Derivational Minimalism is Mildly Context–Sensitive*

Jens Michaelis

Universität Potsdam, Institut für Linguistik, PF 601553, 14415 Potsdam, Germany
michael@ling.uni-potsdam.de

Abstract. The change within the linguistic framework of transformational grammar from GB–theory to minimalism brought up a particular type of formal grammar, as well. We show that this type of a minimalist grammar (MG) constitutes a subclass of mildly context–sensitive grammars in the sense that for each MG there is a weakly equivalent linear context–free rewriting system (LCFRS). Moreover, an infinite hierarchy of MGs is established in relation to a hierarchy of LCFRSs.

1 Introduction

The change within the linguistic framework of transformational grammar from GB–theory to minimalism brought up a new formal grammar type, the type of a minimalist grammar (MG) introduced by Stabler (see e.g. [6, 7]), which is an attempt of a rigorous algebraic formalization of the new linguistic perspectives. One of the questions that arise from such a definition concerns the weak generative power of the corresponding grammar class. Stabler [6] has shown that MGs give rise to languages not derivable by any tree adjoining grammar (TAG). But he leaves open the "...problem to specify how the MG–definable string sets compare to previously studied supersets of the TAG language class." We address this issue here by showing that each MG as defined in [6] can be converted into a linear context–free rewriting system (LCFRS) which derives the same (string) language. In this sense MGs fall into the class of mildly context–sensitive grammars (MCSGs) rather informally introduced in [2] and described in e.g. [3].

The paper is structured as follows. We start by briefly repeating the definition of an LCFRS and the language it derives (Sect. 2). Turning to MGs, we then introduce the concept of a *relevant expression* in order to reduce the closure of an MG to such expressions (Sect. 3). Depending on this *relevant closure*, for a given MG we construct an LCFRS in detail and prove both grammars to be weakly equivalent (Sect. 4). Finally, an infinite hierarchy of MGs is introduced in relation to a hierarchy of LCFRSs. The former is unboundedly increasing, which is shown by presenting for each finite number an MG that derives a language with counting dependencies in size of this number (Sect. 5).

* This work has been carried out within the Innovationskolleg 'Formal Models of Cognitive Complexity' (INK 12) funded by the DFG. I especially wish to thank Marcus Kracht for inspiring discussions, and Peter Staudacher as well as an anonymous referee for a lot of valuable comments on a previous version of this paper.

2 Linear Context–Free Rewriting Systems

In order to keep the paper self–contained, in this section we quickly go through a number of definitions, which will be of interest in Sect. 4 again.

Definition 2.1 ([4]). A *generalized context–free grammar (GCFG)* is a five–tuple $G = (N, O, F, R, S)$ for which the conditions (G1)–(G5) hold.

(G1) N is a finite non–empty set of *nonterminal symbols*.

(G2) $O \subseteq \bigcup_{n \in \mathbb{N}} (\Sigma^*)^{n+1}$ for some finite non–empty set Σ of *terminal symbols* with $\Sigma \cap N = \emptyset$,[1] hence O is a set of finite tuples of finite strings in Σ.

(G3) F is a finite subset of $\bigcup_{n \in \mathbb{N}} F_n$, where F_n is the set of partial functions from O^n to O, i.e. F_0 is the set of constants in O.

(G4) $R \subseteq \bigcup_{n \in \mathbb{N}} (F \cap F_n) \times N^{n+1}$ is a finite set of *(rewriting) rules*.[2]

(G5) $S \in N$ is the distinguished *start symbol*.

Let $G = (N, O, F, R, S)$ be a GCFG. A rule $r = (f, A_0, A_1, \ldots, A_n) \in F_n \times N^{n+1}$ is generally written $A_0 \to f(A_1, \ldots, A_n)$, and just $A_0 \to f$ in case $n = 0$. If the latter, i.e. if $f \in O$ then r is *terminating*, otherwise r is *nonterminating*. For $A \in N$ and $k \in \mathbb{N}$ the set $L^k_G(A) \subseteq O$ is given recursively in the following sense:

(L1) $\theta \in L^0_G(A)$ for each terminating rule $A \to \theta \in R$.

(L2) $\theta \in L^{k+1}_G(A)$, if $\theta \in L^k_G(A)$ or if there is $A \to f(A_1, \ldots, A_n) \in R$ and there are $\theta_i \in L^k_G(A_i)$ for $1 \leq i \leq n$ such that $\theta = f(\theta_1, \ldots, \theta_n)$ is defined.

We say A *derives* θ *(in G)* if $\theta \in L^k_G(A)$ for some $k \in \mathbb{N}$. In this case θ is called an *A–phrase (in G)*. The *language derivable from A (by G)* is the set $L_G(A)$ of all A–phrases (in G), i.e. $L_G(A) = \bigcup_{k \in \mathbb{N}} L^k_G(A)$. The set $L(G) = L_G(S)$ is the *generalized context–free language (GCFL) (derivable by G)*.

Definition 2.2 ([5]). For every $m \in \mathbb{N}$ with $m \neq 0$ an *m–multiple context–free grammar (m–MCFG)* is a GCFG $G = (N, O, F, R, S)$ which satisfies (M1)–(M4).

(M1) $O = \bigcup_{i=1}^{m} (\Sigma^*)^i$.

(M2) For $f \in F$ let $n(f) \in \mathbb{N}$ be the number of arguments of f, i.e. $f \in F_{n(f)}$. For each $f \in F$ there are $r(f) \in \mathbb{N}$ and $d_i(f) \in \mathbb{N}$ for $1 \leq i \leq n(f)$ such that f is a (total) function from $(\Sigma^*)^{d_1(f)} \times \ldots \times (\Sigma^*)^{d_{n(f)}(f)}$ to $(\Sigma^*)^{r(f)}$ for which (f1) and, in addition, the anti–copying condition (f2) hold.

(f1) Let $X = \{x_{ij} \,|\, 1 \leq i \leq n(f), 1 \leq j \leq d_i(f)\}$ be a set of pairwise distinct variables, and let $x_i = (x_{i1}, \ldots, x_{id_i(f)})$ for $1 \leq i \leq n(f)$. For $1 \leq h \leq r(f)$ let f^h be the h–th component of f, i.e. $f(\theta) = (f^1(\theta), \ldots, f^{r(f)}(\theta))$ for all $\theta = (\theta_1, \ldots, \theta_{n(f)}) \in (\Sigma^*)^{d_1(f)} \times \ldots \times (\Sigma^*)^{d_{n(f)}(f)}$. Then for each component f^h there is an $l_h(f) \in \mathbb{N}$ such that f^h can be represented by

[1] $\mathbb{N}$ denotes the set of all non–negative integers. For any non–empty set M and $n \in \mathbb{N}$, M^{n+1} is the set of all $n+1$–tuples in M, i.e. the set of all finite strings in M with length $n+1$. M^* is the set of all finite strings in M including the empty string ϵ.

[2] For any two sets M_1 and M_2, $M_1 \times M_2$ is the set of all pairs with 1st component in M_1 and 2nd component in M_2.

(c$_h$) $$f^h(x_1,\dots,x_{n(f)}) = \zeta_{h0} z_{h1} \zeta_{h1} \dots z_{hl_h(f)} \zeta_{hl_h(f)}$$

with $\zeta_{hl} \in \Sigma^*$ for $0 \leq l \leq l_h(f)$ and $z_{hl} \in X$ for $1 \leq l \leq l_h(f)$.

(f2) For each $1 \leq i \leq n(f)$ and $1 \leq j \leq d_i(f)$ there is at most one $1 \leq h \leq r(f)$ and at most one $1 \leq l \leq l_h(f)$ such that $x_{ij} = z_{hl}$, i.e. z_{hl} is the only occurrence of $x_{ij} \in X$ in all righthand sides of (c$_1$)–(c$_{r(f)}$).

(M3) There is a function d from N to $\mathbb{N}$ such that, if $A_0 \to f(A_1,\dots,A_{n(f)}) \in R$ then $r(f) = d(A_0)$ and $d_i(f) = d(A_i)$ for $1 \leq i \leq n(f)$.

(M4) $d(S) = 1$ for the start symbol S.

The language $L(G)$ is an *m–multiple context–free language (m–MCFL).*

In case that $m = 1$ and that each $f \in F \setminus F_0$ is the concatenation function from $(\Sigma^*)^{n+1}$ to Σ^* for some $n \in \mathbb{N}$, G is a *context–free grammar (CFG)* and $L(G)$ a *context–free language (CFL)* in the usual sense.

Definition 2.3 ([8]). For $m \in \mathbb{N}$ with $m \neq 0$ an m–MCFG $G = (N, O, F, R, S)$ according to Definition 2.2 is an *m–linear context–free linear rewriting system (m–LCFRS)* if for all $f \in F$ the non–erasure condition (f3) holds in addition to (f1) and (f2).

(f3) For each $1 \leq i \leq n(f)$ and $1 \leq j \leq d_i(f)$ there are $1 \leq h \leq r(f)$ and $1 \leq l \leq l_h(f)$ such that $x_{ij} = z_{hl}$, i.e. each $x_{ij} \in X$ has to appear in one of the righthand sides of (c$_1$)–(c$_{r(f)}$).

The language $L(G)$ is an *m–linear context–free rewriting language (m–LCFRL).*

A grammar is also called an *MCFG* (*LCFRS*) if it is an m–MCFG (m–LCFRS) for some $m \in \mathbb{N} \setminus \{0\}$. A language is an *MCFL* (*LCFRL*) if it is derivable by some MCFG (LCFRS). The class of MCFGs is essentially the same as the class of LCFRSs. The latter was first described in [8] and has been studied in some detail in [9]. The "non-erasing property" (f3), motivated by linguistic considerations, is omitted in the general MCFG–definition. [5] shows that for each $m \in \mathbb{N} \setminus \{0\}$ the class of m–MCFLs and that of m–LCFRLs are equal. In Sect. 4 we in fact construct an LCFRS that is weakly equivalent to a given minimalist grammar.

3 Minimalist Grammars

We first give the definition of a minimalist grammar along the lines of [6].[3] Then, we introduce a "concept of relevance" being of central importance later on.

Definition 3.1. A five–tuple $\tau = (N_\tau, \triangleleft^*_\tau, \prec_\tau, <_\tau, Label_\tau)$ fulfilling (E1)–(E3) is called an *expression (over a feature–set F).*

(E1) $(N_\tau, \triangleleft^*_\tau, \prec_\tau)$ is a finite, binary ordered tree. N_τ denotes the non–empty set of nodes. $\triangleleft^*_\tau$ and $\prec_\tau$ denote the usual relations of *dominance* and *precedence* defined on a subset of $N_\tau \times N_\tau$, respectively. I.e. $\triangleleft^*_\tau$ is the reflexive and

[3] Recall that we use ϵ to denote the empty string, whereas [6] uses λ.

transitive closure of $\lhd_\tau$, the relation of *immediate dominance*.[4]

(E2) $<_\tau \subseteq N_\tau \times N_\tau$ denotes the asymmetric relation of *(immediate) projection* which holds for any two siblings in $(N_\tau, \lhd_\tau^*, \prec_\tau)$, i.e. each node different from the root either *(immediately) projects* over its sibling or vice versa.

(E3) The function $Label_\tau$ assigns a string from F^* to every leaf of $(N_\tau, \lhd_\tau^*, \prec_\tau)$, i.e. a leaf–label is a finite sequence of features from F.

The set of all expressions over F is denoted by $Exp(F)$.

Let F be a set of features. Consider $\tau = (N_\tau, \lhd_\tau^*, \prec_\tau, <_\tau, Label_\tau) \in Exp(F)$.

A node $x \in N_\tau$ is a *maximal projection*, if it is the root of τ or if x's sister projects over x. Each $x \in N_\tau$ has a *head* $h(x) \in N_\tau$, a leaf such that $x \lhd_\tau^* h(x)$, and such that each $y \in N_\tau$ on the path from x to $h(x)$ with $y \neq x$ projects over its sister. The *head of* τ is the head of τ's root r_τ.

τ *has feature* $f \in F$ if τ's head–label starts with f. τ is *simple* (a *head*) if it consists of exactly one node, otherwise τ is *complex* (a *non–head*).

Suppose υ and $\phi \in Exp(F)$ to be subtrees of τ with roots r_υ and r_ϕ, respectively, such that $r_\tau \lhd_\tau r_\upsilon, r_\phi$. Then we take $[_< \upsilon, \phi\,]$ ($[_> \phi, \upsilon\,]$) to denote τ in case that $r_\upsilon <_\tau r_\phi$ and $r_\upsilon \prec_\tau r_\phi$ ($r_\phi \prec_\tau r_\upsilon$).

Definition 3.2 ([6]). A 4–tuple $G = (V, Cat, Lex, \mathcal{F})$ that obeys (N1)–(N4) is called a *minimalist grammar (MG)*.

(N1) $V = P \cup I$ is a finite set of *non–syntactic features*, where P is a set of *phonetic features* and I is a set of *semantic features*.

(N2) Cat is a finite set of *syntactic features* partitioned into the sets *base*, *select*, *licensees* and *licensors* such that for each *(basic) category* $\mathtt{x} \in base$ the existence of $=\!\mathtt{x}$, $=\!\mathtt{X}$ and $\mathtt{X}\!= \in select$ is possible, and for each $-\mathtt{x} \in licensees$ the existence of $+\mathtt{x}$ and $+\mathtt{X} \in licensors$. Moreover, the set *base* contains at least the category $\mathtt{c}$.

(N3) Lex is a finite set of expressions over $V \cup Cat$ such that for each tree $\tau = (N_\tau, \lhd_\tau^*, \prec_\tau, <_\tau, Label_\tau) \in Lex$ the function $Label_\tau$ assigns a string from $select^*(licensors \cup \{\epsilon\})select^*(base \cup \{\epsilon\})licensees^* P^* I^*$ to each leaf in $(N_\tau, \lhd_\tau^*, \prec_\tau)$.

(N4) The set $\mathcal{F}$ consists of the structure building functions *merge* and *move* as defined in (me) and (mo), respectively.

(me) The function *merge* is a partial mapping from $Exp(V \cup Cat) \times Exp(V \cup Cat)$ to $Exp(V \cup Cat)$. A pair of expressions (υ, ϕ) belongs to Dom(*merge*) if υ has feature $=\!\mathtt{x}$, $=\!\mathtt{X}$ or $\mathtt{X}\!=$ and ϕ has category $\mathtt{x}$ for some $\mathtt{x} \in base$.[5] Then,

(me.1) $merge(\upsilon, \phi) = [_< \upsilon', \phi'\,]$ if υ is simple and has feature $=\!\mathtt{x}$,

[4] Up to an isomorphism N_τ is a unique *prefix closed* and *left closed* subset of $\mathbb{N}^*$, i.e. $\chi \in N_\tau$ if $\chi\chi' \in N_\tau$, and $\chi i \in N_\tau$ if $\chi j \in N_\tau$ for $\chi, \chi' \in \mathbb{N}^*$ and $i, j \in \mathbb{N}$ with $i < j$, such that for $\chi, \psi \in N_\tau$ hold: $\chi \lhd_\tau \psi$ iff $\psi = \chi i$ for some $i \in \mathbb{N}$, and $\chi \prec_\tau \psi$ iff $\chi = \omega i \chi'$ and $\psi = \omega j \psi'$ for some $\omega, \chi', \psi' \in \mathbb{N}^*$ and $i, j \in \mathbb{N}$ with $i < j$.

[5] For each (partial) mapping f from a set M_1 into a set M_2 we take Dom(f) to denote the *domain of* f, the subset of M_1 for which f is defined.

where υ' and ϕ' are expressions resulting from υ and ϕ, respectively, by deleting the feature the respective head–label starts with.

(me.2) $$merge(\upsilon,\phi) = [_< \upsilon', \phi'\,] \text{ if } \upsilon \text{ is simple and has feature } {}^{=}\mathtt{X},$$

where υ' and ϕ' are expressions resulting from υ and ϕ, respectively, by deleting the feature the respective head label starts with. In addition the phonetic features π_ϕ of the head of ϕ are canceled in ϕ', and the phonetic features π_υ of the head of υ are replaced by $\pi_\upsilon\pi_\phi$ in υ'.

(me.3) $$merge(\upsilon,\phi) = [_< \upsilon', \phi'\,] \text{ if } \upsilon \text{ is simple and has feature } \mathtt{X}^{=},$$

where υ' and ϕ' are expressions resulting from υ and ϕ, respectively, by deleting the feature the respective head label starts with. In addition the phonetic features π_ϕ of the head of ϕ are canceled in ϕ', and the phonetic features π_υ of the head of υ are replaced by $\pi_\phi\pi_\upsilon$ in υ'.

(me.4) $$merge(\upsilon,\phi) = [_> \phi', \upsilon'\,] \text{ if } \upsilon \text{ is complex and has feature } {}^{=}\mathtt{x},$$

where υ' and ϕ' are expressions as in case (me.1).

(mo) The function *move* is a partially defined mapping from $Exp(V \cup Cat)$ to $Exp(V \cup Cat)$. An expression υ belongs to Dom(*move*) in case that υ has feature $+\mathtt{x}$ or $+\mathtt{X} \in licensors$, and υ has exactly one subtree ϕ that is rooted by a maximal projection and has feature $-\mathtt{x} \in licensees$. Then,

(mo.1) $$move(\upsilon) = [_> \phi', \upsilon'\,] \text{ if } \upsilon \text{ has feature } +\mathtt{X}$$

Here υ' results from υ by deleting the feature $+\mathtt{x}$ from υ's head–label, while the subtree ϕ is replaced by a single node labeled ϵ. ϕ' is the expression resulting from ϕ just by deleting the licensee feature $-\mathtt{x}$ that ϕ's head–label starts with.

(mo.2) $$move(\upsilon) = [_> \phi', \upsilon'\,] \text{ if } \upsilon \text{ has feature } +\mathtt{x}$$

Here υ' results from υ by deleting the feature $+\mathtt{x}$ from υ's head–label, while within the subtree ϕ all non–phonetic features are deleted. ϕ' is the expression resulting from ϕ by deleting the licensee feature $-\mathtt{x}$ that ϕ's head–label starts with, and all phonetic features that appear in ϕ.

A feature of the form ${}^{=}\mathtt{X}$, $\mathtt{X}^{=}$ or $+\mathtt{X}$ is called *strong*, one of the form ${}^{=}\mathtt{x}$ or $+\mathtt{x}$ is called *weak*. A strong selection feature ${}^{=}\mathtt{X}$ or $\mathtt{X}^{=}$ triggers *(overt) head movement*, i.e. incorporation of the phonetic head–features of a possibly complex expression into the selecting head (cf. (me.2), (me.3)). A strong licensor $+\mathtt{X}$ triggers *overt (phrasal) movement*, also called *pied–piping* (cf. (mo.1)). A weak licensor $+\mathtt{x}$ triggers *covert (phrasal) movement* (cf. (mo.2)).

Example 3.3. Assume G_2 to be the MG for which $I = \emptyset$ and $P = \{/\mathtt{a}_1/, /\mathtt{a}_2/\}$, while $base = \{\mathtt{c}\} \cup \{\mathtt{b}_1, \mathtt{b}_2, \mathtt{c}_1, \mathtt{c}_2, \mathtt{d}_1, \mathtt{d}_2\}$, $select = \{{}^{=}\mathtt{b}_1, {}^{=}\mathtt{b}_2, {}^{=}\mathtt{c}_1, {}^{=}\mathtt{c}_2, {}^{=}\mathtt{d}_1, {}^{=}\mathtt{d}_2\}$, $licensees = \{-\mathtt{l}_1, -\mathtt{l}_2\}$ and $licensors = \{+\mathtt{L}_1, +\mathtt{L}_2\}$, and while Lex consists of

$\alpha_0 = \mathtt{c}$ $\quad\gamma_1 = {}^{=}\mathtt{b}_2{+}\mathtt{L}_1\mathtt{c}_1{-}\mathtt{l}_1/\mathtt{a}_2/$ $\quad\delta_1 = {}^{=}\mathtt{b}_2{+}\mathtt{L}_1\mathtt{d}_1$ $\quad\zeta_0 = {}^{=}\mathtt{d}_2\mathtt{c}$

$\beta_1 = \mathtt{b}_1{-}\mathtt{l}_1/\mathtt{a}_2/$ $\quad\gamma_1' = {}^{=}\mathtt{c}_2{+}\mathtt{L}_1\mathtt{c}_1{-}\mathtt{l}_1/\mathtt{a}_2/$ $\quad\delta_1' = {}^{=}\mathtt{c}_2{+}\mathtt{L}_1\mathtt{d}_1$

$\beta_2 = {}^{=}\mathtt{b}_1\mathtt{b}_2{-}\mathtt{l}_2/\mathtt{a}_1/$ $\quad\gamma_2 = {}^{=}\mathtt{c}_1{+}\mathtt{L}_2\mathtt{c}_2{-}\mathtt{l}_2/\mathtt{a}_1/$ $\quad\delta_2 = {}^{=}\mathtt{d}_1{+}\mathtt{L}_2\mathtt{d}_2$

Then e.g. $move(merge(\gamma_2, move(merge(\gamma_1, merge(\beta_2, \beta_1))))) \in Exp(V \cup Cat)$.

Let $G = (V, Cat, Lex, \mathcal{F})$ be an MG. Then $CL(G) = \bigcup_{k\in\mathbb{N}} CL^k(G)$ is the *closure of Lex (under the functions in $\mathcal{F}$)*. For $k \in \mathbb{N}$ the sets $CL^k(G) \subseteq Exp(V \cup Cat)$ are inductively defined by

(C1) $CL^0(G) = Lex$

(C2) $CL^{k+1}(G) = CL^k(G)$

$$\cup\, \{merge(\upsilon, \phi) \,|\, (\upsilon, \phi) \in \mathrm{Dom}(merge) \cap CL^k(G) \times CL^k(G)\}$$
$$\cup\, \{move(\upsilon) \,|\, \upsilon \in \mathrm{Dom}(move) \cap CL^k(G)\}$$

Each $\tau \in CL(G)$ is called an *expression in G*. Such a τ is *complete (in G)* if its head–label is in $\{\mathtt{c}\}P^*I^*$ and each other of its leaf–labels is in P^*I^*. Hence, a complete expression has category $\mathtt{c}$, and this instance of $\mathtt{c}$ is the only instance of a syntactic feature within all leaf–labels.

The *(phonetic) yield* $Y(\tau)$ of an expression $\tau \in Exp(V \cup Cat)$ is the string created by concatenating τ's leaf–labels "from left to right" and stripping off all non–phonetic features. $L(G) = \{Y(\tau) \,|\, \tau \in CL(G)$ with τ is complete$\}$ is the *(string) language (derivable by G)* and is called a *minimalist language (ML)*.

Example 3.4. Consider the MG G_2 from Example 3.3. Let $\tau^{(1)} = merge(\beta_2, \beta_1)$, $\tau^{(2)} = merge(\gamma_1, \tau^{(1)})$ and $\upsilon^{(2)} = merge(\delta_1, \tau^{(1)})$. For $k \in \mathbb{N}$ with $k \neq 0$ define

$\tau^{(2k+1)} = move(\tau^{(2k)})$, $\tau^{(4k)} = merge(\gamma_2, \tau^{(4k-1)})$, $\tau^{(4k+2)} = merge(\gamma_1', \tau^{(4k+1)})$,
$\upsilon^{(2k+1)} = move(\upsilon^{(2k)})$, $\upsilon^{(4k)} = merge(\delta_2, \upsilon^{(4k-1)})$, $\upsilon^{(4k+2)} = merge(\delta_1', \tau^{(4k+1)})$

and $\phi^{(4k+2)} = merge(\zeta_0, \upsilon^{(4k+1)})$. Then we have

$$CL^1(G_2) \setminus CL^0(G_2) = \{\tau^{(1)}\} \;\text{ and }\; CL^2(G_2) \setminus CL^1(G_1) = \{\tau^{(2)}, \upsilon^{(2)}\},$$

while for $k \in \mathbb{N} \setminus \{0\}$ and $-1 \leq i \leq 1$ we have

$$CL^{4k+i}(G_2) \setminus CL^{4k+i-1}(G_2) = \{\tau^{(4k+i)}, \upsilon^{(4k+i)}\} \;\text{ and}$$
$$CL^{4k+2}(G_2) \setminus CL^{4k+1}(G_2) = \{\tau^{(4k+2)}, \upsilon^{(4k+2)}, \phi^{(4k+2)}\}.$$

The set of complete expressions in G_2 is $\{\alpha_0\} \cup \{\phi^{(4k+2)} \mid k \in \mathbb{N}, k \neq 0\}$, and the language derivable by G_2 is $\{/\mathtt{a}_1/^n/\mathtt{a}_2/^n \mid n \in \mathbb{N}\}$.

Definition 3.5. For each MG $G = (V, Cat, Lex, \mathcal{F})$, an expression $\tau \in CL(G)$ is called *relevant (in G)* if it has property (R).

(R) For any $-\mathtt{x} \in licensees$ there is at most one proper subtree $\tau_{-\mathtt{x}}$ of τ that is rooted by a maximal projection and has feature $-\mathtt{x}$.[6]

We take $Rel(G)$ to denote the set of all relevant expressions $\tau \in CL(G)$.

[6] In fact, this kind of structure is characteristic of each $\tau \in CL(G)$ involved in creating a complete expression in G as will become clear immediately.

Let $G = (V, Cat, Lex, \mathcal{F})$ be an MG and consider $RCL(G) = \bigcup_{k\in\mathbb{N}} RCL^k(G)$, a particularly restricted closure of G, the *relevant closure (of G)*. For $k \in \mathbb{N}$ the sets $RCL^k(G)$ are inductively defined w.r.t. $Rel(G)$ by

(R1) $RCL^0(G) = \{\tau \in Rel(G) \mid \tau \in Lex\}$

(R2) $RCL^{k+1}(G)$

$$= RCL^k(G)$$
$$\cup \{merge(\upsilon, \phi) \in Rel(G) \mid (\upsilon, \phi) \in \mathrm{Dom}(merge) \cap RCL^k(G) \times RCL^k(G)\}$$
$$\cup \{move(\upsilon) \in Rel(G) \mid \upsilon \in \mathrm{Dom}(move) \cap RCL^k(G)\}$$

Lemma 3.6. *If $\tau \in CL^k(G) \cap Rel(G)$ for some $k \in \mathbb{N}$, then $\tau \in RCL^k(G)$.*

A proof of Lemma 3.6 can straightforwardly be obtained by an induction on $k \in \mathbb{N}$.[7] On the other hand, it is an immediate consequence of the respective definitions that $RCL^k(G) \subseteq CL^k(G) \cap Rel(G)$ for each $k \in \mathbb{N}$. Thus,

Proposition 3.7. *$Rel(G) = RCL(G)$.*

Consequently, since each complete $\tau \in CL(G)$ has property (R), we can fix

Corollary 3.8. *$L(G) = \{Y(\tau) | \tau \in RCL(G)$ with τ is complete$\}$.*

This points out, why it is reasonable to call $RCL(G)$ the relevant closure (of G).

Remark 3.9. For G_2 as in Example 3.4, $RCL^k(G_2) = CL^k(G_2)$ for each $k \in \mathbb{N}$.

4 Weak Generative Power

Let $G_{\mathrm{MG}} = (V, Cat, Lex, \mathcal{F})$ be an MG with $\{-\mathtt{l}_i \mid 1 \leq i \leq m\}$ an enumeration of *licensees* for some $m \in \mathbb{N}$. We will construct an m+2–MCFG $G = (N, O, F, R, S)$ that derives the same language as G_{MG} (Corollary 4.5).

Thus, in G the start symbol S will derive exactly those strings of phonetic features that are the yield of some complete $\tau \in CL(G_{\mathrm{MG}})$. In order to achieve this, G will operate w.r.t. equivalence classes of a finite partition of $RCL(G_{\mathrm{MG}})$ rather than on single expressions. For each $\tau \in RCL(G_{\mathrm{MG}})$ there will be some nonterminal $T \in N$ coding τ's structure as it matters to *merge* and *move*, but ignoring non-syntactic features (cf. (D1),(D2)). τ's phonetic yield will be separately coded by some $p_T \in O$, a finite tuple of strings of phonetic features, that takes into account the structural information stored in T (cf. (D3),(D4)). p_T will be derivable from T in G as a finite recursion on functions in F, since for each particular application of *merge* or *move* in G_{MG} there will be some nonterminating rule in R simulating the corresponding structure building step in G_{MG} (Proposition 4.3).[8] Vice versa, whenever some $p_T \in O$ will be derivable in G

[7] Recall that $move(\tau)$ is defined for $\tau \in CL(G)$ only in case that there is exactly one subtree of τ that is rooted by a maximal projection and has a particular licensee feature allowing the subtree's "movement into specifier position."

[8] Note that for each relevant $\tau \in Lex$ there will be two terminating rules $T \to p_T \in R$ with $T \in N$ and $p_T \in O$ coding τ as just mentioned (cf. (r5)).

from some $T \in N$ that is different from S, there will be some $\tau \in RCL(G_{\mathrm{MG}})$ to which T and p_T correspond as outlined above (Proposition 4.4).

W.l.o.g. we may assume the head–label of each $\tau \in Lex$ to contain at least some category feature $\mathtt{x} \in base$.[9] Moreover, w.l.o.g. we may assume each $\tau \in Lex$ to be simple (a head). Thus, we can identify τ with its head–label. Doing so, for technical reasons we define sets $\mathrm{suf}(Cat)$ and $\mathrm{suf}(-\mathtt{l}_i)$ for $1 \leq i \leq m$ by

$$\mathrm{suf}(Cat) := \{\kappa \in Cat^* \mid \text{ex. } \kappa' \in Cat^* \text{ and } \pi\iota \in P^*I^* \text{ with } \kappa'\kappa\pi\iota \in Lex\}$$
$$\mathrm{suf}(-\mathtt{l}_i) := \{\kappa \in \mathrm{suf}(Cat) \mid \kappa = \epsilon \text{ or } \kappa = -\mathtt{l}_i\lambda \text{ for some } \lambda \in Cat^*\}$$

By (N3) each $\mathrm{suf}(-\mathtt{l}_i)$ as well as $\mathrm{suf}(Cat)$ is finite, and $\mathrm{suf}(-\mathtt{l}_i) \subseteq licensees^*$. Furthermore, we define

$$R_m := \{i_1 \ldots i_n \,|\, n \in \mathbb{N},\, i_1, \ldots, i_n \in \{1, \ldots, m\} \text{ with } i_j \neq i_k \text{ if } j \neq k\}$$

Note that R_m is finite, because in particular $|\alpha| \leq m$ for each $\alpha \in R_m$. Finally, we take `strong`, `weak`, `overt`, `covert`, `true`, `false`, `sim` and `com` to be pairwise distinct new symbols and now give the formal definitions of N and O, the set of nonterminals and the set of tuples of terminal strings, respectively, while we motivate these definitions in more detail, afterwards (cf. Definition 4.1).

• Each nonterminal $T \in N$ is either the start symbol S or an $m{+}2$–tuple of the form $(\widehat{\mu_0}, \widehat{\mu_1}, \ldots, \widehat{\mu_m}, t)$ with $t \in \{\mathtt{sim}, \mathtt{com}\}$ and $\widehat{\mu_i}$ a triple (μ_i, a_i, α_i), where

(n1) $\mu_0 \in \mathrm{suf}(Cat)$ with $\mu_0 \neq \epsilon$ and $a_0 \in \{\mathtt{strong}, \mathtt{weak}\}$,

(n2) $\mu_i \in \mathrm{suf}(-\mathtt{l}_i)$ and $a_i \in \{\mathtt{overt}, \mathtt{covert}, \mathtt{true}, \mathtt{false}\}$ for $1 \leq i \leq m$,

(n3) $\alpha_i \in \{1, \ldots, m\}^*$ for $0 \leq i \leq m$ with $\alpha_0\alpha_1 \ldots \alpha_m \in R_m$

such that for $1 \leq j \leq m$, in addition, (n4) and (n5) hold.

(n4) If $\alpha_j \neq \epsilon$ then $\alpha_i = \beta j \gamma$ for some $0 \leq i \leq m$, $i \neq j$, and $\beta, \gamma \in \{1, \ldots, m\}^*$.

(n5) $\mu_j \neq \epsilon$ iff $a_j \neq \mathtt{false}$ iff $\alpha_i = \beta j \gamma$ for some $0 \leq i \leq m$, $\beta, \gamma \in \{1, \ldots, m\}^*$.

Take $\lhd_T$ to be the following binary relation on $\{0, 1, \ldots, m\}$ induced by the α_i's:

($\lhd_T$) $i \lhd_T j$ iff $\alpha_i = \beta j \gamma$ for some $\beta, \gamma \in \{1, \ldots, m\}^*$.

Hence, if $i \lhd_T j$ then $i \neq j$, $\mu_i \neq \epsilon$ and $a_i \neq \mathtt{false}$ by (n1), (n3)–(n5). Let $\lhd_T^+$ and $\lhd_T^*$ denote the transitive and the reflexive, transitive closure of $\lhd_T$, respectively. Then take $\prec_T$ to be the following binary relation on $\{0, 1, \ldots, m\}$:

($\prec_T$) $j \prec_T k$ iff $\alpha_i = \beta j' \gamma k' \delta$

for some $0 \leq i, j', k' \leq m$ and $\beta, \gamma, \delta \in \{1, \ldots, m\}^*$ such that $j' \lhd_T^* j$, $k' \lhd_T^* k$.

It is easy to verify that the set N is in fact finite. Disregarding non–syntactic features, we can use N to characterize the relevant expressions in G_{MG}, which

[9] Recall that we are actually interested in complete expressions in $CL(G_{\mathrm{MG}})$, created from expressions in *Lex* by a finite number of applications of *merge* and *move*.

constitute the set $RCL(G_{\mathrm{MG}})$ by Proposition 3.7. This set is generally not finite, itself. The phonetic yield of an expression from $RCL(G_{\mathrm{MG}})$ can be characterized then as a particular tuple from $(P^*)^{m+2}$ depending on a corresponding nonterminal from N.

- We let $O = \bigcup_{i=0}^{m} (P^*)^{i+2}$, P the set of phonetic features in G_{MG}.

Consider $\tau \in RCL(G_{\mathrm{MG}})$. For $1 \leq i \leq m$ take, if existing, τ_i to be the unique proper subtree of τ rooted by a maximal projection and having licensee $-\mathtt{l}_i$.[10] Otherwise, take τ_i to be a single node labeled ϵ. Set $\tau_0 = \tau$ and for $0 \leq i \leq m$ let r_i denote the root of τ_i.

Now, let $T = (\widehat{\mu}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_m, t) \in N$ with $t \in \{\mathtt{sim}, \mathtt{com}\}$ and $\widehat{\mu}_i = (\mu_i, a_i, \alpha_i)$ for $0 \leq i \leq m$ according to (n1)–(n5), let $p_T = (\pi_H, \pi_0, \pi_1, \ldots, \pi_m) \in (P^*)^{m+2}$.

Definition 4.1. The pair (T, p_T) *corresponds to* τ if (D1)–(D4) are true.

(D1) For $0 \leq i \leq m$, μ_i is the prefix of τ_i's head-label consisting of just the syntactic features, and $t = \mathtt{sim}$ iff τ is simple.

(D2) For $0 \leq i, j \leq m$ with $\mu_i, \mu_j \neq \epsilon$, $i \lhd_T^+ j$ iff $r_i \lhd_\tau^+ r_j$, and $i \prec_T j$ iff $r_i \prec_\tau r_j$.

(D3) If $a_0 = \mathtt{weak}$ then $\pi_H = \epsilon$ and π_0 is the phonetic yield of $\tau_0 = \tau$ except for each substring that is the phonetic yield of some τ_i with $1 \leq i \leq m$ and $0 \lhd_T^+ i$ such that there is $1 \leq j \leq m$ with $0 \lhd_T^+ j \lhd_T^* i$ and $a_j = \mathtt{overt}$.
If $a_0 = \mathtt{strong}$ then π_H consists of the (ordered) phonetic features π of the head–label of $\tau_0 = \tau$, while π_0 is as in case $a_0 = \mathtt{weak}$ but lacking the substring π.

(D4) For $1 \leq i \leq m$, if $a_i \in \{\mathtt{covert}, \mathtt{true}, \mathtt{false}\}$ then $\pi_i = \epsilon$. If $a_i = \mathtt{overt}$ then π_i is the phonetic yield of τ_i except for each substring that is the phonetic yield of some τ_j with $1 \leq j \leq m$ and $i \lhd_T^+ j$ such that there is $1 \leq k \leq m$ with $i \lhd_T^+ k \lhd_T^* j$ and $a_k = \mathtt{overt}$.

Note that (D1) provides a method to install a finite partition $\mathcal{P}$ on $RCL(G_{\mathrm{MG}})$: In the given manner, to each $\tau \in RCL(G_{\mathrm{MG}})$ exactly one element belonging to the product $\mathrm{suf}(Cat) \times \mathrm{suf}(-\mathtt{l}_1) \times \ldots \times \mathrm{suf}(-\mathtt{l}_m) \times \{\mathtt{sim}, \mathtt{com}\}$ can be assigned.[11] (D2) can be seen then as introducing a refinement $\mathcal{P}_{\mathit{ref}}$ of $\mathcal{P}$: Expressions τ from one equivalence class are distinguished w.r.t. proper dominance, $\lhd_\tau^+$, and precedence, $\prec_\tau$, as it holds between each two distinct maximal projections r_i and r_j whose head–labels start with some licensee $-\mathtt{l}_i$ and $-\mathtt{l}_j$, respectively. This can be achieved by assigning to each $\tau \in RCL(G_{\mathrm{MG}})$ a particular $m+1$–tuple $(\alpha_0, \alpha_1, \ldots, \alpha_m)$ with $\alpha_i \in \{1, \ldots, m\}^*$ for $0 \leq i \leq m$ according to (n3)–(n5).

Again let $T = (\widehat{\mu}_0, \ldots, \widehat{\mu}_m, t) \in N$ with $t \in \{\mathtt{sim}, \mathtt{com}\}$ and $\widehat{\mu}_i = (\mu_i, a_i, \alpha_i)$ for $0 \leq i \leq m$ as in (n1)–(n5), let $p_T = (\pi_H, \pi_0, \ldots, \pi_m) \in (P^*)^{m+2}$ such that (T, p_T) corresponds to $\tau \in RCL(G_{\mathrm{MG}})$ according to Definition 4.1. For $0 \leq i \leq m$ each μ_i and α_i as well as t is unique, because (D1) and (D2) hold.

[10] Recall fn. 6.

[11] As a finite product of finite sets this product is also a finite set.

For each possible combination of a_i's, $0 \leq i \leq m$, there is exactly one p_T that satisfies the requirements of (D3) and (D4).

The μ_i's, the α_i's and t determine the equivalence class of τ w.r.t. the refined partition $\mathcal{P}_{\!ref}$ on $RCL(G_{\mathrm{MG}})$. Since either $a_0 = \mathtt{strong}$ or $a_0 = \mathtt{weak}$, we have added the possibility to respectively code whether the category $\mathtt{x}$ (of the head–label) of τ has to be selected by strong $^{=}\mathtt{X}$ or $\mathtt{X}^{=}$ or by weak $^{=}\mathtt{x}$. For $1 \leq i \leq m$ we have $a_i = \mathtt{false}$ iff there is no subtree τ_i that has licensee $-\mathtt{l}_i$. By $a_i = \mathtt{overt}$, $a_i = \mathtt{covert}$ or $a_i = \mathtt{true}$ we are able to respectively code, whether we expect the maximal subtree τ_i with licensee $-\mathtt{l}_i$ to move overtly, covertly or just to move in a later derivation step. In this sense, according to (D3) and (D4), for $0 \leq i \leq m$ the component π_i of p_T specifies the "non-extractable" part of the phonetic yield of τ_i, i.e. no overt movement can apply such that a proper subconstituent of τ_i is extracted pied piping some (proper) subpart of π_i. Recall that $\tau_0 = \tau$, here.

Example 4.2. Let the MG G_2 be as in Example 3.4. Consider the partition $\mathcal{P}$ on $RCL(G_2)$ induced by $\mathrm{suf}(Cat) \times \mathrm{suf}(-\mathtt{l}_1) \times \mathrm{suf}(-\mathtt{l}_2) \times \{\mathtt{sim}, \mathtt{com}\}$. In case of G_2 the corresponding refinement $\mathcal{P}_{\!ref}$ is identical with $\mathcal{P}$. $RCL(G_2) \setminus RCL^0(G_2)$, the set of complex expressions belonging to $RCL(G_2)$, divides into ten equivalence classes. One of which is finite, namely $\{\tau^{(1)}\}$, represented by $(\mathtt{b}_2{-}\mathtt{l}_2, -\mathtt{l}_1, \epsilon, \mathtt{com})$. The other classes and their respective representatives are

$$\{\tau^{(4k+2)} \mid k \in \mathbb{N}\} \text{ and } (+\mathtt{L}_1\mathtt{c}_1{-}\mathtt{l}_1\,, -\mathtt{l}_1\,, -\mathtt{l}_2\,, \mathtt{com}),$$
$$\{\upsilon^{(4k+2)} \mid k \in \mathbb{N}\} \text{ and } (\ \ +\mathtt{L}_1\mathtt{d}_1\ \ \ , -\mathtt{l}_1\,, -\mathtt{l}_2\,, \mathtt{com}),$$

$$\{\tau^{(4k-1)} \mid k \in \mathbb{N}, k \neq 0\} \text{ and } (\mathtt{c}_1{-}\mathtt{l}_1\,, \epsilon\,, -\mathtt{l}_2\,, \mathtt{com}),$$
$$\{\upsilon^{(4k-1)} \mid k \in \mathbb{N}, k \neq 0\} \text{ and } (\ \ \mathtt{d}_1\ \ \ , \epsilon\,, -\mathtt{l}_2\,, \mathtt{com}),$$

$$\{\tau^{(4k)} \mid k \in \mathbb{N}, k \neq 0\} \text{ and } (+\mathtt{L}_2\mathtt{c}_2{-}\mathtt{l}_2\,, -\mathtt{l}_1\,, -\mathtt{l}_2\,, \mathtt{com}),$$
$$\{\upsilon^{(4k)} \mid k \in \mathbb{N}, k \neq 0\} \text{ and } (\ \ +\mathtt{L}_2\mathtt{d}_2\ \ \ ,\ \ \epsilon\ \ \ , -\mathtt{l}_2\,, \mathtt{com}),$$

$$\{\tau^{(4k+1)} \mid k \in \mathbb{N}, k \neq 0\} \text{ and } (\mathtt{c}_2{-}\mathtt{l}_2\,, -\mathtt{l}_1\,, \epsilon\,, \mathtt{com}),$$
$$\{\upsilon^{(4k+1)} \mid k \in \mathbb{N}, k \neq 0\} \text{ and } (\ \ \mathtt{d}_2\ \ \ ,\ \ \epsilon\ \ \ , \epsilon\,, \mathtt{com}),$$

and finally $\{\phi^{(4k+2)} \mid k \in \mathbb{N}, k \neq 0\}$ and $(\mathtt{c}, \epsilon, \epsilon, \mathtt{com})$. Now, let N_2 be the nonterminal set according to (n1)–(n5) for G_2 and consider e.g.

$$T = ((+\mathtt{L}_1\mathtt{c}_1{-}\mathtt{l}_1, \mathtt{weak}, 2), (-\mathtt{l}_1, \mathtt{overt}, \epsilon), (-\mathtt{l}_2, \mathtt{overt}, 1), \mathtt{com}) \in N_2,$$
$$U = ((+\mathtt{L}_1\mathtt{d}_1, \mathtt{weak}, 2), (-\mathtt{l}_1, \mathtt{overt}, \epsilon), (-\mathtt{l}_2, \mathtt{overt}, 1), \mathtt{com}) \in N_2$$

and $p_T = (\epsilon, /\mathtt{a}_2/, /\mathtt{a}_2/^{k+1}, /\mathtt{a}_1/^{k+1})$, $p_U = (\epsilon, \epsilon, /\mathtt{a}_2/^{k+1}, /\mathtt{a}_1/^{k+1})$ with $k \in \mathbb{N}$. Then (T, p_T) and (U, p_U) correspond to $\tau^{(4k+2)}$ and $\upsilon^{(4k+2)}$, respectively.

Turning back to the general case of the MG G_{MG}, for the corresponding LCFRS G we will now define the set F of functions, manipulating tuples of tuples of terminal strings, and the set R of rewriting rules. In particular for all τ, υ and $\phi \in RCL(G_{\mathrm{MG}})$, if $\tau = merge(\upsilon, \phi)$ then rules of the form $T \to merge_{U,V}(U, V)$ and sometimes $T \to Merge_{U,V}(U, V)$ will belong to R (cf. (r1),(r2)), where $merge_{U,V}$ and $Merge_{U,V} \in F$ will be applicable to the pair (p_U, p_V) resulting in p_T. Similarly, if $\tau = move(\upsilon)$ there will be rules $T \to move_U(U)$ and sometimes $T \to Move_U(U) \in R$ (cf. (r3),(r4)), where $move_U$ and $Move_U \in F$ will be applicable to p_U calculating p_T as value. Here we have T, U and $V \in N$, while p_T, p_U and $p_V \in O$ such that (T, p_T), (U, p_U) and (V, p_V) respectively correspond to τ, υ and ϕ in the way given with Definition 4.1.

• The set F of functions and the set R of rewriting rules are simultaneously defined w.r.t. the occurrence of an $f \in F$ within an $r \in R$.

Nonterminating rules: First of all we define two initial rules by

(r0) $S \to con(T) \in R$ for $T = (\widehat{\mu}_0, \widehat{\mu}_1, \ldots, \widehat{\mu}_m, t) \in N$

with $\widehat{\mu}_0 = (\mathtt{c}, \mathtt{weak}, \epsilon)$, $\widehat{\mu}_i = (\epsilon, \mathtt{false}, \epsilon)$ for $1 \leq i \leq m$ and $t \in \{\mathtt{sim}, \mathtt{com}\}$. The concatenation function $con : (P^*)^{m+2} \to P^*$ is given by $x \mapsto x_H x_0 x_1 \ldots x_m$, where x denotes the $m{+}2$–tuple $(x_H, x_0, x_1, \ldots, x_m)$ consisting of the variables $x_H, x_0, x_1, \ldots, x_m$.

For $\mathtt{x} \in base$ suppose that

$$\mathtt{x}\lambda \in \mathrm{suf}(Cat) \text{ with } \lambda \in Cat^*, \text{ i.e. } \lambda \in licensees^*,$$
$$s\kappa \in \mathrm{suf}(Cat) \text{ with } s \in \{{}^{=}\mathtt{x}, {}^{=}\mathtt{X}, \mathtt{X}^{=}\} \text{ and } \kappa \in Cat^*,$$
$$\nu_i, \xi_i \in \mathrm{suf}(-\mathtt{l}_i) \text{ for } 1 \leq i \leq m \text{ with } \nu_i = \epsilon \text{ or } \xi_i = \epsilon$$

such that for $1 \leq j \leq m$,

$$\nu_j = \xi_j = \epsilon \text{ if } \lambda = -\mathtt{l}_j \lambda' \text{ with } \lambda' \in Cat^*.$$

We choose b_0, $c_0 \in \{\mathtt{strong}, \mathtt{weak}\}$, b_i, $c_i \in \{\mathtt{overt}, \mathtt{covert}, \mathtt{true}, \mathtt{false}\}$ for $1 \leq i \leq m$, β_i, $\gamma_i \in \{1, \ldots, m\}^*$ for $0 \leq i \leq m$, and $u, v \in \{\mathtt{sim}, \mathtt{com}\}$ such that

$$U = ((s\kappa, b_0, \beta_0), (\nu_1, b_1, \beta_1), \ldots, (\nu_m, b_m, \beta_m), u) \in N,$$
$$V = ((\mathtt{x}\lambda, c_0, \gamma_0), (\xi_1, c_1, \gamma_1), \ldots, (\xi_m, c_m, \gamma_m), v) \in N,$$

and such that, additionally,

$$\text{if } s \in \{{}^{=}\mathtt{x}\} \text{ then } c_0 = \mathtt{weak},$$
$$\text{if } s \in \{{}^{=}\mathtt{X}, \mathtt{X}^{=}\} \text{ then } c_0 = \mathtt{strong} \text{ and } u = \mathtt{sim}.$$

Proceeding, if $\lambda = \epsilon$ we set $j = 0$ and take

$$T' = ((\kappa, b_0, \gamma_0\beta_0), \widehat{\mu}'_1, \ldots, \widehat{\mu}'_m, \mathtt{com}) \in N,$$

whereas, if $\lambda = -\mathtt{l}_j \lambda'$ for some $1 \leq j \leq m$ and $\lambda' \in Cat^*$ we take

$$T' = ((\kappa, b_0, j\beta_0), \widehat{\mu}'_1, \ldots, \widehat{\mu}'_m, \mathtt{com}) \in N,$$
$$T'' = ((\kappa, b_0, j\beta_0), \widehat{\mu}''_1, \ldots, \widehat{\mu}''_m, \mathtt{com}) \in N,$$

where for $1 \leq i \leq m$ we have

$$\widehat{\mu}'_i = \widehat{\mu}''_i = \begin{cases} (\nu_i, b_i, \beta_i) & \text{if } i \neq j \text{ and } \xi_i = \epsilon \\ (\xi_i, c_i, \gamma_i) & \text{if } i \neq j \text{ and } \xi_i \neq \epsilon \end{cases}$$

$$\left.\begin{aligned} \widehat{\mu}'_i &= (\lambda, \texttt{covert}, \gamma_0) \\ \widehat{\mu}''_i &= (\lambda, \texttt{overt}, \gamma_0) \end{aligned}\right\} \text{ if } i = j \text{ for } j \neq 0$$

Then, for $merge_{U,V}$ and $Merge_{U,V} \in F$ as defined below, we finally let

(r1) $T' \to merge_{U,V}\big(U, V\big) \in R$, and

(r2) $T'' \to Merge_{U,V}\big(U, V\big) \in R$ if $\lambda = -\mathtt{l}_j\lambda'$ for $1 \leq j \leq m$ and $\lambda' \in Cat^*$.

Take x and y to be the $m+2$–tuples $(x_H, x_0, x_1, \ldots, x_m)$ and $(y_H, y_0, y_1, \ldots, y_m)$ consisting of the variables $x_H, x_0, x_1, \ldots, x_m$ and $y_H, y_0, y_1, \ldots, y_m$, respectively.

The function $merge_{U,V} : (P^*)^{m+2} \times (P^*)^{m+2} \to (P^*)^{m+2}$ is defined by

$$(x, y) \mapsto (\widetilde{x}_H, \widetilde{x}_0, x_1y_1, \ldots, x_my_m)$$

$$\text{with} \begin{cases} \widetilde{x}_H = x_Hy_H \,,\ \widetilde{x}_0 = x_0y_0 & \text{in case } s = {}^{=}\mathtt{x} \,,\ u = \mathtt{sim} \\ \widetilde{x}_H = x_Hy_H \,,\ \widetilde{x}_0 = y_0x_0 & \text{in case } s = {}^{=}\mathtt{x} \,,\ u = \mathtt{com} \\ \widetilde{x}_H = x_Hy_H \,,\ \widetilde{x}_0 = x_0y_0 & \text{in case } s = {}^{=}\mathtt{X} \,,\ b_0 = \mathtt{strong} \\ \widetilde{x}_H = x_H \,,\ \widetilde{x}_0 = x_0y_Hy_0 & \text{in case } s = {}^{=}\mathtt{X} \,,\ b_0 = \mathtt{weak} \\ \widetilde{x}_H = y_Hx_H \,,\ \widetilde{x}_0 = x_0y_0 & \text{in case } s = \mathtt{X}^{=} \,,\ b_0 = \mathtt{strong} \\ \widetilde{x}_H = x_H \,,\ \widetilde{x}_0 = y_Hx_0y_0 & \text{in case } s = \mathtt{X}^{=} \,,\ b_0 = \mathtt{weak} \end{cases}$$

The function $Merge_{U,V} : (P^*)^{m+2} \times (P^*)^{m+2} \to (P^*)^{m+2}$ is defined by

$$(x, y) \mapsto (\widetilde{x}_H, \widetilde{x}_0, x_1y_1, \ldots, x_{j-1}y_{j-1}, x_jy_jy_0, x_{j+1}y_{j+1}, \ldots, x_my_m)$$

$$\text{with} \begin{cases} \widetilde{x}_H = x_Hy_H \,,\ \widetilde{x}_0 = x_0 & \text{in case } s = {}^{=}\mathtt{x} \\ \widetilde{x}_H = x_Hy_H \,,\ \widetilde{x}_0 = x_0 & \text{in case } s = {}^{=}\mathtt{X} \,,\ b_0 = \mathtt{strong} \\ \widetilde{x}_H = x_H \,,\ \widetilde{x}_0 = x_0y_H & \text{in case } s = {}^{=}\mathtt{X} \,,\ b_0 = \mathtt{weak} \\ \widetilde{x}_H = y_Hx_H \,,\ \widetilde{x}_0 = x_0 & \text{in case } s = \mathtt{X}^{=} \,,\ b_0 = \mathtt{strong} \\ \widetilde{x}_H = x_H \,,\ \widetilde{x}_0 = y_Hx_0 & \text{in case } s = \mathtt{X}^{=} \,,\ b_0 = \mathtt{weak} \end{cases}$$

In order to illustrate the way in which G "does its job" concerning the operation *merge*, consider υ and $\phi \in RCL(G_{\mathrm{MG}})$ with respective head–labels $s\kappa\zeta$ and $\mathtt{x}\lambda\eta$ for some $\mathtt{x} \in base$, $s \in \{{}^{=}\mathtt{x}, {}^{=}\mathtt{X}, \mathtt{X}^{=}\}$, $\kappa, \lambda \in Cat^*$ and some ζ, $\eta \in P^*I^*$ such that $\tau = merge(\upsilon, \phi) \in RCL(G_{\mathrm{MG}})$. Assume U, $V \in N$ and $p_U = (\rho_H, \rho_0, \ldots, \rho_m)$, $p_V = (\sigma_H, \sigma_0, \ldots, \sigma_m) \in (P^*)^{m+2}$ to be such that (U, p_U) and (V, p_V) respectively correspond to υ and ϕ in the sense of Definition 4.1. Then U and V are as in (r1), and also as in (r2) in case $\lambda \neq \epsilon$.[12]

For T' as in (r1) and $p_{T'} = merge_{U,V}(p_U, p_V)$, $(T', p_{T'})$ corresponds to τ in any case. For T'' as in (r2) and $p_{T''} = Merge_{U,V}(p_U, p_V)$, also $(T'', p_{T''})$ corresponds to τ in case that $\lambda = -\mathtt{l}_j\lambda'$ for some $1 \leq j \leq m$ and $\lambda' \in Cat^*$. In the latter case, in terms of the MG G_{MG}, by canceling the category feature $\mathtt{x}$ from ϕ's head–label while merging υ and ϕ an expression τ_j that has licensee

[12] In particular, $\rho_j\sigma_j = \epsilon$ in case that $\lambda = -\mathtt{l}_j\lambda'$ for some $1 \leq j \leq m$ and $\lambda' \in Cat^*$.

$-\mathtt{l}_j$ becomes a proper subtree of τ. Up to the deletion of the instance of $\mathtt{x}$, τ_j is identical with ϕ. I.e. in particular the phonetic yield of both is identical. In a derivation creating a complete expression, τ_j must move to check its licensee at some later derivation step. In (r1) this later application of *move* is expected to be covert, coded in T' by $\widehat{\mu}'_j$ stating that the $j+2$–th component of $p_{T'}$ is empty. This chimes in with the definition of $merge_{U,V}$ according to which σ_0, the "non-extractable" part of the yield of ϕ (i.e. of τ_j) specified by V, is "frozen" within the 2nd component of $p_{T'}$, the "non-extractable" part of the yield of τ specified by T'. In (r2) the later application of *move* is expected to be overt, coded in T'' by $\widehat{\mu}''_j$. Here, applying $Merge_{U,V}$ to (p_U, p_V), σ_0 remains a part on its own as $j+2$–th component of $p_{T''}$, since $\rho_j\sigma_j = \epsilon$.

If $s \in \{{}^{=}\mathtt{X}, \mathtt{X}^{=}\}$ then ϕ is selected strongly and υ is simple. In this case $c_0 = \mathtt{strong}$, and therefore the (ordered) phonetic features σ of ϕ's head coincide with σ_H, the 1st component of p_V. Applying $Merge_{U,V}$ or $merge_{U,V}$ to the pair (p_U, p_V), σ_H will be incorporated into the selecting head υ, i.e. concatenated with the phonetic features ρ of υ "in the right manner." Note that in terms of the LCFRS G depending on whether the category feature of υ is expected to be selected strong or weak, i.e. whether $b_0 = \mathtt{strong}$ or $b_0 = \mathtt{weak}$, ρ is either ρ_H or ρ_0 according to (D3).[13] If $s = {}^{=}\mathtt{x}$ then ϕ is selected weakly. Thus, $c_0 = \mathtt{weak}$. Therefore, the phonetic features σ of ϕ's head are a substring of σ_0, the "non–extractable" part of the yield of ϕ, and $\sigma_H = \epsilon$.

Now, for some $1 \leq j \leq m$, suppose that

$$\nu_j \in \mathrm{suf}(-\mathtt{l}_j) \text{ with } \nu_j = -\mathtt{l}_j\lambda \text{ for some } \lambda \in licensees^*,$$
$$l\kappa \in \mathrm{suf}(Cat) \text{ with } l \in \{+\mathtt{l}_j, +\mathtt{L}_j\} \text{ and } \kappa \in Cat^*,$$
$$\nu_i \in \mathrm{suf}(-\mathtt{l}_i) \text{ for } 1 \leq i \leq m \text{ with } i \neq j$$

such that for $1 \leq k \leq m$ with $k \neq j$,

$$\nu_k = \epsilon \text{ if } \lambda = -\mathtt{l}_k\lambda' \text{ with } \lambda' \in Cat^*.$$

Choose $b_0 \in \{\mathtt{strong}, \mathtt{weak}\}$, $b_i \in \{\mathtt{overt}, \mathtt{covert}, \mathtt{true}, \mathtt{false}\}$ for $1 \leq i \leq m$, and $\beta_i \in \{1, \ldots, m\}^*$ for $0 \leq i \leq m$ such that

$$U = ((l\kappa, b_0, \beta_0), (\nu_1, b_1, \beta_1) \ldots, (\nu_m, b_m, \beta_m), \mathtt{com}) \in N,$$

and such that, additionally,

$$\text{if } l = +\mathtt{L}_j \text{ then } b_j \in \{\mathtt{overt}, \mathtt{true}\},$$
$$\text{if } l = +\mathtt{l}_j \text{ then } b_i \in \{\mathtt{covert}, \mathtt{true}\} \text{ for } 1 \leq i \leq m \text{ with } j \lhd_U^* i.$$

If $\lambda = \epsilon$ we set $k = 0$ and take

$$T' = ((\kappa, b_0, \beta_j\beta), \widehat{\mu}'_1, \ldots, \ldots, \widehat{\mu}'_m, \mathtt{com}) \in N,$$

if $\lambda = -\mathtt{l}_k\lambda'$ for some $1 \leq k \leq m$ and $\lambda' \in Cat^*$ we take

[13] The existence of both possibilities is granted by the nonterminating rules (cf. (r5)).

$$T' = ((\kappa, b_0, k\beta), \widehat{\mu}'_1, \dots, \widehat{\mu}'_m, \texttt{com}) \in N \text{ in general, and}$$
$$T'' = ((\kappa, b_0, k\beta), \widehat{\mu}''_1, \dots, \widehat{\mu}''_m, \texttt{com}) \in N \text{ in case that } b_j = \texttt{overt}.$$

Here, $\beta = \zeta_0\eta_0$ if $0 \triangleleft_U j$, where $\zeta_0, \eta_0 \in \{1, \dots, m\}^*$ with $\beta_0 = \zeta_0 j \eta_0$, and $\beta = \beta_0$ otherwise. Further, if $b_j = \texttt{overt}$ then for $1 \leq i \leq m$ we have

$$\left.\begin{aligned} \widehat{\mu}'_i &= (\lambda, \texttt{covert}, \beta_j) \\ \widehat{\mu}''_i &= (\lambda, \texttt{overt}, \beta_j) \end{aligned}\right\} \text{ if } i = k$$

$$\widehat{\mu}'_i = \widehat{\mu}''_i = \begin{cases} (\epsilon, \texttt{false}, \epsilon) & \text{if } i = j \text{ and } j \neq k \\ (\nu_i, b_i, \zeta_i\eta_i) & \text{if } i \triangleleft_U j, \text{ where } \zeta_i, \eta_i \in \{1, \dots, m\}^* \text{ with } \beta_i = \zeta_i j \eta_i \\ (\nu_i, b_i, \beta_i) & \text{otherwise} \end{cases}$$

and, if $b_j \in \{\texttt{covert}, \texttt{true}\}$ then for $1 \leq i \leq m$ we have

$$\widehat{\mu}'_i = \begin{cases} (\lambda, \texttt{true}, \beta_j) & \text{if } i = k \\ (\epsilon, \texttt{false}, \epsilon) & \text{if } i = j \text{ and } j \neq k \\ (\nu_i, \texttt{true}, \beta_i) & \text{if } j \triangleleft_U^+ i \\ (\nu_i, b_i, \zeta_i\eta_i) & \text{if } i \triangleleft_U j, \text{ where } \zeta_i, \eta_i \in \{1, \dots, m\}^* \text{ with } \beta_i = \zeta_i j \eta_i \\ (\nu_i, b_i, \beta_i) & \text{otherwise} \end{cases}$$

Now, for the functions $move_U$, $Move_U \in F$ as defined below we let

(r3) $T' \to move_U\big(U\big) \in R$ in any case, and

(r4) $T'' \to Move_U\big(U\big) \in R$ if $b_j = \texttt{overt}$, $\lambda = -\texttt{l}_k\lambda'$ for $1 \leq k \leq m$, $\lambda' \in Cat^*$.

Again let x denote the $m{+}2$–tuple $(x_H, x_0, x_1, \dots, x_m)$ consisting of the variables $x_H, x_0, x_1, \dots, x_m$.

The function $move_U : (P^*)^{m+2} \to (P^*)^{m+2}$ is defined by

$$x \mapsto (x_H, x_j x_0, x_1, \dots, x_{j-1}, \epsilon, x_{j+1}, \dots, x_m)$$

The function $Move_U : (P^*)^{m+2} \to (P^*)^{m+2}$ is defined by

$$x \mapsto \begin{cases} (x_H, x_0, \dots, x_{j-1}, x_j, x_{j+1}, \dots, x_m) \text{ for } k = j \\ (x_H, x_0, \dots, x_{k-1}, x_j x_k, x_{k+1}, \dots, x_{j-1}, \epsilon, x_{j+1}, \dots, x_m) \text{ for } k < j \\ (x_H, x_0, \dots, x_{j-1}, \epsilon, x_{j+1}, \dots, x_{k-1}, x_j x_k, x_{k+1}, \dots, x_m) \text{ for } k > j \end{cases}$$

Let us briefly discuss how the operation *move* is mimicked by G. Consider τ and $\upsilon \in RCL(G_{\mathrm{MG}})$ for which $\tau = move(\upsilon)$. Hence υ has head–label $l\kappa\zeta$ and a maximal subtree ϕ with head–label $-\texttt{l}_j\lambda\eta$ for some $1 \leq j \leq m$, $l \in \{+\texttt{L}_j, +\texttt{l}_j\}$, $\kappa, \lambda \in Cat^*$ and ζ, $\eta \in P^*I^*$. For $1 \leq i \leq m$ let, if existing, υ_i be the maximal subtree of υ that has licensee $-\texttt{l}_i$, otherwise let υ_i be the simple expression labeled ϵ. Thus, $\phi = \upsilon_j$. Take $U \in N$ and $p_U = (\rho_H, \rho_0, \dots, \rho_m) \in (P^*)^{m+2}$ to be such that (U, p_U) corresponds to υ according to Definition 4.1. Then U is as in (r3), and also as in (r4) in case $\lambda \neq \epsilon$ and $b_j = \texttt{overt}$.

In case that $l = +\mathtt{l}_j$ covert movement applies in terms of the MG G_{MG}. Looking at (D4) we see that in terms of the LCFRS G, by the respective $U \in N$ we ensure that $\rho_j = \epsilon$, but also that $\rho_i = \epsilon$ for each $1 \leq i \leq m$ with $j \triangleleft_T^+ i$. I.e. for each υ_i that is a subtree of υ_j we demand that ρ_i, the "non–extractable" part of the yield of υ_i, is empty. As for the general MG–definition we must be aware of the linguistically rather pathological case that υ_j in fact "hosts" some proper subtree υ_i and at some later derivation step overt movement will apply to υ_i but with empty phonetic yield. This becomes possible since υ_j is moved covertly before υ_i has been extracted such that υ_i's yield gets "frozen" within υ_j's yield which is "left behind."[14] After having lost the phonetic features this way, in terms of the LCFRS G the component b_i gets the value $\mathtt{true}$, which triggers equal behavior w.r.t. a strong licensor and its weak counterpart. This reflects the fact that in terms of the MG G_{MG} overt movement of a constituent with empty phonetic yield has the same effect as moving this constituent covertly (up to leaving behind a "totally empty" structure in the latter case).

For T' as in (r3) and $p_{T'} = \mathit{move}_U(p_U)$, $(T', p_{T'})$ corresponds to τ in any case. For T'' as in (r4) and $p_{T''} = \mathit{Move}_U(p_U)$, also $(T'', p_{T''})$ corresponds to τ in case that $b_j = \mathtt{overt}$ and $\lambda = -\mathtt{l}_k\lambda'$ for some $1 \leq k \leq m$ and $\lambda' \in \mathit{Cat}^*$. Whenever $\lambda = -\mathtt{l}_k\lambda'$ for some $1 \leq k \leq m$ and $\lambda' \in \mathit{Cat}^*$, in terms of the MG G_{MG} an expression τ_k that has licensee $-\mathtt{l}_k$ becomes a proper subtree of τ by canceling the licensee $-\mathtt{l}_j$ from ϕ's head–label while moving ϕ to specifier position of υ. In order to derive a complete expression, the licensee of τ_k has to be canceled by moving τ_k at some later derivation step. Thus, we again can distinguish two general possibilities:[15] Of course, the corresponding instance of $-\mathtt{l}_k$ can be checked overtly or covertly. But, here we pay somewhat more attention than in the analogous "merge–case," since it might be that υ_k has already "lost" its phonetic yield by a particular application of covert movement at some earlier derivation step (see above). According to (D4), only in case that $b_j = \mathtt{overt}$ the corresponding component ρ_j of p_U may include some non-empty phonetic material, and only in this case we have to state explicitly two cases (r3) and (r4), analogous to (r1) and (r2) in the "merge–case." The later application of *move* is "anticipated" as being covert in (r3), and as being overt in (r4).

Terminating rules: Let $\kappa\pi\iota \in \mathit{Lex}$ for some $\kappa \in \mathit{Cat}^*$, $\pi \in P^*$ and $\iota \in I^*$. Then, consider $a_0 \in \{\mathtt{strong}, \mathtt{weak}\}$ and $\pi_H, \pi_0 \in \{\pi, \epsilon\}$ with $\pi_H \neq \pi_0$ such that $\pi_0 = \pi$ iff $a_0 = \mathtt{weak}$. We define two terminating rules by

(r5) $T \to p_T \in R$

with $T = ((\kappa, a_0, \epsilon), \widehat{\nu}_1, \ldots, \widehat{\nu}_m, \mathtt{sim}) \in N$ and $p_T = (\pi_H, \pi_0, \epsilon, \ldots, \epsilon) \in (P^*)^{m+2}$, where $\widehat{\nu}_i = (\epsilon, \mathtt{false}, \epsilon)$ for $1 \leq i \leq m$.

[14] This case is exemplified by the MG G_{con}, where P is $\{/\mathtt{e}_1/, /\mathtt{e}_2/, /\mathtt{e}_3/\}$, I is $\emptyset$, *base* is $\{\mathtt{c}, \mathtt{a}_1, \mathtt{a}_2, \mathtt{a}_3\}$, *select* is $\{{=}\mathtt{a}_1, {=}\mathtt{a}_2, {=}\mathtt{a}_3\}$, *licensor* is $\{+\mathtt{B}_1, +\mathtt{b}_2\}$, *licensees* is $\{-\mathtt{b}_1, -\mathtt{b}_2\}$, *Lex* consists of $\mathtt{a}_1{-}\mathtt{b}_1/\mathtt{e}_1/$, ${=}\mathtt{a}_1\mathtt{a}_2{-}\mathtt{b}_2/\mathtt{e}_2/$, ${=}\mathtt{a}_2{+}\mathtt{b}_2\mathtt{a}_3/\mathtt{e}_3/$ and ${=}\mathtt{a}_3{+}\mathtt{B}_1\mathtt{c}$. The language $L(G_{\mathrm{con}})$ derivable by G_{con} consists of the single string $/\mathtt{e}_3//\mathtt{e}_2//\mathtt{e}_1/$.

[15] Like in the case when a subtree with licensee $-\mathtt{l}_j$ is introduced applying *merge*.

We will continue by proving the weak equivalence of G and G_{MG}. In order to finally do so, we show two propositions in advance.

Proposition 4.3. *Consider $\tau \in RCL(G_{\mathrm{MG}})$. Let $q_0 \in \{\mathtt{strong}, \mathtt{weak}\}$, and let $q_i \in \{\mathtt{overt}, \mathtt{covert}\}$ for $1 \leq i \leq m$. Then there is some $T = (\widehat{\mu}_0, \ldots, \widehat{\mu}_m, t) \in N$ with $t \in \{\mathtt{sim}, \mathtt{com}\}$ and $\widehat{\mu}_i = (\mu_i, a_i, \alpha_i)$ for $0 \leq i \leq m$ as in (n1)–(n5), and there is some $p_T \in (P^*)^{m+2}$ with $p_T \in L_G(T)$ such that (a) and (b) hold.*

(a) (T, p_T) corresponds to τ according to Definition 4.1.

(b) $a_0 = q_0$ and $a_i \in \{q_i, \mathtt{true}\}$ for $1 \leq i \leq m$ in case $\mu_i \neq \epsilon$.

Proof. We have $RCL(G_{\mathrm{MG}}) = \bigcup_{k \in \mathbb{N}} RCL^k(G_{\mathrm{MG}})$ and $L_G(T) = \bigcup_{k \in \mathbb{N}} L_G^k(T)$ for $T \in N$. Showing (4.3_k) by induction on $k \in \mathbb{N}$ we will prove the proposition.

(4.3_k) If $q_0 \in \{\mathtt{strong}, \mathtt{weak}\}$ and $q_i \in \{\mathtt{overt}, \mathtt{covert}\}$ for $1 \leq i \leq m$ then $\tau \in RCL^k(G_{\mathrm{MG}})$ implies that there are $T = (\widehat{\mu}_0, \ldots, \widehat{\mu}_m, t) \in N$ and $p_T \in (P^*)^{m+2}$ with $p_T \in L_G^k(T)$ fulfilling (a) and (b).

Since $RCL^0(G_{\mathrm{MG}}) = Lex$, (4.3_0) holds according to (r5). Considering the induction step, let $\tau \in RCL^{k+1}(G_{\mathrm{MG}})$. There is nothing to show if $\tau \in RCL^k(G_{\mathrm{MG}})$. Otherwise, one of two general cases arises.

Either, there are υ and $\phi \in RCL^k(G_{\mathrm{MG}})$ with respective head–labels $s\kappa\zeta$ and $\mathtt{x}\lambda\eta$ for some $\mathtt{x} \in base$, $s \in \{{=}\mathtt{x}, {=}\mathtt{X}, \mathtt{X}{=}\}$, $\kappa, \lambda \in Cat^*$ and ζ, $\eta \in P^*I^*$ such that $\tau = merge(\upsilon, \phi)$ holds. Let $b_0 = q_0$, let $c_0 = \mathtt{strong}$ iff $s \in \{{=}\mathtt{X}, \mathtt{X}{=}\}$. Now choose

$$U = ((s\kappa, b_0, \beta_0), (\nu_1, b_1, \beta_1), \ldots, (\nu_m, b_m, \beta_m), u) \in N,$$
$$V = ((\mathtt{x}\lambda, c_0, \gamma_0), (\xi_1, c_1, \gamma_1), \ldots, (\xi_m, c_m, \gamma_m), v) \in N$$

and p_U, $p_V \in (P^*)^{m+2}$ such that $p_U \in L_G^k(U)$, $p_V \in L_G^k(V)$, and such that (U, p_U) and (V, p_V) correspond to υ and ϕ, respectively. Here $u, v \in \{\mathtt{sim}, \mathtt{com}\}$, $\nu_i, \xi_i \in \mathrm{suf}(-\mathtt{l}_i)$, $b_i, c_i \in \{\mathtt{overt}, \mathtt{covert}, \mathtt{true}, \mathtt{false}\}$ for $1 \leq i \leq m$, and $\beta_i, \gamma_i \in \{1, \ldots, m\}$ for $0 \leq i \leq m$. In particular, each ν_i and ξ_i for $1 \leq i \leq m$ is unique. By induction hypothesis U, V and p_U, p_V not only exist, but for $1 \leq i \leq m$ they can also be chosen such that $b_i \in \{q_i, \mathtt{true}\}$ for $\nu_i \neq \epsilon$, and $c_i \in \{q_i, \mathtt{true}\}$ for $\xi_i \neq \epsilon$.

Recalling that *merge* is defined for the pair (υ, ϕ), we conclude that $u = \mathtt{sim}$ if $s \in \{{=}\mathtt{X}, \mathtt{X}{=}\}$. Because, $merge(\upsilon, \phi) \in RCL(G_{\mathrm{MG}})$ we also have $\nu_i, \xi_i \in \mathrm{suf}(-\mathtt{l}_i)$ for $1 \leq i \leq m$ with $\nu_i = \epsilon$ or $\xi_i = \epsilon$ such that $\nu_i = \xi_i = \epsilon$ if $\lambda = -\mathtt{l}_i\lambda'$ with $\lambda' \in Cat^*$. Therefore, U and V are as in (r1) in any case, and also as in (r2) in case that $\lambda \neq \epsilon$. Hence (r1') is true in any case, and (r2') in case $\lambda \neq \epsilon$.

(r1') $T' \to merge_{U,V}(U, V) \in R$ and $p_{T'} = merge_{U,V}(p_U, p_V) \in L_G^{k+1}(T')$,

(r2') $T'' \to Merge_{U,V}(U, V) \in R$ and $p_{T''} = Merge_{U,V}(p_U, p_V) \in L_G^{k+1}(T'')$

with $T' \in N$ and $merge_{U,V} \in F$ as in (r1), $T'' \in N$ and $Merge_{U,V} \in F$ as in (r2).

Let $T = T''$ and $p_T = p_{T''}$ in case that $q_j = \mathtt{overt}$ and $\lambda = -\mathtt{l}_j\lambda'$ for some $1 \leq j \leq m$ and $\lambda' \in Cat^*$. Otherwise let $T = T'$ and $p_T = p_{T''}$. Comparing the definition of $merge \in \mathcal{F}$ to the definitions of T and $merge_{U,V}$ or $Merge_{U,V}$, respectively, we see that (T, p_T) corresponds to $\tau = \mathit{merge}(\upsilon, \phi)$, and that T also satisfies the conditions imposed by (b).

The second general case provides an $\upsilon \in RCL^k(G_{\mathrm{MG}})$ for which $\tau = \mathit{move}(\upsilon)$. Thus, υ has head–label $l\kappa\zeta$ and a maximal subtree ϕ with head–label $-\mathtt{l}_j\lambda\eta$ for some $1 \leq j \leq m$, $l \in \{+\mathtt{L}_j, +\mathtt{l}_j\}$, $\kappa, \lambda \in \mathit{Cat}^*$ and ζ, $\eta \in P^*I^*$. For $b_0 = q_0$, by induction hypothesis we can fix existing

$$U = ((l\kappa, b_0, \beta_0), (\nu_1, b_1, \beta_1), \ldots, (\nu_m, b_m, \beta_m), \mathtt{com}) \in N,$$

and $p_U \in (P^*)^{m+2}$ with $p_U \in L^k_G(U)$ such that (U, p_U) corresponds to υ. Again we have $\nu_i \in \mathrm{suf}(-\mathtt{l}_i)$, $b_i \in \{\mathtt{overt}, \mathtt{covert}, \mathtt{true}, \mathtt{false}\}$ for $1 \leq i \leq m$, and $\beta_i \in \{1, \ldots, m\}$ for $0 \leq i \leq m$.[16] By induction hypothesis, for all $1 \leq i \leq m$ with $\mu_i \neq \epsilon$ we can choose U even such that $b_j \in \{\mathtt{overt}, \mathtt{true}\}$ and $b_i \in \{q_i, \mathtt{true}\}$ for $i \neq j$ in case $l = +\mathtt{L}_j$, and such that $b_j \in \{\mathtt{covert}, \mathtt{true}\}$, $b_i \in \{\mathtt{covert}, \mathtt{true}\}$ for $j \triangleleft^+_T i$ and $b_i \in \{q_i, \mathtt{true}\}$ in case $l = +\mathtt{l}_j$. Because $\mathit{move}(\upsilon) \in RCL(G_{\mathrm{MG}})$, we conclude that (r3') holds in any case, and (r4') in case that $\lambda \neq \epsilon$ and $b_j = \mathtt{overt}$.

(r3') $T' \to \mathit{move}_U(U) \in R$ and $p_{T'} = \mathit{move}_U(p_U) \in L^{k+1}_G(T')$

(r4') $T'' \to \mathit{Move}_U(U) \in R$ and $p_{T''} = \mathit{Move}_U(p_U) \in L^{k+1}_G(T'')$

with $T' \in N$ and $\mathit{move}_U \in F$ as in (r3), $T'' \in N$ and $\mathit{Move}_U \in F$ as in (r4).

Let $T = T''$ and $p_T = p_{T''}$ in case that $b_j = q_k = \mathtt{overt}$ and $\lambda = -\mathtt{l}_k\lambda'$ for some $1 \leq k \leq m$ and $\lambda' \in \mathit{Cat}^*$. Otherwise let $T = T'$ and $p_T = p_{T'}$. Looking at the definition of $\mathit{move} \in \mathcal{F}$ and the definitions of T and $\mathit{move}_{U,V}$ or $\mathit{Move}_{U,V}$, respectively, we see that (T, p_T) corresponds to τ, and that also (b) is true. $\square$

Let $T \in N$ and $p_T \in (P^*)^{m+2}$ be such that (a) and (b) of Proposition 4.3 are true w.r.t. given $\tau \in RCL(G_{\mathrm{MG}})$, $q_0 \in \{\mathtt{strong}, \mathtt{weak}\}$ and $q_i \in \{\mathtt{overt}, \mathtt{covert}\}$ for $1 \leq i \leq m$. Note that this does not automatically imply that $p_T \in L_G(T)$.

Proposition 4.4. *If p_T is a T–phrase in G, i.e. if $p_T \in L_G(T)$ for some $T \in N$ with $T \neq S$ and $p_T \in (P^*)^{m+2}$, then there is some $\tau \in RCL(G_{\mathrm{MG}})$ such that (T, p_T) corresponds to τ according to Definition 4.1.*

Proof. Recalling again that $RCL(G_{\mathrm{MG}}) = \bigcup_{k \in \mathbb{N}} RCL^k(G_{\mathrm{MG}})$ holds as well as $L_G(T) = \bigcup_{k \in \mathbb{N}} L^k_G(T)$, we also prove this proposition by induction on $k \in \mathbb{N}$.

(4.4_k) If $p_T \in L^k_G(T)$ then (T, p_T) corresponds to some $\tau \in RCL^k(G_{\mathrm{MG}})$.

Since $\mathit{Lex} = RCL^0(G_{\mathrm{MG}})$, (4.4_0) holds according to (r5). Considering the induction step, suppose that (4.4_k) is true for $k \in \mathbb{N}$. The crucial case arises from $p_T \in L^{k+1}_G(T) \setminus L^k_G(T)$ dividing into two general possibilities.

Either, U, $V \in N$ and p_U, $p_V \in (P^*)^{m+2}$ exist with $p_U \in L^k_G(U)$, $p_V \in L^k_G(V)$. U and V fulfill the restrictions applying in (r1) such that (r1") is true for $T' \in N$ and $\mathit{merge}_{U,V} \in F$ as in (r1), or U and V even satisfy the restrictions applying in (r2) such that (r2") is true for $T'' \in N$ and $\mathit{Merge}_{U,V} \in F$ as in (r2).

[16] Recall that each ν_i for $0 \leq i \leq m$ and each β_i for $0 \leq i \leq m$ is unique.

(r1”) $T \to merge_{U,V}(U,V) \in R$, $p_T = merge_{U,V}(p_U, p_V)$ and $T = T'$

(r2”) $T \to Merge_{U,V}(U,V) \in R$, $p_T = Merge_{U,V}(p_U, p_V)$ and $T = T''$

Then, by induction hypothesis there are υ and $\phi \in RCL^k(G_{\mathrm{MG}})$ such that (U, p_U) and (V, p_V) respectively correspond to υ and ϕ in the sense of Definition 4.1. Recall the restrictions that apply to U and V in (r1) or (r2), respectively. Because of these restrictions we may conclude that $\tau = merge(\upsilon, \phi)$ is not only defined according to (me), but also in $RCL^{k+1}(G_{\mathrm{MG}})$ according to (R2). Since (r1”) or (r2”) is true, we refer to the respective definitions of T' and $merge_{U,V}$ or T'' and $Merge_{U,V}$ to see that (T, p_T) corresponds to τ.

Secondly, $U \in N$ and $p_U \in (P^*)^{m+2}$ may exist with $p_U \in L^k_G(U)$. The restrictions given with (r3) apply to U and (r3”) holds for T' and $move_U \in F$ as in (r3), or even the restrictions given with (r4) apply to U and (r4”) holds for T'' and $Move_U \in F$ as in (r4).

(r3”) $T \to move_U(U) \in R$, $p_T = move_U(p_U)$ and $T = T'$

(r4”) $T \to Move_U(U) \in R$, $p_T = Move_U(p_U)$ and $T = T''$

Here, by hypothesis there is an $\upsilon \in RCL^k(G_{\mathrm{MG}})$ such that (U, p_U) corresponds to υ in the sense of Definition 4.1. Similar as for (r1”) and (r2”), in cases (r3”) and (r4”) it is straightforward to show that $move \in \mathcal{F}$ is defined for υ, and that (T, p_T) corresponds to $\tau = move(\upsilon) \in RCL^{k+1}(G_{\mathrm{MG}})$. □

Corollary 4.5. *$\pi \in L(G)$ iff $\pi \in L(G_{\mathrm{MG}})$ for each $\pi \in P^*$.*

Proof. As for the “if”–part consider complete $\tau \in CL(G_{\mathrm{MG}})$ with phonetic yield $\pi \in P^*$. Let $T = (\widehat{\mu}_0, \ldots, \widehat{\mu}_m, t) \in N$ with $t \in \{\texttt{sim}, \texttt{com}\}$ and $\widehat{\mu}_i = (\mu_i, a_i, \alpha_i)$ for $0 \leq i \leq m$ as in (n1)–(n5), let $p_T = (\pi_H, \pi_0, \ldots, \pi_m) \in (P^*)^{m+2}$. Assume that (T, p_T) corresponds to τ according to (D1)–(D4). By Proposition 4.3 these T and p_T exist even such that $p_T \in L_G(T)$ and $a_0 = \texttt{weak}$. Since τ is complete, $\widehat{\mu}_0 = (\texttt{c}, \texttt{weak}, \epsilon)$ and $\widehat{\mu}_i = (\epsilon, \texttt{false}, \epsilon)$ for $1 \leq i \leq m$ by (D1), and therefore $\pi_1 = \ldots = \pi_m = \epsilon$ by (D4). Moreover, τ’s phonetic head–features are “at the right place,” i.e. $\pi_H = \epsilon$ and $\pi_0 = \pi$ by (D3). Looking at (r0) and (L2), we conclude that $\pi \in L_G(S) = L(G)$.

To prove the “only if”–part, we start with some $\pi \in L(G) = L_G(S)$. The definition of R yields that each rule applying to S is of the form (r0). Thus, according to (L2) there is some $p_T = (\pi_H, \pi_0, \ldots, \pi_m) \in (P^*)^{m+2}$ such that $p_T \in L_G(T)$ and $\pi = con(p_T)$ for $T \in N$ as in (r0). (T, p_T) corresponds to some $\tau \in RCL(G_{\mathrm{MG}})$ by Proposition 4.4. This τ is complete by (D1), π is the yield of τ, since $\pi_H = \pi_1 = \ldots = \pi_m = \epsilon$ and $\pi_0 = \pi$ by (D3) and (D4). □

Consider the $m+2$–LCFRS G as constructed above for a given MG G_{MG} whose set of licensees has cardinality $m \in \mathbb{N}$.

If all licensors in G_{MG} are strong, i.e. only overt movement is available, we do not have to define productions of the form (r1) and (r3) in case $\lambda \neq \epsilon$ for the corresponding $\lambda \in licensees^*$. More concretely, whenever in terms of the MG G_{MG} a subtree that has licensee $-\texttt{x}$ arises from applying *merge* or *move*, in terms of the LCFRS G we do not have to predict the case that this licensee will be canceled by "covert movement." Moreover, according to (D2), the structural relation of any two subtrees with different licensees is of interest only in (r3) for $\lambda \neq \epsilon$. Since productions of this kind are of no use at all, assuming all licensors in G_{MG} are strong, each $\widehat{\mu}_i = (\mu_i, a_i, \alpha_i)$ of some $T = (\widehat{\mu}_0, \ldots, \widehat{\mu}_m, t) \in N$ according to (n1)–(n5) can be reduced to its 1st component μ_i without loosing any "necessary information." This means that expressions from $RCL(G_{\mathrm{MG}})$ in terms of the LCFRS G have to be distinguished only w.r.t. the partition $\mathcal{P}$ induced by $\mathrm{suf}(Cat) \times \mathrm{suf}(-\mathtt{l}_1) \times \ldots \times \mathrm{suf}(-\mathtt{l}_m) \times \{\mathtt{sim}, \mathtt{com}\}$.

In case that all selection features in G_{MG} are weak, G is reducible even to an $m+1$–LCFRS. This is due to the fact that the 1st component of any $p_T \in (P^*)^{m+2}$ appearing in some complete derivation in G_{MG} is necessarily empty in this case. Therefore, if additionally $m = 0$, G_{MG} is a CFG. Vice versa, each CFG is weakly equivalent to some MG of this kind. This can be verified rather straightforwardly e.g. by starting with a CFG in Chomsky normal form.

5 A Hierarchy of MGs

Several well–known grammar types constitute a subclass of MCSGs. There are a.o. the two classes of head grammars (HGs) and TAGs as well as their generalized extensions, the classes of LCFRSs and multicomponent TAGs (MCTAGs), respectively.[17] Like HGs and TAGs, LCFRSs and MCTAGs are weakly equivalent. LCFRSs and MCTAGs are the union of an infinite hierarchy of grammar classes, the respective hierarchy of m–LCFRSs and m–TAGs ($m \in \mathbb{N} \setminus \{0\}$). It is known that each m–LCFRL is an m–TAL, a language derivable by some m–TAG, and that each m–TAL is an $2m$–LCFRL (cf. [9]). We can introduce an infinite hierarchy on the MG–class, as well.

Definition 5.1. For each $m \in \mathbb{N}$ an MG $G = (V, Cat, Lex, \mathcal{F})$ according to Definition 3.2 is an *m–minimalist grammar (m–MG)* if the cardinality of *licensees* is at most m. Then, the ML derivable by G is an *m–minimalist language (m–ML)*.

Let $m \in \mathbb{N}$. It is clear that each m–ML is also an $m+1$–ML.

In Sect. 4 we have shown that each m–ML is an $m+2$–LCFRL. This result can be strengthened for $m = 0$, since the inclusion of 1–TALs within 2–LCFRLs is known to be proper (cf. [5]). Due to its "restricted type," the 2–LCFRS that we have constructed for a given 0–MG can be transformed to a weakly equivalent 1–TAG. Thus, each 0–ML, each language whose realization plainly relies on the "extended" merging–type allowing for overt head movement, is even a 1–TAL, a tree adjoining language. Indeed the class of 0–MLs is a proper extension of the class of CFLs. Referring to the rather categorial type logical approach of [1], [6] presents a 0–MG that derives the copy language $\{ww \mid w \in \{\mathtt{1}, \mathtt{2}\}^*\}$.

[17] We define an MCTAG as in [9] and call it an m–TAG if derived sequences of auxiliary trees can be (simultaneously) adjoined to elementary tree–sequences of length at most $m \in \mathbb{N} \setminus \{0\}$. Then, 1–TAGs are TAGs in the usual sense, and vice versa.

Generalizing Example 3.3, for $m \in \mathbb{N}$ we consider the m–MG G_m with $I = \emptyset$, $P = \{/\mathtt{a}_i/ \mid 1 \leq i \leq m\}$ and *base* $= \{\mathtt{c}\} \cup \{\mathtt{b}_i, \mathtt{c}_i, \mathtt{d}_i \mid 1 \leq i \leq m\}$, while *select* $= \{{=}\mathtt{b}_i, {=}\mathtt{c}_i, {=}\mathtt{d}_i \mid 1 \leq i \leq m\}$, *licensees* $= \{-\mathtt{l}_i \mid 1 \leq i \leq m\}$ and *licensors* $= \{+\mathtt{L}_i \mid 1 \leq i \leq m\}$. *Lex* consists of the simple expressions $\mathtt{c}$ and $\mathtt{b}_1{-}\mathtt{l}_1/\mathtt{a}_m/$, further ${=}\mathtt{b}_i\mathtt{b}_{i+1}{-}\mathtt{l}_{i+1}/\mathtt{a}_{m-i}/$, ${=}\mathtt{c}_i{+}\mathtt{L}_{i+1}\mathtt{c}_{i+1}{-}\mathtt{l}_{i+1}/\mathtt{a}_{m-i}/$ and ${=}\mathtt{d}_i{+}\mathtt{L}_{i+1}\mathtt{d}_{i+1}$ for $1 \leq i < m$, finally the 5 expressions ${=}\mathtt{b}_m{+}\mathtt{L}_1\mathtt{c}_1{-}\mathtt{l}_1/\mathtt{a}_m/$, ${=}\mathtt{b}_m{+}\mathtt{L}_1\mathtt{d}_1$, ${=}\mathtt{c}_m{+}\mathtt{L}_1\mathtt{c}_1{-}\mathtt{l}_1/\mathtt{a}_m/$, ${=}\mathtt{c}_m{+}\mathtt{L}_1\mathtt{d}_1$ and ${=}\mathtt{d}_m\mathtt{c}$. G_m derives the language $\{/\mathtt{a}_1/^n \ldots /\mathtt{a}_m/^n \mid n \in \mathbb{N}\}$. We omit a proof here, pointing to the rather "deterministic manner" in which expressions in G_m can be derived.

Proposition 5.2. *For each* $m \in \mathbb{N}$, $\{a_1^n \ldots a_m^n \mid n \in \mathbb{N}\}$ *is an* m*–ML.*

As shown in [5], for each $m \in \mathbb{N} \setminus \{0\}$, $\{a_1^n \ldots a_{2m}^n \mid n \in \mathbb{N}\}$ is an m–LCFRL, while $\{a_1^n \ldots a_{2m+1}^n \mid n \in \mathbb{N}\}$ is not. Because each m–ML is an $m+2$–LCFRL, we therefore conclude that the hierarchy of ML–classes is infinitely increasing, i.e. there is no $m_b \in \mathbb{N}$ such that for all $m \in \mathbb{N}$ each m–ML is also an m_b–ML.

6 Conclusion

We have shown that MGs as defined in [6] constitute a weakly equivalent (sub)-class of MCSGs as described in e.g. [3]. Thus, the result contributes to solve a problem that has remained open in [6]. Further, we have established an infinite hierarchy on the MG–class in relation to other hierarchies of MCSG-formalisms.

References

1. Thomas L. Cornell. A minimalist grammar for deriving the copy language. Report no. 79, Working papers of the SFB 340, University Tübingen, 1996.
2. Aravind K. Joshi. Tree adjoining grammars: How much context-sensitivity is required to provide reasonable structural descriptions? In D. Dowty, L. Karttunen, and A. Zwicky, editors, *Natural Language Parsing. Theoretical, Computational and Psychological Perspective*, pages 206–250. Cambridge University Press, New York, NY, 1985.
3. Aravind K. Joshi, K. Vijay-Shanker, and David J. Weir. The convergence of mildly context-sensitive grammar formalisms. In P. Sells, S. Shieber, and T. Wasow, editors, *Foundational Issues in Natural Language Processing*, pages 31–81. MIT Press, Cambridge, MA, 1991.
4. Carl J. Pollard. *Generalized Phrase Structure Grammars, Head Grammars, and Natural Language.* PhD thesis, Stanford University, Stanford, CA, 1984.
5. Hiroyuki Seki, Takashi Matsumura, Mamoru Fujii, and Tadao Kasami. On multiple context-free grammars. *Theoretical Computer Science*, 88:191–229, 1991.
6. Edward Stabler. Derivational minimalism. In C. Retoré, editor, *Logical Aspects of Computational Linguistics.* LNCS No.1328, pages 68–95. Springer, Berlin, 1997.
7. Edward Stabler. Acquiring languages with movement. *Syntax*, 1:72–97, 1998.
8. K. Vijay-Shanker, David J. Weir, and Aravind K. Joshi. Characterizing structural descriptions produced by various grammatical formalisms. In *25th Annual Meeting of the ACL (ACL '87)*, Stanford, CA, pages 104–111. ACL, 1987.
9. David J. Weir. *Characterizing Mildly Context-Sensitive Grammar Formalisms.* PhD thesis, University of Pennsylvania, Philadelphia, PA, 1988.