More on the Size of Higman-Haines Sets: Effective Constructions

Hermann Gruber[1] and Markus Holzer[2] and Martin Kutrib[3]

[1] Institut für Informatik, Ludwig-Maximilians-Universität München
Oettingenstraße 67, D-80538 München, Germany
email: gruberh@tcs.ifi.lmu.de
[2] Institut für Informatik, Technische Universität München
Boltzmannstraße 3, D-85748 Garching bei München, Germany
email: holzer@in.tum.de
[3] Institut für Informatik, Universität Giessen
Arndtstraße 2, D-35392 Giessen, Germany
email: kutrib@informatik.uni-giessen.de

Abstract. A not well-known result [9, Theorem 4.4] in formal language theory is that the Higman-Haines sets for *any* language are regular, but it is easily seen that these sets *cannot* be effectively computed in general. Here the Higman-Haines sets are the languages of all scattered subwords of a given language and the sets of all words that contain some word of a given language as a scattered subword. Recently, the exact level of unsolvability of Higman-Haines sets was studied in [10]. We focus on language families whose Higman-Haines sets are effectively constructible. In particular, we study the size of Higman-Haines sets for the lower classes of the Chomsky hierarchy, namely for the families of regular, linear context-free, and context-free languages, and prove upper and lower bounds on the size of these sets.

1 Introduction

Higman's lemma [9] and its generalization, namely Kruskal's Tree Theorem [12], can be used to show that certain rewriting systems terminate. Nevertheless, the result of Higman is not so well known and was frequently rediscovered in the literature, e.g., [8, 13, 14]. Although Higman's result appears to be only of theoretical interest, it has some nice applications in formal language theory. It seems that one of the first applications has been given by Haines in [8, Theorem 3], where it is shown that the set of all scattered subwords, i.e., the *Higman-Haines set* $\textsc{Down}(L) = \{ v \in A^* \mid \text{there exists } w \in L \text{ such that } v \leq w \}$, and the set of all words that contain some word of a given language, i.e., the *Higman-Haines set* $\textsc{Up}(L) = \{ v \in A^* \mid \text{there exists } w \in L \text{ such that } w \leq v \}$, are both regular for *any* language $L \subseteq A^*$. Here, $\leq$ refers to the scattered subword relation. As pointed out in [8], this is an exceptional property which is quite unexpected. Further applications and generalizations of Higman's result can be found, e.g., in [4, 5, 11, 13].

It is worth mentioning that $\textsc{Down}(L)$ and $\textsc{Up}(L)$ cannot be obtained constructively in general. This is clear, because L is empty if and only if $\textsc{Down}(L)$ and $\textsc{Up}(L)$ are empty, but the question whether or not a language is empty is undecidable for recursively enumerable languages and decidable for regular ones. Thus, as expected, for the family of recursively enumerable languages the Higman-Haines sets are not constructible, while it is not hard to see that for regular languages the construction becomes effective. But where exactly is the borderline between language families with non-constructive and constructive Higman-Haines sets? One might expect that, e.g., the family of context-free languages has non-constructive Higman-Haines sets, but surprisingly this is not the case, as proven in [14]. On the other hand, recently it was shown in [10] that, for instance, the family of Church-Rosser languages has non-constructive Higman-Haines sets. This language family lies in between the regular languages and the growing context-sensitive languages, but is incomparable to the family of context-free languages [1]. Moreover, in [10] the exact level of unsolvability of the Higman-Haines sets for certain language families is studied. Thus, the non-constructive side of Higman-Haines sets is well studied, but is there more to be known about effective constructibility issues as presented in [14]? Moreover, are there any results about descriptional complexity issues? To our knowledge this is not the case, except for some results about regular languages accepted by nondeterministic finite automata in [10]. This is the starting point of our investigations about effective Higman-Haines set sizes. In particular we consider the problem of computing the Higman-Haines sets induced by the families of regular, context-free, and linear context-free languages. For the size of the Higman-Haines sets generated by regular languages upper and lower bounds are presented. That is, we prove that an exponential blow-up is sufficient and necessary in the worst case for a deterministic finite automaton to accept the Higman-Haines set $\textsc{Down}(L)$ or $\textsc{Up}(L)$ generated by some language that is represented by another deterministic finite automaton. This nicely contrasts the result about nondeterministic finite automata where a matching upper and lower bound on the size of Higman-Haines sets is shown [10]. Furthermore, we investigate the descriptional complexity of the Higman-Haines sets when the underlying device is a context-free or linear context-free grammar.

The paper is organized as follows. The next section contains preliminaries and basics about Higman-Haines sets. Then Section 3 first recalls the known upper and lower bounds for nondeterministic finite automata [10], and then studies the size of the Higman-Haines set for regular languages in terms of deterministic finite automata size. In addition, Higman-Haines sets induced by context-free and linear context-free languages are investigated.

2 Preliminaries

We denote the set of non-negative integers by $\mathbb{N}$. The powerset of a set S is denoted by 2^S. For an alphabet A, let A^+ be the set of non-empty words w over A. If the empty word λ is included, then we use the notation A^*. For the

length of w we write $|w|$. For the number of occurrences of a symbol a in w we use the notation $|w|_a$. Set inclusion is denoted by $\subseteq$, and strict set inclusion by $\subset$. Let $v, w \in A^*$ be words over alphabet A. We define $v \leq w$ if and only if there are words $v_1, v_2, \ldots, v_k$ and $w_1, w_2, \ldots, w_{k+1}$, for some $k \geq 1$, $v_i \in A^*$, $w_i \in A^*$, such that $v = v_1 v_2 \ldots v_k$ and $w = w_1 v_1 w_2, v_2 \ldots w_k v_k w_{k+1}$. In case of $v \leq w$ we say that v is a scattered subword of w. Let L be a language over alphabet A. Then

$$\textsc{Down}(L) = \{\, v \in A^* \mid \text{there exists } w \in L \text{ such that } v \leq w \,\}$$

and

$$\textsc{Up}(L) = \{\, v \in A^* \mid \text{there exists } w \in L \text{ such that } w \leq v \,\}$$

are the *Higman-Haines sets* generated by L. The next theorem is the surprising result of Haines. It has been shown about half a century ago. Actually, it is a corollary of Higman's work, but let us state it as a theorem.

Theorem 1 ([8, 9]). *Let L be an arbitrary language, then both* $\textsc{Down}(L)$ *and* $\textsc{Up}(L)$ *are regular.*

In order to talk about the economy of descriptions we first have to define what is meant by the *size of automata and grammars*. In general, we are interested to measure the *length of the string that defines an automaton or grammar*. In particular, we sometimes use more convenient size measures, if there is a recursive upper bound for the length of the defining string dependent on the chosen size measure. For example, for *context-sensitive and context-free grammars* M, the size $|M|$ equals the total number of occurrences of terminal and nonterminal symbols in the productions. For *deterministic and nondeterministic finite automata* M, the size $|M|$ equals the product of the number of states and the number of input symbols.

3 Effective Higman-Haines Set Sizes

Next we turn to the family of regular languages and then to the family of context-free languages, whose Higman-Haines sets can effectively be constructed [14]. We are interested in the constructions itself as well as in the sizes of the Higman-Haines sets.

3.1 Regular Languages

Let $M = (S, A, \delta, s_0, F)$ be a nondeterministic finite automaton (NFA), where S is the finite set of *internal states*, A is the finite set of *input symbols*, $s_0 \in S$ is the *initial state*, $F \subseteq S$ is the set of *accepting states*, and $\delta : S \times (A \cup \{\lambda\}) \to 2^S$ is the *partial transition function*. An NFA is deterministic (DFA) if and only if $|\delta(s, a)| \leq 1$, $|\delta(s, \lambda)| \leq 1$, and $|\delta(s, a)| = 1 \iff |\delta(s, \lambda)| = 0$, for all $s \in S$ and $a \in A$. Without loss of generality, we assume that the NFAs are always *reduced*.

This means that there are no unreachable states and that from any state an accepting state can be reached.

Concerning the size of an NFA accepting $\textsc{Down}(L(M))$ or $\textsc{Up}(L(M))$ for a given NFA M, one finds the following situation, which was proven in [10].

Lemma 2. *Let M be an NFA of size n. Then size n is sufficient and necessary in the worst case for an NFA M' to accept $\textsc{Down}(L(M))$ or $\textsc{Up}(L(M))$. The NFA M' can effectively be constructed.*

In the remainder of this subsection we consider DFAs. First observe, that the results presented so far heavily rely on nondeterminism, i.e., even when starting with a DFA M, the resulting automata accepting $\textsc{Down}(L(M))$ or $\textsc{Up}(L(M))$ are nondeterministic in general. So, applying the well-known power-set construction gives an upper bound on the size of an equivalent DFA.

Corollary 3. *For any DFA M of size n, one can effectively construct a DFA accepting $\textsc{Down}(L(M))$ or $\textsc{Up}(L(M))$ whose size is at most 2^n.* □

For the next two theorems we need some more notations. Let $L \subseteq A^*$ be an arbitrary language. Then the *Myhill-Nerode* equivalence relation $\equiv_L$ is defined as follows: For $u, v \in A^*$, let $u \equiv_L v$ if and only if $uw \in L \iff vw \in L$, for all $w \in A^*$. It is well known that the number of states of the minimal deterministic finite automaton accepting the language $L \subseteq A^*$ equals the index, i.e., the cardinality of the set of equivalence classes, of the Myhill-Nerode equivalence relation.

We continue our investigations by proving a non-trivial lower bound for DFAs accepting the language $\textsc{Down}(L(M))$, for some given DFA M, that is quite close to the upper bound of the previous corollary.

Theorem 4. *For every $n \geq 1$, there exists a language L_n over an $(n+2)$-letter alphabet accepted by a DFA of size $(n+2)(n+1)^2$, such that size $2^{\Omega(n \log n)}$ is necessary for any DFA accepting $\textsc{Down}(L_n)$.*

Proof. Let $A = \{a_1, a_2, \ldots, a_n\}$ and $\#, \$ \notin A$. Consider the witness language $L_n = \{\, \#^j \$ w \mid w \in A^*, j \geq 0, i = j \bmod n, |w|_{a_{i+1}} = n \,\} \subseteq (A \cup \{\#, \$\})^*$. A DFA accepting language L_3 is depicted in Figure 1. It is not hard to see that any DFA accepting L_n needs $n+1$ states for each letter a_i to count up to n. Moreover, for the $\#$-prefix n states are used, and finally one non-accepting sink state is needed. This results in $n(n+1)+n+1$ states, which gives size $(n+2)(n^2+2n+1) = (n+2)(n+1)^2$. It is not hard to verify that the DFA is minimal. Recall the construction of an NFA for the down-set. Then one observes that $\textsc{Down}(L_n) = \{\, \#^j a w \mid w \in A^*, j \geq 0, a \in \{\$, \lambda\} \text{ and } \bigvee_{i=1}^{n} |w|_{a_i} \leq n \,\}$.

It remains to be shown that the minimal DFA accepting $\textsc{Down}(L_n)$ has at least $(n+1)^n + 2$ states. First observe that any two different words of the form $w_{i_1,i_2,\ldots,i_n} = \$ a_1^{i_1} a_2^{i_2} \ldots a_n^{i_n}$ with $0 \leq i_j \leq n$ and $1 \leq j \leq n$ are non-equivalent with respect to the Myhill-Nerode relation $\equiv_{\textsc{Down}(L_n)}$. Let $w_{i_1,i_2,\ldots,i_n}$ and $w_{i'_1,i'_2,\ldots,i'_n}$ be two different words. Then $i_k \neq i'_k$, for some $1 \leq k \leq n$. Without loss of generality we assume $i_k < i'_k$. Then the word

$$w_{i_1,i_2,\ldots,i_n} \cdot a_1^{n+1} a_2^{n+1} \ldots a_{k-1}^{n+1} a_k^{(n+1)-i'_k} a_{k+1}^{n+1} \ldots a_n^{n+1}$$

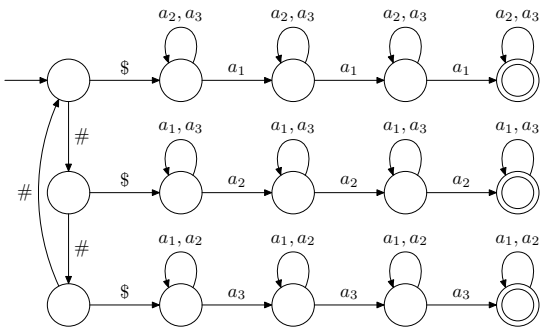

Fig. 1. A DFA of size $5 \cdot 16$ accepting $\textsc{Down}(L_3)$—the non-accepting sink state is not shown.

belongs to $\textsc{Down}(L_n)$ because letter a_k appears at most n times. On the other hand, the word $w_{i'_1,i'_2,\ldots,i'_n} \cdot a_1^{n+1} a_2^{n+1} \ldots a_{k-1}^{n+1} a_k^{(n+1)-i'_k} a_{k+1}^{n+1} \ldots a_n^{n+1}$ is not member of $\textsc{Down}(L_n)$ since all letters a_i, for $1 \leq i \leq n$, appear at least $n+1$ times. Hence, there are $(n+1)^n$ different equivalence classes induced by the words $w_{i_1,i_2,\ldots,i_n}$. Moreover, none of the words λ, $w_{i_1,i_2,\ldots,i_n}$ with $0 \leq i_j \leq n$ and $1 \leq j \leq n$, and $\$a_1^{n+1} a_2^{n+1} \ldots a_n^{n+1}$ belong to the same equivalence classes. For λ and $w_{i_1,i_2,\ldots,i_n}$ this is seen by concatenating the words with \$, and the remaining pairs are shown to be non-equivalent by concatenating them with the empty word λ. Therefore, we have obtained at least $(n+1)^n + 2$ equivalence classes. In fact, one can construct a DFA with exactly this number of states accepting $\textsc{Down}(L_n)$. The details are left to the reader. Therefore, $2^{\Omega(n \log n)}$ is a lower bound on the size of any DFA accepting $\textsc{Down}(L_n)$. □

The next theorem gives a lower bound for the size of any DFA accepting $\textsc{Up}(L(M))$, for a given DFA M. The proof is similar to the proof of the previous theorem.

Theorem 5. *For every $n \geq 1$, there exists a language L_n over an $(n+2)$-letter alphabet accepted by a DFA of size $(n+2)(n+1)^2$, such that size $2^{\Omega(n \log n)}$ is necessary for any DFA accepting $\textsc{Up}(L_n)$.* □

Finally, it is worth to mention that the lower bounds of the previous two theorems slightly improve when the number of states is used to measure the size of DFAs. The next theorem summarizes the lower bounds.

Theorem 6. *For every $n \geq 1$, there exists a language L_n over an $(n+2)$-letter alphabet accepted by a DFA with $(n+1)^2$ states, such that $2^{\Omega(n \log n)}$ states are necessary for any DFA accepting $\textsc{Down}(L_n)$. A similar statement is valid for $\textsc{Up}(L_n)$.* □

3.2 Context-Free and Linear Context-Free Languages

In this subsection we are interested in the size of NFAs accepting the Higman-Haines sets for context-free or linear context-free grammars. Recall that we use

the total number of occurrences of terminal and nonterminal symbols in the productions as size measure for grammars. Let $G = (N, T, P, S)$ be a context-free grammar, where N is the finite set of *nonterminals*, T is the finite set of *terminals*, $P \subseteq N \times (N \cup T)^*$ is the finite set of productions, and $S \in N$ is the *axiom*. A context-free grammar $G = (N, T, P, S)$ is *linear context free* if $P \subseteq N \times T^*(N \cup \{\lambda\})T^*$. Without loss of generality, we assume that the context-free grammars are always *reduced*, i.e., that there are no unreachable or unproductive nonterminals. Moreover, in this section we further assume that the context-free grammars are in Chomsky normalform, i.e., the productions are of the form $P \subseteq N \times (N^2 \cup T)$. For linear context-free grammars the normalform reads as $P \subseteq N \times (NT \cup TN \cup T)$.

As in the previous subsection we first show how to construct an NFA for $\textsc{Down}(L(G))$. In order to simplify the analysis we assume that the right-hand sides of the productions are described by NFAs with input alphabet $N \cup T$. We refer to such a grammar as an *extended* (linear) context-free grammar. Note, that one can assume that for each extended context-free grammar there is exactly one NFA for each nonterminal as a right-hand side. The following theorem is a detailed analysis of the inductive construction presented in [14].

Theorem 7. *Let G be a context-free grammar of size n. Then size $O(n2^{\sqrt{2n}\log n})$ is sufficient for an NFA M' to accept* $\textsc{Down}(L(G))$. *The NFA M' can effectively be constructed.*

Proof. First, the context-free grammar $G = (N, T, P, S)$ is transformed into an extended context-free grammar G'—the details are omitted here. Secondly, we observe that each nonterminal appears at the left-hand side of at least one production, respectively, and at least one nonterminal is rewritten by some terminal symbol. Therefore, the number of nonterminals is at most $\lfloor \frac{n}{2} \rfloor$.

Next, we inductively proceed as in [14]. For a nonterminal $A \in N$ we set the alphabet $V_A = (N \setminus \{A\}) \cup T$, and define the extended context-free grammar $G_A = (\{A\}, V_A, P_A, A)$ with $P_A = \{ A \to M \mid (A \to M) \in P\}$, where M in $(A \to M) \in P$ refers to the NFA of the right-hand side of the production. Further, we set $L_A = L(G_A)$. Observe, that G_A is an extended context-free grammar with only *one* nonterminal and, thus, one can obtain an NFA M_A describing $\textsc{Down}(L(G_A))$ over the alphabet V_A by a subroutine to be detailed below. Then the induction is as follows: Let $G_0 = G'$. If A is not the axiom S of G_0, we can replace each A-transition occurring in the right-hand side automata of non-A-productions of G_0 with a copy of M_A to obtain an extended grammar G_1 having one nonterminal less than G_0, and $\textsc{Down}(L(G_1)) = \textsc{Down}(L(G_0))$. This construction step can be iterated for at most $\lfloor \frac{n}{2} \rfloor - 1$ times, yielding extended context-free grammars $G_2, G_3, \ldots, G_{\lfloor \frac{n}{2} \rfloor - 1}$, satisfying $\textsc{Down}(L(G_i)) = \textsc{Down}(L(G_{i+1}))$, for $0 \leq i < \lfloor \frac{n}{2} \rfloor$, where in the latter grammar $G_{\lfloor \frac{n}{2} \rfloor - 1}$ the only remaining nonterminal is the original axiom S of G. Finally, we apply the mentioned subroutine to construct the NFA M' which results in the finite automaton accepting the language $\textsc{Down}(L(G))$.

It remains to describe the above mentioned subroutine and deduce an upper bound on the size of the automaton M'. The subroutine works for an extended

grammar $G_A = (\{A\}, V_A, \{A \to M\}, A)$ with only *one* nonterminal. Then we distinguish two cases:

1. The production set given by $L(M)$ is linear, i.e., $L(M) \subseteq V_A^*\{A, \lambda\}V_A^*$, or
2. the production set given by $L(M)$ is nonlinear.

In the first case, we construct an NFA M_T with $L(M_T) = L(M) \cap V_A^*$, which is obtained by removing all A-transitions from M. Similarly, we build NFAs M_P and M_S for the quotients

$$L(M_P) = \{\, x \in V_A^* \mid xAz \in L(M) \text{ for some } z \in (V_A \cup \{A\})^* \,\} \text{ and}$$
$$L(M_S) = \{\, z \in V_A^* \mid xAz \in L(M) \text{ for some } x \in (V_A \cup \{A\})^* \,\}.$$

Then it is straightforward to construct an NFA M_A having a single start state and a single accepting state with

$$L(M_A) = \textsc{Down}(L(M_P)^* \cdot L(M_T) \cdot L(M_S)^*) = \textsc{Down}(L(G_A)).$$

The number of alphabetic transitions, i.e., non-λ-transitions, in M_A is at most three times that of M. In the second case, i.e., $L(M)$ is nonlinear, we construct automata M_P, M_T, M_S, and M_I, where the former three NFAs are as in the previous case, and M_I accepts the quotient

$$L(M_I) = \{\, y \in V_A^* \mid xAyAz \in L(M) \text{ for some } x, z \in (V_A \cup \{A\})^* \,\}.$$

Again, it is not hard to construct an NFA M_A with a single start and a single accepting state accepting

$$L(M_A) = \textsc{Down}((L(M_T) \cup L(M_P) \cup L(M_I) \cup L(M_S))^*) = \textsc{Down}(L(G_A))$$

with no more than four times as many alphabetic transitions as M.

The upper bound on the size of an NFA accepting $\textsc{Down}(L(G))$ is deduced as follows: For an extended context-free grammar G, let $|G|_t$ denote the sum of the number of alphabet transitions in the right-hand side automata in the productions of G. We obtain the recurrence $|G_k|_t \leq 4 \cdot (|G_{k-1}|_t)^2$, for $1 \leq k < \lfloor \frac{n}{2} \rfloor$, describing the substitution step in the kth iteration to construct G_k from G_{k-1}. Taking logarithms and setting $H_k = \log |G_k|_t$, we obtain a linear recurrence $H_k \leq 2 \cdot H_{k-1} + 2$. Solving the linear recurrence, we obtain the inequality $H_k \leq 2^k \cdot H_0 + 2^{k+1} - 2$. Since $|G_0|_t \leq n$, we have

$$H_{\lfloor \frac{n}{2} \rfloor - 1} \leq 2^{\lfloor \frac{n}{2} \rfloor - 1} \cdot H_0 + 2^{\lfloor \frac{n}{2} \rfloor} - 2 \leq 2^{\lfloor \frac{n}{2} \rfloor - 1} \cdot \log n + 2^{\lfloor \frac{n}{2} \rfloor} - 2.$$

When replacing the axiom in $G_{\lfloor \frac{n}{2} \rfloor - 1}$ in the final step, the number of alphabetic transitions is increased at most by a factor of four, which results in

$$|G_{\lfloor \frac{n}{2} \rfloor}|_t \leq 2^{2^{\lfloor \frac{n}{2} \rfloor - 1} \cdot \log n + 2^{\lfloor \frac{n}{2} \rfloor}} \leq 2^{2^{\lfloor \frac{n}{2} \rfloor - 1} \cdot \log n + 2^{\lfloor \frac{n}{2} \rfloor - 1} \cdot \log n} \leq 2^{\sqrt{2^n} \log n},$$

for all $n \geq 4$. It remains to be shown that for every NFA with n alphabetical transitions, there is an equivalent NFA with at most $O(n)$ states. An easy

construction can be used to remove all non-initial states having neither ingoing nor outgoing alphabetical transitions after adding some extra λ-transitions where necessary. By a simple counting argument, we find that the latter automaton has at most $2n+1$ states. Hence, this shows that the NFA M' accepting $\textsc{Down}(L(G))$ has size at most $O(n \cdot 2^{\sqrt{2n \log n}})$. □

For the lower bound we obtain:

Theorem 8. *For every $n \geq 1$, there is a language L_n over a unary alphabet generated by a context-free grammar of size $3n+2$, such that size $2^{\Omega(n)}$ is necessary for any NFA accepting $\textsc{Down}(L(G))$ or $\textsc{Up}(L(G))$.*

Proof. For every $n \geq 1$, consider the finite unary languages $L_n = \{a^{2^n}\}$ generated by the context-free grammar $G = (\{A_1, A_2, \ldots, A_{n+1}\}, \{a\}, P, A_1)$ with the productions $A_i \to A_{i+1}A_{i+1}$, for $1 \leq i \leq n$, and $A_{n+1} \to a$. Obviously, grammar G has size $3n+2$. The word a^{2^n} is the longest word in $\textsc{Down}(L(G))$ and the shortest word in $\textsc{Up}(L(G))$. In both cases, any finite automaton accepting the language takes at least as many states as the length of the word. So, it takes at least least 2^n states and, thus, has at least size 2^n. □

We turn our attention to the construction of an NFA accepting $\textsc{Up}(L(G))$, for a context-free grammar G. To this end, we define the *basis of a language* as follows: A word $w \in L$ is called *minimal* in L if and only if there is no different $v \in L$ with $v \leq w$. The set of minimal elements in L is called a *basis* of the language $\textsc{Up}(L)$. Observe that any shortest word in L is minimal in L, and any such word must therefore be part of the basis. In fact, Higman's Lemma [9] says that for any arbitrary language L there exists a natural number n, which depends only on L, such that $\textsc{Up}(L) = \bigcup_{1 \leq i \leq n} \textsc{Up}(\{w_i\})$, for some words $w_i \in L$ with $1 \leq i \leq n$. Sometimes the result is called the *finite basis property*. For the construction of an NFA accepting $\textsc{Up}(L(G))$, where G is a context-free grammar with terminal alphabet A, we proceed as follows:

1. Determine the basis $B \subseteq A^*$ of the language $\textsc{Up}(L(G))$ with the algorithm presented in [14].
2. Construct an NFA M accepting language B, and apply the construction given in the previous subsection to obtain an NFA M' accepting $\textsc{Up}(B)$, which equals the language $\textsc{Up}(L(G))$ by the finite basis property.

The first step basically consists in inductively computing B starting from $B_0 = \emptyset$, and B_{i+1} is obtained by extending B_i by a shortest word w in $L(G) \setminus \textsc{Up}(B_i)$, i.e., setting $B_{i+1} = B_i \cup \{w\}$. This process is repeated as long as $(L(G) \setminus \textsc{Up}(B_i)) \neq \emptyset$. If this condition is met, the set B equals the last extended B_i. Since context-free languages are closed under set difference with regular sets, the set B can be effectively constructed.

Theorem 9. *Let G be a context-free grammar of size n. Then an NFA M' of size $O(\sqrt{n 2^{2^n \log n}})$ is sufficient to accept $\textsc{Up}(L(G))$. The NFA M' can effectively be constructed.*

In the remainder of this section we concentrate on linear context-free languages.

Theorem 10. *Let G be a linear grammar of size n. Then an NFA M' of size $O\left(\sqrt{2^{n^2+\frac{(3n+6)}{2}\log n-(4+\log e)n}}\right)$ is sufficient to accept* $\textsc{Down}(L(G))$. *The NFA M' can effectively be constructed.*

Proof. Let $G = (N, T, P, A_1)$ with $N = \{ A_i \mid 1 \leq i \leq m \}$ be a linear context-free grammar. The basic idea for the construction of M' is to inspect the derivation trees of G and to modify the underlying grammar such that any self-embedding derivation of the form $A \Rightarrow^* xAz$, for some $A \in N$ and $x, z \in T^*$, is replaced by a derivation $A \Rightarrow^* xA$ and $A \Rightarrow^* Az$, while the respective generated languages have the same $\textsc{Down}$-sets. In other words, the derivation that produces the "coupled" terminal words x and z is made "uncoupled" by a right-linear and a left-linear derivation. In order to make the construction work, one has to take care about these self-embedded derivation parts in an appropriate manner. For a formal treatment of the construction we need some notation.

Let $A_1 \Rightarrow^* w$ be a derivation of $w \in T^*$. Then the inner nodes of the derivation tree form a path $p = A_1 A_{i_1} A_{i_2} \cdots A_{i_k}$. We can group the inner nodes as follows: We call a subpath of p that represents a self-embedded derivation with nonterminal A, i.e., which begins and ends with the same nonterminal A, an *A-block*. A *splitting* of p into blocks is an ordered set $\mathcal{B}$ of blocks such that

1. any block in p is a subpath of exactly one element in $\mathcal{B}$,
2. there is at most one A-block for each nonterminal $A \in N$.

A splitting always exists, as the first condition can be ensured by adding blocks to $\mathcal{B}$ as long as necessary. Afterwards we can enforce the remaining conditions by merging blocks. The order of the set $\mathcal{B}$ is given naturally by the occurrence of blocks along the path. For such a splitting, we call a subpath connecting two consecutive blocks an *(A, B)-nonblock*, if the first is an A-block and the second one a B-block. By convention, the borders A and B are part of the nonblock. If the first or the last nonterminal of the path are not part of blocks, we agree that the paths connecting the ends to the first and last block are also nonblocks. A simple example explaining our terminology is depicted in Figure 2, where it is shown that a splitting is not necessarily unique.

Next, for each nonterminal $A \in N$ we build NFAs $M_{A,P}$ and $M_{A,S}$ for the quotients

$$L(M_{A,P}) = \{ x \in T^* \mid A \Rightarrow^* xAz \text{ for some } z \in T^* \} \text{ and}$$
$$L(M_{A,S}) = \{ z \in T^* \mid A \Rightarrow^* xAz \text{ for some } x \in T^* \}.$$

Then it is straightforward to construct an NFA M_A having a single start state and a single accepting state such that $L(M_A)$ is the $\textsc{Down}$-set of the set of all partial derivations corresponding to an A-block, i.e.,

$$L(M_A) = \textsc{Down}(L(M_{A,P})^* \cdot A \cdot L(M_{A,S})^*).$$

The number of states in M_A is at most $2|N| \leq n$, and it contains a single A-transition. Moreover, for every $A, B \in N$ we build NFAs $M_{A,I}$ and $M_{(A,B),I}$ taking care of the terminating derivation part and the nonblocks, namely

$$L(M_{A,I}) = \{y \in T^* \mid A \Rightarrow^* y \text{ is an acyclic derivation, } y \in T^*\} \text{ and}$$
$$L(M_{(A,B),I}) = \{xBz \in T^*NT^* \mid A \Rightarrow^* xBz \text{ is an acyclic derivation, } x, z \in T^*\}.$$

Here a derivation is said to be acyclic, if no nonterminal occurs more than once in the derivation. The Down-set of all partial derivations corresponding to some (A, B)-nonblock is given by $L(M_{(A,B)}) = \text{Down}(L(M_{(A,B),I}))$, the Down-set of the terminating derivation part by $L(M_{(A)}) = \text{Down}(L(M_{(A,I)}))$. We note two features of $L(M_{(A,B),I})$: First, all words in the language are at most of length $|N|$, and secondly, by [2, Lemma 4.3.2], it contains at most $2^{|P|-1}$ words. Then the construction given in [7] yields an NFA $M_{(A,B),I}$ with at most $\frac{3}{\sqrt{2}} \cdot n^{\frac{n}{4}}$ states and at most 2^{n-1} many B-transitions accepting this language, as $|\Sigma| \leq n$ and $|N|$ as well as $|P|$ cannot exceed $n/2$. The same bound on the number of states applies to $M_{A,I}$, and due to Lemma 2 and the constructions of NFAs for Down-sets of NFA languages, the bounds on states and transitions apply also to $M_{(A,B)}$ and $M_{(A)}$.

Finally, for every splitting $\mathcal{B} = \{A_{i_1}, A_{i_2}, \ldots, A_{i_m}\}$ containing m blocks, we obtain an NFA accepting the Down-set of all derivations $A_1 \Rightarrow^* w$ whose trees admit a $\mathcal{B}$-splitting by iterated substitution of transitions by NFAs. We start with the terminating derivation part, i.e, the NFA $M_{(A_{i_m})}$ with no more than $H_0 = \frac{3}{\sqrt{2}} \cdot n^{\frac{n}{4}}$ states. Next we proceed in cycles. In each cycle k, two substitution phases are performed. First, the current NFA, say with H_k states, replaces the sole (A_{i_m-k})-transition of the NFA $M_{A_{i_m-k}}$. This results in at most H_k+n states. Secondly, all (A_{i_m-k})-transitions of the NFA $M_{(A_{i_m-k-1},A_{i_m-k})}$ are replaced by the NFA constructed in the first phase. The result is an NFA with at most $2^{n-1}(H_k + n) + H_0$ states. Clearly, the construction is completed after m cycles.

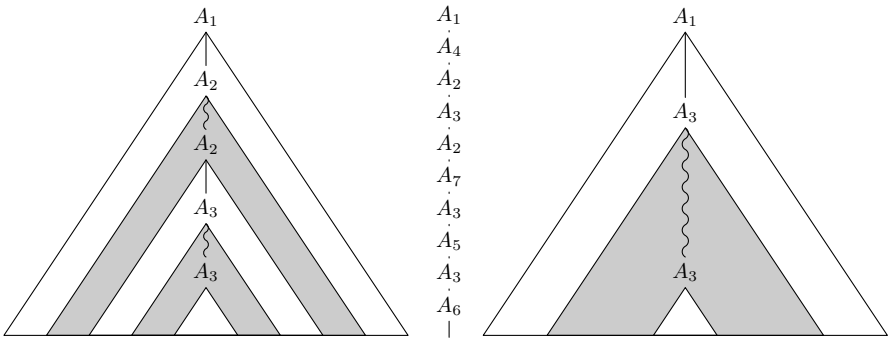

Fig. 2. Two splittings for the path $p = A_1, A_4, A_2, A_3, A_2, A_7, A_3, A_5, A_3, A_6$; blocks are gray shaded and the derivation is drawn by a curled path, while nonblocks are white and their derivation is drawn by a straight line.

For the number of states, we have to solve recurrence $H_m = 2^{n-1}(H_{m-1}+n)+H_0$ with $H_0 = \frac{3}{\sqrt{2}} \cdot n^{\frac{n}{4}}$. Unrolling yields the series

$$H_m = H_0 + (H_0+n)\sum_{i=1}^{m}(2^{n-1})^i = H_0 + (H_0+n)\frac{2^{(n-1)(k+1)}-1}{2^{n-1}-1} - 1.$$

Since $m+1 \leq |N| \leq n/2$, this is less than or equal to

$$H_0 + (H_0+n)\frac{2^{\frac{(n-1)n}{2}}}{2^{n-2}} \leq H_0 + (H_0+n)4\frac{2^{\frac{n^2}{2}}}{2^{\frac{3n}{2}}} \in O\left(\frac{n^{\frac{n}{4}}2^{\frac{n^2}{2}}}{2^{\frac{3n}{2}}}\right) = O\left(\sqrt{2^{n^2+\frac{n}{2}\log n - 3n}}\right).$$

An important observation is that this automaton also accepts the Down-set of all derivations whose trees admit some splitting in $\text{Down}(\mathcal{B})$. So, it suffices to consider $|N|!$ relevant different splittings. Therefore, the number of states of the NFA M' accepting $\text{Down}(L(G))$ is at most $O\left(\left(\frac{n}{2}\right)!\sqrt{2^{n^2+\frac{n}{2}\log n-3n}}\right)$. This implies a size of $O\left(\left(\frac{n}{2}\right)!\sqrt{2^{n^2+\frac{(n+4)}{2}\log n-3n}}\right)$. Finally, Stirling's approximation yields $O\left(\left(\frac{n}{2}\right)!\right) = O\left(\sqrt{n}\left(\frac{n}{2e}\right)^{\frac{n}{2}}\right) = O\left(\sqrt{2^{\log n}}\sqrt{2^{n\log n}}\sqrt{2^{-n(1+\log e)}}\right)$ and, thus, an upper bound of

$$O\left(\sqrt{n}\left(\frac{n}{2e}\right)^{\frac{n}{2}}\sqrt{2^{n^2+\frac{(n+4)}{2}\log n-3n}}\right) = O\left(\sqrt{2^{n^2+\frac{(3n+6)}{2}\log n-(4+\log e)n}}\right).$$

□

In order to show the lower bound we use the finite witness language $L_n = \{\, ww^R \mid w \in \{a,b\}^n \,\}$, which can be generated by a linear context-free grammar $G = (\{\, A_i, A_i', A_i'' \mid 1 \leq i \leq n \,\}, \{a,b\}, P, A_1)$ with the productions $A_i \to aA_i'$, $A_i' \to A_{i+1}a$, $A_i \to bA_i''$, $A_i'' \to A_{i+1}b$, for $1 \leq i < n$, and $A_n \to aA_n'$, $A_n' \to a$, $A_n \to bA_n''$, $A_n'' \to b$. Since any NFA accepting L_n needs at least 2^n states—see, e.g., [6]—the next theorem reads as follows. Observe, that the lower bound also holds for the up-set problem.

Theorem 11. *For every $n \geq 1$, there is a linear context-free language L_n over a binary alphabet generated by a linear context-free grammar of size $12n-2$, such that size $2^{\Omega(n)}$ is necessary for any NFA accepting $\text{Down}(L(G))$ or $\text{Up}(L(G))$.* □

For the size of $\text{Up}(L(G))$, for some linear context-free grammar G of size n, we argue as follows: The basis B of $\text{Up}(L(G))$ contains only words whose lengths are at most n. Then by similar arguments as in the proof of Theorem 9 we obtain the following result, which is much better than that for general context-free grammars.

Theorem 12. *Let G be a linear context-free grammar of size n. Then an NFA M' of size $O(\sqrt{2^{(n+2)\log n}})$ is sufficient to accept $\text{Up}(L(G))$. The NFA M' can effectively be constructed.* □

4 Conclusions

Several questions about the size of Higman-Haines sets remain unanswered. We mention a few of them: Can one obtain better matching upper and lower bounds for context-free and linear context-free languages? Similarly, which are better bounds for *deterministic* finite automata?

There are some other interesting and important subfamilies of the context-free languages, e.g., unary, bounded, deterministic or turn-bounded context-free languages. The sizes of the corresponding Higman-Haines sets are worth studying.

Our investigations are based on the special case of the scattered subword relation. Since the result of Higman and Haines only needs a well-partially-order one may ask similar questions for other well-partially-orders as, e.g., for the Parikh subword quasi-order or for monotone well-quasi-orders—see [3, 11] for further results about these well-quasi-orders.

References

1. Buntrock, G. and Otto, F. *Growing context-sensitive languages and Church-Rosser languages*. Inform. Comput. 141 (1998), 1–36.
2. Dassow, J. and Păun, G. *Regulated Rewriting in Formal Language Theory*. Springer, Berlin, 1989
3. Ehrenfeucht, A., Haussler, D., and Rozenberg, G. *On regularity of context-free languages*. Theoret. Comput. Sci. 27 (1983), 311–332.
4. Fernau, H. and Stephan, F. *Characterizations of recursively enumerable sets by programmed grammars with unconditional transfer*. J. Autom., Lang. Comb. 4 (1999), 117–152.
5. Gilman, R. H.. *A shrinking lemma for indexed languages*. Theoret. Comput. Sci. 163 (1996), 277–281.
6. Glaister, I. and Shallit, J. *A lower bound technique for the size of nondeterministic finite automata*. Inform. Process. Lett. 59 (1996), 75–77.
7. Gruber, H. and Holzer, M. *Results on the average state and transition complexity of finite automata*. Descriptional Complexity of Formal Systems (DCFS 2006), University of New Mexico, Technical Report NMSU-CS-2006-001, 2006, pp. 267–275.
8. Haines, L. H. *On free monoids partially ordered by embedding*. J. Combinatorial Theory 6 (1969), 94–98.
9. Higman, G. *Ordering by divisibility in abstract algebras*. Proc. London Math. Soc. 2 (1952), 326–336.
10. M. Holzer and M. Kutrib. *The size of Higman-Haines sets*. Theoret. Comput. Sci., to appear.
11. Ilie, L. *Decision problems on orders of words*. Ph.D. thesis, Department of Mathematics, University of Turku, Finland, 1998.
12. Kruskal, J. B. *The theory of well-quasi-ordering: A frequently discovered concept*. J. Combinatorial Theory 13 (1972), 297–305.
13. van Leeuwen, J. *A regularity condition for parallel rewriting systems*. SIGACT News 8 (1976), 24–27.
14. van Leeuwen, J. *Effective constructions in well-partially-ordered free monoids*. Discrete Mathematics 21 (1978), 237–252.