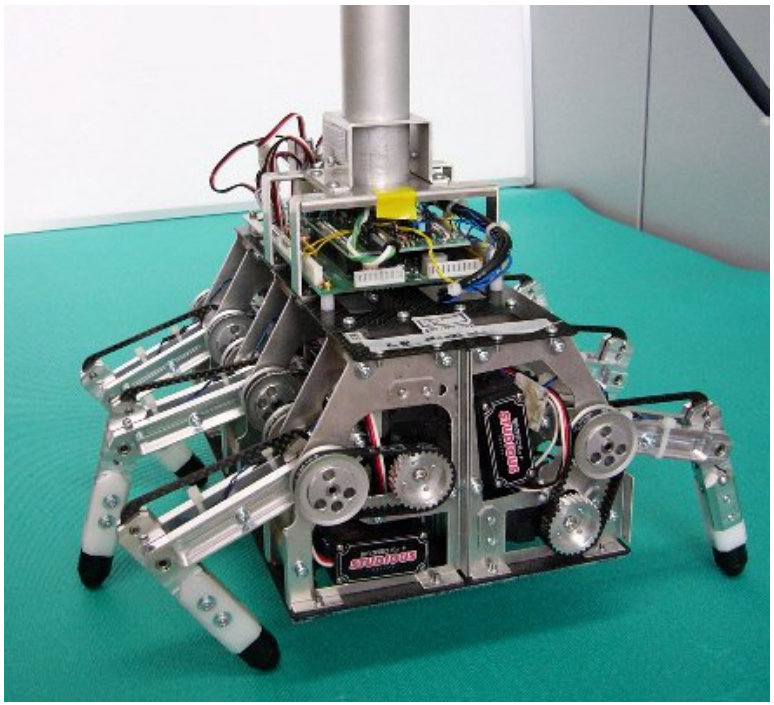
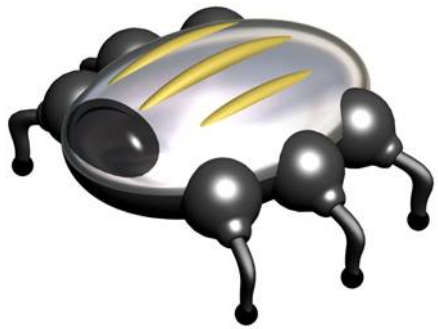
多次元状態-行動空間での強化学習
ーランダム矩形タイルによる汎化方法の提案ー

木村　元
九州大学　大学院工学研究院海洋システム工学部門





http://sysplan.nams.kyushu-u.ac.jp/gen/index.html

ロボットの知能化

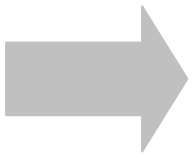
→ 強化学習：試行錯誤を通じて制御規則を獲得

学習アルゴリズムに対する要請：

1）とにかく**単純**な計算処理
- ●プログラムコードが短いこと
- ●自然言語で簡潔に表現できるのが望ましい

} アルゴリズム中に「微分記号」など論外！

2）**安定**した学習動作
- ●問題に依存せずに実装可能であること
- ●パラメータ設定にセンシティブではないこと



Actor-Critic　　Q-learning

ロボットの知能化

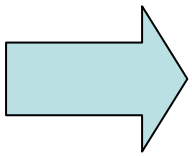
→ 強化学習：試行錯誤を通じて制御規則を獲得

学習アルゴリズムに対する要請：

1）とにかく**単純**な計算処理
- ●プログラムコードが短いこと
- ●自然言語で簡潔に表現できるのが望ましい

} アルゴリズム中に「微分記号」など論外！

2）**安定**した学習動作
- ●問題に依存せずに実装可能であること
- ●パラメータ設定にセンシティブではないこと



Actor-Critic　　Q-learning

ロボットの知能化

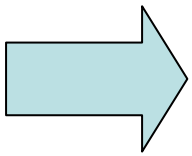
→ 強化学習：試行錯誤を通じて制御規則を獲得

学習アルゴリズムに対する要請：

1）とにかく**単純**な計算処理
- ●プログラムコードが短いこと
- ●自然言語で簡潔に表現できるのが望ましい

} アルゴリズム中に「微分記号」など論外！

2）**安定**した学習動作
- ●問題に依存せずに実装可能であること
- ●パラメータ設定にセンシティブではないこと



~~Actor-Critic~~

Q-learning

本研究の目的

高次元連続状態-行動空間のロボット強化学習

Q-learning

連続空間の関数近似

ボルツマン行動選択

- **これら基礎的な技術と知見をそのまま継承した実装**
- **数万ステップで学習させる （Actor-Criticの数倍程度以内）**

解決すべき課題：

1）状態-行動の評価値（Q値）の表現　←膨大な空間の汎化

2）高次元連続行動空間における行動選択は？

本研究の目的

高次元連続状態-行動空間のロボット強化学習

Q-learning

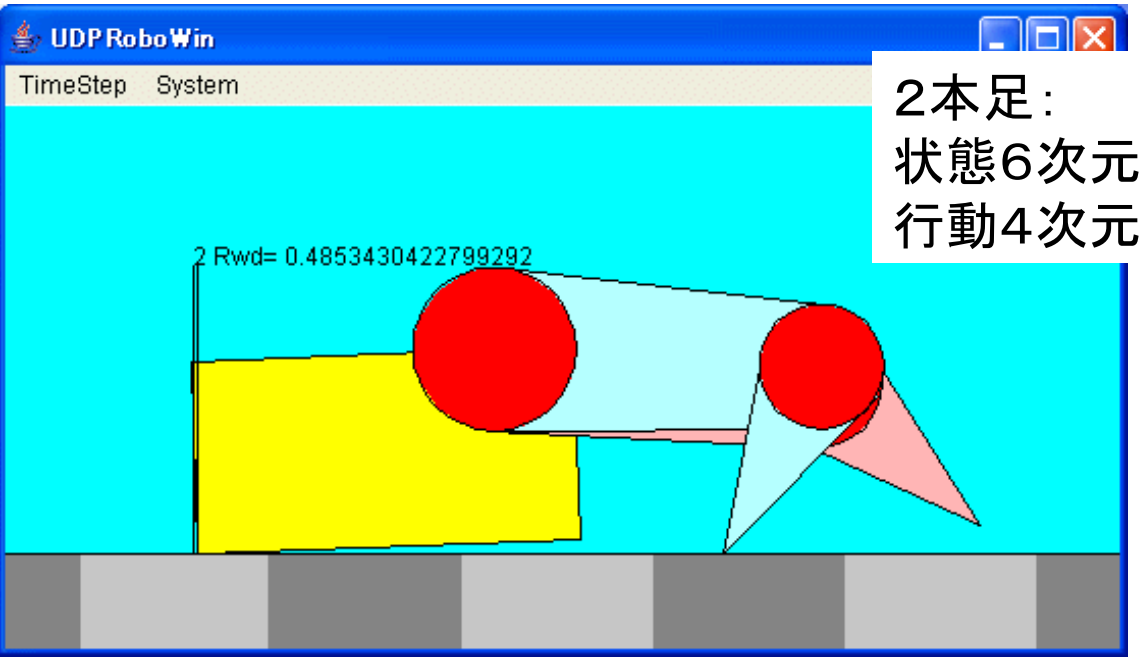
連続空間の関数近似

ボルツマン行動選択

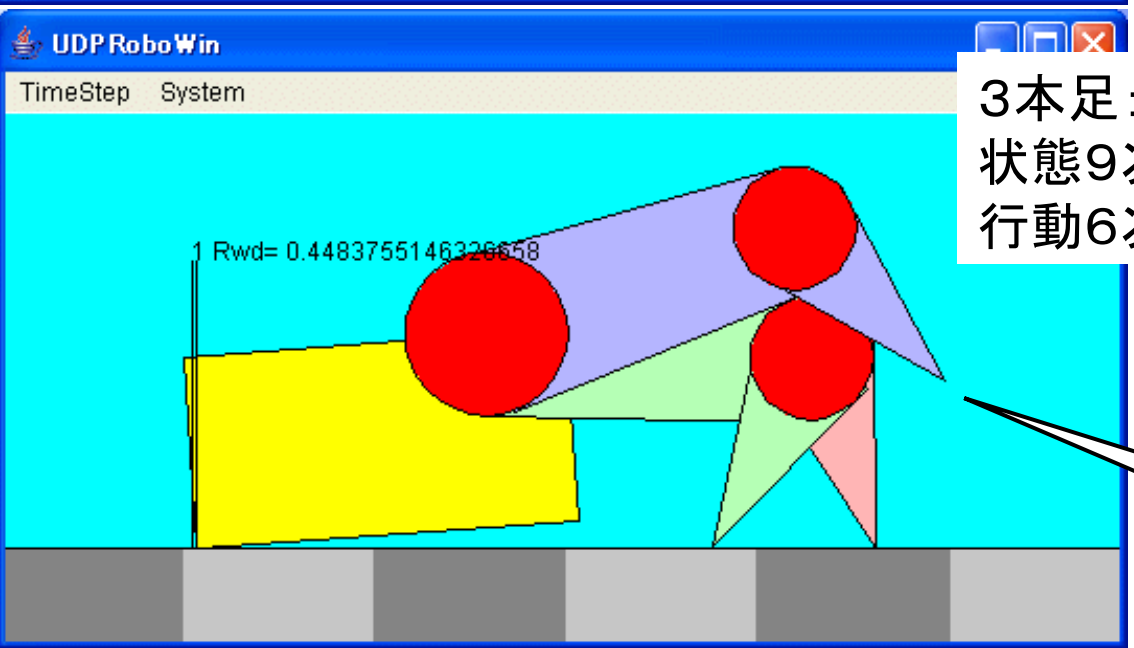
・これら基礎的な技術と知見をそのまま継承した実装
・数万ステップで学習させる （Actor-Criticの数倍程度以内）

解決すべき課題：
1）状態-行動の評価値（Q値）の表現　←膨大な空間の汎化
2）高次元連続行動空間における行動選択は？

対象とする学習問題1：ほふくロボット問題



2本足：
状態6次元
行動4次元



3本足：
状態9次元
行動6次元

関節を動かして、
這って進む動作を獲得する

2本足と3本足の問題

状態： 各関節角度とセンサ

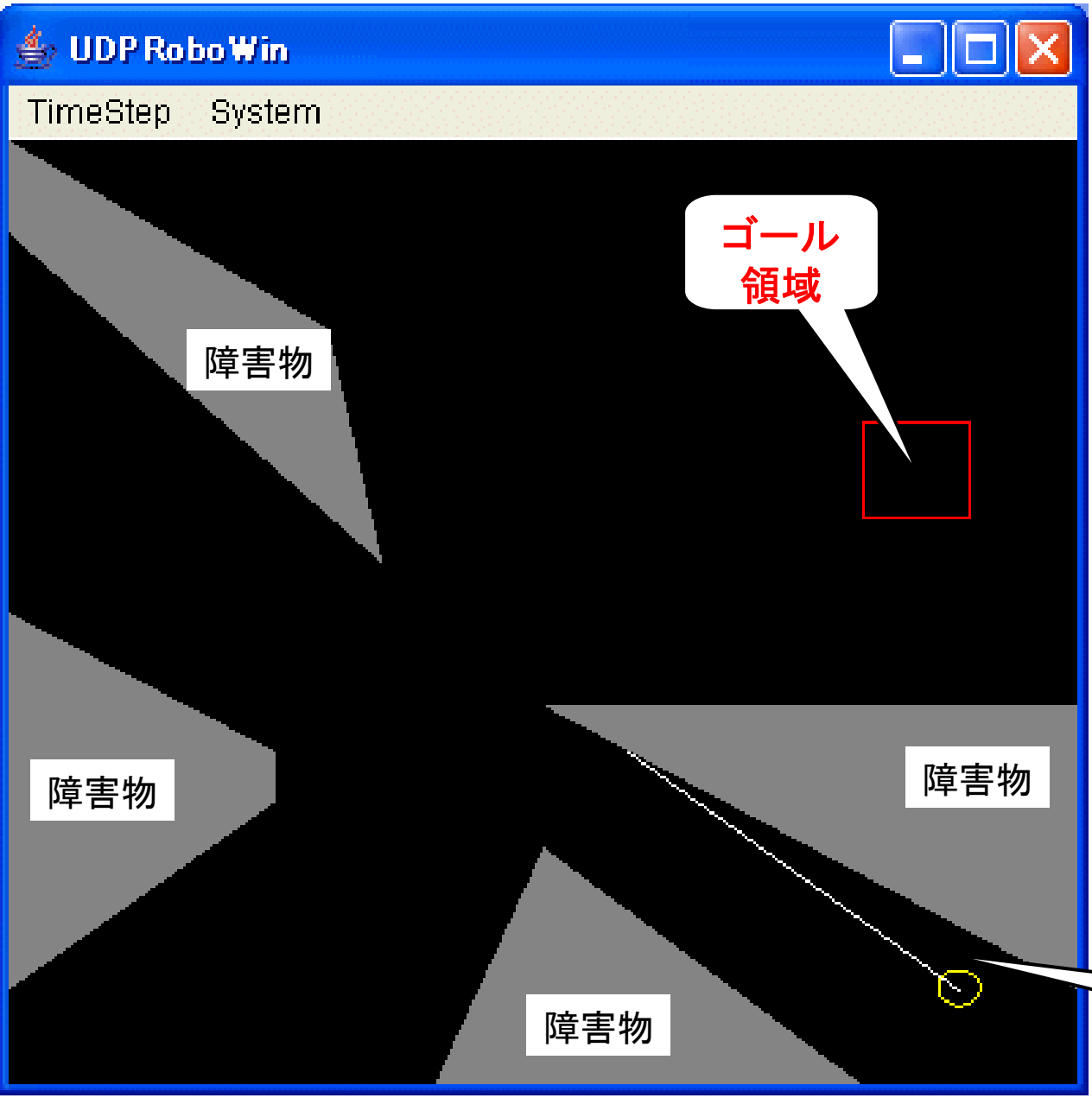
行動： 動作関節角度

足が目標角度へ到達するか、
途中で足先タッチセンサの
値が変化すると
意思決定のイベント発生

毎ステップ前進した距離に
比例して報酬

**足先に
タッチセンサ**

対象とする学習問題2：Rod in Maze (Moore 1995)



2次元平面中でスタート状態からゴール領域まで棒を移動

領域中に固定障害物

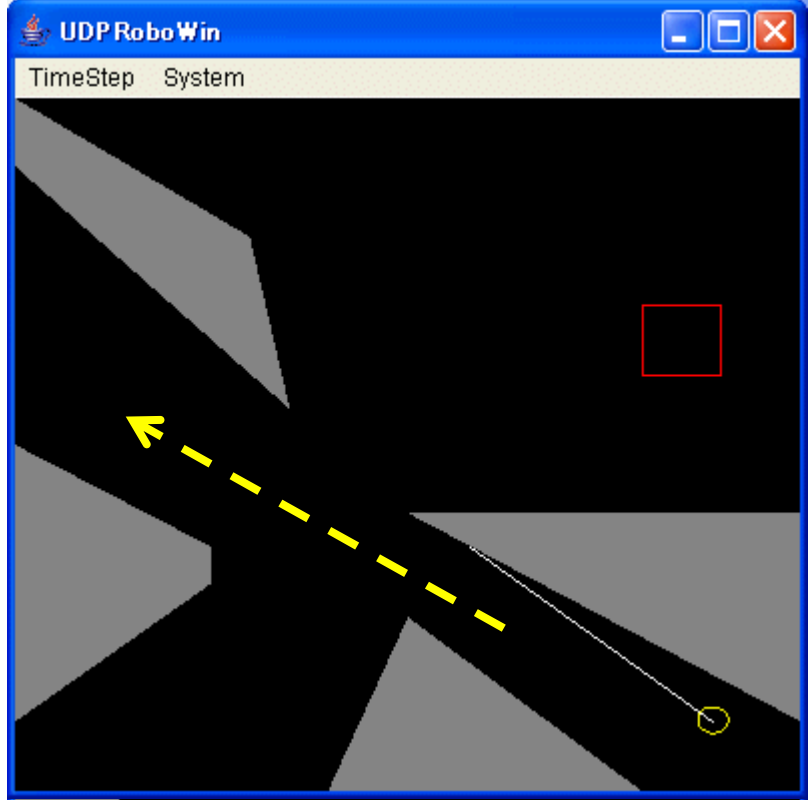
棒の傾き　0～180度のみ

状態：　X, Y座標と角度 θ

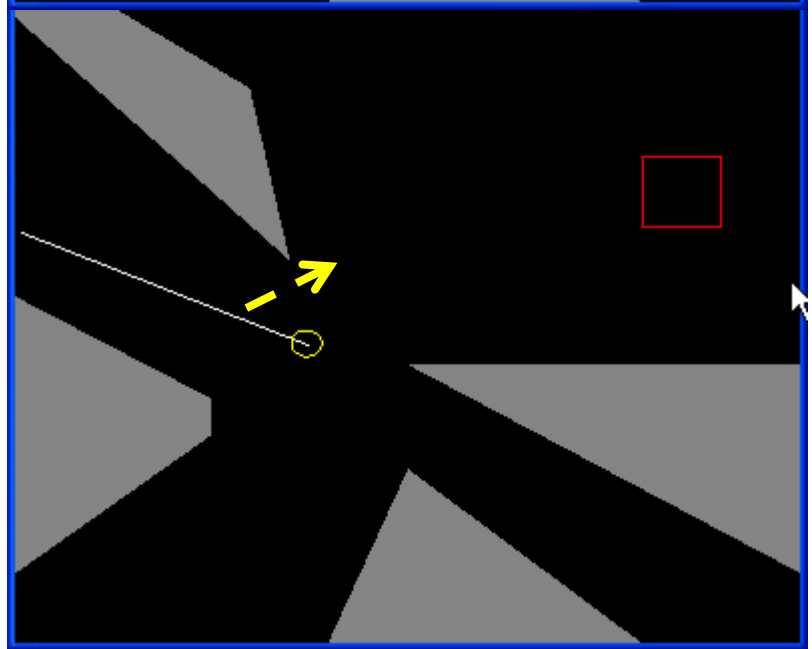
行動：　動作目標X,Y, θ 座標

棒が目標座標へ到達するか、途中で障害物へぶつかると意思決定のイベント発生

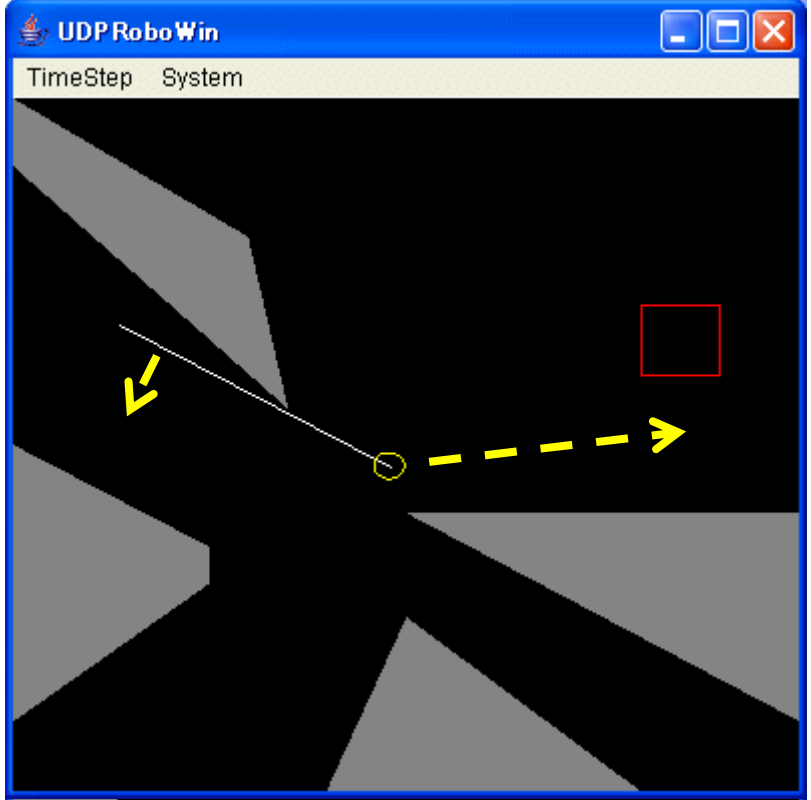
ゴールへ到達すると報酬／再び初期状態へもどる



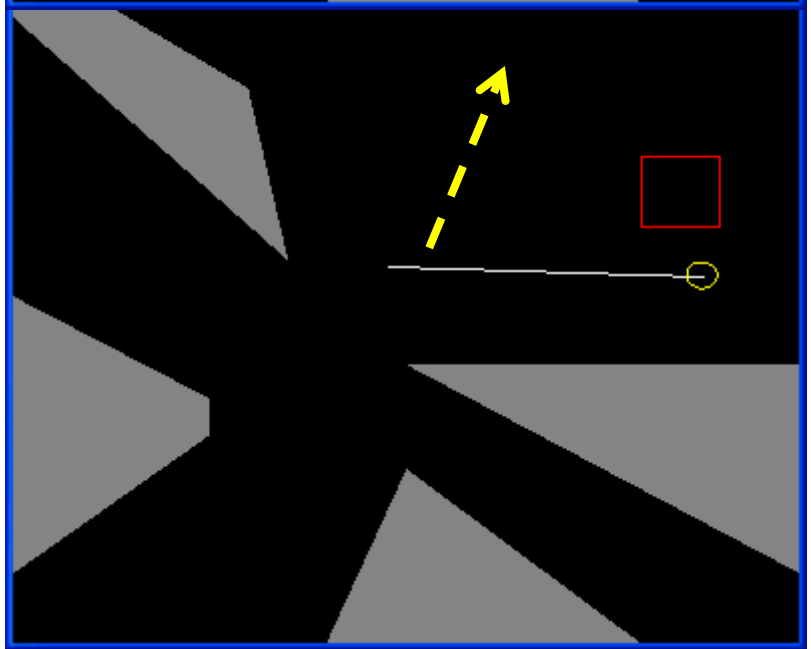
(1)



(2)

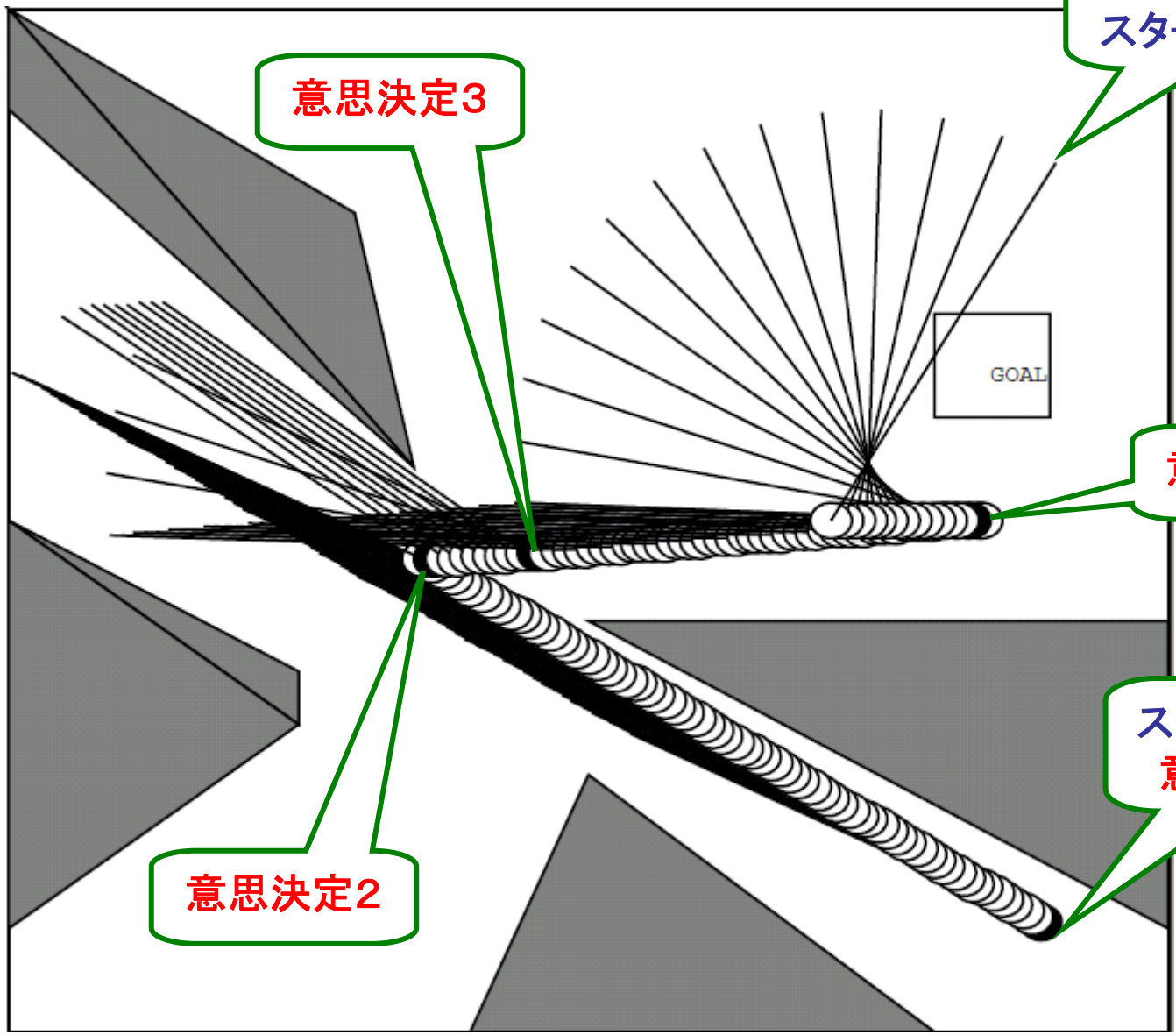


(3)



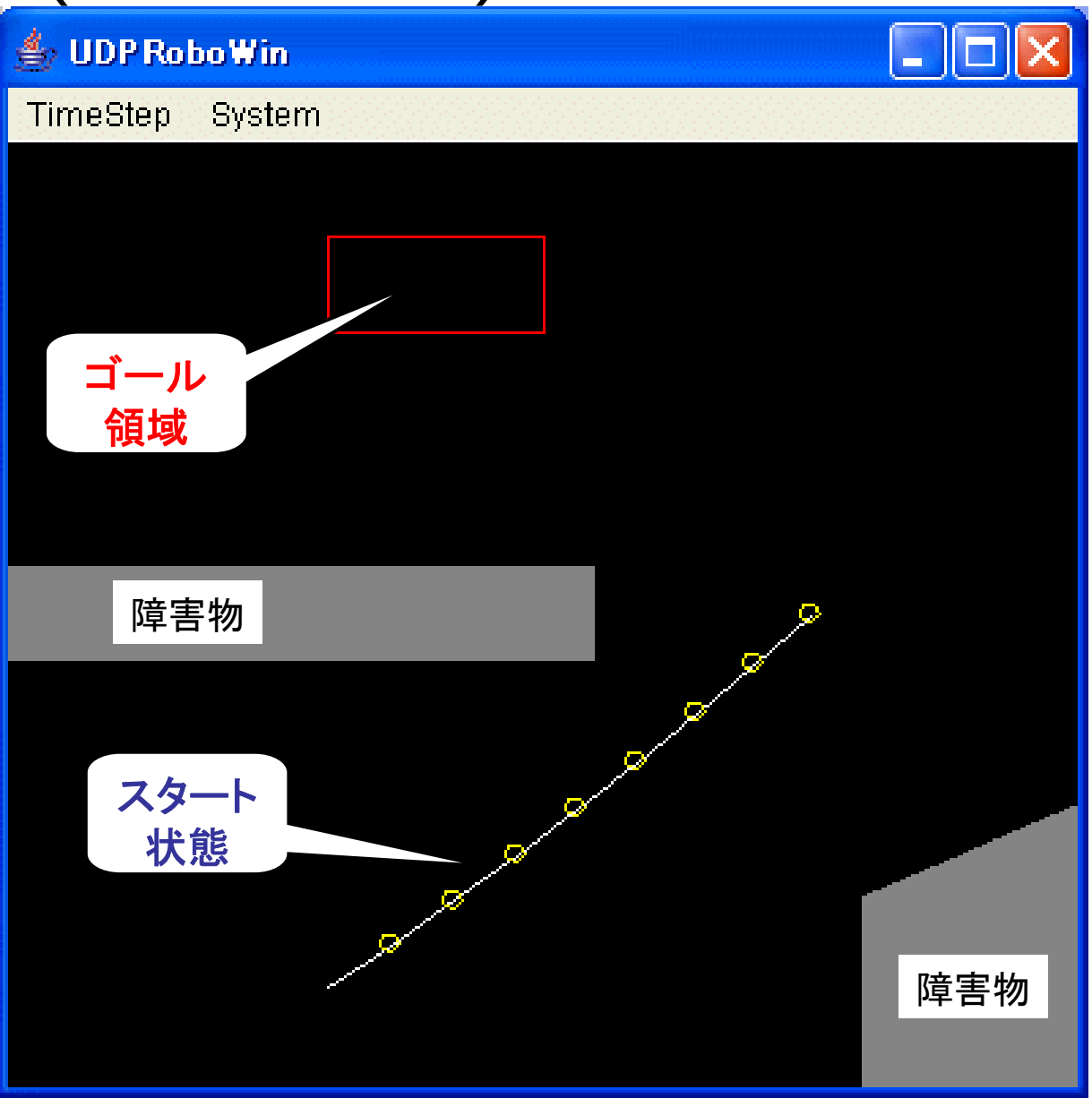
(4)

動作例



3次元状態×**3次元行動**の強化学習問題

対象とする学習問題3：Multi-Joint Arm (Moore 1995)



2次元平面中でスタート状態からゴール領域までアームを移動

領域中に固定障害物

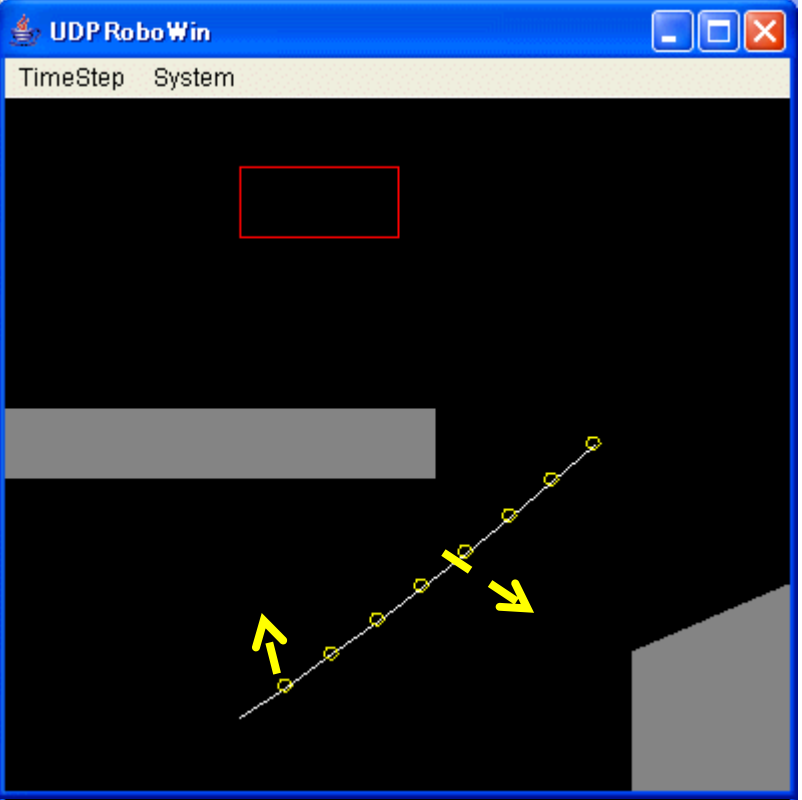
アームは冗長自由度（8関節）

状態： 関節角度 θ（8次元）

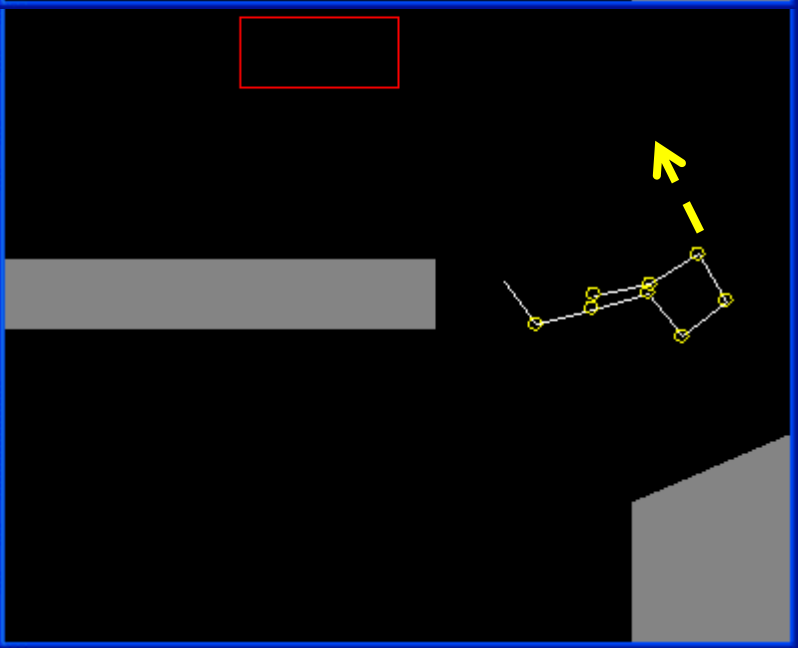
行動： 目標角度 θ（8次元）

アームが目標角度へ到達するか、途中で障害物へぶつかると意思決定のイベント発生

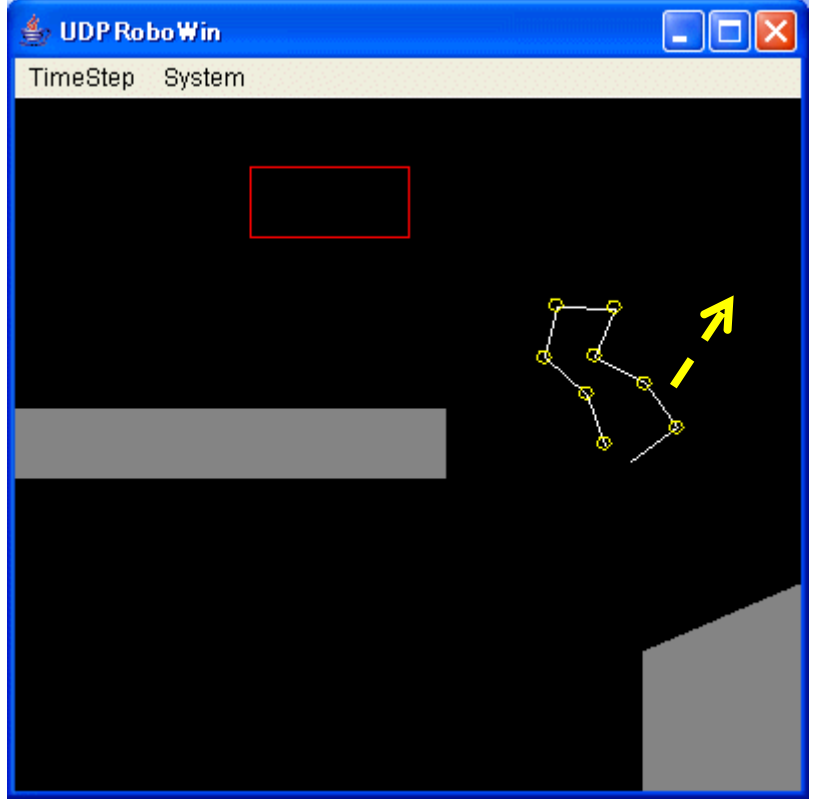
ゴールへ到達すると報酬／再び初期状態へもどる



(1)

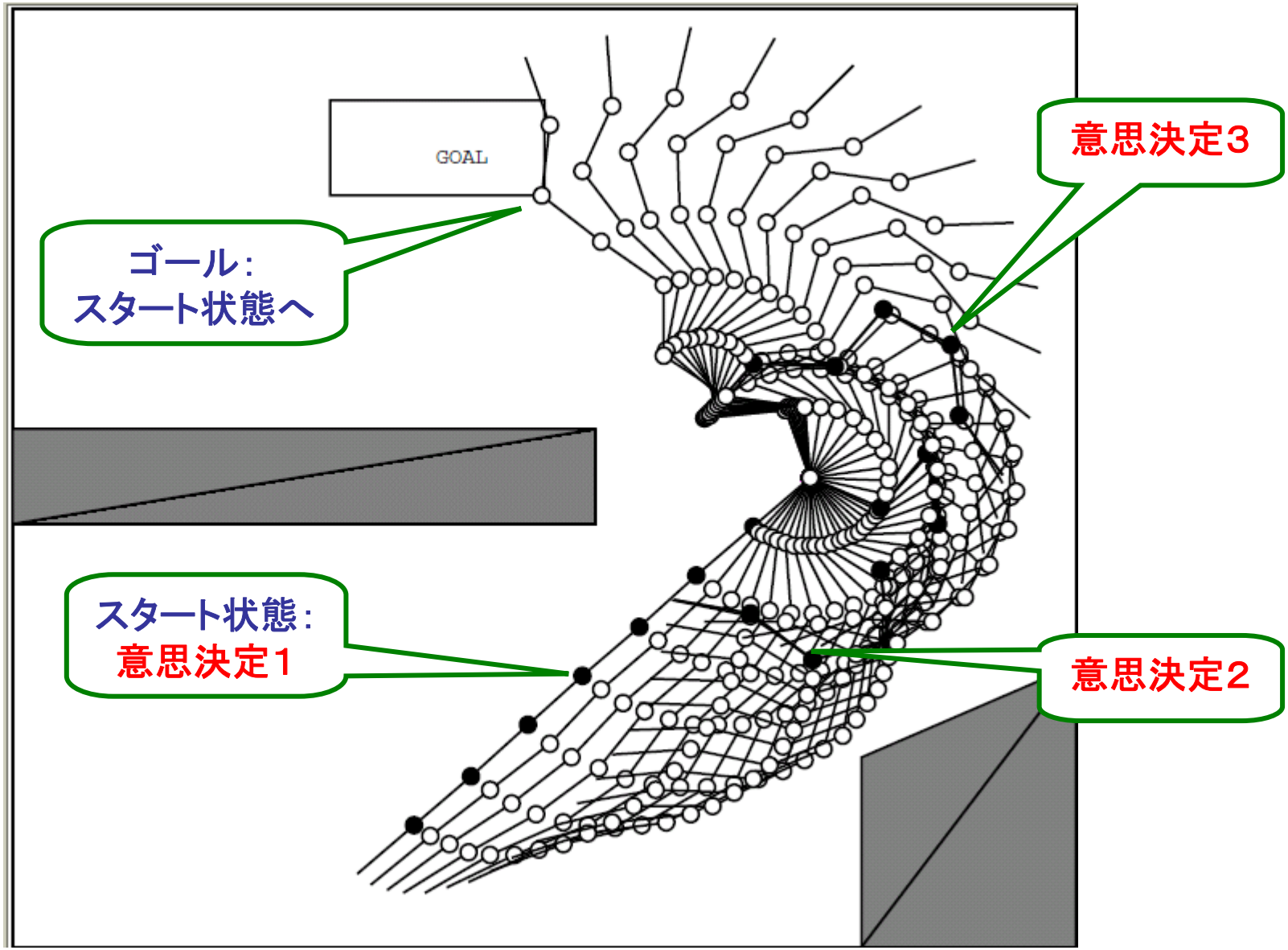


(2)



(3)

動作例



8次元状態×8次元行動の強化学習問題

Q-learning

連続空間の関数近似

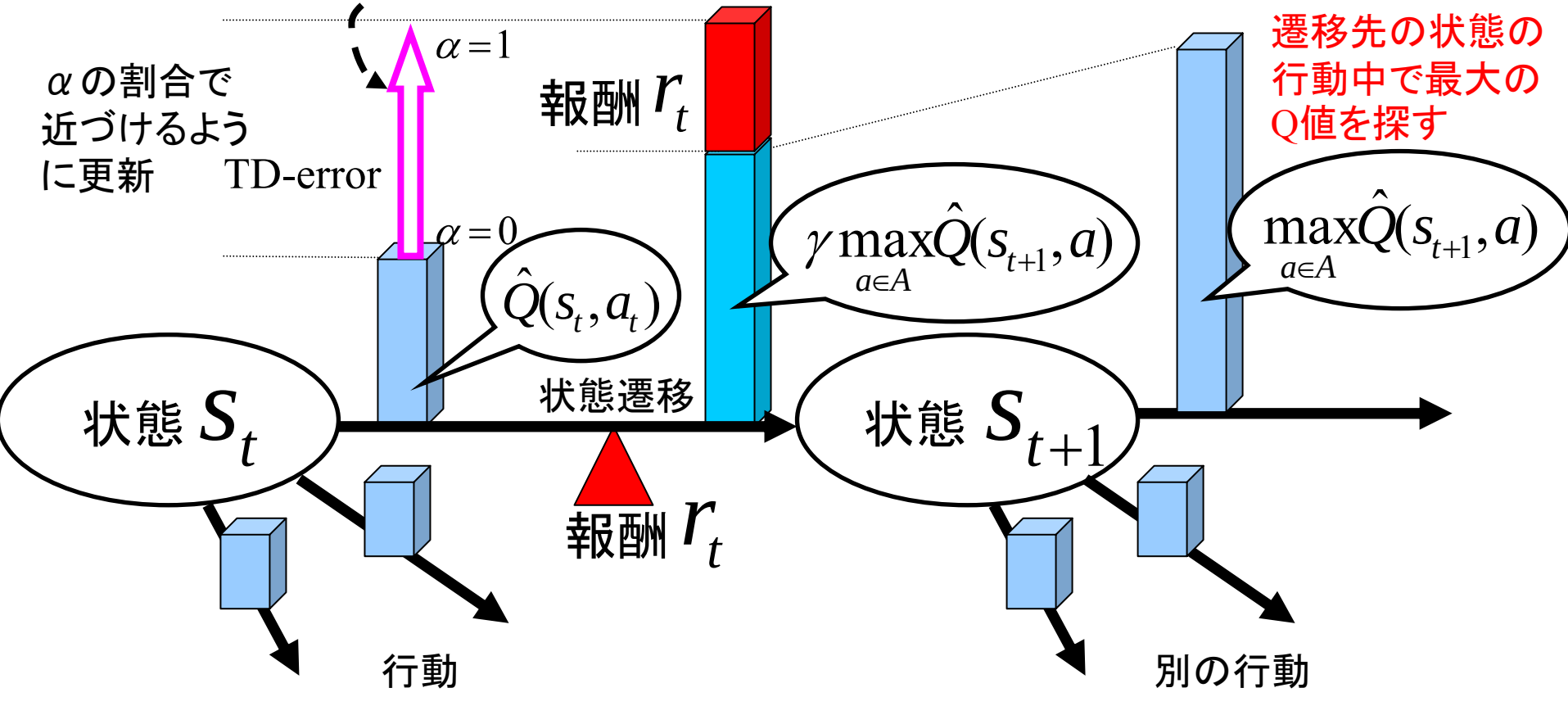
ボルツマン行動選択

$$\hat{Q}(s_t,a_t) \leftarrow \hat{Q}(s_t,a_t) + \alpha \left[r_t + \gamma \max_{a \in A} \hat{Q}(s_{t+1},a) - \hat{Q}(s_t,a_t) \right]$$

状態Stで実行した行動のQ値を更新

γ は割引率（$0 \leqq \gamma \leqq 1$）

α は学習率（$0 < \alpha \leqq 1$）



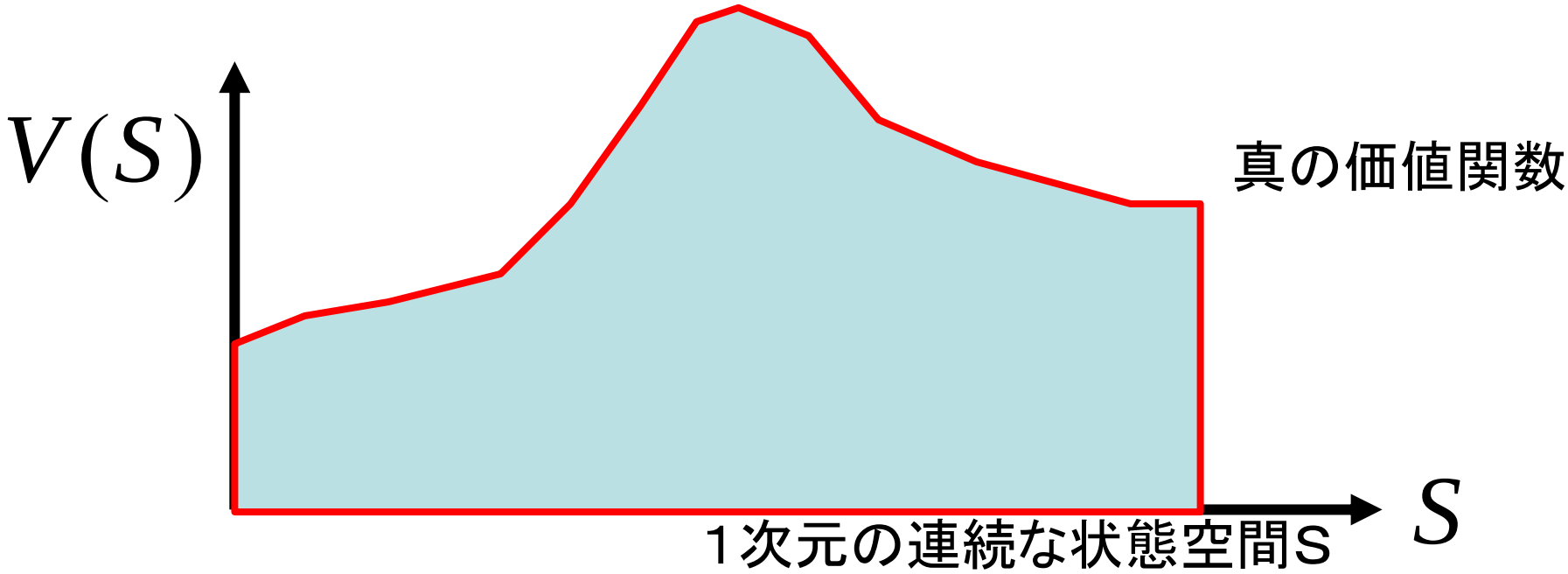
Q-learningの収束定理（Watkins92）

行動選択において全行動を十分な回数選択し，かつ学習率 α が

$$\sum_{t=0}^{\infty} \alpha'(t) \to \infty \qquad \text{and} \qquad \sum_{t=0}^{\infty} \alpha(t)^2 < \infty$$

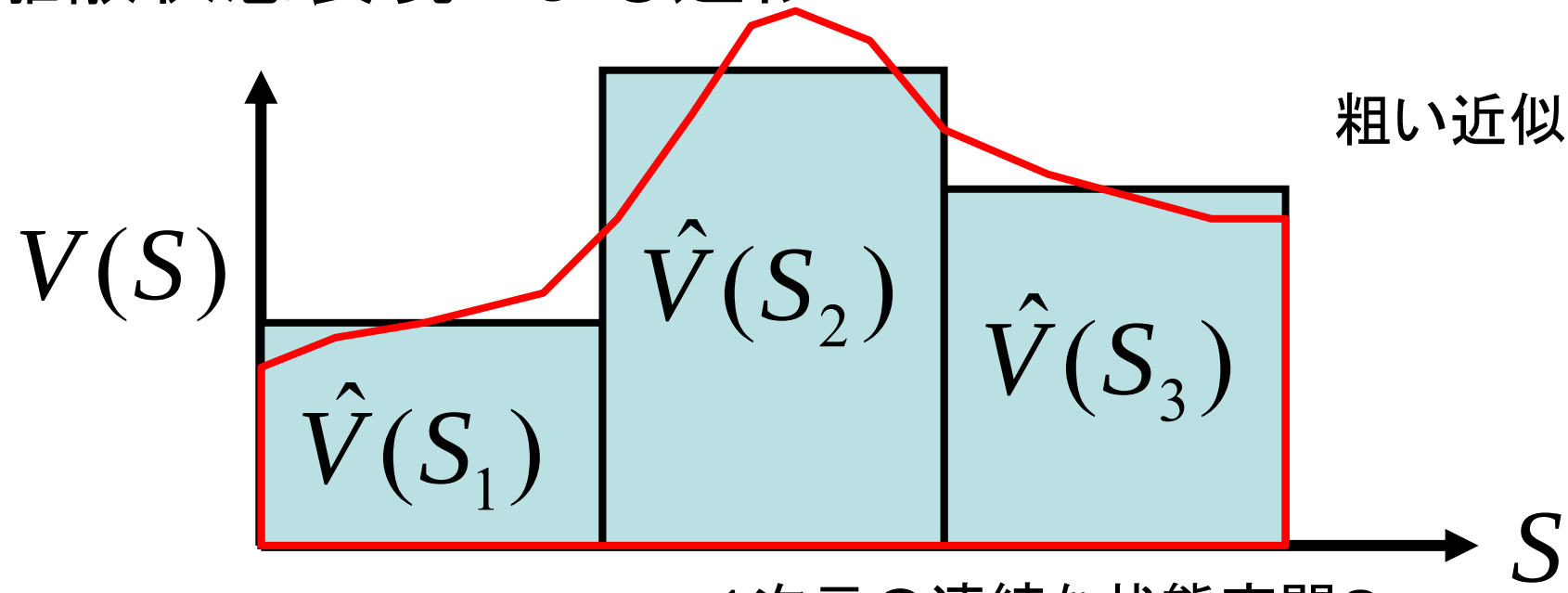
を満たす時間 t の関数になっているとき，アルゴリズムで得るQ値は確率1で最適政策の評価値に概収束する。
ただし環境はエルゴート性を有するMDP

Q-learning | **連続空間の関数近似** | ボルツマン行動選択



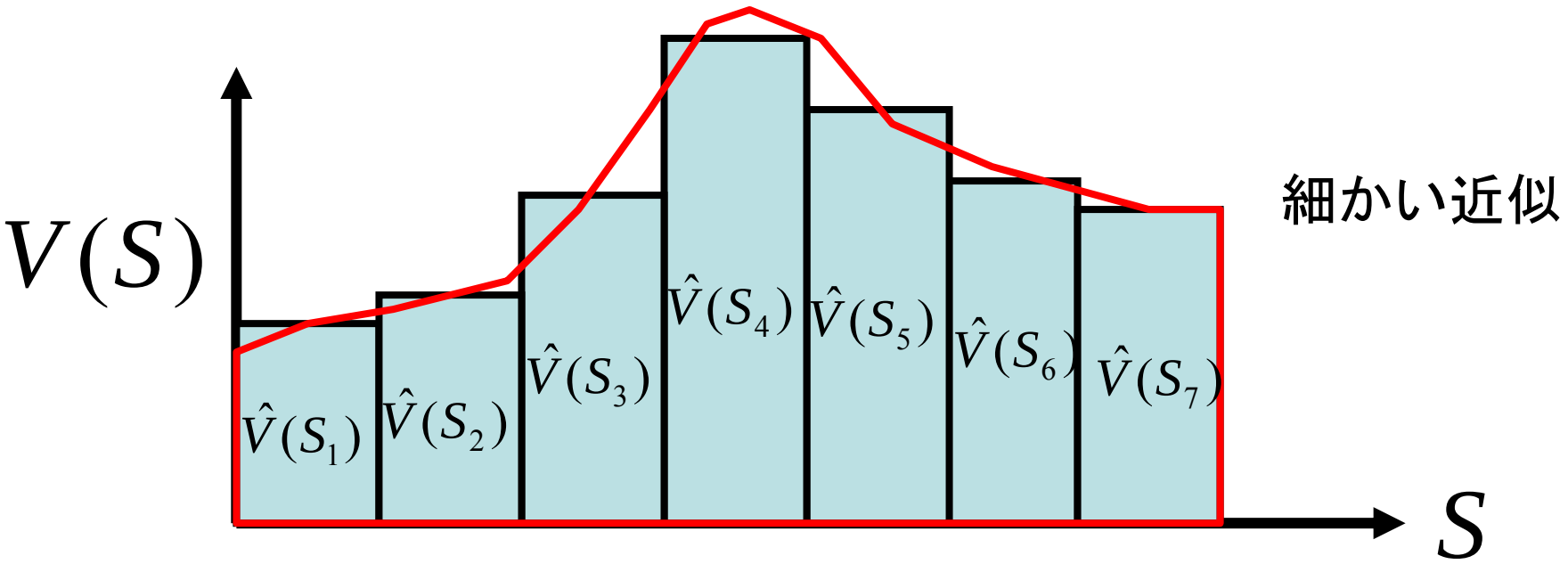
このような離散ではない状態空間のvalueを学習するには？

離散状態表現による近似



粗い近似

1次元の連続な状態空間S

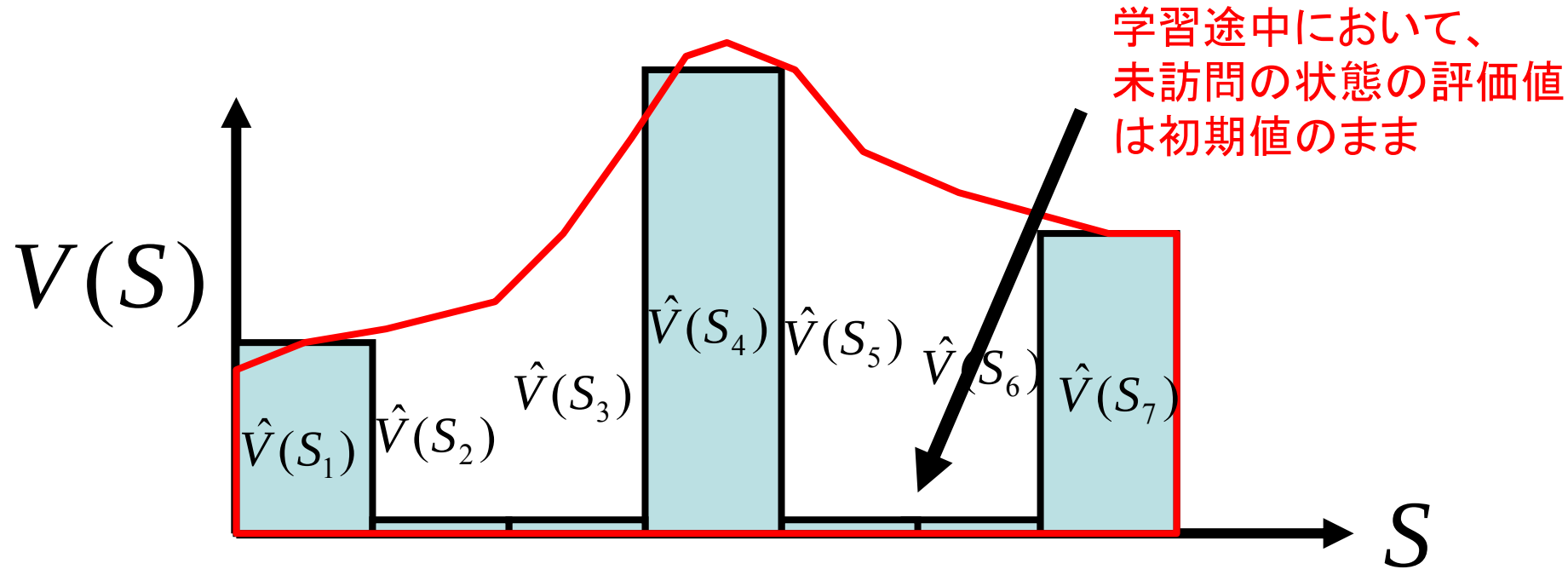


細かい近似

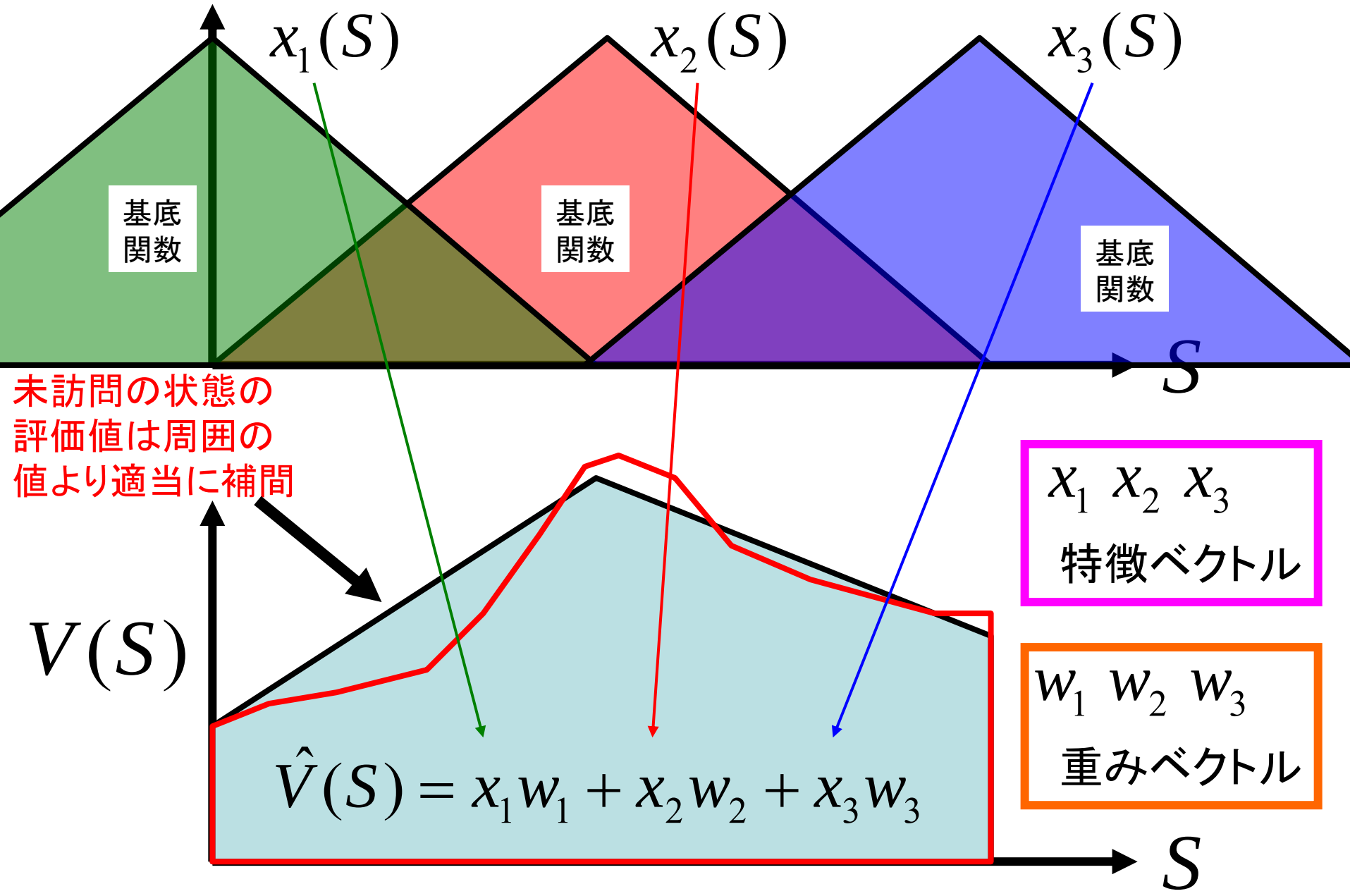
離散状態表現による近似の問題点

粗い近似：
- ●価値関数の学習は早い
- ●非マルコフ性が生じる　→　最適政策が得られない

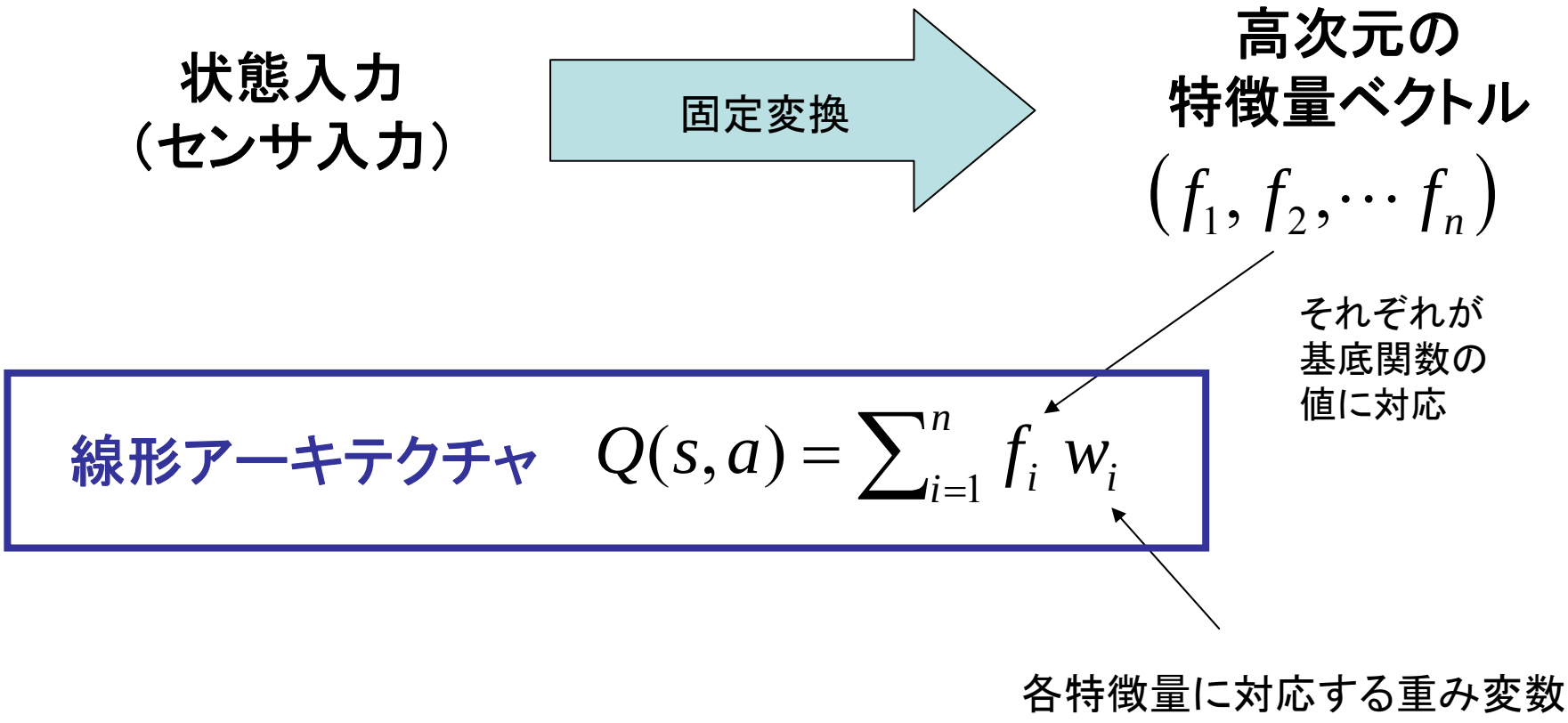
細かい近似
- ●価値関数の学習は指数関数的に遅くなる
- ●特に高次元環境において膨大なメモリーが必要



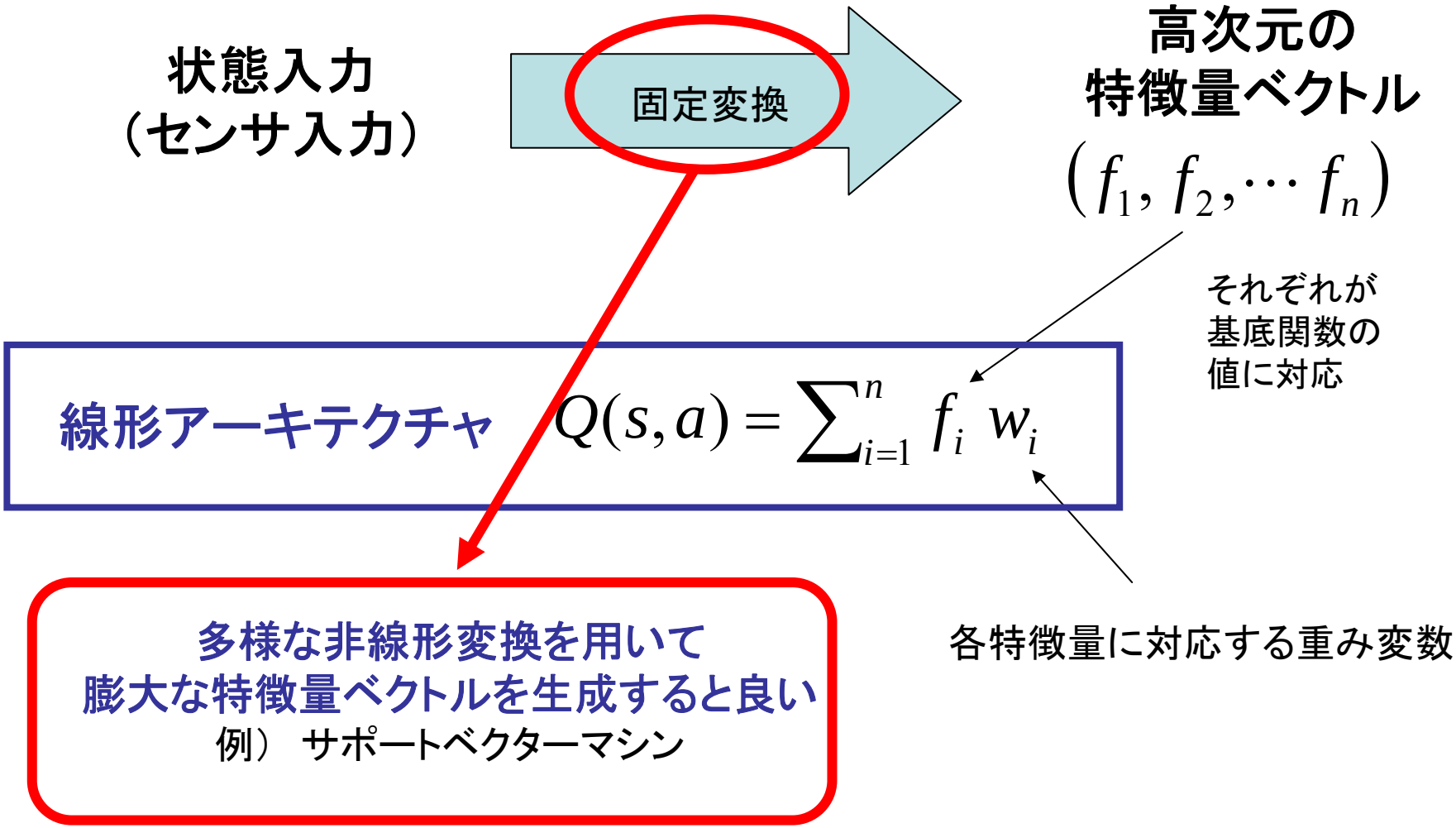
線形アーキテクチャによる汎化と関数近似



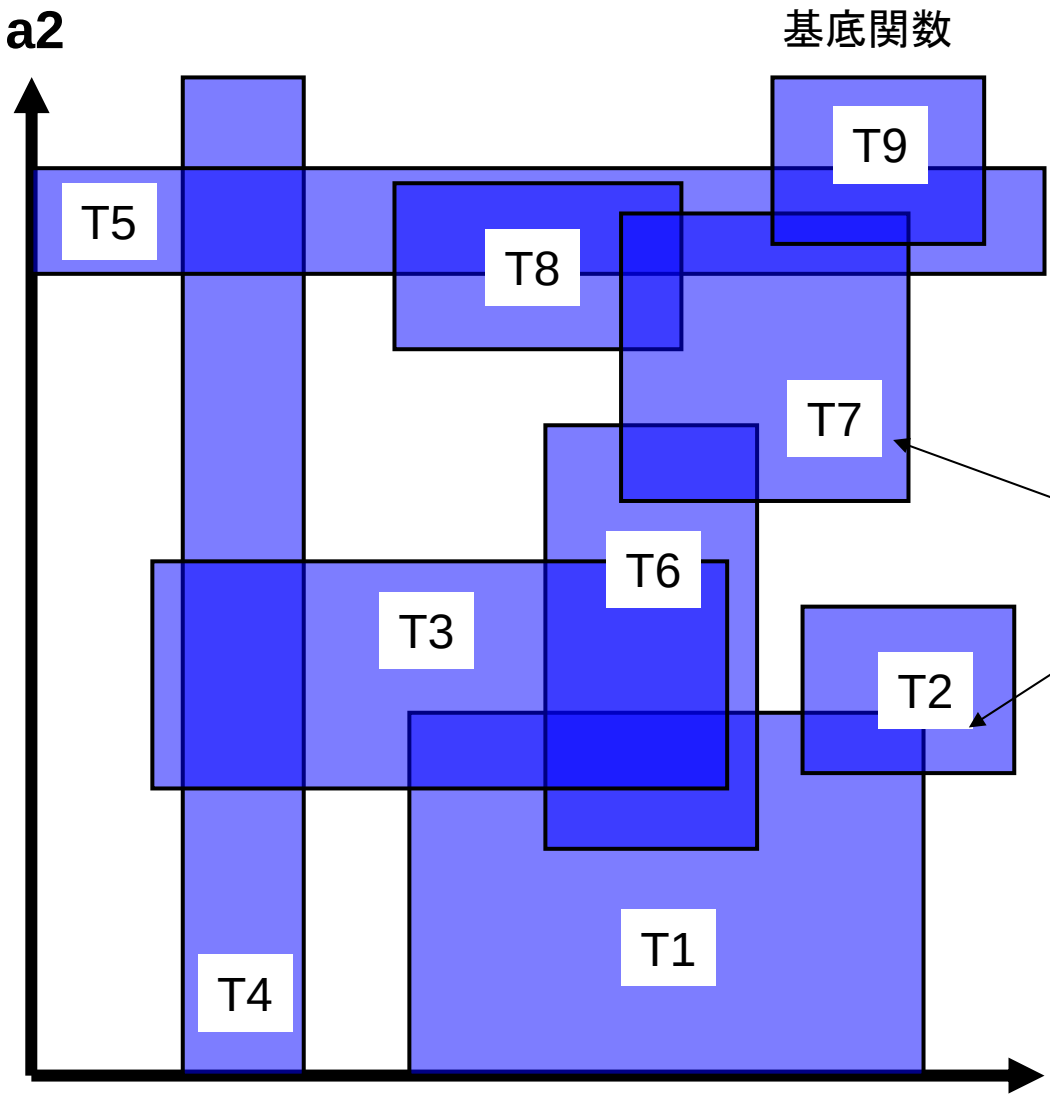
高次元状態・行動空間の汎化：
ランダムタイリングによるQ関数表現



高次元状態・行動空間の汎化：
ランダムタイリングによるQ関数表現



高次元状態・行動空間の汎化：
ランダムタイリングによるQ関数表現



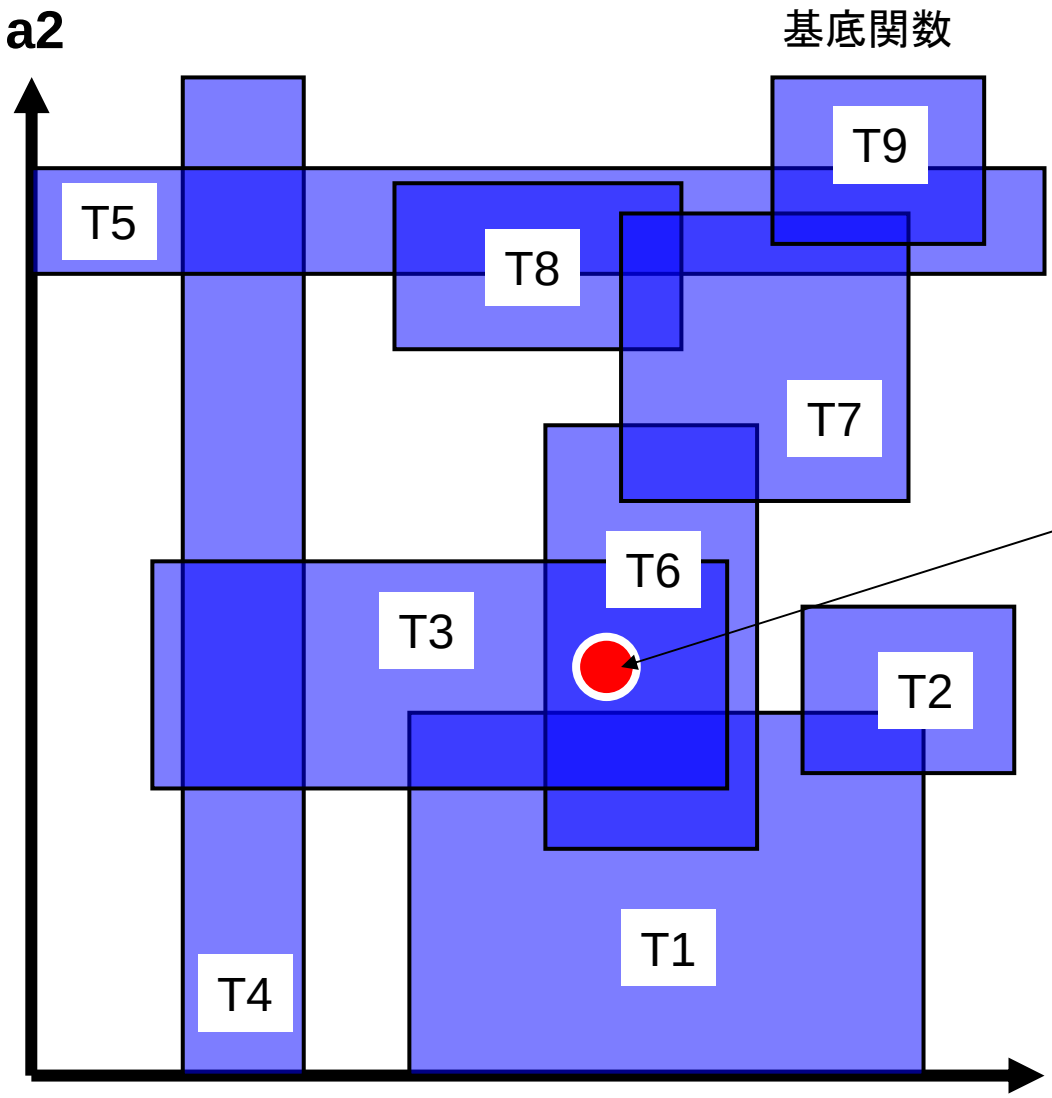
様々な大きさ・次元の超矩形タイルをランダムに配置して特徴量ベクトルを生成

各タイルが特徴量ベクトルの各要素に対応

$$(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9)$$

実験では、各軸あたり0.3～0.6の確率で区域を設定

高次元状態・行動空間の汎化：
ランダムタイリングによるQ関数表現

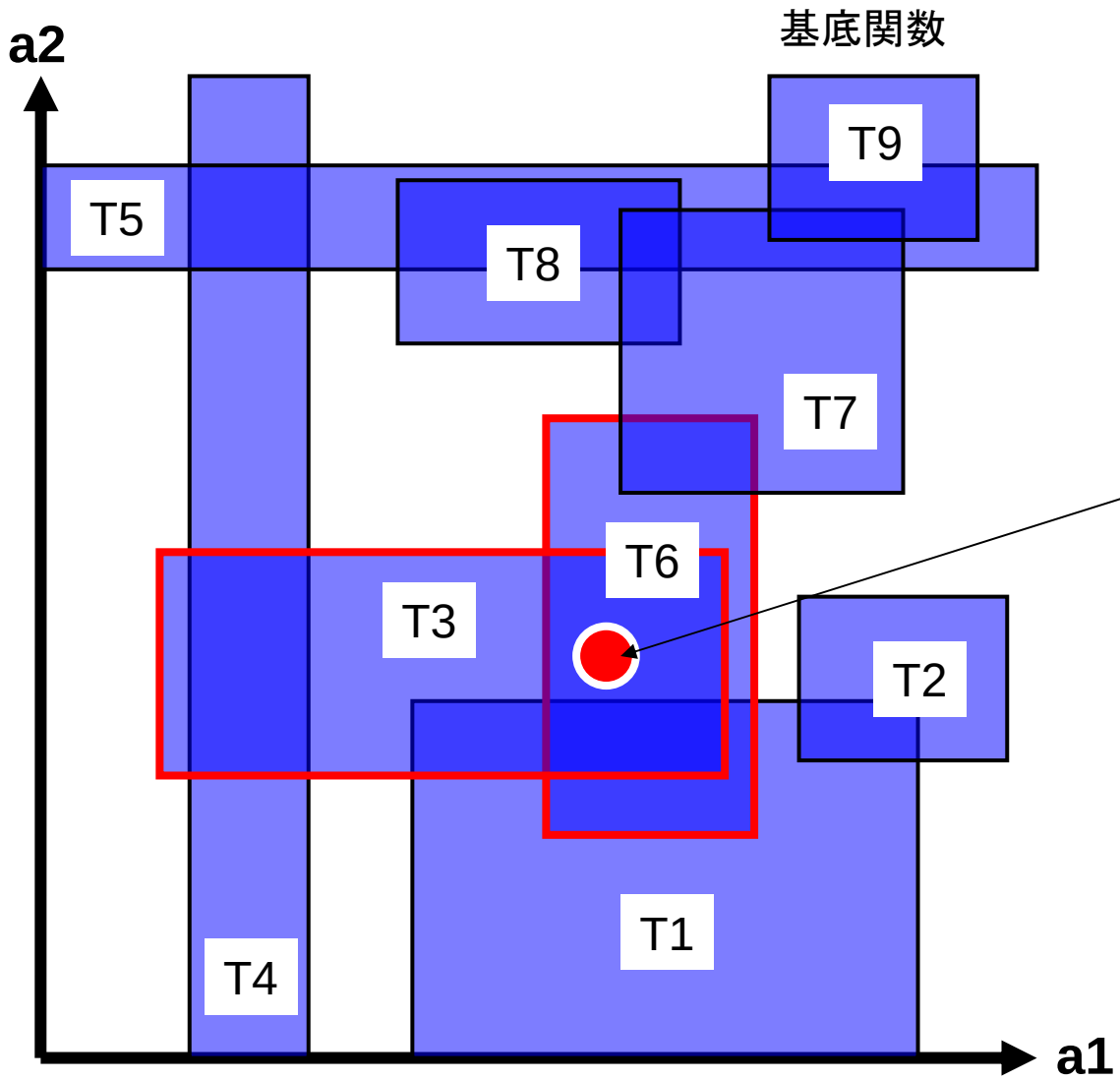


様々な大きさ・次元の超矩形タイルをランダムに配置して特徴量ベクトルを生成

ある値が入力されたとき

$$(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9)$$

高次元状態・行動空間の汎化：
ランダムタイリングによるQ関数表現



様々な大きさ・次元の超矩形タイルをランダムに配置して特徴量ベクトルを生成

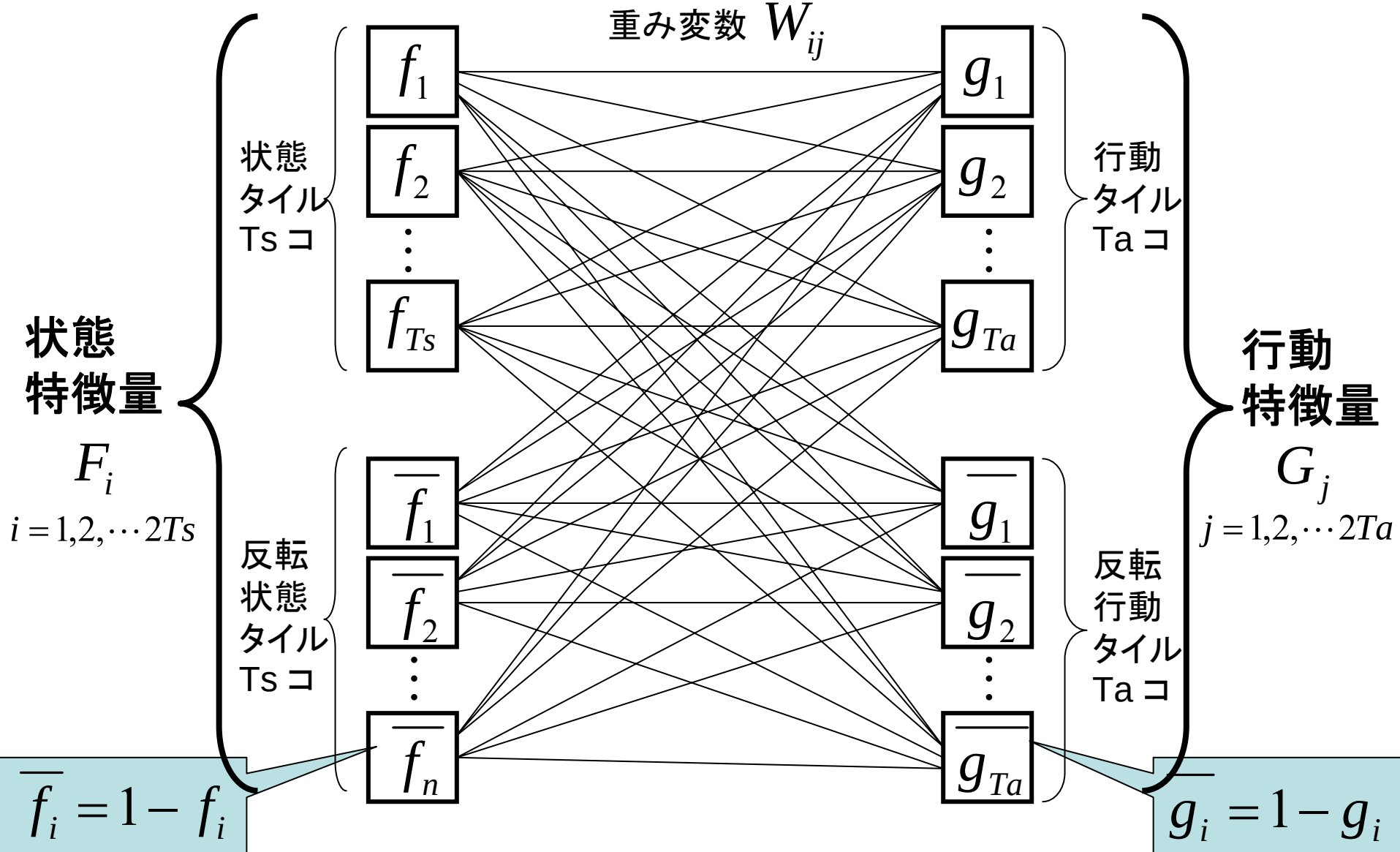
ある値が入力されたとき

$$(0, 0, 1, 0, 0, 1, 0, 0, 0)$$

該当するタイルの特徴量のみ1

ある状態 s, 行動 a
におけるQ値の求め方

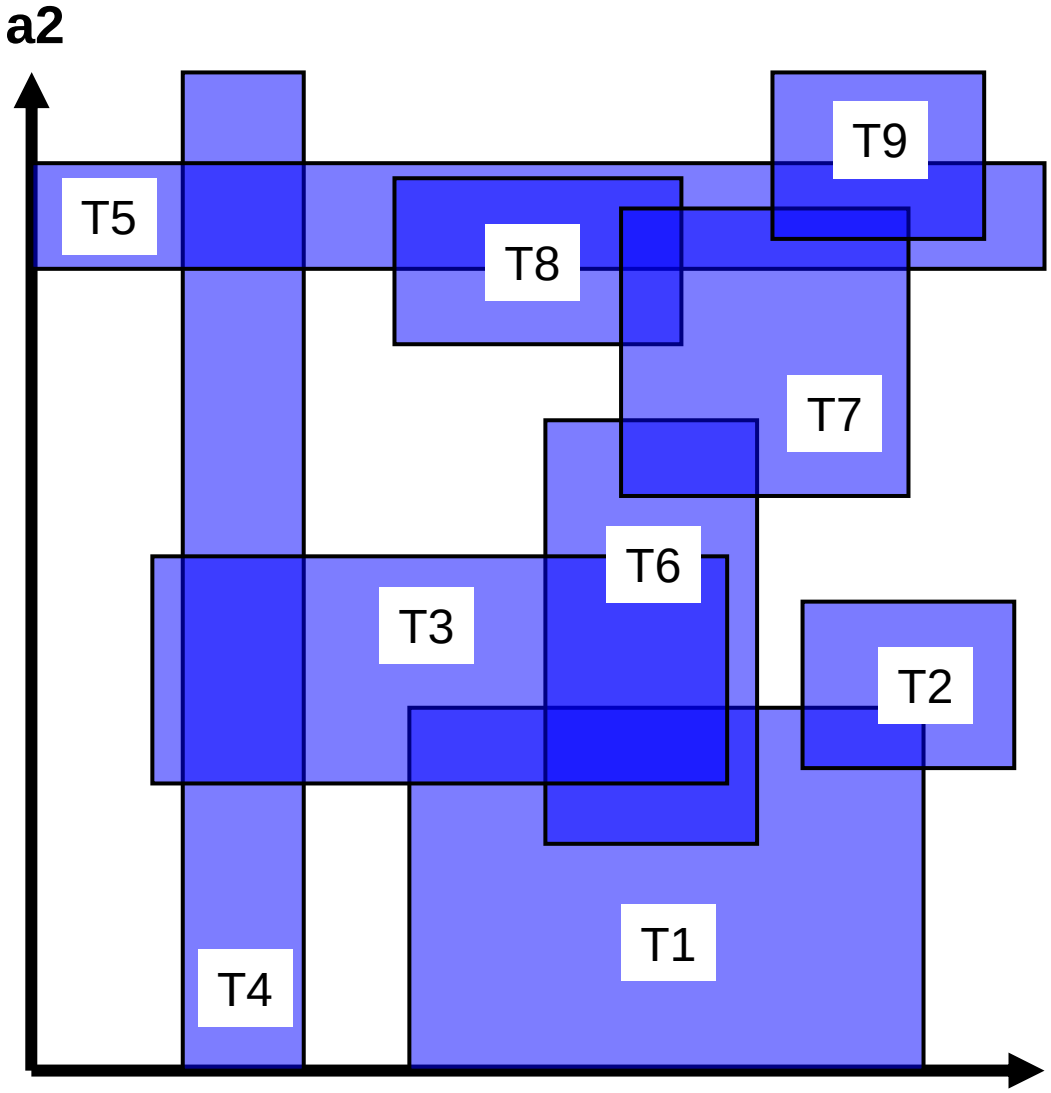
$$Q(s,a) = \frac{1}{Ts\,Ta} \sum_{j=1}^{2Ta} \sum_{i=1}^{2Ts} F_i\, G_j\, W_{ij}$$



Q-learning

連続空間の関数近似

ボルツマン行動選択



ランダムタイリングによる高次元連続空間の汎化

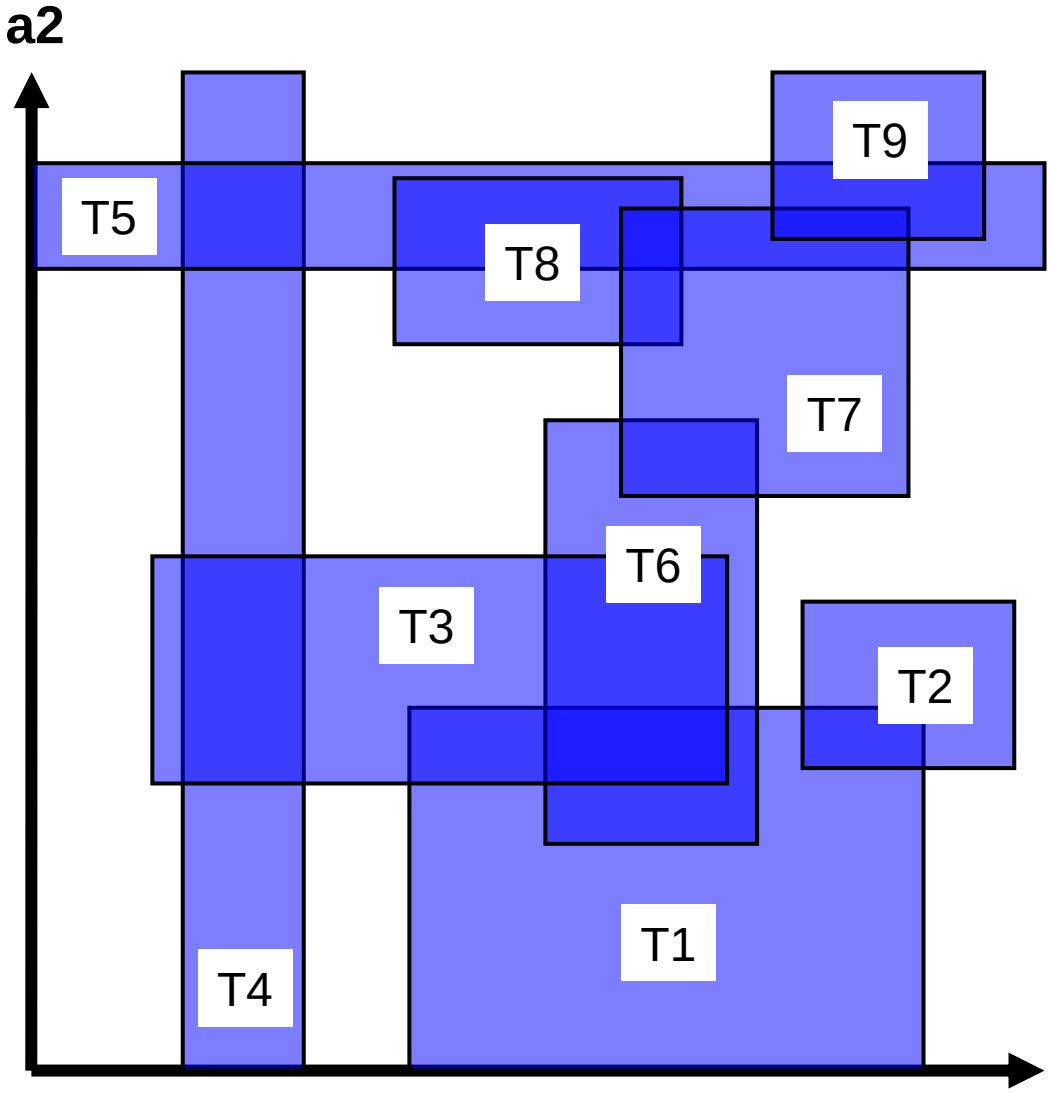
特長

- 実装が簡単
- 空間爆発を回避
- 足りなければタイルを増やすだけ
- 実装が問題に依存しない

Q-learning

連続空間の関数近似

ボルツマン行動選択

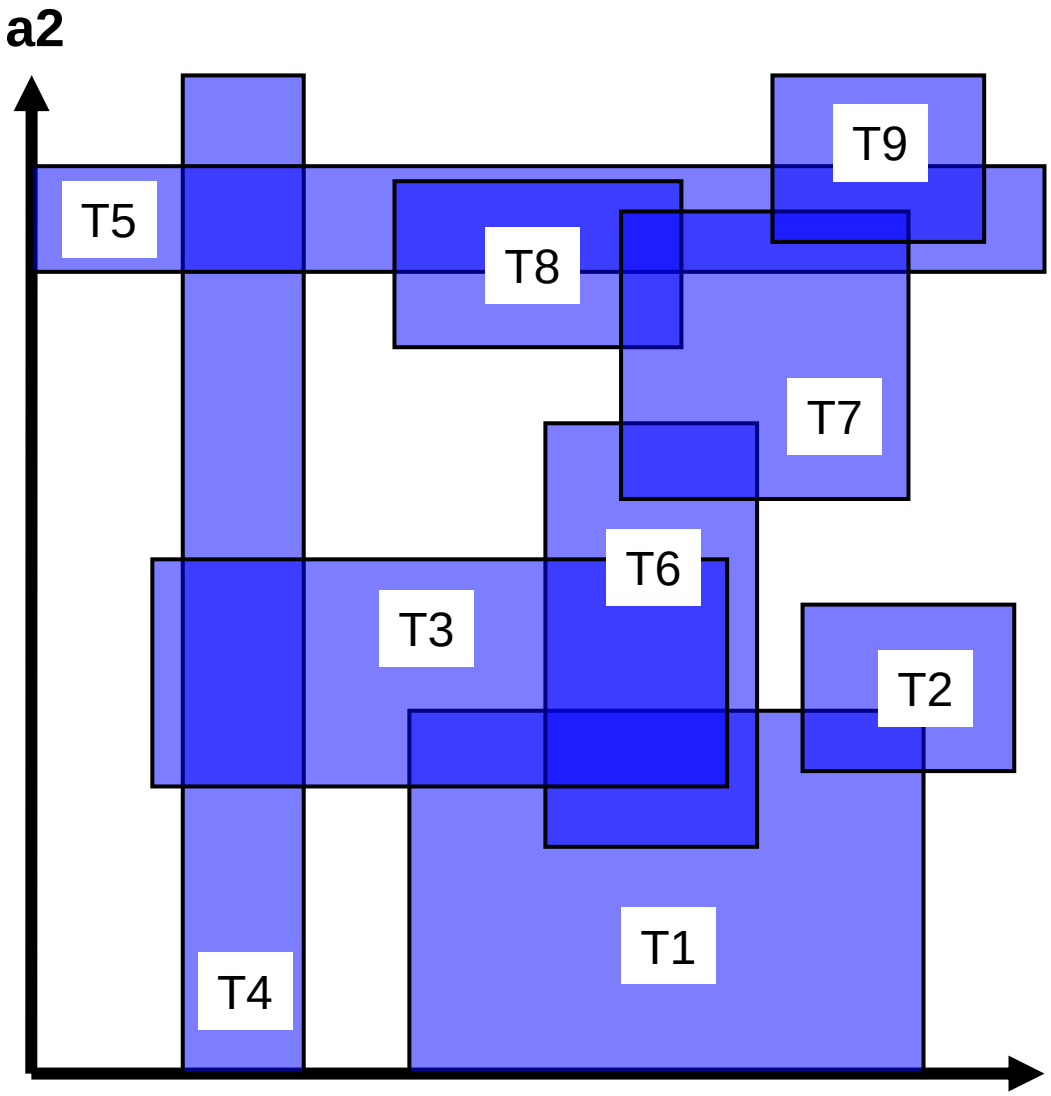


ランダムタイリングによる高次元連続空間の汎化

特長

- 実装が簡単
- 空間爆発を回避
- 足りなければタイルを増やすだけ
- 実装が問題に依存しない

Q-learning | **連続空間の関数近似** | ボルツマン行動選択



ランダム矩形タイルの改善

- エピソードデータにおいて一度も反応しないタイルを新しく作り直す（スコアゼロ）
- エピソードデータに対してスコアの低いタイルを新しく作り直す

Q-learning | **連続空間の関数近似** | ボルツマン行動選択

矩形タイルのスコア

$$H(f_i) = -p(f_i, D)\log p(f_i, D)\frac{1}{\text{矩形}f_i\text{の体積}}$$

エピソードデータに対する矩形タイル反応の情報量（エントロピー）

体積の小さいタイルほど高く評価

データに対し、50%の確率で反応すると最大値

Q-learning

連続空間の関数近似

ボルツマン行動選択

ある状態Sにおいて**Q値のボルツマン分布**に従って
確率的に行動aを選ぶ

行動選択確率:

$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

しかし！
全行動空間の exp(Q) の合計が必要！

Q-learning

連続空間の関数近似

ボルツマン行動選択

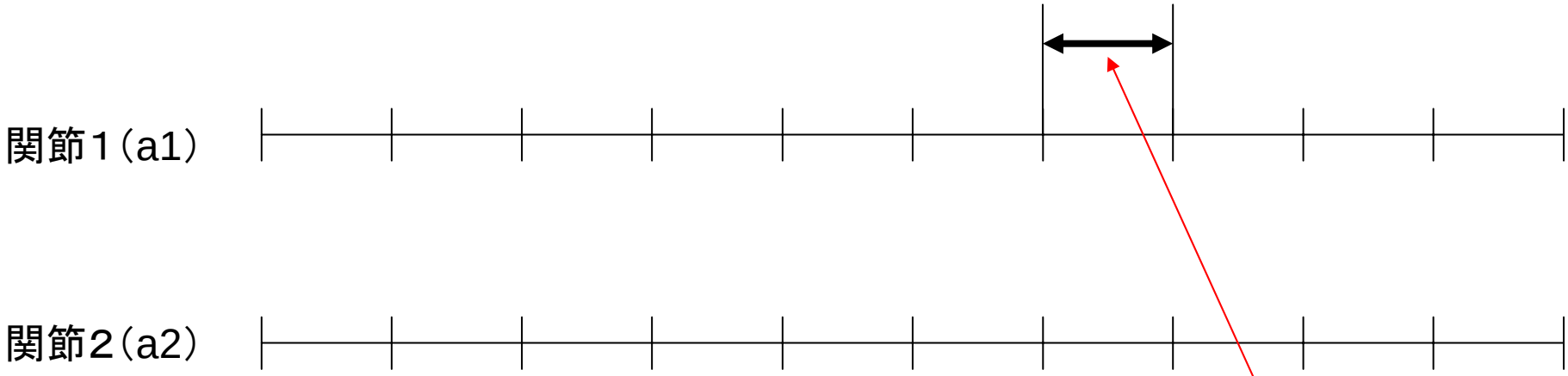
ある状態Sにおいて**Q**値の**ボルツマン分布**に従って
確率的に行動aを選ぶ

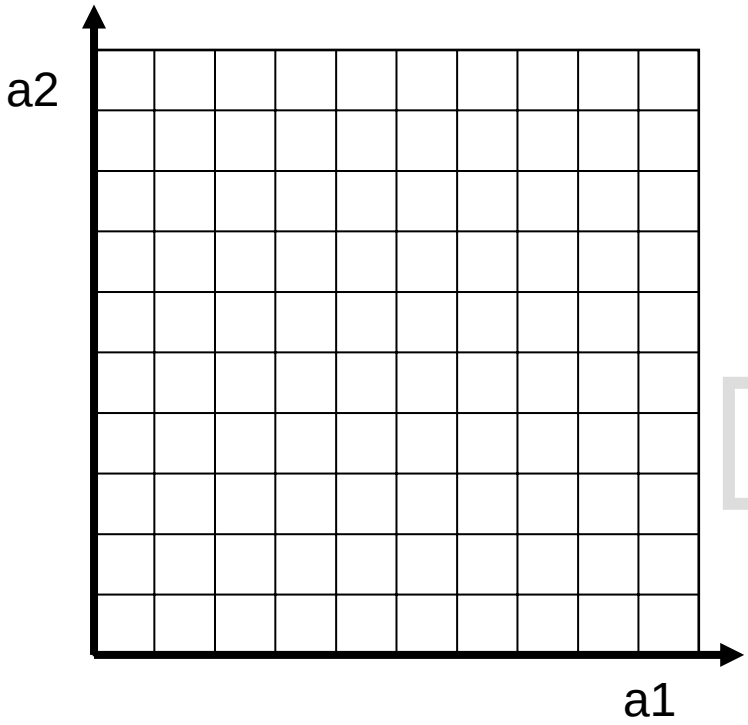
行動選択確率:

$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

しかし！
全行動空間の exp(Q) の合計が必要！

高次元行動空間の扱いの難しさ：行動選択



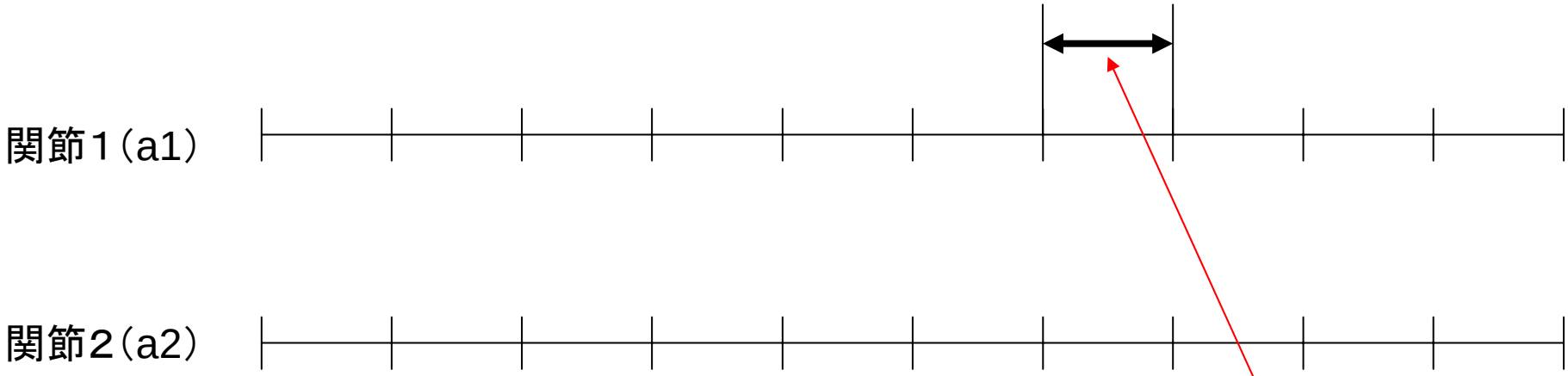


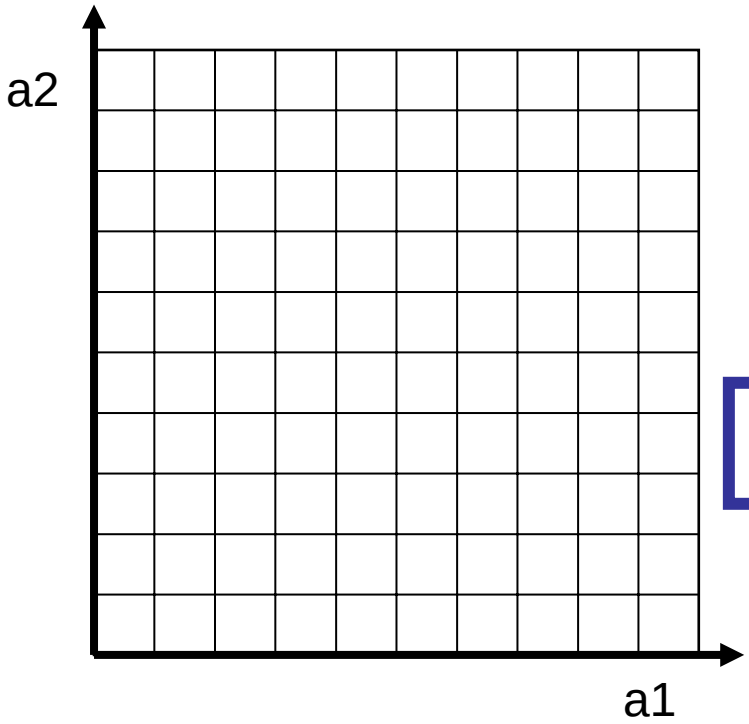
行動空間の各次元を10分割で離散化

8関節 → 離散行動 1000万通り

Q-learningにおける行動選択やQ値の更新の際に
最大のQ値を探す計算に多大な計算コストを要する
従来の計算コストは「次元の呪い」に直面

高次元行動空間の扱いの難しさ：行動選択



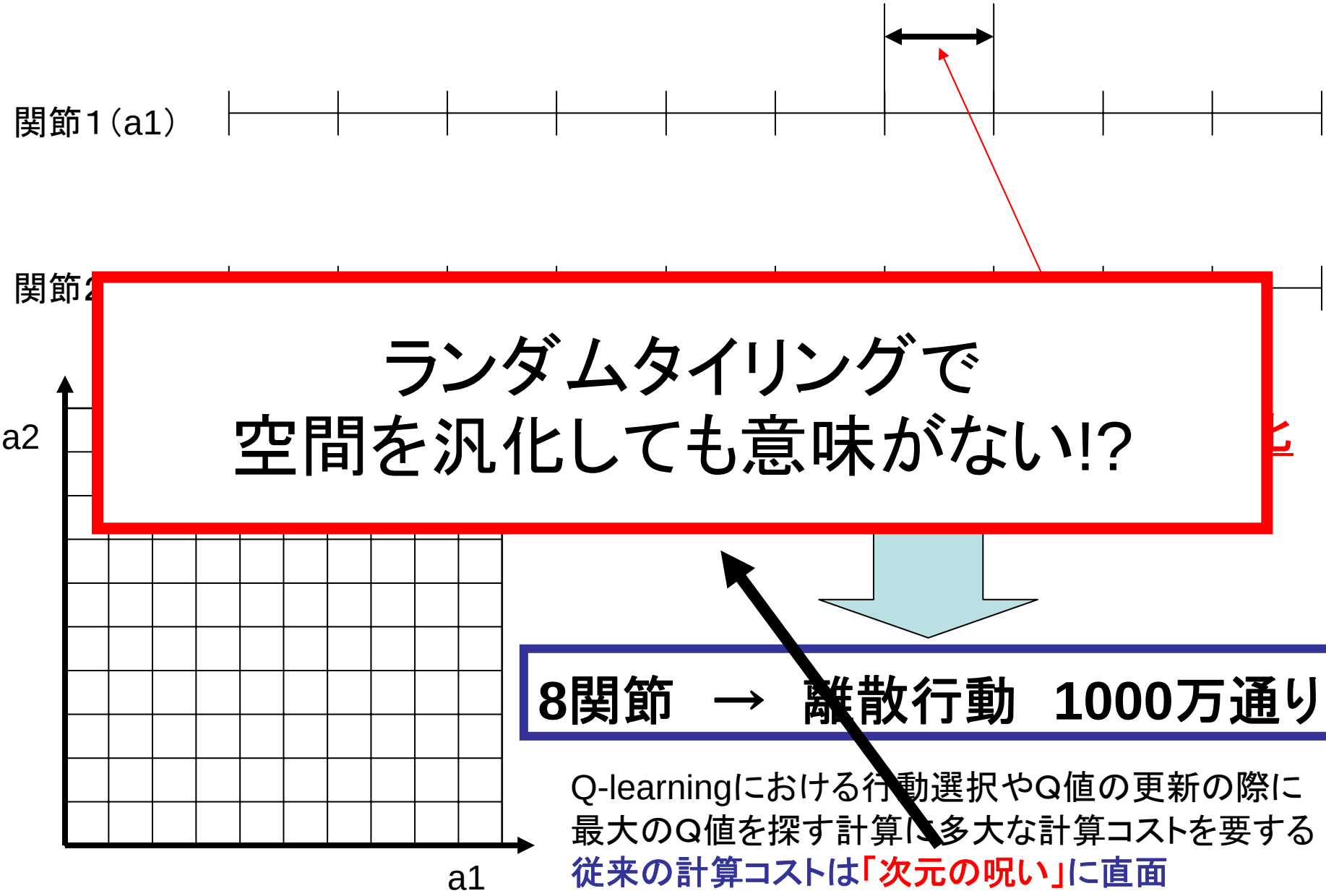


行動空間の各次元を**10分割で離散化**

8関節　→　離散行動　1000万通り

Q-learningにおける行動選択やQ値の更新の際に
最大のQ値を探す計算に多大な計算コストを要する
従来の計算コストは「次元の呪い」に直面

高次元行動空間の扱いの難しさ：行動選択



ランダムタイリングで
空間を汎化しても意味がない!?

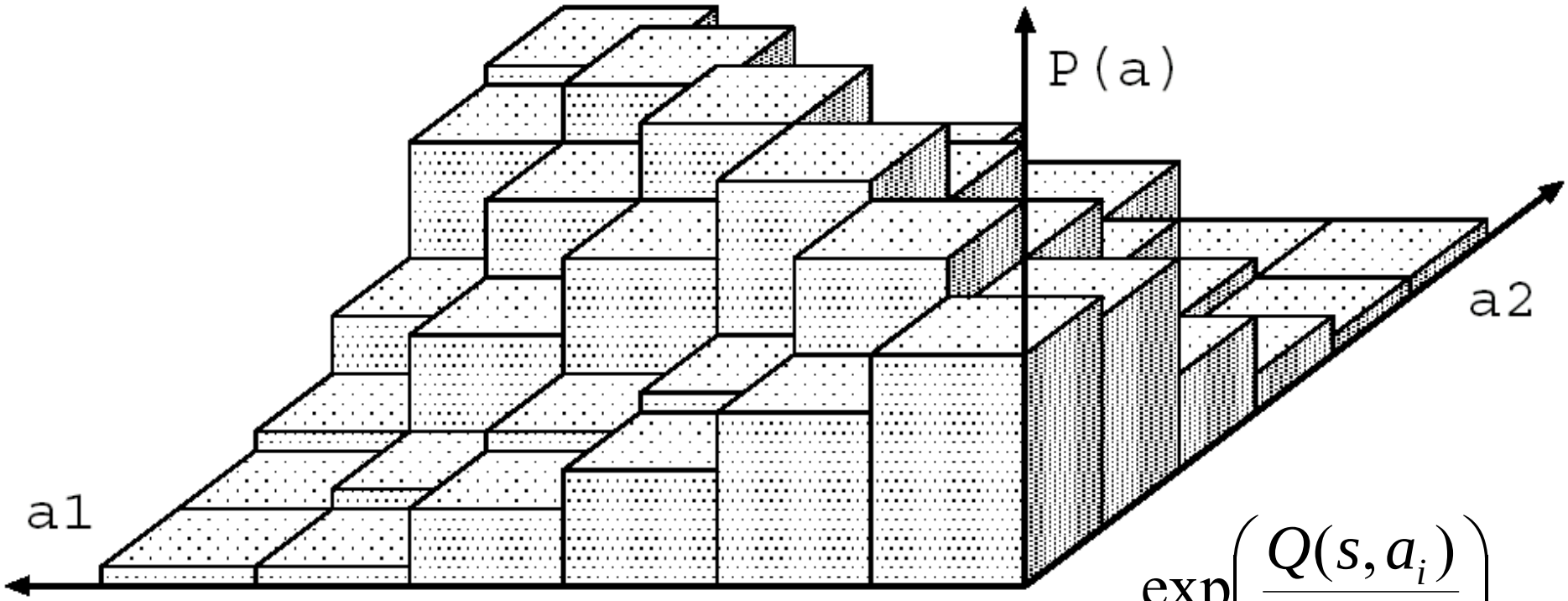
8関節　→　離散行動　1000万通り

Q-learningにおける行動選択やQ値の更新の際に
最大のQ値を探す計算に多大な計算コストを要する
従来の計算コストは「次元の呪い」に直面

Q-learning

連続空間の関数近似

ボルツマン行動選択

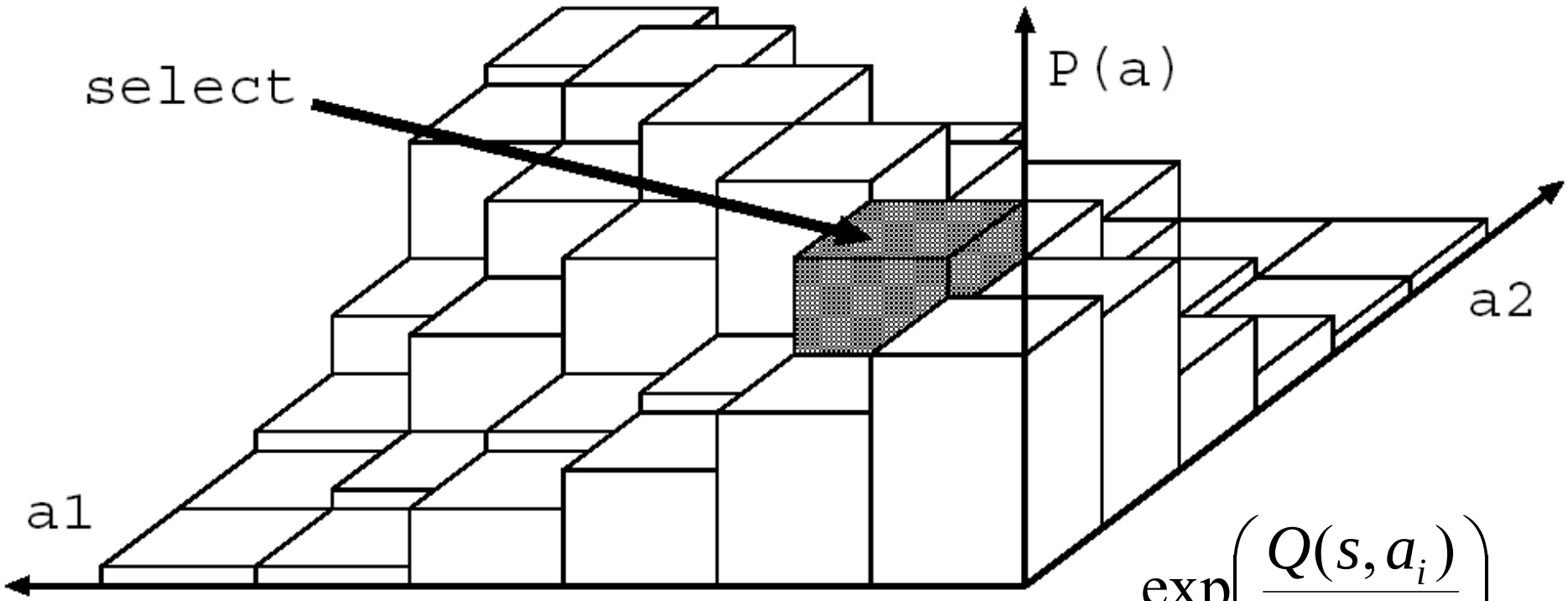


$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

Q-learning

連続空間の関数近似

ボルツマン行動選択



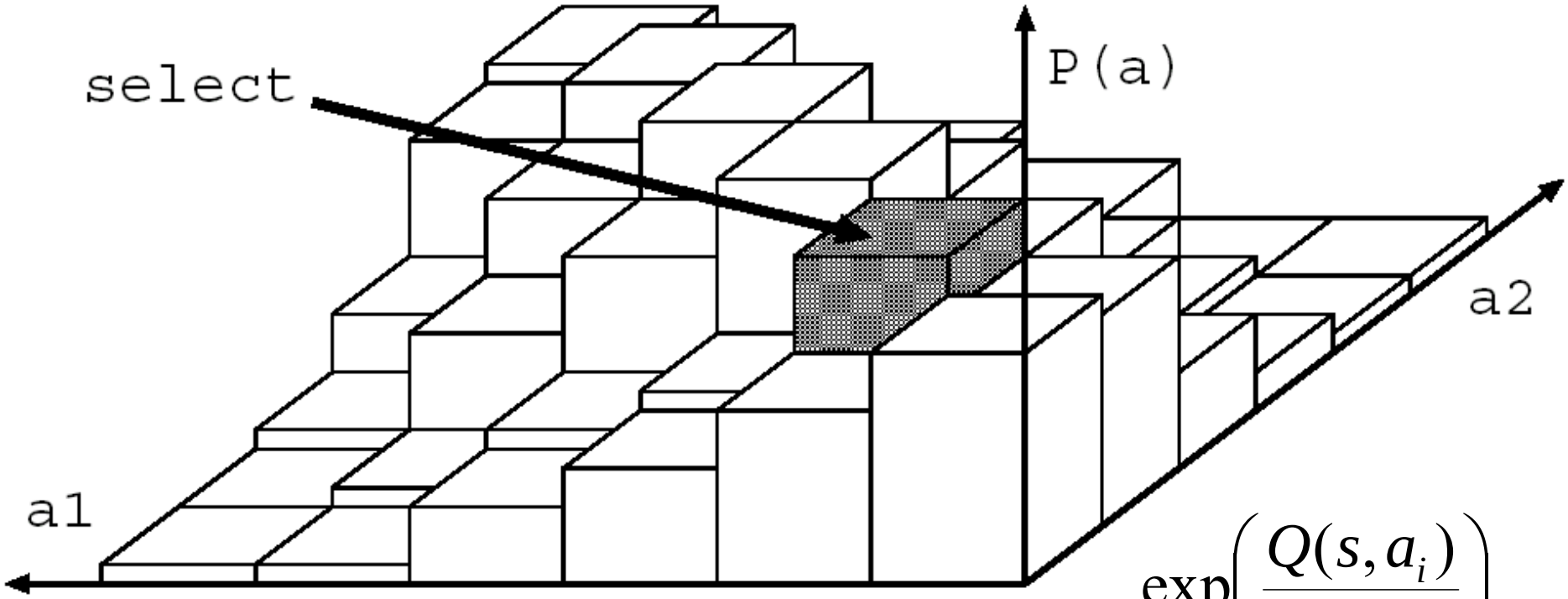
$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

全行動空間の exp(Q)の合計が必要！

Q-learning

連続空間の関数近似

ボルツマン行動選択



$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

全行動空間の exp(Q)
の合計が必要！

Q-learning

連続空間の関数近似

ボルツマン行動選択

高次元空間における確率分布に従ったサンプルを効率よく得るには？　→　統計学的な一般問題

Markov chain Monte-Carlo（MCMC）法の一種

Gibbsサンプリング

Q-learning | 連続空間の関数近似 | ボルツマン行動選択

高次元空間における確率分布に従ったサンプルを効率よく得るには？　→　統計学的な一般問題

Markov chain Monte-Carlo（MCMC）法の一種

Gibbsサンプリング

計算が簡単な1次元の条件付確率分布

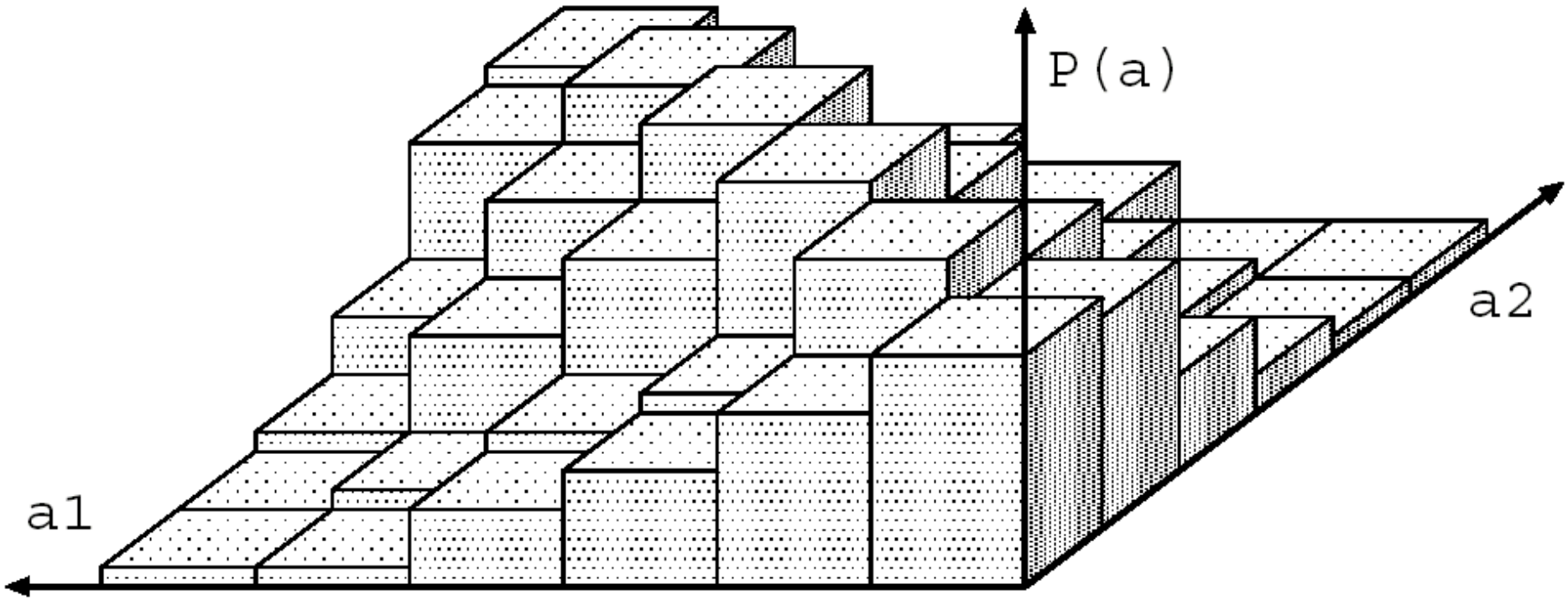
$$a^1(t+1) \approx P(a^1 \mid a^2(t), a^3(t), \cdots a^N(t))$$
$$a^2(t+1) \approx P(a^2 \mid a^1(t), a^3(t), \cdots a^N(t))$$
$$\vdots$$
$$a^N(t+1) \approx P(a^N \mid a^1(t), a^2(t), \cdots a^{N-1}(t))$$

このような反復 t を十分な回数繰返し、最終的に得たa(t)をサンプルとする

Q-learning

連続空間の
関数近似

**ボルツマン
行動選択**

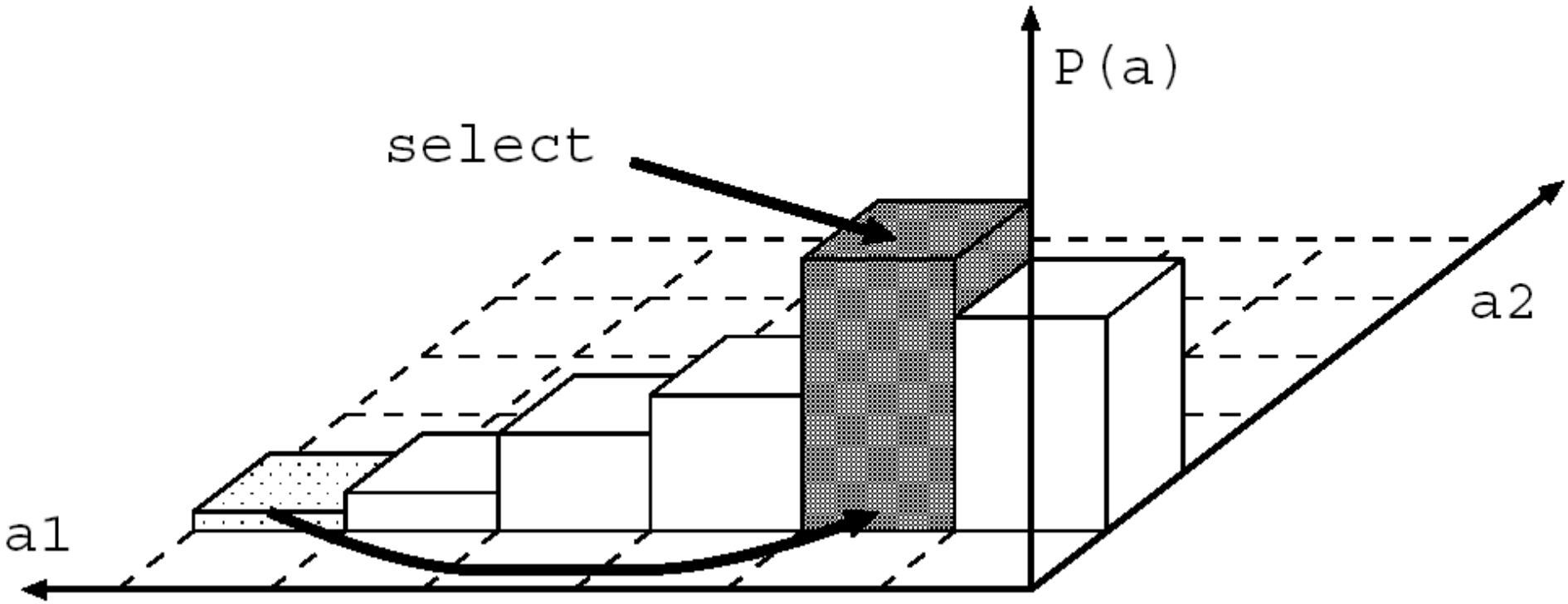


先ほどのボルツマン選択を Gibbsサンプリングしてみる

Q-learning

連続空間の関数近似

ボルツマン行動選択

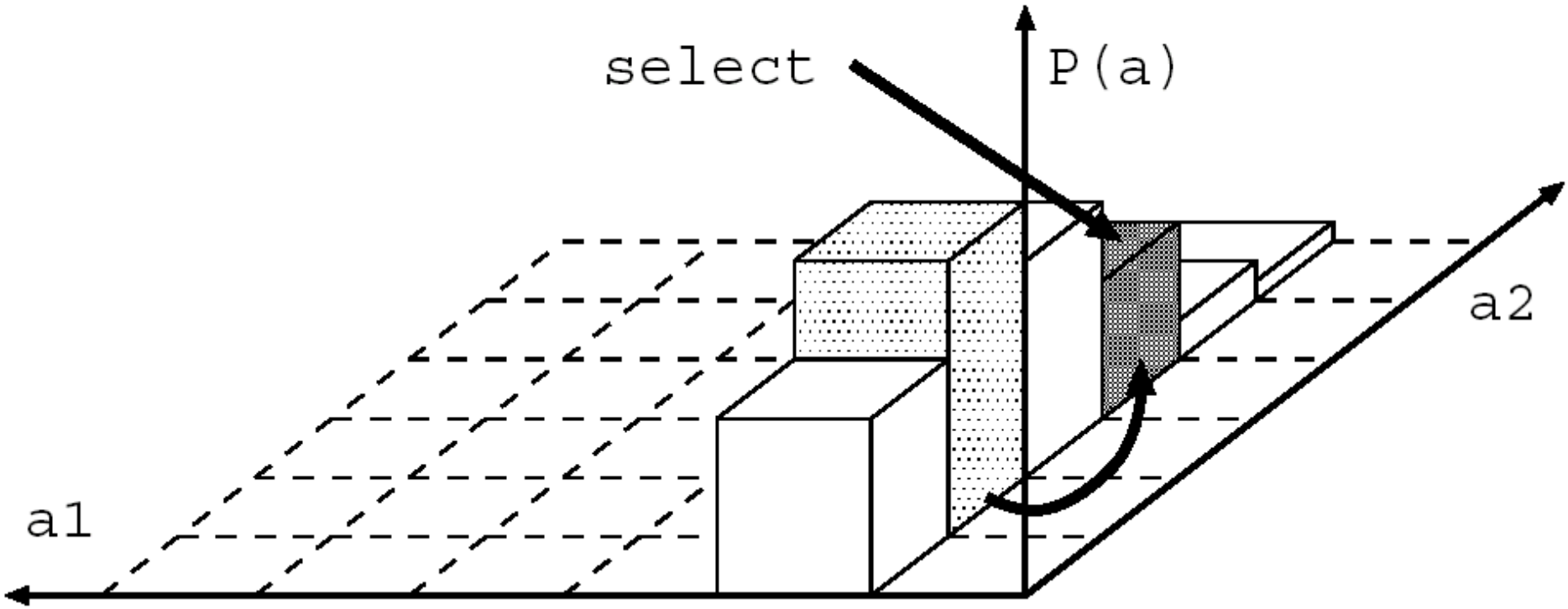


a2軸を固定し、a1軸についてのみボルツマン選択

Q-learning

連続空間の関数近似

ボルツマン行動選択

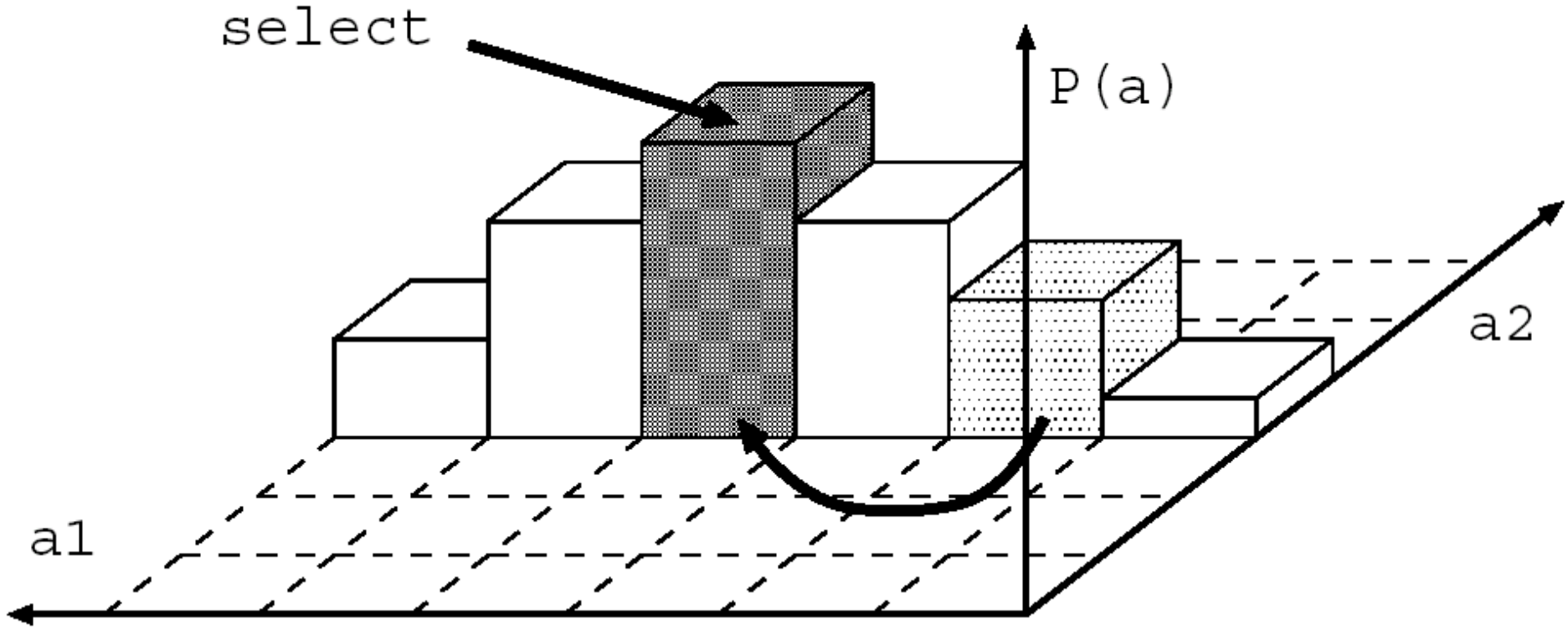


a1軸を先に選択された値に固定し、
今度はa2軸についてのみボルツマン選択

Q-learning

連続空間の
関数近似

**ボルツマン
行動選択**

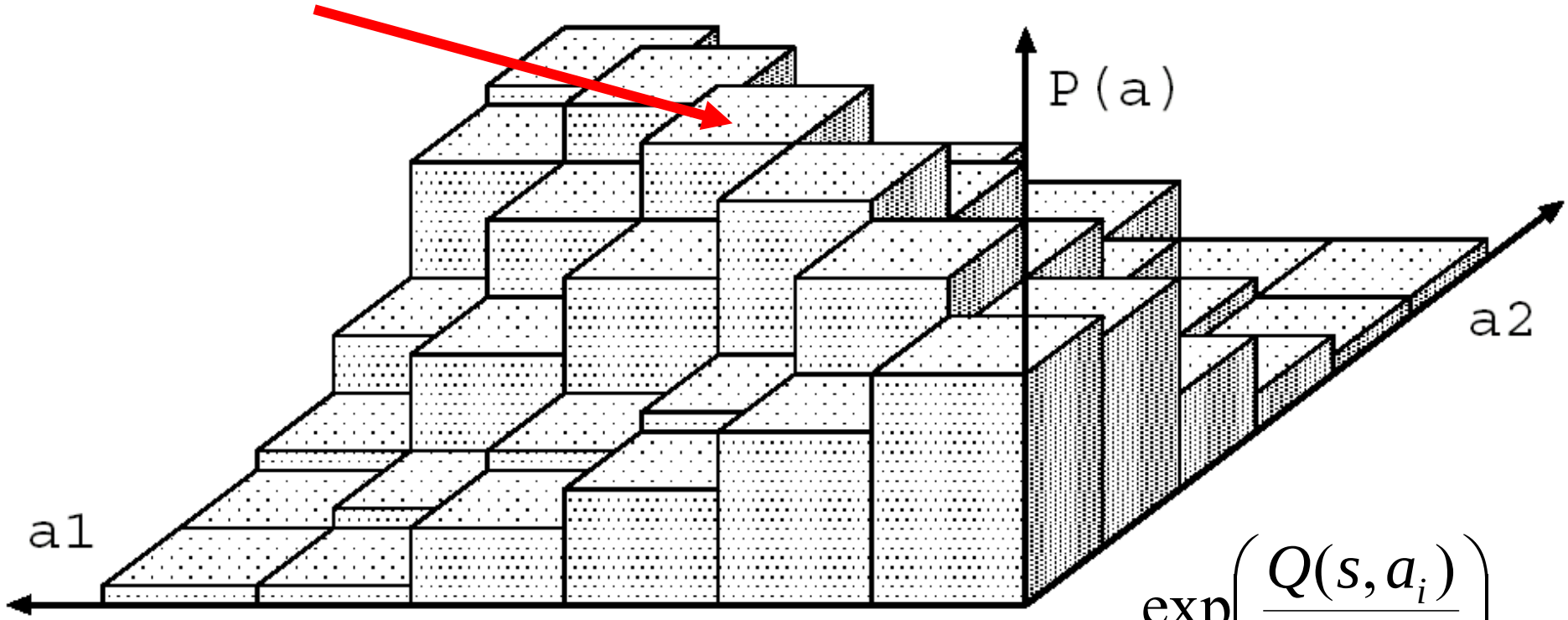


a2軸を先に選択された値に固定し、
再びa1軸についてのみボルツマン選択　これを繰返す

Q-learning

連続空間の関数近似

ボルツマン行動選択



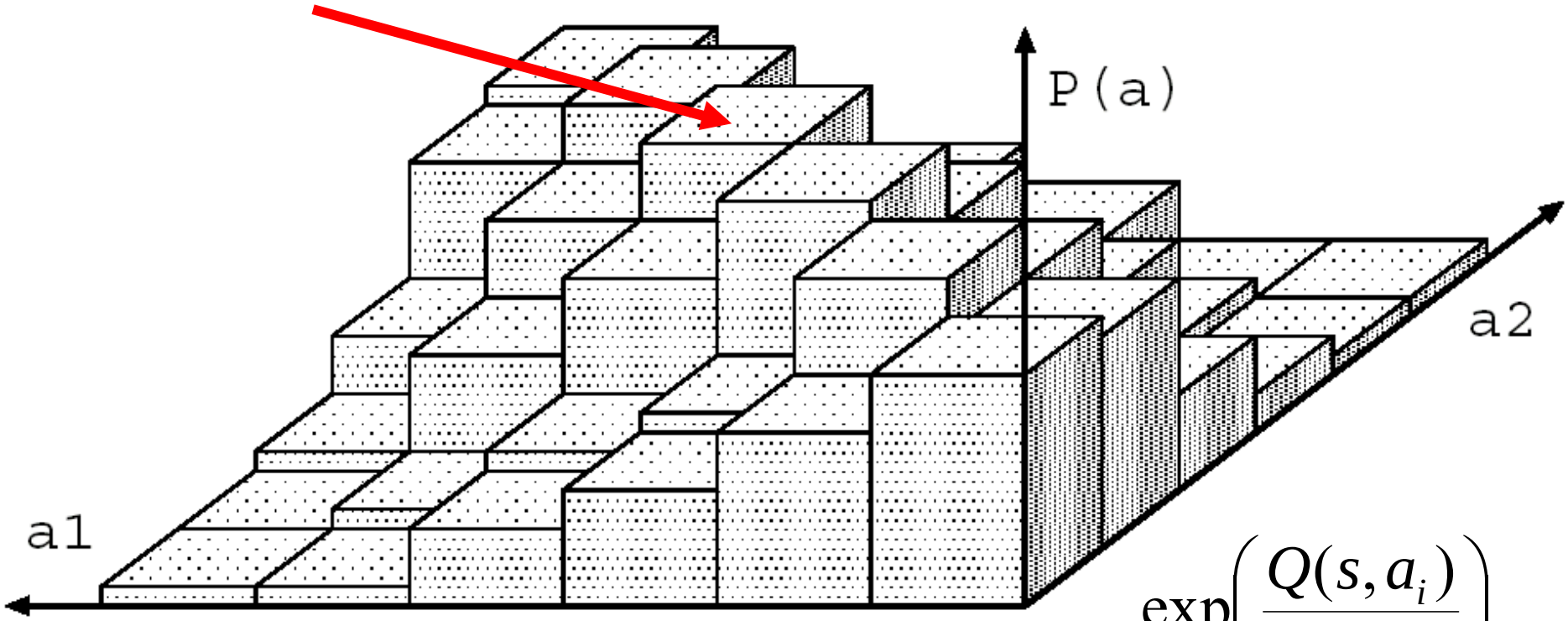
$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

Gibbsサンプリング：
全行動空間の exp(Q) の合計を
計算せずに右式に従うサンプルを得る！

Q-learning

連続空間の関数近似

ボルツマン行動選択



$$P(a_i) = \frac{\exp\left(\frac{Q(s,a_i)}{T}\right)}{\sum_{j=1}^{A} \exp\left(\frac{Q(s,a_j)}{T}\right)}$$

Gibbsサンプリング：

全行動空間の exp(Q) の合計を

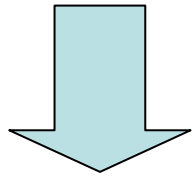
計算せずに右式に従うサンプルを得る！

Q-learning

連続空間の
関数近似

ボルツマン
行動選択

Gibbsサンプリングによる
行動選択処理



特長

・実装が簡単
・空間爆発を回避
・足りなければ計算反復の
回数を増やすだけ
・分割を細かくしても、計算
はあまり変わらない

ただし、
MaxQ を求める
計算が「近似」に

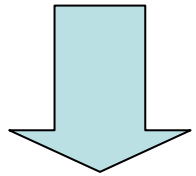
SARSAアルゴリズムなら問題ない

Q-learning

連続空間の
関数近似

ボルツマン
行動選択

Gibbsサンプリングによる
行動選択処理



特長

- 実装が簡単
- 空間爆発を回避
- 足りなければ計算反復の回数を増やすだけ
- 分割を細かくしても、計算はあまり変わらない

ただし、
MaxQ を求める
計算が「近似」に

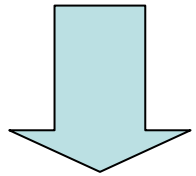
SARSAアルゴリズムなら問題ない

Q-learning

連続空間の
関数近似

ボルツマン
行動選択

Gibbsサンプリングによる
行動選択処理



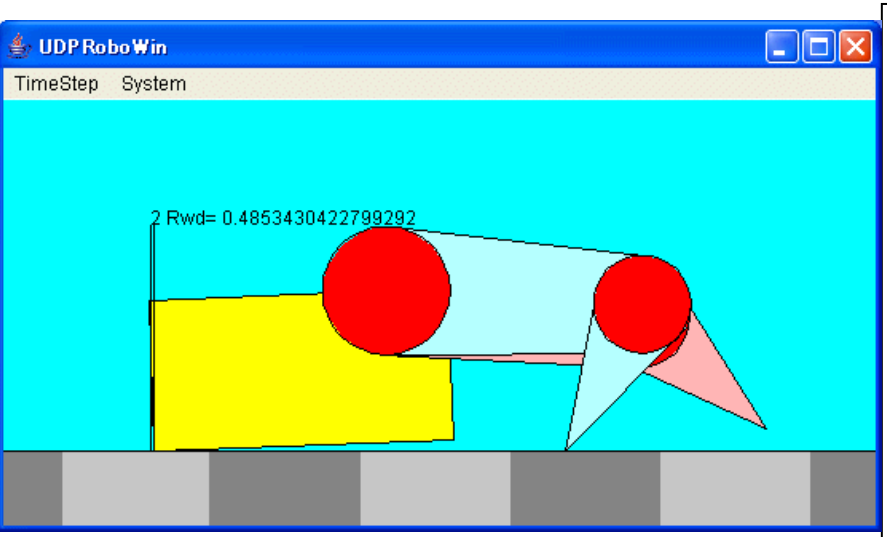
特長

・実装が簡単
・空間爆発を回避
・足りなければ計算反復の
回数を増やすだけ
・分割を細かくしても、計算
はあまり変わらない

ただし、
MaxQ を求める
計算が「近似」に

SARSAアルゴリズムなら問題ない

シミュレーション1： ほふくロボット



2本足： 状態6次元・行動4次元
3本足： 状態9次元・行動6次元

特徴ベクトルの設定

ランダムタイリング

6次元空間10x10x10x10x2x2 の格子状
タイルの境界をこの格子に合わせた
ランダムな大きさのタイルを200個生成
各次元要素をタイルに選択する確率=0.6

9次元空間10x10x10x10x10x10x2x2x2
タイルの境界をこの格子に合わせた
ランダムな大きさのタイルを200個生成
各次元要素をタイルに選択する確率=0.6

Q-learning の設定

割引率 γ=0.9
温　度　T=0.4
学習率 α=0.5

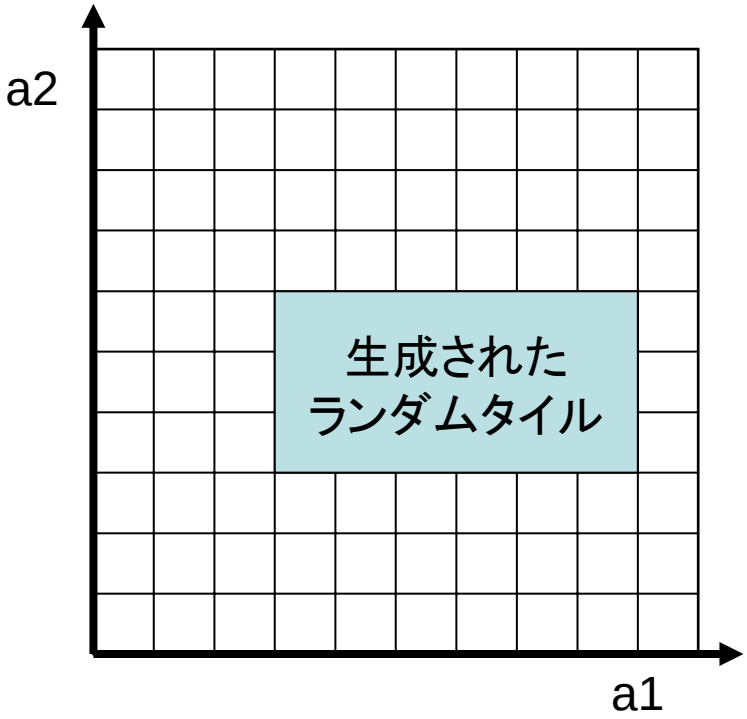
Gibbs-Sampling の設定

反復回数：30回×4 or 6次元

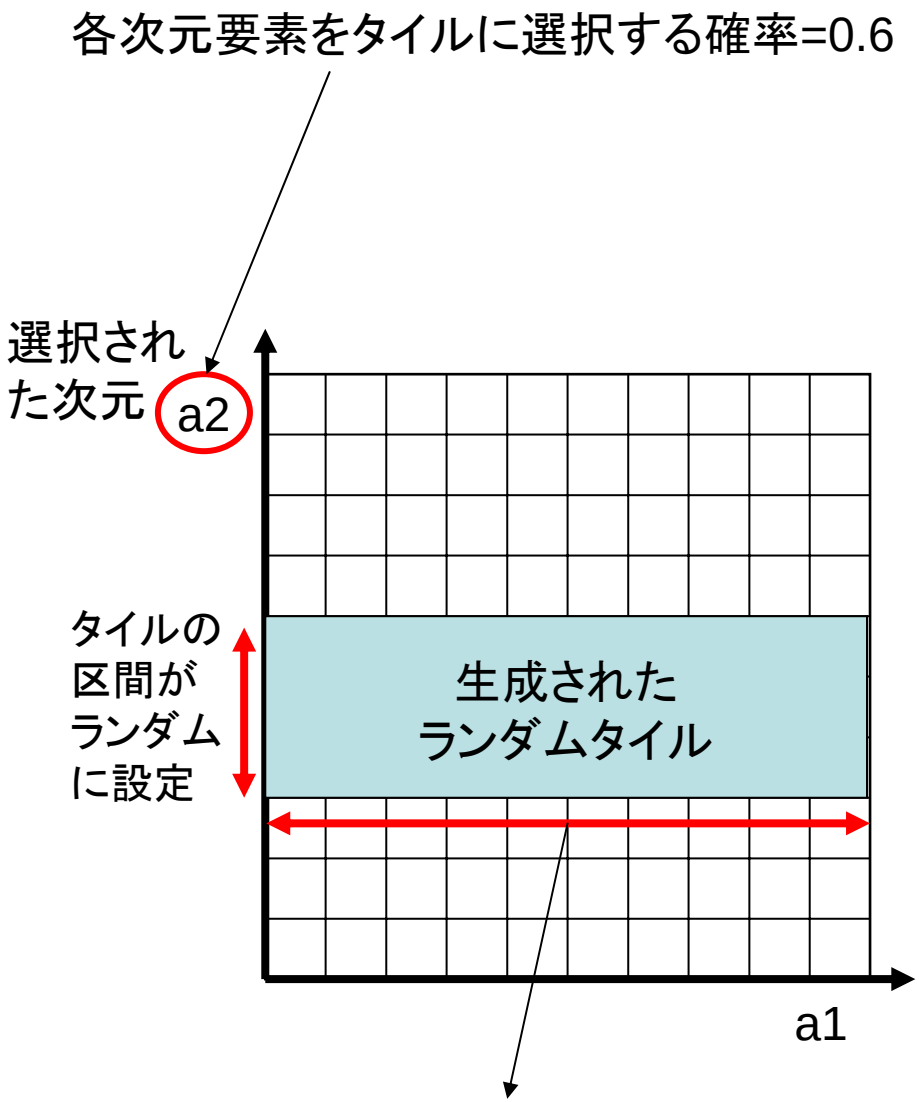
タイル修正のための
データエピソード：

学習初期 2000 step

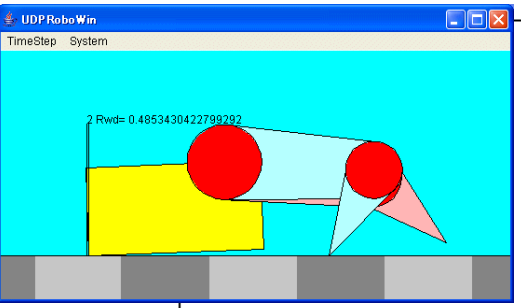
ランダムタイルの生成



3次元空間を10x10x10 の格子状とし、
タイルの境界をこの格子に合わせた
ランダムな大きさのタイルを生成



シミュレーション1： 2本足ロボットの結果



ランダムタイリング
＋修正200タイル

ランダムタイリング
200タイル

RND
GNUE

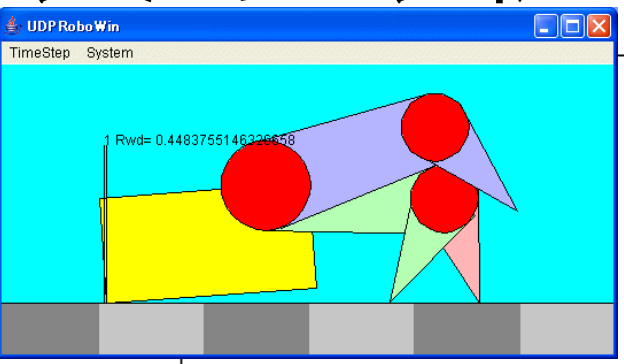
Averaged Reward (1000 steps)

0
5
10
15

0
5000
10000
15000
20000
25000
30000
35000
40000

Learning Steps

シミュレーション1： 3本足ロボットの結果



ランダムタイリング
＋修正200タイル

ランダムタイリング
200タイル

RND
GNUE

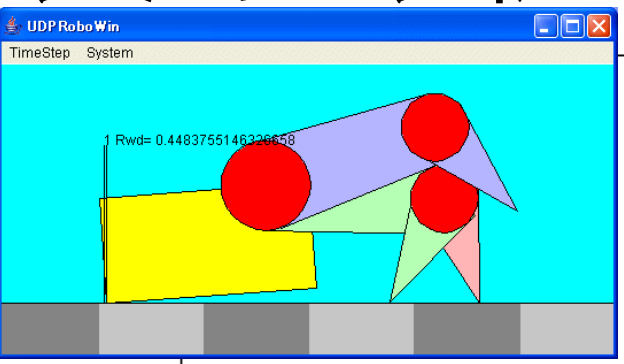
Averaged Reward (1000 steps)

0
5
10
15

0
5000
10000
15000
20000
25000
30000
35000
40000

Learning Steps

シミュレーション1： 3本足ロボットの結果



ランダムタイリング
＋修正200タイル

●ランダムタイルの修正と、
オフライン学習が非常に効果的

ランダムタイリング
200タイル

RND
GNUE

Averaged Reward (1000 steps)

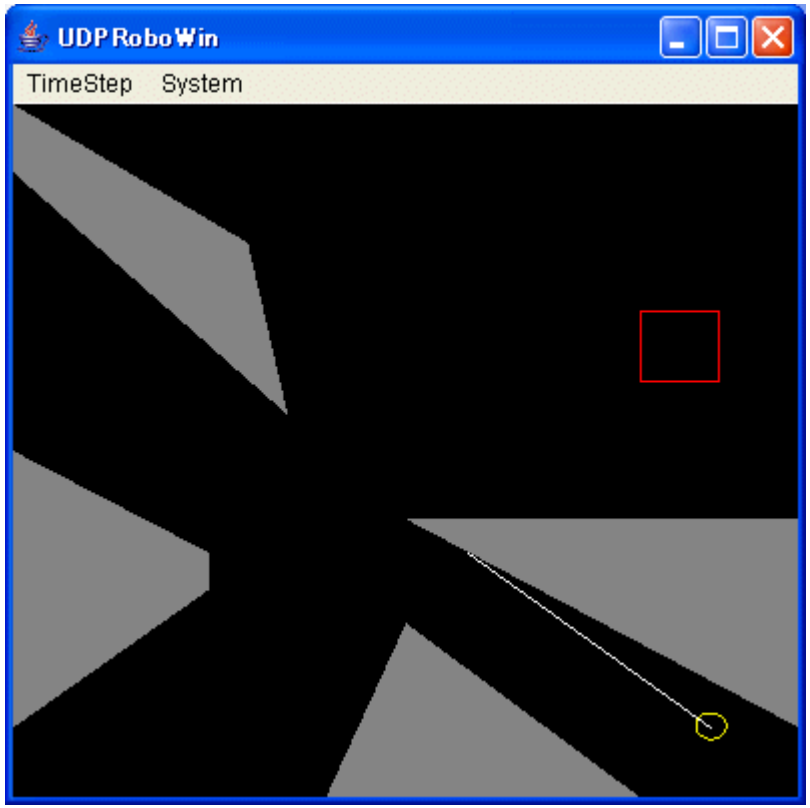
0
5
10
15

0
5000
10000
15000
20000
25000
30000
35000
40000

Learning Steps

シミュレーション2： Rod in Maze

状態3次元
行動3次元



特徴ベクトルの設定

1）ランダムタイリング
3次元空間を10x10x10 の格子状とし、
タイルの境界をこの格子に合わせた
ランダムな大きさのタイルを200個生成
各次元要素をタイルに選択する確率=0.6

2）等間隔グリッドタイリング
3次元空間を 5x5x5 の等間隔グリッドで
分割し、5x5x5=125個のタイル生成

3次元空間を 6x6x6 の等間隔グリッドで
分割し、6x6x6=216個のタイル生成

Q-learning の設定

割引率 γ=0.9
温　度　T=0.4
学習率 α=0.5

Gibbs-Sampling の設定

反復回数：15回×3次元

タイル修正のための
データエピソード：

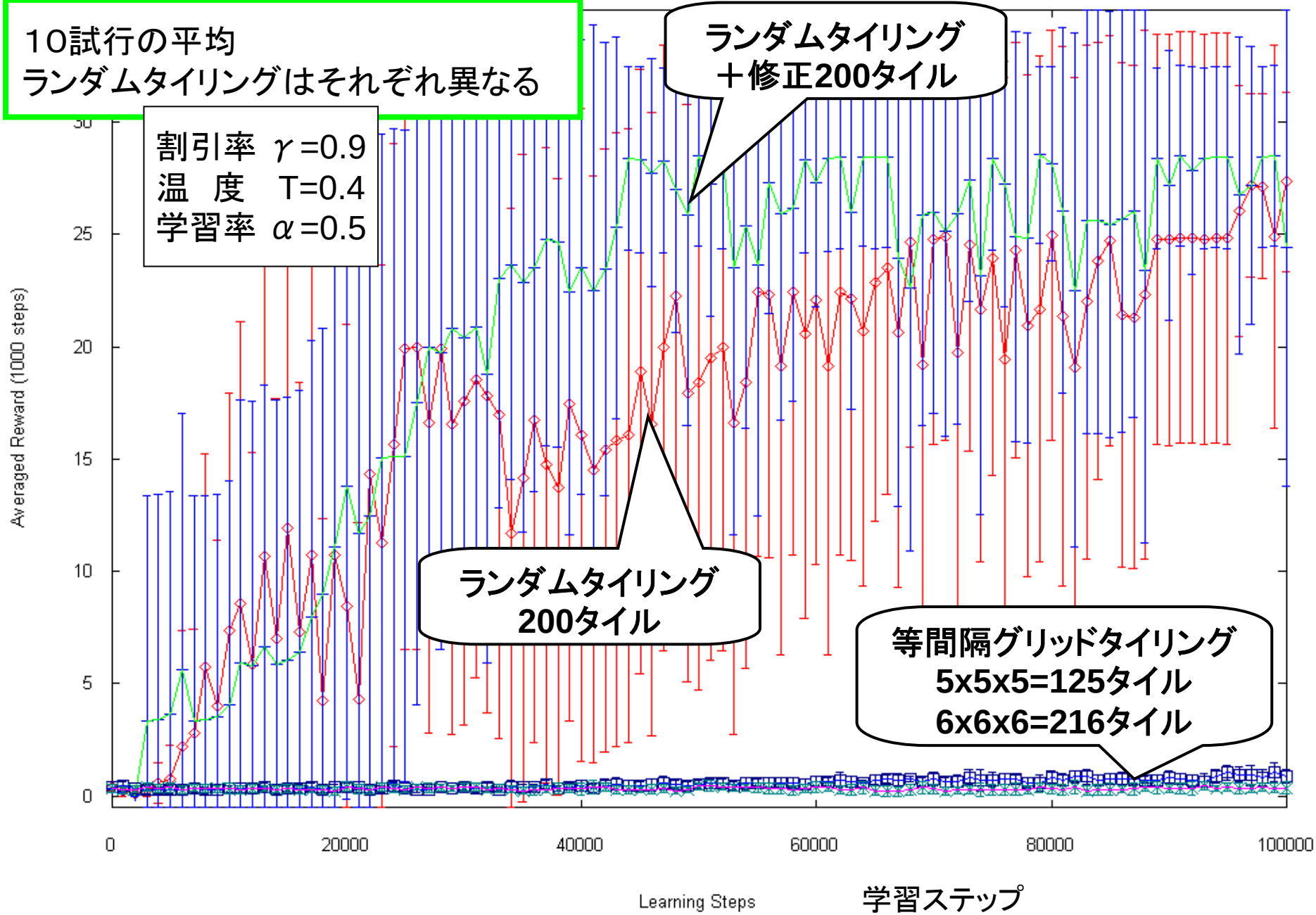
学習初期 2000 step

シミュレーション2結果： Rod in Maze

状態3次元
行動3次元

10試行の平均
ランダムタイリングはそれぞれ異なる

割引率 γ=0.9
温　度　T=0.4
学習率 α=0.5

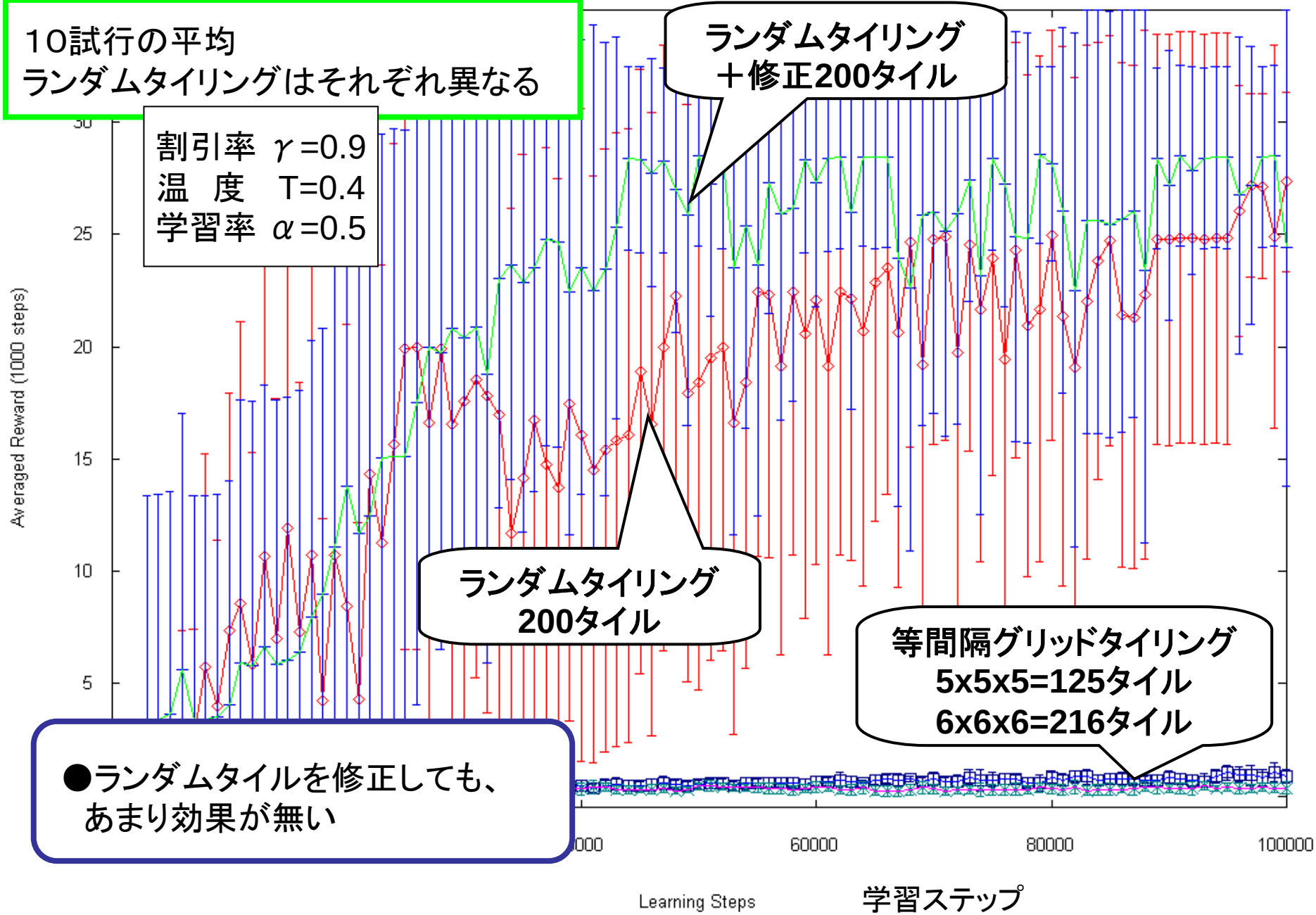


シミュレーション2結果： Rod in Maze

状態3次元
行動3次元

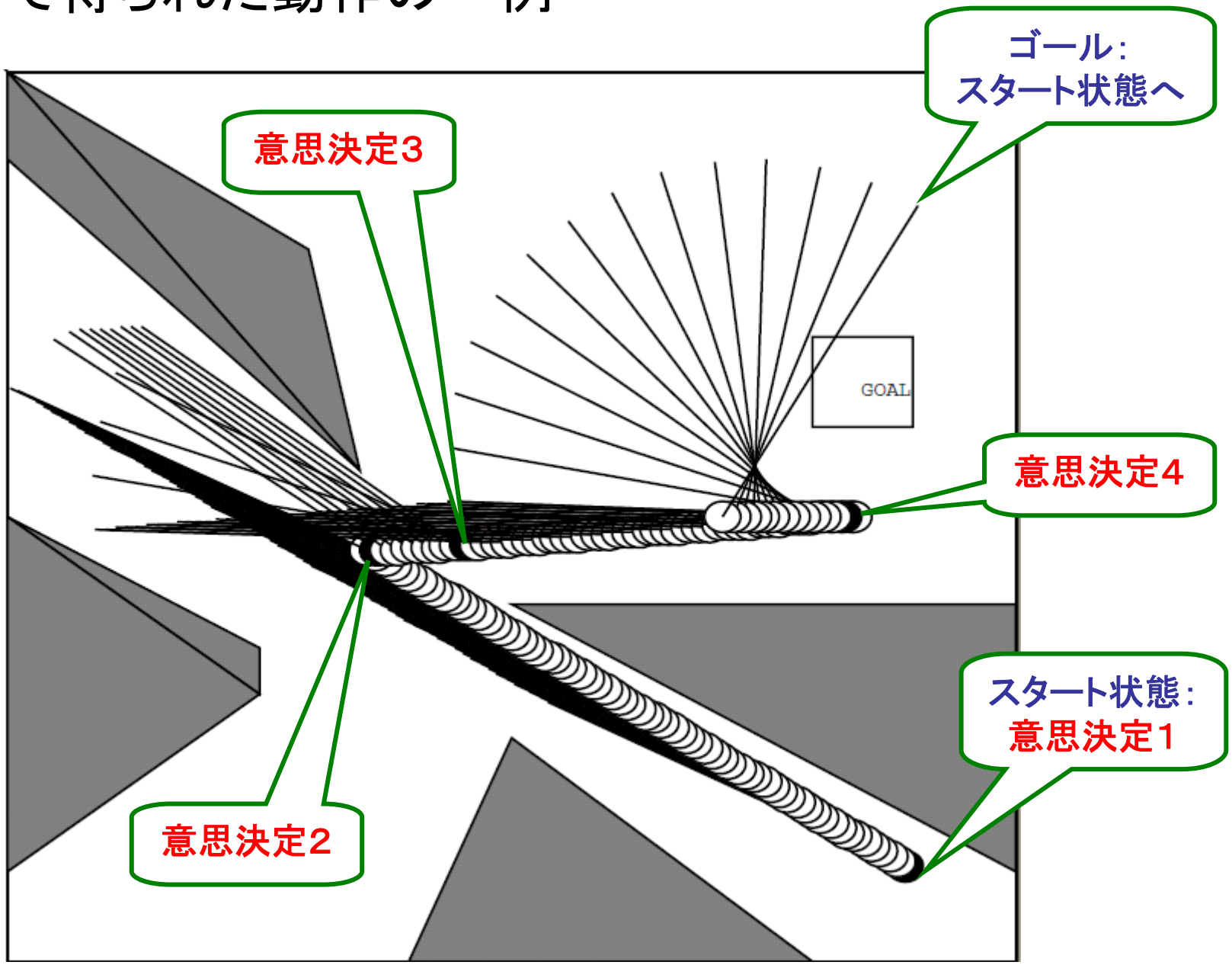
10試行の平均
ランダムタイリングはそれぞれ異なる

割引率 γ=0.9
温　度　T=0.4
学習率 α=0.5



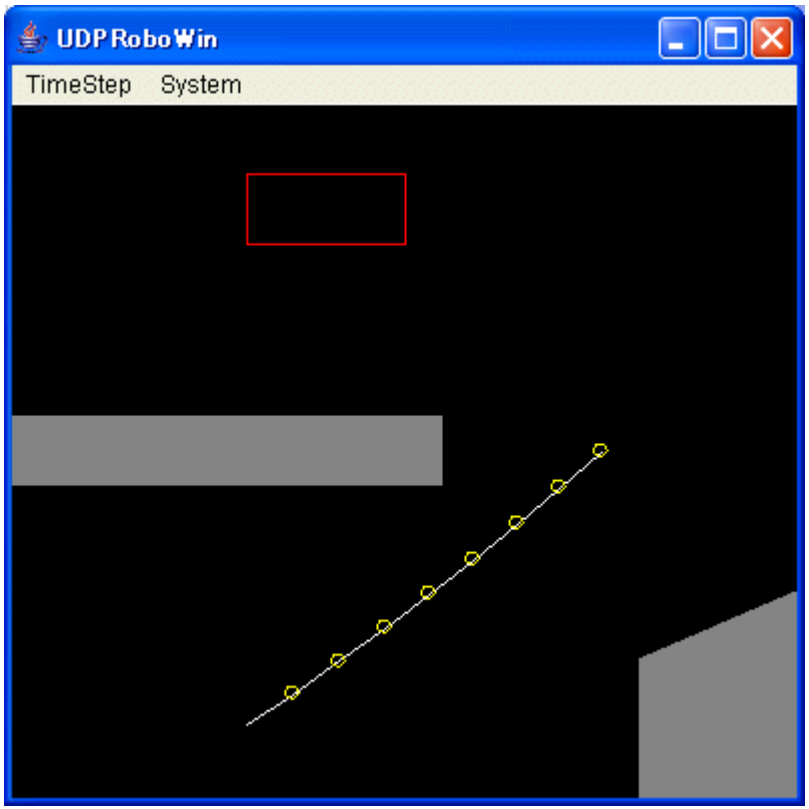
●ランダムタイルを修正しても、
あまり効果が無い

学習で得られた動作の一例



シミュレーション3： Multi-Joint Arm

状態8次元
行動8次元



特徴ベクトルの設定

1）ランダムタイリング
8次元空間を10^8 の格子状とし、
タイルの境界をこの格子に合わせた
ランダムな大きさのタイルを200個生成
各次元要素をタイルに選択する確率=0.3

2）等間隔グリッドタイリング
8次元空間を 2^8 の等間隔グリッドで
分割し、2^8=256個のタイル生成

Q-learning の設定

割引率 γ=0.9
温　度　T=0.4
学習率 α=0.5

Gibbs-Sampling の設定

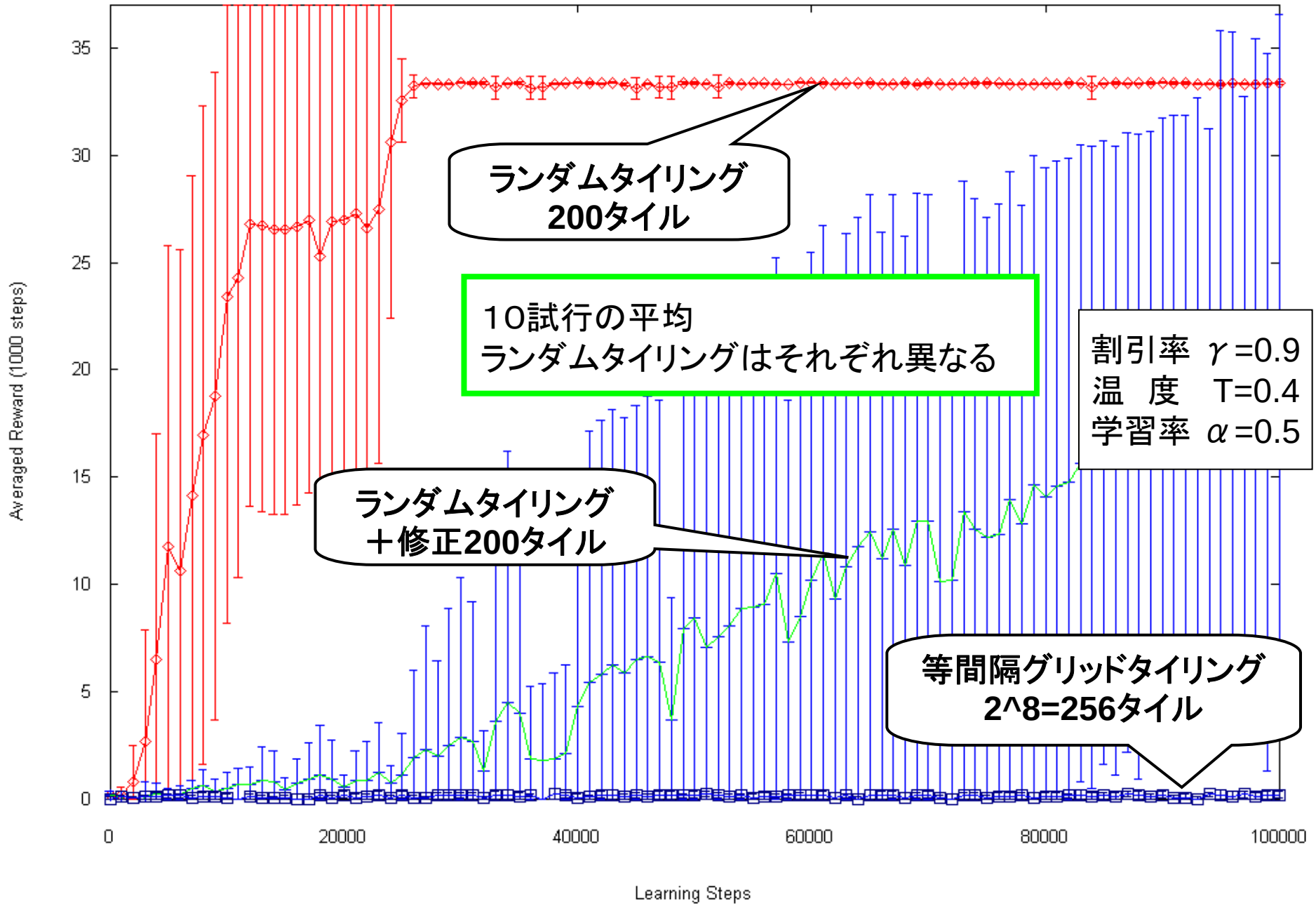
反復回数：40回 × 8次元

タイル修正のための
データエピソード：

学習初期 2000 step

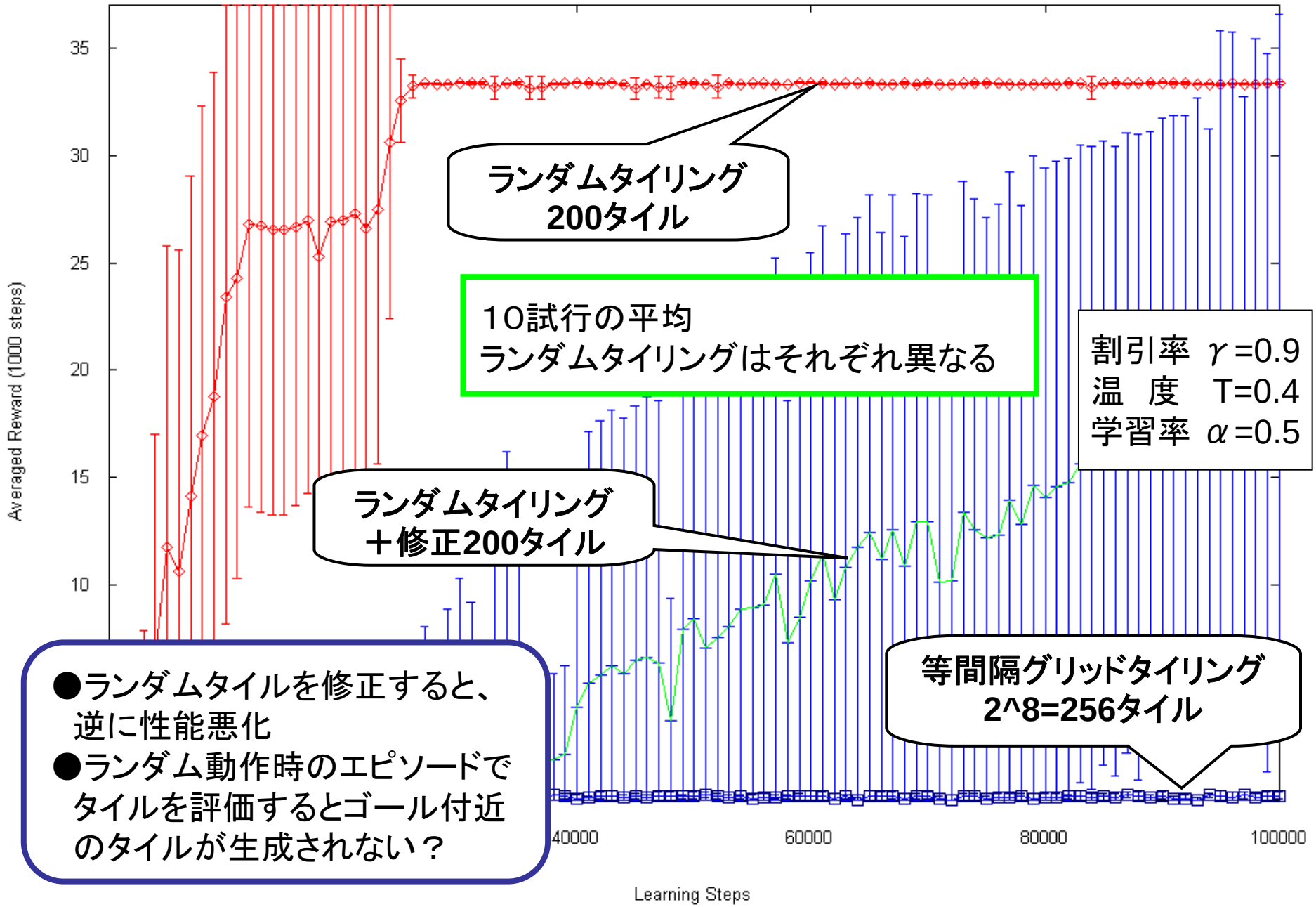
シミュレーション3結果： Multi-Joint Arm

状態8次元
行動8次元



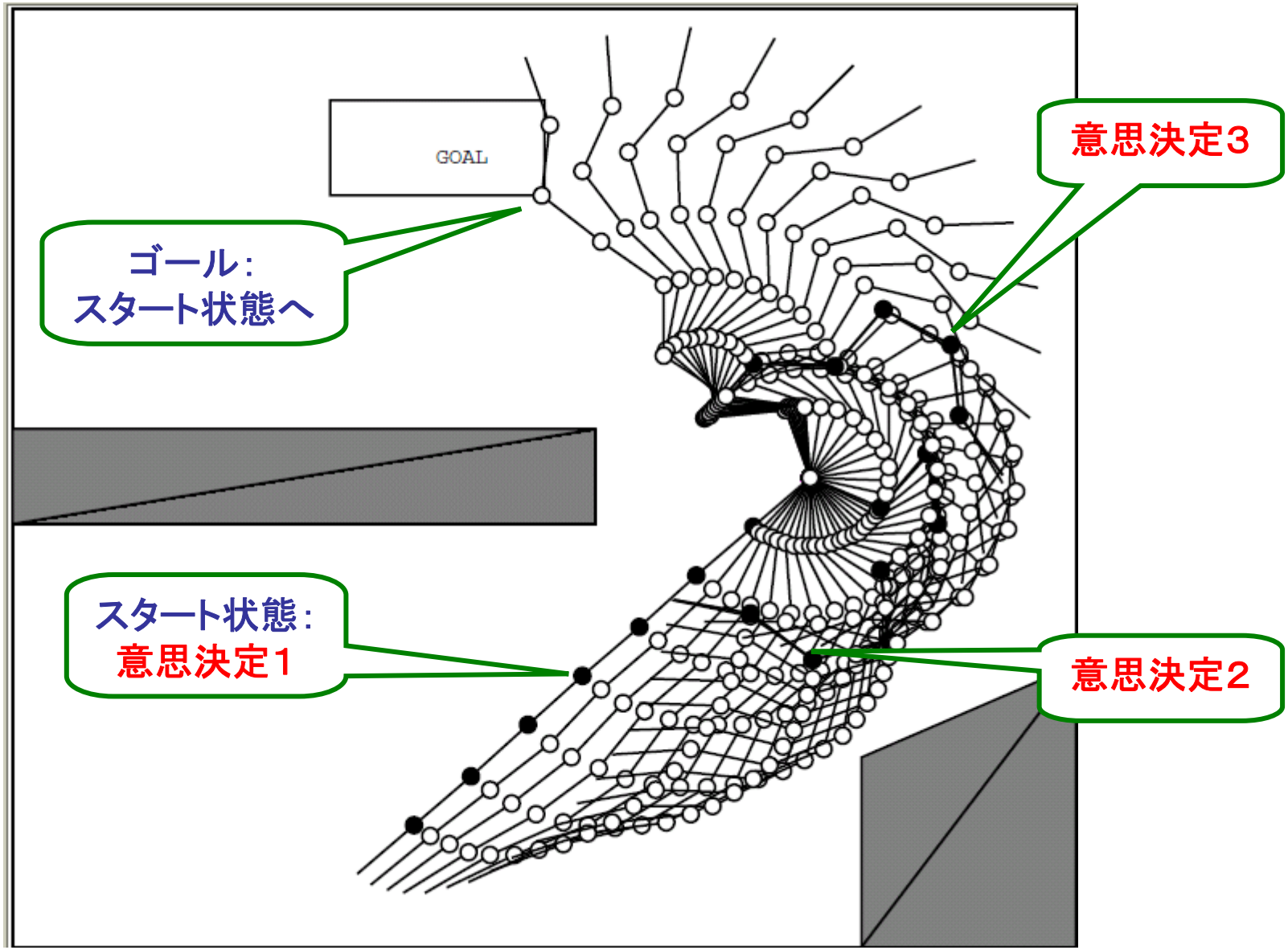
シミュレーション3結果： Multi-Joint Arm

状態8次元
行動8次元



- ランダムタイルを修正すると、逆に性能悪化
- ランダム動作時のエピソードでタイルを評価するとゴール付近のタイルが生成されない？

学習で得られた動作の一例



考察とまとめ

Q-learning

連続空間の関数近似

ボルツマン行動選択

MaxQ を求める計算が近似に

ランダムタイリングによる高次元連続空間の汎化

Gibbsサンプリングによる行動選択処理

- 実装が簡単
- 空間爆発を回避
- 足りなければタイルを増やすだけ
- 実装が問題に依存しない

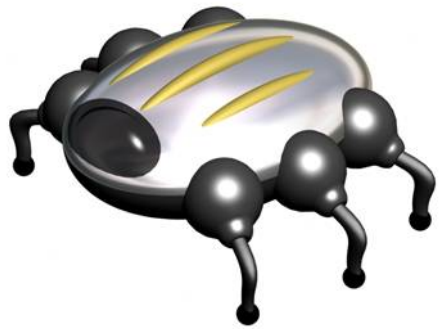
- 実装が簡単
- 空間爆発を回避
- 足りなければ計算反復の回数を増やすだけ
- 分割を細かくしても、計算はあまり変わらない

高次元連続状態-行動空間の強化学習問題へ実装

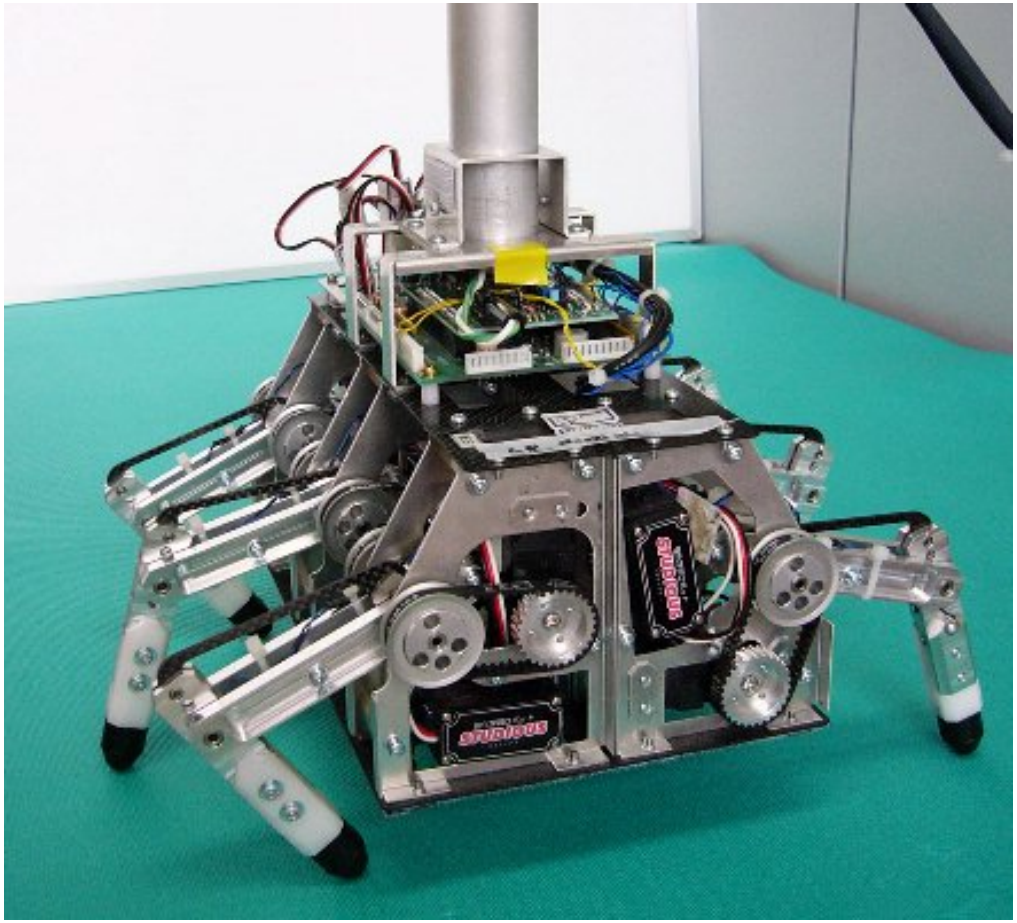
考察とまとめ

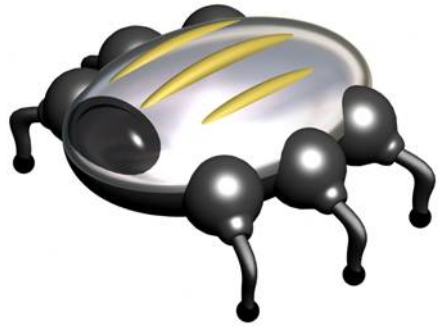
個別ランダムタイルの評価法と修正アルゴリズム

評価に用いるエピソードにおいてランダムな試行を用いる場合、
状態訪問頻度がスタート状態などに偏るときには性能悪化



ランダムタイル群を評価する方法の確立と
タイル群修正アルゴリズムの構築





「強化学習ロボット：スタディアス」
愛・地球博
NEDOプロトタイプロボット展