5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2. 構文木

導出のプロセスは木構造

で表現されることが多い。

例) $P \Rightarrow (P) \Rightarrow (P+P)$

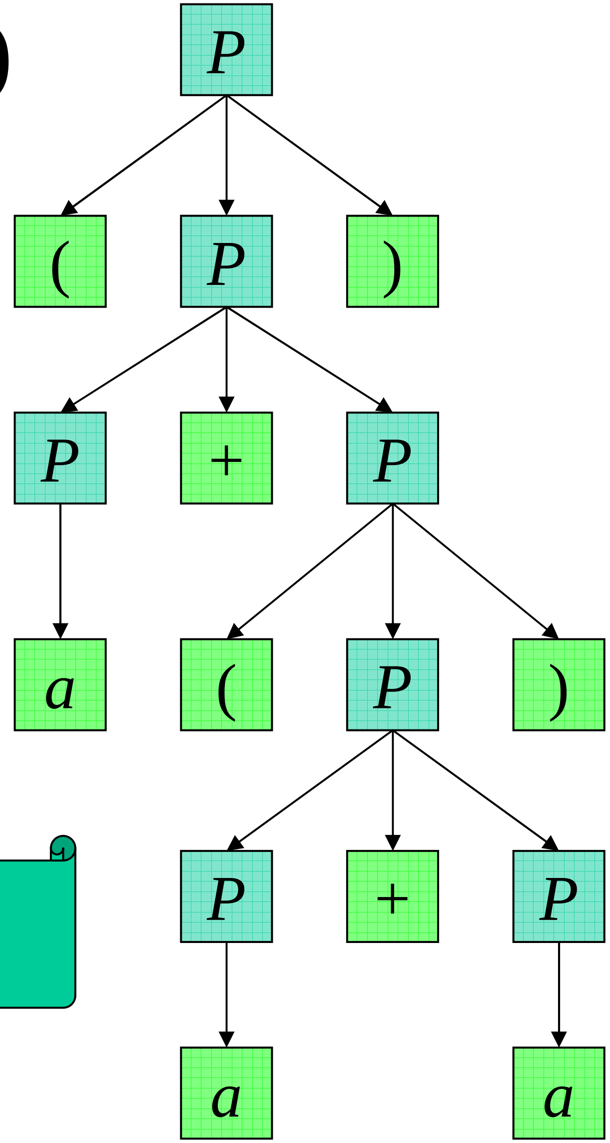
$\Rightarrow (a+P) \Rightarrow (a+(P))$

$\Rightarrow a+(P+P))$

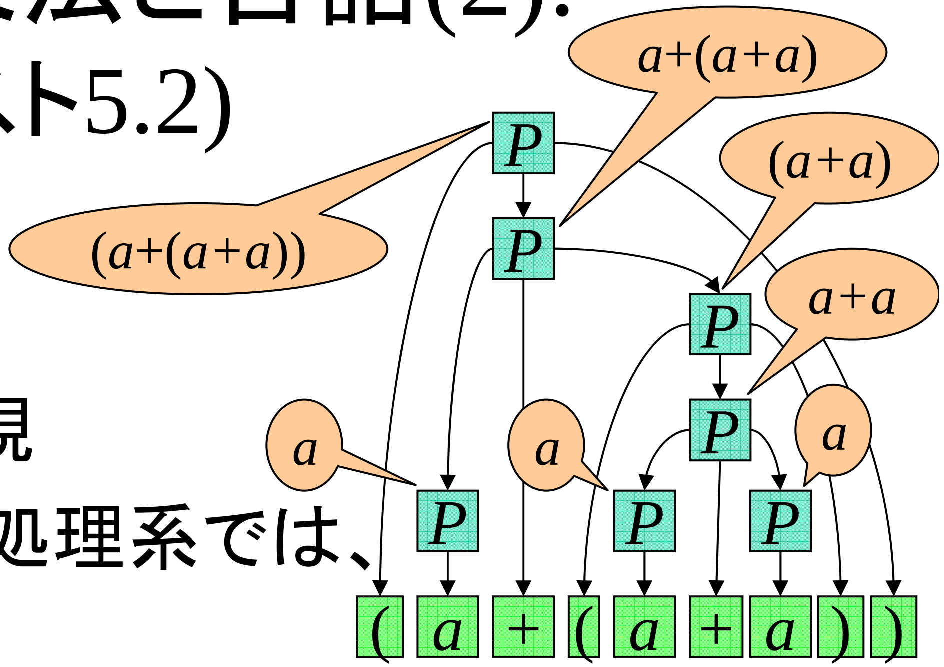
$\Rightarrow (a+(a+P))$

$\Rightarrow (a+(a+a))$

構文木



5. 文脈自由文法と言語(2):
(テキスト5.2)



5.2. 構文木

- 「語」の導出過程を表現
- コンパイラなどの言語処理系では、
    - 式
    - 命令列

などの構造を表現する標準的なデータ構造

✓ <u>与えられた語に対する構文木の構成</u>

は言語処理系では必須の機能

5. 文脈自由文法と言語(2): (テキスト5.2)

5.2. 構文木

- 曖昧な文法＝語に対する構文木が
  複数個存在する文法

  ⇒プログラミング言語では許されない

  ⇒自然言語処理でも大きな問題

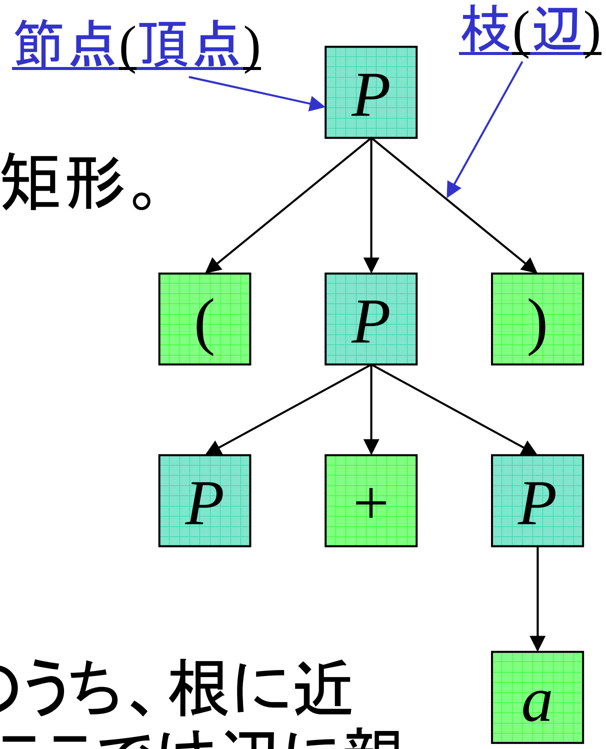
  例) Time flies like an arrow.

  「時は矢のように飛ぶ」のか？

  「時蝿は矢を好む」のか？

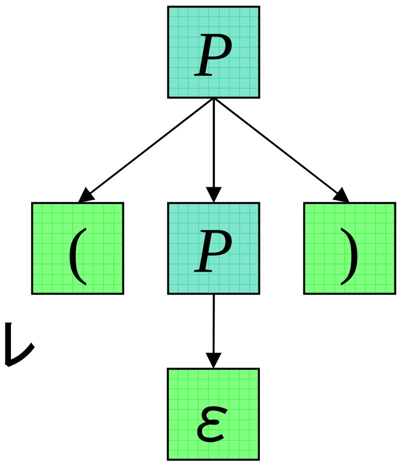
文脈自由文法の曖昧さ

5.2.* 木



- 節点(頂点): 点。ここではラベルを持つ矩形。
- 枝(辺): 二つの点を結ぶ線。
- 木: 頂点が辺で結ばれたもので、閉路がないもの。
- 根: 木の特定の1頂点。(習慣的に)一番上に描く。
- 親子関係: 辺で結ばれた二つの頂点のうち、根に近いほうを親、そうでない方を子という。(ここでは辺に親から子へ方向をつける)
- 葉: 1つの辺にしかつながっていない頂点
- 先祖/子孫:
  1. 頂点 v は頂点 v の先祖かつ子孫
  2. 頂点 v の子孫の子供は v の子孫
  3. 頂点 v の先祖の親は v の先祖
- 子供は左の子と右の子は区別される。(順序つき木)

5.2.1. 構文木の構成



文法 $G=(V,T,P,S)$ に対して、Gの**構文木**とは、

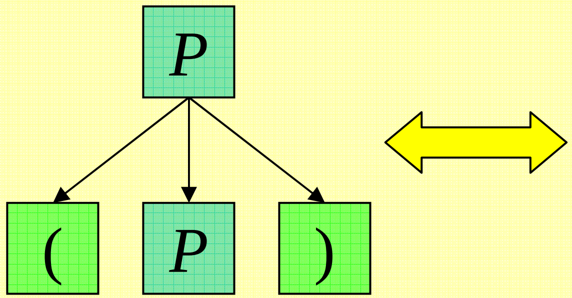
以下の条件を満たす木:

1. 葉でない頂点にはV(非終端記号)がラベル
2. 葉のラベルは次のどれか一つ
   1. Tの要素(導出が終わっている頂点)
   2. Vの要素(まだ導出途中の頂点)
   3. ε (その葉が親の唯一の子供のとき)
3. 葉でない頂点のラベルがAで、子のラベルが左から

   $$X_1, X_2, \ldots, X_k$$

   なら、G は以下の生成規則を持つ

   $$A \rightarrow X_1 X_2 \ldots X_k$$



$$P \rightarrow (P)$$

5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2.1. 構文木の構成

例)

G_p＝{{P}, {0,1}, A, P}

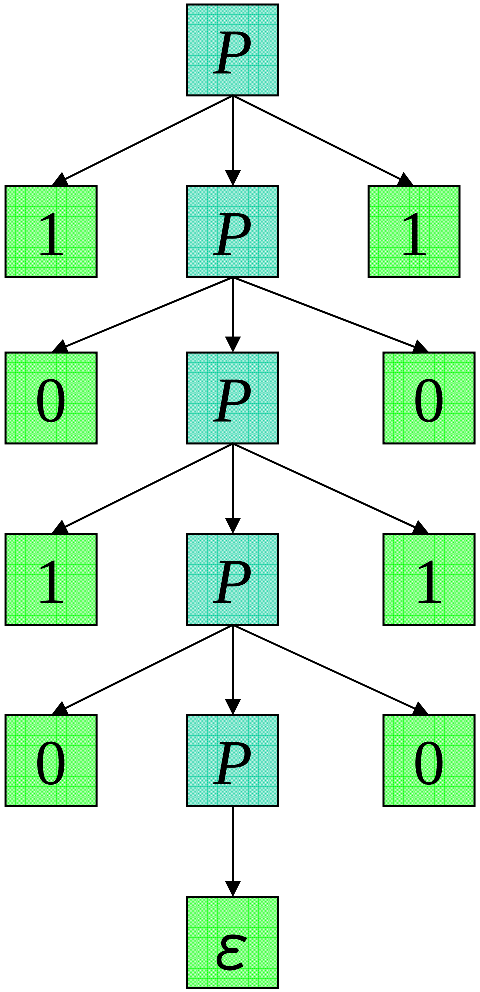
A: $P \rightarrow \varepsilon \mid 0 \mid 1 \mid 0P0 \mid 1P1$

10100101の導出

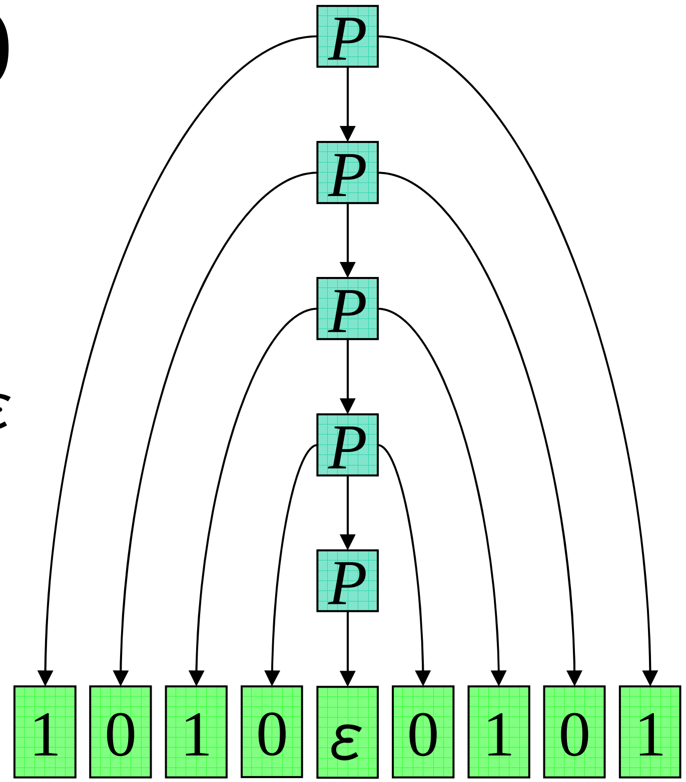
$P \Rightarrow 1P1 \Rightarrow 10P01 \Rightarrow 101P101$

$\Rightarrow 1010P0101 \Rightarrow 10100101$

10100101の構文木



5. 文脈自由文法と言語(2):
(テキスト5.2)



5.2.2. 構文木の成果

構文木が

1.　根のラベルが出発記号
2.　葉のラベルがすべて終端記号か ε

のとき、葉のラベルを左から並べた文字列を構文木の成果と言う。

(c.f. $a\varepsilon = \varepsilon a = a$)

[観測] 文法 G によって導出される語の集合

＝文法 G の構文木の成果である語の集合

10100101

5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2.3. 推論・導出・構文木

- 再帰的推論

  文字列(語＝終端記号列)から出発記号(非終端記号)

- 導出(最左導出と最右導出)

  出発記号(非終端記号)から文字列(語)

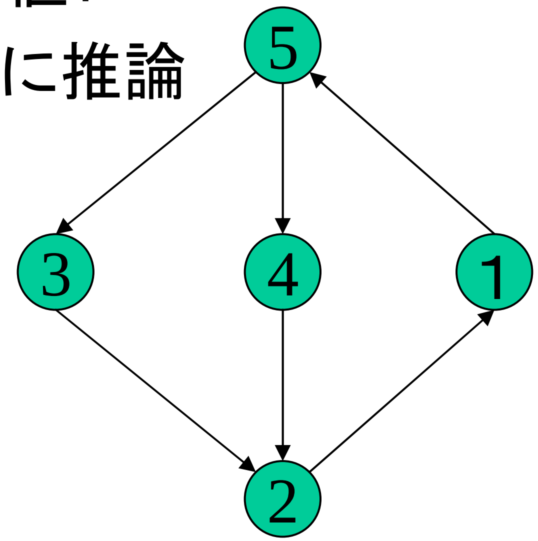
- 構文木

5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2.3. 推論・導出・構文木

文法 $G=(V,T,P,S)$ について以下はすべて同値:

1. 終端記号列 w から変数 S が再帰的に推論できる
2. $S \overset{*}{\Rightarrow} w$
3. $S \underset{左}{\overset{*}{\Rightarrow}} w$
4. $S \underset{右}{\overset{*}{\Rightarrow}} w$
5. S を根とし、w を成果とする構文木が存在。



- 3→2, 4→2 は自明
- 5→3, 5→4 は対称

1→5, 5→3, 2→1 を示す。

5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2.4. 再帰的推論から構文木へ(1→5)

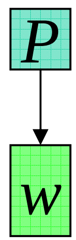
[定理] CFG $G=(V,T,P,S)$ に対し、再帰的推論で語 w が変数 S の言語に属しているなら、S を根として、w を成果とする構文木が存在する。

[証明] w が S の言語に属していることを示す導出のステップ数に関する帰納法。

[基礎] w が S から1ステップで導出できる場合

生成規則 $S \rightarrow w$ が P に入っている。

したがって構文木が存在。



5.2.4. 再帰的推論から構文木へ(1→5)

[定理] CFG G=(V,T,P,S) に対し、再帰的推論で語 w が変数 S の言語に属しているなら、S を根として、w を成果とする構文木が存在する。

[証明] w が S の言語に属していることを示す導出のステップ数に関する帰納法。

[帰納] w が S から n+1 ステップ(n＞1)で導出できる場合

[帰納法の仮定] G において語 x が変数 B の言語に属していて、かつ x が B から n ステップ以下で導出できるなら、B を根として、x を成果とする構文木が存在。

5.2.4. 再帰的推論から構文木へ(1→5)

[証明] w が S の言語に属していることを示す導出のステップ数に関する帰納法。

[帰納] w が S から n+1 ステップ(n＞1)で導出できる場合

[帰納法の仮定] G において語 x が変数 B の言語に属していて、かつ x が B から n ステップ以下で導出できるなら、B を根として、x を成果とする構文木が存在。

w は S から n+1 ステップで導出できるので、P は生成規則

$$S \rightarrow X_1 X_2 \ldots X_k$$

をもち、かつ

$$X_i \stackrel{*}{\Rightarrow} w_i$$

X_iからw_iの導出は高々 n ステップ

($X_i = w_i$ もありえる)

$$w = w_1 w_2 \ldots w_\mathrm{k}$$

を満たす文字列 $w_1, w_2, \ldots, w_k$ が存在する。

5.2.4. 再帰的推論から構文木へ(1→5)

[帰納] w が S から n+1 ステップ(n＞1)で導出できる場合

[帰納法の仮定] G において語 x が変数 B の言語に属していて、かつ x が B から n ステップ以下で導出できるなら、B を根として、x を成果とする構文木が存在。

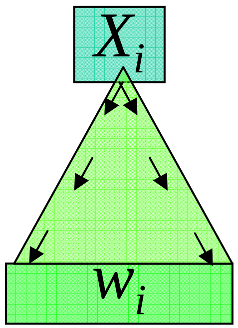
w は S から n+1 ステップで導出できるので、P は規則

$S \rightarrow X_1X_2\ldots X_k$ をもち、かつ

$X_i \overset{*}{\Rightarrow} w_i$ (nステップ以下で導出できる）

$w = w_1w_2\ldots w_{\mathrm{k}}$

を満たす文字列 $w_1, w_2, \ldots, w_k$ が存在する。



G において語 w_i は変数 X_i の言語に属し、かつ n ステップ以下で導出できるので、帰納法の仮定より、X_i を根として w_i を成果とする構文木が存在する。

5.2.4. 再帰的推論から構文木へ(1→5)

[帰納] w が S から $n+1$ ステップ($n>1$)で導出できる場合

w は S から $n+1$ ステップで導出できるので、P は生成規則

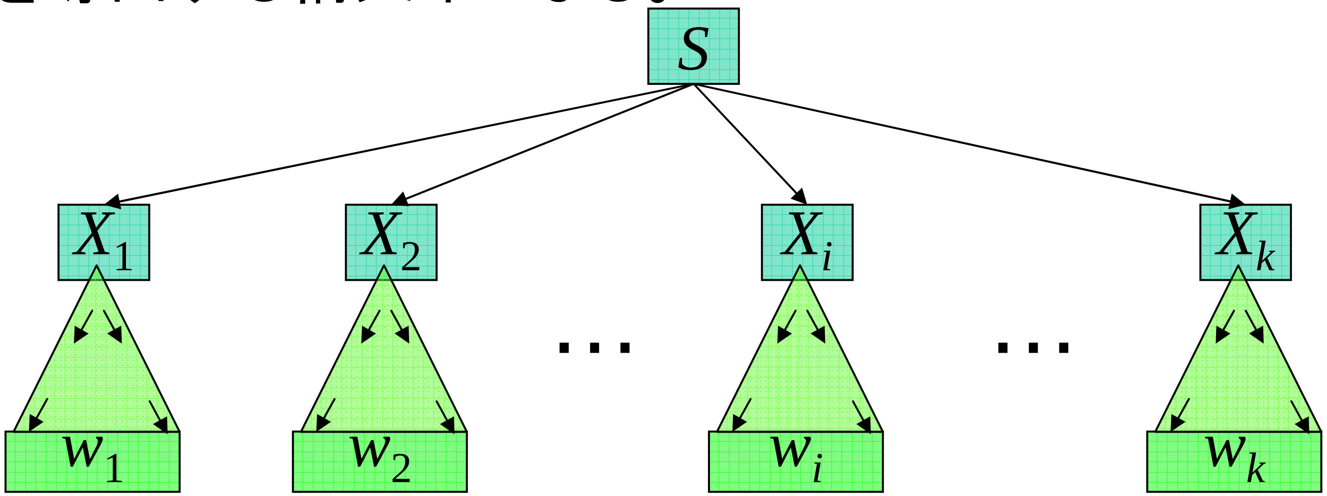
$$S \to X_1X_2\ldots X_k$$

をもち、かつ

$$X_i \overset{*}{\Rightarrow} w_i ,\ w = w_1w_2\ldots w_\mathrm{k}$$

を満たす文字列 $w_1, w_2, \ldots, w_k$ が存在する。

帰納法の仮定より、X_i を根として w_i を成果とする構文木が存在する。これらの構文木から以下の構文木を構成すると、S から w を導出する構文木となる。

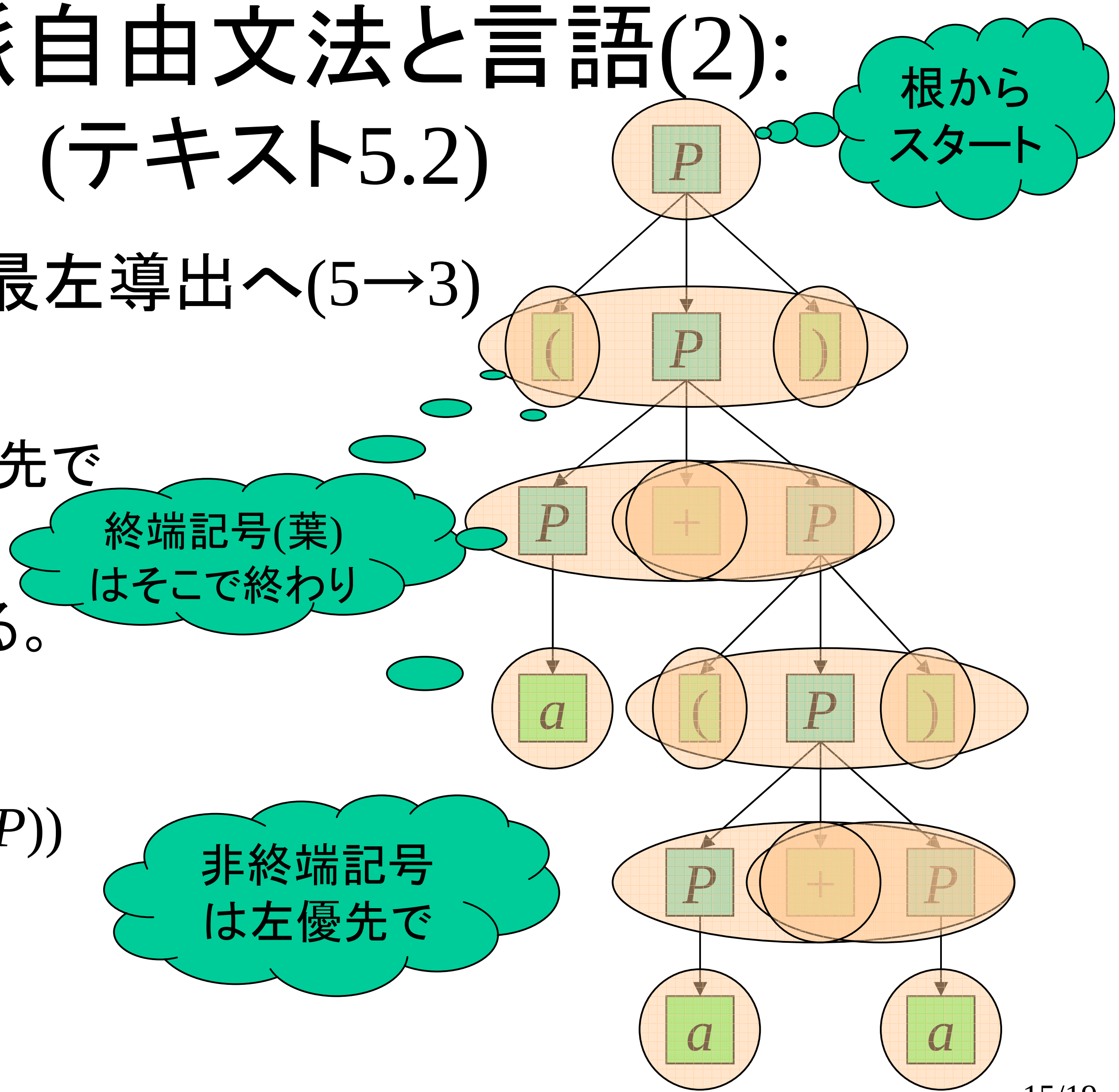


5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2.5. 構文木から最左導出へ(5→3)

直感的には…

構文木を左優先で
探索する
ことに対応する。



例) $P \Rightarrow (P) \Rightarrow (P+P)$

$\Rightarrow (a+P) \Rightarrow (a+(P))$

$\Rightarrow a+(P+P))$

$\Rightarrow (a+(a+P))$

$\Rightarrow (a+(a+a))$

5. 文脈自由文法と言語(2): (テキスト5.2)

5.2.5. 構文木から最左導出へ(5→3)

[定理] CFG $G=(V,T,P,S)$ に対し、変数 S を根とし、w を成果とする構文木があれば、G の最左導出 $S \overset{*}{\underset{左}{\Rightarrow}} w$ が存在する。

[略証] 木の高さ i についての帰納法で証明する。

(木の<u>高さ</u> = 各葉から根までの辺の個数の最大値)

木の高さが0のときは根しかないので、これは構文木ではない。したがって意味のある木の高さの最小値は1。

[基礎] $i=1$のとき: $S \rightarrow w$ が規則に入っている。

これは最左導出。

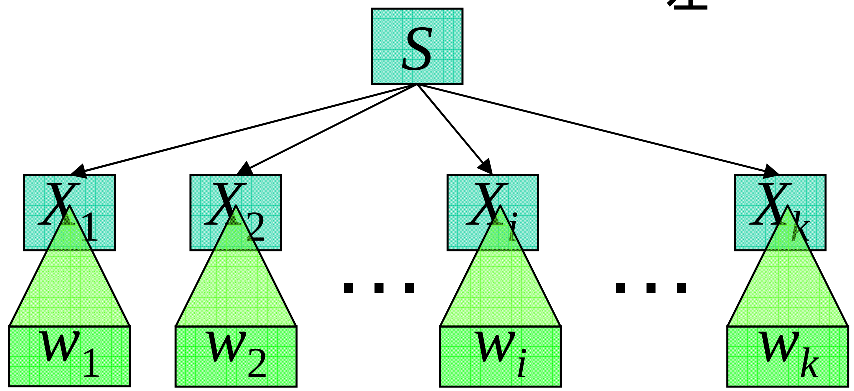
5.2.5. 構文木から最左導出へ(5→3)

[定理] CFG G=(V,T,P,S) に対し、変数 S を根とし、w を成果とする構文木があれば、G の最左導出 $S \overset{*}{\underset{左}{\Rightarrow}} w$ が存在する。

[略証] 木の高さ i についての帰納法で証明する。

[帰納] i≧2以上のとき:

- 根のラベルを S とし、S の子供のラベルを左から $X_1,X_2,\ldots,X_k$ とする。
- 帰納法の仮定から、各 X_i の成果 w_i に対して、最左導出 $X_i \overset{*}{\underset{左}{\Rightarrow}} w_i$ が存在する。
- $w = w_1 w_2 \ldots w_k$ である。



導出 $S \underset{左}{\Rightarrow} X_1 X_2 \ldots X_k$ から、
最左導出

$$S \overset{*}{\underset{左}{\Rightarrow}} w_1 w_2 \ldots w_k$$

が構成できることを示す。具体的には j=1,2,…,k について、

$$S \overset{*}{\underset{左}{\Rightarrow}} w_1 w_2 \ldots w_j X_{j+1} \ldots X_k$$

であることを j に関する帰納法で示す。（以下省略）

5. 文脈自由文法と言語(2):
(テキスト5.2)

5.2.6. 導出から再帰的推論へ(2→1)

[定理] CFG $G=(V,T,P,S)$ に対して、導出 $S\overset{*}{\Rightarrow}w$ があれば、w が S の言語に属することが再帰的推論によって確かめられる。

[略証] 導出 $S\overset{*}{\Rightarrow}w$ の長さに関する帰納法による。

[基礎] 長さが 1 のとき: $S\rightarrow w$ が生成規則に入っている。
したがって1ステップの再帰的推論により確認できる。

[帰納] 導出 $S\overset{*}{\Rightarrow}w$ の長さが n+1 とし、長さ n 以下のすべての導出が再帰的推論によって確かめられるとする。

導出は生成規則 $S\rightarrow X_1X_2\ldots X_k$ により

$$S\Rightarrow X_1X_2\ldots X_k\overset{*}{\Rightarrow}w$$

という形で表現できる。

5.2.6. 導出から再帰的推論へ(2→1)

[定理] CFG $G=(V,T,P,S)$ に対して、導出 $S\overset{*}{\Rightarrow}w$ があれば、w が S の言語に属することが再帰的推論によって確かめられる。

[略証] 導出 $S\overset{*}{\Rightarrow}w$ の長さに関する帰納法による。

[帰納] 導出 $S\overset{*}{\Rightarrow}w$ の長さが $n+1$ とし、長さ n 以下のすべての導出が再帰的推論によって確かめられるとする。

導出は生成規則 $S\rightarrow X_1X_2\ldots X_k$ により

$$S\Rightarrow X_1X_2\ldots X_k\overset{*}{\Rightarrow}w$$

という形で表現できる。さらに

» $X_i\overset{*}{\Rightarrow}w_i \quad (1\leqq i\leqq k)$

» $w=w_1w_2\ldots w_k$

であり、帰納法の仮定から、すべての導出 $X_i\overset{*}{\Rightarrow}w_i$ は再帰的推論によって確かめられる。したがって $S\rightarrow X_1X_2\ldots X_k$ とこれらの推論から、w が S の言語に属することが推論によって確かめられる。