Theoretical and Applied Informatics
ISSN 1896–5334
Vol.24 (2012), no. 3
pp. 175–190
DOI: 10.2478/v10179-012-0011-6

Synchronous Versions of Regulated Grammars: Generative Power and Linguistic Applications

PETR HORÁČEK, ALEXANDER MEDUNA

Formal Model Research Group, Department of Information Systems,
Faculty of Information Technology, Brno University of Technology,
Božetěchova 1/2, 612 66 Brno, Czech Republic
ihoracekp@fit.vutbr.cz, meduna@fit.vutbr.cz

Received 20 June 2012, Revised 30 August 2012, Accepted 18 September 2012.

Abstract: This paper introduces the notion of new synchronous grammars as systems consisting of two context-free grammars with linked rules instead of linked nonterminals. Further, synchronous versions of regulated grammars, specifically, matrix grammars and scattered context grammars, are discussed. From a theoretical point of view, this paper discusses the power of these synchronous grammars. It demonstrates the following main results. First, if we synchronize context-free grammars by linking rules, the grammar generates the languages defined by matrix grammars. Second, if we synchronize matrix grammars by linking matrices, the generative power remains unchanged. Third, synchronous scattered context grammars generate the class of recursively enumerable languages. From a more practical viewpoint, this paper presents linguistic application prospects. The focus is on natural language translation between Japanese and English.

Keywords: regulated grammar, synchronous grammar, generative power, natural language translation

1. Introduction

Language-translation devices fulfill a significant role in the theory of formal languages. Apart from their theoretically oriented investigation, they are frequently applied in a variety of computer science fields, such as natural language translation. The present paper contributes to the study of synchronous versions of these devices.

To express the subject of this paper as well as its achieved results in a greater detail, recall there exists a generalization of context-free grammar (CFG) called synchronous CFG (see [3]), which strongly resembles syntax-directed transduction grammar [8] or syntax-directed translation scheme [1]. In essence, the synchronous CFG is a modification of CFG where every rule has two right-hand sides. In this way, the synchronous

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

CFGs represent translation-defining devices rather than language-defining devices as CFGs are traditionally used.

In [6], we have proposed synchronization based on linked rules instead of nonterminals. We have introduced synchronous matrix grammars and synchronous scattered context grammars. In this paper, we continue with this topic by discussing some theoretical properties of these grammars. Specifically, we investigate their generative power and achieve the following three main results.

I. First, if we synchronize CFGs by linking rules as proposed and defined in this paper, we obtain the generative power coinciding with the power of matrix grammars. Consequently, we significantly increase the power of CFGs in this way because the traditional synchronous CFGs (with linked nonterminals) only generate the family of context-free languages.

II. Second, we show that by synchronizing matrix grammars so we link matrices, we obtain no increase in power. That is, synchronous matrix grammars have the same generative power as matrix grammars.

III. Finally, we study synchronous scattered context grammars (with linked rules). We show that the class of languages defined by synchronous scattered context grammars equals the class of recursively enumerable languages.

In the conclusion of this paper, we sketch application perspectives of grammars discussed in this paper. We focus on Japanese-English translation and present examples illustrating the key principles and advantages of the new grammars. By extending the principle of synchronization beyond CFGs, we are able to describe certain syntactic structures and relations more easily. For example, when translating from Japanese to English, we need to deal with different handling of inflection. The translation is even more complicated by some important structural differences between the two languages.

2. Preliminaries

We assume that the reader is familiar with the basic aspects of modern formal language theory (see [12], [9]) and computational linguistics (see [11]). In this section, we review only selected definitions that are of key importance to the topics discussed in this paper. Further information about matrix grammars and scattered context grammars can be found in [4] and [10], respectively.

Definition 1 (Context-free grammar)**.** *A* context-free grammar *(CFG) G is a quadruple $G = (N, T, P, S)$, where N is a finite set of nonterminals, T is a finite set of terminals, $N \cap T = \emptyset$, $P \subset N \times (N \cup T)^*$ is a finite set of rules, $(u, v) \in P$ is written as $u \to v$, and $S \in N$ is the start symbol.*

Definition 2 (Derivation)**.** *Let G be a CFG. Let $u, v \in (N \cup T)^*$ and $p = A \to x \in P$. Then, we say that uAv* directly derives *uxv according to p in G, written as $uAv \Rightarrow_G$*

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

$uXv\,[p]$ or simply $uAv \Rightarrow uxv$. We further define $\Rightarrow^+$ as the transitive closure of $\Rightarrow$ and $\Rightarrow^*$ as the transitive and reflexive closure of $\Rightarrow$.

Definition 3 (Generated language)**.** *Let G be a CFG. The* language generated by *G, denoted by $L(G)$, is defined as $L(G) = \{w\colon w \in T^*, S \Rightarrow^* w\}$.*

Definition 4 (Matrix grammar)**.** *A* matrix grammar *H is a pair $H = (G, M)$, where $G = (N, T, P, S)$ is a CFG and $M \subset P^*$ is a finite language over P. Members of M are called* matrices*.*

Definition 5 (Derivation in matrix grammar)**.** *Let $H = (G, M)$ be a matrix grammar, $G = (N, T, P, S)$. Then, for $u, v \in (N \cup T)^*$, $m \in M$ we define $u \Rightarrow v\,[m]$ in H, if there are strings $x_0, \dots, x_n$ such that $u = x_0$, $v = x_n$ and $x_0 \Rightarrow x_1\,[p_1] \Rightarrow x_2\,[p_2] \Rightarrow \dots \Rightarrow x_n\,[p_n]$ in G, and $m = p_1 \dots p_n$.*

Definition 6 (Scattered context grammar)**.** *A* scattered context grammar *(SCG) G is a quadruple $G = (N, T, P, S)$, where N is a finite set of nonterminals, T is a finite set of terminals, $N \cap T = \emptyset$, P is a finite set of rules of the form $(A_1, \dots, A_n) \to (x_1, \dots, x_n)$, where $A_1, \dots, A_n \in N$, $x_1, \dots, x_n \in (N \cup T)^*$, and $S \in N$ is the start symbol.*

Definition 7 (Derivation in SCG)**.** *Let $G = (N, T, P, S)$ be an SCG. For $u, v \in (N \cup T)^*$, $p \in P$ we define $u \Rightarrow v\,[p]$, if there is a factorization of $u = u_1 A_1 \dots u_n A_n u_{n+1}$, $v = u_1 x_1 \dots u_n x_n u_{n+1}$ such that $p = (A_1, \dots, A_n) \to (x_1, \dots, x_n)$ and $u_i \in (N \cup T)^*$ for $1 \le i \le n$.*

3. Synchronization and Regulated Rewriting

This section recalls basic definitions from [6]. First, we define rule-based synchronization for CFGs.

Definition 8 (Rule-synchronized CFG)**.** *A* rule-synchronized CFG *(RSCFG) H is a 5-tuple $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$, where $G_I = (N_I, T_I, P_I, S_I)$ and $G_O = (N_O, T_O, P_O, S_O)$ are CFGs, Ψ is a set of* rule labels*, and φ_I is a function from Ψ to P_I and φ_O is a function from Ψ to P_O.*

We use the following notation (presented for input grammar G_I, analogous for output grammar G_O):

$p : A_I \to x_I$ where $p \in \Psi, A_I \to x_I \in P_I$	$\varphi_I(p) = A_I \to x_I$
$x_I \Rightarrow_{G_I} y_I\,[p]$ where $x_I, y_I \in (N \cup T)^*, p \in \Psi$	derivation step in G_I applying rule $\varphi_I(p)$
$x_I \Rightarrow^n_{G_I} y_I\,[p_1 \dots p_n]$ where $x_I, y_I \in (N \cup T)^*, p_i \in \Psi$ for $1 \le i \le n$	derivation in G_I applying rules $\varphi_I(p_1) \dots \varphi_I(p_n)$

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

Definition 9 (Translation in RSCFG)**.** *Let* $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$ *be a RSCFG.* Translation *defined by* H*,* $T(H)$*, is the set of pairs of sentences, which is defined as* $T(H) = \{(w_I, w_O)\colon w_I \in T_I^*, w_O \in T_O^*, S_I \Rightarrow_{G_I}^* w_I\,[\alpha], S_O \Rightarrow_{G_O}^* w_O\,[\alpha], \alpha \in \Psi^*\}$.

In [6], we considered RSCFG only as a variant of synchronous CFG. However, there is in fact a significant difference. While the latter does not increase the generative power over CFG, RSCFG does as we prove in the next section.

Next, to define synchronization for SCGs, we simply replace context-free rules with scattered context rules.

Definition 10 (Synchronous SCG)**.** *A* synchronous SCG *(SSCG)* H *is a 5-tuple* $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$*, where* $G_I = (N_I, T_I, P_I, S_I)$ *and* $G_O = (N_O, T_O, P_O, S_O)$ *are SCGs,* Ψ *is a set of* rule labels*, and* φ_I *is a function from* Ψ *to* P_I *and* φ_O *is a function from* Ψ *to* P_O.

Definition 11 (Translation in SSCG)**.** *Let* $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$ *be a SSCG.* Translation *defined by* H*,* $T(H)$*, is the set of pairs of sentences, which is defined as* $T(H) = \{(w_I, w_O)\colon w_I \in T_I^*, w_O \in T_O^*, S_I \Rightarrow_{G_I}^* w_I\,[\alpha], S_O \Rightarrow_{G_O}^* w_O\,[\alpha], \alpha \in \Psi^*\}$.

Finally, in matrix grammars, we link whole matrices rather than individual rules.

Definition 12 (Synchronous matrix grammar)**.** *A* synchronous matrix grammar *(SMAT)* H *is a 7-tuple* $H = (G_I, M_I, G_O, M_O, \Psi, \varphi_I, \varphi_O)$*, where* (G_I, M_I) *and* (G_O, M_O) *are matrix grammars,* Ψ *is a set of* matrix labels*, and* φ_I *is a function from* Ψ *to* M_I *and* φ_O *is a function from* Ψ *to* M_O.

Definition 13 (Translation in SMAT)**.** *Let* $H = (G_I, M_I, G_O, M_O, \Psi, \varphi_I, \varphi_O)$ *be a SMAT.* Translation *defined by* H*,* $T(H)$*, is the set of pairs of sentences, which is defined as* $T(H) = \{(w_I, w_O)\colon w_I \in T_I^*, w_O \in T_O^*, S_I \Rightarrow_{(G_I,M_I)}^* w_I\,[\alpha], S_O \Rightarrow_{(G_O,M_O)}^* w_O\,[\alpha], \alpha \in \Psi^*\}$.

4. Generative Power of Synchronous Grammars

Synchronous grammars define translation – that is, pairs of sentences. To be able to compare their generative power with well-known models such as CFGs, we can consider their input or output language.

Definition 14 (Input and output language)**.** *Let* H *be a synchronous grammar. Then, we define*
I. the input language *of* H*,* $L_I(H)$*, as* $L_I(H) = \{w_I \in T_I^*\colon (w_I, w_O) \in T(H)\}$*,*
II. the output language *of* H*,* $L_O(H)$*, as* $L_O(H) = \{w_O \in T_O^*\colon (w_I, w_O) \in T(H)\}$.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

Example 1. Consider a RSCFG $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$ with the following rules (nonterminals are in capitals, linked rules share the same label):

G_I				G_O			
1:	S_I	$\rightarrow$	ABC	1:	S_O	$\rightarrow$	A
2:	A	$\rightarrow$	aA	2:	A	$\rightarrow$	B
3:	B	$\rightarrow$	bB	3:	B	$\rightarrow$	C
4:	C	$\rightarrow$	cC	4:	C	$\rightarrow$	A
5:	A	$\rightarrow$	ε	5:	A	$\rightarrow$	B'
6:	B	$\rightarrow$	ε	6:	B'	$\rightarrow$	C'
7:	C	$\rightarrow$	ε	7:	C'	$\rightarrow$	ε

An example of a derivation follows next.

$$
\begin{array}{llll}
S_I & \Rightarrow & ABC & [1] \\
 & \Rightarrow & aABC & [2] \\
 & \Rightarrow & aAbBC & [3] \\
 & \Rightarrow & aAbBcC & [4] \\
 & \Rightarrow & aaAbBcC & [2] \\
 & \Rightarrow & aaAbbBcC & [3] \\
 & \Rightarrow & aaAbbBccC & [4] \\
 & \Rightarrow & aabbBccC & [5] \\
 & \Rightarrow & aabbccC & [6] \\
 & \Rightarrow & aabbcc & [7]
\end{array}
\qquad
\begin{array}{llll}
S_O & \Rightarrow & A & [1] \\
 & \Rightarrow & B & [2] \\
 & \Rightarrow & C & [3] \\
 & \Rightarrow & A & [4] \\
 & \Rightarrow & B & [2] \\
 & \Rightarrow & C & [3] \\
 & \Rightarrow & A & [4] \\
 & \Rightarrow & B' & [5] \\
 & \Rightarrow & C' & [6] \\
 & \Rightarrow & \varepsilon & [7]
\end{array}
$$

We can easily see that $L_I(H) = \{a^n b^n c^n : n \geq 0\}$, which is well known not to be a context-free language. This shows that RSCFGs are stronger than (synchronous) CFGs. Where exactly do synchronous grammars (rule-synchronized) stand in terms of generative power?

Let $\mathscr{L}(RSCFG)$, $\mathscr{L}(SMAT)$, and $\mathscr{L}(SSCG)$ denote the class of languages generated by RSCFGs, SMATs, and SSCGs, respectively, as their input language. Note that the results presented below would be the same if we considered the output language instead.

Definition 15. *Let $G = (N, T, P, S)$ be a CFG. Then, we define the function Δ over $(N \cup T)^*$ as follows:*
I. for all $w \in T^*$, $\Delta(w) = \varepsilon$
II. for all $w = x_0 A_1 x_2 A_2 \ldots x_{n-1} A_n x_n$, $x_i \in T^*$, $0 \leq i \leq n$, $A_j \in N$, $1 \leq j \leq n$ for some $n \geq 1$, $\Delta(w) = A_1 A_2 \ldots A_n$.

In this paper, we demonstrate the following main results.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

Theorem 1.

$$\mathscr{L}(RSCFG) = \mathscr{L}(MAT)$$

Proof. I. First, we prove that $\mathscr{L}(RSCFG) \subseteq \mathscr{L}(MAT)$. For every RSCFG $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$, where $G_I = (N_I, T_I, P_I, S_I)$, $G_O = (N_O, T_O, P_O, S_O)$, we can construct a matrix grammar $H' = (G, M)$, where $G = (N, T, P, S)$, such that $L(H') = L_I(H)$, as follows. Without loss of generality, assume $N_I \cap N_O = \emptyset$, $S \notin N_I \cup N_O$. Set $N = N_I \cup N_O \cup \{S\}$, $T = T_I$, $P = \{S \rightarrow S_I S_O\}$, $M = \{S \rightarrow S_I S_O\}$. For every label $p \in \Psi$, add rules p_I, p_O to P, add matrix $p_I p_O$ to M, where $p_I = \varphi_I(p)$ and $p_O = A \rightarrow x$ such that $\varphi_O(p) = A \rightarrow x'$, $x = \Delta(x')$.[1]

Basic idea. H' simulates the principle of linked rules in H by matrices. That is, for every pair of rules $(A_I \rightarrow x_I, A_O \rightarrow x_O)$ such that $\varphi_I(p) = A_I \rightarrow x_I, \varphi_O(p) = A_O \rightarrow x_O$ for some $p \in \Psi$ in H, there is a matrix $m = A_I \rightarrow x_I A_O \rightarrow \Delta(x_O)$ in H'. If, in H, $x_I \Rightarrow y_I\,[p]$ in G_I and $x_O \Rightarrow y_O\,[p]$ in G_O, then there is a derivation step $x_I\,\Delta(x_O) \Rightarrow y_I\,\Delta(y_O)\,[m]$ in H'. Note that since the rules are context-free, the presence (or absence) of terminals in a sentential form does not affect which rules we can apply. Furthermore, because the nonterminal sets N_I and N_O are disjoint, the sentential form in H' always consists of two distinct parts such that the first part corresponds to the derivation in G_I and the second part to the derivation in G_O.

Claim 2. *If $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H for some $\alpha \in \Psi^*$, then $S \Rightarrow^* w_I\,\Delta(w_O)$ in H'.*

Proof of Claim 2. By induction on the number of derivation steps in H.

Basis. Let $S_I \Rightarrow^0 S_I\,[\varepsilon]$, $S_O \Rightarrow^0 S_O\,[\varepsilon]$ in H. Then, $S \Rightarrow S_I S_O\,[p]$ in G, $p \in M$, and thus $S \Rightarrow S_I S_O\,[p]$ in H'. Claim 2 holds for zero derivation steps in H.

Induction hypotesis. Suppose that Claim 2 holds for j or fewer derivation steps in H.

Induction step. Let $S_I \Rightarrow^j w_I\,[\alpha] \Rightarrow w'_I\,[p]$, $S_O \Rightarrow^j w_O\,[\alpha] \Rightarrow w'_O\,[p]$ in H. Then, by the induction hypotesis, $S \Rightarrow^* w_I\,\Delta(w_O)$ in H'. Without loss of generality, suppose that $w_I = u_I A_I v_I$, $w_O = u_O A_O v_O$, $w'_I = u_I x_I v_I$, $w'_O = u_O x_O v_O$, where $A_I \in N_I$, $A_O \in N_O$, $u_I, v_I, x_I \in (N_I \cup T_I)^*$, and $u_O, v_O, x_O \in (N_O \cup T_O)^*$. That is, $\varphi_I(p) = A_I \rightarrow x_I$, $\varphi_O(p) = A_O \rightarrow x_O$. From the construction of H', we know that $p_I = A_I \rightarrow x_I \in P$, $p_O = A_O \rightarrow \Delta(x_O) \in P$, and $p_I p_O \in M$. Therefore, in H', $S \Rightarrow^* w_I\,\Delta(w_O) = u_I A_I v_I\,\Delta(u_O A_O v_O) \Rightarrow u_I x_I v_I\,\Delta(u_O x_O v_O)\,[p_I p_O] = w_I\,\Delta(w_O)$. Claim 2 holds. Furthermore, if $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H, where $w_I \in T_I^*$, $w_O \in T_O^*$, then $\Delta(w_O) = \varepsilon$, and thus $S \Rightarrow^* w_I$ in H'. $L_I(H) \subseteq L(H')$. $\square$

[1]This removes all terminals from the right-hand side of the rule. Note that if we left the rule unchanged, we would obtain the concatenation of the input and the output sentence. Further, if we wanted $L(H') = L_O(H)$, we would simply modify p_I instead of p_O.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

Claim 3. *If $S \Rightarrow^* w$ in H', then there are strings w_I, w_O such that $w = w_I\,\Delta(w_O)$ and $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H for some $\alpha \in \Psi^*$.*

Proof of Claim 3. By induction on the number of derivation steps in H'.

Basis. Consider a single derivation step in H'. Because $S \to S_I S_O$ is the only rule in P with S as its left-hand side, this must be $S \Rightarrow S_I S_O$. Then, $S_I \Rightarrow^0 S_I\,[\varepsilon]$, $S_O \Rightarrow^0 S_O\,[\varepsilon]$ in H. Claim 3 holds for one derivation step in H'.

Induction hypotesis. Suppose that Claim 3 holds for j or fewer derivation steps in H'.

Induction step. Let $S \Rightarrow S_I S_O \Rightarrow^{j-1} w \Rightarrow w'\,[m]$ in H'. Then, by the induction hypotesis, $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H for some w_I, w_O such that $w = w_I\,\Delta(w_O)$. From the construction of H', we know that $m = p_I p_O$, where for some $p \in \Psi$, $p_I = A_I \to x_I = \varphi_I(p)$ and $p_O = A_O \to \Delta(x_O)$ such that $A_O \to x_O = \varphi_O(p)$. Therefore, there must be a factorization of w and w' such that $w = w_I\,\Delta(w_O) = u_I A_I v_I\,\Delta(u_O A_O v_O)$, $w' = w'_I\,\Delta(w'_O) = u_I x_I v_I\,\Delta(u_O x_O v_O)$ where $A_I \in N_I$, $A_O \in N_O$, $u_I, v_I, x_I \in (N_I \cup T_I)^*$, and $u_O, v_O, x_O \in (N_O \cup T_O)^*$. Therefore, $S_I \Rightarrow^* w_I\,[\alpha] = u_I A_I v_I \Rightarrow u_I x_I v_I\,[p] = w'_I$ and $S_O \Rightarrow^* w_O\,[\alpha] = u_O A_O v_O \Rightarrow u_O x_O v_O\,[p] = w'_O$ in H. Claim 3 holds. Furthermore, if $S \Rightarrow^* w$ in H', where $w \in T^*$, then $w_I \in T_I^*$ and $\Delta(w_O) = \varepsilon$, thus $w_O \in T_O^*$. $L(H') \subseteq L_I(H)$. $\square$

II. Now we have to show that $\mathscr{L}(MAT) \subseteq \mathscr{L}(RSCFG)$ holds. For every matrix grammar $H = (G, M)$, where $G = (N, T, P, S)$, we can construct a RSCFG $H' = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$, where $G_I = (N_I, T_I, P_I, S_I)$, $G_O = (N_O, T_O, P_O, S_O)$, such that $L_I(H') = L(H)$, as follows. Without loss of generality, assume $N \cap \{S_I, S_O, X\} = \emptyset$. Set $N_I = N \cup \{S_I, X\}$, $T_I = T$, $P_I = \{S_I \to SX, X \to \varepsilon\}$, $N_O = \{S_O, X\}$, $T_O = \emptyset$, $P_O = \{S_O \to X, X \to \varepsilon\}$. Set $\Psi = \{0, 1\}$, $\varphi_I = \emptyset$, $\varphi_O = \emptyset$, $\varphi_I(0) = S_I \to SX$, $\varphi_O(0) = S_O \to X$, $\varphi_I(1) = X \to \varepsilon$, $\varphi_O(1) = X \to \varepsilon$. For every matrix $m = p \in M$, where $p \in P$, add rule p to P_I, $X \to X$ to P_O, add label $\langle m \rangle$ to Ψ, and set $\varphi_I(\langle m \rangle) = p$, $\varphi_O(\langle m \rangle) = X \to X$. For every matrix $m = p_1 \ldots p_n \in M$, where $n > 1$, add rules $p_1, \ldots, p_n$ to P_I, add new nonterminals $\langle Xm \rangle_1, \ldots, \langle Xm \rangle_{n-1}$ to N_O, add rules $X \to \langle Xm \rangle_1, \langle Xm \rangle_1 \to \langle Xm \rangle_2, \ldots, \langle Xm \rangle_{n-2} \to \langle Xm \rangle_{n-1}, \langle Xm \rangle_{n-1} \to X$ to P_O, add new labels $\langle m \rangle_1, \ldots, \langle m \rangle_n$ to Ψ, and set $\varphi_I(\langle m \rangle_1) = p_1$, $\varphi_O(\langle m \rangle_1) = X \to \langle Xm \rangle_1$, $\varphi_I(\langle m \rangle_i) = p_i$, $\varphi_O(\langle m \rangle_i) = \langle Xm \rangle_{i-1} \to \langle Xm \rangle_i$ for $1 < i < n$, and $\varphi_I(\langle m \rangle_n) = p_n$, $\varphi_O(\langle m \rangle_n) = \langle Xm \rangle_{n-1} \to X$.

Basic idea. H' simulates matrices in H by derivation in G_O. That is, if $x \Rightarrow y\,[m]$ in H, where $m = p_1 \ldots p_n$ for some n, then there is a sequence of derivation steps $X \Rightarrow \langle Xm \rangle_1\,[\langle m \rangle_1] \Rightarrow \langle Xm \rangle_2\,[\langle m \rangle_2] \Rightarrow \ldots \Rightarrow \langle Xm \rangle_{n-2}\,[\langle m \rangle_{n-2}] \Rightarrow \langle Xm \rangle_{n-1}\,[\langle m \rangle_{n-1}] \Rightarrow X\,[\langle m \rangle_n]$ in G_O and $\varphi_I(\langle m \rangle_i) = p_i$ for $1 \le i \le n$. Now observe that in G_O constructed by the above algorithm, every nonterminal except X can only appear as the left-hand

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

side of no more than one rule. This means that after rewriting X to $\langle Xm\rangle_1$, the only way for the derivation to proceed is the above sequence, and the entire matrix is simulated.

Formally, if $x \Rightarrow y\,[m]$ in H, where $m = p_1 \dots p_n$ for some n, then, in H', $x \Rightarrow^n y\,[m_1 \dots m_n]$ in G_I and $X \Rightarrow^n X\,[m_1 \dots m_n]$ in G_O. Conversely, if, in H', $x \Rightarrow^n y\,[\alpha]$ in G_I and $X \Rightarrow^n X\,[\alpha]$ in G_O for some $n \geq 1$, $\alpha \in \Psi^*$, and there is no $k < n$ such that $x \Rightarrow^k z\,[\beta] \Rightarrow^* y\,[\gamma]$ in G_I and $X \Rightarrow^k X\,[\beta] \Rightarrow^* X\,[\gamma]$ in G_O for some $\beta, \gamma \in \Psi^*$, there has to be $m \in M$ such that $x \Rightarrow y\,[m]$ in H. Therefore, if for some $\alpha \in \Psi^*$, $S_I \Rightarrow^* wX\,[\alpha] \Rightarrow w\,[1]$, $S_O \Rightarrow^* X\,[\alpha] \Rightarrow \varepsilon\,[1]$ in H', where $w \in T_I^*$, then $S \Rightarrow^* w$ in H. On the other hand, if $S \Rightarrow^* w$ in H, where $w \in T^*$, then $\alpha \in \Psi^*$, $S_I \Rightarrow^* wX\,[\alpha] \Rightarrow w\,[1]$, $S_O \Rightarrow^* X\,[\alpha] \Rightarrow \varepsilon\,[1]$ in H' for some $\alpha \in \Psi^*$. $L_I(H') = L(H)$.

Theorem 1 holds. □

Note that G_O constructed by the above algorithm is not only context-free, but also regular.

Theorem 4.

$$\mathscr{L}(SMAT) = \mathscr{L}(MAT)$$

Proof. The inclusion $\mathscr{L}(MAT) \subseteq \mathscr{L}(SMAT)$ follows from definition. It only remains to prove that $\mathscr{L}(SMAT) \subseteq \mathscr{L}(MAT)$. For every SMAT $H = (G_I, M_I, G_O, M_O, \Psi, \varphi_I, \varphi_O)$, where $G_I = (N_I, T_I, P_I, S_I)$, $G_O = (N_O, T_O, P_O, S_O)$, we can construct a matrix grammar $H' = (G, M)$, where $G = (N, T, P, S)$, such that $L(H') = L_I(H)$, as follows. Without loss of generality, assume $N_I \cap N_O = \emptyset$, $S \notin N_I \cup N_O$. Set $N = N_I \cup N_O \cup \{S\}$, $T = T_I$, $P = \{S \to S_I S_O\}$, $M = \{S \to S_I S_O\}$. For every label $p \in \Psi$, add rules $p_{I1}, \dots, p_{In}, p_{O1}, \dots, p_{Om}$ to P, add matrix $p_{I1} \dots p_{In} p_{O1} \dots p_{Om}$ to M, where $p_{I1} \dots p_{In} = \varphi_I(p)$ and for $1 \leq j \leq m$, $p_{Oj} = A_j \to x_j$ such that $\varphi_O(p)[j] = A_j \to x'_j$, $x_j = \Delta(x'_j)$.[2]

Basic idea. H' simulates H by combining the rules of each two linked matrices in H into a single matrix in H'. That is, for every pair of matrices (m_I, m_O) such that $m_I = \varphi_I(p), m_O = \varphi_O(p)$ for some $p \in \Psi$ in H, there is a matrix $m = m_I m'_O$ in H', where m'_O is equal to m_O with all terminals removed (formally defined above). If, in H, $x_I \Rightarrow y_I\,[p]$ in G_I and $x_O \Rightarrow y_O\,[p]$ in G_O, then there is a derivation step $x_I\,\Delta(x_O) \Rightarrow y_I\,\Delta(y_O)\,[m]$ in H'. Note that since the rules are context-free, the presence (or absence) of terminals in a sentential form does not affect which rules we can apply. Furthermore, because the nonterminal sets N_I and N_O are disjoint, the sentential form in H' always consists of two distinct parts such that the first part corresponds to the derivation in G_I and the second part to the derivation in G_O.

[2] Again, this removes all terminals from the right-hand side of the rules (see Theorem 1). $m[j]$ denotes the j-th rule in matrix m.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

Claim 5. *If $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H for some $\alpha \in \Psi^*$, then $S \Rightarrow^* w_I\,\Delta(w_O)$ in H'.*

Proof of Claim 5. By induction on the number of derivation steps in H.

Basis. Let $S_I \Rightarrow^0 S_I\,[\varepsilon]$, $S_O \Rightarrow^0 S_O\,[\varepsilon]$ in H. Then, $S \Rightarrow S_I S_O\,[p]$ in G, $p \in M$, and thus $S \Rightarrow S_I S_O\,[p]$ in H'. Claim 5 holds for zero derivation steps in H.

Induction hypotesis. Suppose that Claim 5 holds for j or fewer derivation steps in H.

Induction step. Let $S_I \Rightarrow^j w_I\,[\alpha] \Rightarrow w'_I\,[p]$, $S_O \Rightarrow^j w_O\,[\alpha] \Rightarrow w'_O\,[p]$ in H. Then, by the induction hypotesis, $S \Rightarrow^* w_I\,\Delta(w_O)$ in H'. Furthermore, if $w_I \Rightarrow w'_I\,[p]$ in (G_I, M_I), $w_O \Rightarrow w'_O\,[p]$ in (G_O, M_O), where $\varphi_I(p) = p_{I1} \dots p_{In}$ for some n, $\varphi_O(p) = p_{O1} \dots p_{Om}$ for some m, then, in G_I, $w_I \Rightarrow w_{I1}\,[p_{I1}] \Rightarrow \dots \Rightarrow w_{In}\,[p_{In}] = w'_I$, and, in G_O, $w_O \Rightarrow w_{O1}\,[p_{O1}] \Rightarrow \dots \Rightarrow w_{On}\,[p_{Om}] = w'_O$. Without loss of generality, suppose that $w_I = u_{I1}A_{I1}v_{I1} \Rightarrow u_{I1}x_{I1}v_{I1}\,[p_{I1}] = u_{I2}A_{I2}v_{I2} \Rightarrow \dots \Rightarrow u_{In}x_{In}v_{In}\,[p_{In}] = w'_I$, $w_O = u_{O1}A_{O0}v_{O1} \Rightarrow u_{O1}x_{O1}v_{O1}\,[p_{O1}] = u_{O2}A_{O2}v_{O2} \Rightarrow \dots \Rightarrow u_{Om}x_{Om}v_{Om}\,[p_{Om}] = w'_O$, where for $1 \le i \le n$, $A_{Ii} \in N_I$, $u_{Ii}, v_{Ii}, x_{Ii} \in (N_I \cup T_I)^*$, and for $1 \le j \le m$, $A_{Oj} \in N_O$, $u_{Oj}, v_{Oj}, x_{Oj} \in (N_O \cup T_O)^*$. That is, $\varphi_I(p) = A_{I1} \to x_{I1} \dots A_{In} \to x_{In}$, $\varphi_O(p) = A_{O1} \to x_{O1} \dots A_{Om} \to x_{Om}$. From the construction of H', we know that for $1 \le i \le n$, $p_{Ii} = A_{Ii} \to x_{Ii} \in P$, for $1 \le j \le m$, $p'_{Oj} = A_{Oj} \to \Delta(x_{Oj}) \in P$, and $t = p_{I1} \dots p_{In} p'_{O1} \dots p'_{Om} \in M$. Therefore, in G, $S \Rightarrow^* w_I\,\Delta(w_O) = u_{I1}A_{I1}v_{I1}\,\Delta(w_O) \Rightarrow u_{I1}x_{I1}v_{I1}\,\Delta(w_O)\,[p_{I1}] = u_{I2}A_{I2}v_{I2}\,\Delta(w_O) \Rightarrow \dots \Rightarrow u_{In}x_{In}v_{In}\,\Delta(w_O)\,[p_{In}] = w'_I\,\Delta\,w_O = w'_I\,\Delta(u_{O1}A_{O1}v_{O1}) \Rightarrow w'_I\,\Delta(u_{O1}x_{O1}v_{O1})\,[p'_{O1}] = w'_I\,\Delta(u_{O2}A_{O2}v_{O2}) \Rightarrow \dots \Rightarrow w'_I\,\Delta(u_{Om}x_{Om}v_{Om})\,[p'_{Om}] = w'_I\,\Delta(w'_O)$, and thus in H', $S \Rightarrow^* w_I\,\Delta\,w_O \Rightarrow w'_I\,\Delta(w'_O)\,[t]$. Claim 5 holds. Furthermore, if $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H, where $w_I \in T_I^*$, $w_O \in T_O^*$, then $\Delta(w_O) = \varepsilon$, and thus $S \Rightarrow^* w_I$ in H'. $L_I(H) \subseteq L(H')$. □

Claim 6. *If $S \Rightarrow^* w$ in H', then there are strings w_I, w_O such that $w = w_I\,\Delta(w_O)$ and $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H for some $\alpha \in \Psi^*$.*

Proof of Claim 6. By induction on the number of derivation steps in H'.

Basis. Consider a single derivation step in H'. Because $S \to S_I S_O$ is the only rule in P with S as its left-hand side, this must be $S \Rightarrow S_I S_O$. Then, $S_I \Rightarrow^0 S_I\,[\varepsilon]$, $S_O \Rightarrow^0 S_O\,[\varepsilon]$ in H. Claim 6 holds for one derivation step in H'.

Induction hypotesis. Suppose that Claim 6 holds for j or fewer derivation steps in H'.

Induction step. Let $S \Rightarrow S_I S_O \Rightarrow^{j-1} w \Rightarrow w'\,[t]$ in H'. Then, by the induction hypotesis, $S_I \Rightarrow^* w_I\,[\alpha]$, $S_O \Rightarrow^* w_O\,[\alpha]$ in H for some w_I, w_O such that $w = w_I\,\Delta(w_O)$. From the construction of H', we know that $t = p_{I1} \dots p_{In} p'_{O1} \dots p'_{Om}$, where for some $p \in \Psi$, $p_{I1} = A_{I1} \to x_{I1} \dots p_{In} = A_{In} \to x_{In} = \varphi_I(p)$ and $p'_{O1} = A_{O1} \to \Delta(x_O)_1 \dots p'_{On} = A_{Om} \to \Delta(x_{Om})$ such that $p_{O1} = A_{O1} \to$

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

$x_{O1} \dots p_{Om} = A_{Om} \rightarrow x_{Om} = \varphi_O(p)$. Then, if $S \Rightarrow S_I S_O \Rightarrow^{j-1} w \Rightarrow w' \, [t]$ in H', $S \Rightarrow^* w = w_I \, \Delta(w_O) = u_{I1} A_{I1} v_{I1} \, \Delta(w_O) \Rightarrow u_{I1} x_{I1} v_{I1} \, \Delta(w_O) \, [p_{I1}] = u_{I2} A_{I2} v_{I2} \, \Delta(w_O) \Rightarrow \dots \Rightarrow u_{In} x_{In} v_{In} \, \Delta(w_O) \, [p_{In}] = w'_I \, \Delta \, w_O = w'_I \, \Delta(u_{O1} A_{O1} v_{O1}) \Rightarrow w'_I \, \Delta(u_{O1} x_{O1} v_{O1}) \, [p'_{O1}] = w'_I \, \Delta(u_{O2} A_{O2} v_{O2}) \Rightarrow \dots \Rightarrow w'_I \, \Delta(u_{Om} x_{Om} v_{Om}) \, [p'_{Om}] = w'_I \, \Delta(w'_O)$ in G, where for $1 \leq i \leq n$, $A_{Ii} \in N_I$, $u_{Ii}, v_{Ii}, x_{Ii} \in (N_I \cup T_I)^*$, and for $1 \leq j \leq m$, $A_{Oj} \in N_O$, $u_{Oj}, v_{Oj}, x_{Oj} \in (N_O \cup T_O)^*$. Therefore, $w_I = u_{I1} A_{I1} v_{I1} \Rightarrow u_{I1} x_{I1} v_{I1} \, [p_{I1}] = u_{I2} A_{I2} v_{I2} \Rightarrow \dots \Rightarrow u_{In} x_{In} v_{In} \, [p_{In}] = w'_I$ in G_I and $w_O = u_{O1} A_{O0} v_{O1} \Rightarrow u_{O1} x_{O1} v_{O1} \, [p_{O1}] = u_{O2} A_{O2} v_{O2} \Rightarrow \dots \Rightarrow u_{Om} x_{Om} v_{Om} \, [p_{Om}] = w'_O$ in G_O, and thus $S_I \Rightarrow^* w_I \, [\alpha] = u_I A_I v_I \Rightarrow u_I x_I v_I \, [p] = w'_I$ and $S_O \Rightarrow^* w_O \, [\alpha] = u_O A_O v_O \Rightarrow u_O x_O v_O \, [p] = w'_O$ in H. Claim 6 holds. Furthermore, if $S \Rightarrow^* w$ in H', where $w \in T^*$, then $w_I \in T_I^*$ and $\Delta(w_O) = \varepsilon$, thus $w_O \in T_O^*$. $L(H') \subseteq L_I(H)$. □

Theorem 4 holds. □

Theorem 7.

$$\mathscr{L}(SSCG) = RE$$

Proof. Clearly, $\mathscr{L}(SSCG) \subseteq RE$ holds. From definition, it follows that $\mathscr{L}(SCG) \subseteq \mathscr{L}(SSCG)$. Because $\mathscr{L}(SCG) = RE$, $RE \subseteq \mathscr{L}(SSCG)$ must also hold. □

5. Applications in Natural Language Translation

In this section, we discuss possibilities of applications in linguistics. More specifically, we present examples of translating some grammatical structures from Japanese to English.

First, to demonstrate the basic principle, we consider a simple Japanese sentence

Takeshi-san wa raishuu Oosaka ni ikimasu.

We will transform this sentence (or, more precisely, the structure of this sentence) into its English counterpart

Takeshi is going to Osaka next week.

In the following examples, words in angled brackets (⟨⟩) are words associated with a terminal or nonterminal symbol in a given sentence or structure. Note that this is not an actual part of the formalism.

Example 2. Consider a RSCFG $H = (G_I, G_O, \Psi, \varphi_I, \varphi_O)$, where $G_I = (N_I, T_I, P_I, \mathrm{S}_I)$, $G_O = (N_O, T_O, P_O, \mathrm{S}_O)$, $N_I = \{\mathrm{S}_I$, NP-SBJ, VP, PP-TMP, PP-DIR$\}$, $T_I = \{$NP, V, Det$\}$, $N_O = \{\mathrm{S}_O$, NP-SBJ, VP, PP-TMP, PP-DIR$\}$, $T_O = \{$NP, V, Aux, Det, Prep$\}$,

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

$P_I =$

$$
\begin{array}{lrll}
\{\, 1: & \mathrm{S}_I \rightarrow \text{NP-SBJ VP} & 4: & \text{PP-TMP} \rightarrow \text{NP} \\
2: & \text{NP-SBJ} \rightarrow \text{NP Det}\langle \textit{wa} \rangle & 4z: & \text{PP-TMP} \rightarrow \varepsilon \\
3: & \text{VP} \rightarrow \text{PP-TMP PP-DIR V} & 5: & \text{PP-DIR} \rightarrow \text{NP Det}\langle \textit{ni} \rangle \\
 & & 5z: & \text{PP-DIR} \rightarrow \varepsilon \quad \},
\end{array}
$$

and $P_O =$

$$
\begin{array}{lrll}
\{\, 1: & \mathrm{S}_O \rightarrow \text{NP-SBJ VP} & 4: & \text{PP-TMP} \rightarrow \text{NP} \\
2: & \text{NP-SBJ} \rightarrow \text{NP} & 4z: & \text{PP-TMP} \rightarrow \varepsilon \\
3: & \text{VP} \rightarrow \text{Aux V PP-DIR PP-TMP} & 5: & \text{PP-DIR} \rightarrow \text{Prep}\langle \textit{to} \rangle \text{ NP} \\
 & & 5z: & \text{PP-DIR} \rightarrow \varepsilon \quad \}.
\end{array}
$$

Strictly according to the definitions, synchronous grammars generate pairs of sentences. However, in practice, we usually have the input sentence in the source language, and we want to translate it into the target language. That is, we want to generate the corresponding output sentence. The translation can be divided into two steps. First, we parse the input sentence using the input grammar. In G_I, a derivation that generates the example sentence may proceed as follows:

$\mathrm{S}_I \Rightarrow$ NP-SBJ VP [1] $\Rightarrow$ NP⟨*Takeshi-san*⟩ Det⟨*wa*⟩ VP [2] $\Rightarrow$ NP⟨*Takeshi-san*⟩ Det⟨*wa*⟩ PP-TMP PP-DIR V⟨*ikimasu*⟩ [3] $\Rightarrow$ NP⟨*Takeshi-san*⟩ Det⟨*wa*⟩ NP⟨*raishuu*⟩ PP-DIR V⟨*ikimasu*⟩ [4] $\Rightarrow$ NP⟨*Takeshi-san*⟩ Det⟨*wa*⟩ NP⟨*raishuu*⟩ NP⟨*Oosaka*⟩ Det⟨*ni*⟩ V⟨*ikimasu*⟩ [5]

We have applied rules denoted by labels $1\ 2\ 3\ 4\ 5$, in that order. Next, we use the sequence obtained in the first step, and apply the corresponding rules in the output grammar. Then, the derivation in G_O proceeds as follows:

$\mathrm{S}_O \Rightarrow$ NP-SBJ VP [1] $\Rightarrow$ NP⟨*Takeshi*⟩ VP [2] $\Rightarrow$ NP⟨*Takeshi*⟩ Aux⟨*is*⟩ V⟨*going*⟩ PP-TMP PP-DIR [3] $\Rightarrow$ NP⟨*Takeshi*⟩ Aux⟨*is*⟩ V⟨*going*⟩ PP-DIR NP⟨*next week*⟩ [4] $\Rightarrow$ NP⟨*Takeshi*⟩ Aux⟨*is*⟩ V⟨*going*⟩ Prep⟨*to*⟩ NP⟨*Osaka*⟩ NP⟨*next week*⟩ [5]

Also note the rules $4z$ and $5z$, which can be used to erase PP-TMP and PP-DIR. This represents the fact that these constituents may be omitted.

Next, compare the following sentences in Japanese (left) and English.

Watashi wa raishuu Oosaka ni ikimasu.	*I am going to Osaka next week.*
Anata wa raishuu Oosaka ni ikimasu.	*You are going to Osaka next week.*
Takeshi-san wa raishuu Oosaka ni ikimasu.	*Takeshi is going to Osaka next week.*

In English, the form of the auxiliary verb *to be* depends on many grammatical categories such as tense, number, or, as shown in this example, person – *am* for first person (present tense, singular), *are* for second, and *is* for third. On the other hand, note that the verb in the Japanese sentences (*ikimasu*, long form of *iku*) is always the same (out of the grammatical categories mentioned, only tense would affect inflection). If we want to

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

translate such sentence from Japanese to English, we need to choose the correct form of the verb. We can get the necessary information by looking at the subject.

Example 3. Consider a SMAT $H = (G_I, M_I, G_O, M_O, \Psi, \varphi_I, \varphi_O)$, where $G_I = (N_I, T_I, P_I, \mathrm{S}_I)$, $G_O = (N_O, T_O, P_O, \mathrm{S}_O)$, $N_I = \{\mathrm{S}_I$, NP-SBJ, VP, PP-TMP, PP-DIR$\}$, $T_I = \{$NP, NP$_1$, NP$_2$, NP$_3$, V, Det$\}$, $N_O = \{\mathrm{S}_O$, NP-SBJ, VP, PP-TMP, PP-DIR, Aux$_x\}$, $T_O = \{$NP, NP$_1$, NP$_2$, NP$_3$, V, Aux$_1$, Aux$_2$, Aux$_3$, Det, Prep$\}$,
$P_I =$

$$\begin{array}{rlrl} \{\ 1: & \mathrm{S}_I \to \text{NP-SBJ VP} & 4: & \text{PP-TMP} \to \text{NP} \\ 2a: & \text{NP-SBJ} \to \text{NP}_1\ \text{Det}\langle \textit{wa} \rangle & 4z: & \text{PP-TMP} \to \varepsilon \\ 2b: & \text{NP-SBJ} \to \text{NP}_2\ \text{Det}\langle \textit{wa} \rangle & 5: & \text{PP-DIR} \to \text{NP Det}\langle \textit{ni} \rangle \\ 2c: & \text{NP-SBJ} \to \text{NP}_3\ \text{Det}\langle \textit{wa} \rangle & 5z: & \text{PP-DIR} \to \varepsilon \\ 3: & \text{VP} \to \text{PP-TMP PP-DIR V} & & \quad \}, \end{array}$$

$P_O =$

$$\begin{array}{rlrl} \{\ 1: & \mathrm{S}_O \to \text{NP-SBJ VP} & 4z: & \text{PP-TMP} \to \varepsilon \\ 2a: & \text{NP-SBJ} \to \text{NP}_1 & 5: & \text{PP-DIR} \to \text{Prep}\langle \textit{to} \rangle\ \text{NP} \\ 2b: & \text{NP-SBJ} \to \text{NP}_2 & 5z: & \text{PP-DIR} \to \varepsilon \\ 2c: & \text{NP-SBJ} \to \text{NP}_3 & 6a: & \text{Aux}_x \to \text{Aux}_1 \\ 3: & \text{VP} \to \text{Aux}_x\ \text{V PP-DIR PP-TMP} & 6b: & \text{Aux}_x \to \text{Aux}_2 \\ 4: & \text{PP-TMP} \to \text{NP} & 6c: & \text{Aux}_x \to \text{Aux}_3 \quad \}, \end{array}$$

$M_I = \{m_1\colon 1, m_{2a}\colon 2a, m_{2b}\colon 2b, m_{2c}\colon 2c, m_3\colon 3, m_4\colon 4, m_{4z}\colon 4z, m_5\colon 5, m_{5z}\colon 5z\}$, and $M_O = \{m_1\colon 1, m_{2a}\colon 2a\,6a, m_{2b}\colon 2b\,6b, m_{2c}\colon 2c\,6c, m_3\colon 3, m_4\colon 4, m_{4z}\colon 4z, m_5\colon 5, m_{5z}\colon 5z\}$.

An example of a derivation follows.

$\mathrm{S}_I \Rightarrow$ NP-SBJ VP $[m_1] \Rightarrow$ NP-SBJ PP-TMP PP-DIR V$\langle$*ikimasu*$\rangle$ $[m_3] \Rightarrow$ NP$_1\langle$*watashi*$\rangle$ Det$\langle$*wa*$\rangle$ PP-TMP PP-DIR V$\langle$*ikimasu*$\rangle$ $[m_{2a}] \Rightarrow$ NP$_1\langle$*watashi*$\rangle$ Det$\langle$*wa*$\rangle$ NP$\langle$*raishuu*$\rangle$ PP-DIR V$\langle$*ikimasu*$\rangle$ $[m_4] \Rightarrow$ NP$_1\langle$*watashi*$\rangle$ Det$\langle$*wa*$\rangle$ NP$\langle$*raishuu*$\rangle$ NP$\langle$*Oosaka*$\rangle$ Det$\langle$*ni*$\rangle$ V$\langle$*ikimasu*$\rangle$ $[m_5]$

$\mathrm{S}_O \Rightarrow$ NP-SBJ VP $[m_1] \Rightarrow$ NP-SBJ Aux$_x\langle$*be*$\rangle$ V$\langle$*going*$\rangle$ PP-DIR PP-TMP $[m_3] \Rightarrow$ NP$_1\langle$*I*$\rangle$ Aux$_1\langle$*am*$\rangle$ V$\langle$*going*$\rangle$ PP-DIR PP-TMP $[m_{2a}] \Rightarrow$ NP$_1\langle$*I*$\rangle$ Aux$_1\langle$*am*$\rangle$ V$\langle$*going*$\rangle$ PP-DIR NP$\langle$*next week*$\rangle$ $[m_4] \Rightarrow$ NP$_1\langle$*I*$\rangle$ Aux$_1\langle$*am*$\rangle$ V$\langle$*going*$\rangle$ Prep$\langle$*to*$\rangle$ NP$\langle$*Osaka*$\rangle$ NP$\langle$*next week*$\rangle$ $[m_5]$

Depending on person of the subject, we apply one of the matrices m_{2a}, m_{2b}, or m_{2c}. In the input grammar (Japanese), these matrices contain only one rule, which involves the subject itself. The verb is unaffected. In the output grammar (English), the matrices contain two rules, which ensure agreement between the subject and the (auxiliary) verb.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

Instead of SMAT, we could also use SSCG to the same effect, with scattered context rules such as $(\text{NP-SBJ}, \text{Aux}_x) \rightarrow (\text{NP}_1, \text{Aux}_1)$.

Let us have a look at some other syntactic structures. For instance, to form a question in Japanese, we can simply take a statement and append the particle *ka* at the end of the sentence. In English, we need to place the auxiliary verb in front of the subject. Compare the following sentences in Japanese (left) and English.

Takeshi-san wa raishuu Oosaka ni ikimasu ka.	*Is Takeshi going to Osaka next week?*
Takeshi-san wa raishuu doko ni ikimasu ka.	*Where is Takeshi going next week?*
Takeshi-san wa itsu Oosaka ni ikimasu ka.	*When is Takeshi going to Osaka?*

Observe that in Japanese, the only difference between the three questions is the word *Oosaka* being replaced by *doko* (*where*), or *raishuu* by *itsu* (*when*). However, in English, the structure of the sentence changes further, as illustrated by the derivation trees in Figure 1. The interrogative pronoun (*where*, *when*) is taken from its "original" position and placed at the beginning of the sentence (this common principle known as wh-movement is present in many languages besides English).

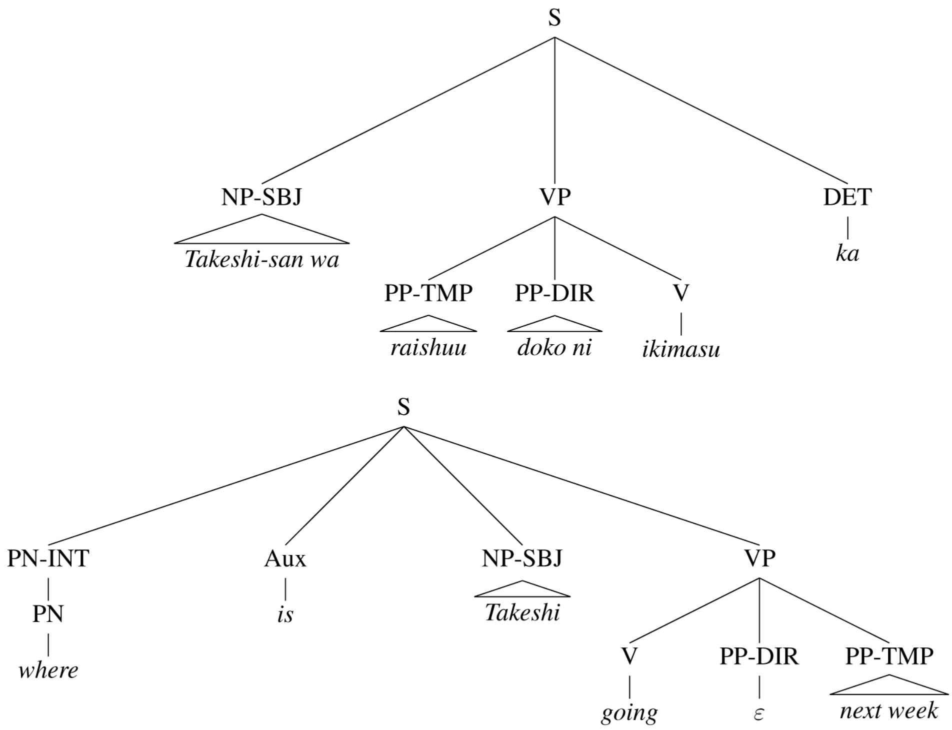

Fig. 1. Derivation trees for Japanese and English question

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

To reflect this in our grammar, we can add nonterminal symbol PN-INT to N_O and terminal symbol PN to both T_I and T_O. To P_I, we add rules

$$1q\colon\ \text{S} \to \text{NP-SBJ VP Det}\langle \textit{ka}\rangle \qquad 4q\colon\ \text{PP-TMP} \to \text{PN}\langle \textit{itsu}\rangle$$
$$5q\colon\ \text{PP-DIR} \to \text{PN}\langle \textit{doko}\rangle\ \text{Det}\langle \textit{ni}\rangle$$

and to P_O, rules

$$1q\colon\ \text{S} \to \text{Aux}_x\ \text{NP-SBJ VP} \qquad 3q\colon\ \text{VP} \to \text{V PP-DIR PP-TMP}$$
$$1qi\colon\ \text{S} \to \text{PN-INT Aux}_x\ \text{NP-SBJ VP} \qquad 7q\colon\ \text{PN-INT} \to \text{PN}$$

Finally, we add matrices $m_{1q}, m_{1qi}\colon 1q\,3$, $m_{4q}\colon 4q$, $m_{5q}\colon 5q$, to M_I and $m_{1q}\colon 1q\,3q$, $m_{1qi}\colon 1qi\,3q$, $m_{4q}\colon 7q\,4z$, $m_{5q}\colon 7q\,5z$ to M_O (recall that the rules $4z$ and $5z$ erase PP-TMP and PP-DIR, respectively).

An example of a derivation follows.

S_I ⇒ NP-SBJ PP-TMP PP-DIR V⟨*ikimasu*⟩ Det⟨*ka*⟩ $[m_{1qi}]$ ⇒ NP$_3$⟨*Takeshi-san*⟩ Det⟨*wa*⟩ PP-TMP PP-DIR V⟨*ikimasu*⟩ Det⟨*ka*⟩ $[m_{2c}]$ ⇒ NP$_3$⟨*Takeshi-san*⟩ Det⟨*wa*⟩ NP⟨*raishuu*⟩ PP-DIR V⟨*ikimasu*⟩ Det⟨*ka*⟩ $[m_4]$ ⇒ NP$_3$⟨*Takeshi-san*⟩ Det⟨*wa*⟩ NP⟨*raishuu*⟩ PN⟨*doko*⟩ Det⟨*ni*⟩ V⟨*ikimasu*⟩ Det⟨*ka*⟩ $[m_{5q}]$

S_O ⇒ PN-INT Aux$_x$⟨*be*⟩ NP-SBJ V⟨*going*⟩ PP-DIR PP-TMP $[m_{1qi}]$ ⇒ PN-INT Aux$_3$⟨*is*⟩ NP$_3$⟨*Takeshi*⟩ V⟨*going*⟩ PP-DIR PP-TMP $[m_{2c}]$ ⇒ PN-INT Aux$_3$⟨*is*⟩ NP$_3$⟨*Takeshi*⟩ V⟨*going*⟩ PP-DIR NP⟨*next week*⟩ $[m_4]$ ⇒ PN⟨*where*⟩ Aux$_3$⟨*is*⟩ NP$_3$⟨*Takeshi*⟩ V⟨*going*⟩ NP⟨*next week*⟩ $[m_{5q}]$

Note that the matrices ensuring subject-verb agreement (see Example 3) work without any changes for all of the structures discussed above. Clearly, it is possible to capture the relation within a purely context-free framework as well. However, the necessary grammatical information would have to be propagated through the derivation tree. This means that we would have to add separate rules covering all possibilities (person, number. . .) for each of the structures, even though the structures themselves are not actually affected.

Using SMAT or SSCG, we are able to describe the relation more easily, with only a relatively small number of rules. The presented example is somewhat special, as *to be* is an irregular verb. In English, we usually only need to distinguish two cases: third person singular and other. In languages with rich inflection, such as Czech (see [5]), this advantage becomes even more important.

6. Conclusion

In this paper, we have discussed the generative power of synchronous grammars based on linked rules. To summarize the main results, we have obtained the following hierarchy of the studied language classes:

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

$$CF \subset \mathscr{L}(RSCFG) = \mathscr{L}(MAT) = \mathscr{L}(SMAT) \subset \mathscr{L}(SSCG) = RE$$

Further research prospects in this direction include study of the properties of synchronous grammars with additional restrictions, such as left-most derivation, or propagating scattered context grammars. We can also consider synchronization of other well-known formal models, using the same principle.

From the more practical point of view, we have presented models that can be used in natural language translation. We have demonstrated the basic principles on examples dealing with Japanese-English translation. We have shown that the extension beyond CFG allows us to describe some grammatical structures and transformations much more easily.

It is important to note that in our examples, we have made several assumptions. For example, we already have the sentence analyzed at low level, meaning that we can discern the individual words and have some basic grammatical information about them. For practical applications in machine translation, a more complex system is needed, incorporating other components such as a part-of-speech tagger and a dictionary to translate the actual meanings of the words. The component based on the discussed formal tools can be used to transform syntactic structures. Recently, many translation systems that use syntactic information have been developed (see [2], [7]). Such approaches are usually called syntax augmented, syntax-aided, or syntax based translation.

Another possible direction of further research is syntax analysis. In practice, we need to be able to parse the input sentence efficiently. While there are many well-known parsing algorithms for CFGs (such as CKY), matrix grammars and SCGs still present an open problem.

Acknowledgement

This work was partially supported by the research plan MSM0021630528, the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070), and the BUT FIT grant FIT-S-11-2.

References

1. A.V. Aho, J.D. Ullman: *Syntax directed translations and the pushdown assembler*. In Journal of Computer and System Sciences Volume 3, 1969, p. 37-56.
2. O. Bojar, M. Čmejrek: *Mathematical model of tree transformations*. Project Euroma-trix Deliverable 3.2, Charles University, Prague, 2007.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM

3. D. Chiang: *An introduction to synchronous grammars* [online]. Part of a tutorial given at 44th Annual Meeting of the Association for Computational Linguistics, 2006, http://www.isi.edu/ chiang/papers/synchtut.pdf
4. J. Dassow, Gh. Păun: *Regulated Rewriting in Formal Language Theory*. Springer, 1989, ISBN 3-540-51414-7.
5. J. Hajič: *Disambiguation of Rich Inflection: Computational Morphology of Czech*. Wisconsin Center for Pushkin Studies, Karolinum, 2004.
6. P. Horáček, A. Meduna: *Regulated rewriting in natural language translation*, In: 7th Doctoral Workshop on Mathematical and Engineering Methods in Computer Science, Brno, CZ, 2011, ISBN 978-80-214-4305-1, p. 35-42.
7. M. Khalilov, J.A.R. Fonollosa: *N-gram-based statistical machine translation versus syntax augmented machine translation: comparison and system combination*. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, EACL '09, Stroudsburg, PA, USA, 2009, p. 424-432.
8. P.M. Lewis, R.E. Stearns: *Syntax-directed transduction*. In Journal of the ACM Volume 15, 1968, p. 465-488.
9. A. Meduna: *Automata and Languages: Theory and Applications*. Springer, 2005, ISBN 1-85233-074-0, 892 p.
10. A. Meduna, J. Techet: *Scattered Context Grammars and their Applications*. WIT Press, UK, 2010, ISBN 978-1-84564-426-0, 224 p.
11. R. Mitkov (ed.): *The Oxford Handbook of Computational Linguistics*. Oxford University Press, 2004, ISBN 978-0-19-927634-9.
12. G. Rozenberg, A. Salomaa: *Handbook of Formal Languages: Volume I*. Springer, 1997.

Authenticated | 195.187.97.1
Download Date | 12/12/12 2:07 PM