

完璧にサンプリングしよう！ —第二話 天と地の狭間で—

来嶋 秀治, 松井 知己

1. CFTP の落とし穴

前回は、無限のシミュレーションを有限時間で実現する CFTP アルゴリズムを紹介した。今回は CFTP アルゴリズムをより効率よくするためのアイデアについて紹介しよう。

有限の状態空間 Ω 上のエルゴード的なマルコフ連鎖を考える。前回紹介した CFTP アルゴリズムは状態空間 Ω と更新関数 $\phi: \Omega \times [0, 1) \rightarrow \Omega$ で定義されたマルコフ連鎖に対して、過去のある時点における全状態からのシミュレーションを実行し、現時点での coalescence を確認するというものであった。しかし「全状態からのシミュレーションを実行」することは、実務上の多くの場合、状態数が膨大になるため大変困難である。実際に MCMC (Markov chain Monte Carlo) 法がよく用いられているイジングモデルと分割表のランダムサンプリングを例に挙げて説明しよう。

1.1 イジングモデル

イジング (Ising) モデルは、物質の磁性を温度とスピンから説明する、平衡統計物理では非常に基本的で重要なモデルである [4]。また、このモデルは 2 値 (白黒) 画像の表現としてもよく知られる [7]。

頂点数 n のグラフ $G=(V, E)$ 上のイジングモデルを考えよう。イジングモデルでは、各頂点上のスピン $\{-1, +1\}$ が与えられた、状態 $\mathbf{x} \in \{-1, +1\}^V$ について議論する。温度 $T > 0$ において、状態 \mathbf{x} はエネルギー

$$H(\mathbf{x}) \stackrel{\text{def}}{=} -\frac{1}{T} \sum_{\{u, v\} \in E} x(u)x(v) \quad (1)$$

を持つ。ただし、 $x(v) \in \{-1, +1\}$ は状態 $\mathbf{x} \in \{-1, +1\}^V$ における頂点 $v \in V$ のスピン $\{-1, +1\}$ を表す。

状態 \mathbf{x} の存在確率 $\Pr(\mathbf{x})$ はエネルギー関数に対して、

$$\Pr(\mathbf{x}) = \frac{1}{Z(T)} e^{-H(\mathbf{x})} \quad (2)$$

で表わされる。この分布は Gibbs 分布と呼ばれ、温度 T の時のスピン状態の存在確率を表す平衡統計物理の基本概念である。ただし、式(2)中の定数

$$Z(T) \stackrel{\text{def}}{=} \sum_{\mathbf{x} \in \{-1, +1\}^V} e^{-H(\mathbf{x})}$$

は分配関数と呼ばれ、統計物理において非常に重要な値である。MCMC 法は、分配関数の計算法として古くから知られる有効な手段である。

イジングモデル (に対するマルコフ連鎖) の状態空間の大きさは $2^{|V|}$ である。イジングモデルは 2 値画像の表現として用いられると述べたが、例えば、図 1 のように 15×15 という小さな画像の例² に対してさえ、状態空間のサイズは $2^{225} \approx 10^{67}$ となる。

1.2 2元分割表

2元分割表は、与えられた行和と列和を満たし、各要素が非負整数の表である (図 2)。分割表に対する統計学的興味として、行和と列和の相関の検定があげられる。正確検定法に必要な p 値の計算に、マルコ

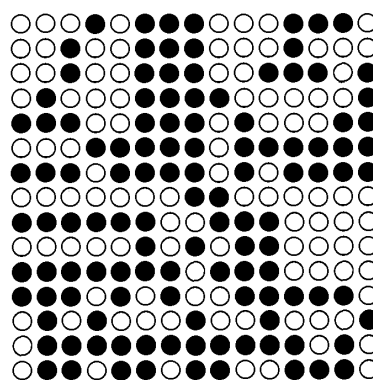


図 1 15×15 格子上的イジングモデルの絵 ($\mathbf{x} \in \{-1, +1\}^V$). \circ は -1 , \bullet は $+1$ を表す

きじま しゅうじ, まつい ともみ
東京大学 大学院情報理工学系研究科
〒 113-8656 文京区本郷 7-3-1

¹ 来: 外部磁場はないのですか? 松: 式が煩雑になるから省略。

² 松: 丸が画像のピクセルで、グラフの頂点だ。来: 各頂点はその上下左右の頂点と隣接しているのが普通ですね。

						12
						18
5	4	3	7	5	6	30

4	1	1	1	3	2	12
1	3	2	6	2	4	18
5	4	3	7	5	6	30

図2 与えられた行和と列和(左), およびそれを満たす分割表(右)の例

フ連鎖を用いて分割表を一様に生成し, モンテカルロ法を用いる手法がよく知られている. しかし, 与えられた周辺和を満たす分割表の個数(すなわち状態空間の大きさ)は一般に膨大となり, さらにそれを数えることは行数を2に固定した場合でさえ困難³であることが知られている.

2. 単調 CFTP

それではいよいよ, 効率的な CFTP アルゴリズムを設計するための「とっても巧妙な更新関数」を導入しよう.

定義 1 更新関数が単調であるとは, 次の二つの条件を満たすことである.

条件 1: 状態空間 Ω 上に半順序関係 “ \geq ” が存在する.

条件 2: 任意の推移は半順序を保存する. すなわち, $x \geq y$ を満たす任意の状態対 $(x, y) \in \Omega^2$ および任意の実数 $\lambda \in [0, 1]$ に対して, $\phi(x, \lambda) \geq \phi(y, \lambda)$ が成り立つ.

単調な更新関数で定義されたマルコフ連鎖のことを, **単調マルコフ連鎖**と呼ぶ.

例 図3は単調マルコフ連鎖の例である. このマルコフ連鎖は状態空間 $\{s_1, s_2, s_3, s_4\}$ を持ち, 半順序(実は全順序)を $s_1 > s_2 > s_3 > s_4$ と定義する. マルコフ連鎖の推移はコインフリップによって行われ, コインがおもて(お)の場合は図3の左の図のように, うら(う)の場合は図3の右の図のように推移する. 図3において, 推移を表す矢印が交差することはなく, これは推移に対して半順序が保存されることを意味する. すなわちこのマルコフ連鎖は単調である.

いま, 半順序集合 (Ω, \geq) に対して, 唯一の最大元 $x_{\max} \in \Omega$ および最小元 $x_{\min} \in \Omega$ が存在するとしよう. すなわち, 任意の x に対して $x_{\max} \geq x \geq x_{\min}$ が成り立

³ 来: #P 完全ですね. 松: そう. 「ナンバーピー」完全だね.

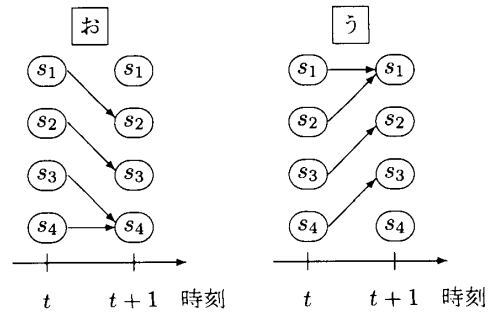


図3 単調マルコフ連鎖

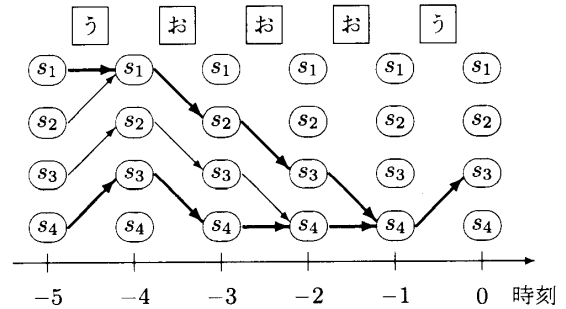


図4 単調マルコフ連鎖の coalesce の様子

つとする. このとき, 次の命題が導かれる.

命題 2 更新関数 ϕ で定義されたマルコフ連鎖は半順序集合 (Ω, \geq) に関して単調であり, $\exists x_{\max}, \exists x_{\min} \in \Omega, \forall x \in \Omega, x_{\max} \geq x \geq x_{\min}$ とする. この時, $T < 0$ および $\lambda \in [0, 1]^T$ に対して coalesce している(すなわち $\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^\lambda(x, \lambda)$) ことの必要十分条件は $\Phi_T^\lambda(x_{\max}, \lambda) = \Phi_T^\lambda(x_{\min}, \lambda)$ である.

例 図4は図3のマルコフ連鎖を用いて時刻-5から時刻0まで推移させた例である. 図4では時刻-5における全状態からのサンプルパスが時刻0で coalesce しているが, このことは, 図の太線, すなわち時刻-5における最大元 s_1 と最小元 s_4 からの2本のサンプルパスを確認するだけでよい.

以上の議論から, 標準的な単調 CFTP アルゴリズムは次のように定義される.

アルゴリズム 1 (単調 CFTP アルゴリズム)

Step 1: シミュレーションの開始時刻を $T := -1$ とする. 空列 $\lambda = ()$ を用意する.

Step 2: 乱数 $\lambda[T], \dots, \lambda[T/2+1]$ を生成し, 数列 λ の先頭に挿入する. すなわち, $\lambda := (\lambda[T], \lambda[T+1], \dots, \lambda[-1])$ とする.

Step 3: 最大元 x_{\max} と最小元 x_{\min} について, 共通の

乱数列 λ を用いて時刻 T から時刻 0 までマルコフ連鎖を推移させる。このとき、

(a) もし、coalesce していれば (すなわち, $\exists y \in \Omega, y = \Phi_T^0(x_{\max}, \lambda) = \Phi_T^0(x_{\min}, \lambda)$), 時刻 0 の状態 y を出力し, 停止する。

(b) そうでなければ, シミュレーションの開始時刻を $T := 2T$ として Step 2 に戻る。

定理 3 エルゴード的な単調マルコフ連鎖に対して, アルゴリズム 1 は確率 1 で有限停止し, アルゴリズム 1 の返す値は定常分布に厳密に従う。

3. 単調マルコフ連鎖の例

定理 3 から, 単調なマルコフ連鎖を設計できたならば, 最大元と最小元の 2 状態からのみシミュレーションを行えば良いので, 状態空間の大きさに依らず coalescence 時間に比例する時間でパーフェクトサンプリングが行える。前の節で非常に簡単な単調マルコフ連鎖の例を示したが, この節ではより実践的な例として, イジングモデルと 2 元 2 行分割表の一樣サンプリングに対する単調マルコフ連鎖を紹介しよう。

3.1 イジングモデル

単純無向グラフ $G=(V, E)$ と温度 T に対するイジングモデルを表現するマルコフ連鎖 \mathcal{M} を次のように定義する。マルコフ連鎖 \mathcal{M} は状態空間 $\{-1, +1\}^V$ を持ち, 現在の状態 $X \in \{-1, +1\}^V$ から次の時刻の状態 X' への推移は次のように定義される。まず, G の頂点 $v \in V$ を一つ, 一樣ランダムに選ぶ。頂点 v に隣接する頂点のうち, 状態 X においてスピンの $+1$ の頂点の個数を $k_+(v, X)$ で表し, -1 の頂点の個数を $k_-(v, X)$ で表す。また,

$$p(v, X) \stackrel{\text{def.}}{=} \frac{e^{\frac{2}{T}k_-(v, X)}}{e^{\frac{2}{T}k_-(v, X)} + e^{\frac{2}{T}k_+(v, X)}} \quad (3)$$

とする。このとき, 確率 $p(v, X)$ で $X'(v) = -1$ とし, 確率 $1 - p(v, X)$ で $X'(v) = +1$ とする。その他の頂点 $w \in V \setminus \{v\}$ のスピンについては $X'(w) = X(w)$ とする。有限マルコフ連鎖 \mathcal{M} は既約で非周期であるので, エルゴード的である。また詳細均衡方程式が成り立つことから, 定常分布は Gibbs 分布(2)となる。このマルコフ連鎖 \mathcal{M} は M-H 鎖 (Metropolis-Hastings chain) と呼ばれる。

上記で定義されたマルコフ連鎖 \mathcal{M} が単調であること

を確認しよう。まず, 状態空間 $\{-1, +1\}^V$ 上に半順序関係 “ \geq ” を導入する。状態 $x, y \in \{-1, +1\}^V$ に対して

$$x \geq y \stackrel{\text{def.}}{\Leftrightarrow} x(v) \geq y(v) \quad (\forall v \in V) \quad (4)$$

とする。ただし, $+1 \geq -1$ である。この半順序集合 $(\{-1, +1\}^V, \geq)$ において, あきらかに最大元および最小元 $x_{\max}, x_{\min} \in \{-1, +1\}^V$ が存在して, それぞれ $x_{\max}(v) = +1 \quad (\forall v \in V), x_{\min}(v) = -1 \quad (\forall v \in V)$ を満たす。

次に, マルコフ連鎖 \mathcal{M} を実現する更新関数 $\phi: \{-1, +1\}^V \times [0, n) \rightarrow \{-1, +1\}^V$ を次のように定義する。いま, V の各頂点には 0 から $n-1$ までの番号が振ってあるとしよう。実数 $\lambda \in [0, n)$ に対して⁴, その整数部分 $[\lambda]$ の番号を持つ頂点を $v \in V$ とする。このとき, 次の時刻の状態 $z = \phi(x, \lambda)$ を, 実数 λ の小数部分 $\lambda - [\lambda]$ を用いて,

$$z(v) = \begin{cases} -1, & (0 \leq \lambda - [\lambda] < p(v, X)), \\ +1 & (p(v, X) \leq \lambda - [\lambda] < 1), \end{cases}$$

$$z(w) = x(w), w \in V \setminus \{v\},$$

と定義すると, ϕ はマルコフ連鎖 \mathcal{M} を実現している。

更新関数 ϕ の単調性を示す。いま $x \geq y$ として, 実数 $\lambda \in [0, n)$ に対応する推移を考える。表記の簡単のため $x' = \phi(x, \lambda)$ および $y' = \phi(y, \lambda)$ と記し, $[\lambda]$ の番号を持つ頂点を v とする。このとき, 更新関数によって v 以外の頂点のスピンは保存されるので, v 以外の任意の頂点 w に対して, $x'(w) \geq y'(w)$ が成り立つ。次に, 頂点 v について $k_-(v, y) \geq k_-(v, x)$ を考慮すると, $p(v, x) \geq p(v, y)$ が成り立つ。このことは更新関数の定義から,

$$x'(v) = -1 \Rightarrow y'(v) = -1,$$

$$y'(v) = +1 \Rightarrow x'(v) = +1$$

が成り立つことを意味する。したがって $x'(v) \geq y'(v)$ となる。

定理 4 更新関数 ϕ は単調である。すなわち, もし $x \geq y$ ならば任意の実数 $\lambda \in [0, n)$ に対して $\phi(x, \lambda) \geq \phi(y, \lambda)$ が成り立つ⁵。

上記より, アルゴリズム 1 では $x_{\max} = \mathbf{1}$ と $x_{\min} = -\mathbf{1}$ の 2 状態からのみシミュレーションすればよい。

⁴ 松: 一樣実数乱数のドメインを $[0, 1)$ から $[0, n)$ に変換したところに注意ね。来: メモしときます。

⁵ 松: 外部磁場があっても単調性の証明は同じだね。来: 式が煩雑になるけどいいんですか? 松: それはイヤ。

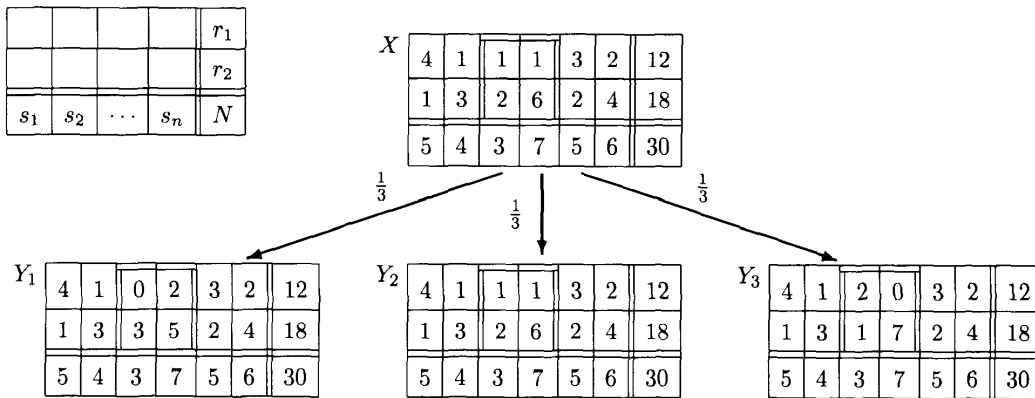


図5 周辺和 (左上), および推移の例

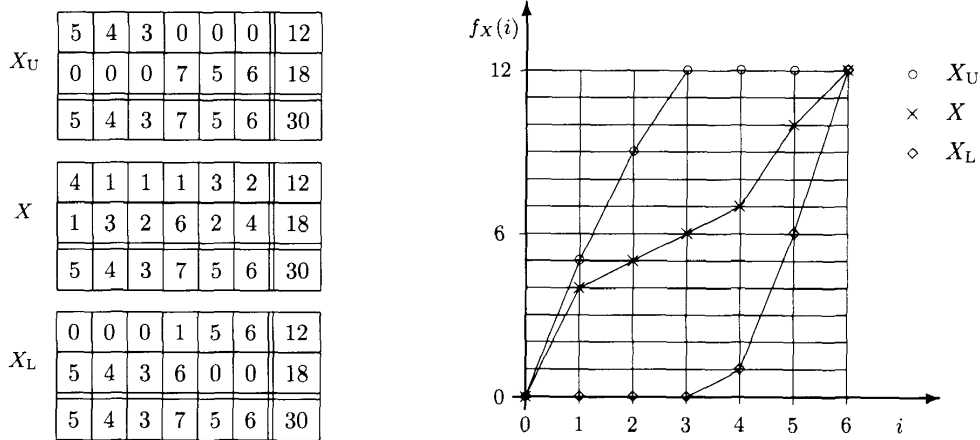


図6 累積和ベクトルの例

3.2 2元2行分割表

行和ベクトル $r=(r_1, r_2)$ および列和ベクトル $s=(s_1, \dots, s_n)$ を満たす $2 \times n$ 分割表の集合を Ξ とする。ただし, $r_1+r_2=s_1+\dots+s_n=N$ である (図5)。以降, Ξ 上の一様分布を定常分布を持つマルコフ連鎖 M を定義する。

分割表 $X \in \Xi$ の i 行 j 列の値を $X[i, j]$ で表す。状態空間 Ξ を持つマルコフ連鎖 M の推移は次のように定義される。まず隣り合う2列 ($j, j+1$) ($j \in \{1, \dots, n-1\}$) を一様選ぶ。次に, j 列目および $j+1$ 列目に

$+k$	$-k$
$-k$	$+k$

(5)

の形の部分表を加えたとき各要素が非負となる表に一樣に推移する。このとき, 整数 k は負の値も取り得ることに注意されたい。図5は推移の例である。状態 X に対して (3, 4) 列が選ばれたとき, 非負性を満たす表は Y_1, Y_2, Y_3 の三つであり, それぞれに確率 $1/3$ で推移する。マルコフ連鎖 M は有限で既約で非周期的

なのでエルゴード的である。また, 詳細均衡方程式を満たすので, マルコフ連鎖の定常分布は一様分布である。

状態空間 Ξ 上に半順序関係を導入しよう。任意の $X \in \Xi$ に対して, 累積和ベクトル $f_X \stackrel{\text{def}}{=} (f_X(0), f_X(1), \dots, f_X(n))$ を

$$f_X(i) \stackrel{\text{def}}{=} \begin{cases} 0 & (i=0), \\ X[1, 1] + \dots + X[1, i] & (i \in \{1, \dots, n\}), \end{cases}$$

と定義する。この f_X は X の一意表現になっている。任意の対 $X, Y \in \Xi$ に対して, $X \geq Y$ の必要十分条件を $f_X - f_Y \geq 0$ とする。2項関係 “ \geq ” が Ξ 上の半順序であることは明らかである。図6の折れ線は累積和ベクトル (の線形汎化関数) である。半順序関係の大小関係は折れ線表現での上下関係に対応する。

この半順序集合における最大元と最小元について考える。二つの特別な分割表 $X_U, X_L \in \Xi$ を北西隅の規則および北東隅の規則で得られる表とする。与えられた半順序集合 (Ξ, \geq) において, 任意の $X \in \Xi$ に対して, $X_U \geq X \geq X_L$ が成り立つ。このことは, 図6にお

いてすべての分割表 $X \in \Xi$ が X_U と X_L の間に挟まれることを表す。

このように半順序を定義すると、マルコフ連鎖 \mathcal{M} に対して単調な更新関数を定義できる。詳細は文献[1]を参照いただきたい。

4. coalescence 時間と mixing 時間

以上の議論から、coalescence 時間の短い単調マルコフ連鎖を設計することで、計算効率の良いパーフェクトサンプリング法が得られることが分かる。一方、通常 MCMC 法で用いられている近似サンプリングでも、マルコフ連鎖の収束の速さは計算効率の重要な鍵となる。実は coalescence 時間とマルコフ連鎖の収束の速さの間には密接な関係がある。次では、この関係について議論しよう。

以降、有限の状態空間 Ω を持ち、推移確率行列 P および定常分布 π を持つエルゴード的なマルコフ連鎖 \mathcal{M} を考える。準備としてマルコフ連鎖の収束スピードを評価するための尺度を導入しよう。(同一の)有限状態空間 Ω 上の二つの確率分布 ν_1 と ν_2 が与えられた時、 ν_1 と ν_2 の間の総分布距離は

$$d_{TV}(\nu_1, \nu_2) \stackrel{\text{def}}{=} \frac{1}{2} \sum_{x \in \Omega} |\nu_1(x) - \nu_2(x)|$$

と定義される。マルコフ連鎖 \mathcal{M} の mixing 時間⁶ は、任意の正数 $\varepsilon < 1$ に対して

$$\tau(\varepsilon) \stackrel{\text{def}}{=} \max_{x \in \Omega} \{ \min\{t \mid \forall s \geq t, d_{TV}(\pi, P_x^s) \leq \varepsilon\} \}$$

と定義される⁷。ただし、 P_x^s は時刻 0 での状態 $x \in \Omega$ とした際の時刻 s での状態の確率分布とする。mixing 時間はマルコフ連鎖に対して固有の定数であり、更新関数には依存しないことに注意されたい。

coalescence 時間は更新関数に依存する確率変数である。関数 ϕ はマルコフ連鎖 \mathcal{M} を実現する更新関数であるとし、確率変数 T_* を更新関数 ϕ に対する coalescence 時間とする。

定理 5 マルコフ連鎖 \mathcal{M} を実現する更新関数 ϕ の coalescence 時間の期待値 $E[T_*]$ に対して、マルコフ連鎖 \mathcal{M} の mixing 時間 $\tau(\varepsilon)$ は

$$0 < \forall \varepsilon < 1, \tau(\varepsilon) \leq E[T_*] \varepsilon^{-1}$$

を満たす。

この定理はマルコフ連鎖の mixing 時間の算定に用いられる coupling 法を利用して示すことができる。定理 5 はマルコフ連鎖 \mathcal{M} に対する任意の更新関数 (における coalescence 時間) について成り立つことに注意が必要である。

定理 5 より、マルコフ連鎖に対して適当な更新関数を設計して、その coalescence 時間を知ることができればマルコフ連鎖の mixing 時間すなわち収束の速さの上限を押しえることができる。例えば単調な更新関数が設計できれば、coalescence 時間の期待値を計算機で推定することはそれほど難しくない。

さらに次ではマルコフ連鎖 \mathcal{M} に対する更新関数 ϕ は半順序 \succeq に対して単調であり、また半順序集合 (Ω, \succeq) は最大元 x_{\max} と最小元 x_{\min} を持つ場合を考えよう。ここで、 x_{\max} と x_{\min} を結ぶ (\succeq に関する) 極大鎖の長さを D とする。このとき、定理 5 とは逆に coalescence 時間の期待値が mixing 時間によって上から押しえられる。

定理 6 更新関数 ϕ を用いたアルゴリズム 1 の coalescence 時間 T_* は

$$E[T_*] \leq 2 \ln(2e(D-1)) \tau\left(\frac{1}{e}\right)$$

を満たす。

すなわち、マルコフ連鎖の収束がある程度の精度 ($\varepsilon \leq 1/e$) で要求される場面において、もし単調な更新関数が設計できれば、定理 5 と定理 6 から mixing 時間の上下限が coalescence 時間の期待値によって

$$\frac{1}{2 \ln(2e(D-1))} E[T_*] \leq \tau\left(\frac{1}{e}\right) \leq \tau(\varepsilon) \leq E[T_*] \varepsilon^{-1}$$

のように押しえられるのである。

5. まとめ

連載第 2 回はパーフェクトサンプリングを実現する「単調 CFTP」を紹介した。これを用いると、単調で収束の速い更新関数を設計できれば、coalescence 時間に比例する時間で完璧なサンプリングができる。また節 4 の議論から、この時間は信頼におけるサンプリングを行うために最低限必要な時間と同程度であることも分かった。

今回紹介したもの以外の単調マルコフ連鎖として、タイリングのランダム生成に対するものがある。Propp のホームページ中にある文献[7]では、この単

⁶ 来：訳は攪拌時間でしょうか？ 松：読めるけど書けん。

⁷ 松：つまり mixing 時間は「初期状態に寄らず、総分布距離が ε 以下になるために必要な時間」っていう定義だね。

調 CFTP アルゴリズムの動く様子が見られる。しかし単調マルコフ連鎖の設計は容易ではなく、あまり多くは知られていない。

節 2 で紹介したオーソドックスな単調 CFTP 以外にも CFTP アルゴリズムが効率的に実行できるものが提案されている。以降、二つの例を簡単に紹介しよう。

5.1 単調 CFTP 以外の例

Propp and Wilson は、連結無向グラフ上の (根付き) 全張木のランダムサンプリングに対して、多項式時間のパーフェクトサンプリング法を提案している [2]。この CFTP アルゴリズムでは、マルコフ連鎖の「最新の推移」が現在の状態に保存されるという性質を用いて、過去からのシミュレーションをすることなく現在の状態を一意に特定する。実はこの問題は、グラフ上の全張木の数え上げを利用した多項式時間の厳密な一様サンプリング法が知られているが、これに比べて CFTP アルゴリズムはおどろくほどシンプルである。

小沢, 高橋, 高橋は待ち行列ネットワークに対して定常分布からのパーフェクトサンプリング法を提案している [5]。彼らのアルゴリズムは広い意味で単調 CFTP アルゴリズムであるが、

- ・マルコフ連鎖が非エルゴード的である,
- ・最大元からのシミュレーションのみで coalesce を確認できる,

という二つの大きな特徴を持つ。

5.2 次回予告

単調 CFTP を導入することによって、アルゴリズム

は効率的になった。しかし、いざ単調 CFTP アルゴリズムを実装してみようとする、「乱数の列を保存する領域」の確保が困難であることが分かる。次回はこの点を克服する「絶妙のアルゴリズム」を紹介しよう。

(「第三話 終わりある未来」に続く)

参考文献

- [1] S. Kijima and T. Matsui: "Polynomial-time perfect sampling algorithm for two-rowed contingency tables," METR 2003-15, (2003).
- [2] J. Propp and D. Wilson: "Exact sampling with coupled Markov chains and applications to statistical mechanics," Random Structures and Algorithms, 9 (1996), 232--252.
- [3] J. Propp and D. Wilson: "How to get a perfect random sample from a generic Markov chain and generate a random spanning tree of directed graph," Journal of Algorithms, 27 (1998), 170-217.
- [4] 伊庭幸人: "統計学者・数理工学者のための統計物理入門一改訂版 (3.02 版) —格子スピン模型とマルコフ連鎖モンテカルロ法を中心に—," ISM Research Memorandum, No. 635, February, 1997.
- [5] 小沢利久, 高橋成晃, 高橋幸雄: "単調なマルコフ連鎖とネットワークモデルへの応用," 日本オペレーションズリサーチ学会 2004 年春季研究発表会アブストラクト集 (2004), 104-105.
- [6] 間瀬茂, 武田純: "空間データモデリング—空間統計学の応用," データサイエンス・シリーズ 7, 共立出版, 2001.
- [7] <http://www.math.wisc.edu/propp/tiling/www/applets/>