



システム制御理論と統計的機械学習

第 5 章：状態推定

加嶋 健司

October 3, 2025

京都大学情報学研究科

本章の流れ

実世界の現象をモデリングする際、状態変数を全て観測できることは稀

5.1 状態推定

観測された情報から状態変数を推定する問題を統計的推論として定式化し、線形システムの可観測性との関係を整理する

5.2 逐次推定

線形ガウスシステムの状態変数を観測から最適に逐次予測するカルマンフィルタの考え方と、出力フィードバック制御の設定で LQR 問題の解法を与える

5.3 スムージングと制御の双対性

非線形システムの出力時系列から状態時系列を MAP 推定する問題を、最適制御問題に変換し、動的計画法を適用する

状態推定

統計的推論としての定式化 (1) 部分観測確率システム

Definition (部分観測 (partially observable) 確率システム)

入力 u , 状態 x , 出力 y , プロセス雑音 v , 観測雑音 w . 雑音 v, w は無記憶

$$\begin{cases} x_{k+1} = f_k(x_k, u_k, v_k), \\ y_k = h_k(x_k, w_k), \end{cases} \quad (5.1)$$

観測雑音 $w_k \in \text{rv}(\mathbb{W})$: $h_k: \mathbb{X} \times \mathbb{W} \rightarrow \mathbb{Y}$ を介して $y_k \in \text{rv}(\mathbb{Y})$ を定める

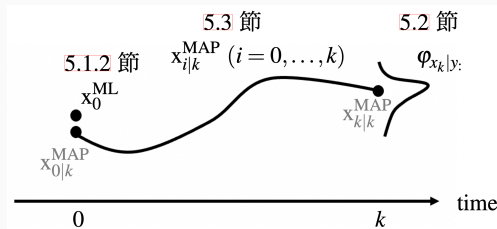
観測値 (observation, measurement) y_k : 状態 x_k の部分的な情報を有する.

関数 f_k, h_k は既知で入出力 u_{\cdot}, y_{\cdot} のみが実際に観測できる状況を想定し, その観測値に基づいて状態 x_{\cdot} の推論を行う.

統計的推論としての定式化 (2) 3 種類の状態推定問題

現在時刻 k までの入出力系列 $u_{0:k}, y_{0:k}$ から

- 現在の状態 x_k を推定するフィルタリング (filtering) \Rightarrow フィルタ
- 未来 $l > k$ における状態 x_l を推定する予測 (prediction) \Rightarrow 予測器
- 過去 $l > k$ における状態 x_l を推定するスムージング (smoothing) \Rightarrow スムーザ
- 5.1 節：初期状態を推定するスムージング問題に対する最尤推定
- 5.2 節：予測とフィルタリング問題に対する状態の分布推定
- 5.3 節：過去の状態軌道を MAP 推定するスムージング問題



線形確率システムと最尤初期状態推定 (1) 可観測性の復習

線形システム $x(k+1) = Ax(k), y(k) = Cx(k)$ に対して

- $y(0), \dots, y(k-1)$ の値から $x(0)$ が一意に定まるとき、このシステムは時刻 k において**可観測 (observable)** であるという。
- 観測系列は可観測性行列と初期状態を用いて以下のように表すことができる。

$$\begin{bmatrix} y(0) \\ y(1) \\ \vdots \\ y(k-1) \end{bmatrix} = O_k x(0), O_k := \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{bmatrix} \quad (5.2)$$

- システムの可観測性, O_k の列フルランク性, 可観測性グラミアン

$$G_k^o := O_k^\top O_k = \sum_{i=0}^{k-1} (A^\top)^i C^\top C A^i$$

の正則性はすべて等価である。

線形確率システムと最尤初期状態推定 (2) 観測雑音を有する線形システム

部分観測確率システム $x_{k+1} = Ax_k, y_k = Cx_k + w_k$ ($y_{0:k-1} = O_k x_0 + w_{0:k-1}$)

定理 5.2.2 – 可観測性グラミアンとフィッシャー情報行列

$w_k, \sim \mathcal{N}(0, I)$, i.i.d. とすると,

$$\text{FIM}_{y_{0:k-1}|x_0=x_0} = G_k^o, \forall x_0 \quad (5.3)$$

さらに, G_k^o が正則のとき, $y_{0:k-1}$ にもとづく x_0 の最尤推定値は

$$x_0^{\text{ML}}(y_{0:k-1}) = (O_k^\top O_k)^{-1} O_k^\top y_{0:k-1} \quad (5.4)$$

で与えられ, 十分統計量である.

- G_k^o が特異な行列に近づくと, 不偏推定量に対する推定分散の限界の下界値である FIM の逆行列 $(G_k^o)^{-1}$ が大きくなり, 初期状態の推定限界が悪化する

逐次推定

逐次ベイズ予測 (1) 線形確定システムの逐次状態推定

線形システム $x(k+1) = Ax(k) + Bu(k)$, $y(k) = Cx(k)$ の場合

オブザーバー

$x(k) \neq \hat{x}(k)$ のとき, その誤差を出力 $y(k)$ と状態推定値から予想される出力値 $C\hat{x}(k)$ の差を通じて定量化し, 行列 H を用いて線形に推定値を修正

$$\hat{x}(k+1) = A\hat{x}(k) + Bu(k) + H(y(k) - C\hat{x}(k)) \quad (5.5)$$

- 出力系列を保持しておく必要はなく, 各時刻の処理も軽微
- $e(k) := x(k) - \hat{x}(k)$ が $e(k+1) = (A - HC)e(k)$ を満たすことから, 行列 $A - HC$ がシューア安定であれば, 任意の $x(0)$, $\hat{x}(0)$ に対して $\lim_{k \rightarrow \infty} \|x(k) - \hat{x}(k)\| = 0$

逐次ベイズ予測 (2) 定式化

過去の観測系列 $y_{0:k-1}$ から現在の状態 x_k を逐次推定する, つまり条件付き確率密度関数 $\varphi_{x_k}(\cdot|y_{0:k-1})$ を求めることが目標

仮定 5.2.1 – 雑音の無記憶性

$x_0 \perp\!\!\!\perp (v_0, w_0)$, $(x_0, v_{0:k-1}, w_{0:k-1}) \perp\!\!\!\perp (v_k, w_k)$, $\forall k \geq 1$ が成り立つ.

仮定 5.2.2 – 確定的出力履歴フィードバック

u_0 は定数, $u_k \in \text{rv}(y_{0:k-1})$, $k \geq 1$ である.

- 制御目的は問わない
- 遷移確率密度関数 $\Psi_k(x_{k+1}, y_k | x_k, u_k) := \varphi(x_{k+1}, y_k | x_k, u_k)$ は u_{\cdot} に依存しない

逐次ベイズ予測 (3) アルゴリズム

$\varphi_{x_k}(\cdot|y_{0:k-1})$ を求める

定理 5.2.3 – 逐次ベイズ予測

$\varphi(y_{0:k}) > 0$ を満たす任意の出力系列 $y_{0:k}$ に対して,

$$\varphi(x_{k+1}, y_k | y_{0:k-1}) = \int_{\mathbb{X}} \Psi_k(x_{k+1}, y_k | x_k, u_k) \varphi(x_k | y_{0:k-1}) dx_k \quad (5.6)$$

$$\varphi(x_{k+1} | y_{0:k}) = \frac{\varphi(x_{k+1}, y_k | y_{0:k-1})}{\varphi(y_k | y_{0:k-1})} \quad (5.7)$$

が成り立つ. ただし, $u_k = u_k|_{y_{0:k-1}}$, $\varphi(x_0|y_{-1}) := \varphi(x_0)$ とする.

- 条件付き期待値と規格化の計算を反復

線形システムの逐次予測 (1) 定式化

線形確率システム

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k + v_k \\ y_k = C_k x_k + w_k \end{cases} \quad (5.8)$$

x_0, v, w : は無記憶性かつ

$$x_0 \sim \mathcal{N}(\bar{x}, \bar{\Sigma}), \begin{bmatrix} v_k \\ w_k \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} N_k & O \\ O & M_k \end{bmatrix} \right), \begin{bmatrix} N_k & O \\ O & M_k \end{bmatrix} \succeq O, M_k \succ O \quad (5.9)$$

Problem (線形システムの逐次予測問題)

観測系列 $y_{0:k-1}$ から現在の状態 x_k の条件付き確率密度関数 $\varphi_{x_k}(\cdot | y_{0:k-1})$ を求めよ。

線形システムの逐次予測 (2) 最適予測器

定理 5.2.4 – 線形システムの逐次予測

$$\tilde{N}_k := N_k + A_k \Sigma_k A_k^\top, \tilde{M}_k := M_k + C_k \Sigma_k C_k^\top, \tilde{L}_k := A_k \Sigma_k C_k^\top$$

$$\Sigma_0 := \bar{\Sigma}, \Sigma_{k+1} := \tilde{N}_k - \tilde{L}_k \tilde{M}_k^{-1} \tilde{L}_k^\top, H_k := \tilde{L}_k \tilde{M}_k^{-1}$$

$y_{0:k-1}$ で条件付けられた x_k の事後確率密度関数は,

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + H_k(y_k - C_k\hat{x}_k), \hat{x}_0 = \bar{x}$$

$$x_k|_{y_{0:k-1}} \sim \mathcal{N}(\hat{x}_k, \Sigma_k), \hat{x}_k = \hat{x}_k|_{y_{0:k-1}}$$

さらに, 推定誤差 $e_k := x_k - \hat{x}_k$ は $y_{0:k-1}, \hat{x}_{0:k}$ のいずれとも独立である.

- LQR 問題との類似性を制御と推定の双対性ともよぶ
- H が大きいと修正は速いが, w の影響が増大する
- 状態の事後分布の分散 Σ_k は観測値に依存しない

線形二次ガウシアン制御 (1) 定式化

問題 5.2.5 – 線形二次ガウシアン (Linear Quadratic Gaussian; LQG)

線形確率システム

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k + v_k \\ y_k = C_k x_k + w_k \end{cases}$$

に対してコスト関数

$$J(u_{\cdot}) := \mathbb{E} \left[\sum_{k=0}^{\bar{k}-1} (x_k^{\top} Q x_k + u_k^{\top} R u_k) + x_{\bar{k}}^{\top} \Pi_{\bar{k}} x_{\bar{k}} \right], \quad Q, R, \Pi_{\bar{k}} \succ O$$

を最小化する出力履歴フィードバック $u_k \in \sigma(y_{0:k-1})$ を求めよ。

線形二次ガウシアン制御 (2) 最適制御則

- LQR 問題の場合 ($y_k = x_k$) は, $u_k = -K_k x_k$ が最適 (定理 4.2.2)
- x_k の事後分布は,

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + H_k(y_k - C_k\hat{x}_k), \hat{x}_0 = \bar{x}$$

$$x_k|_{y_{0:k-1}} \sim \mathcal{N}(\hat{x}_k, \Sigma_k), \hat{x}_k = \hat{x}_k|_{y_{0:k-1}}$$

問題 5.2.6 – LQG 制御問題に対する推定と制御の分離定理

LQG 制御問題の最適制御則

$$u_k = -K_k \hat{x}_k \in \text{rv}(y_{0:k-1})$$

- 状態推定則と状態フィードバック制御則を個別に設計することを正当化しており, **分離定理** (separation theorem) とよばれる.
- プロセス雑音 $\hat{v}_k := H_k(y_k - C_k\hat{x}_k)$ を \hat{x}_k の LQR

線形二次ガウシアン制御 (3) 有色雑音と LQG 制御の例

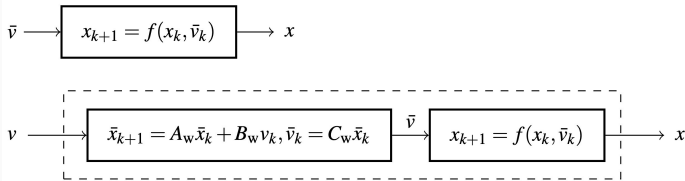
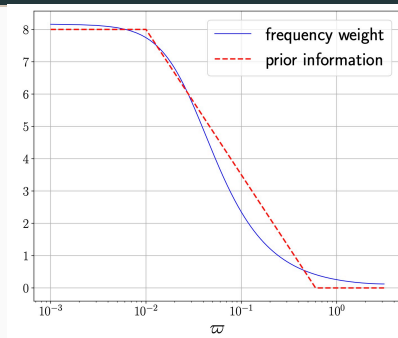
例 5.2.7

二重積分器を時間離化したシステム

$$x_{k+1} = Ax_k + B(u_k + \bar{v}_k), \quad x_k \in \mathbb{R}^2$$

$$Q = \text{diag}(1, 10), \quad R = 10^{-4}$$

$(x_k)_1$ を位置, $(x_k)_2$ は速度, u_k は力など



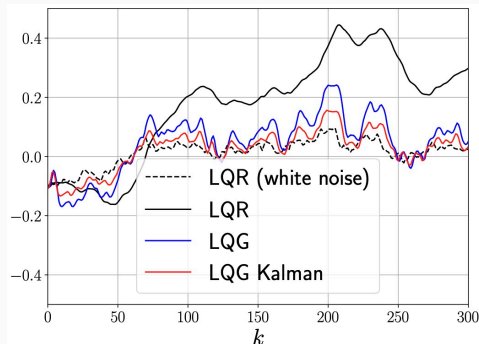
線形二次ガウシアン制御 (4) 有色雑音と LQG 制御の例

例 5.2.7

- \bar{v} が白色, 制御則が LQR (状態 FB)
- \bar{v} が有色, 制御則が LQR (状態 FB)
- \bar{v} が有色, 制御則が LQG (出力 FB)

位置センサーのみが利用可能

$$y_k = (x_k)_1 + w_k, w_k \sim \mathcal{N}(0, 10^{-4}), \text{i.i.d.}$$



(a) 制御則の比較 (位置 $(x_k)_1$)

$$\begin{bmatrix} x_{k+1} \\ \bar{x}_{k+1} \end{bmatrix} = \begin{bmatrix} A & BC_w \\ O & A_w \end{bmatrix} \begin{bmatrix} x_k \\ \bar{x}_k \end{bmatrix} + \begin{bmatrix} O \\ B_w \end{bmatrix} v_k, y_k = Cx_k + w_k$$

カルマンフィルタ (1) 観測更新と時間更新

一段先予測 $x_k|_{y_{0:k-1}} \sim \mathcal{N}(\hat{x}_k, \Sigma_k)$ に対して, $x_k|_{y_{0:k}}$ を求めるフィルタリング問題

定理 5.2.8 – カルマンフィルタ

$$x_k|_{y_{0:k}} \sim \mathcal{N}(\check{x}_k, \check{\Sigma}_k)$$

- **カルマンゲイン** : $\check{H}_k := \check{L}_k \tilde{M}_k^{-1}$, $\check{L}_k := \Sigma_k C_k^\top$
- **観測更新式** :

$$\check{x}_k := \hat{x}_k + \check{H}_k(y_k - C_k \hat{x}_k), \quad \check{\Sigma}_k := \Sigma_k - \check{L}_k \tilde{M}_k^{-1} \check{L}_k^\top \quad (5.10)$$

- $x_k|_{y_{0:k}} \sim \mathcal{N}(\check{x}_k, \check{\Sigma}_k)$ と $x_{k+1} = Ax_k + Bu_k + v_k$ からしたかう**時間更新式**

$$\hat{x}_{k+1} = A_k \check{x}_k + B_k u_k, \quad \Sigma_{k+1} = A_k \check{\Sigma}_k A_k^\top + N_k \quad (5.11)$$

- LQG 制御問題において $u_k \in \text{rv}(y_{0:k-1})$ を $u_k \in \text{rv}(y_{0:k})$ とすると $u_k = -K_k \check{x}_k$ が最適

カルマンフィルタ (2) 拡張カルマンフィルタ

$$\begin{cases} x_{k+1} = Ax_k + Bu_k + v_k \\ y_k = Cx_k + w_k \end{cases}$$

$$\begin{cases} x_{k+1} = f_k(x_k, u_k) + v_k \\ y_k = h_k(x_k) + w_k \end{cases}$$

アルゴリズム 1 最適 LQG 制御則

input k, \hat{x}, Σ, y

output $u, \check{x}, \check{\Sigma}, \hat{x}_+, \Sigma_+$

1: $\tilde{M} \leftarrow M_k + C_k \Sigma C_k^\top, \check{L} \leftarrow \Sigma C_k^\top, \check{H} \leftarrow \check{L} \tilde{M}^{-1}$

2: $\check{x} \leftarrow \hat{x} + \check{H}(y - C_k \hat{x})$

3: $\check{\Sigma} \leftarrow \Sigma - \check{L} \tilde{M}^{-1} \check{L}^\top$

4: $u \leftarrow -K_k \check{x}$

5: $\hat{x}_+ \leftarrow A_k \check{x} + B_k u, \Sigma_+ \leftarrow A_k \check{\Sigma} A_k^\top + N_k$

C_k を \hat{C}_k で置き換え

A_k を \hat{A}_k で置き換え

$$f_k(x, u_k) \simeq f_k(\check{x}_k, u_k) + \hat{A}_k(x - \check{x}_k), \quad \hat{A}_k := \left. \frac{\partial f_k}{\partial x} \right|_{x=\check{x}_k, u=u_k}$$

$$h_k(x) \simeq h_k(\hat{x}_k) + \hat{C}_k(x - \hat{x}_k), \quad \hat{C}_k := \left. \frac{\partial h_k}{\partial x} \right|_{x=\hat{x}_k}$$

スムージングと制御の双対性

最尤スムージング問題 (1) 定式化

Problem (最尤スムージング問題)

$\{x_0, v_0, v_1, \dots, w_1, w_2, \dots\}$ はすべて独立であるとし，非線形確率システム

$$\begin{cases} x_{k+1} = f_k(x_k) + v_k \\ y_k = h_k(x_k) + w_k \end{cases} \quad (5.12)$$

与えられた観測軌道 $y_{0:k}$ に対して，状態軌道の事後確率 $\varphi(x_{0:k} | y_{0:k} = y_{0:k})$ を最大化する $x_{i|k}^{\text{MAP}}, i = 0, 1, \dots, k$ を求めよ．

- ・ 仮定： $\varphi(y_{0:k}) > 0$

システムに混入される雑音を制御入力としてみなすことで

- ・ 大きな駆動雑音 v は実現しにくい \iff 制御入力の大きさに応じたコスト
- ・ 大きな観測雑音 w は実現しにくい \iff 観測結果 $y_{0:k-1}$ と出力との差に応じたコスト

最尤スムージング問題 (2) 最適制御問題との双対性

定理 5.3.1 – 最尤スムージングと最適制御の双対性

最尤スムージング問題の最適解 $x_{i|k}^{\text{MAP}}, i = 0, 1, \dots, k$ は、確定システム

$$x(i+1) = f_k(x(i)) + u(i), i = 0, 1, \dots, k-1 \quad (5.13)$$

に対してコスト関数

$$l_{\text{ini}}(x(0)) + \sum_{i=0}^{k-1} l(i, x(i), u(i)) \quad (5.14)$$

$$l_{\text{ini}}(x) := -\log \varphi_{\hat{x}_0}(x) - \log \varphi_{w_0}(y_0 - h_0(x)) \quad (5.15)$$

$$l(i, x, u) := -\log \varphi_{w_{i+1}}(y_{i+1} - h_{i+1}(f_i(x) + u)) - \log \varphi_{v_i}(u) \quad (5.16)$$

を最小化する $\hat{x}(0), u(0), \dots, u(k-1)$ を求める最適制御問題の最適軌道と一致する。

最尤スムージング問題 (3) 証明の概要

最尤スムージング問題の評価関数は観測軌道 $y_{0:k} = y_{0:k}$ に対する状態軌道 $x_{0:k}$ の事後確率であり、ベイズの定理より

$$\varphi_{x_{0:k}}(x_{0:k} | y_{0:k} = y_{0:k}) \propto \varphi_{y_{0:k}}(y_{0:k} | x_{0:k} = x_{0:k}) \varphi_{x_{0:k}}(x_{0:k}) \quad (5.17)$$

が成り立つ。また、独立性の仮定より

$$\varphi_{x_{0:k}}(x_{0:k}) = \varphi_{x_0}(x_0) \prod_{i=0}^{k-1} \varphi_{x_{i+1}}(x_{i+1} | x_i = x_i) = \varphi_{x_0}(x_0) \prod_{i=0}^{k-1} \varphi_{v_i}(x_{i+1} - f_i(x_i)) \quad (5.18)$$

$$\varphi_{y_{0:k}}(y_{0:k} | x_{0:k} = x_{0:k}) = \prod_{i=0}^k \varphi_{y_i}(y_i | x_i = x_i) = \prod_{i=0}^k \varphi_{w_i}(y_i - h_i(x_i)) \quad (5.19)$$

も成り立つ。従って(5.17)の最大化は(5.14)の最小化と等価である。

最尤スムージング問題 (4) 線形システムの例

例 5.3.2 (線形システムの最尤スムージング)

$$x_{k+1} = Ax_k + B_v v_k, \quad x_0 \sim \mathcal{N}(\mu_0, I), \quad \mu_0 := \mathbb{1}_{3 \times 1}$$

$$y_k = (x_k)_1 + w_k$$

$$A := \begin{bmatrix} -0.3900 & -0.6700 & -0.3400 \\ 0.7100 & -0.5100 & 0.1100 \\ -0.4600 & -0.3500 & -0.1200 \end{bmatrix}, \quad B_v = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

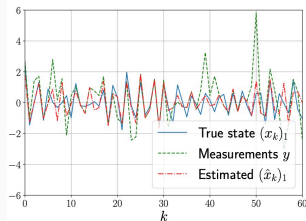
v_k は打ち切り正規分布 $\varphi(v) \propto \mathcal{N}(v|0, 1) \mathbb{1}_{[-1, 1]}(v)$, i.i.d.

w_k はラプラス分布 $w_k \sim \text{Lap}(0, 1)$, i.i.d.

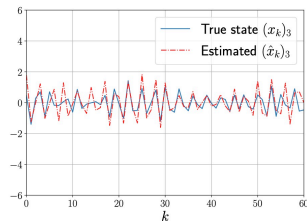
$$\text{minimize}_{\hat{x}(0), \{u(i)\}_{i=0}^{\bar{k}-1}} \quad \frac{1}{2} \|\hat{x}(0) - \mu_0\|^2 + \frac{1}{2} \sum_{i=0}^{\bar{k}-1} u(i)^2 + \sum_{i=0}^{\bar{k}} |(\hat{x}(i))_1 - y_i|$$

$$\text{subject to} \quad \hat{x}(k+1) = A\hat{x}(k) + B_v u(k), \quad |u(k)| \leq 1$$

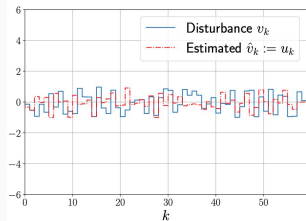
最尤スムージング問題 (5) 線形システムの例 (つづき)



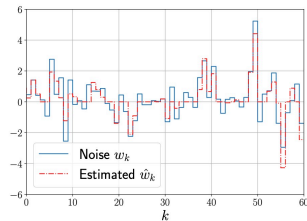
(a) $(x_k)_1$ と観測値・推定値



(b) $(x_k)_3$ と推定値



(c) v_k と推定値



(d) w_k と推定値

Moving horizon 推定 (1) 到着コスト

最尤スムージングの解 $x_{:|k}^{\text{MAP}} := (x_{0|k}^{\text{MAP}}, \dots, x_{k|k}^{\text{MAP}})$

- **Moving Horizon 推定 (MH 推定) :**

各時刻 k において, $x_{k|k}^{\text{MAP}}$ を状態 x_k の推定値とする手法

- **到着コスト (arrival cost) :** $x(i+1) = f_k(x(i)) + u(i)$ の制約のもとで,

$$V^a(k, x) := \inf_{x(0), \{u(i)\}_{i=0}^{k-1} \text{ s.t. } \hat{x}(k)=x} \left\{ \ell_{\text{ini}}(\hat{x}(0)) + \sum_{i=0}^{k-1} \hat{\ell}(i, \hat{x}(i), u(i)) \right\} \quad (5.20)$$

- $\arg \min_x V^a(k, x) = x_{k|k}^{\text{MAP}}, \forall k \in \mathbb{Z}_+$
 - $i < k$ に対して $\arg \min_x V^a(i, x) = x_{i|k}^{\text{MAP}}$ は成り立たない
 - x_k の MAP 推定値 $\arg \max_{x_k} \varphi(x_k | y_{0:k})$ とは一般に一致しない

Moving horizon 推定 (2) 動的計画法

定理 5.3.3 – 到着コストに関する動的計画法

$$V^a(0, x) = \ell_{\text{ini}}(x), \quad (5.21)$$

$$V^a(k, x) = \min_{x'} \{V^a(k-1, x') + \hat{\ell}(k-1, x, x - f_{k-1}(x'))\}, \quad k \geq 1 \quad (5.22)$$

新しく観測値 $y_k = y_k$ を得るたびに、

1. $V^a(k, x)$, $\forall x$ を求め、

2. $x_{k|k}^{\text{MAP}}$ を求める

という手順を繰り返せばよい。

ただし、各時刻において事後確率最大化

軌道 $x_{:,|k}^{\text{MAP}}$ が求まるわけではない。

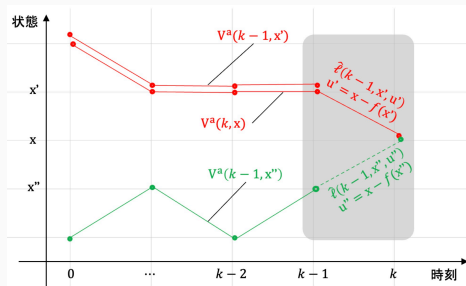


Figure 2: 到着コストに対する動的計画法

Moving horizon 推定 (3) カルマンフィルタとの等価性

定理 5.3.4 – 線形システムにおけるカルマンフィルタと MH 推定の等価性

$$\begin{cases} x_{k+1} = A_k x_k + B_k u_k + v_k \\ y_k = C_k x_k + w_k \end{cases},$$
$$x_0 \sim \mathcal{N}(\bar{x}, \bar{\Sigma}), \begin{bmatrix} v_k \\ w_k \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} N_k & O \\ O & M_k \end{bmatrix} \right), \begin{bmatrix} N_k & O \\ O & M_k \end{bmatrix} \succeq O, M_k \succ O$$

最尤スムージングの事後分布最大化軌道 $x_{\cdot|k}^{\text{MAP}}$ は

$$x_{k|k}^{\text{MAP}} = \check{x}_k (:= \arg \max_{x_k} \varphi(x_k | y_{0:k})) \quad (5.23)$$

- $\varphi(x_k | y_{0:k}) \propto \exp(-V^a(k, x_k)), \forall k, y_{0:k}$ を示すことができる。

第 5 章

1. 状態推定

統計的推論としての定式化

線形確率システムと最尤初期状態推定

2. 逐次推定

逐次ベイズ予測

線形システムの逐次予測

線形二次ガウシアン制御

カルマンフィルタ

3. スムージングと制御の双対性

最尤スムージング問題

Moving horizon 推定