On the Problem of Numerical Modeling of Dangerous Convective Phenomena: Possibilities of Real-Time Forecast with the Help of Multi-core Processors

N.O. Raba[1] and E.N. Stankova[1,2]

[1] Saint-Petersburg State University
198504 St.-Petersburg, Russia, Petergof, Universitetsky pr., 35
[2] Saint-Petersburg Electrotechnical University "LETI",
197376, St.-Petersburg, Russia, ul.Professora Popova 5
no13@inbox.ru, lena@csa.ru

Abstract. 1.5 - D convective cloud model with the detailed description of microphysical processes (including 8 categories of cloud particles) is presented in the article. The model is developed for investigation of convective cloud evolution and dangerous phenomena associated with it (thunderstorms, hails and rain storms). Special attention is paid for investigation of possibilities of real-time forecast of the phenomena using parallelization technique on multi-core processors. Effectiveness of parallelization has been investigated in relation to thread number and calculation work amount.

Keywords: multi-core processors, parallelization, thread, numerical model, real-time weather forecast, ice particles, thunderstorm, convective cloud.

1 Introduction

Investigation of dangerous convective phenomena such as thunderstorms, hail and rain storms) requires consideration of various processes having different nature and scale and presents an extremely complex problem for numerical modeling. Cloud model should reproduce both thermodynamical and microphysical processes. The first describe interaction of updraft and downdraft convective flows, turbulence vortexes and temperature variations. The second – transformations and interactions of small cloud particles – water drops, aerosols and various kinds if ice particles (ice crystals, snowflakes, graupel and frozen drops). Calculations of the whole set of the processes require a large number of computational resources and time, especially in case of using 2 - D and 3 - D models. Such models are usually applied for academic sresearches only. More simple 1 - D and 1.5 - D model could be used for real-time forecast especially in local weather centers, which are not able to obtain expensive supercomputers.

We have elaborated 1.5 D cloud model, characterized by detailed description of interaction of updraft flow inside a cloud and downdraft flow in cloud free environment. That allows reproducing more realistically the whole cycle of cloud

B. Murgante et al. (Eds.): ICCSA 2011, Part V, LNCS 6786, pp. 633–642, 2011.

evolution including the stage of its dissipation. Besides we have realized well-developed microphysical block with the full set of equations describing evolution of mass distribution functions of cloud condensation nuclei, water drops, and 6 types of ice particles. The model does not require supercomputer resources and it could be easily realized on ordinary desktops or workstations in local weather centers for producing forecast of dangerous convective phenomena.

We have investigated if it is possible to use the model for real-time forecast, when calculations should be obtained practically instantly. For this purpose we have -parallelized the model for further realization on multi-core processor computers. The so called space parallelization in conjunction with the multi-thread technology has been used. Effectiveness of parallelization has been investigated in relation to thread number and calculation work amount.

2 Model Description

The detailed description of the dynamical part of the model is presented in [1 and 2]. Below we briefly consider its main features, paying special attention to the updates in microphysical block, which differ the present model from the variant described in [2]. Convective cloud is presented in the model as a system of two cylinders. The inner cylinder (with constant radius a) corresponds to the updraft flow region (cloudy region) and the outer cylinder (with constant radius b) – to the surrounding downdraft flow region (cloudless). This structure was described firstly in [3] for description of cloud dynamics, but we used it for microphysical process simulation also. The model is 1.5-dimensional, that means that though all cloud variables are represented with mean values averaged over the horizontal cross section of the cloud, fluxes in and out of the inner cylinder borders are taken into account.

8 types of cloud particles were presented in the model: aerosol particles (cloud condensation nuclei, water drops, ice crystals in a form of columns, plates and dendrites, snowflakes, graupel and frozen drops).

In generalized form the equations for vertical velocity, temperature and mixing ratios of water vapour, water drops and ice particles inside the inner (equation 2) and outer (equation 3) cylinders can be written as follows:

$$\frac{\partial \phi_{in}}{\partial t} = -w_{in}\frac{\partial \phi_{in}}{\partial z} - \frac{2\alpha^2}{a} | w_{in} - w_{out} | (\phi_{in} - \phi_{out}) + \frac{2}{a} U_a (\phi_{in} - \phi_a) + \frac{1}{\rho_{a_0}} \frac{\partial}{\partial z} K_f \frac{\partial \phi_{in}}{\partial z} + F_{\phi_{in}} - A_{\phi_{in}} + G_{\phi_{in}}, \tag{1}$$

$$\frac{\partial \phi_{out}}{\partial t} = -w_{out}\frac{\partial \phi_{out}}{\partial z} - \frac{2\alpha^2}{a} | w_{in} - w_{out} | (\phi_{out} - \phi_{in}) + \frac{2}{a} U_a (\phi_{out} - \phi_a) + \frac{1}{\rho_{a_0}} \frac{\partial}{\partial z} K_f \frac{\partial \phi_{out}}{\partial z} + F_{\phi_{out}} - A_{\phi_{out}} + G_{\phi_{out}}, \tag{2}$$

Where the variables with subscripts 'in' and 'out' relate to the values, averaged over the inner and outer cylinders consequently. ϕ can take the values of vertical velocity w, temperature T, mixing ration of water vapor Q_v and mixing ratio of cloud droplets or ice particles in the i-th particle-size interval Q_{ji} (j=1 is for water drops, j=2,3,4 – for columnar crystals, plate crystals and dendrites, respectively, j=5,6,7 is for snowflakes, graupel and frozen drops respectively), t and z are independent variables (time and height consequently), α is the coefficient for lateral eddy mixing through the periphery of the cloud, U_a is determined by the equation of mass continuity under assumption of incompressibility which is given as $\frac{2u_a}{a} + \frac{1}{\rho_{a_0}}\frac{\partial(\rho_{a_0} w_{in})}{\partial z} = 0$, ρ_{a_0} is density of the atmospheric air, K_f is the turbulent viscosity coefficient.

Concrete form of the terms F_ϕ, A_ϕ, G_ϕ depends upon the meaning of ϕ.

$F_w = \frac{g(T_v - T_{v0})}{T_{v0}} - gQ_j$, where T_v is the virtual temperature, T_{v0} - is the virtual temperature averaged over the cross sections of the both cylinders, Q_j is the total mixing ratio of cloud particles (hydrometeors) $Q_j = \sum_{i=1}^{N-1} Q_{ji}$, $F_T, F_{Q_v}, F_{Q_{ji}}$ describe the input of the microphysical processes into the change of temperature and mixing rations of water vapor and cloud droplets in the i-th particle-size interval consequently. $A_w = A_{Q_v} = 0$, $A_T = -\gamma_a w/T$, where γ_a is the dry adiabatic lapse rate, $A_{Q_{ji}} = V_{di}\frac{\partial}{\partial z}(Q_{ji})$, where V_{di} - is the value of hydrometeor terminal velocity in the i-th size interval, $G_w = 0$, $G_T, G_{Q_v}, G_{Q_{ji}}$ describe the input of the microphysical process into the change of temperature and mixing rations of water vapor and hydrometeors in the i-th particle-size interval consequently.

Microphysical block of the model is based on solution of stochastic kinetic equations for mass distribution functions f_j describing 7 types of hydrometeors Q_{ji} ($j=1$ is for water drops, j=2,3,4 – for columnar crystals, plate crystals and dendrites, respectively, $j=5,6,7$ is for snowflakes, graupel and frozen drops respectively)).

For the numerical solution of the equations it is necessary to select discrete points m_i ($i = 0,..., N$, $m_0 = 0$) along the m axis to define particle size intervals or bins. Then one can replace the stochastic collection equation by the set of equations for M_{ji} - mass fraction in the mass interval defined by:

$$M_{ji} = \int_{m_{ji-1/2}}^{m_{ji+1/2}} m_j f_j(m_j) dm_j, \tag{3}$$

$i = 1,..., N - 1$.

Distribution functions for drops and ice particles are described by the equation:

$$\frac{\partial f_j}{\partial t} + (w - V_j)\frac{\partial f_j}{\partial z} = \left[\frac{\partial f_j}{\partial t}\right]_{nucl} + \left[\frac{\partial f_j}{\partial t}\right]_{cond/evap} + \left[\frac{\partial f_j}{\partial t}\right]_{coal} + \\ + \left[\frac{\partial f_j}{\partial t}\right]_{freez} + \left[\frac{\partial f_j}{\partial t}\right]_{melt} + \left[\frac{\partial f_j}{\partial t}\right]_{break} \tag{4}$$

where j (= 1,7) denotes the type of heydrometor, the term $\left[\frac{\partial f_j}{\partial t}\right]_{nucl}$ represents the rate of change of f_j due to nucleation processes, $\left[\frac{\partial f_j}{\partial t}\right]_{cond/evap}$ is the rate of the condensational growth/evaporation of droplets (for j = 1) or the deposition/sublimation rate of ice particles (for i > 1), $\left[\frac{\partial f_j}{\partial t}\right]_{coal}$ is associated with coalescence processes, and $\left[\frac{\partial f_j}{\partial t}\right]_{freez}$, $\left[\frac{\partial f_j}{\partial t}\right]_{melt}$, $\left[\frac{\partial f_j}{\partial t}\right]_{break}$ are the rates due to the freezing of liquid water and melting of ice, and breakup processes, respectively., V_j is the terminal velocity of hydrometeors.

The expressions for each term as well as the expressions for bulk densities, the diameter-length and thickness-diameter relations for ice crystals are taken the same as in [4].

Vertical distributions of environmental temperature and relative humidity together with initial impulses of temperature and velocity have been taken as initial conditions. All variables with the exception of temperature and mixing ratio of water vapor are equal to zero at the top and at the bottom boundaries of the cylinders.

Equations are numerically integrated using a finite difference method. Forward-upstream scheme is used. Vertical velocity is averaged over two grid points (point below is taken if w≥0 or point above if w<0). Modified Kovetz-Olund method [5] has been used for microphysical block equation integration.

Time-splitting method is used for sequential calculations of dynamical and microphysical process. Dynamical processes were calculated at the first stage, microphysical processes at the second one.

The results of numerical simulation show that the model is capable to describe processes of water and ice hydrometeor formation and evolution in convective clouds under various vertical distributions of temperature and relative humidity of the outer atmosphere. The model reproduces evolution of vertical velocity, mixing ratio of hydrometeors and hydrometeor spectrum in time and space. It can predict maximum and minimum values of the above mentioned dynamical and microphysical characteristics and besides the values of the height of a cloud base and upper boundary, precipitation rate and total quantity of the rainfall, snowfall and hail. All that characteristics are of major value for prediction of dangerous convective cloud phenomena such as thunderstorms, hails and rain storms.

Real-time forecast providing in the airports and local weather centers need models which can simulate dangerous event evolution nearly instantly. So even our model, which is low dimensional but sufficiently complex in microphysical description should be modified to be used for this purpose. As we mentioned above small local weather centers and airports do not possess high-performance computational facilities and need effective software to provide quick calculations on ordinary desktops. To use ordinary desktop means now to use multi-core desktop. That is why we tried to map our model on multi-core desktop using multi-thread technology for its parallelization.

3 Parallelization Method

The contribution of dynamical and microphysical blocks in overall computational time of the model is quite different. Dynamical block in 1.5-D model is rather simple and demands only about 20 seconds of computer time to obtain evolution characteristics of velocity, temperature and hydrometeor mixing rations during a cloud life cycle.

Computation of microphysical characteristics is essentially more time expensive. It depends upon a number of hydrometeor mass intervals or bins in each space mesh node and the number of hydrometeor types (number of distribution functions). If space mesh consists of the N nodes, the number of bins is equal to N_1, and the number of distribution functions is equal to N_2, then number of operations need to be performed for calculation of one dynamical time step is $O\left(N \cdot N_1 N_2\right)$. The same value for microphysical time step is $O\left(N \cdot N_1^2 N_2^2\right)$. So microphysical calculations will require in $k \cdot N_1 N_2$ more time than calculation of dynamics (k is some constant). Taking into account that the number of N_1 is more or equal to hundred, N_2 is equal to 7 we see that in case of microphysics the number of required calculations increases tremendously and the necessity of parallelization technique use becomes quite evident.

Though dynamical and microphysical processes are calculated sequentially in each node, calculations in several nodes can be provided in parallel.

Numerical scheme for the dynamical part of the model is an explicit one. So we can easily calculate all dynamical characteristics of the cloud at a time step “n+1” if we know them in each node of the mesh at a time step “n”. And though to calculate dynamical characteristic in a mesh node "i" we should know corresponding characteristic in a neighbor mesh node "i–1", or "i+1" we can easily do this as all necessary values have been already calculated at the previous time step. To perform space parallelization [6-9] we decompose computational domain of the model into several sub domains. Each sub domain represents a cylinder of the height Δh and includes parts of the inner and outer cylinders as well as a part of the environment at rest [1,10].

Multi-thread technology was used to realize parallelization methodology. Threads are created, and the data calculated on the previous time step is passed to the threads. It is essential that multi-core processors are in fact of SMP architecture type. So all threads can apply to shared memory where all the parameters calculated at the previous step are stored.

Each thread implements calculation within definite mesh nodes. The transfer to the next time step is implemented when all threads fulfill their calculations.

As at each time step processor should wait for completion of implementation of all threads, the problem of load balancing appears to be challenging. It is not easy to find the solution because calculation of cloud characteristics in different subsections demands quite different time due to the fact that it is not necessary to obtain microphysical characteristics in the mesh subsections where cloud hydrometeors are absent and relative humidity is less than 100%.

Special procedure of mesh subsection redistribution was used to obtain equal time of thread implementation. If "n" threads are launched and the certain thread has number "k", the latter will be responsible for calculation of microphysical and dynamical processes in the mesh nodes with the numbers $(k – 1) + i \cdot n$, $(i = 0, 1, \ldots)$. The procedure results in calculation of neighboring sub domains in different threads and provides acceptable level of load balancing.

As each launch of the thread demands definite time, the number of threads should be diminished in order to decrease computational overheads. It should be noted that some parts of the model program, such as creation and launch of the thread, calculation of boundary characteristics are calculated in single-thread regime.

6 Test Results

The aim of the test simulation was to define effectiveness of parallelization in relation to core and thread number and calculation work amount. We also try to define how the present results will differ from the ones obtained with the help of the model with the same dynamical but simpler microphysical block [1, 10], describing only "warm" cloud evolution without ice phase taking into account. Addition of 6 kinetic equations for distribution functions of ice particles should essentially increase amount of computational work and consequently parallelization effectiveness.

We provide calculations with the help of two core processor computer (Core 2 Duo CPU E8200, 2.66 GHz). Variable parameters are: number of threads for parallelization of microphysics and dynamics, space step, number of bin intervals for calculation of hydrometeor distribution functions. The results are presented in the tables 1-5.

Table 1. Calculation time (seconds) of 1 hr model cloud evolution obtained at different values of space step Δh (m) values; time step Δt = 10sec, the number of bins N_l = 101; t_1 – total time (sec), t_2 – time of dynamical processes calculation (sec), t_3 – time of microphysical processes calculation (sec). Results were obtained without any parallelization (only 1 thread is launched)

Δh	t_1	t_2	t_3
200	138	22	116
100	222	42	180
50	397	88	309

The results presented in the table 1 show that the most part of calculation time is spent for microphysical process calculation for all space steps. t_3 exceeds t_2 in 5 times at Δh = 200 m. and in 3.5 times at Δh = 200 m.

Relationship between t_3 and t_2 depends also upon the stage of cloud evolution. The hydrometeor particle spectra is rather narrow at initial and final stages of cloud development and thus do not need large time for their calculation. Time for microphysical process calculation starts growing at mature stage of cloud development when large precipitation particles are forming and spectra become wider. Numerical experiments conducted for Δh = 200 m, Δt = 10sec and the number of bins N_l = 101 show that ratio of t_3 to t_2 varies from 2.55 at the first 5 min. of cloud development to 11.05 at time period from 20 to 25 min of cloud evolution at the expense of time for microphysical part calculation (6 and 27 sec respectively). Time for dynamical part calculation remains constant and equal to about 2,5 sec. Later on hydrometeor spectra become once more narrow due to precipitation on the ground and time for microphysical processes calculation reduces.

Table 2. Calculation time (seconds) of 1hr model cloud evolution obtained with the help of different number of bins (N_1), used for calculation of hydrometeor distributon functions. Δt = 10sec, Δh = 200m (only 1 thread is launched).

N_l	t_1	t_2	t_3
101	138	22	116
151	230	32	198
201	370	42	328
251	481	52	429

Data presented in table 2 shows that number of bins is a crucial parameter which effects greatly calculation time value. Increasing of number of bins in 2.5 times increases calculation time value in 3.5 times. So we can state that microphysical part needs to be parallelized first of all. The results presented in tables 3 and 4 justify that conclusion.

The results presented in the table 3 show that parallelization of dynamical processes decrease calculation time insignificantly only at about 6,5 percents (129 sec versus 138) at relatively large (Δh = 200m) step of space mesh. The best results are obtained when the number of threads is equal to the number of cores (NTh_l = 2).

Increasing of thread number is practically insignificant. Effectiveness of parallelization at smaller values of Δh slightly increases and at Δh = 50m calculation time decreases at 8,5% in case of $NTh_1 = 2$.

Table 3. Calculation time (seconds) of 1 hr model cloud evolution obtained with the help of different number of threads (NTh_1) used for dynamical process parallelization. NTh_2 – number of threads for microphysics is equal to 1, Δt = 10sec, Δh = 200m, $N_1 = 101$

NTh_1	t_1	t_2	t_3
1	138	22	116
2	129	13	116
4	129	13	116
8	132	14	118
16	133	17	116

Table 4. Calculation time (seconds) of 1hr model cloud evolution obtained with the help of different number of threads (NTh_2) used for microphysical process parallelization. NTh_1 – number of threads for dynamics is equal to 1, Δt = 10sec, Δh = 200m, $N^1 = 101$

NTh_2	t_1	t_2	t_3
1	138	22	116
2	91	22	69
4	92	21	70
8	94	23	71
16	99	22	77

The results presented in the table 4 show that parallelization of microphysical process is much more effective than parallelization of dynamical ones. Calculation time in case of 2 threads decreases approximately at 33% in comparison with 1 thread calculation. The best results are obtained when the number of threads is equal to the number of processor cores ($NTh_2 = 2$). Similar to parallelization of dynamical processes, number of threads influences slightly upon parallelization effectiveness.

It is evident that the best results will be achieved when calculation of both dynamical and microphysical processes will be parallelized. It is justified by the data presented in table 5.

Table 5. Calculation time (seconds) of 1hr model cloud evolution obtained with the help of different number of threads used both for dynamical and microphysical processes parallelization. ($NTh_1 = NTh_2 = NTh$), Δt = 10sec, Δh = 200m, $N^1 = 101$

NTh	t_1	t_2	t_3
2	81	11	70
4	83	13	70
8	86	15	71
16	94	17	77

Results presented in table 5 show that we can archive speed up of 1.7 (40% decrease of calculation time) if we parallelized both computation of microphysical and dynamical processes with the equal number of threads and that number should be equal to the number of processor cores.

Our investigations show that we can decrease time of calculation up to 45% (264sec. versus 481sec) if we use 2 threads for parallelization of both dynamical and microphysical processes for the maximum bin number (N^1 = 251). In this case we have to fulfill maximum amount of calculation work and respectively, effect of parallelization is more evident.

We have also compare present results with the ones obtained with the help of the previous model version [2,8], which does not contain block for calculation of ice particle evolution and considers only water drop distribution function ("warm" microphysics). Parallelization is more effective in the latter case. Calculation time in case of 1 thread is equal to 37 sec. and 18 sec in case of 2 threads using for both microphysical and dynamical processes calculation (N^1 = 251, Δt = 10sec, Δh = 200m). Speedup is equal almost to 2 (50% decrease of calculation time) respectively. We think that addition of ice phase increases the amount of sequential work which resulted in decrease of parallelization effectiveness.

7 Conclusions

Convective cloud model characterized by detailed description of interaction of updraft flow inside a cloud and downdraft flow in cloud free environment and well developed microphysical block is presented in the paper. The model is capable to simulate evolution of cloud condensation nuclei, water drops and 6 types of ice particles. We investigate the possibilities to use the model for real-time forecast of dangerous convective phenomena by means of its parallelization with the help of multi-core computers. Investigation results show that calculation of microphysical processes is much more computationally expensive than calculation of dynamical processes. So parallelization of microphysical part of the model is much more effective than dynamical one. Maximum decrease of computation time for dynamics parallelization is 8.5% versus 33% for microphysics parallelization. The best results have been achieved when calculation of both dynamics and microphysics are parallelized (45% of computation time or speedup equals to 1.8). Number of threads influence slightly on parallelization effectiveness. Maximum speedup has been achieved when the number of threads are equal to the number of processor cores. Comparison of the present results with the ones obtained with the help of the previous model version (without ice phase) show that addition of ice phase increases amount of sequential work that is resulted in decrease of parallelization effectiveness.

References

1. Raba, N.O., Stankova, E.N.: Research of influence of compensating descending flow on cloud's life cycle by means of 1.5-dimensional model with 2 cylinders. In: Proceedings of MGO, vol. 559, pp. 192–209 (2009) (in Russian)

2. Raba, N., Stankova, E.: On the possibilities of multi-core processor use for real-time forecast of dangerous convective phenomena. In: Taniar, D., Gervasi, O., Murgante, B., Pardede, E., Apduhan, B.O. (eds.) ICCSA 2010. LNCS, vol. 6017, pp. 130–138. Springer, Heidelberg (2010) ISBN 978-3-642-12164-7
3. Asai, T., Kasahara, A.: A Theoretical Study of the Compensating Downward Motions Associated with Cumulus Clouds. Journal of the Atmospheric Sciences 24, 487–497 (1967)
4. Khain, A., Sednev, I.: Simulation of Precipitation Formation in the Eastern Mediterranean Coastal Zone Using a Spectral Microphysics Cloud Ensemble Model. Atmospheric Research 43, 77–110 (1996)
5. Stankova, E.N., Zatevakhin, M.A.: The modified Kovetz and Olund method for the numerical solution of stochastic coalescence equation. In: Proceedings 12th International Conference on Clouds and Precipitation, Zurich, August 19-23, pp. 921–923 (1996)
6. Voevodin, V.V.: Informational structure of sequential programs. Russ. J. of Num. An. and Math. Modelling 10(3), 279–286 (1995)
7. Voevodin, V.V.: Mathematical foundations of parallel computing. Series in computer science, vol. 33, p. 343. World Scientific Publishing Co., Singapore (1992)
8. Bogdanov, A.V., Korkhov, V.V., Mareev, V.V., Stankova, E.N.: Architectures and topologies of multiprocessor computational systems, p. 176 (2004) (in Russian)
9. Babb, R.G. (ed.): Programming Parallel Processors. Addison-Wesley Publishing Company, Reading (1988)
10. Raba, N., Stankova, E.: On investigation of parallelization effectiveness with the help of multi-core processors. Procedia Computer Science 1(1), 2757–2762 (2010)