Chapter 45
PROMESPAR*: A High Performance Computing Implementation of the Regional Atmospheric Model PROMES

Juan E. Garrido, Enrique Arias, Diego Cazorla, Fernando Cuartero, Iván Fernández, and Clemente Gallardo

Abstract This paper describes the parallelization process of the code PROMES. The parallel code, called PROMESPAR, has been carried out under a distributed platform (cluster of PCs) and using Message Passing Interface (MPI) communication subroutines.

Keywords Regional atmospheric model · parallelization · message passing interface

45.1 Introduction

Climate change induced by human activities is one of the topics to which more attention is devoted to scientific research today. This is due, not only by the great complexity involved in the processes affecting the climate, but also to the threat involved in the serious impact that occurs on the economics and the environment in many parts of the planet. Three or 4 decades ago, it was believed that the oceans would be able to absorb the pollutants emitted by human activities; but today, maritime degradation is undeniable. Even more recently, the idea that humanity could induce a change in climate was a hypothesis that received little scientific support.

J.E. Garrido (✉), E. Arias, and D. Cazorla
Inst. Investigación en Informática. University of Castilla-La Mancha. 02071 Albacete, Spain
e-mail: Juanenrique.Garrido@uclm.es; Enrique.Arias@uclm.es; Diego.Cazorla@uclm.es

F. Cuartero
Inst. Investigación en Informática. University of Castilla-La Mancha. 02071 Albacete, Spain
e-mail: Fernando.Cuartero@uclm.es

I. Fernández and C. Gallardo
Inst. de Ciencias Ambientales. University of Castilla-La Mancha. 45071 Toledo, Spain
e-mail: Ivan.Fernandez@uclm.es; Clemente.Gallardo@uclm.es

*This work has been supported by National Project CGL2007-66440-C04.

S.-I. Ao and L. Gelman (eds.), *Electronic Engineering and Computing Technology*, Lecture Notes in Electrical Engineering 60, DOI 10.1007/978-90-481-8776-8_45,

However, there is now a broad consensus among scientists, about the evidence of anthropogenic climate change and the need for better knowledge about likely developments in the following decades.

To simulate the climate, we use numerical models reproducing the main processes occurring in the five components of the climate system: Atmosphere, hydrosphere, geosphere, and biosphere, and the exchange of mass and energy between them. The results obtained by the models are evaluated and compared with the observed features of the climate in recent decades. Once it is found the quality level of the climate model is correct, we apply it to simulate potential changes in the climate, considering various scenarios of anthropogenic emissions of greenhouse gases and aerosols. Since this information, we can deduce the potential impact of climate change produced in such a hypothesis.

The history of weather forecasting is intimately associated to development of high performance and parallel computing [9].

In fact, thanks to the parallelization of weather prediction models, it is provided to scientists the ability to deal with longer simulations, to increase the spatial resolution, etc.

Throughout the last decade, several parallel approaches have been developed. Among them [10], we remark the based on vectorial multiprocessors such as CCM2 and its scalable variants [11, 17, 18], massively parallel computers (adaptation from the spectral model of the National Meteorology Centre) [25], distributed memory multiprocessors [5] (integrated prediction system) and passing messages [28], MM5 model [16] and another MM5 versions [23, 24], and the application of Grid technology (IrisGRID [3], CrossGrid [2], Climateprediction.net [1] program).

The paper is organized as follows. Section 45.2 introduces the regional atmospheric model PROMES, and in Section 45.3 the parallelization of PROMES is presented. The experimental results are outlined in Section 45.4. Finally, the conclusions and future work are commented in Section 45.5.

45.2 The Regional Atmospheric Model PROMES

PROMES is a regional atmospheric numerical model used in meteorological and climate research and also for weather forecasting. PROMES has been developed by MOMAC (MOdelizacin para el Medio Ambiente y el Clima) research group at the University of Castilla-La Mancha (UCLM) and the Complutense University of Madrid (UCM), and it was originally described in [7]. It is a hydrostatic limited-area model with sigma levels as vertical coordinates [26] and Lambert conic projection for the horizontal coordinates [4]. The spatial arrangement of variables follows the so called Arakawa-C grid [22]. The studied region is divided on a set of vertical columns and each column in turn is divided into several levels. Thus, the state of the atmosphere is defined at each time in a finite number of grid-boxes arranged in a mesh (Fig. 45.1)

PROMES model uses a split–explicit time integration scheme, based on [13]. The different terms of the primitive equations that govern atmospheric dynamics are

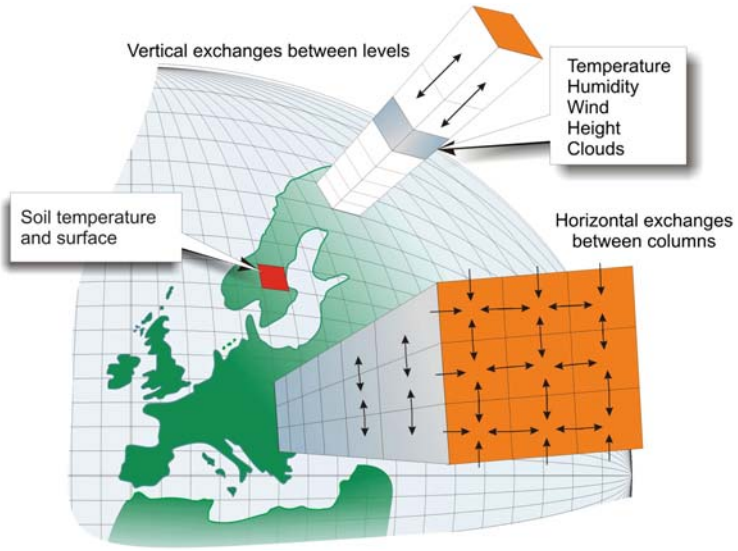


Fig. 45.1 Grid of calculus

integrated with time steps depending on their typical time-scale. Numerical schemes based on finite differences are used for solving those terms. A forward–backward scheme is applied to the gravity waves terms, a cubic–spline upstream method is used to solve the advection terms and a fourth-order explicit scheme is employed for horizontal diffusion. The needed lateral boundary values are updated from analysis or Global Circulation Model data. The vertical interpolation of the large scale variables to model levels follows the method described in [14]. The model variables are relaxed to the external information in a contour band following [8].

The physical parameterizations included in the version of PROMES used in this study were those described below. The absorption and scattering of shortwave radiation is based on [4], and longwave radiation processes are parameterized according to [15, 27]. Shortwave heating and infrared cooling is calculated according to [27]. Turbulent vertical exchange of the prognostic variables in the planetary boundary layer (PBL) is modelled as proposed by [29] by using four regimes: stable, mechanical turbulence, forced convection and free convection. For the first three cases a local K-theory parameterization is applied [6]. In the case of free convection, a non-local scheme is used. Outside the PBL the vertical diffusion is also computed using K-theory. PROMES takes into account the exchanges of energy and water between soil, vegetation and atmosphere by using the landsurface scheme called SECHIBA [12]. SECHIBA gets the atmospheric forcings from PROMES and calculates sensible and latent heat fluxes. At each grid-box, bare soil and up to seven vegetation types are permitted to be present in different proportions. The soil water content is calculated in two layers, meanwhile soil temperature is computed in seven layers [20]. The resolved-scale cloud formation and its associated precipitation processes are modelled according to [19]. Sub-grid scale convective clouds and

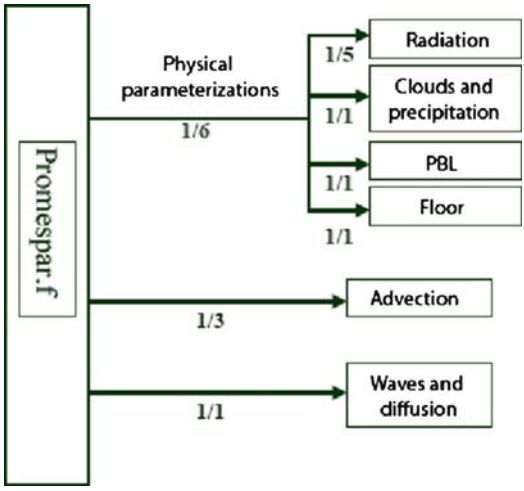


Fig. 45.2 General squeme of PROMES code

their precipitation are parameterized using the [21] method. As a summary of this PROMES description, a scheme of the model code is shown in Fig. 45.2.

As can be seen above, PROMES solved a set of equations and several complex parameterizations that involved a huge number of calculations. Therefore, whether an accurate solution is needed, parallel platforms to solve the problem are essential in order to obtain the results in a reasonable time.

45.3 PROMESPAR: A Distributed Memory Implementation of PROMES

As it was previously commented, in order to obtain a very accurate solution in a reasonable time, it is necessary the use of parallel platforms. In this paper, a distributed memory implementation of PROMES code, called PROMESPAR, is presented.

The parallelization of PROMES consists on dividing the domain on a set of subdomains getting out the work to carried out into the different processors (see Fig. 45.3). Once the domain has been divided the processors just exchange the frontier information.

In order to obtain an equally load balancing, a constrain is applied to the size of the subdomain and the number of processor to be used. This constrain is given by Eq. (45.2)

$$ProcXBlockSize = \left(\frac{OrXmatSize}{XsizeProc}\right) \pm XBorderSize \tag{45.1}$$

$$ProcYBlockSize = \left(\frac{OrYmatSize}{YsizeProc}\right) \pm YBorderSize \tag{45.2}$$

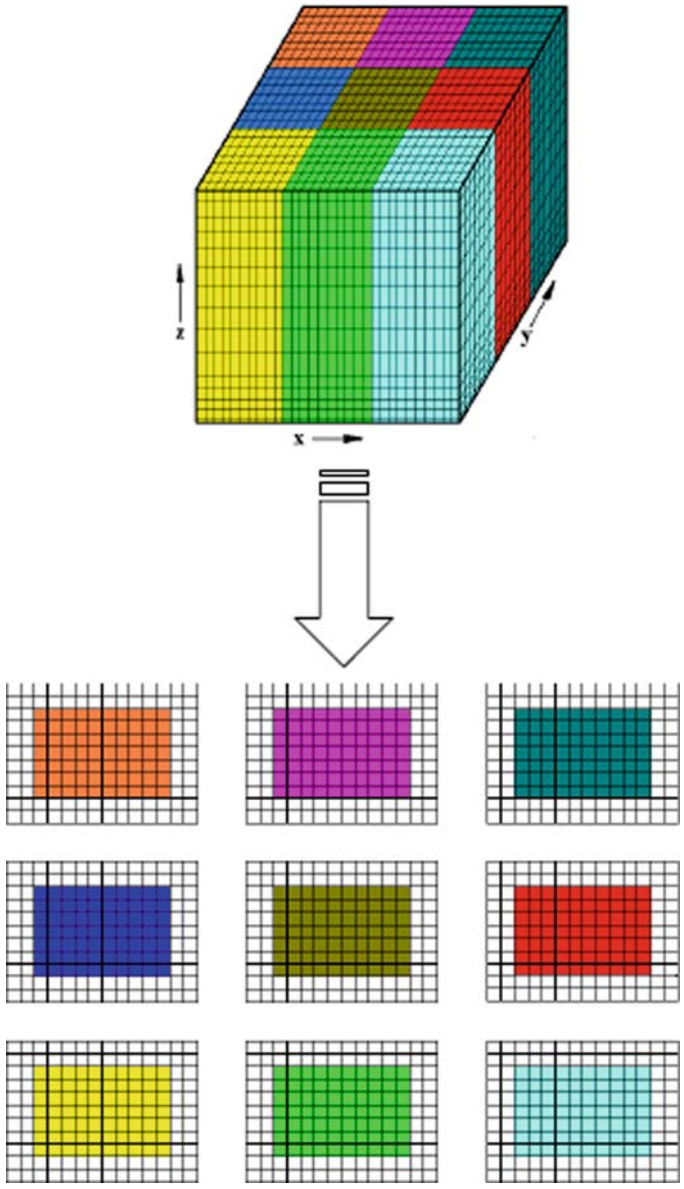


Fig. 45.3 Squeme of splitting the domain into subdomains

where *ProcXBlockSize* and *ProcYBloclSize* mean the size of blocks for each processor at X or y coordinate, respectively, which is computed from the original dimension of the matrix (*OrXmatSize* and *OrYmatSize*) and the number of processors by each coordinate (*XsizeProc* and *YsizeProc*), and taking into account the boundary conditions (*XBorderSize* and *YBorderSize*).

However, processor 0 has additional tasks due to the fact that it acts as master reading initial conditions, boundary values for the domain, etc from files.

In any case, the good load balancing could be affected mainly by two factors:

- **Static imbalance**. Those processors whose subdomains contain maritima zones have less computational load. This circumstance is due to the fact that the computations needed for solving the forecasting model are simplest in this kind of cells (some physical phenomena as the effect of orography, heat exchange with the masses of plants, etc are not taken into account).
- **Dynamic imbalance**. This kind of imbalance is devoted by the initial conditions. For instance, the effect of solar radiation could vary if a cloudy day or a sunny day is considered. These effects are unpredictable. However, other effects as the solar radiation during the night are predictable.

In the implementation of PROMESPAR the following libraries have been considered:

- MPI: Messing Passing Interface use for communications purpose. This library supports the communication between the different processors of the distributed memory platform.
- NETCD: NetCDF (network Common Data Form) is a set of software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.
- IOPSL: Library for input/output operations with meteorological data.
- Other physical libraries: computation of solar radiation, heat exchange ground-atmosphere, etc.

Figure 45.4 represents the workflow of the parallel implementation of PROMES, PROMESPAR.

The workflow in Fig. 45.4 is followed by each processor, and the barriers on Fig. 45.4 mean communication or synchronization takes amount the different processors.

45.4 Experimental Results

The experimental results have been obtained taken into account 24 h of simulation. The distributed memory implementation has been run into a cluster of PCs with 16 Intel processors at 1.8 GHz, each one with 512 MB of main memory and interconnected by a Myrinet Network using NFS file system.

The performance obtained in the parallel implementations are evaluated in terms of

- Execution time: Time spent in order to solve the problem.
- Speed-up: The ratio of the time taken to solve a problem on a processor to the time required to solve the same problem on a parallel computer with p identical processors.

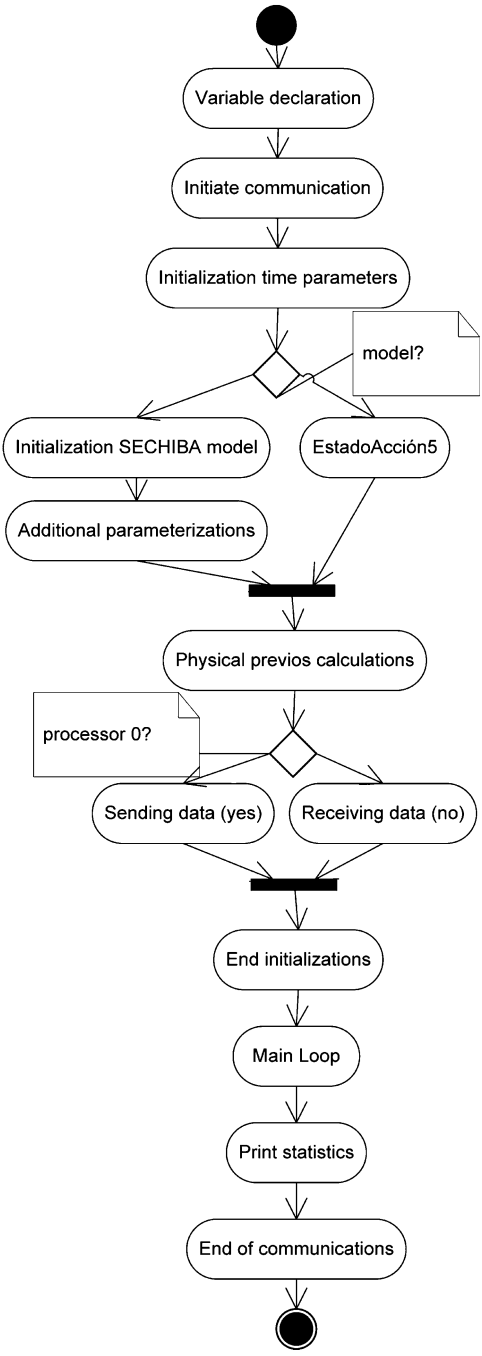


Fig. 45.4 Workflow of PROMESPAR

- Efficiency: A measure of the fraction of time for which a processor is usefully employed; it is defined as the ratio of the speed-up to the number of processors.

Most time consuming has been spent at main loop where are contained the most computational cost operations. In particular, apart from send and receive operations for communication purpose, physical operations are invoked. These operations are shown at Figs. 45.2 and 45.4.

The experimental results considered in this section take into account a 24 h simulation, which is equivalent to carry out 2881 iterations of main loop.

Figures 45.5–45.7 show the results of the previous experiment (24 h simulation) in terms of execution time, speed-up and efficiency.

From the experimental results, the main conclusion is that the best results, in terms of execution time has been obtained considering eight processors. However, in terms of speed-up and efficiency best results are obtained for two processors. This is a normal circumstance due to the influence of the communications. However, for this particular applications the main goal is to reduce the execution time.

As it was previously commented, the most time consuming of PROMESPAR code is spend on main loop. Figure 45.8 show a detailed study of the time spend on main loop. It is possible to observe that *fisicapal*, *Coriolis* and *Diffusion* functions spent the most quantity of time, and obviously the parallelization approach allows to reduce this execution time, overall from one to two processors. Anyway, the reduction of execution time results quite good.

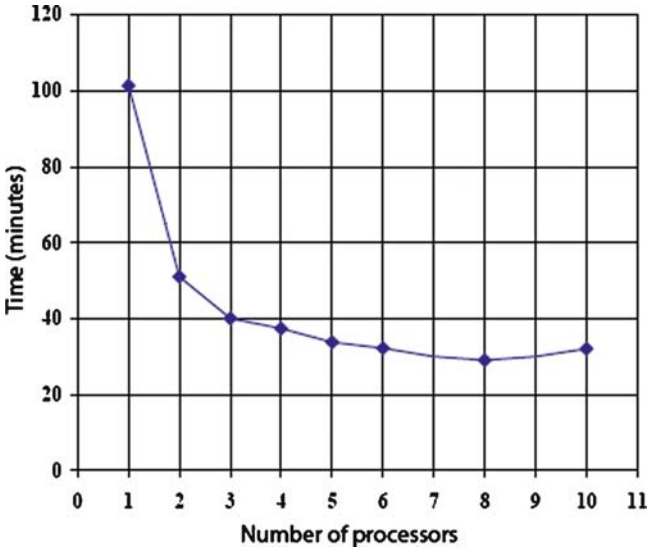


Fig. 45.5 Execution time of PROMESPAR

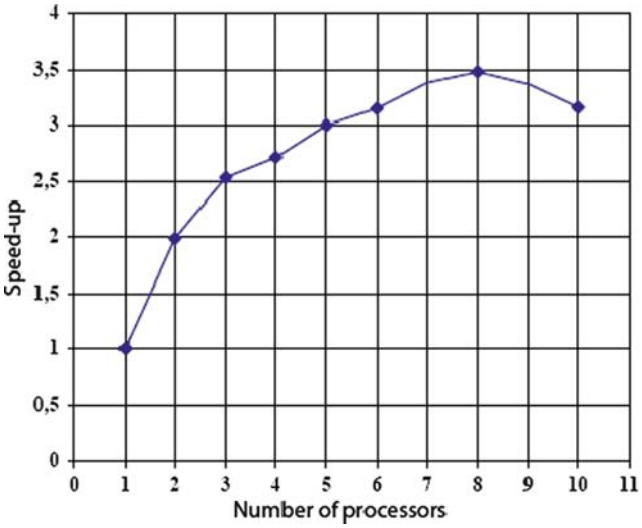


Fig. 45.6 Speed-up of PROMESPAR

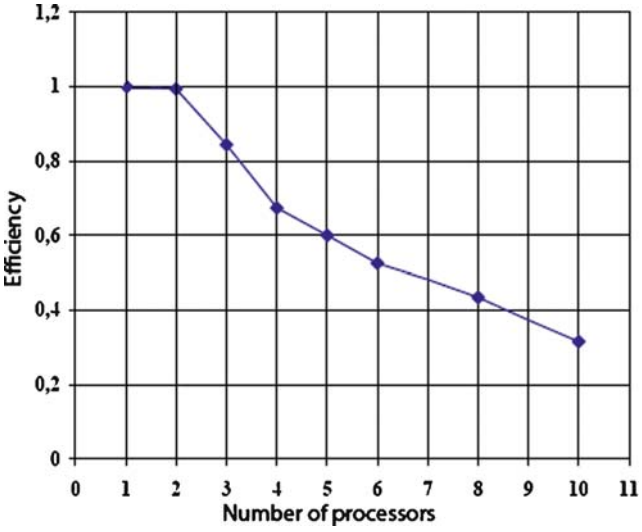
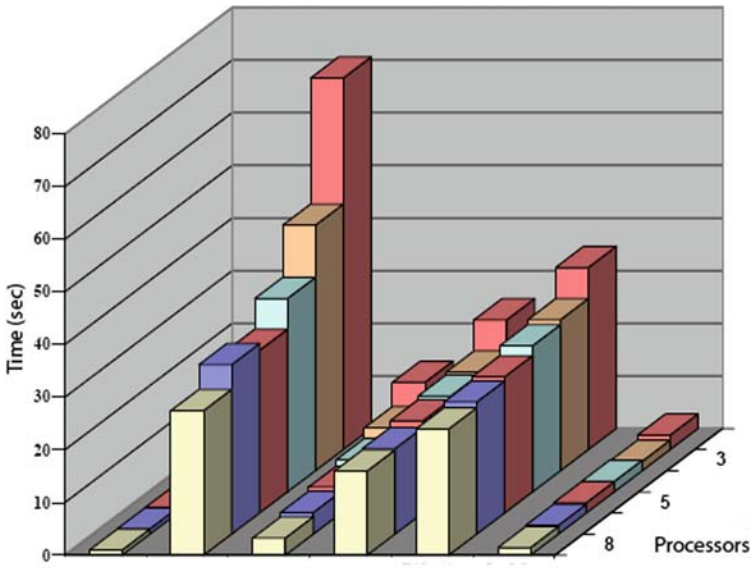


Fig. 45.7 Efficiency of PROMESPAR

45.5 Conclusion

PROMES is a mesoscale regional atmospheric model developed, among others, by some of the authors of this paper. However, due to the high time consuming by PROMES code and the necessity of having more accurate results, both circumstances justify the used of parallelism. In this paper, a distributed memory



	Antes_f	Fisicapal	Advecc.	Coriolis	Difusión	Int&Sta
8	1	27	3	16	24	1,44216
6	1	32,17	4,11579	16,2341	25	1,66534
5	1,04169	31	4,91797	17,5079	25,8628	1,9383
4	1,23075	37	6,15263	18,2263	27,7531	1,98593
3	1	46,6334	8,16989	18,7399	29	1,79524
2	1,96858	70,5729	12,783	24,6558	34,614	2,75723

Measuring points

Fig. 45.8 Execution time of the main loop of PROMESPAR for an hour simulation

implementation of the regional atmospheric model PROMES has been carried out. This parallel implementation is called PROMESPAR.

The experimental results show a dramatically execution time reduction by means of the use of a parallel platform considering the same configuration that the original PROMES code. These results leads to think that either longer or more accurate simulations could be carried out spending the same time, or more complex models could be considered. In fact, the authors are extending PROMES code in order to be able of making climate change studies. Climate change studies consider 100 years simulations spending, obviously, lot of time and then if the researchers want to provide conclusions from these studies the use of parallelism becomes essential.

References

1. Climateprediction.net. http://www.climateprediction.net/.
2. Crossgrid. http://www.crossgrid.org/.
3. Irisgrid. http://www.irisgrid.es.

4. Anthes, R.A., Hsie, E.-Y., Kuo, Y.-H.: Description of the Penn State/NCAR Mesoscale Model Version 4 (MM4). NCAR Technical Note 282. NCAR, Boulder, CO 80307 (1987)
5. Barros, S.R.M., Dent, D., Isaksen, L., Robinson, G., Mozdzynsky, G., Wollenweber, F.: The ifs model: a parallel production weather code. Parallel Comput (21), 1621–1638 (1995)
6. Blackadar, A.K.: Modeling the nocturnal boundary layer. In: American Meteorological Society, Boston (ed.) Proceedings of the Third Symposium on Atmospheric Turbulence, Diffusion and Air Quality, pp. 46–49 (1976)
7. Castro, M., Fernández, C., Gaertner., M.A.: Descrition of a meso-scale atmospheric numerical model. Mathematics, climate and environment, pp. 230–253 (1993)
8. Davies, H.C.: A lateral boundary formulation for multilevel prediction models. Quart. J. Roy. Meteor. Soc. 102, 405–418 (1976)
9. Drake, J., Foster, I.: Introduction to the special issue on parallel computing in climate and weather modelling. Parallel Comput. (21), 1539–1544 (1995a)
10. Drake, J., Foster, I.: Special issue on parallel computing in climate and weather modelling. Parallel Computing (21) (1995b)
11. Drake, J., Foster, I., Michalakes, J., Toonen, B., Worley, P.: Design and performance of a scalable parallel community climate model. Parallel Comput. (21), 1571–1591 (1995)
12. Ducoudr, N., Laval, K., Perrier, A.: SECHIBA, a new set of parameterizations of the hydrologic exchanges at the landatmosphere interface within the LMD atmospheric general circulation model. J. Climate **6**, 248–273 (1993)
13. Gadd, J.J.: A split explicit integration scheme for numerical weather prediction. Q. J. R. Meteorol. Soc. **104**, 569–582 (1978)
14. Gaertner, M.A., Castro, M.: A new method for vertical interpolation of the mass field. Mon. Wea. Rev. **124**, 1596–1603 (1996)
15. Garand, L.: Some improvements and complements to the infrared emissivity algorithm including a parameterization of the absorption in the continuum region. J. Atmos. Sci. **40**, 230–244 (1983)
16. Grell, D.O., Dudhia, J., Stuffer, D.R.: A description of the fifth-generation penn state/ncar mesoscale model (mm5). Ncar/tn-398+str, National Center for Atmosphere Research, Boulder, CO (1994)
17. Hack, James J., Rosinski, James M., Williamson, David L., Boville, Byron A., Truesdale, John E.: Computational design of the ncar community climate model. Parallel Comput. (21), 1545–1569 (1995)
18. Hammond, S., Loft, Richard D., Dennis, John M., Sato, Richard K.: Implementation and performance issues of a massively parallel atmospheric model. Parallel Comput. (21), 1593–1619 (1995)
19. Hsie, E.Y, Anthes, R.A., Keyser, D.: Numerical simulation of frontogenesis in a moist atmosphere. J. Atmos. Sci **41**, 2581–2594 (1984)
20. Jacobsen, I., Heise, E.: A new economic method for the computation of the surface temperature in numerical models. Beitr. Phys. Atmosph. **55**, 128–141 (1982)
21. Kain, J.S., Fritsch, J.M.: Convective parameterization for mesoscale models: The KainFritsch scheme. The Representation of Cumulus Convection in Numerical Models. Meteor. Monogr. No. 46, Amer. Meteor. Soc. 165–170 (1993)
22. Mesinger, F., Arakawa, A.: Numerical methods used in atmospheric models. GARP Publication Series, 17:1–64 (1976)
23. Michalakes, J.: MM90: A scalable parallel implementation of the Penn State/NCAR Mesoscale Model (MM5). Parallel Computing, 23:14, 2173–2186 (1997)
24. Michalakes, J., Canfield, T., Nanhyndiah, R., Hammond, S., Grell, G.: Parallel implementation, validation and performance of mm5. In: Proceedings of the Sixth ECMWF Workshop on the Use of Parallel Processors in Meteorology, World Scientific, River Edge, NJ pp. 266–276 (1995)
25. Sela, J.G.: Weather forecasting on parallel architectures. Parallel Comput (21), 1639–1654 (1995)
26. Shuman, F.G., Hovermale, J.B.: An operational sixlayer primitive equation model. J. Appl. Meteor, **7**, 525–547 (1968)

27. Stephens, G.L.: Radiation profiles in extended water clouds ii: parameterizaton schemes. J. Atmos. Sci. **35**, 2123–2132 (1978)
28. Wehner, M.F., Mirin, A.A., Eltgroth, P.G., Dannevik, W.P., Mechoso, C.R., Farrara, J.D., Spahr, J.A.: Performance of a distributed memory finite difference atmospheric general circulation model. Parallel Comput. (21), 1655–1675 (1995)
29. Zhang, D.L., Anthes, R.A.: A high resolution model of the planetary boundary layer sensitivity test and comparisons with sesame79 data. J. Appl. Meteor. **21**, 1594–1629 (1982)