

Data Assimilation

- A01. Introduction -

Shunji Kotsuki

Environmental Prediction Science Laboratory
Center for Environmental Remote Sensing (CEReS), Chiba University
(shunji.kotsuki@chiba-u.jp)

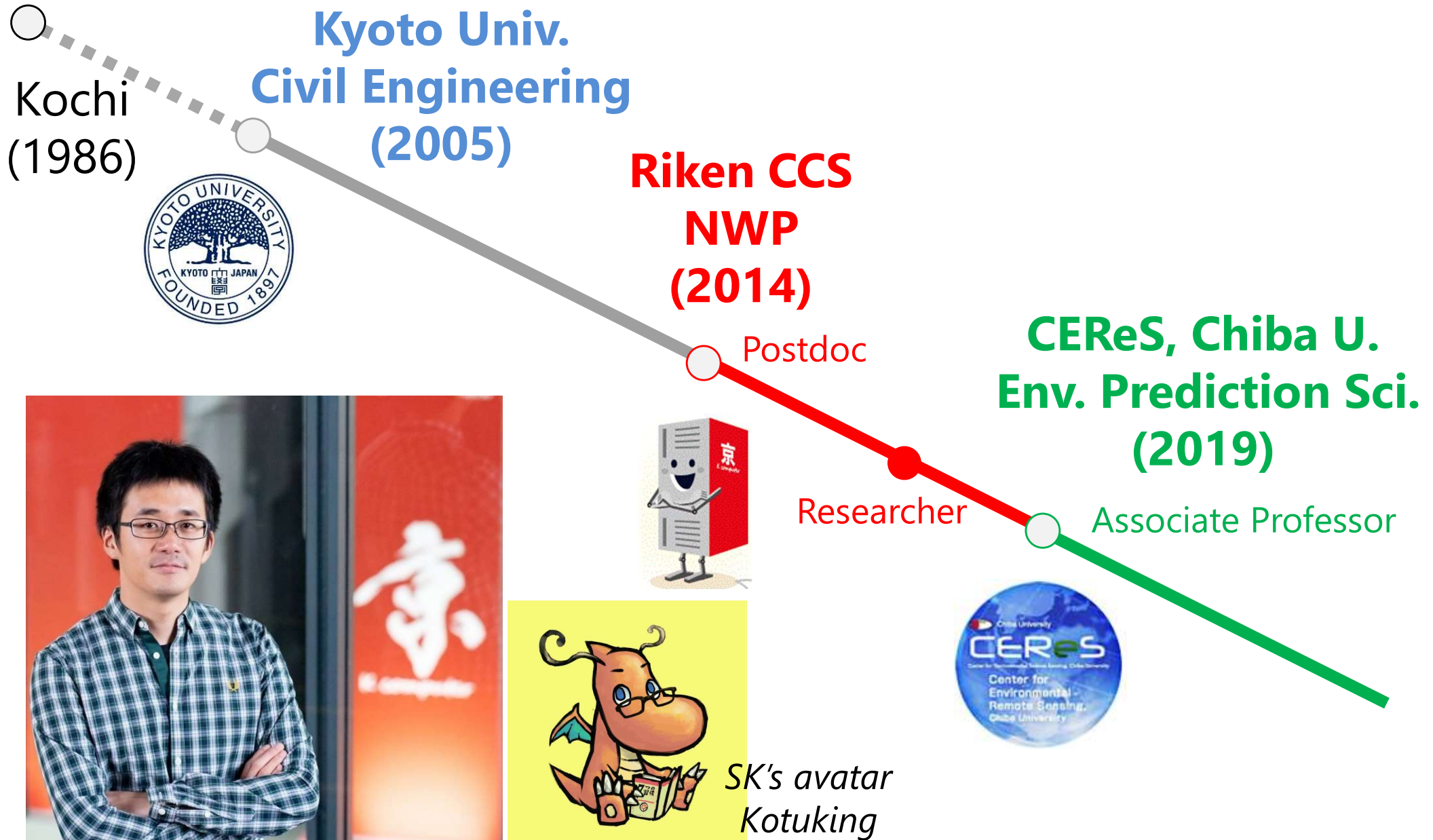


DA Lectures A (Basic Course)



- ▶ (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 & Adaptive Inflation

Who am I? Shunji Kotsuki



Majors: Data Assimilation, Environmental Prediction, Data Science

Brief Understanding of DA



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

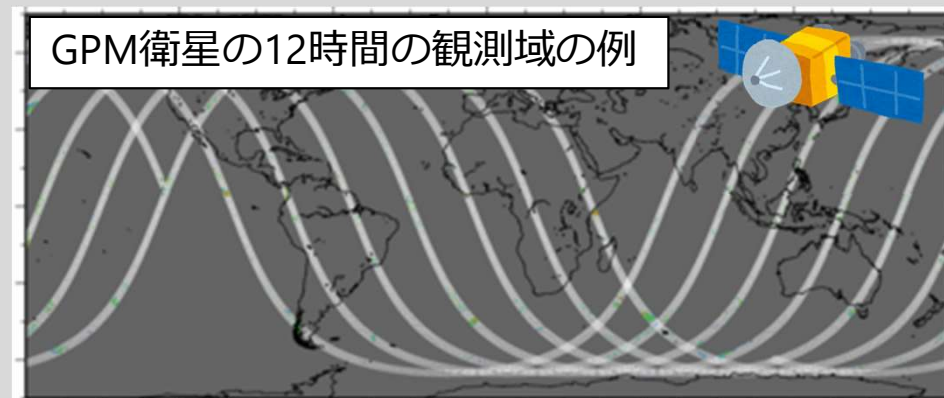
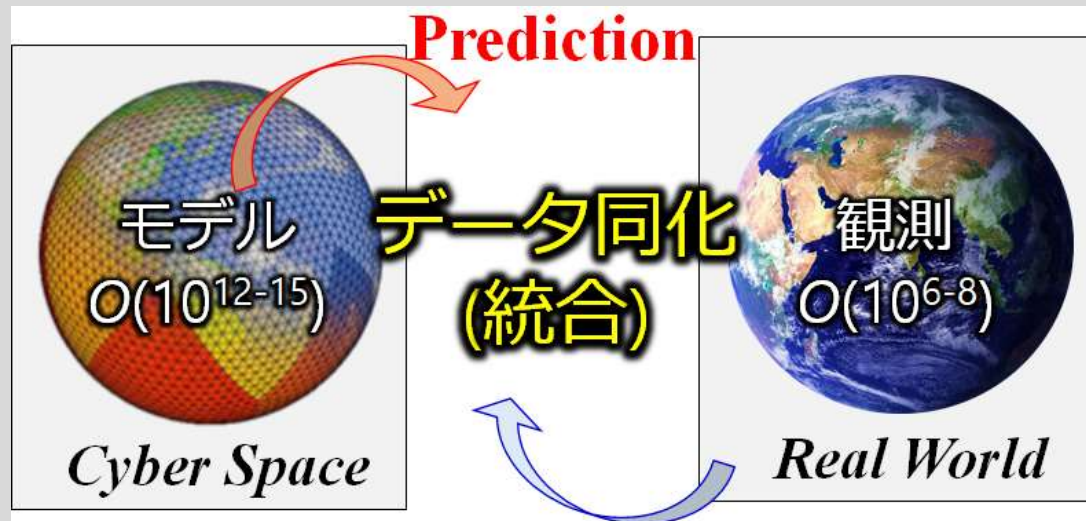
Since DA uses observations,
forecasts must be improved "easily"!



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

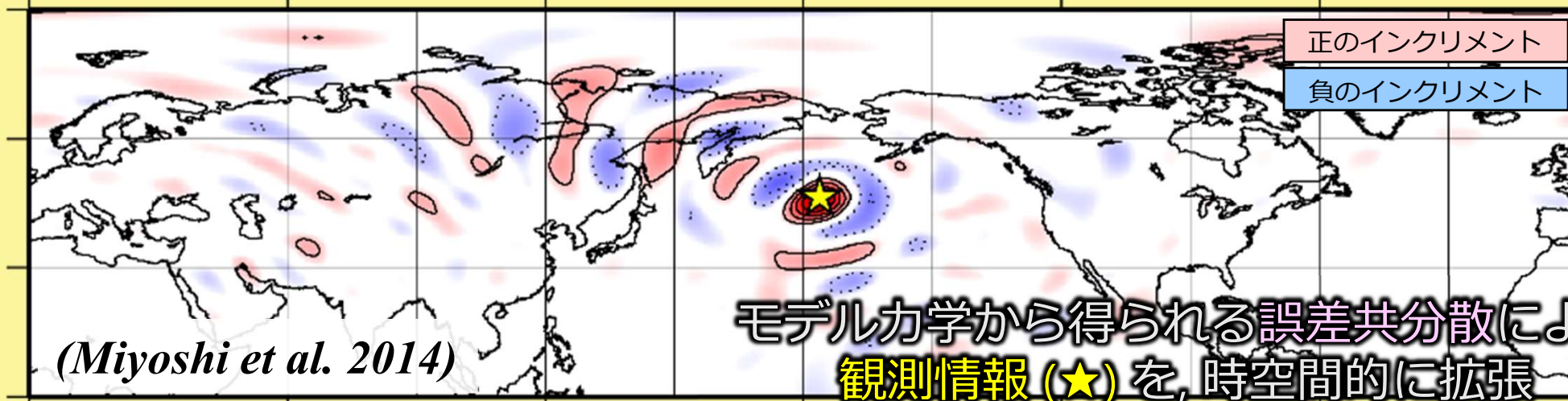
データ同化: モデル&観測の統合

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 **モデル力学を使う同化が重要**

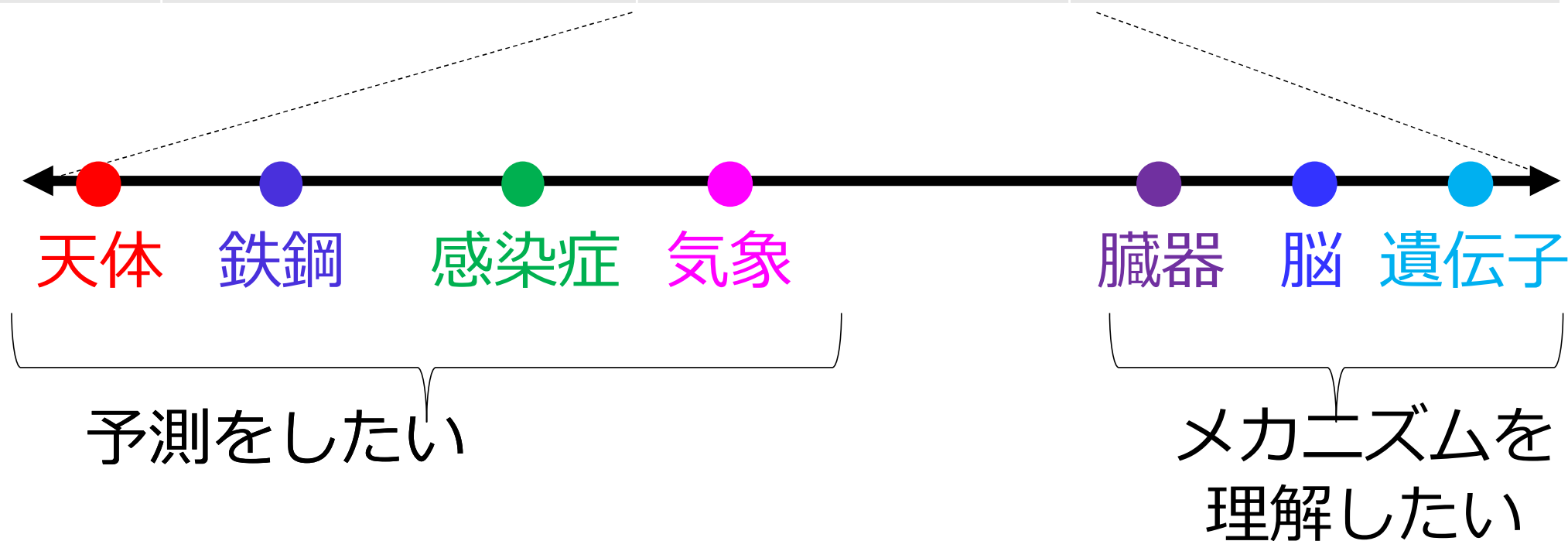
全球大気モデルを用いて推定した大気中層 (500hPa) の気温の空間誤差相関



データ同化研究の広がり



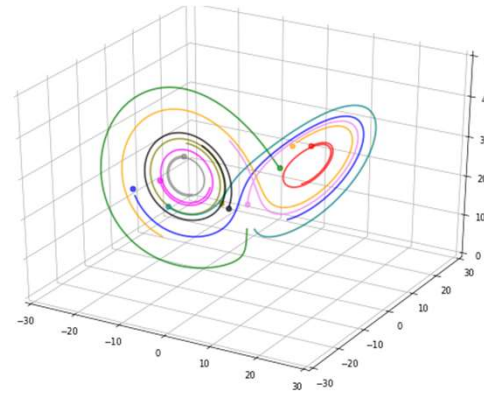
| | シミュレーション (第3の科学) | データ同化 (結び付け) | 機械学習 (第4の科学) |
|-----|---------------------|-----------------|-----------------|
| 性質 | プロセス駆動型 | | データ駆動型 |
| 観測 | 少ない | | 大きい |
| 数理 | 既知 (硬い) | | 未知 (柔らかい) |
| モデル | 大自由度 | | 低自由度 |



DA Research Strategy

1. math & toy models

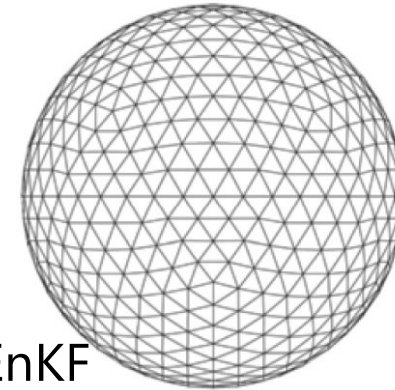
(e.g. Lorenz 96, $n \sim O(10^2)$, $p \sim O(10^2)$)



2. intermediate models

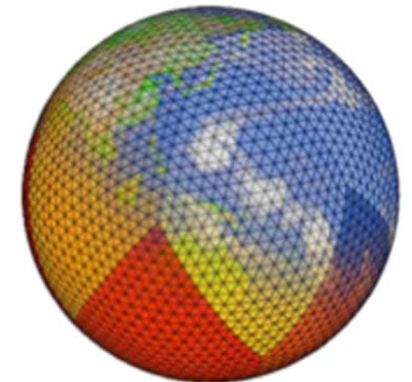
(e.g. SPEEDY, $n \sim O(10^6)$, $p \sim O(10^4)$)

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.

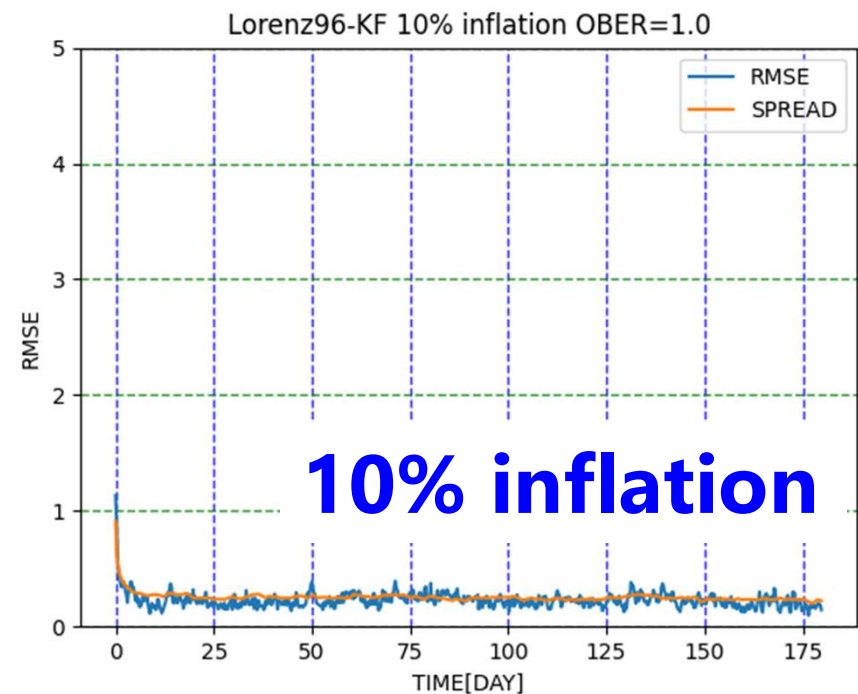
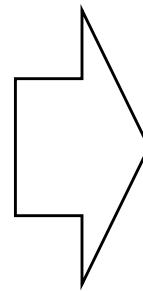
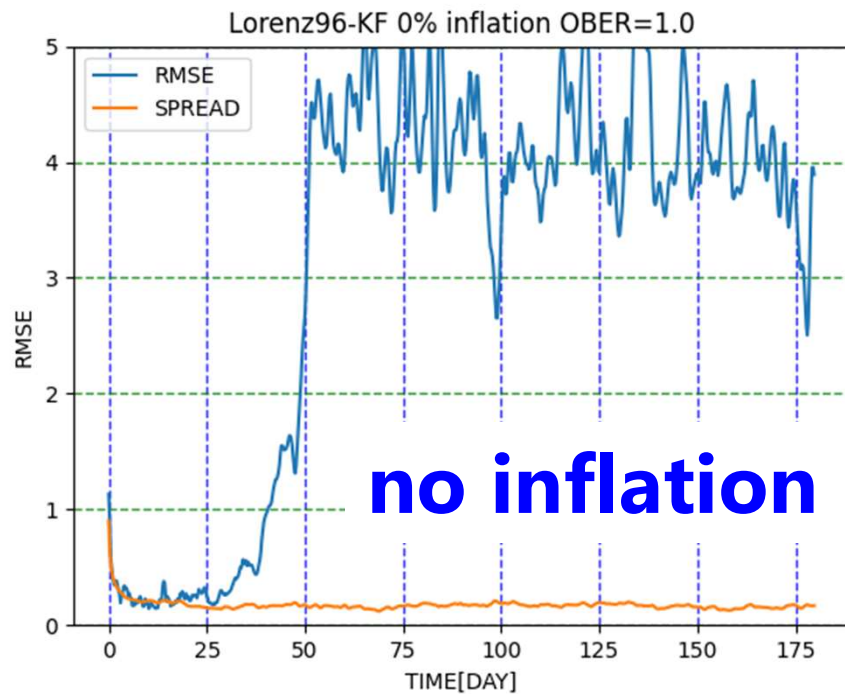
Ex) Variance Inflation (KF, EnKF)

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。

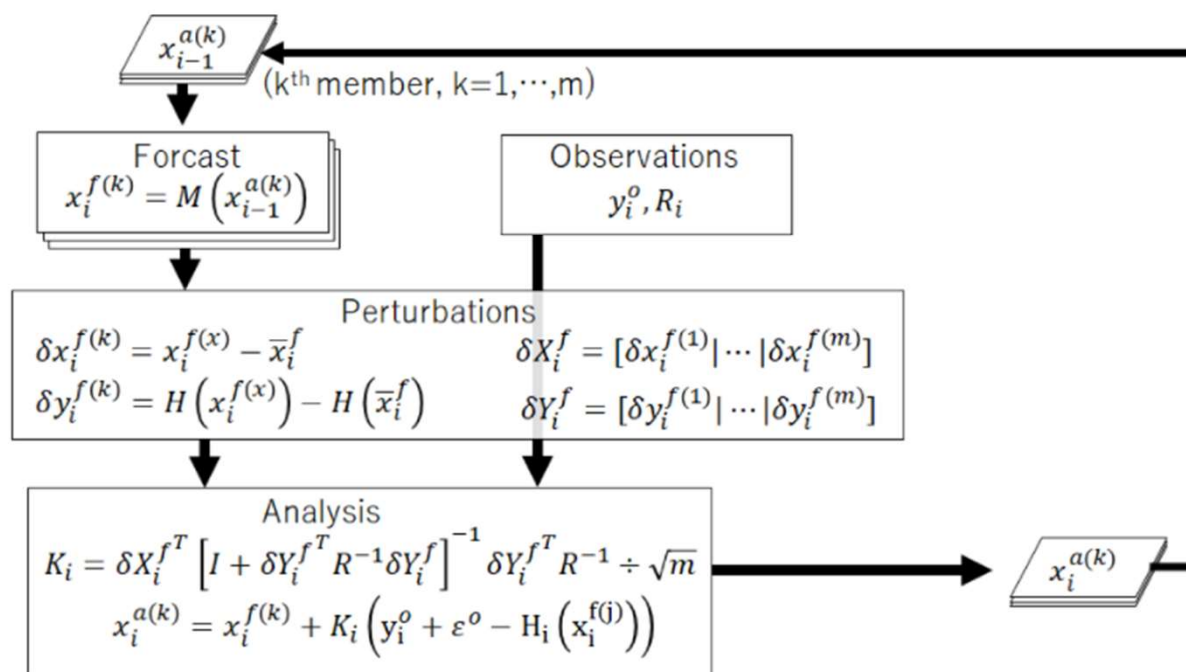
具体的には、制御工学的なフィルターの役割を果たすこともあり現象のシミュレーションと現実世界を結びつける同化する目的でも

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
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
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

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。



Text Books



① Training Description

pswd: ceres

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 Ed

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、こちららご指摘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で、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

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、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.

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

Text Books

② ゼロつく (upon requests)

ゼロからつくるデータ同化

- Lorenz-96 40 変数モデルによる数値実験 -

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

<https://kotsuki-lab.com/>

contact information: shunji.kotsuki@chiba-u.jp

1

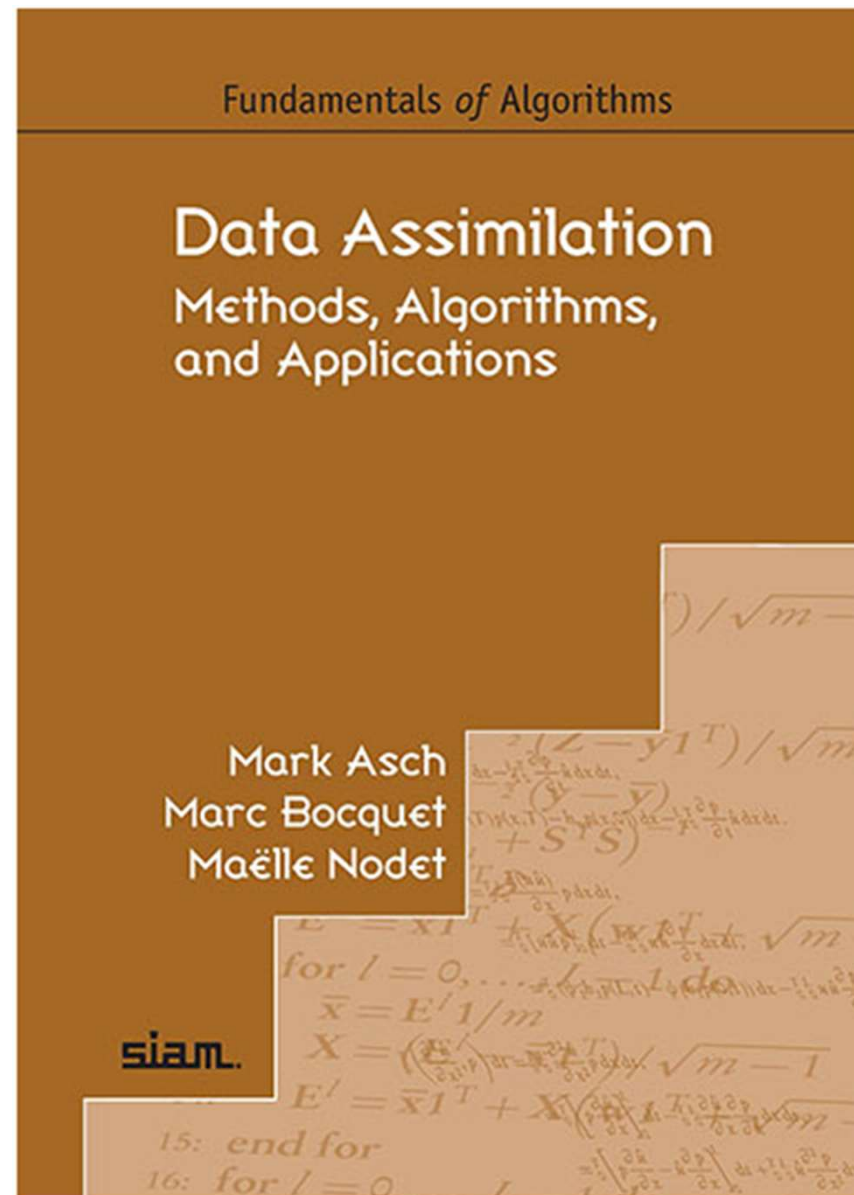
目次

| | |
|--|----|
| | 4 |
| | 5 |
| | 8 |
| | 8 |
| | 8 |
| | 8 |
| | 10 |
| | 10 |
| | 11 |
| | 13 |
| | 14 |
| | 14 |
| | 14 |
| Dynamics)..... | 13 |
| トラクタ13 | 13 |
| デル (L96: 基本課題1)..... | 13 |
| 率 (基本課題2)..... | 13 |
| man Filter)..... | 16 |
| | 16 |
| 導出17 | 17 |
| 変分法 (OI & 3DVAR)..... | 21 |
| | 21 |
| フィルタ (EnKF)..... | 22 |
| | 22 |
| 6.2. 導出..... | 22 |
| 6.3. 観測摂動法 (PO 法; Perturbed Observation)..... | 22 |
| 6.4. 逐次型 Ensemble Square Root Filter (serial EnSRF)..... | 22 |
| 6.5. アンサンブル変換カルマンフィルタ (ETKF)..... | 22 |
| 6.6. 局所変換アンサンブル変換カルマンフィルタ (LETKF)..... | 22 |
| 7. 共分散膨張 (covariance inflation)..... | 23 |
| 7.1. 概略..... | 23 |
| 7.2. multiplicative inflation..... | 23 |
| 7.3. 共分散緩和法..... | 23 |

2

2022年 完成予定

③ TextBook



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

The screenshot shows the website for Kotsuki Lab. The header includes the name 'Kotsuki Lab.' and its affiliation with Chiba University. A navigation menu lists 'Research', 'Achievements', 'Members', 'News', 'Recruit', 'Gallery', 'Contact & Access', and 'Education'. The 'Education' section is highlighted with a red dashed line. It contains information about educational content and a list of Python programming materials, including a manual for Earth Science numerical calculation in Japanese and English. The file 'PythonManual_v20210928.dox' is highlighted with a red box.

The screenshot shows the page for the Python Training Course for Earth Science. The title is 'Python を利用した地球科学・数値解析入門' (Python Training Course for Earth Science). The page is available in both English and Japanese. It provides contact information for Kotsuki Laboratory, CEReS, Chiba University, including the website URL and email address. The page number '1 / 141' is visible at the bottom.

**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

Python Training Course (cont'd)



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

目次

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

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

Outline

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

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 programing skill**
 - ▶ → you will improve programming definitely
- ▶ **know numerical weather prediction**
 - ▶ → you will be more familiar with weather

Let's Start!

Today's Goal

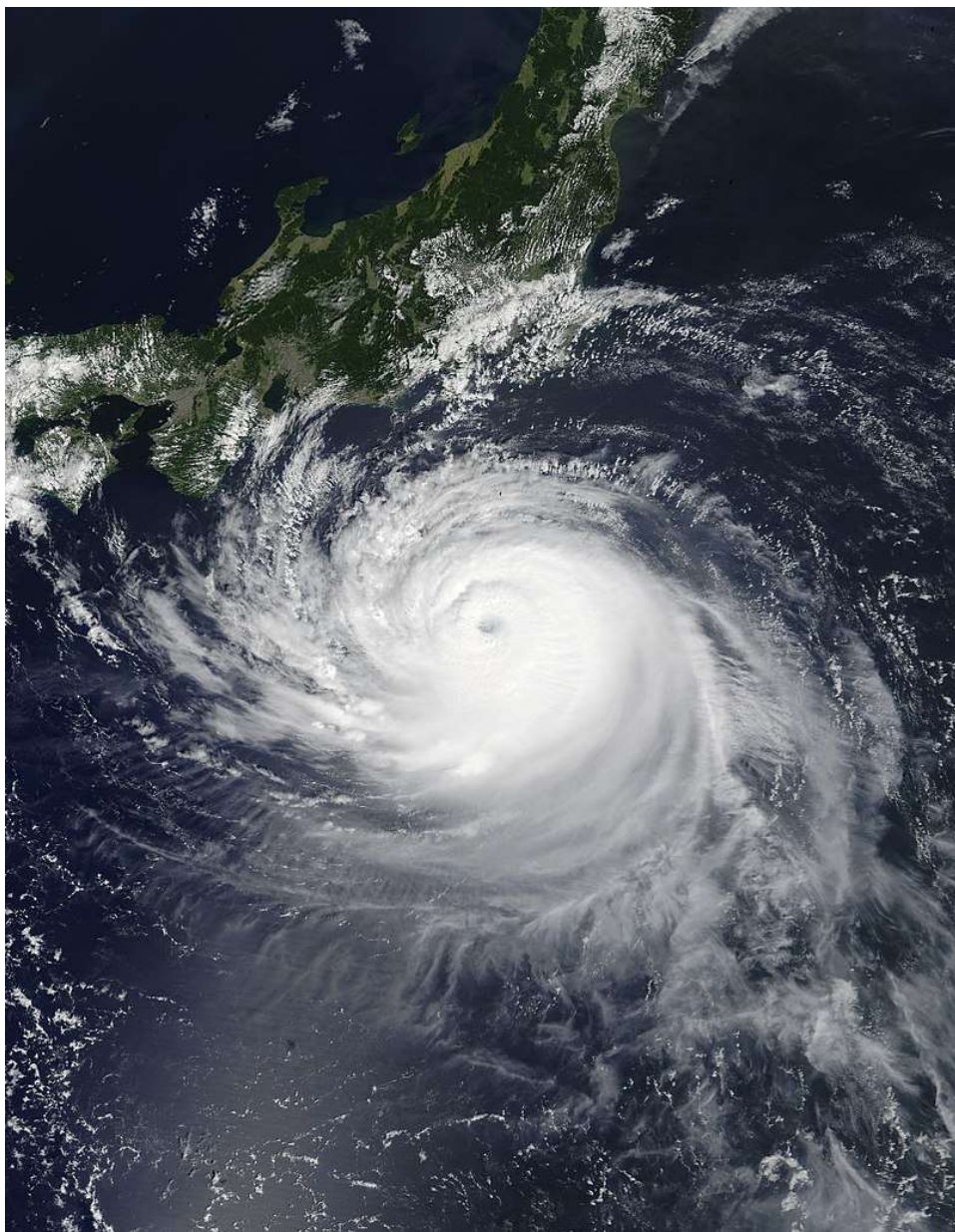


- ▶ To understand numerical weather prediction and the role of the data assimilation
- ▶ To be interested in data assimilation

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

支配方程式 (微分方程式) を解く,とは?

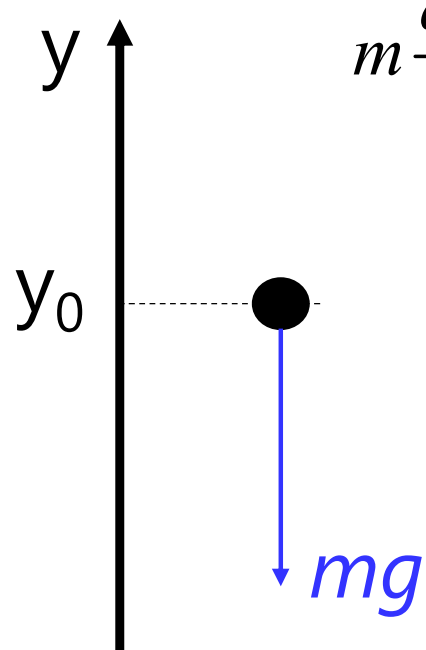
例) 運動方程式: 物体の運動を記述する方程式

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

運動方程式

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

一番簡単な例: 自由落下運動



A vertical y-axis is shown with an upward arrow. A black dot representing an object is at a height y_0 on the axis. A blue arrow labeled mg points downwards from the object, representing the force of gravity.

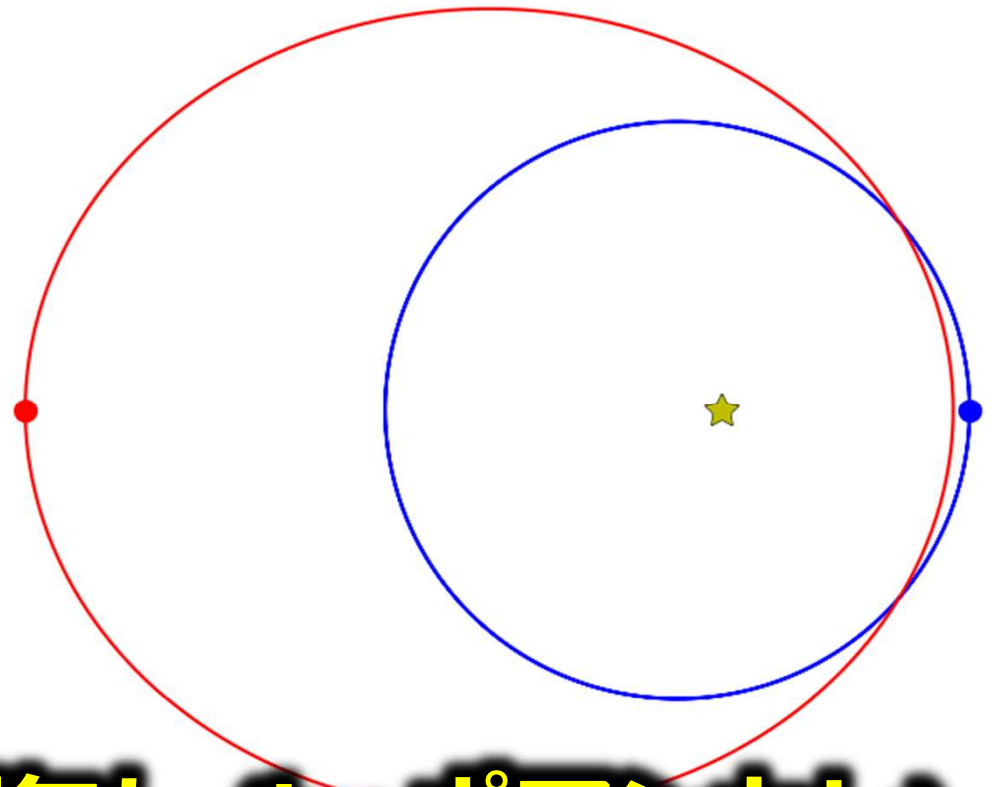
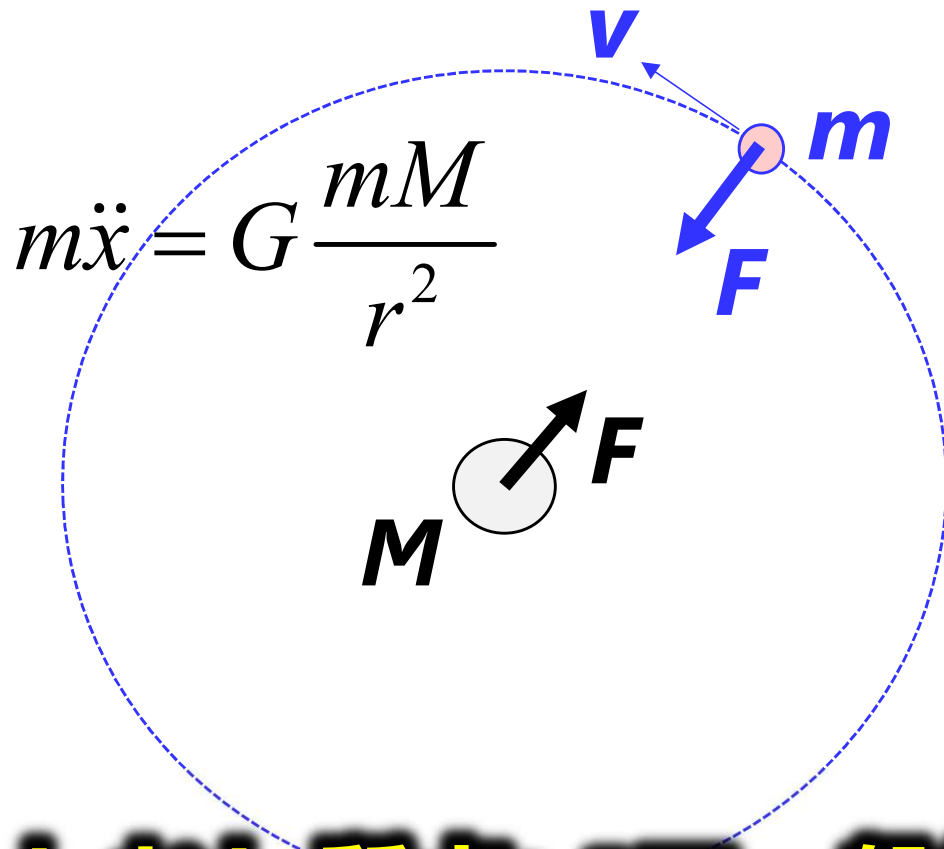
$$m \frac{d^2 y}{dt^2} = -mg \Leftrightarrow \frac{d^2 y}{dt^2} = -g \xrightarrow{\text{積分}} \frac{dy}{dt} = -gt + C = v$$
$$\xrightarrow{\text{積分}} y = -\frac{1}{2}gt^2 + Ct + C'$$

(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

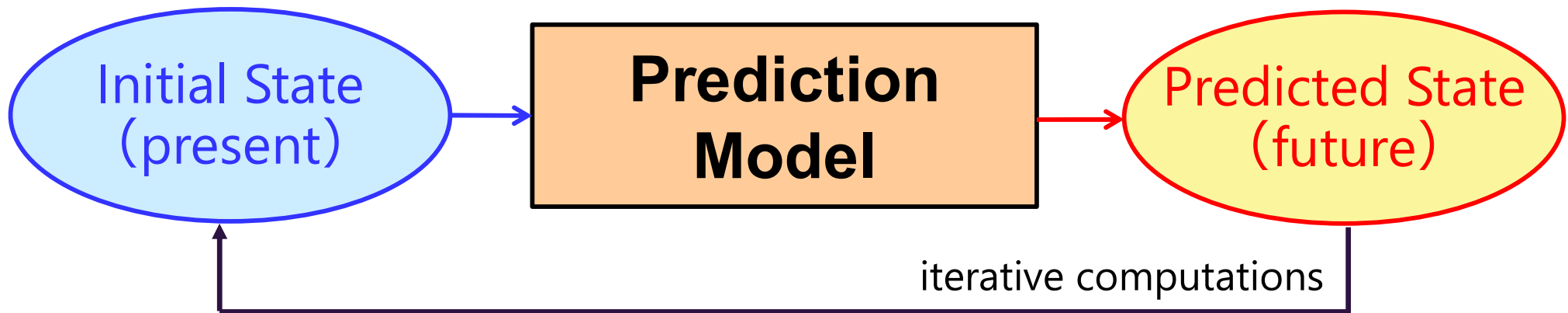
ニュートン力学と求積解の限界

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)

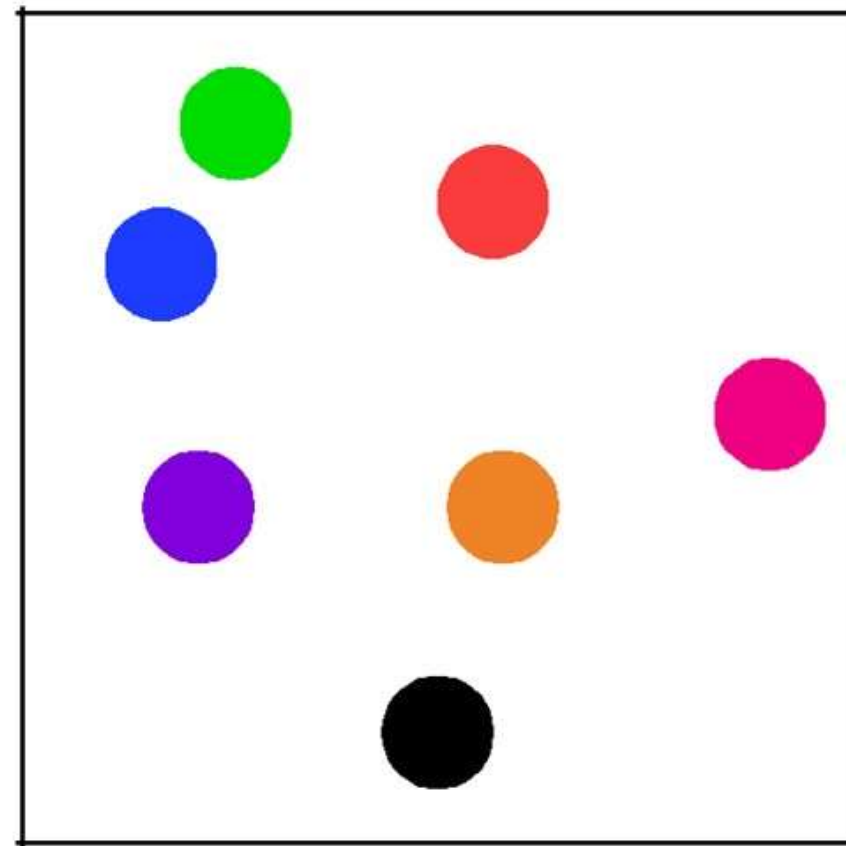


しかし質点 ≥ 3 で一般解無し (by ポアンカレ)
→ コンピュータを使った数値予測へ

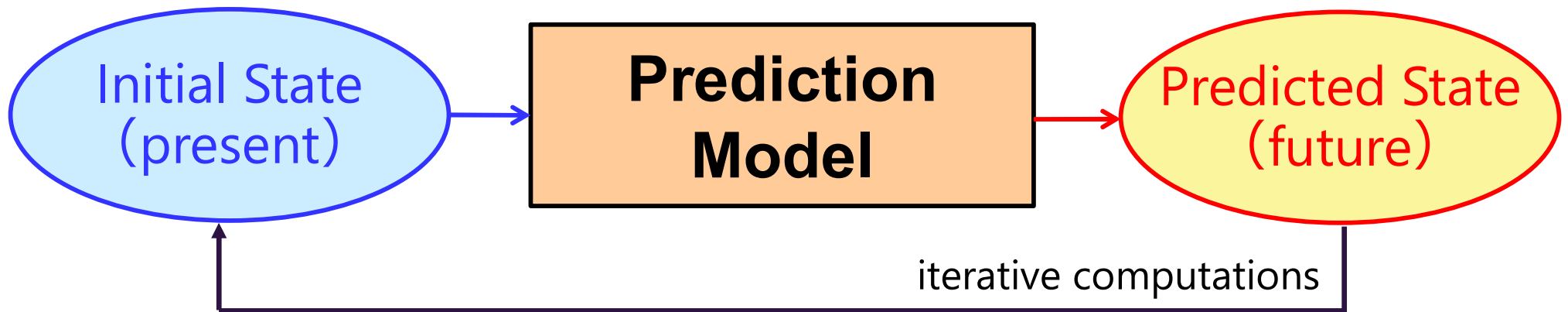
Numerical Simulation w/ Computers



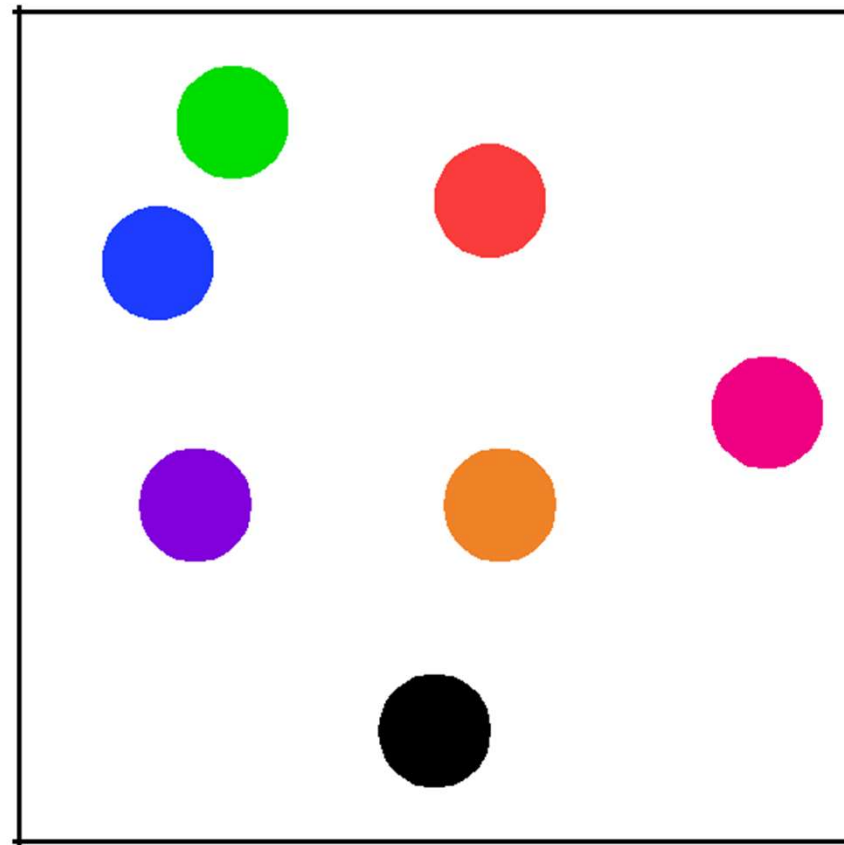
*an example
of billiards*



Numerical Simulation w/ Computers



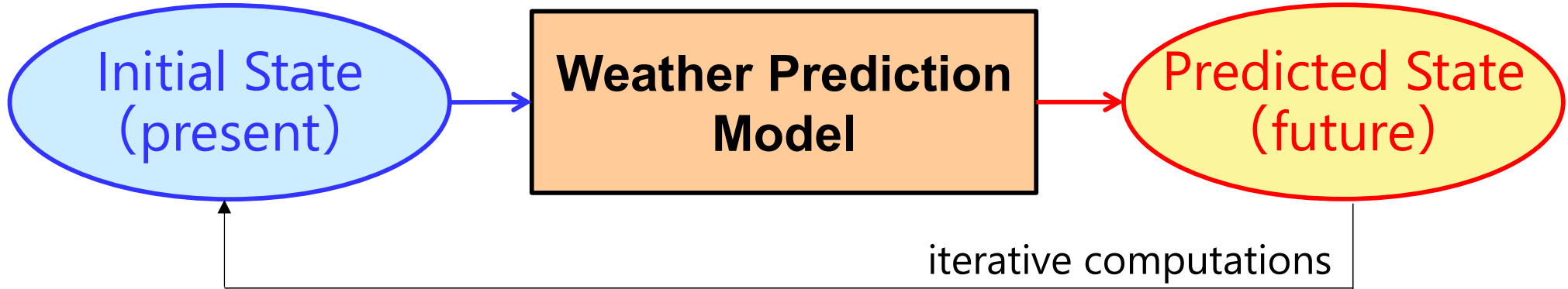
*an example
of billiards*



Numerical Weather Prediction

wind, temperature, humidity, pressure

wind, temperature, humidity, pressure

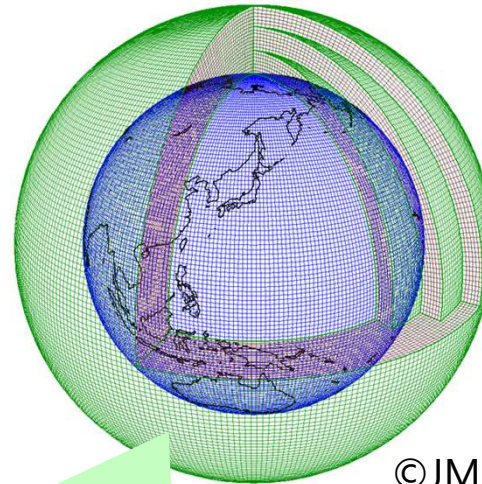


Real Earth

Earth in Computer



discretize



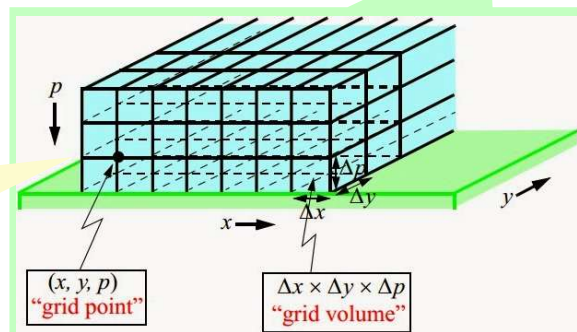
Richardson's Dream (1920; 200km)



Fugaku (2020; 3.5km x 1000 ens)



wind, temperature
humidity, pressure, ...



Physical laws solved in NWP Models

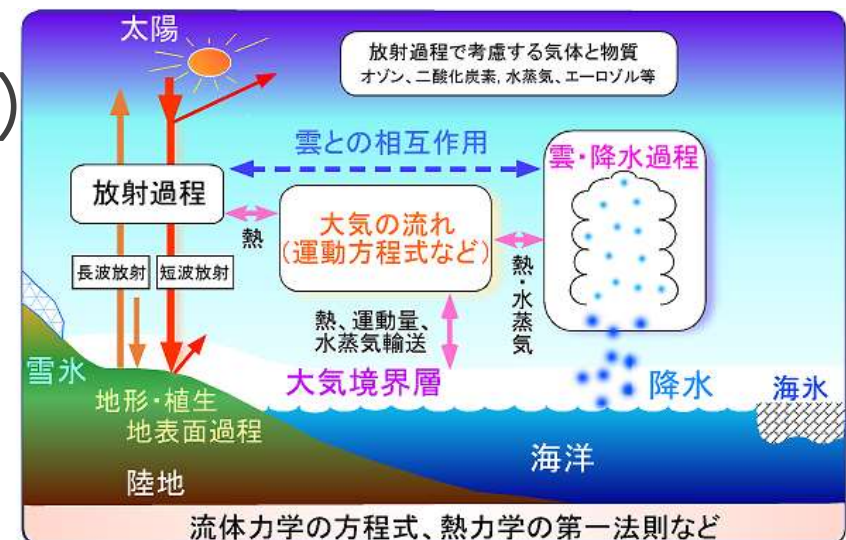
▶ Fluid Dynamics (a.k.a. Dynamical Core)

- ▶ 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



Physical laws solved in NWP Models



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

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

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

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

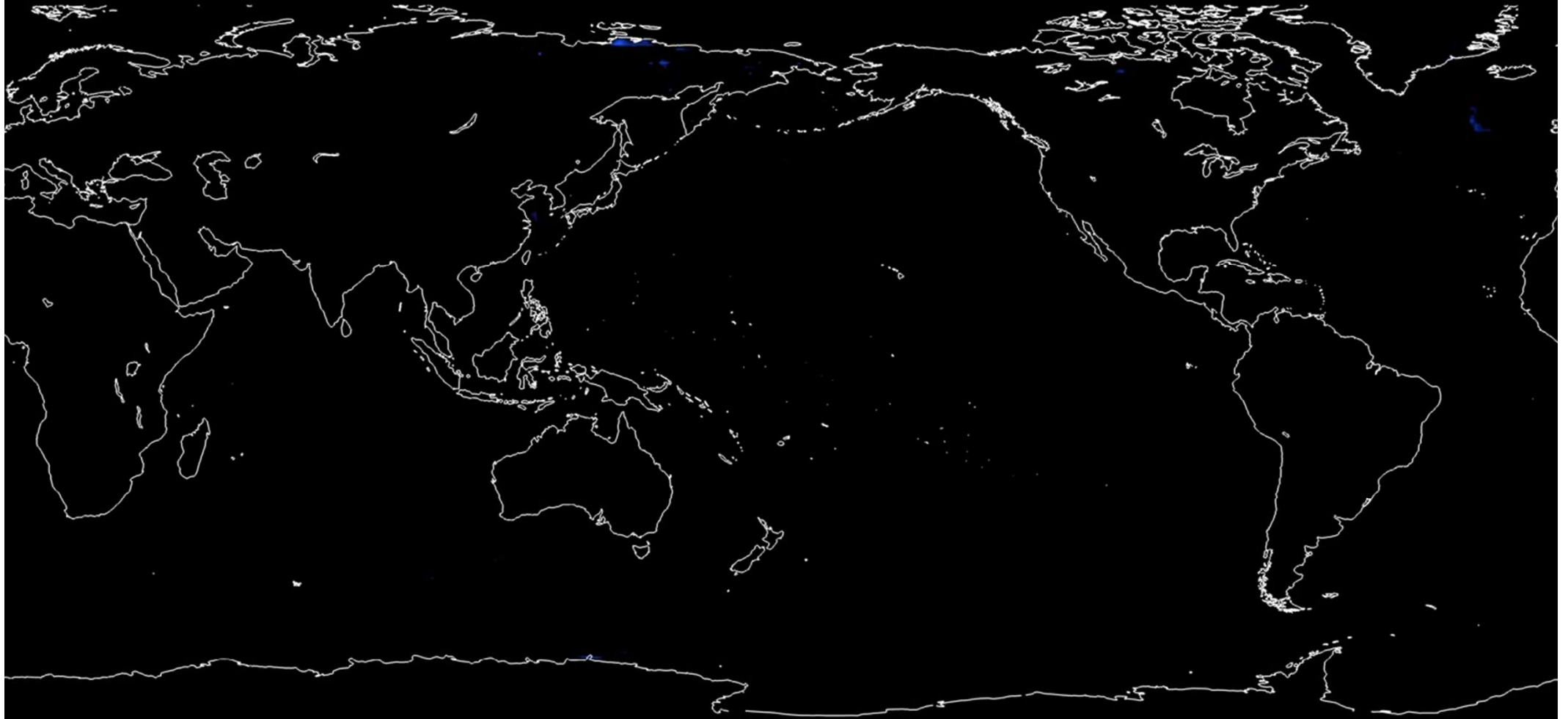
$$p = \rho RT$$

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

\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}).

ρ : 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



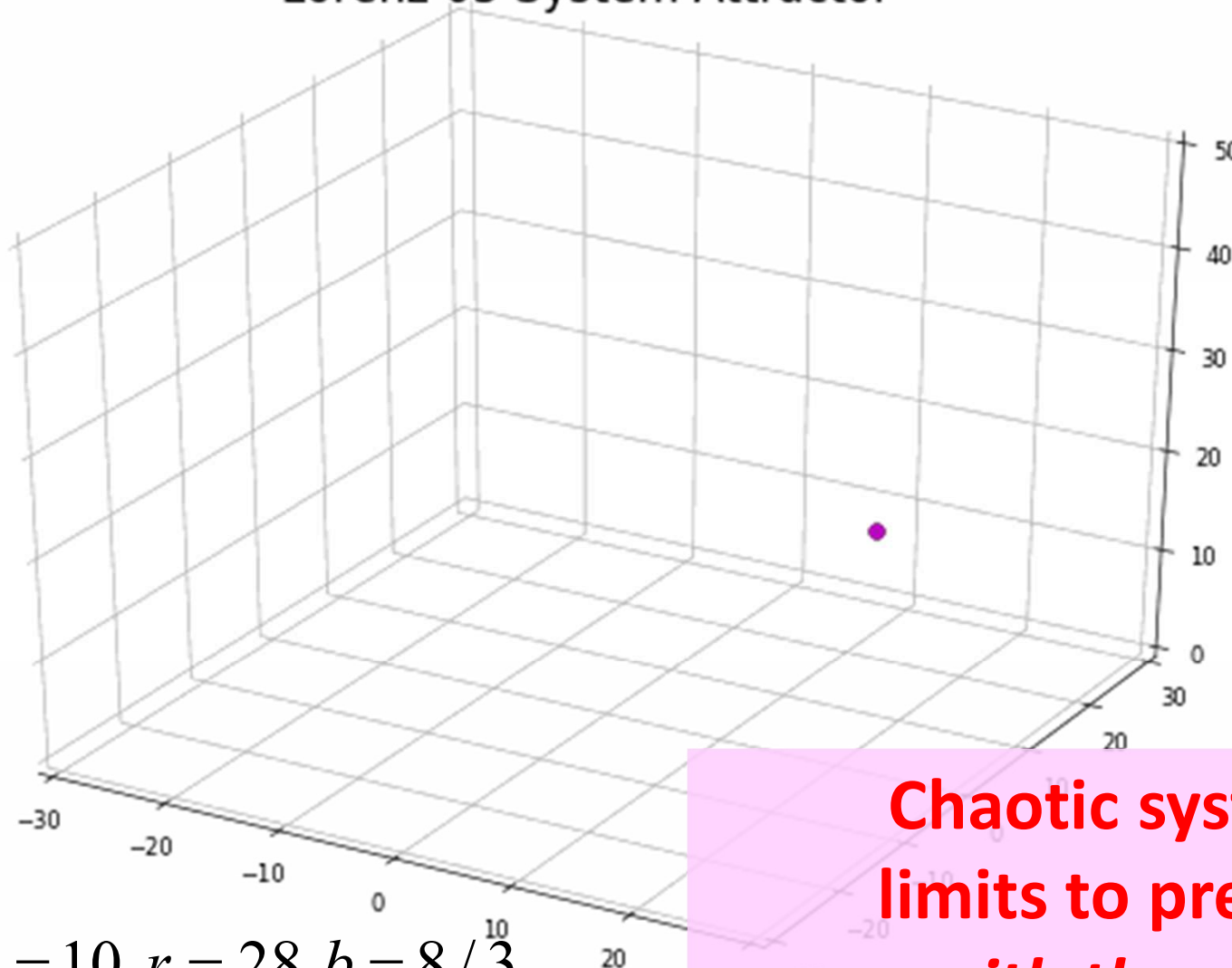
2014/05/25 00:00

Deterministic Chaos and Predictability

Edward Lorenz



Lorenz 63 System Attractor



Lorenz 63 model

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

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

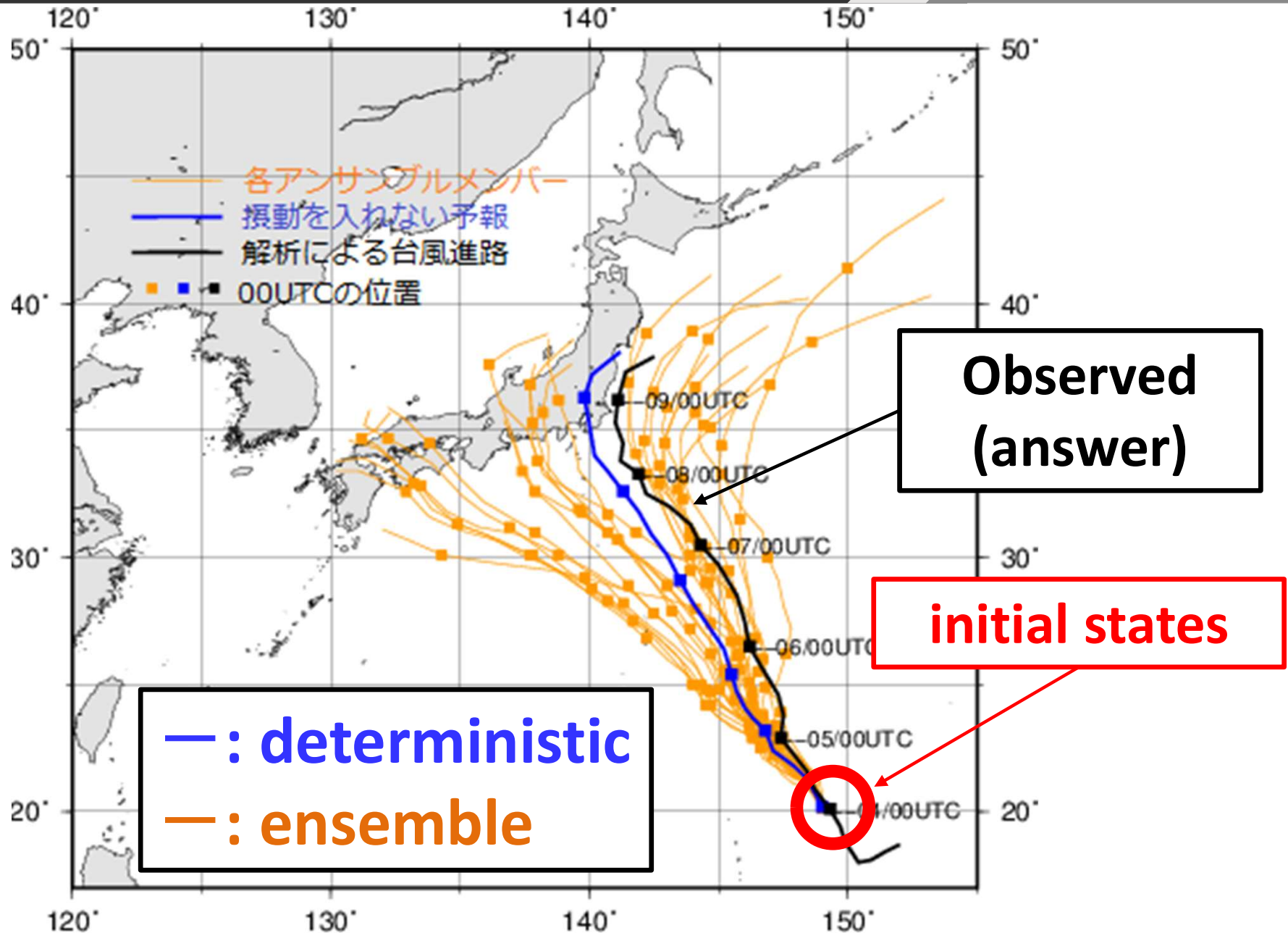
$$\dot{z} = xy - bz$$

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

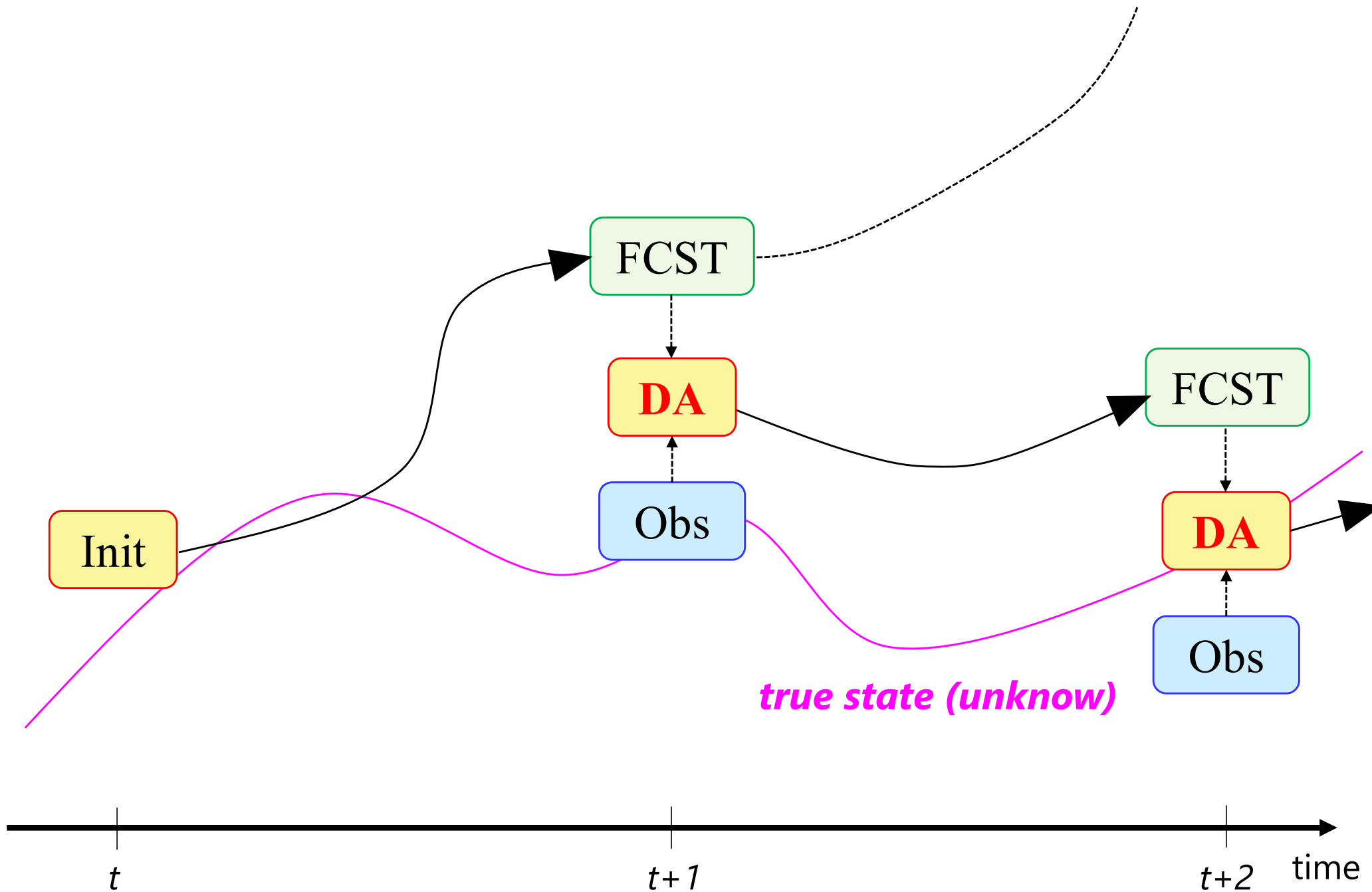
Chaotic systems have limits to predictability even with the perfect model!!

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

Ensemble Prediction (e.g. TC)

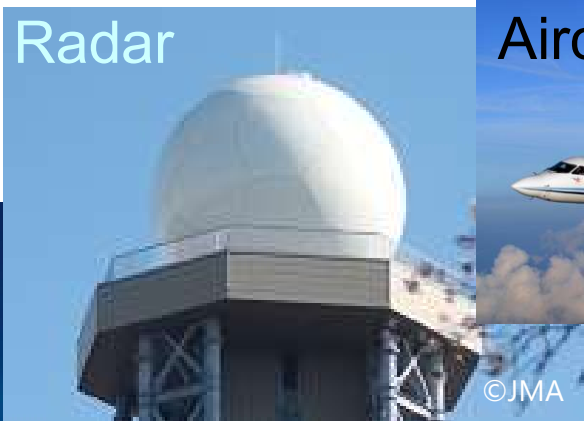


Numerical Weather Prediction

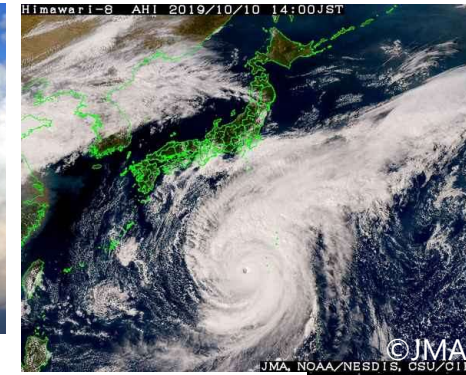


Global Observing System

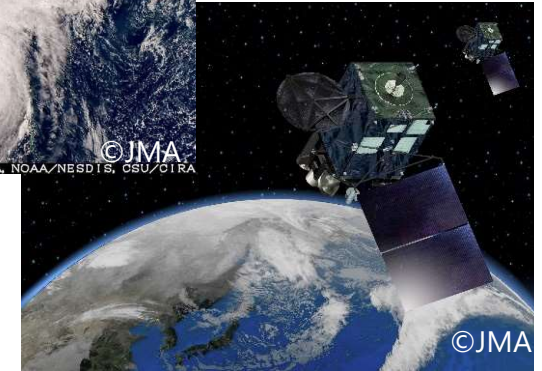
Radar



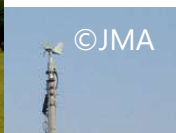
Aircraft



Satellite



Radiosonde



Ship



Surface station

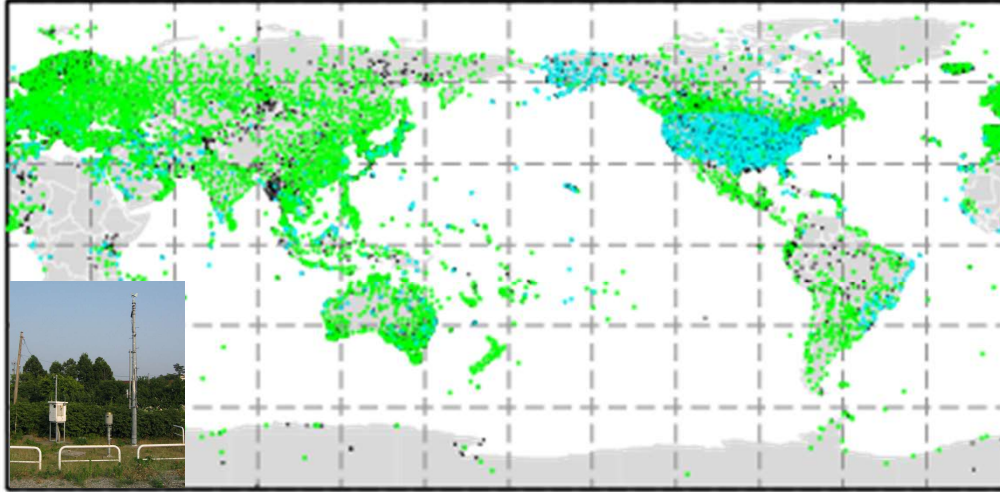


Buoy

Observation Data in NWP

LAND SURFACE

2019/05/01 00:00(UTC)

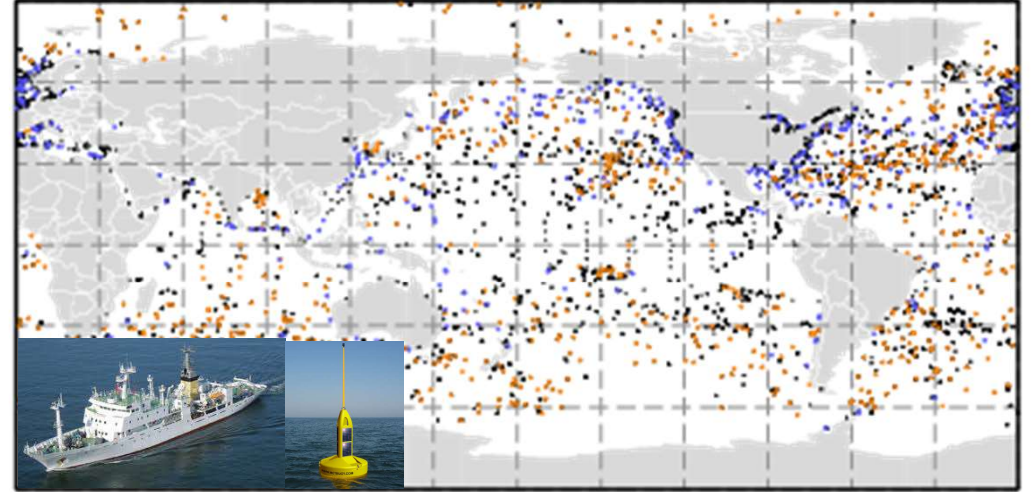


SYNOP[●]: 4163 METAR[●]: 1178
 NOUSE[●]: 18901 NOUSE[●]: 42966
 ALL: 23064 ALL: 44144



SEA SURFACE

2019/05/01 00:00(UTC)

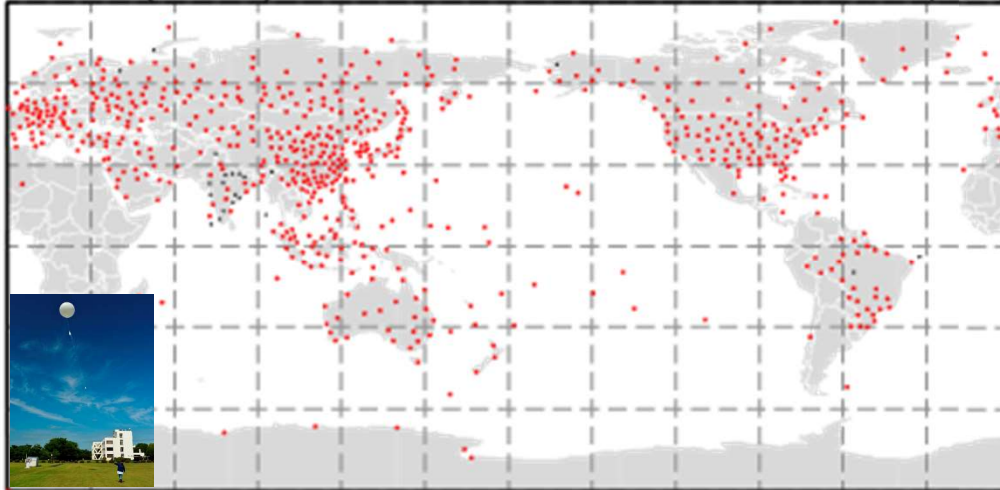


SHIP[●]: 378 DRIFTER[●]: 709
 NOUSE[●]: 5668 NOUSE[●]: 10206
 ALL: 6046 ALL: 10915



UPPER(TEMP)

2019/05/01 00:00(UTC)

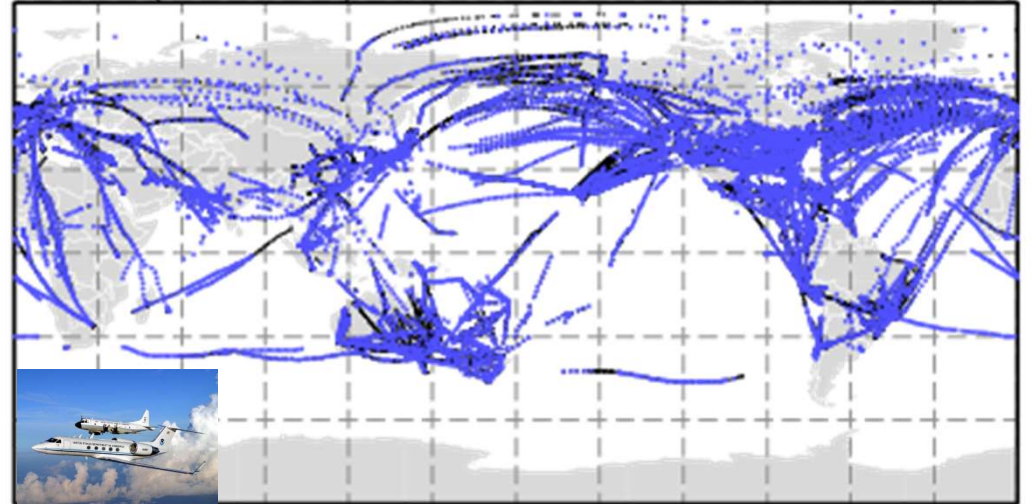


TEMP[●]: 632
 NOUSE[●]: 24
 ALL: 656



UPPER(AVIATION)/BOGUS

2019/05/01 00:00(UTC)



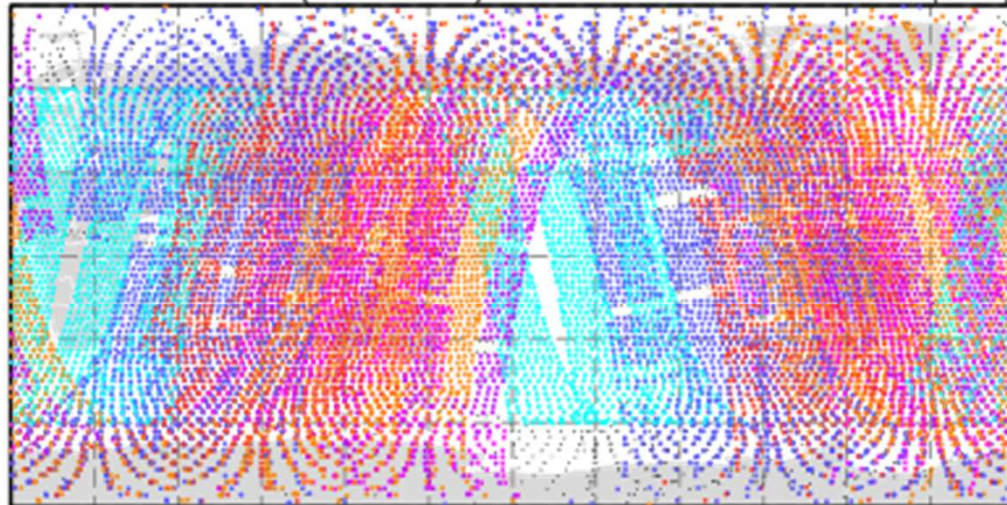
TYBOGUS[●]: 0 YHTC AVIATION[●]: 9919
 NOUSE[●]: 0 NOUSE[▼]: 0 NOUSE[●]: 82767
 ALL: 0 ALL: 0 ALL: 98686



Satellite Data in NWP

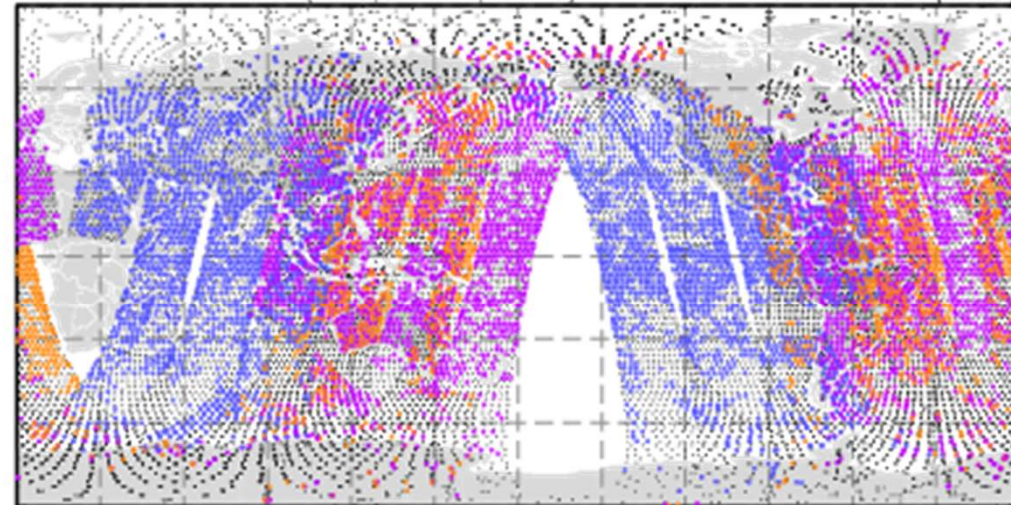


MW-SOUNDER(AMSU-A) 2019/05/01 00:00(UTC)



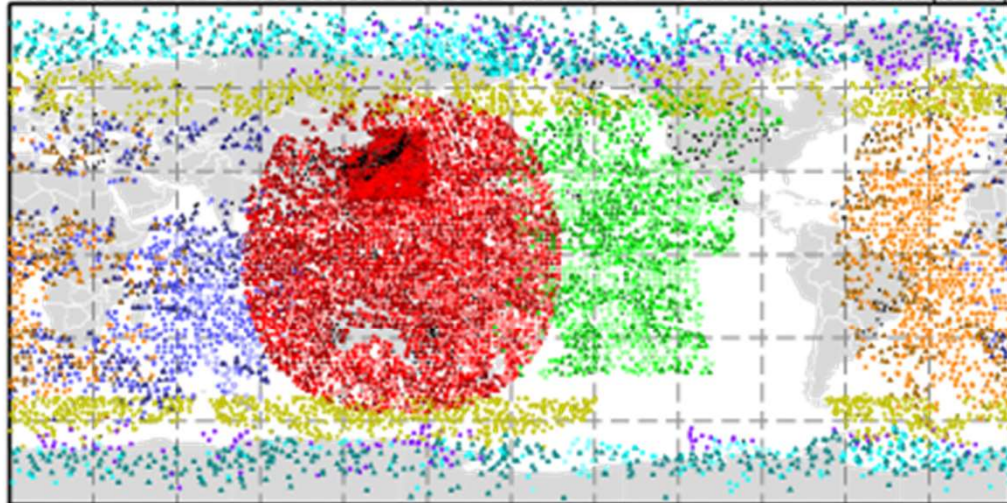
| | | | | | |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| NOAA-15 | NOAA-18 | NOAA-19 | Aqua | Metop-A | Metop-B |
| AMSU-A[●]: 3606 | AMSU-A[●]: 1625 | AMSU-A[●]: 4835 | AMSU-A[●]: 3180 | AMSU-A[●]: 4881 | AMSU-A[●]: 2806 |
| NOUSE[●]: 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)



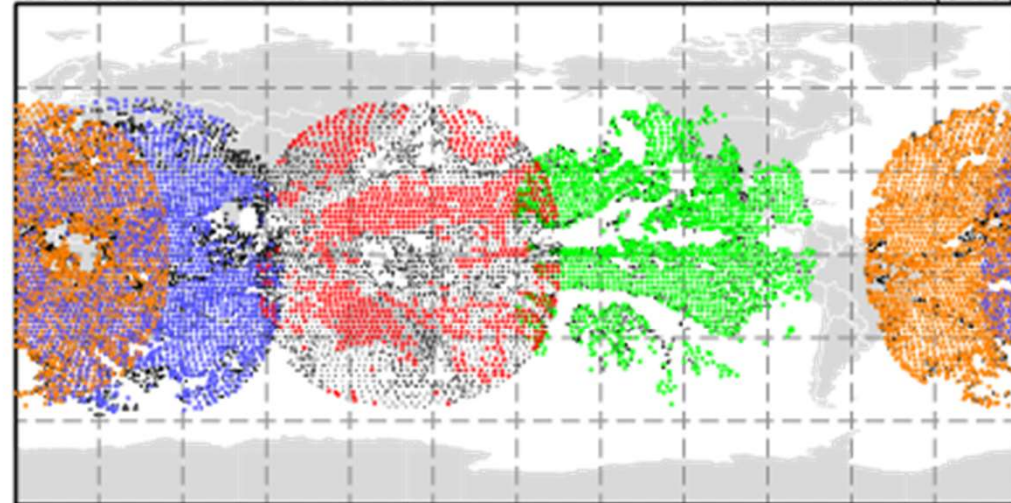
| | | |
|----------------|----------------|----------------|
| NOAA-19 | Metop-A | Metop-B |
| MHS[●]: 3864 | MHS[●]: 2462 | MHS[●]: 3881 |
| NOUSE[●]: 4854 | NOUSE[●]: 3693 | NOUSE[●]: 5843 |
| ALL: 8718 | ALL: 6155 | ALL: 9724 |

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[▽]: 464 | VIS[▽]: 124 | VIS[▽]: 5 | WV[▲]: 773 | | |
| WV[▲]: 1875 | WV[▲]: 513 | WV[▲]: 314 | WV[▲]: 372 | CVV[★]: 107 | | |
| SPC[■]: 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)

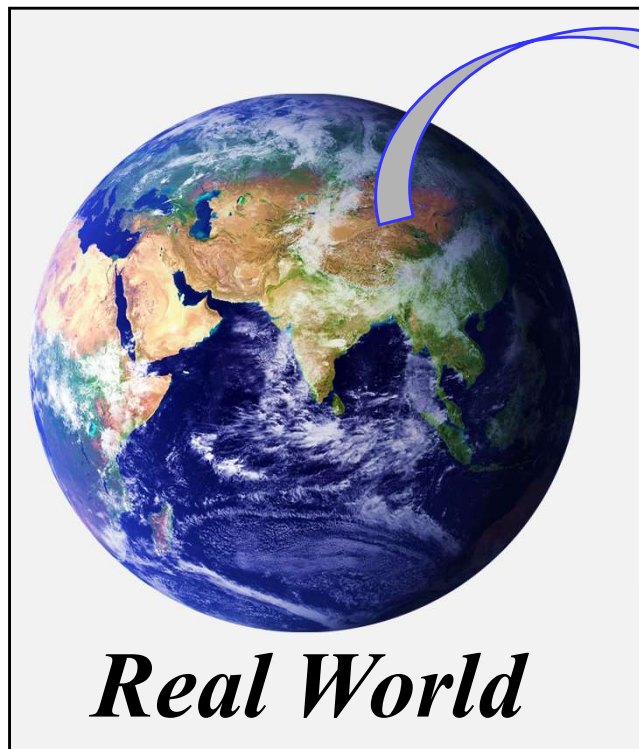


| | | | |
|----------------|-----------------|----------------|----------------|
| Himawari-8 | GOES-15 | Meteosat-8 | Meteosat-11 |
| AH[●]: 2812 | IMAGER[●]: 2257 | SEVIR[●]: 4094 | SEVIR[●]: 5855 |
| NOUSE[●]: 8031 | NOUSE[●]: 2109 | NOUSE[●]: 5723 | NOUSE[●]: 4329 |
| ALL: 10843 | ALL: 4366 | ALL: 9817 | ALL: 10184 |

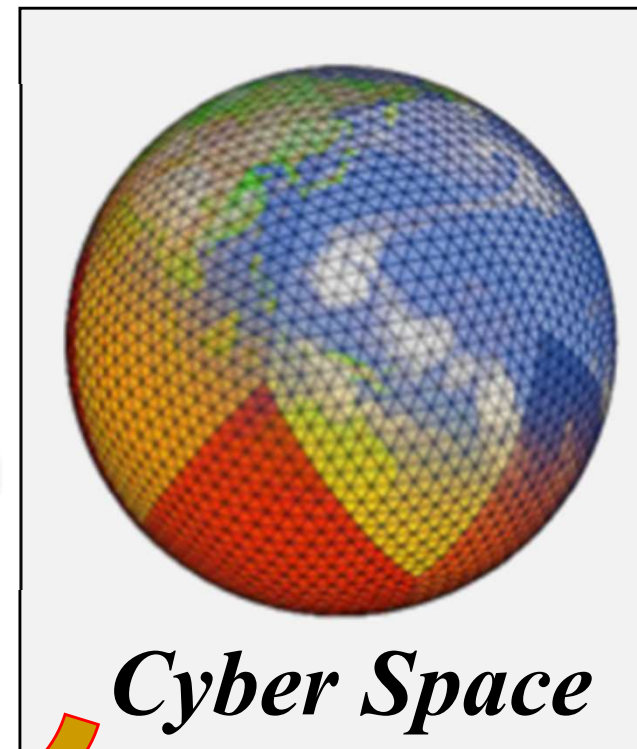
courtesy of JMA (2019/05/01 00:00 UTC)

Data Assimilation

Observation



DA
(best estimates)



Prediction

**Data-driven
(inductive)**

**Process-driven
(deductive)**

**Sparse & Infrequent
(low-dimensional)**

**Dense & Frequent
(high-dimensional)**

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)

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

