CFG and MACRO PEG

古賀 智裕*

2016/06/17†

概 要

CFL ⊂ MACRO PEL が示せたので, ここに書き留めておく.

1 イントロダクション

MACRO PEG [2] とは, PEG [1] にマクロのような機能を持たせて PEG を拡張したものである．この文書では, CFL ⊂ MACRO PEL を示す.

2 CFG と PEG と MACRO PEG の定義

CFG はよく知られているので, その定義は省略する．PEG の定義は [1] を参照のこと．MACRO PEG については [2] を参照のこと. なお, この文書では，PEG と MACRO PEG に関する表記法は [3] で書いたものを採用する．ただし, parsing expression に使われるカッコについては, 省略可能なものは適宜省略する．また, MACRO PEG における規則の書き方は,

$$A[x_1][x_2]\cdots[x_n] \ = \ e;$$

ではなく

$$A[x_1][x_2]\cdots[x_n] \ \leftarrow \ e;$$

という書き方を用いることにする.

定義 2.1 $G = (V_N, V_T, R, S) \in \mathrm{CFG}$ と $\alpha \in (V_T \cup V_N)^*$ に対して, $L_G(\alpha) \subset V_T^*$ を自然に定義する．単に $L(\alpha)$ と書くだけでいいような気もするが, 非終端記号の置換先は R によって規定されているので，$L(\alpha)$ と書くだけでは情報が足りない. そこで, この文書では $L_G(\alpha)$ と書く. なお, この定義のもとでは，$L(G) = L_G(S)$ が成り立つ．ちなみに, $G \in \mathrm{PEG}$ や $G \in \mathrm{MACRO\ PEG}$ に対しても，このような定義は可能だが, 今回は特に必要ない.

例 1 次の CFG を G とする.

$$V_T \ \texttt{:=} \ \{\texttt{a,b}\} \quad V_N \ \texttt{:=} \ \{\texttt{S}\}$$
$$\texttt{S} \ \rightarrow \ \texttt{a S a | b}$$

*こが としひろ，toshihiro1123omega_f_ma_mgkvv_sigma_w7.dion.ne.jp _sigma_ → @ omega →

†最終更新日 2018/12/14

このとき, 次が成り立つ.

$$L(G) = L_G(S) = \{a^n ba^n \mid n \geq 0\}\,,\quad L_G(\varepsilon) = \{\varepsilon\}\,,\ L_G(a) = \{a\}\,,$$
$$L_G(Sa) = \left\{a^n ba^{n+1} \mid n \geq 0\right\}\,,\quad L_G(SS) = \left\{a^n ba^{n+m} ba^m \mid n, m \geq 0\right\}.$$

3 CFL⊂MACRO PEL

準備 3.1 $G = (V_N, V_T, R, S) \in$ CFG であって, 以下の条件を満たすものを任意に取る.

(1) G はグライバッハ標準形になっている.

(2) R の中に $S \to \varepsilon$という規則は存在しない.

このとき, $L(G) = L(G')$ を満たす $G' = (V'_N, V'_T, R', e'_S) \in$ MACRO PEG が, 以下のようにして構成できる: まず, $V'_T := V_T$ と定義する. また,

$$V'_N := \{C_A \mid A \in V_N\} \cup \{C_a \mid a \in V_T\}$$

と定義する. ただし, C_A, C_a という記号は V_N には属さないとする (必要なら V_N の元の名前を変更すればよい). 次に, R' を以下のように定義する. まず, $a \in V_T$ に対して

`Ca[s] ← a s;`

という規則を考え, これを R' に追加する. 次に, A$\in V_N$ を任意に取る. このとき,

`A → a A1 A2 ⋯ Am` (`Ai` $\in V_N - \{S\}$)

という形の規則のみが R の中に存在する[1]. そこで, MACRO PEG 側では

`CA[s] ← Ca[CA1[CA2[ ⋯ [CAm[s]] ⋯ ]]];`

という規則を考え, これを R' に追加する. ただし, 同一の A$\in V_N$ に対して, `A → a A1 A2 ⋯ Am` における右辺の表示は一通りとは限らない. その場合は, `A[s]` の方は / を使って繋いでおくことにする. たとえば, $V_T = \{0, 1\}$ として, ある A$\in V_N$ に対して

`A → 0 B D,  A → 0,  A → 1 E F H`

という 3 通りの規則が R の中に存在した場合には, 対応する `A[s]` は

`CA[s] ← C0[CB[CD[s]]] / C0[s] / C1[CE[CF[CH[s]]]];`

という定義になる. ただし, これらを並べる順番は自由にしてよくて,

`CA[s] ← C0[s] / C1[CE[CF[CH[s]]]] / C0[CB[CD[s]]];`

などと並べ替えたものを採用しても構わない. この場合, MACRO PEG での動作の仕方は変わってしまうが, 探索の順番が変わるだけで, $L(G) = L(G')$ という性質は不変である (後で示される). さて, こうして R' を定義したら, あとは e'_S (start expression, 非終端記号とは別物) を定義するのみである. これは,

[1] `A1 A2 ⋯ Am` の部分は存在しないこともある.

$$e'_S \texttt{ := CS[(.]]}$$

とするだけでよい. こうして定義した $G' = (V'_N, V'_T, R', e'_S) \in$ MACRO PEG は $L(G) = L(G')$ を満たす (後で示される).

定義 3.2 今後, `( . ]` という parsing expression を頻繁に用いる. そこで, `( . ]` のことを\$ と略記することにする. \$ は

```
$ = ( . ];
```

で定義される非終端記号**ではなくて**, `( . ]` の単なるシンタックスシュガーである. なお, `( . ]` は空文字だけを受理する parsing expression であり, 入力文字列の末尾を検出するときに役に立つ.

準備 3.3 ここから先では, 準備 3.1 で定義した G' が $L(G) = L(G')$ を満たすことを証明する. 証明の方針はシンプルであり,

$$\forall v \in V_T^* \ \ [\ v \in L(G) \Leftrightarrow v \in L(G')\ ]$$

という主張を $|v| \geq 0$ に関する数学的帰納法で示すだけである. すなわち,

$$\forall v \in V_T^* \ \ [\ v \in L_G(S) \Leftrightarrow v \text{ を } C_S[\$] \text{ に適用すると成功する }]$$

という主張を $|v| \geq 0$ に関する数学的帰納法で示すだけである. ただし, この主張そのものは帰納法では上手く行かない. そこで, 主張を敢えて強めて,

$$\forall v \in V_T^*,\ \forall A \in V_N \ \ [\ v \in L_G(A) \Leftrightarrow v \text{ を } C_A[\$] \text{ に適用すると成功する }]$$

という主張を $|v| \geq 0$ に関する数学的帰納法で示す. しかし, これでも帰納法が上手く行かない. そこで, 主張をさらに強めて,

$$\begin{aligned}&\forall v \in V_T^*,\ \forall m \geq 0,\ \forall \{A_i\}_{i=1}^m \subset V_N \\ &[\ v \in L_G(A_1 A_2 \cdots A_m) \Leftrightarrow v \text{ を } C_{A_1}[C_{A_2}[\cdots[C_{A_m}[\$]\ ]\cdots]] \text{ に適用すると成功する }]\end{aligned}$$

という主張を $|v| \geq 0$ に関する数学的帰納法で示す[2]. しかし, これでも帰納法が上手く行かない. そこで, 主張をさらに強めて,

$$\begin{aligned}&\forall v \in V_T^*,\ \forall m \geq 0,\ \forall \{b_i\}_{i=1}^m \subset V_T \cup V_N \\ &[\ v \in L_G(b_1 b_2 \cdots b_m) \Leftrightarrow v \text{ を } C_{b_1}[C_{b_2}[\cdots[C_{b_m}[\$]\ ]\cdots]] \text{ に適用すると成功する }]\end{aligned}$$

という主張を $|v| \geq 0$ に関する数学的帰納法で示す. ここまで強めれば, 帰納法が上手く行くようになる. 以下では, まず表記法を整備する.

定義 3.4 (表記法の整備) 準備 3.1 で定義した V'_N について, V'_N の任意の元は C_b $(b \in V_N \cup V_T)$ という形をしている. そこで, 任意の $C_b \in V'_N$ に対して,

$$\overline{C_b} := b$$

[2] $m = 0$ のときは, $L_G(A_1 A_2 \cdots A_m) = L_G(\$)$, $A_1[A_2[\cdots[A_m[\$]\ ]\cdots]] = \$$ と規約する.

として $\overline{C_b}$ を定義する. よって, $\overline{C_b} \in V_T \cup V_N$ が成り立つ. さらに, 一般の $\mathcal{A} \in (V_N')^*$ に対しても, $\overline{\mathcal{A}}$ を以下のように定義する:
$\mathcal{A} \neq \varepsilon$ のとき: $\mathcal{A}$ の文字列長を m とするとき, $\mathcal{A} = \mathcal{A}_1\mathcal{A}_2 \cdots \mathcal{A}_m$ と表して

$$\overline{\mathcal{A}} := \overline{\mathcal{A}_1}\ \overline{\mathcal{A}_2}\ \cdots\ \overline{\mathcal{A}_m}\ (\in (V_T \cup V_N)^+)$$

と定義する. より詳しく見ると, 各 $1 \leq i \leq m$ に対して $\mathcal{A}_i = C_{b_i}$ $(b_i \in V_T \cup V_N)$ という形であるから, $\overline{\mathcal{A}_i} = \overline{C_{b_i}} = b_i$ であり, よって

$$\overline{\mathcal{A}} := b_1 b_2 \cdots b_m$$

と定義していることになる. 同じ意味だが, $\mathcal{A} = C_{b_1}C_{b_2} \cdots C_{b_m}$ であるから,

$$\overline{C_{b_1}C_{b_2} \cdots C_{b_m}} := b_1 b_2 \cdots b_m$$

と定義していることになる.
$\mathcal{A} = \varepsilon$ のとき: ε のことを $\overline{\mathcal{A}}$ と表記する. よって. $\overline{\mathcal{A}} = \overline{\varepsilon} = \varepsilon$ である.

この定義により, たとえば, $A \in V_N,\ a \in V_T$ とするとき, $\overline{C_A} = A$ であり, $\overline{C_a} = a$ であり, $\overline{C_A C_a} = \overline{C_A}\ \overline{C_a} = Aa$ である.

定義 3.5 準備 3.1 で定義した V_N' について, $\mathcal{A} \in (V_N')^*$ を任意に取る. s を任意の記号列とするとき, $\mathcal{A}[s]$ という記法を以下のように定義する:
$\mathcal{A} \neq \varepsilon$ のとき: $\mathcal{A}$ の文字列長を m とするとき, $\mathcal{A} = \mathcal{A}_1\mathcal{A}_2 \cdots \mathcal{A}_m$ と表して

$$\mathcal{A}[s] := \mathcal{A}_1[\mathcal{A}_2[\cdots[\mathcal{A}_m[\ s\ ]\ ]\cdots]\ ]\ \ (s \text{ は任意の記号列})$$

と定義する. より詳しく見ると, 各 $1 \leq i \leq m$ に対して $\mathcal{A}_i = C_{b_i}$ $(b_i \in V_T \cup V_N)$ という形であるから,

$$\mathcal{A}[s] := C_{b_1}[C_{b_2}[\cdots[C_{b_m}[\ s\ ]\ ]\cdots]\ ]\ \ (s \text{ は任意の記号列})$$

と定義していることになる. 同じ意味だが, $\mathcal{A} = C_{b_1}C_{b_2} \cdots C_{b_m}$ であるから,

$$C_{b_1}C_{b_2} \cdots C_{b_m}[s] := C_{b_1}[C_{b_2}[\cdots[C_{b_m}[\ s\ ]\ ]\cdots]\ ]\ \ (s \text{ は任意の記号列})$$

と定義していることになる.
$\mathcal{A} = \varepsilon$ のとき: 記号列 s に対して, s 自身のことを $\mathcal{A}[s]$ と表記する. よって, $\mathcal{A}[s] = \varepsilon[s] = s$ である.

この定義により, たとえば, $A \in V_N,\ a \in V_T$ とするとき, $C_A C_a[s] = C_A[C_a[s]]$ である.

定理 3.6 次が成り立つ.

$$\begin{aligned}
&(1)\ \ \forall \mathcal{A} \in (V_N')^*\ \ [\ \overline{\mathcal{A}} \in (V_T \cup V_N)^*\ ]. \\
&(2)\ \ \forall \mathcal{A}, \mathcal{A}' \in (V_N')^*\ \ [\ \overline{\mathcal{A}\mathcal{A}'} = \overline{\mathcal{A}}\ \overline{\mathcal{A}'}\ ]. \\
&(3)\ \ \forall \mathcal{A}, \mathcal{A}' \in (V_N')^*,\ \forall s\ \ [\ \mathcal{A}\mathcal{A}'[s] = \mathcal{A}[\mathcal{A}'[s]\ ]\ ].
\end{aligned}$$

どれも明らかなので, 証明は省略する.

定理 3.7 準備 3.1 で定義した G' について, 次が成り立つ.

$$\forall \mathcal{A} \in (V'_N)^*,\ \forall v \in V_T^*\ \ [\ v \text{ を } \mathcal{A}[\$] \text{ に適用すると, 無限ループしない }].$$

証明 G と G' の構成から明らかであるが, 帰納法の練習も兼ねて, きちんと証明しておく. $n \geq 0$ に関する命題 $P(n)$ を以下のように定義する.

$$P(n):\ \forall \mathcal{A} \in (V'_N)^*,\ \forall v \in V_T^n\ \ [\ v \text{ を } \mathcal{A}[\$] \text{ に適用すると, 無限ループしない }].$$

任意の $n \geq 0$ に対して $P(n)$ が真であることを, $n \geq 0$ に関する数学的帰納法で示せばよい.
$n = 0$ のとき: $\mathcal{A} \in (V'_N)^*$ と $v \in V_T^0$ を任意に取る. $V_T^0 = \{\varepsilon\}$ だから, $v = \varepsilon$ になるしかない. $\mathcal{A} = \varepsilon$ のときは, v を $\mathcal{A}[\$] = \$$ に適用すると成功するので, 確かに無限ループしない. 以下では, $\mathcal{A} \neq \varepsilon$ としてよい. よって, $\mathcal{A} = C_b\mathcal{A}',\ \mathcal{A}' \in (V'_N)^*$ という形に表せて, $\mathcal{A}[\$] = \mathcal{A}_1[\mathcal{A}'[\$]\ ]$ となる. さて, $b \in V_N$ であるか, もしくは $b \in V_T$ であるかのいずれかである. 後者の場合は,

$$\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ] = b\ \mathcal{A}'[\$] \tag{1}$$

であるから, v をこれに適用すると, 最初の b のところで失敗する. 特に, これは無限ループしない.
$b \in V_N$ のときは, R の中における b の規則は

$$b \to a_1 B_{11} B_{12} \cdots B_{1k_1} |\ \cdots\ | a_l B_{l1} B_{l2} \cdots B_{lk_l},\quad a_i \in V_T,\quad B_{ij} \in V_N - \{S\} \tag{2}$$

という形である. よって, MACRO PEG 側の $C_b[s]$ は

$$C_b[s] \leftarrow C_{a_1}[C_{B_{11}}[\cdots[C_{B_{1k_1}}[s]\ ]\cdots]\ ]\ /\ \cdots\ /\ C_{a_l}[C_{B_{l1}}[\cdots[C_{B_{lk_l}}[s]\ ]\cdots]\ ];$$

という定義になっている. 各 $t \in [1, l]$ に対して, CFG 側の $a_t B_{t1} B_{t2} \cdots B_{tk_t}$ に対応させるために

$$\mathcal{B}_t := C_{B_{t1}} C_{B_{t2}} \cdots C_{B_{tk_t}} \in (V'_N)^+$$

と定義する. ただし, $B_{t1} B_{t2} \cdots B_{tk_t}$ の部分が存在せず, $a_t B_{t1} B_{t2} \cdots B_{tk_t}$ が a_t のみになっている場合もある. そのときは, $\mathcal{B}_t := \varepsilon$ と定義する. よって, 一般的には $\mathcal{B}_t \in (V'_N)^*$ となる. いずれにせよ, この $\mathcal{B}_t$ を用いて

$$C_b[s] \leftarrow C_{a_1}[\mathcal{B}_1[s]\ ]\ /\ \cdots\ /\ C_{a_l}[\mathcal{B}_l[s]\ ]; \tag{3}$$

と簡潔に表せる. よって, v を $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ]$ に適用することは, (3) に $s = \mathcal{A}'[\$]$ を代入したときの $C_b[\mathcal{A}'[\$]]$ に対して v を適用することを意味する. 明らかに, v は

$$C_{a_t}[\mathcal{B}_t[\mathcal{A}'[\$]]\ ] = a_t\ \mathcal{B}_t[\mathcal{A}'[\$]]\ \ (1 \leq t \leq l)$$

の先頭の a_t で失敗する. よって, v を $\mathcal{A}[\$]$ に適用すると失敗する. 特に, これは無限ループしない. よって, $P(0)$ は真である.
次に, $k \geq 0$ を任意に取る. $n = k$ のとき $P(n)$ は真であるとする. $n = k+1$ のときも $P(n)$ が真であることを言いたい. $\mathcal{A} \in (V'_N)^*$ と $v \in V_T^n$ を任意に取る. $|v| = n = k+1 \geq 1$ であるから, $v \neq \varepsilon$ である. $\mathcal{A} = \varepsilon$ のときは, v を $\mathcal{A}[\$] = \$$ に適用すると失敗するので, 無限ループしない. 以下では, $\mathcal{A} \neq \varepsilon$ としてよい. よって, $\mathcal{A} = C_b\mathcal{A}',\ \mathcal{A}' \in (V'_N)^*$ と表せて, $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ]$ となる.
$b \in V_T$ のときは, (1) に v を適用することになる. (1) の b の部分で失敗したならば, 全体でも失敗なので, 無限ループしない. また, (1) の b の部分で成功したならば, v の先頭の 1 文字は b であ

る．また, 続けて v^{+1} を $\mathcal{A}'[\$]$ に適用することになる. $|v^{+1}| = n-1 = k$ と $\mathcal{A}' \in (V_N')^*$ により，帰納法の仮定が使えて, v^{+1}を $\mathcal{A}'[\$]$ に適用すると無限ループしない．すなわち, 成功するか, もしくは失敗するかのいずれかである. よって, この場合も，全体では無限ループしない. これで, $b \in V_T$ のときは終わった. あとは，$b \in V_N$ のときが残っている． このときは, R の中における b の規則は (2) のようになっている. よって, MACRO PEG 側の $C_b[s]$ は (3) のように表せる. また, v を $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ]$ に適用することは，(3) に $s = \mathcal{A}'[\$]$ を代入したときの

$$C_{a_1}[\mathcal{B}_1[\mathcal{A}'[\$]\ ]\ ]\ /\ \cdots\ /\ C_{a_l}[\mathcal{B}_l[\mathcal{A}'[\$]\ ]\ ]$$

という parsing expression に対して v を適用することを意味する．定理 3.6 の (3) により，この parsing expression は

$$C_{a_1}[\mathcal{B}_1\mathcal{A}'[\$]\ ]\ /\ \cdots\ /\ C_{a_l}[\mathcal{B}_l\mathcal{A}'[\$]\ ]$$

と同じであり, これに\$ を適用することになる. よって, まずは $C_{a_1}[\mathcal{B}_1\mathcal{A}'[\$]\ ]$ に v を適用することになる.

$$C_{a_1}[\mathcal{B}_1\mathcal{A}'[\$]\ ] = a_1\ \mathcal{B}_1\mathcal{A}'[\$]$$

に注意して, a_1 の部分で成功するか失敗するかで場合分けする.
a_1 の部分で失敗する場合: $C_{a_1}[\mathcal{B}_1\mathcal{A}'[\$]\ ]$ で失敗となるので，$C_{a_2}[\mathcal{B}_2\mathcal{A}'[\$]\ ]$ に移ることになる.
a_1 の部分で成功する場合: v の先頭の 1 文字は a_1 となる．また, 続けて v^{+1} を $\mathcal{B}_1\mathcal{A}'[\$]$ に適用することになる．$|v^{+1}| = n-1 = k$ と $\mathcal{B}_1\mathcal{A}' \in (V_N')^*$ により，帰納法の仮定が使えて, v^{+1} を $\mathcal{B}_1\mathcal{A}'[\$]$ に適用すると無限ループしない．すなわち, 成功するか, もしくは失敗するかのいずれかである. 成功の場合は, $\mathcal{A}[\$]$ 全体で成功となり，無限ループしない. 失敗の場合は, $C_{a_1}[\mathcal{B}_1\mathcal{A}'[\$]\ ]$ のところが失敗となるので，$C_{a_2}[\mathcal{B}_2\mathcal{A}'[\$]\ ]$ に移ることになる.
この作業は帰納的に繰り返せて, $\mathcal{A}[\$]$ 全体では成功または失敗となり, 無限ループしない[3]． よって, $b \in V_N$ の場合も終わった. 以上より, $n = k+1$ のときも $P(n)$ は真である．数学的帰納法により, 任意の $n \geq 0$ に対して $P(n)$ は真である.

定理 3.8 準備 3.1 で定義した G, G' について, 次が成り立つ.

$\forall \mathcal{A} \in (V_N')^*,\ \forall v \in V_T^*\ s.t.$ 以下の 2 つの命題の真偽は一致する：
(1) v を $\mathcal{A}[\$]$ に適用すると成功する.
(2) $v \in L_G(\overline{\mathcal{A}})$ が成り立つ.

補足 (1) は MACRO PEG に関する命題であり, (2) は CFG に関する命題である.

証明 $n \geq 0$ に関する命題 $P(n)$ を以下のように定義する.

$P(n)$: $\forall \mathcal{A} \in (V_N')^*,\ \forall v \in V_T^n\ s.t.$ 以下の 2 つの命題の真偽は一致する：
(1) v を $\mathcal{A}[\$]$ に適用すると成功する.
(2) $v \in L_G(\overline{\mathcal{A}})$ が成り立つ.

[3] 具体的には,「ある $t \in [1, l]$ が存在して, $C_{a_i}[\mathcal{B}_i\mathcal{A}'[\$]\ ]$ $(1 \leq i < t)$ は失敗し，$C_{a_t}[\mathcal{B}_t\mathcal{A}'[\$]\ ]$ は成功し, ここで全体としては「成功」となる」か, あるいは「全ての $C_{a_i}[\mathcal{B}_i\mathcal{A}'[\$]\ ]$ で失敗し, 全体としては「失敗」となる」かのいずれかである.

任意の $n \geq 0$ に対して $P(n)$ が真であることを示せばよい. 数学的帰納法を使う.
$n = 0$ のとき: $\mathcal{A} \in (V'_N)^*$ と $v \in V_T^0$ を任意に取る. $V_T^0 = \{\varepsilon\}$ だから, $v = \varepsilon$ になるしかない. $\mathcal{A} = \varepsilon$ のときは, 明らかに (1) と (2) が成り立つ. よって, この場合は (1) と (2) の真偽が一致している. 以下では, $\mathcal{A} \neq \varepsilon$ としてよい. よって, $\mathcal{A} = C_b\mathcal{A}'$, $\mathcal{A}' \in (V'_N)^*$ と表せる. このとき, 定理 3.7 の証明における $n = 0$ の場合を見れば, v を $\mathcal{A}[\$]$ に適用すると失敗することが分かる. 特に, (1) は偽である. 次に, (2) も偽であることを示す. 定理 3.6 より,

$$\overline{\mathcal{A}} = \overline{C_b\mathcal{A}'} = \overline{C_b}\ \overline{\mathcal{A}'} = \ b\ \overline{\mathcal{A}'}$$

である. さて, $b \in V_N$ であるか, または $b \in V_T$ であるかのいずれかである. 後者のときは, v は $\overline{\mathcal{A}} = b\overline{\mathcal{A}'}$ には明らかにマッチせず, (2) は偽である. 前者のときは, R の中における b の規則は

$$b \to a_1B_{11}B_{12}\cdots B_{1k_1}|\ \cdots\ |a_lB_{l1}B_{l2}\cdots B_{lk_l},\quad a_i \in V_T,\quad B_{ij} \in V_N - \{S\} \qquad (*)$$

という形である. よって, もし v が $\overline{\mathcal{A}} = b\overline{\mathcal{A}'}$ にマッチするならば, v は $(*)$ における何らかの $i \in [1, l]$ に対する a_i のところにマッチしなければならないが, これは矛盾である. よって, v は $\overline{\mathcal{A}}$ にマッチしない. よって, この場合も (2) は偽である. 以上より, (1) も (2) も偽となるので, 両者の真偽は一致している. よって, $P(0)$ は真である.
次に, $k \geq 0$ を任意に取る. $n = k$ のとき, $P(n)$ は真であるとする. $n = k + 1$ のときも, $P(n)$ が真であることを言いたい. $\mathcal{A} \in (V'_N)^*$ と $v \in V_T^n$ を任意に取る. $|v| = n = k + 1 \geq 1$ であるから, $v \neq \varepsilon$ である. (1) と (2) の真偽が一致することを言いたい. 場合分けで示す.
$\mathcal{A} = \varepsilon$ のとき: $v \neq \varepsilon$ だから, (1) も (2) も成り立たないことが分かる. よって, この場合は (1) と (2) の真偽が一致している.
以下では, $\mathcal{A} \neq \varepsilon$ としてよい. よって, $\mathcal{A} = C_b\mathcal{A}'$, $\mathcal{A}' \in (V'_N)^*$ と表せる. 特に, $\overline{\mathcal{A}} = b\overline{\mathcal{A}'}$ かつ $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ]$ である. $b \in V_N$ または $b \in V_T$ のいずれかであるから, これらで場合分けする. (1) と (2) の真偽が一致することを示すには, 「(1)⇔(2)」という命題が真であることを示せばよい.
$b \in V_T$ のとき: (1) は真とする. このとき, v を $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ] = b\ \mathcal{A}'[\$]$ に適用すると成功する. よって, v の先頭の 1 文字は b であり, かつ, v^{+1} を $\mathcal{A}'[\$]$ に適用すると成功する. $|v^{+1}| = n - 1 = k$ かつ $\mathcal{A}' \in (V'_N)^*$ だから, 帰納法の仮定が使えて, $v^{+1} \in L_G(\overline{\mathcal{A}'})$ が成り立つ. よって, $v = bv^{+1} \in L_G(b\overline{\mathcal{A}'}) = L_G(\overline{\mathcal{A}})$ が成り立つ. よって, (2) は真となる. 逆に, (2) は真とする. このとき, $v \in L_G(\overline{\mathcal{A}}) = L_G(b\overline{\mathcal{A}'})$ であるから, v の先頭の 1 文字は b であり, かつ, $v^{+1} \in L_G(\overline{\mathcal{A}'})$ である. $|v^{+1}| = n - 1 = k$ かつ $\mathcal{A}' \in (V'_N)^*$ だから, 帰納法の仮定が使えて, v^{+1} を $\mathcal{A}'[\$]$ に適用すると成功する. 特に, $v = bv^{+1}$ を $b\mathcal{A}'[\$]$ に適用すると成功する. $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ] = b\ \mathcal{A}'[\$]$ だから, v を $\mathcal{A}[\$]$ に適用すると成功する. よって, (1) は真となる. 以上より, この場合は「(1)⇔(2)」という命題は真である.
$b \in V_N$ のとき: R の中における b の規則は $(*)$ のようになっている. よって, 対応する MACRO PEG 側の $C_b[s]$ は

$$C_1[s] \leftarrow C_{a_1}[C_{B_{11}}[\cdots[C_{B_{1k_1}}[s]\ ]\cdots]\ ]\ /\ \cdots\ /\ C_{a_l}[C_{B_{l1}}[\cdots[C_{B_{lk_l}}[s]\ ]\cdots]\ ];$$

という定義である. 各 $t \in [1, l]$ に対して, CFG 側の $a_tB_{t1}B_{t2}\cdots B_{tk_t}$ に対応させるために

$$\mathcal{B}_t := C_{B_{t1}}C_{B_{t2}}\cdots C_{B_{tk_t}} \in (V'_N)^+$$

と定義する. ただし, $B_{t1}B_{t2}\cdots B_{tk_t}$ の部分が存在せず, $a_tB_{t1}B_{t2}\cdots B_{tk_t}$ が a_t のみになっている場合もある. そのときは, $\mathcal{B}_t := \varepsilon$ と定義する. よって, 一般的には $\mathcal{B}_t \in (V'_N)^*$ となる. いずれに

せよ, この $\mathcal{B}_t$ を用いて

$$C_b[s] \leftarrow C_{a_1}[\mathcal{B}_1[s]\ ]\ /\ \cdots\ /\ C_{a_l}[\mathcal{B}_l[s]\ ]; \qquad (**)$$

と簡潔に表せる. また, $\mathcal{C}_t := C_{a_t}\mathcal{B}_t \in (V'_N)^*$ $(1 \le t \le l)$ と置けば,

$$A_1[s] \leftarrow \mathcal{C}_1[s]\ /\ \cdots\ /\ \mathcal{C}_l[s]; \qquad (***)$$

とも表せることになる. ここで,

$$B_{t1}B_{t2}\cdots B_{tk_t}\overline{\mathcal{A}'} = \overline{\mathcal{B}_t\mathcal{A}'}\ \ (1 \le t \le l) \qquad (****)$$

が成り立つ. なぜなら, $B_{t1}B_{t2}\cdots B_{tk_t}$ の部分が存在しないなら, (左辺) $= \overline{\mathcal{A}'}$ であり, 一方で $\mathcal{B}_t$ の定義から $\mathcal{B}_t = \varepsilon$ となるので (右辺) $= \overline{\varepsilon\mathcal{A}'} = \overline{\mathcal{A}'}$ となり, 確かに (左辺)=(右辺) である. $B_{t1}B_{t2}\cdots B_{tk_t}$ の部分が存在するなら,

$$(\text{右辺}) = \overline{\mathcal{B}_t}\ \overline{\mathcal{A}'} = \overline{C_{B_{t1}}C_{B_{t1}}\cdots C_{B_{t1}}}\ \overline{\mathcal{A}'} = \overline{C_{B_{t1}}}\ \overline{C_{B_{t1}}}\cdots\overline{C_{B_{t1}}}\ \overline{\mathcal{A}'} = B_{t1}B_{t2}\cdots B_{tk_t}\overline{\mathcal{A}'} = (\text{左辺})$$

となる. よって, $(****)$ が成り立つ. さて, もし (2) が真ならば, $v \in L_G(\overline{\mathcal{A}})$ であるから, $\overline{\mathcal{A}} = b\overline{\mathcal{A}'}$ に注意して, ある i に対して

$$v \in L_G(a_iB_{i1}B_{i2}\cdots B_{ik_i}\overline{\mathcal{A}'})$$

が成り立つ. $(****)$ により,

$$v \in L_G(a_iB_{i1}B_{i2}\cdots B_{ik_i}\overline{\mathcal{A}'}) = L_G(a_i\overline{\mathcal{B}_i\mathcal{A}'})$$

となる. 特に, v の先頭の 1 文字は a_i であり, $v = a_iv^{+1}$ と表せて, かつ $v^{+1} \in L_G(\overline{\mathcal{B}_i\mathcal{A}'})$ となる. $|v^{+1}| = n-1 = k$ かつ $\mathcal{B}_i\mathcal{A}' \in (V'_N)^*$ だから, 帰納法の仮定が使えて, v^{+1} を $\mathcal{B}_i\mathcal{A}'[\$]$ に適用すると成功する. よって, $v = a_iv^{+1}$ を $C_{a_i}[\mathcal{B}_i\mathcal{A}'[\$]\ ]$ に適用すると成功する. すなわち, v を $\mathcal{C}_i[\mathcal{A}'[\$]\ ]$ に適用すると成功する. 簡単のため, $u := \mathcal{A}'[\$]$ と略記すれば, v を $\mathcal{C}_i[u]$ に適用すると成功する. このとき, v を $C_b[u]$ に適用すると成功する. なぜなら, 定理 3.7 より, v を

$$\mathcal{C}_t[u] = \mathcal{C}_t[\mathcal{A}'[\$]\ ] = \mathcal{C}_t\mathcal{A}'[\$]\ \ (1 \le t \le l)$$

に適用すると無限ループは起こらず[4], 成功または失敗となる. また, $\mathcal{C}_i[u]$ では成功だったから, $\mathcal{C}_t[u]$ が成功するような $t \in [1, l]$ が少なくとも 1 つは存在することになる. そこで, そのような t のうち最小のものを再び t と置く. このとき, $(***)$ により, $C_b[u]$ は $\mathcal{C}_1[u], \cdots, \mathcal{C}_{t-1}[u]$ までが失敗となり, 次の $\mathcal{C}_t[u]$ で成功となるので, 結局, $C_b[u]$ 全体では成功となる. すなわち, v を $C_b[\mathcal{A}'[\$]\ ]$ に適用すると成功する. $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ]$ だったから, (1) は真となる. 逆に, (1) は真とする. このとき, v を $\mathcal{A}[\$] = C_b[\mathcal{A}'[\$]\ ]$ に適用すると成功するので, $(**)$ により, ある $i \in [1, l]$ が存在して, v を $C_{a_i}[\mathcal{B}_i[\mathcal{A}'[\$]\ ]\ ]$ に適用すると成功する.

$$C_{a_i}[\mathcal{B}_i[\mathcal{A}'[\$]\ ]\ ] = a_i\ \mathcal{B}_i[\mathcal{A}'[\$]\ ] = a_i\ \mathcal{B}_i\mathcal{A}'[\$]$$

であるから, v の先頭の 1 文字は a_i であり, かつ, v^{+1} を $\mathcal{B}_i\mathcal{A}'[\$]$ に適用すると成功する. $|v^{+1}| = n-1 = k$ かつ $\mathcal{B}_i\mathcal{A}' \in (V'_N)^*$ だから, 帰納法の仮定が使えて, $v^{+1} \in L_G(\overline{\mathcal{B}_i\mathcal{A}'})$ である. よって, $v = a_iv^{+1} \in L_G(a_i\overline{\mathcal{B}_i\mathcal{A}'})$ である. $(****)$ により,

$$v \in L_G(a_i\overline{\mathcal{B}_i\mathcal{A}'}) = L_G(a_iB_{i1}B_{i2}\cdots B_{ik_i}\overline{\mathcal{A}'})$$

[4] $\mathcal{C}_t\mathcal{A}' \in (V'_N)^*$ だから, 定理 3.7 が使える.

となる. $(*)$ により, $v \in L_G(b\overline{\mathcal{A}'})$ となる. $\overline{\mathcal{A}} = b\overline{\mathcal{A}'}$ だったから, $v \in L_G(\overline{\mathcal{A}})$ となる. よって, (2) は真となる. 以上より, この場合も「(1)⇔(2)」という命題は真である.
よって, $n = k+1$ のときも $P(n)$ は真である. 数学的帰納法により, 任意の $n \geq 0$ に対して $P(n)$ は真である.

補足 定理 3.7, 3.8 の証明において, $b \in V_N$ のとき, R の中における b の規則は

$$b \to a_1 B_{11} B_{12} \cdots B_{1k_1} |\ \cdots\ | a_l B_{l1} B_{l2} \cdots B_{lk_l},\quad a_i \in V_T,\quad B_{ij} \in V_N - \{S\}$$

という形なのだった. このとき, MACRO PEG 側の $C_b[s]$ は

$$C_b[s] \leftarrow C_{a_1}[B_{11}[\cdots[B_{1k_1}[s]\ ]\cdots]\ ]\ /\ \cdots\ /\ C_{a_l}[B_{l1}[\cdots[B_{lk_l}[s]\ ]\cdots]\ ];$$

という定義になっていると書いたが, 実は若干の語弊がある. というのも, / での繋ぎ方は並べ替えても構わないと書いたので,

$$\alpha_t := C_{a_t}[B_{t1}[\cdots[B_{tk_t}[s]\ ]\cdots]\ ]\ \ (1 \leq t \leq l)$$

と置くとき,

$$C_b[s] \leftarrow \alpha_1\ /\ \alpha_2\ /\ \cdots\ /\ \alpha_l; \tag{$*$}$$

のみならず, 一般的には

$$C_b[s] \leftarrow \alpha_{t_1}\ /\ \alpha_{t_2}\ /\ \cdots\ /\ \alpha_{t_l};\quad (t_1, t_2, \cdots, t_l) \text{ は } (1, 2, \cdots, l) \text{ の並べ替え} \tag{$**$}$$

という定義が可能だからである. $C_b[s]$ を $(**)$ の並び方で定義していた場合, 定理 3.7, 3.8 の証明中での並び方とは異なる並び方になるので, このままでは問題があるように見える. だったら $(**)$ を $(*)$ に並べ直せばいいように思えるが, MACRO PEG においては, / での順番を並べ替えると動作が変わるので, 最初に $(**)$ のように定義されているなら, それを $(*)$ のように並べ直すことは出来ない. そこで, $(**)$ の場合は, R の中における b の規則の方を並べ替えて,

$$b \to a_{t_1} B_{t_1 1} B_{t_1 2} \cdots B_{t_1 k_{t_1}} |\ \cdots\ | a_{t_l} B_{t_l 1} B_{t_l 2} \cdots B_{t_l k_{t_l}} \tag{$***$}$$

という順番になっていると解釈する. CFG では, — での区切り方の順番は意味を持たないというか, 全く本質的ではないので, $(***)$ のように解釈しても何の影響も出ない. そして, $(***)$ のように解釈すれば, 定理 3.7, 3.8 の証明中での並び方と本質的に同じ並び方になるので, 定理 3.7, 3.8 の証明がそのまま通用する. 結局, MACRO PEG 側の $C_b[s]$ の定義は, どのように並べ替えたものを採用しても常に定理 3.7, 3.8 が成り立つことになる.

系 3.9 準備 3.1 で定義した G' は, $L(G) = L(G')$ を満たす.

証明 まずは $L(G) \subset L(G')$ を示す. $v \in L(G)$ を任意に取る. $v \in L_G(S)$ である. $S \in V_N$ だから, $\mathcal{A} := C_S \in (V'_N)^*$ と置けば, $\overline{\mathcal{A}} = S$ であり, よって $v \in L_G(\overline{\mathcal{A}})$ である. 定理 3.8 より, v を $\mathcal{A}[\$]$ に適用すると成功する. すなわち, v を $S[\$]$ に適用すると成功する. すなわち, v を e'_S に適用すると成功する. よって, $v \in L(G')$ である. よって, $L(G) \subset L(G')$ である. あとは $L(G) \supset L(G')$ を示せばよい. $v \in L(G')$ を任意に取る. このとき, v を e'_S に適用すると成功する. すなわち, v を $C_S[\$]$ に適用すると成功する. $S \in V_N$ だから, $\mathcal{A} = C_S \in (V'_N)^*$ と置けば, $\mathcal{A}[\$] = C_S[\$]$ である. よって, v を $\mathcal{A}[\$]$ に適用すると成功する. 定理 3.8 より, $v \in L_G(\overline{\mathcal{A}})$ が成り立つ. $\overline{\mathcal{A}} = S$ だから, $v \in L_G(S)$ である. よって, $v \in L(G)$ である. よって, $L(G) \supset L(G')$ である. よって, $L(G) = L(G')$ である.

定理 3.10 CFL $\subset$ MACRO PEL.

証明　$L \in$ CFL を任意に取る. $\varepsilon \notin L$ のときは, $L = L(G)$ を満たす $G \in$ CFG であって, 準備 3.1 の (1),(2) を満たすものが取れる. よって, $L(G) = L(G')$ を満たす $G' \in$ MACRO PEG が取れる. よって, $L \in$ MACRO PEL となる. 以下では, $\varepsilon \in L$ としてよい. このとき, $L - \{\varepsilon\} = L(G)$ を満たす $G \in$ CFG であって, 準備 3.1 の (1),(2) を満たすものが取れる. よって, $L(G) = L(G')$ を満たす $G' = (V'_N, V'_T, R', e'_S) \in$ MACRO PEG が取れる. ここで, $G'' = (V''_N, V''_T, R'', e''_S) \in$ MACRO PEG を

$$V''_T := V'_T,\ \ V''_N := V'_N,\ \ R'' := R',\ \ e''_S :=\ \$\ /\ e'_S$$

として定義すれば, 明らかに $L(G'') = \{\varepsilon\} \cup L(G')$ となる. よって

$$L(G'') = \{\varepsilon\} \cup L(G') = \{\varepsilon\} \cup L(G) = \{\varepsilon\} \cup (L - \{\varepsilon\}) = L$$

となる. よって, $L \in$ MACRO PEL となる. よって, CFL $\subset$ MACRO PEL が成り立つ.

4　具体例

例 2 以下の CFG を考える.

V_T `:= {a,b}`　V_N `:= {S},`
`S` $\rightarrow$ `a S a | a S b | b S a | b S b | a`

この CFG が受理する言語は

$$\{(a|b)^n a(a|b)^n \mid n \geq 0\}$$

である (この言語を L と置く). この CFG はグライバッハ標準形ではないが, それでも準備 3.1 の戦略がそのまま使えて, 以下の MACRO PEG が得られる.

V'_T `:= {a,b}`　V'_N `:= {CS,Ca,Cb}`
`Ca[s]` $\leftarrow$ `a s;`
`Cb[s]` $\leftarrow$ `b s;`
`CS[s]`$\leftarrow$`Ca[CS[Ca[s]]] / Ca[CS[Cb[s]]] / Cb[CS[Ca[s]]] / Cb[CS[Cb[s]]] / Ca[s];`
e'_S `:= S[ (.]  ]`

この MACRO PEG が受理する言語は, やはり上記の L である. なお, この L は PEG では表現できないことが予想されている (はず).

例 3 次の CFG を考える.

V_T `:= {a,b}`　V_N `:= {S}`
`S` $\rightarrow$ `a S a | b S b | a a | b b`

この CFG が受理する言語は, V_T^* における回文 (ただし, 長さが 2 文字以上かつ偶数のもの) である (この言語を L と置く). この CFG はグライバッハ標準形ではないが, それでも準備 3.1 の戦略がそのまま使えて, 以下の MACRO PEG が得られる.

V_T' `:= {a,b}` V_N' `:= {CS,Ca,Cb}`
`Ca[s]` $\leftarrow$ `a s;`
`Cb[s]` $\leftarrow$ `b s;`
`CS[s]` $\leftarrow$ `Ca[Ca[s]] / Cb[Cb[s]] / Ca[CS[Ca[s]]] / Cb[CS[Cb[s]]];`
e_S' `:= S[ (.]  ]`

なお, ここでの $CS[s]$ は, いわゆる aa / bb / aSa / bSb という順番での定義になっているが, わざとそういうイタズラをしてみた. PEG でこんなことをやったら, aa と bb しか受理できなくなってしまうが, 上記の MACRO PEG の場合は, 上記の L がきちんと受理できる. なお, この L は PEG では表現できないことが予想されている.

5 考察

考察 5.1 準備 3.1 で作った G' が「なぜ上手くいくのか」を考察する. まず, 準備 3.1 では, $G \in$ CFG から $G' \in$ MACRO PEG への変換手順を記したのだったが, これと同じようにして, $G \in$ CFG から $G'' \in$ PEG への変換が考えられる. こちらの変換は, より直接的である. すなわち,

G の中にある任意の規則 $A \to e$ に対して, G'' の中に $A \leftarrow e$ という規則を入れる $(*)$

という手順にするだけでよい. たとえば, 例 3 で挙げた

V_T `:= {a,b}` V_N `:= {S}`
`S` $\to$ `a S a | b S b | a a | b b`

という CFG(これはグライバッハ標準形ではないが, ここでは気にしない) の場合は, 対応する $G'' \in$ PEG は

V_T'' `:= {a,b}` V_N'' `:= {S},`
`S` $\leftarrow$ `a S a / b S b / a a / b b`
e_S'' `:= S (.]`

となる. この場合, G'' は直観に反して複雑な PEG になっており, $L(G'')$ は偶数回文のうち一部のものしか受理できず, よって $L(G) \neq L(G'')$ が成り立つ. しかし, 上記の $(*)$ のようにして一般的に $G \in$ CFG から $G'' \in$ PEG を作ったとき, 定理 3.8 を模倣することで, $L(G) = L(G'')$ が成り立ってしまいそうに見える. しかし, $L(G) \neq L(G'')$ となる例を今見たばかりである[5]. ということは, 定理 3.8 の証明を模倣しても, どこかで証明が破綻することになる. 一体どこで破綻するのかを見れば, G' が上手く行った理由も分かるはずである. 以下で, さらに詳しく考察する.

[5]ただし, この例での G はそもそもグライバッハ標準形ではないので, 本当は適切な例ではない. G をグライバッハに直してから G'' を作っても, やはり $L(G) \neq L(G'')$ となることが確認できるので, 本当はそちらの方が適切である.

定義 5.2 $G = (V_N, V_T, R, e_S) \in$ PEG とする． $PE(V_T \cup V_N) \times PE(V_T \cup V_N)$ 上の同値関係 $\sim_{V_N,V_T,R}$ を，以下のように定義する:

$$e_1 \ \sim_{V_N,V_T,R} \ e_2 \ \Leftrightarrow_{def}$$
$$\forall v \in V_T^*,\ \forall v_f \in V_T^* \cup \{\mathtt{f}\}\ \ [\ (e_1, v, v_f) \in \rho(V_N, V_T, R) \Leftrightarrow (e_2, v, v_f) \in \rho(V_N, V_T, R)\ ]\,.$$

V_N, V_T, R が明確なときには，$\sim_{V_N,V_T,R}$ のことを単に $\sim$ と書く．大袈裟に書いたが，要するに

「 任意の入力文字列に対して，e_1と e_2の出力は同じ (無限ループの場合はともに無限ループ) 」

であるときに $e_1 \sim e_2$ と書くということである．

補題 5.3 $G = (V_N, V_T, R, e_S) \in$ PEG とする．任意の $\alpha, \beta, \gamma \in PE(V_T \cup V_N)$ に対して，

$$\gamma(\alpha/\beta) \sim \gamma\alpha/\gamma\beta$$

が成り立つ．証明は簡単なので省略する．

補題 5.4 $G = (V_N, V_T, R, e_S) \in$ PEG とする．$\alpha, \beta, \gamma \in PE(V_T \cup V_N)$ に対して，

$$(\alpha/\beta)\gamma \sim \alpha\gamma/\beta\gamma$$

は必ずしも成り立たない．特に，$\beta\gamma$ が成功するのに $(\alpha/\beta)\gamma$ は失敗するような $\alpha, \beta, \gamma \in PE(V_T \cup V_N)$ と入力文字列 $v \in V_T^*$ が存在する．

証明 $V_T = \{a, b\}$ とする．V_N, R, e_S は好きなように設定し，あとは $\alpha := a,\ \beta := aa,\ \gamma := b,\ v := aab$ とすればよい．このとき，まず $\beta\gamma = aab$ であるから，これが成功するのは明らかである．しかし，$(\alpha/\beta)\gamma$ は成功しない．なぜなら，

- αに v を適用すると成功し，v の残りの文字列は ab となる
- γに ab を適用すると失敗する

からである．

考察 5.5 補題 5.4 の証明では，要するに

$$(a/aa)b \sim ab/aab$$

が成り立たないことを示したことになる．一方で，$G' = (V_T', V_N', R', e_S') \in$ MACRO PEG を以下のように定義してみる．

V_T' `:= {a,b}` V_N' `:= { Ca,Cb,CX }`
```
Ca[s] = a s;
Cb[s] = b s;
CX[s] = Ca[s] / Ca[Ca[s]];
```
e_S' `:= CX[Cb[ (.]  ]]`

e_S' の右辺にある `CX[Cb[ (.]  ]]` を展開してみると，

```
CX[Cb[ (.]  ]]  = Ca[Cb[ (.]  ]]  / Ca[Ca[Cb[ (.]  ]]]
```

となる. この式の右辺は, いわゆる ab/aab を表現しているものと考えられる. 一方で, この式の左辺は, いわゆる $(a/aa)b$ を表現しているものと考えられる. そして, この式の両辺は, マクロの展開を行っているがゆえに, 恒等式に近い関係になっている. よって, この式全体は,

$$(a/aa)b = ab/aab$$

に近い関係を表現していると考えられる. 同じようにして, 上記の形式で MACRO PEG を作ると,

$$(\alpha/\beta)\gamma = \alpha\gamma/\beta\gamma$$

に近い関係が自然に表現できる. このような芸当は, PEG では不可能だと思われる. そして, 準備 3.1 で作った G' が「上手く行く」理由は, 実はこのあたりに存在すると考えられる. というのも, $G \in \mathrm{CFG}$ と $G'' \in \mathrm{PEG}$ に対して, 定理 3.8 の証明を模倣してみると, PEG においては一般に

$$(\alpha/\beta)\gamma \sim \alpha\gamma/\beta\gamma$$

が成り立っていないがゆえに, ある箇所で証明が失敗するからであり, なおかつ, その他の箇所では PEG でも同じ証明が通用するからである. つまり, $(\alpha/\beta)\gamma \sim \alpha\gamma/\beta\gamma$ が成り立たないという性質だけが, PEG バージョンでの障害になっている. このことを, 以下で考察する.

考察 5.6 $G \in \mathrm{CFG}$ と $G'' \in \mathrm{PEG}$ に対して, 定理 3.8 の証明を模倣してみることにする. が, その前に, まずは定理 3.7 から模倣しなければならない. 定理 3.7 の PEG 版は

$$\forall \mathcal{A} \in (V_T \cup V_N)^*,\ \forall v \in V_T^*\ [\ v \text{ を PEG において } \mathcal{A}\$ \text{に適用すると, 無限ループしない}\ ] \qquad (*)$$

というものになる. $\mathcal{A} \in (V_N')^*$ と書いていた部分は $\mathcal{A} \in (V_T \cup V_N)^*$ に変更される. また, $\mathcal{A}[s]$ と書いていた部分は $\mathcal{A}s$ 変更される. 従って, 今回は $\mathcal{A}[\$]$ とは書かず, $\mathcal{A}\$$ と書かれる. 実は, 上記の $(*)$ は普通に証明できて, しかもその証明は定理 3.7 の証明と大体同じものである. $(\alpha/\beta)\gamma \sim \alpha\gamma/\beta\gamma$ の件があるので, 一部で証明を書き換える必要があるのだが, それでも, 今は無限ループか否かを見ているだけなので, 証明に支障は出ず, $(*)$ は証明できる. さて, ここからは, 定理 3.8 の証明を模倣してみることにする. 定理 3.8 の PEG 版は

$$\begin{aligned}&\forall \mathcal{A} \in (V_T \cup V_N)^*,\ \forall v \in V_T^*\ s.t. \text{ 以下の 2 つの真偽は一致する：}\\&(1)\ \ v \text{ を PEG において } \mathcal{A}\$ \text{に適用すると成功する.}\\&(2)\ \ v \in L_G(\mathcal{A}) \text{ が成り立つ.}\end{aligned}$$

というものになる. $\overline{\mathcal{A}}$ が存在せず, そのかわりに直接的に $\mathcal{A}$ と書けばよいことに注意する. さて, 証明を模倣すると, まずは次のように命題 P(n) を定義することになる.

$$\begin{aligned}P(n):\ &\forall \mathcal{A} \in (V_T \cup V_N)^*,\ \forall v \in V_T^n\ s.t. \text{ 以下の 2 つの真偽は一致する：}\\&\quad(1)\ \ v \text{ を PEG において } \mathcal{A}\$ \text{に適用すると成功する.}\\&\quad(2)\ \ v \in L_G(\mathcal{A}) \text{ が成り立つ.}\end{aligned}$$

任意の $n \geq 0$ に対して $P(n)$ が真であることを, $n \geq 0$ に関する数学的帰納法で示したい.
$n = 0$ のときは, 実は $P(n)$ は真であることが言える. 問題は, $n = k$ のとき $P(n)$ が真であるとして, $n = k + 1$ のときにどうなるかである. 証明のほとんどの部分は, PEG 版でも通用して上手く行く (一部で書き換えが必要だが, どのみち上手く行く). ただし,

「 簡単のため, $u := \mathcal{A}'[\$]$ と略記すれば, v を $\mathcal{C}_i[u]$ に適用すると成功する.
　このとき, v を $C_b[u]$ に適用すると成功する. 」

に対応する PEG 版の部分のみ失敗する. PEG 版は

簡単のため, $u := \mathcal{A}'\$$と略記すれば, v を $\mathcal{C}_i u$ に適用すると成功する.
このとき, v を $b\ u$ に適用すると成功する.

という書き方になるので, ここが失敗することになる. なぜ失敗するのかというと, PEG において $b\ u$ を試すことは, $(\mathcal{C}_1/\mathcal{C}_2/\cdots/\mathcal{C}_l)u$ を試すことと同じだからである. もし

$$(\mathcal{C}_1/\mathcal{C}_2/\cdots/\mathcal{C}_l)u \ \sim\ \mathcal{C}_1 u/\mathcal{C}_2 u/\cdots/\mathcal{C}_l u \tag{$**$}$$

が成り立つのであれば,「v を $b\ u$ に適用すると成功する」ことが言えて, 証明に成功するのだが, 既に見たように, PEG においては一般的にはこのような同値関係は成り立たないので, もはや成功するとは限らなくなる. 実際,

- v を $\mathcal{C}_1$に適用すると成功し, v の残りの文字列は v'になる.
- v'を u に適用すると失敗する.
- v を $\mathcal{C}_i u$ に適用すると成功する (ただし $i \neq 1$).

という状況になっているならば,

- v を $(\mathcal{C}_1/\mathcal{C}_2/\cdots/\mathcal{C}_l)u$ に適用すると失敗する (よって, $b\ u$ は失敗する).
- v を $\mathcal{C}_i u$ に適用すると成功する.

となるので, $\mathcal{C}_i u$ が成功しているにも関わらず, $b\ u$ は失敗となり, ここで証明も失敗する. 一方で, もともとの MACRO PEG 版の場合は, $C_b[u]$ を試すことは

$$\mathcal{C}_1[u]\ /\ \cdots\ /\ \mathcal{C}_l[u]$$

を試すこと**そのもの**であるから, いつの間にか $(**)$ の右辺に移行できていることになる. 従って, このあたりが, G' が上手く行く理由 (かつ, G'' では上手く行かない理由) だと思われる.

考察のまとめ　というわけで, 今回の $G' \in$ MACRO PEG が上手く行く理由は, 考察 5.5 がだいたい本質的であると思われる. また, $G'' \in$ PEG では上手く行かない理由は, 補題 5.4 がだいたい本質的であると思われる.

参考文献

[1] Ford, B.: Parsing Expression Grammars: A Recognition-Based Syntactic Foundation. In: POPL 2004: Proceedings of the 31st ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages, pp. 111-122. ACM, New York (2004)

[2] Kota Mizushima: Macro PEG: PEG with macro-like rules
https://github.com/kmizu/macro_peg

[3] Toshihiro Koga: PEG and MACRO PEG
https://github.com/T-K-1/peg_and_macro_peg/blob/master/peg_and_macro_peg.pdf