Rational index of languages defined by grammars with bounded dimension of parse trees[⋆]

Ekaterina Shemetova[1,3][0000−0002−1577−8347], Alexander Okhotin[1][0000−0002−1615−2725], and Semyon Grigorev[2][0000−0002−7966−0698]

[1] Department of Mathematics and Computer Science, St. Petersburg State University, 14th Line V. O., 29, Saint Petersburg 199178, Russia
alexander.okhotin@spbu.ru

[2] Department of Mathematics and Mechanics, St. Petersburg State University, 7/9 Universitetskaya nab., Saint Petersburg 199034, Russia
semyon.grigorev@spbu.ru

[3] St. Petersburg Academic University, ul. Khlopina, 8, Saint Petersburg 194021, Russia
katyacyfra@gmail.com

Abstract. The rational index ρ_L of a language L is an integer function, where $\rho_L(n)$ is the maximum length of the shortest string in $L \cap R$, over all regular languages R recognized by n-state nondeterministic finite automata (NFA). This paper investigates the rational index of languages defined by grammars with bounded tree dimension, and shows that it is polynomial in n. More precisely, it is proved that for a context-free grammar with tree dimension bounded by d, its rational index is at most $O(n^{2d})$, and that this estimation is asymptotically tight, as there exists a grammar with rational index $\Theta(n^{2d})$. For a multi-component grammar of rank k and with tree dimension bounded by d, the rational index is bounded by $O(n^{2kd})$, and there exists a grammar with rational index $\Omega(n^{2kd})$.

Keywords. Dimension of a parse tree; Strahler number; rational index; context-free grammars; multiple context-free grammars.

1 Introduction

The notion of a rational index of a language was introduced by Boasson, Courcelle and Nivat [3] as a complexity measure for context-free languages. The rational index ρ_L of a language L is an integer function, where $\rho_L(n)$ is the maximum length of the shortest string in a language of the form $L \cap R$, where R is a regular language recognized by an n-state nondeterministic finite automaton (NFA), and the maximum is taken over all such languages R with $L \cap R \neq \varnothing$.

Besides its theoretical value as a measure of complexity of a language, the rational index is useful in determining the parallel complexity of practical problems, such as the CFL-reachability problem and the more general Datalog query

[⋆] Research supported by the Russian Science Foundation, project 18-11-00100.

evaluation. The *CFL-reachability problem* is stated as follows: for a context-free grammar G given an NFA A over the same alphabet, determine whether $L(G) \cap L(A)$ is non-empty. With A regarded as a labelled graph, this is a kind of graph reachability problem with path constraints defined by a context-free grammar. This is an important problem used in static code analysis [26] and graph database query evaluation [32].

The CFL-reachability problem is P-complete already for a fixed context-free grammar [12]. The question on the parallel complexity of this problem was investigated by Ullman and Van Gelder [29] in a much more general case, with a rich logic for database queries instead of grammars, and it was proved that under an assumption called the *polynomial fringe property* the problem is decidable in NC [29]. In the special case of grammars, the result of Ullman and Van Gelder [29] gives an NC^2 algorithm for the CFL-reachability problem, under the assumption that the grammar's rational index is polynomial.

Theoretical properties of the rational index have received some attention in the literature. Pierre and Farinone [24] proved that for every algebraic number $\gamma \geqslant 1$, there is a context-free grammar with a rational index of the order $\Theta(n^\gamma)$. An upper bound on the rational index of a context-free language, shown by Pierre [23], is $2^{\Theta(n^2/\ln n)}$, and this bound is reached on the Dyck language on two pairs of parentheses. For several important subfamilies of grammars, such as the linear and the one-counter languages, there are polynomial upper bounds on the rational index, which imply that the CFL-reachability problem is in NC^2; they can be proved to lie in NL by direct methods not involving the rational index [14, 16].

Other problems on the length of shortest strings have received some attention in literature. Chistikov et al. [5] investigated the length of shortest strings in *one-counter languages*, and, in particular, proved that their rational index is $O(n^2)$. Ellul et al. [8] studied the length of the shortest string which is not accepted by an NFA. Alpoge et al. [1] found upper and lower bounds on the length of shortest strings in regular languages specified in various ways. The maximum length of shortest strings for deterministic two-way finite automata (2DFA) has been investigated in some recent papers [7, 17, 20].

This paper investigates the rational index of a generalization of linear languages: the *languages of bounded tree dimension*, that is, those defined by context-free grammars with a certain limit on branching in the parse trees. The notion of tree dimension is well-known in the literature, and appears under different names: Chytil and Monien [6] use the term *k-caterpillar trees*, Lohrey et al. [18] call this *Horton–Strahler number*, Esparza et al. [10] use the term *Strahler number* of a tree and mention numerous applications and alternative names for this notion, while Luttenberger and Schlund [19] use the term *tree dimension*, which is adopted in this paper.

Linear languages are languages of tree dimension 1, and their rational index is known to be $O(n^2)$ [3]. It can be derived from the work of Chytil and Monien [6] that languages of tree dimension bounded by d have rational index $O(n^{2d})$: this is explained in Section 3 of this paper. The new result of this paper, presented in Section 4, is that, for every d, there is a language of tree dimension bounded by d with rational index $\Theta(n^{2d})$.

The second contribution of this paper concerns another important family of grammars known in the literature as multiple context-free grammars [27] and as linear context-free rewriting systems [30], and hereinafter called *multi-component grammars*. Instead of defining substrings, these grammars define k-tuples of substrings, for bounded k, and otherwise are the same as ordinary (context-free) grammars. In particular, they have similar parse trees, to which the notion of the tree dimension is equally applicable. The relevant definitions are given in Section 5. In the subsequent Section 6, the Chytil–Monien lemma is extended to multi-component grammars of tree dimension bounded by d: they are proved to have rational index $O(n^{2kd})$. In Section 7, a matching lower bound on the rational index is established, thus demonstrating that it is of the order $\Theta(n^{2kd})$ in the worst case.

Some implications of these results are presented in Section 8. The maximum order of magnitude of the rational index is determined for *superlinear languages* [4], and some bounds are obtained for *languages of bounded oscillation* [11, 31], and for the linear subclass of multi-component grammars [9, 15].

In the final Section 9, the results of this paper are adapted to LL(1)-grammars in the Greibach normal form, and their potential application to conjunctive and Boolean grammars is discussed.

2 Definitions

A *(context-free) grammar* is a quadruple $G = (\Sigma, N, R, S)$, where Σ is an alphabet; N is a set of nonterminal symbols; R is a set of rules, each of the form $A \to \alpha$, with $A \in N$ and $\alpha \in (\Sigma \cup N)^*$; and $S \in N$ is the start symbol. A parse tree is a tree, in which every leaf is labelled with a symbol from Σ, while every internal node is labelled with a nonterminal symbol $A \in N$ and has an associated rule $A \to X_1 \ldots X_\ell \in R$, so that the node has ℓ ordered children labelled with $X_1, \ldots, X_\ell$. The language defined by each nonterminal symbol $A \in N$, denoted by $L_G(A)$, is the set of all strings $w \in \Sigma^*$, for which there exists a parse tree, with A as a root and with the leaves forming the string w. The language defined by the grammar is $L(G) = L_G(S)$.

A grammar G is said to be is in the *Chomsky normal form*, if all rules of R are of the form $A \to BC$, with $B, C \in N$, or of the form $A \to a$, with $a \in \Sigma$.

A grammar is *linear* if every rule is either of the form $A \to uBv$, with $u, v \in \Sigma^*$ and $B \in V$, or of the form $A \to w$, with $w \in \Sigma^*$.

A *nondeterministic finite automaton* (NFA) is a quintuple $\mathcal{A} = (\Sigma, Q, Q_0, \delta, F)$, where Q is a finite set of states, Σ is a finite set of input symbols, $Q_0 \subseteq Q$ is the set of initial states, $\delta \colon Q \times \Sigma \to 2^Q$ is the transition function, $F \subseteq Q$ is the set of accepting states. It accepts a string $w = a_1 \ldots a_n$ if there is a sequence of states $q_0, \ldots, q_n \in Q$ with $q_0 \in Q_0$, $q_i \in \delta(q_{i-1}, a_i)$ for all i, and $q_n \in F$. The language of all strings accepted by $\mathcal{A}$ is denoted by $L(\mathcal{A})$.

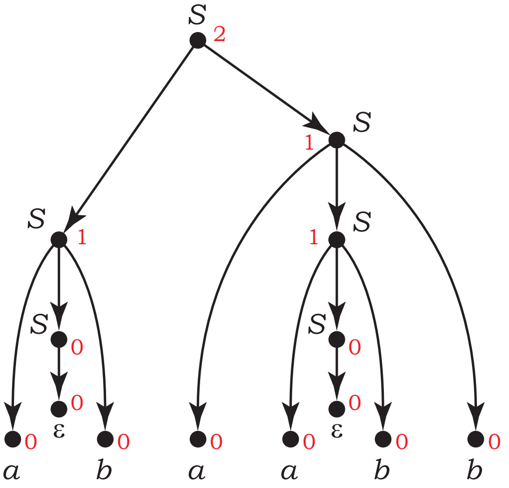

Fig. 1. A parse tree with marked dimensions of its subtrees.

An NFA is said to be a *partial deterministic finite automaton* (DFA), if $|Q_0| = 1$ and $|\delta(q, a)| \leqslant 1$ for all $q \in Q$ and $a \in \Sigma$. In this paper, all DFA are partial.

For a language L over an alphabet Σ, its rational index ρ_L is a function defined as follows:

$$\rho_L(n) = \max_{\substack{\mathcal{A}\text{: NFA with } n \text{ states} \\ L \cap L(\mathcal{A}) \neq \varnothing}} \; \min_{w \in L \cap L(\mathcal{A})} |w|$$

Tree dimension. For each node v in a parse tree t, its *dimension* $\dim v$ is an integer representing the amount of branching in its subtree. It is defined inductively: a leaf v has dimension 0. For an internal node v, if one of its children $v_1, v_2, \ldots, v_k$, with $k \geqslant 1$, has a greater dimension than all the others, then v has the same dimension, and if there are multiple children of maximum dimension, then the dimension of v is greater by one.

$$\dim v = \begin{cases} \max_{i \in \{1,\ldots,k\}} \dim v_i & \text{if there is a unique maximum} \\ \max_{i \in \{1,\ldots,k\}} \dim v_i + 1 & \text{otherwise} \end{cases}$$

The dimension of a parse tree t, denoted by $\dim t$, is the dimension of its root. An example of a parse tree with marked dimensions of its nodes is given in Figure 1, it uses a grammar for the Dyck language ($S \to SS \mid aSb \mid \varepsilon$).

Definition 1 (Grammars of bounded tree dimension). *Let $d \geqslant 1$. A grammar G is said to be of tree dimension bounded by d, if every parse tree t of G has $\dim t \leqslant d$, The least such constant d is called the dimension of G, denoted by $\dim G = d$.*

3 Upper bound on the rational index

The first result of this paper is that, if the dimension of trees in a grammar is bounded by a constant d, then the rational index of its language is bounded by $O(n^{2d})$, where the constant factor depends upon the grammar.

Theorem 1. *Let G be a grammar of tree dimension bounded by d, and let $\mathcal{A}$ be an NFA with n states, with non-empty intersection $L(G) \cap L(\mathcal{A})$. Then the length of the shortest string in $L(G) \cap L(\mathcal{A})$ is in $O(n^{2d})$.*

The main component of the proof is the following lemma by Chytil and Monien [6], which they used in their study of unambiguous grammars of finite index.

Lemma 1 (Chytil and Monien [6, Lem. 7]). *Let $G = (\Sigma, N, R, S)$ be a grammar, let m be the maximal length of the right-hand side of its rules, and assume that there exists a parse tree of some dimension $d \geqslant 1$ in this grammar. Then the grammar defines some string of length at most $(|N|(m-1)+1)^d$.*

Proof. A proof is included for completeness; later in Section 6 it will be generalized for multi-component grammars.

For each nonterminal A that has at least one parse tree of some dimension d, it is proved that A defines some string of length at most $(|N|(m-1)+1)^d$. The proof proceeds by induction on d.

Base case: $d = 0$. If A defines a tree of dimension 0, then the yield of this tree is a string of length 0 or 1.

Induction step: $d-1 \to d$. Let w be the shortest string defined by A with a parse tree of dimension at most d, and among all such parse trees, let t be the one with the fewest nodes. Consider the path in t that proceeds from its root and passes through nodes of dimension d. If all children of the root have dimension $d-1$, then this path consists of a single node; otherwise, one of the children of the root has dimension d, and other children have dimension less than d, and the path continues to the child of dimension d, etc. Such a path is illustrated in Figure 2.

Let $A_1, \ldots, A_h$ be the nonterminals in the labels of nodes on this path, with $A_1 = A$. No nonterminal symbol is repeated twice in this sequence, because if $A_i = A_j$ for some $i < j$, then the segment of this path from A_{i+1} to A_j could be contracted, and the resulting parse tree would either define a string shorter than w, or would be a smaller tree of w, while still having dimension at most d. Therefore, $h \leqslant |N|$.

For each node on this path except the last one, at most $m-1$ subtrees may spawn off to the left and to the right, and each of them has dimension less than d. The last node on this path has at most m children, all of dimension less than d. Overall, there are at most $(m-1) \cdot (h-1) + m \leqslant (m-1)(|N|-1) + m = |N|(m-1)+1$ subtrees, each of dimension at most $d-1$. Each of these subtrees, with some root $X \in \Sigma \cup N$, defines one of the shortest strings in $L_G(X)$, and therefore, by the induction hypothesis, the substring in each subtree is of length

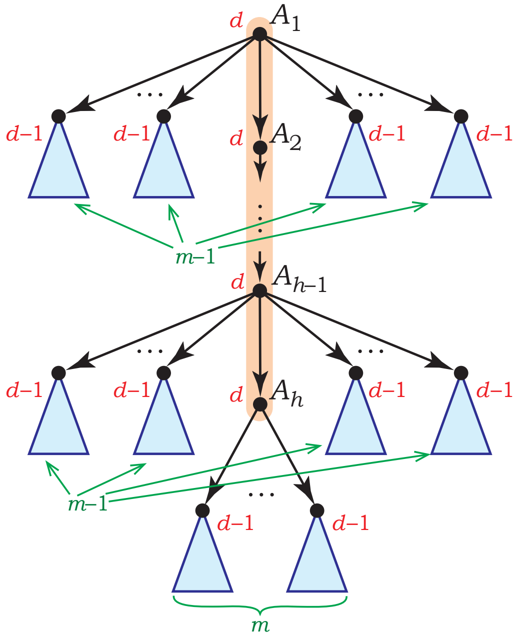

Fig. 2. The path passing through nodes of dimension d in the proof of the Chytil–Monien lemma.

at most $(|N|(m-1)+1)^{d-1}$. Since w is the concatenation of these substrings, its length is at most $(|N|(m-1)+1)^{d-1} \cdot (|N|(m-1)+1) = (|N|(m-1)+1)^d$, as claimed. □

Proof (of Theorem 1). Let N be the set of nonterminal symbols in G. A grammar G' for the language $L(G) \cap L(\mathcal{A})$ is obtained from G and $\mathcal{A}$ by the classical construction by Bar-Hillel et al. [2], which produces $|N| \cdot n^2 + 1$ nonterminal symbols: these are all triples of the form (A, p, q), where $A \in N$ and p, q are two states of the automaton, as well as a new start symbol. Furthermore, each parse tree in the grammar G' has the same structure as some parse tree in G, and differs only in the labelling of internal nodes; in particular, it has the same dimension.

Since G' defines at least one string, the parse tree of that string has the same dimension as some parse tree in G, and its dimension is therefore at most d. Let m be the maximum length of the right-hand sides of rules in G'. Then, by Lemma 1, the length of the shortest string defined by G' is at most $((|N| \cdot n^2 + 1)(m-1)+1)^d = |N| \cdot (m-1) \cdot n^{2d} + o(n^{2d}) = O(n^{2d})$. □

4 Lower bound on the rational index

The upper bound $O(n^{2d})$ on the rational index of a language defined by a grammar with tree dimension bounded by d has a matching lower bound $\Omega(n^{2d})$. It

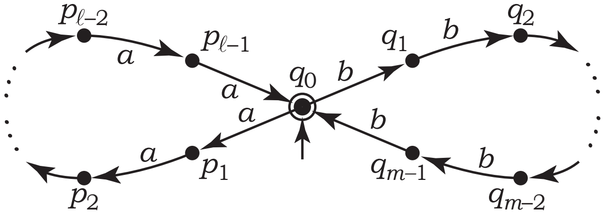

Fig. 3. DFA $\mathcal{B}$ defined in Lemma 2 for $d = 1$.

is first established for a convenient infinite set of values of n, to be extended to arbitrary n in the following.

Lemma 2. *For every $d \geqslant 1$, there is a grammar G of tree dimension bounded by d, such that for every $n \geqslant 2^{d+1}$ divisible by 2^d there is an n-state partial DFA $\mathcal{B}$, such that the shortest string w in $L(G) \cap L(\mathcal{B})$ is of length at least $\frac{1}{2^{d^2+3d-3}}n^{2d}$.*

Proof. The proof is carried out by induction on d. A grammar is constructed for each d, and then, for every n divisible by 2^d, an n-state DFA with the stated property is defined. Each constructed automaton shall have a unique initial state, which is also the unique accepting state.

Basis: $dim(G) = 1$. The family of languages having dimension $d = 1$ coincides with the family of linear languages. Let G be a linear grammar with the rules $S \to aSb \mid ab$, which defines the language $L(G) = \{\, a^i b^i \mid i \geqslant 1 \,\}$.

For every $n \geqslant 4$ divisible by $2^d = 2$, let $\ell = \frac{n}{2}$ and $m = \frac{n}{2} + 1$. Then ℓ and m are coprime integers. Define a DFA $\mathcal{B}$ over the alphabet $\{a, b\}$, which consists of two cycles sharing one node, q_0, which is both the initial and the unique accepting state. The cycle of length ℓ has all transitions by a, and the other by b, as shown in Figure 3. The automaton has $\ell + m - 1 = n$ states.

Every string in $L(G) \cap L(\mathcal{B})$ is of the form $a^i b^i$, with $i \geqslant 1$. For the automaton to accept it, i must be divisible both by ℓ and by m. Since the cycle lengths are relatively prime, the shortest string w with this property has $i = \ell m$, and is accordingly of length $2\ell m$. Its growth with n is estimated as follows.

$$|w| = 2\ell m = 2\frac{n}{2} \cdot \Big(\frac{n}{2} + 1\Big) = \frac{1}{2}n^2 + n$$

This example is well-known to the community [13, 32].

Induction step: $dim(G) = d$. By the induction hypothesis, there is a grammar $\widehat{G} = (\widehat{\Sigma}, \widehat{N}, \widehat{R}, \widehat{S})$ of bounded tree dimension $dim(\widehat{G}) = d-1$, which satisfies the statement of the lemma. The new grammar $G = (\Sigma, N, R, S)$ of tree dimension at most d is defined over the alphabet $\Sigma = \widehat{\Sigma} \cup \{a, b, c\}$, where $a, b, c \notin \widehat{\Sigma}$ are new symbols. It uses nonterminal symbols $N = \widehat{N} \cup \{S, A\}$, adding two new nonterminals $A, S \notin \widehat{N}$ to those in $\widehat{G}$, where S is the new initial symbol. Its set

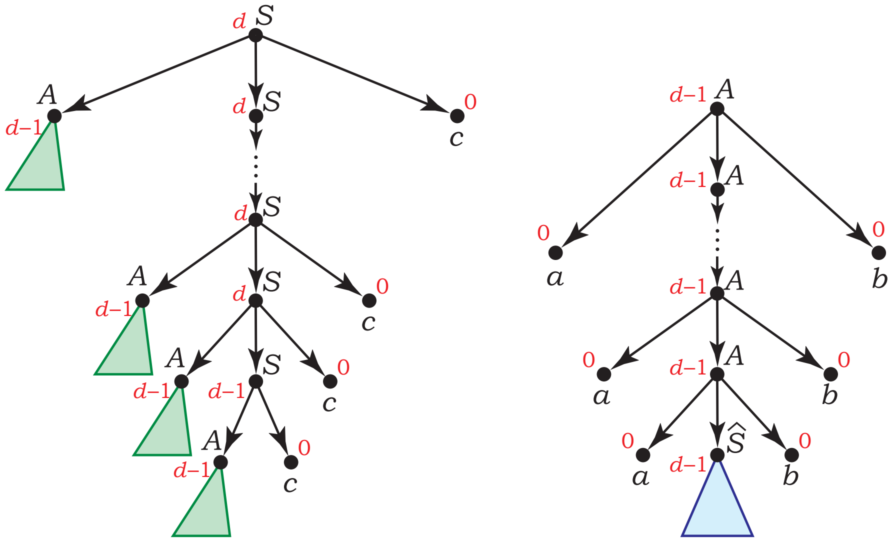

Fig. 4. Parse trees for S and for A, annotated with dimensions of their nodes.

of rules includes all rules from $\widehat{G}$ and the following new rules.

$$S \to ASc \mid Ac$$
$$A \to aAb \mid a\widehat{S}b$$

Here the nonterminal symbol A defines all substrings of the form $a^i u b^i$, with $i \geqslant 1$ and $u \in L(\widehat{G})$, and hence the grammar defines the following language.

$$L(G) = \{\, a^{i_1} w_1 b^{i_1} \ldots a^{i_t} w_t b^{i_t} c^t \mid t \geqslant 1,\ i_1, \ldots, i_t \geqslant 1,\ w_1, \ldots, w_t \in L(\widehat{G}) \,\}$$

To see that trees in the new grammar have dimension at most d, first consider the dimension of any parse tree t with the root labeled by the nonterminal A, which is of the form shown in Figure 4(right). The dimension of the $\widehat{S}$-subtree at the bottom is at most $d-1$ by the properties of $\widehat{G}$. This dimension is inherited by all A-nodes in the tree, because their remaining children are leaves.

Now consider the dimension of a complete parse tree t with the start symbol S in the root, as in Figure 4(left). All A-subtrees in this tree have dimension at most $d-1$. Then the bottom S-subtree, which uses the rule $S \to Ac$, also has dimension at most $d-1$. Every S-subtree higher up in the tree uses a rule $S \to ASc$, and its dimension is at most d, because getting a higher dimension would require two subtrees of dimension d, which is never the case.

Now, for every $n \geqslant 2^{d+1}$ divisible by 2^d, the goal is to construct an n-state DFA over the alphabet Σ, so that the shortest string w in $L(G) \cap L(\mathcal{B})$ is of length at least $\frac{1}{2^{d^2+3d-3}} n^{2d}$. Since the number $\frac{n}{2}$ is at least 2^d and is divisible by 2^{d-1}, the induction hypothesis for the grammar $\widehat{G}$ asserts that there is a DFA

$\widehat{\mathcal{B}} = (\widehat{Q}, \widehat{\Sigma}, \widehat{\delta}, \widehat{q}_0, \{\widehat{q}_0\})$, with $\frac{n}{2}$ states, with the shortest string $\widehat{w}$ in $L(\widehat{G}) \cap L(\widehat{\mathcal{B}})$ of length at least $\frac{1}{2^{(d-1)^2+3(d-1)-3}}(\frac{n}{2})^{2(d-1)}$.

The desired n-state DFA $\mathcal{B} = (\Sigma, Q, q_0, \delta, \{q_0\})$ is constructed as follows. Let $\ell = \frac{n}{4}$ and $m = \frac{n}{4} + 1$, these are two coprime integers. The set of states of $\mathcal{B}$ contains all $\frac{n}{2}$ states from $\widehat{Q}$, in which $\mathcal{B}$ operates as $\widehat{\mathcal{B}}$, and $m + \ell - 1 = \frac{n}{2}$ new states forming a cycle of length ℓ and a chain of length m, which share a common state q_0.

$$Q = \widehat{Q} \cup \{p_1, \ldots, p_{\ell-1}, q_0, \ldots, q_{m-1}\}$$

The new initial state q_0 has a transition by a leading to the initial state of $\widehat{\mathcal{B}}$, from where one can return to q_1 by b.

$$\delta(q_0, a) = \widehat{q}_0$$
$$\delta(\widehat{q}_0, b) = q_1$$

There is a chain of transitions by a from q_{m-1} to q_0, and another chain b in the opposite direction, from q_1 to q_{m-1} and back to q_0.

$$\delta(q_i, a) = q_{i-1}, \qquad \text{with } 1 \leqslant i \leqslant m-1$$
$$\delta(q_i, b) = q_{i+1 \bmod m}, \qquad \text{with } 1 \leqslant i \leqslant m-1$$

There is a cycle by c in the states $q_0, p_1, \ldots, p_{\ell-1}$; for uniformity, denote $p_0 = q_0$.

$$\delta(p_i, c) = p_{i+1 \bmod \ell}, \qquad \text{with } 0 \leqslant i \leqslant \ell-1$$

The general form of $\mathcal{B}$ is shown in Figure 5.

Let w be any string in $L(G) \cap L(\mathcal{B})$. Since w is defined by G, it is of the form $w = a^{i_1} w_1 b^{i_1} \ldots a^{i_t} w_t b^{i_t} c^t$, for some $t \geqslant 1$, $i_1, \ldots, i_t \geqslant 1$ and $w_1, \ldots, w_t \in L(\widehat{G})$. At the same time, w must be accepted by $\mathcal{B}$, and the automaton's behaviour on w is explained in the following claim.

Claim. After reading each prefix $a^{i_1} w_1 b^{i_1} \cdots a^{i_s} w_s b^{i_s}$ of w, with $s \in \{0, \ldots, t\}$, the automaton comes to the state $q_{s \bmod m}$.

Proof. The claim is proved by an induction on s. For the base case, $s = 0$, it holds true, because the initial state is q_0.

For the induction step, assume that the automaton is in the state $q_{(s-1) \bmod m}$ after reading $a^{i_1} w_1 b^{i_1} \cdots a^{i_{s-1}} w_{s-1} b^{i_{s-1}}$ and it reads the next block $a^{i_s} w_s b^{i_s}$; it is claimed that it must finish reading this block in the state $q_{s \bmod m}$. For the automaton to read any symbols of w_s, it must come to the state $\widehat{q}_0$, which is possible only if $i_s = ((s-1) \bmod m) + 1$; hence, i_s is at most m. Similarly, for the automaton to continue reading b^{i_s} after processing w_s, it must finish reading w_s in the state $\widehat{q}_0$. Then, by reading the string b^{i_s}, it gets to the state $q_{i_s \bmod m}$. If $i_s < m$, then $(s-1) \bmod m < m-1$ and $i_s = s \bmod m$, and the state reached is $q_{s \bmod m}$, as claimed (this is the case illustrated in Figure 5). If $i_s = m$, then $(s-1) \bmod m = m-1$, and the automaton reaches q_0; since $s \bmod m = 0$, this is the claimed state. The proof of the claim is complete. □

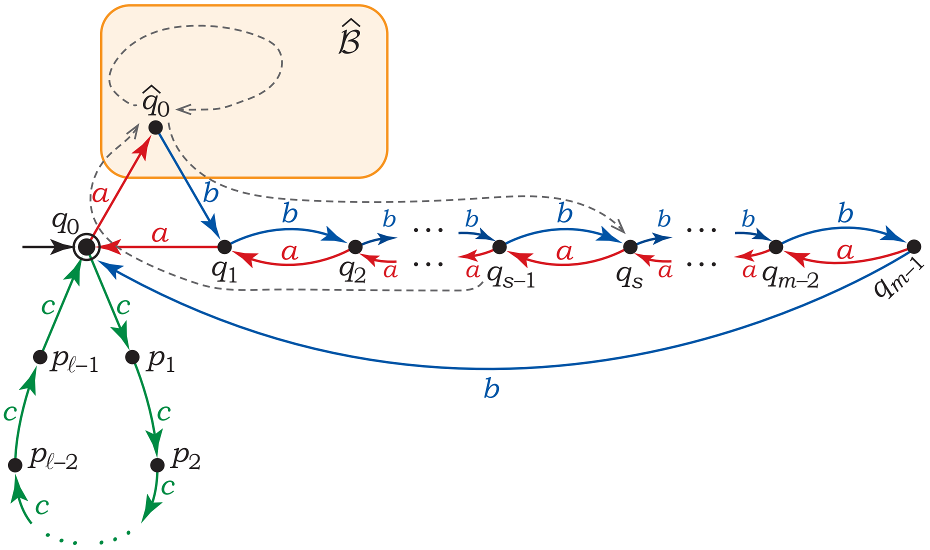

Fig. 5. DFA $\mathcal{B}$ defined in Lemma 2 for d, which incorporates DFA $\widehat{\mathcal{B}}$ for $d-1$. A computation on a block $a^{i_s}w_sb^{i_s}$ that starts in q_{s-1} and ends in q_s is illustrated.

Resuming the proof of the lemma, it is now known that the automaton starts reading the suffix c^t of w in the state $q_{t \bmod m}$. Unless this is the state q_0, symbols c cannot be read. Therefore, t must be divisible by m. At the same time, in order to finish reading c^t in the accepting state, t must also be divisible by ℓ. Furthermore, in such a string, each substring w_i must be accepted by $\widehat{\mathcal{B}}$, because $\mathcal{B}$ starts and finishes reading it in the state $\widehat{q}_0$, and therefore lies in $L(\widehat{G}) \cap L(\widehat{\mathcal{B}})$, and its length is at least $|\widehat{w}|$, where $\widehat{w}$ is the shortest string in $L(\widehat{G}) \cap L(\widehat{\mathcal{B}})$.

Then, the shortest string w with these properties must have $t = \ell m$, because ℓ and m are co-prime, and must have each w_i of length exactly $|\widehat{w}|$. This gives the following lower bound on the length of w.

$$|w| > \ell m|\widehat{w}| \geqslant \frac{n}{4} \cdot \frac{n}{4} \cdot \frac{1}{2^{(d-1)^2+3(d-1)-3}} \Big(\frac{n}{2}\Big)^{2(d-1)} =$$

$$= \frac{n^2}{16} \cdot \frac{1}{2^{d^2+d-5}} \cdot \frac{n^{2d-2}}{2^{2d-2}} = \frac{1}{2^{d^2+3d-3}} n^{2d}$$

The estimation uses that ℓ and m are at least $\frac{n}{4}$, and invokes the induction hypothesis to get a lower bound on $|\widehat{w}|$. □

Theorem 2. *For every $d \geqslant 1$, there is a grammar G of bounded tree dimension d, such that for every $n \geqslant 2^{d+1}$ there is an n-state partial DFA $\mathcal{B}$, such that the shortest string w in $L(G) \cap L(\mathcal{B})$ is of length greater than $\frac{1}{2^{d^2+d-3}3^{2d}} n^{2d}$.*

Proof. Let G be the grammar given for d by Lemma 2. Let $2^d r \leqslant n < 2^d(r+1)$, for some integer r. Then $r \geqslant 2$ (for otherwise $n < 2^{d+1}$), and $2^d r \geqslant 2^{d+1}$.

Since $2^d r$ is divisible by 2^d, by Lemma 2, there is a DFA $\mathcal{B}$ with $2^d r \leqslant n$ states, such that the length of the shortest string w in $L(G) \cap L(\mathcal{B})$ is at least $\frac{1}{2^{d^2+3d-3}}(2^d r)^{2d}$. This is the desired n-state DFA.

The inequality $n < 2^d(r+1)$ implies that $n < 2^d \frac{3r}{2}$, because $r+1$ is at most $\frac{3r}{2}$ for $r \geqslant 2$. Then $2^d r > \frac{2}{3}n$, and the lower bound on the length of w is expressed as a function of n as follows.

$$|w| \geqslant \frac{1}{2^{d^2+3d-3}}(2^d r)^{2d} > \frac{1}{2^{d^2+3d-3}}\big(\frac{2}{3}n\big)^{2d} = \frac{1}{2^{d^2+d-3}3^{2d}}n^{2d}$$

□

For finite automata with fewer than 2^{d+1} states, no lower bounds are given, as the construction in the proof relies on having sufficiently long cycles in the automata.

Overall, the rational index of grammars with tree dimension bounded by d is $\Theta(n^{2d})$ in the worst case.

5 Tree dimension in multi-component grammars

Multi-component grammars define the structure of a sentence w by splitting it into constituents that are finite k-tuples of disjoint substrings of w, for k bounded by a constant. In other words, these are substrings with $k-1$ gaps, and these constituents can be joined to each other by concatenating their components from any sides. In all other respects, these grammars are the same as ordinary (context-free) grammars.

Definition 2 (Vijay-Shankar, Weir, Joshi [30]; Seki, Matsumura, Fujii, Kasami [27]). *A multi-component grammar is a quintuple* $G = (\Sigma, N, \mathrm{rank}, R, S)$*, where*

- Σ *is the alphabet of the language being described;*
- N *is the set of nonterminal symbols;*
- $\mathrm{rank}\colon N \to \mathbb{N}$ *is a function that defines the number of components in each nonterminal symbol, so that if* $\mathrm{rank}\, A = k$*, then* A *describes* k*-tuples of substrings;*
- R *is a set of grammar rules, each of the form*

$$A(\alpha_1, \ldots, \alpha_{\mathrm{rank}\, A}) \leftarrow B_1(x_{1,1}, \ldots, x_{1,\mathrm{rank}\, B_1}), \ldots, B_\ell(x_{\ell,1}, \ldots, x_{\ell,\mathrm{rank}\, B_\ell}), \qquad (*)$$

  where $\ell \geqslant 0$*, the variables* $x_{i,j}$ *are pairwise distinct,* $\alpha_1, \ldots, \alpha_{\mathrm{rank}\, A}$ *are strings over symbols from* Σ *and variables* $x_{i,j}$*, and each variable* $x_{i,j}$ *occurs in* $\alpha_1 \ldots \alpha_{\mathrm{rank}\, A}$ *exactly once;*
- *a nonterminal symbol* $S \in N$ *of dimension 1 is the "initial symbol", that is, the category of all well-formed sentences defined by the grammar.*

A grammar is seen as a logical system for proving elementary propositions of the form $A(w_1, \ldots, w_k)$*, with* $k = \operatorname{rank} A$ *and* $w_1, \ldots, w_k \in \Sigma^*$*, meaning that the given* k*-tuple of strings has the property* A*. A proof proceeds using the rules in* R*, with each rule (*) treated as a schema for derivation rules, for any strings substituted for all variables* $x_{i,j}$*.*

$$\frac{B_1(x_{1,1}, \ldots, x_{1,\operatorname{rank} B_1}) \quad \ldots \quad B_\ell(x_{\ell,1}, \ldots, x_{\ell,\operatorname{rank} B_\ell})}{A(\alpha_1, \ldots, \alpha_{\operatorname{rank} A})}$$

The language generated by the grammar, denoted by $L(G)$*, is the set of all such strings* w *that the proposition* $S(w)$ *can be derived in one or more such steps.*

The rank of a grammar, $\operatorname{rank} G$*, is the largest rank of a nonterminal symbol. A multi-component grammar of rank* k *shall be called a* k*-component grammar.*

Whenever a string w is generated by G, the derivation of a proposition $S(w)$ forms a *parse tree*. Each node in the tree is labelled with a proposition $A(w_1, \ldots, w_k)$, where $k = \operatorname{rank} A$ and $w_1, \ldots, w_k$ are substrings of w. Every node has a corresponding rule (*), by which the proposition is derived, and the direct successors of this node are labelled with $B_1(x_{1,1}, \ldots, x_{1,\operatorname{rank} B_1})$, …, $B_\ell(x_{\ell,1}, \ldots, x_{\ell,\operatorname{rank} B_\ell})$, as in the definition of a derivation step.

Example 1. The language $L = \{\, a^m b^n c^m d^n \mid m, n \in \mathbb{N} \,\}$ is defined by the following 2-component grammar.

$$\begin{aligned} S(xy) &\leftarrow A(x, y) \\ A(x, byd) &\leftarrow A(x, y) \\ A(x, y) &\leftarrow B(x, y) \\ B(\varepsilon, \varepsilon) &\leftarrow \\ B(ax, cy) &\leftarrow B(x, y) \end{aligned}$$

Here B defines all pairs (a^m, c^m), with $m \geqslant 0$, and A defines all pairs $(a^m, b^n c^m d^n)$, with $m, n \geqslant 0$. Finally S concatenates every such pair to a string in L.

Tree dimension. The conventional definition of parse trees for multi-component grammars does not include the terminal symbols in the tree structure, and this makes it incompatible with ordinary parse trees in terms of tree dimension. To keep the definitions compatible, for each node in which a rule of the form (*) is applied, every occurrence of a terminal symbol on the left-hand side of this rule shall be regarded as an extra leaf attached to this node, along with proper children labelled with B_1, …, B_ℓ. Each of these extra leaves has dimension 0. The dimension of every node is then defined inductively on the dimensions of its children, as usual.

Definition 3 (Grammars of bounded tree dimension). *A multi-component grammar* G *is said to be of tree dimension bounded by* d *if every parse tree has dimension at most* d*. The least such constant* d *is called the dimension of* G*, denoted by* $\dim G = d$*.*

6 The Chytil–Monien lemma for multi-component grammars

The rational index of every language defined by a multi-component grammar of a bounded tree dimension is bounded by a polynomial that depends both on the tree dimension and on the maximum number of components in the grammar. This is proved by a generalization of the proof in Section 3, from what was essentially the one-component case to the case of up to k components.

The first step is to adapt the Chytil–Monien lemma [6, Lem. 7].

Lemma 3. *Let $G = (\Sigma, N, \mathrm{rank}, R, S)$ be a multi-component grammar of rank k, let r be the maximum number of nonterminal symbols on the right-hand side of a rule, let ℓ be the maximal number of terminal symbols on the left-hand side of a rule, and assume that there exists a parse tree of dimension $d \geqslant 1$ in this grammar. Then the grammar defines some string of length less than $\frac{\ell+r-1}{r-1}(|N|(r-1)+1)^d$.*

Proof. Let $f(d)$ be the length of the shortest string defined by a grammar of dimension d.

It is claimed that $f(0) \leqslant 1$ and $f(d) \leqslant \big(|N|(r-1)+1\big) \cdot f(d-1) + |N| \cdot \ell$.

A parse tree has dimension $d = 0$ if it is a path formed of nodes of dimension 0. The last node on the path defines at most one symbol, for otherwise it would have dimension greater than zero. Thus, the length of the shortest string $f(0)$ is at most 1.

Let $d \geqslant 1$. Recall the proof of Lemma 1 and consider the path from the root of the parse tree which passes through the nodes of dimension d, where every node except the last one has one child of dimension d and other children of dimension $d-1$. As in the proof of Lemma 1, the length of the path is bounded by $|N|$ (otherwise there is a shorter string). At each node on the path (except the last one) there are at most $r-1$ subtrees of dimension $d-1$. The last node of the path has at most r children of dimension $d-1$. The length of the shortest string defined by a subtree of dimension $d-1$ is $f(d-1)$. Additionally, at each node, at most ℓ extra leaves listed on the left-hand side of a rule are appended; in total, there are at most $|N|\ell$ such leaves.

Putting all together, the length of the shortest string is bounded as follows.

$$f(d) \leqslant \big(|N|(r-1)+1\big) \cdot f(d-1) + |N| \cdot \ell$$

Thus, the length of the shortest string is represented by a recurrence relation of the form $x_0 = 1$, $x_d = ax_{d-1} + b$, where $a = |N|(r-1)+1$ and $b = |N| \cdot \ell$. The upper bound on the solution of this recurrence relation is as follows.

$$\begin{aligned} x_d = a^d + b\frac{a^d-1}{a-1} \leqslant a^d + b\frac{a^d}{a-1} = \Big(\frac{b}{a-1}+1\Big)a^d = \\ = \Big(\frac{|N| \cdot \ell}{|N| \cdot (r-1)}+1\Big)\big(|N|(r-1)+1\big)^d = \frac{\ell+r-1}{r-1}\big(|N|(r-1)+1\big)^d \end{aligned}$$

□

The above lemma is used to prove an upper bound similar to the one in Theorem 1. The bound gets raised into the power of k, the maximum number of components in a grammar, because the construction for the intersection with a regular languages produces more nonterminal symbols.

Theorem 3. *Let G be an multi-component grammar of rank k and of tree dimension bounded by d, and let $\mathcal{A}$ be an NFA with n states, with non-empty intersection $L(G) \cap L(\mathcal{A})$. Then the length of the shortest string in $L(G) \cap L(\mathcal{A})$ is at most $O(n^{2kd})$.*

Proof. Denote the set of nonterminal symbols in G by N. As proved by Seki et al. [27, Thm. 3.9], for a k-component grammar G and an NFA $\mathcal{A}$, the intersection $L(G) \cap L(\mathcal{A})$ is defined by another k-component grammar with at most $|N| \cdot n^{2k}+1$ nonterminal symbols. Their construction generalizes that by Bar-Hillel et al. [2], and produces nonterminal symbols of the form $(A, p_1, q_1, \ldots, p_{\operatorname{rank} A}, q_{\operatorname{rank} A})$, where $A \in N$, $\operatorname{rank} A$ is a rank of the nonterminal A, and p_i, q_i are states of the automaton, which it enters before and after reading the corresponding substrings. The transformation preserves the structure of the parse trees: each tree in G' has the corresponding tree in G that differs only in the labels of its nodes, and has the same dimension.

Since the intersection is non-empty, G' defines at least one parse tree, which has the same dimension as some parse tree in G, that is, at most d. Then, by Lemma 3, the shortest string defined by G' is of length at most $\frac{\ell+r-1}{r-1}((|N| \cdot n^{2k}+1)(r-1)+1)^d$, where ℓ and r are constants depending on G'. This number is estimated as follows.

$$\frac{\ell+r-1}{r-1}\big((|N| \cdot n^{2k}+1)(r-1)+1\big)^d = |N| \cdot (\ell+r-1) \cdot n^{2kd} + o(n^{2kd}) = O(n^{2kd})$$

□

7 A lower bound for multi-component grammars

This section establishes a lower bound on the rational index of multi-component grammars that asymptotically matches the bound in Theorem 3. The main part of the argument is the following lemma that generalizes Lemma 2 to the multi-component case.

Lemma 4. *For every $d \geqslant 1$, $k \geqslant 2$, there is a k-component grammar G of bounded tree dimension d, such that for every $n \geqslant 8 \cdot 2^d k^2 \ln 8k$ divisible by $2^d k$ there is an n-state partial DFA $\mathcal{B}$, such that the shortest string w in $L(G) \cap L(\mathcal{B})$ is of length at least $\frac{1}{k^{2kd-1} \cdot 2^{kd^2+5kd-2k-1}} n^{2kd}$.*

Proof. For every fixed k, the proof is by an induction on d. For every d, a grammar is constructed first. Next, for each n divisible by $2^d \cdot k$, an n-state DFA with a unique initial state and a unique accepting state is constructed.

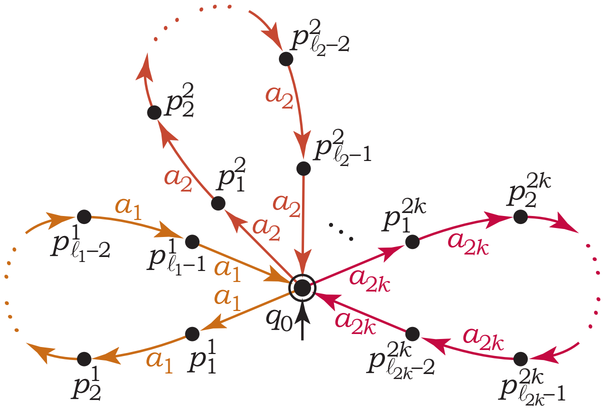

Fig. 6. DFA $\mathcal{B}$ defined in Lemma 4 for $d = 1$ and fixed $k \geqslant 0$.

Basis: $d = 1$. Consider k-component grammar G which defines a language $L = \{\, a_1^i a_2^i \dots a_{2k}^i \mid i \geqslant 1 \,\}$.

$$\begin{aligned} S(x_1 \dots x_k) &\leftarrow A(x_1, \dots, x_k) \\ A(a_1 x_1 a_2, \dots, a_{2k-1} x_k a_{2k}) &\leftarrow A(x_1, \dots, x_k) \\ A(a_1 a_2, \dots, a_{2k-1} a_{2k}) &\leftarrow \end{aligned}$$

The dimension of G is at most 1 for every k. Indeed, every node using the rule of the form $A(a_1a_2, \dots, a_{2k-1}a_{2k}) \leftarrow$ has $2k$ leaves of dimension 0, thus its dimension is 1. The application of the rule $A(a_1x_1a_2, \dots, a_{2k-1}x_ka_{2k}) \leftarrow A(x_1, \dots, x_k)$ does not change the dimension of the node labelled by nonterminal A, because such node has one child labelled by nonterminal A of dimension at most 1 and $2k$ extra children, each of dimension 0. Clearly, the rule $S(x_1 \dots x_k) \leftarrow A(x_1, \dots, x_k)$ does not affect the dimension of the parse tree.

The automaton $\mathcal{B}$ is constructed as a generalization of the automaton $\mathcal{B}$ for context-free grammars of dimension 1 described in Section 4.

Let $n \geqslant 16k^2 \ln 8k$ and n be divisible by $2k$, and let R_{2k} be $2k$th Ramanujan prime [25], that is the least number for which there are at least R_{2k} primes between $\frac{x}{2}$ and x for all $x \geqslant R_{2k}$. According to Sondow [28], R_{2k} is at most $8k \ln 8k$.

Let $\ell_1, \dots, \ell_{2k}$ be different primes, whose values are between $\frac{n}{4k}$ and $\frac{n}{2k}$; such primes exist because $\frac{n}{2k} \geqslant 8k \ln 8k$.

Clearly, $\sum_{i=1}^{2k} \ell_i$ is at most $2k \cdot \frac{n}{2k} = n$. Then the automaton $\mathcal{B}$ over the alphabet $\{a_1, \dots, a_{2k}\}$ is constructed as follows. It consists of $2k$ cycles sharing a common node, which is initial and final state at the same time. Every ith cycle is a cycle of length ℓ_i having transitions by a_i. The automaton has $1 + \sum_{i=1}^{2k}(\ell_i - 1) \leqslant n$ states. DFA $\mathcal{B}$ is illustrated in Figure 6.

Consider the length of the shortest string in $w \in L(G) \cap L(\mathcal{B})$. Every such string is of the form $a_1^i \dots a_{2k}^i$ for $i \geqslant 1$. The automaton accepts the string of this form iff i is a multiple of all lengths of its cycles $\ell_1, \ldots, \ell_{2k}$. All lengths of the cycles are primes by construction, thus the least such i is equal to $\prod_{i=1}^{2k} \ell_i$. Then the length of w is estimated as follows.

$$|w| = 2k \prod_{i=1}^{2k} \ell_i > 2k \prod_{i=1}^{2k} \frac{n}{4k} = 2k \frac{n^{2k}}{(4k)^{2k}} = \frac{1}{k^{2k-1} \cdot 2^{4k-1}} n^{2k}$$

Induction step: $dim(G) = d$. By the induction hypothesis, there exists a k-component grammar $\hat{G} = (\widehat{\Sigma}, \widehat{N}, \mathrm{rank}, \widehat{R}, \widehat{S})$ of tree dimension bounded by $d-1$ that satisfies the condition of the lemma. A k-component grammar of tree dimension bounded by d is then defined over the alphabet $\Sigma = \widehat{\Sigma} \cup \{a, b, c_1, \ldots, c_{2k-1}\}$, where $a, b, c_1, \ldots, c_{2k-1} \notin \widehat{\Sigma}$. The set of nonterminal symbols of new grammar is $N = \widehat{N} \cup \{S, A, C\}$, where all nonterminal symbols from $\widehat{N}$ preserve their original rank, while new nonterminals $A, C, S \notin \widehat{N}$ have $\mathrm{rank}\, S = \mathrm{rank}\, A = 1$ and $\mathrm{rank}\, C = k$. The rules of the grammar G include all rules from $\widehat{G}$ and the following additional rules.

$$\begin{aligned}
S(x_1 \ldots x_k) &\leftarrow C(x_1, \ldots, x_k) \\
C(yx_1c_1, c_2x_2c_3, \ldots, c_{2k-2}x_kc_{2k-1}) &\leftarrow A(y), C(x_1, \ldots, x_k) \\
C(yc_1, c_2c_3, \ldots, c_{2k-2}c_{2k-1}) &\leftarrow A(y) \\
A(ayb) &\leftarrow A(y) \\
A(ayb) &\leftarrow \widehat{S}(y).
\end{aligned}$$

Here, the nonterminal A defines all substrings of the form a^iub^i, where $i \geqslant 1$ and $u \in L(\widehat{G})$, as in the grammar from Lemma 2. A nonterminal symbol C generates all k-tuples of the form $(v_1 \ldots v_t c_1^t, c_2^t c_3^t, \ldots, c_{2k-2}^t c_{2k-1}^t)$, where $t \geqslant 1$ and $v_1, \ldots, v_t$ are strings defined by nonterminal A. Finally, the nonterminal S defines concatenations of all such k-tuples, so that the language generated by grammar G is of the following form.

$$\{\, a^{i_1}w_1b^{i_1} \ldots a^{i_t}w_tb^{i_t}c_1^t \ldots c_{2k-1}^t \mid t \geqslant 1,\ i_1, \ldots, i_t \geqslant 1,\ w_1, \ldots, w_t \in L(\widehat{G}) \,\}$$

Next it is claimed that the dimension of the grammar G is at most d. The dimension of every tree with the root labelled by a nonterminal $\widehat{S}$ is at most $d-1$, that is the maximal dimension of grammar $\widehat{G}$.

Then each subtree with the root labelled by A has dimension bounded by $d-1$. The dimension of subtrees with the root labelled by C is at most d, because the lowest such subtree has the same dimension $d-1$ as subtree labelled by A, and each next subtree has one child labelled by A of dimension $d-1$ and one child C of dimension at most d. The dimension of the root labelled by S is equal to the dimension of the topmost subtree labelled by nonterminal C, and, therefore, is at most d.

Let $n \geqslant 8 \cdot 2^d k^2 \ln 8k$ and n be divisible by $2^d k$. As $\frac{n}{2}$ is greater than $8 \cdot 2^{d-1} k^2 \ln 8k$ and is divisible by $2^{d-1}k$, by the induction hypothesis there exists an $\frac{n}{2}$-state automaton $\widehat{\mathcal{B}} = (\widehat{Q}, \widehat{\Sigma}, \widehat{\delta}, \widehat{q}_0, \{\widehat{q}_0\})$ for grammar $\widehat{G}$ such that the length of the shortest string $\widehat{w}$ in the intersection $L(\widehat{G}) \cap L(\widehat{\mathcal{B}})$ has the length at least $\frac{1}{k^{2k(d-1)-1} \cdot 2^{k(d-1)^2+5k(d-1)-2k-1}} n^{2k(d-1)}$.

Then n-state DFA $\mathcal{B} = (\Sigma, Q, q_0, \delta, \{q_0\})$ is constructed as follows. It consists of DFA $\widehat{\mathcal{B}}$ with at most $\frac{n}{2}$ states, and the second half of the states is used to construct cycles. Let $\ell_1, \ldots, \ell_{2k}$ be primes, such that $\frac{n}{8k} \leqslant \ell_i \leqslant \frac{n}{4k}$ for every i. These numbers define the lengths of the cycles, and the set of states of the new automaton is as follows.

$$Q = \widehat{Q} \cup \{q_0, \underbrace{p^1_1, \ldots, p^1_{\ell_1 - 1}}_{\text{cycle by } c_1}, \ldots, \underbrace{p^{2k-1}_1, \ldots, p^{2k-1}_{\ell_{2k-1}-1}}_{\text{cycle by } c_{2k-1}}, \underbrace{q_1, \ldots, q_{\ell_{2k}-1}}_{\text{chain by } a \text{ and } b}\}$$

Overall, at most $2k\frac{n}{4k} = \frac{n}{2}$ states are used for constructing of cycles.

The transitions are similar to those in the proof of Lemma 2, with the only difference that instead of one cycle by c the automaton in this proof has $2k-1$ cycles by c_1, …, c_{2k-1}. The initial state is q_0, it has a transition by a to the initial state of $\widehat{\mathcal{B}}$, continued by a transition by b to q_1.

$$\delta(q_0, a) = \widehat{q}_0$$
$$\delta(\widehat{q}_0, b) = q_1$$

A chain of transitions by a and another chain by b are as in Lemma 2; they use states from q_1 to $q_{\ell_{2k}-1}$.

$$\delta(q_i, a) = q_{i-1}, \qquad \text{with } 1 \leqslant i \leqslant \ell_{2k} - 1$$
$$\delta(q_i, b) = q_{i+1 \bmod \ell_{2k}}, \qquad \text{with } 1 \leqslant i \leqslant \ell_{2k} - 1$$

For each i, with $1 \leqslant i \leqslant 2k-1$, there is a cycle by c_i in the states $q_0 = p^i_0$, p^i_1, …, $p^i_{\ell_i - 1}$.

$$\delta(p^i_j, c_i) = p^i_{j+1 \bmod \ell_i}, \qquad \text{with } 0 \leqslant j \leqslant \ell_i - 1$$

This automaton $\mathcal{B}$ is illustrated in Figure 7.

Every string w in $L(G) \cap L(\mathcal{B})$ is of the form $w = a^{i_1} w_1 b^{i_1} \ldots a^{i_t} w_t b^{i_t} c_1^t \ldots c_{2k-1}^t$, with $t \geqslant 1$, $i_1, \ldots, i_t \geqslant 1$ and $w_1, \ldots, w_t \in L(\widehat{G})$, because it is defined by G. The string w is also accepted by $\mathcal{B}$, and, up to the first c_1, the automaton's computation proceeds as in the proof of Lemma 2.

Claim. After reading each prefix $a^{i_1} w_1 b^{i_1} \cdots a^{i_s} w_s b^{i_s}$ of w, with $s \in \{0, \ldots, t\}$, the automaton comes to the state $q_{s \bmod \ell_{2k}}$.

Thus, the automaton $\mathcal{B}$ reaches the first c_1 in the state $q_{t \bmod \ell_{2k}}$. For the automaton to proceed further, this state must be q_0, and hence t is divisible by ℓ_{2k}. Next, the automaton reads each of the blocks c_1^t, …, c_{2k-1}^t, and must begin

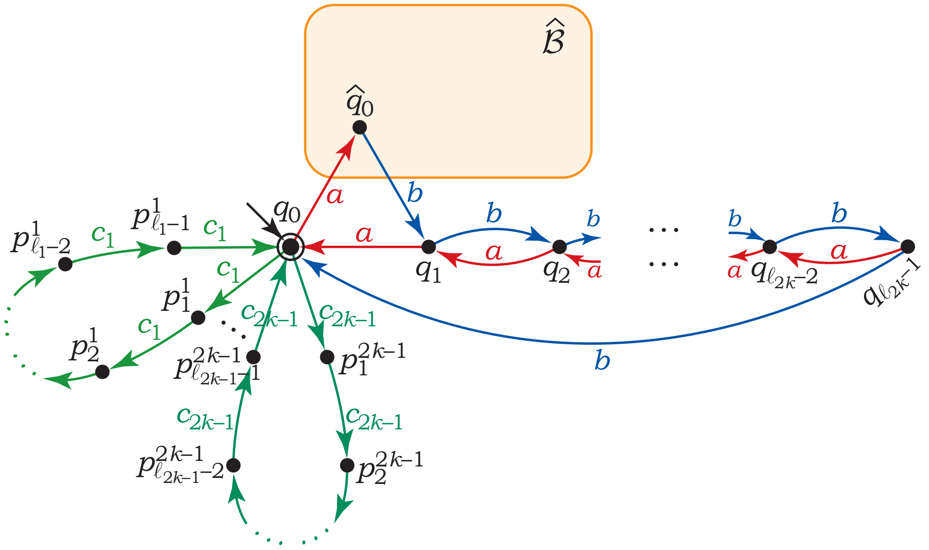

Fig. 7. DFA $\mathcal{B}$ defined in Lemma 4 for $d \geqslant 2$ and fixed k, which incorporates DFA $\widehat{\mathcal{B}}$ for $d-1$ and k.

and end reading each of them in the same state q_0. Therefore, t must also be divisible by all numbers $\ell_1, \dots, \ell_{2k-1}$. As all ℓ_i are primes, t is divisible by their product $\prod_{i=1}^{2k} \ell_i$.

As in the proof of Lemma 2, the automaton begins and ends reading each substring w_i in the state $\widehat{q}_0$, and hence all these strings are in $L(\widehat{G}) \cap L(\widehat{\mathcal{B}})$.

The shortest w satisfying the above constraints then has $t = \prod_{i=1}^{2k} \ell_i$, and its length is at least $t \cdot |\widehat{w}|$, where $\widehat{w}$ is the shortest string in $L(\widehat{G})$. Using a lower bound on $|\widehat{w}|$ given by the induction hypothesis, the lower bound on the length of w is estimated as follows.

$$\begin{aligned}
|w| \geqslant \left(\prod_{i=1}^{2k} \ell_i\right) \cdot |\widehat{w}| \geqslant \\
\geqslant \left(\prod_{i=1}^{2k} \frac{n}{8k}\right) \cdot \frac{1}{k^{2k(d-1)-1} \cdot 2^{k(d-1)^2+5k(d-1)-2k-1}} \left(\frac{n}{2}\right)^{2k(d-1)} = \\
= \frac{n^{2k}}{2^{6k}k^{2k}} \cdot \frac{1}{k^{2kd-2k-1} \cdot 2^{k(d-1)^2+5k(d-1)-2k-1+2k(d-1)}} n^{2k(d-1)} = \\
= \frac{1}{k^{2kd-1} \cdot 2^{kd^2+5kd-2k-1}} n^{2kd}.
\end{aligned}$$

□

Theorem 4. *For every $d \geqslant 1$, $k \geqslant 2$ there is a grammar G of bounded tree dimension d, such that for every $n \geqslant 8 \cdot 2^d k^2 \ln 8k$ there is an n-state partial DFA $\mathcal{B}$, such that the shortest string w in $L(G) \cap L(\mathcal{B})$ is of length at least $\frac{k}{2^{kd^2-kd-2k-1} \cdot (8k+1)^{2kd}} n^{2kd}$.*

Proof. Let G be the grammar given for d by Lemma 4. Let $2^d kr \leqslant n < 2^d k(r+1)$, for some integer r. Then $r \geqslant 8k \ln 8k$ (for otherwise n would be less than $8 \cdot 2^d k^2 \ln 8k$), and $2^d kr \geqslant 2^{d+1}k$.

Since $2^d kr$ is divisible by $2^d k$ and is at least $8 \cdot 2^d k^2 \ln 8k$, by Lemma 4, there is a DFA $\mathcal{B}$ with $2^d kr \leqslant n$ states, such that the length of the shortest string w in $L(G) \cap L(\mathcal{B})$ is at least $\frac{1}{k^{2kd-1} \cdot 2^{kd^2+5kd-2k-1}} \cdot (2^d kr)^{2kd}$. This is the desired n-state DFA.

Since $r > 8k$, the inequality $n < 2^d k(r+1)$ implies that $n < 2^d kr \frac{8k+1}{8k}$ and accordingly $2^d kr > n \frac{8k}{8k+1}$. Then, the lower bound on the length of w is expressed as a function of n as follows.

$$\begin{aligned}
|w| &\geqslant \frac{1}{k^{2kd-1} \cdot 2^{kd^2+5kd-2k-1}} (2^d kr)^{2kd} > \\
&> \frac{1}{k^{2kd-1} \cdot 2^{kd^2+5kd-2k-1}} \left(n \cdot \frac{8k}{8k+1}\right)^{2kd} = \\
&= \frac{1}{k^{2kd-1} \cdot 2^{kd^2+5kd-2k-1}} \cdot \frac{2^{6kd} \cdot k^{2kd}}{(8k+1)^{2kd}} \cdot n^{2kd} = \\
&= \frac{k}{2^{kd^2-kd-2k-1}(8k+1)^{2kd}} \cdot n^{2kd}
\end{aligned}$$

□

The final conclusion on the rational index of k-component grammars with tree dimension bounded by d is that it is in the worst case of the order $\Theta(n^{2kd})$.

8 Rational indices for some language families

For a few families of grammars known in the literature, the results of this paper imply some bounds on their rational index.

Superlinear languages. A grammar $G = (\Sigma, N, R, S)$ is *superlinear* (Brzozowski [4]) if its nonterminal symbols split into two classes, $N = N_{lin} \cup N_{nonlin}$, where rules for each nonterminal $A \in N_{lin}$ are of the form $A \to uBv$ or $A \to w$, with $B \in N_{lin}$, $u, v, w \in \Sigma^*$, while rules for a nontermial $A \in N_{nonlin}$ are of the form $A \to \alpha B \beta$, with $B \in N$ and $\alpha, \beta \in (\Sigma \cup N_{lin})^*$. A language is *superlinear* if it is generated by some superlinear grammar.

Corollary 1. *For every superlinear grammar G, the rational index $\rho_{L(G)}$ is at most $O(n^4)$.*

Proof. Parse trees in a superlinear grammar G have dimension at most 2. Then, by Theorem 1, the rational index $\rho_{L(G)}$ is bounded by $O(n^4)$.

Turning to a lower bound, note that the grammar constructed in Theorem 2 for $d = 2$ is actually superlinear.

Corollary 2. *There exists a superlinear grammar G with rational index $\rho_{L(G)}(n) \geqslant \frac{1}{648}n^4$.*

Bounded-oscillation languages. The notion of oscillation of runs in pushdown automata, applicable to Turing machines with auxiliary pushdown tape, was introduced by Wechsung [31]. *Languages with oscillation bounded by k* are then a generalization of the linear languages (as one-turn pushdown automata are those with oscillation bounded by $k = 1$).

This family was later studied by Ganty and Valput [11], who introduced the corresponding notion of oscillation in parse trees of grammars. Among other results, they prove that oscillation of a parse tree is closely related to its dimension.

Lemma 5 (Ganty and Valput [11]). *Let $G = (\Sigma, N, R, S)$ be a grammar in the Chomsky normal form, and let t be a parse tree in G. Then, $\operatorname{osc} t - 1 \leqslant \dim t \leqslant 2 \operatorname{osc} t$.*

Thus, k-bounded-oscillation grammars have dimension of parse trees bounded by $2k$, and Theorem 1 gives the following upper bound on the rational index of these languages.

Corollary 3. *Let L be a k-bounded-oscillation language. Then $\rho_L(n) = O(n^{4k})$.*

Linear (non-branching) multi-component grammars. A multi-component grammar is called *linear* [9], or, alternatively, *non-branching* [15] if each rule has no more than r occurrences of nonterminals in its body. A unary branching grammar is often called non-branching or linear.

Corollary 4. *For every linear k-component grammar G, the rational index $\rho_{L(G)}$ is at most $O(n^{2k})$.*

Proof. Parse trees in a non-branching k-component grammar G have dimension at most 1. Then, by Theorem 3, the rational index $\rho_{L(G)}$ is bounded by $O(n^{2k})$.

9 Adaptation to other grammar families?

Roughly speaking, it was proved that if the branching of parse trees in an ordinary (context-free) grammar is restricted, then the rational index of the language is polynomial of a certain degree that depends on the bound on the branching. The result is generalized to multi-component grammars, where the degree of the polynomial additionally depends on the rank of the grammar. The question is, could this kind of results hold for any other grammar families of interest, under suitable restrictions on the structure of their parse trees?

LL(k), LR(k) and unambiguous grammars. LL(k) grammars, LR(k) grammars and unambiguous grammars are classical subfamilies of grammars that are notable for their beautiful theoretical properties and diverse practical applications. Under the restriction of d-bounded dimension of parse trees, the rational index of all these grammars is subject to the upper bound $O(n^{2d})$ given in Theorem 1. It turns out that the lower bound $\Omega(n^{2d})$ holds as well, because all grammars constructed in Theorem 2 can be transformed to the most restricted of these subfamilies: *LL(1)-grammars in the Greibach normal form*, also known as *simple grammars.*

A grammar $G = (\Sigma, N, R, S)$ is LL(1) in the Greibach normal form if every rule is of the form $A \to a\alpha$, with $a \in \Sigma$ and $\alpha \in (\Sigma \cup N)^*$, and if, furthermore, there is at most one such rule for every pair (A, a). Both grammars in the proof of Theorem 2 can be rewritten as LL(1) grammars in the Greibach normal form: the first language $\{\, a^i b^i \mid i \geqslant 1 \,\}$ is defined by a grammar with the rules

$$\begin{aligned} S &\to aT \\ T &\to aTb \mid b \end{aligned}$$

The second language consists of all strings $a^{i_1} w_1 b^{i_1} \dots a^{i_t} w_t b^{i_t} c^t$, with $t \geqslant 1$, $i_1, \dots, i_t \geqslant 1$, and with $w_1, \dots, w_t$ defined by a nonterminal symbol $\widehat{S}$ using some rules that already satisfy the requirement of being LL(1) in the Greibach normal form. This language is defined by the following grammar.

$$\begin{aligned} S &\to aAT \\ A &\to aAb \\ A &\to \widehat{S}b && (\widehat{S} \to \sigma \in \widehat{R}) \\ T &\to c \mid aATc \end{aligned}$$

An interested reader can verify that the dimensions of parse trees in the proof of Theorem 2 are the same for this grammar, and hence the argument remains valid with this grammar substituted.

Conjunctive and Boolean grammars. Conjunctive grammars are an extension of ordinary (context-free) grammars with a conjunction operation, so that each rule is of the form $A \to \alpha_1 \& \dots \& \alpha_m$, with $m \geqslant 1$ and $\alpha_1, \dots, \alpha_m \in (\Sigma \cup N)^*$; such a rule states that every string defined by each α_i is then defined by A. Boolean grammars further extend the model with a conjunction operation. These grammars preserve the notion of a parse tree, which becomes an acyclic graph, and are primarily notable for generalizations of classical parsing algorithms to handle these grammars. An interested reader is referred to an up-to-date survey of conjunctive grammars [22] and to an earlier survey of Boolean grammars [21] for more details.

The question is, could any restriction on the structure of branching in parse trees for these grammars lead to any upper bound on the rational index? Unfortunately, no way of adapting the results of this paper to these models is anticipated. Consider that the language of valid accepting computations of a

Turing machine (VALC) is an intersection of two linear languages; a conjunctive grammar can define such an intersection by including two linear grammars, with initial symbols A and B, and adding a rule $S \to A\&B$. Parse trees in this grammar will have the simplest possible structure for a conjunctive grammar, with two linear trees coming out of the root, and with each leaf shared between these two trees. However, since checking the emptiness of VALC is undecidable, there is no *a priori* recursive upper bound on the rational index.

References

1. Alpoge, L., Ang, T., Schaeffer, L., Shallit, J.O.: Decidability and shortest strings in formal languages. In: Holzer, M., Kutrib, M., Pighizzini, G. (eds.) Descriptional Complexity of Formal Systems - 13th International Workshop, DCFS 2011, Gießen/Limburg, Germany, July 25-27, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6808, pp. 55–67. Springer (2011). `https://doi.org/10.1007/978-3-642-22600-7_5`
2. Bar-Hillel, Y., Perles, M., Shamir, E.: On formal properties of simple phreise structure grammars. STUF - Language Typology and Universals **14**(1-4), 143 – 172 (01 Apr 1961). `https://doi.org/10.1524/stuf.1961.14.14.143`
3. Boasson, L., Courcelle, B., Nivat, M.: The rational index: A complexity measure for languages. SIAM J. Comput. **10**(2), 284–296 (1981). `https://doi.org/10.1137/0210020`, `https://doi.org/10.1137/0210020`
4. Brzozowski, J.A.: Regular-like expressions for some irregular languages. In: 9th Annual Symposium on Switching and Automata Theory, Schenectady, New York, USA, October 15-18, 1968. pp. 278–286. IEEE Computer Society (1968). `https://doi.org/10.1109/SWAT.1968.24`
5. Chistikov, D., Czerwinski, W., Hofman, P., Pilipczuk, M., Wehar, M.: Shortest paths in one-counter systems. Log. Methods Comput. Sci. **15**(1) (2019). `https://doi.org/10.23638/LMCS-15(1:19)2019`
6. Chytil, M., Monien, B.: Caterpillars and context-free languages. In: Choffrut, C., Lengauer, T. (eds.) STACS 90, 7th Annual Symposium on Theoretical Aspects of Computer Science, Rouen, France, February 22-24, 1990, Proceedings. Lecture Notes in Computer Science, vol. 415, pp. 70–81. Springer (1990). `https://doi.org/10.1007/3-540-52282-4_33`
7. Dobronravov, E., Dobronravov, N., Okhotin, A.: On the length of shortest strings accepted by two-way finite automata. Fundam. Informaticae **180**(4), 315–331 (2021). `https://doi.org/10.3233/FI-2021-2044`
8. Ellul, K., Krawetz, B., Shallit, J.O., Wang, M.: Regular expressions: New results and open problems. J. Autom. Lang. Comb. **10**(4), 407–437 (2005). `https://doi.org/10.25596/jalc-2005-407`
9. Engelfriet, J.: Context-free graph grammars. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, Volume 3: Beyond Words, pp. 125–213. Springer (1997). `https://doi.org/10.1007/978-3-642-59126-6_3`
10. Esparza, J., Luttenberger, M., Schlund, M.: A brief history of Strahler numbers. In: Dediu, A., Martín-Vide, C., Sierra-Rodríguez, J.L., Truthe, B. (eds.) Language and Automata Theory and Applications - 8th International Conference, LATA 2014, Madrid, Spain, March 10-14, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8370, pp. 1–13. Springer (2014). `https://doi.org/10.1007/978-3-319-04921-2_1`

11. Ganty, P., Valput, D.: Bounded-oscillation pushdown automata. In: Cantone, D., Delzanno, G. (eds.) Proceedings of the Seventh International Symposium on Games, Automata, Logics and Formal Verification, GandALF 2016, Catania, Italy, 14-16 September 2016. EPTCS, vol. 226, pp. 178–197 (2016). https://doi.org/10.4204/EPTCS.226.13
12. Greenlaw, R., Hoover, H.J., Ruzzo, W.L.: Limits to Parallel Computation: P-completeness Theory. Oxford University Press, Inc., New York, NY, USA (1995)
13. Hellings, J.: Explaining results of path queries on graphs - single-path results for context-free path queries. In: Qin, L., Zhang, W., Zhang, Y., Peng, Y., Kato, H., Wang, W., Xiao, C. (eds.) Software Foundations for Data Interoperability and Large Scale Graph Data Analytics - 4th International Workshop, SFDI 2020, and 2nd International Workshop, LSGDA 2020, held in Conjunction with VLDB 2020, Tokyo, Japan, September 4, 2020, Proceedings. Communications in Computer and Information Science, vol. 1281, pp. 84–98. Springer (2020). https://doi.org/10.1007/978-3-030-61133-0_7
14. Holzer, M., Kutrib, M., Leiter, U.: Nodes connected by path languages. In: Mauri, G., Leporati, A. (eds.) Developments in Language Theory - 15th International Conference, DLT 2011, Milan, Italy, July 19-22, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6795, pp. 276–287. Springer (2011). https://doi.org/10.1007/978-3-642-22321-1_24
15. Kanazawa, M.: Ogden's lemma, multiple context-free grammars, and the control language hierarchy. Inf. Comput. **269** (2019). https://doi.org/10.1016/j.ic.2019.104449
16. Komarath, B., Sarma, J., Sunil, K.S.: On the complexity of l-reachability. In: Jürgensen, H., Karhumäki, J., Okhotin, A. (eds.) Descriptional Complexity of Formal Systems - 16th International Workshop, DCFS 2014, Turku, Finland, August 5-8, 2014. Proceedings. Lecture Notes in Computer Science, vol. 8614, pp. 258–269. Springer (2014). https://doi.org/10.1007/978-3-319-09704-6_23
17. Krymski, S., Okhotin, A.: Longer shortest strings in two-way finite automata. In: Jirásková, G., Pighizzini, G. (eds.) Descriptional Complexity of Formal Systems - 22nd International Conference, DCFS 2020, Vienna, Austria, August 24-26, 2020, Proceedings. Lecture Notes in Computer Science, vol. 12442, pp. 104–116. Springer (2020). https://doi.org/10.1007/978-3-030-62536-8_9
18. Lohrey, M., Rosowski, A., Zetzsche, G.: Membership problems in finite groups. In: Szeider, S., Ganian, R., Silva, A. (eds.) 47th International Symposium on Mathematical Foundations of Computer Science, MFCS 2022, August 22-26, 2022, Vienna, Austria. LIPIcs, vol. 241, pp. 71:1–71:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2022). https://doi.org/10.4230/LIPIcs.MFCS.2022.71
19. Luttenberger, M., Schlund, M.: Convergence of Newton's method over commutative semirings. In: Dediu, A., Martín-Vide, C., Truthe, B. (eds.) Language and Automata Theory and Applications - 7th International Conference, LATA 2013, Bilbao, Spain, April 2-5, 2013. Proceedings. Lecture Notes in Computer Science, vol. 7810, pp. 407–418. Springer (2013). https://doi.org/10.1007/978-3-642-37064-9_36
20. Martynova, O., Okhotin, A.: The maximum length of shortest accepted strings for direction-determinate two-way finite automata. CoRR **abs/2210.00235** (2022). https://doi.org/10.48550/arXiv.2210.00235
21. Okhotin, A.: Conjunctive and boolean grammars: The true general case of the context-free grammars. Comput. Sci. Rev. **9**, 27–59 (2013). https://doi.org/10.1016/j.cosrev.2013.06.001

22. Okhotin, A.: A tale of conjunctive grammars. In: Hoshi, M., Seki, S. (eds.) Developments in Language Theory - 22nd International Conference, DLT 2018, Tokyo, Japan, September 10-14, 2018, Proceedings. Lecture Notes in Computer Science, vol. 11088, pp. 36–59. Springer (2018). https://doi.org/10.1007/978-3-319-98654-8_4
23. Pierre, L.: Rational indexes of generators of the cone of context-free languages. Theor. Comput. Sci. **95**(2), 279–305 (1992). https://doi.org/10.1016/0304-3975(92)90269-L
24. Pierre, L., Farinone, J.: Context-free languages with rational index in $\theta(n^\lambda)$ for algebraic numbers λ. RAIRO Theor. Informatics Appl. **24**, 275–322 (1990). https://doi.org/10.1051/ita/1990240302751
25. Ramanujan, S.: A proof of Bertrand's postulate. Journal of the Indian Mathematical Society **11**(181-182), 27 (1919)
26. Reps, T.W.: Program analysis via graph reachability. Inf. Softw. Technol. **40**(11-12), 701–726 (1998). https://doi.org/10.1016/S0950-5849(98)00093-7
27. Seki, H., Matsumura, T., Fujii, M., Kasami, T.: On multiple context-free grammars. Theor. Comput. Sci. **88**(2), 191–229 (1991). https://doi.org/10.1016/0304-3975(91)90374-B
28. Sondow, J.: Ramanujan primes and Bertrand's postulate. Am. Math. Mon. **116**(7), 630–635 (2009), http://www.jstor.org/stable/40391170
29. Ullman, J.D., Gelder, A.V.: Parallel complexity of logical query programs. Algorithmica **3**, 5–42 (1988). https://doi.org/10.1007/BF01762108
30. Vijay-Shanker, K., Weir, D.J., Joshi, A.K.: Characterizing structural descriptions produced by various grammatical formalisms. In: Sidner, C.L. (ed.) 25th Annual Meeting of the Association for Computational Linguistics, Stanford University, Stanford, California, USA, July 6-9, 1987. pp. 104–111. ACL (1987). https://doi.org/10.3115/981175.981190
31. Wechsung, G.: The oscillation complexity and a hierarchy of context-free languages. In: Budach, L. (ed.) Fundamentals of Computation Theory, FCT 1979, Proceedings of the Conference on Algebraic, Arthmetic, and Categorial Methods in Computation Theory, Berlin/Wendisch-Rietz, Germany, September 17-21, 1979. pp. 508–515. Akademie-Verlag, Berlin (1979)
32. Yannakakis, M.: Graph-theoretic methods in database theory. In: Rosenkrantz, D.J., Sagiv, Y. (eds.) Proceedings of the Ninth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, April 2-4, 1990, Nashville, Tennessee, USA. pp. 230–242. ACM Press (1990). https://doi.org/10.1145/298514.298576