Flexible Two-Level Parallel Implementations of a Large Air Pollution Model

Tzvetan Ostromsky[1] and Zahari Zlatev[2]

[1] Central Lab. for Parallel Processing, Bulgarian Academy of Sciences, Acad. G.Bonchev str., bl. 25-A, 1113 Sofia, Bulgaria.
ceco@parallel.bas.bg

[2] Nat. Environmental Research Institute, Frederiksborgvej 399 P. O. Box 358, DK-4000 Roskilde, Denmark.
zz@dmu.dk

Abstract. Large scale air pollution models are powerful tools, designed to meet the increasing demand in different environmental studies. The atmosphere is the most dynamic component of the environment, where the pollutants can quickly be moved in a very long distance. Therefore the advanced modeling must be done in a large computational domain. Moreover, all relevant physical, chemical and photochemical processes must be taken into account. The speed of these processes vary in a wide range. This fact implies that a small time step must be used in order to achieve both numerical stability and sufficient accuracy of the results. Thus the numerical treatment of such an air pollution model becomes in many cases a huge computational problem, a challenging task for the most powerful up-to-date supercomputers.
The Danish Eulerian Model (DEM) is used in this work. The paper focuses on the efficient parallel implementation of DEM on powerful parallel supercomputers. We present a variety of performance and scalability results, obtained on different parallel machines by using standard parallelization tools (MPI for distributed-memory parallelism and OpenMP for shared-memory parallelism). It is shown by experiments that MPI and OpenMP can both be used on separate levels of parallelism to get the best use of the clustered parallel machines. Application of the results in the environmental studies, related to high ozone concentrations in the air, is illustrated by some plots in the last section.

1 The Danish Eulerian Model

The DEM is represented mathematically by the following system of partial differential equations (PDE):

$$\frac{\partial c_s}{\partial t} = -\frac{\partial (u c_s)}{\partial x} - \frac{\partial (v c_s)}{\partial y} - \frac{\partial (w c_s)}{\partial z}$$

$$+\frac{\partial}{\partial x}\left(K_x \frac{\partial c_s}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial c_s}{\partial y}\right) + \frac{\partial}{\partial z}\left(K_z \frac{\partial c_s}{\partial z}\right)$$

$$+E_s - (\kappa_{1s} + \kappa_{2s}) c_s + Q_s(c_1, c_2, \ldots, c_q) \; ; \quad s = 1, \ldots, q \qquad (1)$$

I. Dimov et al. (Eds.): NMA 2002, LNCS 2542, pp. 545–554, 2003.

where c_s are the concentrations of the chemical species involved in the model, u, v and w are the wind components, K_x, K_y and K_z are diffusion coefficients, E_s are the emissions, κ_{1s} and κ_{2s} are the coefficients for dry and wet deposition, and $Q_s(c_1, c_2, \ldots, c_q)$ are expressions that describe the chemical reactions under consideration.

The PDE system (1) is too complex for direct numerical treatment. A splitting procedure, based on ideas in [5,6], is used to split (1) into five sub-models, representing the main physical and chemical processes ($s = 1, 2, \ldots, q$): the horizontal advection (2), the horizontal diffusion (3), the chemistry and the emission (4), the deposition (5) and the vertical exchange (6):

$$\frac{\partial c_s^{(1)}}{\partial t} = -\frac{\partial (u c_s^{(1)})}{\partial x} - \frac{\partial (v c_s^{(1)})}{\partial y} \tag{2}$$

$$\frac{\partial c_s^{(2)}}{\partial t} = \frac{\partial}{\partial x}\left(K_x \frac{\partial c_s^{(2)}}{\partial x}\right) + \frac{\partial}{\partial y}\left(K_y \frac{\partial c_s^{(2)}}{\partial y}\right) \tag{3}$$

$$\frac{dc_s^{(3)}}{dt} = E_s + Q_s(c_1^{(3)}, c_2^{(3)}, \ldots, c_q^{(3)}) \tag{4}$$

$$\frac{dc_s^{(4)}}{dt} = -(\kappa_{1s} + \kappa_{2s}) c_s^{(4)} \tag{5}$$

$$\frac{\partial c_s^{(5)}}{\partial t} = -\frac{\partial (w c_s^{(5)})}{\partial z} + \frac{\partial}{\partial z}\left(K_z \frac{\partial c_s^{(5)}}{\partial z}\right) \tag{6}$$

The discretization of the spatial derivatives in the right-hand-sides of the sub-models (2) – (6) results in five systems of ordinary differential equations:

$$\frac{dg^{(i)}}{dt} = f^{(i)}(t, g^{(i)}), \qquad \begin{matrix} g^{(i)} \in R^{N_x \times N_y \times N_z \times q} \\ f^{(i)} \in R^{N_x \times N_y \times N_z \times q} \end{matrix} ; \qquad i = 1, \ldots 5$$

where N_x, N_y and N_z are the numbers of grid-points along the coordinate axes and q is the number of chemical species. The functions $f^{(i)}$, $i = 1, \ldots 5$, depend on the particular discretization methods used in the sub-models, while the functions $g^{(i)}$, $i = 1, \ldots 5$, represent approximations of the concentrations at the grid-points of the computational domain. More details about the numerical methods, used in the submodels, can be found in [1,3,4,10].

2 Space Domain Discretization

The space domain of the model is part of the Northern hemisphere (4800×4800 km.) that covers Europe, most of the Mediterian and neighboring parts of Asia and the Atlantic Ocean. Grids of different size (and, respectively, with different step) are used in the discretization of that domain. The resulting versions of the

Table 1. Versions of DEM with existing parallel codes. The 3-D version for 288×288 grid is not yet fully operational.

Grid size	Grid step	Number of grid cells	2-D version	3-D version
(32×32)	150 km	1024	Yes	Yes
(96×96)	50 km	9216	Yes	Yes
(288×288)	16.7 km	82944	Yes	Yes*
(480×480)	10 km	230400	Yes	No

model are given in Table 1. Some of these versions are discussed in more detail in [1,4,7,8,10,11]. In the 3-D versions the domain in vertical direction is split into several layers of different (but constant) thickness. The lower layers are thinner, while the upper are thicker. Ten layers are used in the present 3-D versions. The number of cells in the corresponding 2-D version is given in the third column of Table 1. This number should be multiplied by 10 (the number of layers) in order to get the number of cells for the corresponding 3-D version. The 2-D and 3-D versions on the medium-resolution (96×96) grid are mainly used in our experiments (Table 2), as well as the newly developed 3-D version for 288×288 grid (Table 3).

3 Parallelization Techniques

Most of the existing parallel supercomputers can be classified in one of the following three groups:

(i) Shared memory computers, like SGI Origin, SUN Enterprice, etc.;

(ii) Distributed memory computers, like CRAY T3E, IBM SP2;

(iii) Clustered (hybrid) computers – a distributed memory cluster of *nodes*, each node being a separate shared-memory parallel computer (for example, IBM SP3, Beowulf clusters, etc.).

Results on computers from all the three groups are presented in this paper.

One of the main goals in this work was to exploit efficiently the full capacity of the clustered computers (group (iii)). Nevertheless, our strategic principle in code development has been **high portability**. That is why only standard parallelization tools (MPI and OpenMP) are used in the codes under consideration. It should be emphasized that neither extensions of the standard of the above libraries, nor any special properties of the particular computers, that have been used in the experiments, are applied in these codes. With minor changes in the driver routine they can be run on any parallel computing system that supports MPI (either clustered, with shared or distributed memory).

3.1 Shared Memory Parallelization via OpenMP

It is relatively easy to obtain an efficient shared memory parallelization by exploiting the data-independent potentially parallel tasks in each submodel. This

can be achieved by using only standard OpenMP [9] directives. However, it is not always possible to achieve good data locality in the large shared arrays. The small tasks are grouped in chunks where appropriate for more efficient cache utilization. A parameter CHUNKSIZE is provided in the code, which should be tuned with respect to the cache size of the target machine. The main submodels are parallelized as follows:

- **Horizontal advection and diffusion.** The horizontal advection and diffusion submodels are treated together. In fact several independent advection-diffusion subproblems with Dirichlet boundary conditions arise on each time step after the splitting procedure, one for every chemical compound on every layer (in the 3-D version). It means, there are enough parallel tasks on this stage ($N_z \times q$) and these tasks are big enough.
- **Chemistry and deposition.** The calculations of these two processes can be carried out independently for each grid-point. There are many parallel tasks ($N_x \times N_y \times N_z$), but each task is small. For the purpose of efficient cache utilization the tasks should be grouped in chunks. The parameter CHUNKSIZE is used to set their size.
- **Vertical exchange.** This stage is present only in the 3-D versions. The vertical exchange submodel splits into independent relatively simple advection-diffusion subproblems (along each vertical grid-line). The number of parallel tasks is large ($N_x \times N_y$) and they are not very big. Like on the chemistry-deposition stage, the tasks can be grouped in chunks to improve the cache utilization (see Fig. 1).

3.2 Distributed Memory Parallelization via MPI

The MPI (Message Passing Interface, [2]) was initially developed as a standard communication library for distributed memory computers. Later, proving to be efficient, portable and easy to use, it became one of the most popular parallelization tools for application programming. Now it can be used on much wider class of parallel systems, including shared-memory computers and clustered systems (each node of the cluster being a separate shared-memory computer with fixed number of processors). Thus it provides high level of portability to the codes.

Our MPI parallelization is based on the space domain partitioning. The space domain is divided into several sub-domains (the number of the sub-domains being equal to the number of MPI tasks). Each MPI task works on its own sub-domain. On each time step there is no data dependency between the MPI tasks on both the chemistry and the vertical exchange stages. This is not so with the advection-diffusion stage. Spatial grid partitioning between the MPI tasks requires overlapping of the inner boundaries and exchange of certain boundary values on the neighboring subgrids for proper treatment of the boundary conditions. This leads to two main consequences:

(i) certain computational overhead and load imbalance, leading to lower speed-up of the advection-diffusion stage in comparison with the chemistry and the vertical transport (as can be seen on Fig. 1).

(ii) communication necessity for exchanging boundary values on each time step (done in a separate **communication** stage).

In addition, two extra procedures are used for scattering the input data and gathering the results in the beginning and in the end of the run respectively.

- **Pre-processing.** In the beginning of the job the input data, stored in several large files (containing meteorological and emission data sets for the whole domain) is distributed in separate files for each of the sub-domains (to be processed in parallel). In this way, not only each MPI process will be working on its own sub-domain, but will also be accessing only the part of meteorological and emission data relevant to its sub-domain.
- **Post-processing.** During the run each MPI process prepares its own output data files. At the end of the run all the output data of a same kind from all MPI processes are collected and prepared for future use by one of the MPI processes during the post-processing procedure.

The time for pre-processing and post-processing is, in fact, overhead, introduced by the MPI partitioning strategy. Moreover, this overhead is growing up with increasing the number of MPI tasks and little can be done for its parallel processing. Thus the relative weight of these two stages grows up with increasing the number of MPI tasks, which eventually affects the total speed-up and efficiency of the MPI code.

3.3 Mixed Shared-Distributed Memory Parallelization

By mixing both parallelization techniques, described above (i.e. MPI parallelization by using domain decomposition on the top level together with OpenMP-parallelization for building second level, somewhat finer-grain parallelism), a mixed shared-distributed memory parallel version of the model is created. This code should be perfect for huge clustered supercomputers with large number of nodes. By giving each node one MPI task and an OpenMP thread to each processor of the node, optimal performance can be achieved. We should mention, however, that this code is build as an extension of the pure MPI code (by adding OpenMP directives) without destroying its structure. Although it has additional functionality on machines with OpenMP, on machines without OpenMP it is virtually the same MPI code (the OpenMP directives are treated as comments). Thus it looses neither portability, nor efficiency with respect to the the pure MPI code. Some results of experiments with such codes for grid size 96×96 and 288×288 are given in the next section, together with the results of the pure MPI codes.

4 Numerical Results

In Table 2 results from experiments with the 2-D and 3-D versions of DEM (96×96 grid) on three supercomputers of different classes are presented.

Table 2. Total computing time (measured in seconds), speed-up and efficiency of the 2-D (upper table) and 3-D (lower table) versions of DEM. Three supercomputers of different type, all in CINECA – Bologna, are used in the experiments: (i) IBM SP Power 3 (clustered machine, 8 nodes, 16 PE/node, 375 MHz); (ii) SGI Origin 3800 (shared-memory machine, 32 PE R14000/500 MHz); (iii) CRAY T3E - 1200E (distributed-memory machine, 256 PE). There are no experiments on 1 and 2 processors on the CRAY T3E for the 3-D version (the places marked with *) due to the time limit of 6 hours per job.

2-D version of DEM on the CINECA supercomputers
$(96 \times 96 \times 1)$ grid, CHUNKSIZE=48

Proc.	IBM sp3			SGI Origin 3800			CRAY T3E		
	Time [sec]	Speed -up	**E** %	Time [sec]	Speed -up	**E** %	Time [sec]	Speed -up	**E** %
1	2148			2261			6042		
2	1047	2.1	103	1080	2.0	100	3050	2.0	99
4	528	4.1	102	537	4.2	105	1520	4.0	99
6	356	6.0	101	349	6.5	108	1033	5.8	97
8	273	7.9	98	268	8.4	105	780	7.7	97
12	192	11.2	93	183	12.4	103	530	11.4	95
16	157	13.7	86	142	15.9	99	425	14.2	89
24	124	17.3	72	117	19.3	81	315	19.2	80
32	113	19.9	62	85	26.6	83	257	23.5	73
48	95	22.6	47				198	30.5	64

3-D version of DEM on the CINECA supercomputers
$(96 \times 96 \times 10)$ grid, CHUNKSIZE=48

Proc.	IBM sp3			SGI Origin 3800			CRAY T3E		
	Time [sec]	Speed -up	**E** %	Time [sec]	Speed -up	**E** %	Time [sec]	Speed -up	**E** %
1	21516			21653			*	*	*
2	10147	2.1	106	10487	2.1	103	*	*	*
3	6863	3.1	105	7057	3.1	102	19532	3.0	100
4	5110	4.2	105	5412	4.0	100	14653	4.0	100
6	3516	6.1	102	3480	6.2	104	9813	6.0	100
8	2586	8.3	104	2658	8.1	102	7258	8.1	101
12	1759	12.2	102	1797	12.0	100	4867	12.0	100
16	1431	15.0	94	1376	15.7	98	3770	15.5	97
24	940	22.9	95	964	22.5	94	2566	22.8	95
32	764	28.2	88	723	29.9	94	2020	29.0	91
48	607	35.4	74				1444	40.6	85
24x2	**561**	**38.4**	**80**						
12x4	**543**	**39.6**	**83**						

The scalability of the main computational stages of the 3-D MPI code on the T3E is shown in Fig. 1. While the chemistry and the vertical transport stages scale nearly perfect, this is not the case with the advection. The main reasons are

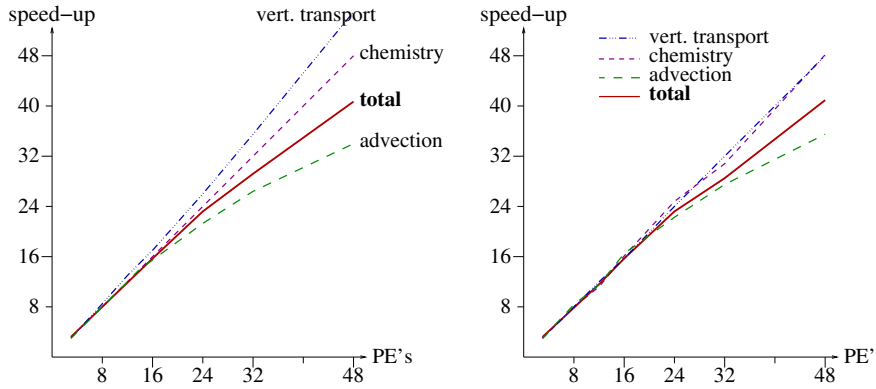


Fig. 1. Scalability of the main stages of the 3-D MPI code on the T3E. Experiments with two versions of the vertical transport stage are presented in the two plots. Results of the simpler version (without using chunks on the vertical transport stage) are given in the left plot. The version in the right plot uses chunks in order to improve the reuse of data in the cache. The superlinear speed-up of the vertical transport in the left plot indicates the cache-size effect on the performance (avoided by using chunks). Both versions use chunks on the chemistry stage.

the overhead due to subdomain overlapping and the non-optimal load-balance on that stage.

Some results of the new 3-D version for 288×288 grid on Sunfire 6800 computer (24 PE UltraSPARC 3, 750 MHz) at the Edinburgh Parallel Computer Centre (EPCC) are presented in Table 3.

5 Using the Model for Ozone Pollution Estimation with Respect to the Human Health

One of the most dangerous pollutants with strong influence in many areas like crops production, ecology and human health is the tropospheric ozone. Not only its concentrations are evaluated by DEM, but also several functions, related to one of the above areas [12,13,14].

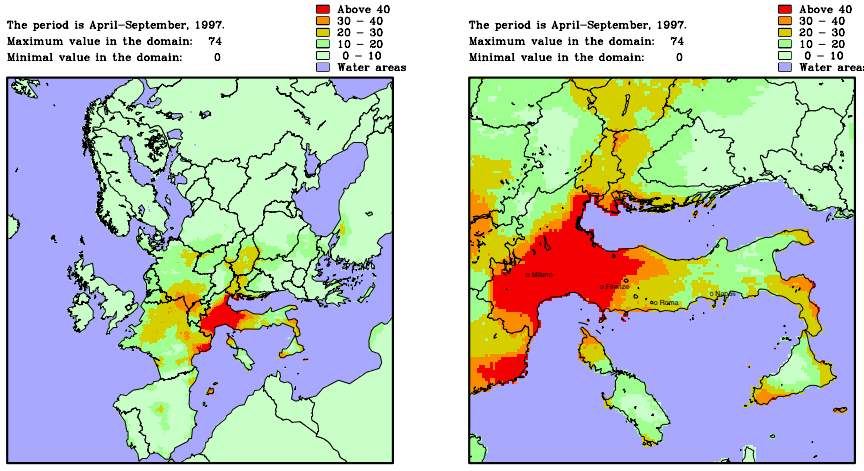
Let us call *"bad" days* the days, in which at least one of the 8-hour averages of the ozone concentration exceeds the critical value of 60 ppb.. In these days people suffering from asthmatic deceases have extra difficulties. There is a discussion in the European Union about this problem. The main suggestion is to reduce the emissions so that the number of "bad" days in one year do not exceed 20.

The plots in Fig. 2 represent in different colors the areas with given number of "bad" days in 1997. The 2-D version of the model with the fine-resolution (10×10 km.) grid is used to create these plots. Fig. 2 (a) represents the overall situation in Europe. The maximal value, 74 "bad" days, exceeds by a factor

Table 3. Performance results of the mixed MPI - OpenMP parallel code (in comparison with the pure MPI code) for the 3-D DEM on $(288 \times 288 \times 10)$ grid (assuming that on 4 proc. the speed-up is 4 and the efficiency – 100 %).

3-D version of DEM on Sunfire 6800 at EPCC $(288 \times 288 \times 10)$ grid, CHUNKSIZE=48					
Number of proc.	MPI tasks	OpenMP threads	Time [sec]	Speed -up	**E** [%]
4	4	1	114140	4.0	100%
6	6	1	79355	5.8	96%
8	8	1	55481	8.2	103%
12	12	1	41430	11.0	92%
12	6	2	38769	11.8	98%
16	16	1	29004	15.7	98%
16	8	2	29003	15.7	98%
24	24	1	19805	23.1	96%
24	12	2	22747	20.1	84%
24	6	4	22280	20.5	85%

of nearly four the accepted norm 20. The situation in Italy (Fig. 2 (b)) is not better. In some places in the Northern part of the country the global maximum of 74 "bad" days is reached.



(a) Europe (b) Italy

Fig. 2. Number of days with an 8-hour average ozone concentration more than the critical value (60 ppb.) in 1997.

6 Conclusions and Plans for Future Work

The standard parallelization tools (MPI for distributed-memory parallelism and OpenMP for shared-memory parallelism) can both be used on separate levels of parallelism to get the best use of the clustered parallel machines. The new hybrid code is highly portable and shows good efficiency and scalability on a variety of parallel machines.

An important and challenging task is development of a refined grid 3-D version of the model, in which the spatial domain is discretized on a $(480 \times 480 \times 10)$ grid. This is a huge computational task. Its solution is a big challenge to the power of the existing now supercomputers. Such a code is under development and positive results are expected in the near future.

If a long sequence of scenarios has to be run, then the two-dimensional versions are usually used. The 3-D versions are more accurate, but about 10 times more expensive (on both time and storage resources) with respect to the corresponding 2-D versions. This fact illustrates the need for further improvements (faster numerical algorithms, better exploitation of the potential power of the modern supercomputers, faster and bigger supercomputers, etc.).

Acknowledgments

This research is supported in part by grant I-901/99 from the Bulgarian NSF. Most of the results, presented in the paper, are obtained by Tz. Ostromsky during his visit to CINECA – Bologna, Italy in November 2001. This visit was supported by the EC via Access to Research Infrastructure action of the Improving Human Potential Programme via the MINOS project. We would like to thank all the people from CINECA for their hospitality and perfect conditions provided to the MINOS visitors. Special thanks to Prof. Giovani Erbacci and to Sigismondo Boschi for the technical support in order to get the best use of the supercomputing facilities at CINECA.

References

1. Alexandrov, V., Sameh, A., Siddique, Y., and Zlatev, Z.: Numerical integration of chemical ODE problems arising in air pollution models. Env. Modeling and Assessment, 2 (1997) 365–377.
2. Gropp, W., Lusk, E., and Skjellum, A.: Using MPI: Portable programming with the message passing interface. MIT Press, Cambridge, Massachusetts (1994)
3. Hesstvedt, E., Hov, Ø., and Isaksen, I. A.: Quasi-steady-state approximations in air pollution modeling: comparison of two numerical schemes for oxidant prediction. Int. Journal of Chemical Kinetics, 10 (1978) 971–994.
4. Hov, Ø., Zlatev, Z., Berkowicz, R., Eliassen, A., and Prahm, L. P.: Comparison of numerical techniques for use in air pollution models with non-linear chemical reactions. Atmospheric Environment, 23 (1988) 967–983.

5. Marchuk, G. I.: Mathematical modeling for the problem of the environment. Studies in Mathematics and Applications, 16, North-Holland, Amsterdam (1985)
6. McRae, G. J., Goodin, W. R., and Seinfeld, J. H.: Numerical solution of the atmospheric diffusion equations for chemically reacting flows. J. Comp. Physics, 45 (1984) 1–42.
7. Ostromsky, Tz., Owczarz, W., Zlatev, Z.: Computational Challenges in Large-scale Air Pollution Modelling. Proc. 2001 International Conference on Supercomputing in Sorrento, ACM Press (2001) 407–418.
8. Ostromsky, Tz., Zlatev, Z.: Parallel Implementation of a Large-scale 3-D Air Pollution Model. In: Margenov, S., Waśniewski, J., Yalamov, P. (eds.): Large-Scale Scientific Computing. Lecture Notes in Computer Science, Vol. 2179, Springer-Verlag (2001) 309–316.
9. WEB-site for OPEN MP tools, http://www.openmp.org
10. Zlatev, Z.: Computer treatment of large air pollution models, Kluwer (1995)
11. Zlatev, Z., Dimov, I., Georgiev, K.: Three-dimensional version of the Danish Eulerian Model. Zeitschrift für Angewandte Mathematik und Mechanik, 76 (1996) 473–476.
12. Zlatev, Z., Dimov, I., Ostromsky, Tz., Geernaert, G., Tzvetanov, I., and Bastrup-Birk, A.: Calculating Losses of Crops in Denmark Caused by High Ozone Levels. Env. Modeling and Assessment, 6 (2001) 35–55.
13. Zlatev, Z., Fenger, J., and Mortensen, L.: Relationships between emission sources and excess ozone concentrations. Computers and Math. with Appl. 32 (1996) 101–123.
14. Zlatev, Z., Geernaert, G., and Skov, H.: A Study of ozone critical levels in Denmark. EUROSAP Newsletter 36 (1999) 1–9.