13. Coupling from the past

来嶋 秀治

滋賀大学 データサイエンス学部

最終日の話題

- ■ ここまではMCMC法を紹介した。
- □ ここからの話
  13. 完璧サンプリング法 (さらに進んだ研究)
  14. 決定性近似 (批判的研究)
  15. 増えるクーポン収集 (別の話題への波及)

2行分割表の完璧サンプリング

Shuji Kijima and Tomomi Matsui, Polynomial time perfect sampling algorithm for two-rowed contingency tables, Random Structures & Algorithms, 29(2), 243—256, 2006.

例: 2行分割表のランダム生成

2元分割表

- ✓ 各セルには非負整数が入る
- ✓ （与えられた）周辺和を満たす

						12
						18
5	4	3	7	5	6	30

問題

Given: 周辺和

出力: 分割表の一様ランダム生成

例: 2行分割表のランダム生成

2元分割表

- ✓ 各セルには非負整数が入る
- ✓ （与えられた）周辺和を満たす

						12
						18
5	4	3	7	5	6	30

5	4	3	0	0	0	12
0	0	0	7	5	6	18
5	4	3	7	5	6	30

状態A

4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30

状態B

0	0	0	1	5	6	12
5	4	3	6	0	0	18
5	4	3	7	5	6	30

状態C

問題

Given: 周辺和

出力: 分割表の一様ランダム生成

例: 2行分割表のランダム生成

2元分割表

- ✓ 各セルには非負整数が入る
- ✓ （与えられた）周辺和を満たす

						12
						18
5	4	3	7	5	6	30

与えられた周辺和を満たす2行分割表の**個数を求める問題**

⇒#P完全 (NP困難) ['97 Dyer, Kannan, & Mount]

問題

Given: 周辺和

出力: 分割表の一様ランダム生成

マルコフ連鎖を用いたサンプリング法

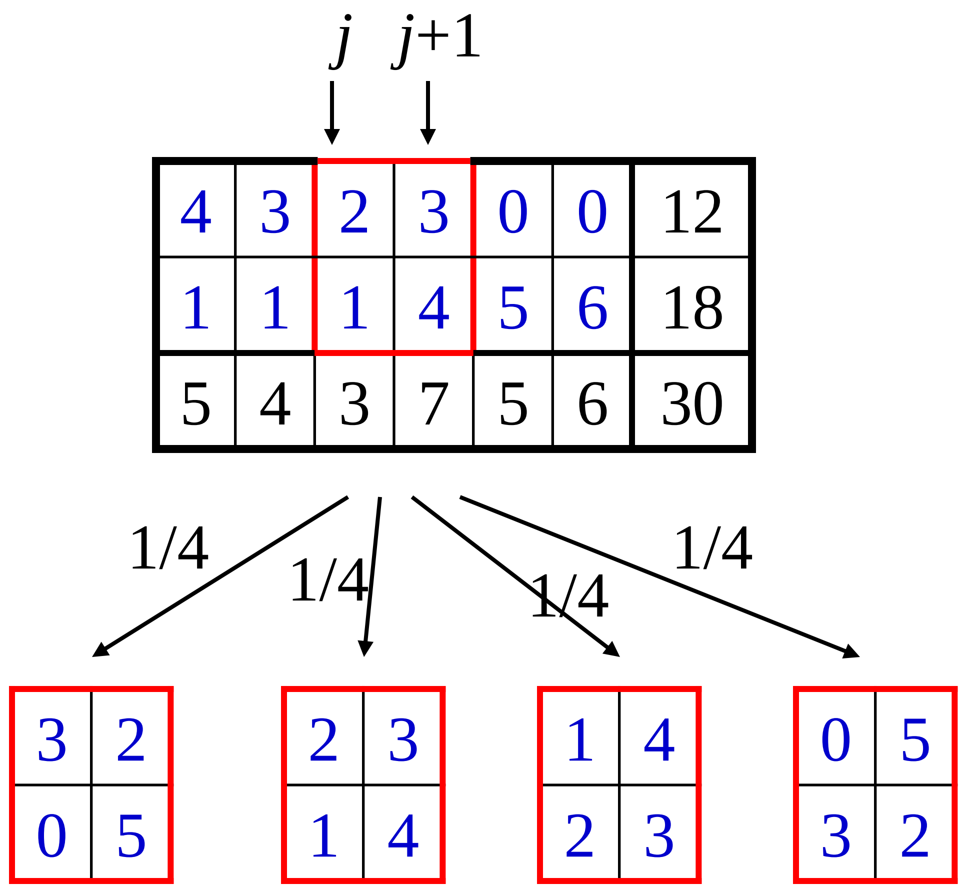
例: 2行分割表に対するマルコフ連鎖 [K & Matsui '06]

- j 列目（とj+1列目）を 1/(n –1)の確率で選ぶ。
- j 列目と j+1列目に対して推移可能な状態に等確率で推移する。

2	3	5
1	4	5
3	7	10

+

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



提案するマルコフ連鎖の特徴

エルゴード的

1. 有限
2. 既約
3. 非周期

定理

提案したマルコフ連鎖 の定常分布は一様分布である.

略証: $\forall(X, Y),\ P(X, Y) > 0 \ \Rightarrow\ \mathrm{P}(Y, X) > 0$ かつ $P(X, Y) = P(Y, X)$

X

4	3	2	3	0	0	12
1	1	1	4	5	6	18
5	4	3	7	5	6	30

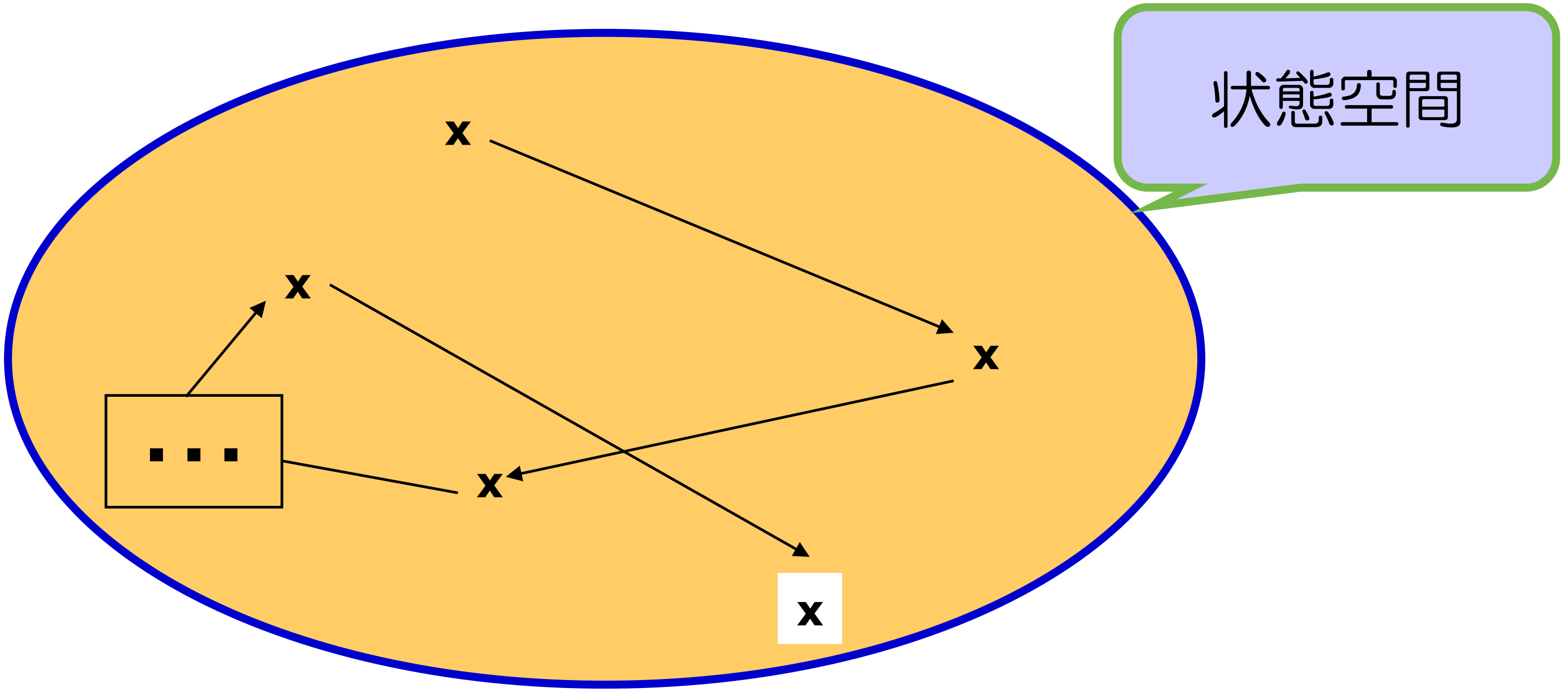
Y

4	3	0	5	0	0	12
1	1	3	2	5	6	18
5	4	3	7	5	6	30

(detailed balance equation $\pi(X)\, P(X, Y) = \pi(Y)\, P(Y, X)$)

「マルコフ連鎖を用いたサンプリング法」のアイデア

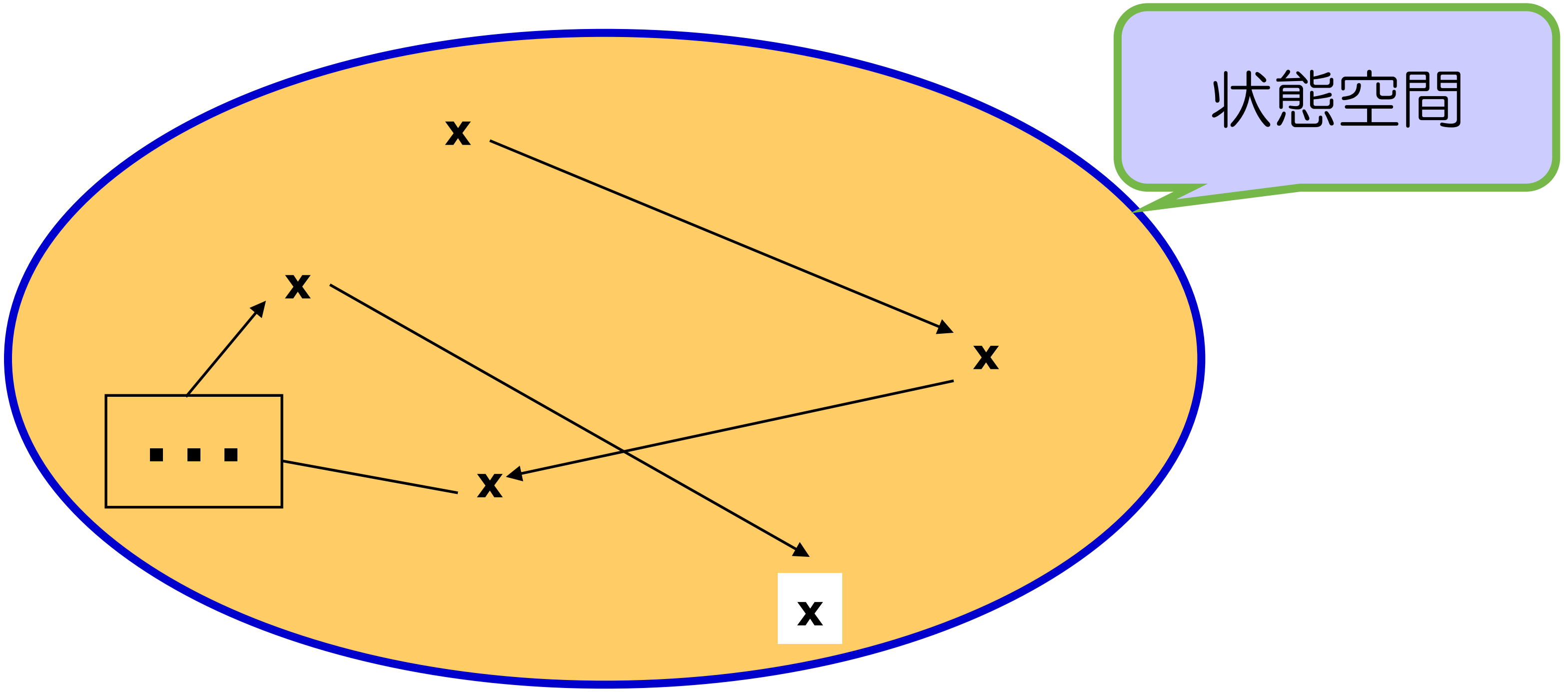
1. 目的の分布を定常分布にもつマルコフ連鎖を設計する。
2. 十分な回数推移させて、定常分布からサンプリングする。



漸近的に定常分布

「マルコフ連鎖を用いたサンプリング法」のアイデア

1. 目的の分布を定常分布にもつマルコフ連鎖を設計する。
2. 十分な回数推移させて、定常分布からサンプリングする。



漸近的に定常分布 => 有限回で停止すると誤差が生じる

研究の主題

「十分な回数推移させて，定常分布からサンプリングする．」

⇒ 何回推移させれば十分か？

- 近似サンプリング
  - total variation distance
  - mixing time
- 完璧サンプリング（定常分布に厳密に従う）
  - CFTP (Coupling From The Past)
    - ✓ マルコフ連鎖のシミュレーション法を工夫
    - ✓ 定常分布に厳密に従う

研究の主題

「十分な回数推移させて，定常分布からサンプリングする．」

⇒ 何回推移させれば十分か？

- 近似サンプリング
  - total variation distance
  - mixing time
- 完璧サンプリング（定常分布に厳密に従う）
  - CFTP (Coupling From The Past)
    - ✓ マルコフ連鎖のシミュレーション法を工夫
    - ✓ 定常分布に厳密に従う

「2元分割表のランダム生成法」に関連する研究

1985, Diaconis & Effron, 2元分割表の検定法の提案,

1995, Diaconis & Saloff-Coste, $m^* \times n^*$ 分割表の近似一様生成,

(1996, Propp & Wilson, CFTPアルゴリズム,)

1997, Dyer, Kannan & Mount, 2行分割表の **#P完全性**,

2000, Dyer & Greenhill, $2 \times n$ 分割表の近似一様生成, $O(n^2 \log(N\varepsilon^{-1}))$

2002, Cryan et al., $m^* \times n$ 分割表の近似一様生成, $O(n^{m2} \log(N \varepsilon^{-1}))$

2006, Kijima & Matsui, $2 \times n$ 分割表の厳密一様生成, $O(n^3 \log N)$

$\Rightarrow$ $m \times n$ 分割表の多項式時間近似一様生成はmajor open problem

本題

「十分な回数推移させて，定常分布からサンプリングする．」

⇒ 何回推移させれば十分か？

- 近似サンプリング
  - total variation distance
  - mixing time
- 完璧サンプリング (perfect sampling)
  - CFTP (Coupling From The Past)
    - ✓ 無限回の推移の後の状態を実現

      ⇒ 定常分布に厳密に従う

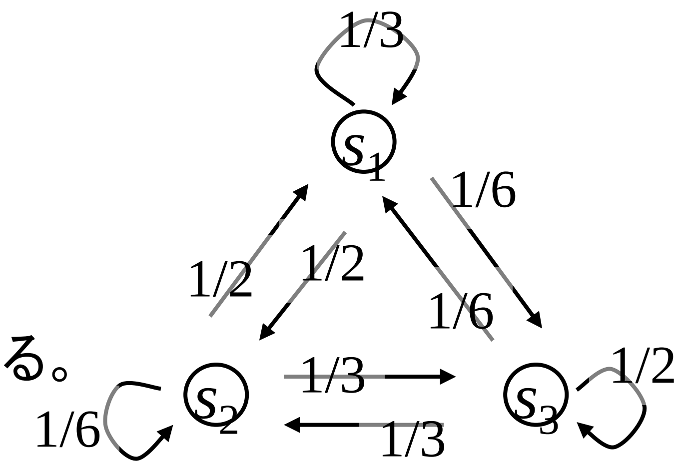
誤差パラメータを自分で設定する必要がない。

-3: Coupling From The Past

- Propp and Wilson (1996)
- マルコフ連鎖のシミュレーションを工夫

  (**更新関数**を利用)

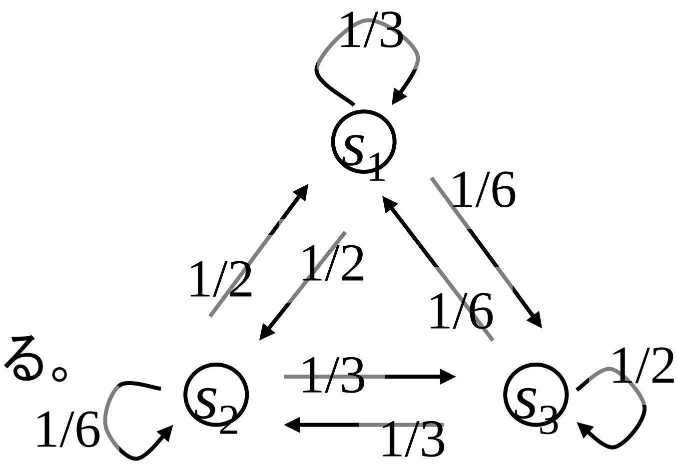
更新関数 （update function）



- マルコフ連鎖 MC （エルゴード的）
  - 状態空間: s_1, s_2, s_3 ; 有限 (3 状態)
  - 推移規則: 乱数 $\lambda \in \{1,\ldots,6\}$ に対して決まる。

更新関数 （update function）

- マルコフ連鎖 MC (エルゴード的)
  - 状態空間: s_1, s_2, s_3 ; 有限 (3 状態)
  - 推移規則: 乱数 $\lambda \in \{1,\ldots,6\}$ に対して決まる。



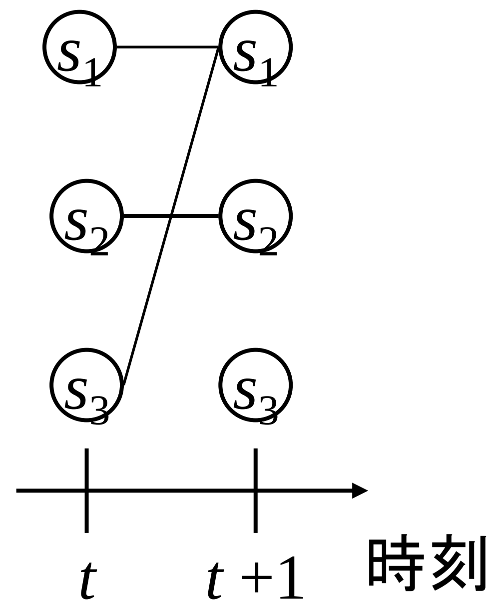
更新関数

現在の状態

乱数

	s_1	s_2	s_3
1	s_3	s_1	s_2
2	s_2	s_3	s_2
3	s_2	s_1	s_3
4	s_1	s_1	s_3
5	s_1	s_2	s_1
6	s_2	s_3	s_3

例　5 ◀ λ



CFTPアルゴリズムと定理

マルコフ連鎖 MC:
- 有限離散の状態空間 Ω
- 更新関数 $\Phi_s^t(x, \lambda)$ ← 時刻 s, 状態 x から時刻 t まで推移(乱数列 λ)
- エルゴード性を満たす。

CFTPアルゴリズム

1. set $T = -1$; set λ: 空列;
2. generate $\lambda[T]$: 乱数; put $\lambda := (\lambda[T], \lambda[T+1], \ldots, \lambda[-1])$;
3. Ω の全ての状態について、時刻 T から 0 まで λ を用いてマルコフ連鎖 MC を推移させる。
   a. if coalesce $(\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^0(x, \lambda)) \Rightarrow$ return y;
   b. otherwise, set $T := T - 1$; step 2.に戻る;

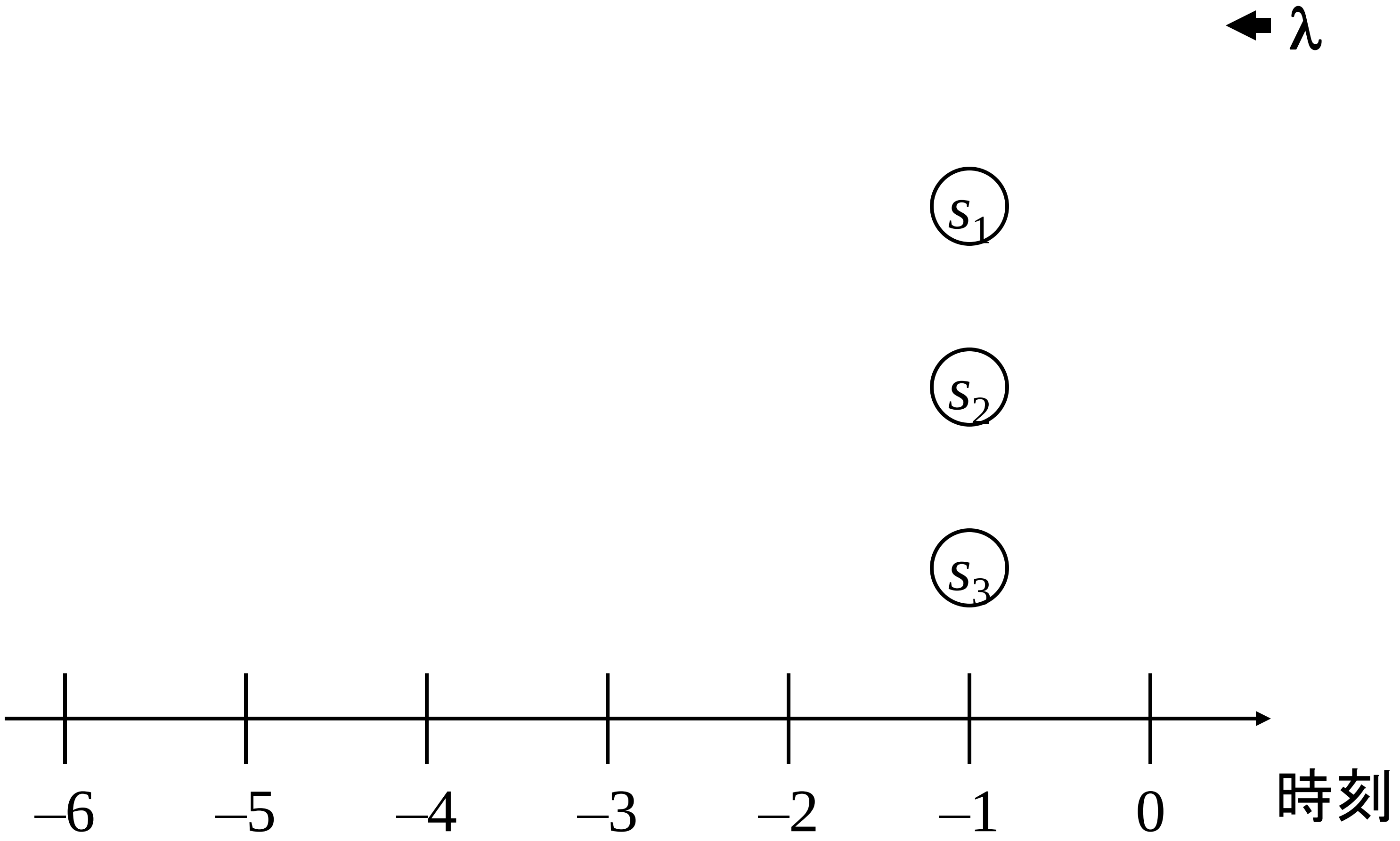
CFTP定理

上のアルゴリズムが停止して値を返す時、

その値はマルコフ連鎖 M の定常分布に厳密に従う確率変数を実現する。

$T = -1$からのシミュレーション

1. set $T = -1$; set λ: 空列;

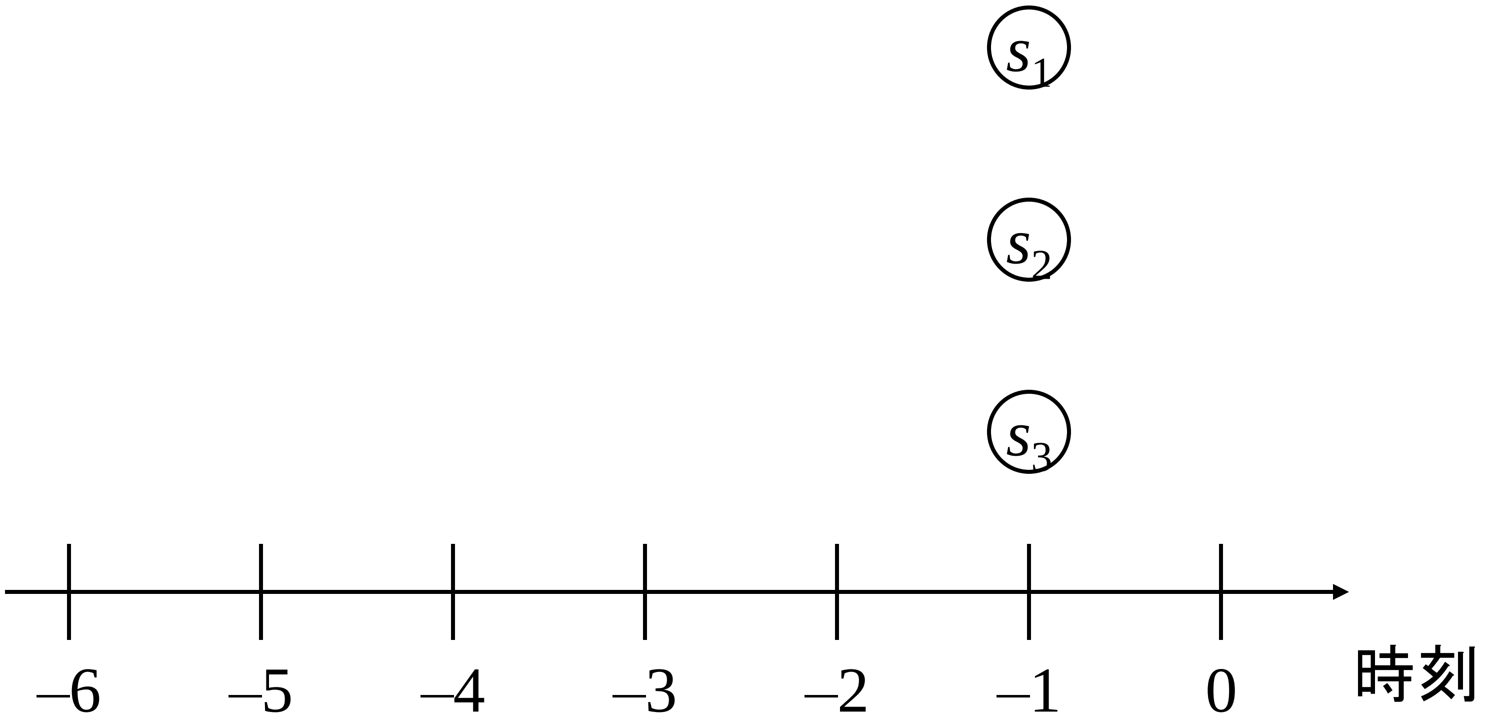


$T = -1$からのシミュレーション

2. generate $\lambda(T)$: 乱数;

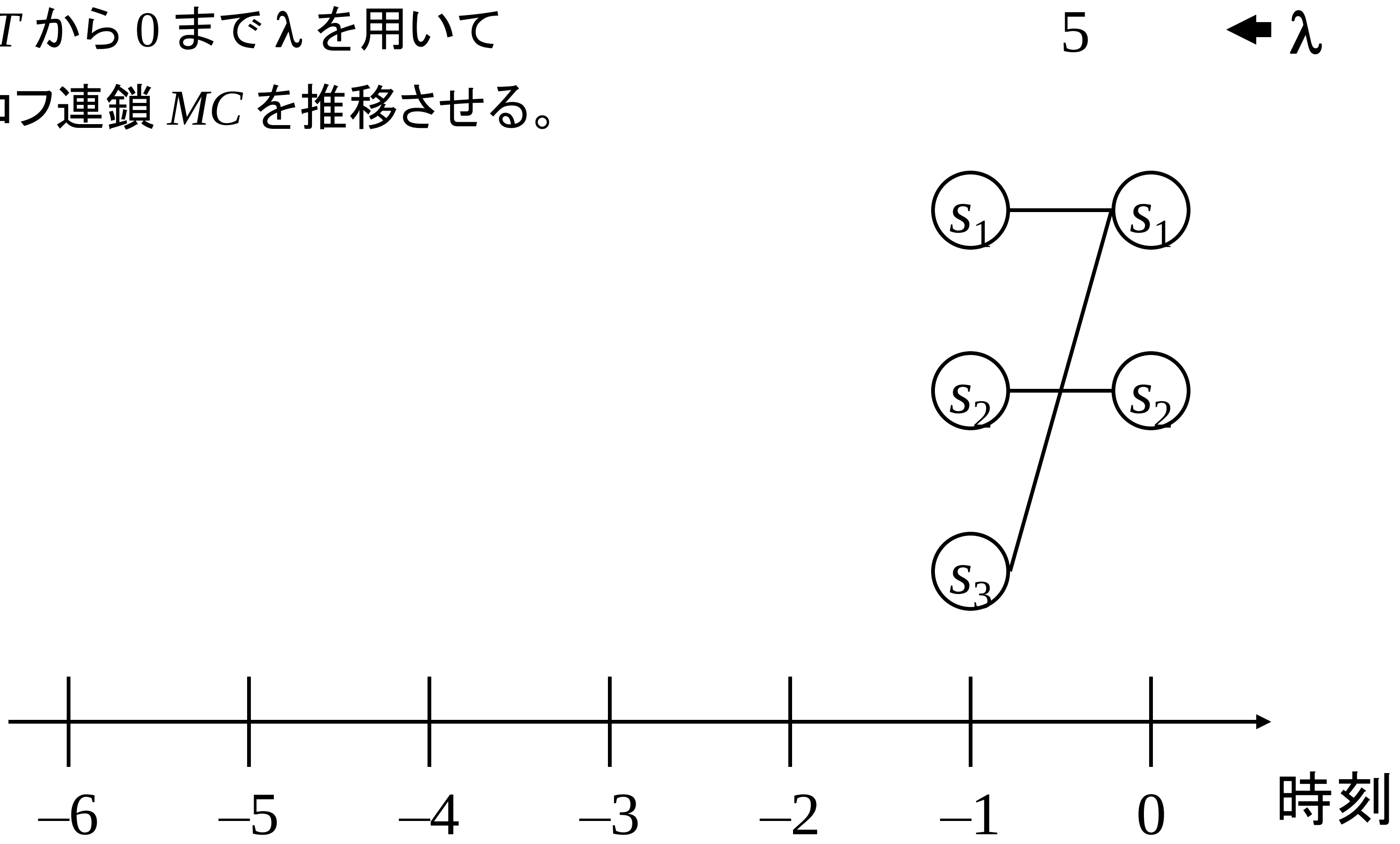
put $\boldsymbol{\lambda} := (\lambda(T), \lambda(T+1),\ldots,\lambda(-1))$;

$\lambda(-1) = 5 \leftarrow \boldsymbol{\lambda}$



$T = -1$からのシミュレーション

3. Ω の全ての状態について、
時刻 T から 0 まで λ を用いて
マルコフ連鎖 MC を推移させる。



$T = -1$からのシミュレーション

3. Ω の全ての状態について、

時刻 T から 0 までλを用いて

マルコフ連鎖 MC を推移させる。

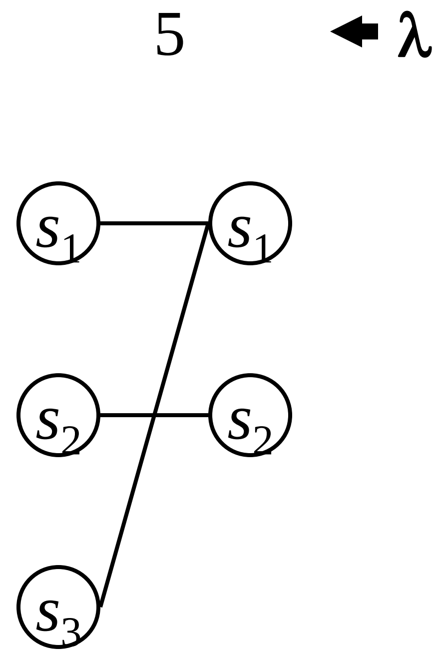
a. if coalesce

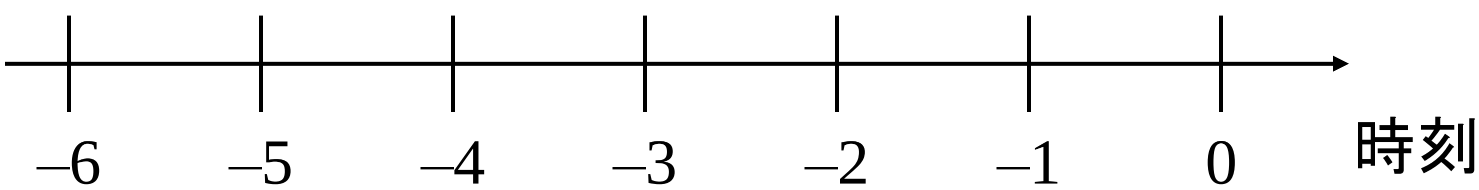
$(\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^0(x, \boldsymbol{\lambda}))$

$\Rightarrow$ return y;

b. otherwise, set $T := T - 1$;

step 2.に戻る;



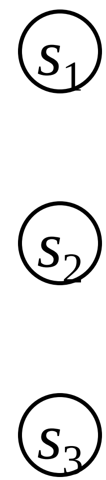


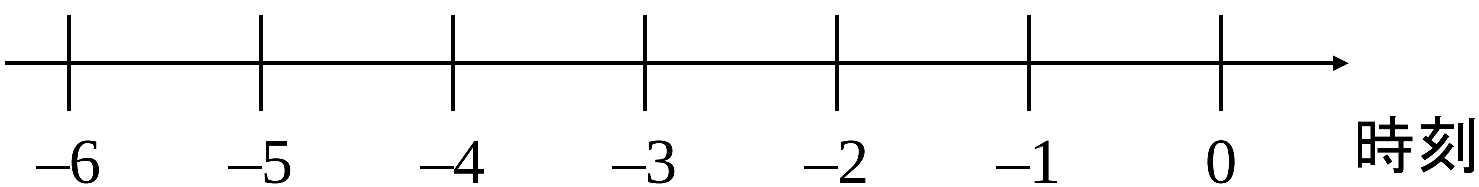
$T = -2$からのシミュレーション

3. Ω の全ての状態について、
   時刻 T から 0 まで λ を用いて
   マルコフ連鎖 MC を推移させる。

   a. if coalesce
      $(\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^0(x, \boldsymbol{\lambda}))$
      $\Rightarrow$ return y;
   b. otherwise, set $T := T - 1$;
      step 2.に戻る;

5 ◀ λ



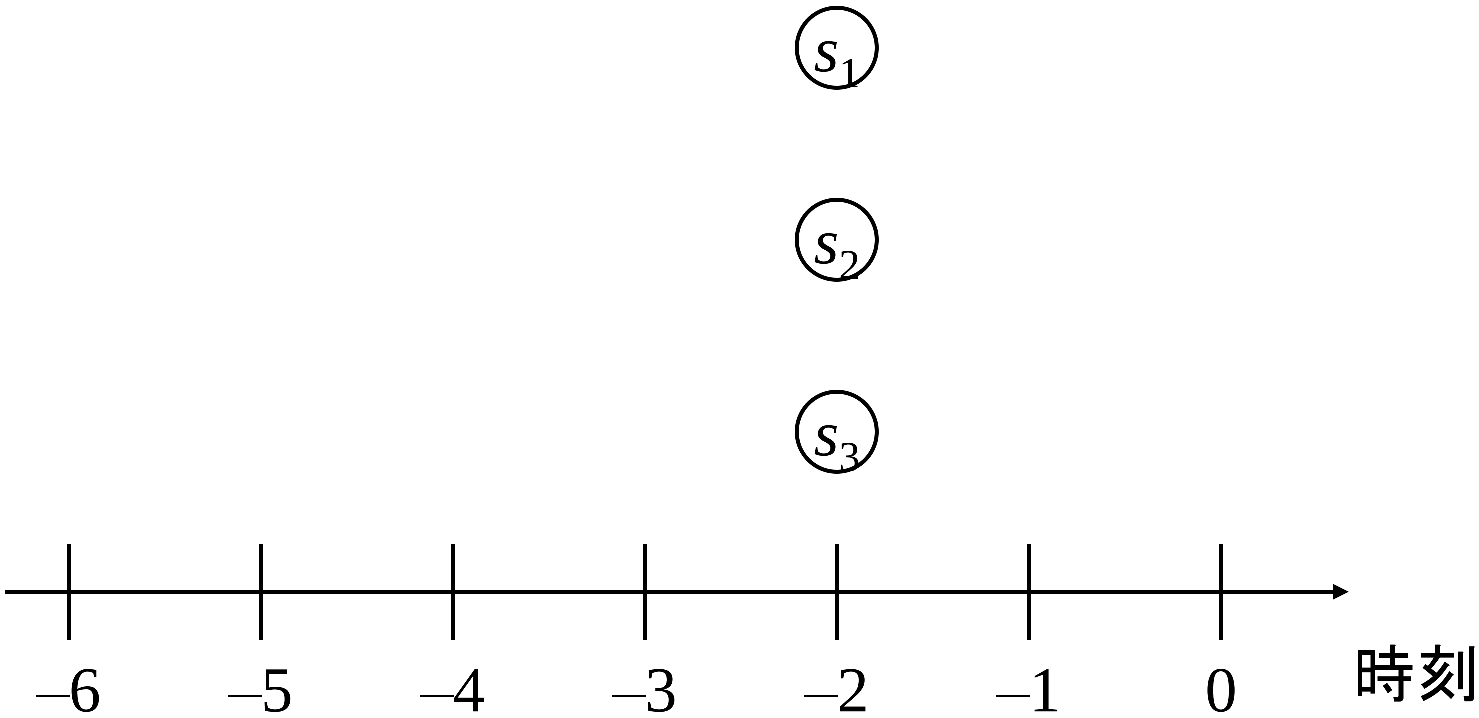


$T = -2$からのシミュレーション

2. generate $\lambda(T)$: 乱数;

put $\boldsymbol{\lambda} := (\lambda(T), \lambda(T+1), \ldots, \lambda(-1))$;

$\lambda(-2) = 2 \quad 5$ ← $\boldsymbol{\lambda}$

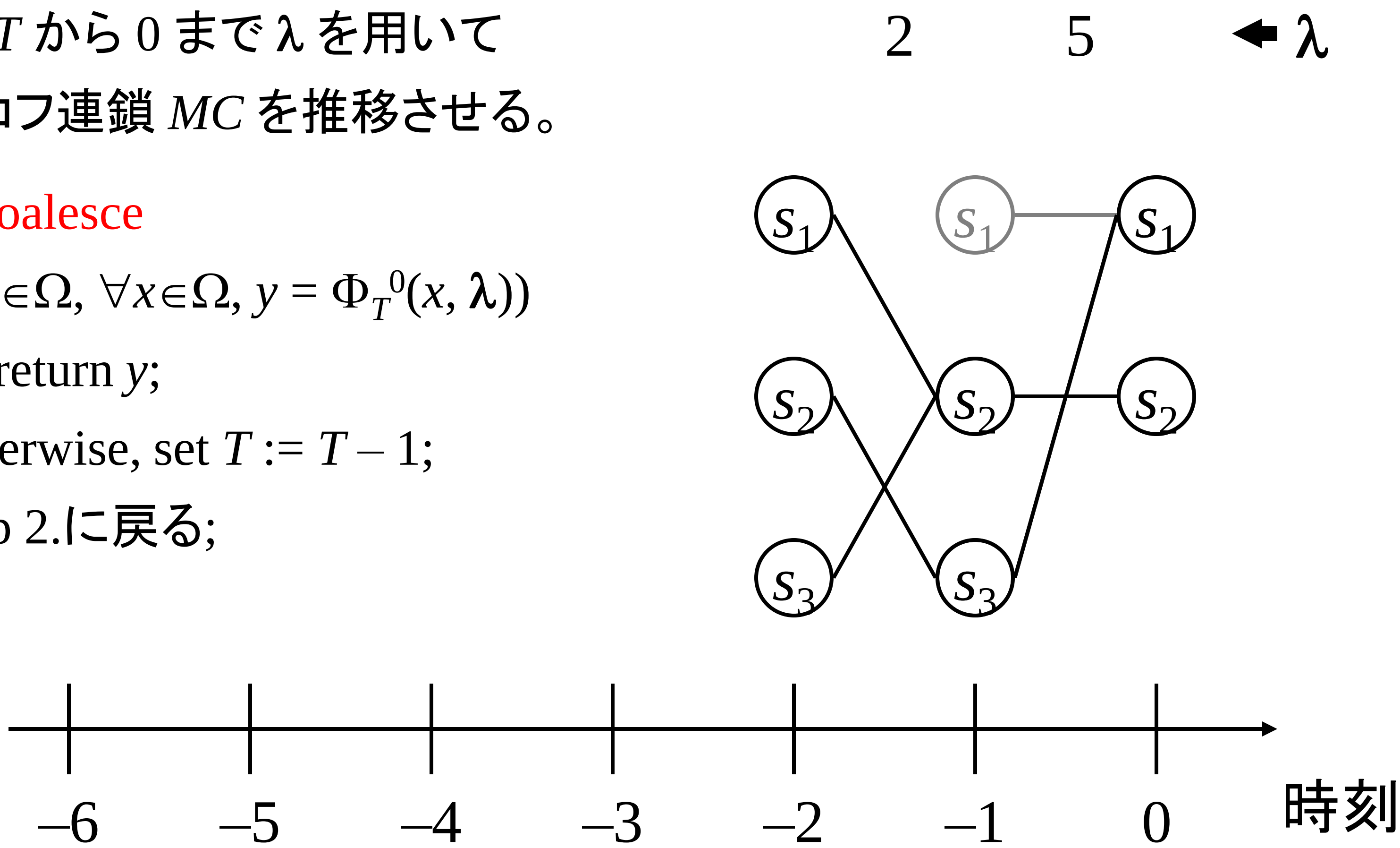


$T = -2$からのシミュレーション

3. Ω の全ての状態について、
   時刻 T から 0 まで λ を用いて
   マルコフ連鎖 MC を推移させる。

   a. if coalesce
   $(\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^{\,0}(x, \lambda))$
   $\Rightarrow$ return y;

   b. otherwise, set $T := T - 1$;
   step 2.に戻る;



$T = -3$からのシミュレーション

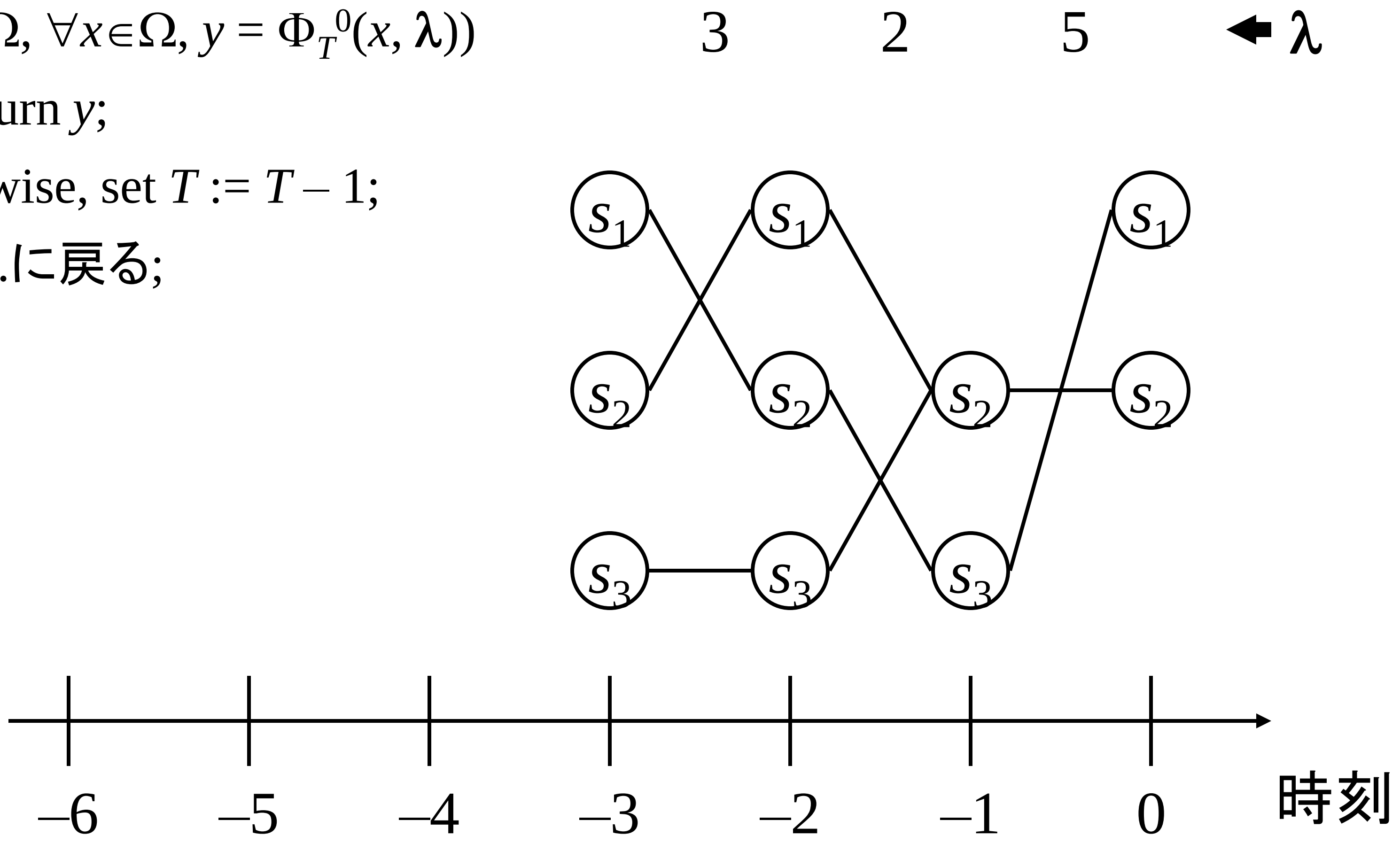
a. if coalesce

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

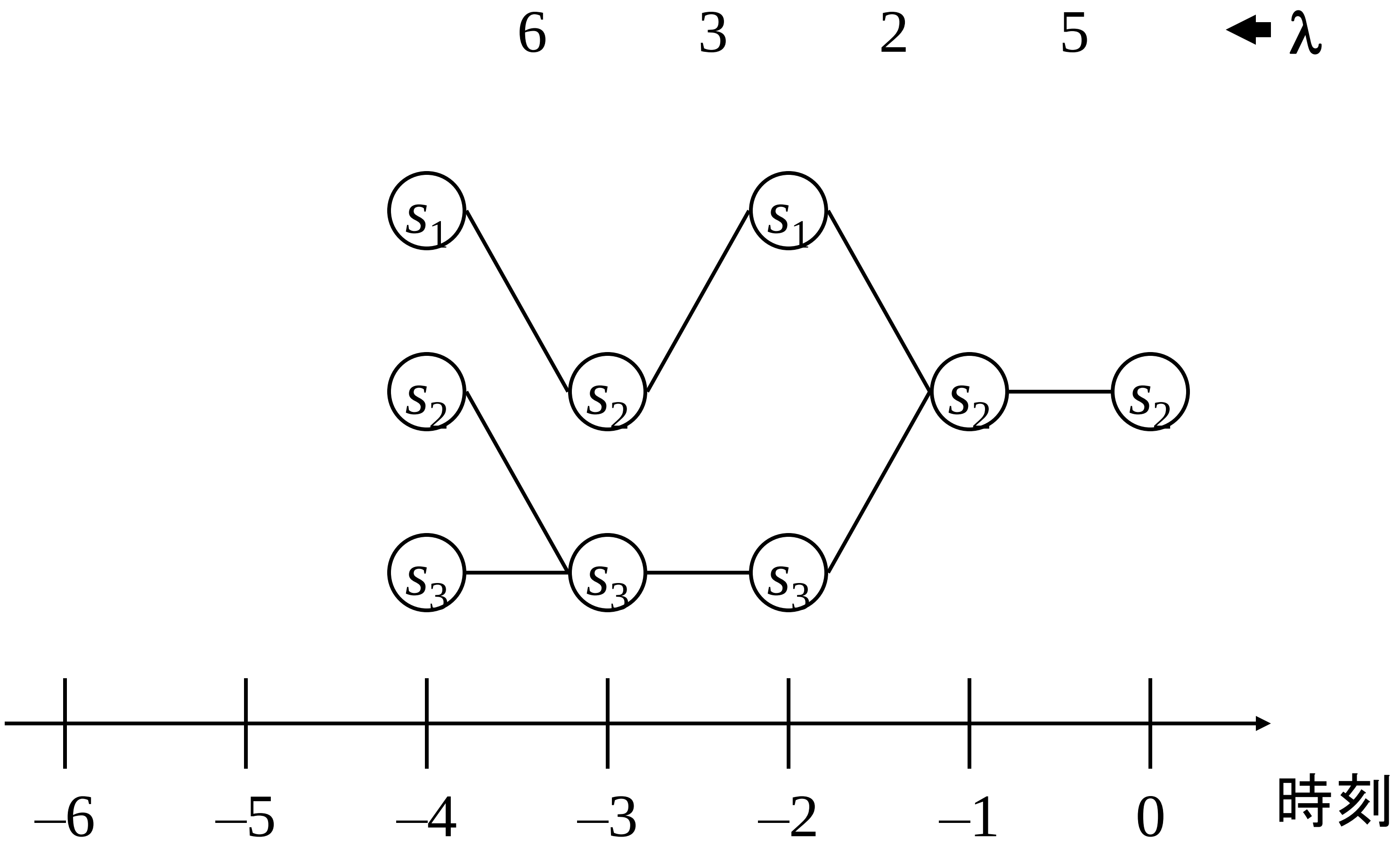
$\Rightarrow$ return y;

b. otherwise, set $T := T - 1$;

step 2.に戻る;



$T = -4$からのシミュレーション

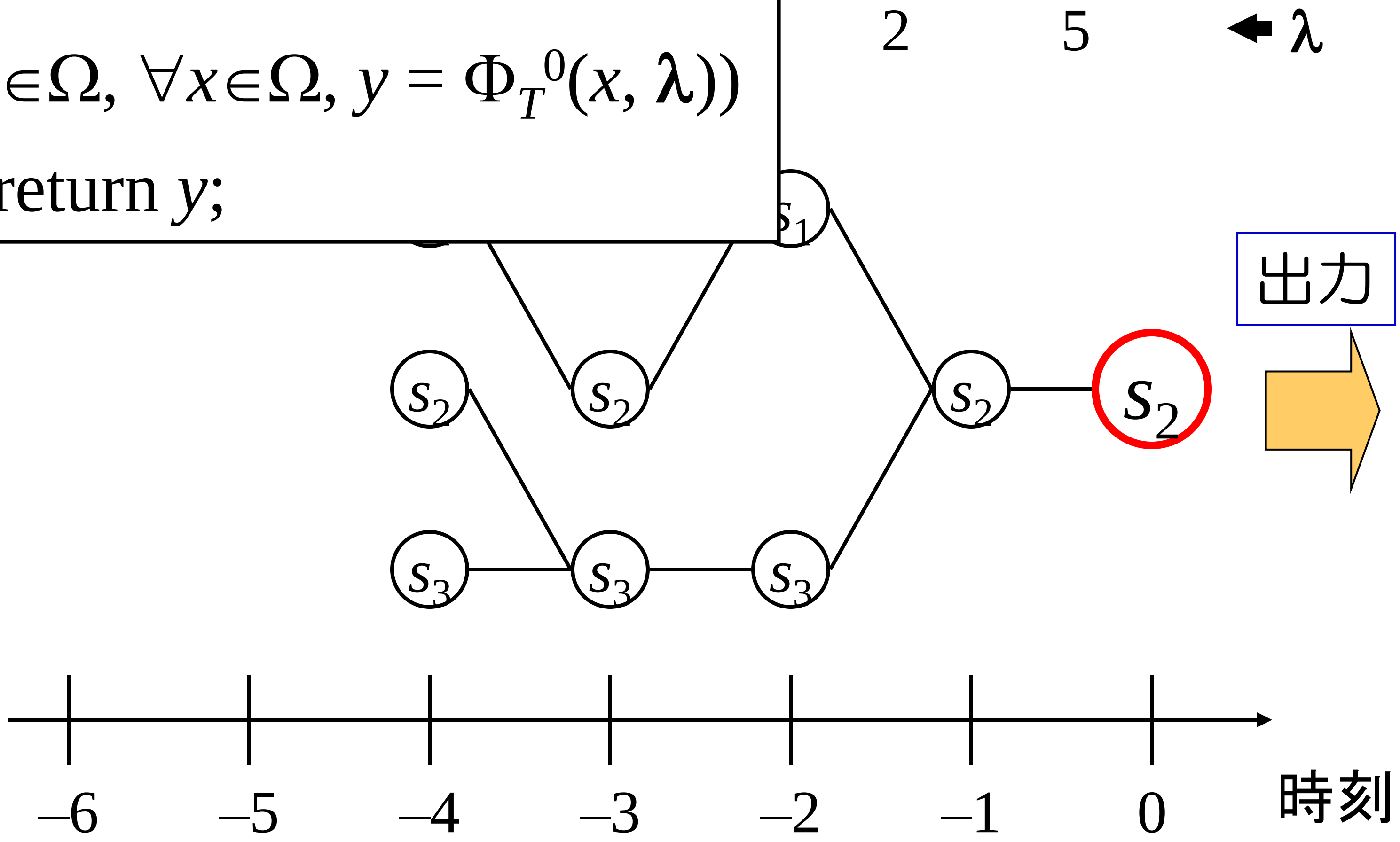


$T = -4$からのシミュレーション

a. if coalesce

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

$\Rightarrow$ return y;



CFTPアルゴリズムと定理

マルコフ連鎖 MC:
- 有限離散の状態空間 Ω
- 更新関数 $\Phi_s^t(x, \lambda)$ ← 時刻 s, 状態 x から時刻 t まで推移(乱数列 λ)
- エルゴード性を満たす。

CFTPアルゴリズム

1. set $T = -1$; set λ: 空列;
2. generate $\lambda[T]$: 乱数; put $\lambda := (\lambda[T], \lambda[T+1], \ldots, \lambda[-1])$;
3. Ω の全ての状態について、時刻 T から 0 まで λ を用いてマルコフ連鎖 MC を推移させる。
   a. if coalesce $(\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^0(x, \lambda)) \Rightarrow$ return y;
   b. otherwise, set $T := T - 1$; step 2.に戻る;

CFTP定理

上のアルゴリズムが停止して値を返す時、

その値はマルコフ連鎖 M の定常分布に厳密に従う確率変数を実現する。

CFTPのアイデア

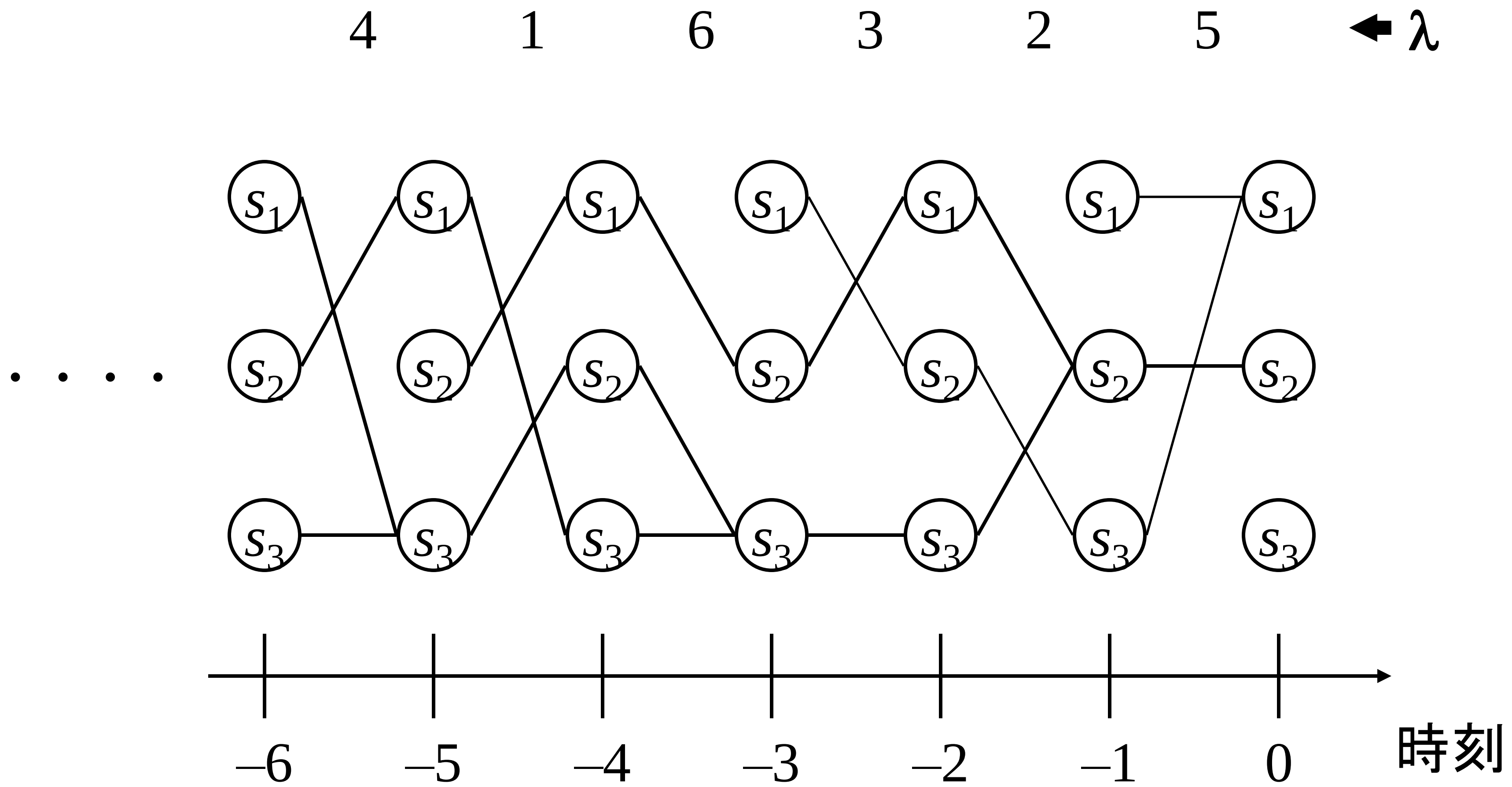
- 仮想的に無限の過去から推移を続けるマルコフ連鎖を考える。
  - 現在（時刻0）の状態は定常分布に厳密に従う。
- 現在（時刻0）の状態は何か？
  - 最近の乱数列から推定する。

    ⇒ 現在の状態を一意に特定する証拠を見つける。

coalescence

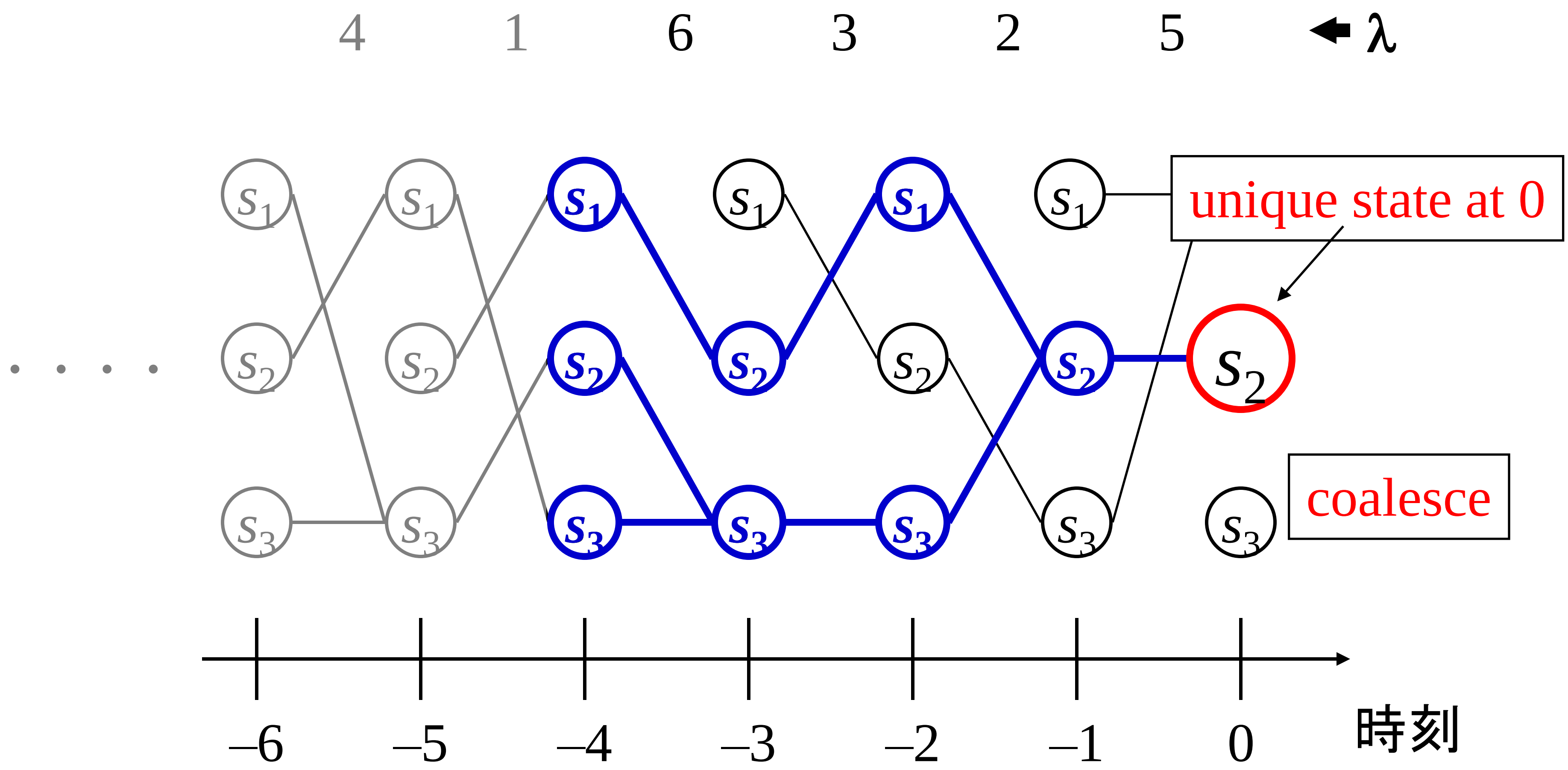
=> 乱数列と対応する推移に依存

CFTPのアイデア



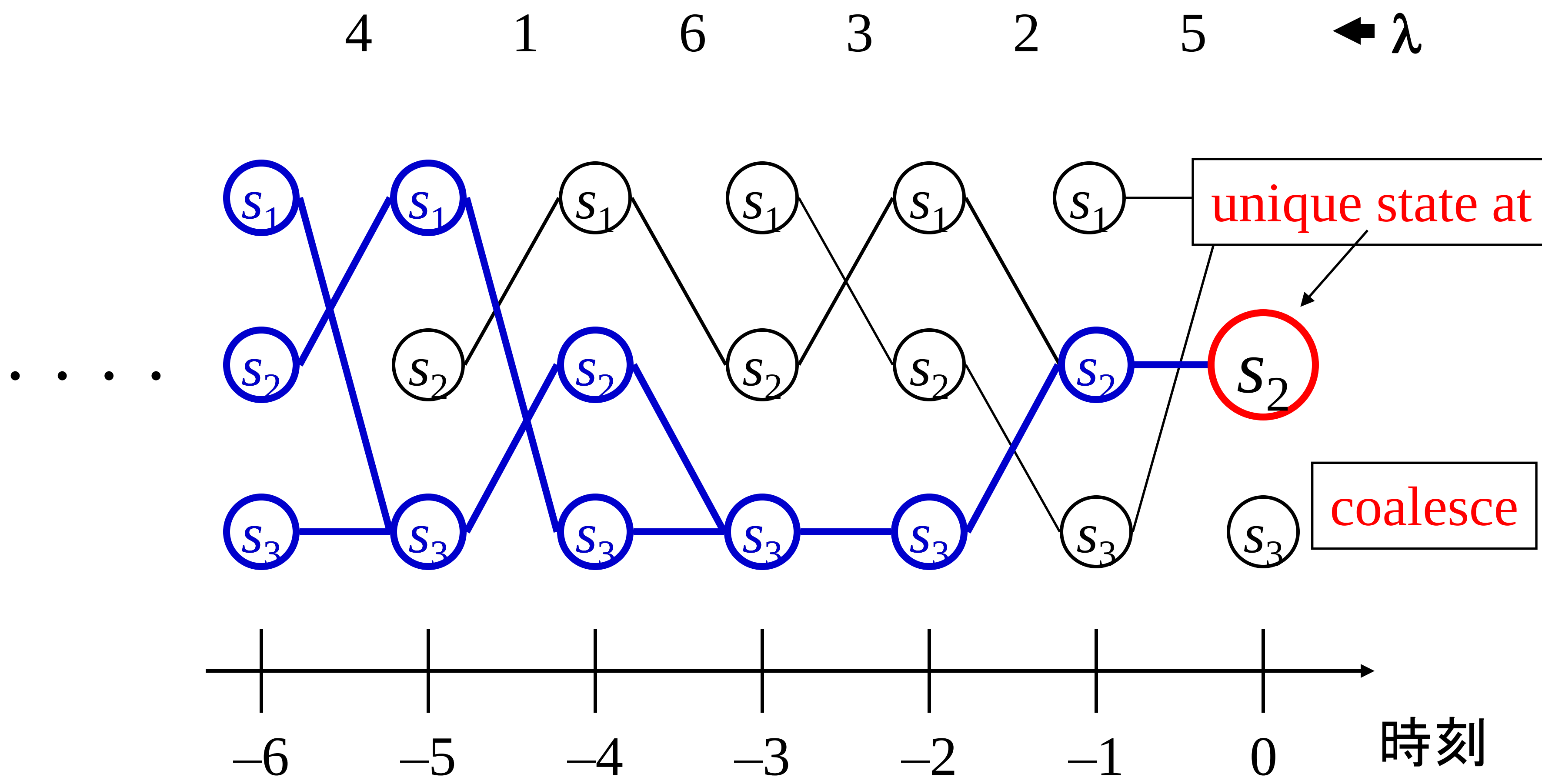
CFTPのアイデア

初期状態を知る必要がない !!



CFTPのアイデア

シミュレーションの開始時刻は、時刻0の状態に影響を及ぼさない



CFTPアルゴリズムと定理

マルコフ連鎖 MC: 有限離散の状態空間 Ω
更新関数 $\Phi_s^{\,t}(x, \lambda)$ ← 時刻 s, 状態 x から時刻 t まで推移 (乱数列 λ)
エルゴード性を満たす。

CFTPアルゴリズム

1. set $T = -1$; set λ: 空列;
2. generate $\lambda[T]$: 乱数; put $\lambda := (\lambda[T], \lambda[T+1], \ldots, \lambda[-1])$;
3. Ω の全ての状態について、時刻 T から 0 まで λ を用いてマルコフ連鎖 MC を推移させる。
   a. if coalesce $(\exists y \in \Omega, \forall x \in \Omega, y = \Phi_T^{\,0}(x, \lambda)) \Rightarrow$ return y;
   b. otherwise, set $T := T - 1$; step 2.に戻る;

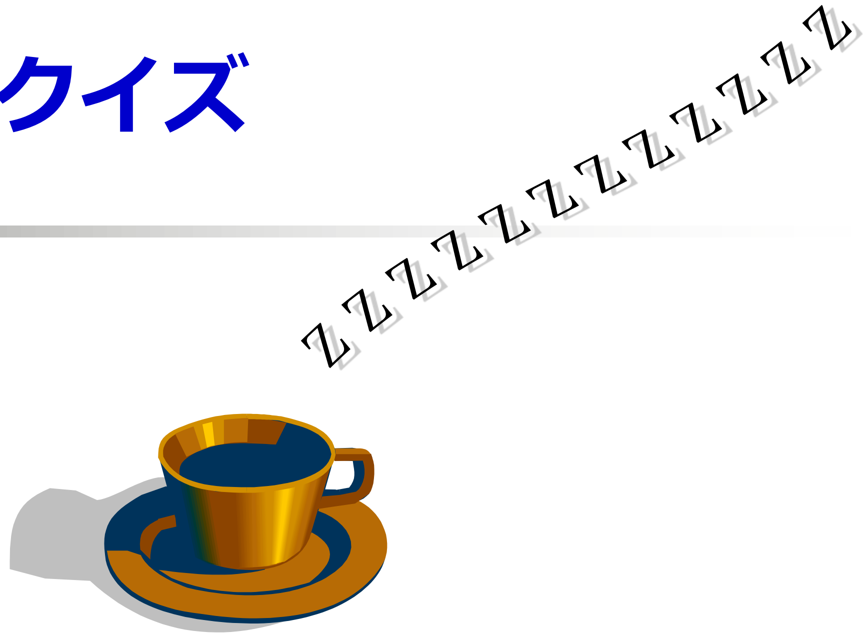
CFTP定理

上のアルゴリズムが停止して値を返す時、

その値はマルコフ連鎖 M の定常分布に<u>厳密に従う</u>確率変数を実現する。

Intermediate：　○×クイズ

Coffee Break

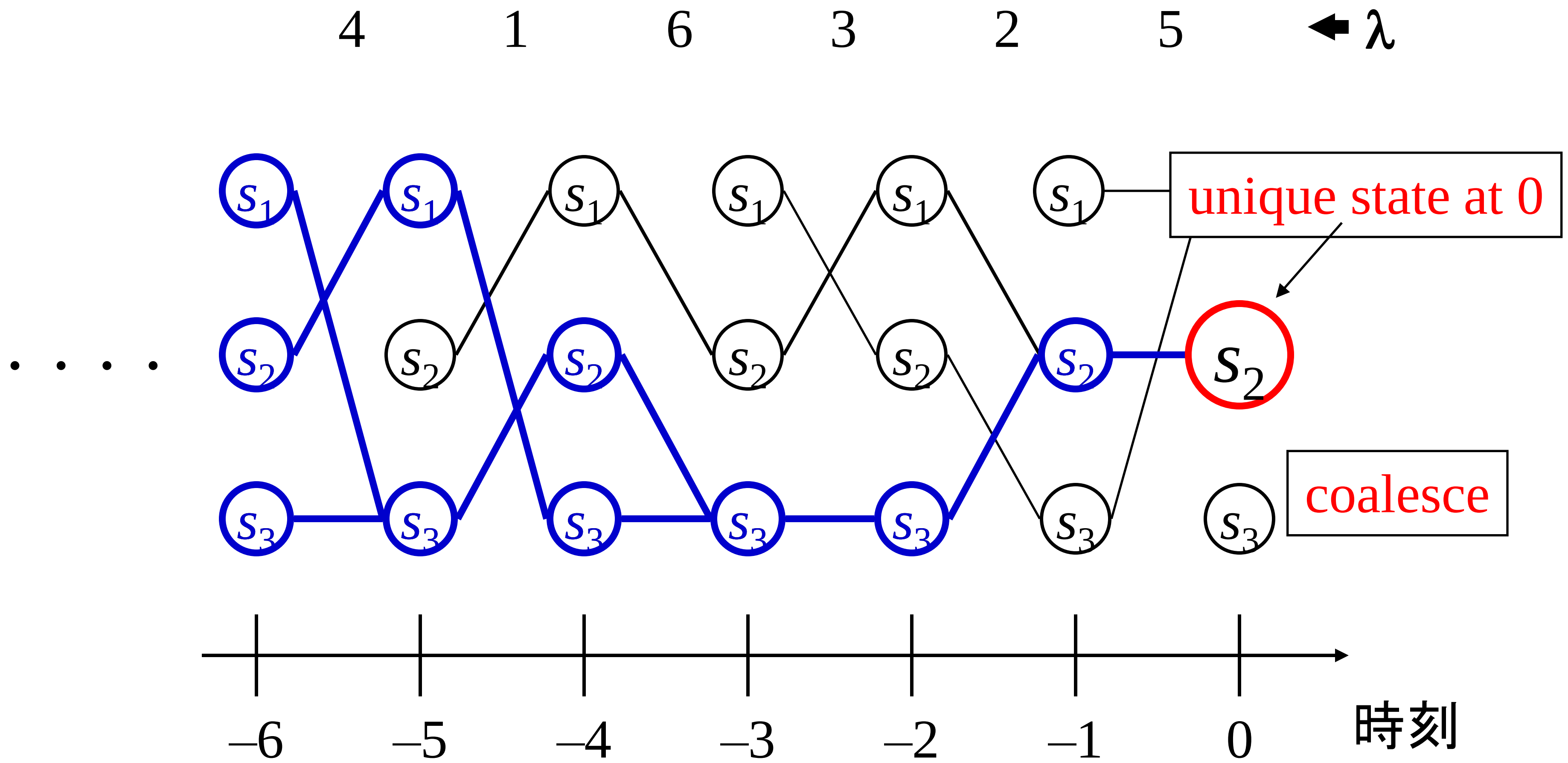


Coffee break （○×クイズ）

1. CFTPアルゴリズムにおいて、coalescence しなかったとき、$T := T - k$ (k : 自然数) としても良い。
   - ○ or ×
2. update function の決め方によっては coalesce しない。
   - ○ or ×
3. Coupling To The Future (CTTF)アルゴリズムは、全ての状態について時刻0から推移を開始し、coalesceしたらその状態を返す。CTTFも定常分布を実現する。
   - ○ or ×

CFTPのアイデア

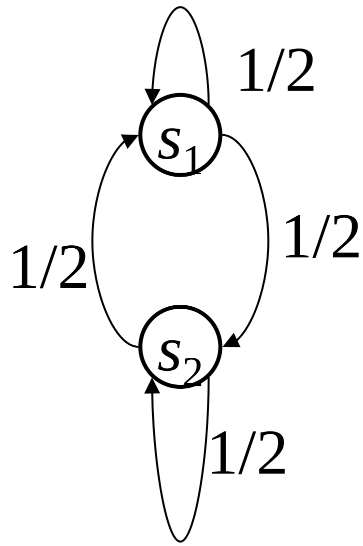
シミュレーションの開始時刻は、時刻0の状態に影響を及ぼさない



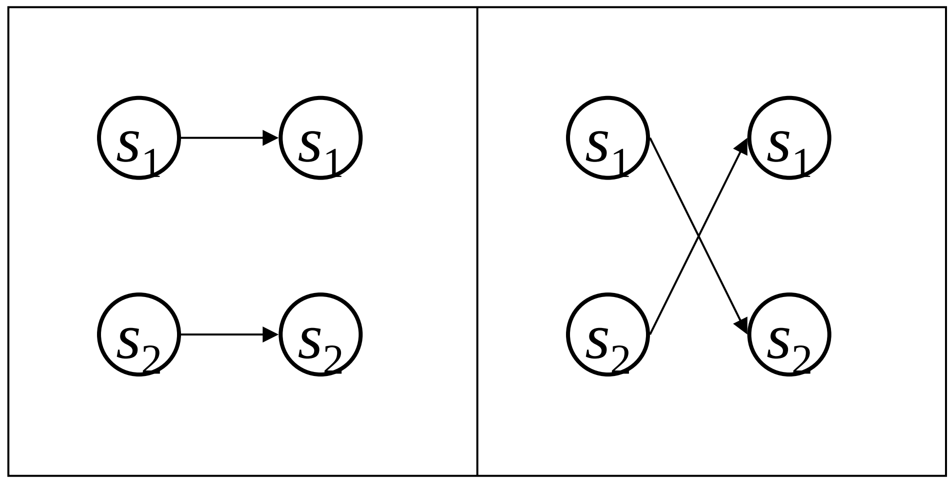
Coffee break （○×クイズ）

1. CFTPアルゴリズムにおいて、coalescence しなかったとき、$T := T - k$ (k : 自然数) としても良い。
   - ○
2. update function の決め方によっては coalesce しない。
   - ○ or ×
3. Coupling To The Future (CTTF)アルゴリズムは、全ての状態について時刻0から推移を開始し、coalesceしたらその状態を返す。CTTFも定常分布を実現する。
   - ○ or ×

マヌケな更新関数

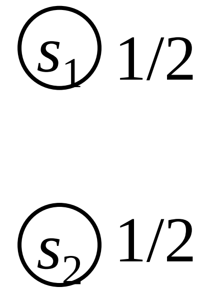


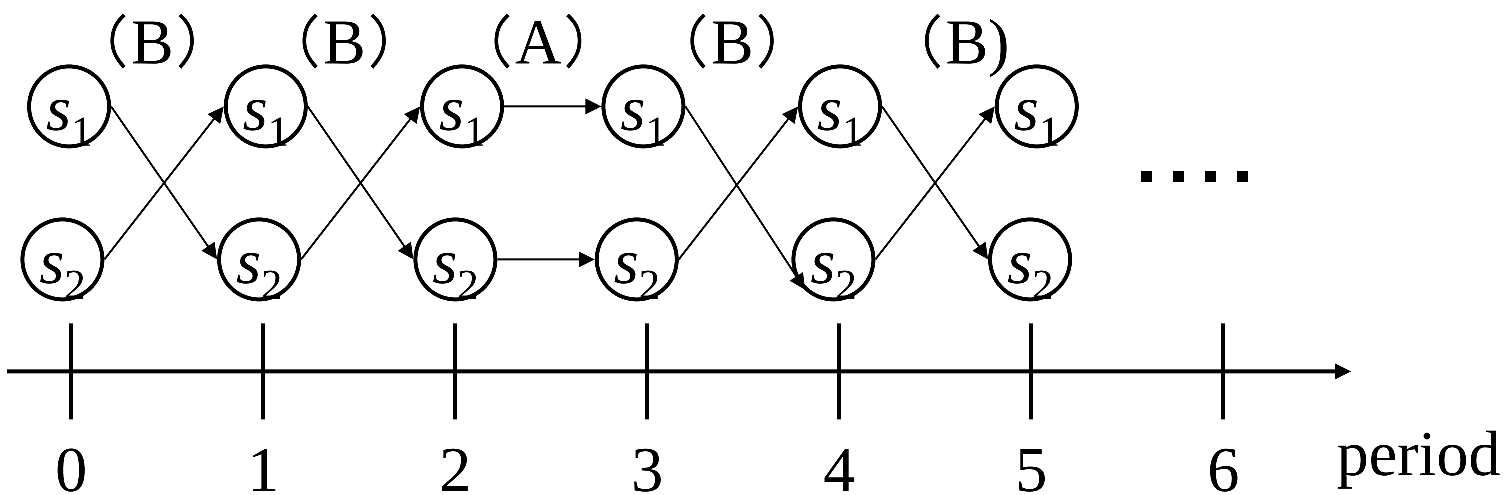
確率：（A) 1/2　　（B) 1/2



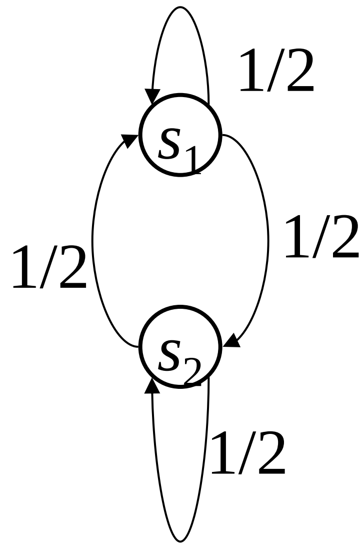
更新関数

定常分布

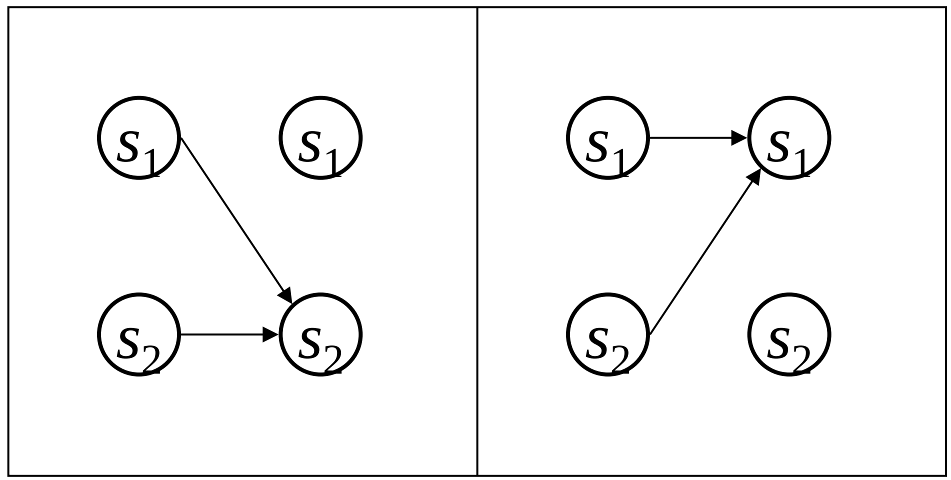




すごい更新関数

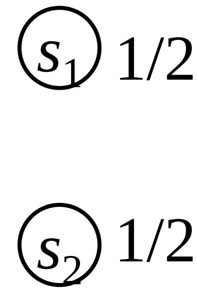


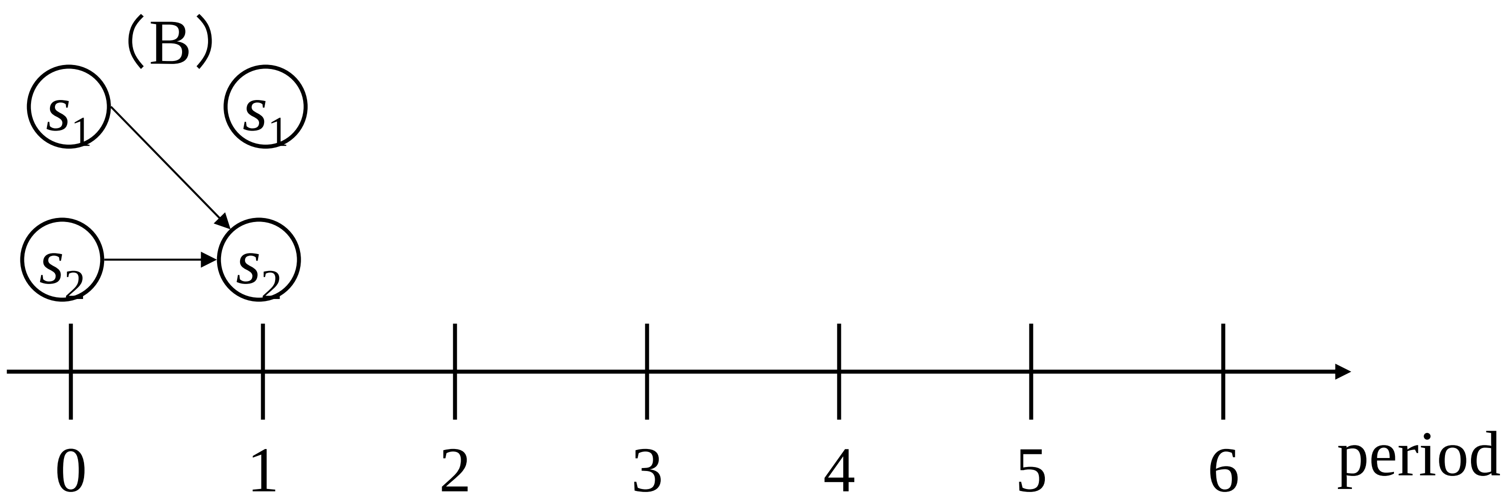
確率: (A) 1/2 (B) 1/2



更新関数

定常分布

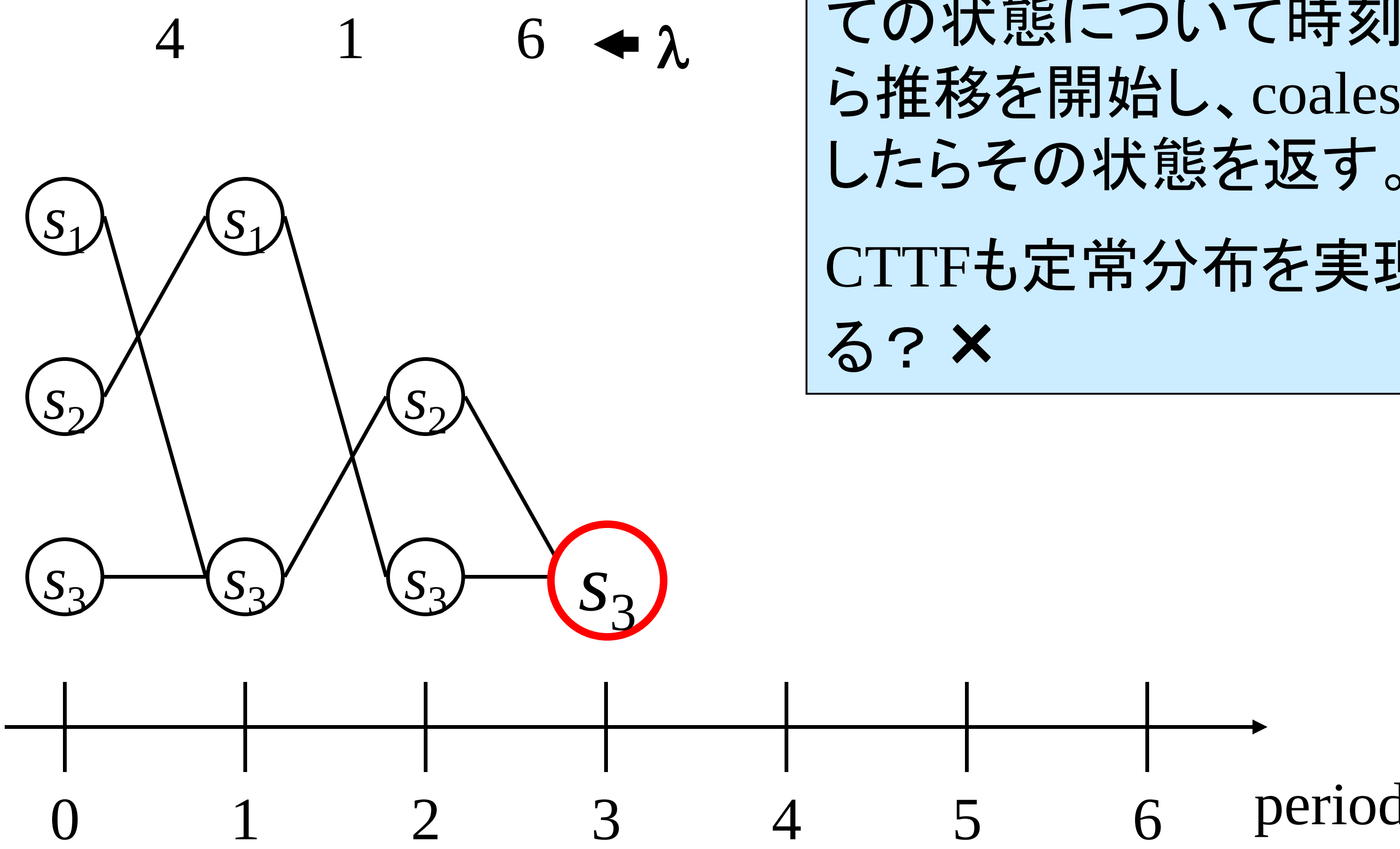




Coffee break （○×クイズ）

1. CFTPアルゴリズムにおいて、coalescence しなかったとき、$T := T - k$ (k : 自然数) としても良い。
   - ○
2. update function の決め方によっては coalesce しない。
   - ○
3. Coupling To The Future (CTTF)アルゴリズムは、全ての状態について時刻0から推移を開始し、coalesceしたらその状態を返す。CTTFも定常分布を実現する。
   - ○ or ×

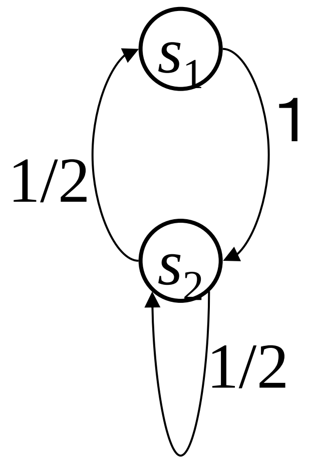
CTTFアルゴリズムのイメージ



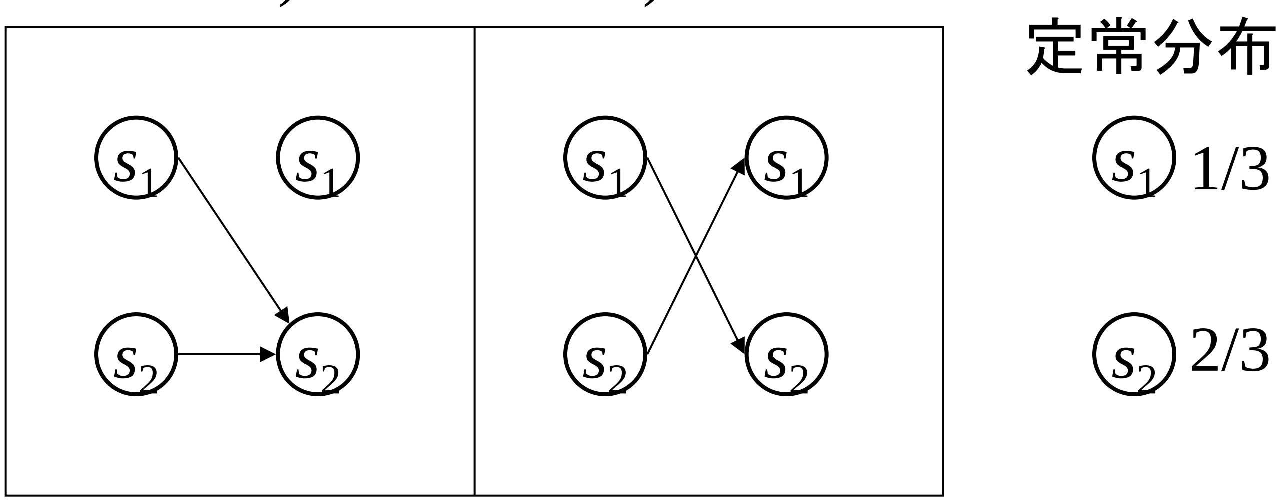
Coupling To The Future (CTTF)アルゴリズムは、全ての状態について時刻0から推移を開始し、coalesceしたらその状態を返す。

CTTFも定常分布を実現する？ ✕

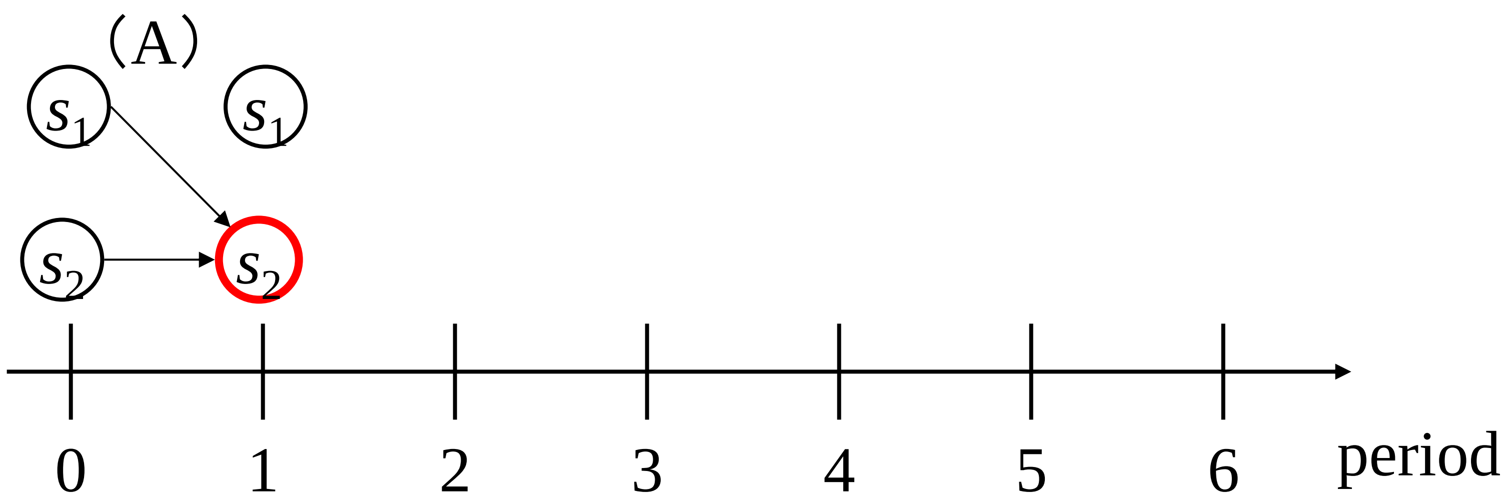
CTTP アルゴリズム



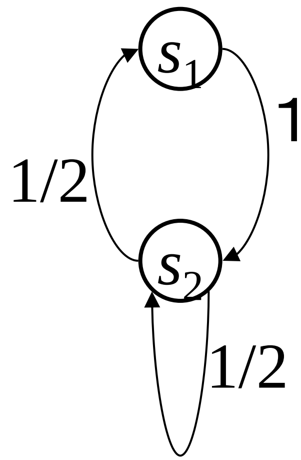
確率: (A) 1/2 (B) 1/2



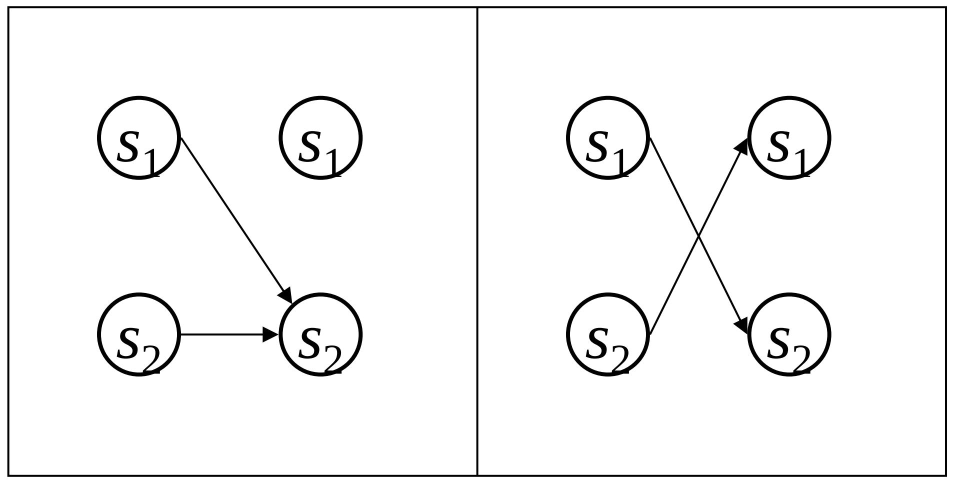
更新関数



CTTP アルゴリズム

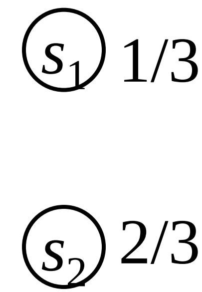


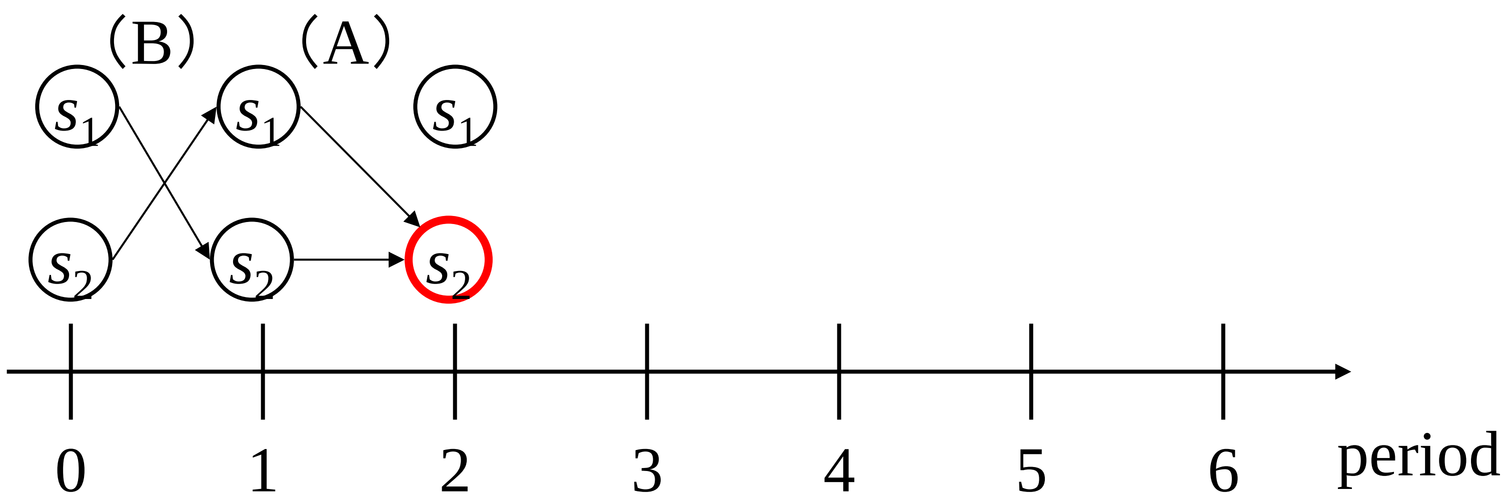
確率:（A) 1/2　　（B) 1/2



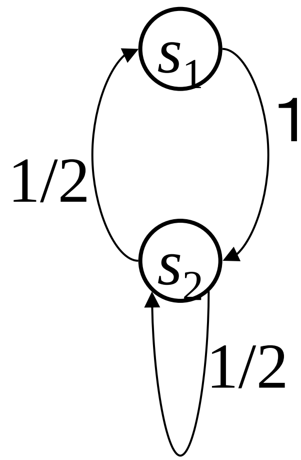
更新関数

定常分布

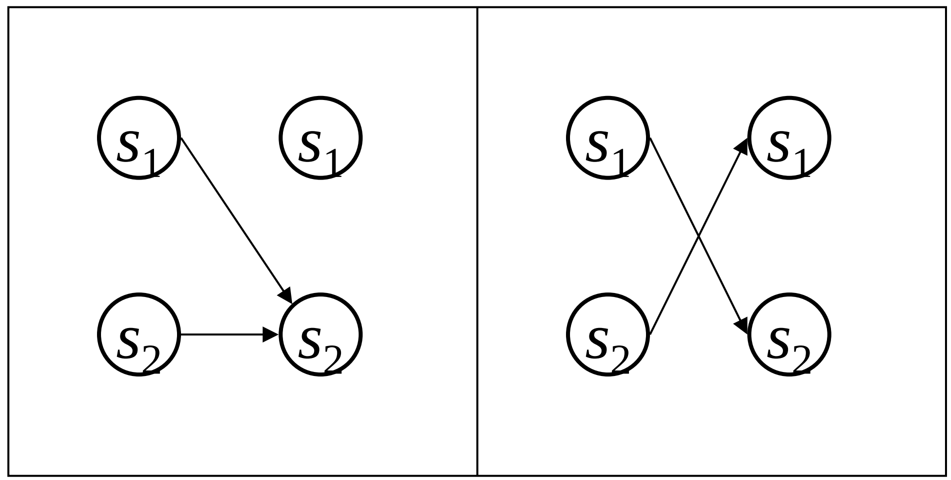




CTTP アルゴリズム

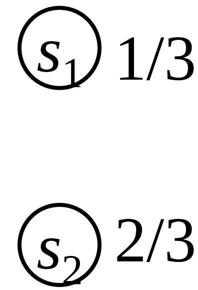


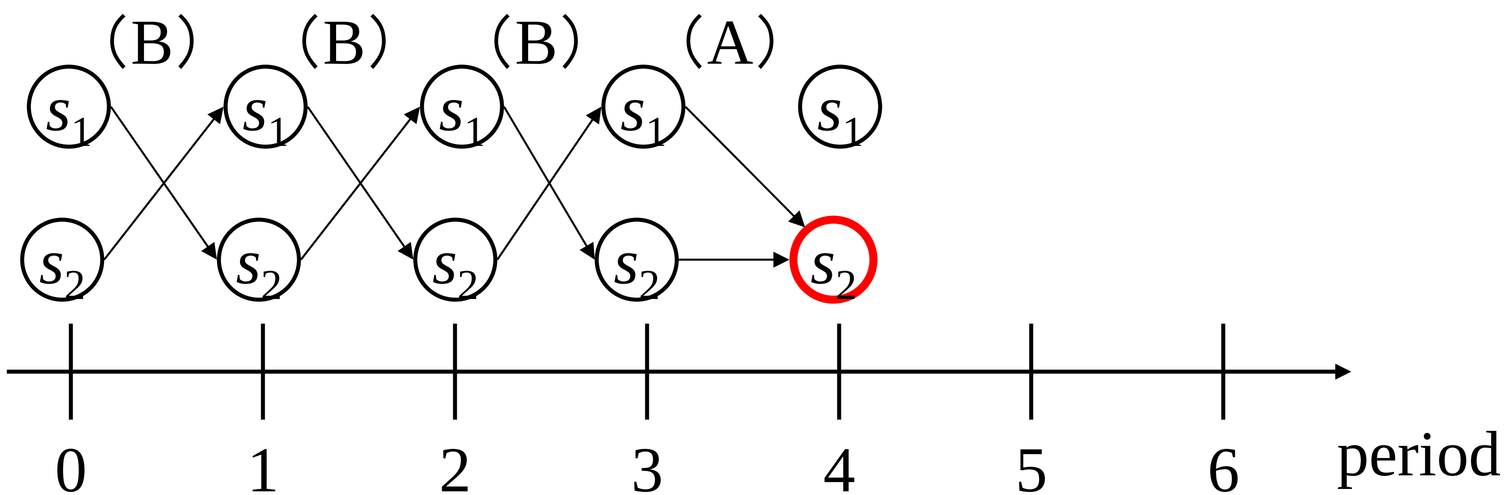
確率：（A） 1/2 （B） 1/2



更新関数

定常分布





Coffee break （○×クイズ）

1. CFTPアルゴリズムにおいて、coalescence しなかったとき、$T := T - k$ (k : 自然数) としても良い。
   - ○
2. update function の決め方によっては coalesce しない。
   - ○
3. Coupling To The Future (CTTF)アルゴリズムは、全ての状態について時刻0から推移を開始し、coalesceしたらその状態を返す。CTTFも定常分布を実現する。
   - ×

-2: 2行分割表のPerfect Sampling

S. Kijima and T. Matsui, Polynomial time perfect sampling algorithm for two-rowed contingency tables, Random Structures & Algorithms, 29:2 (2006), 243--256.

例: 2行分割表のランダム生成

2元分割表

- ✓ 各セルには非負整数が入る
- ✓ （与えられた）周辺和を満たす

						12
						18
5	4	3	7	5	6	30

問題

Given: 周辺和

出力: 分割表の一様ランダム生成

例: 2行分割表のランダム生成

2元分割表

- ✓ 各セルには非負整数が入る
- ✓ （与えられた）周辺和を満たす

						12
						18
5	4	3	7	5	6	30

5	4	3	0	0	0	12
0	0	0	7	5	6	18
5	4	3	7	5	6	30

状態A

4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30

状態B

0	0	0	1	5	6	12
5	4	3	6	0	0	18
5	4	3	7	5	6	30

状態C

問題

Given: 周辺和

出力: 分割表の一様ランダム生成

例: 2行分割表のランダム生成

2元分割表

- ✓ 各セルには非負整数が入る
- ✓（与えられた）周辺和を満たす

						12
						18
5	4	3	7	5	6	30

与えられた周辺和を満たす2行分割表の**個数を求める問題**

⇒#P完全 (NP困難) ['97 Dyer, Kannan, & Mount]

問題

Given: 周辺和

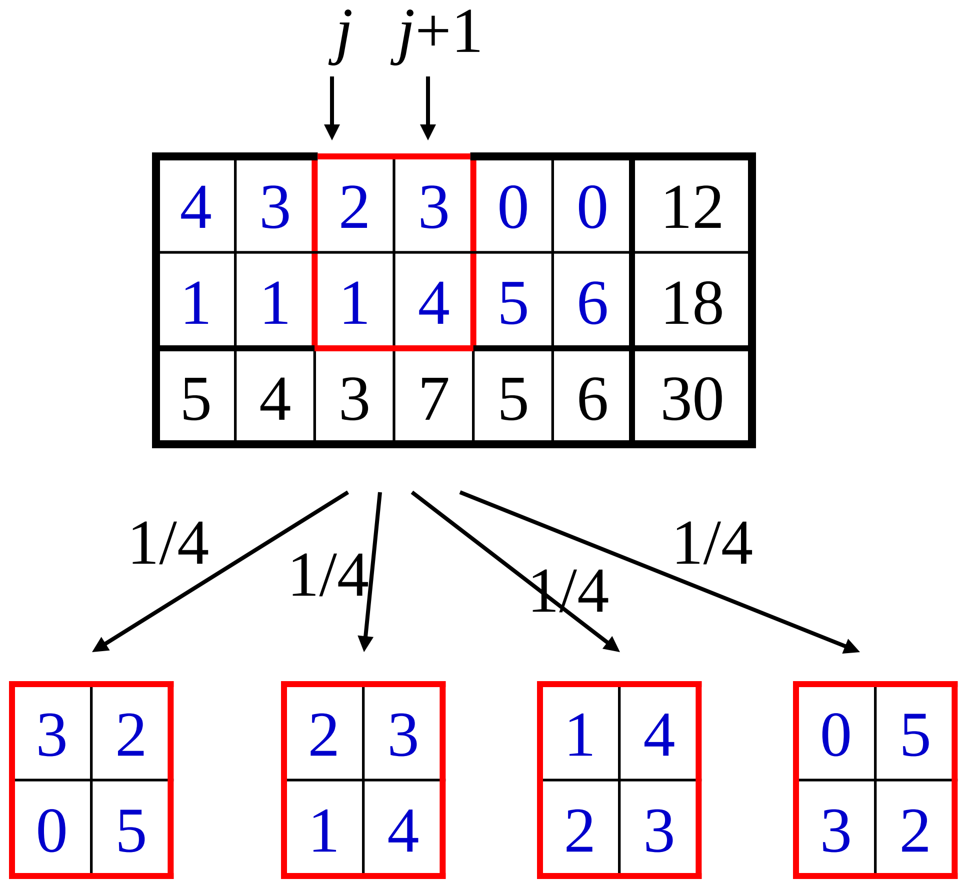
出力: 分割表の一様ランダム生成

マルコフ連鎖を用いたサンプリング法

再掲: マルコフ連鎖

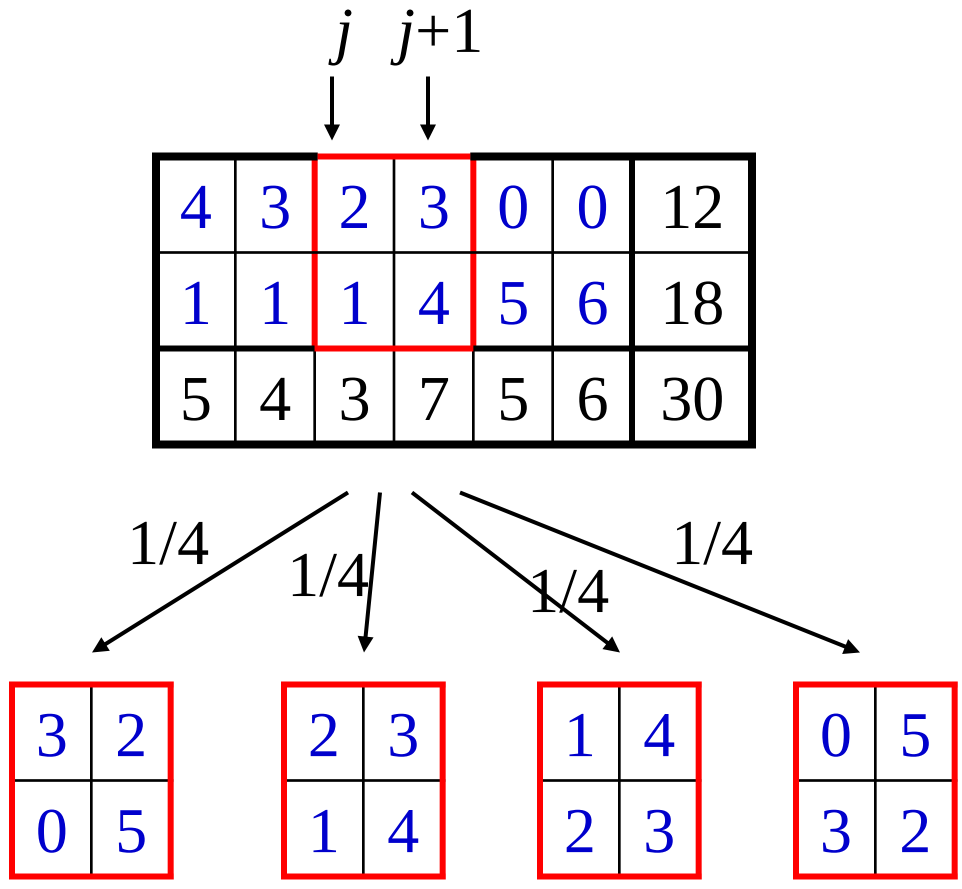
- j 列目（とj+1列目）を 1/(n –1)の確率で選ぶ。
- j 列目と j+1列目に対して 推移可能な状態に 等確率で推移する。

$$\begin{array}{|cc|c|} 2 & 3 & 5 \\ 1 & 4 & 5 \\ \hline 3 & 7 & 10 \end{array} + \begin{array}{|cc|} +k & -k \\ -k & +k \end{array}$$



マルコフ連鎖（更新関数）

- 実数乱数 $\lambda \in [1, n)$ を生成。
- $j = \lfloor \lambda \rfloor$ とする。
- $X(1, j) = max. - \lfloor \theta \lambda' \rfloor$
  - ➢ θ: 推移可能な状態数
  - ➢ $\lambda' := \lambda - \lfloor \lambda \rfloor$
  - ➢ $max.$: $X[1, j]$ の最大値



サンプリングアルゴリズム (単調CFTPアルゴリズム)

1. set $T = -1$; set $\boldsymbol{\lambda}$: 空列;
2. generate $\lambda[T] \in [1, n)$:一様実数乱数;

   put $\boldsymbol{\lambda} := (\lambda[T], \lambda[T+1],\ldots,\lambda[-1])$;
3. X_U, X_L について、時刻 T から 0 まで $\boldsymbol{\lambda}$ を用いて推移。
   a. if coalesce on $Y \Rightarrow$ return Y;
   b. otherwise, set $T := T - 1$; 2.に戻る;

5	4	3	0	0	0	12
0	0	0	7	5	6	18
5	4	3	7	5	6	30

X_U : N-W表

0	0	0	1	5	6	12
5	4	3	6	0	0	18
5	4	3	7	5	6	30

X_L : N-E表

定理: 提案したアルゴリズムにより

$2 \times n$分割表のPerfect Sampling が行える。

定理の証明のポイント

主張

X_U, X_Lからのcoalescence $\Leftrightarrow$ 全状態からのcoalescence

- 状態空間上に半順序関係を導入する。
- X_U, X_Lはそれぞれ半順序関係の最大元、最小元である。
- 半順序関係は推移の後も保たれる。

単調マルコフ連鎖

定義：関数 $f_X(i)$

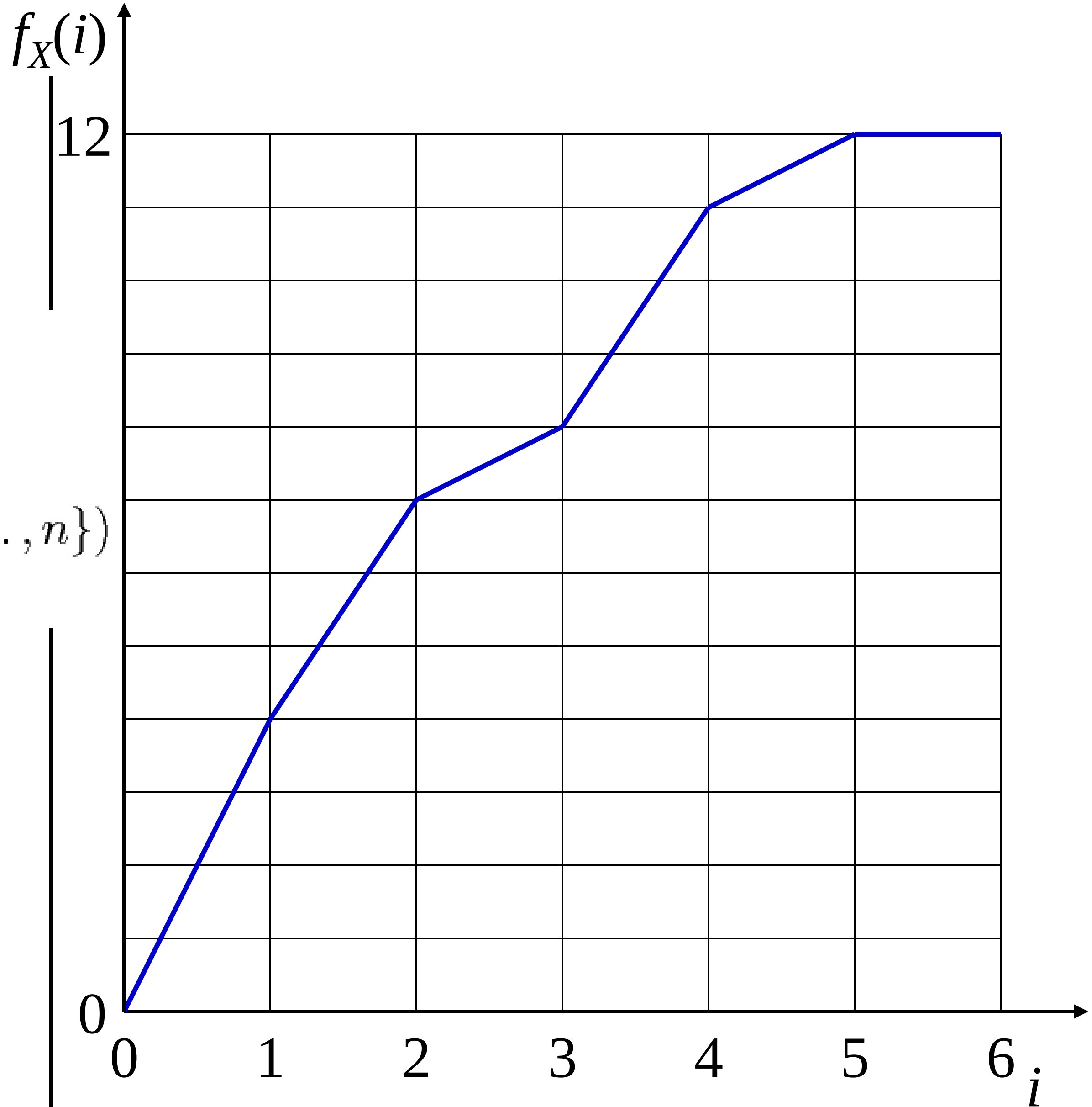
$X : 2 \times n$ 分割表

$f_X : \{0, 1, \ldots, n\} \to \mathrm{Z}_+$

$$f_X(i) \overset{\text{def.}}{=} \begin{cases} 0 & (i = 0) \\ \displaystyle\sum_{j=1}^{i} X[1, j] & (i \in \{1, \ldots, n\}) \end{cases}$$

- 一対一対応

4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30



定義：半順序

$$X \geq Y$$

$$\Leftrightarrow \; f_X(i) \geq f_Y(i)$$

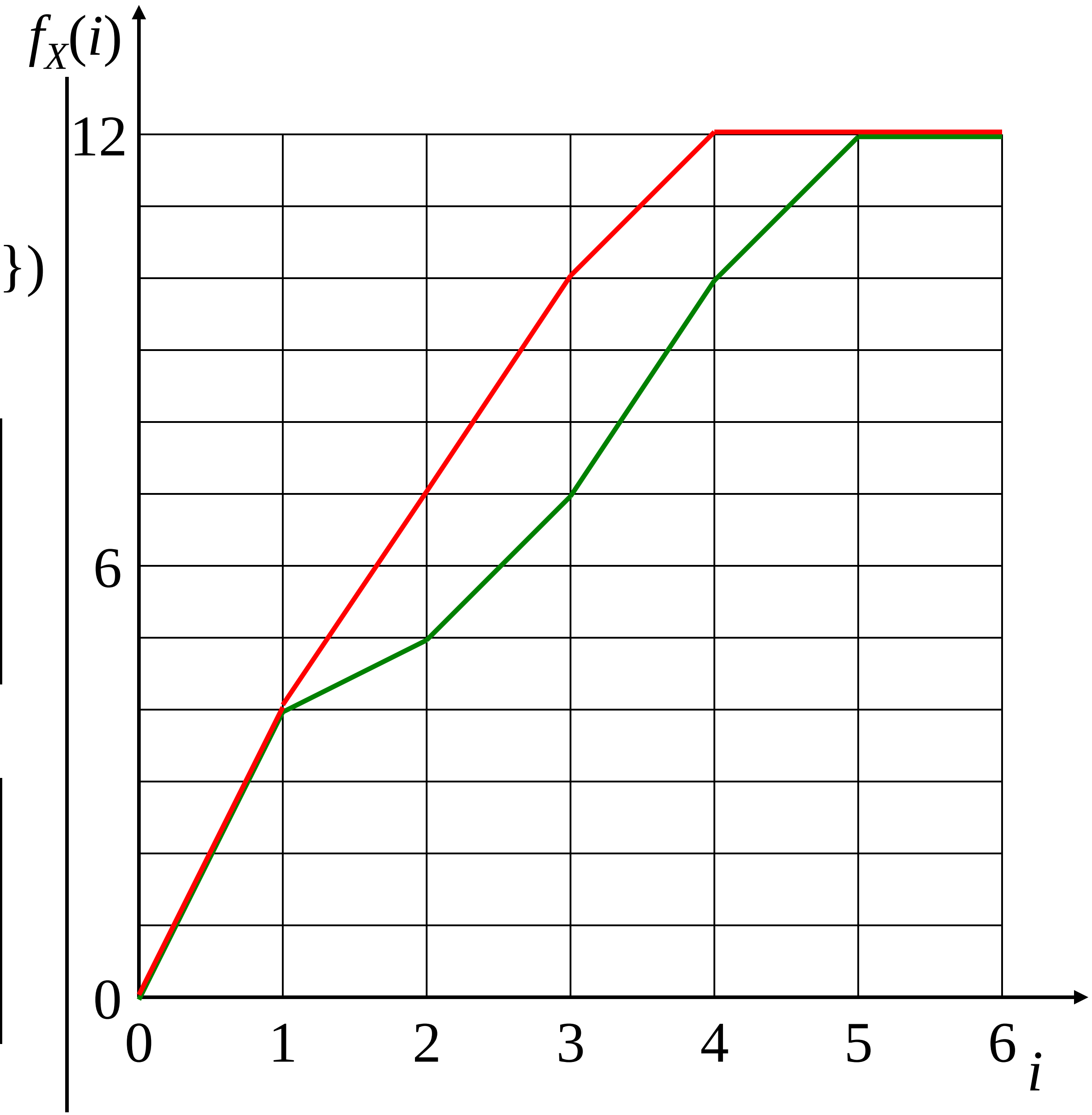
$$(\; \forall\, i \in \{0, \ldots, n\})$$

X

4	3	3	2	0	0	12
1	1	0	5	5	6	18
5	4	3	7	5	6	30

Y

4	1	2	3	2	0	12
1	3	1	4	3	6	18
5	4	3	7	5	6	30



最大元、最小元

5	4	3	0	0	0	12
0	0	0	7	5	6	18
5	4	3	7	5	6	30

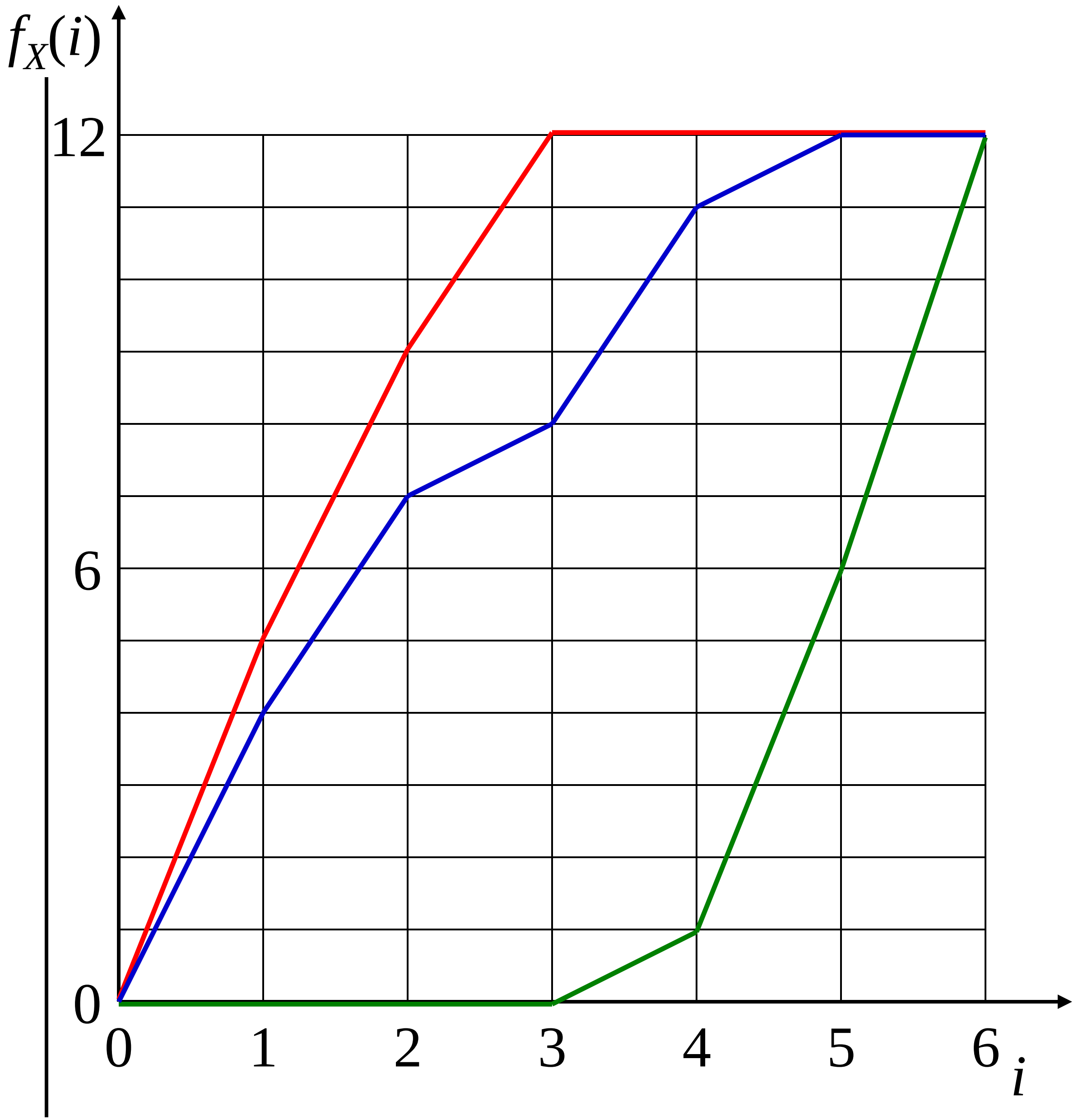
X_{U} : N-W表

0	0	0	1	5	6	12
5	4	3	6	0	0	18
5	4	3	7	5	6	30

X_{L} : N-E表

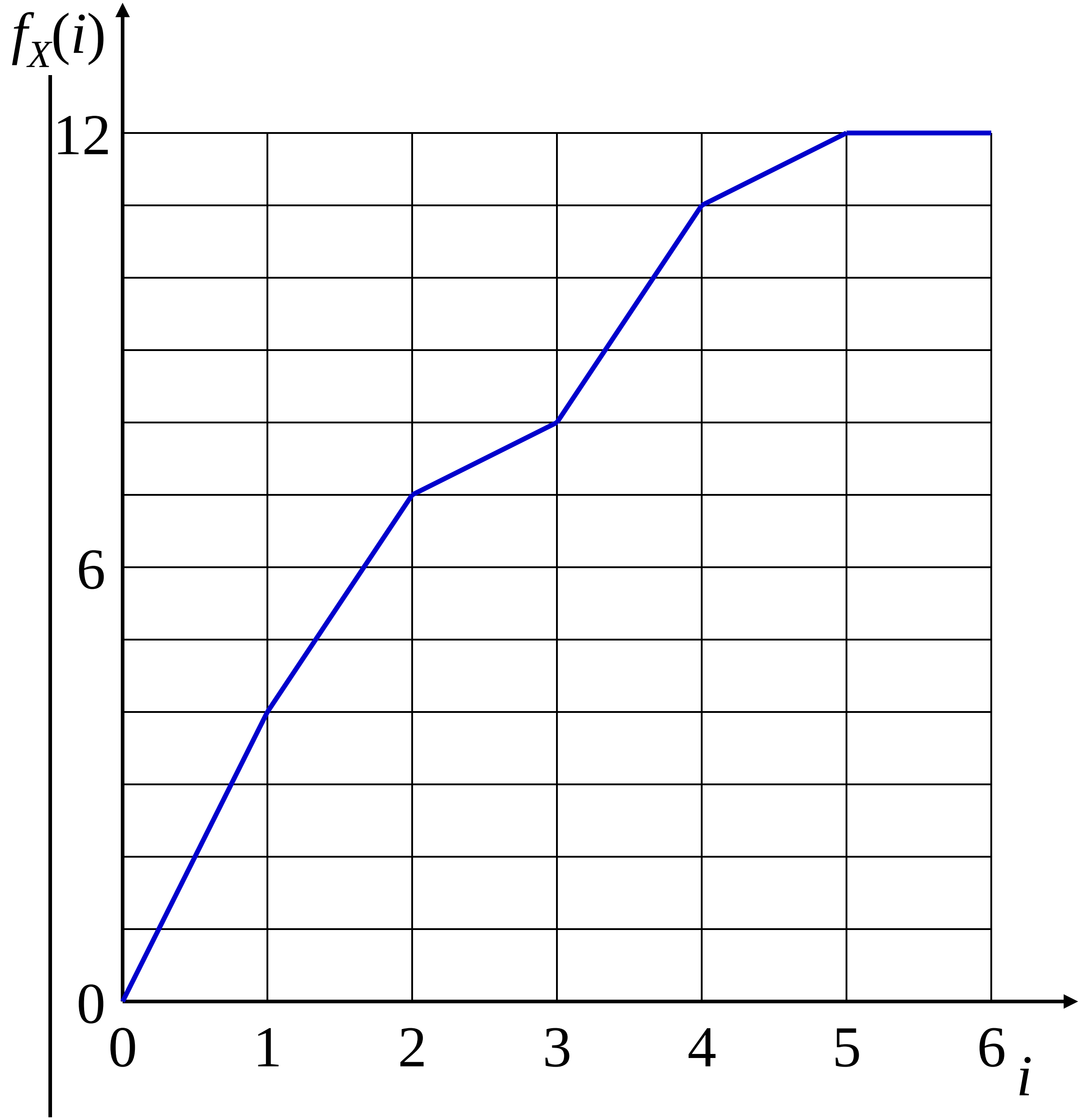
補題

$$X_{\mathrm{U}} \geq \forall X \geq X_{\mathrm{L}}$$



推移

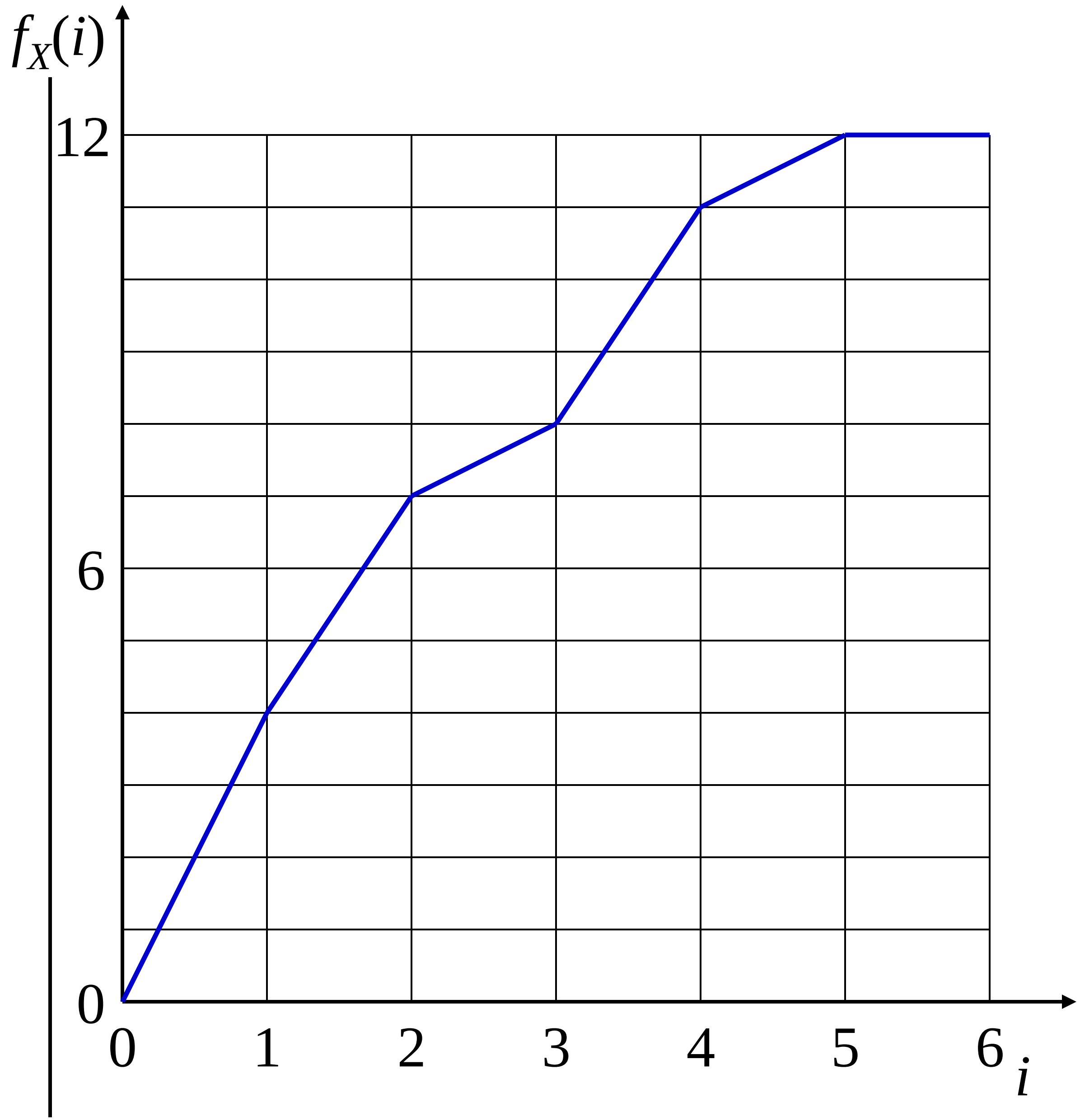
4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30



推移

4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30

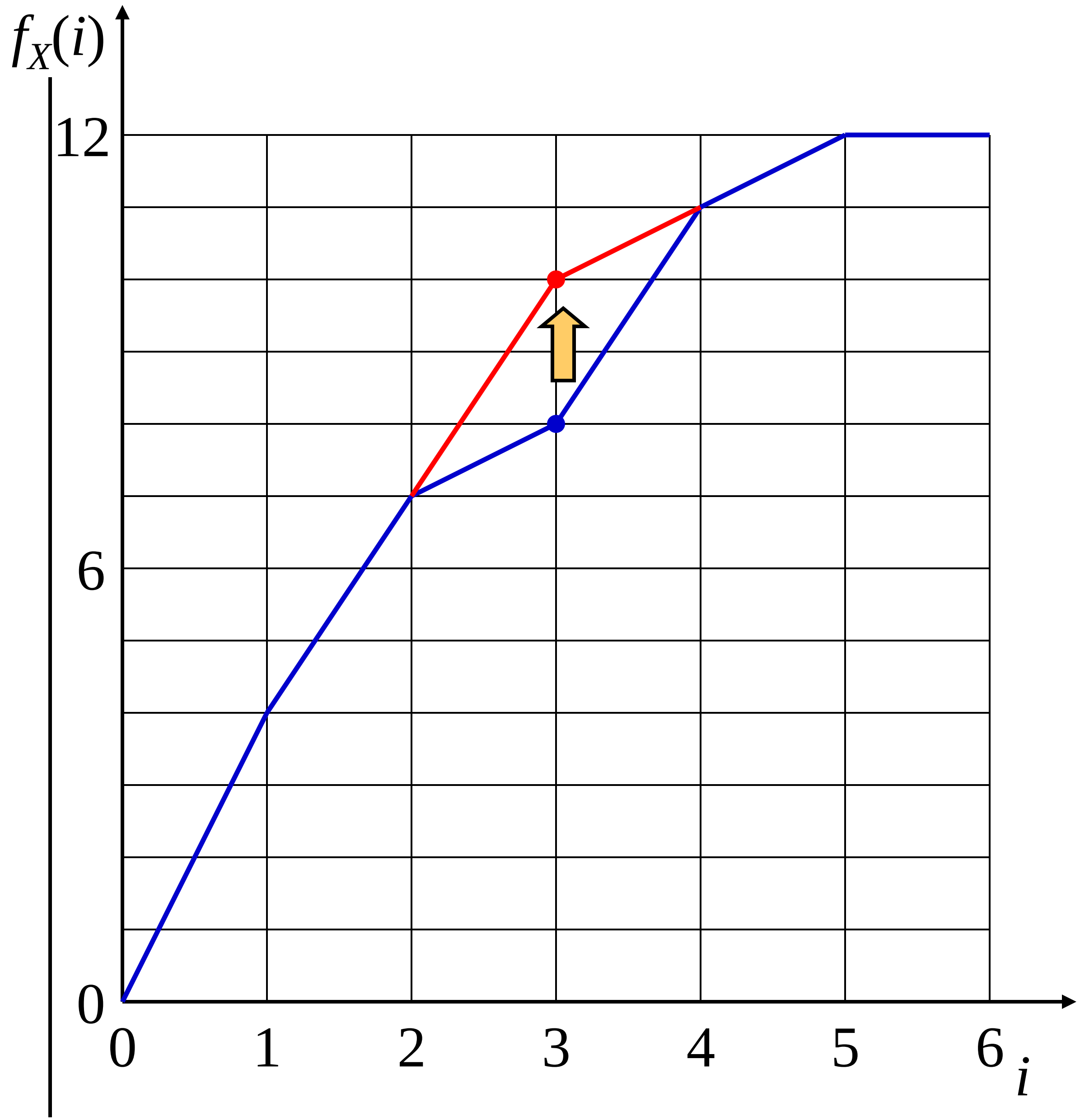
4	3	3	1	1	0	12
1	1	0	6	4	6	18
5	4	3	7	5	6	30



推移

4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30

4	3	3	1	1	0	12
1	1	0	6	4	6	18
5	4	3	7	5	6	30



定義：隣接性 (X -> Y)

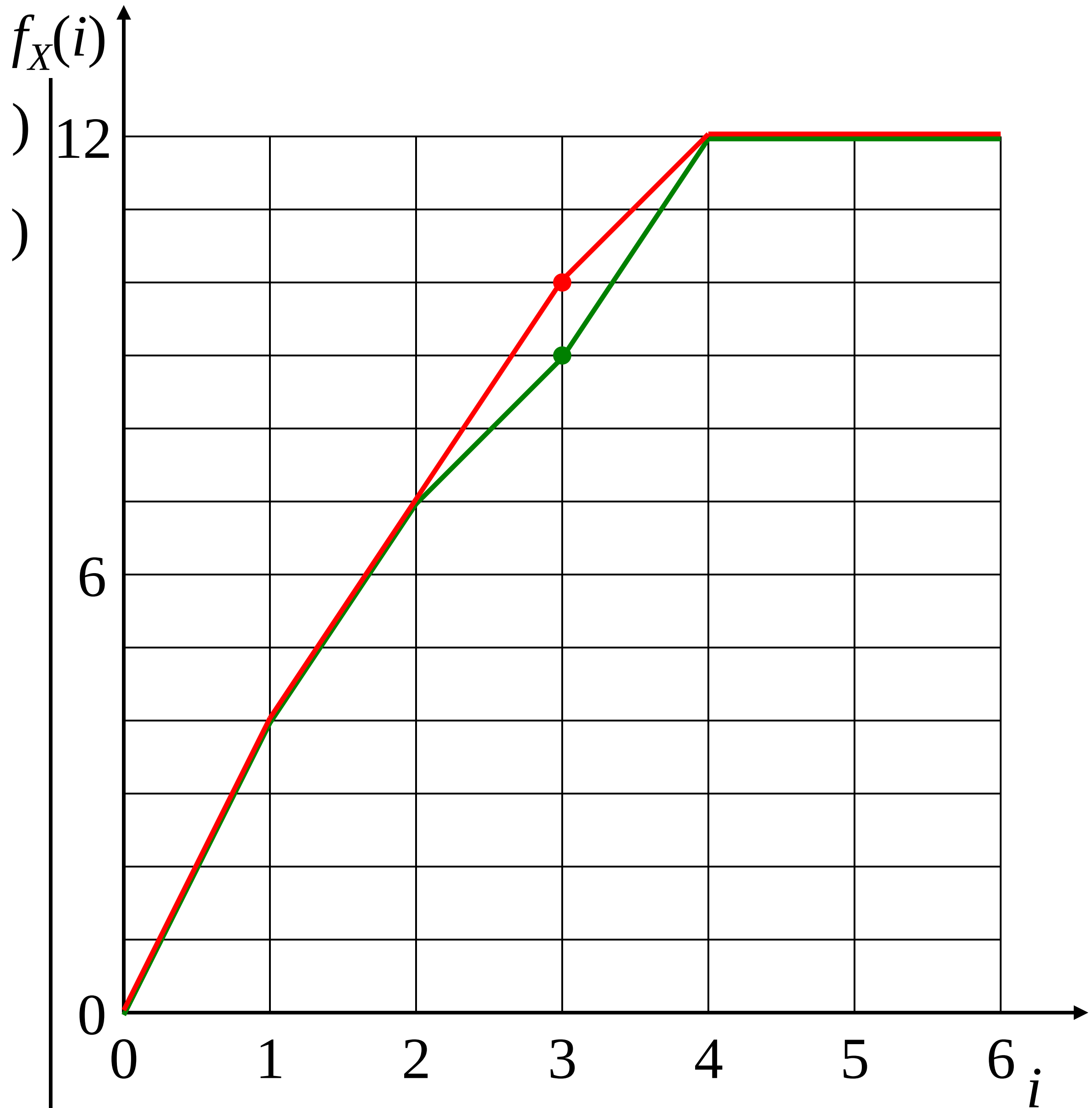
$$f_X(i) - f_Y(i) = \begin{cases} 1 \ (\ i = k\ ) \\ 0 \ (\ i \neq k\ ) \end{cases}$$

X

4	3	3	2	0	0	12
1	1	0	5	5	6	18
5	4	3	7	5	6	30

Y

4	3	2	3	0	0	12
1	1	1	4	5	6	18
5	4	3	7	5	6	30



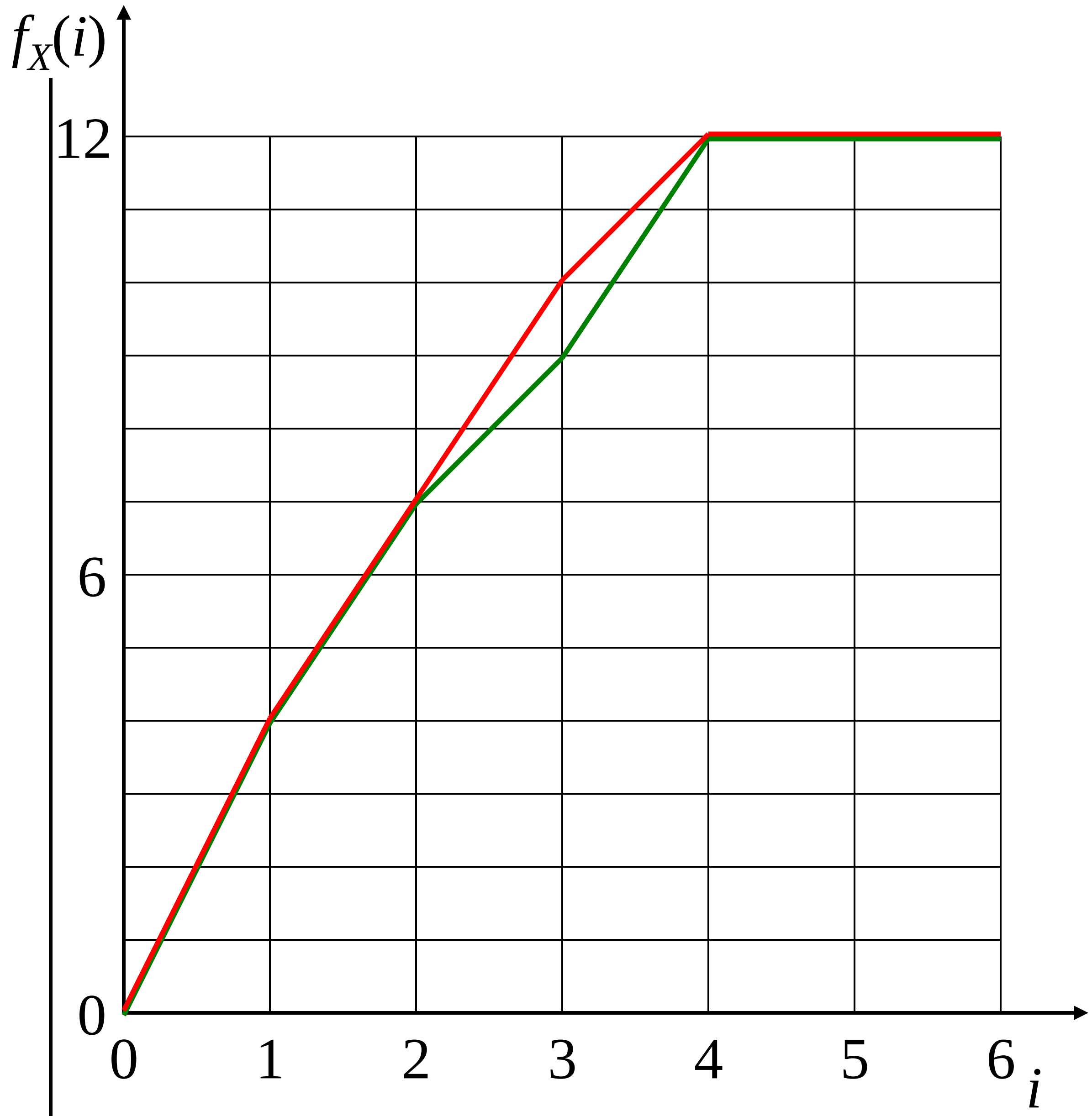
推移と半順序関係

X

4	3	3	2	0	0	12
1	1	0	5	5	6	18
5	4	3	7	5	6	30

Y

4	3	2	3	0	0	12
1	1	1	4	5	6	18
5	4	3	7	5	6	30



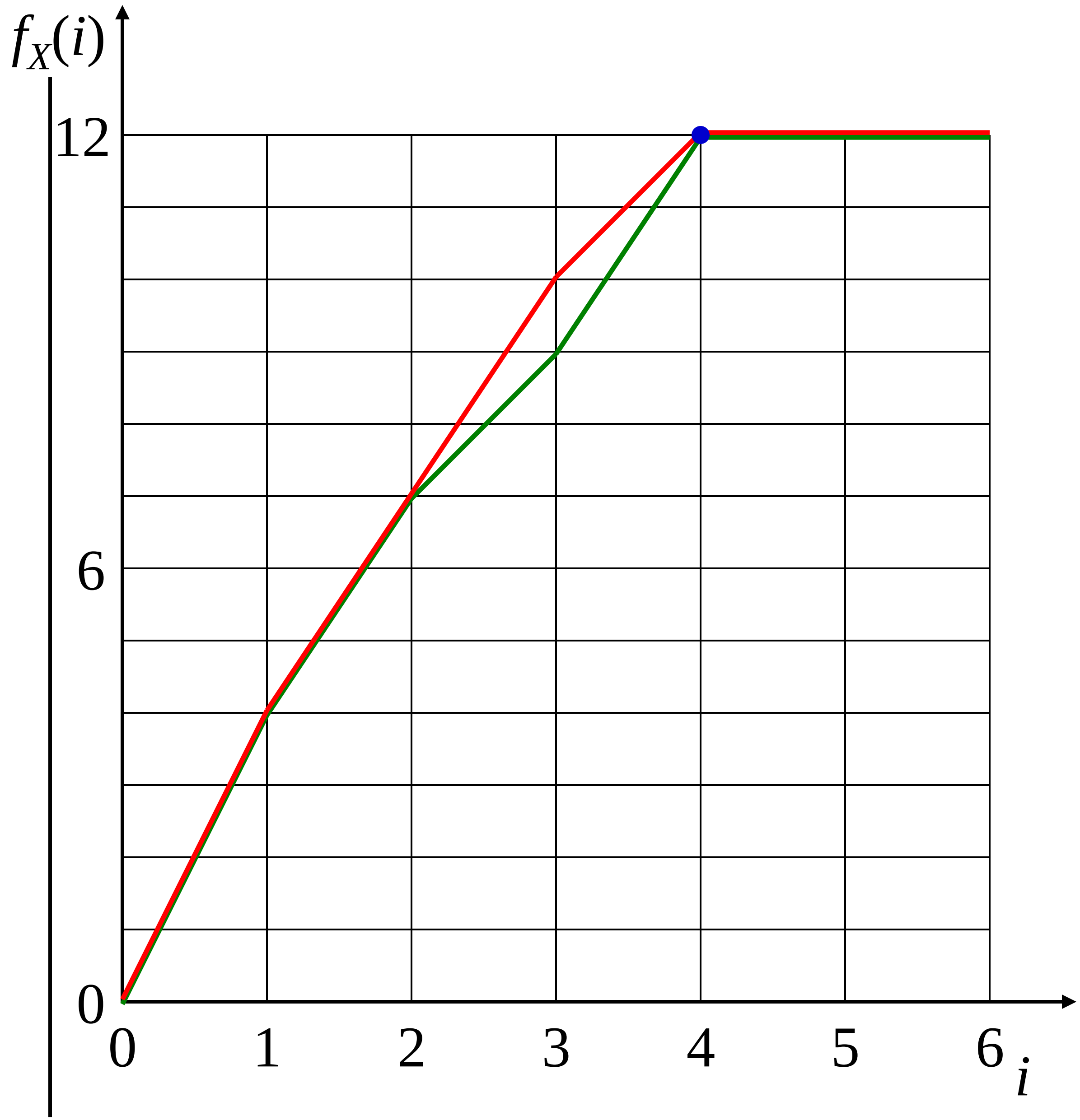
推移と半順序関係

X

4	3	3	2	0	0	12
1	1	0	5	5	6	18
5	4	3	7	5	6	30

Y

4	3	2	3	0	0	12
1	1	1	4	5	6	18
5	4	3	7	5	6	30



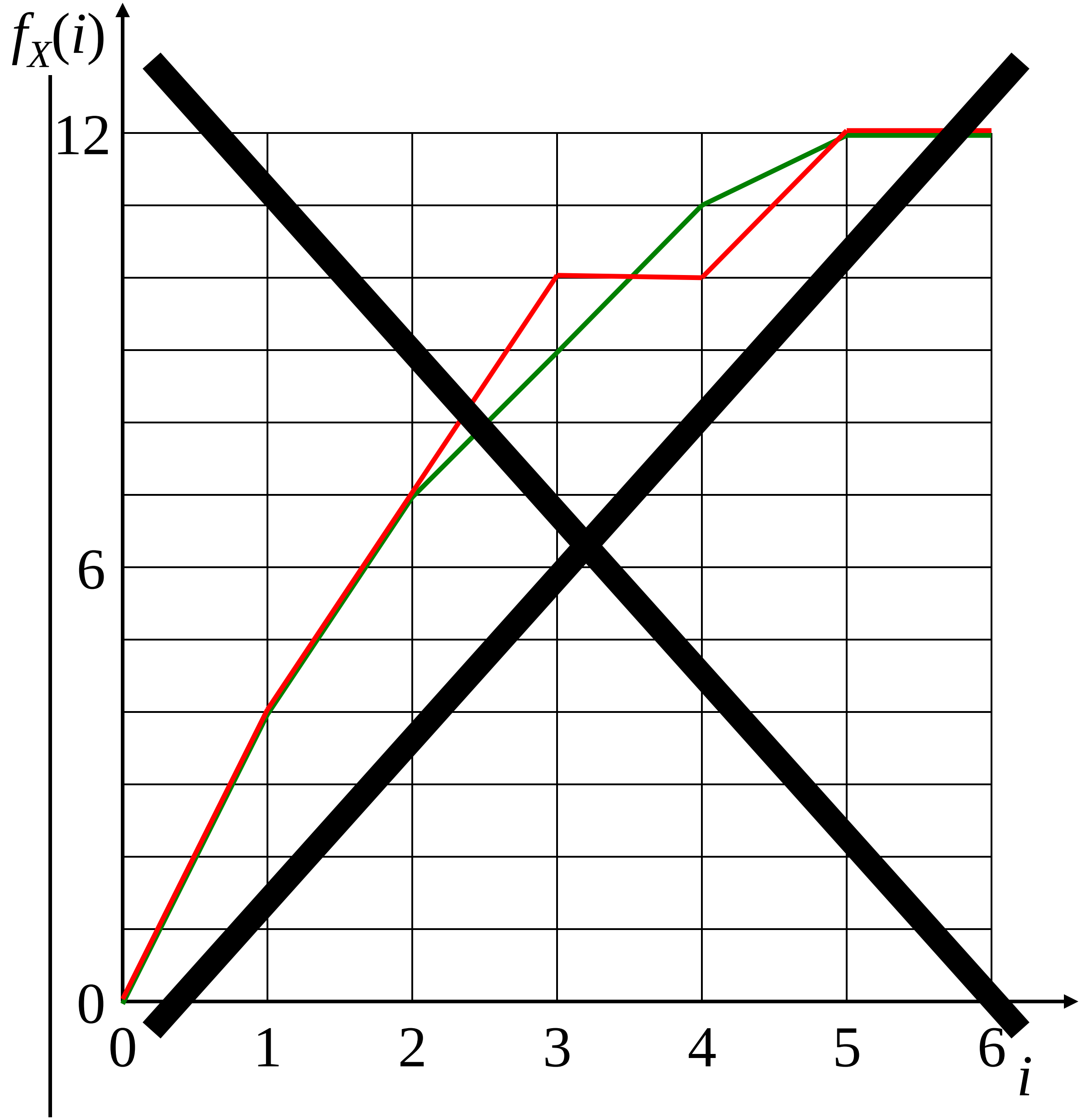
推移と半順序関係

X’

4	3	3	0	2	0	12
1	1	0	7	3	6	18
5	4	3	7	5	6	30

Y’

4	3	2	2	1	0	12
1	1	1	5	4	6	18
5	4	3	7	5	6	30



定理の証明のポイント

主張

$X_{\mathrm{U}}, X_{\mathrm{L}}$からのcoalescence $\Leftrightarrow$ 全状態からのcoalescence

- 状態空間上に半順序関係を導入する。
- $X_{\mathrm{U}}, X_{\mathrm{L}}$はそれぞれ半順序関係の最大元、最小元である。
- 半順序関係は推移の後も保たれる。

↓

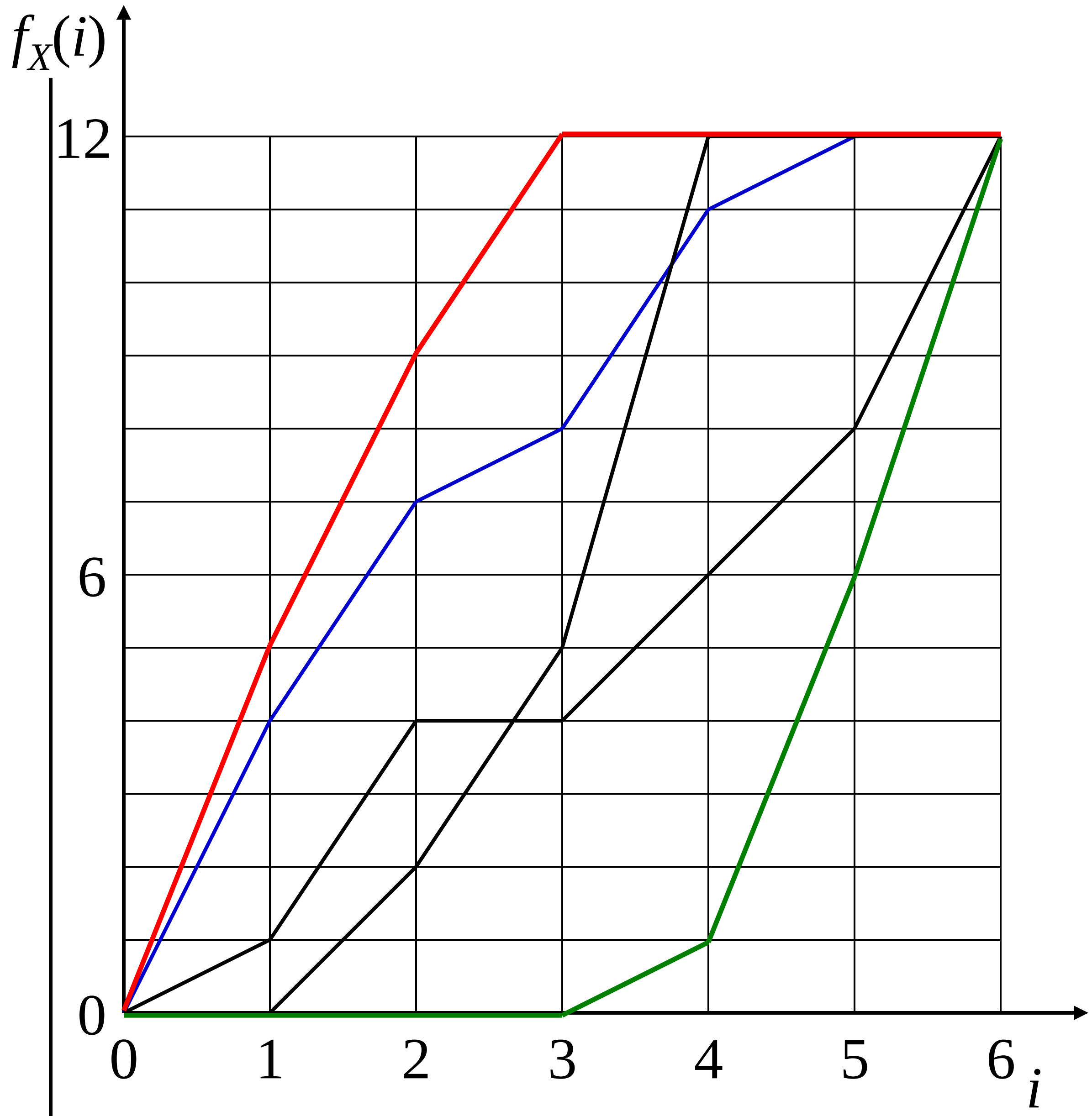
単調マルコフ連鎖

coalescence イメージ

5	4	3	0	0	0	12
0	0	0	7	5	6	18
5	4	3	7	5	6	30

4	3	1	3	1	0	12
1	1	2	4	4	6	18
5	4	3	7	5	6	30

0	0	0	1	5	6	12
5	4	3	6	0	0	18
5	4	3	7	5	6	30

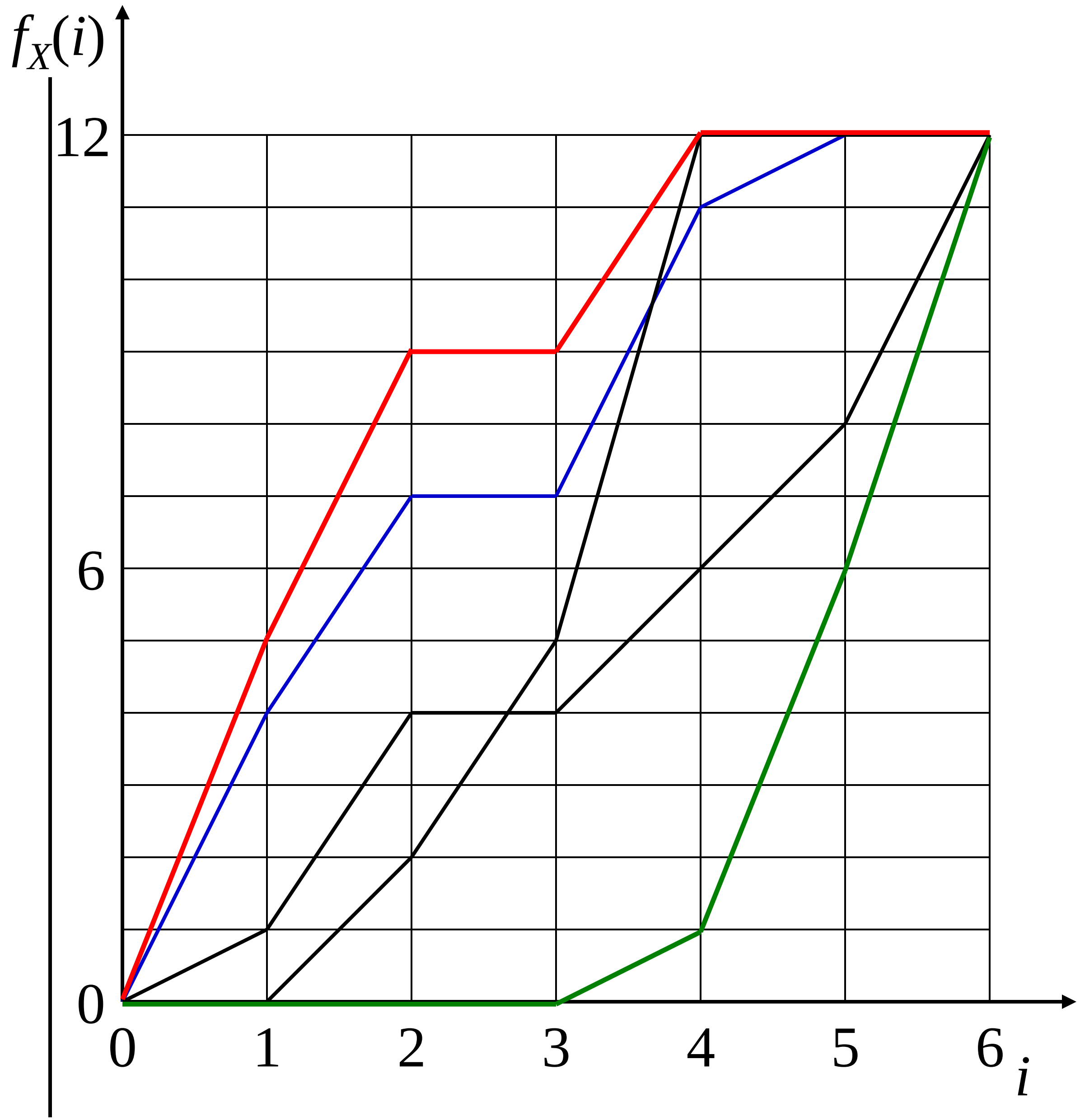


coalescence イメージ

5	4	0	3	0	0	12
0	0	3	4	5	6	18
5	4	3	7	5	6	30

4	3	0	4	1	0	12
1	1	3	3	4	6	18
5	4	3	7	5	6	30

0	0	0	1	5	6	12
5	4	3	6	0	0	18
5	4	3	7	5	6	30

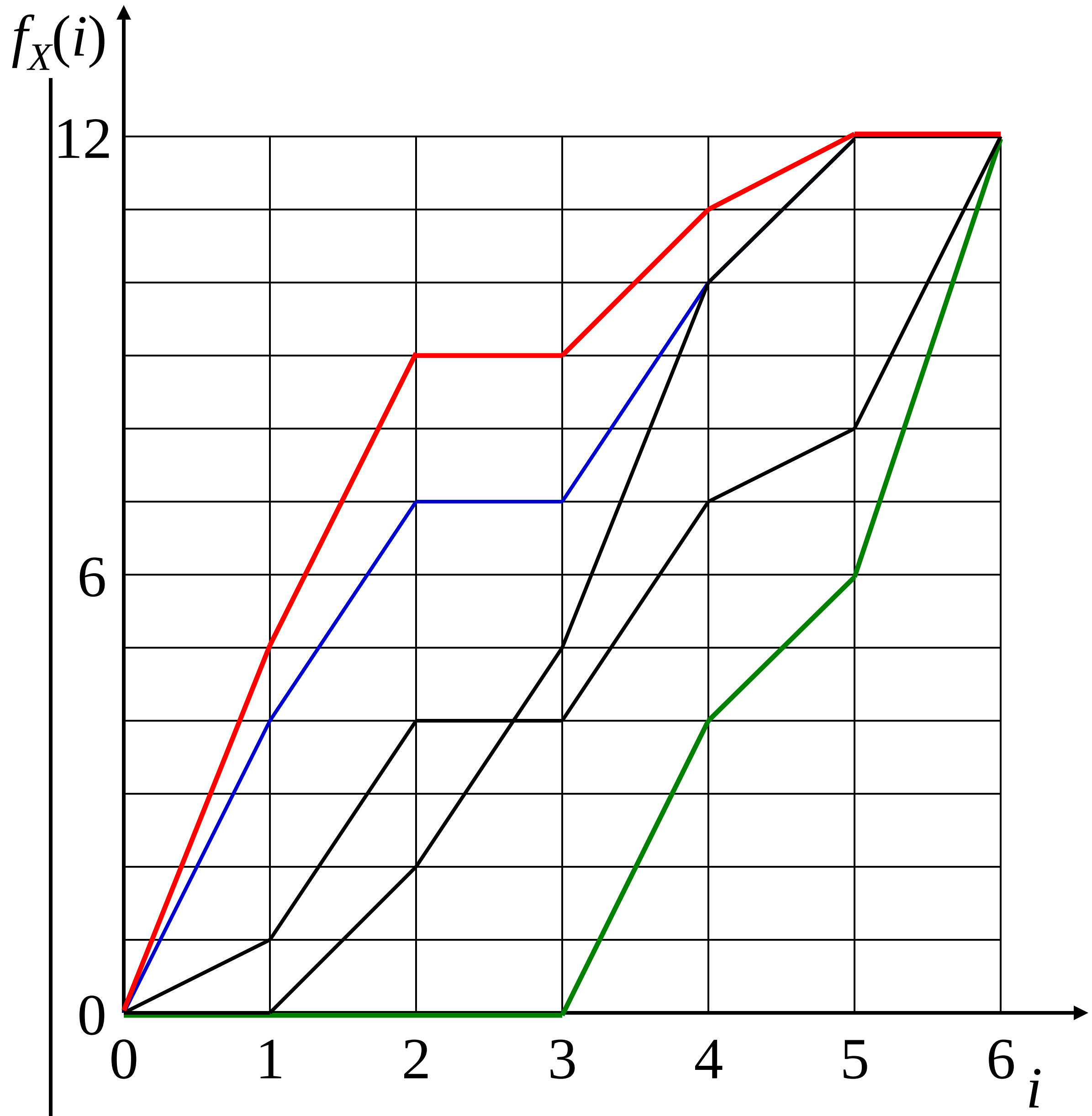


coalescence イメージ

5	4	0	2	1	0	12
0	0	3	5	4	6	18
5	4	3	7	5	6	30

4	3	0	3	2	0	12
1	1	3	4	3	6	18
5	4	3	7	5	6	30

0	0	0	4	2	6	12
5	4	3	3	3	0	18
5	4	3	7	5	6	30

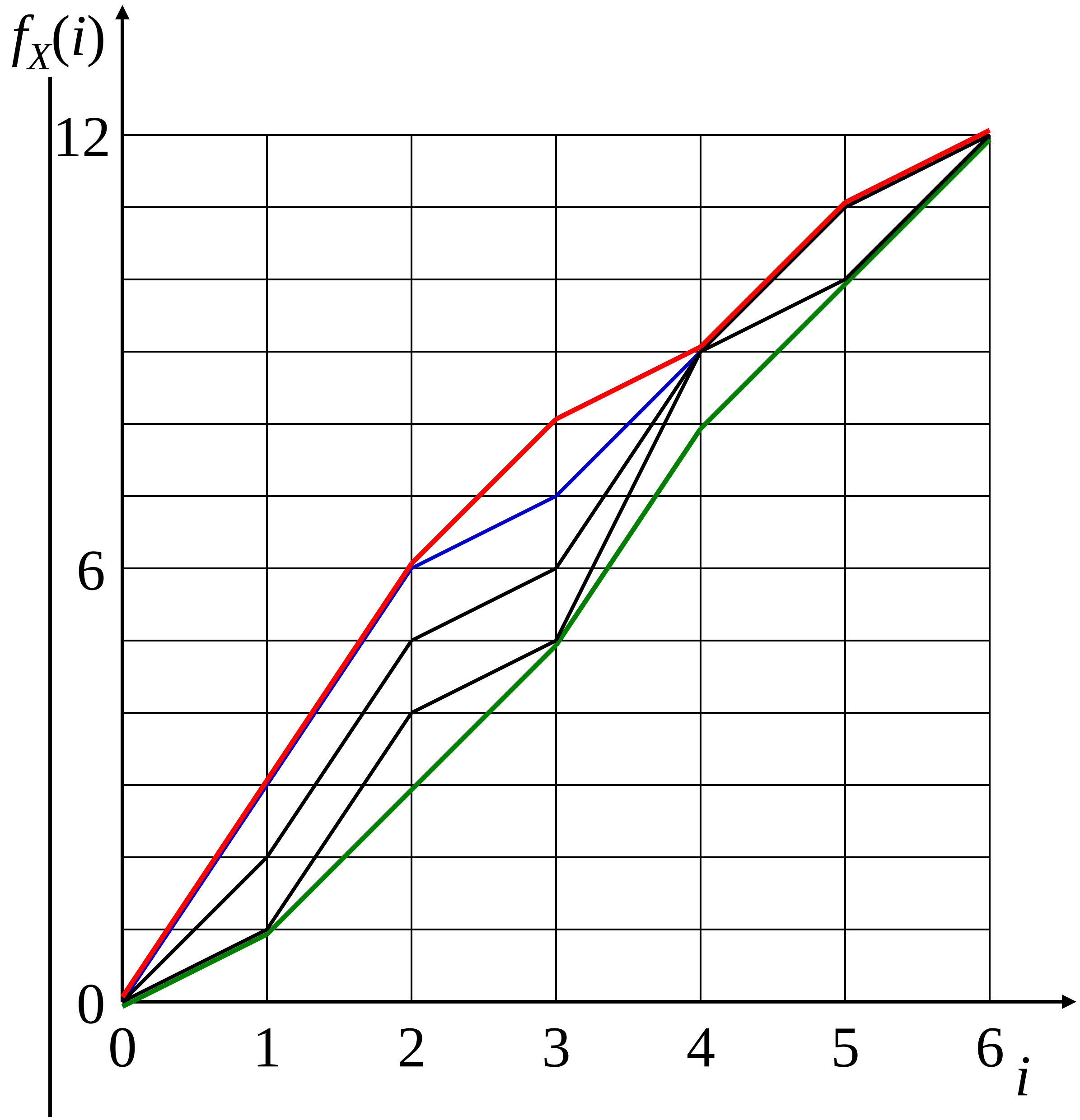


coalescence イメージ

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

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

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

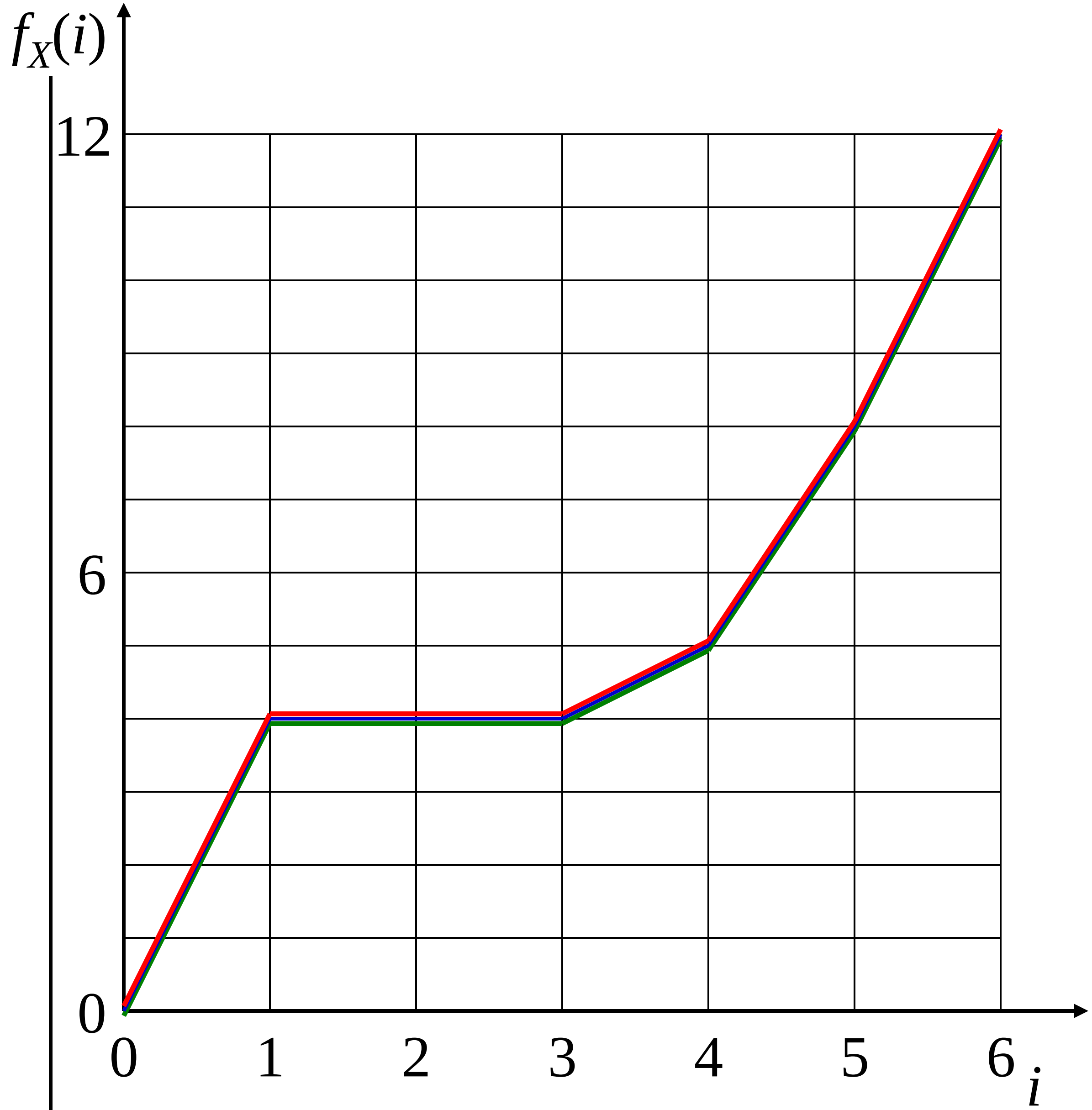


coalescence イメージ

4	0	0	1	3	4	12
1	4	3	6	2	2	18
5	4	3	7	5	6	30

4	0	0	1	3	4	12
1	4	3	6	2	2	18
5	4	3	7	5	6	30

4	0	0	1	3	4	12
1	4	3	6	2	2	18
5	4	3	7	5	6	30



-1: Many Remarks

計算時間

条件

列和ベクトル $\boldsymbol{s}$ は $s_1 \geq s_2 \geq \cdots \geq s_n$ を満たす。

主張

期待計算時間 = $O(n^3 \ln N)$.

						12
						18
5	4	3	7	5	6	30

- CFTPの期待計算時間 = $O (E [T_*])$
  (T_* : coalescence time)
- monotone CFTP の coalescence time (Propp and Wilson ’96)
  - ➢ $E [T_*] \leq 2 \tau (1 + \ln D)$
    (τ : mixing rate, D : マルコフ連鎖の直径)
- マルコフ連鎖の mixing rate
  - ➢ $\tau = n^2 (n-1) \ln N, \ D \leq n N$
    (手法 : path coupling)

MCMC完璧サンプリングの設計例

単調なマルコフ連鎖

- ✓ Isingモデル
- ✓ Pottsモデル（クラッターモデル） [Propp &Wilson 96]
- ✓ tiling [Wilson 01]
- ✓ 2行分割表 [K & Matsui 06]
- ✓ 離散化Dirichlet分布 [Matsui & K 07]
- ✓ 待ち行列ネットワーク [K & Matsui 08]
- ✓ Q-Ising モデル [Yamamoto, K & Matsui 08]

それ以外のCFTP

- ✓ 根付き全張木 [Propp &Wilson 98]

乱数ベクトルの記憶容量に関する問題

- ➢ Wilson の read onceアルゴリズム [Wilson 00]
- ➢ Fillのinterruptibleアルゴリズム [Fill 98]

CFTPを使おう！

利点

- ✓ **精度保証付き** (完璧！)
- ✓ **自動終了**アルゴリズム
- ✓ (精度を保証する意味で)“**速い**”

欠点

- ✓ 単調マルコフ連鎖の設計が難しい
- ✓ コーディングも (慣れるまでは) 煩雑

推奨使用法

- ✓ 精度保証が必要
  - ➢ **完璧サンプリング** (read once (ver. twin-run)を推奨)
- ✓ ヒューリスティクス
  - ➢ **simple MCMC**

単調CFTP --最近の展開

FAQ

「どうやって、単調マルコフ連鎖は設計するのか？」

単調CFTP --最近の展開

FAQ

「どうやって、単調マルコフ連鎖は設計するのか？」

Hasse図上のrandom walk

定理 [K '09+]

分配束上のreversible Hasse walk が 単調な更新関数 を持つ

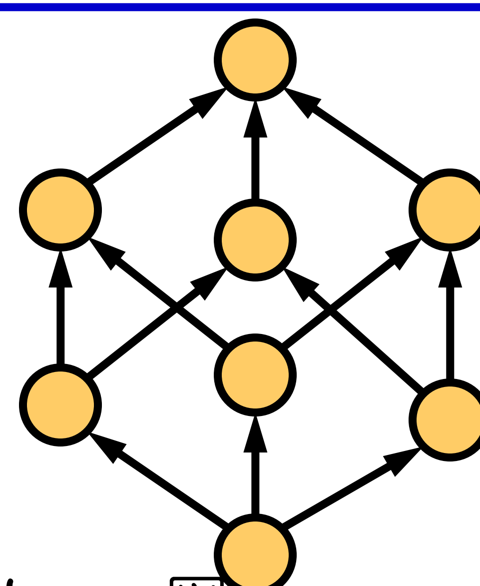
$\Leftrightarrow$ 定常分布が対数優モジュラ

Remark: 対数優モジュラ分布

標本空間: **分配束** D ($\subseteq \{0,1\}^E$)

確率分布: **$\pi(x)$**: 対数優モジュラ

($\Leftrightarrow \forall x,y,\ h(x) \cdot h(y) \leq h(x \cup y) \cdot h(x \cap y)$)



Hasse 図

0: *The end*

— all of your views coalesce.

Thank you for the attention.