PRIMING THE PUMP FOR LOWER BOUNDS ON CHOMSKY FORM

by

Harold N. Gabow

Dept. of Computer Science
University of Colorado
Boulder, Colorado 80309

CU-CS-239-82
Abstract

Two simple examples are given to show that transforming a context-free grammar into Chomsky form necessarily increases the size, in at least some cases: The language of nondecreasing pairs, \( L_n = \{ij | 1 \leq i \leq j \leq n \} \), has a context-free grammar of size \( \Theta(n) \), yet the smallest Chomsky normal form grammar has size \( \Theta(n \log \log n) \). The related language \( M_n = \{ij \bar{i} | 1 \leq i < j \leq n \} \) has a context-free grammar of size \( \Theta(n) \), while the smallest Chomsky grammar has size \( \Theta(n \log n) \).
1. Introduction

This note investigates the problem of how an arbitrary context-free grammar expands when it is placed in Chomsky normal form. Chomsky form plays a fundamental role in the analysis of context-free grammars (e.g., in the derivations of Greibach normal form \(H, pp. 113\)), the pumping lemma and Ogden's lemma \([HU, pp. 125-130]\). The question of expansion is relevant to the efficiency of at least two general context-free parsing algorithms—the Cocke-Kasami-Younger algorithm \(H, pp. 430-441\), and Valiant's algorithm \(H, pp. 442-470\), the latter being the asymptotically best parser known. In both algorithms the given grammar is first transformed into Chomsky form. Clearly the efficiency of the parser depends on the efficiency of the transformation process.

In transforming a grammar to Chomsky form, each step but one expands the grammar size by only a linear factor \(H, pp. 98-106\). The nonlinear step is eliminating chain rules. The obvious algorithm can expand a \(\Theta(n)\)-size grammar to \(\Theta(n^2)\) \(H, pp. 101-102\). No better algorithm is known. Blum \([B1]\) has shown that a language \(L_n\) having an \(O(n)\) grammar has Chomsky grammars only of size \(\Omega(n \log \log n)\). Hence there can be no linear transformation to Chomsky form.\(^1\) Blum has also improved the bound to \(\Omega(n^{2-\epsilon})\) \([B2]\).

This note gives simple proofs of lower bounds of \(\Omega(n \log \log n)\) and \(\Omega(n \log n)\) for conversion to Chomsky form. The bounds are not as strong as \([B2]\) but the proofs are simple, and we hope the technique of "priming the pump" will find other applications.

The specific results are as follows. \(L_n\), the language of nondecreasing pairs \(i \leq j, i \leq j\), has an \(O(n)\) grammar (involving chain rules); any grammar for \(L_n\) without chain rules has size \(\Omega(n \log \log n)\). \(M_n\), the related language \(i \leq j, i \leq j\).

\(^1\)A previous claim for a tight \(\Omega(n^2)\) lower bound is made in \([P]\), but this appears to be unproved \([H2]\).
has an $O(n)$ grammar; any grammar without chain rules has size $\Omega(n \log n)$.
(The proof of the second, tighter bound is actually much easier.)

The two proofs are organized as follows. We start with a minimum size grammar and show that it has a certain structure. The structure gives a recursive equation for the size of the grammar, which implies the lower bound.

The hard part of the proof is to deduce the structure of the minimum grammar. For $L_n$, we failed in our attempts to start with an arbitrary minimum grammar and deduce the requisite structure. However a simple device leads to success: We assume part of the desired structure, and deduce the rest. This enables us to deduce a recursive equation that implies the lower bound. We call this approach "priming the pump". (See also the cover of [H]).

Before presenting the results we review some basic facts. (Complete developments can be found in [H] or [HU]). We specify a context-free grammar by giving its productions, where each production has the form $A \rightarrow a$. (Capital letters $A, B, \ldots$ are variables; $S$ is the start symbol; lower case letters $a, b, \ldots$ are terminals.) A production $A \rightarrow \alpha$ is an $A$-rule. A grammar is in Chomsky normal form if every production has the form $A \rightarrow BC$ or $A \rightarrow a$. A chain rule is a production of the form $A \rightarrow B$, and of course is not allowed in Chomsky form. A grammar $G$ has size $|G|$, the number of productions. Although this is not the standard definition of size ([H, p. 94]), all grammars considered in this paper (e.g., Chomsky grammars) have productions of bounded length. For these grammars it is easy to see that our definition differs from the usual one by only a constant factor. Since our results are asymptotic this presents no problem.

As a final convention we use interval notation to denote sets of integers. Thus $(a, b] = \{i \mid i \text{ is an integer}, a < i \leq b\}$, and similarly for the other types of intervals.
2. An $n \log \log n$ bound

This section discusses the language $L_n$ that has $\log \log n$ expansion. To define this language fix an integer $n \geq 1$. Let $\Sigma_n = \{i, \overline{i} \mid 1 \leq i \leq n\}$, an alphabet of $2n$ distinct symbols. Then $L_n$ is the language of nondecreasing pairs of symbols\(^2\), i.e.,

$$L_n = \{i \overline{j} \mid 1 \leq i \leq j \leq n\}.$$ 

$L_n$ has the following context-free grammar with chain rules: $S \to A_1; A_i \to A_{i+1}$ and $B_i \to B_{i+1}$ for $1 \leq i < n$; $A_i \to iB_i$ and $B_i \to \overline{i}$ for $1 \leq i \leq n$. This grammar has size $\Theta(n)$.

Now we show that any Chomsky grammar for $L_n$ has size $\Omega(n \log \log n)$. First we give a simple normalization.

Lemma 2.1. Let $G$ be a Chomsky grammar for $L_n$. There is a Chomsky grammar $G'$ for $L_n$ such that $|G'| \leq |G|$; $G'$ has variables $S$ and $A_i, B_i$ for $1 \leq i \leq n$; every production of $G'$ is of the form $S \to A_kB_j, A_k \to k$ or $B_j \to \overline{i}$; for $l \leq i \leq n$, the variables $A_i$ and $B_i$ satisfy $i \in \{k \mid A_k \to k\} \subseteq \lfloor 1, i \rfloor$ and $i \in \{l \mid B_l \to \overline{l}\} \subseteq \lfloor i, n \rfloor$.

Proof. Without loss of generality assume that $G$ is reduced. Aside from the start symbol $S$ there are two types of variables in $G$: those that derive only unbarred terminals $k$ and those that derive only barred terminals $\overline{i}$. For if $S \to AB$ then $A$ derives only unbarred terminals $k$ and $B$ derives only barred terminals $\overline{i}$.

Now consider a word $i \overline{i} \in L_n$, with a derivation $S \to AB \to iB \to i \overline{i}$. Variable $A$ derives only terminals $k \in \lfloor 1, i \rfloor$ and $B$ derives only terminals $\overline{l}$ where $l \in \lfloor i, n \rfloor$. (Otherwise a word not in $L_n$ can be derived.) Let $A$ and $B$ be $A_i$ and $B_i$ respectively of the Lemma.

\(^2\)In [Y], $L_n$ is suggested as a candidate for a lower bound on Chomsky form. However the conjecture of $\Theta(n \log \log n)$ as the size of a minimum Chomsky grammar is incorrect.
If the $A_i$ and $B_i$ exhaust the variables of $G$ then it is easy to see we are done. So assume that $A$ is a variable that derives only unbarred symbols and is not among the $A_i$. Let $i = \max \{ k \mid A \rightarrow k \}$. Change all occurrences of $A$ in productions to $A_i$. It is easy to see that the grammar remains valid for $L_n$ and the size does not increase.

A similar modification can be done for variables $B$ that derive only barred symbols. The resulting grammar has the desired properties of $G'$. *

Now we formalize the idea of "priming the pump". A "primed grammar" is one that has some of the desired structure, from which the rest of the structure is easily deduced. To give the exact definition, first let $\alpha_i, 0 \leq i \leq \lfloor \sqrt{n} \rfloor$, be a sequence of numbers such that $\alpha_0 = 0, \alpha_i - \alpha_{i-1} \in \{ \lfloor \sqrt{n} \rfloor, \lceil \sqrt{n} \rceil, \lceil \sqrt{n} \rceil + 1 \}$ for $1 \leq i \leq \lfloor \sqrt{n} \rfloor$, and $\alpha_{\lfloor \sqrt{n} \rfloor} = n$. So if $n$ is a perfect square, $\alpha_i = i \sqrt{n}$. It is easy to see that a sequence $\alpha_i$ exists for any $n$, since $\lfloor \sqrt{n} \rfloor^2 \leq n \leq (\lfloor \sqrt{n} \rfloor + 1)^2$.

A Chomsky grammar for $L_n$ is *primed* if for all $i$ in $0 \leq i < \lfloor \sqrt{n} \rfloor$, the set of terminals $\{ \alpha_i, \alpha_{i+1} \}$ is included in both $\{ k \mid A_{\alpha_i+1} \rightarrow k \}$ and $\{ l \mid B_{\alpha_i+1} \rightarrow l \}$. This is illustrated in Figure 1. (Note that the variables $A_{\alpha_i+1}$ and $B_{\alpha_i+1}$ may generate terminals other than those shown in Figure 1.)

Now we deduce the structure of primed grammars.

**Lemma 2.2.** Let $G$ be a primed grammar of minimum size. Let $i$ be any index, $0 \leq i \leq \lfloor \sqrt{n} \rfloor$.

(a) There is a terminal $\alpha_i \in (\alpha_i, \alpha_{i+1}]$ that is only generated by variables $A_j$ with $j \leq \alpha_i + 1$.

(b) There is a terminal $\beta_i$, where $\beta_i \in (\alpha_i, \alpha_{i+1}]$, that is only generated by variables $B_j$ with $j \geq \alpha_i + 1$.

**Proof.** (a) If the Lemma is false then each terminal $k$ in $(\alpha_i, \alpha_{i+1}]$ is generated
by a variable $A_j$ with $j > \alpha_{i+1}$. Such a variable can only be used to derive words $k\bar{l}$ with $l \geq j > \alpha_{i+1}$. So replace the productions 
\{ $A_j \rightarrow k \mid k \in (\alpha_i, \alpha_{i+1}], j > \alpha_{i+1}$ \} by \{ $S \rightarrow A_{\alpha_{i+1}} B_{\alpha_{j+1}} \mid j \geq i+1$ \}. The resulting grammar generates $L_n$ and is primed. Its size is less than $|G|$, since $[\sqrt{n}]$ or more productions are replaced by $[\sqrt{n}] - 1$ or less productions. This contradiction proves the Lemma.

(b) The proof is analogous. ■

Lemma 2.3. Let $G$ be a primed grammar of minimum size. For all indices $i, j$
$0 \leq i < j < [\sqrt{n}]$, $G$ has a production $S \rightarrow A_k B_l$ where $k \in (\alpha_i, \alpha_{i+1}]$, and $l \in (\alpha_j, \alpha_{j+1}]$.

Proof. Consider the terminals $a_i, b_j$ given by Lemma 2.2. $a_i b_j \in L_n$ since $b_j > \alpha_j \geq a_i$. A derivation of $a_i b_j$ begins with a production $S \rightarrow A_k B_l$ of the desired form, by Lemma 2.2. ■

Now consider the triangles on the diagonal of Figure 2.1. More precisely for
$0 \leq i < [\sqrt{n}]$ define the triangle

$T_{i+1} = \{ kl \mid k \leq l \text{ and } k, l \in (\alpha_i+1, \alpha_{i+1}) \}$.

Also define a grammar $G_{i+1} = \{ S \rightarrow A_r B_s \mid S \xrightarrow{A_r} A_r B_s \text{ and } r, s \in (\alpha_i+1, \alpha_{i+1}) \} \cup \{ A_r \rightarrow k \mid A_r \xrightarrow{G} k \text{ and } r, k \in (\alpha_i+1, \alpha_{i+1}) \} \cup \{ B_s \rightarrow \bar{l} \mid B_s \xrightarrow{G} \bar{l} \} \text{ and } s, l \in (\alpha_i+1, \alpha_{i+1}) \}$.

Lemma 2.4. $G_{i+1}$ is a Chomsky grammar for $T_{i+1}$.

Proof. $G_{i+1}$ only generates words of $L_n$ in $(\alpha_i+1, \alpha_{i+1})^2$, so $L(G_{i+1}) \subseteq T_{i+1}$.

To show the opposite inclusion consider a word $k\bar{l} \in T_{i+1}$. It has a derivation in $G$. 


\[ S \xrightarrow{G} A_r B_s \xrightarrow{G} k B_s \xrightarrow{G} k \overline{l}. \]

\( r \leq s \) since \( A_r B_s \xrightarrow{\cdot r \overline{s}} k \), \( k \leq r \) and \( s \leq l \) by definition of \( A_r \) and \( B_s \). Since \( \alpha_{i+1} < k \) and \( l < \alpha_{i+1} \) we deduce \( r, s \in (\alpha_{i+1}, \alpha_{i+1}) \). So the above derivation holds in \( G_{i+1} \) also. \( \ast \)

Now define these quantities:

\[ s(n) = \text{the minimum size of a Chomsky grammar for } L_n; \]

\[ p(n) = \text{the minimum size of a primed grammar for } L_n. \]

**Lemma 2.5.** \( s(n) \) is an increasing function of \( n \).

**Proof.** Consider a minimum Chomsky grammar for \( L_{n+1} \). Deleting every occurrence of the symbols \( n+1 \) and \( \overline{n+1} \) from the grammar gives a grammar for \( L_n \). Thus \( s(n+1) > s(n) \). \( \ast \)

**Lemma 2.6.** \( s(n) = \Omega(n \log \log n) \).

**Proof.** First consider a primed grammar \( G \) for \( L_n \). We count three types of productions in \( G \): (i) the productions \( A_{\alpha_{i+1}} \Rightarrow k \) and \( B_{\alpha_{i+1}} \Rightarrow \overline{l} \) required by the definition of primed grammar; (ii) the productions \( S \Rightarrow A_k B_l \) given by Lemma 2.3; (iii) the productions in \( G_{i+1} \) given by Lemma 2.4. It is easy to see that these three types are mutually exclusive. There are \( 2n \) productions of type (i) and \[ \frac{\sqrt{n}(\lceil \sqrt{n} \rceil - 1)}{2} \] productions of type (ii). For type (iii) notice that \( T_{i+1} \) is isomorphic to the language \( L_r \) where \( r = \alpha_{i+1} - \alpha_i - 2 \geq \lceil \sqrt{n} \rceil - 2 \). This gives the following relation:

\[ p(n) \geq 2n + \frac{\sqrt{n}(\lceil \sqrt{n} \rceil - 1)}{2} + \lceil \sqrt{n} \rceil s(\lceil \sqrt{n} \rceil - 2). \]

\[ \geq 2n + \frac{n}{8} + \lceil \sqrt{n} \rceil s(\lceil \sqrt{n} \rceil - 2), \text{ for } n \geq 16. \]
Now consider a minimum Chomsky grammar for $L_n$. It can be primed by adding at most $2n$ productions of the form $A_{a_{i+1}} \rightarrow k$ and $B_{a_{i+1}} \rightarrow \bar{l}$. Thus $s(n) + 2n \geq p(n)$. So the above inequality implies the following relations:

$$s(n) \geq \frac{n + \lfloor \sqrt{n} \rfloor}{8} s^2 \lfloor \sqrt{n} \rfloor - 2$$, for $n \geq 16$

$$s(n) \geq 1$$ otherwise.

Let $t(n) = \frac{8s(n)}{n}$. Hence

$$t(n) \geq 1 + \frac{\lfloor \sqrt{n} \rfloor - 2}{n} t^2 \lfloor \sqrt{n} \rfloor - 2$$

$$\geq 1 + (1 - \frac{4}{\sqrt{n}}) t^2 \lfloor \sqrt{n} \rfloor - 2$$, for $n \geq 16$

$$t(n) \geq \frac{8}{15}$$ otherwise.

It is easy to verify by induction that for some constant $C$, $t(n) \geq C \log \log n$.

Hence $s(n) = \Omega(n \log \log n)$. *

Now we show that the bound on $s$ is tight.

**Lemma 2.7.** $s(n) = O(n \log \log n)$.

**Proof.** For any $n \geq 2$ construct a grammar $G$ for $L_n$ as follows. Define integers $a_i$, $0 \leq i \leq \lfloor \sqrt{n} \rfloor$, as above. There are three types of productions. The following productions generate terminals:

$A_i \rightarrow k$ for $1 \leq i \leq \lfloor \sqrt{n} \rfloor$ and $k \in (a_{i-1}, a_i]$

$B_i \rightarrow \bar{l}$ for $1 \leq i \leq \lfloor \sqrt{n} \rfloor$ and $k \in [a_i, a_{i+1}]$.

(By convention $a_{\lfloor \sqrt{n} \rfloor + 1} = n$. Also note that $A_i$ and $B_i$ differ from the same symbols used in the lower bound proof.) The following productions generate words of $L_n$ not in the diagonal triangles:
Finally to generate the diagonal triangles \((\alpha_{i-1}, \alpha_i)^2 \cap L_n\), for \(1 \leq i \leq \sqrt{n}\) let \(G_i\) be a minimum size Chomsky grammar for \(\{kl | k \leq l\) and \(k, l \in (\alpha_i, \alpha_{i+1})\}\), with start symbol \(S_i\) (and all variables distinct from those of other grammars). Replace \(S_i\) by \(S\) and add all productions of \(G_i\) to \(G\).

It is easy to verify that \(G\) generates \(L_n\). This implies the following recurrence for \(s(n)\):

\[
s(n) \leq 4n + \sqrt{n}s(\sqrt{n}), \text{ for } n \geq 2;
\]

\[
s(1) = 3.
\]

As in Lemma 2.6 it is easy to verify that \(s(n) = O(n \log \log n)\). •

Now we summarize the results.

Theorem 2.1. \(L_n\) is a context-free language with a grammar of size \(\Theta(n)\), and smallest Chomsky form grammar of size \(\Theta(n \log \log n)\). •

Several languages related to \(L_n\) have the same \(\log \log n\) increase in size. For instance, fix an integer \(k \geq 2\), and consider the languages of nondecreasing \(k\)-sequences. More specifically the language is

\[
\{(1, i_1) \cdots (k, i_k) | 1 \leq i_1 \cdots \leq i_k \leq n\}. 
\]

Here the symbols \((j, i_j)\) are the terminal of the languages. So for \(k = 2\) the language is \(L_n\). Each of these languages has a \(\Theta(n)\) grammar with chain rules, but the smallest Chomsky grammar is \(\Theta(n \log \log n)\).

3. An \(n \log n\) bound

This section discusses the language \(M_n\) that has \(\log n\) expansion. To define \(M_n\) fix an integer \(n \geq 1\), with \(\Sigma_n\) as in Section 2. Then \(M_n\) is a variant of \(L_n\):
\[ M_n = \{ i \overline{j} i \mid 1 \leq i \leq j \leq n \}. \]

\( M_n \) has this context-free grammar with chain rules:

\[ S \rightarrow A_1; A_i \rightarrow A_{i+1} \text{ and } B_i \rightarrow B_{i+1} \text{ for } 1 \leq i < n; A_i \rightarrow i B_i i \text{ and } B_i \rightarrow \overline{i} \text{ for } 1 \leq i \leq n. \]

This grammar has size \( \Theta(n) \).

We analyze Chomsky grammars for \( M_n \) in two steps. First we show that a Chomsky grammar for \( M_n \) is essentially a regular grammar for \( L_n \). Then we analyze regular grammars for \( L_n \) and show they have size \( \Omega(n \log n) \).

**Lemma 3.1.** Let \( G \) be a Chomsky grammar for \( M_n \). Then there is a regular grammar \( G' \) for \( L_n \) with \( |G'| \leq |G| \).

**Proof.** Without loss of generality assume that \( G \) is reduced. Say that \( C \) is an \( i \)-variable if there is only one \( C \)-rule, \( C \rightarrow i \). Define the regular grammar \( G' \) to contain these productions: (1) any production of \( G \) of the form \( D \rightarrow \overline{j} \); (2) a production \( S \rightarrow i D \) if \( G \) has a rule \( A \rightarrow CD \) where \( A \neq S \) and \( C \) is an \( i \)-variable; (3) a production \( S \rightarrow i C \) if \( G \) has a rule \( B \rightarrow CD \) where \( B \neq S \) and \( D \) is an \( i \)-variable.

First note \( L_n \subseteq (G') \). For if \( i \overline{j} i \in L_n \) then \( i \overline{j} i \in M_n \), so \( G \) has a derivation \( S \rightarrow AB \overset{*}{\rightarrow} i \overline{j} i \). Either \( A \overset{*}{\rightarrow} i \overline{j} \) or \( B \overset{*}{\rightarrow} \overline{j} i \). In the first case it is easy to see that every \( A \)-rule has the form \( A \rightarrow CD \) where \( C \) is an \( i \)-variable (otherwise a word not in \( M_n \) can be generated). In particular there is a rule \( A \rightarrow CD \) where \( C \) is an \( i \)-variable and \( D \rightarrow \overline{j} \). So in \( G' \), \( S \overset{*}{\rightarrow} i \overline{j} \) by productions of type(2) and (1) above.

A similar argument applies to the second case.

Next note \( L(G') \subseteq L_n \). For suppose \( G' \) has a type(2) production \( S \rightarrow i D \) corresponding to the \( G \) rule \( A \rightarrow CD \), \( C \) an \( i \)-variable. It is easy to see (from the above paragraph) that \( G \) has a production \( S \rightarrow AB \). So a derivation in \( G' \), \( S \rightarrow i D \rightarrow i \overline{j} \), gives a derivation in \( G \), \( S \rightarrow AB \rightarrow Ai \rightarrow CDi \rightarrow C\overline{j}i \rightarrow i \overline{j} i \). Thus \( i \leq j \) as desired.
We conclude \( L(G') = I_n \). •

To analyze regular grammars for \( I_n \) we prime the pump, as follows. In a regular grammar for \( I_n \), a variable \( A \neq S \) is an \( A_j \)-variable if \( j = \min\{k \mid A \rightarrow k\} \). A regular grammar for \( I_n \), \( n \geq 2 \), is primed if there is an \( A_{\lfloor n/2 \rfloor} \)-variable \( A \) where \( \{k \mid A \rightarrow k\} = [\lfloor n/2 \rfloor, n] \). Without loss of generality this variable is unique, and we refer to it as \( A_{\lfloor n/2 \rfloor} \).

Lemma 3.2. There is a primed grammar of minimum size where for all \( i \),
\[
1 \leq i \leq \lfloor n/2 \rfloor, \quad S \rightarrow i \ A_{\lfloor n/2 \rfloor}.
\]

Proof. Let \( G \) be a primed grammar of minimum size. Not every \( j \), \( j \geq \lfloor n/2 \rfloor \), is generated by an \( A_k \)-variable, \( i \leq \lfloor n/2 \rfloor \). For suppose otherwise. Productions \( A_i \rightarrow j \), \( i \leq \lfloor n/2 \rfloor \leq j \), can only be used to derive words \( k j \), \( k \leq i \). So replace all such productions by \( S \rightarrow i \ A_{\lfloor n/2 \rfloor} \), \( 1 \leq i \leq \lfloor n/2 \rfloor \). The resulting grammar generates \( I_n \) and is primed. Its size is less than \( |G| \), since at least \( n - \lfloor n/2 \rfloor + 1 = \lfloor n/2 \rfloor + 1 \) productions are replaced by \( \lfloor n/2 \rfloor \) productions. This contradiction shows that there is some \( j \), \( j \geq \lfloor n/2 \rfloor \), not generated by any \( A_k \)-variable, \( i \leq \lfloor n/2 \rfloor \).

So for any \( i \leq \lfloor n/2 \rfloor \), the word \( i j \) is generated from a production \( S \rightarrow i \ A_k \), \( k > \lfloor n/2 \rfloor \). We can replace this production by \( S \rightarrow i \ A_{\lfloor n/2 \rfloor} \), as desired. •

Now define these quantities:
\[
s(n) = \text{the minimum size of a regular grammar for } I_n.
\]
$p(n) =$ the minimum size of a primed grammar for $L_n$.

Lemma 3.3. $s(n) = \Omega(n \log n)$.

Proof. Consider a primed grammar of minimum size for $L_n$. There are \( \lfloor \frac{n}{2} \rfloor + 1 \) productions $A_{\lfloor \frac{n}{2} \rfloor} \rightarrow \bar{j}$, and $\lfloor \frac{n}{2} \rfloor$ productions using $A_{\lfloor \frac{n}{2} \rfloor}$ from Lemma 3.2. The remaining productions partition into a grammar for $L_{\lfloor \frac{n}{2} \rfloor-1}$ and a grammar on the numbers $[\lfloor \frac{n}{2} \rfloor + 1, n]$ that is isomorphic to $L_{\lfloor \frac{n}{2} \rfloor}$. Thus $p(n) \geq n + 1 + s(\lfloor \frac{n}{2} \rfloor - 1) + s(\lfloor \frac{n}{2} \rfloor)$, for $n \geq 2$.

Now consider a minimum regular grammar for $L_n$. It can be primed by adding at most $\lfloor \frac{n}{2} \rfloor$ rules $A_{\lfloor \frac{n}{2} \rfloor} \rightarrow \bar{j}$. So $s(n) + \lfloor \frac{n}{2} \rfloor \geq p(n)$, and the above inequality shows

$s(n) \geq \lfloor \frac{n}{2} \rfloor + 1 + s(\lfloor \frac{n}{2} \rfloor - 1) + s(\lfloor \frac{n}{2} \rfloor)$, for $n \geq 2$.

$s(1) \geq 1$.

It is easy to verify by induction that for some constant $C$, $s(n) \geq Cn \log n$, as desired. *

Now we show that the bound on $s$ is tight.

Lemma 3.4. $s(n) = O(n \log n)$.

Proof. For any $n \geq 2$ construct a grammar $G$ for $L_n$ as follows. First introduce production for a "primed" variable $A$:

$A \rightarrow \bar{j}$, for $\lfloor \frac{n}{2} \rfloor \leq j \leq n$;
\( S \rightarrow iA, \text{ for } 1 \leq i \leq \lfloor \frac{n}{2} \rfloor. \)

Let \( G_1 \) be a minimum size regular grammar for \( L_{\lfloor \frac{n}{2} \rfloor+1} \); let \( G_2 \) be a minimum size regular grammar for the analogous language over \( \lfloor \frac{n}{2} \rfloor + 1, n \rfloor. \) Let \( G_i \) have start symbol \( S_i, i = 1, 2, \) and all variables distinct from those of the other grammar. Replace \( S_i \) by \( S \) and add all productions of \( G_i \) to \( G. \)

Clearly \( L(G) = L_n. \) This gives the following recurrence:

\[
\begin{align*}
    s(n) & \leq n + 1 + s\left(\lfloor \frac{n}{2} \rfloor - 1 \right) + s\left(\left\lceil \frac{n}{2} \right\rceil \right), \text{ for } n \geq 2; \\
    s(1) & = 2.
\end{align*}
\]

It is easy to verify that \( s(n) = O(n \log n). \)

Lemmas 3.1 and 3.4 imply the main result.

**Theorem 3.1.** \( M_n \) is a context-free language with a grammar of size \( \Theta(n) \), and smallest Chomsky form grammar of size \( \Theta(n \log n) \).}

**Acknowledgement**

The author thanks Prof. Michael Harrison of the University of California at Berkeley for his active interest and encouragement of this work.
References


Figure 1.

A primed grammar.