IL NUOVO CIMENTO Vol. 28 C, N. 2 Marzo-Aprile 2005
DOI 10.1393/ncc/i2005-10183-4

# Operational Numerical Weather Prediction systems based on Linux cluster architectures[*]

M. Pasqui[1][**], M. Baldi[1], G. Giuliani[2], B. Gozzini[1], G. Maracchi[1] and S. Montagnani[2]

[1] *IBIMET-CNR - Via Caproni 8, 50145 Firenze, Italy*
[2] *LaMMA-IBIMET - Via Madonna del Piano, 50019 Sesto Fiorentino (Firenze), Italy*

(ricevuto l' 8 Febbraio 2005; approvato il 3 Giugno 2005)

**Summary.** — The progress in weather forecast and atmospheric science has been always closely linked to the improvement of computing technology. In order to have more accurate weather forecasts and climate predictions, more powerful computing resources are needed, in addition to more complex and better-performing numerical models. To overcome such a large computing request, powerful workstations or massive parallel systems have been used. In the last few years, parallel architectures, based on the Linux operating system, have been introduced and became popular, representing real "high performance–low cost" systems. In this work the Linux cluster experience achieved at the Laboratory for Meteorology and Environmental Analysis (LaMMA-CNR-IBIMET) is described and tips and performances analysed.

PACS 92.60.-e – Meteorology.
PACS 92.60.Ry – Climatology.
PACS 92.60.Bh – General circulation.
PACS 07.05.Tp – Computer modeling and simulation.
PACS 01.30.Cc – Conference proceedings.

## 1. – Introduction

Improved computer technology has been a milestone in the last decades for the progress in weather forecast and atmospheric science. In addition, numerical weather prediction (NWP) is primarily an initial and boundary-value problem with a crucial role played by data assimilation, however, progress in climate prediction has been strongly limited by the availability and costs of computing power. In order to have more accurate weather forecasts and climate predictions, more powerful computing resources are

[*] Paper presented at CAPI 2004, 8° Workshop sul calcolo ad alte prestazioni in Italia, Milan, November 24-25, 2004.
[**] E-mail: m.pasqui@ibimet.cnr.it

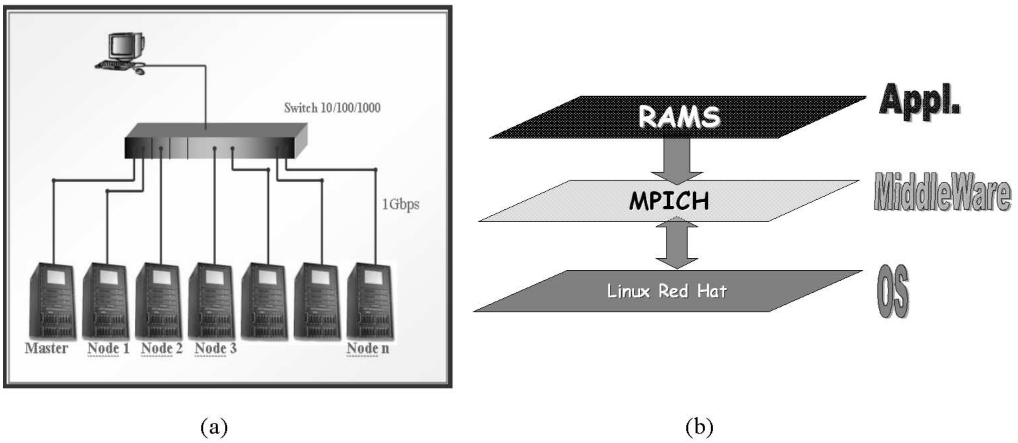


Fig. 1. – a) Beowulf Network simplified Architecture. b) Conceptual scheme of parallel computing based on a Linux Beowulf Cluster, where RAMS is the Numerical Weather Model, MPICH stands for "Message Passing Interface CHamaleont", *i.e.* the actual parallel environment.

needed, in addition to more complex and better performing numerical models. It is well known that atmospheric processes occur at a variety of time and spatial scales: from micro-physical processes in clouds to the global circulation, resulting in a necessity of developing more complex numerical models in order to better simulate physical phenomena. In particular there is a need to better describe the state of the atmosphere especially in case of extreme events such as floods and quantify climate change. Such challenging tasks should be addressed working on several aspects of the whole modelling process, *i.e.* improving physical, chemical and biological processes in the models through more detailed parametrizations, increasing the spatial and/or temporal resolution, performing ensemble simulations, performing longer simulations and therefore enhancing computing power. In this framework NWP models represent so far one of the most computational demanding activities in the computer era. To overcome such a large computing request, powerful workstations or massive parallel systems are used and therefore a high amount of money is needed both to purchase and maintain those systems. However, in the last few years, parallel computational systems, based on the Linux operating system, have been introduced and became popular, representing real "high performance–low cost" systems. Such parallel architectures represent now a real alternative to parallel systems which are much more demanding in terms of costs [13]. Different Linux cluster architectures have been proposed, but, for our activities, we adopted the so-called Beowulf architecture. At a glance, it consists of a number of PCs, called "*nodes*" connected together through a switch, within a local area network. A special node, called "*master*" is the Dynamical Name Server and provides all the network specifics: it plays the role of an actual *gate* between users and computational nodes.

A so-called "*middleware*", which is a collection of libraries, plays the formal role of communicating and transferring information among nodes (fig. 1). Therefore the number of operations needed during an atmospheric simulation is distributed among available cluster nodes reducing the total amount of overall time needed. A potentially large number of problems emerges, essentially due to the communication aspects between nodes. Network traffic, calibration and synchronicity among available nodes can reduce

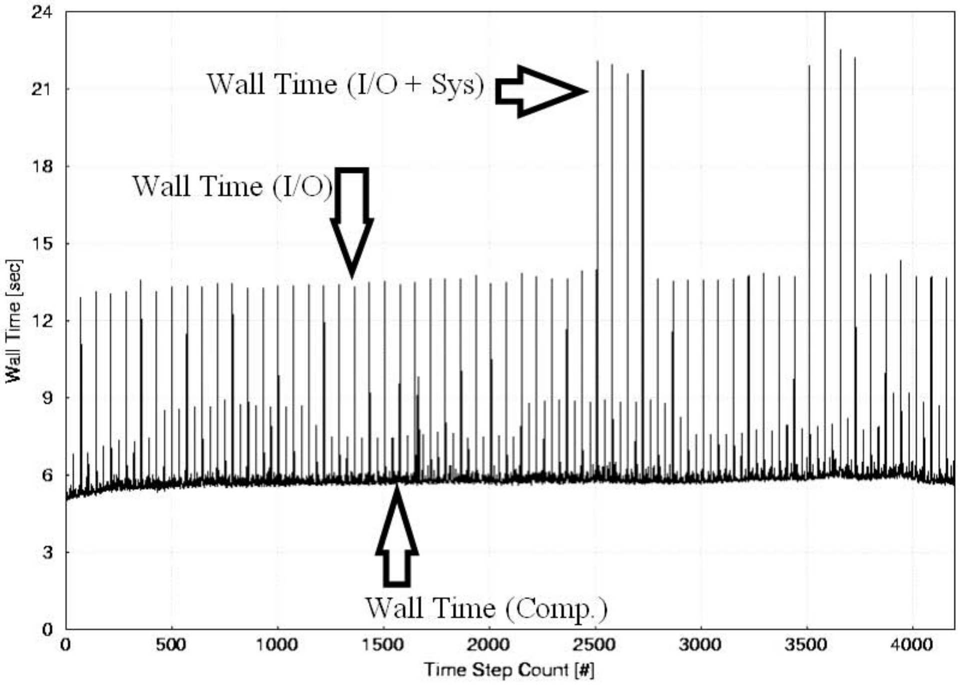


Fig. 2. – Wall clock time series for a typical simulation, the total amount of time needed during a simulation time step could be divided into three different classes: computational (Comp), Input/Output data transfer (I/O), Linux operating system activities (Sys).

parallel computational efficiency, however good and reliable performances can be achieved due both to the growth CPU computational power and to the availability of efficient and robust network drivers.

## 2. – RAMS model: parallel design and domain decomposition

The original version of the parallel RAMS (Regional Atmospheric Modelling System), first developed at CSU in 1991, used the Parallel Virtual Machine (PVM) software, developed at Oak Ridge National Laboratory, for communication among the processors of the parallel computer platform [2]. However, since then, a new *de facto* standard called MPI (Message Passing Interface MPI) and its portable implementation Message Passing Interface CHamaleont (MPICH) have been developed by a consortium of industry, government, and university computer scientists. RAMS has been designed in order to be used both under PVM or MPI. RAMS is structured in a standard master-node configuration, where the master process is a main controlling process handling model initialization and output while the nodes are the main workers, performing virtually all of the floating point computations needed for the model simulations (RAMS–User Manual).

Since the early 1990s the model has been largely improved both in the physics and sub-grid parameterisations (mainly the moist and land surface processes), and in the computational design (new numerical schemes and parallel computing). A general description of the model can be found in [12], while the current status and future perspectives of RAMS can be found in [3] and on the web site `http://www.atmet.com`. As a curiosity, [14] list a series of problems occurred when running RAMS for the first time on Linux clusters.

The basic structure of RAMS in this master-node configuration is the standard method of *domain decompositions* which is the process of spatially subdividing any computational domain into rectangular subdomains (all having approximately the same area), and each processor handles all the necessary computations for the integration time step in a different subdomain. Each node is given a set of grid points and a surrounding boundary region, which will be referred to as the *subdomain*. Again, all computation is done on the node processes. In applications requiring nested grids, the domain decomposition is completely independent for each grid. Thus, decomposition is based only on the grid size and work factors plus the number and speed of the processors, and does not depend on where the grid is placed within its parent grid, where the other grids are nested, or how the child and parent grids are decomposed. During each model timestep, the nodes exchange information at the subdomain boundaries.

The parallel version of RAMS was designed to be very compatible with the latest single-processor version. Along with the considerations for parallel efficiency (such as communication/computation overlap), attention is needed in order to schedule correctly the operations thus the boundary regions of the subdomains had access to the correct information passed from adjoining nodes.

The code was developed to handle the passing of data (messages) between the subdomains. Since different amounts of data need to be passed at different places during the execution, different sets of code were implemented to handle this task. Code to handle the bookkeeping chores was also developed. This includes the tasks of domain decomposition and dynamic load balancing.

Every simulation step is executed in a specific amount of time called "Wall Clock Time" and depends on the capacity of the computing system adopted.

This time is the total time spent by the system to perform calculations and for the Input/Output data transfer, represented for example by the transfer of boundary conditions to the nodes and/or output data file production. Such activities, even though much time-consuming, are less frequent than the pure computations which represent a minor fraction of overall simulation time. For what concern the specific RAMS configuration, all the Input/Output transfer and computational job distribution are performed by the special node called "master" which does not play any actual computation.

Typical performance values are presented in fig. 2, regarding the Wall Clock Time series. Several regimes should be recognized due to different computational activities. In this case the I/O time step duration is essentially linked to the hourly model output frequency, and, during the first period, the larger amount of time needed is due to the other Linux System activities not connected with the simulation.

## 3. – The history of the operational System used at LaMMA-Regione Toscana

The Regional Atmospheric Modeling System (RAMS), developed at Colorado State University has been run as operational mesoscale weather forecasts at the Regional Meteorological Service of Tuscany, Italy, at the Laboratory for Meteorology and Environmental Analysis (`http://www.lamma.rete.toscana.it/`) managed by Institute of Biometeorology (`http://www.ibimet.cnr.it`).

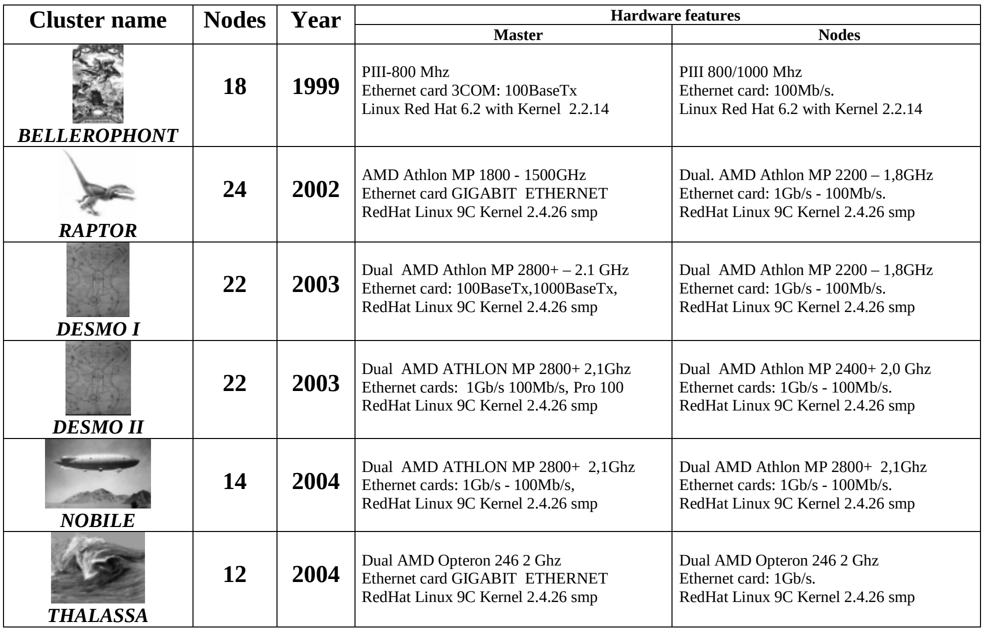
For a complete description of RAMS applications at LaMMA see [7, 5, 8, 6, 9-11, 1].

TABLE I. – *History of the operational RAMS configuration used at LaMMA.*

| Year | Resolution | | | $x$-points | $y$-points | Time-step | NNDTRAT |
|---|---|---|---|---|---|---|---|
| | Horizontal (km) | Vertical levels | Min/Max (m) | | | (s) | |
| 1999 | 40 | 24 | 100/1200 | 40 | 36 | 90 | - |
| | 8 | | | 42 | 52 | 18 | 5 |
| 2000 | 20 | 26 | 50/1000 | 100 | 97 | 60 | - |
| | 4 | | | 90 | 97 | 15 | 4 |
| 2003 | 32 | | | 184 | 120 | 90 | - |
| | 8 | 36 | 50/1100 | 158 | 178 | 22.5 | 4 |
| | 2 | | | 158 | 178 | 5.625 | 4 |
| 2004 | 20 | 35 | 50/1100 | 200 | 172 | 60 | - |
| | 6.5 | 36 | 50/1100 | 180 | 134 | 20 | - |

We started our experience using clusters driven by the need of high-performance computing power at low costs [13]. At that time the Linux kernels, version *2.2.12*, had serious problems with the MPICH libraries. Starting from kernel *2.2.18*, problems were substantially reduced and further developments were essentially focused to increase the network efficiency. Kernels evolution along with MPICH and better compilers, guaranteed the Linux clusters to be a long-life robust tool for atmospheric science. In the last five years this configuration has been widely applied in our Laboratory, for different activities, both

TABLE II. – *Characteristic of the different Linux clusters at LaMMA since 1999.*



| Cluster name | Nodes | Year | Hardware features | |
|---|---|---|---|---|
| | | | Master | Nodes |
| BELLEROPHONT | 18 | 1999 | PIII-800 Mhz<br>Ethernet card 3COM: 100BaseTx<br>Linux Red Hat 6.2 with Kernel 2.2.14 | PIII 800/1000 Mhz<br>Ethernet card: 100Mb/s.<br>Linux Red Hat 6.2 with Kernel 2.2.14 |
| RAPTOR | 24 | 2002 | AMD Athlon MP 1800 - 1500GHz<br>Ethernet card GIGABIT ETHERNET<br>RedHat Linux 9C Kernel 2.4.26 smp | Dual. AMD Athlon MP 2200 – 1,8GHz<br>Ethernet card: 1Gb/s - 100Mb/s.<br>RedHat Linux 9C Kernel 2.4.26 smp |
| DESMO I | 22 | 2003 | Dual AMD Athlon MP 2800+ – 2.1 GHz<br>Ethernet card: 100BaseTx,1000BaseTx,<br>RedHat Linux 9C Kernel 2.4.26 smp | Dual AMD Athlon MP 2200 – 1,8GHz<br>Ethernet card: 1Gb/s - 100Mb/s.<br>RedHat Linux 9C Kernel 2.4.26 smp |
| DESMO II | 22 | 2003 | Dual AMD ATHLON MP 2800+ 2,1Ghz<br>Ethernet cards: 1Gb/s 100Mb/s, Pro 100<br>RedHat Linux 9C Kernel 2.4.26 smp | Dual AMD Athlon MP 2400+ 2,0 Ghz<br>Ethernet cards: 1Gb/s - 100Mb/s.<br>RedHat Linux 9C Kernel 2.4.26 smp |
| NOBILE | 14 | 2004 | Dual AMD ATHLON MP 2800+ 2,1Ghz<br>Ethernet cards: 1Gb/s - 100Mb/s,<br>RedHat Linux 9C Kernel 2.4.26 smp | Dual AMD Athlon MP 2800+ 2,1Ghz<br>Ethernet cards: 1Gb/s - 100Mb/s.<br>RedHat Linux 9C Kernel 2.4.26 smp |
| THALASSA | 12 | 2004 | Dual AMD Opteron 246 2 Ghz<br>Ethernet card GIGABIT ETHERNET<br>RedHat Linux 9C Kernel 2.4.26 smp | Dual AMD Opteron 246 2 Ghz<br>Ethernet card: 1Gb/s.<br>RedHat Linux 9C Kernel 2.4.26 smp |

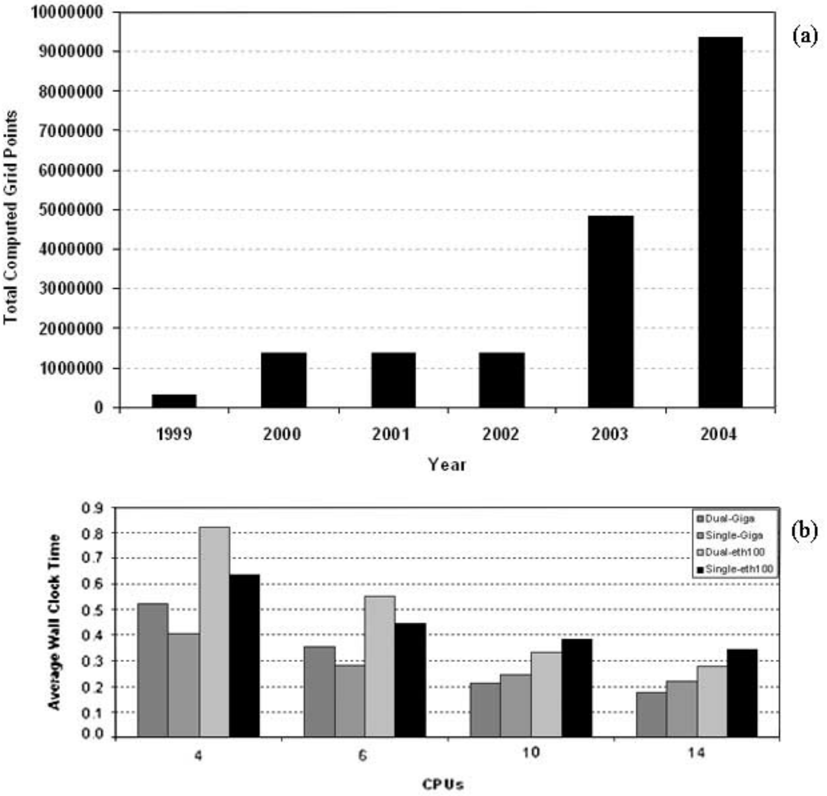


Fig. 3. – (a) Total Computed Grid Points (CGP) per simulation time step according to the operational applications using RAMS model at LaMMA in the years. (b) Average "Wall Clock Time" *vs.* number of CPU for different network architectures: Gigabit (Giga) and Ethernet10/100 (eth100) and different node architectures: Dual and Single CPUs (Dual stands for 2 CPU on the same node while Single stands for 1 CPU per node). All numbers are for the Nobile Cluster.

for operational and research tasks. At the same time, increased computing power has been requested, essentially due to the deeper knowledge of atmospheric processes, and to the development of more accurate versions of RAMS. Such increase was possible because of an analogous increase in the computing power. In table I a summary of the evolution of the operational RAMS configuration is shown, while in table II the evolution of the clusters characteristic at LaMMA is reported. A good measure of the computer power needed by RAMS is given by the total number of computed grid points (CGP). The total CGP is the total amount of grid points computed in a time step: in a single grid it is equal to $N_{\rm grid} = N_x \times N_y \times N_{\rm lev}$.

In a two-way grid configuration, since the inner-grid time step is smaller with respect to the outer-grid time step the total CGP is equal to: CGP $= N_{\rm outer\text{-}grid} + (N_{\rm inner\text{-}grid} \times$ NNDTRAT) where NNDTRAT represents the ratio between two time steps. This number is useful in order to compare different grid configurations. Using the total CGP numbers, this growing request is evident in fig. 3a. In 1999 we started with a small 2-grids configuration and a single run per day, performed on an Alpha Station 500, then, in the year 2000, we started to use BELLEROPHONT cluster, and now we are able to complete 2 runs per day for a larger grid configuration using DESMO cluster family.

We moved from Intel CPUs to dual Athlon MP and now we use the Opteron 64bit always looking for best performances. From the network point of view we started with Ethernet 10/100 Mbit/s since 2003 when the gigabit (1000 Mbit/s) replaced the previous network architecture. Nobile cluster skills are shown in fig. 3b. Using the 2003 CGP configuration as test case, the average Wall Clock Time for Nobile cluster is shown as a function of number of nodes (4 to 14 nodes), for different network architectures (Gigabit or Ethernet10/100) and for different cluster architecture (Dual or Single CPU. Dual same number of CPU as Single but on half a number of PCs, thus half network connection).

Using a similar cluster configuration, [6] re-examined the meteorological predictability of the most important floods occurred in the Arno River Basin, Italy, in the 20th century, hence providing a comprehensive study of quantitative precipitation forecast (QPF) with respect to the vertical and horizontal spatial resolution with the RAMS model. The vertical resolution is a key component in similar forecasting exercises, since more than 35 vertical levels have to be used, not equally spaced, and with a finer resolution in the lower layer of the atmosphere (atmospheric boundary layer). The horizontal spacing of the meshes of the highest-resolution grids over the target area (*e.g.* river basin) should not be more than few km (2 km or less) being this value the critical grid spacing required to ensure accurate QPF (see also [4]). In fact, an operational QPF system should be tuned on the most strict requirements. All the efforts in order to improve QPF at river basin level is essentially a matter of a suitable balance between the computational load needed and the timing strength in an operational framework. Only using the more advanced cluster architectures, *i.e.* Gigabit networks and most up-to-date CPUs, it is possible to reach the computational power needed by reasonable QPF skills at river basin scale.

## 4. – Conclusions

Nowadays Linux Clusters represent a real and powerful alternative to massive parallel systems, or supercomputers. A brief list of well-known advantages of those clusters is useful to better understand this point:

- *Affordable price/good performance ratio*: a high-performance PC cluster is cheaper of about one order of magnitude with respect to a supercomputer.
- *Scalability*: this architecture is extremely flexible and it is possible to build up clusters with the desired number of CPUs, the price being almost proportional to the number of PCs required.
- *Easy maintenance and upgrade*: a Linux cluster can be maintained and upgraded with a minimum effort and with low-cost components. It does not require special components.
- *Standard parallel platform*: libraries and software for parallel programming available for clusters are identical to those for supercomputers (*e.g.* PVM and MPI) thus computational codes can be compiled on both architectures without major modification.
- *Open Source Software*: most of the software used for clusters is free- or share-ware, accurately tested, robust and continuously upgraded.

The Linux cluster choice we made back in 1999 provides us the possibility to increase our know-how on parallel computing architectures, and thus not just use a "supercomputer blackbox", and to focus our efforts on applications. Moreover, the growing power and robustness of clusters let us not only increase the quality of our applications increasing, for example, model grid resolution, using nested grids, improving the physics in the model and developing specific sub-grid parameterisations, but also expand the range of possible applications from meteorology, to coupled atmosphere-ocean modelling, to air quality, up to regional reanalysis and climatology.

* * *

This work was supported by several different projects: *LaMMA–Tuscany Region* (DOCUP project founded by EU) and European Project CLEOPATRA: Contract EVG1-CT-2002-00070. Authors acknowledge C. J. Tremback, M. Bell, G. Brugnoni and L. Ruffo for their technical help. They also thank G. Zipoli, S. Sorbi, F. Meneguzzo for their support since the very beginning.

## REFERENCES

[1] Baldi M., Pasqui M., Cesarone F. and De Chiara G., *Heat waves in the Mediterranean Region: Analysis and model results*, in *16th Conference on Climate Variability and Change* (AMS, San Diego, CA) 2005.

[2] Bequelin A., Dongarra J., Geist G. A., Manchek R. and Sunderam V., A User's Guide to PVM - Parallel Virtual Machine, Technical Report ORNL/TM-11826, Knoxville, TN (1991).

[3] Cotton W. R., Pielke Sr. R. A., Walko R. L., Liston G. E., Tremback C., Jiang H., Mcanelly R. L., Harrington J. Y., Nicholls M. E., Carrio G. G. and Mcfadden J. P., *RAMS: Current status and future directions*, *Meteorol. Atmos. Phys.*, **82** (2001) 5.

[4] Mass C. F., Ovens D., Westrick K. and Colle B. A., *Does increasing horizontal resolution produce more skillful forecasts? Bull. Am. Meteorol. Soc.*, **83** (2002) 407.

[5] Meneguzzo F., Menduni G., Maracchi G., Zipoli G., Gozzini B., Grifoni D., Messeri G., Pasqui M., Rossi M. and Tremback C. J., *Explicit forecasting of precipitation: sensitivity of model RAMS to surface features, microphysics, convection, resolution*, in *Mediterranean Storms. 3rd Plinius Conference*, edited by Deidda R., Mugnai A. and Siccardi F., GNDCI Publ. N.2560, ISBN 88-8080-031-0, 79-84 (2001).

[6] Meneguzzo F., Pasqui M., Menduni G., Messeri G., Gozzini B., Grifoni D., Rossi M. and Maracchi G., *Sensitivity of meteorological high-resolution numerical simulations of the biggest floods occurred over the Arno river basin, Italy, in the 20th century*, *J. Hydrol.*, **288** (2004) 37.

[7] Pasqui M., Gozzini B., Grifoni D., Meneguzzo F., Messeri G., Pieri M., Rossi M. and Zipoli G., *Performances of the operational RAMS in a Mediterranean region as regards to quantitative precipitation forecasts. Sensitivity of precipitation and wind forecasts to the representation of the land cover*, in *Proceedings of "4th RAMS Users Workshop", Cook College - Rutgers University, New Jersey, USA, 2000.*

[8] Pasqui M., Grifoni D., Maracchi G., Meneguzzo F., Messeri G., Montagnani S., Redini M., Rossi M. and Todini F., *Historical severe floods prediction with model RAMS over central Italy*, in *5th RAMS Users Workshop", Santorini, Greece, 2002.*

[9] Pasqui M., Tremback C. J., Meneguzzo F., Giuliani G. and Gozzini B., *A soil moisture initialization method, based on antecedent precipitation approach, for regional atmospheric modeling system: a sensitivity study on precipitation and temperature*, in *18th Conference on Hydrology* (AMS, Seattle) 2004.

[10] Pasqui M., Pasi F. and Gozzini B., *Sahara dust impact on precipitation in severe storm events over west–central Mediterranean area*, in *Proceedings of the $14^{th}$ International Conference on Cloud and Precipitation, Bologna, Italy, 2004*.

[11] Pasqui M., Walko R. L., Migliorini S., Antonini A., Melani S. and Messeri G., *Data assimilation scheme of satellite derived heating rates for soil state initialization in a regional atmospheric mesoscale model: methodology*, in *$9^{th}$ Symposium on Integrated Observing and Assimilation Systems for the Atmosphere, Oceans, and Land Surface* (IOAS-AOLS) (AMS, San Diego, CA) 2005.

[12] Pielke R. A., Cotton W. R., Walko R. L., Tremback C. J., Lyons W. A., Grasso L. D., Nicholls M. E., Moran M. D., Wesley D. A., Lee T. J. and Copeland J. H., *A comprehensive meteorological modeling system-RamS*, *Meteorol. Atmos. Phys.*, **49** (1992) 69.

[13] Soderman D., Meneguzzo F., Gozzini B., Grifoni D., Messeri G., Rossi M., Montagnani S., Pasqui M., Orlandi A., Ortolani A., Todini E., Menduni G. and Levizzani V., *Very high resolution precipitation forecasting on low cost high performance computer systems in support of hydrological modeling*, in *Prepr. 17th Conference on Hydrology* (AMS, Long Beach) 2003.

[14] Uno I., Baldi M. and Emori S., *Application of RAMS 4.28 to transboundary air pollution studies in East Asia by Pentium Linux Cluster-Performance, Tips and Problems. RAMS Users Workshop, New Brunswick, USA, 2000.*