マルコフ連鎖モンテカルロ法における近似精度保証と完璧サンプリング法

来嶋 秀治

論文要旨

本論文では $n-1$ 次元単体中の整数格子点上に定義された対数分離凹分布からのランダムサンプリング法とその応用について議論する．具体的には，マルコフ連鎖モンテカルロ (MCMC) 法に基づく近似サンプリング法，完璧サンプリング法，および正規化定数の計算法に対する効率的アルゴリズムの設計と性能解析を行う．さらに，得られた計算法を実際の問題に適用する．

MCMC 法は，サンプリングにマルコフ連鎖を利用するモンテカルロ法であり，数値積分，シミュレーションなどに用いられる．大規模な空間を持つ対象に対して効果的な計算法であり，特にランダムサンプリング自体が困難な問題に対して強力に効果を発揮する．MCMC 法を実際に用いる上で，大きな問題となるのがマルコフ連鎖の収束の速さである．すなわち，「マルコフ連鎖を何回推移させれば，定常分布に十分近くなるのか」という問題である．この問題が本論文の主題のひとつである．

本論文で扱う対数分離凹分布は連続空間中の対数凹分布に素朴に対応する離散分布のひとつである．しかし，数多く存在する連続空間中の対数凹分布に対する算定法を離散的な対数凹分布に直接適用することは困難である．その理由として，連続分布の算定に用いられるスケーリングの技法が離散分布に対しては一般に使えないこと，マルコフ連鎖の推移の方向の自由度が減ることの 2 点が挙げられる．この点を踏まえ，連続空間を対象とする先行研究のほとんどでは収束の速さをコンダクタンス法に基づいて算定しているのに対し，本研究ではカップリング法に基づく算定を行う．

本論文では $n-1$ 次元単体中の整数格子点上に定義された対数分離凹分布に対して，2 つの新しい “hit-and-run” 型のマルコフ連鎖を提案する．1 つは効率的な近似サンプリング法を目的とするマルコフ連鎖である．このマルコフ連鎖が $\mathrm{O}(n^2 \ln(K\varepsilon^{-1}))$ 時間で収束することを経路カップリング法を用いて示す．ただし K は単体の一辺の長さである．この証明において交互不等式のアイデアを導入し，対数分離凹分布が交互不等式を満たすことを利用する．もう 1 つは効率的な完璧サンプリング法を目的とするマルコフ連鎖である．このマルコフ連鎖が「単調」であることを示し，単調 CFTP アルゴリズムに基づく完璧サンプリング法を実現する．単調性の証明においても交互不等式が証明の鍵となる．さらにこの単調マルコフ連鎖が高速に収束するための条件を提示し，条件を満たす場合に $\mathrm{O}(n^3 \ln(Kn))$ 時間で収束することを示す．証明においては，新たに提案する特殊な距離を導入した経路カップリング法を用いる．

本論文のもうひとつの主題は MCMC 法の解の近似精度保証と計算時間の議論である．本論文では，$n-1$ 次元単体中の整数格子点上に定義された対数分離凹関数の積分法について，MCMC 法に基づく乱択近似計算法を与える．提案する計算法は自己帰着性を利用した再帰的なモンテカルロ法で，上述の近似サンプリング法および完璧サンプリングを利用する．提案する乱択近似計算法に関して，以下の 2 つの技術的な注意点を挙げる．ひとつは従来研究の多くが 2 分法に基づく再帰を行っているのに対し，本研究では n 分法に基づく再帰を採用する点である．この変更により，近似解の精度保証において注意深い議論が必要となる．もうひとつは得られる近似解の期待値の真の値からの偏りについて議論を行う点である．従来研究では，近似解の期待値の真の値からの偏りについての議論が不十分であり，計算機実験によって大きな偏りが生じる場合があることも確認される．本研究ではこの偏りを小さくするための方法を提案し，得られる近似解の期待値の真の値からの偏りを理論的に算定すると共に，実験的に偏りが小さくなることを示す．この議論により，しばしば実用の場面において現れる，精度の保証された近似解を得るのに必要な計算時間が確保できない場合にも，質の良い近似解を与えることが可能となる．

本論文では $n-1$ 次元単体中の整数格子点上に定義された対数分離凹分布の例として，待ち行列ネットワークの基本的で重要なモデルのひとつである閉ジャクソンネットワークの積形式解，統計学における基本的な多変量統計データである 2 行分割表の一様分布および多項超幾何分布，生物情報学の多くの統計的手法において多項分布の共役分布としてしばしば現れる (離散化) Dirichlet 分布を取り扱う．これらの問題に対して，効率的な近似サンプリング法，完璧サンプリング法および乱択近似計算法を与える．

目次

1 はじめに 1

1.1 研究の背景 1

1.2 本研究の概要と関連する先行研究 4

1.3 本論文の構成 9

2 準備 11

2.1 定常分布の設計 11

2.2 マルコフ連鎖の収束時間の算定 11

2.3 完璧サンプリング法 14

2.4 乱択近似計算法と精度保証 28

3 対数分離凹分布からのサンプリング 33

3.1 対数分離凹分布 33

3.2 近似サンプリング法 33

3.3 完璧サンプリング法 38

4 モンテカルロ積分と近似精度保証 45

4.1 再帰構造とモンテカルロ法 45

4.2 完璧サンプリング法を用いた場合 47

4.3 近似サンプリング法を用いた場合 49

4.4 最大領域の判定が容易な場合 53

5 応用 57

5.1 待ち行列ネットワーク 57

5.2 分割表 60

5.3 離散化 Dirichlet 分布に従うランダムサンプリング法 67

6 まとめ 81

謝辞 85

参考文献 87

1　はじめに

1.1　研究の背景

マルコフ連鎖モンテカルロ (*MCMC*: *Markov chain Monte Carlo*) 法は，サンプリングにマルコフ連鎖を利用するモンテカルロ法であり，数値積分，シミュレーションなどに用いられる．大規模な空間を持つ対象に対して効果的な計算法であり，特にランダムサンプリング自体が困難な問題に対して強力に効果を発揮する．MCMC 法という言葉は，狭義にはサンプリングにマルコフ連鎖を利用するモンテカルロ法を指す．しかし近年では，広義に「マルコフ連鎖を用いたサンプリング法」を利用した確率的アルゴリズム全般を指す事も多い．この中にはサンプリング法そのものも含まれる．本論文では，断りの無い場合，MCMC 法を広義の意味で用いるものとする．

MCMC 法におけるサンプリングの基本的なアイデアは，エルゴード的なマルコフ連鎖を繰り返し推移させ，漸近的に定常分布に従うサンプルを得るという非常に単純なものである．したがって，適切なマルコフ連鎖を設計することで，所望の分布に対するサンプリング法を構築することができ，各種の確率的アルゴリズムに組み込むことができる．この枠組みの簡明さゆえに，MCMC 法は，統計物理学，経済学，統計学，バイオインフォマティクス，オペレーションズリサーチなどの様々な分野に頻繁に現れる．

MCMC 法が広く用いられ，効果的である要因として，以下の 2 つの利点が挙げられる．ひとつは，対象集合が膨大で集合全体を取り扱うことが困難な集合に対して，取扱いの容易な局所的操作でアルゴリズムが構成される点である．もうひとつは，所望の定常分布を設計する容易な方法が存在する点である．

後者に関しては，マルコフ連鎖の各状態の生起確率と，状態間の推移確率に関する局所的な均衡式に着目することで，所望の定常分布をもつマルコフ連鎖を設計することができる．古くから良く知られる M-H (Metropolis-Hastings) 法，Gibbs サンプラー，熱浴 (heat bath) 法をはじめとするほとんどすべての MCMC 法が，この局所的な均衡式を満足するという概念に基づくものである．

一方，前者に関しては大きな問題が存在する．MCMC 法が機能するためには，局所操作が既約性を満足する，すなわち「局所操作により，すべての状態間が到達可能である」ことが必要不可欠である．これが MCMC 法を設計する上で重要となる，「マルコフ基底を構成する問題」である．マルコフ基底の構成は一般には簡単ではなく，計算量的には別解問題 (ASP) の NP 困難性と言う問題を内包している．具体的な例を挙げると，ある 3 元分割表が与えられた時，その分割表の 2 次元周辺和と同一の周辺和を満たす別の 3 元分割表を見つける問題は NP 完全であり [46]，高次元分割表のマルコフ基底を見つけることは容易ではない．サイズを制限した高次元分割表に対しては，Aoki と Takemura が Gröbner 基底を利用して極小マルコフ基底を導出することに成功している [5]．この他にも，多くの組合せ的対象に対しては，組合せ最適化の研究者を中心に局所操作について議論されている．例えば列挙算法における逆探索法 [7] は，列挙する対象からなる状態空間

を連結にする技法と見ることもでき，既約なマルコフ基底の設計に一役買っている．したがって，対象が組合せ的な構造を持つ場合には既約なマルコフ基底を構成する局所操作が知られていることも多い．

MCMC 法を実際に用いる場面において，大きな問題となるのがマルコフ連鎖の収束の速さである．すなわち，「マルコフ連鎖を何回推移させれば，定常分布に十分近くなるのか」という問題である．この問題が本論文の主題のひとつである．1980 年代からマルコフ連鎖の混交時間 (*mixing time*) の算定に関する研究が盛んに行われている．混交時間とは，マルコフ連鎖の初期状態に依存した分布と定常分布との誤差を十分小さいものとみなすために必要な時間であり，ならし時間 (burn-in time) とも呼ばれる．マルコフ連鎖の混交時間は推移確率行列の第 2 固有値の大きさによって決まる．したがって，推移確率行列が陽に与えられたマルコフ連鎖に対しては，推移確率行列の第 2 固有値を計算することで混交時間を算定することが可能である [101]．しかし，通常 MCMC 法を適用したい対象は状態空間が非常に大きく，推移確率行列を陽に持つことさえ困難なため，推移確率行列の第 2 固有値を直接計算する手法は今のところ現実的なものとはなっていない．マルコフ連鎖の混交時間の上界を算定するための手法として，カップリング (*coupling*) 法，コンダクタンス (*conductance*) 法などの算定手法が提案され，多くのマルコフ連鎖の解析に用いられている．以下，これらについて簡単にまとめる．

カップリング法は，エルゴード的なマルコフ連鎖の状態分布が唯一の定常分布に収束することを示す道具として用いられる手法である．これに対し Aldous は 1983 年に，カップリング法を利用して混交時間を算定する手法を提案した [3]．カップリング法は，異なる初期状態をもつ複数のマルコフ連鎖が同一の状態に到達するまでの時間 (カップリング時間: coupling time) をもって混交時間を見積もる手法である．直感的には，カップリング時間はマルコフ連鎖が初期状態に依存しない状態に到達するまでの時間とみなされる．1997 年，Bubley と Dyer はカップリング法を改良した経路カップリング (*path coupling*) 法を提案し，さらに使いやすい物にしている [13, 12]．カップリング法では，全状態対に関してカップリング時間を算定する必要があるのに対し，経路カップリング法は，状態空間上に (疎な) グラフ構造を人工的に導入し，枝に対応する状態対のカップリング時間の算定のみで，全状態対のカップリング時間を算定できるようにしたものである．その他，状態対の “距離” によってカップリング法を適応的に変化させる多段階 (multiphase) カップリングなどの算定手法も提案されている．カップリング法を用いて混交時間が算定された例としては，Ising モデル [19]，グラフの彩色 [13]，2 行分割表 [27]，単体上の一様分布 [96] があげられる．

コンダクタンス法は 1989 年に Sinclair と Jerrum が提案した手法である [102, 100]．この手法は，マルコフ連鎖の推移グラフ上で確率の “流れ” を考え，コンダクタンス (流れにくさの量) を計算することで，混交時間を見積もるものである．Sinclair と Jerrum は推移確率行列の第 2 固有値の大きさがコンダクタンスによって大まかに見積もれることを示している．しかし，推移グラフのコンダクタンスの計算は一般には簡単ではない．そこで，Sinclair と Jerrum はコンダクタンスを見積もるための基準経路 (*canonical path*) 法を同時に提案している [102, 100]．これは推移グラフ上の全頂点間に基準経路を構成し，経路の集積する “ボトルネック” を見つけることで，コン

ダクタンスを算定するものである．この他にも局所的なコンダクタンスの算定により混交時間を算定する手法などが提案されている．コンダクタンス法を用いて混交時間の算定された例としては，2 部グラフの完全マッチング [102, 51]，凸体上の一様サンプリング [26]，0-1 ナップサック多面体 [81] が挙げられる．

カップリング法，コンダクタンス法とも，マルコフ連鎖の構造に注目した手続き的手法と言える．一般的にコンダクタンス法とカップリング法を比較すると，コンダクタンス法は多くのマルコフ連鎖に対して計算が行いやすい反面算定が緩いことが多いのに対して，カップリング法は状態対の推移の構成は困難だが，算定できたなら真の混交時間に近い値が得られることが経験的に知られている [40]．

カップリング法，コンダクタンス法の他には，Sobolev 不等式を用いて推移確率行列の第 2 固有値を直接算定する手法がある [84]．この手法により，平衡マトロイド (balanced matroid) の基 [29] や凸体上の対数凹分布 [34] に対するマルコフ連鎖の混交時間の算定が行われている．また，カップリングを実際にシミュレーションし，収束時間を実験的に算定しようと言う研究もある．あるいは，マルコフ連鎖をあらかじめ設計して収束時間を算定する手法とは異なる文脈で，Boyd, Diaconis, Xiano は状態空間と推移グラフの与えられたマルコフ連鎖に対して半正定値計画(SDP) を解き，第 2 固有値の大きさを最小にする推移確率行列の設計について議論している [11]．現在でも未解決問題の決着に向け，多くの研究者が高速に収束するマルコフ連鎖の設計および混交時間の算定に取り組んでいる．

通常の MCMC 法は，マルコフ連鎖の収束の速さを見積もり，定常分布との誤差を保証する「近似サンプリング法 (*approximate sampler*)」である．近似サンプリング法とは，所望の分布との誤差の大きさが保証されたサンプリング法のことである．これに対し，所望の分布との誤差が全く無いサンプリング法，すなわち，所望の分布に厳密に従うサンプリング法を「完璧サンプリング法 (*perfect sampler*)」と呼ぶ．1996 年に Propp と Wilson はマルコフ連鎖を用いて完璧サンプリング法を実現する過去からのカップリング (*CFTP*: *coupling from the past*) アルゴリズムを提案し [94]，多くの研究者に衝撃を与えた．CFTP アルゴリズムは，所与のマルコフ連鎖のシミュレーション法自体を工夫することで完璧サンプリング法を実現し，完璧シミュレーション (perfect simulation) とも呼ばれる．マルコフ連鎖を用いて完璧サンプリング法を実現するアルゴリズムに関してはその後も盛んに研究が行われ，CFTP アルゴリズムの他に，2000 年には Wilson 自身によって CFTP アルゴリズムを改良した read once アルゴリズムが [106]，また，1998 年に Fill によって中断可能 (interruptible) アルゴリズム [30, 31] が提案されている．

これらのアルゴリズムを効率的に実現するにはマルコフ連鎖が「単調性 (*monotonicity*)」という “良い” 性質を持つ必要がある．しかし，マルコフ連鎖の単調性の条件が厳しいため，これまでに実用的な問題に対して単調なマルコフ連鎖が実現できた例は非常に少ない．単調 CFTP アルゴリズムの適用例として，Ising モデル [94] とタイリング [107] が挙げられる．このほか，単調性を利用しない効率的な CFTP アルゴリズムとして，全張木のランダムサンプリング法がある [95]．

狭義の MCMC 法の解の近似精度保証と計算時間の算定において，モンテカルロ法の設計とそ

の精度保証の議論を欠くことはできない．この問題が本論文のもうひとつの主題である．狭義の MCMC 法の用いられる多くの対象は，積分，若しくは数え上げ (*counting*) の中でも比較的困難な問題であることが多い．1979 年に Valiant は数え上げが困難な問題の計算量クラスとして，#P 完全 (#P 困難) の概念を提案した [103]．#P 完全は NP 困難の中でも難しいとされる部分クラスである．数え上げ問題に対する乱択近似解法の計算量的なクラスとして，1983 年に，Karp と Luby により全多項式時間乱択近似スキーム(*FPRAS*: *fully polynomial-time randomized approximation scheme*) の概念が提案された [55, 56]．FPRAS は誤差 ε 以内の近似解を高い確率で出力する多項式時間の乱択計算法である．論文 [55] では，#P 完全問題である DNF 数え上げ問題に対する FPRAS が提案されている．

FPRAS の設計に関する研究の中心的な話題として，高次元空間中の凸体の体積計算法が挙げられる．n 次元空間中の凸体の体積計算は#P 困難問題として知られている．1991 年に Dyer, Frieze, Kannan によって n 次元中の凸体の体積計算に対する初の FPRAS が提案された [26]．しかし，Dyer, Frieze, Kannan の結果は $\mathrm{O}^*(n^{23})$ と言う計算時間であり，その後，多くの研究者により改良が施された．ただし，ソフトオー (soft-O) 表記 O^* は，誤差および $\log n$ の多項式係数を省略したオーダー表記法である [54]．現在では Lovász と Vempara による $\mathrm{O}^*(n^4)$ の FPRAS が得られるに至っている [74]．

対象の空間が再帰的な組合せ構造を持つ場合に対しては，1989 年に Sinclair と Jerrum が自己帰着性を利用した再帰的モンテカルロ法を提案している．彼らはこの手法を用いて，#P 完全問題であるパーマネントの計算に関して，入力が密な行列の場合に FPRAS を設計している [102]．一般の $n \times n$ 行列のパーマネントの計算に対しては ,Jerrum, Sinclair, Vigoda が 2001 年に ,アニーリング法を利用した MCMC 法に基づいて，この問題に対する初の FPRAS を与えている [51]．MCMC 法に基づく乱択近似スキームの議論を行う上で特に注意を要するのは，多くの場合において，モンテカルロ法が近似サンプリング法を用いている点である．このため (狭義) MCMC 法の近似精度保証には，近似サンプリング法による分布の誤差に注意した議論が必要不可欠となる．

1.2 本研究の概要と関連する先行研究

本論文では $n-1$ 次元単体中の整数格子点上に定義された対数分離凹 (*logarithmic separable concave*) 分布からのランダムサンプリング法とその応用について議論する．具体的には，マルコフ連鎖モンテカルロ法に基づく近似サンプリング法，完璧サンプリング法，および正規化定数の計算法に対する効率的アルゴリズムの設計と性能解析を行う．また，得られた計算法を実際の問題に適用する．

対数凹関数とは関数値の対数に着目すると凹性をもつ (すなわち「上に凸」の) 関数である．一様分布，二項分布，幾何分布，ポアソン分布，指数分布，正規分布など，多くの基本的な確率関数あるいは密度関数が対数凹 (*log-concave*) である．多変量分布に対しても，多変量正規分布をはじめとして，様々な対数凹分布が統計学，経済学，バイオインフォマティクス，オペレーションズ

リサーチなどの分野でしばしば現れ，その中で対数凹関数からのサンプリング法は様々な応用をもつ．

連続空間中の凸体上に定義される対数凹関数からのサンプリングに対しては，マルコフ連鎖を用いた近似サンプリング法がいくつか提案され研究されている．1999 年に，Friez と Kannan は “ball-walk” 型のマルコフ連鎖が多項式時間で収束することを対数 Sobolev 不等式を用いて示している [34]．2003 年に，Lovász と Vempala は ‘ball-walk’ 型と ‘hit-and-run’ 型のマルコフ連鎖が共に，任意の初期点に対し $\mathrm{O}(n^4\mathrm{poly}(\ln n, \ln \varepsilon^{-1}))$ で amortized には $\mathrm{O}(n^3\mathrm{poly}(\ln n, \ln \varepsilon^{-1}))$ で収束することをコンダクタンス法を用いて示している [72]．2005 年に，Randall と Winkler は単体上の一様分布に対して，近似サンプリングを実現するマルコフ連鎖について議論している [96]．彼女らは連続空間を取り扱っており，‘hit-and-run’ 型のマルコフ連鎖が $\Theta(n^3 \ln(n\varepsilon^{-1}))$ で収束することを示している．彼女らはこの上界を 2 段階カップリング法に基づいて，下界を確率的な 2 次モーメントの議論に基づいて示している．

本論文で扱う対数分離凹分布は連続空間中の対数凹分布に素朴に対応する離散分布のひとつである．しかし，上述の通り数多く存在する (連続空間中の) 対数凹分布に対する算定法を離散的な対数凹分布に直接適用することは困難である．その理由として，連続分布の算定に用いられるスケーリングの技法が離散分布に対しては一般に使えないこと，マルコフ連鎖の推移の方向が制限されることの 2 点が挙げられる．この問題は，組合せ最適化の分野を中心に議論されている “離散凸” の取り扱いの難しさに通じる [87, 88]．この点を踏まえ，先行研究のほとんどがコンダクタンス法に基づく混交時間の算定を行っているのに対して，本研究ではカップリング法に基づく算定を行う．

本論文では $n-1$ 次元単体中の整数格子点上に定義された対数分離凹分布に対して，2 つの新しい “hit-and-run” 型のマルコフ連鎖を提案する．1 つは効率的な近似サンプリング法を目的とするマルコフ連鎖である．このマルコフ連鎖が $\mathrm{O}(n^2 \ln(K\varepsilon^{-1}))$ 時間で収束することを経路カップリング法 [13] を用いて示す．ただし K は単体の一辺の長さである．この証明において交互不等式のアイデアを導入し，対数分離凹分布が交互不等式を満たすことを利用する．もう 1 つは効率的な完璧サンプリング法を目的とするマルコフ連鎖である．このマルコフ連鎖が「単調」であることを示し，単調 CFTP アルゴリズムに基づく完璧サンプリング法を実現する．単調性の証明においても交互不等式が証明の鍵となる．さらにこの単調マルコフ連鎖が高速に収束するための条件を提示し，条件を満たす場合に $\mathrm{O}(n^3 \ln(Kn))$ 時間で収束することを経路カップリング法を用いて示す．この経路カップリング法では，新たに提案する特殊な距離を用いている．

本論文ではまた，$n-1$ 次元単体中の整数格子点上に定義された対数分離凹関数の積分法について，MCMC 法に基づく乱択近似計算法を与える．提案する計算法は自己帰着性を利用した再帰的なモンテカルロ法で，上述の近似サンプリング法および完璧サンプリング法を利用する．提案する乱択近似計算法に関して，以下では 2 つの技術的な注意点を挙げる．1 つは，従来研究の多くが 2 分法に基づく再帰を行っているのに対して，本論文では (多次元) n 分法に基づく再帰を採用している点である．この変更により解法の解析が複雑となり，近似解の精度保証において注意深い議論が必要となる．もう 1 つは，得られる近似解の期待値の，真の値からの偏りについて議論を行う点

である．従来研究では，近似解の期待値の真の値からの偏りについての議論がほとんどされてこなかった．本論文では，実際の計算機実験に基づき，従来の方法では大きな偏りが生じる場合があることを確認した．本研究ではこの偏りを小さくするための方法を提案し，得られる近似解の期待値の真の値からの偏りを理論的に算定すると共に，実験的に偏りが小さくなることを示す．この議論により，しばしば実用の場面において現れる，精度の保証された近似解を得るのに必要な計算時間が確保できない場合でも，質の良い近似解を与えることが可能となる．

本論文では，先行研究と同様に，1 次元の対数凹分布からのサンプリングを基本操作として扱い，この操作にかかる計算時間についての詳細な議論は，FPRAS の議論を行う場合を除いて，行わない．これは，本研究が空間の次元の大きさに依存する問題の難しさに焦点を置き，次元の大きさに関して効率的なアルゴリズムの設計を目標とするからである．1 次元の様々な分布に対しては，2 分探索法，適応棄却サンプリング，スライスサンプリングなどの多くの手法が研究されている．それに比べて，高次元の分布に従うサンプリングに関する研究は多くなく，この問題に対して，MCMC 法は大きな成功を収めていると言える．

以下，$n-1$ 次元単体中の整数格子点上の対数分離凹分布の応用例について述べる．

待ち行列ネットワーク　待ち行列ネットワークは，通信ネットワーク，情報処理システムなどに広く応用を持ち，オペレーションズリサーチの分野を中心に古くから研究が行われてきた．ジャクソンネットワークをはじめとして BCMP ネットワークや呼損ネットワークなど，多くの重要な待ち行列ネットワークはネットワーク中のジョブの均衡分布として積形式解を持つことが知られている．

ジャクソンネットワークは待ち行列ネットワークにおける基本的で重要なモデルのひとつである．モデルでは，顧客は (単一または) 複数のサーバを持つノード上で先着順にサービスを享受し (FCFS: first-come-first-served)，ノード上でサービスを受け終わると次のノードへ確率的に推移する．ジャクソン [47] はジャクソンネットワーク中の顧客の定常分布が積形式解を持つことを示している [47, 48]．積形式解の正規化定数を計算することで平均滞在客数，平均滞在時間, スループットなどのネットワークの重要な評価基準を計算することができる．

積形式解の正規化定数の計算法としてよく知られるものに Buzen のアルゴリズムがある [16]．しかし，Buzen のアルゴリズムは閉ネットワーク中の顧客数に関する再帰計算を行うため，顧客数に比例する擬多項式時間アルゴリズムである．Chen と O'Cnneide [18] はマルコフ連鎖モンテカルロ法に基づく乱択近似アルゴリズムを提案し，非常に特殊な場合にのみ弱多項式時間アルゴリズムとなることを示している．また小沢，高橋，高橋 [91] は閉ジャクソンネットワーク単一サーバモデルに対して，完璧サンプリング法を提案しているが，このマルコフ連鎖の収束時間は擬多項式時間かかることがわかっている．

閉ジャクソンネットワークの積形式解は対数分離凹分布である．本論文では，閉ジャクソンネットワークの積形式解に対して，近似サンプリング法および，完璧サンプリング法を与える．先行研究における MCMC 法がネットワーク中の顧客の推移を模倣するマルコフ連鎖であったのに対し

表 1　分割表の一様サンプリングに関連する研究．

1985,	Diaconis, Effron [22],	2 元分割表の検定法の提案,
1995,	Diaconis, Saloff-Coste [23],	$m^* \times n^*$ 分割表の弱多項式時間近似一様生成法,
1997,	Dyer, Kannan, Mount, [28]	2 元分割表数え上げ問題の計算複雑度,
1998,	Hernek [42],	$2 \times n$ 分割表の擬多項式時間近似一様生成法,
2000,	Dyer, Greenhill [27],	$2 \times n$ 分割表の弱多項式時間近似一様生成法,
2002,	Cryan et al. [21],	$m^* \times n$ 分割表の弱多項式時間近似一様生成法,
2004,	Matsui, Matsui, Ono [76],	$2 \times \cdots \times 2 \times n$ 分割表の弱多項式時間近似一様生成法,
2006,	Kijima, Matsui [61],	$2 \times n$ 分割表の弱多項式時間完璧一様生成法,
		(注: m^*, n^* はそれぞれ定数であることを表す .)

て，本論文で与える 2 つのマルコフ連鎖は定常分布を積形式解にもつ人工的なマルコフ連鎖であり，ネットワーク中の顧客の推移を模倣するものではない．

特に単一サーバモデルの場合はマルコフ連鎖の推移が多項式時間で行えるので，提案する近似サンプリング法は多項式時間サンプリング法となる．これを用いて，積形式解の正規化定数の計算に対する全多項式時間乱択近似スキーム (FPRAS) を与える．さらに，単一サーバモデルに対しては単調マルコフ連鎖が $\mathrm{O}(n^3 \ln(nK))$ 時間で収束することを示す．したがって，提案する完璧サンプリング法の期待計算時間は多項式時間である．

分割表　分割表は与えられた行和，列和を満たすような非負整数からなる表 (行列) で，統計データを扱う際に用いられる．分割表に対する統計学的興味として，行と列の独立性があげられる．Fischer の提案した正確検定はその検定法の 1 つである [32]．更に [22] において，Diaconis と Effron は分割表の独立性の検定の議論を行っている．正確検定は分割表を全列挙することで可能となるが，分割表の個数が膨大で，全列挙することは現実的に困難である事が多い．実用的には，正確検定で用いる p 値の計算にマルコフ連鎖モンテカルロ (MCMC) 法が用いられる．

分割表の一様生成にマルコフ連鎖を用いる手法に関する研究は数多く存在する．Diaconis と Saloff-Coste [23] は 2 元分割表に対する，素朴なマルコフ連鎖の収束性について議論している．彼らは分割表の行数，列数を固定した時，素朴なマルコフ連鎖が多項式時間で収束することを示した．Dyer, Kannan, Mount [28] は 2 元分割表の個数を数える問題が行数を 2 に固定した場合でさえ#P 完全であることを示した．また，彼らはこの数え上げ問題の近似解法を設計する中で，別のマルコフ連鎖を提案している．周辺和が十分大きい時，彼らの提案したマルコフ連鎖は行数および列数の多項式時間で収束する．

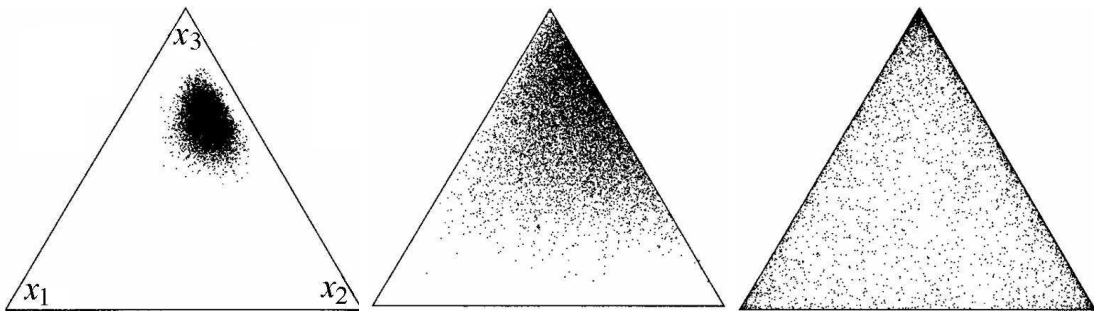
2 行分割表に対して，Hernek [42] は素朴なマルコフ連鎖の混交時間が表の合計値と列数の多項式時間で押さえられることを示した．Hernek はマルコフ連鎖の混交時間をカップリング法を用いて示している．Dyer と Greenhill [27] は 2 行分割表に対して，高速に収束する新しいマルコフ連鎖を提案した．そのマルコフ連鎖は表の合計値の対数と列数の多項式時間で混交する．彼らはそれ

を経路カップリング法を用いて示している．Matsui, Matsui, Ono [76] は Dyer, Greenhill の結果を $2 \times \cdots \times 2 \times J$ 分割表に拡張している．また，Cryan, Dyer, Goldberg, Jerrum, Martin [21] は行数 (または列数) 固定の時，Dyer と Greenhill のマルコフ連鎖を拡張した 2×2 連鎖が高速に収束することをコンダクタンス法を用いて示している．現在，一般の $m \times n$ 分割表に対して，高速に収束するマルコフ連鎖の存在については大きな未解決問題である．

本論文では 2 行分割表に対するサンプリング法を対数分離凹関数の観点から取り扱う．2 行分割表の一様分布および多項超幾何分布が $n-1$ 次元単体中の整数格子点上に定義された対数分離凹関数で表現できることを利用して，単調 CFTP アルゴリズムに基づく完璧サンプリング法を与える．特に，一様サンプリング法に対しては単調マルコフ連鎖が $\mathrm{O}(n^3 \ln nK)$ 時間で収束することを示す．したがって ,提案する完璧サンプリング法の期待計算時間は多項式時間である．また ,Dyer と Greenhill の一様分布に対する近似サンプリング法 [27] の多項式性および Matsui, Matsui, Ono の多項超幾何分布に対する近似サンプリング法 [76] の多項式性についてそれぞれの分布が対数分離凹関数で表現できることを利用した証明を与える．2 行分割表の近似数え上げ法に対して，新しい全多項式時間乱択近似計算法を提案する．提案手法に対して，従来あまり議論のされていなかった ,近似解の期待値の ,真の値からの偏りについて理論的な精度保証を与え ,Dyer と Greenhill の提案手法と計算機実験による比較を行う．

離散化 Dirichlet 分布　Dirichlet 分布は生物情報学の多くの統計的手法における多項分布の事前，事後分布としてしばしば現れる．生物情報学において，観測データから (共通の) 疾患を引き起こす遺伝子を見つけるための強力な道具として隠れマルコフモデルをはじめとする統計的手法が広く用いられている．これらの手法において EM アルゴリズム，MCMC 法，Gibbs サンプラーが用いられる．Dirichlet 分布は多項分布のパラメータの共役事前分布であり，これらの手法の多項分布の事前あるいは事後確率としてしばしば現れる [97]．例えば，Niu, Qin, Xu, Liu は各項目に対するディプロタイプの配置を確率的に定めるベイジアンハプロタイプ推定法を提案している [89]．他の例として， Pritchard, Stephens, Donnelly の母集団構造推定アルゴリズムが挙げられる [93]．このアルゴリズムでは MCMC 法を用いている．これらの例において，Dirichlet 分布は様々な次元で，様々なパラメータをとって現れる．したがって，任意の次元とパラメータを持つ Dirichlet 分布からサンプリングを行える効率的アルゴリズムが望まれる．Dirichlet 分布は同系交配係数とアレル頻度交換の効果の大きさを見積もり [71]，また多形性と疾患の関係を調べるためのメタ解析を行う際にも用いられる [15]．Burr は [14] で，細分された母集団中の希少アレルの分布に対する擬似均衡理論の検査に Dirichlet 分布を用いている．Kitada, Hayashi, Kishino は母集団と有効な母集団の大きさとの間の遺伝的距離の推定に Dirichlet 分布を用いている [70]．Graham, Curran, Weir は法医学のためのマイクロサテライト座位に対する条件付遺伝子型確率の見積もりに Dirichlet 分布を用いている [39]．

(連続) Dirichlet 分布からのサンプリングのひとつの方法としては棄却サンプリングがある (例えば [25] 参照). しかしパラメータが小さいと，$n = 2$ の場合 (ベータ分布) でさえ棄却の確率はすぐに大きくなってしまい，パラメータ値が小さいとき，効率的でない．他の手法として，領



(a) $(u_1, u_2, u_3) = (10, 20, 50)$.　(b) $(u_1, u_2, u_3) = (1, 2, 5)$.　(c) $(u_1, u_2, u_3) = (0.1, 0.2, 0.5)$.

図 1　3 次元 Dirichlet 分布 [25].

域を離散化し，Metropolis-Hastings アルゴリズムを用いる手法がある．近年，Matsui, Motoki, Kamatani が離散化 Dirichlet 分布に従うサンプリングのためのマルコフ連鎖を提案している [77]．そのマルコフ連鎖の混交時間は $\mathrm{O}(n^2 \ln(K\varepsilon^{-1})$ で押さえられる．

本論文では，離散化 Dirichlet 分布に対するサンプリング法を，対数分離凹分布の観点から取り扱う．Dirichlet パラメータがすべて 1 以上の場合は対数分離凹分布であり，また，一般のパラメータに対しても交互不等式が成り立つことを示し，単調 CFTP アルゴリズムに基づく完璧サンプリング法を与える．さらに，任意の Dirichlet パラメータに対して，提案する単調マルコフ連鎖が $\mathrm{O}(n^3 \ln K)$ 時間で収束することを示す．本論文で提案する単調マルコフ連鎖は Matsui, Motoki, Kamatani のマルコフ連鎖の推移のいくつかを制限したものであり，各推移では離散化ベータ分布 (2 次元の Dirichlet 分布) に従う確率変数の生成を行う．

1.3　本論文の構成

本論文の構成は以下の通りである．まず第 2 章では，本論文で用いる既存の手法を概観する．すなわち，マルコフ連鎖の混交時間の算定，完璧サンプリング法の実現，およびモンテカルロ法の精度保証を行うための技法について述べる．第 3 章では，$n-1$ 次元単体中の整数格子点上の対数分離凹分布からのサンプリング法を提案する．3.1 節で対数分離凹分布を定義し，3.2 節で近似サンプリング法を，3.3 節で完璧サンプリング法を取り扱う．第 4 章では，$n-1$ 次元単体中の整数格子点上の対数分離凹関数の積分に対する乱択近似スキームを提案する．4.1 節で基本的なアイデアについて述べ，4.2 節で完璧サンプリング法を用いた場合，4.3 節で近似サンプリング法を用いた場合を取り扱う．4.4 節ではより偏りの小さい計算法について議論する．第 5 章では応用について述べる．5.1 節では待ち行列ネットワークを，5.2 節では 2 元分割表を，5.3 節では離散化 Dirichlet 分布を扱う．第 6 章で本論文をまとめ，今後の課題について述べる．

2 準備

本論文では実数全体の集合を $\mathbb{R}$ で表し，また整数 (非負整数，正整数) 全体の集合をそれぞれ $\mathbb{Z}$ ($\mathbb{Z}_+$, $\mathbb{Z}_{++}$) で表す．

2.1 定常分布の設計

本論文では，有限の状態空間 Ω と推移確率行列 P をもつマルコフ連鎖 $\mathcal{M}$ を扱う．マルコフ連鎖 $\mathcal{M}$ が既約 (*irreducible*) であるとは任意の状態対 $\{x, y\} \in \binom{\Omega}{2}$ に対して，$\exists t > 0$, $\Pr(X^t = y \mid X^0 = x) > 0$ が成り立つことを言う．ただし，各 $t \geq 0$ に対する確率変数 X^t はマルコフ連鎖 $\mathcal{M}$ に従う確率的な時系列とする．マルコフ連鎖 $\mathcal{M}$ が非周期的 (*aperiodic*) であるとは $\forall x \in \Omega$, $\gcd\{t \in \mathbb{Z}_{++} \mid \Pr(X^t = x \mid X^0 = x) > 0\} = 1$ が成り立つことを言う．ただし gcd は最大公約数を表す．既約で非周期的な有限マルコフ連鎖をエルゴード的 (*ergodic*) と呼ぶ．エルゴード的なマルコフ連鎖は唯一の定常分布を持ち，極限分布は定常分布に一致する．次の定理はマルコフ連鎖の定常分布を設計する上で重要な定理である．

定理 2.1 ([41] 参照 .) 状態空間 Ω と推移確率行列 P を持つエルゴード的なマルコフ連鎖 $\mathcal{M}$ の定常分布を π とする．関数 $q : \Omega \to \mathbb{R}_{++}$ が次の詳細均衡式 (*detailed balance equation*)

$$q(x)P(x, y) = q(y)P(y, x), \qquad \forall \{x, y\} \in \binom{\Omega}{2} \tag{2.1.1}$$

を満たす時，関数 q に対して，正の定数 C が存在して，任意の $x \in \Omega$ に対して $\pi(x) = C \cdot q(x)$ が成り立つ．ただし，$P(x, y)$，$P(y, x)$ は，推移確率行列 P における x から y へ，y から x へのそれぞれの推移確率を表す． ■

2.2 マルコフ連鎖の収束時間の算定

2.2.1 カップリング補題

まず，状態空間 Ω 上の分布の間の距離を以下のように定義する．

定義 2.2 ([41] 参照 .) 同一の有限状態空間 Ω 上の 2 つの確率分布 ν_1 と ν_2 が与えられた時，ν_1 と ν_2 の間の全変動距離 (*total variation distance* [6]) は

$$\mathrm{d_{TV}}(\nu_1, \nu_2) \stackrel{\text{def.}}{=} \max_{A \subseteq \Omega} \left\{ \sum_{x \in A} (\nu_1(x) - \nu_2(x)) \right\} \equiv \frac{1}{2} \sum_{x \in \Omega} |\nu_1(x) - \nu_2(x)|$$

と定義される．

全変動距離の定義を用いて収束時間の尺度を以下のように定義する．

定義 2.3 （[41] 参照 .) 任意の正数 $\varepsilon < 1$ に対して，状態空間 Ω を持つエルゴードマルコフ連鎖 $\mathcal{M}$ の混交時間 (*mixing time*) は

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

と定義される．ただし，π はマルコフ連鎖 $\mathcal{M}$ の定常分布とし，P_x^s は初期状態 $x \in \Omega$ として時刻 0 から時刻 $s \geq 0$ まで推移させた時のマルコフ連鎖 $\mathcal{M}$ の確率分布とする．

また特に $\tau = \tau(1/\mathrm{e})$ を混交率 (*mixing rate*) と呼ぶ．

次のカップリング (*coupling*) は古くから知られる定理で，エルゴード的なマルコフ連鎖が唯一の定常分布に収束することを示す際に用いられる [3, 12, 41, 40]．

定理 2.4 （カップリング定理 ([41] 参照 .)) 確率変数 X, Y は同一の状態空間 Ω にあり，それぞれ確率分布 μ, ν にしたがう．この時 $\mathrm{d_{TV}}(\mu, \nu) \leq \Pr(X \neq Y) = \sum_{x \neq y} \Pr(X = x, Y = y)$ が成り立つ．

証明. 部分集合 $A^* \subseteq \Omega$ は $\max_{A \subseteq \Omega} \left\{\sum_{s \in A} (\mu(s) - \nu(s))\right\}$ を実現する A とする．この時，

$$\begin{aligned}
\mathrm{d_{TV}}(\mu, \nu) &= \max_{A \subseteq \Omega} \left\{ \sum_{s \in A} (\mu(s) - \nu(s)) \right\} = \sum_{s \in A^*} (\Pr(X = s) - \Pr(Y = s)) \\
&\leq \sum_{s \in A^*} \Pr(X = s, Y \neq s) \leq \sum_{s \in \Omega} \Pr(X = s, Y \neq s) = \Pr(X \neq Y)
\end{aligned}$$

より題意を得る． □

2.2.2 経路カップリング法

本小節では混交時間の上限を算定するための手法として，Bubley と Dyer [13] の提案した経路カップリング (*path coupling*) 法について述べる．下の定理は Bubley と Dyer のオリジナルのものに対して，枝の長さの整数性を仮定していない点が異なるが，最小距離 d を導入することで同様の証明が行うことができる．

定理 2.5 （経路カップリング [13]) 状態空間 Ω を持つマルコフ連鎖 $\mathcal{M}$ はエルゴード的とする．単純無向グラフ $\mathcal{H} = (\Omega, \mathcal{F})$ は頂点集合 Ω と枝集合 $\mathcal{F} \subseteq \binom{\Omega}{2}$ を持ち，連結なグラフとする．また，枝の長さを $l : \mathcal{F} \to \mathbb{R}_{++}$ とする．グラフ $\mathcal{H}$ の任意の頂点間 $\{x, y\}$ に対して x と y の間の距離を $d(x, y)$ あるいは $d(y, x)$ で表し，x と y の間のグラフ $\mathcal{H}$ 上の最短パスの長さとする．ただし，パスの長さとはパス上の枝の長さの合計のことであり，任意の $x \in \Omega$ に対して $d(x, x) = 0$ とする．いま $\mathcal{F}$ 上に定義されたジョイントプロセス $(X, Y) \mapsto (X', Y')$ は，X および Y に関する周辺推移確率が $\mathcal{M}$ に従い

$$0 < \exists \beta < 1,\ \forall \{X, Y\} \in \mathcal{F},\ \mathrm{E}[d(X', Y')] \leq \beta \cdot d(X, Y)$$

を満たすとする．この時，マルコフ連鎖 $\mathcal{M}$ の混交時間 $\tau(\varepsilon)$ は

$$\tau(\varepsilon) \leq (1-\beta)^{-1} \ln\left(\frac{D}{d\varepsilon}\right)$$

を満たす．ただし，$d \stackrel{\text{def.}}{=} \min\{d(x,y) \mid \forall\{x,y\} \in \binom{\Omega}{2}\}$ とし $D \stackrel{\text{def.}}{=} \max\{d(x,y) \mid \forall\{x,y\} \in \binom{\Omega}{2}\}$ とする．

証明. まず，$\mathcal{F}$ 上のジョイントプロセス $(X,Y) \mapsto (X',Y')$ が，任意の $\{X,Y\} \in \mathcal{F}$ に対して

$$\mathrm{E}[d(X',Y')] \leq \beta \cdot d(X,Y)$$

を満たすという定理の仮定の下で，Ω 上に拡張されたジョイントプロセス $(X,Y) \mapsto (X',Y')$ が存在して，任意の $(X,Y) \in \Omega^2$ に対して

$$\mathrm{E}[d(X',Y')] \leq \beta \cdot d(X,Y) \tag{2.2.1}$$

が成り立つことを示す．

いま，Ω 上のジョイントプロセス $(X,Y) \mapsto (X',Y')$ を以下のように定義する．$X = Y$ の場合には $X' = Y'$ とする．$\{X,Y\} \in \mathcal{F}$ の場合には，$\mathcal{F}$ 上のジョイントプロセスと同一とする．それ以外の場合，すなわち $X \neq Y$ かつ $\{X,Y\} \notin \mathcal{F}$ の場合には，$\{X,Y\}$ に対して，ある要素列 $X = Z_0, Z_1, \ldots, Z_p = Y$ $(Z_i \in \Omega)$ が存在して，[任意の $i \in \{0,\ldots,p-1\}$ に対して $\{Z_i, Z_{i+1}\} \in \mathcal{F}$] かつ $[d(X,Y) = \sum_{i=0}^{p-1} d(Z_i, Z_{i+1})]$ が成り立つ．この要素列 $Z_0, Z_1, \ldots, Z_p$ に対し，要素 Z'_i $(i \in \{0,\ldots,p\})$ を $\mathcal{F}$ 上のジョイントプロセスにしたがって順次 $(Z_i, Z_{i+1}) \mapsto (Z'_i, Z'_{i+1})$ と得られる状態とし，この場合のジョイントプロセスを $(X,Y) \mapsto (Z'_0, Z'_p)$ とする．

以上で定義した Ω 上のジョイントプロセス $(X,Y) \mapsto (X',Y')$ に対して，不等式 $(2.2.1)$ が成り立つことを示す．$X = Y$ の場合は不等式 $(2.2.1)$ の両辺とも 0 で成り立つ．$\{X,Y\} \in \mathcal{F}$ の場合には，$\mathcal{F}$ 上のジョイントプロセスと同一なので，定理の仮定から不等式 $(2.2.1)$ が成り立つ．それ以外の場合には，上で定義した Z'_i $(i \in \{0,\ldots,p\})$ を用いて，状態間の距離がグラフ上の最短パスの長さで定義されていることから，任意の推移に対して

$$d(X',Y') \leq \textstyle\sum_{i=0}^{p-1} d(Z'_i, Z'_{i+1})$$

が成り立つ．一般に $\{Z'_i, Z'_{i+1}\} \notin \mathcal{F}$ であることに注意されたい．以上の議論と期待値の線形性および定理の仮定から，任意の $(X,Y) \in \Omega^2$ に対して

$$\begin{aligned}\mathrm{E}\left[d(X',Y')\right] &\leq \mathrm{E}\left[\textstyle\sum_{i=0}^{p-1} d(Z'_i, Z'_{i+1})\right] = \textstyle\sum_{i=0}^{p-1} \mathrm{E}\left[d(Z'_i, Z'_{i+1})\right] \\ &\leq \textstyle\sum_{i=0}^{p-1} \beta \cdot d(Z_i, Z_{i+1}) = \beta \cdot d(X,Y)\end{aligned}$$

を得る．

以上の議論から，任意の $(X,Y) \in \Omega^2$ に対して式 $(2.2.1)$ が成り立つので，式 $(2.2.1)$ の両辺の，確率変数の対 $(X,Y) \in \Omega^2$ に関する期待値は

$$\mathrm{E}\left[d(X',Y')\right] \leq \beta \cdot \mathrm{E}\left[d(X,Y)\right] \tag{2.2.2}$$

を満たす．

いま，確率変数 X が分布 μ に従い，確率変数 Y が分布 π に従うとすると，定理 2.4 より，

$$\begin{aligned}\mathrm{d_{TV}}(\mu,\pi) &\le \Pr(X\neq Y)=\sum_{x\neq y}\Pr(X=x,Y=y)\\ &\le \sum_{x,y}\frac{d(x,y)}{d}\Pr(X=x,Y=y)=\frac{\mathrm{E}[d(X,Y)]}{d}\end{aligned}\tag{2.2.3}$$

が得られる．

時刻 $t\ge 0$ において，マルコフ連鎖 $\mathcal{M}$ に従う確率変数 X^t は分布 μ^t に従い，マルコフ連鎖 $\mathcal{M}$ に従う確率変数 Y^t は定常分布 π に従っているものとする．また，X^t, Y^t の推移はジョイントプロセス $(X,Y)\mapsto(X',Y')$ に従うものとする．この時，$t=-\ln\beta\,\ln(D/d\varepsilon)\le\tau(\varepsilon)=(1-\beta)^{-1}\ln(D/d\varepsilon)$ とすると，式 (2.2.2) および式 (2.2.3) より

$$\begin{aligned}\mathrm{d_{TV}}(\mu^t,\pi) &\le \frac{\mathrm{E}[d(X^t,Y^t)]}{d}\le\beta\frac{\mathrm{E}[d(X^{t-1},Y^{t-1})]}{d}\le\cdots\\ &\le\beta^t\frac{\mathrm{E}[d(X^0,Y^0)]}{d}\le\beta^t\frac{D}{d}\le-\mathrm{e}^{\ln\frac{D}{d\varepsilon}}\frac{D}{d}=\varepsilon\end{aligned}$$

が成り立つ．よって題意は示された． □

2.3 完璧サンプリング法

2.3.1 過去からのカップリング法

本節では 1996 年に Propp と Wilson の提案した過去からのカップリング（*CFTP*: *coupling from the past*）について述べる．CFTP のアイデアは以下のとおりである．エルゴード的なマルコフ連鎖を無限回推移させることで，状態分布は定常分布に収束する．いま，仮想的に無限の過去から無限回推移しているマルコフ連鎖が存在するならば，そのマルコフ連鎖の現在の状態は定常分布に厳密に従うランダムサンプルである．マルコフの連鎖の現在の状態が，過去のある時点の状態に依らず，現時点における唯一実現可能な状態であるという証拠が得られたなら，その状態は定常分布に従う確率標本である．これが CFTP の発想である．

有限の状態空間 Ω をもち，推移確率行列 P をもつマルコフ連鎖 $\mathcal{M}$ は，既約で非周期，すなわちエルゴード的とする．$\mathcal{M}$ の推移規則は，一様実数乱数 $\lambda\in[0,1)$ が与えられた時，更新関数（*update function*）と呼ばれる決定的関数 $\phi:\Omega\times[0,1)\to\Omega$ によって記述される．ここで更新関数は $\forall x,y\in\Omega,\ P(x,y)=\Pr(\phi(x,\Lambda)=y)$ が満たされていなければ成らない．乱数列 $\boldsymbol{\lambda}=(\lambda[t_1],\lambda[t_1+1],\ldots,\lambda[t_2-1])\in[0,1)^{t_2-t_1}$ が与えられた時の時刻 t_1 から t_2 への $\mathcal{M}$ の推移も決定的関数 $\Phi_{t_1}^{t_2}(x,\boldsymbol{\lambda}):\Omega\times[0,1)^{t_2-t_1}\to\Omega$ で表現される．ただし，時刻 t_1,t_2 $(t_1<t_2)$ に対して $\Phi_{t_1}^{t_2}(x,\boldsymbol{\lambda})\stackrel{\mathrm{def.}}{=}\phi(\phi(\cdots(\phi(x,\lambda[t_1]),\ldots,\lambda[t_2-2]),\lambda[t_2-1])$ と定義する．

これらの準備を用いて，標準的な CFTP アルゴリズムは以下のように記述される．

アルゴリズム 2.1（CFTP [94]）

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

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

Step 3. Ω の全ての状態について，乱数列 $\boldsymbol{\lambda}$ 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．

Step 4. 以下の合流確認（*coalescence check*）を行う．

(a) もし $\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_T^0(x, \boldsymbol{\lambda})$ ならば y を返し，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 $T := 2T$ としてステップ 2 に戻る．

アルゴリズム 2.1 の Step 4 (a) における，確率的事象 $\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_T^0(x, \boldsymbol{\lambda})$ を合流（*coalescence*）と呼ぶ．また，更新関数 ϕ を持つマルコフ連鎖 $\mathcal{M}$ の合流時間（*coalescence time*）T_* を

$$T_* \stackrel{\text{def.}}{=} \min\left\{ t > 0 \mid \exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_{-t}^0(x, \boldsymbol{\lambda}^t) \right\}$$

と定義する．ただし $\boldsymbol{\lambda}^t = (\lambda[-t], \lambda[-t+1], \ldots, \lambda[-1])$ は $[0,1)$ 上の一様実数乱数列である．ここで T_* は確率変数である．また，アルゴリズム 2.1 の中ではシミュレーションの開始時刻を $T := 2T$ と更新しているため，合流時間は陽に求まらないことに注意を要する．

定理 2.6 （CFTP [94]）エルゴードマルコフ連鎖 $\mathcal{M}$ は有限の状態空間 Ω をもち，単調な更新関数 $\phi : \Omega \times [0,1) \to \Omega$ で定義されているとする．この時，CFTP アルゴリズム（アルゴリズム 2.1）が確率 1 で有限時間停止するなら，得られる値は $\mathcal{M}$ の定常分布に厳密に従う確率変数の実現値である．すなわち，CFTP アルゴリズム（アルゴリズム 2.1）はマルコフ連鎖 $\mathcal{M}$ の定常分布からの完璧サンプリング（*perfect sampling*）を実現する．

証明. アルゴリズム 2.1 が停止し，得られた値を y とする．この時，停止した時点における反復でのシミュレーションの初期時刻を $T < 0$，推移に使う乱数の列を $\boldsymbol{\lambda} = (\lambda[T], \lambda[T+1], \ldots, \lambda[-1])$ とする．いま，一様乱数 $\lambda[s']\ (t \le \forall s' < T)$ を生成し，$\boldsymbol{\lambda}' = (\lambda[t], \lambda[t+1], \ldots, \lambda[T-1], \lambda[T], \lambda[T+1], \ldots, \lambda[-1])$ とする．ここで，$\lambda[s]\ (T \le s < 0)$ はアルゴリズムで得られた乱数であることに注意する．この時，このマルコフ連鎖は時刻 t の状態 $x \in \Omega$ に対して時刻 0 の状態 $\Phi_t^0(x, \boldsymbol{\lambda}')$ は P_x^t に従うランダムサンプルである．マルコフ連鎖 $\mathcal{M}$ はエルゴード的なので，$\forall \varepsilon,\ \exists t < T,\ \forall x \in \Omega,\ \mathrm{d_{TV}}(P_x^t, \pi) \le \varepsilon$ が成り立つ．一方，CFTP アルゴリズムで得られるサンプルは $\lambda[s']\ (t \le \forall s' < T)$ によらず，$y = \Phi_T^0(x, \boldsymbol{\lambda}) = \Phi_t^0(x, \boldsymbol{\lambda}')$ となる．よって題意を得る． □

定理 2.7 ([94]) 有限の状態空間を持ち，推移確率の与えられた任意のエルゴード的なマルコフ連鎖に対して，確率 1 で収束するような更新関数が存在する．

証明略． ■

2.3.2 単調マルコフ連鎖

定理 2.6 より，アルゴリズム 2.1 が確率 1 で有限時間停止するならば，無限時間のシミュレーション結果が (確率的に) 有限時間のアルゴリズムで得られる．しかし，このアルゴリズムでは状態空間の状態数回のシミュレーションを行う必要があり，状態数の多い状態空間に対しては実現が困難である．この問題に対して，マルコフ連鎖が次に定義する単調性を持つ時，効率化を図ることが可能となる．

定義 2.8 (単調更新関数 [94]) 更新関数が単調 (*monotone*) であるとは，次の 3 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 "$\succeq$" が存在する．

条件 2. 半順序集合 $(\Omega, \succeq)$ に対して，唯一の最大元 $x_{\max}$ および最小元 $x_{\min}$ の対が存在する．すなわち，任意の $x \in \Omega$ に対して，$x_{\max} \succeq x \succeq x_{\min}$ が成り立つ．

条件 3. 任意の推移は半順序を保存する．すなわち，$x \succeq y$ を満たす任意の状態対 $\{x, y\} \in \binom{\Omega}{2}$ および任意の実数 $\lambda \in [0, 1)$ に対して，$\phi(x, \lambda) \succeq \phi(y, \lambda)$ が成り立つ．

また簡単のため，マルコフ連鎖が単調な更新関数で定義される時，マルコフ連鎖が単調 (*monotone*) であると言う．単調マルコフ連鎖に対して，次の補題が成り立つ．

補題 2.9 (単調 CFTP [94]) エルゴードマルコフ連鎖 $\mathcal{M}$ は有限の状態空間 Ω をもち，単調な更新関数 $\phi : \Omega \times [0, 1) \to \Omega$ で定義されているとする．この時，全状態 Ω からの合流の条件

$$\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_T^0(x, \boldsymbol{\lambda})$$

は，半順序集合 $(\Omega, \succeq)$ の最大元 $x_{\max}$ と最小元 $x_{\min}$ からの合流条件

$$\exists y \in \Omega,\ y = \Phi_T^0(x_{\max}, \boldsymbol{\lambda}) = \Phi_T^0(x_{\min}, \boldsymbol{\lambda})$$

と等価である．

証明略． ■

補題 2.9 から，アルゴリズム 2.1 の Step 3 を変更した次の単調 **CFTP** (*monotone CFTP*) アルゴリズムが設計できる．

アルゴリズム 2.2 (単調 CFTP [94])

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

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

Step 3. Ω の 2 つの状態 $x_{\max}$ と $x_{\min}$ について，乱数列 $\boldsymbol{\lambda}$ 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．

Step 4. 以下の合流確認を行う．

(a) もし $\exists y \in \Omega,\ y = \Phi_T^0(x_{\max}, \boldsymbol{\lambda}) = \Phi_T^0(x_{\min}, \boldsymbol{\lambda})$ ならば y を返し，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 $T := 2T$ としてステップ 2 に戻る．

単調 CFTP アルゴリズム (アルゴリズム 2.2) に対して，次の定理が成り立つ．

定理 **2.10** （ 単調 CFTP [94]) エルゴードマルコフ連鎖 $\mathcal{M}$ は有限の状態空間 Ω をもち，単調な更新関数 $\phi : \Omega \times [0,1) \to \Omega$ で定義されているとする．この時，単調 CFTP アルゴリズム (アルゴリズム 2.2) は確率 1 で有限時間停止し，得られる値は $\mathcal{M}$ の定常分布に厳密に従う確率変数の実現値である．さらに，マルコフ連鎖の 1 回の推移にかかる計算量を $\mathcal{T}$，合流時間を T_* とするとアルゴリズム 2.2 の計算量は $\mathrm{O}(\mathcal{T} \cdot T_*)$ である．

証明. マルコフ連鎖 $\mathcal{M}$ の合流時間を T_* とし，$K = \lceil \log_2 T_* \rceil$ とする．アルゴリズム 2.2 は $(K+1)$ 回目の反復で T を -2^K とした時に終了する．したがってアルゴリズム 2.2 で生成する乱数の総数は $2^K \leq 2T_*$ で押さえられ，推移回数は $2(2^0 + 2^1 + 2^2 + \cdots + 2^K) < 2 \cdot 2 \cdot 2^K \leq 8T_*$ で押さえられる．以上の議論から題意を得る． □

2.3.3 Read Once アルゴリズム

単調 CFTP により，高速なパーフェクトサンプリング法の設計が可能となるが，(単調) CFTP アルゴリズムにおいて，乱数の記憶領域が問題となる．CFTP アルゴリズムでは合流が確認できなかった場合には，更に過去に遡ってシミュレーションを行わなければならないが，このとき，生成された乱数を記憶しておかなければならない．たとえば，単調 CFTP アルゴリズムならば，計算時間に比例する記憶領域が要求される．本小節ではこの点について改良された，Wilson の Read Once アルゴリズム [106] について述べる

これまでは時刻 $t < 0$ から時刻 0 の間の推移による合流だけを考えていたが，これを一般化し，時刻 t_1 から時刻 t_2 までの推移における，(全状態からの) 合流を，

(乱) 数列 $\boldsymbol{\lambda} \in [0,1)^{t_2 - t_1}$ に対して，$\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_{t_1}^{t_2}(x, \boldsymbol{\lambda})$ が成り立つ

と定義する．

以下，マルコフ連鎖の推移を定義している更新関数は，合流する確率が正 (非零) であると仮定する．図 2 は t_1 から t_2 の間の推移を模式的に表したものである．図 2 の左側は，全状態について数列 $\boldsymbol{\lambda}_s$ を用いて時刻 t_1 から時刻 t_2 まで推移を行い，時刻 t_2 で合流している様子を表している．右側は，数列 $\boldsymbol{\lambda}_f$ を用いると時刻 t_2では合流していない様子である．図 2 は単調マルコフ連鎖の最大元と最小元からの推移を表していると思って見るとイメージしやすい．ただし以下の議論では，計算量の議論の一部を除いて，更新関数の単調性の仮定は必要としない．

まず， 特殊な CFTP アルゴリズムを導入する．

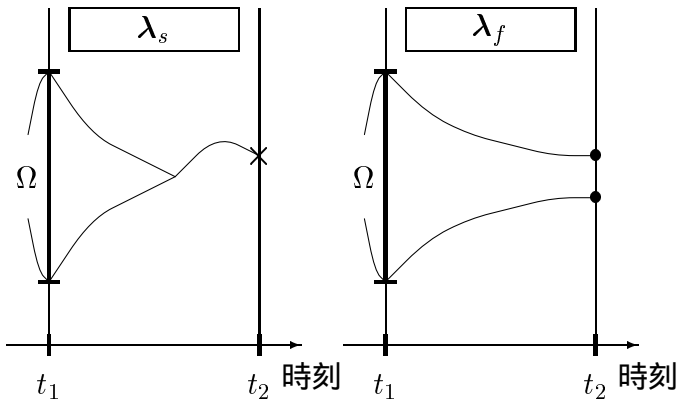


図 2　時刻 t_1 から時刻 t_2 までの推移の模式図

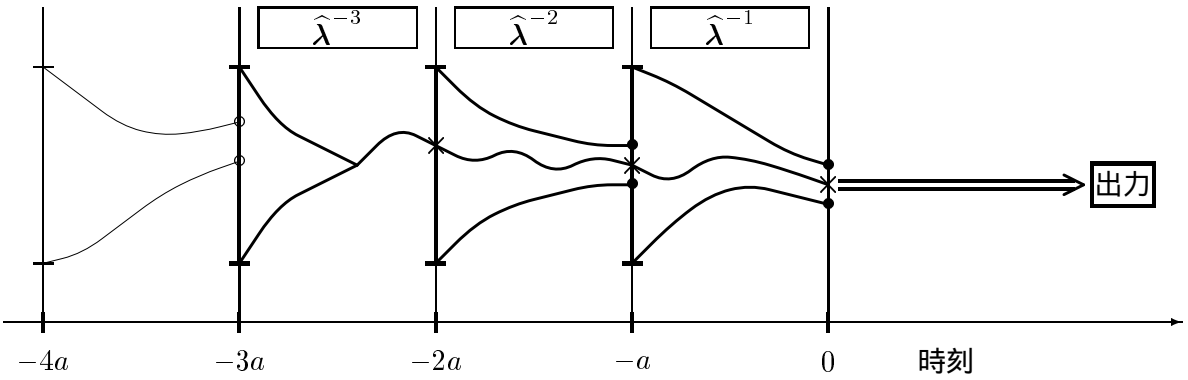


図 3　a 遡る CFTP アルゴリズムの概念図

アルゴリズム 2.3（a 遡る CFTP アルゴリズム）

Step 0. 正整数 $a > 0$ を決める．

Step 1. シミュレーションの開始時刻を $T := 0$ とする．空列 $\boldsymbol{\lambda} = ()$ を用意する．反復回数 $r := 1$ とする．

Step 2. シミュレーションの開始時刻を $T := T - a$ とする．

Step 3. 乱数列 $\widehat{\boldsymbol{\lambda}}^{-r} = (\widehat{\lambda}[T], \ldots, \widehat{\lambda}[T + a - 1])$ を生成し，数列 $\widehat{\boldsymbol{\lambda}}$ の先頭に挿入する．すなわち，$\widehat{\boldsymbol{\lambda}} := (\widehat{\boldsymbol{\lambda}}^{-r}; \widehat{\boldsymbol{\lambda}}^{-r+1}; \cdots; \widehat{\boldsymbol{\lambda}}^{-1}) = (\widehat{\lambda}[T], \widehat{\lambda}[T + 1], \ldots, \widehat{\lambda}[-1])$ とする．

Step 4. Ω のすべての状態について，共通の乱数列 $\widehat{\boldsymbol{\lambda}}^{-r}$ を用いて時刻 T から時刻 $T + a$ までマルコフ連鎖を推移させる．このとき，

(a) もし，時刻 $T + a$ で合流していれば (すなわち $\exists z \in \Omega$, $\forall x \in \Omega$, $z = \Phi_T^{T+a}(x, \widehat{\boldsymbol{\lambda}}^{-r})$)，時刻 0 の状態 $y = \Phi_T^0(x, \widehat{\boldsymbol{\lambda}})$ を出力し，停止する．

(b) そうでなければ，反復回数を $r := r + 1$ として Step 2 に戻る．

図 3 はアルゴリズム 2.3 が 3 反復 ($r = 3$) で終了した様子を表している．上記のアルゴリズムが完璧サンプリングを実現している事は容易に示される．アルゴリズム 2.3 において正整数 a の

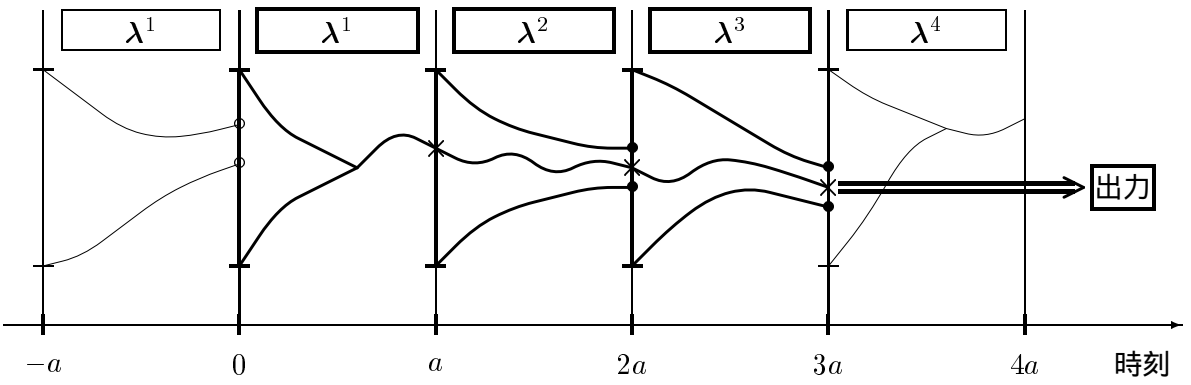


図 4　Read Once アルゴリズムの概念図

決め方は重要である．時間 a で合流する確率を p とすると，アルゴリズム 2.3 の期待計算時間は a/p である．すなわち，合流するような乱数列が存在しないほど a が小さいと，アルゴリズム 2.3 は停止しない．また，a を合流時間の期待値 $\mathrm{E}[T_*]$ 程度にすると p は定数となるので，例えば単調 CFTP ならば期待計算時間は $\mathrm{O}(\mathrm{E}[T_*])$ となり，標準単調 CFTP と同程度の計算時間となる．以上の準備をふまえて，Read Once アルゴリズムを記述する．

アルゴリズム 2.4（Read Once [106]）

Step 0. 正整数 $a > 0$ を与える．

Step 1. 乱数列 $\boldsymbol{\lambda}^1 = (\lambda[0], \lambda[1], \ldots, \lambda[a-1])$ を生成する．Ω のすべての状態について，共通の乱数列 $\boldsymbol{\lambda}^1$ を用いて，時刻 0 から時刻 a までマルコフ連鎖を推移させる．このとき，

(a) もし，時刻 0 と時刻 a の間で合流していれば（すなわち $\exists z \in \Omega, \forall x \in \Omega, z = \Phi_0^a(x, \boldsymbol{\lambda}^1)$），変数 y に時刻 a の状態を代入する．すなわち $y := \Phi_0^a(x, \boldsymbol{\lambda}^1)$ とする．反復回数を $r := 1$ とする．

(b) そうでなければ，Step 1 へ戻る．

Step 2. 乱数列 $\boldsymbol{\lambda}^{r+1} = (\lambda[ra], \lambda[ra+1], \ldots, \lambda[(r+1)a-1])$ を生成する．Ω のすべての状態について，共通の乱数列 $\boldsymbol{\lambda}^{r+1}$ を用いて，時刻 ra から時刻 $(r+1)a$ まで推移させる．特に $y' = \Phi_{ra}^{(r+1)a}(y, \boldsymbol{\lambda}^{r+1})$ とする．

(a) もし，時刻 ra と時刻 $(r+1)a$ の間で合流していれば（すなわち $\exists z \in \Omega, \forall x \in \Omega, z = \Phi_{ra}^{(r+1)a}(x, \boldsymbol{\lambda}^{r+1})$），時刻 ra の状態 y を出力し，停止する．

(b) そうでなければ，$y := y'$ と更新する．反復回数を $r := r+1$ とし，Step 2 に戻る．

図 4 は Read Once アルゴリズム（アルゴリズム 2.4）が 3 反復（$r = 3$）で終了した様子を表している．尚，アルゴリズム 2.4 では，記述の便宜上，数列 $\boldsymbol{\lambda}^r$ $(r \in \mathbb{Z}_{++})$ と表記するが，実際には，この数列を記憶する必要は無いことに注意されたい．

定理 2.11（Read Once [106]）エルゴード的なマルコフ連鎖に対して，ある a が存在して，アルゴリズム 2.4 は確率 1 で有限停止するなら，アルゴリズム 2.4 の返す値は定常分布に厳密に従う．

証明略． ■

Read Once アルゴリズム (アルゴリズム 2.4) のアイデアは，a 遡る CFTP アルゴリズム (アルゴリズム 2.3) の模倣である．アルゴリズムからもわかるとおり，Read Once アルゴリズムでは乱数を記憶する必要が無い．Read Once アルゴリズムの期待計算時間も，a 遡るアルゴリズムと同様であることがわかる．Read Once アルゴリズムの Step 1 で，合流した状態を得るための前処理が行われていることから，正確には，Read Once アルゴリズムで 1 つのサンプルを得るのに必要な期待計算時間は a 遡る CFTP アルゴリズムの 2 倍程度であることがわかる．

Read Once アルゴリズムの長所として，アルゴリズムの出力する状態 y と，アルゴリズム 2.4 の終了判定のために行った推移の合流した状態 (Step 2 (a) で得られた状態 z) は互いに独立，となる点がある．したがって，サンプルが複数個必要な時に，アルゴリズム 2.4 を連続して実行することができる．具体的にはサンプルが N 個必要な場合にはアルゴリズム 2.4 の Step 2 (a) を次のように置き換えれば良い．

Step 2. (a) もし時刻 $(r+1)a$ で合流していれば y を出力する．もし N 個のサンプルが出力されていたら停止する．そうでなければ，$y := \Phi_{ra}^{(r+1)a}(x, \boldsymbol{\lambda}^{r+1})$ とし，$r := 1$ と更新して Step 2 に戻る．

このように置き換えられた Read Once アルゴリズムで N 個のサンプルを生成するのに必要な期待計算時間は $(N+1)a/p$ となるので，a 遡る CFTP アルゴリズムとほぼ同じ計算時間で，乱数を記憶することなくサンプルが得られることがわかる．

Twin Run アルゴリズム　Read Once アルゴリズムの計算時間は a の決め方に大きく依存する．特に a が小さいと，アルゴリズムは止まらない可能性がある．次に，パラメータ a を全く与えないで Read Once アルゴリズムを実現する Wilson [106] の提案した Twin Run アルゴリズムを記述する．

Twin Run アルゴリズムのアイデアは，サンプリングとは別の乱数列を生成してステップバックする時間を定めるものである．まず準備として，アルゴリズム中で呼び出す 2 つの手続きを導入する．

プロセス 2.12 (*-successful)

Step 1. シミュレーションの終了時刻 T を $T := 1$ に設定する．

Step 2. 2 つの乱数 $\lambda_1[T-1]$, $\lambda_2[T-1]$ を生成する．便宜上，$\boldsymbol{\lambda}_1 = (\lambda_1[0], \lambda_1[1], \ldots, \lambda_1[T-1])$, $\boldsymbol{\lambda}_2 = (\lambda_2[0], \lambda_2[1], \ldots, \lambda_2[T-1])$ と表す．数列 $\boldsymbol{\lambda}_1$ と $\boldsymbol{\lambda}_2$ を用いて，それぞれ推移を実行する．このとき，

(a) 数列 $\boldsymbol{\lambda}_1$ および $\boldsymbol{\lambda}_2$ について，ともに時刻 T で合流している (すなわち $\exists z_1, z_2 \in \Omega$, $\forall x \in \Omega$, $z_1 = \Phi_0^T(x, \boldsymbol{\lambda}_1)$, $z_2 = \Phi_0^T(x, \boldsymbol{\lambda}_2)$) 場合．先に合流していた乱数列の添え字を $j \in \{1, 2\}$ とする．ただし，同時の場合は $j = 1$ とする．数列 $\boldsymbol{\lambda}_j$ に対する時刻 T の状態 $z_j = \Phi_0^T(x, \boldsymbol{\lambda}_j)$ を返して終了する．

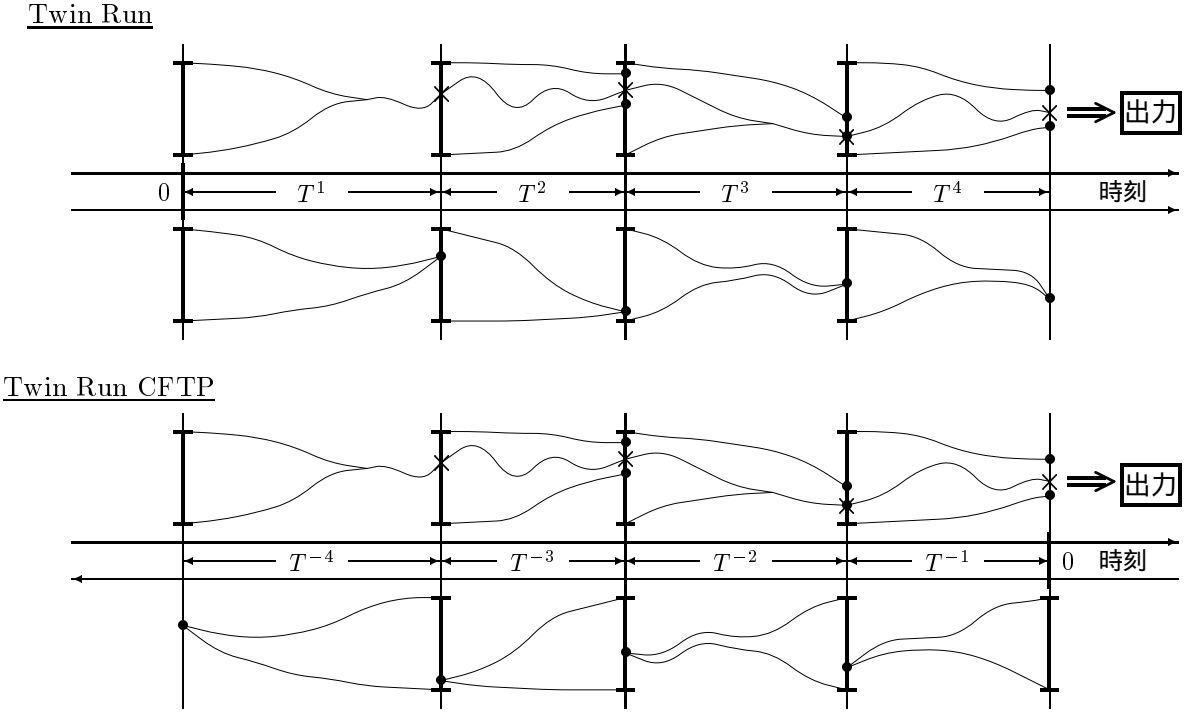


図 5　Twin Run アルゴリズムの概念図

(b) それ以外の時，$T := T + 1$ とし Step 2 に戻る．

プロセス 2.13 (*-failing)

Step 0. 入力された状態を $y \in \Omega$ とする．

Step 1. シミュレーションの終了時刻 T を $T := 1$ に設定する．

Step 2. 2 つの乱数 $\lambda_1[T-1]$, $\lambda_2[T-1]$ を生成する．便宜上，$\boldsymbol{\lambda}_1 = (\lambda_1[0], \lambda_1[1], \ldots, \lambda_1[T-1])$, $\boldsymbol{\lambda}_2 = (\lambda_2[0], \lambda_2[1], \ldots, \lambda_2[T-1])$ と表す．数列 $\boldsymbol{\lambda}_1$ と $\boldsymbol{\lambda}_2$ を用いて，それぞれ推移を実行する．特に $y_1 := \Phi_0^T(y, \boldsymbol{\lambda}_1)$，$y_2 := \Phi_0^T(y, \boldsymbol{\lambda}_2)$ とする．このとき，

(a) 数列 $\boldsymbol{\lambda}_1$ と $\boldsymbol{\lambda}_2$ の少なくとも一方が時刻 T で合流している場合．合流していない乱数列の添え字を $j \in \{1, 2\}$ とする．同時の場合は $1/2$ の確率で j を 1 または 2 とする．状態 $y_j \in \Omega$ を返して終了する．

(b) それ以外の時，$T := T + 1$ とし Step 2 に戻る．

以上の準備の下，Twin Run アルゴリズムを記述する．

アルゴリズム 2.5 （Twin Run [106]）

Step 1. プロセス 2.12 を実行し，出力を x とする．$1/2$ の確率で (a),(b) のいずれかを実行する．

(a) $x \in \Omega$ を出力しアルゴリズムを終了する．

(b) Step 2 に進む．

Step 2. 入力 $x \in \Omega$ を与えてプロセス 2.13 を実行し，変数 $x \in \Omega$ を出力された値に更新する．1/2 の確率で次の (a), (b) のいずれかを実行する．

(a) $x \in \Omega$ を出力しアルゴリズムを終了する．

(b) Step 2 にもどる．

Twin Run アルゴリズム (アルゴリズム 2.5) は，次の CFTP アルゴリズムを模倣する．

アルゴリズム 2.6 (Twin Run CFTP)

Step 1. シミュレーションの開始時刻を $T := 0$ とする．空列 $\boldsymbol{\lambda} = ()$ を用意し，反復回数を $r := 1$ にする．

Step 2. 前向きカップリングを実行し，その合流時間を $T^{-r} > 0$ とする．シミュレーションの開始時刻を $T := T - T^{-r}$ とする．

Step 3. 乱数列 $\boldsymbol{\lambda}^{-r} = (\lambda[T], \ldots, \lambda[T + T^{-r} - 1])$ を生成し，数列 $\boldsymbol{\lambda}$ の先頭に挿入する．すなわち，$\boldsymbol{\lambda} := (\boldsymbol{\lambda}^{-r}; \boldsymbol{\lambda}^{-r+1}; \cdots; \boldsymbol{\lambda}^{-1}) = (\lambda[T], \lambda[T+1], \ldots, \lambda[-1])$ とする．

Step 4. Ω のすべての状態について，共通の乱数列 $\boldsymbol{\lambda}^{-r}$ を用いて時刻 T から時刻 $T + T^{-r}$ までマルコフ連鎖を推移させる．このとき，

(a) もし ,時刻 $T+T^{-r}$ で合流していれば(すなわち ,$\exists z \in \Omega, \forall x \in \Omega, z = \Phi_T^{T+T^{-r}}(x, \boldsymbol{\lambda}^{-r})$), 時刻 0 の状態 $y = \Phi_T^0(x, \boldsymbol{\lambda})$ を出力し，停止する．

(b) そうでなければ，反復回数を $r := r + 1$ として Step 2 に戻る．

アルゴリズム 2.5 と 2.6 が本質的に等しい事は容易に分かる．以下ではアルゴリズム 2.6 の基本的なアイデアを簡単に記す．図 5 の上半分は，Twin Run アルゴリズムが 4 反復 ($r = 4$) で終了した様子を，下半分はアルゴリズム 2.6 が 4 反復 ($r = 4$) で終了した様子を表している．図 5 の Twin Run CFTP (下半分) の上段は，アルゴリズム 4 で出力する状態を決定する推移を示しており，これは本質的に Twin Run (上半分) の上段と同じものである．図 4 最下段の時間軸が右から左へ向かっているのは，値 T^{-r} を決定するのに，(別の乱数列を用いて) 前向き coupling を行っている事を表している．実はアルゴリズム 2.6 は, a 遡るアルゴリズム (アルゴリズム 2.3) では固定されていた遡る時間 (a) を，反復毎に Step 2 で決定するように変更したものである．アルゴリズム 2.6 は本質的に 2 本の乱数列を用いているが，出力を決定するのに用いているのはそのうち 1 本 (上段) だけであり，2 本目の乱数列 (下段) は遡る幅を決定する時計の役割にしか用いていない．

確率変数 T^{-r} の分布関数は前向きカップリングの合流時間 T^* の分布関数に等しく，これは，1 本目の乱数列 (上段) を使った CFTP の合流時間 T_* の分布関数にも等しい．したがってアルゴリズム 4 の各反復において，(Step 4 (a) で) アルゴリズムが終了する確率はいつも 1/2 である．すなわちアルゴリズム 2.6 は，合流時間 T_* を知ることなく，アルゴリズム 2.3 を毎反復 1/2 の確率で終了させる仕掛けとなっている. このことから明らかなように，アルゴリズム 2.6 の反復回数の期待値は 2 回である．

アルゴリズム 2.5 と 2.6 が本質的に等しいことから，以下の定理が成り立つ．

定理 2.14 （Twin Run [106]）エルゴード的な有限マルコフ連鎖に対して，Twin Run アルゴリズム（アルゴリズム 2.5）は確率 1 で有限停止するなら，アルゴリズム 2.5 の返す値は定常分布に厳密に従う．

証明略． ■

Twin Run アルゴリズムの特徴は，パラメータ a の設定を不要とするアイデアに加えて，手続き A, B を持ち込むことで，記憶領域を殆ど必要としないという成果を同時に達成している点である．

2.3.4 Fill の中断可能アルゴリズム

これまでに説明してきた CFTP アルゴリズムおよび Read Once アルゴリズムは終了時間が確率的なアルゴリズムであり，途中で中断してしまうと，偏りが生じてしまう．この小節では Fill の中断可能アルゴリズムについて述べる．この小節でもまた，状態空間 Ω，推移確率行列 P，更新関数 ϕ および定常分布 π を持つエルゴードマルコフ連鎖を考える．Fill のアルゴリズムの説明に入る前に新たに *reversal chain* を定義する．reversal chain は状態空間 Ω をもつマルコフ連鎖で，次のように定義される推移確率行列 $\widetilde{P}$ をもつ．

$$\widetilde{P}(y,x) \stackrel{\text{def.}}{=} \frac{\pi(x)P(x,y)}{\sum_{x'\in\Omega}\pi(x')P(x',y)}$$

これらの準備の下，Fill のアルゴリズム [30, 31] は次のように記述される．

アルゴリズム 2.7 （Fill [30, 31]）

Step 1. $T \in \mathbb{Z}_{++}$ と $y^* \in \Omega$ を選ぶ．(どのように選んでも良い．)

Step 2. 時刻 $t = T-1$ とし，状態 $X' = y^*$ とする．配列 $\boldsymbol{\lambda} = (\lambda[0], \lambda[1], \ldots, \lambda[T-1])$ を用意する．

Step 3. 状態 X' に対して，reversal chain の推移確率行列 $\widetilde{P}$ に従って状態 X へ推移する．

Step 4. 状態 X, X' に対して，更新関数 $\phi(X, \lambda[t]) = X'$ を満たすような乱数 $\lambda[t]$ を一様ランダムに選ぶ．

Step 5. $t > 0$ の場合，$t := t-1$ とし，状態 X' に X を代入して Step 3 に戻る．そうでなければ，変数 z に $z := X$ を代入する．

Step 6. 数列 $\boldsymbol{\lambda}$ を用いて合流確認を行う．

(a) もし合流していたら（すなわち $\forall x \in \Omega,\ y^* = \Phi_0^T(x, \boldsymbol{\lambda})$），$z$ を出力して停止する．

(b) そうでなければ，何も値を返さずに停止する．

定理 2.15 （Fill [30, 31]）アルゴリズム 2.7 により出力された値は定常分布 π に厳密に従う確率変数である．

証明略． ■

Fill のアルゴリズムは次のように解釈できる．

i) 状態 $X^0 \sim \pi$ を仮想的に サンプルする．

ii) 状態 $y^* \in \Omega$ を 選ぶ (どのように選んでも良い) .

iii) もし時刻 0 から時刻 T まで推移させたマルコフ連鎖の時刻 T の状態が y^* に合流していたなら，X^0 を定常分布からのサンプリングとして採用し，それ以外の場合は棄却することに決める．

iv) 時刻 T で状態 y^* に合流していると 仮定する．

v) (X^0 の値が何であっても) マルコフ連鎖の時刻 T の値は y^* である (と仮定している) ので，そのようなマルコフ連鎖のサンプルパスをランダムに 決める．

($\Rightarrow$ このマルコフ連鎖について分かっていることは，時刻 T の状態が y^* と言うことだけなので，時刻 T から reversal chain の推移確率行列 $\widetilde{P}$ を使ってシミュレートする．)

vi) 時刻 0 の状態を z とする．

vii) この時，同時にサンプルパス $X^t = \Phi_0^t(z, \boldsymbol{\lambda}_t)\ (0 \leq t < T)$ を実現するような乱数の列 $\boldsymbol{\lambda} = (\lambda[0], \lambda[1], \ldots, \lambda[T-1])$ (この数列は一意ではないことに注意) を 探す．

viii) この数列 $\boldsymbol{\lambda}$ が本当に合流しているか確認する．合流していたら (すなわち $\forall x \in \Omega$, $\Phi_0^T(x, \boldsymbol{\lambda}) = y^*$: このとき，合流するなら必ず y^* である．) , $z \in \Omega$ をサンプルとして 受理する．そうでなければ ,(y^* で合流していないのだから) 棄却する．

Fill のアルゴリズムを理解する上で鍵となるのが reversal chain である．reversal chain の推移確率行列 $\widetilde{P}$ について考察する．ここで，

$$\pi(x)P(x,y) = \Pr(X^{t-1} = x,\ X^t = y \mid X^0 \sim \pi)$$

に注意すると，

$$\begin{aligned}
\widetilde{P}(y,x) &= \frac{\pi(x)P(x,y)}{\sum_{x' \in \Omega} \pi(x')P(x',y)} \\
&= \frac{\Pr(X^{t-1} = x,\ X^t = y \mid X^0 \sim \pi)}{\sum_{x' \in \Omega} \Pr(X^{t-1} = x',\ X^t = y \mid X^0 \sim \pi)} \\
&= \frac{\Pr(X^{t-1} = x,\ X^t = y \mid X^0 \sim \pi)}{\Pr(X^t = y \mid X^0 \sim \pi)} \\
&= \Pr(X^{t-1} = x \mid X^t = y,\ X^0 \sim \pi)
\end{aligned}$$

である．このことは定常状態にあるマルコフ連鎖について，時刻 t の状態が $y \in \Omega$ であるという条件での時刻 $t-1$ の状態が reversal chain によって実現できることを意味する．

2.3.5 完璧サンプリング法のための各アルゴリズムの性能の考察

CFTP アルゴリズム，Read Once アルゴリズム，Fill の中断可能アルゴリズムに共通していえることは，アルゴリズムの効率化を図る上で，合流の確認を効率的に行うことが必要不可欠という点である．特にマルコフ連鎖の全状態からの合流の確認という条件は，MCMC 法が状態空間の非常に大きな対象を扱うことを鑑みても，重要な問題となる．この点について，単調なマルコフ連鎖

の設計は，いずれのアルゴリズムにおいても共通に効率化がなされる重要な条件となる．単調マルコフ連鎖が得られれば，現実的な実装が可能となる．

3 つのアルゴリズムを比較すると，まず，CFTP アルゴリズムには 2 つの問題点がある．ひとつは最悪ケースで無限のメモリ領域を確保する必要がある点である．もうひとつは計算打ち切りによる歪みの問題である．これに対して，Read Once アルゴリズムは CFTP の 2 つの問題点の内メモリの点について改良されている．すなわち，定数サイズのメモリ領域で動き，また計算時間も CFTP の 3 倍程度で終了する．しかし途中打ち切りによる歪みは解消されない．一方，Fill のアルゴリズムは計算の打ち切りによる歪みが起こらないことを保障している．また，メモリ領域に関しても，合流時間程度の有限の領域で計算が行える．計算時間についても，reversal chain が容易に構成できると言う条件の下で，CFTP の 2 倍程度で終了する．

実装のことを考えるならば，メモリ領域について心配のない Read Once アルゴリズム，特に Twin Run アルゴリズムが一番容易と思われる．CFTP と Fill のアルゴリズムについては合流時間をあらかじめ (予備実験等で) 知り，メモリ容量を考慮した設計を行う必要があるが，一般に合流時間は問題の入力値に依存する．しかし，合流時間が分かっていれば，最もシンプルな CFTP アルゴリズムが最速となりうる．CFTP，Read Once のいずれについても最悪ケースの起こる確率は時間漸近的に 0 であるが，打ち切りの誤差は理論的に回避できない．この意味で，Fill のアルゴリズムはロバストなアルゴリズムと言える．

もちろん，以上の考察は一般論であって，それぞれのアルゴリズムに対して，所与のマルコフ連鎖がその特徴を積極的に用いた効率化が可能な場合には上の議論は当てはまらない．

2.3.6　合流時間と混交時間

CFTP アルゴリズムにおける合流時間と，近似サンプリングにおいてマルコフ連鎖の収束の速さの指標となる混交時間との間には，次のような関係があることが知られている [94, 24]．まず，以下の議論を容易にするために，前向きカップリング (*CITF*: *coupling into the future*) を定義する．これは CFTP がシミュレーションの初期時刻を過去に遡って行ったのに対して，終了時刻を未来に進めていくシミュレーションである．すなわち CITF とは Ω の全ての状態について，共通の乱数列 $\boldsymbol{\lambda} = (\lambda[0], \lambda[1], \ldots, \lambda[T-1])$ を用いて時刻 0 から時刻 T までマルコフ連鎖を推移させることを言う．CITF の合流時間を $\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_0^T(x, \boldsymbol{\lambda})$ と定義し，合流時間 T^* は $T^* \stackrel{\text{def.}}{=} \min\{t > 0 \mid \exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_0^t(x, \boldsymbol{\lambda})\}$ と定義する．

この時，次の命題が成り立つ．

命題 2.16 ([94]) CFTP アルゴリズムの合流時間 T_* と CITF の合流時間 T^* は確率変数であり，

$$\forall t > 0,\ \Pr(T_* > t) = \Pr(T^* > t)$$

を満たす．

証明. 数列 $\boldsymbol{\lambda} \in [0,1)^t$ に対して $\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_{-t}^0(x, \boldsymbol{\lambda})$ が成り立つ時，同一の $\boldsymbol{\lambda}$ に対して $\exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi_0^t(x, \boldsymbol{\lambda})$ も成り立つ．確率変数列 $\boldsymbol{\Lambda} \in [0,1)^t$ は一様実数乱数の列とす

ると，

$$\begin{aligned}\Pr(T_* > t) &= 1 - \Pr(T_* \leq t) \\ &= 1 - \Pr(\boldsymbol{\Lambda} \in \{\boldsymbol{\lambda} \in [0,1)^t \mid \exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi^0_{-t}(x, \boldsymbol{\lambda})\}) \\ &= 1 - \Pr(\boldsymbol{\Lambda} \in \{\boldsymbol{\lambda} \in [0,1)^t \mid \exists y \in \Omega,\ \forall x \in \Omega,\ y = \Phi^t_0(x, \boldsymbol{\lambda})\}) \\ &= 1 - \Pr(T^* \leq t) \\ &= \Pr(T^* > t)\end{aligned}$$

となり，題意が示される． □

系 2.17 確率変数 T^* と T_* に対して $\mathrm{E}[T_*] = \mathrm{E}[T^*]$ が成り立つ． ■

次に，単調なマルコフ連鎖について，混交時間から合流時間の期待値が算定できることを示す．

定理 2.18 （[94]）状態空間 Ω をもち，更新関数 ϕ で定義される単調マルコフ連鎖 $\mathcal{M}$ において，定常分布を π，混交率を τ とする．単調マルコフ連鎖の定義を満たす半順序集合を $(\Omega, \succeq)$ と書き，最大元と最小元をそれぞれ $x_{\max}$ および $x_{\min}$ で表す．状態空間中に「高さ」h を定義する．状態 x の高さを $h(x)$ で表す．高さ h は $h(x_{\min}) = 0$ と任意の $x \succ y$ に対して $h(x) > h(y)$ を満たすとする．$D \stackrel{\text{def.}}{=} h(x_{\max}) = D$ とし，$d \stackrel{\text{def.}}{=} \min\{|h(x) - h(y)| \mid \forall (x,y) \in \Omega^2,\ h(x) \neq h(y)\}$ と表す．この時，CFTP の合流時間 T_* は $\mathrm{E}(T_*) \leq 2\tau(1 + \ln(2D/d))$ を満たす．

証明. いま，初期状態 $x_{\min}$ から k 回マルコフ連鎖を推移させた状態を X_0^k であらわし，初期状態 $x_{\max}$ から k 回マルコフ連鎖を推移させた状態を X_1^k であらわす．ここで，マルコフ連鎖の単調性より $X_1^k \succeq X_0^k$ が成り立つ．この時，

$$\begin{aligned}\Pr(T^* > k) &= \Pr(X_0^k \neq X_1^k) \\ &\leq \sum_{(x,y)\in\Omega^2} \Pr(X_0^k = x, X_1^k = y) \frac{h(y) - h(x)}{d} \\ &= \frac{1}{d} \sum_{x\in\Omega} \sum_{y\in\Omega} \left[\Pr(X_0^k = x, X_1^k = y) h(y) - \Pr(X_0^k = x, X_1^k = y) h(x)\right] \\ &= \frac{1}{d} \left(\sum_{x\in\Omega} \sum_{y\in\Omega} \Pr(X_0^k = x, X_1^k = y) h(y) - \sum_{x\in\Omega} \sum_{y\in\Omega} \Pr(X_0^k = x, X_1^k = y) h(x) \right) \\ &= \frac{1}{d} \left(\sum_{y\in\Omega} \Pr(X_1^k = y) h(y) - \sum_{x\in\Omega} \Pr(X_0^k = x) h(x) \right) \\ &= \frac{1}{d} \left(\sum_{x\in\Omega} P^k_{x_{\max}}(x)(x) h(x) - \sum_{x\in\Omega} P^k_{x_{\min}}(x) h(x) \right) \\ &= \frac{1}{d} \left(\sum_{x\in\Omega} (P^k_{x_{\max}}(x) - \pi(x)) h(x) - \sum_{x\in\Omega} (P^k_{x_{\min}}(x) - \pi(x)) h(x) \right)\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{1}{d}\left(\max_{A\subseteq\Omega}\left\{\sum_{x\in A}(P^k_{x_{\max}}(x)-\pi(x))h(x)\right\}+\max_{A\subseteq\Omega}\left\{\sum_{x\in A}(\pi(x)-P^k_{x_{\min}}(x))h(x)\right\}\right)\\
&\leq \frac{D}{d}\left(\max_{A\subseteq\Omega}\left\{\sum_{x\in A}(P^k_{x_{\max}}(x)-\pi(x))\right\}+\max_{A\subseteq\Omega}\left\{\sum_{x\in A}(\pi(x)-P^k_{x_{\min}}(x))\right\}\right)\\
&\leq \frac{2D}{d}\max_{x\in\Omega}\left\{\mathrm{d_{TV}}(\pi,P^k_x)\right\}
\end{aligned}
$$

が成り立つ．ここで $\varepsilon = \Pr(T^* > k)$ とすると，劣乗法性（*submultiplicativity* [94]）$\Pr(T^* > K_1+K_2) \leq \Pr(T^*>K_1)\Pr(T^*>K_2)$ より，$\Pr(T^*>\alpha k)\leq\varepsilon^\alpha$ を得る．ゆえに

$$
\begin{aligned}
\mathrm{E}(T^*) &= \textstyle\sum_0^{+\infty} t\Pr(T^*=t)\\
&\leq k+\textstyle\sum_k^{+\infty}(t-k)\Pr(T^*=t)\\
&\leq k+k\Pr(T^*>t)+\textstyle\sum_{2k}^{+\infty}(t-2k)\Pr(T^*=t)\\
&\leq k+k\Pr(T^*>t)+k\Pr(T^*>2t)+\textstyle\sum_{3k}^{+\infty}(t-3k)\Pr(T^*=t)\\
&\leq k+k\varepsilon+k\varepsilon^2+\cdots\\
&= \frac{k}{\Pr(T^*\leq k)}
\end{aligned}
$$

を得る．マルコフ連鎖 $\mathcal{M}$ の混交率を τ とし，$k=\tau(1+\ln\frac{2D}{d})$ とすると

$$
\Pr(T^*>k)\leq\frac{2D}{d}\max_x\left\{\mathrm{d_{TV}}(\pi,P^k_x)\right\}\leq\frac{2D}{d}\frac{1}{\frac{2D}{d}\mathrm{e}}\leq\frac{1}{\mathrm{e}}
$$

を得る．したがって

$$
\mathrm{E}(T_*)=\mathrm{E}(T^*)\leq\frac{k}{\Pr(T^*\leq k)}\leq\frac{k}{1-\frac{1}{\mathrm{e}}}<2k=2\tau\left(1+\ln\frac{2D}{d}\right)
$$

を得る． □

最後にマルコフ連鎖の合流時間の期待値から混交時間を算定する定理を述べる．この定理について述べている文献はこれまでに無いが，Propp と Wilson の結果から導くことができる．

定理 2.19 有限の状態空間 Ω をもち，推移確率 P をもつ単調なマルコフ連鎖 $\mathcal{M}$ の定常分布を π であらわす．マルコフ連鎖 $\mathcal{M}$ の合流時間の期待値 $\mathrm{E}[T_*]$ が与えられた時，マルコフ連鎖の混交時間は $0<\forall\varepsilon<1$, $\tau(\varepsilon)\leq\mathrm{E}[T_*]\varepsilon^{-1}$ で抑えられる．

証明. いま 2 つの確率変数 X^k と Y^k は，それぞれ初期状態 x から k 回マルコフ連鎖を推移させた状態とする．定理 2.4（カップリング定理）から，マルコフ連鎖が単調であることに注意すると，$\forall k\in\mathbb{Z}_+$,

$$
\mathrm{d_{TV}}(P^k_x,\pi)\leq\Pr[X^k\neq Y^k]\leq\Pr[X^k_0\neq X^k_1]=\Pr[T^*>k]=\Pr[T_*>k]
$$

が得られる．ただし，X_0^k および X_1^k は，それぞれ初期状態 $x_{\min}$ および $x_{\max}$ から CITF を行ったときの時刻 k における状態である．T_* の非負性から，

$$k\Pr[T_* > k] = k\sum_{t=k+1}^{\infty}\Pr[T_* = t] \le \sum_{t=1}^{\infty} t\Pr[T_* = t] = \mathrm{E}[T_*]$$

が成り立つので，

$$\mathrm{d_{TV}}(P_x^k, \pi) \le \Pr[T_* > k] \le \frac{\mathrm{E}[T_*]}{k}$$

である．今，$\tau(\varepsilon) \stackrel{\mathrm{def.}}{=} \mathrm{E}[T_*]\varepsilon^{-1}$ とすると，

$$\mathrm{d_{TV}}(P_x^{\tau(\varepsilon)}, \pi) \le \frac{E[T_*]}{E[T_*]\varepsilon^{-1}} = \varepsilon$$

となり題意を得る． □

2.4 乱択近似計算法と精度保証

本節では，モンテカルロ法に対する近似精度保証のための準備を行う．

2.4.1 Chernoff 限界

本小節では，モンテカルロ法に対する近似精度保証の議論を行う上で重要な Chernoff 限界を導入する．

定理 2.20 （Chernoff 限界（ [85] p. 64, p. 66））確率変数 $X_1, \ldots, X_n$ は確率 $\Pr(X_i) = p_i$ の独立の Poisson 試行とする．確率変数 $X = \sum_{i=1}^n X_i$ としその期待値を $\mu = \mathrm{E}[X]$ とする．この時，任意の正数 $\delta < 1$ に対して次の **Chernoff 限界**（*Chernoff bound*）

$$\Pr[X \ge (1+\delta)\mu] \le \mathrm{e}^{-\frac{1}{3}\mu\delta^2} \tag{2.4.1}$$

$$\Pr[X \le (1-\delta)\mu] \le \mathrm{e}^{-\frac{1}{2}\mu\delta^2} \tag{2.4.2}$$

が成り立つ． ■

本論文では，期待値の不偏性の議論をするために，定理 2.20 を変形して得られる次の補題を準備する．

補題 2.21 確率変数 X_i $(1 \le i \le M)$ は確率 p で $X_i = 1$，確率 $1-p$ で $X_i = 0$ の独立同一分布に従うものとする．確率変数 $U = \sum_i^M X_i$ とした時，任意の正数 $\lambda < 1$ に対して，もし $M \ge (4+2\sqrt{3})/p\lambda$ が成り立つならば

1. $\Pr\left[p - \frac{U+1}{M+1} \ge \lambda p\right] \le \mathrm{e}^{-\frac{1}{2}\lambda^2 Mp}$,

2. $\Pr\left[\frac{U+1}{M+1} - p \geq \lambda p\right] \leq \mathrm{e}^{-\frac{1}{4}\lambda^2 Mp}$,

3. $\Pr\left[\left|\frac{U+1}{M+1} - p\right| \geq \lambda p\right] \leq 2\mathrm{e}^{-\frac{1}{4}\lambda^2 Mp}$,

が成り立つ．

証明. 1. いま $\frac{U+1}{M+1} > \frac{U}{M}$ が成り立つので，$p - \frac{U+1}{M+1} < p - \frac{U}{M}$ が成り立つ．従って，もし $p - \frac{U+1}{M+1} \geq \lambda p$ が成り立つならば $p - \frac{U}{M} \geq \lambda p$ が成り立つ．したがって

$$\Pr\left[p - \frac{U+1}{M+1} \geq \lambda p\right] \leq \Pr\left[p - \frac{U}{M} \geq \lambda p\right] \leq \mathrm{e}^{-\frac{1}{2}\lambda^2 Mp},$$

を得る．ただし，最後の不等式は Chernoff 限界（式 2.4.2）から得られる．
2. 条件 $M \geq (4+2\sqrt{3})/p\lambda$ が成り立つとき，Chernoff 限界（式 2.4.1）から

$$\begin{aligned}\Pr\left[\frac{U+1}{M+1} - p \geq \lambda p\right] &\leq \Pr\left[\frac{U}{M} + \frac{1}{M} - p \geq \lambda p\right] = \Pr\left[\frac{U}{M} - p \geq \left(\lambda - \frac{1}{Mp}\right)p\right] \\ &\leq \mathrm{e}^{-\frac{1}{3}\left(\lambda - \frac{1}{Mp}\right)^2 Mp} = \mathrm{e}^{-\frac{1}{3}\left(1 - \frac{1}{Mp\lambda}\right)^2 \lambda^2 Mp} \leq \mathrm{e}^{-\frac{1}{4}\lambda^2 Mp}\end{aligned}$$

が得られる．
3. 以上の 1，2 から

$$\Pr\left[\left|\frac{U+1}{M+1} - p\right| \geq \lambda p\right] \leq 2\mathrm{e}^{-\frac{1}{4}\lambda^2 Mp}$$

が得られる □

2.4.2 全多項式時間乱択近似スキーム

本小節では乱択近似計算法とその計算量について議論する．任意に与えられた ε $(0 < \varepsilon < 1)$ および，任意に与えられた δ $(0 < \delta < 1)$ に対して，乱択近似計算法 $\mathcal{S}$ の出力 $Z \in \mathbb{Z}_{++}$ が，真の値 $A \in \mathbb{Z}_{++}$ に対して

$$\Pr\left[\frac{|Z - A|}{A} \leq \varepsilon\right] \geq 1 - \delta$$

を満たすとき，乱択近似計算法 $\mathcal{S}$ を乱択近似スキーム（*RAS*: *Randomized Approximation Scheme*）と呼ぶ．特に，乱択近似計算法 $\mathcal{S}$ が問題の入力サイズ，ε^{-1} および $\ln \delta^{-1}$ の多項式時間で 必ず 停止する時，乱択近似計算法 $\mathcal{S}$ を全多項式乱択近似スキーム（*FPRAS*: *Fully Polynomial-time Randomized Approximation Scheme*）と呼ぶ．

FPRAS の設計において，乱択近似計算法は多項式時間で必ず停止する必要がある．一方で，ラスベガス型をはじめとする確率的アルゴリズムでは，しばしば計算時間の多項式性は期待計算時間

としてのみ保証される．すなわち，乱数に依存して，(最悪の場合) 無限の計算時間が必要となることもある．本小節の残りでは，期待多項式時間の乱択近似計算法を多項式時間停止の乱択近似計算法に変換する方法について議論する．

いま，任意に与えられた正数 ε $(0 < \varepsilon < 1)$ に対して，乱択近似計算法 $\mathcal{S}$ の出力する近似解 Z は真の値 A に対して

$$\Pr\left[\frac{|Z - A|}{A} \leq \varepsilon\right] \geq \frac{1}{8}$$

を満たすとする．また，乱択計算法 $\mathcal{S}$ の計算時間 $T \in \mathbb{Z}_{++}$ は確率的で，確率変数 T の期待値を μ とする．このとき，

$$8\mu \cdot \Pr[T \geq 8\mu] \;=\; 8\mu \cdot \sum_{t=\lceil 8\mu \rceil}^{+\infty} \Pr[T = t] \;\leq\; \sum_{t=1}^{+\infty} t \cdot \Pr[T = t] \;=\; \mathrm{E}[T] \;= \mu$$

より，

$$\Pr[T \geq 8\mu] \leq \frac{\mu}{8\mu} = \frac{1}{8}$$

を得る．すなわち，乱択近似計算法 $\mathcal{S}$ が期待計算時間の 8 倍以内の時間で停止する確率は $7/8$ 以上である．

上述の乱択近似計算法 $\mathcal{S}$ に対して，乱択近似計算法 $\mathcal{S}'$ は，$\mathcal{S}$ が時間 8μ 以内に停止したら $\mathcal{S}$ の出力 Z を出力し，そうでなければ値 0 を出力するものとする．したがって，$\mathcal{S}'$ の出力 Z' は

$$\Pr\left[\frac{|Z' - A|}{A} \leq \varepsilon\right] \;\geq\; 1 - \frac{1}{8} - \frac{1}{8} \;=\; \frac{3}{4}$$

を満たす．

次に，乱択近似計算法 $\mathcal{S}'$ に対して，S' の反復により所望の解の得られる確率を高める方法について議論する．乱択近似計算法 $\mathcal{S}''$ は $\mathcal{S}'$ を独立に M 回行ってその中央値 (median) を出力するものとする．このとき，$\mathcal{S}''$ の計算時間は必ず $8\mu \cdot M$ より小さくなることに注意されたい．最後に，次の補題を示す．

命題 2.22 任意に与えられる δ $(0 < \delta < 1)$ に対して，$M = -12 \ln \delta$ とすると，乱択近似計算法 $\mathcal{S}''$ の出力する近似解 Z'' は

$$\Pr\left[\frac{|Z'' - A|}{A} \leq \varepsilon\right] \geq 1 - \delta$$

を満たす．

証明. 乱択近似計算法 $\mathcal{S}'$ を独立に M 回行って得られた値の中央値 Z'' が $\frac{|Z'' - A|}{A} > \varepsilon$ となるのは，$\mathcal{S}'$ を独立に M 回行った時に

事象 I. $\mathcal{S}'$ の出力 Z' が $\frac{Z'-A}{A} > \varepsilon$ となる事象が $M/2$ 回以上起こる，または

事象 II. $\mathcal{S}'$ の出力 Z' が $\frac{A-Z'}{A} > \varepsilon$ となる事象が $M/2$ 回以上起こる，

のいずれかが起こった場合である．事象 I または II の起こる確率は，$\frac{|Z'-A|}{A} > 1-\varepsilon$ が $M/2$ 回以上起こる確率より小さい．議論の簡便のため，次の命題を考える．

命題 **2.23** 確率 $3/4$ で表，$1/4$ で裏の出るコインを M 回トスしたときに裏が $M/2$ 回以上出る確率は高々 $\mathrm{e}^{-M/12}$ である．

証明. M 回コイントスして表の出た回数を X とする．明らかに $\mathrm{E}[X] = M/4$ である．$X \geq M/2$ となる確率を計算すると，Chernoff 限界より，

$$\Pr\left[X \geq \frac{M}{2}\right] = \Pr\left[X \geq (1+1)\frac{M}{4}\right] \leq \mathrm{e}^{-\frac{1}{3}\frac{M}{4}1^2} = \mathrm{e}^{-\frac{M}{12}}.$$

■

以上の議論から，乱択近似計算法 $\mathcal{S}''$ の出力 Z'' に対して

$$\Pr\left[\frac{|Z''-A|}{A} \leq \varepsilon\right] \;\geq\; 1-\mathrm{e}^{-\frac{M}{12}} \;=\; 1-\mathrm{e}^{-\frac{-12\ln\delta}{12}} \;=\; 1-\delta$$

を得る． □

3　対数分離凹分布からのサンプリング

3.1　対数分離凹分布

本論文では $n-1$ 次元単体中の非負整数格子点からなる集合

$$\Delta(K) \stackrel{\text{def.}}{=} \left\{ \boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{Z}_+^n \mid \textstyle\sum_{i=1}^n x_i = K \right\}$$

を扱う．簡明のため，混乱の恐れの無い場合に限り $\Delta(K)$ を Δ と略記する．関数ベクトル $\boldsymbol{f} = (f_1, f_2, \ldots, f_n)$ は，各 $i \in \{1, \ldots, n\}$ について $f_i : \mathbb{Z}_+ \to \mathbb{R}_{++}$ を満たすとする．このとき，Δ 上に定義される確率関数 π を

$$\pi(\boldsymbol{x}) = \frac{1}{C} \prod_{i=1}^{n} f_i(x_i) \quad (\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \Delta)\,, \tag{3.1.1}$$

とし，分布 π に従うランダムサンプリング法を提案する．ただし $C \stackrel{\text{def.}}{=} \sum_{\boldsymbol{x} \in \Delta} \prod_{i=1}^{n} f_i(x_i)$ は正規化定数である．一変数関数 f_i が

$$\ln f_i(x+1) \geq \frac{1}{2} \left(\ln f_i(x) + \ln f_i(x+2) \right) \qquad (\forall x \in \mathbb{Z}_+), \tag{3.1.2}$$

を満たす時，対数凹関数 (*logarithmic concave function*) であると定義する．式 (3.1.1) の確率関数 π において，各関数 f_i が対数凹関数のとき，確率関数 π は対数分離凹 (*logarithmic separable concave*) であると言う．

3.2　近似サンプリング法

以下では状態空間 Δ を持つマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ を提案する．現在の状態 $X \in \Delta$ から次の状態 X' へのマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の推移は以下のように定義される．まず，相異なる添え字の対 $\{j_1, j_2\} \in \binom{\{1,\ldots,n\}}{2}$ を一様ランダムに選ぶ．次に，$k = X_{j_1} + X_{j_2}$ とし，$\{0, \ldots, k\}$ 上の確率変数 L を確率

$$\Pr[L = l] \;=\; \frac{f_{j_1}(l) f_{j_2}(k-l)}{\sum_{s=0}^{k} f_{j_1}(s) f_{j_2}(k-s)} \qquad (l \in \{0, 1, \ldots, k\})$$

で発生させる．最後に，次の時刻の状態 X' を

$$X'_i = \begin{cases} L & (i = j_1), \\ k - L & (i = j_2), \\ X_i & (\text{それ以外}), \end{cases}$$

と定める．このとき，明らかに $X' \in \Delta$ が成り立つ．いま $f_i(x)$ は正の関数なので，有限マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ は既約で非周期的であり，すなわちエルゴード性を満たす．したがって $\mathcal{M}_{\mathrm{A}}$ は唯一の定常分布をもつ．また π に関する詳細均衡式を容易に示すことができ，唯一の定常分布は，式 (3.1.1) で定義される分布 π となる．

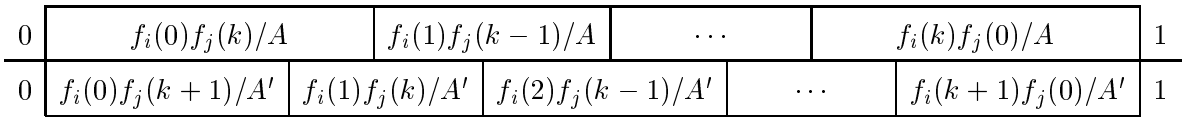


図 6　添え字対 (i,j) および非負整数 k に対する交互不等式の図．図中で $A \stackrel{\text{def.}}{=} \sum_{s=0}^{k} f_i(s)f_j(k-s)$ および $A' \stackrel{\text{def.}}{=} \sum_{s=0}^{k+1} f_i(s)f_j(k+1-s)$ は正規化定数である．

任意の $k \in \{0,1,\ldots,K\}$ および任意の相異なる添え字の対 (i,j) に対して，累積分布関数 $g_{ij}^k : \mathbb{Z} \to \mathbb{R}_+$ を

$$g_{ij}^k(l) \stackrel{\text{def.}}{=} \frac{\sum_{s=0}^{l} f_i(s)f_j(k-s)}{\sum_{s=0}^{k} f_i(s)f_j(k-s)} \qquad (l \in \{0,1,\ldots,k\}) \tag{3.2.1}$$

と定義する．また，便宜上 $g_{ij}^k(-1) \stackrel{\text{def.}}{=} 0$ と定義する．この関数 g_{ij}^k を用いて，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ を以下のようにシミュレートすることができる．まず，相異なる添え字の対 $\{i,j\}$ を確率 $2/(n(n-1))$ で一様ランダムに選ぶ．次に，$k = X_i + X_j$ と置き，一様実数乱数 $\Lambda \in [0,1)$ を生成し，$g_{ij}^k(l-1) \le \Lambda \le g_{ij}^k(l)$ を満足する $l \in \{0,\ldots,k\}$ を用いて次の時刻の状態を

$$X_i' = \begin{cases} l & (i = j_1), \\ k-l & (i = j_2), \\ X_i & (\text{それ以外}), \end{cases}$$

とする．

本小節の残りではマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の混交時間を算定する．ある $k \in \{0,1,\ldots,K-1\}$ および関数の対 f_i, f_j に対する交互不等式 $(alternating\ inequalities)$ を，

> 任意の $l \in \{0,1,\ldots,k\}$ について
>
> $$\frac{\sum_{s=0}^{l} f_i(s)f_j(k+1-s)}{\sum_{s=0}^{k+1} f_i(s)f_j(k+1-s)} \le \frac{\sum_{s=0}^{l} f_i(s)f_j(k-s)}{\sum_{s=0}^{k} f_i(s)f_j(k-s)} \le \frac{\sum_{s=0}^{l+1} f_i(s)f_j(k+1-s)}{\sum_{s=0}^{k+1} f_i(s)f_j(k+1-s)} \tag{3.2.2}$$

と定義する (図 6 参照). 不等式 (3.2.2) は関数 g_{ij}^k を用いて

$$g_{ij}^{k+1}(l) \le g_{ij}^k(l) \le g_{ij}^{k+1}(l+1) \quad (l \in \{0,\ldots,k\}) \tag{3.2.3}$$

と等価に表現できる．

次の補題は，任意の対数分離凹分布 π に対して，交互不等式 (3.2.2) が任意の添え字対について成り立つことを意味する．

補題 3.1 関数 f_i および f_j が共に対数凹関数の時，任意の $k \in \mathbb{Z}_+$ に対して交互不等式 (3.2.2) が成り立つ．

証明. 明らかに $k = 0$ の時は成り立つので，以下 $k \in \mathbb{Z}_{++}$ について考える．

まず，各 $k \in \{1, \ldots, K-1\}$ について，交互不等式 (3.2.2) が成り立つことと，

任意の $l \in \{0, 1, \ldots, k-1\}$ について

$$\left(\sum_{s=0}^{l} f_i(s) f_j(k+1-s)\right)\left(\sum_{s'=l+1}^{k} f_i(s') f_j(k-s')\right)$$
$$\leq \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s'=l+1}^{k+1} f_i(s') f_j(k+1-s')\right), \text{ および} \quad (3.2.4)$$

$$\left(\sum_{s=0}^{l} f_i(k+1-s) f_j(s)\right)\left(\sum_{s'=l+1}^{k} f_i(k-s') f_j(s')\right)$$
$$\leq \left(\sum_{s=0}^{l} f_i(k-s) f_j(s)\right)\left(\sum_{s'=l+1}^{k+1} f_i(k+1-s') f_j(s')\right), \quad (3.2.5)$$

が成り立つことが等価であることを示す．各 $l \in \{0, 1, \ldots, k-1\}$ について，不等式 (3.2.2) の最初の不等式を以下のように式変形すると式 (3.2.4) が得られる．

$$\frac{\sum_{s=0}^{l} f_i(s) f_j(k+1-s)}{\sum_{s=0}^{k+1} f_i(s) f_j(k+1-s)} \leq \frac{\sum_{s=0}^{l} f_i(s) f_j(k-s)}{\sum_{s=0}^{k} f_i(s) f_j(k-s)}$$
$$\Leftrightarrow \left(\sum_{s=0}^{l} f_i(s) f_j(k+1-s)\right)\left(\sum_{s=0}^{k} f_i(s) f_j(k-s)\right)$$
$$\leq \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s=0}^{k+1} f_i(s) f_j(k+1-s)\right)$$
$$\Leftrightarrow \left(\sum_{s=0}^{l} f_i(s) f_j(k+1-s)\right)\left(\sum_{s=0}^{l} f_i(s) f_j(k-s) + \sum_{s=l+1}^{k} f_i(s) f_j(k-s)\right)$$
$$\leq \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s=0}^{l} f_i(s) f_j(k+1-s) + \sum_{s=l+1}^{k+1} f_i(s) f_j(k+1-s)\right)$$
$$\Leftrightarrow \left(\sum_{s=0}^{l} f_i(s) f_j(k+1-s)\right)\left(\sum_{s=l+1}^{k} f_i(s) f_j(k-s)\right)$$
$$\leq \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s=l+1}^{k+1} f_i(s) f_j(k+1-s)\right).$$

同様に，各 $l \in \{0, 1, \ldots, k-1\}$ について，不等式 (3.2.2) の後ろの不等式を以下のように式変形すると式 (3.2.5) が得られる．

$$\frac{\sum_{s=0}^{l} f_i(s) f_j(k-s)}{\sum_{s=0}^{k} f_i(s) f_j(k-s)} \leq \frac{\sum_{s=0}^{l+1} f_i(s) f_j(k+1-s)}{\sum_{s=0}^{k+1} f_i(s) f_j(k+1-s)}$$
$$\Leftrightarrow \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s=0}^{k+1} f_i(s) f_j(k+1-s)\right)$$
$$\leq \left(\sum_{s=0}^{l+1} f_i(s) f_j(k+1-s)\right)\left(\sum_{s=0}^{k} f_i(s) f_j(k-s)\right)$$
$$\Leftrightarrow \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s=0}^{l+1} f_i(s) f_j(k+1-s) + \sum_{s=l+2}^{k+1} f_i(s) f_j(k+1-s)\right)$$
$$\leq \left(\sum_{s=0}^{l+1} f_i(s) f_j(k+1-s)\right)\left(\sum_{s=0}^{l} f_i(s) f_j(k-s) + \sum_{s=l+1}^{k} f_i(s) f_j(k-s)\right)$$
$$\Leftrightarrow \left(\sum_{s=0}^{l} f_i(s) f_j(k-s)\right)\left(\sum_{s=l+2}^{k+1} f_i(s) f_j(k+1-s)\right)$$
$$\leq \left(\sum_{s=0}^{l+1} f_i(s) f_j(k+1-s)\right)\left(\sum_{s=l+1}^{k} f_i(s) f_j(k-s)\right)$$
$$\Leftrightarrow \left(\sum_{s'=k-l}^{k} f_i(k-s') f_j(s')\right)\left(\sum_{s''=0}^{k-l-1} f_i(k+1-s'') f_j(s'')\right)$$
$$\leq \left(\sum_{s''=k-l}^{k+1} f_i(k+1-s'') f_j(s'')\right)\left(\sum_{s'=0}^{k-l-1} f_i(k-s') f_j(s')\right)$$

$$\Leftrightarrow \left(\sum_{s'=l'+1}^{k} f_i(k-s')f_j(s')\right)\left(\sum_{s''=0}^{l'} f_i(k+1-s'')f_j(s'')\right)$$
$$\leq \left(\sum_{s''=l'+1}^{k+1} f_i(k+1-s'')f_j(s'')\right)\left(\sum_{s'=0}^{l'} f_i(k-s')f_j(s')\right),$$

ただし，$l' = k-l-1 \in \{0,1,\ldots,k-1\}$ である．

次に，関数 f_j が対数凹関数のとき不等式 $(3.2.4)$ が成り立つことを示す．不等式 $(3.2.4)$ の両辺を展開して考えると，任意の $s, s' \in \{0,1,\ldots,k\}$ について，$0 \leq s < s' \leq k$ を満足するとき

$$f_i(s)f_j(k+1-s)f_i(s')f_j(k-s') \leq f_i(s)f_j(k-s)f_i(s')f_j(k+1-s') \qquad (3.2.6)$$

が成り立つことを示せば十分である．補題の仮定から関数 f_j は対数凹なので，$(k-s') < (k-s'+1) \leq (k-s) < (k-s+1)$ が成り立つことを考慮すると，不等式

$$\ln f_j(k-s') + \ln f_j(k-s+1) \leq \ln f_j(k-s'+1) + \ln f_j(k-s)$$

が成り立つ．すなわち

$$f_j(k-s')f_j(k-s+1) \leq f_j(k-s'+1)f_j(k-s)$$

が成り立つ．関数 f_i は任意の $z \in \mathbb{Z}_+$ に対して $f_i(z) > 0$ より，任意の $s, s' \in \{0,1,\ldots,k\}$ について，$0 \leq s < s' \leq k$ を満足するとき不等式 $(3.2.6)$ が成り立つ．

同様に添え字 i と j を交換して議論することで関数 f_i が対数凹のとき不等式 $(3.2.5)$ が得られる． □

定理 3.2 関数ベクトル $\boldsymbol{f} = (f_1, f_2, \ldots, f_n)$ が任意の $k \in \{0,1,\ldots,K\}$ および任意の対 f_i, f_j に関して交互不等式 $(3.2.2)$ を満たすとき，任意の ε $(0 < \varepsilon < 1)$ に対して $\mathcal{M}_{\mathrm{A}}$ の混交時間 $\tau(\varepsilon)$ は

$$\tau(\varepsilon) \leq \frac{n(n-1)}{2}\ln(K\varepsilon^{-1})$$

を満たす．

証明. 単純無向グラフ $\mathcal{G} = (\Delta, \mathcal{E})$ を以下のように定義する．グラフ $\mathcal{G}$ は頂点集合 Δ を持ち，以下に定義される枝集合 $\mathcal{E}$ を持つ．頂点対 $\{x,y\}$ がグラフ $\mathcal{G}$ の枝であることの必要十分条件は $(1/2)\sum_{i=1}^{n}|x_i - y_i| = 1$ を満足することである．明らかにグラフ $\mathcal{G}$ は連結である．枝 $e \in \mathcal{E}$ の長さ $\ell_{\mathrm{A}}(e)$ を $\ell_{\mathrm{A}}(e) \stackrel{\mathrm{def.}}{=} 1$ と定義する．また，任意の頂点対 $\{x,y\} \in \binom{\Delta}{2}$ 間の距離 $d_{\mathrm{A}}(x,y)(= d_{\mathrm{A}}(y,x))$ をグラフ $\mathcal{G}$ 上での x と y の間の ℓ_{A} に関する最短パスの長さと定義する．明らかに $\mathcal{G}$ の直径 $\max_{x,y\in\Delta}\{d_{\mathrm{A}}(x,y)\}$ は K 以下である．

任意の枝 $\{X,Y\} \in \mathcal{E}$ に対するジョイントプロセス $(X,Y) \mapsto (X',Y')$ を以下のように定義する．まず，相異なる添え字対 $\{i_1,i_2\}$ を一様ランダムに選択する．次に，$k_X = X_{i_1} + X_{i_2}$ および $k_Y = Y_{i_1} + Y_{i_2}$ とし，一様実数乱数 $\Lambda \in [0,1)$ を生成し，$l_X \in \{0,1,\ldots,k_X\}$ および $l_Y \in \{0,1,\ldots,k_Y\}$ をそれぞれ

$$g_{i_1 i_2}^{k_X}(l_X - 1) \leq \Lambda < g_{i_1 i_2}^{k_X}(l_X), \qquad g_{i_1 i_2}^{k_Y}(l_Y - 1) \leq \Lambda < g_{i_1 i_2}^{k_Y}(l_Y),$$

を満足するものとする．最後に

$$X'_i = \begin{cases} l_X & (i = i_1), \\ k_X - l_X & (i = i_2), \\ X_i & (\text{それ以外}), \end{cases} \qquad Y'_i = \begin{cases} l_Y & (i = i_1), \\ k_Y - l_Y & (i = i_2), \\ X_i & (\text{それ以外}), \end{cases}$$

とする．

任意の枝 $\{X, Y\} \in \mathcal{E}$ に対して

$$\mathrm{E}[d_{\mathrm{A}}(X', Y')] \le \beta d_{\mathrm{A}}(X, Y), \quad \text{かつ} \quad \beta = 1 - \frac{2}{n(n-1)},$$

が成り立つことを示す．枝 $\{X, Y\} \in \mathcal{E}$ は添え字 $j \in \{j_1, j_2\}$ に対して $|X_j - Y_j| = 1$ が成り立ち，添え字 $j \notin \{j_1, j_2\}$ に対しては $|X_j - X_j| = 0$ が成り立つものとする．

場合 1: 添え字対 j_1 および j_2 がいずれも選ばれない，すなわち $\{i_1, i_2\} \cap \{j_1, j_2\} = \emptyset$ の場合．この時 $k = X_{i_1} + X_{i_2}$ と置くと，$Y_{i_1} + Y_{i_2} = k$ より，任意の $l \in \{0, \ldots, k\}$ に対して $\Pr[X'_{i_1} = l] = \Pr[Y'_{i_1} = l]$ が成り立つ．したがって $X'_{i_1} = Y'_{i_1}$ および $X'_{i_2} = Y'_{i_2}$ が成り立つ．ゆえに $d_{\mathrm{A}}(X', Y') = d_{\mathrm{A}}(X, Y)$ を得る．

場合 2: 添え字対 j_1 および j_2 が共に選ばれる，すなわち $\{i_1, i_2\} = \{j_1, j_2\}$ の場合．場合 1 と同様に $X'_{i_1} = Y'_{i_1}$ および $X'_{i_2} = Y'_{i_2}$ が成り立つ．ゆえに $d_{\mathrm{A}}(X', Y') = 0$ を得る．

場合 3: 添え字対 j_1 および j_2 のうちいずれか一方のみが選ばれる, すなわち $|\{i_1, i_2\} \cap \{j_1, j_2\}| = 1$ の場合．一般性を失うことなく $i_1 = j_1$ かつ $X_{i_1} + 1 = Y_{i_1}$ とする．このとき $k = X_{i_1} + X_{i_2}$ かつ $Y_{i_1} + Y_{i_2} = k + 1$ となる．乱数 $\Lambda \in [0, 1)$ が与えられた時のジョイントプロセスを考える．整数 $l \in \{0, 1, \ldots, k\}$ は $g^k_{i_1 i_2}(l-1) \le \Lambda < g^k_{i_1 i_2}(l)$ を満足するものとすると，交互不等式 (3.2.3) より $g^{k+1}_{i_1 i_2}(l-1) \le \Lambda < g^{k+1}_{i_1 i_2}(l+1)$ が成り立つ．したがって $X'_{i_1} = l$ とすると，ジョイントプロセスの定義から Y'_{i_1} は $\{l, l+1\}$ の要素となる．すなわち $[X'_{i_1} = Y'_{i_1}$ かつ $X'_{i_2} + 1 = Y'_{i_2}]$ または $[X'_{i_1} + 1 = Y'_{i_1}$ かつ $X'_{i_2} = Y'_{i_2}]$ のいずれかとなる．ゆえに $d_{\mathrm{A}}(X', Y') = d_{\mathrm{A}}(X, Y)$ が成り立つ．

場合 2 の起こる確率が $2/(n(n-1))$ であることから

$$\mathrm{E}[d_{\mathrm{A}}(X', Y')] \le \left(1 - \frac{2}{n(n-1)}\right) d_{\mathrm{A}}(X, Y)$$

が成り立つ．グラフ $\mathcal{G}$ の直径は K 以下なので，定理 2.5 (経路カップリング) より混交時間 $\tau(\varepsilon)$ は

$$\tau(\varepsilon) \le \frac{n(n-1)}{2} \ln(K\varepsilon^{-1})$$

を満たす． □

系 3.3 式 (3.1.1) で定義される分布 π が対数分離凹の時，任意の ε $(0 < \varepsilon < 1)$ に対して $\mathcal{M}_{\mathrm{A}}$ の混交時間 $\tau(\varepsilon)$ は

$$\tau(\varepsilon) \le \frac{n(n-1)}{2} \ln(K\varepsilon^{-1})$$

を満たす．証明略． ■

3.3 完璧サンプリング法

3.3.1 単調マルコフ連鎖

本節では新たに別のマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ を提案する．マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の推移規則は以下に定義される更新関数 $\phi_{\mathrm{P}} : \Delta \times [1, n) \to \Delta$ によって定められる．現在の状態 $X \in \Delta$ に対して，次の時刻の状態 $X' = \phi_{\mathrm{P}}(X, \lambda) \in \Delta$ は一様実数乱数 $\lambda \in [1, n)$ に対して

$$X'_i = \begin{cases} l & (i = \lfloor \lambda \rfloor), \\ k - l & (i = \lfloor \lambda \rfloor + 1), \\ X_i & (\text{それ以外}), \end{cases}$$

ただし $k = X_{\lfloor \lambda \rfloor} + X_{\lfloor \lambda \rfloor + 1}$ であり，$l \in \{0, 1, \ldots, k\}$ は

$$g^k_{\lfloor \lambda \rfloor (\lfloor \lambda \rfloor + 1)}(l - 1) < \lambda - \lfloor \lambda \rfloor \leq g^k_{\lfloor \lambda \rfloor (\lfloor \lambda \rfloor + 1)}(l).$$

を満足するものとする．マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ は $\mathcal{M}_{\mathrm{A}}$ における推移のうち，選ばれる添え字対を連続するもののみに制限した推移と見ることができる．マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ はエルゴード的であり，唯一の定常分布として式 (3.1.1) で定義される分布 π を持つ．

以下 ,マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の単調性を示す．まず ,集合 Δ 上に半順序 "$\succeq$" を導入する．任意の状態 $x \in \Delta$ に対して累積和ベクトル $(cumulative\ sum\ vector)$ $h_x = (h_x(0), h_x(1), \ldots, h_x(n)) \in \mathbb{Z}_+^{n+1}$ を

$$h_x(i) \stackrel{\text{def.}}{=} \begin{cases} 0 & (i = 0), \\ \sum_{j=1}^{i} x_j & (i \in \{1, 2, \ldots, n\}) \end{cases}$$

と定義する．任意の状態対 $x, y \in \Delta$ に対して，$x \succeq y$ を $h_x \geq h_y$ であることと定義する．次に 2 つの特別な状態 $x_{\max}, x_{\min} \in \Delta$ を $x_{\max} \stackrel{\text{def.}}{=} (K, 0, \cdots, 0)$ および $x_{\min} \stackrel{\text{def.}}{=} (0, \ldots, 0, K)$ と定義する．明らかに，任意の $x \in \Delta$ に対して $x_{\max} \succeq x \succeq x_{\min}$ が成り立つ．

定理 3.4 関数ベクトル $\boldsymbol{f} = (f_1, \ldots, f_n)$ が任意の $k \in \{0, 1, \ldots, K\}$ および任意の連続する対 f_i, f_{i+1} に関して交互不等式 (3.2.2) を満足する時，マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ は半順序集合 $(\Delta, \succeq)$ 上で単調である．すなわち $[\forall \lambda \in [1, n),\ \forall X, \forall Y \in \Delta,\ \ X \succeq Y \Rightarrow \phi_{\mathrm{P}}(X, \lambda) \succeq \phi_{\mathrm{P}}(Y, \lambda)]$ と $[\forall x \in \Delta,\ x_{\max} \succeq x \succeq x_{\min}]$ が成り立つ．

証明. 状態対 $X \in \Delta$ および $Y \in \Delta$ に対して，ある添え字 $j \in \{1, \ldots, n\}$ が存在し，

$$X_i - Y_i = \begin{cases} +1 & (i = j), \\ -1 & (i = j + 1), \\ 0 & (\text{それ以外}), \end{cases}$$

を満たす時，X が Y を (j で) 被覆する $(cover)$ と言い，$X \cdot\!\succ Y$ (または $X \cdot\!\succ_j Y$) と書く．任意の状態対 X, Y が $X \succeq Y$ を満足するとき，適当な長さの状態の列 $Z_1, Z_2, \ldots, Z_r$ が存在して $X = Z_1 \cdot\!\succ Z_2 \cdot\!\succ \cdots \cdot\!\succ Z_r = Y$ が成り立つ．したがって状態対 $X, Y \in \Delta$ が $X \cdot\!\succ Y$ を

満たす時，任意の $\lambda \in [1,n)$ に対して $\phi_{\mathrm{P}}(X,\lambda) \succeq \phi_{\mathrm{P}}(Y,\lambda)$ が成り立つことを示すことで，$\phi_{\mathrm{P}}(X,\lambda) = \phi_{\mathrm{P}}(Z_1,\lambda) \succeq \phi_{\mathrm{P}}(Z_2,\lambda) \succeq \cdots \succeq \phi_{\mathrm{P}}(Z_r,\lambda) = \phi_{\mathrm{P}}(Y,\lambda)$ を示すことができる．

状態対 $X, Y \in \Delta$ が $X \cdot \succ Y$ を満たす時，任意の $\lambda \in [1,n)$ に対して $\phi_{\mathrm{P}}(X,\lambda) \succeq \phi_{\mathrm{P}}(Y,\lambda)$ が成り立つことを示す．簡単のため $\phi_{\mathrm{P}}(X,\lambda)$ を X' で表し，$\phi_{\mathrm{P}}(Y,\lambda)$ を Y' で表す．任意の添え字 $i \neq \lfloor\lambda\rfloor$ について $h_X(i) = h_{X'}(i)$ および $h_Y(i) = h_{Y'}(i)$ が成り立つので，$X \succeq Y$ より，$h_{X'}(i) - h_{Y'}(i) = h_X(i) - h_Y(i) \geq 0$ を得る．以下，$h_{X'}(\lfloor\lambda\rfloor) \geq h_{Y'}(\lfloor\lambda\rfloor)$ を示す．

場合 1: まず，$\lfloor\lambda\rfloor \neq j-1$ かつ $\lfloor\lambda\rfloor \neq j+1$ の場合を考える．この時，$k = X_{\lfloor\lambda\rfloor} + X_{\lfloor\lambda\rfloor+1}$ と置くと，$Y_{\lfloor\lambda\rfloor} + Y_{\lfloor\lambda\rfloor+1} = k$ である．したがって $l \in \{0,\ldots,k\}$ が

$$g^k_{\lfloor\lambda\rfloor(\lfloor\lambda\rfloor+1)}(l-1) \leq \lambda - \lfloor\lambda\rfloor < g^k_{\lfloor\lambda\rfloor(\lfloor\lambda\rfloor+1)}(l),$$

を満たす時，$X'_{\lfloor\lambda\rfloor} = Y'_{\lfloor\lambda\rfloor} = l$ が成り立つ．ゆえに $h_{X'}(\lfloor\lambda\rfloor) = h_{Y'}(\lfloor\lambda\rfloor)$ が成り立つ．

場合 2: 次に，$\lfloor\lambda\rfloor = j-1$ の場合を考える．この時，$k+1 = X_{j-1} + X_j$ と置くと，$X \cdot \succ_j Y$ より $Y_{j-1} + Y_j = k$ である．累積和ベクトルの定義より，

$$\begin{aligned} h_{X'}(\lfloor\lambda\rfloor) - h_{Y'}(\lfloor\lambda\rfloor) &= h_{X'}(j-1) - h_{Y'}(j-1) \\ &= h_{X'}(j-2) + X'_{j-1} - h_{Y'}(j-2) - Y'_{j-1} \\ &= h_X(j-2) + X'_{j-1} - h_Y(j-2) - Y'_{j-1} \\ &= X'_{j-1} - Y'_{j-1} \end{aligned}$$

が成り立つ．したがって，$X'_{j-1} \geq Y'_{j-1}$ を示せば十分である．いま $l \in \{0,1,\ldots,k\}$ が λ に対して $g^k_{(j-1)j}(l-1) \leq \lambda - \lfloor\lambda\rfloor < g^k_{(j-1)j}(l)$ を満たすとする．この時，交互不等式 (3.2.3) $g^{k+1}_{(j-1)j}(l-1) \leq g^k_{(j-1)j}(l-1) < g^{k+1}_{(j-1)j}(l) \leq g^{k+1}_{(j-1)j}(l+1)$ より，$g^{k+1}_{(j-1)j}(l-1) \leq \lambda - \lfloor\lambda\rfloor < g^{k+1}_{(j-1)j}(l+1)$ が成り立つ．したがって，$Y'_{j-1} = l$ の時，$X'_{j-1} = l$ または $l+1$ が成り立つ．言い換えると，

$$\begin{pmatrix} X'_{j-1} \\ Y'_{j-1} \end{pmatrix} \in \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \ldots, \begin{pmatrix} k \\ k \end{pmatrix}, \begin{pmatrix} k+1 \\ k \end{pmatrix} \right\}$$

が成り立つ．したがって $X'_{j-1} \geq Y'_{j-1}$ を得る．ゆえに $h_{X'}(\lfloor\lambda\rfloor) \geq h_{Y'}(\lfloor\lambda\rfloor)$ を得る．

場合 3: 最後に $\lfloor\lambda\rfloor = j+1$ の場合を考える．この場合も場合 2 と同様に $h_{X'}(\lfloor\lambda\rfloor) \geq h_{Y'}(\lfloor\lambda\rfloor)$ が得られる． □

系 3.5 式 (3.1.1) で定義される分布 π が対数分離凹の時，マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ は半順序集合 $(\Delta,\succeq)$ 上で単調である．証明略． ■

以上の議論と 2.3.2 節の定理 2.10 から次の定理が導かれる．

定理 3.6 関数ベクトル $\boldsymbol{f} = (f_1,\ldots,f_n)$ が任意の $k \in \{0,1,\ldots,K\}$ および任意の連続する対 f_i, f_{i+1} に関して交互不等式 (3.2.2) を満足する時，アルゴリズム 2.2 に基づく完璧サンプリング法が存在し，期待計算量は $\mathrm{O}(\mathcal{T} \cdot \mathrm{E}[T_*])$ である．ただし $\mathcal{T}$ は一回の推移にかかる計算量であり，$\mathrm{E}[T_*]$ は合流時間の期待値である．証明略． ■

系 **3.7** 式 (3.1.1) で定義される分布 π が対数分離凹の時，アルゴリズム 2.2 に基づく完璧サンプリング法が存在し，期待計算量は $\mathrm{O}(\mathcal{T}\cdot\mathrm{E}[T_*])$ である．ただし $\mathcal{T}$ は一回の推移にかかる計算量であり，$\mathrm{E}[T_*]$ は合流時間の期待値である．証明略． ■

3.3.2 期待計算時間

ここでは完璧サンプリング法の期待計算時間について議論する．まず，次の条件を導入する．

条件 **3.8** 関数ベクトル $\boldsymbol{f}=(f_1,f_2,\ldots,f_n)$ の添え字の順は，任意の $k\in\{0,1,\ldots,K-1\}$ および任意の $i\in\{1,\ldots,n-1\}$ に対して，

$$\sum_{l=0}^{k}\left(\frac{\sum_{s=0}^{l}f_i(s)f_{i+1}(k-s)}{\sum_{s=0}^{k}f_i(s)f_{i+1}(k-s)}-\frac{\sum_{s=0}^{l}f_i(s)f_{i+1}(k+1-s)}{\sum_{s=0}^{k+1}f_i(s)f_{i+1}(k+1-s)}\right)$$
$$\equiv\sum_{l=0}^{k}\left(g^{k}_{i(i+1)}(l)-g^{k+1}_{i(i+1)}(l)\right)\geq\frac{1}{2} \tag{3.3.1}$$

を満たす．

定理 **3.9** 関数ベクトル $\boldsymbol{f}=(f_1,\ldots,f_n)$ が任意の $k\in\{0,1,\ldots,K\}$ および任意の連続する対 f_i,f_{i+1} に関して交互不等式 (3.2.2) を満足し，かつ条件 3.8 を満足する時，更新関数 ϕ_{P} で定義されるマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の合流時間 T^* は

$$\mathrm{E}[T^*]\leq 2n^2(n-1)(1+\ln(Kn))$$

を満足する．

証明. 単純無向グラフ $\mathcal{G}=(\Delta,\mathcal{E})$ を第 3.2 節の定理 3.2 の証明中で定義されたものとする．各枝 $e=\{x,y\}\in\mathcal{E}$ に対して，唯一の添え字対 $j_1,j_2\in\{1,2,\ldots,n\}$ が存在して

$$|x_i-y_i|=\begin{cases}1 & (i=j_1,j_2),\\ 0 & (\text{それ以外}),\end{cases}$$

を満足する．この添え字対を e の支持対 (*supporting pair*) と呼ぶ．支持対 $\{j_1,j_2\}$ をもつ枝 $e\in\mathcal{E}$ の長さ $\ell_{\mathrm{P}}(e)$ を $j^*=\max\{j_1,j_2\}$ を用いて

$$\ell_{\mathrm{P}}(e)\stackrel{\mathrm{def.}}{=}\frac{1}{n-1}\sum_{i=1}^{j^*-1}(n-i)=\sum_{i=1}^{j^*-1}\left(1-\frac{i-1}{n-1}\right)$$

と定義する．このとき，$1\leq\min_{e\in\mathcal{E}}\ell_{\mathrm{P}}(e)\leq\max_{e\in\mathcal{E}}\ell_{\mathrm{P}}(e)\leq n/2$ が成り立つことに注意されたい．任意の頂点対 $\{x,y\}\in\binom{\Delta}{2}$ に対して距離 $d_{\mathrm{P}}(x,y)(=d_{\mathrm{P}}(y,x))$ を，グラフ $\mathcal{G}$ 上での x と y の間の ℓ_{P} に関する最短パスの長さと定義する．このとき，任意の頂点対 $\{x,y\}\in\binom{\Delta}{2}$ に対して $d_{\mathrm{P}}(x,y)\leq(n/2)\sum_{i=1}^{n}(1/2)|x_i-y_i|\leq(n/2)K$ が成り立つ．すなわち，グラフ $\mathcal{G}$ の直

径 $\max_{(X,Y)\in\Delta^2}\{d_{\mathrm{P}}(X,Y)\}$ は $Kn/2$ で上から押さえられる．また，距離の定義から任意の枝 $\{x,y\}\in\mathcal{E}$ に対して $d_{\mathrm{P}}(x,y)=\ell_{\mathrm{P}}(\{x,y\})$ が成り立つ．

以下，任意の枝 $\{X,Y\}\in\mathcal{E}$ に対して

$$\mathrm{E}[d_{\mathrm{P}}(\phi_{\mathrm{P}}(X,\Lambda),\phi_{\mathrm{P}}(Y,\Lambda))]\leq\beta\cdot d_{\mathrm{P}}(X,Y),\quad\text{かつ}\quad\beta=1-\frac{1}{n(n-1)^2},\tag{3.3.2}$$

が成り立つことを示す．なお，枝 $\{X,Y\}\in\mathcal{E}$ の支持対を $\{j_1,j_2\}$ とし，一般性を失うことなく $j_1<j_2$ かつ $X_{j_2}+1=Y_{j_2}$ が成り立つとする．また，簡単のため $X'=\phi(X,\lambda)$ および $Y'=\phi(Y,\lambda)$ と表記する．

場合 **1:** まず $\lfloor\Lambda\rfloor=j_2-1$ の場合について，

$$\mathrm{E}[d_{\mathrm{P}}(X',Y')\mid\lfloor\Lambda\rfloor=j_2-1]\leq d_{\mathrm{P}}(X,Y)-\frac{1}{2}\cdot\frac{n-j_2+1}{n-1}$$

を示す．もし $j_1=j_2-1$ ならば，$X'=Y'$ が常に成り立つため $d_{\mathrm{P}}(X',Y')=0$ となる．以下 $j_1<j_2-1$ の場合を考える．このとき $k=X_{j_2-1}+X_{j_2}$ と置くと，$X_{j_2}+1=Y_{j_2}$ の仮定から $Y_{j_2-1}+Y_{j_2}=k+1$ が成り立つ．いま $l\in\{0,1,\ldots,k\}$ を $g^k_{(j_2-1)j_2}(l-1)\leq\Lambda-\lfloor\Lambda\rfloor<g^k_{(j_2-1)j_2}(l)$ を満足するものとする．この時，交互不等式 (3.2.3) より

$$g^{k+1}_{(j_2-1)j_2}(l-1)\leq\Lambda-\lfloor\Lambda\rfloor<g^{k+1}_{(j_2-1)j_2}(l+1)$$

が成り立つので，$Y'_{j_2-1}=l$ ならば $X'_{j_2-1}=l$ または $l+1$ が成り立つ．すなわち

$$\begin{pmatrix}X'_{j_2-1}\\Y'_{j_2-1}\end{pmatrix}\in\left\{\begin{pmatrix}0\\0\end{pmatrix},\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}1\\1\end{pmatrix},\begin{pmatrix}1\\2\end{pmatrix},\ldots,\begin{pmatrix}k\\k\end{pmatrix},\begin{pmatrix}k\\k+1\end{pmatrix}\right\},$$

が成り立つ．この時，$X'_{j_2-1}=Y'_{j_2-1}$ の場合は状態対 $\{X',Y'\}$ の支持対が $\{j_1,j_2\}$ となるので $d_{\mathrm{P}}(X',Y')=d_{\mathrm{P}}(X,Y)$ が成り立ち，$X'_{j_2-1}\neq Y'_{j_2-1}$ の場合は状態対 $\{X',Y'\}$ の支持対が $\{j_1,j_2-1\}$ となるので $d_{\mathrm{P}}(X',Y')=d_{\mathrm{P}}(X,Y)-(n-j_2+1)/(n-1)$ が成り立つ．後者の生起確率を考えると

$$\Pr[X'_{j_2-1}\neq Y'_{j_2-1}\mid\lfloor\Lambda\rfloor=j_2-1]=\sum_{l=0}^{k'}\left(g^{k'}_{(j_2-1),j_2}(l)-g^{k'+1}_{(j_2-1),j_2}(l)\right),$$

であり，定理の仮定から条件 3.8 が満たされることを考慮すると

$$\begin{aligned}\Pr\left[X'_{j_2-1}\neq Y'_{j_2-1}\mid\lfloor\Lambda\rfloor=j_2-1\right]&\geq 1/2,\\ \Pr\left[X'_{j_2-1}=Y'_{j_2-1}\mid\lfloor\Lambda\rfloor=j_2-1\right]&\leq 1/2\end{aligned}$$

が得られる．したがって

$$\begin{aligned}\mathrm{E}\left[d_{\mathrm{P}}(X',Y')\mid\lfloor\Lambda\rfloor=j_2-1\right]&\leq\frac{1}{2}d_{\mathrm{P}}(X,Y)+\frac{1}{2}\left(d_{\mathrm{P}}(X,Y)-\frac{n-j_2+1}{n-1}\right)\\&=d_{\mathrm{P}}(X,Y)-\frac{1}{2}\cdot\frac{n-j_2+1}{n-1}\end{aligned}$$

を得る．

場合 2: 次に $\lfloor \Lambda \rfloor = j_2$ の場合について考える．このときも場合 1 と同様に

$$\mathrm{E}[d_{\mathrm{P}}(X', Y') | \lfloor \Lambda \rfloor = j_2] \ \leq \ d_{\mathrm{P}}(X, Y) + \frac{1}{2} \cdot \frac{n - j_2}{n - 1}$$

が得られる．

場合 3: 最後に $\lfloor \Lambda \rfloor \neq j_2 - 1$ かつ $\lfloor \Lambda \rfloor \neq j_2$ の場合を考える．この時，状態対 $\{X', Y'\}$ の支持対 $\{j_1', j_2'\}$ は $j_2 = \max\{j_1', j_2'\}$ を満足する．ゆえに $d_{\mathrm{P}}(X, Y) = d_{\mathrm{P}}(X', Y')$ を得る．

場合 1 の生起確率が $1/(n-1)$ であり，場合 2 の生起確率が $1/(n-1)$ 以下であることから，任意の枝 $\{X, Y\} \in \mathcal{E}$ に対して

$$\begin{aligned}
\mathrm{E}[d(X', Y')] &\leq d_{\mathrm{P}}(X, Y) - \frac{1}{n-1} \cdot \frac{1}{2} \cdot \frac{n - j_2 + 1}{n-1} + \frac{1}{n-1} \cdot \frac{1}{2} \cdot \frac{n - j_2}{n-1} \\
&= d_{\mathrm{P}}(X, Y) - \frac{1}{2(n-1)^2} \\
&\leq \left(1 - \frac{1}{2(n-1)^2} \cdot \frac{1}{\max_{\{X,Y\} \in \mathcal{E}} \{d_{\mathrm{P}}(X, Y)\}}\right) d_{\mathrm{P}}(X, Y) \\
&= \left(1 - \frac{1}{n(n-1)^2}\right) d_{\mathrm{P}}(X, Y)
\end{aligned}$$

を得る．

いま，任意の頂点対 $\{X, Y\} \in \binom{\Delta}{2}$ に対して，頂点列 $Z_0, Z_2, \ldots, Z_p$ は $Z_0 = X$，$Z_p = Y$ で，各 $i \in \{1, \ldots, p\}$ に対して $\{Z_{i-1}, Z_i\} \in \mathcal{E}$ であり，さらに

$$d_{\mathrm{P}}(X, Y) = \sum_{i=1}^{p} d_{\mathrm{P}}(Z_{i-1}, Z_i)$$

を満足するものとする．このとき，距離 d_{P} がグラフ $\mathcal{G}$ の最短パスで定義されていることを考慮すると，各 $i \in \{1, \ldots, p\}$ について $Z_i' = \phi_{\mathrm{P}}(Z_i, \Lambda)$ と表すならば，

$$\begin{aligned}
d_{\mathrm{P}}(X', Y') &\leq \sum_{i=1}^{p} d_{\mathrm{P}}(Z_{i-1}', Z_i') \\
&\leq \sum_{i=1}^{p} \left(1 - \frac{1}{n(n-1)^2}\right) d_{\mathrm{P}}(Z_{i-1}, Z_i) \\
&= \left(1 - \frac{1}{n(n-1)^2}\right) \sum_{i=1}^{p} d_{\mathrm{P}}(Z_{i-1}, Z_i) \\
&= \left(1 - \frac{1}{n(n-1)^2}\right) d_{\mathrm{P}}(X, Y)
\end{aligned}$$

が成り立つ．また，$D \stackrel{\mathrm{def.}}{=} d_{\mathrm{P}}(x_{\max}, x_{\min})$ とし，$\tau_0 \stackrel{\mathrm{def.}}{=} n(n-1)^2(1 + \ln D)$ とおくと

$$\Pr[T_* > \tau_0] = \Pr\left[\Phi^0_{-\tau_0}(x_{\max}, \boldsymbol{\Lambda}) \neq \Phi^0_{-\tau_0}(x_{\min}, \boldsymbol{\Lambda})\right]$$

$$
\begin{aligned}
&= \Pr\left[\Phi_0^{\tau_0}(x_{\max}, \mathbf{\Lambda}) \neq \Phi_0^{\tau_0}(x_{\min}, \mathbf{\Lambda})\right] \\
&\leq \sum_{\{X,Y\} \in \binom{\Delta}{2}} d_{\mathrm{P}}(X,Y) \Pr\left[X = \Phi_0^{\tau_0}(x_{\max}, \mathbf{\Lambda}), Y = \Phi_0^{\tau_0}(x_{\min}, \mathbf{\Lambda})\right] \\
&= \mathrm{E}\left[d_{\mathrm{P}}\left(\Phi_0^{\tau_0}(x_{\max}, \mathbf{\Lambda}), \Phi_0^{\tau_0}(x_{\min}, \mathbf{\Lambda})\right)\right] \\
&\leq \left(1 - \frac{1}{n(n-1)^2}\right)^{\tau_0} d_{\mathrm{P}}(x_{\max}, x_{\min}) \\
&= \left(1 - \frac{1}{n(n-1)^2}\right)^{n(n-1)^2(1+\ln D)} D \ \leq \ \mathrm{e}^{-1}\mathrm{e}^{-\ln D} D \ = \ \frac{1}{\mathrm{e}}
\end{aligned}
$$

が得られる．ここで合流時間の劣乗法性に注目すると，任意の $k \in \mathbb{Z}_+$ に対して

$$
\Pr[T_* > k\tau_0] \leq (\Pr[T_* > \tau_0])^k \leq (1/\mathrm{e})^k
$$

が成り立つので，

$$
\begin{aligned}
\mathrm{E}[T_*] = \textstyle\sum_{t=0}^{\infty} t\Pr[T_* = t] \ &\leq \ \tau_0 + \tau_0 \Pr[T_* > \tau_0] + \tau_0 \Pr[T_* > 2\tau_0] + \cdots \\
&\leq \tau_0 + \tau_0/\mathrm{e} + \tau_0/\mathrm{e}^2 + \cdots \ = \ \tau_0/(1 - 1/\mathrm{e}) \ \leq \ 2\tau_0
\end{aligned}
$$

を得る．ゆえに $D \leq Kn/2$ より，$\mathrm{E}[T_*] \leq 2n(n-1)^2(1 + \ln Kn)$ が得られる． □

系 3.10 関数ベクトル $\boldsymbol{f} = (f_1, \ldots, f_n)$ が任意の $k \in \{0, 1, \ldots, K\}$ および任意の連続する対 f_i, f_{i+1} に関して交互不等式 (3.2.2) を満たし，かつ条件 3.8 を満たす時，任意の ε $(0 < \varepsilon < 1)$ に対して，マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の混交時間 $\tau(\varepsilon)$ は

$$
\tau(\varepsilon) \leq n(n-1)^2 \ln(Kn\varepsilon^{-1})
$$

を満たす．

証明. 定理 3.9 の証明中の式 (3.3.2) と経路カップリング定理 2.5 から得られる． □

条件 1 を満たす具体例については，次の章でジャクソンネットワーク単一サーバモデルや離散化 Dirichlet 分布がある事を示す．

4　モンテカルロ積分と近似精度保証

この章では式 $(3.1.1)$ の正規化定数 C の乱択近似計算法を提案する．

4.1　再帰構造とモンテカルロ法

この節では，まず再帰アルゴリズムの概要について述べる．

非負整数 $N \in \mathbb{Z}_+$ と非負整数ベクトル $\boldsymbol{c} \in \mathbb{Z}_+^n$ に対して，定数 $C(N;\boldsymbol{c})$ を

$$C(N;\boldsymbol{c}) \stackrel{\text{def.}}{=} \sum_{\boldsymbol{x}\in\Delta(N)} \prod_{i=1}^{n} f_i(x_i + c_i)$$

と定義する．明らかに $C(K;\boldsymbol{0}) = C$ が成り立ち，すなわち所望の正規化定数である．いま，各 $j \in \{1,\ldots,n\}$ に対して，集合 $\Xi_j(N) \subset \Delta(N)$ を

$$\Xi_j(N) \stackrel{\text{def.}}{=} \left\{ \boldsymbol{x} \in \Delta(N) \mid x_j \geq \lceil \tfrac{N}{n} \rceil \right\}$$

と定義する．このとき，$\Xi_j(N)$ の凸包もまた単体であることに注意が必要である．ここで

$$F_j(N;\boldsymbol{c}) \stackrel{\text{def.}}{=} \sum_{\boldsymbol{x}\in\Xi_j(N)} \prod_{i=1}^{n} f_i(x_i + c_i)$$

とする．また，$\boldsymbol{e}_j$ を添え字 j の値が 1 でその他の値が 0 の単位ベクトルとする．いま

$$\left[\forall \boldsymbol{y} \in \Delta\left(N - \lceil \tfrac{N}{n} \rceil\right),\ \boldsymbol{y} + \lceil \tfrac{N}{n} \rceil \boldsymbol{e}_j \in \Xi_j(N)\right] \text{ かつ}$$
$$\left[\forall \boldsymbol{x} \in \Xi_j(N),\ \boldsymbol{x} - \lceil \tfrac{N}{n} \rceil \boldsymbol{e}_j \in \Delta\left(N - \lceil \tfrac{N}{n} \rceil\right)\right]$$

より，$\Xi_j(N)$ と $\Delta\left(N - \lceil \frac{N}{n} \rceil\right)$ の間には全単射が存在するので，

$$\begin{aligned} F_j(N;\boldsymbol{c}) &= \sum_{\boldsymbol{x}\in\Xi_j(N)} \prod_{i=1}^{n} f_i(x_i + c_i) \\ &= \sum_{\boldsymbol{y}\in\Delta\left(N-\lceil \frac{N}{n} \rceil\right)} f_j\left(y_j + c_j + \lceil \tfrac{N}{n} \rceil\right) \prod_{i\neq j} f_i(y_i + c_i) \\ &= C\left(N - \lceil \tfrac{N}{n} \rceil ; \boldsymbol{c} + \lceil \tfrac{N}{n} \rceil \boldsymbol{e}_j\right) \end{aligned}$$

が成り立つ．ここで

$$C(N;\boldsymbol{c}) = \frac{C(N;\boldsymbol{c})}{F_j(N;\boldsymbol{c})} \cdot C\left(N - \lceil \tfrac{N}{n} \rceil ; \boldsymbol{c} + \lceil \tfrac{N}{n} \rceil \boldsymbol{e}_j\right)$$

が成り立つので，もし

$$\frac{F_j(N;\boldsymbol{c})}{C(N;\boldsymbol{c})} = \sum_{\boldsymbol{x}\in\Xi_j(N)} \frac{\prod_{i=1}^{n} f_i(x_i + c_i)}{C(N;\boldsymbol{c})}$$

および $C\left(N-\left\lceil\frac{N}{n}\right\rceil;\boldsymbol{c}+\left\lceil\frac{N}{n}\right\rceil\boldsymbol{e}_j\right)$ の値が分かれば $C(N;\boldsymbol{c})$ を計算することができる．したがって上記の式を再帰的に計算することで $C(K;0)$ を計算することができる．ベクトル $\boldsymbol{x}\in\Delta(N)$ に対して関数 $\pi^N_{\boldsymbol{c}}:\Delta(N)\to\mathbb{R}_{++}$ を

$$\pi^N_{\boldsymbol{c}}(\boldsymbol{x})\stackrel{\text{def.}}{=}\frac{1}{C(N;\boldsymbol{c})}\prod_{i=1}^{n}f_i(x_i+c_i) \tag{4.1.1}$$

と定義すると，$\pi^N_{\boldsymbol{c}}(\boldsymbol{x})$ は $\Delta(N)$ 上の確率関数となり，

$$\frac{F_j(N;\boldsymbol{c})}{C(N;\boldsymbol{c})}=\sum_{\boldsymbol{x}\in\Xi_j(N)}\pi^N_{\boldsymbol{c}}(\boldsymbol{x})$$

が成り立つ．それぞれ $j^r\in\{1,2,\dots,n\}$ であるような任意の添え字からなる列 $(j^0,j^1,\dots,j^{R-1})$ に対して上記の再帰計算を当てはめると

$$\begin{aligned}C(K;\mathbf{0})&=C(K^1;\boldsymbol{c}^1)\cdot\frac{C(K^0;\boldsymbol{c}^0)}{F_{j^0}(K^0;\boldsymbol{c}^0)}\\&=C(K^2;\boldsymbol{c}^2)\cdot\frac{C(K^1;\boldsymbol{c}^1)}{F_{j^1}(K^1;\boldsymbol{c}^1)}\cdot\frac{C(K^0;\boldsymbol{c}^0)}{F_{j^0}(K^0;\boldsymbol{c}^0)}\\&=\cdots\\&=C(K^R;\boldsymbol{c}^R)\cdot\prod_{r=0}^{R-1}\frac{C(K^r;\boldsymbol{c}^r)}{F_{j^r}(K^r;\boldsymbol{c}^r)}\end{aligned}$$

が成り立つ．ただし，$K^0\stackrel{\text{def.}}{=}K$, $\boldsymbol{c}^0\stackrel{\text{def.}}{=}\mathbf{0}$, $K^r\stackrel{\text{def.}}{=}K^{r-1}-\left\lceil\frac{K^{r-1}}{n}\right\rceil$, および各 $r=1,2,\dots,R$ に対して $\boldsymbol{c}^r\stackrel{\text{def.}}{=}\boldsymbol{c}^{r-1}+\left\lceil\frac{K^{r-1}}{n}\right\rceil\boldsymbol{e}_{j^{r-1}}$ である．いま $R\in\mathbb{Z}_{++}$ を $K^R=0$ を満足する最小のものとすると，

$$C(K;\mathbf{0})=\prod_{i=1}^{n}f_i(c^R_i)\cdot\prod_{r=0}^{R-1}\frac{C(K^r;\boldsymbol{c}^r)}{F_{j^r}(K^r;\boldsymbol{c}^r)} \tag{4.1.2}$$

が成り立つ．

補題 **4.1** 任意の $K\in\mathbb{Z}_{++}$ に対して反復回数 R は $R\le n\ln K+1$ を満足する．

証明. もし $n=1$ ならば $R=1$ となり主張を得る．もし $n\ge2$ かつ $K=1,2$ ならば，それぞれ $R=1,2$ となり主張を得る．以下，$n\ge2$ かつ $K\ge3$ の場合を考える．まず R' を

$$R'\stackrel{\text{def.}}{=}\min\left\{r\;\middle|\;K\left(\tfrac{n-1}{n}\right)^r<1\right\},$$

と定義すると任意の $K'\in\mathbb{Z}_{++}$ に対して $K'-\lceil K'/n\rceil\le K'(n-1)/n$ が成り立つので，$R\le R'$ が成り立つ．したがって

$$K\left(\tfrac{n-1}{n}\right)^{n\ln K}\le1$$

を示せば十分である．いま，$\ln K > 0$ であることを考えると

$$\left(\tfrac{n-1}{n}\right)^{n \ln K} = \left(\left(1 - \tfrac{1}{n}\right)^{n}\right)^{\ln K} \leq \left(\tfrac{1}{\mathrm{e}}\right)^{\ln K} = 1/K$$

が成り立つ．よって題意を得る． □

正規化定数 C の計算に対する FPRAS を得るには次の 3 点

- 効率的に分布 $\pi_{\boldsymbol{c}}^{N}$ からサンプリングする方法，
- 再帰の方法，すなわち，添え字 j^r の選び方 ，
- 精度保証のある近似解を得るためのモンテカルロ法のサンプリング回数，

についての議論が必要である．以下この章では，上記 3 点に注意し，FPRAS を得るための近似計算法について議論する．

4.2 完璧サンプリング法を用いた場合

この節では，分布 $\pi_{\boldsymbol{c}}^{N}$ に従う完璧サンプリング法の存在を仮定して，正規化定数 $C = C(K; \mathbf{0})$ を計算するための乱択近似計算法を設計する．第 2.3 節で定義したマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}(N; \boldsymbol{c})$ を用いた完璧サンプリング法を利用し，モンテカルロ法に基づいて各 $r \in \{0, 1, \ldots, R-1\}$ に対して $C(K^r; \boldsymbol{c}^r)/F_j(K^r; \boldsymbol{c}^r)$ を近似的に計算することができる．アルゴリズムの全体は以下のように記述される．

アルゴリズム **4.1** (完璧サンプリング法を用いた乱択近似スキーム)

ステップ **0.** 各パラメータを $r := 0$, $N := K$, $\boldsymbol{c} := \mathbf{0}$ と置く．

ステップ **1.** パラメータ $N \geq 1$ の間以下の i から iv を繰り返す．

i. 分布 $\pi_{\boldsymbol{c}}^{N}$ に従う Q_{P} 個のランダムサンプルを生成する．

ii. 各 $i \in \{1, \ldots, n\}$ に対して，U_i を $x_i \geq N/n$ を満たすサンプルの個数とする．

iii. 添え字 $I \in \{1, \ldots, n\}$ を $U_I = \max\{U_i\}$ を満足するものとし，$Z^r := U_I/Q_{\mathrm{P}}$ とする．

iv. ベクトルを $\boldsymbol{c} := \boldsymbol{c} + \lceil N/n \rceil \cdot \boldsymbol{e}_I$ と更新し，$N := N - \lceil N/n \rceil$ および $r := r + 1$ と更新する．

ステップ **2.** 値 $Z := \prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} 1/Z^r$ を出力する．

パラメータを Q_{P} を適切に設定することで，以下の定理が得られる．アルゴリズム 4.1 のステップ 1 の iii において，反復を通して $U_I \geq N/n$ および $Z^r \geq 1/n$ が成り立つことに注意されたい．

定理 **4.2** 任意の正数 $\varepsilon, \delta < 1$ に対して，パラメータを $Q_{\mathrm{P}} = 24nR^2\varepsilon^{-2}\ln(2nR/\delta)$ と置くと，アルゴリズム 4.1 の出力する近似解 Z は

$$\Pr\left[|Z - C| \leq \varepsilon C\right] \geq 1 - \delta$$

を満足する．このとき，サンプリングにかかる計算量を T_{S} とすると，アルゴリズム 4.1 の計算量は $\mathrm{O}((T_{\mathrm{S}} + n)n^4(\ln K)^3(\ln n + \ln\ln K)\varepsilon^{-2}\ln(\delta^{-1}))$ である．

証明. 計算量に関しては，ステップ 1 の ii で 1 サンプル当り n 回の判定を行っていることに注意すると，ステップ 1 の 1 反復にかかる時間が $(T_{\mathrm{S}}+n)Q_{\mathrm{P}}$ であり，ステップ 1 の反復回数が R 回である．以上の事実と補題 4.1 およびパラメータ Q_{P} の定義から題意を得る．

以下，定理の前半の主張を示す．まず，アルゴリズム 4.1 のステップ 1 の r $(r \in \{0\dots,R-1\})$ 回目の反復について，以下の主張 1，2 を示す．添え字 I 整数 N およびベクトル $\boldsymbol{c}$ をアルゴリズム 4.1 のステップ 1 の r 回目の反復時の値とし，ω^r を

$$\omega^r \stackrel{\text{def.}}{=} \frac{F_I(N;\boldsymbol{c})}{C(N;\boldsymbol{c})}$$

とする．各添え字 $i \in \{1,\dots,n\}$ に対して，分布 $\pi^N_{\boldsymbol{c}}$ に従うサンプルが $x_i \geq N/n$ を満足する確率を p_i とする．特に $\omega^r = p_I$ であることを注意されたい．

主張 **1.** 各 $r \in \{0,\dots,R-1\}$ に対して，$\Pr\left[\omega^r \leq \frac{1}{2n}\right] < \frac{\delta}{2R}$ が成り立つ．

添え字の集合 $S \subset \{1,\dots,n\}$ を $S \stackrel{\text{def.}}{=} \{i \mid p_i \leq 1/(2n)\}$ と定義する．任意の $i \in S$ に対して，事象 $U_i/Q_{\mathrm{P}} \geq 1/n$ の生起確率は

$$\begin{aligned}\Pr\left[\frac{U_i}{Q_{\mathrm{P}}} \geq \frac{1}{n}\right] &= \Pr\left[\frac{U_i}{Q_{\mathrm{P}}} \geq \left(1+\left(\frac{1}{np_i}-1\right)\right)p_i\right] \\ &\leq \mathrm{e}^{-\frac{1}{3}\left(\frac{1}{np_i}-1\right)^2 Q_{\mathrm{P}}p_i} \leq \mathrm{e}^{-\frac{1}{3}1^2 Q_{\mathrm{P}}\frac{1}{2n}} < \mathrm{e}^{-\ln\frac{2nR}{\delta}} \leq \frac{\delta}{2nR}\end{aligned}$$

を満足する．ただし，不等式の最初の不等号は Chernoff 限界を用いて得られる．いま，$U_I/Q_{\mathrm{P}} \geq 1/n$ より，事象 $I \in S$ の生起確率は

$$\Pr[I \in S] \leq (n-1)\cdot\frac{\delta}{2nR} < \frac{\delta}{2R}$$

を満足し，主張 1 を得る．

主張 **2.** 各 $r \in \{0,\dots,R-1\}$ に対して $\omega^r \geq \frac{1}{2n}$ ならば $\Pr\left[|Z^r-\omega^r| \geq \frac{\varepsilon}{2R}\omega^r\right] \leq \frac{\delta}{2R}$ が成り立つ．ただし Z^r はアルゴリズム 4.1 のステップ 1 の iii で定義されるものである．

Chernoff 限界を用いると，$n \geq 2$ より

$$\begin{aligned}\Pr\left[|Z^r-\omega^r| \geq \frac{\varepsilon}{2R}\omega^r\right] &\leq 2\mathrm{e}^{-\left(\frac{\varepsilon}{2R}\right)^2\frac{1}{3}24nR^2\varepsilon^{-2}\ln\left(\frac{2nR}{\delta}\right)\omega^r} = 2\mathrm{e}^{-2n\ln\left(\frac{2nR}{\delta}\right)\omega^r} \\ &\leq 2\mathrm{e}^{-2n\ln\left(\frac{2nR}{\delta}\right)\frac{1}{2n}} = 2\mathrm{e}^{-\ln\frac{2nR}{\delta}} = \frac{2\delta}{2nR} \leq \frac{\delta}{2R}\end{aligned}$$

が成り立ち，主張 2 を得る．

主張 1，2 を用いて，次の主張 3 が得られる．

主張 **3.** 各 $r \in \{0,\dots,R-1\}$ に対して，$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-1} \leq \frac{\omega^r}{Z^r} \leq \left(1+\frac{\varepsilon}{2R}\right)\right] \geq 1-\frac{\delta}{R}$ が成り立つ．

もし $\omega^r \geq \frac{1}{2n}$ が成り立てば，主張 2 より，

$$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-1} Z^r \leq \widehat{\omega}^r \leq \left(1-\frac{\varepsilon}{2R}\right)^{-1} Z^r\right] \geq 1-\frac{\delta}{2R} \qquad (\forall r \in \{1,\ldots,R-1\})$$

が得られる．主張 1 より確率 $1-\frac{\delta}{2R}$ 以上で $\omega^r \geq \frac{1}{2n}$ が成り立つので，各 $r \in \{1,\ldots,R-1\}$ に対して

$$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-1} \leq \frac{\omega^r}{Z^r} \leq \left(1+\frac{\varepsilon}{2R}\right)\right] \geq 1-\frac{\delta}{2R}-\frac{\delta}{2R} \geq 1-\frac{\delta}{R}$$

が成り立ち，主張 4 を得る．

主張 3 から次の主張 4 が得られる．

主張 **4.** アルゴリズム 4.1 中の確率変数 $Z^0,\ldots,Z^{R-1}$ は

$$\Pr\left[(1-\varepsilon) \leq \frac{\omega^0\cdots\omega^{R-1}}{Z^0\cdots Z^{R-1}} \leq (1+\varepsilon)\right] \geq 1-\delta$$

を満足する．

主張 3 は各 r 反復について成り立つので，これらを掛け合わせると

$$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-R} \leq \frac{\omega^0\cdots\omega^{R-1}}{Z^0\cdots Z^{R-1}} \leq \left(1+\frac{\varepsilon}{2R}\right)^{R}\right] \geq \left(1-\frac{\delta}{R}\right)^{R} \geq 1-\delta.$$

が得られる．いま $\left(1+\frac{\varepsilon}{2R}\right)^{R} \leq 1+\varepsilon$ および $\left(1+\frac{\varepsilon}{2R}\right)^{-R} \geq (1+\varepsilon)^{-1} \geq 1-\varepsilon$ より，主張 4 を得る．

最後に主張 4 から定理の主張を導く．ベクトル $\boldsymbol{c}=(c_1,\ldots,c_n)$ をアルゴリズム 4.1 のステップ 2 中で得られるものとすると，式 (4.1.2) より $\prod_{i=1}^n f_i(c_i)/(\omega^0\cdots\omega^{R-1}) = C$ に注意すると，アルゴリズム 4.1 の出力は $Z=\prod_{i=1}^n f_i(c_i)/(Z^0\cdots Z^{R-1})$ である．主張 4 より

$$\Pr\left[(1-\varepsilon) \leq \frac{Z}{C} \leq (1+\varepsilon)\right] \geq 1-\delta$$

が成り立つので，題意を得る． □

4.3 近似サンプリング法を用いた場合

前節では分布 $\pi_{\boldsymbol{c}}^N$ に従う完璧サンプリング法を仮定したモンテカルロ法について議論した．しかし，近似サンプリング法しか持たない場合，サンプルの分布のずれに着目した，より注意深い議論が必要となる．この節では多項式時間のサンプリング法の存在を仮定し，正規化定数 C を計算するための FPRAS を設計する．マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N;\boldsymbol{c})$ を用いた近似サンプリング法を利用したモンテカルロ法により，各 $r \in \{0,1,\ldots,R-1\}$ に対して $C(K^r;\boldsymbol{c}^r)/F_i(K^r;\boldsymbol{c}^r)$ を近似的に計算することができる．アルゴリズムの全体は以下のように記述される．

アルゴリズム **4.2** (近似サンプリング法を用いた乱択近似スキーム)

ステップ **0.** 各パラメータを $r := 0$, $N := K$, $\boldsymbol{c} := \boldsymbol{0}$ と置く．

ステップ **1.** パラメータ $N \geq 1$ の間以下の i から iv を繰り返す．

i. マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N; \boldsymbol{c})$ を $T_{\mathrm{A}}(N)$ 回推移させて Q_{A} 個のランダムサンプルを生成する．

ii. 各 $i \in \{1, \ldots, n\}$ に対して，U_i を $x_i \geq N/n$ を満たすサンプルの個数とする．

iii. 添え字 $I \in \{1, \ldots, n\}$ を $U_I = \max\{U_i\}$ を満足するものとし，$Z^r := U_I / Q_{\mathrm{A}}$ とする．

iv. ベクトルを $\boldsymbol{c} := \boldsymbol{c} + \lceil N/n \rceil \cdot \boldsymbol{e}_I$ と更新し，$N := N - \lceil N/n \rceil$ および $r := r + 1$ と更新する．

ステップ **2.** 値 $Z := \prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} 1/Z^r$ を出力する．

アルゴリズムでは，各反復においてマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N; \boldsymbol{c})$ を $T_{\mathrm{A}}(N)$ 回ずつ推移させて Q_{A} 個のサンプルを生成する．パラメータを $Q_{\mathrm{A}} = 150nR^2\varepsilon^{-2}\ln(2nR/\delta)$ および $T_{\mathrm{A}}(k) = \left\lceil \frac{n(n-1)}{2} \ln \frac{10nRk}{\varepsilon} \right\rceil$ と設定することで，以下の定理が得られる．アルゴリズム 4.2 のステップ 1 の iii において，反復を通して $U_I \geq N/n$ および $Z^r \geq 1/n$ が成り立つことに注意されたい．

定理 4.3 パラメータを $Q_{\mathrm{A}} = 150nR^2\varepsilon^{-2}\ln(2nR/\delta)$ および $T_{\mathrm{A}}(k) = \left\lceil \frac{n(n-1)}{2} \ln \frac{10nRk}{\varepsilon} \right\rceil$ と置くと，アルゴリズム 4.2 の出力する近似解 Z は

$$\Pr\left[|Z - C| \leq \varepsilon C\right] \geq 1 - \delta$$

を満足する．このとき，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の推移にかかる計算量を $\mathcal{T}$ と置くと，アルゴリズム 4.2 の計算量は $\mathrm{O}(\mathcal{T}n^6(\ln K)^4(\ln n + \ln\ln K)^2\varepsilon^{-2}\ln(\varepsilon^{-1})\ln(\delta^{-1}))$ である．

証明. 計算量に関しては ,ステップ 1 の i で 1 サンプル当りマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ を $\mathrm{O}(n^2\ln(nK\varepsilon^{-1}))$ 回推移させていることに注意すると，ステップ 1 の 1 反復にかかる時間が $\mathrm{O}(\mathcal{T}n^2\ln(nK\varepsilon^{-1})Q_{\mathrm{P}})$ であり，ステップ 1 の反復回数が R 回である．以上の事実と補題 4.1 およびパラメータの与えられ方から題意を得る．

以下，定理の前半の主張を示す．まず，アルゴリズム 4.2 のステップ 1 の r $(r \in \{0 \ldots, R-1\})$ 回目の反復について，以下の主張 1 から 4 を示す．いま，添え字 I 整数 N およびベクトル $\boldsymbol{c}$ をアルゴリズム 4.2 のステップ 1 の r 回目の反復時の値とし，ω^r を

$$\omega^r \stackrel{\text{def.}}{=} \frac{F_I(N; \boldsymbol{c})}{C(N; \boldsymbol{c})}$$

とする．各添え字 $i \in \{1, \ldots, n\}$ に対して，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N; \boldsymbol{c})$ を $T_{\mathrm{A}}(N)$ 回推移させて得られるサンプルが $x_i \geq N/n$ を満足する確率を p_i とする．特に $\widehat{\omega}^r \stackrel{\text{def.}}{=} p_I$ とすると，この $\widehat{\omega}^r$ は ω^r を近似している．

主張 1. 各 $r \in \{0, \ldots, R-1\}$ に対して，$\Pr\left[\widehat{\omega}^r \leq \frac{1}{2n}\right] < \frac{\delta}{2R}$ が成り立つ．

添え字の集合 $S \subset \{1,\dots,n\}$ を $S \stackrel{\text{def.}}{=} \{i \mid p_i \le 1/(2n)\}$ と定義する．任意の $i \in S$ に対して，事象 $U_i/Q_{\mathrm{A}} \ge 1/n$ の生起確率は

$$\begin{aligned}\Pr\left[\frac{U_i}{Q_{\mathrm{A}}} \ge \frac{1}{n}\right] &= \Pr\left[\frac{U_i}{Q_{\mathrm{A}}} \ge \left(1+\left(\frac{1}{np_i}-1\right)\right)p_i\right] \\ &\le \mathrm{e}^{-\frac{1}{3}\left(\frac{1}{np_i}-1\right)^2 Q_{\mathrm{A}} p_i} \le \mathrm{e}^{-\frac{1}{3}1^2 Q_{\mathrm{A}}\frac{1}{2n}} < \mathrm{e}^{-\ln\frac{2nR}{\delta}} \le \frac{\delta}{2nR}\end{aligned}$$

を満足する．ただし，不等式の最初の不等号は Chernoff 限界を用いて得られる．いま，$U_I/Q_{\mathrm{A}} \ge 1/n$ より，事象 $I \in S$ の生起確率は

$$\Pr[I \in S] \le (n-1)\cdot\frac{\delta}{2nR} < \frac{\delta}{2R}$$

を満足し，主張 1 を得る．

主張 2. 各 $r \in \{0,\dots,R-1\}$ に対して，$\widehat{\omega}^r \ge \frac{1}{2n}$ ならば，$|\omega^r - \widehat{\omega}^r| \le \frac{\varepsilon}{5R}\widehat{\omega}^r$ が成り立つ．

混交時間の定義 (2.3) および全変動距離の定義 (2.2) より，定理 3.2 から $T_{\mathrm{A}}(N)$ 回の推移により

$$|\omega^r - \widehat{\omega}^r| \le \mathrm{d}_{\mathrm{TV}}\left(P_{\boldsymbol{y}}^{T_{\mathrm{A}}(N)}, \pi\right) \le \frac{\varepsilon}{10nR}$$

が得られる．ただし，$P_{\boldsymbol{y}}^{T_{\mathrm{A}}(N)}$ はマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N;\boldsymbol{c})$ を任意の初期状態 $\boldsymbol{y} \in \Delta(N)$ から $T_{\mathrm{A}}(N)$ 回推移させた後の分布を表す．したがって，$\widehat{\omega}^r \ge 1/n$ とすると，

$$|\omega^r - \widehat{\omega}^r| \le \frac{\varepsilon}{10nR} = \frac{\varepsilon}{5R}\cdot\frac{1}{2n} \le \frac{\varepsilon}{5R}\widehat{\omega}^r$$

が成り立ち，主張 2 を得る．

主張 3. 各 $r \in \{0,\dots,R-1\}$ に対して $\widehat{\omega}^r \ge \frac{1}{2n}$ ならば $\Pr\left[|Z^r - \widehat{\omega}^r| \ge \frac{\varepsilon}{5R}\widehat{\omega}^r\right] \le \frac{\delta}{2R}$ が成り立つ．ただし Z^r はアルゴリズム 4.2 のステップ 1 の iii で定義されるものである．

Chernoff 限界を用いると，$n \ge 2$ より

$$\begin{aligned}\Pr\left[|Z^r - \widehat{\omega}^r| \ge \frac{\varepsilon}{5R}\widehat{\omega}^r\right] &\le 2\mathrm{e}^{-\left(\frac{\varepsilon}{5R}\right)^2\frac{1}{3}150nR^2\varepsilon^{-2}\ln\frac{2nR}{\delta}\widehat{\omega}^r} = 2\mathrm{e}^{-2n\ln\frac{2nR}{\delta}\widehat{\omega}^r} \\ &\le 2\mathrm{e}^{-2n\ln\frac{2nR}{\delta}\frac{1}{2n}} = 2\mathrm{e}^{-\ln\frac{2nR}{\delta}} = \frac{2\delta}{2nR} \le \frac{\delta}{2R}\end{aligned}$$

が成り立ち，主張 3 を得る．

主張 1 から 3 を用いて，次の主張 4 が得られる．

主張 4. 各 $r \in \{0,\dots,R-1\}$ に対して，$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-1} \le \frac{\omega^r}{Z^r} \le \left(1+\frac{\varepsilon}{2R}\right)\right] \ge 1-\frac{\delta}{R}$ が成り立つ．

もし $\widehat{\omega}^r \geq \frac{1}{2n}$ が成り立てば，主張 2 より，

$$\left(1-\frac{\varepsilon}{5R}\right)\widehat{\omega}^r \leq \omega^r \leq \left(1+\frac{\varepsilon}{5R}\right)\widehat{\omega}^r \quad (\forall r \in \{1,\ldots,R-1\}) \tag{4.3.1}$$

が得られる．また，もし $\widehat{\omega}^r \geq \frac{1}{2n}$ が成り立てば，主張 3 より，

$$\Pr\left[\left(1+\frac{\varepsilon}{5R}\right)^{-1} Z^r \leq \widehat{\omega}^r \leq \left(1-\frac{\varepsilon}{5R}\right)^{-1} Z^r\right] \geq 1-\frac{\delta}{2R} \quad (\forall r \in \{1,\ldots,R-1\}) \tag{4.3.2}$$

が得られる．したがって，もし $\widehat{\omega}^r \geq \frac{1}{2n}$ が成り立てば不等式 (4.3.1) と不等式 (4.3.2) より，各 $r \in \{1,\ldots,R-1\}$ に対して確率 $1-\frac{\delta}{2R}$ 以上で

$$\left(1-\frac{\varepsilon}{5R}\right)\left(1+\frac{\varepsilon}{5R}\right)^{-1} \leq \frac{\omega^r}{Z^r} \leq \left(1+\frac{\varepsilon}{5R}\right)\left(1-\frac{\varepsilon}{5R}\right)^{-1} \tag{4.3.3}$$

が成り立つ．式 (4.3.3) の右辺について，

$$(\text{r.h.s.}) = \left(1-\frac{\varepsilon}{5R}\right)\left(1+\frac{\varepsilon}{5R}\right)^{-1} = \left(1+\frac{2\varepsilon}{5R-\varepsilon}\right) \leq \left(1+\frac{2\varepsilon}{4R}\right) = \left(1+\frac{\varepsilon}{2R}\right)$$

が成り立ち，式 (4.3.3) の左辺に対しては，

$$(\text{l.h.s.}) = \left(1+\frac{\varepsilon}{5R}\right)\left(1-\frac{\varepsilon}{5R}\right)^{-1} = (\text{r.h.s.})^{-1} \geq \left(1+\frac{\varepsilon}{2R}\right)^{-1}$$

が成り立つので，主張 1 を用いて

$$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-1} \leq \frac{\omega^r}{Z^r} \leq \left(1+\frac{\varepsilon}{2R}\right)\right] \geq 1-\frac{\delta}{2R}-\frac{\delta}{2R} \geq 1-\frac{\delta}{R}$$

が成り立ち，主張 4 を得る．

主張 4 から次の主張 5 が得られる．

主張 5. アルゴリズム 4.2 中の確率変数 $Z^0,\ldots,Z^{R-1}$ は

$$\Pr\left[(1-\varepsilon) \leq \frac{\omega^0\cdots\omega^{R-1}}{Z^0\cdots Z^{R-1}} \leq (1+\varepsilon)\right] \geq 1-\delta$$

を満足する．

主張 4 は各 r 反復について成り立つので，これらを掛け合わせると

$$\Pr\left[\left(1+\frac{\varepsilon}{2R}\right)^{-R} \leq \frac{\omega^0\cdots\omega^{R-1}}{Z^0\cdots Z^{R-1}} \leq \left(1+\frac{\varepsilon}{2R}\right)^{R}\right] \geq \left(1-\frac{\delta}{R}\right)^{R} \geq 1-\delta$$

が得られる．いま $\left(1+\frac{\varepsilon}{2R}\right)^{R} \leq 1+\varepsilon$ および $\left(1+\frac{\varepsilon}{2R}\right)^{-R} \geq (1+\varepsilon)^{-1} \geq 1-\varepsilon$ より，主張 5 を得る．

最後に主張 5 から定理の主張を導く．ベクトル $\boldsymbol{c} = (c_1, \ldots, c_n)$ をアルゴリズム 4.2 のステップ 2 中で得られるものとすると，式 (4.1.2) より $\prod_{i=1}^{n} f_i(c_i)/(\omega^0 \cdots \omega^{R-1}) = C$ に注意すると，アルゴリズム 4.2 の出力は $Z = \prod_{i=1}^{n} f_i(c_i)/(Z^0 \cdots Z^{R-1})$ である．主張 5 より

$$\Pr\left[(1-\varepsilon) \leq \frac{Z}{C} \leq (1+\varepsilon)\right] \geq 1-\delta$$

が成り立つので，題意を得る． □

4.4 最大領域の判定が容易な場合

4.2，4.3 節では，モンテカルロ法の精度を上げる目的で十分大きな $F_{j^r}(K^r; \boldsymbol{c}^r)/C(K^r; \boldsymbol{c}^r)$ を得るために，モンテカルロ法のサンプリング結果を用いて添え字 $j^r \in \{1, \ldots, n\}$ を決定した．しかし，対象によっては十分大きな $F_{j^r}(K^r; \boldsymbol{c}^r)/C(K^r; \boldsymbol{c}^r)$ を与える添え字 j^r が容易に見つかる場合がある．例えば，$j^r = \arg\max_i \{F_i(K^r; \boldsymbol{c}^r)/C(K^r; \boldsymbol{c}^r)\}$ が容易に求まる場合，$F_{j^r}(K^r; \boldsymbol{c}^r)/C(K^r; \boldsymbol{c}^r) \geq 1/n$ が成り立つ．本節では，$F_{j^r}(K^r; \boldsymbol{c}^r)/C(K^r; \boldsymbol{c}^r) \geq 1/n$ を満たす添え字 j^r が容易に見つかることを仮定した場合の乱択近似スキームについて議論する．アルゴリズムの全体は以下のように記述される．

アルゴリズム **4.3** (乱択近似スキーム)

ステップ 0. 各パラメータを $r := 0$, $N := K$, $\boldsymbol{c} := \boldsymbol{0}$ と置く．
ステップ 1. パラメータ $N \geq 1$ の間以下の i から v を繰り返す．

- o. 添え字 $i \in \{1, \ldots, n\}$ を $F_i(N; \boldsymbol{c}) \geq C(N; \boldsymbol{c})/n$ を満足するものとする．
- i. 分布 $\pi^N_{\boldsymbol{c}}$ に対する完璧/近似サンプリング法を用いて Q 個のランダムサンプルを生成する．
- ii. 添え字 i に対して，U_i を $x_i \geq N/n$ を満たすサンプルの個数とする．
- iii. $Z^r := (U_i + 1)/(Q + 1)$ とする．
- iv. ベクトルを $\boldsymbol{c} := \boldsymbol{c} + \lceil N/n \rceil \cdot \boldsymbol{e}_i$ と更新し，$N := N - \lceil N/n \rceil$ および $r := r + 1$ と更新する．

ステップ 2. 値 $Z := \alpha(\boldsymbol{c}) \cdot \prod_{r=0}^{R-1} 1/Z^r$ を出力する．

アルゴリズム中で $Z_i = U_i/M$ の代わりに $Z_i = (U_i + 1)/(M + 1)$ を用いている点に注意が必要である．

このアルゴリズムの利点として，アルゴリズム 4.1，4.2 ではステップ 1 の ii での n 回の判定を行っていたのに対して，ステップ 1 の ii で添え字 i についてだけ判定すれば良い点が挙げられる．また添え字 j^r がはあらかじめ決定的に計算できるため，並列化が容易になる点もこのアルゴリズムの利点である．しかし，このアルゴリズムの最大の利点は得られる解の期待値の真の値からの偏りが理論的に算定できる点である．この期待値の真の値からの偏りの議論が本節の主要な目的である．偏りが無視できないような例として，第 5.2.3 節の図 8 において，2 行分割表の例を挙げる．

以下，アルゴリズム 4.3 で得られる解の精度について議論する．次の定理はそれぞれ完璧サンプリング法と近似サンプリング法を用いた場合についてのものである．

定理 4.4 各反復において完璧サンプリング法を用い，任意の正数 $\varepsilon, \delta < 1$ に対して，パラメータを $Q = 16nR^2\varepsilon^{-2}\ln(2R/\delta)$ と置くと，アルゴリズム 4.3 の出力する近似解 Z は

$$\Pr\left[|Z - C| \leq \varepsilon C\right] \geq 1 - \delta$$

を満足する． ■

定理 4.5 任意の正数 $\varepsilon, \delta < 1$ に対して，各反復においてマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N; \boldsymbol{c})$ を $\left\lceil \frac{n(n-1)}{2} \ln \frac{6nRk}{\varepsilon} \right\rceil$ 回ずつ推移させた近似サンプリング法を用い，パラメータを $Q = 144nR^2\varepsilon^{-2}\ln(2R/\delta)$ と置くと，アルゴリズム 4.3 の出力する近似解 Z は

$$\Pr\left[|Z - C| \leq \varepsilon C\right] \geq 1 - \delta$$

を満足する． ■

上記 2 つの定理は 4.2 節，4.3 節と同様の議論で示されるため，ここでは証明を略す．ただし，上記の定理の証明においてはアルゴリズム中で $Z_i = U_i/M$ の代わりに $Z_i = (U_i + 1)/(M + 1)$ を用いているため，第 2.4.1 小節で導入した Chernoff 限界の変種の補題 2.21 を用いる必要があることに注意されたい．

以下，アルゴリズム 4.3 で得られる解の期待値の，真の値からの偏りについて議論する．

定理 4.6 任意の正数 $\varepsilon, \delta < 1$ に対して，各反復においてマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}(N; \boldsymbol{c})$ を $\left\lceil \frac{n(n-1)}{2} \ln \frac{6nRk}{\varepsilon} \right\rceil$ 回ずつ推移させた近似サンプリング法を用い，パラメータを $Q = 144nR^2\varepsilon^{-2}\ln(2R/\delta)$ と置くと，アルゴリズム 4.3 の出力する近似解 Z は

$$\frac{\left|\mathrm{E}\left[Z\right] - C\right|}{C} \leq \frac{\varepsilon}{4} + \mathrm{e}^{-120R^3\varepsilon^{-2}\ln(2R/\delta)} \leq \left(\frac{1}{4} + \frac{1}{10^{36}}\right)\varepsilon$$

を満足する．

証明. いま，添え字 i，整数 N およびベクトル $\boldsymbol{c}$ をアルゴリズム 4.2 のステップ 1 の r 回目の反復時の値とし，ω^r を

$$\omega^r \stackrel{\mathrm{def.}}{=} \frac{F_i(N; \boldsymbol{c})}{C(N; \boldsymbol{c})}$$

とする．また，ステップ 1 の ii で得られるサンプルの個数を U_i に対して $\widehat{\omega}^r \stackrel{\mathrm{def.}}{=} \mathrm{E}[U_i/Q]$ とする．もし完璧サンプリング法を用いていた場合 $\omega^r = \widehat{\omega}^r$ であることに注意されたい．このとき，

$$\begin{aligned}
\mathrm{E}\left[\frac{1}{Z^r}\right] &= \sum_{U_i=0}^{Q} \frac{Q+1}{U_i+1}\binom{Q}{U_i}(\widehat{\omega}^r)^{U_i}(1-\widehat{\omega}^r)^{Q-U_i} \\
&= \frac{1}{\widehat{\omega}^r}\sum_{U_i=0}^{Q}\binom{Q+1}{U_i+1}(\widehat{\omega}^r)^{U_i+1}(1-\widehat{\omega}^r)^{Q-U_i} \\
&= \frac{1}{\widehat{\omega}^r}\left(1-(1-\widehat{\omega}^r)^{Q+1}\right)
\end{aligned}$$

が成り立つ．

次に，ベクトル $\boldsymbol{c} = (c_1, \ldots, c_n)$ および $Z^0, \ldots, Z^{R-1}$ はアルゴリズム 4.5 のステップ 2 におけるものとする．確率変数 $Z^0, \ldots, Z^{R-1}$ は独立であることに注意すると，

$$\begin{aligned}\mathrm{E}\left[Z\right] &= \prod_{i=j}^{n} f_j(c_j) \cdot \mathrm{E}\left[\prod_{r=0}^{R-1} \frac{1}{Z^r}\right] = \prod_{i=j}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} \mathrm{E}\left[\frac{1}{Z^r}\right] \\ &= \prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} \frac{1}{\widehat{\omega}^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right)\end{aligned}$$

が得られる．以上のことから，

$$\begin{aligned}&|\mathrm{E}[Z] - C| \\ &= \left|\textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} \frac{1}{\widehat{\omega}^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right) - \prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} \frac{1}{\omega^r}\right| \\ &= \textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \left|\prod_{r=0}^{R-1} \frac{1}{\widehat{\omega}^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right) - \prod_{r=0}^{R-1} \frac{1}{\omega^r}\right| \\ &= \textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \Bigg|\prod_{r=0}^{R-1} \frac{1}{\widehat{\omega}^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right) \\ &\quad - \left(\prod_{r=0}^{R-1} \frac{1}{\omega^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right) + \prod_{r=0}^{R-1} \frac{1}{\omega^r}(1-\widehat{\omega}^r)^{Q+1}\right)\Bigg| \\ &= \textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \left|\prod_{r=0}^{R-1} \frac{1}{\widehat{\omega}^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right) - \prod_{r=0}^{R-1} \frac{1}{\omega^r}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right)\right| \\ &\quad + \textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \left|\prod_{r=0}^{R-1} \frac{1}{\omega^r}(1-\widehat{\omega}^r)^{Q+1}\right| \\ &= \textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} \frac{1}{\omega^r} \cdot \left|\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1\right| \cdot \prod_{r=0}^{R-1}\left(1 - (1-\widehat{\omega}^r)^{Q+1}\right) \\ &\quad + \textstyle\prod_{j=1}^{n} f_j(c_j) \cdot \prod_{r=0}^{R-1} \frac{1}{\omega^r} \cdot \prod_{r=0}^{R-1}(1-\widehat{\omega}^r)^{Q+1} \\ &\leq C \cdot \left(\left|\textstyle\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1\right| + \prod_{r=0}^{R-1}(1-\widehat{\omega}^r)^{Q+1}\right) \qquad (4.4.1)\end{aligned}$$

が得られる．

まず，式 (4.4.1) の $|\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1|$ の項について議論する．各 $r \in \{0, \ldots, R-1\}$ について，全変動距離の定義から $|\widehat{\omega}^r - \omega^r| \leq \frac{\varepsilon}{6nR}$ であり，かつ $\widehat{\omega}^r \geq \frac{1}{n} - \frac{\varepsilon}{6nR} \geq \frac{5}{6n}$ なので，

$$\left|1 - \frac{\omega^r}{\widehat{\omega}^r}\right| = \frac{|\omega^r - \widehat{\omega}^r|}{\widehat{\omega}^r} \leq \frac{\varepsilon}{6nR} \cdot \frac{6n}{5} = \frac{\varepsilon}{5R}$$

が成り立ち，

$$1 - \frac{\varepsilon}{5R} \leq \frac{\omega^r}{\widehat{\omega}^r} \leq 1 + \frac{\varepsilon}{5R}$$

を得る．もし $\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} \geq 1$ ならば，

$$\begin{aligned}\left|\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1\right| &= \prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1 \leq \left(1 + \frac{\varepsilon}{5R}\right)^R - 1 \leq \mathrm{e}^{\frac{\varepsilon}{5}} - 1 = 1 + \sum_{k=1}^{\infty} \frac{1}{k!}\left(\frac{\varepsilon}{5}\right)^k - 1 \\ &\leq \sum_{k=1}^{\infty}\left(\frac{\varepsilon}{5}\right)^k = \frac{\frac{\varepsilon}{5}}{1 - \frac{\varepsilon}{5}} = \frac{\varepsilon}{5 - \varepsilon} \leq \frac{\varepsilon}{4}\end{aligned}$$

が成り立つ．もし $\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} < 1$ ならば，

$$\left| \prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1 \right| = 1 - \prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} \leq 1 - \left(1 - \frac{\varepsilon}{5R}\right)^R = 1 - \left(\left(1 - \frac{\varepsilon}{5R}\right)^{\frac{5R}{\varepsilon}}\right)^{\frac{\varepsilon}{5}}$$
$$\leq 1 - \left(\left(1 - \frac{1}{5}\right)^5\right)^{\frac{\varepsilon}{5}} = 1 - \left(\frac{4}{5}\right)^{\varepsilon} \leq -\varepsilon \ln\left(\frac{4}{5}\right) \leq 0.224\varepsilon \leq \frac{\varepsilon}{4}$$

が成り立つ．したがって，$\left|\prod_{r=0}^{R-1} \frac{\omega^r}{\widehat{\omega}^r} - 1\right| \leq \frac{\varepsilon}{4}$ が得られる．

次に，式 $(4.4.1)$ の $\prod_{r=0}^{R-1}(1-\widehat{\omega}^r)^{Q+1}$ について議論する．

$$\prod_{r=0}^{R-1}(1-\widehat{\omega}^r)^{Q+1} \leq \left(1 - \frac{5}{6n}\right)^{R(Q+1)} \leq \left(\mathrm{e}^{-1}\right)^{\frac{5}{6n}R(Q+1)}$$
$$\leq \mathrm{e}^{-\frac{5}{6n}144nR^3\varepsilon^{-2}\ln(2R/\delta)} = \mathrm{e}^{-120R^3\varepsilon^{-2}\ln(2R/\delta)}$$

が得られる．以上の議論から題意を得る． □

同様に，完璧サンプリング法を用いた場合には以下の定理が得られる．

定理 4.7 各反復において完璧サンプリング法を用い，任意の正数 $\varepsilon, \delta < 1$ に対して，パラメータを $Q = 16nR^2\varepsilon^{-2}\ln(2R/\delta)$ と置くと，乱択近似スキーム (アルゴリズム 4.3) の出力する近似解 Z は

$$\frac{|\mathrm{E}[Z] - C|}{C} \leq \mathrm{e}^{-16R^3\varepsilon^{-2}\ln(2R/\delta)} \leq \frac{1}{10^7}\varepsilon$$

を満足する． ■

なお，定理 4.7 は定理 4.6 の場合と同様の方法で証明されるため，ここでは省略する．

5　応用

本章では対数分離凹関数を中心に，3 章，4 章の応用について述べる．

5.1　待ち行列ネットワーク

5.1.1　閉ジャクソンネットワーク

閉ジャクソンネットワークは以下の条件を満たす待ち行列ネットワークである [47]．

条件 **5.1** （閉ジャクソンネットワーク）
(i) ネットワークは $n \in \mathbb{Z}_{++}$ 個のノードをもつ．ノード $i \in \{1, \ldots, n\}$ は $s_i \in \mathbb{Z}_{++}$ 個のサーバをもち，s_i 人までの顧客を同時に処理することができる．
(ii) 各ノードでは顧客は先着順で処理される (FCFS: first-come-first-served)．
(iii) ノード $i \in \{1, \ldots, n\}$ での顧客のサービス要求時間は，サーバによらず平均 $1/\mu_i \in \mathbb{R}_{++}$ の指数分布に従う．
(iv) ノード $i \in \{1, \ldots, n\}$ でのサービスが終了した顧客は確率 $W_{ij} \in \mathrm{R}_+$ でノード $j \in \{1, \ldots, n\}$ に移動する．顧客の推移確率行列 $W = (W_{ij})$ は既約で非周期的，すなわちエルゴード的とする．
(v) ネットワークに顧客の出入りは無い．したがってネットワーク中の顧客数は常に一定である．ネットワーク中の顧客数を $K \in \mathbb{Z}_{++}$ と表す．

閉ジャクソンネットワークは，ネットワーク中の顧客の定常分布として積形式解を持つことが知られる．以下，この積形式解を記す．ノード数 n 顧客数 K の閉ジャクソンネットワークにおいて，ネットワーク中のノード上の滞在顧客数の状態は $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \Delta(K)$ として表現される．ただし x_i はノード i に滞在する顧客数である．顧客の推移確率行列 W はエルゴード的なので，左固有値 1 を持ち，それに付随する左固有ベクトルは定数倍を除き唯一に定まる．ベクトル $\theta \in \mathbb{R}^n_{++}$ を推移確率行列 W の左固有値 1 に付随する左固有ベクトルとする．すなわち $\theta W = \theta$ が成り立つ．このとき，関数 $\alpha_i : \mathbb{Z}_+ \to \mathbb{R}_{++}$ を整数 $z \in \mathbb{Z}_+$ に対して

$$\alpha_i(z) \stackrel{\text{def.}}{=} \frac{1}{\prod_{j=1}^{z} \min\{j, s_i\}} \left(\frac{\theta_i}{\mu_i}\right)^z \equiv \begin{cases} \dfrac{1}{z!}\left(\dfrac{\theta_i}{\mu_i}\right)^z & (z \le s_i), \\ \dfrac{1}{s_i^{z-s_i} s_i!}\left(\dfrac{\theta_i}{\mu_i}\right)^z & (z > s_i), \end{cases} \tag{5.1.1}$$

と定義する．ただし，$\prod_{j=1}^{0} \min\{j, s_i\} \stackrel{\text{def.}}{=} 1$ および $0! \stackrel{\text{def.}}{=} 1$ とする．

閉ジャクソンネットワークの積形式解 (*product form solution*) $\pi_{\mathrm{J}} : \Delta(K) \to \mathbb{R}_{++}$ は

$$\pi_{\mathrm{J}}(\boldsymbol{x}) \stackrel{\text{def.}}{=} \frac{1}{C_{\mathrm{J}}} \prod_{i=1}^{n} \alpha_i(x_i), \qquad (\boldsymbol{x} \in \Delta(K)) \tag{5.1.2}$$

と定義される．ただし，$C_{\mathrm{J}} \stackrel{\text{def.}}{=} \sum_{\boldsymbol{x} \in \Delta} \prod_{i=1}^{n} \alpha_i(x_i)$ は正規化定数である．以下，表記の簡便のため，ベクトル $\boldsymbol{x} \in \mathbb{Z}^n_+$ に対して $\alpha(\boldsymbol{x}) \stackrel{\text{def.}}{=} \prod_{i=1}^{n} \alpha_i(x_i)$ と記述する．

本節の目的は，ジャクソンネットワークの積形式解 (5.1.2) に対する以下の議論である．

i) マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の混交時間の多項式性の証明 (系 3.3)，
ii) マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の単調性と単調 CFTP アルゴリズムの設計 (アルゴリズム 2.2，系 3.7)，
iii) 閉ジャクソンネットワークの特殊なモデル (単一サーバモデル: 5.1.2 節) に対する，
  a. 多項式時間の近似サンプリング法の設計 (系 3.3)，
  b. 多項式時間の完璧サンプリング法の設計
  (アルゴリズム 2.2，系 3.7，条件 3.8，定理 3.9))，
  c. 正規化定数 C を計算する多項式時間乱択近似スキーム (FRPAS) の設計
  (アルゴリズム 4.1， 定理 4.2，アルゴリズム 4.2，定理 4.3)．

上記目的のうち，iii については次の第 5.1.2 節で議論する．以下，上記 i, ii, の目的で，積形式解が対数分離凹関数であることを示す．

補題 5.2 閉ジャクソンネットワークの積形式解 π_{J} (式 (5.1.2)) は対数分離凹関数である．

証明. 式 (5.1.1) で定義される関数 α_i が対数凹関数であること，すなわち

$$\ln \alpha_i(z) - \ln \alpha_i(z-1) \geq \ln \alpha_i(z+1) - \ln \alpha_i(z) \qquad (\forall z \in \mathbb{Z}_{++}) \tag{5.1.3}$$

が成り立つことを示す．関数 α_i の定義式 (5.1.1) から，

$$\ln \alpha_i(z) = \begin{cases} z \ln\left(\frac{\theta_i}{\mu_i}\right) - \sum_{j=1}^{z} \ln j & (z \leq s_i), \\ z \ln\left(\frac{\theta_i}{\mu_i}\right) - (z - s_i) \ln s_i - \sum_{j=1}^{s_i} \ln j & (z > s_i), \end{cases}$$

が成り立つ．したがって

$$\ln \alpha_i(z) - \ln \alpha_i(z-1) = \begin{cases} \ln\left(\frac{\theta_i}{\mu_i}\right) - \ln z & (z \leq s_i), \\ \ln\left(\frac{\theta_i}{\mu_i}\right) - \ln s_i & (z > s_i), \end{cases}$$

を得る．上記の議論から式 (5.1.3) が成り立ち，積形式解 (5.1.3) は対数分離凹関数である． □

5.1.2 閉ジャクソンネットワーク単一サーバモデル

各 $i \in \{1, \ldots, n\}$ に対して $s_i = 1$ を満たす時，単一サーバモデルと呼ばれる．このとき関数 α_i は，定義式 (5.1.1) から

$$\alpha_i(z) = \left(\frac{\theta_i}{\mu_i}\right)^z \qquad (z \in \{0, \ldots, K\}) \tag{5.1.4}$$

となる．

次の補題は，閉ジャクソンネットワーク単一サーバモデルの積形式解に対して，マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の合流時間の期待値が多項式時間であることを，定理 3.9 に基づいて保証するものである．

0	$f_i(0)f_j(k)/A$	$f_i(1)f_j(k-1)/A$	$\cdots$	$f_i(k)f_j(0)/A$	1
0	$f_i(0)f_j(k+1)/A'$	$f_i(1)f_j(k)/A'$	$f_i(2)f_j(k-1)/A'$	$\cdots$	$f_i(k+1)f_j(0)/A'$	1

図 7　再掲: 添え字対 (i,j) および非負整数 k に対する交互不等式の図．図中で $A \overset{\text{def.}}{=} \sum_{s=0}^{k} f_i(s)f_j(k-s)$ および $A' \overset{\text{def.}}{=} \sum_{s=0}^{k+1} f_i(s)f_j(k+1-s)$ は正規化定数である．

補題 5.3 関数ベクトル $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ の添え字の順が

$$\frac{\theta_1}{\mu_1} \geq \frac{\theta_2}{\mu_2} \geq \cdots \geq \frac{\theta_n}{\mu_n}$$

を満足する時，条件 3.8 が成り立つ．

証明. 以下，記法の簡便のため $\rho_i \overset{\text{def.}}{=} \theta_i/\mu_i$，$\rho_j \overset{\text{def.}}{=} \theta_j/\mu_j$ とする．相異なる添え字の対 $i,j \in \binom{\{1,\ldots,n\}}{2}$ に対して $\rho_i \geq \rho_j$ が成り立つ時 ,任意の $k \in \{1,\ldots,K-1\}$ に対して条件 3.8 の式 (3.3.1)

$$\sum_{l=0}^{k} \left(g_{ij}^{k}(l) - g_{ij}^{k+1}(l)\right) \geq \frac{1}{2}$$

が成り立つことを示せば十分である．関数 $g_{ij}^{k}(k) = 1$ であることを考慮すると，

$$\sum_{l=0}^{k} \left(g_{ij}^{k}(l) - g_{ij}^{k+1}(l)\right) + \sum_{l=0}^{k} \left(g_{ij}^{k+1}(l) - g_{ij}^{k}(l-1)\right) = g_{ij}^{k}(k) = 1$$

が成り立つので，式 (3.3.1) は

$$\sum_{l=0}^{k} \left(g_{ij}^{k}(l) - g_{ij}^{k+1}(l)\right) - \sum_{l=0}^{k} \left(g_{ij}^{k+1}(l) - g_{ij}^{k}(l-1)\right) \geq 0 \tag{5.1.5}$$

と等価である (図 7 参照). 以下，この不等式を扱う．

式 (5.1.5) の左辺を変形すると，

$$\begin{aligned}
&\textstyle\sum_{l=0}^{k} \left(g_{ij}^{k}(l) - g_{ij}^{k+1}(l)\right) - \sum_{l=0}^{k} \left(g_{ij}^{k+1}(l) - g_{ij}^{k}(l-1)\right) \\
&= \textstyle\sum_{l=0}^{k} \left(g_{ij}^{k}(l) - g_{ij}^{k+1}(l)\right) - \sum_{l=0}^{k} \left(g_{ij}^{k+1}(k-l) - g_{ij}^{k}(k-l-1)\right) \\
&= \textstyle\sum_{l=0}^{k} \left(g_{ij}^{k}(l) - g_{ij}^{k+1}(l) - g_{ij}^{k+1}(k-l) + g_{ij}^{k}(k-l-1)\right)
\end{aligned} \tag{5.1.6}$$

が得られる．足し算の各項に注目すると，$\rho_i \geq \rho_j$ の仮定より，$l = k$ に対しては

$$\begin{aligned}
g_{ij}^{k}(k) - g_{ij}^{k+1}(k) - g_{ij}^{k+1}(0) + g_{ij}^{k}(-1) &= 1 - g_{ij}^{k+1}(k) - g_{ij}^{k+1}(0) \\
&= 1 - \frac{\sum_{s=0}^{k} \rho_i^s \rho_j^{k+1-s}}{\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}} - \frac{\sum_{s=0}^{0} \rho_i^s \rho_j^{k+1-s}}{\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}} \\
&= \frac{\rho_i^{k+1}}{\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}} - \frac{\rho_j^{k+1}}{\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}} \geq 0
\end{aligned}$$

成り立つ．同様に，各 l $(0 \leq l \leq k-1)$ に対して

$$
\begin{aligned}
& g_{ij}^{k}(l) - g_{ij}^{k+1}(l) - g_{ij}^{k+1}(k-l) + g_{ij}^{k}(k-l-1) \\
&= g_{ij}^{k}(l) - g_{ij}^{k+1}(l) - \frac{\sum_{s=0}^{k-l} \rho_i^s \rho_j^{k+1-s}}{\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}} + \frac{\sum_{s=0}^{k-l-1} \rho_i^s \rho_j^{k-s}}{\sum_{s=0}^{k} \rho_i^s \rho_j^{k-s}} \\
&= g_{ij}^{k}(l) - g_{ij}^{k+1}(l) - \left(1 - \frac{\sum_{s=k-l+1}^{k+1} \rho_i^s \rho_j^{k+1-s}}{\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}}\right) + \left(1 - \frac{\sum_{s=k-l}^{k} \rho_i^s \rho_j^{k-s}}{\sum_{s=0}^{k} \rho_i^s \rho_j^{k-s}}\right) \\
&= \frac{\sum_{s=0}^{l} \rho_i^s \rho_j^{k-s}}{A_{ij}^{k}} - \frac{\sum_{s=0}^{l} \rho_i^s \rho_j^{k+1-s}}{A_{ij}^{k+1}} + \frac{\sum_{s=k-l+1}^{k+1} \rho_i^s \rho_j^{k+1-s}}{A_{ij}^{k+1}} - \frac{\sum_{s=k-l}^{k} \rho_i^s \rho_j^{k-s}}{A_{ij}^{k}} \\
&= \frac{\sum_{s=0}^{l} \rho_i^s \rho_j^{k-s}}{A_{ij}^{k}} - \frac{\sum_{s=0}^{l} \rho_i^s \rho_j^{k+1-s}}{A_{ij}^{k+1}} + \frac{\sum_{s=0}^{l} \rho_i^{k+1-s} \rho_j^s}{A_{ij}^{k+1}} - \frac{\sum_{s=0}^{l} \rho_i^{k-s} \rho_j^s}{A_{ij}^{k}} \\
&= \left(\frac{1}{A_{ij}^{k}} - \frac{\rho_j}{A_{ij}^{k+1}}\right) \sum_{s=0}^{l} \rho_i^s \rho_j^{k-s} + \left(\frac{\rho_i}{A_{ij}^{k+1}} - \frac{1}{A_{ij}^{k}}\right) \sum_{s=0}^{l} \rho_i^{k-s} \rho_j^s \\
&= \frac{\sum_{s=0}^{l} \rho_i^s \rho_j^{k-s}}{A_{ij}^{k} A_{ij}^{k+1}} \left(\sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s} - \sum_{s=0}^{k} \rho_i^s \rho_j^{k+1-s}\right) \\
&\quad + \frac{\sum_{s=0}^{l} \rho_i^{k-s} \rho_j^s}{A_{ij}^{k} A_{ij}^{k+1}} \left(\sum_{s=1}^{k+1} \rho_i^s \rho_j^{k+1-s} - \sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}\right) \\
&= \frac{1}{A_{ij}^{k} A_{ij}^{k+1}} \left(\rho_i^{k+1} \sum_{s=0}^{l} \rho_i^s \rho_j^{k-s} - \rho_j^{k+1} \sum_{s=0}^{l} \rho_i^{k-s} \rho_j^s\right) \\
&= \frac{1}{A_{ij}^{k} A_{ij}^{k+1}} \sum_{s=0}^{l} \left(\rho_i^{k+1+s} \rho_j^{k-s} - \rho_i^{k-s} \rho_j^{k+1+s}\right) \\
&= \frac{1}{A_{ij}^{k} A_{ij}^{k+1}} \sum_{s=0}^{l} \left(\rho_i^{k-s} \rho_j^{k-s} \left(\rho_i^{2s+1} - \rho_j^{2s+1}\right)\right) \geq 0
\end{aligned}
$$

が成り立つ．ただし，上記において $A \stackrel{\text{def.}}{=} \sum_{s=0}^{k} \rho_i^s \rho_j^{k-s}$ および $A' \stackrel{\text{def.}}{=} \sum_{s=0}^{k+1} \rho_i^s \rho_j^{k+1-s}$ とする．以上の議論から題意が得られる． □

特に単一サーバモデルの場合，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ およびマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の一回の推移にかかる計算時間は明らかに $\mathrm{poly}(\ln(K))$ であるため，以下が得られる．

i) 系 3.3 に基づく多項式時間の近似サンプリング法．
ii) 系 3.7，定理 3.9 に基づく多項式時間の完璧サンプリング法．
iii) 第 4 章のアルゴリズム 4.1 およびアルゴリズム 4.2 基づく，正規化定数 C_{J} の計算に対する多項式時間乱択近似スキーム (FPRAS)．

5.2 分割表

5.2.1 2 行分割表の一様サンプリング法

2 つのベクトル $\boldsymbol{r} = (r_1, r_2) \in \mathbb{Z}_{++}^2$ と $\boldsymbol{s} = (s_1, \ldots, s_n) \in \mathbb{Z}_{++}^n$ は，$\sum_{i=1}^{2} r_i = \sum_{j=1}^{n} s_j = N \in \mathbb{Z}_{++}$ を満たすとする．行和および列和ベクトル $(\boldsymbol{r}, \boldsymbol{s})$ を持つ $2 \times n$ 分割表の集合 Ξ を

$$
\Xi \stackrel{\text{def.}}{=} \left\{ X \in \mathbb{Z}_{+}^{2 \times n} \;\middle|\; \begin{array}{l} \sum_{j=1}^{n} X[i,j] = r_i \ (1 \leq \forall i \leq 2), \\ \sum_{i=1}^{2} X[i,j] = s_j \ (1 \leq \forall j \leq n) \end{array} \right\}
$$

と定義する．ただし，$X[i,j]$ は i 行 j 列の値を表す．また，集合 Ξ' を

$$\Xi' \stackrel{\text{def.}}{=} \{\boldsymbol{x} = (x_1, \ldots, x_n) \in \Delta(r_1) \mid 0 \leq x_i \leq s_i \ (1 \leq \forall i \leq n)\} \tag{5.2.1}$$

と定義する．この時，$X \in \Xi$ に対して $x_i = X[1,i]$ $(\forall i \in \{1, \ldots, n\})$ を対応させる写像は全単射となっている．

本節の目的は以下の議論である．

i) 2 行分割表の集合 Ξ 上の一様分布に対する，
   a. 近似サンプリング法の多項式性 (系 3.3)，
   b. 単調 CFTP アルゴリズムに基づく多項式時間完璧サンプリング法の設計
      (アルゴリズム 2.2，系 3.7，条件 3.8，定理 3.9))．
ii) 2 行分割表の集合 Ξ の要素数の計算に対する乱択近似スキーム (FPRAS) の設計
   (アルゴリズム 4.1， 4.2，4.3，定理 4.2， 4.3，4.5，4.7)．
iii) 2 行分割表の集合 Ξ 上の多項超幾何分布に対する，
   a. マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の多項式時間収束性 (系 3.3)，
   b. マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の単調性と単調 CFTP アルゴリズムの設計
      (アルゴリズム 2.2，系 3.7)．

このうち，i-a の一様分布に対する $(1/2)n(n-1)\ln K$ 時間の近似サンプリング法は Dyer と Greenhill [27] によって与えられ，iii-a の多項超幾何分布に対する $(1/2)n(n-1)\ln K$ 時間収束のマルコフ連鎖を用いた近似サンプリング法は Matsui, Matsui, Ono [76] によって与えられている．本節では，これらの結果を，本論文の対数分離凹関数の性質を用いて統一的に示す．

以下では，$2 \times n$ 分割表全体 Ξ 上の一様サンプリング，すなわち Ξ' 上の一様サンプリングを提案する．これを実現するために集合 $\Delta(r_1)$ 上の分布 $\pi_{\mathrm{cu}} : \Delta(r_1) \to \mathbb{R}_{++}$ を十分大きな正数 $M \in \mathbb{R}_{++}$ を用いて，

$$\pi_{\mathrm{cu}}(\boldsymbol{x}) = \frac{1}{C_{\mathrm{cu}}} \mathrm{e}^{-M \sum_{i=1}^{n} \max\{0, x_i - s_i\}} \qquad (\boldsymbol{x} \in \Delta(r_1)) \tag{5.2.2}$$

と定義する．ただし，$C_{\mathrm{cu}} \stackrel{\text{def.}}{=} \sum_{\boldsymbol{x} \in \Delta(r_1)} \mathrm{e}^{-M \sum_{i=1}^{n} \max\{0, x_i - s_i\}}$ は正規化定数である．この時，任意の $\boldsymbol{x} \in \Delta(r_1)$ に対して

$$\begin{cases} \pi_{\mathrm{cu}}(\boldsymbol{x}) = \dfrac{1}{C_{\mathrm{cu}}} & (\boldsymbol{x} \in \Xi'), \\ \pi_{\mathrm{cu}}(\boldsymbol{x}) \leq \dfrac{1}{C_{\mathrm{cu}}} \mathrm{e}^{-M} & (\boldsymbol{x} \notin \Xi'), \end{cases}$$

が成り立つ．したがって，確率関数 π_{cu} に従うランダムサンプルが $\boldsymbol{x} \in \Xi'$ なら採択し，$\boldsymbol{x} \notin \Xi'$ なら棄却することにすると，得られるランダムサンプルは Ξ' 上の一様分布に従う．また，M を十分大きくすると $\boldsymbol{x} \notin \Xi'$ の出現確率は十分小さくできる．

補題 5.4 式 (5.2.2) で定義された分布 π_{cu} は対数分離凹である．

証明. 以下，各 $i \in \{1, \ldots, n\}$ に対して，関数 $f_i : \{0, \ldots, r_1\} \to \mathbb{R}_{++}$ を

$$f_i(z) \stackrel{\mathrm{def.}}{=} \mathrm{e}^{-M \max\{0, z - s_i\}}$$

と定義すると，分布 π_{cu} は

$$\pi_{\mathrm{cu}}(\boldsymbol{x}) = \frac{1}{C_{\mathrm{cu}}} \mathrm{e}^{-M \sum_{i=1}^{n} \max\{0, x_i - s_i\}} = \frac{1}{C_{\mathrm{cu}}} \prod_{i=1}^{n} \mathrm{e}^{-M \max\{0, x_i - s_i\}} = \frac{1}{C_{\mathrm{cu}}} \prod_{i=1}^{n} f_i(x_i)$$

と記述される．関数 f_i に対して

$$\ln f_i(z) = -M \max\{0, z - s_i\}$$

が成り立ち，明らかに f_i は対数凹関数である．ゆえに題意を得る． □

5.2.2 2 行分割表の多項超幾何分布に従うランダムサンプリング法

行和 $\boldsymbol{r} \in \mathbb{Z}_{++}^{m}$ と，列和 $\boldsymbol{s} \in \mathbb{Z}_{++}^{n}$ が $\sum_{i=1}^{m} r_i = \sum_{j=1}^{n} s_j = N \in \mathbb{Z}_{+}+$ を満たす時，行和 $\boldsymbol{r}$ および列和 $\boldsymbol{s}$ を満たす $m \times n$ 分割表全体の集合を $\Xi(\boldsymbol{r}, \boldsymbol{s})$ で表し，このとき，$X = (X[i, j])$ に対して，多項超幾何分布は

$$\Pr[X \mid X \in \Xi(\boldsymbol{r}, \boldsymbol{s})] \stackrel{\mathrm{def.}}{=} \frac{\left(\prod_{i=1}^{m} r_i\right)\left(\prod_{j=1}^{n} s_j\right)}{K! \prod_{i=1}^{m} \prod_{j=1}^{n} X[i, j]!} = \frac{1}{C_{\mathrm{cm}}} \prod_{i=1}^{m} \prod_{j=1}^{n} \frac{1}{X[i, j]!} \tag{5.2.3}$$

と与えられる．

特に $2 \times n$ 分割表に対しては，$\boldsymbol{x} \in \Xi'$ に対して

$$\Pr[X \mid X \in \Xi(\boldsymbol{r}, \boldsymbol{s})] = \frac{1}{C} \prod_{i=1}^{n} \frac{1}{x_i!} \frac{1}{(s_i - x_i)!} \tag{5.2.4}$$

と書ける．ただし $0! = 1$ とする．これを Δ 上に拡張した分布

$$\pi_{\mathrm{cm}}(\boldsymbol{x}) \stackrel{\mathrm{def.}}{=} \frac{1}{C_{\mathrm{cm}}} \prod_{j=1}^{n} \frac{\mathrm{e}^{-M \max\{0, x_j - s_j\}}}{x_j!(\max\{0, s_j - x_j\})!} \tag{5.2.5}$$

と定義する．一様分布の場合と同様，任意の $\boldsymbol{x} \in \Delta(r_1)$ に対して

$$\begin{cases} \pi_{\mathrm{cm}}(\boldsymbol{x}) = \dfrac{1}{C_{\mathrm{cm}}} \displaystyle\prod_{j=1}^{n} \frac{1}{x_j!(s_j - x_j)!} & (\boldsymbol{x} \in \Xi'), \\ \pi_{\mathrm{cm}}(\boldsymbol{x}) \leq \dfrac{1}{C_{\mathrm{cm}}} \mathrm{e}^{-M} & (\boldsymbol{x} \notin \Xi'), \end{cases}$$

が成り立つ．したがって，π_{cm} に従うランダムサンプルが $\boldsymbol{x} \in \Xi'$ なら採択し，$\boldsymbol{x} \notin \Xi'$ なら棄却することにすると，得られるランダムサンプルは Ξ' 上の多項超幾何分布に従う．また，M を十分大きくすると，$\boldsymbol{x} \notin \Xi'$ の出現確率は十分小さくできる．

補題 5.5 式 (5.2.5) で定義された分布 π_{cm} は対数分離凹である．

証明. 以下，各 $j \in \{1,\dots,n\}$ に対して，関数 $f_j : \{0,\dots,r_1\} \to \mathbb{R}_{++}$ を

$$f_j(z) \stackrel{\mathrm{def.}}{=} \frac{1}{z!(\max\{0, s_j - z\})!}$$

と定義すると，分布 π_{cm} は

$$\pi_{\mathrm{cm}} = \frac{1}{C_{\mathrm{cm}}} \prod_{j=1}^{n} \left(f_j(x_j) \cdot \mathrm{e}^{-M \max\{0, x_j - s_j\}} \right)$$

と記述される．関数 f_j に対して

$$\ln f_j(z) = \begin{cases} -\sum_{t=1}^{z} \ln t - \sum_{t'=1}^{s_j - z} \ln t' & (0 \le z \le s_j) \\ -\sum_{t=1}^{z} \ln t & (s_j < z \le k) \end{cases}$$

が成り立つ．したがって

$$\ln f_j(z+1) - \ln f_j(z) = \begin{cases} -\ln(z+1) + \ln(s_j - z) & (0 \le z < s_j) \\ -\ln(z+1) & (s_j \le z \le k) \end{cases}$$

は単調非増加関数であるので $\ln f_j$ は凹関数である．いま，補題 5.4 の証明から関数 $\ln(\mathrm{e}^{-M \max\{0, x_j - s_j\}})$ も凹関数である．凹関数の和は凹関数であることから題意を得る． □

5.2.3 2 行分割表の一様サンプリング法の多項式性と近似数え上げ法

第 5.2.1 節で 2 行分割表の一様サンプリング法について述べた．本節では FPRAS を目標とし，2 行分割表の一様サンプリング法の多項式性について議論する．まず，与えられた周辺和を満たす 2 行分割表全体の Ξ' 上の一様サンプリング法を目標として，Ξ' を状態空間として持つマルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ および $\mathcal{M}'_{\mathrm{P}}$ を以下のように定義する．

マルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ の状態 $X \in \Xi'$ に対して，次の時刻の状態 X' は以下のように定められる．まず，相異なる添え字の対 $\{j_1, j_2\} \in \binom{\{1,\dots,n\}}{2}$ を一様ランダムに選ぶ．このとき，$a = X_{j_1} + X_{j_2}$, $b = s_{j_1} + s_{j_2} - a$ とし，$k = 1 + \min\{a, b, s_{j_1}, s_{j_2}\}$ とする．次に $\{1,\dots,k\}$ 上の一様確率変数 L を確率 $1/k$ で発生させる．最後に，次の時刻の状態 X' を

$$X'_i = \begin{cases} \min\{a, s_{j_1}\} - k + L & (i = j_1), \\ a - X'_{j_1} & (i = j_2), \\ X_i & (\text{それ以外}), \end{cases}$$

と定める．このとき，$X' \in \Xi'$ が成り立つ．このマルコフ連鎖は Dyer と Greenhill [27] のものと同じである．マルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ の推移にかかる計算量 $\mathcal{T}$ は 2×2 分割表の個数を数える手間と同じオーダーであり，すなわち $\mathrm{O}(\ln K)$ であることに注意されたい．

同様にマルコフ連鎖 $\mathcal{M}'_{\mathrm{P}}$ の更新関数 ϕ' は現在の状態 $X \in \Xi'$ および一様実数乱数 $\Lambda \in [1, n)$ に対して以下のように定義される．まず $a = X_{\lfloor \Lambda \rfloor} + X_{\lfloor \Lambda \rfloor + 1}$, $b = s_{\lfloor \Lambda \rfloor} + s_{\lfloor \Lambda \rfloor + 1} - a$ とし，$k = 1 + \min\{a, b, s_{\lfloor \Lambda \rfloor}, s_{\lfloor \Lambda \rfloor + 1}\}$ とする．次の時刻の状態 $X' = \phi'(X, \lambda)$ は

$$X'_i = \begin{cases} \min\{a, s_{\lfloor \Lambda \rfloor}\} - k + \lfloor k\Lambda \rfloor + 1 & (i = \lfloor \Lambda \rfloor), \\ a - X'_{\lfloor \Lambda \rfloor} & (i = \lfloor \Lambda \rfloor + 1), \\ X_i & (\text{それ以外}), \end{cases}$$

とする．このとき，$X' \in \Xi'$ が成り立つ．また，マルコフ連鎖 $\mathcal{M}'_{\mathrm{P}}$ の推移にかかる計算量 $\mathcal{T}$ も $\mathrm{O}(\ln K)$ である．

以上のように定義されたマルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ および $\mathcal{M}'_{\mathrm{P}}$ は，$M \to +\infty$ とした場合の分布 $\pi_{\mathrm{cu}}(5.2.2)$ に対する初期状態を $\boldsymbol{x} \in \Xi'$ としたマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ および $\mathcal{M}_{\mathrm{P}}$ と一致する．すなわちマルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ および $\mathcal{M}'_{\mathrm{P}}$ は，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ および $\mathcal{M}_{\mathrm{P}}$ の推移のうち任意の状態対 $(\boldsymbol{x}, \boldsymbol{y}) \in \binom{\Xi'}{2}$ に対する推移だけに制限したものとみることができる．このことを確認する意味で，まずマルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ の既約性について議論する．以下の議論は $\mathcal{M}'_{\mathrm{P}}$ に対しても同様に行うことができる．マルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ および $\mathcal{M}'_{\mathrm{P}}$ の既約性は直接示すことも可能であるが，ここでは $\mathcal{M}_{\mathrm{A}}$ の多項式時間収束性を利用した証明を行う．

命題 5.6 マルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ は既約である．

証明. マルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ が既約でないとして矛盾を導く．もしマルコフ連鎖 $\mathcal{M}'_{\mathrm{A}}$ が既約でないならば，状態対 $\{\boldsymbol{x}, \boldsymbol{y}\} \in \binom{\Xi'}{2}$ が存在して，$\mathcal{M}'_{\mathrm{A}}$ の推移グラフ上には $\boldsymbol{x}$ から $\boldsymbol{y}$ への経路が存在しない．このとき，初期状態 $\boldsymbol{x} \in \Xi'$ のマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ に着目すると，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ がある時刻 $T > 0$ で状態 y へ到達するには，ある時刻 $t < T$ が存在して，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の時刻 t の状態 X が $X \in \Delta \setminus \Xi'$ であることが要請される．

いま，$T = n^2 \ln(\Delta^2/2)$ とし，分布 π_{cu} (式 $(5.2.2)$) に対して $M = \ln(3T|\Delta|)$ とする．補題 5.4 および系 3.3 より，M の値によらず，初期状態 $\boldsymbol{x}$ のマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の時刻 T の全変動距離は $1/(2|\Delta|)$ 以下である．

初期状態 $\boldsymbol{x} \in \Xi'$ のマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ が，時刻 T までに空間 $\Delta \setminus \Xi'$ 中の状態をとる確率は，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の定義から高々 $T \cdot \mathrm{e}^{-M} = 1/(3|\Delta|)$ である．したがって，時刻 T のマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の状態が $\boldsymbol{y}$ である確率も高々 $1/(3|\Delta|)$ である．一方 $\boldsymbol{y} \in \Xi'$ であるから，$\mathcal{M}_{\mathrm{A}}$ の定常分布は $\pi_{\mathrm{cu}}(\boldsymbol{y}) \geq 1/|\Delta|$ を満たす．このことは全変動距離が $1/(2|\Delta|)$ 以下に矛盾する． □

いま $X^{\mathrm{U}} = (X^{\mathrm{U}}_1, \ldots, X^{\mathrm{U}}_n) \in \Xi'$ を，各 $i \in \{1, \ldots, n\}$ に対して $X^{\mathrm{U}}_i = \min\{s_i, r_1 - \sum_{j=1}^{i-1} X^{\mathrm{U}}_j\}$ と定義し，$X^{\mathrm{L}} = (X^{\mathrm{U}}_1, \ldots, X^{\mathrm{U}}_n) \in \Xi'$ を，各 $i \in \{1, \ldots, n\}$ に対して $X^{\mathrm{U}}_i = \min\{s_i, r_1 - \sum_{j=i+1}^{n} X^{\mathrm{U}}_j\}$ と定義する．このとき第 $3.3.1$ 節で累積和ベクトル $(3.3.1)$ を用いて定義した半順序 '$\succeq$' に関して，X^{U}，X^{L} はそれぞれ最大元，最小元であり，すなわち，任意の $\boldsymbol{x} \in \Xi'$ に対して $X^{\mathrm{U}} \succeq \boldsymbol{x} \succeq X^{\mathrm{L}}$ が成り立つ．このとき，$M \to \infty$ とすると，分布 π_{cu} に対するマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の更新関数 ϕ とマルコフ連鎖 $\mathcal{M}'_{\mathrm{P}}$ の更新関数 ϕ' が一致することから次の命題が得られる．

命題 5.7 更新関数 ϕ' で定義されたマルコフ連鎖 $\mathcal{M}'_{\mathrm{P}}$ は単調である． ■

また，命題 5.7 は定理 3.4 と同様の方法で直接示すことも可能である [61]．

次の定理は，更新関数 ϕ' で定義されるマルコフ連鎖 $\mathcal{M}'_{\mathrm{P}}$ の合流時間の期待値が多項式時間であることを意味する．

定理 5.8 列和ベクトル $\boldsymbol{s} = (s_1, \ldots, s_n)$ の添え字順が $s_1 \geq s_2 \geq \cdots \geq s_n$ を満たす時，更新関数 ϕ' で定義されるマルコフ連鎖 $\mathcal{M}'_{\mathrm{P}}$ の合流時間 T^* は

$$\mathrm{E}[T^*] \leq 2n^2(n-1)(1+\ln(Kn))$$

を満足する．

定理 5.8 は，定理の条件が条件 3.8 を示すことで定理 3.9 に基づいて示すことができるが，定理 5.8 の条件が条件 3.8 を満たすことを示すには複雑な場合分けを要する．ここでは，定理 5.8 に対する比較的簡単で直接的な証明を与える．この証明は本質的に定理 3.9 の証明と同じであることを明記しておく．なお，定理 5.8 の条件は $\mathrm{O}(n \ln n)$ で得られることに注意されたい．

証明. 単純無向グラフ $\mathcal{G} = (\Xi', \mathcal{E})$ を以下のように定義する．グラフ $\mathcal{G}$ は頂点集合 Ξ' を持ち，以下に定義される枝集合 $\mathcal{E}$ を持つ．頂点対 $\{x, y\}$ がグラフ $\mathcal{G}$ の枝であることの必要十分条件は $(1/2)\sum_{i=1}^{n} |x_i - y_i| = 1$ を満足することである．グラフ $\mathcal{G}$ は連結である．各枝 $e = \{x, y\} \in \mathcal{E}$ に対して，唯一の添え字対 $j_1, j_2 \in \{1, 2, \ldots, n\}$ が存在して

$$|x_i - y_i| = \begin{cases} 1 & (i = j_1, j_2), \\ 0 & (\text{それ以外}), \end{cases}$$

を満足する．この添え字対を e の支持対と呼ぶ．支持対 $\{j_1, j_2\}$ をもつ枝 $e \in \mathcal{E}$ の長さ $\ell_{\mathrm{P}}(e)$ を $j^* = \max\{j_1, j_2\}$ を用いて

$$\ell_{\mathrm{P}}(e) \stackrel{\mathrm{def.}}{=} \frac{1}{n-1} \sum_{i=1}^{j^*-1} (n-i) = \sum_{i=1}^{j^*-1} \left(1 - \frac{i-1}{n-1}\right)$$

と定義する．このとき，$1 \leq \min_{e \in \mathcal{E}} \ell_{\mathrm{P}}(e) \leq \max_{e \in \mathcal{E}} \ell_{\mathrm{P}}(e) \leq n/2$ が成り立つことに注意されたい．任意の頂点対 $\{x, y\} \in \binom{\Xi'}{2}$ に対して距離 $d_{\mathrm{P}}(x, y) (= d_{\mathrm{P}}(y, x))$ を，グラフ $\mathcal{G}$ 上での x と y の間の ℓ_{P} に関する最短パスの長さと定義する．このとき，任意の頂点対 $\{x, y\} \in \binom{\Xi'}{2}$ に対して $d_{\mathrm{P}}(x, y) \leq (n/2)\sum_{i=1}^{n}(1/2)|x_i - y_i| \leq (n/2)K$ が成り立つ．すなわち，グラフ $\mathcal{G}$ の直径 $\max_{(X,Y) \in (\Xi')^2}\{d_{\mathrm{P}}(X, Y)\}$ は $Kn/2$ で上から押さえられる．また，距離の定義から任意の枝 $\{x, y\} \in \mathcal{E}$ に対して $d_{\mathrm{P}}(x, y) = \ell_{\mathrm{P}}(\{x, y\})$ が成り立つ．

以下，任意の枝 $\{X, Y\} \in \mathcal{E}$ に対して

$$\mathrm{E}[d_{\mathrm{P}}(\phi_{\mathrm{P}}(X, \Lambda), \phi_{\mathrm{P}}(Y, \Lambda))] \leq \beta \cdot d_{\mathrm{P}}(X, Y), \quad \text{かつ} \quad \beta = 1 - \frac{1}{n(n-1)^2},$$

が成り立つことを示す．なお ,枝 $\{X, Y\} \in \mathcal{E}$ の支持対を $\{j_1, j_2\}$ とし ,一般性を失うことなく $j_1 < j_2$ かつ $X_{j_2} + 1 = Y_{j_2}$ が成り立つとする．また ,簡単のため $X' = \phi(X, \lambda)$ および $Y' = \phi(Y, \lambda)$ と表記する．以下 ,$a_X \stackrel{\text{def.}}{=} X_{\lfloor\Lambda\rfloor} + X_{\lfloor\Lambda\rfloor+1}$, $a_Y \stackrel{\text{def.}}{=} Y_{\lfloor\Lambda\rfloor} + Y_{\lfloor\Lambda\rfloor+1}$, $b_X \stackrel{\text{def.}}{=} s_{\lfloor\Lambda\rfloor} + s_{\lfloor\Lambda\rfloor+1} - a_X$, $b_Y \stackrel{\text{def.}}{=} s_{\lfloor\Lambda\rfloor} + s_{\lfloor\Lambda\rfloor+1} - a_Y$, $k_X \stackrel{\text{def.}}{=} 1 + \min\{a_X, b_X, s_{\lfloor\Lambda\rfloor}, s_{\lfloor\Lambda\rfloor+1}\}$, $k_Y \stackrel{\text{def.}}{=} 1 + \min\{a_Y, b_Y, s_{\lfloor\Lambda\rfloor}, s_{\lfloor\Lambda\rfloor+1}\}$ とする．

場合 **1:** $\lfloor\Lambda\rfloor = j_2 - 1$ の場合について，

$$\mathrm{E}[d(X', Y') \mid \lfloor\Lambda\rfloor = j_2 - 1] \leq d_{\mathrm{P}}(X, Y) - \frac{1}{2} \cdot \frac{n - j_2 + 1}{n - 1}$$

を示す．

i. $j_1 = j_2 - 1$ の時，定理の仮定より $X' = Y'$ である．ゆえに $d_{\mathrm{P}}(X', Y') = 0$．
ii. $j_1 \neq j_2 - 1$ かつ $k_X = k_Y$ の時．定理の仮定より $s_{j_2-1} \geq s_{j_2}$ である．いま，$k_X = k_Y$ かつ $a_Y > a_X$ かつ $b_Y < b_X$ なので $k_X = k_Y = \min\{s_{j_2-1}, s_{j_2}\} = s_{j_2}$ となる．したがって $s_{j_2-1} \geq a_Y > a_X \geq s_{j_2}$ となり関数 ϕ' の定義より $X'_{j_2-1} = a_X - (\Lambda - \lfloor\Lambda\rfloor)k_X$ と $Y'_{j_2-1} = a_Y - (\Lambda - \lfloor\Lambda\rfloor)k_Y$ を得る．この時 $a_X = a_Y - 1$ から $X'_{j_2-1} = Y'_{j_2-1} - 1$ である．また，$X'_{j_2} = a_X - X'_{j_2-1}$ かつ $Y'_{j_2} = a_Y - Y'_{j_2-1}$ より $X'_{j_2} = Y'_{j_2}$ である．ゆえにこの条件のもと確率 1 で $d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y) - (n - j_2 + 1)/(n - 1)$ である．
iii. $j_1 \neq j_2 - 1$ かつ $k_X \neq k_Y$ の時．明らかに，$|k_X - k_Y| = 1$ である．まず，$k_X = k_Y - 1$ の場合を考える．この時，2 つの場合を考えれば良い．$\lfloor(\Lambda - \lfloor\Lambda\rfloor)k_X\rfloor = \lfloor(\Lambda - \lfloor\Lambda\rfloor)k_Y\rfloor$ と $\lfloor(\Lambda - \lfloor\Lambda\rfloor)k_X\rfloor = \lfloor(\Lambda - \lfloor\Lambda\rfloor)k_Y\rfloor - 1$ の 2 つの場合を考えれば十分である．前者の場合，$X'_{j_2-1} = Y'_{j_2-1}$ かつ $X'_{j_2} = Y'_{j_2} - 1$ なので，$d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y)$ である．後者の場合，$X'_{j_2-1} = Y'_{j_2-1} - 1$ かつ $X'_{j_2} = Y'_{j_2}$ なので，$d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y) - (n - j_2 + 1)/(n - 1)$ である．2 つの場合は確率 $1/2$ で現れるので，$\mathrm{E}[d(X', Y') \mid \lfloor\Lambda\rfloor = j_2 - 1, j_1 \neq j_2 - 1, k_X = k_Y - 1] = d_{\mathrm{P}}(X, Y) - (1/2)(n - j_2 + 1)/(n - 1)$ である．
iv. $j_1 \neq j_2 - 1$ かつ $k_X = k_Y + 1$ の時．iii と同様に示される．

場合 **2:** $\lfloor\Lambda\rfloor = j_2$ の場合について，

$$\mathrm{E}[d_{\mathrm{P}}(X', Y') \mid \lfloor\Lambda\rfloor = j_2] \leq d_{\mathrm{P}}(X, Y) + \frac{1}{2} \cdot \frac{n - j_2}{n - 1}$$

を示す．

i. $k_X = k_Y$ の時，1-i の場合と同様に $X'_{j_2} = Y'_{j_2} - 1$ かつ $X'_{j_2+1} = Y'_{j_2+1}$ を得る．ゆえにこの条件の下確率 1 で $d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y)$ である．
ii. $k_X \neq k_Y$ の時，1-iii の場合と同様に条件付確率 $1/2$ で $d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y)$ と条件付確率 $1/2$ で $d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y) + (n - j_2)/(n - 1)$ を示すことができる．ゆえにこの場合 $\mathrm{E}[d_{\mathrm{P}}(X', Y') \mid \lfloor\Lambda\rfloor = j_2, k_X \neq k_Y] = d_{\mathrm{P}}(X, Y) + (1/2)(n - j_2)/(n - 1)$ となる．

場合 **3:** $\lfloor \Lambda \rfloor \neq j_2 - 1$ かつ $\lfloor \Lambda \rfloor \neq j_2$ の場合．この時，$\{X', Y'\}$ もまた G の枝となる．いま，$\{j_1', j_2'\}$ を $\{X',\ Y'\}$ の支持対とすると，明らかに $j_2 = \max\{j_1', j_2'\}$ である．したがって $d_{\mathrm{P}}(X', Y') = d_{\mathrm{P}}(X, Y)$ を得る．

場合 1 の生起確率が $1/(n-1)$ であり，場合 2 の生起確率が $1/(n-1)$ 以下であることから，任意の枝 $\{X, Y\} \in \mathcal{E}$ に対して

$$\begin{aligned}
\mathrm{E}[d_{\mathrm{P}}(X', Y')] &\leq d_{\mathrm{P}}(X, Y) - \frac{1}{n-1}\frac{1}{2}\frac{n - j_2 + 1}{n-1} + \frac{1}{n-1}\frac{1}{2}\frac{n - j_2}{n-1} \\
&= d_{\mathrm{P}}(X, Y) - \frac{1}{2(n-1)^2} \\
&\leq \left(1 - \frac{1}{2(n-1)^2}\frac{1}{\max_{\{X,Y\}\in\mathcal{E}}\{d_{\mathrm{P}}(X, Y)\}}\right) d_{\mathrm{P}}(X, Y) \\
&= \left(1 - \frac{1}{n(n-1)^2}\right) d_{\mathrm{P}}(X, Y).
\end{aligned}$$

を得る．

以上の議論から，定理 3.9 の証明と同様に題意が導かれる． □

5.2.4 2 行分割表の近似数え上げ法に関する計算機実験

2 行分割表に対して Dyer と Greenhill のアルゴリズムを適用したものと，アルゴリズム 4.3 のアイデアに基づくアルゴリズムに対して，計算機実験を行い比較した．ここで用いたインスタンスは行和ベクトルが $(42, 38)$，列和ベクトルが $(10, \ldots, 10)$ の 2×8 分割表である．このインスタンスの厳密解は 9,162,736 である．実験では各々のアルゴリズムについて 10,000 回の試行を行った．図 8 は実験結果のヒストグラムである．図において D-G は Dyer と Greenhill のアルゴリズムによる結果を，K-M はアルゴリズム 4.3 のアイデアを用いた結果を表している．この実験により，最大領域が簡単に判定できる場合には，アルゴリズム 4.3 のアイデアに基づくアルゴリズムの方が偏りが小さくなることがわかる．

5.3 離散化 Dirichlet 分布に従うランダムサンプリング法

5.3.1 離散化 Dirichlet 分布

離散化ディリクレ分布はバイオインフォマティクスの分野で統計的手法などにしばしば現れる．パラメータ (ディリクレパラメータ) $(u_1, u_2, \ldots, u_n) \in \mathrm{R}_+^n$ が与えられたとき，$\Xi_{++} \stackrel{\mathrm{def.}}{=} \{x = (x_1, \ldots, x_n) \in \mathbb{Z}_{++} \mid \sum_{i=1}^n x_i = K\}$ $(K \geq n)$ 上の離散化ディリクレ分布 π_{D} は

$$\pi_{\mathrm{D}}(\boldsymbol{x}) \stackrel{\mathrm{def.}}{=} \frac{1}{C_{\mathrm{D}}} \prod_{i=1}^n (x_i/K)^{u_i - 1} \qquad (\boldsymbol{x} \in \Xi_{++}), \tag{5.3.1}$$

と定義される．ただし，C_{D} は正規化定数である．パラメータが $u_i \geq 1$ を満たすとき，π_{D} は対数分離凹関数となる．(連続の) ディリクレ分布に対しては棄却サンプリングを行う方法があるが，

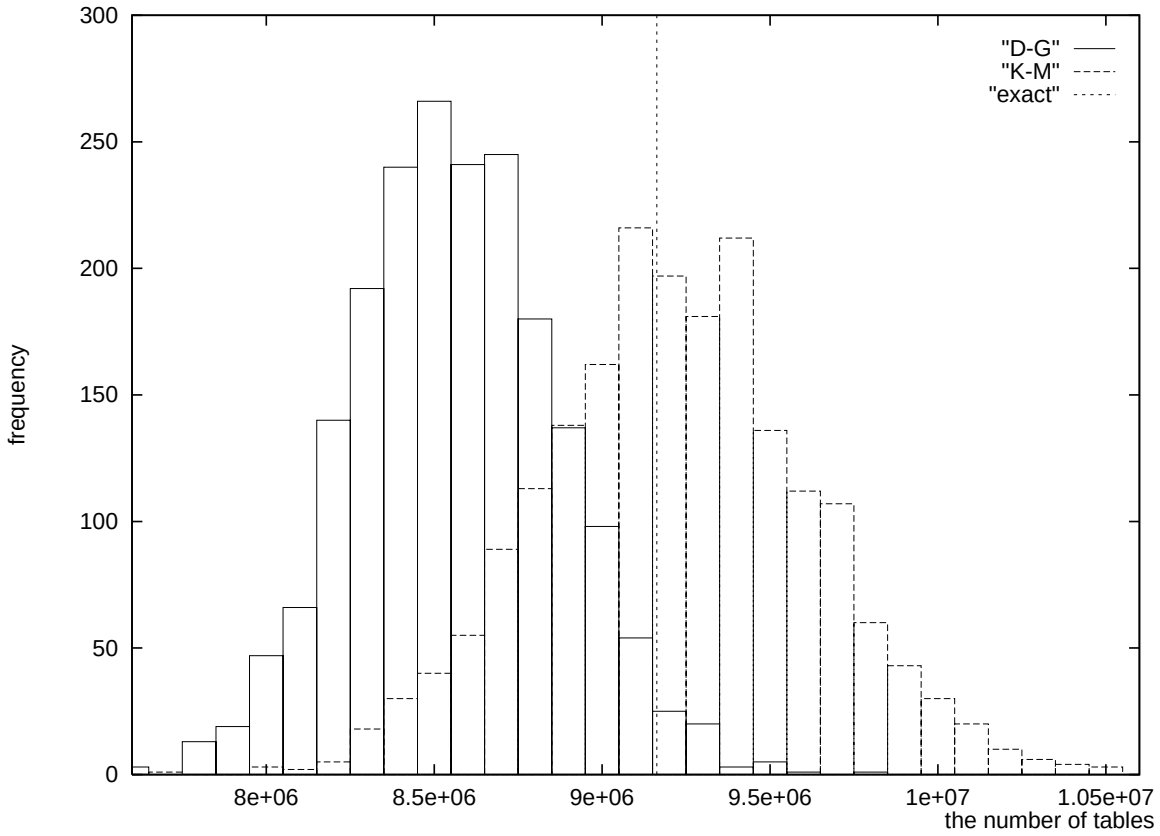


図 8　2 行分割表の近似数え上げ法に対する計算機実験のヒストグラム．

パラメータの値が非常に小さいとき，棄却率が非常に大きくなり，非効率的になることが知られている．

$K \geq n$ が要求されるので，以下，記法を容易にするために $\boldsymbol{x} = \boldsymbol{x} - \mathbf{1} \in \Delta(K)$ に対して

$$\pi'_{\mathrm{D}} \stackrel{\mathrm{def.}}{=} \frac{1}{C'_{\mathrm{D}}} \prod_{i=1}^{n} (x_i + 1)^{u_i - 1} \qquad (x \in \Delta(K - n)), \tag{5.3.2}$$

という分布を扱う．明らかにこの分布を扱うことで離散化 Dirichlet 分布 5.3.1 と等価な変形が存在する．明らかに $\boldsymbol{x} \in \Xi_{++}(K + n)$ と $\boldsymbol{y} = \boldsymbol{x} - \mathbf{1} \in \Delta(K)$ には一対一対応が存在する．また，

$$\pi_{\mathrm{D}}(\boldsymbol{x} + \mathbf{1}) = \pi'_{\mathrm{D}}(\boldsymbol{x}) \qquad \boldsymbol{x} \in \Delta(K)$$

が成り立つ．

本節の目的は，離散化 Dirichlet 分布に対する以下の議論である．

i)　マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の多項式時間収束性 (系 3.3)，

ii) マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の単調性と単調 CFTP アルゴリズムの設計
(アルゴリズム 2.2，系 3.7)．

iii) マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の合流時間の期待値の多項式性 (条件 3.8，定理 3.9)．

このうち ,i の $(1/2)n(n-1)\ln K$ 時間収束のマルコフ連鎖を用いた近似サンプリング法は Matsui, Motoki, Kamatani [77] によって与えられている．

5.3.2 対数分離凹分布になる場合

補題 5.9 Dirichlet パラメータが各 $i \in \{1,\dots,n\}$ に対して $u_i \geq 1$ を満たす時，式 (5.2.5) で定義された分布 π'_{D} は対数分離凹である．

証明. 以下，各 $j \in \{1,\dots,n\}$ に対して，関数 $f_j : \{0,\dots,r_1\} \to \mathbb{R}_{++}$ を

$$f_j(z) \stackrel{\mathrm{def.}}{=} (z+1)^{u_i-1}$$

と定義すると，分布 π'_{D} は

$$\pi'_{\mathrm{D}} = \frac{1}{C'_{\mathrm{D}}} \prod_{j=1}^{n} f_j(x_j)$$

と記述される．関数 f_j に対して

$$\ln f_j(z) = (u_j - 1)\ln(z+1)$$

が成り立つ．明らかに $\ln(z+1)$ は凹関数であり仮定より $u_j \geq 1$ なので，関数 f_j は対数凹関数である．ゆえに題意を得る． □

5.3.3 一般の場合

Dirichlet 分布はパラメータ $0 < p_i < 1$ を含む場合 ,対数 (分離) 凹関数にならない．しかし ,交互不等式 (3.2.2) を満たすので，定理 3.2 および定理 3.4 が成り立つ．すなわちマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ は $\mathrm{O}(n^2 \ln K)$ で収束し，マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ は単調マルコフ連鎖となる．

以下，一般の Dirichlet 分布 π_{D} に対して交互不等式が成り立つことを示す．

補題 5.10 Dirichlet パラメータ $(u_1,\dots,u_n) \in \mathbb{Z}_+$ が与えられたとき，関数ベクトル $\boldsymbol{f} = (f_1,\dots,f_n)$ は各 $i \in \{1,\dots,n\}$ に対して $f_i \stackrel{\mathrm{def.}}{=} (z+1)^{u_i-1}$ とする．このとき，任意の $k \in \mathbb{Z}_+$ および任意の対 f_i, f_j に関して交互不等式 (3.2.2) が成り立つ．

証明. 以下では $k \geq 1$ について，交互不等式 (3.2.3) の 2 番目の不等式にあたる

$$g_{ij}^{k-1}(l-1) \leq g_{ij}^{k}(l) \tag{5.3.3}$$

を示す．1 番目の不等式は同様に示すことができる．簡単のため

$$\begin{aligned} A_{k-1} &= \textstyle\sum_{s=0}^{k-1} f_i(s) f_j(k-s-1) &&= \textstyle\sum_{s=1}^{k} s^{u_i-1}(k-s)^{u_j-1}, \\ A_k &= \textstyle\sum_{s=0}^{k} f_i(s) f_j(k-s) &&= \textstyle\sum_{s=1}^{k+1} s^{u_i-1}(k-s+1)^{u_j-1}, \end{aligned}$$

と表す．いま g_{ij}^k の定義から

$$\begin{aligned}
H(s) &\overset{\text{def.}}{=} g_{ij}^k(l) - g_{ij}^{k-1}(l-1) \\
&= \sum_{s=1}^{l} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} - \sum_{s=1}^{l-1} \frac{s^{u_i-1}(k-s)^{u_j-1}}{A_{k-1}} \\
&= \left(1 - \sum_{s=l+1}^{k} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k}\right) - \left(1 - \sum_{s=l}^{k-1} \frac{s^{u_i-1}(k-s)^{u_j-1}}{A_{k-1}}\right) \\
&= \sum_{s=l+1}^{k} \frac{(s-1)^{u_i-1}(k-s+1)^{u_j-1}}{A_{k-1}} - \sum_{s=l+1}^{k} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} \\
&= \sum_{s=l+1}^{k} \left(\frac{(s-1)^{u_i-1}(k-s+1)^{u_j-1}}{A_{k-1}} - \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k}\right) \\
&= \sum_{s=l+1}^{k} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} \left(\left(1-\frac{1}{s}\right)^{u_i-1} - \frac{A_k}{A_{k-1}}\right)
\end{aligned}$$

を得る．同様に

$$\begin{aligned}
H(s) &= \sum_{s=1}^{l} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} - \sum_{s=1}^{l-1} \frac{s^{u_i-1}(k-s)^{u_j-1}}{A_{k-1}} \\
&\geq \sum_{s=2}^{l} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} - \sum_{s=2}^{l} \frac{(s-1)^{u_i-1}(k-s+1)^{u_j-1}}{A_{k-1}} \\
&= \sum_{s=2}^{l} \left(\frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} - \frac{(s-1)^{u_i-1}(k-s+1)^{u_j-1}}{A_{k-1}}\right) \\
&= \sum_{s=2}^{l} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} \left(\frac{A_k}{A_{k-1}} - \left(1-\frac{1}{s}\right)^{u_i-1}\right)
\end{aligned}$$

も示される．関数 $\psi : \{2,3,\ldots,k\} \to \mathbb{R}$ を導入し，$\psi(s) \overset{\text{def.}}{=} \left(1-\frac{1}{s}\right)^{u_i-1} - \frac{A_k}{A_{k-1}}$ と定義すると，次の不等式が得られる．

$$H(s) = \sum_{s=l+1}^{k} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} \psi(s) \tag{5.3.4}$$

$$\geq -\sum_{s=2}^{l} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k} \psi(s) \tag{5.3.5}$$

場合 1: $u_i \geq 1$ の場合を考える．

いま $u_i - 1 \geq 0$ なので $\psi(s)$ は単調非減少関数である．したがって $\psi(s) \geq 0$ が成り立てば $0 \leq \psi(s) \leq \psi(s+1) \leq \cdots \leq \psi(k)$ となり，等式 (5.3.4) から $H(s) \geq 0$ が言える．もし $\psi(s) < 0$

なら $\psi(2) \leq \psi(3) \leq \cdots \leq \psi(s) < 0$ となるので不等式 (5.3.5) を用いて

$$H(s) \geq -\sum_{s=2}^{l} \frac{s^{u_i-1}(k-s+1)^{u_j-1}}{A_k}\psi(s) \geq 0$$

が得られる．

場合 2: $0 \leq u_i \leq 1$ の場合を考える．

この場合 $u_i - 1 \leq 0$ なので $\psi(s)$ は非増加関数である．もし不等式 $\psi(k-1) \geq 0$ が成り立てば，$\psi(2) \geq \psi(3) \geq \cdots \geq \psi(k) \geq 0$ となり等式 (5.3.4) から $H(s) \geq 0$ を得る．したがって

$$\psi(k) = \left(\frac{k-1}{k}\right)^{u_i-1} - \frac{A_k}{A_{k-1}} \geq 0$$

を示せば十分である．

以下この証明において簡単のため $\alpha_{i'} = u_{i'} - 1\ (\forall i')$ で表す．関数 $H^1_{ij}(k-1)$ を

$$H^1_{ij}(k) \stackrel{\text{def.}}{=} (k-1)^{\alpha_i}A_k - k^{\alpha_i}A_{k-1}$$

と定義する．もし条件

$$-1 \leq \forall\alpha_i \leq 0,\ -1 \leq \forall\alpha_j,\ \forall k \in \{2,3,4,\ldots\},\ H^1_{ij}(k) \geq 0$$

が成り立てば，目標の各 $k \in \{2,3,4,\ldots\}$ に対して $\psi(k) \geq 0$ を得る．以下 $H^1_{ij}(k)$ を式変形する．

$$\begin{aligned}
H^1_{ij}(k) &= (k-1)^{\alpha_i}\sum_{s=1}^{k} s^{\alpha_i}(k-s+1)^{\alpha_j} - k^{\alpha_i}\sum_{s=1}^{k-1} s^{\alpha_i}(k-s)^{\alpha_j} \\
&= \sum_{s=1}^{k}(k-1)^{\alpha_i}s^{\alpha_i}(k-s+1)^{\alpha_j}\frac{(k-s)+(s-1)}{k-1} - k^{\alpha_i}\sum_{s=1}^{k-1} s^{\alpha_i}(k-s)^{\alpha_j} \\
&= \sum_{s=1}^{k-1}\Bigg[(k-1)^{\alpha_i}s^{\alpha_i}(k-s+1)^{\alpha_j}\left(\frac{k-s}{k-1}\right) \\
&\qquad +(k-1)^{\alpha_i}(s+1)^{\alpha_i}(k-s)^{\alpha_j}\left(\frac{s}{k-1}\right) - k^{\alpha_i}s^{\alpha_i}(k-s)^{\alpha_j}\Bigg] \\
&= \sum_{s=1}^{k-1}\frac{(k-1)^{\alpha_i}s^{\alpha_i}(k-s)^{\alpha_j}}{k-1} \\
&\qquad \cdot\left[\left(1+\frac{1}{k-s}\right)^{\alpha_j}(k-s) + \left(1+\frac{1}{s}\right)^{\alpha_i}s - \left(\frac{k}{k-1}\right)^{\alpha_i}(k-1)\right]
\end{aligned}$$

が得られる．以上から，関数

$$H^2(k,\alpha_i,\alpha_j,s) \stackrel{\text{def.}}{=} \left(1+\tfrac{1}{k-s}\right)^{\alpha_j}(k-s) + \left(1+\tfrac{1}{s}\right)^{\alpha_i}s - \left(\tfrac{k}{k-1}\right)^{\alpha_i}(k-1)$$

が任意の $s \in \{1,2,\ldots,k-1\}$ に対して非負であることを示せば十分である．いま $1+1/(k-s) > 1$ かつ $\alpha_j \geq -1$ より，

$$H^2(k,\alpha_i,\alpha_j,s) \geq H^2(k,\alpha_i,-1,s) = \frac{(k-s)^2}{k-s+1} + \left(1+\frac{1}{s}\right)^{\alpha_i}s - \left(\frac{k}{k-1}\right)^{\alpha_i}(k-1)$$

である．ここで関数 H^2 を α_i で微分すると，

$$\begin{aligned}\frac{\partial}{\partial\alpha_i}H^2(k,\alpha_i,-1,s) &= \left(1+\tfrac{1}{s}\right)^{\alpha_i} s\ln\left(1+\tfrac{1}{s}\right) - \left(\tfrac{k}{k-1}\right)^{\alpha_i}(k-1)\ln\left(\tfrac{k}{k-1}\right)\\ &= \left(1+\tfrac{1}{s}\right)^{\alpha_i}\ln\left(1+\tfrac{1}{s}\right)^{s} - \left(1+\tfrac{1}{k-1}\right)^{\alpha_i}\ln\left(1+\tfrac{1}{k-1}\right)^{(k-1)}\end{aligned}$$

となる．正整数の対 s,k は $1\le s\le k-1$ を満たすので，α_i が非正であることから

$$\begin{array}{rcccl} 0 & \le & \left(1+\dfrac{1}{s}\right)^{\alpha_i} & \le & \left(1+\dfrac{1}{k-1}\right)^{\alpha_i} \qquad \text{および}\\ 0 & \le & \ln\left(1+\dfrac{1}{s}\right)^{s} & \le & \ln\left(1+\dfrac{1}{k-1}\right)^{k-1}\end{array}$$

が得られる．したがって関数 $H^2(k,\alpha_i,-1,s)$ は $\alpha_i\le 0$ に関して単調非減少である．ゆえに

$$\begin{aligned}H^2(k,\alpha_i,-1,s) &\ge H^2(k,0,-1,s) = \tfrac{(k-s)^2}{k-s+1} + \left(1+\tfrac{1}{s}\right)^0 s - \left(\tfrac{k}{k-1}\right)^0(k-1)\\ &= \tfrac{(k-s)^2}{k-s+1} + s - k + 1 = \tfrac{(k-s)^2+1^2-(k-s)^2}{k-s+1} = \tfrac{1}{k-s+1} \ge 0\end{aligned}$$

が成り立つ． □

さらに，離散化 Dirichlet 分布に対して単調マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ は条件 3.8 を満たす．

補題 5.11 Dirichlet パラメータ $(u_1,u_2,\ldots,u_n)$ の添え字の順が

$$u_1\ge u_2\ge\cdots\ge u_n$$

を満足する時，条件 3.8 が成り立つ．

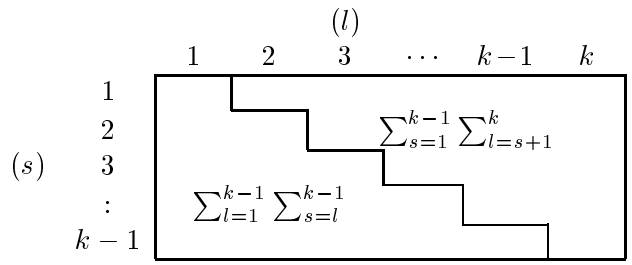
証明. 簡単のため

$$\begin{array}{rclcl} A_{k-1} &=& \sum_{s=0}^{k-1} f_i(s)f_j(k-s-1) &=& \sum_{s=1}^{k} s^{u_i-1}(k-s)^{u_j-1},\\ A_k &=& \sum_{s=0}^{k} f_i(s)f_j(k-s) &=& \sum_{s=1}^{k+1} s^{u_i-1}(k-s+1)^{u_j-1},\end{array}$$

とし，次の等式変形を行う．

$$\begin{aligned} G &\overset{\text{def.}}{=} \textstyle\sum_{l=1}^{k-1}[g_{ij}^k(l) - g_{ij}^{k+1}(l)] - \sum_{l=1}^{k-1}[g_{ij}^{k+1}(l) - g_{ij}^k(l-1)]\\ &= \textstyle\sum_{l=1}^{k-1} g_{ij}^k(l) - \sum_{l=1}^{k-1} g_{ij}^{k+1}(l)\\ &\quad \textstyle - \sum_{l=1}^{k-1} g_{ij}^{k+1}(l) + \sum_{l=1}^{k-1} g_{ij}^k(l-1)\\ &= \textstyle\sum_{l=1}^{k-1} g_{ij}^k(l) - \sum_{l=1}^{k-1} g_{ij}^{k+1}(l)\\ &\quad \textstyle - \sum_{l=1}^{k-1} g_{ij}^{k+1}(l) + \sum_{l=2}^{k-1} g_{ij}^k(l-1)\\ &= \textstyle\sum_{l=1}^{k-1}\frac{1}{A_{k-1}}\sum_{s=1}^{l} s^{u_i-1}(k-s)^{u_j-1} - \sum_{l=1}^{k-1}\frac{1}{A_k}\sum_{s=1}^{l} s^{u_i-1}(k-s+1)^{u_j-1}\\ &\quad \textstyle - \sum_{l=1}^{k-1}\frac{1}{A_k}\sum_{s=1}^{l} s^{u_i-1}(k-s+1)^{u_j-1} + \sum_{l=2}^{k-1}\frac{1}{A_{k-1}}\sum_{s=1}^{l-1} s^{u_i-1}(k-s)^{u_j-1}\end{aligned}$$

$$
\begin{aligned}
&= \tfrac{1}{A_{k-1}}\textstyle\sum_{s=1}^{k-1}(k-s)s^{u_i-1}(k-s)^{u_j-1} - \tfrac{1}{A_k}\sum_{s=1}^{k-1}(k-s)s^{u_i-1}(k-s+1)^{u_j-1}\\
&\quad -\tfrac{1}{A_k}\textstyle\sum_{s=1}^{k-1}(k-s)l^{u_i-1}(k-s+1)^{u_j-1} + \tfrac{1}{A_{k-1}}\sum_{s=1}^{k-1}(k-s-1)l^{u_i-1}(k-s)^{u_j-1}\\
&= \tfrac{1}{A_{k-1}}\textstyle\sum_{s=1}^{k-1}(2k-2s-1)s^{u_i-1}(k-s)^{u_j-1} - \tfrac{1}{A_k}\sum_{s=1}^{k-1}(2k-2s)s^{u_i-1}(k-s+1)^{u_j-1}\\
&= \tfrac{1}{A_{k-1}A_k}\Big(\ A_k\textstyle\sum_{s=1}^{k-1}(2k-2s-1)l^{u_i-1}(k-s)^{u_j-1}\\
&\qquad -A_{k-1}\textstyle\sum_{s=1}^{k}(2k-2s)l^{u_i-1}(k-s+1)^{u_j-1}\Big)\\
&= \tfrac{1}{A_{k-1}A_k}\Big(\ \textstyle\sum_{l=1}^{k}l^{u_i-1}(k-l+1)^{u_j-1}\sum_{s=1}^{k-1}(2k-2s-1)s^{u_i-1}(k-s)^{u_j-1}\\
&\qquad -\textstyle\sum_{l=1}^{k-1}l^{u_i-1}(k-l)^{u_j-1}\sum_{s=1}^{k}(2k-2s)s^{u_i-1}(k-s+1)^{u_j-1}\Big)\\
&= \tfrac{1}{A_{k-1}A_k}\Big(\ \textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2k-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\qquad -\textstyle\sum_{l=1}^{k-1}\sum_{s=1}^{k}(2k-2s)(ls)^{u_i-1}((k-l)(k-s+1))^{u_j-1}\Big)\\
&= \tfrac{1}{A_{k-1}A_k}\Big(\ \textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2k-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\qquad -\textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2k-2l)(ls)^{u_i-1}((k-s)(k-l+1))^{u_j-1}\Big)\\
&= \tfrac{1}{A_{k-1}A_k}\textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}\big((2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\big)\\
&= \tfrac{1}{A_{k-1}A_k}\cdot\tfrac{1}{2}\Big(\ \textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\qquad +\textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\Big)\\
&= \tfrac{1}{2A_{k-1}A_k}\Big(\textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\quad +\textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2(k-l+1)-2(k-s)-1)\\
&\qquad ((k-l+1)(k-s))^{u_i-1}((k-(k-l+1)+1)(k-(k-s)))^{u_j-1}\Big)\\
&= \tfrac{1}{2A_{k-1}A_k}\Big(\ \textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\qquad -\textstyle\sum_{l=1}^{k}\sum_{s=1}^{k-1}(2l-2s-1)((k-l+1)(k-s))^{u_i-1}(ls)^{u_j-1}\Big)
\end{aligned}
$$



$$
\begin{aligned}
&= \tfrac{1}{2A_{k-1}A_k}\Big(\ \textstyle\sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\qquad +\textstyle\sum_{l=1}^{k-1}\sum_{s=l}^{k-1}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1}\\
&\qquad -\textstyle\sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)((k-l+1)(k-s))^{u_i-1}(ls)^{u_j-1}\\
&\qquad -\textstyle\sum_{l=1}^{k-1}\sum_{s=l}^{k-1}(2l-2s-1)((k-l+1)(k-s))^{u_i-1}(ls)^{u_j-1}\Big)
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{2A_{k-1}A_k} \Big(\sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1} \\
&\quad - \sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)(s(l-1))^{u_i-1}((k-s+1)(k-l+1))^{u_j-1} \\
&\quad - \sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)((k-l+1)(k-s))^{u_i-1}(ls)^{u_j-1} \\
&\quad + \sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)((k-s+1)(k-l+1))^{u_i-1}(s(l-1))^{u_j-1}\Big) \\
&= \frac{1}{2A_{k-1}A_k}\sum_{s=1}^{k-1}\sum_{l=s+1}^{k}(2l-2s-1)\ \Big((ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1} \\
&\quad -(s(l-1))^{u_i-1}((k-s+1)(k-l+1))^{u_j-1} \\
&\quad -((k-l+1)(k-s))^{u_i-1}(ls)^{u_j-1} \\
&\quad +((k-s+1)(k-l+1))^{u_i-1}(s(l-1))^{u_j-1}\Big).
\end{aligned}
$$

いま関数 $G_0(l,s,u_i,u_j)$ を

$$
G_0(l,s,u_i,u_j) \stackrel{\text{def.}}{=} \left(\begin{array}{l}
(ls)^{u_i-1}((k-l+1)(k-s))^{u_j-1} \\
-(s(l-1))^{u_i-1}((k-s+1)(k-l+1))^{u_j-1} \\
-((k-l+1)(k-s))^{u_i-1}(ls)^{u_j-1} \\
+((k-s+1)(k-l+1))^{u_i-1}(s(l-1))^{u_j-1}
\end{array} \right)
$$

と定義する．この時 $1 \le s < s+1 \le l \le k$ より，$(2l-2s-1) > 0$ は明らかに成り立つ．したがって $\forall s \in \{1,2,\ldots,k-1\}$, $\forall l \in \{2,3,\ldots,k\}$, $\forall u_i \ge \forall u_j$, $G_0(s,l,u_i,u_j) \ge 0$ を示せば十分である．明らかに

$$
\begin{aligned}
&G_0(l,s,u_i,u_j) \\
&= \left(\begin{array}{l}
(s(l-1))^{u_i-1}(k-l+1)^{u_j-1}\left((1+\frac{1}{l-1})^{u_i-1}(k-s)^{u_j-1} - (k-s+1)^{u_j-1}\right) \\
+((k-l+1)(k-s))^{u_i-1}s^{u_j-1}\left(-l^{u_j-1} + (1+\frac{1}{k-s})^{u_i-1}(l-1)^{u_j-1}\right)
\end{array} \right)
\end{aligned}
$$

なので，$G_0(l,s,u_i,u_j)$ は u_i に関して非減少である．したがって $G_0(l,s,u_i,u_j) \ge G_0(l,s,u_j,u_j)$ を得る．ここで u_i に u_j を代入すると $G_0(l,s,u_i,u_j)$ の定義から $G_0(l,s,u_j,u_j) = 0$ となり，題意は示される． □

5.3.4 計算機実験

表 2 は離散化の影響についてまとめたものである．いくつかの離散化 Dirichlet 分布に対して離散化幅を $1/K$ とした時の期待値 $\mathrm{E}_K[P_i]$, 分散 $\mathrm{Var}_K[P_i]$, および共分散 $\mathrm{Cov}_K[P_i,P_j]$ を素朴な手法で計算した．

本小節では計算実験の結果を記す．全ての計算機実験において，擬似乱数としてメルセンヌツイスター [80] を用いた．実験は以下のスペックの Linux マシンを用いた．

マシン: Dell Precision 450
CPU: Intel Xeon 2.8GHz (FSB 533MHz) × 2
OS: RedHat Linux 8.0 (Kernel 2.4.18-14smp)
メモリ: Dual channel PC2100 DDR SDRAM 2GByte
コンパイラ: Intel C++ Compiler 7.0

表 2　離散化の影響

(u_1, u_2, u_3, u_4)	統計量の最大差	K		
		10	50	100
$(1, 1, 1, 1)$	$\lvert\mathcal{E}_K[P_i] - \mathcal{E}[P_i]\rvert$	0	0	0
	$\lvert\mathrm{Var}_K[P_i] - \mathrm{Var}[P_i]\rvert$	0.015	0.003	0.0015
	$\lvert\mathrm{Cov}_K[P_i, P_j] - \mathrm{Cov}[P_i, P_j]\rvert$	0.005	0.001	0.0005
$(4, 3, 2, 1)$	$\max(\lvert\mathcal{E}_K[P_i] - \mathcal{E}[P_i]\rvert)$	0.051	0.0092	0.0046
	$\max(\lvert\mathrm{Var}_K[P_i] - \mathrm{Var}[P_i]\rvert)$	0.0036	0.00049	0.00023
	$\max(\lvert\mathrm{Cov}_K[P_i, P_j] - \mathrm{Cov}[P_i, P_j]\rvert)$	0.0080	0.0074	0.0073
$(0.1, 0.1, 0.1, 0.1)$	$\lvert\mathcal{E}_K[P_i] - \mathcal{E}[P_i]\rvert$	0	0	0
	$\lvert\mathrm{Var}_K[P_i] - \mathrm{Var}[P_i]\rvert$	0.11	0.071	0.061
	$\lvert\mathrm{Cov}_K[P_i, P_j] - \mathrm{Cov}[P_i, P_j]\rvert$	0.035	0.024	0.020
$(0.4, 0.3, 0.2, 0.1)$	$\max(\lvert\mathcal{E}_K[P_i] - \mathcal{E}[P_i]\rvert)$	0.13	0.10	0.092
	$\max(\lvert\mathrm{Var}_K[P_i] - \mathrm{Var}[P_i]\rvert)$	0.090	0.055	0.045
	$\max(\lvert\mathrm{Cov}_K[P_i, P_j] - \mathrm{Cov}[P_i, P_j]\rvert)$	0.051	0.042	0.040
$(2, 1.5, 1, 0.5)$	$\max(\lvert\mathcal{E}_K[P_i] - \mathcal{E}[P_i]\rvert)$	0.079	0.029	0.019
	$\max(\lvert\mathrm{Var}_K[P_i] - \mathrm{Var}[P_i]\rvert)$	0.014	0.0032	0.0019
	$\max(\lvert\mathrm{Cov}_K[P_i, P_j] - \mathrm{Cov}[P_i, P_j]\rvert)$	0.015	0.013	0.013

近似サンプリング法と完璧サンプリング法のマルコフ連鎖の違いは添え字の選び方だけであり，各推移にかかる計算時間の影響はほぼ皆無と言える．推移を 10^{10} 回行うための実際の計算時間は上記の計算環境で 2∼6 時間であった．

近似サンプリング法: 各パラメータの設定について，10^9 回の試行をを行った．実験では乱数の種を決定的に与えた上で各試行についてマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ を 50 回の推移を行った．初期状態は Δ 中の整数ベクトルで，$(K/n, \ldots, K/n)$ の整数丸めで得られるものとする．(第 5.3.4 章では上記の設定を少し変更する．)

完璧サンプリング法: 各 Dirichlet パラメータに対して，アルゴリズム 1 (完璧サンプリングアルゴリズム) の試行を 10^4 回行った．各試行では擬似乱数の種を与え，合流時間 T_* を厳密に計測した．

Dirichlet パラメータの影響

まず，Dirichlet パラメータと近似サンプリング法の収束時間の関係について考察する．この実験では次元 n を 4 に固定し，離散化幅 $1/K$ を $1/100$ に固定する．Dirichlet パラメータとして $(1, 1, 1, 1)$, $(4, 3, 2, 1)$, $(2, 1.5, 1, 0.5)$, $(0.1, 0.1, 0.1, 0.1)$, $(0.4, 0.3, 0.2, 0.1)$, $(10^{-5}, 10^{-5}, 10^{-5}, 10^{-5})$ について実験を行った．なお，$(1, 1, 1, 1)$ は Δ 上の一様分布である．

図 9 はマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ を用いた近似サンプリング法の結果である．縦軸は 10^9 回の試行によ

り得られた近似分布と真の分布との全変動距離 ε を表し，横軸はマルコフ連鎖の推移回数を表す．図 9 において，全変動距離が値 0.005 の周辺で横ばい状態になっているが，これは試行回数が 10^9 であることによる有効桁数の限界によるものである．実験の結果から Dirichlet パラメータに関して，パラメータの値が 1 以上の場合の方が，パラメータの値が 1 未満の場合に比べて収束が速いことが窺える．この考察は次元 4 の場合について行ったものであることに注意が必要である．

表 3 は CFTP アルゴリズムの合流時間に関する実験結果をまとめたものである．近似サンプリング法に比べて，Dirichlet パラメータが 1 より小さい場合の方が合流が速いことが窺える．

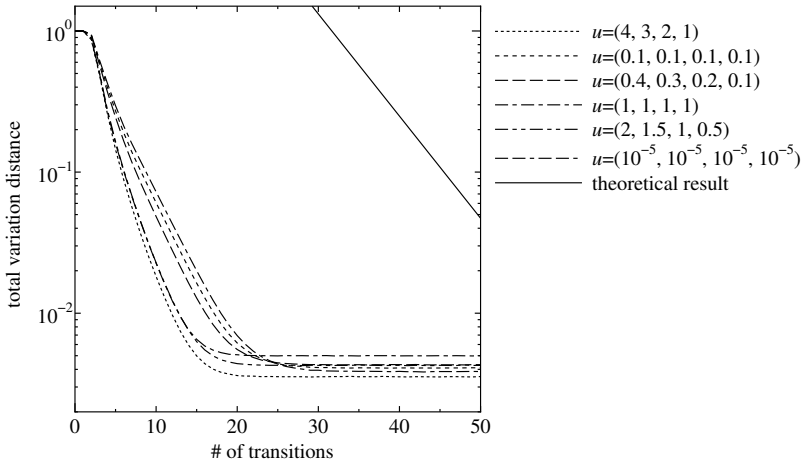


図 9　近似サンプリング法: Dirichlet パラメータ (u_1, u_2, u_3, u_4) の影響．

表 3　完璧サンプリング法: Dirichlet パラメータ (u_1, u_2, u_3, u_4) の影響．

			合流時間			
n	$1/K$	(u_1, u_2, u_3, u_4)	ave.	(s. d.)	max.	min.
4	1/100	(1,1,1,1)	45.5	(16.4)	129	4
4	1/100	(4,3,2,1)	41.3	(13.1)	114	9
4	1/100	(2,1.5,1,0.5)	39.6	(14.5)	127	3
4	1/100	(0.1,0.1,0.1,0.1)	34.9	(17.9)	141	3
4	1/100	(0.4, 0.3, 0.2, 0.1)	36.4	(17.5)	140	3
4	1/100	$(10^{-5}, 10^{-5}, 10^{-5}, 10^{-5})$	33.2	(17.7)	125	3

ave.: average,　s. d.: standard deviation

離散化幅の影響 次に離散化幅 $1/K$ が混交時間に与える影響について実験する．再び次元 n を 4 に，パラメータを $(1, 1, 1, 1)$ に固定する．離散化の大きさ K を 10，20，50，100 および 200 とする．

図 10 は近似サンプリング法について実験し，各 K に対して，全変動距離 ε をプロットしたものである．図 10 から K が混交時間に与える影響はほとんど無いことが伺える．詳細には，K に依

らず推移回数に比して全変動距離は有効桁数の影響で横ばいになるまで減少している．定理 3.2 の証明中で $(K-n)$ の項はグラフ $\mathcal{G}=(\Delta,\mathcal{E})$ の直径として人工的に導入されたものであるが，実験結果から混交時間は K に依存しないことが示唆される．すなわち，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の真の混交時間は強多項式時間であると予想される．このことは，マルコフ連鎖の直径が K には依存せず，n 以下であるという事実からも支持されるものである．

完璧サンプリング法に対しては $K=400$ の場合と，16 次元の場合を追加して実験を行った．表 4 は合流時間の分布についてまとめたものである．合流時間の平均は K に比して増加している．したがって，完璧サンプリング法の合流時間に対する本研究での算定は $\ln K$ の線形項を含んでいるが，計算実験の結果はこの算定が K に関してタイトであることを支持している．

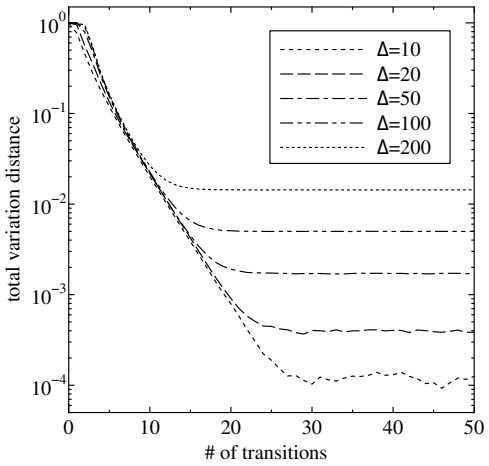


図 10　近似サンプリング法: 離散化幅 $1/K$ の影響．

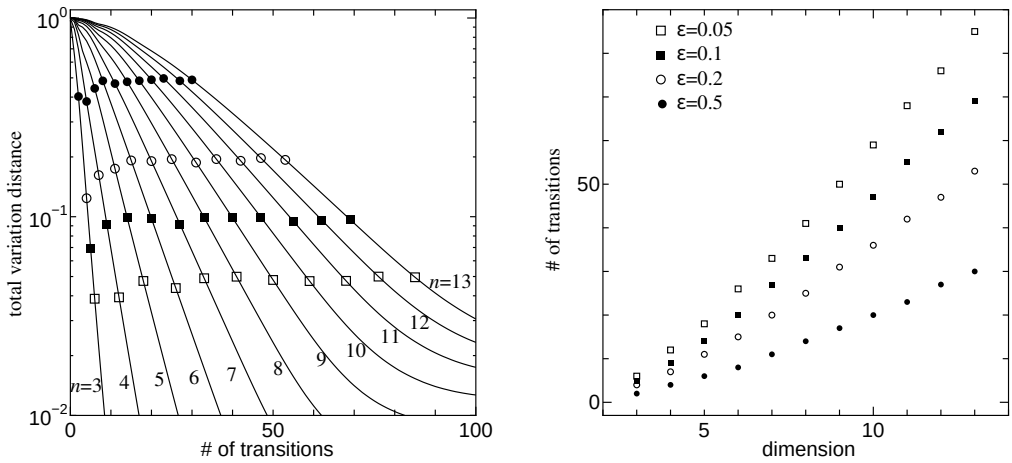
次元の影響　最後に次元 n の影響について実験する．近似サンプリング法に対しては，メモリの制限から離散化幅 $1/K$ を $1/26$ に固定し，次元 n を 3 から 13 の間で変化させた．各パラメータはすべて 1 とした．各次元について，初期状態を $(K-n+1,1,\ldots,1)$ として，近似サンプリング法を用いて 2×10^9 回の試行を行った．実験結果が図 11 である．次元と混交時間の関係について調べることを目的とし，全変動距離がはじめて 0.5，0.2，0.1 および 0.05 を下回った推移回数に着目する．図 11 において，これらの時間はそれぞれ •，◦，■ および □ で表されている．図 11 (b) は全変動距離 ε に関する結果をまとめたものである．十分な大きさの次元について実験が行われたとは言えないが，実験結果は混交時間が $\mathrm{o}(n^2)$ であることを示唆している．

完璧サンプリング法に対しては $K=400$ の場合とパラメータ $(u_1,\ldots,u_n)=(0.5,\ldots,0.5),(2,\ldots,2)$ の場合を追加して実験を行った．表 5 は合流時間の分布をまとめたものである．実験の結果は，完璧サンプリング法の合流時間の算定がタイトであることを示唆している．すなわち固定された K に対して，完璧サンプリングの計算時間は $\Theta(n^3)$ である．

表 4　完璧サンプリング法: 離散化幅 $1/K$ の影響．

			合流時間				
n	$1/K$	$(u_1, u_2, \ldots, u_n)$	ave.	(s. d.)	max.	min.	$\frac{\text{ave.}}{\ln K}$
4	1/10	(1,1,1,1)	24.9	(13.1)	107	3	10.8
4	1/20	(1,1,1,1)	32.2	(14.4)	121	3	10.7
4	1/50	(1,1,1,1)	40.1	(15.8)	150	3	10.3
4	1/100	(1,1,1,1)	45.5	(16.4)	129	4	9.9
4	1/200	(1,1,1,1)	50.5	(17.3)	153	6	9.5
4	1/400	(1,1,1,1)	55.6	(18.0)	149	8	9.0
16	1/20	$(1, 1, \ldots, 1)$	2,030.7	(940.1)	7,617	308	469.9
16	1/50	$(1, 1, \ldots, 1)$	3,612.4	(978.0)	10,967	1,236	640.1
16	1/100	$(1, 1, \ldots, 1)$	4,303.7	(1,000.2)	11,612	1,942	647.8
16	1/200	$(1, 1, \ldots, 1)$	4,888.2	(1,017.4)	10,530	2,235	639.5
16	1/500	$(1, 1, \ldots, 1)$	5,624.8	(1,028.9)	12,334	3,001	627.4
16	$1/10^3$	$(1, 1, \ldots, 1)$	6,152.6	(1,048.9)	15,050	3,376	617.4
16	$1/10^4$	$(1, 1, \ldots, 1)$	7,899.4	(1,113.5)	15,388	4,862	594.5
16	$1/10^5$	$(1, 1, \ldots, 1)$	9,626.1	(1,133.3)	15,961	6,341	579.5
16	$1/10^6$	$(1, 1, \ldots, 1)$	11,368.5	(1,180.5)	17,790	7,768	570.4
16	$1/10^7$	$(1, 1, \ldots, 1)$	13,086.4	(1,216.0)	19,576	9,346	562.8
16	$1/10^8$	$(1, 1, \ldots, 1)$	14,826.5	(1,271.0)	22,178	10,984	557.9
16	$1/10^9$	$(1, 1, \ldots, 1)$	16,559.6	(1,308.0)	23,744	12,631	553.9

ave.: average,　s. d.: standard deviation



(a): 次元と全変動距離．　　(b): 次元と混交時間．

図 11　近似サンプリング法: 次元 n の影響．

表 5　完璧サンプリング法: 次元 n の影響．

			合流時間				
n	$1/K$	$(u_1, u_2, \ldots, u_n)$	ave.	(s. d.)	max.	min.	$\frac{\ln(\text{ave.})}{\ln n}$
3	1/20	$(1, 1, 1)$	10.2	(5.4)	42	2	1.54
4	1/20	$(1, 1, 1, 1)$	32.2	(14.4)	121	3	1.92
5	1/20	$(1, 1, \ldots, 1)$	71.3	(29.1)	238	6	2.09
6	1/20	$(1, 1, \ldots, 1)$	129.4	(48.8)	454	14	2.16
7	1/20	$(1, 1, \ldots, 1)$	213.3	(79.6)	754	36	2.24
4	1/400	$(1, 1, 1, 1)$	55.6	(18.0)	149	8	2.90
8	1/400	$(1, 1, \ldots, 1)$	611.1	(134.4)	1,502	218	3.09
16	1/400	$(1, 1, \ldots, 1)$	5,431.5	(1,034.4)	12,896	2,550	3.10
32	1/400	$(1, 1, \ldots, 1)$	45,324.8	(8,146.6)	103,033	27,448	3.09
64	1/400	$(1, 1, \ldots, 1)$	364,470.2	(64,404.3)	860,360	226,763	3.08
128	1/400	$(1, 1, \ldots, 1)$	2,865,607.6	(525,024.8)	6,589,297	1,778,885	3.06
4	1/400	$(0.5, 0.5, \ldots, 0.5)$	48.6	(19.6)	162	3	2.80
8	1/400	$(0.5, 0.5, \ldots, 0.5)$	577.2	(156.4)	1,466	152	3.06
16	1/400	$(0.5, 0.5, \ldots, 0.5)$	5,313.5	(1,119.6)	15,007	2,438	3.09
32	1/400	$(0.5, 0.5, \ldots, 0.5)$	44,859.7	(8,458.7)	117,760	24,232	3.09
64	1/400	$(0.5, 0.5, \ldots, 0.5)$	362,574.2	(65,575.4)	843,410	207,957	3.08
4	1/400	$(2, 2, \ldots, 2)$	60.9	(16.2)	172	20	2.96
8	1/400	$(2, 2, \ldots, 2)$	635.0	(123.3)	1,741	316	3.10
16	1/400	$(2, 2, \ldots, 2)$	5,543.1	(972.4)	13,273	3,218	3.11
32	1/400	$(2, 2, \ldots, 2)$	45,525.0	(7,876.1)	98,410	28,206	3.09
64	1/400	$(2, 2, \ldots, 2)$	366,243.8	(63,508.6)	830,563	229,377	3.08

ave.: average,　s. d.: standard deviation

6 まとめ

本論文では $n-1$ 次元単体中の整数格子点上に定義された対数分離凹分布に対してマルコフ連鎖モンテカルロ (MCMC) 法に基づく効率的な近似サンプリング法，完璧サンプリング法，および正規化定数の計算法を提案し，その性能解析を行った．さらに提案手法を閉ジャクソンネットワークの積形式解，2 行分割表の一様分布および多項超幾何分布，離散化 Dirichlet 分布に適用し，新たに多項式時間の乱択計算法を提案した．

本論文の特長的な成果の一つとして，既存の研究では成功例の少ない単調マルコフ連鎖の設計に成功した点が挙げられる．単調 CFTP アルゴリズムに基づくサンプリング法の利点として以下の 2 点が挙げられる．ひとつは，通常の MCMC 法の計算時間は誤差パラメータに大きく依存するのに対し，単調 CFTP アルゴリズムの計算量は誤差パラメータに依存しないので，高精度のサンプリング法が必要な場面においては効率的となり得る点である．もうひとつは，通常の MCMC 法では，理論的に精度の保証された近似サンプリングを得るためには上界値の回数の推移が必ず必要であるのに対し，単調 CFTP アルゴリズムは自動終了アルゴリズムであるため，インスタンスによっては見積もった計算時間より短い時間で終了する可能性がある点である．

既存の MCMC 法の理論的研究の多くが，物理現象のシミュレーションや数学的な興味に基づいた特定の対象に対するものであったのに対し，本論文では生物情報，経営，OR といった応用分野に焦点をあてている点も大きな特色である．ジャクソンネットワーク，Dirichlet 分布といった応用に対し，本論文で得られた結果は単調マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の合流時間の期待値が $2n^3 \ln(nK)$ 以下，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の混交時間が $n^2 \ln(K\varepsilon^{-1})$ 以下と，定数部分も含めて現実的な計算量と言える．マルコフ連鎖の収束の速さの算定に関する他の理論的な研究，凸体の一様サンプリングの $\mathrm{O}(n^4)$ [74]，2 部グラフの完全マッチングの $\mathrm{O}(n^7)$ [10]，あるいは Cryan et al. の $3 \times n$ 分割表の一様サンプリングの $\mathrm{O}(n^{\text{数百}})$ [21] といった「多項式時間」に比較すると，本研究の算定は十分実用的と言える．実用性のひとつの裏付けとして，本論文では提案したアルゴリズムを実装し，計算機実験を行っている．本論文は理論的な算定と計算機実験を併せて行った数少ない研究と言える．また，本論文で得られた計算量は Dirichlet 分布の Dirichlet パラメータや待ち行列のサービス率などの分布のパラメータに依存しないので，様々な入力に対してロバストである．本論文ではまた，正規化定数の乱択近似計算法における近似解の期待値の，真の値からの偏りに対する理論的な算定も行った．実用の場面において，時に理論的な精度保証の得られる計算時間を確保できない場合にも本研究の手法は質の良い近似解を与える．

本論文の成果と既存の研究を表 6 にまとめる．表 6 の 2 行分割表の一様分布に対する近似サンプリング法，および FPRAS の結果は Dyer と Greenhill [27] によるもの，2 行分割表の多項超幾何分布に対する近似サンプリング法の結果は Matsui, Matsui, Ono [76] によるもの，離散化 Dirichlet 分布に対する近似サンプリング法の結果は Matsui, Motoki, Kamatani [77] によるものである．これら既存の結果も，本論文の対数分離凹関数の性質を用いて統一的に示すことが可能となった．

表 6　本論文の成果一覧．

	近似サンプリング法 混交時間 $\tau(\varepsilon)$	単調 CFTP 期待合流時間 $\mathrm{E}[T_*]$	正規化定数の計算
ジャクソンネットワーク （複数サーバ）	$\frac{n(n-1)}{2}\ln(K\varepsilon^{-1})$	多項式性未解決	擬多項式時間 (Gordon, Newell [38])
ジャクソンネットワーク （単一サーバ）	$\frac{n(n-1)}{2}\ln(K\varepsilon^{-1})$	$\mathrm{O}(n^3\ln(nK))$	FPRAS
2 行分割表 （一様分布）	$\frac{n(n-1)}{2}\ln(K\varepsilon^{-1})$ (Dyer, Greenhill [27])	$\mathrm{O}(n^3\ln(nK))$	FPRAS (Dyer, Greenhill [27])
2 行分割表 （多項超幾何分布）	$\frac{n(n-1)}{2}\ln(K\varepsilon^{-1})$ (Matsui, et al. [76])	多項式性未解決	解析解（式 (5.2.3)） ([43] 参照)
離散化 Dirichlet 分布	$\frac{n(n-1)}{2}\ln(K\varepsilon^{-1})$ (Matsui, et al. [77])	$\mathrm{O}(n^3\ln(nK))$	扱っていない

表 6 において近似サンプリング法の項にはマルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の混交時間の上界を記載している．これらの上界は，本論文の手法にて示すことが可能である．単調 CFTP の項にはマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の期待合流時間の上界を記載している．単調 CFTP の項については，すべての事例に対する単調 CFTP アルゴリズムに基づく完璧サンプリング法を本論文にて与えた．特に $\mathrm{O}(n^3\ln(nK))$ とあるものは，マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の合流時間の期待値の多項式性を示したものである．多項式性未解決とあるものは，単調マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の多項式性が未解決であることを表す．正規化定数の計算の項について，ジャクソンネットワーク単一サーバの FPRAS を本論文にて与えた．2 行分割表の一様分布に対する FPRAS は，既存の手法と同じ計算量の異なる計算方法を与えた．その他の事例に対しても本論文の手法にて乱択近似計算が行えるが，提案手法の多項式性は未解決である．ただし，2 元分割表の多項超幾何分布の正規化定数は式 (5.2.3) により陽な関数で表現される．以下，表 6 の未解決問題をまとめる．

ジャクソンネットワーク（複数サーバ）の完璧サンプリング法:

マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ に基づく単調 CFTP が設計可能だが，合流時間の期待値の多項式性が不明．

ジャクソンネットワーク（複数サーバ）の正規化定数の計算:

マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の 1 回の推移が多項式時間で実行できるか否かが不明．実行できれば，マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ とアルゴリズム 4.2 に基づく FPRAS が設計可能．既存の計算法として，1967 年の Gordon と Newell による擬多項式時間の計算方法がある [38]．

2 行分割表の多項超幾何分布の完璧サンプリング法:

マルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ に基づく単調 CFTP が設計可能だが，合流時間の多項式性が不明．マルコフ連鎖を利用せずに，擬多項式時間の完璧サンプリングが可能．

離散化 Dirichlet 分布の FPRAS:

マルコフ連鎖 $\mathcal{M}_{\mathrm{A}}$ の 1 回の推移が多項式時間で実行できるか否かが不明．離散化 Dirichlet 分布の正規化定数を求める問題については，現実的な応用が（現在のところ）見当たらないため，本研究では扱っていない．連続 Dirichlet 分布の正規化定数は明示的な関数で表現可能．

最後に，今後の課題と未解決問題について述べる．第一に本論文の理論的な内容に関する未解決問題として，提案した完璧サンプリング法が条件 3.8 を満たさない場合についてのマルコフ連鎖 $\mathcal{M}_{\mathrm{P}}$ の合流時間の期待値の多項式性が未解決である．この問題が肯定的に解決されると，ジャクソンネットワーク，2 行分割表の多項超幾何分布に対して，多項式時間の期待合流時間をもつ単調 CFTP に基づく完璧サンプリング法が得られる．本論文の拡張として 2 つの方向性が考えられる．ひとつの方針としては分布に関する拡張である．単体中の整数格子点上に定義される（変数分離形ではない）一般の対数凹関数に対する混交時間が多項式時間のマルコフ連鎖の設計は未解決である．このようなマルコフ連鎖が設計されるとユニバーサルポートフォリオに対する効率的な乱択近似計算法が設計できる [52]．もうひとつの方針は状態空間の拡張である．(単体とは限らない）一般の多面体に対する多項式時間収束のマルコフ連鎖の存在は未解決であり，特にポリマトロイドの基多面体に限っても未解決である．これが実現すれば一般の 2 元分割表の多項式時間サンプリング，マトロイドの基のサンプリング，呼損ネットワークなどの応用がある．

以下，関連する未解決問題について述べる．一般の $m \times n$ サイズの 2 元分割表に対する一様サンプリング法の多項式性は未解決である [21]．特に，熱浴法によるマルコフ連鎖の混交時間は多項式時間と予想されている．一般のマトロイドの基の一様サンプリング法の多項式性は未解決である [29]．この問題に対しては交換則に基づくマルコフ連鎖の混交時間は多項式時間と予想されている．BCMP ネットワークはジャクソンネットワークの拡張に当たる待ち行列ネットワークモデルで，積形式解をもつことが知られている [9]．この積形式解に対するサンプリング法の多項式性は未解決である．呼損ネットワークは，現実に即した待ち行列ネットワークモデルのひとつで，積形式解を持つことが知られている [57]．しかし，モデルが複雑でネットワークに対する性能指標の計算が困難なため，効率的な近似計算法の開発が望まれている．MCMC 法の設計は可能だが，精度保証に関する成果は得られていない．

謝辞

本論文の完成には，多くの方々のお世話になりました．一部の主だった方の名をここに挙げさせていただくと共に謝意を表します．

直接の指導に当っていただいた中央大学理工学部の松井知己教授には，卒業研究，修士課程，博士課程に渡って，研究指導，論文添削，発表指導，研究の心構えと進め方，その他研究活動全般について暖かく，辛抱強く，手厚いご指導を頂きました．心より感謝致します．

指導教員である杉原厚吉教授には，研究活動の全般にわたる丁寧なご指導，ご支援賜りました．数理情報学専攻の室田一雄教授，竹村彰通教授，牧野和久助教授には研究に関する様々なご指導，ご助言をいただきました．数理 5 研の皆様，数理 2 研の皆様，数理情報学専攻の諸先生方はじめ皆様には大変お世話になりました．

共同研究者である北陸先端科学技術大学院大学の元木光雄助手，東京女子医科大学附属膠原病リウマチ痛風センターの鎌谷直之所長には，論文執筆を通じて様々なご助力をいただきました．

文教大学の根本俊男助教授，京都大学数理解析研究所の岩田覚助教授，国立情報学研究所の宇野毅明助教授，名古屋大学の柳浦睦憲助教授，東北大学の塩浦昭義助教授，兵庫県立大学の藤江哲也助教授，北陸先端科学技術大学院大学の上原隆平助教授，東海大学の松井泰子助教授，群馬大学の中野眞一教授，慶應義塾大学の田村明久教授には研究会，セミナー等を通じて様々な勉強をさせていただきました．

東京農工大学の宮代隆平助手 ,数理 5 研の松浦史郎研究員 ,上智大学の宮本裕一郎助手 ,ERATO 合原複雑数理モデルプロジェクトの齊藤廣大研究員，北陸先端科学技術大学院大学の清見礼助手には研究生活における貴重な情報を教えていただきました．

鹿児島大学の青木敏助教授，上智大学の森口聡子助手，豊橋技術科学大学の岡本吉央助手，電気通信大学の梅谷俊治助手，数理 2 研の土村展之助手，数理 3 研の谷口隆晴助手，数理 5 研の今堀慎治助手には様々なご助言，ご助力をいただきました．

Massachusetts 大学 Lowell 校の Ravi Montenegro 講師，New South Wales 大学の Catherine Greenhill 講師，Microsoft Research の David Wilson 研究員，McGill 大学 Luc Devroye 教授，New York 大学の Alan Sokal 教授，Australian 国立大学の Brendan McKay 教授，そして ETH Zürich の福田公明教授には研究への真摯で有益なコメントをいただきました．

その他，ここには名前を書ききれなかった多くの方々から温かい応援をいただきました．皆様，本当にありがとうございました．心よりお礼申し上げます．

参考文献

[1] A. Agresti, A survey of exact inference for contingency tables, Statistical Science, **7** (1992), 131–153.

[2] A. Agresti, Categorical Data Analysis, John Wiley & Sons, 2002.

[3] D. Aldous, Random walks on finite groups and rapidly mixing Markov chains, in Séminarie de Probabilitiés XVII 1981/1982, vol. 986 of Springer-Verlag Lecture Notes in Mathematics, D. Dold and B. Eckmann, (ed.), Springer, 1983, 243–297.

[4] M. Y. An, Log-concave Probability Distributions: Theory and Statistical Testing, Papers 96-01, Centre for Labour Market and Social Research, Danmark, 1996. (available from http://ideas.repec.org/s/fth/clmsre.html)

[5] S. Aoki and A. Takemura, Minimal basis for connected Markov chain over $3 \times 3 \times K$ contingency tables with fixed two-dimensional marginals, Australian and New Zealand Journal of Statistics, **45** (2003), 229–249.

[6] 青本和彦，上野健爾，加藤和也，神保道夫，砂田利一，高橋陽一郎，深谷賢治，俣野博，室田一雄（編），岩波数学入門辞典，岩波書店，2005．

[7] D. Avis and K. Fukuda, Reverse search for enumeration, Discrete Applied Mathematics, **65** (1996), 21–46.

[8] M. Bagnoli and T. Bergstrom, Log-concave probability and its applications, Economic Theory, **26** (2005), 445–469.

[9] F. Baskett, K. M. Chandy, R. R. Muntz, and F. G. Palacious, Open, closed and mixed networks of queues with different classes of customers, Journal of ACM, **22** (1975), 248–260.

[10] I. Bezakova, D. Stefankovic, E. Vigoda, and V. V. Vazirani, Accelerating simulated annealing algorithm for the permanent and combinatorial counting problems, Proceedings of the 17th ACM-SIAM Symposium on Discrete Algorithms (SODA 2006), 900–907.

[11] S. Boyd, P. Diaconis and L. Xiano, Fastest mixing Markov chain on a graph, SIAM Review, **46** (2004), 667–689.

[12] R. Bubley, Randomized Algorithms: Approximation, Generation, and Counting, Springer, 2001.

[13] R. Bubley and M. Dyer, Path coupling: a technique for proving rapid mixing in Markov chains, Proceedings of the 38th Annual Symposium on Foundations of Computer Science (FOCS 1997), 223–231.

[14] T. L. Burr, Quasi-equilibrium theory for the distribution of rare alleles in a subdivided population: justification and implications, Theoretical Population Biology, **57** (2000), 297–306.

[15] D. Burr, H. Doss, G. E. Cooke, and P. J. Goldschmidt-Clermont, A meta-analysis of studies on the association of the platelet PlA polymorphism of glycoprotein IIIa and risk of coronary heart disease, Statistics in Medicine, **22** (2003), 1741–1760.

[16] J. P. Buzen, Computational algorithms for closed queueing networks with exponential servers, Communications of the ACM, **16** (1973), 527–531.

[17] X. Chao, M. Miyazawa, and M. Pinedo, Queueing Networks, Customers, Signals and Product Form Solutions, John Wiley & Sons, Inc, 1999.

[18] W. Chen and C. A. O'Cinneide, Towards a polynomial-time randomized algorithm for closed product-form networks, ACM Transactions on Modeling and Computer Simulation, **8** (1998), 227–253.

[19] C. Cooper and A. Frieze, Mixing properties of the Swendsen-Wang process in classes of graphs, Random Structures and Algorithms **15**, (1999), 242-261.

[20] M. Cryan and M. Dyer, A polynomial-time algorithm to approximately count contingency tables when the number of rows is constant, Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC 2002), 240–249.

[21] M. Cryan, M. Dyer, L. A. Goldberg, M. Jerrum, and R. Martin, Rapidly mixing Markov chains for sampling contingency tables with constant number of rows, Proceedings of the 43rd Annual Symposium on Foundations of Computer Science (FOCS 2002), 711–720.

[22] P. Diaconis and B. Effron, Testing for independence in a two-way table: new interpretations of the chi-square statistic (with discussion), The Annals of Statistics, **13** (1985), 845–913.

[23] P. Diaconis and L. Saloff-Coste, Random walk on contingency tables with fixed row and column sums, Technical report, Department of Mathematics, Harvard University, 1995.

[24] X. K. Dimakos, A guide to exact simulation, International Statistical Review, **69** (2001), 27–48.

[25] R. Durbin, R. Eddy, A. Krogh, and G. Mitchison, Biological sequence analysis: probabilistic models of proteins and nucleic acids, Cambridge Univ. Press, 1998.

[26] M. Dyer, A. Frieze, and R. Kannan, A random polynomial time algorithm for approximating the volume of convex bodies, Journal of ACM, **38** (1991), 1–17.

[27] M. Dyer and C. Greenhill, Polynomial-time counting and sampling of two-rowed contingency tables, Theoretical Computer Sciences, **246** (2000), 265–278.

[28] M. Dyer, R. Kannan, and J. Mount, Sampling contingency tables, Random Structures and Algorithms, **10** (1997), 487–506.

[29] T. Feder and M. Mihail, Balanced matroid, Proceedings of the 24th ACM Symposium on Theory of Computing (STOC 1992), 26-38.

[30] J. Fill, An interruptible algorithm for perfect sampling via Markov chains, The Annals of Applied Probability, **8** (1998), 131–162.

[31] J. Fill, M. Machida, D. Murdoch, and J. Rosenthal, Extension of Fill's perfect rejection sampling algorithm to general chains, Random Structures and Algorithms, **17** (2000), 290–316.

[32] R. A. Fisher, The logic of inductive inference (with discussion), Journal of Royal Statistical Society, **98** (1935), 39–54.

[33] A. Frieze, R. Kannan, and N. Polson, Sampling from log-concave distribution, The Annals of Applied Probability, **4** (1994), 812–837.

[34] A. Frieze and R. Kannan, Log-Sobolev inequalities and sampling from log-concave distributions, The Annals of Applied Probability, **9** (1999), 14–26.

[35] 伏見正則，確率と確率過程, 講談社，1987．

[36] 伏見正則，乱数, 東京大学出版会，1989．

[37] E. Gelenbe and G. Pujolle, Introduction to Queueing Networks, Second Edition, John Wiley & Sons, Inc, 1998.

[38] W. J. Gordon and G. F. Newell, Closed queueing systems with exponential servers, Operations Research, **15** (1967), 254–265.

[39] J. Graham, J. Curran and B. S. Weir, Conditional genotypic probabilities for microsatellite loci, Genetics, **155** (2000), 1973–1980.

[40] V. Guruswami, Rapidly mixing Markov chains: a comparison of techniques (a survey), 2000.
(available from http://www.cs.washington.edu/homes/venkat/pubs/papers.html)

[41] O. Häggström, Finite Markov Chains and Algorithmic Application, Cambridge University Press, 2002.

[42] D. Hernek, Random generation of $2 \times n$ contingency tables, Random Structures and Algorithms, **13** (1998), 71–79.

[43] 廣津千尋, 離散データ解析，教育出版，1982．

[44] W. Höffding, Probability inequalities for sums of bounded random variables, Journal of American Statistical Association, **58** (1963), 13–30.

[45] 伊庭幸人, 統計学者・数理工学者のための統計物理入門—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して— (改訂版), Reseach Memorandum, 635, Institute of Statistical Mathematics, 1997.

[46] R. Irving and M. Jerrum, Three-dimensional statistical data security problems, SIAM Journal on Computing, **23** (1994), 170–184.

[47] J. R. Jackson, Networks of waiting lines, The Journal of Operations Research Society of America, **5** (1957), 518–521.

[48] J. R. Jackson, Jobshop-like queueing systems, Management Science, **10** (1963), 131–142.

[49] M. Jerrum, Counting, Sampling and Integrating: Algorithms and Complexity, ETH Zürich, Birkhauser, Basel, 2003.

[50] M. Jerrum and A. Sinclair, The Markov chain Monte Carlo method: an approach to approximate counting and integration, in Approximation Algorithm for NP-hard Problems, D. Hochbaum, (ed.), PWS, 1996, 482–520.

[51] M. Jerrum, A. Sinclair, and E. Vigoda, A polynomial-time approximation algorithm for the permanent of a matrix with non-negative entries, Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC 2001), 712–721.

[52] A. Kalai and S. Vempala, Efficient algorithms for universal portfolio, Journal of Machine Learning Research, **3** (2002), 423–440.

[53] 亀田壽夫，紀一誠，李頡，性能評価の基礎と応用, 共立出版, 1998.

[54] R. Kannan, L. Lovász, and M. Simonovitz, Random walks and an $O^*(n^5)$ volume algorithm for convex bodies, Random Structures and Algorithms, **11** (1997), 1–50.

[55] R. Karp and M. Luby, Monte Carlo algorithms for enumeration and reliability problems, Proceedings of 24th IEEE Annual Symposium on Foundations of Computer Science (FOCS 1983), 56–64.

[56] R. Karp, M. Luby, and N. Madras, Monte Carlo approximation algorithms for enumeration problems, Journal of Algorithms, **10** (1989), 429–448.

[57] F. Kelly, Loss networks, Annals of Applied Probability, **1** (1991), 319–378.

[58] 来嶋秀治, 2 元分割表の個数数え上げ問題, 東京大学 工学部 計数工学科 卒業論文, 2002.

[59] 来嶋秀治, パーフェクトサンプリング法の設計, 東京大学 大学院情報理工学系研究科 数理情報学専攻 修士論文, 2004.

[60] S. Kijima and T. Matsui, Approximate counting scheme for $m \times n$ contingency tables, IEICE Transactions on Information and Systems, **E87**-D (2004), 308–314.

[61] S. Kijima and T. Matsui, Polynomial time perfect sampling algorithm for two-rowed contingency tables, Random Structures and Algorithms, **29** (2006), 243–256.

[62] S. Kijima and T. Matsui, Rapidly mixing chain and perfect sampler for logarithmic separable concave distributions on simplex, DMTCS Proceedings Series, **AD** (2005), 371–382.

[63] S. Kijima and T. Matsui, Approximate/perfect samplers for closed Jackson networks, Proceedings of the 2005 Winter Simulation Conference (WSC 2005), 862–868.

[64] S. Kijima and T. Matsui, Polynomial-time randomized approximation and perfect sampler for closed Jackson networks with single servers, Mathematical Engineering Technical Reports, 2005-12, University of Tokyo, 2005.

[65] S. Kijima and T. Matsui, Randomized approximation scheme and perfect sampler for closed Jackson networks with multiple servers, Mathematical Engineering Technical Reports, 2006-34, University of Tokyo, 2006.

[66] 来嶋 秀治, 松井 知己, 平衡状態を探す：マルコフ連鎖/CFTP, 数学セミナー, 43 巻 (2004), 8 号, 42–46.

[67] 来嶋 秀治, 松井 知己, 完璧にサンプリングしよう !, オペレーションズ・リサーチ, **50** (2005), 169–174, 264–269, 329–334.

[68] 来嶋 秀治, 松井 知己, CFTP を用いたパーフェクトサンプリング, 田中和之 編著, 確率的情報処理と統計力学, 臨時別冊・数理科学 SGC ライブラリ 50, サイエンス社, 2006, 80–89.

[69] 紀一誠 , 待ち行列ネットワーク, 朝倉書店, 2002.

[70] S. Kitada, T. Hayashi, and H. Kishino, Empirical Bayes procedure for estimating genetic distance between populations and effective population size, Genetics, **156** (2000), 2063–2079.

[71] G. Laval, M. SanCristobal, and C. Chevalet, Maximum-likelihood and Markov chain Monte Carlo approaches to estimate inbreeding and effective size form allele frequency changes, Genetics, **164** (2003), 1189–1204.

[72] L. Lovász and S. Vempala, Logconcave functions: geometry and efficient sampling algorithms, Proceedings of the 44th Symposium on Foundations of Computer Science (FOCS 2003), 640–649.

[73] L. Lovász and S. Vempala, Hit-and-run is fast and fun, Microsoft Research Technical Report, 2003-05, Microsoft Research, 2003.

[74] L. Lovász and S. Vempala, Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithms, Microsoft Research Technical Report, 2003-31, Microsoft Research, 2003.

[75] T. Matsui and S. Kijima, Polynomial time perfect sampler for discretized Dirichlet distribution, Mathematical Engineering Technical Reports, 2003-17, University of Tokyo, 2003.

[76] T. Matsui, Y. Matsui, and Y. Ono, Random generation of $2 \times 2 \times \cdots \times 2 \times J$ contingency tables, Theoretical Computer Science, **326** (2004), 117–135.

[77] T. Matsui, M. Motoki, and N. Kamatani, Polynomial time approximate sampler for discretized Dirichlet distribution, 14th International Symposium on Algorithms and Computation (ISAAC 2003), Lecture Notes in Computer Science, **2906** (2003), 676–685.

[78] T. Matsui, M. Motoki, N. Kamatani, and S. Kijima, Polynomial time approximate/perfect samplers for discretized Dirichlet distribution, Mathematical Engineering Technical Reports, 2006-09, University of Tokyo, 2006.

[79] M. Matsumoto and T. Nishimura, Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator, ACM Transactions on Modeling and Computer Simulation, **8** (1998), 3–30.

[80] Mersenne Twister Home Page, http://www.math.keio.ac.jp/~matumoto/mt.html

[81] B. Morris and A. Sinclair, Random walks on truncated cubes and sampling 0-1 knapsack solutions, SIAM Journal on Computing, **34** (2005), 195–226.

[82] 元木光雄, 鎌谷直之, Dirichlet 分布に従う多項分布の母数の Markov chain を用いた approximate sampler, 2001 年冬の LA シンポジウム.

[83] R. Motowani and P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.

[84] R. Montenegro and P. Tetali, Mathematical Aspects of Mixing Times in Markov Chinas, Now Publishers, 2006.

[85] M. Mitzenmacher and E. Upfal, Probability and Computing, Randomized Algorithms and Probabilistic Analysis, Cambridge University Press, 2005.

[86] K. Murota, Matrices and Matroids for Systems Analysis, Springer, 2000.

[87] 室田 一雄，離散凸解析，共立出版, 2001.

[88] K. Murota, Discrete Convex Analysis, SIAM, 2003.

[89] T. Niu, Z. S. Qin, X. Xu and J. S. Liu, Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms, American Journal of Human Genetics, **70** (2002), 157–169.

[90] 大森裕浩, 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** (2001), 305–344.

[91] T. Ozawa, Perfect simulation of a closed Jackson network, ORSJ Queueing symposium, Hikone, 2004.

[92] 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, 104–105.

[93] J. K. Pritchard, M. Stephens, and P. Donnelly, Inference of population structure using multilocus genotype data, Genetics, **155** (2000), 945–959.

[94] J. Propp and D. Wilson, Exact sampling with coupled Markov chains and applications to statistical mechanics, Random Structures and Algorithms, **9** (1996), 223–252.

[95] J. Propp and D. Wilson, How to get a perfectly random sample from a generic Markov chain and generate a random spanning tree of a directed graph, Journal of Algorithms, **27** (1998), 170–217.

[96] D. Randall and P. Winkler, Mixing points on an interval, Proceedings of the Second Workshop on Analytic Algorithms and Combinatorics (ANALCO 2005), 216–221.

[97] C. P. Robert, The Bayesian Choice, Springer, 2001.

[98] K. W. Ross and J. Wang, Asymptotically optimal importance sampling for product-form queuing networks, ACM Transactions on Modeling and Computer Simulation **3** (1993), 244–268.

[99] 坂田年男, 澤江隆一, 計算機統計学と数学の連携 (分割表の解析の最近の話題から), 日本計算機統計学会第 16 回大会論文集, 2002, 92–101.

[100] A. Sinclair, Improved bounds for mixing rates of Markov chains and multicommodity flow, Combinatorics, Probability and Computing, **1** (1992), 351–370.

[101] A. Sinclair, Algorithms for Random Generation and Counting: A Markov Chain Approach, Birkhäuser, Boston, 1993.

[102] A. Sinclair and M. Jerrum, Approximate counting, uniform generation and rapidly mix-

ing Markov chains, Information and Computation, **82** (1989), 93–133.

[103] L. G. Valiant, The complexity of computing the permanent, Theoretical Computer Science, **8** (1979), 189–201.

[104] V. V. Vazirani, Approximation Algorithms, Springer, 2003.

[105] Web Site for Perfectly Random Sampling with Markov Chains, http://research.microsoft.com/~dbwilson/exact/

[106] D. Wilson, How to couple from the past using a read-once source of randomness, Random Structures and Algorithms, **16** (2000), 85–113.

[107] D. Wilson, Mixing times of lozenge tiling and card shuffling Markov chains, Annals of Applied Probability, **14** (2004), 274–325.