

# DYNAMICS SETUP AND MODULES: PROPOSAL OF REORGANISATION.

YESSAD Karim.

June 10, 2013

Version 2.

# 1 Introduction and purpose.

In the early stages of ARPEGE/IFS, variables controlling the adiabatic model (dynamics) have been put in YOMDYN/NAMDYN/SUDYN. When it was not possible to wait for SUDYN to compute them, for example because they were used in SUALLO to allocate arrays, these variables were put in YOMCT0/NAMCT0/SUCT0. But YOMCT0 must in theory contain 0-level control variables, not adiabatic dynamics ones.

Developments done on ARPEGE/IFS have lead to create some other modules like YOMDYNA, YOMCVER, YOMDYNCORE and more recently YOMDYNDIFF.

My feeling is that the current repartition of variables, in particular between YOMCT0, YOMDYNA and YOMDYN is completely outdated, and will create issues for the current OOPS reorganisation and for future developments. Additionally, we must notice that YOMCT0 and YOMDYN are too big: each time we modify these modules we must recompile nearly all the code.

Some work has started to move variables between CY38R1 and CY40, but this is a minimal amount of work compared to what is desirable to be done.

In this paper I will do a study about a different repartition of dynamics variables between different modules. The work proposed can be seen as a work complementary to the current OOPS setup reorganisation. It can be spread among several cycles.

## 2 Shortcomings of the current module repartition.

I see at least several shortcomings:

- YOMDYN has a lot of variables and is used in a huge number of routines; each time we modify YOMDYN we have to recompile most of the code.
- The same remark is valid for YOMCT0 too, which has become with the time a sort of “all purpose module” containing things which have nothing to do in this module.
- The set-up of GMV (SETUP\_GMV) cannot be called in SU0YOMA (with SUGFL and SU\_SURF\_FLDS) because it uses some calculations still done in SUDYN. Moving the call to SETUP\_GMV just after the call to SUGFL requires to move the following variables from YOMDYN/NAMDYN/SUDYN to something computed under SUDYNA: LSLINL, LSLINLC1, LSLINLC2, LSETTLS, LSETTLST.
- YOMDYN mixes geometry-dependent variables, and also a lot of not geometry-dependent variables which could be set-up at the level of SUDYNA.
- YOMDYN mixes a lot of topics: semi-Lagrangian scheme, semi-implicit scheme, horizontal diffusion scheme and some other things.
- YOMCT0 still contains some purely dynamics variables which must be moved in YOMDYNA or in other new modules set-up at the level of SUDYNA.
- SUDYN is difficult to read and requires a lot of testings to ensure that we do not put inconsistent values in namelists.
- There is no separate module for not geometry-dependent variables used in the semi-Lagrangian scheme only.
- There are too many testings and it would better to replace some of them by resettings.

## 3 Proposal of new repartition of adiabatic dynamics variables.

### 3.1 Introduction.

The basic idea is to have:

- Only LSLAG, LTWOTL, LNHDYN will be left in YOMCT0.
- Additionally, TSTEP will be moved from YOMDYN to YOMCT0 (during work done for command-line removal); an alternate method is to keep TSTEP in YOMDYN (but removed from NAMDYN) and to create a new variable TSTEP0 in YOMCT0/NAMCT0.
- Not geometry-dependent variables will go in YOMDYNA or in other modules set-up under SUDYNA.
- Only geometry-dependent variables must remain in YOMDYN.
- More modules, but with a reasonable size.

## 3.2 More details.

- YOMCT0: keep LSLAG, LTWOTL, LNHDYN; put TSTEP (currently in YOMDYN) or create TSTEP0.
- Variables which must be set-up under SUDYNA:
  - ICI scheme (to be moved in YOMDYNICI/NAMDYNICI/SUDYNICI)
    - \* currently in YOMCT0/SUCT0/NAMCT0: LPC\_FULL, LPC\_CHEAP, LPC\_NESC, LPC\_NESCT.
    - \* currently in YOMDYN/SUDYN/NAMDYN: NSITER, NCURRENT\_ITER, LRHDLASTITERPC.
  - DYNCORE variables (YOMDYNCORE/NAMDYNCORE/SUDYNCORE): current code can remain unchanged.
  - VFE variables (YOMCVER/NAMCVER/SUCVER): current code can remain unchanged.
  - not geometry-dependent semi-Lagrangian variables, which are currently in YOMDYN/SUDYN/NAMDYN (to be moved in YOMDYNL/NAMDYNL/SUDYNL):
    - \* LSETTLS and LSETTST.
    - \* LSLINLC1, LSLINLC2, LSLINL.
    - \* LELTRA; but this option coded for research and no longer used in operational models could be pruned.
    - \* N[X]LAG.
    - \* NITMP.
    - \* VETAON and VETAOX.
    - \* LADVF, LADVFW, L2TLFF, RW2TLFF.
    - \* VESL and XIDT.
    - \* LQM.. and LQMH... for monotonic interpolations.
    - \* RCMSLP0.
    - \* NSPLTHOI, LSPLTHOIGFL, NSLDIMK.
  - other variables (to be moved in YOMDYNA/NAMDYNA/SUDYNA):
    - \* upper radiative condition: LRUBC (currently in YOMCT0/NAMCT0/SUCT0); RTEMRB and NRUBC (currently in YOMDYN/SUDYN/NAMDYN).
    - \* LRFRIC (currently in YOMCT0/NAMCT0/SUCT0); goes with LRFRICISOTR.
    - \* LVERCOR (currently in YOMCT0/NAMCT0/SUCT0); goes with LRWSDL.. keys.
    - \* VCPR, VCTR, VCAK (currently in YOMDYN/SUDYN/NAMDYN); goes with LVERCOR. But in this case VCPR must be setup with 101325 and not YRVAB%VP00 which is not yet known in SUDYNA.
    - \* Asselin timefiltering constants REPS1 to REPSP1 (currently in YOMDYN/SUDYN/NAMDYN).
    - \* LDRY\_ECMWF (currently in YOMDYN/SUDYN/NAMDYN). This variable must be renamed (I suggest LRINC\_RCPDRY).
    - \* NCOMP\_CVGQ (currently in YOMDYN/SUDYN/NAMDYN); moving this variable would enable to make it available in SUGFL1 when reading NAMGFL.
- Variables which must be set-up under SUDYN:
  - all those remaining in YOMDYN.
  - YOMDYNDIFF variables (excepted for LGRADSP).

But there are calculations or allocations which are currently done in SUDYN and which have no relationship with YOMDYN variables:

- NSMAX\_TRAJ, NSMAX\_BACKGR, TSTEP\_TRAJ. In particular TSTEP\_TRAJ could be computed earlier in the set-up if the timestep is known in SUCT0.
- array FTHSOR for ISP diagnostics (must go in 3.4 of SUSC2B, with allocation of GFUBUF and XFUBUF).

A better place must be found to set-up or allocate them.

- Other open issues:
  - better place to find for LREGETA and LVFE\_REGETA, currently in YOMCT0: maybe in a future module containing vertical geometry; goes with YRVETA.
  - forcings (variables spread among YOMCT0 and YOMLSFORC).
    - \* SULSFORC could be called earlier in the code (just after SUGFL) provided we move TSTEP from YOMDYN to YOMCT0.
    - \* LSFORC and LSFORCS could go in YOMDYNA/NAMDYNA/SUDYNA.

- LIMPF (YOMDYNA or YOMDYN?). Keeping it in YOMDYN allows to keep the test forbidding to use LIMPF=T with stretched/tilted models. So the set-up of this variable is partly geometry-dependent.
- LGRADSP: it is better to store it in YOMDYNA (current place in NAMDYNA and setup in SUDYNA is OK).
- TDT: where to put it and set-up it?
- YOMVSEP variables (NVSEPC and NVSEPL): best place to set-up them?
- YOMMASS variables (LMASDRY, LMASCOR, GMASSI): best place to set-up them?
- The final YOMDYN could be splitted to make smaller modules:
  - \* Horizontal diffusion.
  - \* Rayleigh friction.
  - \* Semi-implicit scheme.
  - \* Other variables.
- NEMDYN variables (TCDIS,RTHRESIDG): could they go in NAMDYN? Could TCDIS,RTHRESIDG and LESIDG set-up in SUDYN (remove SUEDYN)?
  - \* LAM models: what is currently done in SUEDYN.
  - \* global models: TCDIS=0.5 (unused), RTHRESIDG=0.5 (unused), LESIDG=F (and reset to F after reading NAMDYN).

### 3.3 Expected final structure for SUDYNA.

- Part 1: first set of default values.
- Part 2: read NAMDYNA, do resettings and tests.
- Part 3: additional settings.
  - SUDYNICI (keys for PC-ICI schemes, reads NAMDYNICI).
  - SUDYNCORE (reads NAMDYNCORE).
  - SUCVER (reads NAMCVER).
  - SUDYNSL (reads NAMDYNSL).
- Part 4: printings.
  - expanded printings.
  - PRT\_DYNICI.
  - PRT\_CVER.
  - PRT\_DYNSL.

### 3.4 Expected final structure for SUDYN.

- Part 1: first set of default values.
- Part 2: read NAMDYN, call SUEDYN to read NEMDYN.
- Part 3: reset variables and test.
- Part 4: SUINTDYNSL.
- Part 5: SUALDYN and SUELDYN.
- Part 6: Printings.
- Part 7: Set-up semi-implicit scheme variables and vertical modes (SUSI, SUNHSI, SUHEG).
- Part 8: Set-up horizontal diffusion (SU(E)HDF., SUHDU).
- Part 9: Set-up RCORDI arrays (SURCORDI).
- Part 10: Set-up Rayleigh friction (SURAYFRIC).

## 4 Calendar.

- Moving TSTEP from YOMDYN to YOMCT0, or creating TSTEP0 in YOMCT0/NAMCT0, must be done for CY40T1.
- Other reorganisations can wait after CY41: they must not start before OOPS geometry reorganisation and command line removal.