

# Data Assimilation - A01. Introduction -

Shunji Kotsuki

Center for Environmental Remote Sensing / Institute of Advanced Academic Research  
( [shunji.kotsuki@chiba-u.jp](mailto:shunji.kotsuki@chiba-u.jp) )



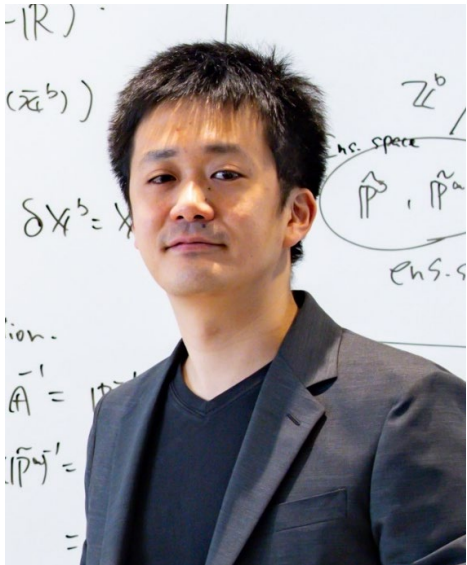
CHIBA  
UNIVERSITY



# DA Lectures A (Basic Course)

- ▶ (1) Introduction and NWP
- ▶ (2) Deterministic Chaos and Lorenz-96 model
- ▶ (3) A toy model and Bayesian estimation
- ▶ (4) Kalman Filter (KF)
- ▶ (5) 3D Variational Method (3DVAR)
- ▶ (6) Ensemble Kalman Filter (PO method)
- ▶ (7) Serial Ens. Square Root Filter (Serial EnSRF)
- ▶ (8) Local Ens. Transform Kalman Filter (LETKF)
- ▶ (9) Innovation Statistics
- ▶ (10) Adaptive Inflation
- ▶ (11) 4D Variational Method (4DVAR)

# 自己紹介: 小槻 峻司



(38歳)

趣味:  
言葉の収集

- 1986.05 高知県高知市に生まれる
- 2005.04 京都大学 工学部 地球工学科 入学
- 2013.11 同 大学院 工学研究科 工学博士
- 2014.01 理化学研究所 計算科学 特別研究員
- 2017.10 同 研究員
- 2019.10 千葉大学 環境リモセンセンター 准教授
- 2022.07 千葉大学 国際高等研究基幹 / 環境リモセン 教授

文部科学省関連の活動

- ・ 2017年 卓越研究員、2022年 文部科学大臣表彰 (若手)
- ・ 2020年~ 富岳成果加速・防災減災、2022年~ 気候変動予測・先端研究プロ

**現象科学**  
(土木・気象・水文)



**形式科学**  
(計算・統計数理)

JSTさきがけ・数理



**衛星・情報科学**  
(リモセン・AI)

内閣府MS・気象制御

# Brief Understanding of DA



DA is a kind of optimization.  
It is just a "tool" for empirical science.

Since DA uses observations,  
forecasts must be improved "easily"!

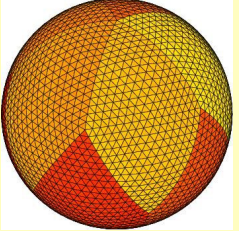


**No, DA is not a simplistic optimization  
Studying DA provides you deeper  
understanding on stat. math & dynamics.**

# Role of Data Assimilation in NWP

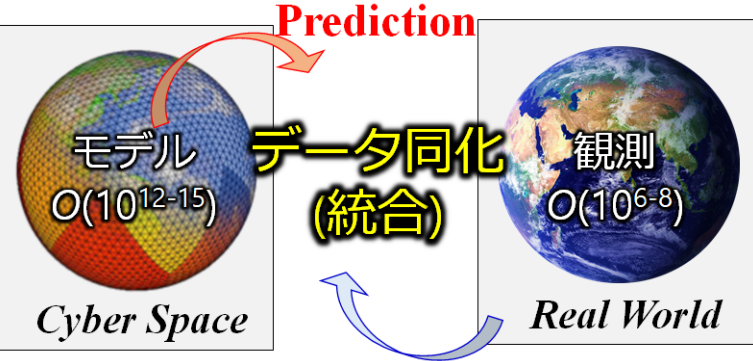
## Necessary researches for achieving better predictions

① NWP models  
→ Employ predictions from initial states



② Data assimilation  
→ Estimate initial states

Prediction



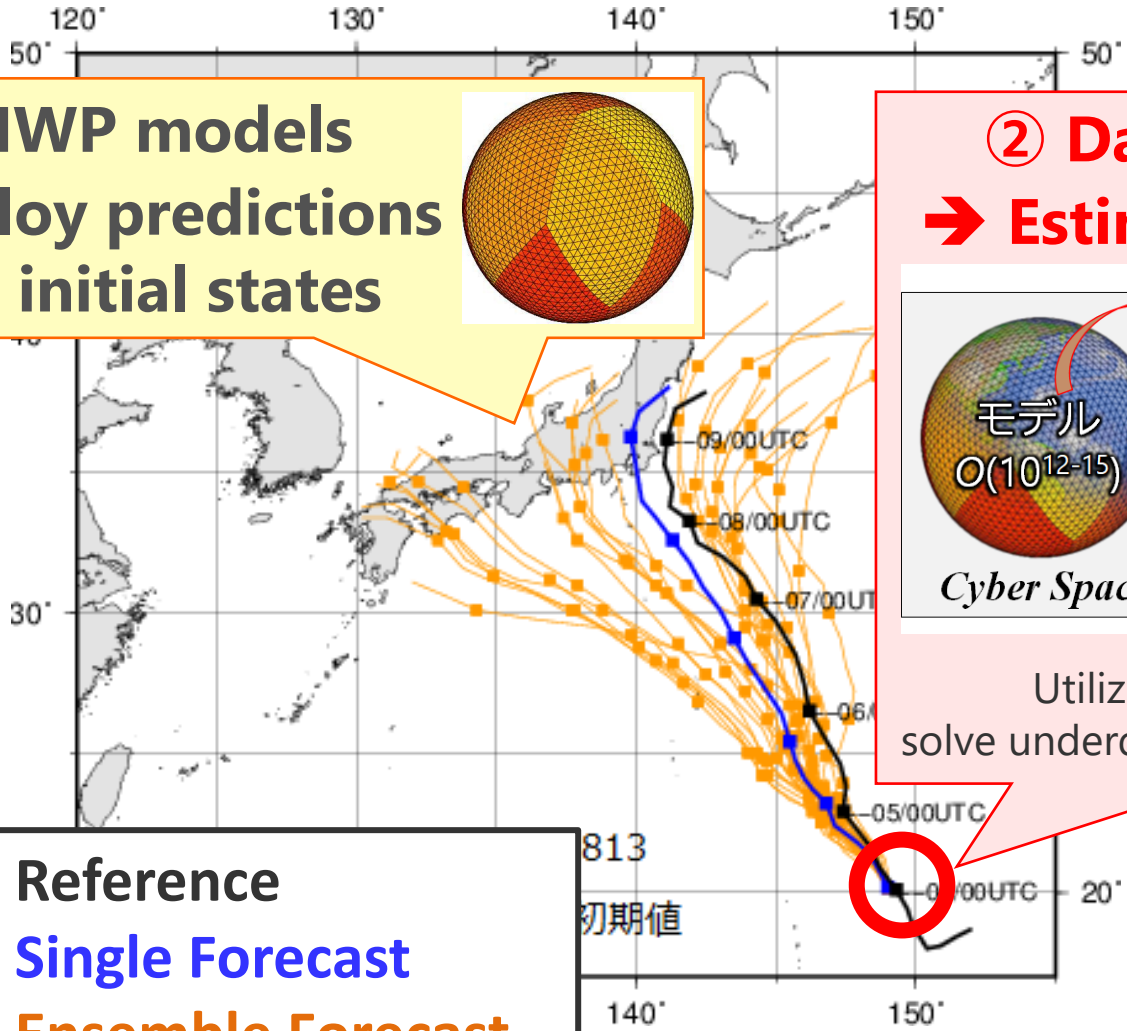
モデル  $O(10^{12-15})$  Cyber Space

データ同化 (統合)

観測  $O(10^{6-8})$  Real World

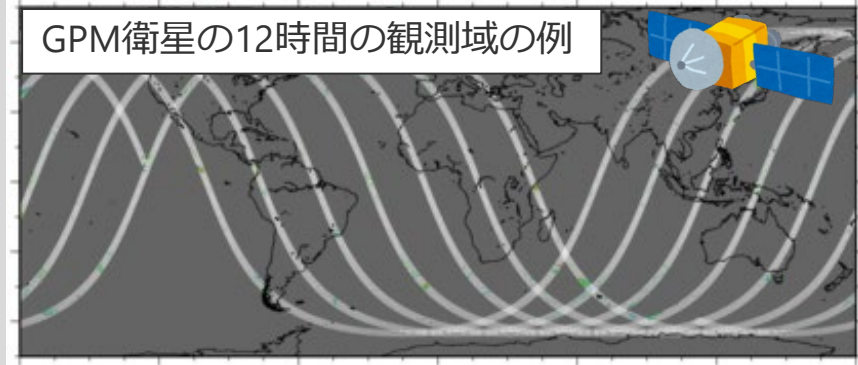
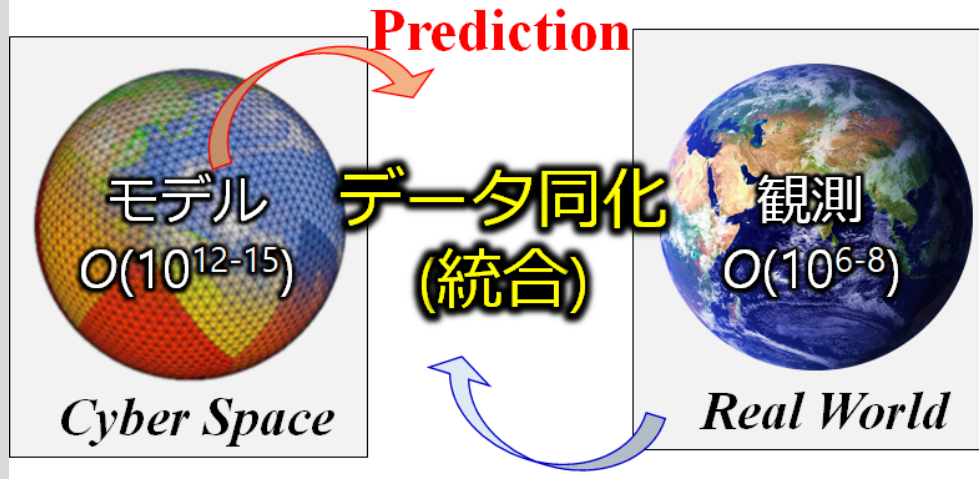
Utilizing sparsity of data to solve underdetermined inverse problems

—: Reference  
—: Single Forecast  
—: Ensemble Forecast



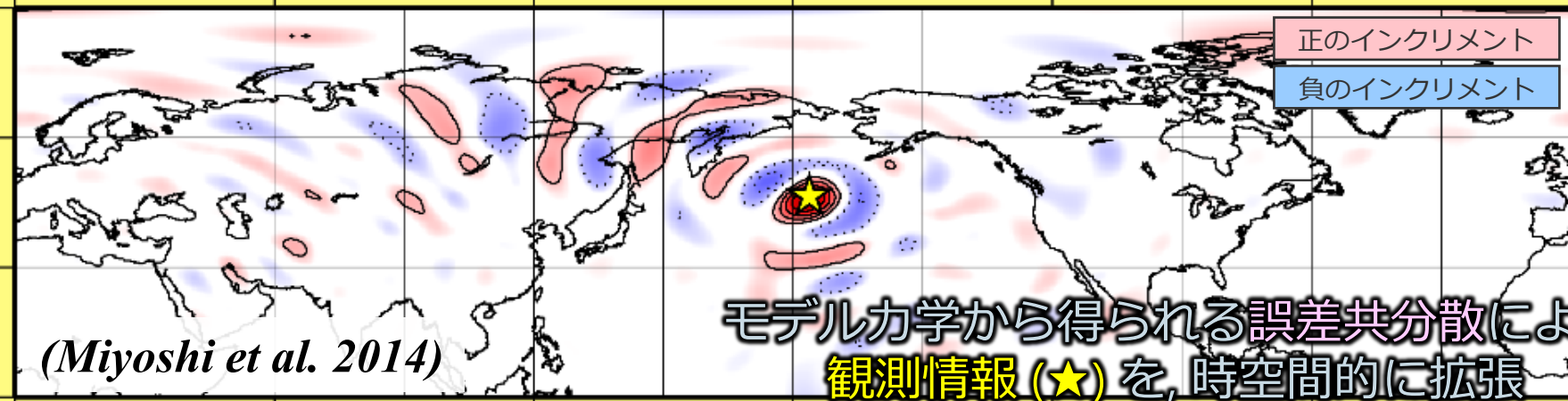
# データ同化: モデル&観測の統合

**データ同化**: 観測とモデルを統合し最適な状態推定を行う。  
天気予報で発展。観測が限られる場合に有効。



地球観測衛星でも時空間的に疎!  
→ 状態推定に **モデルカ学を使う同化が重要**

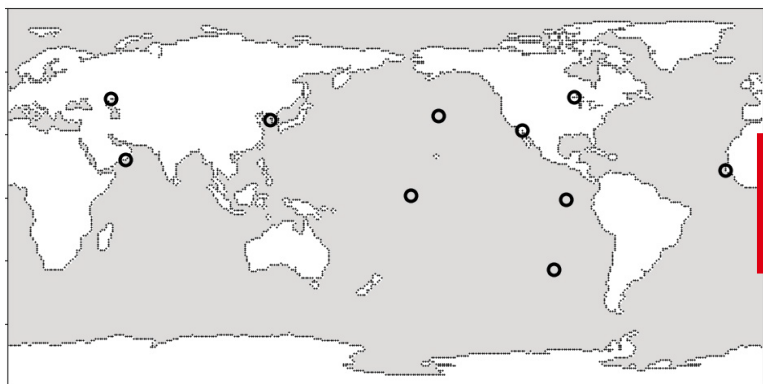
## 全球大気モデルを用いて推定した大気中層 (500hPa) の気温の空間誤差相関



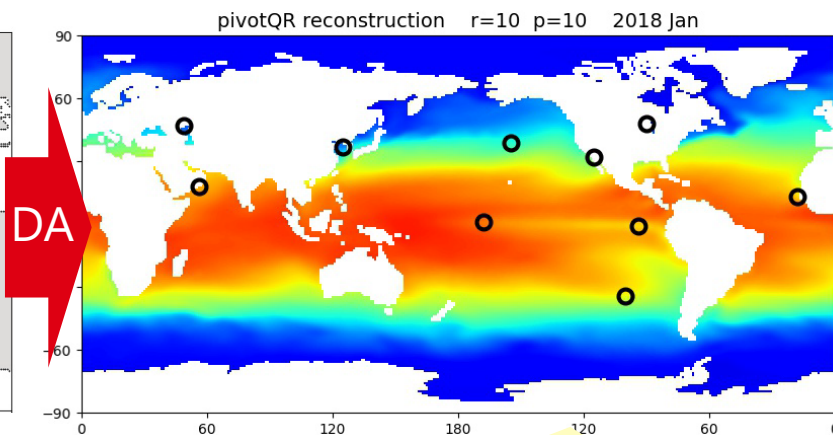
# Essence of data assimilation

*a case of sea surface temperature*

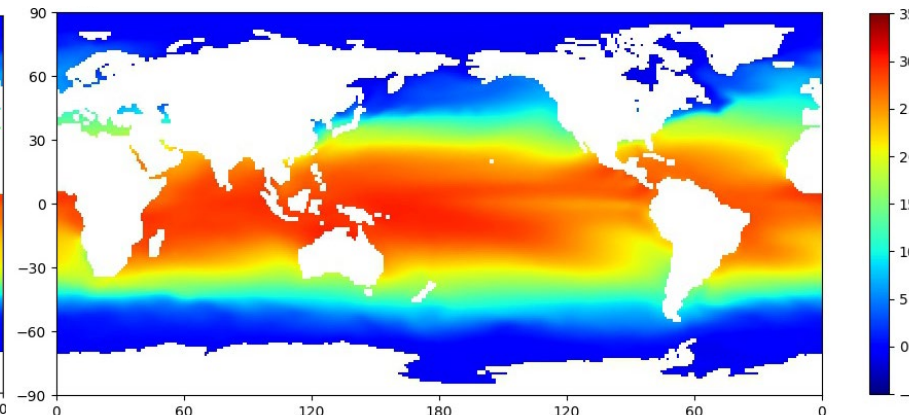
10 obs stations



**Data Assimilation**



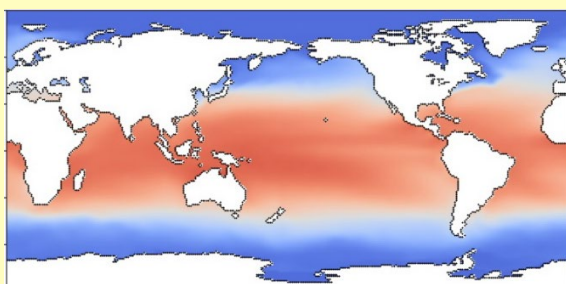
**Reference (Truth)**



*How can DA do this?*

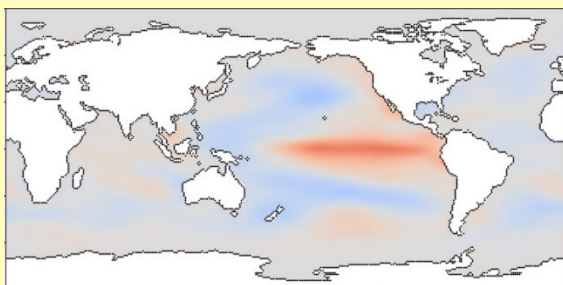
**Because SST can be represented as a superposition of modes.**

1st mode: average



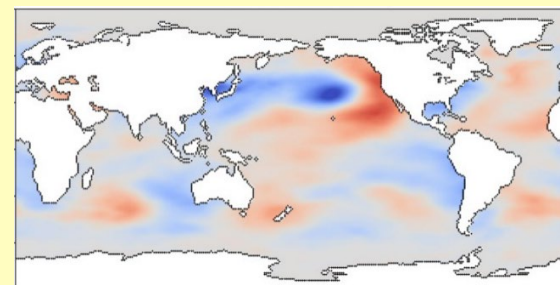
$\times a_1$

2nd mode: ENSO



$\times a_2$

3rd mode: ENSO Modoki?



$\times a_3$

**IMPORTANT MESSAGE!**

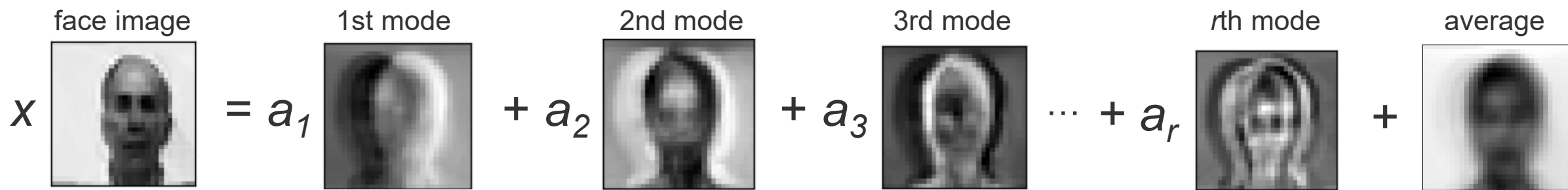
*The essential degrees of freedom is not high w.r.t. the data size.*

**Mode: eigenvector**

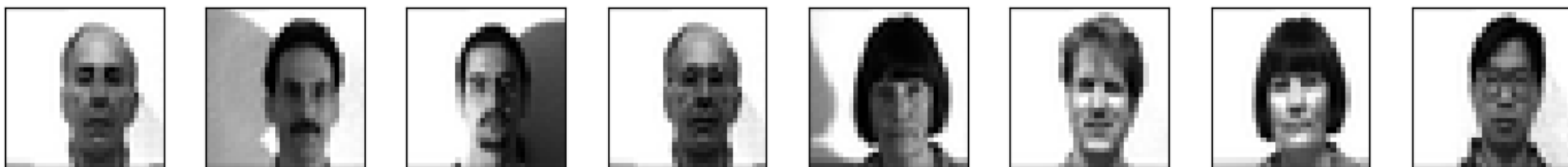
# Eigen modes (low-dim features) exist in many images

Facial images can also be represented as a superposition of modes

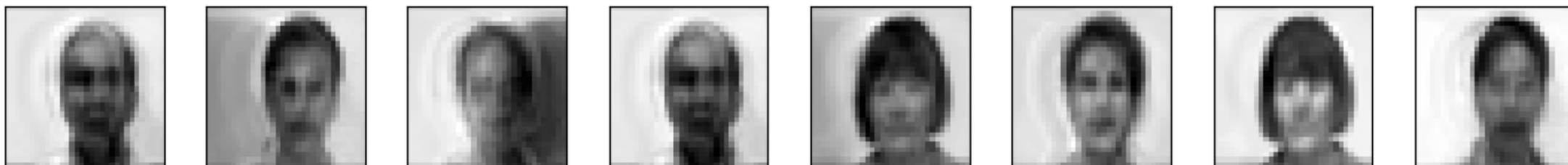
face image  $x$  =  $a_1$  1st mode +  $a_2$  2nd mode +  $a_3$  3rd mode + ... +  $a_r$   $r$ th mode + average



Original images



Represented by 10 modes



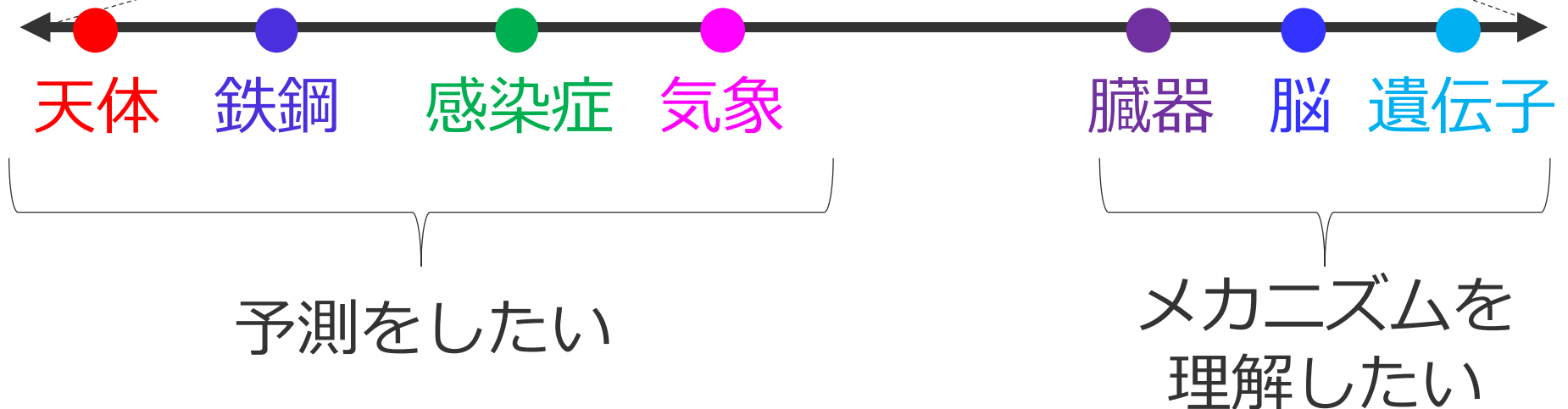
Represented by 50 modes





# データ同化研究の広がり

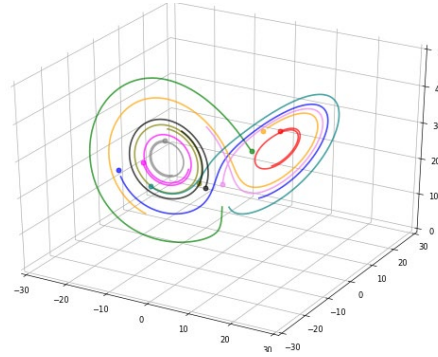
	シミュレーション (第3の科学)	データ同化 (結び付け)	機械学習 (第4の科学)
性質	プロセス駆動型		データ駆動型
観測	少ない		大きい
数理	既知 (硬い)		未知 (柔らかい)
モデル	大自由度		低自由度



# DA Research Strategy

## 1. math & toy models

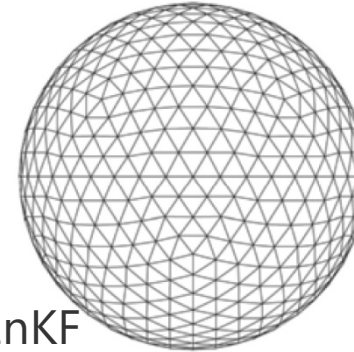
(e.g. Lorenz 96,  $n \sim O(10^2)$ ,  $p \sim O(10^2)$ )



## 2. intermediate models

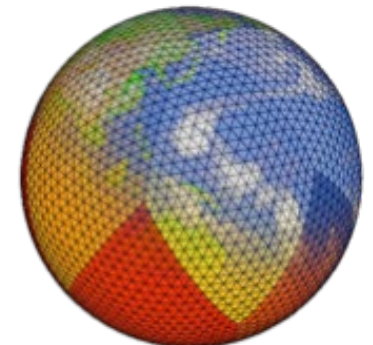
(e.g. SPEEDY,  $n \sim O(10^6)$ ,  $p \sim O(10^4)$ )

having  $n \times n$   $\mathbf{P}^b$  is unaffordable ( $> 100$  Gb)  $\rightarrow$  EnKF



## 3. realistic models

(e.g. NWP,  $n > O(10^8)$ ,  $p > O(10^6)$ )



# DA Study w/ 40-variable Lorenz-96

Lorenz-96 model (Lorenz 1996)

For  $j=1, \dots, N$ ,  $X_j = X_{j+N}$

$$dX_j / dt = \underbrace{(X_{j+1} - X_{j-2})X_{j-1}}_{\text{Advection term}} - \underbrace{X_j}_{\text{Dissipation term}} + \underbrace{F}_{\text{Forcing term}}$$

Advection term

Dissipation term

Forcing term

力学系モデル・データ同化基礎技術の速習コース

Training Course of Dynamical Model and Data Assimilation

January 31, 2020, Shunji Kotsuki

updated 2020/03/19, 2020/06/29, 2021/07/15

**目的:** 簡易力学モデル Lorenz の 40 変数モデル (以下 L96; Lorenz 1996) を使って複数のデータ同化手法を自ら実装し、様々な実験を行う。データ同化システムを実際に、0 からコーディングすることで、力学モデリングやデータ同化に関する実践的な「使える」基礎技術を体得する。

**Purpose:** Using the 40-variable dynamical a.k.a. Lorenz-96 (L96; Lorenz 1996), we are going to perform various experiments with multiple data assimilation (DA) methods. By actually coding a data assimilation system from scratch, you will acquire practically "usable" basic techniques related to mechanical modeling and data assimilation.

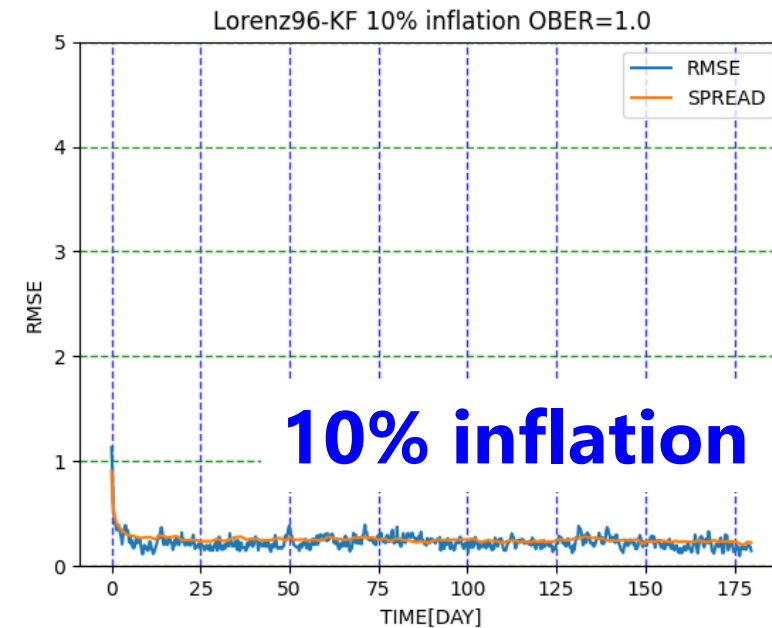
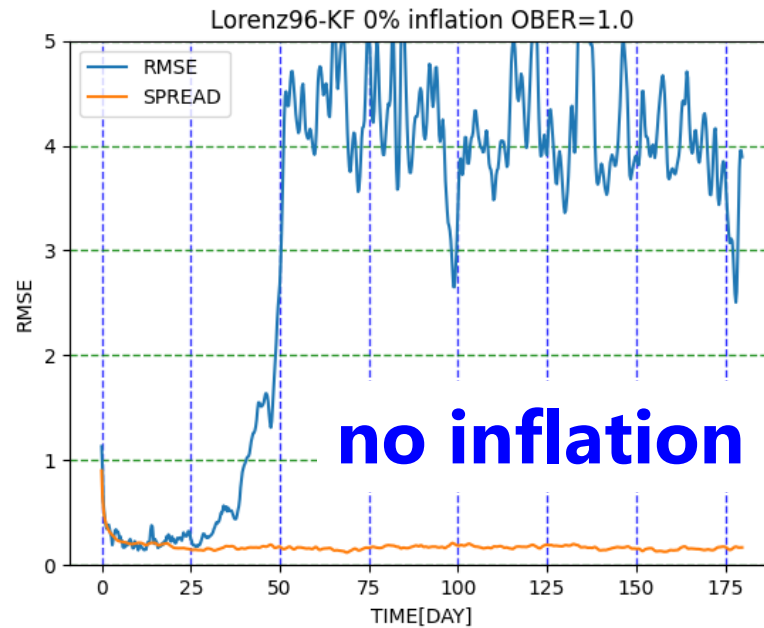
# Ex) Variance Inflation (KF, EnKF)

## Empirical treatment for variance underestimation due to

- (1) limited ensemble size
- (2) model nonlinearity
- (3) model imperfection

$$\mathbf{P}_{inf}^b = \alpha \times \mathbf{P}^b$$

*inflation factor (a tuning parameter)*



$$RMSE = \sqrt{\sum (x - x^{tru})^2 / n}$$

$$Spread = \sqrt{tr(\mathbf{P}^b) / n} = \sqrt{\sum \langle (x - x^{tru})^2 \rangle / n}$$

# Qiita記事もあります (in Jpn)

@elect-gombe が2020年04月17日に更新

## いろんなカルマンフィルターでデータ同化を実装する (L-96)

カルマンフィルター, Lorenz96

▲ この記事は最終更新日から1年以上が経過しています。

### データ同化の概要

モデルから求まる予報と観測を混ぜていい感じの解析値を得ることで観測に含まれる誤差を減らす。その混ぜる割合によって、誤差がどのくらいまで軽減されるのかが変わります。また、データ同化にはもう一つあって、観測データを予報空間に取り込む方法としても考えられます。この考え方を同化と呼ぶこともできます。

具体的には、制御工学的なフィルターの役割を果たすこともありますし、それとは別に気象現象のシミュレーションと現実世界を結びつける同化する目的でも使われます。

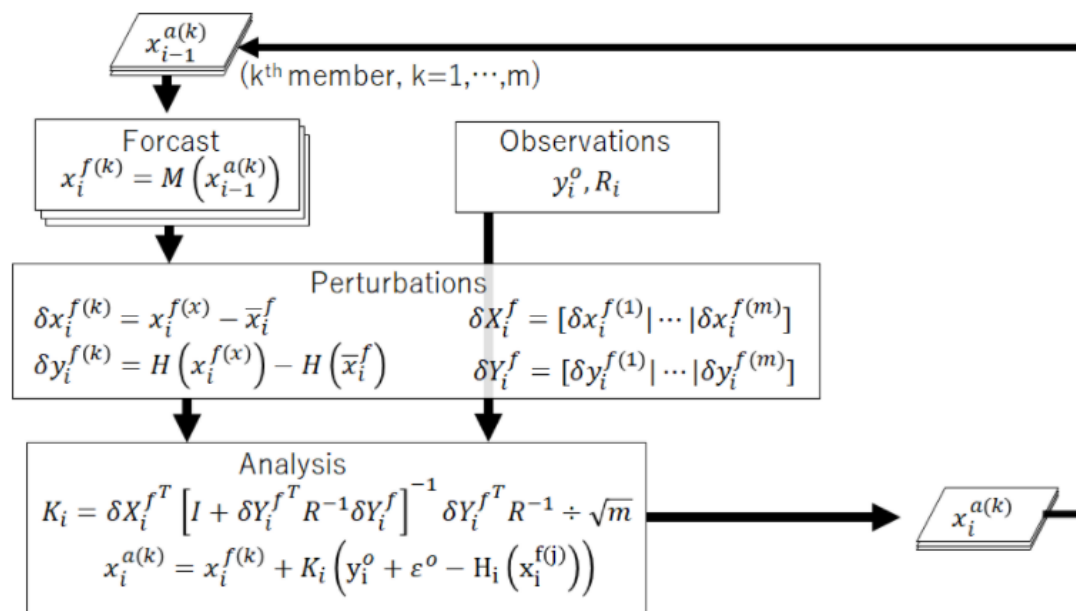
データ同化には大きく分けて予報ステップと解析ステップの二つがあります。

1. 一つ前の時刻の解析解からモデルを使って今の時刻の予報を求める
  2. その予報と観測から解析解を求める
- これらの実装に様々な方法が生まれます。

まず最初に、三次元変分法では、一定割合でデータを混ぜることでノイズを減らします。

### PO法

アンサンブルメンバーの生成などによってアンサンブルカルマンフィルタの実装には流派があります。この中の一つがPO法で、これは観測に観測の分散と同じだけの分散のノイズをわざと加えることで同じようなアンサンブルメンバーの同化を行えないのかという考え方です。

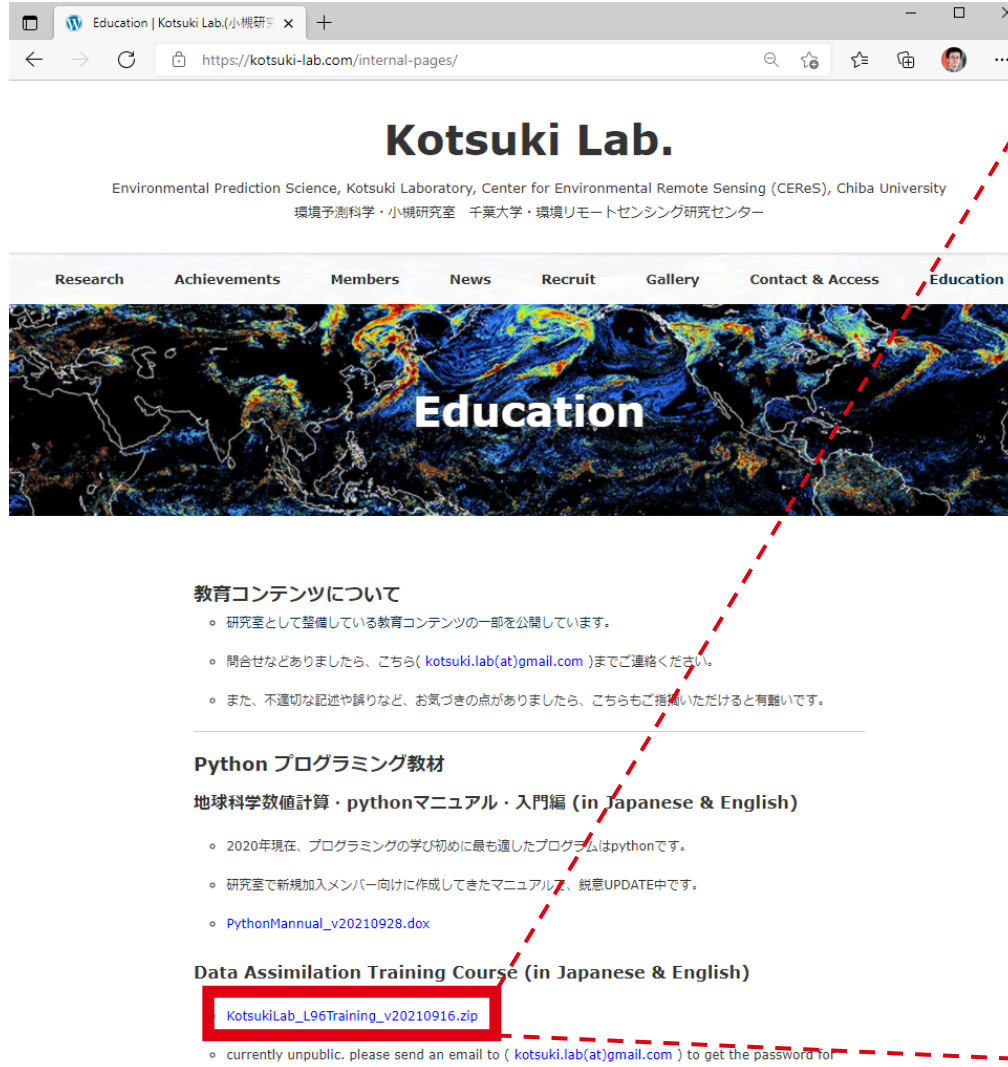


PO法をそのまま実装すると、サンプリングエラーによりうまく同化できないことがあります。サンプル数が少なくてうまく共分散行列が推定できなくなるということです。下の図を見てください。カルマンフィルターが理想的な誤差共分散行列です。



# Text Books

## ① Training Description



Education | Kotsuki Lab.(小槻研) x +  
https://kotsuki-lab.com/internal-pages/

### Kotsuki Lab.

Environmental Prediction Science, Kotsuki Laboratory, Center for Environmental Remote Sensing (CEReS), Chiba University  
環境予測科学・小槻研究室 千葉大学・環境リモートセンシング研究センター

Research Achievements Members News Recruit Gallery Contact & Access Education

## Education

#### 教育コンテンツについて

- 研究室として整備している教育コンテンツの一部を公開しています。
- 問合せなどありましたら、こちら( [kotsuki.lab\(at\)gmail.com](mailto:kotsuki.lab(at)gmail.com) )までご連絡ください。
- また、不適切な記述や誤りなど、お気づきの点がありましたら、こちらまでご連絡いただけると有難いです。

#### Python プログラミング教材

##### 地球科学数値計算・pythonマニュアル・入門編 (in Japanese & English)

- 2020年現在、プログラミングの学び初めに最も適したプログラムはpythonです。
- 研究室で新規加入メンバー向けに作成してきたマニュアルが、鋭意UPDATE中です。
- [PythonManual\\_v20210928.dox](#)

#### Data Assimilation Training Course (in Japanese & English)

[KotsukiLab\\_L96Training\\_v20210916.zip](#)

- currently unpublic. please send an email to ( [kotsuki.lab\(at\)gmail.com](mailto:kotsuki.lab(at)gmail.com) ) to get the password for

pswd: ceres

### 力学系モデル・データ同化基礎技術の速習コース

#### Training Course of Dynamical Model and Data Assimilation

January 31, 2020, Shunji Kotsuki

updated 2020/03/19, 2020/06/29, 2021/07/15

**目的:** 簡易力学モデル Lorenz の 40 変数モデル (以下 L96; Lorenz 1996) を使って複数のデータ同化手法を自ら実装し、様々な実験を行う。データ同化システムを実際に、0 からコーディングすることで、力学モデリングやデータ同化に関する実践的な「使える」基礎技術を体得する。

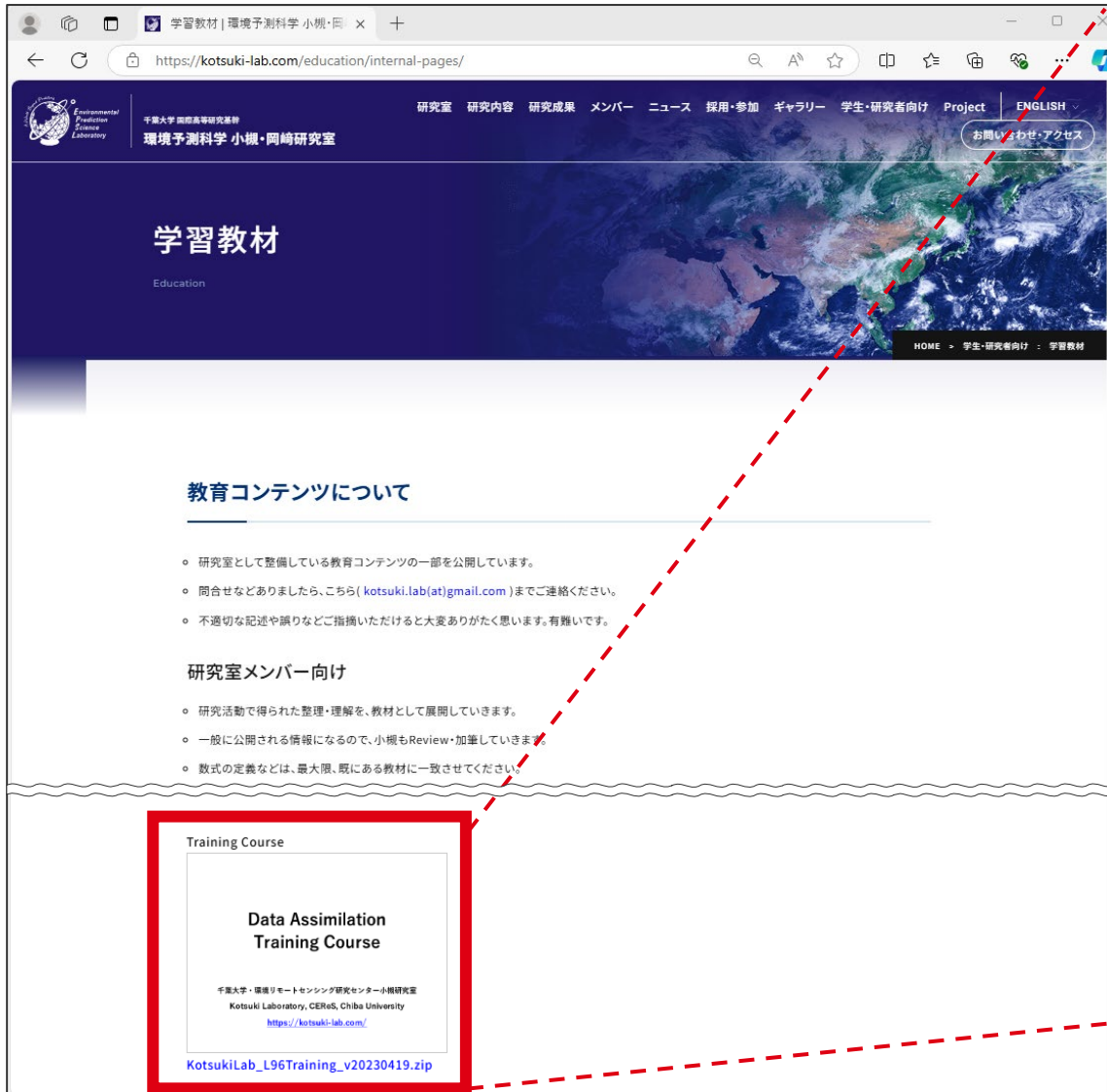
**Purpose:** Using the 40-variable dynamical a.k.a. Lorenz-96 (L96; Lorenz 1996), we are going to perform various experiments with multiple data assimilation (DA) methods. By actually coding a data assimilation system from scratch, you will acquire practically "usable" basic techniques related to mechanical modeling and data assimilation.

**方法:** 以下の課題を自ら実装し、解決していく。使用言語やプラットフォームは問わない。研究室の MTG において、各自が進捗を報告し、問題点を解消していく。質問は MTG の他も、居室で適宜受け付ける。使用言語については、特に拘りがなければ、行列演算の容易な python が扱いやすい。また、単精度ではなく倍精度でコーディングする事。でないと、既往研究と比較して正しく動作しているか確認できない。

**Method:** Implement and solve the following problems yourself. Any programming languages or platforms can be used in this exercise. At the Kotsuki Lab. mtg, each personnel will report the progress, and try to solve the problems. Questions are accepted during the MTG as well as at the office when necessary. As for the programming language, python, which is easy to perform matrix operations, is recommended unless specific language is preferred. Also, you should code in double precision instead of single precision. Otherwise, confirming whether performing properly or not compared to the previous studies will not be possible.



## ① Training Description



The screenshot shows the website for the Environmental Prediction Science Laboratory. The main heading is "学習教材" (Education). Below it, there is a section titled "教育コンテンツについて" (About Educational Content) with three bullet points. Further down, there is a section titled "研究室メンバー向け" (For Lab Members) with three bullet points. At the bottom, a red box highlights a download link for "Data Assimilation Training Course" with the filename "KotsukiLab\_L96Training\_v20230419.zip".

pswd: ceres

### 力学系モデル・データ同化基礎技術の速習コース

#### Training Course of Dynamical Model and Data Assimilation

January 31, 2020, Shunji Kotsuki

updated 2020/03/19, 2020/06/29, 2021/07/15

**目的:** 簡易力学モデル Lorenz の 40 変数モデル (以下 L96; Lorenz 1996) を使って複数のデータ同化手法を自ら実装し、様々な実験を行う。データ同化システムを実際に、0 からコーディングすることで、力学モデリングやデータ同化に関する実践的な「使える」基礎技術を体得する。

**Purpose:** Using the 40-variable dynamical a.k.a. Lorenz-96 (L96; Lorenz 1996), we are going to perform various experiments with multiple data assimilation (DA) methods. By actually coding a data assimilation system from scratch, you will acquire practically "usable" basic techniques related to mechanical modeling and data assimilation.

**方法:** 以下の課題を自ら実装し、解決していく。使用言語やプラットフォームは問わない。研究室の MTG において、各自が進捗を報告し、問題点を解消していく。質問は MTG の他も、居室で適宜受け付ける。使用言語については、特に拘りがなければ、行列演算の容易な python が扱いやすい。また、単精度ではなく倍精度でコーディングする事。でないと、既往研究と比較して正しく動作しているか確認できない。

**Method:** Implement and solve the following problems yourself. Any programming languages or platforms can be used in this exercise. At the Kotsuki Lab. mtg, each personnel will report the progress, and try to solve the problems. Questions are accepted during the MTG as well as at the office when necessary. As for the programming language, python, which is easy to perform matrix operations, is recommended unless specific language is preferred. Also, you should code in double precision instead of single precision. Otherwise, confirming whether performing properly or not compared to the previous studies will not be possible.







# Voice from Students



工学部・情報・4年生

比較的簡単にEnKFまでクリア。  
データ同化にハマる学生続出。



理学部・地球科学科・4年生

プログラミングについてはほとんど知らない状態から始めました。地球科学科の場合、プログラミングは3年の前期にある授業でしか学ばず、それもfortranで平均の計算や大きい順にソーティングするくらいしか学んでません（必修ではない）。「プログラミングってこんな感じか」という感覚をつかんでいるくらいでも、十分に役立ちます。

Pythonは全く知らず、小槻研のマニュアルを土台に、調べながら進めました。プログラミングスキルが高くない場合、進みは遅いかもですが、やる気があればトレーニングコースは完走できると思います！同化面白い！！

# For Beginners of Python

## ▶ Further Information

▶ <https://kotsuki-lab.com/internal-pages/>

Python Programming Training



学習教材 | 環境予測科学 小槻 研

https://kotsuki-lab.com/education/internal-pages/

千葉大学 環境予測科学 小槻 研  
環境予測科学 小槻 研研究室

学習教材  
Education

教育コンテンツについて

Python プログラミング教材

地球科学数値計算・pythonマニュアル・入門編 (in Japanese & English)

2020年現在、プログラミングの学び初めに最も適したプログラムはpythonです。  
研究室で新規加入メンバー向けに作成してきたマニュアルで、観意UPDATE中です。  
また、教科書化に向けて出版社と調整中です。

Pythonを利用した地球科学・数値解析入門  
Python Training Course for Earth Science

千葉大学・環境リモートセンシング研究センター・小槻研究室  
Kotsuki Laboratory, CEReS, Chiba University  
<https://kotsuki-lab.com/>

PythonManual\_v20220808.docx



地球科学・Python入門 / Python Training Course for Earth Science (ver 2.0).

In English  
& Japanese

Pythonを利用した地球科学・数値解析入門  
Python Training Course for Earth Science

千葉大学・環境リモートセンシング研究センター・小槻研究室  
Kotsuki Laboratory, CEReS, Chiba University  
<https://kotsuki-lab.com/>

contact information: ↓  
[shunji.kotsuki@chiba-u.jp](mailto:shunji.kotsuki@chiba-u.jp)

1 / 141.

# Python Training Course (cont'd)

地球科学・Python 入門 / Python Training Course for Earth Science (ver 2.0)<sup>4</sup>

## 目次<sup>4</sup>

目次.....	3 <sup>4</sup>
Outline.....	4 <sup>4</sup>
1. はじめに (Introduction).....	5 <sup>4</sup>
1.1. なぜ Python を推奨するのか? (Why Python?).....	5 <sup>4</sup>
1.2. 本資料の構成 (Structure of this manual).....	8 <sup>4</sup>
1.3. お願い と 感謝 (Request and appreciation).....	8 <sup>4</sup>
1.4. Python のダウンロード Downloading Python.....	9 <sup>4</sup>
1.5. 本書の注意書き (Caution).....	15 <sup>4</sup>
1.6. スクリプトの実行の仕方 (How to run the script).....	18 <sup>4</sup>
1.7. ライブラリのインストールの仕方 (How to install the library).....	20 <sup>4</sup>
1.8. ノートブックの便利な機能の紹介 (Introducing useful features of notebooks).....	21 <sup>4</sup>
2. 基本演習 (Basic exercises).....	24 <sup>4</sup>
2.1. 基本演算 (Basic operations).....	24 <sup>4</sup>
2.2. 変数と配列 (Variables and arrays).....	26 <sup>4</sup>
2.3. 制御文 (Control statement).....	37 <sup>4</sup>
2.4. ファイルの入出力 (File input / output).....	43 <sup>4</sup>
2.5. グラフの描画 (Drawing a graph).....	45 <sup>4</sup>
2.6. script による命令実行 (復習) (Command execution by script (review)).....	54 <sup>4</sup>
2.7. Python の間違いやすい落とし穴 (Common mistakes in Python).....	56 <sup>4</sup>
3. Python によるテキストデータの解析 (Text data analysis with Python).....	62 <sup>4</sup>
3.1. 解析とは何か? (What is analysis?).....	62 <sup>4</sup>
3.2. 解析の工程 (Analysis process).....	63 <sup>4</sup>
3.3. 積算棒グラフの作成 (Creating an integrated bar graph).....	64 <sup>4</sup>
4. Python によるバイナリデータの処理 (Processing binary data with Python).....	81 <sup>4</sup>
4.1. バイナリデータとは (What is binary data).....	81 <sup>4</sup>
4.2. 二次元データのデータ格納方法 (How to store 2D data).....	82 <sup>4</sup>
4.3. 衛星データを扱う工程 (Process of handling satellite data).....	87 <sup>4</sup>
4.4. Ctl ファイル付きのバイナリデータ処理 (Binary data processing with Ctl file).....	87 <sup>4</sup>
5. Python によるデータの取得方法 (How to obtain data with Python).....	102 <sup>4</sup>
5.1. データの取得方法 (How to obtain data).....	102 <sup>4</sup>
5.2. データのダウンロード (Data download).....	110 <sup>4</sup>
5.3. 画像の作成 (Creating images).....	116 <sup>4</sup>
5.4. 国ごとの時系列グラフ (Time series graph by country).....	130 <sup>4</sup>
5.5. 加重平均 (Weighted average).....	132 <sup>4</sup>

地球科学・Python 入門 / Python Training Course for Earth Science (ver 2.0)<sup>4</sup>

## Outline<sup>4</sup>

Outline.....	3 <sup>4</sup>
1. Introduction.....	4 <sup>4</sup>
1.1. Why Python?.....	4 <sup>4</sup>
1.2. Structure of this manual.....	7 <sup>4</sup>
1.3. Request and appreciation.....	7 <sup>4</sup>
1.4. Downloading Python.....	8 <sup>4</sup>
1.5. Caution.....	14 <sup>4</sup>
1.6. How to run the script.....	17 <sup>4</sup>
1.7. How to install the library.....	19 <sup>4</sup>
1.8. Introducing useful features of notebooks.....	20 <sup>4</sup>
2. Basic exercises.....	23 <sup>4</sup>
2.1. Basic operations.....	23 <sup>4</sup>
2.2. Variables and arrays.....	25 <sup>4</sup>
2.3. Control statement.....	36 <sup>4</sup>
2.4. File input and output.....	42 <sup>4</sup>
2.5. Drawing a graph.....	44 <sup>4</sup>
2.6. Command execution by script (review).....	53 <sup>4</sup>
2.7. Common mistakes in Python.....	55 <sup>4</sup>
3. Text data analysis with Python.....	61 <sup>4</sup>
3.1. What is analysis.....	61 <sup>4</sup>
3.2. Analysis process.....	62 <sup>4</sup>
3.3. Creating an integrated bar graph.....	63 <sup>4</sup>
4. Processing binary data by Python.....	80 <sup>4</sup>
4.1. What is binary data.....	80 <sup>4</sup>
4.2. Data storage method for two-dimensional data.....	81 <sup>4</sup>
4.3. Process of handling satellite data.....	86 <sup>4</sup>
4.4. Dealing with binary data with ctl file.....	86 <sup>4</sup>
5. How to obtain data with Python.....	101 <sup>4</sup>
5.1. How to obtain data.....	101 <sup>4</sup>
5.2. Downloading data.....	109 <sup>4</sup>
5.3. Creating images.....	115 <sup>4</sup>
5.4. Time series graph by country.....	129 <sup>4</sup>
5.5. Weighted average.....	131 <sup>4</sup>

# For those who would like to

- ▶ **deepen understanding on DA**
  - ▶ → you will understand the essence of DA
- ▶ **Use DA for research**
  - ▶ → you will acquire practical skills
- ▶ **improve programming skill**
  - ▶ → you will improve programming definitely
- ▶ **know numerical weather prediction**
  - ▶ → you will be more familiar with weather

# Let's Start!

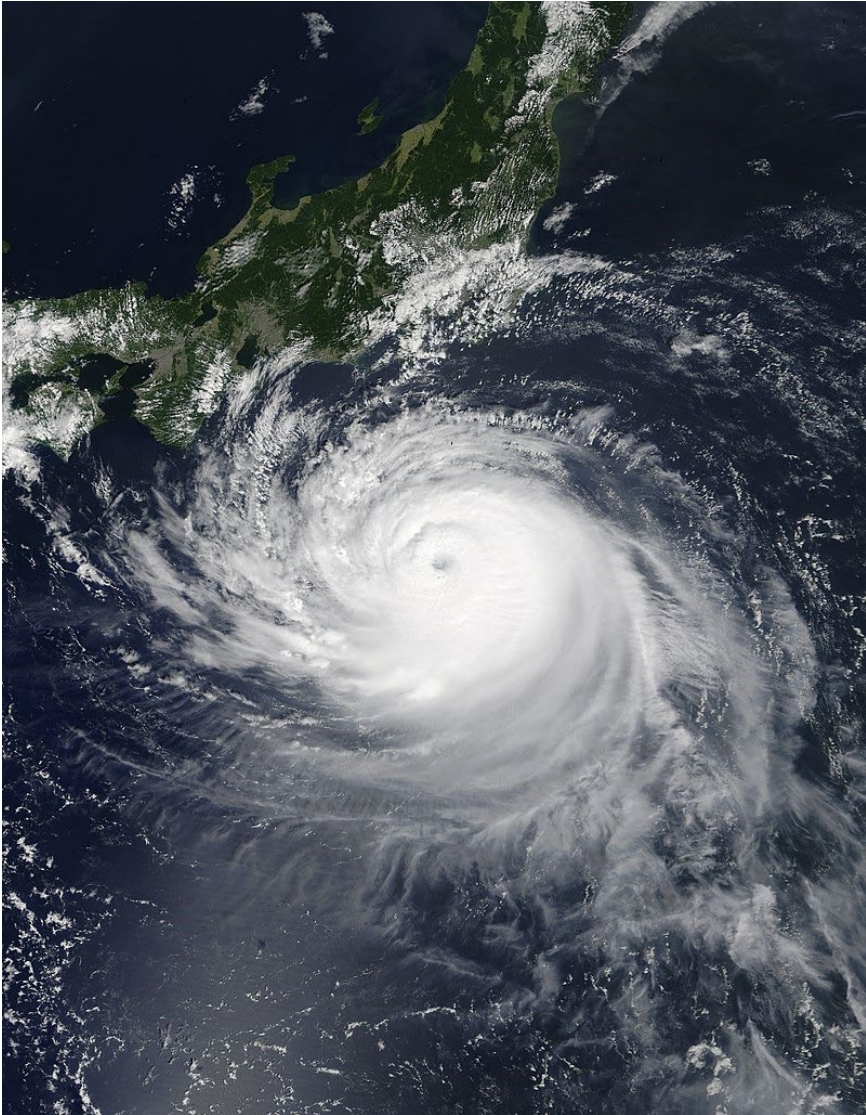
# Today's Goal

- ▶ To understand numerical weather prediction and the role of the data assimilation
- ▶ To be interested in data assimilation

# Data Assimilation & NWP

# 激化する気象災害

## 令和元年房総半島台風 (2019)



被災住居



倒壊した鉄柱@市原市

images from wikipedia, 市民防災研究所



# 支配方程式 (微分方程式) を解く,とは?

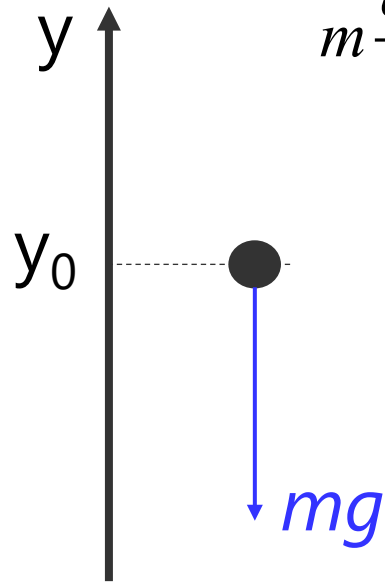
例) 運動方程式: 物体の運動を記述する方程式

$$m \frac{d^2 \mathbf{r}}{dt^2} = \mathbf{F} \xrightarrow{\text{積分}} \frac{d\mathbf{r}}{dt} \xrightarrow{\text{積分}} \mathbf{r}$$

運動方程式

$\mathbf{F}$ : カベクトル  $\mathbf{r}$ : 物体の位置ベクトル

一番簡単な例: 自由落下運動



$$m \frac{d^2 y}{dt^2} = -mg \Leftrightarrow \frac{d^2 y}{dt^2} = -g \xrightarrow{\text{積分}} \frac{dy}{dt} = -gt + C = v$$

$$\xrightarrow{\text{積分}} y = -\frac{1}{2}gt^2 + Ct + C'$$

(i) 初期条件  $v(0) = 0$  より  $C = 0$

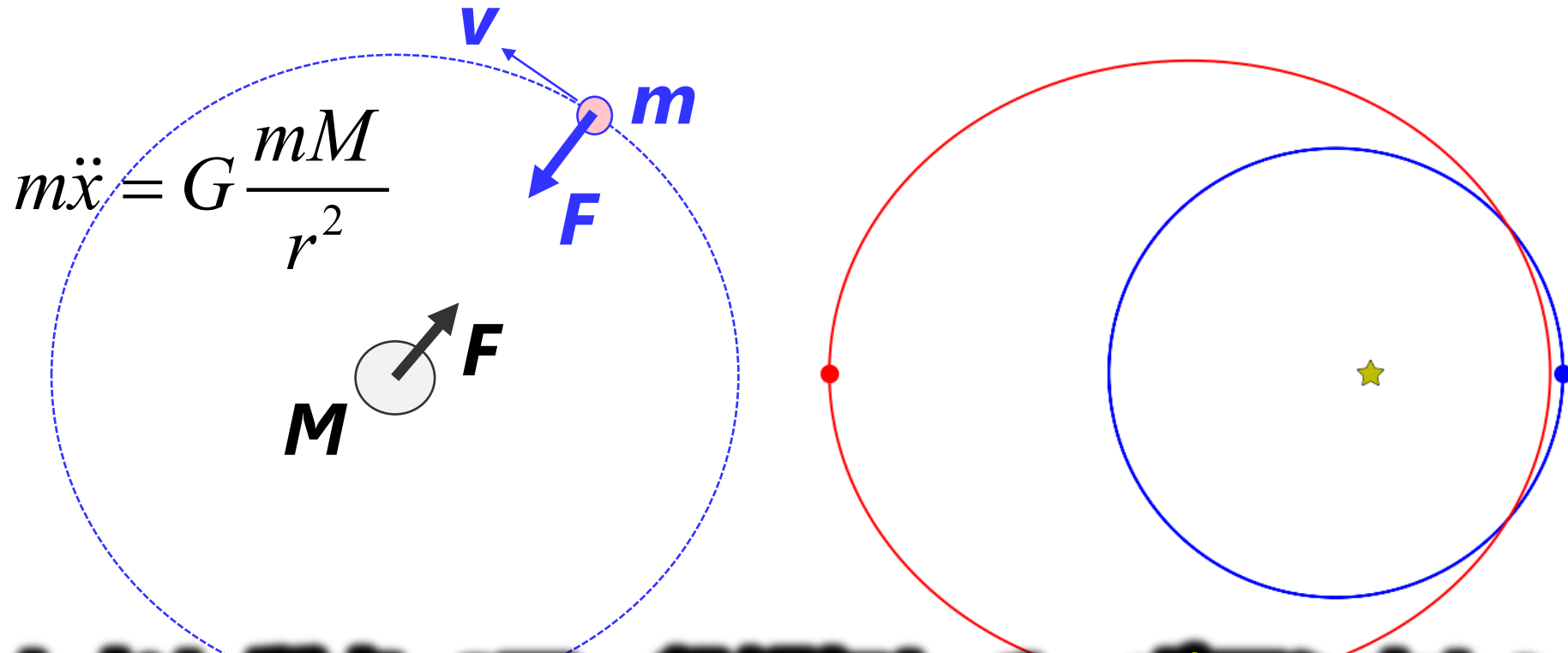
(ii) 初期条件  $y(0) = y_0$  より  $C' = y_0$

$$\therefore y = -\frac{1}{2}gt^2 + y_0$$

- ① 支配方程式を積分して初めて状態が分かる
- ② 予測には初期条件(積分定数)が必要

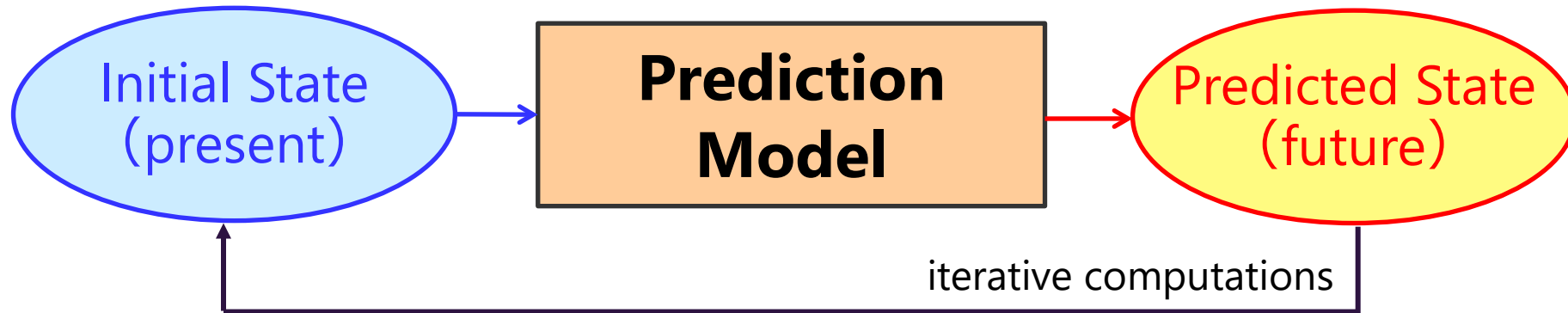
# ニュートン力学と求積解の限界

世界は時間について微分方程式 (運動方程式)  
で記述されていて積分で予測する (=求積する)

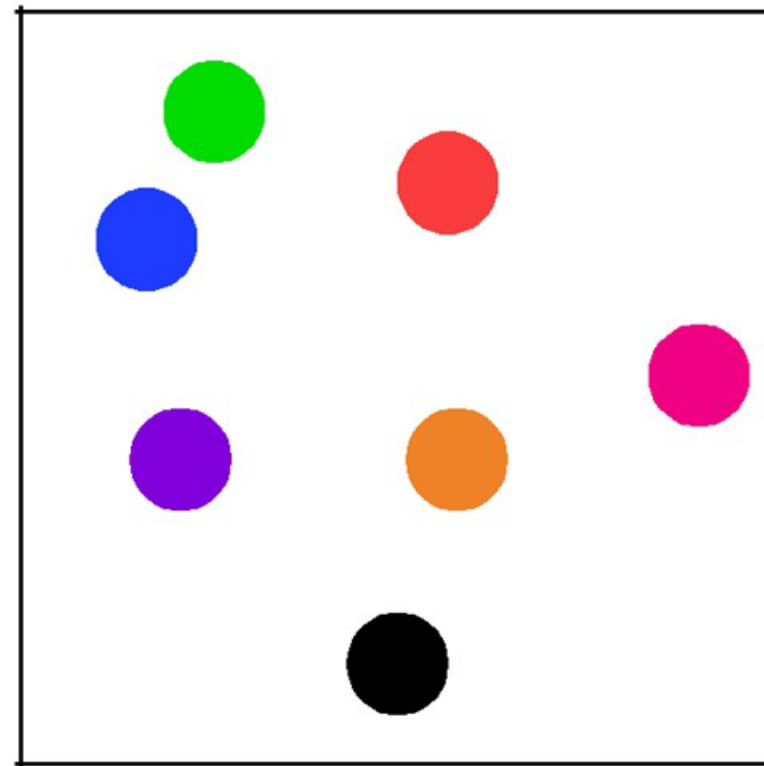


しかし質点 $\geq 3$ で一般解無し (by ポアンカレ)  
→ コンピュータを使った数値予測へ

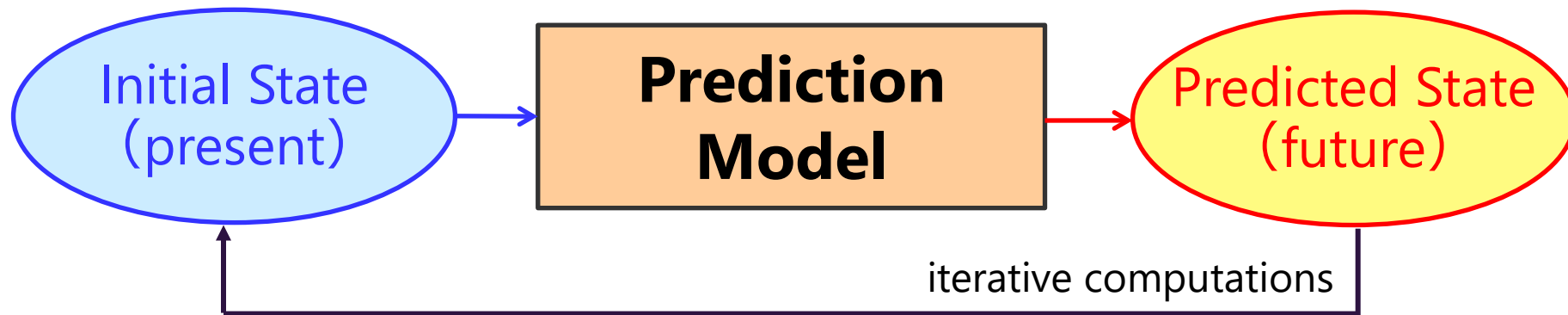
# Numerical Simulation w/ Computers



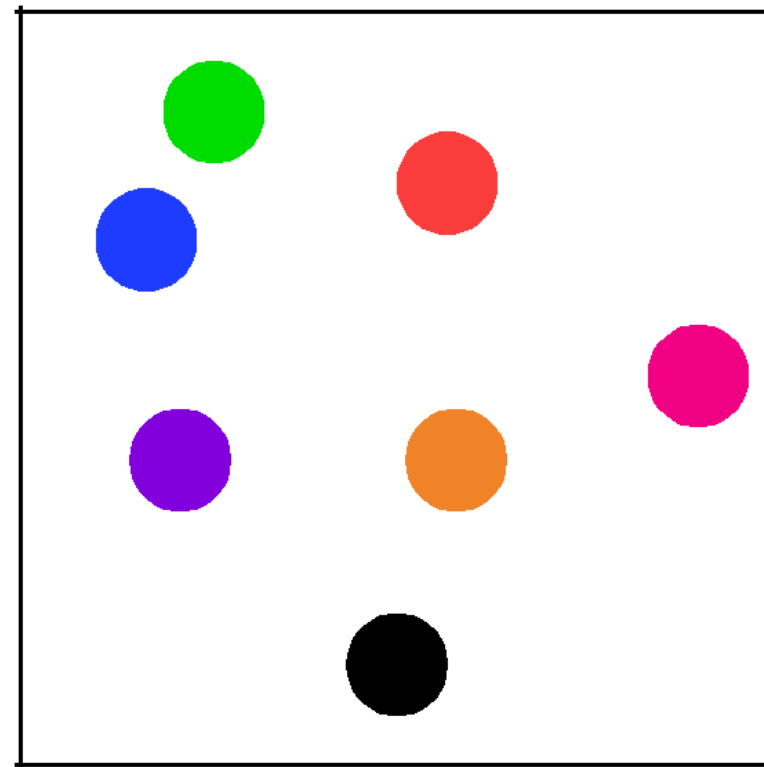
*an example  
of billiards*



# Numerical Simulation w/ Computers



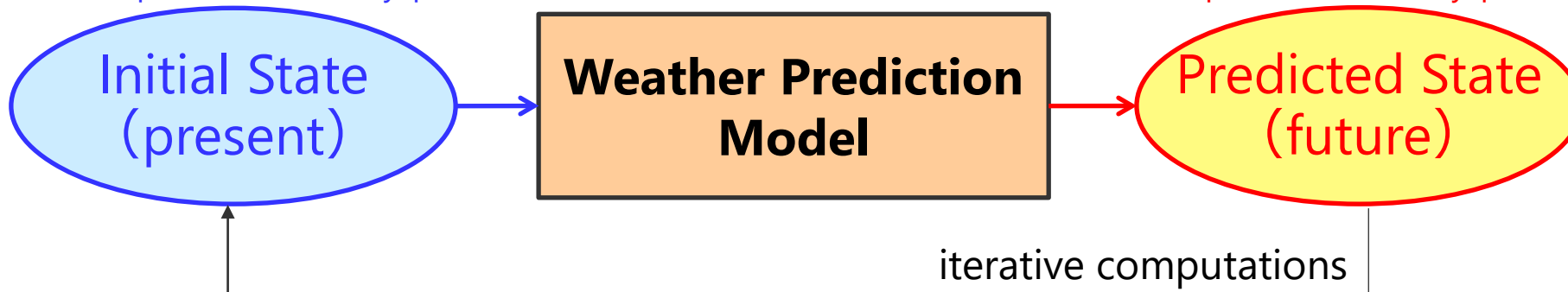
*an example  
of billiards*



# Numerical Weather Prediction

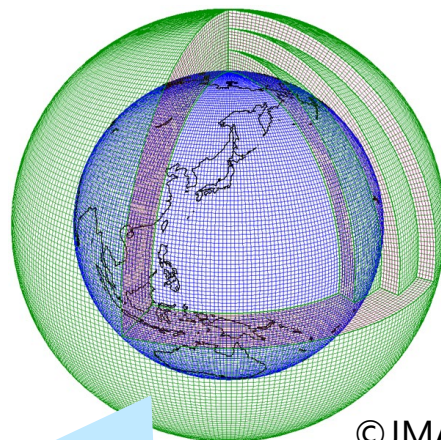
wind, temperature, humidity, pressure

wind, temperature, humidity, pressure



*Real Earth*

*Earth in Computer*



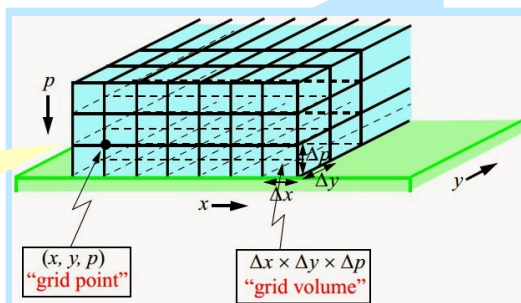
Richardson's Dream (1920; 200km)



Fugaku (2020; 3.5km x 1000 ens)



wind, temperature  
humidity, pressure, ...

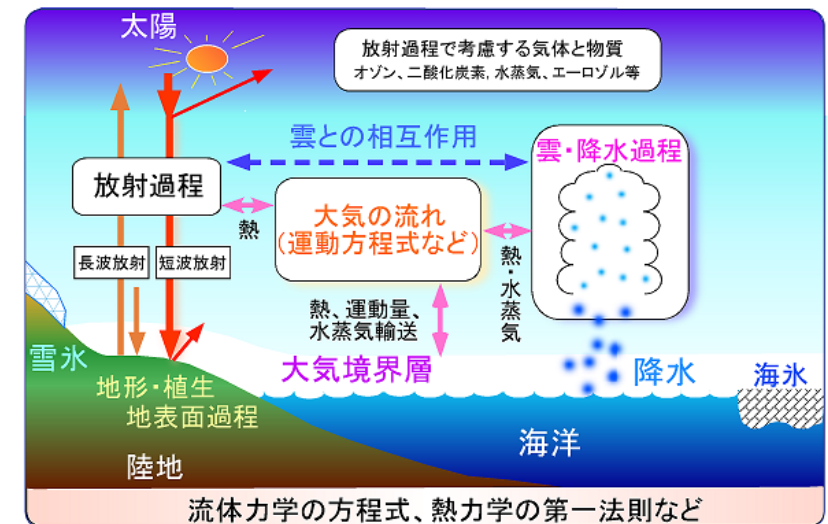


# Physical laws solved in NWP Models

- ▶ **Fluid Dynamics (a.k.a. Dynamical Core)**
  - ▶ Navier-Storks Eq. (流体方程式)
  - ▶ Continuous Eq. (連続の式)
  - ▶ Eqs. of State (状態方程式)
  - ▶ First laws of thermal dynamics (熱力学第一法則)
  
- ▶ **Physics (a.k.a. Physical Process)**
  - ▶ Aerosol and trace gasses (エアロゾル)
  - ▶ Atmospheric radiation (放射)
  - ▶ Cloud and precipitation (雲と雨)
  - ▶ Land & vegetation (陸面&植生)
  - ▶ Ocean (海洋)
  - ▶ Urban (都市)
  - ▶ Etc...

thanks to Prof. Y. Sato

©JMA



# Physical laws solved in NWP Models

## ► Fluid Dynamics (a.k.a. Dynamical Core)

$$\frac{\partial \rho}{\partial t} = -\nabla(\rho \mathbf{v}) \quad (\text{Continuous Eq.})$$

$$\frac{\partial \mathbf{v}}{\partial t} = -\mathbf{v}\nabla\mathbf{v} - \frac{1}{\rho}\nabla p - g\mathbf{k} - f\mathbf{k} \times \mathbf{v} + \mathbf{F} \quad (\text{NS Eq.})$$

$$\frac{\partial \theta}{\partial t} = -\mathbf{v}\nabla\theta + Q \quad \theta = T(p_0/p)^{R/C_p}$$

$$p = \rho RT$$

(Eqs. of State)

$$\frac{\partial q_n}{\partial t} = -\mathbf{v}\nabla q_n + Q_n$$

$\mathbf{F}$ ,  $Q$ , and  $Q_n$  represent effects of physical processes into fluid. (additional terms e.g., Coriolis force is also implemented into  $\mathbf{F}$ ).

$\rho$ : density,  $\mathbf{v}$ : wind,  $p$ : pressure,  $\mathbf{k}$ : unit vector for vertical direction

$T$ : temperature,  $g$ : gravitational acceleration,  $q_n$ : tracers for physical process

# Simulated Global Precipitation



2014/05/25 00:00



# Deterministic Chaos and Predictability

## Edward Lorenz



Lorenz 63 model

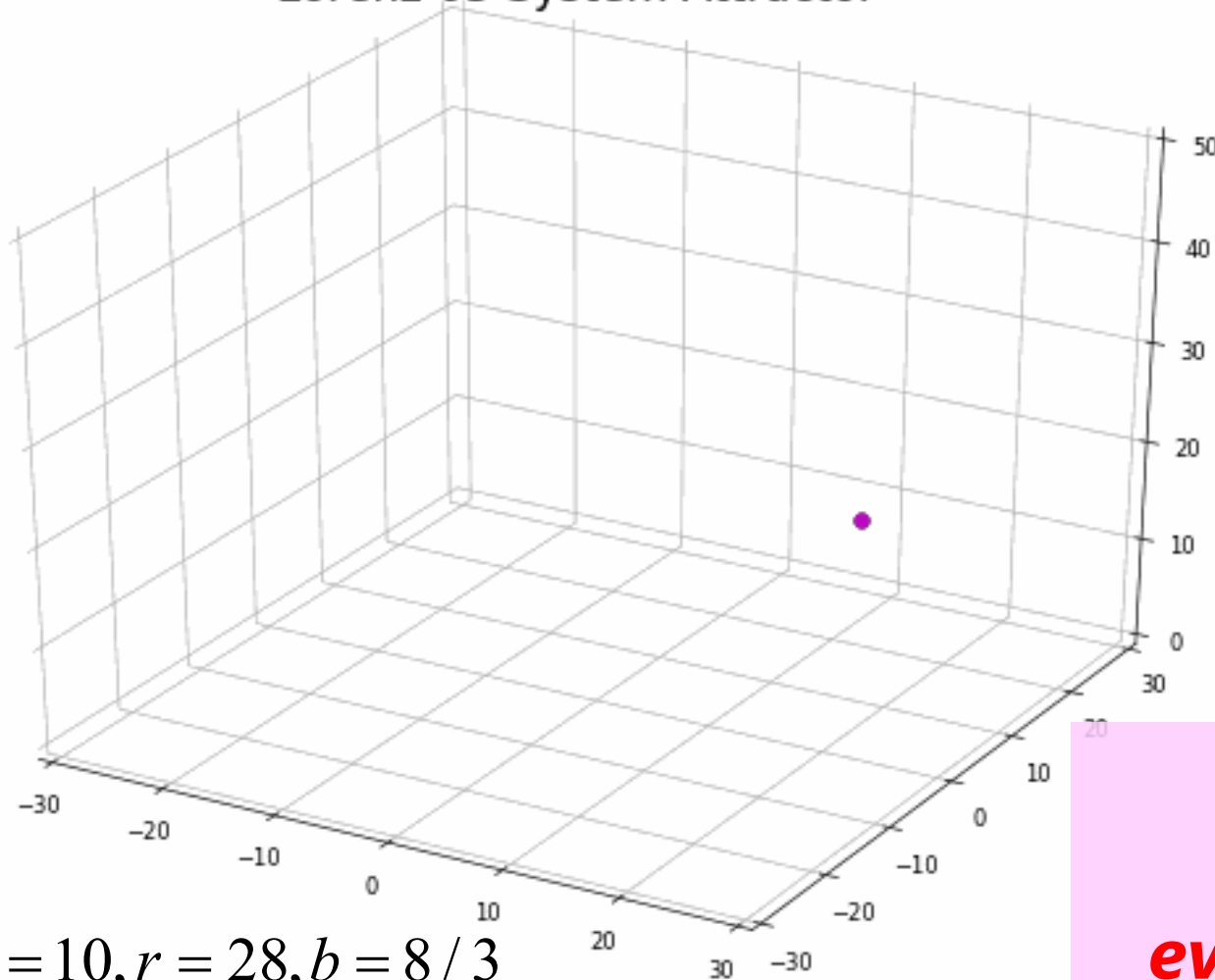
$$\dot{x} = p(y - x)$$

$$\dot{y} = -xz + rx - y$$

$$\dot{z} = xy - bz$$

**Chaotic systems have  
limits to predictability  
even with the perfect model!!**

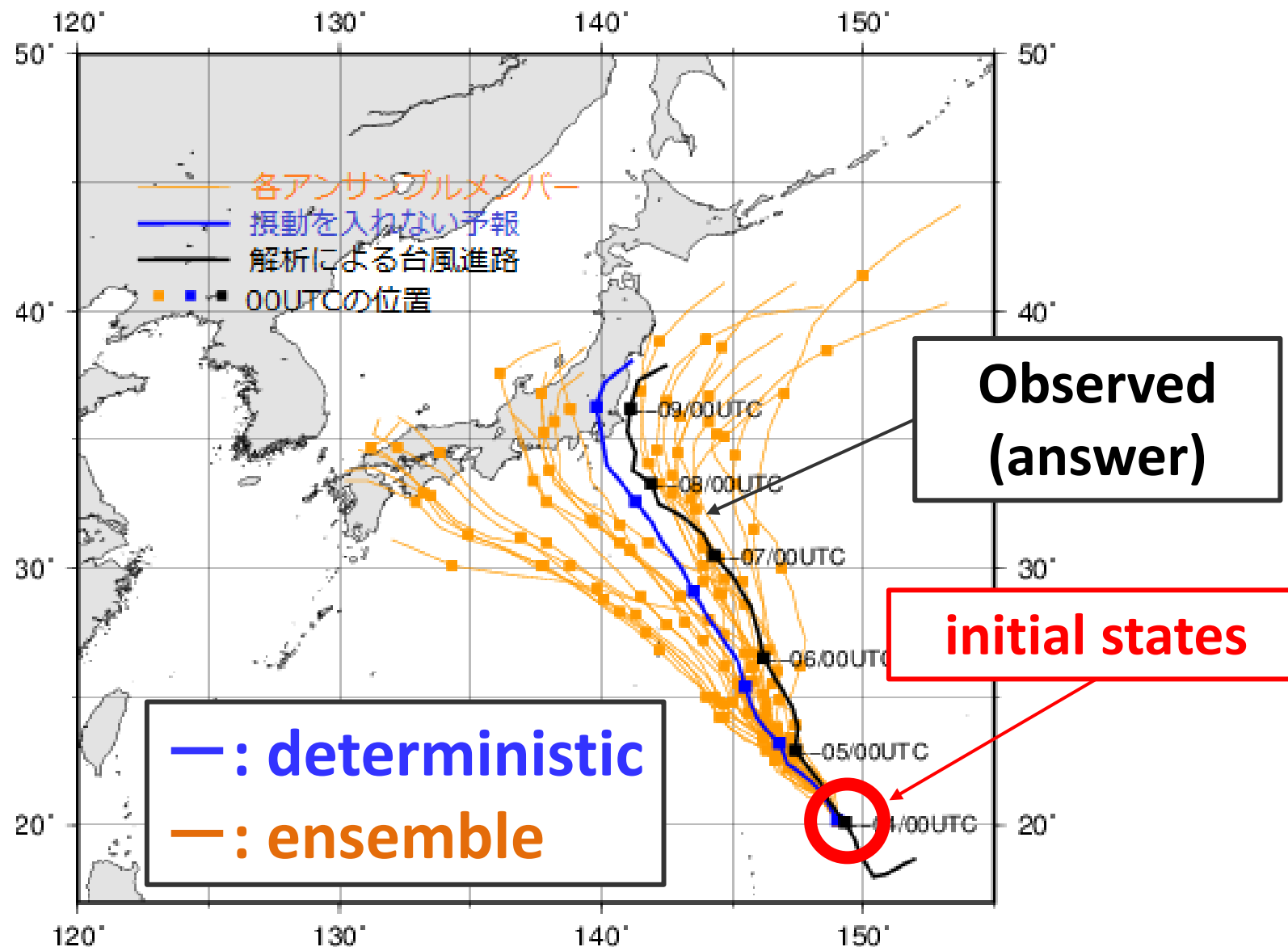
Lorenz 63 System Attractor



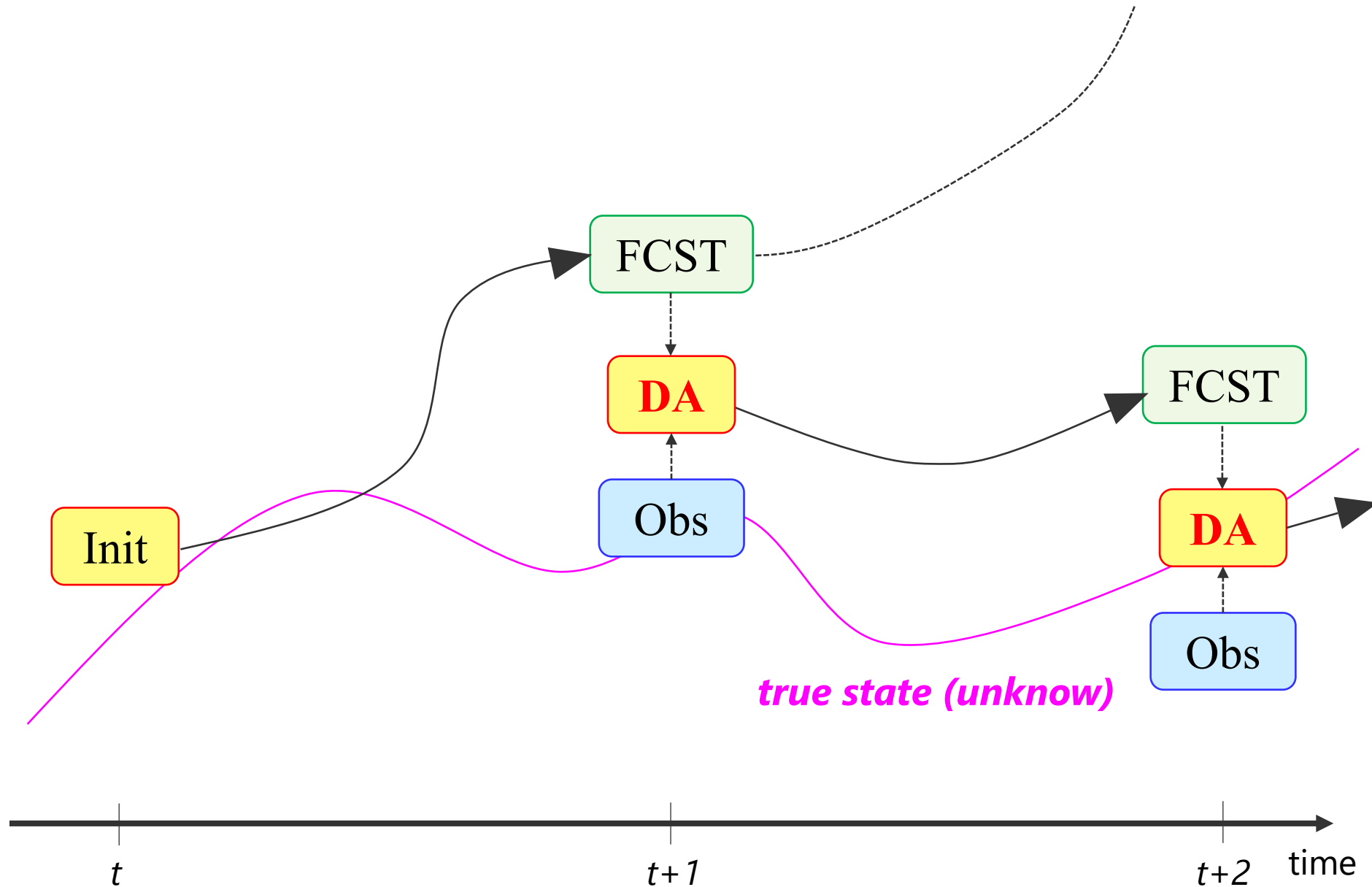
$$p = 10, r = 28, b = 8/3$$

Initial Conditions ::  $x=y=z=15.000, 15.001, 15.002, \dots, 15.009$

# Ensemble Prediction (e.g. TC)

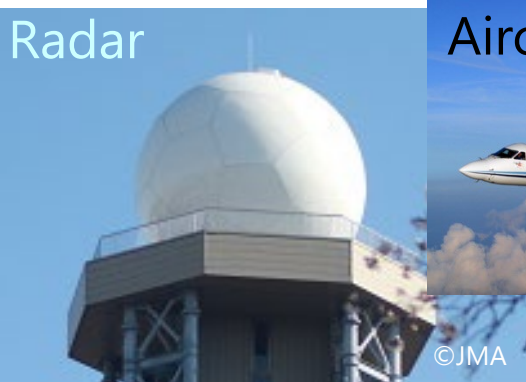


# Numerical Weather Prediction

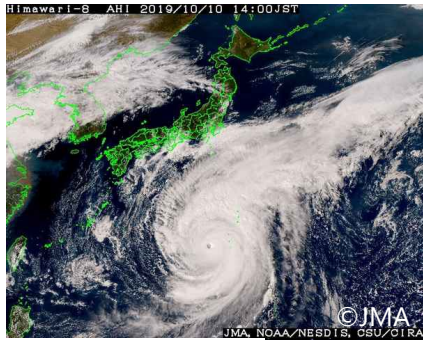


# Global Observing System

Radar



Aircraft



Satellite



Radiosonde



Ship



Surface station

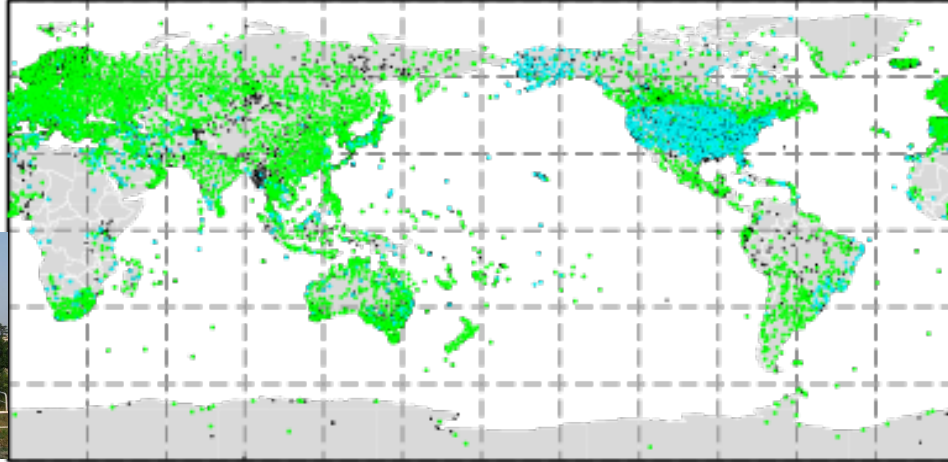
Buoy



©wiki

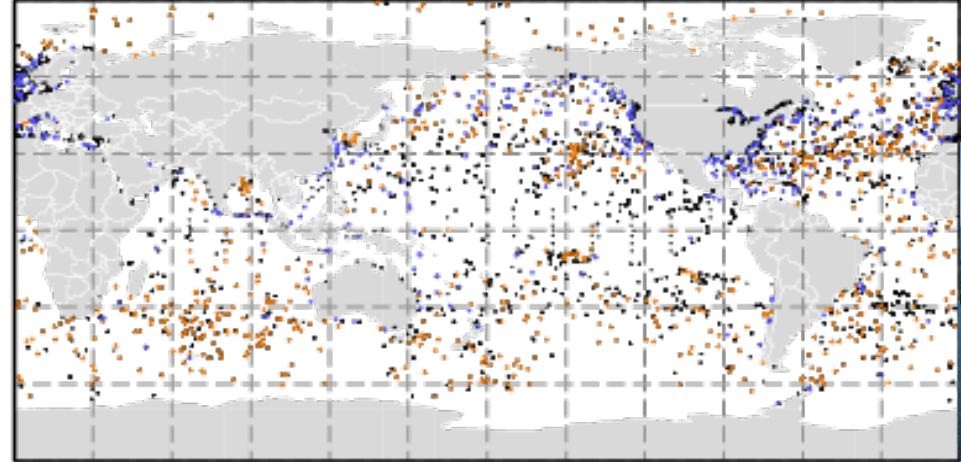
# Observation Data in NWP

LAND SURFACE 2019/05/01 00:00(UTC)



SYNOP[●]: 4163 METAR[●]: 1178  
 NOUSE[●]: 18901 NOUSE[●]: 42966  
 ALL: 23054 ALL: 44144

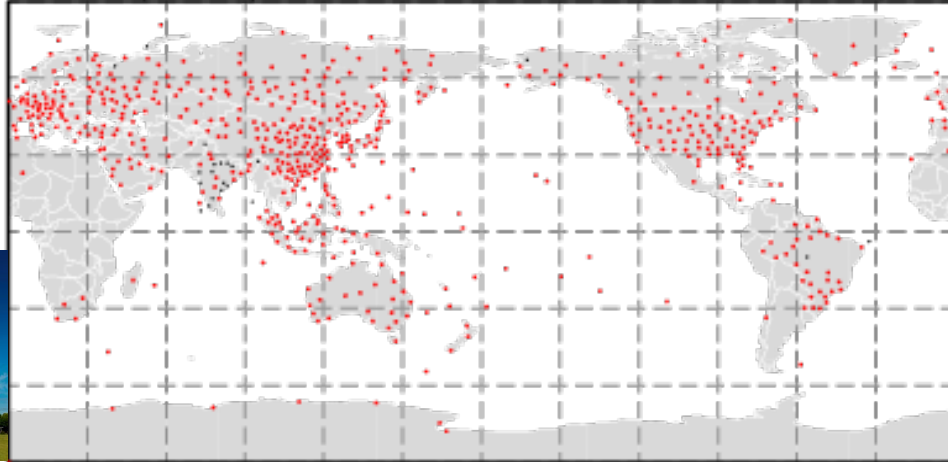
SEA SURFACE 2019/05/01 00:00(UTC)



SHIP[●]: 378 DRIFTER[●]: 709  
 NOUSE[●]: 5668 NOUSE[●]: 10206  
 ALL: 6046 ALL: 10915

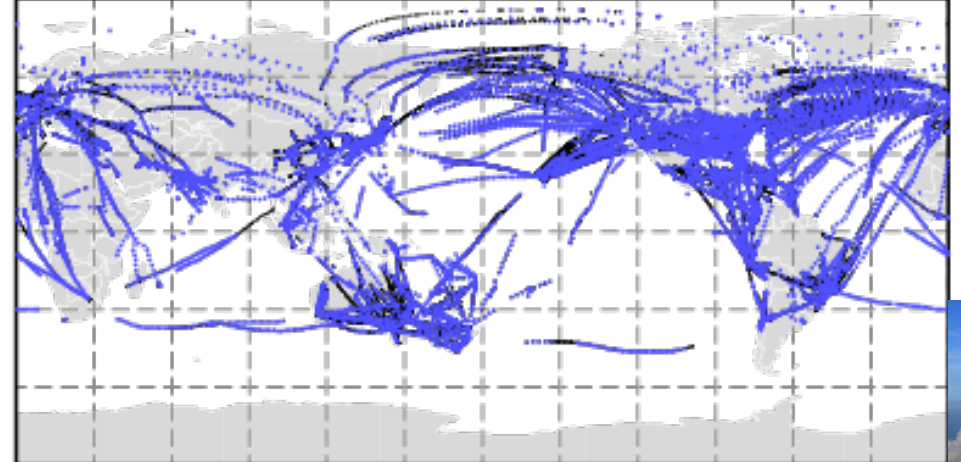


UPPER(TEMP) 2019/05/01 00:00(UTC)

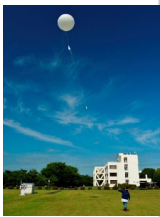


TEMP[●]: 632  
 NOUSE[●]: 24  
 ALL: 656

UPPER(AVIATION)/BOGUS 2019/05/01 00:00(UTC)



TYBOGUS[●]: 0 YHTC AVIATION[●]: 9919  
 NOUSE[●]: 0 NOUSE[▼]: 0 NOUSE[●]: 88767  
 ALL: 0 ALL: 0 ALL: 98686

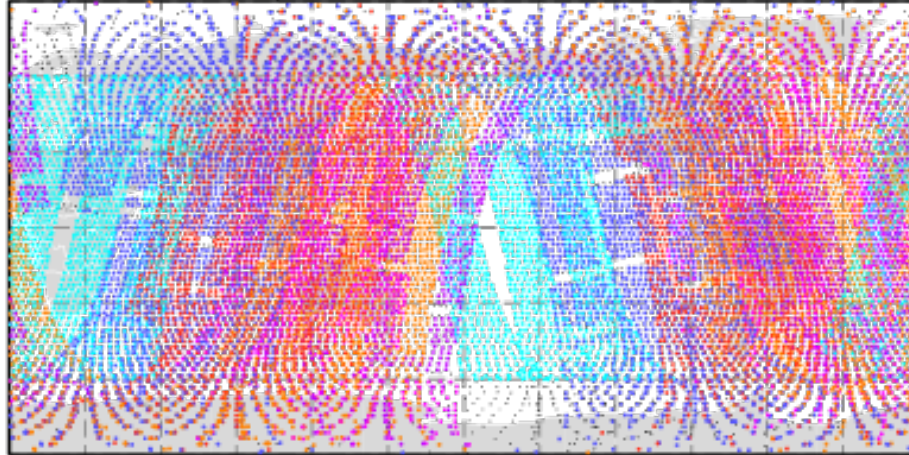


courtesy of JMA (2019/05/01 00:00 UTC)

# Satellite Data in NWP

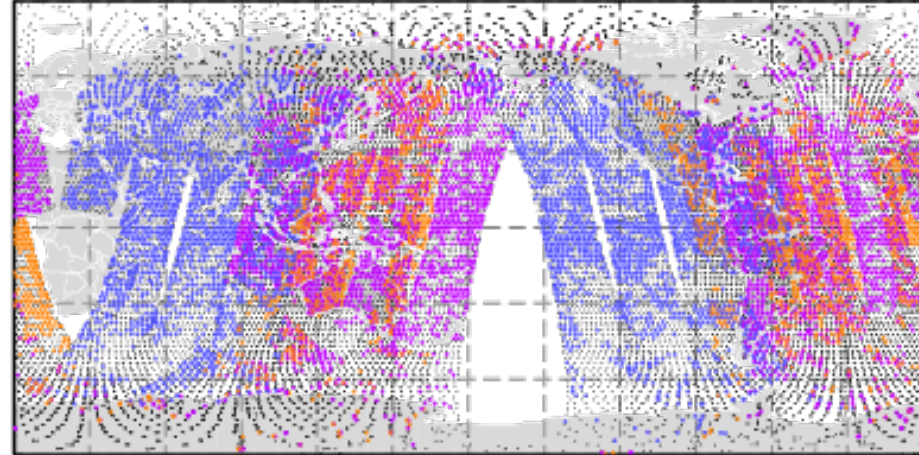


MW-SOUNDER(AMSU-A) 2019/05/01 00:00(UTC)



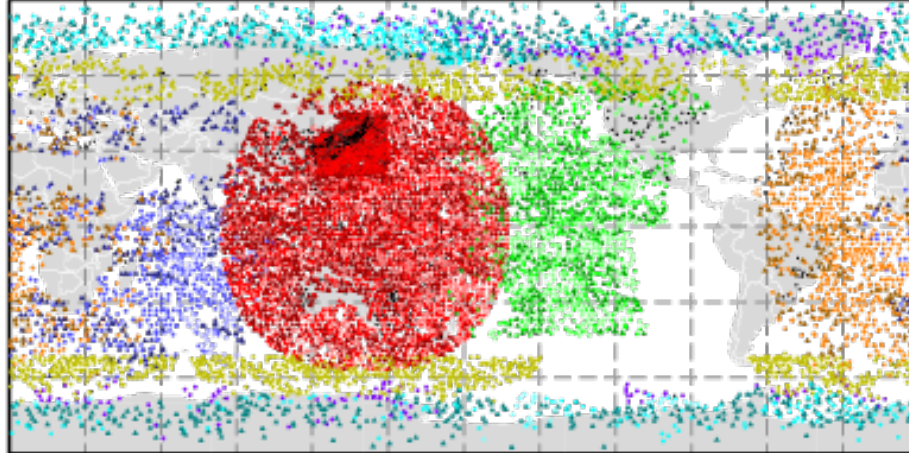
NOAA-15	NOAA-18	NOAA-19	Aqua	Metop-A	Metop-B
AMSU-A[●]: 3606	AMSU-A[●]: 1625	AMSU-A[●]: 4835	AMSU-A[●]: 3160	AMSU-A[●]: 4881	AMSU-A[●]: 2806
NOUSE[●]: 58	NOUSE[●]: 74	NOUSE[●]: 487	NOUSE[●]: 477	NOUSE[●]: 59	NOUSE[●]: 181
ALL: 3664	ALL: 1699	ALL: 5322	ALL: 3637	ALL: 4940	ALL: 2987

MW-SOUNDER(MHS,SAPHIR,MWHS) 2019/05/01 00:00(UTC)



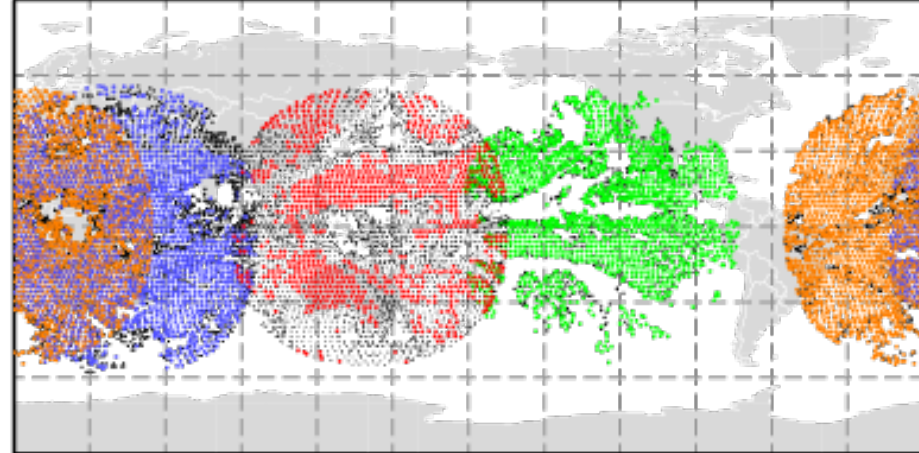
NOAA-19	Metop-A	Metop-B
MHS[●]: 3864	MHS[●]: 3162	MHS[●]: 3881
NOUSE[●]: 4854	NOUSE[●]: 3693	NOUSE[●]: 5843
ALL: 8718	ALL: 6155	ALL: 9724

ATMOSPHERIC MOTION VECTOR 2019/05/01 00:00(UTC)



Himawari-8	GOES-15	Meteosat-8	Meteosat-11	MODIS	LEO GEO	AVHRR
IR[●]: 1729	IR[●]: 429	IR[●]: 440	IR[●]: 757	IR[●]: 572	IR[●]: 1719	IR[●]: 313
VIS[☐]: 1160	VIS[☐]: 484	VIS[☐]: 124	VIS[☐]: 5	WV[▲]: 773	WV[▲]: 107	
WV[▲]: 1875	WV[▲]: 513	WV[▲]: 314				
SFC[■]: 2518						
NOUSE[●]: 5981	NOUSE[●]: 133	NOUSE[●]: 83	NOUSE[●]: 132	NOUSE[●]: 49	NOUSE[●]: 146	NOUSE[●]: 36
ALL: 13263	ALL: 1539	ALL: 961	ALL: 1266	ALL: 1501	ALL: 1865	ALL: 349

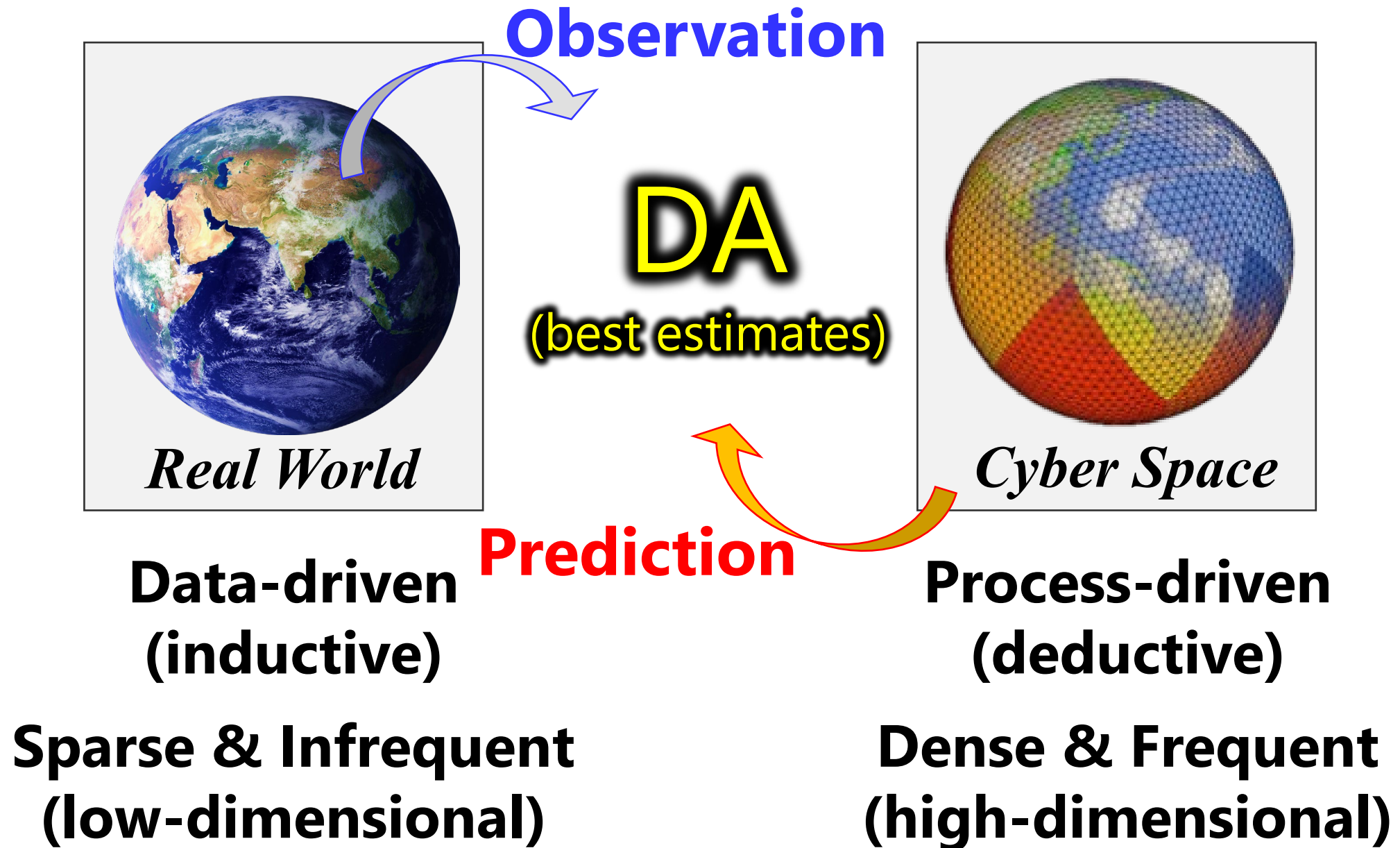
CLEAR SKY RADIANCE 2019/05/01 00:00(UTC)



Himawari-8	GOES-15	Meteosat-8	Meteosat-11
AH[●]: 2812	IMAGER[●]: 2257	SEVIR[●]: 4094	SEVIR[●]: 5865
NOUSE[●]: 8031	NOUSE[●]: 2109	NOUSE[●]: 5723	NOUSE[●]: 4329
ALL: 10843	ALL: 4366	ALL: 9817	ALL: 10184

courtesy of JMA (2019/05/01 00:00 UTC)

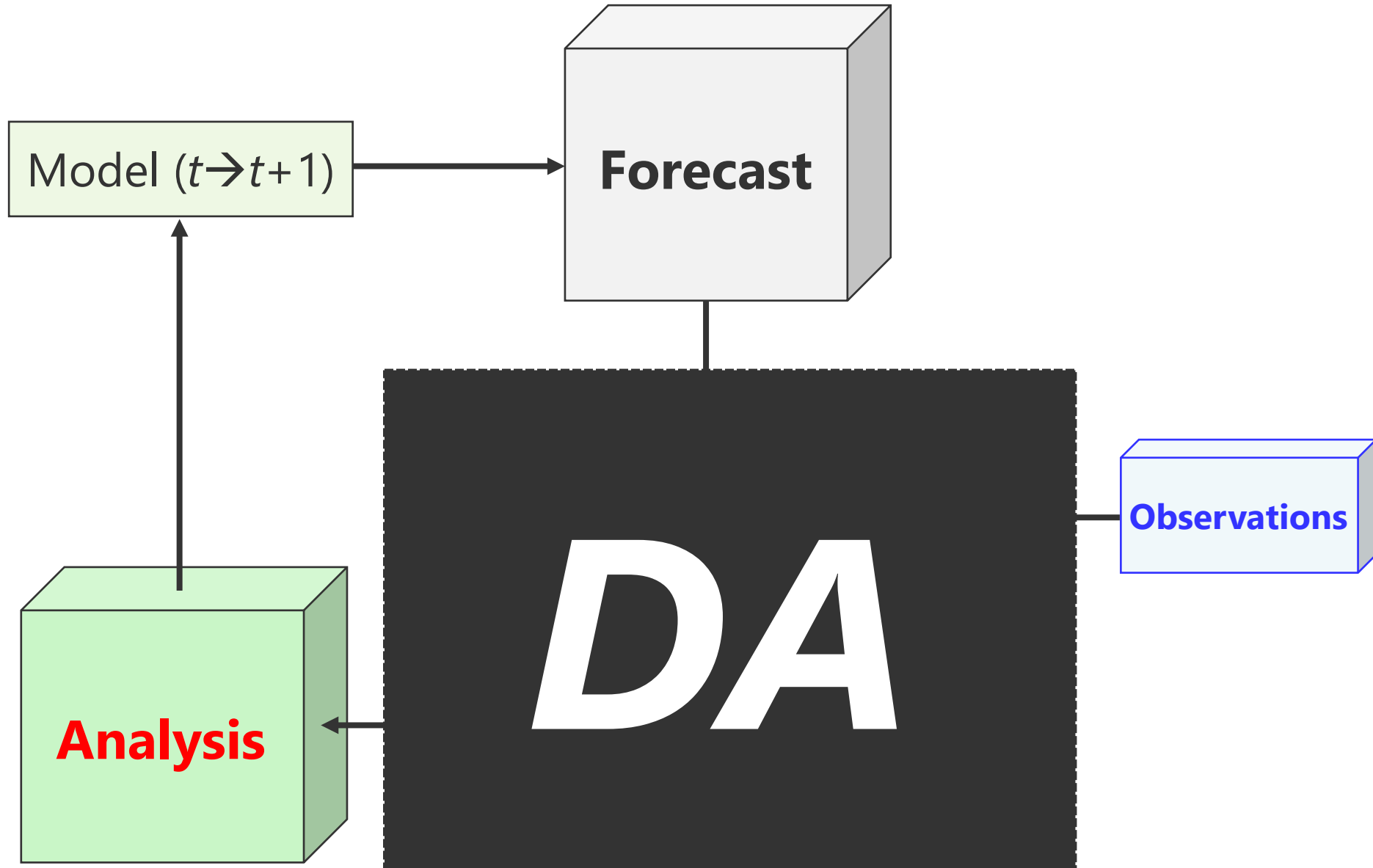
# Data Assimilation



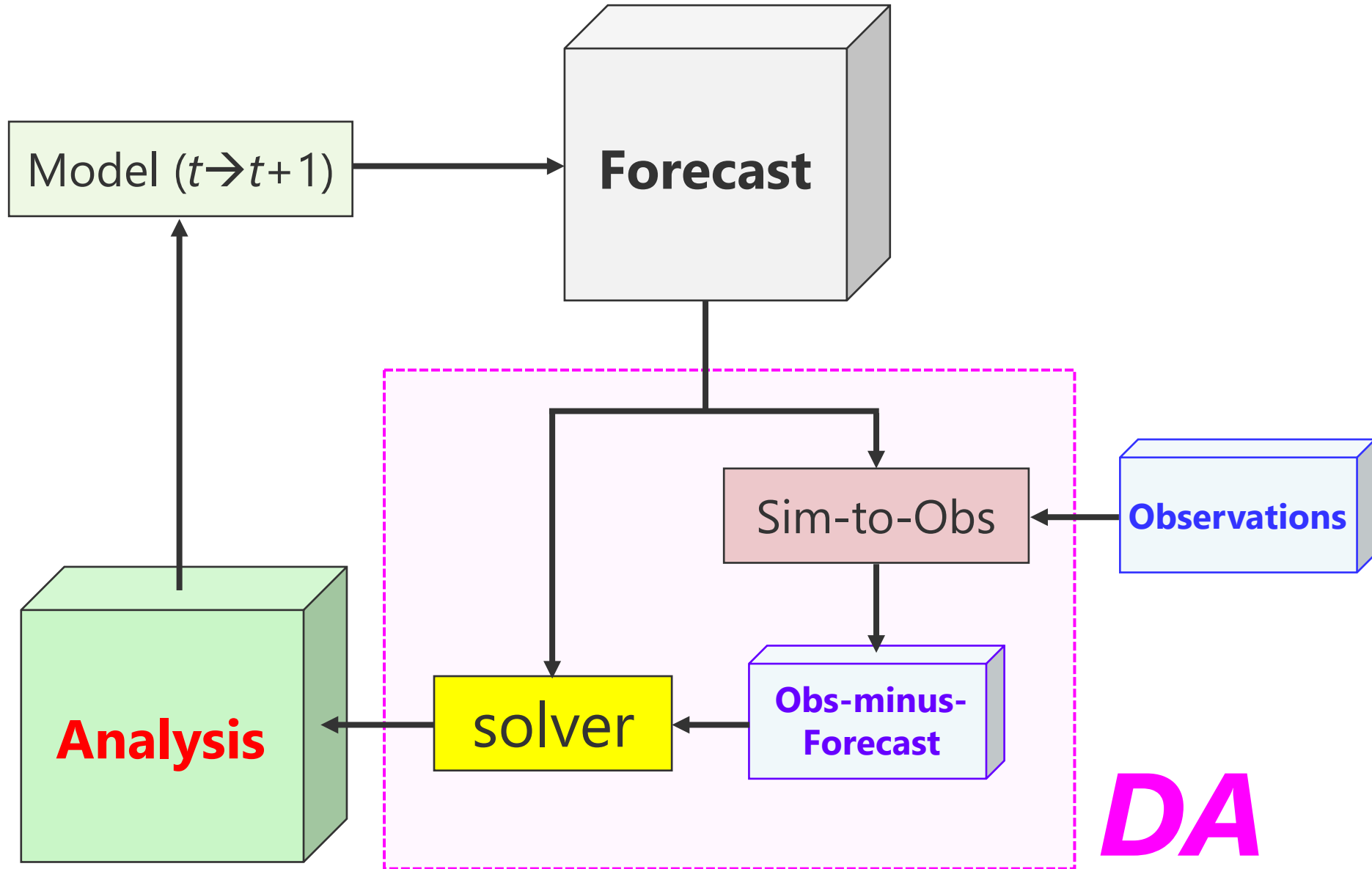
# What DA can Do?



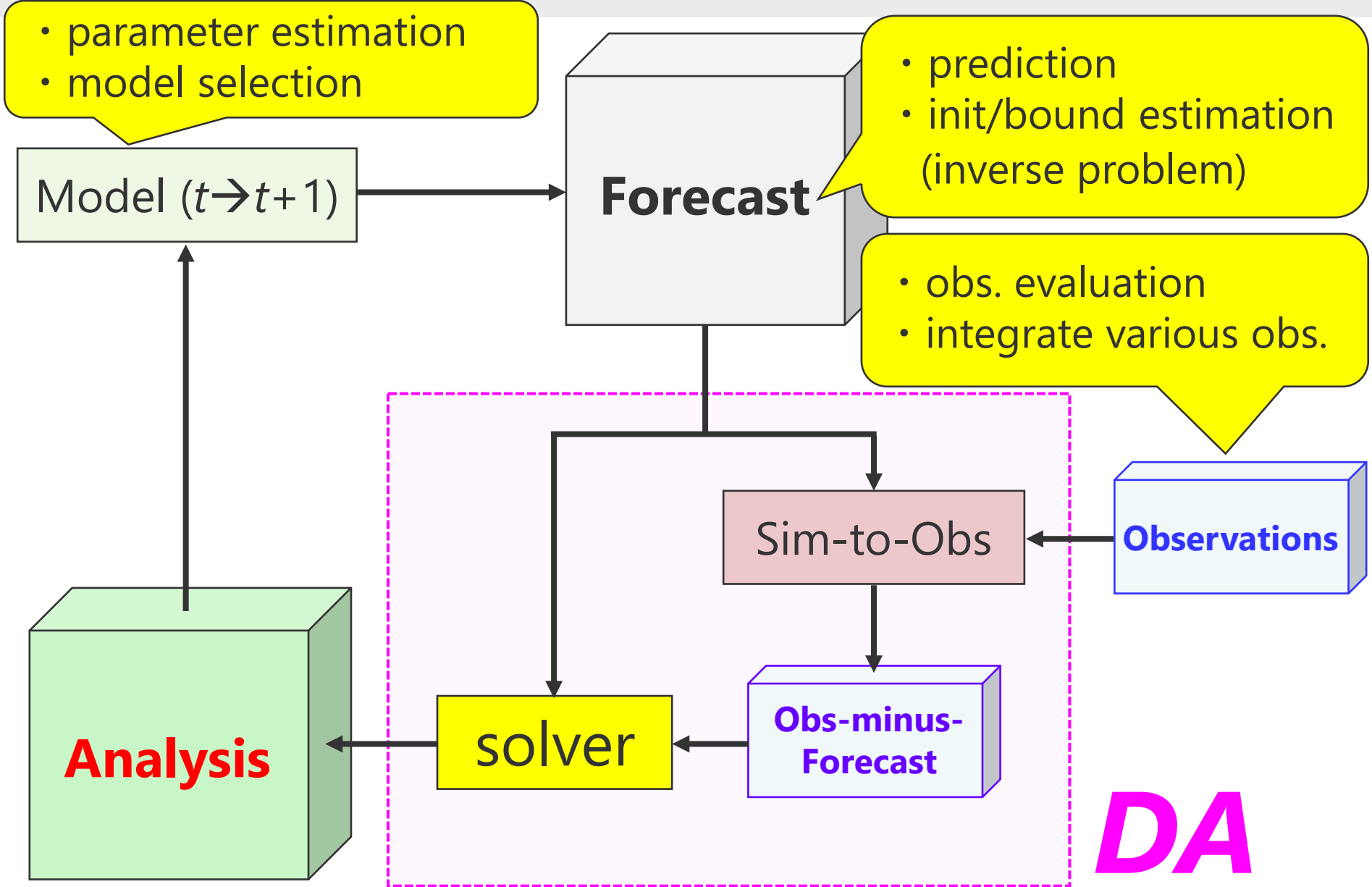
# Workflow of DA



# Workflow of DA



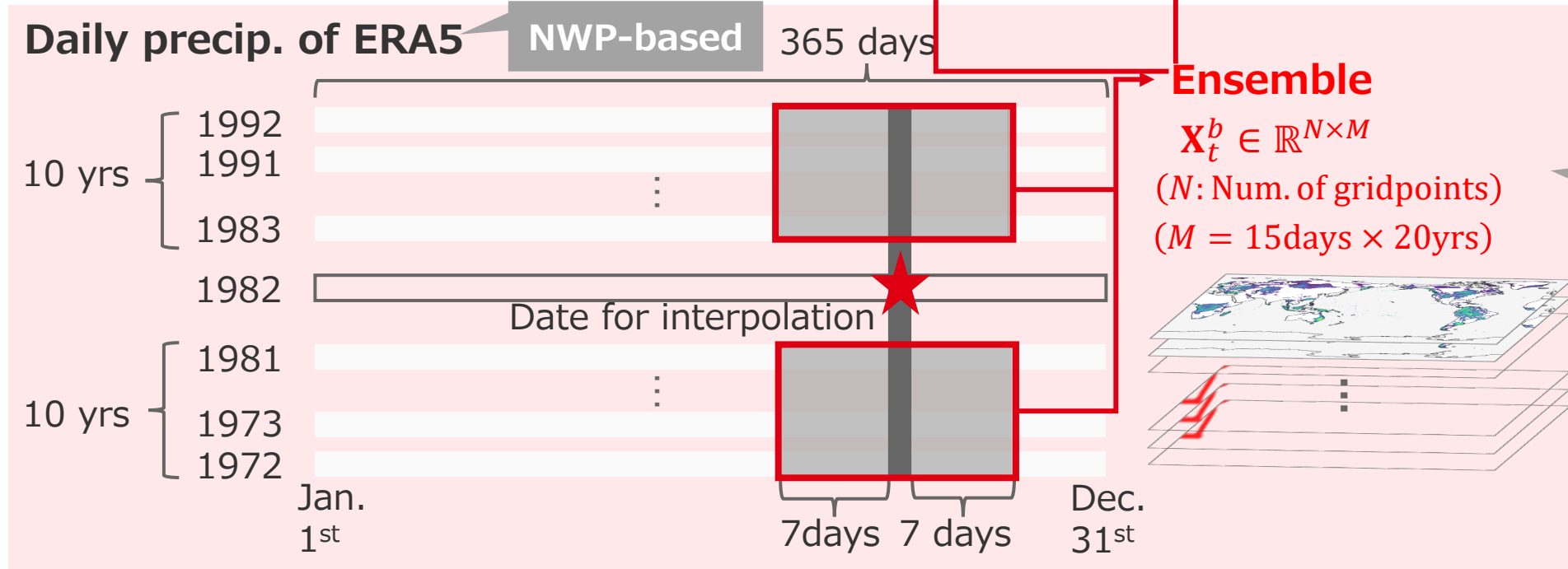
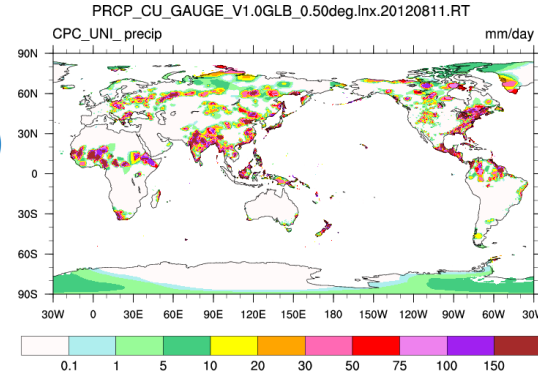
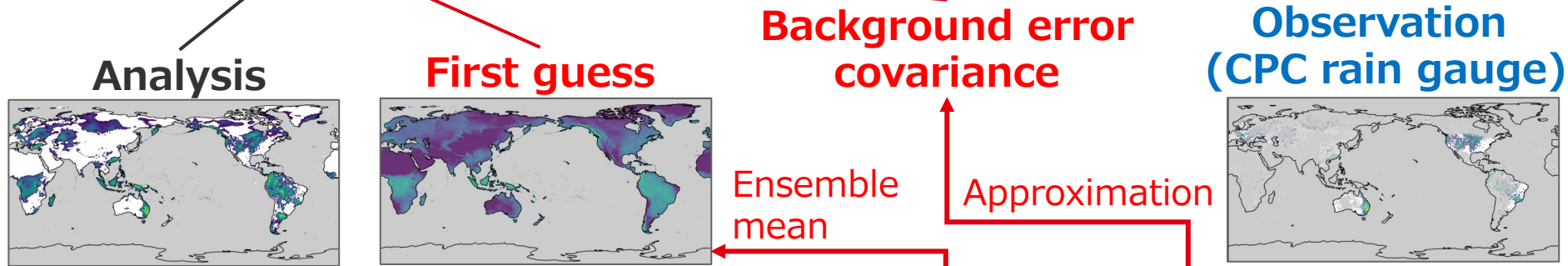
# Workflow of DA



# Interpolation method using the LETKF

Interpolate the global precipitation (P) field based on the LETKF

$$\mathbf{x}_t^a = \mathbf{x}_t^b + \mathbf{P}_t^b \mathbf{H}_t^T [\mathbf{H}_t \mathbf{P}_t^b \mathbf{H}_t^T + \mathbf{R}_t]^{-1} (\mathbf{y}_t^o - H(\mathbf{x}_t^b))$$



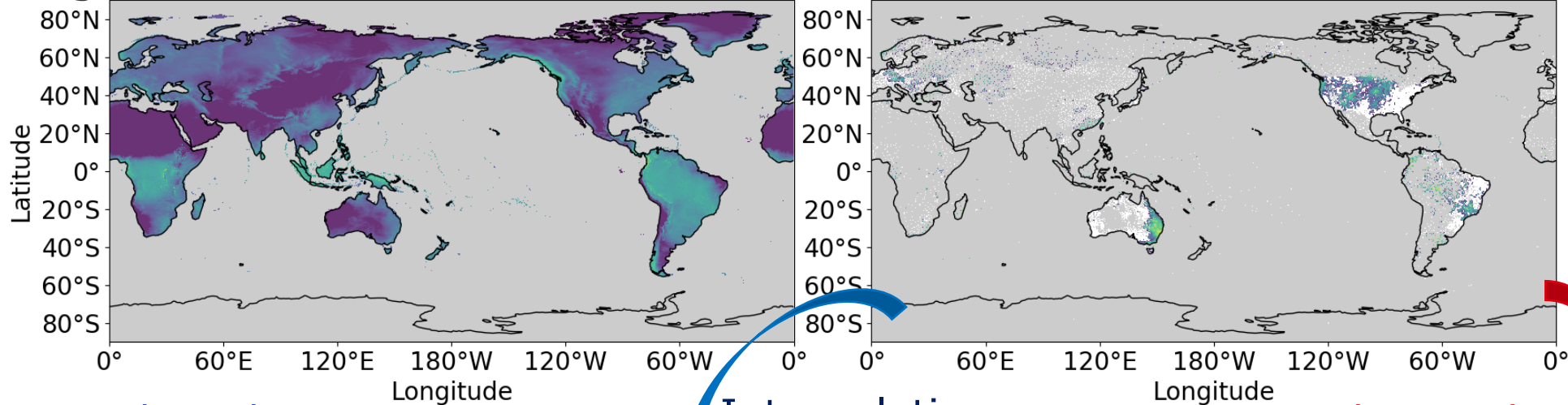
# Results: Examples of daily precipitation fields

- 1988/11/15

First guess

(a) Background (from ERA5)

(b) CPC\_gauge Observation input

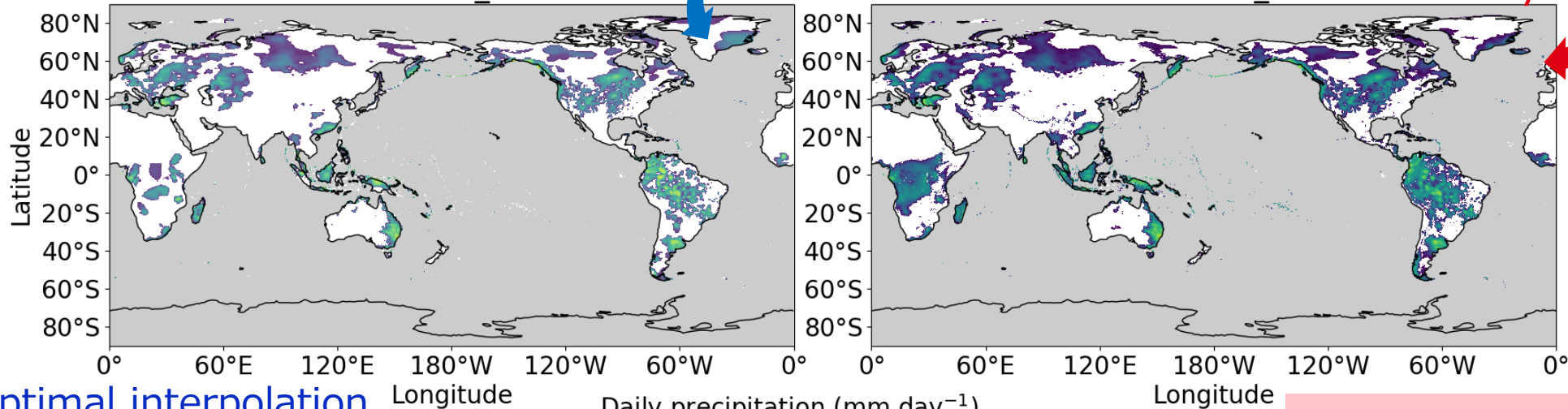


Interpolation (DA)

Based on the OI (c) CPC\_est

Interpolation

(d) LETKF\_est This study



OI: optimal interpolation

Daily precipitation (mm day<sup>-1</sup>)



Successfully estimated

# AI-based Weather Prediction w/ DA

AI model (ViT)  
(Cyber Space)

Obs  
(Real World)

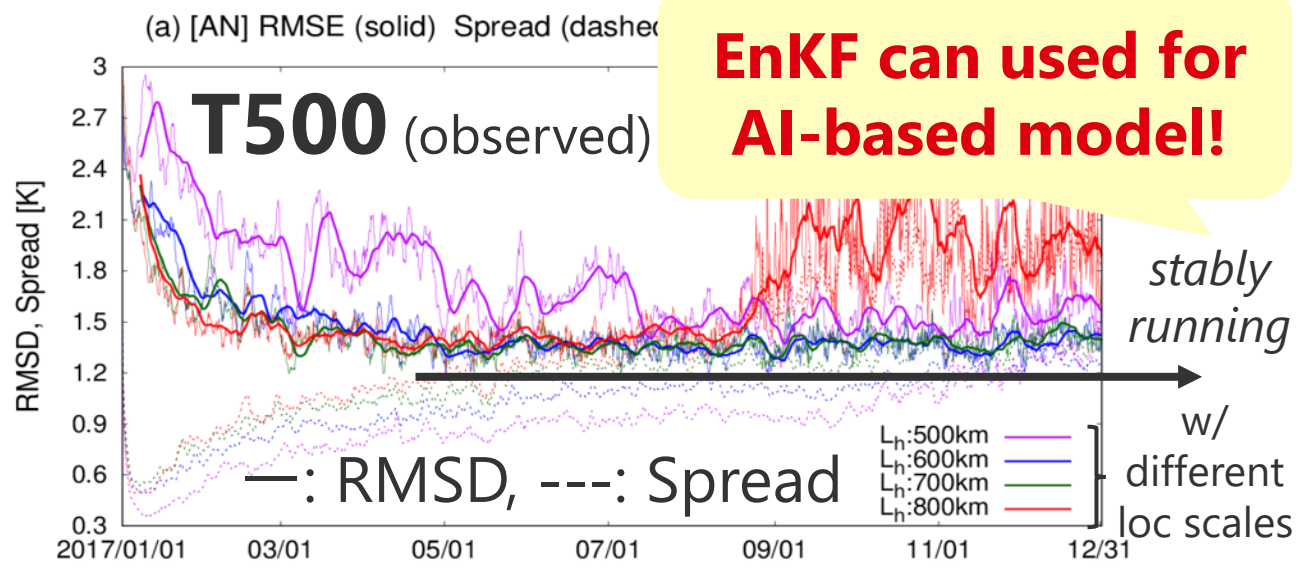


## ● ClimaX-LETKF Experiments

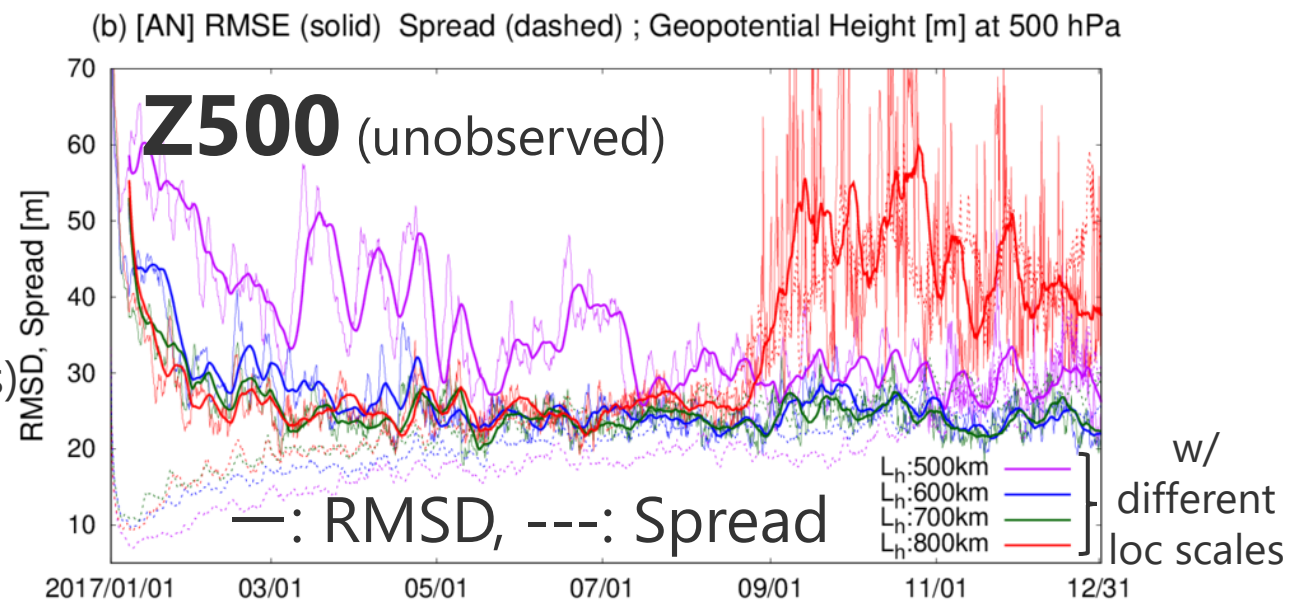
- low-res. ClimaX (w/ modifications)
- 20-member LETKF
- Obs: psuedo radiosondes ( $T, U, V, Q, P_s$ )

## ● Verification (RMSE)

- against Weather Bench (ERA5)



**EnKF can be used for AI-based model!**

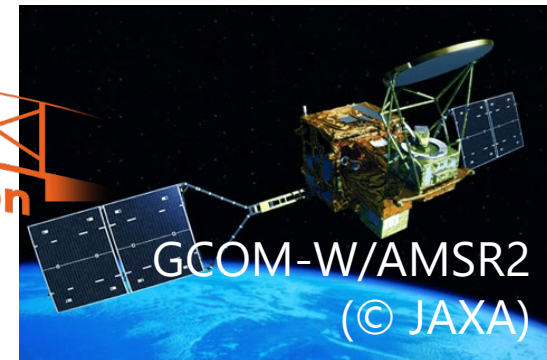


# Model Parameter Estimation

## Berry (1967)'s LSC scheme

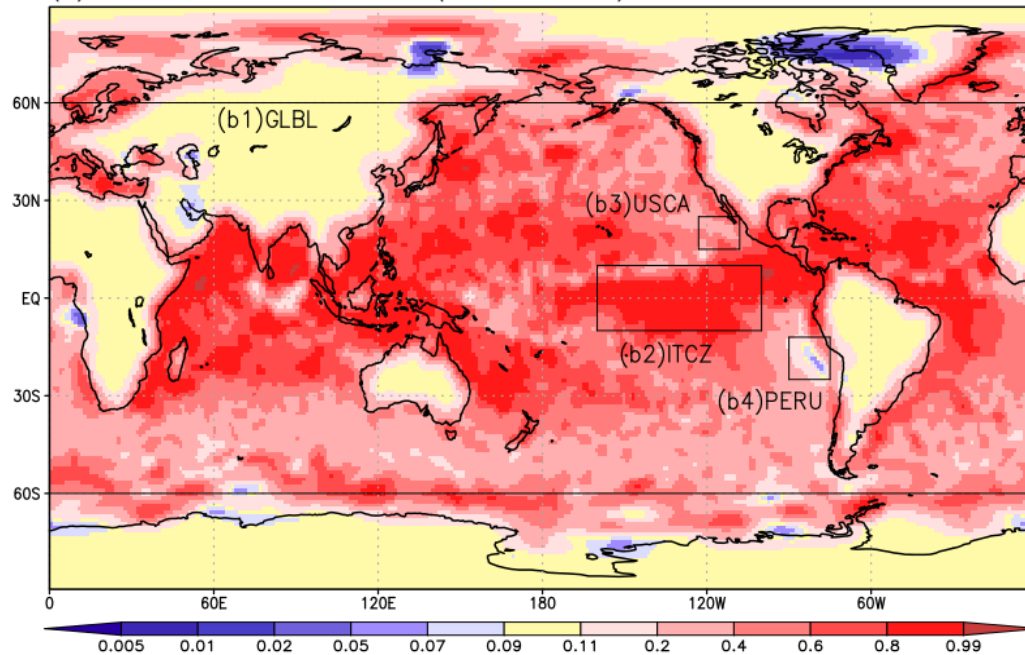
$$P = \frac{B1\rho l^2}{B2 + B3 \frac{N_c}{\rho l}}$$

$\rho$  : air density  
 $P$ : precipitation rate  
 $l$  : cloud water mixing ratio  
 $N_c$ : total # of cloud droplet



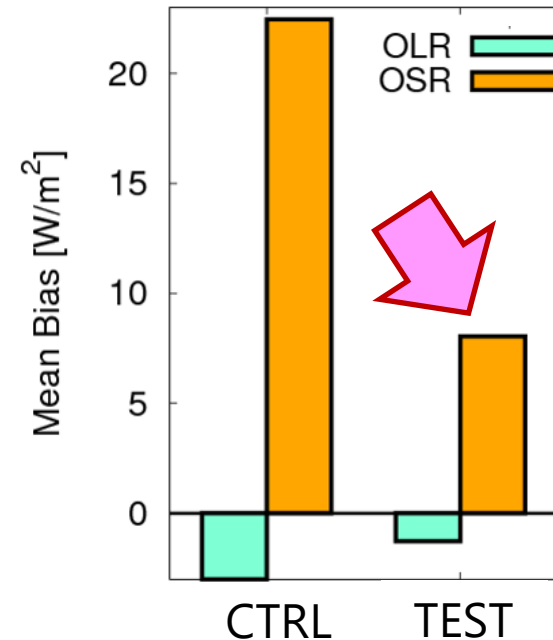
## Parameter Fields estimated by LETKF

(a) Estimated B1 Parameter (LWP-L200km) Period: 2015010100 - 2015123118



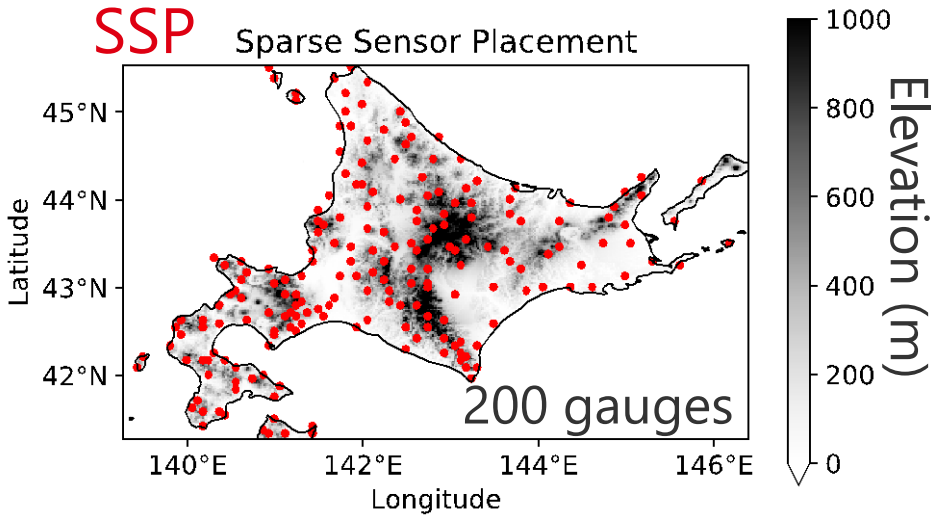
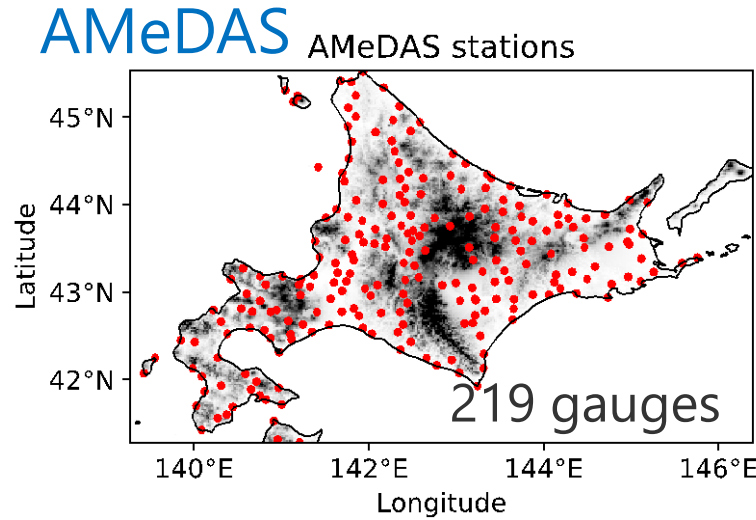
w/ 112-km NICAM-LETKF

## Radiational bias mitigated (vs. CERES)



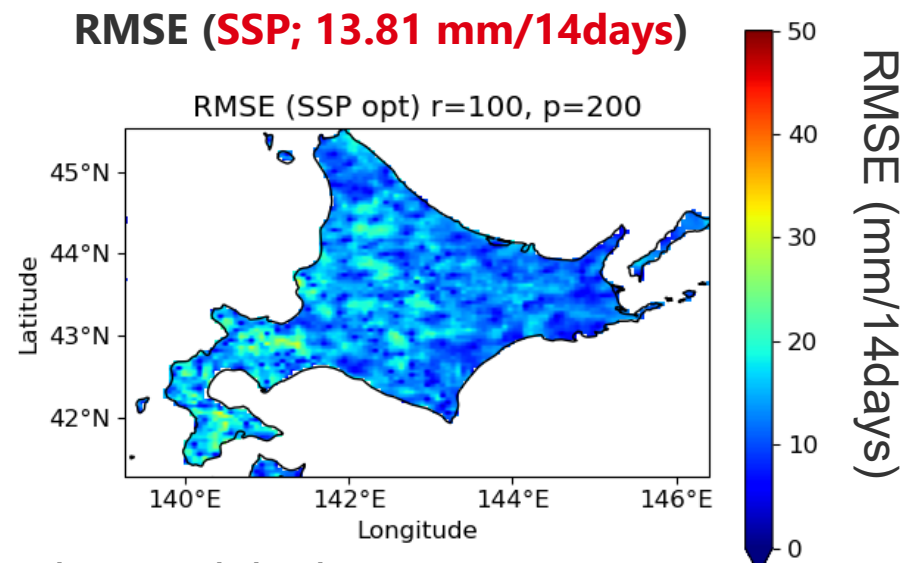
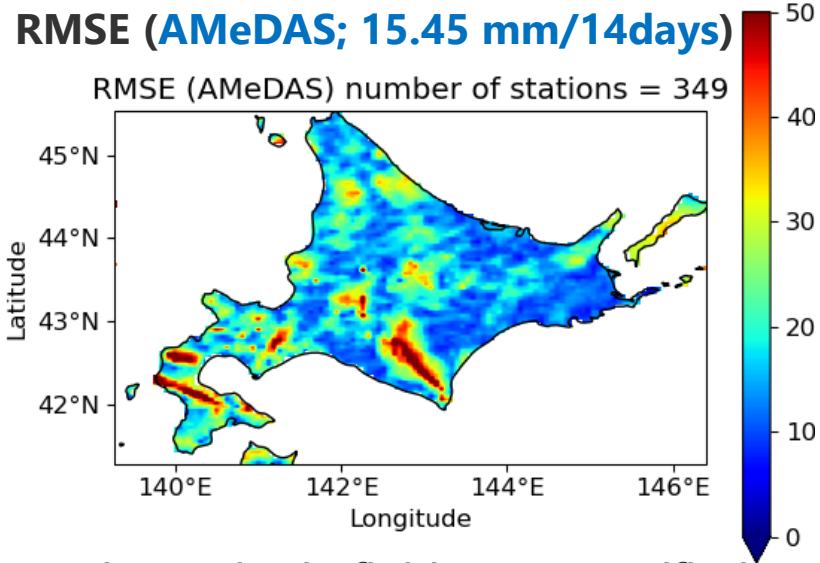
Kotsuki et al. (2018 JGR; 2020 JGR)

# Sparse Sensor Placement: for improving B/C of obs



to estimate rain fields by LETKF

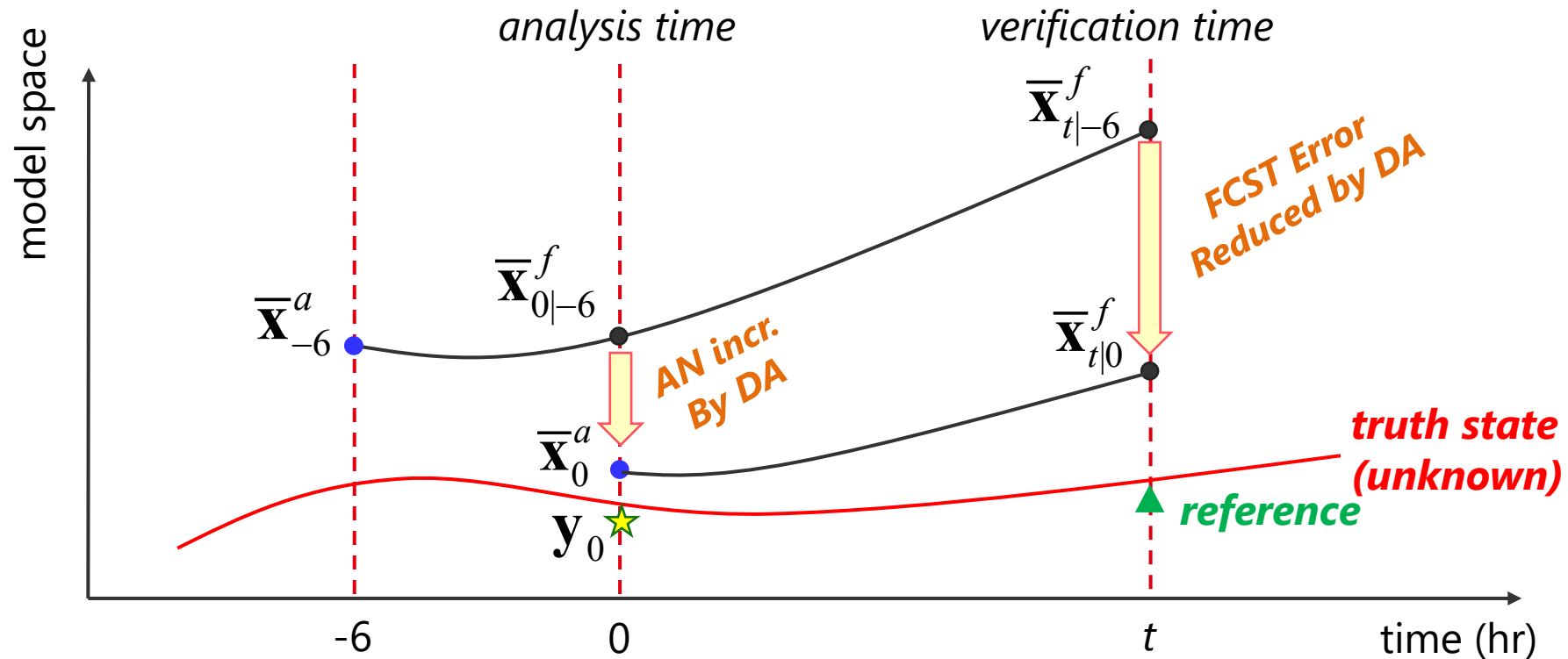
to estimate rain fields by LETKF



estimated rain fields were verified against radar precipitation [Shiojiri and Kotsuki, in prep.]



# Forecast Sensitivity to Observations



## Moist Total Energy for Error Norm

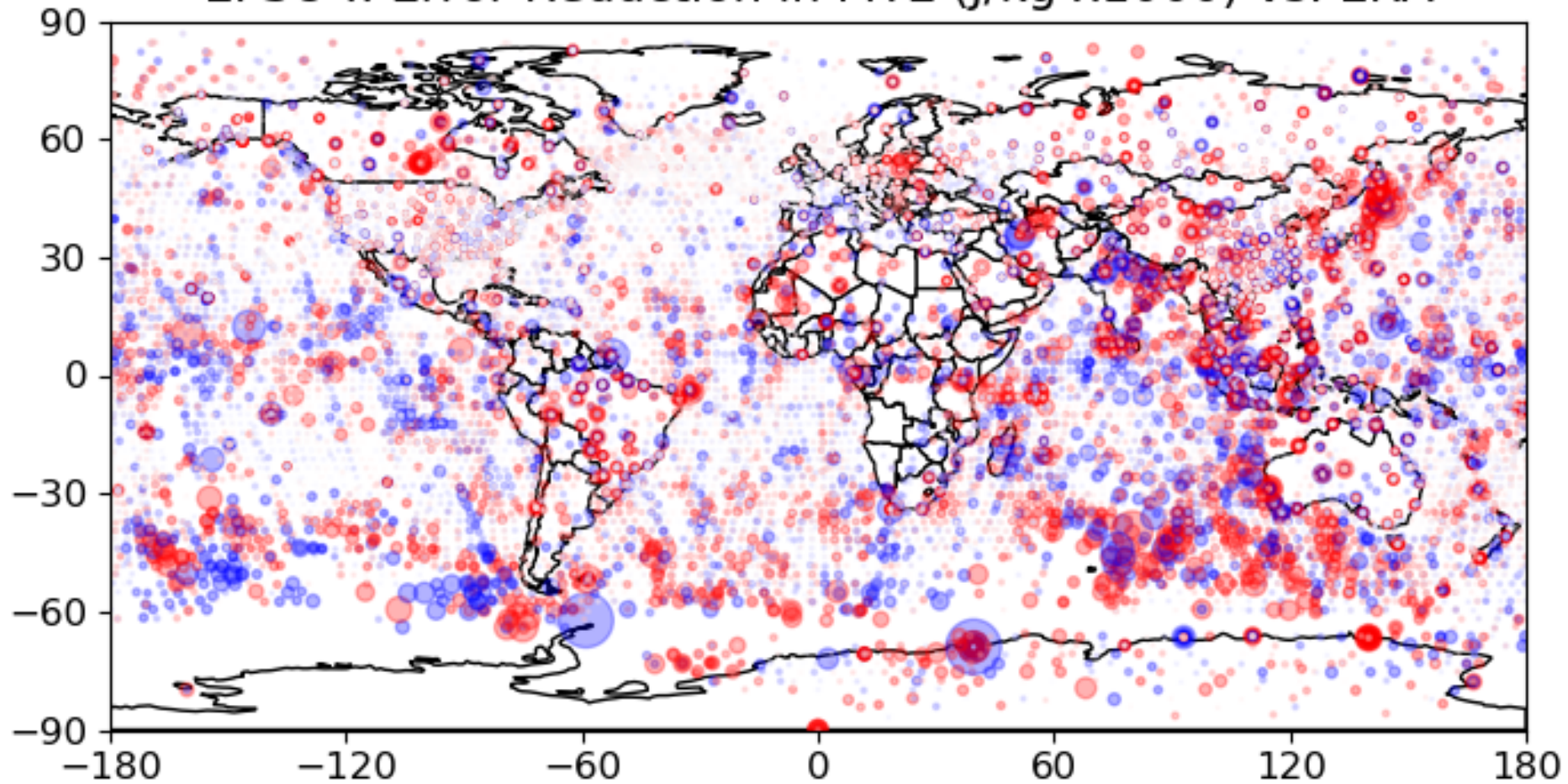
$$\Delta e_{MTE}^2 = \left( \mathbf{e}_{t|0}^T \mathbf{C} \mathbf{e}_{t|0} - \mathbf{e}_{t|-6}^T \mathbf{C} \mathbf{e}_{t|-6} \right) / 2 \quad \mathbf{e}_t = \bar{\mathbf{x}}_t^f - \mathbf{x}_t^{ERA,ANL}$$

$$\approx \frac{1}{2} \frac{1}{m-1} \delta \mathbf{y}_0^T \mathbf{R}^{-1} \mathbf{Y}_0^a \mathbf{X}_{t|0}^{fT} \mathbf{C} \left( \mathbf{e}_{t|0} + \mathbf{e}_{t|-6} \right)$$

EFSO can evaluate how each obs improves/degrades forecasts

# Evaluating Values of Observations: FSO

EFSO :: Error Reduction in MTE (J/kg x1000) vs. ERA



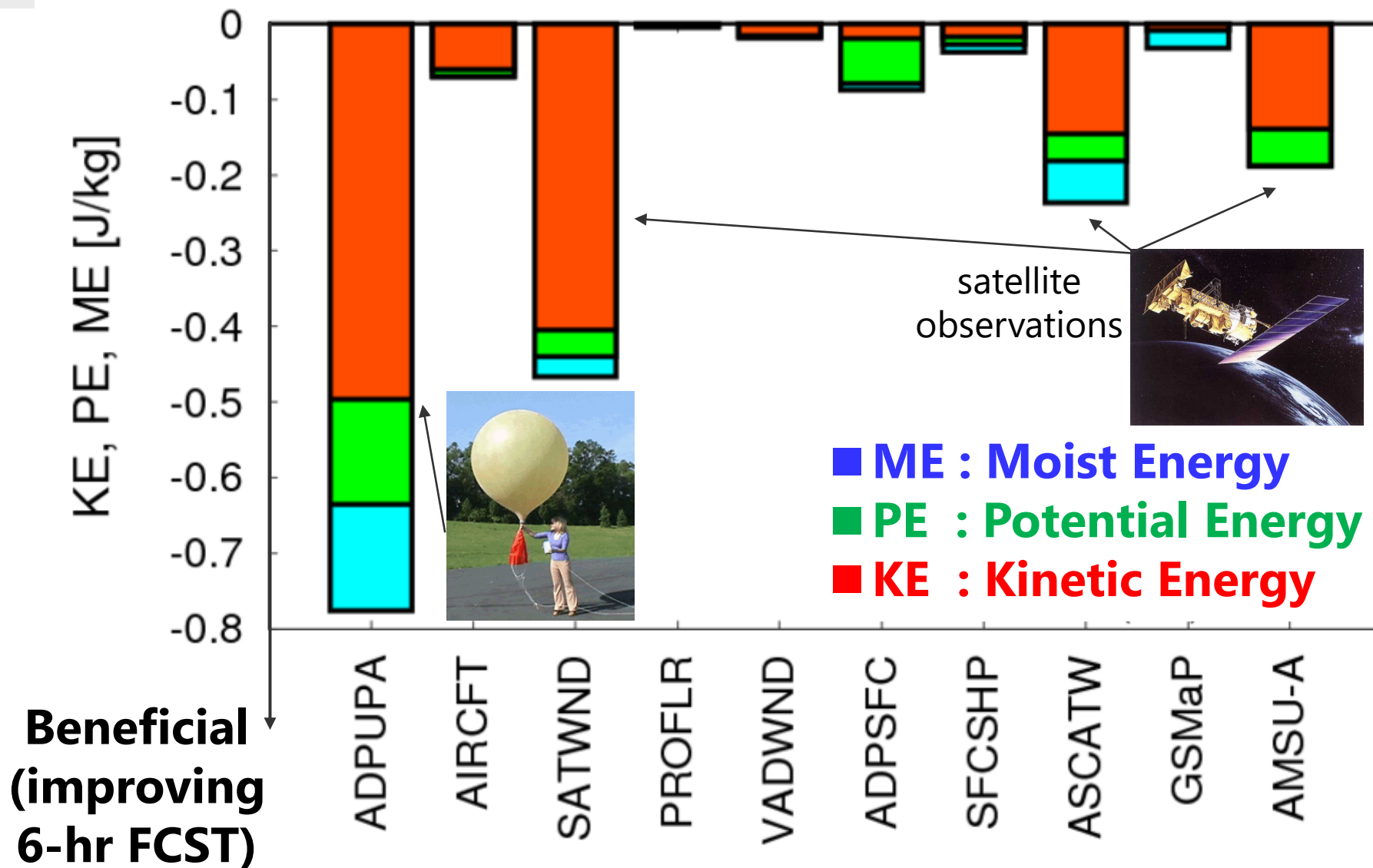
● **Beneficial observations**

● **Detrimental observations**

2014/07/11/00UTC; vs. ERA interim

**Kotsuki et al. (2019; QJRMS)**

# Evaluating Values of Observations: FSO

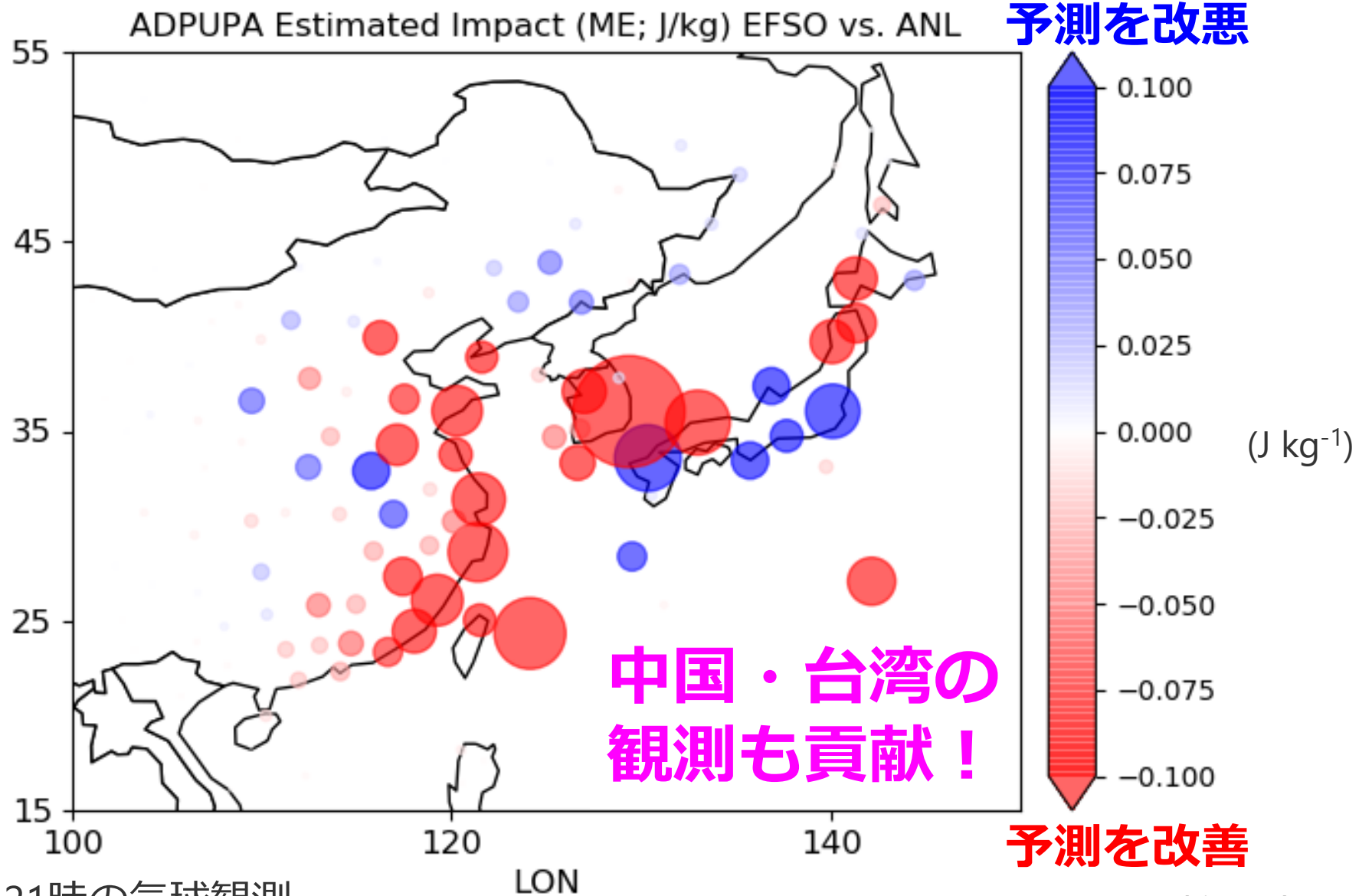


average in July 2014; FT:06hr; vs. ERA

**Kotsuki et al. (2019; QJRMS)**

# 平成30年7月豪雨, 重要な観測点

気球観測



2018年7月3日21時の気球観測

Kotsuki et al. (2019, SOLA)

# Summary

# Today's Goal

- ▶ To understand numerical weather prediction and the role of the data assimilation
- ▶ To be interested in data assimilation

**Thank you for your attention!**

**Presented by Shunji Kotsuki**  
([shunji.kotsuki@chiba-u.jp](mailto:shunji.kotsuki@chiba-u.jp))

**Further information is available at**  
<https://kotsuki-lab.com/>

