Data Assimilation
- A04. Kalman Filter -

Shunji Kotsuki

Center for Environmental Remote Sensing / Institute of Advanced Academic Research

(shunji.kotsuki@chiba-u.jp)

CHIBA UNIVERSITY

Environmental
Prediction
Science
Laboratory

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 Inflations
- (11) 4D Variational Method (4DVAR)

Today's Goal

- **Lecture: Kalman Filter (KF)**
  - to introduce background error covariance
  - to introduce analysis error covariance
  - to introduce Kalman gain

- **Training: Lorenz 96**
  - to develop Tangent Linear Model (TLM)
  - to implement Kalman filter into L96

Review:

Minimum Variance Estimation

(復習: 最小分散推定)

Minimum Variance Estimation

forecast $x_1 = x^{tru} + \varepsilon_1$

observation $x_2 = x^{tru} + \varepsilon_2$

x^{tru}: truth

ε : random error

< ・ >: *expectation*

Assumption (1) : unbiased error

$$\langle x_1 \rangle = \langle x_2 \rangle = x^{tru} \quad \Leftrightarrow \quad \langle \varepsilon_1 \rangle = \langle \varepsilon_2 \rangle = 0$$

Assumption (2) : uncorrelated error

$$\langle \varepsilon_1 \cdot \varepsilon_2 \rangle = 0$$

Minimum Variance Estimation

forecast $x_1 = x^{tru} + \varepsilon_1$ (1) unbias $\langle x_1 \rangle = \langle x_2 \rangle = x^{tru}$

observation $x_2 = x^{tru} + \varepsilon_2$ (2) uncorr. $\langle \varepsilon_1 \varepsilon_2 \rangle = 0$

$x^a = \alpha x_1 + (1-\alpha) x_2$ & minimize variance of analysis (α)

$$(\sigma^a)^2 = \langle (x^a - x^{tru})^2 \rangle = \langle (\alpha(\underline{x_1 - x^{tru}}) + (1-\alpha)(\underline{x_2 - x^{tru}}))^2 \rangle$$

$$= \alpha^2 \langle \varepsilon_1^2 \rangle + \cancel{2\alpha(1-\alpha)\langle \varepsilon_1 \varepsilon_2 \rangle} + (1-\alpha)^2 \langle \varepsilon_2^2 \rangle$$

$$= \alpha^2 \sigma_1^2 + (1-\alpha)^2 \sigma_2^2$$

definition of variance

$$V(x) = E\left(\left(x - E(x)\right)^2\right)$$

$$\sigma^2 = \langle \varepsilon \cdot \varepsilon \rangle$$

σ : standard deviation

σ^2: variance

Minimum Variance Estimation

forecast $x_1 = x^{tru} + \varepsilon_1$

observation $x_2 = x^{tru} + \varepsilon_2$

(1) unbias $\langle x_1 \rangle = \langle x_2 \rangle = x^{tru}$

(2) uncorr. $\langle \varepsilon_1 \varepsilon_2 \rangle = 0$

$$x^a = \alpha x_1 + (1-\alpha) x_2$$

$$(\sigma^a)^2 = \alpha^2 \sigma_1^2 + (1-\alpha)^2 \sigma_2^2$$

$$\frac{\partial (\sigma^a)^2}{\partial \alpha} = 2\alpha\sigma_1^2 - 2(1-\alpha)\sigma_2^2 = 0$$

$$\Leftrightarrow \quad \alpha = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}$$

weighted average by variance (σ^2; =accuracy)

$$x^a = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} x_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} x_2$$

$$= \underbrace{x_1}_{\text{first guess}} + \underbrace{\frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} (x_2 - x_1)}_{\text{increment}}$$

Kalman Filter

Exercise

- **to introduce Kalman gain w/ following Equations**

$$\frac{\partial}{\partial \mathbf{X}} tr(\mathbf{X}\mathbf{Y}\mathbf{X}^T) = \mathbf{X}(\mathbf{Y} + \mathbf{Y}^T)$$

$$\frac{\partial}{\partial \mathbf{X}} tr(\mathbf{X}\mathbf{Y}) = \mathbf{Y}^T$$

Assumption & Definition

Assumption (1) : unbiased error

$$\mathbf{x}^b = \mathbf{x}^{tru} + \boldsymbol{\varepsilon}^b \qquad \langle \boldsymbol{\varepsilon}^b \rangle = 0$$

$$\mathbf{x}^a = \mathbf{x}^{tru} + \boldsymbol{\varepsilon}^a \qquad \langle \boldsymbol{\varepsilon}^a \rangle = 0$$

$$\mathbf{y}^o = \underset{\substack{\parallel \\ H(\mathbf{x}^{tru})}}{\mathbf{y}^{tru}} + \boldsymbol{\varepsilon}^o \qquad \langle \boldsymbol{\varepsilon}^o \rangle = 0$$

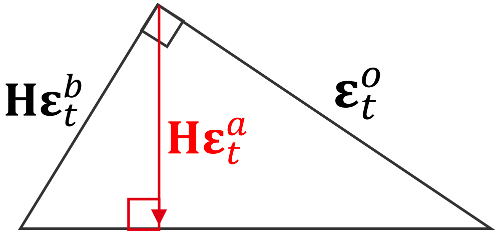
Assumption (2) : uncorrelated error

$$\left\langle \mathbf{H}\boldsymbol{\varepsilon}_t^b (\boldsymbol{\varepsilon}_t^o)^T \right\rangle = \left\langle (\boldsymbol{\varepsilon}_t^o)^T \mathbf{H}\boldsymbol{\varepsilon}_t^b \right\rangle = 0$$

since background and obs errors are independent

$$\langle \mathbf{H}\boldsymbol{\varepsilon}_t^a (\boldsymbol{\varepsilon}_t^o)^T \rangle \neq 0$$

$$\left\langle \boldsymbol{\varepsilon}_t^b (\boldsymbol{\varepsilon}_t^a)^T \right\rangle \neq 0$$



$\mathbf{H}\boldsymbol{\varepsilon}_t^b$
$\boldsymbol{\varepsilon}_t^o$
$\mathbf{H}\boldsymbol{\varepsilon}_t^a$

x	model state	$\in \mathbb{R}^n$
ε	error	
y	observation	$\in \mathbb{R}^p$
M( )	nonlinear model	
M	Jacobian of *M*	$\in \mathbb{R}^{n \times n}$
K	Kalman gain	$\in \mathbb{R}^{n \times p}$
H( )	nonlin. obs. operator	
H	Jacobian of *H*	$\in \mathbb{R}^{p \times n}$
P	model error covariance	$\in \mathbb{R}^{n \times n}$
R	obs. error covariance	$\in \mathbb{R}^{p \times p}$
n	# of model vars.	
p	# of observations	
m	# of ensemble	
tru	truth	
b	background	
a	analysis	
t	time	
o	observation	
< >	expectation	

Error Covariance

Variance, Standard Deviation

$$Var(x) \equiv \langle (x - \langle x \rangle)^2 \rangle \equiv \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 \qquad Std(x) \equiv \sqrt{Var(x)}$$

Covariance

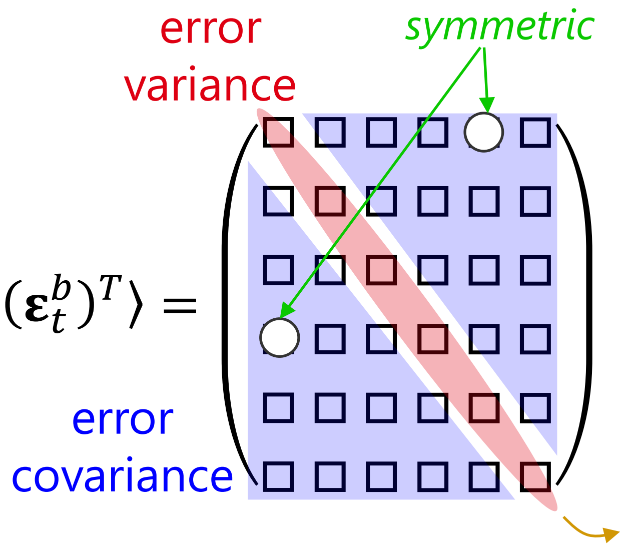
$$Cov(x,y) \equiv \langle (x - \langle x \rangle)(y - \langle y \rangle) \rangle \equiv \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})$$

Correlation

$$Corr(x,y) \equiv Cov(x,y)/Std(x)/Std(y)$$

Error Covariance

$$\mathbf{P}_t^b \equiv \left\langle \left(\boldsymbol{\varepsilon}_t^b - \cancel{\langle \boldsymbol{\varepsilon}_t^b \rangle}\right)\left(\boldsymbol{\varepsilon}_t^b - \cancel{\langle \boldsymbol{\varepsilon}_t^b \rangle}\right)^T \right\rangle = \langle \boldsymbol{\varepsilon}_t^b (\boldsymbol{\varepsilon}_t^b)^T \rangle =$$



$$tr(\mathbf{P}_t^b) = \sum_{l=1}^{n} \left(\boldsymbol{\varepsilon}_t^b\right)_l^2$$

$$\mathbf{R} \equiv \langle \boldsymbol{\varepsilon}_t^o (\boldsymbol{\varepsilon}_t^o)^T \rangle$$

P and **R** are symmetric matrices by definition.

Linear Approximations

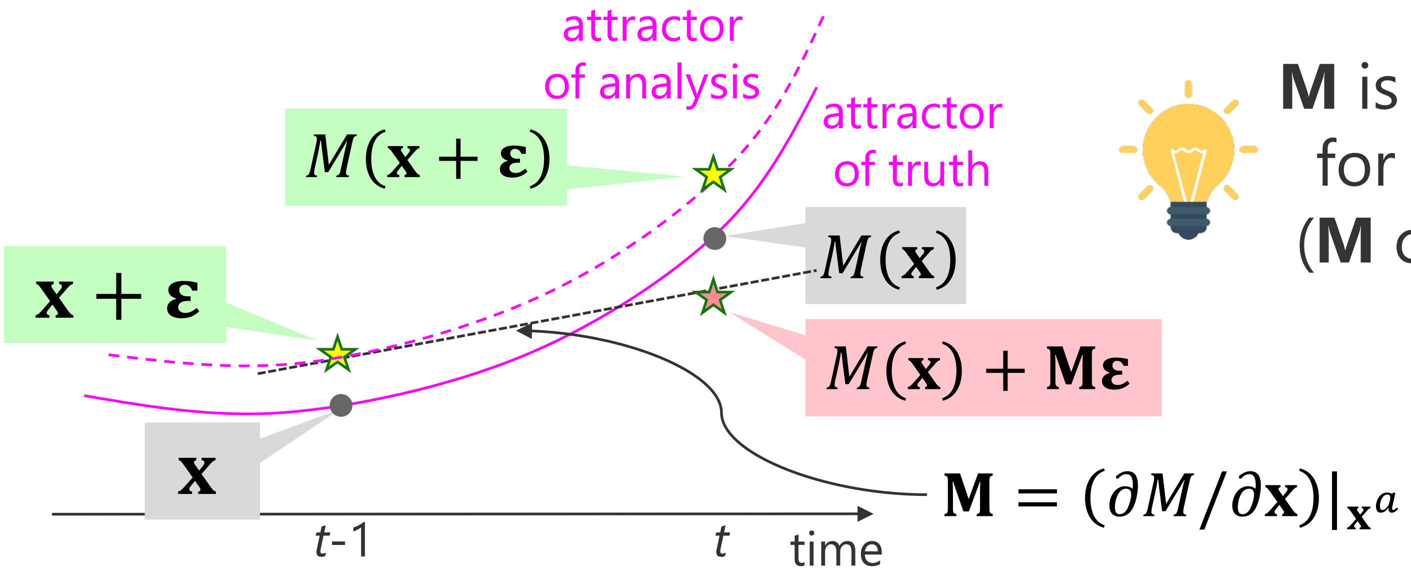
Tayler expansion (scalar)

$$f(x+\varepsilon) = f(x) + \frac{f'(x)}{1!}(\varepsilon) + \frac{f''(x)}{2!}(\varepsilon)^2 + \cdots$$

Tangent Linear Model (TLM)

$$M(\mathbf{x}+\boldsymbol{\varepsilon}) = M(\mathbf{x}) + \mathbf{M}\boldsymbol{\varepsilon} + O\left((\boldsymbol{\varepsilon})^2\right)$$

$$\approx M(\mathbf{x}) + \mathbf{M}\boldsymbol{\varepsilon} \qquad \text{where} \qquad \mathbf{M} = (\partial M/\partial \mathbf{x})|_{\mathbf{x}}$$


attractor
of analysis
attractor
of truth
$M(\mathbf{x}+\boldsymbol{\varepsilon})$
$\mathbf{x}+\boldsymbol{\varepsilon}$
$M(\mathbf{x})$
$M(\mathbf{x}) + \mathbf{M}\boldsymbol{\varepsilon}$
$\mathbf{x}$
$\mathbf{M} = (\partial M/\partial \mathbf{x})|_{\mathbf{x}^a}$
t-1
t
time

M is the linearized model for propagating errors. (**M** cannot be used to **x**)

Forecast Error Covariance

State Prediction

$$\mathbf{x}_t^{tru} = M(\mathbf{x}_{t-1}^{tru})$$

suppose that M=M^{tru}

Error Prediction

$$\boldsymbol{\varepsilon}_t^b = \mathbf{x}_t^b - \mathbf{x}_t^{tru}$$

$$= M(\mathbf{x}_{t-1}^{tru} + \boldsymbol{\varepsilon}_{t-1}^a) - M(\mathbf{x}_{t-1}^{tru})$$

$$= \cancel{M(\mathbf{x}_{t-1}^{tru})} + \mathbf{M}_{t-1}\boldsymbol{\varepsilon}_{t-1}^a + \cancel{O\left(\left(\boldsymbol{\varepsilon}_{t-1}^a\right)^2\right)} - \cancel{M(\mathbf{x}_{t-1}^{tru})}$$

$$\approx \mathbf{M}_{t-1}\boldsymbol{\varepsilon}_{t-1}^a$$

Error Covariance Prediction

$$\mathbf{P}_t^b = \left\langle \boldsymbol{\varepsilon}_t^b \left(\boldsymbol{\varepsilon}_t^b\right)^T \right\rangle \approx \mathbf{M}_{t-1}\langle \boldsymbol{\varepsilon}_{t-1}^a (\boldsymbol{\varepsilon}_{t-1}^a)^T \rangle \mathbf{M}_{t-1}^T$$

$$\mathbf{P}_t^b \approx \mathbf{M}_{t-1}\mathbf{P}_{t-1}^a\mathbf{M}_{t-1}^T$$

$$\mathbf{M}_{t-1} = (\partial M/\partial \mathbf{x})|_{\mathbf{x}_{t-1}^a}$$

Tangent Linear Model (Jacobian of *M*)

Analysis Error Covariance

$$\mathbf{x}_t^a \equiv \mathbf{x}_t^b + \mathbf{K}\left(\mathbf{y}_t^o - H(\mathbf{x}_t^b)\right)$$

$$H\left(\mathbf{x}_t^b\right) \approx H(\mathbf{x}_t^{tru}) + \mathbf{H}\boldsymbol{\varepsilon}_t^b$$

$$\mathbf{x}_t^a - \mathbf{x}_t^{tru} = \mathbf{x}_t^b - \mathbf{x}_t^{tru} + \mathbf{K}\left(\mathbf{y}_t^o - H(\mathbf{x}_t^{tru}) - \mathbf{H}\boldsymbol{\varepsilon}_t^b\right)$$

$$\Leftrightarrow \boldsymbol{\varepsilon}_t^a = \boldsymbol{\varepsilon}_t^b + \mathbf{K}\left(\boldsymbol{\varepsilon}_t^o - \mathbf{H}\boldsymbol{\varepsilon}_t^b\right)$$

$$\Leftrightarrow \boldsymbol{\varepsilon}_t^a = (\mathbf{I} - \mathbf{KH})\boldsymbol{\varepsilon}_t^b + \mathbf{K}\boldsymbol{\varepsilon}_t^o$$

No correlation b/w $\mathbf{H}\boldsymbol{\varepsilon}_t^b$ and $\boldsymbol{\varepsilon}_t^o$

$$\left\langle \mathbf{H}\boldsymbol{\varepsilon}_t^b(\boldsymbol{\varepsilon}_t^o)^T \right\rangle = \left\langle \boldsymbol{\varepsilon}_t^b(\mathbf{K}\boldsymbol{\varepsilon}_t^o)^T \right\rangle = 0$$

$$\mathbf{P}_t^a = \langle \boldsymbol{\varepsilon}_t^a(\boldsymbol{\varepsilon}_t^a)^T \rangle = (\mathbf{I} - \mathbf{KH})\left\langle \boldsymbol{\varepsilon}_t^b\left(\boldsymbol{\varepsilon}_t^b\right)^T \right\rangle(\mathbf{I} - \mathbf{KH})^T$$

$$+\mathbf{K}\langle \boldsymbol{\varepsilon}_t^o(\boldsymbol{\varepsilon}_t^o)^T \rangle\mathbf{K}^T + \cancel{\left(cross \quad term\right)}$$

$$\mathbf{P}_t^a = \langle \boldsymbol{\varepsilon}_t^a(\boldsymbol{\varepsilon}_t^a)^T \rangle = (\mathbf{I} - \mathbf{KH})\mathbf{P}_t^b(\mathbf{I} - \mathbf{KH})^T + \mathbf{KRK}^T$$

Kalman Gain

- **KF minimizes analysis error variance**

  ➔ to find **K** that minimizes trace($\mathbf{P}^a$)

$$\partial\left(tr(\mathbf{P}_t^a)\right)/\partial\mathbf{K} = 0$$

$$\Leftrightarrow \partial\left(tr\left((\mathbf{I}-\mathbf{KH})\mathbf{P}_t^b(\mathbf{I}-\mathbf{KH})^T + \mathbf{KRK}^T\right)\right)/\partial\mathbf{K} = 0$$

$$\Leftrightarrow \partial\left(tr\left(\mathbf{P}_t^b - \mathbf{KHP}_t^b - \mathbf{P}_t^b(\mathbf{KH})^T + \mathbf{KHP}_t^b(\mathbf{KH})^T + \mathbf{KRK}^T\right)\right)/\partial\mathbf{K} = 0$$

$$\Leftrightarrow 0 - \left(\mathbf{HP}_t^b\right)^T - \left(\mathbf{HP}_t^b\right)^T + 2\mathbf{KHP}_t^b\mathbf{H}^T + 2\mathbf{KR} = 0$$

$$\Leftrightarrow \mathbf{K}\left(\mathbf{HP}_t^b\mathbf{H}^T + \mathbf{R}\right) = \mathbf{P}_t^b\mathbf{H}^T$$

$$\Leftrightarrow \mathbf{K} = \mathbf{P}_t^b\mathbf{H}^T\left(\mathbf{HP}_t^b\mathbf{H}^T + \mathbf{R}\right)^{-1}$$

Eq. (1) $$\frac{\partial}{\partial\mathbf{X}}tr(\mathbf{XYX}^T) = \mathbf{X}(\mathbf{Y}+\mathbf{Y}^T)$$

Eq. (2) $$\frac{\partial}{\partial\mathbf{X}}tr(\mathbf{XY}) = \mathbf{Y}^T$$

Analysis Error Covariance

Substitute $\mathbf{K} = \mathbf{P}_t^b \mathbf{H}^T \left(\mathbf{H}\mathbf{P}_t^b \mathbf{H}^T + \mathbf{R}\right)^{-1}$

into $\mathbf{P}_t^a = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_t^b(\mathbf{I} - \mathbf{K}\mathbf{H})^T + \mathbf{K}\mathbf{R}\mathbf{K}^T$

$$
\begin{aligned}
\mathbf{P}_t^a &= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_t^b(\mathbf{I} - \mathbf{K}\mathbf{H})^T + \mathbf{K}\mathbf{R}\mathbf{K}^T \\
&= \mathbf{P}_t^b - \mathbf{K}\mathbf{H}\mathbf{P}_t^b - \left(\mathbf{K}\mathbf{H}\mathbf{P}_t^b\right)^T + \mathbf{K}\mathbf{H}\mathbf{P}_t^b\mathbf{H}^T\mathbf{K}^T + \mathbf{K}\mathbf{R}\mathbf{K}^T \\
&= \mathbf{P}_t^b - \mathbf{P}_t^b\mathbf{H}^T\mathbf{S}^{-1}\mathbf{H}\mathbf{P}_t^b - \mathbf{P}_t^b\mathbf{H}^T\mathbf{S}^{-1}\mathbf{H}\mathbf{P}_t^b + \mathbf{K}\mathbf{S}\mathbf{K}^T \\
&= \mathbf{P}_t^b - 2\mathbf{P}_t^b\mathbf{H}^T\mathbf{S}^{-1}\mathbf{H}\mathbf{P}_t^b + \mathbf{P}_t^b\mathbf{H}^T\mathbf{S}^{-1}\mathbf{H}\mathbf{P}_t^b \\
&= \mathbf{P}_t^b - \mathbf{P}_t^b\mathbf{H}^T\mathbf{S}^{-1}\mathbf{H}\mathbf{P}_t^b \\
&= (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_t^b
\end{aligned}
$$

where $\mathbf{S} = \left(\mathbf{H}\mathbf{P}_t^b\mathbf{H}^T + \mathbf{R}\right)$

Kalman Filter

Prediction (state)

$$\mathbf{x}_t^b = \underbrace{M}_{\text{nonlinear model}}(\mathbf{x}_{t-1}^a)$$

Prediction (error covariance)

$$\mathbf{P}_t^b = \underbrace{\mathbf{M}\mathbf{P}_{t-1}^a\mathbf{M}^T}_{\text{linearized model}} + \mathbf{Q}$$

Kalman gain

$$\mathbf{K}_t = \mathbf{P}_t^b\mathbf{H}^T\left[\mathbf{H}\mathbf{P}_t^b\mathbf{H}^T + \mathbf{R}\right]^{-1}$$

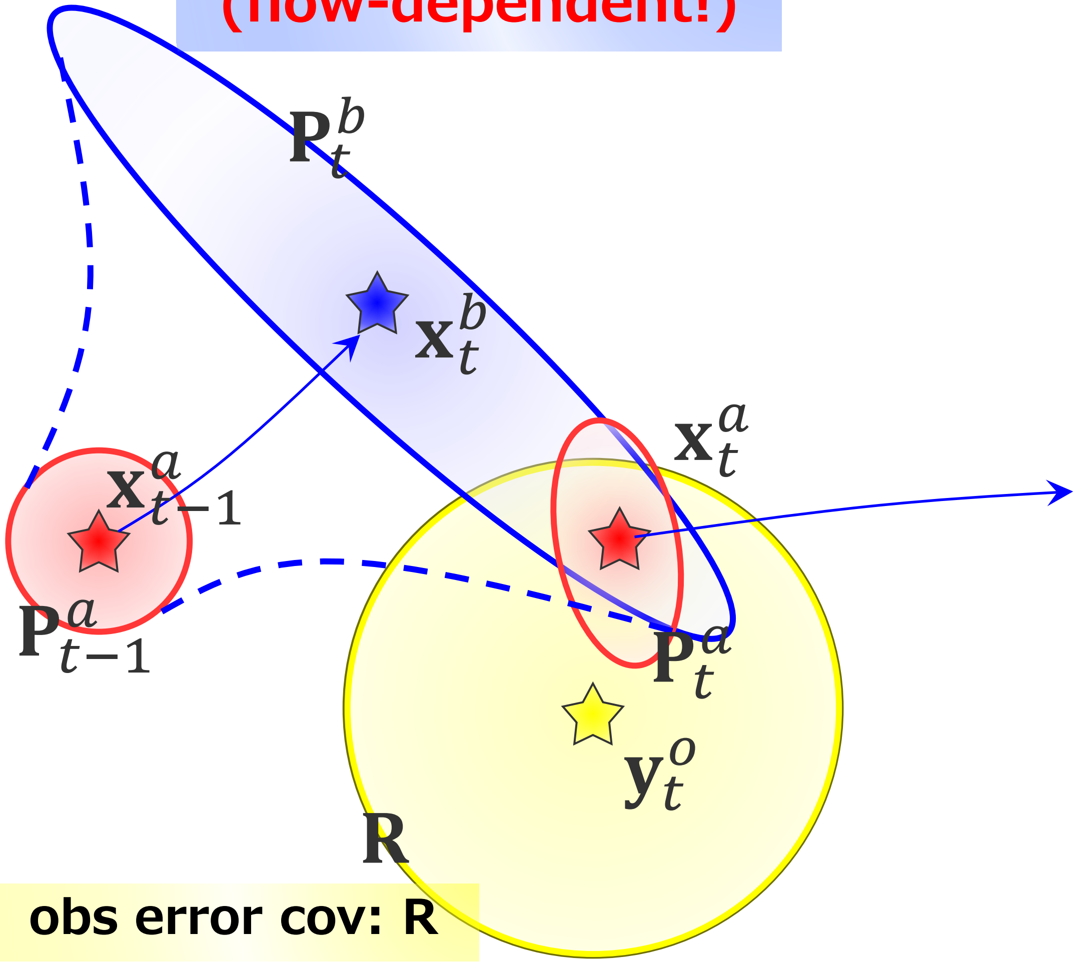
Analysis (state)

$$\mathbf{x}_t^a = \mathbf{x}_t^b + \mathbf{K}_t(\mathbf{y}_t^o - H(\mathbf{x}_t^b))$$

Analysis (error covariance)

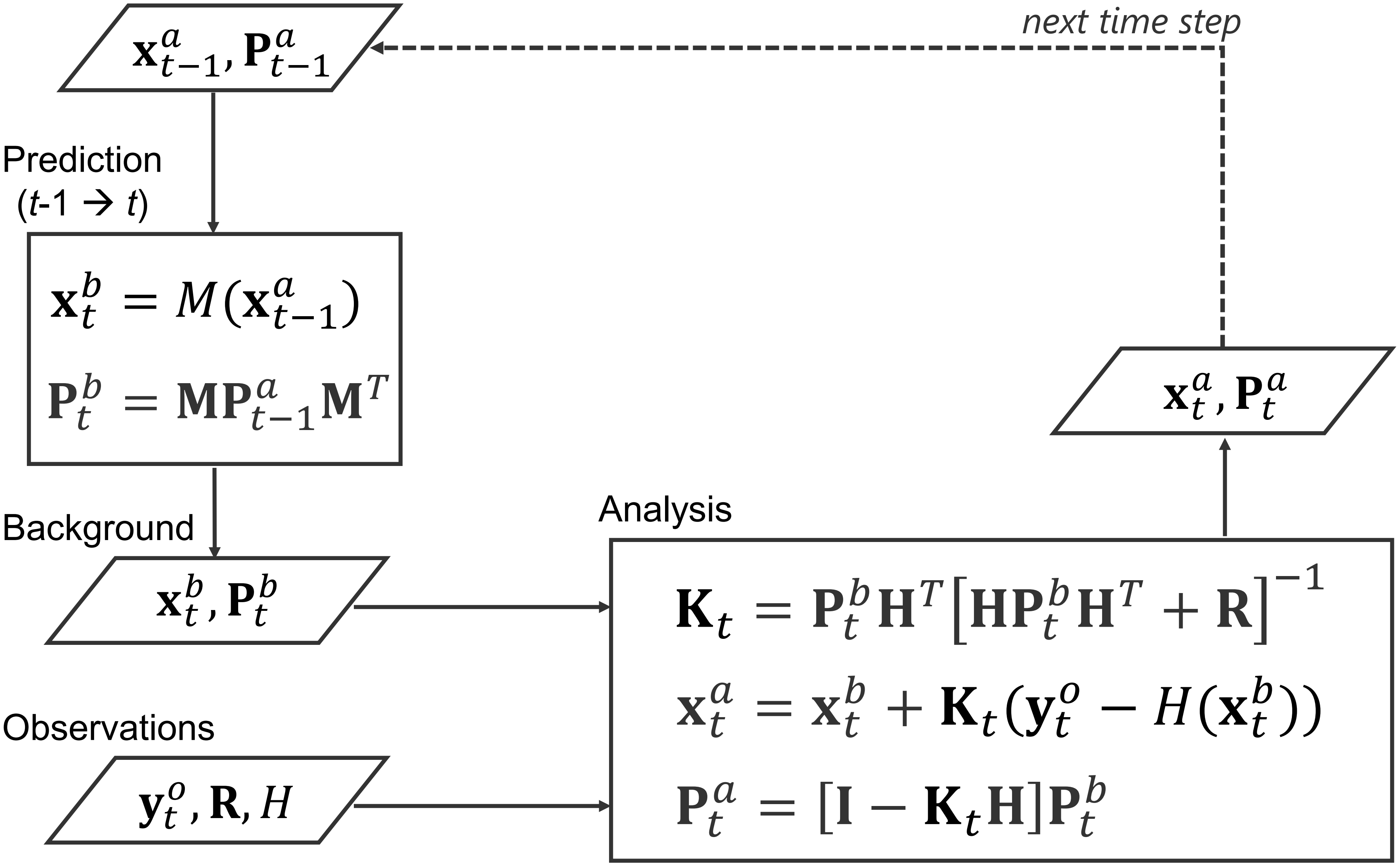
$$\mathbf{P}_t^a = [\mathbf{I} - \mathbf{K}_t\mathbf{H}]\mathbf{P}_t^b$$

Prior Error Cov. $\mathbf{P}^b$
(flow-dependent!)



obs error cov: R

Kalman Filter Algorithm



$\mathbf{x}_{t-1}^{a}, \mathbf{P}_{t-1}^{a}$

Prediction
(t-1 → t)

$$\mathbf{x}_t^b = M(\mathbf{x}_{t-1}^a)$$

$$\mathbf{P}_t^b = \mathbf{M}\mathbf{P}_{t-1}^a\mathbf{M}^T$$

Background

$\mathbf{x}_t^b, \mathbf{P}_t^b$

Observations

$\mathbf{y}_t^o, \mathbf{R}, H$

Analysis

$$\mathbf{K}_t = \mathbf{P}_t^b\mathbf{H}^T\left[\mathbf{H}\mathbf{P}_t^b\mathbf{H}^T + \mathbf{R}\right]^{-1}$$

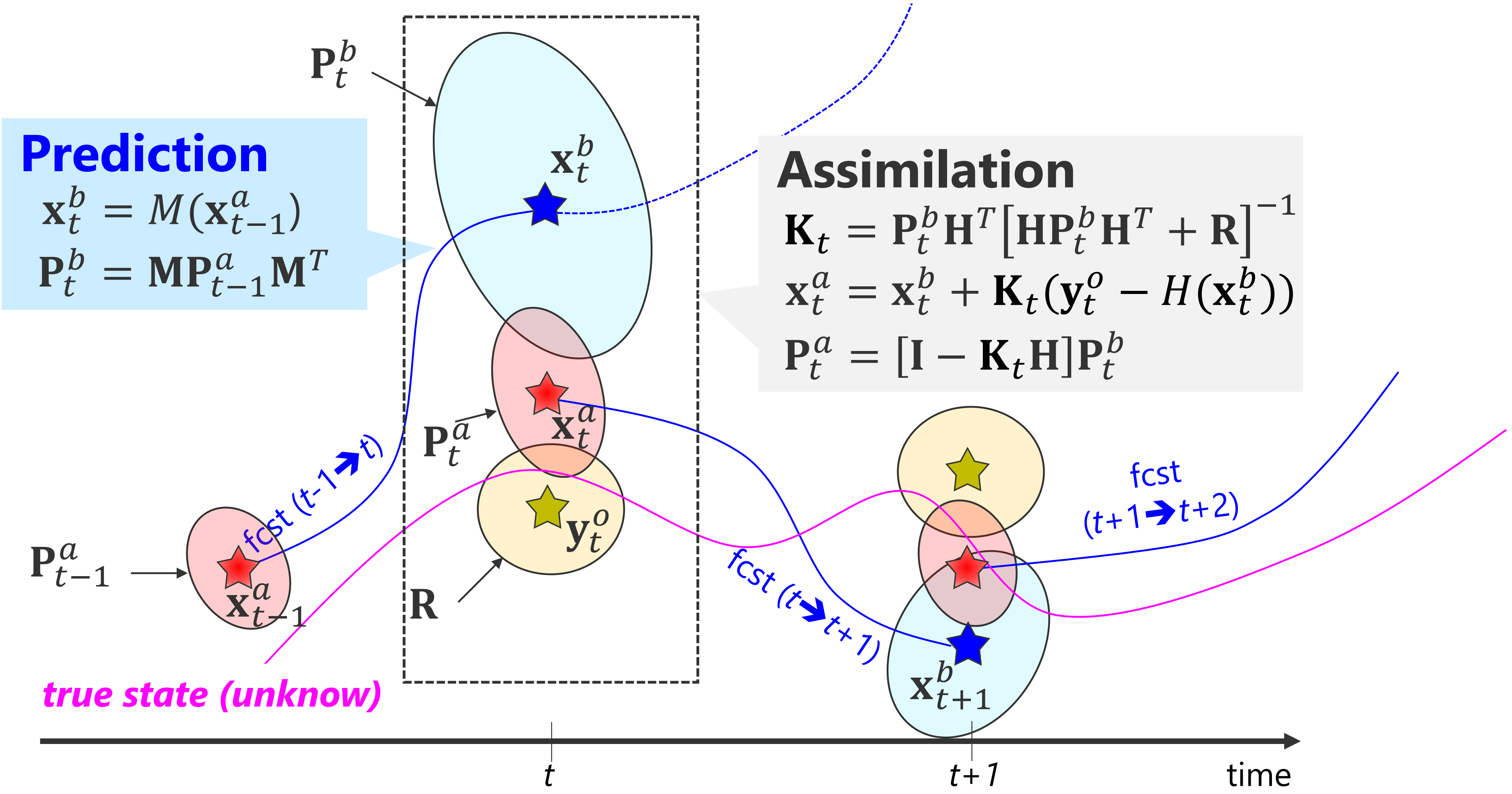
$$\mathbf{x}_t^a = \mathbf{x}_t^b + \mathbf{K}_t(\mathbf{y}_t^o - H(\mathbf{x}_t^b))$$

$$\mathbf{P}_t^a = [\mathbf{I} - \mathbf{K}_t\mathbf{H}]\mathbf{P}_t^b$$

$\mathbf{x}_t^a, \mathbf{P}_t^a$

next time step

Sequential Kalman Filter



Prediction

$$\mathbf{x}_t^b = M(\mathbf{x}_{t-1}^a)$$

$$\mathbf{P}_t^b = \mathbf{M}\mathbf{P}_{t-1}^a\mathbf{M}^T$$

Assimilation

$$\mathbf{K}_t = \mathbf{P}_t^b\mathbf{H}^T\left[\mathbf{H}\mathbf{P}_t^b\mathbf{H}^T + \mathbf{R}\right]^{-1}$$

$$\mathbf{x}_t^a = \mathbf{x}_t^b + \mathbf{K}_t(\mathbf{y}_t^o - H(\mathbf{x}_t^b))$$

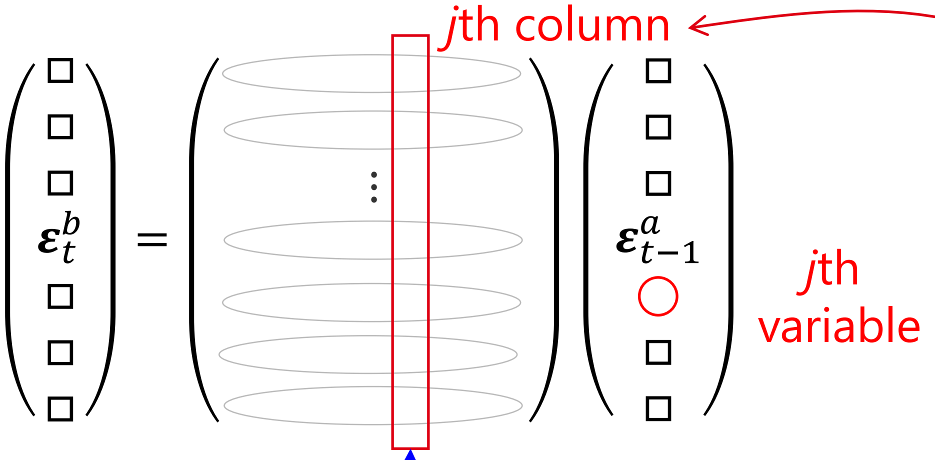
$$\mathbf{P}_t^a = [\mathbf{I} - \mathbf{K}_t\mathbf{H}]\mathbf{P}_t^b$$

★ : forecast ★ : observation ★ : analysis (**DA**) ● : error cov. (accuracy)

Tangent Linear Model (Numerical)

Require: to get M such that $\boldsymbol{\varepsilon}_t^b = \mathbf{M}\boldsymbol{\varepsilon}_{t-1}^a$

$$\mathbf{M} = (\partial M/\partial \mathbf{x})|_{\mathbf{x}^a}$$


*j*th column
$\boldsymbol{\varepsilon}_t^b$
=
⋮
$\boldsymbol{\varepsilon}_{t-1}^a$
*j*th variable

*j*th column of **M** describes how error of *j*th variable propagates

$$M(\mathbf{x}_t^a + \delta\mathbf{e}_j) \approx M(\mathbf{x}_t^a) + \mathbf{M}\delta\mathbf{e}_j \quad \textit{where} \quad \mathbf{e}_j = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

*j*th variable

$$\Leftrightarrow \mathbf{M}\mathbf{e}_j \approx \frac{M(\mathbf{x}_t^a + \delta\mathbf{e}_j) - M(\mathbf{x}_t^a)}{\delta}$$

*j*th element of **M**

computable

$$\delta \ll 1$$
$$(e.g.\,10^{-5})$$

- repeat these steps for *j*=1,…,*n* (e.g. *n*=40 for L96)

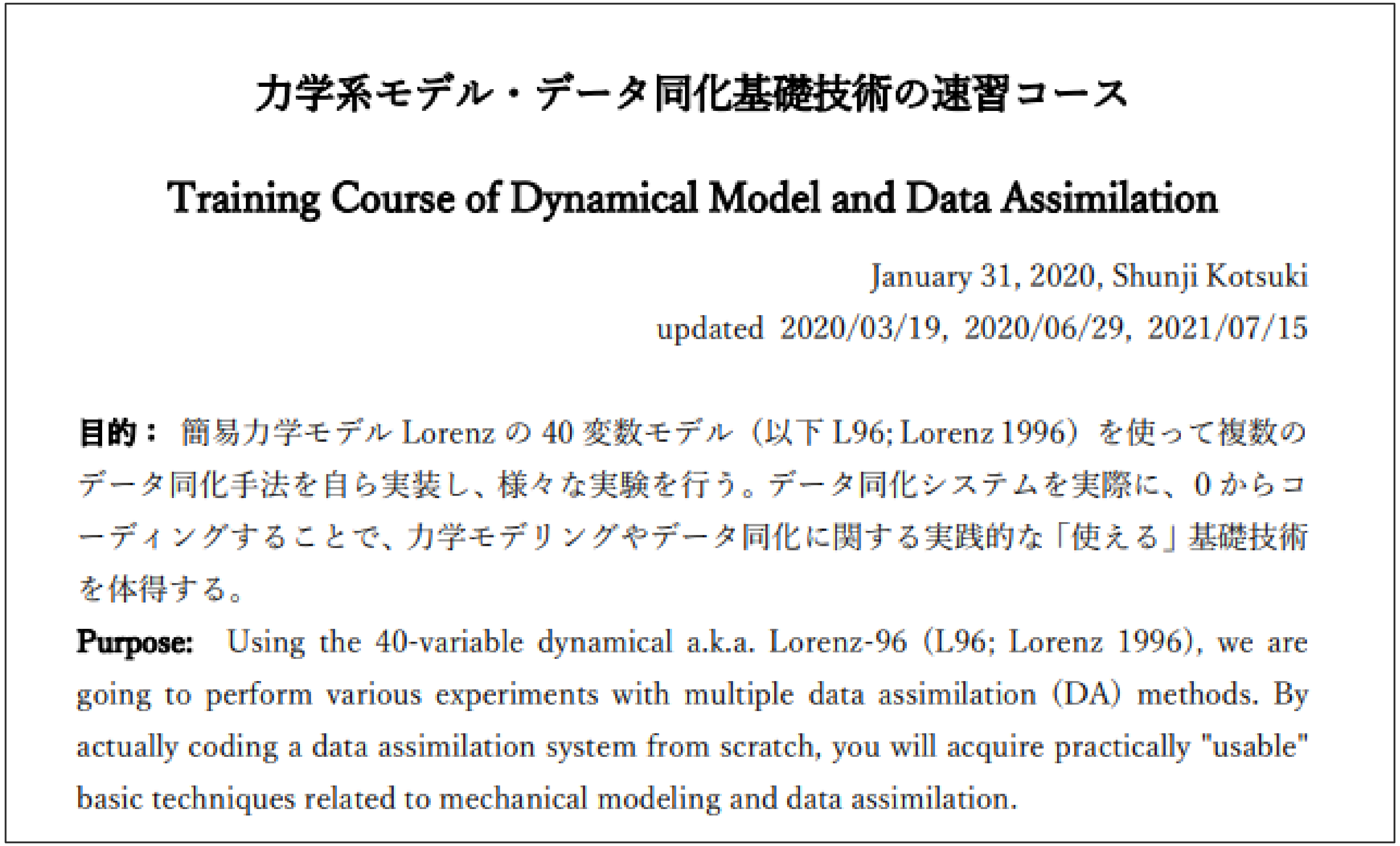
Training Course

DA Study w/ 40-variable Lorenz-96

Lorenz-96 model (Lorenz 1996)

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

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



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

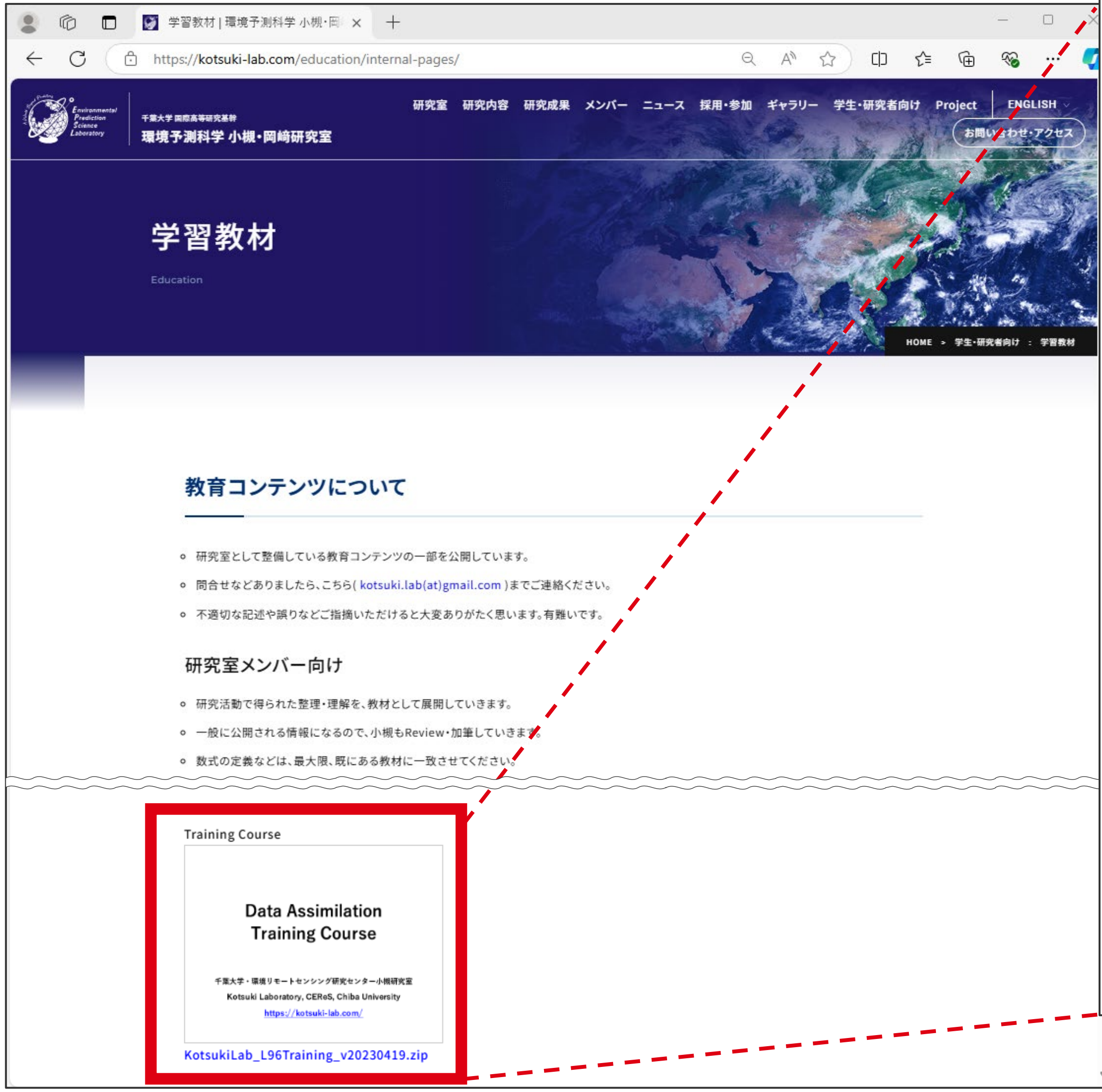
目的： 簡易力学モデルLorenzの40変数モデル（以下L96; Lorenz 1996）を使って複数のデータ同化手法を自ら実装し、様々な実験を行う。データ同化システムを実際に、0からコーディングすることで、力学モデリングやデータ同化に関する実践的な「使える」基礎技術を体得する。

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.

Text Books

① Training Description

pswd: ceres


学習教材 | 環境予測科学 小槻・岡
https://kotsuki-lab.com/education/internal-pages/
Environmental Prediction Science Laboratory
千葉大学 国際高等研究基幹
環境予測科学 小槻・岡崎研究室
研究室
研究内容
研究成果
メンバー
ニュース
採用・参加
ギャラリー
学生・研究者向け
Project
ENGLISH
お問い合わせ・アクセス
HOME > 学生・研究者向け : 学習教材

学習教材

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)までご連絡ください。
- 不適切な記述や誤りなどご指摘いただけると大変ありがたく思います。有難いです。

研究室メンバー向け

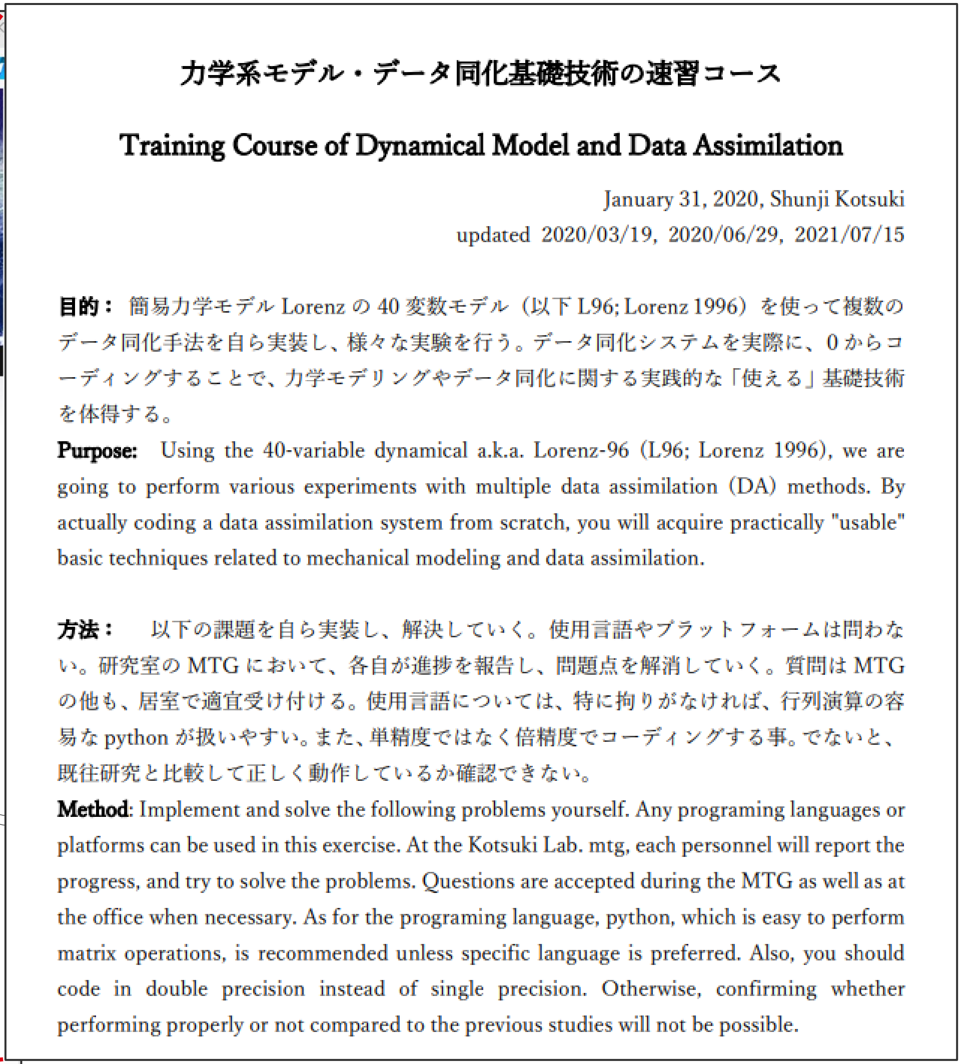
- 研究活動で得られた整理・理解を、教材として展開していきます。
- 一般に公開される情報になるので、小槻もReview・加筆していきます。
- 数式の定義などは、最大限、既にある教材に一致させてください。

Training Course

Data Assimilation
Training Course

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/

KotsukiLab_L96Training_v20230419.zip



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.

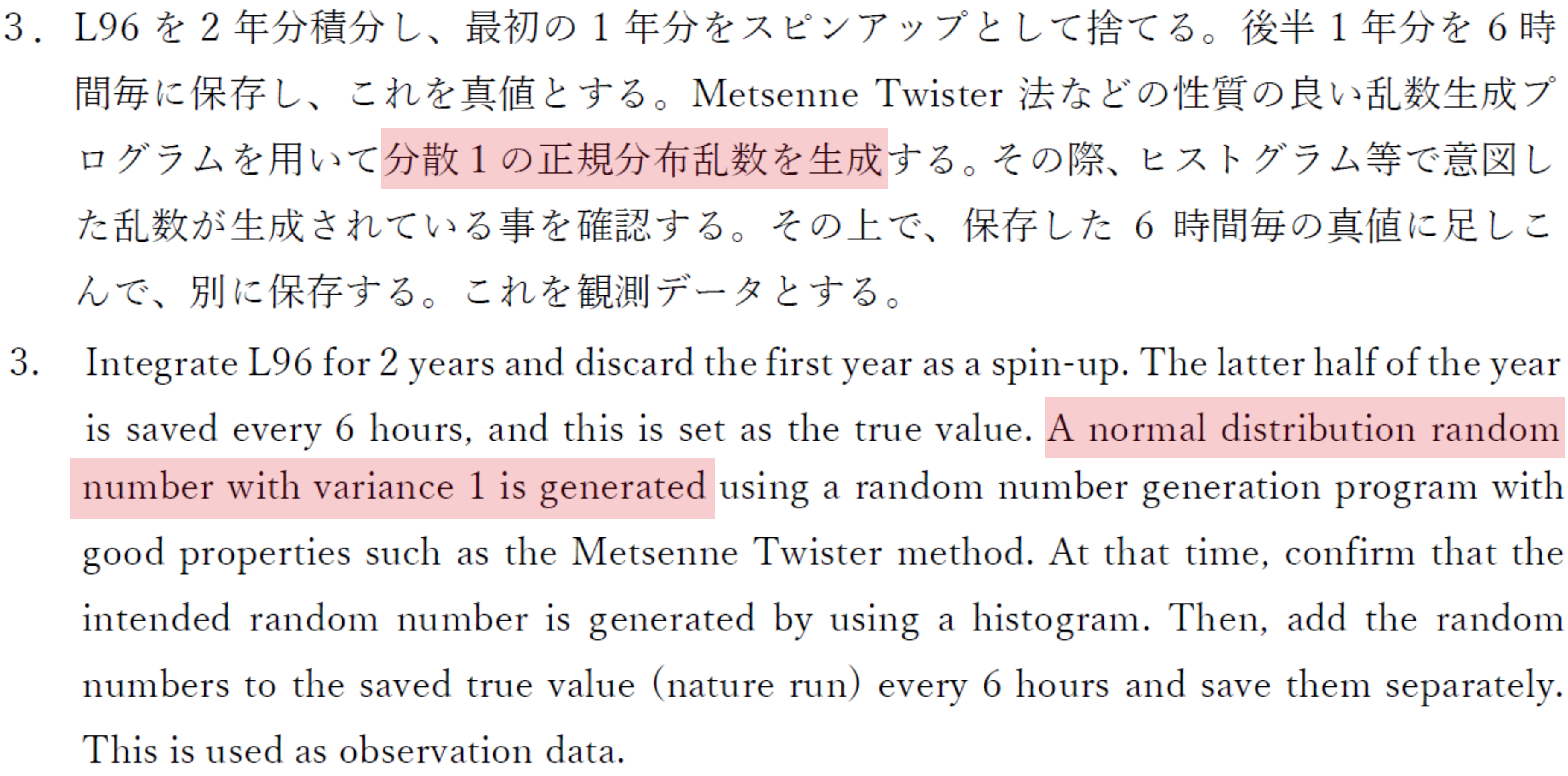
方法： 以下の課題を自ら実装し、解決していく。使用言語やプラットフォームは問わない。研究室の MTG において、各自が進捗を報告し、問題点を解消していく。質問は MTG の他も、居室で適宜受け付ける。使用言語については、特に拘りがなければ、行列演算の容易な python が扱いやすい。また、単精度ではなく倍精度でコーディングする事。でないと、既往研究と比較して正しく動作しているか確認できない。

Method: Implement and solve the following problems yourself. Any programing 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 programing 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/

Basic Task 3

Basic Task 3



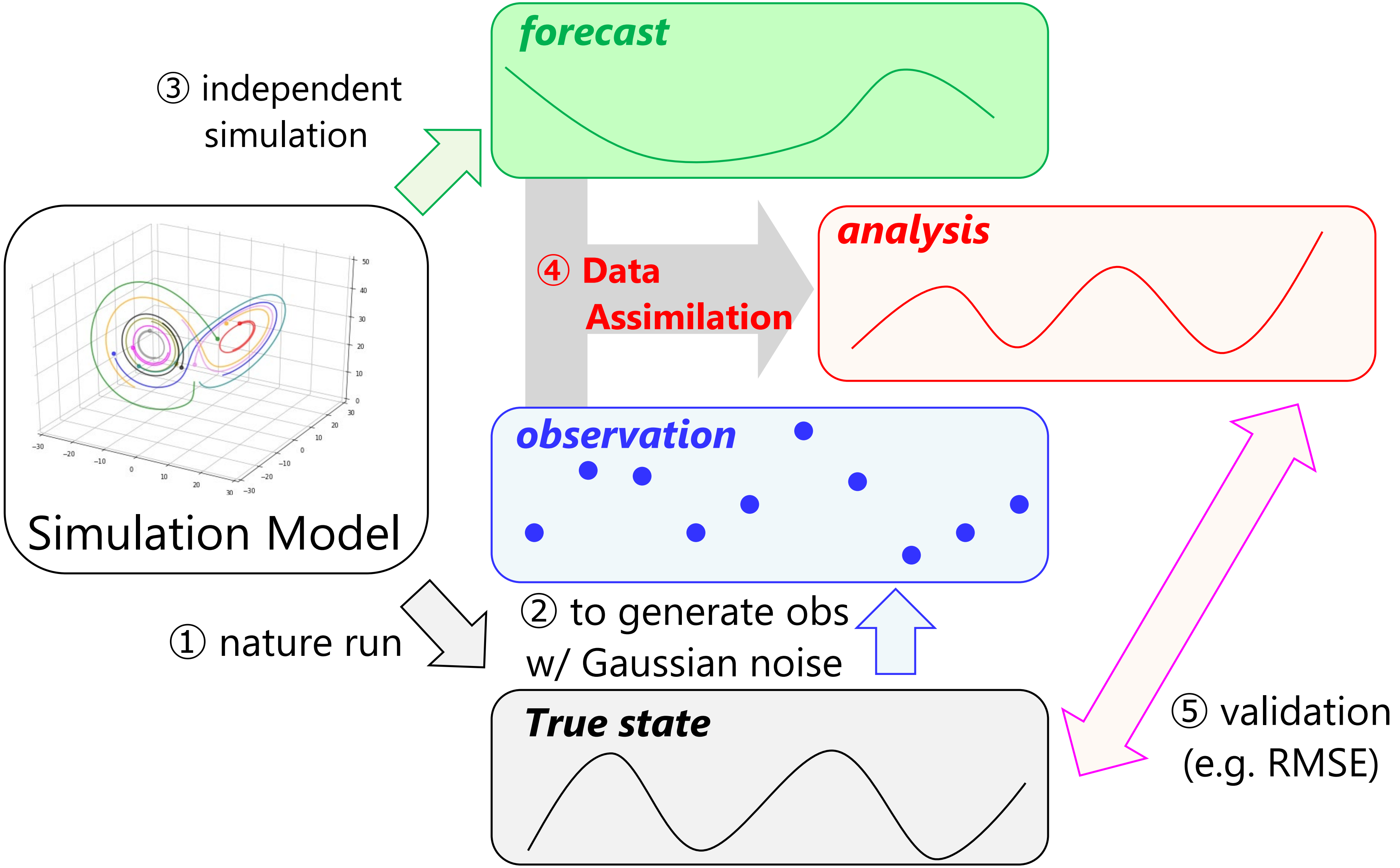
3．L96を2年分積分し、最初の1年分をスピンアップとして捨てる。後半1年分を6時間毎に保存し、これを真値とする。Metsenne Twister法などの性質の良い乱数生成プログラムを用いて分散1の正規分布乱数を生成する。その際、ヒストグラム等で意図した乱数が生成されている事を確認する。その上で、保存した6時間毎の真値に足しこんで、別に保存する。これを観測データとする。

3. Integrate L96 for 2 years and discard the first year as a spin-up. The latter half of the year is saved every 6 hours, and this is set as the true value. A normal distribution random number with variance 1 is generated using a random number generation program with good properties such as the Metsenne Twister method. At that time, confirm that the intended random number is generated by using a histogram. Then, add the random numbers to the saved true value (nature run) every 6 hours and save them separately. This is used as observation data.

This means the experiments assume **R** to be **I** (i.e., identity matrix)

OSSE: Observing Sys. Sim. Experiment

also known as <u>Idealized Twin Experiment</u>



forecast

③ independent simulation

analysis

④ Data Assimilation

observation

Simulation Model

① nature run

② to generate obs w/ Gaussian noise

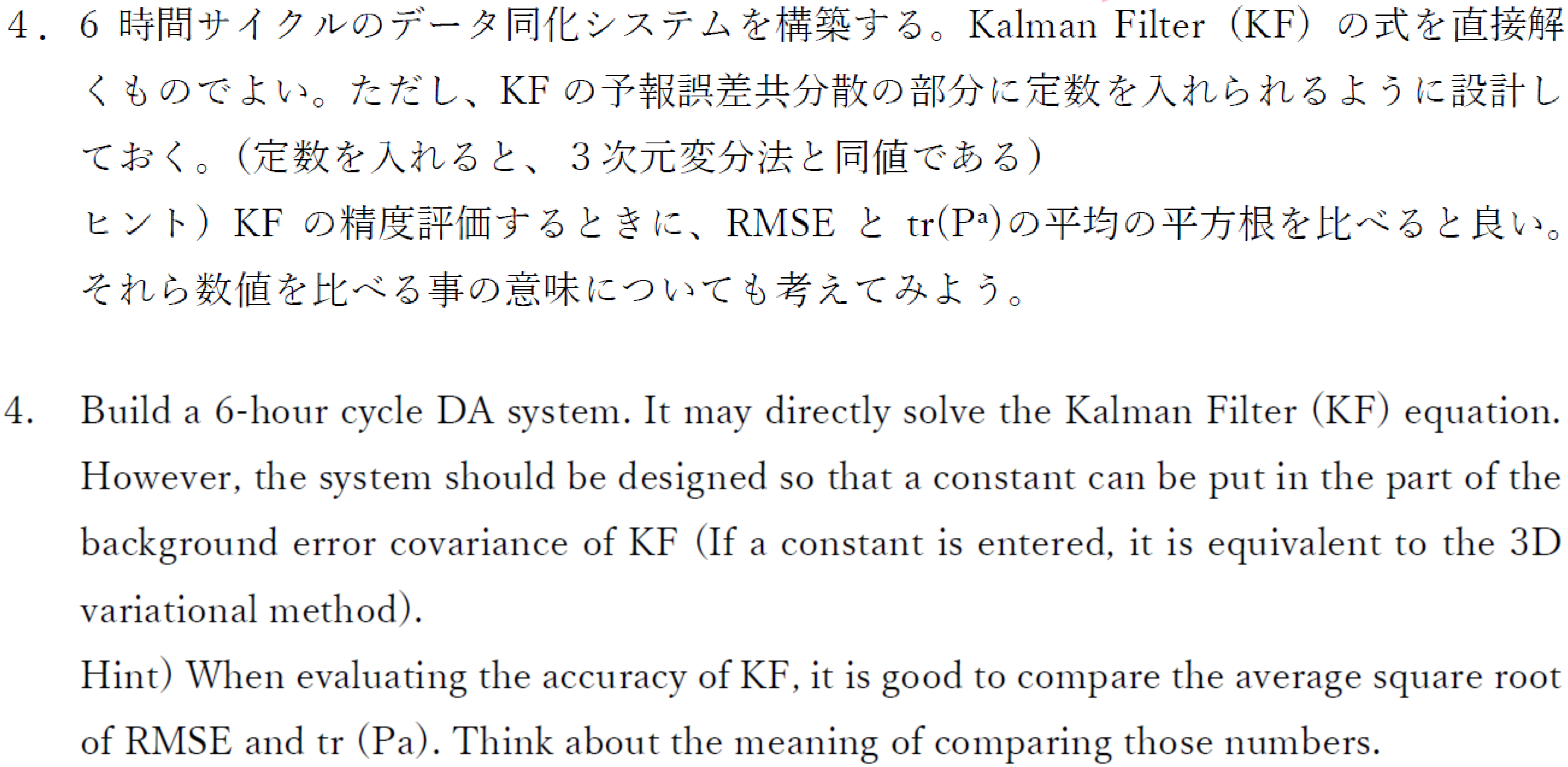
True state

⑤ validation (e.g. RMSE)

Basic Task 4

Basic Task 4

An additional treatment
will be needed.
Let's think about by your self.



4．6時間サイクルのデータ同化システムを構築する。Kalman Filter（KF）の式を直接解くものでよい。ただし、KFの予報誤差共分散の部分に定数を入れられるように設計しておく。（定数を入れると、３次元変分法と同値である）

ヒント）KFの精度評価するときに、RMSEとtr(P^a)の平均の平方根を比べると良い。それら数値を比べる事の意味についても考えてみよう。

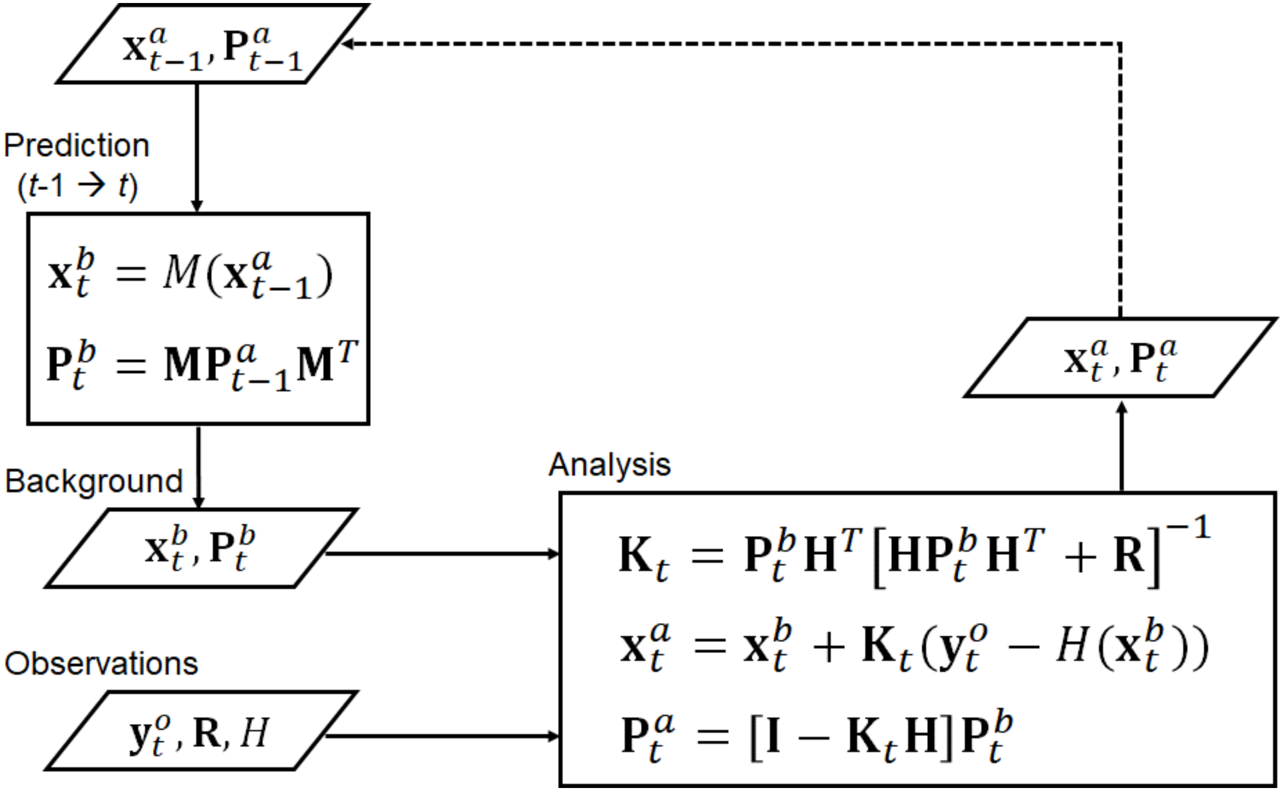
4. Build a 6-hour cycle DA system. It may directly solve the Kalman Filter (KF) equation. However, the system should be designed so that a constant can be put in the part of the background error covariance of KF (If a constant is entered, it is equivalent to the 3D variational method).

Hint) When evaluating the accuracy of KF, it is good to compare the average square root of RMSE and tr (Pa). Think about the meaning of comparing those numbers.

KF (also known as Extended KF)

Initial Condition

- $\mathbf{x}_0^a$
  - ➔ randomly chosen from nature run in spin up
- $\mathbf{P}_0^a$
  - ➔ should be large (e.g. $10.0 \times \mathbf{I}$)



$\mathbf{x}_{t-1}^a, \mathbf{P}_{t-1}^a$

Prediction
(t-1 → t)

$$\mathbf{x}_t^b = M(\mathbf{x}_{t-1}^a)$$

$$\mathbf{P}_t^b = \mathbf{M}\mathbf{P}_{t-1}^a\mathbf{M}^T$$

Background

$\mathbf{x}_t^b, \mathbf{P}_t^b$

Observations

$\mathbf{y}_t^o, \mathbf{R}, H$

Analysis

$$\mathbf{K}_t = \mathbf{P}_t^b\mathbf{H}^T\left[\mathbf{H}\mathbf{P}_t^b\mathbf{H}^T + \mathbf{R}\right]^{-1}$$

$$\mathbf{x}_t^a = \mathbf{x}_t^b + \mathbf{K}_t(\mathbf{y}_t^o - H(\mathbf{x}_t^b))$$

$$\mathbf{P}_t^a = [\mathbf{I} - \mathbf{K}_t\mathbf{H}]\mathbf{P}_t^b$$

$\mathbf{x}_t^a, \mathbf{P}_t^a$

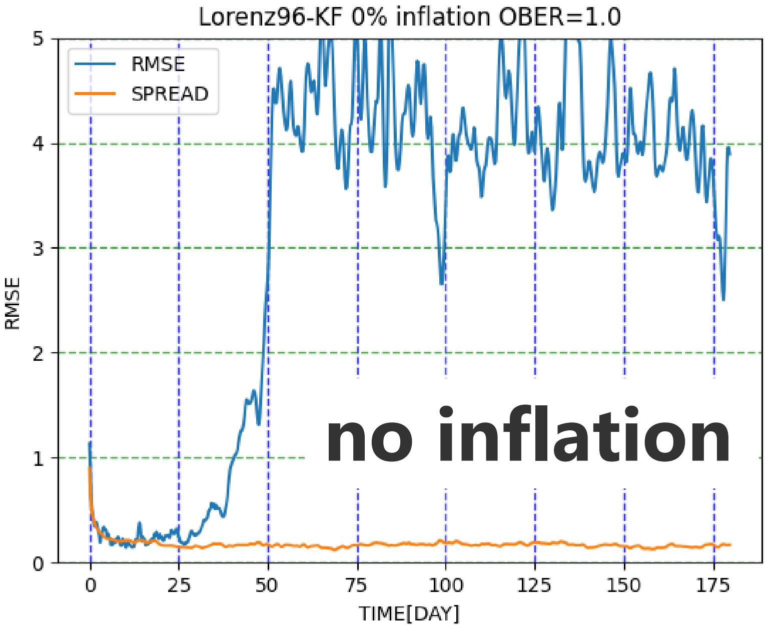
Variance Inflation (KF)

Empirical treatment for variance underestimation due to

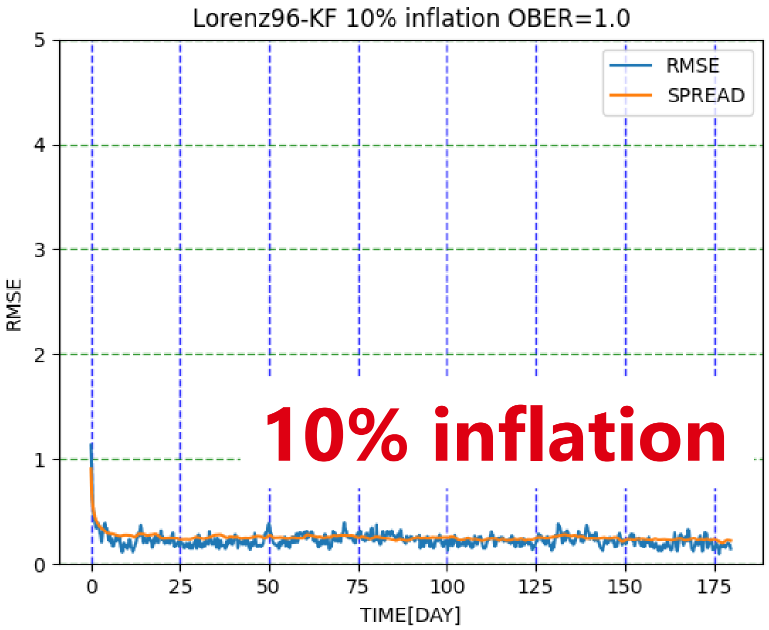
(1) limited ensemble size
(2) model nonlinearity
(3) model imperfection

$$\mathbf{P}^b_{inf} = (1+\delta)\times \mathbf{P}^b$$

inflation factor (a tuning parameter)


Lorenz96-KF 0% inflation OBER=1.0

no inflation


Lorenz96-KF 10% inflation OBER=1.0

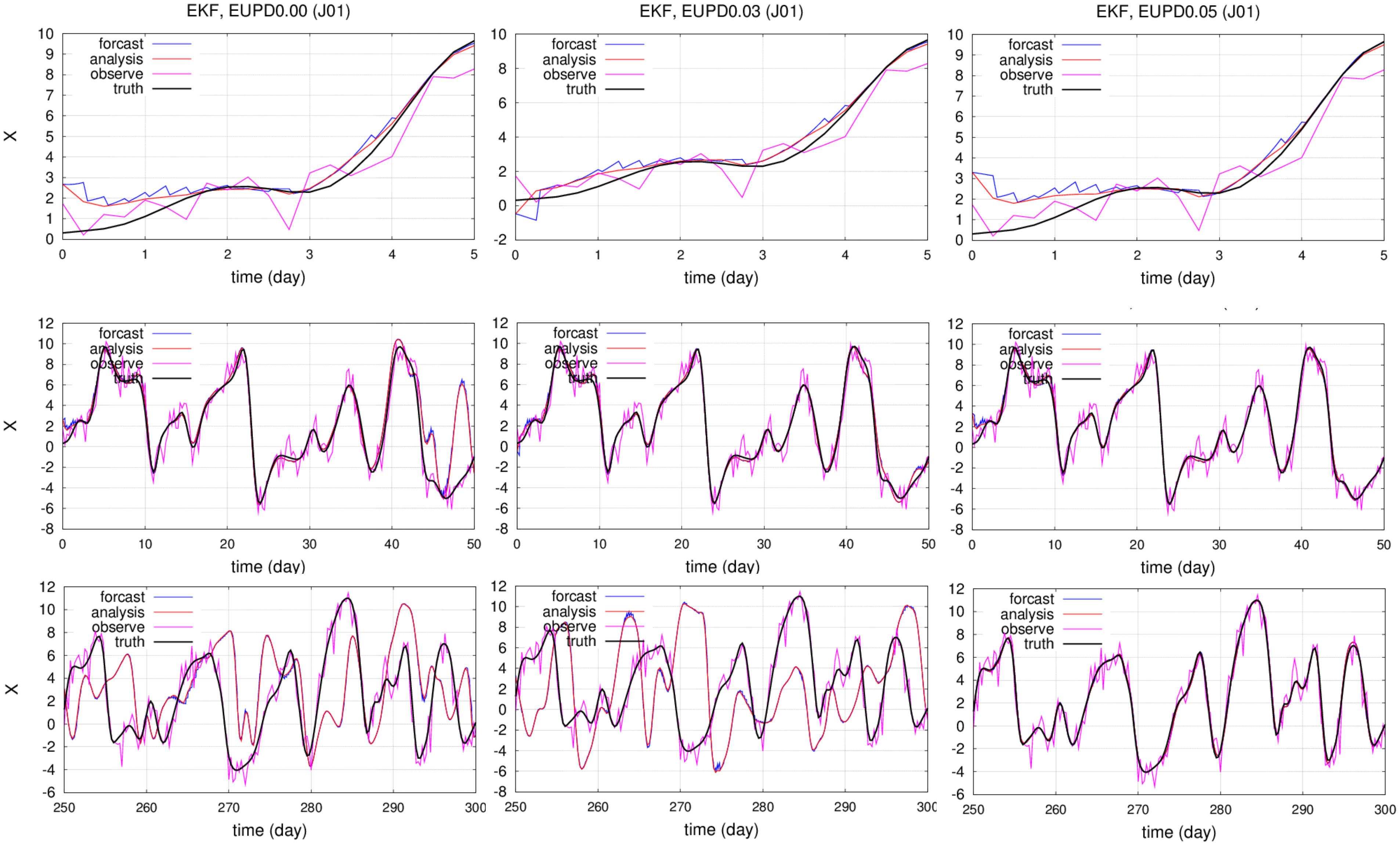
10% inflation

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

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

First Variable *X*(1) as a func. of time

$\delta = 0.00$ | $\delta = 0.03$ | $\delta = 0.05$



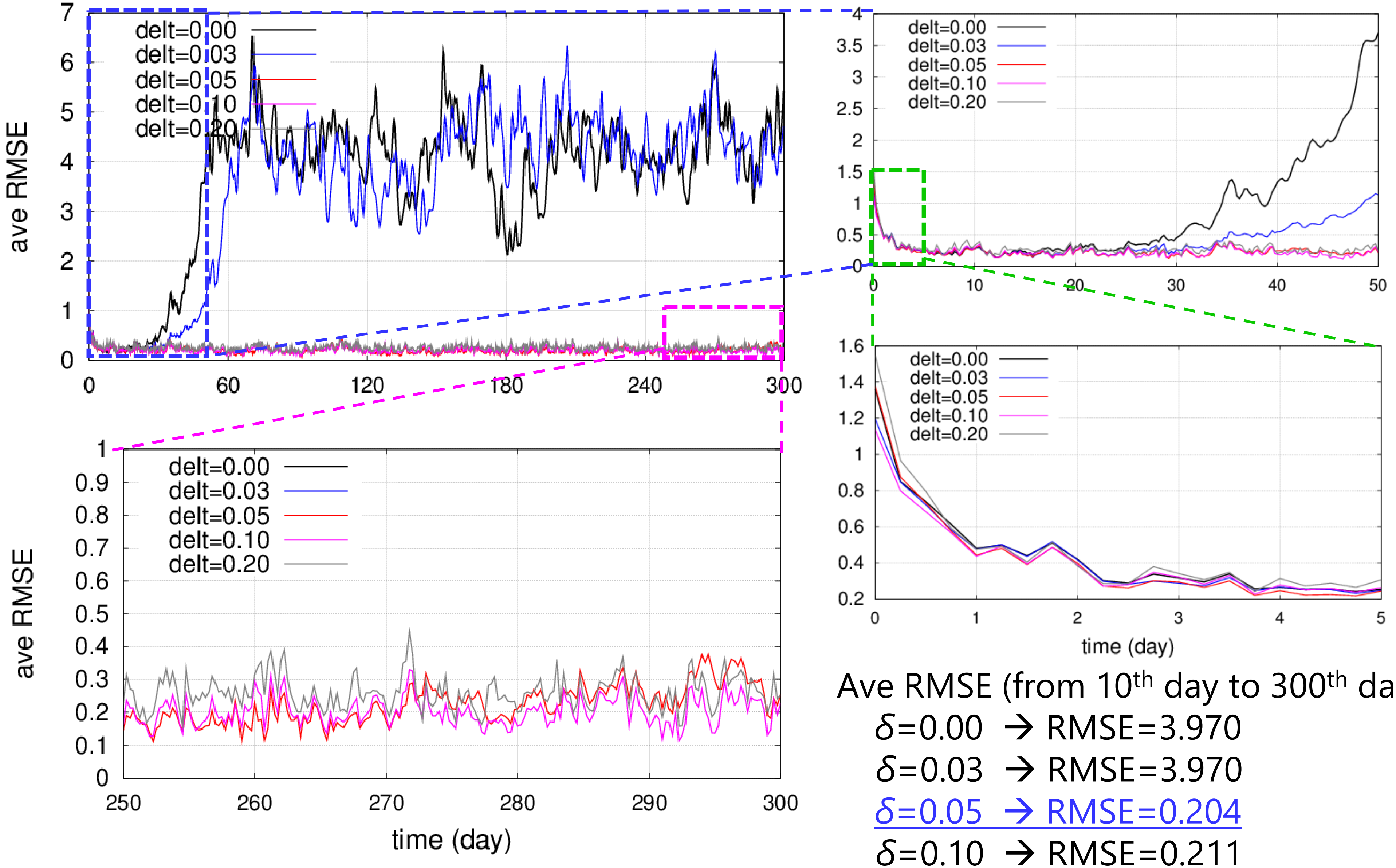
—: forecast

—: analysis

—: observation

—: truth

Analysis RMSE



Ave RMSE (from 10th day to 300th day)

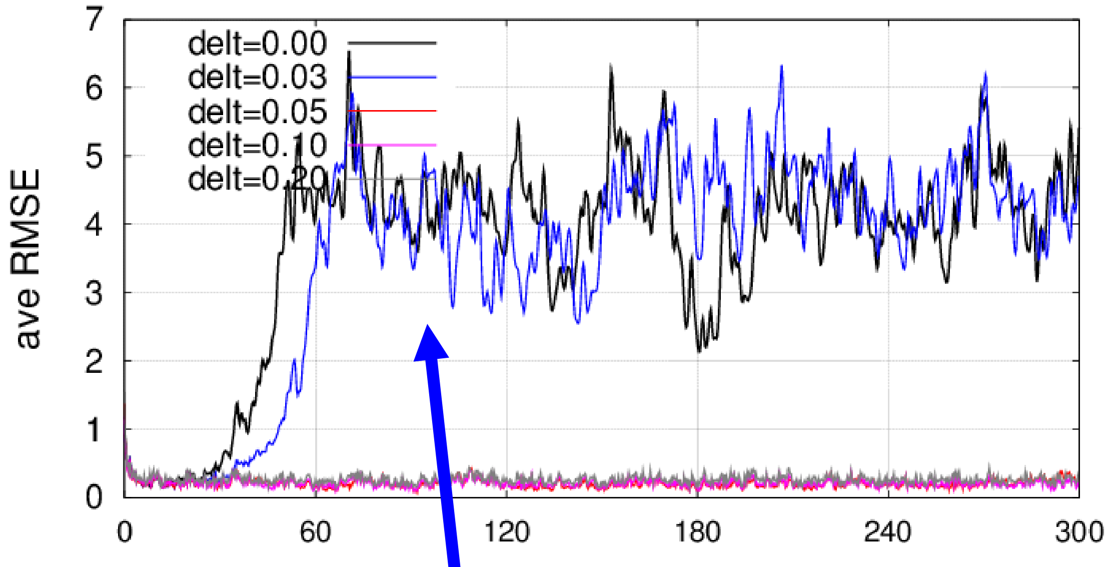
δ=0.00 → RMSE=3.970

δ=0.03 → RMSE=3.970

δ=0.05 → RMSE=0.204

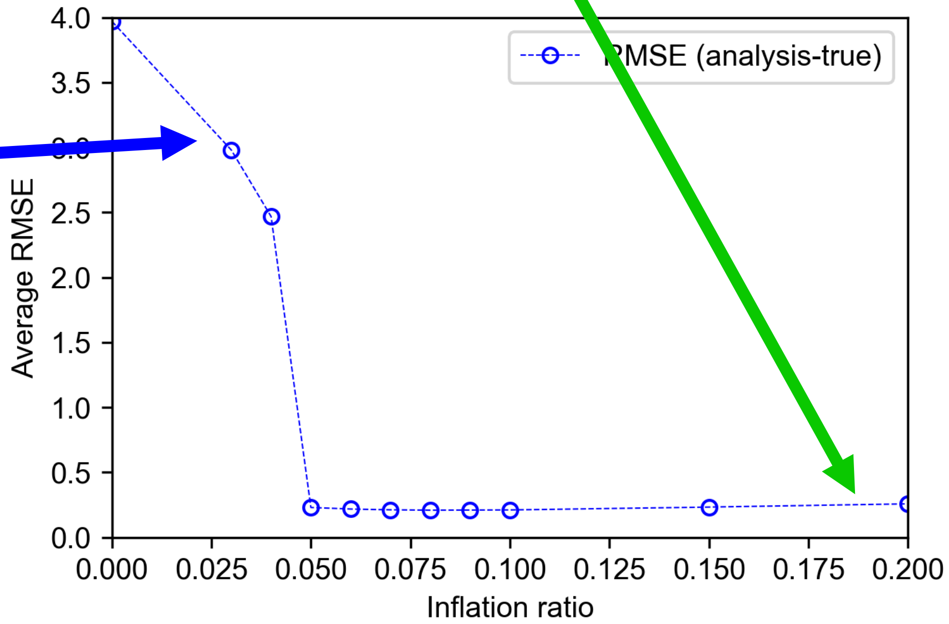
δ=0.10 → RMSE=0.211

Sensitivity to Infl. Factor



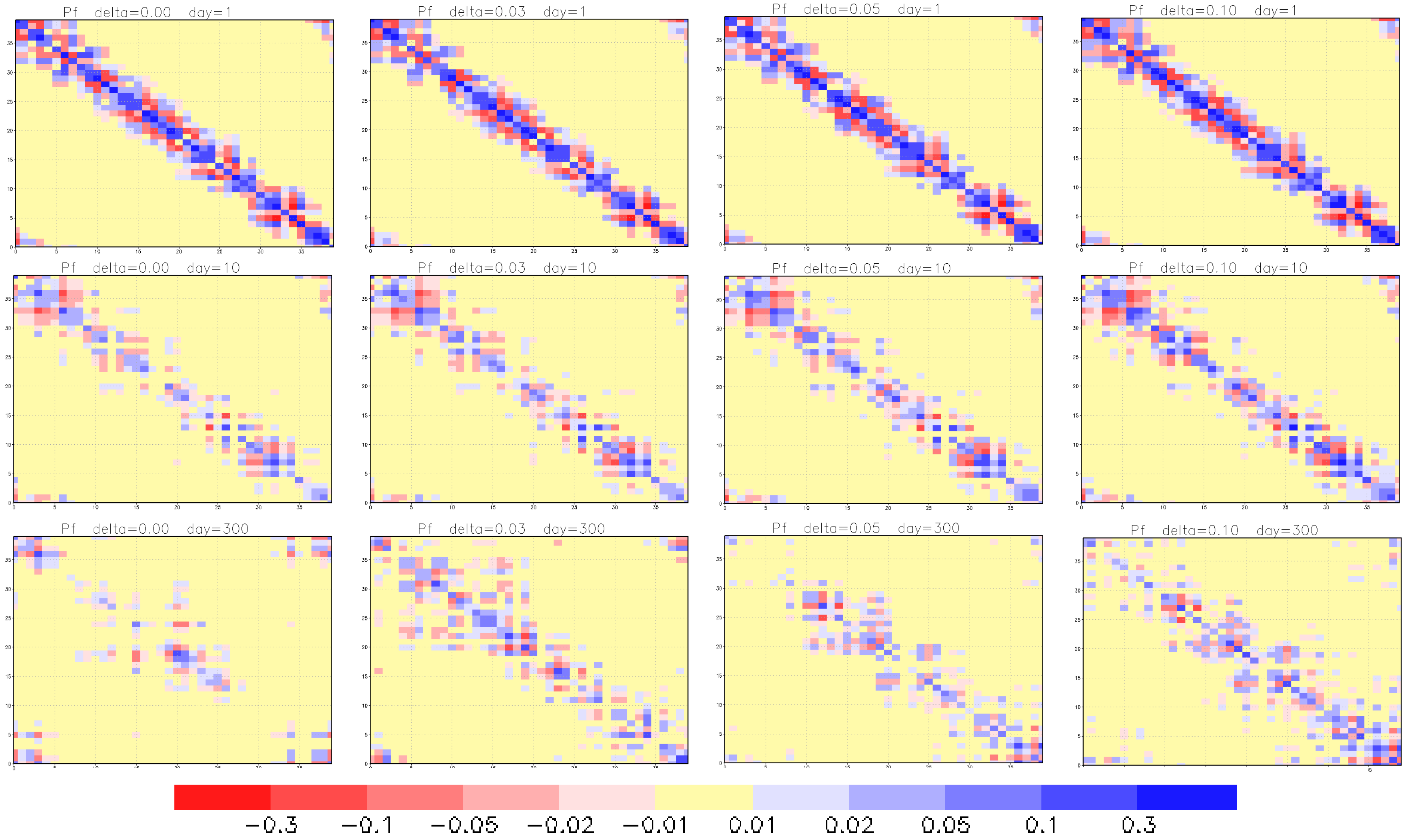
too large inflation degrades gradually

too small inflation causes filter divergence



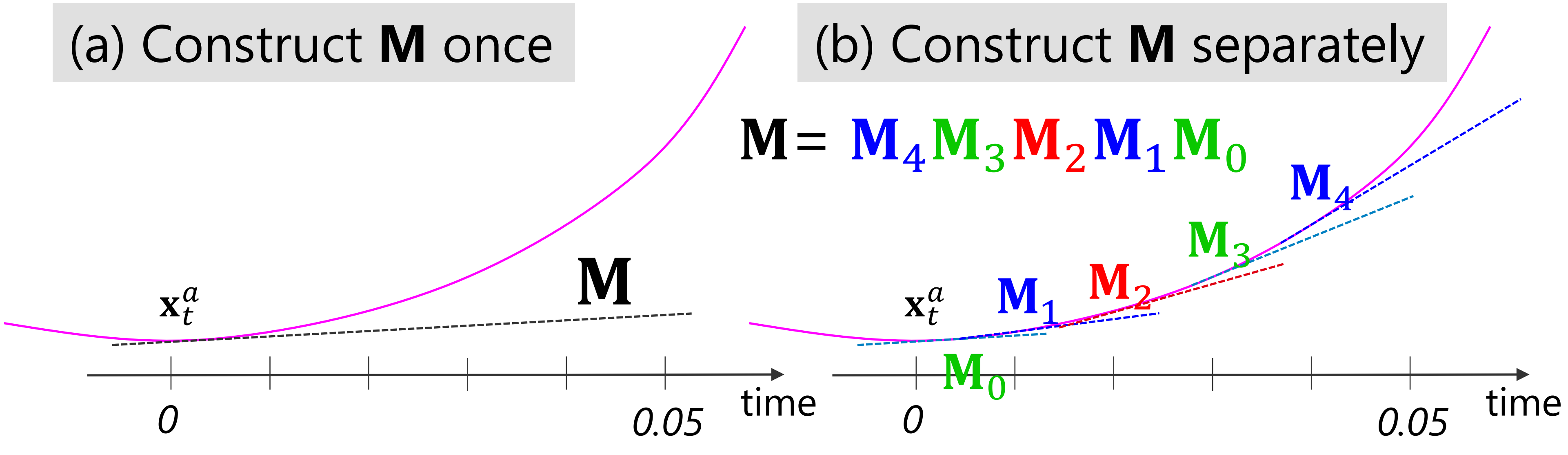
FCST Error Covariance $\mathbf{P}_t^b$

$\delta = 0.00$ $\delta = 0.03$ $\delta = 0.05$ $\delta = 0.10$



Tips

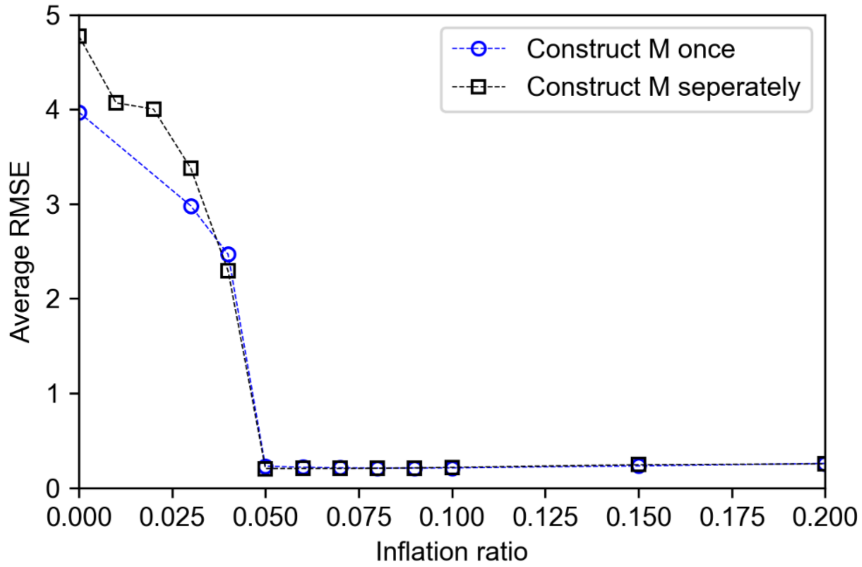
(1) Splitting M into sub Ms

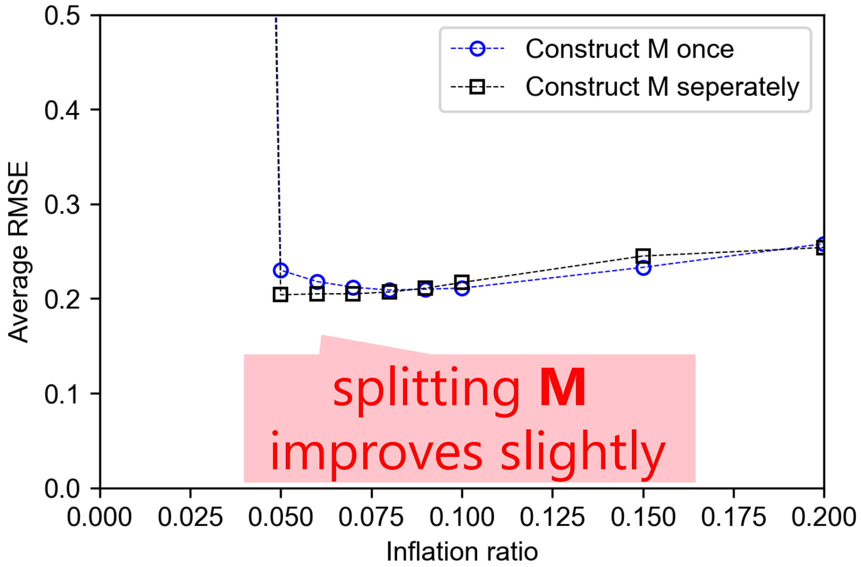


(a) Construct **M** once

(b) Construct **M** separately

$$\mathbf{M} = \mathbf{M}_4\mathbf{M}_3\mathbf{M}_2\mathbf{M}_1\mathbf{M}_0$$





splitting **M** improves slightly

(2) Alternative Way of M

(1) $$\frac{dX_j}{dt} = (X_{j+1} - X_{j-2})X_{j-1} - X_j + F$$

(2) $$\frac{d(X_j + \delta X_j)}{dt} = (X_{j+1} + \delta X_{j+1})(X_{j-1} + \delta X_{j-1}) - (X_{j-2} + \delta X_{j-2})(X_{j-1} + \delta X_{j-1}) - (X_j + \delta X_j) + F$$

(2) – (1) & ignore second order terms gives

$\delta X_j = \delta X_j(t)$

$$\frac{d(\delta X_j)}{dt} \approx X_{j+1}\delta X_{j-1} + X_{j-1}\delta X_{j+1} - X_{j-2}\delta X_{j-1} - X_{j-1}\delta X_{j-2} - \delta X_j$$

$$\frac{\delta X_j(t + dt) - \delta X_j}{dt} = -X_{j-1}\delta X_{j-2} + \left(X_{j+1} - X_{j-2}\right)\delta X_{j-1} - \delta X_j + X_{j-1}\delta X_{j+1}$$

$$\delta X_j(t + dt) = -X_{j-1}\delta X_{j-2}dt + \left(X_{j+1} - X_{j-2}\right)\delta X_{j-1}dt$$
$$+(1 - dt)\delta X_j + X_{j-1}\delta X_{j+1}\, dt$$

(2) Alternative Way of M

$$\delta X_j(t+dt) = -X_{j-1}\delta X_{j-2}dt + \left(X_{j+1} - X_{j-2}\right)\delta X_{j-1}dt$$
$$+(1-dt)\delta X_j + X_{j-1}\delta X_{j+1}\ dt$$

For example

$$\delta X_1(t+dt) = -X_{40}\delta X_{39}dt + (X_2 - X_{39})\,\delta X_{40}dt$$
$$+(1-dt)\delta X_1 + X_{40}\delta X_2\ dt$$

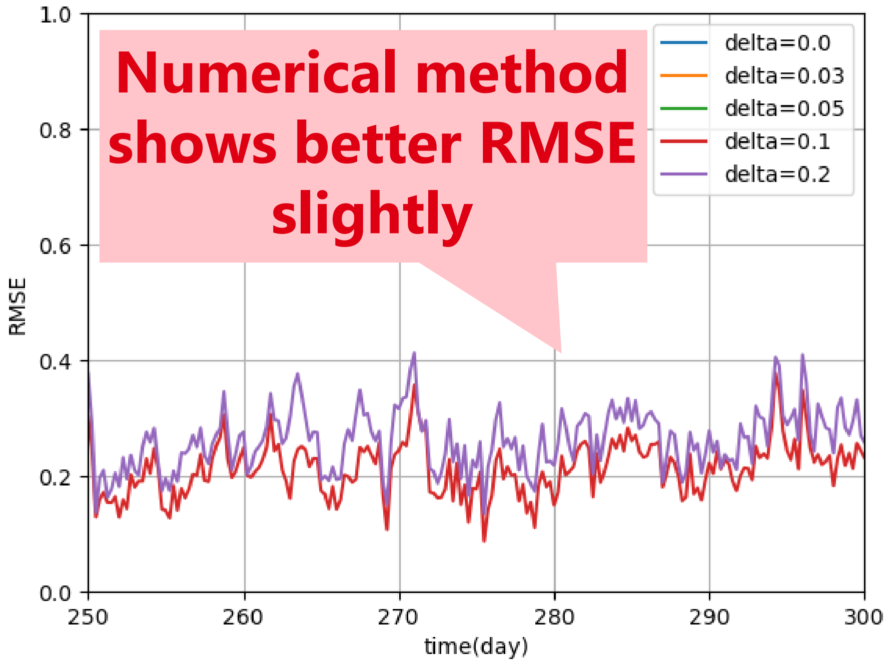
$$\begin{pmatrix} \delta X_1(t+dt) \\ \delta X_2(t+dt) \\ \vdots \\ \delta X_{40}(t+dt) \end{pmatrix} = \begin{pmatrix} 1-dt & X_{40}dt & \cdots & (X_2 - X_{39})dt \\ (X_3 - X_{40})dt & 1-dt & \cdots & -X_1 dt \\ \vdots & \vdots & \ddots & \vdots \\ X_{39}dt & 0 & \cdots & 1-dt \end{pmatrix} \begin{pmatrix} \delta X_1 \\ \delta X_2 \\ \vdots \\ \delta X_{40} \end{pmatrix}$$

=M

(2) KF Comparison of M

Numerical Method

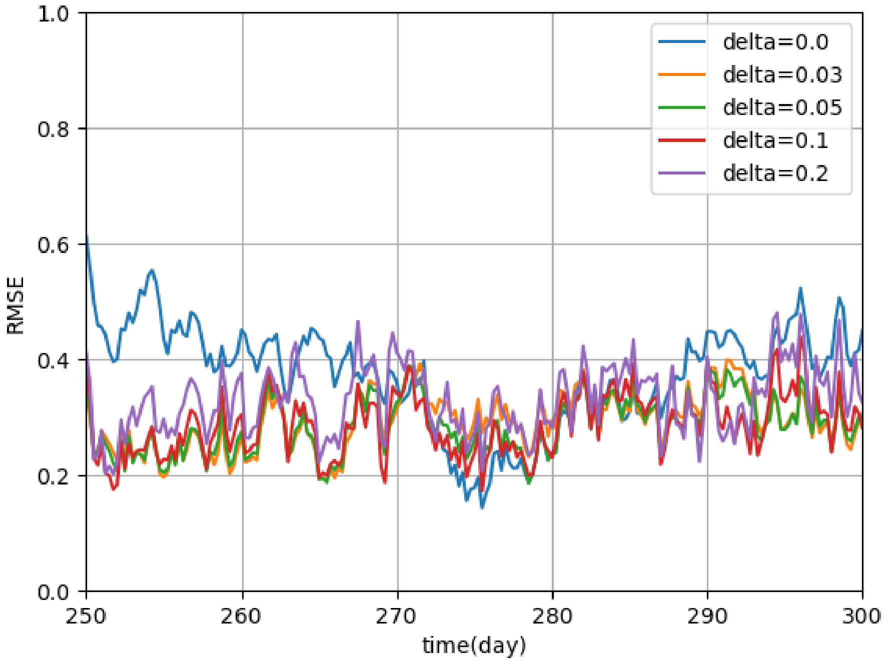
$$\Leftrightarrow \mathbf{M}\mathbf{e}_j \approx \frac{M\left(\mathbf{x}_t^a + \delta \mathbf{e}_j\right) - M\left(\mathbf{x}_t^a\right)}{\delta}$$



Numerical method shows better RMSE slightly

Mathematical Approx.

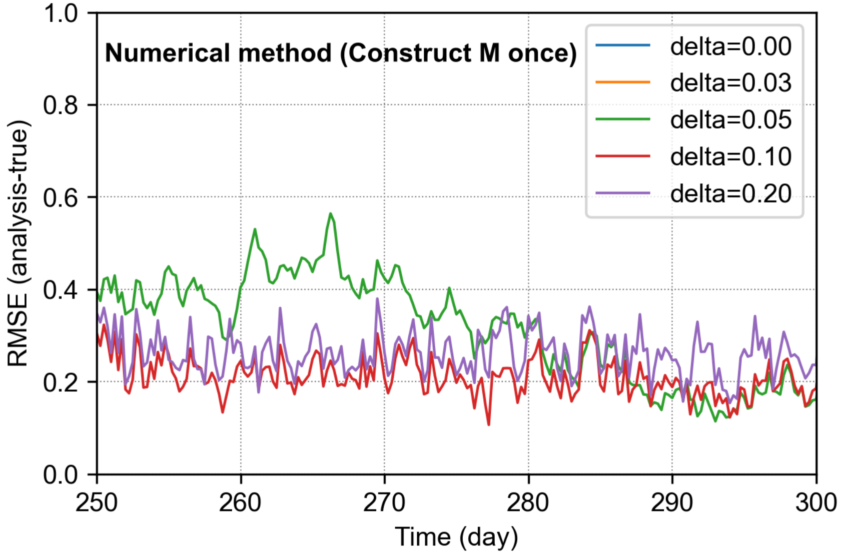
$$\mathbf{M} = \begin{pmatrix} 1 - dt & X_{40}dt & \cdots & (X_2 - X_{39})dt \\ (X_3 - X_{40})dt & 1 - dt & \cdots & -X_1 dt \\ \vdots & \vdots & \ddots & \vdots \\ X_{39}dt & 0 & \cdots & 1 - dt \end{pmatrix}$$

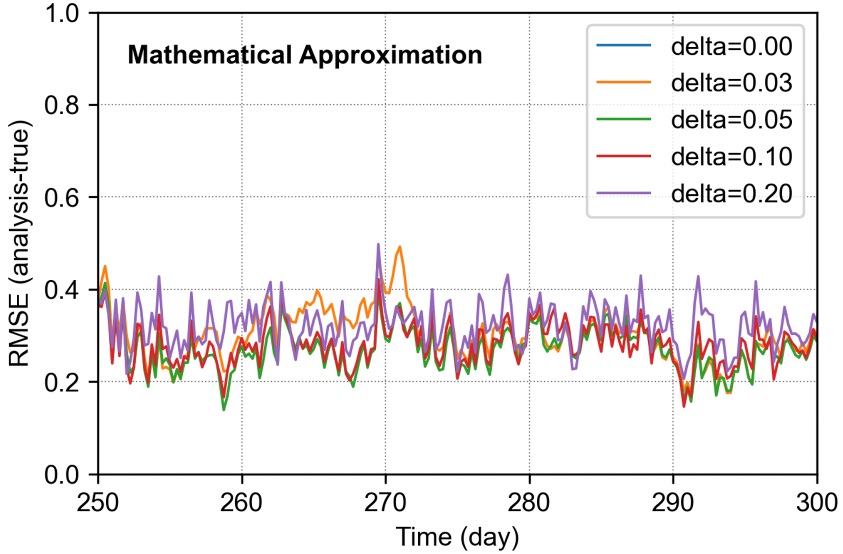


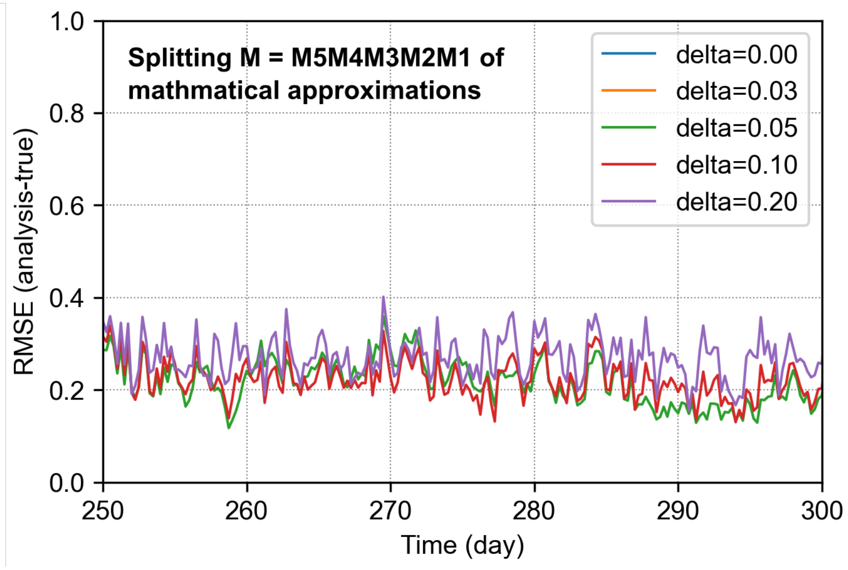
Splitting $\mathbf{M}$ (i.e., $\mathbf{M} = \mathbf{M}_4\mathbf{M}_3\mathbf{M}_2\mathbf{M}_1\mathbf{M}_0$) is necessary for this method to include impacts beyond neighboring grids.

We would be appreciated if you obtained different results

Mao追試







- なんでインフレーションが要らない？
  ➔ 非対角成分が理由？

Thank you for your attention!

Presented by Shunji Kotsuki
(shunji.kotsuki@chiba-u.jp)

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

A World Beyond Predictions

Environmental
Prediction
Science
Laboratory