

# 確率単純文法のサブクラスにおける一般化と 強化学習への応用

柴田 剛志

近山 隆

shibata@logos.t.u-tokyo.ac.jp

chikayama@logos.t.u-tokyo.ac.jp

東京大学工学系研究科電子工学専攻

## Abstract

We introduce an extended Q-Learning formalized by Probabilistic Simple Grammars as one of methods to find optimal actions with subtasks. Using the extended Q-Learning, it is possible for computers to learn optimal actions under some environments where it is impossible or difficult using Q-Learning.

If Simple Grammars (SGs) which represent environments are not given, to identify them is required. A class of Right-unique Simple Grammars (RSGs), which is a subclass of SGs, is known as an effective learnable class from positive data. We propose the way to obtain a SG from given L-equivalent RSGs, which is the most general with respect to probability.

## 1 はじめに

本稿では確率単純文法の一般化という概念について説明し、どうしてそのようなことを考えたのかということの例として、強化学習の環境同定問題への応用を述べる。

まず、強化学習のサブタスク化の手法の一つとして、SGに基づいて定式化したQ学習を紹介する。一般的に言ってサブタスク化の手法はセミマルコフ決定過程 (SMDP) と呼ばれる決定過程をベースとして構築されることが多い [1] [2] [3] が、それとは異なった観点であり、紹介する方法を使うと、いままで学習することが難しかった環境で、最適な行動を学習することができることがある。

環境をあらわす文法を同定するためには、素直に考えると正例からの学習が考えられる。一般的には、正例のみから効率的に文法を学習することは難しいといわれているが [4]、SGのサブクラスのうち、いくつかは効率的に学習することができることが近年示されている。とくにそのうちRSGというクラス [5] の下での環境は有限状態数のMDPを包含する。

したがって、一見RSGのクラスを考えれば文法の同定は解決されるかのように考えられる。しかし、正例から文法を推定することができても、L等価ではあるが確率言語として等価でないものを出力する可能性もある。そこで、本稿では、複数のL等価なRSGが与えられたとき、確率言語として最も一般的なものを出力する方法を提案する。

## 2 確率文法による強化学習の拡張

SGに対して、行動とルールの選択確率と報酬の集合を付け加えることで、その決定確率文法  $G(U, P, C)$  をつぎのように定義する。

- $G = \langle V, \Sigma, R, S \rangle : \text{SG.}$
- $U$  : 行動の有限集合。
- $P$  : ルールの選択確率  $P(R|U, V)$ 。ただし次を満たす。

$$P(A \rightarrow a\alpha|u, B) = 0 \text{ if } A \neq B$$

- $C$  : 報酬  $C : R \times U \rightarrow [\text{実数}]$ 。

$G(U, P, C)$  から最左導出  $S \Rightarrow^* xA_t\alpha(|x| = t)$  によって得られる決定過程は次を満たす。

$$P(r_{t+1}|u_t, r_t, \dots, u_1, r_1, u_0) = P(r_{t+1}|u_t, A_t)$$

ここで、 $r_t$  は  $t$  回目の導出に用いた生成規則、 $u_t$  は  $t$  回目の導出の際にとられた行動である。これは MDP のスタック付きへの拡張といえる。この決定過程に対する拡張した  $Q$  学習は次のように表すことができる。 $p$  をステップサイズパラメータ、 $q$  を割引率として、

$$Q(A_t, u_t) := (1 - p)Q(A, u) + p(C(r_t, u_t) + q \sum_{i=0}^{\#(r_t)} \max_{v \in U} Q(N(r_t, i), v))$$

ここで、

$$\begin{aligned} \#(A \rightarrow aB_0 \dots B_{m-1}) &= m - 1 \\ N(A \rightarrow aB_0 \dots B_{m-1}, i) &= B_i \end{aligned}$$

である。上式はある条件のもとで最適行動価値に収束することがわかっている。

### 3 確率単純文法の一般化

#### 3.1 単純文法のサブクラス

はじめに本稿で用いる記号についての約束ごとを書いておく。

|                                |                  |
|--------------------------------|------------------|
| $a, b, c, \dots$               | 終端記号             |
| $A, B, C, \dots$               | 非終端記号            |
| $\dots, x, y, z$               | 終端記号の列           |
| $\alpha, \beta, \gamma, \dots$ | 非終端記号の列          |
| $A \Rightarrow x\alpha$        | 生成規則を 1 回だけ用いた導出 |
| $A \Rightarrow^* x\alpha$      | 生成規則を 0 回以上用いた導出 |

文脈自由文法  $G = \langle V, \Sigma, R, S \rangle$  が Simple Grammar(SG) であるとは

- $G$  は Greibach 標準形
- $A \rightarrow \alpha\alpha \in R$  and  $A \rightarrow a\beta \in R$  implies  $\alpha = \beta$

また、 $G$  が既約とは、 $\forall A \in V$  で、 $S \Rightarrow^* xA\alpha \Rightarrow^* xy$  となることである。

$G$  が Right-unique Simple Grammar(RSG) であるとは、SG でかつ

$$A \rightarrow \alpha\alpha \in R \text{ and } B \rightarrow b\beta \in R \text{ implies } \alpha = \beta$$

となることである。

また、RSG  $G$  が標準形とは、既約かつ、任意の  $C \in V$  に対して  $A \rightarrow \alpha\alpha C\beta \in R$  となる  $a, \alpha, \beta$  が一意に定まることとする。即ち、任意の  $A \rightarrow \alpha\alpha C\beta, A' \rightarrow \alpha'\alpha'D\beta' \in R$  に対して、 $C = D$  ならば、 $a = a', \alpha = \alpha', \beta = \beta'$  となることである。

RSG のクラスは正例から効率的に学習可能であることが知られている。

#### 3.2 確率単純文法の generality について

$G = \langle V, \Sigma, R, S \rangle$  を SG とする。これに、生成規則の選択確率  $P: R \rightarrow [0, 1]$  を加えたものを Probabilistic Simple Grammar (PSG) といい、 $G_P$  であらわす。ただし、任意の  $A \in V$  で、 $\sum_{r \in R_A} P(r) = 1$  とする。

PSG  $G_P$  に対して、その確率言語  $K(G_P): \Sigma^* \rightarrow [0, 1]$  は、

$$K(G_P)(w) = \begin{cases} \prod_{i=1}^{|w|} P(r(w)_i) & w \in L(G) \\ 0 & w \notin L(G) \end{cases}$$

で定義される。単純文法なので、 $w \in L(G)$  のとき  $w$  を導出するときにその適用した生成規則は一意に決まって、それを  $(r(G, w)_1, \dots, r(G, w)_{|w|})$  であらわしている。

単純文法  $G$  の確率単純文法としての generality を次のように定義する。

$$\mathbb{K}(G) = \{K(G_P) \mid \text{とりうる } P \text{ 全て}\}$$

すなわち、生成確率  $P$  を変化させたときの確率言語全てである。

**定義 1.**  $G = \langle V, \Sigma, R, S \rangle$  および  $G' = \langle V', \Sigma, R', S' \rangle$  を  $L(G') = L(G)$  をみたす単純文法とする。 $G$  が  $G'$  より general であるということを次のように定義する。 $\psi: V \rightarrow V'$  が存在して、任意の  $x \in \Sigma^*$  に対して、 $S \Rightarrow_G^* xA\alpha$  ならば  $S' \Rightarrow_{G'}^* x\psi(A)\alpha'$

これは要するに  $H$  で最左導出していった  $A$  がでてきたときは、かならず  $G$  では  $\psi(A)$  がでてくるということである。つぎが成り立つ。

**事実 1.**  $G, G'$  を単純文法とする。  $G'$  が  $G$  よりも *general* ならば  $K(G) \subset K(G')$  である。

本稿の目的は、RSG を含むような、SG のあるサブクラスにおいて、 $L(G_1) = \dots = L(G_m)$  をみたす任意の  $G_1, \dots, G_m$  に対して、 $K(G_1) \cup \dots \cup K(G_m) \subset K(G_*)$  となる  $G_*$  を求めることである。

このようなことを考えるのは、次のような理由からである。もし前節の Q 学習において、環境をあらわす RSG  $G$  に対して、 $L(G) = L(H)$  だが  $K(G) \neq K(H)$  となる  $H$  を間違えて出力したとしよう。そうすると、そのときもし最適行動をとったときの確率言語  $K(G_P)$  が  $K(G_P) \notin K(H)$  となっていたとしたら、 $H$  の下では最適行動を学習することができない場合がある。しかし、 $L(G) = L(G_*)$  となるような任意の  $G$  について  $K(G) \subset K(G_*)$  となるような  $G_*$  を構成することができたら  $G_*$  の下では最適行動が学習できる。

なお、この問題は、SG に固有の問題といってよく、例えば CFG の場合だとルールの直和をとることとで解決できる自明な問題となる。

### 3.3 generalization in USGs

SG と RSG のクラスの場合とともに、 $K(G_1) \cup K(G_2) \subset K(G_*)$  となる RSG または SG  $G_*$  が存在しないような RSG または SG の組  $G_1, G_2$  ( $L(G_1) = L(G_2)$ ) の例を作ることができる。これは、RSG を正例から学習できても、その結果を PRSG としては、*generality* が一致しない可能性があるため使うことができないことを意味している。

しかし、RSG を真に含むような SG のサブクラスで常に前節の条件を見たすようなクラスを、次のように RSG を拡張することによって作ることができる。

$G = \langle V, \Sigma, R, S \rangle$  を SG とする。  $A \in V$  に対して、

$$s(A) = \{a \in \Sigma \mid A \rightarrow a\alpha \in R\}$$

$$s(A_1 \dots A_m) = s(A_1) \dots s(A_m)$$

とにおいて、 $A \in V, U \subset V$  に対して、

$$\bar{A} = \{A' \in V \mid s(A) = s(A')\}$$

$$\bar{U} = \{A' \in V \mid \exists A \in U, s(A) = s(A')\}$$

$$\overline{A_1 \dots A_m} = \bar{A}_1 \dots \bar{A}_m$$

として、 $G$  が Unifiable Simple Grammar (USG) とは次を満たすこととする。

$$(\bar{A} = \bar{B} \text{ and } A \rightarrow a\alpha \in R)$$

$$\text{implies } (B \rightarrow a\beta \in R \text{ and } \bar{\alpha} = \bar{\beta})$$

明らかに RSG は USG のサブクラスである。

USG である  $G = \langle V, \Sigma, R, S \rangle, H = \langle V', \Sigma', R', S' \rangle$  に対し、 $G$  と  $H$  が同型であるとは、次を満たすこととする。

$$\bullet s(S) = s(S')$$

$$\bullet \forall A \in V, \forall B \in V' \text{ に対し、}$$

$$(s_G(A) = s_H(B) \text{ and } A \rightarrow a\alpha \in R \text{ and } B \rightarrow a\beta \in R') \text{ implies } s_G(\alpha) = s_H(\beta)$$

**事実 2.**  $G$  を RSG とする。  $G$  と同型である既約な RSG の中で最も *general* なものは標準形の RSG である。

USG が前節の性質を満たすことを以下で見てゆく。

$V$  をノードの集合、 $A, B \in V$  に対し、エッジを

$$e(A, B) = \begin{cases} 1 & \text{if } A \rightarrow \dots B \in R, B \neq S \\ 0 & \text{otherwise} \end{cases}$$

とすると、 $\langle V, e \rangle$  は有向グラフとなる。  $A$  の上流を  $up_G(A) = \{B \in V \mid A = B \text{ または } B \text{ から } A \text{ へのパスが存在する}\}$  とおく。

次に、 $U_1 \cap U_2 = \emptyset$  なる  $U_1, U_2 \subset V$  に対して、 $W(U_1, U_2) \subset V^*$  を

$$W(U_1, U_2) = \{\alpha \in V^* \mid \forall A \in U_1$$

$$(\alpha = \alpha' A \beta \text{ implies } \exists B \in U_2 (\beta = B \beta'))\}$$

とおく。

事実 3.  $\alpha\beta \in W(U_1, U_2)$  iff

$$\left\{ \begin{array}{ll} \alpha = \alpha' A \text{ and} \\ \alpha', \beta' \in W(U_1, U_2) & \text{if } \beta = B\beta' \text{ and} \\ & (A, B) \in (U_1, U_2) \\ \alpha, \beta \in W(U_1, U_2) & \text{otherwise} \end{array} \right.$$

次を満たす対  $\langle U_1, U_2 \rangle \in \mathcal{P}(V) \times \mathcal{P}(V)$  を *NbPair* と呼ぶことにする。

1.  $U_1, U_2 \neq \emptyset$ .
2.  $S \notin U_1$ .
3.  $A \in U_1$  ならば  $\bar{A} \subset U_1$ .
4.  $A \in U_1$  ならば、任意の  $B \in \text{up}_G(A)$  について  $B \in U_1$ .
5.  $\exists B \in V (U_2 = \bar{B})$ .
6. 任意の  $A \rightarrow \alpha\alpha \in R$  に対して、
  - $A \in U_1$  ならば  $\forall B \in U_2 (\alpha B \in W(U_1, U_2))$ .
  - $A \notin U_1$  ならば  $\alpha \in W(U_1, U_2)$ .

$U \subseteq V$  が  $\exists U' \subseteq V (\langle U, U' \rangle \text{ は NbPair})$  を満たすとき、 $U$  を *NbPair* の左側と呼ぶ

補題 1.  $G$  が既約ならば  $U_1 \cap U_2 = \emptyset$ .

証明.  $G$  既約および定義の 6 より  $\exists A \in U_1 (\alpha A \beta \in W(U_1, U_2))$ .  $W$  の定義より  $U_2 \not\subset U_1$ . 定義の 3 と 5 より  $U_2 \cap U_1 = \emptyset$ .  $\square$

補題 2.  $G$  が規約とする。  $\langle U_1, U_2 \rangle$  および  $\langle U_1, U_3 \rangle$  が共に *NbPair* ならば、  $U_2 = U_3$  である。

証明.  $A \in U_1$  とする。  $\text{up}_G(A) \subset U_1$  より  $S \notin \text{up}_G(A)$ . よって  $G$  既約および定義の 6 より  $\exists A' \in \text{up}_G(A) (\alpha A' \beta \in W(U_1, U_2))$  である。  $W$  の定義より  $\beta = B\beta'$  and  $B \in U_2$ . 定義の 5 より  $U_2 = \bar{B}$ .  $\square$

この補題により、任意の *NbPair* の左側  $U \subset V$  に対して、  $\langle U, U' \rangle$  が *NbPair* になるような  $U'$  は一意に定まるので、これを  $\text{Nb}_G(U) = U'$  と表記する。

補題 3.  $\langle U_1, U_2 \rangle$  を *NbPair* とする。  $S \xrightarrow{*} x\alpha$  ならば  $\alpha \in W(U_1, U_2)$ .

証明.  $|x|$  に関する帰納法で示す。

(base)  $S \notin U_1$  より  $S \in W(U_1, U_2)$ .

(step)  $S \xrightarrow{*} xA\beta \Rightarrow xA\beta \Rightarrow x\alpha\alpha\beta$  and  $A\alpha \in W(U_1, U_2)$  とする。

Case 1.  $A \in U_1$  のとき。  $W$  の定義より  $\beta = B\beta'$  and  $B \in U_2$ . また *NbPair* の定義の 6 より  $\alpha B \in W(U_1, U_2)$ . したがって  $\alpha\beta \in W(U_1, U_2)$ .

Case 2.  $A \notin U_1$  のとき。  $W$  の定義より  $\beta \in W(U_1, U_2)$ . また *NbPair* の定義の 4 より  $\alpha = \alpha' C$  ならば  $C \notin U_1$ . したがって  $\alpha\beta \in W(U_1, U_2)$ .  $\square$

補題 4.  $G$  を規約とする。 *NbPair* の定義の 6 は次と同値である。

- 任意の  $x$  について  $S \xrightarrow{*} x\alpha$  ならば  $\alpha \in W(U_1, U_2)$

証明. 与条件より *NbPair* の定義の 6 を示す。  $\forall A \rightarrow \alpha\beta \in R$  に対して、  $A\beta \xrightarrow{*} \alpha\alpha\beta$  and  $A\beta, \alpha\beta \in W(U_1, U_2)$  となる  $\beta$  が存在する。

Case 1.  $A \in U_1$  のとき。  $A\beta \in W(U_1, U_2)$  より  $\beta = B\beta'$  and  $B \in U_2$ . したがって  $\alpha B \in W(U_1, U_2)$ .

Case 2.  $A \notin U_1$  のとき。 *NbPair* の定義の 4 より  $\alpha = \alpha' A'$  implies  $A' \notin U_1$ . したがって、  $\alpha\beta \in W(U_1, U_2)$  より  $\alpha \in W(U_1, U_2)$ .  $\square$

$U$  が  $G$  の *NbPair* の左側であるとき、  $\Phi(G, U) = (\langle V', \Sigma, R', S \rangle)$  を既約化したものと定義する。 既約化とは、  $S$  よりたどり着くことのできない  $V$  の元および  $R$  の元を消去することである。 ここで  $V', R'$  は、

- $V' = (V - U) \cup U'$  where  $U' = \{A_B \mid A \in U \text{ and } B \in \text{Nb}_G(U)\}$
- $R' = \{A \rightarrow a\phi(\alpha) \mid A \rightarrow \alpha\alpha \in R \text{ and } A \in V - U\} \cup \{A_B \rightarrow a\phi(\alpha B) \mid A \rightarrow \alpha\alpha \in R \text{ and } A \in U\}$

$R$  and  $A_B \in U'$  } where

$$\phi(\varepsilon) = \varepsilon$$

$$\phi(\alpha A \beta) =$$

$$\begin{cases} \phi(\alpha') A_B \phi(\beta') & \text{if } \beta = B\beta' \text{ and } (A, B) \in U \times \text{Nb}_G(U) \\ \phi(\alpha) A \phi(\beta) & \text{otherwise} \end{cases}$$

である。

**補題 5.**  $G' = \Phi(G, U)$  とする。任意の  $x \in \Sigma^*$  に対して、 $S \xrightarrow{G} x\alpha$  かつ  $\beta = \phi(\alpha)$  iff  $S \xrightarrow{G'} x\beta$  になりたつ。

**証明.**  $|x|$  についての帰納法で示す。

(base)  $S = \phi(S)$

(step)

$$S \xrightarrow{G} xA\beta$$

$$S \xrightarrow{G'} x\phi(A\beta)$$

とし、 $A \rightarrow \alpha\alpha' \in R$  とする。

**Case 1.**  $A \notin U$  のとき。このとき、 $\phi(A\beta) = A\phi(\beta)$  および  $A \rightarrow \alpha\phi(\alpha) \in R'$  になりたつ。したがって、

$$S \xrightarrow{G} xA\beta \Rightarrow \alpha\alpha\beta$$

$$S \xrightarrow{G'} xA\phi(\beta) \Rightarrow x\alpha\phi(\alpha)\phi(\beta) = x\alpha\phi(\alpha\beta)$$

**Case 2.**  $A \in U$  のとき。このとき、補題 3 より、 $\phi(A\beta) = A_B\phi(\beta')$  になりたつ。一方、 $A_B \rightarrow \alpha\phi(\alpha B) \in R'$  だから、

$$S \xrightarrow{G} A\beta \Rightarrow x\alpha\alpha\beta$$

$$S \xrightarrow{G'} xA_B\phi(\beta') \Rightarrow x\alpha\phi(\alpha B)\phi(\beta')$$

$$= x\alpha\phi(\alpha B\beta') = \phi(\alpha\beta)$$

逆に、 $\phi(A\beta) = A'\beta'$  として、 $A' \rightarrow \alpha\alpha' \in R'$  とする。

**Case 1.**  $A \notin U$  のとき。  $A' = A$  and  $\alpha' = \phi(\alpha)$  and  $A \rightarrow \alpha \in R$  および、補題 3 より  $\alpha \in W(U, \text{Nb}_G(U))$  だから、

$$S \xrightarrow{G} xA\beta \Rightarrow \alpha\alpha\beta$$

$$S \xrightarrow{G'} x\phi(A\beta) = xA\phi(\beta) \Rightarrow x\alpha\phi(\alpha)\phi(\beta) = x\alpha\phi(\alpha\beta)$$

**Case 2.**  $A \in U$  のとき。  $A' = A_B$  and  $\alpha' = \phi(\alpha B)$  and  $A \rightarrow \alpha \in R$  および、補題 3 より  $\beta = B\beta'$  and  $B \in \text{Nb}_G(U)$  だから、

$$S \xrightarrow{G} xA\beta \Rightarrow \alpha\alpha\beta$$

$$S \xrightarrow{G'} x\phi(A\beta) = xA_B\phi(\beta')$$

$$\Rightarrow x\alpha\phi(\alpha B)\phi(\beta') = x\alpha\phi(\alpha\beta)$$

□

即ち、 $L(G) = L(\Phi(G, U))$  である。

**補題 6.**  $\Phi(G, U) \in \text{USGs}$

**補題 7.**  $G$  に  $\text{NbPair}$  の左側  $U$  が存在するとする。 $\Phi(G, U)$  は  $G$  よりも general である。

**証明.**  $\psi: V' \rightarrow V$  を、 $\psi(A_B) = A$  if  $A_B \in U'$ 、 $\psi(A) = A$  if  $A \in V' - U'$  とすれば補題 5 より  $\forall x \in \Sigma^*$  に対して、 $S \xrightarrow{\Phi(G, U)} xA\alpha$  ならば  $S \xrightarrow{G} x\psi(A)\alpha'$ 。 □

1: 入力:  $G \in \text{USGs}$

2: while  $G$  の  $\text{NbPair}$  の左側  $U$  が存在する do

3:  $G := \Phi(G, U)$

4: end while

5: 出力:  $G^O := G$

**補題 8.** 上記アルゴリズムは任意の  $G \in \text{USGs}$  に対して停止する。

**証明.**  $H = \Phi(G, U)$  とおく。  $\forall x \in \Sigma^*(S \xrightarrow{G} xA'\alpha$  and  $A' \in \bar{A}$  implies  $\bar{\alpha} = \overline{\gamma\alpha'})$  をみたすような  $\bar{\gamma}$  のうちで最長のものを  $p_G(\bar{A})$  と書くことにする。  $n(\bar{A}) = |\bar{\gamma}(\bar{A})|$  とする。もし任意の  $A \in V$  について  $n(\bar{A}) = 0$  ならば、補題 4 より  $\text{NbPair}$  の左側は存在しない。したがって、

$$n(G) = \sum_{\bar{A} \subset V} n(\bar{A})$$

とおいて、 $n(G)$  が必ず 1 以上減少することを示せばよい。

まず、 $(A, B) \in U \times \text{Nb}(U)$  に対しては、 $p_G(\bar{A}) = \overline{B\beta}$  とかけるから、 $p_H(\overline{AB}) = \overline{\phi(\beta)}$  となる。したがって  $p_H(\overline{AB}) \leq p_G(\bar{A}) - 1$ 。

つぎに、 $A \notin U$  に対しては、 $p_G(\bar{A}) = \bar{\gamma}$  とおくと、 $p_H(\bar{A}) = \overline{\phi(\gamma)}$  となるから、 $p_H(\bar{A}) \leq p_G(\bar{A})$ 。したがって  $n(H) \leq n(G) - 1$ 。 □

補題 9.  $G, G' \in \text{USG}$  とする。  $L(G) = L(H)$  ならば  $G^O$  と  $H^O$  は同型。

証明.  $G = \langle V, \Sigma, R, S \rangle, H = \langle V', \Sigma, R', S' \rangle$  とおく。また、  $L(G, \bar{A}) = \{x \in \Sigma^* \mid \exists A' \in \bar{A} (A' \xrightarrow{G} x)\}$  とおく。これは  $\text{USG}$  の定義から明らかに  $L(G, A) = \{x \in \Sigma^* \mid A \xrightarrow{G} x\}$  と等しい。

まず、  $A \in V, B \in V'$  に対し、  $s_G(A) = s_H(B)$  ならば  $L(G, \bar{A}) = L(G, \bar{B})$  を示す。

$$\bar{A} \xrightarrow{G} y \text{ and } \bar{B} \xrightarrow{H} y \bar{C} \bar{\gamma}$$

となる  $y$  が存在したとすると、  $L(G) = L(H)$  および  $s_G(A) = s_H(B)$  より、任意の  $x$  について

$$S \xrightarrow{G} x \bar{A} \alpha \text{ iff } S \xrightarrow{H} x \bar{B} \beta$$

が成り立つから、

$$S \xrightarrow{G} xy \bar{\alpha} \text{ iff } S \xrightarrow{H} xy \bar{C} \bar{\gamma} \beta$$

が成り立つ。したがって、  $s_G(D) = s_H(C)$  をみたす  $D \in V$  があって、  $\bar{\alpha} = \bar{C} \bar{\alpha}'$  となるが、そうすると任意の  $x$  に対し、

$$S \xrightarrow{G} x \bar{A} D \alpha$$

がみたされることになるので、  $\langle \text{up}_G(A), \bar{D} \rangle$  は  $\text{NbPair}$  となって、  $G^O$  の定義に矛盾する。したがって、  $L(G, \bar{A}) = L(G, \bar{B})$ 。

次に、  $G^O$  と  $H^O$  が同型であることを示す。  
 $s_G(A) = s_H(B)$  and  $A \rightarrow a\alpha \in R$  and  $B \rightarrow a\beta \in R'$  とする。

Case 1.  $\alpha = \varepsilon$  のとき。  $L(G, A) = L(H, B)$  より、  $\beta = \varepsilon$ 。

Case 2.  $\alpha = A_1 \cdots A_m$  のとき。  $\beta = B_1 \cdots B_n$  かつ  $n \geq m$  としてよい。  $s_G(A_i) = s_H(B_i)$  を  $i$  についての帰納法で示す。

(base)  $L(G, A) = L(H, B)$  より、  $s_G(A_1) = s_H(B_1)$ 。

(step)  $s_G(A_1) = s_H(B_1), \dots, s_G(A_i) = s_H(B_i)$  より、  $L(G, A_1 \cdots A_i) = L(H, B_1 \cdots B_i)$ 。したがって、  $L(G, A) = L(H, B)$  より  $s_G(A_{i+1}) = s_H(B_{i+1})$  である。このことより、  $L(G, A_1 \cdots A_m) = L(H, B_1 \cdots B_m)$  が成り立つから、  $L(G, A) = L(H, B)$  より  $n = m$ 。  $\square$

$G_1 = \langle V_1, \Sigma, R_1, S_1 \rangle, \dots, G_n = \langle V_n, \Sigma, R_n, S_n \rangle$  を  $L(G_1) = \dots = L(G_n)$  をみたす  $\text{USG}$  とする。  
 $G_1^O \cdots G_n^O$  に対して、  $G_* = \langle V_*, \Sigma, R_*, S_* \rangle$  を

$$S_* = (S_1, \dots, S_n)$$

$$V_* = \bigcup \{ \bar{A}_1 \times \cdots \times \bar{A}_n \subset V_1^O \times \cdots \times V_n^O \mid s_{G_1^O}(A_1) = \cdots = s_{G_n^O}(A_n) \}$$

$$R_* = \{$$

$$(A_1, \dots, A_n)$$

$$\rightarrow a(B_{1,1}, \dots, B_{n,1}) \cdots (B_{1,m}, \dots, B_{n,m})$$

$$\mid (A_1, \dots, A_n) \in V_* \text{ and}$$

$$A_1 \rightarrow aB_{1,1} \cdots B_{1,m} \in R_1^O \text{ and } \cdots \text{ and}$$

$$A_n \rightarrow B_{n,1} \cdots B_{n,m} \in R_n^O$$

$\}$

とし、  $\langle V_*, \Sigma, R_*, S_* \rangle$  を既約化したものとする。

補題 10.  $L(G_1) = L(G_*)$  である。

定理 1.  $G_1 \cdots G_n$  を  $L(G_1) = \dots = L(G_n)$  をみたす  $\text{USG}$  とする。  $G_1 \cdots G_n$  より上記のように構成した  $G_*$  は、任意の  $G_i$  よりも *general* である。

証明.  $G_1^O$  よりも  $G_*$  が *general* であることを示せば十分である。  $\pi: V_* \rightarrow V_1^O$  を  $\pi(A_1, \dots, A_n) = A_1$  とする。

補題 9 および  $R_*$  の定義より、  $A \rightarrow a\alpha \in R_1^O$  iff  $(A_* \rightarrow a\alpha_* \in R_* \text{ and } \pi(A_*) = A \text{ and } \pi(\alpha_*) = \alpha)$

である。よって任意の  $S_* \rightarrow x\alpha_*$  に対して  $S \rightarrow x\pi(\alpha_*)$  となる。  $\square$

## 4 結論

全ての  $\text{USG}$  は、 $L$  等価という条件の下で、同型なものを除いてただひとつの  $\text{USG}$  に変換することができる。さらにこの変換の過程は *generality* を損なわないようにとることができる。また同型なものどうしからは、単に並置するだけでより *general* なものが作れる。

一方で、 $\text{RSG}$  は正例からの学習の結果、 $L$  等価な全ての標準形の  $\text{RSG}$  を出力することができるこ

とが知られている。RSG は USG のサブクラスであったから、L 等価な全ての RSG よりも general なものを作ることができる。したがって、RSG は、正例から generality を損なわない USG を学習することができる。

これを用いれば、環境をあらわす RSG が未知であっても、それを上記の方法で推定した後、Q 学習を行うことで最適行動を学習することができる。

問題点としては、L 等価な標準形の RSG の数および NbPair 全ての数に対して、指数的に  $V$  のサイズが増える可能性があることである。実際に、そのような RSG の例を作ることが可能である。しかし、一方で環境がほとんど普通の MDP に近いような場合であれば、NbPair は十分少ないことが予想できる。したがって、今後の課題として、NbPair と L 等価な標準形の RSG の数に対して  $V$  のサイズのオーダーを細かく計算することや、NbPair と L 等価な標準形の RSG の数が少なく抑えられるのはどのようなときかを明確にすることを考えている。

from positive data. 京都大学数理解析研究所講究録, Vol. 1375, pp. 106–112, 2004.

## 参考文献

- [1] Andrew G. Barto and Sridhar Mahadevan. Recent advances in hierarchical reinforcement learning. *Discrete Event Dynamic Systems: Theory and Applications*, Vol. 13, pp. 41–77, 2003.
- [2] Thomas G. Dietterich. Hierarchical reinforcement learning with the maxq value function decomposition. *Journal of Artificial Intelligence Research*, Vol. 13, pp. 227–303, 2000.
- [3] Ronald E. Parr. *Hierarchical control and learning for Markov decision processes*. PhD thesis, University of California at Berkeley, 1998.
- [4] Yasubumi Sakakibara. Recent advances of grammatical inference. *Theoretical Computer Science*, Vol. 185, pp. 15–45, 1997.
- [5] Ryo Yoshinaka. Polynomial-time identification of an extension of very simple grammars