On a revised ocean-atmosphere physical coupling interface and about technical coupling software

Sophie Valcke[1] and Eric Guilyardi[2&3]

[1]*CERFACS, 42 Avenue G. Coriolis*
31057 Toulouse Cedex 1, France
[2]*IPSL/LOCEAN, 75252 Paris Cedex, France*
[3]*NCAS Climate, U. of Reading, Reading, UK*

ABSTRACT

In the first part of this paper, we report on a proposition discussed during the EU PRISM project for a revised physical interface linking the main model components of present-day Earth System models (ESMs), i.e. the atmosphere, the ocean, the sea-ice and the land surface. One main difference with current ad-hoc interfaces is the introduction of two new modules: the Surface Layer Turbulence (SLT) and the ocean surface (OS) modules. This added modularity allows physically consistent interpolations from grid to grid, ensures that the exchanges between the model components are "process-based", and helps controlling unstable computations by distinguishing fast and slow processes.

In the second part, we present different technical approaches that can be followed to couple ocean and atmosphere models. We describe in more detail the OASIS coupling software, and in particular the different remapping algorithms offered by OASIS that can be used to transform the coupling fields and to tackle specific coupling problems at the ocean-atmosphere interface such as the need to recreate subgrid variability when going from low to high resolution grids, incoherent coastlines in the ocean and in the atmosphere, or the treatment of vector coupling fields.

1 A revised ocean-atmosphere physical coupling interface

1.1 Introduction

For historical and practical reasons, present-day physical interfaces between Earth System component models are very often the result of an ad-hoc approach, even if much progress has been achieved over the last years to identify the physical, numerical and algorithmic constraints of Earth System Models (ESMs) as a whole and each component individually, including laws of conservation. Because of the diversity of Earth System component models, the design of a physical interface clearly does not have a single optimal solution and any proposition is a compromise between those constraints.

Following the PILPS experience (Polcher et al. 1998), a revised physical coupling interface between the atmosphere, the ocean, the sea-ice and the land surface was discussed during the EU PRISM project (http://prism.enes.org) based on the following guiding principles but considering also historical and practical constraints not likely to evolve in the next few years:

- identify physically based interfaces across which the conservation of energy, mass and momentum can be ensured;
- identify which process needs to be computed by which component/module and ensure that there is no duplication or inconsistency in these computations;
- take into account numerical constraints (stability, impact of different component and coupling time steps, ...) and interpolation constraints (subgrid scale heterogeneity issues, local conservation,...).

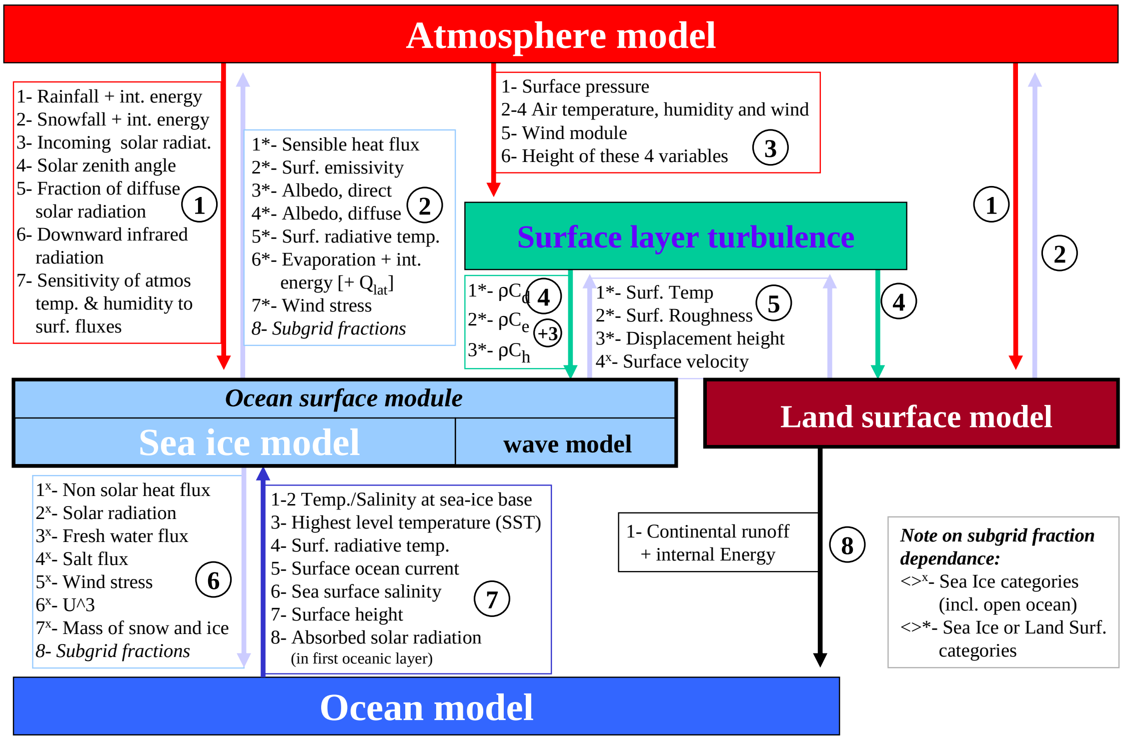

Figure 1: A proposition for a revised physical coupling interface between atmosphere, ocean, sea ice and land surface component models

1.2 The revised physical interface

Following those guidelines, the interfaces described in Figure 1 are proposed for atmosphere, ocean, sea ice and land surface components. The exchanges are represented by groups of fields attached to solid coloured arrows and numbered from ① to ⑧ . In the remainder of the text, each field is identified with a code *I.J* where *I* in the exchange number and *J* the field number. For example field *2.6* is the evaporation.

The introduction of new modules allows 1) to clearly identify where the computation of some physical processes happens, 2) to easily control unstable computations by distinguishing where fast and slow processes are computed and 3) to allow physically consistent interpolations from grid to grid. Increasing the modularity is a long term goal for most components but we restricted ourselves to these key modules for practical reasons.

1.2.1 The Surface Layer Turbulence module

The Surface Layer Turbulence module (SLT module) contains the description of the turbulence inducing diffusion in the surface layer of the atmosphere (above the ocean+sea ice+land surface system). It computes the surface layer turbulent coefficients (exchange ④) from the surface boundary conditions (exchange ⑤ provided by the Ocean Surface module and the land surface component) and the atmosphere prognostic variables at lowest level (exchange ③). It also provides the atmospheric variables of exchange ③ to the Ocean Surface module and the land surface component (see next section). As it is non physical to interpolate the turbulent exchange coefficients, they need to be computed on the finer grid. The Surface Layer Turbulence module needs then to be run on the finer grid.

1.2.2 The Ocean Surface module

The Ocean Surface (OS) module is introduced to help separate the fast ocean and sea ice surface processes, involving heat, water and momentum exchanges with the atmosphere from the slower deeper processes. It sends surface boundary conditions (exchange ⑤) to the SLT module, receives turbulent coefficients (exchanges ④) and atmosphere prognostic variables at lowest level (exchange ③) from this module, and computes a number of surface fields (wind stress, sensible heat flux,...) with bulk formulas. In addition, it receives fields from and sends field to the atmosphere (respectively ① and ②) and to the ocean (exchanges ⑥ and ⑦). A wave model can be included in the OS module to provide sea surface roughness (field *5.2*) to the SLT module.

1.2.3 Energy fluxes and conservation

Between Ocean Surface module and ocean component:

The OS module, via the sea ice model, provides the ocean model with the net solar radiation (*6.2*) and non-solar heat flux (*6.1*) entering the ocean surface. In return, the ocean model provides the OS module with: the temperature at the sea-ice base (*7.1*) used by the sea-ice model to compute the oceanic heat flux at the ice-ocean interface, the sea surface temperature (*7.3*) required for the calculation of the atmospheric turbulent heat fluxes, the sea surface radiative temperature (*7.4*) required for the calculation of the long-wave radiation over leads, the surface height (*7.7*) further transferred to the SLT module, and the fraction of solar radiation absorbed with the first oceanic layer (*7.8*) needed for the computation of the energy budget of leads by the sea-ice model.

Between Ocean Surface module, SLT module and atmosphere component:

The atmosphere component provides the OS module with the incoming solar radiation possibly for different spectral intervals (*1.3*), the solar zenith angle (*1.4*), the fraction of diffuse solar radiation (*1.5*) and the downward infrared radiation (*1.6*). In return, the OS module provides the emissivity (*2.2*), the albedo for direct and diffuse radiation (*2.3* and *2.4*), possibly for different spectral intervals consistently with the partitioning of incoming solar radiation, and the surface radiative temperature (*2.5*); these fields are calculated while solving the surface radiation budget either over free ocean or over sea-ice.

In order to evaluate the surface turbulent fluxes, the atmosphere component provides the Surface Layer Turbulence module and the OS module with surface pressure (*3.1*), air temperature (*3.2*), specific humidity (*3.3*), wind components (*3.4*) and mean scalar wind speed (possibly including gustiness effects due to free convection in boundary layer or due to deep convection, *3.5*), and the height of the level where all these variables are calculated (*3.6*). The atmosphere also provides the Ocean Surface Module with the sensitivity of atmosphere temperature and humidity to surface fluxes (*1.7*). The OS module provides the surface temperature (*5.1*), the surface roughness (*5.2*), the displacement height (*5.3*), and the surface velocity (*5.4*) to the Surface Layer Turbulence module. With fields received through exchanges ③ and ⑤ , the SLT module can compute the exchange coefficient for sensible heat (*4.2*) and for moisture (*4.3*) which are passed to the OS module. With this information, the OS module calculates the sensible heat flux (*2.1*) which is then transferred to the atmosphere component (the latent heat flux can be obtained from the evaporation *2.6*, see next section).

These exchanges ① , ② , ③ , and ④ allow an implicit calculation of the energy fluxes over the whole column from the base of the sea ice to the top of the atmosphere.

1.2.4 Mass fluxes and associated energy conservation

The atmospheric model provides the OS module with rainfall (*1.1*) and snowfall (*1.2*), and associated internal energies, both used by the sea ice model and the ocean model. In return, the OS module provides the atmospheric model with the evaporation/sublimation (*2.6*), based on the exchange coefficient for moisture (*4.3*). This field is needed for the calculation of the hydrological cycle in the atmosphere. If the atmosphere model requires an *average* value (i.e. it cannot distinguish subgrid-scale values, see next section) the *average* latent heat flux must also be provided (due to the non-linearities in its computation). The OS module transfers to the ocean model the net fresh water flux (*6.3*), the net salt flux (*6.4*) and the total mass of snow and ice (*6.7*). The net freshwater flux (*6.3*) results, on one hand, from the net atmospheric water flux over open ocean (Rainfall+Snowfall-Evaporation) and, on the other hand, from snow melting on top of sea ice, ice growth/melting, snow-ice formation, runoff of rainfall through sea ice into the ocean, snowfall and rainfall over leads, and evaporation over leads. The net salt flux (*6.4*) is provided by the sea ice component and results from ice growth/melting and snow-ice formation. In return, the ocean component provides the salinity at the sea ice base (*7.2*), which is used by the sea ice model to compute the freezing point of sea water and, in the future, the salinity of newly-formed sea ice and snow ice, and the sea surface salinity (*7.6*). The total mass of snow and ice (*6.7*) is provided to the ocean model to compute the depression of ice below the water level. Finally, the land surface model provides the continental run-off to the ocean (*8.1*) and the associated internal energy.

1.2.5 Momentum fluxes computations

The surface turbulent wind stress is computed by the Ocean Surface module. The computation uses the drag coefficient (*4.1*) provided by the SLT module, the wind at the lowest level (*3.4*), its module (*3.5*) and its height (*3.6*) provided by the atmosphere via the SLT module, and the surface ocean currents (*7.5*) provided by the ocean. The OS module then transfers the wind stress to the atmosphere (*2.7*) and to the ocean (*6.5*). The OS module also compute the "wind work" U^3 (*6.6*) and provides it to the ocean.

1.3 Subgrid scale computations

Subgrid scale issues are central in Earth System models and must be properly taken into account by physical interfaces. They can either be computed by a component parameterization and exchanged as such (as for example wind gustiness, *3.5*), or occur because of different grid resolutions or multiple sea ice or surface categories into one grid cell. In this second case, two options are available: either the coarser grid component can accommodate subgrid scale information (like 'tiling' in atmosphere) or it can only handle an average value. In the later case, the averaging should be done on the finer grid (i.e. in the Ocean Surface module or land surface component in the present proposal); in the former case, each subgrid flux together with the corresponding subgrid fraction are passed from the finer to the coarser grid. In the present proposal, we assumed that the ocean and sea ice grid is finer than that of the atmosphere component (note that method described here can apply to the reverse configuration, although not often seen in climate studies). Hence, fields of exchanges ② , ⑤ and ⑥ can become arrays. In that case, subgrid fractions of the different sea ice or surface categories (*2.8*) are passed from the OS module to the atmosphere (which will blend the information either in its boundary layer, up to a blending height or even all the way up in the atmospheric column - e.g. delocalized physics - depending on the complexity of the tiling scheme in the atmosphere) and subgrid fractions of different sea ice categories including open ocean (*6.8*) are passed from the sea ice model to the ocean model.

1.4 Time sequence

A schematic possible time sequence for ocean/atmosphere/sea ice coupling via the Surface Layer Turbulence and Ocean Surface modules is presented on Figure 2. This sequence ensures that the atmosphere time integra-

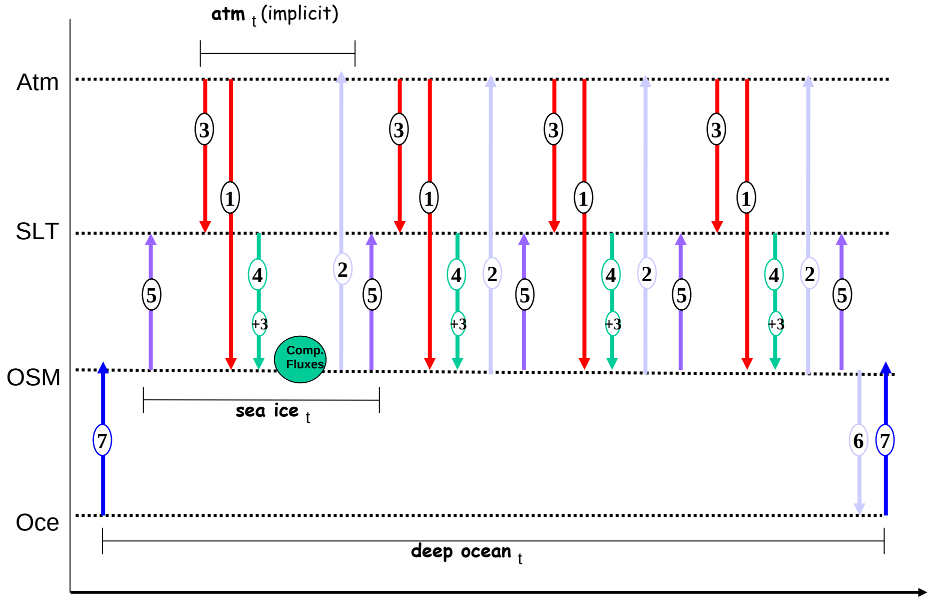

Figure 2: A possible time sequence for ocean/atmosphere/sea ice coupling exchanges via the Surface Layer Turbulence and ocean surface modules

tion of fluxes remains implicit. A classification of the different components involved in the ocean-atmosphere exchanges in terms of speed of processes gives (going from the slowest to the fastest): the (deep) ocean model, the sea ice model (excluding energy flux calculation), the ocean surface module (for energy flux calculation), the Surface Layer Turbulence module, the atmosphere model. Therefore, the coupling exchanges should follow (with F_x being the frequency of exchange ⓧ): $F_7 = F_6 \leq F_5 = F_3 = F_1 = F_4 = F_2$. The coupling exchanges ① , ② , ③ , ④ and ⑤ can be performed many times while only one ⑥ and one ⑦ exchanges take place, as illustrated on Figure 2.

2 About technical coupling software

Although some of the concepts described above are starting to make their way in coupled ocean-atmosphere modelling, the proposed revised physical interface is currently not fully implemented in existing ESMs, that result of an ad-hoc assemblage of existing component models developed and used independently by different research groups.

In this section, we present the different techniques available to assemble existing model codes, and in particular the OASIS coupler and the different algorithms offered by this software to transform the coupling field and to answer specific coupling problems at the ocean-atmosphere interface.

2.1 Different technical approaches to assemble component model codes

Coupling component model codes means managing synchronised exchange of information ("coupling fields") between those codes, and transforming the coupling information provided by one code to ensure that it can be "ingested" by the other code. Ideally, the technical solution chosen should be easy to implement with existing codes, flexible, efficient and portable. Different technical approaches exist to assemble component model codes.

2.1.1 Merging of the codes

The most natural approach is to merge the existing codes into one new application, which means that one code remains a main program and calls the other code as a subroutine. The coupling information can be exchanged by argument passing or by sharing a common module. This approach ensures efficient memory exchanges and portability (in as much as the original codes were themselves portable). However this approach is not flexible, as the coupling algorithm and the coupling exchanges must be hard-coded while merging the codes, and supposes that the user implements and uses in the source or in the target code his own transformations and interpolations. Other disadvantages of this approach are that many conflicts in namespaces, I/O, etc. are likely to appear and that the memory requested by the resulting code may be very large, depending of course on how the original codes are programmed. This first approach is in general not recommended.

2.1.2 Direct use of existing communication protocols

The second approach is to keep the original component models separate but implement the coupling exchanges directly where needed in the codes using an existing protocol such as MPI, CORBA, Unix pipes, or files. Compared to the first one, the advantage of this approach is that no conflict will appear but it is not either flexible as the coupling exchanges, specific to each coupling configuration, will also be hard-coded in the codes. It also requires that the scientist masters the communication protocol and implements his own transformations and interpolations. Finally, its portability depends on the portability of the chosen communication protocol.

2.1.3 Use of a coupling framework

The third solution is to use a coupling framework, such as ESMF (http://www.esmf.ucar.edu) or FMS (http://www.gfdl.noaa.gov/ fms/). This approach supposes that the user splits the original codes into elemental units (at least in initialisation, main, and termination units), adapt the units to the framework standard data structure and calling interface, and finally uses the framework to build and control a hierarchical merged application integrating the different units. This approach is fully flexible (the different units can be easily reused in different applications), allows the user to use the different tools offered by the framework (parallelisation, regridding, time management, etc.) and ensure efficient coupling exchanges within the merged application. But it requires a deeper level of interference in the codes and imposes strict coding rules in order to take full advantage of the framework functionalities. This approach is therefore probably the most recommended one in a controlled top-down development environment.

2.1.4 Use of a coupler

In many cases though, the different component models chosen to form an ESM come from different research groups that also use these components independently in stand-alone mode for other research purposes and that are not likely to follow strict coding rules imposed by external constraints. In this case, a less intrusive approach based on the use of a coupler and associated coupling library, such as MpCCI (http://www.mpcci.de), MCT (http://www.mcs.anl.gov/mct), PALM (http://www.cerfacs.fr/globc/PALM_WEB/index.html) or OASIS (https://oasistrac.cerfacs.fr/) is probably the best trade-off that can be chosen. In particular, it ensures that the original codes will run as separate executables with main characteristics (e.g. internal parallelisation) unchanged with respect to the uncoupled mode. The drawback is that the execution of the resulting coupled model may in some cases be less efficient than a more integrated one-executable approach. This approach is flexible as the coupling exchanges generally follow the principle of "end-point" data exchange (see section 2.2.2). The portability of the coupling depends on the portability of the coupler, criteria usually of great importance for the coupler development teams. This approach also allows the user to take advantage of the different transformation and regridding routines offered with the coupler.

2.2 The OASIS coupler

We describe here in more detail the OASIS coupler which represents one implementation of the approach presented in 2.1.4.

2.2.1 Historic and community

In 1991, CERFACS was commissioned to develop a software interface to couple existing numerical General Circulation Models of the ocean and of the atmosphere. Today, both the widely used OASIS3 version (Valcke 2006), which is the result of more than 15 years of evolution, and the newer fully parallel OASIS4 version (Valcke & Redler, 2006), which writing started during the PRISM EU project and which is presently developed thanks to an active collaboration between NEC Laboratories Europe IT Research Division in Germany, the Centre National de la Recherche Scientifique (CNRS) and CERFACS, are available.

The OASIS community has steadily grown since its first release. The OASIS coupler is currently used by about 25 modelling groups in Europe, Australia, Asia and North America, on the different computing platforms used by the climate modelling community.

2.2.2 Data exchanges with OASIS

To exchange coupling information with other components, a component model needs to call few specific routines of the OASIS coupling library for its initialisation, grid and partition definition, field declaration, field **Get** and **Put** actions (to receive or send a field by respectively) and termination.

In OASIS, the communication follows the “end-point” principle, i.e. there is no reference in the component model code to the origin of a Get action or to the destination of a Put action; the source and target component models (coupling exchange) or the input or output file (I/O) are set externally by the user. This ensures an easy transition between different coupling configurations, in particular from the coupled mode (Get/Put actions leading to a coupling exchange performed using MPI) to the forced mode (Get/Put action leading to reading/writing from/to a file using the GFDL mpp_io library, Balaji 2001), totally transparent for the component model itself. Furthermore, the Get/Put routines can be called at each time step in the component model code but the receiving/sending actions will effectively be performed only at appropriate times from/to the appropriate source/target following the configuration externally defined by the user.

2.2.3 Regridding algorithms available in OASIS

For each coupling exchange, OASIS performs the transformations and regridding needed to express the source field on the grid of the target model. The following 2D regridding algorithms based on the SCRIP library (Jones 1999) are available in OASIS3 and OASIS4, with 3D extensions in OASIS4 only (for more details, the reader is referred to the SCRIP User Guide available at http://climate.lanl.gov/Software/SCRIP/):

- N nearest-neighbour: the N closest source neighbours are used. The weight of each neighbour is inversely proportional to d, its Great Circle distance to the target point, or to $exp(-1/2 \cdot d^2/\sigma^2)$ where σ^2 is the variance of a Gaussian function.

- bilinear: the 4 enclosing source neighbour points are used and their respective weight is evaluated using a general bilinear iteration in a continuous local coordinate system.

- bicubic: the value of the source field, its gradients and cross gradient with respect to the local directions i and j at the 4 enclosing source neighbour points are used. For Reduced Gaussian grid, a standard bicubic algorithm with the 16 enclosing source neighbours is used.

- 1st order conservative remapping: the weight of a source cell is proportional to area of the source cell intersected by target cell. Using the divergence theorem, the SCRIP library evaluates this area with the line integral along the cell borders enclosing the area. As the real shape of the borders is not known (only the location of the 4 corners of each cell is known), the library assumes that the borders are linear in latitude and longitude between two corners. In general, this assumption becomes less valid closer to the pole and in that region, the library evaluates the intersection between two border segments using a Lambert equivalent azimuthal projection. One limitation of the SCRIP algorithm is that is also supposes, for line integral calculation, that $sin(latitude)$ is linear in longitude on the cell borders which again is in general not valid close to the pole. A projection or at least a normalization by the true area of the cells (i.e. by the areas as considered by the component models) is needed. Another step that could improve the precision of the calculation would be to use a border middle point i.e. describe the cell with 8 points.

2.2.4 The SUBGRID transformation

One problem with the 1^{st} order conservative remapping when going from a low resolution source grid to a high resolution target grids is that all (small) target grid cells located entirely under the same (big) source cell will receive the same value; the shape of the source cells will therefore be "visible" on the coupling field after the remapping. One way to circumvent this problem is to use a 2^{nd} order conservative remapping that allows to recreate sub source grid variability while conserving the surface integral of the field. The SCRIP library proposes such a 2^{nd} order conservative algorithm which uses the derivative of the field with respect to the latitude and the derivative of the field with respect to the longitude. This algorithm is available in OASIS3 but has not been fully validated. One has to note here that the derivative of the field with respect to the latitude or the longitude cannot be easily computed for grids other than logically rectangular ones.

Another possibility is to use the SUBGRID transformation available in OASIS3.

`SUBGRID` can be used to interpolate a field from a coarse grid to a finer target grid. A first-order Taylor expansion of the field on the fine grid relatively to a state variable is performed (for instance, an expansion of the heat flux relatively to the SST):

$$Q_o^i = Q_a + \frac{\partial Q_a}{\partial T_a}(T_o^i - T_a)$$

where Q_o^i (Q_a) is the flux on the fine (coarse) grid, T_o^i (T_a) an auxiliary field on the fine (coarse) grid (e.g. the SST) and $\partial Q_a/\partial T_a$ the derivative of the flux versus the auxiliary field on the coarse grid. It can be shown that the transformation is conservative if T_a corresponds to a 1^{st} order conservative remapping of T_o^i from the fine to the coarse grid.

2.2.5 Problems caused by ocean and atmosphere non-matching coastline

A traditional issue in ocean-atmosphere coupled modelling is the one of non matching coastlines due to different sea-land masks in the ocean and atmosphere models. The only way to solve this issue is to support subsurfaces for each cell in the atmosphere (i.e. "tiling") and let the fraction of sea and land in each cell to be dictated by the ocean model sea-land mask.

In models where subsurfaces are not supported, workarounds have to be applied when possible. In the following, we suppose that both in the ocean and atmosphere models land cells are masked and sea cells are non masked. When an atmospheric non masked cell covers masked and non masked ocean cells, the flux corresponding to the part of the atmospheric cell intersecting masked ocean cells will be lost. The only workaround in OASIS to correct this lost is to impose global conservation: the area integral of the flux is calculated on both the source grid before interpolation and the target grid after interpolation, without considering values of masked points, and the difference is uniformly distributed over the non masked target grid points. This solution is obviously not ideal as all non masked ocean cell flux will be modified but at least ensures that the total energy of the coupled system does not drift.

When the opposite situation occurs, i.e. a non masked ocean cell is covered by masked and non masked atmosphere cells, two normalisation options are available in the SCRIP library implemented in OASIS. The flux received from the non masked atmospheric cell can be normalized either by 1) the total ocean cell area (*DESTAREA* option), or by 2) the ocean intersected cell area (*FRACAREA* option). With option 1), local (and therefore global) flux conservation is ensured, but unreasonable flux values may result. With option 2), the flux is not locally conserved, but the value of the flux itself is reasonable. With option 2), it is even possible in OASIS, to assign the nearest-neighbour value to sea ocean cells that would be covered only by land atmospheric cells (*FRACNNEI* option). Of course, with option 2) the user can decide to impose global conservation with the technique described in the previous paragraph.

2.2.6 Interpolation of vectors

At the ocean-atmosphere interface, vector fields such as the wind stress or the ocean currents have to be interpolated. The vector components of these fields are usually expressed in a local referential or in the Earth spherical referential. The interpolation of vectors component per component given in a local referential is not exact, especially where the referential changes rapidly, i.e. near the poles in our case.

The OASIS coupler allows an exact interpolation of vector field thanks to the following procedure. If the vector components are provided in a local referential, they are first projected onto the spherical referential. The resulting vector is then projected in a cartesian referential, into which the 3 resulting components are interpolated separately. The interpolated components are then projected back in the spherical referential. It is then verified that the resulting vertical component in the spherical referential is zero. Finally, the resulting zonal and meridional components are projected to the target grid local referential if needed.

This procedure allows an exact interpolation of the vector fields expressed in the local referential.

3 Conclusions

In the first part of this paper, we have presented a proposition for a revised physical interface between the atmosphere, the ocean, the sea-ice and land surface ESM components. Compared with current ad-hoc approaches, increased modularity is proposed with the introduction of a Surface Layer Turbulence (SLT) and an Ocean Surface (OS) modules, which ensure that computation of surface fluxes are physically based, consistent interpolations are performed from grid to grid, and unstable computations are controlled by distinguishing fast and slow processes. The SLT module runs on the finer grid and computes the surface layer turbulent coefficients from the surface boundary conditions and the atmosphere prognostic variables at lowest level. The OS module computes the surface fluxes at the surface. The exchanges between the OS module, the SLT module and the atmosphere model allow a calculation of the surface fluxes at the resolution of the finer grid while ensuring an implicit calculation of the energy fluxes over the whole column from the base of the sea ice to the top of the atmosphere.

In the second part, we have presented different techniques technical approaches that can be used to assemble component model codes. The approaches based on the use of a coupling framework (2.1.3) or on the use of a coupler (2.1.4) both offer the benefits of shared software developmemt and seem to be the most appropriate, respectively in a controlled development environment or to couple independently developed components. The OASIS coupler and some of its regridding algorithms have been then described. In particular, the SUBGRID transformation to reproduce subgrid variability when going from a coarse to a fine grid was presented. The workarounds available in OASIS to address the problem of non matching coastlines between ocean and atmosphere models needed when subsurfaces are not supported in the atmosphere were then explained; this ill-posed problem cannot be fully resolved even if global conservation can at least be ensured. Finally, It has been described how the interpolation of vector fields must be done on the vector components projected in a cartesian space in order to be exact.

Acknowledgements

We would like to thank Jan Polcher, Thierry Fichefet, Gurvan Madec, Olivier Marti and Serge Planton for their key contribution to the discussions that lead to the revised physical interface discussed here. Support from EU projects PRISM (EESD-1999-9) and METAFOR (INFRA-2007-1.2.2 - G. A. no 211753) are acknowledged.

References

Balaji 2001: Parallel Numerical Kernels for Climate Models, ECMWF TeraComputing Workshop 2001, World Scientific Press, Reading.

Jones P.W. 1999: First- and Second-Order Conservative Remapping Schemes for Grids in Spherical Coordinates. *Monthly Weather Review*, 127, 2204-2210.

Polcher J., McAvaney B., Viterbo P., Gaertner M.-A., Hahmann A., Mahfouf J.-F., Noilhan J., Phillips T., Pitman A., Schlosser C.A., Schulz J.-P., Timbal B., Verseghy D., Xue Y. 1998: A proposal for a general interface between land-surface schemes and general circulation models. *Global and Planetary Change*, 19:261-276

Valcke, S. 2006: OASIS3 User Guide (prism_2-5). PRISM Support Initiative Report No 3, 68 pp. (http://www.prism.enes.org/Publications/Reports/oasis3_UserGuide_T3.pdf)

Valcke S. and Redler R. 2006: OASIS4 User Guide (OASIS4_0_2), PRISM Support Initiative Report No 4, 64 pp. (http://www.prism.enes.org/Publications/Reports/OASIS4_User_Guide_T4.pdf)