

**Issue** RP 0097 July 2010

Scientific Computing and Operation (SCO)

# A PERFORMANCE EVALUATION METHOD FOR COUPLED MODELS

By Italo Epicoco University of Salento, Italy italo.epicoco@unisalento.it

Silvia Mocavero CMCC silvia.mocavero@cmcc.it

and Giovanni Aloisio CMCC University of Salento, Italy giovanni.aloisio@unisalento.it

**SUMMARY** In the High-Performance Computing context, the performance evaluation of a parallel algorithm is made mainly considering the elapsed time running the parallel application with both different number of cores or different problem sizes (for scaled speed-up). Typically, parallel applications embed mechanisms for efficiently using the allocated resources, guarantying for example a good load balancing and reducing the parallel overhead. Unfortunately, this assumption is not true for coupled models. These models are born from the coupling of stand-alone climate applications. The component models are developed independently from each other and they follow different development roadmaps. Moreover, they are characterized by different levels of parallelization, different requirements in terms of workload and they have their own scalability curve. Considering a coupled model as a single parallel application, we can note the lacking of a policy for balancing the computational load on the available resources. This work tries to address the issues related to performance evaluation of a coupled model, and to answer to the following questions: allocated a given number of processors for the whole coupled model, how to configure the run in order to balance the workload? How many processors must be assigned to each of the component models? The methodology here described has been applied for evaluating the scalability of the CMCC-MED coupled model designed by INGV and the ANS Division of the CMCC. The evaluation has been carried out on two different computational architectures: a scalar cluster based on IBM Power6 processors; and a vector cluster based on NEC-SX9 processors.

Keywords: HPC; Performance model; Porting; Coupled models

The work here presented carried out with support of the funding of the european project IS-ENES.

**JEL:** C63

## INTRODUCTION

Different technical solutions are used within the ESM (Earth System Modeling) community to couple geophysical model codes. Two main approaches, besides hard-coding, can be cited: using an external entity, for the transformation of the coupling fields, and linking its communication libraries within applications sometimes referred to as "coupler" approach; using the coupling libraries or functions to build an integrated coupled application based on basilar scientific units, sometimes referred to as "framework" approach. The different implementations of coupled models in the community lie in the continuum between those two approaches. In the former approach the component models preserve the original codes almost unchanged and interface each other with a communication library. The component models are coupled through an external entity for transforming the coupling fields (i.e. the OASIS coupler [11]). This approach results also in a parallel program launched through a MPMD (Multiple Program Multiple Data) [3] approach.

The "framework", or ESMF [1] [6], approach foresees a revision of the already developed software modules in order to adapt code data structure to the calling interface of the framework, to split the original code into elemental units, to write or use coupling units and to use the library to build a hierarchical merged code. This approach often results in a single parallel program launched with a SPMD (Single Program Multiple Data) [3] approach.

For both approaches, the final user (that often is not a computer scientist) must specify how the allocated cores should be spread in order to balance the computational load required by the component models.

A parallel algorithm is mainly evaluated considering as metrics: the parallel efficiency; the parallel speed-up; and, if we take into account also

the problem size, the scaled speed-up. The coupled models differ from the typical scientific parallel application mainly because each of the component models is designed independently from the others; they are characterized by different levels of parallelization, different requirements in terms of workload and they have their own scalability curve. During the design of a coupled model, the effort is mainly focused on the development of the interfaces among the component models and often they lack of automatic or dynamic load balancing policies. This implies that the user or modeler has to statically configure the coupled model parameters for balancing the workload among the allocated computational resources. If the total number of processors changes, then the parameters must be accordingly re-tuned.

This work tries to address the issues related to the performance evaluation of a coupled model, and to answer the question: established the total number of cores allocated for the whole coupled model, how we have to distribute them for each of the component models?

The report is organized in three parts: we firstly describe the methodology we have followed for measuring the performances of a coupled model; thereafter we report the results of our analysis performed on a real coupled model used as a case study; concluding with overall considerations and future direction of the research.

## **METHODOLOGY**

During the analysis of scalability of a coupled model, the main issue is how to find the best configuration among the component models in order to obtain a balanced run. The best balancing among different component models can be easily defined if an analytic performance model is provided for each of them. Unfortunately this can not be guaranteed every time. In these cases, an experimental analysis for building the scalability curve of each component is necessary. It is worth noting here that the proposed methodology can be applied when an analytic performance model of the coupled applications can not be obtained because the components are provided as black boxes or because the definition of an analytic performance model of the components is out of the scope of the research. Four steps compose the methodology we propose:

- Experimentally analyzing each component model, including the coupler, for building the corresponding scalability curves
- Defining an analytic performance model, at coarse grain level, of the whole coupled model. The performance model must take into account the relationship between each component model and the coupler
- Using the experimental data given during the stage 1, and evaluating the model defined in stage 2, the best configurations must be extracted for different numbers of available cores
- Experimentally evaluation of the behavior of the coupled model considering only the best configuration for a given number of allocated cores.

#### **CMCC-MED CASE STUDY**

The methodology has been applied to the CMCC-MED [4] [5] model, developed under the framework of the EU CIRCE Project (Climate Change and Impact Research: the Mediterranean Environment). It provides the possibility to accurately assess the role and feedbacks of the Mediterranean Sea in the global climate system. From a computational point of view, it represents a typical coupled model with a MPMD approach.

The present section is organized as follow: the description of the coupled model with its components is given; the design of the experiment and the description of the computing environment are presented; thereafter for each of the four stages of the methodology, a detailed description of the results is presented.

#### **MODEL DESCRIPTION**

The CMCC-MED is a global coupled oceanatmosphere general circulation model (AOGCM) coupled with a high-resolution model of the Mediterranean Sea. The atmospheric model component (ECHAM5) [9] has a horizontal resolution of about 80 Km with 31 vertical levels, the global ocean model (OPA8.2) [8] has a horizontal resolution of about  $2^{\circ}$  with an equatorial refinement (0.5°) and with 31 vertical levels, the Mediterranean Sea model (NEMO in the MFS implementation [7] [10]) has a horizontal resolution of  $1/16^{\circ}(\sim$ 7 Km) and 72 vertical levels. The communication between the atmospheric model and the ocean models is performed through the CMCC parallel version of OASIS3 coupler [2], and the exchange of SST, surface momentum, heat, and water fluxes occurs every 2h40m. The total number of fields exchanged through the coupler is 35. The connection between the global ocean and Mediterranean occurs through the exchange of dynamical and tracer fields via simple input/output operations. In particular, horizontal velocities, tracers and sea level are transferred from the global ocean to the Mediterranean model through the open boundaries in the Atlantic box. Similarly, vertical profiles of temperature, salinity and horizontal velocities at Gibraltar Strait are transferred from the regional Mediterranean model to the global ocean. The ocean-to-



ocean exchange occurs with a daily frequency, with the exchanged variables being averaged over the daily time-window.

# **EXPERIMENTS DESIGN**

In table 1 the compilation keys for each component model are reported.

Table 1

Compilation keys	for component	mod	els.
------------------	---------------	-----	------

Model Name	Compilation keys
NEMO Mediterranean Sea	key_dynspg_fit key_ldfslp key_zdfric key_dtasal key_dtatem key_vectopt_loop key_vectopt_memory key_oasis3 key_coupled key_fix_circe key_obc key_qb5 key_mfs key_cpl_discharge_echam5 key_cpl_ocevel key_mpp_mpi key_cpl_rootexchg key_useexchg
ECHAM5 Atmospheric	_cpl_opa_lim _prism _CLIM_Box _grids_writing _cpl_maskvalue _cpl_wind_stress
OPA8.2 Ocean Global	key_coupled key_coupled_prism key_coupled_echam5 key_coupled_echam5_intB key_orca_r2 key_ice_lln key_lim_fdd key_freesurf_cstvol key_zdftke key_flxqsr key_trahdfiso key_trahdfcoef2d key_dynhdfcoef3d key_trahdfeiv key_convevd key_temdta key_saldta key_coupled_surf_current key_saldta_monthly key_diaznl key_diahth key_monotasking

Each component model is used with the spatial and temporal resolutions shown in table 2, while the coupler OASIS3 has been configured as in table 3.

Table 2           Spatial and temporal resolution of the component models			
	<b>OAP8.2</b>	ECHAM5	NEMO
time step grid points vertical levels	4800s 182x149 31	240s 480x240 31	600s 871x253 72

#### Table 3 OASIS3 configuration

#### **OASIS3** Configuration

coupling pe number of f	riod 96 ields :	00s 35	
	Exported fields	Imported fields	LAG
OPA8.2	17	6	4800s
ECHAM5	9	26	240s
NEMO	9	3	600s

A detailed view of the transformations performed by OASIS3 on the exchanged fields is given in figure 1.



OASIS3 transformations over the exchanged fields.

# **COMPUTING ENVIRONMENT**

All of the experiments have been made on two different architectures available at the CMCC Supercomputing Centre: a scalar cluster based on IBM Power6 processors and a vector cluster based on NEC-SX9 processors. The IBM cluster, named Calypso, has 30 IBM p575 nodes, each of them equipped with 16 Power6 dualcore CPUs at 4.7GHz (8MB L2/DCM, 32MB L3/DCM). With Simultaneous Multi Threading (SMT) support enabled, each node hosts 64 virtual cores. The whole cluster provides a computational power of 18 TFLOPS of peak performance. Each node has 128GB of shared memory (4GB per core), two local SAS disks of 146,8GB at 10k RPM and two Infiniband network cards each one with four 4X IB galaxy-2 and four Gigabit network adapters. Some nodes are used as GPFS and TSM servers and have also two fibre channel adapters at 4Gb/s FC and two fibre channel adapters at 8Gb/s for interconnecting to the storage system. Calypso has 2 storage racks, each one equipped with 280 disks of 750GB, providing a total capacity of 210TB of raw storage accessible from the computational nodes through GPFS. Calypso interconnects also a tape library with 1280 cartridges LTO4 at 800GB (1PB total capacity) and Tivoli TSM for handling Hierarchical Storage Management. The default compilers are IBM XL C/C++, and IBM XL FORTRAN. The default resource scheduler manager is LSF. The NEC cluster, named Ulysses, has 7 nodes based on SX9 processors. Each node has 16 CPUs at 3.2GHz, 512GB of shared memory, a local SAS D3-10 disk of 3.4TB and uses IXS Super-Switch interconnection with a bandwidth of 32GB/s per node (16GB/s for each direction) to the high-speed interconnection and four 4Gb/s FC adapters to storage system. The whole cluster provides a computational power of 11.2 TFLOPS of peak performance. Ulysses has 3 storage racks with three SAS D3-10 disks at 9.2TB and three SAS D3-10 disks at 6.9TB for a total capacity of 48.3TB of raw storage. The GFS is used for handling the storage system. The default compilers are SX C/C++ and SX FORTRAN. The default resource scheduler

manager is NQSII.

# STAGE1: COMPONENT MODELS EVALUATION

As already mentioned, the components have been evaluated on both architectures. Before proceeding with the analysis on IBM Power6, a code porting activity has been needed. The porting on the IBM cluster consists of the following three steps:

- A1: Compilation, configuration and execution of the component models as they are executed on the vector cluster, without any code optimization
- A2: Analysis of bottlenecks and definition of component models to be improved in order to optimize the coupled model performance
- A3: Optimization of the coupled model as result of the previous activity, taking into consideration the target architecture and the availability of native libraries performing better w.r.t. those actually used.

The version of ECHAM5 included within the CMCC-MED coupled model is optimized for the NEC-SX9 cluster and it is characterized by several physical improvements introduced by CMCC-ANS division. Moreover, a stand-alone version of the atmospheric model provided by IBM and optimized for Power6 scalar architecture, is available. In order to maintain the optimizations of the stand-alone version, an integration of the physical changes within it has been started. To date, the ECHAM5 standalone version has been coupled within the CMCC-MED model and we are working on the integration of the physical improvements. The Mediterranean component NEMO at 1/16° has been developed by INGV (IOIPSL provides a

version of NEMO from  $1^{\circ}$  to  $1/12^{\circ}$ ).

The porting activity of the CMCC-MED model on IBM Power6 is currently at stage A2. During the stage A1, we used only the compiler optimization flags for tuning the model on the scalar architecture. Moreover, to compile the CMCC-MED on the IBM Power6 cluster, some modifications on the code were needed.

From a preliminary performance analysis of ECHAM5 and NEMO component models we observed that NEMO performances are limited by the communication overhead when open boundaries are activated and ECHAM5 does not scale well since we are using a version deeply optimized for vector clusters.

Several runs have been performed to evaluate the scalability of all of the model components. For each of these, we report the elapsed time for executing one-day simulation (it does not include the I/O time for writing the restart files).

Each model has been separately evaluated but within the coupled model. We used the PRISM libraries [11] for instrumenting the code and for extracting the elapsed time of the single component model. We have considered the time elapsing between a *prism\_get* and a *prism\_put* as the time spent by the model for simulating all the time steps occurred within a coupling period. The coupling time has been evaluated considering the elapsed time between a *clim\_import* and a *clim\_export*.

The components we have taken into account are ECHAM5, NEMO and the OASIS3 coupler that are the most computational intensive components. The OPA8.2 model is run in configuration ORCA2 using the sequential version. It is not been analyzed since it did not represent a bottleneck for any of the configurations taken into consideration.

All of the experiments have been performed using only MPI1 approach. Even if the ECHAM5 model supports a hybrid parallelization based on OpenMP/MPI, in our experiments the number of threads for process has been set to 1. In the following, we describe the analysis of scalability for each component model.

	Table 4           Compilation flags for ECHAM5			
	Compilation flags	linked libraries		
NEC- SX9	-pi exp=cuentr, cuad- jtq, cubasmc ex- pin=cuentr.f90, cuad- jtq.f90, cubasmc.f90 -Wf, -init heap=zero stack=zero -Popenmp -Chopt -sx9	libsupport.a lib- blas.a liblapack.a libpsmile.MPI1.a libmpp_io.a libnetcdf.a		
IBM Power6	-qsmp=omp -q64 -O5 - qstrict -qarch=pwr6 -qtune=pwr6 - qcache=auto -qfixed -qsuppress=1518- 061:1518-128 - qMAXMEM=-1 -Q -qsuffix=cpp=f90 -qzerosize -qessl - qnosave -qalias=nostd	libsupport.a libessl.a libmassv.a libla- pack3264.a libxlf90_r.a libpsmile.a libmpp_io.a libnetcdf.a		

The ECHAM5 component has been compiled on NEC-SX9 and IBM Power6 using the compilation flags in table 4. On NEC-SX9, ECHAM5 scales up to 28 processors. On IBM Power6 some problems occurred when using block domain decomposition. Using only 1D decomposition, with a resolution of T159 (corresponding to 240x480 grid points), the maximum number of processes is 60. Hence, the scalability of ECHAM5 on IBM Power6 has been evaluated up to 60 cores (figure 2).

The NEMO component model has been compiled on NEC-SX9 and IBM Power6 using the compilation flags in table 5.

On the vector cluster, NEMO model presents a not regular trend in the scalability. The scalability is limited to 24 vector processors. On IBM Power6, the model scales better up to 64 scalar cores (figure 3). It is worth noting here that the elapsed time on both machines differs of one order of magnitude.





Table 5           Compilation flags for NEMO			
	Compilation flags	linked libraries	
NEC- SX9	-size_t64 -dw -Wf"-A dbl4" -sx9 -pi auto -P stack -C vopt - Wf"-init stack=nan" -WI"-f nan" -Wf"-P nh" -Wf,-pvctl noassume loopcnt=10000	liboasis3.MPI1.a libpsmile.MPI1.a libmpp_io.a lib- clim.MPI1.a libnetcdf.a	
IBM Power6	-q64 -c -qfixed -qrealsize=8 - qstrict -qarch=pwr6 -qtune=pwr6 - qcache=auto -O5	liboasis3.MPI1.a libpsmile.MPI1.a libmpp⊥io.a lib- clim.MPI1.a libnetcdf.a	

The OASIS3 coupler has been compiled using the compilation flags in table 6.

The parallel approach used in the CMCC version of OASIS3 follows an embarrassing parallel algorithm. The scalability on both architectures shows that the communication overhead is negligible. The nonlinear trend of the scalability, corresponding to some number of processes, is due to the not balanced workload on different fields (figure 4). The parallel algorithm limits the scalability to the number of exchanged fields (in our case it is equal to 35).



Scalability of NEMO on IBM Power6 (blu line) and NEC-SX9 (red line).

Table 6           Compilation flags for OASIS3			
	Compilation flags	linked libraries	
NEC- SX9	-Pstack -pi auto nest=3 line=10000 exp=iminim, rmaxim, rminim, grid_search_bilin ex- pin=src/, libsrc/scrip -Ep -sx9 -Wf,"-P nh" -Wf,"-pvctl noassume loopcnt=5000000 vworksz=100M" -Wf,"- A idbl4" -Wf,"-msg o" -Wf,"-pvctl fullmsg" - Wf,"-L fmtlist transform map summary noin- clist" -Chopt -Wf,"-ptr byte"	libpsmile.a libanaisg.a libanaism.a libf- scint.a libscrip.a libclim.MPI1.a libmpp_io.a	
IBM Power6	-qrealsize=8 -q64 -qalias=nostd -O0 -qsuffix=cpp=F90	libpsmile.a libanaisg.a libanaism.a libf- scint.a libscrip.a libclim.MPI1.a libmpp_io.a	

# STAGE2: DEFINITION OF THE PERFORMANCE MODEL

For the definition of the analytic performance model, we can consider that all the model components are executed in parallel between two coupling steps. When a coupling step happens, each model sends their fields to the cou-

Centro Euro-Mediterraneo per i Cambiamenti Climatici



pler and waits for receiving from the coupler the fields coming from the other models. The coupler, only after having received all of the fields to be exchanged, performs their transformations, sends the changed fields to the models and waits for receiving fields at the next coupling step. During the coupling period all the models are synchronized and waits for coupler ending the transformations (coupler transformations and models elaboration could be overlapped using the OASIS3 utility named SEQ [11]. During the analysis of the CMCC-MED performance, this feature has not been used). The analytic model defines the execution time for a single coupling step as the maximum computing time spent by the component models plus the coupling time (the last time step of NEMO, performed between a coupling step and the next one, is overlapped with the coupling activity for modeler's choice):

$$T_{K} = max\{t_{N} * (n_{N} - 1), t_{E} * n_{E}\} + \alpha \quad (1)$$

$$\alpha = \begin{cases} max\{t_{N}, t_{O}\}, & \text{if } t_{E}n_{E} \le t_{N}n_{N}(2) \\ t_{O}, & \text{if } t_{E}n_{E} > t_{N}n_{N}(3) \end{cases}$$

where:

 $t_N$  is the computing time for executing a time step of NEMO component model

 $t_E$  is the computing time for executing a time

step of ECHAM5 component model

 $\boldsymbol{n}_N$  is the number of timesteps of NEMO component model

 $n_{E}$  is the number of timesteps of ECHAM5 component model

 $t_O$  is the computing time for one coupling step. Let  $p_E$ ,  $p_N$ ,  $p_O$  be the number of processes allocated respectively for ECHAM5 and NEMO component models and for the OASIS3 coupler, we have to find their values minimizing the execution time  $T_K$ . The constraint imposes that the total number of processors must be equal to the allocated processors K.

$$p_E + p_N + p_O + 1 = K \tag{4}$$

# STAGE3: DEFINITION OF THE BEST CONFIGURATIONS

During this stage we used the experimental results obtained in the stage 1 for evaluating the performance of the whole coupled model. The definition of the best configuration, given the number of processors K represents the solution of an optimization problem which aims to the minimization of the objective function  $T_k$ given the constraint on the total number of processes. The solution is exhaustively searched with a complexity of  $O(\frac{H}{6})$  where H is the maximum value for K. K has been set to the total number of CPUs/cores to be used and the execution time has been evaluated using the performance model for all of the permutations of  $p_E$ ,  $p_N$ , and  $p_O$  satisfying equation 4. Taking into account the time step intervals and the coupling period (reported on tables 2 and 3) we have  $n_N = 16$  and  $n_E = 40$ .

All of the considered configurations are reported in tables 7 and 8.

# STAGE4: COUPLED MODEL PERFORMANCE ANALYSIS

Considering the trend of the performance model, evaluated on the NEC-SX9 (figure 5),



Best configurations on NEC-SX9				
K	ECHAM5	NEMO	OASIS3	
12	6	4	1	
14	8	4	1	
16	10	4	1	
18	10	4	3	
20	12	4	3	
22	14	4	3	
24	14	4	5	
26	16	6	3	
28	18	6	3	
30	18	6	5	
32	20	8	3	
34	20	10	3	
36	20	10	5	
38	20	10	7	
40	22	12	5	
42	22	12	7	
44	22	12	9	
46	24	16	5	
48	22	12	13	
50	22	14	13	
52	24	14	13	
54	24	16	13	
56	26	24	5	
58	26	24	7	
60	26	24	9	
62	24	24	13	
64	26	24	13	
66	28	24	13	

Table 7

we can make the following considerations:

- The execution time could decrease with a greater number of processors. The actually availability of processors on the NEC-SX9 cluster limits the analysis of scalability to 4 nodes.
- Established the number of nodes to be used, the best performance have been obtained when the allocated nodes are fully used.

On the IBM Power6 cluster (figure 6) we can make the following considerations:

1. The limit of the scalability has been

Best configurations on IBM Power6				
K	ECHAM5	NEMO	OASIS3	
20	8	8	3	
24	8	12	3	
28	8	16	3	
32	12	16	3	
36	12	20	3	
40	12	24	3	
44	12	24	7	
48	16	28	3	
52	16	32	3	
56	16	36	3	
60	16	36	7	
64	20	40	3	
68	20	44	3	
72	20	44	7	
76	20	48	7	
80	20	52	7	
84	20	56	7	
88	20	60	7	
92	24	64	3	
96	24	64	7	
100	28	64	7	
104	24	64	15	

Table 8



reached at 104 cores. Even if the cluster provides a greater number of nodes, our analysis stopped at 3 nodes since we have reached the minimum elapsed time.

2. Established the number of nodes to be used, the best performance have been obtained not always when the allocated

nodes are fully used, i.e. with 3 nodes it is not necessary to use all 192 cores, being 104 enough (the elapsed time on 104 and 192 is the same).



The scalability of the coupled model has been experimentally evaluated varying the number of nodes and using the best configurations suggested by the performance model. The experiments take into account a 5 days simulation and include the I/O time for writing the restart files. The results confirmed the likelihood of the performance model with the real computational behavior of the coupled model.

# **CONCLUSIONS AND FUTURE WORK**

A methodology for analyzing the performances of a climate coupled model has been presented. CMCC-MED has been used as case study for validating the methodology on both architectures available at the CMCC Supercomputing Center. The analysis showed a good scalability of ECHAM5 on both architectures and an irregular behavior of NEMO on NEC-SX9. An interesting result is represented also by OASIS3. It is worth noting that for a high level of processes, it is more convenient to allocate more processes to OASIS3 rather then the other component models. A more detailed performance model can be provided considering the *SEQ* mechanism included into OASIS3 for overlapping the coupling time with the computing time of the slowest component model. The methodology will be applied in the near future also to other coupled models currently used at CMCC.



Centro Euro-Mediterraneo per i Cambiamenti Climatici

# **Bibliography**

- [1] V. Balaji, A. da Silva, and M. Suarez et al. Future directions for the earth system modeling framework. *White paper*, 2004.
- [2] I. Epicoco, S. Mocavero, and G. Aloisio. Experience on the parallelization of the oasis3 coupler. *Proc. 8th Australasian Sympo*sium on Parallel and Distributed Computing (AusPDC 2010), Brisbane, Australia, pages pp. 51–59, 2010.
- [3] I. Foster, J. Geisler, S. Tuecke, and C. Kesselman. Multimethod communication for high-performance metacomputing. *Proceedings of ACM/IEEE Supercomputing, Pittsburgh, PA*, pages 1–10, 1997.
- [4] S. Gualdi, E. Scoccimarro, A. Bellucci, P. Oddo, A. Sanna, and P.G. Fogli et al. Regional climate simulations with a global high resolution coupled model: the euromediterranean case. *Submitted.*, 2010.
- [5] S. Gualdi, E. Scoccimarro, and A. Navarra. Changes in tropical cyclone activity due to global warming: Results from a high-resolution coupled general circulation model. *Journal of Clim*, Volume 21:pp. 5204–5228, 2008.
- [6] C. Hill, C. De Luca, V. Balaji, M. Suarez, and A. da Silva. Architecture of the earth

system modeling framework. *Computing in Science and Engineering*, Volume 6, Number 1:pp. 18–28, 2004.

- [7] G. Madec. Nemo ocean engine. Technical Report Technical Report 27 ISSN No 1288-1619, Institut Pierre-Simon Laplace (IPSL), 2008.
- [8] G. Madec, P. Delecluse, M. Imbard, and C. Levy. Opa 8.1 ocean general circulation model reference manual. Technical Report Technical Report, Institut Pierre-Simon Laplace (IPSL), 1998.
- [9] E. Roeckner, R. Brokopf, M. Esch, M. Giorgetta, S. Hagemann, and L. Korn blueh et al. The atmospheric general circulation model echam5. Technical Report Technical Report 354, Max Planck Institute (MPI), 2004.
- [10] M. Tonani, N. Pinardi, S. Dobricic, I. Pujol, and C. Fratianni. A high-resolution free-surface model of the mediterranean sea. *Ocean Science*, Ocean Sci., 4:pp. 1– 14, 2008.
- [11] S. Valcke and E. Guilyardi. On a revised ocean-atmosphere physical coupling interface and about technical coupling software. *Proceedings of the ECMWF Workshop on Ocean-Atmosphere Interactions, Reading, UK*, 2008.

© Centro Euro-Mediterraneo per i Cambiamenti Climatici 2011

Visit www.cmcc.it for information on our activities and publications.

The Euro-Mediteranean Centre for Climate Change is a Ltd Company with its registered office and administration in Lecce and local units in Bologna, Venice, Capua, Sassari and Milan. The society doesn't pursue profitable ends and aims to realize and manage the Centre, its promotion, and research coordination and different scientific and applied activities in the field of climate change study.

