

## A Dynamically Adapting Weather and Dispersion Model: The Operational Multiscale Environment Model with Grid Adaptivity (OMEGA)

DAVID P. BACON, NASH'AT N. AHMAD, ZAFER BOYBEYI, THOMAS J. DUNN, MARY S. HALL, PIUS C. S. LEE, R. ANANTHAKRISHNA SARMA, AND MARK D. TURNER

*Center for Atmospheric Physics, Science Applications International Corporation, McLean, Virginia*

KENNETH T. WAIGHT III, STEVE H. YOUNG, AND JOHN W. ZACK

*MESO Inc., Troy, New York*

(Manuscript received 30 March 1999, in final form 13 July 1999)

### ABSTRACT

The Operational Multiscale Environment Model with Grid Adaptivity (OMEGA) and its embedded Atmospheric Dispersion Model is a new atmospheric simulation system for real-time hazard prediction, conceived out of a need to advance the state of the art in numerical weather prediction in order to improve the capability to predict the transport and diffusion of hazardous releases. OMEGA is based upon an unstructured grid that makes possible a continuously varying horizontal grid resolution ranging from 100 km down to 1 km and a vertical resolution from a few tens of meters in the boundary layer to 1 km in the free atmosphere. OMEGA is also naturally scale spanning because its unstructured grid permits the addition of grid elements at any point in space and time. In particular, unstructured grid cells in the horizontal dimension can increase local resolution to better capture topography or the important physical features of the atmospheric circulation and cloud dynamics. This means that OMEGA can readily adapt its grid to stationary surface or terrain features, or to dynamic features in the evolving weather pattern. While adaptive numerical techniques have yet to be extensively applied in atmospheric models, the OMEGA model is the first model to exploit the adaptive nature of an unstructured gridding technique for atmospheric simulation and hence real-time hazard prediction. The purpose of this paper is to provide a detailed description of the OMEGA model, the OMEGA *system*, and a detailed comparison of OMEGA forecast results with data.

### 1. Introduction

Over the past 40 years, numerical weather prediction has undergone a decade-by-decade advance. The 1960s saw the initial success of the barotropic model. In the 1970s came the baroclinic model and the introduction of the university-born Regional Atmospheric Modeling System (Pielke et al. 1992), and the Pennsylvania State University–National Center for Atmospheric Research (Penn State–NCAR) Mesoscale Model now in its fifth version (Grell et al. 1994). In the 1980s increased computational power allowed the introduction of more elaborate model physics. In the 1990s mesoscale models have driven toward finer and finer resolutions, mostly through the use of nested grids (e.g., the Advanced Regional Prediction System model; Xue et al. 1995) or

variable horizontal resolution (e.g., the Global Environmental Multiscale model; Côté et al. 1998a,b).

At the same time meteorology was benefiting from this research and technology boom, computational fluid dynamics (CFD) researchers were creating new innovative numerical techniques designed to model fluid flows around complex boundaries. In the 1970s and early 1980s the models developed for aerospace engineering and plasma physics were surprisingly similar to their counterparts in the atmospheric sciences. The grids were composed of regular, rectangular cells extending from no-slip or free-slip surfaces. As more computational power became available and atmospheric modelers were pushing more physics into their models, CFD practitioners were busy refining complex gridding techniques around irregular surfaces.

The Operational Multiscale Environmental Model with Grid Adaptivity (OMEGA) was developed under a program that linked the new gridding technologies of computational fluid dynamics with state-of-the-art numerical weather prediction. This paper will describe the model grid structure, the dynamical equations, physics, and parameterizations. It will also introduce the con-

---

*Corresponding author address:* Dr. David P. Bacon, Science Applications International Corporation, 1710 Goodridge Dr., P.O. Box 1303, McLean, VA 22102.  
E-mail: David.P.Bacon@saic.com

cepts and techniques of dynamic grid adaptation and one case study with extensive analysis that provides a validation of the model. Tests have been performed on several components of the model; significant ones, such as tests of the advection scheme, are presented in the paper. The final testing of the model, however, is in its performance. OMEGA has been running in an operational mode at several sites for the past three years; it has also performed well in several field experiments. It is not practical to present all of the results in one paper, however; so we present a single case study that was especially challenging due to the scarcity of data over the domain and the complexity of the terrain involved.

OMEGA is a multiscale, nonhydrostatic atmospheric simulation model with an adaptive grid that permits a spatial resolution ranging from roughly 100 km to less than 1 km without the need for nested grids.

This paper will introduce the meteorological community to OMEGA's unstructured grid approach to numerical weather prediction and real-time hazard prediction. OMEGA was designed as a multiscale simulation tool that simulates finescale flows without the need for multiple nested grids. OMEGA was built to address atmospheric transport and diffusion applications. To improve the fidelity of hazardous dispersion models, it is essential that the meteorological forecast itself be improved. This is because the modeling of atmospheric dispersion involves virtually all scales of atmospheric motion from microscale turbulence to planetary-scale waves.

Current operational atmospheric simulation systems (Hoke et al. 1989; Janjic 1990; Mesinger et al. 1988) use fixed grid spacing or rely on nesting to achieve a cascade of scales. The trick to using these models in forecasting mode is to place the nest over all areas of concern and to do so within economical and computational limits. Even with recent advances in computational power (McPherson 1991), the current architecture and physics of today's generation of atmospheric models cannot fully simulate the scale interaction of the atmosphere. Although several groups have developed nonhydrostatic, nested (multiply nested in some cases) atmospheric models (Dudia 1993; Skamarock and Klemp 1993), these represent an incremental evolutionary path in atmospheric simulation; as long as the same basic physics is solved, with roughly the same order of accuracy, and the same grid resolution, the computational performance of different simulation systems must be roughly the same.

For the reasons given above, it is impossible to change the basic performance of an atmospheric simulation system without changing the basic paradigms utilized. OMEGA advances the state of the art in numerical weather prediction through the application of advanced numerical methods including a dynamically adapting triangular prism computational mesh; it advances the state of the art in dispersion modeling by embedding the dispersion calculation within the NWP model giving

it access to the full resolution of the atmospheric simulation at every time step.

The basic philosophy of OMEGA development has been the creation of an operational tool for real-time hazard prediction. The model development has been guided by two basic design considerations in order to meet the operational requirements: 1) the application of an unstructured mesh numerical technique to atmospheric simulation, and 2) the use of an embedded atmospheric dispersion algorithm. In addition, as an operational tool, OMEGA was constructed using the maximum amount of automation, in model configuration, data ingest, data quality control, and data assimilation, grid generation, model operation, and postprocessing.

The first version of OMEGA (v. 1.0) went into operation in 1995. Since that time, a large number of additions and improvements have been made to the model including new hydrodynamic solvers, new surface and boundary layer models, and new system capability. This paper presents the current version of the OMEGA modeling system (v. 4.0) and a case study utilizing the model in a stressing, data-sparse situation.

The OMEGA modeling system is an operational real-time meteorological forecasting and atmospheric dispersion system. The OMEGA system consists of the following:

- routines to maintain and manage real-time weather data feeds from the National Oceanic and Atmospheric Administration, National Centers for Environmental Prediction (NCEP), and/or the U.S. Navy Fleet Numerical Meteorology and Oceanography Center (FNMOC);
- worldwide datasets for surface elevation, land/water, vegetation coverage, soil type, land use, deep soil temperature, deep soil moisture, and sea surface temperature at varying resolutions;
- an integrated graphical user interface (XOMEGA) that provides a user-friendly method for rapid model reconfiguration;
- the OMEGA grid generator that accesses the surface datasets and creates OMEGA grid and terrain files;
- a meteorological data preprocessor that ingests gridded terrain, gridded meteorological analyses and forecasts, and raw observations, and performs a detailed quality control of the ingested data, followed by an optimum interpolation (OI) data assimilation to produce initial and boundary conditions for OMEGA;
- the OMEGA atmospheric simulation model and its embedded Atmospheric Dispersion Model (ADM);
- the OMEGA graphical postprocessing tool (XGRID) that enables the user to display OMEGA output as two-dimensional slices (horizontal slices overlaid on mapping information from the Digital Chart of the World or vertical slices), skew  $T$ - $\log p$  profiles for any location, and animations; and
- additional postprocessors to provide for data extraction and reformatting for external applications.

TABLE 1. An overview of OMEGA.

Governing equations	Fully nonhydrostatic
Dimensionality	3D
Grid structure	Unstructured triangular prisms
Grid adaptivity	Both static and dynamic grid adaptation
Coordinate system	Rotating Cartesian coordinates
Numeric	Finite volume
PBL	Treated separately as viscous sublayer, surface layer, and transition layer
Turbulence closure	Order 1.5 turbulent kinetic energy closure
Cumulus parameterization	Modified Kuo scheme
Microphysics	Extensive bulk water
Radiation	Shortwave absorption by water vapor and longwave emissivities of water vapor and carbon dioxide
Lower boundary	Based on Monin–Obukhov similarity theory
Upper boundary	Rigid, free-slip surface
Lateral boundaries	Radiative boundary condition, large-scale nudging boundary condition
Initialization	Based on 4D data assimilation
Transport and diffusion	Embedded Eulerian and Lagrangian (Monte Carlo particle and probabilistic puff) aerosol dispersion algorithms

The resulting modeling system is capable of rapid reconfiguration for operation anywhere in the world, with automatic linkage to baseline datasets and real-time meteorological data feeds. The description of the main components of the system will be discussed in the following sections.

## 2. The OMEGA model description

The basic features of the OMEGA model are provided in Table 1. OMEGA is a fully nonhydrostatic, three-dimensional prognostic model. It is based on an adaptive, unstructured triangular prism grid that is referenced to a rotating Cartesian coordinate system. The model uses a finite-volume flux-based numerical advection algorithm derived from Smolarkiewicz (1984). OMEGA has a detailed physical model for the planetary boundary layer (PBL) with a 2.5 level Mellor and Yamada (1974) closure scheme. OMEGA uses a modified Kuo scheme to parameterize cumulus effects (Kuo 1965; Anthes 1977), and an extensive bulk water microphysics pack-

age derived from Lin et al. (1983). OMEGA models the shortwave absorption by water vapor and longwave emissivities of water vapor and carbon dioxide using the computationally efficient technique of Sasamori (1972). OMEGA uses an optimum interpolation analysis scheme (Daley 1991) to create initial and boundary conditions and supports piecewise four-dimensional data assimilation using a previous forecast as the first guess for a new analysis. Finally, OMEGA contains both Eulerian (grid based) and Lagrangian (grid free) dispersion models embedded into the model.

### a. The OMEGA grid structure

A unique feature of the OMEGA model is its unstructured grid. The flexibility of unstructured grids facilitates the gridding of arbitrary surfaces and volumes in three dimensions. In particular, unstructured grid cells in the horizontal dimension can increase local resolution to better capture topography or the important physical features of atmospheric circulation flows and cloud dynamics. The underlying mathematics and numerical implementation of unstructured adaptive grid techniques have been evolving rapidly, and in many fields of application there is recognition that these methods are more efficient and accurate than the structured logical grid approach used in more traditional codes (Baum and Löhner 1994; Schnack et al. 1998). To date, however, unstructured grids and grid adaptivity have not been used in the atmospheric science community (Skamarock and Klemp 1993). OMEGA represents the first attempt to use this CFD technique for atmospheric simulation.

OMEGA is based on a triangular prism computational mesh that is unstructured in the horizontal dimension and structured in the vertical (Figs. 1 and 2). The rationale for this mesh is the physical reality that the atmosphere is highly variable horizontally, but always stratified vertically. While completely unstructured three-dimensional meshes have been used for other pur-

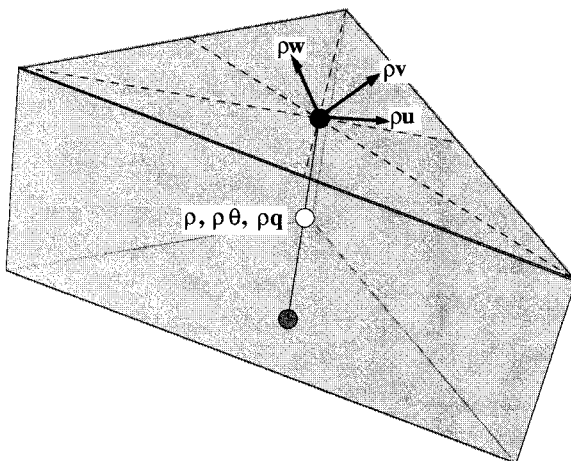


FIG. 1. OMEGA grid element.

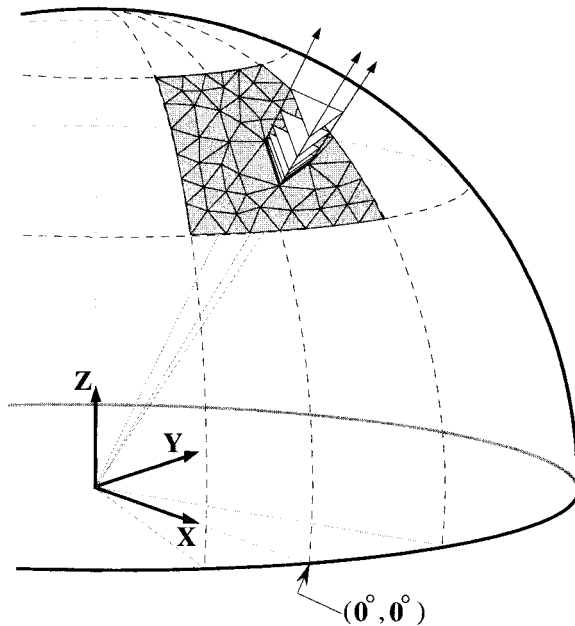


FIG. 2. OMEGA coordinate system and vertical alignment of OMEGA grid.

poses (Baum et al. 1993; Luo et al. 1994), the benefit of having a structured vertical dimension in an atmospheric grid is a significant reduction in the computational requirements of the model. Specifically, the structured vertical grid enables the use of a tridiagonal solver for implicit solution of both vertical advection and vertical diffusion. Since in many grids the vertical grid spacing is one or more orders of magnitude smaller than the horizontal grid spacing, the ability to perform vertical operations implicitly relaxes the limitation on the time step.

In discussing unstructured grids, it is first necessary to define the elemental dimensional objects that describe the properties of a volumetric mesh. The lowest-order object of a grid is the *vertex*, which is specified by its position  $(x, y, z)$ . An *edge* conveys the connectivity of the mesh and is defined by the indices of its starting and terminating vertices. The *face* represents the interface area between adjacent volumetric cells and can be described from the list of edges that bound it, or by the sequence of vertices that form its corners. The *cell* or *control volume* is in turn specified by the list of faces that contain it. In this volumetric mesh, scalar quantities are defined in the cell centroid, while the vector quantities are defined at the center of vertically stacked faces (Fig. 1).

On an unstructured grid, the number of edges that meet at a vertex is arbitrary. Consequently there is no longer a simple algebraic construct that can be used to deduce the relationship of indices for the various elemental objects, as in the case of structured grids that have been used as the basic structure for atmospheric and ocean circulation models up until now. Rather, the

formation of the grid is tied to the actual solution of the model equations and to the topography. This means that the initial grid can be readily adapted to surface features or other fixed terrain features as well as the initial weather.

An important feature of the unstructured triangular grid methodology is the calculation of the normal to each face, which is required to calculate the flux across the face. Since these normals must be computed, there is no benefit from orienting the grid in any particular fashion, so long as the numerical resolution is sufficient to evaluate the critical fluxes. This leads to a natural separation between the coordinate system for the fundamental equation set and the grid structure. The coordinate system can be as simple as possible (such as Cartesian) while the grid structure, in this coordinate system, is extremely complex. OMEGA uses a rotating Cartesian coordinate system, but the grid structure is terrain following. Figure 2 shows a rotating Cartesian coordinate system in which the origin is the center of the earth, the  $z$  axis passes through the North Pole, the  $x$  axis passes through the intersection of the equator and the prime meridian, and the  $y$  axis is orthogonal to both.

In this coordinate frame, the equations of motion are in their simplest possible form (without going into a nonrotating frame that would lead to unusual boundary conditions as the surface terrain moved through the grid) with only two terms that are somewhat nonconventional: gravity and the Coriolis acceleration. Gravity in this frame is directed in the radial  $(-\hat{r})$  direction, which implies that it potentially has components in all three coordinate directions. The Coriolis force is by definition  $-2\rho\boldsymbol{\Omega} \times \mathbf{v}$  and likewise has components in all three directions.

An important aspect of the OMEGA grid structure is that vertically stacked cells all possess the same footprint. This is accomplished by creating a surface grid and then projecting radials from the center of the earth through the vertices of this grid. Horizontal layers are constructed by specifying a set of vertices along each radial (Fig. 2). Because the OMEGA grid structure results in the mixing of the earth-relative horizontal and vertical components, it is essential that the numerical scheme be able to separate these. Given a grid structure that may be a few meters in vertical resolution and a few kilometers or a few tens of kilometers in horizontal resolution, the numerical resolution must be accurate to better than 1 part in  $10^5$ .

#### b. Fundamental equation set

In this section, we document the fully elastic non-hydrostatic equation set used in OMEGA, including the applicable assumptions. For brevity, we classify the five mixing ratios for water substances into two groups: precipitating water substances,  $Q_p$  (where the subscript  $p$  designates either rain or snow), and nonprecipitating water substances,  $Q_n$  (where  $n$  designates either ice crys-

tals, water vapor, or water droplets). Furthermore, we cast the equations in their conservative form consistent with the fully elastic mass-conservation equation ( $\partial\rho/\partial t = -\nabla \cdot \rho\mathbf{V}$ ). This form is better suited for the state-of-the-art upwind advection schemes we have used, which have significantly less numerical diffusion (important for the finer resolution we are trying to achieve) than the schemes used in other nonhydrostatic models such as the Terminal Area Simulation System (Proctor 1987).

We begin by decomposing the atmospheric pressure and density into a time-invariant hydrostatic base state and a perturbation upon that state such that

$$p(x, y, z, t) = p_0(x, y, z) + p'(x, y, z, t), \quad \text{and} \quad (1)$$

$$\rho(x, y, z, t) = \rho_0(x, y, z) + \rho'(x, y, z, t), \quad \text{with} \quad (2)$$

$$\frac{\partial p_0}{\partial z}(x, y, z) = -g\rho_0(x, y, z). \quad (3)$$

We then have the following equation set for the conservation of mass, momentum, and energy.

*Conservation of mass:*

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho\mathbf{V}) = F_\rho,$$

$$\frac{\partial \rho Q_n}{\partial t} + \nabla \cdot (\rho Q_n \mathbf{V}) = \rho M_n + F_{Q_n},$$

$$\frac{\partial \rho Q_p}{\partial t} + \nabla \cdot (\rho Q_p \mathbf{V}) = \rho M_p + \frac{\partial}{\partial z}(Q_p W_p \rho) + F_{Q_p},$$

and

$$\frac{\partial \rho Q_a}{\partial t} + \nabla \cdot (\rho Q_a \mathbf{V}) = \rho M_a + \frac{\partial}{\partial z}(Q_a W_a \rho) + F_{Q_a}, \quad (4)$$

where the subscript  $a$  refers to aerosols or gases. In Eqs. (4),  $\rho$  represents the dry air density, calculated from the total density as

$$\rho = \rho_{\text{total}}/(1 + Q_{\text{vapor}}). \quad (5)$$

*Conservation of momentum:*

$$\begin{aligned} \frac{\partial \rho \mathbf{V}}{\partial t} + \nabla \cdot (\rho \mathbf{V} \mathbf{V}) &= -\nabla p' - (\rho - \rho_0)g\hat{\mathbf{r}} \\ &\quad - 2\rho\boldsymbol{\Omega} \times \mathbf{V} + \mathbf{F}_M, \end{aligned} \quad (6)$$

where the density used in the calculation of momentum is the total density.

*Conservation of energy:*

$$\frac{\partial E}{\partial t} + \nabla \cdot (E\mathbf{V}) = \frac{E}{c_p T} \sum_j (L_j S_j) + S_R + F_h. \quad (7)$$

Terms have been arranged such that the conservative advection terms appear on the left side of each equation. The source terms on the right side of the momentum equation also include buoyancy and gravitational effects,  $-(\rho - \rho_0)g\hat{\mathbf{r}}$  (where  $\hat{\mathbf{r}}$  is the radial unit vector),

and the Coriolis force ( $-2\rho\boldsymbol{\Omega} \times \mathbf{V}$ ). Subgrid-scale turbulence contributions,  $F$ , are discussed in detail later in this section. For the remaining equations;  $T$  is the temperature,  $L_i$  and  $S_i$  denote the latent heat and rate of phase conversion of either vaporization, fusion, or sublimation; and  $W_p$  represents the terminal velocity of each of the precipitating water substances. In addition,  $S_i$  depends on the microphysics that governs the rate of phase transitions and  $W_a$  depends on the assumed size distribution and mass of the hydrometeors. Also,  $M_n$  and  $M_p$  are the nonprecipitating and precipitating microphysics source terms. Finally, subscript  $a$  refers to transport of aerosol or gas.

The initial energy density  $E$  is calculated from the total pressure as

$$E = p^{1-\kappa} \left( \frac{p_{\text{ref}}^\kappa}{R_d} \right) \quad (8)$$

and throughout the simulation pressure is diagnosed from energy density using this relationship. Potential temperature,  $\theta$ , and temperature,  $T$ , are related by Poisson's equation,

$$T = \theta \left( \frac{p}{p_{\text{ref}}} \right)^\kappa, \quad (9)$$

where  $p_{\text{ref}} = 1000$  mb,  $\kappa = R_d/c_p = 0.286$ ,  $R_d$  is the gas constant for dry air, and  $c_p$  is the specific heat of dry air at constant pressure. Finally,  $S_R$  represents the contribution of radiation flux to heating the atmosphere.

### c. Hydrodynamic solver

The hydrodynamic elements of the OMEGA model are based on numerical methods of solution of the Navier–Stokes equations on an unstructured grid in the horizontal direction and a structured grid in the vertical. In the calculation of momentum, the pressure gradient, Coriolis, and buoyancy terms are calculated explicitly along with the advection terms. An implicit vertical filter and an explicit horizontal filter are applied to the vertical momentum. The calculation of the new momentum at each time step thus involves several steps, which are described below. All implicit operations are performed by tridiagonal matrix inversion.

#### 1) ADVECTION—IMPLEMENTATION OF THE FINITE VOLUME UPWIND SOLVER

In this section we describe the implementation on unstructured grids of the multidimensional positive definite advection transport algorithm (MPDATA) originally developed on regular grids by Smolarkiewicz (1984), Smolarkiewicz and Clark (1986), and Smolarkiewicz and Grabowski (1990). The resulting scheme is second-order accurate in time and space, conservative, combines the virtues of the MPDATA (e.g., ability to separately ensure monotonicity and positive definite-

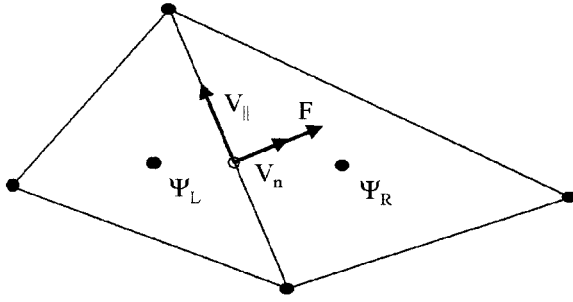


FIG. 3. Arrangement of the variables on a two-dimensional unstructured grid. The generic advected variable,  $\Psi$ , is placed at cell centroids. The subscripts L and R designates their left-hand-side and right-hand-side cell placements. The velocity is decomposed into normal and parallel components (not to scale). The flux  $F$  is collocated with  $V_n$  at the face centers and points only in the normal direction.

ness) with the flexibility of unstructured grids (Baum and Löhner 1994), is compatible with adaptive mesh algorithms (Fritts 1988), and can run efficiently on highly parallel computers. Below, we describe the essential methodology and demonstrate the method on two-dimensional passive advection test problems.

In discussing unstructured grids, it is necessary to define the nomenclatures. To reiterate, the basic control volume element in our structured–unstructured computational domain is a truncated triangular prism. Each prism is bounded by five faces. For advection across each face, it is convenient to define a local coordinate system with its origin located at the center of the face. Each face separates the left-hand side (lhs) from the right-hand side (rhs) such that the flow from the lhs cell to the rhs cell is considered positive. For stability, the advected variable, hereafter denoted as  $\Psi$ , is placed at the cell centroid, while the velocity vector is defined on the cell face at the origin of the local coordinate system. Figure 3 shows the basic arrangement of the variables on a two-dimensional grid.

In its explicit form MPDATA adapts naturally to the above construct. As posed by Smolarkiewicz, the algorithm can be generalized to the following steps.

- 1) At each cell face the low-order flux is found in conservative form using the standard first-order-accurate “upwind” scheme.
- 2) The advected variable is integrated using the low-order flux.
- 3) At each cell face, the low-order scheme is expanded in a Taylor series and the truncation error in the flux is explicitly identified.
- 4) The error term is cast in the form of error velocity,  $V_e$ .
- 5) The correction velocity,  $V_c$  ( $= -V_e$ ), is optionally limited to preserve monotonicity of the advected variable (Smolarkiewicz and Gradowski 1990).
- 6) Replacing  $V$  with  $V_c$ , steps 1 through 5 are repeated a chosen number of times ( $=N_c$ ) to achieve greater accuracy.

Although Smolarkiewicz derived  $V_c$  for a rectangular grid, it can be generalized to a grid with arbitrary control volume shape as long as the bounding faces are flat. Consider the generic advection equation,

$$\frac{\partial \Psi}{\partial t} + \nabla \cdot (\mathbf{V}\Psi) = 0. \quad (10)$$

To compute the change in  $\Psi$  from time  $t = t_0$  to  $t_0 + \Delta t$ , it is necessary to integrate the flux  $\Psi_j \mathbf{V}_j$  through each face  $j$  during the period  $\Delta t$ :

$$\Delta \Psi_j = \int_{t=t_0}^{t=t_0+\Delta t} \Psi_j \mathbf{V}_j \cdot \mathbf{a}_j dt. \quad (11)$$

Here  $\mathbf{a}_j = a_j \hat{\mathbf{e}}_n$  is the area vector of face  $j$ , where  $\hat{\mathbf{e}}_n$  denotes the unit vector normal to the face and pointing from left to right. For this integral, second-order accuracy in space is achieved automatically by placing  $\mathbf{F} = \Psi_j \mathbf{V}_j$  at the center of each face (in practice,  $\Psi_j$  at the faces is obtained by interpolation). Similarly, to ensure second-order accuracy in time,  $\Psi_j \mathbf{V}_j$  should be evaluated at  $t = t_0 + \Delta t/2$ . Assuming that a leapfrog algorithm is used (i.e.,  $\mathbf{V}_j$  is defined at  $t = t_0 + \Delta t/2$ ), we need only expand  $\Psi_j$  in a Taylor’s series as

$$\Psi_j = \Psi_j^0 + \frac{\partial \Psi_j}{\partial t} \frac{\Delta t}{2} + O(2), \quad (12)$$

where the superscript 0 denotes an evaluation at  $t = t_0$ . Substituting (12) for  $\Psi_j$  in (11) and performing the time integral, Eq. (11) becomes

$$\Delta \Psi_j = \left\{ \Psi_j^0 + \frac{\partial \Psi_j}{\partial t} \frac{\Delta t}{2} \right\} \mathbf{V}_j \cdot \mathbf{a}_j \Delta t. \quad (13)$$

Now substituting (10) for  $\partial \Psi / \partial t$  in (13) we have

$$\Delta \Psi_j = \left\{ \Psi_j^0 - \left[ \mathbf{V}_j \cdot \nabla \Psi_j + (\nabla \cdot \mathbf{V}_j) \Psi_j \right] \frac{\Delta t}{2} \right\} \mathbf{V}_j \cdot \mathbf{a}_j \Delta t. \quad (14)$$

If we further let  $v_n = \mathbf{V}_j \cdot \hat{\mathbf{e}}_n$  denote the component of the velocity normal to the face (and thus  $\mathbf{V}_n = v_n \hat{\mathbf{e}}_n$ ), then the first-order upwind flux term is given by

$$\Delta \Psi_j^{\text{upwind}} = 0.5 \{ (\mathbf{V}_n + |v_n| \hat{\mathbf{e}}_n) \Psi_L + (\mathbf{V}_n - |v_n| \hat{\mathbf{e}}_n) \Psi_R \} \cdot \mathbf{a}_j \Delta t. \quad (15)$$

Now letting  $\delta_L$  denote the vector pointing from the cell centroid on the left to the face, and  $\delta_R$  the vector pointing from the face to the cell centroid on the right, we can write

$$\Psi_L = \Psi_j - \delta_L \cdot \nabla \Psi_j \quad \text{and} \quad \Psi_R = \Psi_j + \delta_R \cdot \nabla \Psi_j. \quad (16)$$

Similarly, we will let

$$\delta \Psi_L = \Psi_j - \Psi_L \quad \text{and} \quad \delta \Psi_R = \Psi_R - \Psi_j. \quad (17)$$

Now we can rewrite (15) as

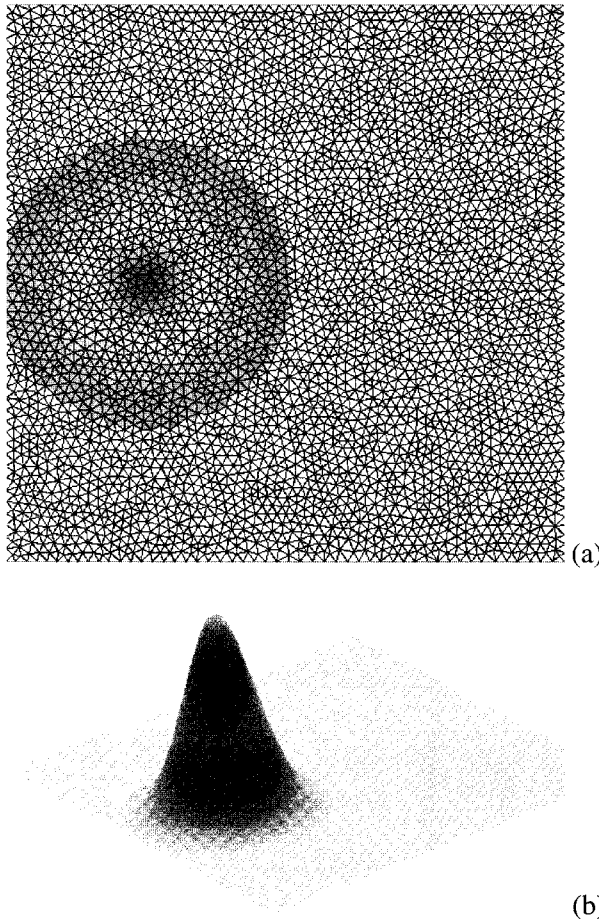


FIG. 4. Rotating cone advection test: (above) the computational mesh used for the test and (below) the initial conditions.

$$\Delta\Psi_j^{\text{upwind}} = \{\mathbf{V}_n\Psi_j^0 - 0.5|v_n|\hat{\mathbf{e}}_n(\delta_L\Psi + \delta_R\Psi)\} \cdot \mathbf{a}_j\Delta t. \quad (18)$$

The correction term is the difference between (14) and (18). After some algebraic manipulation, this correction term can be written as

$$\Delta\Psi_c = \left\{ |v_n| \left( \frac{\Psi_R - \Psi_L}{\Psi_R + \Psi_L} \right) - v_n \left[ \mathbf{V}_j \cdot \frac{\nabla\Psi_j}{\bar{\Psi}} + (\nabla \cdot \mathbf{V}_j) \right] \frac{\Delta t}{2} \right\} \bar{\Psi} a_j \Delta t, \quad (19)$$

where  $\bar{\Psi} = (\Psi_R + \Psi_L)/2$ . The correction velocity is now

$$\mathbf{v}_c = |v_n| \left( \frac{\Psi_R - \Psi_L}{\Psi_R + \Psi_L} \right) - v_n \left[ \mathbf{V}_j \cdot \frac{\nabla\Psi_j}{\bar{\Psi}} + (\nabla \cdot \mathbf{V}_j) \right] \frac{\Delta t}{2}. \quad (20)$$

In actual implementation, the correction term becomes

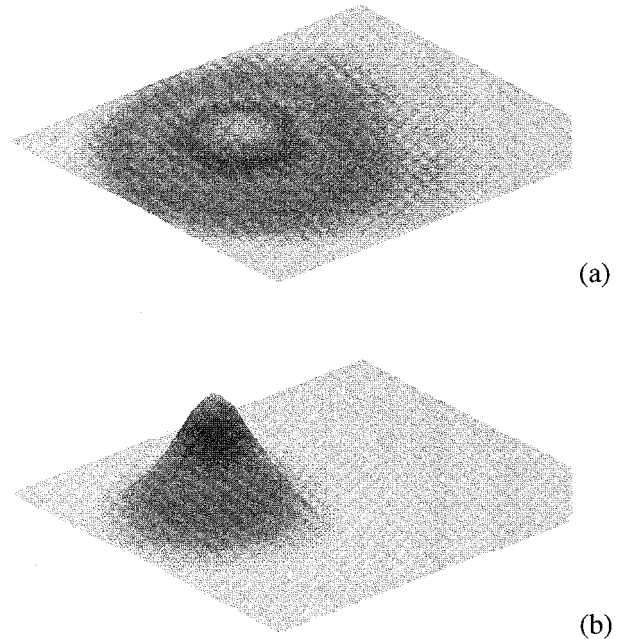


FIG. 5. Rotating cone advection test after five rotations. (above) The results using a first-order "upwind" scheme; (below) the effect of adding a correction step.

$$\Delta\Psi_c = \left\{ |v_n| \left( \frac{\tilde{\Psi}_R - \tilde{\Psi}_L}{\tilde{\Psi}_R + \tilde{\Psi}_L} \right) - v_n \left[ \mathbf{V}_j \cdot \frac{\nabla\tilde{\Psi}_j}{\tilde{\Psi}} + \nabla \cdot \mathbf{V}_j \right] \frac{\Delta t}{2} \right\} \tilde{\Psi}_{L,R} a_j \Delta t, \quad (21)$$

where  $\tilde{\Psi}$  is the value of the advected quantity following the first-order upwind advection step. We limit the number of correction steps to  $N_c = 1$ , since additional correction steps are not cost effective. The effect of additional correction steps is to bring the solution closer to second-order accuracy, which is very nearly achieved with just one correction step; the additional accuracy attainable is limited.

To compare with known results (Smolarkiewicz 1984; Smolarkiewicz and Clark 1986; Smolarkiewicz and Grabowski 1990), we performed the rotating cone test on triangular grids. Figure 4 shows the mesh and the initial placement of the passive scalar. Figure 5 shows the solutions after six revolutions of the cone. Figure 5a shows the results of the first-order "upwind" scheme ( $N_c = 0$ ). As shown, the original cone shape has completely diffused. In Fig. 5b, we show the case with one correction step. This case has a background value of zero, which when coupled with the positive definiteness aspect of the algorithm, generates extra numerical diffusion that automatically ensures monotonicity. Nevertheless, the cone shape has been substantially preserved. We have also performed other standard tests such as the multiple vortex test suggested by Staniforth (1987). In

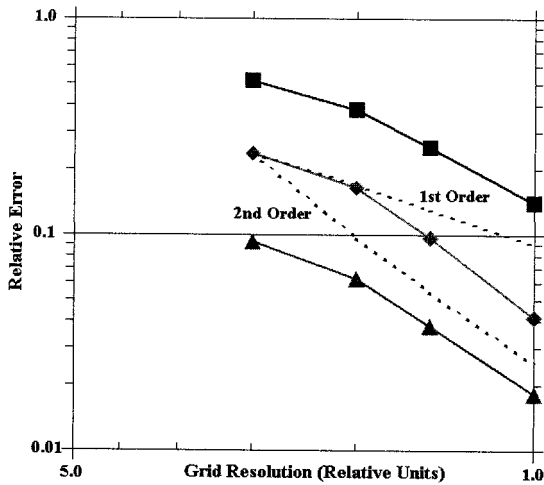


FIG. 6. Error scaling with grid resolution. Three measurements of deviation from the original cone are used: root-mean-square error over the whole mesh (triangles), root-mean-square error over an area extended by the original cone (diamonds), and deviation in the height of the cone (squares).

general, all of our results are consistent with the previous tests.

The claimed near-second-order accuracy of our scheme is demonstrated by testing how the accumulated error in the computation scales with grid size. Three measurements of error are used for the rotating cone test: the root-mean-square error over the whole mesh, the root-mean-square error over an area extended by the original base of the cone, and deviation in the height of the cone. The mesh sizes range from one-and-a-half to one-half of those shown in Fig. 4, that is, a factor of 3 variation. Figure 6 shows that except for the leftmost data point, all three measurements of error scale approximately as  $\Delta^2$  (the dotted lines indicate the slopes of both  $\Delta^2$  and  $\Delta$  scaling for comparison). The leftmost data points were generated from runs with the largest grid size.

## 2) IMPLICIT ACOUSTIC DAMPING

One of the major goals of the OMEGA model development is to achieve a simulation capability that transcends scales through the use of variable resolution. The accurate portrayal of aerosols in the planetary boundary layer will require fine resolution in the structured vertical direction and selective locations of fine horizontal resolution. Accurate, stable numerical solutions require that the time step of integration be determined by the finest grid spacing. This problem is magnified when using the fully compressible set of nonhydrostatic equations, since fast-moving sound waves are inherent in the solution. Indeed, in the structured direction, the CFL time step can become too restrictive. To overcome this restriction, the growth and propagation of acoustic waves are controlled through the application of vertical

and horizontal filters that act on the vertical component of the momentum.

The momentum equation is integrated using a semi-implicit, fractional time step scheme. After explicit calculation of the advection and source terms, an implicit calculation is performed in which a correction term is added to the vertical component of momentum. This term, which has the effect of slowing the growth and speed of sound waves, is described below. The next partial step consists of the application of the filter described in the following section. Like the implicit correction, this term is based on and added to the vertical momentum. The implicit calculation of the eddy diffusion term completes the momentum advance to time step  $n + 1$ .

The first implicit calculation solves the following equation for  $w^{\hat{n}}$ :

$$w^{\hat{n}} = w^n + c_s v_s \left[ \frac{\partial^2}{\partial z^2} (w^{\hat{n}} - w^n) \right] dt^2, \quad (22)$$

where  $w^n$  is the vertical component of momentum at the beginning of the  $n$ th time step and  $w^{\hat{n}}$  is the value after the addition of the (explicitly calculated) terms representing advection, the Coriolis force, the pressure gradient contribution, and the buoyancy effect. Here,  $v_s$  is the sound speed, and  $c_s$  is an empirically chosen coefficient. A stability analysis of the above equation shows that  $c_s$  must be greater than 1.0. The default value currently used in OMEGA is 2.5.

Eddy diffusion is also treated implicitly in the radial direction. The fractional step approach is necessary because the acoustic damping term described above effectively increases the inertia at short wavelengths, that is,  $\rho_{\text{effective}} \sim \rho(1 + k^2)$ , where  $k$  is the wavenumber in the direction of propagation. This can reduce the action of the eddy diffusion term unless the operations are split into separate fractional steps.

Again, this implementation of the dispersion operator removes the severe time step restriction that can result from diffusion flux along the finescale radial direction. A separate explicit treatment is used for momentum diffusion in the horizontal direction.

## 3) ACOUSTIC FILTERING

The purpose of filtering in the OMEGA model is to remove from the solution a particular type of undesirable feature: high-frequency oscillations in the pressure field. Since these oscillations are reminiscent of the undesirable oscillations occurring in spectral solutions, it is possible to borrow a filtering algorithm from spectral methodology, in which filtering can be accomplished in either physical or modal space. The filter applied in OMEGA is analogous to the Vandeven filter of order three when defined in physical space on a structured grid (cf., Karniadakis and Sherwin 1999). This filter, which is similar to a classical cosine filter, has the form



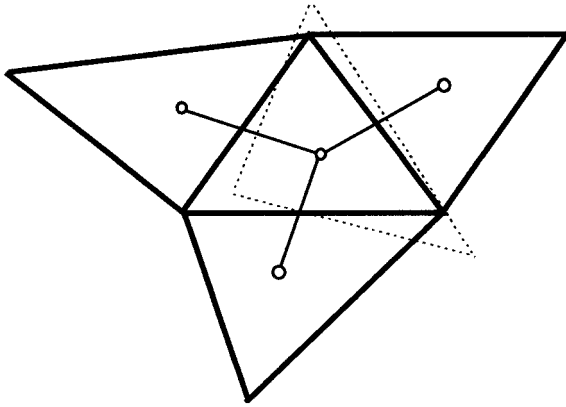


FIG. 7. The ghost triangle used in the calculation of the Laplacian.

of a second-order artificial viscosity term when the grid spacing is uniform.

Since the OMEGA grid is structured in the vertical direction, calculation of the vertical component of the filter is straightforward, while the horizontal component requires special consideration. The horizontal term is based on the horizontal component of the Laplacian, which is calculated using Green's formula:

$$\nabla^2 w = \frac{1}{A} \oint \frac{\partial w}{\partial \mathbf{n}}, \quad (23)$$

where  $A$  is the area enclosed by a curve. The discretized form becomes

$$\nabla^2 w = \frac{1}{A} \sum_{i=1}^3 \frac{\partial w}{\partial \mathbf{n}_i} dL_i, \quad (24)$$

where  $A$  is now the area of the triangle,  $\mathbf{n}_i$  is the outward normal vector to the  $i$ th edge, and  $dL_i$  is the length of the  $i$ th edge. The normal derivative is defined as

$$\frac{\partial w}{\partial \mathbf{n}_i} = \frac{w_i - w_c}{d_i}, \quad (25)$$

where  $w_i$  is the value of  $w$  at the centroid of the triangle that shares the  $i$ th edge of the target triangle,  $w_c$  is the value of  $w$  at the centroid of the target triangle, and  $d_i$  is the distance between the two centroids.

This approximation is first-order accurate only if the line segment connecting the centroids is perpendicular to the shared edge; that is, this line segment coincides with the outward normal vector. To improve the accuracy of the approximation, we define a "ghost" triangle whose edges are the perpendicular bisectors of the line segments connecting the centroid of the target triangle with the centroids of its neighbors. The area and edge lengths of this ghost triangle are used in the formula for  $\nabla^2 w$ . Figure 7 shows the ghost triangle constructed around the centroid of the inner cell.

#### 4) HORIZONTAL DIFFUSION

The horizontal diffusion term in the momentum equation can be expanded as

$$\nabla \cdot (K_d \nabla \mathbf{u}) = \nabla K_d \cdot \nabla \mathbf{u} + K_d \nabla^2 \mathbf{u}, \quad (26)$$

where  $K_d$  is the horizontal diffusion coefficient, given by

$$K_d = 0.36D\Delta^2. \quad (27)$$

Here  $D$  is the deformation rate, and  $\Delta$  is the horizontal grid spacing (McCorcle 1988). Assuming local deformation gradients are small, this can be written as

$$\nabla \cdot (K_d \nabla \mathbf{u}) \approx K_d \nabla^2 \mathbf{u}. \quad (28)$$

We define orthonormal vectors  $\mathbf{x}$  and  $\mathbf{y}$  lying in a plane tangent to the surface of the earth at the centroid of the triangle in which the deformation rate is sought. Projecting the velocity  $\mathbf{u}$  onto this local coordinate system gives  $u$  and  $v$ , the horizontal components of velocity. By assuming the velocity to be locally linear in  $x$  and  $y$ , it is possible to express  $u$  and  $v$  as  $a_u x + b_u y + c_u$  and  $a_v x + b_v y + c_v$ , respectively. For the horizontal deformation rate  $D$  we will use the approximation

$$D \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}. \quad (29)$$

Using the values of  $u$  and  $v$  at the vertices of the triangle, it is then possible to solve for  $b_u$  and  $a_v$ , which gives the local deformation rate as  $D = b_u + a_v$ . In this way, actual horizontal deformation is obtained, as opposed to deformation in the plane of the triangle. Since the horizontal grid spacing is squared in the expression for the deformation coefficient, we use the area of the triangle. The Laplacian of each component of the velocity is calculated using the discretization of Green's formula based on the ghost triangles described in the previous section.

#### 5) SUBCYCLING OVER SMALL CELLS

In OMEGA the integration time step depends on the local Courant number, which is a function of the velocity (advective or acoustic), and the dimension of the cell. The OMEGA user interface and grid generator allow the user to specify selected regions in which the resolution can be as fine as 1 km or less. The cell size in the outer portion of the domain may be nearly two orders of magnitude greater than this. Hence the time step can range from subsecond to tens of seconds depending on the flow characteristics and dimension of the cell. For this reason, it is highly desirable to utilize a scheme in which the hydrodynamic solver is subcycled over small cells, thus preventing the time step over the entire domain from being limited to that required by the smallest cell.

In OMEGA, we use the term "inner mask" to refer to a collection of cells over which the solution will be

subcycled. This terminology distinguishes such a collection of cells from a “subdomain,” which is user specified as part of the grid definition process. An inner mask is selected by the model, which calculates the Courant-limited time step for each cell based on sound speed and uses this as the criterion for mask definition. Cells in an inner mask need not be contiguous. We impose only the restriction that any cell having two neighbors in an inner mask is added to the inner mask. For stability, no more than two time steps are taken in a cell before the neighboring cell is updated; however, masks can be nested, so that the time step in the outermost portion of the domain can be four or more times that of the smallest cells.

The difficulty in implementing a time-splitting scheme consists in the treatment of the mask boundaries. In OMEGA, the edges of cells in an inner mask belong to the mask. Thus advection across those edges takes place at the smaller time step associated with the inner mask. Advected quantities are updated in cells within the inner mask, while quantities advected into or out of cells having only one edge on the inner mask are stored. Advected quantities in these bounding cells are updated simultaneously with advection across edges in the outer mask.

To test the time-split advection scheme, we used a test grid composed of a large domain with two levels of nesting. We performed the canonical “rotating cone” test, defining the initial conditions and the flow field in such a way as to cause the cone to pass directly through the high-resolution subdomain. The model took two time steps in the moderate-resolution area and four in the high-resolution area for each time step taken in the base-resolution area. There was no resultant deformation of the cone or degradation of the solution.

#### d. Model physics

This section briefly discusses the OMEGA model physics including atmospheric turbulence, cloud microphysics, convective parameterization, and radiation transport. Atmospheric turbulence affects, among other things, the rate at which surface moisture and energy enters the atmosphere. Cloud microphysics and convective parameterization are important in the vertical distribution of this moisture and energy and in the formation of precipitation. And finally, radiation transport is important in determining the solar heating of the earth’s surface and the atmosphere.

##### 1) TURBULENCE AND THE PLANETARY BOUNDARY LAYER

The parameterization of turbulence in OMEGA [i.e., the forcing terms,  $F$ , in Eqs. (4), (6), and (7)] is divided into two parts: horizontal and vertical. The horizontal diffusion is a function of the deformation of momentum (discussed in the previous section), while the vertical

diffusion is implemented using a multilevel planetary boundary layer model. The atmospheric boundary layer in OMEGA is treated separately as the viscous sublayer, the surface layer, and the transition layer.

The viscous sublayer is defined as the level near the ground ( $z < z_0$ ,  $z_0$  is the surface roughness) where the transfer of the dependent variables by molecular motion becomes important. Temperature and specific humidity in this layer are related to their ground surface values (Deardorff 1974). The surface layer is that part of the planetary boundary layer immediately above the viscous sublayer where small-scale turbulence dominates the transfer of energy, moisture, and momentum, and the vertical variation of the vertical fluxes is roughly less than 10%. In this layer, Monin–Obukhov similarity theory provides an acceptable framework for the interaction between the atmosphere and the underlying surface and virtually all numerical models make use of it one way or the other. In the OMEGA model, the revised formulas purposed by Beljaars and Holtslag (1991) are used. Finally, the vertical turbulent exchange above the surface layer is parameterized using the so-called level 2.5 closure technique developed by Mellor and Yamada (1974). This scheme (Mellor and Yamada 1982) has been selected from the hierarchy of available closure schemes, because it is a good compromise between accuracy and efficiency, and because it has been applied to many atmospheric modeling studies. The level 2.5 turbulence closure is based on the following prognostic equation for the turbulent kinetic energy,  $e$ :

$$\frac{\partial(\rho e)}{\partial t} = -\nabla \cdot (\rho e \mathbf{v}) + \nabla_r \cdot (K_e \nabla_r \rho e) + \left[ -\overline{u'w'} \frac{\partial u}{\partial z} - \overline{v'w'} \frac{\partial v}{\partial z} + \frac{g}{\theta} \overline{w'\theta'} \right] - \varepsilon. \quad (30)$$

The OMEGA model requires as boundary conditions the temperature and humidity at the roughness level. These must be calculated prognostically taking the interaction of the atmosphere and the land surface into account. Numerical studies have demonstrated that contrasts in land surface characteristics (e.g., soil texture and moisture, vegetation type) can generate local circulations as strong as sea breezes (McCorcle 1988; Chang and Wetzel 1991). In consequence, landscape variations strongly influence atmospheric dispersion patterns. The land surface module implemented in the OMEGA model is based on the scheme proposed by Noilhan and Planton (1989) and uses worldwide datasets for soil type, land use/land cover, vegetation index, climatological sea surface temperature, climatological subsurface temperature, and climatological soil moisture (discussed in section 5). These surface characteristics are used as primary parameters, while the spatial variation of secondary parameters such as surface roughness, albedo, thermal diffusivity of the soil, etc. are specified based on the primary parameters.

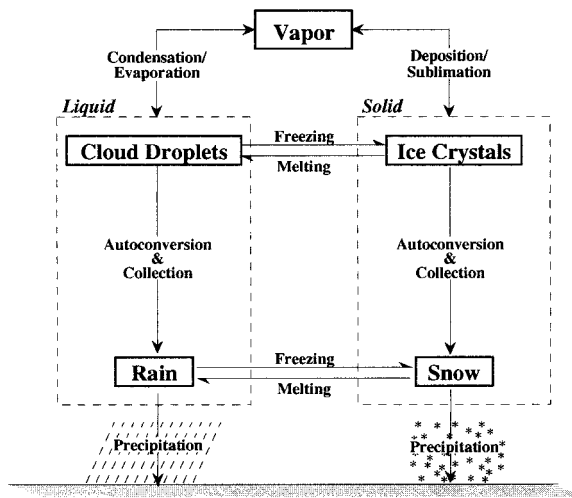


FIG. 8. OMEGA microphysical processes.

Finally, the surface of water bodies such as lakes and oceans is represented in the OMEGA model rather simply. Because of the large heat capacity of water and strong surface mixing, the surface temperature of water bodies,  $T_{\text{sea}}$ , is initially specified based climatological data and then assumed to be constant during the period of the mesoscale simulation. The air just above the water surface is assumed to be fully saturated by water vapor. The spatial variation of surface roughness length,  $z_0$ , is determined using the Charnock's relationship (Charnock 1955) over the water surface.

## 2) MICROPHYSICS

The OMEGA microphysics package falls under the category of *bulk water microphysics* in which the production rates are functions of the total mass density of each water species (Lin et al. 1983). The water species in OMEGA are divided into vapor, cloud droplets, ice crystals, rain and snow fields; however liquid and solid phases of water are not allowed to coexist. This implies that neither supercooled liquid nor melting solids (such as wet hailstones) can exist in the model domain. Because of this simplification, the number of microphysical processes to be dealt with is much smaller than that in many cloud models (Klemp and Wilhelmson 1978; Proctor 1987). Though liquid and solid phases of water are not allowed to coexist in OMEGA, OMEGA has separate array spaces allocated for all the different microphysical species. This will facilitate the addition of the physics dealing with melting hail and snow and supercooling of liquid droplets at a later date. The microphysical processes included in OMEGA (Fig. 8) are condensation (or evaporation) of cloud water, autoconversion of cloud water to rain, condensation on (or evaporation of) raindrops, accretion of cloud water by raindrops, fallout of rain, and generation of ice crystals from ice nuclei, deposition (sublimation) ice crystal growth,

autoconversion of ice crystals to snow, deposition on (or sublimation of) snow, accretion of ice crystals by snow, and finally fallout of snow.

## 3) CONVECTIVE PARAMETERIZATION

While the goal of OMEGA is to try to explicitly resolve large areas of convection, there will always be regions that are not sufficiently resolved. To circumvent this problem a version of cumulus parameterization that was originally proposed by Kuo (1965, 1974) and later modified by Anthes (1977) is incorporated to account for the effect of subgrid-scale deep cumulus convection on the local environment. The coupling between the subgrid-scale cumulus parameterization scheme and the explicit cloud microphysics is still a great research area for numerical modelers. More recently, Molinari and Dudek (1986) proposed that the use of explicit cumulus physics representations becomes necessary for horizontal grid resolutions less than 3 km. At this scale, large deep convective clouds are often resolvable (e.g., Lilly 1990). For horizontal grid scales larger than 50–60 km, Molinari and Dudek suggested using cumulus parameterizations of convectively unstable grid points and explicit condensation at convectively stable grid points. The most troublesome scales for parameterizing convective processes are those between 3 and 50 km.

Since OMEGA uses an unstructured grid, this scale spanning problem becomes an important issue. In OMEGA, the convective parameterization only applies to those regions that are convectively unstable to deep penetrative convection and in which the total horizontal moisture convergence exceeds a critical value. As the cumulus parameterization is a mechanism to account for subgrid convection in large cells, in OMEGA we also are using a simple concept by which we can smoothly transition from regions where no convective parameterization is applied to regions where it is applied. This is achieved by including a factor in the cumulus adjustment scheme that varies between 0 and 1 depending on cell area, while explicit cloud microphysics is active over the whole domain.

## 4) RADIATION

Because of the divergent behavior of different parts of the atmospheric radiation spectra, it is convenient to develop separate parameterizations for long and short wavelengths. The radiative source–sink term in the conservation of energy relation [Eq. (7)] can then be written as

$$S_R = \left( \frac{\partial T}{\partial t} \right)_{\text{LW}} + \left( \frac{\partial T}{\partial t} \right)_{\text{SW}}, \quad (31)$$

where the terms on the right-hand side represent the temperature change resulting from longwave and shortwave radiative divergence flux in the vertical direction.

The divergence of radiative energy in the horizontal direction is neglected, since its variation is much larger in the vertical direction on the mesoscale. The method of parameterizing this vertical flux takes into account the absorption of shortwave radiation by water vapor and the longwave energy emitted by water vapor and carbon dioxide using the computationally efficient technique of Sasamori (1972). It is essentially similar to the one used by Mahrer and Pielke (1977).

#### *e. Model initial conditions*

The function of the OMEGA data preprocessor is to convert a diverse mixture of real-time atmospheric and surface data for input into an OMEGA simulation. The process produces two datasets. One dataset specifies the initial values of all the model's dependent variables as well as the surface-based model parameters. A second dataset consists of the time-dependent behavior of all of the model's dependent variables along the lateral boundaries of the simulation domain. The OMEGA preprocessor can ingest real-time atmospheric data from NCEP and FNMOC. The core of the OMEGA analysis and initialization procedure consists of seven steps: 1) ingestion of the available data prepared by the PREP-DAT module and the preliminary INIT file created by the GENINIT module, 2) objective analysis of sea surface temperature data, 3) processing and quality control of surface and upper air data, 4) objective analysis of surface and upper air data, 5) adjustments to ensure that the analyzed data meet desired constraints, 6) generation of objective analysis performance statistics, and 7) output of the initialization data to a modified INIT file.

The multivariate algorithm in the OMEGA data preprocessor is based on that described by DiMego (1988), which adapts in a natural way to the unstructured grid. The analysis proceeds horizontally layer by layer, cell by cell. The analysis is performed at the top center of each OMEGA grid cell as well as at the bottom of the lowest layer of cells. Corrections to the first-guess field for height and wind are computed using the multivariate algorithm while those for relative humidity are computed using the simpler univariate algorithm. Corrections are computed for an entire layer and smoothed with a four-point smoother before being added to the first guess. The smoother sets the value at a given cell to 50% of the original value plus 1/6 the value at each of the three adjacent cells. The height corrections are converted to pressure corrections since OMEGA defines grid locations by absolute altitude. The variables are then vertically interpolated to cell centroids. Temperature is analyzed only at the surface. At cell centroids, it is derived hydrostatically from the pressure at the top and bottom of the cell. If a level is less than 10 mb from the last analyzed level, the analysis at this level is skipped. The corrections to the first guess are then interpolated from the corrections at the nearest analyzed layers above and below. This reduces the problem of

noisy temperature fields due to small differences in the height correction at closely spaced analysis layers.

The surface temperature data are used to hydrostatically adjust the surface height increment in the lowest levels as in DiMego (1988). At the ground, the surface height increment will be used directly. In the first few model levels above the ground (up to about 40 mb above the surface), the surface height increment is hydrostatically corrected by assuming that the surface temperature perturbation is valid over this layer. The corrected height increment is considered to be valid at the model level. In effect, a minisounding is created from each surface observation. The profile extends about 40 mb above the ground. Only the nearest height increment to a given level is included in the analysis matrix at any given grid point. This has the effect of influencing the thickness field near the ground so that the hydrostatically derived temperature field closely resembles a direct surface temperature analysis.

The final process in the preparation of the initialization file is the adjustment of the vertical temperature profile. Since height rather than temperature is analyzed by the 3D OI scheme, temperature profiles will sometimes include regions of superadiabatic lapse rates or temperature spikes. This is particularly true near the surface where closely spaced analysis layers mean that a small error in the height analysis at one level can produce large temperature anomalies in the layers above and below. Spikes are removed, one at a time, starting with the most significant spike. The spike is removed by adjusting the height of the layer interface lying between the grid point that contains the spike and the adjacent grid point that most resembles a spike of the opposite sign. This method is based on the assumption that a small error in the height analysis at one layer interface relative to the interfaces above and below will introduce a warm-cold dipole in the temperature field at the layers immediately above and below. Once all significant spikes are removed, superadiabatic layers are removed in a similar manner. Often, the removal of temperature spikes also removes superadiabatic layers so that this final step is not needed. The adjustment of the intermediate height is done so that temperature lapse rate between the grid points above and below is set to the mean lapse rate of a layer extending one level above and below any contiguous layers containing a spike or superadiabatic lapse rate.

#### *f. Model boundary conditions*

In cloud-scale models, in which the timescale is small enough to assume that the environment outside the model domain is unchanged during the simulation, inflow boundaries are usually prescribed from environmental conditions. In mesoscale and larger-scale models this assumption can no longer be made. The environment outside the computational domain has to be derived from larger-scale forecast tools such as the Nested Grid

Model, the Eta Model, or the Medium Range Forecast model run at NCEP, or the Navy Operational Global Atmospheric Prediction System (NOGAPS) run at the Navy Fleet Numerical Meteorology and Oceanography Center. Boundary conditions at user-specified time intervals are calculated based on one of these forecasts, and linear interpolation is used to determine boundary values at intermediate times.

The lateral boundaries are open and should allow the unimpeded flow of air. At outflow boundaries interior values rather than forecast data are used in the calculation of the fluxes across the boundary faces. To allow the propagation of acoustic disturbances across the boundaries, we use a radiative boundary condition with a uniform phase speed on outflow boundaries. At the end of each time step, a “nudging” scheme is applied to the new values of all variables, which nudges values in cells immediately inside the boundary to the forecast values in the ghost cells across the boundary. At inflow boundaries forecast data are used to calculate both the velocity normal to each boundary face and the fluxes across the face.

At the top of the computational domain, initial values of density and potential temperature are calculated assuming a hydrostatic balance, and these values remain fixed throughout the simulation. A homogeneous Neumann boundary condition is applied to the horizontal components of momentum, while the vertical momentum is assumed to be zero. Thus the upper boundary is a rigid, free-slip surface. To minimize reflection off of this boundary, a diffusive “sponge” is applied to the vertical component of the momentum. The coefficient of the sponge is exponential and, thus, approaches zero rapidly below the top few layers of the grid.

The surface boundary conditions applied in the OMEGA model are formulated with the aid of Monin–Obukhov similarity theory. This theory allows one to determine the values of the model’s prognostic variables at a diagnostic level between the first and second computational levels.

At the ground surface, a no-slip condition is imposed. The hydrostatic equation is used to obtain the surface pressure from level 2 (the lowest atmospheric layer in the model), while the surface density,  $\rho$ , is computed from the known surface pressure and ground surface temperature using the ideal gas law. For all Eulerian tracers ( $Q_i$ ) and for the turbulent kinetic energy, a zero flux boundary condition is assumed at the ground.

### 3. Atmospheric Dispersion Model

Atmospheric dispersion over mesoscale travel times and distances involves virtually every atmospheric dynamical process operating on scales larger than molecular dissipation and smaller than the latitudinal variation of the Coriolis.

For example, thermal and mechanical forcing due to mesoscale terrain inhomogeneities can generate exter-

nally forced mesoscale flow inhomogeneities such as upslope precipitation events, lee cyclogenesis, drylines, land and sea breezes, mountain-valley winds, and urban heat island circulations. Moist processes can generate mesoscale convective systems such as cumulus-scale convection, extratropical squall lines, mesoscale convective complexes, mesoscale cellular convection, tropical cyclones, mesoscale rainbands, tornadoes, and tornadogenesis. The atmospheric inertial mode can generate internally generated circulations such as fronts and jet streaks, buoyant instability, synoptic instability, Kelvin–Helmholtz instability waves, quasi-stationary convective events, and mesoscale structure of hurricanes. This wide range of mesoscale circulations and their associated mesoscale vertical ascents and descents can have a significant impact on the dispersion of pollutants.

In addition, the presence of wind shear, mainly caused by surface friction and large-scale baroclinicity, further complicates the flow dynamics and the dispersion of pollutants over mesoscale travel times and distances by affecting the local structure and dynamics of the planetary boundary layer. Many flow instabilities occur partly as a result of the presence of wind shear, which may enhance turbulent diffusion by increasing turbulence intensities. Also, synoptic-scale wave–wave interactions may produce higher wavenumber circulations or flow features. Latitude affects both the inertial period and diurnal forcing. Time of day affects the dynamics of the boundary layer, while time of year affects the diurnal forcing.

In summary, mesoscale energy is distributed over a variety of flow modes, including mean mesoscale circulations, mesoscale eddies, and internal gravity waves. Flow dynamics are also quite variable and complex on this scale. Mesoscale flows may be hydrostatic or nonhydrostatic. Nonhydrostatic motions may contain significant features on scales ranging from several meters to several tens of kilometers with timescales of minutes to many hours. Hydrostatic motions, in which the nonhydrostatic motions are embedded, have motion scales with orders of magnitude larger than the nonhydrostatic motions. Therefore, mesoscale atmospheric flows possess various mesoscale time and space scales. Even for horizontally homogeneous flows over flat and uniform terrain, mesoscale frequencies such as the diurnal heating cycle and formation of a nocturnal low-level jet will usually be present. The transport and diffusion of atmospheric pollutants can be affected by this wide range of flow scales through variations in the mean transport wind, differential advection due to vertical and horizontal wind shear, and vertical mixing (cf. Moran 1992). Therefore, these mesoscale space–time flow scales should be represented accurately when the mesoscale dispersion processes are studied. Otherwise, significant components of mesoscale transport and diffusion may be neglected.

ADM is embedded within the OMEGA model. There-

fore, the ADM model has the benefit of using the OMEGA model-predicted meteorological fields (such as mean wind components and atmospheric turbulence information) at the OMEGA model's spatial and temporal resolution. The ADM model is composed of an Eulerian model, a Lagrangian particle dispersion model, and a probabilistic Lagrangian puff model. Since each modeling technique has its own advantages and disadvantages, we have decided to implement the above three dispersion models in the OMEGA model. In principal, these dispersion models advect massless tracers and gases and/or solid and liquid particles using the OMEGA-resolved wind field, while simultaneously solving a diffusion model that simulates the effect of unresolved physics including turbulence. Massless tracers and gases are in thermodynamic and dynamic equilibrium with the ambient atmosphere whereas solid and liquid particles are allowed to settle out with a size- and density-dependent vertical slip velocity due to gravity (Bacon and Sarma 1991):

$$V_t = \left( \frac{30\mu}{\rho_a D} \right) \left( -1 + \sqrt{1 + \frac{D^3 g \rho_a \rho_p}{405\mu^2}} \right) \quad (32)$$

where  $\mu$  is the dynamic viscosity of air,  $\rho_a$  is the ambient air density,  $\rho_p$  is the particle density,  $D$  is the particle diameter, and  $g$  is the acceleration due to gravity. The features of these dispersion models and their basic advantages and disadvantages briefly will be discussed in the following sections.

#### a. Eulerian dispersion model

The Eulerian dispersion model embedded within the OMEGA model assumes that the released gas instantaneously and uniformly fills the OMEGA cell containing the source. The Eulerian tracers are also currently assumed to be massless and hence are advected by the mean wind and diffused by its turbulence components. The conservation equation for the tracer field can then be written as

$$\frac{\partial \chi}{\partial t} + \nabla \cdot (\chi \mathbf{V}) = F_\chi + S_\chi, \quad (33)$$

where  $\chi$  is the tracer concentration,  $F_\chi$  represents sub-grid-scale turbulent transport, and  $S_\chi$  represents the tracer sources/sinks. Sources and sinks ( $S_\chi$ ) are specified via their source strength, latitude, longitude, and altitude locations.

In general, Eulerian models are well suited to handle complicated physical processes such as reactive nonlinear chemistry and wet deposition. These models are also well suited to handle a large number of geographically fixed sources, since their gridpoint calculations are independent of the number of sources. However, Eulerian models suffer from several limitations. First, the treatment of advection in Eulerian models usually introduces artificial numerical diffusion and sometimes spurious

oscillatory behavior, a problem when advecting non-negative physical quantities such as concentrations. Second, use of the gradient-transfer hypothesis limits these models to timescales much larger than the turbulence integral timescale and to pollutant spatial scales much larger than the turbulence integral spatial scales. Third, the pollutant mass being modeled must have a spatial extent at least equal to four or more horizontal and vertical grid increments in order for the gradients to be adequately defined and the advection phase errors minimized. Finally, these characteristics make point and line sources particularly difficult to treat in Eulerian models.

#### b. Lagrangian particle dispersion model

The Lagrangian particle dispersion model implemented in the OMEGA model applies the Monte Carlo technique to simulate dispersion of nonbuoyant and dynamically passive pollutants released from multiple emission sources with different geometry and release duration. The model can be used for a computer visualization of the OMEGA flow field in complex terrain with the aid of particle distributions and trajectories. The Lagrangian particle dispersion model basically simulates the dispersion of pollutants by tracking a large set of Lagrangian particles moving, at each time step, with pseudovelocities. These pseudovelocities simulate the effects of the two basic dispersion components: 1) transport due to the mean wind, and 2) diffusion due to turbulent velocity fluctuations (Boyd et al. 1995). In addition, particle velocity is adjusted so that particles are allowed to fall out with a vertical slip velocity due to gravity. Subsequent positions of each particle ( $x$ ,  $y$ ,  $z$ ) representing a discrete element of pollutant mass are computed from the following:

$$x(t + \Delta t) = x(t) + [\bar{u}(t) + u'(t)]\Delta t, \quad (34)$$

$$y(t + \Delta t) = y(t) + [\bar{v}(t) + v'(t)]\Delta t, \quad \text{and} \quad (35)$$

$$z(t + \Delta t) = z(t) + [\bar{w}(t) + w'(t)]\Delta t, \quad (36)$$

where  $\bar{u}$ ,  $\bar{v}$ , and  $\bar{w}$  are the resolvable-scale components of wind velocity obtained directly from the OMEGA model, and  $u'$ ,  $v'$ , and  $w'$  are the corresponding subgrid-scale turbulent velocity fluctuations whose statistics are derived from the boundary layer formulations used in the OMEGA model.

An important advantage of turbulent kinetic energy closure implemented in the OMEGA model is that it allows one to determine a variety of turbulent variables including the Lagrangian turbulent statistics. The Lagrangian turbulent statistics necessary to run the Lagrangian particle dispersion model are obtained from the level 2.5 Mellor and Yamada scheme (cf. Uliasz 1990).

Lagrangian particle dispersion models have a number of advantages over other simulation techniques. First, Lagrangian particle dispersion models combine all uncertainties into the correct determination of the pseu-

dovelocities. Second, a Lagrangian framework is more natural for modeling turbulent diffusion, which is a Lagrangian phenomenon. Third, Lagrangian particle models do not suffer from computational phase dispersion because no advection terms need to be considered in the mass conservation equations. Fourth, diffusion over small space and time scales, including emissions from multiple-point pollutant sources, can be treated easily by these models. Fifth, Lagrangian particle models are usually able to incorporate more turbulence properties than other simulation techniques due to their ability to incorporate random turbulent components. Finally, Lagrangian particle dispersion models are flexible, conceptually simple, and computationally inexpensive when a reasonable number of particles are used. In principle, Lagrangian particle dispersion models can provide a degree of resolution and accuracy to complex flow solutions not obtainable by other simulation techniques, if the mean flow, Lagrangian timescales, and turbulent statistics are provided. However, they are not well suited for modeling the dispersion of nonlinear reacting pollutants and the question of statistical sampling error must be addressed when estimating pollutant concentrations in areas of low particle density, sometimes requiring the release of a very large number of particles.

### c. Probabilistic puff model

The simpler version of the Lagrangian particle dispersion model allows us to track the motion of puffs with a Gaussian distribution of pollutant mass. Continuous plumes are commonly represented by subdividing them into discrete, overlapping puffs. Puff models are conceptually simple and this simplicity usually translates into relatively straightforward solutions to the mathematical and computer programming requirements.

The probabilistic puff model implemented in the OMEGA model generates a new group of Lagrangian puffs ( $N$ ) at a user's defined time interval ( $\Delta t$ ). The number of puffs released depends on local wind speed and turbulent diffusion at the source. Here  $N$  is chosen in such a way that the distance between two puffs should be approximately equal to half of the horizontal standard deviation,  $\sigma_h$ . Each puff generated represents a discrete element of the total material released into the atmosphere.

The puffs are moved using the OMEGA model mean wind field only. Locations of each puff center are described by Eqs. (34)–(36) where turbulent components are omitted. Puff diffusion, the distribution of material within each puff (around the puff centroid) in space according to a Gaussian function whose  $\sigma_x$ ,  $\sigma_y$ , and  $\sigma_z$  values increase with travel time or distance, is estimated based on Taylor's homogeneous diffusion theory. It is assumed that the theory is applicable over short time-scales (cf. Uliasz 1990 or Arya, 1999), such as the time step,  $\Delta t$ , used to transport puffs in OMEGA:

$$\begin{aligned}\sigma_x(t + \Delta t) &= \sigma_x(t) + \sigma_u \Delta t & \text{for } t \leq 2T_L^u \text{ and} \\ \sigma_x^2(t + \Delta t) &= \sigma_x^2(t) + 2T_L^u \sigma_u^2 \Delta t & \text{for } t > 2T_L^u,\end{aligned}\quad (37)$$

where  $\sigma_u$  is the variance of wind velocity component and  $T_L^u$  is the Lagrangian integral timescale. These turbulence characteristics are obtained from the turbulence closure scheme used in the OMEGA model. In a similar way  $\sigma_y$  and  $\sigma_z$  are determined (Uliasz 1990).

As the puff grows, the local representation of the turbulence and velocity fields using puff centroid location becomes increasingly inaccurate. When the meteorological fields are inhomogeneous, the accuracy of the calculation can only be maintained by splitting puffs into smaller components that can sample the variations in the meteorology explicitly. A puff splitting scheme is implemented in such a way that when the growth of a puff extends either vertically or horizontally over one cell area of the OMEGA model grid, then the puff is divided into two or more puffs such that the sum of their masses equals the mass of the initial puff.

The key element to the practical application is the elimination of duplicate puffs. The generation of new puffs, especially due to vertical diffusion, can quickly overwhelm even the fastest and largest computers. For this reason, overlapping puffs with Gaussian concentration distributions are merged when they are separated by about  $\sigma_h$  or less. This reduces the number of puffs used in the probabilistic puff model. Every 10 min all adjacent puffs are checked to see if the puff centroids are within  $\sigma_h$  of each other. If so, the material from both is merged into a single puff. The centroid of the merged puff is assigned new coordinates and  $\sigma_h$  values that are averages of those in the original puffs. Puffs are also removed from further consideration when the puff centroid point is outside of the modeling domain. This approach should cause no problems within the area of interest, except when a wind reversal brings puffs back into the modeling domain.

The concentration within each puff is distributed in space according to a Gaussian function whose  $\sigma_x$ ,  $\sigma_y$ , and  $\sigma_z$  values increase with travel time or distance. Knowledge of puff location, the material in it, and its geometrical dimensions allows concentration to be determined anywhere by summing contributions from each puff. Local concentration values are receptor oriented and based on summing contributions from each puff using generalized Gaussian distribution (Sykes et al. 1986).

Observations have shown that concentration fluctuations are sometimes at least as large as the mean. Therefore, there is a growing need for methods of predicting fluctuations of concentration. For example, some accidentally released gases are toxic and their flammability is determined by short-term concentration levels [e.g., the accidental release of MIC gas at Bhopal (Boybeyi et al. 1995)]. In this emergency case, the prediction of concentration fluctuations is crucial. On the other hand,

the magnitude of turbulent fluctuations of concentration determines the uncertainty in air quality models. Uncertainty calculation is also important in most of emergency hazard predictions. Wilson et al. (1982a,b) were concerned with simulating concentration fluctuations from a continuous source observed in a wind tunnel and proposed a simplified empirical model based on the standard Gaussian formula. This formulation for the concentration fluctuation calculation is used in the OMEGA probabilistic puff model.

Puff models are conceptually simple, and computationally inexpensive. These models are also used widely in atmospheric dispersion studies. However, puff models usually ignore entrainment, cloud venting, and other mixing effects with the ambient environment, and they cannot accommodate nonlinear chemistry. Finally, treatment of multiple pollutant sources in puff models may quickly become very complicated.

#### 4. The OMEGA grid adaptivity

Since the accurate solution of any complex computational problem depends on fine spatial discretization of the computational domain, the accurate representation of multiscale events in numerical models has long been a principal issue in computational fluid dynamics. For example, one typically desires to capture not only the development and evolution of small-scale features but also their interaction with and influence upon the larger-scale flow. This is a particularly important requirement in atmospheric models, because numerous events such as fronts, clouds, and plumes are not only relatively localized with respect to their environment, but are also forced on scales larger than their own. Because practical limitations in computer size and speed prohibit the use of uniformly high spatial resolution appropriate for the smallest scales of interest, numerous techniques have been developed to deal with multiscale flows.

Grid nesting techniques involve the sequential placement of multiple finer-scale meshes in desired regions of the domain so as to provide increased spatial resolution locally. Although the decision to spawn one or more submeshes is typically subjective and manually directed, many formulations allow the submeshes to move with particular features in the flow, such as hurricanes (Jones 1977). A principal limitation of the grid nesting technique is that one must know a priori, and for the duration of the calculation, which regions of the domain will require high spatial resolution. In other words, the trajectory of the moving grid has to be predefined and therefore cannot be used for prediction.

Another principal limitation of grid nesting technique is the interaction among multiple nested meshes, particularly the tendency for propagating dispersive waves to discontinuously change their speeds upon passing from one mesh to the next and to reflect off the boundaries of each nest due to an impedance mismatch across

the mesh boundaries (Zhang et al. 1986). Without the sharp internal boundaries of a nested grid, this numerical difficulty is missing in unstructured grids.

Grid refinement techniques are a relatively new and powerful class of methods and can be subdivided into two basic categories. The first includes methods in which grid points are added locally to the computational domain as the calculation proceeds, or finite elements are subdivided locally, to provide increased spatial resolution based on predetermined physical criteria. This grid refinement technique has been widely used in aerospace applications with successful results. For example, Lottati and Eidelman (1994), Baum and Löhner (1994), and Baum et al. (1993) have applied this dynamic grid generation for simulating flows over aircrafts, tracking shock waves, and in simulating fluid-structure interactions. The second category of refinement technique involves methods that redistribute, in some predetermined manner, a fixed number of grid points so as to provide locally increased resolution and thus an improved solution in certain regions of the domain. For example, Dietachmayer and Droegemeier (1992) and Fiedler et al. (1998) have used this method for atmospheric flows using structured grids. However, their methodologies have been restricted to idealized problems so far and are not suitable for real atmospheric flow simulations, which can include complicated terrain features. Structured grids also are not suitable for dynamic grid adaptation, because the grid generation requires a high degree of user interaction and user expertise. It is not realistic to use manual grid-generation code for real-time predictions.

The unstructured grid technique is rather new to the atmospheric science community. In many fields of engineering applications, the unstructured grid method has been in use for more than a decade due to its efficiency in the modeling of irregular domains (cf. the above references). The flexibility of unstructured grids and their ability to adapt to transient physical phenomena are the features that give unstructured grid algorithms for partial differential equations their great power. Grid adaptivity improves the fidelity of all finite-difference or finite-element numerical schemes, by increasing resolution in regions of high gradient. The improvement comes from the ability to adapt the grid structure to the flow, and from the local refinement of the grid in the vicinity of rapidly changing horizontal spatial structures in the atmosphere.

Using unstructured grids also eliminates the disadvantages of nested grid technique. The main advantage of unstructured grids is the ease with which dynamic solution adaptation can be implemented. There is no longer a need for involved user expertise/interaction for creating topologies of complicated terrain features; the whole procedure can be fully automated. Automation of the process is not only highly desirable, but can also be required in operational settings. Also, since the unstructured grid is a single mesh with a smooth and con-



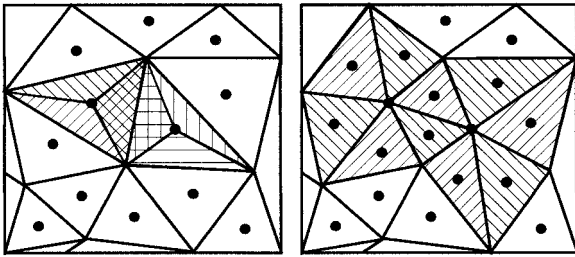


FIG. 9. Vertex addition and reconnection.

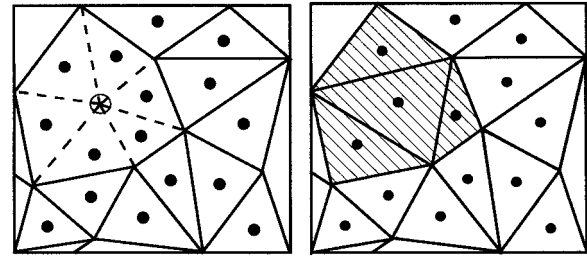


FIG. 10. Vertex deletion and reconnection.

tinuous transition from coarse to fine regions within the whole domain, the model is naturally two-way scale interactive without the interpolation error caused by the transfer of information from one nest to another.

To the best of our knowledge, OMEGA is the only operational atmospheric flow model based on an unstructured grid technique, which fully exploits the advantages and flexibility of unstructured grids. It can adapt its grid both statically and dynamically to different criteria. For real-time flow predictions, the capability of grid adaptivity, given the CPU constraint, becomes important. This capability is also crucial in responding to emergency scenarios such as release of hazardous materials. OMEGA with its grid adaptation capability has a unique advantage over other atmospheric flow models in providing accurate solutions quickly in an operational setting. The grid adaptivity in the OMEGA model takes place in two different ways: 1) static grid adaptivity, and 2) dynamic grid adaptivity.

#### a. Static grid adaptivity

The total number of grid points necessary to perform a successful numerical computation that recovers the correct physics can be greatly reduced in an unstructured grid. By this we mean that the recovery of the model physics at the smallest length scale resolved does not require the complete domain to have the same resolution. The resources of the numerical and computational machinery are focused on the regions of importance. This is especially significant for three-dimensional hydrodynamic problems, where our experience has shown the resulting economy can make the difference between tractability and intractability (cf., Baum et al. 1993).

In OMEGA, the adaptation of the unstructured grid takes place through a variety of grid operations (Figs. 9–11). The first is vertex addition, which is followed by vertex reconnection. Figure 9 illustrates these two steps when some activity that would indicate a need for more resolution is noted in two cells. The vertex addition step is accomplished by adding a vertex at the centroid of each affected cell and connecting it to the vertices of the cell. The reconnection step then involves the evaluation of each new cell to see if it is possible to create grid cells with lower aspect ratios by removing an edge and reconnecting the vertices.

Figure 10 shows the reverse process, in which the grid is coarsened through the process of vertex deletion. Like vertex addition, vertex deletion is followed by a vertex reconnection step. It is important to note that the goal of grid adaptation is not to *move* the grid, but rather to refine the grid in advance of any important physical process that might require additional grid resolution, and to coarsen the grid behind the region. This differentiates the method from the adaptation techniques described by Dietachmeyer and Drogemeier (1992), which used vertex movement to adapt to atmospheric features.

A different type of grid modification is also shown in Fig. 11. In this figure we show vertex relaxation, in which the vertices are allowed to move as a mass-spring system, and edge bifurcation, which is equivalent to vertex addition in the special case of an edge cell. Vertex relaxation is used to keep the aspect ratio as near unity as possible to minimize numerical errors in computing gradients.

The OMEGA grid in the “static” adaptation case is adapted a priori to resolve static features such as terrain gradients, land–water boundaries, and/or any other feature that the user includes in the adaptation scheme. The grid does not change during the course of the OMEGA run. The OMEGA grid automatically adapts to static features of the underlying topography such as complex terrain and land–water boundaries.

An example of the flexibility of the OMEGA grid is shown in Fig. 12. This figure shows a grid generated for the southeastern United States and Mexico. The grid was adapted to the underlying topography, the land–water boundaries, and to the initial weather conditions. The synoptic situation chosen was Hurricane Linda at 0000UTC on 10 September 1997. In this example, we

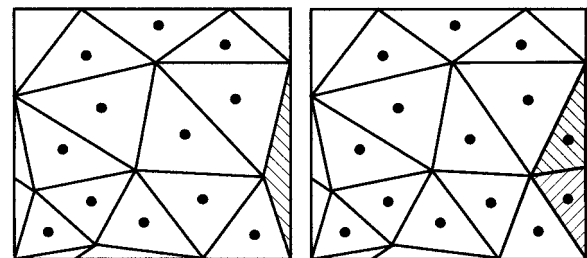


FIG. 11. Vertex relaxation and edge bifurcation.

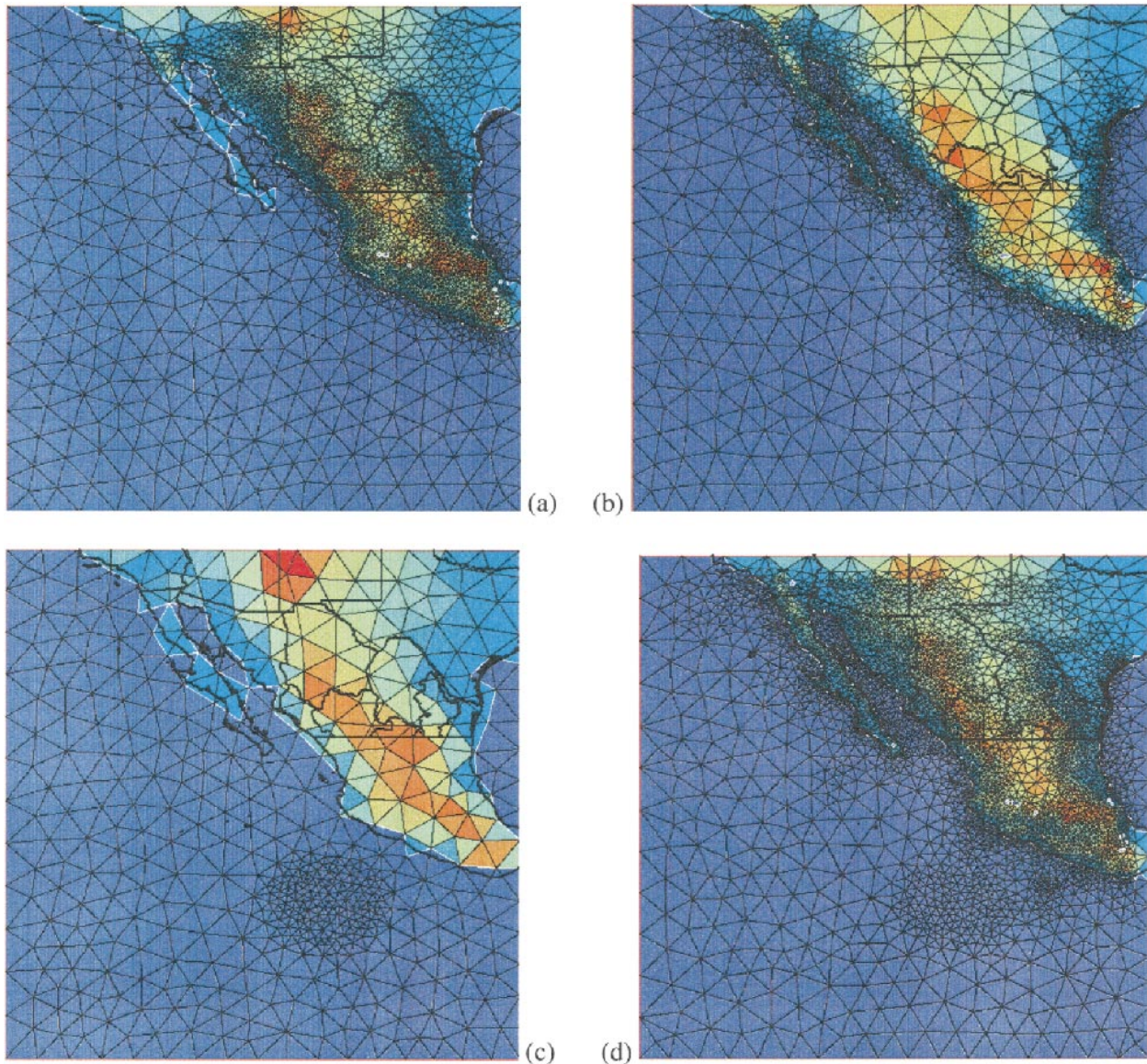


FIG. 12. The OMEGA grid can adapt to (a) the topography, (b) the land–water boundary, and (c) the weather existing at the time of model initialization (Hurricane Linda). The resulting grid is seen in (d).

have broken the grid-generation process into different steps for illustration. The first grid shown was generated by adapting to gradients in elevation (refining the grid in mountainous areas), the second to gradients in the land–water index (refining the grid in coastal areas), the third to the hurricane, and the fourth to all of these criteria. The final surface grid consists of approximately 7300 triangles with edges ranging from roughly 15 to 300 km.

In addition to the automatic static adaptation to geographical features, the grid can be refined in one or more specified geographical areas, such as theaters of operation, by the creation of up to 99 rectangular subdomains in which higher resolutions are specified. Within each subdomain, grid generation is governed by the user-

specified minimum and maximum resolutions. Subdomains can overlap to create a high-resolution region of almost any shape, or to produce a high-resolution area within an intermediate-resolution region. In addition to subdomains, the user can specify a location on the grid and a radius of influence around it; the grid generator will then refine the grid within the region of influence. A smoothing process is also performed on the resolution specifications to avoid high expansion ratios at the boundaries of subdomains. A very important point, however, must be made: the result of the use of subdomains and/or circular region refinement is a *single*, not nested, grid with variable resolution.

An important feature of the use of an unstructured grid is the ability to simulate mountains and coastal

features without the “stair-step” geometry required by nested grid models (cf., Zhang et al. 1999). Triangular grids can naturally follow the coastline better, leading to improved land–water circulations, and can better capture the geometry of mountainous regions. This is especially important for near-surface simulations.

#### *b. Dynamic grid adaptation*

The grid adaptation described so far is a “static” adaptation; adaptation is implemented in the grid generator prior to the beginning of an OMEGA run, and the grid does not change during the simulation. OMEGA also has the ability to adapt its grid during a simulation to different criteria such as frontal activity, convection, hurricanes, and/or a pollutant plume. This enables atmospheric features that require additional grid points for adequate simulation to be resolved as they appear. Thus through the combination of adaptation methods (Figs. 9–11) and criteria (Figs. 12a–c), the grid can be coarse where the circulation is regular and smooth, but greatly refined where there are sharp gradients, where there are discontinuities in the flow, where topographic features are important, or where model physics or dispersion source terms require fine resolution. The underlying philosophy of dynamic adaptation is to provide a refined grid to resolve important physical events and features as the simulation is in progress.

The dynamic grid adaptation in OMEGA consists of four major steps: 1) at a predetermined time step specific variables or their gradients are evaluated to see if they meet the adaptivity criteria, 2) the mesh is refined where these criteria are satisfied, 3) the physical variables are interpolated to new cell centers, and finally 4) the mesh is coarsened where the criteria are not met. For example, if there are sharp discontinuities such as propagating fronts the mesh is refined ahead of the front so that a refined mesh already exists when the front arrives at a specific location. As the front propagates, the refined mesh at the previous location of the front is coarsened to the resolution specific to the background mesh. This type of coarsening and refining is performed at predetermined time step increments.

Physical variables, which are cell centered, are interpolated to vertices before each adaptation cycle using the pseudo-Laplacian weighted averaging scheme. As new cells are created or removed locally a simple linear interpolation is done to assign values to new vertices and cell centers. All the diagnostic variables are calculated for new cells, the particle location array is updated, underlying terrain is updated, and the mesh is remasked to calculate new time steps for the time-splitting scheme. The original base-state variables are also saved, and as new cells are created, base-state profiles are generated for these cells using the initial base state. At the beginning of a simulation the minimum elevation is saved. As the grid is refined or coarsened, no cell is allowed to have elevation lower than the initial mini-

mum elevation value. This is to ensure that no extrapolation is done in the calculation of the base state. The whole procedure requires only a few user inputs at the beginning of the simulation and after the initial inputs, it is completely automated. After the grid has been modified all the physical variables are interpolated to the new mesh and the diagnostic variables are calculated.

Setting the right criteria for adaptation is very important. There is a significant cost associated with a grid adaptation; hence, the ideal criteria are those that require minimum computational effort to evaluate yet indicate key regions requiring additional resolution. If the adaptation criteria are particle locations representing a pollutant plume or puff, then it becomes a simple step of locating the cells containing particles, and then tagging those cells for refinement. This method can become costly if there are multiple release sources and the frequency of particle releases is very high. The alternative method is adapting to the flow itself. This is not as straightforward as tagging the cells for particle locations. Atmospheric flows are almost always turbulent and highly stratified in the vertical. The gradients are often weak and hard to detect. One advantage in simulating atmospheric flows, however, is the large temporal scale of the simulation (usually 12–48 h). Therefore, even if an event is missed at first, the chances of capturing it at a later time are higher.

The OMEGA grid dynamically adapts to any criteria specified by the user. For example, OMEGA can provide high-fidelity solutions in the regions of particle plumes by refining the grid around the particle locations (Fig. 13) and coarsening the grid where high resolution is not required (e.g., areas from which the plume has already passed). The criteria for adaptation is set by defining a Gaussian function around each particle. The region close to particles therefore gets higher weighting and is tagged for refinement; areas devoid of particles and consisting of relatively flat terrain are tagged for coarsening.

Dynamic adaptation was demonstrated for a simple flow, but including the effects of sulfate chemistry in Sarma et al. (1999). This case demonstrated how the use of dynamic adaptation could capture 85% of the total sulfate production computed by a uniform high-resolution simulation, in 20% of the computational time. For our comparison here, we present the results from two pairs of simulations: 1) a high-resolution static simulation compared against an initially coarse-resolution dynamically adapting simulation and 2) a coarse-resolution static simulation compared against a dynamically adapting simulation. The former demonstrates that the dynamically adapting solution does, in fact, match that of a static solution using high resolution at all times while consuming considerably fewer computational cycles. The latter demonstrates that dynamic adaptation can improve the quality of a simulation by resolving terrain features that might otherwise be missed due to the use of static resolution.

The first pair of simulations, illustrating the advantage

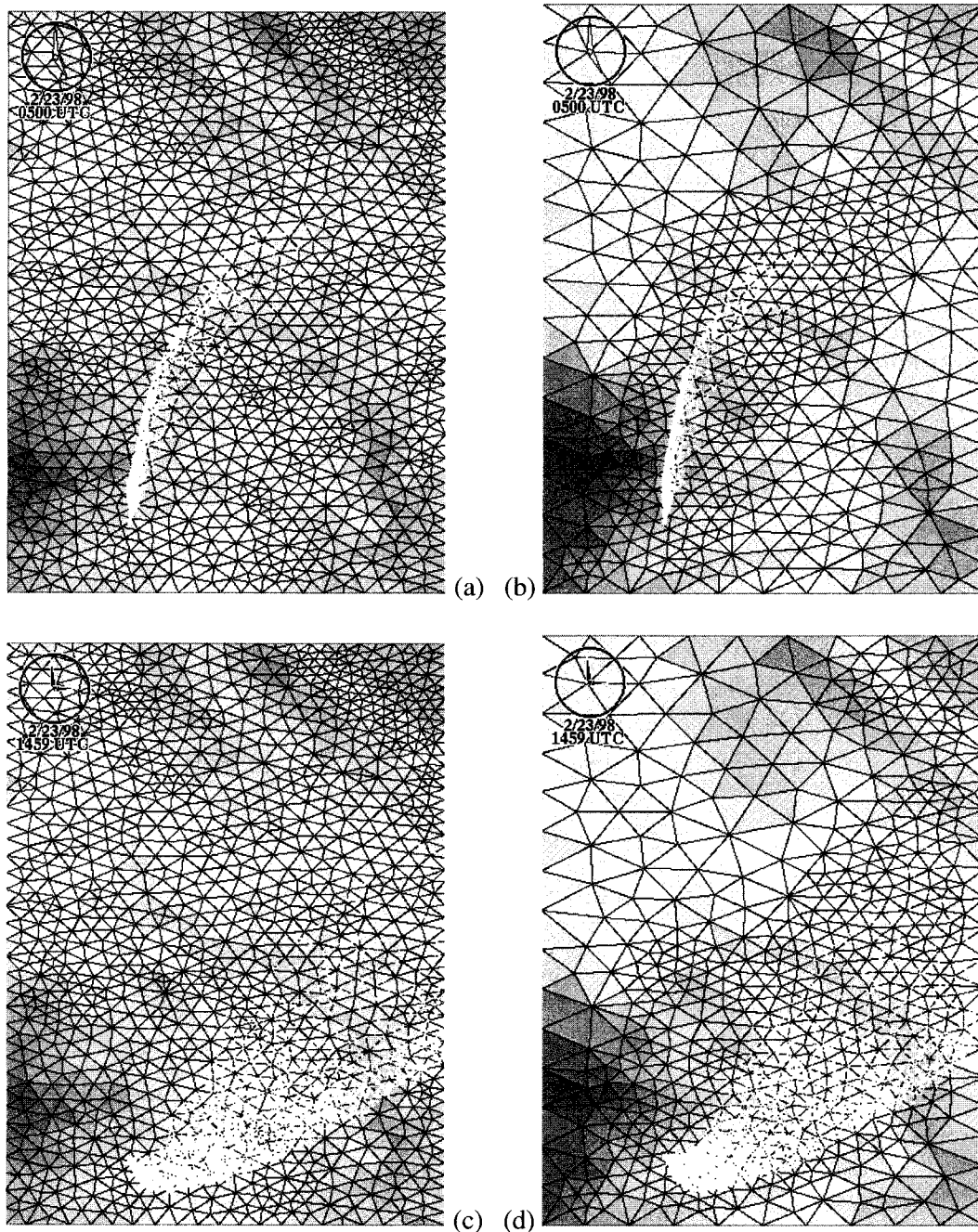


FIG. 13. (a) Global refinement case after 5 h of simulation, (b) dynamic adaptation case after 5 h, (c) global refinement case after 15 h, and (d) dynamic adaptation case after 15 h.

of dynamic grid adaptation over global refinement, is shown in Fig. 13. The first case (global refinement case) had a domain of roughly  $300 \text{ km} \times 400 \text{ km}$  covering parts of Nebraska, South Dakota, Iowa, and Minnesota. A variable grid resolution of 5 km to 15 km was specified. The grid had roughly 2300 cells in the horizontal layer with 30 vertical levels. In the vertical, a stretched grid was used and the first level near earth's surface had a thickness of 15 m. A particle source was defined in

the southwest corner of the domain. The simulation was initialized using the U.S. Navy's NOGAPS gridded data. The simulation was run for 15 h.

The second case (dynamic adaptation case) used identical parameters to the global refinement simulation except for the initial grid size. The starting grid had roughly 600 cells in the horizontal (not shown). In this case a variable resolution of 10–36 km was prescribed in the initial grid. The adaptation criteria was set to particle

locations. A refinement cycle was invoked after every hour of simulation time. Cells were tagged by defining a Gaussian function around the particles. Tagged cells were refined and all physical fields were interpolated at each adaptation cycle. A coarsening cycle was invoked after every 3 h to remove triangles from areas where higher resolution was not needed. This was done by identifying the cells that did not contain particles and were in the relatively flat regions and then deleting them. The final grid after 15 h of simulation had roughly 1100 cells and the resolution varied from 6 to 37 km. This simulation required less than 60% of the CPU time yet produced nearly identical results compared to global refinement. The simulation required only the initial user inputs, after which the whole process was seamless and automated.

In each adaptation cycle, OMEGA updates the underlying terrain. Thus, the refinement cycle can resolve finer topographic features, improving the solution. The second pair of simulations demonstrates this fact (Fig. 14). The simulations were conducted in the Four Corners region of the United States, which is characterized by complex terrain features. The domain and the initial conditions for both simulations were identical. The simulation domain was 300 km  $\times$  300 km in size. The initial grid had roughly 450 cells with a varying resolution from 7 to 42 km. In the first simulation, the grid was not refined during the simulation. Figure 14a shows the particle locations 12 h into the simulation. In the second simulation, the grid was adapted dynamically during the simulation. The adaptation criteria was set to particle locations. Figure 14b shows the particle locations for the second simulation after 12 h. The final grid had roughly 900 cells in a horizontal layer. As can be seen from the figure, by resolving finer terrain features as the grid dynamically adapted to the evolving plume, a different and presumably more accurate representation of the flow field (since it will be affected by the terrain) and hence plume trajectory were achieved. This figure schematically shows that fine spatial discretization of the computational domain may result in a more accurate solution of any complex computational problem.

### 5. Operational considerations

The ultimate consideration of the OMEGA design has always been worldwide operational utility. The worldwide real-time operational requirement influenced a number of design decisions. The OMEGA model has been designed as a delicate balance between physical fidelity, numerical accuracy, and operational constraints.

First of all, the worldwide operational requirement forced us to develop a number of worldwide databases including 1-km resolution terrain elevation, standard deviation of terrain elevation (an indicator of surface roughness), land–water boundary information, land use, vegetation, coarser 5 arc-minute databases of the ele-

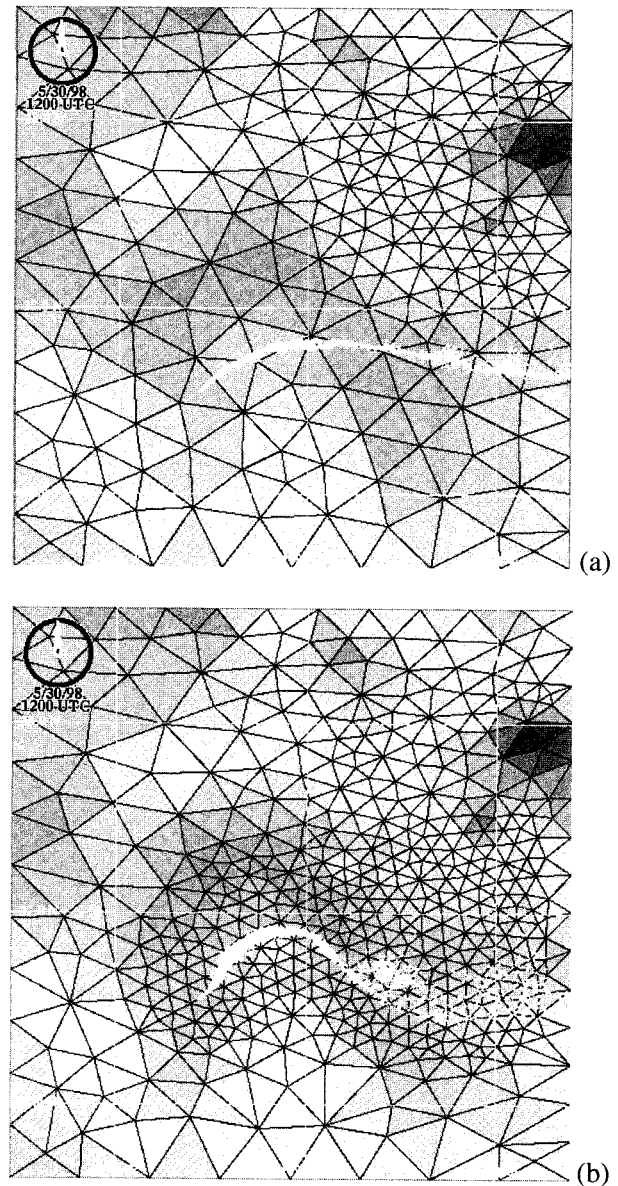


FIG. 14. Simulation results after 12 h of simulation for (a) the static case and (b) the dynamic adaptation case.

vation and land–water data, and datasets for soil texture, soil moisture, soil temperature, and sea surface temperature at various resolutions.

The design point of 1 km for the highest grid resolution was also based upon a recognition that although worldwide terrain elevation and land–water boundary information is available at even higher resolution, worldwide albedo, land use, vegetation, soil texture, and soil moisture, as well as the worldwide meteorological data observations, are rarely available at even 1-km resolution. This determined one level of requirements for the physics that had to be contained in OMEGA. Worldwide utility required that any physical models that were incorporated into OMEGA would have to have world-

TABLE 2. Datasets used to determine surface characteristics for the OMEGA model.

Surface property	U.S. data	Global data
Terrain elevation land–water	Global elevation and land–water datasets, 5' × 5' and	
	80°–90° Latitude	30" × 180"
	75°–80°	30" × 120"
	70°–75°	20" × 90"
	50°–70°	30" × 60"
	0°–50°	30" × 30"
Soil type	Webb global soil texture class from <i>GED</i> CD-ROM, 1° resolution	
Climatological vegetation index	USGS NDVI annual datasets, 2" resolution	Monthly GVI from <i>GED</i> CD-ROM, 10' resolution, USGS NDVI biweekly datasets, 30 arc-second resolution
Land use/ land cover	USGS BATS Land Cover, 2 minute resolution	Olson World Ecosystems BATS, land cover from <i>GED</i> CD-ROM, 30' resolution, USGS 30 arc-seconds datasets
Climatological sea surface temperature	Biweekly USGS SST climatology, 12' resolution	
Climatological subsurface temperature	Average monthly air temperature from <i>GED</i> CD-ROM, 30' resolution	
Climatological soil moisture	Budyko soil moisture budget using average monthly temperature and precipitation from <i>GED</i> CD-ROM, 30' resolution	

wide applicability and could not rely on datasets that were not available with this coverage. This, for example, eliminated the use of a second-order closure turbulence model since the Reynolds stress is not operationally available for either initial or boundary conditions.

As part of the real-time operational requirement, a great deal of automation was also desirable in the OMEGA system. This resulted in the creation of a highly automated grid generator, an automated meteorological and surface data assimilation system, and a user-friendly

X-windows- and Motif-based Graphical User Interface (GUI) and graphic postprocessors.

With this as a basis, the design objective of OMEGA was a tool for real-time hazardous prediction capable of performing a fully coupled atmospheric forecast and aerosol or gas dispersion calculation at 10 times real time. As will be seen, the tremendous flexibility and portability of the OMEGA modeling system permits the model to meet these constraints on platforms ranging from CRAY supercomputers to IBM R/S 6000 workstations. A built-in runtime estimator aids the user in configuring the model on a given platform to meet the required mission time line. The remainder of this section provides the details of the operational considerations of the OMEGA modeling system.

a. Worldwide datasets

A key advantage of the OMEGA model is its worldwide datasets. The OMEGA model has eight major worldwide databases for terrain elevation, land–water distribution, soil type, land use–land cover, climatological vegetation index, climatological sea surface temperature, climatological subsurface temperature, and climatological soil moisture (Table 2). In addition to these datasets, the worldwide mapping dataset from the Digital Chart of the World is also included in the OMEGA modeling system.

The first two characteristics (terrain elevation and land–water distribution) are determined from global da-

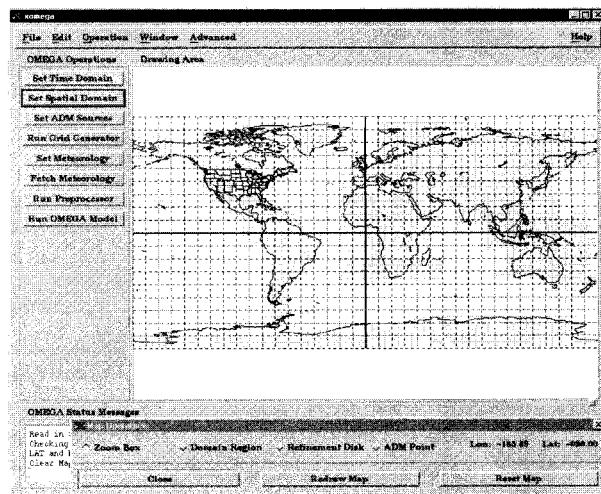


FIG. 15. XOMEGA, the OMEGA GUI

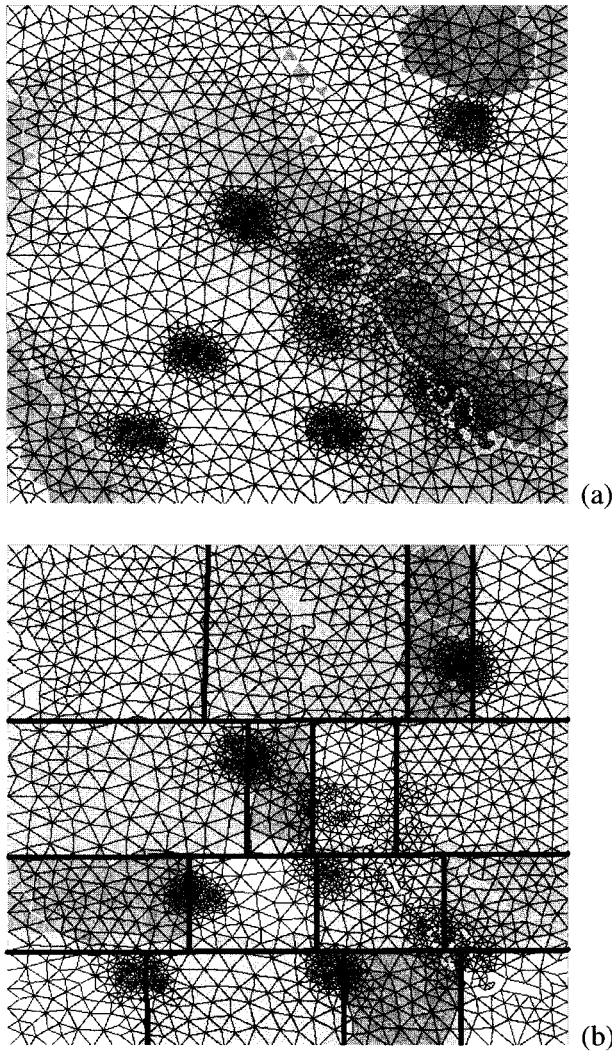
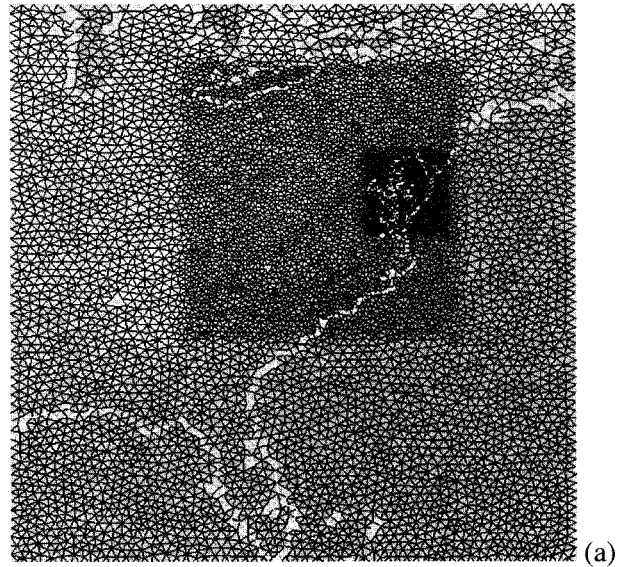


FIG. 16. (a) A selected model domain terrain topography and grid configuration; (b) domain decomposition of 16 subdomains of roughly equal number of cells in each subdomain. The heavy lines show the first cut at decomposition, which is refined by further passes.

tasets with two different resolutions: 1) moderate resolution of 5 arc-minutes (10 km), and 2) high resolution of 30 arc-second (1 km). The other characteristics rely on newer datasets from various sources. For example, a 1° global soil-type database was created from the *Global Ecosystems Database (GED)* CD-ROM (Webb et al. 1992). The 12 types used in the OMEGA preprocessor are sand, loamy sand, silt loam, loam, sandy clay loam, silty clay loam, silt, clay loam, sandy clay, silty clay, clay, and peat.

In OMEGA, vegetation type is defined for each grid cell based on three different datasets. The first dataset is derived from the global monthly Normalized Difference Vegetation Index (NDVI) datasets covering the period 1985–90 published on the *Global Ecosystems Database* CD-ROM, based on biweekly calibrated global



**OMEGA Speedup**

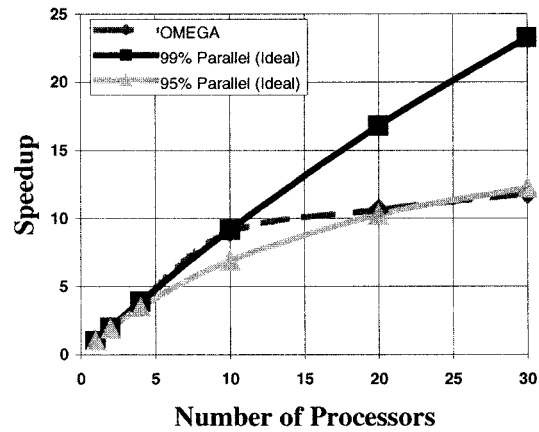
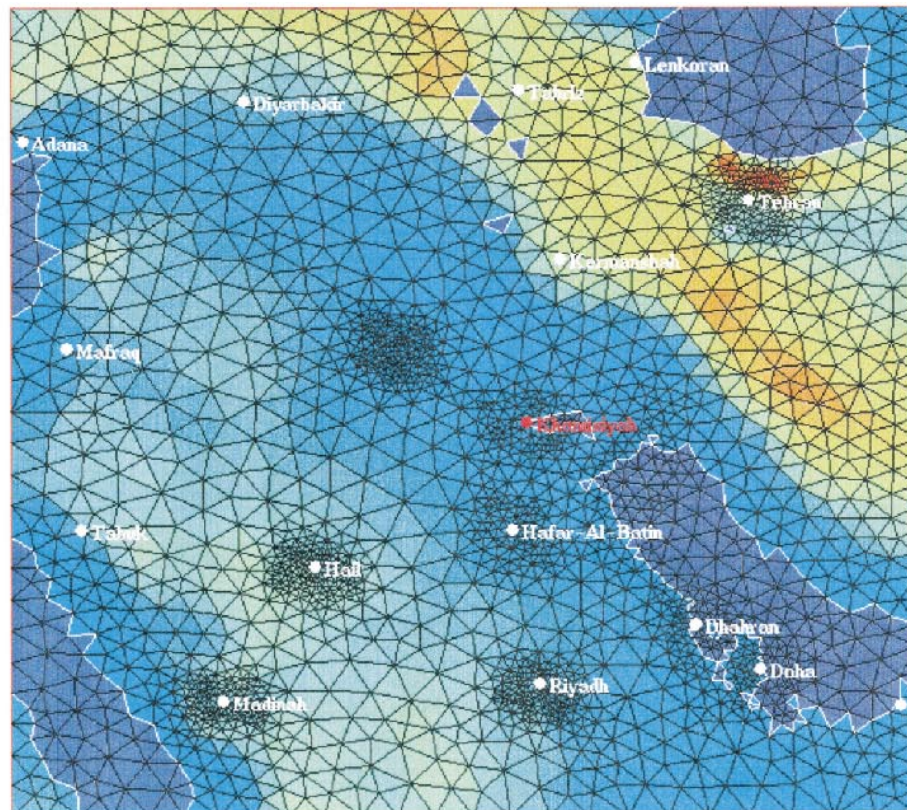


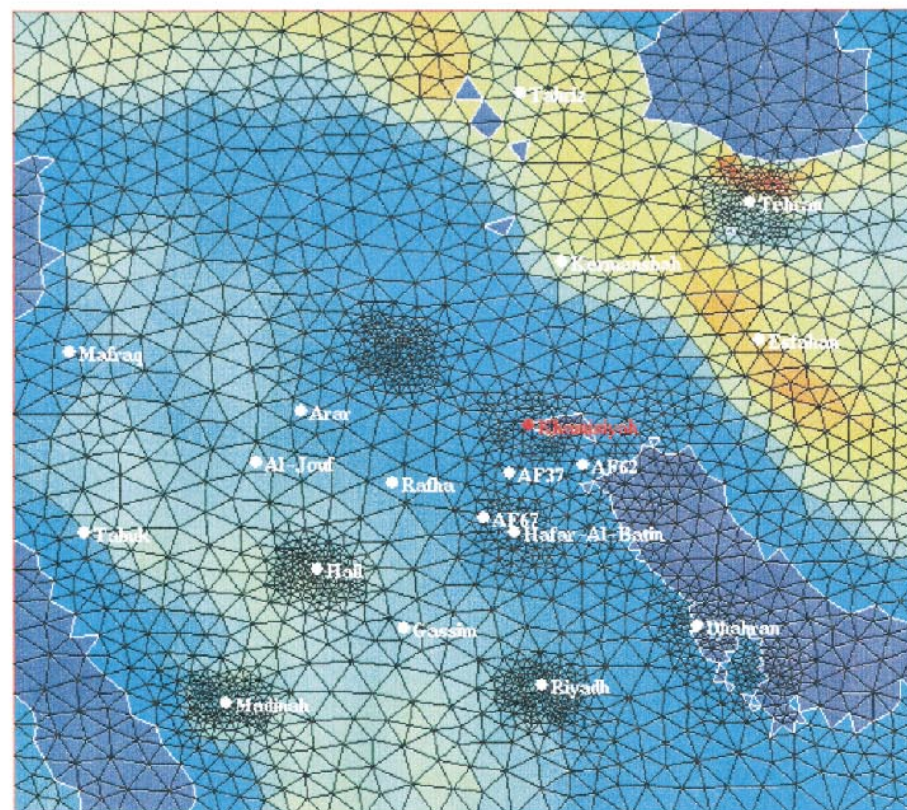
FIG. 17. (a) The grid used to test the parallel performance of OMEGA and (b) the performance of OMEGA for 1–30 processors.

vegetation index (GVI) data (EDC-NESDIS 1992). These monthly datasets were averaged to form a set of monthly climatology files with a resolution of 10 arc-minutes (18 km). The second dataset is derived from a series of high-resolution Advanced Very High Resolution Radiometer (AVHRR) datasets (Eidenshink 1992) over the conterminous United States published by the Earth Resources Observation System Data Center of the United States Geological Survey (USGS). The datasets were averaged from 1990 to 1993 to produce biweekly climatological datasets covering the conterminous United States with a resolution of 2 arc-minutes (about 4 km). The third database is a USGS biweekly dataset covering the globe with a resolution of 30 arc-seconds (1 km).

Similarly, three separate land use datasets were also



(a)



(b)

FIG. 18. Grid for forensic reconstruction of Khamisiyah, Iraq; and observational sites used for OMEGA comparison: (a) rawinsonde sites and (b) surface observation sites.



created for the OMEGA model. Each set uses the Biosphere–Atmosphere Transfer Scheme (BATS) set of 18 land cover categories, with an additional 19th category added for urban land cover. The categories are included for crop/mixed farming, short grass, evergreen needleleaf tree, deciduous needleleaf tree, deciduous broadleaf tree, evergreen broadleaf tree, tall grass, desert, tundra, irrigated crop, semidesert, ice cap/glacier, bog or marsh, inland water, ocean, evergreen shrub, deciduous shrub, mixed woodland, and urban. A 2 arc-minute (4 km) land use dataset covering the United States was formed from a 1-km BATS dataset and AVHRR data. A 30 arc-minute (55 km) global dataset was formed from the Olson (1992) World Ecosystems dataset on the *Global Ecosystems Database* CD-ROM. Also a 30 arc-second (1 km) global dataset was formed from USGS.

OMEGA uses a global sea surface temperature climatological dataset produced by the USGS (Schweitzer 1993). The data have been processed into a biweekly global, 12 arc-minute resolution (about 20 km) dataset. This data does not extend beyond about 70° latitude in either hemisphere. Some areas are not covered resulting in large lakes without data; for example the Caspian Sea in central Asia is completely missed.

A monthly global subsurface temperature climatology was built from monthly average surface air temperatures from the *Global Ecosystems Database* CD-ROM (Legates and Willmott 1992). These data were built by using monthly average surface air temperature as an estimate of subsurface temperature. The dataset resolution is 30 arc-minutes. Finally, Budyko's simple soil moisture budget as described by Sellers (1965) was used to generate soil moisture estimates from the temperature and precipitation data with a 30 arc-minute resolution (Legates and Willmott 1992). This simple approach is far from perfect, but seems to give reasonable results for a range of climatic types. An iterative application of the Budyko equations was used to obtain volumetric soil water contents that balanced the annual water cycle.

With the exception of the land–water and terrain elevation databases each of the other surface characteristics databases are organized as sets of latitude–longitude tiles. The global-scale databases use 30° × 30° tiles, while the U.S. databases consist of 5° × 5° tiles. FORTRAN logic was devised to allow databases at both resolutions (U.S. and global coverages) to be used simultaneously. The higher-resolution data are applied to the OMEGA grid first, then the coarser-resolution data are used in any grid cells that have not been filled with values. After all of the available databases have been searched, the database values that have accumulated in each OMEGA grid cell are averaged to obtain the final value. In the case of the discrete fields (land–water, soil type and land use), the dominant (most frequent) value in each cell is considered to represent the entire cell.

#### b. Automation

Another key advantage of OMEGA is its integrated Graphical User Interface, XOMEGA (Fig. 15). XOM-

TABLE 3. Upper air error statistics.

Pressure (mb)	No. of points	Me (K)	Mae (K)	Rmse (K)
Temperature				
<100	1006	0.29	1.35	1.76
<350	902	0.27	1.31	1.72
<500	706	0.33	1.34	1.71
<850	185	−0.31	1.53	2.02
Wind speed				
<100	743	−0.22	6.86	10.47
<350	651	−0.18	6.51	10.21
<500	532	−0.29	5.95	9.78
<850	142	0.32	4.37	5.46
Wind direction				
<100	743	5.91	22.08	35.61
<350	651	6.33	23.96	37.69
<500	532	6.48	26.75	40.67
<850	142	7.88	42.02	56.64

EGA provides a user-friendly method for the rapid re-configuration of the model. Starting from a browse map of the world, an operator can use XOMEGA to set up and start a simulation anywhere in the world in a matter of minutes. XOMEGA simplifies the process of defining the computational domain, generating a grid with appropriate resolutions, acquiring the meteorological data files, building the initial and boundary conditions for the OMEGA model, specifying the source characteristics for the ADM model if there are any, running the OMEGA model, and finally analyzing the OMEGA model results.

In addition to XOMEGA, an X-windows graphics postprocessing and analysis tool has been developed for the OMEGA model in order to analyze the OMEGA model results. XGRID brings a wealth of functionality to the OMEGA modeling system and includes the ability to overlay OMEGA output on mapping data from the Digital Chart of the World. XGRID also permits the examination of vertical cross sections of the OMEGA output, and the creation of standard skew  $T$ – $\log p$  diagrams. Using XGRID, the user can examine both meteorological and ADM results. A scripting capability makes it easy to create routine products automatically, creating either Postscript or GIF files, which can be spooled into a printer or incorporated into other products, or MPEG animations.

#### c. Runtime issues

The runtime requirement for the operational models is the primary issue driving the development. The OMEGA model is CPU bound. Normally less than 1% of the CPU is used for input/output and memory management. OMEGA is dominated by vector operations, which are inherently faster than scalar operations and hence good execution speeds are being achieved. OMEGA also showed a computational intensity ratio of flops to memory references of 0.99.

TABLE 4. Surface temperature statistics.

Station	Alt. diff. (m)	No. of points	Me (K)	Mae (K)	Rmse (K)
AF37	0	58	-1.23	2.21	3.02
AF62	0	68	1.22	2.06	3.32
AF67	0	73	-1.44	2.05	2.77
Al Jouf	111	69	-2.84	2.96	3.48
Arar	55	24	-4.47	4.47	4.81
Dhahran	39	72	-0.30	1.00	1.38
Esfahan	551	15	-7.12	7.12	7.50
Gassim	144	70	-0.72	2.23	2.50
Hafar al Batin	-32	71	-0.97	1.22	1.72
Hail	114	69	-1.90	2.29	2.60
Kermanshah	5	12	-1.68	2.12	2.79
Madinah	216	72	-2.39	2.67	2.91
Mafrq	-393	7	3.15	3.15	3.50
Rafha	172	65	-2.28	2.45	3.08
Riyadh	13	70	0.66	1.00	1.28
Tabriz	362	70	0.58	1.89	2.27
Tabuk	90	14	-3.43	3.54	4.07
Tehran	169	17	1.37	3.30	3.68

The optimization level used in the OMEGA model maximizes inlining and vectorization. Inlining is a technique to reduce CPU associated with subroutine/function calls. It can cost up to 150 clock cycles to call a single argument subprogram. Vectorization allows a single operation to be performed on vectors of operands instead of sequentially for each operand. With the optimization level used in the OMEGA model, the overall computational speed achieved is slightly over 40 Mflops.

Parallelization is another state-of-the-art technique to boost computational speedup by multiprocessing. However, unstructured grid models tend to demand significantly more synchronization and geometry mapping overhead in their parallelization algorithm. A version of OMEGA with explicit parallelization using the Message Passing Interface (MPI) library is undergoing testing on parallel workstation platforms. The functionality of MPI's allows (a) parallelization in an overarching man-

ner in the OMEGA model time advancement loop, making message passing minimal; (b) pinpointing solutions to the idle time, or the imbalanced load problem, such as techniques accounting for the time-masked subcycling and dynamic domain-decomposition; and (c) transportability.

Parallelization by domain decomposition of the model domain into contiguous subdomains is currently under development and testing. Figure 16 shows an example of the technique applied to a typical OMEGA grid. The figure shows how the solution area is decomposed into 16 subdomains resulting in roughly the same number of cells in each of the subdomains. On top of the communication overheads, stacking and unstacking arrays for lumped message passing is an inherent overhead for unstructured grid models. These extra CPU costs are both related to message passing; however, by far the majority of the total overhead is due to uneven load balancing among the processors. For application on

TABLE 5. Surface wind speed statistics.

Station	Alt. diff. (m)	No. of points	Me (kt)	Mae (kt)	Rmse (kt)
AF37	0	58	-2.08	3.27	3.99
AF62	0	69	-0.63	2.83	3.62
AF67	0	73	-0.53	2.42	3.16
Al Jouf	111	69	-4.14	4.78	6.26
Arar	55	24	-1.94	2.97	3.59
Dhahran	39	72	-1.69	3.18	4.11
Esfahan	551	15	4.0	5.77	6.56
Gassim	144	70	0.23	2.93	3.91
Hafar al Batin	-32	71	-4.53	5.62	6.07
Hail	114	70	-1.27	2.99	4.40
Kermanshah	5	10	0.15	5.86	6.64
Madinah	216	72	0.99	2.94	3.60
Mafrq	-393	7	9.16	9.16	9.83
Rafha	172	65	-5.77	5.98	7.00
Riyadh	13	70	-0.26	2.70	3.58
Tabriz	362	70	1.53	4.19	5.08
Tabuk	90	14	4.29	6.25	7.09
Tehran	169	17	-1.42	5.12	6.47

TABLE 6. Surface wind direction statistics.

Station	Alt. diff. (m)	No. of points	Me (°)	Mae (°)	Rmse (°)
AF37	0	58	-4.92	31.10	45.40
AF62	0	69	10.79	60.53	81.85
AF67	0	73	6.35	29.91	38.63
Al Jouf	111	69	3.99	75.48	94.58
Arar	55	24	-3.21	55.77	74.81
Dhahran	39	72	34.45	48.72	66.96
Esfahan	551	15	-1.01	104.75	117.34
Gassim	144	70	18.21	31.47	42.77
Hafar al Batin	-32	71	-18.13	25.46	30.35
Hail	114	70	16.70	68.53	85.41
Kermanshah	5	10	10.31	57.51	68.27
Madinah	216	72	13.09	58.68	69.84
Mafraq	-393	7	105.68	105.68	113.49
Rafha	172	65	-26.07	47.89	65.76
Riyadh	13	70	-6.70	30.08	37.36
Tabriz	362	70	-7.38	50.52	69.14
Tabuk	90	14	-3.75	55.15	67.42
Tehran	169	17	39.63	79.96	89.86

massively parallelized machines the scheme is designed to utilize over 100 processors. MPI command calls facilitate the parallelization of the time advancement loop of the model.

The parallel efficiency of OMEGA is roughly 95%. Amdahl's law (1967) states that the speedup achieved by using  $N$  processors of a numerical algorithm that is  $f_p$  percent parallelized is

$$\text{speedup} = \frac{1}{1 - f_p + f_p/N}$$

Figure 17a shows a large grid that was constructed for testing the parallel version of OMEGA. This grid has 13 935 cells in each of 31 levels. The grid resolution ranged from 5 to 50 km. Figure 17b shows the performance of OMEGA for 1–30 processors. Also shown are the theoretical ideal performance curves for 95% and 99% parallelization.

*d. Portability issues*

OMEGA produces a variety of output files (both binary and ASCII) depending on the mode of operation. The frequency of output is controlled by the user and is generally set for hourly. In order to allow OMEGA output from one system to be read and/or visualized on different hardware, we have developed a machine-independent, packed binary output data format for OMEGA.

The packed binary format is a form of compression that results in files that are written and read on the byte level by routines written in the C programming language. These files are now completely portable and may be cross ported to any machine, including PCs. There is a small penalty in runtime, which is more than compensated for by savings in postprocessing. The packed binary format files are also smaller and more efficient to read. In general, data are written using eight bytes to represent one number. With packed binary, this same

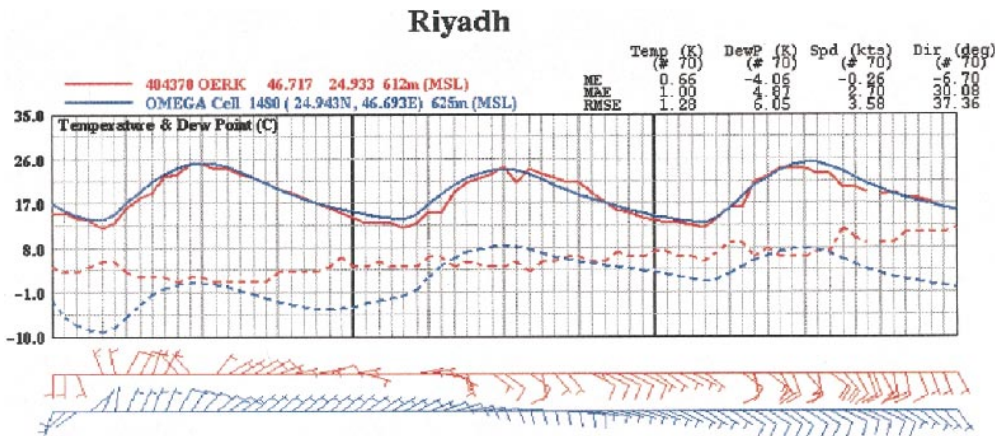


FIG. 19. The OMEGA-model-predicted and observed diurnal cycle of the atmosphere at Riyadh.

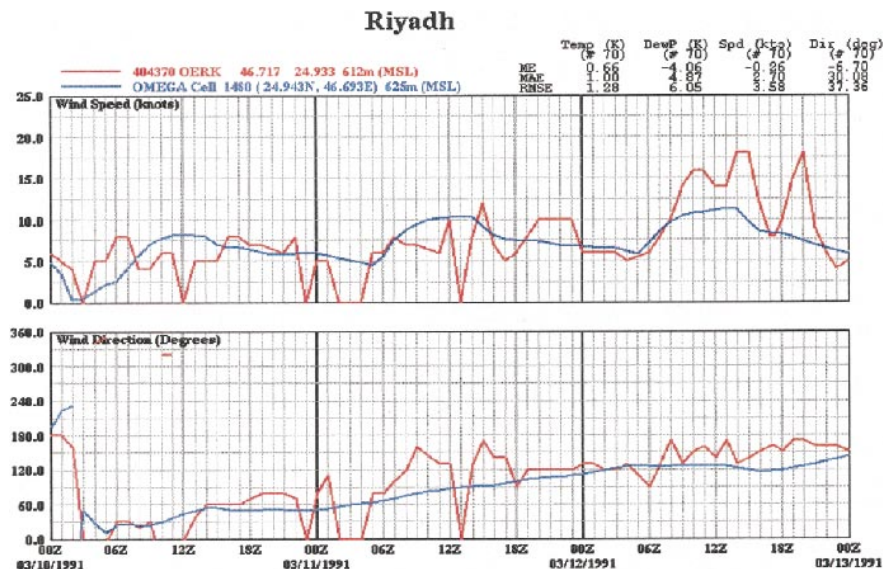


FIG. 20. The OMEGA-model-predicted and observed diurnal cycle of the wind speed and direction at Riyadh.

number may be represented by as little as one byte. Although accuracy diminishes as fewer bytes are used, for most purposes, we have found that the use of two bytes, which results in accuracy to five significant digits, is sufficient and requires one-quarter the amount of disk space used generally.

**6. Forensic reconstruction of Khamisiyah, Iraq**

The creation of OMEGA was prompted, in part, by the lack of a high-fidelity, high-resolution atmospheric dispersion capability during the Desert Storm military campaign. It was recognized that both the meteorological and dispersion simulation capabilities would have to be improved in order to deal with situations that were data sparse or data denied. OMEGA was created to ob-

tain the maximum utility from the available meteorological and surface information. Given this genesis, it was natural that OMEGA would be called upon to reconstruct the meteorological event in Khamisiyah, Iraq, on 10 March 1991.

In early March 1991, U.S. forces destroyed many weapons bunkers in Iraq. Some years later, it was discovered that a cache of weapons destroyed at Khamisiyah on the afternoon of 10 March contained chemical agents. The reconstruction of the atmospheric conditions at the time of the release was a major undertaking involving the acquisition of raw meteorological data from many sources, gridded analyses, and the detailed simulation of the mesoscale circulations of the region. During this reconstruction, it was discovered that much of the operational data were delayed or denied for the area

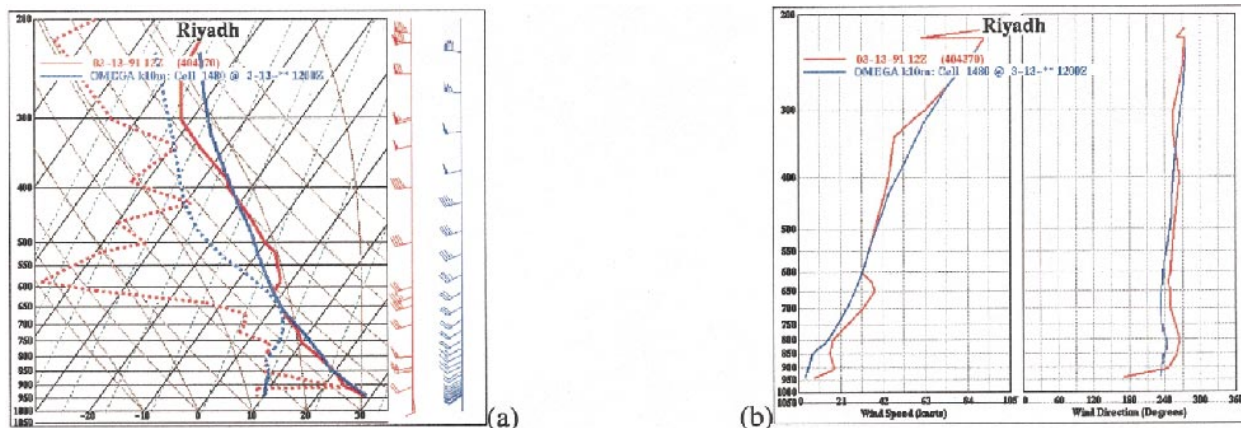


FIG. 21. The OMEGA-model-predicted and observed vertical profiles of (left) temperature and dewpoint and (right) wind speed and direction for Riyadh at 84 h after initialization.

### Hafar Al Batin

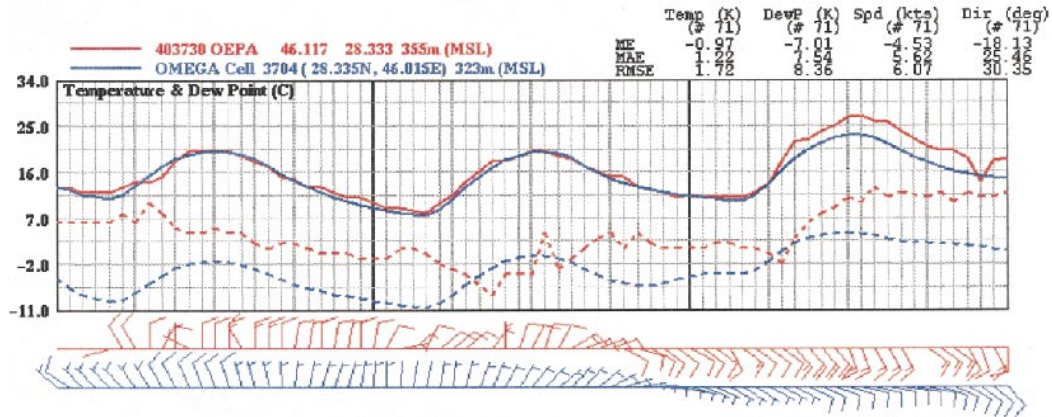


FIG. 22. OMEGA-predicted and observed diurnal cycle of the atmosphere for Hafar al Batin.

and thus did not make it into the archived analyses and operational data archives. This led to a search for the late arriving data and the rerunning of the simulations. Thus, it is important to understand that the simulation presented here is merely the latest in a chain of simulations conducted to reconstruct the meteorological events at the time of the event.

The computational domain enclosed the area bounded by approximately 22°–40°N and 35°–55°E. Figure 18 shows the OMEGA grid used for this simulation. This grid contained 4333 cells in each of 30 levels. The grid provided high resolution around the Khamisiyah pit and some of the observation locations using the static ad-

aptation capability of the OMEGA model. The simulation was initialized with gridded data from the Global Optimal Interpolation archives of NCAR and observational data obtained from multiple sources. Model initialization time was 0000 UTC 10 March 1991 and the simulation period was 84 h extending to 1200 UTC 13 March 1991.

The OMEGA results were then compared to observational data. The only observational data that were available were rawinsonde (15 sites) and surface observations (18 sites) from several locations in the region (see Fig. 18). In order to compare point data with the three-dimensional output of OMEGA, we wrote a post-

### Hafar Al Batin

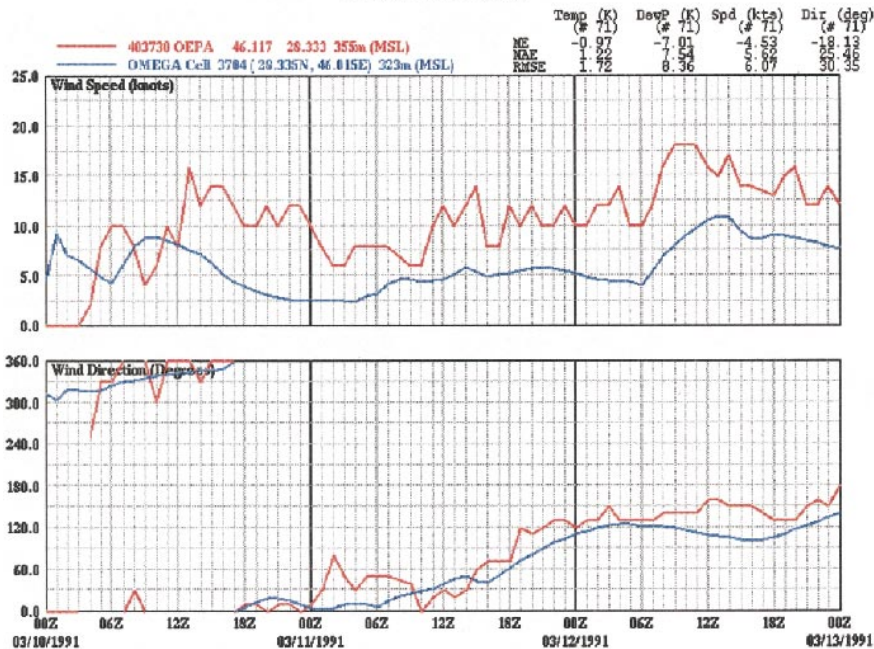


FIG. 23. The OMEGA-predicted and observed wind speed and direction for Hafar al Batin.

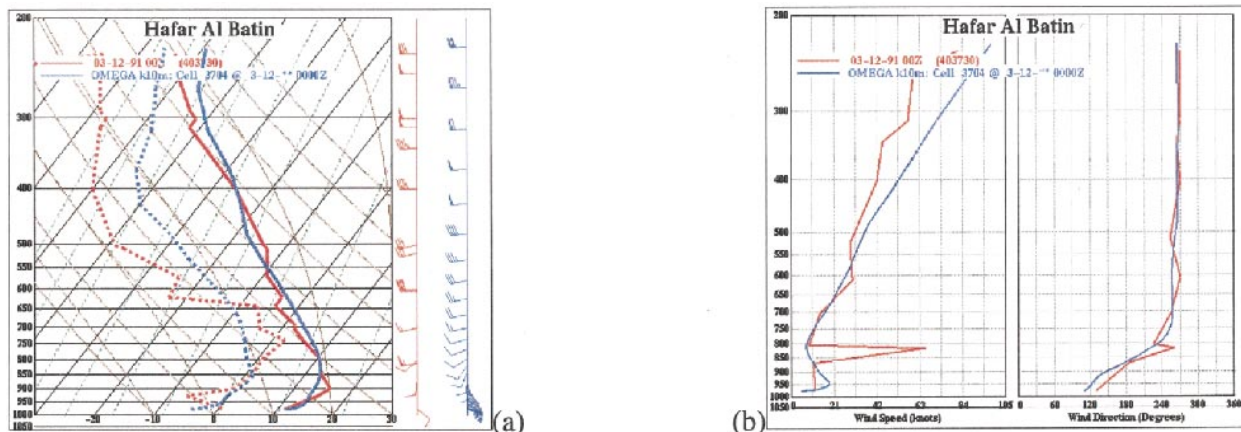


FIG. 24. The OMEGA-model-predicted and observed the vertical profile of (left) temperature and dewpoint and (right) wind speed and direction for Hafar al Batin, the closest observation to Khamisiyah.

processor that locates the OMEGA cell centroids closest to the observational points, and outputs pseudo-surface observations and pseudo-rawinsonde observations at these locations. These pseudoobservations were compared against the actual observations and error statistics were also calculated. The statistics produced were mean error (me), mean absolute error (mae), and root-mean-square error (rmse) for temperature, wind speed, and wind direction. Table 3 shows these error statistics for upper air at various pressure levels computed over the 3-day forecast duration (0000 UTC 10 Mar 1991–0000 UTC 13 Mar 1991) using all the rawinsonde data (every 12 h) and all the locations (15 sites) within the computational domain, while Tables 4–6 show the error statistics for surface level computed over the 3-day forecast duration (0000 UTC 10 Mar 1991–0000 UTC 13 Mar 1991) using all the surface observation data (every hour) and all the locations (18 sites).

The above statistical results indicate that the OMEGA model predicted the meteorological conditions that might have existed during the period of 10–13 March 1991 reasonably well. For example, upper air temperature statistics indicate less than 0.5-K temperature bias, less than 1.5-K mean absolute error, and less than 2-K root-mean-square error in upper air error statistics, while surface error statistics indicate wide range values. These wide range values could be due to the altitude difference between the observational site and the sampled closest OMEGA cell.

To examine the observational datasets in greater detail, two sites across the simulation domain were also chosen to compare both the rawinsonde and hourly surface observations with OMEGA prediction. The following sites were chosen: Riyadh and Hafar Al Batin (see Fig. 18). Figure 19 shows comparison of the 72-h time history of OMEGA-predicted surface temperature (blue solid line), dewpoint temperature (blue dashed lines), and wind speed and direction (blue wind bars) with surface observations (red) for Riyadh. A more detailed

evaluation of the surface wind speed and direction is also shown in Fig. 20. Note that 72-h surface statistics are also calculated and printed in the top-right corner of each figure. Both the figures indicate that the OMEGA-predicted values during the 72-h period show very good agreement with observations. For example, temperature statistics indicate 0.66-K bias, 1-K mean absolute error, and 1.28-K root-mean-square error, while both surface wind speed and direction also show very good agreement with observations. Figure 21 shows the comparison of the OMEGA model sounding (blue line) with the observed sounding (red line) at Riyadh 1200 UTC 13 March 1991 (84 h of forecast). The sounding shows very dry air with very little vertical directional wind shear. The wind is predominately from the west except for a small segment from ground level to 1 km above the surface where the wind is from the southwest. Temperature, wind speed, and wind direction profiles show very good agreement with the observed values in both soundings and surface observations. Figures 22 and 23 show the same diurnal cycle for Hafar Al Batin. Again the predicted values show very good agreement with observed values with the exception of wind speed. In this location, the wind speed is slightly underestimated as compared to the observed values. Note also that the predicted surface temperature is slightly deviated from the observation on the third day of the forecast. We believe that this may be due to the fact that the OMEGA model is run in a straight forecast mode. This can be overcome by running the model in reanalysis mode, reinitializing the model every 12 h using the OMEGA-model-predicted fields as a first guess, and ingesting new observations.

The corresponding sounding profile taken at Hafar Al Batin at 1200 UTC 12 March 1991 (48 h of forecast) is shown in Fig. 24. As the night progresses, both temperature and wind exhibit a very complex behavior, as opposed to the case presented at Riyadh (daytime situation). The bottom portion of the atmospheric bound-

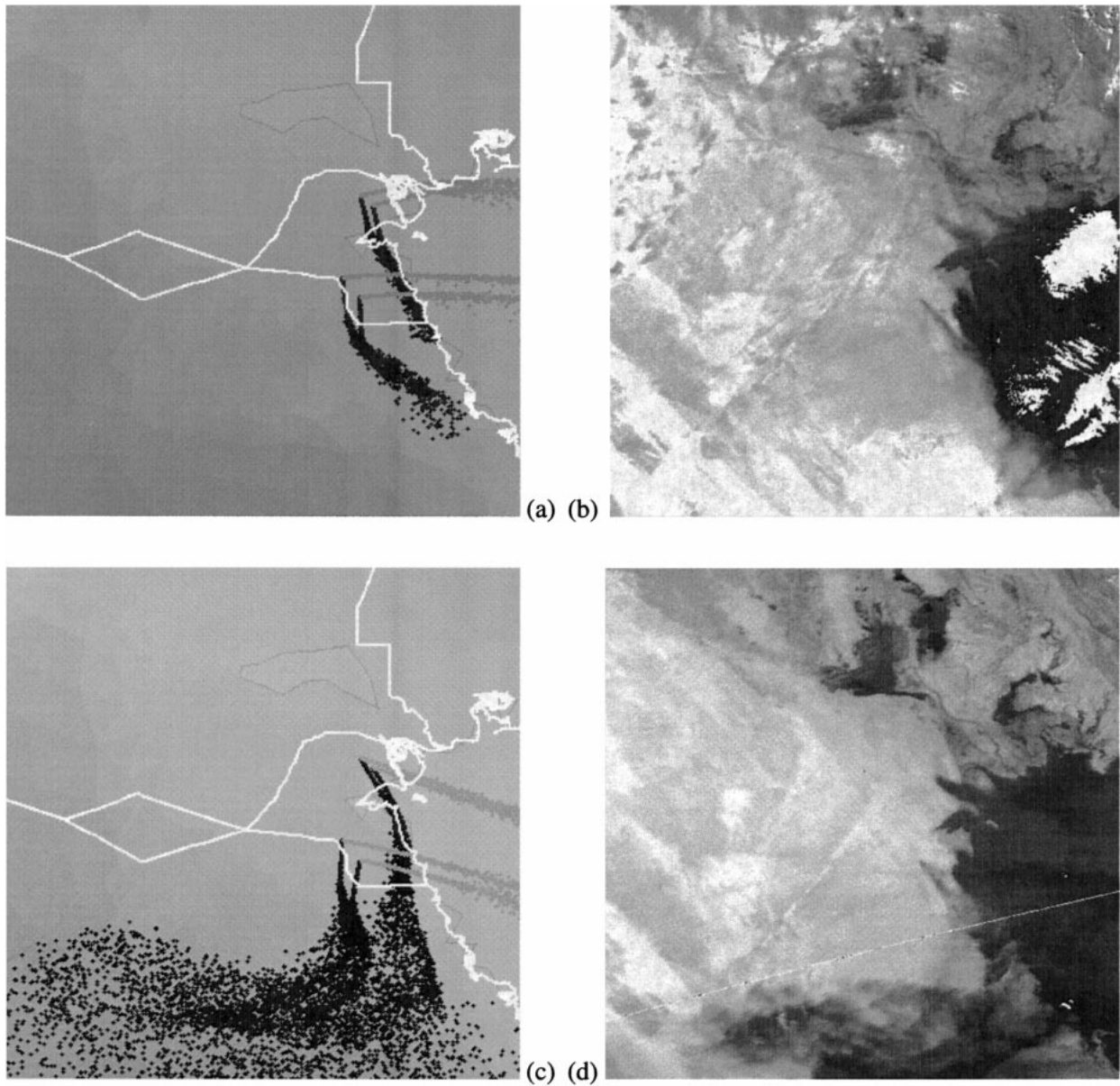


FIG. 25. A close-up view of the northern Persian Gulf region showing the OMEGA-simulated oil field fire plumes and the AVHRR image data. The top row shows (a) the forecasted dispersion and (b) the observed at 1030 UTC 10 Mar 1991; the bottom row shows the same for 1020 UTC 11 Mar 1991.

ary layer is transformed by its contact with the ground into a nocturnal stable boundary layer. The OMEGA model captures this feature reasonably well (Fig. 24). The temperature near the surface drops dramatically because the outgoing net radiation is not compensated by the downward heat flux. This is characterized in Fig. 24a by statically stable air with weaker, sporadic turbulence near the bottom. At the same time, the OMEGA simulations indicate a strong wind shear forms near the surface due to the downward transfer of momentum. However, the observation does not show this feature, possibly due to a lack of resolution near the surface.

(Also note in Fig. 24b that a clearly bad velocity data point was reported at the 810-mb level.)

The statistics and the detailed analysis at Hafar Al Batin and Riyadh presented here have demonstrated that the OMEGA model reconstructed the meteorological conditions in southeastern Iraq very well for the period from 10 to 13 March 1991. Table 6 also indicates that there is distinctly better agreement between the observations and the simulation results for those regions that are well resolved spatially (cf. Fig. 18). This shows the potential that an adaptive grid simulation system brings to atmospheric simulation.

Finally, Fig. 25 shows OMEGA-predicted plumes from the oil well fires in Kuwait compared with the observed plumes. Careful examination of the AVHRR imagery shows that there seem to be both a surface component and a high-altitude component to the pollutant source. Because of this unknown aspect of the oil fires, they were simulated in OMEGA using two different elevated releases—one at 20 m and one at 1500 m. The results shows that OMEGA predicted a plume trajectory and spread well in qualitative agreement with the satellite observations.

## 7. Conclusions

The OMEGA modeling system represents a significant departure from the traditional methods used in numerical weather prediction and real-time hazard prediction. For the first time in recent years, advanced numerical and grid-generation methods developed by the computational fluid dynamics community have been successfully applied to the problem of atmospheric simulation. This has permitted the development of an extremely high resolution and flexible operational atmospheric simulation system. Future articles will present a spectrum of case studies using OMEGA providing additional validation of the model as well as demonstrating in more depth the flexibility and power of unstructured static and dynamically adapting grids.

To the best of our knowledge, the OMEGA model and its Atmospheric Dispersion Model are the only operational atmospheric flow system based on unstructured grid technique, which fully exploits the advantages and flexibility of unstructured grids. It can adapt its grid both statically and dynamically to different criteria such as fronts, clouds, hurricanes, and plumes, etc. For real-time flow predictions, the capability of grid adaptivity, given the CPU constraint, becomes important. This capability is also crucial in responding to emergency scenarios such as release of hazardous materials. OMEGA with its grid adaptation capability has a potentially unique advantage over other atmospheric flow models in providing accurate solutions quickly in an operational setting.

*Acknowledgments.* This work was supported by the Defense Threat Reduction Agency (DTRA) under Contracts DNA001-92-C-0076, DNA001-95-C-0130, and DTRA 01-99-C-0007. The authors would like to thank their program managers, Dr. Charles Galloway, Dr. James Hodge, and Maj. Thomas Smith, for their support. The authors would also like to thank to Dr. Michael D. McCorcle, Mr. Christopher Agritellis, Mr. Douglas E. Mays, and Mr. Tim Wait for their many contributions to this effort.

## REFERENCES

- Amdahl, G. M., 1967: Validity of the single processor approach to achieving large scale computer capabilities. *Proc. of the AFIPS Spring Joint Computer Conference*, Vol. 30, AFIPS Press, 483–485.
- Anthes, R. A., 1977: A cumulus parameterization scheme utilizing a one-dimensional cloud model. *Mon. Wea. Rev.*, **105**, 270–286.
- Arya, S. P., 1999: *Air Pollution Meteorology and Dispersion*. Oxford University Press, 157 pp.
- Bacon, D. P., and R. A. Sarma, 1991: Agglomeration of dust in convective clouds initialized by nuclear bursts. *Atmos. Environ.*, **25A**, 2627–2642.
- Baum, J. D., and R. Löhner, 1994: Numerical simulation of shock-elevated box interaction using an adaptive finite element shock capturing scheme. *AIAA Journal*, **32**, 682–692.
- , H. Luo, and R. Löhner, 1993: Numerical simulation of a blast inside a Boeing 747. *24th Fluid Dynamics Conf.*, AIAA 93-3091, 1–8.
- Beljaars, A. C. M., and A. A. M. Holtslag, 1991: Flux parameterization over land surfaces for atmospheric models. *J. Appl. Meteor.*, **30**, 327–341.
- Boybeyi, Z., S. Raman, and P. Zannetti, 1995: Numerical investigation of possible role of local meteorology in Bhopal gas accident. *Atmos. Environ.*, **29**, 479–496.
- Chang, J. T., and P. J. Wetzel, 1991: Effects of spatial variations of soil moisture and vegetation on the evolution of a prestorm environment: A numerical case study. *Mon. Wea. Rev.*, **119**, 1368–1390.
- Charnock, H., 1955: Wind stress on a water surface. *Quart. J. Roy. Meteor. Soc.*, **81**, 639–640.
- Côté, J., J.-G. Desmarais, S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth, 1998a: The operational CMC-MRB Global Environmental Multiscale (GEM) model. Part II: Results. *Mon. Wea. Rev.*, **126**, 1397–1418.
- , S. Gravel, A. Méthot, A. Patoine, M. Roch, and A. Staniforth, 1998b: The operational CMC-MRB Global Environmental Multiscale (GEM) model. Part I: Design considerations and formulation. *Mon. Wea. Rev.*, **126**, 1373–1395.
- Daley, R., 1991: *Atmospheric Data Analysis*. Cambridge University Press, 457 pp.
- Deardorf, J. W., 1974: Three-dimensional study of the height and mean structure of a heated planetary boundary layer. *Bound.-Layer Meteor.*, **7**, 81–106.
- Dietachmeyer, S. G., and K. K. Drogemeier, 1992: Application of continuous dynamic grid adaptation techniques to meteorological modeling. Part I: Basic formulation and accuracy. *Mon. Wea. Rev.*, **120**, 1675–1706.
- DiMego, G. J., 1988: The National Meteorological Center regional analysis system. *Mon. Wea. Rev.*, **116**, 977–1000.
- Dudhia, J., 1993: A nonhydrostatic version of the Penn State/NCAR mesoscale model: Validation tests and simulation of an Atlantic cyclone and cold front. *Mon. Wea. Rev.*, **121**, 1493–1513.
- EDC-NESDIS, 1992: Monthly global vegetation index from Gallo bi-weekly experimental calibrated GVI (April 1985–December 1990). Digital raster data on a 10-minute geographic (lat/long) 1080 × 2160 grid. *Global Ecosystems Database*, version 1.0, National Geophysical Data Center, NOAA, Boulder, CO, 69 independent single-attribute data layers on CD-ROM, disk A.
- Eidenshink, J. E., 1992: The conterminous United States AVHRR data set. *Photogramm. Eng. Remote Sens.*, **58**, 809–813.
- Fiedler, B., Z. Huo, and P. Zhang, 1998: Dynamic grid adaption for atmospheric boundary layers. Preprints, *12th Conf. on Numerical Weather Prediction*, Phoenix, AZ, Amer. Meteor. Soc., 253–254.
- Fritts, M. H., 1988: Adaptive gridding strategies for Lagrangian calculations. *Comput. Phys. Comm.*, **48**, 75–88.
- Grell, G. A., J. Dudhia, and D. R. Stauffer, 1994: A description of the Fifth-Generation Penn State/NCAR Mesoscale Model (MM5). NCAR/TN-398+IA, National Center for Atmospheric Research, Boulder, CO, 107 pp.
- Hoke, J. E., N. A. Phillips, G. J. DiMego, J. J. Tuccillo, and J. G. Sela, 1989: The regional analysis and forecast system of the National Meteorological Center. *Wea. Forecasting* **4**, 323–334.



- Janjic, Z. I., 1990: The step-mountain coordinate: Physical package. *Mon. Wea. Rev.*, **118**, 1429–1443.
- Jones, R. W., 1977: A nested grid for a three-dimensional model of a tropical cyclone. *J. Atmos. Sci.*, **34**, 1528–1553.
- Karniadakis, G. E. M., and S. J. Sherwin, 1999: *Spectral/hp Element Methods for CFD*. Numerical Mathematics and Scientific Computation Series, Oxford University Press, 390 pp.
- Klemp, J. B., and R. Wilhelmson, 1978: The simulation of three-dimensional convective storm dynamics. *J. Atmos. Sci.*, **35**, 1070–1096.
- Kuo, H. L., 1965: On formation and intensification of tropical cyclones through latent heat release by cumulus convection. *J. Atmos. Sci.*, **22**, 40–63.
- , 1974: Further studies of the parameterization of the influence of cumulus on large scale flow. *J. Atmos. Sci.*, **31**, 1232–1240.
- Legates, D. R., and C. J. Willmott, 1992: Monthly average surface air temperature and precipitation. Digital raster data on a 30 minute geographic (lat/long)  $360 \times 720$  grid. *Global Ecosystems Database*, version 1.0, National Geophysical Data Center, NOAA, Boulder, CO, 48 independent and 4 derived single-attribute spatial layers on CD-ROM, disk A.
- Lilly, D. K., 1990: Numerical predictions of thunderstorms—Has its time come? *Quart. J. Roy. Meteor. Soc.*, **116**, 779–798.
- Lin, Y.-L., R. D. Farley, and H. D. Orville, 1983: Bulk parameterization of the snow field in a cloud model. *J. Climate Appl. Meteor.*, **22**, 1065–1092.
- Lottati, I., and S. Eidelman, 1994: A second-order Godunov scheme on a spatial adapted triangular grid. *Appl. Numer. Math.*, **14**, 353–365.
- Luo, H., J. D. Baum, R. Lohner, and J. Cabello, 1994: Implicit schemes and boundary conditions for compressible flows on unstructured meshes. Preprints, *32d Aerospace Sciences Meeting and Exhibit*, AIAA 94-0816, Reno, NV, 12 pp.
- Mahrer, Y., and R. A. Pielke, 1977: The effects of topography on the sea and land breezes in a two-dimensional numerical model. *Mon. Wea. Rev.*, **105**, 1151–1162.
- McCorcle, M. D., 1988: Simulation of soil moisture effects on the Great Plains low-level jet. *Mon. Wea. Rev.*, **116**, 1705–1720.
- McPherson, R., 1991: 2001—An NMC odyssey. Preprints, *Ninth Conf. on Numerical Weather Prediction*, Denver, CO, Amer. Meteor. Soc., 1–4.
- Mellor, G. L., and T. Yamada, 1974: A hierarchy of turbulence closure models for planetary boundary layers. *J. Atmos. Sci.*, **31**, 1791–1806.
- , and —, 1982: Development of a turbulence closure model for geophysical fluid problems. *Rev. Geophys. Space Phys.*, **20**, 851–875.
- Mesinger, F., Z. I. Janjic, S. Nickovic, D. Gavrilov, and D. G. Deaven, 1988: The step-mountain coordinate: Model description and performance for cases of Alpine lee cyclogenesis and for a case of an Appalachian redevelopment. *Mon. Wea. Rev.*, **116**, 1494–1518.
- Molinari, J., and M. Dudek, 1986: Implicit versus explicit convective heating in numerical weather prediction models. *Mon. Wea. Rev.*, **114**, 1822–1831.
- Moran, M. D., 1992: Numerical modeling of mesoscale atmospheric dispersion. Ph.D. dissertation, Colorado State University, Fort Collins, CO, 758 pp. [Available from Dept. of Atmospheric Science, Colorado State University, Fort Collins, CO 80523.]
- Noilhan, J., and S. Planton, 1989: A simple parameterization of land surface processes for meteorological models. *Mon. Wea. Rev.*, **117**, 536–549.
- Olson, J. S., 1992: World ecosystems (WE1.4). Digital raster data on a 10-minute geographic  $1080 \times 2160$  grid. *Global Ecosystems Database*, version 1.0, National Geophysical Data Center, NOAA, Boulder, CO, 3 independent single-attribute spatial layers on CD-ROM, disk A.
- Pielke, R. A., and Coauthors, 1992: A comprehensive meteorological modeling system—RAMS. *Meteor. Atmos. Phys.*, **49**, 69–91.
- Proctor, F. H., 1987: The Terminal Area Simulation System/Volume I: Theoretical formulation, NASA Contractor Rep. 4046, 154 pp. [Available from U.S. Dept. of Commerce, National Technical Info. Service, Springfield, VA 22151.]
- Sarma, A., N. Ahmad, D. Bacon, Z. Boybeyi, T. Dunn, M. Hall, and P. Lee, 1999: Application of adaptive grid refinement to plume modeling. C. A. Brebbia, N. Jacobson, and H. Power, Eds. *Air Pollution VII*, WIT Press, 59–68.
- Sasamori, T., 1972: A linear harmonic analysis of atmospheric motion with radiative dissipation. *J. Meteor. Soc. Japan*, **50**, 505–518.
- Schnack, D. D., I. Lottati, Z. Mikic, and P. Satyanarayana, 1998: A finite-volume algorithm for three-dimensional magnetohydrodynamics on an unstructured, adaptive grid in axially symmetric geometry. *J. Comput. Phys.*, **140**, 71–121.
- Schweitzer, P. N., 1993: *Modern Average Global Sea-Surface Temperature*. U.S. Geological Survey Digital Data Series DDS-10, CD-ROM.
- Sellers, W. D., 1965: *Physical Climatology*. University of Chicago Press, 272 pp.
- Skamarock, W. C., and J. B. Klemp, 1993: Adaptive grid refinement for two-dimensional and three-dimensional nonhydrostatic atmospheric flow. *Mon. Wea. Rev.*, **121**, 788–804.
- Smolarkiewicz, P. K., 1984: A fully multidimensional positive definite advection transport algorithm with small implicit diffusion. *J. Comput. Phys.*, **54**, 325–362.
- , and T. L. Clark, 1986: The multidimensional positive definite advection transport algorithm: Further development and application. *J. Comput. Phys.*, **67**, 396–438.
- , and W. W. Gradowski, 1990: The multidimensional positive advection transport algorithm: Nonoscillatory option. *J. Comput. Phys.*, **86**, 355–375.
- Stanforth, A., J. Côté, and J. Pudykiewicz, 1987: Comments on “Smolarkiewicz’s deformational flow.” *Mon. Wea. Rev.*, **115**, 894–900.
- Sykes, R. I., W. S. Lewellen, and S. F. Parker, 1986: A Gaussian plume model of atmospheric dispersion based on second-order closure. *J. Climate Appl. Meteor.*, **25**, 322–331.
- Uliasz, M., 1990: Development of the mesoscale dispersion modeling system using personal computers. Part I: Models and computer implementation. *Meteor. Z.*, **40**, 110–120.
- Webb, R. S., C. E. Rosenzweig, and E. R. Levine, 1992: A global data set of soil particle size properties. Digital raster data on a 1-degree geographic (lat/long)  $180 \times 360$  grid. *Global Ecosystems Database*, version 1.0, National Geophysical Data Center, NOAA, Boulder, CO, 2 independent and 1 derived spatial layer with 65 attributes on CD-ROM, disk A.
- Wilson, D. J., J. E. Fackrell, and A. G. Robins, 1982a: Concentration fluctuations in an elevated plume: A diffusion-dissipation approximation. *Atmos. Environ.*, **16**, 2581–2589.
- , A. G. Robins, and J. E. Fackrell, 1982b: Predicting the spatial distribution of concentration fluctuations from a ground level source. *Atmos. Environ.*, **16**, 497–504.
- Xue, M., K. K. Drogemeier, V. Wong, A. Shapiro, and K. Brewster, 1995: ARPS version 4.0 user’s guide. CAPS, 380 pp. [Available from CAPS, University of Oklahoma, Norman, OK 73019.]
- Zhang, D.-L., H.-R. Chang, N. L. Seaman, T. T. Warner, and J. M. Fritsch, 1986: A two-way interactive nesting procedure with variable terrain resolution. *Mon. Wea. Rev.*, **114**, 1330–1339.
- , Y. Liu, and M. K. Yau, 1999: Surface winds at landfall of Hurricane Andrew (1992)—A reply. *Mon. Wea. Rev.*, **127**, 1711–1721.