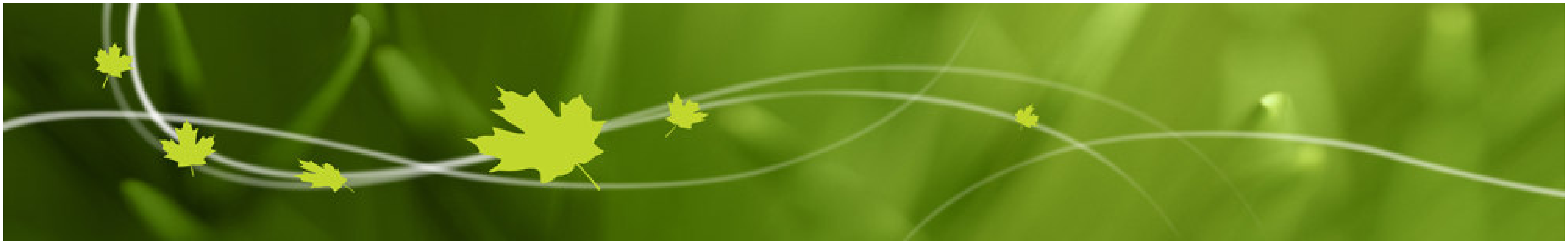
Environment Canada
Environnement Canada

Canada



Time integration schemes for numerical weather prediction

Colm Clancy

Janusz Pudykiewicz

25th of March 2014

Motivation

Williams (MWR, 2009):

Time-stepping

"…has received scant attention compared to the extensive research efforts devoted…"

Motivation

Williams (MWR, 2011):

“Contemporary atmospheric and oceanic numerical simulations are typically unconverged as the time step is reduced”

“…different time-stepping schemes in AGCMs produce substantially different climates.”

Motivation

Heimsund and Berntsen (Ocean Modelling, 2004)

“when using methods with implicit features and low viscosity, it may happen that models are stable for longer time steps but become unstable as the time step is reduced”

After spatial discretisation…

$$\frac{du}{dt} = F(u)$$

Two approaches to be discussed

- Semi-implicit predictor-corrector methods

  Clancy and Pudykiewicz (JCP, 2013)

- Exponential integration

  Clancy and Pudykiewicz (Tellus A, 2013)

1. Semi-implicit predictor-corrector methods

Separate ‘fast’ linear terms

$$\frac{du}{dt} = F(u)$$

$$\frac{du}{dt} = Lu + N(u)$$

Traditional semi-implicit (SILF)

$$\frac{u_{n+1} - u_{n-1}}{2\,\Delta t} = \underbrace{\frac{Lu_{n+1} + Lu_{n-1}}{2}}_{\text{Implicit (trapezoidal)}} + \underbrace{N\left(u_n\right)}_{\text{Leapfrog}}$$

Traditional semi-implicit (SILF)

$$\frac{u_{n+1} - u_{n-1}}{2\,\Delta t} = \frac{L u_{n+1} + L u_{n-1}}{2} + N\left(u_n\right)$$

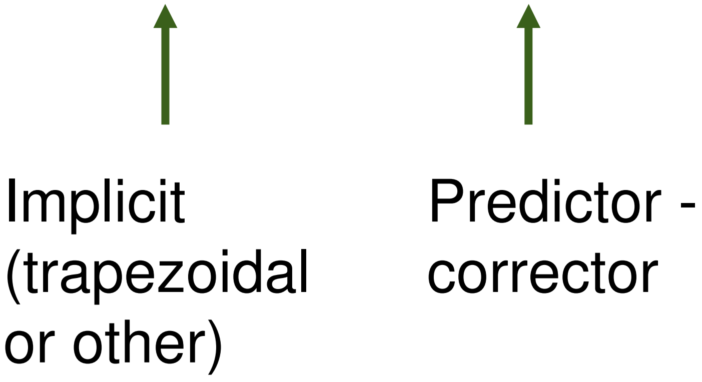
Need for Robert-Asselin filter: reduces accuracy and stability

Proposed improvements:

- Williams (MWR; 2009, 2011, 2013)
- Li and Trenchea (JCP, 2014)
- Moustaoui et al. (MWR, 2014)
- Durran and Blossey (MWR, 2012)

Proposed alternative

$$\frac{du}{dt} = Lu + N(u)$$



Implicit (trapezoidal or other)

Predictor - corrector

Predictor-corrector for nonlinear terms

Leapfrog trapezoidal (**LFT**) Kurihara (MWR, 1965)

$$\frac{u_* - u_{n-1}}{2\,\Delta t} = N(u_n)$$

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{1}{2}N(u_*) + \frac{1}{2}N(u_n)$$

Predictor-corrector for nonlinear terms

Adams-Bashforth trapezoidal (**ABT**) Kar (MWR, 2012)

$$\frac{u_* - u_n}{\Delta t} = \frac{3}{2}N(u_n) - \frac{1}{2}N(u_{n-1})$$

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{1}{2}N(u_*) + \frac{1}{2}N(u_n)$$

Predictor-corrector for nonlinear terms

Adams-Bashforth-Moulton (**ABM**) Durran (1999)

$$\frac{u_* - u_n}{\Delta t} = \frac{3}{2}N(u_n) - \frac{1}{2}N(u_{n-1})$$

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{5}{12}N(u_*) + \frac{8}{12}N(u_n) - \frac{1}{12}N(u_{n-1})$$

Implicit for linear terms

Trapezoidal (**T**)

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{1}{2} L u_{n+1} + \frac{1}{2} L u_n$$

AM2*: Durran and Blossey (MWR, 2012)

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{3}{4} L u_{n+1} + \frac{1}{4} L u_{n-1}$$

Sample combinations: T-ABT

$$\frac{u_* - u_n}{\Delta t} = \frac{1}{2}Lu_* + \frac{1}{2}Lu_n + \frac{3}{2}N(u_n) - \frac{1}{2}N(u_{n-1})$$

$$\frac{u_{n+1} - u_n}{\Delta t} = \frac{1}{2}Lu_{n+1} + \frac{1}{2}Lu_n + \frac{1}{2}N(u_*) + \frac{1}{2}N(u_n)$$

Sample combinations: AM2*-ABM

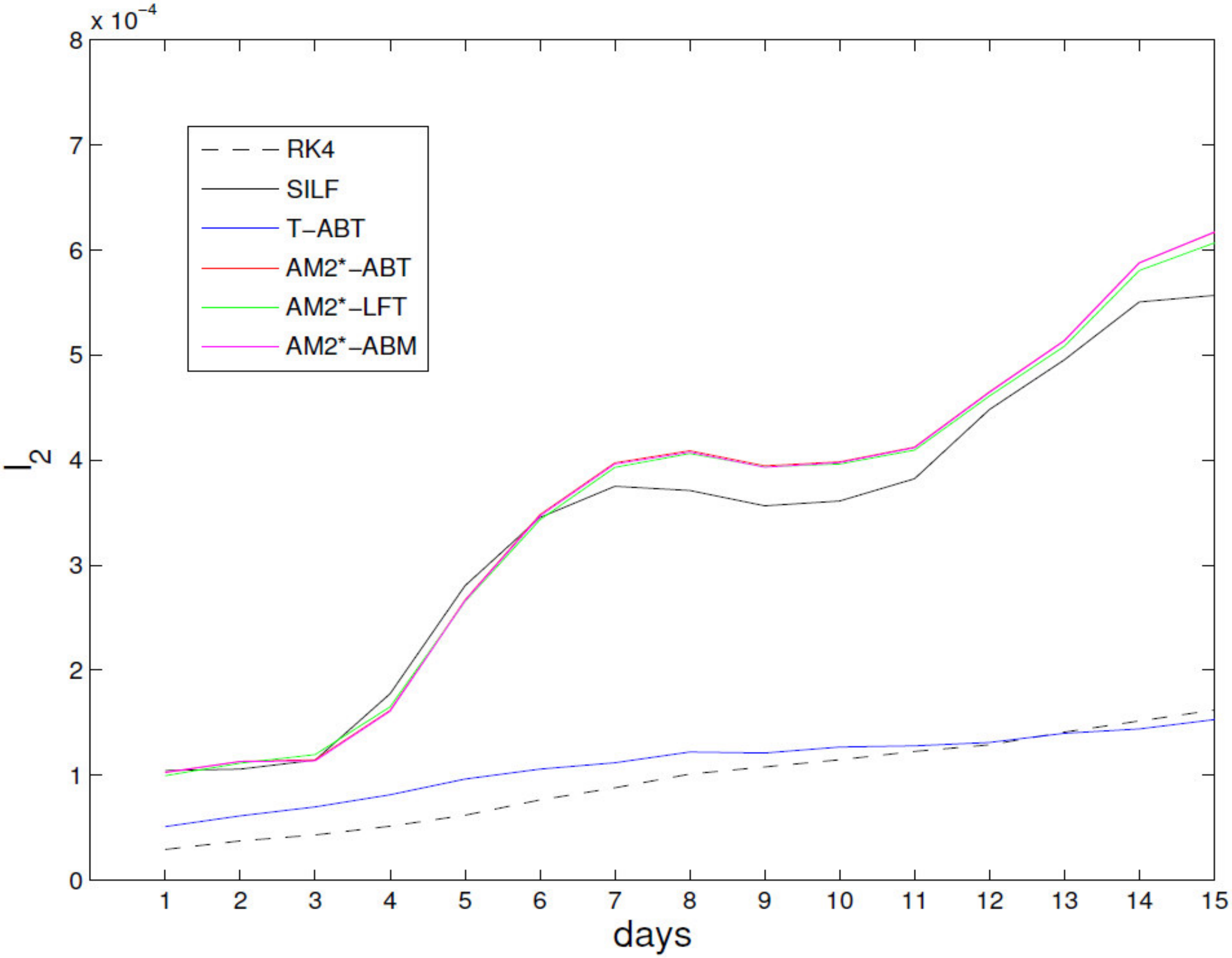
$$
\begin{aligned}
\frac{u_* - u_n}{\Delta t} &= \frac{3}{4} L u_* + \frac{1}{4} L u_{n-1} \\
&+ \frac{3}{2} N(u_n) - \frac{1}{2} N(u_{n-1})
\end{aligned}
$$

$$
\begin{aligned}
\frac{u_{n+1} - u_n}{\Delta t} &= \frac{3}{4} L u_{n+1} + \frac{1}{4} L u_{n-1} \\
&+ \frac{5}{12} N(u_*) + \frac{8}{12} N(u_n) - \frac{1}{12} N(u_{n-1})
\end{aligned}
$$

Shallow water tests

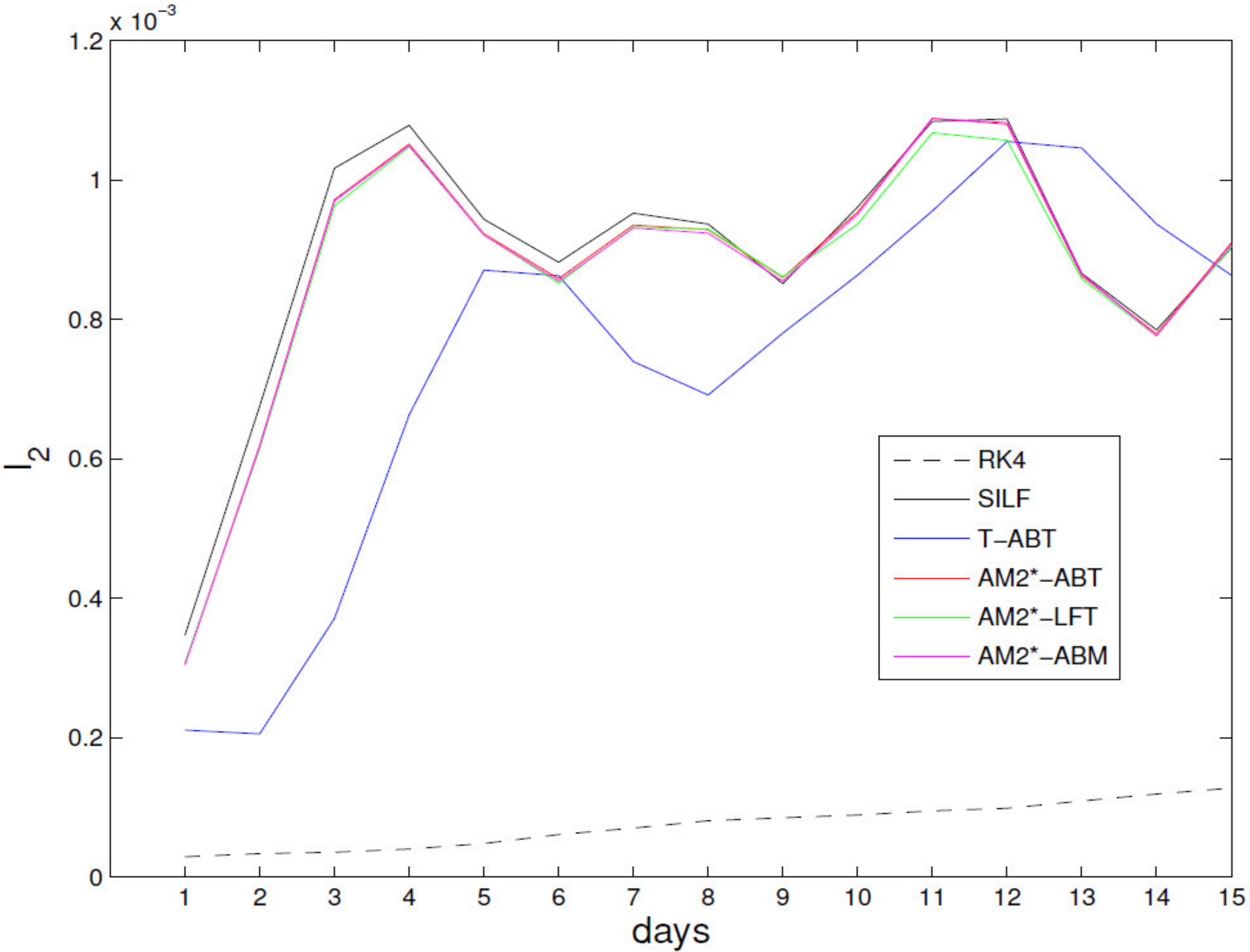
- Shallow water model of Pudykiewicz (JCP, 2011)
- Iterative GCR(4) solver for Helmholtz equations (Smolarkiewicz and Margolin, 2000)
- No explicit diffusion
- Filter of Williams (MWR, 2011) for the semi-implicit leapfrog
- Spatial resolution: grid 6 (40,962 nodes, ~112km).
  Reference: grid 7 (163,842 nodes, ~56km) with RK4 at 90s time-step

Williamson et al. (1992) – Mountain case



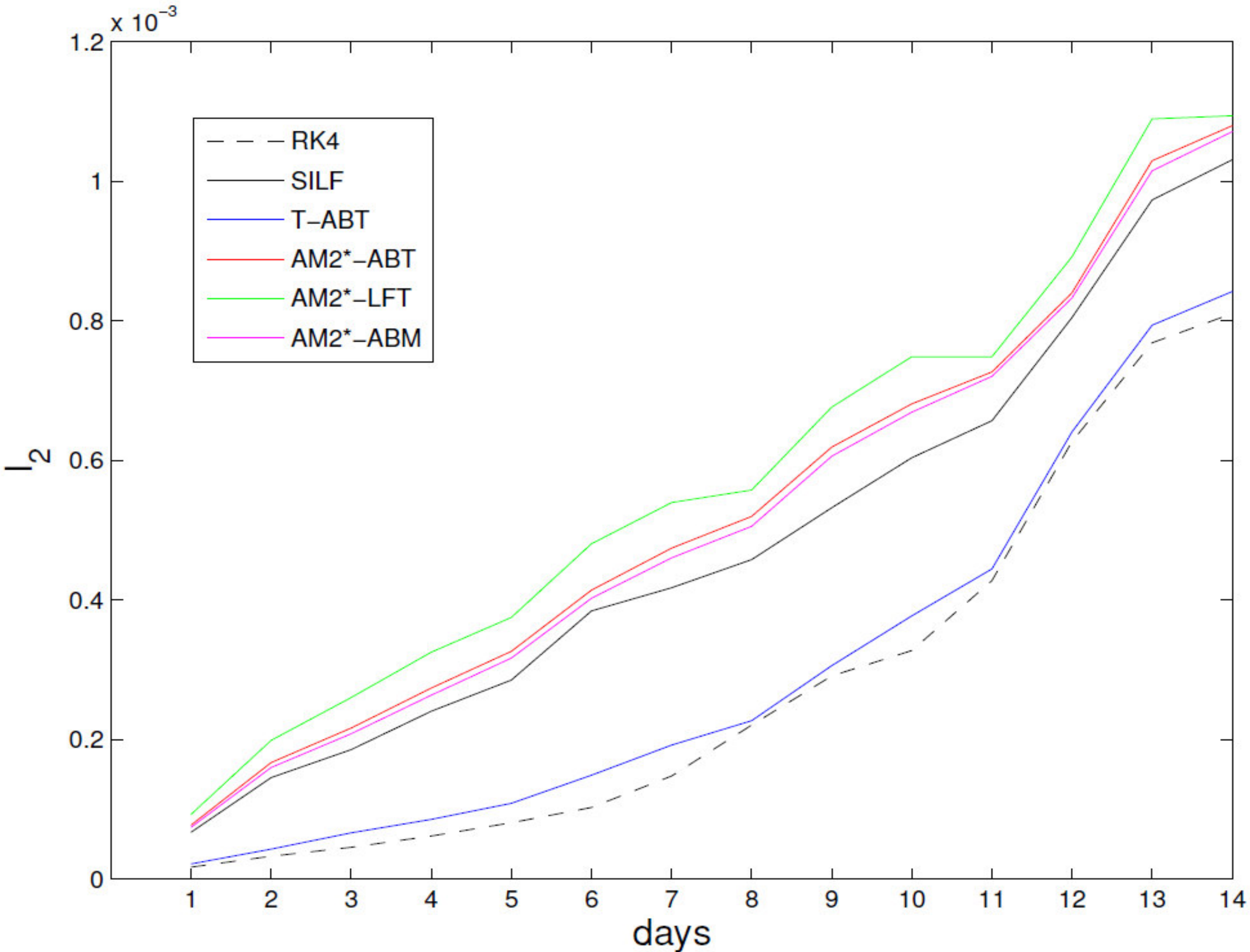
$\Delta t = 240s$

Williamson et al. (1992) – Mountain case



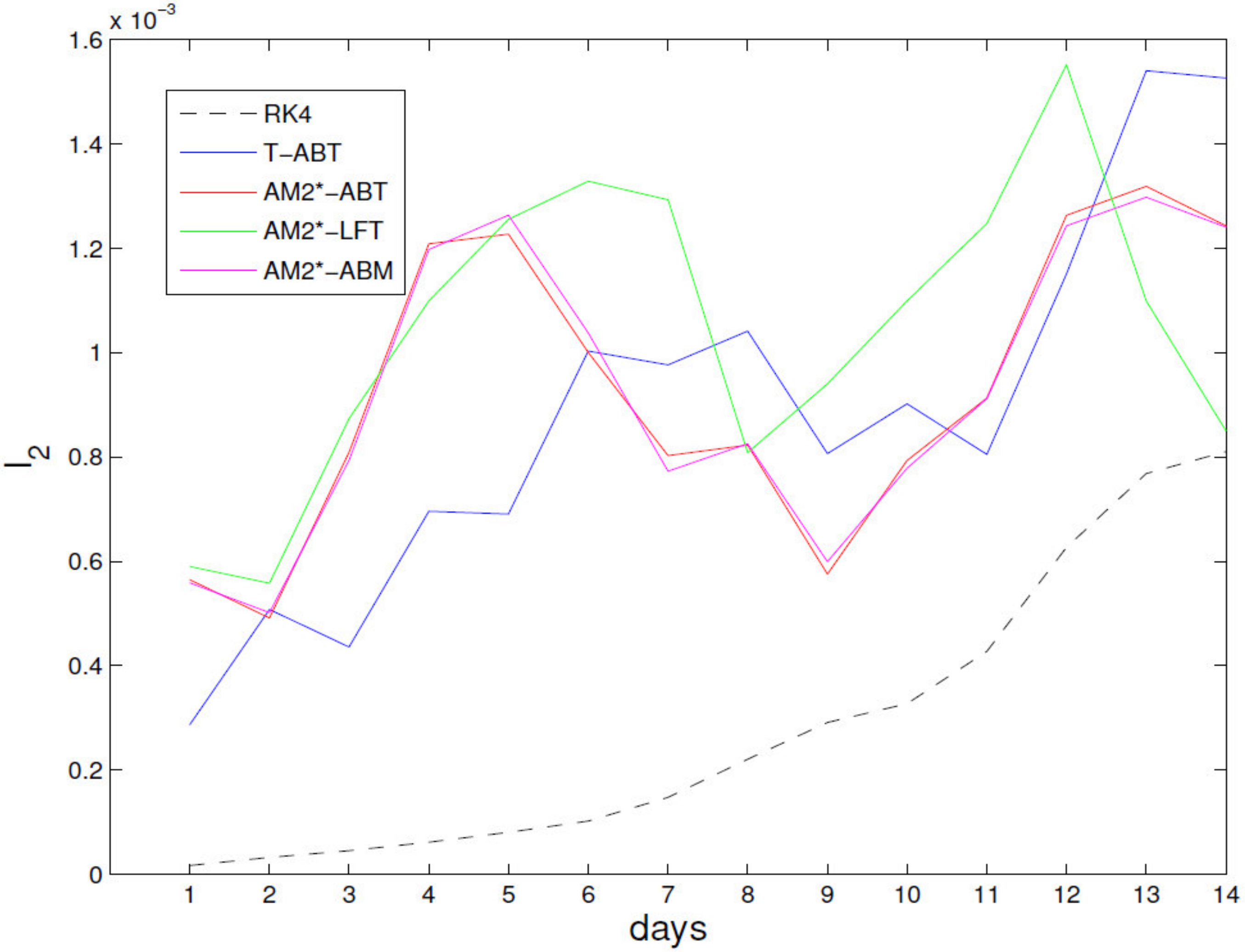
Δ t = 900s

Williamson et al. (1992) – RH wave case



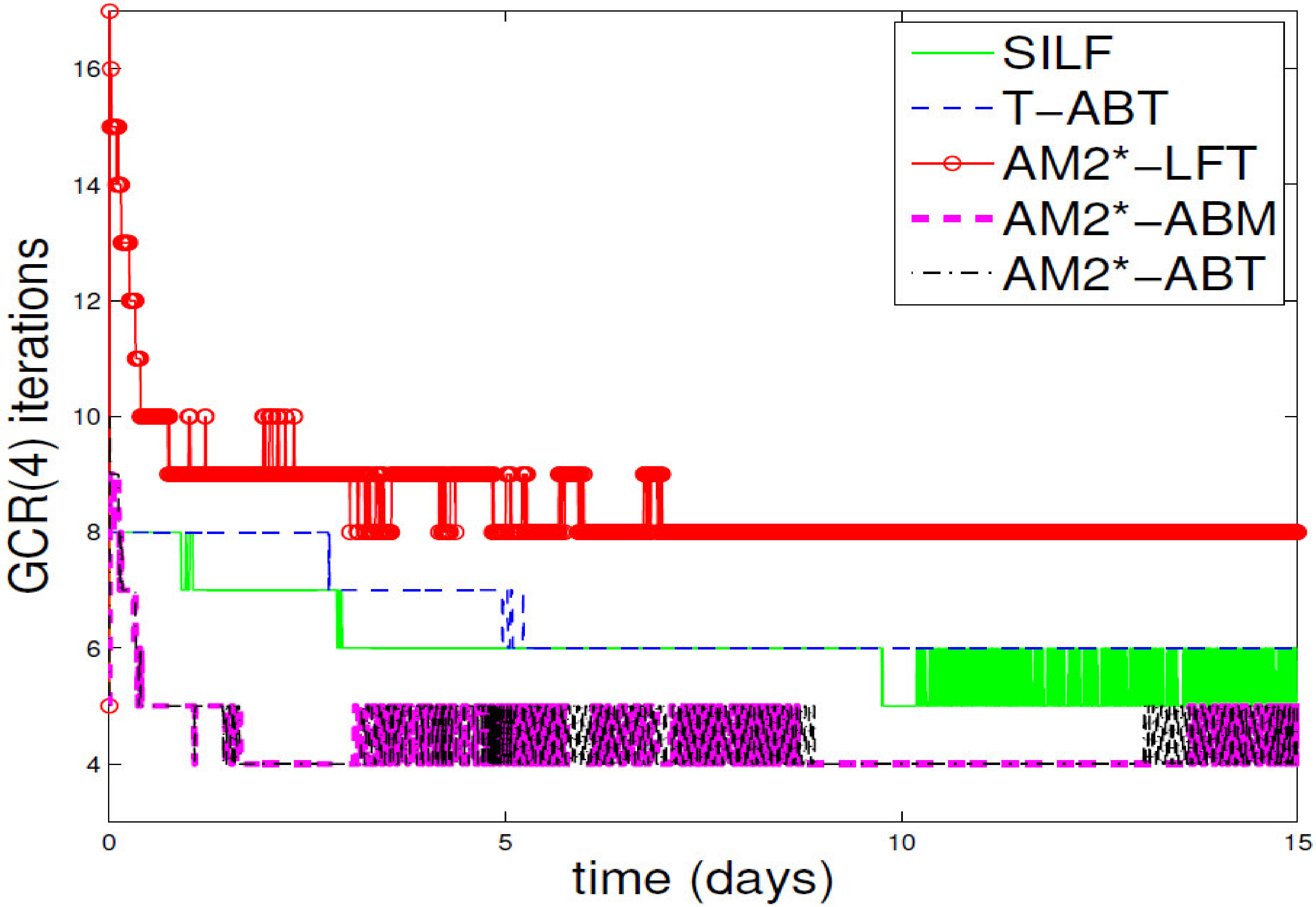
Δ t = 240s

Williamson et al. (1992) – RH wave case



$\Delta t = 900s$

Efficiency


SILF
T–ABT
AM2*–LFT
AM2*–ABM
AM2*–ABT
GCR(4) iterations
16
14
12
10
8
6
4
0
5
10
15
time (days)

Conclusions from linear analysis & numerical tests

Semi-implicit predictor-corrector methods as an alternative to traditional semi-implicit:

- Accuracy comparable or better
- Better stability
- No time filter required
- Efficiency not affected

Clancy and Pudykiewicz (JCP, 2013)

2. Exponential integration

Linear ODE: integrating factors

$$\frac{dy}{dt} = c\,y, \qquad y(0) = 1$$

$$y(t) = e^{ct}$$

Linear ODE: integrating factors

$$\frac{dy}{dt} = c\,y, \qquad y(0) = 1$$

$$\frac{dy}{dt} e^{-ct} = c\,y\,e^{-ct}$$

Linear ODE: integrating factors

$$\frac{dy}{dt} = c\,y + r, \qquad y(0) = 1$$

$$\frac{dy}{dt} e^{-ct} = c\,y\,e^{-ct} + r\,e^{-ct}$$

Back to discretised PDE system

$$\frac{du}{dt} = F(u)$$

Expand around time level *n*:

$$\frac{d}{dt}u(t) = F_n + J_n\left(u(t) - u_n\right) + R\left(u(t)\right)$$

$$R\left(u(t)\right) = F\left(u(t)\right) - F_n - J_n\left(u(t) - u_n\right)$$

Integrating factor solution: $e^{-J_n t}$

Multiply by integrating factor
to get *exact* solution:

$$u_{n+1} = u_n + \left(e^{\Delta t J_n} - I\right) J_n^{-1} F_n + \int_0^{\Delta t} R\left(u(n\Delta t + s)\right) ds$$

Some options

$$u_{n+1} = u_n \quad + \quad \left(e^{\Delta t\, J_n} - I\right) J_n^{-1} F_n$$

$$\begin{aligned} u_{n+1} &= u_n + \left(e^{\Delta t\, J_n} - I\right) J_n^{-1} F_n \\ &+ \frac{2}{3}\Delta t \left(e^{\Delta t\, J_n} - I - \Delta t\, J_n\right) \left(\Delta t\, J_n\right)^{-2} R_{n-1} \end{aligned}$$

Family of functions of matrix exponentials

$$\varphi_0(M) = e^M$$

$$\varphi_1(M) = \left(e^M - I\right) M^{-1}$$

$$\varphi_2(M) = \left(e^M - I - M\right) M^{-2}$$

Family of functions of matrix exponentials

$$\varphi_0(M) = e^M$$

$$\varphi_1(M) = \left(e^M - I\right) M^{-1}$$

$$\varphi_2(M) = \left(e^M - I - M\right) M^{-2}$$

$$u_{n+1} = u_n \quad + \quad \left(e^{\Delta t J_n} - I\right) J_n^{-1} F_n$$

$$\hookrightarrow \quad u_{n+1} = u_n + \varphi_1(\Delta t J_n) \Delta t \, F_n$$

Family of functions of matrix exponentials

$$u_{n+1} = u_n + \left(e^{\Delta t J_n} - I\right) J_n^{-1} F_n$$

$$+ \frac{2}{3}\Delta t \left(e^{\Delta t J_n} - I - \Delta t\, J_n\right) \left(\Delta t\, J_n\right)^{-2} R_{n-1}$$

$$u_{n+1} = u_n + \varphi_1(\Delta t J_n)\Delta t\, F_n + \frac{2}{3}\varphi_2(\Delta t J_n)\Delta t\, R_{n-1}$$

Generally looking for: $u_{n+1} = u_n + \sum_{k=0}^{p} \varphi_k(M) b_k$

Complications

$$e^M = I + M + \frac{1}{2!}M^2 + \frac{1}{3!}M^3 + \ldots$$

Moler and Van Loan, (SIAM Rev 2003),
Nineteen Dubious Ways to Compute the Exponential of a Matrix

Hope

- Don’t need matrix exponential itself; just the action on a vector
- Numerous algorithms appearing and rapidly improving
- Designed specifically for sums involving phi-functions:

$$\sum_{k=0}^{p} \varphi_k(M) b_k$$

phipm: Niesen and Wright (2012)

Krylov space:

$$K_m(J,b) = \mathrm{span}\left\{b, Jb, J^2b, \ldots, J^{m-1}b\right\}$$

$$J \rightarrow H_m = V_m^T \, J \, V_m \qquad\qquad m \ll N$$

expmv: Al-Mohy and Higham (2011)

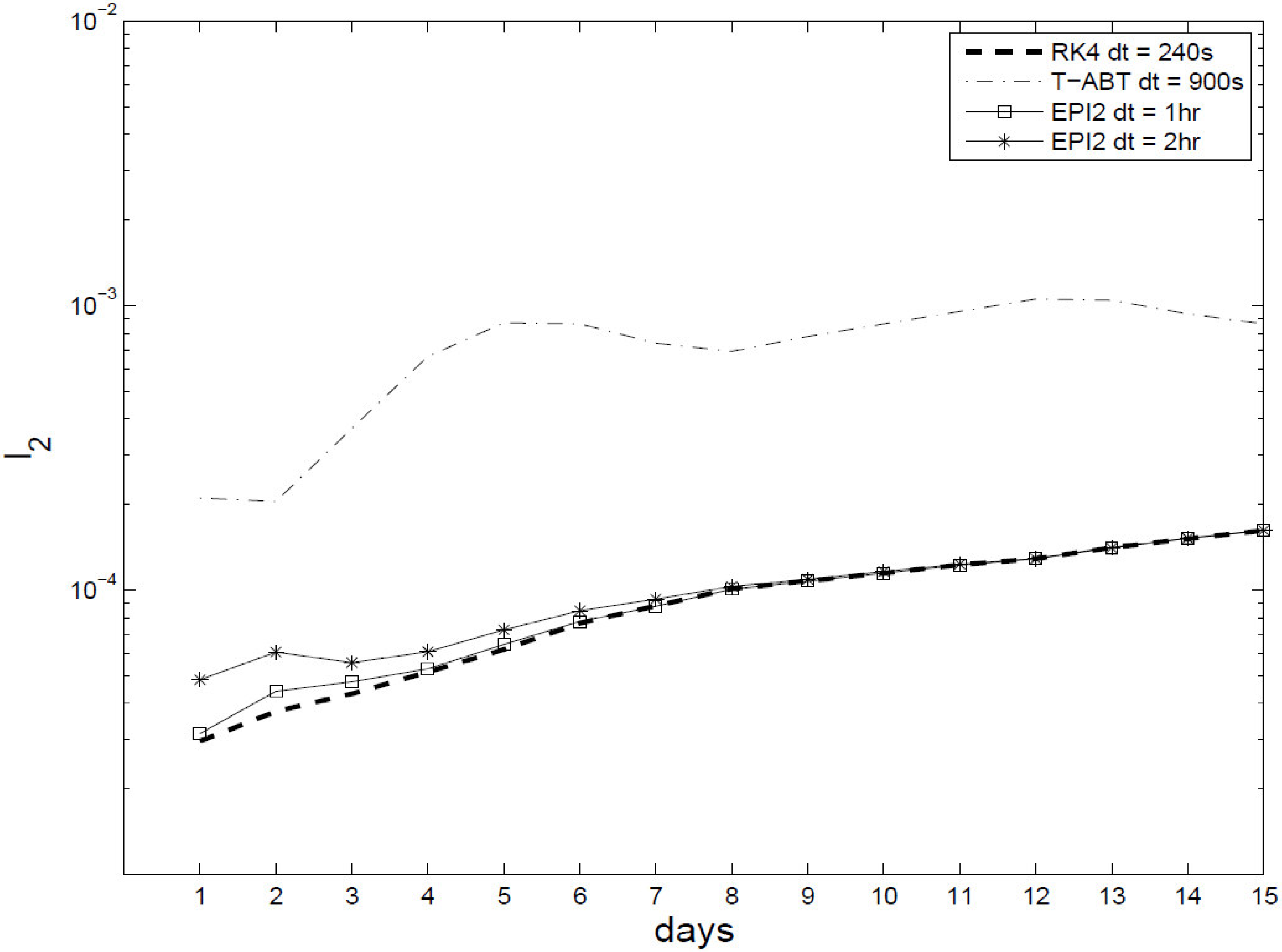
$$e^J = \left(e^{s^{-1}J}\right)^s$$

$$e^J b = \left[T_m(s^{-1}J)\right]^s \, b$$

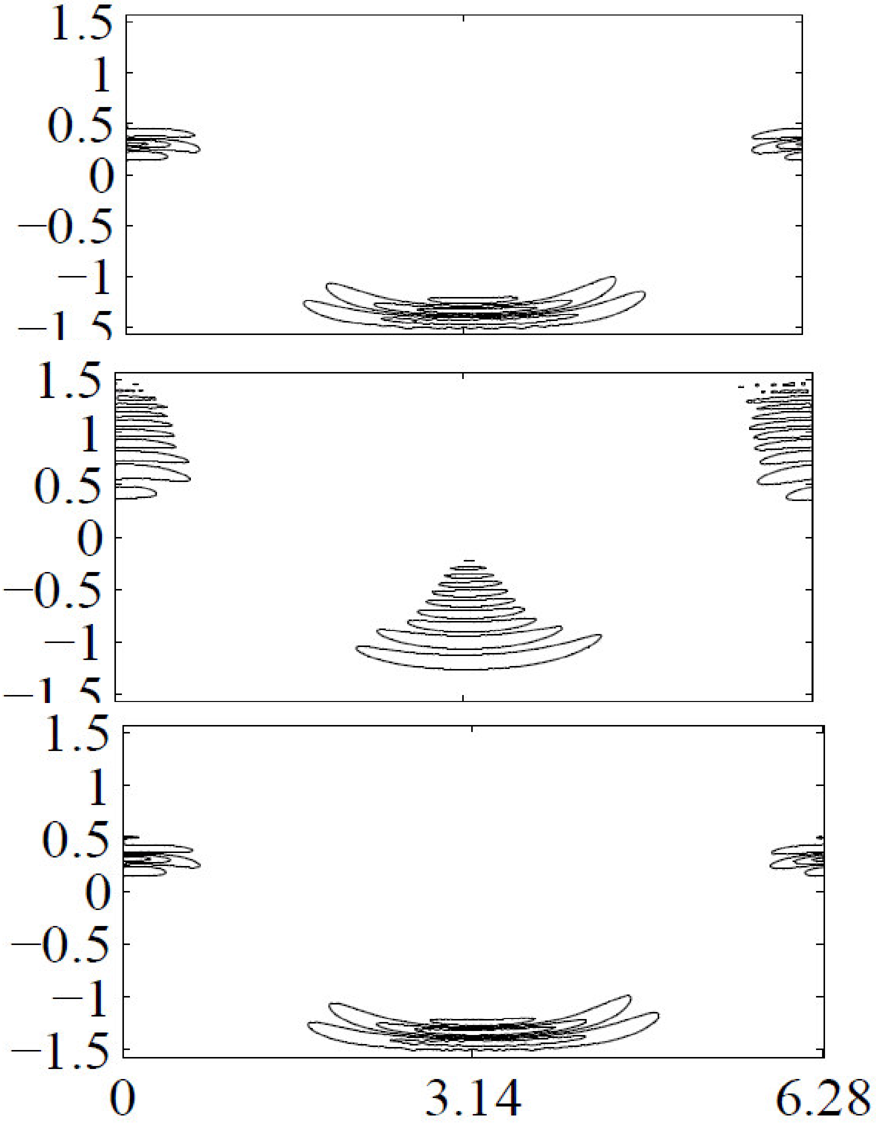
Shallow water tests

- Full details in Clancy and Pudykiewicz (Tellus A, 2013)
- Compared with explicit RK4 and the semi-implicit predictor corrector T-ABT
- **phipm** algorithm tested
- Exponential methods showed high accuracy and stability

Williamson et al. (1992) – Mountain case



Galewsky et al. (2004) – divergence after 12 hours



RK4 240s

T-ABT 900s

EPI2 7200s

Shallow water tests

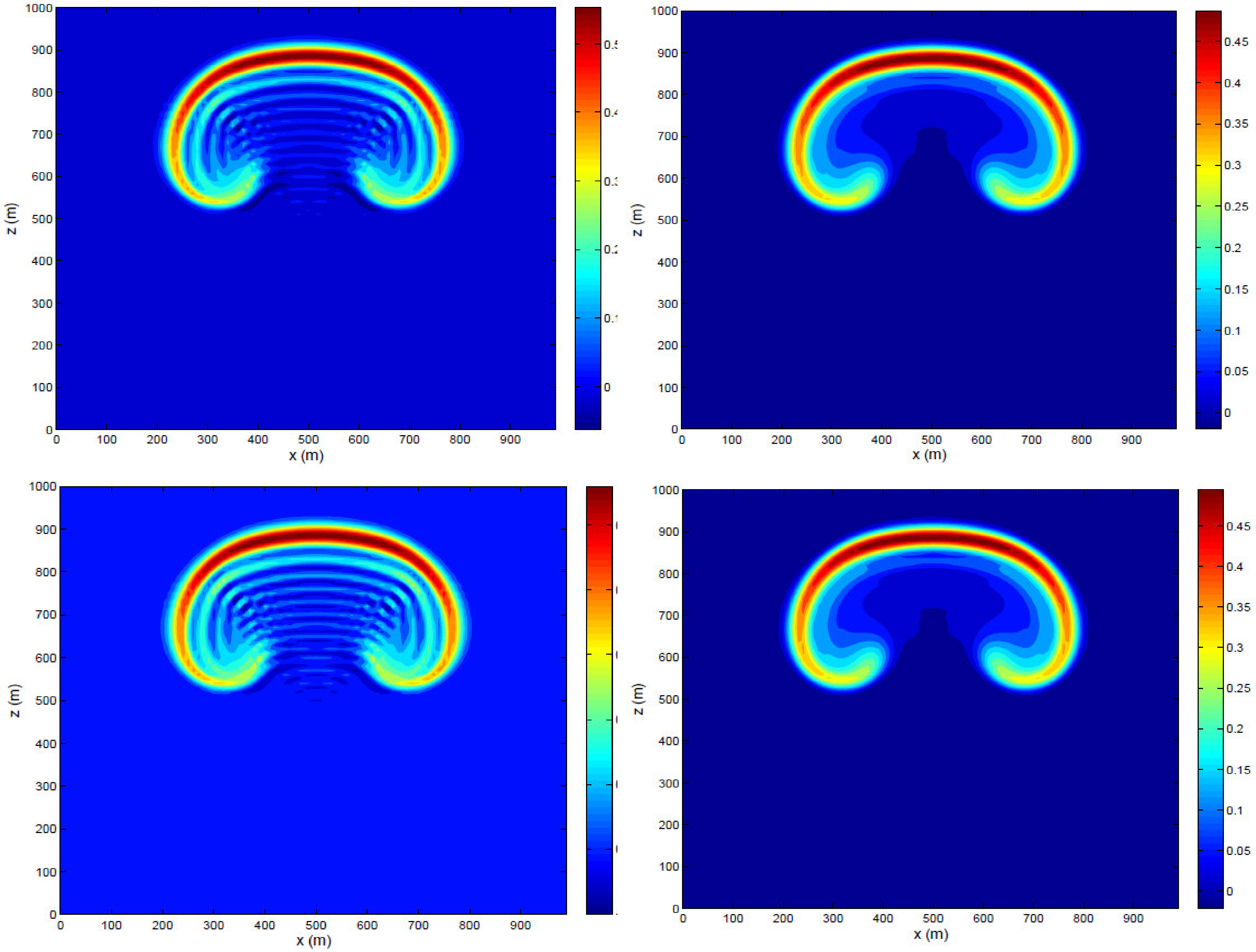
- Full details in Clancy and Pudykiewicz (Tellus A, 2013)
- Compared with explicit RK4 and the semi-implicit predictor corrector T-ABT
- **phipm** algorithm tested
- Exponential methods showed high accuracy and stability
- Execution time comparable with the explicit

Euler equations in 2D

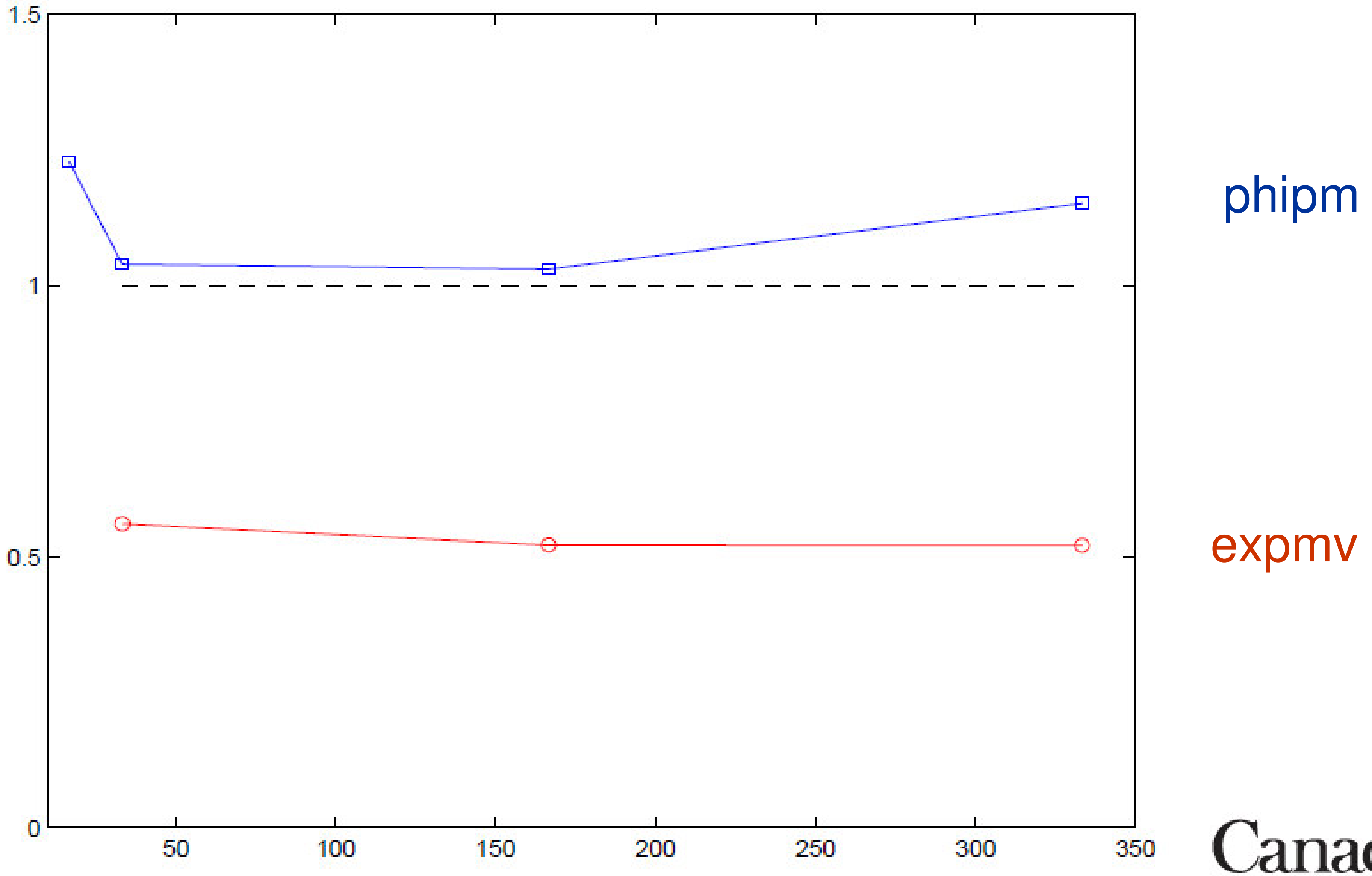
$$
\begin{aligned}
\frac{\partial \mathbf{u}}{\partial t} &= -\mathbf{u}.\nabla \mathbf{u} - c_p \theta \nabla \pi - g\mathbf{k} \\
\frac{\partial \pi}{\partial t} &= -\mathbf{u}.\nabla \pi - \frac{R}{c_v} \pi \nabla . \mathbf{u} \\
\frac{\partial \theta}{\partial t} &= -\mathbf{u}.\nabla \theta
\end{aligned}
$$

Giraldo and Restelli (JCP, 2008): inertia-gravity waves, thermal bubble

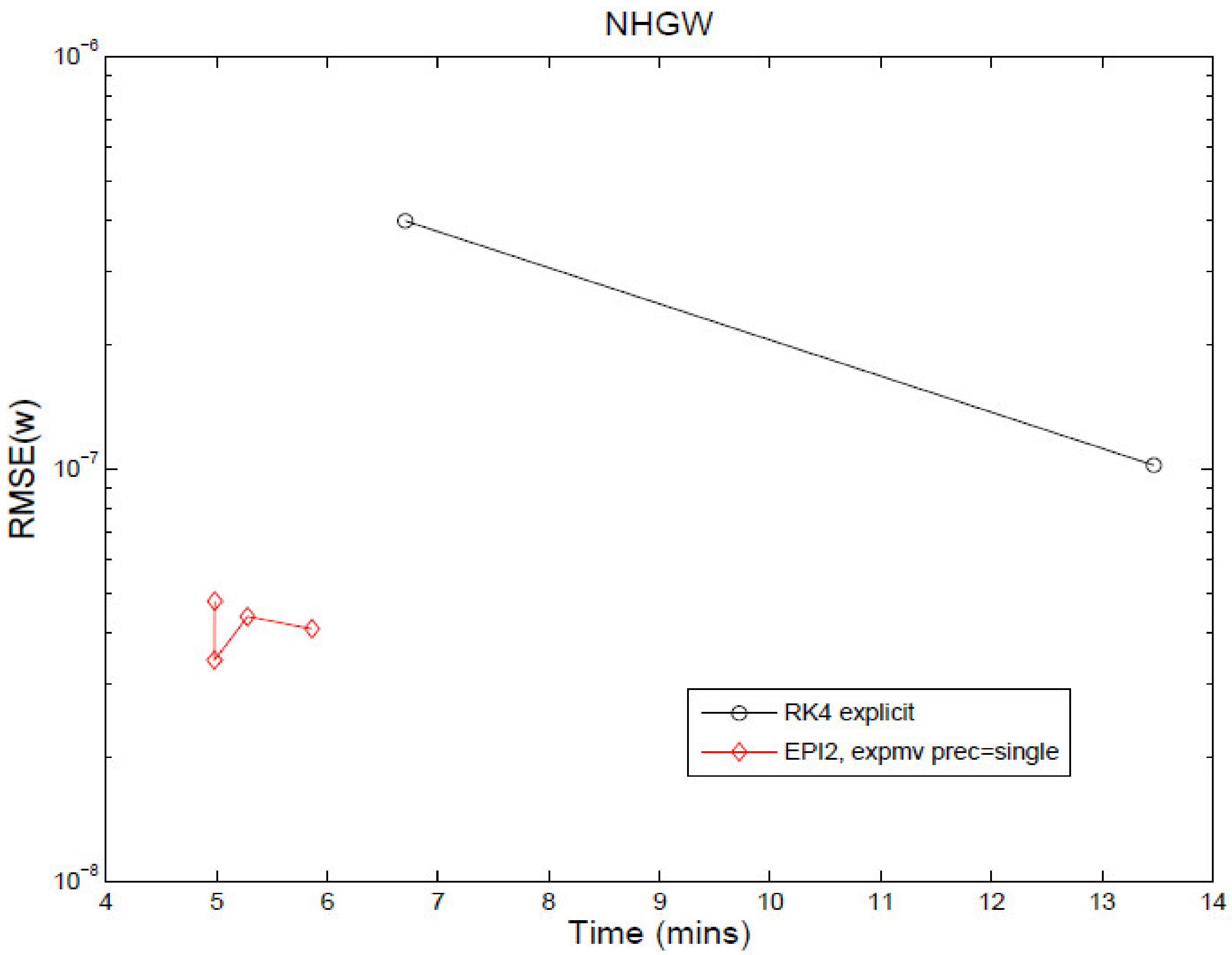
“Simple” spatial discretisation:
x-z plane, unstaggered grid, centred differencing, periodic in x


z (m)
x (m)
0
100
200
300
400
500
600
700
800
900
1000
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0

Time-step vs execution time relative to explicit



Error vs execution time



Summary and future

Exponential integration methods offer very high accuracy and stability

Summary and future

Exponential integration methods offer very high accuracy and stability

Improving efficiency is an ongoing effort. Algorithm progress is encouraging

Summary and future

Exponential integration methods offer very high accuracy and stability

Improving efficiency is an ongoing effort. Algorithm progress is encouraging

Other uses and approaches?

Advection, split vertical….