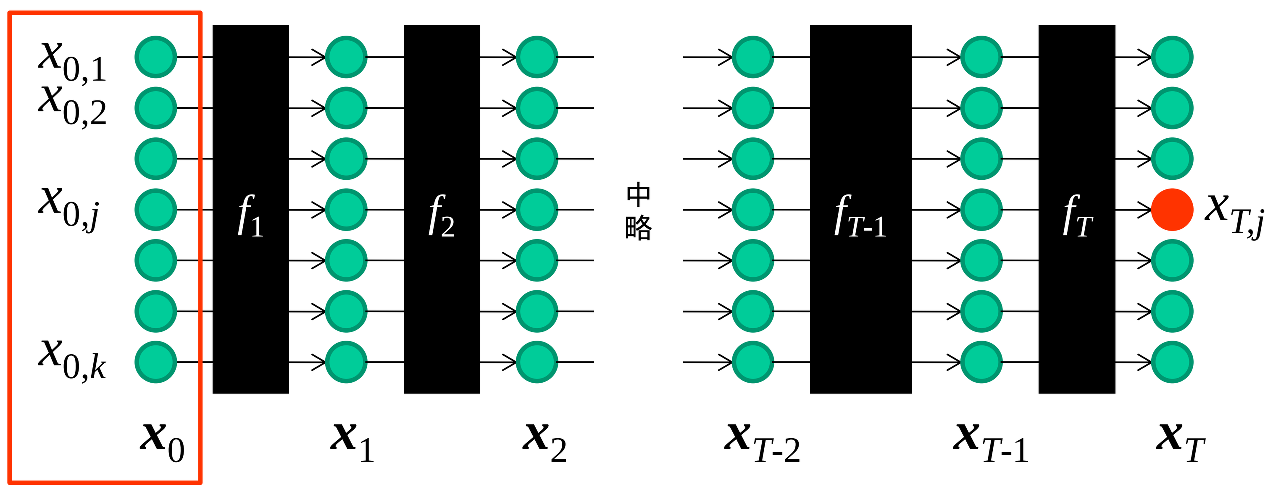
アンサンブル感度解析

上野　玄太　　モデリング研究系　准教授

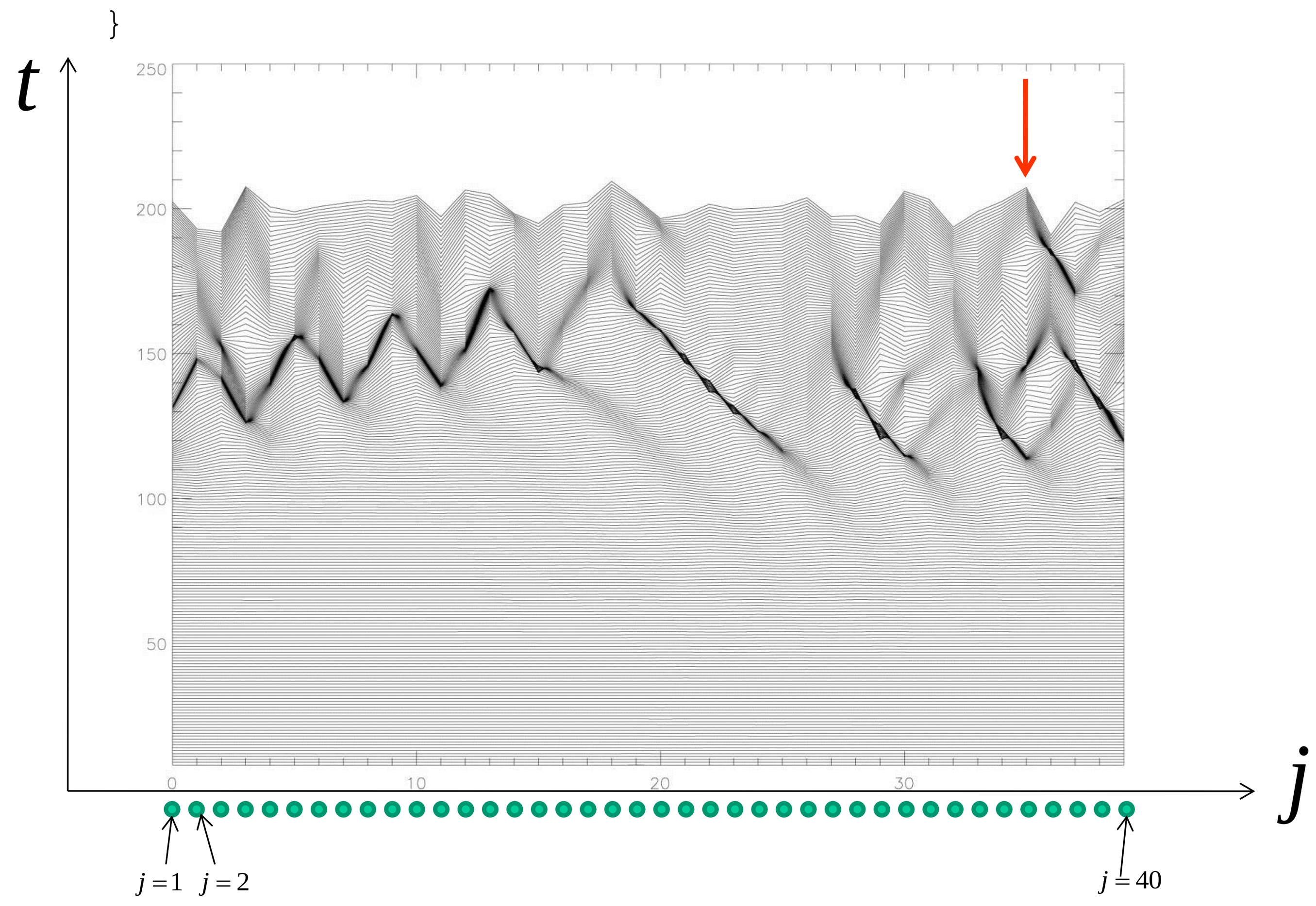
【問題設定】

•最終状態 x_T の j 成分 ($x_{T,j}$ と書く) に多く寄与するのは、初期状態 x_0 のどの成分か (もしくは成分群か)？



【問題例】

```
f(x) {
  for (j) dx[j] = (x[j+1] - x[j-2]) * x[j-1]
                  - x[j] + F;
  for (j) x[j] += dx[j] * dt;
}
```



• 赤矢印で示す山 (t = 200, j = 36) の由来はどこか？

【アンサンブル微分近似法（提案法）】

$$\mathbf{X} = \left(x^{(1)} \quad x^{(2)} \quad \cdots \quad x^{(N)} \right)$$

$$\mathbf{F}(\mathbf{X}) = \left(f\left(x^{(1)}\right) \quad f\left(x^{(2)}\right) \quad \cdots \quad f\left(x^{(N)}\right) \right)$$

$$\mathbf{F}(\mathbf{X}) - \hat{\mathbf{F}}(\mathbf{X}) \simeq \left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}} \left(\mathbf{X} - \hat{\mathbf{X}}\right)$$

$$\Downarrow$$

$$\left(\left.\frac{\partial f}{\partial x}\right|_{x=\hat{x}} \right)' \simeq \left(\left(\mathbf{X} - \hat{\mathbf{X}}\right)\left(\mathbf{X} - \hat{\mathbf{X}}\right)' \right)^{-1} \left(\mathbf{X} - \hat{\mathbf{X}}\right)\left(\mathbf{F}(\mathbf{X}) - \hat{\mathbf{F}}(\mathbf{X})\right)'$$

【データ同化の逐次型と非逐次型の比較】

	逐次型	非逐次型
	本書で取り扱う	取り扱わない
代表的手法	アンサンブルカルマンフィルタ	4次元変分法（アジョイント法）
得る解	状態ベクトル $x_t\ (t=1,\cdots,T)$ の事後周辺分布	状態ベクトル列 $x_{1:T}$ の事後確率最大(MAP)解
数理的観点から	統計的推測	最適化
シミュレーションの規模	小～中規模に適している	超大規模まで可能
使われている領域	すべて	気象・海洋予報の現業が中心
プログラムの実装	既存のシミュレーションを平易にプラグイン	エキスパートが個々のシミュレーションコードに即した勾配計算プログラムを書く必要あり
High Performance Computing	スカラー並列計算向き	ベクトル計算機向き
シミュレーションモデルの比較	尤度により可能	困難
感度計算法 (時間を遡るもの)	未整備 (今回の内容で整備)	アジョイントコードで 実施可能

樋口知之[編著]「データ同化入門」朝倉書店 (2011)

【感度計算】

• $\dfrac{\partial x_{T,j}}{\partial x_0}$ が知りたい！　$x_{T,j}$: 最終状態の注目要素
x_0: 初期状態の全要素

• 直接の差分商 $\dfrac{x_{T,j}\left(x_0 + \Delta x\right) - x_{T,j}\left(x_0\right)}{\Delta x}$ は数値的な問題がある

• 分割して計算するのが通常

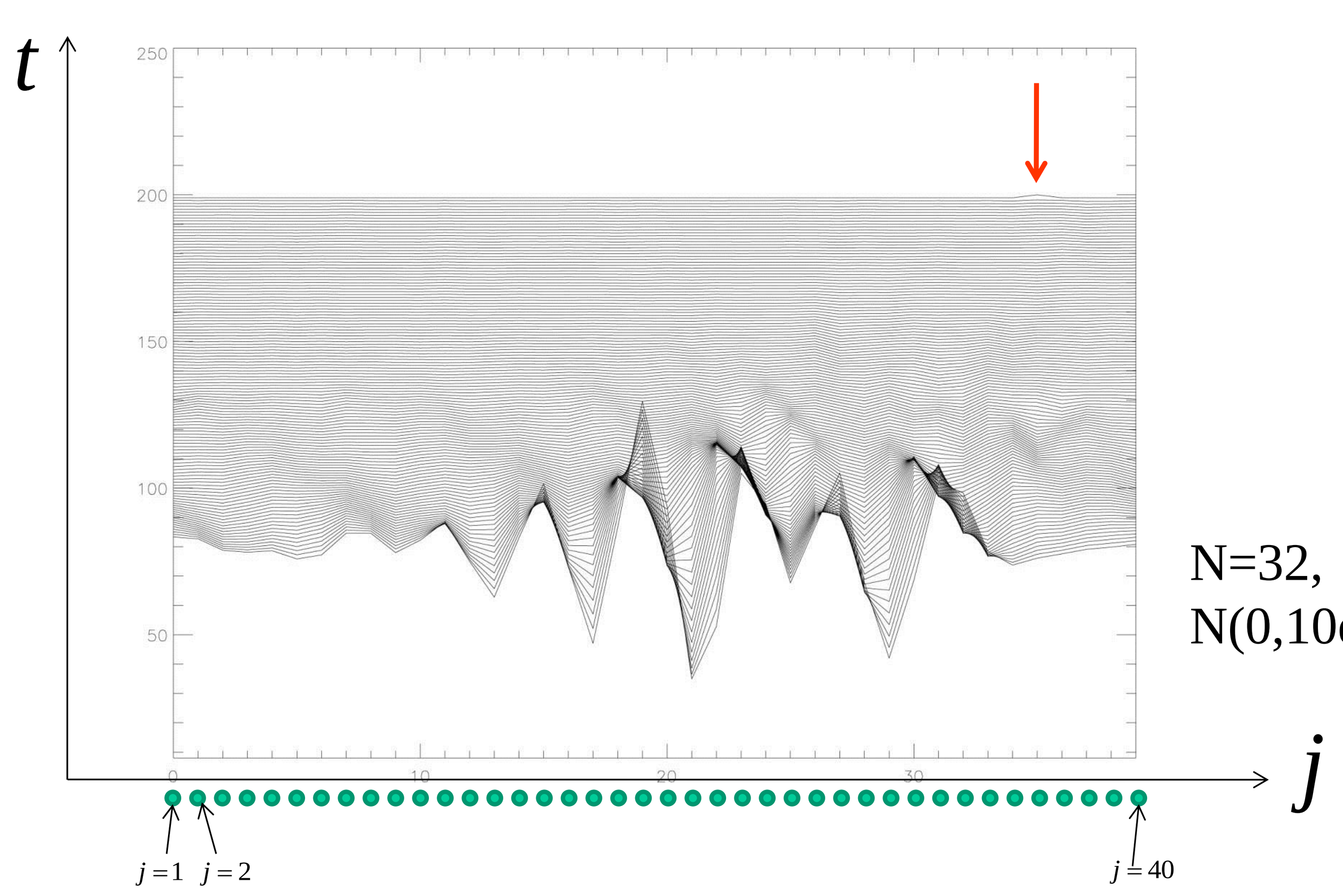
$$\left(\frac{\partial x_{T,j}}{\partial x_0}\right)' = \left(\frac{\partial f_1}{\partial x_0}\right)' \left(\frac{\partial f_2}{\partial x_1}\right)' \cdots \left(\frac{\partial f_T}{\partial x_{T-1}}\right)' \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

【アジョイントコードによる微分計算法（従来法）】

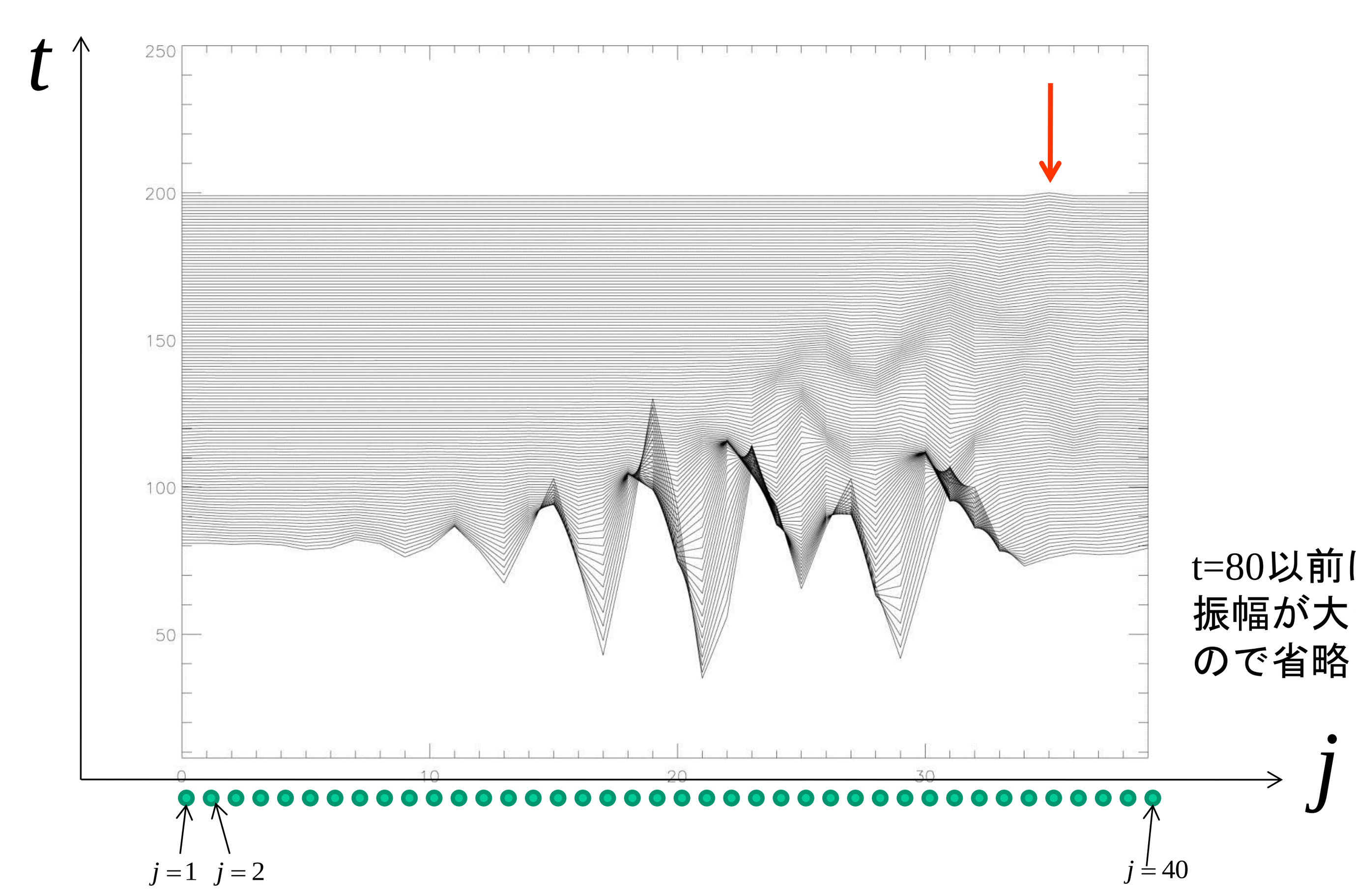
$\left(\dfrac{\partial f_t}{\partial x_{t-1}}\right)' z$ のプログラム（アジョイントコード）

```
adj_f(z) {
  for (j) dx[j] = z[j] * dt;
  read, x;  //f(x) の実行時に書き出しておいた x を読み込む
  for (j) {
    z[j+1] += x[j-1] * dx[j];
    z[j-2] += -x[j-1] * dx[j];
    z[j-1] += (x[j+1] - x[j-2]) * dx[j];
    z[j]   += -dx[j];
  }
}
```

• もとの $f_t(x)$ とほぼ同じ行数のプログラムの作成が必要
• 既存の技術で可能であるが、実は非常に手間がかかる



N=32,
N(0,10e-20)



t=80以前は
振幅が大きい
ので省略

• 左の方から流れてきている。

【まとめ】

•データ同化計算結果 (再解析データ) における感度解析法の提案
•従来は、個々のシミュレーションモデルに即した微分係数の計算プログラム (アジョイントコード) を作成。実装・保守の手間がかかる。
•微分係数のアンサンブル近似法を提案し、シミュレーションモデルの内部構造を意識せず実行可
•簡単なモデルでの感度実験を行い、良好な結果を得た (状態ベクトル 40次元、粒子数32)
•単なる線型回帰で実施できることが朗報。