

Chapter 2

Model formulation

2.1 Introduction

This chapter provides a description of the numerical model that was utilized for this study. In addition, the performance of the model is evaluated using several benchmark cases. The decision to develop a new code, rather than use an existing one, was motivated by many factors. First, it was desired to take advantage of recent developments in nonhydrostatic modeling that would improve the accuracy of the solution. To this end, the dynamic core of the model was developed around a higher order, more accurate advection scheme, as compared to those used in other available atmospheric models. This numerical technique, recently published by Wicker and Skamarock (2002), also forms the core of the new Weather Research and Forecasting (WRF) model.

New equation sets were considered, and evaluated, for the model. Ultimately, it was decided to integrate potential temperature¹ (θ) rather than temperature (T) or ice-liquid water potential temperature (θ_{il}), for reasons that are discussed in Bryan and Fritsch (2002c). Furthermore, the new model incorporates several commonly neglected but important effects that are traditionally ignored in cloud models. The use of this new equation set substantially improves the conservation of mass and energy over simulations using standard equation sets.

Additional improvements are discussed later in this chapter. Ultimately, a numerical modeling system was created that represents a significant improvement over existing systems, particularly as it applies to the simulation of moist processes. This latter point – a more accurate representation of moist processes – is crucial for this work. As discussed in Chapter 1, available observations are inadequate for answering questions about the structure and lifecycle of MAULs. Since it is necessary to utilize numerical simulations to learn about MAULs, it was deemed necessary to have a numerical model that is state-of-the-art in terms of numerics and physics, and is able to satisfy basic conservation laws, such as conservation of mass and energy.

¹ A complete list of symbols is provided in the Appendix.

2.2 Derivation of governing equations

The equations for the model were derived from an equation set governing the conservation of mass, total energy, and total momentum,

$$\frac{\partial \rho_a}{\partial t} + \frac{\partial (\rho_a u_j)}{\partial x_j} = 0 \quad (2.1)$$

$$\frac{\partial E_t}{\partial t} + \frac{\partial [(E_t + p)u_j]}{\partial x_j} = 0 \quad (2.2)$$

$$\frac{\partial M_i}{\partial t} + \frac{\partial (M_i u_j)}{\partial x_j} + \frac{\partial p}{\partial x_i} + \delta_{i3} \rho_t g = 0 \quad (2.3)$$

$$\frac{Dr_v}{Dt} = -\dot{r}_{cond} - \dot{r}_{dep} \quad (2.4)$$

$$\frac{Dr_l}{Dt} = \dot{r}_{cond} - \dot{r}_{frz} \quad (2.5)$$

$$\frac{Dr_i}{Dt} = \dot{r}_{dep} + \dot{r}_{frz} \quad (2.6)$$

where

$$\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + u_j \frac{\partial}{\partial x_j}, \quad (2.7)$$

$$E_t \equiv \rho_a \left(c_v T + c_{vv} r_v T + c_{pv} (r_l + r_i) T - L_v r_l - L_s r_i + (1 + r_t) \frac{1}{2} (u^2 + v^2 + w^2) + (1 + r_t) g z \right), \quad (2.8)$$

and

$$M_i \equiv (\rho_a + \rho_l + \rho_i) u_i. \quad (2.9)$$

In (2.3), Einstein summation convention is used, where the subscript “ $i = 1, 2, 3$ ” signifies the x , y , and z components, respectively, and δ_{ij} is the Kronecker delta. Hydrometeor fallout is neglected from these equations, and from the following derivations, but will be added to the hydrometeor equations at the end of the derivation.

2.2.1 Thermodynamic equation

An air temperature equation is derived utilizing Kirchoff's relations,

$$\frac{DL_v}{DT} = c_{pv} - c_l, \quad \frac{DL_s}{DT} = c_{pv} - c_i, \quad \frac{DL_f}{DT} = c_l - c_i \quad (2.10)$$

and the following relations,

$$c_p = c_v + R, \quad c_{pv} = c_{vv} + R_v. \quad (2.11)$$

It will be assumed here, and for the remainder of this work, that the temperatures of liquid water (T_l) and ice water (T_i) shall always be equal to the air temperature (T).

Therefore, in the derivation of the air temperature equation it is assumed that an instantaneous diffusion of heat occurs between the hydrometeors and the air. This assumption is justified if the time scale of this diffusion process is less than the time scale of typical model time steps. Bannon (2002) notes that this assumption is valid for small liquid water drops, but fails for large water drops and large ice particles (e.g., hail and graupel). The alternative to this assumption would be to integrate governing equations for T_l and T_i , and to explicitly model the diffusion of heat between hydrometeors and air. This practice has generally been avoided in meteorological models due to the computation expense involved (e.g., Hall 1980). A recent study by Walko et al. (2000) presents encouraging results using an efficient technique to include this process into a meteorological model.

In most meteorological models, the problem of heat diffusion between air and hydrometeors is ignored altogether by neglecting the heat capacities of liquid water and ice from the governing equations. The assumption of instantaneous heat diffusion made in this work is the opposite extreme to this commonly used approximation. It is argued here that the instantaneous diffusion method is, in general, the preferable extreme, since it is possible to maintain total energy conservation. Nevertheless, further research into this area is clearly warranted.

Under these assumptions, the equation governing the temperature of air can be derived using (2.1)-(2.6), yielding

$$c_{vml} \frac{DT}{Dt} = -\frac{p}{\rho_a} \frac{\partial u_j}{\partial x_j} + (L_v - R_v T) \dot{r}_{cond} + (L_s - R_v T) \dot{r}_{dep} + L_f \dot{r}_{frz}. \quad (2.12)$$

It is now possible to derive equations for a non-dimensional pressure (π) and dry air potential temperature (θ), which are solved in the numerical model, where

$$\pi \equiv \left(\frac{p}{p_{00}} \right)^{\frac{R}{c_p}}, \quad (2.13)$$

$$\theta \equiv \frac{T}{\pi}, \quad (2.14)$$

and $p_{00} = 1000 \text{ mb}$ is a reference pressure. To derive governing equations for these two variables, the equation of state,

$$p = \rho_a RT \left(1 + \frac{r_v}{\varepsilon} \right), \quad (2.15)$$

is used with (2.1) and (2.12) to derive a prognostic pressure equation,

$$\frac{D \ln p}{Dt} = -\frac{c_{pml}}{c_{vml}} \frac{\partial u_j}{\partial x_j} + \left(\frac{L_v}{c_{vml} T} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{cond} + \left(\frac{L_s}{c_{vml} T} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{dep} + \frac{L_f}{c_{vml} T} \dot{r}_{frz}. \quad (2.16)$$

Then, using (2.13) and (2.14), prognostic equations for π and θ are derived, yielding

$$\begin{aligned} \frac{D \ln \pi}{Dt} = & -\frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \frac{\partial u_j}{\partial x_j} \\ & + \frac{R}{c_p} \left[\left(\frac{L_v}{c_{vml} T} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{cond} + \left(\frac{L_s}{c_{vml} T} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{dep} + \frac{L_f}{c_{vml} T} \dot{r}_{frz} \right] \end{aligned} \quad (2.17)$$

and

$$\begin{aligned} \frac{D \ln \theta}{Dt} = & -\left(\frac{R_m}{c_{vml}} - \frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \right) \frac{\partial u_j}{\partial x_j} + \frac{c_v}{c_p c_{vml} T} (L_v \dot{r}_{cond} + L_s \dot{r}_{dep} + L_f \dot{r}_{frz}) \\ & - \frac{R_v}{c_{vml}} \left(1 - \frac{R}{c_p} \frac{c_{pml}}{R_m} \right) (\dot{r}_{cond} + \dot{r}_{dep}). \end{aligned} \quad (2.18)$$

Equations (2.17) and (2.18) are used in the present model.

For reference, and for comparison to other meteorological models, it is noted that a different form of the temperature equation can be derived by using (2.16) to eliminate the divergence term on the right-hand side of (2.12), yielding

$$\frac{DT}{Dt} = \frac{1}{\rho_a c_{pml}} \frac{Dp}{Dt} + \frac{L_v}{c_{pml}} \dot{r}_{cond} + \frac{L_s}{c_{pml}} \dot{r}_{dep} + \frac{L_f}{c_{pml}} \dot{r}_{frz}. \quad (2.19)$$

This equation is similar in form to that used in the Pennsylvania State University – National Center for Atmospheric Research mesoscale model (MM5, Dudhia 1993); this equation becomes identical to that presented in Dudhia (1993) if the specific heats of

water vapor, liquid, and ice are neglected, so that $c_{pml} \equiv c_p$. Similarly, the “traditional” potential temperature equation (i.e., that used in most of today’s meteorological models) follows directly from (2.18) assuming that $R_m \equiv R$, $c_{pml} \equiv c_p$, and $c_{vml} \equiv c_v$, yielding,

$$\frac{D \ln \theta}{Dt} = \frac{L_v}{c_p T} \dot{r}_{cond} + \frac{L_s}{c_p T} \dot{r}_{dep} + \frac{L_f}{c_p T} \dot{r}_{frz}. \quad (2.20)$$

Regarding the equation used in this model, (2.18), it is noted that the potential temperature equation has a source term even in the absence of phase changes, due to the first term on the right-hand side of (2.18). Thus, potential temperature, as defined by (2.14), is technically not a conserved variable when water (in any phase) is present – which is clearly contrary to that typically argued. An alternative definition of potential temperature that is conservative in the absence of phase changes is presented here for completeness. If an alternative definition for nondimensional pressure is defined as

$$\tilde{\pi} \equiv \left(\frac{p}{p_{00}} \right)^{\frac{R_m}{c_{pml}}}, \quad (2.21)$$

and if potential temperature is defined as

$$\tilde{\theta} \equiv \frac{T}{\tilde{\pi}}, \quad (2.22)$$

then the prognostic equations are

$$\begin{aligned} \frac{D \ln \tilde{\pi}}{Dt} = & -\frac{R_m}{c_{vml}} \frac{\partial u_j}{\partial x_j} + \left(\frac{R_m L_v}{c_{pml} c_{vml} T} - \frac{R_v}{c_{vml}} \right) \dot{r}_{cond} + \left(\frac{R_m L_s}{c_{pml} c_{vml} T} - \frac{R_v}{c_{vml}} \right) \dot{r}_{dep} \\ & + \frac{R_m L_f}{c_{pml} c_{vml} T} \dot{r}_{frz} + \ln \tilde{\pi} \frac{D}{Dt} \ln \left(\frac{R_m}{c_{pml}} \right) \end{aligned} \quad (2.23)$$

$$\frac{D \ln \tilde{\theta}}{Dt} = \frac{1}{c_{pml} T} (L_v \dot{r}_{cond} + L_s \dot{r}_{dep} + L_f \dot{r}_{frz}) - \ln \tilde{\pi} \frac{D}{Dt} \ln \left(\frac{R_m}{c_{pml}} \right). \quad (2.24)$$

The final term on the right-hand side of (2.23) and (2.24) can be expanded,

$$\frac{D}{Dt} \ln \left(\frac{R_m}{c_{pml}} \right) = \left(\frac{R_v}{R_m} - \frac{c_{pv}}{c_{pml}} \right) \frac{D r_v}{Dt} - \frac{c_{pl}}{c_{pml}} \frac{D r_l}{Dt}, \quad (2.25)$$

revealing that this term is non-zero when phase changes occur. It is also non-zero in the presence of hydrometeor fallout, due to the $\frac{Dr_l}{Dt}$ term. Nevertheless, (2.24) reveals that

$\tilde{\theta}$ is conserved in the absence of precipitation processes. It could be argued that this definition of potential temperature is preferable for use in a numerical modeling system, since it is truly conservative for mixing and advection (if phase changes are applied in a separate step, as is typically done in meteorological models). However, $\tilde{\pi}$ is not a particularly useful variable to use in the pressure gradient terms in the momentum equation; specifically, if the pressure gradient is written in terms of $\frac{\partial \tilde{\pi}}{\partial x_i}$, then additional

terms involving the spatial derivatives of r_v , r_l , and r_i would be necessary. It is also noted that a model formulated with $\tilde{\pi}$ and $\tilde{\theta}$ would be more expensive to run than a model using π and θ , and neither formulation exactly conserves mass or energy. The numerical model was coded and tested using $\tilde{\pi}$ and $\tilde{\theta}$. It was found to be noticeably more expensive than a model using π and θ , with no perceptible improvements in accuracy (e.g., in terms of mass or energy conservation, or for the benchmark tests which are presented in the next section).

2.2.2 Momentum equation

For, the momentum equation, originally expressed as

$$\frac{Du_i}{Dt} = -\frac{1}{\rho_a(1+r_i)} \frac{\partial p}{\partial x_i} - \delta_{i3}g, \quad (2.26)$$

the pressure gradient term is rewritten in terms of π , where it can be shown from (2.13) that

$$\frac{\partial p}{\partial x_i} = c_p \rho_a \theta \left(1 + \frac{r_v}{\varepsilon} \right) \frac{\partial \pi}{\partial x_i}. \quad (2.27)$$

The resulting momentum equation is,

$$\frac{Du_i}{Dt} = -c_p \theta \rho \frac{\partial \pi}{\partial x_i} - \delta_{i3}g, \quad (2.28)$$

where θ_ρ is the density potential temperature²,

$$\theta_\rho \equiv \theta \frac{(1 + r_v/\varepsilon)}{(1 + r_t)}. \quad (2.29)$$

Next, the model equations are restated in terms of a time-independent and horizontally-homogeneous base state (noted by a “0” subscript) and the perturbations from that base state (noted by prime superscripts):

$$p(x, y, z, t) = p_0(z) + p'(x, y, z, t), \quad (2.30a)$$

$$\pi(x, y, z, t) = \pi_0(z) + \pi'(x, y, z, t), \quad (2.30b)$$

$$\theta(x, y, z, t) = \theta_0(z) + \theta'(x, y, z, t), \quad (2.30c)$$

$$r_v(x, y, z, t) = r_{v0}(z) + r'_v(x, y, z, t). \quad (2.30d)$$

The base state momentum and liquid and ice water mixing ratios are assumed to be zero. By definition, the base state fields are in hydrostatic balance,

$$\frac{dp_0}{dz} = -\rho_0 g(1 + r_{v0}), \quad (2.31a)$$

which can be written in terms of π using (2.27), yielding

$$\frac{d\pi_0}{dz} = -\frac{g}{c_p \theta_{\rho 0}}. \quad (2.31b)$$

Using (2.28) and (2.31b), the final momentum equation is,

$$\frac{Du_i}{Dt} = -c_p \theta_\rho \frac{\partial \pi'}{\partial x_i} + \delta_{i3} g \left(\frac{\theta_\rho}{\theta_{\rho 0}} - 1 \right). \quad (2.32)$$

The buoyancy term [last on the right-hand side of (2.32)] is different from that used in other models. In most meteorological models, the buoyancy term usually is expressed as

$$\frac{\partial w}{\partial t} = -g \frac{\rho'}{\rho_0} \quad (2.33a)$$

$$\equiv g \left(\frac{\theta'}{\theta_0} + \left(\frac{1}{\varepsilon} - 1 \right) q'_v - r'_l - r'_i + \frac{c_v p'}{c_p p_0} \right) \quad (2.33b)$$

when the pressure gradient term is written in terms of p , or

² This variable is called *density potential temperature* because it is the potential temperature at which the same volume of dry air would have the same density (Emanuel 1994).

$$\frac{\partial w}{\partial t} \equiv g \left(\frac{\theta'}{\theta_0} + \left(\frac{1}{\varepsilon} - 1 \right) q'_v - r'_l - r'_i \right) \quad (2.33c)$$

when the pressure gradient is written in terms of π . The approximation from (2.33a) to (2.33b) is typically made in models that integrate a pressure equation. If density is integrated in the model, as in the height-coordinate core of the Weather Research and Forecasting (WRF) model, the buoyancy formulation involving density (2.33a) is usually retained, which is exact. The two approximate forms [i.e., (2.33b) and (2.33c)] are derived by invoking Taylor series approximations, thereby decreasing the accuracy of buoyancy tendency. In contrast, the buoyancy term used in this model [last on the right-hand side of (2.32)] is exact.

2.2.3 Pressure equation

A prognostic equation for π' is derived noting that π_0 is time-independent.

Furthermore, the advection of π is split into two terms to facilitate the use of a split acoustic time step (Klemp and Wilhelmson 1978; Skamarock and Klemp 1992), which will be discussed further in the next section. It is noted that

$$u_j \frac{\partial \pi}{\partial x_j} = u_j \frac{\partial \pi_0}{\partial x_j} + u_j \frac{\partial \pi'}{\partial x_j} = -\frac{g}{c_p \theta_{p0}} w + u_j \frac{\partial \pi'}{\partial x_j}, \quad (2.34)$$

where the base-state hydrostatic equation, (2.31b), is used. The prognostic equation for π' is thus

$$\begin{aligned} \frac{\partial \pi'}{\partial t} = & -u_j \frac{\partial \pi'}{\partial x_j} + \frac{g}{c_p \theta_{p0}} w - \pi \frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \frac{\partial u_j}{\partial x_j} \\ & + \frac{R}{c_p} \left[\left(\frac{L_v}{c_{vml} \theta} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{cond} + \left(\frac{L_s}{c_{vml} \theta} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{dep} + \frac{L_f}{c_{vml} \theta} \dot{r}_{frz} \right]. \end{aligned} \quad (2.35)$$

Terms involving phase changes [i.e., the terms involving \dot{r}_{cond} , \dot{r}_{dep} , and \dot{r}_{frz} on the right-hand side of (2.16) and (2.35)] are typically neglected in nonhydrostatic numerical models. These terms are necessary for the conservation of mass and energy, since the pressure equations are derived from equations governing the conservation of mass and energy. The motivation for the neglect of these diabatic terms comes mainly from two studies. Klemp and Wilhelmson (1978) recognized that mass cannot be

conserved with this assumption, but noted that tests with and without the diabatic terms yielded essentially identical results. They concluded that these terms have little effect on cloud-scale and mesoscale processes. Furthermore, they noted that these diabatic terms act as a source term for pressure, thereby increasing or decreasing mean pressure at a given level; since this is also true for the anelastic system, a commonly used alternative to the fully compressible system presented here, they note that their approximate system was no less valid than other systems used previously. Finally, Klemp and Wilhelmson argued that *pressure gradients* are still calculated accurately, implying that the acceleration of momentum is still handled correctly; therefore, they argued that knowledge of the exact pressure at every gridpoint is unnecessary.

The second study commonly used to justify the neglect of the diabatic terms in the pressure equation is Dudhia (1993). He argued that meteorological models with a rigid top (e.g., a height-coordinate model) may be more accurate when the diabatic terms are neglected. The basis of this argument is that the atmosphere expands vertically in the presence of diabatic heating. Since a model with a rigid top cannot allow this expansion, Dudhia argued that the neglect of diabatic terms in the pressure equation is consistent with this inability to expand.

Despite the arguments made by Klemp and Wilhelmson (1978) and Dudhia (1993), as well as a long history of the neglect of these terms in many meteorological models, these terms are retained in the numerical model developed here. The main reason justifying the inclusion of these terms here is the conservation of mass and energy. On a theoretical level, the conservation of these two properties is fundamental, and the neglect of terms necessary to achieve conservation should not be treated lightly. It is realized that with certain numerical techniques used in this (and other) meteorological models, it is impossible to conserve total mass and energy anyway. Nevertheless, it has been found that total mass and total energy errors can be several orders of magnitude lower in this numerical model when all terms in the governing equations are retained.

A simulation with a benchmark solution was designed to test the arguments of Klemp and Wilhelmson (1978), Dudhia (1993), and those put forth in the previous paragraph. Detailed results will be shown later in this chapter. It will be shown that, at least in some instances, a correct result cannot be obtained unless complete mass-

conserving and energy-conserving equation sets are used. Furthermore, the inclusion of all terms adds negligible additional computations to the model; total run-times are only a few percent higher than simulations which neglected these terms.

2.2.4 *Summary of governing equations*

The governing equations for the numerical model are summarized in Table 2.1. The fallout term for hydrometeors has been added to the r_l and r_i equations, where V_{tl} and V_{ti} are the (positive definite) terminal fall velocities of the liquid water and ice particles, respectively. These equations have been derived from conservation equations for total mass, total energy, and total momentum for a moist atmosphere. In the derivation of the momentum, potential temperature, and nondimensional pressure equations, the fallout of hydrometeors has been neglected; this limitation will be investigated in a future study. Aside from the assumption of instantaneous diffusion of heat between hydrometeors and air, no additional assumptions are made, no terms are ignored, nor are any terms replaced by approximate forms by invoking Taylor series expansions. The main advantages of these equations (over those used in other studies) are;

- The inclusion of the specific heats of liquid and frozen water particles,
- The inclusion of the diabatic terms in the pressure equation; and
- An unapproximated buoyancy term in the momentum equation.

2.3 Numerics

The numerical model was written in the framework of the “Arakawa C” grid (Arakawa and Lamb 1977), in which the momentum points are staggered one-half grid spacing from the locations of the scalars (e.g., θ , r_l , r_i , and π). The east-west momentum (u) is staggered in the x direction, the north-south momentum (v) is staggered in the y direction, and the vertical momentum (w) is staggered in the vertical direction. The vertical coordinate is height (z); therefore, the model gridpoints remain fixed in space over time. Vertical velocity is located at the surface and model top, so the first location of the scalar points is at a height of $\Delta z/2$. To maintain mass conservation, and to be

consistent with the use of a height coordinate with flat upper and lower surfaces, w is imposed to be zero at the upper and lower boundaries. In other words, no flow through these two surfaces is permitted.

Since the upper boundary is a rigid lid, the location of the boundary is placed far into the stratosphere to minimize the effects of this unphysical boundary condition. Studies by Hirschberg and Fritsch (1992, 1993) and Chagnon and Bannon (2001) found that a rigid upper boundary is valid for the study of phenomena that are smaller than synoptic scales (e.g., less than ~ 5000 km) and if the lid is placed far above sources of heating (e.g., at pressures of less than ~ 50 mb).

For all simulations presented in this work, the grid spacings in the horizontal and vertical directions are constant, although the horizontal grid spacing does not have to be the same as the vertical grid spacing (i.e., $\Delta x = \Delta y = \text{constant}$ and $\Delta z = \text{constant}$). This decision facilitated the incorporation of high order (e.g., third to sixth order) approximations to the advection terms in a simple yet accurate manner.

The code does not have a specification for the Earth's curvature, given the (relatively) small size of the domains to be used for the following studies, and the highly idealized setup. Also, there are no map-scale factors necessary for map projections, nor are any curvature terms included in the momentum equations.

The advection terms are written as the sum of a flux-form term and a divergence term, i.e.,

$$u_j \frac{\partial \alpha}{\partial x_j} = \frac{\partial (u_j \alpha)}{\partial x_j} - \alpha \frac{\partial u_j}{\partial x_j}, \quad (2.36)$$

where α represents the advected variable. Writing the advection terms in this way, as opposed to the advective form [left-hand side of (2.36)], can sometimes improve conservation of the variable being advected in a numerical model (e.g., Wilhelmson and Chen 1982, Dudhia 1993). Xue and Lin (2001) point out that it is sometimes possible to formulate an advective form that is numerically equivalent to the flux form with similar conservative properties – although probably only for staggered grid structures.

Nevertheless, the form represented by the right-hand side of (2.36) is utilized here to facilitate the incorporation of the high order discretizations presented in Wicker and Skamarock (2002).

Since some terms in the governing equations are not coded in a conservative form, this numerical model does not exactly conserve mass, momentum, or energy. Nevertheless, it will be shown in later sections that total mass and total energy errors when using this model can be several orders of magnitude lower than those from other commonly used numerical models due to the retention of terms in the governing equations (such as the diabatic contribution to the pressure equation).

Following the technique introduced by Klemp and Wilhelmson (1978), the portions of the governing equations that support acoustic waves are integrated on a smaller time step than other terms. This procedure greatly increases the efficiency of elastic nonhydrostatic models, with little loss of accuracy (see Skamarock and Klemp 1994). The model equations are restated here, where terms on the left-hand side are integrated on the small time step and terms on the right-hand side are integrated on the large time step:

$$\frac{\partial u_i}{\partial t} + c_p \theta_\rho \frac{\partial \pi'}{\partial x_i} - K_d \frac{\partial D}{\partial x_i} = -\frac{\partial(u_i u_j)}{\partial x_j} + u_i \frac{\partial u_j}{\partial x_j} + \delta_{i3} g \left(\frac{\theta_\rho}{\theta_{\rho 0}} - 1 \right) \quad (2.37)$$

$$\begin{aligned} \frac{\partial \pi'}{\partial t} - \frac{g}{c_p \theta_{\rho 0}} + \pi \frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \frac{\partial u_j}{\partial x_j} = & -u_j \frac{\partial \pi'}{\partial x_j} \\ & + \frac{R}{c_p} \left[\left(\frac{L_v}{c_{vml} \theta} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{cond} + \left(\frac{L_s}{c_{vml} \theta} - \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{dep} + \frac{L_f}{c_{vml} \theta} \dot{r}_{frz} \right] \end{aligned} \quad (2.38)$$

$$\begin{aligned} \frac{\partial \theta}{\partial t} = & -\frac{\partial(u_j \theta)}{\partial x_j} + \theta \frac{\partial u_j}{\partial x_j} - \theta \left(\frac{R_m}{c_{vml}} - \frac{R c_{pml}}{c_p c_{vml}} \right) \frac{\partial u_j}{\partial x_j} \\ & + \left[\frac{c_v L_v}{c_{vml} c_p \pi} - \theta \frac{R_v}{c_{vml}} \left(1 - \frac{R}{c_p} \frac{c_{pml}}{R_m} \right) \right] \dot{r}_{cond} \end{aligned} \quad (2.39)$$

$$\frac{\partial r_v}{\partial t} = -\frac{\partial(u_j r_v)}{\partial x_j} + r_v \frac{\partial u_j}{\partial x_j} - \dot{r}_{cond} \quad (2.40)$$

$$\frac{\partial r_c}{\partial t} = -\frac{\partial(u_j r_c)}{\partial x_j} + r_c \frac{\partial u_j}{\partial x_j} + \dot{r}_{cond} \quad (2.41)$$

To account for the slight instability associated with the Klemp-Wilhelmson small-time-step technique, a “divergence damper” is applied to the winds [last on the left-hand side of (2.37)]. Skamarock and Klemp (1992) show that these terms damp only the acoustic

waves and have little impact on “meteorologically significant” waves. As a positive side effect, the divergence damper allows a slightly larger acoustic time step, since they damp the very fast moving sound waves that limit the length of the small time step.

To account for phase changes, the model uses a saturation adjustment technique, similar to that proposed by Soong and Ogura (1973) and described in detail by Klemp and Wilhelmson (1978). In this technique, the equations are advanced forward in two steps: a dynamical step and a microphysical step. In the dynamical step, the model equations are integrated forward with all terms involving phase changes neglected. Then, the microphysics step is applied, in which only the terms involving phase changes (as well as the hydrometeor fallout terms) are included.

Pressure tendencies due to phase changes are included in the microphysics step. Since changes in pressure affect the saturation vapor pressure, an iterative scheme had to be developed for condensation. In this iterative scheme, equations (2.38)-(2.41) are advanced forward using a guess for \dot{r}_{cond} . The new values of θ and π are then used to calculate a new value of saturation mixing ratio (r_{vs}), which is then used to calculate a new guess for \dot{r}_{cond} . This cycle is repeated until the newest value of θ converges (to within machine accuracy) to the previous value. During each iteration, the value of \dot{r}_{cond} is determined by the following equation from Rutledge and Hobbs (1983),

$$\dot{r}_{cond} \cong \frac{r_v - r_{vs}}{\Delta t \left(1 + \frac{L_v^2 r_{vs}}{c_{pml} R_v T^2} \right)}, \quad (2.42)$$

where Δt is the time step. The iterative technique usually converges in 4-6 iterations.

The ability of the model to conserve mass and energy is evaluated by calculating the total mass (ρ_t) and the total energy (E_t) in the domain during a model integration.

The dry air density, ρ_a , is determined using the equation of state, (2.15). For simulations with no flow across the boundaries, and without external sources of mass, momentum, or energy, the domain-total sum of ρ_t and E_t should, theoretically, remain constant over time.

2.4 Physics

This section describes the physical processes that are included in the model. Currently, the model does not include atmospheric radiation or surface fluxes, since these processes will not be studied in the present research. There is an option to account for drag on the winds at the lowest model level in a simple manner. The remainder of this section describes the microphysics options and the subgrid-scale turbulence parameterization.

2.4.1 Microphysics

Three bulk microphysics schemes have been incorporated into the model. The first is the water-only parameterization of Kessler (1969), based on the equations presented in Klemp and Wilhelmson (1978). This scheme neglects the ice phase, and subdivides the liquid phase into two categories: cloudwater and rainwater. Thus, when this scheme is used, three mixing ratios are integrated in the model: water vapor (r_v), cloudwater (r_c), and rainwater (r_r).

The other two microphysics codes were obtained from MM5 (version 3.3). One is a version of the commonly used Lin et al. (1983) scheme, with modifications by Tau and Simpson (1989) and Braun and Tau (2000). This is a mixed phase bulk ice microphysics scheme, i.e., it allows for supercooled water, and for frozen particles at temperatures above 0 °C. There are prognostic equations for water vapor (r_v), cloudwater (r_c), rainwater (r_r), pristine ice crystals (r_i), snow (r_s), and large ice (i.e., graupel or hail, r_g). For the last category, the user must select a size distribution and density that favors graupel particles or hail. The main difference between these two categories, other than the assumed Marshall-Palmer slope/intercept which defines the size distribution, is the typical value for terminal fall velocity: for graupel, V_t values are typically 3-6 m s⁻¹, while for hail, V_t is between 10-20 m s⁻¹.

The third microphysics code is a version of the Reisner et al. (1998) scheme. This is also a mixed phase bulk ice microphysics scheme. There are prognostic equations for water vapor (r_v), cloudwater (r_c), rainwater (r_r), pristine ice crystals (r_i), snow (r_s), graupel (r_g), and an equation for the number density of pristine ice crystals (N_i).

2.4.2 Subgrid turbulence

All models that have grid spacing larger than the Kolmogorov microscale require a subgrid turbulence scheme. For most free atmospheric flows, this means that models with grid spacing larger than roughly 1 cm must parameterize the effects of subgrid-scale motions (i.e., turbulence). This condition arises because the governing equations must be filtered to remove the small-scale details that cannot be resolved by the model grid. In the process of filtering the governing equations, new terms arise – the subgrid terms. Since these subgrid terms include unresolved quantities, they must be modeled (i.e., parameterized) based on resolved fields.

One method to model the subgrid terms is through large eddy simulation (LES). In this technique, the grid spacing (Δ) must be much smaller than the scale of the largest resolved turbulent eddies. This condition ensures that essentially all of the energy- and flux-containing turbulence is accounted for on the model grid. It also implies that the grid spacing is within the inertial subrange, which is the spectral region where energy is transferred from large scales to small scales. Ultimately, energy is dissipated by eddies the size of the Kolmogorov microscale. In LES, it is considered sufficient to resolve the large energy-containing eddies, while parameterizing the rate of energy transfer to subgrid scales.

In the cloud-scale modeling community, the subgrid terms have been modeled with LES assumptions for decades. The appropriateness of this methodology has been addressed by Bryan and Fritsch (2001, 2002b). Since the goal of this research is to explicitly simulate the turbulent processes in the convective region of squall lines, and since grid spacing within the inertial subrange is used, the new numerical model has essentially been designed as a LES model. The formulation of the subgrid terms that follows is similar to that presented in Klemp and Wilhelmson (1978), with some slight modifications based on the study of Deardorff (1980).

The subgrid terms that arise from spatial filtering are summarized here as additional terms to the governing equations, (2.37) and (2.39)-(2.41), that is,

$$\frac{\partial u_i}{\partial t} = \dots + D_{u_i} , \quad (2.43)$$

$$\frac{\partial \theta}{\partial t} = \dots + D_{\theta} , \quad (2.44)$$

$$\frac{\partial r_{\chi}}{\partial t} = \dots + D_{r_{\chi}} , \quad (2.45)$$

where r_{χ} represents one of the water mixing ratios (vapor, cloudwater, rainwater, etc.).

Following standard practice, there is no subgrid term for the pressure equation. The subgrid term for momentum is modeled following Klemp and Wilhelmson (1978),

$$D_{u_i} = \frac{\partial}{\partial x_j} \left[K_m \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] , \quad (2.46)$$

where K_m is the subgrid eddy viscosity. The subgrid terms for scalars (i.e., θ and r_{χ}) can be modeled in two ways in this numerical modeling system. The first is a “standard” or “dry” formulation, in which θ and r_{χ} are mixed without consideration of phase changes:

$$D_{\theta} = \frac{\partial}{\partial x_j} \left(K_h \frac{\partial \theta}{\partial x_j} \right) , \text{ and} \quad (2.47)$$

$$D_{r_{\chi}} = \frac{\partial}{\partial x_j} \left(K_h \frac{\partial r_{\chi}}{\partial x_j} \right) , \quad (2.48)$$

where K_h is the subgrid eddy diffusivity for scalars. The justification for this methodology is discussed in Klemp and Wilhelmson (1978). Should any phase changes occur due to subgrid tendencies, they will be handled later in the code by the explicit microphysics (i.e., in the microphysics step). This technique is often considered to be appropriate to the philosophy of “saturation adjustment”, in which the equations are integrated forward without regards to phase changes and then “adjusted” at the end of the time step to account for supersaturation.

However, several studies (e.g., Betts 1973, Deardorff 1976, Tripoli and Cotton 1981, Stauffer et al. 2001) have show the benefits of a second formulation using thermodynamic variables that are conserved during phase changes. Based on these studies, this model has an option to mix conserved variables, and then extract tendencies of θ and r_{χ} based on the results. The liquid water potential temperature (θ_l) of Emanuel (1994) is used in the model,

$$\theta_l \equiv T \left(\frac{p_{00}}{p} \right)^{\beta_1} \left(1 - \frac{r_l}{\varepsilon + r_t} \right)^{\beta_1} \left(1 - \frac{r_l}{r_t} \right)^{-\beta_2} \exp \left[\frac{-L_v r_l}{(c_p + c_{pv} r_t) T} \right], \quad (2.49a)$$

where,

$$\beta_1 \equiv \frac{R + R_v r_t}{c_p + c_{pv} r_t}, \quad (2.49b)$$

$$\beta_2 \equiv \frac{R_v r_t}{c_p + c_{pv} r_t}. \quad (2.49c)$$

Bryan and Fritsch (2002c) show that this definition of liquid water potential temperature is exact for reversible phase changes. Unlike other commonly used definitions of θ_l , this definition accounts for the specific heat of liquid water, and is therefore applicable in cases of large liquid water contents (e.g., deep moist convection). For the process under consideration here (subgrid turbulent motions), the only phase changes accounted for are the condensation of water vapor and the evaporation of cloud water; all other phase changes are handled by the equations in the explicit microphysics scheme. This assumption is consistent with the derivation of θ_l , in which reversible phase changes have been assumed. Therefore, for (2.49), r_l is simply the cloudwater mixing ratio, $r_l = r_c$, and r_t is simply the sum of water vapor and cloudwater mixing ratio,

$$r_t = r_v + r_c. \quad (2.50)$$

All other hydrometeor classes (r_r , r_i , r_s , r_g) are mixed according to (2.48); phase changes for these processes are restricted to those allowed by the microphysics code, since they are not reversible, and application of liquid water potential temperature would be inappropriate. The subgrid tendencies of θ_l and r_t are

$$\left. \frac{\partial \theta_l}{\partial t} \right|_{subgrid} = \frac{\partial}{\partial x_j} \left(K_h \frac{\partial \theta_l}{\partial x_j} \right) \quad (2.51a)$$

$$\left. \frac{\partial r_t}{\partial t} \right|_{subgrid} = \frac{\partial}{\partial x_j} \left(K_h \frac{\partial r_t}{\partial x_j} \right). \quad (2.51b)$$

After calculating the right-hand side of (2.51), θ_l and r_t are advanced forward one time step. New values of θ , r_v , and r_c are then retrieved from the new values of θ_l and r_t by using the definitions, (2.49) and (2.50), the fact that supersaturation with respect to liquid

water is not permitted, and the most recent value of pressure, which is assumed not to change due to subgrid mixing. An iterative scheme based on the Newton-Raphson technique is used to accomplish this retrieval. These new values of θ , r_v , and r_c are used to define D_θ , D_{r_v} , and D_c .

The subgrid coefficients, K_m and K_h , are derived from the subgrid turbulent kinetic energy, E , following Deardorff (1980),

$$K_m = 0.1l\sqrt{E} \quad , \quad (2.52)$$

$$K_h = (1 + 2l/\Delta s)K_m \quad , \quad (2.53)$$

where Δs is a measure of the grid size,

$$\Delta s = (\Delta x \Delta y \Delta z)^{1/3} \quad , \quad (2.54)$$

l is a subgrid length scale,

$$l = \begin{cases} \Delta s & \text{in unstable thermodynamic conditions} \\ \min(\Delta s, 0.76\sqrt{E/N^2}) & \text{in stable thermodynamic conditions} \end{cases} \quad (2.55)$$

and N^2 is the Brunt-Väisälä frequency squared. The definition of N^2 , and the definition of a neutral thermodynamic profile, will be provided in Chapter 3. A prognostic equation is used to predict E ,

$$\frac{\partial E}{\partial t} = -\frac{\partial}{\partial x_j} (u_j E) + E \frac{\partial u_j}{\partial x_j} + K_m \frac{\partial u_i}{\partial x_j} D_{ij} - K_h N^2 + \frac{\partial}{\partial x_j} \left(K_m \frac{\partial E}{\partial x_j} \right) - \frac{c_\epsilon}{l} E^{3/2} \quad , \quad (2.56)$$

where

$$D_{ij} = \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \quad , \quad \text{and} \quad (2.57a)$$

$$c_\epsilon = 0.19 + 0.51l/\Delta s \quad . \quad (2.57b)$$

The first two terms on the right-hand side of (2.56) represent the flux-form advection of E , the third term is the shear production term, the fourth term is the buoyancy term, the fifth term is the turbulent flux divergence of E , and the sixth term is the dissipation term.

2.5 Evaluation using two-dimensional dry simulations

The numerical model was used to investigate the properties of four advection schemes during simulations of a highly nonlinear flow: leapfrog-in-time with second

order spatial derivatives (LF2); leapfrog-in-time with fourth order spatial derivatives (LF4); third order Runge-Kutta in time with a third order upstream-biased approximation to the advection terms (RK3-UP3); and third order Runge-Kutta in time with fifth order upstream-biased advection (RK3-UP5). The experimental design is similar to the experiments performed by Tripoli (1992). A warm bubble is placed in a dry, hydrostatic atmosphere and allowed to rise. The simulations are performed with a two-dimensional (x - z) version of the model with no-slip boundary conditions on the upper and lateral boundaries. A constant eddy viscosity is imposed, so that the details of the advection schemes can be studied without needing to understand the role of subgrid-scale turbulence effects. Experience has shown that for very high resolution, and given a constant eddy mixing coefficient, the details of the results converge to one solution no matter what advection scheme is chosen. The properties of various advection schemes are then studied by coarsening the resolution and comparing the results to the reference (high resolution) output.

For the first experiment, the initial background (i.e., base state) atmosphere has a constant potential temperature of 300 K. The initial warm bubble is circular, with a radius of 800 m, and is placed in the middle of the domain centered at 2 km above ground. The initial maximum perturbation temperature is +3 K at the center, gradually decreasing to zero at the edges. The perturbation pressure is set to zero over the entire domain. The domain for this experiment is 9 km high and 9 km wide. No smoothers or filters are used in the experiments shown in this section, other than an Asselin filter for the leapfrog advection schemes and the “divergence damper” on the acoustic steps. For this first experiment, a constant eddy mixing coefficient of $15 \text{ m}^2 \text{ s}^{-1}$ was applied over the entire domain in both the vertical and horizontal dimensions.

The reference (or “converged”) simulation was performed with a grid spacing of 12.5 m. The results after 20 min using LF4 advection/time integration are shown in Fig. 2.1. The thermal rises and expands over time. Two “rotors” form on the sides of the thermal, while the top of the thermal is stretched. Large θ gradients develop in the middle of the thermal (i.e., within the “arch” spanning between the two rotors).

Results from simulations using the four advection schemes with 100 m grid spacing are shown in Fig. 2.2. The two simulations using the leapfrog schemes cannot

maintain the proper shape of the thermal (Fig. 2.2a and 2.2b). In particular, the central arch of the thermal is not continuous as in the reference simulation. The rotors do not maintain a circular shape, but rather are noisy and contain many details that are not present in the reference solution.

In contrast, the simulations using Runge-Kutta time differencing and upstream-biased spatial derivatives maintain the shape of the thermal very well (Fig. 2.2c and 2.2d). The third order upstream (UP3) scheme is much more diffusive than the fifth order scheme (UP5) – the rotors in the UP3 simulation are nearly perfect circles, and do not contain any of the details present in the reference simulation. In contrast, the simulation using UP5 is strikingly similar to the reference simulation – most notable are the details within the rotors, the presence of a single arch connecting the two rotors (although the center of the arch is beginning to weaken), and the magnitude of the θ field is represented very well. The gradients captured in the UP5 simulation are superior to the gradients in the UP3 simulation.

A second series of tests was conducted following the methodology of Wicker and Skamarock (1998). In this experiment, simulations of a rising thermal in an initially calm environment are compared to simulations with an initially constant wind. The lateral boundaries for this experiment are periodic. Thus, the thermal in the simulations with an initial horizontal wind is advected laterally while rising. The simulations are compared when the thermal should have been advected once around the periodic domain.

The base state atmosphere for this second experiment is identical to the first. A larger thermal of radius 2 km is used, but with a slightly weaker initial maximum perturbation in the center of +2 K. The domain is 10 km high and 20 km wide, with 125 m grid spacing. In Figs. 2.3-2.5, the simulations labeled as “no translation” are initialized with a constant wind of zero everywhere; in these simulations, the thermal moves straight upward over time. The simulations labeled as “with translation” are initialized with a constant horizontal (u) wind of 20 m s^{-1} everywhere; the thermal in these simulations is therefore transported laterally while it is moving upwards. Since the upper and lower boundaries are free slip, the thermal in the simulations with an initial 20 m s^{-1} should move laterally through the periodic domain and should end up back at the center in 1000 s. For these experiments, no background mixing is applied (i.e., the eddy mixing

coefficient is set to zero everywhere), which reveals more details of the different advection schemes.

Simulations using the leapfrog schemes are shown in Fig. 2.3. The thermal in the LF2 experiment is grossly distorted when it is advected laterally, and exhibits strong damping (Fig. 2.3c). Furthermore, the thermal is no longer symmetric. The downwind (right) side of the thermal is positioned well when compared with the control case (i.e., compared to the simulation without translation), but the upwind (left) side of the thermal is too far west and much lower than the rotor in the control case (i.e., as compared to Fig. 2.3a).

The simulation using the LF4 scheme maintains the shape of the thermal very well, and has excellent placement (Fig. 2.3b). However, the gradients in the thermal have been smoothed considerably. Oscillations are apparent on the upwind (left) side of the thermal, especially in the vertical velocity field (Fig. 2.5b).

The simulations using the Runge-Kutta schemes with upwind-biased space differencing show excellent results (Fig. 2.4). The shape, details, position, and gradients within the thermals have been maintained well. The RK3/UP5 scheme is the only scheme that maintains a high degree of symmetry – see, e.g., the vertical velocity field Fig. 2.5d.

Based on these simulations, the RK3/UP5 clearly emerges as the best scheme (out of those tested here) in terms of shape preservation, maintenance of translation speed, minimal overshoots and undershoots, and comparison to reference solutions. As discussed in Wicker and Skamarock (1998, 2002), the relatively large time step for numerical stability also makes the RK3/UP5 scheme attractive.

2.6 Evaluation using two-dimensional moist simulation

Unfortunately, most of the analytic/benchmark cases commonly used today do not include moist processes. Moreover, despite the varying methods used to include moist processes in numerical models, there does not appear to be a commonly agreed-upon method to evaluate a moist model formulation. Typically, a model developer will demonstrate that a model produces reasonable fields of parameters such as vertical velocity, cloud/rain water mixing ratios, rainfall, etc., for a case of deep, moist

convection. While this is an important and necessary step in the development of a numerical model, the lack of a known solution limits the conclusions that can be drawn from such tests. Another common method compares the results from a new model to published results from a different model. While, again, this is an important step in evaluating the fidelity of a new numerical model, this method makes it possible to propagate questionable assumptions through time.

A new benchmark simulation was designed for testing numerical models with moisture. The design of the simulation is analogous to the nonlinear warm thermal benchmark case used by Tripoli (1992) and Wicker and Skamarock (1998), but includes phase changes of water vapor and cloud water.

The simulation presented by Wicker and Skamarock (1998) with no initial winds (Fig 2.4b) was chosen as the dry reference case. An interesting extension of this benchmark simulation that, to our knowledge, has not been presented before, is the relative independence of the results to the base state value of potential temperature. Specifically, if the initial unperturbed state is hydrostatically balanced and neutrally stable, and if an identical initial buoyancy perturbation is applied, the results are similar no matter what value of potential temperature is used. For example, Fig. 2.6 presents the results of simulations where initial environments of 240 K, 270 K, and 300 K are used. In all cases, an identical initial buoyancy perturbation is applied, where buoyancy for the dry case can be defined as,

$$B = g \frac{\theta'}{\theta_0}. \quad (2.58)$$

The results displayed in Fig. 2.6 are all similar. Although the fields from these various simulations are not exactly the same, the key point is that the thermal has the same structural details, suggesting a relative independence to the exact thermodynamic value of the neutrally stable base state. Therefore, this property of the solution provides a benchmark for testing new model formulations.

Seeking to obtain a similar result for a moist atmosphere, the initial environment is again specified to be hydrostatic and characterized by exactly neutral stability. In the dry case, it is possible to define neutral stability based on only one thermodynamic variable: potential temperature. However, a moist atmosphere is not as simple. To

simplify the specification of the moist base state, two assumptions are made: 1) the total water mixing ratio is constant at all levels, i.e., $r_t = r_v + r_c = \text{constant}$; and, 2) phase changes are exactly reversible, i.e., $\Delta r_v = \Delta r_{vs} = -\Delta r_c$. Under these two assumptions, a neutrally stable environment can be obtained using one conservative thermodynamic variable (see, e.g., Durran and Klemp 1982). We use the wet equivalent potential temperature, defined for a reversible moist adiabatic atmosphere by

$$\theta_e = T \left(\frac{p_d}{p_{00}} \right)^{-R/(c_p + c_{pl}r_t)} \exp \left(\frac{L_v r_v}{(c_p + c_{pl}r_t)T} \right) \quad (2.59)$$

(Durran and Klemp 1982; Emanuel 1994, pg 120), where p_d is the partial pressure of dry air. Using (2.29), (2.31b), and (2.59), the vertical profiles of π , θ , r_v , and r_c can be obtained if values for θ_e and r_t are provided. The value of r_t must be greater than r_{vs} at all levels, so that the initial sounding is saturated, and $r_c > 0$ at all levels.

All other parameters are the same as for the dry case; i.e., the surface pressure is 1000 mb, the initial wind field is zero, grid spacing is 100 m, and the domain dimensions are as before. No microphysics parameterization is used, other than the assumption of reversible phase changes. Precipitation fallout is not allowed.

For the complete moist thermal simulation, the initial buoyancy field is identical to the dry benchmark case when θ_0 is 300 K. For moist conditions, buoyancy is given by

$$B = g \left(\frac{\theta_\rho}{\theta_{\rho 0}} - 1 \right). \quad (2.60)$$

Therefore, using (2.60) for the moist case, (2.58) for the dry case, and the definition of θ_ρ , the initial θ field is:

$$\theta = \theta_{\rho 0} \frac{(1 + r_t)}{(1 + r_{vs}/\varepsilon)} \left(\frac{\theta'_\rho|_{300}}{300 \text{ K}} + 1 \right). \quad (2.61)$$

Since it has been assumed that $r_t = \text{constant}$, and $r_v = r_{vs}(p, T)$, the buoyant perturbation has slightly more water vapor and slightly less cloud water than the base state sounding.

The results after 1000 s of integration for a case in which $\theta_e = 320$ K and $r_t = 0.020$ are presented in Fig. 2.7. The results of this moist case are very similar to the results of the dry case (Fig. 2.4b), especially with regards to the structural details such as the two

“rotors” that form on the sides of the thermal and the thin “arch” that connects them. The moist thermal rises slightly faster than the dry thermal, and after 1000 s the vertical velocity field has higher maximum and minimum values. Nevertheless, the structural details are remarkably similar.

It is important to reiterate that the model formulation for this simulation does not neglect any term in the governing equations. In particular, the specific heat of liquid water (c_{pl}) is included, and the diabatic contribution to the pressure equation is included; it is a common assumption in numerical models to neglect these two effects.

Furthermore, the error in total mass and energy conservation is quite small (about 10^{-4} percent), especially compared to model formulations that ignore certain terms in the governing equations (which will be presented shortly). Given this high degree of accuracy in mass and energy conservation, and the similarity to the dry case, it seems reasonable that this case can be considered a moist benchmark to which moist numerical models can be compared. Additionally, as in the dry case, the simulation proposed here is remarkably insensitive to the values used to define the initial neutrally stable sounding. This result helps establish the fact that the design of the moist benchmark simulation is robust, i.e., the correct result is not dependent on a specific initial thermodynamic environment. This is an important point, since it provides further confidence that the results truly represent a benchmark solution.

The assumptions of reversible phase changes and the absence of hydrometeor fallout clearly make this test case a simplification of reality. On the other hand, the case has been valuable for testing formulations of numerical models. As an example, four different model formulations (Table 2.2) are tested and presented in this section.

Equation Set A makes two approximations that are commonly used in cloud models: the diabatic contribution to the pressure equation is ignored, and the specific heat of water is neglected. This equation set is similar to that used in the Klemp-Wilhelmson model, MM5, and ARPS, as well as several other numerical models. For Equation Set B, only the specific heat of water is neglected in the thermodynamic and pressure equations. Since the diabatic contribution to the pressure equation is included, this equation set conserves mass. This formulation is similar to that used in COAMPS, and in some respects is similar to models that integrate a density equation rather than a pressure

equation (such as the height-coordinate version of the WRF model). In Equation Set C, the specific heat of water is included, as is the diabatic contribution to the pressure equation, but the term involving divergence in the thermodynamic equation is neglected; scale analysis suggests that this term is small and, presumably, negligible. This equation set has not been used in the literature, but is included here as an example of how approximate forms of the governing equations can be tested numerically. Equation Set D uses the ice-liquid water potential temperature (θ_{il}) of Tripoli and Cotton (1981). Equation set D also neglects the diabatic contribution to the pressure equation, and the specific heat of water in the pressure equation. This equation set is similar to that used in RAMS and in the University of Wisconsin Nonhydrostatic Modeling System.

Results (Figs. 2.8 and 2.9) clearly show the dramatic impact of neglecting terms from the complete thermodynamic and pressure equations – none of the simulations using approximate equations compare well with the benchmark solution (Fig. 2.7). In all of these cases, the thermal rises much slower than the thermal in the benchmark run. In the θ_e fields, large undershoots (i.e., anomalously low values, depicted by dashed contours) develop in all cases.

It is interesting to note that the output from runs A and B are very similar. Both thermals rise to ~ 6.9 km, and the vertical motion patterns are nearly identical (Figs. 2.8a and 2.8b). This result suggests that the extra effort required to conserve mass in a numerical model (by including the diabatic contribution to the pressure equation) may not lead to significant improvements in results unless total energy is also conserved (as in the benchmark). Despite the similarity of the w and θ_e fields, the time series of mass and energy errors are dramatically different (Fig. 2.10). The time series for run A (the short-dashed lines in Fig. 2.10) has an oscillatory nature, with a period of about 62 seconds. In contrast, the time series for run B (dotted lines in Fig 2.10) evolves smoothly, with only very minor changes through the simulations. Furthermore, run A loses considerable mass and energy throughout the simulation, with a total mass error that is about 30 times greater than the mass error in the benchmark simulation after 1000 s. Run B has nearly identical mass errors to the benchmark simulation, and has a slight increase in total energy. It is unclear how such dramatically different runs in terms of mass and energy errors can have such similar dynamic and thermodynamic fields (e.g., Figs. 2.8 and 2.9).

The results from run C were surprising. Among all the simulations, this run least resembles the benchmark case. The thermal only reached 5.8 km, and the θ_e pattern is quite different from the other runs (Fig. 2.9c). However, it is interesting to note that mass and energy errors from run C are comparable to those in the benchmark simulation (Fig. 2.10). Apparently, this formulation produces unacceptable results due to an approximation that was made in only one equation, without making a consistent approximation in another equation. Perhaps a “counter-balancing” assumption in the pressure equation would improve the results. It is worthwhile to note that other equations sets make consistent approximations throughout; e.g., in equation set B, the specific heats of water are ignored in every equation. Whatever the reason behind the poor results of equation set C, this test highlights the danger of neglecting terms that may seem unimportant under a scale analysis.

The simulation that used θ_{il} as the governing variable (run D) produced results that most closely match the benchmark. The thermal reaches ~ 7.6 km, and becomes only slightly distorted in shape (Fig. 2.9d). On the other hand, this formulation has the largest total mass and total energy errors out of all runs presented here (Fig. 2.10).

The value chosen for r_t in these tests is abnormally high for the imposed temperature sounding. One might wonder whether these results only come about due to this unphysical initial environment. A comparison of simulations with different values for θ_e and r_t (not shown) reveals that the differences presented here are accentuated over those one would expect to find in more “normal” environments. Nevertheless, it is clear that the mass-conserving and energy-conserving form of the thermodynamic and pressure equations can produce the desired results in all environments, and that these equations should be preferred over approximate equation sets.

2.7 Evaluation using three-dimensional simulations of deep moist convection

Additional simulations have been conducted using more realistic model configurations to address whether the conclusions drawn from the previous section hold for more typical uses of numerical models. This section presents three-dimensional simulations of a supercell thunderstorm and then a simulation of a squall line using the various equation sets. For these simulations, the Kessler microphysics parameterization

is used, hydrometeor fallout is allowed, and a complete subgrid turbulence model is included.

2.7.1 Supercell

Overall, results using the five equation sets are similar. Convective organization is not modified in any of the tests we have performed (e.g., all of the simulations with environments favoring supercells *do* produce supercells). For example, Fig. 2.11 shows results from the supercell simulation from the commonly used equation set A (Table 2.2) and the new (i.e., benchmark) equation set (Table 2.1). In both simulations, a supercell with hook echo develops on the southern flank of the line, with multicellular convection along the rest of the line. Some of the conclusions noted in the previous section are evident in subtle ways. For example, the simulations with the benchmark equation set tends to have the strongest updrafts (Fig. 2.12a), the highest cloud tops, and generally higher rainfall rates (Fig. 2.12b). At the end of the simulation, about 10% more rainfall occurred with the benchmark equation set.

A domain-wide pressure drift is apparent in simulations with equation set A. For the supercell simulation, the minimum surface pressure generally falls throughout the simulation with equation set A (Fig. 2.12c). The simulation with the benchmark equation set has larger fluctuations in minimum surface pressure, but is 1.5 mb higher after 120 min. The domain-wide mean pressure is 1.9 mb higher at this time. The main reason behind the differences is most likely the neglect of the diabatic term from the prognostic pressure equation. Klemp and Wilhelmson (1978) noted that this term is technically necessary to accurately predict pressure, and that the neglect of this term amounts to adding or removing mass. A time series of total mass error confirms this analysis: total mass is lost at a greater rate during the simulation with equation set A (Fig. 2.12d). The larger fluctuations in pressure noted in Fig. 2.12c are related to different pressure structures that develop using the different equation sets; this conclusion is based on an analysis of pressure fields from the supercell simulation (not shown), and is highlighted in simulations of a squall line in the next section.

2.7.2 Squall line

Three-dimensional simulations of a squall line were conducted using a domain that is 300 km long in the cross-line (x) direction, 60 km long in the along-line (y) direction, and 18 km deep. Grid spacing of 500 m was used in both the horizontal and vertical directions. The initial thermodynamic environment is identical to that used for the supercell simulation. The initial wind profile is unidirectional, with no initial wind in the y direction, and a u profile that increases linearly from zero at the surface to 10 m s^{-1} at 2.5 km, and is constant at 10 m s^{-1} above 2.5 km. The squall line was initiated using a north-south line thermal located initially at $x=200 \text{ km}$.

Plots of pressure at $z=5.25 \text{ km}$ from four simulations are shown in Fig. 2.13. The convective region of the squall line is located at approximately $x=200 \text{ km}$ in all of the simulations, with the stratiform region extending roughly 50 km to the west. The pressure field using equation set A (Fig. 2.13a) is clearly much smoother than that from the benchmark equation set (Fig. 2.13b). The difference between the minimum and maximum values is 2.0 mb in the simulation with equation set A, but is 3.6 mb with the benchmark equations. It is unclear what the pressure field at mid-levels in an actual squall line looks like, given the difficulties in measuring this variable with high spatial resolution in more than one dimension. Nevertheless, additional simulations were performed to see whether the different pressure structures were a manifestation of the equations sets themselves, or due to numerical techniques. It should be noted that the inclusion of the diabatic contribution to the pressure equation is almost universally ignored in elastic nonhydrostatic atmospheric models. Those studies that do include the term usually do not specifically mention how the term was implemented.

For the simulations noted as using the benchmark equation, the diabatic term to the pressure equation (the second on the right-hand side of (2.38), hereafter referred to as $\dot{\pi}_{diab}$) is implemented at the end of a large time step – i.e., within the “microphysics” step discussed in section 2.3. To test whether this particular implementation of $\dot{\pi}_{diab}$ was related to the pressure structures in Fig. 2.13b, two additional simulations were performed. Both simulations include $\dot{\pi}_{diab}$ on the smaller acoustic time steps that are used to account for the propagation of acoustic waves. For one simulation, an

approximate value for $\dot{\pi}_{diab}$ was included on each small time step. Specifically, at the end of each microphysics step the value of $\dot{\pi}_{diab}$ is stored in an array before proceeding to the next model time increment. Then, when the small time step marching begins, the saved value for $\dot{\pi}_{diab}$ is applied during each small step (i.e., the value is held constant, as it is for the nonacoustic terms such as buoyancy and advection). At the end of every small time step marching subsection, this approximate tendency for $\dot{\pi}_{diab}$ is subtracted off (since it is only an approximate value). Then, when the microphysics step begins, the diabatic term is calculated in the same manner as in the benchmark implementation. The process is then repeated. [This technique has been implemented into the height-coordinate version of the Weather Research and Forecasting (WRF) model – a model that integrates a density equation, and in which mass conservation has been stressed.] The output from this simulation is displayed in Fig. 2.13c. In some areas of the squall line's convective region, the pressure field is smooth like when the $\dot{\pi}_{diab}$ is ignored; in particular, the pressure feature at approximately $(x,y)=(205 \text{ km}, 5 \text{ km})$ in Fig. 2.13c is similar in amplitude and structure to those in Fig. 2.13a. However, the pressure field is also dominated by small wavelength spikes in pressure; for example there are several instances where the pressure at one grid point is more than 1 mb higher than the pressure at any adjacent grid point. These features show up in Fig. 2.13c as small dark circles. Although it is unknown whether such large gradients in pressure actually exist in mid-levels of deep moist convection, it does not seem that these spikes represent a well-resolved solution. Rather, such small-scale features are more likely a manifestation of numerical instability.

The second experiment implemented the $\dot{\pi}_{diab}$ term directly on the small time steps. That is, condensation adjustment was applied after every small time step. For this simulation, only the condensation term was calculated on the small steps: all other microphysical terms (e.g., evaporation of rain water) were calculated in the microphysics step, with the $\dot{\pi}_{diab}$ tendency from these processes included in the same manner as the benchmark technique. Since the tendency from condensation is the largest contributor to $\dot{\pi}_{diab}$ in the convective region, it was felt that this would be a sufficient test of this methodology. Results from this technique exhibit the same smooth, well-resolved

pressure structures (Fig. 2.13d) as in the simulation that neglected $\dot{\pi}_{diab}$ (Fig. 2.13a). This simulation is considered the most realistic, since the diabatic term is implemented with the least number of assumptions. It seems reasonable, and consistent with other studies (e.g., Chagnon and Bannon 2001), that acoustic waves would emanate from regions of strong diabatic heating. When including the $\dot{\pi}_{diab}$ on the small time steps, these waves can be created in a realistic manner. It seems that the methods that attempt to account for the $\dot{\pi}_{diab}$ term on (or after) the large time step are not accurately accounting for this process. It is therefore concluded that the large amplitude pressure features in Fig. 2.13b and 2.13c are numerical manifestations of this inappropriate implementation of $\dot{\pi}_{diab}$.

The effect of the various implementations is summarized via one-dimensional (y-direction) pressure spectra at 5.25 km above ground in Fig. 2.14. These spectra were determined by computing one spectrum every two minutes for 1 hour and then temporally averaging. To ensure that roughly the same area of the squall line was being analyzed at each time, the spectra were computed at the y-slice that had the highest vertical velocity variance at each time – at this level, this places the spectra within the convective region of the squall line. The simulation with equation set A (curve a) and the simulation with $\dot{\pi}_{diab}$ computed on the small steps (curve d) have essentially identical shapes. These two curves have a slope of $\kappa^{-7/3}$ from wavelengths of about 4Δ to about 12Δ , which is consistent with turbulence theory (e.g., Lumley and Panofsky 1964; Hill and Wilczak 1995). In contrast, the simulation in which $\dot{\pi}_{diab}$ was implemented during the microphysics step only (curve b) has a pressure spectrum that increases sharply at 12Δ , has a peak at about 5Δ , and remains quite high (compared to curves a and d) all the way to wavelengths of 2Δ . The simulation using an approximation to $\dot{\pi}_{diab}$ on the small steps (curve c) has a nearly constant slope from 12Δ to 2Δ , but is still considerably higher than curves a and d.

Based on the analyses of Fig. 2.13 and 2.14, and the theory that pressure spectra in an inertial subrange should have a $\kappa^{-7/3}$ slope, it is concluded that the technique in which the diabatic term is included on the small (acoustic) time steps is the most accurate

implementation. Unfortunately, this technique is considerably more expensive to implement; the simulation took almost twice as much time to complete as the other two. All simulations used a small time step that is 1/12 of the large time step. Using the Runge-Kutta technique of Wicker and Skamarock (2002), this means a total of 19 small time steps are used per large step. Thus, saturation adjustment was applied 19 more times per time step in the simulation with $\dot{\pi}_{diab}$ on the small steps as compared to the benchmark implementation.

In all simulations hereafter in this work, the diabatic term was applied on the microphysics step only, since the additional cost of applying this term on the small steps was deemed unacceptable. This approach can also be justified since numerical problems are not apparent in any field other than pressure. For example, spectra of vertical velocity from the three techniques are nearly identical. Furthermore, domain-wide statistics such as total rainfall and maximum updraft from the three simulations using the $\dot{\pi}_{diab}$ are similar to each other (especially when compared to output using equation set A). Finally, it was found that the apparent instability can be efficiently smoothed through the divergence damper (see section 2.3). Skamarock and Klemp (1992) and Wicker and Skamarock (1998) found that this artificial smoother can sometimes be necessary to control an instability inherent in the split acoustic time step technique. Two-dimensional (x - z) simulations of a thunderstorm cell using 500 m grid spacing are presented in Figs. 2.15-2.16. In the first simulation, a light amount of smoothing was applied, with a non-dimensional value for the divergence damping coefficient (α_d) of 0.02. Small-wavelength oscillations are apparent in the perturbation pressure field and in the zero vertical velocity contour. In contrast, a simulation in which α_d was set to 0.05 has much fewer (and lower amplitude) perturbation pressure oscillations, and no apparent oscillations in the vertical velocity field. It is noted that a value of $\alpha_d=0.05$ is well within that used by other modeling studies. Therefore, increasing the effect of the divergence damper can be an effective (and inexpensive) method to control pressure oscillations when the diabatic contribution to the pressure equation is included.

2.8 Parallelization

One goal of the present research is to simulate deep moist convection with very high resolution (i.e., grid spacing of order 100 m). Since the simulation of convection and its near environment requires domain sizes of several hundred km in the horizontal directions and of order 20 km in the vertical, these simulations require about 10^7 to 10^8 grid points at this resolution. Most atmospheric models (including the one developed here) require several hundred bytes of memory per grid point. Therefore, the goal of high resolution can require tens to hundreds of Gigabytes of memory to run. Current single-processor computers do not have nearly enough memory to meet these requirements. And, in addition to the memory constraints, simulation on a single processor could take an unacceptably long period of time to complete, given the large number of grid points as well as the prohibitively small time step required to maintain numerical stability with this small grid spacing. Therefore, a version of the model was developed that could be run on distributed memory parallel computers (i.e., computing systems with multiple processors, each of which has its own memory, or shares memory with only a few other processors).

The strategy for parallelizing the model code centered on domain decomposition. In this approach, the total domain is decomposed into subsections (or subdomains). Each processor is assigned one of the subdomains, which means that it contains all of the data associated with that portion of the domain. Communication between processors (e.g., transfer of data from one processor to another) is limited to instances when the information stored on one processor is needed by another processor. Since this model employs finite differencing techniques, communication is usually only required along the edges of each subdomain.

Communication between processors was handled using Message Passing Interface (MPI) – a standardized set of functions and subroutines that can be used to send data between processors in a distributed memory computing system. To increase the overall parallel efficiency of the code, the code utilizes non-blocking MPI routines to “overlap” communications with computations. That is, data that will be needed by another processor can be sent (communicated) before it is actually used; in the meantime, other computations are proceeding on both the sending processor and the receiving processor. In essence, both communications and computations are occurring at the same time.

For computer systems that have shared memory nodes (i.e., when two or more processors have access to the same bank of memory), the model can also do shared memory parallelization utilizing OpenMP. In this configuration, MPI is used to communicate information between nodes, while OpenMP is used to parallelize processing within a node. This configuration can significantly reduce the amount of data that must be communicated, which can sometimes reduce the time required to complete a simulation.

In the present implementation, each processor has a subdomain that is equal in size, and therefore has the same memory. A common problem with evenly-sized subdomains in atmospheric modeling, however, is load balancing. Even though each processor has the same number of grid points to work with, it is common to have different physical processes occurring in different portions of the domain. For example, one part of the domain may have precipitation processes occurring, which requires significant calculations; at the same time, another portion of the domain does not have any precipitation, and its processor quickly exits the moist microphysics subroutine. Load imbalancing occurs because all processors must communicate information at a common time. For the hypothetical example, one processor sits idle until the other processor finishes its work in the moist microphysics subroutine. This limitation is usually not a problem when the Kessler (no ice) microphysics scheme is used, since there are only a few short calculations required. However, when an ice microphysics scheme is used, load imbalancing can be a problem. In some simulations, processors near the edge of the domain that have little or no moisture can spend up to 20% of the time sitting idle waiting for other processors to complete their microphysical calculations.

2.8.1 Evaluation of parallelization

The parallelized numerical model was tested on two parallel computing systems. One is a Linux cluster at the Center for Academic Computing at The Pennsylvania State University, hereafter referred to as Lion-X. This machine has sixty-four 500 MHz Intel processors in thirty-two dual-processor nodes. There are two network devices to handle inter-processor communication: a fast ethernet switch, which can transfer data at 100

Mb/s; and a Myricom Myrinet switch, rated at 1.28 Gb/s. Results from both network devices are presented in this section.

The second computing device is an IBM SP cluster at the National Center for Atmospheric Research, hereafter referred to as Blackforest. This machine has more than 1,000 Power-3 IBM processors, each with a clock speed of 375 MHz. The Power-3 processors can perform up to 4 operations per cycle, and hence have an overall speed up to four times greater than the Intel processors of the Linux cluster. The communications device of Blackforest is rated at more than 1 Gb/s. Blackforest is composed of 4-processor nodes with 2 GB of shared memory per node. Based on this hardware setup, two tests were performed on Blackforest: one in which MPI was used exclusively to transfer information between processors; and a second in which OpenMP was used to parallelize intra-node processing, and MPI was used to transfer data between nodes.

2.8.2 *Fixed-domain test*

Two tests were used to evaluate the parallel performance of the numerical model. The first is a fixed-domain test, in which the number of grid points in the domain is kept constant as processors are added. For this test, as the number of processors is increased, each processor has fewer grid points. The goal of using more processors in this setup is to decrease the time required to run the simulation; hence, this type of test would be important for real-time modeling centers, such as the National Centers for Environmental Prediction.

A two-hour simulation of a supercell thunderstorm was used for the fixed-domain test. The domain has 128x128 grid points in the horizontal and 40 vertical levels for runs on Blackforest, and 120x120x40 grid points for runs on Lion-X. The initial conditions are identical to those used by Weisman and Rotunno (2000), except only the “1/4 circle” hodograph case is considered. Timing results are listed in Table 2.3. The results are also presented graphically in Fig. 2.17, where “speedup” is the ratio of time required to run the simulation on one processor to the time required to run on n processors,

$$speedup = \frac{\text{time on 1 processor}}{\text{time on } n \text{ processors}} . \quad (2.62)$$

For a perfectly parallelized code, the speedup would be identical to the number of processors used, i.e., ideal performance on Fig. 2.17a would be a straight line with slope of 1 (the thick gray line). Results from Lion-X (the solid lines in Fig. 2.17) show a clear advantage of the faster Myrinet communication device over the roughly 10 times slower ethernet device. In fact, results are nearly perfect out to 32 processors, and excellent out to 60 processors. A plot of parallel efficiency is provided in panel 2.17b, where

$$parallel\ efficiency = 100 \times \frac{speedup}{n} . \quad (2.63)$$

Ideally, a parallel efficiency of 100% is desired. The results using the Myrinet communication device show greater than 90% efficiency on Lion-X, which is an exceptionally good result.

Results on Blackforest do not show the same spectacular results: in fact, parallel efficiency is between 50%-70% with 64 processors, and drops to about 40% with 128 processors. Further analysis of the runs on Blackforest reveals that the model spends a greater percentage of time waiting for communications to complete as compared to runs on Lion-X, despite the fact that the communication device of Blackforest is rated at about the same speed as the Myrinet device of Lion-X. However, the processors of Blackforest are considerably faster (almost 4 times as fast) as those of Lion-X. It is concluded that Lion-X shows superior parallel performance *for this particular test* since communications on Lion-X are completed while calculations are being performed (since non-blocking communications are used with MPI to overlap communications and computations); in contrast, the faster Blackforest processors complete the computations in considerably less time – and before communications are complete – and then sit idle until communications finish. It could therefore be concluded that this experiment was not ideally designed to test the advantages of the IBM-SP, and perhaps the parallel performance of Blackforest in Fig. 2.17 is underrated. Furthermore, it is noted that the time required to complete this simulation with 128 processors on Blackforest is about 3 minutes (see Table 2.3); clearly, this number of processors is unnecessary for this particular simulation. Overall however, the parallel performance of the code can be considered successful. Note, for example, that greater than 90% parallel efficiency is achieved with 16 processors on Blackforest.

A comparison of the MPI-only approach (long dashed lines in Fig. 2.17) to that using both MPI and OpenMP simultaneously (short dashed lines) reveals greater performance for the MPI-only approach. This result was surprising, since it was expected that the lower communication requirements of the OpenMP approach would result in enhanced performance. It was found that using OpenMP for shared-memory parallelization within the 4-processor nodes of Blackforest caused the decrease in performance. It is unclear, at this point, why greater efficiency could not be achieved using OpenMP, and further study will be carried out in the future.

2.8.3 Scaled-domain test

The second type of test used to evaluate the parallel performance of the numerical model is a “scaled-domain” test, in which each processor has the same number of grid points as the number of processors are increased. Thus, the overall domain during this test becomes larger as processors are added, but the work required *per processor* remains roughly the same. It is important to note that a scaled-domain test is more relevant to this study than the fixed-domain test presented in the last section, because the goal is to simulate squall lines with very high resolution, which requires very large numbers of grid points (e.g., 10^7 to 10^8 grid points).

The sub-domains for this test each have $100 \times 100 \times 72$ grid points; this is roughly the number of grid points per processor that is required to simulate mesoscale convective systems (e.g., squall lines) with very high resolution. For example, using these sub-domain dimensions with a grid spacing of 125 m and with 128 processors results in a numerical domain with horizontal dimensions of 200 km x 100 km and a vertical extent of 18 km; such a domain is suitable for the simulation of deep convective phenomena (squall lines, supercells, bow echoes). Finally, it is noted that this number of grid points requires about 200 MB of memory per processor, which is about half of the maximum memory available per processor on Blackforest.

Timing results for a model integration of 100 time steps are presented in Table 2.4 and Fig. 2.18. For this test, an ideal result would be an identical time as the number of processors are increased, or a perfectly horizontal line on Fig. 2.18 (see thick gray lines). Results for both Lion-X and Blackforest show a marked increase from one processor to four processors, and then nearly constant timing results after four processors. It is

assumed that the decrease in performance from one to four processor occurs because of the increased memory access requirements; in other words, the four processor runs have more grid points *per node* than the one processor runs. Beyond four processors, the number of grid points per node remains the same.

Considering only the performances for at least four processors, the results using Lion-X show a degradation in performance out to 60 processors. Specifically, the 60 processor simulations require about 10% more time than the 4 processor simulations. This performance is considered acceptable, and can be used as a benchmark to estimate timing for longer simulations. For Blackforest, the performance is nearly ideal from 4 processors to 128 processors. (The increase in performance from 4 to 16 processors for the MPI-only run may be an anomaly.) This result proves the utility of the numerical model for extremely large domain simulations (the 128 processor run has 9.2×10^7 grid points), as well as the robustness of the parallelization technique.

Table 2.1. The governing equations of the numerical model. In sensitivity tests presented later, these equations are referred to as the “benchmark” equation set.

$\frac{Du_i}{Dt} = -c_p \theta_\rho \frac{\partial \pi'}{\partial x_i} + \delta_{i3} g \left(\frac{\theta_\rho}{\theta_{\rho 0}} - 1 \right)$
$\begin{aligned} \frac{D\theta}{Dt} = & -\theta \left(\frac{R_m}{c_{vml}} - \frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \right) \frac{\partial u_j}{\partial x_j} + \frac{c_v}{c_p c_{vml} \pi} (L_v \dot{r}_{cond} + L_s \dot{r}_{dep} + L_f \dot{r}_{frz}) \\ & - \theta \frac{R_v}{c_{vml}} \left(1 - \frac{R}{c_p} \frac{c_{pml}}{R_m} \right) (\dot{r}_{cond} + \dot{r}_{dep}) \end{aligned}$
$\begin{aligned} \frac{\partial \pi'}{\partial t} = & -u_j \frac{\partial \pi'}{\partial x_j} + \frac{g}{c_p \theta_{\rho 0}} - \pi \frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \frac{\partial u_j}{\partial x_j} \\ & + \frac{R}{c_p} \left[\left(\frac{L_v}{c_{vml} \theta} - \pi \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{cond} + \left(\frac{L_s}{c_{vml} \theta} - \pi \frac{R_v c_{pml}}{R_m c_{vml}} \right) \dot{r}_{dep} + \frac{L_f}{c_{vml} \theta} \dot{r}_{frz} \right] \end{aligned}$
$\frac{Dr_v}{Dt} = -\dot{r}_{cond} - \dot{r}_{dep}$
$\frac{Dr_l}{Dt} = \dot{r}_{cond} - \dot{r}_{frz} + \frac{1}{\rho_a} \frac{\partial}{\partial z} (\rho_a V_{tl} r_l)$
$\frac{Dr_i}{Dt} = \dot{r}_{dep} + \dot{r}_{frz} + \frac{1}{\rho_a} \frac{\partial}{\partial z} (\rho_a V_{ti} r_i)$

Table 2.2. Summary of equation sets used in sensitivity tests.

Eqt. set	Thermodynamic equation	Pressure equation
A	$\frac{D\theta}{Dt} = \frac{L_v}{c_p \pi} \dot{r}_{cond}$	$\frac{D\pi}{Dt} = -\pi \frac{R}{c_v} \frac{\partial u_j}{\partial x_j}$
B	$\frac{D\theta}{Dt} = \frac{L_v}{c_p \pi} \dot{r}_{cond}$	$\frac{D\pi}{Dt} = -\pi \frac{R}{c_v} \frac{\partial u_j}{\partial x_j} + \left(\frac{RL_v}{c_p c_v \theta} - \pi \frac{R_v}{c_v} \right) \dot{r}_{cond}$
C	$\frac{D\theta}{Dt} = \left[\frac{c_v L_v}{c_p c_{vml} \pi} - \frac{R_v}{c_{vml}} \theta \left(1 - \frac{R}{c_p} \frac{c_{pml}}{R_m} \right) \right] \dot{r}_{cond}$	$\frac{D\pi}{Dt} = -\pi \frac{R}{c_p} \frac{c_{pml}}{c_{vml}} \frac{\partial u_j}{\partial x_j} + \left(\frac{RL_v}{c_p c_{vml} \theta} - \pi \frac{RR_v c_{pml}}{c_p R_m c_{vml}} \right) \dot{r}_{cond}$
D	$\frac{D\theta_{il}}{Dt} = 0$	$\frac{D\pi}{Dt} = -\pi \frac{R}{c_v} \frac{\partial u_j}{\partial x_j}$

Table 2.3. Timing results from the “fixed-domain” test. Columns show the number of processors used (N), the number of horizontal grid points per processor (g.p./proc.), and the time (in seconds) required to complete the simulation. For runs using Blackforest, the columns labeled “MPI” refer to the MPI-only parallelization, while “MPI-OMP” refer to runs using MPI to communicate data between nodes and OpenMP to parallelize processing within a node.

N	Lion-X, ethernet		Lion-X, myrinet		Blackforest MPI		Blackforest MPI-OMP	
	g.p/proc.	time (s)	g.p/proc.	time (s)	g.p/proc.	time (s)	g.p/proc.	time (s)
1	120x120	28091.7	120x120	26903.3	128x128	9384.7	128x128	9393.0
4	60x60	7022.6	60x60	6726.3	64x64	2536.6	64x64	2564.3
8	60x30	3795.8	60x30	3374.6	64x32	1165.7	64x32	1262.3
16	30x30	2201.1	30x30	1723.5	32x32	634.9	32x32	655.3
24	30x20	1569.1	30x20	1134.2				
32	30x15	1219.4	30x15	8544.6	32x16	356.2	32x16	383.4
48	20x15	941.2	20x15	591.4				
60	20x12	782.8	20x12	460.8				
64					16x16	221.6	16x16	273.8
128					16x8	175.7	16x8	188.4

Table 2.4. As in Table 2.3, except timing results from the “scaled-domain” test.

N	Lion-X, ethernet		Lion-X, myrinet		Blackforest MPI		Blackforest MPI-OMP	
	g.p/proc.	time (s)	g.p/proc.	time (s)	g.p/proc.	time (s)	g.p/proc.	time (s)
1	100x100	3030.7	100x100	3073.9	100x100	1335.8	100x100	1337.7
4	100x100	3687.0	100x100	3602.8	100x100	2486.0	100x100	2350.0
16	100x100	3792.4	100x100	3665.4	100x100	2428.4	100x100	2389.2
32	100x100	3822.6	100x100	3695.0	100x100		100x100	
60	100x100	4065.9	100x100	3860.1	100x100		100x100	
64	100x100		100x100		100x100	2465.0	100x100	2485.5
128	100x100		100x100		100x100	2552.9	100x100	2581.1

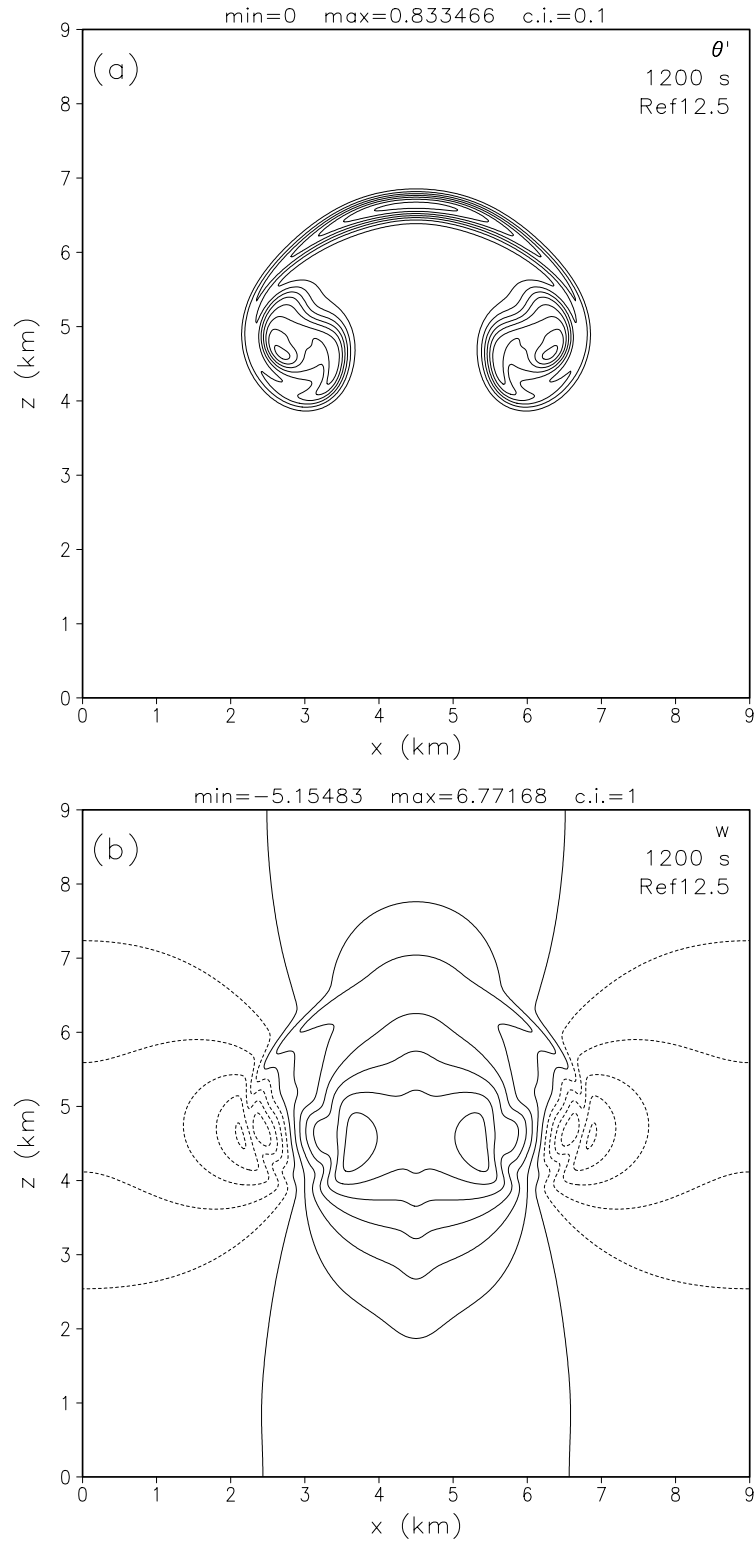


Fig. 2.1. Results of the “reference” warm bubble experiment after 20 min. Grid spacing is 12.5 m. a) Potential temperature perturbation (θ' , K) is plotted with a 0.1 K contour interval. The zero contour is excluded. b) Vertical velocity (w) is plotted with a 1 m s^{-1} contour interval. Dashed lines represent negative contours.

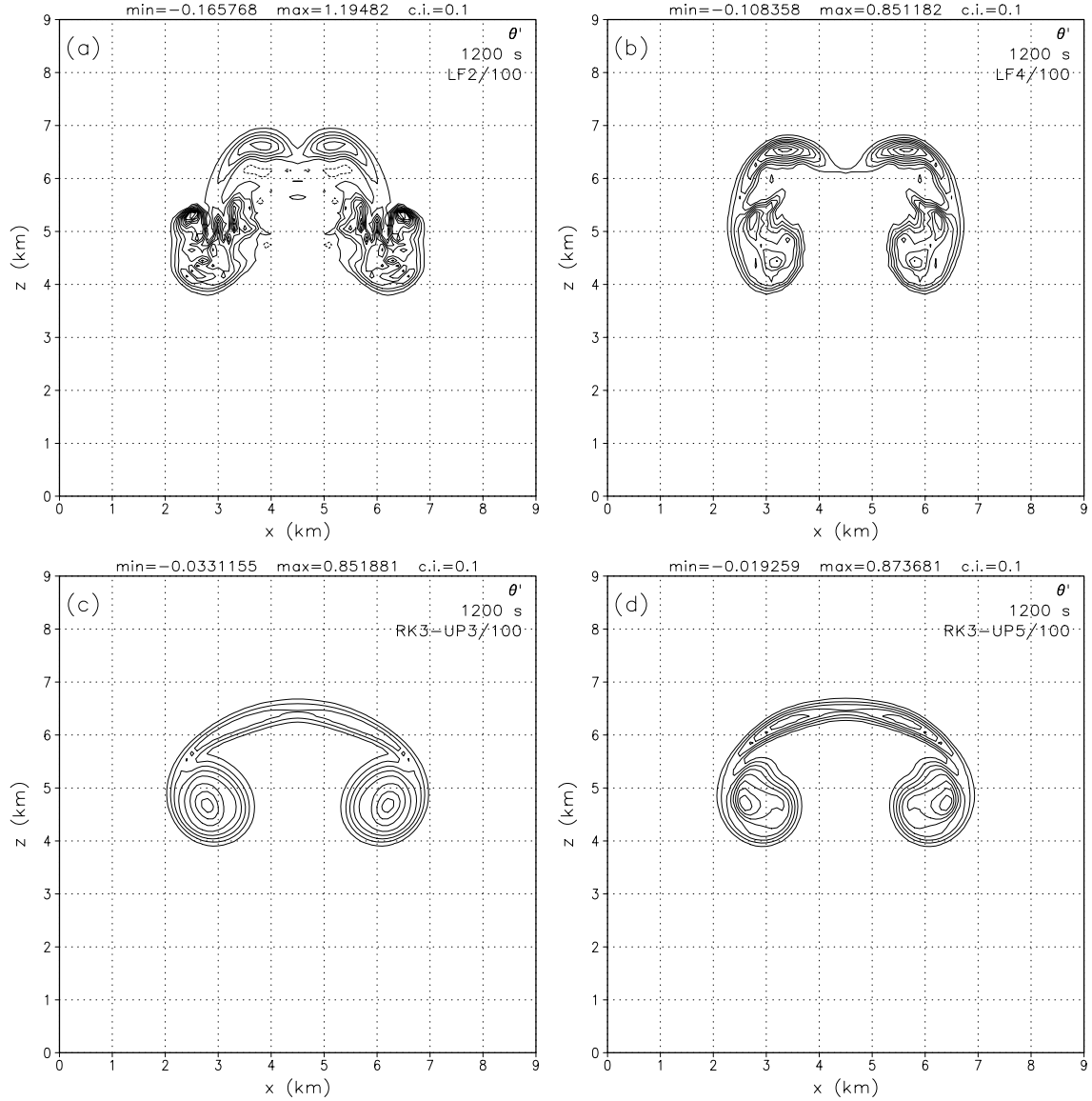


Fig. 2.2. Results of warm bubble simulations with 100 m grid spacing after 20 min. Potential temperature perturbation (K) is plotted with a contour interval of 0.1 K. Dashed lines represent negative contours. The zero contour is omitted. a) Leapfrog in time, second order in space; b) leapfrog in time, fourth order in space; c) third order Runge-Kutta in time, third order in space; and d) third order Runge-Kutta in time, fifth order in space.

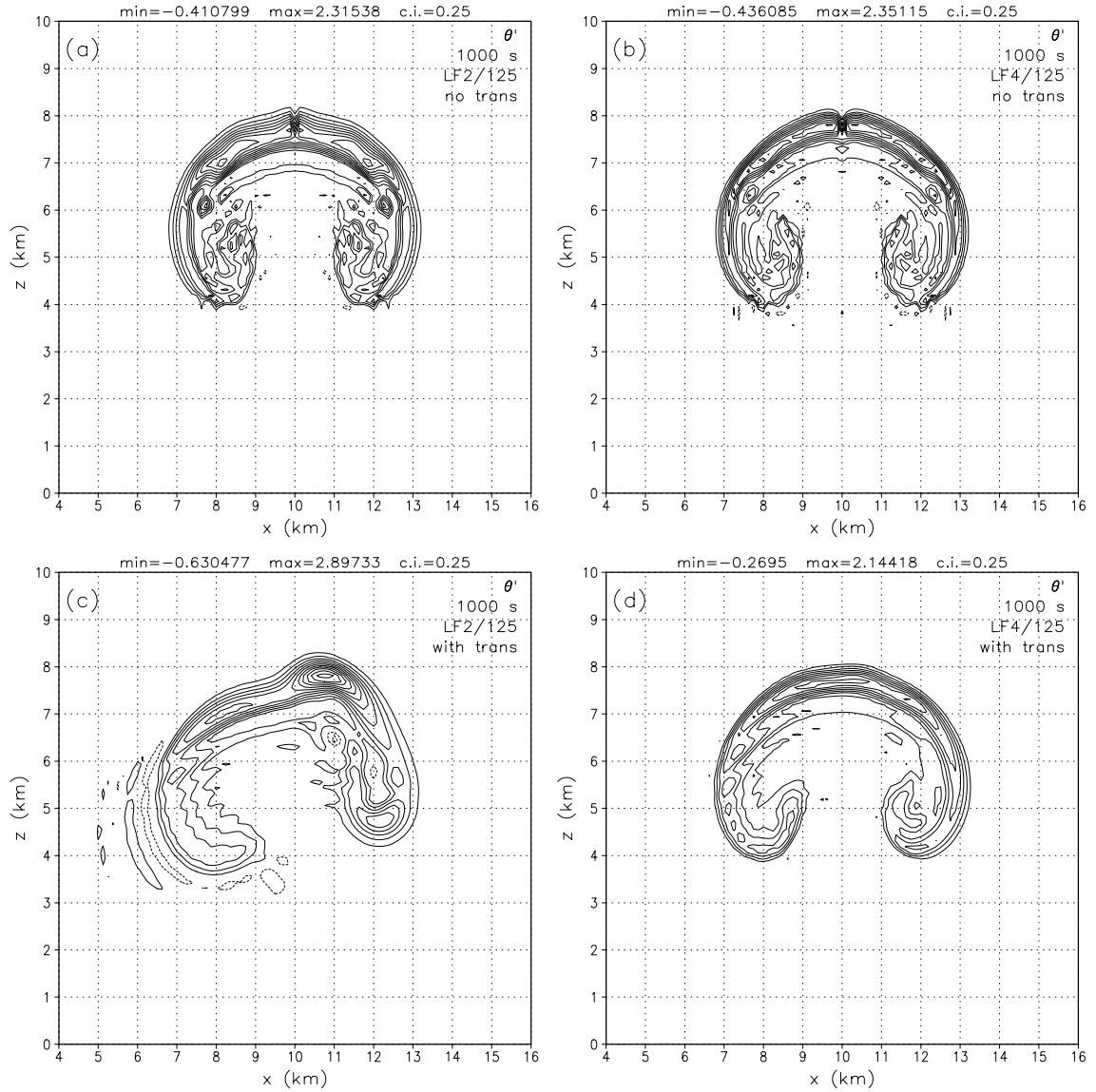


Fig. 2.3. Results of the second warm bubble experiment after 1000 s. The grid spacing is 125 m. Potential temperature perturbation is plotted with a contour interval of 0.25 K. Dashed lines represent negative contours, solid lines represent positive contours. The zero contour is omitted. (a) Leapfrog in time, second order in space with no initial winds; (b) leapfrog in time, fourth order in space with no initial winds; (c) as in (a), but with an initial wind of 20 m s^{-1} ; (d) as in (b), but with an initial wind of 20 m s^{-1} .

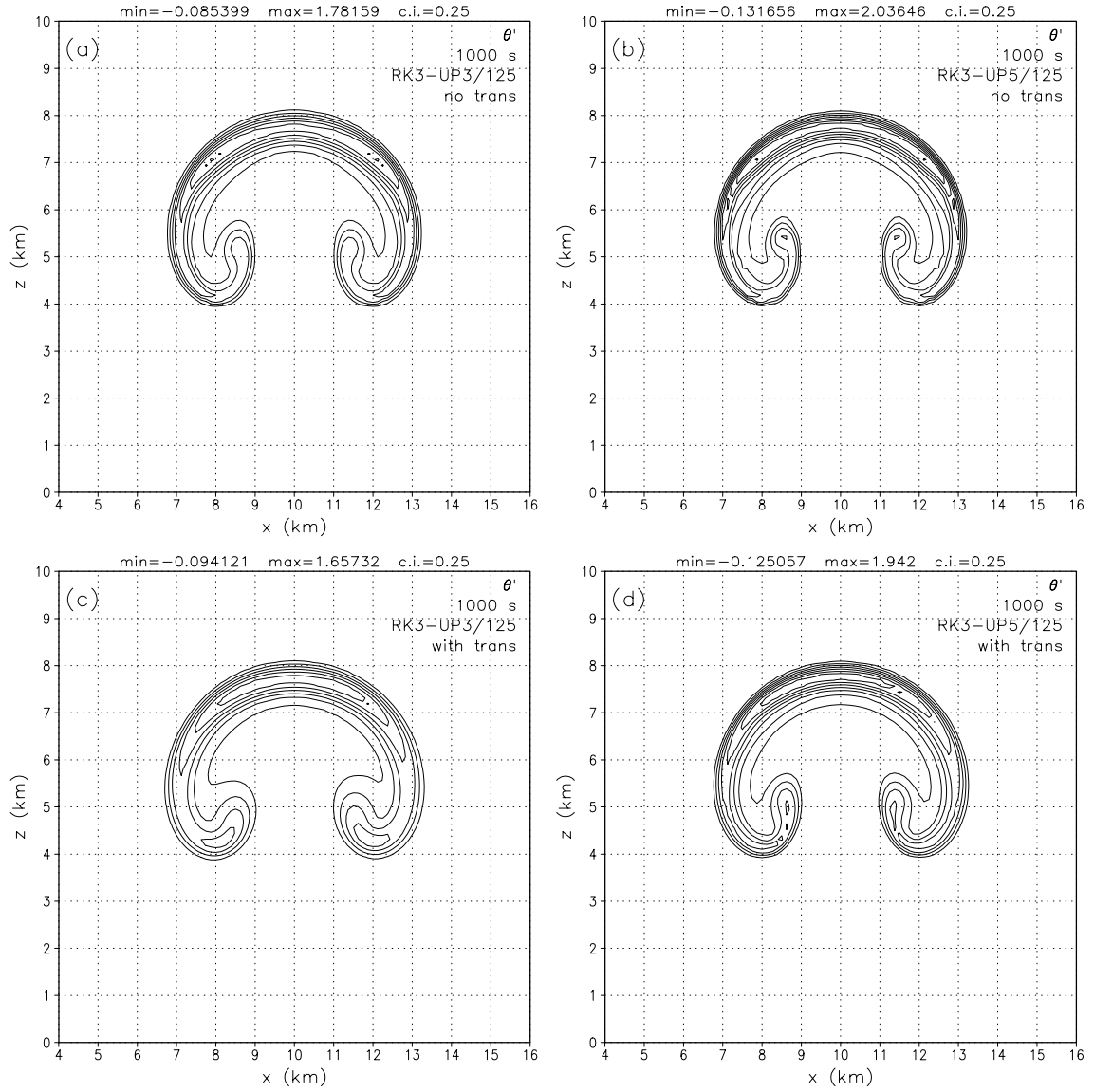


Fig. 2.4. As in Fig. 3, except (a) third order Runge-Kutta in time, third order spatial derivatives, with no initial winds; (b) third order Runge-Kutta in time, fifth order spatial derivatives, with no initial winds; (c) as in (a), but with an initial wind of 20 m s^{-1} ; (d) as in (b), but with an initial wind of 20 m s^{-1} .

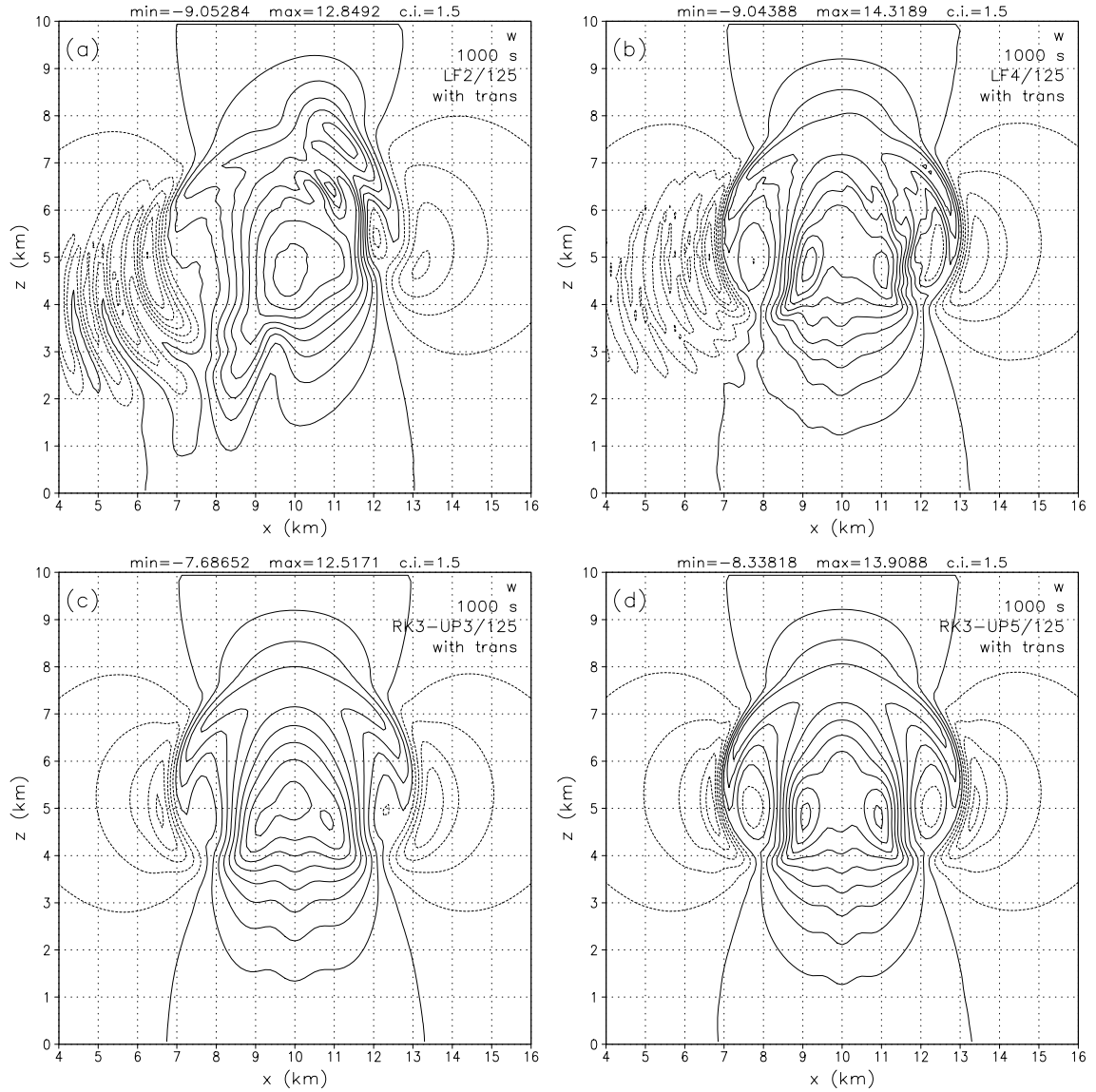


Fig. 2.5. Vertical velocity at 100 s for the second warm bubble experiment with an initial wind of 20 m s^{-1} . Contour interval is 1.5 m s^{-1} . Dashed lines represent negative contours. (a) Leapfrog in time, second order in space; (b) leapfrog in time, fourth order in space; (c) third order Runge-Kutta in time, third order in space; (d) third order Runge-Kutta in time, fifth order in space.

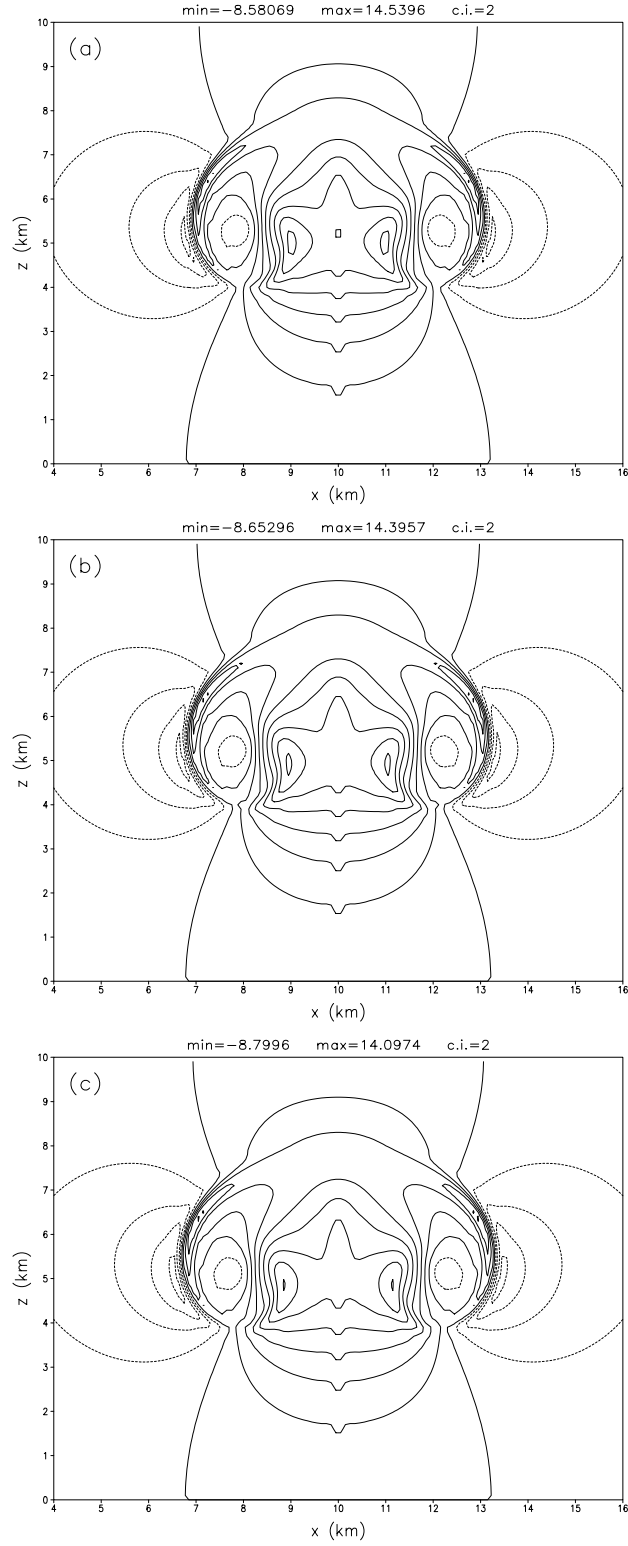


Fig. 2.6. Results of dry thermal simulations for (a) $\theta_0 = 300$ K; (b) $\theta_0 = 270$ K; and (c) $\theta_0 = 240$ K. Vertical velocity is contoured every 2 m s^{-1} . Negative contours are dashed.

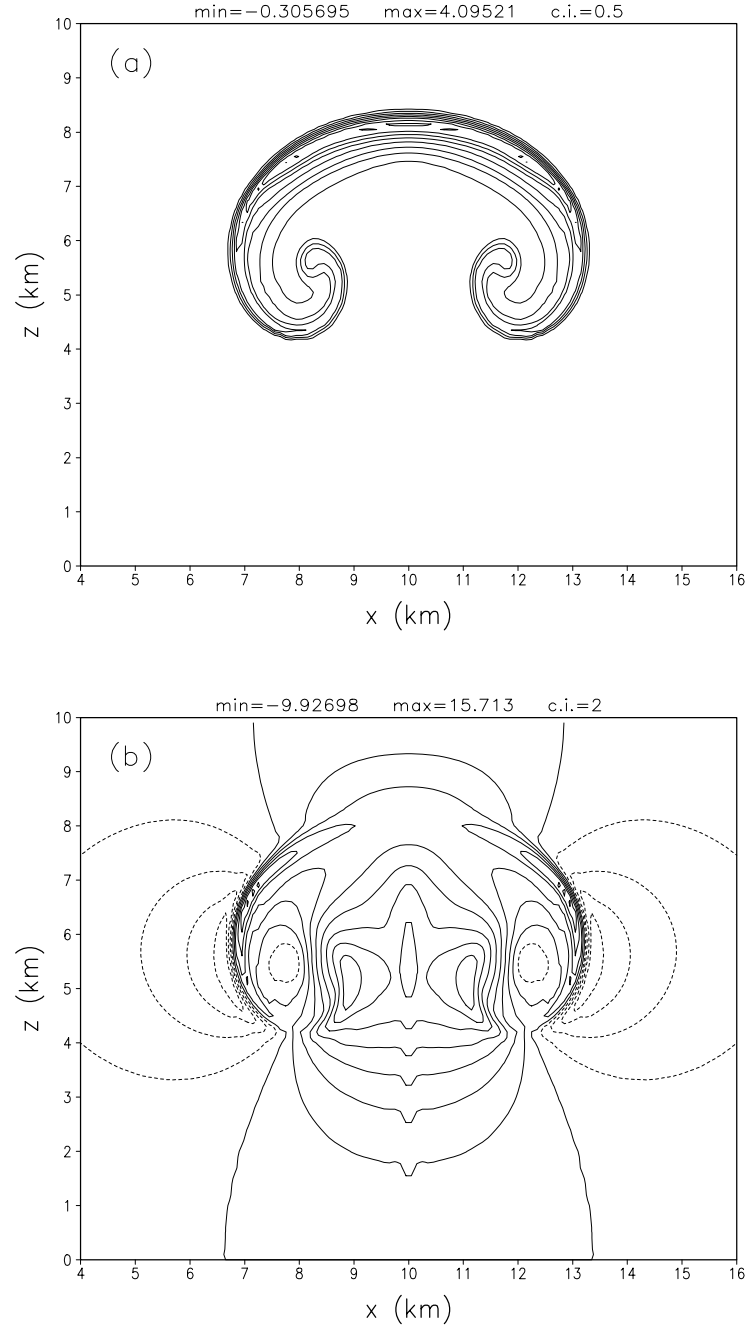


Fig. 2.7. Results of the moist thermal simulation for $\theta_e=320$ K and $r_t=0.020$. (a) Perturbation wet equivalent potential temperature is contoured every 0.5 K. The zero contour is omitted. (b) Vertical velocity is contoured every 2 m s^{-1} . Negative contours are dashed.

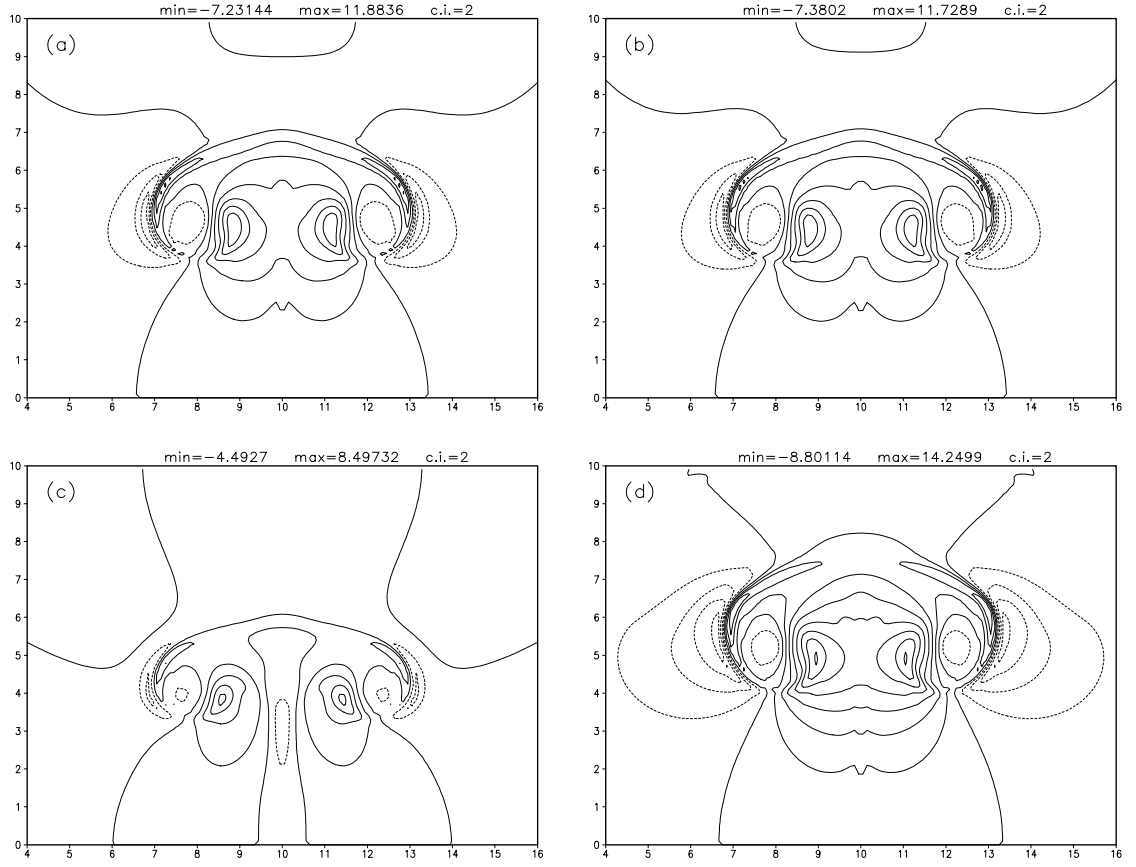


Fig. 2.8. Vertical velocity from moist thermal simulations for various model formulations: (a) equation set A, (b) equation set B, (c) equation set C, and (d) equation set D. See text and Table 2 for details of the equation sets. Contour interval is 2 m s⁻¹. Negative values are dashed.

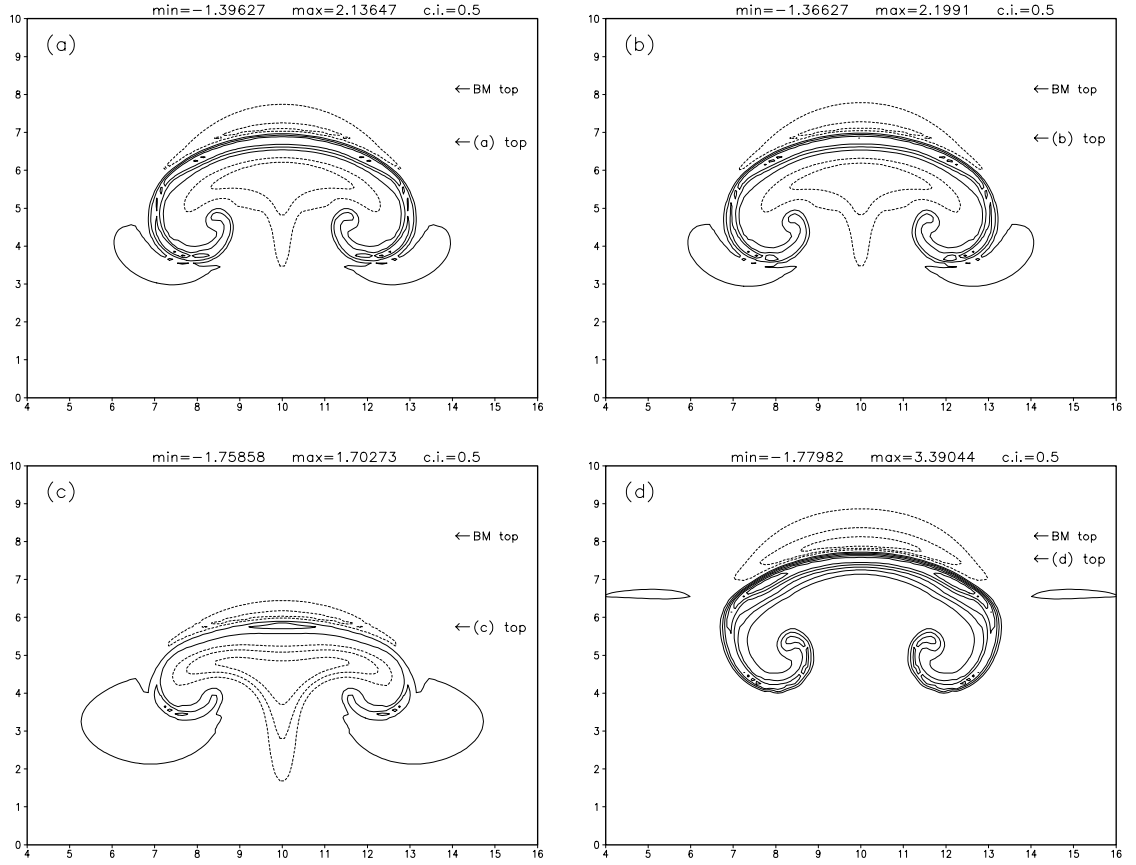


Fig. 2.9. As in Fig. 8, except perturbation wet equivalent potential temperature is contoured every 0.5 K. Negative contours are dashed. The zero contour is omitted. Near the right-hand side of each panel, the height of the top of the thermal is indicated, along with the height of the top of the thermal from the benchmark (BM) simulation.

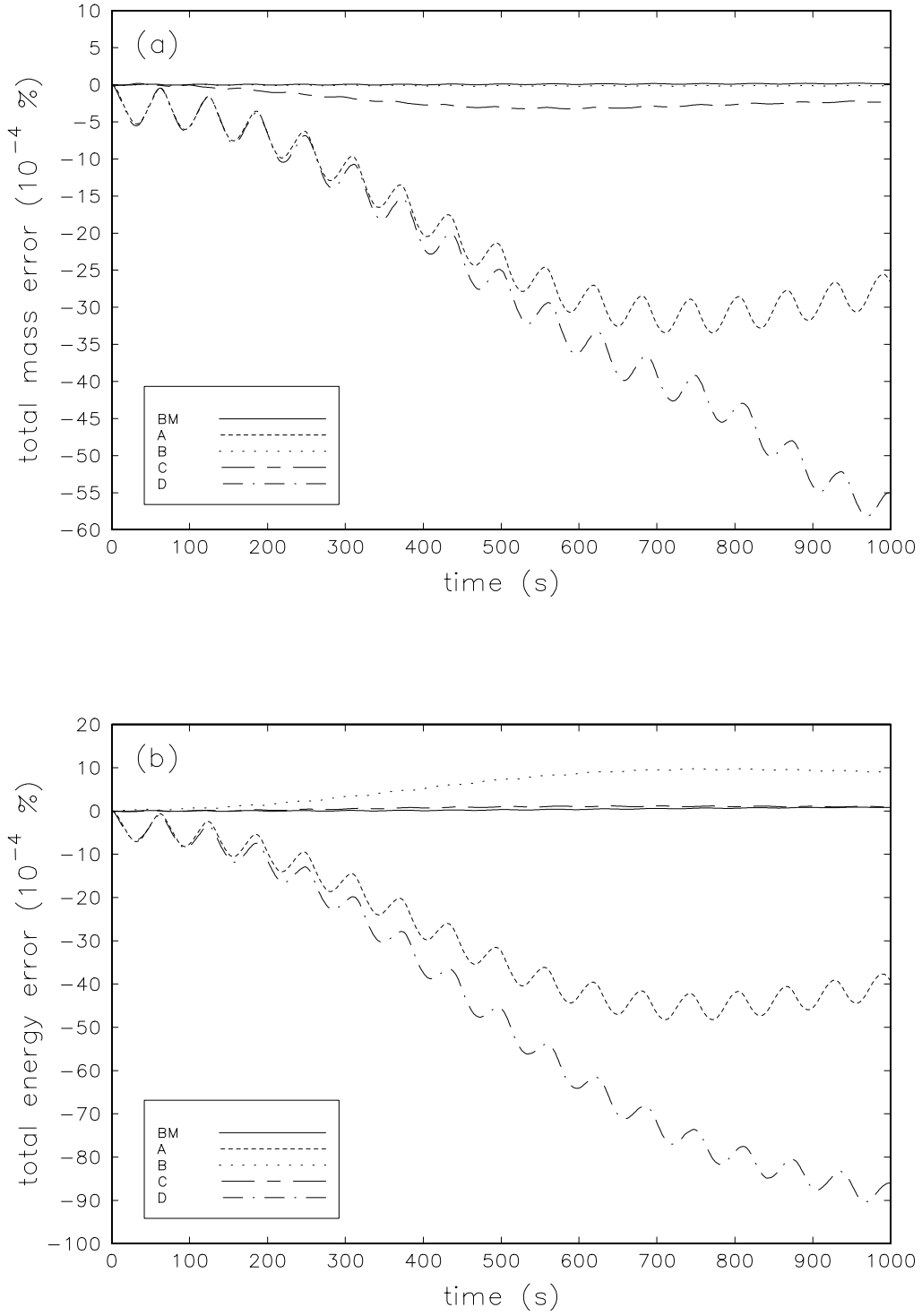


Fig. 2.10. Time series of (a) total mass error, and (b) total energy error from simulations using the five equation sets. Error is expressed as 10^{-4} % of the total mass/energy at the beginning of the simulation.

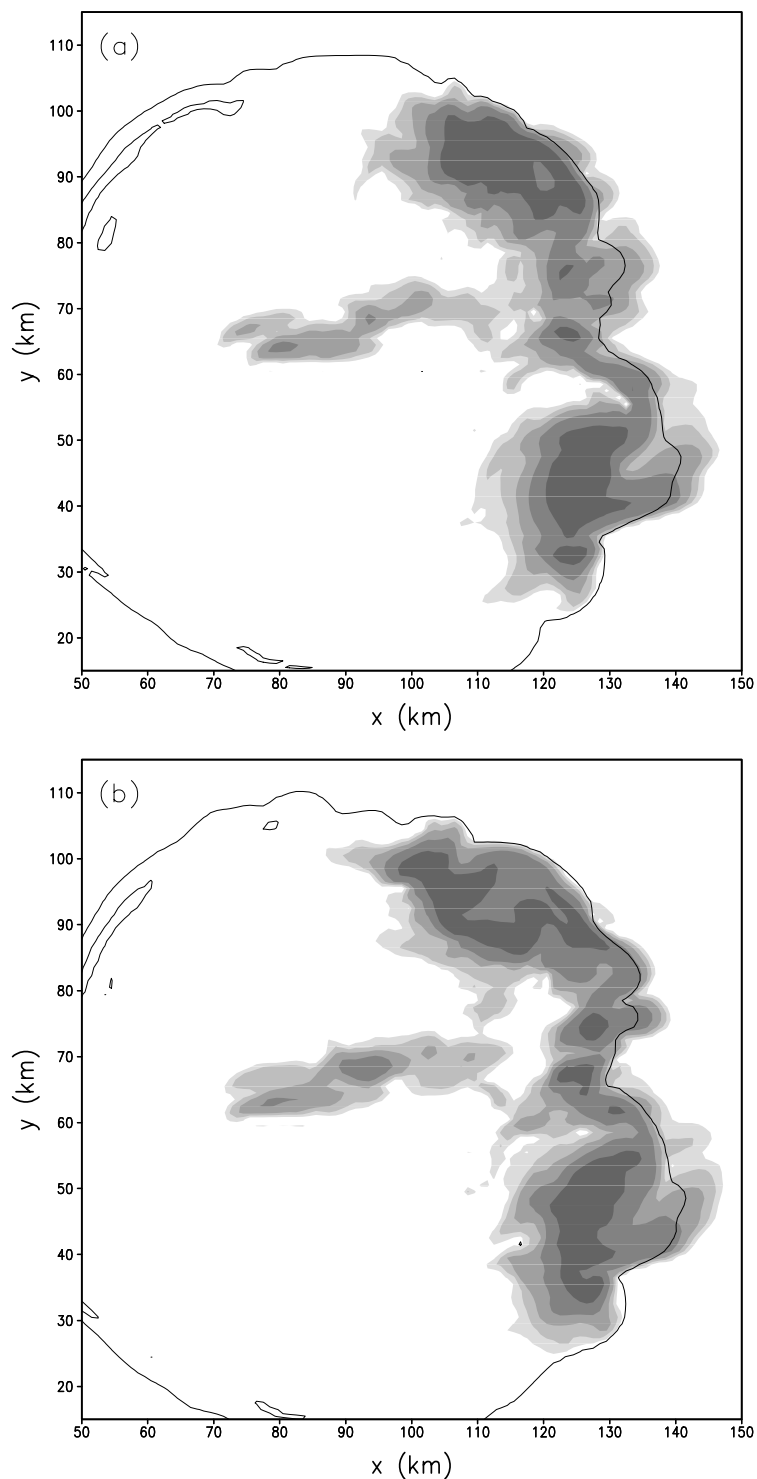


Fig. 2.11. Surface reflectivity (shaded) from: (a) a simulation using equation set A; and (b) a simulation using the new equation set. The five levels of shading correspond to, from lightest to darkest, 10-20, 20-30, 30-40, 40-50, and >50 dBZ. A black contour shows the $-1\text{ }^{\circ}\text{C}$ perturbation potential temperature contour, which indicates the approximate position of the surface gust front.

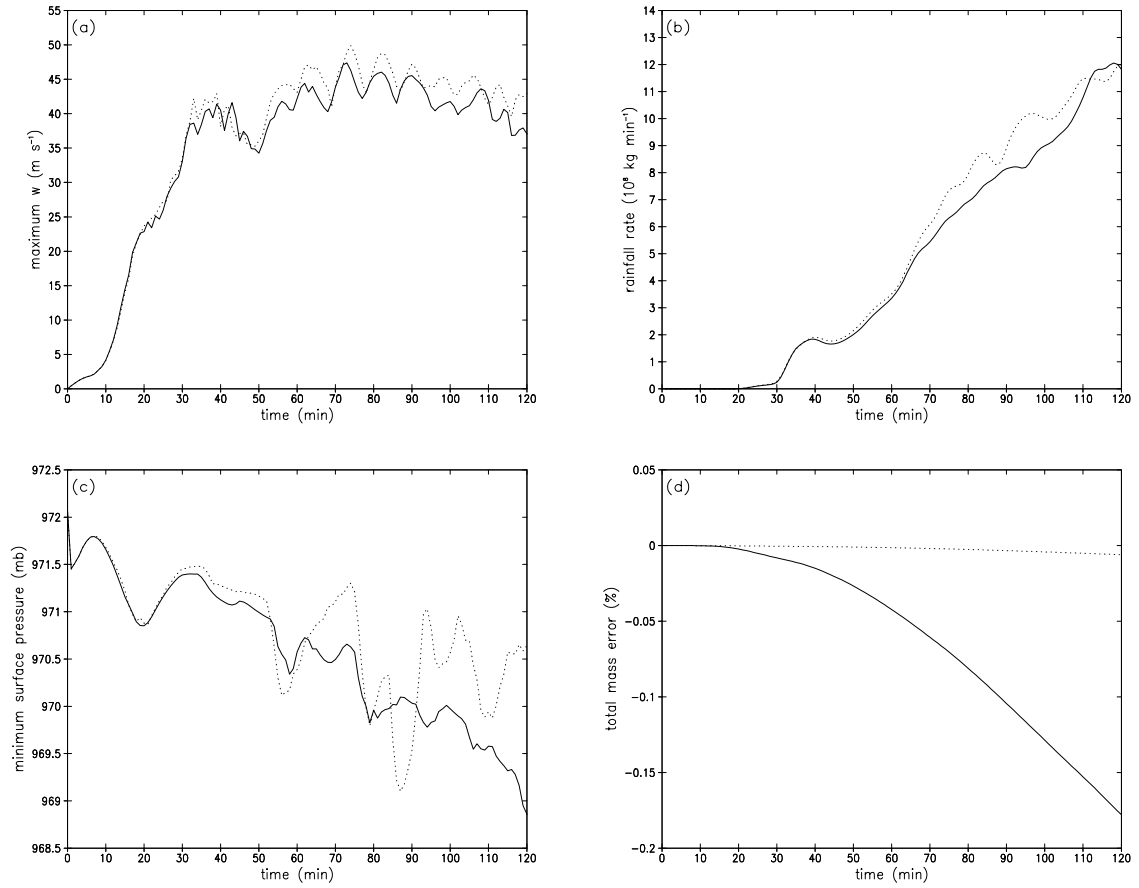


Fig. 2.12. Time series from a simulation using equation set A (solid lines) and a simulation using the new (i.e., benchmark) equation set (dotted lines). The time-resolution of the data is 1 min. (a) Maximum vertical velocity (m s^{-1}); (b) rainfall rate (10^8 kg min^{-1}); (c) minimum surface pressure (mb); and (d) total mass error (%).

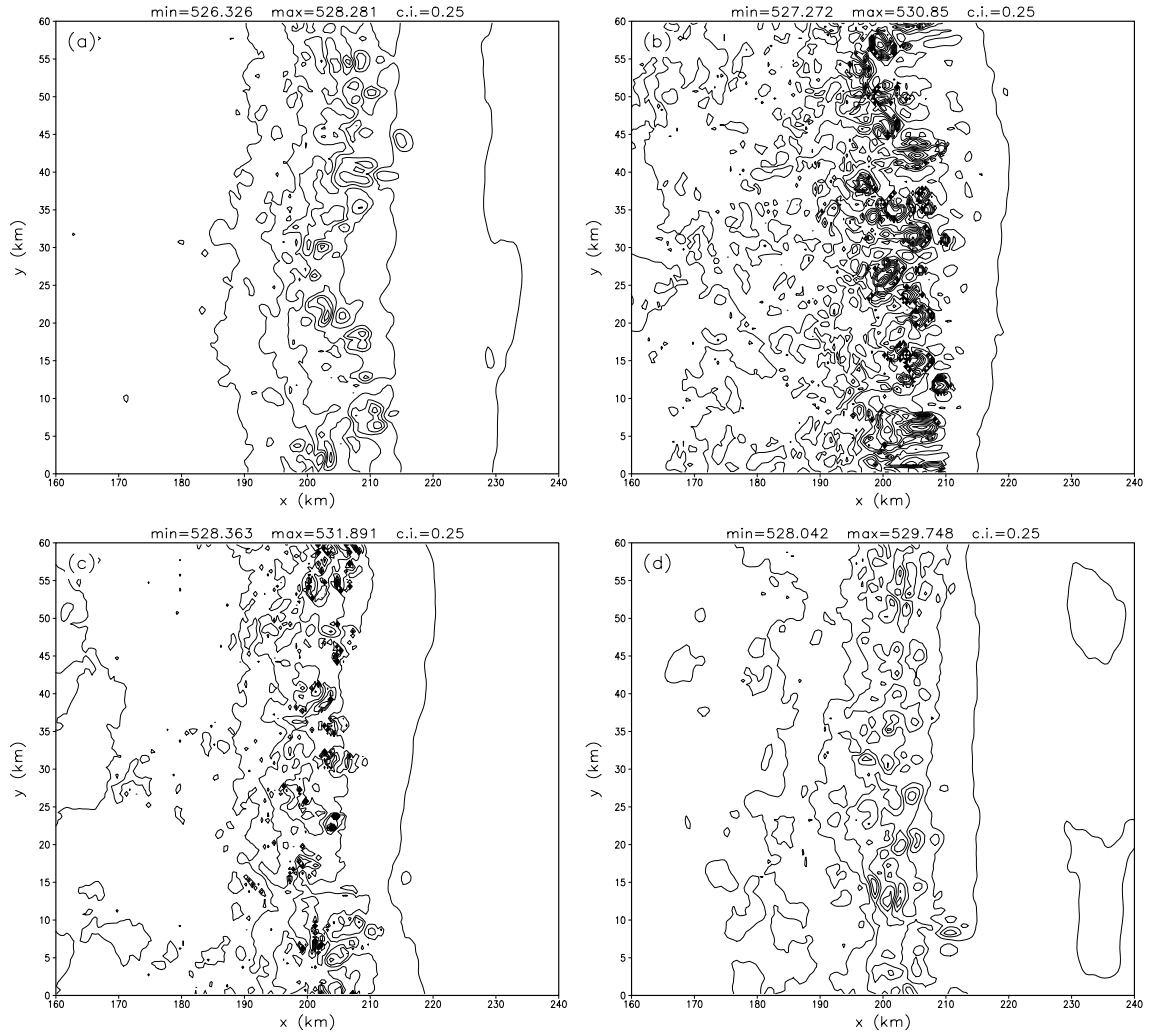


Fig. 2.13. Pressure at 5.25 km above ground at 3 hr for: (a) a simulation using equation set A; and simulations using the new (i.e., benchmark) equation set, where: (b) the diabatic term is computed on the large time steps; (c) the diabatic term is computed on the large time steps and the tendency from the last large step is also applied during the small (acoustic) time steps; and (d) the diabatic term is computed on the small (acoustic) time steps. The contour interval is 0.5 mb.

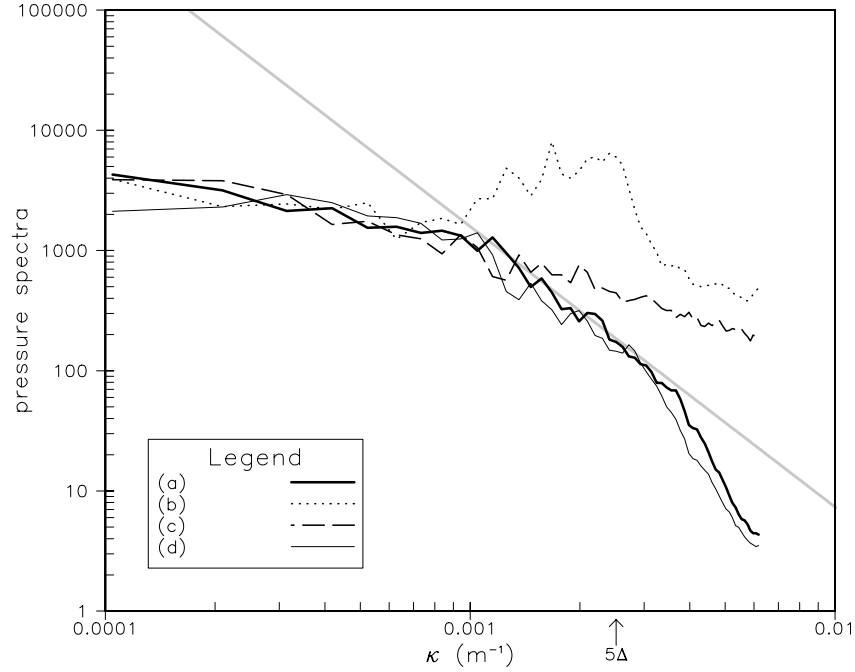


Fig. 2.14. One-dimensional (y-direction) pressure spectra at 5 km above ground. The four curves refer to the four simulations represented in Fig. 2.13: (a) a simulation using equation set A; and simulations using the new (i.e., benchmark) equation set, where: (b) the diabatic term is computed on the large time steps; (c) the diabatic term is computed on the large time steps and the tendency from the last large step is also applied during the small time steps; and (d) the diabatic term is computed on the small time steps. The spectra were computed using output every 2 min for 1 hr (from 120 min to 180 min). The thick gray line has a slope of $\kappa^{-7/3}$.

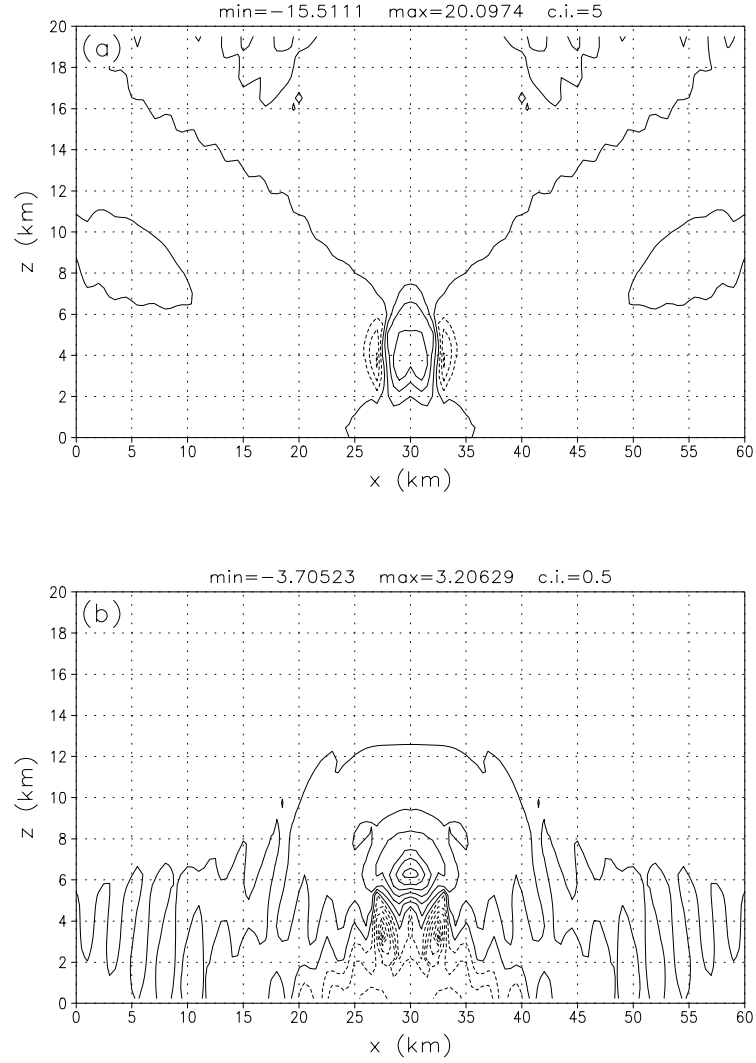


Fig. 2.15. Results of two-dimensional simulation when $\alpha_d = 0.02$. (a) Vertical velocity is contoured every 5 m s^{-1} , with negative contours dashed. (b) Perturbation pressure is contoured every 0.5 mb , with negative contours dashed.

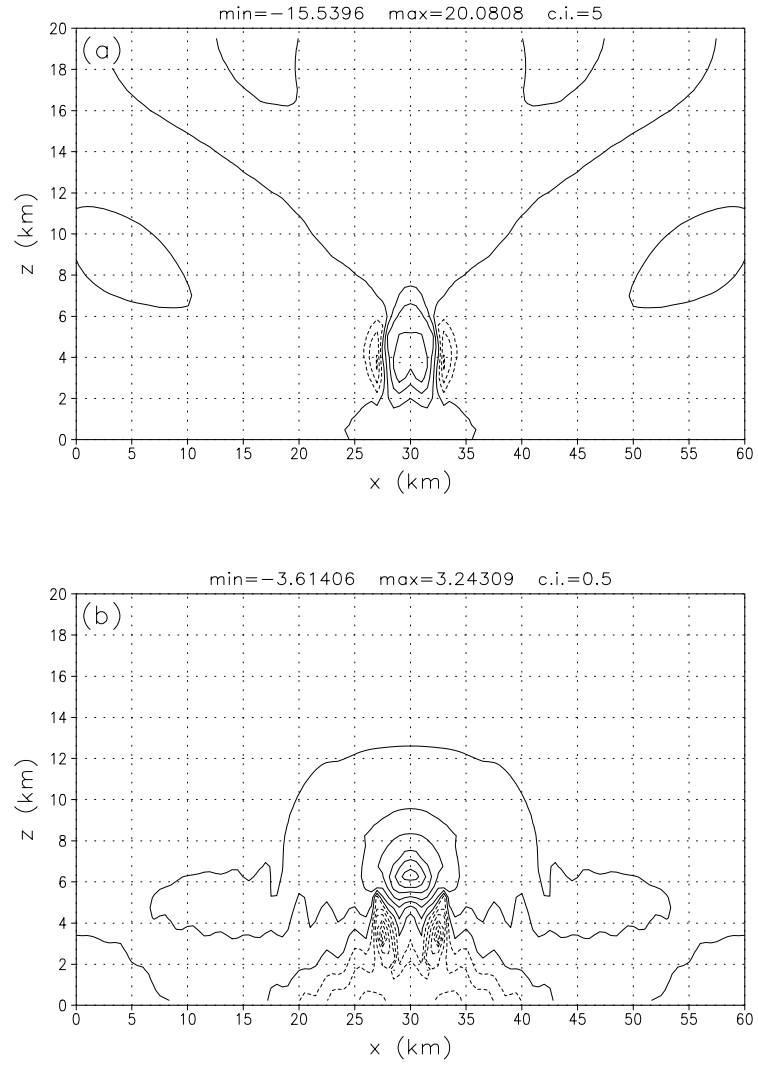


Fig. 2.16. As in Fig. 2.15, but for $\alpha_d = 0.05$.

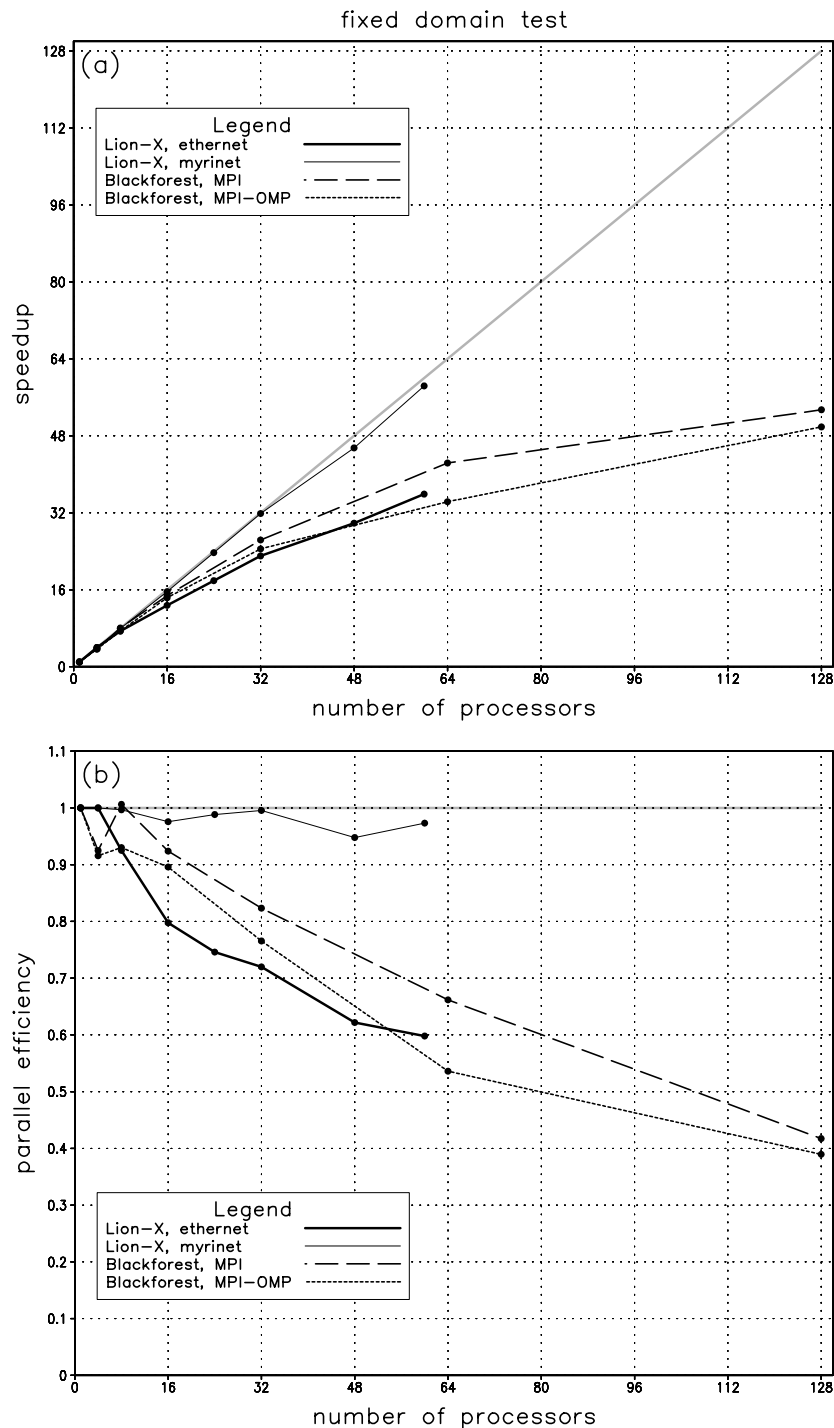


Fig. 2.17. Parallel performance for the “fixed-domain” test: (a, top) speedup versus number of processors; and (b, bottom) parallel efficiency for the same data. The thick gray line represents ideal performance; the thick solid line is output from the Linux cluster (Lion-X) using ethernet communications; the thin solid line is output from Lion-X using myrinet communications; the long dashed line is output from the IBM-SP (Blackforest) using only MPI; and the short dashed line is output from Blackforest using OpenMP within nodes and MPI between nodes.

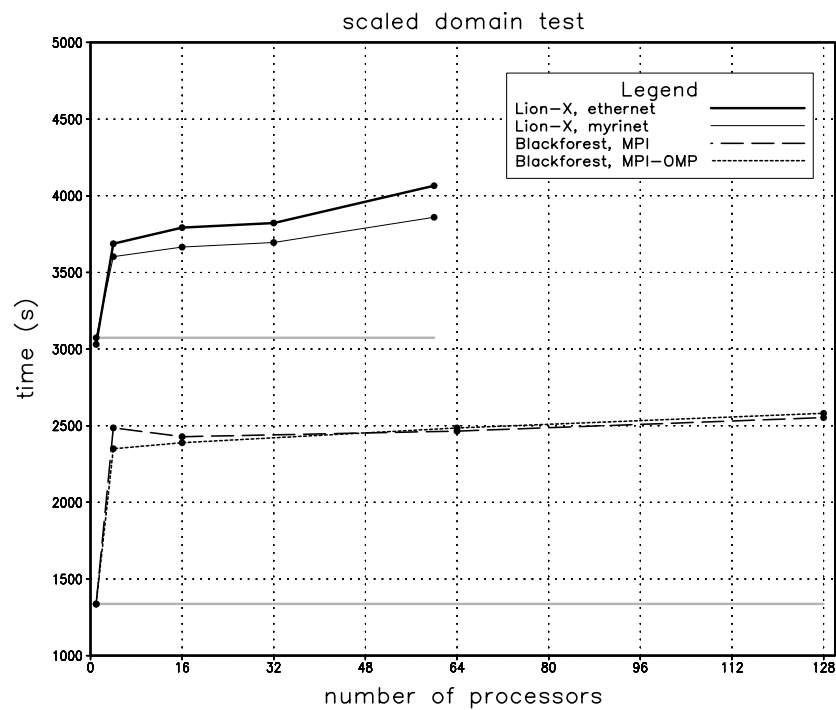


Fig. 2.18. Parallel performance for the “scaled-domain” test, in which the number of grid points per processor remains the same. Total time for the simulations is plotted versus number of processors used. Line styles are the same as for Fig. 2.17.

References

- Adlerman, E. J., and K. K. Droegemeier, 2002: The sensitivity of numerically simulated cyclic mesocyclogenesis to variations in model physical and computational parameters. *Mon. Wea. Rev.*, **130**, 2671-2691.
- Arakawa, A., and V. R. Lamb, 1977: Computational design of the basic dynamical process of the UCLA general circulation model, *Methods in Computational Physics*, **17**, Academic Press, 173-265.
- Bannon, P. R., 1996: On the anelastic approximation for a compressible atmosphere. *J. Atmos. Sci.*, **53**, 3618-3628.
- Bannon, P. R., 2002: Theoretical foundations for models of moist convection. *J. Atmos. Sci.*, **59**, 1967-1982.
- Betts, A. K., 1973: Non-precipitating cumulus convection and its parameterization. *Quart. J. Roy. Meteor. Soc.*, **99**, 178-196.
- Bjerknes, J., 1938: Saturated-adiabatic ascent of air through dry-adiabatically descending environment. *Quart. J. Roy. Meteor. Soc.*, **64**, 325-330.
- Bluestein, H. B., 1986: Fronts and jet streaks: A theoretical perspective. *Mesoscale Meteorology and Forecasting*, P. S. Ray, Ed., Amer. Meteor. Soc., 173-215.
- Blyth, A. M., W. A. Cooper, and J. B. Jensen, 1988: A study of the source of entrained air in Montana cumuli. *J. Atmos. Sci.*, **45**, 3944-3964.
- Bolton, D., 1980: The computation of equivalent potential temperature. *Mon. Wea. Rev.*, **108**, 1046-1053.
- Braun, S. A., and W.-K. Tao, 2000: Sensitivity of high-resolution simulations of Hurricane Bob (1991) to planetary boundary layer parameterizations. *Mon. Wea. Rev.*, **128**, 3941-3961.
- Bryan, G. H., and J. Michael Fritsch, 2000: Moist absolute instability: The sixth static stability state. *Bull. Amer. Meteor. Soc.*, **81**, 1207-1230.
- Bryan, G. H., and J. Michael Fritsch, 2001: On adequate resolution for the simulation of deep moist convection: Theory and preliminary simulations. Preprints, *Ninth Conference on Mesoscale Processes*, Fort Lauderdale, FL, Amer. Meteor. Soc., 288-292.

- Bryan, G. H., and J. Michael Fritsch, 2002a: A benchmark simulation for moist nonhydrostatic numerical models. *Mon. Wea. Rev.*, **130**, 2917-2928.
- Bryan, G. H., and J. Michael Fritsch, 2002b: What is appropriate resolution for simulation of thunderstorms? An answer from a turbulence perspective. Preprints, *21st Conference on Severe Local Storms*, San Antonio, TX, Amer. Meteor. Soc., 255-258.
- Bryan, G. H., and J. Michael Fritsch, 2002c: A reevaluation of ice-liquid water potential temperature. *Mon. Wea. Rev.*, accepted pending minor revisions.
- Carpenter, R. L., Jr., K. K. Droegemeier, and A. M. Blyth, 1998a: Entrainment and detrainment in numerically simulated cumulus congestus clouds. Part I: General results. *J. Atmos. Sci.*, **55**, 3417-3432.
- Carpenter, R. L., Jr., K. K. Droegemeier, and A. M. Blyth, 1998b: Entrainment and detrainment in numerically simulated cumulus congestus clouds. Part II: Cloud budgets. *J. Atmos. Sci.*, **55**, 3433-3439.
- Carpenter, R. L., Jr., K. K. Droegemeier, and A. M. Blyth, 1998c: Entrainment and detrainment in numerically simulated cumulus congestus clouds. Part III: Parcel analysis. *J. Atmos. Sci.*, **55**, 3440-3455.
- Chagnon, J. M., and P. R. Bannon, 2001: Hydrostatic and geostrophic adjustment in a compressible atmosphere: Initial response and final equilibrium to an instantaneous localized heating. *J. Atmos. Sci.*, **58**, 3776-3792.
- Davis, C. A., 2002: Probing rotationally dominated mesoscale convective systems. Preprints, *21st Conference on Severe Local Storms*, San Antonio, TX, Amer. Meteor. Soc., 109-112.
- Deardorff, J. W., 1976: Usefulness of liquid-water potential temperature in a shallow-cloud model. *J. Appl. Meteor.*, **15**, 98-102.
- Deardorff, J. W., 1980: Stratocumulus-capped mixed layer derived from a three-dimensional model. *Bound.-Layer Meteor.*, **18**, 495-527.
- Doswell, C. A., III: 2001: Severe convective storms – an overview. *Severe Convective Storms, Meteor. Monogr.*, No. 50, Amer. Meteor. Soc., 1-26.
- Droegemeier, K. K., and R. B. Wilhelmson, 1987: Numerical simulation of thunderstorm outflow dynamics. Part I: Outflow sensitivity experiments and turbulence

- dynamics. *J. Atmos. Sci.*, **44**, 1180-1210.
- 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.
- Durran, D. K., and J. B. Klemp, 1982: On the effects of moisture on the Brunt-Väisälä Frequency. *J. Atmos. Sci.*, **39**, 2152-2158.
- Emanuel, K. A., 1991: A scheme for representing cumulus convection in large-scale models. *J. Atmos. Sci.*, **48**, 2313-2335.
- Emanuel, K. A., 1994: *Atmospheric Convection*. Oxford, 580 pp.
- Gordon, N. D., 1981: Comments on "A three-dimensional numerical model of an isolated thunderstorm. Part II: Dynamics of updraft splitting and mesovortex couplet evolution". *J. Atmos. Sci.*, **38**, 1798.
- Grell, G. A., and D. Dévényi, 2001: Parameterized convection with ensemble closure/feedback assumptions. Preprints, *Ninth Conference on Mesoscale Processes*, Fort Lauderdale, FL, Amer. Meteor. Soc., 12-16.
- Hall, W. D., 1980: A detailed microphysical model within a two-dimensional dynamic framework: Model description and preliminary results. *J. Atmos. Sci.*, **37**, 2486-2507.
- Hill, R. J., and J. M. Wilczak, 1995: Pressure structure functions and spectra for locally isotropic turbulence. *J. Fluid Mech.*, **296**, 247-269.
- Hirschberg, P. A., and J. M. Fritsch, 1992: Reply. *Mon. Wea. Rev.*, **120**, 2402-2404.
- Hirschberg, P. A., and J. M. Fritsch, 1993: On understanding height tendency. *Mon. Wea. Rev.*, **121**, 2646-2661.
- Kain, J. S., and J. M. Fritsch, 1990: A one-dimensional entraining/detraining plume model and its application in convective parameterization. *J. Atmos. Sci.*, **47**, 2784-2802.
- Kain, J. S., and J. M. Fritsch, 1998: Multiscale convective overturning in mesoscale convective systems: Reconciling observations, simulations, and theory. *Mon. Wea. Rev.*, **126**, 2254-2273.
- Kankiewicz, J. A., 1998: The fine-scale characteristics of the 7-8 May 1995 squall line as revealed by ELDORA. Atmospheric Science Paper No. 657, Colorado State

- University, Fort Collins, CO, 116 pp.
- Kessler, E., 1969: *On the Distribution and Continuity of Water Substance in Atmospheric Circulation, Meteor. Monogr.*, No. 32, Amer. Meteor. Soc., 84 pp.
- Kingsmill, D. E., and R. A. Houze Jr., 1999: Thermodynamic characteristics of air flowing into and out of precipitating convection over the west Pacific warm pool. *Quart. J. Roy. Meteor. Soc.*, **125**, 1209-1229.
- Klemp, J. B., and R. B. Wilhelmson, 1978: The simulation of three-dimensional convective storm dynamics. *J. Atmos. Sci.*, **35**, 1070-1096.
- Klemp, J. B., and R. Rotunno, 1983: A study of the tornadic region within a supercell thunderstorm. *J. Atmos. Sci.*, **40**, 359-377.
- Knupp, K. R., and W. R. Cotton, 1982: An intense, quasi-steady thunderstorm over mountainous terrain. Part II: Doppler radar observations of the storm morphological structure. *J. Atmos. Sci.*, **39**, 343-358.
- Leary, C. A., and R. A. Houze, Jr., 1979: The structure and evolution of convection in a tropical cloud cluster. *J. Atmos. Sci.*, **36**, 437-457.
- Lilly, D. K., 1960: On the theory of disturbances in a conditionally unstable atmosphere. *Mon. Wea. Rev.*, **88**, 1-17.
- Lilly, D. K., 1986: Instabilities. *Mesoscale Meteorology and Forecasting*, P. S. Ray, Ed., Amer. Meteor. Soc., 259-271.
- Lin, Y.-L., R. D. Farley, and H. D. Orville, 1983: Bulk parameterization of the snow field in a cloud model. *J. Climate and Appl. Meteor.*, **22**, 1065-1092.
- Lumley, J., and H. Panofsky, 1964: *The Structure of Atmospheric Turbulence*. Wiley-Interscience, 239 pp.
- Parker, M. D., and R. H. Johnson, 2000: Organizational modes of midlatitude mesoscale convective systems. *Mon. Wea. Rev.*, **128**, 3413-3436.
- Reisner, J., R. M. Rasmussen, and R. T. Brientjes, 1998: Explicit forecasting of supercooled liquid water in winter storms using the MM5 mesoscale model. *Quart. J. Roy. Meteor. Soc.*, **124**, 1071-1107.
- Rogers, R. F., and J. M. Fritsch, 2001: Surface cyclogenesis from convectively driven amplification of midlevel mesoscale convective vortices. *Mon. Wea. Rev.*, **129**, 605-637.

- Rotunno, R., and J. B. Klemp, 1982: The influence of the shear-induced pressure gradient on thunderstorm motion. *Mon. Wea. Rev.*, **110**, 136-151.
- Rotunno, R., J. B. Klemp, and M. L. Weisman, 1988: A theory for strong, long-lived squall lines. *J. Atmos. Sci.*, **45**, 463-485.
- Rutledge, S. A., and P. V. Hobbs, 1983: The mesoscale and microscale structure and organization of clouds and precipitation in midlatitude cyclones. VIII: A model for the “seeder-feeder” process in warm-frontal rainbands. *J. Atmos. Sci.*, **40**, 1185-1206.
- Schlesinger, R. E., 1980: A three-dimensional numerical model of an isolated thunderstorm. Part II: Dynamics of updraft splitting and mesovortex couplet evolution. *J. Atmos. Sci.*, **37**, 395-420.
- Schlesinger, R. E., 1981: Reply. *J. Atmos. Sci.*, **38**, 1798-1799.
- Skamarock, W. C., and J. B. Klemp, 1992: The stability of time-split numerical methods for the hydrostatic and nonhydrostatic elastic equations. *Mon. Wea. Rev.*, **120**, 2109-2127.
- Skamarock, W. C., and J. B. Klemp, 1994: Efficiency and accuracy of the Klemp-Wilhelmson time-splitting technique. *Mon. Wea. Rev.*, **122**, 2623-2630.
- Soong, S.-T., and Y. Ogura, 1973: A comparison between axi-symmetric and slab-symmetric cumulus cloud models. *J. Atmos. Sci.*, **30**, 879-893.
- Stauffer, D. R., R. C. Muñoz, and N. L. Seaman, 2001: On the importance of saturation effects in the turbulence scheme of a mesoscale model. Preprints, *9th Conference on Mesoscale Processes*, Fort Lauderdale, FL, Amer. Meteor. Soc., 1-5.
- Straka, J. M., R. B. Wilhelmson, L. J. Wicker, J. R. Anderson, and K. K. Droegemeier, 1993: Numerical solutions of a non-linear density current: A benchmark solution and comparisons. *Int. J. Numer. Methods*, **17**, 1-22.
- Tao, W.-K., and J. Simpson, 1989: Modeling study of a tropical squall-type convective line. *J. Atmos. Sci.*, **46**, 177-202.
- Tripoli, G. J., 1992: A nonhydrostatic mesoscale model designed to simulate scale interaction. *Mon. Wea. Rev.*, **120**, 1342-1359.
- Tripoli, G. J., and W. R. Cotton, 1981: The use of ice-liquid water potential temperature as a thermodynamic variable in deep atmospheric models. *Mon. Wea. Rev.*, **109**,

1094-1102.

- Wakimoto, R. M., W.-C. Lee, H. B. Bluestein, C.-H. Liu, and P. H. Hildebrand, 1996: ELDORA observations during VORTEX 95. *Bull. Amer. Meteor. Soc.*, **77**, 1465-1481.
- Walko, R. L., W. R. Cotton, G. Feingold, and B. Stevens, 2000: Efficient computation of vapor and heat diffusion between hydrometeors in a numerical model. *Atmos. Research*, **53**, 171-183.
- Weisman, M. L., 1993: The genesis of severe, long-lived bow echoes. *J. Atmos. Sci.*, **50**, 645-670.
- Weisman, M. L., and J. B. Klemp, 1982: The dependence of numerically simulated convective storms on vertical wind shear and buoyancy. *Mon. Wea. Rev.*, **110**, 504-520.
- Weisman, M. L., and R. Rotunno, 2000: The use of vertical wind shear versus helicity in interpreting supercell dynamics. *J. Atmos. Sci.*, **57**, 1452-1472.
- Weisman, M. L., and R. J. Trapp, 2002: Outstanding issues related to the bow echo-derecho component of BAMEX. *21st Conference on Severe Local Storms*, San Antonio, TX, Amer. Meteor. Soc., 113-116.
- Weisman, M. L., W. C. Skamarock, and J. B. Klemp, 1997: The resolution dependence of explicitly modeled convective systems. *Mon. Wea. Rev.*, **125**, 527-548.
- Wicker, L. J., and R. B. Wilhelmson, 1995: Simulation and analysis of tornado development and decay within a three-dimensional supercell thunderstorm. *J. Atmos. Sci.*, **52**, 2675-2703.
- Wicker, L. J., and W. C. Skamarock, 1998: A time-splitting scheme for the elastic equations incorporating second-order Runge-Kutta time differencing. *Mon. Wea. Rev.*, **126**, 1992-1999.
- Wicker, L. J., and W. C. Skamarock, 2002: Time splitting methods for elastic models using forward time schemes. *Mon. Wea. Rev.*, **130**, 2088-2097.
- Wilhelmson, R., and Y. Ogura, 1972: The pressure perturbation and the numerical modeling of a cloud. *J. Atmos. Sci.*, **29**, 1295-1307.
- Wilhelmson, R. B., and C.-S. Chen, 1982: A simulation of the development of successive cells along a cold outflow boundary. *J. Atmos. Sci.*, **39**, 1466-1483.

- Wurman, J., J. Straka, E. Rasmussen, M. Randall, and A. Zahari, 1997: Design and deployment of a portable, pencil-beam, pulsed, 3-cm Doppler radar. *J. Atmos. Oceanic Technol.*, **14**, 1502-1512.
- Xue, M., and S.-J. Lin, 2001: Numerical equivalence of advection in flux and advective forms and quadratically conservative high-order advection schemes. *Mon. Wea. Rev.*, **129**, 561-565.
- Yuter, S. E., and R. A. Houze, Jr., 1995a: Three-dimensional kinematic and microphysical evolution of Florida cumulonimbus. Part I: Spatial distribution of updrafts, downdrafts, and precipitation. *Mon. Wea. Rev.*, **123**, 1921-1940.
- Yuter, S. E., and R. A. Houze, Jr., 1995b: Three-dimensional kinematic and microphysical evolution of Florida cumulonimbus. Part II: Frequency distributions of vertical velocity, reflectivity, and differential reflectivity. *Mon. Wea. Rev.*, **123**, 1941-1963.
- Yuter, S. E., and R. A. Houze, Jr., 1995c: Three-dimensional kinematic and microphysical evolution of Florida cumulonimbus. Part III: Vertical mass transport, mass divergence, and synthesis. *Mon. Wea. Rev.*, **123**, 1964-1983.

Appendix
List of Symbols

Symbol	Description	Value, definition, or equation number
c_i	Specific heat of frozen water	$2106 \text{ J kg}^{-1} \text{ K}^{-1}$
c_p	Specific heat of dry air at constant pressure	$1004 \text{ J kg}^{-1} \text{ K}^{-1}$
c_l	Specific heat of liquid water	$4186 \text{ J kg}^{-1} \text{ K}^{-1}$
c_{pm}	Specific heat of moist air at constant pressure	$c_{pm} \equiv c_p + c_{pv}r_v + c_l r_l$
c_{pv}	Specific heat of water vapor at constant pressure	$1885 \text{ J kg}^{-1} \text{ K}^{-1}$
c_v	Specific heat of dry air at constant volume	$717 \text{ J kg}^{-1} \text{ K}^{-1}$
c_{vm}	Specific heat of moist air at constant volume	$c_{vm} \equiv c_v + c_{vv}r_v + c_l r_l$
c_{vv}	Specific heat of water vapor at constant volume	$1424 \text{ J kg}^{-1} \text{ K}^{-1}$
e	Water vapor pressure	
e_s	Saturation vapor pressure	
E_t	Total energy	
g	Acceleration due to gravity	9.81 m s^{-2}
H	Relative humidity	$H \equiv e/e_s$
L_f	Latent heat of freezing	$L_f = L_s - L_v$
L_s	Latent heat of sublimation	$L_s = L_{s0} - (c_{pi} - c_{pv})(T - T_0)$
L_{s0}	Reference value of L_s	$2.836 \times 10^6 \text{ J kg}^{-1}$
L_v	Latent heat of vaporization	$L_v = L_{v0} - (c_{pl} - c_{pv})(T - T_0)$
L_{v0}	Reference value of L_v	$2.5 \times 10^6 \text{ J kg}^{-1}$
N_m	Brunt-Vaisala frequency for moist air	
p	Total pressure	$p \equiv p_d + e$
p_d	Dry air pressure	
p_{00}	Reference pressure	1000 mb
r_i	Frozen water (ice) mixing ratio	
r_l	Liquid water mixing ratio	
r_s	Saturation mixing ratio with respect to water	
r_t	Total water mixing ratio	
r_v	Water vapor mixing ratio	
R	Gas constant of dry air	$287 \text{ J kg}^{-1} \text{ K}^{-1}$
R_m	Gas constant of moist air	$R_m \equiv R + R_v r_v$
R_v	Gas constant of water vapor	$461 \text{ J kg}^{-1} \text{ K}^{-1}$
t	time	
T	Temperature	
T_0	Reference temperature	273.15 K
u_i	Velocity vector	

U_i	Momentum vector	
x_i	Direction vector	
γ	Ratio of c_p to c_v	$\gamma \equiv c_p / c_v$
Γ_m	Moist adiabatic lapse rate	
δ_{ij}	Kronecker delta	
ε	Ratio of R to R_v	$\varepsilon \equiv R / R_v$
θ	Potential temperature	
θ_e	Wet equivalent potential temperature	
θ_l	Liquid water potential temperature	
θ_ρ	Density potential temperature	
π	Non-dimensional pressure	
ρ_a	Dry air density	