1 文脈自由文法，括弧文法，全体の概要

まず文脈自由文法（CFG）のことを思い出す．CFG $\mathcal{G}$ とは，4 つ組 $(\mathcal{V}, \Sigma, \mathcal{P}, \mathcal{S})$ で，

- 変数（非終端記号，nonterminal）の有限集合 $\mathcal{V}$ と，
- 文字（終端記号，terminal）の有限集合 Σ と，
- 生成規則 $A \to w \in \mathcal{P}$ の有限集合（ただし，$A \in \mathcal{V}$ かつ $w \in (\mathcal{V} \uplus \Sigma)^*$）と，
- 開始記号の集合 $S \in \mathcal{S} \subseteq \mathcal{V}$

からなる．

終端記号だけからなる列を，終端記号列と呼ぶ．変数と終端記号からなる列を，文形式と呼ぶことにする．

生成規則は，記号列の書換の仕方を定める．例えば，文形式 $w_1 A w_2$ を規則 $A \to w$ で書き換えると $w_1 A w_2 \Rightarrow w_1 w w_2$ となる．このような 1-step の書き換え $\Rightarrow$ を定義した上で，文法 $\mathcal{G}$ の定める言語 $L(\mathcal{G})$ とは，

$$L(\mathcal{G}) \triangleq \{w \in \Sigma^* : S \in \mathcal{S}, S \Rightarrow^* w\}$$

として定義される．何かしらの開始記号から，文形式ないし終端記号列 w が導出されることを $\mathcal{S} \Rightarrow^* w$ と書くことにする．

標準的な CFG の定義と異なる点として，複数の開始記号を許可するということがある．このようにする理由は，2 章の定理 1 において明らかになる．

括弧文法とは，括弧を表すために用いる終端記号 $\{(,)\} \subseteq \Sigma$ を備えた文法で，加えて全ての生成規則が次の形をしたものである：

$$A \to (\, w\,).$$

ただし,「w の中には括弧が現れず，また空文字列ではない $w \neq \epsilon$」の 2 点を要求する．

これ以下では，括弧文法だけを考えていくことになる．また，概要は以下の通りである：

1. 最終目的は，与えられた 2 つの括弧文法 $\mathcal{G}_1, \mathcal{G}_2$ の包含判定問題 $L(\mathcal{G}_1) \subseteq_? L(\mathcal{G}_2)$ が決定可能であることを示すことにある．
2. これを示すために，2 つの括弧文法の同等性判定 $L(\mathcal{G}_1) =_? L(\mathcal{G}_2)$ が決定可能であることを示す．
3. 同等性判定が解けたならば，包含判定アルゴリズムが得られる．なぜならば，
$$L(\mathcal{G}_1) \subseteq L(\mathcal{G}_2) \iff L(\mathcal{G}_1) \cup L(\mathcal{G}_2) = L(\mathcal{G}_2)$$
が成立し，定義から明らかだが，括弧文法は union を取る操作について閉じているためである．
4. 同等性判定を示すために，以下の流れを踏む：
   (a) 与えられた括弧言語に対する,「ある意味での」決定化に相当する手続きを施し，*backward-deterministic*（**後ろ向き決定的**）文法に変換できることをみる．
   (b) 変数上に，区別可能・不能の概念を導入し，そこから同値類を導入する．さらにこれをもとに，既約（reduced）かつ後ろ向き決定的文法を得る．
   (c) 既約かつ後ろ向き決定的な 2 つの文法が，同じ言語を生成することと同型（変数の一対一書き換えで移りあえる）であることは等しいと示される．特に後者が成立しているかどうかは計算可能なので，これで等価性判定を行う．

非常に大雑把にいって，非決定性有限オートマトンの等価性判定と似た雰囲気を持っている，と思えるところが少し面白い．

2　後ろ向き決定的括弧文法

括弧文法が後ろ向きに決定的（以下では単に「決定的」と書く）であるとは，2 つの生成規則 A, B で

$$A \to (\, w\,), \quad B \to (\, w\,)$$

とするものがないことを言う．ここで一度，定義内容の習熟のために次の命題を考える．

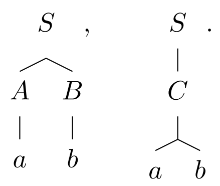
命題 1. 括弧文法 $\mathcal{G}$ が決定的であるならば，$\mathcal{G}$ は無曖昧である．

Proof. ある CFG が「無曖昧」であるとか，それに関わる「導出木」の定義は，標準的な教科書を参照．

決定性の定義そのものは，括弧文法であることに限定されないので，命題をより一般化し，文脈自由文法について示そうと意気込んではならない．

$$S \to AB,\ S \to C, \quad A \to a, \quad B \to b, \quad C \to ab.$$

この CFG は定義上は決定的であるようにみえるが，$S \Rightarrow^* ab$ について曖昧である：



従って，本命題は括弧文法特有の現象に従って証明されなければならない．

ある開始記号 S から導出される終端記号列 w について，２つの異なる導出木 T_1, T_2 が存在したとする．w を，$w = w_1(w_2)w_3$（ただし，w_2 には括弧が現れない）として書くことにすると，$S \Rightarrow^* w_1(\,w_2\,)w_3$ であることから，

$$S \Rightarrow^* w_1 A w_3 \Rightarrow w_3 \Rightarrow^* w_1(\,w_2\,)w_3 \qquad \text{（ただし，}A\text{ は変数）}$$

と分解することができる．内部に括弧が現れない文字列に注目できれば，分解の位置はどこでも良い．

ここで，文法が決定的なので，分解の位置を決めたときに，A は $(\,w_2\,)$ から唯一つ定まることに注意．すなわち，生成規則に $A \to (\,w_2\,)$ が存在する．T_1 と T_2 も $A \to (\,w_2\,)$ の部分を書き戻すことで，２つの導出木 T_1' と T_2' が得られる．このとき，$T_1 \neq T_2$ であるから，$T_1' \neq T_2'$ でなければならない．再びこの議論を T_1' と T_2' に適用してどんどん木を書き戻していくと，両方から S に到達してしまい，矛盾する． □

終端記号列 $w \in \Sigma^*$ が受理されるかどうかを考える際には，上の命題の証明でみたように，$w = w_1(\,w_2\,)w_3$ で w_2 に括弧がでないような分解をして，生成規則を逆向きに適用することで，後ろ向きに（ボトムアップに，葉から根/開始記号に向かって）計算を行っていけば良い：

$$v\,(\,w_1\,\{A_1, A_2, \ldots, A_n\}\,w_3\,)v' \Rightarrow v\,(\,w_1\,(\,w_2\,)\,w_3\,)\,v'.$$

ここで，$\{A_i\}$ は，生成規則で $A_i \to (\,w_2\,)$ の形をするもの全てである．一般の括弧文法は決定的ではないので，上の図のように，一歩前の候補となりうる変数全てを保持しておく必要がある．

ここでもし，生成規則に $B_1 \to w_1 A_1 w_3$ と $B_2 \to w_1 A_2 w_3$ があり，他の A_n についてはそのようなものがない場合には，更に推し進めて

$$v\,\{B_1, B_2\}\,v' \Rightarrow v\,(\,w_1\,\{A_1, A_2, \ldots, A_n\}\,w_3\,)\,v' \Rightarrow v\,(\,w_1\,(\,w_2\,)\,w_3\,)\,v'$$

としていく．この考え方が，次の定理を導く．

定理 1. 括弧文法 $\mathcal{G}$ について，決定性括弧文法 $\mathcal{G}'$ で $L(\mathcal{G}) = L(\mathcal{G}')$ とするものが構成できる．

Proof. 証明は，非終端記号の各部分集合 $\subseteq \mathcal{V}$ を，新たに一つの非終端記号としてみなすことで行われる．証明の前に，定義のところで述べた,「複数の開始記号」の使い道がこの定理にあることをみる．

$$\mathcal{G} : S \to (a), \quad S \to (A), \quad A \to (a).$$

これを決定化するためには，開始記号を複数使わなければならない．確認してみてください．

ここでは，変数記号の集合に $\mathcal{A} \subseteq \mathcal{V}$ を使うことにする．次の形の生成規則を，文法に加えるべきかどうかを考える：

$$\mathcal{A} \to (w_1 \mathcal{A}_1 w_2 \mathcal{A}_2 w_3).$$

ここで定理直前の議論を思い出すと，右側は実際には $X = \{(w_1 A_1 w_2 A_2 w_3) : A_1 \in \mathcal{A}_1, A_2 \in \mathcal{A}_2\}$ という形式に，ボトムアップでの構文解析で畳み込めたということを意図している筈である．従って，$\mathcal{A}$ は，文法 $\mathcal{G}$ の生成規則を使って，X を畳み込んだ結果全体を表すべきである．すなわち，各組合せ $(\, w_1\, A_1\, w_2\, A_2\, w_3\,)$ （$A_1 \in \mathcal{A}_1, A_2 \in \mathcal{A}_2$）について，$A \to (\, w_1\, A_1\, w_2\, A_2\, w_2\,) \in \mathcal{G}$ ならば，A を $\mathcal{A}$ に追加する．

右側が左側を一意に定めるので，決定的な文法 $\mathcal{G}'$ が生成されることは間違いない．

$L(\mathcal{G}') \subseteq L(\mathcal{G})$ を示す．これには，次の主張を示せば十分である．

主張 1. $\mathcal{G}' : \{A_1, A_2, \ldots, A_n\} \Rightarrow^+ (w)$ ならば，任意の $i \in [1..n]$ について $\mathcal{G} : A_i \Rightarrow^* (w)$ が成立する．

まず長さが 1 の時を示す．すなわち

$$\{A_1, A_2, \ldots, A_n\} \Rightarrow (w).$$

$\mathcal{G}'$ における生成規則の定め方から，$A_i \to (w)$ が $\mathcal{G}$ にあるので，明らか．

導出の長さが $n+1$ の時を考える：

$$\{A_1, A_2, \ldots, A_n\} \Rightarrow (w_1\{A'_1, \ldots, A'_m\}w_2\{A''_1, \ldots, A''_l\}w_3) \Rightarrow^+ (w_1(v_1)w_2(v_2)w_3).$$

$\{A'_1, \ldots, A'_m\} \Rightarrow^+ (v_1)$ であるから $A'_j \Rightarrow^* (v_1)$ で $\{A''_1, \ldots, A''_l\} \Rightarrow^+ (v_2)$ であるから $A''_k \Rightarrow^* (v_2)$ が成立する．

さて，$\{A_1, A_2, \ldots, A_n\} \to (w_1\{A'_1, \ldots, A'_m\}w_2\{A''_1, \ldots, A''_l\}w_3)$ ということは，任意の $i \in [1..n]$ について，ある $A' \in \{A'_1, \ldots, A'_m\}$ と $A'' \in \{A''_1, \ldots, A''_l\}$ で，$A_i \to (\, w_1\, A'\, w_2\, A''\, w_3\,)$ とするものがある．従って，

$$A_i \Rightarrow (\, w_1\, A'\, w_2\, A''\, w_3\,) \Rightarrow^+ (\, w_1\, (v_1)\, w_2\, (v_2)\, w_3\,)$$

が成立する．

次に $L(\mathcal{G}) \subseteq L(\mathcal{G}')$ を示す．これには，次の主張を示せば十分である．

主張 2. $\mathcal{G} : A \Rightarrow^+ (w)$ ならば，ある $A \in \mathcal{A}$ で，$\mathcal{G}' : \mathcal{A} \Rightarrow^+ (w)$ が成立する．

まず導出の長さが 1 のときを考える：

$$A \Rightarrow (w)$$

であるときには，$\mathcal{G}$ の生成規則に $A \to (w)$ が存在し，従って $\mathcal{G}'$ では $\mathcal{A} \to (w)$ かつ $A \in \mathcal{A}$ とするものがある．

導出の長さが $n+1$ のときを考える：

$$A \Rightarrow (w_1 A' w_2 A'' w_3) \Rightarrow^+ (w_1(v_1)w_2(v_2)w_3)$$

として書ける．帰納法の仮定により，$A' \in \mathcal{A}'$ と $A'' \in \mathcal{A}''$ で，

$$(w_1\mathcal{A}'w_2\mathcal{A}''w_3) \Rightarrow^+ (w_1(v_1)w_2(v_2)w_3)$$

が成立する．

さて，$A \Rightarrow (w_1\, A'\, w_2\, A''\, w_3)$ であるから，$\mathcal{G}$ に生成規則 $A \to (w_1A'w_2A''w_3)$ が存在する．従って，$\mathcal{G}'$ には，$A \in \mathcal{A}$ で以下を満たすものが存在する：

$$\mathcal{A} \Rightarrow (w_1\mathcal{A}'w_2\mathcal{A}''w_3) \Rightarrow^+ (w_1(v_1)w_2(v_2)w_3).$$

これによって主張が示された． □

3 区別できない変数を同一視するための準備

まず context と呼ぶ概念を定義する．context とは，1 つの hole を持つ文形式 $\beta \equiv w_1[\_]w_2$ のようなものである．Hole を埋める—$\beta[w_3]$—ことで，完全な文形式—$w_1w_3w_2$—を得ることができる．

決定的括弧文法において，2 つの変数 A と B が「n-step で区別可能」とは，ある context β で，以下を満たすものが存在するとき：

- $\beta[A]$ がある開始記号のもとで，最短 n-step で導出可能である一方，$\beta[B]$ が導出可能ではない．

  または，

- $\beta[A]$ が導出可能ではないが，$\beta[B]$ が最短 n-step で導出可能である．

変数 A と B が区別可能であるとは，何らかの n のもとで，n-step 区別可能であるとき．逆に A と B が区別可能ではないことを，区別不能という．直感的には，2 つが区別不能であるというのは，導出において $\beta[A]$ を得たならば，A を B で置き換えても良い（$\beta[A_2]$）という，置換可能であることを述べているはずである．このようにして，お互いに移りあえるものを一纏めにするということは，次章で行う．

（**Remark**：区別可能とか区別不能という語は，McNaughton 自身が論文において *distinguishable* という言葉遣いをしているところから，訳出しただけのつもりです．オートマトンの最小化でも，Nerode partition を通じて最小化を行いますが，あの時の考え方と何かしら類似点があるのかどうかが良くわかっていません．何かあるなら，それを分析する立場から形式化したいです．ということで，この distinguishable という言葉遣いが本当に適切かどうか今ひとつ分からないので，何かコメントがあれば教えてください．）

定理 2. 決定的な文法について，2 つの変数 A_1 と A_2 が区別可能かどうかは，決定可能である．

この定理を証明するために，McNaughton の論文にあるものと同じ（ような）補題を示す．

補題 1. 変数 A_1 と A_2 が $(n+1)$-step 区別可能であるときに，別の区別可能な変数の組 B_1 と B_2 で n-step 区別可能なものがある．

Proof. $\mathcal{S} \Rightarrow^* \beta[A_1]$ で $\mathcal{S} \not\Rightarrow^* \beta[A_2]$ だとする．ただし，$\mathcal{S} \Rightarrow^* \beta[A_1]$ には $(n+1)$-step かかる．

このとき，例えば，$\beta = (w_1(v_1[\_]v_2)w_2)$ であったとすると，$\mathcal{S} \Rightarrow^* \beta[A_1]$ であるから，

$$\mathcal{S} \Rightarrow^* (w_1Xw_2) \Rightarrow (w_1(v_1A_1v_2)w_2)$$

として直前の形が決定性から唯一つ定まる．

ここで，$\mathcal{S} \not\Rightarrow^* \beta[A_2]$ となる原因は，2 通りある：

1. $Y \to (v_1 A_2 v_2)$ とするような生成規則が存在しないか，

2. または，そのようなものが（唯一）存在したとしても $\mathcal{S} \not\Rightarrow^* (w_1 Y w_2)$ となる．

前者の場合は，より小さい context $\beta' = (v_1[\_]v_2)$ で「A_1 と A_2」を区別することができる．しかし，このとき $\mathcal{S} \Rightarrow^* \beta'[A_1]$ は，$\mathcal{S} \Rightarrow^* \beta[A_1]$ より明らかに短い遷移で導出可能であるから，仮定に反する．

後者の場合は，より小さい context $\beta'' = (w_1[\_]w_2)$ で「X と Y」を区別できる．このとき，$\mathcal{S} \Rightarrow^* \beta''[X]$ は，丁度 n-step あると導出可能である．一方で，$\mathcal{S} \Rightarrow^* \beta''[X]$ が $(n-1)$-step 以下で導出可能であったとすると，これは $\mathcal{S} \Rightarrow^* \beta''[X] \Rightarrow \beta[A_1]$ の最短導出が $(n+1)$-step であることに反する．従って，このような組 (X, Y) で要件を満たすものが存在する． $\square$

さて，定理 2 は，いま示した補題から得られる次の性質によって示される．

補題 2. A_1 と A_2 が区別できるのであれば，n^2-step 以内で区別されなければならない．ただし，「n は決定性文法における変数の個数 $|\mathcal{V}|$」とする．

Proof. 仮に A_1 と A_2 が (n^2+1)-step で区別可能とする．補題から，別の組 (B_1, B_2) で，n^2-step で区別可能となるのが存在する．更に補題から，別の組 (C_1, C_2) で，最低でも (n^2-1)-step を区別に要するものが存在する．ところで，このようにして出てくる組は，全て相異なるものでなければならない．例えば $A_1 = C_1$ かつ $A_2 = C_2$ だとすると，A_1 と A_2 の区別には (n^2+1)-step 必要だったはずが，(n^2-1)-step でも区別可能であることになり，矛盾する．

このとき，(A_1, A_2) が (n^2+1)-step を要するために，この議論によって相異なる (n^2+1) 個の組が出現してしまい，矛盾． $\square$

さて，区別できない変数 A_1 と A_2 を $A_1 \sim A_2$ と書くことにする．このとき，$\sim$ は同値関係となることは，定義から明らかである．この同値関係を用いて，変数集合 $\mathcal{V}$ を直和分割 $\mathcal{V}/{\sim}$ し，同値類を $E_i \subseteq \mathcal{V}$ で書く：

$$\mathcal{V}/{\sim} = \{E_1, E_2, \ldots, E_m\}.$$

4 無用な記号の除去

主定理の証明のために，形式言語理論で良く行われる次の議論を，括弧文法に適用しておく．

与えられた括弧文法 $\mathcal{G}$ から，「無用な（*useless*）変数」を，言語を保存しつつ，取り去ることができる．変数 A は，以下のどれか一つでも満たすときに，無用であると呼ばれる：

- A から終端記号列を作ることができない（A は生成的でない）：$\neg\exists w \in \Sigma^*.\ A \Rightarrow^* w$.

- どの開始記号からも，A を含む文形式を生成できない（A に到達できない）：$\forall w_1, w_2.\ \mathcal{S} \not\Rightarrow^* w_1 A w_2$.

文法から無用な変数を除去するには，次の手順を用いれば良い：

1. ある変数 A が生成的かどうかは簡単に判定できる．生成的でない変数を，文法から完全に取り去ってしまう．

2. ある変数 A が $\mathcal{S}$ から到達可能かどうかは簡単に判定できる．到達できない変数を，文法から完全に取り去ってしまう．

この議論は，括弧文法に限らず，CFG 一般で行うことができ，特に上の二段の手続きは言語を保存することが容易に示される（Chomsky 標準形にする際などに，この手の単純化を行うため，標準的なテキストを参考にされたい）.

決定的な文法について，上の手続きを行ったとしても，既に存在する変数や生成規則などを取り除くだけであるから，得られる文法は依然として決定的である.

5　既約決定文法

ここでは，与えられた決定性文法を，既約文法に変換する操作を行う.

以下の全てを満たす括弧文法を「既約」な文法と呼ぶ：

- 決定的である.
- 無用（useless）な変数が一つも存在しない.
- 文法中の相異なる 2 つの変数 A_1 と A_2 は，かならず区別される：$A_1 \not\sim A_2$.

我々は，既に決定化と無用な変数の除去を論じたので，最後の条件を達成できることをここで見ることにする.（本節の証明には，決定性や無用な変数云々は必要でなく，同値類だけが重要である.）

定理とその証明に入る前に，一つだけ補助概念を導入しておく．文法 $\mathcal{G} = (\mathcal{V}, \Sigma, \mathcal{S}, \mathcal{P})$ の上の，前章の内容から「計算」できる，同値関係 $\sim$ について，変数 $A \in \mathcal{V}$ を代表元とする同値類を $[A] = \{A' \in \mathcal{V} : A \sim A'\}$ と書くことにする．このとき，$\mathcal{G}$ の文形式 w に関して,「対応する文形式」$[w]$ を帰納的に定義しておく：

- $w = \epsilon$ であるならば，$[w] \triangleq \epsilon$ である.
- $w = a\,w'$ かつ $a \in \Sigma$ であるならば，$[w] \triangleq a\,[w']$ である.
- 同様に $w = A\,w'$ かつ $A \in \mathcal{V}$ であるならば，$[w] \triangleq [A]\,[w']$ となる.

命題 2. $\mathcal{G}$ の開始記号 $S \in \mathcal{S}$ について，$S \Rightarrow^* w$ かつ，$[w] = [w']$ であるならば，$S \Rightarrow^* w'$ が成立する.

Proof. w が終端記号列であるならば，$w = w'$ でなければならないので，明らか．w が本質的に文形式の場合かつ，$w \neq w'$ の場合を考える．仮定 $[w] = [w']$ により，$|w| = |w'|$ である．従って，w と w' の同一位置の変数出現で相異なるものが存在する．そのような出現の最左部分に注目し，$w = w_1 A w_2$ と $w' = w_1 A' w_2'$ と書く．このとき，$A \sim A'$ であるから，$S \Rightarrow^* w$ により，$S \Rightarrow w_1 A' w_2$ も導出可能である．今度は，$w_1 A' w_2$ と $w_1 A' w_2'$ の組合せで，同じ議論を行い，これを繰り返していくと w' が導出可能であることが分かる．□

定理 3. 与えれた括弧文法 $\mathcal{G}$ を，同値関係 $\sim$ で割って得られる文法 $\mathcal{G}'$ は，$L(\mathcal{G}) = L(\mathcal{G}')$ を満たす.

Proof. $\mathcal{G}$ の変数全体を $\mathcal{V}$ と書くことにすると，$\mathcal{G}'$ の変数は，m 個の同値類 $\mathcal{V}/{\sim} = \{E_1, E_2, \ldots, E_m\}$ である．特に，括弧文法 $\mathcal{G}$ が決定的であるならば，実際に同値類が計算できることに注意.

$\mathcal{G}$ の規則

$$A_1 \to (\, A_2\, b\, c\, A_3\, A_4 \,)$$

をみたら，$\mathcal{G}'$ では $[A_1]$ から $[(\, A_2\, b\, c\, A_3\, A_4 \,)]$ を導出できるように，

$$[A_1] \to (\, [A_2]\, b\, c\, [A_3]\, [A_4] \,)$$

という規則を加えることにする.

$\mathcal{G}$ の開始記号の集合を $\mathcal{S}$ とすると，$\mathcal{G}'$ での開始記号は $\{[S] : S \in \mathcal{S}\}$ で作る．この開始記号集合の作り方の正当性は，次の主張から保証される.

主張 3. S を $\mathcal{G}$ の開始記号とすると，任意の $S' \in [S]$ は，$\mathcal{G}$ の開始記号である．

Proof. $S \sim A$ とする A を仮定する．ここで，context $\beta = [\_]$ を考えると，$S \Rightarrow^* [\,S\,]$ であるから，$\mathcal{S} \Rightarrow^* [\,A\,]$ でなければならない．もし，A が開始記号でないとすると，A が導出されるときには，$\mathcal{S} \Rightarrow^* (\ldots A \ldots)$ の形を余儀なくされる．従って，$A \in \mathcal{S}$ でなければならない． □

主張 4. $\mathcal{G} : A \Rightarrow^* w$ であるとすると，$\mathcal{G}' : [A] \Rightarrow^* [w]$ が成立する．

Proof. 長さに関する帰納法で示す．$A \Rightarrow w$ すなわち，$A \to w$ の生成規則を $\mathcal{G}$ が持つ．このとき，$[A] \to [w]$ を $\mathcal{G}'$ に加えているので，$[A] \Rightarrow [w]$ となる．

一般について，次の場合を考える

$$\mathcal{G} : A \Rightarrow^* w_1 B w_2 \Rightarrow w_1(\,w_3\,)w_2\ .$$

帰納法の仮定により，$\mathcal{G}' : [A] \Rightarrow^* [w_1][B][w_2]$ を持つ．さて，$\mathcal{G}$ では $B \to (\,w_3\,)$ を持つことになるので，$\mathcal{G}'$ でも $[B] \to (\,[w_3]\,)$ を持つから，$\mathcal{G}' : [A] \Rightarrow^* [w_1](\,[w_3]\,)[w_2]$ が成立する． □

次には，この補題の逆に相当するものを示す．

主張 5. $\mathcal{G}' : [S] \Rightarrow^* w$ であるとすると，$[w'] = w$ とする任意の w' について，$\mathcal{G} : \mathcal{S} \Rightarrow^* w'$ が成立する．

Proof. 導出の長さに関する帰納法で示す．

$\mathcal{G}' : [S] \Rightarrow (\,[w]\,)$ であるとする．これは，$S \sim S'$，$[w'] = [w]$ とする生成規則 $\mathcal{G} : S' \to (w')$ から得られるので，$\mathcal{G} : \mathcal{S} \Rightarrow (\,w'\,)$ が成立する．あとは，命題 2 により，任意の $[w''] = [w]$ について $\mathcal{G} : \mathcal{S} \Rightarrow (\,w''\,)$ が成立する．

一般の場合で，

$$\mathcal{G}' : [S] \Rightarrow^* [w_1\,A\,w_2] \Rightarrow [w_1\,(\,w_3\,)\,w_2] = [w_1]\,(\,[w_3]\,)\,[w_2]$$

を仮定する．任意の対応する文字列 $w'_1\,(\,w'_3\,)\,w'_2$ が $\mathcal{G}$ で導出されることを示せば良い．$\mathcal{G}'$ の $[A] \to (\,[w_3]\,)$ から，$\mathcal{G}$ には，$[A''] = [A]$ かつ $[w''_3] = [w_3]$ とする生成規則 $A'' \to (\,w''_3\,)$ が存在する．帰納法の仮定から，$\mathcal{G} : \mathcal{S} \Rightarrow^* w'_1\,A''\,w'_2$ がわかり，加えて以下が成立：

$$S \Rightarrow^* w'_1\,A''\,w'_2 \Rightarrow w'_1\,w''_3\,w'_2.$$

このとき，$[w_3] = [w'_3]$ であるから，命題 2 から，$S \Rightarrow^* w'_1\,w'_3\,w'_2$ が導かれる． □

主張を合わせると，$L(\mathcal{G}) = L(\mathcal{G}')$ が得られる． □

さて，既に決定的な文法については，上の構成によって得られた文法は，決定性を失うことはない．

補題 3. 上記手続きによって得られた文法 $\mathcal{G}'$ は，決定文法である．

Proof. $\mathcal{G}'$ が決定文法ではないことを仮定して，矛盾を導く．このとき，次のような生成規則が存在しなければならない：

$$[A] \to (\,[A_1]\,b\,c\,[A_2]\,[A_3]\,), \qquad [A'] \to (\,[A_1]\,b\,c\,[A_2]\,[A_3]\,).$$

ただし，$A \not\sim A'$ である．この 2 つの規則により，文法 $\mathcal{G}$ で以下が存在しなくてはならない

$$A \to (\,A_1\,b\,c\,A_2\,A_3\,), \qquad A' \to (\,A'_1\,b\,c\,A'_2\,A'_3\,).$$

ただし，$A_i \sim A'_i$ である．

さて，$A \not\sim A'$ であるから，以下のどちらかが成立する：

- ある context β で，$\beta[A]$ が導出可能だが，$\beta[A']$ は導出不能.
- ある context β' で，$\beta'[A]$ は導出不能だが，$\beta[A]$ が導出可能.

ここでは，前者であると仮定するが，後者でも全く同じ議論が可能である.

この context β について，$\beta[(A_1bcA_2A_3)]$ と $\beta[(A'_1bcA'_2A'_3)]$ は，両方とも導出可能か，両方とも導出不能でなければならない．$\beta[A]$ が導出可能であるから，$\beta[(A_1bcA_2A_3)]$ は必ず導出されるため，両方が導出可能でなくてはならない．両方導出可能である場合には，$\mathcal{G}$ の決定性により，$\beta[A]$ と $\beta[A']$ も共に導出される必要がある．しかしこれは β が A と A' を区別することに矛盾する． □

同様に，既に無用な変数が取り除かれているのであれば，得られた文法にも，無用な変数は存在しない.

補題 4. $\mathcal{G}$ が無用な変数を含まないのであれば，得られた $\mathcal{G}'$ にも無用な変数が存在しない.

Proof. $\mathcal{G}$ では，各変数 A について，$\mathcal{S} \Rightarrow^* w_1Aw_2 \Rightarrow^* w_1w_3w_2$ が保証されている．このとき，上の定理の「主張 4」により，$\mathcal{G}'$ においても $[A]$ に到達可能であることと，$[A]$ が生成的であることが分かる．従って，$[A]$ は無用な変数ではない． □

ここまでの結果をまとめると，次のことが成立する：

定理 4. 与えられた括弧文法から，既約な括弧文法で，同じ言語を受理するものが得られる.

6 同型な文法

有限オートマトンの最小化を経たのち，状態名の置き換えで移りあえるならば，有限オートマトンは等価であるという古典的な議論がある．これを，既約決定性括弧文法についても展開する.

2 つの既約決定性括弧文法 $\mathcal{G}_1$ と $\mathcal{G}_2$ が同型（isomorphic）であるとは

1. 2 つの変数間に，1 対 1 の対応が存在し，これによって生成規則が移りあえる.
2. 同じ対応によって，開始記号も移りあえる.

明らかに，2 つの文法が同型であるならば，その生成する言語は等しいが，その逆も以下で示される.

定理 5. 既約括弧文法 $\mathcal{G}_1 = (\mathcal{V}_1, \Sigma, \mathcal{S}_1, \mathcal{P}_1)$ と $\mathcal{G}_2 = (\mathcal{V}_2, \Sigma, \mathcal{S}_2, \mathcal{P}_2)$ が $L(\mathcal{G}_1) = L(\mathcal{G}_2)$ を満たすとき，$\mathcal{G}_1$ と $\mathcal{G}_2$ は同型である.

Proof. 先に証明の流れを記す：

- $\mathcal{G}_1$ の変数 A の生成する終端記号列の集合を Δ_A とする．$\mathcal{G}_2$ に $\Delta_A = \Delta_B$ とする変数 B が存在する.
- 決定性の文法では，2 つの異なる変数が，同じ終端記号列を生成することはありえない（c.f. 決定化の導く無曖昧性）
- これが一対一の対応を与えて，実際に生成規則が確かにこれで移りあえることを確認する.

A を $\mathcal{G}_1$ の変数であるとすると，A は無用ではないので，

$$\mathcal{S}_1 \Rightarrow^* w_1Aw_2 \Rightarrow^* w_1(w)w_2 \qquad (\star)$$

とできる．さて，$L(\mathcal{G}_1) = L(\mathcal{G}_2)$ であるから，$\mathcal{S}_2 \Rightarrow^* w_1Bw_2 \Rightarrow^* w_1(w)w_2$ とする B が存在する．ここで我々は，$\Delta_A = \Delta_B$ を示したい.

仮に $\Delta_A \neq \Delta_B$ とし，特に $w' \in \Delta_A \setminus \Delta_B$ を固定する（この集合が空の時には，$\Delta_B \setminus \Delta_A$ から選び，対称的な議論を展開すれば良い）.

$(\star)$ により，$w_1(w')w_2 \in L(\mathcal{G}_1) = L(\mathcal{G}_2)$ であるが，$w' \notin \Delta_B$ なので，B とは異なる B' で $w' \in \Delta_{B'}$ とするものが存在する.

context β を一つ固定し，$\beta[B]$ が導出可能だとする．このとき，$\mathcal{G}_2$ は既約文法で無用な変数が一つもないので，β から更に計算を進めることで，hole 以外は「終端記号からなる context」 β' を獲得できる．このとき，$\beta'[B]$ は当然導出可能であり，従って $\beta'[w]$ も導出可能である.

$$\mathcal{S}_2 \Rightarrow^* \beta[B] \Rightarrow^* \beta'[B] \Rightarrow^* \beta'[w] \in \Sigma^*.$$

また，$\beta'[w] \in L(\mathcal{G}_1)$ でなければならないから,（次の主張 6 により）$\beta'[w'] \in L(\mathcal{G}_1)$ が成立し，$\beta'[w'] \in L(\mathcal{G}_2)$ が要求される.

主張 6. $\mathcal{G}_1$ 上の context β' で，$\beta'[w] \in L(\mathcal{G}_1)$ かつ $w, w' \in \Delta_A$ とする．このとき，$\beta'[w'] \in L(\mathcal{G}_1)$ である.

Proof. $\mathcal{S}_1 \Rightarrow^* \beta'[w]$ であり，$w \in \Delta_A$ であるから，決定性から，必ず $\mathcal{S}_1 \Rightarrow^* \beta'[A] \Rightarrow^* \beta'[w]$ が成立する．従って，$\mathcal{S}_1 \Rightarrow^* \beta'[A] \Rightarrow^* \beta'[w']$ が成立する. □

さて，決定性によって必ず $\beta'[B']$ が導出されなければならない．β' の作り方から，$\beta[B]$ も導出可能である.

$$\mathcal{S}_2 \Rightarrow^* \beta[B] \Rightarrow^* \beta'[B] \Rightarrow^* \beta'[w'] \in \Sigma^*$$

まとめると，$\beta[B]$ が導出可能であれば $\beta[B']$ が導出可能.

全く同じ議論で，$\beta[B']$ が導出可能であれば $\beta[B]$ も導出可能となり，これはすなわち $B \sim B'$ を意味する．一方，我々は B と B' を相異なるものとしてとったわけで，これは $B \not\sim B'$ を意味するから矛盾が起こる．（一纏めにして得られる性質を利用するのは，ここが最初で最後である.）

上の $\Delta_A = \Delta_B$ の議論で，変数の対応のさせかた（と同時に，開始記号の対応）が分かったので，これが実際に同型を与えることを確認する.

$A_1 \to (A_2 bc A_3 A_4)$ を $\mathcal{G}_1$ の規則とする．このとき，$A_1 \Rightarrow^* (w_2\, bc\, w_3\, w_4)$ で $w_i \in \Sigma^*$ とするものがある．上の議論から，$B_i \Rightarrow^* w_i$ とするものが存在することは分かる．しかも，$\mathcal{G}_2$ は決定的な文法であるから，$B_1 \Rightarrow (B_2 bc B_3 B_4) \Rightarrow^* (w_2\, bc\, w_3\, w_4)$ 以外はありえないことが分かる.

□

7　主定理

定理 6. 2 つの括弧文法 $\mathcal{G}_1$ と $\mathcal{G}_2$ が，$L(\mathcal{G}_1) = L(\mathcal{G}_2)$ とするかどうかは決定可能である.

Proof. まずはじめに，以下の流れで 2 つの文法を単純化する.

1. 定理 1 によって，2 つの文法が決定化される.
2. さらに定理 4 によって，$L(\mathcal{G}_i) = L(\mathcal{G'}_i)$ とする既約括弧文法 $\mathcal{G'}_1$ と $\mathcal{G'}_2$ を得る.

最後に，定理 5 とその直前の議論により，$L(\mathcal{G'}_1) = L(\mathcal{G'}_2)$ を示すには，変数間の一対一対応で両者を同じくするものを見つけられるかどうかに，議論が帰着される．これは計算可能であるから，決定可能. □