Noname manuscript No.
(will be inserted by the editor)

Scattered Context Grammars with One Non-Context-Free Production are Computationally Complete

Alexander Meduna, Zbyněk Křivka

Received: date / Accepted: date

Abstract This paper investigates the reduction of scattered context grammars with respect to the number of non-context-free productions. It proves that every recursively enumerable language is generated by a scattered context grammar that has no more than one non-context-free production. An open problem is formulated.

Keywords Scattered context grammars · Size reduction · The number of non-context-free productions · Computational completeness

1 Introduction

Formal language theory has always struggled to reduce their grammars as much as possible (for an overview of results concerning this reduction in terms of classical grammars, consult Sections 1.2 and 1.3 in Chapter 4 in [6]). As a central topic, this trend has studied how to reduce the number of grammatical components, such as nonterminals or productions, without disturbing the generative power. The present paper contributes to this trend in terms of scattered context grammars (for an overview of important results concerning the reduction in terms of scattered context grammars, consult Chapter 6 in [5]).

Concerning the number of nonterminals in scattered context grammars, two-nonterminal scattered context grammars are computationally complete—that is, they characterize the family of recursively enumerable languages (see [1]). On the other hand, one-nonterminal scattered context grammars are less powerful (see [3]).

Alexander Meduna
Department of Information Systems
Faculty of Information Technology
Brno University of Technology
Bozetechova 2, 612 66 Brno, the Czech Republic, Europe
E-mail: meduna@fit.vutbr.cz

The present paper reduces the number of non-context-free productions in scattered context grammars. In fact, it proves that scattered context grammars with a single non-context-free production are computationally complete. Of course, this statement represents the best possible result regarding this reductions because scattered context grammars without any non-context-free production only characterize the family of context-free languages.

2 Definitions

This paper assumes that the reader is familiar with the language theory (see [4]), including scattered context grammars (see [5]).

For a set, Q, card(Q) denotes the cardinality of Q. For an alphabet, V, V^* represents the free monoid generated by V under the operation of concatenation. The unit of V^* is denoted by ε. Set $V^+ = V^* - \{\varepsilon\}$; algebraically, V^+ is thus the free semigroup generated by V under the operation of concatenation. For $w \in V^*$, $|w|$ and Reverse(w) denote the length of w and the reversal of w, respectively. Furthermore, suffix(w) denotes the set of all suffixes of w, and prefix(w) denotes the set of all prefixes of w. For $L \subseteq V^*$, alph(L) denotes the set of all symbols occurring in a word of L. For $w \in V^*$ and $T \subseteq V$, occur(w, T) denotes the number of occurrences of symbols from T in w, and Erase(w, T) denotes the string obtained by removing all occurrences of symbols from T in w. For instance, occur$(abdabc, \{a, d\}) = 3$ and Erase$(abdabc, \{a, d\}) = bbc$. If $T = \{a\}$, where $a \in V$, we simplify occur$(w, \{a\})$ and Erase$(w, \{a\})$ to occur(w, a) and Erase(w, a), respectively.

A scattered context grammar is a quadruple, $G = (N, T, P, S)$, where N and T are alphabets such that $N \cap T = \emptyset$. Symbols in N are referred to as nonterminals while symbols in T are terminals. N contains S—the start symbol of G. P is a finite non-empty set of productions such that every $p \in P$ has the form

$$(A_1, A_2, \ldots, A_n) \to (x_1, x_2, \ldots, x_n),$$

where $n \geq 1$, and for all $i = 1, 2, \ldots, n$, $A_i \in N$ and $x_i \in (N \cup T)^*$. If each x_i satisfies $|x_i| \leq 1$, $i = 1, 2, \ldots, n$, then $(A_1, A_2, \ldots, A_n) \to (x_1, x_2, \ldots, x_n)$ is said to be simple. If $n = 1$, then $(A_1) \to (x_1)$ is referred to as a context-free production; for brevity, we hereafter write $A_1 \to x_1$ instead of $(A_1) \to (x_1)$. If for some $n \geq 1$, $(A_1, A_2, \ldots, A_n) \to (x_1, x_2, \ldots, x_n) \in P, v = u_1A_1u_2A_2 \ldots u_nA_nu_{n+1}$, and $w = u_1x_1u_2x_2 \ldots u_nx_nu_{n+1}$ with $u_i \in (N \cup T)^*$ for all $i = 1, 2, \ldots, n$, then v directly derives w in G, symbolically written as $v \Rightarrow w[(A_1, A_2, \ldots, A_n) \to (x_1, x_2, \ldots, x_n)]$ or, simply, $v \Rightarrow w$ in G. In the standard manner, extend $\Rightarrow$ to $\Rightarrow^n$, where $n \geq 0$; then, based on $\Rightarrow^n$, define $\Rightarrow^+$ and $\Rightarrow^*$. The set of all sentential forms of G, F(G), is defined as F$(G) = \{w \in (N \cup T)^* : S \Rightarrow^* w\}$. The language of G, $L(G)$, is defined as $L(G) = \mathrm{F}(G) \cap T^*$, so $L(G) = \{w \in T^* : S \Rightarrow^* w\}$. A derivation of the form $S \Rightarrow^* w$ with $w \in T^*$ is called a successful derivation.

A queue grammar (see [2]) is a sextuple, $Q = (V, T, W, F, s, P)$, where V and W are alphabets satisfying $V \cap W = s, T \subseteq V, F \subseteq W, s \in (V - T)(W - F)$, and $P \subseteq (V \times (W - F)) \times (V^* \times W)$ is a finite relation such that for every $a \in V$, there exists an element $(a, b, z, c) \in P$. If $u, v \in V^*W$ such that $u = arb$; $v = rzc$; $a \in V$; $r, z \in V^*$; $b, c \in W$; and $(a, b, z, c) \in P$, then $u \Rightarrow v[(a, b, z, c)]$ in G or, simply, $u \Rightarrow v$. In the standard manner, extend $\Rightarrow$ to $\Rightarrow^n$, where $n \geq 0$; then, based on $\Rightarrow^n$, define $\Rightarrow^+$ and $\Rightarrow^*$. The language of Q, $L(Q)$, is defined as $L(Q) = \{w \in T^* : s \Rightarrow *wf$ where $f \in F\}$. As a slight modification of the notion of a queue grammar, we define the notion of a left-extended queue grammar as a sextuple , $Q = (V, T, W, F, s, P)$, where V, T, W, F, and s have the same meaning as in a queue grammar. $P \subseteq (V \times (W - F)) \times (V^* \times W)$ is a finite relation (as opposed to an ordinary queue grammar, this definition does not require that for every $a \in V$, there exists an element $(a, b, z, c) \in P$). Furthermore, assume that $\# \notin V \cup W$. If $u, v \in V^*\{\#\}V^*W$ so that $u = w\#arb; v = wa\#rzc; a \in V; r, z, w \in V^*; b, c \in W$; and $(a, b, z, c) \in P$, then $u \Rightarrow v[(a, b, z, c)]$ in G or, simply, $u \Rightarrow v$. In the standard manner, extend $\Rightarrow$ to $\Rightarrow^n$, where $n \geq 0$; then, based on $\Rightarrow^n$, define $\Rightarrow^+$ and $\Rightarrow^*$. The language of $Q, L(Q)$, is defined as $L(Q) = \{v \in T^* : \#s \Rightarrow^* w\#vf$ for some $w \in V^*$ and $f \in F\}$. Less formally, during every step of a derivation, a left-extended queue grammar shifts the rewritten symbol over #; in this way, it records the derivation history, which plays a crucial role in the proof of Lemma 3 in the next section.

3 Results

This section demonstrates that for every recursively enumerable language, L, there exists a scattered context grammar, $G = (N, T, P, S)$, such that $L = L(G)$ and P contains a single non-context-free production of the form $(1, 2, 0, 3, 0, 2, 1) \Rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$.

Lemma 1 *For every recursively enumerable language, L, there exists a left-extended queue grammar, Q, satisfying $L(Q) = L$.*

Proof Recall that every recursively enumerable language is generated by a queue grammar (see [3]). Clearly, for every queue grammar, there exists an equivalent left-extended queue grammar. Thus, this lemma holds. □

Lemma 2 *Let H be a left-extended queue grammar. Then, there exists a left-extended queue grammar, $Q = (V, T, W, F, s, R)$, such that $L(H) = L(Q)$ and every $(a, b, x, c) \in R$ satisfies $a \in V - T, b \in W - F$, and $x \in ((V - T)^* \cup T^*)$.*

Proof Let $H = (\varsigma, T, \Omega, \Phi, \sigma, \Pi)$ be any left-extended queue grammar. Set $\Omega^{'} = \{q^{'} : q \in \Omega\}$, $\Omega^{''} = \{q^{''} : q \in \Omega\}$, and $\varsigma^{'} = \{a^{'} : a \in \varsigma\}$. Define the bijection α from Ω to $\Omega^{'}$ as $\alpha(q) = q^{'}$ for every $q \in \Omega$. Analogously, define the bijection β from Ω to $\Omega^{''}$ as $\beta(q) = q^{''}$ for every $q \in \Omega$. Finally, define the bijection δ from ς to $\varsigma^{'}$ as $\delta(a) = a^{'}$ for every $a \in \varsigma$. In the standard manner, extend δ so it is defined from ς^* to $(\varsigma^{'})^*$. Set

$$U = \{\langle y, p\rangle : y \in T^*, p \in \Omega, \text{ and } (a, q, xy, p) \in \Pi \text{ for some } a \in \varsigma, q \in \Omega, x \in \varsigma^*\}$$

Without any loss of generality, assume that $(\delta(\varsigma) \cup T \cup \alpha(\Omega) \cup \beta(\Omega) \cup U) \cap \{1, f\} = \emptyset. Set V = \delta(\varsigma) \cup \{1\} \cup T, W = \alpha(\Omega) \cup \beta(\Omega) \cup \{f\} \cup U, F = \{f\},$ and $s = \delta(a)\alpha(q)$. Define the left-extended queue grammar

$$Q = (V, T, W, F, s, R)$$

with R constructed in the following way:

I. if $(a, q, xy, p) \in \Pi$, where $a \in \varsigma; q \in \Omega - \Phi; x, y \in \varsigma^*$; and $p \in \Omega$, then add $(\delta(a), \alpha(q), \delta(x)\delta(y), \alpha(p))$ and $(\delta(a), \alpha(q), \delta(x)1\delta(y), \alpha(p))$ to R;
II. if $(a, q, xy, p) \in \Pi$, where $a \in \varsigma; q \in \Omega - \Phi; x \in \varsigma^*; y \in T^*, p \in \Omega(\langle y, p\rangle \in U)$, then add $(\delta(a), \alpha(q), \delta(x), \langle y, p\rangle)$ and $(1, \langle y, p\rangle, y, \beta(p))$ to R;
III. f $(a, q, x, p) \in \Pi$, where $a \in \varsigma; q \in \Omega - \Phi; x \in T^*$; and $p \in \Omega$, then add $(\delta(a), \beta(q), \delta(x), \beta(p))$ to R;
IV. f $(a, q, x, p) \in \Pi$, where $a \in \varsigma; q \in \Omega - \Phi; x \in T^*$; and $p \in \Phi$, then add $(\delta(a), \beta(q), x, f)$ to R (recall that $F = \{f\}$).

Clearly, for every $(a, b, x, c) \in R, a \in V - T, b \in W - F$, and $x \in ((V - T)^* \cup T^*)$. Leaving a rigorous proof that $L(H) = L(Q)$ to the reader, we next give its sketch.

To see that $L(H) \subseteq L(Q)$, consider any $v \in L(H)$. As $v \in L(H)$,

$$\#\sigma \Rightarrow^* w\#vt$$

in $H, w \in \varsigma^*, v \in T^*$, and $t \in \Phi$. Express $\#\sigma \Rightarrow^* w\#vt$ in H as

$$\#\sigma \Rightarrow^* u\#zq \Rightarrow ua\#xyp \Rightarrow^* w\#vt$$

where $a \in \varsigma, u, x \in \varsigma^*, y \in \text{prefix}(v), z = ax, w = uax$, and during $ua\#xyp \Rightarrow^* w\#vt$, only terminals are generated so that the resulting terminal string equals v. Q simulates $\#\sigma \Rightarrow^* u\#zq \Rightarrow ua\#xyp \Rightarrow^* w\#vt$ as follows. First, Q uses productions introduced in I to simulate $\#\sigma \Rightarrow^* u\#zq$. During this initial simulation, it once uses a production that generates 1 so that it can then simulate $u\#zq \Rightarrow ua\#xyp$ by making two derivation steps according to productions $(\delta(a), \alpha(q), \delta(x), \langle y, p\rangle)$ and $(1, \langle y, p\rangle, y, \beta(p))$ (see II). Notice that by using $(1, \langle y, p\rangle, y, \beta(p))$, Q produces y, which is a prefix of v. After the application of $(1, \langle y, p\rangle, y, \beta(p))$, Q simulates $ua\#xyp \Rightarrow^* w\#vt$ by using productions introduced in III followed by one application of a production constructed in IV, during which Q enters f and, thereby, completes the generation of v. Thus, $L(H) \subseteq L(Q)$.

To establish that $L(Q) \subseteq L(H)$, consider any $v \in L(Q)$. Since $v \in L(Q)$,

$$\#s \Rightarrow^* w\#vf$$

in Q, where $w \in V^*$ and $v \in T^*$. Examine I through IV. Observe that Q passes through states of $\alpha(\Omega), U, \beta(\Omega)$, and $\{f\}$ in this order so that it occurs several times in states of $\alpha(\Omega)$, once in a state of U, several times in $\beta(\Omega)$, and once in f. As a result, Q uses productions introduced in I, and during this initial part of derivation it precisely once uses a production that generates 1 so that it can subsequently make two consecutive derivation steps according to $(\delta(a), \alpha(q), \delta(x), \langle y, p\rangle)$ and $(1, \langle y, p\rangle, y, \beta(p))$ (see II). By using the latter, Q produces y, which is a prefix of v. After the application of $(1, \langle y, p\rangle, y, \beta(p))$, Q applies productions introduced in III, which always use states of $\beta(\Omega)$. Finally, it once applies a production constructed in IV to enter f and, thereby, complete the generation of v. To summarize these observations, we can express $\#s \Rightarrow^* w\#vf$ in Q as

$$\#s \Rightarrow^* u\#zq \Rightarrow ua\#xyp \Rightarrow^* w\#vf$$

where $a \in V, x \in V^*, y \in T^*, w = uax$ so that during $\#s \Rightarrow^* u\#zq$, Q uses productions introduced in I, then it applies $(1, \langle y, p\rangle, y, \beta(p))$ from II to make $u\#zq \Rightarrow ua\#xyp$, and finally it performs $ua\#xyp \Rightarrow^* w\#vf$ by several applications of productions introduced in III and one application of a production constructed in IV. At this point, by an examination of I through IV, we see that H makes

$$\#\sigma \Rightarrow^* u\#zq \Rightarrow ua\#xyp \Rightarrow^* w\#vt$$

with $t \in \Phi$, so $v \in L(H)$. Therefore, $L(H) \subseteq L(Q)$.

As $L(H) \subseteq L(Q)$ and $L(Q) \subseteq L(H), L(H) = L(Q)$. □

Lemma 3 *Let Q be a left-extended queue grammar. Then, there exists a scattered context grammar, $G = (N, T, P, S)$, such that $L(Q) = L(G)$, whereas P contains one non-context-free production of this form*

$$(1, 2, 0, 3, 0, 2, 1) \rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$$

Proof Let $Q = (V, T, W, F, s, R)$ be a left-extended queue grammar. Without any loss of generality, assume that Q satisfies the properties described in Lemma 2 and that $\{0, 1, 2, 3\} \cap (V \cup W) = \emptyset$. For some positive integer, $n \geq 1$, set

$$X = \{103\}^+\{1\}\{10\}^+ \cap \{x|x \in \{1, 0, 3\}^*, \mathrm{occur}(x, 1) = n\}$$

and introduce an injection ι from VW to X so that ι remains an injection when its domain is extended to $(VW)^*$ in the standard way; after this extension, ι thus represents an injection from $(VW)^*$ to X^* (a proof that such an injection necessarily exists is simple and left to the reader). Based on ι, define the substitution ν from V^* to X^* by $\nu(a) = \{\iota(aq) : q \in W\}$ for every $a \in V$. Notice that for every $x \in \nu(a)$, x represents a string of the form

$$103103 \ldots 10311010 \ldots 10$$

with n occurrences of 1, whereas substring 11 occurs in x precisely once. Define the homomorphism from $\{0, 1\}^*$ to $\{0, 1, 3\}^*$ by $\beta(0) = 30$ and $\beta(1) = 1$. Set

$$Y = \{301\}^*\{030\}\{10\}^* \cap \{x|x \in \{1, 0, 3\}^*, \mathrm{occur}(x, 1) = n\}$$

and define the substitution μ from W^* to Y^* by

$$\mu(q) = \{z10301w|u11v \in \iota(aq), a \in V, z = \beta(\mathrm{Reverse}(u)), w = \mathrm{Reverse}(\mathrm{Erase}(v, \{3\}))\}$$

Notice that for every $y \in \mu(q)$, y represents a string of the form

$$301301 \ldots 30103010101 \ldots 01$$

with n occurrences of 1, whereas 030 occurs in y precisely once.

Define the function Θ from X^* to Y^* recursively as follows

1. $\Theta(\varepsilon) = \varepsilon$
2. If $\Theta(x) = y, i \in \{2, \ldots, n-1\}, v \in X, v = 103103 \ldots 1031(10)^i, u \in Y, u = (301)^i 03010101 \ldots 01$, then$\Theta(ux) = yv$.

To illustrate, assume that $10310311010 \in X$; then, $\Theta(10310311010) = 30130103010101$.

$$\text{Set } U = \{\langle p, i\rangle : p \in W - F \text{ and } i \in \{1, 2\}\} \cup \{S\}.$$

Construction Introduce the scattered context grammar $G = (U \cup \{0, 1, 2, 3\}, T, P, S)$ with $P = M \cup O$ constructed in the following way.

To construct M, perform 1 through 5, given next.

1. if $a_0 q_0 = s$, where $a \in V - T$ and $q \in W - F$, then add $S \rightarrow 1u\langle q, 1\rangle w1$ to P, for all $u \in \nu(a0)$ and $w \in \mu(q_0)$;
2. if $(a, q, y, p) \in R$, where $a \in V - T, p, q \in W - F$, and $y \in (V - T)^*$, then add $\langle q, 1\rangle \rightarrow u\langle p, 1\rangle w$ to P, for all $u \in \nu(y)$ and $w \in \mu(p)$;
3. for every $q \in W - F$, add $\langle q, 1\rangle \rightarrow 2\langle q, 2\rangle$ to P;
4. if $(a, q, y, p) \in R$, where $a \in V - T, p, q \in W - F, y \in T^*$, then add $\langle q, 2\rangle \rightarrow y\langle p, 2\rangle w$ to P, for all $w \in \mu(p)$;
5. if $(a, q, y, p) \in R$, where $a \in V - T, q \in W - F, y \in T^*$, and $p \in F$, then add $\langle q, 2\rangle \rightarrow y302$;

Set

$$O = \{(1, 2, 0, 3, 0, 2, 1) \rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2), 2 \rightarrow \varepsilon\}$$

.

Basic Idea G can generate every $y \in L(G)$ as

$$S \Rightarrow^* 1u_0 02y302v_0 1 \Rightarrow 1u_1 y v_1 1 \Rightarrow 1u_2 y v_2 1 \Rightarrow \cdots \Rightarrow 1u_{m-1} y v_{m-1} 1 \Rightarrow 2y2 \Rightarrow^2 y$$

where G makes $S \Rightarrow^* 1u_0 02y302v_0 1$ by using productions from P, where $u_0 \in \nu(a_0 \ldots a_m)$ with $a_0, \ldots, a_m \in V - T$, and $v_0 \in \nu(q_m \ldots q_0)$ with $q_0, \ldots, q_m \in Q$. During $2y2 \Rightarrow^2 y$, G applies $2 \rightarrow \varepsilon$ twice to obtain $y \in L(G)$. During

$$1u_0 02y302v_0 1 \Rightarrow 1u_1 y v_1 1 \Rightarrow 1u_2 y v_2 1 \Rightarrow \cdots \Rightarrow 1u_{m-1} y v_{m-1} 1$$

G only applies $(1, 2, 0, 3, 0, 2, 1) \rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$. Consider any $1u_i y v_i 1$ in this derivation, $1 \leq i \leq m-1$, satisfy one of the following conditions i, ii, or iii:

(i) u_i ends with 120 while v_i starts with 3021;
(ii) u_i ends with 1203 while v_i starts with 021;
(iii) u_i ends with 12 while v_i starts with 03021.

Consider the symbols $1, 2, 0, 3, 0, 2, 1$ satisfying i, ii or iii. During $1u_i yv_i1 \Rightarrow 1u_{i+1}yv_{i+1}1$, G simultaneously rewrites these symbols by $(1, 2, 0, 3, 0, 2, 1) \rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$. Thus, from the definition of Θ and the way G performs $1u_0 02y302v_0 1 \Rightarrow^* 1u_{m-1}yv_{m-1}1$, we see that $\Theta(u_0) = v_0$. Examine the construction of P to see that $S \Rightarrow^* 1u_0 yv_0 1$ in G if and only if Q makes $\#a_0q_0 \Rightarrow^* a_0 \dots a_m\#yf$ according to (a_0, q_0, z_0, q_1) through (a_m, q_m, z_m, q_{m+1}), where $q_{m+1} \in F$. From this equivalence, we conclude that $L(G) = L(Q)$.

Formal Proof For brevity and readability, the following rigorous proof omits some obvious details, which the reader can easily fill in.

Claim A, proved next, establishes a derivation form by which G can generate each member of $L(G)$. This claim fulfills a crucial role in the demonstration that $L(G) \subseteq L(Q)$, given later in this proof (see Claim C)

Claim (A) G can generate every $h \in L(G)$ in this way

$$\begin{aligned}
&S\\
\Rightarrow &1x\langle q_0, 1\rangle t_0 1 \Rightarrow 1g_0\langle q_1, 1\rangle t_1 1 \Rightarrow \dots \Rightarrow 1g_{k-1}\langle q_k, 1\rangle t_k 1\\
\Rightarrow &1g_k\langle q_{k+1}, 1\rangle t_{k+1} 1 \Rightarrow 1g_k\langle q_{k+1}, 2\rangle t_{k+1} 1\\
\Rightarrow &1g_k 20y_1\langle q_{k+2}, 2\rangle t_{k+2} 1 \Rightarrow 1g_k 20y_1 y_2\langle q_{k+3}, 2\rangle t_{k+3} 1 \Rightarrow \dots\\
\Rightarrow &1g_k 20y_1 y_2 \dots y_{m-1}\langle q_{k+m}, 2\rangle t_{k+m} 1 \Rightarrow 1g_k 20y_1 y_2 \dots y_{m-1} y_m 032t_{k+m} 1\\
\Rightarrow &1u_1 y_1 y_2 \dots y_{m-1} y_m v_1 1 \Rightarrow 1u_2 y_1 y_2 \dots y_{m-1} y_m v_2 1 \Rightarrow \dots\\
\Rightarrow &1u_\nu y_1 y_2 \dots y_{m-1} y_m v_{\varpi} 1 \Rightarrow 2y_1 y_2 \dots y_{m-1} y_m 2 \Rightarrow^2 y_1 y_2 \dots y_{m-1} y_m
\end{aligned}$$

in G, where $k, m \geq 1; q_0, q_1, \dots, q_{k+m} \in W - F; y_1, \dots, y_m \in T^*; x \in \nu(a_0)$, where $a_0 \in (V - T)$ and $s = a_0 q_0; t_i \in \mu(q_i \dots q_1 q_0)$ for $i = 0, 1, \dots, k + m; g_j \in \nu(d_0 d_1 \dots d_j)$ with $d_0 = a_0$ and $d_1, \dots, d_j \in (V - T)^*$ for $j = 0, 1, \dots, k; d_0 d_1 \dots d_k = a_0 a_1 \dots a_{k+m}$ with $a_1, \dots, a_{k+m} \in V - T$ (that is, $g_k \in \nu(a_0 a_1 \dots a_{k+m})$) ; $1u_i y_1 y_2 \dots y_{m-1} y_m v_i 1$ satisfies either i $120 \in \text{suffix}(u_i)$ and $3021 \in \text{prefix}(v_i)$ or ii $1203 \in \text{suffix}(u_i)$ and $021 \in \text{prefix}(v_i)$ or iii $12 \in \text{suffix}(u_i)$ and $03021 \in \text{prefix}(v_i)$ so during $1u_i y_1 y_2 \dots y_{m-1} y_m v_i 1 \Rightarrow 1u_{i+1} y_1 y_2 \dots y_{m-1} y_m v_{i+1} 1, G$ simultaneously rewrites these prefixes and suffixes by $(1, 2, 0, 3, 0, 2, 1) \rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ for $0 \leq i \leq m - 1$, where $u_0 = g_k 20, v_0 = 032t_{k+m}; \Theta(g_k) = t_{k+m}$.

Proof (of Claim A) Examine the construction of P. Consider M. Observe that every derivation begins with an application of a production having S on its left-hand side. Set

$$\begin{aligned}
U_1 =&\{\langle p, 1\rangle : p \in W\}\\
U_2 =&\{\langle p, 2\rangle : p \in W\}\\
M_1 =&\{p : p \in M \text{ and } \text{lhs}(p) \in U_1\}\\
M_2 =&\{p : p \in M \text{ and } \text{lhs}(p) \in U_2\}
\end{aligned}$$

Consider any successful derivation that generate $h \in L(G)$. Let us make some observations. All application of productions from $1 - P$ precede the applications of productions from $2 - P = \{\langle p, 1\rangle : p \in W\}$. Furthermore, an application of $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ requires the occurrence of 3 in the sentential form, and this occurrence is produced only by a production constructed in the fifth step of the construction of M. After a production from this step is applied, no production from M can be applied throughout the rest of the derivation, so only productions from O can be used during this rest. An application of $2 \to \varepsilon$ eliminates 2, which cannot be produced by either of the two members of O. Thus, G applies $2 \to \varepsilon$ during the last two steps of the derivation. Taking these observations into account, we see that the generation of $h \in L(G)$ can be expressed as

$$
\begin{aligned}
& S \\
\Rightarrow & 1x\langle q_0, 1\rangle t_0 1 \Rightarrow 1g_0\langle q_1, 1\rangle t_1 1 \Rightarrow \cdots \Rightarrow 1g_{k-1}\langle q_k, 1\rangle t_k 1 \\
\Rightarrow & 1g_k\langle q_{k+1}, 1\rangle t_{k+1} 1 \Rightarrow 1g_k\langle q_{k+1}, 2\rangle t_{k+1} 1 \\
\Rightarrow & 1g_k 20 y_1\langle q_{k+2}, 2\rangle t_{k+2} 1 \Rightarrow 1g_k 20 y_1 y_2\langle q_{k+3}, 2\rangle t_{k+3} 1 \Rightarrow \ldots \\
\Rightarrow & 1g_k 20 y_1 y_2 \ldots y_{m-1}\langle q_{k+m}, 2\rangle t_{k+m} 1 \Rightarrow 1g_k 20 y_1 y_2 \ldots y_{m-1} y_m 032 t_{k+m} 1 \\
\Rightarrow^* & 2 y_1 y_2 \ldots y_{m-1} y_m 2 \Rightarrow^2 y_1 y_2 \ldots y_{m-1} y_m
\end{aligned}
$$

in G, $h = y_1 y_2 \ldots y_{m-1} y_m$, $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ is the only production applied during $1g_k 20 y_1 y_2 \ldots y_{m-1} y_m 032 t_{k+m} 1 \Rightarrow^* 2 y_1 y_2 \ldots y_{m-1} y_m 2$, and all the other involved symbols satisfy what is stated in Claim A (these symbols include x, gs, and ts).

Before going further, let us consider any three strings of the form $103103 \ldots 1031(10)^i \in X, 1203021, (301)^j 03010101 \ldots 01 \in Y$, where $i, j \in \{1, \ldots, n-1\}$ (see the definition of X for n) and study how to erase the concatenation

$$103103 \ldots 1031(10)^i 1203021(301)^j 03010101 \ldots 01$$

by repeatedly applying $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$. We intend to demonstrate that this erasure can be performed provided that $i = j$. First of all, notice that an occurrence of 1 between the two occurrences of 2 implies that this occurrence of 1 cannot be removed by $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$. Thus, $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ is always applied so the nearest possible pair of 1s that encloses 2s are rewritten by $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$. Specifically, the two underlined 1s are changed to 2s in

$$103103 \ldots 1031(10)^i \underline{1}203021\underline{1}(301)^j 03010101 \ldots 01$$

by using $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$. Next, we show that if $i \neq j$, then G cannot erase $103103 \ldots 1031(10)^i \underline{1}203021\underline{1}(301)^j 03010101 \ldots 01$.

Let $i < j$. As G always rewrites the nearest possible pair of 1s that encloses the two 2s, it obtains

$$103103 \ldots 103\underline{23}02(301)^{ji} 03010101 \ldots 01$$

after i derivation steps. As between 2 and 3 appears no 0, $(1, \underline{2}, \underline{0}, \underline{3}, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ is inapplicable (see the underlined symbols), which rules out the erasure.

Let $j < i$. After making i steps, G obtains

$$103103 \ldots 1031(10)^{i-j}120302103010101 \ldots 01,$$

from which G directly derives $103103 \ldots 1031(10)^{i-j-1}2003020101 \ldots 01$. After the next change of the closest pair of 1s to 2s, G obtains a string with no 3 occurring between the two 2s. As a result, $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ is inapplicable, and the erasure is ruled out. Consequently, $i = j$.

Return to

$$1g_k20y_1y_2 \ldots y_{m-1}y_m032t_{k+m}1 \Rightarrow^* 2y_1y_2 \ldots y_{m-1}y_m2$$

with $g_k \in \nu(a_0a_1 \ldots a_{k+m})$ and $t_{k+m} \in \mu(q_{k+m} \ldots q_1q_0)$. We have demonstrated that the erasure of $103103 \ldots 1031(10)^i1203021(301)^j03010101 \ldots 01$ implies $i = j$. Considering this implication together with the definitions of ν, μ and Θ, we see that $1g_k20032t_{k+m}1 \Rightarrow^* 22$ with $\Theta(g_k) = t_{k+m}$. Consequently, to express $1g_k20y_1y_2 \ldots y_{m-1}y_m032t_{k+m}1 \Rightarrow^* 2y_1y_2 \ldots y_{m-1}y_m2$ in a step-by-step way, we have

$$1g_k20y_1y_2 \ldots y_{m-1}y_m032t_{k+m}1 \Rightarrow 1u_1y_1y_2 \ldots y_{m-1}y_mv_11 \Rightarrow 1u_2y_1y_2 \ldots y_{m-1}y_mv_21 \Rightarrow \cdots \Rightarrow 1u_\nu y_1y_2 \ldots y_{m-1}y_mv_\varpi 1 \Rightarrow 2y_1y_2 \ldots y_{m-1}y_m2$$

in G, where $1u_iy_1y_2 \ldots y_{m-1}y_mv_i1$ satisfies either (i) $120 \in \text{suffix}(u_i)$ and $3021 \in \text{prefix}(v_i)$ or (ii) $1203 \in \text{suffix}(u_i)$ and $021 \in \text{prefix}(v_i)$ or (iii) $12 \in \text{suffix}(u_i)$ and $03021 \in \text{prefix}(v_i)$ so during $1u_iy_1y_2 \ldots y_{m-1}y_mv_i1 \Rightarrow 1u_{i+1}y_1y_2 \ldots y_{m-1}y_mv_{i+1}1$, G simultaneously rewrites these prefixes and suffixes by $(1, 2, 0, 3, 0, 2, 1) \to (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ for $0 \leq i \leq m-1$, where $u_0 = g_k20, v_0 = 032t_{k+m}$, and $\Theta(g_k) = t_{k+m}$. Of course,

$$2y_1y_2 \ldots y_{m-1}y_m2 \Rightarrow^2 y_1y_2 \ldots y_{m-1}y_m$$

is performed by applying $2 \to \varepsilon$ twice.

Putting all these partial derivations and their properties together, we obtain Claim A.

QED (Claim A)

Claim (B) Q generates every $h \in L(Q)$ in this way

$$
\begin{array}{ll}
\#a_0q_0 & \\
\Rightarrow a_0\#x_0q_1 & [(a_0, q_0, z_0, q_1)] \\
\Rightarrow a_0a_1\#x_1q_2 & [(a_1, q_1, z_1, q_2)] \\
\ldots & \\
\Rightarrow a_0a_1 \ldots a_k\#x_kq_{k+1} & [(a_k, q_k, z_k, q_{k+1})] \\
\Rightarrow a_0a_1 \ldots a_ka_{k+1}\#x_{k+1}y_1q_{k+2} & [(a_{k+1}, q_{k+1}, y_1, q_{k+2})] \\
\ldots & \\
\Rightarrow a_0a_1 \ldots a_ka_{k+1} \ldots a_{k+m-1}\#x_{k+m-1}y_1 \ldots & \\
\quad y_{m-1}q_{k+m} & [(a_{k+m-1}, q_{k+m-1}, y_{m-1}, q_{k+m})] \\
\Rightarrow a_0a_1 \ldots a_ka_{k+1} \ldots a_{k+m}\#y_1 \ldots y_mq_{k+m+1} & [(a_{k+m}, q_{k+m}, y_m, q_{k+m+1})]
\end{array}
$$

where $k, m \geq 1, a_i \in V-T$ for $i = 0, \ldots, k+m, x_j \in (V-T)^*$ for $j = 1, \ldots, k+m, s = a_0q_0; a_jx_j = x_{j-1}z_j$ for $j = 1, \ldots, k, a_1 \ldots a_kx_{k+1} = z_0 \ldots z_k, a_{k+1} \ldots a_{k+m} = x_k, q_0, q_1, \ldots, q_{k+m} \in W - F$ and $q_{k+m+1} \in F, z_1, \ldots, z_k \in (V - T)^*, y_1, \ldots, y_m \in T^*, h = y_1y_2 \ldots y_{m-1}y_m$.

Proof (of Claim B) Recall that Q satisfies the properties given in Lemma 2. These properties imply that Claim B holds.

QED (Claim B)

Claim (C) Let G generate $h \in L(G)$ in the way described in Claim A; then, $h \in L(Q)$.

Proof (of Claim C) Let $h \in L(G)$. Consider the generation of h as described in Claim A. Examine the construction of P to see that at this point R contains $(a_0, q_0, z_0, q_1), \ldots, (a_k, q_k, z_k, q_{k+1}), (a_{k+1}, q_{k+1}, y_1, q_{k+2}), \ldots, (a_{k+m-1}, q_{k+m-1}, y_{m-1}, q_{k+m})$, $(a_{k+m}, q_{k+m}, y_m, q_{k+m+1})$, where $z_1, \ldots, z_k \in (V - T)^*$, and $y_1, \ldots, y_m \in T^*$. Then, Q makes the generation of h in the way described in Claim B. Thus, $h \in L(Q)$.

QED (Claim C)

Claim (D) Let Q generates $h \in L(Q)$ in the way described in Claim B; then, $h \in L(G)$.

Proof This is left to the reader.

QED (Claim D)

Claims A through B imply that $L(Q) = L(G)$. Furthermore, $(1, 2, 0, 3, 0, 2, 1) \rightarrow (2, \varepsilon, \varepsilon, \varepsilon, \varepsilon, \varepsilon, 2)$ is the only non-context-free production in P. Therefore, this lemma holds. □

Theorem 1 *For every recursively enumerable language, L, there exists a scattered context grammar, $G = (N, T, P, S)$, such that $L = L(G)$ and P contains a single non-context-free production.*

Proof Recall that for every recursively enumerable language, L, there exists a queue grammar that generates L (see [2]). Thus, this theorem follows from Lemmas 1 through 3. □

As already pointed out, two-nonterminal scattered context grammars are computationally complete (see [1]). So are scattered context grammars with a single non-context-free production (see Theorem 1). Consider two-nonterminal scattered context grammars with one non-context-free production. Are they computationally complete, too?

Acknowledgements The authors thank Jakub Martiško for his useful comments. This work was supported by The Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability (NPU II), project IT4Innovations excellence in science – LQ1602; the TAČR grant TE01020415; and the BUT grant FIT-S-17-3964.

References

1. Csuhaj-Varjú, E. and Vaszil, Gy.: Scattered context grammars generate any recursively enumerable languages with two nonterminals, Information Processing Letters 110(20) (2010), 902-907.
2. Kleijn, H. C. M. and Rozenberg, G.: On the Generative Power of Regular Pattern Grammars, Acta Informatica 20, pp. 391-411, 1983.
3. Meduna, A. : Generative Power of Three-Nonterminal Scattered Context Grammars, Theoretical Computer Science 237 (2000), 625-631.
4. Meduna A.: Automata and Languages: Theory and Applications, Springer, London, 2000.
5. Meduna, A. and Techet, J.: Scattered Context Grammars and their Applications, WIT Press, Southhampton, UK, 2010.
6. Rozenberg, G. and Salomaa, A. (eds.): Handbook of Formal Languages, Volume 1, Springer, Berlin 1997.