Ocean-Land-Atmosphere Model: Features and Performance Evaluation*

Claudio Schepke, Nicolas Maillard
Grupo de Processamento Paralelo e Distribuído
Instituto de Informática – Universidade Federal do Rio Grande do Sul
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil
{cschepke,nicolas}@inf.ufrgs.br

Abstract

The Ocean-Land-Atmosphere Model (OLAM) is a global numerical simulation model based on the Regional Atmospheric Modeling System (RAMS). OLAM provides local mesh refinement to cover selected geographic areas with more resolution. It introduces a new dynamic core based on a global geodesic grid with triangular mesh cells and uses also a finite volume discretization of the full compressible Navier Stokes equations. This paper describes OLAM basis and functionalities as well as parallel performance evaluation of the model. The results show good performance in parallel executions on a cluster validating the implementation of the model.

1. Introduction

Numerical models have been used extensively in the last years in order to understand and predict the climate and weather phenomena. In general, two approaches were followed for the development of numerical models: global and regional. Global models have spatial resolution of about 2-5 degrees of latitude and therefore can not represent very well the scale of regional weather phenomena. To represent all meteorological processes is necessary a large number of points on the grid. Consequently, this requires a large power processing capacity. Moreover, regional models have more accurate resolution, but are used for specific regions. It must integrate large-scale atmospheric conditions into its borders side. Therefore it does not simulate the phenomena of large scale.

To solve this problem recently was developed at Duke University a new model that represents a new generation of meteorological models. The main feature of this model called **Ocean-Land Atmosphere Model** (OLAM) is the ability to represent global phenomena weather and also allows grids nesting with high resolution enabling more accurate representation of the phenomena of local scale [3].

OLAM was developed to extend features of the Regional Atmospheric Modeling System (RAMS) to a global model domain. OLAM uses many functions of RAMS, including physical parametrizations, data assimilation, initialization methods, logic and coding structure, and I/O formats [2]. OLAM introduces a new dynamic core based on a global geodesic grid with triangular mesh cells. It use also a finite volume discretization of the full compressible Navier Stokes equations. The refinement of a local mesh can be made in OLAM to cover selected geographic areas with more resolution. This is similar as applied in regional simulation atmospheric models.

OLAM was developed in FORTRAN 90 and recently parallelized with MPI. Because this, we chose this model to evaluate its parallel performance and to describe its formulation basis and functional structure. Next section details OLAM features. Section 3 discuss the domain decomposition of the model. A parallel performance evaluation executed in a multicore/multicomputer system is presented in Section 4. At the last section some considerations are described.

2. Ocean-Land-Atmosphere Model

Ocean-Land-Atmosphere Model (OLAM) is a numerical simulation model based on the Regional Atmospheric Modeling System (RAMS) [3]. RAMS is a limited area model designed to simulate mesoscale and cloud-scale processes for scientific study and numerical weather prediction [1].

RAMS is the state-of-the-art in terms of parametrizations. It represents cloud and precipitation microphysical processes, land-vegetation-atmosphere interactions, radiative transfer, subgrid turbulent transfer, and subgrid cumulus convection. The RAMS grid is relocatable to any place on earth, and can be sized to cover an area from only a few meters across to as large as a hemisphere. Grid nesting is

* Apoio: CAPES

often used in RAMS to provide very high resolution in selected regions of a large computational domain. These features, plus a large selection of options for configuring and customizing any simulations, have made RAMS a highly versatile research tool.

In spite of RAMS' assets, its limited-area domain has obvious disadvantages, particularly in limiting the range and scope of scientific problems that can be investigated. Thus, OLAM was developed as a global version of RAMS in order to eliminate these disadvantages and extends model applicability to global scale circulations and to two-way dynamic interactions between global and local scales. The effort in this approach was to retain many components of RAMS and to replace those that were not adaptable to the global domain. In particular, both models adopt similar physical parametrizations, coding and logical structures, I/O file formats, and procedures for compilation, initialization, and execution.

A new dynamic core was developed for OLAM because some aspects of the RAMS dynamic core are unsuitable for global modeling. RAMS uses a polar stereographic projection, which is not expandable to the globe. It makes the Boussinesq approximation based on a horizontally homogeneous reference state that cannot accurately represent widely in different environments within the same domain. However, in several aspects the OLAM dynamic core is similar to RAMS.

OLAM has a local mesh refinement capability, which is essential for intended high resolution simulations of mesoscale and microscale phenomena over limited geographic areas. Other RAMS features that were incorporated into OLAM are the momentum form of the Navier-Stokes equations, a finite volume discretization of advective and turbulent transport, and explicit representation of acoustic waves using numerical time splitting.

3. Global Grid Structure

OLAM's global computational mesh consists of spherical triangles, a type of geodesic grid that is a network of arcs that follow great circles on the sphere [3]. The geodesic grid offers important advantages over the commonly used latitude-longitude grid. It allows mesh size to be approximately uniform over the globe, and avoids singularities and grid cells of very high aspect ratio near the poles. OLAM's grid construction begins from an icosahedron inscribed in the spherical earth, as is the case for most other atmospheric models that use geodesic grids.

An icosahedron is a regular polyhedron that consists of 20 equilateral triangle faces, 30 triangle edges, and 12 vertices, with 5 edges meeting at each vertex. The icosahedron is oriented such that one vertex is located at each geographic pole, which places the remaining 10 vertices at latitudes of $\pm tan^{-1}(1/2)$. Uniform subdivision of each icosahedral triangle into $N \times N$ smaller triangles, where N is the number of divisions, is performed in order to construct a mesh of higher resolution to any degree desired. The subdivision adds $30(N^2 - 1)$ new edges to the original 30 and $10(N^2 - 1)$ new vertices to the original 12, with 6 edges meeting at each new vertex. All newly constructed vertices and all edges are then projected radially outward to the sphere to form geodesics. Figure 1 shows an example of the mesh at this step with $N = 10$. Heavier lines denote original undivided icosahedron. The projection causes most triangles to deviate from equilateral shape, which is impossible to avoid [3].

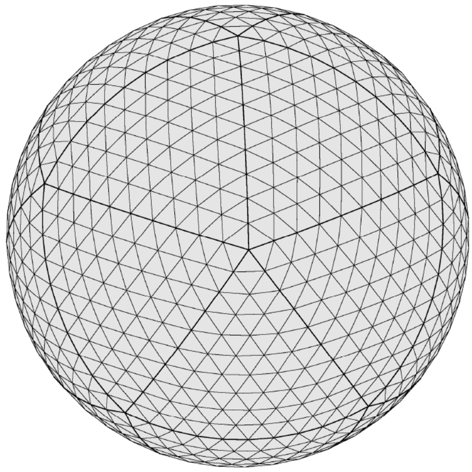

Figure 1. OLAM subdivided icosahedral mesh

OLAM uses an unstructured approach and represents each grid cell with single horizontal index [3]. Required information on local grid cell topology is stored and accessed by means of linked lists. If local horizontal mesh refinement is required, it is performed at this step of mesh construction. Refinement follows a three-neighbor rule that each triangle must share finite length of its edges with exactly three other triangles. The range of possible topologies that obey this rule is enormous. A example of local mesh refinement is showed in Figure 2 where resolution is exactly doubled in a selected geographic area by uniformly subdividing each of the previously existing triangles into $2x2$ smaller triangles. Auxiliary edges are inserted at the boundary between the original and refined regions for adherence to the three-neighbor rule. Each auxiliary line in this example connects a vertex that joins 7 edges with a vertex that joins 5 edges. More generally, a transition from coarse to fine resolution is achieved by use of vertices with more than 6 edges on the coarser side and vertices with fewer than 6 edges on the

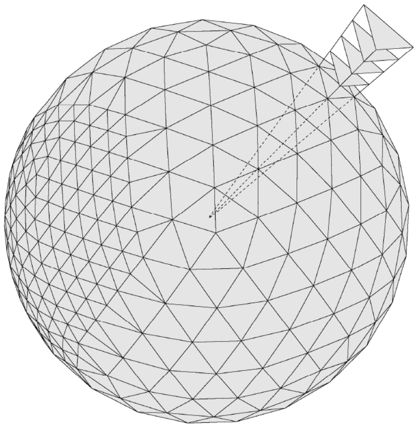

Figure 2. Example of local mesh refinement applied to left portion of figure, and projection of a surface triangle cell to larger concentric spheres to generate multiple vertical model levels. In this example, $N = 5$.

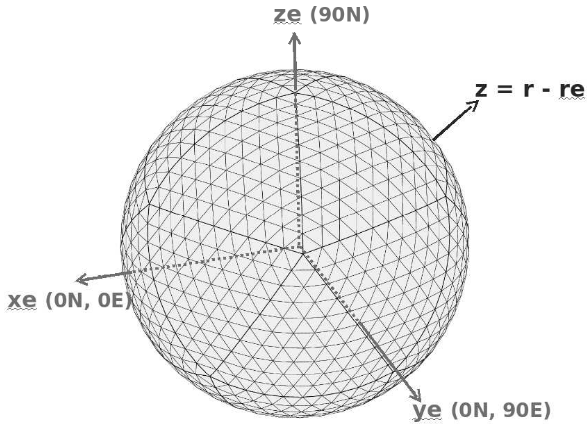

Figure 3. Cartesian coordinate system with origin at Earth center

finer side of the transition.

The final step of the mesh construction is the definition of its vertical levels. To do this, the lattice of surface triangular cells is projected radially outward from the center of the earth to a series of concentric spheres of increasing radius. The vertices on consecutive spheres are connected with radial line segments (see Figure 2). This creates prism-shaped grid cells having two horizontal faces (perpendicular to gravity) and three vertical faces. The horizontal cross section of each grid cell and column expands gradually with height. The vertical grid spacing between spherical shells may itself vary and usually is made to expand with increasing height.

OLAM uses a C-staggered grid discretization for an unstructured mesh of triangular cells [3]. Scalar properties are defined and evaluated at triangle barycenters, and velocity component normal to each triangle edge is defined and evaluated at the center of each edge. The numerical formulation allows for nonperpendicularity between the line connecting the barycenters of two adjacent triangles and the common edge between the triangles. Control volume surfaces for horizontal momentum are the same as for scalars. This is accomplished by defining the control volume for momentum at any triangle edge to be the union of the two adjacent triangular mass control volumes. This means that no spatial averaging is required to obtain mass flux across momentum control volume surfaces.

OLAM uses a rotating Cartesian system with origin at the Earth's center, z-axis aligned with the north geographic pole, and x- and y-axes intersecting the equator at 0 deg and 90 deg E. longitude, respectively, as shown in the Figure 3. The three-dimensional geometry of the computational mesh, particularly relating to terms in the momentum equation and involving relative angles between proximate grid cell surfaces, is worked out in this Cartesian system. The procedure involves computation and storage of the unit vector normal to each surface, and solution of linear systems that contain the unit vector coefficients.

4. Performance Evaluation

In this section we present an evaluation of the parallel performance of OLAM 3.0 version. For that, the code of OLAM was executed on the cluster ICE of Instituto de Informática at Universidade Federal do Rio Grande do Sul. This cluster is composed by 14 Dell PowerEdge 1950 nodes. Each node has two processors Intel Xeon E5310 Quad Core of 1.60 GHz. Each processor has 2x4MB of cache memory (1066 FSB) and uses EM64T technology. The interconnection of the nodes is realized by a 3Com model 2816 switch. We used MPICH version 2-1.0.7 for communication among the processes.

We divide each side of icosahedral triangle in 25 parts in us case study. So, the distance among the points on the globe was around 200 Km. The atmosphere layer (z dimension) was divided in 28 layers. We simulate 24 hours of integration of an atmosphere without any physical calculation, because we have interest only on the impact of fluid dynamics executions and communications cost. Each timestep of integration simulate 60 seconds of the real time.

Figure 4 show a comparison of the parallel execution time of OLAM, distributing the processes among the cluster nodes in three different ways. These ways consist in distribute cyclically the processes respectively in one core, four cores and eight cores per processor. For example, suppose that the global domain of the model is divided in 18 processes. Adopting the distribution approach using *one core*,

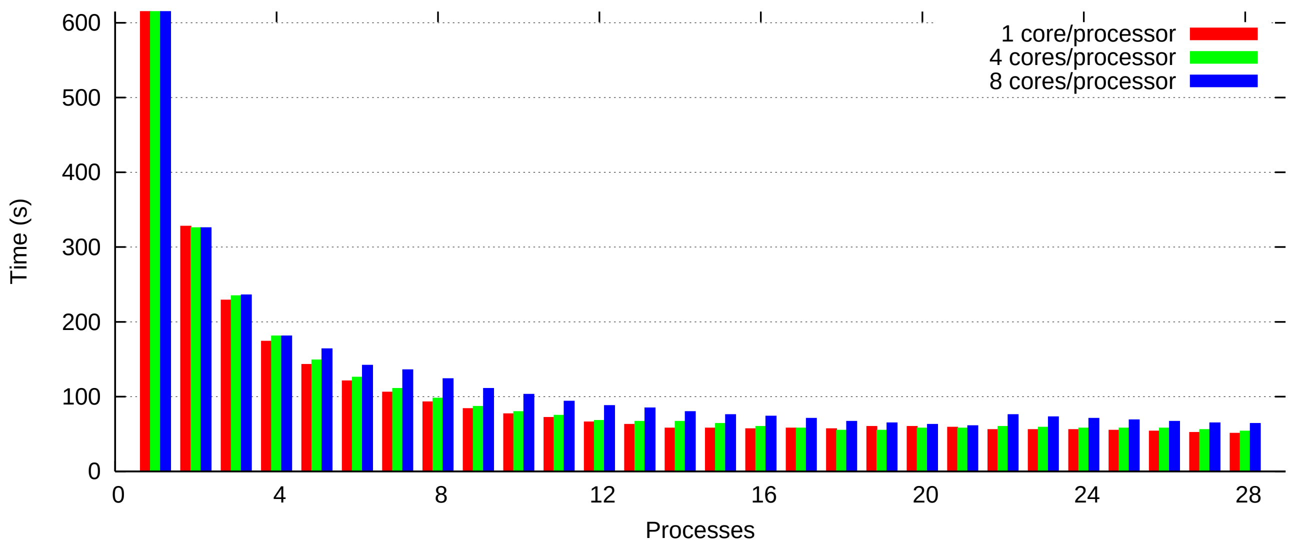

Figure 4. Execution time

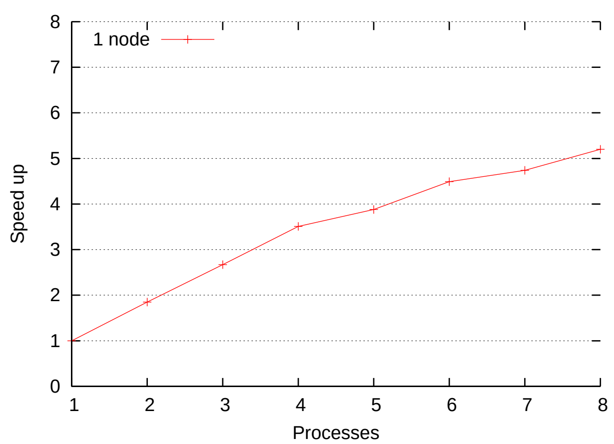

Figure 5. Speed up

4 cluster nodes will receive 2 processes and the other 10 nodes only one process. Using the *four cores* approach, 4 nodes will execute 4 processes and one node 2 processes. The *eight cores* approach will distribute 8 processes in 2 nodes and 2 processes will run in one node. So, it is possible to evaluate de impact of multicores in the parallel performance of OLAM.

The results show that using only a core per node is better than use more cores per node. Quad-core processors share the bus access. Because this, the performance on access simultaneously the memory is not so good, instead large volume of data are manipulated in OLAM code.

In another test, we executed OLAM in a only node. In Figure 5 is presented the speed up obtained using all 8 cores of two processors in a node. The results show a increase of performance when more processes are used. However, the speed up is less expressive when more than 4 processes are executed, because the access to the memory is limited.

5. Conclusion and Future Works

This paper described the formulation of the Ocean-Land-Atmosphere Model (OLAM). The model consists essentially of a global triangular-cell grid mesh with local refinement capability, the full compressible nonhydrostatic Navier-Stokes equations, a finite volume formulation of conservation laws for mass, momentum, and potential temperature, and numerical operators that include time splitting for acoustic terms.

The present work focused on evaluating the parallel potentialities of OLAM. The numerical simulations realized in this study provides the evaluation of the implementation, showing good performance for parallel executions. In future works more tests will occur, including more mesh refinement and inclusion of a domain decomposition distribution controller.

References

[1] R. A. Pielke and et al. A Comprehensive Meteorological Modeling System - RAMS. *Meteorology and Atmospheric Physics*, 49(1):69–91, 1992.

[2] R. L. Walko and R. Avissar. OLAM: Ocean-Land-Atmosphere Model - Model Input Parameters - Version 3.0. Technical report, Duke University, November 2008.

[3] R. L. Walko and R. Avissar. The Ocean-Land-Atmosphere Model (OLAM). Part I: Shallow-Water Tests. *Monthly Weather Review*, 136(11):4033–4044, 2008.