PERFORMANCE ANALYSIS OF EC-EARTH3.2: COUPLING

BSC-CES-2016-006

EC-Earth3.2, Earth System Model, Coupling

Mario C. Acosta, Xavier Yepes-Arbós, Sophie Valcke[1], Eric Maisonnave[1], Kim Serradell, Oriol Mula-Valls[2] and Francisco Doblas-Reyes

Earth Sciences Department

Barcelona Supercomputing Center - Centro Nacional de Supercomputación (BSC-CNS)

23 December 2016

[1]Centre Europeen de Reserche et de Formation avancee en Calcul Scientifique (CERFACS)
[2]HPCnow!

Series: Earth Sciences (ES) Technical Report

A full list of ES Publications can be found on our website under:

https://earth.bsc.es/wiki/doku.php?id=library:external:technical_memoranda

Barcelona Supercomputing Center–Centro Nacional de Supercomputación (BSC-CNS)

C/Jordi Girona, 31 | 08034 Barcelona (Spain)

Summary

Earth system models include different components that can all run on different grid configurations, supporting a variety of spatial resolutions and time scales and exchanging boundary data with each other through a coupler. This process of coupling among components is not a trivial task and can be computational expensive. The main goal of this document is to study the computational cost that coupling represents for climate models using for this purpose EC-Earth as an example.

The coupling between the two main components of EC-Earth, the ocean component (NEMO) and the atmospheric component (IFS) is evaluated, using OASIS3-MCT as coupler. The results show that the coupling could represent an important bottleneck (up to 50% of the time step of IFS) of the execution if a global conservation operation, combined to a large number of parallel processes and a high coupling frequency, is applied to the coupling fields without activating the optimization options.

Two options available in OASIS3-MCT are studied in this document to improve the coupling configuration. These options can reduce the coupling cost by more than 90%. The first one optimizes the field exchanges between components and minimizes the interpolation and communication process. The second one shows that OASIS3-MCT is able to reduce the MPI overhead by using global communications instead of one-to-all/all-to-one communications and by removing the serial calculations done by default on the master process.

A reproducibility test was done to evaluate the two optimizations. This test takes into account the chaotic nature of climate models and the small differences introduced by parallel operations (such as reduction operations). The test proves that the significant differences are less than 1%, comparing the simulation results of one year experiments, using the default configuration and both optimizations (together and separately). Similar results are obtained when the results of two identical parallel simulations are compared.

Additional tests have been done to evaluate the difficulty of load balancing the components of coupled models. This document evaluates the computational performance of each component independently and studies how to achieve the best load balance among components by finding the optimal number of resources for each one.

Contents

Index of figures

Index of tables

1. Introduction

Earth System Model (ESM)-based experiments are typically complex and large in size. ESMs include different components that can all run on different grid configurations, supporting a variety of spatial resolutions and time scales and exchanging boundary data with each other through a coupler. We call component or model an ensemble of discretized equations that mathematically represent one from the several climate sub-systems. A climate model is the assembly of some of these numerical models, which can be independently developed, interacting in a coupled way.

The components of these climate models such as the atmosphere and ocean components continuously exchange several quantities such as momentum, heat or freshwater. The exchanges occurring between components is called coupling. Exchanged information (coupling fields) is discretized on grids, which can be different from one model to another. It might be then necessary to interpolate information from the source model grid to the target model grid. Exchanges between the coupled system models are periodic. To be able to perform its own calculations, a model is using boundary information coming from another model: typically, at the end of a coupling time step, each model waits for information coming from the other model to resume its calculations.

This process of coupling among components is not a trivial task and there are fundamental reasons why this process of coupling is difficult to implement in numerical models of the coupled system, at any scale. Obviously, the time discretization of the conservation equations implicitly inhibits flux variability over a time step. In this regard, the best possible scheme would have equal ocean and atmosphere time steps. On the other hand, some of the variables must be coupled in a conservative way so that the total flux or energy in the source and target grids is the same, increasing the complexity of the coupling algorithm. These issues represent serious practical implications for the numerical implementation and the computational expense of the coupling.

The main goal of this document is to study the computational cost that the coupling represents for climate models, taking into account the options used and the features required. A deep analysis is done for the coupling between the two main components of ESMs models, the atmospheric and ocean components, using for this purpose EC-Earth as an example. Some particular problems in the EC-Earth implementation are also discussed.

2. The model

EC-Earth is a project, a consortium and a model system. The EC-Earth consortium consists of several academic institutions and meteorological services from different countries in Europe. The EC-Earth model is a global, coupled climate model that consists of two main components: IFS for the atmospheric model and NEMO for the ocean model. They are coupled using OASIS3-MCT. It has other sub-components: LIM for the sea ice, XIOS for NEMO's input/output, and Run-off mapper for freshwater distribution (rivers, ice …) to the ocean. A brief description of the components follows:

- The OASIS3-MCT coupler: this is a coupling library to be linked to the component models whose main function is to interpolate and exchange the coupling fields between them (using the same or different grids).

- The Integrated Forecasting System (IFS) as atmosphere model: this is an operational global meteorological forecasting model developed and maintained by the European Centre of Medium-Range Weather Forecasts (ECMWF). The dynamical core of IFS is hydrostatic, two-time-level, semi-implicit, semi-Lagrangian and applies spectral transformations between grid-point space and spectral space. Vertically, the model is discretised using a finite-element scheme. A reduced Gaussian grid is used in the horizontal. The IFS cycle used in this analysis is 36r4.

- The Nucleus for European Modelling of the Ocean (NEMO) as ocean model: NEMO is a state-of-the-art modelling framework for oceanographic research, operational oceanography seasonal forecast and climate studies. It discretizes the 3D Navier-Stokes equations, being a finite difference, hydrostatic, primitive equation model, with a free sea surface and a non-linear equation of state. The ocean general circulation model (OGCM) is OPA (Océan PArallélisé). OPA is a primitive equation model which is numerically solved in a global ocean curvilinear grid known as ORCA. EC-Earth 3.2.0 uses NEMO's version 3.6 with XML Input Output Server (XIOS) version 1.0. XIOS is an asynchronous input/output server used to minimize previous I/O problems.

- The Louvain-la-Neuve sea-Ice Model 2/3 (LIM2/3): LIM is a thermodynamic-dynamic sea-ice model directly included in OPA.

- The runoff-mapper component is used to distribute the runoff from land to the ocean through rivers. It runs using its own binary and is coupled through OASIS3-MCT.

EC-Earth uses the MPI (Message Passing Interface) paradigm, using a specified number of tasks for NEMO and IFS models, one process for XIOS and another one for the runoff-mapper.

3. Coupling

As explained in the previous section, EC-Earth uses the latest version of the OASIS3-MCT (OASIS3-MCT) library. The information given in this section to explain how OASIS3-MCT works is based on the OASIS3-MCT documentation (for more details see Valcke, S., Craig, T., and Coquart, L.: OASIS3-MCT User Guide, OASIS3-MCT 1.0. CERFACS Technical Report, TR/CMGC/12/49) and LUCIA documentation (for more details see Maisonnave, E., Caubel, A., 2014: LUCIA, load balancing tool for OASIS3-MCT coupled systems, TR/CMGC/14/63). To set up such coupling, the developer has to modify the source codes, calling the OASIS3-MCT library subroutines using OASIS3-MCT API (Application Programming Interface). This interface gathers different operations: initialization, MPI partitioning description, coupling field definition, coupling fields send and receive operations and termination. The coupling characteristics for each field are defined in a configuration text file called "namcouple". The MPI communication library ensures the exchange of numerical arrays that hold coupling fields. If the results needed by both models are the results produced at the previous coupling time step, both models can run at the same time (concurrently).

For a given model, a coupling time step can be decomposed as follows (not necessarily in the same order):

- The model performs its own calculations.
- The coupling library, directly linked to the source model, performs interpolations before sending the coupling fields.
- The coupling library sends the coupled variables to other components via MPI communications (done in Oasis_put subroutines).
- The coupling library, directly linked to the target model, receives the coupled variables from the source models via MPI communications (done in Oasis_get subroutines).

The interpolation can be explained briefly as the process to transform the value of one variable from the grid of the source component to the grid of the target component. Usually the source and target grids are different. This requires some transformations before (or after, but we assume here that the interpolation is always done before the communication) sending the variables to the other component. This transformation can be conservative, i.e. conservative interpolations are available in OASIS3-MCT. However, in some cases, e.g. when the coastlines do not match in the source and target models (which is the case in EC-Earth) additional computation and communication between the source and target grids are done ensuring global conservation (i.e. ensuring that the total integrated value of the coupling field is exactly the same in both grids).

In figure 1 we can see the different steps that can be identified with LUCIA, a tool provided with OASIS3-MCT to analyse coupling exchanges. It is convenient to use LUCIA and the metrics

provided by this tool to compare with results obtained by other institutions.

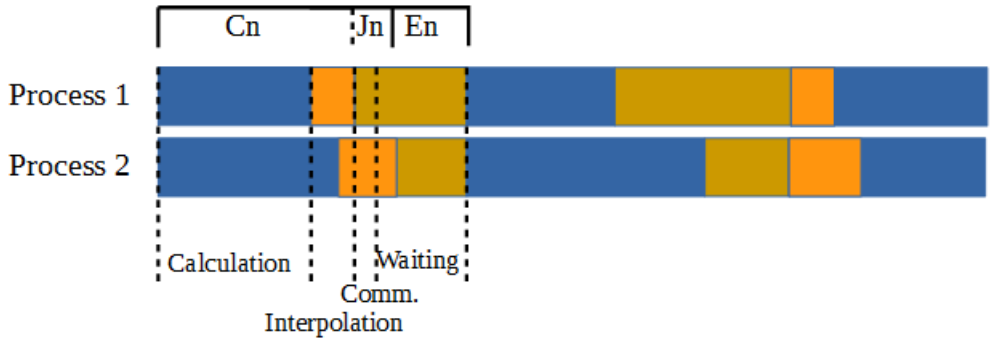


Figure 1. En (Communication and waiting time), Cn (Calculation and interpolation time) and Jn definition for two processes of a parallel model doing coupling. Each is task is also classified by colours: blue for calculation time, orange for interpolation and other transformation time and brown for communication and synchronization (waiting) time.

LUCIA provides the following information:

En: Time spent by the component sending and receiving coupling MPI messages and waiting for the other component used for coupling. This time encompasses every communication time. Since OASIS3-MCT uses non-blocking send (MPI_WAITALL + MPI_ISEND), the sending time is the time necessary to write messages into the MPI buffer. The receiving time encompasses the time spent to read messages in the MPI buffer and the possible load unbalance time between components: a model can have to wait for the other one to end its calculations and send the requested information.

En measures the time spent between the latest send and first receive operations for all MPI processes (and for all coupling fields) of the component; note that for this reason it includes waiting and communication time; the metrics and values provided by LUCIA must be analysed taking this into account.

Cn: The time spent by the component to perform its own calculations and OASIS3-MCT interpolations and transformations. This time is the complement to *En* time: the sum *Cn* + *En* must be equal to the total simulation time. ***Cn*** includes model calculation times but also, when the model is parallel, the possible model internal unbalancing (so called **jitter**). ***Jitter (Jn)*** is the adjustment time needed before all MPI processes are able to send or receive a coupling variable. This time can be linked to the model itself (some calculations are more costly in some MPI sub-domains than in others) but also to OASIS3-MCT transformations because some processes can handle bigger sub-domains (which increases load unbalance between processes and, then, ***Jn***).

Note that the computation and calculations of the component itself are included in ***Cn*** with the time needed for OASIS3-MCT to do the transformations. This means that ***En*** and ***Cn*** times

measured by LUCIA do not provide enough information to get the computational cost required by the coupling, so additional results and metrics must be used for this purpose.

It is also important to take into account that unbalanced computation among the components can affect the final execution time of the coupled model. Figure 2 shows how the coupling is done with OASIS3-MCT between two components (the process is presented in the simplest way). In this case both of them can do some computation in parallel. The communication is done to transfer coupled fields after each time step. This implies that the fastest component has to wait until the slowest component finishes a time step before receiving the fields needed to start the next time step. This must be taken into account to achieve the load balance of the model and avoid wasting resources, because the final execution time of the coupled model is limited by the execution time of the slowest component. This means that it is not enough to find the number of processes until each component scales, but also it is necessary to find a load balance between components.

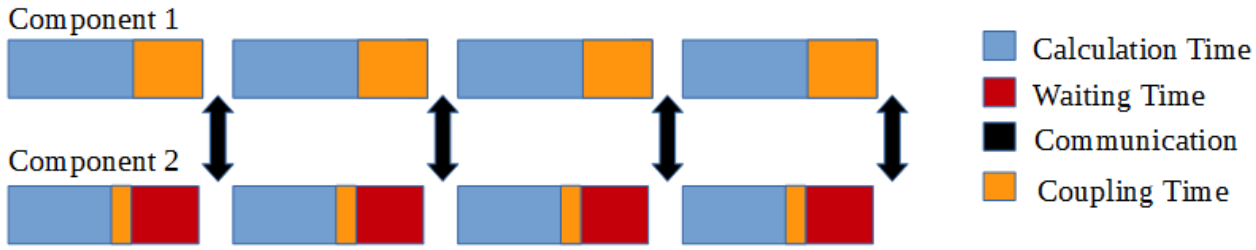


Figure 2. Execution of four time steps and the coupling between two components at the end of each time step. Colours show how much time is used for each task. For this hypothetical case, the fastest model has to wait until the slowest model finishes before exchanging coupled fields.

4. Experiment design

Several tests have been done to evaluate the final execution time (time to complete the simulation) and the execution time of each component. The tests have been done in order to fulfil the following goals:

- Find a load balance between the main components of EC-Earth (IFS and NEMO).
- Explore if increasing or decreasing the number of processes used for IFS could introduce an overhead in the execution time of NEMO (by increasing its waiting time) and vice versa.
- Analyse the computational cost of the coupling in EC-Earth, taking into account both the interpolation/transformation and communication times.

To address these goals the tests have been performed in two groups, one focusing on each component. Each group studies how to scale IFS and NEMO respectively, taking into account its dependencies with the other component. Each group is divided into four or five blocks. Each block analyses the impact of the number of processes used for the other component.

- IFS group: Each block of this group evaluates the performance of the coupled system for a different number of processes for IFS (128, 384, 512, 640 and 768). Each of these five blocks considers different numbers of processes used for NEMO.
- NEMO group: A different number of processes for NEMO is considered for each block (64, 128, 256 and 384). Each of these four blocks considers different numbers of processes for IFS.

For each test, EC-Earth 3.2.0 (trunk revision r3255) has been used. The LUCIA results are averaged from three different 3-month or 4-month simulations. The rest of the parameters use the default standard configuration proposed by the EC-Earth development portal. The next table summarizes the complete configuration:

Model	nemo-3.6
Configuration name	ORCA1L75_LIM3
Resolution	ORCA1L75

Modules	OPA and LIM3
Time step	2700 seconds (32 time steps per day)
Compilation keys	key_trabbl key_vvl key_dynspg_ts key_ldfslp key_traldf_c2d key_traldf_eiv key_dynldf_c3d key_zdfddm key_zdftmx key_mpp_mpi key_zdftke key_lim3 key_iomput key_oasis3 key_oa3mct_v3

Table 1. Configuration for NEMO-LIM.

Model	ifs-36r4
Resolution	T255L91
Time step	2700 seconds (32 time steps per day)

Table 2. Configuration for IFS.

Component model	OASIS3-MCT_3.0
Coupling frequency	2700 seconds (default value)

Table 3. Configuration for OASIS3-MCT_3.0.

For the case study presented here, a particular computing environment has been defined and used for the EC-Earth model on the supercomputer MareNostrum3 (MN3). MN3 is an Intel machine located at the Barcelona Supercomputing Center (BSC). Each computing node has two processors Intel Sandy Bridge-EP E5-2670/1600 of 8-core at 2.6 GHz. CPU nodes with 8x4GB DDR3-1600 DIMMS (2GB/core) are connected with a high-performance Infiniband FDR10 network. An auxiliary Gigabit Ethernet network is used for the shared file system. MN3 uses IBM LSF to manage the execution of jobs, a batch scheduler system where the users only access the login nodes and submit jobs.

5. Results

The tests described in the previous section allow evaluating the computational cost of coupling EC-Earth. The results are discussed separately for IFS and NEMO.

5.1. IFS group

This group considers different combinations of numbers of processes, divided into five blocks, as illustrated in figure 3. Each block considers a fixed number of processes for IFS but different numbers of processes for NEMO. For example, in the first block, four numbers of processes for NEMO (64, 128, 256 and 384) have been tested, while maintaining the number of processes for IFS to 128. Figure 3 shows the *Cn* time (blue line, see Section 3) and the total execution time to complete a simulation of three months (red line). The results show that the IFS *Cn* times for the different runs inside each block are almost equal. This means that the increase in the number of processes for NEMO does not affect the execution time of IFS as can be expected, this increment could be possible since the number of communications between components could increase, when more subdomains are needed. It is also shown in the figure that the total execution time remains almost invariant inside each block (except for the 512+128 case, see below). This is because the communication time is not affected by the number of processes used for NEMO or, at least, because the overhead produced by other issues such as synchronization or latency dominate.

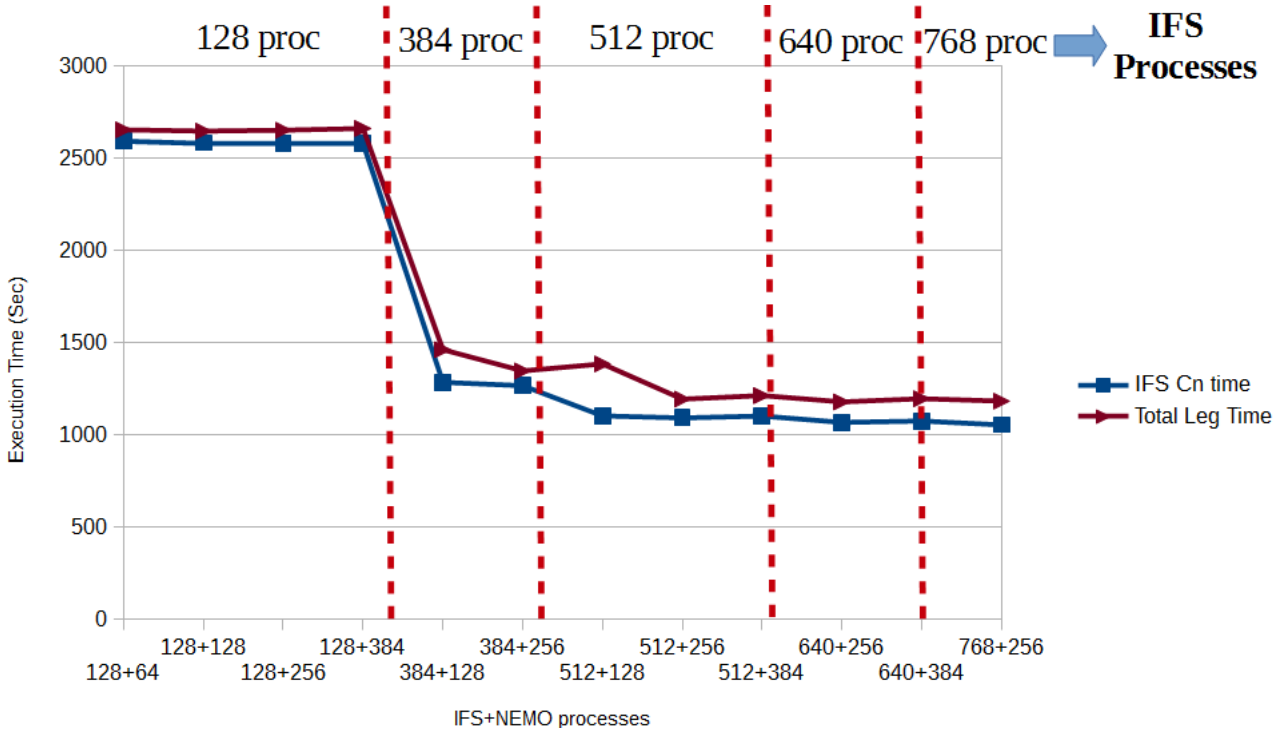


Figure 3. Execution time (seconds) for different combinations of IFS processes divided into blocks of experiments. Each block contains different combinations of NEMO processes. The figure shows the IFS Cn time (execution time, blue line) and total execution time (total leg time, red line).

An important conclusion of figure 3 is that this kind of test can be used to perform a scalability analysis of IFS, at least for the IFS component coupled (the IFS standalone execution could have a different behaviour). The figure shows that the IFS execution time decreases for the first three blocks (until 512 processes). Beyond this number of processes the same execution time is obtained (for 640 and 768 processes).

The figure also shows that the total execution time is very similar to IFS *Cn* time. This happens because in all these experiments IFS is always the slowest component. This shows that the total execution time depends on the slowest model, and for EC-Earth is equal to the IFS total execution time plus a small extra time which represents 4% approximately. It is important to highlight that in this case, the extra time does not correspond to *En* time of IFS (communication + waiting time); indeed several tests show that if IFS is slow enough, the *En* time of IFS is reduced to almost 0. This means that in this kind of tests, the communication time as measured by LUCIA is not representative.

Comparing all the results, there is only one exception where total execution time does not equal IFS *Cn* time + 4% (approximately), i.e. for the case 512+128 (IFS+NEMO processes), for which the total execution time is bigger. This happens because in this case IFS is not the slowest model. Actually, this is the combination reaching the best load balance between IFS and NEMO, i.e. the combination for which *Cn* times for IFS and NEMO are similar. So it would be expected to be the execution time leading to the minimal use of resources, even if the total time increases slightly.

This can be explained considering figure 4, which contains the results for only the block three (512 processes for IFS) and three combinations for NEMO (128, 256 and 384).

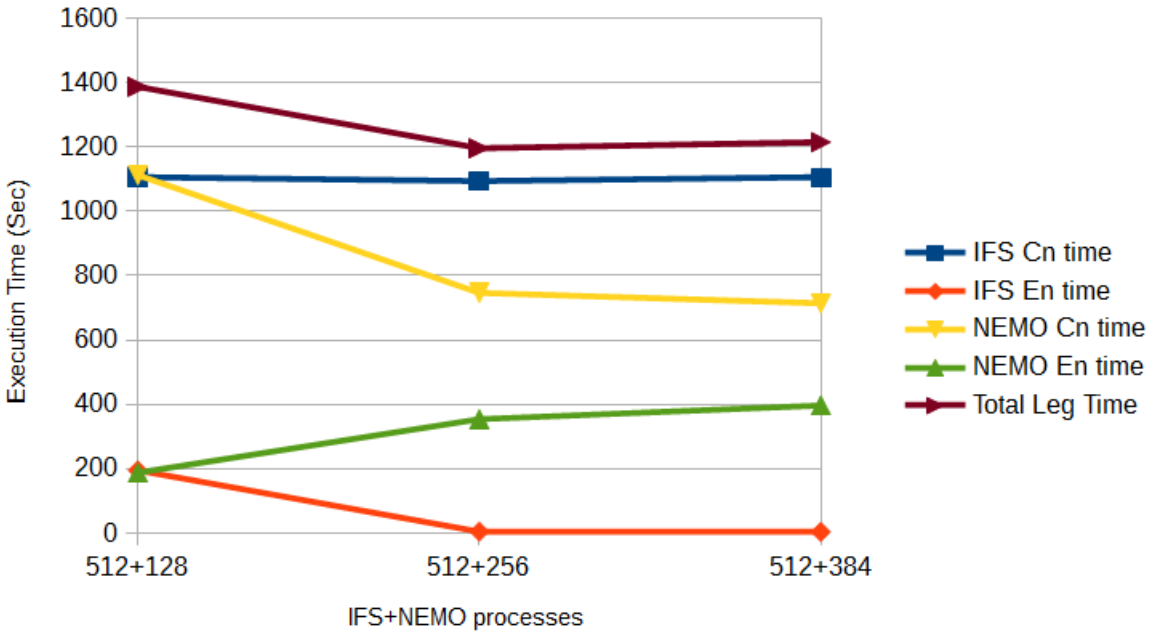


Figure 4. Timings (seconds) for a simulation with 512 processes for IFS and three numbers of processes

for NEMO. It shows IFS Cn and En time (blue line and orange time respectively), NEMO Cn and En time (yellow line and green time respectively) and the total execution time (red line).

The figure shows that when IFS and NEMO *Cn* times are equal, *En* times are equal too (512+128). However, the final execution time is larger than for the other cases shown. In the other two cases (512+256 and 512+384), the same behaviour than in all the other tests happens: the Nemo *Cn* time is smaller than the IFS *Cn* time, the IFS *En* time is almost 0, the NEMO *En* time increases and the final execution time is similar to the execution time of the slowest model (plus a small extra time). To fully understand the 512+128 results, the fact that the IFS radiation computation is calculated only once every four time steps should be taken into account.

This is illustrated in figure 5. Load balance between two components is achieved when both components have similar computation and interpolation time so the communication time is done at the same time and neither IFS nor NEMO have to wait. Here it is shown that once over fourth IFS time step takes longer (when radiation is calculated), the other three being smaller than the NEMO *Cn* time (which is always the same for every time step). So even if the average IFS and NEMO *Cn* times are equal in the end, there are three time steps for which IFS waits, and one time step for which NEMO waits (i.e. when the radiation is calculated in IFS), and, as a result, increasing the final execution time. Due to this, load balance for IFS and NEMO is not achieved even if the LUCIA tool suggests that the *Cn* times of these two components are equal. One solution could be to use more resources for NEMO in order to reduce its calculation time, achieving load balance for at least three out of four time steps, and so that NEMO would only wait when radiation is calculated in IFS.

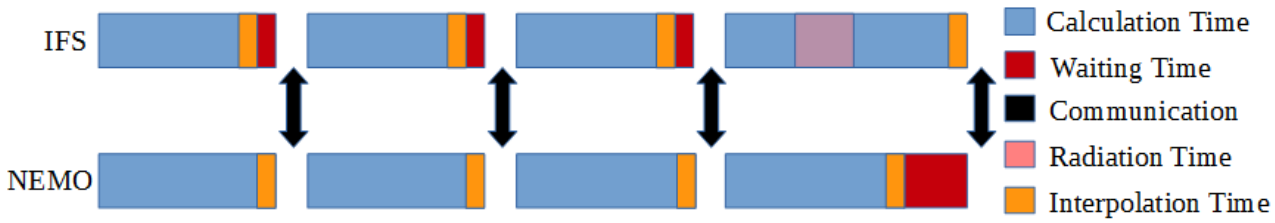


Figure 5. Four time steps of IFS and NEMO components using the configuration by default (sequential mode).

5.2. NEMO group

Figure 6 shows the results for these tests. Different combinations of number of processes are tested, divided into four blocks as can be seen on the top of the figure. Each block combines a fixed number of processes for NEMO with different number of processes for IFS. For example, in the second block, three numbers of processes for IFS (128, 384 and 512) have been tested maintaining the number of processes for NEMO to 128. The figure shows the *Cn* (yellow line, see Section 3) and the total execution time to complete a simulation of three months (red line). The results show that *Cn* times for NEMO inside each block, where the

number of processes used is exactly the same, are equal. This means that, as expected, a variation in the number of processes for IFS does not affect the execution time of NEMO, which is coherent with the results obtained for the IFS group tests.

In this case, total execution time is not similar to NEMO *Cn* time. This is because, as it was shown in the IFS group tests, the total execution time depends on the slower model, which is IFS in this case. These results can be used to evaluate the scalability of NEMO, observing that the coupled system is able to scale until only 256 processes.

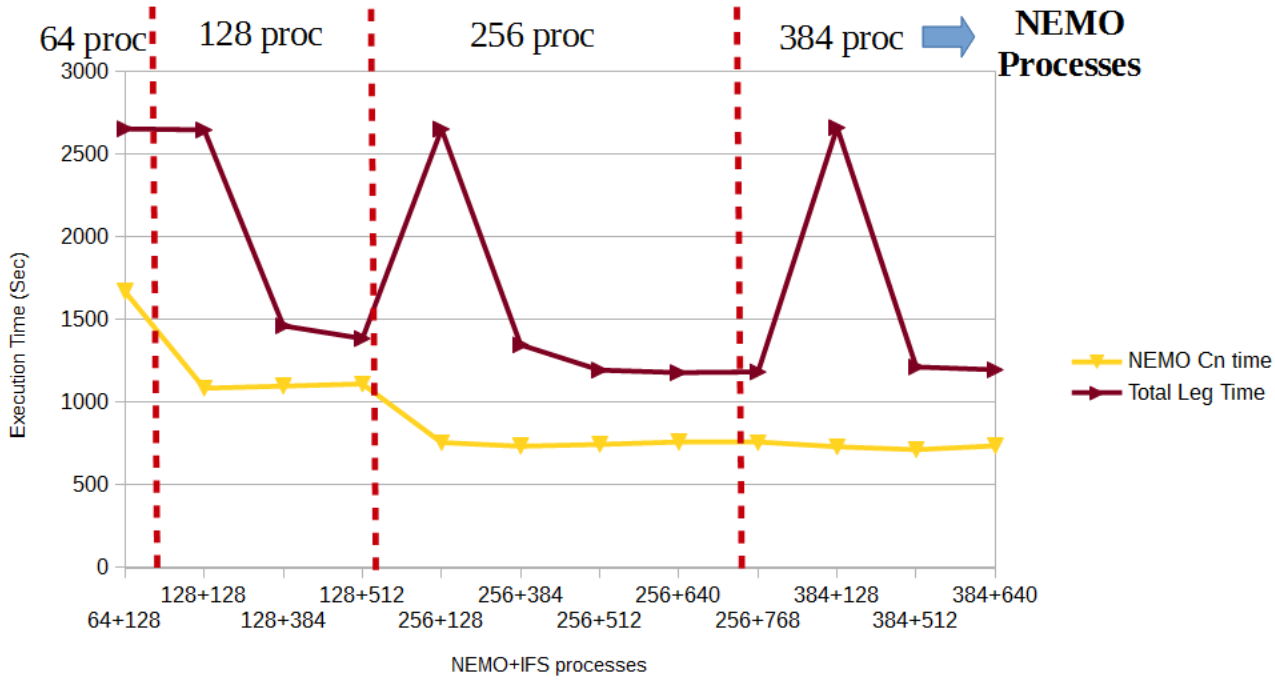


Figure 6. Timings (seconds) for different combinations of NEMO and IFS processes. Each block contains, for a fixed number of NEMO processes, different combinations of IFS and NEMO processes. It shows the NEMO Cn (yellow line) and the total execution time (red line).

It is possible to understand the computational cost of each model considering figure 7, which shows the results for the block three (256 processes for NEMO) and five combinations for IFS (128, 384, 512, 640 and 768). The figure shows that as the NEMO *Cn* time is shorter than the IFS *Cn* time, the IFS *En* time is close to 0. Also, the NEMO *En* time follows the same tendency as the IFS *Cn* time because NEMO has to wait for IFS.

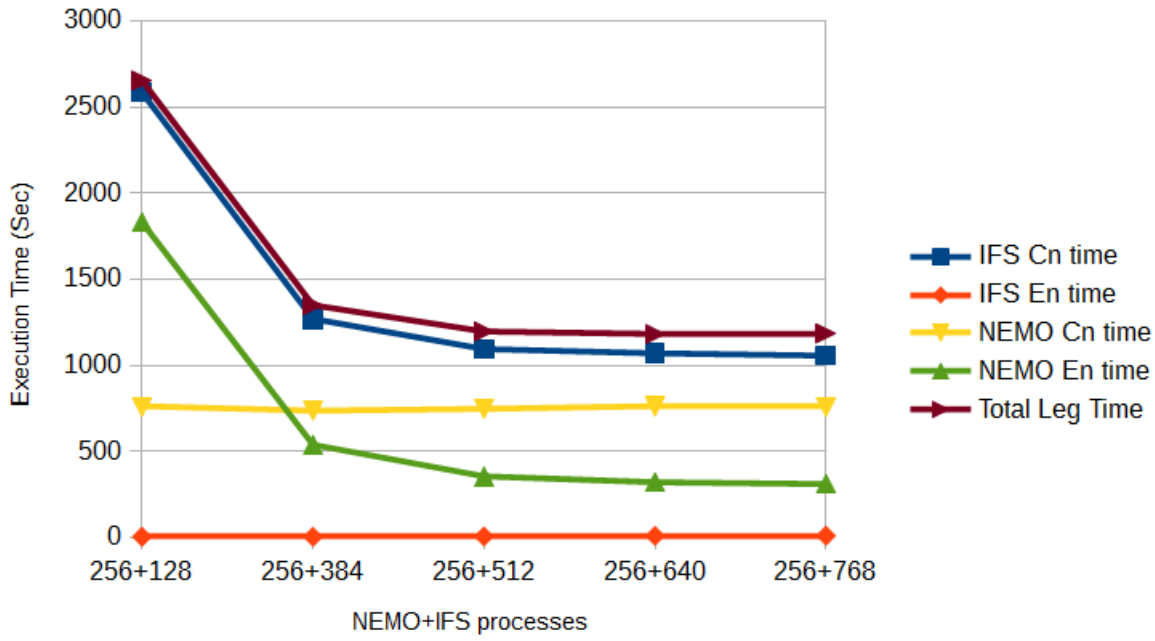


Figure 7. Execution time (seconds) for 256 processes for NEMO and five different numbers of processes for IFS. It shows the IFS Cn and En time (blue line and orange time respectively), the NEMO Cn and En time (yellow line and green time respectively) and the total execution time (red line).

6. Coupling performance analysis

A deep profiling analysis of the coupling is required to detect bottlenecks and hotspots in the parallel execution of the model. This will allow optimizing the implementation according to the possible problems presented.

To analyse the computational performance of the coupling on a HPC platform, the set of performance tools developed at the Computer Science department of the BSC were used. This suite is open-source and includes a tool to collect execution data (Extrae) and a tool to visualize the performance data collected (Paraver).

Figure 8 shows a parallel execution of EC-Earth using 128 MPI processes for IFS and 64 processes for NEMO and figure 9 using 512 processes for IFS and 128 for NEMO. Each trace shows two complete time-steps (one calculating the IFS radiation and another one without radiation). The traces show that the coupling time (red color) increases with the number of processes used for IFS (the coupling transformations are done in this case in IFS at the end of each time step). The results show that using 128 processes for IFS the coupling time is small comparing to other calculations, but using 512, the coupling time increases and becomes very significant with respect to the total IFS time-step duration. This is due to several one-to-all/all-to-one communications done in the coupling transformations performed, at least partially, on IFS master process only. This implementation increments the overhead when more and more processes are used. Comparing the two figures, the coupling from IFS to NEMO represents less than 15% of the execution time for a time step without radiation and 128 IFS processes but goes up to more than 50% when 512 processes are used.

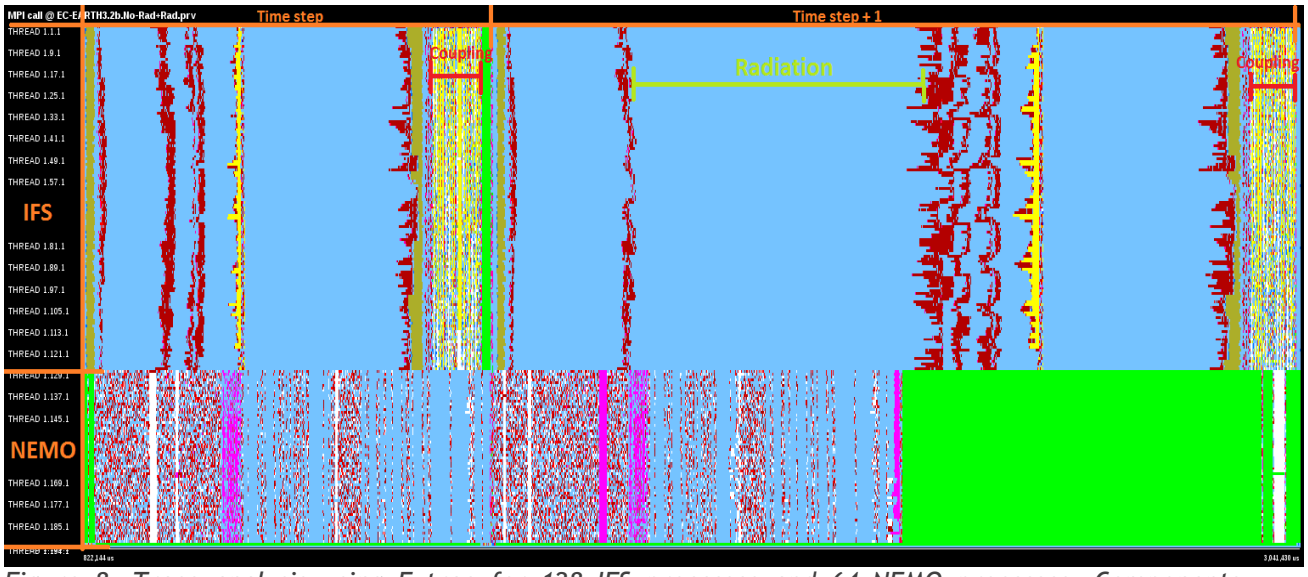


Figure 8. Trace analysis using Extrae for 128 IFS processes and 64 NEMO processes. Components, Radiation, Coupling and start and end for each time step are shown.

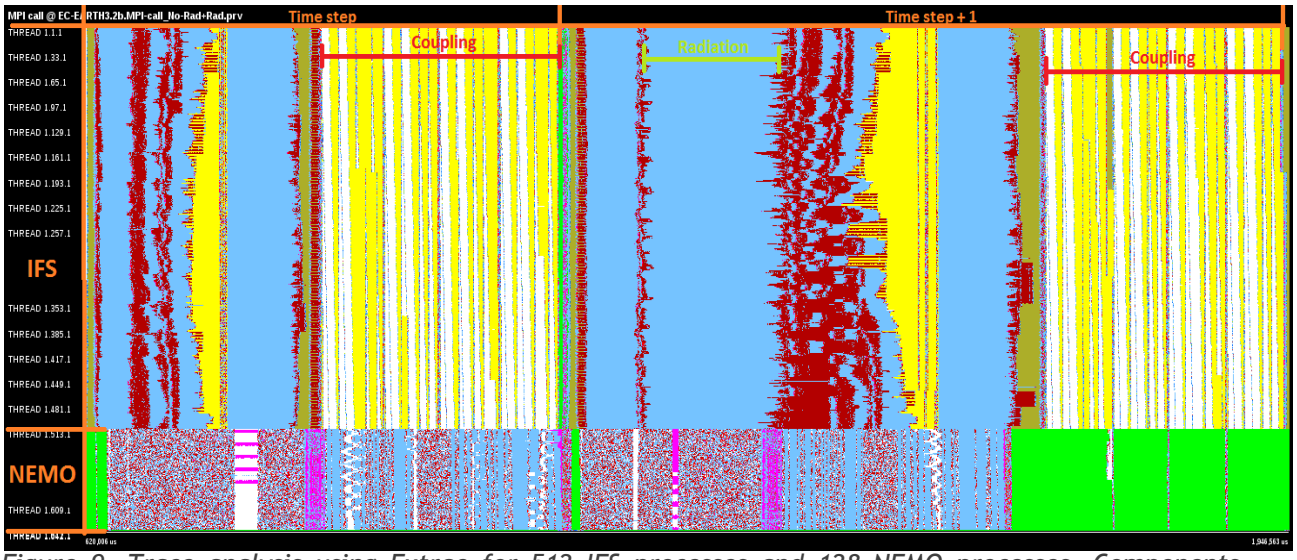


Figure 9. Trace analysis using Extrae for 512 IFS processes and 128 NEMO processes. Components, Radiation, Coupling and start and end for each time step are shown.

6.1. Proposed alternatives

According to the performance analysis, the coupling from IFS to NEMO in EC-Earth could represent a bottleneck in the parallel execution of EC-Earth. Activation of some optimisations available in the OASIS3-MCT coupler were suggested by CERFACS and are studied in detail hereafter. The following aspects are considered:

1. Algorithm implementation. Performance when OASIS3-MCT "dft" and "opt" options for global conservation is used.

2. Optimized coupling exchanges. Performance when all fields of the same type are packed or not.

3. See future work for more proposals in progress.

6.1.1. Algorithm implementation

The MPI implementation of the global conservation available in OASIS3-MCT ("CONSERV" post-processing operation) and used in EC-Earth has been studied. Figure 10 shows the algorithm implementation. This global conservation operation ensures that the total integrated value of a coupling field is the same in both the source and target grids. To accomplish this, the total sum on each grid is computed and the difference is uniformly distributed on the target grid.

Figure 10 (section 1) shows the implementation using the default "bfb" option: the algorithm includes a serial part, some one-to-all/all-to-one communications and broadcasts. The algorithm corresponding to an alternative partially parallel implementation is also shown in

Figure 10 (section 2). Using this option, one-to-all/all-to-one communications are substituted for communications between neighbours, in order to solve the needed dependencies to do each subdomain transformation directly per subdomain. In this option, only one operation is serial (Source-Target) and only one broadcast is necessary to communicate the difference, exploiting the bandwidth to do all broadcasts at the same time. In this case, the global conservation operation is done only for a master process, which has to communicate the data to other subdomains. Finally, figure 10 (section 3) shows a parallel implementation provided by OASIS using the "opt" option. In this case, there is no master process and the calculations are done redundantly, removing the final broadcast. Note that as detailed here, the "opt" option is more efficient than the "bfb" and intermediate algorithms but does not ensure bit-for-bit reproducibility when the grid decomposition or number of processes of the component are changed, which is only achieved for the "bfb" alternative.

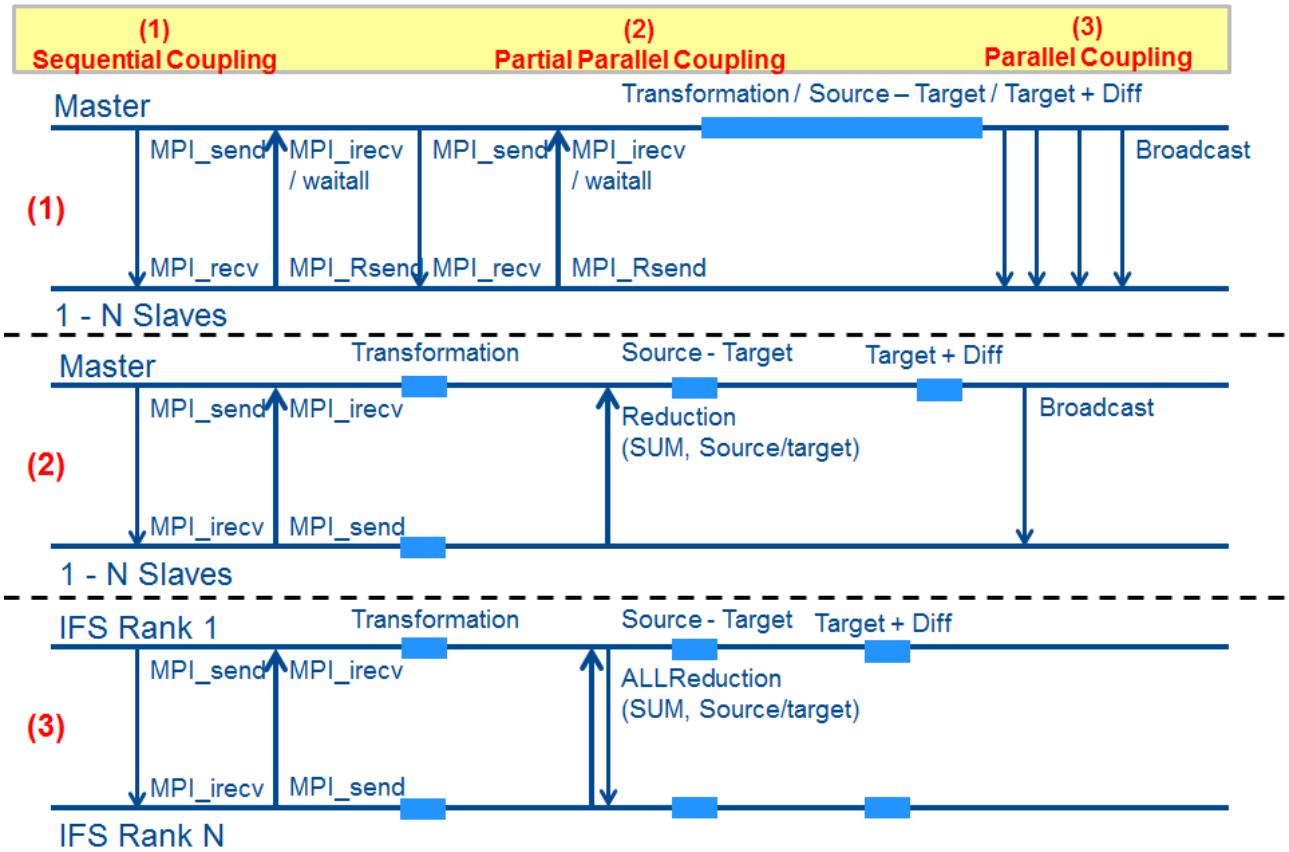


Figure 10. Different algorithm implementations for the global conservation operation with OASIS3-MCT in EC-Earth: (1) default, (2) alternative using a master process for global conservation operation, (3) parallel implementation using redundant computation.

Performance results using “opt” option for CONSERV operation

OASIS3-MCT includes two alternative implementations for some operations. For example, as explained above, communications all-to-one/one-to-all using MPI_send/MPI_irecv are substituted by collective and non-blocking communications MPI_igather/MPI_iscatter when the “opt” option is activated in the namcouple file for the global conservation operation (“CONSERV” post-processing operation).

The “opt” option for global CONSERV operations has been tested for EC-Earth. For this test, a modified namcouple has been used where the “opt” option has been added.

Figure 11 shows a parallel execution of EC-Earth similar to that in figure 9, using 512 MPI processes for IFS and 128 processes for NEMO. Each trace shows two complete time-steps (one calculating IFS radiation and other without radiation). The traces show that the coupling time (red color) is reduced dramatically when “opt” is used. The green color represents waiting time, which is now equivalent to the time used for coupling without “opt”.

The results show that coupling time has been reduced by more than 90% compared to figure 9. The reason is that all-to-one/one-to-all MPI communications have been changed for global communications (gather/scatter and reduction) and the coupling calculation is done across all IFS processes instead of using only one IFS master process for these calculations. Since the coupling time has been reduced considerably, IFS processes have now to wait until NEMO finishes its calculations (see the new green interval shown in figure 11). The green interval represents approximately the execution time reduced for coupling in the time step when radiation is not calculated.

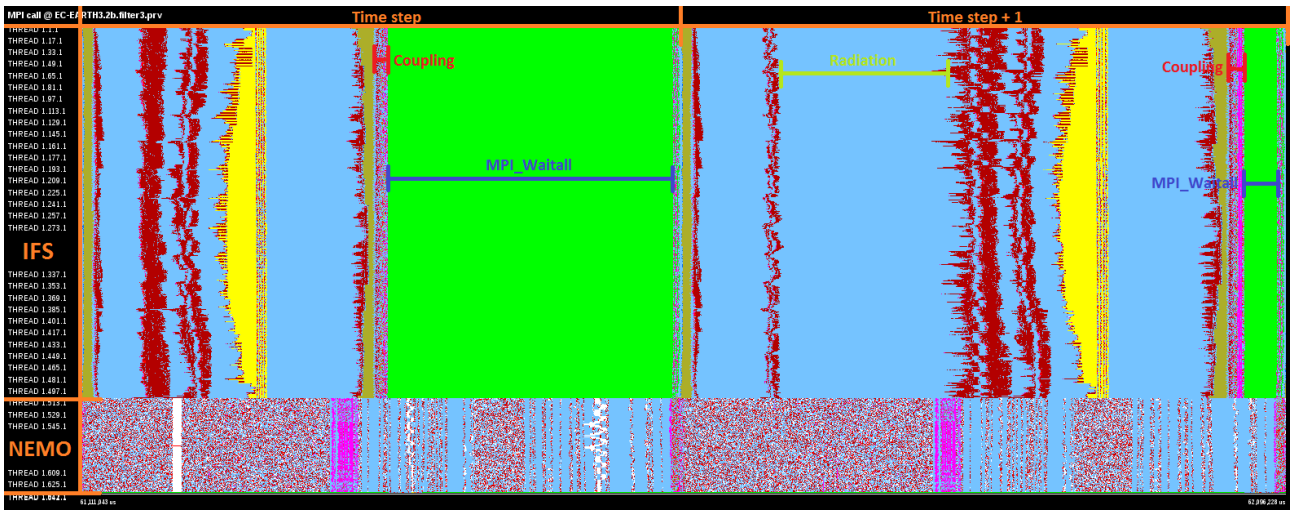


Figure 11. Trace analysis using Extrae for 512 IFS processes and 128 NEMO processes. Components (orange colour), Radiation (green colour), Coupling (red colour) and start and end for each time step (amber colour) are highlighted.

Execution time has also been measured using LUCIA. Table 4 shows *Cn*, *En*, *Jn* and final

execution time for simulations of three months using 512 processes for IFS and 128 for NEMO. The results show that all times are reduced, mainly the *Cn* time for IFS, where a global conservation operation from IFS to NEMO is done. Using the "opt" option the final execution time is reduced by around 40% compared to the original coupling used by default. This can be observed taking into account the increment of *Jn* (more than 550 seconds), equal to the MPI_Waitall function, when IFS is waiting for NEMO.

Note that this improvement in the final execution time is significant when coupling time for IFS is representative, using for example 512 processes for IFS (compare figures 8 and 9). On the other hand, the improvement in the final execution time is only possible when the slowest component is optimized. For example, figure 2 shows that if component 2 reduces the computation and/or coupling time, the waiting time of this component will increase, but the final execution time would be the same, which is equal to the total execution time of component 1, the slowest component. For these tests, where both components are balanced using the original namcouple, the modified namcouple makes IFS faster than NEMO, creating a lack of balance between the two components that reduces the possible improvement in the final execution time. In any case, the optimization could be used to reduce the IFS execution time when IFS is slower or the number of MPI processes when it is faster.

	Cn Time IFS (s)	*En Time IFS (s)*	*Cn Time NEMO (s)*	*En Time NEMO (s)*	*Inter. Time IFS (s)*	*Jn Time IFS (s)*	*Jn Time NEMO (s)*	*Final Time (s)*
Original namcouple	*1594.17*	*477.99*	*1889.66*	*186.80*	*276.37*	*186.11*	*209.33*	*2169*
Modified namcouple	*1409.12*	*598.43*	*1796.02*	*212.50*	*67.31*	*615.69*	*1235.49*	*2048*

Table 4. Execution time for each stage of IFS and NEMO using the LUCIA tool as the average of three executions of chunks of four months.

Reproducibility Test

Simulation results using the original namcouple and the modified namcouple activating option "opt" have been compared to evaluate if the optimizations change the results calculated by EC-Earth. The methodology applied will be published, but until then it can be found in https://dev.ec-earth.org/attachments/download/815/20160526_EC-Earth3.2_MarioAcosta.pdf.

The reproducibility test is divided into two parts. In the first one a Reichler-Kim normalized index (e2) for 13 variables is calculated (see Reichler, T., and J. Kim (2008): How Well do Coupled Models Simulate Today's Climate? Bull. Amer. Meteor. Soc., 89, 303-311 for more details). The normalized error variance e2 is calculated for each variable by squaring the grid-point differences between the simulated and observed climates. The normalized index is then compared between the two experiments to evaluate if the mean climate results produced by

two experiments are similar, taking into account the chaotic nature of multi-member experiments, where differences among perturbed tests up to 5% are expected.

In the second part, both experiments (using original and modified namcouple) are compared directly using a Kolmogorov-Smirnov test. So in this case it is possible to evaluate if the solutions obtained for the two experiments are similar or not. More information about these two tests and a complete explanation can be found in the link above.

Two experiments have been done, using the default (a0bd) and the modified namcouple (a0bk) for one year and with 5 members (5 simulations using different perturbed initial conditions for the ocean and ice).

The first test compares the results of each experiment to simulation results obtained for CMIP5, obtaining a normalized index for each variable evaluated. These normalize indexes for both experiments are shown in figure 12, not only to evaluate how close to the CMIP5 results are (the more closed to 1.0, the more similar to CMIP5 results they are), but also to check if a similar precision is obtained for both experiments. Simulations statistically different from the others, according to the test, would be shown in red and similar in blue. This test shows that the default configuration results are as similar to CMIP5 results as the results using the modified namcouple.

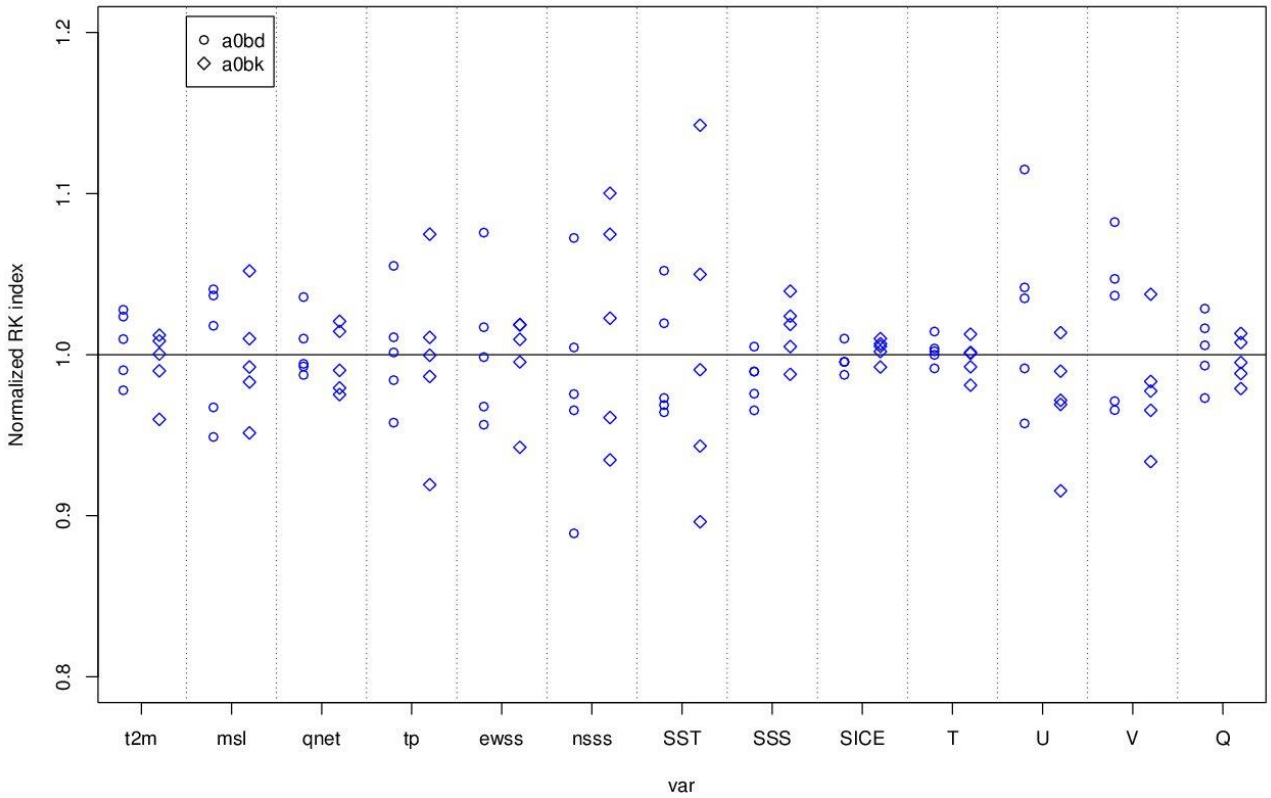


Figure 12. Reichler Kim normalized index for 13 variables of five-member ensemble simulations for the two experiments, using circles for the experiment using the optimized namcouple (a0bd) and the modified namcouple using diamonds (a0bk). Simulations statistically different according to the test

would be shown with red symbols.

For the second test, all the variables have been compared directly between them. Figure 13 shows an example for temperature (t2m) where less than a 1% of the values show significant differences between both experiments. Similar results are obtained if other variables are compared. The results show that the "opt" option does not produced results statistically different. Experiments compared in the past proved that for parallel executions using EC-Earth, differences of 1-2% are expected. This means that the small differences obtained are similar to the differences obtained when two identical experiments are run in parallel, taking into account that some parallel operations such as the reduction introduce some round-off differences (note that the possible methods to achieve bit-for-bit reproducibility for EC-Earth are not exploring in this study).

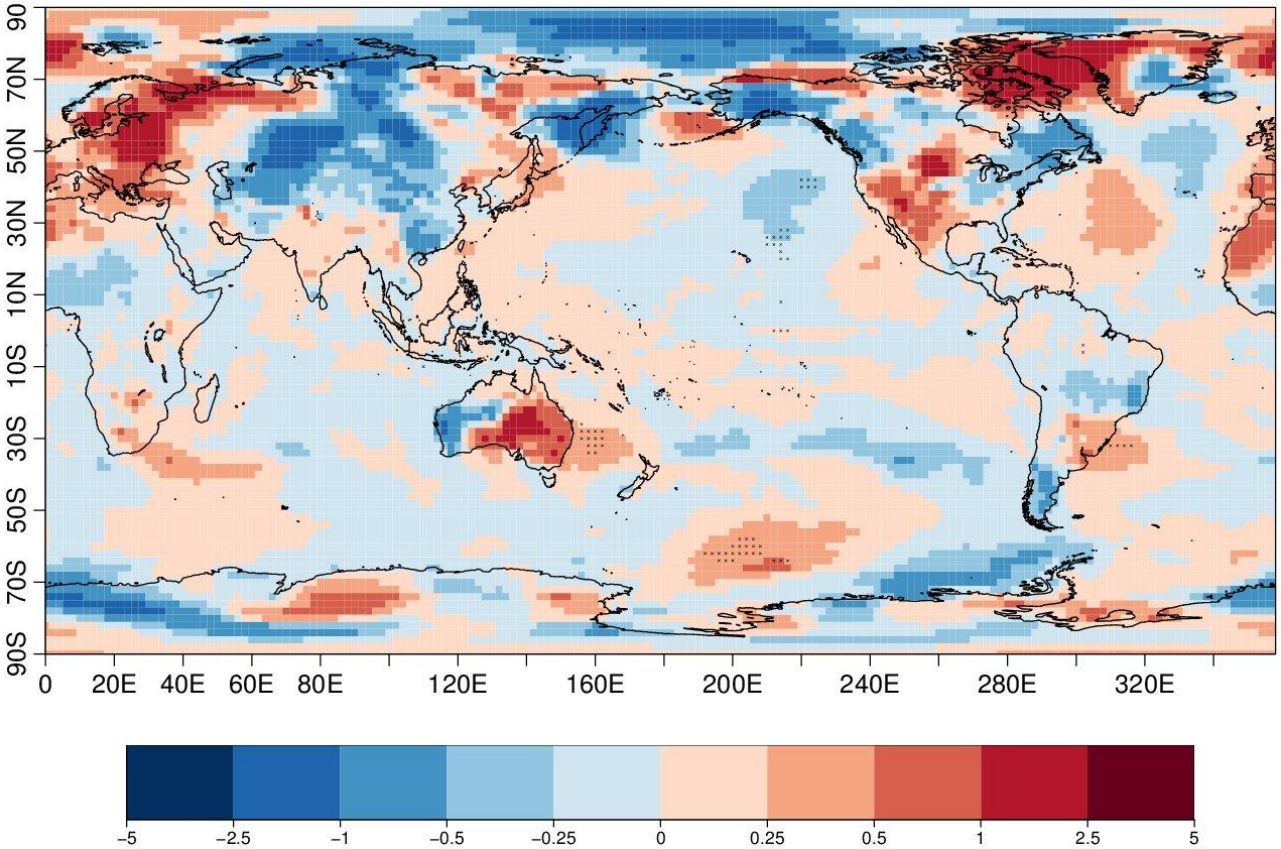


Figure 13. Differences in near-surface air temperature between the five-member ensemble experiments a0bd and a0bk. Black dotted regions indicate where the difference is significant according to a Kolmogorov-Smirnov test.

6.1.2. Optimized coupling exchanges

The OASIS3-MCT namcouple configuration file used by default for EC-Earth organizes fields sent and received among IFS, NEMO and Runoff mapper according to different categories. For

example, those fields sent from IFS to NEMO and Runoff mapper are organized in six groups (table 5), where each group contain one or more fields and the coupling can assure conservation (from A to D) or not (groups E and F).

Group	*From IFS to*	*Category*	*Type*
A	*NEMO*	*Stresses for oce and ice*	*LOCTRANS SCRIPR CONSERV GLOBAL*
B	*NEMO*	*Solar/non-solar radiation over ocean+ice*	*LOCTRANS SCRIPR CONSERV GLOBAL*
C	*NEMO*	*Precipitation and evaporation*	*LOCTRANS SCRIPR CONSERV GLOBAL*
D	*Runoff mapper*	*Runoff*	*LOCTRANS SCRIPR CONSERV GLOBAL*
E	*NEMO*	*dQns/dT (ice)*	*LOCTRANS SCRIPR*
F	*NEMO*	*Solar/non-solar radiation over ice*	*LOCTRANS SCRIPR*

Table 5. Groups of fields sent from IFS to NEMO.

OASIS3-MCT is able to pack several fields at the same time when the coupling operations are done (i.e. stresses for ocean and ice are packed in the same group). For EC-Earth, this post-processing coupling (at the end of each time step) is done separately and sequentially for each group, without more communication or computation in the meantime. This implementation invites to pack those groups with similar type of coupling together. This will use efficiently the bandwidth when the MPI communications are done and reduce the overhead produced by the coupling. As it is shown in table 5, groups (A-B-C) and (E-F) use the same coupling. A modified namcouple was developed including all the categories using the same coupling in a single group (table 6). A similar process was done for communications from NEMO to IFS.

Group	*From IFS to*	*Category*	*Type*
A	*NEMO*	*Stresses for oce and ice, Solar/non-solar radiation over ocean+ice and Precipitation and evaporation*	*LOCTRANS SCRIPR CONSERV GLOBAL*
B	*Runoff mapper*	*Runoff*	*LOCTRANS SCRIPR CONSERV GLOBAL*
C	*NEMO*	*Solar/non-solar radiation over ice and dQns/dT (ice)*	*LOCTRANS SCRIPR*

Table 6. Groups of fields sent from IFS to NEMO using a modified namcouple.

Figure 14 shows the impact of the optimized coupling field exchanges presented in table 6.

Figure 14(1) represents post-processing coupling from IFS to NEMO and runoff mapper, using the packing set by default (table 5) and showing only the conservative operations for groups from A to D (E and F groups are too small to be correctly represented). Figure 14(2) represents the same post-processing conservative operations, using in this case the optimized coupling field exchanges (table 6), reducing the computational work of this process a 40% approximatively.

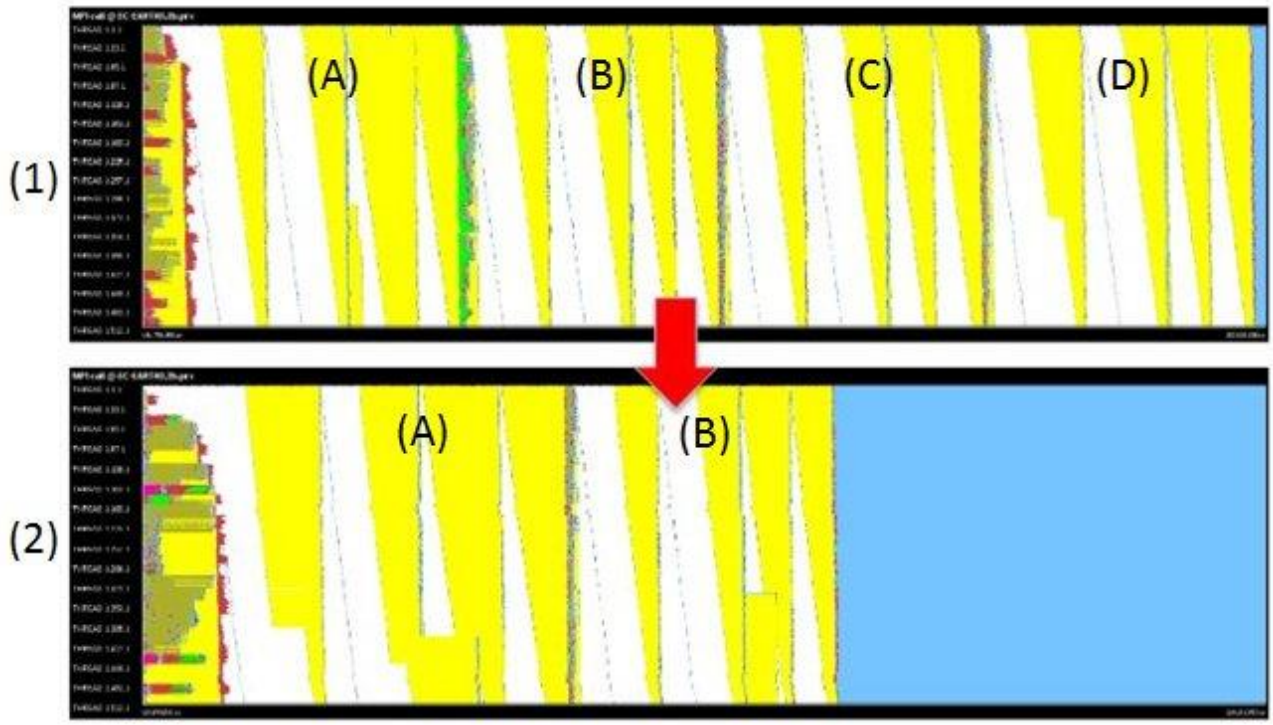


Figure 14. Trace analysis using Extrae for 512 IFS processes and 128 NEMO processes. It shows only the post-processing coupling in IFS in one time step. (1) Using original namcouple and (2) using the modified namcouple activating optimized coupling exchanges.

The execution time has also been measured using LUCIA. Table 7 shows *Cn*, *En*, *Jitter* and final execution times for chunks of four months using 512 processes for IFS and 128 for NEMO. The results show that all times are reduced, mainly *Cn* for IFS, where the global "CONSERV" post-processing operation is done. Taking into account these results, using modified namcouple shown in figure 14(2) the final execution time is reduced by 6.5% approximately.

	Cn Time IFS (s)	*En Time IFS (s)*	*Cn Time NEMO (s)*	*En Time NEMO (s)*	*Inter. Time IFS (s)*	*Jn Time IFS (s)*	*Jn Time NEMO (s)*	*Final Time (s)*
Original Namcouple	*1594.17*	*477.99*	*1889.66*	*186.80*	*276.37*	*186.11*	*209.33*	*2169*
Modified Namcouple	*1511.77*	*555.44*	*1920.04*	*148.13*	*184.99*	*62.03*	*12.54*	*2114*

Table 7. Execution time for each stage of IFS and NEMO using the LUCIA tool as an average of three executions of chunks of four months.

Reproducibility Test

Simulation results using or not optimized coupling exchanges have been compared to evaluate if the optimizations change the results calculated for EC-Earth. The methodology applied has been explained in 6.1.1 section.

The experiments have been done for one complete year and using five-member ensembles. Two experiments have been done, one using the original namcouple (a090) and another one with the modified namcouple using the field gathering optimization (a08z).

The first test compares the results of each experiment to simulation results obtained for CMIP5, obtaining a normalized index for each variable evaluated. These normalize indexes for both experiments are shown in figure 15, not only to evaluate how close to the CMIP5 results are (the more closed to 1.0, the more similar to CMIP5 results they are), but also to check if a similar precision is obtained for both experiments. Simulations statistically different from the others, according to the test, would be shown in red and similar in blue. This test shows that the default configuration results are as similar to CMIP5 results as the results using the modified namcouple.

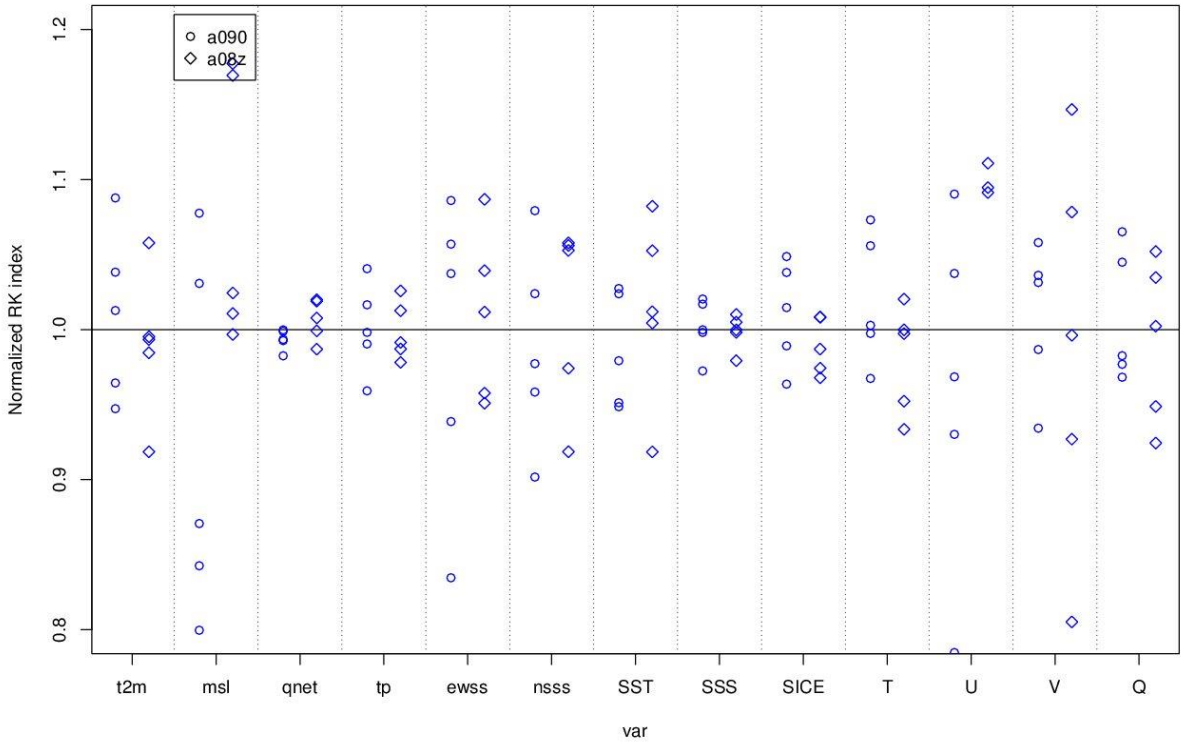
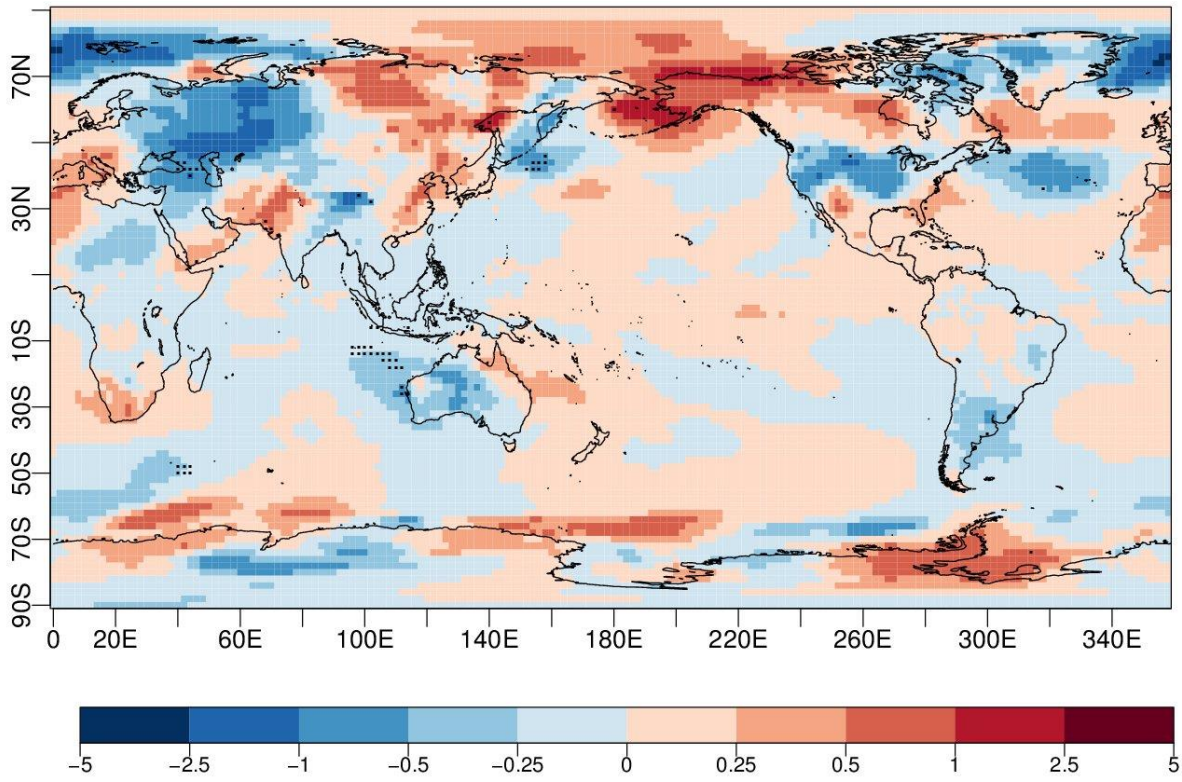


Figure 15. Reichler Kim normalized index for 13 variables of five-member ensemble simulations for the two experiments, using circles for the experiment using the optimized namcouple (a090) and the modified namcouple using diamonds (a08z). Simulations statistically different according to the test

would be shown with red symbols.

On the other hand, for the second test all the variables have been compared directly between both experiments. Figure 16 shows an example for near-surface air temperature where less than a 1% of the values show significant differences between both experiments. The results are similar for other variables. The results show that the modified namcouple does not change the precision of the results since experiments compared in the past prove that in standard parallel executions using EC-Earth, differences of 1-2% are expected. This means that the small differences obtained are similar to the differences obtained when two identical



experiments are run.

Figure 16. Differences of near-surface air temperature between the five-member ensemble experiments a090 and a08z. Black dotted regions indicate where the difference is significant according to a Kolmogorov-Smirnov test.

7. Conclusions

This document studies some aspects of the computational cost that coupling may represent for climate models. An analysis is done for the coupling between the two main components of EC-Earth, the atmospheric and ocean component (IFS and NEMO) via OASIS3-MCT. Several tests have been done to evaluate the coupling and to obtain several conclusions:

- Finding a load balance between the main components of EC-Earth (IFS and NEMO) is mandatory to optimize the resources used. The LUCIA tool provided by OASIS3-MCT could be used for this purpose.

- The optimization of one component (i.e. the reduction of the execution time of IFS) does not ensure the reduction of the execution time of EC-Earth if there are other slower components.

- The scalability analysis of each component can be calculated independently, similar to the process used in Section 5. This method should be used to facilitate the scalability analysis of coupled models and find the load balance between components at the same time.

- The final execution time of the model is not optimal when only the *Cn* LUCIA metric is used to find the load balance. This happens because LUCIA does not provide sufficient information, because it aggregates coupling time steps that don't have the same durations (IFS radiation extra time per four time steps). But LUCIA provides a second information which have to be into account for this cases: both IFS and NEMO *En* times are not zero, which means that something happen with the load balance. The results show that the final execution time of the model is lower when the regular time step of NEMO and the time step of IFS without radiation are considered to find the load balance. LUCIA developers are implementing an improved detailed LUCIA analysis, provided for each coupling time step. This will be available in the next OASIS release.

- The coupling time (which includes interpolation, other transformations and communication) can be very high when the global conservation method is applied without activating the optimization option. The results show that using 128 processes for IFS the coupling time is small compared to other calculations, but when using 512 the coupling time increases and becomes very significant with respect to the total IFS time-step duration. This is due to several one-to-all/all-to-one communications done in the coupling transformations performed, at least partially, on the IFS master process only. This implementation increases the overhead when more and more processes are used. It represents more than a 50% of time execution when 512 processes are used.

- Coupling field gathering, an option offered by OASIS3-MCT, can be used to optimize coupling exchanges between components. The results show that gathering all the

fields that use similar coupling transformations reduces drastically the coupling overhead. This happens because OASIS3-MCT is able to do communications and interpolation of all the fields gathered at the same time. A reproducibility test has been done and shows that, although the results are different when performing identical experiments, there are no significant differences when the optimized coupling exchanges are activated. The differences between the experiments are less than 1%.

- The "opt" option of OASIS3-MCT can be used to activate an optimized global conservation transformation. The results show that when using this option, the coupling time from IFS to NEMO is reduced by 90%. The reasons are that all-to-one/one-to-all MPI communications are replaced by global communications (gather/scatter and reduction) and that the coupling calculations are done by all IFS processes instead of only the IFS master process. The drawback of the "opt" option is that it does not ensure bit-for-bit reproducibility when the grid decomposition or number of processes of the component are changed. A reproducibility test has been done and shows that there are not significant differences when the "opt" option is used. As in the previous point, the differences are less than 1%.

Future work

High resolution tests. The modifications introduced here should be tested for grids at higher resolution in order to evaluate the coupling performance. At higher resolution, the computation time will change and the coupling overhead could be more important. Preliminary tests using Paraver show that using T511-ORCA25 resolution. The postprocessing coupling is larger than in these tests using standard resolution, representing up to 60% of the execution time of IFS.

Load balance study. Taking into account points 1, 2 and 4 of the conclusions, a load balance study is as important as optimize one component of an Earth System Model. The main objective is try to achieve quickly the load balance of EC-Earth and, in case of optimizing only one component, achieve quickly a new load balance to take advantage of the optimizations.

"dst" option study. Coupling transformations could be done on the target grid instead of the source grid. This is one of the advantages of OASIS to reduce the execution time of the slowest component and achieve load balance among components. This option should be explored.