

Fundamentals in Atmospheric Modeling

Song-You Hong

(NCAR affiliate scientist, songyouhong@gmail.com)

(Former Director, Korea Institute of Atmospheric Prediction Systems: KIAPS)

(Former Professor, YonSei University: YSU)



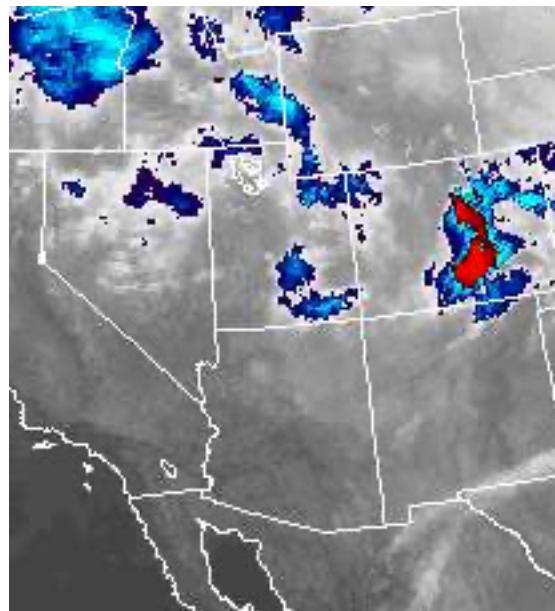
List of presentations

- Concept of modeling
- Structure of models
- Predictability
- Regional modeling



How were the today's forecasts made ?

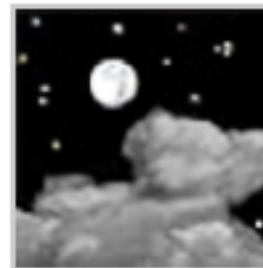
Observation



Forecasts

Boulder CO 7 Day Forecast

OVERNIGHT



Partly
Cloudy
Low: 29 °F

TUESDAY



Mostly
Sunny
High: 50 °F

TUESDAY
NIGHT



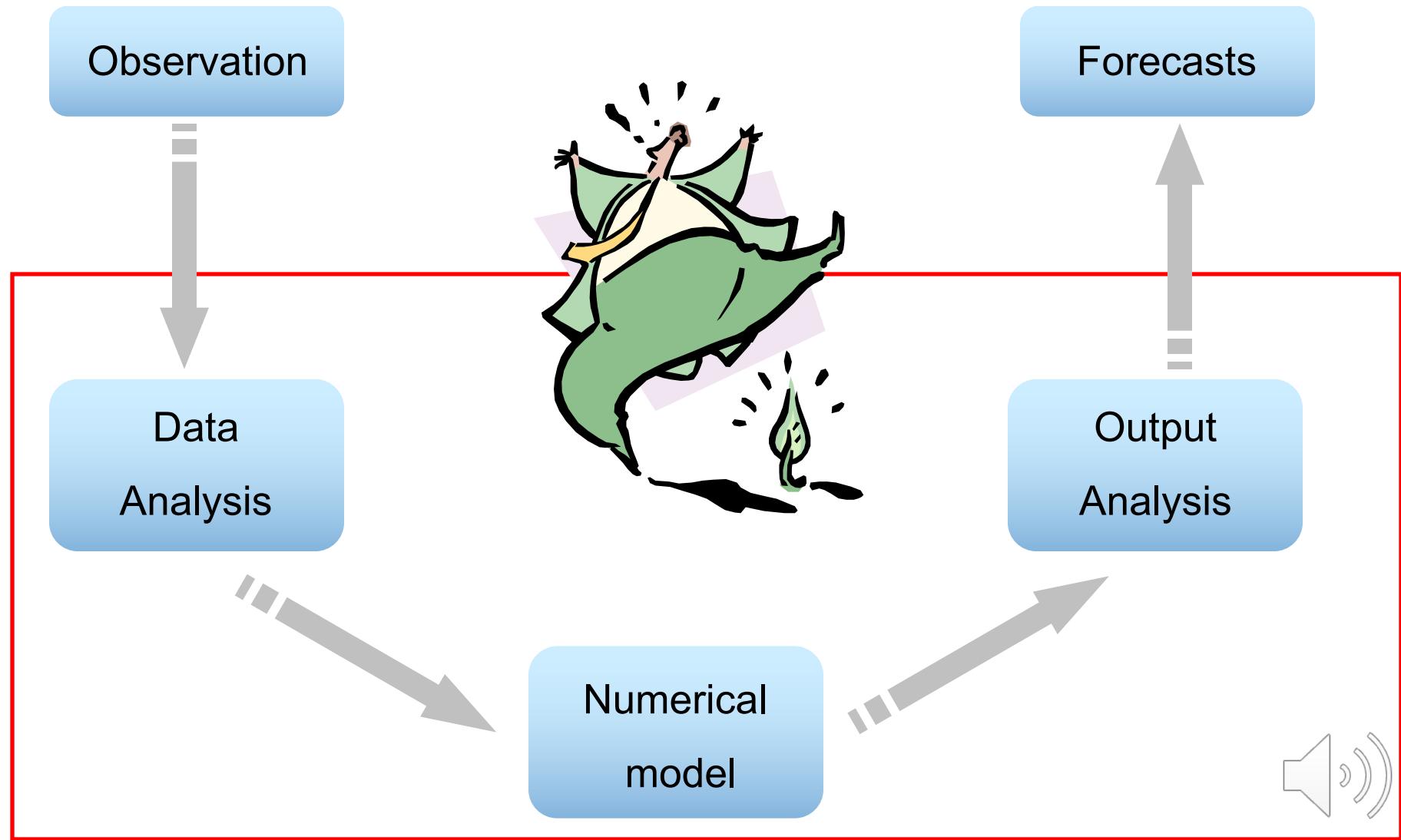
Chance
Snow
Low: 19 °F



Then, what ?



Numerical weather prediction is a crucial component

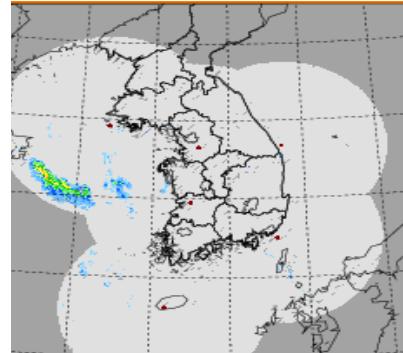
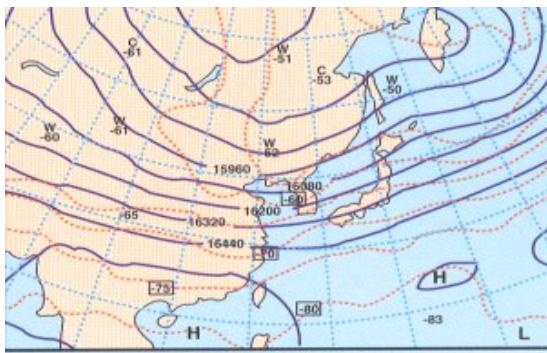


Then, how ?

Step1:
Observation



Step2:
Data analysis



Also construct 3-D data of wind, temperature, etc.



Theory of numerical weather prediction (NWP)

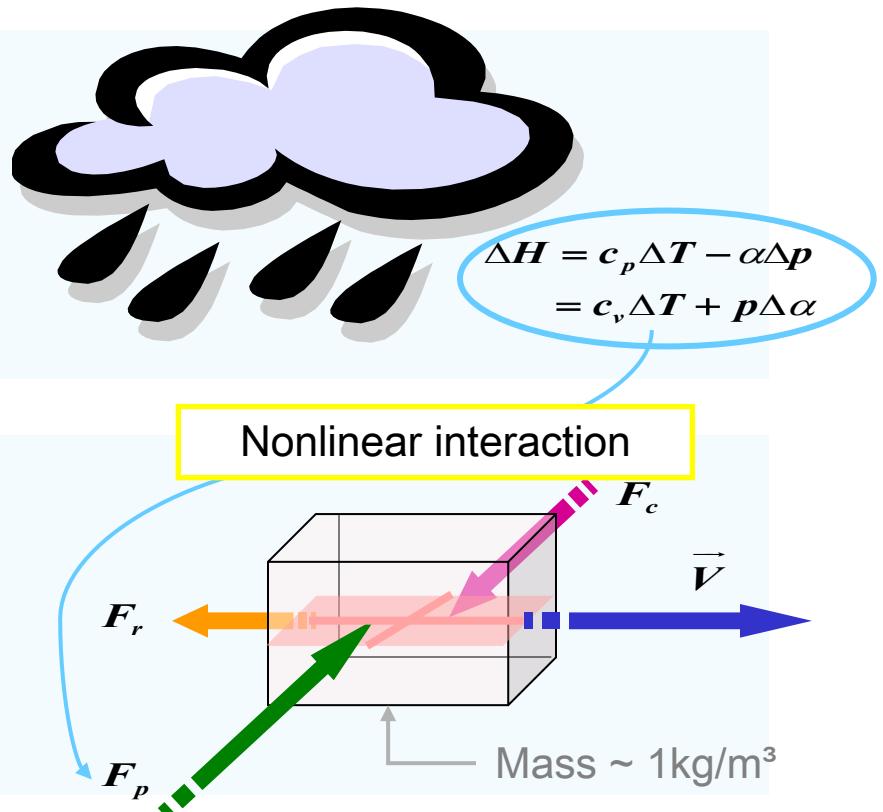
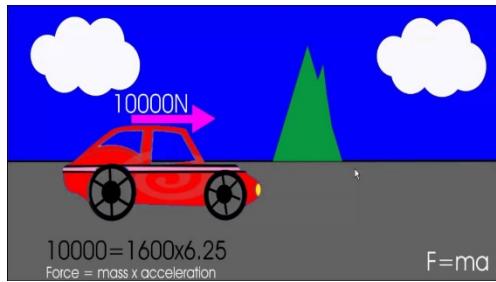
Thermodynamics

$$\text{Heat} = \text{Energy} + \text{Work}$$

Dynamics

$$\text{Force} = \text{Mass} \times \text{Acceleration}$$

- Mass $\approx 1 \text{ kg/m}^3$
- Force: PGF, Coriolis, Friction...



3 kg

$F = 12 \text{ N}$

4 m/s/s

$\Rightarrow v = v_0 + at$

Theory of NWP : Atmosphere is a **conserved** property

- Momentum

$$F = ma$$

Force = mass x acceleration

- Mass

$$\frac{1}{M} \frac{dM}{dt} = 0$$

Mass of a fluid is conserved

- Moisture

$$\frac{dq}{dt} = E - C$$

Moisture change
= evaporation - condensation

- Energy

$$Q = C_v \frac{dT}{dt} + p \frac{d\alpha}{dt}$$

Heat
= internal energy change – work done

- Ideal gas

$$p\alpha = RT$$

Pressure x specific volume
= gas constant x temperature



The governing equations : mathematical expression

V. Bjerknes (1904) pointed out for the first time that there is a complete set of
7 equations with 7 unknowns that governs the evolution of the atmosphere:

$$\frac{d\mathbf{v}}{dt} = -\alpha \nabla p - \nabla \phi + \mathbf{F} - 2\Omega \times \mathbf{v} \quad (1-3), \quad \text{East-west, North-south, and vertical}$$

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot (\rho \mathbf{v}) \quad (4)$$

$$p = \rho R T \quad (5)$$

$$\frac{ds}{dt} = C_p \frac{1}{\theta} \frac{d\theta}{dt} = \frac{Q}{T} \quad (6)$$

$$\frac{dq}{dt} = E - C \quad (7)$$

7 equations, 7 unknown ($u, v, w, T, p, \text{den}$ and q)

solvable



History of numerical weather prediction

1904 : Norwegian V. Bjerknes (1862-1951) :

Setup the governing equations

1922 : British L. F. Richardson (1881-1953) :

Integrate model → failed

1939 : Swedish C.-G. Rossby :

1948, 1949, J. G. Charney (1917-1981)

1950 : Princeton Group

(Charney, Fjortoft,
von Newman)

ENIAC

(Electrical Numerical
Integrator and Computer)
→ first success

Computer Age (1946~)

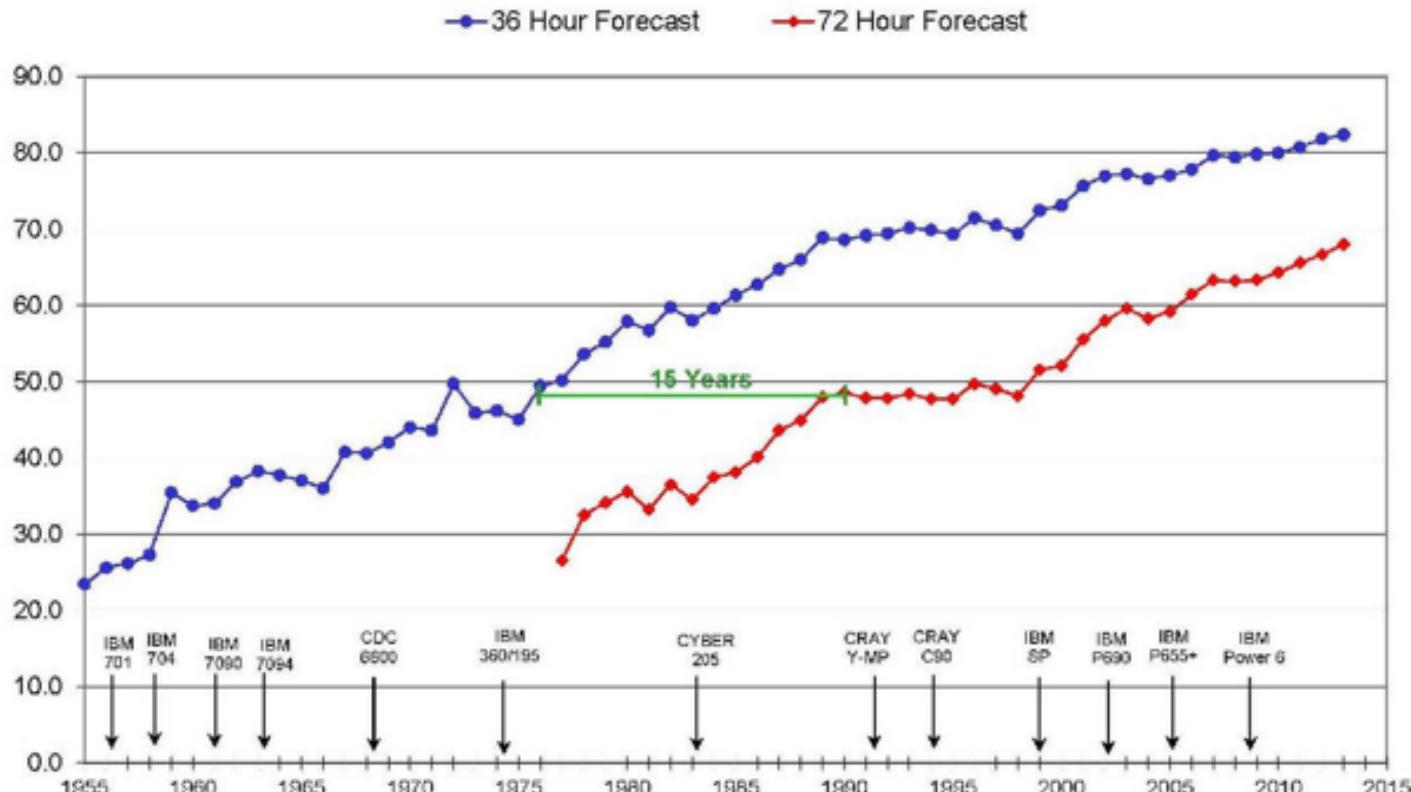
- von Neumann and Charney
 - Applied ENIAC to weather prediction
- Carl-Gustaf Rossby
 - The Swedish Institute of Meteorology
 - First routine real-time numerical weather forecasting. (1954)
 - (US in 1958, Japan in 1959)



History of NWP skill : NCEP GFS (global forecast system)



NCEP Operational Forecast Skill
36 and 72 Hour Forecasts @ 500 MB over North America
[$100 * (1-S1/70)$ Method]



1day / 10year

1990: 48 hr fcst skill = 2020: 120 hr fcst skill



Factors for the improvement (Kalnay 2002)

- Supercomputers
- Physical processes
- Initial conditions



Super-computer for weather models

ENIAC, 1946 (500 FLOPS)



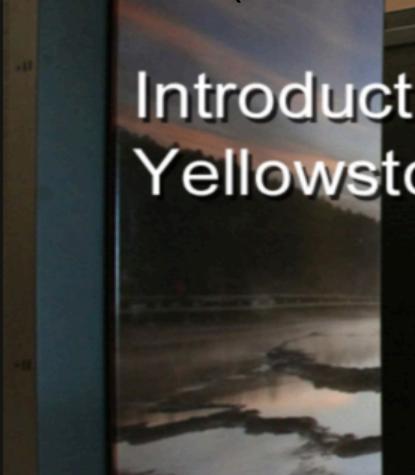
Sunway (125 Peta=10**15FLOPS)



XC40 (2.9 Peta)



IBM (1.5 Peta)



K-computer (11.2 Peta)



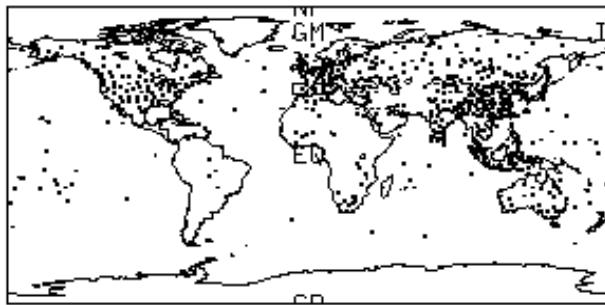
Initial condition (data assimilation)



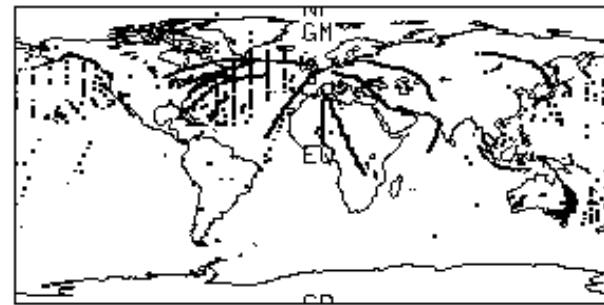
Various observations

DATA DISTRIBUTION 01SEP9700Z-01SEP9700Z

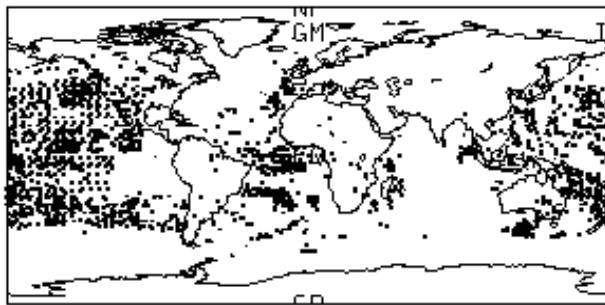
RAOBS



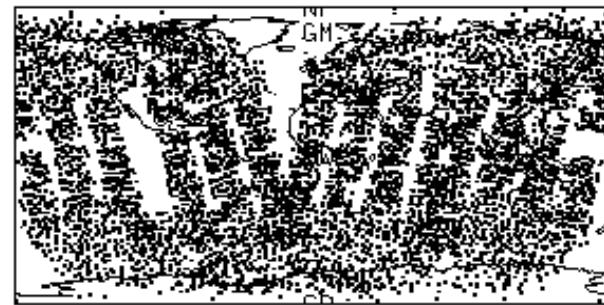
AIRCRAFT



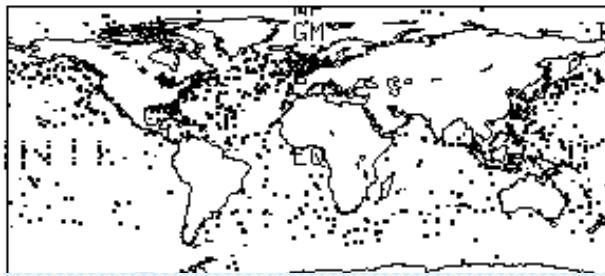
SAT WIND



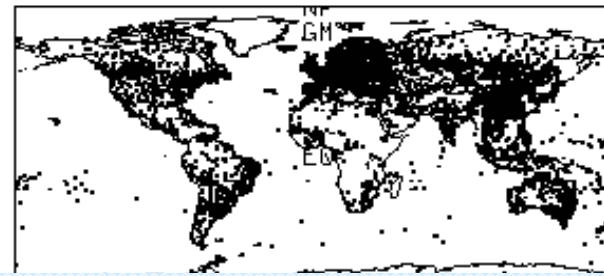
SAT TEMP



SFC SHIP



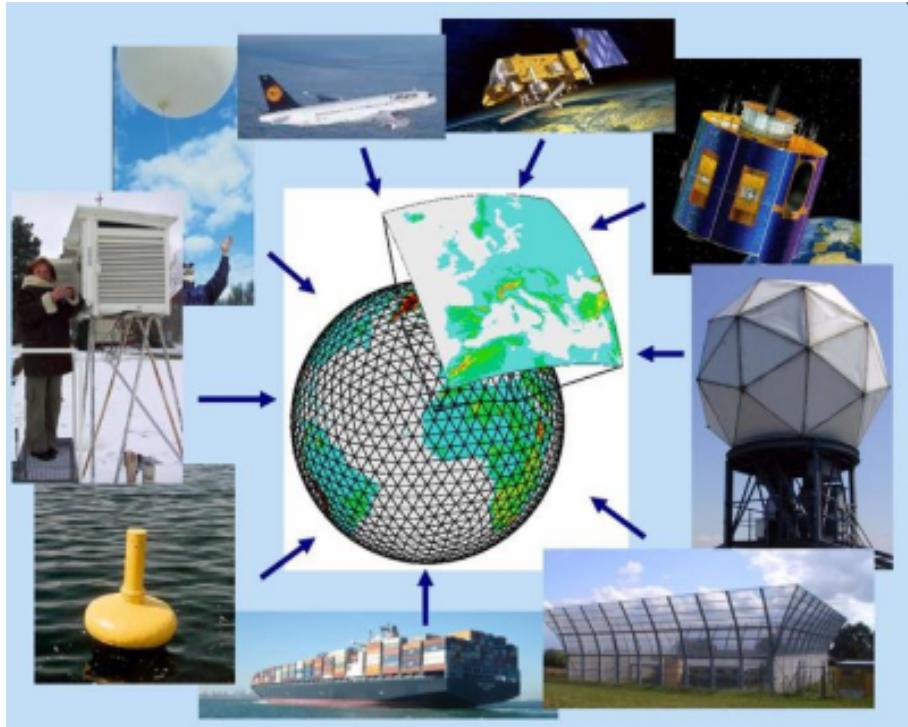
SFC LAND



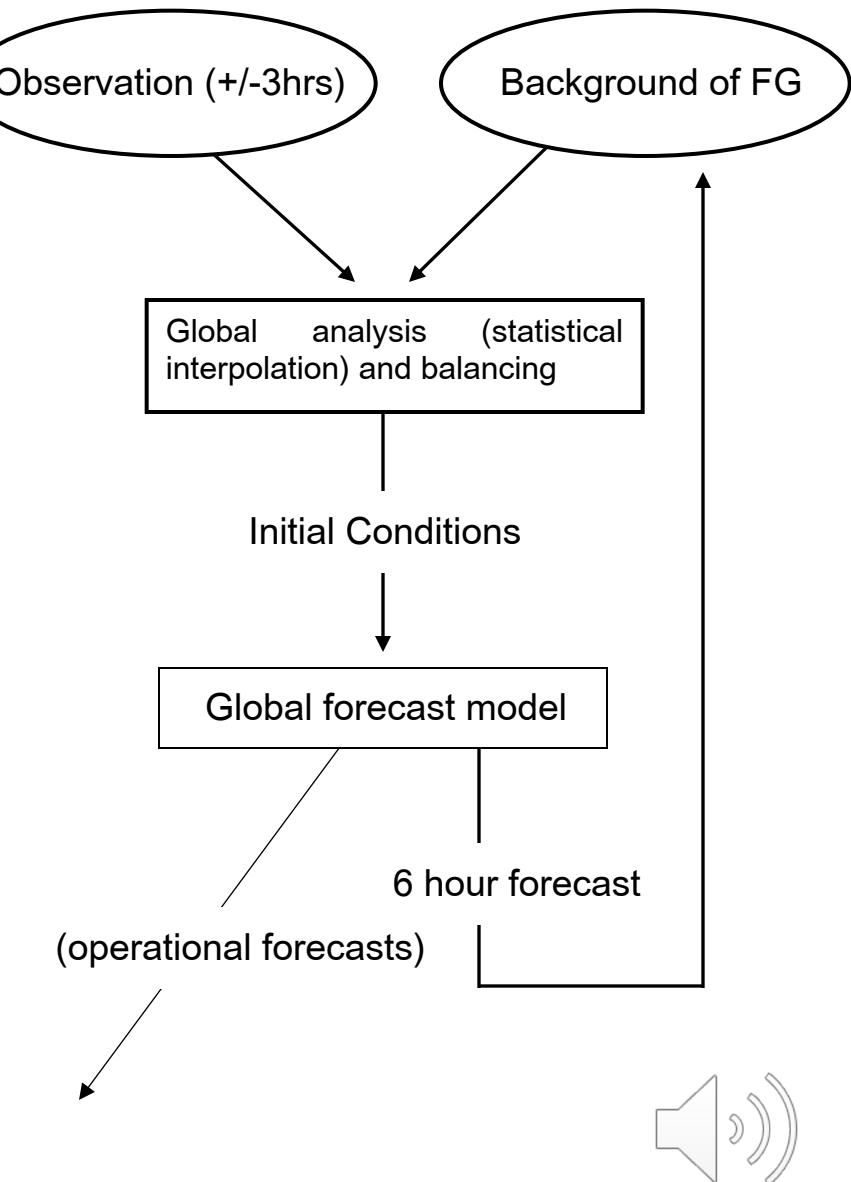
Heterogeneous in space and time....but the initial data should be on a model grid at a specific time!



Data Assimilation



Data assimilation best combines observations and a model



Model

- Dynamics : Identity (Speed)
- Physics : Components (Predictability)



Step3: Integration

$$\begin{aligned} u_t + uu_x + vu_y + wu_z &= -\frac{1}{\rho} p_x + \left(f + \frac{u}{a \tan \phi} \right) v + F_x \\ v_t + uv_x + vv_y + wv_z &= -\frac{1}{\rho} p_y - \left(f + \frac{u}{\tan \phi} \right) u + F_y \\ w_t + uw_x + vw_y + ww_z &= -\frac{1}{\rho} p_z - g + F_z \\ \rho_t + u\rho_x + v\rho_y + w\rho_z &= -\rho(u_x + v_y + w_z) \end{aligned}$$

$$T_t + uT_x + vT_y + wT_z - \frac{1}{\rho C_p} (p_t + up_x + vp_y + wp_z) = \frac{1}{C_p} Q$$

$$q_t + uq_x + vq_y + wq_z = M$$

PHYSICS

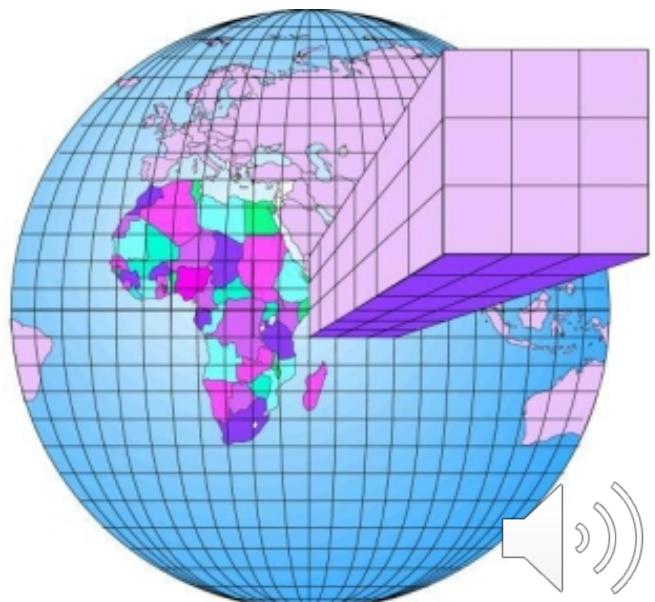
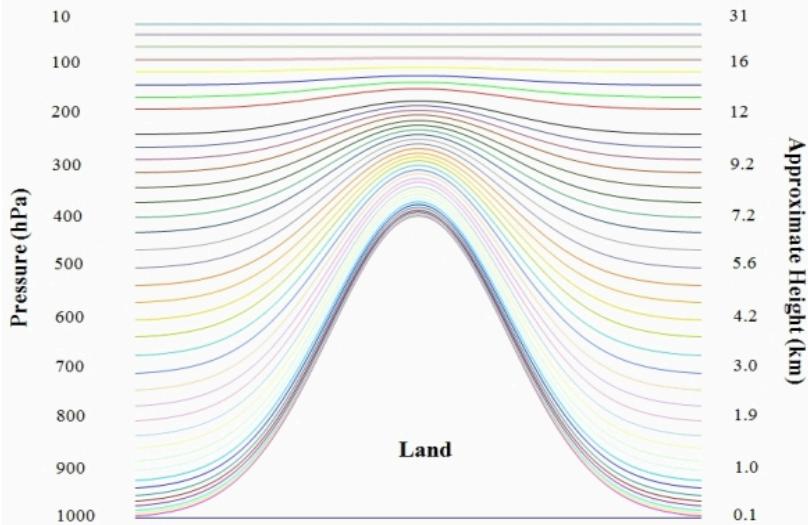
$$p = \rho RT$$

unknown : $[u, v, w, \rho, T(\theta), q, p]$

If we consider O_3 ,

$$C_t + uC_x + vC_y + wC_z = O_3$$

Dynamics : Grid system



Dynamics : Numerical method (spatial)

Ex) $\frac{\partial \phi}{\partial t} = -c \frac{\partial \phi}{\partial x}$; advection eq.

Finite difference method (FD) :

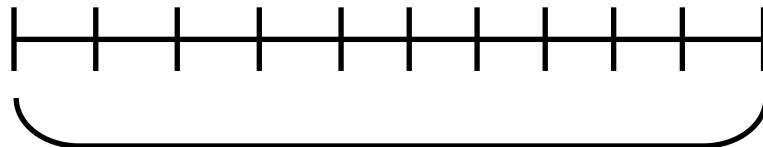
Spectral method (SP) :

Finite element method (FE) :

1) Finite difference (FD) method

- Specify the value at physical grids

$$\frac{\Delta \phi}{\Delta t} = \frac{\phi_2 - \phi_1}{t_2 - t_1}$$

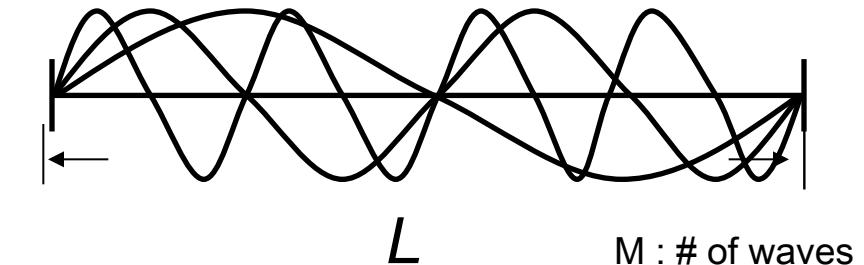


2) Spectral Method

- Determine basis function to get $H(\phi(x))$

$e_m(x)$ (basis funct), $m = m_1 \dots m_n \dots \rightarrow$ infinite

$$\Rightarrow \phi(x, t) = \sum_{m=m_1}^M \phi_m(t) e_m(x)$$



* Resolution Increases

$\left\{ \begin{array}{l} \Delta x \rightarrow \text{decreases} \\ (\Delta x = L / M) \\ M \rightarrow \text{increases} \end{array} \right.$



Dynamics : Numerical method (temporal)

a) $\frac{u^{n+1} - u^{n-1}}{2\Delta t} = F(u^n)$: leap-frog good for hyperbolic
 unstable for parabolic

b) $\frac{u^{n+1} - u^n}{\Delta t} = F(u^n)$: Euler-forward good for diffusion
 unstable for hyperbolic

c) $\frac{u^{n+1} - u^n}{\Delta t} = F\left(\frac{u^n + u^{n+1}}{2}\right)$: Crank-Nicholson

d) $\frac{u^{n+1} - u^n}{\Delta t} = F(u^{n+1})$: Fully implicit, backward

e) $\frac{u^* - u^n}{\Delta t} = F(u^n)$: $\frac{u^{n+1} - u^n}{\Delta t} = F(u^*)$: Euler-backward (Matzuno)

f) $\frac{u^{\frac{n+1}{2}*} - u^n}{\Delta t/2} = F(u^n)$: $\frac{u^{n+1*} - u^n}{\Delta t} = F\left(u^{\frac{n+1}{2}*}\right)$

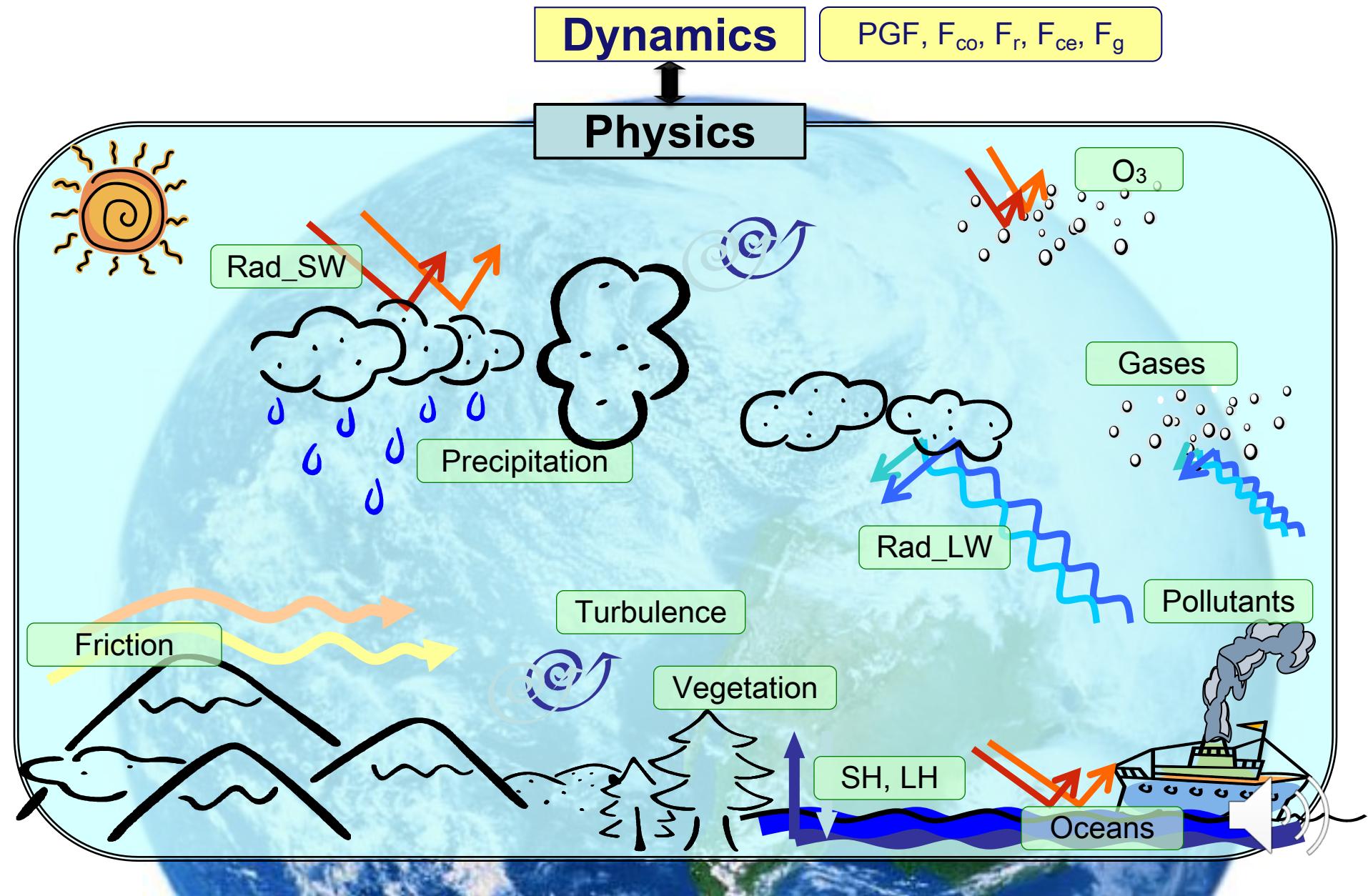
$$\frac{u^{n+1} - u^n}{\Delta t} = \frac{1}{6} \left[F(u^n) + 4F\left(u^{\frac{n+1}{2}*}\right) + F(u^{n+1*}) \right]$$
 : RK(Runge-Kutta)-3rd order

g) $\frac{u^{n+1} - u^{n-1}}{2\Delta t} = F_1(u^n) + F_2\left(\frac{u^{n+1} - u^{n-1}}{2}\right)$: Semi-Implicit

h) $\frac{u^* - u^n}{\Delta t} = F_1(u^n); \quad \frac{u^{n+1} - u^*}{\Delta t} = F_2(u^*)$: Fractional steps



Physics modules : Branches of atmospheric sciences

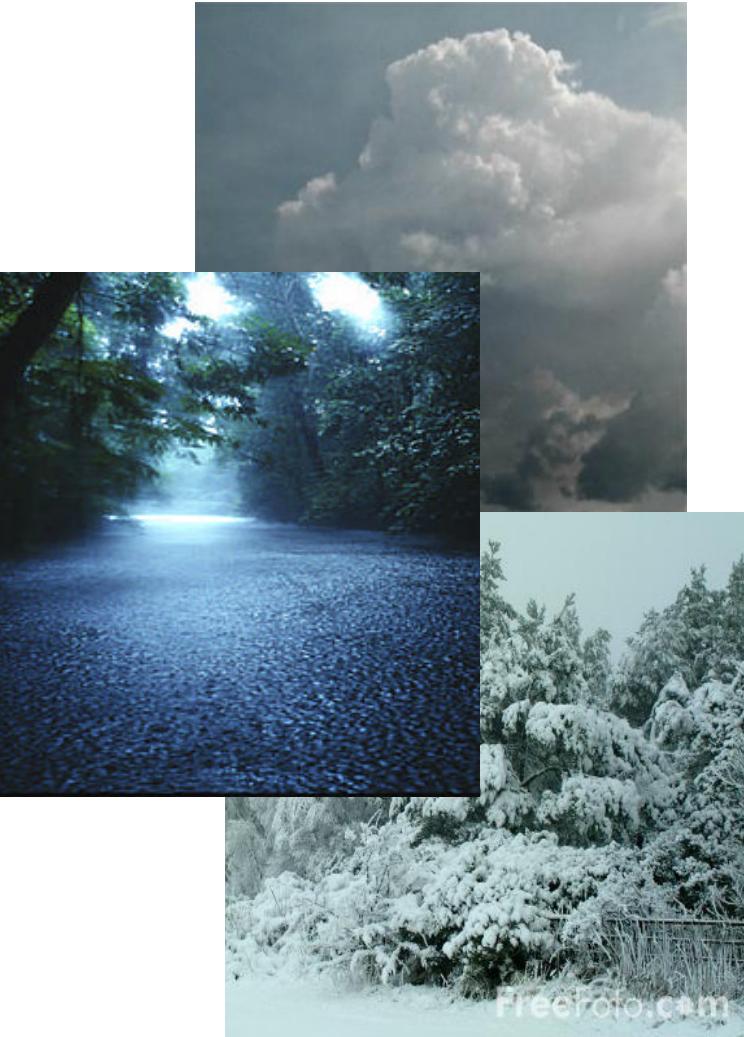


Physics module (example): Cloud and precipitation

Real atmosphere

Theory
Formulation

Model (computer program)



A screenshot of a terminal window titled "login1" showing a nested script structure. The outermost script is "module_mp_wsm3". Inside it, there is a function definition for "cmcal(x)" which calculates a value based on x and qmin. Below that, a section for "compute internal functions" contains several sub-functions and their implementations. The innermost function is "compute the minor time steps". This section includes code to calculate loops, dtclcd, and dtclcd, and a loop from 1 to loops. It also initializes large-scale variables like tvec1, den, and vssqrt. Further down, it defines cvap, hvap, hsub, ttp, dldt, xa, xb, and xai, and performs calculations involving them. Finally, it defines a loop for kts to kte, calculating qs, psat, and qs0 values, and updating qs and qs0. A speaker icon is located in the bottom right corner of the terminal window.

```
MODULE module_mp_wsm3
!
!-----+
! compute internal functions
!
  cmcal(x) = cpd*(1.-max(x,qmin))+max(x,qmin)*cpv
!
!-----+
! compute the minor time steps.
!
  loops = max(nint(delt/dtclcdcr),1)
  dtclcd = delt/loops
  if(delt.le.dtclcdcr) dtclcd = delt
  do loop = 1,loops
!
!-----+
! initialize the large scale variables
!
  do i = its,ite
    mstep(i) = 1
    flgcldi(i) = .true.
  enddo
!
  do k = kts,kte
    CALL vsrec(tvec1(its),den(its,k),ite -its+1)
    do i = its,ite
      tvec1(i) = tvec1(i)*den0
    enddo
    CALL vssqrt(denfac(its,k),tvec1(its),ite -its+1)
  enddo
!
  cvap = cpv
  hvap=xlv0
  hsub=xls
  ttp=t0c+0.01
  dldt=cvap-cliq
  xa=dldt/rv
  xb=xa+hvap/(rv*ttp)
  dldti=cvap-cice
  xai=dldti/rv
  xbi=xai+hsub/(rv*ttp)
  do k = kts,kte
    do i = its,ite
      tr=ttpl/t(i,k)
      if(t(i,k).lt.ttp) then
        qs(i,k) = psat*(exp(log(tr)*(xai)))*exp(xbi*(1.-tr))
      else
        qs(i,k) = psat*(exp(log(tr)*(xa)))*exp(xbi*(1.-tr))
      endif
      qs0(i,k) = psat*(exp(log(tr)*(xa)))*exp(xb*(1.-tr))
      qs0(i,k) = (qs0(i,k)-qs(i,k))/qs(i,k)
      qs(i,k) = ep2 * qs(i,k) / (p(i,k) - qs(i,k))
      qs(i,k) = max(qs(i,k),qmin)
      rh(i,k) = max(q(i,k) / qs(i,k),qmin)
    enddo
  enddo
!
```

Physics module (example): Cloud and precipitation

```

! login1
!
do k = kts, kte
  do i = its, ite
    supsat = max(q(i,k),qmin)-qs(i,k)
    satdt = supsat/dtclcd
    if(t(i,k).ge.t0c) then
      =====
      warm rain processes
      - follows the processes in RH83 and LFO except for autoconversion
      =====
      paut1: auto conversion rate from cloud to rain [HDC 16]
      (C->R)
        if(qci(i,k).gt.qc0) then
          paut(i,k) = qck1*exp(log(qci(i,k))*((7./3.)))
          paut(i,k) = min(paut(i,k),qci(i,k)/dtclcd)
        endif
      =====
      pracw: accretion of cloud water by rain [D89 B15]
      (C->R)
        if(qrs(i,k).gt.qcrmin.and.qci(i,k).gt.qmin) then
          pacri(i,k) = min(pacrr*rslope3(i,k)*rslopeb(i,k)
                            *qci(i,k)*denfac(i,k),qci(i,k)/dtclcd)
        endif
      =====
      pres1: evaporation/condensation rate of rain [HDC 14]
      (V->R or R->V)
        if(qrs(i,k).gt.0.) then
          coeres = rslope2(i,k)*sqrt(rslope(i,k)*rslopeb(i,k))
          pres(i,k) = (rh(i,k)-1.)*(precr1*rslope2(i,k)
                                +precr2*work2(i,k)*coeres)/work1(i,k)
        if(pres(i,k).lt.0.) then
          pres(i,k) = max(pres(i,k),-qrs(i,k)/dtclcd)
          pres(i,k) = max(pres(i,k),satdt/2)
        else
          pres(i,k) = min(pres(i,k),satdt/2)
        endif
      endif
    else
  enddo
enddo
=====

```

T>0°C

Autoconversion

$$P_{aut1} = \min \left(\frac{0.104 g E_C \rho^{\frac{4}{3}}}{\mu (N_c \rho_w)^{\frac{1}{3}}} q_c^{\frac{7}{3}}, \frac{q_c}{dt} \right)$$

Accretion

$$P_{racw} = \frac{\pi a_r E_{CR} N_{0r} q_c}{4} \left(\frac{\rho_0}{\rho} \right)^{\frac{1}{2}} \frac{\Gamma(3+b_r)}{\lambda_r^{3+b_r}}$$

Evaporation

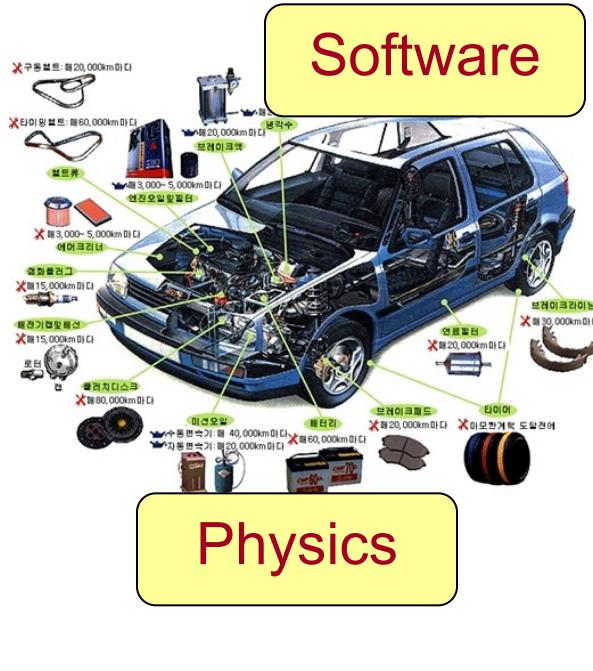
$$Pres1 = \frac{2\pi N_{0r} (S_w - 1)}{(A_w + B_w)} \left[\frac{0.78}{\lambda_r^2} + \frac{a_r^{\frac{1}{2}} 0.31 \Gamma(b_r/2 + 5/2)}{\lambda_r^{b_r/2 + 5/2}} \left(\frac{\mu}{D} \right)^{\frac{1}{3}} \left(\frac{1}{\mu} \right)^{\frac{1}{2}} \left(\frac{\rho_0}{\rho} \right)^{\frac{1}{4}} \right]$$



Car and model



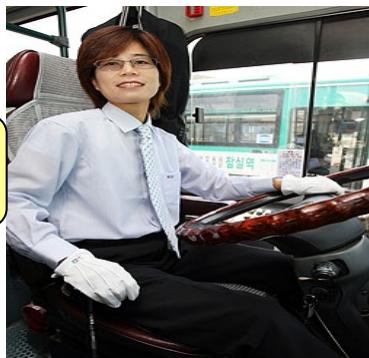
Dynamics



Physics



Data assimilation



Driver



Forecaster



Classification of models

- Dynamic core

Hydrostatic	Non-hydrostatic
Large-scale	Small-scale (heavy rainfall, complex mountain)

- Scale

Global	Regional
10 km – 100 km (NWP – Climate)	1 km - 10 km (NWP-Climate)

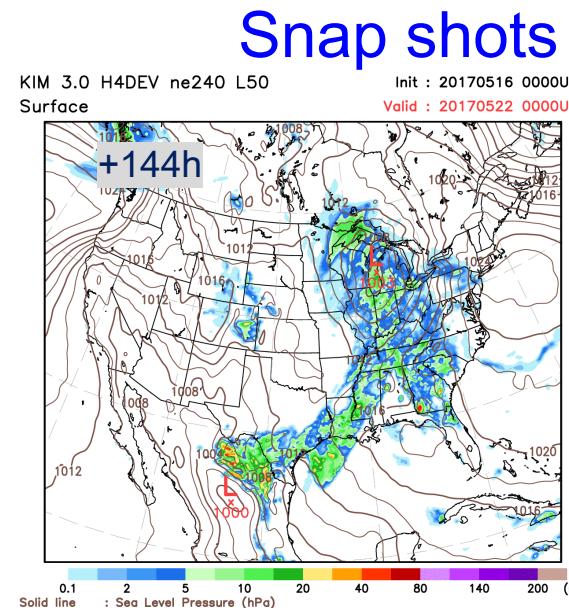
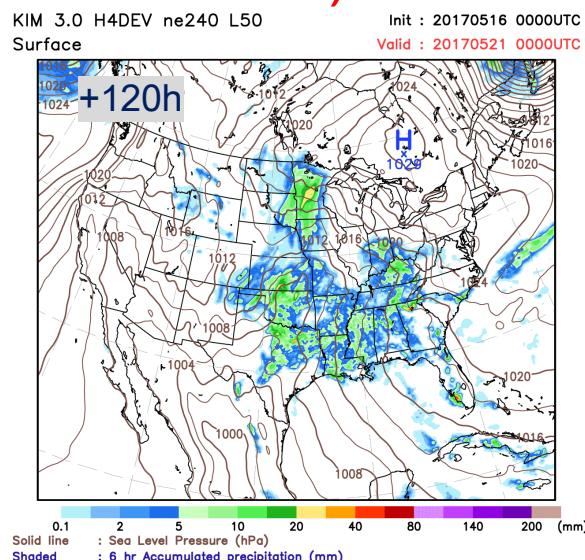
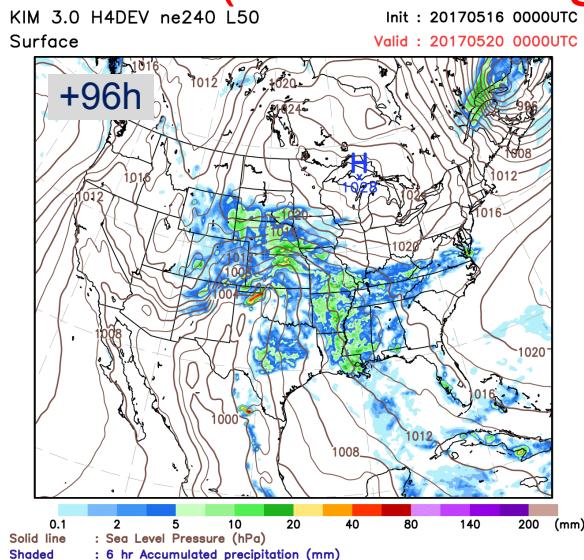
- Purpose

Initial data-> FORECAST	Forcing → RESPONSE
NWP : upto 2 weeks	GCM (General circulation model)

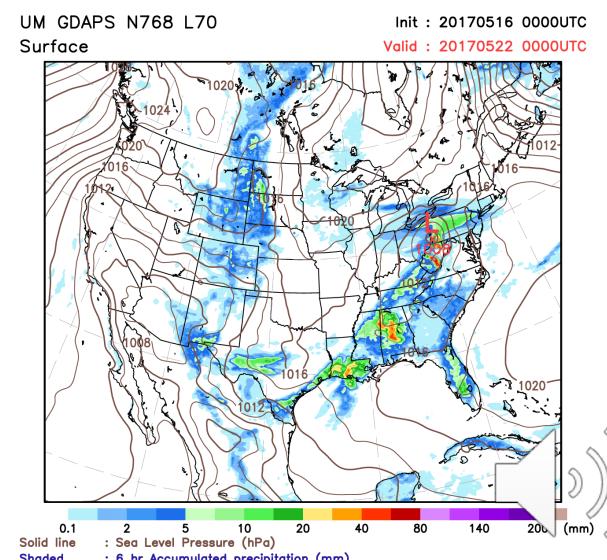
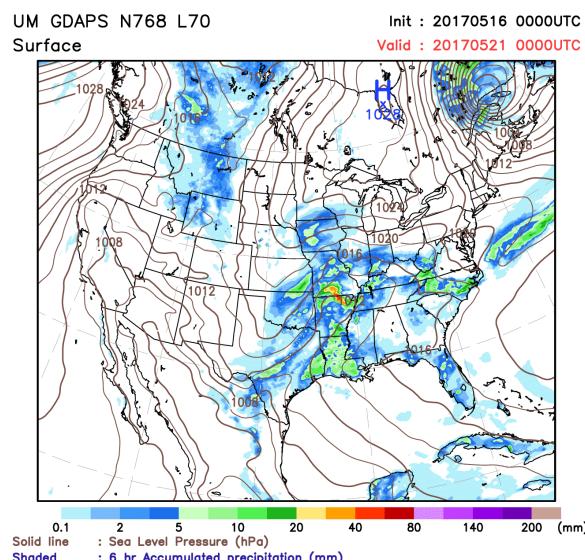
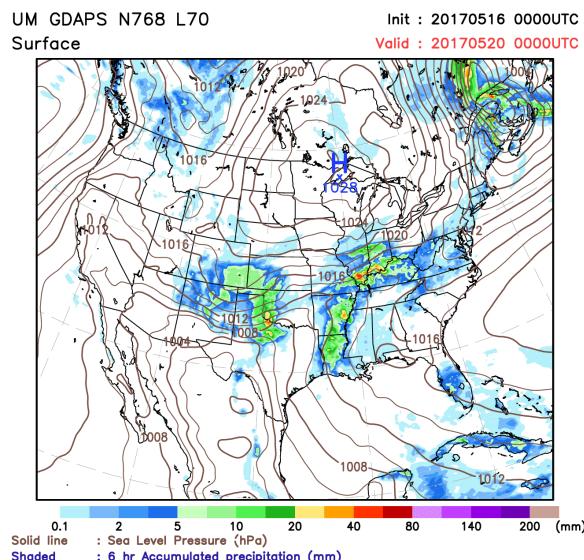


KIAPS real time forecasts (INIT 2017051600) : 12 km global NWP

KIM (Korean Integrated Model)



KMA UM



Predictability



Chaos theory (Lorenz)

Charney (1951) : Uncertainties in initial condition and model



Lorenz (1962,1963) : Unstable nature of atmosphere



Purpose : NWP is better than statistical forecast

Tool : 4 K memory computer

Model : 12 variables (heating and dissipation forcing)

Results : differences -> non-periodicity



Initial condition (3 decimal point) : different after 2 month



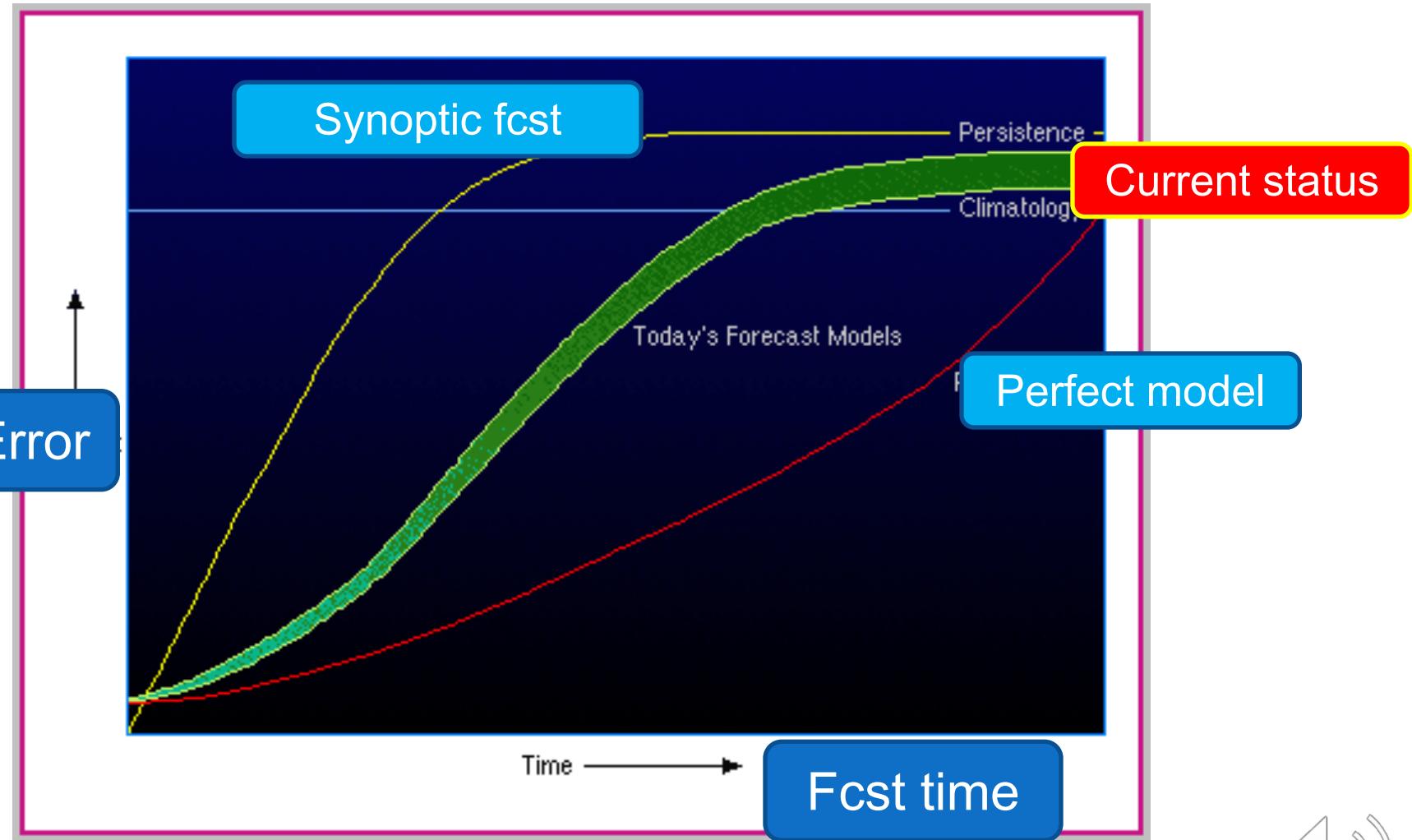
Round-off error -> cause of non-periodicity



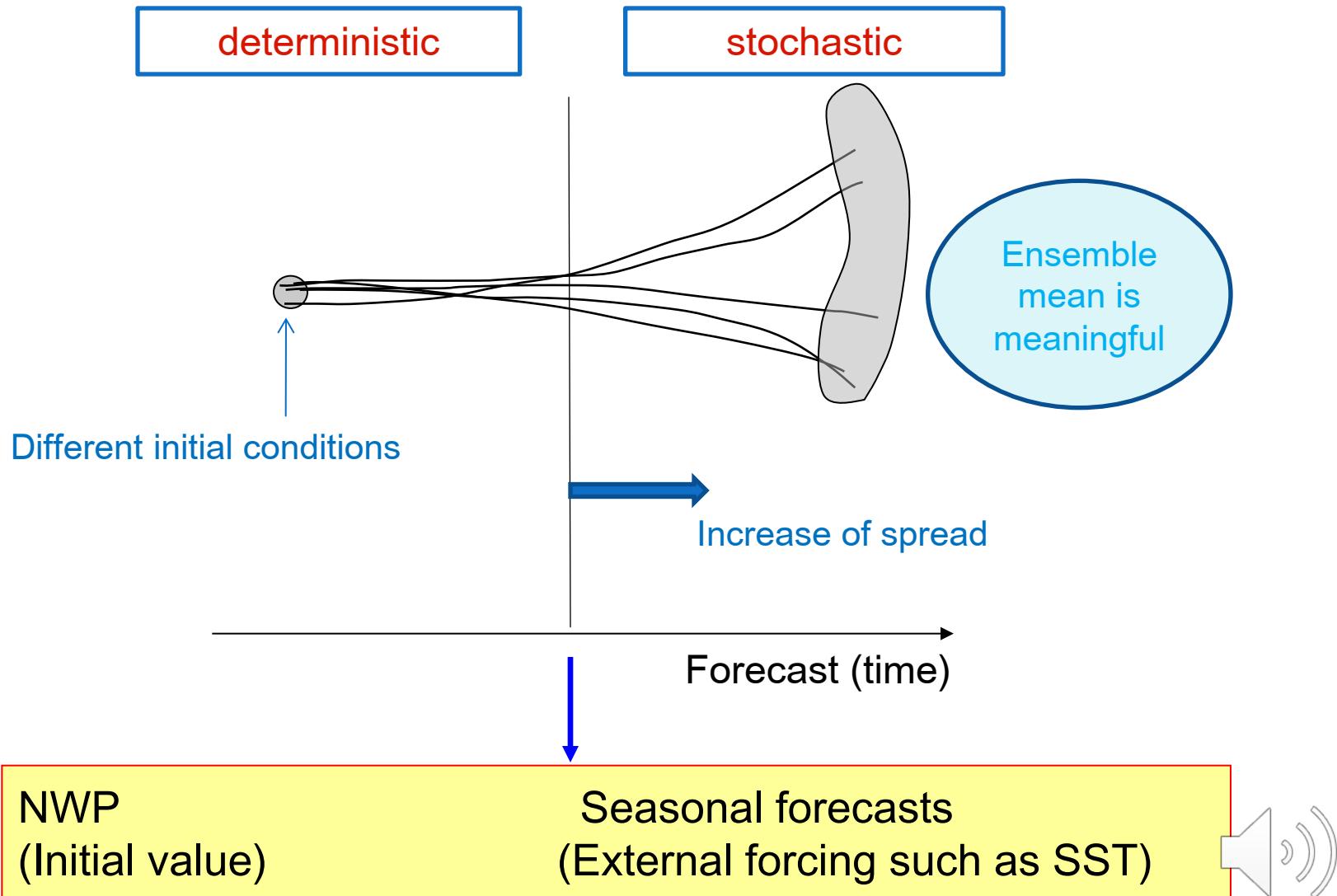
Chaos theory– two weeks for NWP



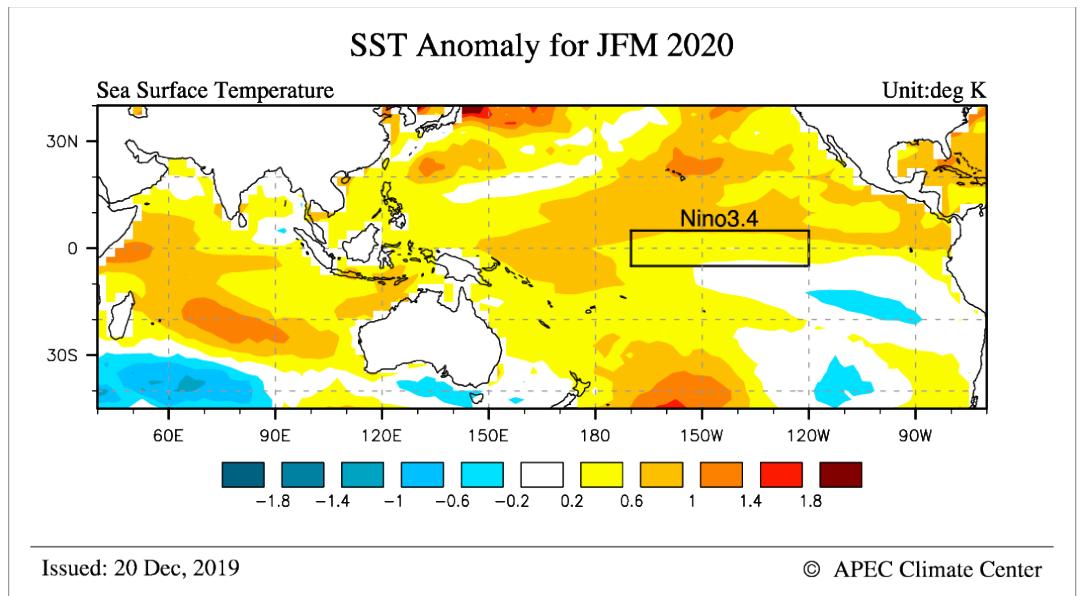
Predictability : Atmosphere is chaotic



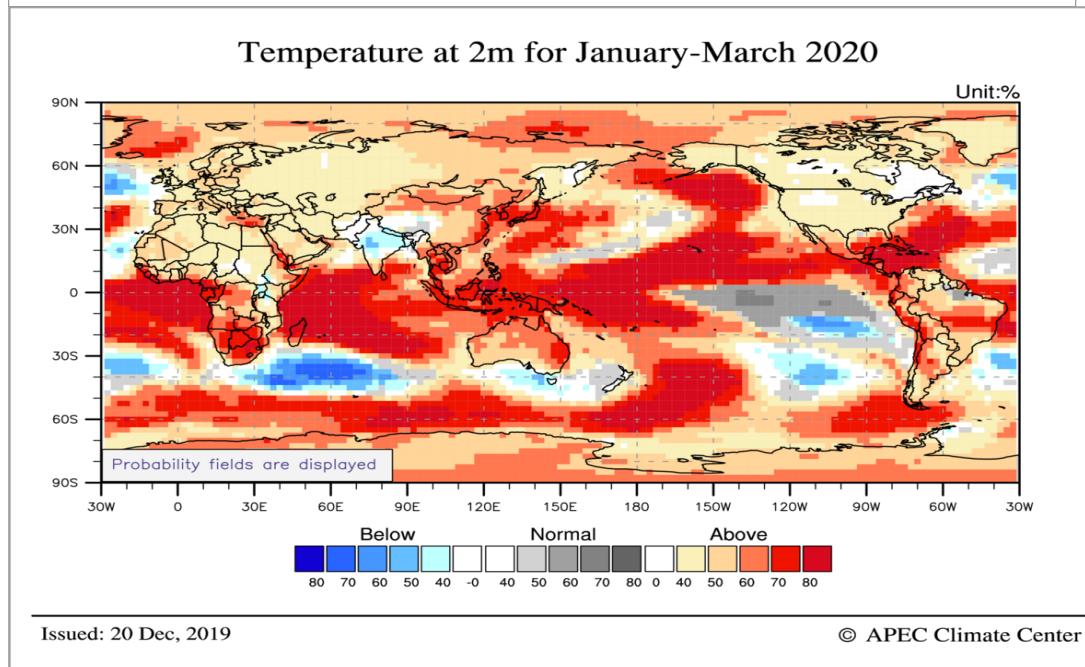
Ensemble forecasts : Seasonal and beyond



Ensemble forecasts : Seasonal and beyond



SST anomaly in
JFM 2020

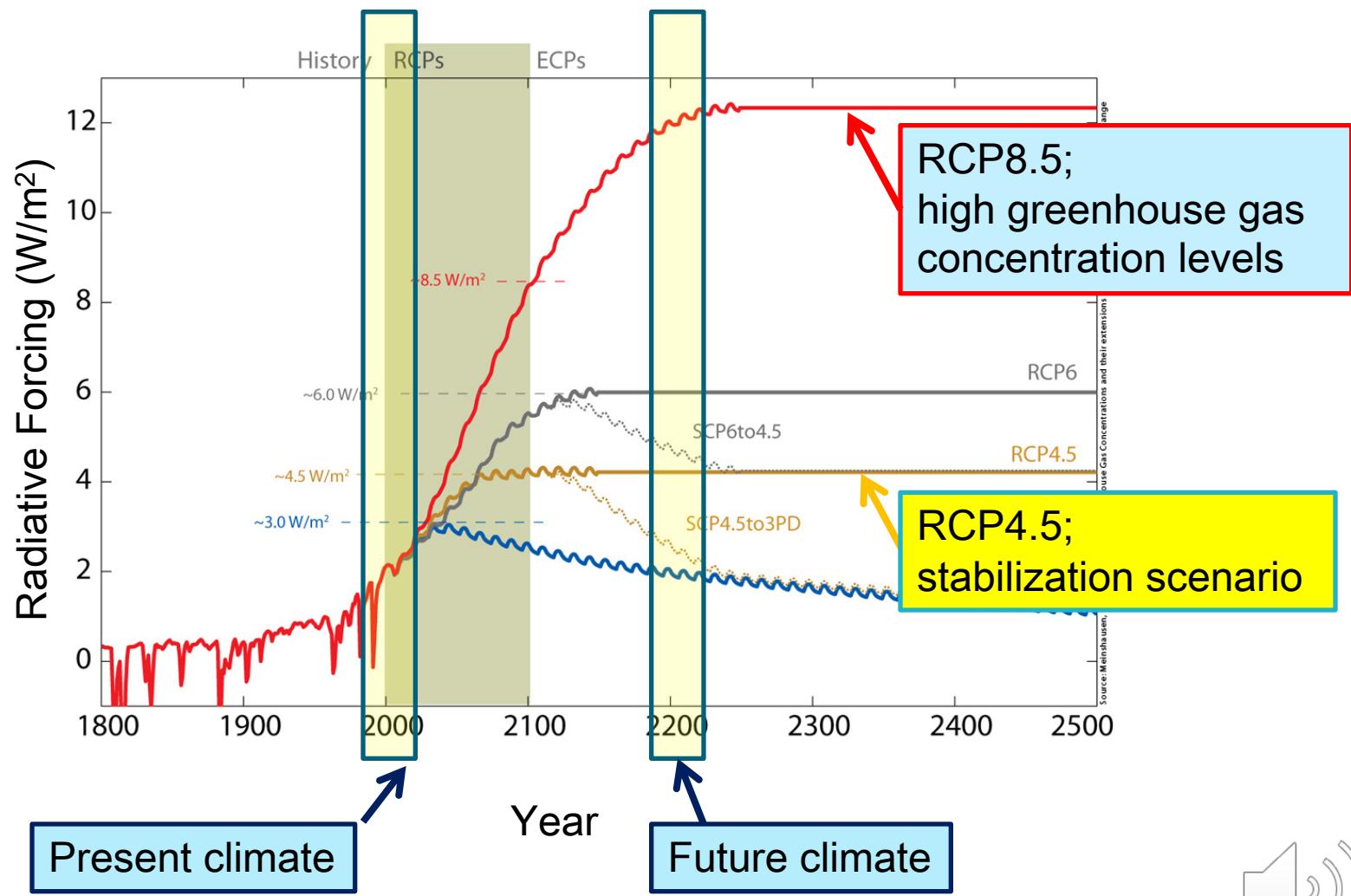


Temperature
anomaly in JFM 2020



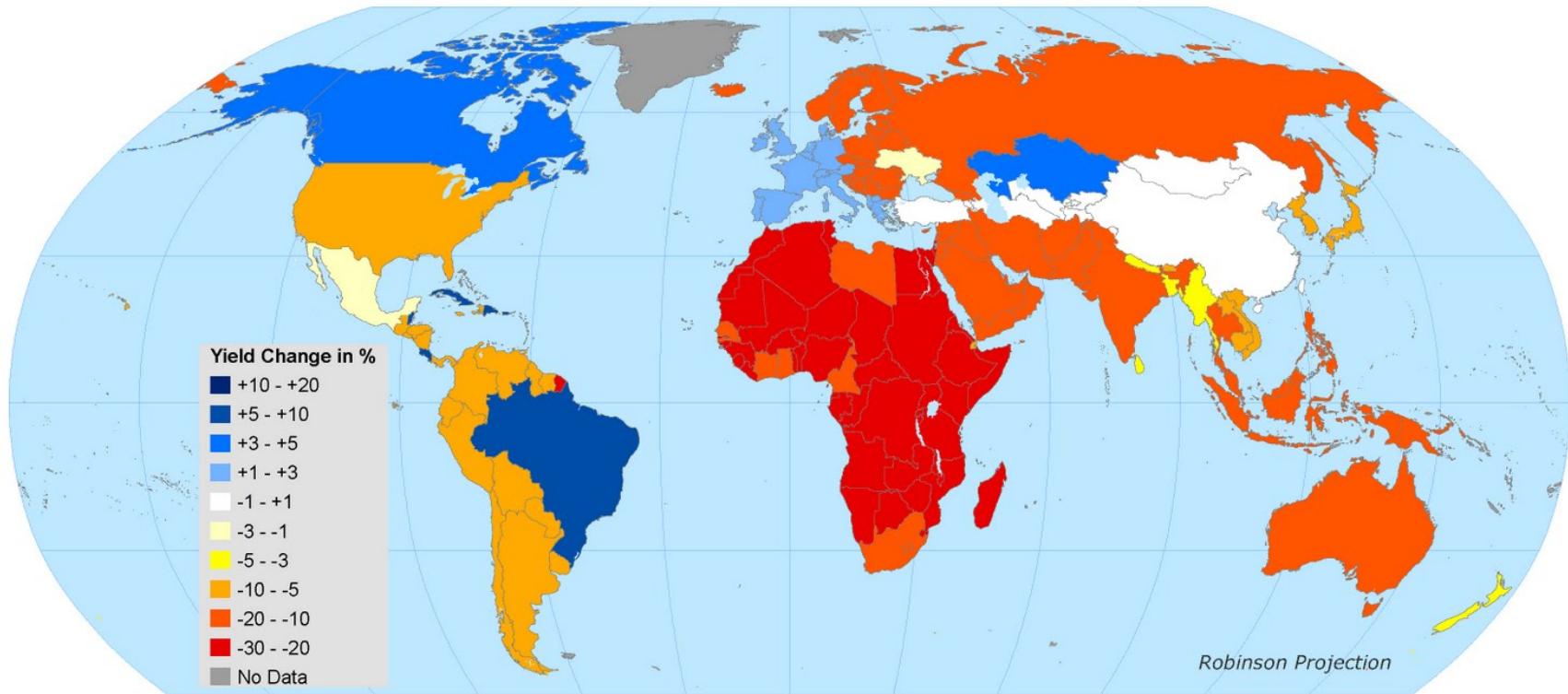
Climate prediction : For given representative change pathway (RCP) scenarios,

Climate changes = future minus present



Climate prediction : For given RCP scenarios

Effects of Climate Change on Global Food Production



Projected Maize Yield Change in %

1970-2000 Baseline to 2080, SRES A1F Scenario



Copyright 2010. The Trustees of Columbia University in the City of New York.
Source: Iglesias, A., and C. Rosenzweig. 2010. Effects of Climate Change
on Global Food Production. Data available at
<http://sedac.ciesin.columbia.edu/mva/cropclimate/>

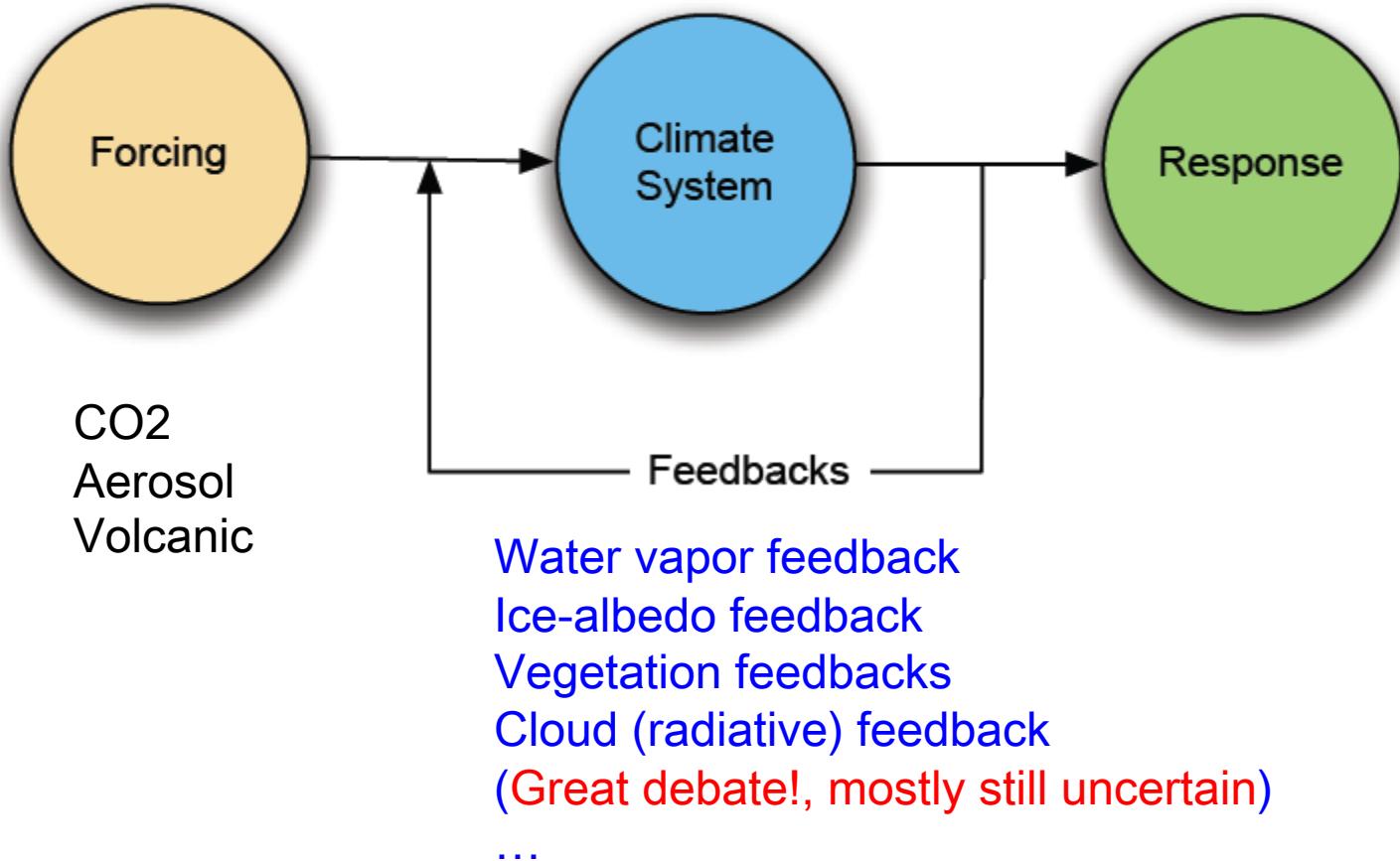
Publish Date: March 2010

This map is for illustrative purposes and does not imply the expression of any opinion on the part of the co-authors, CIESIN, or their sponsors concerning the legal status of any country or territory or concerning the delimitation of frontiers or boundaries.



This document is licensed under a
Creative Commons 3.0 Attribution License
<http://creativecommons.org/licenses/by/3.0/>

Climate prediction : Climate system sensitivity



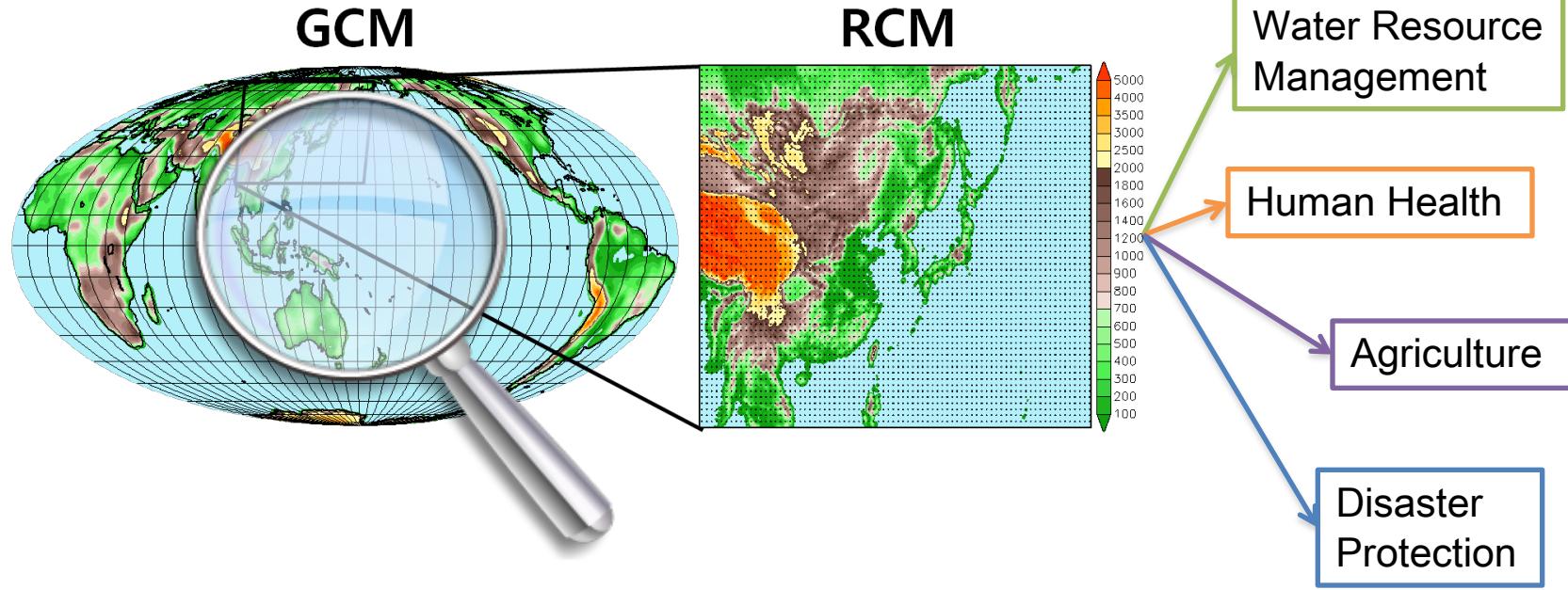
NWP / GCM : models could be unified.



Global vs Regional



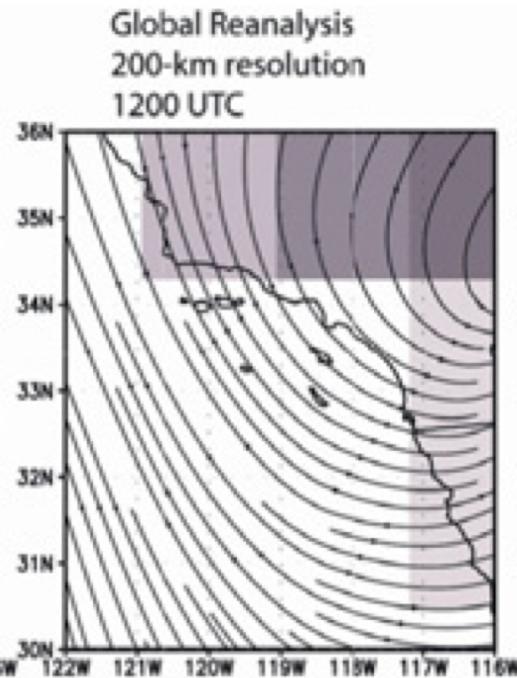
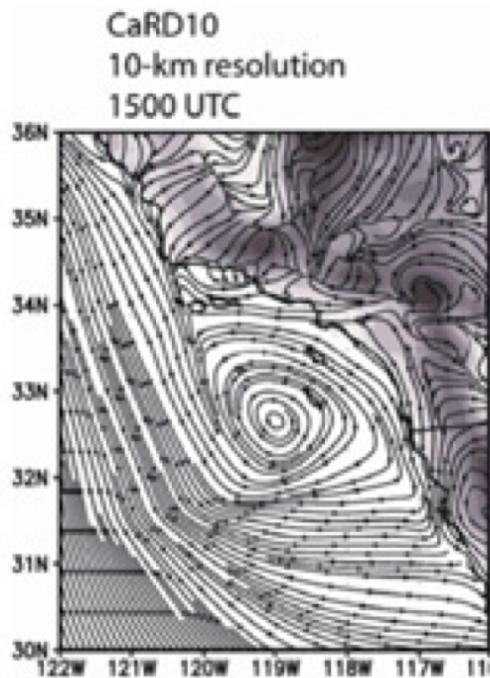
Regional modeling : need for applications



Regional model is a magnifying glass



High resolution benefit ? ---- Very clear !



10 km

200 km

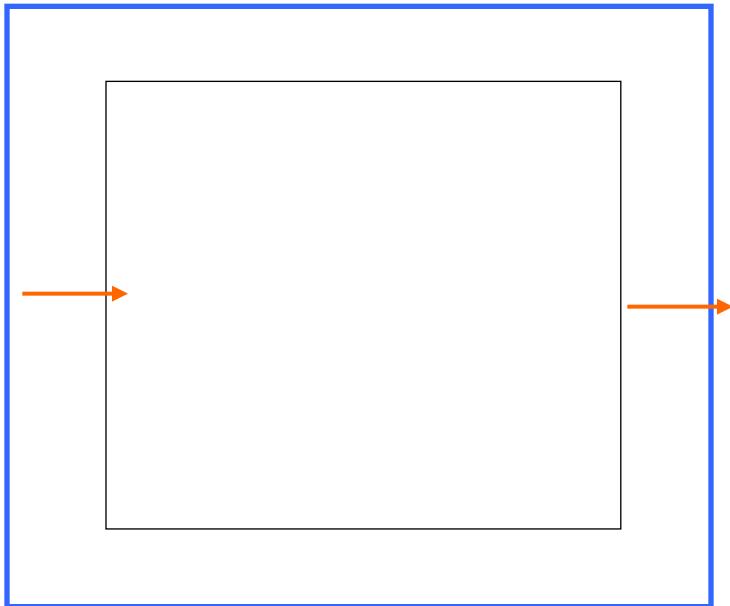
Observed



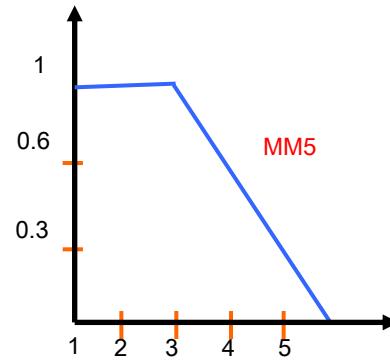
Another inherent issue in regional modeling

: lateral boundary treatment is empirical

Buffer zone



$F(n)$: weighting of global

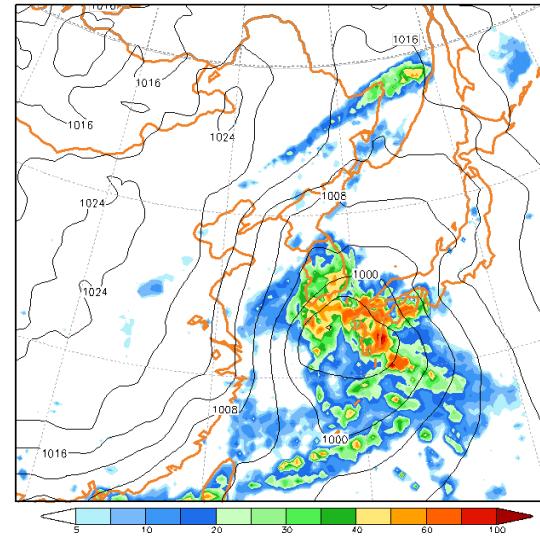
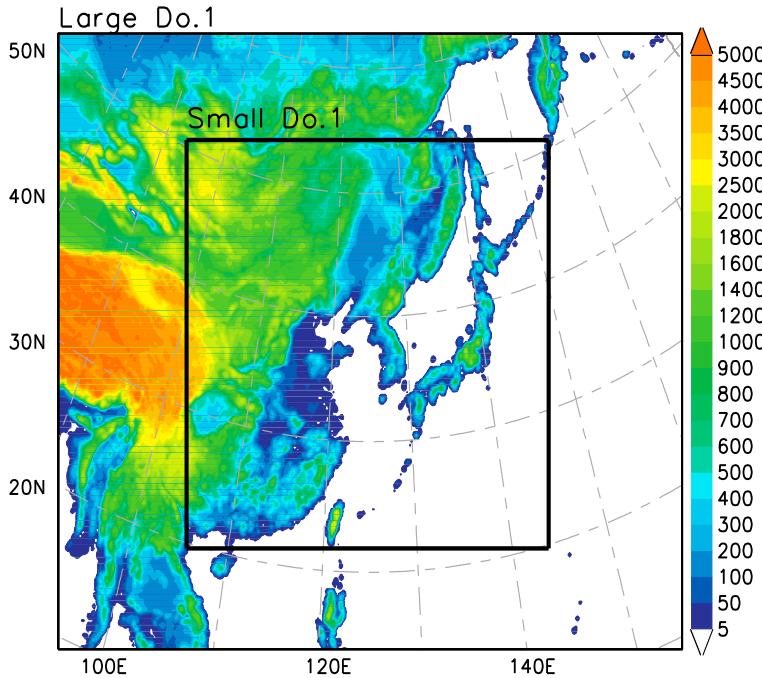


$$\frac{\partial A}{\partial t} \Big|_n = F(n)F_1(A_{CM} - A_{FM}) - F(n)F_2\nabla^2(A_{CM} - A_{FM})$$

So, empirical



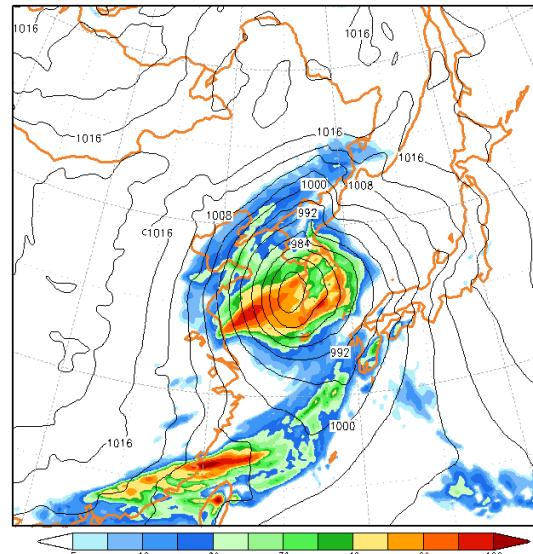
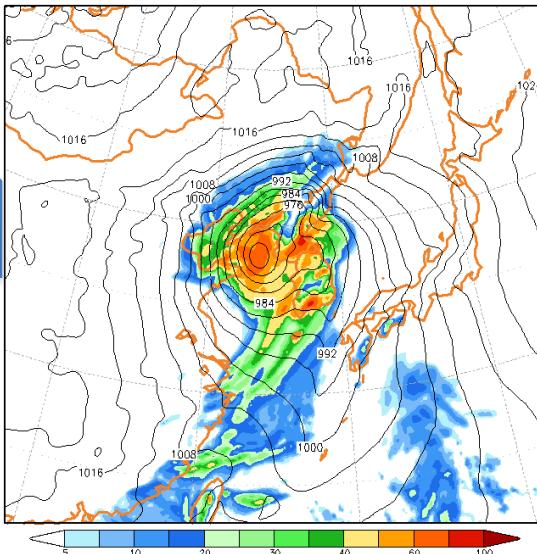
Domain size sensitivity : A mid-latitude cyclone



Observed

Large

Away from OBS
But more freedom

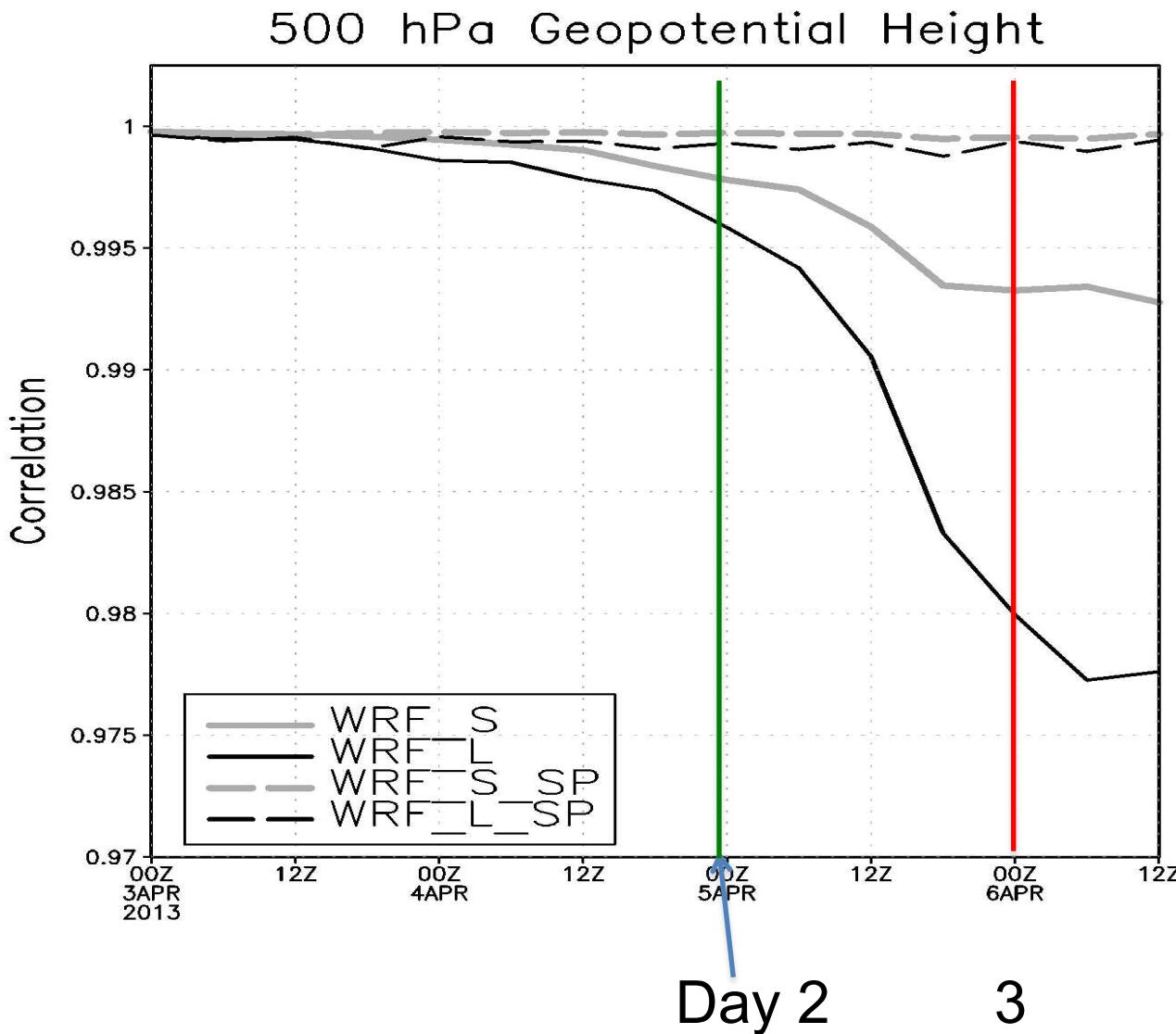


Small

Close to OBS
But less freedom



Domain size sensitivity : Pattern correlation with global



Fundamental limit of the regional model : low resolution global and mathematically ill-posed setup

Small domain keeps the large-scale from the global but loses its freedom

Spectral nudging keeps the large-scale, but may lose the regional details



Thanks for your attention !
songyouhong@gmail.com

Hong, S.-Y., and M. Kanamitsu, 2014: Dynamical downscaling: Fundamental issues from an NWP point of view and recommendations. *Asia-Pac. J. Atmos. Sci.*, 50, 83-104, doi: 10.1007/s13143-014-0029-2.

Dudhia, J., 2014: A history of mesoscale model Development. *Asia-Pac. J. Atmos. Sci.*, 50, 121-131.

