SUT Journal of Mathematics
Vol. 36, No. 1 (2000), 43–82

A Comparison of Bottom-Up Pushdown Tree Transducers and Top-Down Pushdown Tree Transducers

Katsunori Yamasaki and Yoshichika Sodeshima

(Received December 3, 1999)

Abstract. In this paper we introduce a bottom-up pushdown tree transducer (b-PDTT) which results from generalizing a bottom-up tree transducer by adding a pushdown storage (where the pushdown storage have the form of trees, i.e., a tree-pushdown storage) and may be considered as a dual concept of the top-down pushdown tree transducer (t-PDTT). After proving some fundamental properties of b-PDTT, such that any b-PDTT can be realized by a linear stack with single state and converted into G-type normal form which corresponds to Greibach normal form in a context-free grammar, we compare the translational capability of a b-PDTT with that of a t-PDTT.

AMS 1991 *Mathematics Subject Classification.* 68Q68, 68Q42.

Key words and phrases. Tree automata, pushdown tree automata, finite state tree transducer (translation), pushdown tree transducer (translation).

§1. Introduction

J. W. Thatcher[20] introduced a tree automaton as an extension of a finite state automaton. The set of trees accepted by this tree automaton is called a recognizable set and it is shown that a set of derivation trees (a local set) of a context-free grammar (CFG) is recognizable. Moreover it is shown that any recognizable set is the image of a projection of a local set. Consequently, the yield of a recognizable set is a context-free language. After that, W. C. Rounds[18] introduced the context-free tree grammar (CFTG). Later the top-down pushdown tree automaton (t-PDTA) was introduced [12, 13, 24] and many properties were investigated.

From the 1970's, the field of translation schema, i.e., the translation from tree languages to tree languages was extensively investigated [5, 7, 18, 21]. This translation is called the finite state transformation (FST) by J.W.Thatcher, the finite state translation (FST) by W.C.Rounds, and the finite state tree translation (fst) by J.Engelfriet. In this paper, we call the translation device the finite state tree transducer (FST), and the translation performed by the FST is called the finite state tree translation. Such a translation includes the concepts of a syntax-directed translation (SDT) introduced by E.T.Irons[16], the attributed translation originated by D.E.Knuth[17], which can be considered as an extension of SDT, and the L-system (especially ET0L) of A. Lindenmayer (1968)[7]. In response to the above studies, we introduce a pushdown tree transducer (PDTT)[28] which may be considered as an extension of a pushdown transducer (PDT) and a finite state tree transducer (FST), and we discuss the fundamental properties of a top-down case.

In this paper we introduce a bottom-up PDTT (b-PDTT) which may be considered as a dual concept of top-down PDTT (t-PDTT) and discuss the relation between two kinds of PDTT. These b-PDTT and t-PDTT extend a bottom-up FST and a top-down FST, respectively, which are syntax-directed translations of a context-free language in compiler theory.

The organizations of this paper are as follows. In Section 2 we state some preliminary definitions and notations. In Section 3 we introduce a t-PDTT and a b-PDTT and state some related concepts to them. After that we consider the relation between t-PDTT (b-PDTT) and top-down FST (bottom-up FST), and also t-PDTA (bottom-up pushdown tree automaton). In Section 4 we show some fundamental properties of b-PDTT, such that, the final state translation is equivalent to the empty stack translation, and we can always convert a b-PDTT M into a single state b-PDTT M' which is equivalent to M. Moreover we show that any b-PDTT can be always converted into a linear stack (i.e., the stack symbols are monadic) b-PDTT with single state. In Section 5 we introduce an extended linear stack t-PDTT which is an extension of the linear stack t-PDTT and show that for any extended linear stack t-PDTT there always exists an equivalent t-PDTT. And it is shown that any b-PDTT can be converted into a G-type normal form which corresponds to Greibach normal form in CFG. In Section 6 we compare the translational capability of b-PDTT and t-PDTT, and show that the class of translations generated by nondeleting linear t-PDTTs properly contains the class of translations generated by nondeleting linear b-PDTTs. In Section 7 some conclusions are stated.

§2. Preliminaries

We first introduce basic definitions and concepts. For a denumerably infinite set $X = \{x_1, x_2, \cdots\}$ we let $X_n = \{x_1, \cdots, x_n\}$ and an element of X is called a *variable*. An alphabet Σ is ranked if $\Sigma = \bigcup_{n=0}^{\infty} \Sigma_n$, where the Σ_n are (not necessarily disjoint) subsets of Σ such that only finitely many of them are nonempty. If $\sigma \in \Sigma_n$, then we say that σ has the rank n. For a ranked alphabet Σ and a variable set X_n ($X \cap \Sigma = \emptyset$), we inductively define the set $T_\Sigma(X_n)$ as the minimum set satisfying (1) for any $\sigma \in \Sigma_0 \cup X_n$, $\sigma \in T_\Sigma(X_n)$, and for any $x \in X_n$, $x \in T_\Sigma(X_n)$, and (2) for any l with $l \geq 1$, if $t_1, \cdots, t_l \in T_\Sigma(X_n)$ and $\sigma \in \Sigma_l$ then $\sigma(t_1, \cdots, t_l) \in T_\Sigma(X_n)$. When $X_n = \emptyset$ we denote $T_\Sigma(X_n)$ as T_Σ. We call T_Σ the set of Σ-trees (or simply trees) and a subset of T_Σ is called a tree language. When a leaf of a tree is an element of X_n, that leaf is called an *indexed leaf*. Moreover, for $T_\Sigma(X_n)$ and a special symbol \$ not contained in $\Sigma \cup X$, the set $T_{\Sigma\langle\$\rangle}(X_n)$ is defined as the minimal set satisfying the following conditions: (1) $\$ \in T_{\Sigma\langle\$\rangle}$ and for any $x \in X_n$, $x \in T_{\Sigma\langle\$\rangle}(X_n)$, (2) for any $\sigma \in \Sigma_0$, $\sigma(\$) \in T_{\Sigma\langle\$\rangle}(X_n)$, and (3) for any $\sigma \in \Sigma_l$ with $l \geq 1$ and any $t_1, \cdots, t_l \in T_{\Sigma\langle\$\rangle}(X_n)$, $\sigma(t_1, \cdots, t_l) \in T_{\Sigma\langle\$\rangle}(X_n)$. We denote $T_{\Sigma\langle\$\rangle}(X_n)$ as $T_{\Sigma\langle\$\rangle}$ if $X_n = \emptyset$. For a ranked alphabet Σ and any $t \in T_{\Sigma\langle\$\rangle}(X_n)$, if the indexed leaves of t, in the order from the left, are $x_{i_1}, \cdots, x_{i_l}$ ($x_{i_j} \in X_n$ with $1 \leq j \leq l$), then we denote t as $t_{\Sigma\langle\$\rangle}\langle x_{i_1}, \cdots, x_{i_l}\rangle$. In addition, let Q be a finite set of symbols, Γ be another ranked alphabet distinct from the ranked alphabet Σ, $\sharp$ be a special symbol not contained in Γ, and Y_m be another variable set $\{y_1, \cdots, y_m\}$ distinct from the variable set X_m, where $\Gamma \cap Y = \emptyset$. (In this paper we use two distinct ranked alphabets Σ and Γ, and also use two distinct variable sets X and Y. Note that Σ, X are used for a input tree and Γ, Y for a stack tree.) For $1 \leq j \leq l$, if we substitute $([q_j, u_j], x_{i_j})$ ($q_j \in Q, u_j \in T_{\Gamma\langle\sharp\rangle}(Y_m)$) for x_{i_j} in $t_{\Sigma\langle\$\rangle}\langle x_{i_1}, \cdots, x_{i_l}\rangle$, we denote the resulting tree as $t_{\Sigma\langle\$\rangle}\langle([q_1, u_1], x_{i_1}), \cdots, ([q_l, u_l], x_{i_l})\rangle$, and $([q_j, u_j], x_{i_j})$ is called an *indexed label* instead of an indexed leaf. The set of such trees is denoted by $T_{\Sigma\langle\$\rangle}\langle[Q \times T_{\Gamma\langle\sharp\rangle}(Y_m)] \times X_n\rangle$. In addition, $t_{\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}(\emptyset)$ is simply denoted by $t_{\langle\$\rangle}$ if no confusion occurs and the notation $t_{\Sigma\langle\$\rangle}\langle t_1 \leftarrow t'_1, \cdots, t_n \leftarrow t'_n\rangle$ indicates that the subtree t_i is replaced by t'_i with $1 \leq i \leq n$. Based on the above definitions, we define a *substitution function* which is important to describe the behavior of the tree transducers.

Definition 1. (Substitution Function) For $u \in T_{\Sigma\langle\$\rangle}(X_n)$ and a sequence of trees $(t_1, \cdots, t_n)$, we inductively define a substitution function $u(t_1/x_1, \cdots, t_n/x_n)$ (or simply $u(t_i/x_i)$ if no confusion occurs) as follows:

(1) for $u = \sigma$ ($\sigma \in \Sigma_0$), $u(t_1/x_1, \cdots, t_n/x_n) = \sigma$,

(2) for $u = x_j$ ($x_j \in X_n$), $u(t_1/x_1, \cdots, t_n/x_n) = t_j$, and

(3) for any $\sigma \in \Sigma_m$ ($m \geq 1$), if $u = \sigma(u_1, \cdots, u_m)$,where $u_j \in T_{\Sigma\langle\$\rangle}(X_n)$ with $1 \leq j \leq m$,

$$u(t_1/x_1, \cdots, t_n/x_n)$$
$$= \sigma(u_1(t_1/x_1, \cdots, t_n/x_n), \cdots, u_m(t_1/x_1, \cdots, t_n/x_n)).$$

§3. Pushdown Tree Transducers and Translations

In this section we introduce a *pushdown tree transducer* (PDTT for short) which is an extension of a finite state tree transducer (or finite state tree transformation)[5], a pushdown transducer[2], and a pushdown tree automaton [12, 23]. After that we define some related concepts of PDTT such as a (*direct*) *generation*, a *generation form*, and a *translation*.

Definition 2. A pushdown tree transducer (PDTT) M is an 8-tuple $(Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F)$, where:

(1) Q is a finite set of states.

(2) Σ is a ranked alphabet of input symbols and \$ is a special symbol not contained in Σ (we call \$ the input bottom marker).

(3) Δ is a ranked alphabet of output symbols and \$' is a special symbol not contained in Δ (we call \$' the output bottom marker).

(4) Γ is a ranked alphabet of pd-symbols and $\sharp$ is a special symbol not contained in Γ (we call $\sharp$ the pd-bottom marker).

(5) q_0 is the starting state ($q_0 \in Q$).

(6) Z_0 is the starting pd-symbol ($Z_0 \in \Gamma_0$).

(7) F is a set of final states ($F \subseteq Q$).

(8) δ is a *translation function*, which is either

(a) (i) for $n \neq 0$, a mapping from $Q \times (\Sigma_n(X_n) \cup \varepsilon(X_1)) \times \Gamma_m(Y_m)$ into the finite subsets of $T_{\Delta\langle\$'\rangle}\langle[Q \times T_{\Gamma\langle\sharp\rangle}(Y_m)] \times X_n\rangle$, and

(ii) for $n = 0$, a mapping from $Q \times (\Sigma_n(\$) \cup \varepsilon(\$)) \times \Gamma_m(Y_m)$ into the finite subsets of $T_{\Delta\langle\$'\rangle}\langle[Q \times T_{\Gamma\langle\sharp\rangle}(Y_m)] \times \{\$, \$'\}\rangle$, or

(b) (i) for $n \neq 0$, a mapping from $(\Sigma_n(X_n) \cup \varepsilon(X_1)) \times ([Q \times \Gamma_{m_1}(Y_{m_1})] \times \cdots \times [Q \times \Gamma_{m_n}(Y_{m_n})])$ into the finite subsets of $[Q \times T_{\Gamma\langle\sharp\rangle}(Y_{m_1} \cup \cdots \cup Y_{m_n})] \times T_{\Delta\langle\$'\rangle}(X_n)$, and

(ii) for $n = 0$, a mapping from $(\Sigma_n(\$) \cup \varepsilon(\$)) \times [Q \times \Gamma_m(Y_m)]$ into the finite subsets of $[Q \times T_{\Gamma\langle\sharp\rangle}(Y_m)] \times T_{\Delta\langle\$'\rangle}$.

Here, $\Sigma_n(X_n) = \{\sigma(x_1, \cdots, x_n) \mid$ for any $\sigma \in \Sigma_n\}$, $\Gamma_m(Y_m) = \{Z(y_1, \cdots, y_m) \mid$ for any $Z \in \Gamma_m\}$, $\Gamma_{m_i}(Y_{m_i}) = \{Z_i(y_1, \cdots, y_{m_i}) \mid$ for any $Z \in \Gamma_{m_i}\}$ with $1 \leq i \leq n$, and ε is the empty symbol whose rank is 1 (in the case of (a)(i) and (b)(i)) or 0 (in the case of (a)(ii) and (b)(ii)). Furthermore we assume $Y_m = \{\sharp\}$ when $m = 0$, and also $Y_{m_i} = \{\sharp\}$ when $m_i = 0$ with $1 \leq i \leq n$.

In the case of (a), M is called a *top-down PDTT* (t-PDTT for short) and in the case of (b), a *bottom-up PDTT* (b-PDTT for short). Before we define the direct generation of t-PDTT (b-PDTT), we need the following definitions.

Definition 3. For a ranked alphabet Σ, a *cut* of $t \in T_\Sigma$ is a set C of nodes of t such that

(1) no two nodes in C are on the same path in t,

(2) no other node of t can be added to C without violating (1).

In addition, if $\{n_1, \cdots, n_l\} = C$ and it is ordered from the left, then we call $\{n_1, \cdots, n_l\}$ an *interior frontier node.*

Definition 4. For a t-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F)$, if $\{n_1, \cdots, n_l\}$ is the subset of a cut C of $t_{\Delta\langle\$'\rangle} \in T_{\Delta\langle\$'\rangle}$ which is ordered from the left then, for $q_i \in Q$, $u_i \in T_{\Gamma\langle\sharp\rangle}$, and $t_i \in T_{\Sigma\langle\$\rangle}$ $(1 \leq i \leq l)$,

$t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], t_1), \cdots, ([q_l, u_l], t_l)\rangle$

denotes the tree obtained by substituting $([q_i, u_i], t_i)$ for $t'_i \in T_{\Delta\langle\$'\rangle}$, where t'_i is the subtree of $t_{\Delta\langle\$'\rangle}$ whose root is n_i.

We call $t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], t_1), \cdots, ([q_l, u_l], t_l)\rangle$ an *instantaneous description* (ID for short) of M and the set of such trees is denoted by $T_{\Delta\langle\$'\rangle}\langle[Q \times T_{\Gamma\langle\sharp\rangle}(Y_m)] \times T_{\Sigma\langle\$\rangle}\rangle$. Note that, for $\{x_1, \cdots, x_l\} \subseteq X$, where X is the set of variables, the definition of $t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], x_1), \cdots, ([q_l, u_l], x_l)\rangle$ is different from the definition of $t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], t_1), \cdots, ([q_l, u_l], t_l)\rangle$ with $t_i \in T_{\Sigma\langle\$\rangle}$ $(1 \leq i \leq l)$.

Definition 5. (Direct Generation of t-PDTT) For a t-PDTT M and $t \in T_{\Delta\langle\$'\rangle}\langle[Q \times T_{\Gamma\langle\sharp\rangle}] \times T_{\Sigma\langle\$\rangle}\rangle$, a tree t is said to (directly) *generate* a tree t' in the t-PDTT M (denoted by $t \vdash_M t'$) whenever the following conditions hold: Let t be $t_{\Delta\langle\$'\rangle}\langle\cdots, ([q, u], \hat{t}), \cdots\rangle$, where $q \in Q$, $u = Z(u_1, \cdots, u_m)$ ($Z \in \Gamma_m$ and $u_i \in T_{\Gamma\langle\sharp\rangle}$ with $1 \leq i \leq m$), and $\hat{t} \in T_{\Sigma\langle\$\rangle}$.

(1) If $\hat{t} = \sigma(\$)$, where $\sigma \in \Sigma_0 \cup \{\varepsilon\}$, and the translation function applied to t is $\delta(q, \sigma(\$), Z(y_1, \cdots, y_m)) \ni t''_{\Delta\langle\$'\rangle}\langle([q'_1, u'_1], \$''_1), \cdots, ([q'_l, u'_l], \$''_l)\rangle$, where $q'_j \in Q$, $u'_j \in T_{\Gamma\langle\sharp\rangle}(Y_m)$ with $1 \leq j \leq l$, and $\$''_j \in \{\$, \$'\}$, then

$$t' = t_{\Delta\langle\$'\rangle}\langle\cdots, t''_{\Delta\langle\$'\rangle}\langle([q'_1, u'_1(u_i/y_i)], \$''_1), \cdots, ([q'_l, u'_l(u_i/y_i)], \$''_l)\rangle, \cdots\rangle.$$

(2) If $\hat{t} = \sigma(t_1, \cdots, t_n)$, where $\sigma \in \Sigma_n \cup \{\varepsilon\}$ (if $\sigma = \varepsilon$ then $\hat{t} = \varepsilon(t_1)$) with $n \geq 1$, and the translation function applied to t is $\delta(q, \sigma(x_1, \cdots, x_n), Z(y_1, \cdots, y_m)) \ni t''_{\Delta\langle\$'\rangle}\langle([q'_1, u'_1], x'_1), \cdots, ([q'_l, u'_l], x'_l)\rangle$, where $q'_j \in Q$, $u'_j \in T_{\Gamma\langle\sharp\rangle}(Y_m)$ with $1 \leq j \leq l$, and $\{x'_1, \cdots, x'_l\} \subseteq X_n$, then

$$t' = t_{\Delta\langle\$'\rangle}\langle\cdots, t''_{\Delta\langle\$'\rangle}\langle([q'_1, u'_1(u_i/y_i)], t'_1), \cdots, ([q'_l, u'_l(u_i/y_i)], t'_l)\rangle, \cdots\rangle,$$

where, for some i $(1 \leq i \leq n)$, if $x'_j = x_i$ $(1 \leq j \leq l)$ then $t'_j = t_i$.

Let $\vdash^*_M$ be the reflexive and transitive closure of $\vdash_M$, and if t_1 generates t_2 by exactly k steps then we denote this generation by $t_1 \vdash^k_M t_2$ as usual. If $([q_0, Z_0(\sharp)], t) \vdash^*_M t'$ then t' is called a *generation form*. Here, if a generation form of M is $t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], t_1), \cdots, ([q_l, u_l], t_l)\rangle$ then, for $1 \leq j \leq l$, M's input head can read one of the input symbols of the root of subtree $t_j \in T_{\Sigma\langle\$\rangle}$ with the state q_j and the stack $u_j \in T_{\Gamma\langle\sharp\rangle}$. So, if some u_j $(1 \leq j \leq l)$ is $\sharp$ then M stops its generation for the subtree t_j. From the above definitions, we can now define the *translation*s of t-PDTT as follows.

Definition 6. (Translations of t-PDTT)

(1) The translation generated by t-PDTT M with final state (abbreviated by the final state translation) is the set of pairs

$$M_F = \{\langle t, t'\rangle \in T_\Sigma \times T_\Delta \mid ([q_0, Z_0(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^*_M t'_{\Delta\langle\$'\rangle}\langle([q_1, u_1], \$'), \cdots, ([q_l, u_l], \$')\rangle, \text{ where } q_i \in F,\ u_i \in T_{\Gamma\langle\sharp\rangle}\ (1 \leq i \leq l)\}.$$

(2) The translation generated by t-PDTT M with empty stack (abbreviated by the empty stack translation) is the set of pairs

$$M_\emptyset = \{\langle t, t'\rangle \in T_\Sigma \times T_\Delta \mid ([q_0, Z_0(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^*_M t'_{\Delta\langle\$'\rangle}\langle([q_1, \sharp], \$'), \cdots, ([q_l, \sharp], \$')\rangle, \text{ where } q_i \in Q\ (1 \leq i \leq l)\}.$$

In the case of (2), we describe M as $(Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, \emptyset)$.

Remark. For a t-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F)$, we have the following immediate consequences: If any δ is of the form $\delta(q, \sigma(x_1, \cdots, x_n), Z_0(\sharp)) \subseteq T_{\Delta\langle\$'\rangle}\langle[Q \times \{Z_0(\sharp)\}] \times X_n\rangle$ with $\Gamma = \{Z_0\}$, then M is a top-down finite state tree transducer (t-FST) [5, 18, 20]. Note that $Z_0(\sharp)$ is a redundant symbol, we can omit $Z_0(\sharp)$. If $\Sigma = \Delta$ and any δ is of the form $\delta(q, \sigma(x_1, \cdots, x_n), Z(y_1, \cdots, y_m)) \ni \sigma(([p_1, u_1], x_1), \cdots, ([p_n, u_n], x_n))$ where $p_i \in Q$ and $u_i \in T_{\Gamma\langle\sharp\rangle}(Y_m)$ with $1 \leq i \leq n$, then M is a top-down pushdown tree automaton (t-PDTA) [12, 23]. We also note that the symbols $\$$ and $\$'$ are introduced for permitting the ε-generation after reading a leaf label and $\$'$ indicates the end of the generation.

Now, we shall give an example for explaining the above definitions.

Example 1. Consider the following t-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q, Z, \emptyset)$ where $Q = \{p, q\}, \Sigma = \{a, b\}$ $(\Sigma_0 = \Sigma_2 = \Sigma), \Delta = \{\sigma, \gamma, \tau\} (\Delta_0 = \Delta, \Delta_2 = \{\sigma, \tau\}), \Gamma = \{X, Y, Z\}$ $(\Gamma_0 = \Gamma, \Gamma_1 = \{Z\}, \Gamma_2 = \{X, Y\})$, and δ is defined as follows:

(1) $\delta(q, a(x_1, x_2), Z(\sharp))$
$= \{\sigma(([p, Y(X(\sharp), Y(\sharp))], x_2), \tau(([p, Y(\sharp)], x_1), \tau(([q, \sharp], \$'))))\}$,

(2) $\delta(p, b(x_1, x_2), Y(y_1, y_2)) = \{\sigma(([p, X(y_2, y_1)], x_1), \gamma(([q, \sharp], \$')))\}$,

(3) $\delta(p, b(\$), Y(\sharp)) = \{\tau(([p, Z(Z(Z(\sharp)))], \$))\}$,

(4) $\delta(p, a(\$), X(y_1, y_2)) = \{\sigma(([p, Z(\sharp)], \$))\}$,

(5) $\delta(p, \varepsilon(\$), Z(y)) = \{([p, y], \$)\}$, and

(6) $\delta(p, \varepsilon(\$), Z(\sharp)) = \{([q, \sharp], \$')\}$.

Then M translates the input tree $t = a(b, b(a, b))$ into the output tree $t' = \sigma(\sigma(\sigma, \gamma), \tau(\tau, \tau))$ by the following procedure. Here the notation $\vdash_{(k)}$ $(1 \leq k \leq 6)$ denotes that the k-th translation function is applied.

$([q, Z(\sharp)], a(b(\$), b(a(\$), b(\$))))$
$\vdash_{(1)} \sigma(([p, Y(X(\sharp), Y(\sharp))], b(a(\$), b(\$))), \tau(([p, Y(\sharp)], b(\$)), \tau(([q, \sharp], \$'))))$
$\vdash_{(2)} \sigma(\sigma(([p, X(Y(\sharp), X(\sharp))], a(\$)), \gamma(([q, \sharp], \$'))), \tau(([p, Y(\sharp)], b(\$)), \tau(([q, \sharp],$
$\$'))))$
$\vdash_{(3)} \sigma(\sigma(([p, X(Y(\sharp), X(\sharp))], a(\$)), \gamma(([q, \sharp], \$'))), \tau(\tau([p, Z(Z(Z(\sharp)))], \$),$
$\tau(([q, \sharp], \$'))))$
$\vdash_{(4)} \sigma(\sigma(\sigma(([p, Z(\sharp)], \$)), \gamma(([q, \sharp], \$'))), \tau(\tau(([p, Z(Z(Z(\sharp)))], \$)), \tau(([q, \sharp],$
$\$'))))$
$\vdash^2_{(5)} \sigma(\sigma(\sigma(([p, Z(\sharp)], \$)), \gamma(([q, \sharp], \$'))), \tau(\tau(([p, Z(\sharp)], \$)), \tau(([q, \sharp], \$'))))$
$\vdash_{(6)} \sigma(\sigma(\sigma(([q, \sharp], \$')), \gamma(([q, \sharp], \$'))), \tau(\tau(([q, \sharp], \$')), \tau(([q, \sharp], \$'))))$

Therefore, we obtain $\langle t, t' \rangle \in M_\emptyset$.

Next, we consider the case of b-PDTT. As shown in the definition of b-PDTT, the translation function is not easy to understand, so we will exhibit the translation function more concretely.

(1) For any $\sigma \in \Sigma_n \cup \{\varepsilon\}$ $(n \geq 1)$, $q_i \in Q$ and $Z_i \in \Gamma_{m_i}$ $(1 \leq i \leq n)$,

$$\delta(\sigma(x_1, \cdots, x_n), ([q_1, Z_1(y_{11}, \cdots, y_{1m_1})], \cdots, [q_n, Z_n(y_{n1}, \cdots, y_{nm_n})])) \ni ([p, u], t')$$
$$(p \in Q, u \in T_{\Gamma\langle\sharp\rangle}(Y_{m_1+\cdots+m_n}), t' \in T_{\Delta\langle\$'\rangle}(X_n)),$$

(2) For any $\sigma \in \Sigma_0 \cup \{\varepsilon\}$, $q \in Q$ and $Z \in \Gamma_m$,

$$\delta(\sigma(\$), [q, Z(y_1, \cdots, y_m)]) \ni ([p, u], t')$$
$$(p \in Q, u \in T_{\Gamma\langle\sharp\rangle}(Y_m), t' \in T_{\Delta\langle\$'\rangle}).$$

Since the notation of the translation function, especially the notation of the variable, is somewhat complicated, we use the following abbreviations for making the subsequent arguments simple. (1) For the series of $y_1, \cdots, y_m$, $y_1, \cdots, y_m$ is abbreviated as $\overrightarrow{y}$ if m is known. Therefore, $Z(y_1, \cdots, y_m)$ is abbreviated as $Z(\overrightarrow{y})$ for $Z \in \Gamma_m$ and $Z_i(\overrightarrow{y_i})$ means $Z_i(y_{i1}, \cdots, y_{im})$ for $Z_i \in \Gamma_m$. (2) $Z_1(y_{11}, \cdots, y_{1k_1}), \cdots, Z_n(y_{n1}, \cdots, y_{nk_n})$ in $\delta(\sigma(x_1, \cdots, x_n), ([q_1, Z_1(y_{11}, \cdots, y_{1k_1})], \cdots, [q_n, Z_n(y_{n1}, \cdots, y_{nk_n})]))$ is abbreviated as $Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n})$. (3) For $\overrightarrow{y_1} : y_{11}, \cdots, y_{1k_1}, \;\cdots\; , \overrightarrow{y_n} : y_{n1}, \cdots, y_{nk_n}$, $(y_{11}, \cdots, y_{1k_1}, \cdots, y_{n1}, \cdots, y_{nk_n})$ is abbreviated as $(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n})$. (4) For $\sigma \in \Sigma_n \cup \{\varepsilon\}$ $(n \geq 0)$ and any $Z_i \in \Gamma$ $(1 \leq i \leq n)$, the notation $\delta(\sigma(\overrightarrow{x}), ([q_1, Z_1(\overrightarrow{y_1})], \cdots, [q_n, Z_n(\overrightarrow{y_n})])) \ni ([p, u], t')$ includes $\delta(\sigma(\$), [q, Z(\overrightarrow{y})]) \ni ([p, u], t')$ if no confusion occurs. Under above abbreviations, the translation function is denoted by $\delta(\sigma(\overrightarrow{x}), ([q_1, Z_1(\overrightarrow{y_1})], \cdots, [q_n, Z_n(\overrightarrow{y_n})])) \ni ([p, u], t')$ where $t' \in T_{\Delta\langle\$'\rangle}(X_n)$, for $\sigma \in \Sigma_n \cup \{\varepsilon\}$ with $n \geq 0$.

Definition 7. For a b-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F)$, if $\{n_1, \cdots, n_l\}$ is an interior frontier node of $t_{\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}$, then for $q_i \in Q$, $u_i \in T_{\Gamma\langle\sharp\rangle}$, and $t_i \in T_{\Delta\langle\$'\rangle}$ $(1 \leq i \leq l)$,

$$t_{\Sigma\langle\$\rangle}[([q_1, u_1], t_1), \cdots, ([q_l, u_l], t_l)]$$

denotes the tree obtained by substituting $([q_i, u_i], t_i)$ for $t'_i \in T_{\Sigma\langle\$\rangle}$, where t'_i is the subtree of $t_{\Sigma\langle\$\rangle}$ whose root is n_i.

We call $t_{\Sigma\langle\$\rangle}[([q_1, u_1], t_1), \cdots, ([q_l, u_l], t_l)]$ an *instantaneous description* (ID for short) of M and the set of such trees is denoted by $T_{\Sigma\langle\$\rangle}[[Q \times T_{\Gamma\langle\sharp\rangle}(Y_m)] \times T_{\Delta\langle\$'\rangle}]$.

Definition 8. (Direct Generation of b-PDTT) For a b-PDTT M and $t \in T_{\Sigma\langle\$\rangle}[[Q \times T_{\Gamma\langle\sharp\rangle}] \times T_{\Delta\langle\$'\rangle}]$, a tree t is said to (directly) *generate* a tree t' in the b-PDTT M (denoted by $t \vdash_M t'$) whenever the following conditions hold: Let t be $t_{\Sigma\langle\$\rangle}[\cdots, \hat{t}, \cdots]$.

(1) If $\hat{t} = \sigma(([q, u], \$))$, where $\sigma \in \Sigma_0 \cup \{\varepsilon\}$, $q \in Q$, $u = Z(u_1, \cdots, u_m)$ $(Z \in \Gamma_m$ and $u_i \in T_{\Gamma\langle\sharp\rangle}$ with $1 \leq i \leq m)$, and the translation function applied to t is $\delta(\sigma(\$), [q, Z(y_1, \cdots, y_m)]) \ni ([q', u'], t'')$, where $q' \in Q$, $u' \in T_{\Gamma\langle\sharp\rangle}(Y_m)$, and $t'' \in T_{\Delta\langle\$'\rangle}$, then

$$t' = t_{\Sigma\langle\$\rangle}[\cdots, ([q', u'(u_1/y_1, \cdots, u_m/y_m)], t''), \cdots].$$

(2) If $\hat{t} = \sigma(([q_1, u_1], t_1), \cdots, ([q_n, u_n], t_n))$, where $\sigma \in \Sigma_n \cup \{\varepsilon\}$ with $n \geq 1$, $q_j \in Q$, $u_j = Z_j(u_{j1}, \cdots, u_{jm_j})$ $(m_j \geq 1$, $Z_j \in \Gamma_{m_j}$ and $u_{ji} \in T_{\Gamma\langle\sharp\rangle}$ with $1 \leq i \leq m_j)$ with $1 \leq j \leq n$, and the translation function applied to t is $\delta(\sigma(x_1, \cdots, x_n), ([q_1, Z_1(y_1, \cdots, y_{m_1})], \cdots, [q_n, Z_n(y_{m_1+\cdots+m_{n-1}+1}, \cdots, y_{m_1+\cdots+m_n})]) \ni ([q', u'], t'')$, where $q' \in Q$, $u' \in T_{\Gamma\langle\sharp\rangle}(Y_{m_1+\cdots+m_n})$, and $t'' \in T_{\Delta\langle\$'\rangle}(X_n)$, then

$$t' = t_{\Sigma\langle\$\rangle}[\cdots, ([q', u'(u_{11}/y_1, \cdots, u_{1m_1}/y_{m_1}, \cdots, u_{n1}/ y_{m_1+\cdots+m_{n-1}+1}, \cdots, u_{nm_n}/y_{m_1+\cdots+m_n})], t''(t_1/x_1, \cdots, t_n/x_n)), \cdots].$$

Let $\vdash_M^*$ be the reflexive and transitive closure of $\vdash_M$, and if t_1 generates t_2 by exactly k steps then we denote this generation by $t_1 \vdash_M^k t_2$ as usual. If $t_{\Sigma\langle\$\rangle}[([q_0, Z_0(\sharp)], \$), \cdots, ([q_0, Z_0(\sharp)], \$)] \vdash_M^* t'$ then t' is called a *generation form*. From the above definitions, we can now define the *translation*s of b-PDTT as follows.

Definition 9. (Translations of b-PDTT)

(1) The translation generated by b-PDTT M with final state (abbreviated by the final state translation) is the set of pairs

$$M_F = \{\langle t, t'\rangle \in T_\Sigma \times T_\Delta \mid t_{\Sigma\langle\$\rangle}[([q_0, Z_0(\sharp)], \$), \cdots, ([q_0, Z_0(\sharp)], \$)] \vdash_M^* ([q, u], t'_{\Delta\langle\$'\rangle}) \text{ where } q \in F,\ u \in T_{\Gamma\langle\sharp\rangle}\}.$$

(2) The translation generated by b-PDTT M with empty stack (abbreviated by the empty stack translation) is the set of pairs

$$M_\emptyset = \{\langle t, t'\rangle \in T_\Sigma \times T_\Delta \mid t_{\Sigma\langle\$\rangle}[([q_0, Z_0(\sharp)], \$), \cdots, ([q_0, Z_0(\sharp)], \$)] \vdash_M^* ([q, \sharp], t'_{\Delta\langle\$'\rangle}) \text{ where } q \in Q\}.$$

In the case of (2), we describe M as $(Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, \emptyset)$ same as t-PDTT.

Remark. For a b-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, \emptyset)$, we have the following immediate consequences: If any δ is of the form $\delta(\sigma(x_1, \cdots, x_n), ([q_1, Z_0(\sharp)], \cdots, [q_n, Z_0(\sharp)])) \ni [Q \times \{Z_0(\sharp)\}] \times T_{\Delta\langle\$'\rangle}(X_n)$ with $\Gamma = \{Z_0\}$, then M is a bottom-up finite state tree transducer (b-FST) [5, 18, 20]. Note that $Z_0(\sharp)$ is a redundant symbol, so we can omit $Z_0(\sharp)$ the same as t-FST. If $\Sigma = \Delta$ and any δ is of the form $\delta(\sigma(x_1, \cdots, x_n), ([q_1, u_1], \cdots, [q_n, u_n])) \ni ([q, u], \sigma(x_1, \cdots, x_n))$ with $q \in Q$ and $u \in T_{\Gamma\langle\sharp\rangle}(Y_m)$, then M is a bottom-up pushdown tree automaton (b-PDTA) [25]. We also note that the symbols \$ and \$$'$ are not necessary in a b-PDTT since reading a leaf label before ε- generation does not depend on \$ and \$$'$. Nevertheless, for the sake of representational matching of t-PDTT and b-PDTT, we have \$ and \$$'$ remaining.

In the same way as the t-PDTT case, we shall give an example for explaining the above definitions.

Example 2. Consider the following b-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q, Z, \emptyset)$ where $Q = \{p, q\}, \Sigma = \{a, b\}$ $(\Sigma_0 = \Sigma_2 = \Sigma), \Delta = \{\sigma, \gamma, \tau\} (\Delta_0 = \Delta, \Delta_2 = \{\sigma, \tau\}), \Gamma = \{X, Y, Z\}$ $(\Gamma_0 = \Gamma, \Gamma_2 = \{X, Y\})$, and δ is defined as follows:

(1) $\delta(a(\$), [p, Z(\sharp)]) = \{([q, X(Y(\sharp), X(\sharp))], \sigma(\$'))\}$,

(2) $\delta(b(\$), [q, Z(\sharp)]) = \{([p, Y(\sharp)], \tau(\$'))\}$,

(3) $\delta(b(x_1, x_2), ([p, X(y_1, y_2)], [p, Y(\sharp)])) = \{([p, Y(y_2, y_2)], \sigma(x_1, \gamma(\$')))\}$, and

(4) $\delta(a(x_1, x_2), ([p, Y(\sharp)], [p, Y(y_1, y_2)])) = \{([p, \sharp], \sigma(x_2, \tau(x_1, \tau(\$'))))\}$.

Then M translates the input tree $t = a(b, b(a, b))$ into the output tree $t' = \sigma(\sigma(\sigma, \gamma), \tau(\tau, \tau))$ by the following procedure. Here the notation $\vdash_{(k)}$ $(1 \leq k \leq 4)$ denotes that the k-th translation function is applied.

$a(b(([q, Z(\sharp)], \$)), b(a(([q, Z(\sharp)], \$)), b(([q, Z(\sharp)], \$))))$

$\vdash_{(1)} a(b(([q, Z(\sharp)], \$)), b(([p, X(Y(\sharp), X(\sharp))], \sigma(\$')), b(([q, Z(\sharp)], \$))))$

$\vdash^2_{(2)} a(([p, Y(\sharp)], \tau(\$')), b(([p, X(Y(\sharp), X(\sharp))], \sigma(\$')), ([p, Y(\sharp)], \tau(\$')))$

$\vdash_{(3)} a(([p, Y(\sharp)], \tau(\$')), ([p, Y(X(\sharp), X(\sharp))], \sigma(\sigma(\$'), \gamma(\$'))))$

$\vdash_{(4)} ([p, \sharp], \sigma(\sigma(\sigma(\$'), \gamma(\$')), \tau(\tau(\$'), \tau(\$'))))$.

Therefore, we obtain $\langle t, t' \rangle \in M_\emptyset$.

Finally we define the equivalency of PDTT M^1 and M^2 as follows.

Definition 10. For a t-PDTT or b-PDTT M^1 and M^2, a translation generated by M^1 is equivalent to a translation generated by M^2 if $M^1_F = M^2_F$ (or $M^1_\emptyset = M^2_\emptyset$). In addition, in the case of $M^1_F = M^2_F$ (or $M^1_\emptyset = M^2_\emptyset$) we say that M^1 *is F-equivalent* (or *$\emptyset$-equivalent*) *to* M^2, and conversely.

§4. Fundamental Properties of PDTT

In this section we show some fundamental properties of b-PDTT, such that (1) the final state translation is equivalent to the empty stack translation and conversely, (2) any b-PDTT can be converted into a single state b-PDTT, and (3) there exists a symmetric stack form in b-PDTT. Moreover, for any t-PDTT, we can convert a t-PDTT with a tree structure of stack into a t-PDTT with a linear structure of a stack, i.e., a stack symbol is monadic (the stack symbol is monadic if and only if $\Gamma_n = \emptyset$ with $n \geq 2$). And, in the case of b-PDTT, a linear structure of a stack and a single state are concluded. In the top-down case, the above properties are already shown in [28], so we mainly discuss about the bottom-up case. First of all, we consider the equivalency of two translations in b-PDTT.

Theorem 1. For any b-PDTT M, there exists a b-PDTT M' such that $M'_\emptyset = M_F$, and conversely.

Proof. The proof is almost the same as the b-PDTA case[25], so we omit the proof. □

Before proving the next theorem, we introduce an abbreviation for a single state b-PDTT. In a single state b-PDTT, we sometimes omit the states since the information of the state is redundant. That is, a translation function is denoted by a mapping from $(\Sigma_n(X_n) \cup \varepsilon(X_1)) \times (\Gamma_{m_1}(Y_{m_1}) \times \cdots \times \Gamma_{m_n}(Y_{m_n}))$ into the finite subsets of $T_{\Gamma\langle\sharp\rangle}(Y_{m_1} \cup \cdots \cup Y_{m_n}) \times T_{\Delta\langle\$'\rangle}(X_n)$. Furthermore we have the following definition in preparation for the proof. Note that this definition is slightly different from the definition in [18].

Definition 11. For the set of states $Q = \{1, 2, \cdots, l\}$, the *encode function* e^i $(i \in Q)$ is a mapping from $T_{\Gamma\langle\sharp\rangle}(Y_m)$ to $T_{\Gamma'\langle\sharp\rangle}(Y_{m\cdot l})$, where $\Gamma' = \{Z^{(i)} \mid Z \in \Gamma,$ $1 \leq i \leq l\}$, which is inductively defined as follows:
(1) $e^i(\sharp) = \sharp$,
(2) for any $Z \in \Gamma_0$, $e^i(Z(\sharp)) = Z^{(i)}(\sharp)$,
(3) for any $y_j \in Y_m$ with $1 \leq j \leq m$, $e^i(y_j) = y_j^{(i)}$, and
(4) for any $Z \in \Gamma_m$ with $m \geq 1$ and $u_k \in T_{\Gamma\langle\sharp\rangle}(Y_m)$ with $1 \leq k \leq m$,

$$e^i(Z(u_1, \cdots, u_m)) = Z^{(i)}(e^1(u_1), \cdots, e^l(u_1), \cdots, e^1(u_m), \cdots, e^l(u_m)).$$

Theorem 2. For any b-PDTT M there exists a single state b-PDTT M' such that $M'_\emptyset = M_\emptyset$.

Proof. For a b-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, \emptyset)$, we define a single state b-PDTT $M' = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N \cup \{\sharp\}, \delta', *, S, \emptyset)$ as follows:

(1) For $Q = \{1, 2, \cdots, l\}$, $N = \{Z^{(i)} \mid Z \in \Gamma,\ 1 \leq i \leq l\}$ $(r_N(Z^{(i)}) = l \cdot r_\Gamma(Z))$, and

(2) $S = Z_0^{(q_0)}$,

(3) For any $\sigma \in \Sigma_n \cup \{\varepsilon\}$ with $n \geq 0$, if $\delta(\sigma(\overrightarrow{x}), ([q_1, Z_1(\overrightarrow{y_1})], \cdots, [q_n, Z_n(\overrightarrow{y_n})]))$ $\ni ([p, u], \tilde{t})$, where $\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$, then

$$\delta'(\sigma(\overrightarrow{x}), (e^{q_1}(Z_1(\overrightarrow{y_1})), \cdots, e^{q_n}(Z_n(\overrightarrow{y_n})))) \ni (e^p(u), \tilde{t}).$$

For M' defined above, the following lemma holds.

Lemma 1. For any $t \in T_\Sigma$, $Z_{0(i)} = Z_0$ and $S_{(i)} = S$ with $1 \leq i \leq r'$ ($Z_{0(i)}$ (or $S_{(i)}$) shows i-th Z_0 (or S)),

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(r')}(\sharp)], \$)]$$
$$\vdash^k_{M'} t_{\Sigma\langle\$\rangle}[([*, e^{q_1}(u_1)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_r}(u_r)], \tilde{t}_{r\Delta\langle\$'\rangle})],$$

if and only if

$t_{\Sigma\langle\$\rangle}[([q_0, Z_{0(1)}(\sharp)], \$), \cdots, ([q_0, Z_{0(r')}(\sharp)], \$)]$
$\vdash_M^k t_{\Sigma\langle\$\rangle}[([q_1, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([q_r, u_r], \tilde{t}_{r\Delta\langle\$'\rangle})]$
where $\tilde{t}_{i\Delta\langle\$'\rangle} \in T_{\Delta\langle\$'\rangle}$, $q_i \in Q$, and $u_i \in T_{\Gamma\langle\sharp\rangle}$ with $1 \leq i \leq r$.

Proof of Lemma. We describe only the outline of the proof since the proof is almost the same as the proof of Lemma 1 in [25].

If part: The proof is an induction on the number k of generation steps.
(1) Basis: We omit the proof since it is trivially clear.
(2) Inductive step: Suppose that the lemma holds for the number k of generation steps. For
$t_{\Sigma\langle\$\rangle}[([q_0, Z_{0(1)}(\sharp)], \$), \cdots, ([q_0, Z_{0(r')}(\sharp)], \$)]$
$\vdash_M^k t_{\Sigma\langle\$\rangle}[([q_1, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([q_r, u_r], \tilde{t}_{r\Delta\langle\$'\rangle})]$,
without loss of generality, we assume that $\sigma(([q_1, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([q_n, u_n], \tilde{t}_{n\Delta\langle\$'\rangle}))$, where $u_i = Z_i(\overrightarrow{u_i})$ with $1 \leq i \leq n$, is the subtree of $t_{\Sigma\langle\$\rangle}[([q_1, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([q_r, u_r], \tilde{t}_{r\Delta\langle\$'\rangle})]$ which is generated by the next step. If the translation function applied next is

(a) $\delta(\sigma(\overrightarrow{x}), ([q_1, Z_1(\overrightarrow{y_1})], \cdots, [q_n, Z_n(\overrightarrow{y_n})])) \ni ([p, v], \tilde{t})$,

where $\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$, then
$t_{\Sigma\langle\$\rangle}[([q_1, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([q_r, u_r], \tilde{t}_{r\Delta\langle\$'\rangle})]$
$\vdash_M t_{\Sigma\langle\$\rangle}[([p, v(u_{ij}/y_{ij})], \tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle\$'\rangle}/x_n)), ([q_{n+1}, u_{n+1}], \tilde{t}_{n+1\Delta\langle\$'\rangle}), \cdots, ([q_r, u_r], \tilde{t}_{r\Delta\langle\$'\rangle})]$,
holds. On the other hand since
$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(r')}(\sharp)], \$)]$
$\vdash_{M'}^k t_{\Sigma\langle\$\rangle}[([*, e^{q_1}(u_1)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_r}(u_r)], \tilde{t}_{r\Delta\langle\$'\rangle})]$
holds by the inductive hypothesis, there exists the translation function
$\delta'(\sigma(\overrightarrow{x}), (e^{q_1}(Z_1(\overrightarrow{y_1})), \cdots, e^{q_n}(Z_n(\overrightarrow{y_n})))) \ni (e^p(v), \tilde{t})$,
where $e^{q_i}(Z_i(\overrightarrow{y_i})) = Z_i^{(q_i)}(y_{i1}^{(1)}, \cdots, y_{i1}^{(l)}, \cdots, y_{ik_i}^{(1)}, \cdots, y_{ik_i}^{(l)})$ for $\overrightarrow{y_i} = (y_{i1}, \cdots, y_{ik_i})$ and $1 \leq i \leq n$, in M'. And if

$$\alpha = t_{\Sigma\langle\$\rangle}[([*, e^{q_1}(u_1)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_n}(u_n)], \tilde{t}_{n\Delta\langle\$'\rangle}), ([*, e^{q_{n+1}}(u_{n+1})], \tilde{t}_{n+1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_r}(u_r)], \tilde{t}_{r\Delta\langle\$'\rangle})]$$
$$= t_{\Sigma\langle\$\rangle}[([*, e^{q_1}(Z_1(\overrightarrow{u_1})), \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_n}(Z_n(\overrightarrow{u_n})), \tilde{t}_{n\Delta\langle\$'\rangle}), ([*, e^{q_{n+1}}(u_{n+1})], \tilde{t}_{n+1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_r}(u_r)], \tilde{t}_{r\Delta\langle\$'\rangle})],$$

where $e^{q_i}(Z_i(\overrightarrow{u_i})) = Z_i^{(q_i)}(e^1(u_{i1}), \cdots, e^l(u_{i1}), \cdots, e^1(u_{ik_i}), \cdots, e^l(u_{ik_i}))$ for $\overrightarrow{u_i} = (u_{i1}, \cdots, u_{ik_i})$ and $1 \leq i \leq n$, then

$$\alpha \vdash_{M'} t_{\Sigma\langle\$\rangle}[([*, e^p(v)(e^h(u_{ij})/y_{ij}^{(h)})], \tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle\$'\rangle}/x_n)), ([*, e^{q_{n+1}}(u_{n+1})], \tilde{t}_{n+1\Delta\langle\$'\rangle}), \cdots, ([*, e^{q_r}(u_r)], \tilde{t}_{r\Delta\langle\$'\rangle})].$$

Therefore, if
$t_{\Sigma\langle\$\rangle}[([q_0, Z_{0(1)}(\sharp)], \$), \cdots, ([q_0, Z_{0(r')}(\sharp)], \$)]$
$\vdash_M^{k+1} t_{\Sigma\langle\$\rangle}[([p, v(u_{ij}/y_{ij})], \tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle\$'\rangle}/x_n)), ([q_{n+1}, u_{n+1}],$

$\tilde{t}_{n+1\Delta\langle \$' \rangle}), \cdots, ([q_r, u_r], \tilde{t}_{r\Delta\langle \$' \rangle})],$

then

$$t_{\Sigma\langle \$ \rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(r')}(\sharp)], \$)]$$
$$\vdash^{k+1}_{M'} t_{\Sigma\langle \$ \rangle}[((([*, e^p(v(u_{ij}/y_{ij}))], \tilde{t}(\tilde{t}_{1\Delta\langle \$' \rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle \$' \rangle}/x_n)), ([*, e^{q_{n+1}}$$
$$(u_{n+1})], \tilde{t}_{n+1\Delta\langle \$' \rangle}), \cdots, ([*, e^{q_r}(u_r)], \tilde{t}_{r\Delta\langle \$' \rangle})]$$

and the lemma holds for the number $k+1$ of generation steps. If the translation function applied next is

(b) $\delta(\sigma(\overrightarrow{x}), [q_1, Z_1(\overrightarrow{y_1})]) \ni ([p, v], \tilde{t}),$

then the lemma holds by the same way as (a) case.

Only if part: We can prove the only if part in almost the same way as the if part.

(End of the proof of Lemma 1)

For any $\langle t, t' \rangle \in T_\Sigma \times T_\Delta$, the following statement holds.

$$t_{\Sigma\langle \$ \rangle}[([q_0, Z_{0(1)}(\sharp)], \$), \cdots, ([q_0, Z_{0(r')}(\sharp)], \$)]$$
$$\vdash^k_M ([p, \sharp], t'_{\Delta\langle \$' \rangle})$$

if and only if

$$t_{\Sigma\langle \$ \rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(r')}(\sharp)], \$)]$$
$$\vdash^k_{M'} ([*, e^p(\sharp)], t'_{\Delta\langle \$' \rangle})$$
$$= ([*, \sharp], t'_{\Delta\langle \$' \rangle}).$$

Thus the theorem holds. □

Definition 12. For a t-PDTT or b-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F$ (or $\emptyset$)), if $\Gamma_n = \emptyset$ with $n \geq 2$ (i.e., Γ is monadic) then M is said to be a linear stack t-PDTT (for shortly lst-PDTT) or a linear stack b-PDTT (for shortly lsb-PDTT).

For a ranked alphabet Σ, the element of T_Σ has the form of $\sigma_1(\sigma_2(\cdots(\sigma_m)\cdots))$ when Σ is monadic. Clearly the notation is similar to the string of the alphabet Σ_1, i.e., $\sigma_1\sigma_2\cdots\sigma_m \in \Sigma_1^*$. So we denote the set of trees $\sigma_1(\sigma_2(\cdots(\sigma_m)\cdots))$ as $(\Sigma)^*$ to distinguish from the string. We sometimes abbreviate $(\Sigma)^*$ to Σ^* if no confusion occurs. For a t-PDTT, the following theorem is proved in [28].

Theorem 3. For any t-PDTT M there exists a lst-PDTT M' such that $M'_\emptyset = M_\emptyset$.

For a b-PDTT, we have also a similar result. Before proving it, we have one normal form of a single state b-PDTT. In the case of context-free tree grammar, this normal form corresponds to the symmetric form[23] of the context-free tree grammar.

Theorem 4. For any single state b-PDTT $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N \cup \{\sharp\}, \delta, *, S, \emptyset)$, there exists a single state b-PDTT $M' = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N' \cup \{\sharp\}, \delta', *, S, \emptyset)$, with $M'_\emptyset = M_\emptyset$, whose translation function is one of the following forms, i.e., for any $\sigma \in \Sigma_n \cup \{\varepsilon\}$ $(n \geq 0)$ and any $Z_i \in N'(1 \leq i \leq n)$,

(1) $\delta'(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n})))$
$\ni (A(B_1(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n}), \cdots, B_m(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n})), \tilde{t})$
$(\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n), A \in N'_m$ with $m \geq 1$ and $B_j \in N'$ with $1 \leq j \leq m)$,

(2) $\delta'(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (y_{ij}, \tilde{t})$
$(\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$ and $y_{ij} \in \{\overrightarrow{y_i}\}$ with $1 \leq i \leq n)$, and

(3) $\delta'(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (Z(\sharp), \tilde{t})$
$(\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$ and $Z \in N'_0)$.

Proof. For the single state b-PDTT M, we convert δ into δ' in M' as follows. That is, for any $\sigma \in \Sigma \cup \{\varepsilon\}$ and any $Z_1, \cdots, Z_n \in N$, if
$\delta(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (u, \tilde{t})$,
then N' contains Z_i with $1 \leq i \leq n$ and the translation function δ' in M' is obtained as follows.

(1) If $u = y_{ij}$ $(y_{ij} \in \{\overrightarrow{y_1}, \cdots, \overrightarrow{y_n}\})$ or $Z(\sharp)$ $(Z \in N_0)$, then the translation function
$$\delta'(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (u, \tilde{t})$$
is contained in M', and Z is added to N'_0 if $u = Z(\sharp)$.

(2) If $u = A(u_1, \cdots, u_m)$ $(A \in N_m$ with $m \geq 1)$, then we have the translation function
$$\delta'(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n})))$$
$$\ni (A(B_1(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n}), \cdots, B_m(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n})), \tilde{t})$$
in M', and $B_1, \cdots, B_m$ are added to N' and the ranks of $B_1, \cdots, B_m$ are the cardinal numbers of $\{\overrightarrow{y_1}, \cdots, \overrightarrow{y_n}\}$. And for $1 \leq k \leq m$,

(a) If $u_k = y_{ij}$ $(y_{ij} \in \{\overrightarrow{y_1}, \cdots, \overrightarrow{y_n}\})$ or $Z(\sharp)$ $(Z \in N_0)$, then the translation function
$$\delta'(\varepsilon(x), B_k(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n})) \ni (u_k, x)$$
is contained in M' and Z is added to N'_0 if $u_k = Z(\sharp)$.

(b) If $u_k = B(v_1, \cdots, v_{m'})$ ($B \in N_{m'}$ with $m' \geq 1$), we repeat this procedure for $\delta(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (u, \tilde{t})$ with any $\sigma \in \Sigma \cup \{\varepsilon\}$ and $Z_1, \cdots, Z_n \in N$, and $u = u_k$ (i.e., $A(u_1, \cdots, u_m) = B(v_1, \cdots, v_{m'})$).

For the translation function δ' obtained by the above procedure, δ' is one of the forms of (1), (2), or (3). $M'_\emptyset = M_\emptyset$ can be proved easily. This completes the proof. □

We call a single state b-PDTT M *symmetric stack form* in b-PDTT if the translation function is one of the forms of (1), (2), or (3) in Theorem 4.

Definition 13. Given a ranked alphabet Σ, let Σ' be the ranked alphabet, where $\Sigma' = \Sigma$, $\sigma \in \Sigma'_0$ if $\sigma \in \Sigma_0$ and $\sigma \in \Sigma'_1$ otherwise. For $t \in T_\Sigma$, the set $P(t)$, the set of paths through t, is the subset of $T_{\Sigma'}$ defined inductively by

(1) $P(\sigma) = \{\sigma\}$ for any $\sigma \in \Sigma_0$, and

(2) $P(\sigma(t_1, \cdots, t_n)) = \bigcup_{i=1}^{n} \{\sigma(w) \mid w \in P(t_i)\}$, where $\sigma \in \Sigma_n$ and $t_1, \cdots, t_n \in T_\Sigma$.

If $T \subseteq T_\Sigma$ then $P(T) = \bigcup_{t \in T} P(t)$ and P is called a *path function*. Furthermore an element of $P(T)$ is called a *path*.

Theorem 5. For any single state b-PDTT $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N \cup \{\sharp\}, \delta, *, S, \emptyset)$, there exists a single state lsb-PDTT M' such that $M'_\emptyset = M_\emptyset$.

Proof. We can assume that M is in the symmetric stack form by Theorem 4. For such M, we construct a single state lsb-PDTT $M' = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N' \cup \{\sharp\}, \delta', *, S, \emptyset)$ as follows:

(1) $N' = N'_1 \cup N'_0$, where $N'_1 = \{Z^{(j)} \mid$ for any $Z \in N_m (m \geq 1), 1 \leq j \leq m\}$, and $N'_0 = \{Z \mid$ for any $Z \in N_0\}$.

(2) δ' is defined as follows:

(a) If $\delta(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (A(B_1(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n}), \cdots, B_m(\overrightarrow{y_1}, \cdots, \overrightarrow{y_n})), \tilde{t})$, where $\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$, then

$$\delta'(\sigma(\overrightarrow{x}), (Z_1^{(j_1)}(z_1), \cdots, Z_n^{(j_n)}(z_n))) \ni (A^{(p_1)}(B_{p_1}^{(p_2)}(z_{p_3})), \tilde{t}),$$

where $1 \leq p_1 \leq m$ and $1 \leq p_2 \leq k_1 + \cdots + k_n$ with $k_h = r_N(Z_h)$ $(1 \leq h \leq n)$, however, for fixed p_2 if $p_2 = k_1 + \cdots + k_{r-1} + s$ $(1 \leq r \leq n, 1 \leq s \leq k_r)$ then we define $p_3 = r$ and $j_r = s$, moreover the indexes j_h $(1 \leq h \leq n$ and $h \neq r)$, but not j_r, take the value from 1 to k_h.

(b) If $\delta(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (y_{rs}, \tilde{t})$, where $\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$, $1 \leq r \leq n$, and $1 \leq s \leq k_r$, then

$$\delta'(\sigma(\overrightarrow{x}), (Z_1^{(j_1)}(z_1), \cdots, Z_n^{(j_n)}(z_n))) \ni (z_r, \tilde{t}),$$

where $j_r = s$ and $1 \leq j_h \leq k_h$ with $k_h = r_N(Z_h)$ $(1 \leq h \leq n, h \neq r)$.

(c) If $\delta(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n}))) \ni (Z(\sharp), \tilde{t})$, where $Z \in N_0$ and $\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$, then

$$\delta'(\sigma(\overrightarrow{x}), (Z_1^{(j_1)}(z_1), \cdots, Z_n^{(j_n)}(z_n))) \ni (Z(\sharp), \tilde{t}),$$

where $1 \leq j_h \leq k_h$ with $k_h = r_N(Z_h)$ $(1 \leq h \leq n)$.

For the single state lsb-PDTT M' defined above, we have the following two lemmas.

Lemma 2. Let P be a path function from $T_{N\langle\sharp\rangle}$ to the subset of $T_{N'\langle\sharp\rangle}$. For any $\langle t, t'\rangle \in T_\Sigma \times T_\Delta$, if

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_M^k ([*, u], t'_{\Delta\langle\$'\rangle}),$$

then for any $v \in P(u)$,

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_{M'}^k ([*, v], t'_{\Delta\langle\$'\rangle})$$

holds.

Proof of Lemma. The proof is an induction on the number k of generation steps.

(1) Basis: We omit the proof since it is trivially clear.

(2) Inductive step: Suppose that the lemma holds for the number k of generation steps. If

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l)}(\sharp)], \$)]$$
$$\vdash_M^k t_{\Sigma\langle\$\rangle}[([*, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_n], \tilde{t}_{n\Delta\langle\$'\rangle})],$$

where $\tilde{t}_{i\Delta\langle\$'\rangle} \in T_{\Delta\langle\$'\rangle}$ with $1 \leq i \leq n$, then there exists the generation

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l)}(\sharp)], \$)]$$
$$\vdash_{M'}^k t_{\Sigma\langle\$\rangle}[([*, v_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, v_n], \tilde{t}_{n\Delta\langle\$'\rangle})],$$

where $v_i \in P(u_i)$ with $1 \leq i \leq n$ by the inductive hypothesis. Now let $u_i = Z_i(\overrightarrow{u_i}) = Z_i(u_{i1}, \cdots, u_{ik_i})$ with $1 \leq i \leq n$ and we denote $v_i = Z_i^{(j_i)}(v_{ij_i})$ $(v_{ij_i} \in P(u_{ij_i})$ with $1 \leq j_i \leq k_i)$ since $v_i \in P(u_i) = P(Z_i(u_{i1}, \cdots, u_{ik_i}))$. Moreover let

$$\alpha_M = t_{\Sigma\langle\$\rangle}[([*, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_n], \tilde{t}_{n\Delta\langle\$'\rangle})]$$
$$= t_{\Sigma\langle\$\rangle}[([*, Z_1(\overrightarrow{u_1})], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(\overrightarrow{u_n})], \tilde{t}_{n\Delta\langle\$'\rangle})],$$

and

$$\alpha_{M'} = t_{\Sigma\langle\$\rangle}[([*, v_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, v_n], \tilde{t}_{n\Delta\langle\$'\rangle})]$$
$$= t_{\Sigma\langle\$\rangle}[([*, Z_1^{(j_1)}(v_{1j_1})], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n^{(j_n)}(v_{nj_n})], \tilde{t}_{n\Delta\langle\$'\rangle})].$$

Under above descriptions and $t_{\Sigma\langle\$\rangle}[([*, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_n], \tilde{t}_{n\Delta\langle\$'\rangle})] = \sigma$ $(([*, u_1], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_n], \tilde{t}_{n\Delta\langle\$'\rangle}))$,

(a) If the translation function applied next is $\delta(\sigma(\overrightarrow{x}),(Z_1(\overrightarrow{y_1}),\cdots,Z_n(\overrightarrow{y_n}))) \ni (A(B_1(\overrightarrow{y_1},\cdots,\overrightarrow{y_n}),\cdots,B_m(\overrightarrow{y_1},\cdots,\overrightarrow{y_n})),\tilde{t})$, where $\tilde{t}\in T_{\Delta\langle\$'\rangle}(X_n)$ and $A, B_i \in N$ with $1\le i\le m$, then there exists the generation $\alpha_M \vdash_M ([*, A(B_1(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}),\cdots,B_m(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}))],\tilde{t})$. On the other hand, for any p_1 $(1\le p_1\le m)$ and p_2 $(1\le p_2\le k_1+\cdots+k_n)$, there exists the translation function

$$\delta'(\sigma(\overrightarrow{x}),(Z_1^{(j_1)}(z_1),\cdots,Z_n^{(j_n)}(z_n))) \ni (A^{(p_1)}(B_{p_1}^{(p_2)}(z_{p_3})),\tilde{t}),$$

where if $p_2=k_1+\cdots+k_{r-1}+s$ $(1\le r\le n$ and $1\le s\le k_r)$ then $p_3=r$, $j_r=s$ and $1\le j_h\le k_h$ $(1\le h\le n$ and $h\ne r)$ in M'. Therefore, for $1\le r\le n$ and $1\le s\le k_r$, there exists the generation

$$\alpha_{M'}(r,s)=t_{\Sigma\langle\$\rangle}[([*,Z_1^{(j_1)}(v_{1j_1})],\ \tilde{t}_{1\Delta\langle\$'\rangle}),\cdots,([*,Z_r^{(s)}(v_{rs})],\ \tilde{t}_{r\Delta\langle\$'\rangle}),\cdots,([*,Z_n^{(j_n)}(v_{nj_n})],\ \tilde{t}_{n\Delta\langle\$'\rangle})]$$
$$\vdash_{M'}([*,A^{(p_1)}(B_{p_1}^{(p_2)}(v_{rs}))],\tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1,\cdots,\tilde{t}_{n\Delta\langle\$'\rangle}/x_n)),$$

where $\alpha_{M'}(r,s)$ is the notation to distinguish from $\alpha_{M'}$, i.e., $\alpha_{M'}(r,s)$ to fix the indexes i and j^i of $Z_i^{(j_i)}$ in $\alpha_{M'}$ to r and s, respectively. That is, if

$$t_{\Sigma\langle\$\rangle}[([*,S_{(1)}(\sharp)],\$),\cdots,([*,S_{(l)}(\sharp)],\$)]$$
$$\vdash_M^{k+1}([*,A(B_1(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}),\cdots,B_m(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}))],\tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1,\cdots,\tilde{t}_{n\Delta\langle\$'\rangle}/x_n))$$
$$=([*,A(B_1(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}),\cdots,B_m(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}))],t'_{\Delta\langle\$'\rangle}),$$

then there exists the generation

$$t_{\Sigma\langle\$\rangle}[([*,S_{(1)}(\sharp)],\$),\cdots,([*,S_{(l)}(\sharp)],\$)]$$
$$\vdash_{M'}^{k+1}([*,A^{(p_1)}(B_{p_1}^{(p_2)}(v_{rs}))],\tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1,\cdots,\tilde{t}_{n\Delta\langle\$'\rangle}/x_n))$$
$$=([*,A^{(p_1)}(B_{p_1}^{(p_2)}(v_{rs}))],t'_{\Delta\langle\$'\rangle}),$$

where $1\le p_1\le m$, $1\le r\le n$, $1\le s\le k_r$,$v_{rs}\in P(u_{rs})$, and $1\le p_2\le k_1+\cdots+k_n$. Therefore, clearly

$$\{A^{(p_1)}(B_{p_1}^{(p_2)}(v_{rs}))\mid\ 1\le p_1\le m,1\le r\le n,1\le s\le k_r,\text{ and any } v_{rs}\in P(u_{rs})\}$$
$$=P(A(B_1(\overrightarrow{u_1},\cdots,\overrightarrow{u_n}),\cdots,\ B_m(\overrightarrow{u_1},\cdots,\overrightarrow{u_n})))$$

and the lemma holds for the number $k+1$ of generation steps.

(b) If the translation function applied next is $\delta(\sigma(\overrightarrow{x}),(Z_1(\overrightarrow{y_1}),\cdots,Z_n(\overrightarrow{y_n}))) \ni (y_{rs},\tilde{t})$, where $\tilde{t}\in T_{\Delta\langle\$'\rangle}(X_n)$, $1\le r\le n$, and $1\le s\le k_r$, then there exists the generation $\alpha_M\vdash_M([*,u_{rs}],\tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1,\cdots,\tilde{t}_{n\Delta\langle\$'\rangle}/x_n))$. On the other hand there exists the translation function

$\delta'(\sigma(\overrightarrow{x}), (Z_1^{(j_1)}(z_1), \cdots, Z_n^{(j_n)}(z_n))) \ni (z_r, \tilde{t}),$
where $j_r = s$, $1 \leq j_h \leq k_h$ with $1 \leq h \leq n$, and $h \neq r$ in M'. Therefore, we obtain

$$\alpha_{M'}(r,s) = t_{\Sigma\langle\$\rangle}[([*, Z_1^{(j_1)}(v_{1j_1})], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_r^{(s)}(v_{rs})], \tilde{t}_{r\Delta\langle\$'\rangle}), \cdots, ([*, Z_n^{(j_n)}(v_{nj_n})], \tilde{t}_{n\Delta\langle\$'\rangle})] \vdash_{M'} ([*, v_{rs}], \tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle\$'\rangle}/x_n))$$

and the lemma holds for the number $k+1$ of generation steps since any $v_{rs} \in P(u_{rs})$.

(c) If the translation function applied next is $\delta(\sigma(\overrightarrow{x}), (Z_1(\overrightarrow{y_1}), \cdots, Z_n(\overrightarrow{y_n})))$ $\ni (Z(\sharp), \tilde{t})$, where $\tilde{t} \in T_{\Delta\langle\$'\rangle}(X_n)$, and $Z \in N_0$, then there exists the generation $\alpha_M \vdash_M ([*, Z(\sharp)], \tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle\$'\rangle}/x_n))$. On the other hand there exists the translation function
$\delta'(\sigma(\overrightarrow{x}), (Z_1^{(j_1)}(z_1), \cdots, Z_n^{(j_n)}(z_n))) \ni (Z(\sharp), \tilde{t}),$
where $1 \leq j_h \leq k_h$ with $1 \leq h \leq n$ in M'. Therefore, we obtain

$$\alpha_{M'} = t_{\Sigma\langle\$\rangle}[([*, Z_1^{(j_1)}(v_{1j_1})], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n^{(j_n)}(v_{nj_n})], \tilde{t}_{n\Delta\langle\$'\rangle})] \vdash_{M'} ([*, Z(\sharp)], \tilde{t}(\tilde{t}_{1\Delta\langle\$'\rangle}/x_1, \cdots, \tilde{t}_{n\Delta\langle\$'\rangle}/x_n))$$

and the lemma holds for the number $k+1$ of generation steps since $P(Z(\sharp)) = \{Z(\sharp)\}$. (End of the proof of Lemma 2)

Lemma 3. Let P be a path function from $T_{N\langle\sharp\rangle}$ to the subset of $T_{N'\langle\sharp\rangle}$. For any $\langle t, t'\rangle \in T_\Sigma \times T_\Delta$, if

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_{M'}^k ([*, v], t'_{\Delta\langle\$'\rangle}),$$

then there exists u such that $v \in P(u)$ and

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_M^k ([*, u], t'_{\Delta\langle\$'\rangle})$$

holds.

Proof of Lemma. This lemma can be proved almost the same as the previous Lemma 2. (End of the proof of Lemma 3)

Under Lemma 2 and Lemma 3, for any $\langle t, t'\rangle \in T_\Sigma \times T_\Sigma$ and $u \in T_{N\langle\sharp\rangle}$,

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_M^k ([*, u], t'_{\Delta\langle\$'\rangle})$$

if and only if

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_{M'}^k ([*, v], t'_{\Delta\langle\$'\rangle}),$$

where any $v \in P(u)$.

Thus $M_\emptyset = M'_\emptyset$ holds since $P(\sharp) = \{\sharp\}$ and this completes the proof. □

We have the following corollary immediately from Theorem 2 and Theorem 5.

Corollary. The class of translations generated by b-PDTT is equal to the class of translations generated by a single state lsb-PDTT.

Example 3. Based on the b-PDTT M in Example 2, we have a single state lsb-PDTT M^3 which is equivalent to M as follows.

First of all, we obtain a single state b-PDTT $M^1 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N \cup \{\sharp\}, \delta, *, Z^{(q)}, \emptyset)$, where $N = \{X^{(p)}, X^{(q)}, Y^{(p)}, Y^{(q)}, Z^{(p)}, Z^{(q)}\}$ ($N_0 = N, N_4 = \{X^{(p)}, Y^{(p)}\}$) which is equivalent to M according to the proof of Theorem 2. Then the translation functions are

(1-1) $\delta(a(\$), Z^{(q)}(\sharp)) = \{(X^{(p)}(Y^{(p)}(\sharp), Y^{(q)}(\sharp), X^{(p)}(\sharp), X^{(q)}(\sharp)), \sigma(\$'))\}$,

(1-2) $\delta(b(\$), Z^{(q)}(\sharp)) = \{(Y^{(p)}(\sharp), \tau(\$'))\}$,

(1-3) $\delta(b(\overrightarrow{x}), (X^{(p)}(y_1^{(p)}, y_1^{(q)}, y_2^{(p)}, y_2^{(q)}), Y^{(p)}(\sharp))$
$= \{(Y^{(p)}(y_2^{(p)}, y_2^{(q)}, y_2^{(p)}, y_2^{(q)}), \sigma(x_1, \gamma(\$')))\}$, and

(1-4) $\delta(a(\overrightarrow{x}), (Y^{(p)}(\sharp), Y^{(p)}(y_1^{(p)}, y_1^{(q)}, y_2^{(p)}, y_2^{(q)}))) = \{(\sharp, \sigma(x_2, \tau(x_1, \tau(\$'))))\}$.

Next, we obtain a single state b-PDTT with a symmetric stack form $M^2 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N^2 \cup \{\sharp\}, \delta_2, *, S, \emptyset)$, where $N^2 = \{X^{(p)}, X^{(q)}, Y^{(p)}, Y^{(q)}, W^1, W^2, W^3, W^4, Z^{(q)}\}$ ($N_0^2 = \{X^{(p)}, X^{(q)}, Y^{(p)}, Y^{(q)}, Z^{(q)}\}, N_4^2 = \{X^{(p)}, Y^{(p)}, W^1, W^2, W^3, W^4\}$) which is equivalent to M^1 according to the proof of Theorem 4. Therefore the translation functions are as follows:

(2-1) $\delta_2(a(\$), Z^{(q)}(\sharp)) = \{(X^{(p)}(Y^{(p)}(\sharp), Y^{(q)}(\sharp), X^{(p)}(\sharp), X^{(q)}(\sharp)), \sigma(\$'))\}$,

(2-2) $\delta_2(b(\$), Z^{(q)}(\sharp)) = \{(Y^{(p)}(\sharp), \tau(\$'))\}$,

(2-3)$_1$ $\delta_2(b(\overrightarrow{x}), (X^{(p)}(\overrightarrow{y}), Y^{(p)}(\sharp))$
$= \{(Y^{(p)}(W^1(\overrightarrow{y}), W^2(\overrightarrow{y}), W^3(\overrightarrow{y}), W^4(\overrightarrow{y})), \sigma(x_1, \gamma(\$')))\}$,

(2-3)$_2$ $\delta_2(\varepsilon(x), W^1(\overrightarrow{y})) = \{(y_3, x)\}$,

(2-3)$_3$ $\delta_2(\varepsilon(x), W^2(\overrightarrow{y})) = \{(y_4, x)\}$,

(2-3)$_4$ $\delta_2(\varepsilon(x), W^3(\overrightarrow{y})) = \{(y_3, x)\}$,

(2-3)$_5$ $\delta_2(\varepsilon(x), W^4(\overrightarrow{y})) = \{(y_4, x)\}$, and

(2-4) $\delta_2(a(\overrightarrow{x}), (Y^{(p)}(\sharp), Y^{(p)}(\overrightarrow{y}))) = \{(\sharp, \sigma(x_2, \tau(x_1, \tau(\$'))))\}$.

Moreover, We obtain a single state lsb-PDTT $M^3 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, N^3 \cup \{\sharp\}, \delta_3, *, S, \emptyset)$, $N^3 = N_1^3 \cup N_0^3$ ($N_1^3 = \{Z^{(j)} \mid$ for any $Z \in N_4^2, 1 \leq j \leq 4\}$, $N_0^3 = \{Z \mid$ for any $Z \in N_0^2\}$) which is equivalent to M^2 according to the proof of Theorem 5. Therefore the translation functions are as follows:

(3-1) $\delta_3(a(\$), Z^{(q)}(\sharp)) = \{(X^{(p)(1)}(Y^{(p)}(\sharp)), \sigma(\$')), (X^{(p)(2)}(Y^{(q)}(\sharp)), \sigma(\$')),$
$(X^{(p)(3)}(X^{(p)}(\sharp)), \sigma(\$')), (X^{(p)(4)}(X^{(q)}(\sharp)), \sigma(\$'))\}$,

(3-2) $\delta_3(b(\$), Z^{(q)}(\sharp)) = \{(Y^{(p)}(\sharp), \tau(\$'))\}$,

(3-3)$_1$ for $1 \leq i \leq 4$, $\delta_3(b(\overrightarrow{x}), (X^{(p)(i)}(y_i), Y^{(p)}(\sharp)) = \{(Y^{(p)(j)}(W^{j(i)}(y_i)), \sigma$
$(x_1, \gamma(\$'))) \mid 1 \leq j \leq 4\}$,

(3-3)$_2$ $\delta_3(\varepsilon(x), W^{1(3)}(\overrightarrow{y})) = \{(y_3, x)\}$,

(3-3)$_3$ $\delta_3(\varepsilon(x), W^{2(4)}(\overrightarrow{y})) = \{(y_4, x)\}$,

(3-3)$_4$ $\delta_3(\varepsilon(x), W^{3(1)}(\overrightarrow{y})) = \{(y_1, x)\}$,

$(3\text{-}3)_5$ $\delta_3(\varepsilon(x), W^{4(2)}(\overrightarrow{y})) = \{(y_2, x)\}$, and

(3-4) for $1 \leq i \leq 4$, $\delta_3(a(\overrightarrow{x}), (Y^{(p)}(\sharp), Y^{(p)(i)}(y_i))) = \{(\sharp, \sigma(x_2, \tau(x_1, \tau(\$'))))\}$.
This M^3 also translates $t = a(b, b(a, b))$ into $t' = \sigma(\sigma(\sigma, \gamma), \tau(\tau, \tau))$. Therefore, $\langle t, t' \rangle \in M^3_\emptyset$ and $M_\emptyset = M^3_\emptyset$ holds.

§5. Extended lst-PDTT and G-type Normal Form b-PDTT

In this section we introduce an extended lst-PDTT for the sake of comparing the translational capability of t-PDTT with that of b-PDTT. Moreover, we introduce a G-type normal form in b-PDTT which corresponds to the Greibach normal form in context-free grammar, to simplify the following argument.

Definition 14. A lst-PDTT $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F)$ is called an extended lst-PDTT when the translation function is defined as a mapping from (i) for $n \neq 0$, a mapping from $Q \times (\Sigma_n(X_n) \cup \varepsilon(X_1)) \times (\Gamma)^m(Y)$ with $m \geq 0$ into the finite subsets of $\{t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], x'_1), \cdots, ([q_l, u_l], x'_l)\rangle \mid$ for any $l \geq 1$, $q_i \in Q$ and $u_i \in (\Gamma)^{l_i}(Y)$ $(l_i \geq 0)$ with $1 \leq i \leq l$, and $\{x'_1, \cdots, x'_l\} \subseteq X_n\}$, and (ii) for $n = 0$, a mapping from $Q \times (\Sigma_n(\$) \cup \varepsilon(\$)) \times (\Gamma)^m(Y)$ with $m \geq 0$ into the finite subsets of $\{t_{\Delta\langle\$'\rangle}\langle([q_1, u_1], \$''_1), \cdots, ([q_l, u_l], \$''_l)\rangle \mid$ for any $l \geq 1$, $q_i \in Q$ and $u_i \in (\Gamma)^{l_i}(Y)$ $(l_i \geq 0)$ with $1 \leq i \leq l$, and $\$''_i \in \{\$, \$'\}\}$, where for any $k \geq 0$, $(\Gamma)^k(Y) = \{Z_1(Z_2(\cdots(Z_k(Y))\cdots)) \mid$ for any $Z_i \in \Gamma$ with $1 \leq i \leq n$ and $y' = y$ or $\sharp\}$.

As is easily seen by this definition, we extend the domain of the translation function in lst-PDTT from $Q \times (\Sigma_n(X_n) \cup \varepsilon(X_1)) \times \Gamma(Y)$ (or $Q \times (\Sigma_n(\$) \cup \varepsilon(\$)) \times \Gamma(Y)$) to $Q \times (\Sigma_n(X_n) \cup \varepsilon(X_1)) \times (\Gamma)^m(Y)$ (or $Q \times (\Sigma_n(\$) \cup \varepsilon(\$)) \times (\Gamma)^m(Y)$) with $m \geq 0$. Note that $\delta(q, \sigma(x_1, \cdots, x_n), \sharp)$ $(q \in Q$ and $\sigma \in \Sigma_n)$ is defined in extended lst-PDTT since $\sharp \in (\Gamma)^m(Y)$ with $m \geq 0$. And we also note that the definitions of extended lst-PDTT, such as generation, translation and so on, are the same as those of t-PDTT.

Theorem 6. For any extended lst-PDTT M, there exists an extended lst-PDTT M' such that $M'_\emptyset = M_F$, and conversely.

Proof. The proof is similar to the proof of Lemma 2.22 and 2.23 in [2], so we omit the proof. □

Theorem 7. For any extended lst-PDTT M, there exists a lst-PDTT M' such that $M'_F = M_F$.

Proof. The proof is similar to the proof of Lemma 2.21 in [2], so we omit the proof. □

Next, we consider a C-type normal form and a G-type normal form in b-PDTT which correspond to the Chomsky normal form and the Greibach normal form in context-free grammar (CFG). That is, the symmetric stack form in b-PDTT can be made more simple since the structure of the stack is linear.

Definition 15. Let M be a single state b-PDTT such that $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$. If the translation function is one of the following forms:

(a) for $Z \in \Gamma$,

$$\delta_0(\varepsilon(x), (Z(y)) \ni (A(B(y)), \tilde{t}),$$

where $A \in \Gamma_1, Z, B \in \Gamma$, and

(b) for $\sigma \in \Sigma_n$ $(n \geq 0)$ and $Z_i \in \Gamma$ $(1 \leq i \leq m)$,

$$\delta_0(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (y_r, \tilde{t}),$$

where $1 \leq r \leq n$,

then M is called *C-type normal form* in b-PDTT. And if the translation functions are only of the following form, i.e., for $\sigma \in \Sigma_n$ $(n \geq 0)$ and $Z_i \in \Gamma$ $(1 \leq i \leq n)$,

$$\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (X_1(X_2(\cdots(X_m(y_r)\cdots)), \tilde{t}),$$

where $X_i \in \Gamma$ $(1 \leq i \leq m)$, $m \geq 0$, and $1 \leq r \leq n$, then M is called *G-type normal form* in b-PDTT.

Before proving the next theorem, we have some definitions. For any $\sigma \in \Sigma_n \cup \{\varepsilon\}$ $(n \geq 0)$ and $Z_i \in \Gamma$ $(1 \leq i \leq n)$, we call the translation function $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (\alpha(y_r), \tilde{t})$ with $\alpha \in (\Gamma)^*$ and $1 \leq r \leq n$, *Z_r-translation function.* And for $A \in \Gamma$, we call the translation function $\delta(\varepsilon(x), A(y)) \ni (A(\alpha(y)), x)$ with $\alpha \in (\Gamma)^*$, *left recursive.*

Theorem 8. For any b-PDTT M, there exists a G-type normal form b-PDTT M' such that $M'_\emptyset = M_\emptyset$, and conversely.

Proof. The proof is simple, but somewhat long, so we describe only the outline of the proof. The detailed proof is shown in Lemma 1,2,3 and Theorem 2 of [29]. From Theorem 4, we can suppose that M is a symmetric stack form b-PDTT. For $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$, we can prove that there exists a C-type normal form b-PDTT $M^0 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma^o \cup \{\sharp\}, \delta_o, *, S, \emptyset)$ with $M^0_\emptyset = M_\emptyset$, and the following two lemmas hold without proofs.

Lemma 4. For a single state lsb-PDTT $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$, suppose that a translation function

$\delta(\varepsilon(x), A(y)) \ni (B(\alpha(y)), x)$,

where $A, B \in \Gamma$ and $\alpha \in (\Gamma)^*$, exists in M and let R_B be the set of B-translation functions in M. And for any translation function

$\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, B(y_r), \cdots, Z_n(y_n))) \ni (\beta(y_r), \tilde{t})$,

where $\sigma \in \Sigma \cup \{\varepsilon\}$ and $\beta \in (\Gamma)^*$, in R_B, let R'_B be the set of translation functions

$\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, A(y_r), \cdots, Z_n(y_n))) \ni (\beta(\alpha(y_r)), \tilde{t})$.

Now we convert M into a single state lsb-PDTT $M' = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta', *, S, \emptyset)$ by removing the translation function $\delta(\varepsilon(x), A(y)) \ni (B(\alpha(y)), x)$ and adding the set of the translation functions R'_B. Then $M'_\emptyset = M_\emptyset$ holds.

Lemma 5. For a single state lsb-PDTT $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$, let

$\delta(\varepsilon(x), A(y)) \ni (A(\alpha_i(y)), x)$,

where $1 \leq i \leq r$, be the set of left recursive translation functions in A-translation functions of M and let

$\delta(\sigma_i(\overrightarrow{x}), (Z_{i1}(y_1), \cdots, A(y_{r_i}), \cdots, Z_{in_i}(y_{n_i}))) \ni (\beta_i(y_{r_i}), \tilde{t})$,

where $\sigma_i \in \Sigma \cup \{\varepsilon\}$ and $\beta_i \in (\Gamma)^*$ with $1 \leq i \leq s$, be the set of non-left recursive A-translation functions in M. Now, we convert M into a single state lsb-PDTT $M' = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{W\} \cup \{\sharp\}, \delta', *, S, \emptyset)$ by replacing all A-translation function by one of the following translation functions:

(a) $\delta'(\sigma_i(\overrightarrow{x}), (Z_{i1}(y_1), \cdots, A(y_{r_i}), \cdots, Z_{in_i}(y_{n_i}))) \ni (\beta_i(y_{r_i}), \tilde{t})$ or $(\beta_i(W(y_{r_i})), \tilde{t})$ with $1 \leq i \leq s$, and

(b) $\delta'(\varepsilon(x), W(y)) \ni (\alpha_i(y), x)$ or $(\alpha_i(W(y)), x)$ with $1 \leq i \leq r$.

Then $M'_\emptyset = M_\emptyset$ holds.

We have the G-type normal form in b-PDTT by the following procedures. First, Let a b-PDTT $M^0 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma^0 \cup \{\sharp\}, \delta_0, *, S, \emptyset)$ with $\Gamma^0 = \{A_1, A_2, \cdots, A_m\}$ be a C-type normal form in b-PDTT satisfying $M^0_\emptyset = M_\emptyset$. Second, we convert M^0 into a single state lsb-PDTT $M^1 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma^1 \cup \{\sharp\}, \delta_1, *, S, \emptyset)$ by replacing the set of left recursive-translation functions in A_1-translation functions of M^0 according to Lemma 5. Then $M^1_\emptyset = M^0_\emptyset$ holds. Third, we convert M^1 into a single state lsb-PDTT $M^2 = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma^2 \cup \{\sharp\}, \delta_2, *, S, \emptyset)$ by the following procedures: (1) the translation functions, whose forms are $\delta_1(\varepsilon(x), A_2(y)) \ni (A_1(B(y)), \tilde{t})$, in A_2-translation functions of M^1 are replaced according to Lemma 4, and (2) the set of left recursive-translation functions in A_2-translation functions of M^1

is replaced according to Lemma 5. Then $M_\emptyset^2 = M_\emptyset^1$ holds. After that we define single state lsb-PDTTs $M^3, \cdots, M^m$ same as M^2, then $M_\emptyset^m = \cdots = M_\emptyset^2$ holds. Moreover, we convert M^m into a single state lsb-PDTT $M' = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma' \cup \{\sharp\}, \delta', *, S, \emptyset)$ according to Lemma 4, then the translation function of M' consists only of the form: $\delta'(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (\alpha(y_r), \tilde{t})$, where $\sigma \neq \varepsilon$, $1 \leq r \leq n$, and $Z_i \in \Gamma$ with $1 \leq i \leq n$. Thus we have the theorem since $M'_\emptyset = M_\emptyset^m = \cdots = M_\emptyset^0 = M_\emptyset$ holds. □

§6. A Comparison of b-PDTT and t-PDTT

We have introduced a b-PDTT which is considered as a dual concept of t-PDTT and shown some fundamental properties in the previous sections. Based on these facts, we now compare the translational capability of t-PDTT with that of b-PDTT in this section.

Definition 16. For a t-PDTT (or b-PDTT) $M = (Q, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, q_0, Z_0, F)$ with a translation function $\delta(\sigma(x_1, \cdots, x_n), p, Z(\overrightarrow{y})) \ni t'_{\Delta\langle\$'\rangle} \langle([q_1, u_1], x_{1'}), \cdots, ([q_l, u_l], x_{l'})\rangle$ (or $\delta(\sigma(x_1, \cdots, x_n), ([p_1, Z_1(\overrightarrow{y_1})], \cdots, [p_n, Z_n(\overrightarrow{y_n})]) \ni ([q, u], t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{l'}\rangle))$ where $\{x_{1'}, \cdots, x_{l'}\} \subseteq X_n$, the translation function is said to be *variable linear* (or simply *linear*) if no variables in X_n occurs more than once in $\{x_{1'}, \cdots, x_{l'}\}$. A PDTT is *variable linear* (or simply *linear*) if any translation function of the PDTT is linear. The translation function is said to be *variable nondeleting* (or simply *nondeleting*) if each variable in X_n occurs at least once in $\{x_{1'}, \cdots, x_{l'}\}$. And a PDTT is *variable nondeleting* (or simply *nondeleting*) if any translation function of the PDTT is nondeleting.

Remark. The previous theorems (from Theorem 1 to Theorem 8) also hold in the case of linear or nondeleting linear t-PDTT (b-PDTT), since their proofs do not depend on whether a translation function is linear, nondeleting linear or not. We also note that the output tree of the translation function in b-PDTT is denoted by $t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{l'}\rangle$ instead of the notation $t' \in T_{\Delta\langle\$'\rangle}(X_n)$ for the sake of comparing the translational capability of t-PDTT and b-PDTT.

We henceforth assume that all t-PDTT and b-PDTT have a linear stack structure. Therefore a linear lst-PDTT (linear lsb-PDTT) is abbreviated as a lt-PDTT (lb-PDTT). Furthermore the class of translations generated by a transducer C (for example C=t-PDTT, lb-PDTT) is denoted by **C** (for example **t-PDTT**, **lb-PDTT**). As indicated in [5], translations of t-PDTT and b-PDTT differ in the following points. That is,

(1) t-PDTT can copy a subtree of the input tree before nondeterministic generation. And decides whether to delete a subtree or not before generating it.

(2) b-PDTT can copy (or delete) a subtree of the input tree after nondeterministic generation.

It will be anticipated that the translational capability of t-PDTT is different from that of b-PDTT from the above mentioned facts. In fact, we can show the next theorem.

Theorem 9. **b-PDTT** and **t-PDTT** are incomparable.

Proof. Consider the t-PDTT $M^1 = (\{*\}, \Sigma\cup\{\$\}, \Sigma\cup\{\$'\}, \Gamma\cup\{\sharp\}, \delta_1, *, Z_0, \emptyset)$, where $\Sigma = \{a, b, \sigma\}, \Gamma = \{Z, Z_0\}$, and δ_1 is defined as follows:

$\delta_1(*, a(x), Z_0(\sharp)) = \{a(([*, Z(\sharp)], x))\}$,
$\delta_1(*, a(x), Z(y)) = \{a(([*, Z(Z(y))], x))\}$,
$\delta_1(*, \sigma(x_1, x_2), Z(y)) = \{\sigma(([*, Z(y)], x_1), ([*, Z(y)], x_2))\}$,
$\delta_1(*, b(x), Z(y)) = \{b(([*, y], x))\}$, and
$\delta_1(*, b(\$), Z(\sharp)) = \{b(([*, \sharp], \$'))\}$.

This M^1 generates the translation $M^1_\emptyset = \{\langle a^n(\sigma(b^n, b^n)), a^n(\sigma(b^n, b^n))\rangle \mid n \geq 1\}$. Here, suppose that there exists a G-type normal form b-PDTT $M = (\{*\}, \Sigma \cup \{\$\}, \Sigma \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$ with $M_\emptyset = M^1_\emptyset$. And for $t = a^n(\sigma(b^n, b^n))$, there exists a generation

$t_{\Sigma\langle\$\rangle}[([*, S(\sharp)], \$), ([*, S(\sharp)], \$)]$
$\vdash^*_M t_{\Sigma\langle\$\rangle}[([*, X_n(\alpha_n(\sharp))], b^n(\$')), ([*, Y_n(\beta_n(\sharp))], b^n(\$'))]$
$\vdash_M t_{\Sigma\langle\$\rangle}[([*, \gamma_n(\theta_n(\sharp))], \sigma(b^n(\$'), b^n(\$')))]$
$\vdash^*_M([*, \sharp], a^n(\sigma(b^n(\$'), b^n(\$'))))$,

where the applied translation function is one of the following forms: (1) $\delta(\sigma(x_1, x_2), (X_n(y_1), Y_n(y_2))) \ni (\gamma_n(y_1), \sigma(x_1, x_2))$ with $\theta_n(\sharp) = \alpha_n(\sharp)$, (2) $\delta(\sigma(x_1, x_2), (X_n(y_1), Y_n(y_2))) \ni (\gamma_n(y_2), \sigma(x_1, x_2))$ with $\theta_n(\sharp) = \beta_n(\sharp)$, and (3) $\delta(\sigma(x_1, x_2), (X_n(y_1), Y_n(y_2))) \ni (\gamma_n(\sharp), \sigma(x_1, x_2))$ with $\theta_n(\sharp) = \sharp$. That is, $\theta_n(\sharp)=\alpha_n(\sharp)$, $\beta_n(\sharp)$, or $\sharp$. On the other hand, we pay attention to a series of the pair (X_i, Y_i) with $1 \leq i \leq n$. If $n \geq |\Gamma|^2+1$ then there exist n_1 and n_2, where $1 \leq n_1 < n_2 \leq n$, such that $(X_{n_1}, Y_{n_1}) = (X_{n_2}, Y_{n_2})$ for $(X_1, Y_1), (X_2, Y_2), \cdots, (X_n, Y_n)$. Thus for $t' = a^{n_1}(\sigma(b^{n_1}, b^{n_1}))$, if the generation

$t'_{\Sigma\langle\$\rangle}[([*, S(\sharp)], \$), ([*, S(\sharp)], \$)]$
$\vdash^*_M t'_{\Sigma\langle\$\rangle}[([*, X_{n_1}(\alpha_{n_1}(\sharp))], b^{n_1}(\$')), ([*, Y_{n_1}(\beta_{n_1}(\sharp))], b^{n_1}(\$'))]$
$\vdash_M t'_{\Sigma\langle\$\rangle}[([*, \gamma_{n_1}(\theta_{n_1}(\sharp))], \sigma(b^{n_1}(\$'), b^{n_1}(\$')))]$
$\vdash^*_M([*, \sharp], a^{n_1}(\sigma(b^{n_1}(\$'), b^{n_1}(\$'))))$

exists in M, then there exist the following generations:

(1) If $\theta_{n_1} = \alpha_{n_1}$ then, for $t'' = a^{n_1}(\sigma(b^{n_1}, b^{n_2}))$, the generation

$$t''_{\Sigma\langle\$\rangle}[([*, S(\sharp)], \$), ([*, S(\sharp)], \$)]$$
$$\vdash^*_M t''_{\Sigma\langle\$\rangle}[([*, X_{n_1}(\alpha_{n_1}(\sharp))], b^{n_1}(\$')), ([*, Y_{n_2}(\beta_{n_2}(\sharp))], b^{n_2}(\$'))]$$
$$= t''_{\Sigma\langle\$\rangle}[([*, X_{n_1}(\alpha_{n_1}(\sharp))], b^{n_1}(\$')), ([*, Y_{n_1}(\beta_{n_2}(\sharp))], b^{n_2}(\$'))]$$
$$\vdash_M t''_{\Sigma\langle\$\rangle}[([*, \gamma_{n_1}(\theta_{n_1}(\sharp))], \sigma(b^{n_1}(\$'), b^{n_2}(\$')))]$$
$$\vdash^*_M ([*, \sharp], a^{n_1}(\sigma(b^{n_1}(\$'), b^{n_2}(\$'))))$$

exists in M.

(2) If $\theta_{n_1} = \beta_{n_1}$ then, for $t'' = a^{n_1}(\sigma(b^{n_2}, b^{n_1}))$, the generation

$$t''_{\Sigma\langle\$\rangle}[([*, S(\sharp)], \$), ([*, S(\sharp)], \$)]$$
$$\vdash^*_M ([*, \sharp], a^{n_1}(\sigma(b^{n_2}(\$'), b^{n_1}(\$'))))$$

exists in M by the same way as (1).

(3) If $\theta_{n_1} = \sharp$ then, for $t'' = a^{n_1}(\sigma(b^{n_2}, b^{n_2}))$, the generation

$$t''_{\Sigma\langle\$\rangle}[([*, S(\sharp)], \$), ([*, S(\sharp)], \$)]$$
$$\vdash^*_M ([*, \sharp], a^{n_1}(\sigma(b^{n_2}(\$'), b^{n_2}(\$'))))$$

exists in M by the same way as (1).

However, $M^1_\emptyset$ does not contain the translations $\langle a^{n_1}(\sigma(b^{n_1}, b^{n_2})), a^{n_1}(\sigma(b^{n_1}, b^{n_2}))\rangle$, $\langle a^{n_1}(\sigma(b^{n_2}, b^{n_1})), a^{n_1}(\sigma(b^{n_2}, b^{n_1}))\rangle$, and $\langle a^{n_1}(\sigma(b^{n_2}, b^{n_2})), a^{n_1}(\sigma(b^{n_2}, b^{n_2}))\rangle$ since $n_1 \neq n_2$, i.e., $M^1_\emptyset \neq M_\emptyset$ holds and this contradicts the assumption. Thus **t-PDTT$\not\subseteq$b-PDTT** holds.

Next, Consider the b-PDTT $M^2 = (\{*\}, \Sigma \cup \{\$\}, \Sigma \cup \{\$'\}, \{Z\} \cup \{\sharp\}, \delta_2, *, Z, \emptyset)$ where $\Sigma = \{a, b\}$, and δ_2 is defined as follows:

$\delta_2(b(\$), Z(\sharp)) = \{(Z(\sharp), b(\$'))\}$,
$\delta_2(b(x), Z(\sharp)) = \{(Z(\sharp), b(x))\}$, and
$\delta_2(a(x_1, x_2), (Z(\sharp), Z(\sharp))) = \{(\sharp, a(x_1))\}$.

This M^2 generates the translation $M^2_\emptyset = \{\langle a(b^n, b^m), a(b^n)\rangle \mid n \geq 1, m \geq 1\}$. Here, suppose that there exists a t-PDTT $M = (\{*\}, \Sigma \cup \{\$\}, \Sigma \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$ with $M_\emptyset = M^2_\emptyset$. And for $t = a(b^n, b^m)$ $(n \geq 1, m \geq 1)$, since there must exist a translation function such that $\delta(*, a(x_1, x_2), X(y)) = a([*, Y(y)], x_1)$ $(X \in \Gamma,\ Y \in \Gamma \cup \{\varepsilon\})$, we have the generation

$$([*, S(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^*_M ([*, X(\alpha(\sharp))], t_{\Sigma\langle\$\rangle})$$
$$\vdash_M a([*, Y(\alpha(\sharp))], b^n(\$))$$
$$\vdash^*_M a(b^n([*, \sharp], \$')),$$

where $\alpha \in (\Gamma)^*$. On the other hand for $t' = a(b^n, t'')$ $(n \geq 1$, any $t'' \in T_\Sigma)$, the generation

$$([*, S(\sharp)], t'_{\Sigma\langle\$\rangle}) \vdash^*_M ([*, X(\alpha(\sharp))], t'_{\Sigma\langle\$\rangle})$$
$$\vdash_M a([*, Y(\alpha(\sharp))], b^n(\$))$$

$\vdash_M^* a(b^n([*, \sharp], \$'))$

exists. However, $M_\emptyset^2$ does not contain the translations $\langle a(b^n, t''), a(b^n)\rangle$ where $n \geq 1$ and any $t'' \in T_\Sigma$, i.e., $M_\emptyset^2 \neq M_\emptyset$ holds and this contradicts the assumption. Thus **b-PDTT**$\not\subseteq$**t-PDTT** holds and this completes the proof. □

Before proving the similar results for lt-PDTT (lb-PDTT) and nondeleting lt-PDTT (nondeleting lb-PDTT), we have the following theorem.

Theorem 10.

(1) **nondeleting lt-PDTT**$\subset$**lt-PDTT**$\subset$**t-PDTT**,

(2) **nondeleting lb-PDTT**$\subset$**lb-PDTT**$\subset$**b-PDTT**.

Proof. We pay attention to the translation $\tau_1 = \{\langle a(b^n), a(b^n, b^n)\rangle \mid n \geq 1\}$. Clearly there exists a t-PDTT M^1 with $M_\emptyset^1 = \tau_1$, but no lt-PDTT M with $M_\emptyset = \tau_1$ exists. Contrary to this, for the translation $\tau_2 = \{\langle a(b^n), a\rangle \mid n \geq 1\}$, there exists a lt-PDTT M^2 with $M_\emptyset^2 = \tau_2$, but no nondeleting lt-PDTT M with $M_\emptyset = \tau_2$ exists. On the other hand **nondeleting lt-PDTT**$\subseteq$**lt-PDTT**$\subseteq$**t-PDTT** holds. Thus we have **nondeleting lt-PDTT**$\subset$**lt-PDTT**$\subset$**t-PDTT**. Furthermore, we have **nondeleting lb-PDTT**$\subset$**lb-PDTT**$\subset$**b-PDTT** by almost a similar procedure to t-PDTT case, and this completes the proof. □

Theorem 11.

(1) **b-PDTT** is incomparable with **lt-PDTT** and **nondeleting lt-PDTT**.

(2) **lb-PDTT** is incomparable with **t-PDTT**, **lt-PDTT**, and **nondeleting lt-PDTT**.

Proof. Let M^1 be as in the Theorem 9. Clearly this M^1 is a nondeleting lt-PDTT and **nondeleting lt-PDTT** $\not\subseteq$ **b-PDTT** holds. Moreover, we have the inclusion properties **lt-PDTT** $\not\subseteq$ **b-PDTT**, **t-PDTT** $\not\subseteq$ **b-PDTT**, **nondeleting lt-PDTT** $\not\subseteq$ **lb − PDTT**, **lt-PDTT** $\not\subseteq$ **lb-PDTT**, and **t-PDTT** $\not\subseteq$ **lb-PDTT** since **nondeleting lt-PDTT** $\subset$ **lt-PDTT** $\subset$ **t-PDTT** and **lb-PDTT** $\subset$ **b-PDTT** hold from Theorem 10. Next, let M^2 be as in the Theorem 9. Clearly this M^2 is the lb-PDTT and **lb-PDTT** $\not\subseteq$ **t-PDTT** holds. Moreover, we have the inclusion properties **lb-PDTT** $\not\subseteq$ **lt-PDTT**, **lb-PDTT** $\not\subseteq$ **nondeleting lt-PDTT**, **b-PDTT** $\not\subseteq$ **t-PDTT**, **b-PDTT** $\not\subseteq$ **lt-PDTT**, and **b-PDTT** $\not\subseteq$ **nondeleting lt-PDTT** since **nondeleting lt-PDTT** $\subset$ **lt-PDTT** $\subset$ **t-PDTT** and **lb-PDTT** $\subset$ **b-PDTT** hold from Theorem 10. This completes the proof. □

We prove the next theorem, which is the main subject of this paper, before ending this section.

Theorem 12. Let M be a nondeleting lb-PDTT. And for any $\langle t, t'\rangle \in M_\emptyset$, there exists a nondeleting lt-PDTT M''' such that $\langle t, t'\rangle \in M'''_\emptyset$.

Proof. Suppose that $M = (\{*\}, \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma \cup \{\sharp\}, \delta, *, S, \emptyset)$ is a G-type normal form in nondeleting lb-PDTT. From M, we have the extended nondeleting lt-PDTT $M' = (Q', \Sigma \cup \{\$\}, \Delta \cup \{\$'\}, \Gamma' \cup \{\sharp\}, \delta', q_0, W_0, \emptyset)$ as follows:

(1) $Q' = \{*\} \cup \{q_0\} \cup \{q_X \mid \text{ for any } X \in \Gamma\}$,

(2) $\Gamma' = \Gamma \cup \{W_0\} \cup \{W_X \mid \text{ for any } X \in \Gamma\}$, where $\Gamma'_1 = \Gamma_1, \Gamma'_0 = \Gamma_0 \cup \{W_0\} \cup \{W_X \mid \text{ for any } X \in \Gamma\}$, and

(3) δ' is defined as follows:

(a) For $\delta(\sigma(x_1, \cdots, x_n), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (\alpha(y_r), t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$, where $\sigma \in \Sigma_n$ and $\{x_{1'}, \cdots, x_{n'}\} = X_n$,

(i) if $\alpha(y_r) = Y_1(Y_2(\cdots Y_m(y_r)\cdots))$ with $m \geq 1$, then
$$\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(y)\cdots))) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(y)], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle,$$
where $Z_{j'} = Z_i$ for $x_{j'} = x_i$ with $1 \leq i \leq n$ and $1 \leq j' \leq n$,

(ii) if $\alpha(y_r) = y_r$, then for any $W \in \Gamma_1$,
$$\delta'(*, \sigma(\overrightarrow{x}), W(y)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(W(y))], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle,$$
where $Z_{j'} = Z_i$ for $x_{j'} = x_i$ with $1 \leq i \leq n$ and $1 \leq j' \leq n$,

(iii) if $\alpha(y_r) = Y_1(Y_2(\cdots Y_m(\sharp)\cdots))$ with $m \geq 1$, then
$$\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(\sharp)\cdots))) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle,$$
where $Z_{j'} = Z_i$ for $x_{j'} = x_i$ with $1 \leq i \leq n$ and $1 \leq j' \leq n$, and

(iv) if $\alpha(y_r) = \sharp$, then
$$\delta'(q_0, \sigma(\overrightarrow{x}), W_0(\sharp)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z'_1}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle,$$
where $Z_{j'} = Z_i$ for $x_{j'} = x_i$ with $1 \leq i \leq n$ and $1 \leq j' \leq n$.

In the case of (i) or (ii), we assume that $Z_r(y_r) \neq Z_r(\sharp)$ and in the case of (iii) or (iv), we permit that $Z_r(y_r) = Z_r(\sharp)$.

(b) δ' which is independent of δ is as follows:

(i) for any $X \in \Gamma$, $\delta'(q_X, \varepsilon(x), W_X(\sharp)) \ni ([q_X, Z(\sharp)], x)$ with any $Z \in \Gamma_0$,

(ii) for any $X, Z \in \Gamma$, $\delta'(q_X, \varepsilon(x), Z(y)) \ni ([q_X, W(Z(y))], x)$ with any $W \in \Gamma$ and including $Z(y) = Z(\sharp)$,

(iii) for any $X \in \Gamma$, $\delta'(q_X, \varepsilon(x), X(y)) \ni ([*, X(y)], x)$ with including $X(y) = X(\sharp)$, and

(iv) $\delta'(*, \varepsilon(x), S(\sharp)) \ni ([*, \sharp], x)$.

For the extended nondeleting lt-PDTT M' defined above, we have the following two lemmas.

Lemma 6. For any $\langle t, t' \rangle \in T_\Sigma \times T_\Delta$ and $\gamma \in (\Gamma)^+$,

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle,$$

if and only if

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash^*_M ([*, \gamma(\sharp)], t'_{\Delta\langle\$'\rangle}).$$

Proof of Lemma. If part: The proof is an induction on the number k of generation steps.

(1) Basis: If

$$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_M ([*, \gamma(\sharp)], t'_{\Delta\langle\$'\rangle}),$$

then there exists the translation function

$$\delta(\sigma(\$_{(1)}, \cdots, \$_{(l_0)}), (S_{(1)}(\sharp), \cdots, S_{(l_0)}(\sharp))) \ni (\gamma(\sharp), t'_{\Delta\langle\$'\rangle})$$

in M and $t_{\Sigma\langle\$\rangle} = \sigma(\$_{(1)}, \cdots, \$_{(l_0)})$ holds. On the other hand there exists the translation function

$$\delta'(*, \sigma(\$_{(1)}, \cdots, \$_{(l_0)}), \gamma(\sharp)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{S_{(1)}}, W_{S_{(1)}}(\sharp)], \$'), \cdots, ([q_{S_{(l_0)}}, W_{S_{(l_0)}}(\sharp)], \$')\rangle$$

in M' since $\gamma \neq \varepsilon$. Therefore,

$$\begin{aligned}([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle}) &= ([*, \gamma(\sharp)], \sigma(\$_{(1)}, \cdots, \$_{(l_0)})) \\ &\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{S_{(1)}}, W_{S_{(1)}}(\sharp)], \$'), \cdots, ([q_{S_{(l_0)}}, W_{S_{(l_0)}}(\sharp)], \$')\rangle \\ &\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{S_{(1)}}, S_{(1)}(\sharp)], \$'), \cdots, ([q_{S_{(l_0)}}, S_{(l_0)}(\sharp)], \$')\rangle \\ &\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle\end{aligned}$$

holds.

(2) Inductive step: Suppose that the lemma holds for the number k of generation steps. For the generation

$$\begin{aligned}&t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \\ &\vdash^k_M t_{\Sigma\langle\$\rangle}[([*, Z_1(v_1(\sharp))], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(v_n(\sharp))], \tilde{t}_{n\Delta\langle\$'\rangle})] \\ &\vdash_M ([*, \gamma(\sharp)], t'_{\Delta\langle\$'\rangle}),\end{aligned}$$

if the translation function applied last is

$$\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (\alpha(y_r), t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle),$$

where $\sigma \in \Sigma_n$, then $t'_{\Delta\langle\$'\rangle} = t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i), \cdots, x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i)\rangle$ and $t_{\Sigma\langle\$\rangle} = \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})$ where $t_{i\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}$ with $1 \leq i \leq n$. And the following facts hold:

(i) If $\alpha(y_r) = Y_1(Y_2(\cdots Y_m(y_r)\cdots))$with $m \geq 1$, then $\gamma(\sharp) = Y_1(Y_2(\cdots Y_m(v_r (\sharp))\cdots))$ holds. And there exists the translation function

$$\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(y)\cdots)))$$
$$\ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(y)], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}} (\sharp)], x_{n'})\rangle$$

in M'. Therefore we have the generation

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$= ([*, Y_1(Y_2(\cdots Y_m(v_r(\sharp))\cdots))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(v_r(\sharp))], x_{r'} (t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}, (\sharp)] x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, Z_1(v_{1'}(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(v_r(\sharp))], x_{r'} (t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, Z_{n'}(v_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(v_r(\sharp))], x_{r'} (t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(v_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle.$$

(ii) If $\alpha(y_r) = y_r$, then $\gamma(\sharp) = v_r(\sharp)$ with $v_r \neq \varepsilon$ holds. And for any $W \in \Gamma_1$, there exists the translation function

$$\delta'(*, \sigma(\overrightarrow{x}), W(y))$$
$$\ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(W(y))], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$$

in M'. Therefore, for $v_r(\sharp) = W(v_{r'}(\sharp))$, we have the generation

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$= ([*, W(v_{r'}(\sharp))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(W(v_r(\sharp)))], x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(v_r(\sharp))], x_{r'} (t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(v_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle.$$

(iii) If $\alpha(y_r) = Y_1(Y_2(\cdots Y_m(\sharp)\cdots))$ with $m \geq 1$, then $\gamma(\sharp) = Y_1(Y_2(\cdots Y_m(\sharp) \cdots))$ holds. And there exists the translation function

$$\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(\sharp)\cdots)))$$
$$\ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$$

in M'. Therefore we have the generation

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$$

$= ([*, Y_1(Y_2(\cdots Y_m(\sharp)\cdots))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$
$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}$
$(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$
$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(v_n(\sharp))], x_{n'}$
$(t_{i\Sigma\langle\$\rangle}/x_i))\rangle.$

(iv) If $\alpha(y_r) = \sharp$ then $\gamma = \varepsilon$ holds. However this is not the case since $\gamma \neq \varepsilon$.

For any i $(1 \le i \le n)$, if there exists a generation
$t_{i\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_i)}(\sharp)], \$)] \vdash^{k_i}_M ([*, Z_i(v_i(\sharp))], \tilde{t}_{i\Delta\langle\$'\rangle})$
in M, then there exists the generation
$([*, Z_i(v_i(\sharp))], t_{i\Sigma\langle\$\rangle}) \vdash^*_{M'} \tilde{t}_{i\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_i)}(\sharp)], \$')\rangle$
in M' by the inductive hypothesis since $k_i \le k$. Thus we have the generation
$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$
$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(v_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/$
$x_i))\rangle$
$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle$
since $t'_{\Delta\langle\$'\rangle} = t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i), \cdots, x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i)\rangle$, i.e., the lemma holds for the number $k+1$ of generation steps.

Only if part: Before proving the only if part, we define a relation $\vdash^{(k)}_{M'}$ as follows. For $\alpha = t_{\Delta\langle\$'\rangle}[([q_1, u_1(\sharp)], t_{1\Sigma\langle\$\rangle}), \cdots, ([q_s, u_s(\sharp)], t_{s\Sigma\langle\$\rangle})]$ in M', we define S_t as a mapping from α to a subset of states in M' with $S_t(\alpha) = \{q_1, \cdots, q_s\}$. And for a series of generation $\alpha_1 \vdash_{M'} \alpha_2 \vdash_{M'} \cdots \vdash_{M'} \alpha_{m-1} \vdash_{M'} \alpha_m$ in M, where we assume that none of the translation function is applied to the node whose state is $*$ in $\alpha_i \vdash_{M'} \alpha_{i+1}$ with $1 \le i \le m-1$, if $S_t(\alpha_1) = S_t(\alpha_m) = \{*\}$ and $S_t(\alpha_i) \neq \{*\}$ with $2 \le i \le m-1$ then we denote $\alpha_1 \vdash^{(1)}_{M'} \alpha_m$. When $\beta_1 \vdash^{(1)}_{M'} \beta_2 \vdash^{(1)}_{M'} \cdots \beta_k \vdash^{(1)}_{M'} \beta_{k+1}$, we denote $\beta_1 \vdash^{(k)}_{M'} \beta_{k+1}$. Under this definition, the proof is an induction on the number k of $\vdash^{(k)}_{M'}$.

(1) Basis: For the generation
$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^{(1)}_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle,$
let $t_{\Sigma\langle\$\rangle}$ be $\sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})$ where $\sigma \in \Sigma_n$ and $t_{i\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}$ with $1 \le i \le n$. We consider the following cases according to the translation function applied to this generation.

(i) Suppose that a translation function $\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(y)\cdots))) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(y)], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ is applied. If $\gamma(\sharp) = Y_1(Y_2(\cdots Y_m(\gamma'(\sharp))\cdots))$ with $\gamma' \neq \varepsilon$, then there exists the generation

$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$
$= ([*, Y_1(Y_2(\cdots Y_m(\gamma'(\sharp))\cdots))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$

$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle(([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(\gamma'(\sharp))], x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle(([q_{Z_{1'}}, Z_{1'}(\gamma_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(\gamma'(\sharp))], x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, Z_{n'}(\gamma_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle(([*, Z_{1'}(\gamma_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(\gamma'(\sharp))], x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(\gamma_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle,$$

i.e.,

$$([*, \gamma(\sharp)], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$$
$$\vdash^{(1)}_{M'} t'_{\Delta\langle\$'\rangle}\langle(([*, Z_{1'}(\gamma_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(\gamma'(\sharp))], x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(\gamma_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$= t'_{\Delta\langle\$'\rangle}\langle(([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle.$$

However, we obtain $Z_{r'}(\gamma') \neq S$ since $Z_{r'} \neq \varepsilon$ and $\gamma' \neq \varepsilon$, and this is not the case.

(ii) Suppose that a translation function $\delta'(*, \sigma(\overrightarrow{x}), W(y)) \ni t'_{\Delta\langle\$'\rangle}\langle(([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(W(y))], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ is applied. If $\gamma(\sharp) = W(\gamma'(\sharp))$ then there exists the generation

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$= ([*, W(\gamma'(\sharp))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle(([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(W(\gamma'(\sharp)))], x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle.$$

However, we obtain $Z_{r'}(W(\gamma')) \neq S$ since $Z_{r'} \neq \varepsilon$ and $W \neq \varepsilon$, and this is also not the case.

(iii) Suppose that a translation function $\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(\sharp)\cdots))) \ni t'_{\Delta\langle\$'\rangle}\langle(([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ is applied. If $\gamma(\sharp) = Y_1(Y_2(\cdots Y_m(\sharp)\cdots))$ with $m \geq 1$ then there exists the generation

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$= ([*, Y_1(Y_2(\cdots Y_m(\sharp)\cdots))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle}))$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle(([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle(([*, Z_{1'}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$= t'_{\Delta\langle\$'\rangle}\langle(([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle.$$

Thus $t_{\Sigma\langle\$\rangle} = \sigma(\$_{(1)}, \cdots, \$_{(l_0)})$ holds since $n = l_0$, $t_i = \$$ and $Z_{i'} = S$ with

$1 \leq i \leq n$. On the other hand, since there exists the translation function $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (Y_1(Y_2(\cdots Y_m(\sharp)\cdots)), t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$ in M, we have the generation

$$\begin{aligned}
&t_{\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \\
&= \sigma(([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(n)}(\sharp)], \$)) \\
&\vdash_M ([*, (Y_1(Y_2(\cdots Y_m(\sharp)\cdots))], t'_{\Delta\langle\$'\rangle}\langle \$'_{(1)}, \cdots, \$'_{(n)}\rangle) \\
&= ([*, \gamma(\sharp)], t'_{\Delta\langle\$'\rangle})
\end{aligned}$$

and the lemma holds.

(iv) Suppose that a translation function $\delta'(q_0, \sigma(\overrightarrow{x}), W_0(\sharp)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ is applied. However this is not the case since $\gamma \in (\Gamma)^+$.

(2) Inductive step: Suppose that the lemma holds for the number k of $\vdash^{(k)}_{M'}$. For the generation

$$([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^{(k+1)}_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle,$$

let $t_{\Sigma\langle\$\rangle}$ be $\sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})$ where $\sigma \in \Sigma_n$ and $t_{i\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}$ with $1 \leq i \leq n$. We consider the following cases according to the translation function applied to this generation.

(i) Suppose that a translation function $\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(y)\cdots))) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(y)], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ is applied. If $\gamma(\sharp) = Y_1(Y_2(\cdots Y_m(\gamma_r(\sharp))\cdots))$, then there exists the generation

$$\begin{aligned}
&([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle}) \\
&= ([*, Y_1(Y_2(\cdots Y_m(\gamma_r(\sharp))\cdots))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})) \\
&\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(\gamma_r(\sharp))], x_{r'} \\
&\qquad (t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle \\
&\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(\gamma_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(\gamma_r(\sharp))], x_{r'} \\
&\qquad (t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(\gamma_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle,
\end{aligned}$$

where $\gamma_i \in (\Gamma)^*$ with $1 \leq i \leq n$.

(ii) Suppose that a translation function $\delta'(*, \sigma(\overrightarrow{x}), W(y)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(W(y))], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ is applied. If $\gamma(\sharp) = W(\gamma'(\sharp))$, then there exists the generation

$$\begin{aligned}
&([*, \gamma(\sharp)], t_{\Sigma\langle\$\rangle}) \\
&= ([*, W(\gamma'(\sharp))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})) \\
&\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{r'}(W(\gamma'(\sharp)))],
\end{aligned}$$

$$x_{r'}(t_{i\Sigma\langle\$\rangle}/x_i)),\cdots,([q_{Z_{n'}},W_{Z_{n'}}(\sharp)],x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*,Z_{1'}(\gamma_1(\sharp))],x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)),\cdots,([*,Z_{r'}(\gamma_r(\sharp))],x_{r'}$$
$$(t_{i\Sigma\langle\$\rangle}/x_i)),\cdots,([*,Z_{n'}(\gamma_n(\sharp))],x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle,$$

where $\gamma_r(\sharp)=W(\gamma'(\sharp))=\gamma(\sharp)$.

(iii) Suppose that a translation function $\delta'(*,\sigma(\overrightarrow{x}),Y_1(Y_2(\cdots Y_m(\sharp)\cdots)))\ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}},W_{Z_{1'}}(\sharp)],x_{1'}),\cdots,([q_{Z_{n'}},W_{Z_{n'}}(\sharp)],x_{n'})\rangle$ is applied. If $\gamma(\sharp)=Y_1(Y_2(\cdots Y_m(\sharp)\cdots))$, then there exists the generation

$$([*,\gamma(\sharp)],t_{\Sigma\langle\$\rangle})$$
$$=([*,Y_1(Y_2(\cdots Y_m(\sharp)\cdots))],\sigma(t_{1\Sigma\langle\$\rangle},\cdots,t_{n\Sigma\langle\$\rangle}))$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}},W_{Z_{1'}}(\sharp)],x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)),\cdots,([q_{Z_{n'}},W_{Z_{n'}}(\sharp)],$$
$$x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*,Z_{1'}(\gamma_1(\sharp))],x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)),\cdots,([*,Z_{n'}(\gamma_n(\sharp))],$$
$$x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle.$$

(iv) Suppose that a translation function $\delta'(q_0,\sigma(\overrightarrow{x}),W_0(\sharp))\ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}},W_{Z_{1'}}(\sharp)],x_{1'}),\cdots,([q_{Z_{n'}},W_{Z_{n'}}(\sharp)],x_{n'})\rangle$ is applied. However this is not the case since $\gamma\in(\Gamma)^+$.

For any j with $1\le j\le n$, there exists a generation

$$([*,Z_j(\gamma_j(\sharp))],t_{j\Sigma\langle\$\rangle})\vdash^{(k_j)}_{M'}\tilde{t}_{j\Delta\langle\$'\rangle}\langle([*,S_{(1)}(\sharp)],\$'),\cdots,([*,S_{(l_j)}(\sharp)],\$')\rangle$$

in M' and $t'_{\Delta\langle\$'\rangle}=t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i),\cdots,x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i)\rangle$ since

$$t'_{\Delta\langle\$'\rangle}\langle([*,Z_{1'}(\gamma_1(\sharp))],x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)),\cdots,([*,Z_{n'}(\gamma_n(\sharp))],x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash^{(k)}_{M'} t'_{\Delta\langle\$'\rangle}\langle([*,S_{(1)}(\sharp)],\$'),\cdots,([*,S_{(l_0)}(\sharp)],\$')\rangle.$$

Thus there exists the generation

$$t_{j\Sigma\langle\$\rangle}[([*,S_{(1)}(\sharp)],\$),\cdots,([*,S_{(l_j)}(\sharp)],\$)]\vdash^*_M([*,Z_j(\gamma_j(\sharp))],\tilde{t}_{j\Delta\langle\$'\rangle}),$$

i.e.,

$$t_{\Sigma\langle\$\rangle}[([*,S_{(1)}(\sharp)],\$),\cdots,([*,S_{(l_0)}(\sharp)],\$)]$$
$$\vdash^*_M t_{\Sigma\langle\$\rangle}[([*,Z_1(\gamma_1(\sharp))],\tilde{t}_{1\Delta\langle\$'\rangle}),\cdots,([*,Z_n(\gamma_n(\sharp))],\tilde{t}_{n\Delta\langle\$'\rangle})]$$

by the inductive hypothesis since $k_i\in k$. Here, in the case of (i), since there exists a translation function $\delta(\sigma(\overrightarrow{x}),(Z_1(y_1),\cdots,Z_n(y_n)))\ni(Y_1(Y_2(\cdots Y_m(y_r)\cdots)),t'_{\Delta\langle\$'\rangle}\langle x_{1'},\cdots,x_{n'}\rangle)$ in M,

$$t_{\Sigma\langle\$\rangle}[([*,Z_1(\gamma_1(\sharp))],\tilde{t}_{1\Delta\langle\$'\rangle}),\cdots,([*,Z_n(\gamma_n(\sharp))],\tilde{t}_{n\Delta\langle\$'\rangle})]$$
$$\vdash_M([*,(Y_1(Y_2(\cdots Y_m(\gamma(\sharp)))\cdots))],t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i),\cdots,x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/$$
$$x_i)\rangle)$$
$$=([*,\gamma(\sharp)],t'_{\Delta\langle\$'\rangle})$$

holds. In the case of (ii), since there exists a translation function $\delta(\sigma(\overrightarrow{x}),(Z_1(y_1),\cdots,Z_n(y_n)))\ni(y_r,t'_{\Delta\langle\$'\rangle}\langle x_{1'},\cdots,x_{n'}\rangle)$ in M and $\gamma_r(\sharp)=\gamma(\sharp)$,

$$t_{\Sigma\langle\$\rangle}[([*, Z_1(\gamma_1(\sharp))], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(\gamma_n(\sharp))], \tilde{t}_{n\Delta\langle\$'\rangle})]$$
$$\vdash_M ([*, \gamma_r(\sharp)], t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i), \cdots, x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i)\rangle)$$
$$= ([*, \gamma(\sharp)], t'_{\Delta\langle\$'\rangle})$$

holds. And in the case of (iii), since there exists a translation function $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (Y_1(Y_2(\cdots Y_m(\sharp)\cdots)), t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$ in M,

$$t_{\Sigma\langle\$\rangle}[([*, Z_1(\gamma_1(\sharp))], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(\gamma_n(\sharp))], \tilde{t}_{n\Delta\langle\$'\rangle})]$$
$$\vdash_M ([*, Y_1(Y_2(\cdots Y_m(\sharp)\cdots))], t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i), \cdots, x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i)\rangle)$$
$$= ([*, \gamma(\sharp)], t'_{\Delta\langle\$'\rangle})$$

holds. Thus the lemma holds for the number $k+1$ of $\vdash_{M'}^{(k+1)}$

(End of the proof of Lemma 6)

Lemma 7. For any $\langle t, t'\rangle \in T_\Sigma \times T_\Delta$ and subtrees $t_{1\Sigma\langle\$\rangle}, \cdots, t_{l\Sigma\langle\$\rangle}$ of $t_{\Sigma\langle\$\rangle}$,

$$([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], t_{1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle,$$

if and only if

$$t_{\Sigma\langle\$\rangle}[([*, u_1(\sharp)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})] \vdash_M^* ([*, \sharp], t'_{\Delta\langle\$'\rangle}),$$

where $t_{i\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_i)}(\sharp)], \$)] \vdash_M^* ([*, u_i], \tilde{t}_{i\Delta\langle\$'\rangle})$ and $([*, u_i], t_{i\Sigma\langle\$\rangle}) \vdash_{M'}^* \tilde{t}_{i\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_i)(\sharp)}], \$')\rangle$, $u_{j'} = u_i \in (\Gamma)^+$ for $t_{j'\Sigma\langle\$\rangle} = t_{i\Sigma\langle\$\rangle}$, and $t'_{\Delta\langle\$'\rangle} = t'_{\Delta\langle\$'\rangle}\langle\tilde{t}_{1'\Delta\langle\$'\rangle}, \cdots, \tilde{t}_{l'\Delta\langle\$'\rangle}\rangle$ ($\tilde{t}_{j'\Delta\langle\$'\rangle} = \tilde{t}_{i\Delta\langle\$'\rangle}$ for $t_{j'\Sigma\langle\$\rangle} = t_{i\Sigma\langle\$\rangle}$) with $1 \leq i \leq l$ and $1 \leq j' \leq l$.

Proof of Lemma. If part: The proof is an induction on the number k of generation steps.

(1) Basis: For $t_{\Sigma\langle\$\rangle}[([*, u_1(\sharp)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})] \vdash_M ([*, \sharp], t'_{\Delta\langle\$'\rangle})$ and $u_i = Z_i(v_i)$ with $1 \leq i \leq l$, without loss of generality we assume that the applied translation function is $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_l(y_l))) \ni (\sharp, t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{l'}\rangle)$ and $t_{\Sigma\langle\$\rangle} = \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{l\Sigma\langle\$\rangle})$ ($\sigma \in \Sigma_l$ and $t_{1\Sigma\langle\$\rangle}, \cdots, t_{l\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}$). Therefore, since there exists the translation function $\delta'(q_0, \sigma(\overrightarrow{x}), W_0(\sharp)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{l'}}, W_{Z_{l'}}(\sharp)], x_{l'})\rangle$ in M', we obtain

$$([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{l'}}, W_{Z_{l'}}(\sharp)], x_{l'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$\vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_{1'}(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{l'}(v_{l'}(\sharp))], x_{l'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$$
$$= t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, u_{l'}(\sharp)], x_{l'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle,$$

where $v_{j'} = v_h$ for $x_{j'} = x_h$ with $1 \leq j \leq l$ and $1 \leq h \leq l$, and the lemma holds.

(2) Inductive step: Suppose that the lemma holds for the number k of generation steps. Without loss of generality we assume that

$$\beta = t_{\Sigma\langle\$\rangle}[([*, u_1(\sharp)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_n(\sharp)], \tilde{t}_{n\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$$

$$= t_{\Sigma\langle\$\rangle}[([*, Z_1(v_1(\sharp))], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(v_n(\sharp))], \tilde{t}_{n\Delta\langle\$'\rangle}), ([*, u_{n+1}(\sharp)], \tilde{t}_{n+1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$$
$$\vdash_M^{k+1} ([*, \sharp], t'_{\Delta\langle\$'\rangle})$$

is generated by the next step. If $\sigma(([*, Z_1(v_1(\sharp))], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(v_n(\sharp))], \tilde{t}_{n\Delta\langle\$'\rangle}))$ is the subtree of β and the applied translation function is

(i) $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n)))$
$\ni (Y_1(Y_2(\cdots Y_m(y_r)\cdots)), \tilde{t}_{(1)\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$,

then

$$\beta \vdash_M t_{\Sigma\langle\$\rangle}[([*, Y_1(Y_2(\cdots Y_m(v_r(\sharp))\cdots))], \tilde{t}_{(1)\Delta\langle\$'\rangle}), ([*, u_{n+1}(\sharp)], \tilde{t}_{n+1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$$
$$\vdash_M^k ([*, \sharp], t'_{\Delta\langle\$'\rangle})$$

holds. Without loss of generality, there exists the generation

$$([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$\vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, Y_1(Y_2(\cdots Y_m(v_r(\sharp))\cdots))], \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})), ([*, u_{n+1'}(\sharp)], t_{n+1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$$

in M' by the inductive hypothesis and $t'_{\Delta\langle\$'\rangle} = t'_{\Delta\langle\$'\rangle}\langle\tilde{t}_{(1)\Delta\langle\$'\rangle}, \tilde{t}_{n+1\Delta\langle\$'\rangle}, \cdots, \tilde{t}_{l\Delta\langle\$'\rangle}\rangle$. On the other hand there exists the translation function $\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(y_r)\cdots))) \ni \tilde{t}_{(1)\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(y)], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$ in M', we obtain

$$([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle})$$
$$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle\tilde{t}_{(1)\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{j\Sigma\langle\$\rangle}/x_j)), \cdots, ([*, Z_{r'}(y)], x_{r'}(t_{j\Sigma\langle\$\rangle}/x_j)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{j\Sigma\langle\$\rangle}/x_j))\rangle, ([*, u_{n+1'}(\sharp)], t_{n+1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$$
$$\vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle\tilde{t}_{(1)\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_{1'}((\sharp))], x_{1'}(t_{j\Sigma\langle\$\rangle}/x_j)), \cdots, ([*, Z_{n'}(v_{n'}((\sharp))], x_{n'}(t_{j\Sigma\langle\$\rangle}/x_j))\rangle, ([*, u_{n+1'}(\sharp)], t_{n+1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$$
$$= t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, u_{l'}(\sharp)], x_{l'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle.$$

where $v_{m'} = v_h$ for $x_{m'} = x_h$ with $1 \le m \le n$ and $1 \le h \le n$. Thus the lemma holds. If the applied translation function is

(ii) $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (y_r, \tilde{t}_{(1)\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$, and

(iii) $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, (y_n)))$
$\ni (Y_1(Y_2(\cdots Y_m(\sharp)\cdots)), \tilde{t}_{(1)\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$,

then we obtain the lemma by a similar way to (i). Thus the lemma holds for the number $k+1$ of generation steps.

Only if part: the proof is an induction on the number k of $\vdash_{M'}^{(k)}$

(1) Basis: For $([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash_{M'}^{(1)} t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], t_{1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$, the applied translation function is $\delta'(q_0, \sigma(\overrightarrow{x}), W_0(\sharp)) \ni t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$. That is, for $t_{\Sigma\langle\$\rangle} = \sigma(t_{1\Sigma\langle\$\rangle}, \cdots, t_{n\Sigma\langle\$\rangle})$, where $\sigma \in \Sigma_n$ and $t_{j\Sigma\langle\$\rangle} \in T_{\Sigma\langle\$\rangle}$ with $1 \le i \le n$,

$([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle})$
$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$
$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, Z_{1'}(v_1(\sharp))], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, Z_{n'}(v_n(\sharp))], x_{n'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle$
$= t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], x_{1'}(t_{i\Sigma\langle\$\rangle}/x_i)), \cdots, ([*, u_{l'}(\sharp)], x_{l'}(t_{i\Sigma\langle\$\rangle}/x_i))\rangle,$

where $v_{j'} = v_h$ for $x_{j'} = x_h$ with $1 \leq j \leq l$ and $1 \leq h \leq l$, i.e., $l = n$ and $u_{i'} = Z_{i'}(v_i)$ with $1 \leq i \leq l$ is concluded. On the other hand, since there exists the translation function $\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n))) \ni (\sharp, t'_{\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$ in M, we obtain

$t_{\langle\$\rangle}[([*, u_1(\sharp)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$
$= \sigma(([*, Z_1(v_1(\sharp))], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(v_n(\sharp))], \tilde{t}_{n\Delta\langle\$'\rangle}))$
$\vdash_M ([*, \sharp], t'_{\Delta\langle\$'\rangle})$

since $t'_{\Delta\langle\$'\rangle} = t'_{\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i), \cdots, x_{n'}(\tilde{t}_{i\Delta\langle\$'\rangle}/x_i)\rangle$ and the lemma holds.

(2) Inductive step: Suppose that the lemma holds for the number k of $\vdash^{(k)}_{M'}$. For $([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash^{(k)}_{M'} t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], t_{1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$, if the translation function applied next is

(i) $\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(y)\cdots)))$
$\ni \tilde{t}_{1\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(y)], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle,$

then for $u_{1'} = Y_1(Y_2(\cdots Y_m(v_1)\cdots))$ with $v_1 \neq \sharp$ and $t_{1'\Sigma\langle\$\rangle} = \sigma(t_{11\Sigma\langle\$\rangle}, \cdots, t_{1n\Sigma\langle\$\rangle})$, there exists a generation

$t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], t_{1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$
$\vdash_{M'} t'_{\Delta\langle\$'\rangle}\langle\tilde{t}_{1\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}(t_{1j\Sigma\langle\$\rangle}/x_j)), \cdots, ([*, Z_{r'}(v_1(\sharp))], x_{r'}(t_{1j\Sigma\langle\$\rangle}/x_j)), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'}(t_{1j\Sigma\langle\$\rangle}/x_j))\rangle, ([*, u_{2'}(\sharp)], t_{2'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$
$\vdash^*_{M'} t'_{\Delta\langle\$'\rangle}\langle\tilde{t}_{1\Delta\langle\$'\rangle}\langle([*, Z_{1'}(w_1((\sharp))], x_{1'}(t_{1j\Sigma\langle\$\rangle}/x_j)), \cdots, ([*, Z_{r'}(v_1(\sharp))], x_{r'}(t_{1j\Sigma\langle\$\rangle}/x_j)), \cdots, ([*, Z_{n'}(w_n(\sharp)], x_{n'}(t_{1j\Sigma\langle\$\rangle}/x_j))\rangle, ([*, u_{2'}(\sharp)], t_{2'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle,$

where $w_i \in (\Gamma)^+$ with $1 \leq i \leq n$ and $i \neq r$ in M'. On the other hand, there exists the translation function

$\delta(\sigma(\overrightarrow{x}), (Z_1(y_1), \cdots, Z_n(y_n)))$
$\ni (Y_1(Y_2(\cdots Y_m(y_r)\cdots)), \tilde{t}_{1\Delta\langle\$'\rangle}\langle x_{1'}, \cdots, x_{n'}\rangle)$

in M and supposing $t_{1\Sigma\langle\$\rangle} = t_{1'\Sigma\langle\$\rangle}$ and $\tilde{t}_{1i\Delta\langle\$'\rangle} \in T_{\Delta\langle\$'\rangle}$ which satisfies $t_{1i\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_i)}(\sharp)], \$)] \vdash^*_M ([*, u_i], \tilde{t}_{1i\Delta\langle\$'\rangle})$ with $1 \leq i \leq n$, we obtain

$t_{\Sigma\langle\$\rangle}[\sigma(([*, Z_1(w_1(\sharp))], \tilde{t}_{11\Delta\langle\$'\rangle}), \cdots, ([*, Z_r(v_1(\sharp))], \tilde{t}_{1r\Delta\langle\$'\rangle}), \cdots, ([*, Z_n(w_n(\sharp))], \tilde{t}_{1n\Delta\langle\$'\rangle})), ([*, u_2(\sharp)], \tilde{t}_{2\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$
$\vdash_M t_{\Sigma\langle\$\rangle}[([*, Y_1(Y_2(\cdots Y_m(v_1)\cdots))], \tilde{t}_{1\Delta\langle\$'\rangle}), ([*, u_2(\sharp)], \tilde{t}_{2\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$

$= t_{\Sigma\langle\$\rangle}[([*, u_1(\sharp)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$
$\vdash_M^* ([*, \sharp], t'_{\Delta\langle\$'\rangle})$

since $\tilde{t}_{1\Delta\langle\$'\rangle} = \tilde{t}_{1\Delta\langle\$'\rangle}\langle x_{1'}(\tilde{t}_{1i\Delta\langle\$'\rangle}/x_i), \cdots, x_{n'}(\tilde{t}_{1i\Delta\langle\$'\rangle}/x_i)\rangle$. Moreover we have a similar result when we suppose that $t_{j\Sigma\langle\$\rangle} = t_{1'\Sigma\langle\$\rangle}$ with $2 \leq j \leq l$ and the lemma holds. If the translation function applied next is

(ii) $\delta'(*, \sigma(\overrightarrow{x}), W(y)) \ni \tilde{t}_{1\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([*, Z_{r'}(W(y))], x_{r'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$, and

(iii) $\delta'(*, \sigma(\overrightarrow{x}), Y_1(Y_2(\cdots Y_m(\sharp)\cdots))) \ni \tilde{t}_{1\Delta\langle\$'\rangle}\langle([q_{Z_{1'}}, W_{Z_{1'}}(\sharp)], x_{1'}), \cdots, ([q_{Z_{n'}}, W_{Z_{n'}}(\sharp)], x_{n'})\rangle$,

then we obtain the lemma by a similar way to (i). Thus the lemma holds for the number $k+1$ of $\vdash_{M'}^{(k+1)}$. (End of the proof of Lemma 7)

Under these lemmas and any $\langle t, t'\rangle \in T_\Sigma \times T_\Delta$, $u_i \in (\Gamma)^+$ with $1 \leq i \leq l$,

$t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)]$
$\vdash_M^* t_{\Sigma\langle\$\rangle}[([*, u_1(\sharp)], \tilde{t}_{1\Delta\langle\$'\rangle}), \cdots, ([*, u_l(\sharp)], \tilde{t}_{l\Delta\langle\$'\rangle})]$
$\vdash_M^* ([*, \sharp], t'_{\Delta\langle\$'\rangle})$

if and only if

$([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle})$
$\vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, u_{1'}(\sharp)], t_{1'\Sigma\langle\$\rangle}), \cdots, ([*, u_{l'}(\sharp)], t_{l'\Sigma\langle\$\rangle})\rangle$
$\vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle$,

where for the subtree $t_{i\Sigma\langle\$\rangle}$ $(1 \leq i \leq l)$ of $t_{\Sigma\langle\$\rangle}$, $u_{j'} = u_i$ for $t_{j'\Sigma\langle\$\rangle} = t_{i\Sigma\langle\$\rangle}$ with $1 \leq j \leq l$, $t_{i\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_i)}(\sharp)], \$)] \vdash_M^* ([*, u_i], \tilde{t}_{i\Delta\langle\$'\rangle})$ and $([*, u_i], t_i) \vdash_{M'}^* \tilde{t}_{i\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle$. Thus we obtain

$$M_\emptyset = \{\langle t, t'\rangle \in T_\Sigma \times T_\Delta \mid t_{\Sigma\langle\$\rangle}[([*, S_{(1)}(\sharp)], \$), \cdots, ([*, S_{(l_0)}(\sharp)], \$)] \vdash_M^* ([*, \sharp], t'_{\Delta\langle\$'\rangle})\}$$
$$= \{\langle t, t'\rangle \in T_\Sigma \times T_\Delta \mid ([q_0, W_0(\sharp)], t_{\Sigma\langle\$\rangle}) \vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, S_{(1)}(\sharp)], \$'), \cdots, ([*, S_{(l_0)}(\sharp)], \$')\rangle \vdash_{M'}^* t'_{\Delta\langle\$'\rangle}\langle([*, \sharp_{(1)}], \$'), \cdots, ([*, \sharp_{(l_0)}], \$')\rangle\}$$
$$= M'_\emptyset.$$

Moreover, for the extended lst-PDTT M', the extended lst-PDTT M'' such that $M''_F = M'_\emptyset$ always exists from Theorem 6. Also the lst-PDTT M''' such that $M'''_F = M''_F$ always exists from Theorem 7. Thus the theorem holds. □

On the other hand, we have the next corollary immediately since **nondeleting lt-PDTT**$\not\subseteq$**b-PDTT** and **nondeleting lb-PDTT**$\subset$**b-PDTT** hold from Theorem 10 and Theorem 11.

Corollary. nondeleting lb-PDTT$\subset$**nondeleting lt-PDTT**.

§7. Conclusions

We have introduced a bottom-up pushdown tree transducer (b-PDTT) which results from generalizing a bottom-up tree transducer by adding a pushdown storage (where the pushdown storage have the form of trees, i.e., a tree-pushdown storage) and may be considered as a dual concept of the top-down pushdown tree transducer (t-PDTT). For such a b-PDTT we have shown some fundamental properties such that, the final state translation is equivalent to the empty stack translation, and we can always convert a b-PDTT M into a single state b-PDTT M' which is equivalent to M. Based on the above properties, we have shown that any b-PDTT can always be converted into a linear stack (i.e., a stack symbol is monadic) b-PDTT with single state, and converted into a G-type normal form which corresponds to the Greibach normal form in a context-free grammar (for t-PDTT, no normal form exists which corresponds to the Greibach normal form in a context-free grammar). Finally we have compared b-PDTT with t-PDTT and it is shown that the class of translations generated by nondeleting linear t-PDTTs properly contains the class of translations generated by nondeleting linear b-PDTTs.

References

[1] A.V. Aho, "Indexed Grammars — An Extension of Context-free Grammars," J. ACM, vol.15, no.4, pp.647–671, 1968.

[2] A. V. Aho and J. D. Ullman, The Theory of Parsing, Translation, and Compiling, 1, 2, Prentice-Hall, 1972.

[3] W. Damm and I. Guessarian, " Combining T and level-N," Proc. of the 9th Mathematical Foundations of Computer Since, Lecture Notes in Computer Science, vol.118, pp.262–270, 1981.

[4] W. Damm and I. Guessarian, "Implementation techniques for recursive tree transducers on higher order data types," Report 83–16, Laboratories Informatique Theorique et Programmtion, Universite Paris VII, 1983.

[5] J. Engelfriet, "Bottom-up and top-down transformations – A comparison," Math. Syst.Theory, vol.9, 3, pp.198-231, 1975.

[6] J. Engelfriet and G. Slutzki, "Bounded nesting in macro grammars," Information and Control, vol.42, pp.157–193, 1979.

[7] J. Engelfriet, G. Rozenberg, and G. Slutzki, "Tree transducers, L-systems, and two-way machines," J. Comput.& Syst. Sci., vol.20, pp.150–202, 1980.

[8] J.Engelfriet, "Some open question and recent results on tree transducers and tree languages," in Formal Language Theory; Perspectives and Open Problems, ed. R.V. Book, NY. Academic Press, 1980.

[9] J. Engelfriet, "Context-free grammars with storage," Tech. Report 86–11, University of Leiden, 1986.

[10] J. Engelfriet and H. Vogler, "Pushdown machines for the macro tree transducer," Theoretical Comput. Sci., vol.42, pp.251–369, 1986.

[11] M.J. Fischer, "Grammars with MACRO-like productions," Proc. 9th IEEE symp. on Switching and Automata Theory, pp.131–142, Oct. 1968.

[12] I. Guessarian, "On pushdown tree automata," Proc. of 6th CAAP, Genos, Lecture Notes in Computer Science, pp.211–223, Springer-Verlag, 1981 .

[13] I. Guessarian, "Pushdown tree automata," Math. Syst. Theory, vol.16, pp.237–263, 1983.

[14] M.A. Harrison, Introduction to Formal Language Theory, Addison-Wesley, 1978.

[15] J.E. Hopcroft and J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 1979.

[16] E.T. Irons, "A syntax directed compiler for ALGOL60," Commum. ACM, vol.4, pp.51–55, 1961.

[17] D. Knuth, "Semantics of context-free languages," Math. Syst. Theory, vol.2, 2, pp.127–145, 1968.

[18] W.C. Rounds, "Mappings and grammars on trees,", Math. Syst. Theory, vol.4, 3, pp.257–287, 1970.

[19] K.M. Schimpf & J.H. Gallier, "Tree pushdown automata," J. Comput. & Syst. Sci., vol.30, pp.25–40, 1985.

[20] J.W. Thatcher, "Characterizing derivation trees of context-free grammars through a generalization of finite automata theory," J. Comput. & Syst. Sci., vol.1, pp.317–322, 1967.

[21] J.W. Thatcher, "Generalized^2 sequential machine maps," J. Comput. & Syst. Sci., vol.4, pp.339-367, 1970.

[22] H. Vogler, "Basic tree transducers," J. of Comput. & Syst. Sci., vol.34, pp.87–128, 1987.

[23] K. Yamasaki, "Fundamental properties of pushdown tree automata and context-free tree grammars —An extension of pushdown automata and context-free grammars—," IEICE Trans., vol.J71-D, no.9, pp.1580-1591, Sept. 1988 (in Japanese).

[24] K. Yamasaki, "On relations between pushdown tree automata (PDTA) and indexed grammars," IEICE Trans., vol.J71-D, no.12, pp.2485–2497, Dec. 1988 (in Japanese).

[25] K. Yamasaki, "Fundamental properties of bottom-up pushdown tree automata," Systems and Computers in Japan (WILEY), vol.21, No.14, pp.11–22, 1990.

[26] K. Yamasaki, "On standard form of bottom-up pushdown tree automata," IEICE Technical Report, COMP90-43, Sept. 1990 (in Japanese).

[27] K. Yamasaki, "A comparison of acceptance capabilities of bottom-up pushdown tree automata and top-down pushdown tree automata," Trans. IPSJ, vol.32, no.9, pp.1035–1045, Sept. 1991 (in Japanese).

[28] K. Yamasaki, "Fundamental properties of pushdown tree transducer(PDTT) — A Top-down Case," IEICE Trans. Inf. & Syst., vol. E76-D, No.10, Oct., pp.1234–1242, 1993.

[29] K. Yamasaki and Y.Sodeshima, "A comparison of bottom-up pushdown tree transducer and top-down pushdown tree transducer," IEICE Technical Report, COMP97-78-86, Jan. 1998.

Katsunori Yamasaki and Yoshichika Sodeshima
Department of Information Sciences, Faculty of Science and Technology, Science University of Tokyo
2641 Yamazaki, Noda, Chiba 278-8510, Japan