15th ALADIN Workshop

Some Interfacing Issues

Bart Catry (ALADIN-2)

Overview

This time, no equations... (or hardly any)
→ a more *consensual* talk about:

- The Barycentric choice (rule B)
- Code changes
- Flux vs tendency (rule D)
- Diagnostics (rules C)
- Possible Extensions (rules E)

TCWGPDI

One of the goals of a common Physics-Dynamics-Interface (PDI) for ALADIN/ALARO/AROME/HIRLAM is that one can "play" with physical parameterizations almost at leisure

This puts some constraints on

- the PDI
- the low level parameterizations
- the diagnostics

These constraints were illustrated and discussed on the TCWGPDI, held in Prague (November 2004)

A document containing the necessary rules (rule A – rules E) was distributed as widely as possible (ALADIN, AROME, Meso-NH and HIRLAM communities)

Equations

In order to have a common PDI one needs a set of governing diabatic equations at the interface level and any inconsistency with these equations has to be corrected at the lower level (i.e. the parameterizations)

Following the choice of AROME to develop a barycentric set of basic equations, it was found that this barycentric option makes it possible to find a general set of flux-conservative equations for the following prognostic quantities: *u, v, w, p, T, TKE* (?)

These equations were derived from first principles (under an agreed set of basic hypotheses) independently of the choice of the dynamical core they are applied to when discretisation takes place

Equations (2)

Also a number of existing specific options (e.g. NDPSFI, LCONDWT) can be easily treated

Moreover, a new option has been proposed for the compressible case: the possibility to project the total heat source on both temperature and pressure changes (a switch termed [T,p] for now)

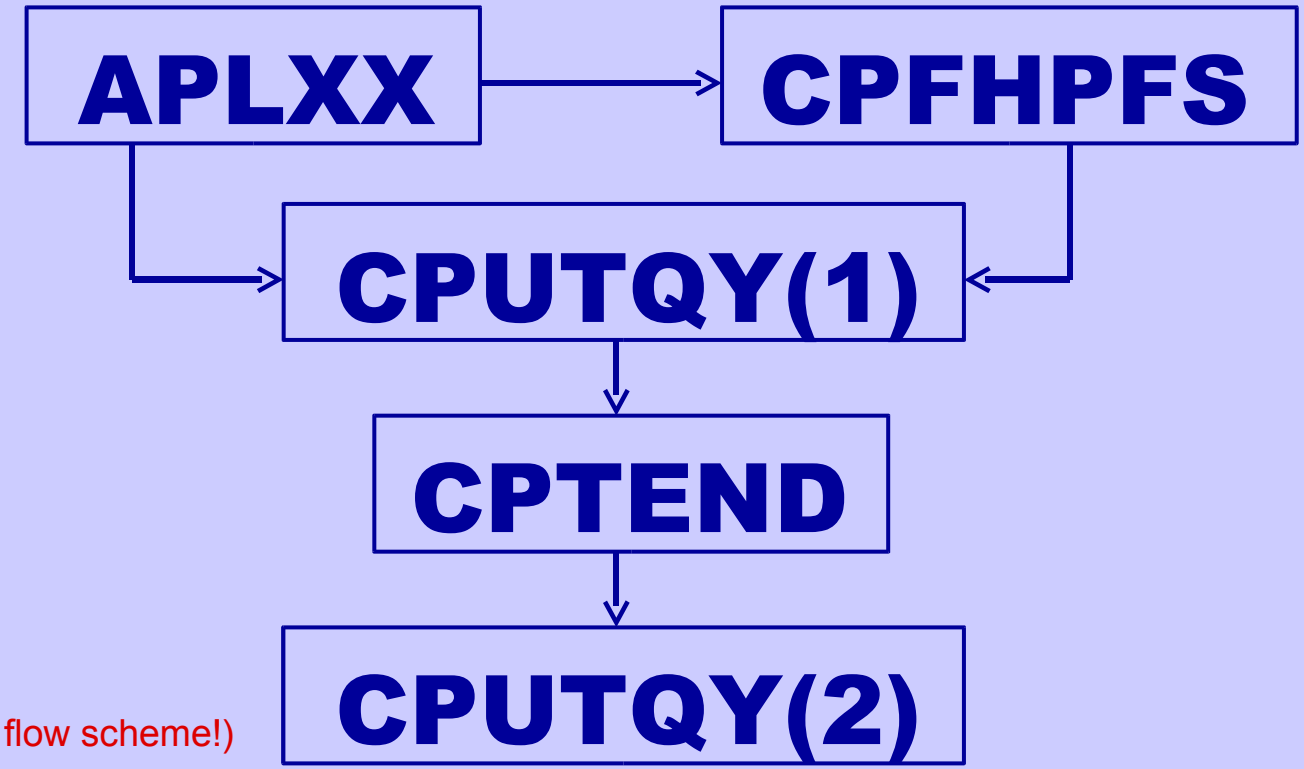
This switch allows us to use one of the two sets below:

$$\begin{cases} c_p \dfrac{dT}{dt} - RT \dfrac{d\ln(p)}{dt} = Q \\ c_v \dfrac{d\ln(p)}{dt} + c_p D_3 = 0 \end{cases}$$

$$\begin{cases} c_v \dfrac{dT}{dt} + RT \cdot D_3 = Q + T \dfrac{dR}{dt} \\ c_v \dfrac{d\ln(p)}{dt} + c_p D_3 = \dfrac{Q}{T} + \dfrac{c_p}{R} \dfrac{dR}{dt} \end{cases}$$

Code Changes

Apart from rewriting the core of the PDI (speaking in "ALADIN routines" this means CPFHPFS, CPTEND, CPUTQY and APLPAR), only marginal changes are required in the low level routines



(this is not yet a flow scheme!)

Code Changes (2)

- Only changes in the input/output stream will affect low level routines and for some physical parameterizations also some additional diagnostics output will be needed
- The ALADIN switch NDPSFI (δ_m) disappears from low-level physical routines and appears only at interfacing level and in some dynamical routines. Even better, the use of NDPSFI in dynamics has also become more transparent
- The ALADIN switch LCONDWT will be incorporated in a larger system of choosing the number of prognostic species. Actually, the number of species and associated mass fluxes will always be the same (as seen by the PDI, but see next slide)

Pseudo Fluxes

The thermodynamic equation which was found to obey rule B can be written in a flux conservative form:

$$\frac{\partial}{\partial t}\left(c_p T\right) = -g\frac{\partial}{\partial p}\left[F\right]$$

As it is more physical to sum fluxes than tendencies at the interfacing level, we choose to work further in this spirit

The full interface is of course as general as possible, including all possible species and processes. Since currently no model is that sophisticated, pseudo-fluxes will need to be created (as scientific as possible) to stay as close as possible to the previous solution

The production of these pseudo-fluxes can be inside the existing routines (HIRLAM) or inside some intermediate routines (Meso-NH)

Flux vs Tendency

All this has as consequence that the flux vs tendency dilemma has become a technical issue where one uses dummy-fluxes to compensate for any lack of information

Codewise this means a gradual transition of APLPAR (or APLAROME) to a general parameterization calling sequence APLXX

Diagnostics

Since the basis of the flux-tendency conversion will be the DDH-type of diagnostics it is of utmost importance that there exists a good bijection between the output fluxes of the physical routines and these diagnostics

This is of course pertinent with respect to the use of the pseudo fluxes

If this condition is fulfilled, we should have meaningful diagnostics which are cross-model comparable

Possible Extensions

Let us repeat the spirit of "rules E" where one makes the following distinction:

a process is a physical phenomenon the model may or may not describe. There should exist a transparent structure to know where and when a certain process is treated (to avoid a double treatment) (e.g. a convection scheme with precipitation release cannot be combined with a general precipitation scheme)

a scheme should be a list of the to be treated processes in the parameterization (e.g. a condensation scheme with or without precipitation release)

an off-line controlling mechanism is required

Possible Extensions (2)

The formalism of processes and schemes is easily integrated in the proposed interfacing strategy (as long as we keep in mind the DDH constraints):

(1) At low level the input/output stream needed to treat a certain physical phenomenon should remain the same (to serve as input in the PDI and DDH)

(2) At interfacing level we should still obey "rules A+D", meaning that for instance when one switches between parallel or sequential physics, a change in input (defined via the statuses) will be required

Apart from this, only the way of calling the different parameterizations (or processes/schemes if you want) will need to be changed

Possible Extensions (3)

Since it is anyhow necessary to list the different processes to check to consistency of the PDI, this list can be the starting point of a structure to define the physics configuration to be used in a given run of the model

The addition of new species (e.g. graupel and hail) can be done with minor code changes: as the ratios of the species will be treated in a kind of array, one just needs to add some additional fields and equation wise graupel and hail are the same as snow (at interface level)

Any other suggestions ?