May 23, 2025

14. Deterministic Random Walk

来嶋 秀治

滋賀大学 データサイエンス学部

計算機科学のMCMC法

FPRAS

“Valiant計画” (since 1979)

#P困難な$A \in \mathbb{R}$に対して，近似解$Z \in \mathbb{R}$を精度

$$\Pr[(1-\epsilon)A \leq Z \leq (1+\epsilon)A] \geq 1-\delta$$

で多項式時間$(\mathrm{poly}(\mathrm{input}, \epsilon^{-1}, \log \delta^{-1}))$で求められるか？

そもそも，乱数は必須か？

高次元多面体の体積計算の決定性近似

Lovász [1986]

“membership oracleで与えられたn次元凸体の体積一般に対して,

近似比 1.999^n の**決定性**多項式時間アルゴリズムは**存在しない**. ”

高次元多面体の体積計算の決定性近似

Lovász [1986]

“membership oracleで与えられた n次元凸体の体積一般に対して，

近似比 1.999^n の**決定性**多項式時間アルゴリズムは**存在しない**．”

乱択近似 (1980～)

- ✓ Jerrum, Valiant, Vazirani (1986): 近似数え上げ≃ランダムサンプリング
- ✓ Toda (1991): $\mathrm{PH} \subseteq \mathrm{P}^{\#\mathrm{P}}$ (Gödel賞)
- ✓ Dyer, Frieze, Kanna (1991): 高次元凸体体積の乱択近似可能 (Fulkerson賞)
- ✓ Jerrum, Sinclair, Vigoda (2004): パーマネントの乱択近似可能 (Fulkerson賞)

高次元多面体の体積計算の決定性近似

Lovász [1986]

“membership oracleで与えられた n次元凸体の体積一般に対して,

近似比 1.999^n の**決定性**多項式時間アルゴリズムは**存在しない**. ”

乱択近似 (1980～)

- ✓ Jerrum, Valiant, Vazirani (1986): 近似数え上げ≃ランダムサンプリング
- ✓ Toda (1991): $\mathrm{PH} \subseteq \mathrm{P}^{\#\mathrm{P}}$ (Gödel賞)
- ✓ Dyer, Frieze, Kanna (1991): 高次元凸体体積の乱択近似可能 (Fulkerson賞)
- ✓ Jerrum, Sinclair, Vigoda (2004): パーマネントの乱択近似可能 (Fulkerson賞)

決定性近似 (話者の研究 2008～)

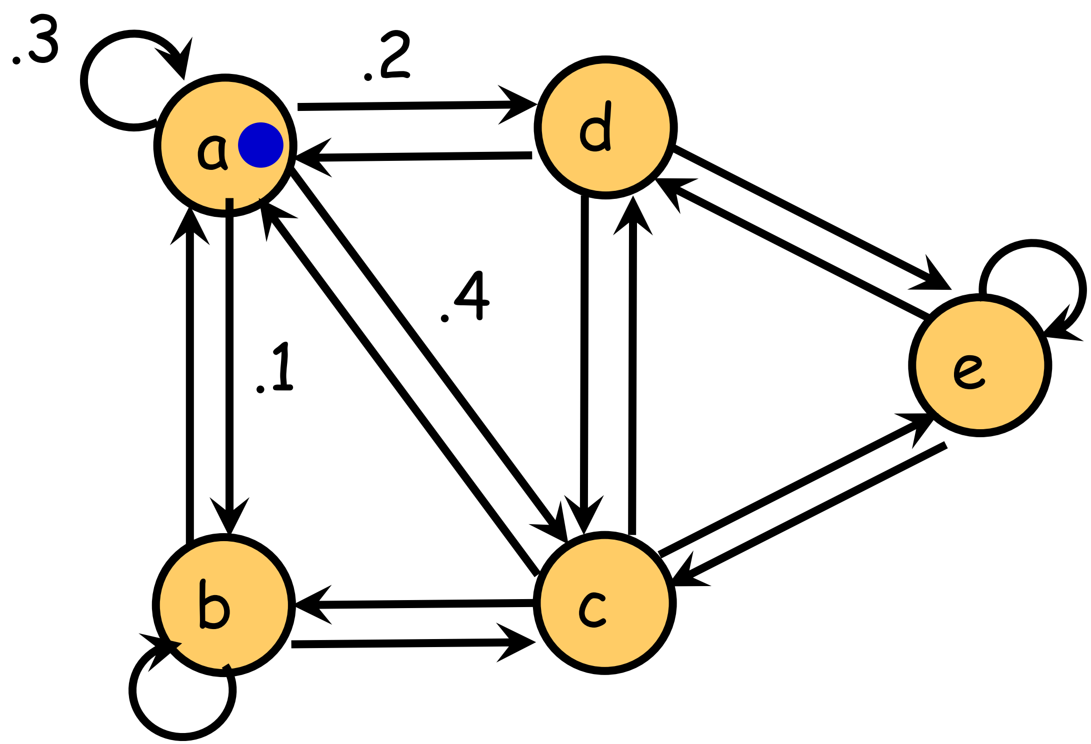
- ✓ #P困難な体積計算に対する決定性近似 (FPTAS)
  - ナップサック多面体 (H多面体: 線形不等式系で記述される多面体)
  - ‘双対’ナップサック多面体 (V多面体: 端点集合で記述される多面体)
- ✓ Deterministic random walk
  - マルコフ連鎖 (MCMC)を決定性過程に置き換えられるか？

決定性ランダムウォーク

ランダムウォーク

トークンがグラフ上をランダムウォークする.

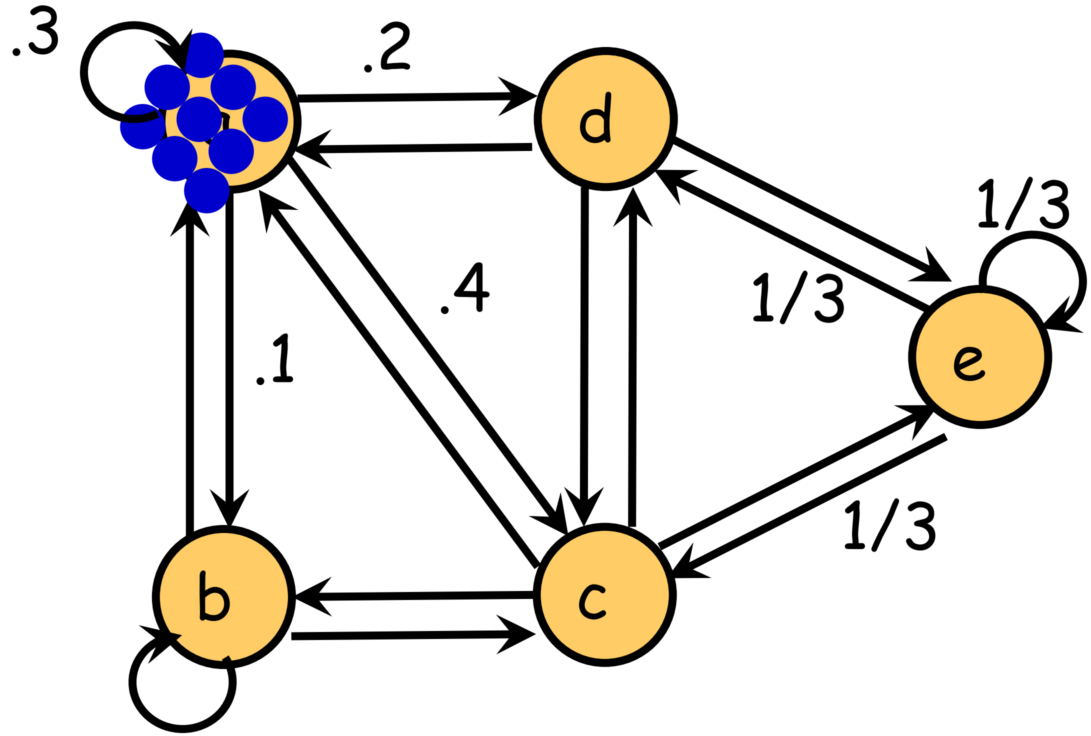
- ✓ π^0: 初期分布 (トークンは確率$\pi^0{}_v$で頂点vに居る)
- ✓ P: 推移確率行列 (確率P_{uv}で頂点u頂点vに移動)
- ✓ 時刻tの確率分布$\pi^t:=\pi^0P^t$ (頂点vに居る確率 $(\pi^t)_v$)



ランダムウォーク (複数トークンによる分布の近似)

Nの個のトークンがグラフ上を独立にランダムウォークする.

- ✓ μ^0: 初期配置 ($\pi^0 \approx \mu^0/N$)
- ✓ P: 推移確率行列 (確率P_{uv}で頂点u頂点vに移動)
- ✓ 時刻tの期待配置$\mu^t := \mu^0 P^t$ ($\pi^t \approx \mu^t/N$)



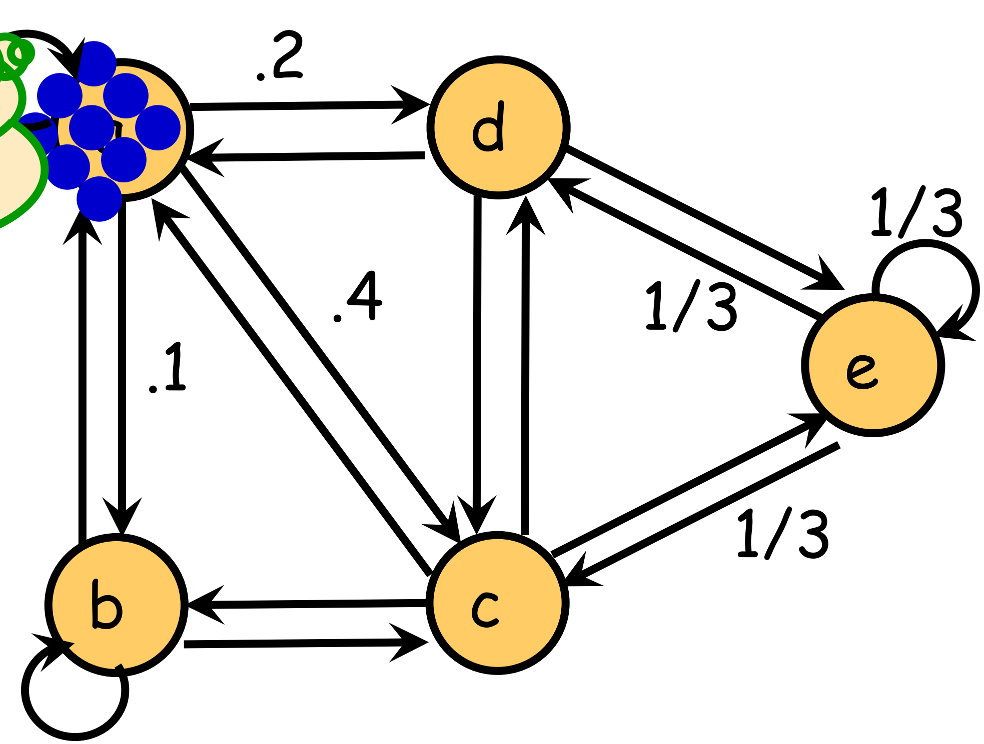
ランダムウォーク (複数トークンによる分布の近似)

N個のトークンがグラフ上を独立にランダムウォークする.

- ✓ μ^0: 初期配置 ($\pi^0 \approx \mu^0/N$)
- ✓ P: 推移確率行列 (確率P_{uv}で頂点u頂点vに移動)
- ✓ 時刻tの期待配置$\mu^t := \mu^0 P^t$ ($\pi^t \approx \mu^t/N$)

(Nが非常に大きければ,)
トークンを約2:4:1:3の割合で
隣接点にばらまいている.

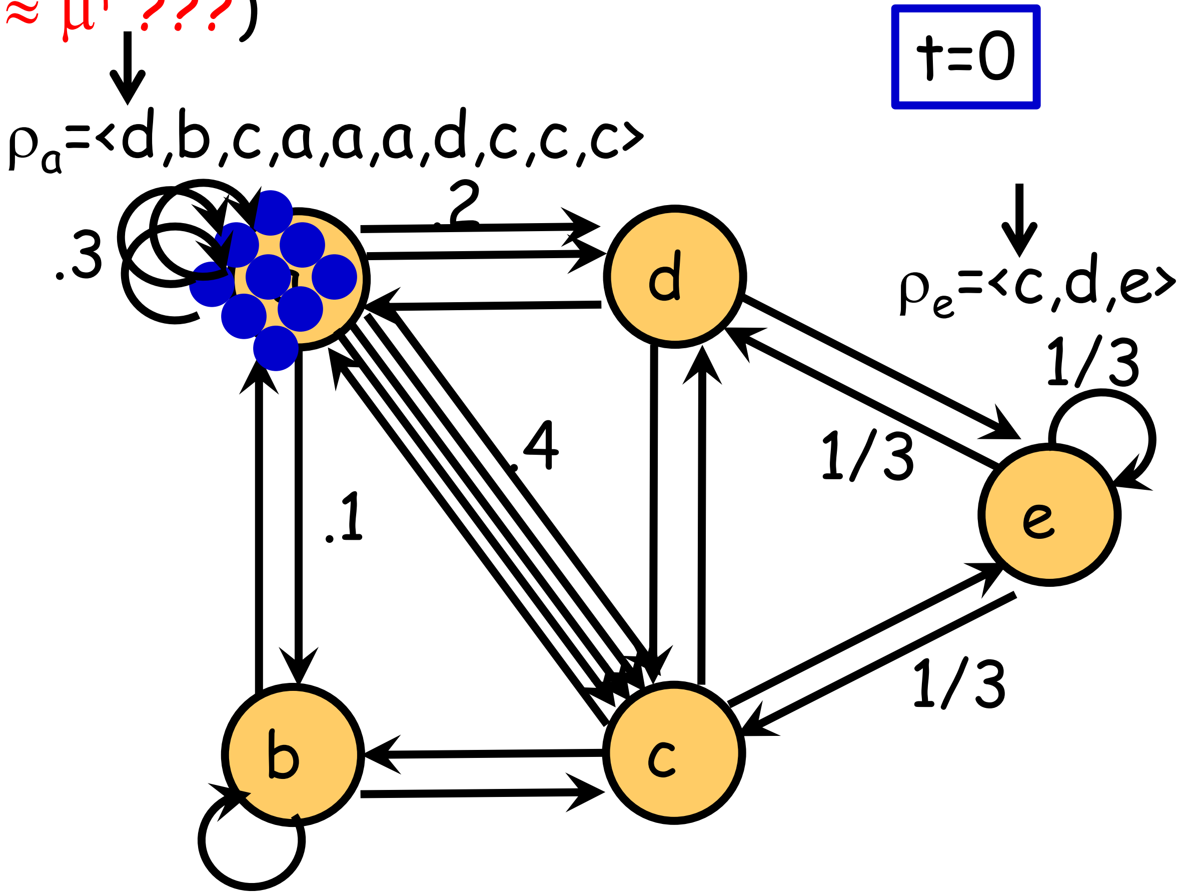
決定的過程でも大差ない?



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

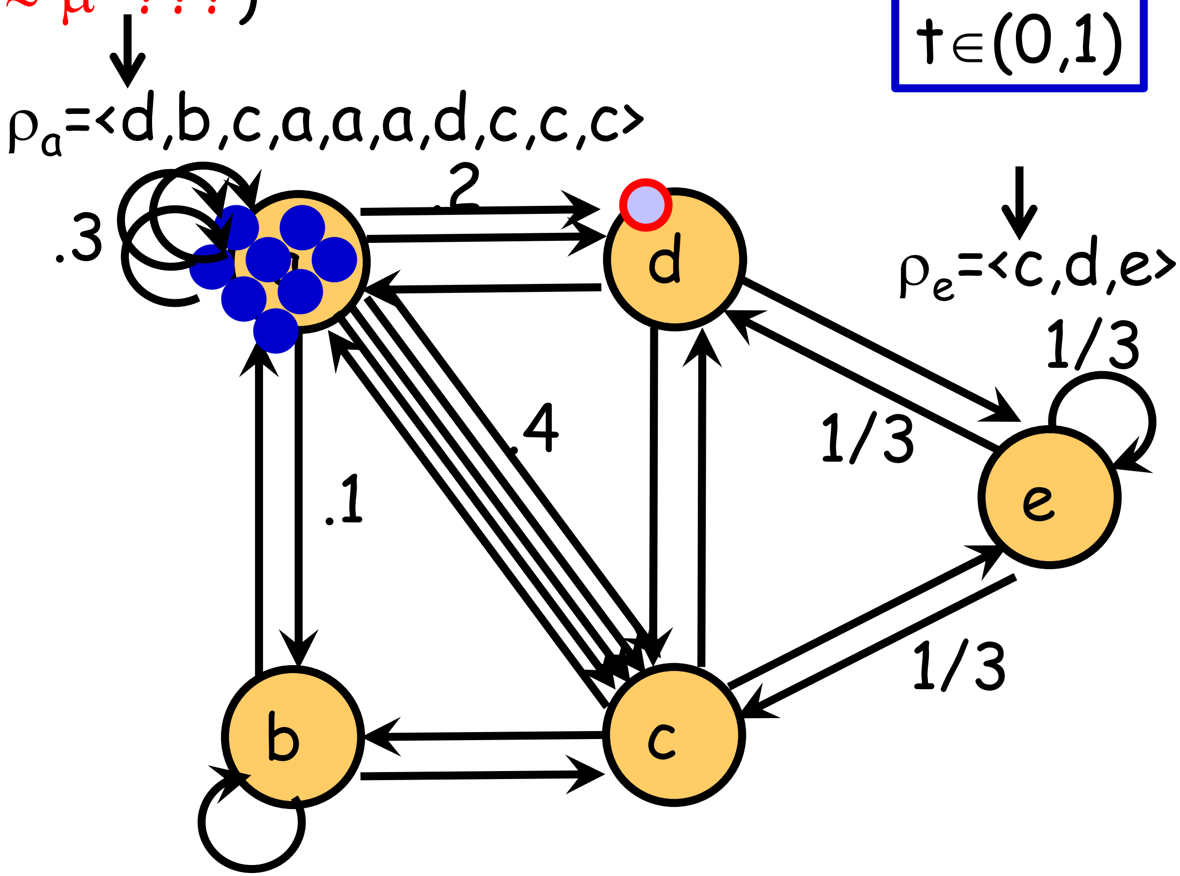


deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

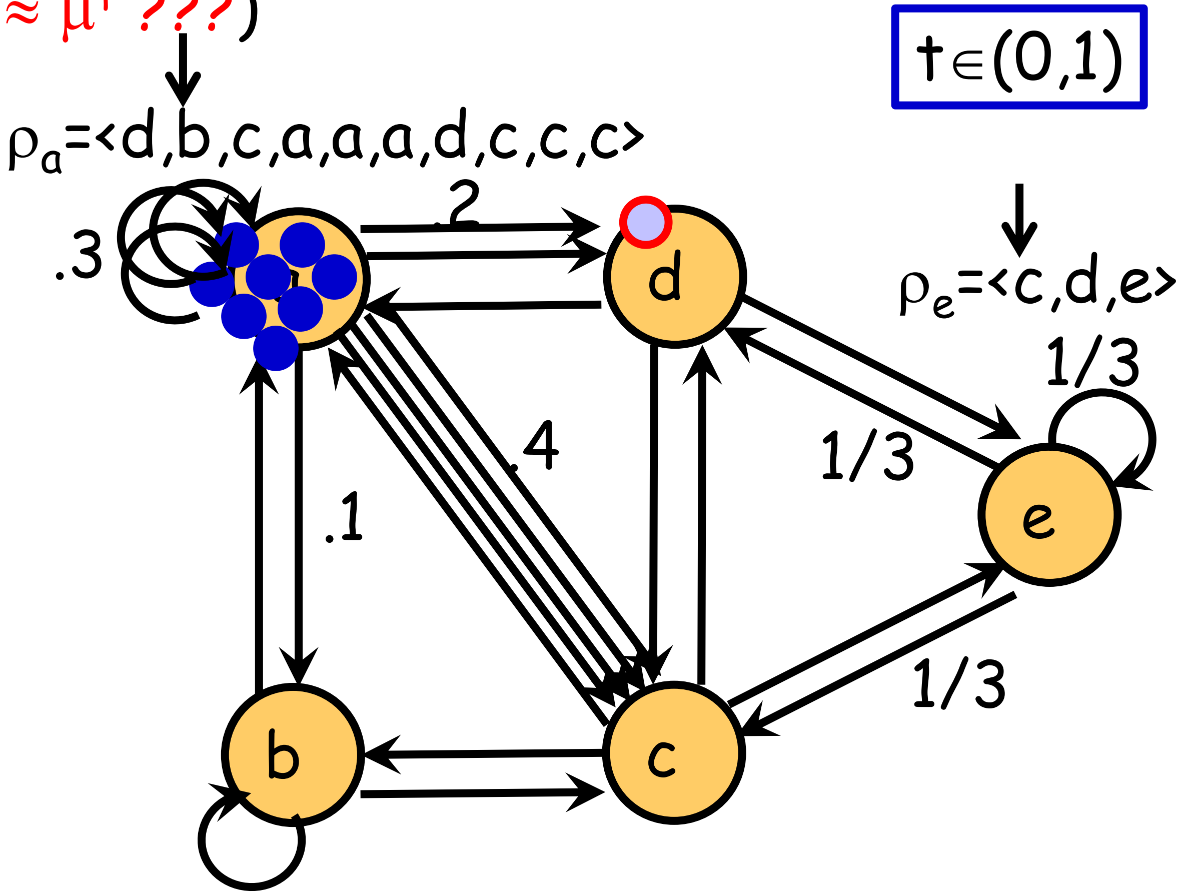
$t \in (0,1)$



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)

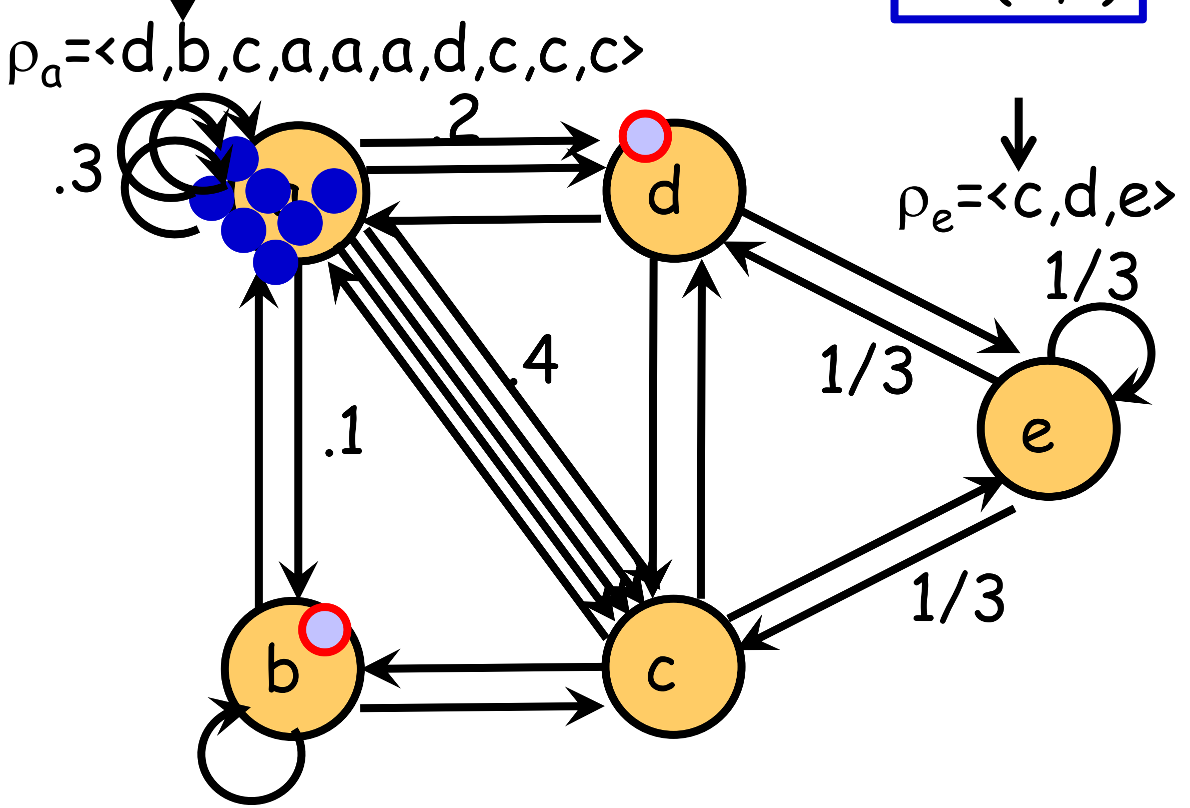


deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)

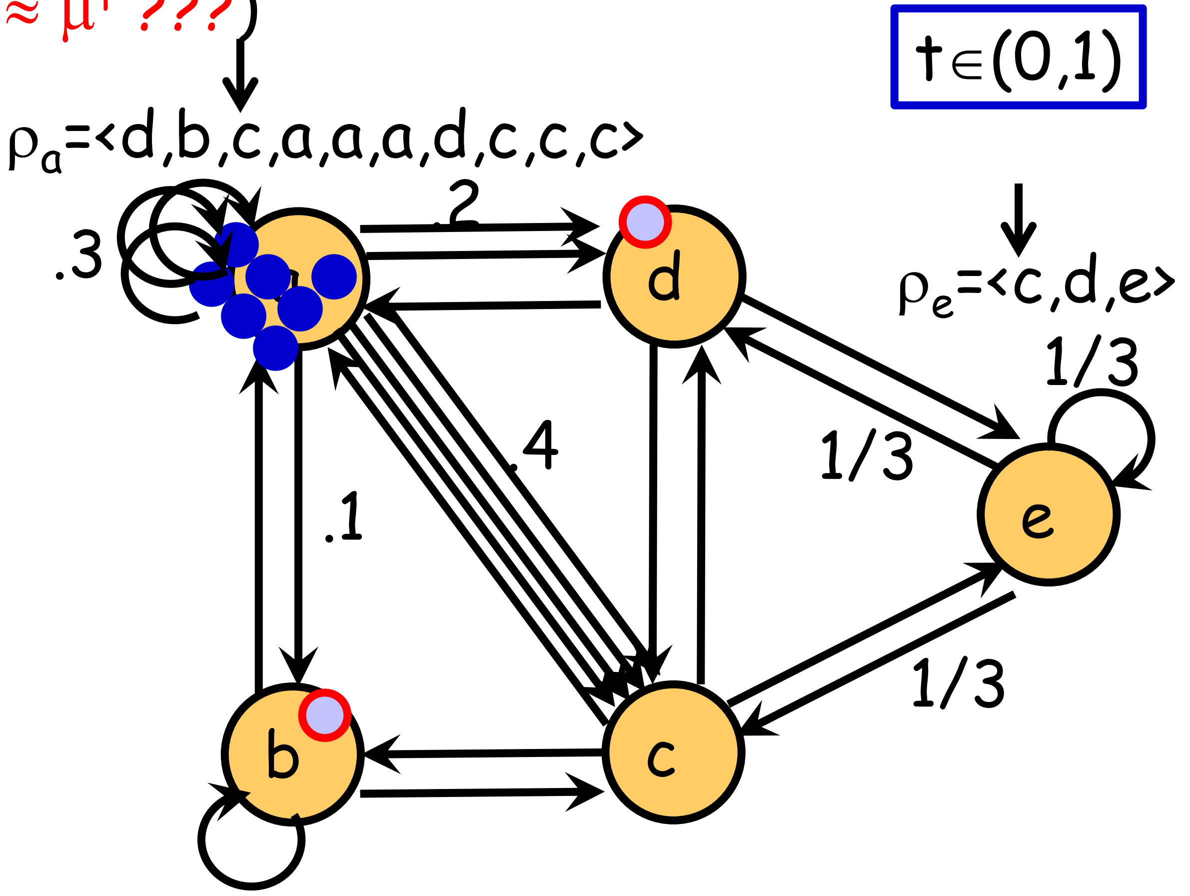
$t \in (0,1)$



deterministic RW (Propp機械; rotor-router)

Nの**トークン**がグラフ上を移動する.

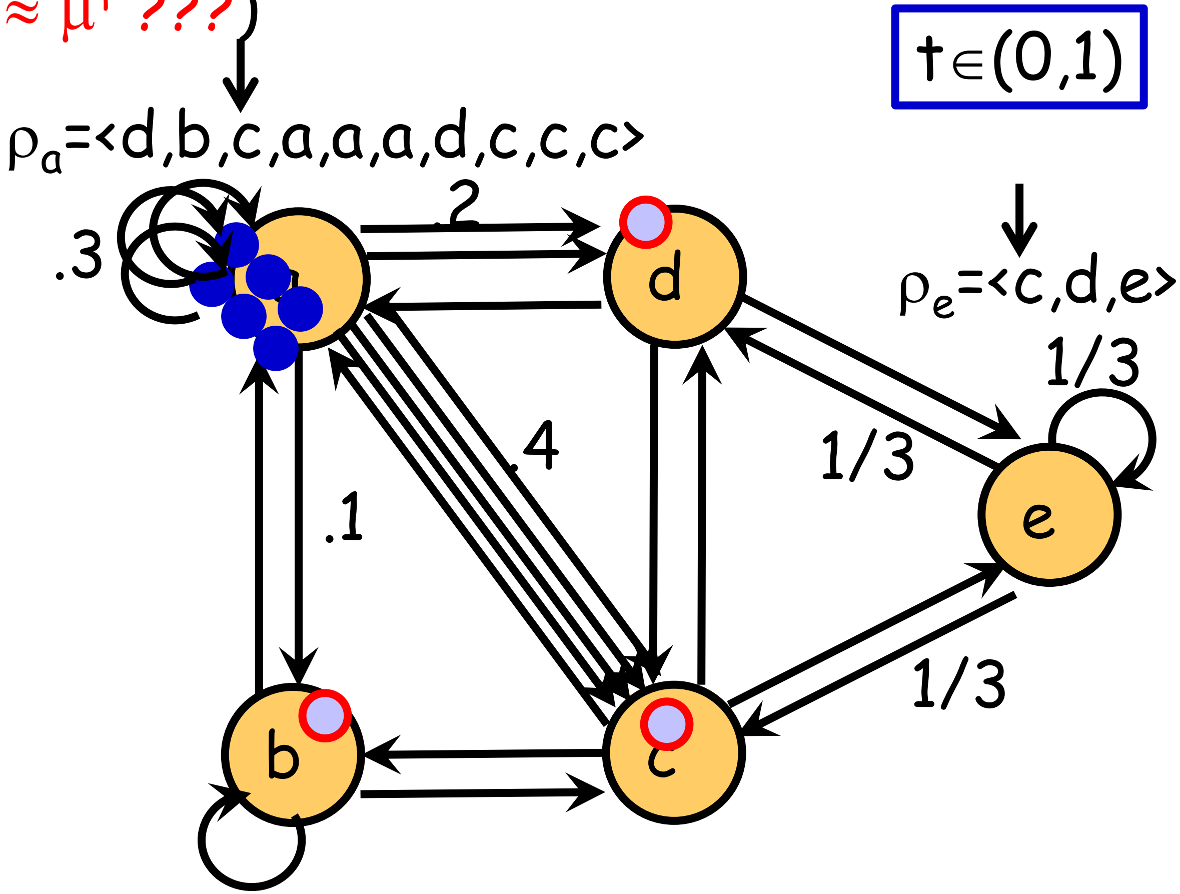
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

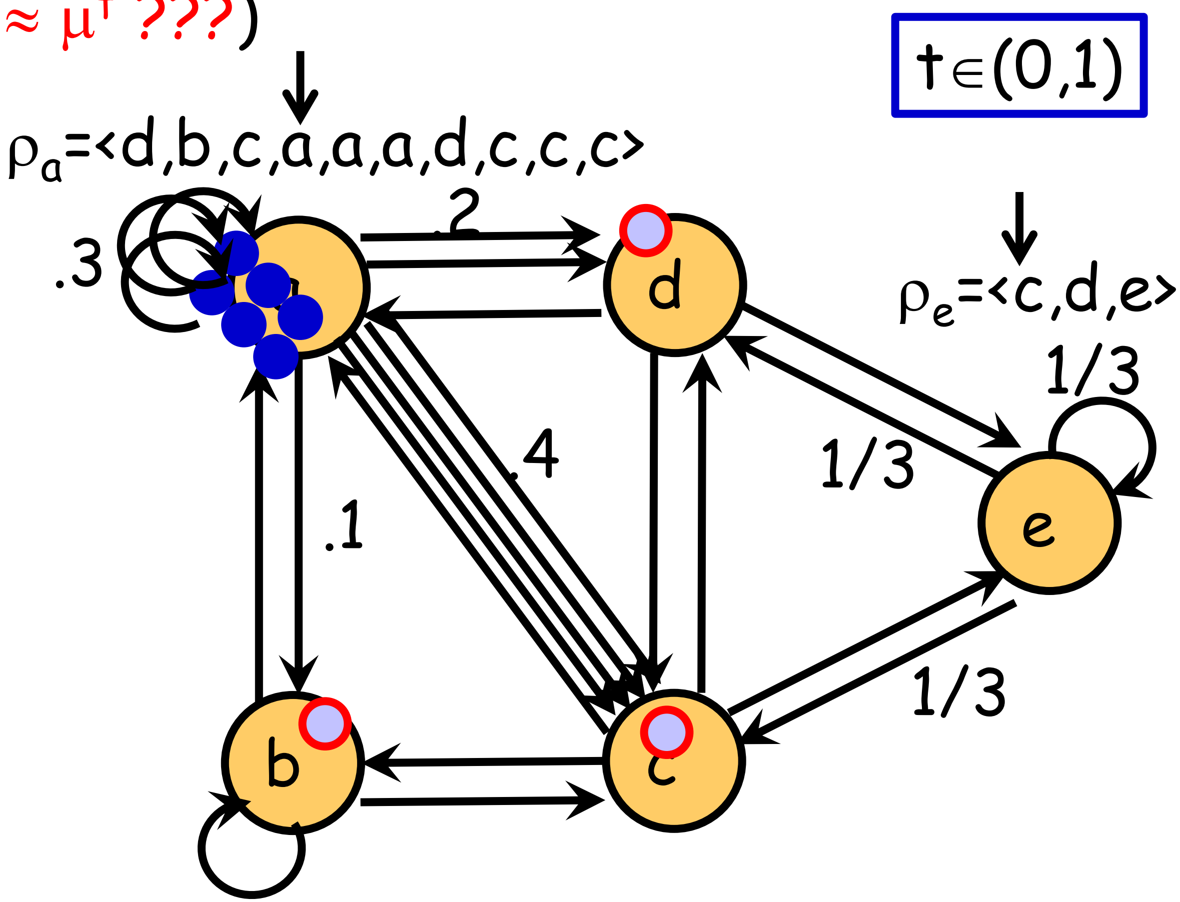
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

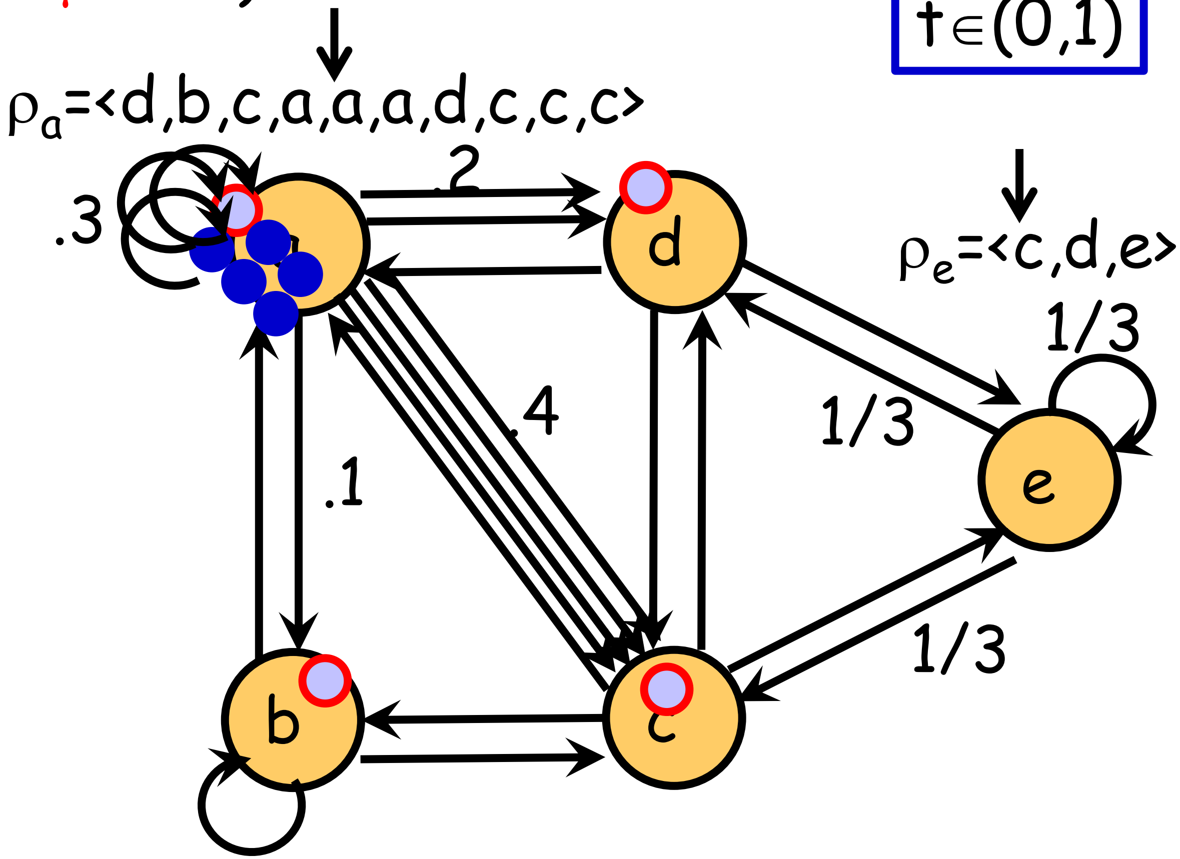


deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)

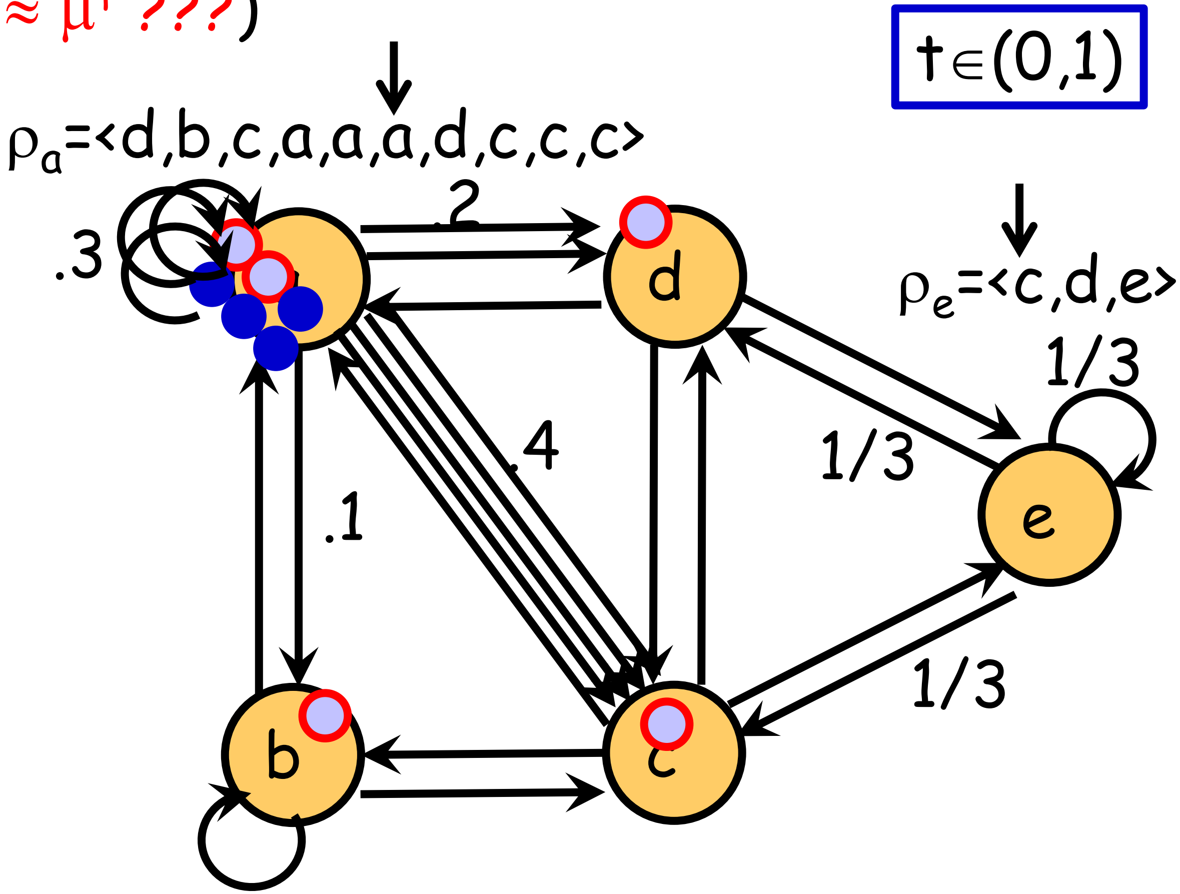
$t \in (0,1)$



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)

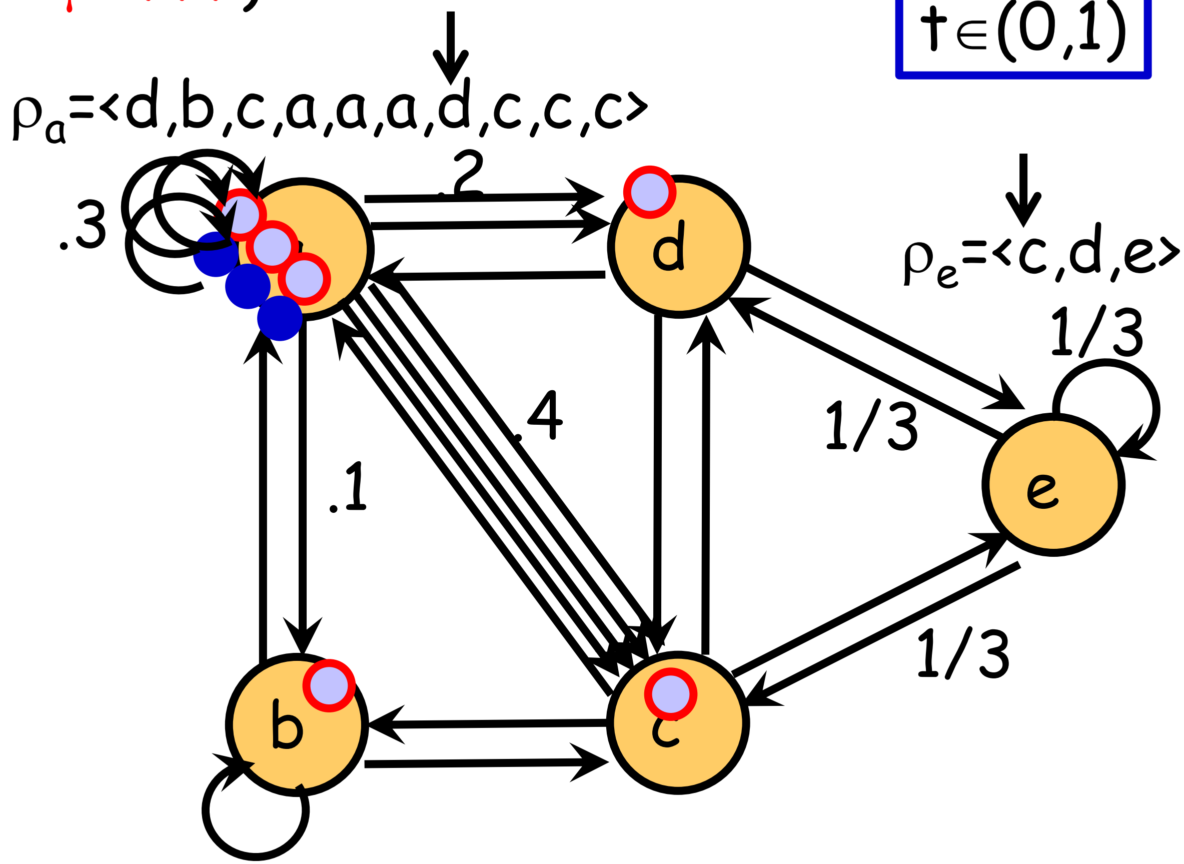


deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: “**rotor router**” (比率P_{uv}で頂点u頂点vに“順次”移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)

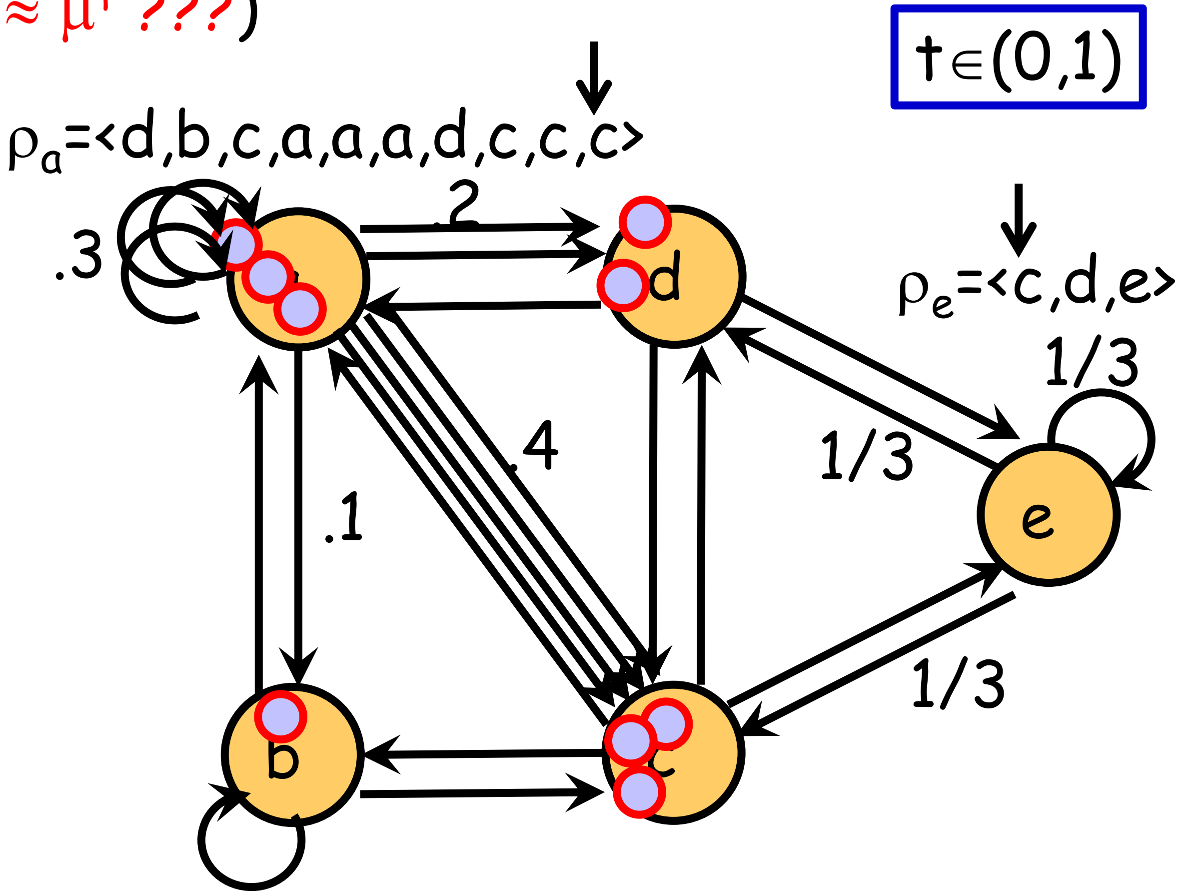
$t \in (0,1)$



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

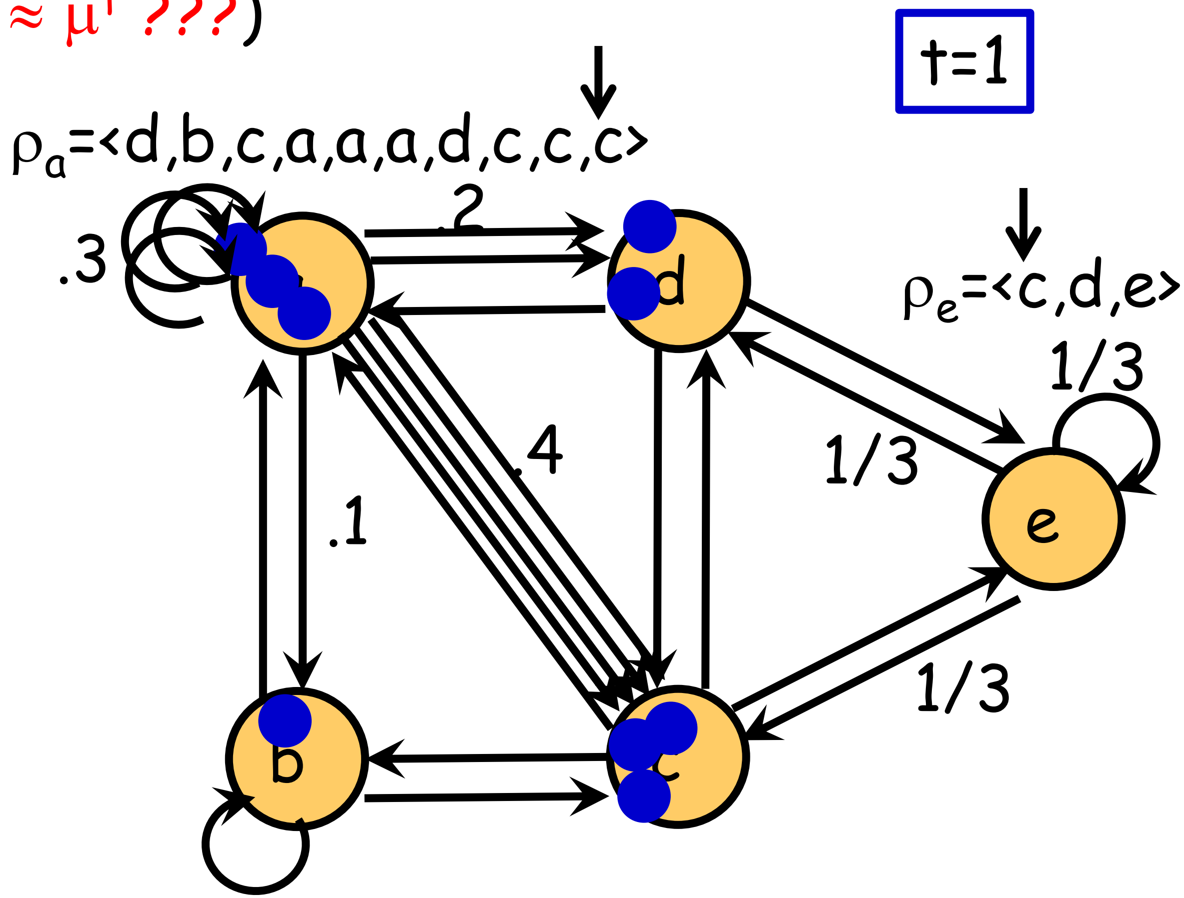
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

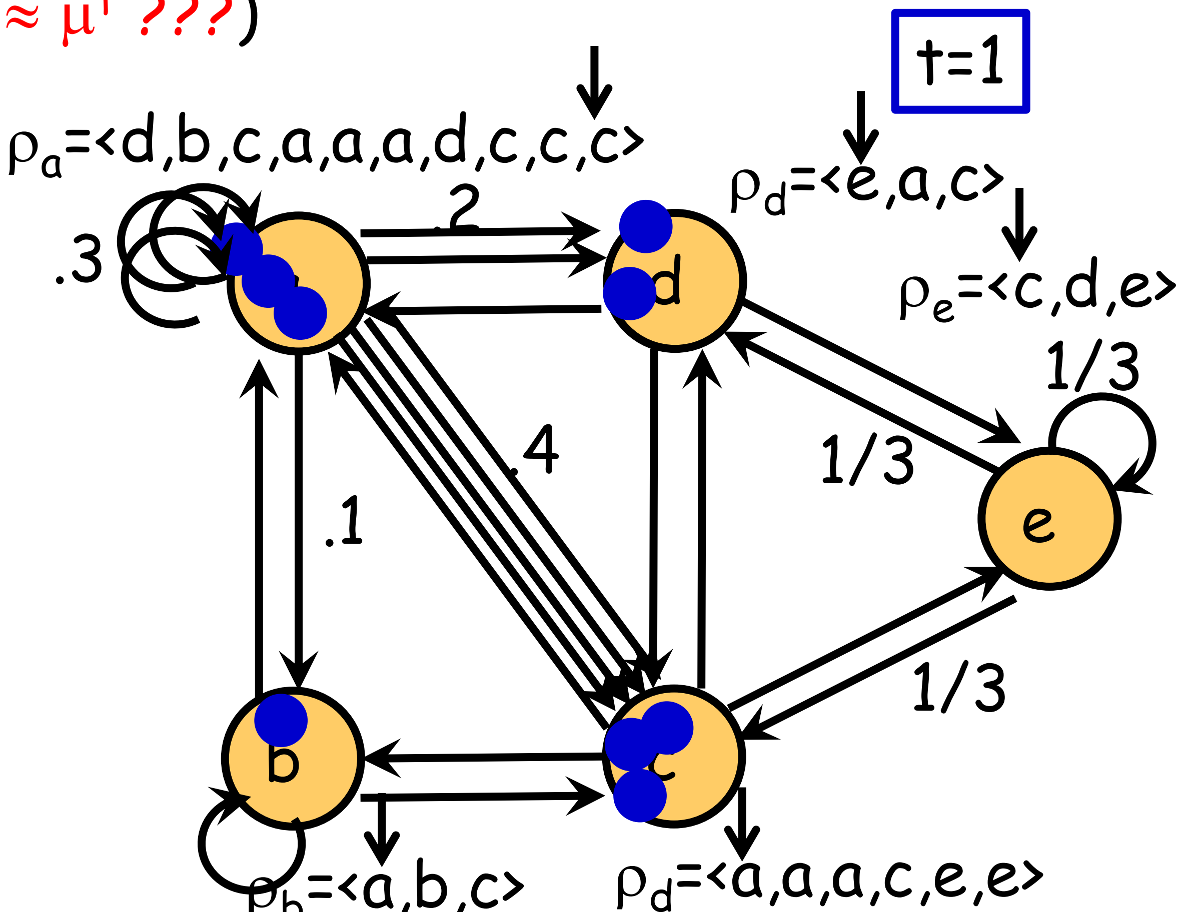
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

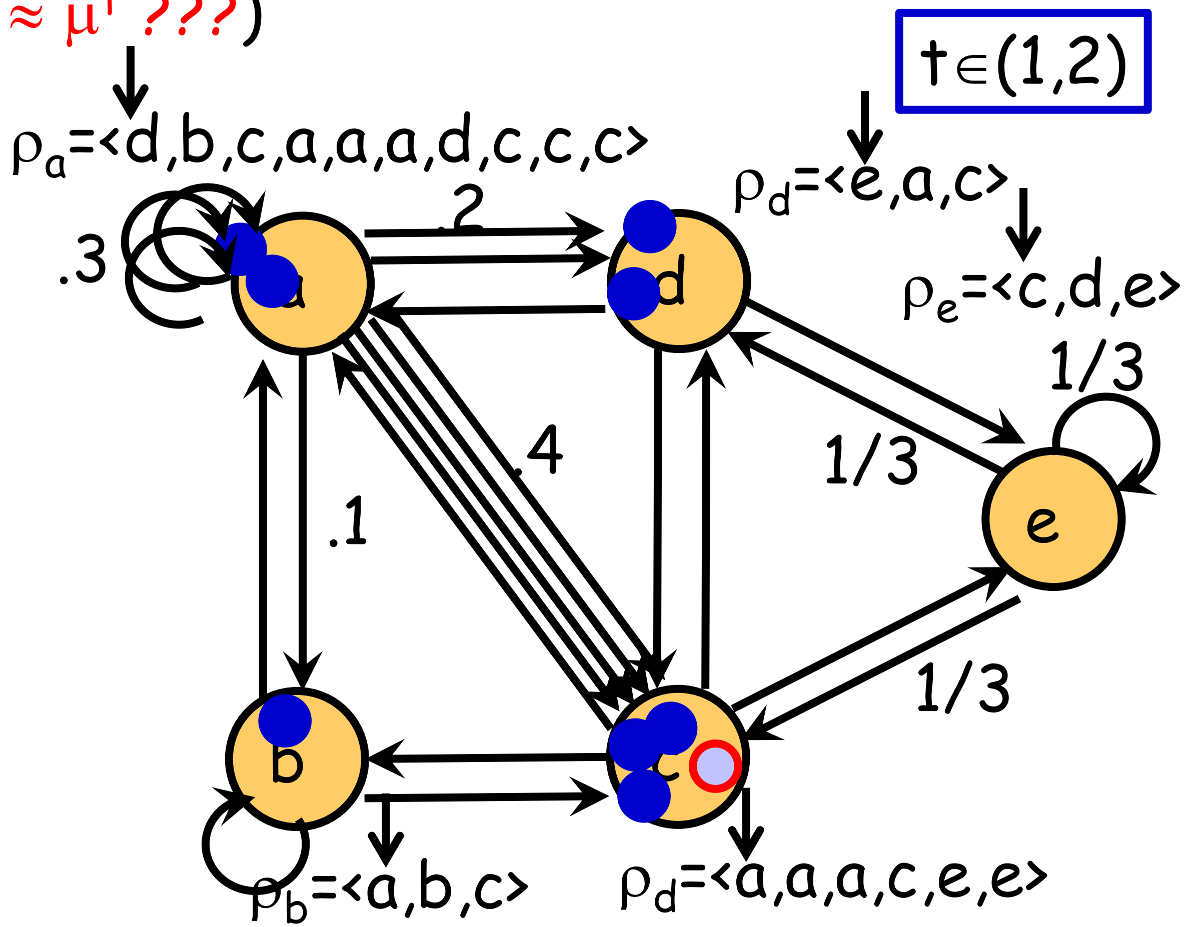
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

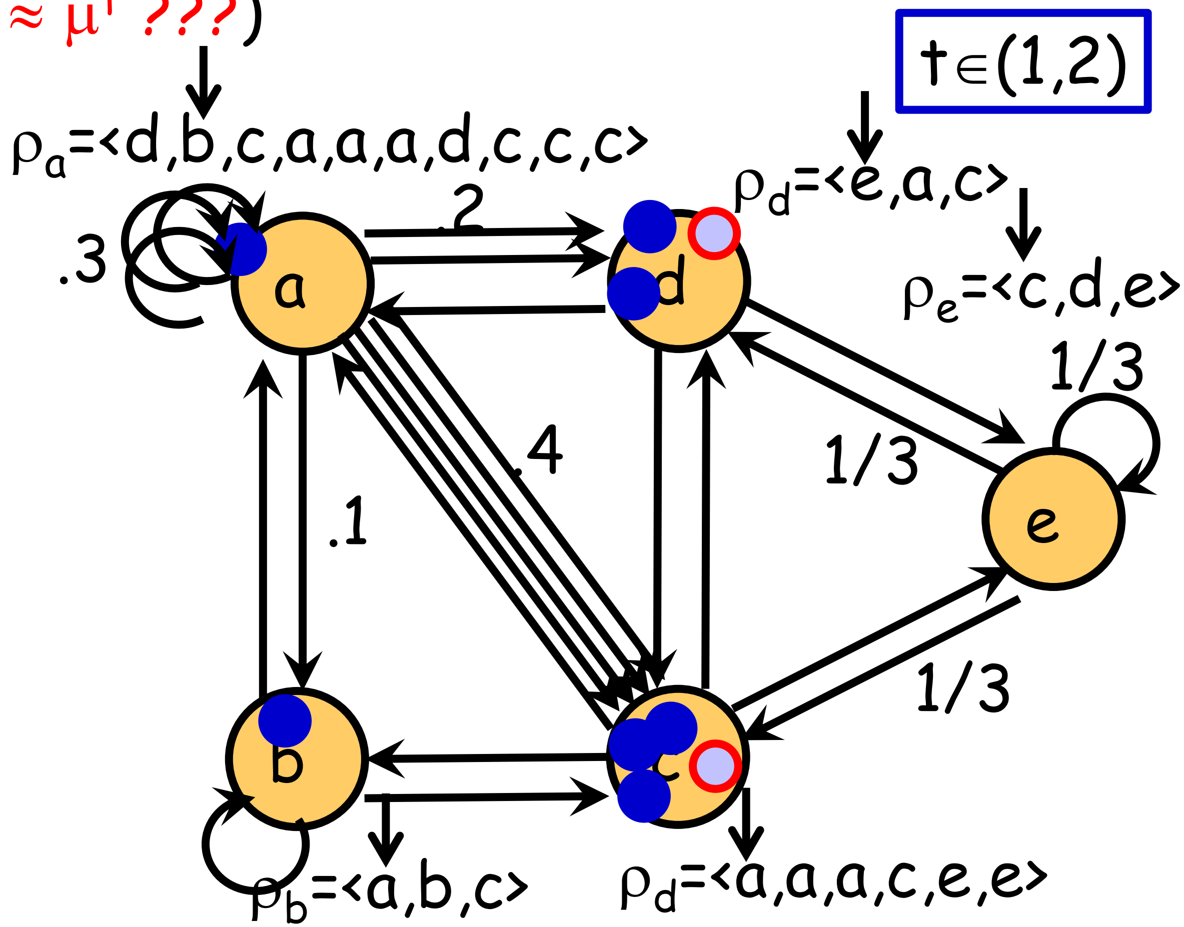
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

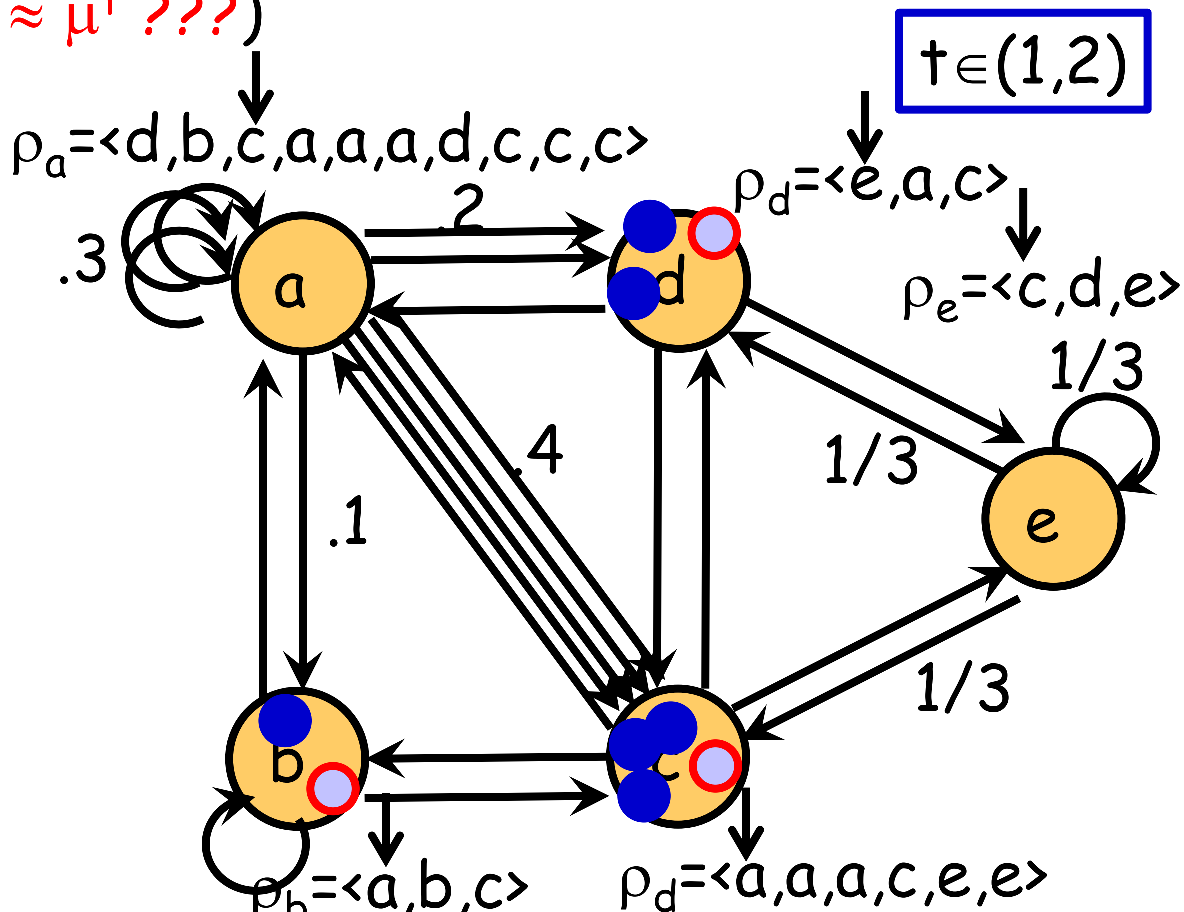
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

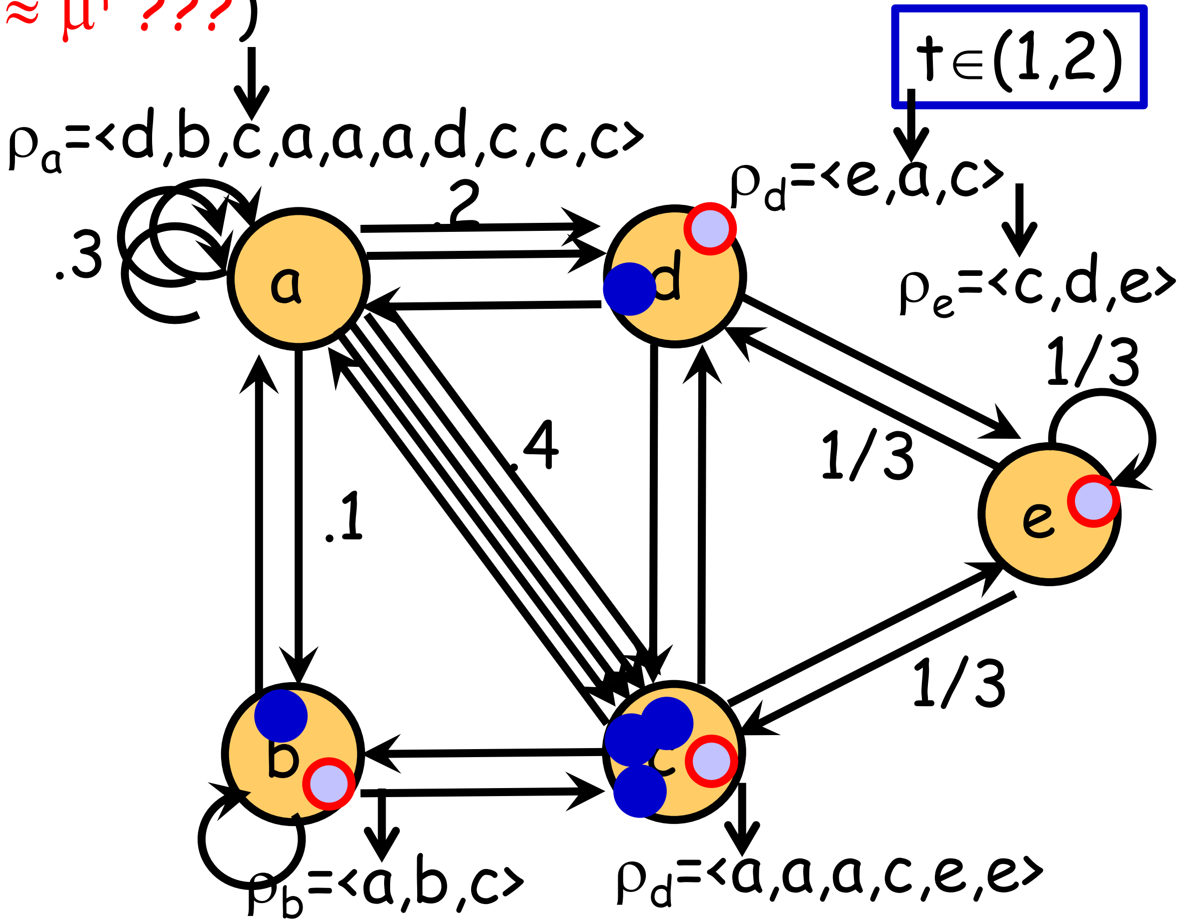
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

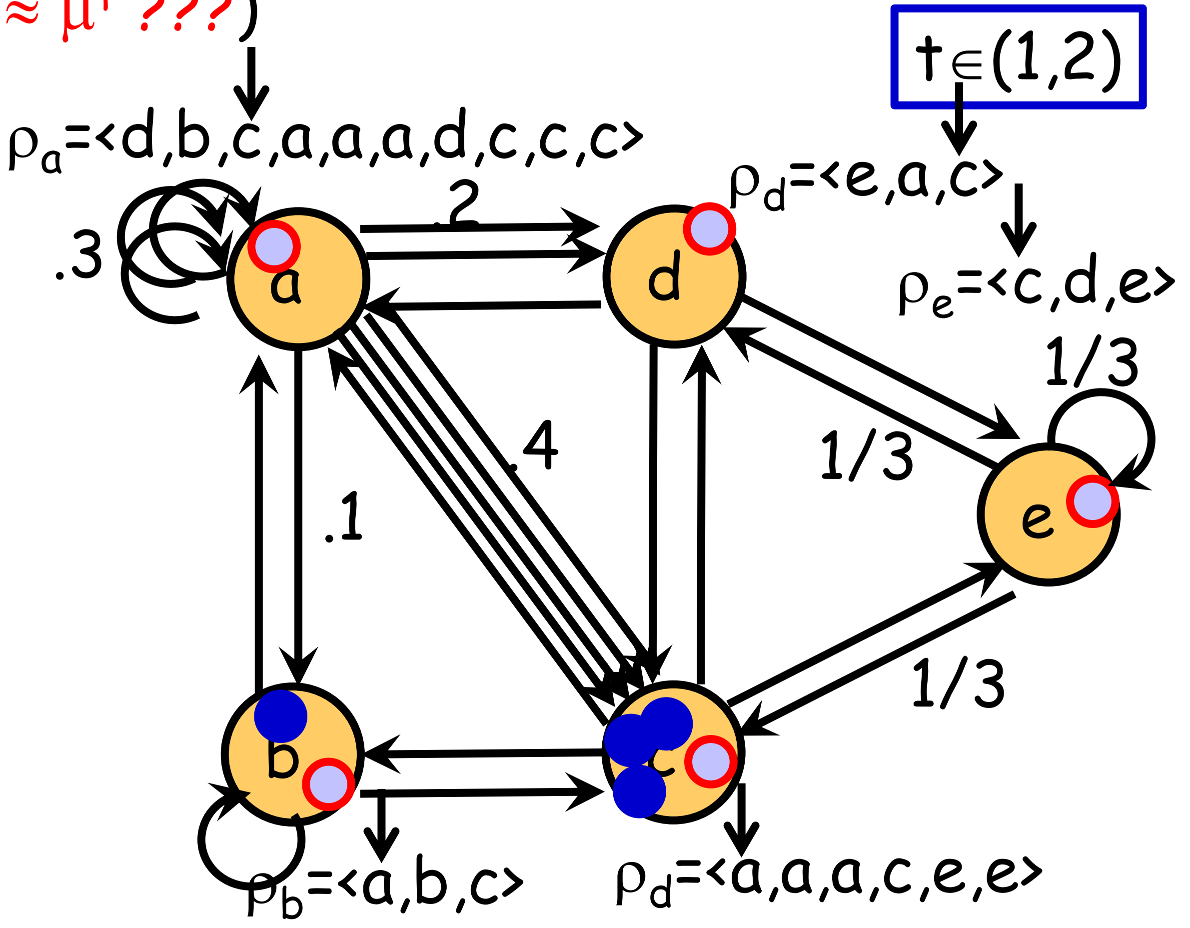
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

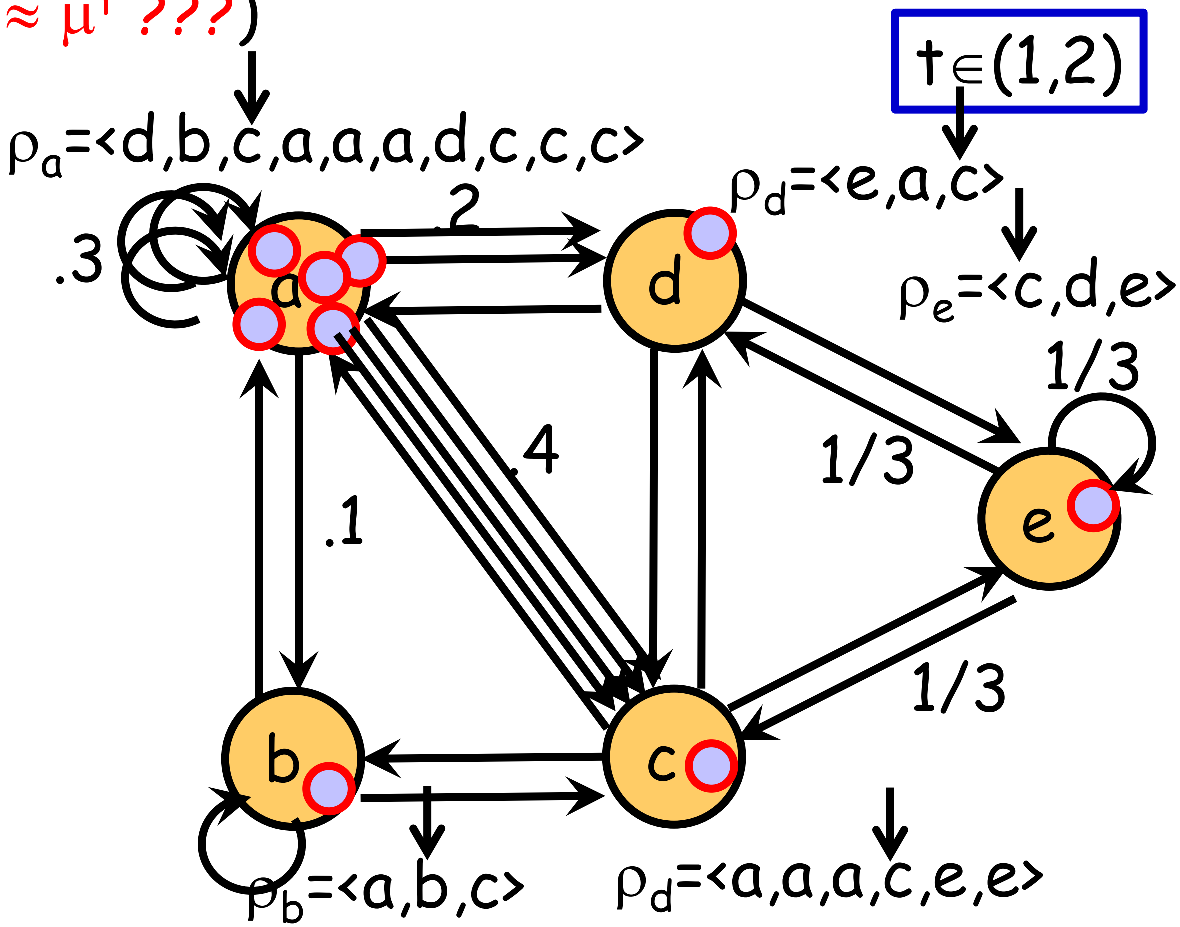
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

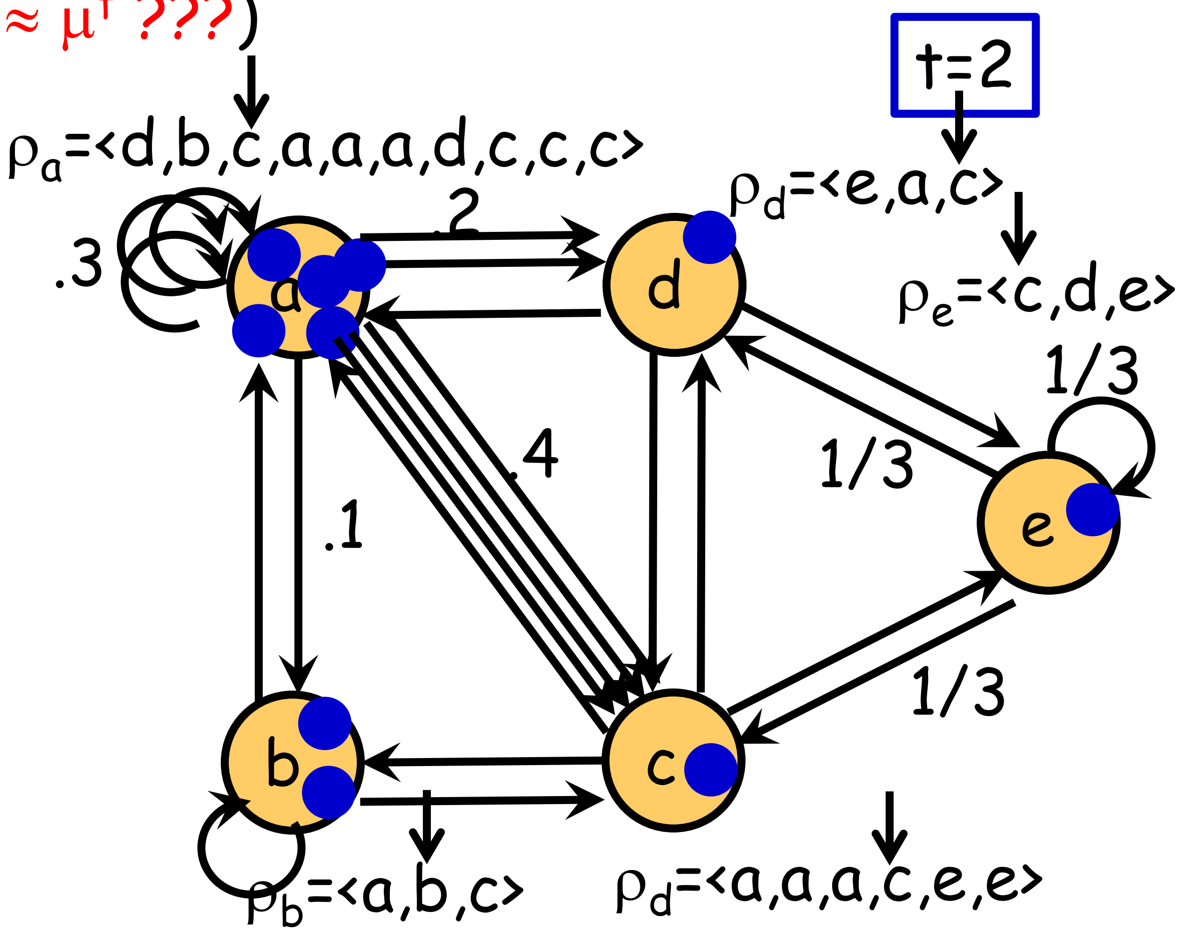
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個のトークンがグラフ上を移動する.

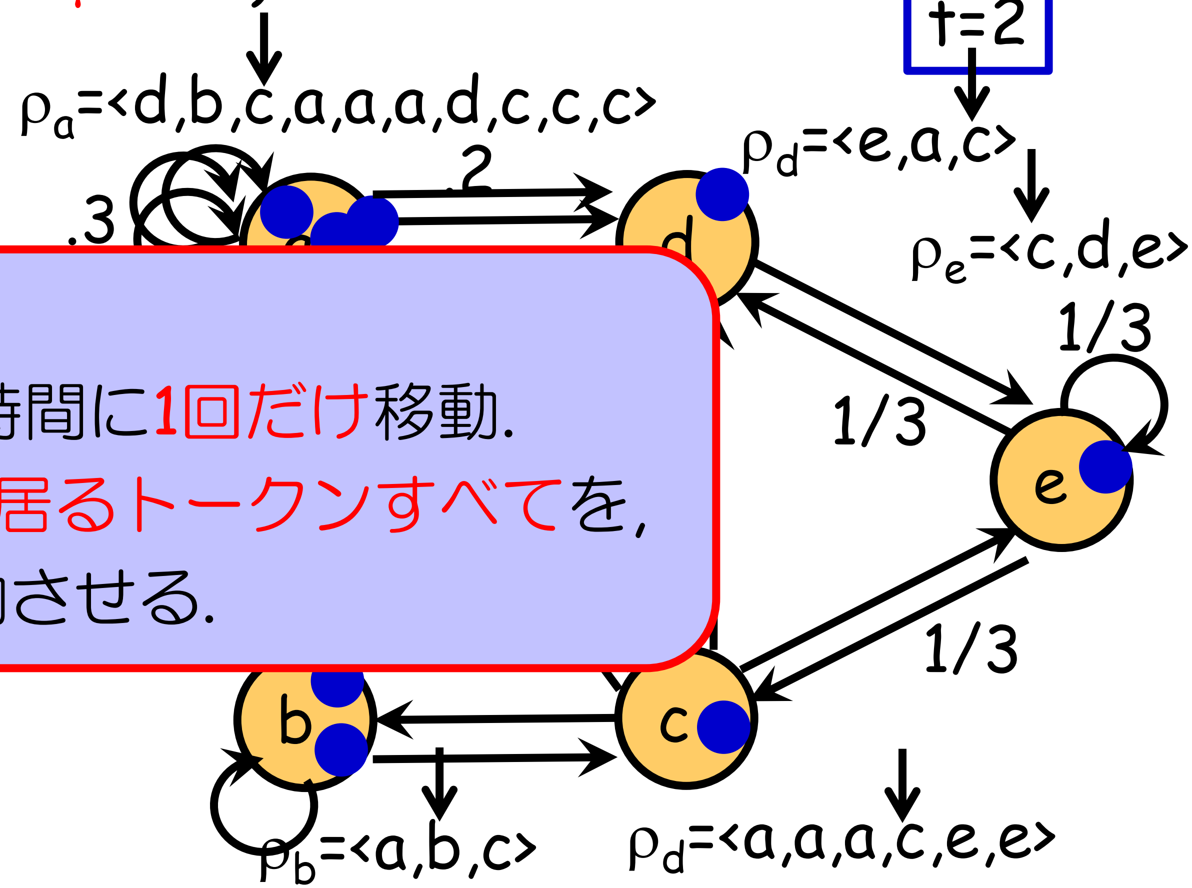
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



deterministic RW (Propp機械; rotor-router)

N個の**トークン**がグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ ρ: "**rotor router**" (比率P_{uv}で頂点u頂点vに"順次"移動)
- ✓ 時刻**t**の配置χ^t ($\chi^t \approx \mu^t$???)



Remark

- ✓ 各トークンは単位時間に**1**回だけ移動.
- ✓ 各ノードは時刻**t**に居るトークンすべてを, 時刻**t+1**までに移動させる.

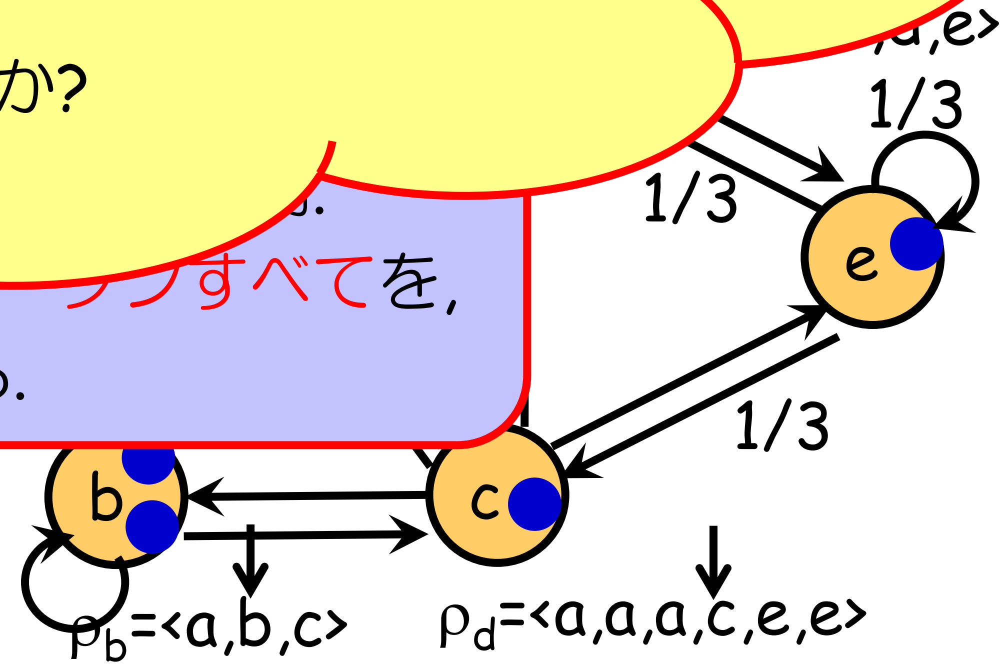
Propp機械はランダムウォークを模倣できるか?

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)

- ✓ (エルゴード)マルコフ連鎖は定常分布に収束する
  - ➢ "平滑化"される.
- ✓ Propp機械はランダムウォークを模倣できるか?
  - ➢ "平滑化"できるか?
  - ➢ "だま"はできないのか?

- ✓ 各ノードは時刻tに居るト すべてを, 時刻t+1までに移動させる.



誤差の下界

定理 [K, Koga, Makino 10+]

ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),

ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

但し, mは多重グラフの頂点数,枝数.

だまのできる例

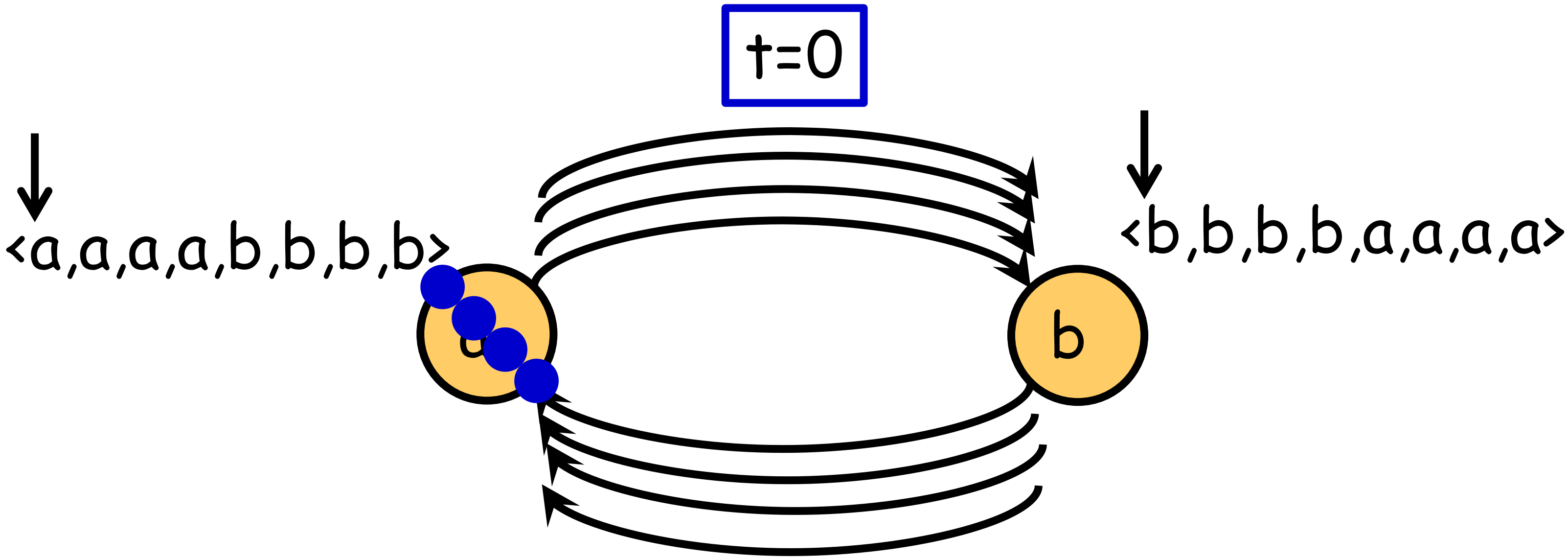
定理 [K, Koga, Makino 10+]

ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),

ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

但し, mは多重グラフの頂点数,枝数.



だまのできる例

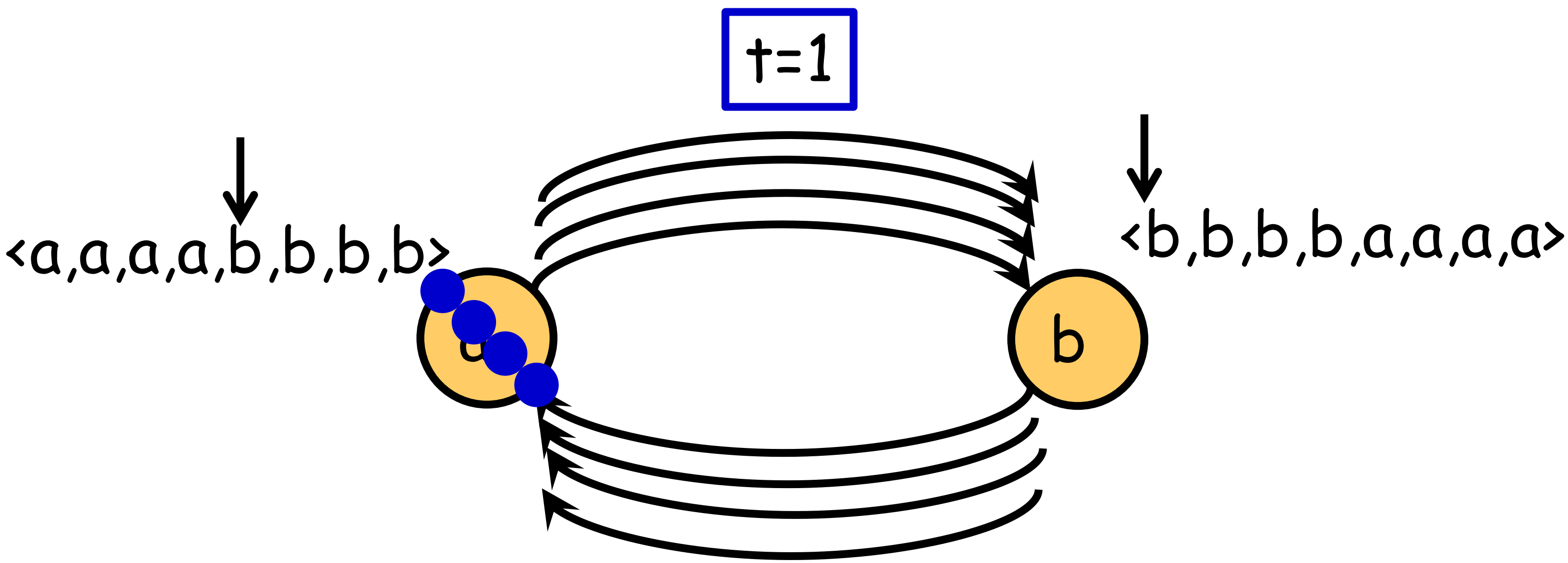
定理 [K, Koga, Makino 10+]

ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),

ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

但し, mは多重グラフの頂点数,枝数.



だまのできる例

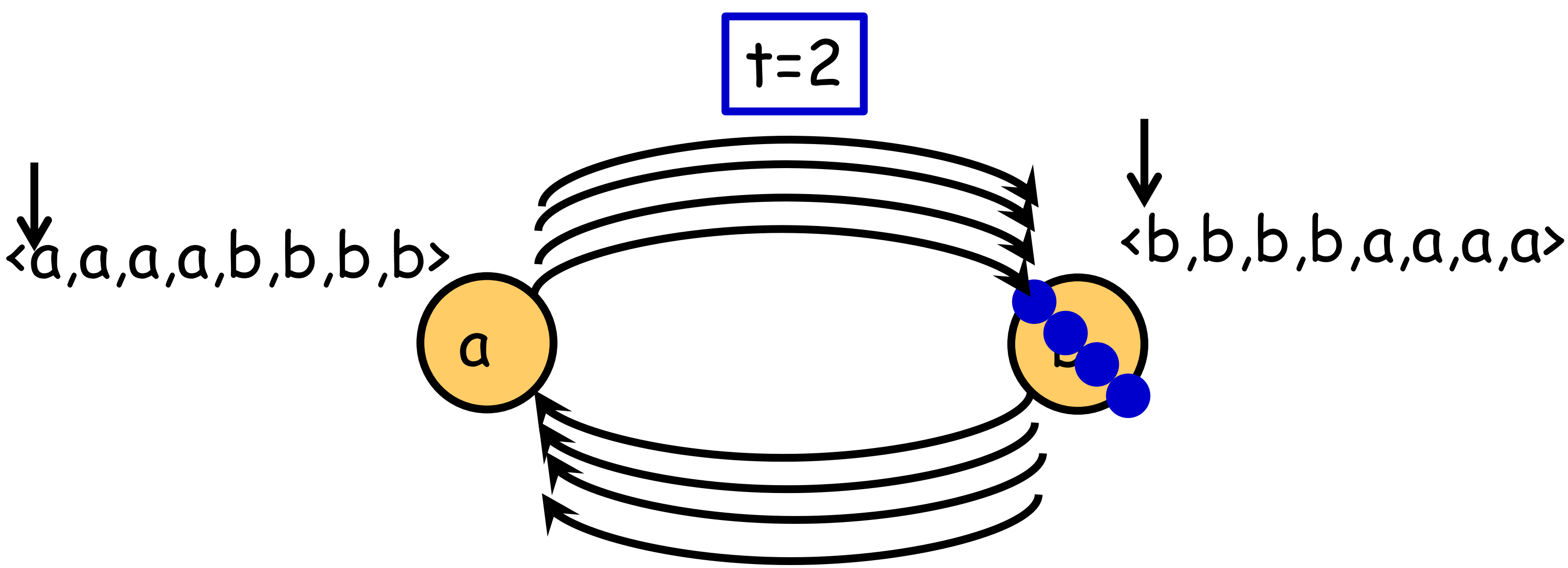
定理 [K, Koga, Makino 10+]

ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),

ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

但し, mは多重グラフの頂点数,枝数.



だまのできる例

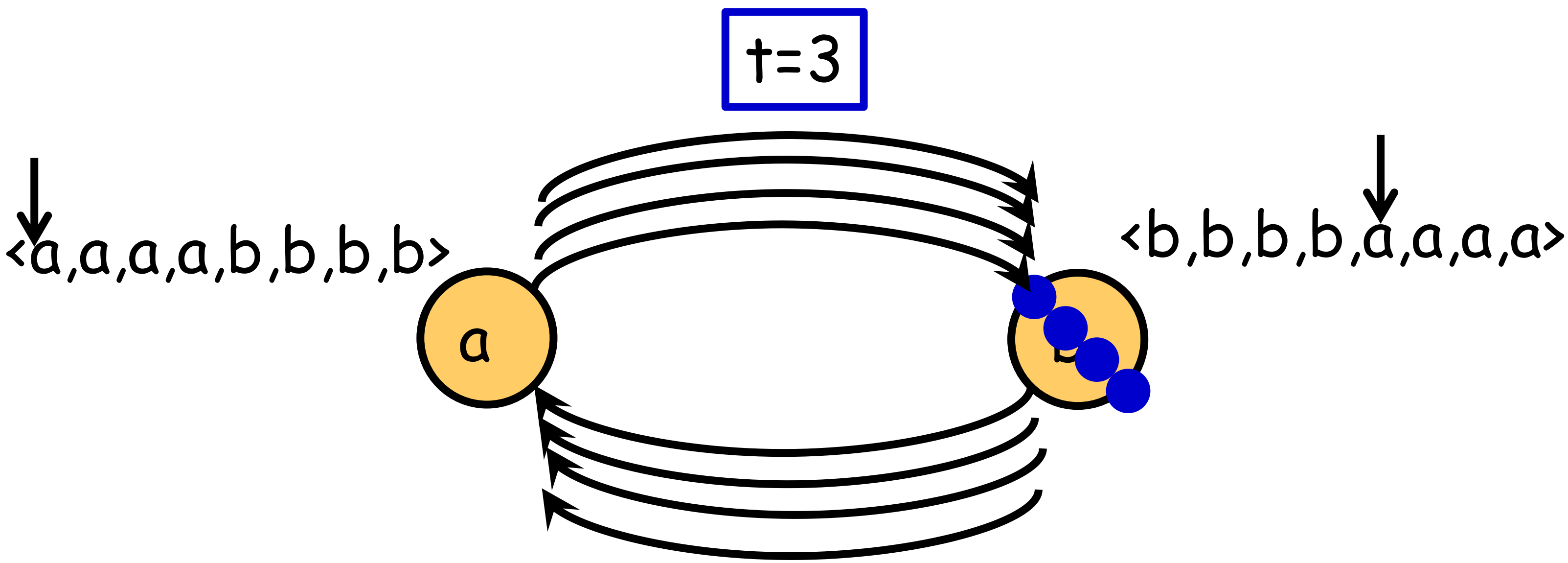
定理 [K, Koga, Makino 10+]

ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),

ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

但し, mは多重グラフの頂点数,枝数.



だまのできる例

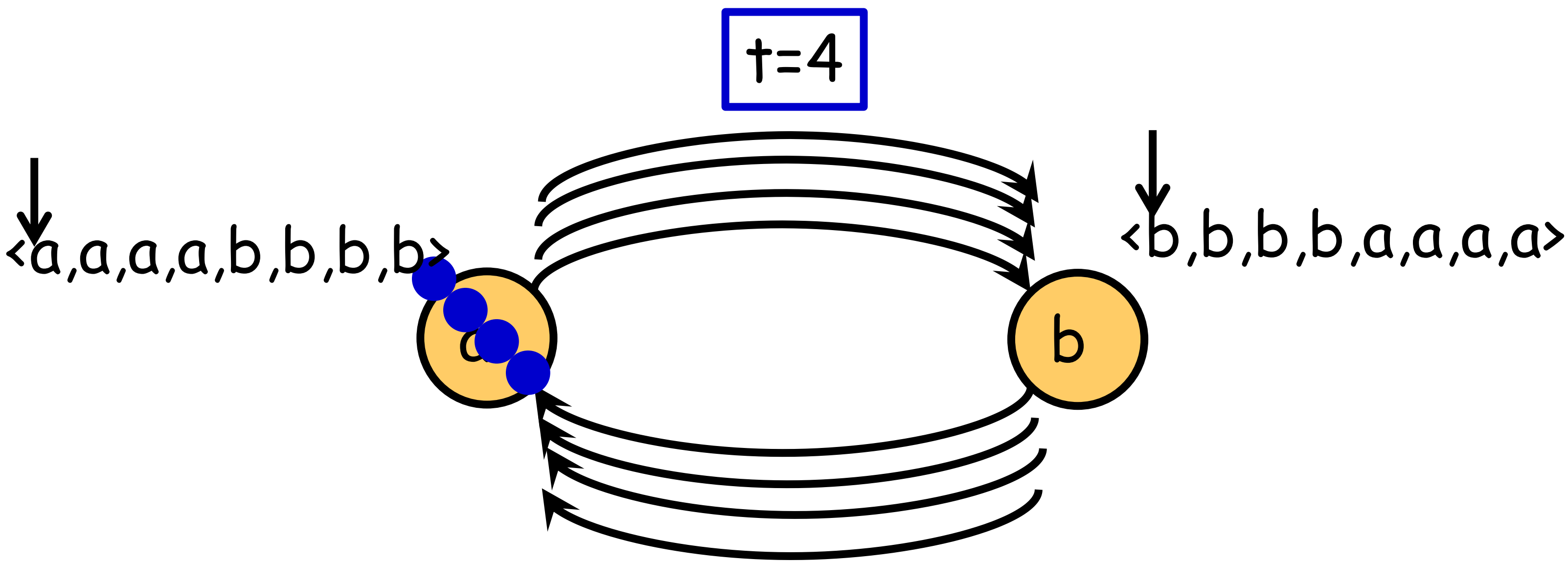
定理 [K, Koga, Makino 10+]

ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),

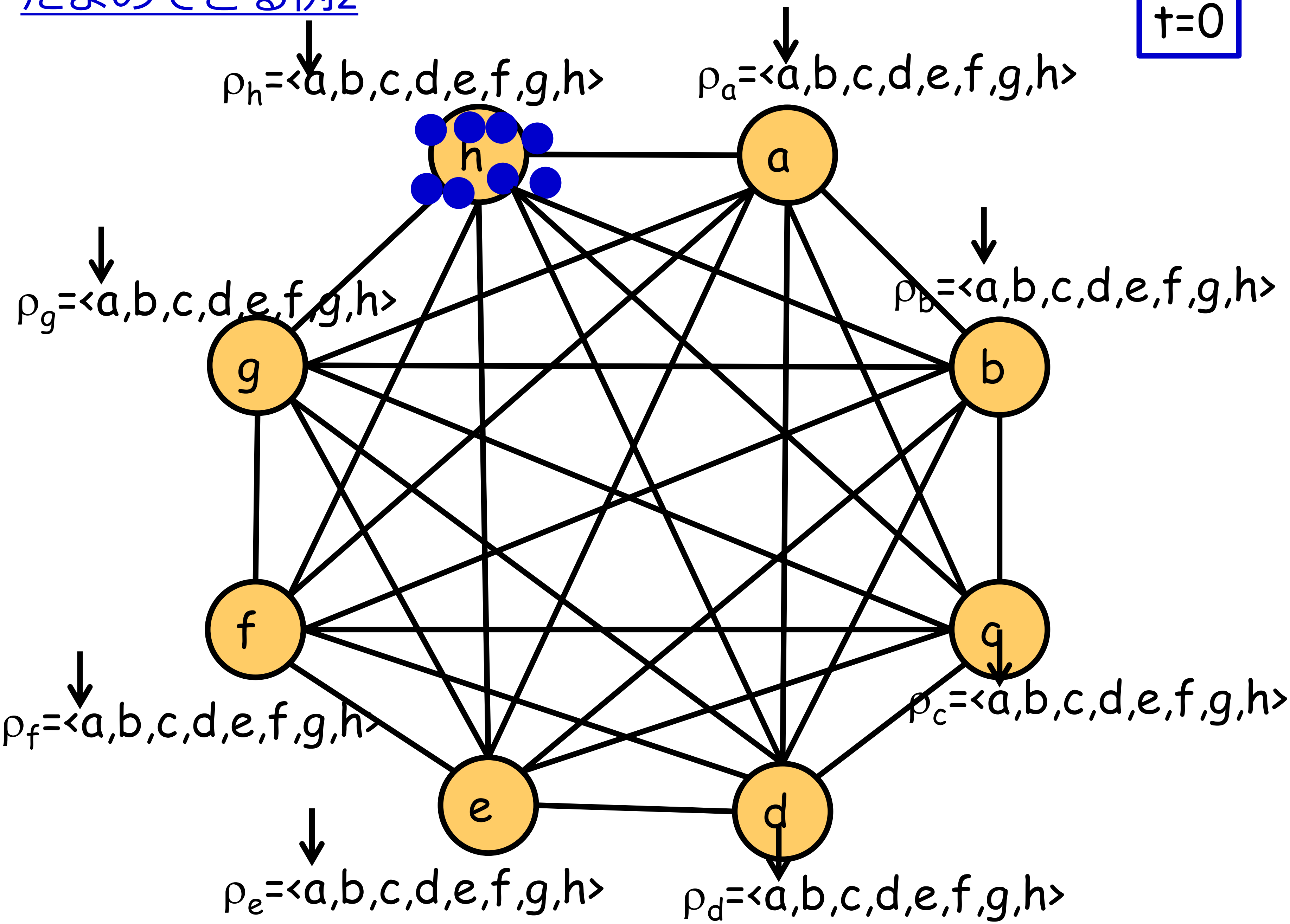
ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

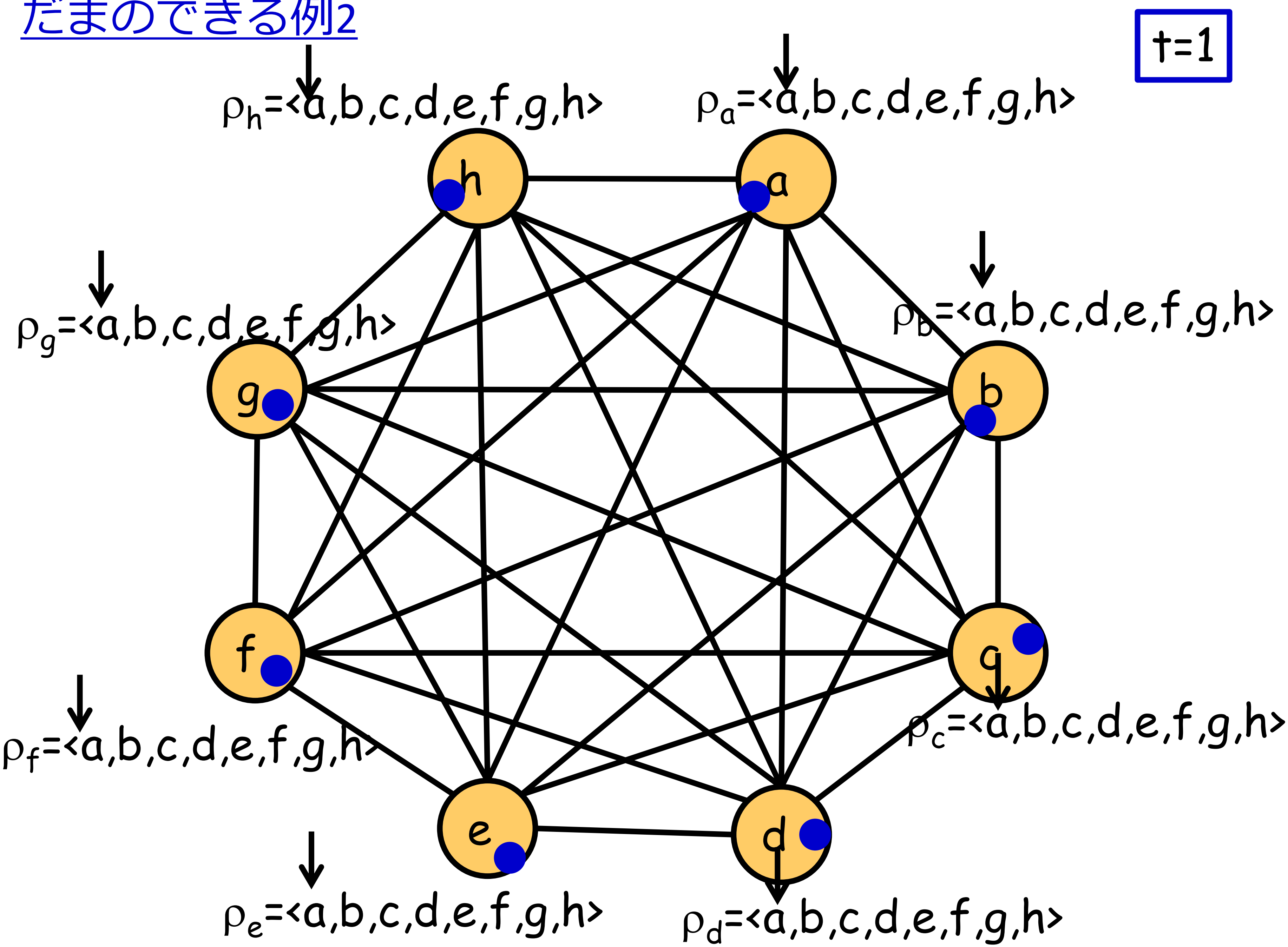
但し, mは多重グラフの頂点数,枝数.



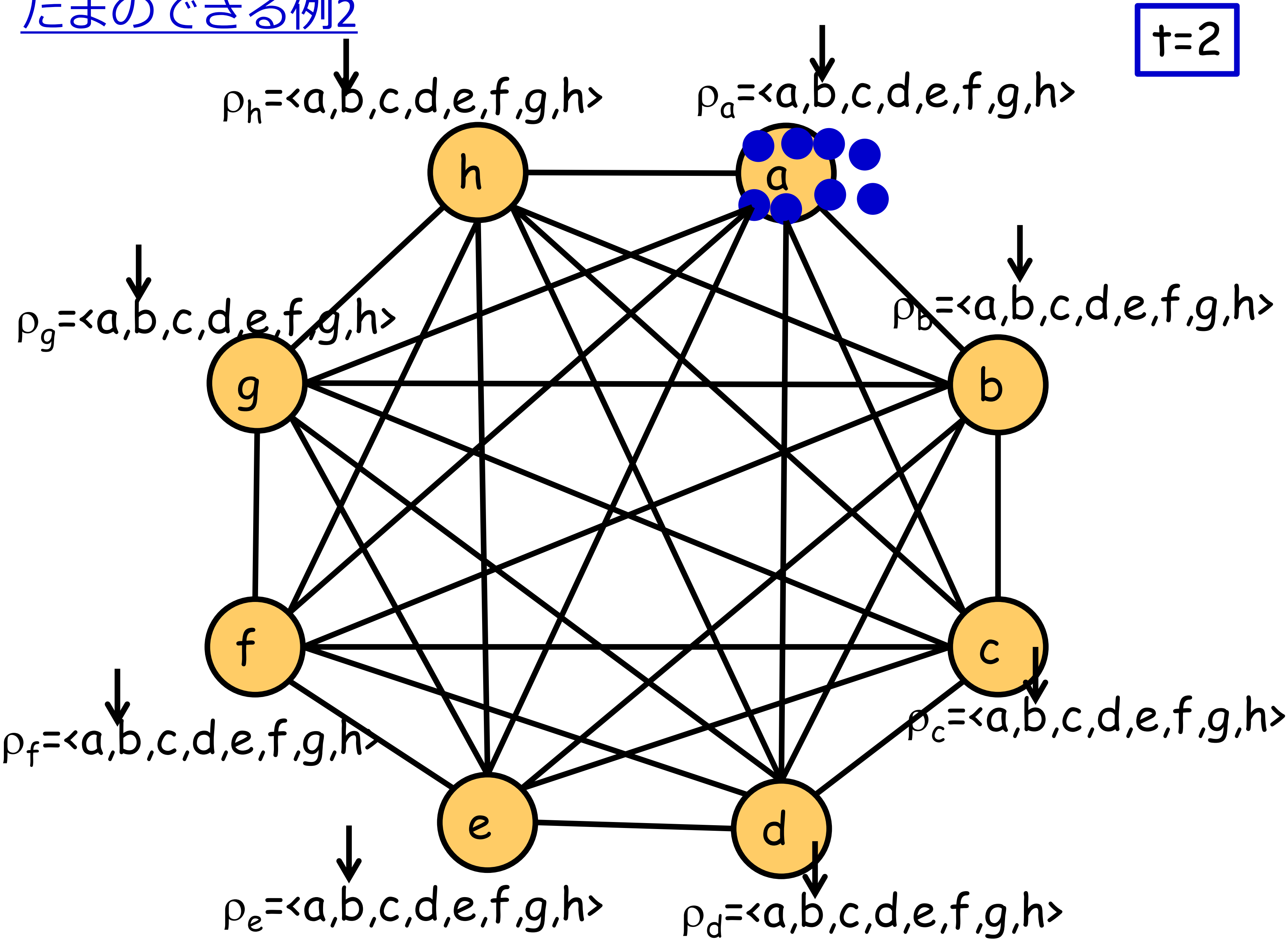
だまのできる例2


t=0
ρ_h=<a,b,c,d,e,f,g,h>
ρ_a=<a,b,c,d,e,f,g,h>
ρ_b=<a,b,c,d,e,f,g,h>
ρ_g=<a,b,c,d,e,f,g,h>
ρ_f=<a,b,c,d,e,f,g,h>
ρ_c=<a,b,c,d,e,f,g,h>
ρ_e=<a,b,c,d,e,f,g,h>
ρ_d=<a,b,c,d,e,f,g,h>
h
a
b
g
f
c
e
d

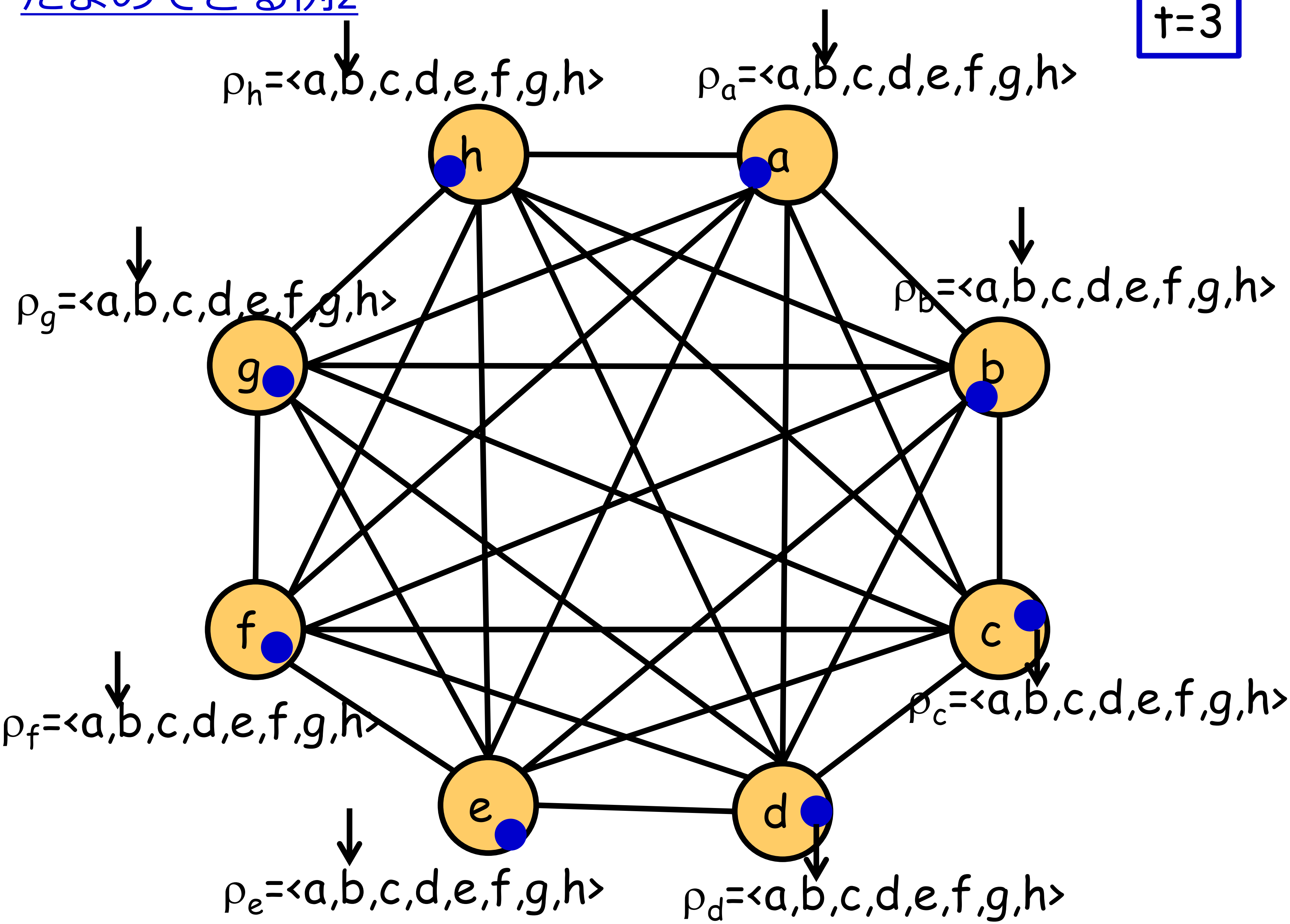
だまのできる例2


t=1
ρh=<a,b,c,d,e,f,g,h>
ρa=<a,b,c,d,e,f,g,h>
ρg=<a,b,c,d,e,f,g,h>
ρb=<a,b,c,d,e,f,g,h>
ρf=<a,b,c,d,e,f,g,h>
ρc=<a,b,c,d,e,f,g,h>
ρe=<a,b,c,d,e,f,g,h>
ρd=<a,b,c,d,e,f,g,h>
h
a
g
b
f
c
e
d

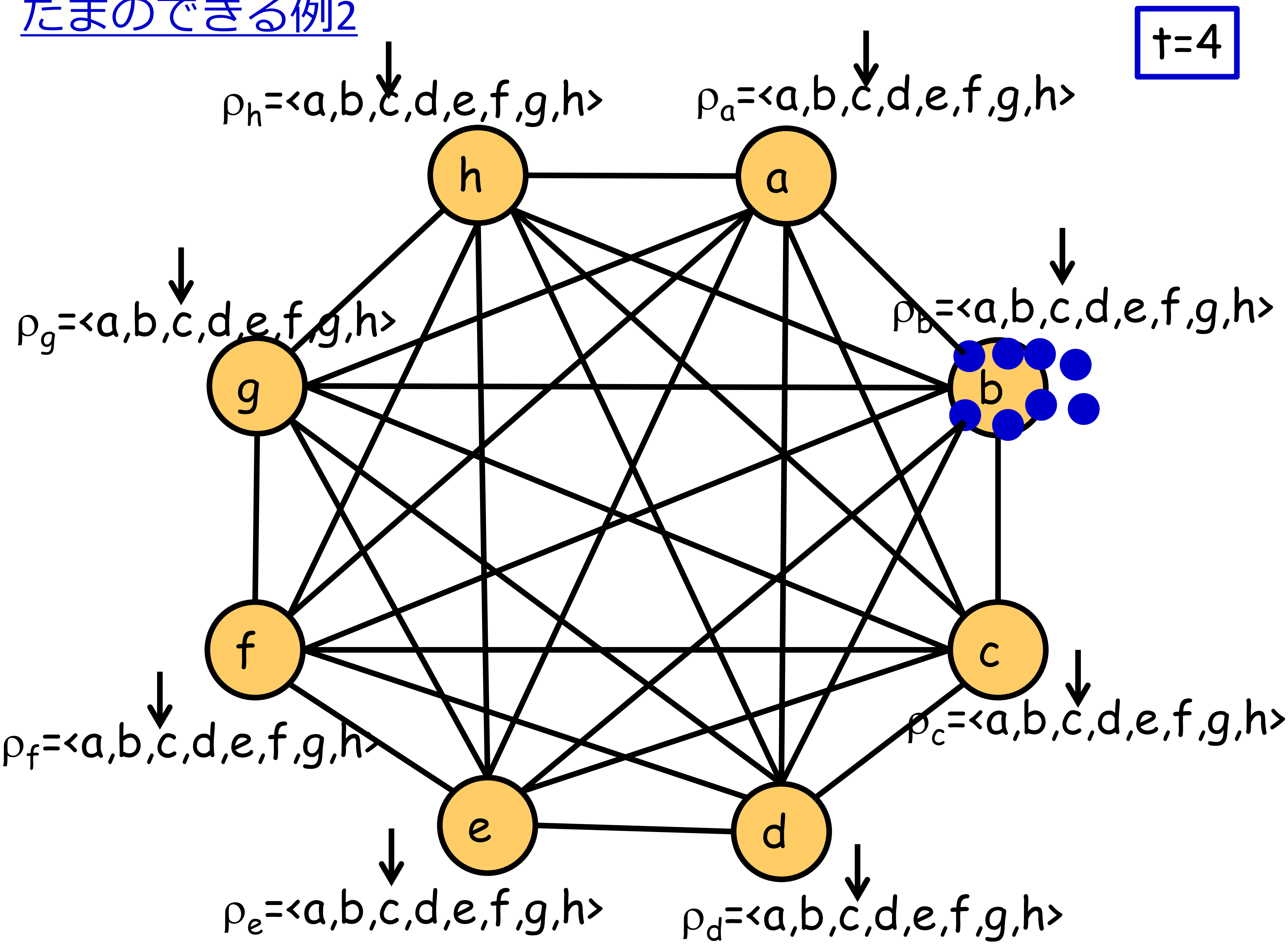
だまのできる例2


t=2
ρ_h=<a,b,c,d,e,f,g,h>
ρ_a=<a,b,c,d,e,f,g,h>
ρ_g=<a,b,c,d,e,f,g,h>
ρ_b=<a,b,c,d,e,f,g,h>
ρ_f=<a,b,c,d,e,f,g,h>
ρ_c=<a,b,c,d,e,f,g,h>
ρ_e=<a,b,c,d,e,f,g,h>
ρ_d=<a,b,c,d,e,f,g,h>
h
a
g
b
f
c
e
d

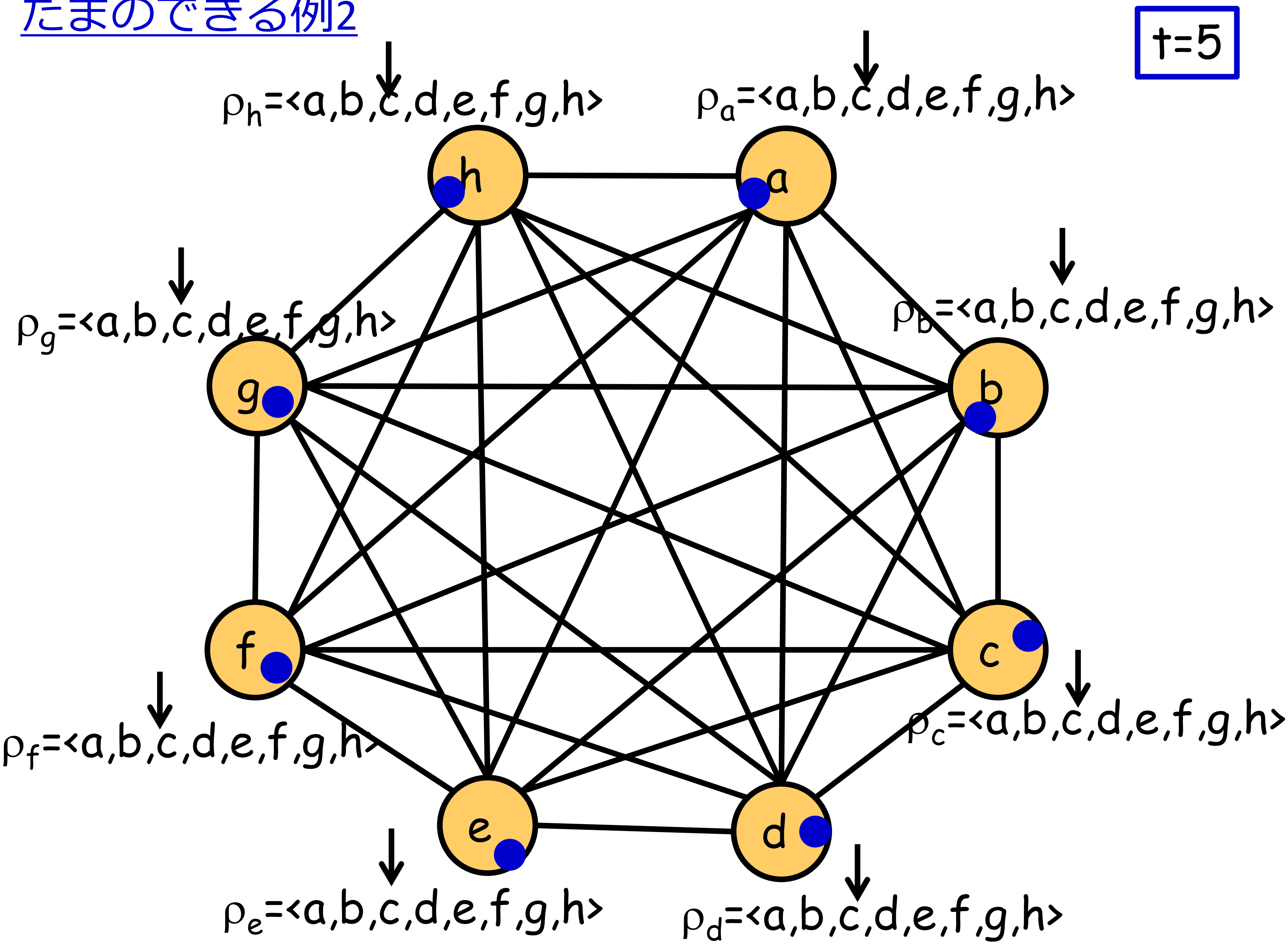
だまのできる例2


t=3
ρ_h=<a,b,c,d,e,f,g,h>
ρ_a=<a,b,c,d,e,f,g,h>
ρ_g=<a,b,c,d,e,f,g,h>
ρ_b=<a,b,c,d,e,f,g,h>
ρ_f=<a,b,c,d,e,f,g,h>
ρ_c=<a,b,c,d,e,f,g,h>
ρ_e=<a,b,c,d,e,f,g,h>
ρ_d=<a,b,c,d,e,f,g,h>
h
a
g
b
f
c
e
d

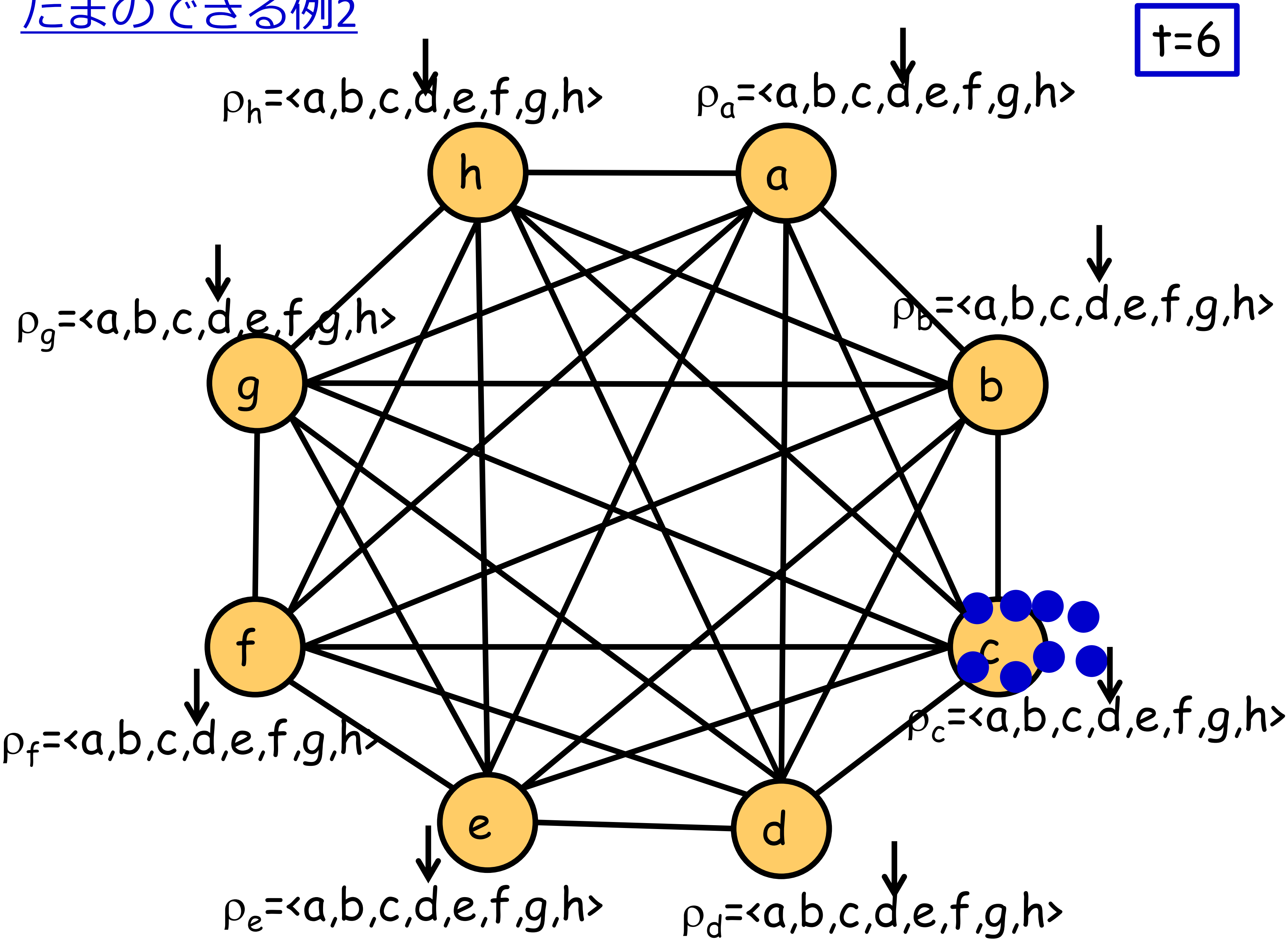
だまのできる例2


t=4
$\rho_h=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_a=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_g=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_b=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_f=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_c=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_e=\langle a,b,c,d,e,f,g,h\rangle$
$\rho_d=\langle a,b,c,d,e,f,g,h\rangle$
h
a
g
b
f
c
e
d

だまのできる例2


t=5
ρh=<a,b,c,d,e,f,g,h>
ρa=<a,b,c,d,e,f,g,h>
ρg=<a,b,c,d,e,f,g,h>
ρb=<a,b,c,d,e,f,g,h>
ρf=<a,b,c,d,e,f,g,h>
ρc=<a,b,c,d,e,f,g,h>
ρe=<a,b,c,d,e,f,g,h>
ρd=<a,b,c,d,e,f,g,h>
h
a
g
b
f
c
e
d

だまのできる例2


t=6
ρh=<a,b,c,d,e,f,g,h>
ρa=<a,b,c,d,e,f,g,h>
ρg=<a,b,c,d,e,f,g,h>
ρb=<a,b,c,d,e,f,g,h>
ρf=<a,b,c,d,e,f,g,h>
ρc=<a,b,c,d,e,f,g,h>
ρe=<a,b,c,d,e,f,g,h>
ρd=<a,b,c,d,e,f,g,h>
h
a
g
b
f
c
e
d

誤差の下界

定理 [K, Koga, Makino 10+]
ある多重有向グラフ$\mathcal{G}$=(V,$\mathcal{E}$),
ある初期状態, あるrotor-routerが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \geq \Omega(m)$$

但し, mは多重グラフの頂点数,枝数.

主結果 2

定理 [K, Koga, Makino 10+]

対応する推移確率行列Pの固有値がすべて非負ならば,
任意の多重有向グラフ, 任意の初期状態, 任意のrotor-router,
任意の頂点w, 任意の時刻tについて,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \leq (2m-n)\left(\max_{i\in\{1,\ldots,\kappa\}} n_i + n + 3\right) \;\leq\; 4mn + \mathrm{O}(m)$$

但し, n,mは多重グラフの頂点数,枝数. n_iはJordan cellのサイズ.

Remark

「推移確率行列Pの固有値がすべて非負」という条件.

- reversible lazy Markov chainはこの条件を満たす.
  - ✓ MCMC法で使われるマルコフ連鎖
- +Pが対称 ⇒ Pは半正定値行列

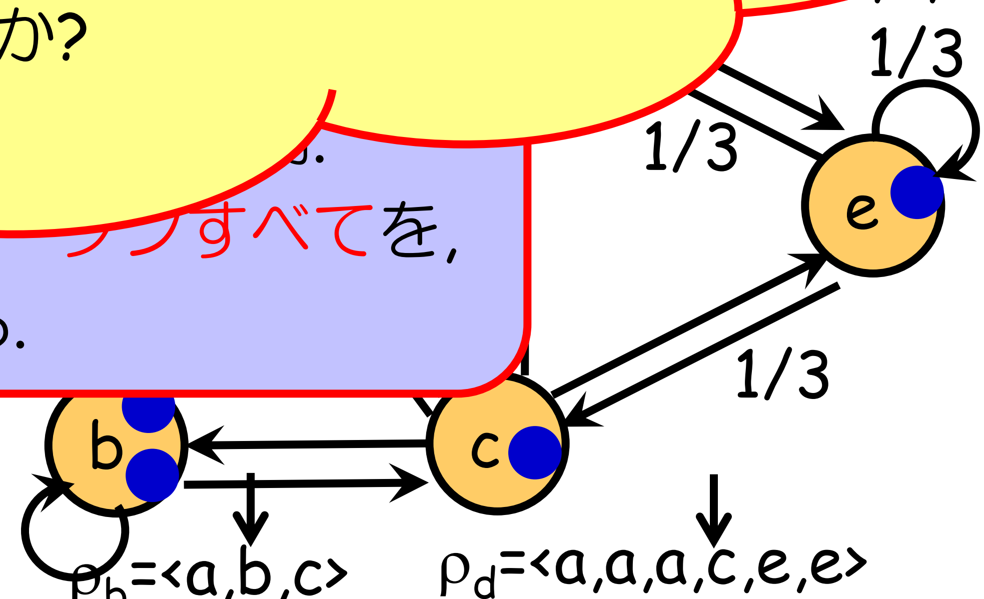
Propp機械はランダムウォークを模倣できるか?

N個のトークンがグラフ上を移動する.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)

- ✓ (エルゴード)マルコフ連鎖は定常分布に収束する
  - ➢ "平滑化"される.
- ✓ Propp機械はランダムウォークを模倣できるか?
  - ➢ "平滑化"できるか?
  - ➢ "だま"はできないのか?

- ✓ 各ノードは時刻tに居るトークンすべてを, 時刻t+1までに移動させる.



チップ数に依存しない.

先行研究と本研究の成果

<u>定理 [Cooper & Spencer 2006]</u>

Z^d上で, 任意の初期状態, 任意のrotor-router,
任意の頂点w, 任意の時刻tについて,
(dのみに依存する)定数C_dが存在して,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \leq C_d$$

単一頂点誤差に関する研究(1/2)

2006	Cooper, Spencer	Z^d上のロータールーター ➢ 誤差 $\le C_d$
2007	Cooper, Doerr, Spencer, Tardos	Z^1上のロータールーター ➢ $C_1 \le 2.29$
2008	Cooper, Doerr, Friedrich, Spencer	無限のk正則木上のロータールーター ➢ 誤差 $> \Omega(\sqrt{kT})$ at time T
2009	Doerr, Friedrich	Z^2上のロータールーター ➢ $C_2 \le 7.83$ (上右下左) ➢ $C_2 \le 7.29$ (上下左右)
2012	Kijima, Koga, Makino	有限多重有向グラフ G上のロータールーター ➢ 誤差$\le$ 4mn+O(m) $\{0,1\}^d$上のPropp機械 ➢ 誤差$\le$ O(d^3) (頂点数のpoly log)
2012+	Kajino et al.	(後述)
2012+	Shiraga et al.	(後述)

単一頂点誤差に関する研究(2/2)

2012 (2010)	Kijima, Koga, Makino	有限多重有向グラフ G上のロータールーター ➢ 誤差 ≤ 4m*n+O(m*) (P: 有理数 + 既約 + 非周期 + 可逆 + lazy) $\{0,1\}^d$上のPropp機械 ➢ 誤差 ≤ O(d^3) (頂点数のpoly log)
2012+ (2011)	Kajino, Kijima, Makino	有限多重有向グラフ G上のロータールーター ➢ 誤差 $\leq \mathrm{O}\left(\alpha\ \frac{m^* n^2}{1-\lambda}\right)$ (P: 有理数 + 既約) $\{0,1\}^d$上のPropp機械 ➢ 誤差 ≤ O(d^2) (頂点数のpoly log)
2012+	Shiraga, Yamauchi, Kijima, Yamashita	有限有向グラフ G上の関数ルーター ➢ 誤差 $\leq \mathrm{O}\left(\sqrt{\frac{\pi_{\max}}{\pi_{\min}} \frac{mn}{1-\lambda}} \log M\right)$ (P: 実数 + 既約 + 非周期 + 可逆)

K., Koga, Makino

定理 [K, Koga, Makino 10+]

対応する推移確率行列Pの固有値がすべて非負ならば,
任意の多重有向グラフ, 任意の初期状態, 任意のrotor-router,
任意の頂点w, 任意の時刻tについて,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \le (2m-n)\left(\max_{i\in\{1,\ldots,\kappa\}} n_i + n + 3\right) \;\le\; 4mn + \mathrm{O}(m)$$

但し, n,mは多重グラフの頂点数,枝数. n_iはJordan cellのサイズ.

Remark

「推移確率行列Pの固有値がすべて非負」という条件.

- reversible lazy Markov chainはこの条件を満たす.
  - ✓ MCMC法で使われるマルコフ連鎖
- +Pが対称 ⇒ Pは半正定値行列

K., Koga, Makino

定理 [K, Koga, Makino 10+]
$\{0,1\}^d$超立方体の稜線グラフに対して,
任意の初期状態, 任意のrotor-router,
任意の頂点w, 任意の時刻tについて,

$$|\chi_w^{(T)} - \mu_w^{(T)}| \le \frac{3}{2}d^3 + \mathrm{O}(d^2)$$

定理 [K, Koga, Makino 10+]
JohnsonグラフJ(d,c)に対して,
任意の初期状態, 任意のrotor-router,
任意の頂点w, 任意の時刻tについて,

$$|\chi_w^{(T)} - \mu_w^{(T)}| \le 2c^3 \cdot (d-c)^2 + \mathrm{O}(c^3 \cdot (d-c))$$

JohnsonグラフJ(d,c) $=(V_J,E_J)$
$V_J=\{S \subset \{1,\dots,d\} \mid |S|=c\}$,
$E_J=\{ \{S,T\} \in V_J^2 \mid |S \oplus T|=2 \}$

頂点数に対して
対数多項式上界

一般グラフに対する定理の証明

$$X_v^{(t)} := \sum_{s=0}^{t} \chi_v^{(s)}$$

$$s_v(i) := \min\left\{t \geq 0 \mid i < \sum X_v^{(t)}\right\},$$

Cooper & Spencer 2006の手法を

推移確率行列Pで理解

補題 1

$$\chi_w^{(T)} - \mu_w^{(T)} = \sum_{v \in V} \sum_{i=0}^{X_v^{(T-1)}-1} \left(P^{T-s_v(i)-1}(\rho_v(i), w) - P^{T-s_v(i)}(v, w)\right).$$

「時刻t以前Propp機械, 時刻t以後RW」過程. (初期配置はχ^0(と同一)とする.)

$\zeta(w;t,T)$: 時刻Tの期待トークン配置.

Remark

✓ $\zeta(w;T,T) = \chi^T$

✓ $\zeta(w;0,T) = \mu^T$

$$\chi_w^{(T)} - \mu_w^{(T)} = \sum_{t=0}^{T-1} (\zeta(w; t+1, T) - \zeta(w; t, T)).$$

$$\zeta(w; t+1, T) - \zeta(w; t, T) = \sum_{v \in V} \sum_{i=X_v^{(t-1)}}^{X_v^{(t)}-1} \left(P^{T-t-1}(\rho_v(i), w) - P^{T-t}(v, w)\right)$$

∎

関連研究

- ✓ IDLA (Internal Diffusion-Limited Aggregation)
  - Levine & Peres 2005
- ✓ Information Spreading
  - Doerr, Friedrich, & Sauerwald 2008
  - Doerr, Friedrich, Kunnemann, & Sauerwald 2009
- ✓ Hitting time, Cover time
  - Friedrich & Sauerwald 2010
  - Holroyd & Propp 2010+

cf. [K, Koga, Makino 10+]

$|\chi^t_v/N - \pi^t_v| \leq O(mn/N)$

定理 [Holroyd & Propp 2010+]

単一トークンのPropp機械を考える.

F^t_v := 時刻0からtまでに頂点vを訪れた回数.

任意の有限グラフについて,

$|F^t_v/t - \pi^*_v| \leq O(mn/t)$

ただしπ*は対応するマルコフ連鎖の定常分布.

今後の課題

- ✓ 上下界の一致. ($O(mn)$, $\Omega(m)$)
- ✓ 組合せ構造に由来するグラフに対するpolylogの上界.
- ✓ Blanket time vs Mixing time.
- ✓ MCMC法の脱乱択化.
- ✓ 乱数とは？
  - ➢ 乱択アルゴリズムにおける「乱数」の持つべき性質は？
    - ◆ 準モンテカルロ (quasi Monte Carlo)
    - ◆ カオス系列 (Chaos time series)

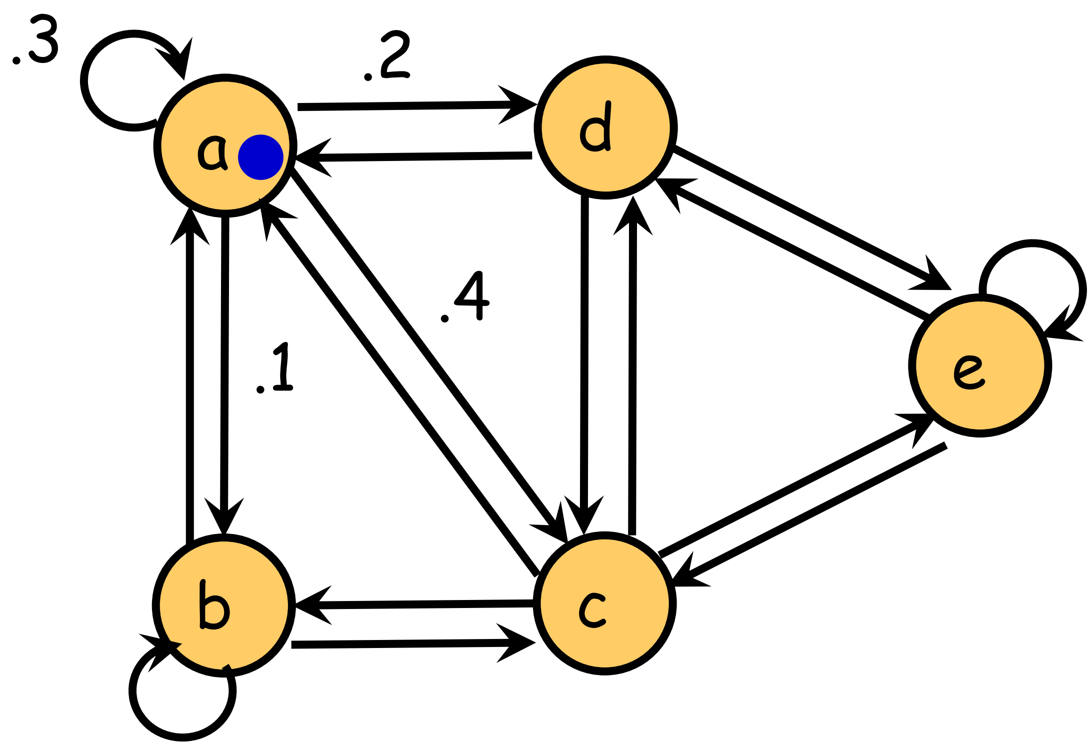
3. 関数ルーターモデル

無理数の遷移確率行列を実現

ランダムウォーク

トークンがグラフ上をランダムウォークする.

- ✓ π^0: 初期分布 (トークンは確率$\pi^0{}_v$で頂点vに居る)
- ✓ P: 推移確率行列 (確率P_{uv}で頂点u頂点vに移動)
- ✓ 時刻tの確率分布$\pi^t:=\pi^0P^t$ (頂点vに居る確率 $(\pi^t)_v$)



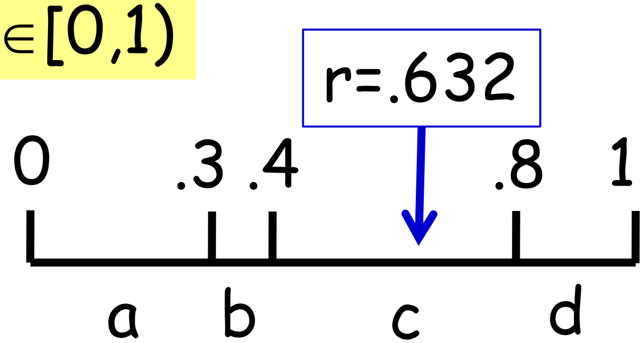
ランダムウォーク

トークンがグラフ上をランダムウォークする.

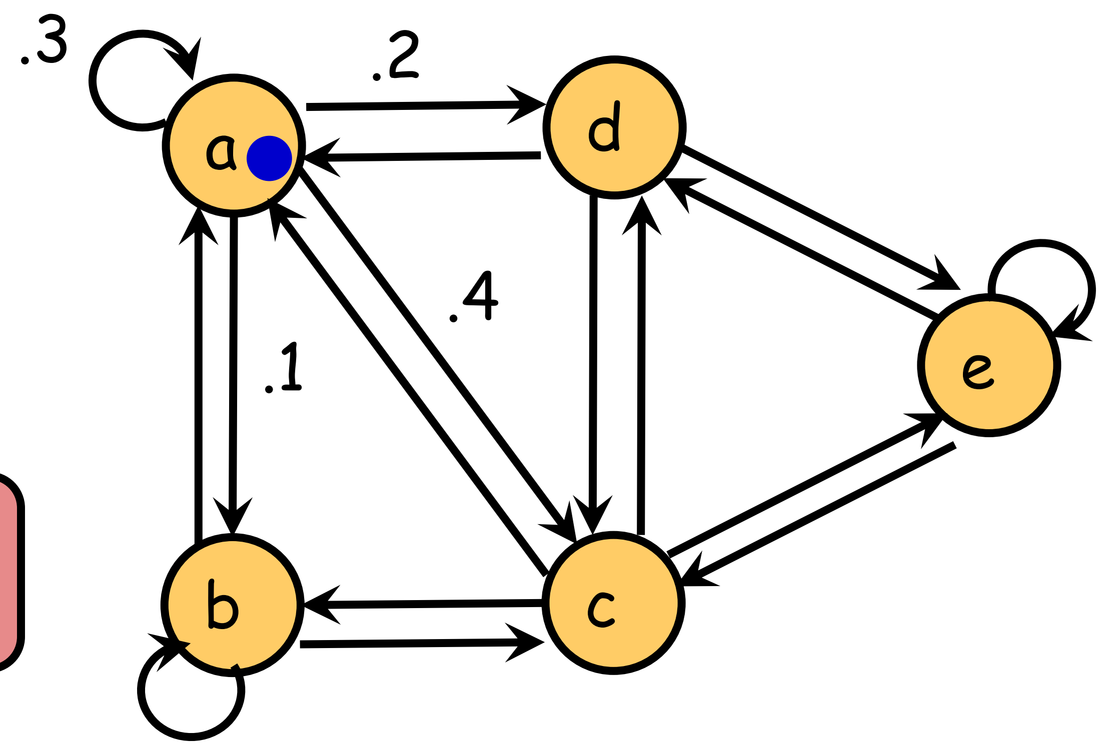
- ✓ π^0: 初期分布 (トークンは確率π^0_vで頂点vに居る)
- ✓ P: 推移確率行列 (確率P_{uv}で頂点u頂点vに移動)
- ✓ 時刻tの確率分布$\pi^t{:=}\pi^0P^t$ (頂点vに居る確率 $(\pi^t)_v$)

ランダムウォークを
乱数を使って実現

$r\in[0,1)$



乱数の代わりに
超一様分布列を使う.



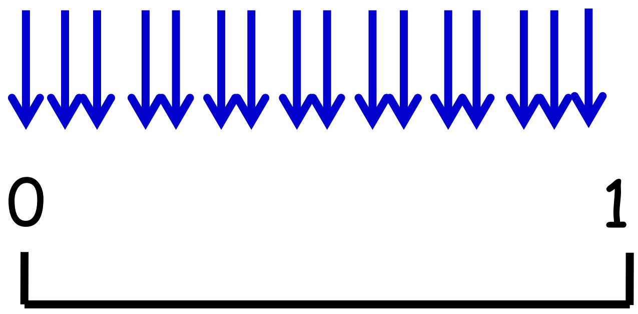
Van der Corput列

自然数 $i = \sum_{j=0}^{\lfloor \lg i \rfloor} \beta_j(i) 2^j$

ただし $\beta_j(i) \in \{0,1\}$ $(j = 0,1,\ldots,\lfloor \lg i \rfloor)$

$$\psi(i) \coloneqq \sum_{j=0}^{\lfloor \lg i \rfloor} \beta_j(i) 2^{-(j+1)}$$

i	$(i)_2$	$(\psi(i))_2$	$\psi(i)$
0	0	0	0
1	1	0.1	1/2
2	10	0.01	1/4
3	11	0.11	3/4
4	100	0.001	1/8
5	101	0.101	5/8
6	110	0.011	3/8
…	…	…	…

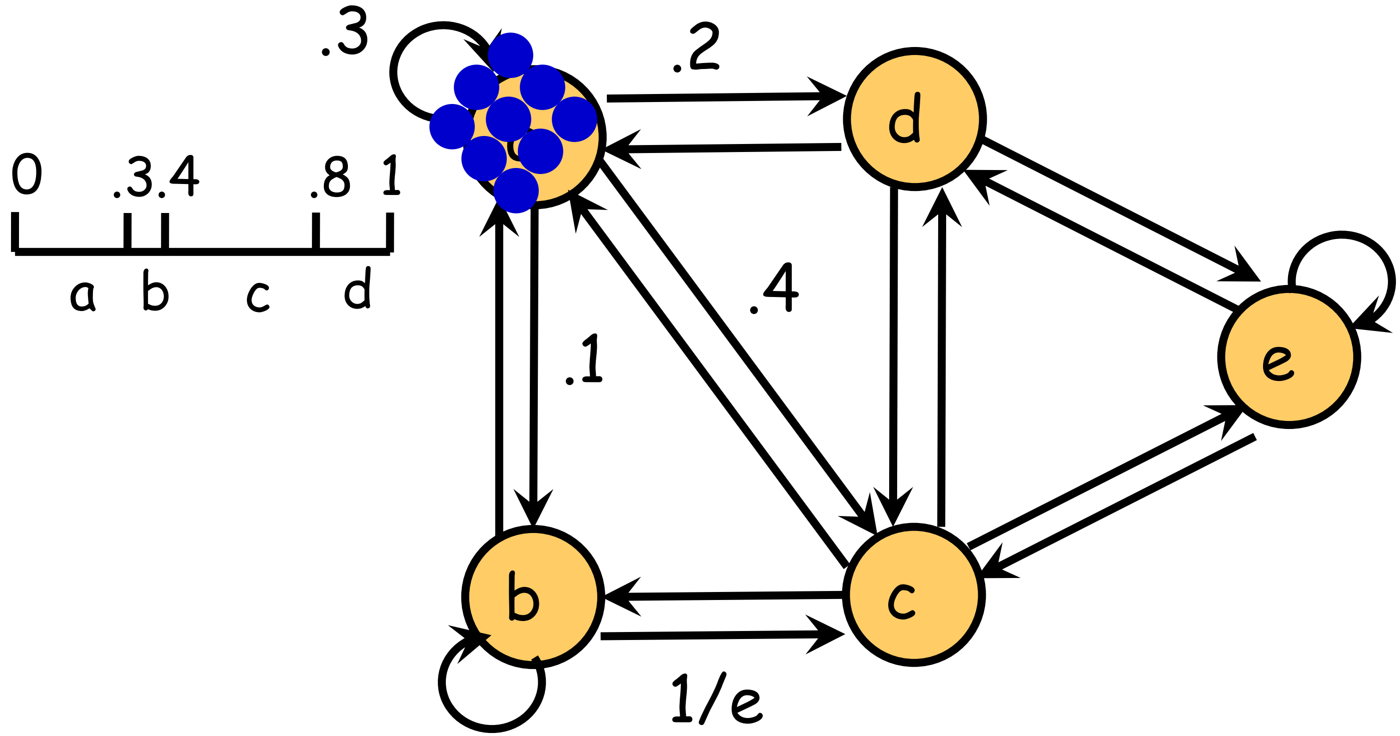


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$\boxed{t=0}$

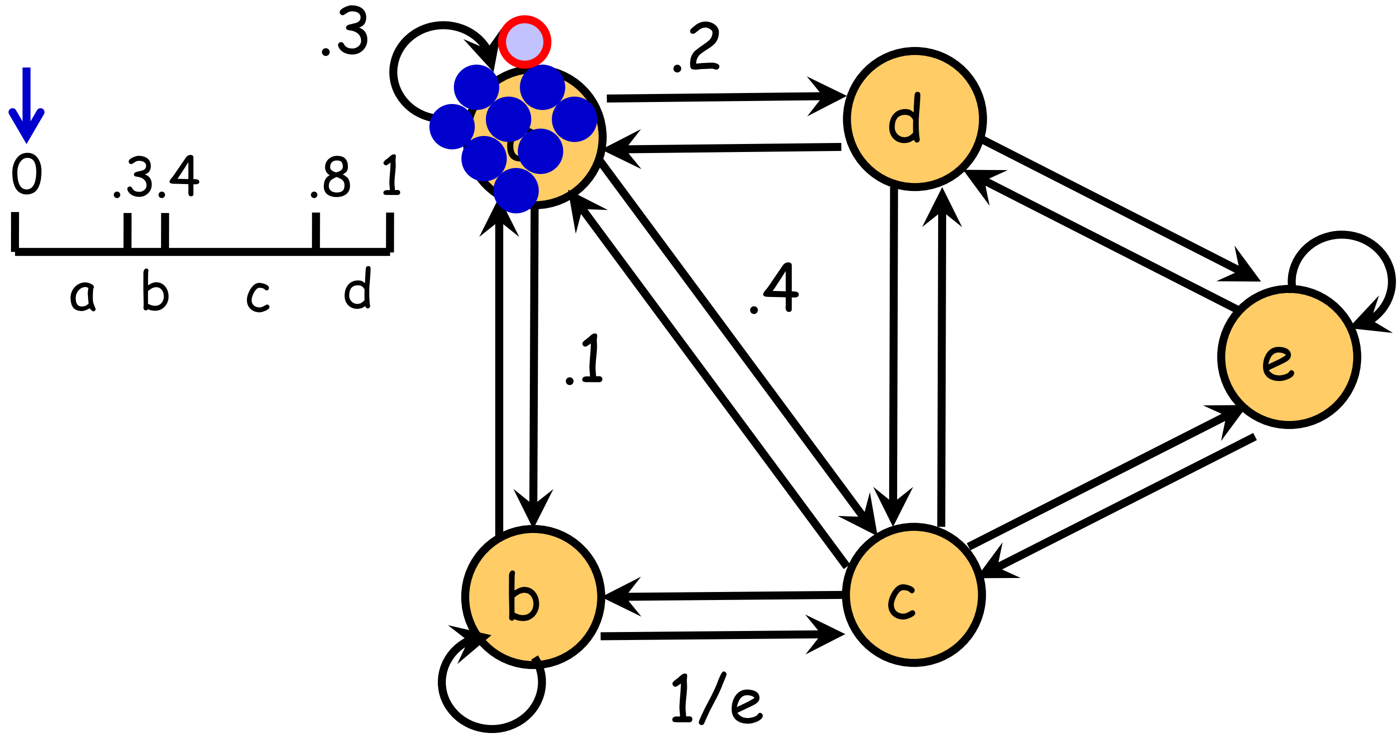


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

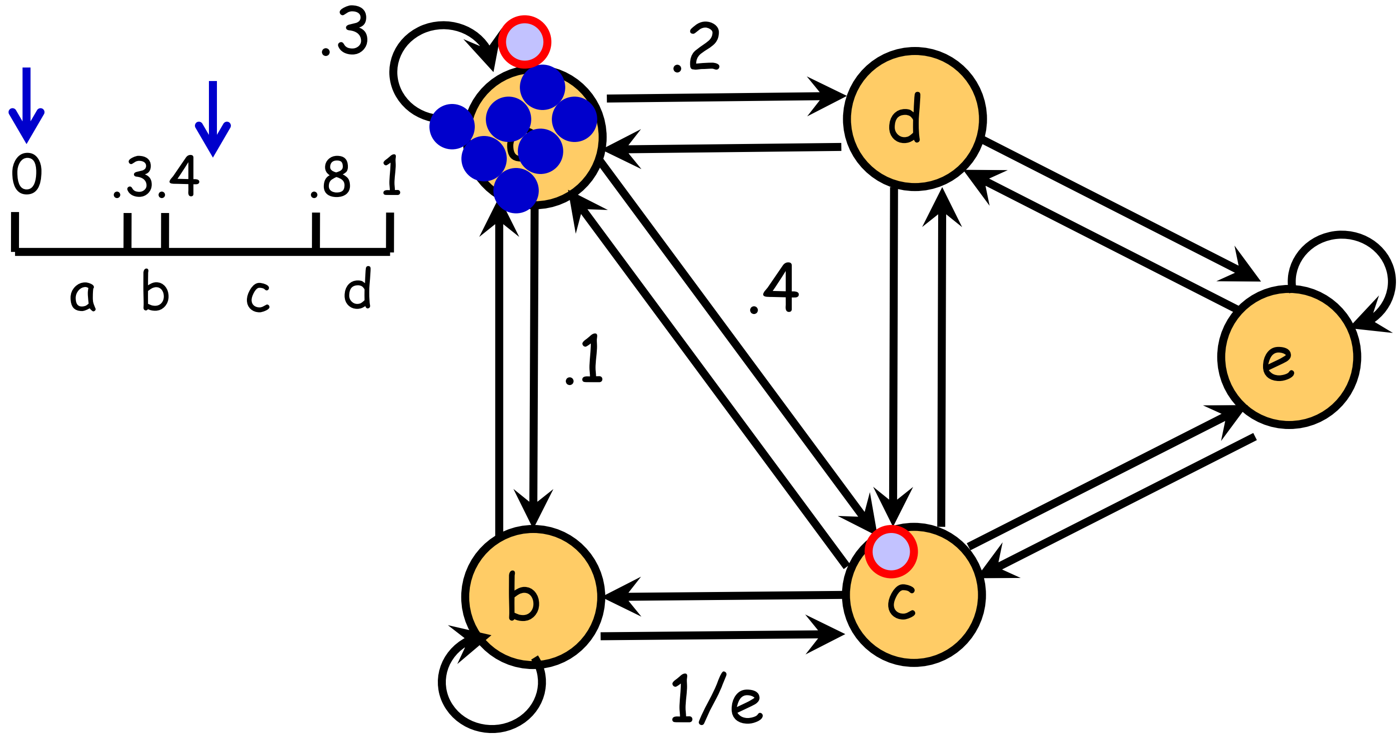


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

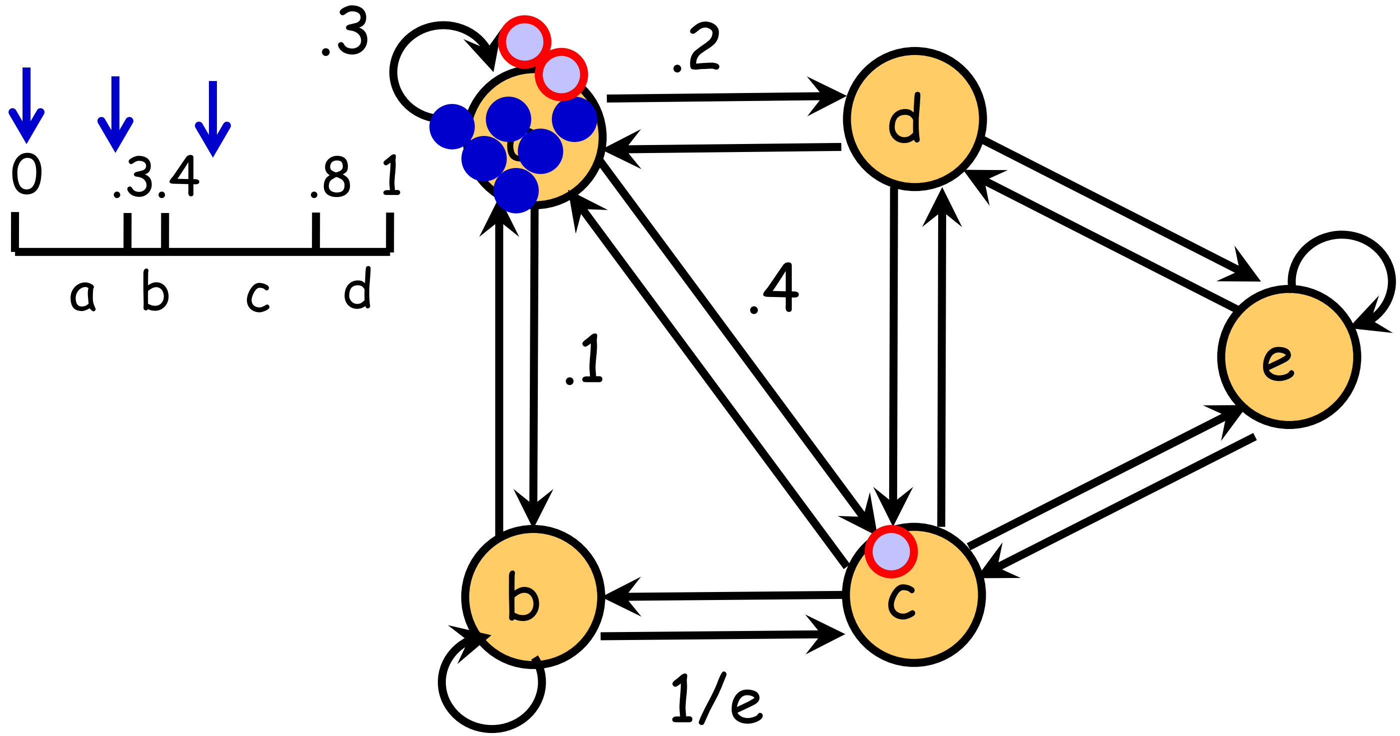


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "**関数ルーター**" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

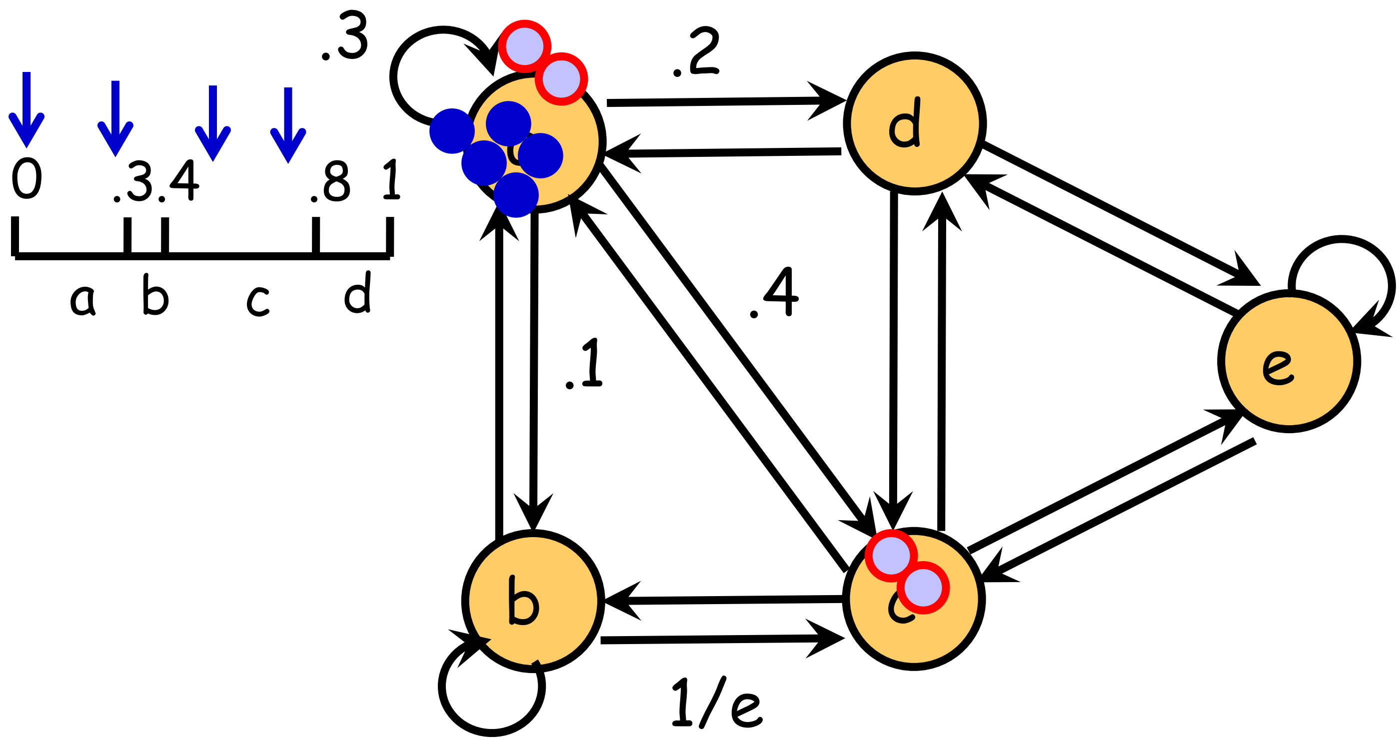


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

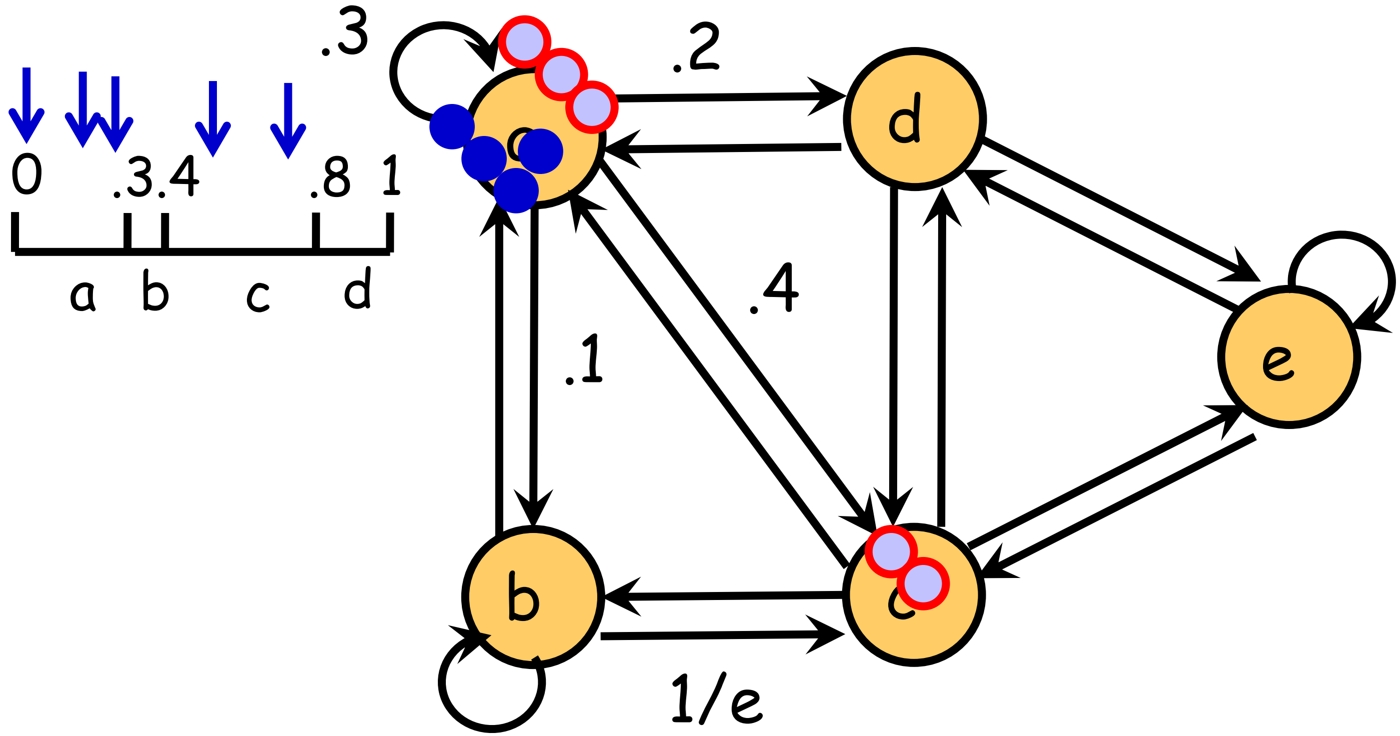


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

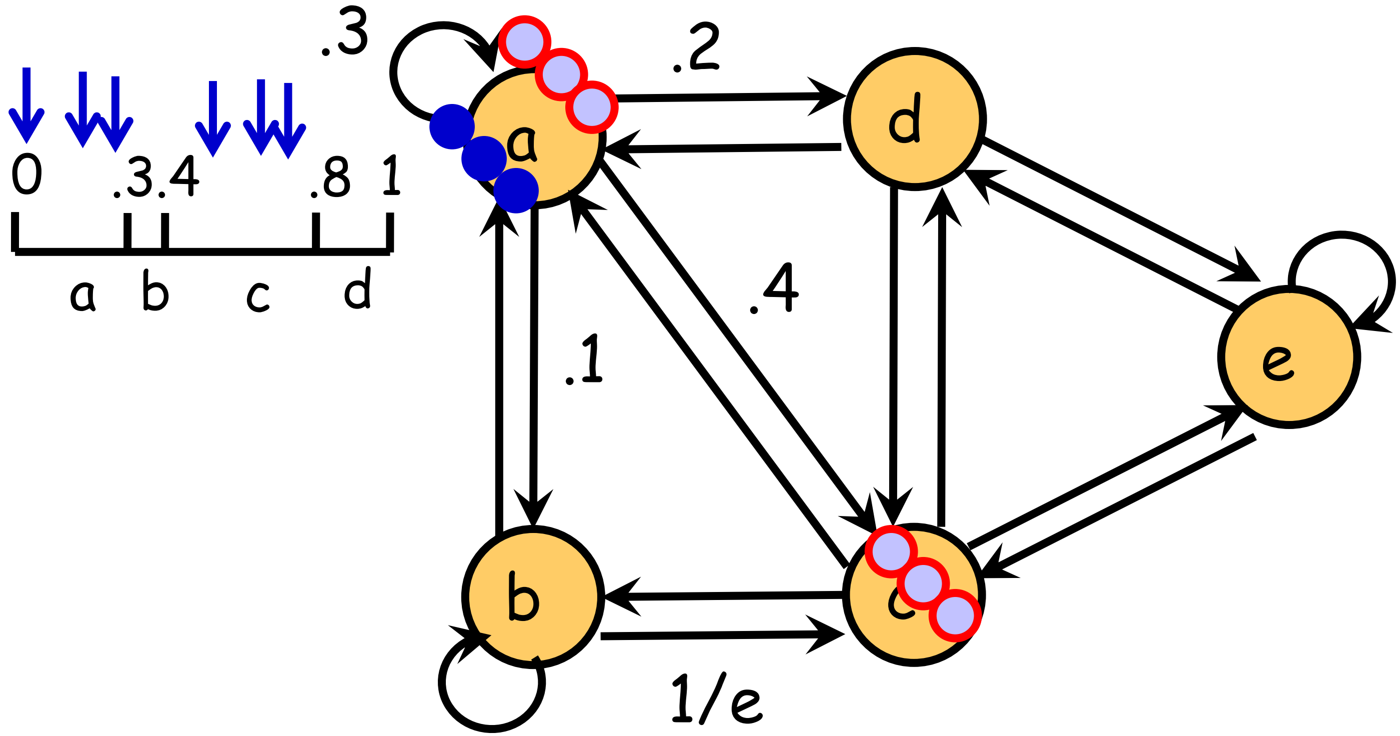


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

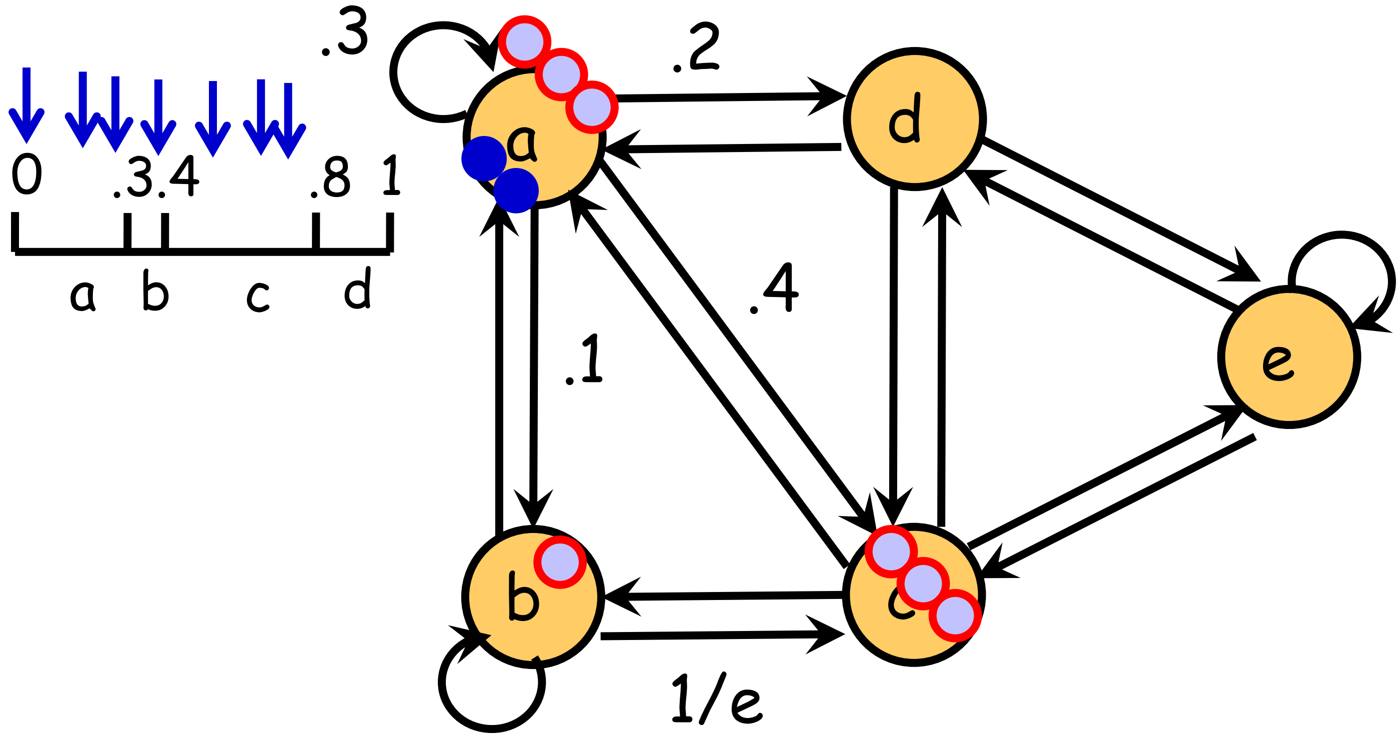


関数ルーターモデル

Nトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

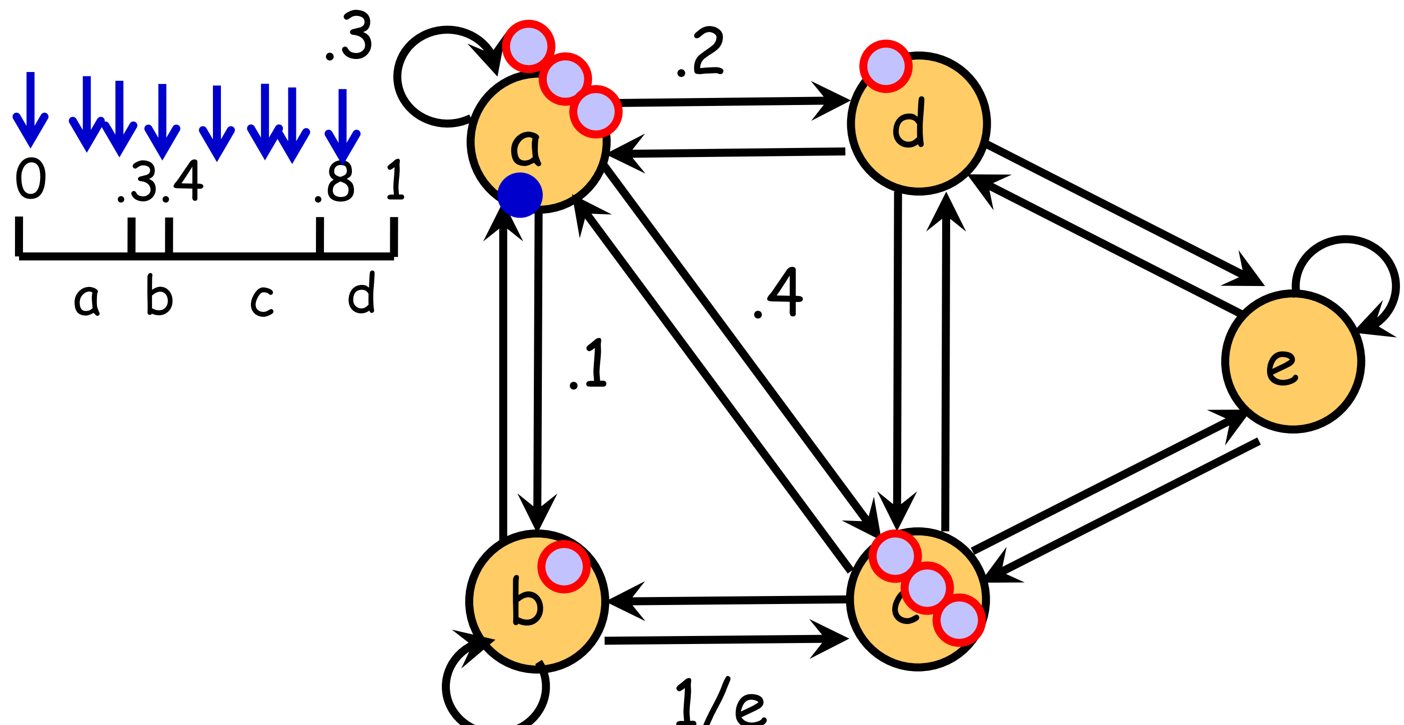


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

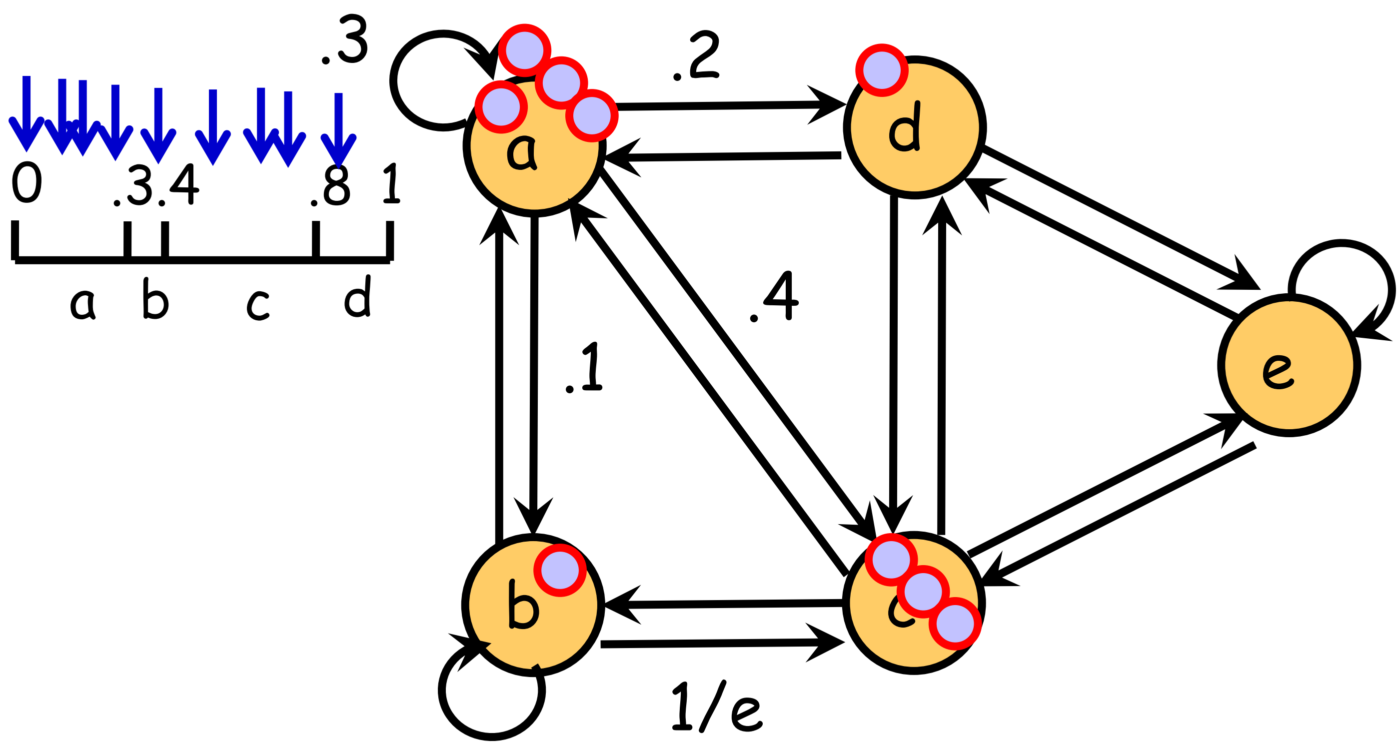


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (0,1)$

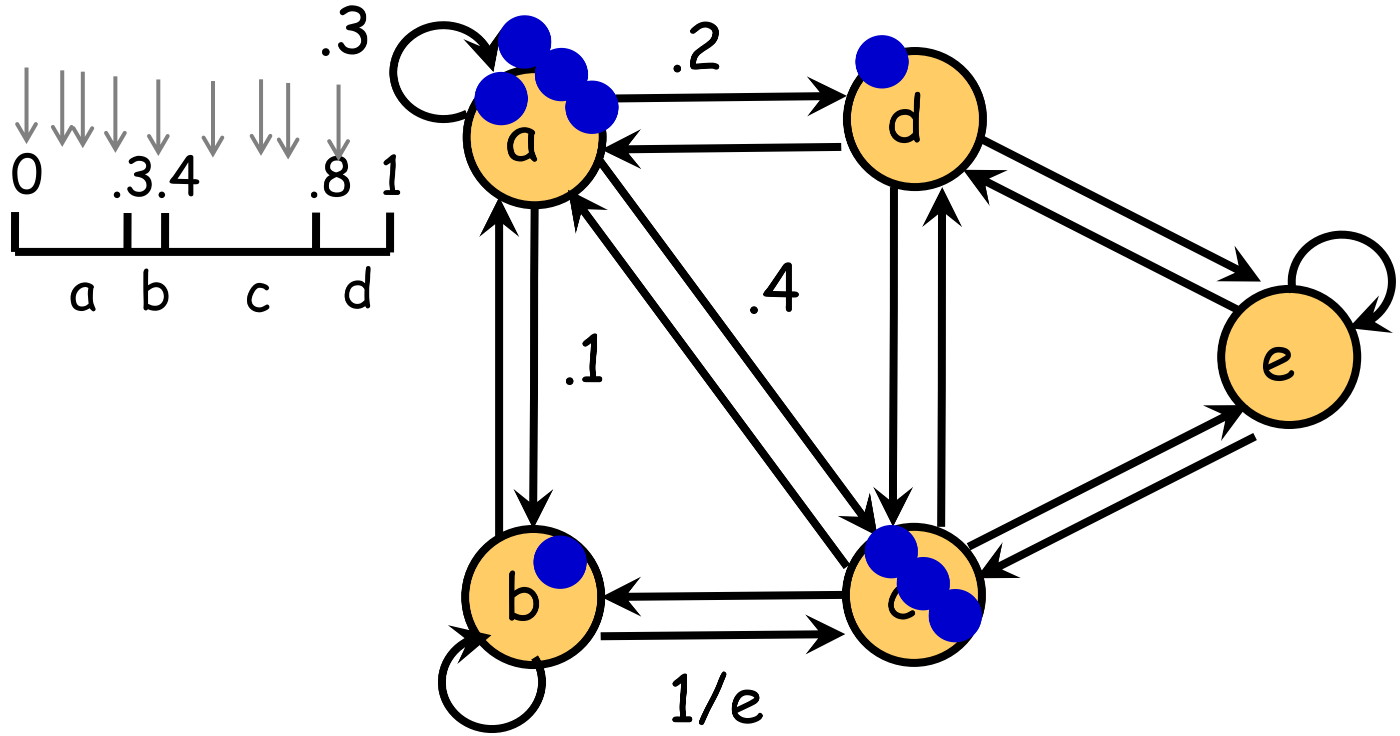


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: “関数ルーター” (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

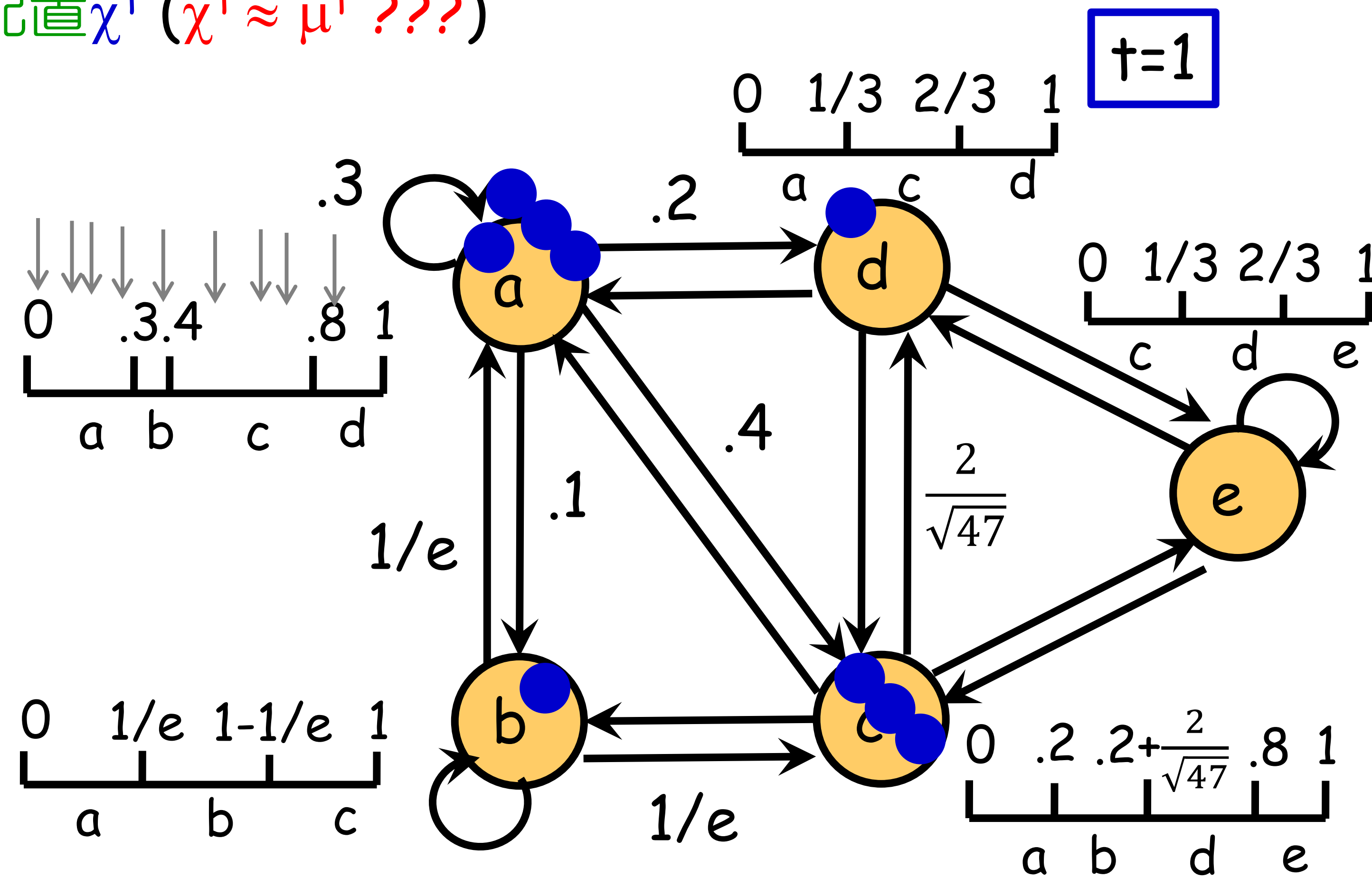
t=1



関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

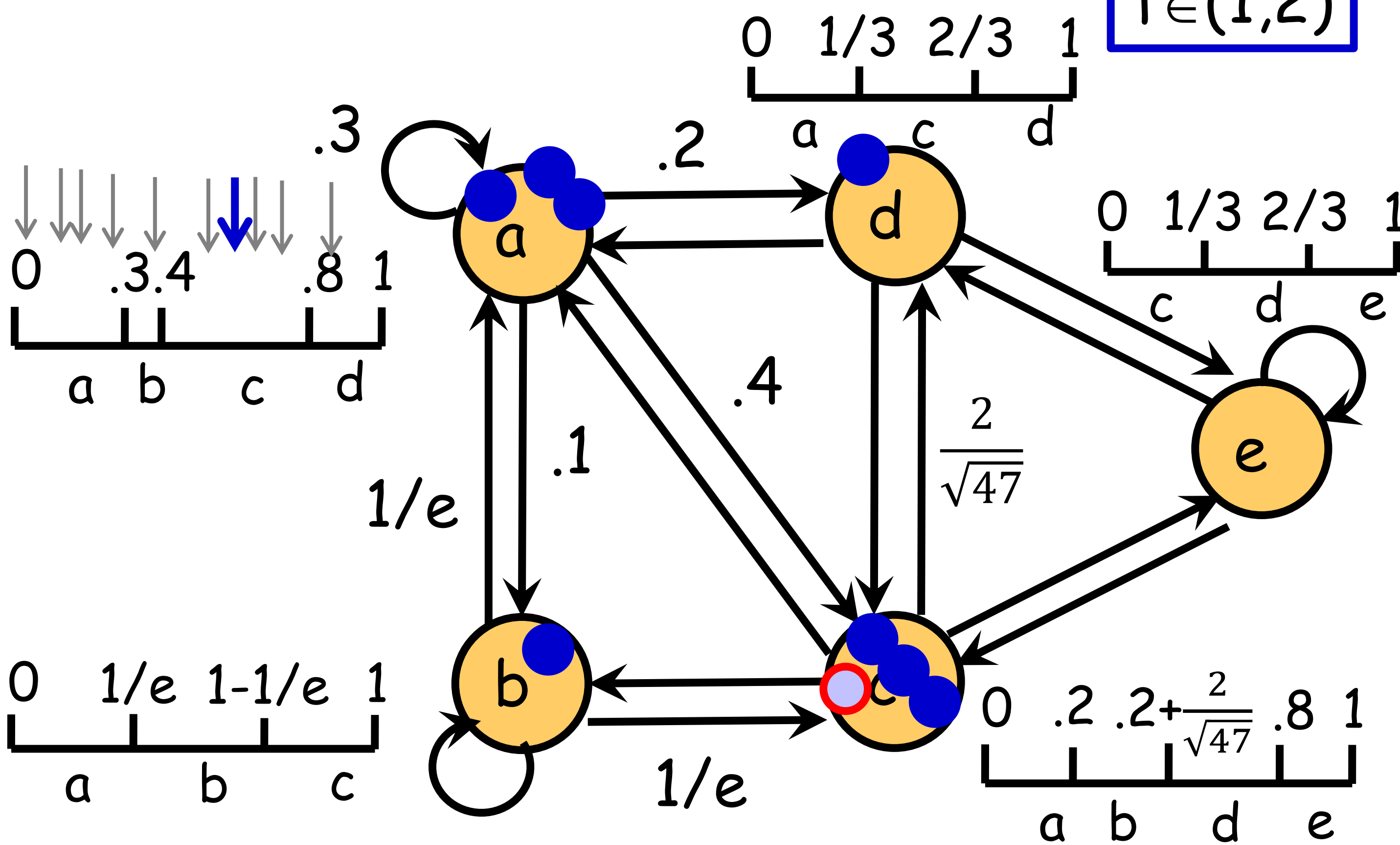


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (1,2)$

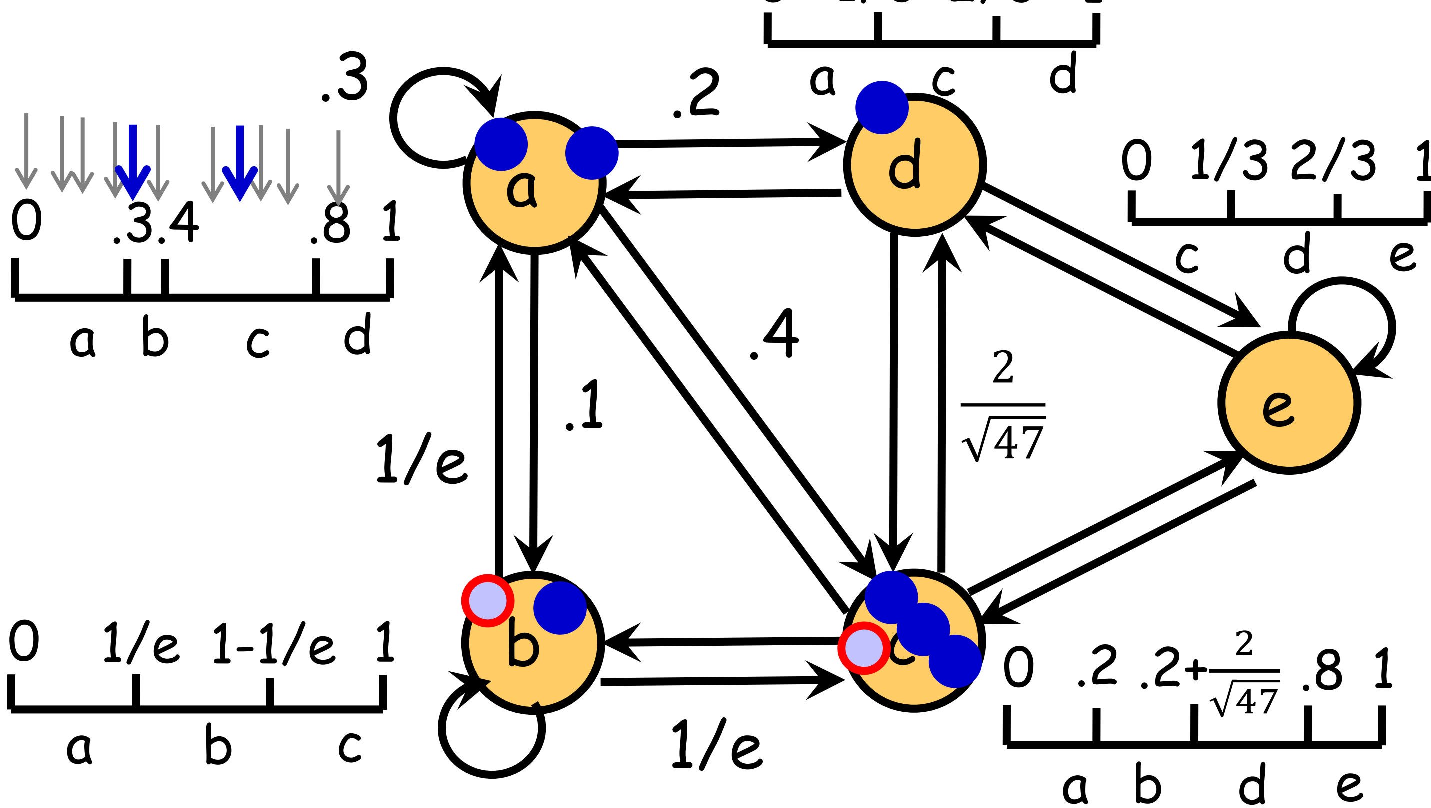


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

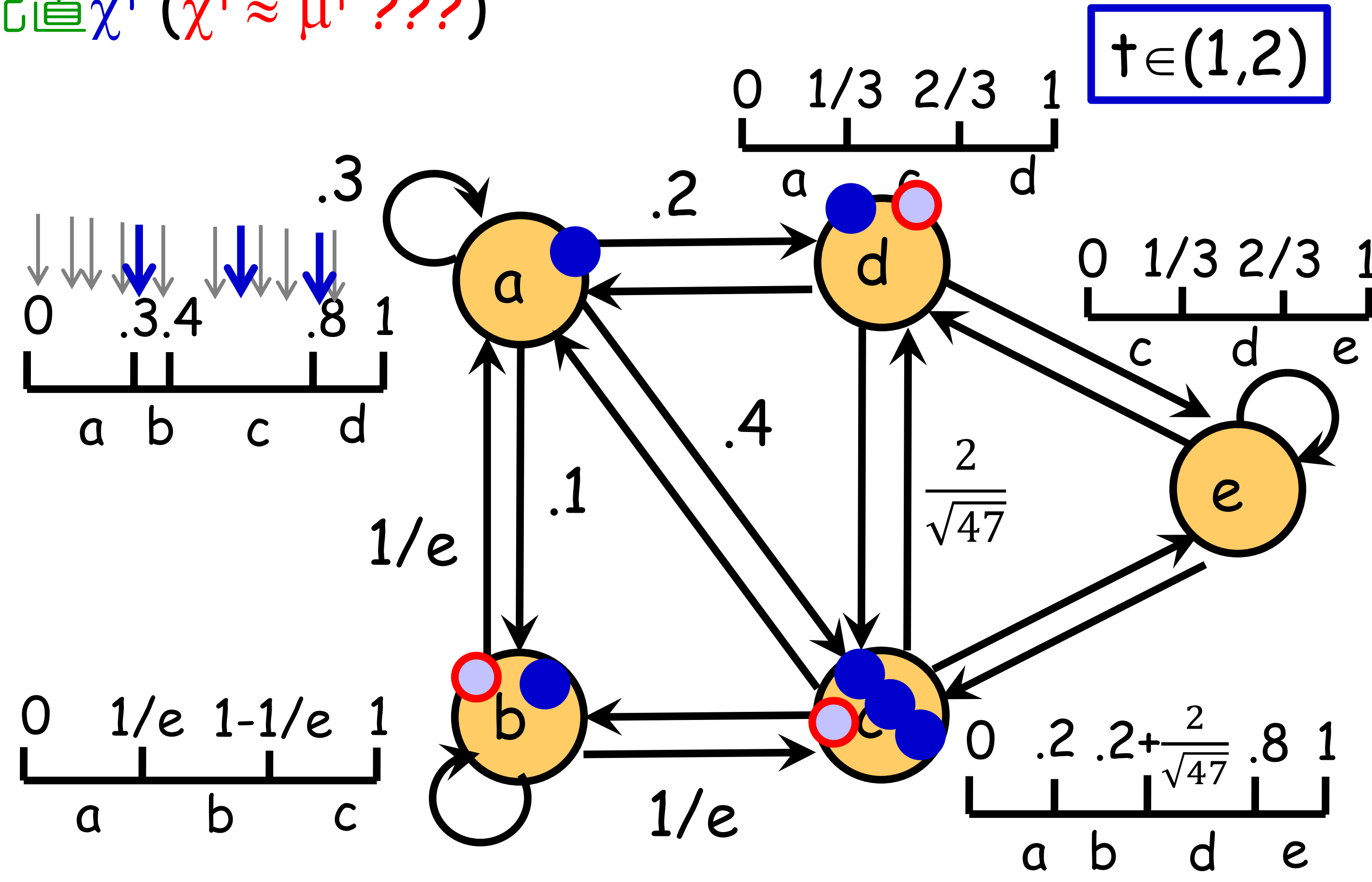
$t \in (1,2)$



関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

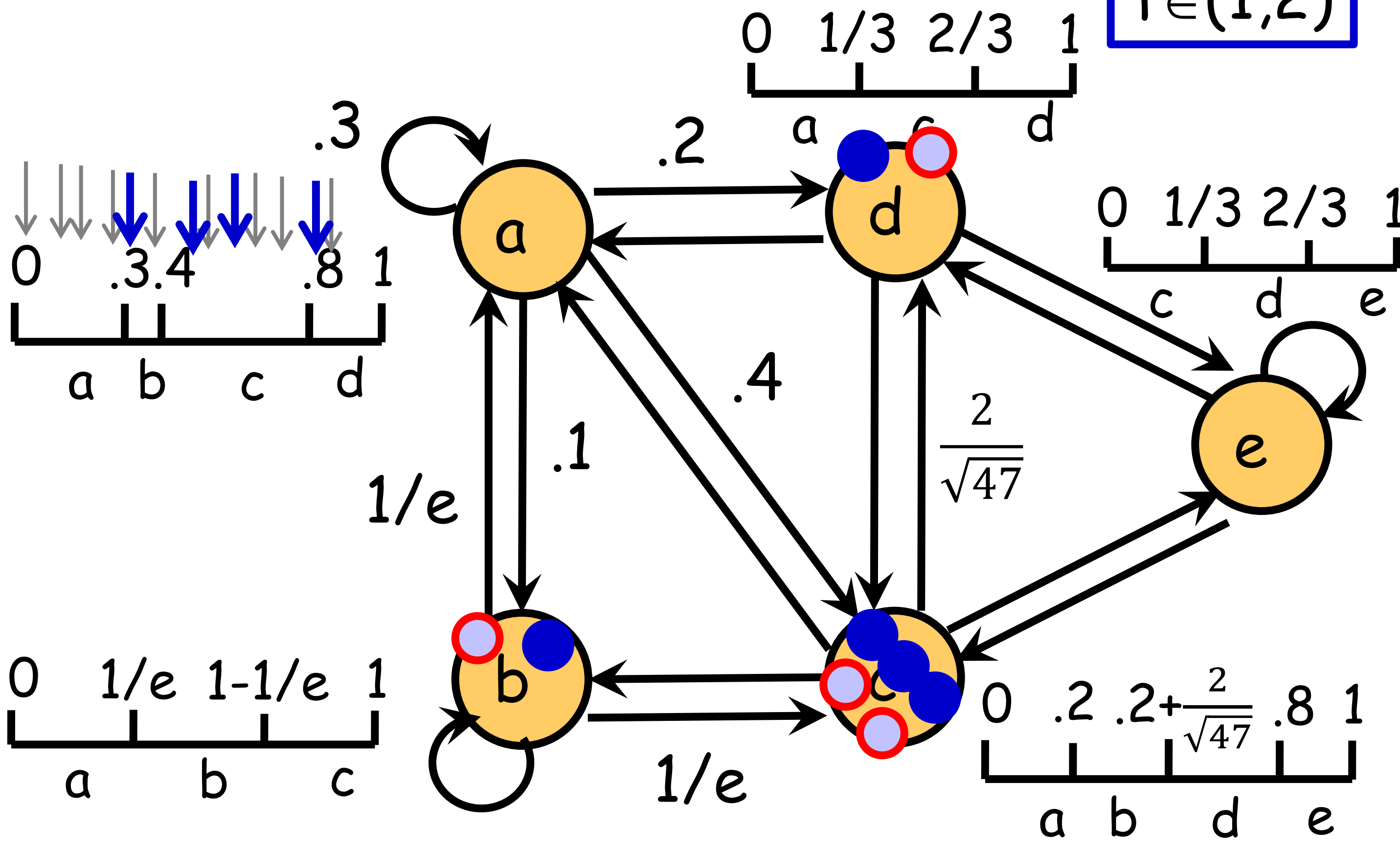


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (1,2)$

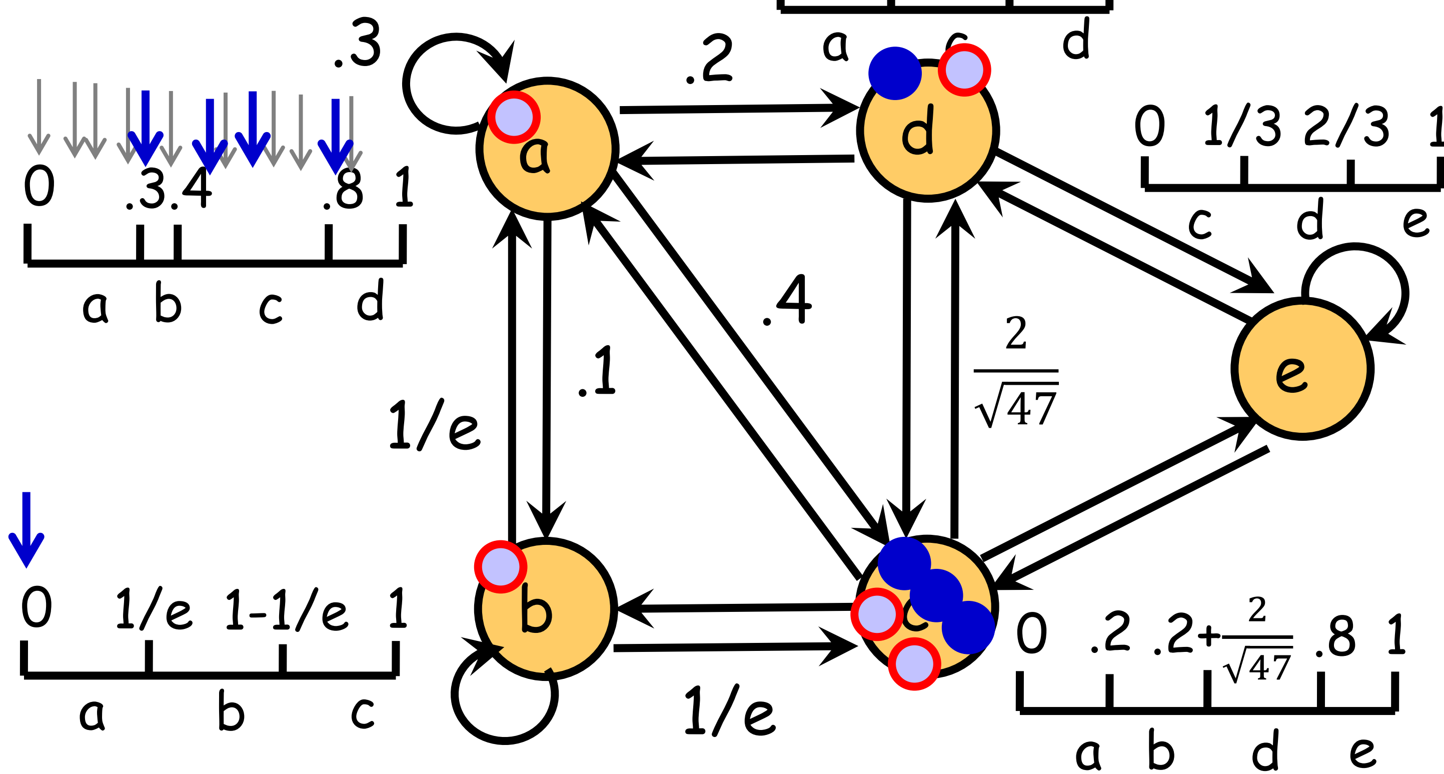


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (1,2)$

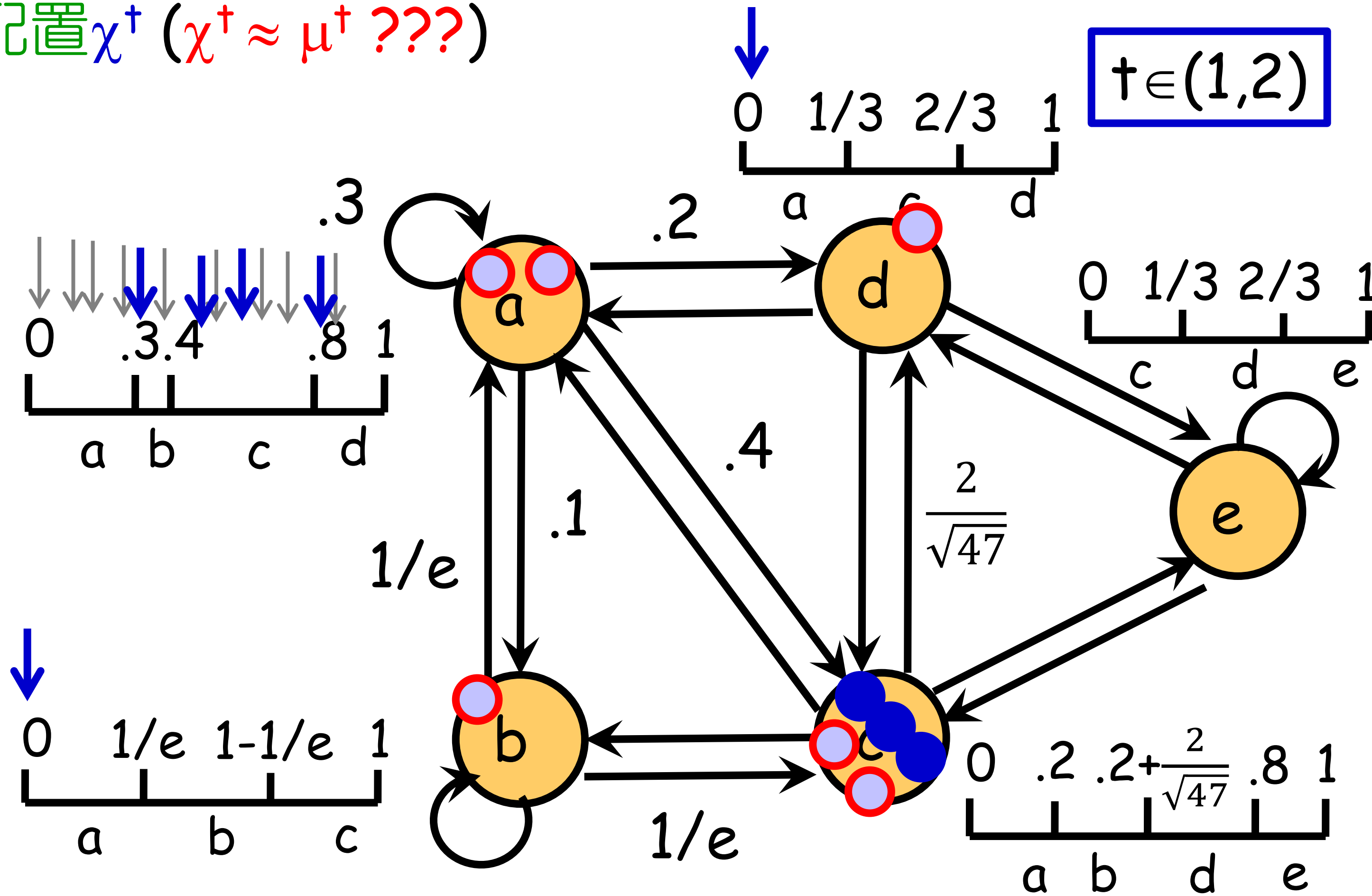


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

$t \in (1,2)$

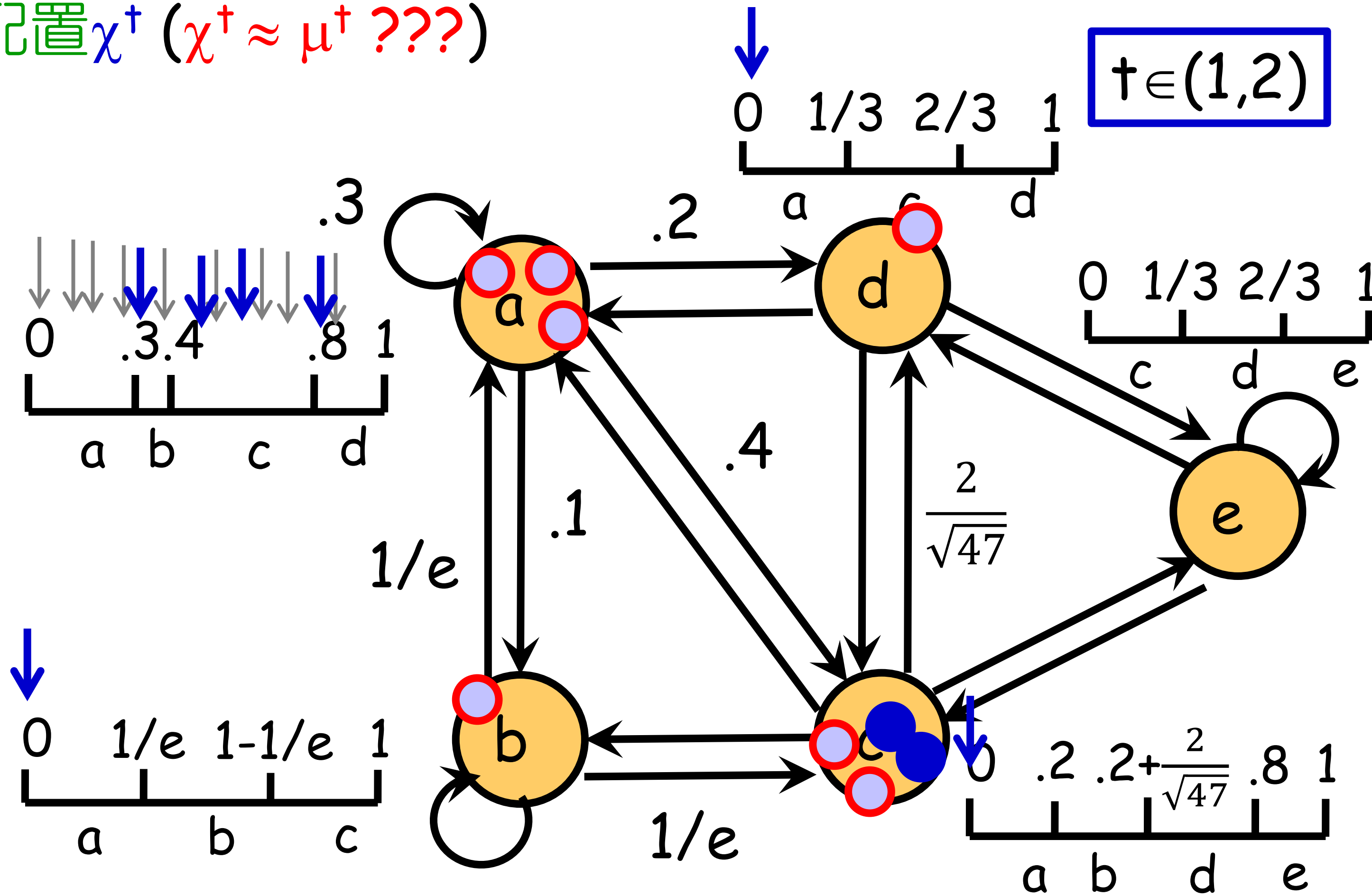


関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)

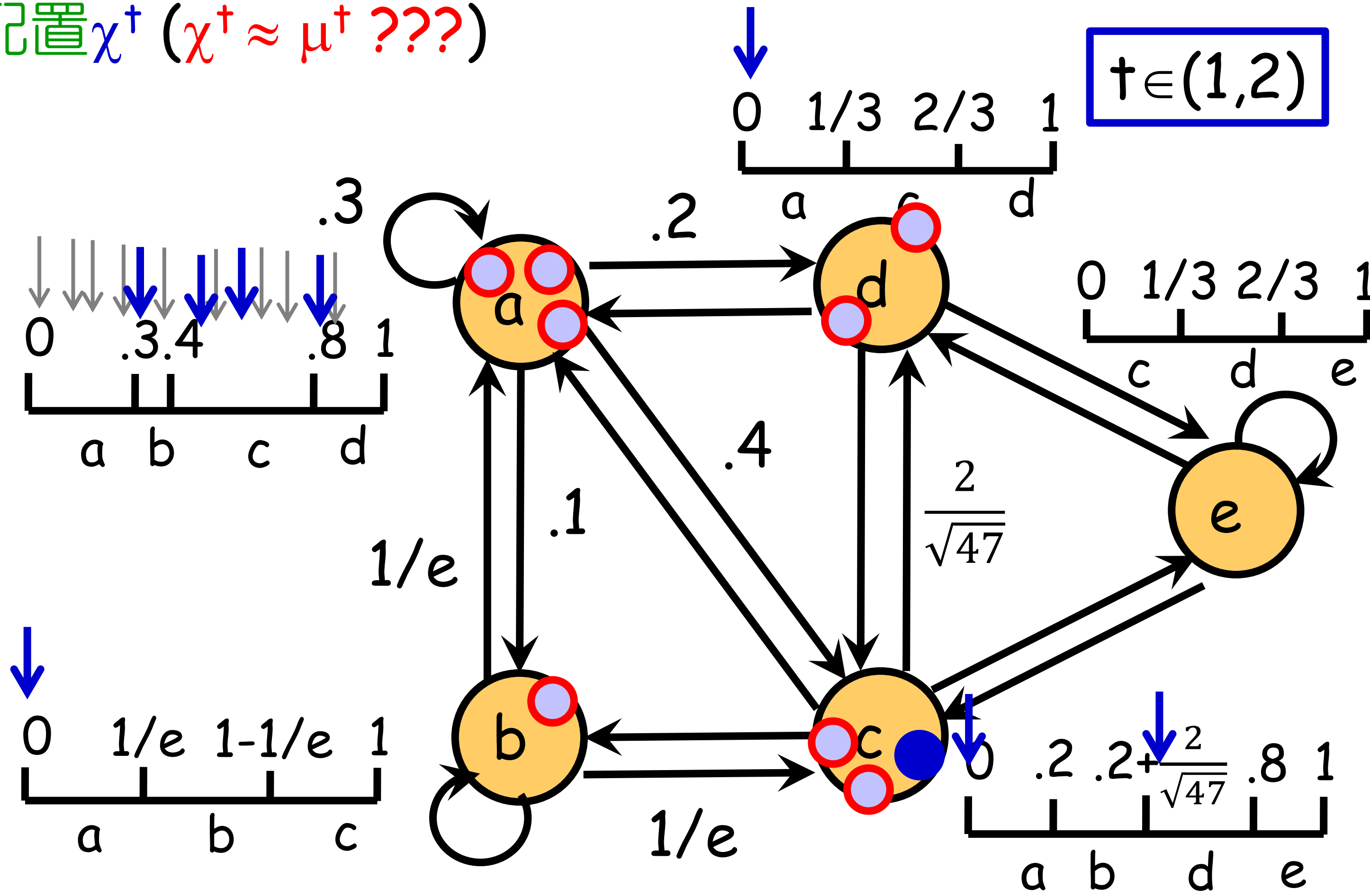
$\boxed{t \in (1,2)}$



関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

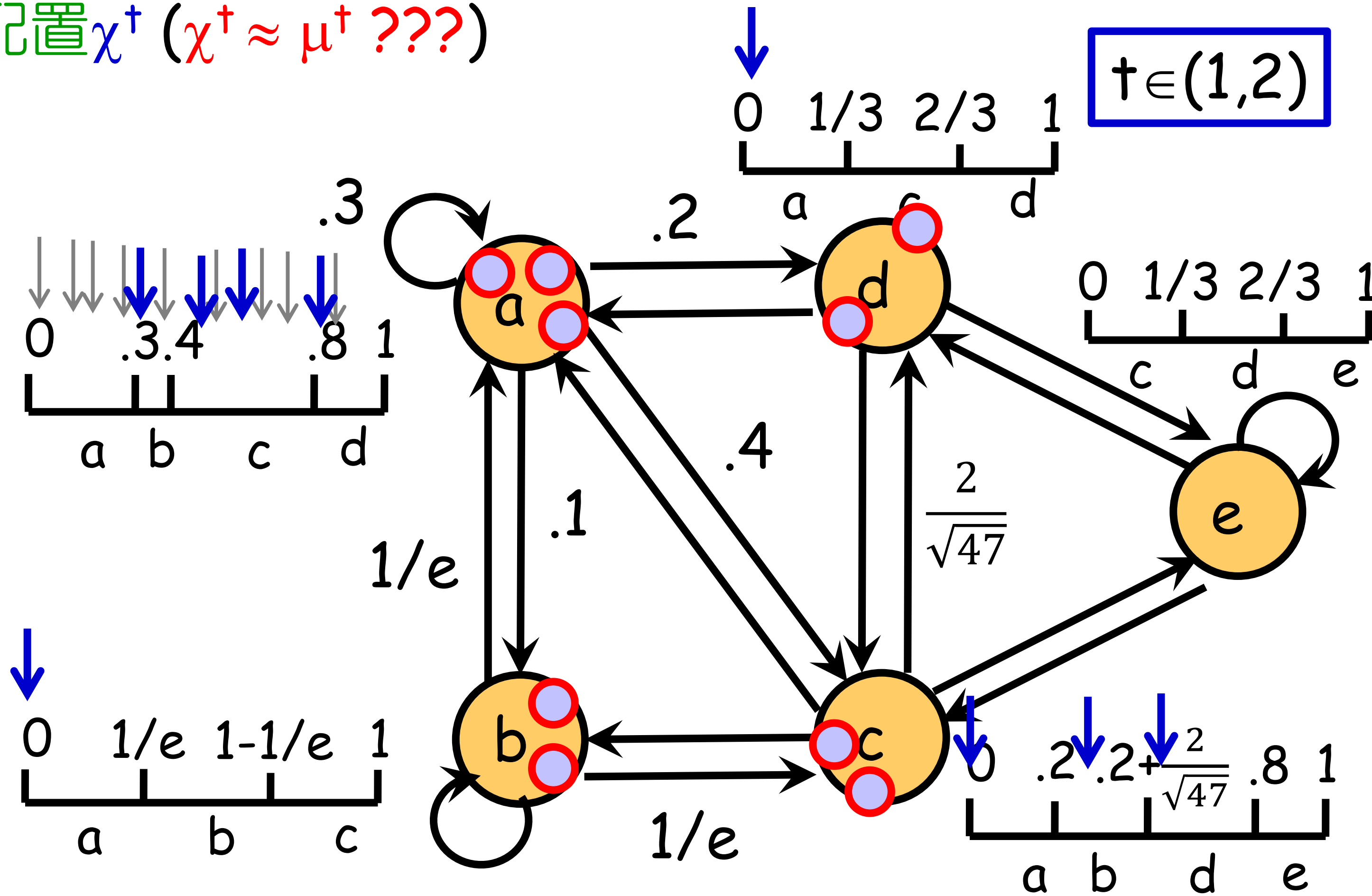
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

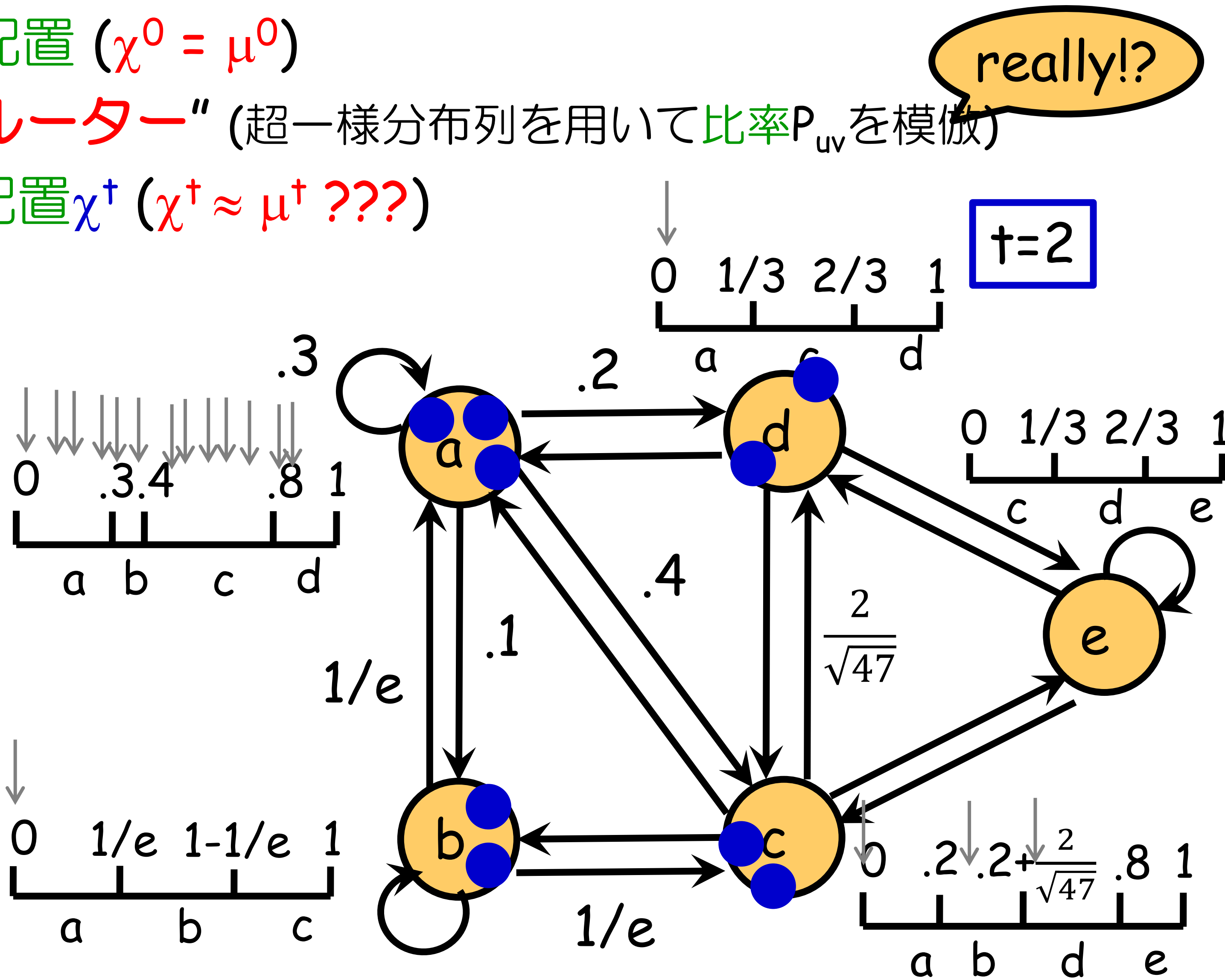
- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: "関数ルーター" (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



関数ルーターモデル

N個のトークンが決定的にグラフ上を移動.

- ✓ χ^0: 初期配置 ($\chi^0 = \mu^0$)
- ✓ σ: “関数ルーター” (超一様分布列を用いて比率P_{uv}を模倣)
- ✓ 時刻tの配置χ^t ($\chi^t \approx \mu^t$???)



$\left|I_{u,v}[0,z)\right|$: 頂点uから発射された総トークン数がz個の時, vに発射されたトークン数

van der Corput列の誤差

定理

任意の行列Pに対して,

$$\left|\frac{\left|I_{u,v}[0,z)\right|}{z} - P(u,v)\right| < \frac{2\lfloor \lg z \rfloor + 2}{z} = O\left(\frac{\log z}{z}\right)$$

が任意の$u, v \in V$ および$z \in \mathbb{Z}_{z \geq 0}$について成り立つ.

Unfortunately this bound is tight.

命題

ある行列 P に対して，ある $u, v \in V$で,

$$\left|\frac{\left|I_{u,v}[0,z)\right|}{z} - P(u,v)\right| > \frac{\lg\left(\frac{3}{4}z\right)}{3z} = \Omega\left(\frac{\log z}{z}\right)$$

が無数の$z \in \mathbb{Z}_{\geq 0}$について成り立つ.

下界の例

命題

無数の$M\in\mathbb{Z}_{>0}$に対して $\left|\chi_w^{(T)} - \mu_w^{(T)}\right| > \lg\left(\frac{3}{4}M\right)$
の成り立つ例が存在する.

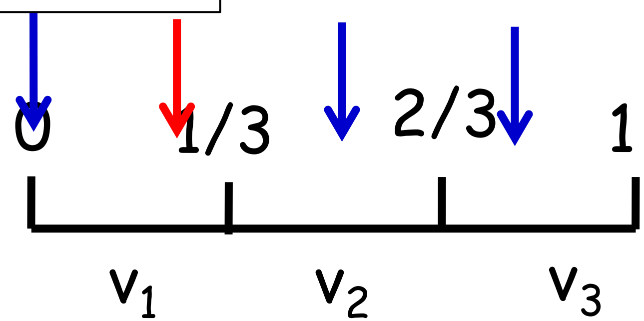
K_3上の単純ランダムウォークを考える.

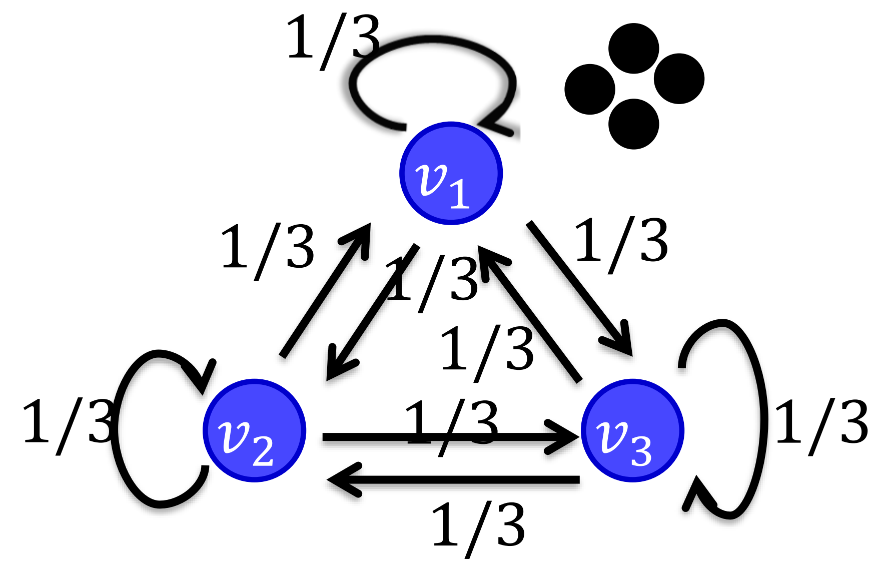
総トークン数を$M := \sum_{l=1}^{k} 4^l$ ($k\in\mathbb{Z}_{>0}$)とする.

このとき，$4^l \equiv 1 \pmod 3$

$(4^l - 1 = 3(4^{l-1} + \cdots + 1))$

直観的には





主定理

定理[白髪, 山内, K., 山下12+]

Pがエルゴード的で可逆の時,

任意の初期配置状態, 任意の$w \in V$, 任意の時刻 t に対して

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| < \sqrt{\frac{\pi(w)}{\pi_{\min}} \cdot \frac{m(n-1)}{1-\lambda^*} \cdot 2(\lg M + 1)}$$

が成り立つ．ただし， $n=|V|$, $m=|E|$,

λ^*はPの第２固有値，πはPの定常分布，$\pi_{\min} = \min_{v \in V} \pi(v)$ とする.

P が可逆とは詳細釣合の式

$$\pi(u)\, P(u,v) = \pi(v)\, P(v,u)$$

が任意の $u,v \in V$に成り立つことをいう

MCMC法では,
しばしば仮定される.

A sketch of proof (1/4)

$\chi_w^{(T)}$: #tokens @ w∈ V , @ time T > 0 in detRW
$\mu_w^{(T)}$: E[#tokens] @ w∈ V , @ time T > 0 in RW

$Z_{v,u}^{(t)}$: #tokens v→u (v,u∈V) during [t,t+1)

Rem. $\sum_{u\in N(v)} Z_{v,u}^{(t)} = \chi_v^{(t)}$

understanding [Cooper, Spencer2006]

by transition matrix P

Lemma 1.

$$\chi_w^{(T)} - \mu_w^{(T)} = \sum_{t=0}^{T-1} \sum_{v\in V} \sum_{u\in N(v)} \left(Z_{v,u}^{(t)} - \chi_v^{(t)} P(v,u) \right) P^{T-t-1}(u,w)$$

Proof sketch.

$$\chi^{(T)} - \mu^{(T)} = \chi^{(T)} - \mu^{(0)} P^T = \chi^{(T)} P^0 - \chi^{(0)} P^T$$

$$= \sum_{t=0}^{T-1} \left(\chi^{(t+1)} P^{T-t-1} - \chi^{(t)} P^{T-t} \right)$$

$$= \sum_{t=0}^{T-1} \left(\chi^{(t+1)} - \chi^{(t)} P \right) P^{T-t-1}$$

Carefully considering each element, we obtain the claim. $\blacksquare$

A sketch of proof (2/4)

It is known that

$\Pi^{1/2}\, P\, \Pi^{-1/2}$ is symmetric if P is reversible,

where $\Pi^{1/2}$ = diag($\sqrt{\pi(1)}, \ldots, \sqrt{\pi(n)}$).

Thus, P is diagonalized as $\Lambda = B^\top \Pi^{1/2}\, P\, \Pi^{-1/2} B$,

where B is orthonormal, and each eigenvalue is real.

Then,

$$P^z(v,u) = \boldsymbol{e_v} P^z \boldsymbol{e_u} = \boldsymbol{e_v} B^\top \Pi^{-\frac{1}{2}} \Lambda^z \Pi^{\frac{1}{2}} B \boldsymbol{e_u} = \frac{1}{\sqrt{\pi(v)}} e_v B^\top \Lambda^z B \sqrt{\pi(u)} \boldsymbol{e_u}$$

$$= \sqrt{\frac{\pi(u)}{\pi(v)}} \sum_{j=1}^{n} e_v b_j^\top (\lambda_j)^z b_j e_u^\top = \sqrt{\frac{\pi(u)}{\pi(v)}} \sum_{j=1}^{n} (\lambda_j)^z b_j(v) b_j(u)$$

A sketch of proof (3/4)

Lemma 2.

$$\sum_{u \in N(v)} \left(Z_{v,u}^{(t)} - \chi_v^{(t)} P(v,u) \right) \sqrt{\frac{\pi(w)}{\pi(u)}} b_1(v) b_1(u) = 0$$

since

$$b_1^{\top} = \left(\sqrt{\pi(1)}, \dots, \sqrt{\pi(n)} \right)^{\top}$$

holds.

A sketch of proof (4/4)

by Lemma 1

by "reversible"

$$
\begin{aligned}
&\left|\chi_w^{(T)} - \mu_w^{(T)}\right| \\
&= \sum_{t=0}^{T-1} \sum_{v \in V} \sum_{u \in N(v)} \left(Z_{v,u}^{(t)} - \chi_v^{(t)} P(v,u) \right) P^{T-t-1}(u,w) \\
&\leq \sum_{t=0}^{T-1} \sum_{v \in V} \sum_{u \in N(v)} \left| \left(Z_{v,u}^{(t)} - \chi_v^{(t)} P(v,u) \right) \sqrt{\frac{\pi(w)}{\pi(v)}} \sum_{j=1}^{n} \left(\lambda_j\right)^{T-t-1} b_j(v) b_j(u) \right| \\
&\leq \frac{\sqrt{\pi(w)}}{\sqrt{\pi_{\min}}} \sum_{t=0}^{T-1} |\lambda_2|^{T-t-1} \sum_{v \in V} \sum_{u \in N(v)} \sum_{j=2}^{n} \left| \left(Z_{v,u}^{(t)} - \chi_v^{(t)} P(v,u) \right) \right| |b_j(v) b_j(u)| \\
&\leq \frac{\sqrt{\pi(w)}}{\sqrt{\pi_{\min}}} \cdot \frac{1}{1-\lambda_2} \sum_{v \in V} \sum_{u \in N(v)} \sum_{j=2}^{n} \left| \left(Z_{v,u}^{(t)} - \chi_v^{(t)} P(v,u) \right) \right| \\
&\leq \frac{\sqrt{\pi(w)}}{\sqrt{\pi_{\min}}} \cdot \frac{m(n-1)}{1-\lambda_2} \cdot 2(\lg M + 1)
\end{aligned}
$$

note: λ_1=1

Main Thm.

Thm. [Shiraga, Yamauchi, K., Yamashita 12+]

If P is ergodic and reversible, then

for any ini. config., for any $w \in V$, for any time t,

$$\left|\chi_w^{(T)} - \mu_w^{(T)}\right| < \sqrt{\frac{\pi_w}{\pi_{\min}} \cdot \frac{m(n-1)}{1-\lambda^*} \cdot 2(\lg M + 1)}$$

holds where, $n=|V|$, $m=|E|$, M=#tokens ($=|\chi^{(0)}|$)

λ^* is the second largest eigenvalue of P, and

π is the stationary distribution.

P is reversible if the "detailed balance equation"

$$\pi(u)\, P(u,v) = \pi(v)\, P(v,u)$$

holds for any $u,v \in V$.

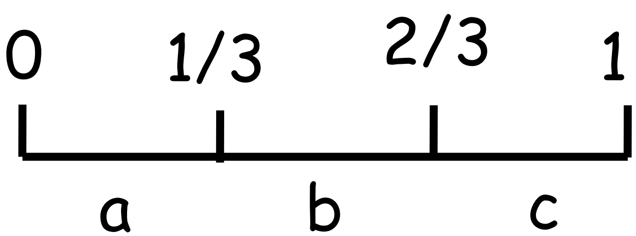
often assumed
in the context of MCMC

注意

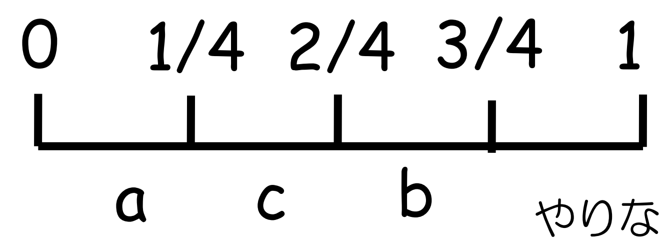
"棄却サンプリング"のアイデアに基づいて,
関数ルーター (の変種)がロータールーターに一致する
(ただし, 推移確率行列は有理数とする.)

ロータールーター: $\rho=\langle a,b,c\rangle$

(素朴な) 関数ルーター



(変形) 関数ルーター



やりなおし

"関数ルーター" はロータールーターの一般化モデル

参考文献

- Shuji Kijima, Kentaro Koga and Kazuhisa Makino, Deterministic random walks on finite graphs, Random Structures & Algorithms, 46(4): 739--761, 2015.
- Takeharu Shiraga, Yukiko Yamauchi, Shuji Kijima and Masafumi Yamashita, Deterministic random walks for rapidly mixing chains, SIAM Journal on Discrete Mathematics, 32(3):2180--2193, 2018.
- Takeharu Shiraga, Yukiko Yamauchi, Shuji Kijima and Masafumi Yamashita, Total variation discrepancy of deterministic random walks for ergodic Markov chains, Theoretical Computer Science, 699:63--74, 2017.

The end

Thank you for the attention.