4M.7

A New Dynamically Adaptive Mesoscale Atmospheric Model

Xudong Xiao,[*] D. Scott McRae,[†] Hassan A. Hassan [‡]
Department of Mechanical and Aerospace Engineering, Campus Box 7910
North Carolina State University, Raleigh, NC 27695

1. Introduction

Resolution of atmospheric phenomena is an issue that permeates all of numerical meteorology. Unfortunately, it is clear that resolution limitations due to computer capability, algorithm and physics model will persist for the foreseeable future. The list of phenomena that a mesoscale model should resolve to be effective and useful include: gravity waves, rotors, cyclones, fronts, clouds, clear air turbulence and optical turbulence *etc*. However, resolution of these phenomena is seldom adequate since the computational mesh (regardless of type) acts as a "band-pass filter" for frequencies that may be available in a Fourier sense to construct the solution. This filtering of frequencies is essentially independent of the accuracy and type of numerical algorithm used to solve the governing equations. Noting that the numerical algorithm also will induce dissipation (filtering) due to algorithm error and/or will require added dissipation for stability (such as the leap-frog scheme), it is clear that the highest frequency that the mesh will resolve is seldom approached in the numerical results.

The simplest method for addressing this limitation is to selectively (locally) reduce mesh spacing in model regions where phenomena of interest are occurring or expected, thereby increasing the frequency spectrum that can be resolved by the mesh. If the added dissipation is scaled by the mesh spacing (algorithm error is scaled by the mesh if the algorithm is consistent) then the solution will be better resolved, i.e. more frequencies will be available in a Fourier sense to construct the solution . Many mesoscale models locally reduce mesh spacing through sequential nesting where high resolution is expected to be needed. In some cases, the nests can also be made to follow the path of the target feature. However, nesting has its own fundamental limitations. The first is inherent inflexibility in that the hierarchy of nests provides only one resolution within each level of nest, whether the resolution provided is locally needed or not. Also the nest boundary inserts an abrupt change in resolution. Typically, some space must be provided between the target phenomena and the nest boundary to allow more well resolved features to appear. Depending on the specific code and implementation, other issues may appear where nests are used.

Optical turbulence provides a specific example, where considerable experimental effort has been expended toward expanding the data base of temperature fluctuations in the atmosphere, with accepted average profiles having been obtained for selected important locales as noted in Jumper and Beland (2000) which provides an excellent assessment of past work and the issues. Unfortunately, measuring directly the detailed state of the atmosphere is not usually possible or practical for a significant range of conditions, locations and times, especially given terrain and local condition influences. This situation points to the need for simulation/modeling tools that could be used to predict, based on inputs of terrain and local conditions, local levels of optical turbulence. However, the presently available atmospheric modeling tools do not provide a means of simulating directly the scales needed to assess local optical turbulence, due to resolution and other issues.

This fundamental resolution limitation is addressed in the present work by installing in a standard mesoscale model an *r*-refinement dynamic solution adaptive grid algorithm(DSAGA), developed for and applied initially to aerospace applications(Laflin and McRae 1996; Laflin 1997; McRae and Laflin 1999). This algorithm is automated and requires no interaction during meteorological model execution. The DSAGA begins with an initial mesh distribution (usually even). The number of initial nodes are then preserved as they are dynamically relocated in order to increase resolution of the evolv-

[*]Research Assistant Professor
[†]Professor, email: mcrae@eos.ncsu.edu
[‡]Professor

ing solution. This procedure has the advantage that the adaptation criteria that control the redistribution can be targeted to resolve one or many occurrences of a given phenomenon or to resolve more than one phenomenon. The remainder of this paper will describe the application of DSAGA to the well-known mesoscale code MM5. Transformation of the governing equations to a general coordinate system will be discussed, followed by a description of the adaptive algorithm, and of a new turbulence model developed for application to optical turbulence. Initial applications will be presented.

2. Modification to MM5

The first component of the current research is the well-developed regional scale meteorological/atmospheric model MM5. This model in its standard form uses an evenly-spaced computational grid and provides the option of increased local resolution through three levels of grid nesting. The transmittal of information between nests by shared grid nodes and other coding decisions dictate a factor of three maximum reduction of cell dimension achievable through this nesting. When considered in view of the usual regional scale grid spacing of 5 to 10 km, it is clear that the resulting node spacing of 1.5 km does not provide the resolution needed for the characterization of optical turbulence or gravity waves without modification or enhancement. One of the major research tasks is to provide the needed resolution within the MM5 framework.

The current code structure and physical modeling in MM5 is retained. A 3-D grid domain (nest) is imbedded in the inner MM5 nest with increased mesh node density and with the NCSU dynamic solution adaptive mesh algorithm DSAGA (Laflin and McRae 1996; Laflin 1997; McRae and Laflin 1999) applied to further increase resolution where needed. The flow in this imbedded region is solved, to LES scales, with the standard non-hydrostatic equations transformed to a general time-varying structured grid. DSAGA uses *r*-refinement adaptation to resolve automatically selected scales and features in the solution. The goal of the adaptation is to provide LES-scale resolution for the developing turbulence. This capability will also allow resolution of the dynamic processes from gravity wave generation to break-down. This adaptive algorithm is developed sufficiently such that any criteria or linear combination of criteria may be used to promote grid clustering, thereby insuring resolution of the features from which the criteria are derived.

Within the dynamically resolved nest, an extension of k–ζ (enstrophy, or variance of vorticity) turbulence model(Robinson et al. 1995; Xiao et al. 2004) is used to provide LES sub-grid scale turbulence modeling. This model includes fundamental flow physics and has proved to be superior to standard models when used in modeling turbulence.

The outer MM5 and imbedded high-resolution fields are executed alternately. After each iteration on the outer MM5 domain, the boundary values and tendencies for the imbedded mesh are interpolated from the outer MM5 domain. Temporal accuracy is preserved by advancing the solution in the imbedded domain to the same time level as the outer domain. This procedure is the same as MM5. But the time step in the imbedded domain may vary due to the grid adaptation.

The rest of this paper will be developed as follows: the governing equations and some discretization issues will be given in Section 3. Section 4 describes the modification of MM5's semi-implicit solver. The three key components of DSAGA algorithm, namely the weight function, grid node repositioning and the solution redistribution will be described in Section 5. Section 6 contains the description of the k–ζ turbulence model. The results of a 2D and a 3D test case are presented in Section 7, followed by concluding remarks in Section 8.

3. Governing Equations

The non-hydrostatic governing equations in MM5(Grell et al. 1995), defined in the x, y, σ coordinate system, are transformed in all three dimensions to a uniform computational coordinate system, using the chain-rule according to:

$$\begin{aligned} \tau &= t \\ \xi_i &= \xi_i(x, y, \sigma, t), \quad i = 1, 2, 3 \end{aligned} \tag{1}$$

where σ is the nondimensional pressure coordinate. The resulting equations read

$$\frac{\partial \mathbf{U}}{\partial \tau} + \frac{\partial \mathbf{U}}{\partial \xi_i}\frac{\partial \xi_i}{\partial t} + m^2 \frac{\partial \mathbf{E}}{\partial \xi_i}\frac{\partial \xi_i}{\partial x} + m^2 \frac{\partial \mathbf{F}}{\partial \xi_i}\frac{\partial \xi_i}{\partial y} + \frac{\partial \mathbf{G}}{\partial \xi_i}\frac{\partial \xi_i}{\partial \sigma} = \mathbf{S} \tag{2}$$

where

$$\begin{aligned} \mathbf{U} &= [p^*p', p^*u, p^*v, p^*w, p^*T]^T, \\ \mathbf{E} &= \frac{u}{m}[p^*p', p^*u, p^*v, p^*w, p^*T]^T, \\ \mathbf{F} &= \frac{v}{m}[p^*p', p^*u, p^*v, p^*w, p^*T]^T, \\ \mathbf{G} &= \dot{\sigma}[p^*p', p^*u, p^*v, p^*w, p^*T]^T, \end{aligned} \tag{3}$$

m is the map scale, p^* the reference pressure, and p' the pressure perturbation, T the temperature, and u, v and w are velocity components in the x-, y- and z-direction, respectively. All other terms, such as pressure gradient, Coriolis force, and gravity terms are included in **S**, c.f. Grell et al. (1995) for more details. The above equations are discretized in the Arakawa-B(Arakawa and Lamb 1977) type staggered grid, using the same finite difference stencils as MM5, e.g. the stencils used in the x-direction in MM5 are applied to ξ_1 direction here. These equations are also solved using the leap-frog scheme. In order to obtain accurate discretization in the curvilinear staggered grid, three sets of metric derivatives are calculated to be consistent to the differencing of flow variables. The variables and metric derivatives are defined at three different locations as shown in Fig. 1, i.e., the cell center(cross, for p' and T), the center of cell edges(dot, for u and v) and the center of $\xi_3 = const$ cell surface(Δ, for w and $\dot{\sigma}$).

4. Modification of Semi-implicit scheme

In MM5, in order to remove the limitation on the time step due to small mesh spacing in the vertical direction, the following two coupled equations for w and p' are solved implicitly:

$$\frac{\partial w}{\partial t} - \frac{\rho_0 g}{\rho p*}\frac{\partial p'}{\partial \sigma} + \frac{g}{\gamma}\frac{p'}{p} = S_w \quad (4)$$

$$\frac{\partial p'}{\partial t} - \frac{\rho_0 g \gamma p}{p*}\frac{\partial w}{\partial \sigma} - \rho_0 g w = S_{p'} \quad (5)$$

This results in a tridiagonal system along the σ-direction for w' in the *uniform* mesh, which can be solved directly. But in the adaptive mesh, the σ variation is transformed as $\partial_\sigma = \xi_{i,\sigma}\partial_{\xi_i}$, where $\xi_{i,\sigma}$ stands for $\frac{\partial \xi_i}{\partial \sigma}$. Therefore, Eqs(4) and (5) must be solved iteratively. The iteration scheme chosen is as follows:

1. $p'^{(0)} = p'^t$, $w^{(0)} = w^t$;
2. solve the following system iteratively:

$$\frac{w^{(i+1)} - w^t}{\Delta t} - \frac{\rho_0 g}{\rho p*}(\frac{\partial p'}{\partial \xi_3})^{(i+1)}\xi_{3,\sigma} + \frac{g}{\gamma}\frac{p'^{(i+1)}}{p}$$
$$= S_w + \frac{\rho_0 g}{\rho p*}[(\frac{\partial p'}{\partial \xi_1})^{(i)}\xi_{1,\sigma} + (\frac{\partial p'}{\partial \xi_2})^{(i)}\xi_{2,\sigma}] \quad (6)$$

$$\frac{p'^{(i+1)} - p'^t}{\Delta t} - \frac{\rho_0 g \gamma p}{p*}(\frac{\partial w}{\partial \xi_3})^{(i+1)}\xi_{3,\sigma} - \rho_0 g w^{(i+1)}$$
$$= S_{p'} + \frac{\rho_0 g \gamma p}{p*}[(\frac{\partial w}{\partial \xi_1})^{(i)}\xi_{1,\sigma}$$
$$+(\frac{\partial w}{\partial \xi_2})^{(i)}\xi_{2,\sigma}] \quad (7)$$

3. when converged, $p'^{t+1} = p'^{(i+1)}$ and $w'^{t+1} = w'^{(i+1)}$

where t and $t+1$ denotes two time levels, respectively, "(0)" the initial values and i the number of iterations. For simplicity, $p'^{(i)}$ and $w^{(i)}$ are used to illustrate the algorithm. However, they are implemented as an averaged value of $p'^{(i)}$ and p'^t , and $w^{(i)}$ and w^t, respectively(Grell et al. 1995). Note that the Eqs(6) and (7) are in a form similar to Eqs(4) and (5) so that the tridiagonal solver can be applied to the ξ_3-direction. To implement this semi-implicit solver in transformed space, an outer loop is added to the original MM5 loop to update the RHS of the above equations. For the two-dimensional test case, the residual, $||w^{(i+1)} - w^{(i)}||_2/||w^{(1)} - w^{(0)}||_2$ can converge to 10^{-3} in 6–7 iterations.

5. Adaptive Grid Algorithm

The dynamic solution adaptive mesh algorithm(DSAGA) (Laflin and McRae 1996; Laflin 1997; McRae and Laflin 1999) is installed such that r-refinement adaptation is performed to increase resolution of selected features and/or regions of the computational domain in a solver-independent manner. This technique called *r*-refinement adaptation maintains the number of nodes fixed and relocates them dynamically as the node distribution needed to resolve the solution changes. A weight function based on the solution features/properties chosen for inincreased resolution guides the relocation. In this procedure, the governing equations are solved in the following four steps:

1. In the fixed grid first solve

$$\frac{\partial \mathbf{U}}{\partial t} = \mathbf{S} - \left(m^2 \frac{\partial \mathbf{E}}{\partial \xi_i}\frac{\partial \xi_i}{\partial x} + m^2 \frac{\partial \mathbf{F}}{\partial \xi_i}\frac{\partial \xi_i}{\partial y} + \frac{\partial \mathbf{G}}{\partial \xi_i}\frac{\partial \xi_i}{\partial \sigma} \right)$$

(8)

to advance the solution: $\mathbf{U}^t \rightarrow \mathbf{U}^{t+1}$;

2. A weight function is computed based on $\mathbf{U}^{t+1}$ obtained from the last step;
3. Move the grid using weight function and center of mass equation;
4. Redistribute $\mathbf{U}$ on newly generated grid by solving

$$\frac{\partial \mathbf{U}}{\partial \tau} + \frac{\partial \mathbf{U}}{\partial \xi_i}\frac{\partial \xi_i}{\partial t} = \frac{\partial \mathbf{U}}{\partial t} \tag{9}$$

For the results presented, the weight function is based on the magnitude of vorticity to improve the resolution of shear layers in atmospheric flows. The weight function is defined as

$$W = |\nabla \times \vec{V}| \tag{10}$$

After calculating the values of weight function for all grid cells, they are restricted by

$$W = \max(W, a_1 W_{ave}) \tag{11}$$
$$W = \min(W, a_2 W_{ave}) \tag{12}$$

where W_{ave} is the average value of W on the entire domain, and a_1 and a_2 are two coefficients to prescribe the floor and ceiling values of weight function, respectively. Then the weight functions are smoothed using an elliptic smoother(Benson and McRae 1991) in order to promote mesh smoothness. Next the resulting weight functions at each cell are rescaled as

$$\tilde{W}_{i,j,k} = \frac{1}{r} + (1 - \frac{1}{r})\frac{W_{i,j,k} - W_{min}}{W_{max} - W_{min}} \tag{13}$$

where r is a user provided ratio to control $\tilde{W}_{max}/\tilde{W}_{min}$ to prevent grid over-refinement.

Based on the rescaled weight function $\tilde{W}$ the new grid node location is calculated from the center of mass equation(CME):

$$P_{i,j,k} = \frac{\sum_{n=1}^{nadj} \tilde{W}_c P_c}{\sum_{n=1}^{nadj} \tilde{W}_c} \tag{14}$$

where the $P_{i,j,k}$ is the coordinate location of a grid node, $\tilde{W}_c$ and P_c are the weight function and the coordinate location at its neighboring cell centers, and $nadj$ is the number of adjacent cells surrounding node (i,j,k). One may choose physical coordinates,(x, y, σ) in Eq(14). But as pointed out in Laflin (1997), it can cause grid-line cross-over when the center of mass is located outside of the domain at the boundary with sharp curvature. Laflin(Laflin 1997) chose a local evolving parametric space to resolve this problem and to move grid nodes efficiently. In his procedure, the cell center coordinates on the RHS of Eq(14) are always initialized from the current(therefore evolving) local uniform computational space. In the current application, we found that this procedure can cause over-refined and unsmooth grid in the 2D test . Therefore, we choose to use the coordinates in a non-evolving computation space, $(\tilde{\xi}_1, \tilde{\xi}_2, \tilde{\xi}_3)$, as an alternative, where

$$\tilde{\xi}_i(x, y, z) = \xi_i(x, y, z, 0), \quad i = 1, 2, 3 \tag{15}$$

Therefore, ($\tilde{\xi}_1$, $\tilde{\xi}_2$, $\tilde{\xi}_3$) are initialized as

$$(\tilde{\xi}_1^{(0)}, \tilde{\xi}_2^{(0)}, \tilde{\xi}_3^{(0)})_{i,j,k} = (i, j, k); \tag{16}$$

With this choice, we can use symmetric Gauss Seidel(SGS) in $\tilde{\xi}_i$ space to solve CME and advance rapidly the mesh corresponding to a set of weight function. If $\tilde{W}$ is uniform, the initial grid will not be changed. Since $\tilde{W}$ is an invariant function of x, y and σ, it should be updated at each SGS iteration. In practical computation, $\tilde{W}$ is frozen for every 5 SGS iterations to reduce the overhead involved in calculating and smoothing weight functions. As will be shown in the result section, the quality of the mesh is not deteriorated significantly by this method. The refined mesh can track the high vorticity region with clustered grid nodes.

If $\tilde{\xi}_i$ coordinates are used in CME, the grid node displacement in the units of the current computational space, $\xi_i^{(n)}$, is obtained by

$$\Delta\xi_i = \frac{\partial \xi_i}{\partial \tilde{\xi}_j}(\tilde{\xi}_j^{(n+1)} - \tilde{\xi}_j^{\ (n)}) \tag{17}$$

where the superscript "(n)" and "$(n+1)$" denote the quantities on the current grid (step 1) and the new grid, respectively. If Laflin's proceduce(Laflin 1997) is used, then $\Delta\xi_i$ is a direct output from CME.

To obtain the solution in the new grid in physical space, Eq(9) is discretized as

$$\begin{aligned}&\mathbf{U}_{i,j,k}^{t+1}(x^{(n+1)}, y^{(n+1)}, z^{(n+1)}) = \\ &\mathbf{U}_{i,j,k}^{t+1}(x^{(n)}, y^{(n)}, z^{(n)}) + \frac{\partial \mathbf{U}}{\partial \xi_l}\Delta\xi_l\end{aligned} \tag{18}$$

where $\mathbf{U}^t$ and the time step cancel and $\Delta\xi_l$ is the displacement of a grid node in the current computational space:

$$\begin{aligned}\Delta\xi_l &= \xi_l(x^{(n+1)}, y^{(n+1)}, z^{(n+1)}) - \\ &\quad \xi_l(x^{(n)}, y^{(n)}, z^{(n)})\end{aligned} \tag{19}$$

Eq(18) can be regarded as a linear advection equation with a transport velocity $\Delta\xi_l$. The 5th order WENO Lax-Friedrich upwind scheme(WENO-LF-5)(Shu 1997) is employed here to solve Eq(18) to redistribute solutions, which is similar to the interpolation scheme used in MM5 for mesh refinement(developed by Smolarkiewiz and Grell(Smolarkiewicz and Grell 1992)).

In practical computation, the $\Delta\xi_l$ is limited by

$$\Delta\xi_l = \Delta\xi_l * 0.5/\max(0.5, \Delta\xi_{l,max}) \tag{20}$$

when $\Delta\xi_i$ is larger than 0.5 to prevent the numerical instabilty due to using large $\Delta\xi_l$ in Eq(18). This limitation step only takes place at the first a few iterations when the grid is preadapted based on the initial flow field. When this happens, $\tilde{\xi}^{(n+1)}$ needs to be recomputed based on $\tilde{\xi}^{(n)}$ and $\Delta\xi_i$ using the interpolation scheme.

A pseudo code summarizing the entire procedure is provided below:

1. Advance the solution M iterations on current mesh
2. To reposition the grid nodes:
   a) calculate the weight function W and smooth it;
   b) solve the center of mass equations (Eq(14)) using SGS for 5 iterations;
   c) solve Eq(17) for displacement of grid nodes in computational space if $\tilde{\xi}$ is used in CME;
   d) calculate the new location of grid nodes in physical space, redistribute the solution and calculate the metric derivatives;
   e) repeat above step(a)-(d) N times as needed.
3. goto Step 1

6. Turbulence Closure

The k–ζ two-equation turbulence mode(Robinson et al. 1995) is employed in this study. This model is based on the exact equations that govern the variance of velocity, i.e. the kinetic energy of the fluctuations (the k equation) and the exact equation that governs the variance of vorticity, i.e. the enstrophy or the ζ equation. The distinguishing feature of this model is that the length scale equation, i.e. the ζ equation, is derived from an exact equation and is not an assumed relation, such as the ϵ equation. This has led to accurate determination of free shear layer flows.

To deal with the variable resolution in the adaptive mesh, a hybrid RANS/LES approach(Xiao et al. 2004) is used such that LES subgrid model and RANS model can be automatically selected depending on the turbulence length scale and the mesh spacing. The k-equation for this approach is given by:

$$\begin{aligned}\frac{\partial\rho k}{\partial t} + \nabla\cdot(\rho\vec{V}k) = \nabla\cdot\left[\left(\frac{\mu}{3}+\frac{\mu_t}{\sigma_k}\right)\nabla k\right] + \bar{\bar{\tau}} :: \bar{\bar{S}} \\ -\frac{\mu_t g}{Pr_t\theta}\frac{\partial\theta}{\partial z} + (1-\Gamma)\left(\frac{1}{C_k}\frac{\mu_t}{\bar{\rho}^2}\nabla\rho\cdot\nabla p + C_1\frac{\rho k}{\tau_\rho} - \mu\zeta\right) \\ -\Gamma C_d\rho\frac{k^{3/2}}{\Delta}\end{aligned} \tag{21}$$

where $\bar{\bar{\tau}}$ is the turbulent stress tensor, $\bar{\bar{S}}$ the deformation rate tenor, C_1, C_k and C_d are the model constants, Γ the blending function, and the Δ the minimum mesh spacing. In the above equation, the temperature fluctuation term is simulated by $\mu_t/Pr_t\frac{\partial\theta}{\partial z}$. The Pr_t is the turbulent Prandtl number. The model with temperature variation included is under development. Preliminary results can be found in Xiao et al. (2005). The ζ-equation in the hybrid scheme remains unchanged.

When $\Gamma = 0$, Eq(21) recover the RANS model. $\Gamma = 1$ turns Eq(21) into a LES subgrid model. The eddy viscosity in Eq(21) is defined by

$$\mu_t = (1-\Gamma)\mu_{t,RANS} + \Gamma\mu_{t,LES} \tag{22}$$

where RANS eddy viscosity is given by

$$\mu_{t,RANS} = C_\mu\frac{\rho k^2}{\nu\zeta}, \tag{23}$$

and LES eddy viscosity is given by

$$\mu_t = C_s\rho\sqrt{k}\Delta. \tag{24}$$

The C_s and C_μ are model constants. When production balances dissipation at $\partial_z\theta = 0$ in the LES region, Eq(21) yields a Smagorinsky-type eddy viscosity (Xiao et al. 2004). The following blending function developed in Xiao et al. (2004),

$$\Gamma = \tanh\left(\frac{l_\epsilon}{\alpha_1\Delta}\right)^2, \tag{25}$$

is chosen to bridge the LES subgrid model and the RANS model(Xiao et al. 2004), where

$$l_\epsilon = \frac{k^{3/2}}{\nu\zeta} \tag{26}$$

and α_1 is a model constant. The transition between LES model and RANS model can occur within the log layer with this blending function.

7. Results and Discussion

7a. *2D Case*

A two-dimensional case, a flow over a generic mountain, is used to investigate the ability of DSAGA to resolve shear layers, gravity waves and their breakdown. This case is the same as that used in Doyle et al. (2000) for evaluating the wave breaking prediction for various models. For this case, the current adaptive flow solver is executed on the full domain. The mountain profile is prescribed by a witch of Agnesi curve

$$z_s(x) = \frac{ha^2}{x^2 + a^2}, \tag{27}$$

where $h = 2$ km is the mountain height and $a = 10$ km is the mountain half width. In order to compare with those results in Doyle et al. (2000), the initial conditions are the same as theirs – horizontally homogeneous with the initial flow profile provided from the upstream sounding data of 11 January 1972 Grand Junction, Colorado, as shown in Fig. 2. The same uniform grid is used to initialize the computation, which has 221×126 nodes with $\Delta z \approx 200$m and $\Delta x = 1$km. Free slip boundary conditions are applied to the lower boundary. Orlanski radiation boundary conditions(Orlanski 1976) are applied to both inflow and outflow boundaries. A 5 km sponge layer is set at the top of the domain as a boundary condition to prevent the reflection of gravity wave from the top boundary. A fourth order damping term is added to momentum and temperature equations to stabilize the flow solver. For example, the damping term in the x-momentum equation is:

$$D(u) = \frac{\epsilon}{\Delta t_{loc}} p^*(u_{i-2,j} - 4u_{i-1,j} + 6u_{i,j} - 4u_{i+1,j} + u_{i+2,j}) \tag{28}$$

where $\epsilon = 3 \times 10^{-3}$ and Δt_{loc} is local time scale varied with local mesh spacing in the ξ_1-direction. It is implemented as

$$\Delta t_{loc} = \frac{CFL}{u\xi_{1,x} + v\xi_{1,y} + w\xi_{1,z} + a * A_1}, \tag{29}$$

where $CFL = 0.8$ is a safety factor to control the time step , and

$$A_1 = \sqrt{\xi_{1,x}^2 + \xi_{1,y}^2 + \xi_{1,z}^2}, \tag{30}$$

The global time step is chosen as the minimum value of Δt_{loc}.

In the following results, the grid adaptation is started from $t = 0$. The grid is preadapted from the uniform mesh by $N = 100$ adaptation cycles in order to resolve the initial flow field. The $N = 3$ is then used for grid adaptation after every 4 iterations($M = 4$). M and N are the parameters that appear in the pseudo code in Section 5. The weight function ratio $r = 15$ is used in Eq(13) throughout the simulation. The time-independent $\tilde{\xi}_i$ coordinates are used in the CME. Two sets of results are obtained: one without turbulence modeling, the other with our hybrid RANS/LES model. For the former, the inherent numerical damping in the adaptive solver provides dissipation mechanism for turbulence, which can be considered as a Monotonically Integrated LES(MILES) simulation(Boris et al. 1992). The monotonicity is introduced from the solution distribution step because of the WENO-LF-5 scheme. For the hybrid RANS/LES simulation, some tests showed that the hybrid scheme cannot switch from RANS to LES and turbulent kinetic energy(TKE) dissipated very quickly, which is probably due mainly to the coarse mesh spacing in the near wall region and the free slip boundary condition. Therefore, for the RANS/LES simulation, $\Gamma = 1$ is used. Thus only the LES component in the hybrid model is used. A new blending function that can control RANS/LES switching outside boundary layer needs to be developed in the future.

Figure 3 shows the potential temperature contours(θ), or the isotropes, for MILES simulation on different meshes. Compared to the uniform mesh solution, the isotropes on the adaptive mesh clearly show the dynamic process of gravity wave overturning and break-down to turbulent eddies at $18 < z < 20$ km and $13 < z < 16$ km. Note that the breakdown is occurring in two distinct shear layers. Similar RANS/LES solutions are presented in Fig.4. The coherent turbulent structure at upper level atmosphere is essentially absent from the uniform mesh solutions because the 1 km mesh spacing is not able to resolve the 1.5 km long eddies at $z \approx 20$ km. As shown in Fig. 5, the wave break-down is closely correlated with the regions of high vorticity (thus the choice of vorticity for the weight function). As a result of this, the grid nodes cluster around the wave breaking region and provide local resolution of $\Delta x \approx 300$ m and $\Delta z \approx 50$ m for resolving the small eddies.

The potential temperature contours at $t = 3$ h for MILES and RANS/LES on different meshes are shown in Fig.6. Their corresponding meshes are presented in Fig. 7. Two bands of highly refined mesh at $z \approx 13$ km and 20 km above the mountain provide better resolution on small eddies within those regions. Another region of refined grid is aligned with the vortex street above the lee side of the mountain gives better resolution of lower-tropospheric eddies in front of the hydraulic jump, which is shown in more detail in Fig. 8. A large eddy on the top of the hydraulic jump in the RANS/LES solution in Fig. 6(c) shows a significant difference compared to the MILES solution, which is probably due to the enhanced diffusion resulting from the k-equation. This eddy is so strong that it attracts many surrounding grid nodes. When shown in the rescaled plot in Fig. 9, this eddy is similar to the downward wave breaking predicted by ARPS and RIMS in (Doyle et al. 2000).

7b. *3D Case*

In this case, the adaptive algorithm is integrated into the MM5 code to simulate an atmospheric flow over a west coastal mountainous area. Fig. 10 presents the layout of the 3-level nested grid with the terrain contours for this simulation. MM5 is employed in three levels. Its solution in level 3 provides the boundary conditions for the current adaptive flow solver. The initial grid for the adaptive solver is the same as the level 3 grid with $121 \times 121 \times 81$ nodes. The initial uniform grid with mesh spacing $\Delta x = 5km$ is preadapted to the initial flow field. Again the vorticity is used as the weight function.

To advance the solution to the same level as the outer MM5 solution, the time step in the imbedded adaptive mesh is determined by:

$$\Delta t = \frac{\Delta t_{MM5}}{NINT(\frac{\Delta t_{MM5}}{\min(\Delta t_{loc})} + 0.5)} \tag{31}$$

where Δt_{MM5} is the time step on the outer MM5 domain, Δt_{loc} is local time step defined in Eq(29), NINT is a Fortran function for nearest integer.

Figure 11 shows the evolution of the grid on the surface. The contours for the magnitude of weight function are also shown in the same figure. As can be seen from the grid node distribution, the grid nodes distribute at different location at different time. Fig.12 gives a detailed 3-D view of the grid system in the high resolution region ($t = 11h$), with the contours of potential temperatures on the vertical grid surface. The contours show that a hydraulic jump occurs on the lee side of the mountain, which is well captured/resolved by the fine grid in that region. The grid lines on the vertical surface clearly show that the grid adaption is truely three dimensional.

8. Concluding Remark

The non-hydrodynamic MM5 governing equations are transformed into a general curvilinear coordinate system. The original MM5 semi-implicit flow solver is modified accordingly. With these modifications and the integrated NCSU dynamic dynamic solution adaptive mesh algorithm(DSAGA), a new adaptive atmospheric flow solver has been developed. In the current DSAGA scheme, a non-evolving parametric space is used to prevent the grid cross-over and over refinement problems. The WENO-LF-5 5th order upwind scheme is used to redistribute the solution and calculate the new grid node locations. A hybrid RANS/LES model has been incorporated into the new flow solver to model different scales of turbulence.

A two dimensional flow with a realistic inflow profile over a mountain prescribed by Agnesi curve is used to test the current flow solver. The results obtained shows that the vorticity-based weight function is capable of clustering grid nodes in wave breaking regions and vortex streets with a minimum vertical mesh spacing of 50 m. The dynamic process of gravity wave breaking and the coherent structures of gravity wave induced turbulence are well resolved by the adaptive mesh. The fact that differences exist between MILES and RANS/LES indicates that as the resolution and the blending function are improved, further details of the turbulence structure will emerge.

The three-dimensional results show that the new adaptive solver is able to communicate with the outer MM5 domain and provide three dimensional grid adaptation to capture hydraulic jump in a mountainous area. Further improvement in weight function will lead to improved resolution.

In summary, the new adaptive atmospheric solver has shown its capability of providing the necessary resolution for optical turbulence modeling and has the potential to begin detailed comparison with observations in the near future.

9. Acknowledgments

This work was funded by the US Army Research Laboratory, Battlefield Environments Division, WSMR and the HELJTO through TCN 02133, Battelle Columbus Division; monitored by Dr. David Tofsted, ARLWSMR, and by the US Air Force Research Laboratory, Space Vehicles Directorate, Hanscom AFB, MA; monitored by Dr. George Jumper, AFRL/VSBYA. The authors thank Dr. Tofsted and Dr. Patrick Hanes of ARLWSMR, Dr. George Jumper and Dr. Frank Ruggiero of AFRL, and Dr. Yuh-Lang Lin and Dr. Mike Kaplan of NCSU for many helpful conversations.

References

Arakawa, A. and V. R. Lamb, 1977: Computational design of the basic dynamical processes of the ucla general circulation model. *Methods in Compu. Phys.*, **17**, 173–265.

Benson, R. A. and D. S. McRae, 1991: A solution-adaptive mesh algorithm for dynamic/static refinement of two and three dimensional grids. *Proceedings of the Third International Conference on Numerical Grid Generation in Computational Field Simulations*, Barcelona, Spain, 185–199.

Boris, J. P., F. F. Grinstein, E. S. Oran, and R. J. Kolbe, 1992: New insights into large eddy simulation. *Fluid Dynamics Research*, **10**, 65–90.

Doyle, J. D., D. Durran, C. Chen, B. A. Collie, M. Georgelin, V. Grubisic, W. R. Hsu, C. Y. Huang, D. Landau, Y. L. Lin, G. S. Poulos, W. Y. Sun, D. B. Webber, M. G. Wurtele, and M. Xue, 2000: An intercomparison of model-predicted wave breaking for the 11 January 1972 boulder windstorm. *Monthly Weather Review*, **128**.

Grell, G. A., J. Dudhia, and D. R. Stauffer, 1995: A description of the fifth-generation penn state/ncar mesoscale model(mm5). NCAR TECHNICAL NOTE NCAR/TN–398+STR.

Jumper, G. Y. and R. R. Beland, 2000: Progress in the understanding and modeling of atmospheric optical turbulence. AIAA Paper 2000-2355.

Laflin, K. R., 1997: *Solver-independent r-refinement adaptation for dynamic numerical simulations*. Ph.D. thesis, North Carolina State University, Raleigh, NC.

Laflin, K. R. and D. S. McRae, 1996: Solution-dependent grid-quality assessment and enhancement. *Proceedings: 5th International Conference on Numerical Grid Generation in Computational Field Simulations*, 3–12.

McRae, D. and K. Laflin, 1999: Dynamic grid adaption and grid quality. *Handbook of Grid Generation*, J. F. Thompson, B. K. Soni, and N. P. Weatherill, eds., CRC Press,Boca Raton, 34–1 to 34–33.

Orlanski, I., 1976: A simple boundary condition for unbounded hyperbolic flows. *Journal of Computational Physics*, **21**, 251–269.

Robinson, D. F., J. E. Harris, and H. A. Hassan, 1995: Unified turbulence closure model for axisymmetric and planar free shear flows. *AIAA Journal*, **33**, 2325–2331.

Shu, C.-W., 1997: Essential non-oscllatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. ICASE Report 97-65, nASA/CR-97-206253.

Smolarkiewicz, P. K. and G. A. Grell, 1992: A class of monotone interpolation schemes. *Journal of Computational Physics*, **101**, 431–440.

Xiao, X., J. R. Edwards, H. A. Hassan, and R. A. Baurle, 2004: Inflow boundary conditions for hybrid large eddy/reynolds averaged navier-stokes simulation. *AIAA Journal*, **41**, 1481–1489.

Xiao, X., J. R. Edwards, H. A. Hassan, and R. L. Gaffney, 2005: Role of turbulent prandtl number on heat flux at hypersonic mach number. Technical Report AIAA Paper 2005-1098.

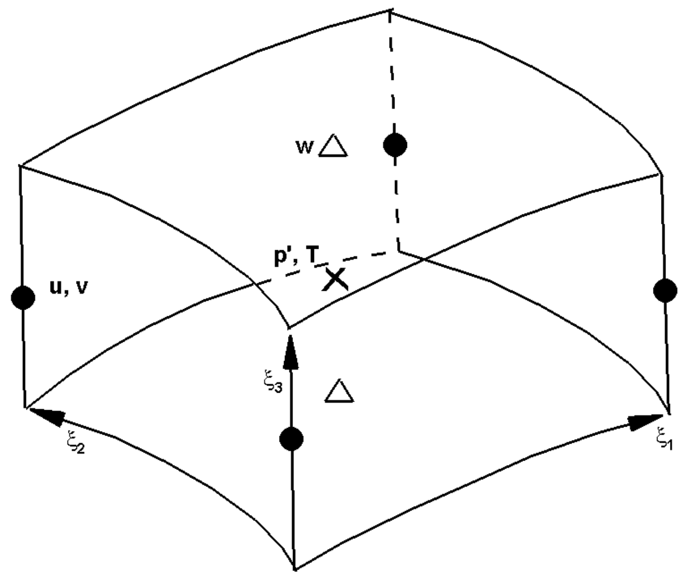


Figure 1: The location of different variables

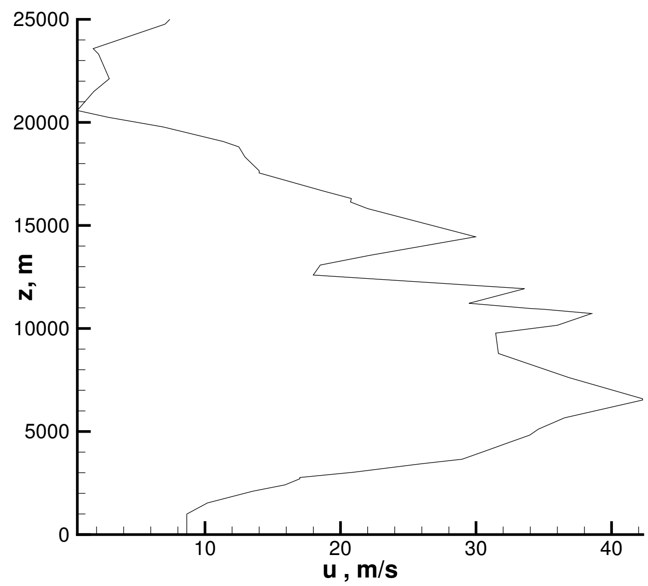


Figure 2: Inflow velocity profile from the Grand Junction, CO, sounding for 1200 UTC 11 January 1972

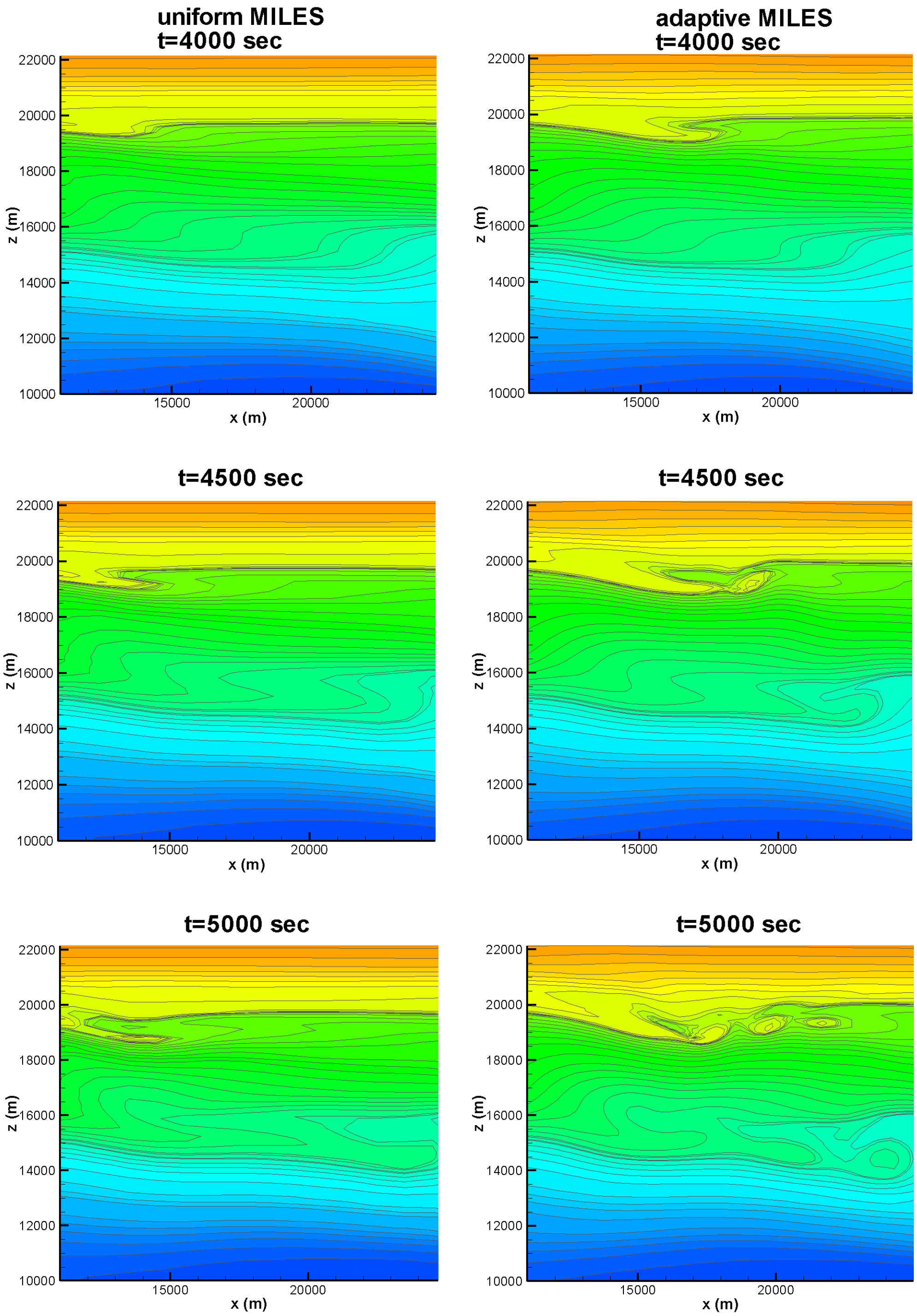


Figure 3: The potential temperature contours, MILES scheme,left column for uniform mesh,right column for adaptive mesh

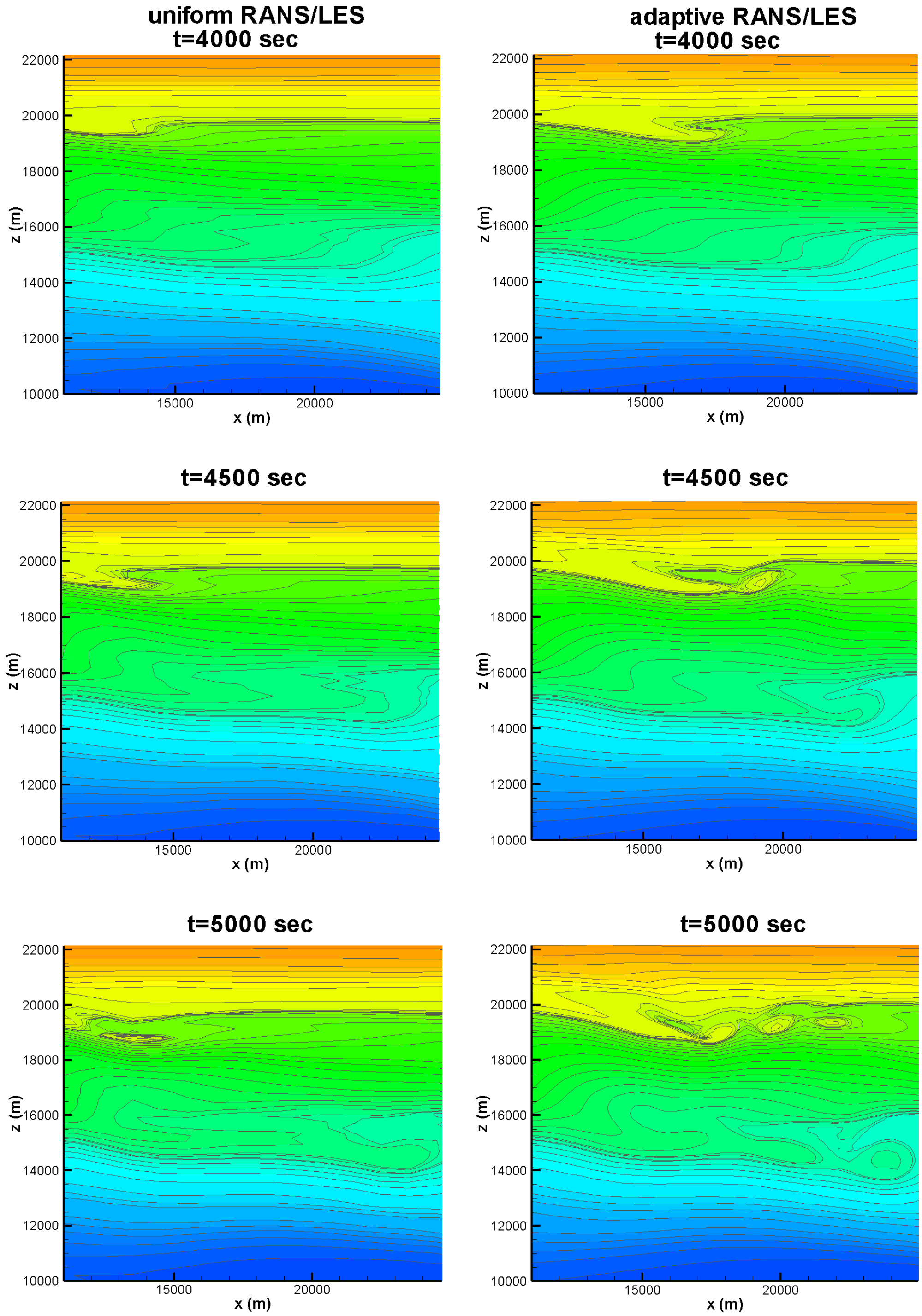
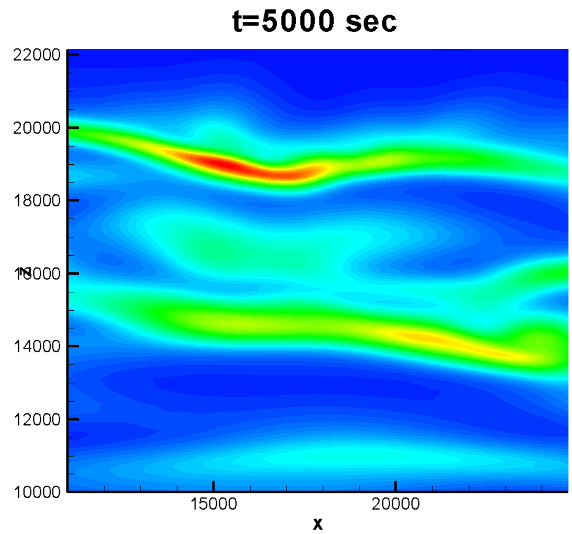
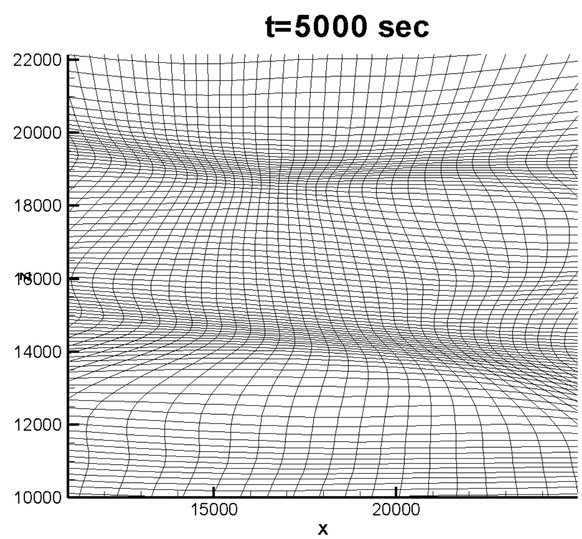


Figure 4: The potential temperature contours, RANS/LES scheme, left column for uniform mesh, right column for adaptive mesh



(a) weight function contours



(b) mesh

Figure 5: Contours of weight function and the adaptive mesh, MILES

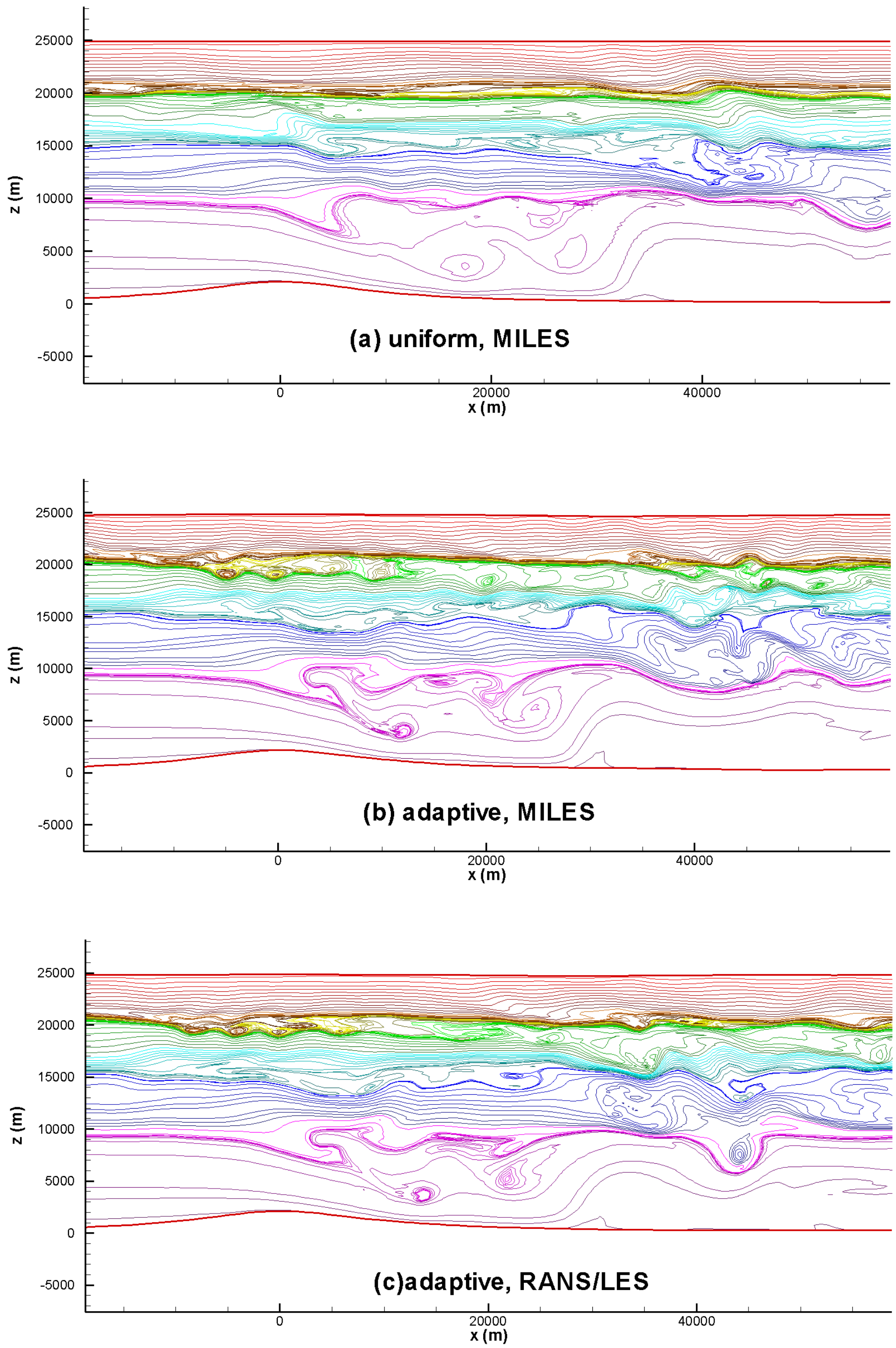


Figure 6: Comparison of potential temperature contours, t=3 h

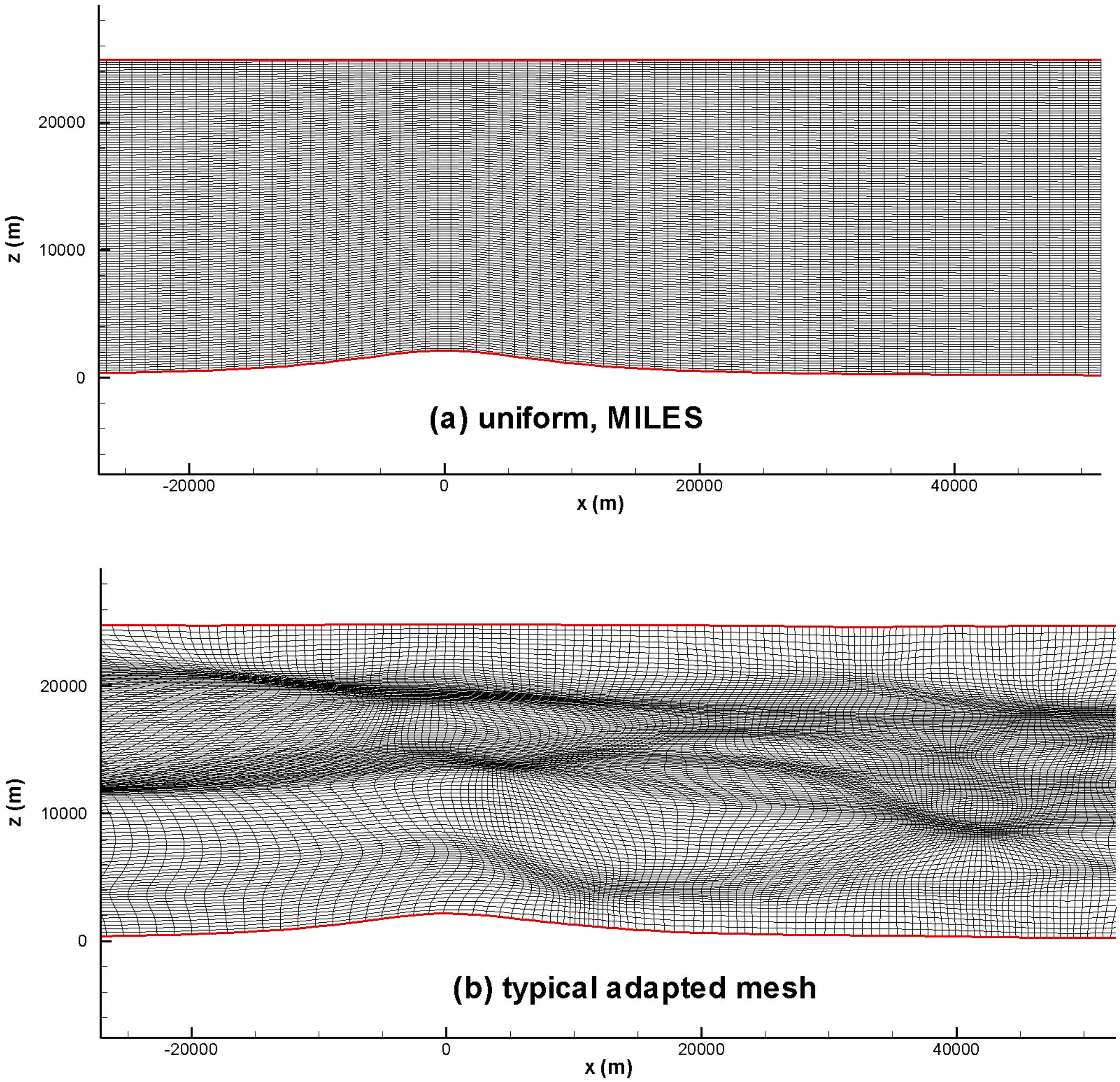


Figure 7: Comparison of adaptive meshes with uniform mesh, t= 3 h

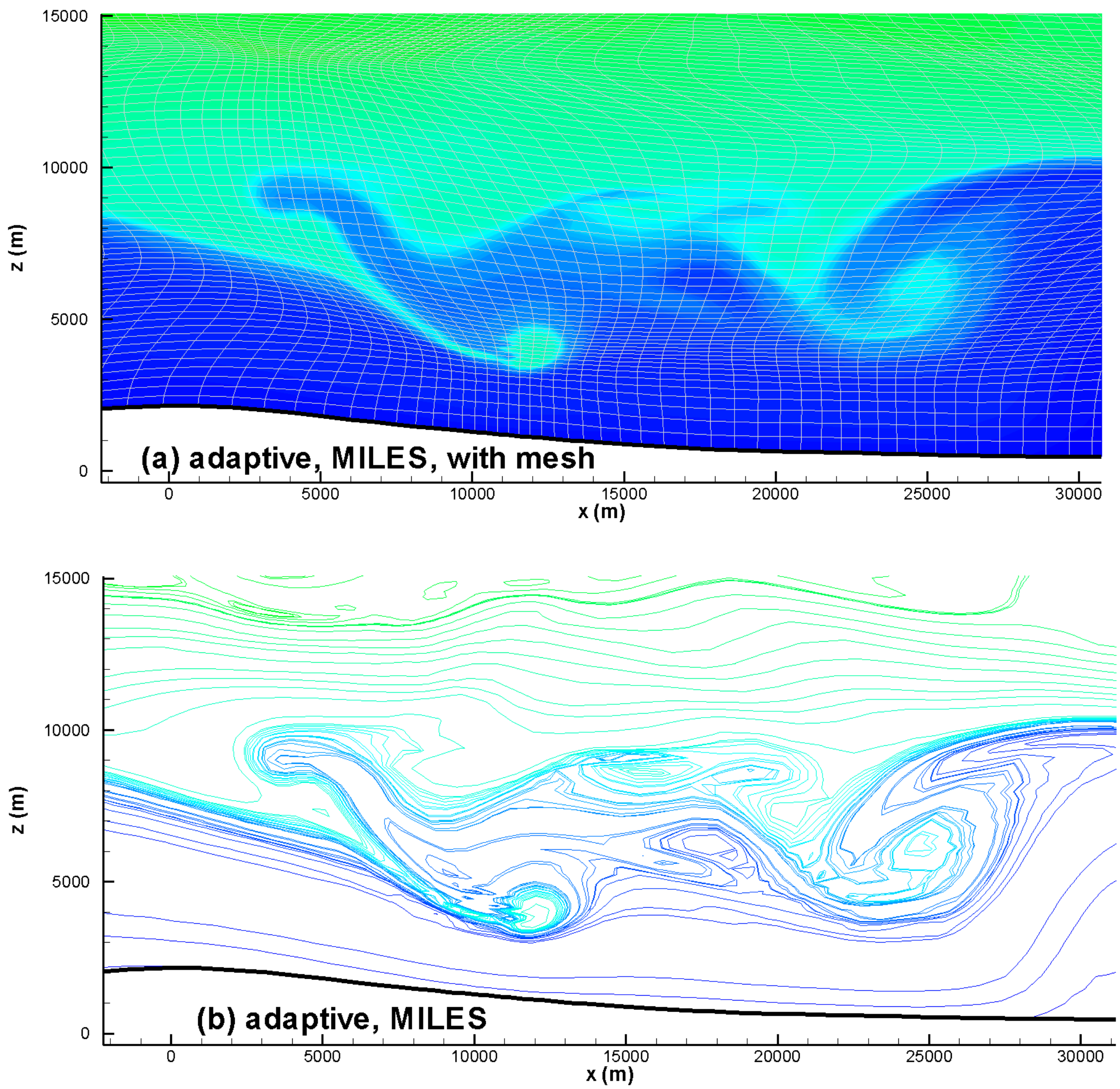


Figure 8: Potential temperature contours in the region ahead of hydraulic jump, t=3 h

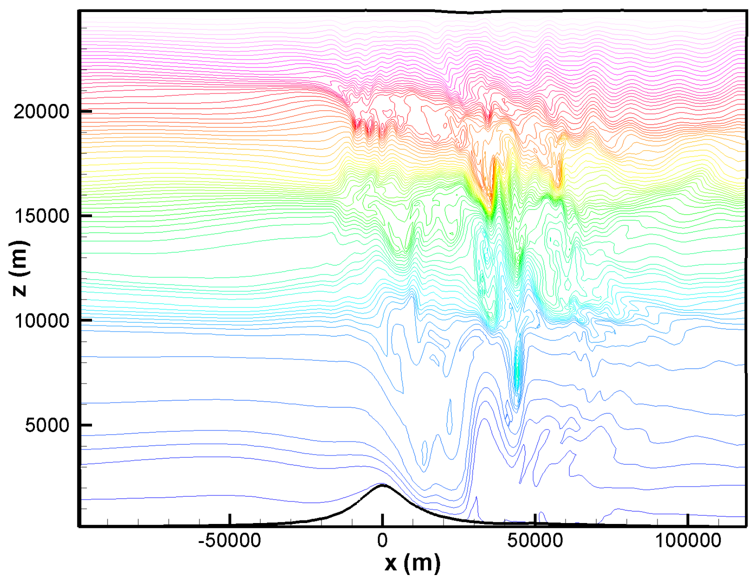


Figure 9: The contours of potential temperature in the same format as in (Doyle et al. 2000), RANS/LES, t=3h

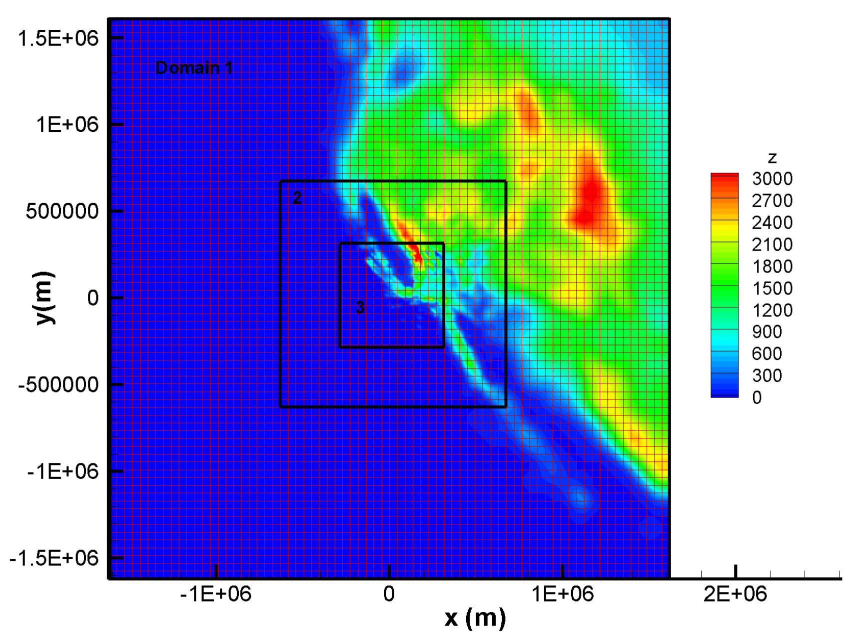
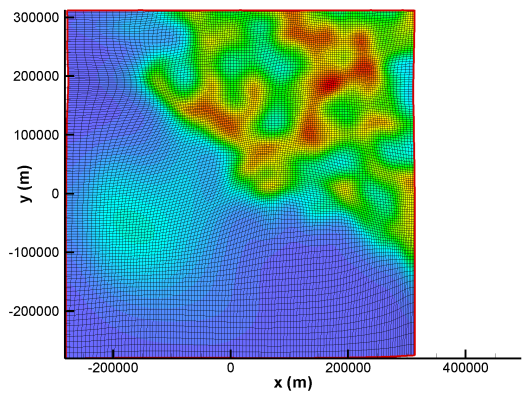
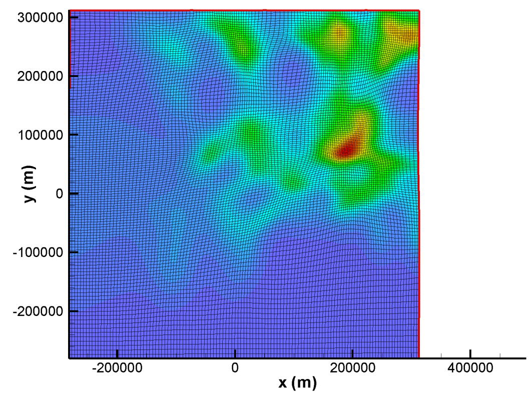


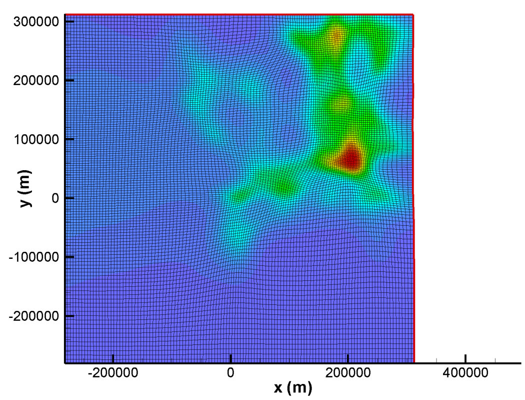
Figure 10: The layout of three-level nested grid layout and terrain contours.



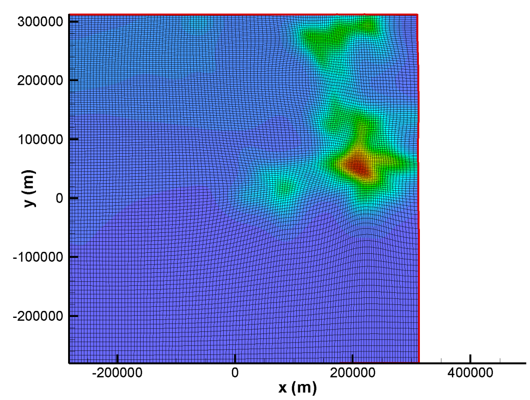
(a) t=1h



(b) t=5h



(c) t=7h



(d) t=11h

Figure 11: Grid evolution and the weight function magnitude contours on the surface

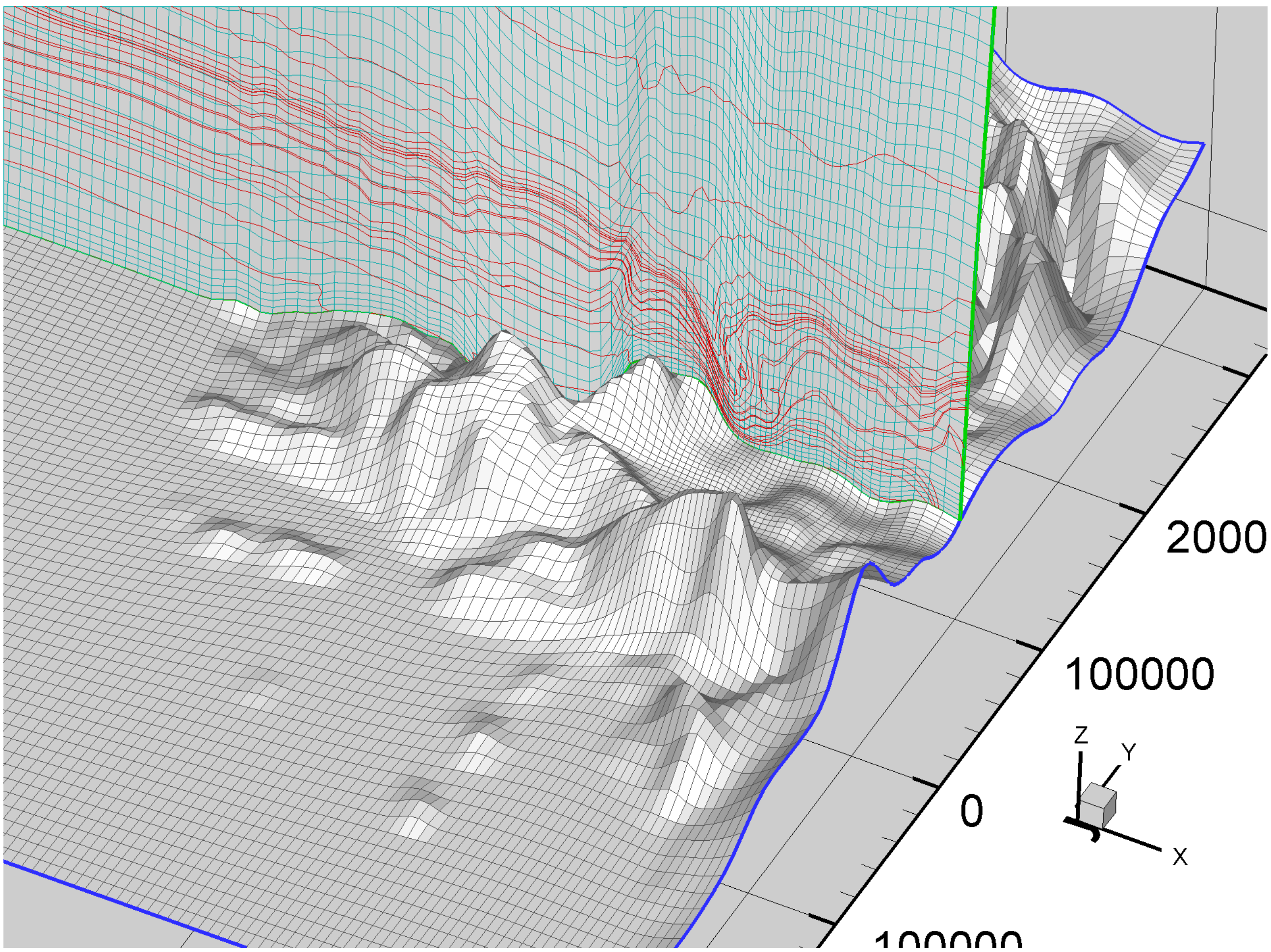


Figure 12: The potential contours and grid at $t = 11$h