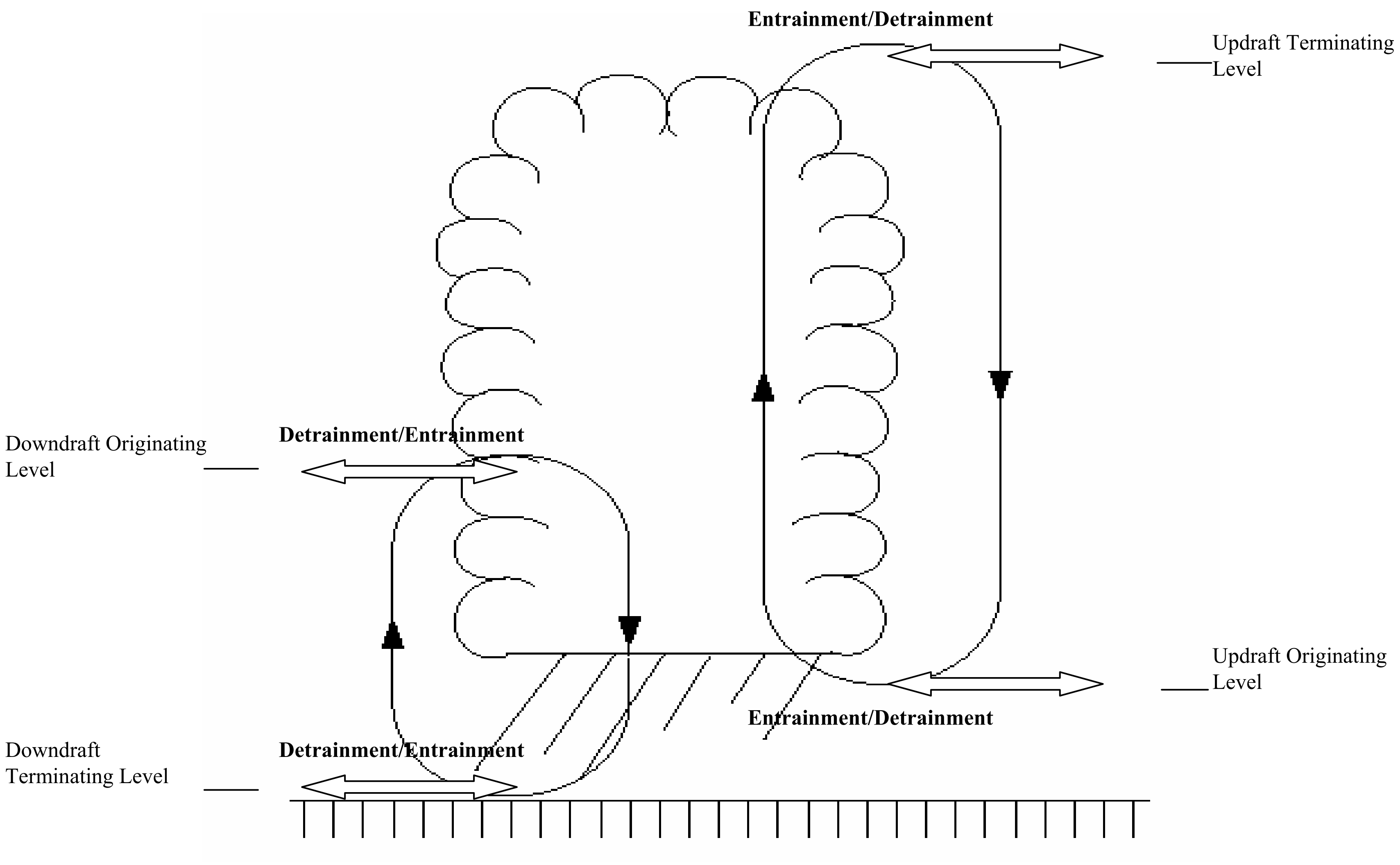
Grell Convection Scheme in RAMS

Jih-Wang Wang

2006/4/26

Grell Scheme for convection and transport



Grell Scheme Procedure

1. Find out

$$z_T \quad (\text{where } h_u(z_B) = \widetilde{h}^*(z_T))$$
$$z_B \quad (LCL)$$
$$z_0 \quad (\text{where } \widetilde{h}(z) \text{ minimum})$$

2. Check criteria to start or not

3. Determine mass flux based on ABE

4. Parameterize entrainment/detrainment rate and mass flux ratio

5. Quasi-equilibrium adjustment

6. Compute precipitation amount

Grell Scheme Features

- Entrainment/Detrainment based on cloud radius = 12km
- Entrainment/Detrainment happens only at originating/terminating level
- Mass-flux scheme, i.e., rearrangement of mass in a vertical column
- One type cloud, shallow cumulus parameterization usually accompanied
- Virtually instantaneous adjustment scheme

Ensemble Approach

- 5 different closure types with regard to cloud base mass flux – dynamic control
- 144 combinations to parameterize entrainment/detrainment rate – static control
- 6 combinations for mass flux ratio
- Sensitivity tests against closure types, mass flux ratio, and entrainment/detrainment rate
- Who knows whether it’s better or not?

My Understanding of LUT

$$[X]^{\tau} \rightarrow \langle parameterization \rangle \rightarrow [X]^{\tau+1}$$

$$[X]^{\tau} \rightarrow \left\langle \begin{matrix} [x_1] \rightarrow \langle eq1 \rangle \rightarrow [y_1] \\ [x_2] \rightarrow \langle eq2 \rangle \rightarrow [y_2] \\ [x_3] \rightarrow \langle eq3 \rangle \rightarrow [y_3] \\ \ldots \end{matrix} \right\rangle \rightarrow [X]^{\tau+1}$$

$$[X]^{\tau} \rightarrow \left\langle \begin{matrix} [x_1] \rightarrow \langle LUT1 \rangle \rightarrow [y_1] \\ [x_2] \rightarrow \langle LUT2 \rangle \rightarrow [y_2] \\ [x_3] \rightarrow \langle LUT3 \rangle \rightarrow [y_3] \\ \ldots \end{matrix} \right\rangle \rightarrow [X]^{\tau+1}$$

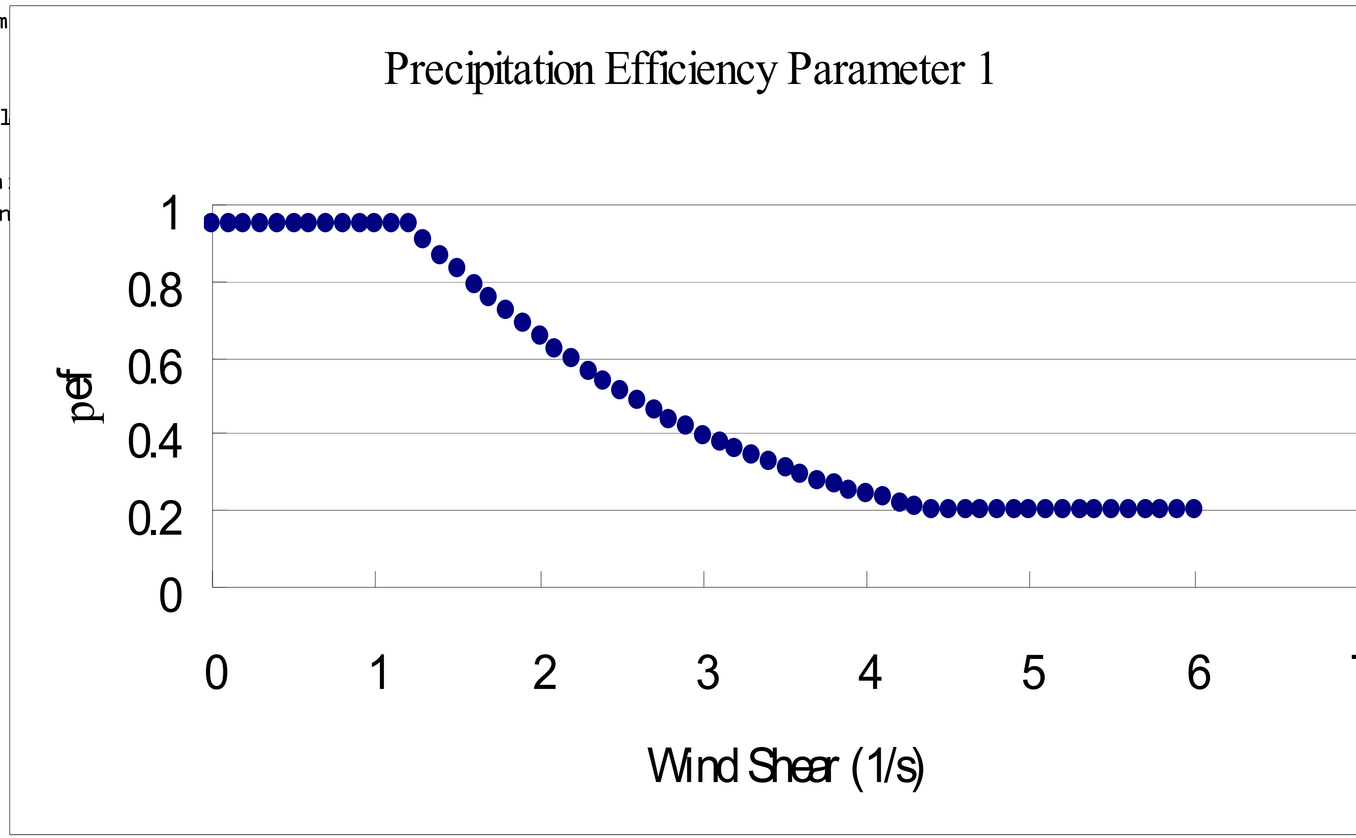
$$[X]^{\tau} \rightarrow \langle LUT \rangle \rightarrow [X]^{\tau+1}$$

```
pef=(1.591-.639*VSHEAR(I)+.0953*(VSHEAR(I)**2)-.00496*(VSHEAR(I)**3))
if(pef.gt.edtmax)pef=edtmax
if(pef.lt.edtmin)pef=edtmin

!--- cloud base precip efficiency

zkbc=z(i,kbcon(i))*3.281e-3
prezk=.02
if(zkbc.gt.3.)then
  prezk=.96729352+zkbc*(-.70034167+zkbc*(.162179896+zkbc* &
     (-1.2569798E-2+zkbc*(4.2772E-4-zkbc*5.44E-6))))
endif
if(zkbc.gt.25)then
  prezk=2.4
endif
pefb=1./(1.+prezk)
if(pefb.gt.edtmax)pefb=edtmax
if(pefb.lt.edtmin)pefb=edtmin
EDT(I)=1.-.5*(pefb+pef)
einc=edt(i)/float(maxens2+1)
do k=1,maxens2
  edtc(i,k)=edt(i)+float(k-maxens2/2-1)*einc
  edtc(i,k)=edt(i)-float(k)*einc
enddo
```

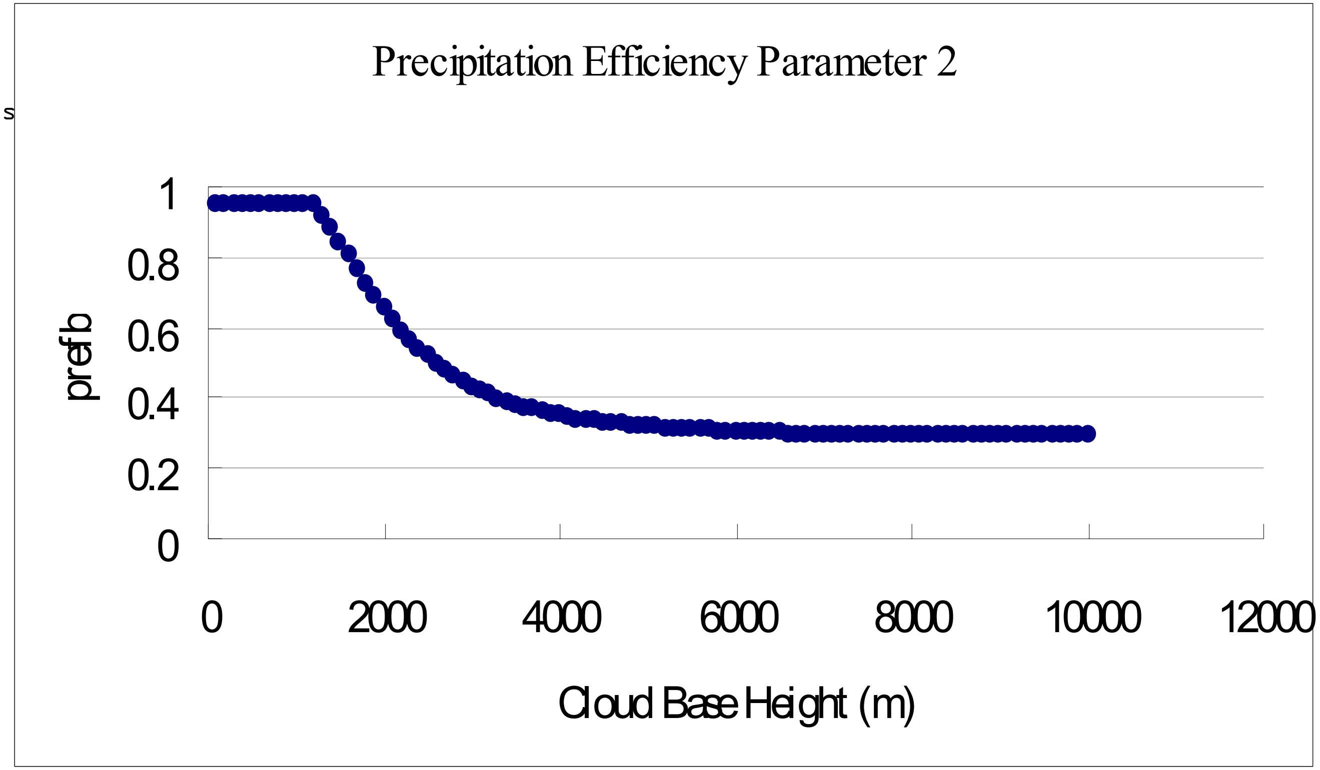
$$\left\{\begin{array}{c}\text{VSHEAR (wind shear)}\\ \text{z (cloud base height)}\end{array}\right. \rightarrow \langle\text{code}\rangle \rightarrow \text{edtc (ratio between downdraft and updraft mass flux)}$$

$$m_0 = \frac{\beta I_1 m_b}{I_2} = \xi m_b$$

```
      pef=(1.591-.639*VSHEAR(I)+.0953*(VSHEAR(I)**2) - .00496*
(VSHEAR(I)**3))
      if (pef.gt.edtmax) pef=edtmax
      if (pef.lt.edtmin) pef=edtmin

      !--- cloud base precip efficiency

      zkbc=z(i,kbcon(i))*3.281e-3
      prezk=.02
      if (zkbc.gt.3.) then
        prezk = .96729352+zkbc*(-.70034167+zkbc*(.16217896+zkbc*
&
            (-1.2569798E-2+zkbc*(4.2772E-4-zkbc*5.44E-6))))
      endif
      if (zkbc.gt.25) then
        prezk=2.4
      endif
      pefb = 1./(1.+prezk)
      if (pefb.gt.edtmax) pefb=edtm
      if (pefb.lt.edtmin) pefb=edtm
      EDT(I) = 1.-.5*(pefb+pef)
      einc = edt(i)/float(maxens2+1
      do k=1,maxens2
        edtc(i,k)=edt(i)+float(k-m a
        edtc(i,k)=edt(i)-float(k)*ein
      enddo
```



```
      pef=(1.591-.639*VSHEAR(I)+.0953*(VSHEAR(I)**2)-.00496*
(VSHEAR(I)**3))
      if (pef.gt.edtmax) pef=edtmax
      if (pef.lt.edtmin) pef=edtmin

      !--- cloud base precip efficiency

      zkbc=z(i,kbcon(i))*3.281e-3
      prezk=.02
      if (zkbc.gt.3.) then
        prezk = .96729352+zkbc*(-.70034167+zkbc*(.162179896+zkbc*
&
            (-1.2569798E-2+zkbc*(4.2772E-4-zkbc*5.44E-6))))
      endif
      if (zkbc.gt.25) then
        prezk=2.4
      endif
      pefb = 1./(1.+prezk)
      if (pefb.gt.edtmax) pefb=edtmax
      if (pefb.lt.edtmin) pefb=edtmin
      EDT(I) = 1.-.5*(pefb+pef)
      einc = edt(i)/float(maxens2+1)
      do k=1,maxens2
        edtc(i,k)=edt(i)+float(k-maxens
        edtc(i,k)=edt(i)-float(k)*einc
      enddo
```

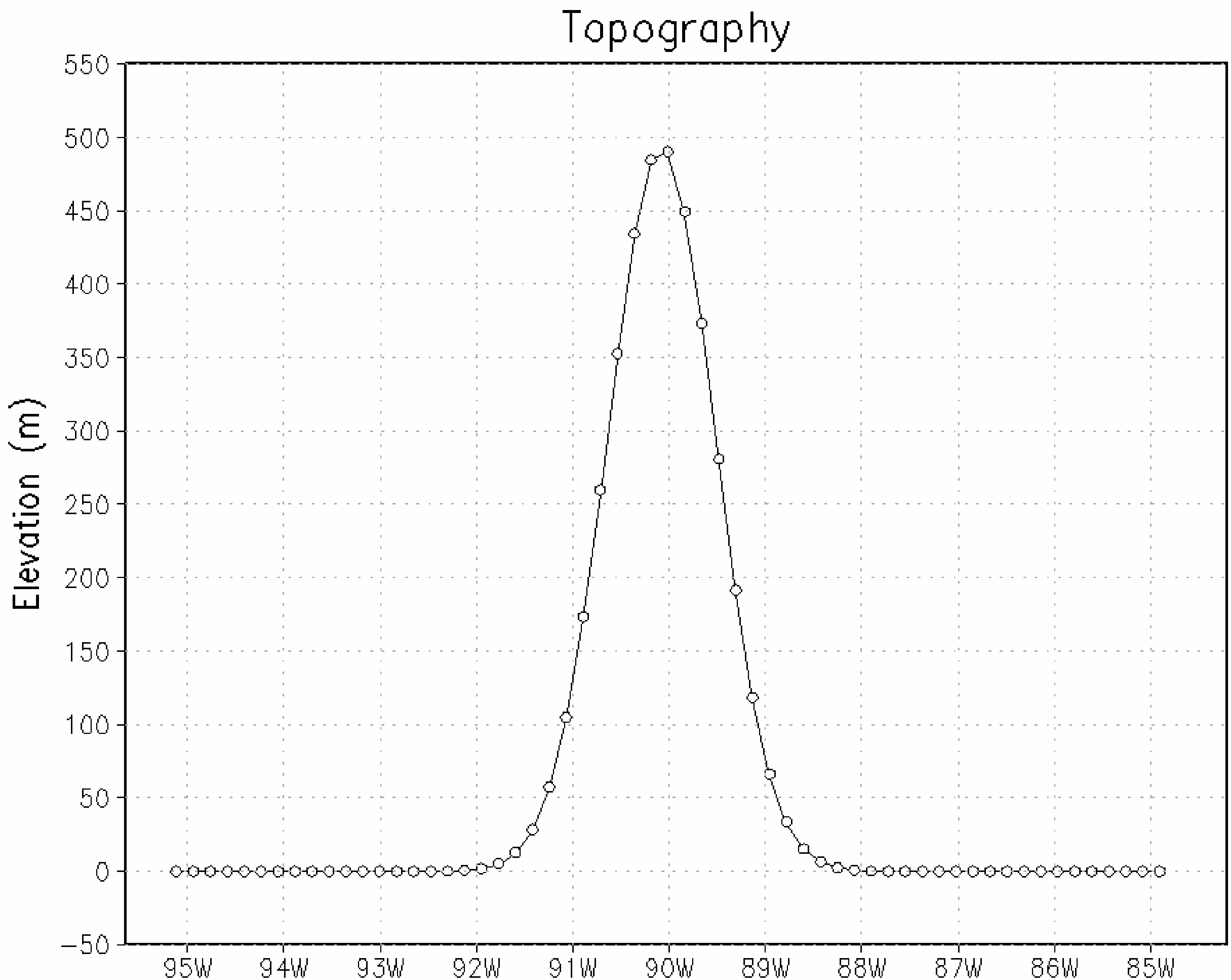


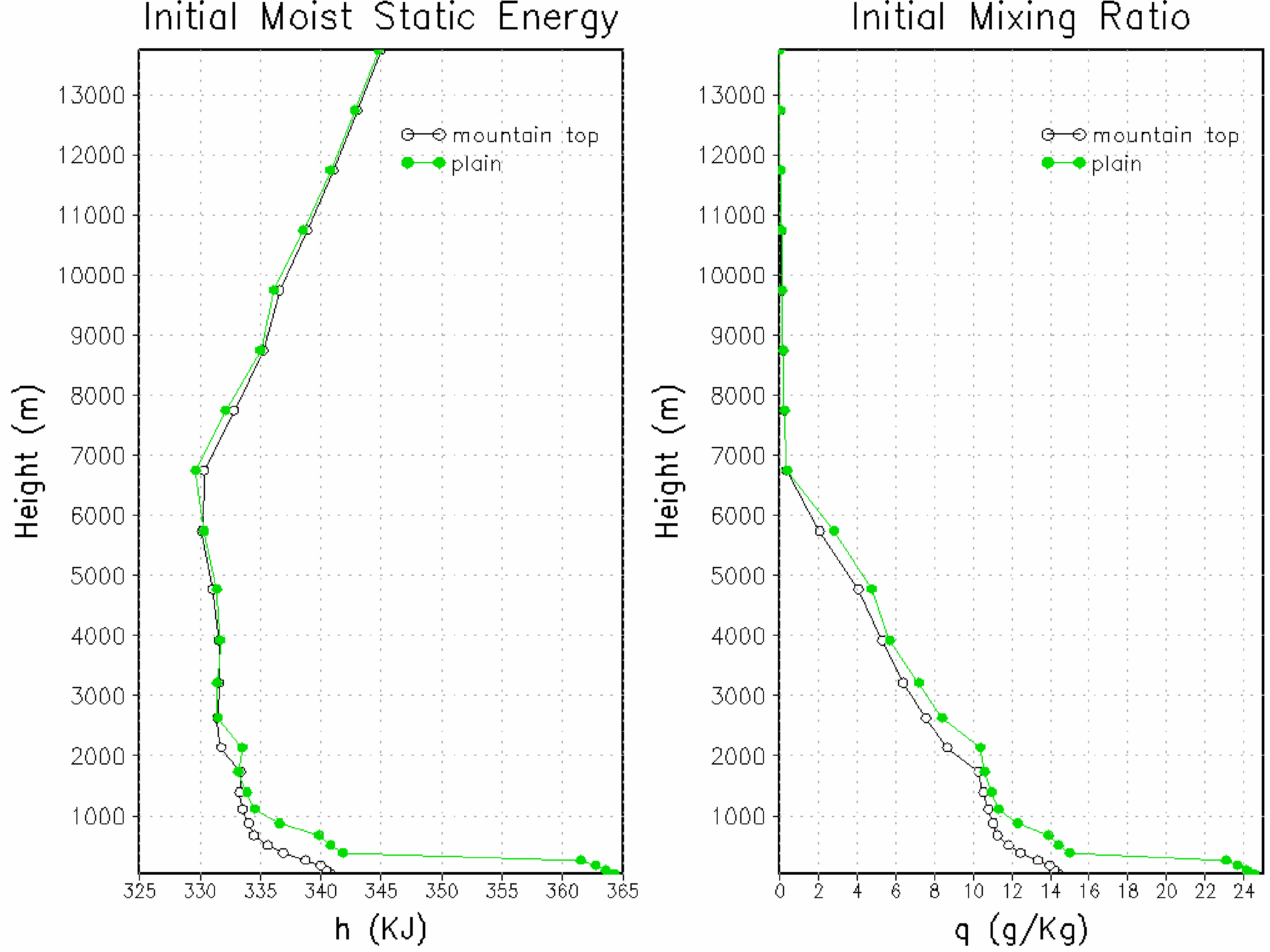
Procedure to Implement LUT Approach

- Find out the range for VSHEAR and z
- Given VSHEAR and z, run the piece of code offline to get the LUT for edtc=f(VSHEAR,z)
- Replace the piece of code with LUT reading code and linear interpolation
- Check simulation results
- Estimate model efficiency change and resources usage

RAMS Model Options and Setups

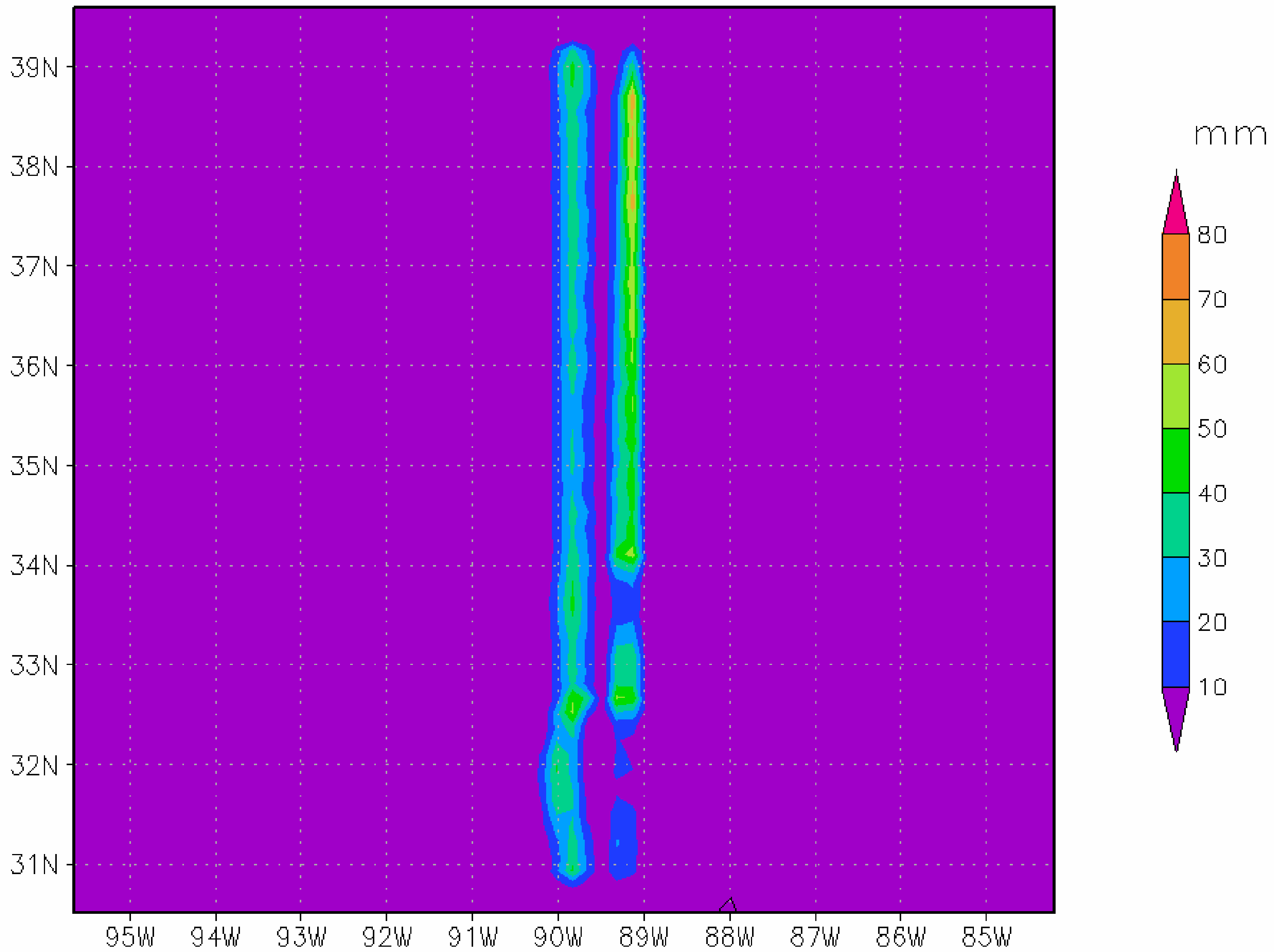
- **Basic Equations**
  Nonhydrostatic; compressible
- **Grid Stagger/Structure**
  Arakawa C grid; One Grid
- **Convective Parameterization**
  Grell scheme
- **Surface Layer Flux**
  None
- **Boundary Condition**
  Cyclic for momentum; zero gradient inflow/outflow for others
- **Radiation Parameterization**
  Harrington scheme
- **Initialization**
  Unstable sounding; horizontally homogeneous
- **Terrain**
  N-S mountain range in the middle (Gaussian function)
- **Grid Spacing**
  16 km x 16 km x (60 m ~ 1000 m)
- **Time Step**
  30 secs
- **Grid Cell Number**
  60 x 60

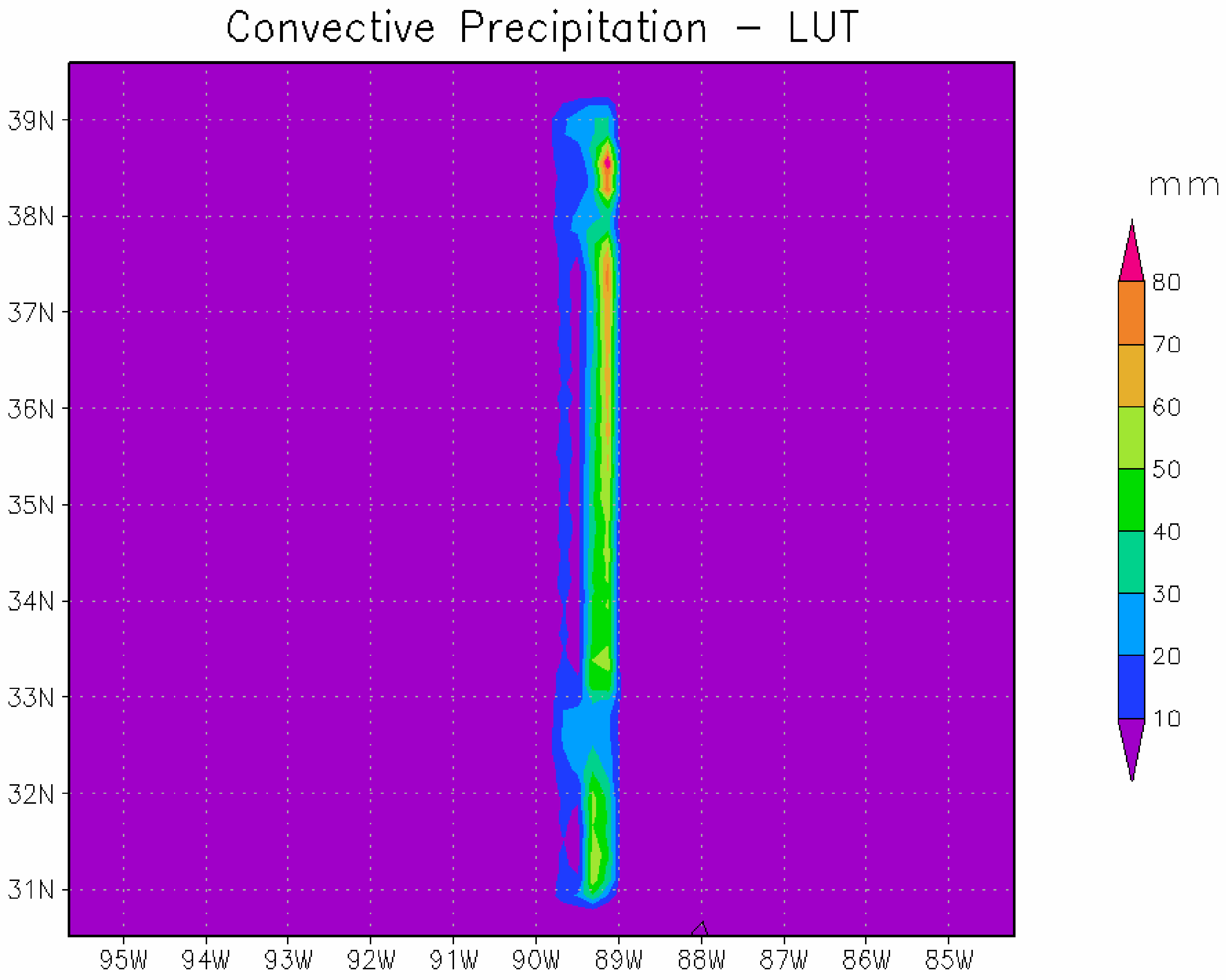

Topography
Elevation (m)
550
500
450
400
350
300
250
200
150
100
50
0
−50
95W
94W
93W
92W
91W
90W
89W
88W
87W
86W
85W



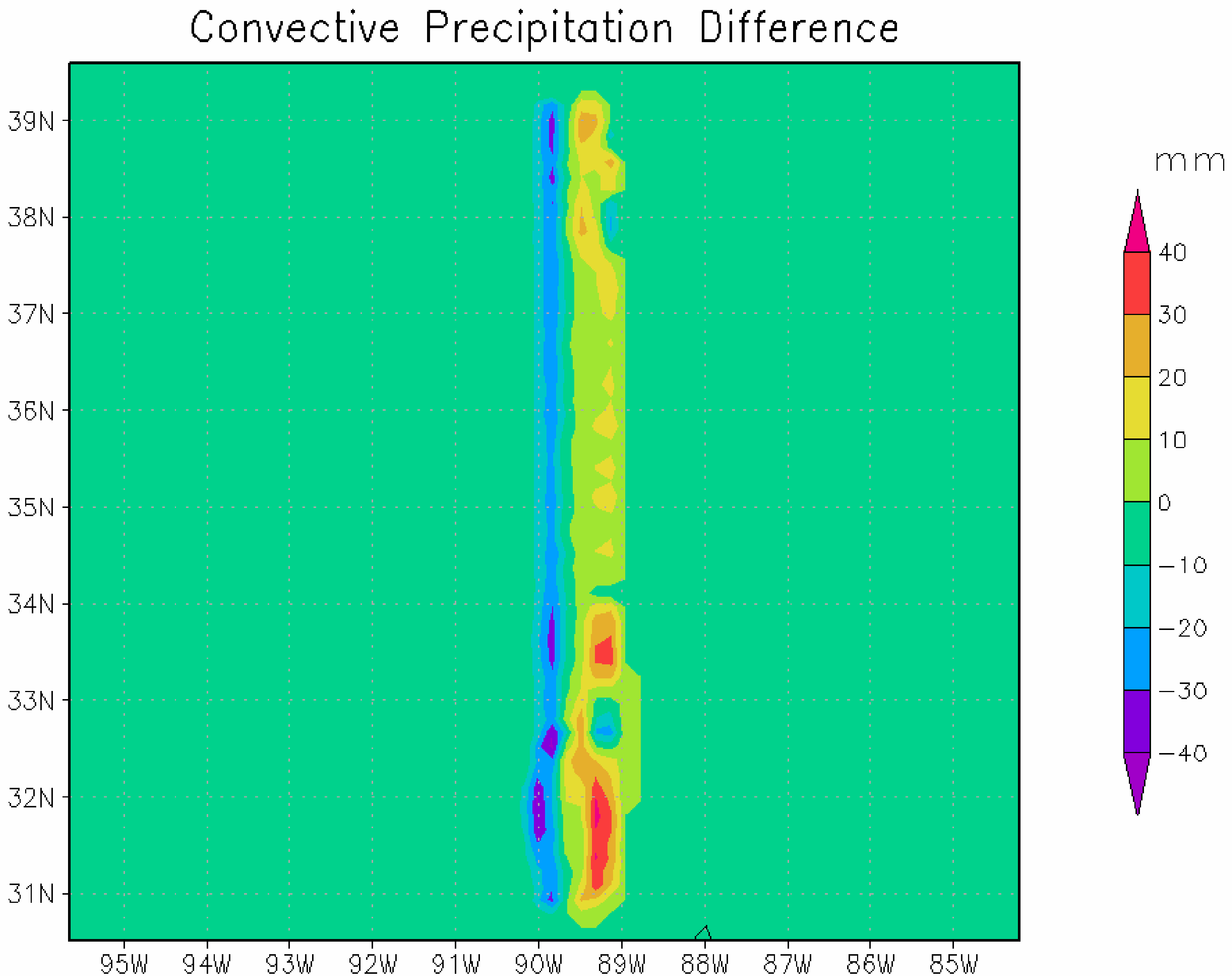
Initial Moist Static Energy
Height (m)
h (KJ)
mountain top
plain
Initial Mixing Ratio
Height (m)
q (g/Kg)
mountain top
plain

Convective Precipitation – Explicit

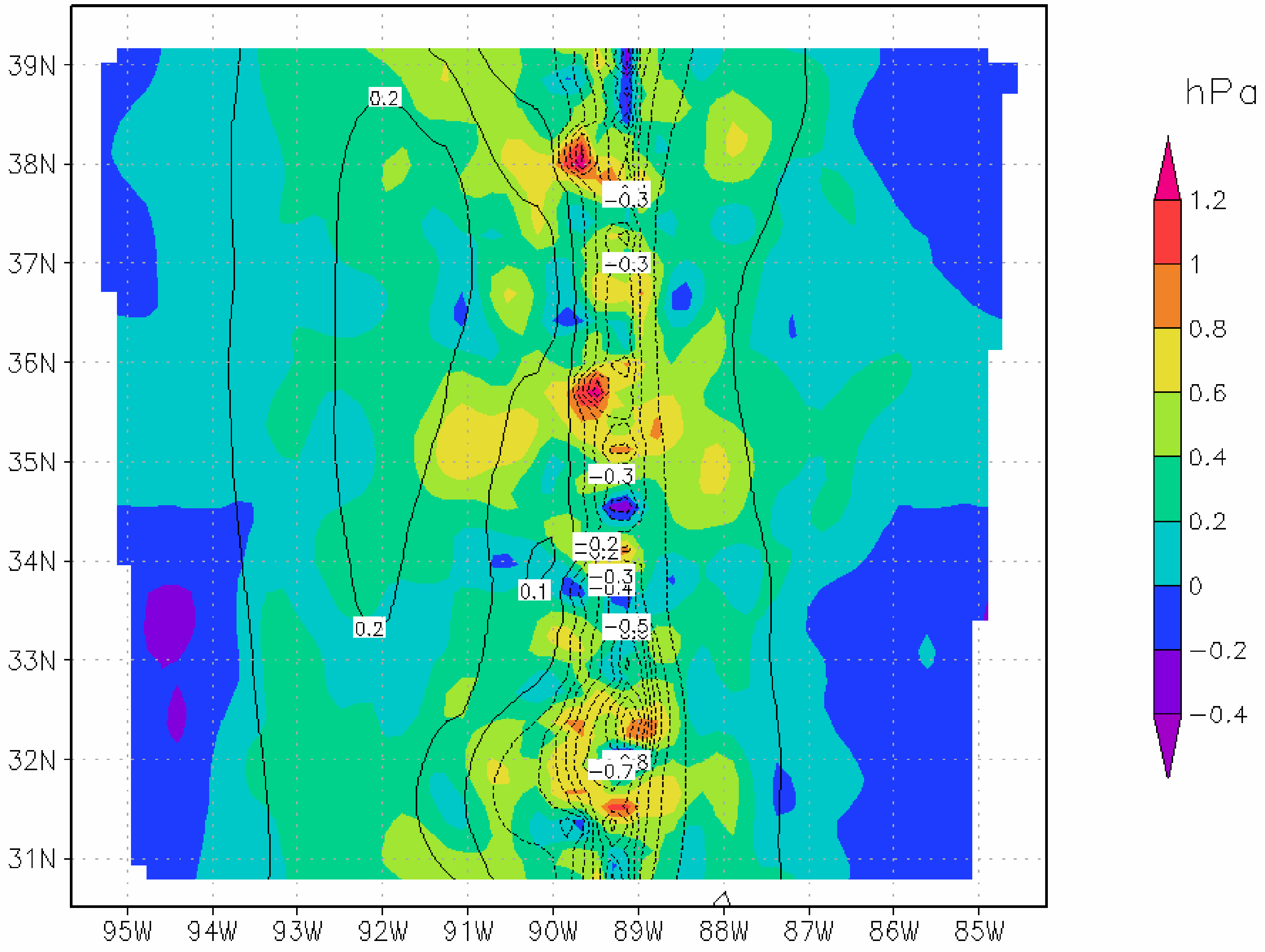


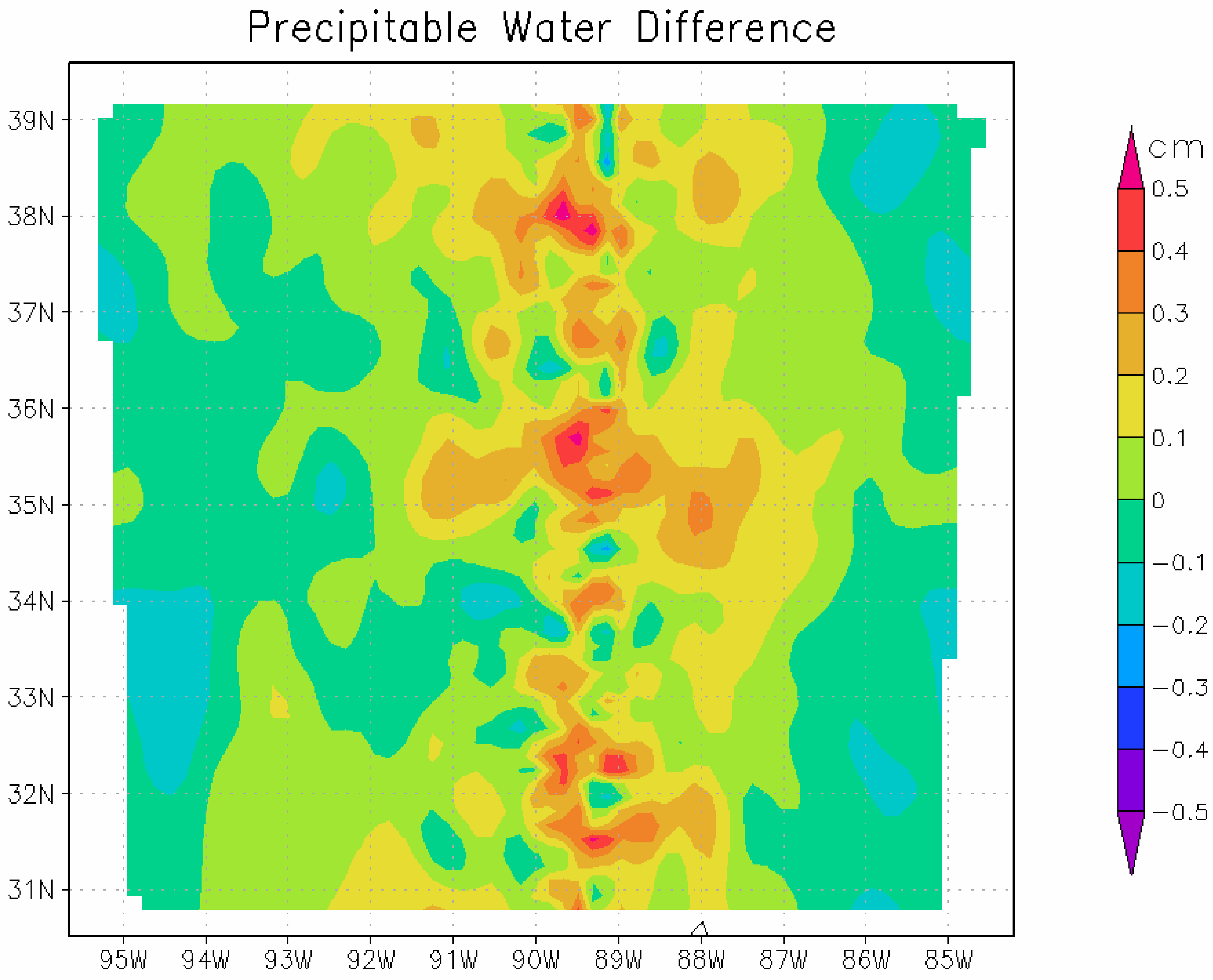


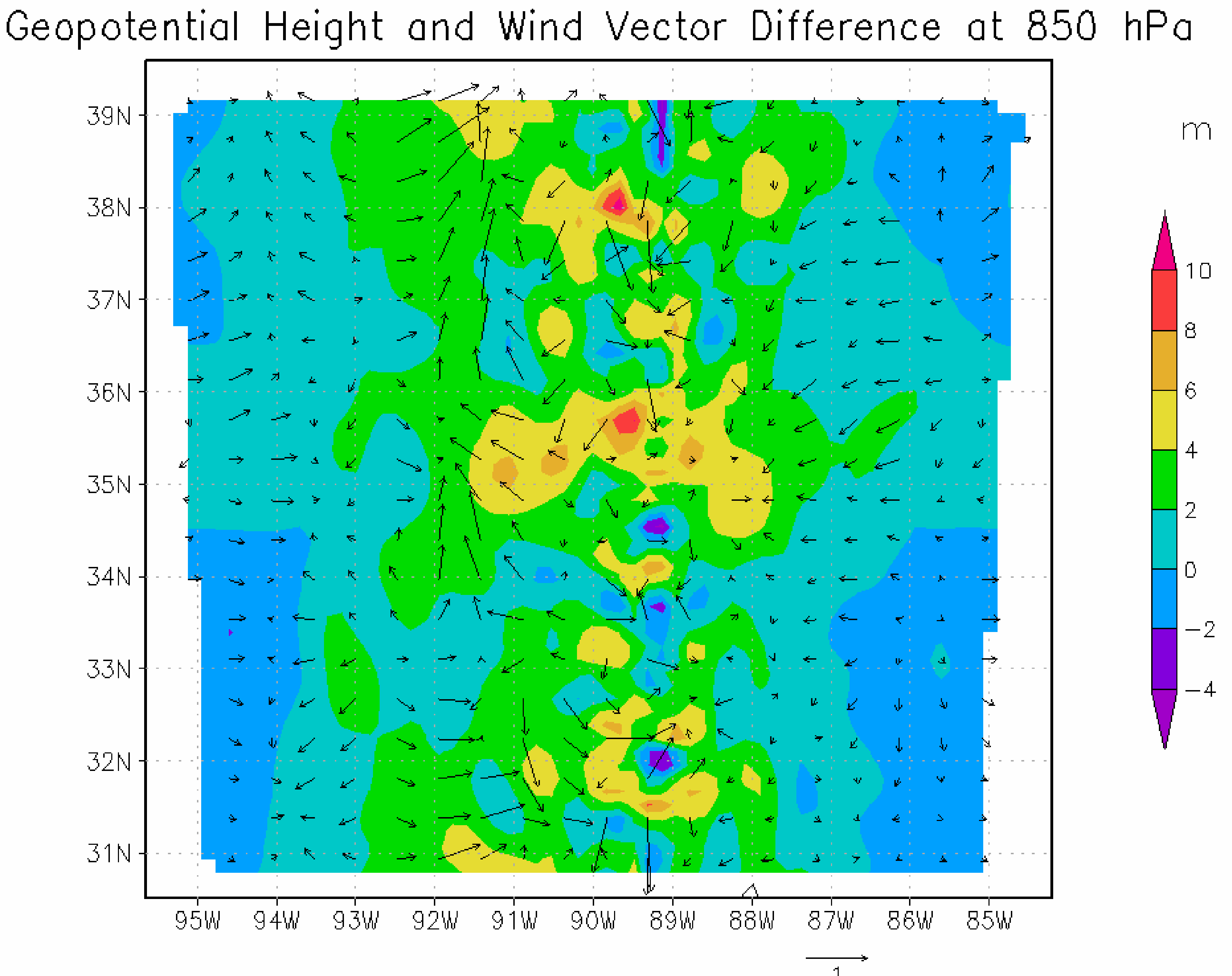
Convective Precipitation – LUT
39N
38N
37N
36N
35N
34N
33N
32N
31N
95W
94W
93W
92W
91W
90W
89W
88W
87W
86W
85W
mm
80
70
60
50
40
30
20
10


Convective Precipitation Difference
39N
38N
37N
36N
35N
34N
33N
32N
31N
95W
94W
93W
92W
91W
90W
89W
88W
87W
86W
85W
mm
40
30
20
10
0
−10
−20
−30
−40

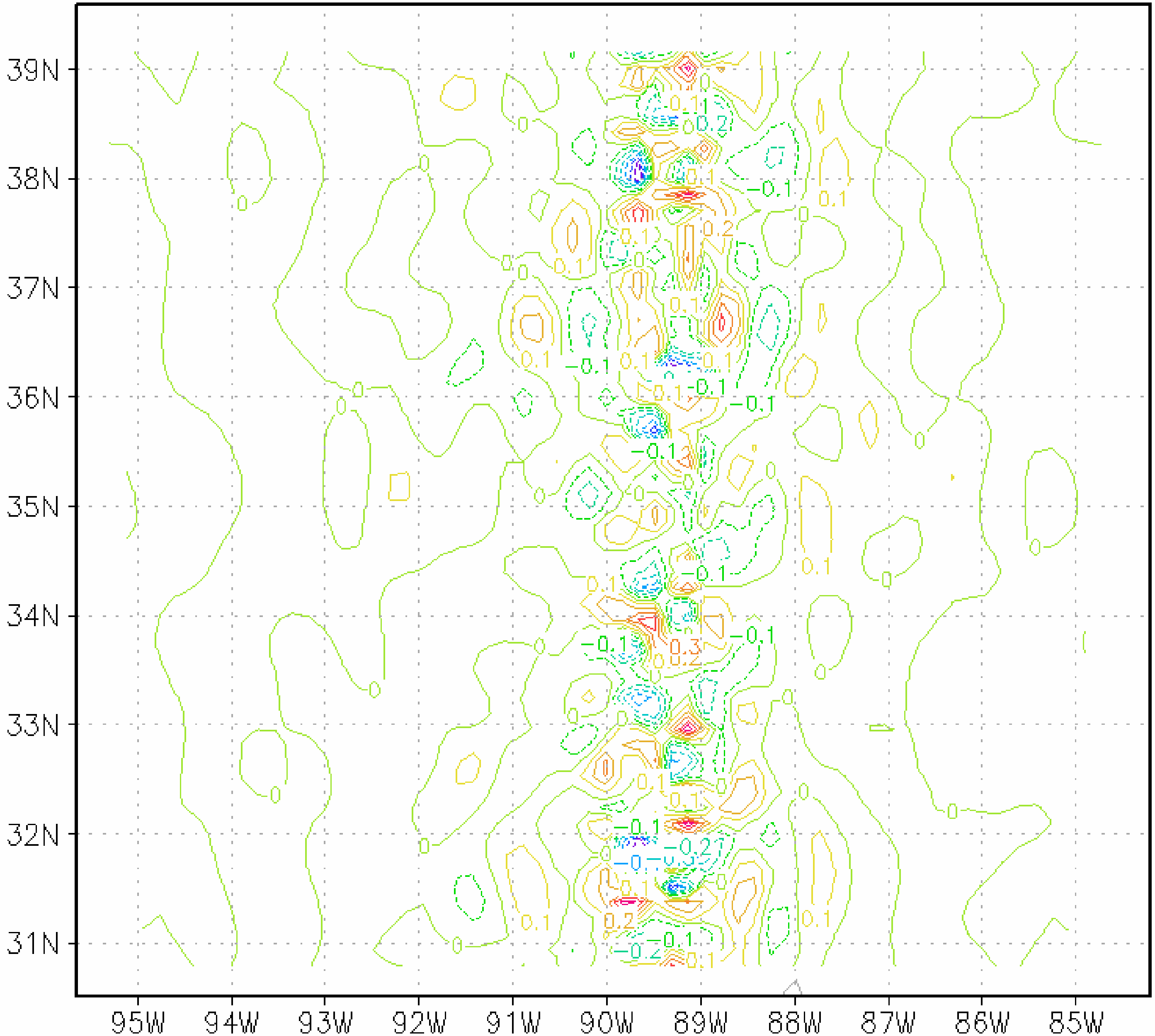
Sea Surface Pressure and Lowest Level Temperature Difference

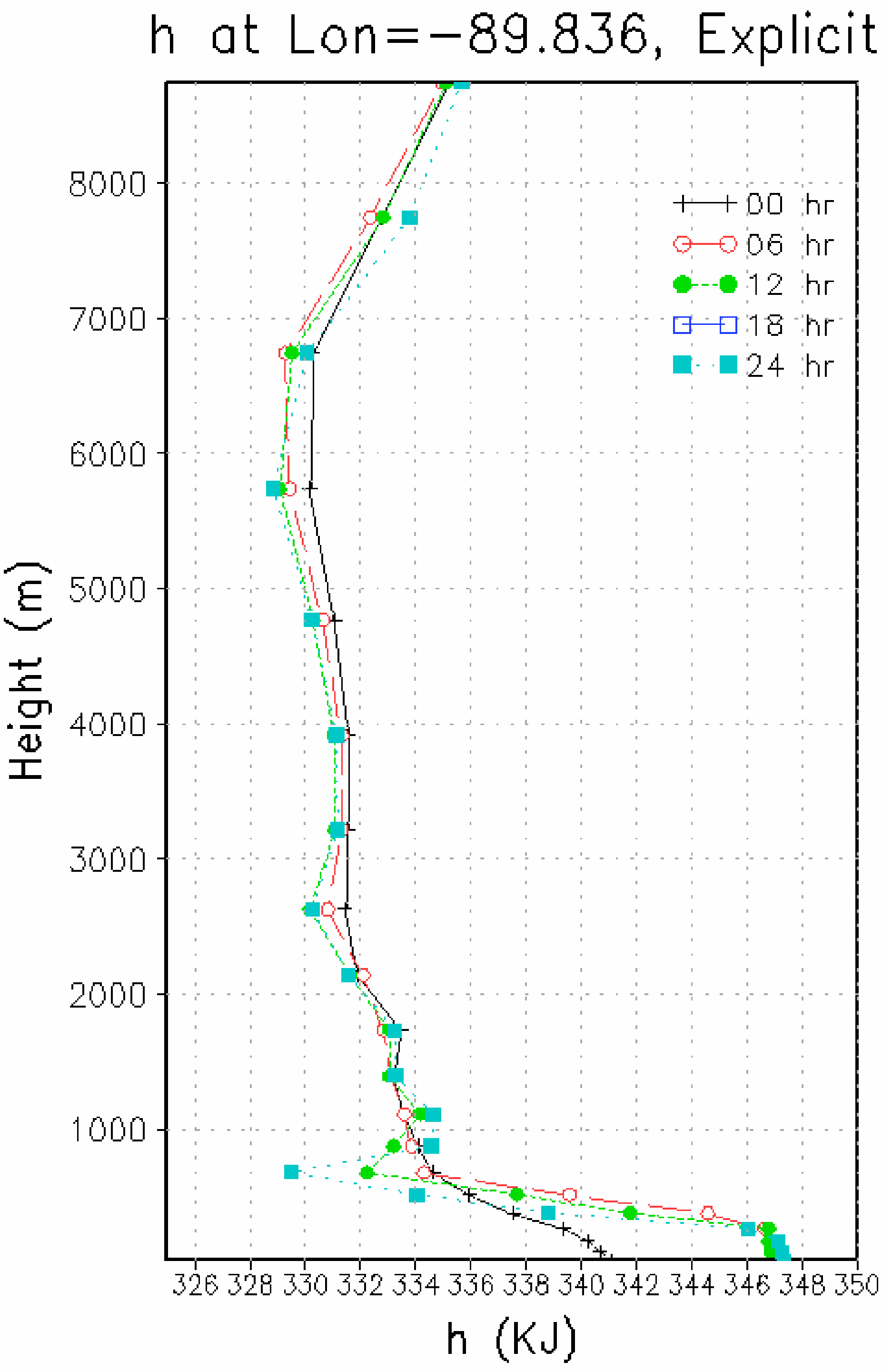



Precipitable Water Difference
39N
38N
37N
36N
35N
34N
33N
32N
31N
95W
94W
93W
92W
91W
90W
89W
88W
87W
86W
85W
cm
0.5
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5

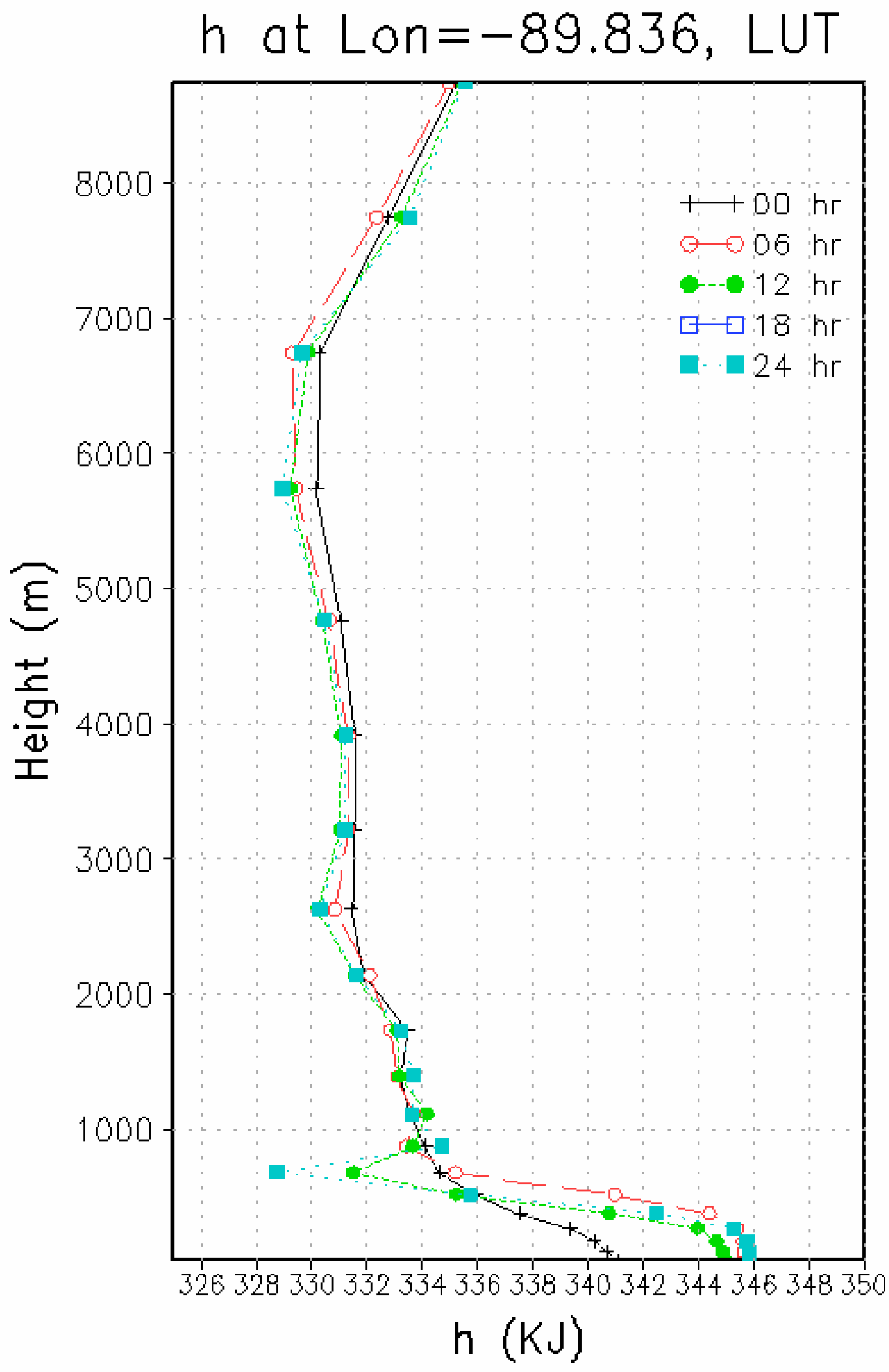

Geopotential Height and Wind Vector Difference at 850 hPa
39N
38N
37N
36N
35N
34N
33N
32N
31N
95W
94W
93W
92W
91W
90W
89W
88W
87W
86W
85W
1
m
10
8
6
4
2
0
−2
−4

Vertical Velocity Difference at 500 hPa

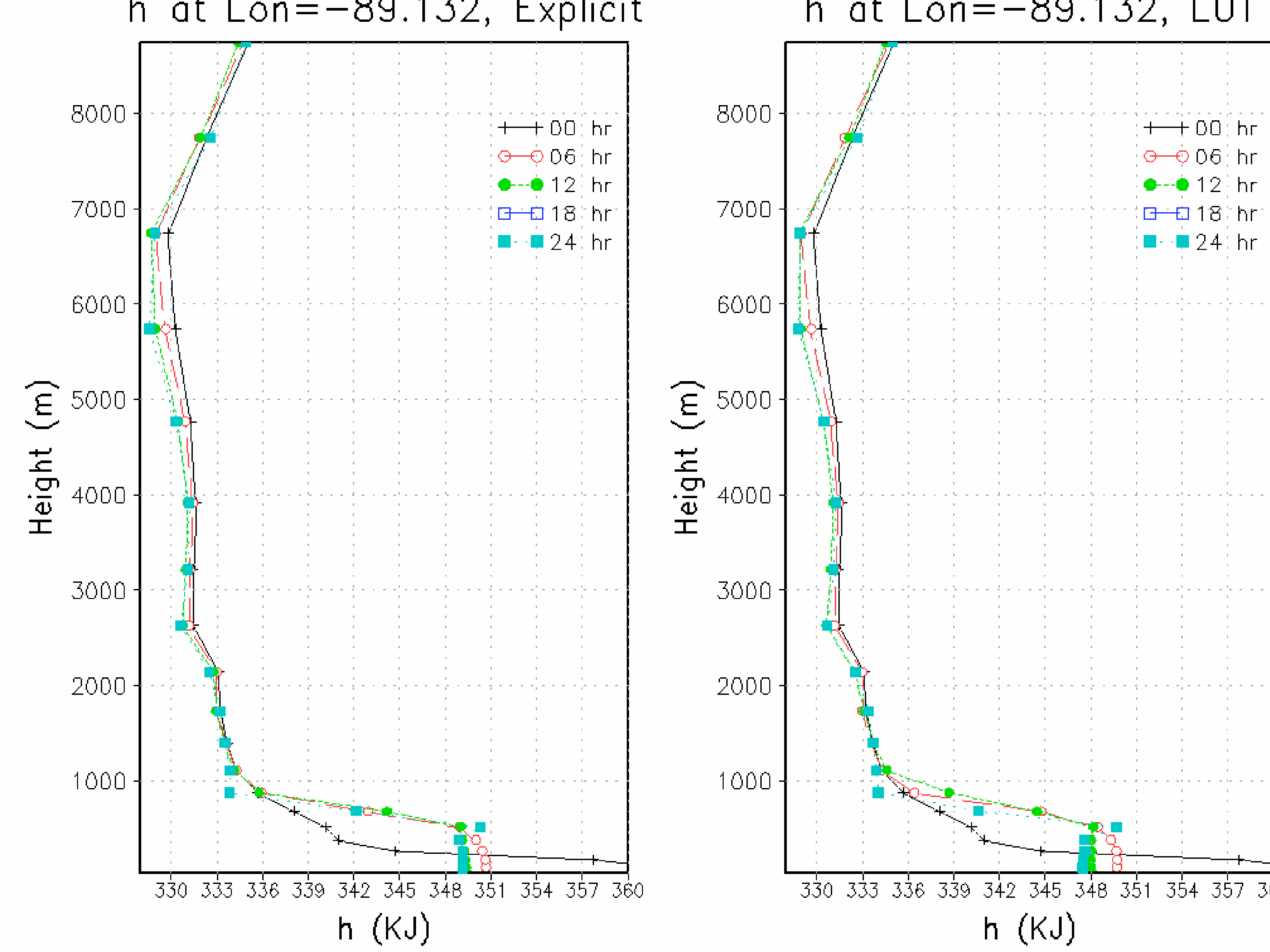




h at Lon=-89.836, Explicit
Height (m)
h (KJ)
00 hr
06 hr
12 hr
18 hr
24 hr



h at Lon=-89.836, LUT
Height (m)
h (KJ)
00 hr
06 hr
12 hr
18 hr
24 hr



h at Lon=-89.132, Explicit
h at Lon=-89.132, LUT
Height (m)
h (KJ)
00 hr
06 hr
12 hr
18 hr
24 hr
1000
2000
3000
4000
5000
6000
7000
8000
330
333
336
339
342
345
348
351
354
357
360

Efficiency and Resources Usage

- 4mins/100mins faster in LUT than in Explicit for this sensitivity test
- Need ~ 64 Kbytes memory to store the table

Conclusion

- More than a dozen tunable parameters
- Very sensitive to mass flux ratio
- LUT method is a little faster than the original code, but needs ~64 K memory for each node

Reference

- Grell, G. A., J. Dudhia, and D. R. Stauffer, A description of the Fifth-generation Penn State/NCAR Mesoscale Model (MM5), *NCAR Tech Note TN-398 + STR*, 122 pp., 1994.
- G. A. Grell and D. Devenyi, 2002. A generailized approach to parameterizing convection combing ensemble and data assimilation techniques, *Geophysical Research Letters*, 29(14), No. 38.
- Arakawa, A., 2004. Review Article: The cumulus parameterization problem: Past, Present, and Future, *Journal of Climate*, 17(13), 2493-2525.
- Grell, G. A., 1993. Prognostic evaluation of assumptions used by cumulus parameterizations, *Monthly Weather Review*, 121, 764-787.