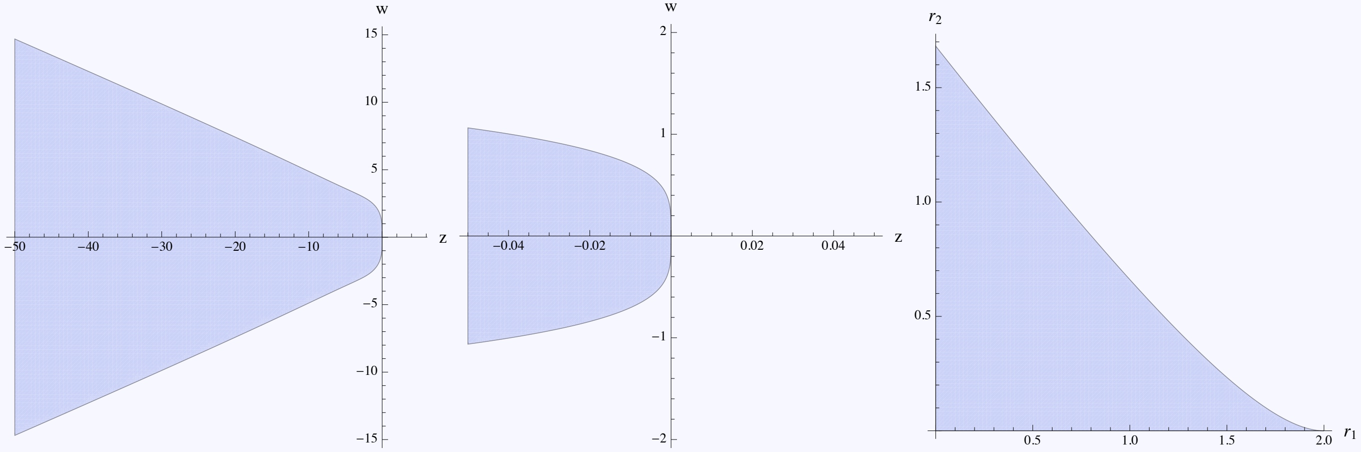


FIG. 4.1. *For the IMEX method (4.2): Left: Stability region; center: zoom of the stability region; right: region of absolute monotonicity.*

(see Higueras et al. (2013), submitted, and their ASC Report 14/2012)

Results I

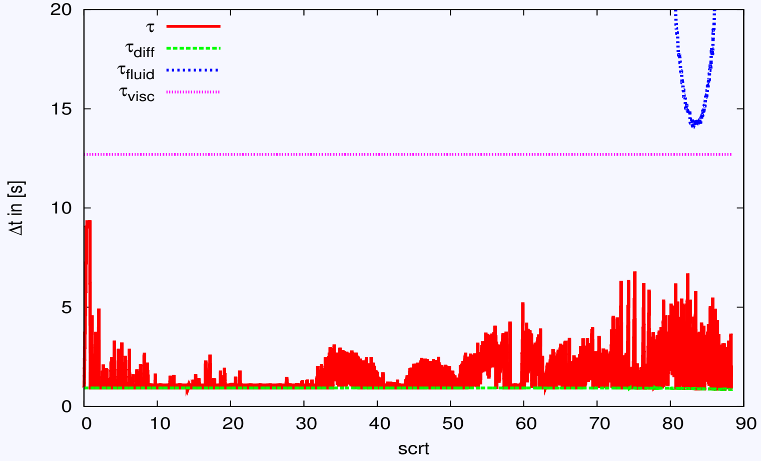


FIG. 5.1. *Simulation A: Time–step evolution with time integrator IMEX SSP1(1,1,1)–LPM.*

case with Pr = 0.1

failure of Euler forward-backward scheme, i.e. IMEX SSP1(1,1,1)-LPM.

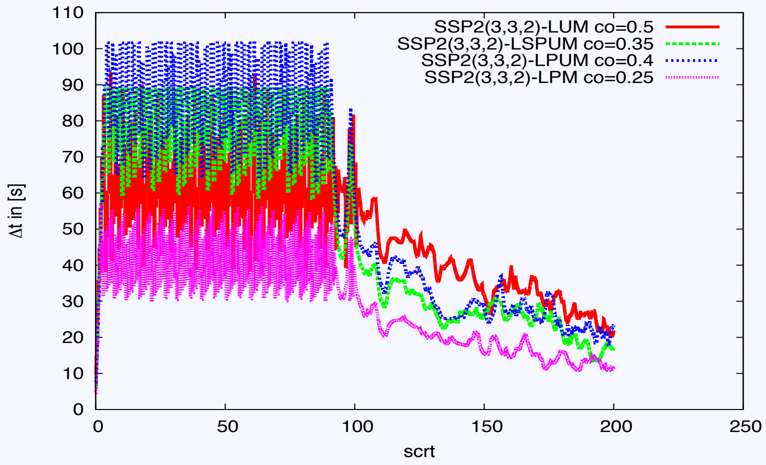


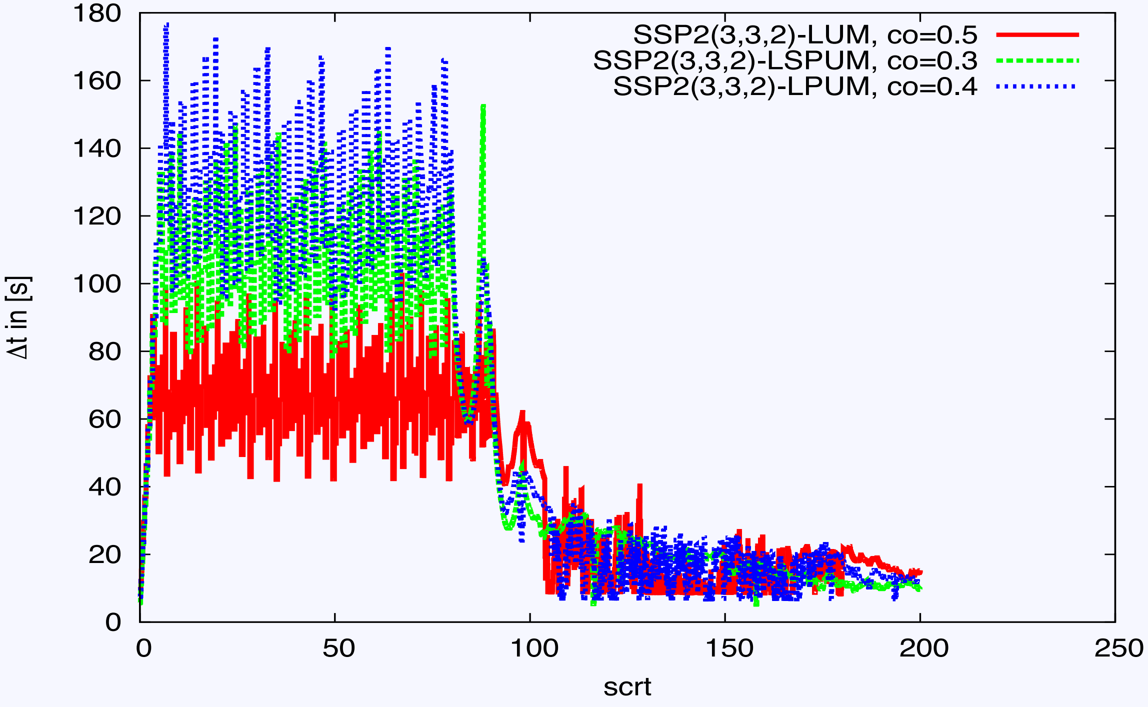
FIG. 5.2. *Simulation A: Time–step evolution over the entire 200 scrt.*

case with Pr = 0.1

LUM-scheme is previously best method: modified SSP2(3,3,2)

(see Higueras et al. (2013), submitted, and their ASC Report 14/2012)

Results II



case with Pr = 0.05

(based on pre-cond. CG solver and Schur complement, see Happenhofer et al. (2013), JCP 236, 96)

(data from Higueras et al., ASC Report 14/2012)

FIG. 5.3. *Simulation B: Time–step evolution over the entire 200 scrt.*

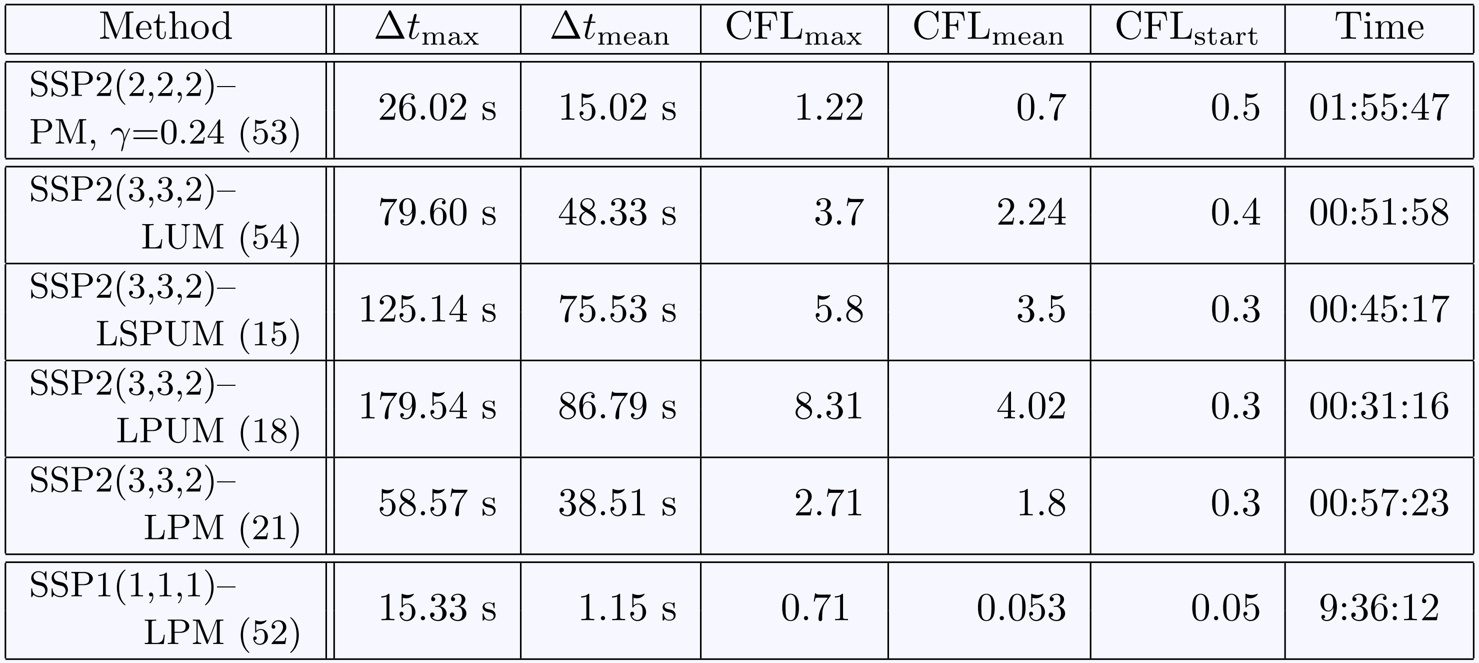
SSP2(3,3,2)–LSPUM (3.30)		
Resolution	$\mathrm{CFL}_{\mathrm{mn}}$	$\mathrm{CFL}_{\mathrm{max}}$
200x200	1.27	4.55
400x400	4.90	7.88
800x800	5.00	8.56
1600x1600	5.09	8.23

SSP2(3,3,2)–LPUM (4.2)		
Resolution	$\mathrm{CFL}_{\mathrm{mn}}$	$\mathrm{CFL}_{\mathrm{max}}$
200x200	1.27	4.55
400x400	5.73	9.48
800x800	5.71	10.27
1600x1600	6.47	10.19

TABLE 5.3

Resolution Test for Simulation B. Left: SSP2(3,3,2)–LSPUM method (3.30). *Right: SSP2(3,3,2)–LPUM method* (4.2).

Results III



Method	$\Delta t_{\max}$	Δt_{mean}	$\text{CFL}_{\max}$	CFL_{mean}	$\text{CFL}_{\text{start}}$	Time
SSP2(2,2,2)–PM, γ=0.24 (53)	26.02 s	15.02 s	1.22	0.7	0.5	01:55:47
SSP2(3,3,2)–LUM (54)	79.60 s	48.33 s	3.7	2.24	0.4	00:51:58
SSP2(3,3,2)–LSPUM (15)	125.14 s	75.53 s	5.8	3.5	0.3	00:45:17
SSP2(3,3,2)–LPUM (18)	179.54 s	86.79 s	8.31	4.02	0.3	00:31:16
SSP2(3,3,2)–LPM (21)	58.57 s	38.51 s	2.71	1.8	0.3	00:57:23
SSP1(1,1,1)–LPM (52)	15.33 s	1.15 s	0.71	0.053	0.05	9:36:12

Table 2: Simulation of double-diffusive convection: Time–steps, CFL–numbers and wallclock-times over the first 80 scrt.

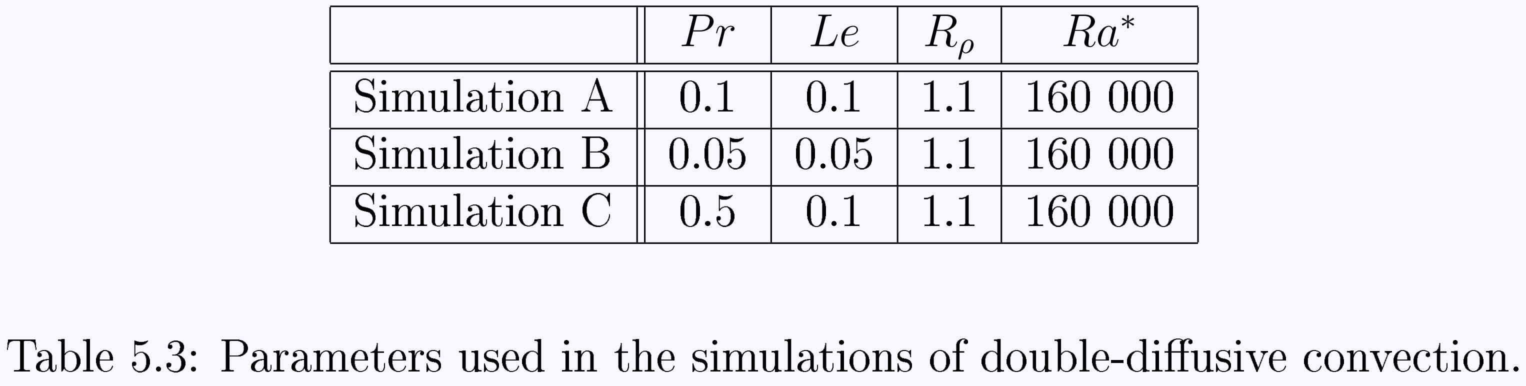
again a case with Pr = Le = 0.05, 429 × 428 points, VSC-1 @ 64 CPUs but this time based on parallel multigrid solver (see Higueras et al. (2013), submitted)

Results IV

Comments on the multigrid method used

- Typically three levels with a coarsening factor of 2 (GMG, geometric multigrid, rather than AMG, the algebraic multigrid approach)
- Use GCA (Galerkin coarse grid approximation): $A_{l+1} = r\, A_l\, p$, since we also want to use polar coordinate grids (where DCA, discrete coarse grid approximation, is highly inefficient)
- V-cycling (λ=1) turned out to be more efficient than alternatives (W-, F-)
- Intergrid operators (restriction, prolongation) canonically defined within FEM discretization
- Smoothing at grid level: GS-RB (red-black-Gauss-Seidel) and ZGSx (Zebra x-line Gauss-Seidel) proved most efficient for the problems considered (CG worked, too, but only for simpler cases)
- CG used to solve at coarsest level (typical errors: $\epsilon_{\mathrm{coarse}} \lesssim 0.1\ \epsilon_{\mathrm{fine}} = 10^{-7}$
- different options for parallelization (Happenhofer (2013), PhD thesis, in prep.)

Results V



	Pr	Le	R_ρ	Ra^*
Simulation A	0.1	0.1	1.1	160 000
Simulation B	0.05	0.05	1.1	160 000
Simulation C	0.5	0.1	1.1	160 000

Table 5.3: Parameters used in the simulations of double-diffusive convection.

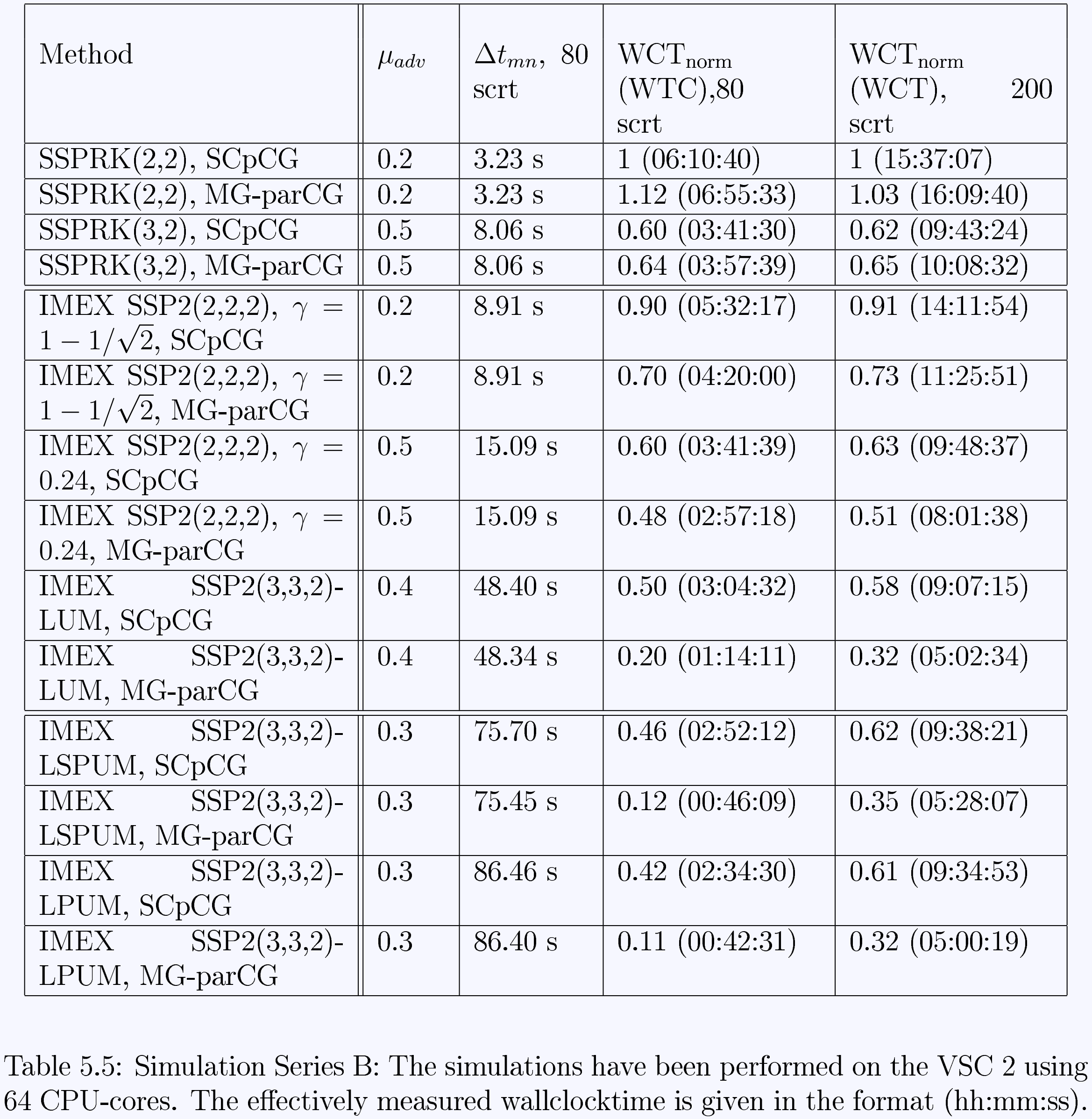
The efficiency of (parallel) multigrid solvers for generalized Helmholtz equations on regular meshes is demonstrated in a comparison from Happenhofer (2013) (PhD thesis, in prep.).

Heat and concentration diffusion limit the timestep more strongly in Simulation B (Pr, Le).

In that case, the MG-parCG method (multigrid method with directly parallelized CG method for the coarsest grid) converges much faster than our fastest, parallelized Krylov type solver, the SCpCG method (the conjugate gradient method pre-conditioned by incomplete Cholesky decomposition and parallelized by means of the Schur complement).

If the diffusion terms are all integrated in time by an explicit method, convergence of SCpCG is fast enough and its overhead small enough, so SCpCG remains slightly more efficient.

Results VI



Method	μ_{adv}	Δt_{mn}, 80 scrt	WCT$_{\text{norm}}$ (WTC),80 scrt	WCT$_{\text{norm}}$ (WCT), 200 scrt
SSPRK(2,2), SCpCG	0.2	3.23 s	1 (06:10:40)	1 (15:37:07)
SSPRK(2,2), MG-parCG	0.2	3.23 s	1.12 (06:55:33)	1.03 (16:09:40)
SSPRK(3,2), SCpCG	0.5	8.06 s	0.60 (03:41:30)	0.62 (09:43:24)
SSPRK(3,2), MG-parCG	0.5	8.06 s	0.64 (03:57:39)	0.65 (10:08:32)
IMEX SSP2(2,2,2), $\gamma = 1 - 1/\sqrt{2}$, SCpCG	0.2	8.91 s	0.90 (05:32:17)	0.91 (14:11:54)
IMEX SSP2(2,2,2), $\gamma = 1 - 1/\sqrt{2}$, MG-parCG	0.2	8.91 s	0.70 (04:20:00)	0.73 (11:25:51)
IMEX SSP2(2,2,2), $\gamma =$ 0.24, SCpCG	0.5	15.09 s	0.60 (03:41:39)	0.63 (09:48:37)
IMEX SSP2(2,2,2), $\gamma =$ 0.24, MG-parCG	0.5	15.09 s	0.48 (02:57:18)	0.51 (08:01:38)
IMEX SSP2(3,3,2)-LUM, SCpCG	0.4	48.40 s	0.50 (03:04:32)	0.58 (09:07:15)
IMEX SSP2(3,3,2)-LUM, MG-parCG	0.4	48.34 s	0.20 (01:14:11)	0.32 (05:02:34)
IMEX SSP2(3,3,2)-LSPUM, SCpCG	0.3	75.70 s	0.46 (02:52:12)	0.62 (09:38:21)
IMEX SSP2(3,3,2)-LSPUM, MG-parCG	0.3	75.45 s	0.12 (00:46:09)	0.35 (05:28:07)
IMEX SSP2(3,3,2)-LPUM, SCpCG	0.3	86.46 s	0.42 (02:34:30)	0.61 (09:34:53)
IMEX SSP2(3,3,2)-LPUM, MG-parCG	0.3	86.40 s	0.11 (00:42:31)	0.32 (05:00:19)

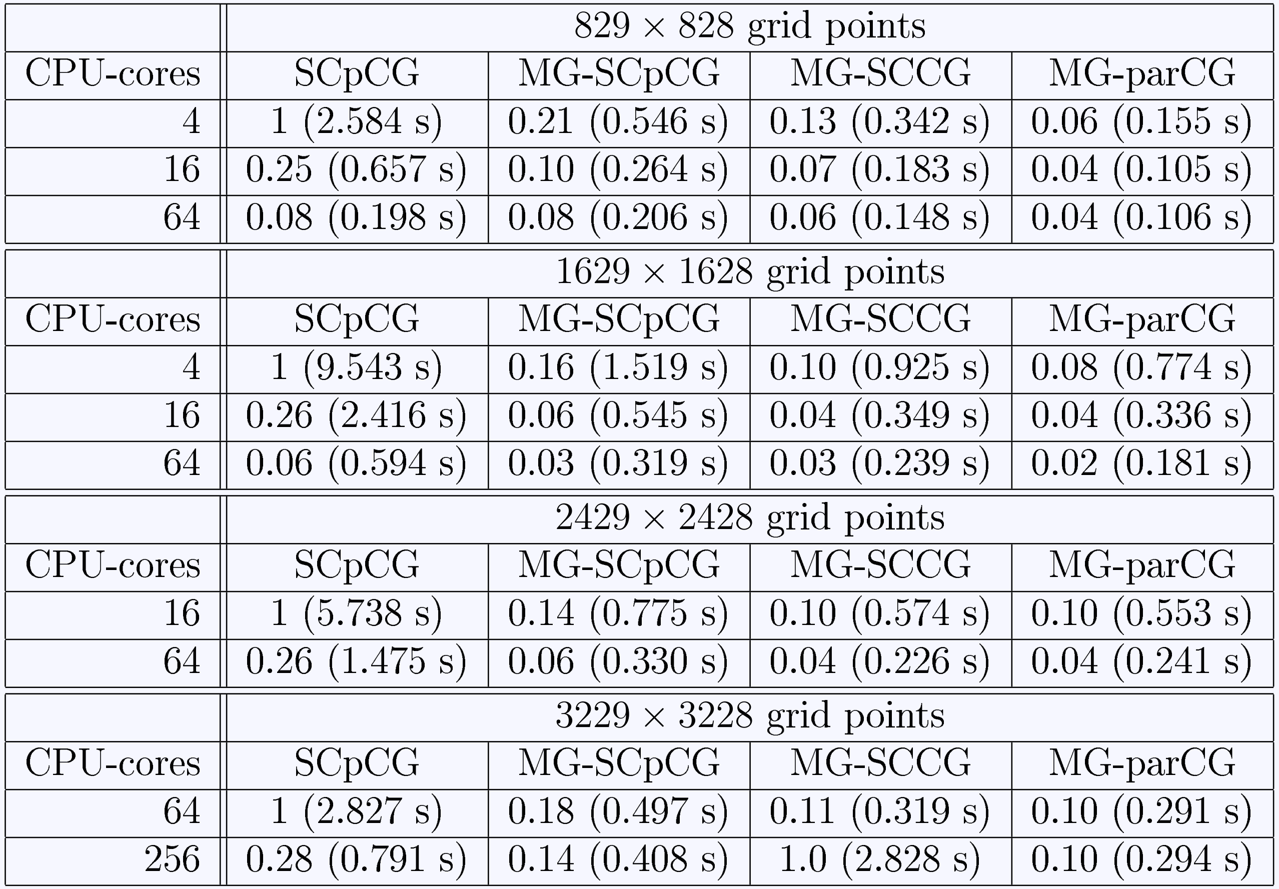
Table 5.5: Simulation Series B: The simulations have been performed on the VSC 2 using 64 CPU-cores. The effectively measured wallclocktime is given in the format (hh:mm:ss).

The efficiency of (parallel) multigrid solvers for the semiconvection problem on regular meshes (from Happenhofer (2013), PhD thesis, in prep.).

Only for the MG-parCG solver the effective gain in time allows actually outperforming the best explicit method (SSPRK(3,2), SCpCG) by halving total wall clock time.

For the latter, the SCpCG solver is more efficient than MG-parCG to solve the generalized Helmholtz equation occurring in Kwatra’s method, since the time steps remain small, so the pCG method converges sufficiently rapidly.

Results VII



	829 × 828 grid points			
CPU-cores	SCpCG	MG-SCpCG	MG-SCCG	MG-parCG
4	1 (2.584 s)	0.21 (0.546 s)	0.13 (0.342 s)	0.06 (0.155 s)
16	0.25 (0.657 s)	0.10 (0.264 s)	0.07 (0.183 s)	0.04 (0.105 s)
64	0.08 (0.198 s)	0.08 (0.206 s)	0.06 (0.148 s)	0.04 (0.106 s)
	1629 × 1628 grid points			
CPU-cores	SCpCG	MG-SCpCG	MG-SCCG	MG-parCG
4	1 (9.543 s)	0.16 (1.519 s)	0.10 (0.925 s)	0.08 (0.774 s)
16	0.26 (2.416 s)	0.06 (0.545 s)	0.04 (0.349 s)	0.04 (0.336 s)
64	0.06 (0.594 s)	0.03 (0.319 s)	0.03 (0.239 s)	0.02 (0.181 s)
	2429 × 2428 grid points			
CPU-cores	SCpCG	MG-SCpCG	MG-SCCG	MG-parCG
16	1 (5.738 s)	0.14 (0.775 s)	0.10 (0.574 s)	0.10 (0.553 s)
64	0.26 (1.475 s)	0.06 (0.330 s)	0.04 (0.226 s)	0.04 (0.241 s)
	3229 × 3228 grid points			
CPU-cores	SCpCG	MG-SCpCG	MG-SCCG	MG-parCG
64	1 (2.827 s)	0.18 (0.497 s)	0.11 (0.319 s)	0.10 (0.291 s)
256	0.28 (0.791 s)	0.14 (0.408 s)	1.0 (2.828 s)	0.10 (0.294 s)

Table 4.16: Results of the Multigrid scaling test. The normed wallclocktime is given. Note that for every testsuite, a different reference value $\mathrm{WTC}_{\mathrm{ref}}$ is assumed. To enable a comparison also between the distinct testsuites and thus to evaluate the effect of using more grid points on the same number of CPU-cores, the measured computational times are given in brackets.

Scaling of CG and MG methods for Kwatra's method only (the most expensive part for this problem anyway).

Though MG-parCG shows only weak scaling, it is always the fastest method for this kind of problem.

Using the Schur complement for solving the coarse grid equations in MG methods does not pay off here (from Happenhofer (2013), PhD thesis, in prep.).

Applications: Cepheids I

Simulations of pulsating Cepheids

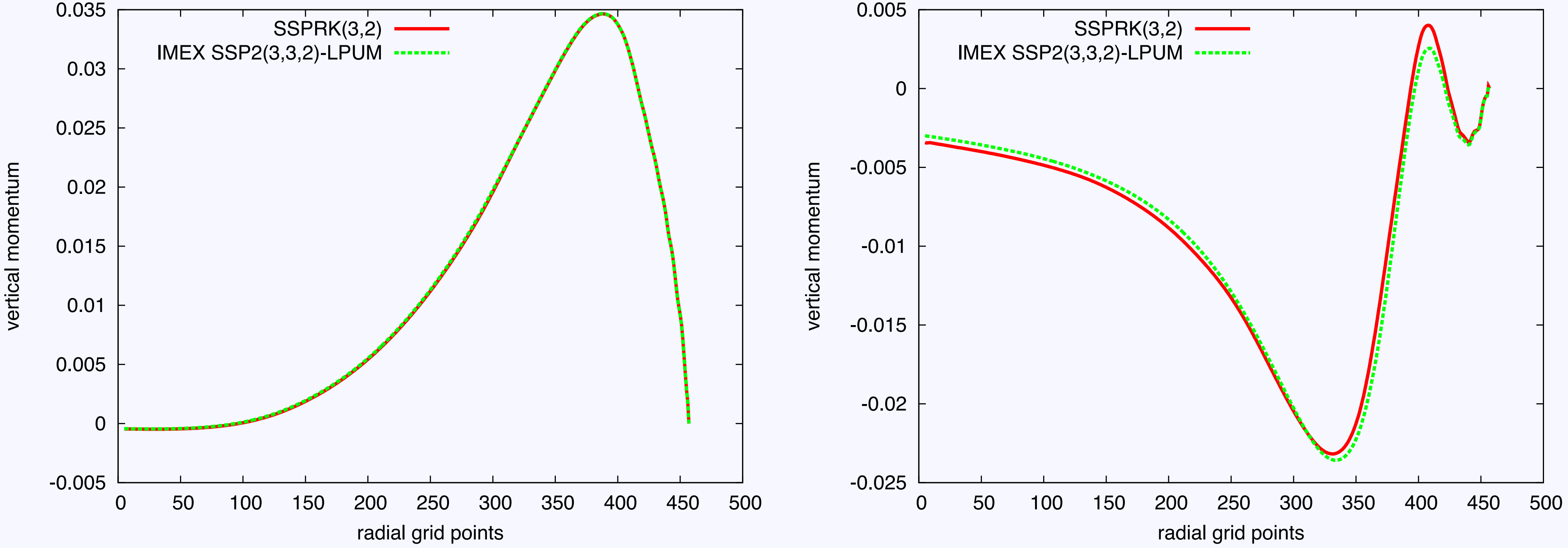
- preparing 2D and 3D simulations first consider 1D case: convection-pulsation interaction neglected
- for details on 2D models with explicit time integration see Mundprecht et al. (2013) (MNRAS, in print, for a preprint see arXiv:1209.2952)
- Cepheid model parameters:
  - $T_{\mathrm{eff}} = 5125$ K, $\log(g) \sim 1.97$, $M = 5\,M_\odot$, $R \sim 38.5\,R_\odot$, $L \sim 913\,L_\odot$, $X = 0.7$, $Y = 0.29$, $Z = 0.01$, $P = 3.85$ d, first overtone (1O)
  - outer 42% of R, vertical grid spacing: 0.47 Mm ... 124 Mm (modelling only the outer 42% ➜ implies P somewhat too short)
- 1D models: are used to create initial state for the 2D simulations
  - also assume spherical geometry
  - radially stretched grid co-moving with mean pulsation velocity
  - closed boundary conditions

Applications: Cepheids II

More details on the 1D models

- Simulation parameters:
  - $q = 1.007$, number of radial grid points = 454, Dirichlet boundary conditions, $T_{top} \sim 34{,}000$ K, $T_{bottom} \sim 320{,}000$ K
- method used to solve the nonlinear equations in the IMEX RK method
  - necessary because radiative conductivity depends on T, ρ, and composition
  - scalar, nonlinear Helmholtz equation on a polar, radially stretched grid
  - in tests, MG-FAS (multigrid full approximation storage) in matrix free form has lead to restrictive timesteps (unless local linearizations were used) and
  - suffers from high costs of frequent calls of EOS / opacity evaluations
- thus prefer Multigrid-Newton method which
  - benefits from the choice of FEM basis functions ((bi-) linear rectangular)
  - locality of equation of state (all partial derivatives such as $\partial K/\partial \rho$ precomputed as part of the EOS and opacity routines)
  - Jacobian matrix thus has same structure as the linear variant of the problem (thus low memory storage requirements)

Applications: Cepheids III



(a) t = 1 scrt

(b) t = 2 scrt

Vertical momentum distribution during contraction (1 scrt) and expansion (2 scrt) as obtained for two 1D simulation runs differing only in the time integration method and the resulting timestep size (N. Happenhofer, PhD thesis, 2013, to be submitted).

Applications: Cepheids IV

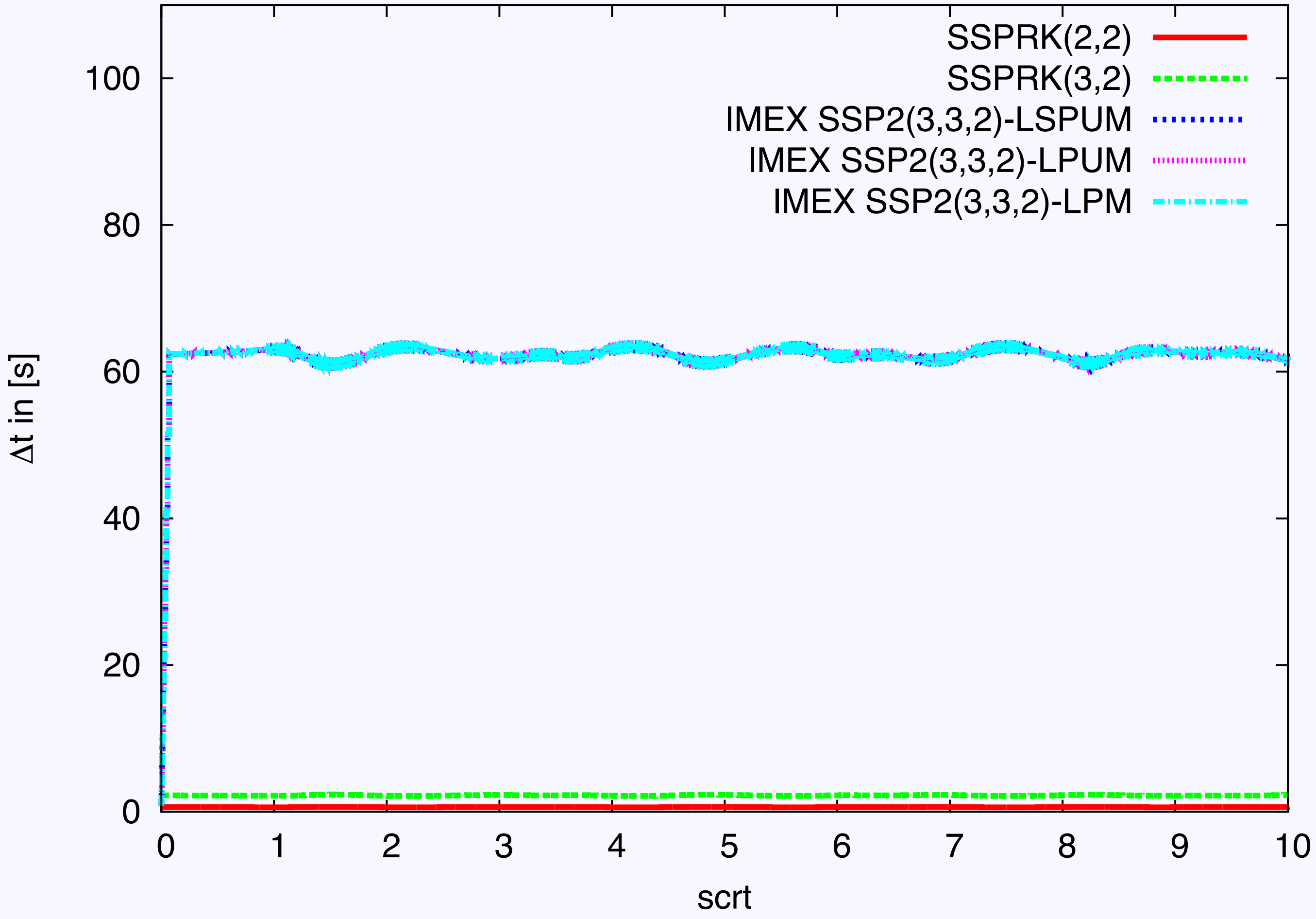
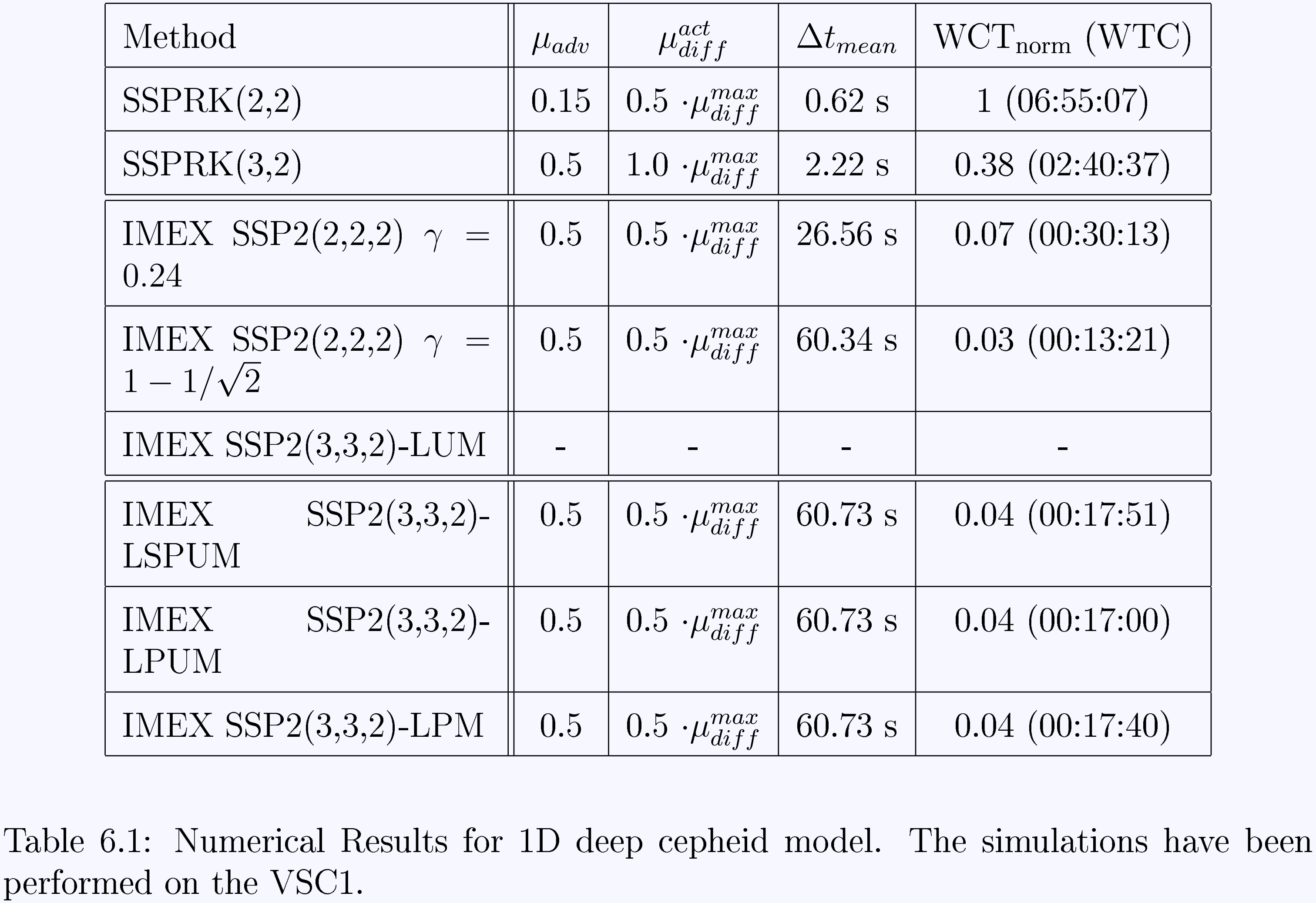


Figure 6.3: The timestep evolution with different time integrators in the 1D deep cepheid simulation.

(N. Happenhofer, PhD thesis, 2013, to be submitted)

Applications: Cepheids V



Method	μ_{adv}	μ_{diff}^{act}	Δt_{mean}	WCT$_{\text{norm}}$ (WTC)
SSPRK(2,2)	0.15	$0.5 \cdot \mu_{diff}^{max}$	0.62 s	1 (06:55:07)
SSPRK(3,2)	0.5	$1.0 \cdot \mu_{diff}^{max}$	2.22 s	0.38 (02:40:37)
IMEX SSP2(2,2,2) $\gamma = 0.24$	0.5	$0.5 \cdot \mu_{diff}^{max}$	26.56 s	0.07 (00:30:13)
IMEX SSP2(2,2,2) $\gamma = 1 - 1/\sqrt{2}$	0.5	$0.5 \cdot \mu_{diff}^{max}$	60.34 s	0.03 (00:13:21)
IMEX SSP2(3,3,2)-LUM	-	-	-	-
IMEX SSP2(3,3,2)-LSPUM	0.5	$0.5 \cdot \mu_{diff}^{max}$	60.73 s	0.04 (00:17:51)
IMEX SSP2(3,3,2)-LPUM	0.5	$0.5 \cdot \mu_{diff}^{max}$	60.73 s	0.04 (00:17:00)
IMEX SSP2(3,3,2)-LPM	0.5	$0.5 \cdot \mu_{diff}^{max}$	60.73 s	0.04 (00:17:40)

Table 6.1: Numerical Results for 1D deep cepheid model. The simulations have been performed on the VSC1.

Mean time steps achieved over 10 sound crossing times (N. Happenhofer, PhD thesis, 2013, to be submitted). Radiative diffusion no longer limits Δt.

Applications: Cepheids VI

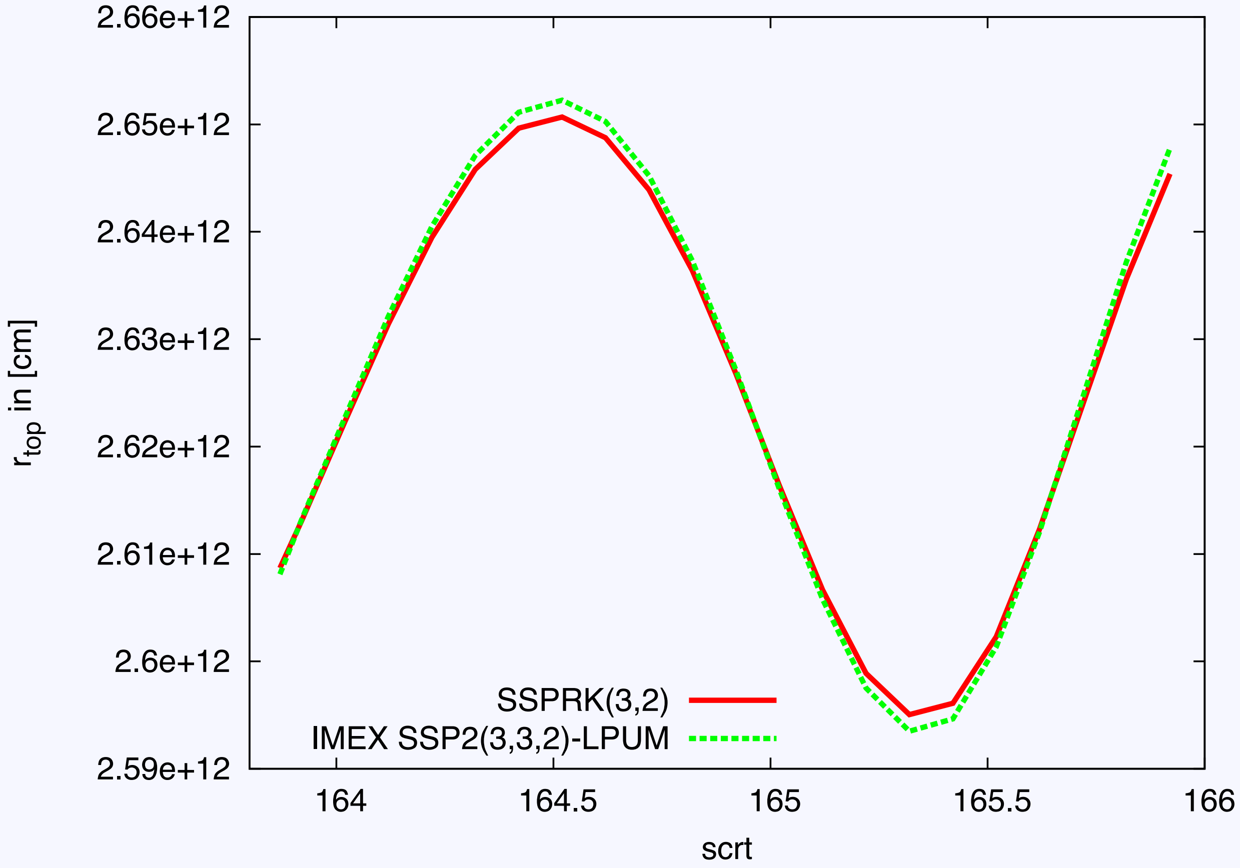
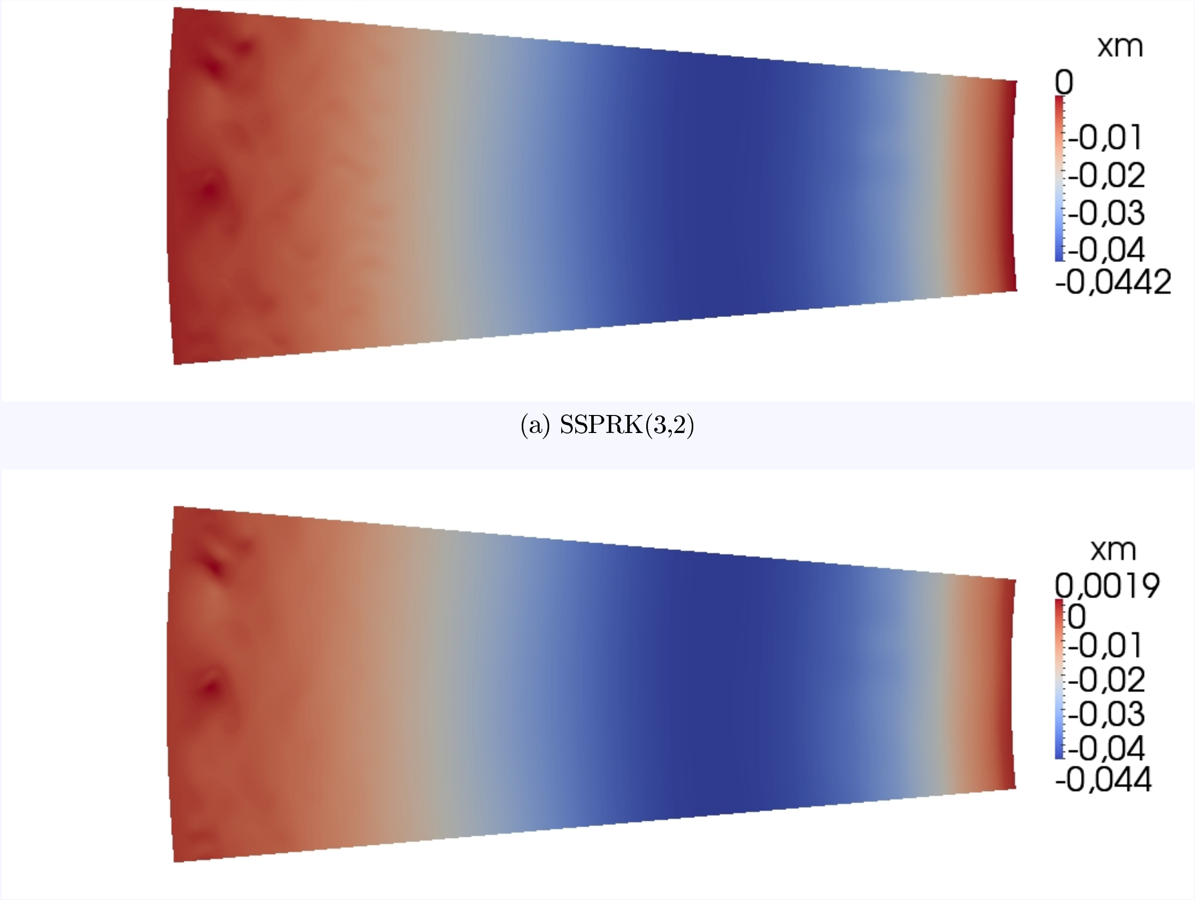


Figure 7.1: The evolution of the maximum radius r_{top} is depicted.

Outlook: 2D Cepheid model, 275 radial and 500 azimuthal grid points, q=1.11, 150 scrt in 1D followed by 13.82 scrt in 2D with explicit method as a starting point for another 2.2 scrt (~1 pulsation period) (N. Happenhofer, PhD thesis, 2013, to be submitted).

Applications: Cepheids VII



(a) SSPRK(3,2)

(b) IMEX SSP2(3,3,2)-LPUM

Figure 7.2: Snapshots of the 2D deep cepheid model at t=164.32 scrt, computed with SSPRK(3,2) and IMEX SSP2(3,3,2)-LPUM

Evolution of momentum in the 2D spherical wedge (N. Happenhofer, PhD thesis, 2013, to be submitted). Note: longterm stability of the boundary conditions needs improvement.

...THANK YOU FOR YOUR TIME !