Dynamics methods for high resolution NWP modeling

Is the high resolution modeling really a limitation for spectral methods with SI SL?

Filip Váňa

profiting from discussions with M. Tolstykh, S. Saarinen and J. Vivoda

filip.vana@chmi.cz

ONPP / ČHMÚ - LACE

Issues to be discussed

Features of focus

- Transport scheme - SL advection
- Time scheme - SI
- Spectral techniques

Other issues of lesser interest for this talk

- Staggering
- Vertical coordinate
- Elastic vs. anelastic approximation
- Hydrostatic vs. non-hydrostatic
- ...

Dynamics is a system

- There are (as always) some rules/implications...
  - not much sense for semi-Lagrangian without SI (or RK)
  - elastic NH dynamics with explicit time stepping allows no homogeneous solution
  - any solution aiming at rather long and ’homogeneous’ (i.e. without splitting) time-steps requires a solver
  - ...

Dynamics is a system

- There are (as always) some rules/implications...
  - not much sense for semi-Lagrangian without SI (or RK)
  - elastic NH dynamics with explicit time stepping allows no homogeneous solution
  - any solution aiming at rather long and ’homogeneous’ (i.e. without splitting) time-steps requires a solver
  - ...
- Usually any change implies some consequences elsewhere (SI $\Rightarrow$ SL, ...)

Dynamics is a system

- There are (as always) some rules/implications...
  - not much sense for semi-Lagrangian without SI (or RK)
  - elastic NH dynamics with explicit time stepping allows no homogeneous solution
  - any solution aiming at rather long and 'homogeneous' (i.e. without splitting) time-steps requires a solver
  - ...
- Usually any change implies some consequences elsewhere (SI $\Rightarrow$ SL, ...)
- Better to improve a system by rather extending the existing solutions, unless there are some really blocking obstacles...

Beware of machine error

- Error of the scheme should not be considered with respect to a single timestep. It is the whole forecast which matters!

Beware of machine error

- Error of the scheme should not be considered with respect to a single timestep. It is the whole forecast which matters!
- Test of the linear TL and AD models

Eulerian advection (1 hour, 30 timesteps, Δt = 120 s)

*ADJOINT TEST: THE DIFFERENCE IS **10.395** TIMES THE ZERO OF THE MACHINE*

SL advection (1 hour, 30 timesteps, Δt= 120 s)

*ADJOINT TEST: THE DIFFERENCE IS **16.562** TIMES THE ZERO OF THE MACHINE*

SL advection (1 hour, 10 timesteps, Δt= 360 s)

*ADJOINT TEST: THE DIFFERENCE IS **5.452** TIMES THE ZERO OF THE MACHINE*

Beware of machine error

- Error of the scheme should not be considered with respect to a single timestep. It is the whole forecast which matters!
- Test of the linear TL and AD models

Eulerian advection (1 hour, 30 timesteps, Δt = 120 s)

*ADJOINT TEST: THE DIFFERENCE IS **10.395** TIMES THE ZERO OF THE MACHINE*

SL advection (1 hour, 30 timesteps, Δt= 120 s)

*ADJOINT TEST: THE DIFFERENCE IS **16.562** TIMES THE ZERO OF THE MACHINE*

SL advection (1 hour, 10 timesteps, Δt= 360 s)

*ADJOINT TEST: THE DIFFERENCE IS **5.452** TIMES THE ZERO OF THE MACHINE*

$\Rightarrow$ **For two comparable schemes the one allowing longer timestep should be always preferable**

Semi Implicit scheme

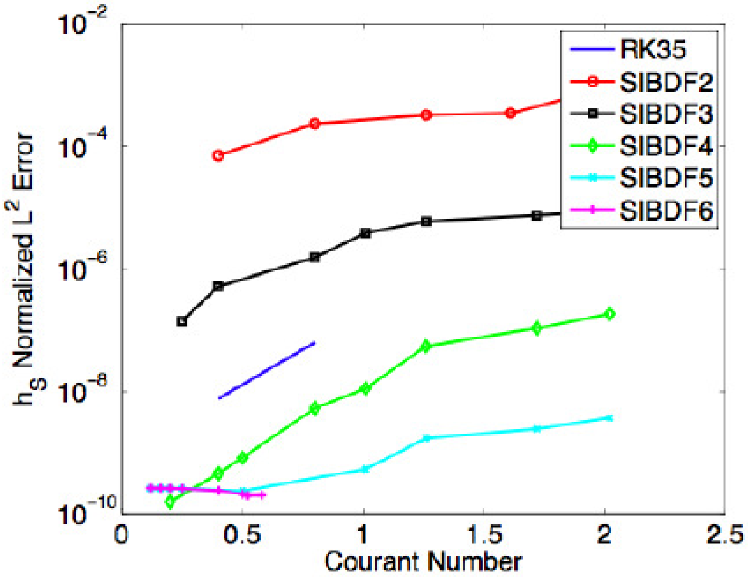
Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua

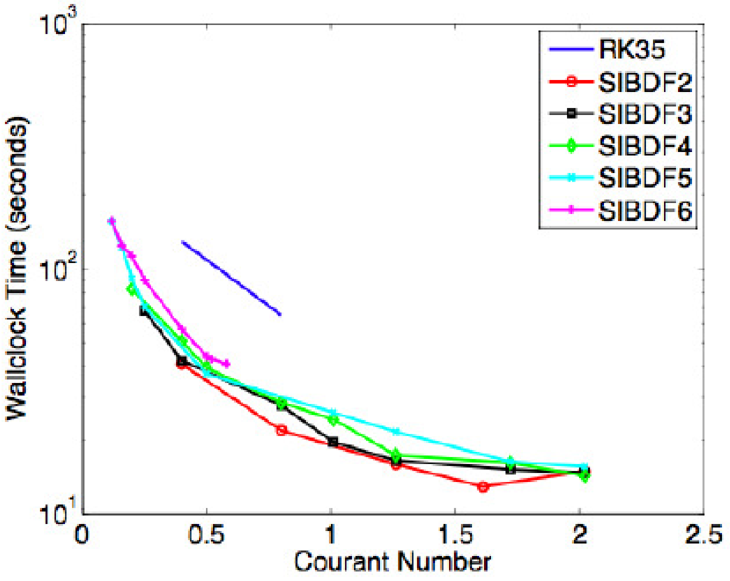
Semi Implicit scheme

Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua
- Good accuracy & efficient performance



Accuracy



Performance

Semi Implicit scheme

Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua
- Good accuracy & efficient performance
- Efficient way to treat acoustic waves

Semi Implicit scheme

Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua
- Good accuracy & efficient performance
- Efficient way to treat acoustic waves
- Best results are obtained for highly oversampled systems (i.e. $1/\Delta t$ is larger than typical small scale signal frequency).

Semi Implicit scheme

Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua
- Good accuracy & efficient performance
- Efficient way to treat acoustic waves
- Best results are obtained for highly oversampled systems (i.e. $1/\Delta t$ is larger than typical small scale signal frequency).

Disadvantages (mostly w.r.t. spectral models)

- Requires solver = non-local solution (scalable iterative solvers are available for GP)

Semi Implicit scheme

Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua
- Good accuracy & efficient performance
- Efficient way to treat acoustic waves
- Best results are obtained for highly oversampled systems (i.e. $1/\Delta t$ is larger than typical small scale signal frequency).

Disadvantages (mostly w.r.t. spectral models)

- Requires solver = non-local solution (scalable iterative solvers are available for GP)
- Stable SI solver with a constant reference state might be a tricky issue $\Rightarrow$ a need to change a prognostic variable might affect conservativeness

Semi Implicit scheme

Advantages (general)

- Allows long timestep - stability limited by stability of the nonlinear residua
- Good accuracy & efficient performance
- Efficient way to treat acoustic waves
- Best results are obtained for highly oversampled systems (i.e. $1/\Delta t$ is larger than typical small scale signal frequency).

Disadvantages (mostly w.r.t. spectral models)

- Requires solver = non-local solution (scalable iterative solvers are available for GP)
- Stable SI solver with a constant reference state might be a tricky issue $\Rightarrow$ a need to change a prognostic variable might affect conservativeness
- Further stabilization implies iterations

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables
- Elegant solution for metric terms over the orography features (Klemp et al. 2003, Girard et al. 2005)

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables
- Elegant solution for metric terms over the orography features (Klemp et al. 2003, Girard et al. 2005)
- Non dispersive, handle well 'threshold bound' quantities

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables
- Elegant solution for metric terms over the orography features (Klemp et al. 2003, Girard et al. 2005)
- Non dispersive, handle well 'threshold bound' quantities
- Apparently no limit from the high resolution (used in DNS, handles well sharp cube obstacles in LES and urban resolving models)

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables
- Elegant solution for metric terms over the orography features (Klemp et al. 2003, Girard et al. 2005)
- Non dispersive, handle well 'threshold bound' quantities
- Apparently no limit from the high resolution (used in DNS, handles well sharp cube obstacles in LES and urban resolving models)
- Gives a chance to control dissipation of the model through the interpolator (truncated splines in MC2, generalized 2^{nd} order interpolator in Aladin, SLHD...)

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables
- Elegant solution for metric terms over the orography features (Klemp et al. 2003, Girard et al. 2005)
- Non dispersive, handle well 'threshold bound' quantities
- Apparently no limit from the high resolution (used in DNS, handles well sharp cube obstacles in LES and urban resolving models)
- Gives a chance to control dissipation of the model through the interpolator (truncated splines in MC2, generalized 2^{nd} order interpolator in Aladin, SLHD...)
- Offers selective filtering of various parameters (but beware of consistency, for example T vs. q)

Semi-Lagrangian scheme - I.

Advantages

- Allows long timestep - Δt is limited by accuracy
- Natural separation of transport and diffusion for advected variables
- Elegant solution for metric terms over the orography features (Klemp et al. 2003, Girard et al. 2005)
- Non dispersive, handle well 'threshold bound' quantities
- Apparently no limit from the high resolution (used in DNS, handles well sharp cube obstacles in LES and urban resolving models)
- Gives a chance to control dissipation of the model through the interpolator (truncated splines in MC2, generalized 2nd order interpolator in Aladin, SLHD...)
- Offers selective filtering of various parameters (but beware of consistency, for example T vs. q)
- In spectral model the interpolation stencil offers natural (and efficient) entry point for 3D physics inclusion

What? Inherent diffusion is of any profit?!

FIG. 1. Potential temperature distribution at $t = 7$ min for an initially circular bubble with a diameter of 500 m and uniform potential temperature excess of 0.5°C over an isentropic environment.

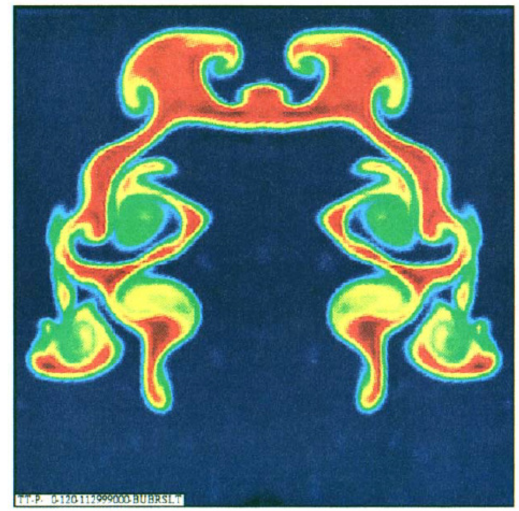


FIG. 2. Same as Fig. 1 but at $t = 10$ min.

”... The results of Smolarkiewicz and Grabowski (= anelastic Eulerian model) are noisy. This is not the case with the proposed model in spite of the fact that it did not use any explicit time filter or diffusion.”

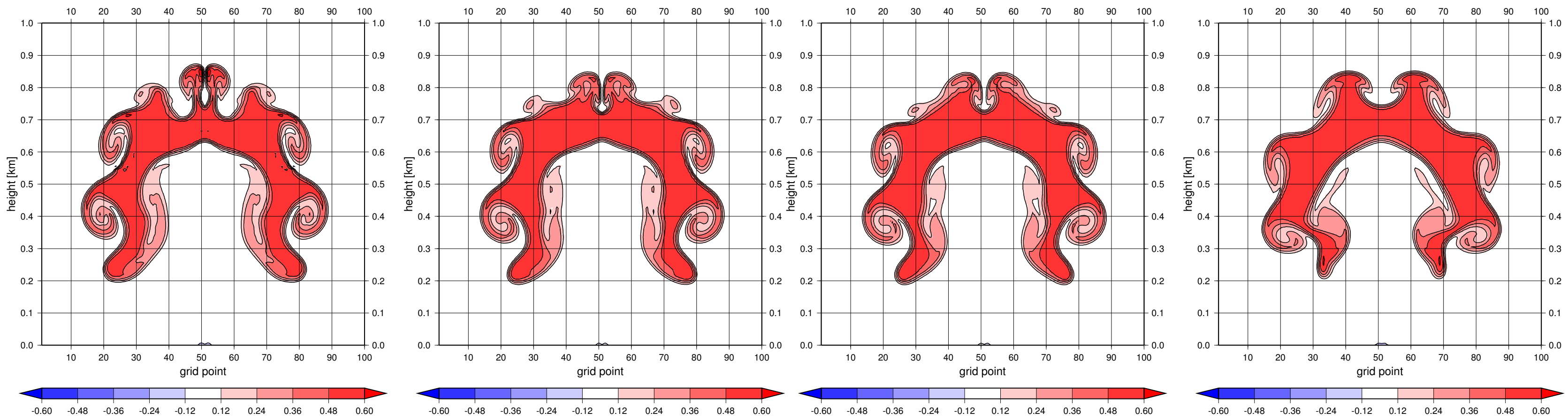
A. Robert (Bubble Convection Experiments with a Semi-implicit Formulation of the Euler Equations, J.A.S., 1993)

Semi-Lagrangian scheme - II.

What? Inherent diffusion is of any profit?!

General interpolator of Aladin

$(\kappa = -2)$ $(\kappa = 0)$ $(\kappa = 1)$ $(\kappa = 6)$



$\Rightarrow$ decreased diffusivity of SL interpolator needs to be compensated by increased horizontal and vertical diffusion. Lagrangian cubic seems to perform extremely well for the real atmosphere.

Semi-Lagrangian scheme - III.

Limitations

- Trajectory research
  - still some ways to improve within the current code
  - higher order iterative methods in spirit of RK

Semi-Lagrangian scheme - III.

Limitations

- Trajectory research
  - still some ways to improve within the current code
  - higher order iterative methods in spirit of RK
- Mass conservation
  - conservative cascade remapping (Nair, Lauritzen, Ullrich, Shashkin, Tolstykh, Zerroukat,...)
    - better than 2^{nd} order convergence
    - one order of magnitude more accurate than Eulerian finite-volume transport schemes
    - the method based on piecewise parabolic distribution of cell averaged density resembles much more expensive PPM (piecewise parabolic methods)

Semi-Lagrangian scheme - III.

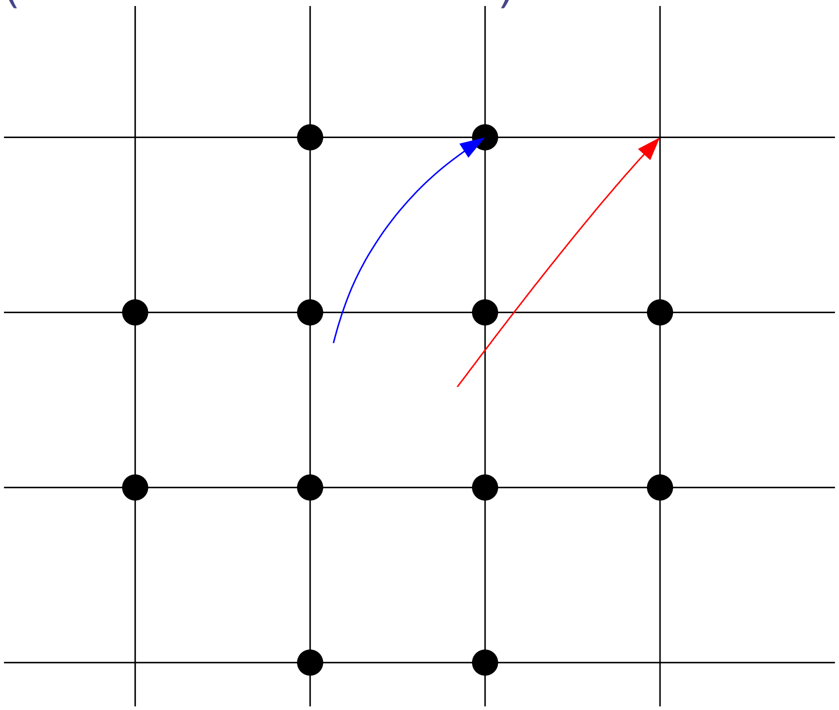
Limitations - cont.

- Communication (MPP)

Semi-Lagrangian scheme - III.

Limitations - cont.

- Communication (MPP)
- Memory conflicts (vector architectures)

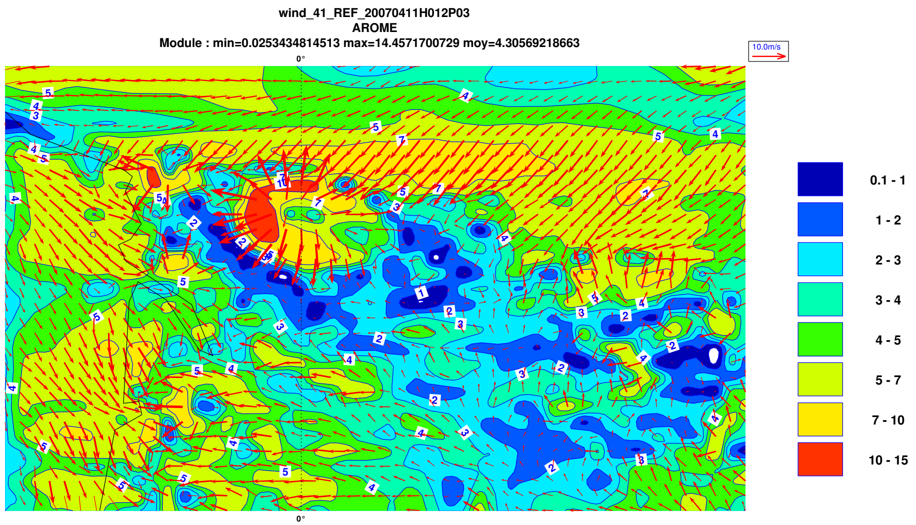


Semi-Lagrangian scheme - III.

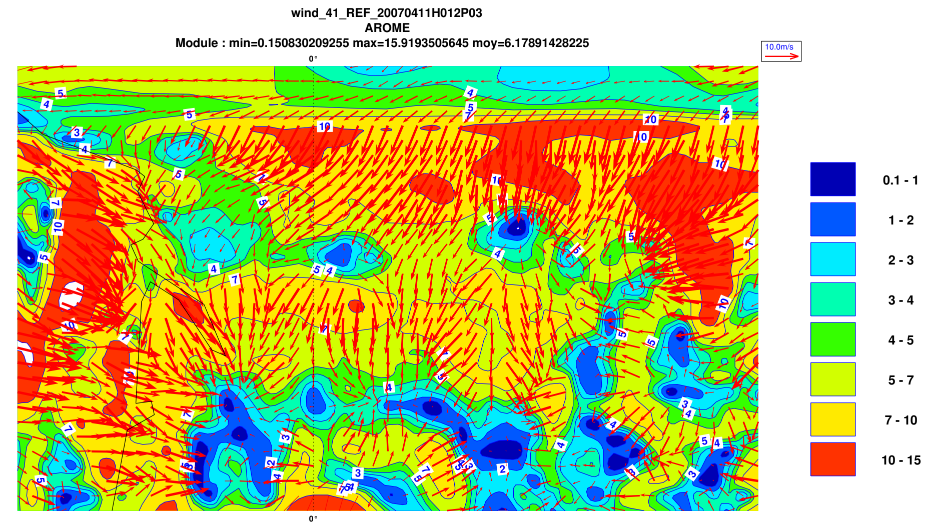
Limitations - cont.

- Communication (MPP)
- Memory conflicts (vector architectures)
- Long timesteps has better chance to destabilize model fields

  Spectral HD is proportional to the field itself (it doesn't care about any atmospheric balance)



full HD

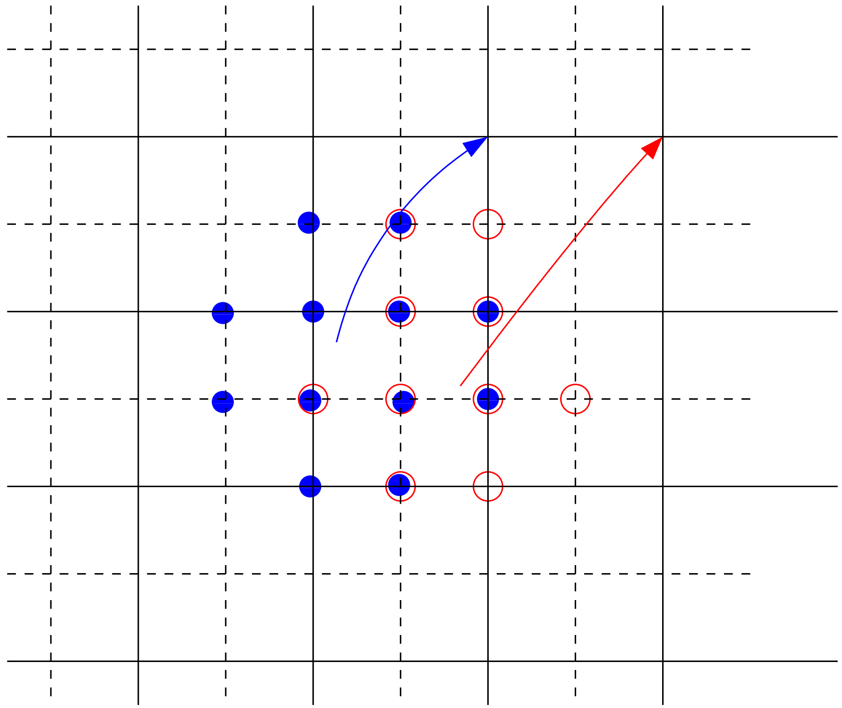


reduced HD

Semi-Lagrangian scheme - III.

P-refinement - a solution for memory conflicts?

Vector machines: memory is not an issue, FFT^{-1} is cheap



Aladin/CE, NEC SX9:

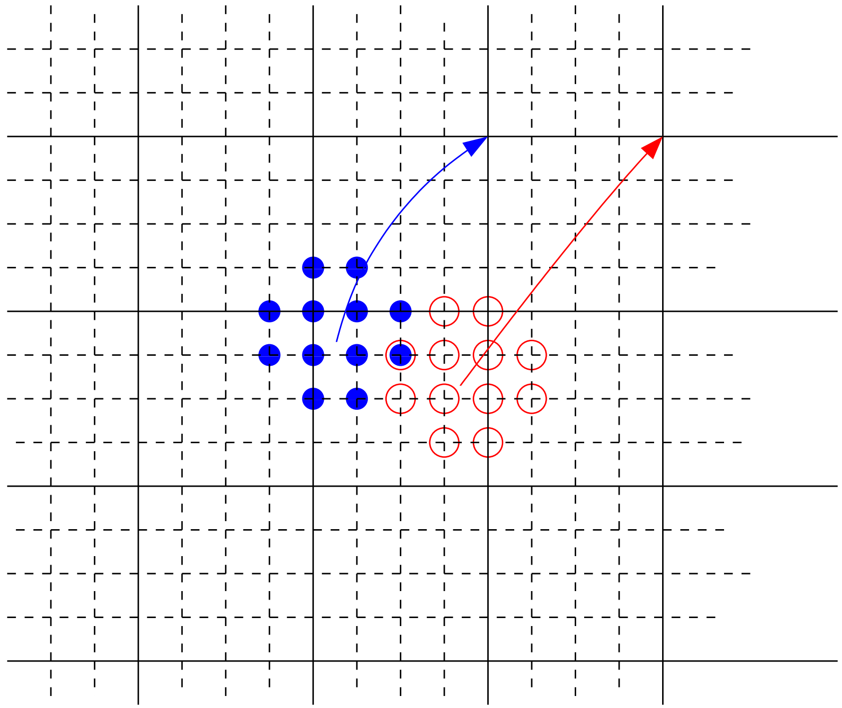
- FFT^{-1} performed to double resolution (+5%)
- memory conflicts reduction by $\approx$ 60-70% (-5%)
- SL comms reduced by up to 50%

better handling of imbalances

Semi-Lagrangian scheme - III.

P-refinement - a solution for memory conflicts?

Vector machines: memory is not an issue, FFT^{-1} is cheap



Aladin/CE, NEC SX9:

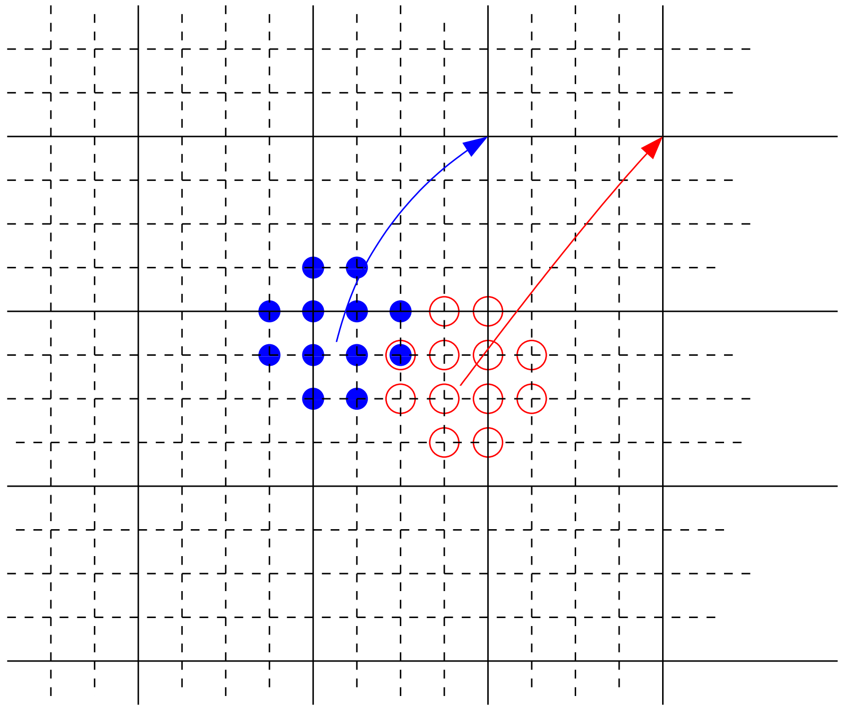
- FFT^{-1} performed to four times higher resolution (+15%)
- nearly no memory conflicts (-8%)
- SL comms reduced by up to 75%

better handling of imbalances

Semi-Lagrangian scheme - III.

P-refinement - a solution for memory conflicts?

Vector machines: memory is not an issue, FFT^{-1} is cheap



Aladin/CE, NEC SX9:

- FFT^{-1} performed to four times higher resolution (+15%)
- nearly no memory conflicts (-8%)
- SL comms reduced by up to 75%

better handling of imbalances
no gain for SX9 (without MPI), but there's a potential...

Spectral methods

Advantages

- Very accurate (until certain extent)
- Efficient SI and spectral HD
- Reduced memory conflicts, easier decomposition along horizontal
- Offers a nice tool for sub-grid modeling of turbulence
- SSDFI

Spectral methods

Advantages

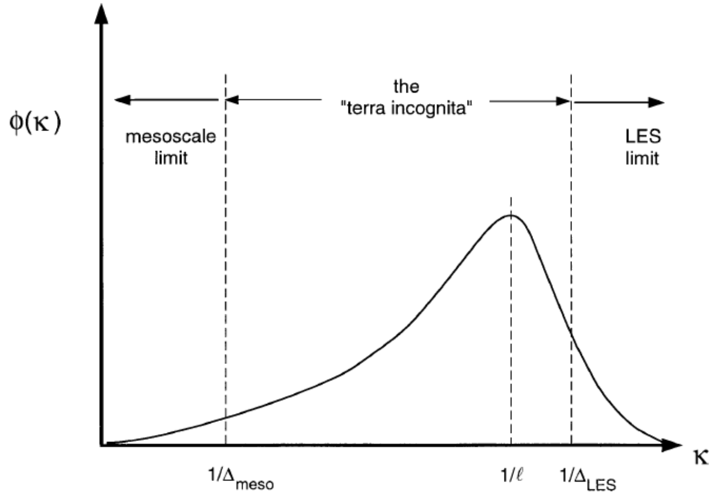
- Very accurate (until certain extent)
- Efficient SI and spectral HD
- Reduced memory conflicts, easier decomposition along horizontal
- Offers a nice tool for sub-grid modeling of turbulence
- SSDFI

Limitations

- Global character, FFTs are expensive for MPP with poor network
- Limitations for the SI background profiles
- More difficult handling of sharp features
- Orography representation needs some special care
- With locally conservative schemes, an extra care is required to derivatives computations

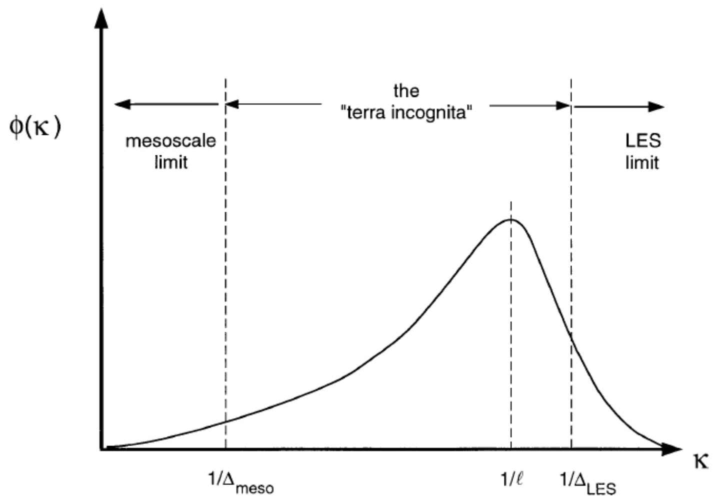
3D turbulence

- Entering the scales of turbulence ($L \approx dx$) we are appearing with our models in the "terra incognita" of turbulence modeling



3D turbulence

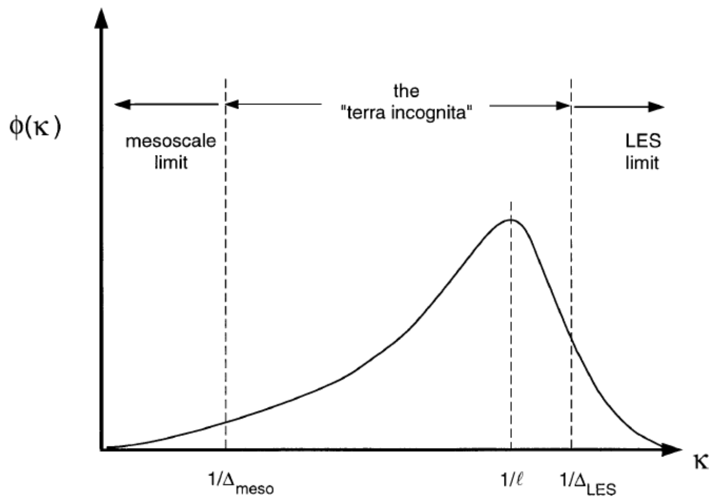
- Entering the scales of turbulence ($L \approx dx$) we are appearing with our models in the "terra incognita" of turbulence modeling



- The 3D interactions of turbulence should be no longer neglected

3D turbulence

- Entering the scales of turbulence ($L \approx dx$) we are appearing with our models in the "terra incognita" of turbulence modeling



- The 3D interactions of turbulence should be no longer neglected
- Problem how to do it in the most optimal way (with respect to the forecast quality and also the cost)

3D turbulence in ALADIN

Options to be considered for horizontal components

(from short till a long-term range)

- Retuned SLHD

  $\Rightarrow$ *SL*

- Extension of TOUCANS based on QNSE

  $\Rightarrow$ *SL*

- Previous combined with (Lagrangian-averaged) dynamic model

  $\Rightarrow$ *spectral methods, SL*

- Full (Lagrangian-averaged) dynamic model

  $\Rightarrow$ *spectral methods, (SL)*

3D turbulence in ALADIN

Options to be considered for horizontal components

(from short till a long-term range)

- Retuned SLHD

  ⇒ *SL*

- Extension of TOUCANS based on QNSE

  ⇒ *SL*

- Previous combined with (Lagrangian-averaged) dynamic model

  ⇒ *spectral methods, SL*

- Full (Lagrangian-averaged) dynamic model

  ⇒ *spectral methods, (SL)*

⇒ Trajectory computations and spectral methods are becoming increasingly popular in turbulence modeling.

SLHD

Discretized 2TL SI SL equation for Ψ^+ reads:

$$\Psi_F^+ = \left(1 - \frac{\Delta t}{2}\mathcal{L}\right)^{-1} \left[\underbrace{\left(1 + \frac{\Delta t}{2}\mathcal{L}\right)\Psi_O^0 + \Delta t \mathcal{F}_O^0}_{I} + \underbrace{\frac{\Delta t}{2}\mathcal{N}_O^*}_{I_L} + \frac{\Delta t}{2}\mathcal{N}_F^* \right]$$

The high order interpolator I weights are evaluated as:

$$\begin{pmatrix} \tilde{w}_1 \\ \tilde{w}_2 \\ \tilde{w}_3 \end{pmatrix} = \begin{pmatrix} 1-2\varepsilon & \varepsilon & 0 \\ \varepsilon & 1-2\varepsilon & 0 \\ 0 & \varepsilon & 1 \end{pmatrix} \cdot \begin{pmatrix} w_{lag,1} + \kappa(w_{quad,1} - w_{lag,1}) \\ w_{lag,2} + \kappa(w_{quad,2} - w_{lag,2}) \\ w_{lag,3} + \kappa(w_{quad,3} - w_{lag,3}) \end{pmatrix},$$

SLHD is introduced when κ and ε are defined as:

$$\kappa = \kappa_{min} + (\kappa_{max} - \kappa_{min}) \frac{\Delta t F(d, D_2)}{1 + \Delta t F(d, D_2)}, \qquad \varepsilon = \varepsilon_{H,V} \frac{\Delta t F(d, D_2)}{1 + \Delta t F(d, D_2)}$$

3D extension of TOUCANS

$$\frac{\partial \Psi}{\partial t} + ... = \boxed{-K_H \frac{\partial^2 \Psi}{\partial x^2} - K_H \frac{\partial^2 \Psi}{\partial y^2}} - \frac{\partial}{\partial z}\left(K_V \frac{\partial \Psi}{\partial z}\right) - K_{Num}\mathcal{D}(\Psi)$$

3D extension of TOUCANS

$$\frac{\partial \Psi}{\partial t} + ... = -K_H \frac{\partial^2 \Psi}{\partial x^2} - K_H \frac{\partial^2 \Psi}{\partial y^2} - \frac{\partial}{\partial z}\left(K_V \frac{\partial \Psi}{\partial z}\right) - K_{Num}\mathcal{D}(\Psi)$$

where

- $K_H = K_H(x, y, z, t)$ but we assume $\nabla_H K_H \cdot \nabla_H \Psi = 0$
- $\nabla_H^2 \Psi$ is evaluated by the GP Laplacian of SLHD
- $K_V \neq K_H$
- K_V and K_H are derived in a consistent way with TOUCANS (emulating QNSE):

$$\begin{aligned} K_{m,V} &= L_K C_K \sqrt{e}\chi_3(Ri) \\ K_{h,V} &= L_K C_K C_3 \sqrt{e}\phi_3(Ri) \end{aligned} \Rightarrow \begin{aligned} K_{m,H} &= L_K^H C_K \sqrt{e}\chi_H(Ri) \\ K_{h,H} &= L_K^H C_K C_3 \sqrt{e}\phi_H(Ri) \end{aligned}$$

Dynamic model - an introduction

Germano identity

- Spacial coarse graining of a field $q(\mathbf{x}, t)$ at a scale Δx is denoted as the convolution $G_{\Delta x} * q = \overline{q}$.

Dynamic model - an introduction

Germano identity

- Spacial coarse graining of a field $q(\mathbf{x},t)$ at a scale Δx is denoted as the convolution $G_{\Delta x} * q = \overline{q}$.
- A subgrid modeling of non-linear term $\mathcal{N}(q)$ of a variable $q(\mathbf{x},t)$ requires models for the residual $\overline{\mathcal{N}(q)} - \mathcal{N}(\overline{q})$.

Dynamic model - an introduction

Germano identity

- Spacial coarse graining of a field $q(\mathbf{x}, t)$ at a scale Δx is denoted as the convolution $G_{\Delta x} * q = \overline{q}$.
- A subgrid modeling of non-linear term $\mathcal{N}(q)$ of a variable $q(\mathbf{x}, t)$ requires models for the residual $\overline{\mathcal{N}(q)} - \mathcal{N}(\overline{q})$.
- Additional test-filtering operation is introduced at scales $\alpha \Delta x$ with $\alpha > 1$, denoted by a convolution $G_{\alpha\Delta x} * q = \widehat{q}$.

Dynamic model - an introduction

Germano identity

- Spacial coarse graining of a field $q(\mathbf{x}, t)$ at a scale Δx is denoted as the convolution $G_{\Delta x} * q = \overline{q}$.
- A subgrid modeling of non-linear term $\mathcal{N}(q)$ of a variable $q(\mathbf{x}, t)$ requires models for the residual $\overline{\mathcal{N}(q)} - \mathcal{N}(\overline{q})$.
- Additional test-filtering operation is introduced at scales $\alpha\Delta x$ with $\alpha > 1$, denoted by a convolution $G_{\alpha\Delta x} * q = \widehat{q}$.
- The Germano identity than simply reads:

$$\widehat{\overline{\mathcal{N}(q)}} - \mathcal{N}(\widehat{\overline{q}}) = \left[\widehat{\overline{\mathcal{N}(q)}} - \widehat{\mathcal{N}(\overline{q})}\right] + \left[\widehat{\mathcal{N}(\overline{q})} - \mathcal{N}(\widehat{\overline{q}})\right].$$

Dynamic model in Aladin - I.

- Spectral filters with sharp cutoff are ideally suited for the filtering (Gaussian filter: $\alpha' = \sqrt{\alpha^2 + 1}$). When allowing different truncations in the model we can have $\overline{(...)}$ and $\widehat{(...)}$ operators; usually $\alpha = 2$.

Dynamic model in Aladin - I.

- Spectral filters with sharp cutoff are ideally suited for the filtering (Gaussian filter: $\alpha' = \sqrt{\alpha^2 + 1}$). When allowing different truncations in the model we can have $\overline{(...)}$ and $\widehat{(...)}$ operators; usually $\alpha = 2$.
- Germano identity for SGS stress tensor: $T_{ij} = \widehat{\tau_{ij}} + L_{ij}$

Dynamic model in Aladin - I.

- Spectral filters with sharp cutoff are ideally suited for the filtering (Gaussian filter: $\alpha' = \sqrt{\alpha^2 + 1}$). When allowing different truncations in the model we can have $\overline{(...)}$ and $\widehat{(...)}$ operators; usually $\alpha = 2$.
- Germano identity for SGS stress tensor: $T_{ij} = \widehat{\tau_{ij}} + L_{ij}$
- Previous applied to TOUCANS-3D leads (for example) to:

$$T_i = L_K^{\alpha H} C_K \chi(\widehat{Ri}) \sqrt{\widehat{e}} \widehat{\frac{\partial u_i}{\partial x_i}}, \quad \tau_i = L_K^H C_K \chi(Ri) \sqrt{e} \frac{\partial u_i}{\partial x_i}$$

$$L_i = \widehat{u_i^2} - (\widehat{u_i})^2, \quad i = 1, 2$$

Dynamic model in Aladin - I.

- Spectral filters with sharp cutoff are ideally suited for the filtering (Gaussian filter: $\alpha' = \sqrt{\alpha^2 + 1}$). When allowing different truncations in the model we can have $\overline{(...)}$ and $\widehat{(...)}$ operators; usually $\alpha = 2$.
- Germano identity for SGS stress tensor: $T_{ij} = \widehat{\tau_{ij}} + L_{ij}$
- Previous applied to TOUCANS-3D leads (for example) to:

$$T_i = L_K^{\alpha H} C_K \chi(\widehat{Ri}) \sqrt{\widehat{e}} \widehat{\frac{\partial u_i}{\partial x_i}}, \quad \tau_i = L_K^H C_K \chi(Ri) \sqrt{e} \frac{\partial u_i}{\partial x_i}$$

$$L_i = \widehat{u_i^2} - (\widehat{u_i})^2, \quad i = 1, 2$$

- Optionally $\widehat{\tau_i}$ can be evaluated in the SL stencil (to ease dataflow and to save some CPU)

Dynamic model in Aladin - II.

- Minimizing the $\mathcal{E} = [L_1 - (T_1 - \tau_1)]^2 + [L_2 - (T_2 - \tau_2)]^2$ functional a correction $C'_K = \xi C_K$ is obtained.

Dynamic model in Aladin - II.

- Minimizing the $\mathcal{E} = [L_1 - (T_1 - \tau_1)]^2 + [L_2 - (T_2 - \tau_2)]^2$ functional a correction $C'_K = \xi C_K$ is obtained.
- It has been proved, that much better (i.e. more robust and accurate) results can be obtained when $\mathcal{E}$ is weighted along trajectory.

Dynamic model in Aladin - II.

- Minimizing the $\mathcal{E} = [L_1 - (T_1 - \tau_1)]^2 + [L_2 - (T_2 - \tau_2)]^2$ functional a correction $C'_K = \xi C_K$ is obtained.
- It has been proved, that much better (i.e. more robust and accurate) results can be obtained when $\mathcal{E}$ is weighted along trajectory.
- Applying the correction ξ consistently for the whole TOUCANS-3D, the model is fully defined.

Dynamic model in Aladin - II.

- Minimizing the $\mathcal{E} = [L_1 - (T_1 - \tau_1)]^2 + [L_2 - (T_2 - \tau_2)]^2$ functional a correction $C'_K = \xi C_K$ is obtained.
- It has been proved, that much better (i.e. more robust and accurate) results can be obtained when $\mathcal{E}$ is weighted along trajectory.
- Applying the correction ξ consistently for the whole TOUCANS-3D, the model is fully defined.
- Alternatively, the dynamic method can be applied independently to every group of diffused variables.

Dynamic model in Aladin - II.

- Minimizing the $\mathcal{E} = [L_1 - (T_1 - \tau_1)]^2 + [L_2 - (T_2 - \tau_2)]^2$ functional a correction $C'_K = \xi C_K$ is obtained.
- It has been proved, that much better (i.e. more robust and accurate) results can be obtained when $\mathcal{E}$ is weighted along trajectory.
- Applying the correction ξ consistently for the whole TOUCANS-3D, the model is fully defined.
- Alternatively, the dynamic method can be applied independently to every group of diffused variables.
- Evaluating the $\hat{\tau}$ by spectral space, one would have a full freedom to evaluate any $\frac{\partial \tau_{ij}}{\partial x_i}$ at the same time.
  $\Rightarrow$ Full LES sophistication starts to be available.

Open questions

- Are we ready to accept some degradation of results with respect to better scalability of the methods?

Open questions

- Are we ready to accept some degradation of results with respect to better scalability of the methods?
- What are the expectations for the mass conservative schemes with respect to the physics and LBC? (Are we ready to move into those methods, or rather support some optional switch to activate them?)

Open questions

- Are we ready to accept some degradation of results with respect to better scalability of the methods?
- What are the expectations for the mass conservative schemes with respect to the physics and LBC? (Are we ready to move into those methods, or rather support some optional switch to activate them?)
- Knowing that in a long term future we would like to proximate the LES realm (having 3D Poisson iterative solvers for pressure, spectral methods for filtering and Lagrangian methods for spacial averaging), should we take the risk and to abandon those methods meanwhile?

Open questions

- Are we ready to accept some degradation of results with respect to better scalability of the methods?
- What are the expectations for the mass conservative schemes with respect to the physics and LBC? (Are we ready to move into those methods, or rather support some optional switch to activate them?)
- Knowing that in a long term future we would like to proximate the LES realm (having 3D Poisson iterative solvers for pressure, spectral methods for filtering and Lagrangian methods for spacial averaging), should we take the risk and to abandon those methods meanwhile?
- Why to prepare for a potential use of o(10^5-10^6) cores (with a commodity network only), when this represents relatively extreme case? Are the "advocates" of those MPP solutions aware about the demands (cost, electricity consumption, size,...) that parallel architectures create?

Open questions

- Are we ready to accept some degradation of results with respect to better scalability of the methods?
- What are the expectations for the mass conservative schemes with respect to the physics and LBC? (Are we ready to move into those methods, or rather support some optional switch to activate them?)
- Knowing that in a long term future we would like to proximate the LES realm (having 3D Poisson iterative solvers for pressure, spectral methods for filtering and Lagrangian methods for spacial averaging), should we take the risk and to abandon those methods meanwhile?
- Why to prepare for a potential use of o(10^5-10^6) cores (with a commodity network only), when this represents relatively extreme case? Are the "advocates" of those MPP solutions aware about the demands (cost, electricity consumption, size,...) that parallel architectures create?
- What is the optimal solution with respect to the future computers? Going to o(10^5) is the only one side of the problem. There are other relatively seldom discussed views: Cray's new GEMINI switch, SW features like SHMEM (shared memory access library) or UPC (Unified Parallel C). In contrary, NWP people are frequently mentioning the potential of GPU. But to profit form it would require to the solve the problem of data transfer in/out to GPU.

And some remarks...

- Beware of academic tests results also with respect to the computational efficiency. The whole solutions should be always benchmarked. (For example we don't see any communication problem of SL in Arome with o(10^3) cores, as most of the communications observed are coming from the poor Surfex I/O method there).

And some remarks...

- Beware of academic tests results also with respect to the computational efficiency. The whole solutions should be always benchmarked. (For example we don't see any communication problem of SL in Arome with o(10^3) cores, as most of the communications observed are coming from the poor Surfex I/O method there).
- Non-oscillatory methods (like WENO and PPM) have small real part in their spectra to fight with oscillations. They would require RK3 timestepping as simple leapfrog would be unstable with them. (Complexity in time vs. complexity in space)

And some remarks...

- Beware of academic tests results also with respect to the computational efficiency. The whole solutions should be always benchmarked. (For example we don't see any communication problem of SL in Arome with o(10^3) cores, as most of the communications observed are coming from the poor Surfex I/O method there).
- Non-oscillatory methods (like WENO and PPM) have small real part in their spectra to fight with oscillations. They would require RK3 timestepping as simple leapfrog would be unstable with them. (Complexity in time vs. complexity in space)
- CFD offers plenty of methods, but not all of them are usable for NWP kind of problem (i.e. mode with many repetitive timesteps).

And some remarks...

- Beware of academic tests results also with respect to the computational efficiency. The whole solutions should be always benchmarked. (For example we don't see any communication problem of SL in Arome with o(10^3) cores, as most of the communications observed are coming from the poor Surfex I/O method there).
- Non-oscillatory methods (like WENO and PPM) have small real part in their spectra to fight with oscillations. They would require RK3 timestepping as simple leapfrog would be unstable with them. (Complexity in time vs. complexity in space)
- CFD offers plenty of methods, but not all of them are usable for NWP kind of problem (i.e. mode with many repetitive timesteps).
- Any solution aiming at rather long and homogeneous (i.e. without splitting) time-steps requires a solver; the way to solve becomes technical but the argument in favor of going explicit falls as long as one can manage relatively efficient solvers.

And some remarks...

- Beware of academic tests results also with respect to the computational efficiency. The whole solutions should be always benchmarked. (For example we don't see any communication problem of SL in Arome with o(10^3) cores, as most of the communications observed are coming from the poor Surfex I/O method there).
- Non-oscillatory methods (like WENO and PPM) have small real part in their spectra to fight with oscillations. They would require RK3 timestepping as simple leapfrog would be unstable with them. (Complexity in time vs. complexity in space)
- CFD offers plenty of methods, but not all of them are usable for NWP kind of problem (i.e. mode with many repetitive timesteps).
- Any solution aiming at rather long and homogeneous (i.e. without splitting) time-steps requires a solver; the way to solve becomes technical but the argument in favor of going explicit falls as long as one can manage relatively efficient solvers.
- There has been much work on constructing high-order spatial discretization methods but not as much effort has been spent on implicit time-integration. There might be a potential to obtain high-accuracy yet efficient time-integrators.