

粒子フィルタの基礎と応用： フィルタ・平滑化・パラメータ推定

矢野 浩一*

A Tutorial on Particle Filters: Filters, Smoothing, and Parameter Estimation

Koiti Yano*

本稿では、粒子フィルタ、粒子平滑化、粒子フィルタによるパラメータ推定を解説する。粒子フィルタは非線形・非ガウス状態空間モデルの状態推定を実行するシミュレーションベースのアルゴリズムであり、1990年代初頭に発表された後、科学・技術の幅広い分野で活用されてきた。しかし、日本国内ではその知識が十分に普及しているとはいいがたいため、本稿では第2節で粒子フィルタとその適用例、第3節で粒子固定ラグ平滑化の基礎、第4節で粒子固定ラグ平滑化へのリサンプリング・ムーブ法の適用、第5節で粒子フィルタと自己組織化状態空間モデルによるパラメータ推定、第6節で実物景気循環モデルの基礎と粒子フィルタを用いた状態推定例を解説する。

Particle filters and smoothers are simulation-based methods to estimate non-linear non-Gaussian state space models. The filters and smoothers are widely applied to science and engineering from the early 1990s. We describe an introduction to the particle filter and some applications in Section 2. The particle fixed-lag smoother is denoted in Section 3, and we apply the resample-move method to the particle fixed-lag smoother in Section 4. We explain parameter estimation and a self-organizing state space model in Section 5. In Section 6, we estimate a Real Business Cycle model based on the filter.

キーワード：粒子フィルタ、粒子平滑化、リサンプリング・ムーブ、自己組織化状態空間モデル、パラメータ推定、動学的マクロ経済学、実物景気循環モデル

1. はじめに

本稿では Kitagawa (1993, 1996), Gordon *et al.* (1993) で提案された粒子フィルタ (particle filter), 粒子固定ラグ平滑化 (particle fixed-lag smoother), Kitagawa (1998) で提案された自己組織化状態空間モデルと粒子フィルタによるパラメータ推定を解説する。¹⁾

* 駒澤大学経済学部：〒154-8525 東京都世田谷区駒沢 1-23-1 (E-mail: koiti@komazawa-u.ac.jp).

¹⁾ 粒子フィルタは本来であれば、原論文である Kitagawa (1993, 1996) に従い「モンテカルロフィルタ (Monte Carlo filter)」と呼ばれるべきであると筆者は考えており、矢野・佐藤 (2006) の当時にはそうしていた。しかし、その後「粒子フィルタ (パーティクルフィルタ)」という別称が普及してしまったため、本稿ではやむを得ず「粒子フィルタ (パーティクルフィルタ)」という名称を用いる。

Kalman (1960) 以降、線形・ガウス型状態空間モデルの状態推定はカルマン・フィルタ (Kalman Filter) を用いるのが一般的となり、カルマン・フィルタは 1960 年代以降、科学・技術・社会科学の多くの分野で幅広く用いられてきた。状態空間モデルが非線形・非ガウス型である場合は一般にはカルマン・フィルタは使用できないが、モデルを線形で一次近似し、その範囲内であれば、拡張カルマンフィルタが適用可能である。しかし、線形近似の範囲内から外れる場合は状態推定が適切に行えないなどの問題があることも古くからよく知られてきた。そのため、1970 年代以降カルマン・フィルタの限界を解決するべくより一般的な状態推定アルゴリズムが Kitagawa (1987), Kramer and Sorenson (1988), Tanizaki (1993) 等で提案されてきた。しかし、これらの手法は、アルゴリズムの複雑さや次元の呪いなどの問題、さらに当時は CPU の能力不足などにより広く普及することはなかった。

そのような状況を打開すべく、Kitagawa (1993, 1996), Gordon *et al.* (1993) が提案したアルゴリズムが「粒子フィルタ (particle filter)」である。これは非線形・非ガウス状態空間モデルの状態推定アルゴリズムであり、モンテカルロ法 (モンテカルロ・シミュレーション) によって実現される。ベイズの定理に基づく比較的シンプルな手法であり、プログラムが非常に容易であるという特徴を有する。モンテカルロ法を用いるため、計算機の負荷が小さいとは言いがたいが、提案された 1990 年代初頭以後、CPU や並列化計算 (近年では GPGPU[General-Purpose computing on GPU]) 等の急速な発展により計算負荷の問題は徐々に解消されてきている。

粒子フィルタはその提案後直ちに多くの分野に応用され、現在では時系列解析を含め科学・技術・社会科学において標準的手法となっている。邦文による入門としては比較的初期に執筆された北川 (2005), 樋口 (2005), 伊庭 (2005), 谷崎 (2006), 矢野・佐藤 (2006), 生駒 (2008) など、近年執筆されたものとしては樋口 (2011a,b), 中村・樋口 (2009) などがある。英文による入門としては Doucet *et al.* (2001), Liu (2002), Ristic *et al.* (2004), Cappe *et al.* (2007), Candy (2009), Prado and West (2010), Doucet and Johansen (2011) などがある。また、応用例としては、粒子フィルタと遺伝的アルゴリズムの類似性に着目した樋口 (1996), 気象学に応用した樋口 (2011b), 金融市場に適用した佐藤・高橋 (2005), 矢野・佐藤 (2006) などがある。

本稿では、この粒子フィルタの応用例として、実物景気循環モデル (Real Business Cycles Model, 以下 RBC) の 2 次近似の状態推定を行う。RBC は近年、マクロ経済学でしばしば用いられる動学的マクロ経済モデルの基礎である。また、RBC を 2 次近似 (定常状態近傍での 2 次までの Taylor 展開) したものは非線形状態空間モデルとなるため、その推定は粒子フィルタの典型的応用例と言える。なお、RBC を始めとした動学的マクロ経済モデルへの応用は Fernandez-Villaverde and Rubio-Ramirez (2007), An and Schorfheide (2007), Yano (2009), Yano *et al.* (2010), 矢野他 (2011) 等、既に多数に渡るため、関連文献は

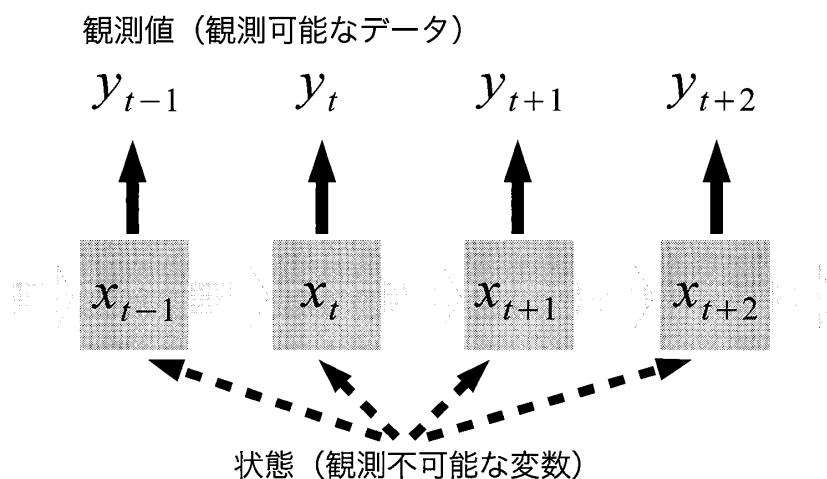


図1 状態空間モデル.

DeJong and Dave (2011) 等を参照されたい.

本稿の構成は以下の通りである. 第2節で粒子フィルタの基礎と適用例, 第3節で粒子固定ラグ平滑化, 第4節で粒子固定ラグ平滑化へのリサンプル・ムーブ法の適用を解説する. その後, 第5節で自己組織化状態空間モデルとパラメータ推定について解説し, 第6節で粒子フィルタによるRBCの状態推定を説明する. 最後にまとめと今後の課題を述べる.

2. 粒子フィルタ

2.1 状態空間モデル

粒子フィルタは, 以下に述べる非線形・非ガウス状態空間モデルを推定対象として想定している. まず, 時点 t の状態 x_t が状態 x_{t-1} に依存して確率的に決まる場合を考える. さらにその状態 x_t から確率的に観測値 y_t が生成されているとする.²⁾ なお, 状態は未知 (直接観測することが不可能) であり, 観測値は既知 (直接観測が可能) であると仮定する. この状況を表すと図1となる.

状態 x_t の時間変化を関数 f , 状態 x_t から観測値 y_t への変換を関数 h で表すと, ここまでの設定は以下の2式に集約される.

$$x_t = f(x_{t-1}, \xi_s, v_t), \quad (2.1)$$

$$y_t = h(x_t, \xi_m, \epsilon_t). \quad (2.2)$$

式 (2.1) をシステム方程式, 式 (2.2) を観測方程式と呼び, この2式をまとめて「非線形・非ガウス状態空間モデル」もしくは「一般化状態空間モデル」という (以下, 状態空間モデル). なお, v_t はシステムノイズ, ϵ_t は観測ノイズであり, ξ_s と ξ_m はそれぞれシステム方

²⁾ 本論文では数式のボールド体はベクトルを表す.

程式と観測方程式のパラメータである。また、 $\theta' = [\xi_s', \xi_s']$ と表す。³⁾ なお式 (2.1), (2.2) の詳細は付録 A1 に記述する。

式 (2.1) は、状態 x_t が条件付き確率分布で

$$x_t \sim \mathbb{P}(x_t | x_{t-1}), \quad (2.3)$$

と表され、式 (2.2) は、観測値 y_t が条件付き確率分布で

$$y_t \sim \mathbb{P}(y_t | x_t) \quad (2.4)$$

と表されることを示している。なお、ここで $\sim$ は「条件付き確率分布 $\mathbb{P}(x_t | \cdot)$ から確率変数 x_t がサンプリングされること」を示す。なお、本稿では確率分布を $\mathbb{P}(\cdot)$ 、確率密度関数を $p(\cdot)$ と表す。ただし、以下では誤解が生じない限り確率分布と確率密度関数を単に分布と表現する。

本稿で我々が考える問題は、時点 1 から時点 t までの観測値 y_t を所与とした時、式 (2.1), (2.2) の状態空間モデルを用いて、「未知の状態 x_t を推定する」ことである。このような問題を状態推定という。以下、時点 1 から時点 t までの観測値 y_t の組を $y_{1:t} = \{y_1, y_2, \dots, y_t\}$ と表すことにする。

状態推定では、状態 x_t と観測値 $y_{1:t} = \{y_1, y_2, \dots, y_t\}$ を用いて、予測分布 (predictive distribution)、フィルタ分布 (filtering distribution)、平滑化分布 (smoothing distribution) をそれぞれ以下のように定義する。

$$\begin{aligned} \text{予測分布: } x_{t|t-1} &= \mathbb{P}(x_t | y_{1:(t-1)}), \\ \text{フィルタ分布: } x_{t|t} &= \mathbb{P}(x_t | y_{1:t}), \\ \text{平滑化分布: } x_{t|T} &= \mathbb{P}(x_t | y_{1:T}), \end{aligned} \quad (2.5)$$

ただし、 $1 \leq t \leq T$ である。予測分布は $y_{1:(t-1)}$ (時点 1 から時点 $t-1$ までの観測値) を所与として状態 x_t の分布を求めること、フィルタ分布は $y_{1:t}$ (時点 1 から時点 t までの観測値) を所与として状態 x_t の分布を求めること、平滑化分布は $y_{1:T}$ (時点 1 から時点 T までの観測値) を所与として状態 x_t の分布を求めることである。

2.2 粒子フィルタ

粒子フィルタで特徴的な発想は、推定すべき分布 (予測分布・フィルタ分布・平滑化分布) を「状態ベクトル x_t^i とその点に対応する『尤度』 w_t^i の集合で近似する」ところにある。⁴⁾ 本来であれば、推定すべき分布の解析解が得られれば最も良いはずであるが、

³⁾ 行列・ベクトルの右肩に $'$ を書く場合、転置を意味する。

⁴⁾ 尤度については付録 A1 で解説する。

粒子フィルタが想定する非線形・非ガウス状態空間モデルではそのような解析解が得られることはほとんどない。そのため、モンテカルロ法で状態ベクトル $\mathbf{x}_t^i$ を発生させ、さらにその尤度 w_t^i を適切に評価して経験分布を構成する。

たとえば、フィルタ分布 $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ を経験分布で近似することを考えよう。粒子フィルタでは以下のような記法が採用されることが多い。

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \simeq \sum_{i=1}^M \frac{w_t^i}{\sum_{i=1}^M w_t^i} \delta(\mathbf{x}_t - \mathbf{x}_{t|t-1}^i). \quad (2.6)$$

ここで、 $\mathbf{x}_{t|t-1}^i$ は式 (2.1) を用いてモンテカルロ法で発生させた状態ベクトル、 w_t^i は $\mathbf{x}_{t|t-1}^i$ に対応する尤度、 M はモンテカルロ法の試行回数、 $\delta(\cdot)$ はディラックのデルタ関数である。なお、本稿では $\simeq$ は分布の近似を意味する。また、以下では状態ベクトルを単に状態と呼ぶ。式 (2.6) の右辺が $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ の経験分布であり、モンテカルロ法で生成した状態 $\mathbf{x}_{t|t-1}^i$ と尤度 w_t^i から構成されている。なお、この分野では、状態 $\mathbf{x}_{t|t-1}^i$ を「粒子」、尤度 w_t^i を粒子の「重み」と呼ぶ習慣があり、これから述べるアルゴリズムが「粒子フィルタ (particle filter)」と呼ばれる由来となっている。⁵⁾

また、粒子フィルタでは、各ステップで「重みに比例して粒子をリサンプリング (復元抽出)」する (リサンプリングの詳細は 2.3 節)。リサンプリングされた粒子を $\hat{\mathbf{x}}_t^i$ とすると、それらの重みはすべて $1/M$ に規格化され、フィルタ分布 $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ の経験分布は以下のよう書き直すことができる。

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) \simeq \frac{1}{M} \sum_{i=0}^M \delta(\mathbf{x}_t - \hat{\mathbf{x}}_t^i). \quad (2.7)$$

なお、予測分布・平滑化分布も式 (2.6), (2.7) と同様の記法で表現できる。ただし、デルタ関数の定義⁶⁾に厳密に従えば、式 (2.6), (2.7) は数学的には誤った表記であるが、粒子フィルタを解説する際に便利であるためしばしば用いられる。本稿でもその慣例に従いこれらの記法を採用する。

以上で粒子フィルタにおける経験分布の記法が分かった。それらと状態空間モデル (式 (2.1), (2.2)) を用いてどのようにすれば状態推定が可能になるだろうか？

粒子フィルタは基本的には (1) 予測, (2) 尤度計算, (3) リサンプリングで構成されるモンテカルロ法によるアルゴリズムである。具体的には、時点 $t-1$ の粒子 $\hat{\mathbf{x}}_{t-1}^i$ と時点 t の

⁵⁾ 粒子フィルタの導出については付録 A1 で解説する。

⁶⁾ ディラックのデルタ関数の定義は以下の通り。

$$\begin{aligned} \delta(x) &= 0, \text{ if } x \neq 0, \\ \int_{-\infty}^{\infty} \delta(x) dx &= 1. \end{aligned}$$

定義から分かるようにデルタ関数を適切に用いるには本来は積分記号が必要になるはずである。

観測値 $\mathbf{y}_t$, 及び式 (2.1), (2.2) のパラメータ θ が所与であるとして, 時点 t において以下の手順を実行する.

1. $i = 1$ から $i = M$ まで以下を繰り返す

- (a) 予測: $\mathbf{x}_{t|t-1}^i \sim p(\mathbf{x}_t | \mathbf{x}_{t-1})$. つまり, 式 (2.1) と所与の $\hat{\mathbf{x}}_{t-1}^i$ と θ を用いて, モンテカルロ法により予測分布 $p(\mathbf{x}_t | \mathbf{x}_{t-1})$ に従う粒子 $\mathbf{x}_{t|t-1}^i$ を発生させる
- (b) 尤度計算: $w_t^i = p(\mathbf{y}_t | \mathbf{x}_{t|t-1}^i)$. つまり, 式 (2.2), 観測値 $\mathbf{y}_t$, 粒子 $\mathbf{x}_{t|t-1}^i$ を用いて, 重み (尤度) w_t^i を計算する.

2. リサンプリング: w_t^i , ($i = 1, \dots, M$) に比例して $\mathbf{x}_{t|t-1}^i$, ($i = 1, \dots, M$) をリサンプリングし, リサンプリングされた粒子 $\hat{\mathbf{x}}_t^i$ を用いて式 (2.7) を計算する

観測値 $\mathbf{y}_{1:T}$ が所与である場合, この手順を時点 1 から時点 T まで繰り返す. なお, 時点 0 における状態 $\mathbf{x}_0$ の初期分布 $p(\mathbf{x}_0 | \emptyset)$ は所与であり, 時点 0 で状態 $\mathbf{x}_0^i \sim p(\mathbf{x}_0 | \emptyset)$, ($i = 1, \dots, M$) を発生させることが可能であると仮定している.⁷⁾ 言うまでもなく, ステップ 1(a) が予測分布のシミュレーション, ステップ 1(b) と 2 がフィルタ分布のシミュレーションである.

粒子フィルタで時点 1 から時点 T までの式 (2.7) が得られれば, そこからフィルタ分布の状態 $\mathbf{x}_t$ の代表値を計算することは容易である. 状態 $\mathbf{x}_t$ の j 番目の要素を $x_{j,t}$, 粒子 $\hat{\mathbf{x}}_t^i$ ($i = 1, \dots, M$) の j 番目の要素を $\hat{x}_{j,t}^i$, ($i = 1, \dots, M$) とすると, $x_{j,t}$ の代表値は $\hat{x}_{j,t}^i$, ($i = 1, \dots, M$) の期待値⁸⁾, もしくは $\hat{x}_{j,t}^i$, ($i = 1, \dots, M$) の中央値を求めれば良い. また, 粒子フィルタでは信用区間 (credible interval) を求めることも容易である. なぜなら式 (2.7) を見れば明らかのように, 時点 t における粒子 $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) が分布を表しているため, たとえば 95% 信用区間はリサンプリング後の粒子から 2.5% 百分位数と 97.5% 百分位数を求めればよい.

パラメータ θ の尤度関数 $\mathcal{L}$ は, $\mathbf{y}_t$ の予測分布 $p(\mathbf{y}_t | \mathbf{y}_{1:(t-1)}, \theta)$ を用いて以下のように表される.

$$\mathcal{L}(\theta | \mathbf{y}_{1:t}) = \prod_{t=1}^T p(\mathbf{y}_t | \mathbf{y}_{1:(t-1)}, \theta) \quad (2.8)$$

さらに粒子の重み w_t^i を用いて尤度関数 $\mathcal{L}$ は以下のように近似できる.

$$\mathcal{L}(\theta | \mathbf{y}_{1:t}) \simeq \prod_{t=1}^T \frac{1}{M} \sum_{i=1}^M w_t^i, \quad (2.9)$$

⁷⁾ $\emptyset$ は空集合 (つまり観測値がまったくないこと) を表す. $p(\mathbf{x}_0 | \emptyset)$ が適切に得られるかどうかは, 粒子フィルタに限らずベイズ統計学全般において必要な問題であるが, 本稿では取り扱わない.

⁸⁾ リサンプリングされた粒子 $\hat{\mathbf{x}}_t^i$ を用いる場合, 単純に平均値を計算すれば期待値となる.

よって両辺で対数を取れば、粒子フィルタにおけるパラメータ θ の対数尤度関数の推定量は以下のように得られる。

$$\log \mathcal{L}(\theta | \mathbf{y}_{1:t}) \simeq \sum_{t=1}^T \log \left(\sum_{i=1}^M w_t^i \right) - T \log M, \quad (2.10)$$

ただし、 $\log$ は自然対数を表す。

式 (2.10) の右辺は重み w_t^i から構成されていることから分かるようにモンテカルロ法による誤差を含む推定量である。⁹⁾ つまり (対数) 尤度関数の正確な推定量を得るためにはかなり大量の粒子を発生させる (つまり多くのモンテカルロ試行を実施する) 必要があるため、計算負荷が大きい。さらに式 (2.10) は重みで定式化されているため微分を計算できない。そのため Newton 法などの微分を用いる非線形関数最適化アルゴリズムを用いることができない。以上のような理由で粒子フィルタでは最尤法によるパラメータ推定が困難である。(対数) 尤度関数を用いずにパラメータ推定を行う方法として第 5 節で述べる自己組織化状態空間モデルがある。

ここまでの記述をまとめると粒子フィルタのアルゴリズムは以下の通りである。

1. 観測値 $\mathbf{y}_{1:T}$, パラメータ θ , 初期分布 $p(\mathbf{x}_0 | \emptyset)$ を所与とし、時間のインデックス $t = 0$ を設定する
2. 状態 $\mathbf{x}_0^i \sim p(\mathbf{x}_0 | \emptyset)$, ($i = 1, \dots, M$) を発生させる
3. $t \leftarrow t + 1$ とする
 - (a) $i = 1$ から $i = M$ まで以下を繰り返す
 - i. 必要な疑似乱数を発生させる
 - ii. 疑似乱数と式 (2.1) により $\mathbf{x}_t$ の予測分布を発生させる: $\mathbf{x}_{t|t-1}^i \sim p(\mathbf{x}_t | \mathbf{x}_{t-1})$
 - iii. 式 (2.2) により重み (尤度) w_t^i を計算する: $w_t^i = p(\mathbf{y}_t | \mathbf{x}_{t|t-1}^i)$
 - (b) リサンプリング: 重み w_t^i , ($i = 1, \dots, M$) に比例して粒子 $\mathbf{x}_{t|t-1}^i$, ($i = 1, \dots, M$) をリサンプリングし、リサンプリングされた粒子 $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) から式 (2.7) を計算する
 - (c) リサンプリングされた粒子 $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) を用いて代表値 (期待値, 中央値等) と信用区間を算出する
4. 所定の回数 T に到達していればパラメータ θ に対する対数尤度関数 (2.10) を計算して終了。そうでない場合は 3. に戻る

⁹⁾ Higuchi (1995) が式 (2.10) におけるモンテカルロ法による誤差について分析している。

このように粒子フィルタは非常に単純なアルゴリズムであり、プログラムを組むことも容易である。ただし、コンピュータの処理性能はそれなりに要求される。CPU への負荷が大きいのは疑似乱数発生とリサンプリングであるが、それもマルチコア CPU や GPGPU 等の発展により徐々に解決されてきており、今後はさらに多くの分野に適用されていくと見込まれている。

2.3 リサンプリングと粒子の縮退

前節で述べたように粒子フィルタのアルゴリズムでは基本的に各時点 t に粒子のリサンプリングが行われる。その理由は「リサンプリングを行わないと、時間 t の増加に従い粒子の分散が次第に増大していつてしまう」からである (このことはシミュレーションによって簡単に確認できる)。粒子の分散が大きくなると、少数の粒子の重みだけ大きく、その他の多数の粒子の重みはほとんどゼロになってしまう。こうなると、「少数の粒子だけで分布 (予測分布・フィルタ分布・平滑化分布) を近似する」ことになり、分布の近似精度が極端に悪化することが知られている。この問題を「粒子の縮退 (degeneracy)」という。

この場合の粒子の縮退の解決策として提案された手法として、(1) リサンプリング、(2) 補助粒子フィルタ (auxiliary particle filter) が代表的である。¹⁰⁾

リサンプリングとは、既述の通り「粒子 $\mathbf{x}_{t|t-1}^i$, ($i = 1, \dots, M$) を重み, w_t^i , ($i = 1, \dots, M$) に比例して復元抽出すること」である。リサンプリングの手法としては Kitagawa (1993, 1996) で提案された「ランダム・サンプリング」が標準的であり、そのアルゴリズムは以下の通りである。

1. 重み w_t^i , $i = \{1, \dots, M\}$ を所与とし、インデックス $i = 0$ を設定する
2. インデックス $i \leftarrow i + 1$ とする
 - (a) $u_t^i = \text{Uniform}[0, 1]$
 - (b) $(1/C) \sum_{l=1}^{j-1} w_t^l < u_t^i \leq (1/C) \sum_{l=1}^j w_t^l$ を満たす j を探す
 - (c) その j に該当する $\mathbf{x}_{t|t-1}^j$ をリサンプリング後の粒子 $\hat{\mathbf{x}}_t^i$ として採用する
3. i が粒子数の上限 M に到達していれば終了。そうでない場合は 2. に戻る

ただし、 $\text{Uniform}[0, 1]$ は 0 以上 1 以下の一様乱数、 $C = \sum_{l=1}^M w_t^l$ である。

ただし、このランダム・サンプリングは比較的計算負荷が大きいことが知られており、Kitagawa (1993, 1996) では計算負荷軽減のため層化抽出を利用した手法を提案している。

¹⁰⁾ Tanizaki (2001) では粒子フィルタで用いられる重点サンプリングとそれ以外のサンプリング (棄却サンプリング, Metropolis-Hastings アルゴリズム) の紹介と比較が行われている。

Douc (2005) はそれらの手法の比較検討を行っている。なお、粒子の分散がそれほど大きくなければリサンプリングは必ずしも必須ではない。そのため、分散がある閾値を超えるまではリサンプリングを行わないアルゴリズムも提案されている (Liu and Chen (1998)). しかし、近年ではある程度コンピュータの処理性能が向上したため、通常は毎ステップ毎にリサンプリングが実行されることが多い。

粒子のリサンプリングを行った場合、ほぼ確実に起こる現象が「粒子の多様性低下」である。たとえば、リサンプリング前に粒子 $\mathbf{x}_{t|t-1}^i$, ($i = 1, \dots, M$) の種類が M であった場合、リサンプリング後の粒子 $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) の種類は M 以下になる。これは「重み w_t^i の大きい粒子が複数回 (リサンプリングで) 選択される可能性が高い」からである。このような「粒子の多様性低下」は、後述の粒子フィルタベースの固定ラグ平滑化や自己組織化状態空間モデルでは深刻な問題となる。なお、「粒子の種類が少数となり、分布をうまく近似できなくなる」という意味で、この問題も「粒子の縮退」と呼ぶことが多い。なお、「分布を構成するのに有効な粒子が少数になり、分布がうまく近似できない」という意味では両者 (粒子の分散増大による縮退 [本節] と粒子の多様性の低下による縮退 [第 3 節]) は同じであるが、問題の発生原因は異なる。

補助粒子フィルタは Pitt and Shephard (1999) によって提案されたアルゴリズムであり、時点 $t-1$ に得られた粒子 $\mathbf{x}_{t-1}^i$, ($i = 1, \dots, M$) に補助インデックス k を付加し、時点 t の観測値 $\mathbf{y}_t$ に基づき、望ましい $\mathbf{x}_{t-1}^i$, ($i = 1, \dots, M$) の選別した上で、予測分布 $p(\mathbf{x}_t | \mathbf{y}_{1:t-1})$ を発生させ、なおかつ重み w_t^i , ($i = 1, \dots, M$) を修正するという手法である (詳しくは Pitt and Shephard (1999) を参照されたい)。このアイディアは縮退を解決する有効な手段である¹¹⁾が、チュートリアルという本稿の趣旨に沿わないため省略する。補助粒子フィルタの近年、急速な発展を遂げており、多くの分野で採用されている。近年の発展は Whiteley and Johansen (2011) にサーベイされているため、興味のある読者は参照されたい。

2.4 適用例

この節では、初期分布、パラメータを予め与え、モンテカルロ法で状態空間モデルから状態 (真の状態) と観測値を発生させ、その観測値を元に粒子フィルタで状態を推定する。このように人為的に発生させた観測値の場合、真の状態が分かっているため、粒子フィルタで推定した状態との比較が可能である。なお、このような手法をしばしばデータ生成 (Data Generation) という。手順としては

1. 初期分布 $p(\mathbf{x}_0 | \emptyset)$, パラメーター θ を与えて、所与の状態空間モデルからモンテカルロ法で状態 (真の状態) $\mathbf{x}_{1:T}$ と観測値 $\mathbf{y}_{1:T}$ を発生させる

¹¹⁾ なお、補助粒子フィルタを採用する場合、リサンプリングは必要ない。

2. 初期分布 $p(\mathbf{x}_0|\emptyset)$, パラメーター θ を所与とし, 発生させた観測値 $\mathbf{y}_{1:T}$ と所与の状態空間モデルに粒子フィルタを適用して, 状態を推定する (粒子フィルタによって推定された粒子は $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) であり, それが推定された状態の経験分布を構成する)
3. 真の状態 $\mathbf{x}_t$ と推定された状態 $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) の代表値を比較し, 粒子フィルタが適切に真の状態を推定できているか確認する. (必要に応じて) 信用区間を計算する

本稿における適用例は以上の手順を踏んで行っている. なお, 代表値としてはすべて期待値を用い, 必要に応じて 95% 信用区間を表示する.¹²⁾

第 1 の例として, 以下の線形・ガウス型状態空間モデルの状態推定を考える. 状態をスカラー x_t , 観測値をスカラー y_t として

$$\begin{aligned} x_t &= x_{t-1} + v_t \\ y_t &= x_t + \epsilon_t, \end{aligned} \quad (2.11)$$

ただし, システムノイズ v_t , 観測ノイズ ϵ_t はそれぞれ正規乱数であるとする.

式 (2.11) で, $x_0 = 0$ を初期分布, システムノイズを $v_t \sim N(0, 1)$, 観測ノイズを $\epsilon_t \sim N(0, 0.25)$, $T = 50$ としてデータ生成を行った. さらに式 (2.11), 初期分布, システムノイズ・観測ノイズの期待値・分散は所与であるとして, 観測値 $y_{1:T}$ から粒子フィルタで状態を推定した. 結果を図 2 に示す.¹³⁾ 図中の線がデータ生成によって生成された真の状態 x_t (図の “True State”), 丸と破線が粒子フィルタによって推定された粒子 $\hat{x}_t^i$, ($i = 1, \dots, M$) の期待値 (図の “Estimated State”) である. この結果から, 粒子フィルタによって推定された状態はデータ生成による真の状態をよく追跡 (track) していることが分かる. また, 状態 $\hat{x}_t$ の信用区間を図 3 に示す. なお, このように粒子フィルタで信用区間を計算することは非常に容易であるが, 表示が煩雑になるため, 以下の図では省略する.

ここまで述べた式 (2.11) のような線形・ガウス型状態空間モデルの状態推定はカルマン・フィルタでも可能であり, 粒子フィルタを使う必要はない. そこで第 2 の例としてカルマン・フィルタでは推定の難しい非線形状態空間モデルの状態推定を考えよう. Kitagawa (1993, 1996), Gordon *et al.* (1993) で用いられた非線形・ガウス型状態空間モデルは以下の通りである. 状態をスカラー x_t , 観測値をスカラー y_t として

$$\begin{aligned} x_t &= \frac{1}{2}x_{t-1} + 25\frac{x_{t-1}}{1 + x_{t-1}^2} + 8\cos(1.2t) + v_t \\ y_t &= \frac{1}{2}x_t^2 + \epsilon_t, \end{aligned} \quad (2.12)$$

¹²⁾ 代表値として中央値を用いても何ら問題はない.

¹³⁾ 各時点 t では 1 万個の粒子 ($M = 10,000$) を発生させた. 以下, シミュレーションではすべて同様に行った.

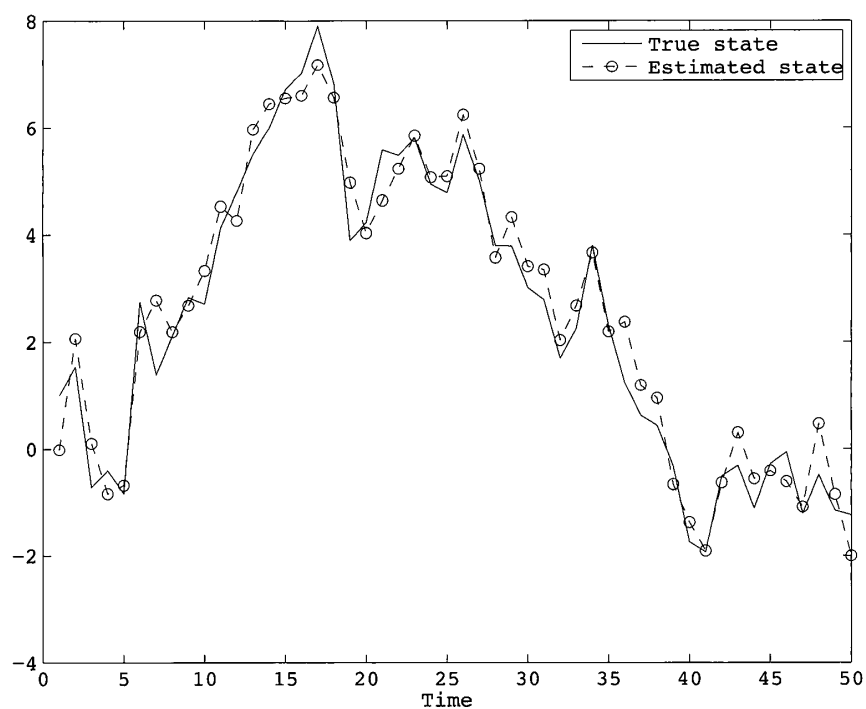


図2 1次線形ガウス状態空間モデルの推定結果.

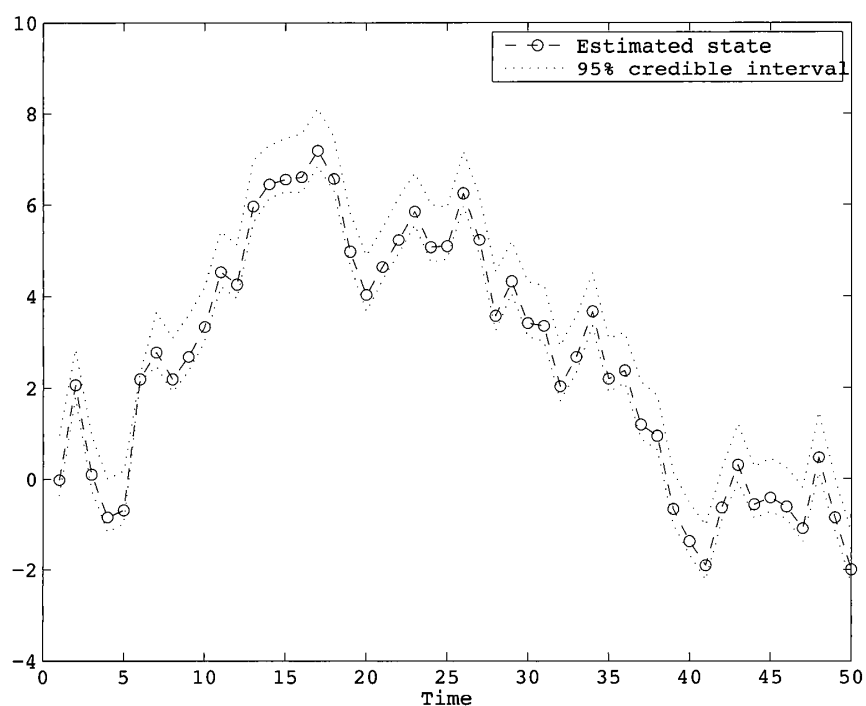


図3 1次線形ガウス状態空間モデル推定の信用区間.

ただし、システムノイズ v_t 、観測ノイズ ϵ_t はそれぞれ正規乱数であるとする。式 (2.12) で、 $x_0 = 0$ を初期分布、 $v_t \sim N(0, 1)$ 、 $\epsilon \sim N(0, 10)$ 、 $T = 100$ としてデータ生成を行った。この状態空間モデルは2乗や余弦 $\cos$ が含まれていることから明らかなようにカルマン・フィルタでの推定が非常に難しいが、このような場合も粒子フィルタは線形近似等なしに推

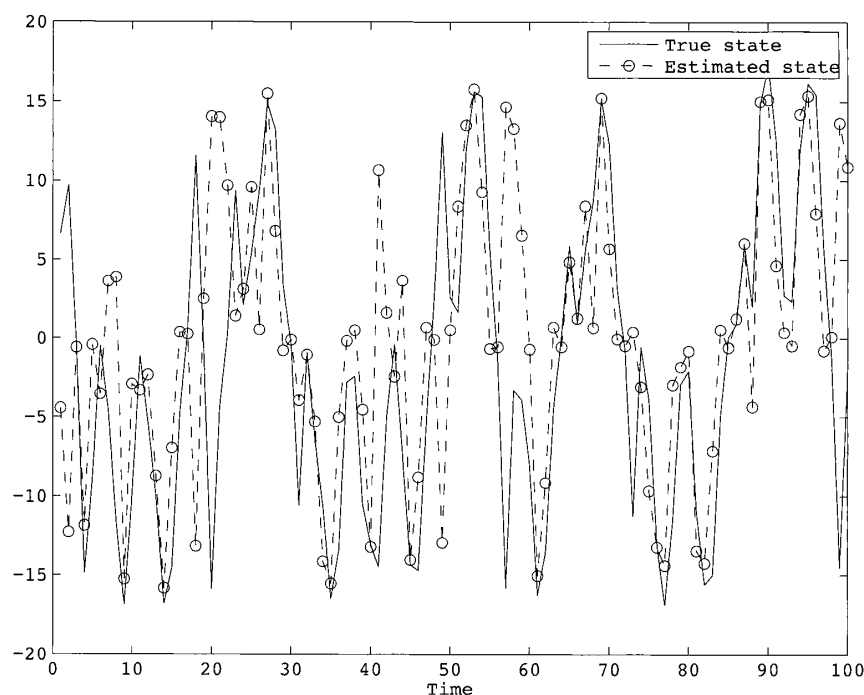


図4 1次非線形状態空間モデルの推定結果.

定が可能である.

データ生成による観測値を用い, 式 (2.12)・初期値 x_0 及びシステムノイズ・観測ノイズの期待値・分散は既知であると仮定して, 観測値 $y_{1:T}$ から粒子フィルタで状態を推定した. 図4中の線がデータ生成によって生成された真の状態 x_t (図の “True State”), 丸と破線が粒子フィルタによって推定された粒子 $\hat{x}_t^i$, ($i = 1, \dots, M$) の期待値 (図の “Estimated State”) である. 図から分かるように非線形・ガウス型状態空間モデル (2.12) によって生成された状態が粒子フィルタによりうまく追跡されていることが分かる.

第3の例としてファイナンスの実証でよく用いられる確率的ボラティリティモデル (Stochastic Volatility Model, 以下 SV) の推定を行う. SV は以下のように定義される.¹⁴⁾

$$\begin{aligned} x_t &= \mu + \phi x_{t-1} + v_t \\ y_t &= \epsilon_t \exp\left(\frac{x_t}{2}\right), \end{aligned} \quad (2.13)$$

ここで x_t はボラティリティ, y_t は観測値, システムノイズ v_t , 観測ノイズ ϵ_t はそれぞれ正規乱数, μ, ϕ は定数である. データ生成は $x_0 = 0$ を初期分布, $v_t \sim N(0, 0.5)$, $\epsilon \sim N(0, 1)$, $\mu = 0$, $\phi = 0.98$, $T = 200$ として行った. これまでと同様に状態空間モデル, 初期分布, パラメータは所与であるとして, 観測値 $y_{1:T}$ から粒子フィルタで状態を推定した. 真の状

¹⁴⁾ 矢野・佐藤 (2006) では SV モデルを粒子フィルタ (モンテカルロ・フィルタ) で推定し, 円ドル為替レートの実証分析を行っている.

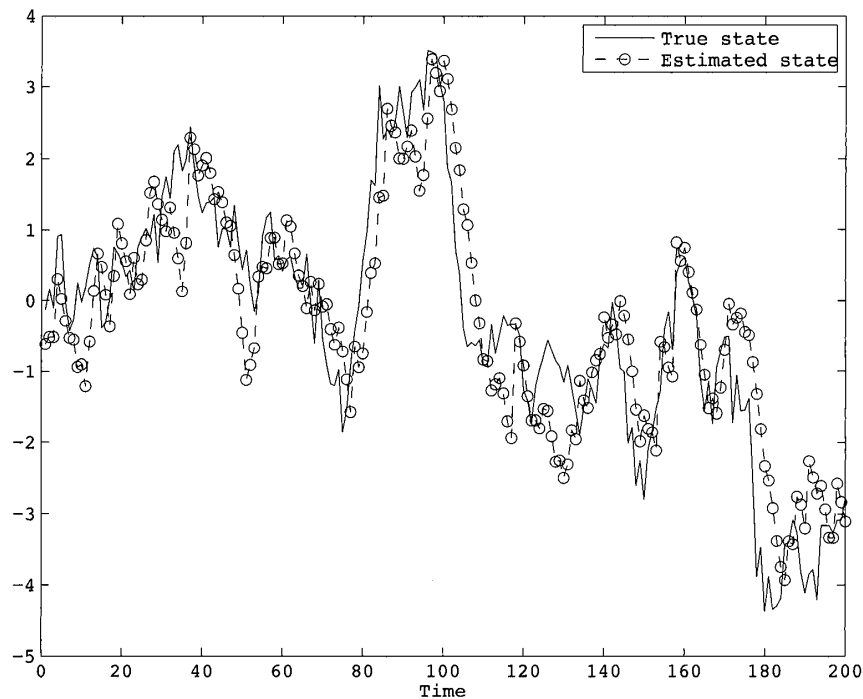


図5 確率的ボラティリティ・モデルの推定結果.

態 x_t と $\hat{x}_t^i$, ($i = 1, \dots, M$) の期待値を図5に示す(線・丸と破線の意味は図2と同じ). この場合も真の状態を粒子フィルタがうまく追跡できている.

ここまでの非線形例で, いずれも状態推定に「1次の線形近似」等は不要である点を強調しておきたい. かつては非線形・非ガウス状態空間モデルを推定するには拡張カルマンフィルタ等の線形近似等の近似が必要であったが, 粒子フィルタについてはそのような作業は不要であり, 所与の状態空間モデルを粒子フィルタで推定するだけで良い.

このように粒子フィルタは線形・ガウス型状態空間から非線形・非ガウス型状態空間モデルまで柔軟に適用できるアルゴリズムなのである.

3. 粒子平滑化

さらに Kitagawa (1993, 1996) は粒子フィルタベースの平滑化(以下, 粒子平滑化)の手法も提案している(提案された手法は2つあるが, 極めてシンプルな「粒子を保存するだけ」の平滑化について述べる). この手法では時点 t のとき, 時点 t から時点 $t-L$, $1 \leq L \leq t$ までの粒子を保存し, 時点 t の粒子の重みに比例してリサンプリングするだけで平滑化分布が得られる. ラグ L が固定されているため「固定ラグ平滑化」と呼ばれる. この手法は計算負荷が粒子フィルタとまったく同じであり, 要求されるのは粒子を保存するメモリ空間だけであるという利点がある. ただし, 後述の理由により L は20から30程度の大きさ

にすることが多い。¹⁵⁾

粒子固定ラグ平滑化のアルゴリズムは以下の通りである.

1. 観測値 $\mathbf{y}_{1:T}$, パラメータ $\boldsymbol{\theta}$, 初期分布 $p(\mathbf{x}_0|\emptyset)$ を所与とし, 時間のインデックス $t = 0$ を設定する
2. 状態 $\mathbf{x}_0^i \sim p(\mathbf{x}_0|\emptyset)$, $(i = 1, \dots, M)$ を発生させる
3. $t \leftarrow t + 1$ とする
 - (a) $\{\mathbf{0}_t^i, \hat{\mathbf{x}}_{t-1}^i, \dots, \hat{\mathbf{x}}_{t-L}^i\}$ を保存する配列 $\mathbf{S}^i$, $(i = 1, \dots, M)$ を用意する
 - (b) $i = 1$ から $i = M$ まで以下を繰り返す
 - i. 必要な疑似乱数を発生させる
 - ii. 疑似乱数と式 (2.1) により $\mathbf{x}_t$ の予測分布を発生させる: $\mathbf{x}_{t|t-1}^i \sim p(\mathbf{x}_t|\mathbf{x}_{t-1})$
 - iii. 式 (2.2) により重み (尤度) w_t^i を計算する: $w_t^i = p(\mathbf{y}_t|\mathbf{x}_{t|t-1}^i)$
 - iv. $\mathbf{x}_{t|t-1}^i$ を $\mathbf{S}^i$ に収納し, $\mathbf{S}^i = \{\mathbf{x}_{t|t-1}^i, \hat{\mathbf{x}}_{t-1}^i, \dots, \hat{\mathbf{x}}_{t-L}^i\}$ とする
 - (c) リサンプリング: $\mathbf{x}_{t|t-1}^i$, $(i = 1, \dots, M)$ に対応する w_t^i , $(i = 1, \dots, M)$ に比例して該当する配列 $\mathbf{S}^i$, $(i = 1, \dots, M)$ をリサンプリング (復元抽出) する
 - (d) $\mathbf{S}^i$, $(i = 1, \dots, M)$ 内の粒子 $\hat{\mathbf{x}}_{t-L}^i$, $(i = 1, \dots, M)$ から代表値 (平均値, 中央値等) と信用区間を算出
4. 所定の回数 T に到達していればパラメータ $\boldsymbol{\theta}$ に対する対数尤度関数 (2.10) を計算して終了. そうでない場合は 3. に戻る

粒子固定ラグ平滑化の欠点は L を大きく設定すると $\mathbf{S}^i$, $(i = 1, \dots, M)$ 内の平滑化分布を表す粒子の多様性が減少し, 平滑化分布がうまく近似できなくなる点である. これはリサンプリング毎に $p(\mathbf{x}_{t-l}|\mathbf{y}_{1:t})$, $l = \{1, \dots, L\}$ を構成する粒子の多様性が低下, つまり粒子が縮退し, 経験分布がうまく構成できなくなるためである.

この問題を明示するため, 図 6 に $L = 50$ とした場合の平滑化分布 $p(\mathbf{x}_{t-L}|\mathbf{y}_{1:t})$ を示す. 図を見るとヒストグラムの 1.6 の部分に多くの粒子が集中し, それ以外の部分には極端に粒子が少ない分布になっている. 図 6 の経験分布は粒子の縮退が発生していることを示している. このような問題を避けるために, 一般に L を 20 から 30 程度に制限するか, 次節で述べるようにリサンプル・ムーブ法を適用することが多い.

¹⁵⁾ 粒子固定ラグ平滑化の導出は付録 A2 で解説する.

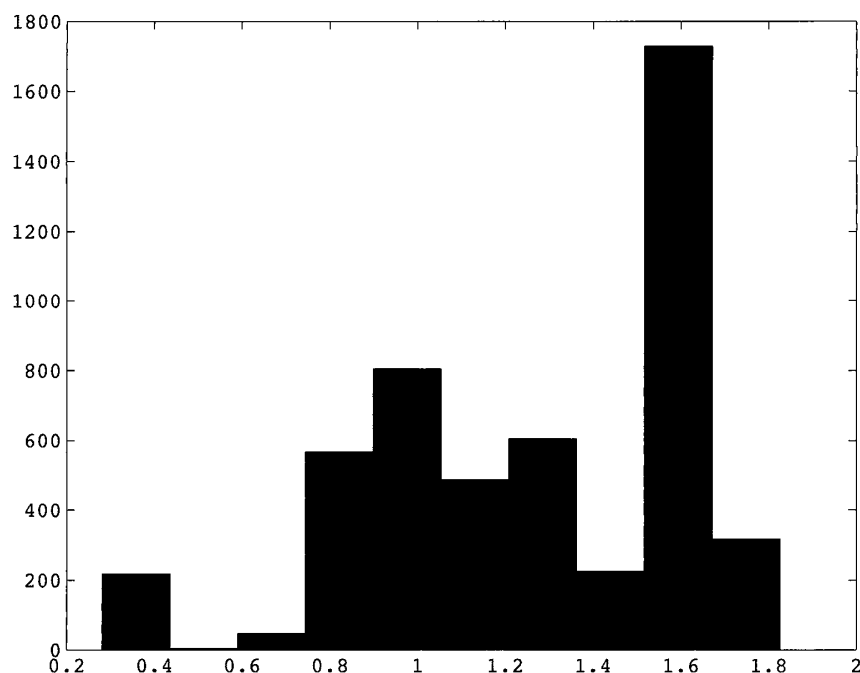


図 6 粒子固定ラグ平滑化の結果.

4. リサンプル・ムーブ法の利用

前節で述べたように Kitagawa (1993, 1996) が提案した粒子固定ラグ平滑化では粒子の縮退が起こるため、一般に L は 20 から 30 程度に制限されることが多いという欠点があった。これを解決するためにはいくつかやり方があるが、有力な手法は Gilks and Berzuini (2001) で提案されたりサンプル・ムーブ法である。

リサンプリング後の粒子から構成されるフィルタ分布を $p(\hat{\mathbf{x}}_t|\mathbf{y}_t)$ 、それに対して不変である分布を $p(\tilde{\mathbf{x}}_t|\mathbf{y}_t)$ と表し、マルコフ推移カーネル K が適切に設定されていれば、マルコフ連鎖モンテカルロ法 (Markov Chain Monte Carlo method, MCMC)¹⁶⁾ で知られているように以下が成り立つ。

$$p(\tilde{\mathbf{x}}_t|\mathbf{y}_t) = \int K(\tilde{\mathbf{x}}_t|\hat{\mathbf{x}}_t)p(\hat{\mathbf{x}}_t|\mathbf{y}_t)d\mathbf{x}_t \quad (4.1)$$

この式は、リサンプリング後の粒子から構成されるために縮退したフィルタ分布 $p(\hat{\mathbf{x}}_t|\mathbf{y}_t)$ にマルコフ推移カーネルを適用することで粒子を確率的に移動させることを意味し、粒子 $\hat{\mathbf{x}}_t^i$, ($i = 1, \dots, M$) がマルコフ推移カーネル K で確率的に移動させられるため、粒子の多様性が増加することを意味している。

この発想に基づき Gilks and Berzuini (2001) が提案したアルゴリズムは以下のようになっている。

¹⁶⁾ MCMC 全般については渡部 (2000), 大森 (2001, 2005), 和合 (2005) 等を参照されたい。

1. リサンプル: 粒子 $\mathbf{x}_{t|t-1}^i$ ($i = 1, \dots, M$) を重み w_t^i , ($i = 1, \dots, M$) に比例してリサンプリングし, $\hat{\mathbf{x}}_t^i$ ($i = 1, \dots, M$) とする
2. 移動(ムーブ): マルコフ推移カーネルを適用し, $\hat{\mathbf{x}}_t^i$ ($i = 1, \dots, M$) を $\tilde{\mathbf{x}}_t^i$ ($i = 1, \dots, M$) に移動させる

ここから分かるようにリサンプルとムーブ(粒子の移動)から構成されるアルゴリズムであるため, 「リサンプル・ムーブ」と呼ばれている.

不変分布を与えるマルコフ推移カーネルの代表例が MCMC である. 粒子の多様性低下を防ぐという目的のためには Gibbs サンプルング・Metropolis-Hastings サンプルングどちらを用いても構わないが, 本稿では Random-walk Metropolis-Hastings サンプルング(以下, RWMH)を用いてシミュレーションする. 各時点 t に実行すべきアルゴリズムを以下に示す(具体的には粒子固定ラグ平滑化アルゴリズム 3(c) の後に以下のアルゴリズムを追加する).

1. 時点 t における「リサンプリング後の粒子」 $\hat{\mathbf{x}}_t^i$, $i = \{1, \dots, M\}$ を用意する
2. インデックス $i = 1$ から $i = M$ まで繰り返す
 - (a) 以下のランダムウォークに従い, 粒子 $\hat{\mathbf{x}}_t^i$ から新しい粒子 $\tilde{\mathbf{x}}_t^i$ 生成する:

$$\tilde{\mathbf{x}}_t^i \sim \hat{\mathbf{x}}_t^i + \mathbf{e}_t, \quad \mathbf{e}_t \sim N(\boldsymbol{\theta}, c\boldsymbol{\Sigma}_m) \quad (4.2)$$

- (b) 採択確率を計算する

$$r = \frac{P(\mathbf{y}_t|\tilde{\mathbf{x}}_t^i)P(\tilde{\mathbf{x}}_t^i \leftarrow \hat{\mathbf{x}}_t^i)}{P(\mathbf{y}_t|\hat{\mathbf{x}}_t^i)P(\hat{\mathbf{x}}_t^i \leftarrow \tilde{\mathbf{x}}_t^i)} \quad (4.3)$$

- (c) 新しい粒子の候補 $\tilde{\mathbf{x}}_t^i$ を採択するか棄却するかを以下のルールにしたがって決定する

$$\tilde{\mathbf{x}}_t^i = \begin{cases} \tilde{\mathbf{x}}_t^i & \text{[確率 } \min(r, 1) \text{ で採択]} \\ \text{古い } \hat{\mathbf{x}}_t^i \text{ を保持} & \text{[それ以外]} \end{cases} \quad (4.4)$$

ただし, c はスケールパラメータ, $\boldsymbol{\Sigma}_m$ は分散である. この手順をリサンプリングの後に入れることで粒子の縮退をある程度防ぐことができる. なお, Gilks and Berzuini (2001) が指摘する通り $\hat{\mathbf{x}}_t^i$, $i = \{1, \dots, M\}$ は既に $p(\mathbf{x}_t|\mathbf{y}_{1:t})$ をよく近似していると考えられる場合, Burn-in は必ずしも必要ではない.¹⁷⁾

図 7 に $L = 50$ とした場合の平滑化分布 $p(\mathbf{x}_{t-L}|\mathbf{y}_{1:t})$ を示す. 図 6 と比較すると粒子が一部に偏らず, 比較的スムーズな経験分布が得られていることが分かる. これはリサンプル・ムーブ法を採用したお陰で粒子の縮退があまり深刻となっていないことを示している.

¹⁷⁾ リサンプル・ムーブ以外の手法として, Doucet *et al.* (2006) によって効率ブロックサンプリングが開発されており, 今後はこの方向での研究も進むのではないかと考えられる.

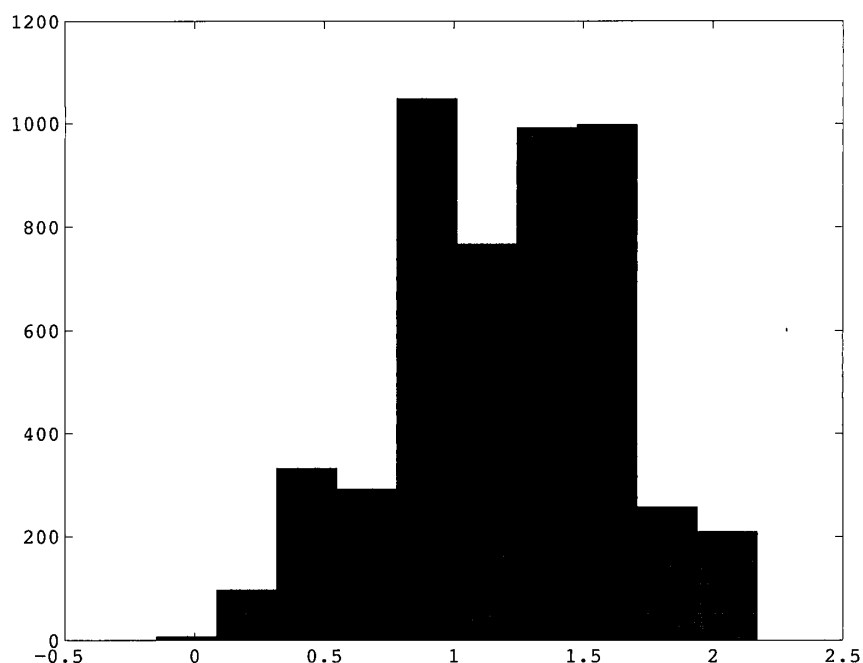


図7 リサンプル・ムーブ法を併用した粒子固定ラグ平滑化の結果.

5. 自己組織化状態空間モデル

第2節に述べたように粒子フィルタの尤度関数はモンテカルロ法の誤差を含み、なおかつ微分成分が求まらないため、対数尤度に Newton 法などの非線形最適化アルゴリズムを適用してのパラメータ θ を推定することが困難である。それらの問題に対して Kitagawa (1998) は自己組織化状態空間モデルを提案した。

この手法は非線形・非ガウス状態空間モデルの状態推定とパラメータ推定を同時に行うものである。ベイズ推定では「状態 x_t とパラメータ θ 双方が分布を有する」ため、状態空間モデルで「状態とパラメータを同時に求める」には状態とパラメータを含めた同時密度 $p(x_t, \theta | y_{1:t})$ を考える必要がある。

つまり、Kitagawa (1998) に従い状態ベクトルとパラメータを組み合わせる拡張状態ベクトルを以下のように定義すれば良い。

$$z_t = \begin{bmatrix} x_t \\ \theta \end{bmatrix}, \quad (5.1)$$

さらに拡張システム方程式と拡張観測方程式を以下のように定義する。

$$\begin{aligned} z_t &= F(z_{t-1}, v_t, \xi_s), \\ y_t &= H(z_t, \epsilon_t, \xi_m), \end{aligned} \quad (5.2)$$

ただし、

$$F(z_{t-1}, v_t, \xi_s) = \begin{bmatrix} f(x_{t-1}, v_t, \xi_s) \\ \theta \end{bmatrix}$$

また

$$H(\mathbf{z}_t, \boldsymbol{\epsilon}_t, \boldsymbol{\xi}_m) = h(\mathbf{x}_t, \boldsymbol{\epsilon}_t, \boldsymbol{\xi}_m)$$

である。式 (5.2) を自己組織化状態空間モデル (a Self-Organizing State Space model, 以下 SOSS モデル) と呼ぶ。パラメータ $\boldsymbol{\theta}$ を含むように状態ベクトルが拡張されているため、粒子フィルタを式 (5.2) に適用すれば (対数) 尤度関数の最大化なしにパラメータと状態が同時に求まる (詳しくは Kitagawa (1998) を参照)。なお、(対数) 尤度関数の定義自体は粒子フィルタと同じである。

なお、状態を求めるとは、 $p(\mathbf{x}_t, \boldsymbol{\theta} | \mathbf{y}_{1:t})$ の周辺密度 $p(\mathbf{x}_t, | \mathbf{y}_{1:t})$ を計算することである。

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) = \int p(\mathbf{x}_t, \boldsymbol{\theta} | \mathbf{y}_{1:t}) d\boldsymbol{\theta} \quad (5.3)$$

また、パラメータを求めるとは、同じく周辺密度 $p(\boldsymbol{\theta}_t, | \mathbf{y}_{1:t})$ を計算すること：

$$p(\boldsymbol{\theta} | \mathbf{y}_{1:t}) = \int p(\mathbf{x}_t, \boldsymbol{\theta} | \mathbf{y}_{1:t}) d\mathbf{x}_t \quad (5.4)$$

具体的には粒子フィルタのアルゴリズムにパラメータ $\boldsymbol{\theta}$ を含めた式 (5.2) を与え、時間 0 の時点でのパラメータ $\boldsymbol{\theta}$ のための初期粒子 $\boldsymbol{\theta}_0^i, \{i = 1, \dots, M\}$ は一様分布からサンプリングすれば良い。

$$\theta_{j,0}^i \sim \text{Uniform}[P_j - r_j, P_j + r_j], \quad (5.5)$$

ただし、 $\theta_{j,0}^i$ は $\boldsymbol{\theta}_0^i$ の j 番目の要素、 $\text{Uniform}[P_j - r_j, P_j + r_j]$ は $P_j - r_j$ から $P_j + r_j$ までの一様分布、 P_j は範囲 $(P_j - r_j, P_j + r_j)$ の中心、 r_j は定数である。拡張状態ベクトルは粒子フィルタを用いて推定される。つまり、対数尤度の最大化抜きで状態とパラメータを同時に求めることができる。

ただし、この手法には「パラメータの初期粒子 $\boldsymbol{\theta}_0^i$ の範囲内に (求めたい) 真のパラメータが含まれていなければならない」という重要な条件がある。それに対して Hürseler and Künsch (2001) は SOSS モデルのパラメータ $\boldsymbol{\theta}$ の初期分布を事前に決定することは困難であると指摘した (つまり、 $\boldsymbol{\theta}$ についての事前情報が少ない場合、 P_j と r_j を適切に選ぶ手段がほとんどない)。この困難が生じる理由は粒子フィルタで求められた $\hat{\boldsymbol{\theta}}$ はパラメータの初期分布 $\boldsymbol{\theta}_0$ のサブサンプルであるためである (そもそも $\hat{\boldsymbol{\theta}}$ に誤差項 (イノベーション項) がないたため、初期の粒子からリサンプリングを繰り返すと粒子数は単調に減少する)。しかし、実際には事後分布の平均、分散、分布の形が事前に分かっていることはほとんどない。そのため、初期粒子を発生させた時にその粒子の中に真のパラメータとなる値が含まれていることはほとんどない。つまり、対策としては (i) 初期粒子 $\boldsymbol{\theta}_0$ を発生させる分布を適切に選ぶ、もしくは (ii) 初期粒子 $\boldsymbol{\theta}_0$ を発生させる分布が適切でなくても自動的に真のパラメータを探索する仕組みが必要になるわけである。

(i) の方法を選択したのは Yano (2008) である。ここでは Nelder-Mead 法を用いて自己組織化状態空間モデルの尤度関数を最大化することで、パラメータの初期粒子 θ_o^i の分布を探索する手法が提案されている。

(ii) の方法を選択したのは, Kitagawa (1998) である。ここでは上記の問題を解決するためにではパラメータ θ に人工的な誤差項 (イノベーション項) η_t を追加し, 「時変パラメータ」として θ_t を推定する手法を提案されている。

$$\theta_t = \theta_{t-1} + \eta_t. \quad (5.6)$$

これは「パラメータを状態であるかのように推定する」という発想の転換を行ったものであり, パラメータに関する事前情報が極めて少ない場合には有効な手法である。¹⁸⁾

このアイデアをさらに発展させ, Liu and West (2001) では, 補助粒子フィルタを用い, $p(\theta_{t+1}|\theta_t)$ を以下のように設定してパラメータ推定を行うことを提案している。

$$p(\theta_{t+1}|\theta_t) \sim N(a\theta_t + (1-a)\bar{\theta}_t, (1-a^2)V_t), \quad (5.7)$$

ただし, $\bar{\theta}_t$ は θ_t を表す粒子の値の平均, V_t は同じく分散, $a = (3\delta_t - 1)/2\delta_t$ である。さらに δ_t は $(0, 1]$ である (Liu and West (2001) は 0.95 から 0.99 の間の値を用いるように提案している)。Liu and West (2001) の特色は時間 t が増加するにしたがって V_t が減少し 0 に収束する点である。この方式は θ_t がパラメータ θ に近く (なおかつ V_t が十分に大きい) 場合は有効であるが, θ について事前情報が乏しい場合には推定に失敗する可能性がある (Yano (2008) に推定に失敗する場合のシミュレーションが示されている)。なお Liu and West (2001) を拡張し, Carvalho *et al.* (2010) は Parameter Learning という手法を提案しており, この分野ではさらなる発展が続いていくものと考えられる。

6. 実物景気循環モデルへの応用

近年, 「ミクロ経済学的基礎」に基づく動学的マクロ経済学が広く普及してきた。そこで採用される動学的マクロ経済モデルは, 数学的には制約条件付き関数最大化問題として定式化される。本節ではその一つである, 実物景気循環モデルの状態推定を考える。

動学的マクロ経済モデルの基本は Kydland and Prescott (1982) によって開発された「実物景気循環モデル (Real Business Cycle Model, 以下 RBC)」である。RBC モデルは (1) 無限の視野を持つ家計の効用関数最大化, (2) 企業の利潤最大化などのミクロ経済学のフレームワークによって定式化されている。¹⁹⁾

¹⁸⁾ 応用例としてはたとえば Yano (2009), Yano *et al.* (2010), 矢野他 (2011) 等を参照されたい。

¹⁹⁾ RBC を含めた動学的マクロ経済モデルについては英文では McCandless (2008), Benassy (2011), 邦文では加藤 (2006), 江口 (2011) 等が初心者向け解説が充実している。また, 筆者が動学的マクロ経済学とベイズ計量経済学の講義資料をアップロードしている矢野 (2011) も参照されたい。

まず、経済で無限期間の視野を有する家計を想定し、その効用関数 U が以下で与えられているとする。

$$U = E_0 \left[\sum_{t=0}^{\infty} \beta^t \log(C_t) \right], \quad (6.1)$$

ただし、 C_t は消費、 β は主観的割引率であり、 E_0 は条件付き期待を表す。家計の制約条件は

$$C_t + K_{t+1} = A_t K_t^\alpha + (1 - \delta) K_t, \quad (6.2)$$

ただし、 K_t は資本、 A_t は技術レベル、資本の所得分配率 α と資本減耗率 δ はパラメータであり、 K_0 は所与とする。また、技術レベル A_t は以下の 1 階自己回帰で表せると仮定する。

$$\log(A_{t+1}) = \rho_a \log(A_t) + v_t, \quad (6.3)$$

ただし、 ρ_a はパラメータであり、 $v_t \sim N(0, \sigma_d^2)$ である。

ラグランジュの未定乗数法を使ってこの問題の 1 階の条件を求めよう。ラグランジアン (ラグランジュ関数) $\mathcal{L}$ は以下となる。

$$\mathcal{L} = E_0 \left[\sum_{t=0}^{\infty} \beta^t [\log C_t + \lambda_t \{A_t K_t^\alpha + (1 - \delta) K_t - C_t - K_{t+1}\}] \right] \quad (6.4)$$

ただし、 λ_t はラグランジュの未定乗数である。

ラグランジアン $\mathcal{L}$ を C_t , K_t , λ_t に対して微分し、1 階の条件をまとめると、

$$1 = E_0 \left[\beta \{ \alpha A_{t+1} K_{t+1}^{\alpha-1} + (1 - \delta) \} \frac{C_t}{C_{t+1}} \right] \quad (6.5)$$

$$0 = A_t K_t^\alpha + (1 - \delta) K_t - C_t - K_{t+1} \quad (6.6)$$

$$0 = \rho_a \log(A_t) - \log(A_{t+1}) + v_t \quad (6.7)$$

このように 1 階の条件は、3 つの内生変数 C_t, K_t, A_t によって構成される 3 本の非線形の式になる。

RBC の 1 階の条件をまとめると、動学的マクロ経済モデルは一般的に以下のように定式化される。

$$E_t \left[f(\mathbf{X}_{t+1}, \mathbf{Y}_{t+1}, \mathbf{X}_t, \mathbf{Y}_t) \right] = 0, \quad (6.8)$$

ただし、 $\mathbf{X}_t$ は先決変数 (pre-determined variable) のベクトル、 $\mathbf{Y}_t$ は非先決変数 (non-predetermined variable) のベクトルである。

先決変数とは時点 t の値が過去の時点 $t-1$ に依存して決まる変数であり、非先決変数とは時点 t の値が次時点 $t+1$ の値に依存して決まる変数である。本稿の RBC モデルでは時間 t の資本 K_t と技術レベル A_t は時点 $t-1$ に依存して決まる変数であるため、先決変数である。それに対して、消費 C_t は時点 $t+1$ での消費の期待 (予想) C_{t+1} に依存して決まる

表1 RBCのパラメータ.

パラメータ	β	α	δ	ρ_a	σ_d
数値	0.99	0.33	0.0025	0.9	0.0214

性質を有しているため、非先決変数である。そのため、本稿の RBC モデルを式 (6.8) に直すと、 $\mathbf{X}'_t = [K_t, A_t]'$ 、 $\mathbf{Y}_t = [C_t]$ となる。

この一般的な定式 (6.8) は、いくつかの安定条件の下で、以下のように変形できることが Schmitt-Grohe and Uribe (2004) で示されている。

$$\begin{aligned}\mathbf{X}_{t+1} &= h(\mathbf{X}_t, \boldsymbol{\sigma}_d) + \mathbf{v}_{t+1} \\ \mathbf{Y}_{t+1} &= g(\mathbf{X}_{t+1}, \boldsymbol{\sigma}_d),\end{aligned}\tag{6.9}$$

ただし、 $\boldsymbol{\sigma}_d$ は定数である。

さらに Schmitt-Grohe and Uribe (2004), Gomme and Klein (2011) 等は式 (6.9) を定常状態近傍で 2 次近似すると以下となることを示した。

$$\begin{aligned}\mathbf{X}_{t+1} &= \boldsymbol{\sigma}_d^2 \mathbf{K}_x + \mathbf{P} \mathbf{X}_t + \frac{1}{2} (\mathbf{I}_{nx} \otimes \mathbf{X}'_t) \mathbf{G} \mathbf{X}_t + \mathbf{v}_{t+1} \\ \mathbf{Y}_{t+1} &= \boldsymbol{\sigma}_d^2 \mathbf{K}_y + \mathbf{F} \mathbf{X}_{t+1} + \frac{1}{2} (\mathbf{I}_{ny} \otimes \mathbf{X}'_{t+1}) \mathbf{E} \mathbf{X}_{t+1},\end{aligned}\tag{6.10}$$

ただし、 $\mathbf{P}$, $\mathbf{E}$, $\mathbf{F}$, $\mathbf{G}$, $\mathbf{K}_x$, $\mathbf{K}_y$ は行列であり、詳細は Gomme and Klein (2011) に従う。²⁰⁾ 式 (6.10) をシステム方程式 (2.1) とし、さらに $[\mathbf{X}'_t, \mathbf{Y}'_t]$ に観測誤差 ϵ_t を加えて観測方程式 (2.2) とする。なお、パラメータは Gomme and Klein (2011) に従い、表 1 のようにキャリブレーションした。

初期分布 ($C_0 = 0, K_0 = 0, A_0 = 0$)、パラメータを所与として、式 (6.10) を用いて真の状態 $\mathbf{X}_t$ と観測値 $\mathbf{Y}_{1:T}$ をデータ生成し、その生成した観測値に基づき粒子フィルタで消費 C_t 、資本 K_t 、技術レベル A_t を推定した。図 8 に推定結果を示す。この図は粒子フィルタが真の状態をうまく追跡できていることを示している。式 (6.10) は 2 次項を含むため非線形・ガウス型状態空間モデルであり、粒子フィルタの有効性を示す良い適用例である。

7. まとめ

本稿では「粒子フィルタと粒子平滑化」について (1) 粒子フィルタの基礎と応用例, (2) 粒子固定ラグ平滑化, (3) 粒子固定ラグ平滑化へのリサンブル・ムーブ法の適用, (4) 粒子フィルタにおけるパラメータ推定を解説した上で, (5) 動学的マクロ経済モデル (実物景気循環モデル) 推定への応用を解説した。

²⁰⁾ $\mathbf{P}$, $\mathbf{E}$, $\mathbf{F}$, $\mathbf{G}$, $\mathbf{K}_x$, $\mathbf{K}_y$ の計算は Paul Klein のサイトで公表されている Matlab プログラムを使用した。
<http://paulklein.se/newsite/codes/codes.php>

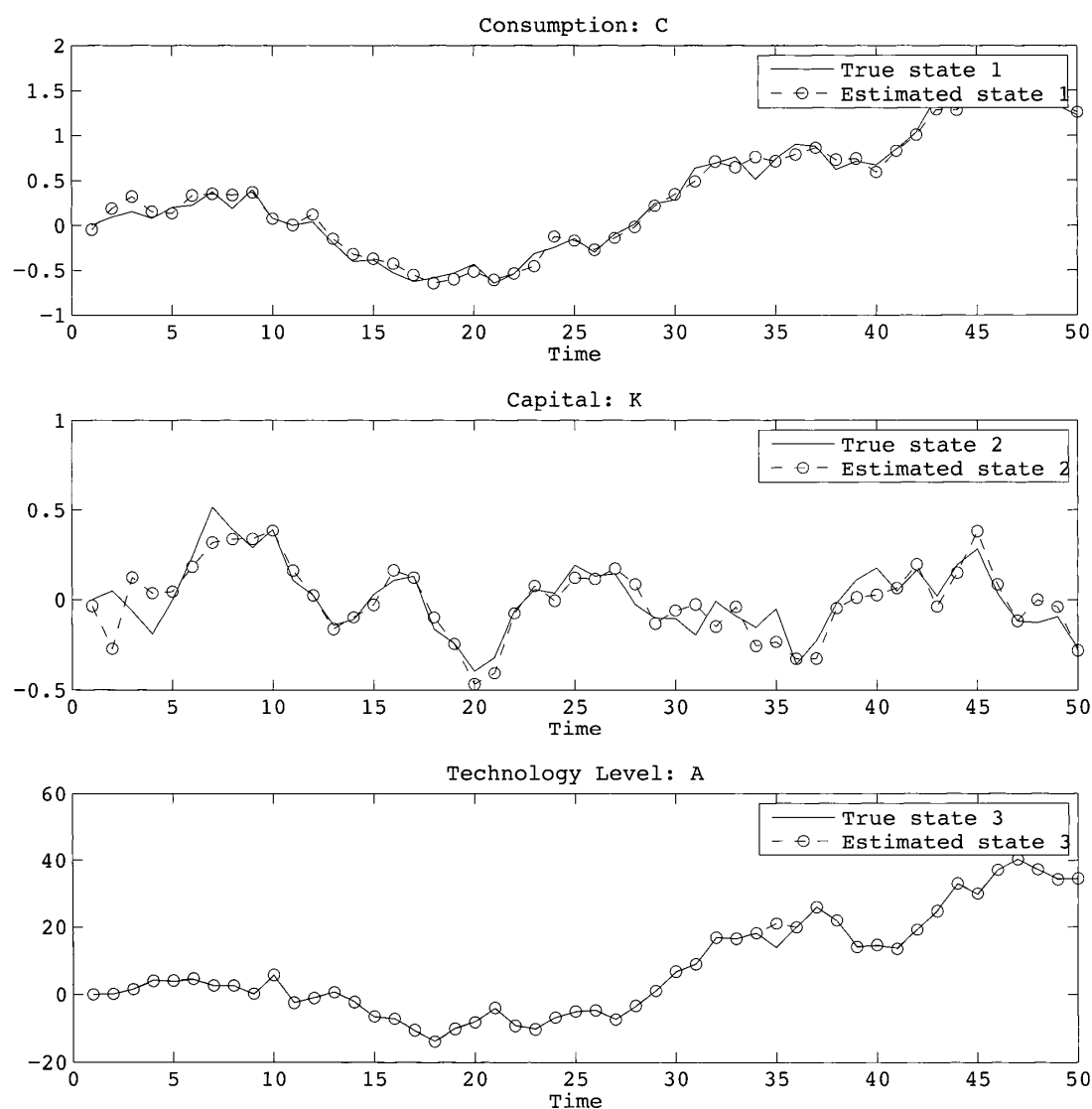


図8 RBCモデルの推定結果.

本稿では取り上げられなかったが、粒子フィルタは Kitagawa (1993, 1996), Gordon *et al.* (1993) 以後, auxiliary particle filter (Pitt and Shephard (1999)), Rao-Blackwellized particle filter (Doucet *et al.* (2000)), merging particle filter (中野他 (2008)) など多くの発展形を生み出した。これらについては粒子フィルタの教科書 (Candy (2009), Prado and West (2010)) やサーベイ論文 (Cappe *et al.* (2007), Doucet and Johansen (2011)) 等を参照されたい。また、本稿で紹介したように粒子フィルタと MCMC をお互いに活かした応用も進んでおり、その方向も今後大いに発展していくものと考えられる。さらにパラメータ推定は本稿で取り上げた自己組織化状態空間モデルや時変係数モデル以外の提案後, Yano (2008) により自己組織化状態空間モデルと Nelder-Mead 法を組み合わせた手法, Carvalho *et al.* (2010) により Particle learning という新しい手法が提案されるなど、今後もさらに発展していくものと思われる。

また、粒子フィルタは純粋な時系列解析の手法に留まらず、工学・気象学・経済・金融等の多くの分野で応用されている。特にコンピュータビジョン (Computer Vision) や気象学における応用は非常に盛んである。CV についてはオープンソースのコンピュータビジョン向けライブラリ OpenCV に粒子フィルタが *condensation* という名称で実装されており、盛んに用いられている。また、気象学においても様々に応用されており、それらの諸研究が樋口 (2011a) にまとめられている。また、経済分野でも応用が進んでおり、既に DeJong and Dave (2011) などの計量経済学の教科書で 1 章分を割いて取り上げられている。粒子フィルタは今後もそれらの諸分野での応用が進み、さらに新しい分野へも適用されていくだろう。

謝辞

本稿は日本経済学会 2013 年度春季大会におけるチュートリアル・セッション「マクロ時系列分析」の発表内容を元に大幅改稿したものである。チュートリアルを組織され、本特集号を企画された谷崎久志教授 (大阪大学) に御礼を申し上げます。また、有益なコメントをくださった日本統計学会誌レフェリー、岡田多恵氏 (大東文化大学)、相馬尚人氏 (早稲田大学) に感謝します。なお、本稿に残る誤りは全て著者に帰するものである。本研究は JSPS 科研費 24530316 の助成を受けたものである。

A 粒子フィルタ・粒子固定ラグ平滑化の導出

A1 粒子フィルタの導出

粒子フィルタ (Particle Filter) は Kitagawa (1996), Gordon *et al.* (1993) によって提案された非線形非ガウス状態空間モデルの状態推定アルゴリズムである。時系列データ $\mathbf{y}_t$, $t = \{1, 2, \dots, T\}$ を所与として非線形非ガウス状態空間モデルは以下のように定式化される。

$$\begin{aligned} \mathbf{x}_t &= f(\mathbf{x}_{t-1}, \boldsymbol{\xi}_s, \mathbf{v}_t), \\ \mathbf{y}_t &= h(\mathbf{x}_t, \boldsymbol{\xi}_m, \boldsymbol{\epsilon}_t), \end{aligned} \tag{A.1}$$

ここで $\mathbf{x}_t$ は $n_x \times 1$ の未知の状態ベクトル、 $\mathbf{v}_t$ は $n_v \times 1$ のシステムノイズベクトル (密度関数は $q(\mathbf{v})^{21)}$ 、 $\boldsymbol{\epsilon}_t$ は $n_\epsilon \times 1$ の観測ノイズベクトル (密度関数は $r(\boldsymbol{\epsilon})$)、 $\boldsymbol{\xi}_s$ は関数 f の $n_s \times 1$ のシステムパラメータベクトル、 $\boldsymbol{\xi}_m$ は関数 h の $n_m \times 1$ の観測パラメータベクトルを表す。関数 f は $f: \mathbf{R}^{n_x} \times \mathbf{R}^{n_v} \rightarrow \mathbf{R}^{n_x}$ 、関数 h は $h: \mathbf{R}^{n_x} \times \mathbf{R}^{n_\epsilon} \rightarrow \mathbf{R}^{n_y}$ の一般には非線形の関数である。以下では式 (A.1) の第 1 式をシステム方程式、第 2 式を観測方程式と

²¹⁾ システムノイズ $\mathbf{v}_t$ は現在ならびに過去の状態ベクトル $\mathbf{x}_t$ とは独立である。

呼ぶ。また、パラメータ θ を以下のように定義する。

$$\theta = \begin{bmatrix} \xi_s \\ \xi_m \end{bmatrix}. \quad (\text{A.2})$$

なお、以下では時点 1 から時点 t までの観測値 $\mathbf{y}_t$ の組み合わせを $\mathbf{y}_{1:t} = \{\mathbf{y}_1, \dots, \mathbf{y}_t\}$ と表す。

観測値から式 (A.1) の状態 $\mathbf{x}_t$ を推定することを状態推定と呼び、パラメータ θ を推定することをパラメータ推定と呼ぶ。状態推定アルゴリズムはベイズの定理をその理論的基盤としていることが広く知られている (たとえば Arulampalam *et al.* (2002) を参照せよ)。ベイズの定理は以下の通りである。

$$\mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:t}) = \frac{\mathbb{P}(\mathbf{y}_t | \mathbf{x}_t) \mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:(t-1)})}{\mathbb{P}(\mathbf{y}_t | \mathbf{y}_{1:(t-1)})}, \quad t \geq 1, \quad (\text{A.3})$$

ただし、 $\mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:(t-1)})$ を事前確率分布 (事前分布)、 $\mathbb{P}(\mathbf{y}_t | \mathbf{x}_t)$ を尤度、 $\mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:t})$ を事後確率分布 (事後分布)、 $\mathbb{P}(\mathbf{y}_t | \mathbf{y}_{1:(t-1)})$ を規格化定数と呼ぶ。観測値がまったくない初期時点における事前確率分布 (以下、初期分布) を $\mathbb{P}(\mathbf{x}_1) = \mathbb{P}(\mathbf{x}_1 | \emptyset)$ と表す。²²⁾ ベイズの定理から明らかであるように、初期分布 $\mathbb{P}(\mathbf{x}_1)$ を適切に選択すれば、繰り返し計算で事後分布 $\mathbb{P}(\mathbf{x}_t | \mathbf{y}_{1:t})$ を推定することができる。

なお、ベイズの定理 (A.3) を確率密度に書き換え、規格化定数を省略すると

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) \propto p(\mathbf{y}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{y}_{1:(t-1)}), \quad t \geq 1, \quad (\text{A.4})$$

となる。

粒子フィルタでは時間 t におけるフィルタ分布を以下のように「粒子」を使って近似する。

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) \simeq \sum_{i=1}^M \frac{w_t^i}{\sum_{i=1}^M w_t^i} \delta(\mathbf{x}_t - \mathbf{x}_{t|t-1}^i), \quad (\text{A.5})$$

ただし、状態ベクトル $\mathbf{x}_{t|t-1}^i$ は粒子、 w_t^i は粒子 $\mathbf{x}_{t|t-1}^i$ の重みを表し、 M は粒子数 (モンテカルロ・シミュレーションの試行数)、 δ はディラックのデルタ関数を表す (重みの導出法は後述)。粒子フィルタの標準的なアルゴリズムでは各時間 t において分布 (A.5) から M 個の粒子のサンプリングを行い、リサンプリング後の粒子 $\hat{\mathbf{x}}_t^i$, $i = \{1, \dots, M\}$ で $p(\mathbf{x}_t | \mathbf{y}_{1:t})$ を以下のように近似する。

$$p(\mathbf{x}_t | \mathbf{y}_{1:t}) \simeq \frac{1}{M} \sum_{i=1}^M \delta(\mathbf{x}_t - \hat{\mathbf{x}}_t^i). \quad (\text{A.6})$$

²²⁾ $\emptyset$ は観測値が存在しないことを示す。

ここまでの基礎知識を元に粒子フィルタのアルゴリズムを導出しよう．まず，予測分布 $p(\mathbf{x}_t|\mathbf{y}_{1:(t-1)})$ は以下の式 (A.7) のように展開できる (チャップマン・コルモゴロフの方程式, Chapman-Kolmogorov Equation という)．

$$p(\mathbf{x}_t|\mathbf{y}_{1:(t-1)}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:(t-1)})d\mathbf{x}_{t-1} \quad (\text{A.7})$$

式 (A.6) を 1 期ずらして t を $t-1$ に入れ替え，式 (A.7) の右辺に代入する．

$$\begin{aligned} p(\mathbf{x}_t|\mathbf{y}_{1:(t-1)}) &= \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:(t-1)})d\mathbf{x}_{t-1} \\ &\simeq \frac{1}{M} \sum_{i=1}^M \int p(\mathbf{x}_t|\mathbf{x}_{t-1})\delta(\mathbf{x}_{t-1} - \hat{\mathbf{x}}_{t-1}^i)d\mathbf{x}_{t-1} \\ &\simeq \frac{1}{M} \sum_{i=1}^M p(\mathbf{x}_t|\hat{\mathbf{x}}_{t-1}^i) \\ &\simeq \frac{1}{M} \sum_{i=1}^M \delta(\mathbf{x}_t - \mathbf{x}_{t|t-1}^i). \end{aligned} \quad (\text{A.8})$$

ただし，粒子 $\mathbf{x}_{t|t-1}^i$ は，時点 $t-1$ の $\hat{\mathbf{x}}_{t-1}^i$ とシステム方程式を用いてモンテカルロ法で発生させる．つまり，

$$\mathbf{x}_{t|t-1}^i \sim p(\mathbf{x}_t|\hat{\mathbf{x}}_{t-1}^i), \quad (\text{A.9})$$

で得られるものである．

式 (A.8) を式 (A.4) に代入すると以下の式が得られる．

$$\begin{aligned} p(\mathbf{x}_t|\mathbf{y}_{1:t}) &\propto p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:(t-1)}) \\ &\propto p(\mathbf{y}_t|\mathbf{x}_t) \times \frac{1}{M} \sum_{i=1}^M \delta(\mathbf{x}_t - \mathbf{x}_{t|t-1}^i) \\ &\simeq \frac{1}{M} \sum_{i=1}^M p(\mathbf{y}_t|\mathbf{x}_{t|t-1}^i)\delta(\mathbf{x}_t - \mathbf{x}_{t|t-1}^i). \end{aligned} \quad (\text{A.10})$$

式 (A.10) の左辺は時間 t におけるフィルタ分布であり，結局，式 (A.10) と式 (A.5) を比較すると，式 (A.5) の重み w_t^i を以下の式で与えることが妥当であると考えられる．

$$\frac{w_t^i}{\sum_{i=1}^M w_t^i} \propto p(\mathbf{y}_t|\mathbf{x}_{t|t-1}^i). \quad (\text{A.11})$$

つまり重み w_t^i は尤度に比例するように生成すればよい．なお，一般的には尤度から生成した重み w_t^i は $\sum_{i=1}^M w_t^i = 1$ を満たさない．しかし，式 (A.5) は確率密度を意味するので， w_t^i を $w_t^i / \sum_{i=1}^M w_t^i$ として規格化して使用する必要があることに注意されたい．

A2 粒子固定ラグ平滑化の導出

続いて粒子固定ラグ平滑化を導出する.

一般に以下の式が成り立つ.

$$\mathbb{P}(\mathbf{x}_{1:t}|\mathbf{y}_{1:t}) = \frac{\mathbb{P}(\mathbf{y}_t|\mathbf{x}_{1:t})\mathbb{P}(\mathbf{x}_{1:t}|\mathbf{y}_{1:(t-1)})}{\mathbb{P}(\mathbf{y}_t|\mathbf{y}_{1:(t-1)})}. \quad (\text{A.12})$$

上式で $\mathbb{P}(\mathbf{y}_t|\mathbf{x}_{1:t}) = \mathbb{P}(\mathbf{y}_t|\mathbf{x}_t)$ と仮定すれば以下の式が得られる.

$$\mathbb{P}(\mathbf{x}_{1:t}|\mathbf{y}_{1:t}) = \frac{\mathbb{P}(\mathbf{y}_t|\mathbf{x}_t)\mathbb{P}(\mathbf{x}_{1:t}|\mathbf{y}_{1:(t-1)})}{\mathbb{P}(\mathbf{y}_t|\mathbf{y}_{1:(t-1)})}. \quad (\text{A.13})$$

式 (A.13) を用いて付録 A1 と同様に展開すれば粒子平滑化が導出できる.

参 考 文 献

- An, S. and Schorfheide, F. (2007). Bayesian analysis of DSGE models, *Econom. Rev.*, **26**(2–4), 113–172.
- Arulampalam, M., Maskell, S., Gordon, N. and Clapp, T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, *IEEE Trans. Signal Process.*, **50**, 174–188.
- Benassy, J.-P. (2011). *Macroeconomic Theory*, Oxford University Press.
- Candy, J. V. (2009). *Bayesian Signal Processing: Classical, Modern and Particle Filtering Methods*, Wiley-Interscience.
- Cappe, O., Godsill, S. J. and Moulines, E. (2007). An Overview of Existing Methods and Recent Advances in Sequential Monte Carlo, *PROCEEDINGS-IEEE*, **95**(5), 899.
- Carvalho, C. M., Johannes, M. S., Lopes, H. F. and Polson, N. G. (2010). Particle learning and smoothing, *Stat. Sci.*, **25**(1), 88–106.
- DeJong, D. N. and Dave, C. (2011). *Structural Macroeconometrics (second edition)*, New Jersey, Princeton University Press.
- Douc, R. (2005). Comparison of resampling schemes for particle filtering, in *In 4th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pp. 64–69.
- Doucet, A. and Johansen, A. M. (2011). A tutorial on particle filtering and smoothing: Fifteen years later.
- Doucet, A., de Freitas, N., Murphy, K. and Russell, S. (2000). Rao-blackwellised particle filtering for dynamic Bayesian networks, in *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, UAI'00, pp. 176–183, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc.
- Doucet, A., de Freitas, N. and Gordon, N., eds. (2001). *Sequential Monte Carlo Methods in Practice*, New York, Springer-Verlag.
- Doucet, A., Briers, M. and Sncal, S. (2006). Efficient block sampling strategies for sequential Monte Carlo, *J. Comput. Graph. Stat.*, 693–711.
- 江口允崇 (2011). 『動学的一般均衡モデルによる財政政策の分析』三菱経済研究所.
- Fernandez-Villaverde, J. and Rubio-Ramirez, J. F. (2007). Estimating macroeconomic models: A likelihood approach, *Rev. Econ. Stud.*, **74**(4), 1059–1087. Forthcoming.
- Gilks, W. R. and Berzuini, C. (2001). Following a moving target-Monte Carlo inference for dynamic Bayesian models, *J. R. Stat. Soc. Ser. B*, **63**(1), 127–146.
- Gomme, P. and Klein, P. (2011). Second-order approximation of dynamic models without the use of tensors, *J. Econ. Dyn. Control.*, **35**(4), 604–615, April.
- Gordon, N., Salmond, D. and Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation, *IEEE Proceedings-F*, **140**, 107–113.
- Higuchi, T. (1995). On the resampling scheme in the filtering procedure of the Kitagawa Monte Carlo filter, ISM Research Memo, No. 556.

- 樋口知之 (1996). 「遺伝的アルゴリズムとモンテカルロフィルタ (特集計算統計学の発展)」『統計数理』**44**(1), 19–30.
- 樋口知之 (2005). 「粒子フィルタ」『電子情報通信学会誌』**88**(12), 989–994, December.
- 樋口知之 (編) (2011a). 『データ同化入門』朝倉書店.
- 樋口知之 (2011b). 『予測にいかす統計モデリングの基本—ベイズ統計入門から応用まで』講談社.
- Hürseler, M. and Künsch, H. R. (2001). Approximating and maximizing the likelihood for a general state-space model, in *Sequential Monte Carlo Methods in Practice* (eds. A. Doucet, N. de Freitas, and N. Gordon), New York, Springer-Verlag, pp. 159–175.
- 伊庭幸人 (2005). 「逐次モンテカルロ法入門」甘利俊一, 竹内啓, 竹村彰通, 伊庭幸人 (編) 『計算統計 II マルコフ連鎖モンテカルロ法の基礎』岩波書店.
- 生駒哲一 (2008). 「逐次モンテカルロ法とパーティクルフィルタ」北川源四郎, 竹村彰通編 (編) 『「21 世紀の統計科学」第 III 巻』東京大学出版会.
- Kalman, R. E. (1960). A new approach to linear filtering and prediction problems, *Trans. ASME—J. Basic Eng. (Ser. D)*, **82**, pp. 35–45.
- 加藤涼 (2006). 『現代マクロ経済学講義—動学的一般均衡モデル入門』東洋経済新報社.
- Kitagawa, G. (1987). Non-Gaussian state space modeling of nonstationary time series, *J. Am. Stat. Assoc.*, **82**, 1032–1041.
- Kitagawa, G. (1993). A Monte Carlo filtering and smoothing method for non-Gaussian nonlinear state space models, Technical report, Institute of Statistical Mathematics Research Memorandum, No. 462.
- Kitagawa, G. (1996). Monte Carlo filter and smoother for non-Gaussian nonlinear state space models, *J. Comput. Graph. Stat.*, **5**(1), 1–25.
- Kitagawa, G. (1998). A self-organizing state-space model, *J. Am. Stat. Assoc.*, **93**(443), 1203–1215.
- 北川源四郎 (2005). 『時系列解析入門』岩波書店.
- Kramer, S. C. and Sorenson, H. W. (1988). Recursive Bayesian estimation using piece-wise constant approximations, *Automatica*, **24**(6), 789–801, November.
- Kydland, F. E. and Prescott, E. C. (1982). Time to build and aggregate fluctuations, *Econometrica*, **50**(6), 1345–1370.
- Liu, J. and West, M. (2001). Combined parameter and state estimation in simulation-based filtering, in *Sequential Monte Carlo Methods in Practice* (eds. A. Doucet, N. de Freitas, and N. Gordon), New York, Springer-Verlag, pp. 197–223.
- Liu, J. S. (2002). *Monte Carlo Strategies in Scientific Computing*, Springer.
- Liu, J. S. and Chen, R. (1998). Sequential Monte Carlo methods for dynamic systems, *J. Am. Stat. Assoc.*, **93**, 1032–1044.
- McCandless, G. (2008). *The ABCs of RBCs: An Introduction to Dynamic Macroeconomic Models*, Harvard University Press.
- 中村和幸, 樋口知之 (2009). 「最近のベイズ理論の進展と応用 [II] : 逐次ベイズとデータ同化」『電子情報通信学会誌』**92**(12), 1062–1067, December.
- 中野慎也, 上野玄太, 中村和幸 (2008). 「Merging Particle Filter とその特性 (特集データマイニングと統計数理)」『統計数理』**56**(2), 225–234.
- 大森裕浩 (2001). 「マルコフ連鎖モンテカルロ法の最近の展開」『日本統計学会誌』**31**(3), 305–344, December.
- 大森裕浩 (2005). 「マルコフ連鎖モンテカルロ法の基礎と統計科学への応用」伊庭幸人 (編) 『計算統計 II—マルコフ連鎖モンテカルロ法とその周辺』岩波書店.
- Pitt, M. K. and Shephard, N. (1999). Filtering via simulation: Auxiliary particle filters, *J. Am. Stat. Assoc.*, **94**(446), 590–599.
- Prado, R. and West, M. (2010). *Time Series: Modeling, Computation, and Inference*, Chapman and Hall.
- Ristic, B., Arulampalam, S. and Gordon, N. (2004). *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House Publishers.
- 佐藤整尚, 高橋明彦 (2005). 「モンテカルロフィルタを用いた金利モデルの推定」甘利俊一, 竹内啓, 竹村彰通, 伊庭幸人 (編) 『計算統計 II マルコフ連鎖モンテカルロ法の基礎』岩波書店.

- Schmitt-Grohe, S. and Uribe, M. (2004). Solving dynamic general equilibrium models using a second-order approximation to the policy function, *J. Econ. Dyn. Control.*, **28**(4), 755–775.
- Tanizaki, H. (1993). *Nonlinear Filters: Estimation and Applications*, Springer.
- Tanizaki, H. (2001). Nonlinear and non-Gaussian state space modeling using sampling techniques, *Ann. Inst. Stat. Math.*, **53**, 63–81.
- 谷崎久志 (2006). 「非線形・非正規状態空間モデルの推定について」『国民経済雑誌』**193**(4), 37–52.
- Whiteley, N. and Johansen, A. M. (2011). Recent developments in auxiliary particle filtering, in *Bayesian Time Series Models* (eds. D. Barber, A. T. Cemgil, and S. Chiappa), Cambridge University Press.
- Yano, K. (2008). A self-organizing state space model and simplex initial distribution search, *Comp. Stat.*, **23**, 197–216.
- Yano, K. (2009). Dynamic stochastic general equilibrium models under a liquidity trap and self-organizing state space modeling, *ESRI Discussion Paper*, Vol. No. 206.
- 矢野浩一 (2011). 「動学的確率的一般均衡理論&ベイズ・マクロ計量経済学 (講義資料配布サイト)」<https://sites.google.com/site/macroeconometrics2011/>.
- 矢野浩一, 佐藤整尚 (2006). 「初期分布探索付き自己組織化状態空間モデルによる金融時系列解析の最前線:t 分布付き確率的ボラティリティ変動モデルへの応用」『FSA リサーチレビュー 2006』143–166 頁, <http://www.fsa.go.jp/frtc/nenpou/2006a/05.pdf>.
- Yano, K., Iida, Y. and Wago, H. (2010). Estimating new Keynesian DSGE models in a liquidity trap using the Monte Carlo particle filter: An application to the Japanese economy, Econometric Society World Congress 2010, <http://www.webmeets.com/ESWC/2010/prog/viewpaper.asp?pid=2576>.
- 矢野浩一, 飯田泰之, 和合肇 (2011). 「流動性制約家計を含むニューケインジアン DSGE モデル: 粒子フィルター法による日本経済の推計」浅子和美, 飯塚信夫, 宮川努 (編)『世界同時不況と景気循環分析』東京大学出版会.
- 和合肇 (編) (2005). 『ベイズ計量経済分析—マルコフ連鎖モンテカルロ法とその応用』東洋経済新報社.
- 渡部敏明 (2000). 『ボラティリティ変動モデル』朝倉書店.